Biomedical Signal Processing and Control 22 (2015) 54–64

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

journal homepage: www.elsevier.com/locate/bspc

Closed-loop control of anesthesia and mean arterial pressure using reinforcement learning

Regina Padmanabhan [a], Nader Meskin [a,*], Wassim M. Haddad [b]

[a] Department of Electrical Engineering, Qatar University, Qatar
[b] School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0150, USA

ARTICLE INFO

Article history:
Received 4 November 2014
Received in revised form 30 April 2015
Accepted 21 May 2015

Keywords:
Active drug dosing
Anesthesia control
Hemodynamic regulation
Reinforcement learning

ABSTRACT

General anesthesia is required for some patients in the intensive care units (ICUs) with acute respiratory distress syndrome. Critically ill patients who are assisted by mechanical ventilators require moderate sedation for several days to ensure cooperative and safe treatment in the ICU, reduce anxiety and delirium, facilitate sleep, and increase patient tolerance to endotracheal tube insertion. However, most anesthetics affect cardiac and respiratory functions. Hence, it is important to monitor and control the infusion of anesthetics to meet sedation requirements while keeping patient vital parameters within safe limits. The critical task of anesthesia administration also necessitates that drug dosing be optimal, patient specific, and robust. In this paper, the concept of reinforcement learning (RL) is used to develop a closed-loop anesthesia controller using the bispectral index (BIS) as a control variable while concurrently accounting for mean arterial pressure (MAP). In particular, the proposed framework uses these two parameters to control propofol infusion rates to regulate the BIS and MAP within a desired range. Specifically, a weighted combination of the error of the BIS and MAP signals is considered in the proposed RL algorithm. This reduces the computational complexity of the RL algorithm and consequently the controller processing time.

1. Introduction

Recent research in clinical pharmacology has focused on identifying "best practices" for ensuring patient safety by maximizing the desired drug effect and minimizing the drug induced side effects. This is particularly important when developing paradigms for anesthetic drug dosing. Surgical patients typically require deep sedation over a short duration of time. However, ICU patient sedation, especially patients assisted by mechanical ventilation to treat pulmonary insufficiency, can be more challenging [1]. Critically ill patients who are assisted by mechanical ventilators require moderate sedation for several days to ensure cooperative and safe treatment in the ICU. Moreover, clinical research shows that closed-loop control of anesthetic drug administration can have positive outcomes in terms of patient safety, early recovery, reduced treatment cost, and effective and practical use of clinician expertise [1–4].

When continuous anesthetic drug infusion is used, a protocol incorporating daily awakening from sedation is advocated for minimizing sedative accumulation, slowing the build up to drug tolerance, and reducing the length of ICU stay [5]. Accounting for the residual drug effect of prior sedation (i.e., the effect of the drug left over due to sedation before interruption) while calculating the drug dose to be administrated after the period of daily interruption of sedation, further restricts the probability of oversedation and the associated complications. Ideally, anesthetic drug dosing should account for the patient's physiological condition, drug interaction due to coadministration of several anesthetic and physiologic drugs, residual drug effect due to daily interruption from sedation, physiological disturbances such as hemorrhage or renal impairment, interpatient variability, and changes in the characteristics of the monitoring devices and the infusion pump apparatus.

Optimal drug dosing that considers the aforementioned factors is essential since oversedation or undersedation is not acceptable. Oversedation can cause hypotension, prolonged recovery time, delayed weaning from mechanical ventilation, ileus, nausea, and immunosuppression; whereas undersedation can cause anxiety, agitation, hyperoxia, tachycardia, myocardialischemia, atelectasis, tracheal tube intolerance, and infection [6]. Achieving acceptable clinical effects, while avoiding or minimizing undesired effects, is

* Corresponding author. Tel.: +974 44034224.
E-mail addresses: regina.ajith@qu.edu.qa (R. Padmanabhan), nader.meskin@qu.edu.qa (N. Meskin), wm.haddad@aerospace.gatech.edu (W.M. Haddad).

http://dx.doi.org/10.1016/j.bspc.2015.05.013
1746-8094/

a major objective in general anesthesia. Furthermore, open-loop control can be tedious, imprecise, and time-consuming. Hence, closed-loop control for anesthesia administration is imperative to improve quality of medical care and to restrain the increasing cost of health care [1].

Currently, intraoperative anesthesia administration is facilitated manually or is assisted by an open-loop target controlled infusion (TCI) pump, which is programmed using a nominal patient model to calculate the required drug dose. However, due to the interdisciplinary (medicine–mathematics–engineering) nature of the problem and the associated clinical and ethical constraints in performing clinical experiments, there is a lack of accurate mathematical models that characterize the drug disposition (pharmacokinetics) and the drug effect (pharmacodynamics) in the human body. The case of mechanically ventilated critically ill patients in the ICU is challenging since such patients require administration of multiple drugs to regulate key physiological variables, such as the level of unconsciousness, heart rate, mean arterial pressure (MAP), respiratory rates, and other vital parameters within desired limits.

A few clinical and in silico trials have been conducted to validate various closed-loop control strategies for anesthesia administration [2–4,7]. Due to system complexity and system uncertainty, however, fixed-gain linear controllers have proved inadequate [1,2]. Investigations using model predictive controllers (MPC) for surgical patients with system constraints have also proved deficient in terms of prolonged parameter identification time and model dependence [8]. Even though optimal control strategies can offer "the best" solution for a given system with a given set of state and control constraints, the method is model-based and requires refinements to address system uncertainties and system disturbances [9,10]. In general, adaptive disturbance rejection controllers can work well without an accurate system model in the presence of system uncertainties and system disturbances [11,12]. However, adaptive controllers cannot directly address system optimality considerations. Hence, it is imperative to develop control techniques that can account for system modeling uncertainty and system disturbances, while providing optimal solutions that improve the reliability and applicability of closed-loop control for ICU sedation.

Reinforcement learning (RL) is a developing and promising approach which offers an ideal framework for on-line identification and control of complex uncertain nonlinear dynamical systems [13]. RL allows for the learning of optimal actions without the knowledge of the complete system dynamics (or system disturbances). Moreover, since the controller (RL agent) design is performed by interacting with the system, unknown and time-varying dynamics as well as changing performance requirements can be accounted for by the controller. RL exploits the computational efficiency and speed of digital computers to stochastically employ all possible control actions and assesses a best or optimal action.

Reinforcement learning-based feedback control methods have demonstrated promising performance in robot control, wind turbine speed control, image evaluation, and autonomous helicopter control [14,15]. In medical pharmacology, reinforcement learning has been used for long-term clinical planning tasks, such as the optimization of erythropoietin dosage for the treatment of anemia in hemodialysis patients [16]. RL methods basically explore the response of a system for every possible action and then learn the optimal action by evaluating how close the last action drives the system towards a desired state. The controller then exploits the learned optimal policies. RL is suitable for drug disposition control scenarios as it does not rely on a system model and learns optimal control policies based on the response to the control actions (drug infusion) of the system.

In [17], the authors discuss RL-based optimal control of hypnosis for intraoperative patients. Specifically, the authors modeled the drug disposition system as a discrete-time system with three states corresponding to $\Delta BIS > 0$, $\Delta BIS < 0$, and $\Delta BIS = 0$, where ΔBIS denotes the change in the bispectral index (BIS); and three control actions (propofol dose) u corresponding to 0 mg, 20 mg, or 40 mg. The BIS index is derived from the electroencephalogram and provides a measure of depth for anesthesia [18]. In [19], the authors present the first clinical trial for closed-loop control of anesthesia administration using reinforcement learning on 15 human volunteers. In this study, RL demonstrated patient specific control of anesthesia administration marked by improved control accuracy as compared to performance metrics of other studies reported in the literature.

Surgery is a highly uncertain and hostile environment. Sedation requirements during surgery typically involve moderate to deep sedation for a short duration. In the case of ICU sedation, however, even though sedation requirements are light to moderate, they usually require long term (for several days) continuous infusion of anesthetics; and since the patient is critically ill, several life supporting drugs are typically required. Hence, drug interaction can be a factor. In addition, sedated patients may require daily interruption from sedation to reduce drug tolerance development and overall drug dosage. Hence, residual drug effect due to prior sedation periods need to be accounted.

Furthermore, long term anesthetic infusion often results in drug habituation, and hence, patient pharmacologic response may change. Hence, the case of ICU sedation is challenging and necessitates long term maintenance of moderate sedation along with the regulation of vital physiological parameters of critically ill patients. Propofol administration lowers sympathetic tone and causes vasodilation, which can decrease preload and cardiac output and consequently lower the mean arterial pressure and other interrelated hemodynamic parameters. This can lead to blood pressure instability, overdose, and cardiovascular collapse [20]. Therefore, ensuring a desired range for MAP as one of the important hemodynamic parameters is vital during propofol infusion [21,22].

