MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

Policy iteration-based optimal control design for nonlinear descriptor systems

Wang, Y.; Wu, J.; Long, C.

TR2016-051 July 2016

Abstract

This paper considers state feedback optimal control design for a class of nonlinear descriptor systems. Prior work either stops at the Hamilton-Jacobian-Bellman equations and thus is non-constructive, or converts the optimal control problem into a large scale nonlinear optimization problem and thus is open-loop control design. This paper proposes a generalized policy iteration algorithm to compute the state feedback optimal control policy in a constructive manner, and presents the convergence analysis. Compared with the conventional one for systems in a classic state space form, the generalized policy iteration algorithm for nonlinear descriptor systems differs in the presence of an extra partial differential equation system from which the value function is solved. Necessary and sufficient conditions guaranteeing solvability of the value function are established. Sufficient solvability conditions for a special case, where the value function is a linear combination of a set of basis functions, are also derived.

2016 American Control Conference (ACC)

Policy iteration-based optimal control design for nonlinear descriptor systems

Yebin Wang

***Abstract*— This paper considers state feedback optimal control design for a class of nonlinear descriptor systems. Prior work either stops at the Hamilton-Jacobian-Bellman equations and thus is non-constructive, or converts the optimal control problem into a large scale nonlinear optimization problem and thus is open-loop control design. This paper proposes a generalized policy iteration algorithm to compute the state feedback optimal control policy in a constructive manner, and presents the convergence analysis. Compared with the conventional one for systems in a classic state space form, the generalized policy iteration algorithm for nonlinear descriptor systems differs in the presence of an extra partial differential equation system from which the value function is solved. Necessary and sufficient conditions guaranteeing solvability of the value function are established. Sufficient solvability conditions for a special case, where the value function is a linear combination of a set of basis functions, are also derived.**

I. Introduction

This paper investigates the state feedback optimal control design for a class of nonlinear descriptor systems

$$E\dot{x} = f(x,u), \quad Ex(0) = Ex_0, \tag{1}$$

where $E \in \mathbb{R}^{n\times n}$ is a constant matrix with a rank $r = \text{rank}(E) \le n$, $x \in \Omega_x \subset \mathbb{R}^n$ the system state vector, Ω_x a compact set containing the origin in its interior, $u \in \mathbb{R}^m$ the control input, and $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is a vector field. All components of f are locally Lipschitz in x. When E is an identity matrix, the system (1) is in classic state space form. The optimal control objective is represented by

$$J(u) = S(Ex(T)) + \int_0^T L(x,u)\mathrm{d}t, \tag{2}$$

where S, L are continuously differentiable in all arguments.

This problem has been studied by researchers from different perspectives, for instance dynamic programming [1]–[3], the minimum principle [4], etc. In work [1], the state feedback optimal control design arrives at solving the following Hamilton-Jacobian-Bellman (HJB) equations

$$\frac{\partial V^*(x,t)}{\partial x} = W^*E,\ V^*(Ex(T),T) = S(Ex(T)) \tag{3a}$$

$$\frac{\partial V^*(x,t)}{\partial t} = \min_u\{L(x,u) + W^* f(x,u)\}. \tag{3b}$$

In work [2], $\mathcal{H}_\infty$ control for nonlinear descriptor systems (1) is studied and sets of equations similar to (3) are derived. Work [3] deals with optimal control design for a special class of nonlinear descriptor systems with $E = \text{diag}\{I_r, 0\}$. Focusing on a special case of systems (1), work [4] computes the open-loop optimal control trajectory through solving a discretized numerical optimization problem. These work suffer, more or less, limitations and thus motivate researches beyond the existing frontier. For instance, the aforementioned HJB equations are not much instructive to allow straightforward computation of the optimal control policy [3]. Turning to numerical optimization generally ends up with solving a large scale nonlinear (very often non-convex) optimization problem, which could be expensive in computation or even fail to work out a valid solution [5].

This paper contributes to alleviate the non-constructive restriction in the existing work by resolving two challenges in solving the HJB (3): coming up with a constructive algorithm to solve W in (3), and deriving solvability conditions under which the value function V can be computed from (3a) given W. Major contributions of this paper are

- based on the HJB (3), propose a generalized policy iteration algorithm such that the state feedback optimal control policy of the system (1) can be computed iteratively and constructively;
- establish necessary and sufficient conditions on W under which the solution of the value function is guaranteed to exist;
- when W is parameterized as a linear combination of a set of basis functions [6], [7], derive sufficient conditions on the basis functions such that the generalized policy iteration algorithm succeeds in producing the value function.

It is worth mentioning that viewed as a powerful tool to construct state feedback optimal control policy, the conventional policy iteration algorithm has been widely used for systems in a classic state space form [6]–[10]. To the best of our knowledge, its extension to nonlinear descriptor systems has not been reported. In fact, the extension turns out to be non-trivial due to the presence of E. The solvability conditions established in this paper are the key to ensure the soundness of the generalized policy iteration algorithm. Also, the solvability conditions can be readily utilized to verify whether the set of basis functions are appropriate.

