Available online at www.sciencedirect.com

ScienceDirect

IFAC Papers Online
CONFERENCE PAPER ARCHIVE

IFAC-PapersOnLine 48-20 (2015) 517–522

Policy Iteration Algorithm for the Control of Oxygenation

Anake Pomprapa *, **Milod Mir Wais** *, **Marian Walter** *, **Berno J.E. Misgeld** *, **Steffen Leonhardt** *

* *Philips Chair for Medical Information Technology, RWTH Aachen University, Aachen, Germany (e-mail: pomprapa@hia.rwth-aachen.de).*

Abstract: The policy iteration algorithm (PIA) is a quasi non-identifier approach of nonlinear optimal control based on a reinforcement learning and iterative algorithm in order to solve the Hamilton-Jacobi-Bellman (HJB) equation. The synthesized state-feedback controller corresponding to the converged solution should be applicable for the control of cardiopulmonary system. In this article, the simulation results for the control of oxygenation were carried out using a simplified first-order model with time delay based on porcine dynamics. The distinctive results of oxygenation control can then be achieved based on the proposed control strategy. In addition, the practical example of water level for interacting three-tank system, which has the nonlinear dynamics similar to that of the oxygenation, was implemented in order to prove the concept of this control scheme.

Keywords: policy iteration algorithm, optimal control, reinforcement learning, control of oxygenation, biomedical control systems, closed-loop ventilation.

1. INTRODUCTION

Oxygenation is one of the key parameters for monitoring and control in intensive care and critical care medicine, especially for patients with acute respiratory distress syndrome (Pomprapa et al. (2014b, 2015)). In such critical situations of oxygen deficiency, oxygen therapy is required by adjusting the fraction of inspired oxygen (FiO_2) in order to maintain tissue and brain function (Claure and Bancalari (2013)). Those patients require not only the stabilization but also an improvement of the oxygenation, which can be measured in terms of arterial oxygen tension (PaO_2) or arterial oxygen saturation (SaO_2) (Pomprapa et al. (2014a)). Therefore, we focus on hypoxia management, which results in a single-input single-output (SISO) system during mechanical ventilation therapy. In this article, FiO_2 and SaO_2 are regarded as the input and output variables of the system, respectively.

Because of the complexity in biomedical systems such as oxygenation dynamics, nonlinearities and uncertainties are common in control practice. In addition, a mathematical model describing the system is difficult to identify accurately for individuals under time constraint, leading to a great challenge in the control strategy. The PIA has come into our attention in this particular application because it uses only partial knowledge of the system dynamics (Vrabie et al. (2009); Vrabie (2009)), namely only the knowledge of input dynamics with all accessible states. PIA is classified as a reinforcement learning approach with the actor-critic architecture, where an actor subsystem performs the optimal action in each state and a critic subsystem evaluates the long-term performance for each state (Sutton et al. (1992)). Therefore, this technique is mainly based on a two-step iteration, namely policy improvement and policy evaluation. These steps should be carried out iteratively until obtaining the converged optimal solution (Vamvoudakis et al. (2009)). This technique should then be suitable for oxygen therapy during the critical time, which requires neither a complete mathematical model nor an implementation of system identification.

For a linear system, the optimal solution can be achieved by optimizing a Hamiltonian function based on an infinite horizon optimal control problem, so called the well-known linear quadratic regulator (LQR) problem with state feedback structure, and the solution can be derived by solving the algebraic Riccati equation (ARE). However, for a nonlinear system, the Hamilton-Jacobi-Bellman (HJB) equation has to be solved to obtain the estimated optimal solution, instead of the ARE, which serves as a foundation of this technique in synthesizing a dynamic controller. The PIA controller has been applied in some practical examples: a DC-DC converter (Wernrud (2007)), a power plant for the optimal-load-frequency controller (Wang et al. (1993)), a hybrid system of a jumping robot (Suda and Yamakita (2013)), a double-link pendulum for swing-up control and an adaptive steering control of a tanker ship (Xu et al. (2007)). Therefore, the PIA should work out in a closed-loop control of oxygenation using mechanical ventilation therapy. In addition, a proof of concept for the performance of this technique is implemented in a computerized nonlinear interacting three-tank system for the control of water level. The resulting control performance is also demonstrated in this work.

This particular contribution is organized as follows. It begins with the control system design in section 2 to provide the mathematical foundation of the PIA technique for the nonlinear optimal control based on the HJB equation. In section 3, modelling and control of oxygenation is proposed

10.1016/j.ifacol.2015.10.193

based on a first-order time delay. Furthermore, a practical implementation of the control algorithm is given in section 4 based on the control of water level for the nonlinear three-tank system. A discussion is intensively provided in section 5 and the article ends with the conclusion in section 6.

