ECAI 2012
Luc De Raedt et al. (Eds.)

doi:10.3233/978-1-61499-098-7-949

Partially Observable Markov Decision Process for Closed-Loop Anesthesia Control

Eddy C. Borera [1] and **Brett L. Moore** and **Larry D. Pyeatt**

Abstract. Recently, researchers have favored computer-automated drug delivery system to reduce the risks of intraoperative awareness and postoperative morbidity, and their use is associated with a number of favorable patient outcomes. For example, Proportional-Integral-Derivative, *maximum a posteriori* (MAP) Bayesian approaches, fuzzy logic, and reinforcement learning, have been developed and applied successfully in simulated patients or volunteers. Despite the successes, variations of errors in the observed data are known to affect the performances of the controllers, especially when a patient state estimation is required. To have a better controller, we apply Partially Observable Markov Decision Process (POMDP) to achieve better drug delivery policy, even when there is incomplete information about patients' current states during operations. In this paper, a POMDP model for closed-loop anesthesia control is introduced. Then, a state-of-the-art POMDP solver is used to compute a good control policy, in other words, propofol rates to administer to a patient, in efforts to reduce the risk of intraoperative awareness and postoperative side effects in patients.

1 INTRODUCTION

Automated controllers have been applied in anesthesia control with great successes, both in simulations and on volunteers. For instance, Absalom et al. [1] have proposed and applied Proportional-Integral-Derivative(PID) controllers successfully to patients undergoing general anesthesia. Also, many other approaches have been applied in computer-automated control systems; and these include: fuzzy logic, stochastic control, dynamic programming, *maximum a posteriori* (MAP) Bayesian techniques, etc. Recently, a reinforcement learning (RL) controller has been successfully implemented and tested on human volunteers [15].

Our work in closed-loop anesthesia control uses the bispectral index of the electroencephalogram (EEG), or BIS (Aspect Medical Systems, Newton, MA). Currently, BIS enjoys the greatest clinical acceptance as a measure of hypnotic effect. BIS, measured as a single value that lies in the range [0, 100], is a statistically derived indicator of cortical activity [17]. BIS values near 100 are associated with normal wakefulness; values near zero correlate to iso-electric brain states.

1.1 Propofol-Induced Hypnosis

Propofol is a short-acting sedative agent administered intravenously to achieve induction and maintenance of general anesthesia in the operating room and other critical care arenas. Propofol suppresses cortical brain function, yielding *hypnosis*, but offers no analgesic effect (pain relief). The anesthesia community has studied automated delivery of propofol for two principal reasons. First, the short-acting nature of the drug, characterized by rapid onset and recovery, permits titration to desired effect. Second, indication of propofol effect may be observed in the EEG [9].

2 Motivations for computer-automated controllers

Previously, propofol has been administered to patients manually. In this case, anesthesiologists repetitively evaluate patient's state before injecting propofol to reach a desired set point value. Accuracy of drug infusion is preferred to avoid underdosing and overdosing patients, which may cause intraoperative awareness and postoperative side effects respectively. Recently, researchers have proposed computer-automated controller to assist anesthesiologists. The ultimate goal is to have a good and accurate controller which is tailored to any patient undergoing anesthesia control process.

Existing controllers are mostly developed for population-based models, which make decisions based on results from the PK/PD models of choices. Intra-variability in patients challenge these controllers, and good performance is only guaranteed for ideal patients that have the same characteristics of the patients used during the PK/PD studies. Despite the limitations, automated-controller have delivered successes both in simulations and clinical trials.

3 Challenges in anesthesia control

The anesthesia process is synonymous to modeling consciousness, which is a very complex task. Absalom et al. [3] mentioned some differences between anesthesia control and aviation control. For example, in aviation control outputs, which consist of angle, velocity and pitch, can be measured accurately. Also, the relationship between inputs and outputs is predictable, well-defined, and linear [3]. However, this not the case in anesthesia control since the input-output relationship is non-linear.

Currently, EEG is widely used to estimate patient's brain activities, which have been broadly accepted to be associated with patient's level of consciousness. BIS has been used as the *de facto* measure of level of consciousness in anesthesia control field. This value is computed from histories of EEG signals. The main challenge on relying with BIS is that the EEG signals are known to exhibit some noises, which complicate drug-effect estimations and the overall drug infusion policy.