The main objective of this paper is to apply reinforcement learning for the control of continuous intravenous infusion of propofol for ICU patients by utilizing the BIS index, while simultaneously regulating the mean arterial pressure at a desired range. Specifically, a weighted combination of the error of the BIS and MAP signals is considered in the proposed RL algorithm. This reduces the computational complexity of the RL algorithm and consequently the controller processing time. The proposed method is tested by means of simulations on 30 randomized simulated patients. Moreover, the paper presents a general framework to utilize RL-based methods such as the Q-learning algorithm for the control of multiple parameters in nonlinear dynamical systems.

The remainder of the paper is organized as follows. Section 2 presents a RL-based control problem for dynamical systems and illustrates the development of an optimal control policy using a Q-learning algorithm. In addition, the pharmacokinetics and pharmacodynamics of the drug propofol in human body are discussed. In Section 2.4, the implementation of a reinforcement learning-based, closed-loop control predicated on BIS and MAP measurements is presented. Then, in Section 3, simulation results are provided and the performance of the proposed framework is evaluated. In Section 4, the limitations of this study are discussed. Finally, in Section 5, conclusions and recommendations for future work are presented.

2. Methods

In this section, the development of a RL-based control agent for the control of dynamical systems is presented. Subsequently, the pharmacological model of propofol with respect to the bispectral index and mean arterial pressure is introduced. This model is used

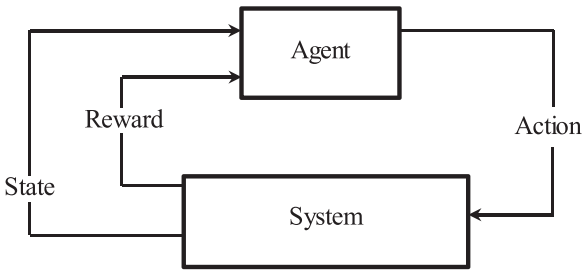


Fig. 1. Reinforcement learning schematic.

to represent a patient model which is then used to train the RL agent.

2.1. Problem formulation

Consider the nonlinear dynamical system given by

$$\dot{x}(t) = f(x(t), u(t)), \quad x(0) = x_0, \quad t \geq 0, \tag{1}$$

$$y(t) = h(x(t)), \tag{2}$$

where for every $t \geq 0$, $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}$ is the control input, $y(t) \in \mathbb{R}^l$ is the output of the system, $f : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$ is locally Lipschitz continuous, and $h : \mathbb{R}^n \to \mathbb{R}^l$ is continuous. The problem of deriving control laws that track a desired trajectory can be considered as a sequential decision making problem represented by a finite Markov decision process (MDP) [13]. Reinforcement learning-based approaches are suitable in solving MDPs for goal oriented decision making [23]. The goal is to reach a desired terminal state with decisions based on the best sequence of actions that will transition the system to the desired state.

A finite MDP is characterized by the 4-tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$, where $\mathcal{S}$ represents the environment or system in terms of a finite set of states, $\mathcal{A}$ is a finite sequence of actions available for the states $s_k \in \mathcal{S}$, $\mathcal{P}$ is a state transition probability matrix, with $\mathcal{P}_{a_k}(s_k, s_{k+1})$ denoting the probability that an action $a_k \in \mathcal{A}$ takes the state $s_k \in \mathcal{S}$ at time step k to the state s_{k+1} at time step $k+1$, and $\mathcal{R}$ is the associated reward function that quantifies the desirability of an action $a_k \in \mathcal{A}$ for all $s_k \in \mathcal{S}$. The transition probability matrix $\mathcal{P}$ depends on the system dynamics represented by the function $f(\cdot, \cdot)$ which is assumed to be unknown. The elements in the finite sequence $\mathcal{A}$ are represented as $(\mathcal{A}_i)_{i \in \mathbb{I}^+}$, where $\mathbb{I}^+ \triangleq \{1, 2, \ldots, M\}$ and M denotes the total number of actions.

Reinforcement learning approaches, such as Q-learning [24], have gained considerable attention in recent years as a learning method that does not need an accurate system model and can be used online while the system changes during the learning process. Fig. 1 shows the schematic of a model-independent reinforcement learning approach where the agent or controller imparts an action on the system and a reward is observed associated with the state transition to learn a useful policy or an action plan. To facilitate learning, the discrete states $s_k \in \mathcal{S}$ representing the behavior of the system should be measurable. Hence, we define the states of the reinforcement learning environment in terms of the system output $y(t)$, $t \geq 0$, as

$$s_k = g(y(t)), \quad kT \leq t < (k+1)T, \tag{3}$$

where $g : \mathbb{R}^l \to \mathcal{S} \subset \mathbb{R}$ is a piecewise continuous mapping between the system output and the finite state representation in the RL framework, $k = 1, 2, \ldots$, and $T \geq 0$.

The Q-learning algorithm can train an agent to control the states s_k of the RL environment without knowledge of the system state $x(t)$, $t \geq 0$; only the data measured along the system trajectories at time steps $k \in \{1, 2, \ldots\}$, $kT \leq t < (k+1)T$, is required. Specifically, at each time step k, the agent observes the system to determine the current state s_k from the set of states $\mathcal{S}$ and selects an action a_k from the action sequence $\mathcal{A}$. In response, the system stochastically transitions to a new state s_{k+1} with a numerical reward $r_{k+1} \in \mathbb{R}$. The agent seeks to maximize the reward it receives over an infinite horizon. A common objective is to choose each action a_k so as to maximize the expected value of the discounted return [24], [19] given by

$$J(r_k) = \mathbb{E}\left[\sum_{k=1}^{\infty} \theta^{(k-1)} r_k\right], \tag{4}$$

where $\mathbb{E}[\cdot]$ denotes expectation and $\theta \in [0, 1]$ is a *discount rate parameter* which represents the horizon of interest to the agent. For $\theta = 0$ and $k \in [1, \infty)$, $J(r_k) = r_1$, that is, for learning, the agent considers only the current reward. Alternatively, for θ approaching 1, the weight of the costs incurred in the future is increased.

2.2. Learning an optimal policy

Reinforcement learning methods attempt to improve the agent's decision making policy over time as the agent learns the optimal policy from an initial arbitrary policy. A policy is a mapping from states to actions or to the probability distributions over the actions [23]. In RL frameworks, a policy can be a path plan to move from an initial position to the target position; it can be a rule base or a look-up-table such as "if in this state, then do this." RL-based control is predicated on learning an optimal policy while interacting with the system.

The most widely used reinforcement learning algorithms, such as the Watkins Q-learning algorithm [24], use each state transition to update each entry of a Q table which forms the control policy. The policy is stored in a table so that appropriate responses can be retrieved quickly with respect to the state of the system. The entry $Q(s_k, a_k)$ of the Q table for each pair of state s_k and action a_k at time step $k \in \{1, 2, \ldots\}$ represents the quality of the state-action pair. The agent or controller observes the measured variables, which reflect the behavior or status quo of the system, and executes actions using a learned optimal control policy represented as $Q(s_k, a_k) : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$.

For every time step k and state s_k, the agent or controller chooses the action a_k as

$$a_k = (\mathcal{A}_i)_{i \in \mathbb{I}^+}, \quad i = \text{argmax} Q(s_k, \cdot). \tag{5}$$

The numerical reward $r_k \in \mathbb{R}$ guides the agent to whether the action chosen at the time step k was "good" or "bad." After the transition $s_k \to s_{k+1}$, having taken an action a_k and received a reward r_{k+1}, the algorithm is updated by

$$Q_k(s_k, a_k) \leftarrow Q_{k-1}(s_k, a_k) + \eta_k(s_k, a_k)[r_{k+1} + \theta \max_{a_{k+1}} Q_{k-1}(s_{k+1}, a_{k+1}) - Q_{k-1}(s_k, a_k)], \tag{6}$$

where $\eta_k(s_k, a_k) \in [0, 1)$ is the step size parameter or learning rate that governs the size of adjustment after each experiment and θ is the discount rate parameter defined in (4).

There are various proofs in the literature that show the convergence of the Q-learning algorithm (6) to the optimal Q-function that maximizes (4) [24,25,23]. In particular, in [25] it has been shown that (6) converges to the optimal Q-function with probability one as long as

$$\sum_{k=1}^{\infty} \eta_k(s_k, a_k) = \infty, \qquad \sum_{k=1}^{\infty} \eta_k^2(s_k, a_k) < \infty, \quad (s_k, a_k) \in \mathcal{S} \times \mathcal{A}. \tag{7}$$

Note that $\sum_{k=1}^{\infty} \eta_k(s_k, a_k) = \infty$ requires all state-action pairs be visited infinitely often, whereas $\sum_{k=1}^{\infty} \eta_k^2(s_k, a_k) < \infty, (s_k, a_k) \in \mathcal{S} \times \mathcal{A}$,

is required to ensure convergence with probability one. Eq. (6) involves a temporal difference learning algorithm and is suitable for dynamical systems since updates are made at each time step as observations of the data are made along a particular system trajectory. The Q-learning algorithm is initialized with an initial arbitrary estimate of the unknown $Q(s, a)$ and then iterative estimate updates are performed until convergence is reached; that is, until the change in the Q table, denoted by ΔQ, is equal to zero, or when the updates satisfy a minimum threshold required by a given control task satisfying $\Delta Q \leq \delta$, where δ is a prespecified tolerance parameter.

