Which States Matter? An Application of an Intelligent Discretization Method to Solve a Continuous POMDP in Conservation Biology

Sam Nicol^{1*}, Iadine Chadès²

¹ Institute of Arctic Biology, University of Alaska Fairbanks, Fairbanks, Alaska, United States of America, ² CSIRO Ecosystem Sciences, CSIRO, Brisbane, Queensland, Australia

Abstract

When managing populations of threatened species, conservation managers seek to make the best conservation decisions to avoid extinction. Making the best decision is difficult because the true population size and the effects of management are uncertain. Managers must allocate limited resources between actively protecting the species and monitoring. Resources spent on monitoring reduce expenditure on management that could be used to directly improve species persistence. However monitoring may prevent sub-optimal management actions being taken as a result of observation error. Partially observable Markov decision processes (POMDPs) can optimize management for populations with partial detectability, but the solution methods can only be applied when there are few discrete states. We use the Continuous U-Tree (CU-Tree) algorithm to discretely represent a continuous state space by using only the states that are necessary to maintain an optimal management policy. We exploit the compact discretization created by CU-Tree to solve a POMDP on the original continuous state space. We apply our method to a population of sea otters and explore the trade-off between allocating resources to management and monitoring. We show that accurately discovering the population size is less important than management for the long term survival of our otter population.

Citation: Nicol S, Chadès I (2012) Which States Matter? An Application of an Intelligent Discretization Method to Solve a Continuous POMDP in Conservation Biology. PLoS ONE 7(2): e28993. doi:10.1371/journal.pone.0028993

Editor: Mark S. Boyce, University of Alberta, Canada

Received: June 14, 2011; **Accepted:** November 17, 2011; **Published:** February 17, 2012

Copyright: © 2012 Nicol, Chadès. This is an open-access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Funding: The work was supported by a University of Queensland postgraduate research scholarship. The University of Queensland website is <http://www.uq.edu.au/>. An additional stipend scholarship was provided to SN by the applied environmental decision analysis research center (<http://www.aeda.edu.au/>). IC received salary from CSIRO ecosystem sciences while participating in the research (<http://www.csiro.au/org/CSE.html>). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: scnicol@alaska.edu

Introduction

Conservation managers must manage populations of endangered species despite being uncertain about the population dynamics and the exact size of the population of interest. Resources spent on intensive monitoring reduce expenditure on management that directly improves species persistence. However monitoring may prevent poor management actions being taken due to observation error in abundance data. Partially observable Markov decision processes (POMDPs) are decision models that account for stochastic population dynamics as well as imperfect detection of the population. POMDPs are notoriously difficult to solve when the number of states is even moderately large because the solution algorithm efficiency decays exponentially with the size of the state space (finite-horizon problems have a complexity that is P-SPACE complete [1]; for infinite horizon problems the complexity is undecidable [2]). In recent years, efficient methods have been developed in artificial intelligence to solve POMDPs approximately providing a suite of tools that can be used in ecology and conservation biology. The methods to solve POMDPs require populations of individuals to be split into discrete states, but choosing a discretization introduces a dilemma: if there are too many states in the discretization, then the POMDP solution algorithm becomes intractable; but if too few states are chosen,

then the discretization is too coarse and the results can lose their biological significance. Our paper's answer to the dilemma is to have just enough states so that managers can adequately discern between the effects of management actions.

Many ecological studies use continuous models (e.g. theta-logistic model, Beverton-Holt model [3]) to estimate population size. These models have the advantage of having well-known solutions and established methods for parameter estimation. However these models do not naturally suggest a state space discretization which is required for many optimal decision making techniques. In this paper we make use of an automated state space discretization method named Continuous U-Tree (CU-Tree) [4]. CU-Tree compactly discretizes the state space by finding the areas of the state space that are necessary to discern between the effects of the management actions. CU-Tree only adds states where they are statistically necessary to maintain the optimal solution, which can reduce the number of states without simplifying the biological dynamics. We exploit this property to find an optimal management policy for a partially observable population of the Washington sea otter.