Absalom et al. [3] also mentioned the asymmetrical process in drug administration because drugs infused to patients cannot be removed. Also, PK/PD models are developed and tailored for patients that share similar characteristics than the subjects used during their

[1] Texas Tech University, U.S.A, email: eddy.borera@ttu.edu

studies. Therefore, their accuracy is limited and vary according to patient's response to the drugs. In addition, unknown parameters may affect drug delivery effects to patients.

Despite the development, variations, and successes of PK/PD models in drug control, further studies are still needed to improve and evaluate their performance in more challenging situations. Choice of PK/PD models for any computer-automated controllers is still controversial [3] since they offer different population-based parameters that needed to be tailored for a specific patient undergoing general anesthesia.

4 BACKGROUND

4.1 Pharmacokinetics

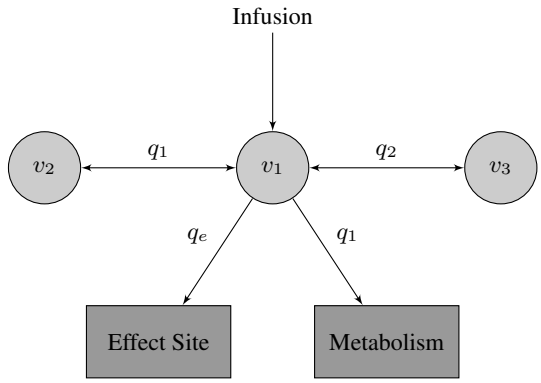

Figure 1: Three-compartment pharmacokinetic model. The central, slow, and rapid compartments are represented by their volumes v_1, v_2, and v_3 respectively.

A pharmacokinetic (PK) model describes the drug concentration time course in a patient [19], which can be represented by a n-compartment mammillary model. Multiple PK models have been developed for different populations. One of the most widely used models is the Schnider PK model, which is characterized by the central, slow, rapid, and effect site compartments.

First, as illustrated in Figure 1, drug is infused into the central compartment v_1, which is effectively the volume of blood within the patient's circulatory system [15]. Then, some concentration gradients govern the subsequent transport of the drug concentrations to the slow compartment v_2 and the rapid compartment v_3, which represent the less and highly perfused organs respectively. The effect-site compartment models the delayed drug effects for the blood-brain interaction [15]. For the rest of the paper, we use v_e to denote the volume of the effect site compartment, which is used to compute the drug concentration of the compartment.

Given a propofol infusion I $(\mu g/min)$, the drug concentrations in all four compartments are represented by the following differential equations:

$$\frac{\delta\psi_1}{\delta dt} = \frac{1}{v_1}\left[I - (q_1+q_2+q_3)\,\psi_1 + q_2\psi_2 + q_3\psi_3\right] \quad (1)$$

$$\frac{\delta\psi_2}{\delta dt} = \frac{q_2}{v_2}(\psi_1 - \psi_2) \quad (2)$$

$$\frac{\delta\psi_3}{\delta dt} = \frac{q_3}{v_3}(\psi_1 - \psi_3) \quad (3)$$

$$\frac{\delta\psi_e}{\delta dt} = \frac{q_e}{v_e}(\psi_1 - \psi_e), \quad (4)$$

where v_i (ml), q_i (ml/min), and ψ_i $(\mu g/ml)$ represent the volume, clearance, and drug concentrations of the i^{th} compartment respectively. Similarly for the effect-site compartment, these three parameters are denoted by v_e, q_e, and ψ_e respectively.

In the Schnider model, the volume and clearance parameters were studied and derived from 24 volunteers (11 females, 13 males; weight range 44–123 kg; age range 25–81 year; height range 155–196 cm) [2]. The Marsh PK model [14] is also well-known in the literature. It is characterized by 3-compartment mammillary model, where its parameters were derived from children. More PK models are presented in [21, 12]. The superiority of a specific model is still debatable. However, some researchers favor the Schnider model over the Marsh Model as it takes patients' ages into consideration to compute for the parameters. Also, it targets the effect-site compartment in contrasts to the Marsh model, which target plasma compartments [2].

4.2 Pharmacodynamics

A pharmacodynamic (PD) model is used to determine the effects of drugs on a patient. It characterizes the relationship between drug concentrations and their effects to a subject. Researchers have discovered that they are non-linear and challenging to model. As in pharmacokinetic, a pharmacodynamic model is usually developed from a set of ideal patients which may consist of female, male, children, elderly, etc. Then, the resulting model parameters are applied to estimate the effects of drug concentrations in a random patient. Pharmacodynamics are modeled using the effect site compartment as shown in Figure 1 combined with a non-linear function that defines relations between the effect site concentration and drug effect in human brains [11]

Various PD models have been used in drug infusion control to estimate drug effects on patients' brains. In this study, we use the Doufas' PD model [8], which is a 3-layer artificial neural network that was trained on healthy patients to compute for the appropriate network layer connection weights. This artificial neural network model approximates the non-linear function that characterizes the relationship between drug concentrations and their effects on patients.