The framework discussed in this section is used to develop an anesthesia controller based on Q-learning for closed-loop regulation of the bispectral index and mean arterial pressure by controlling the continuous infusion of propofol. Specifically, our aim is to develop a controller for regulating the system output variable $y(t) \in \mathbb{R}^l$, $t \geq 0$, using a single control input $u(t) \in \mathbb{R}$, $t \geq 0$. Since anesthesia administration is a life critical task, one cannot experiment on the patient with random actions to arrive at optimal policies. Instead, we use simulated patients as modeled in the following subsection to learn the optimal policy. In this case, the RL system shown in Fig. 1 is replaced by a patient model represented by the nominal pharmacokinetic and pharmacodynamic model shown in Fig. 2.

2.3. Pharmacokinetic and pharmacodynamic patient model

Propofol infusion leads to sedation which is usually quantified in terms of the BIS index. In addition, because most anesthetics lower sympathetic tone and induce venodilation, they indirectly affect the mean arterial pressure of the patient. Specifically, propofol infusion reduces the cardiac output of the patient, which in turn reduces the drug disposition. Hence, nonlinear pharmacokinetics and pharmacodynamics lead to a nonlinear dynamical patient model $f(x(t), u(t))$, where $u(t)$, $t \geq 0$, is the continuous intravenous infusion of propofol and $x(t)$, $t \geq 0$, denotes the system states.

In this paper, a nonlinear three-compartment model with an effect-site compartment is used for representing the patient dynamics controlled by a continuous intravenous (to the central compartment) infusion of propofol. In this model, the mass of the drug in the intravascular blood (i.e., blood within the arteries or the veins) as well as the highly perfused organs (organs with high ratios of perfusion to weight) such as the heart, brain, kidneys, and liver is denoted by $x_1(t)$, $t \geq 0$. The remainder of the drug in the body is assumed to reside in two peripheral compartments, comprised of muscle and fat, and the masses in these compartments are denoted by $x_2(t)$, $t \geq 0$, and $x_3(t)$, $t \geq 0$, respectively. These compartments receive less than 20% of the cardiac output.

A mass balance for the three-compartment model yields [3,21,26]

$$\dot{x}_1(t) = -[a_{11}(c(t)) + a_{21}(c(t)) + a_{31}(c(t))]x_1(t) + a_{12}(c(t))x_2(t) + u_1(t), \quad x_1(0) = x_{10}, \quad t \geq 0, \tag{8}$$

$$\dot{x}_2(t) = a_{21}(c(t))x_1(t) - a_{12}(c(t))x_2(t), \quad x_2(0) = x_{20}, \tag{9}$$

$$\dot{x}_3(t) = a_{31}(c(t))x_1(t) - a_{13}(c(t))x_3(t), \quad x_3(0) = x_{30}, \tag{10}$$

$$\dot{c}_{\text{eff}}(t) = a_{\text{eff}}(x_1(t)/V_c - c_{\text{eff}}(t)), \quad c_{\text{eff}}(0) = c_{\text{eff0}}, \tag{11}$$

where $a_{ij}(c)$, $i, j = 1, 2, 3$, denote the nonnegative mass transfer coefficients between the jth and ith compartment, $c(t)$, $t \geq 0$, is the drug concentration in intravascular blood, and V_c is the volume of the central compartment. The drug effect in terms of the BIS and MAP is linear for lower drug doses; however, higher drug dose and prolonged drug titration result in a nonlinear saturation (i.e., sigmoidal) effect described by the Hill equation [21].

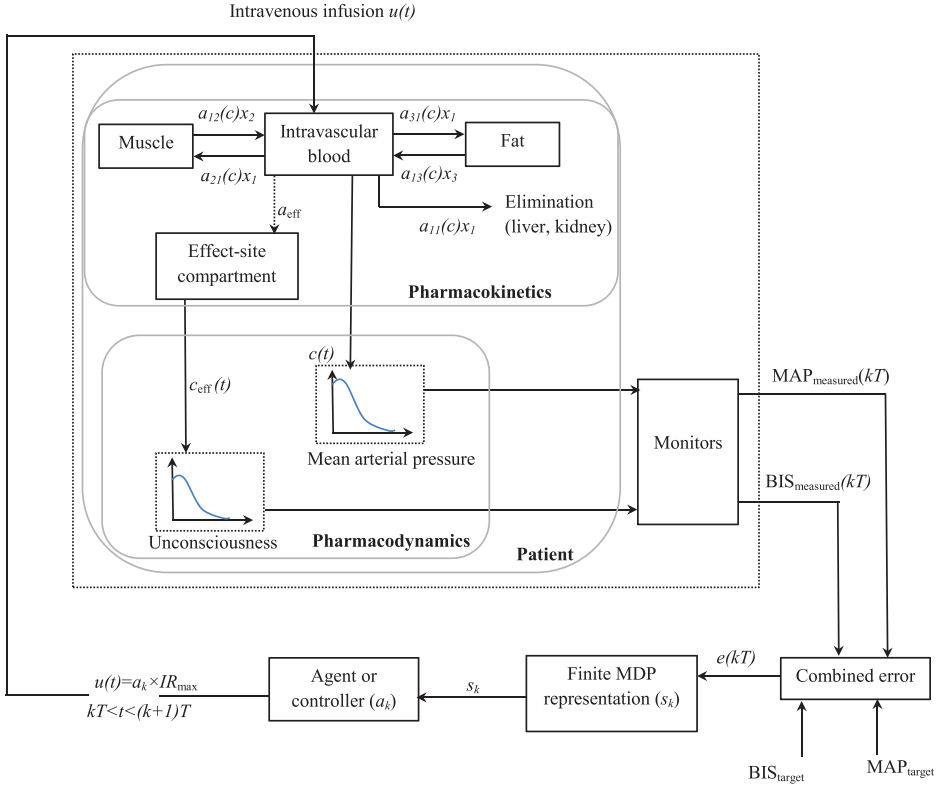


Fig. 2. Closed-loop control using reinforcement learning showing the interaction between the agent and the simulated patient for simultaneous regulation of BIS and MAP management.

Thus, the function $h(\cdot)$ in (2) is a nonlinear function given by $h(x) = [\mathrm{BIS}_{\mathrm{measured}}(c_{\mathrm{eff}}), \mathrm{MAP}_{\mathrm{measured}}(c)]^{\mathrm{T}}$, where $\mathrm{BIS}_{\mathrm{measured}}(c_{\mathrm{eff}})$ and $\mathrm{MAP}_{\mathrm{measured}}(c)$ are the drug effects captured by

$$\mathrm{BIS}_{\mathrm{measured}}(c_{\mathrm{eff}}(t)) = \mathrm{BIS}_0 \left(1 - \frac{(c_{\mathrm{eff}}(t))^{\gamma}}{(c_{\mathrm{eff}}(t))^{\gamma} + (\mathrm{EC}_{50})^{\gamma}}\right), \tag{12}$$

$$\mathrm{MAP}_{\mathrm{measured}}(c(t)) = \mathrm{MAP}_0 \left(1 - \frac{(c(t))^{\alpha}}{(c(t))^{\alpha} + \mathrm{C}_{50}^{\alpha}}\right), \tag{13}$$

where BIS_0 denotes the base line value, which, by convention, is typically assigned a value of 100 to represent an awake state, EC_{50} is the concentration at half maximal effect (of the BIS) and represents the patient's sensitivity to the drug, γ determines the degree of nonlinearity, MAP_0 is the initial value of mean arterial pressure of the patient prior to propofol infusion, and α and C_{50} represent the degree of nonlinearity and concentration at half maximal effect (of the MAP), respectively. For further details on the parameters used and other related issues to pharmacokinetic and pharmacodynamic modeling used in this study; see [21].

2.4. Closed-loop control of BIS and MAP using RL

In this section, the concept of RL-based control and the Q-learning algorithm is utilized to develop a drug dosing algorithm for the simultaneous control of anesthesia and hemodynamic management. For the dynamical system given by (1) representing the patient dynamics, the control variable $u(t)$, $t \geq 0$, is the continuous intravenous infusion of propofol. However, in the RL framework, the agent interacts with the patient at discrete time steps. Specifically, the propofol infusion rate at each time step k is given by

$$IR_k = a_k \times IR_{\max}, \tag{14}$$

where $k \in \{1, 2, \ldots\}$, $IR_{\max}$ is the maximum allowable infusion rate, and a_k is a particular action from the action sequence $\mathcal{A}$ selected at the kth time step. Thus, between any two time steps k and $k+1$, the infusion rate remains constant and is given by $u(t) = IR_k$, $kT \leq t < (k+1)T$, where T is the time duration between any two time steps. The action $a_k = (\mathcal{A}_i)_{i \in \mathbb{I}^+}$ at the kth time step can vary from 0 (no infusion) to 1 (maximum rate of infusion) within the finite action sequence $\mathcal{A} = (0, 0.01, 0.02, 0.03, 0.04, 0.05, 0.08, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1)$, where $\mathbb{I}^+ = \{1, 2, \ldots, 20\}$. Since $IR_{\max}$ is a configurable parameter, one of the benefits of the infusion rate scheme given by (14) is that it is easy to set its value according to the sedation requirements of the patient in the ICU.