Case study: The Washington sea otter

We demonstrate the benefits of an intelligent discretization approach in conservation by designing an optimal management policy for a partially observable population of the Washington sea

otter (*Enhydra lutris*). The Washington sea otter population is relatively small in geographic area and is located close to busy shipping lanes, meaning oil spills due to shipping accidents pose a considerable threat to the population. The population is partially observable, meaning that it is not possible to count the true population size exactly. The management objective is to maintain the sea otter population within 20% of the population carrying capacity given our belief in the accuracy of our population census (see methods section for further discussion of the management objective and the model).

We allow for four possible management actions to be taken at any time step. Only one action can be taken during each (annual) time step. We assume that a management agency can: (i) reduce the frequency of oil spills, for example by restricting shipping near sea otter habitat; (ii) reduce the damage caused by an oil spill; (iii) re-introduce otters, for example otters can be introduced from healthy populations in other parts of North America, or (iv) do nothing. When considering the partially observable problem, we also allow for a monitoring action, which improves our ability to detect the true population size.

We use the best-fit logistic growth (Ricker) model obtained by [5] (hereafter the ‘Gerber’ model) to determine the optimal state-based management actions to take under different oil spill scenarios. We then consider how imperfect detection during a population census will affect the results.

The paper is organized as follows: we first present the results of applying our new method to the sea otter problem. We then discuss the discretization method and POMDP methods in the context of solving complex optimization problems in conservation biology. In the final section we give further details of our methods including the ecological problem, continuous population model, the state space discretization algorithm and POMDP.

Results

Our first step in solving the sea otter management problem was to determine the optimal management strategy for the population assuming perfect detectability. As noted above, we need to keep the number of states that are used in the discretization small so that the partially observable problem will be tractable. This is achieved using the CU-Tree algorithm (see methods section). The CU-Tree algorithm compactly represents the solution for any population size using 13 states (figure 1). Re-introduction of new otters is the best option for small populations of sea otters and reducing oil spill damage is the best option for populations larger than 497 otters.

Managing the sea otter with imperfect detection

The CU-Tree solution assumes that the observed population size is the true population size. We call this solution perfectly observable, or just the Markov decision process (MDP) solution. In

reality, managers are never certain that the observed state after sampling is the true state. A population census may underestimate (e.g. the survey may not detect all individuals) or overestimate the number of individuals in the population (e.g. by double counting or inaccurate extrapolation of representative samples). We account for this uncertainty by recording the probability that we are in a state based on the previous behaviour of the population. Our optimization model becomes a partially observable Markov decision process (POMDP).

The structure of the optimal solution for the perfectly-detectable population (figure 1) is such that there are only two important actions—introduce otters when the population is small, or reduce the intensity of oil spills if the population is large. This means that even though the true population cannot be counted exactly, the true population size will not affect the best management action unless the true population size is close to the point where the best management action changes. Using this information we constructed and solved a POMDP for the sea otter problem (see methods section and figure 2). Solving the POMDP means we can take the best action even if we are unsure about the true size of the population.

We found exact solutions to the finite-horizon POMDP for up to 6 timesteps. However if we need to manage the population indefinitely an infinite time horizon solution is required. This was too computationally demanding to solve using an exact algorithm, so we solved the infinite time problem using the point-based approximation algorithm Perseus [6].

An optimal POMDP solution is presented as a policy graph. A policy graph depicts states and optimal actions as nodes of a graph. The outcomes of the possible observations are contained in the arrows connected to the node. Using the Perseus solution, we derived an approximate infinite-time horizon policy graph where all states are initially allocated equal belief (i.e. we have no prior information about the population) (figure 3). It is impractical to show the full policy graph because the solution contains 353 nodes. Figure 3 shows all the optimal observations and actions for a 10-timestep simulation. The optimal strategy is to initially avoid extinction by assuming the population is small and re-introducing otters. This is repeated unless we observe a large population sufficiently many times that we become sure that the correct observation is large, at which point the optimal action becomes to reduce the damage caused by oil spills if they occur. Monitoring is optimal when we make conflicting observations and are in a belief state where we are uncertain about which of the two actions is optimal.