The collective application of a PK and a PD model (PK/PD) allows us to estimate a general, population-based BIS response to propofol infusion. In this case, the PK model estimates drug concentration in different compartments. Then, the associated PD model calculates the estimated drug-concentration effects in the effect site.

5 RELATED WORK

One of the main reasons to consider POMDP in this application is that a BIS value observed from a patient is known to exhibits some noise [24]. Therefore, it is not surprising that determining the exact patients' states throughout a surgical operation is a daunting task for any controllers — humans or computers. The best they can do, in such case, is to apply a reliable filtering technique to minimize error variances. Despite their successes in other applications, they have limitations. Previously, a Kalman filter was introduced as a filtering technique in propofol hypnosis control [19]. It is a well known state estimation method, but the noise models are assumed to be well defined; otherwise, error variances can affect the filtering, smoothing, or predicting processes. These errors, however, are patient-specific since patients' responses to propofol rely on various factors (known and unknown). Many other filtering approaches have since been proposed: exponential, adaptive neural network [10, 5], Bayesian filtering techniques [6, 7], etc.

To deal with these problems, a POMDP model, which does not use any filtering techniques, is introduced. POMDP is a framework for planning in partially observable environments, where observed data exhibit some uncertainties. Therefore, the proposed POMDP controller does not require a filtering process, which are necessary in other computer-automated controllers. In contrast, this new controller computes a control policy using the noisy measurements.

5.1 Previous Stochastic Controllers

Our work follows from previous approaches by Hu et al. [11], where they proposed a 3-compartment stochastic PK/PD control model. They introduce this new approach to improve previous controllers that separate state estimation and the control process , where they tend to ignore variability when computing for policy [11]. To avoid the assumption that patient's state is known with certainty, Hu et al. suggested a more robust technique that consider uncertainties in patient's PK/PD parameters. The patient's state was represented as $s = (m_1, \cdots, m_k)$. Due to the curse of dimensionality that plague stochastic controllers, they performed a 3-point discretization on each parameter of interest in s. For example, for $i \in [1, k]$ and a deviation σ, m_i can be discretized into m_i, $m_i - \sigma$, or $m_i + \sigma$, which reduces the number of all possible states to 3^k. This process makes value function computations feasible.

Our approach avoids this discretization problem by sampling important sates from a closed-form of the state space. We use a Monte Carlo method to approximate state value functions, and only important states that can be visited during simulations are considered. Also, we chose the POMCP planner by Silver [22] because it does not require a full computation of the belief update, which is computationally expensive for a problem with very large number of states as the one we are trying to solve. Instead, it estimates $b(s)$ using particle filters. Also, it has been claimed to be able to solve problems up to 10^{56} number of states [22]. More information about POMCP can be seen in [22].

6 TECHNIQUES

6.1 Partially Observable Markov Decision Process

A Partially Observable Markov Decision Process (POMDP) is a framework that models interactions between an agent and a stochastic partially observable environment [23]. It can be denoted as a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{Z}, \mathcal{T}, \mathcal{O}, \mathcal{R}, \gamma,)$, where $\mathcal{S}$, $\mathcal{A}$, and $\mathcal{Z}$ represent the set of all possible states, actions, and observations respectively. At every time step, the agent resides in a state $s \in \mathcal{S}$, which is partially observable, and performs an action $a \in \mathcal{A}$ to receive an expected reward $r(b, a) = \sum_s \mathcal{R}(s, a)b(s)$, where b is a probability distribution over the set of states. The agent, then, moves to a state s', where s' can be the same as the previous state s. The transitional probability $\mathcal{T}(s', s, a) = Pr(s' \mid s, a)$ determines the results of a stochastic action a in state s. Finally, upon reaching a new state, the agent perceives an observation $z \in \mathcal{Z}$, where $\mathcal{O}(z, s', a) = Pr(z \mid s', a)$ is the probability of observing z in the state s' after taking action a in state s. Given the agent's incomplete information about its current state, it maintains a *belief*: a probability distribution over all states. Suppose the agent's current belief state is denoted by $b(s)$, where $s \in \mathcal{S}$. Then, after taking an action a and receiving an observation z, the agent's new belief to be in state s' can be computed as follows:

$$\begin{aligned} b'(s') &= \tau(b, a, z)(s) \\ &= \eta\mathcal{O}(z, s', a) \sum_{s \in \mathcal{S}} \mathcal{T}(s', s, a)b(s) \\ &= \eta Pr(z|s', a) \sum_{s \in \mathcal{S}} Pr(s'|s, a)b(s), \end{aligned} \tag{5}$$

where η is the normalizing factor. Due to the *curse of dimensionality*, computing and updating the agent's belief is only feasible for problems with very small number of states, and this process is known to be $O(|\mathcal{Z}||\mathcal{S}|^2)$. Other factors that can affect algorithms' performance include: problem description, data structure, etc. To have better performances, some solvers use factored belief state, action, or observation representations to approximate the value of the agent's belief. In POMDP, *history* represents the sequence of actions performed and observations received overtime. For example, a history $h_t = (a_o z_o, a_1 z_1, ..., a_t z_t)$ list in details the action and observation pairs taken and perceived respectively up to time step t.

6.2 Online Planner

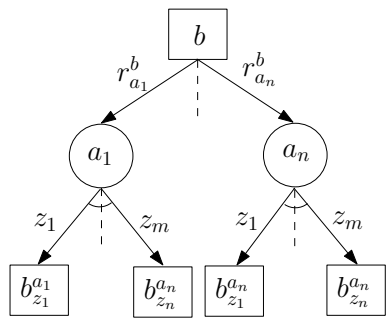

Figure 2: This figure illustrates an AND-OR tree, where beliefs are represented as OR-nodes and actions as AND-nodes. From b, the agent chooses one action, a_1 for example, and receives a reward $r^b_{a_1}$. Then from a_1, it considers all possible observations z_k, where $1 \leq k \leq |\mathcal{Z}|$, which give new sets of possible belief states $b^{a_1}_{z_k}$.

Compared to offline techniques, online algorithms combine both planning and policy execution at each time step. First, the agent determines its current belief b_t, which is updated from b_{t-1}. Then, it computes a local policy by performing value iterations, policy iterations, or simulation based techniques to determine the optimal or near optimal action to execute in b_t. Some online approaches construct AND-OR trees where, as illustrated in Figure 2, the AND and OR nodes are represented by the agents' actions and belief states respectively. In Figure 2, the agent's current belief b_t serves as the root node of the tree, and the outgoing edges represent the choices it can make by performing any possible actions. Then, the actions would lead the agent to consider all possible observations, as they will yield a new set of reachable belief states. In this case, $V(b)$, the value of a belief state b, is computed while traversing and constructing the tree. In some cases, this value is denoted differently as $V(\tau(b_t, a, z))$, which basically means a value of new belief b_{t+1} after taking action a and observing z i.e. $b_{t+1} = \tau(b_t, a, z)$.

Let $Q(b, a)$ represents the value of taking an action a in some belief state b; therefore, the value of b can be computed as follows:

$$V(b) = \max_a Q(b, a) \tag{6}$$

where

$$Q(b,a) \quad = \quad r(b,a) + \gamma Pr(z|b,a)V(\tau(b,a,z)) \tag{7}$$

These techniques mainly differ on how to expand the search tree. Monte Carlo Rollout methods, for example, perform a certain number of simulations, then average the returned rewards in order to approximate the Q-action values. The value of the current belief state is nothing but the maximum average returned value by the simulations. Throughout the simulation, the agent can choose the action to be executed at every belief node randomly or according to a predefined policy π. Similarly, the observations are sampled according to the observation probability distribution $Pr(z|s',a)$. In this case, the value of the current belief state b is estimated as follows:

$$\tilde{V}(b) = \frac{1}{K}\sum_{i=1}^{K} R(Sim_i) \tag{8}$$

, where $R(Sim_i)$ is the reward returned by the i^{th} simulation and K is the total number of simulations. The accuracy of $\tilde{V}(b)$, of course, depends on the number of simulations and the method of choosing the actions during the exploration. Also, using an offline policy, in this case, can also improve the policy quality, especially if applied with enough number of simulations.

Other methods that maintain lower and upper bounds values in order to prune *non-optimal* actions are discussed in [18, 4].