The remainder of this paper is structured as follows. Section II introduces fundamentals of nonlinear descriptor systems, and formulates the optimal control problem. The generalized policy iteration algorithm for nonlinear descriptor systems is presented in Section III. Solvability conditions for the value function is discussed in Section IV. Finally, Section V offers some future research directions and concludes this paper.

Y. Wang is with Mitsubishi Electric Research Laboratories, Cambridge, MA 02139, USA (email: yebinwang@ieee.org).

II. Preliminaries and Problem Formulation

We state assumptions on nonlinear descriptor systems, the definition in regard to admissible control policy, and problem statement. Readers are referred to [1], [2], [11] and references therein for details on nonlinear descriptor systems.

Assumption 2.1: For any initial condition Ex_0 and any control u, system (1) has a unique solution.

Assumption 2.2: $L(x, u) \geq 0$, for any x, u, and there exists a factorization $L(x, u) = h^\top(x, u)h(x, u)$ where $h(x, u)$ is continuous differentiable and to be interpreted as an output. Further, $S(Ex(T)) \geq 0$, for any $x(T)$.

Assumption 2.3: The function $L(x, u)$ and the integral are defined such that $\int_0^T L(x, u)\mathrm{d}t = \infty$ whenever the output $h(x, u)$ has an impulsive component.

Definition 2.4 (Admissible feedback control): A feedback control policy $u(x) \in U \subset \mathbb{C}^1[0, T]$ is admissible if, for any initial condition Ex_0, the resultant closed-loop system has no impulsive solution. Correspondingly, U is called the admissible control set.

In Definition 2.4, the admissible control is assumed to be state feedback. This technical assumption makes the resultant optimal control problem exposed to well-established theories, e.g. dynamic programming. Defining U as the set of all the admissible feedback control policies, we assume that there exists an initial control policy $u_0(x)$ such that $u_0 \in U$.

Taking account of Definition 2.4 and Assumption 2.3, we know that given an admissible control policy $u(x)$, the output $h(x, u)$ does not have impulsive components, and the resultant cost function (2) has a finite value. Conversely, when the cost function (2) associated with the closed-loop system goes to infinity, then $h(x, u)$ has impulsive components, and the closed-loop system has an impulsive solution, which means the corresponding control policy is not admissible.

Assumption 2.5: There exists an non-empty admissible control set for the system (1).

Without loss of generality, this paper deals with the cost function (2) with $T = \infty$. For such a case, the admissible control should yield a finite value of the cost function, and a stable closed-loop system. The optimal control problem for nonlinear descriptor system (1) can be formulated as follows.

Problem 2.6 (Optimal control problem): Given the system (1), find $u^* \in U$ which minimizes the cost function (2), i.e. $u^* = \arg\min_{u \in U} J(u)$.

Problem 2.6 is difficult to solve for at least two reasons. First, nonlinearities involved in the problem make it almost impossible to find an analytic solution. Second, the corresponding numerical optimization problem is generally non-convex.

III. Main Results

A. A Generalized Policy Iteration Algorithm

We generalize the conventional policy iteration algorithm for systems to the nonlinear descriptor system case. Assume that an admissible control policy $u_0(x)$ is known. The generalized policy iteration algorithm can be summarized in the following two steps, with $i = 0, 1, \cdots$.

1) Policy evaluation

Solve for the positive definite function $V_i(x)$ and $W_i(x)$ satisfying

$$W_i(x)f(x, u_i) + L(x, u_i) = 0, \quad \forall x \in \Omega_x \tag{4a}$$
$$W_i(x)E = \nabla V_i, \tag{4b}$$

where $\nabla V_i = \partial V_i / \partial x$ is a row vector.

2) Policy improvement

Update the control policy according to

$$u_{i+1}(x) = \arg\min_{u \in U} \{L(x, u) + W_i(x)f(x, u)\}, \forall x \in \Omega_x. \tag{5}$$

For the case $E = I$, the generalized algorithm is reduced to the conventional policy iteration [9], [12]–[14], where (4a) is used to solve a Lyapunov function $\mathcal{W}_i$ with $\nabla \mathcal{W}_i = W_i$. As a system of first order linear partial differential equations, the closed-form solution of (4a) is difficult to establish. Instead, a good approximate solution is usually of practical interest. Given parameterizations of u_i and W_i, (4a) is reduced to algebraic equations, and thus the approximate solution can be computed. The two steps (4)-(5) can be repeated until the convergence is attained.

Remark 3.1: With $\nabla \mathcal{W}_i = W_i$, the function $\mathcal{W}_i$ is not necessarily positive definite. However, while solving W_i, we should make sure that (4b) admits a postiche definite solution V_i. This is because V_i is required to show the stability of the closed-loop system, and the convergence of the generalized algorithm. Section IV identifies conditions under which a solution of (4b) is ensured.