In reinforcement learning the controller or agent makes a decision about the action to be taken at each time step based on the current state of the system $s_k = g(y(t)), s_k \in \mathcal{S}, t \in [kT, (k+1)T)$. Hence, the states s_k of the system should be observable for decision making. Therefore, we define the states s_k of the reinforcement learning system based on the measurable parameters $\mathrm{BIS}_{\mathrm{measured}}(c_{\mathrm{eff}}(t))$ and $\mathrm{MAP}_{\mathrm{measured}}(c(t))$, $kT \leq t < (k+1)T$. Specifically, in this paper, the state s_k is defined based on the error $e(t)$, $kT \leq t < (k+1)T$, given by

$$e(t) = \sqrt{\beta \mathrm{BIS}_{\mathrm{error}}^2(t) + \mathrm{MAP}_{\mathrm{error}}^2(t)}, \tag{15}$$

where $\beta > 0$ is a weighing factor, which can be used to weigh the importance of anesthesia control over hemodynamic control,

$$\mathrm{BIS}_{\mathrm{error}}(t) = \frac{\mathrm{BIS}_{\mathrm{measured}}(c_{\mathrm{eff}}(t)) - \mathrm{BIS}_{\mathrm{target}}}{\mathrm{BIS}_{\mathrm{target}}} \times 100, \tag{16}$$

and

$$\mathrm{MAP}_{\mathrm{error}}(t) = \frac{\mathrm{MAP}_{\mathrm{measured}}(c(t)) - \mathrm{MAP}_{\mathrm{target}}}{\mathrm{MAP}_{\mathrm{target}}} \times 100. \tag{17}$$

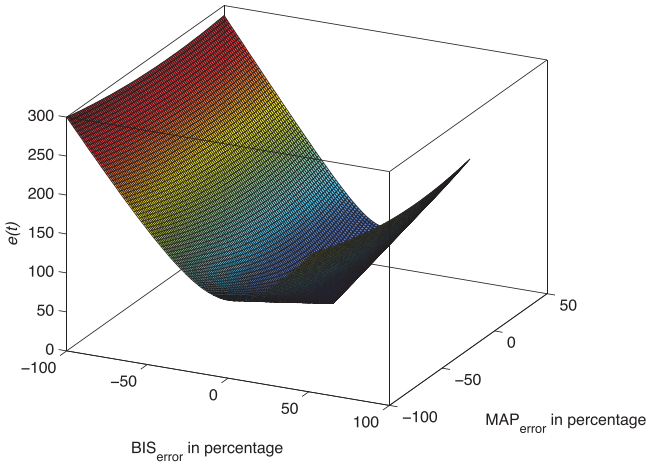


Fig. 3. Normalized percentage error of BIS and MAP versus combined error $e(t)$.

For ICU sedation, the goal is to learn the sequence of infusion rates that result in a minimum $\mathrm{BIS}_{\mathrm{error}}$ and $\mathrm{MAP}_{\mathrm{error}}$. Thus, defining the system state s_k based on $e(t)$, $t \geq 0$, the agent can consider the single measurement given by (15) for training purposes rather than two separate measurements involving the $\mathrm{BIS}_{\mathrm{error}}(t)$ and $\mathrm{MAP}_{\mathrm{error}}(t)$, $t \geq 0$. This reduces the complexity of the training algorithm considerably. Specifically, in this case, the agent acts based on measurements from both the BIS and MAP simultaneously. The parameters BIS and MAP can be directly measured in the ICU; however, for our simulations, the BIS and the MAP are calculated based on the propofol concentration using the pharmacodynamic models (12) and (13). To account for the possible measurement constraints in the BIS and MAP sensors, a sampling time $T = 6$ seconds is selected. Note that the agent interacts with the patient at discrete time steps, that is every 6 seconds [19].

For our simulation, we considered $\mathrm{BIS}_{\mathrm{measured}}(t) \in [0, 100]$ and $\mathrm{MAP}_{\mathrm{measured}}(t) \in [0, 120]$, and set $\mathrm{BIS}_{\mathrm{target}} = 65$ and $\mathrm{MAP}_{\mathrm{target}} = 80$. Thus, $\mathrm{BIS}_{\mathrm{error}}(t)$, $t \geq 0$, is positive for $\mathrm{BIS}_{\mathrm{measured}}(t) \in (65, 100]$, and negative for $\mathrm{BIS}_{\mathrm{measured}}(t) \in [0, 65)$. However, using (15), the value of $e(t)$ for $\mathrm{BIS}_{\mathrm{measured}}(t) \in (65, 100]$ and $\mathrm{BIS}_{\mathrm{measured}}(t) \in [30, 65)$ are the same as shown in Fig. 3. To achieve $\mathrm{BIS}_{\mathrm{target}}$, the infusion of the sedative drug propofol should be increased if $\mathrm{BIS}_{\mathrm{error}}(t)$ is positive and the infusion should be decreased if $\mathrm{BIS}_{\mathrm{error}}(t)$ is negative for a given time $t > 0$. The state action association table (the Q table) should reflect this discrepancy after training. Hence, at any time step k, we assign $s_k \in \{1, 2, \ldots, 13\}$ for $e(kT) \in [0, e_{\mathrm{p}}(t)]$, where $e_{\mathrm{p}}(t)$ denotes the maximum error in the region of error $e(kT)$ where $\mathrm{BIS}_{\mathrm{error}}(t)$ is positive and $s_k \in \{14, 15, \ldots, 20\}$ for $e(kT) \in [0, e_{\mathrm{n}}(t)]$, where $e_{\mathrm{n}}(t)$ denotes the maximum error in the region of error $e(kT)$ where $\mathrm{BIS}_{\mathrm{error}}(t)$ is negative. Table 1 shows the mapping between $e(kT)$ and s_k; note that if $e(kT) \in [0, 2]$, then $s_k = 1$. Hence, the entries in the Q table are updated corresponding to the states $s_k = 1$ to $s_k = 13$ for positive values of $\mathrm{BIS}_{\mathrm{error}}(t)$, $t \geq 0$, and $s_k = 14$ to $s_k = 20$ for negative values of $\mathrm{BIS}_{\mathrm{error}}(t)$, $t \geq 0$, using (6).

For our numerical implementation, we choose a dense discretization in the neighborhood of $e(kT) = 0$ for training the Q-function (see Table 1). Moreover, in the region of error $e(kT)$ where $\mathrm{BIS}_{\mathrm{error}}(t)$ is positive we use 13 states, as the optimal control action should vary with the value of $e(kT)$ so as to minimize oversedation or undersedation of the patient. The patient is oversedated in the region of error $e(kT)$ where $\mathrm{BIS}_{\mathrm{error}}(t)$ is negative, and hence, the ideal infusion rate should be zero as the error $e(kT)$ approaches the value 300. Thus, a dense discretization is not required and hence we assign only 7 states in this region.

The basic idea of RL lies in the judicious choice of the reward function, which is used to reinforce the agent. For ICU sedation,

Table 1
State assignment based on $e(t)$.

$BIS_{error} > 0$		$BIS_{error} < 0$	
State s_k	$e(kT)$	State s_k	$e(kT)$
1	[0, 2]	14	[0, 10]
2	(2, 4]	15	(10, 50]
3	(4, 10]	16	(50, 100]
4	(10, 15]	17	(100, 150]
5	(15, 25]	18	(150, 200]
6	(25, 35]	19	(200, 250]
7	(35, 45]	20	(250, 300]
8	(45, 60]		
9	(60, 80]		
10	(80, 100]		
11	(100, 120]		
12	(120, 140]		
13	(140, 165]		

the action that reduces the difference between the measured $BIS_{measured}$ and $MAP_{measured}$ and the targeted BIS_{target} and MAP_{target} values, respectively, should have a higher reward. With a well-defined reward, the agent can learn the correct sequence of propofol infusion decisions for each patient, which results in long term maintenance of the desired BIS_{target} and MAP_{target} levels. The reward function should reinforce the agent towards the optimal policy while accounting for the simultaneous control of the BIS and MAP. Thus, the reward r_{k+1} is computed by

$$r_{k+1} = \begin{cases} \dfrac{e(kT) - e((k+1)T)}{e(kT)}, & e((k+1)T) < e(kT), \\ 0, & e((k+1)T) \geq e(kT). \end{cases} \tag{18}$$

If the error at time step $k+1$ is greater than or equal to error at time step k, then we assign $r_{k+1} = 0$, which serves to penalize a "bad" control action. Note that r_{k+1} is used to update the Q table using (6). Hence, for a given state s_k, if all the actions in the action sequence $\mathcal{A}$ are executed by the agent, then the action which results in the maximum difference $e(kT) - e((k+1)T)$ will have the highest reward. This is reflected in the update of the corresponding entry of the Q table.