6.3 Partially Observable Monte Carlo Planning

A partially observable Monte Carlo Planning (POMCP) is based on a partially observable version of UCT (PO-UCT), which is known as an Upper Confidence Bound technique for the Bandit problems, and it uses AND-OR trees [13] (Figure 2) to approximate value functions. It constructs a Monte Carlo Tree search, where histories instead of belief states are used to represent nodes. Similar to UCT in fully observable Markov processes, the value of a history node $V(h)$ is defined by the number of times it was visited during the simulations $N(h)$. PO-UCT also uses the UCB method:

$$\hat{a} = \underset{a}{\operatorname{argmax}}\, V(ha) + c\sqrt{\frac{logN(h)}{N(ha)}} \tag{9}$$

to select the actions to take while traversing a tree and uses a rollout policy when outside the tree [16]. As mentioned earlier, updating a belief can be a burden for both offline and online solvers. Instead of computing $\tau(b,a,z)$, POMCP [22] approximate this value by maintaining unweighted particle filters during simulations. More information on UCT and POMCP are presented in [13, 22].

6.4 POMDP Model

We utilize the Schnider *pharmacokinetic* (PK) model [20] to describe the time-dependent distribution of propofol within the surgical patient. It is a multi-compartment distribution model that permits the estimation of propofol concentration in various regions of the patient. The model provides an *effect site* compartment to model the point of propofol's action on the central nervous system. By estimating the concentration of propofol at this site of influence, the hypnotic effect can also be estimated.

To estimate the hypnotic effect of propofol concentrations, we utilized the Doufas *pharmacodynamic* (PD) model [8]. Propofol's dose response curve is non-linear and sigmoid-like; to smoothly approximate this curve, we trained a three-layer feed-forward neural network using the observations of [8].

In this study, at every episode, the agent tries to achieve a certain BIS value, which we will refer for the rest of the paper as the BIS target ($\mathrm{BIS}_{\mathrm{target}}$). Throughout the process, multiple targets might be set at different time intervals. In general, these values range from 40 to 60, where the former is set for a patient to undergo a deeper hypnotic state, and the later for lighter purpose anesthesia [15]. In this case, lower targets would require more amount of propofol while higher targets require less. As mention earlier, it is very challenging to determine the exact amounts of propofol to be administered to a patient in order to achieve these targets. In this section, we present a POMDP model that aims to tackle this control problem by taking advantages of the underlying probabilistic model of the anesthesia control.

6.4.1 States

In this problem, a state is represented as of a 7-dimensional feature vector $s = (v_1, v_2, v_3, q_1, q_2, q_3, q_e)$, where the parameters are taken from Equations 1 — 4. We are not considering v_e as a state parameter because, it can be computed from v_1 as $v_e = v_1/10000$. In their work [11], Hu et al. included two more parameters in their feature vector. We only consider 7 parameters because we utilize the Doufas PD model [8], which estimates propofol effects with a trained neural network. These state parameters v_i and q_i represent the volume and the clearance of the i^{th} compartment respectively, so the resulting state space is continuous. Hence, the total number of states are infinite. Previous techniques that share similarities to ours tackled this problem using discretization techniques. In our approach, POMCP samples for states during transitions; therefore, it only considers important states that were visited when building the OR-tree to approximate the value functions. It is worth to mention that POMCP only computes value functions for sampled histories rather than all possible states.

6.4.2 Actions

The decision maker can choose from the following propofol rates:

$$\mathcal{A} = \begin{Bmatrix} 0.0 & 0.1 & 0.2 & 0.3 & 0.4 & 0.5 & 0.6 & 0.7 & 0.8 & 0.9 \\ 1.0 & 1.1 & 1.2 & 1.3 & 1.4 & 1.5 & 1.6 & 1.7 & 1.8 & 1.9 \\ 2.0 & 2.1 & 2.2 & 2.3 & 2.4 & 2.5 & 2.6 & 2.7 & 2.8 & 2.9 \\ 3.0 & 3.1 & 3.2 & 3.3 & 3.4 & 3.5 & 3.6 & 3.7 & 3.8 & 3.9 \\ 4.0 & 5.0 & 6.0 & & & & & & & \end{Bmatrix}$$

in (ml/min).

Also, when an action is chosen, it is applied without interruption for a predefined time duration, which range from 15 to 45 seconds. Usually, lower propofol rates are applied longer since they have smaller effects to patients. On the other hands, higher rates should be applied cautiously as the drug effects vary from patients to patients. As in any Markov model, the agent transitions to a new state s', which is sampled from $\mathcal{T}(s,a,s')$ after performing action a in state s. This transition is stochastic, and the agent does not fully know in which of all the possible states it is currently in. To learn better about the current state it requests an observation from the controller.