For a class of control-affine nonlinear descriptor systems

$$E\dot{x} = f(x) + g(x)u, \quad Ex(0) = Ex_0, \tag{6}$$

where $g : \mathbb{R}^n \to \mathbb{R}^{n \times m}$ consists of m smooth vector fields, and a commonly used cost functional given by

$$J(u) = \int_0^\infty \left[Q(x) + u^\top Ru\right] dt, \quad x(0) = x_0 \in \Omega_x \tag{7}$$

where $Q(x)$ is a positive definite function on Ω_x, and $R = R^\top$ is a positive definite matrix, the generalized policy iteration algorithm can be slightly modified by replacing the step (5) with $u_{i+1}(x) = -\frac{1}{2}R^{-1}g^\top W_i^\top$.

The generalized policy iteration algorithm is meaningful only if the resultant control u_{i+1} has the following properties

1) the new control u_{i+1} gives a stable closed-loop system;
2) the new control u_{i+1} results in improvement of the closed-loop system performance measured by the cost function.

B. Convergence Analysis

In the convergence analysis, we assume that given W_i, the value function V_i can always be solved. We have the following result about the stability of the closed-loop system as an outcome of the generalized algorithm.

Proposition 3.2: Given an admissible control strategy u_i for the system (1), the improved control policy u_{i+1} yields a stable closed-loop system.

Proof: We first verify that given a stabilizing control policy u_i, the V_i solved in the policy evaluation step 1) is positive definite. From (4), we have $\dot{V}_i = -L(x, u_i)$. Considering that u_i is a stabilizing control policy, we have $x(\infty) = 0$ and $V_i(x(\infty)) = S(Ex(\infty))$. With $L(x, u) \geq 0$, we obtain $V_i(x(t)) = S(Ex(\infty)) + \int_0^\infty L(x, u_i)\mathrm{d}t > 0$.

Letting V_i be a Lyapunov function candidate for the system (1) with the control policy $u = u_{i+1}$, we compute its time derivative

$$\begin{aligned}
\dot{V}_i &= W_i f(x, u_{i+1}) \\
&= W_i(f(x, u_{i+1}) + f(x, u_i) - f(x, u_i) \\
&= -L(x, u_i) + W_i(f(x, u_{i+1}) - f(x, u_i)) \\
&= -L(x, u_i) + L(x, u_{i+1}) + W_i f(x, u_{i+1}) \\
&\quad - L(x, u_{i+1}) - W_i f(x, u_i)
\end{aligned} \tag{8}$$

Since u_{i+1} is obtained by solving (5), we have $L(x, u_{i+1}) + W_i f(x, u_{i+1}) \leq L(x, u_i) + W_i f(x, u_i) = 0$, and

$$\begin{aligned}
\dot{V}_i &\leq -L(x, u_i) - L(x, u_{i+1}) - W_i f(x, u_i) \\
&= -L(x, u_{i+1}) - \underbrace{(L(x, u_i) + W_i f(x, u_i))}_{\equiv 0} \\
&\leq -L(x, u_{i+1}).
\end{aligned} \tag{9}$$

Since $V_i > 0, \forall x \neq 0$, and $\dot{V}_i$ is negative definite $\forall x \neq 0$, we conclude that V_i is a Lyapunov function for the closed-loop system (1) with u_{i+1}. The closed-loop system is asymptotically stable. ∎

We can also establish that throughout iterations of the generalized algorithm, the values of the cost function monotonically decrease.

Proposition 3.3: The cost of the closed-loop system with the improved control policy u_{i+1} is no greater than that with the control policy u_i.

Proof: Denote W_i and $\bar{W}$ solutions of (4a) corresponding to control policies u_i and u_{i+1}, respectively, which means

$$W_i f(x, u_i) + L(x, u_i) = 0 \tag{10a}$$
$$\bar{W}_i f(x, u_{i+1}) + L(x, u_{i+1}) = 0. \tag{10b}$$

Also denote V_i and $\bar{V}_i$ satisfying $\nabla V_i = W_i E$ and $\nabla \bar{V}_i = \bar{W}_i E$, respectively. Equations (10) are equivalent to the following two Lyapunov equations

$$\nabla V_i \frac{dx}{dt} + L(x, u_i) = 0$$
$$\nabla \bar{V}_i \frac{dx}{dt} + L(x, u_{i+1}) = 0.$$

Note that V_i and $\bar{V}_i$ are the cost functions corresponding to the control policy u_i and u_{i+1}, respectively. Subtracting (10b) from (10a) gives

$$W_i f(x, u_i) - \bar{W}_i f(x, u_{i+1}) + L(x, u_i) - L(x, u_{i+1}) = 0,$$

where the left-hand side (LHS) is rearranged in such a way

$$\begin{aligned}
&W_i f(x, u_i) - \bar{W}_i f(x, u_{i+1}) + L(x, u_i) - L(x, u_{i+1}) \\
&= (W_i - \bar{W}_i) f(x, u_{i+1}) + W_i(f(x, u_i) - f(x, u_{i+1})) \\
&\quad + L(x, u_i) - L(x, u_{i+1}) \\
&= (W_i - \bar{W}_i) f(x, u_{i+1}) + \underbrace{W_i f(x, u_i) + L(x, u_i)}_{\equiv 0} \\
&\quad - \underbrace{(W_i f(x, u_i) + L(x, u_i))}_{\leq W_i f(x, u_{i+1}) + L(x, u_{i+1}) = 0} = 0 \\
&\Leftrightarrow (W_i - \bar{W}_i) f(x, u_{i+1}) = W_i f(x, u_{i+1}) + L(x, u_{i+1}).
\end{aligned}$$