Reinforcement learning algorithms utilize the computational power of digital computers to execute all possible actions from each state and observe which action will drive the system closer towards the desired state. The objective is to drive the system from a given initial state $s_k \in \mathcal{S}$, $\mathcal{S} = \{1, 2, \ldots, 20\}$, to the desired state $s_k = 1$ as $k \to \infty$. A *policy* is the series of state actions that will drive the system from an initial state to the desired state. There can be many such policies for a given discrete set of states and actions. Among all possible policies, the *optimal policy* is the one which incurs a maximum reward. Thus, successful training is achieved when, for each state $s_k \in \mathcal{S}$, the agent identifies the best action a_k^* among all possible actions $a_k \in \mathcal{A}$ resulting in a maximum reward. Maximizing the reward in turn implies that the action a_k^* will drive the system closer to the desired state $s_k = 1$ as compared to all other possible actions in the given action sequence. The learned optimal policy is unique for a given set of states and action sequence [23]. A more dense discretized state-action space increases the flexibility of the agent in identifying the best action for each state. However, as the number of states and actions increase, the computational cost and algorithm convergence time also increase.

The first step in the development of the RL agent is the learning phase in which the agent learns by experimenting with the possible actions and observing the response of the simulated patient in a sequence of scenarios as illustrated in Fig. 4. A *scenario* is a sequence of state transitions from any initial state to the desired state and in each sedation scenario, the agent interacts with the simulated patient model.

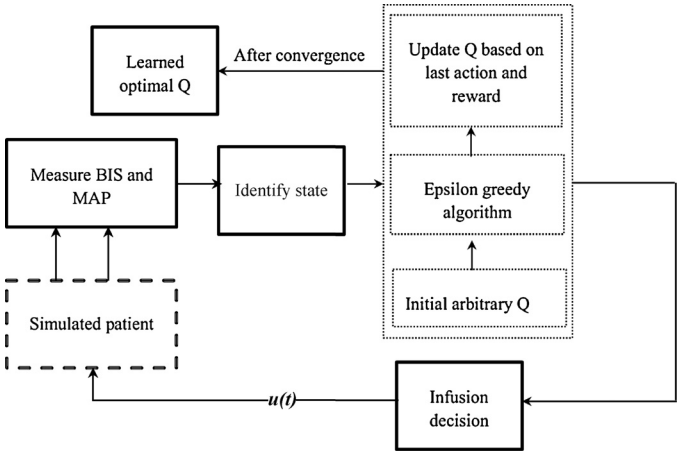


Fig. 4. Schematic of training sequence to obtain the optimal Q table.

The patient model represented by a nominal population pharmacokinetic and pharmacodynamic model is set to an arbitrary initial state and then, using the pharmacodynamic model for the BIS and MAP, the combined error $e(t), t \geq 0$, is calculated and the current state s_k is constructed. Given the current state s_k, the agent chooses a_k^*, which gives the maximum value in the Q table. However, since initially the Q table has zero entries, the agent updates the Q table for each input a_k according to the system response and reward. The response of a patient to propofol infusion depends on the pharmacokinetics and pharmacodynamics of the patient. Hence, to gather information on the patient dynamics, the agent infuses propofol at various rates and observes the response of the patient. Specifically, to associate the state with the best action, for every state, the agent executes all possible actions in the action sequence $\mathcal{A}$ randomly.

When the system is at state $s_k \in \mathcal{S}$, a "good" control action $a_k \in \mathcal{A}$ executed by the agent results in $r_{k+1} > 0$ for $e((k+1)T) < e(kT)$, and a "bad" control action $a_k \in \mathcal{A}$ results in $r_{k+1} = 0$, for $e((k+1)T) \geq e(kT)$. Since r_{k+1} is used to update the Q table using (6), the agent executes every possible control action $a_k \in \mathcal{A}$ for all $s_k \in \mathcal{S}$ to assess the effectiveness of each action in incurring a positive reward. In the terminology of [23] this is known as "exploration" of the state-action pairs. To learn an optimal policy within a given discrete set of states and actions, the agent should explore all the possible state-action pairs and exploit its knowledge regarding the previous control actions that were effective [23].

Here, we use an ϵ-greedy policy to learn the optimal policy [23]; that is, to learn from information predicated on the pharmacology of a patient, the agent executes random actions with probability ϵ, where ϵ is a small positive number. Then, the agent assess the corresponding reward to update the Q table using (18) and (6). Next, using (7) with $k \to \infty$, exploration of the Q table associates the optimal action at each state with the maximum reward, and the Q table converges to the optimal Q-function. To ensure convergence and to learn the optimal policy, the learning rate $\eta_k(s_k, a_k)$ should be reduced over time and all the state-action pairs in the Q table should be executed frequently; ideally with $k \to \infty$ [23].

2.5. Details of the simulation

In this section, we present a numerical example that illustrates the proposed RL approach for the closed-loop control of BIS and MAP. For our problem, we iterated on 50,000 (arbitrarily high) scenarios, where a scenario represents the series of transitions from an arbitrary initial state to the required final state $s_k = 1$. Furthermore, we initially assigned $\eta_k(s_k, a_k) = 0.2$ (for scenarios 1 to 499) and subsequently halved $\eta_k(s_k, a_k)$ every 500th scenario. For each scenario,

Table 2
Perturbation values.

Parameter	Perturbation range
Concentration at half maximal effect of BIS, EC_{50}	0.004 ± 0.001 [g/l]
Concentration at half maximal effect of MAP, C_{50}	0.004 ± 0.001 [g/l]
Degree of nonlinearity of BIS(c_{eff}), γ	3 ± 1
Degree of nonlinearity of MAP(c), α	3 ± 1
Time lag between $c_{eff}(t)$ and $c(t)$, a_{eff}	$\in[0.17, 1]$ [min^{-1}]
Volume of central compartment, V_c	16 ± 1 [l]
Transfer coefficients, a_{ij}	$\pm 0.5\%$ [min^{-1}]

a new set of randomized initial states $x_1(0) \in [0, 0.084]$ g, $x_2(0) \in [0, 0.067]$ g, $x_3(0) \in [0, 0.039]$ g, and $c_{eff}(0) \in [0, 0.005]$ g/L of propofol was assigned to the simulated patient model and then the learning phase was repeated until convergence and the performance goals were met; that is, keeping the BIS and the MAP values within the desired ranges. For our simulation, the Q table converged before reaching the maximum iteration. After convergence, for every state s_k, the agent chose an action $a_k = (\mathcal{A}_i)_{i \in \mathbb{I}^+}$, where $i = \text{argmax} Q(s_k, \cdot)$.

After the learning process identifies the best control policy as the learned Q table, that is, the best sequence of infusion rates required for each state to reach the desired goal, the performance of the learned agent is evaluated over individual patients in a sequence of scenarios in order to check how the agent can perform based on its optimal control policy in practical situations. The evaluation of the proposed approach is investigated in a population of 30 simulated patient models over a sequence of hypnosis scenarios lasting for 2 hours. The pharmacokinetic and the pharmacodynamic values of the simulated patients are chosen randomly from a predefined range as listed in Table 2. ICU patients often require moderate sedation, and hence, for our simulation we choose $BIS_{target} = 65$ and $MAP_{target} = 80$. During anesthesia administration oversedation and undersedation are not acceptable. Hence, after training we avoided exploration or random actions to update the Q table, but used the optimal $Q(s_k, a_k)$ discussed in the previous section for making drug infusion decisions for the 30 simulated patients.

Note that the range of $BIS_{error}(t)$ and $MAP_{error}(t)$, $t \geq 0$, are identical, namely 0–100%. Hence, we used a positive parameter β to weigh the control of BIS relative to MAP. Selecting a high value for β reduces the control over MAP, whereas selecting a small value for β will risk the regulation of BIS. Hence, for our simulation we set $\beta = 8$ by trial and error. As per ASHP guidelines [5], the recommended propofol dosage is a bolus dose of 20 mg during the initial induction period and continuous infusion in the range of 5–80 μg/kg/min during the maintenance period of anesthesia administration. During the maintenance period of anesthesia administration, the maximum requirement for a 100 kg patient is 8 mg/min. Note that we use the same RL agent during the induction period and maintenance period, and depending on the agent decision for a_k, $IR_k = a_k \times IR_{max}$. Thus, for training the RL agent we set $IR_{max} = 20$ mg/min.

To analyze the performance of the learned agent and evaluate the steady state performance of the anesthesia control, we present simulation results and statistical results using the 30 simulated patients. In order to further investigate the effect of simultaneous regulation of the BIS and MAP parameters on the sedation level (BIS) of a patient, we also present two different case studies in the results section.

3. Results

Using the learned optimal policy, the closed-loop control strategy shown in Fig. 5 is implemented. At every time step, a decision on the infusion rate is made using the learned optimal Q table. The optimal Q table is constructed using a simulated patient model characterized by a nominal pharmacokinetic and pharmacodynamic model.