6.4.3 Observations

After performing an action a_t, the agent receives a BIS value measured from the patient. As a $\text{BIS}_{\text{measured}}$ alone does not suffice in determining if a target has been reached; therefore, both $\text{BIS}_{\text{measured}}$ and $\text{BIS}_{\text{target}}$ at every episode are used. Each observation is an error that represents a distance to the $\text{BIS}_{\text{target}}$. A positive $\text{BIS}_{\text{error}}$ means that the observed value is above the target, and a negative indication represents a lower value than a current assigned $\text{BIS}_{\text{target}}$. Since a BIS value is bounded in $[0, 100]$, the maximum number of observations possible in this model is 201, and they are determined as follows:

$$\mathcal{Z} = \{-100, -99, \cdots, 0, \cdots + 99, +100\}$$

It is known that the number of observations affect the performance of a POMDP solver. For many years, researchers have tried to alleviate this sort of problem by factorizing observations that share some similarities. Observation factorization can yield good results for some problems, especially the ones that exhibit similar characteristics.

Also, as stated earlier, only certain ranges of BIS are targeted for general anesthesia. These values range from 40 to 60. Therefore, the number of observations can be reduced further. For example, given that $\text{BIS}_{\text{target}} \in [40, 60]$, possible observations range from 40 below target to 60 above target. The observation set can be reduced further to:

$$\hat{\mathcal{Z}} = \{-40, -39, \cdots, 0, \cdots + 59, +60\}$$

6.4.4 Rewards

In this model, the reward function is set to minimize a control error by maximizing the following equation:

$$r = 100 - \mid \text{BIS}_{\text{target}} - \text{BIS}_{\text{measured}} \mid$$

6.4.5 Solving the Partially Observable Markov Decision Process model

The POMDP model is solved using the POMCP solver in [22]. At every decision step, it runs simulations for the anesthesia control process to build and update a history tree. The agent runs simulations on PK/PD models to test multiple actions and record observation after each action. The total number of visits to an observation during simulations define the quality of action that would likely lead to the observation. In POMCP, each node in the tree represent a history h, and it records number of visits $N(h)$ throughout all simulations and estimated value $V(h)$ of a history h. This a combination of a Monte Carlo method and Upper Confidence Tree methods on histories rather than states. This is basically the idea behind the POMCP method. More information can be seen in [22].

The POMCP solver uses particle filters to approximate the value of the new belief state b'. In this case, the solver generate a set of unweighted state particles to represent the current belief state. To update the belief state after taking an action a in state s, simulations are run to sample for possible next state particles s' i.e. $s' \sim \mathcal{G}(s, a)$, where $\mathcal{G}$ is a transition model that is defined by the pharmacokinetic model (1–4).

To solve the POMDP model, action values are approximated based on how many times they were suggested by the simulations during trials. During simulations, POMCP updates information values of all node visited until the process terminates. For example, suppose the previous history is h_{t-1}, and the simulation suggests to apply a_t of propofol. After some updates, the simulation proposes an observation zt_t. In this case, the number of visits to node $h_t = (h_{t-1}, a_t, z_t)$ is incremented.

At each decision step, the agent computes propofol rates to apply by searching the tree according to the most recent histories. At this point, the agent only chooses the action that has highest value by applying the following equation:

$$\hat{a} = \underset{a}{\text{argmax}}\, V(ha) + c\sqrt{\frac{logN(h)}{N(ha)}}, \tag{10}$$

which is know as the Upper Confidence Based (UCB) policy for the multi-armed bandit problem [13]. In Eq. 10, $N(h)$ represents the total number of visits to a history node h, $V(ha)$ is the value of taking action a from node h, and c is UCB parameters that weight confidence to future actions.

7 Experiments

Table 1: Demographics of Simulated patients

	Range
Age	[18 , 45] yrs
Weight	[45 , 90] kg
Height	[150 , 190] cm

To test the efficiency of these POMDP controllers, we run simulations on 1000 simulated intraoperative patients, which are chosen randomly. The overall demographics data about the patients are shown in Table 1. As before, parameters for these simulated patients were designed to closely follow real world scenarios of patients undergoing general anesthesia. During the experiments, patient's profiles were represented as combination of patient's age, height, gender, weight, and random noises to challenge the controller. The Schnider's pharmacokinetic [20] was used to estimate propofol concentrations in all compartments. The Doufas' pharmacodynamic [8] was utilized to estimate propofol effects. The effects were measured as BIS values given the current estimate of propofol concentrations in patient's body compartments and their non-linear effects to the patient.