Integrating the above equation over $[0, T]$, and considering $V_i(x(T)) = \bar{V}_i(x(T))$, we have

$$\begin{aligned}
&V_i(x(T)) - V_i(x(0)) - (\bar{V}_i(x(T)) - \bar{V}_i(x(0))) \\
&- \int_0^T (W_i f(x, u_{i+1}) + L(x, u_{i+1}))\mathrm{d}t = 0 \\
&\Leftrightarrow \bar{V}_i(x(0)) - V_i(x(0)) \\
&= \int_0^T (W_i f(x, u_{i+1}) + L(x, u_{i+1}))\mathrm{d}t \leq 0
\end{aligned}$$

We therefore show that $\bar{V}_i(x(0)) \leq V_i(x(0))$, which suggests that the generalized algorithm results in monotonic reduction of the cost function. ∎

Combining Propositions 3.2-3.3, we have the following conclusion about the convergence of the generalized algorithm.

Theorem 3.4: Consider system (1) and the cost function (2). Suppose that $u_0 \in U$, and a positive definite solution V_i of (4) exists, for $i = 0, 1, \cdots$. Then,

1) $u_{i+1} \in U$ for $i \geq 0$;
2) $J(u_{i+1}) \leq J(u_i)$ for $i \geq 0$;
3) $\lim_{i \to \infty} J(u_i) = J^*$ with $0 \leq J^* < \infty$.

Proof: We simply describe the rough idea. Fact 1) can be readily shown by employing induction and Proposition 3.2. Similarly fact 2) is a natural consequence of utilizing induction and Proposition 3.3. Fact 2) indicates that the sequence $\{J(u_i)\}$ is monotonically decreasing. It is also evident that the sequence has a lower bound 0. Since a monotonic decreasing sequence with a lower bound always converges, fact 3) is established. ∎

Note that we however do not establish the optimality of J^* and the corresponding control policy.

C. Parameterizations in the Algorithm

The generalized policy iteration algorithm requires to solve the partial differential equation (4). Next we provide a practical implementation method by parameterizing the control policy and a pseudo value function $\mathcal{W}_i$.

To begin with, let $\{\phi_j(x)\}_{j=1}^N$ with $\phi_j : \mathbb{R}^n \to \mathbb{R}$ and $\{\psi_j(x)\}_{j=1}^q$ with $\psi_j : \mathbb{R}^n \to \mathbb{R}^m$ be two sets of linearly independent, continuously differentiable functions and vector fields, respectively. In addition, we assume that $\phi_j(0) = 0$, $\forall\ 1 \leq j \leq N$ and $\psi_j(0) = 0$, $\forall\ 1 \leq j \leq q$.

Assumption 3.5: There exists a smooth pseudo value function $\mathcal{W}_i(x)$ such that $\nabla \mathcal{W}_i(x) = W_i(x)$.

Assumption 3.6: Provided that $u_i \in U$, and $u_i(x) \in \mathrm{span}\{\psi_1(x), \cdots, \psi_q(x)\}$, then,

$$\mathcal{W}_i(x) \in \mathrm{span}\{\phi_1(x), \cdots, \phi_N(x)\},$$
$$u_{i+1}(x) \in \mathrm{span}\{\psi_1(x), \cdots, \psi_q(x)\},$$

where $\mathcal{W}_i(x)$ and $u_{i+1}(x)$ are obtained from (4a) and (5).

Assumptions 3.5-3.6 are involved in the standard policy iteration as well. Under Assumption 3.6, we can find three sets of weights $\{w_{i,1}, w_{i,2}, \cdots, w_{i,N}\}$, $\{c_{i,1}, c_{i,2}, \cdots, c_{i,q}\}$, and $\{c_{i+1,1}, c_{i+1,2}, \cdots, c_{i+1,q}\}$, such that

$$u_i(x) = \sum_{j=1}^{q} c_{i,j}\psi_j(x)$$
$$\mathcal{W}_i(x) = \sum_{j=1}^{N} w_{i,j}\phi_j(x)$$
$$u_{i+1}(x) = \sum_{j=1}^{q} c_{i+1,j}\psi_j(x).$$

Remark 3.7: When Assumption 3.6 is not satisfied, these weights can still be numerically obtained based on neural network approximation methods, such as the off-line approximation using Galerkin's method [15]. In addition, for uncertain nonlinear systems, these weights can be trained using approximate-dynamic-programming-based online learning methods [16], [17]. When approximation methods are used, Ω_x is required to be a compact set to guarantee the boundedness of the approximation error.