The value of information: comparing performance with and without perfect observability

To test the value of monitoring the otter population, we simulated the partially observable sea otter population and tested a

Figure 1. The CU-Tree state discretization showing the optimal management strategy for the fully-observable sea otter population.
doi:10.1371/journal.pone.0028993.g001

Figure 2. Schematic showing how observation probability is accounted for in the sea otter POMDP. Only two observations matter for management, corresponding to “small” and “large” populations. For populations that are very large or very small (outside the rectangle in the figure), the manager can observe the correct state perfectly. For the states within the rectangle (i.e. states within 250 otters of the population size where the optimal action changes), managers can only observe the correct state with probability 0.5, unless monitoring is undertaken.
doi:10.1371/journal.pone.0028993.g002

perfectly observable (MDP) strategy and a POMDP strategy in which we account for our observation error. In the MDP strategy we ignore the possibility that our observation can be inaccurate (so it is never worthwhile to monitor, but it is possible to unwittingly choose suboptimal actions), but in the POMDP strategy we explicitly account for observation error and allow for monitoring. Figure 4 plots the performance of the MDP and POMDP strategies on the otter problem averaged over 20 simulation runs. Initially, the POMDP outperforms the MDP because the POMDP maintains some belief that the population is in the reward state at each time step, while the MDP must explicitly visit the reward state to accumulate any reward (figure 4a) (see methods section for the definition of reward state). However in the long term the MDP and the POMDP give the same performance for this particular problem (figure 4b). This behaviour is a result of the high growth rate of the sea otter population (which causes the population to rapidly approach the carrying capacity) as well as the simple state discretization created by CU-Tree (in which all of the reward is contained in one state, and only two states are affected by poor observation). We obtained similar results for different relative values of monitoring cost (figure 5). Because the same reward is achieved in the long term regardless of whether or not we account for partial observability, investing in intensive monitoring is unnecessary in the case of the Washington sea otter. However this is not a general result for all POMDP problems. The sea otter population is robust to suboptimal MDP management but more threatened populations have been shown to benefit from being managed as a POMDP [7].

Discussion

We have shown how discretization methods can be used to make optimal decisions for a continuous partially observable Markov decision process. By choosing states based on the effects of management actions, the number of states required to compute the optimal solution is greatly reduced.

Given that the CU-Tree solution suggests just two actions, it is tempting to reduce the state space to only two states, with one representing each action. This creates fewer states, but the values of the two states are no longer known. The minimum requirement for preserving the optimal policy is that two states that are aggregated into the same class must have the same optimal Q-value [8] (see methods section for definition of Q-value). Under this condition the aggregate states will yield a value function whose greedy policy is optimal in the original, non-aggregated MDP (i.e. the action that maximizes immediate reward in the aggregate state

is the same as the actions that maximize the long-term expected sum of rewards of the original states that comprise the aggregate state). This means that we can devise aggregation techniques that alter the reward structure and transition probabilities between states, but we must preserve the mapping of Q-values to actions to maintain the same optimal management strategy. CU-Tree satisfies the conditions for maintaining the optimal actions after aggregation.

CU-Tree automatically splits the states wherever the Kolmogorov-Smirnov test finds that the sampled distributions on either side of the proposed split are significantly different. The splits do not occur in exactly the same location in every run of the CU-Tree algorithm, but depend on which states are sampled during the data gathering phase. The number of splits made by the algorithm also varies, and generally increases as the number of loops through the data gathering phase increases (we ran the algorithm for 50 loops). This means there is a trade-off between creating more splits to refine the Q-values (see methods section) and keeping the number of splits low enough to effectively solve the POMDP. We found that CU-Tree rapidly identified the population size where the actions change (496 otters) but took longer to split the other states more finely. As described previously, what matters more than the exact split locations is the management action that is recommended for each state. The states that appear as narrow bands in figure 1 are a product of this stochastic state-splitting.