For each patient, a study lasts at most about 250 minutes, where the controller is assigned to achieve randomly chosen 1, 2, or 3 anesthesia depth targets. The agent estimates a patient's state, then apply the action that would give the highest long-term rewards. In this case, it will try to reduce the errors as much as possible throughout the study. Results from the new POMDP controller are compared against the performance from a reinforcement learning controller that uses an adaptive neural network filter (RL-ANNF) [5]. The POMDP model is solved online with a future reward discount $\gamma = 0.69$ that we chose after running multiple trials.

7.1 Results

As shown in Table 2, the new POMDP controller delivered improved control performance in most steady state control metrics. The MDAPE and Woble were reduced from 3.15% to 0.13%. The controlled metric, which indicates the percentage of $\text{BIS}_{\text{measured}}$ to be within ± 5 BIS, was improved from 93.49% to 99.69%. These results show that the new POMDP controller produces good control quality.

Table 2: Simulated steady-state performance metrics.

	POMDP	RL-ANNF
MDPE‡	-0.04	-0.124
MDAPE‡	0.13	3.15
Wobble‡	0.13	3.15
Divergence⋆	0.000	0.001
Controlled ‡	99.69	93.49
T_{peak} †	2.0	3.45
T_{ss} †	3.25	7.5
RMSE §	0.14	0.43

†(min), ‡(%), ⋆(%/hr), §(BIS)

The RSME metric was reduced from 4.1 BIS to 0.14 BIS. These moderate control quality improvements highlight the efficiency of a POMDP model when applied to anesthesia hypnosis control. Errors were reduced because the POMDP controller relies on probabilistic values rather than modes, which is the case in current controller that utilizes patient state filter.

8 CONCLUSION

In this paper, a POMDP population-based controller is introduced to tackle observation uncertainties in patient's PK/PD parameters. The control model uses a 7-dimension state vector, and it considers deviations from a control target as observations. It is solved with the POMCP solver by Silver et al., which has been claimed to be able to solve problems up to 10^{56} number of states [22]. We tested this new approach on randomly selected simulated patients and compared results to a controller that assumes fully observability.

8.1 Discussion

The proposed model in this paper is based on the PK/PD models, which are population-based models. Therefore, the efficiency of the new controller depends on the variability of the population parameters. Also, various factors affect the response of propofol on a patient. For example, height, weight, gender, ethnicity, and patient's health are known to challenge good control.

To improve the new controller, we suggest to develop a POMDP model that relies on drug effects measurements i.e. $\mathrm{BIS}_{\mathrm{measured}}$. In this case, the model will be able to adapt to broader patients. However, the lack of robust state transition and observation models complicate the application of more patient-specific POMDP model. We anticipate that these challenges can be resolved with further study.

REFERENCES

[1] A R Absalom and G N C Kenny, 'Closed-loop control of propofol anaesthesia using bispectral index(TM): performance assessment in patients receiving computer-controlled propofol and manually controlled remifentanil infusions for minor surgery', *Brit J Anaesth*, **90**(6), 737–741, (2003).

[2] A. R. Absalom, V. Mani, T. De Smet, and M. M. R. F. Struys, 'Pharmacokinetic models for propofoldefining and illuminating the devil in the detail', *British Journal of Anaesthesia*, **103**(1), 26–37, (2009).

[3] Anthony R Absalom, Robin De Keyser, and Michel M R F Struys, 'Closed loop anesthesia: are we getting close to finding the holy grail?', *Anesthesia & Analgesia*, **112**(3), 516–8, (2011).

[4] Blai Bonet and Héctor Geffner, 'Solving pomdps: Rtdp-bel vs. point-based algorithms', in *IJCAI-09*, (July 2009).

[5] Eddy C. Borera, Brett L. Moore, Anthony G. Doufas, and Larry D. Pyeatt, 'An Adaptive Neural Network Filter for Improved Patient State Estimation in Closed-Loop Anesthesia Control', in *Proceedings of the 23rd IEEE International Conference on Tools with Artificial Inteligence (ICTAI)*, pp. 41–46, Boca Raton, Florida, USA, (November 7–9 2011).

[6] T De Smet, M M R F Struys, S D Greenwald, E P Mortier, and S L Shafer, 'Estimation of optimal modeling weights for a bayesian-based closed-loop system for propofol administration using the bispectral index as a controlled variable: a simulation study', *Anesth Analg*, **105**, 1629–38, (6 2007).