IV. Solvability Conditions

Main characteristics of the generalized policy iteration algorithm is the extra system of partial differential equations given by (4b). What it tells us is that: the parameterizations of $\mathcal{W}_i$ and u_i are critical to secure the convergence of the generalized algorithm. Otherwise, the resultant solution $(\mathcal{W}^*, u^*)$ may not make sense because (4b) with $W^* = \nabla\mathcal{W}^*$ does not admit a solution of the value function V. Hence, it is important to pick appropriate parameterizations of $\mathcal{W}_i$ so that it is compatible with (4b). In this section, we resort to exterior differential systems to obtain explicit solvability conditions of (4b).

A. Elements of Differential Geometry

To enhance the self-completeness of this paper, we give a brief overview on elements of differential geometry entailed by the derivation of solvability conditions. Some definitions appearing in this section such as algebraic ideal, are omitted because their absence will not compromise the readability and comprehension of this section. Interested readers are referred to [18]–[21] for details on differential geometry.

Given a n-dimensional manifold M, the tangent space of M at p and is denoted by T_pM. The dual space of T_pM at each $p \in M$ is called the cotangent space to the manifold M at p, and is denoted by T_p^*M.

Definition 4.1 (Distributions and codistribitions): Let M be a C^∞ manifold. A distribution $\mathcal{D}$ (resp. a codistribution Λ) is an assignment, to each point $p \in M$, of a subspace $\mathcal{D}_p$ of T_pM (resp. a subspace Λ_p of T_p^*M).

More explicitly, according to [21], an r-dimensional distribution $\mathcal{D}$ on the manifold M is a map which assigns to each $p \in M$ an r-dimensional subspace of $\mathbb{R}^n$ such that for each $p \in M$ there exists a neighborhood $\mathcal{U}$ of p and r smooth vector fields $f_1, \cdots, f_r$ with the properties

1) $f_1(p), \cdots, f_r(p)$ are linearly independent, $\forall p \in \mathcal{U}$;
2) $\mathcal{D}(p) = \mathrm{span}\{f_1(p), \cdots, f_r(p)\}$, $\forall p \in \mathcal{U}$.

Let S be a real vector space. A multilinear function $T : S^k \to \mathbb{R}$ is called a k-tensor, and the set of all k-tensors on S is denoted $\mathcal{L}^k(S)$. The set of all alternating k-tensors on S is denoted as $\Lambda^k(S)$. A k-tensor field on M is a section of $\mathcal{L}^k(M)$, i.e., a function ω assigning to every $p \in M$ a k-tensor $\omega(p) \in \mathcal{L}^k(T_pM)$.

Definition 4.2 (k-form): If a k-tensor field ω is a section of $\Lambda^k(M)$, then ω is called a differential form of order k or a k-form on M.

The 0-form on a manifold M is a function $f : M \to \mathbb{R}$. The differential $\mathrm{d}f$ of a 0-form f is defined pointwisely as the 1-form, for a vector field X_p,

$$\mathrm{d}f(p)(X_p) = X_p(f).$$

Given $f \in \Lambda^k(S)$ and $g \in \Lambda^l(S)$, the wedge product is denoted $f \wedge g \in \Lambda^{k+l}(S)$.

Definition 4.3 (Exterior derivative): Let ω be a k-form on a manifold M whose representation in a chart $(\mathcal{U}, x)$ is given by $\omega = \sum_I \omega_I \mathrm{d}x^I$ for ascending multi-indices I. The exterior derivative or differential operator, d, is a linear map taking the k-form ω to the $(k+1)$-form $\mathrm{d}\omega$ by

$$\mathrm{d}\omega = \sum_I \mathrm{d}\omega_I \wedge \mathrm{d}x^I.$$

Note that the ω_I are smooth functions (0-forms) whose differential $\mathrm{d}\omega_I$ has already been defined as

$$\mathrm{d}\omega_I = \sum_{j=1}^{n} \frac{\partial \omega_I}{\partial x^j}\mathrm{d}x^j.$$

Therefore, for any k-form ω,

$$\mathrm{d}\omega = \sum_I \sum_{j=1}^{n} \frac{\partial \omega_I}{\partial x^j}\mathrm{d}x^j \wedge \mathrm{d}x^I.$$

Definition 4.4 (Frobenius condition): A set of linearly independent 1-forms $\alpha^1, \ldots, \alpha^s$ in the neighborhood of a point is said to satisfy the Frobenius condition if one of the following equivalent conditions holds

1) $\mathrm{d}\alpha^i$ is a linear combination of $\alpha^1, \ldots, \alpha^s$.
2) $\mathrm{d}\alpha^i \wedge \cdots \wedge \alpha^s = 0$ for $1 \le i \le s$.