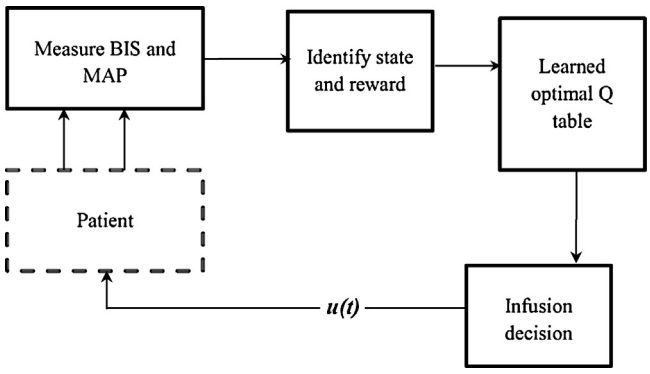


Fig. 5. RL-based optimal and robust closed-loop control of BIS and MAP.

To quantify the performance of the trained agent in the closed-loop anesthesia control, we use the median performance error (MDPE), median absolute performance error (MDAPE), and root mean square error (RMSE) [19]. The instantaneous performance error (PE) is defined as

$$\text{PE}(t) \triangleq \frac{\text{Measured Value}(t) - \text{Target Value}}{\text{Target Value}} \times 100, \tag{19}$$

where Measured Value and Target Value in (19) refer to the measured and target values of the BIS and MAP. Note that for the controlled variables BIS and MAP, the performance error is the same as the $BIS_{error}(t)$ and $MAP_{error}(t)$, $t \geq 0$, given by (16) and (17), respectively. The median performance error (MDPE) gives the control bias observed and is computed by

$$\text{MDPE}_i = \text{median}(\text{PE}_{ij}), \quad j = 1, \ldots, N, \tag{20}$$

whereas

$$\text{MDAPE}_i = \text{median}(|\text{PE}_{ij}|), \quad j = 1, \ldots, N, \tag{21}$$

and

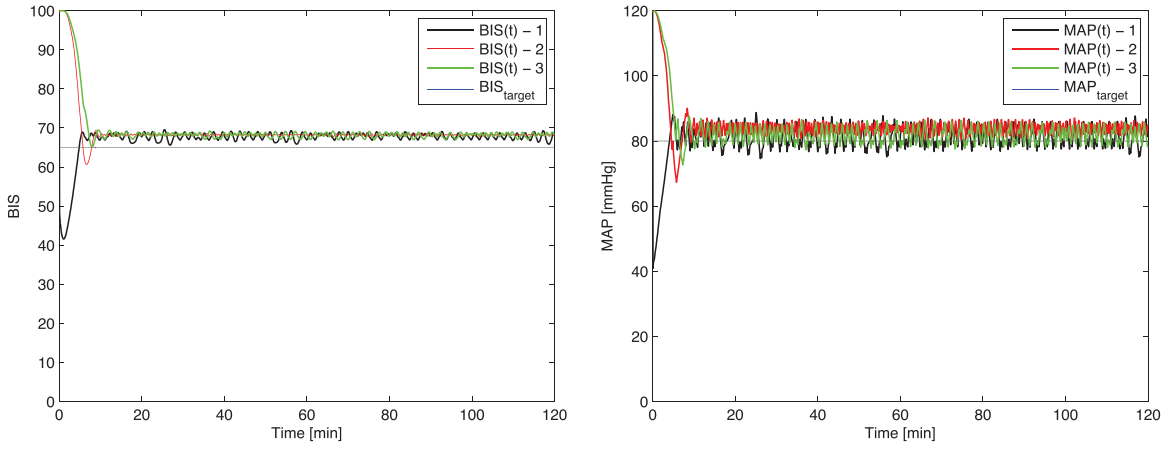
$$\text{RMSE}_i = \sqrt{\frac{\sum_{j=1}^{N} (\text{Measured Value}(t) - \text{Target Value})^2}{N}}, \tag{22}$$

where $i \in \{1, \ldots, 30\}$ represents the ith patient, j represents the set of PE measurements for an individual, $t \in [kT, (k+j-1)T]$, $j = 1, \ldots, N$, and N is the number of measurements for each patient. MDAPE_i is the median absolute performance error and reflects the size of the error and the accuracy of the agent in maintaining the control variables BIS and MAP for each patient [19]. RMSE_i represents the standard deviation between the target value and measured values of the controlled variables for each patient.

Table 3 summarizes the performance metrics for the RL agent during the hypnosis scenarios considered. In this table, the range of the values of MDPE, MDAPE, and RMSE for 30 patients are listed. Note that the amount of inaccuracy that is reflected in the value of the MDAPE metric for 30 patients are in the acceptable performance range [19]. To further analyze the performance of the proposed approach, a statistical analysis is conducted in which during the 2

Table 3
Performance metrics for control variables BIS and MAP.

Performance metrics (for 30 patients)	Controlled variables	
	BIS	MAP
MDPE [%]	3.97 ± 2.32	4.05 ± 2.50
MDAPE [%]	4.19 ± 6.43	5.31 ± 5.30
Min–Max	66.43–68.25	75.52–89.46
Interquartile range	0.55	7.16
RMSE	2.12–3.30	2.30–9.50



(a) BIS index versus time for $BIS_{target} = 65$.

(b) MAP versus time for $MAP_{target} = 80$.

Fig. 6. Simulation results for three patients chosen randomly from the test set of 30 patients.

hour propofol infusion considered in our simulation, we analyzed the central tendency and range of measured variables for the 30 simulated patients for each controlled variable. In particular, we address the amount of time that the response is within a defined band of the target value, that is, ±5, and the percentage of all of the patients for which the response is within a defined band. During the 2 hour propofol infusion considered in our simulation, the measured value of the BIS is within ±5 of BIS_{target} for 90.41% of the time for all of the 30 simulated patients. The measured value of MAP is within ±5 of MAP_{target} for 76.65% of the time for 60% of all of the patients.

For our calculations, we considered the time range $t \in [0, 10)$ as induction period and $t \in [10, 120]$ as maintenance period of anesthesia administration. Furthermore, we have listed the minimum and maximum values of the controlled variables during the maintenance period of anesthesia administration in Table 3. Note that the listed minimum and maximum values are the mean of the minimum and maximum values for the 30 simulated patients. Table 3 also gives the interquartile range (IQR) to show the midspread (variability) of the controlled variables. Interquartile range is the range of the middle 50% of a sorted data set. To calculate the interquartile range, we used the mean value of each controlled variable over the range $t \in [10, 120]$ for the 30 simulated patients. Note that the interquartile range of the controlled variables BIS and MAP is 0.55 and 7.16, respectively, which shows that the BIS has less variability than the MAP of the simulated patients.

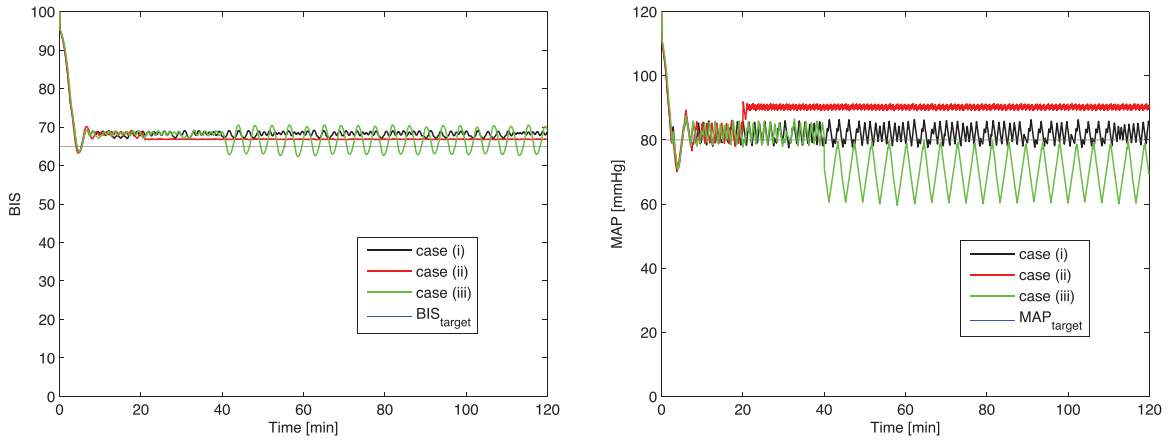
To further elucidate these variations around the target MAP_{target} and BIS_{target} values the RL-based, closed-loop anesthesia scenario for three simulated patients chosen randomly from the set of the 30 simulated patients is plotted in Fig. 6. Note that Patient 1 is assigned a nonzero initial condition to address the case of post operative patients who require prolonged moderate sedation while in the ICU. Such patients may have some quantity of sedative drugs in their body due to deep sedation during surgery. Patients 2 and 3 are assumed to have a zero initial concentration of propofol in their body. It can be seen that the RL agent is able to maintain the BIS value and MAP value around the target values. It is clear from our simulations that the trained RL agent demonstrates acceptable performance for the simultaneous control of BIS and MAP [19]. The performance evaluation measures given in Table 3 and the plots shown in Fig. 6 illustrate the significance of the β parameter in (15) for weighing the control of BIS relative to MAP.