Other state aggregation techniques exist that may benefit conservation. Lee and Lau [9] used an adaptive vector quantization method to partition a state space using a simulator in problems with multiple dimensions. This approach is similar to CU-Tree, however it is not clear in CU-Tree how to rank and sort the states if there are data in more than one dimension. Conservation management problems with many continuous state variables include predator-prey systems and stage-structured population models. Optimization of these large state-space problems is currently considered intractable by the majority of the conservation community. There is an opportunity to apply recent advances in continuous POMDP methods from artificial intelligence (e.g. [10], [11]) to these conservation problems.

Using CU-Tree to manage continuous populations with imperfect detectability

Where partial observability has been accounted for in optimal management problems in conservation, it has been confined to problems with few states or very coarse discretizations (e.g. [12], [13], [14],[7]). The compact discretizations created by CU-Tree

Figure 3. Approximate infinite-time horizon policy graph where all states have the same initial belief.
doi:10.1371/journal.pone.0028993.g003

maintain the resolution of the original population and provide a missing link between the well-developed methods of population detectability in continuous models [15] and POMDP solution algorithms.

Although CU-Tree enabled us to reduce the state space down to 13 states in the sea otter population, not all populations will result

in such a small state space. Even with 13 states, we were unable to solve the POMDP exactly for more than 6 timesteps, so more effective state space compaction methods would be very useful. Although approximate methods like Perseus [6] can obtain near-optimal policies for the infinite-time case, very large state spaces will reduce the accuracy of approximate methods.

Figure 4. Performance plots showing rewards accumulated using both the MDP and POMDP strategies calculated over (a) a finite 6-year horizon calculated using an exact algorithm (Cassandra's algorithm); and (b) between 20 and 40 years calculated using the infinite time approximate Perseus solution. More information on Cassandra's algorithm and Perseus are included in the methods section of the manuscript.

doi:10.1371/journal.pone.0028993.g004

Despite these issues, populations are frequently modelled with the same structure (e.g. logistic growth) and similar management objectives (i.e. grow or eliminate the population). We suspect this structure is reflected in the optimal management strategy. Because the population increases smoothly, similar population sizes will tend to require the same management action (e.g. small population sizes generally need an action that prevents extinction, whereas the action for larger populations is often to prevent catastrophic mortality). This local similarity between datapoints should result in a discretization with relatively few states. An area for future research is to investigate whether the underlying structure of population models may ensure that the number of states and observations (i.e. areas where the management action change) remains relatively small.

Lessons for sea otter management

The optimal management strategy for the sea otter with imperfect observability depends on the relative cost of monitoring compared to management. Where monitoring costs were the same as management, monitoring was never optimal. As monitoring costs decrease relative to management, monitoring becomes optimal when conflicting

information is received about the true state of the otter population (figure 3). Despite these differences, monitoring did not affect long-term utility for this problem for a range of relative costs (see figure 5). For a detailed study about how management and monitoring decisions are affected by monitoring cost and belief state, see [13].

Under the assumptions of this study, imperfect detection of the sea otter population size was not important in the long term, as it gave results that were the same as if perfect detection was assumed. This long-term behaviour suggests that if a healthy population of otters can be maintained for some time, then the best course of action is to manage the population optimally despite having some uncertainty about the exact population size. Rather than expend funding on monitoring, the funding would be better used on managing the threat of oil spills on the otter population.

Materials and Methods

Management objective

Published estimates of the carrying capacity K for the Washington sea otter population vary [5,16] depending on the

Figure 5. Performance plots showing rewards accumulated using both the MDP and POMDP strategies calculated with variable costs the long term (20–40 years). All figures were calculated with the infinite time approximate Perseus algorithm. Figures show: (a) cost of monitoring = 10% of management action cost; (b) cost of monitoring = 20% of management action cost; (c) cost of monitoring = 100% of management action cost. In the long term, monitoring does not make a difference to the objective regardless of the relative management cost. More information on Cassandra's algorithm and Perseus are included in the methods section of the manuscript.

doi:10.1371/journal.pone.0028993.g005

modelling method used. For this study we adopt the best estimate of $K=621$ otters from the Gerber model. We assumed that a healthy population size is a population that is within 20% of the carrying capacity, so the management objective is to achieve a sea otter population between 497 and 746 individuals.