Theorem 4.5 (Frobenius Theorem for codistribution): Let $\mathcal{I}$ be an algebraic ideal generated by the independent 1-forms $\alpha^1, \ldots, \alpha^s$ which satisfies the Frobenius condition. Then in a neighborhood of x, there exist functions h^i with $1 \le i \le s$ such that

$$\mathcal{I} = \{\alpha^1, \ldots, \alpha^s\} = \{\mathrm{d}h^1, \ldots, \mathrm{d}h^s\}.$$

Remark 4.6: Frobenius Theorem for codistributions states that the codistribution is integrable if the exterior derivative of every one-form taking values in the codistribution lies in the algebraic ideal generated by the codistribution. This is not easy to verify for k-forms in high dimensional manifolds. As shown later, Theorem 4.5 is useful to substantiate the sufficiency of solvability conditions.

Example 1: [19] We verify Frobenius conditions for the unicycle system

$$\begin{aligned}\dot{x} &= \cos(\theta)u_1\\ \dot{y} &= \sin(\theta)u_1\\ \dot{\theta} &= u_2.\end{aligned} \tag{11}$$

With a distribution consisting of $g^1 = [\cos(\theta), \sin(\theta), 0]^\top, g^2 = [0, 0, 1]^\top$, we have the codistribution $\mathcal{I} = \{\omega\}$, where

$$\omega = \sin(\theta)\mathrm{d}x - \cos(\theta)\mathrm{d}y + 0\mathrm{d}\theta.$$

The exterior derivative of ω is

$$\mathrm{d}\omega = \cos(\theta)\mathrm{d}\theta \wedge \mathrm{d}x + \sin(\theta)\mathrm{d}\theta \wedge \mathrm{d}y$$

and therefore

$$\begin{aligned}\mathrm{d}\omega \wedge \omega &= -\cos^2(\theta)\mathrm{d}\theta \wedge \mathrm{d}x \wedge \mathrm{d}y + \sin^2(\theta)\mathrm{d}\theta \wedge \mathrm{d}y \wedge \mathrm{d}x\\ &= -\mathrm{d}x \wedge \mathrm{d}y \wedge \mathrm{d}\theta \neq 0.\end{aligned}$$

The second condition in Definition 4.4 does not hold, and thus ω is not integrable.

Next we recite another theorem exploited to establish the necessity of solvability conditions.

Theorem 4.7: Let M be a manifold and $p \in M$. Then the exterior derivative is the unique linear operator

$$\mathrm{d} : \Omega^k(M) \to \Omega^{k+1}(M)$$

for $k \geq 0$, that satisfies for every form ω, $\mathrm{d}(\mathrm{d}\omega) = 0$.

B. Solvability Conditions

In this section, Theorems 4.5 and 4.7 are applied to derive solvability conditions of (4b). From Assumption 3.5, we know $\mathcal{W}_i$ is a 0-form, and its 1-form W_i is denoted by

$$W_i = \sum_{k=1}^{n} \frac{\partial \mathcal{W}_i}{\partial x_k}\mathrm{d}x_k.$$

We have the following solvability conditions.

Proposition 4.8: Given Assumption 3.5, (4b) is solvable if and only if the following conditions hold

$$\frac{\partial(W_iE_k)}{\partial x_j} = \frac{\partial(W_iE_j)}{\partial x_k},\ 1 \leq j \leq n; j+1 \leq k \leq n. \tag{12}$$

Proof: Sufficiency: Denoting $E = [E_1, \cdots, E_n]$ with $E_k \in \mathbb{R}^n$, we rewrite the LHS of (4b) as follows

$$\begin{aligned}W_iE =& W_i[E_1, \cdots, E_n]\\ =& [W_iE_1, \cdots, W_iE_n] = \widehat{W}_i \in \mathbb{R}^n.\end{aligned}$$

The row vector $\widehat{W}_i$ also takes the expression $\widehat{W}_i = \sum_{k=1}^{n} W_iE_k\mathrm{d}x_k$. With Definition 4.3, we compute the exterior derivative $\mathrm{d}\widehat{W}_i$

$$\mathrm{d}\widehat{W}_i = \sum_{j=1}^{n}\sum_{k=1}^{n} \frac{\partial(W_iE_k)}{\partial x_j}\mathrm{d}x_j \wedge \mathrm{d}x_k.$$

Considering the properties of the wedge product [18]

$$\mathrm{d}x_k \wedge \mathrm{d}x_j = \begin{cases}0, & k = j;\\ \neq 0, & k \neq j,\end{cases}$$

and $\mathrm{d}x_k \wedge \mathrm{d}x_j = -\mathrm{d}x_j \wedge \mathrm{d}x_k$, we have

$$\mathrm{d}\widehat{W}_i = \sum_{j=1}^{n}\sum_{k=j+1}^{n}\left[\frac{\partial(W_iE_k)}{\partial x_j} - \frac{\partial(W_iE_j)}{\partial x_k}\right]\mathrm{d}x_j \wedge \mathrm{d}x_k. \tag{13}$$

Combining conditions (12) with the expression of $\mathrm{d}\widehat{W}_i$ given by (13), we have $\mathrm{d}\widehat{W}_i = 0$, or equivalently the 1-form $\widehat{W}_i$ satisfies the first condition in Definition 4.4. According to Theorem 4.5, there exists a 0-form V_i such that $\mathrm{d}V_i = \widehat{W}_i$, which infers the existence of solutions to (4b). The proof of sufficiency is therefore concluded.