2. CONTROL SYSTEM DESIGN

2.1 Nonlinear Optimal Control Based on the HJB Equation

Let us consider a continuous-time nonlinear dynamical system in the following form

$$\dot{x}(t) = \mathcal{F}(x(t), u(t)), \tag{1}$$

where $x(t) \in \chi \subseteq \mathbb{R}^n$ is the states of the system, $u(t) \in \upsilon \subseteq \mathbb{R}$ represents the control input, and $\mathcal{F} : \chi \times \upsilon \times \mathbb{R}_+ \rightarrow \mathbb{R}^n$ is Lipschitz continuous on $\chi \times \upsilon$, such that the state vector $x(t)$ is unique for a given initial condition x_0. It is assumed that the system is stabilizable.

The control objective is to minimize the cost function satisfying eq. (2), which is used as a policy evaluation.

$$V(x_0) = \int_0^\infty r(x(\tau), u(\tau))d\tau, \tag{2}$$

where $r(x, u)$ is defined by $Q(x) + u^T R u$, $Q(x) \in \mathbb{R}$ is positive definite and continuously differentiable (i.e. if $x = 0$, then $Q(x) = 0$ and $Q(x) > 0$ for all x), and $R \in \mathbb{R}$ is a positive definite matrix for a penalty or weighting of the control input.

This particular system dynamics can be written in an affine form, as follows.

$$\dot{x}(t) = f(x) + g(x) \cdot u \tag{3}$$

Let us define the Hamiltonian of the control problem in eq. (4).

$$\begin{aligned} H(x, u, V_x^*) = r(x(t), u(t)) + \nabla_x V^*(f(x(t)) \\ + g(x(t))u(t)), \end{aligned} \tag{4}$$

where the optimal cost function $\nabla_x V^*$ satisfies the following HJB equation and ∇_x denotes the partial differential in x.

$$\min_u H(x, u, \nabla_x V^*) = 0 \tag{5}$$

The optimal solution is given by

$$u^* = -\frac{1}{2} R^{-1} g^T(x) \nabla_x V^*(x), \tag{6}$$

which is an algorithm for policy improvement (Ohtake and Yamakita (2010)). Note that, for a linear continuous-time system, the HJB equation becomes the well-known Riccati equation (Vamvoudakis et al. (2009)) and the converged solution is equivalent to the response from a LQR controller.

2.2 Policy Iteration Algorithm

The cost fuction of eq. (2) can be rewritten in the form of eq. (7).

$$V(x(t)) = \int_t^{t+T} r(x(\tau), u(\tau))d\tau + V(x(t+T)), \tag{7}$$

where $V(0) = 0$. Hence, eq. (7) is numerically solved for $V(x(t))$ as a function of time and the control signal u^* can be updated based on eq. (6). The function of $V(x(t))$ can then be estimated by either a linear function or a nonlinear function in terms of $x(t)$, so that the unknown parameters can be computed by using a simple least squares algorithm, a gradient descent algorithm, a recursive least squares algorithm (Vrabie et al. (2009)) or using a neural network (Bhasin et al. (2013)) for the parameter estimation. The proof of guarantee convergence for the control policy as well as stability are given by Vrabie et al. (2009) in case of a linear system and by Vamvoudakis and Lewis (2010) in case of a nonlinear system. The block diagram of PIA is shown in Fig. 1 with internal actor and critic subsystems of state feedback architecture.

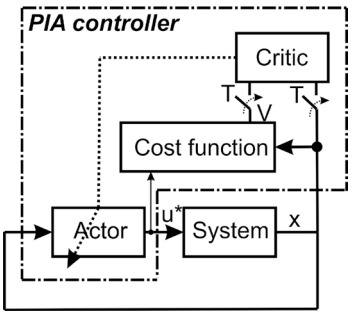

Fig. 1. Block diagram of the policy iteration algorithm with actor and critic subsystems of full state feedback structure.

In summary, the method of PIA can be formulated in the following sequential steps:

(1) Assign an initial controller that can stabilize the system.
(2) Compute the time response of $V(x(t))$ numerically, where $V(0) = 0$.
(3) Based on the response of $V(x(t))$ in the time domain, estimate the unknown parameters in the evaluation function by using the least squares algorithm (Vrabie et al. (2009)) or using other parameter estimation techniques.
(4) Update the control policy of eq. (6) as a policy improvement.
(5) Return to compute the time response of $V(x(t))$ at step (2) and continue the sequential steps for policy evaluation, if the estimated parameters are not converged or the norm between current estimated parameters and previous estimated parameters is more than a predefined value (ϵ). Otherwise, an iteration of the updated control law should be stopped.

The prerequisite for this algorithm is to have an initial controller that can stabilize the system. In practice, a simple proportional-integral (PI) controller may be used as the controller for the first evaluation of the system performance. Thereafter, PIA will optimize the Hamiltonian cost function at every iteration to achieve the extremal cost for the underlying nonlinear system. With these steps, online implementation can be realized to have an optimal controller for the complicated nonlinear plant with unknown internal dynamics.