We note that this target differs from that in the Washington sea otter recovery plan [17] that is currently used to manage the population (downlist from State Endangered to State Threatened when population is maintained between 1640–2187 otters). We use the Gerber [5] estimate as it provides a population growth model, which is essential for obtaining the state-based management solution. We used a different target to the recovery plan as the Gerber model will very rarely (if ever) reach the target population size given the current parameters.

The model

Gerber et al. [5] found that the data fit a Beverton-Holt model and logistic growth (Ricker) model equally well. In this study a modified stochastic Ricker model

$$N_{t+1} = N_t \exp\left(r\left(1 - \frac{N_t}{K}\right) + \varepsilon\right) - J_t(f, t)N_t, \quad (1)$$

is used to model the otter population, where N_t is the number of otters at time t , the growth rate $r=0.25$ and the carrying capacity $K=621.4$ [5]. The environmental stochasticity ε is normally distributed according to $\mathcal{N}(0, 0.015)$. The term $J_t(f, t)N_t$ gives the otter mortality due to an oil spill. Spills occur randomly with frequency f and damage $J_t(f, t)$, where t is a random draw from a uniform distribution that determines whether an oil spill occurs. If a spill occurs, the proportion of otters killed by a spill $J_t(f, t)$ is chosen randomly between a maximum and a minimum damage.

Although the sea otter is unlikely to go extinct in the absence of oil spills, the risk posed by oil spills on the Washington coast remains uncertain [5]. For the purposes of this paper we assumed that the oil spill frequency f is 0.3, and oil spill mortality $J_t(f, t)$ is between 0–50%. We carried out sensitivity analysis on these parameter values but found that the choice of frequency and mortality did not significantly affect the recommended management action (see file S1).

We allow four management actions for the sea otter population. Only one action can be taken each year. We assume that a management agency can: (i) reduce the frequency of oil spills by 20%, ($f \rightarrow 0.8f$ in equation 1) for example by restricting shipping near sea otter habitat; ii) reduce the damage caused by an oil spill by 20% ($J_t \rightarrow 0.8J_t$); (iii) re-introduce 50 otters ($N_{t+1} \rightarrow (N_{t+1} + 50)$ in equation 1), for example otters can be introduced from healthy populations in other parts of North America, or (iv) do nothing (no effect on model). Although translocation of 50 otters is quite a large translocation, we expect that the costs of redirecting oil tankers or oil spill cleanup will be expensive. Translocating 50 otters may become feasible when compared with these costly actions. There may also be other barriers to management implementation such as inter-agency cooperation, but these matters are not addressed here. In this manuscript we focus solely on achieving the management objective for the sea otter population.

Optimal management of sea otters

State space discretization. In this section we demonstrate how to use the expected effects of management actions to choose a compact set of states that are necessary to solve the optimal decision problem in a Markov decision process framework [18]. We first assume that the population is perfectly observable, then add partial

observability later in the paper. The continuous U-Tree algorithm [19] (CU-Tree), an extension of the earlier U-Tree algorithm [4], generates a state space discretization and optimal management policy from a continuous state space with discrete actions. The method finds the positions where the optimal action changes to a greater resolution than a coarse uniform discretization and does not require prior knowledge of the transition probabilities.

We use CU-Tree to learn a discretization from the population model given by equation 1. CU-Tree goes through two phases to build a discretization: a data gathering phase, in which the population model is simulated to obtain data, and an expansion phase, where the simulated data is used to create new states by splitting the existing states according to a splitting criterion. We denote the state space using $\mathcal{S}$ and the action space using $\mathcal{A}$. Initially, CU-Tree has just one state s , which encompasses the whole range of possible population sizes (i.e. $\mathcal{S} = \{s\}$).