Necessity: Assume that (4b) has a solution denoted by V_i. From (4b), the 1-form of V_i is written as $\widehat{W}_i$. We further compute 1-form of $\widehat{W}_i$, and have (13). According to Theorem 4.7, $\mathrm{d}(\mathrm{d}V_i) = \mathrm{d}(\mathrm{d}\widehat{W}_i) = 0$, which implies conditions (12). Thus necessity is established. ■

We next look into a special case where the solution of (4a) satisfies Assumptions 3.5-3.6. That is: the function $\mathcal{W}_i$ is linearly parameterized as follows

$$\mathcal{W}_i(x) = \sum_{l=1}^{N} w_l\phi_l(x) = w^\top\Phi(x),$$

where $w = [w_1, \cdots, w_N]^\top$ is a vector of constants and $\Phi(x) = [\phi_1, \ldots, \phi_N]^\top$. This special case is particularly interesting in practice, because the specific parameterizations have been pervasively adopted in literature. We have

$$W_i(x) = \frac{\partial \mathcal{W}_i}{\partial x} = w^\top\frac{\partial \Phi}{\partial x},$$

and w can be determined on the basis of (4a). We derive solvability conditions on basis functions $\{\phi_l, 1 \leq l \leq N\}$.

With $\nabla V_i^\top = W_i(x)E$, we have

$$\nabla V_i^\top = w^\top\frac{\partial \phi}{\partial x}E = w^\top\underbrace{\begin{bmatrix}\frac{\partial \phi_1}{\partial x}E_1 & \cdots & \frac{\partial \phi_1}{\partial x}E_n\\ \vdots & \ddots & \vdots\\ \frac{\partial \phi_N}{\partial x}E_1 & \cdots & \frac{\partial \phi_N}{\partial x}E_n\end{bmatrix}}_{\Gamma(x)}.$$

According to Frobenius Theorem 4.5, (4b) is solvable when the co-distribution $\Gamma(x)$ is integrable. A sufficient condition for the integrability of $\Gamma(x)$ is that each row vector of $\Gamma(x)$ is a 1-form of a smooth function ω_k for $1 \leq k \leq N$, i.e.,

$$\mathrm{d}\omega_k = \begin{bmatrix}\frac{\partial \phi_k}{\partial x}E_1 & \cdots & \frac{\partial \phi_k}{\partial x}E_n\end{bmatrix}. \tag{14}$$

Then we have the solution of the value function V_i given by

$$V_i = w^\top\Omega, \quad \Omega = [\omega_1, \ldots, \omega_N]^\top. \tag{15}$$

We have Proposition 4.8 for the special case.

Proposition 4.9: Given Assumptions 3.5-3.6, (4b) has a solution in the form of (15) if for every ϕ_l, $1 \leq l \leq N$, the following conditions hold

$$\frac{\partial(\frac{\partial \phi_l}{\partial x}E_k)}{\partial x_j} = \frac{\partial(\frac{\partial \phi_l}{\partial x}E_j)}{\partial x_k},\ 1 \leq j \leq n; j+1 \leq k \leq n. \tag{16}$$

Remark 4.10: For nonlinear descriptor systems with a constant E, conditions (12) boil down to

$$\frac{\partial W_i}{\partial x_j}E_k = \frac{\partial W_i}{\partial x_k}E_j,\ 1 \leq j \leq n; j+1 \leq k \leq n. \qquad (17)$$

Similarly, with E constant, conditions (16) are simplified into

$$\frac{\partial^2 \phi_l}{\partial x \partial x_j}E_k = \frac{\partial^2 \phi_l}{\partial x \partial x_k}E_j,\ 1 \leq j \leq n; j+1 \leq k \leq n. \quad (18)$$

Both Propositions 4.8-4.9 imposes conditions on parameterizations of $\mathcal{W}_i$. This is useful for incorporating the verification of solvability conditions (12) or (16) into the generalized policy iteration algorithm. Algorithm 1 illustrates detailed steps for the generalized policy iteration algorithm with solvability check steps, where K and P are sufficiently large positive constants.

Algorithm 1: Generalized policy iteration algorithm with solvability verification steps

```
Initialize k = 0, K, P, flag=0;
while (k ≤ K) and (flag=0) do
    Choose a set of basis functions {φ_1, ···, φ_N};
    Initialize l = 1, flag=1;
    for l ≤ N do
        flag = flag ⋂ verify conditions (16) for φ_l;
        if flag=0 then
            break;
        l = l + 1;
    k = k + 1;
Solve an initial stabilizing control policy u_0;
Initialize i = 0;
for i ≤ P do
    Execute the policy evaluation to solve W_i and V_i;
    Execute the policy improvement to update u_{i+1};
    i = i + 1;
return (u_P, V_P);
```

As shown in Algorithm 1, after the verification of basis functions against conditions (16), we only need to solve (4a) for W_i and update u_{i+1} according to (5). The value function V_i is merely needed for analysis, and its existence is crucial to the convergence of the generalized algorithm. In case that V_i is required, symbolic computation softwares such as Maple$^{\text{TM}}$ and Mathematica$^{\text{TM}}$ can be readily utilized to compute the analytical solution V_i of the integrable first order PDE system (4b).