Our simulations demonstrate comparable performance with the recent clinical trial conducted on 15 human volunteers for the performance evaluation of RL-based, closed-loop control of intraoperative anesthesia administration [19]. In this clinical trial, the range of the percentage MDPE value is -2.8 to 8.8, the range of the percentage MDAPE value is 3.4–9.6, and the range of the RMSE value is 3.3–6.5 for the 15 patients. These results are comparable with our results for the 30 simulated patients generated randomly with the pharmacological parameters given in Table 3. In addition to closed-loop control of the BIS, we developed a methodology for regulating MAP. It should be noted that the RL agent demonstrates optimal and robust performance without relying on a system model (see Fig. 6 and Table 3). This is very important for active control of complex uncertain biological systems where system modeling can be very challenging.

Furthermore, adding an additional parameter can, in general, affect the control of the primary variable (BIS). However, in the perspective of overall patient safety, instead of strictly controlling sedation alone, a balanced management of sedation along with other vital patient parameters such as hemodynamics, respiratory, pain, and relaxation is preferred. Hemodynamic and respiratory parameters such as cardiac output, heart rate variability, mean arterial pressure, and respiratory rate are some of the factors that are affected by propofol infusion. We choose MAP as the secondary control variable because propofol infusion depresses the sympathetic tone and results in venodilation. The increase in the venous capacitance leads to a decrease in the cardiac preload resulting in depressed cardiac output and MAP [27].

In order to further investigate the effect of simultaneous regulation of the BIS and MAP parameters on the sedation level (BIS) of a patient, two different case studies are presented. In the first case study, a hemodynamic disturbance is considered in which the MAP is altered by d units. This case study considers the effect of other factors such as hemorrhage on MAP as an exogenous disturbance. In the second case study, the MAP is set to a constant value irrespective of propofol infusion, which corresponds to patients that remain intubated in the ICU with post-aortic aneurysm repair or septic patients with respiratory failure. In this case, MAP is independent from sedation.

For the first case study, in order to show the performance of the controller in the presence of a hemodynamic disturbance that alters the MAP by d units, we simulated a patient chosen randomly from the test set of 30 patients using the following three scenarios for MAP: (*i*) MAP(t), $t \geq 0$, (*ii*) MAP(t) + d, $t > 20$, and (*iii*) MAP(t) − d, $t > 40$. Fig. 7 shows the simulation results for $MAP_{target} = 80$ and $d = 10$. Note that the control of the BIS is weighed more heavily



(a) BIS index versus time for $BIS_{target} = 65$. (b) MAP versus time for $MAP_{target} = 80$.

Fig. 7. Simulation results for a patient chosen randomly from the test set of 30 patients; case (*i*), MAP(*t*), $t \geq 0$, case (*ii*), MAP(*t*) + *d*, $t > 20$, and case (*iii*), MAP(*t*) − *d*, $t > 40$, where $d = 10$ units represents a disturbance in the hemodynamic system of patient.

Table 4
Performance metrics for the control variable BIS by keeping MAP constant; for 30 simulated patients.

Performance metrics	MAP(*t*) = 120, $t \geq 0$	MAP(*t*) = 100, $t \geq 0$	MAP(*t*) = 60, $t \geq 0$	MAP(*t*) = 40, $t \geq 0$
MDPE [%]	9.43 ± 0.63	-0.84 ± 0.45	-0.87 ± 0.49	9.44 ± 0.62
MDAPE [%]	9.43 ± 0.95	1.56 ± 0.43	1.58 ± 0.45	9.44 ± 0.56
Min–Max	69.18–74.28	62.51–66.50	62.48–66.64	69.14–74.31
Interquartile range	2.6	2.15	2.16	2.67

by setting $\beta = 8$. For each scenario, the mean values for each controlled variable is calculated over the interval $t \in [10, 120]$, $t \in [20, 120]$, and $t \in [40, 120]$ as (*i*) BIS = 68.18, MAP = 81.82, (*ii*) BIS = 66.84, MAP = 90.17, and (*iii*) BIS = 66.86, MAP = 69.85, respectively. Note that due to the disturbance *d*, MAP_{error} contributes more to the error signal $e(t)$, $t \geq 0$, and hence, compared to the scenario (*i*), the patient is sedated slightly more in the scenarios (*ii*) and (*iii*). Fig. 7 shows that even though propofol infusion is adjusted to account for the increase in error due to the MAP value, the BIS value of the patient is still within ±5 units of the BIS_{target}, which is acceptable [19].

For the second case study, the controller is tested on 30 simulated patients with $MAP_{target} = 80$ and $BIS_{target} = 65$ for the following four scenarios: (*i*) MAP(*t*) = 120, $t \geq 0$, (*ii*) MAP(*t*) = 100, $t \geq 0$, (*iii*) MAP(*t*) = 60, $t \geq 0$, and (*iv*) MAP(*t*) = 40, $t \geq 0$. In all the four scenarios, the effect of propofol infusion on the hemodynamic parameter MAP is not considered and MAP(*t*), $t \geq 0$, is set to a constant value irrespective of propofol infusion. Table 4 shows the performance metrics for the BIS controlled variable for the 30 simulated patients. It is clear from Table 4 that, for the scenarios where MAP(*t*) = 100, $t \geq 0$, and MAP(*t*) = 60, $t \geq 0$, the BIS value of the patient is still within ±5 units of the BIS_{target}, which is acceptable [19]. However, for the extreme scenarios where MAP(*t*) = 120, $t \geq 0$, and MAP(*t*) = 40, $t \geq 0$, the value of the BIS is within ±10 units of the BIS_{target}. In order to achieve better performance in BIS regulation for these extreme scenarios, an additional RL agent which is trained mainly for the regulation of the BIS can be used. That is using a RL agent which is trained by choosing a very large value of β.

4. Discussion

During real time implementation, the *Q* table may require updating if the parameters of the patient under treatment are significantly different from the nominal patient model used for training. This can be achieved by updating the *Q* table and occasionally avoiding the best action and employing random actions. However, for the case of anesthesia administration, this can only be done by ensuring patient safety and requires further analysis. Batch mode RL-based algorithms that require less interactions with the system to derive a good policy can be considered. However, these algorithms require the agent to store state transition experiences. Moreover, convergence of the algorithm and distance of the solution to the true optimal solution depends on the characteristics of the function approximators involved [28]. Further research is required to assess useful function approximators for anesthesia control application.

In general, RL offers an ideal framework for on-line identification. This is demonstrated in several real time applications such as robotic control, helicopter control, and wind turbine control [14,15]. However, in the case of clinical applications, decision making based on online identification requires caution. The present work can be considered as a preliminary study towards the development of patient safe RL-based closed-loop control of anesthesia. Note that any change in the patient model is reflected in the pharmacologic response of the patient, and hence, in the error signal $e(t)$, $t \geq 0$, and the state s_k. As the controller executes decisions based on the state s_k, small changes in the patient model such as those caused by habituation to long term infused anesthetic drug, can be addressed to a certain extent. However, if the habituation or any other clinical situation causes large and nonlinear changes in the patient model, then adaptive decision making is required.

The proposed controller is intended for use in ICU sedation. Hence, this is one limitation of the proposed method. There is a time delay between the time of action a_k and time for the corresponding pharmacologic response in the patient. We envisage to improve the proposed controller by accounting for the initial drug mixing delay, which can be significant during the induction period, in our future work. Another method to account for drug mixing delays is to incorporate an additional RL agent in the closed-loop system, which is trained for regulation in the face of large errors that may arise during the initial mixing or online identification periods.

Moreover, it should be emphasized that the propofol dosage required to achieve a desired sedation level varies with the age, gender, height, and weight of the patient. Such patient characteristics are reflected in the pharmacokinetic and pharmacodynamic model parameters EC_{50}, C_{50}, γ, α, a_{eff}, V_c, and a_{ij}. For training the RL agent, we set $IR_{\max} = 20$ mg/min. In the proposed controller, the infusion rate is given by $IR_k = a_k \times IR_{\max}$. Note that any change in the patient model is reflected in the pharmacologic response of the patient, and hence, in the error signal $e(t)$, $t \geq 0$, and the state s_k. As the controller executes decisions based on the state s_k, small changes in the patient model, such as those caused by variation in the patient characteristics, are taken into account by the controller to give a patient specific IR_k. See Table 2 for the range of the patient characteristics used to generate the 30 simulated patients (adults).

Figs. 6 and 7, and Tables 3 and 4 show that the RL agent trained by setting $IR_{\max} = 20$ mg/min exhibits acceptable performance for the 30 simulated patients with different patient characteristics. However, it should be mentioned that the patient characteristics will significantly vary between the patient groups; for example, infants, children, adults, and obese patients. According to the pharmacokinetics and pharmacodynamics of the target patient population, the value of $IR_{\max}$ should be fixed while training the RL agent. Thus, in the case of patients with significantly different pharmacokinetics and pharmacodynamics, the required drug dosage varies significantly, and hence, the RL agent needs to be trained with a different $IR_{\max}$. Alternatively, to account for the vivid patient characteristics of different patient groups, a bank of RL agents can be developed in which each RL agent is trained by using appropriate $IR_{\max}$.