In the data gathering phase, the model (equation 1) is used to simulate a number of trajectories of a given length (we used trajectories of 50 timesteps in this paper). All simulated data is stored as datapoints. A datapoint is defined as the quadruple $(N_t, a, N_{t+1}, r(N_{t+1}, a))$, where N_t and N_{t+1} are the population sizes at times t and $t+1$ respectively, a is the action taken at time t , and $r(N_{t+1}, a)$ is the immediate reward associated with population size N_{t+1} and action a . Datapoint rewards $r(N_t, a)$ are allocated according to the current population size and management action. We allocate a reward of 100 points for all population sizes within 20% of the carrying capacity ($497 < N_t < 746$) and subtract the cost of action from this reward. Management actions other than ‘do nothing’ incur a cost of 10 units each. The datapoint reward $r(N_{t+1}, a)$ is defined as:

$$r(N_{t+1}, a) = \begin{cases} 100 - c(a) & , \text{if } 497 < N_{t+1} < 746 \\ -c(a) & , \text{otherwise} \end{cases} \quad (2)$$

where the cost of action $c(a)=0$ if the action is ‘do nothing’, and $c=10$ otherwise.

Estimates of the transition probability $P(s'|s, a)$, state reward function $R(s, a)$ and Q-values $Q(s, a)$ are updated at each simulation timestep. The transition probability from state s to state s' after taking action a and the expected future reward function are updated according to [20]:

$$P(s'|s, a) = \frac{|\{(N_t, a, N_{t+1}, r(N_{t+1}, a)) \in T(s, a) \text{ s.t. } N_{t+1} \in s'\}|}{|T(s, a)|} \quad (3)$$

$$R(s, a) = \frac{\sum_{(N_t, a, N_{t+1}, r(N_{t+1}, a)) \in T(s, a)} r(N_{t+1}, a)}{|T(s, a)|} \quad (4)$$

where $(N_t, a, N_{t+1}, r(N_{t+1}, a))$ is the datapoint being updated and $T(s, a)$ is the set of all transition tuples currently stored in the transition history that have starting state s and use action a .

The Q-values record the current value of each state and are updated using the Bellman equation [18, 21]:

$$Q(s, a) = R(s, a) + \gamma \sum_{s' \in \mathcal{S}} P(s'|s, a) V(s') \quad (5)$$

where the value function $V(s)$ is given by $V(s) = \max_{a \in \mathcal{A}} Q(s, a)$. As datapoints are added to the Q-values, $Q(s, a)$ will converge to

the global reward obtained by applying a management action a to state s .

After the data collection phase, an expansion phase checks whether new states should be added. Trial splits are added between every stored datapoint and the trial split that results in the largest difference in Q -values is selected. If the difference in Q -values of the distributions on either side of the trial split is statistically significant then a split is added and a new state is created. Significance is tested using a two-sample Kolmogorov-Smirnov test [22] with a null hypothesis that the two distributions are from the same distribution ($p=0.01$). The algorithm is repeated until no significant splits can be added. The block diagram in figure 6 summarises the steps involved in CU-Tree.

CU-Tree will eventually converge to the optimal management policy. The optimal management policy is obtained from the discretization using the Q -values generated by CU-Tree. The optimal policy $\pi^*(s)$ for state s is the action that maximizes the value function:

$$\pi^*(s) = \arg \max_{a \in A} [Q^*(s, a)]. \quad (6)$$

where $Q^*(s, a)$ are the Q -values $Q(s, a)$ after many iterations of the algorithm. The sampling loop was repeated 250 times to generate the discretization for the sea otter population. A MATLAB implementation of CU-Tree applied to the sea otter problem is included in file S2.

Partially observable Markov decision processes. The CU-Tree solution assumes that the observed population size is the true population size. We call this solution perfectly observable, or just the Markov decision process (MDP) solution. In reality, the problem is partially observable because managers cannot detect all the individuals in the population in a census. A partially observable Markov decision process (POMDP) consists of a discrete state space S , a discrete set of actions A which can be applied to S , and immediate rewards $r(s, a) \in R$ that will be received for taking action $a \in A$ on state $s \in S$. The environment changes from state s to state s' according to a transition probability model $P(s'|s, a)$. Because the problem is partially observable, a manager of the system cannot know the state of the system exactly but will instead make observations $O \in o$ that give information about the true state of the system through an observation function $P(o|s, a)$. As observation data is collected, the manager builds up a belief state $b(s, a)$ that gives the probability that the state observed after taking an action is the true population state. Interested readers can refer to [23] for further information on POMDPs.