V. Conclusion and Future Work

This paper generalized the policy iteration algorithm to a class of nonlinear descriptor systems so that constructive optimal control design can be performed. We exposed main differences between the generalized policy iteration algorithm and the standard one: the value function has to satisfy an additional system of first order partial differential equations (PDEs). We established necessary and sufficient conditions for solvability of the extra PDE system. For a special case where the value function takes linear parameterizations of a set of basis functions, we attained sufficient conditions on the basis functions to enable the generalized policy iteration algorithm. Future work includes exploration of conditions on the extra PDE system to ensure the uniqueness of the solution. This paper presented results which are merely good for analysis, i.e., given a set of basis functions, one can verify whether the basis functions satisfy the derived solvability conditions. The synthesis of basis functions, which answers how to choose basis functions, remains open. Also, this paper does not pay attention to the optimality of the solution that the generalized policy iteration algorithm converges to. Performance and optimality analysis could be another interesting topic to study in the future.

References

[1] H. Xu and K. Mizukami, "Hamilton-Jacobi equation for descriptor systems," *Syst. Control Lett.*, vol. 21, pp. 321–327, 1993.

[2] H.-S. Wang, C.-F. Yung, and F.-R. Chang, "$\mathcal{H}_\infty$ control for nonlinear descriptor systems," *IEEE Trans. Automat. Control*, vol. 47, no. 11, pp. 1919–1924, Nov. 2002.

[3] J. Sjöberg, "Optimal control and model reduction of nonlinear dae models," Ph.D. dissertation, Linköping University, 2008.

[4] M. Gerdts, "Optimal control of ordinary differential equations and differential-algebraic equations," Ph.D. dissertation, University of Bayreuth, 2006.

[5] Y. Wang, K. Ueda, and S. A. Bortoff, "A Hamiltonian approach to compute an energy efficient trajectory for a servomotor system," *Automatica*, vol. 49, no. 12, pp. 3550–3561, Dec. 2013.

[6] Z. Chen and S. Jagannathan, "Generalized Hamilton-Jacobi-Bellman formulation-based neural network control of affine nonlinear discrete-time systems," *IEEE Transactions on Neural Networks*, vol. 19, no. 1, pp. 90–106, 2008.

[7] T. Cheng, F. L. Lewis, and M. Abu-Khalaf, "Fixed-final-time-constrained optimal control of nonlinear systems using neural network HJB approach," *IEEE Transactions on Neural Networks*, vol. 18, no. 6, pp. 1725–1736, 2007.

[8] D. Kleinman, "On an iterative technique for Riccati equation computations," *IEEE Transactions on Automatic Control*, vol. 13, no. 1, pp. 114–115, 1968.

[9] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. Athena Scientific Belmont, 1995.

[10] X. Xu, D. Hu, and X. Lu, "Kernel-based least squares policy iteration for reinforcement learning," *IEEE Transactions on Neural Networks*, vol. 18, no. 4, pp. 973–992, 2007.

[11] E. Jonckheere, "Variational calculus for descriptor problems," *IEEE Trans. Automat. Control*, vol. 33, no. 5, pp. 491–495, May. 1988.

[12] R. A. Howard, *Dynamic Programming and Markov Processes*. Cambridge, MA: MIT Press, 1960.

[13] G. N. Saridis and C.-S. G. Lee, "An approximation theory of optimal control for trainable manipulators," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 3, pp. 152–159, 1979.

[14] R. J. Leake and R.-W. Liu, "Construction of suboptimal control sequences," *SIAM Journal on Control*, vol. 5, no. 1, pp. 54–63, 1967.

[15] R. W. Beard, G. N. Saridis, and J. T. Wen, "Galerkin approximations of the generalized Hamilton-Jacobi-Bellman equation," *Automatica*, vol. 33, no. 12, pp. 2159–2177, 1997.

[16] Z.-P. Jiang and Y. Jiang, "Robust adaptive dynamic programming for linear and nonlinear systems: An overview," *European Journal of Control*, vol. 19, no. 5, pp. 417–425, 2013.

[17] D. Vrabie and F. L. Lewis, "Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems," *Neural Networks*, vol. 22, no. 3, pp. 237–246, 2009.

[18] M. Spivak, *A Comprehensive Introduction to Differential Geometry*, 3rd ed. Publish or Perish, Inc., 1999.

[19] G. J. Pappas, J. Lygeros, D. Tilbury, and S. S. Sastry, *Essays on Mathematical Robotics*. Springer, 1998, ch. Exterior differential systems in control and robotics, pp. 271–372.

[20] F. Bullo and A. D. Lewis, *Geometric Control of Mechanical Systems*. Springer, 2004.

[21] R. Marino and P. Tomei, *Nonlinear Control Design: Geometric, Adaptive, and Robust*. Hertfordshire, UK: Prentice-Hall, 1995.