Training the Q table separately using a reward function that penalizes the variation from the target $\text{BIS}_{\text{target}}$ and $\text{MAP}_{\text{target}}$ values, and using denser discretization around the optimal infusion rate derived for $s_k = 1$, will remove the offset seen in Fig. 6 and allow for better regulation of the BIS and MAP around the target values. Another limitation of this study is the choice of the discrete action space and state space. A further refined discretization of these spaces or, ideally, a continuous-time action space and state space, will potentially allow for a more robust adaptation of the RL agent resulting in a more patient specific optimal policy. Hence, an important issue for future research is the development of RL-based control techniques in the continuous-time domain. The performance of the proposed RL algorithm can be further improved by adjusting the learning rate $\eta(s_k, a_k)$ and discount factor θ. Choosing a better reward function, which can result in a faster convergence rate [29], as well as refining the state space and action space discretization can further refine the optimal RL agent. Since anesthesia administration is a life critical task, more in silico and clinical trials need to be performed for further validating the RL-based, closed-loop control of anesthesia administration.

5. Conclusions

In this paper, a reinforcement learning-based approach for the simultaneous control of sedation and hemodynamic parameter management is proposed using the regulation of the anesthetic drug propofol. Simulation results using 30 patient models with varying pharmacokinetic and pharmacodynamic parameters show that the proposed RL control strategy is promising in designing closed-loop controllers for ICU sedation to regulate sedation and hemodynamic parameters simultaneously. Furthermore, our simulations show that the RL-based, closed-loop control is robust to system uncertainties. With additional experiments to further refine and validate the optimal RL agent, and accounting for other vital parameters such as respiratory rate and heart rate variability, this method can prove promising in automating anesthesia administration in the ICU.

Acknowledgements

This publication was made possible by NPRP grant No. 4-187-2-060 from the Qatar National Research Fund (a member of the Qatar Foundation). The statements made herein are solely the responsibility of the authors.

References

[1] W.M. Haddad, J.M. Bailey, B. Gholami, A.R. Tannenbaum, Clinical decision support and closed-loop control for intensive care unit sedation, Asian J. Control 15 (2) (2013) 317–339.
[2] A.R. Absalom, R.D. Keyser, M.M.R.F. Struys, Closed-loop anesthesia: are we getting close to finding the holy grail? Anesth. Analg. 112 (3) (2011) 516–518.
[3] K. Soltesz, J.O. Hahn, T. Hagglund, G.A. Dumont, J.M. Ansermino, Individualized closed-loop control of propofol anesthesia: a preliminary study, Biomed. Signal Process. Control 8 (6) (2013) 500–508.
[4] J.O. Hahn, G.A. Dumont, J.M. Ansermino, Robust closed-loop control of hypnosis with propofol using WAVcns index as the controlled variable, Biomed. Signal Process. Control 7 (5) (2012) 517–524.
[5] J. Jacobi, G.L. Fraser, D.B. Coursin, R.R. Riker, D. Fontaine, E.T. Wittbrodt, D.B. Chalfin, M.F. Masica, H.S. Bjerke, W.M. Coplin, D.W. Crippen, B.D. Fuchs, R.M. Kelleher, P.E. Marik, S.A. Nasraway, M.J. Murray, W.T. Peruzzi, P.D. Lumb, Clinical practice guidelines for the sustained use of sedatives and analgesics in the critically ill adult, ASHP Therapeutic Guidelines, Am. J. Health. Syst. Pharm. 59 (2002) 150–178.
[6] G. Werrett, Sedation in intensive care patients Update in Anaesthesia, vol. 16, 2003, pp. 1–5.
[7] J.M. Bailey, W.M. Haddad, Drug dosing control in clinical pharmacology, IEEE Control Syst. Mag. 23 (2) (2005) 35–51.
[8] E. Furutani, K. Tsuruoka, S. Kusudo, A hypnosis and analgesia control system using a model predictive controller in total intravenous anesthesia during day-case surgery, in: SICE Annual Conference, Taipei, Taiwan, 2010 August, pp. 223–226.
[9] B. Gholami, W.M. Haddad, J.M. Bailey, A.R. Tannenbaum, Optimal drug dosing control for intensive care unit sedation using a hybrid deterministic-stochastic pharmacokinetic and pharmacodynamic model, Optim. Control Appl. Methods 34 (2013) 547–561.
[10] R. Padmanabhan, N. Meskin, W.M. Haddad, Optimal control of midazolam infusion for post operative patients in intensive care units, Asian J. Control 17 (3) (2015) 1–12.
[11] W.M. Haddad, T. Hayakawa, J.M. Bailey, Adaptive control for nonnegative and compartmental dynamical systems with applications to general anesthesia, Int. J. Adapt. Control Signal Process. 17 (2003) 209–235.
[12] R. Padmanabhan, N. Meskin, W.M. Haddad, Direct adaptive disturbance rejection control for sedation and analgesia, in: Middle East Conference on Biomedical Engineering, Doha, Qatar, 2014 February, pp. 175–179.
[13] D. Vrabie, K.G. Vamvoudakis, F.L. Lewis, Optimal Adaptive Control and Differential Games by Reinforcement Learning Principle, Institution of Engineering and Technology, London, UK, 2013.
[14] M. Sedighizadeh, A. Rezazadeh, Adaptive PID controller based on reinforcement learning for wind turbine control World Academy of Science, Engineering and Technology, vol. 2, 2008, pp. 01–23.
[15] P. Abbeel, A. Coates, M. Quigley, A.Y. Ng, An application of reinforcement learning to aerobatic helicopter flight, Neural Inf. Process. Syst. 19 (2007) 1–8.
[16] J. Martin-Guerrero, F. Gomez, E. Soria-Olivas, J. Schmidhuber, M. Climente-Marti, N. Jemenez-Torres, A reinforcement learning approach for individualizing erythropoietin dosages in hemodialysis patients, Exp. Syst. Appl. 36 (2009) 9737–9742.
[17] B.L. Moore, P. Panousis, V. Kulkarni, L.D. Pyeatt, A.G. Doufas, Reinforcement learning for closed-loop propofol anesthesia, in: Proc. 22th Annual Conf. on Innovative Applications of Artificial Intelligence, Atlanta, Georgia, USA, 2010 July, pp. 1807–1813.
[18] J.W. Johansen, P.S. Sebel, T. Smet, M.M. Struys, Development and clinical application of electroencephalographic bispectrum monitoring, Anesthesiology 93 (2000) 1336–1344.
[19] B.L. Moore, L.D. Pyeatt, V. Kulkarni, Panousis P. Kevin, A.G. Doufas, Reinforcement learning for closed-loop propofol anesthesia: a study in human volunteers, J. Mach. Learn. Res. 15 (2014) 655–696.
[20] S.Z. Fan, Q. Wei, P.F. Shi, Y.J. Chen, Q. Liu, J.S. Shieh, A comparison of patient's heart rate variability and blood flow variability during surgery based on the Hilbert Huang transform, Biomed. Signal Process. Control 7 (5) (2012) 465–473.
[21] W.M. Haddad, V. Chellaboina, Q. Hui, Nonnegative and Compartmental Dynamical Systems, Princeton, NJ, Princeton University Press, 2010.
[22] R.R. Rao, B.W. Bequette, Simultaneous regulation of hemodynamic and anesthetic states: a simulation study, Ann. Biomed. Eng. 28 (1) (2000) 71–84.
[23] R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction, MIT Press, Cambridge, MA, 1998.
[24] C.J.C.H. Watkins, P. Dayan, Q-learning, Mach. Learn. J. 8 (3) (1992) 279–292.

[25] D.P. Bertsekas, J.N. Tsitsiklis, Neuro-Dynamic Programming, Athena Scientific, MA, 1996.
[26] A.R. Absalom, V. Mani, T. Smet, M.M. Struys, Pharmacokinetic models for propofol defining and illuminating the devil in the detail, Br. J. Anaesth. 103 (1) (2009) 26–37.
[27] B. Robinson, T.J. Ebert, T.J.O. Brien, M.D. Colinco, M. Muzi, Mechanisms whereby propofol mediates peripheral vasodilation in humans: sympathoinhibition or direct vascular relaxation? Anesthesiology 86 (1997) 64–72.
[28] R. Fonteneau, S.A. Murphy, L. Wehenkel, D. Ernst, Batch mode reinforcement learning based on the synthesis of artificial trajectories, Ann. Oper. Res. 208 (1) (2013) 383–416.
[29] L. Matignon, G.J. Laurent, N.L. Fort-Piat, Reward function and initial values: better choices for accelerated goal-directed reinforcement learning, in: 16th International Conference on Artificial Neural Networks, ICANN'06, Athens, Greece, 2006 February, pp. 840–849.