[7] T De Smet, M M R F Struys, M M Neckebroek, K Van den Hauwe, S Bonte, and E P Mortier, 'The accuracy and clinical feasibility of a new bayesian-based closed-loop control system for propofol administration using the bispectral index as a controlled variable', *Anesth Analg*, **107**, 1200–1210, (2008).

[8] A G Doufas, M Bakhshandeh, A R Bjorksten, S L Shafer, and D I Sessler, 'Induction speed is not a determinant of propofol pharmacodynamics', *Anesthesiology*, **101**, 1112–21, (2004).

[9] P S Glass, M Bloom, L Kearse, C Rosow, P Sebel, and P Manberg, 'Bispectral analysis measures sedation and memory effects of propofol, midazolam, isoflurane, and alfentanil in healthy volunteers', *Anesthesiology*, **86**(4), 836–847, (Apr 1997).

[10] Wassim M. Haddad, James M. Bailey, Tomohisa Hayakawa, and Naira Hovakimyan, 'Neural network adaptive output feedback control for intensive care unit sedation and intraoperative anesthesia', *IEEE Transactions on Neural Networks*, **18**, 1049–1066, (2007).

[11] Chuanpu Hu, William S. Lovejoy, and Steven L. Shafer, 'Comparison of some suboptimal control policies in medical drug therapy', *Operations Research*, **44**(5), pp. 696–709, (1996).

[12] Bideshwar K. Kataria, Sudha A. Ved, Honorato F. Nicodemus, Gregory R. Hoy, Dawn Lea, Michel Y. Dubois, Jaap W. Mandema, and Steven L. Shafer, 'The pharmacokinetics of propofol in children using three different data analysis approaches', *Anesthesiology*, **80**, 104–122, (1994).

[13] Levente Kocsis and Csaba Szepesvri, 'Bandit based monte-carlo planning', in *In: ECML-06. Number 4212 in LNCS*, pp. 282–293. Springer, (2006).

[14] B Marsh, M White, N Morton, and G N C Kenny, 'Pharmacokinetic model driven infusion of propofol in children', *Brit J of Anaesth*, **67**(1), 41–8, (Jul 1991).

[15] B L Moore, P Panousis, V Kulkarni, L D Pyeatt, and A G Doufas, 'Reinforcement learning for closed-loop propofol anesthesia: A human volunteer study', in *Conf Proc AAAI Innov App AI*, pp. 1807–13, (2010).

[16] Joelle Pineau, Geoffrey J. Gordon, and Sebastian Thrun, 'Point-based value iteration: An anytime algorithm for pomdps', pp. 1025–1032, (2003).

[17] I J Rampil, 'A primer for EEG signal processing in anesthesia', *Anesthesiology*, **89**(4), 980–1002, (Oct 1997).

[18] Stphane Ross, Joelle Pineau, Sbastien Paquet, and Brahim Chaib-draa, 'Online planning algorithms for pomdps', *Journal of Artificial Intelligence Research*, **32**, 663–704, (2008).

[19] V. Sartori, P.M. Schumacher, T. Bouillon, M. Luginbühl, and M. Morari, 'On-line estimation of propofol pharmacodynamic parameters', in *IEEE Engineering in Medicine and Biology Society*, Shanghai, China, (Sept. 2005).

[20] T Schnider, C F Minto, P L Gambus, C Andresen, D B Goodale, S L Shafer, and E J Youngs, 'The influence of method of administration and covariates on the pharmacokinetics of propofol in adult volunteers', *Anesthesiology*, **88**(5), 1170–1182, (May 1998).

[21] Jürgen Schüttler and Harald Ihmsen, 'Population pharmacokinetics of propofol: A multicenter study', *Anesthesiology*, **92**, 727–738, (2000).

[22] D. Silver and J. Veness, 'Monte-Carlo Planning in Large POMDPs', in *Advances in Neural Information Processing Systems (NIPS)*, (2010).

[23] M T J Spaan, 'Cooperative active perception using POMDPs', in *AAAI 2008 Workshop on Advancements in POMDP Solvers*, (July 2008).

[24] M M R F Struys, T De Smet, S D Greenwald, A R Absalom, S Bingé, and E P Mortier, 'Performance evaluation of two published closed-loop control systems using bispectral index monitoring: a simulation study', *Anesthesiology*, **100**(3), 640–700, (Mar 2004).