The CU-Tree discretization for the sea otter problem reduced the state space to 13 states, and we use this to define the state space for the POMDP ($|S| = 13$). The objective of the POMDP is to discover the optimal action to take given a set of beliefs that we are in each state—in this case to find the actions that will maintain the sea otter population within 20% of the population carrying capacity with the highest probability. We allow the same four actions as the perfectly observable case, as well as an option to monitor the population ($|A| = 5$ actions). Monitoring the popula-

Figure 6. Block diagram of the continuous U-tree algorithm. Datapoints are collected and stored by simulation using the population model. For each state, datapoints are sorted into ascending population sizes. Trial splits are added between every stored datapoint and the trial split that results in the largest difference in Q -values is selected. If the difference in Q -values of the distributions on either side of the trial split is statistically significant, then a split is added and a new state is created. If the difference in Q -values is not significant, then no split is added. The algorithm is repeated until a stopping criterion is reached. doi:10.1371/journal.pone.0028993.g006

tion temporarily increases the probability that we observe the correct state. We assume that all management actions cost 10 units, but monitoring costs 2 units. Transition probabilities and rewards are derived from CU-Tree (equations 3 and 4).

To implement a POMDP we define the possible observations $o \in O$ and the probabilities that our observations are correct $P(o|s,a)$. CU-Tree defined 13 states for the sea otter population (see results section). The optimal actions for the first 10 states are identical, and the optimal actions for the remaining 3 states recommend a second action. This means that a manager must choose between two actions based on his belief of the current system state. For this reason we allow just two observations – a manager will either observe a ‘small’ (0–496 otters) population, or a ‘large’ (>496 otters) population and act accordingly. Note that by choosing just two observation states we have not reduced the number of states in the problem. We cannot reduce the number of states to less than the number of states in the CU-Tree discretization because this will mean that we change the problem we are solving and lose our estimates of the transition probabilities and rewards for each state [8].

We assumed that managers always observe correctly in all states except for those near where the optimal decision changes. We set the probability $P(o|s,a)$ of observing correctly to be 1 for all states that occur outside of a ‘window’ of width 250 otters around the point where the optimal decision changes (around the point where the observation changes from ‘small’ to ‘large’) (figure 2). For states that fall within this width, managers may observe incorrectly – we set the probability of making the correct observation from these states to be 0.5. If the action ‘monitor the population’ is chosen, the probability of making the correct observation in these states rises to 0.95.

References

- Papadimitriou CH, Tsitsiklis JN (1987) The complexity of Markov decision processes. *Mathematics of Operations Research* 12: 441–450.
- Madani O, Hanks S, Condon A (2003) On the undecidability of probabilistic planning and related stochastic optimization problems. *Artificial Intelligence* 147: 5–34.
- Morris W, Doak D (2002) *Quantitative Conservation Biology: Theory and Practice of Population Viability Analysis*. Sunderland, MA: Sinauer Associates Inc.
- Uther WT, Veloso MM (1998) Tree based discretization for continuous state space reinforcement learning. In: *AAAI-98*. Madison, Wisconsin, USA. pp 769–774.
- Gerber LR, Buenau K, Vanblaricom G (2004) Density dependence and risk of extinction in a small population of sea otters. *Biodiversity and Conservation* 13: 2741–2757.
- Spaan M, Vlassis N (2005) Perseus: Randomized point-based value iteration for POMDPs. *JAIR* 24: 195–220.
- McDonald-Madden E, Chadès I, McCarthy M, Linkie M, Possingham H (2011) Allocating conservation resources between areas where persistence of a species is uncertain. *Ecological Applications* 21: 844–858.
- Li L, Walsh TJ, Littman ML (2006) Towards a unified theory of state abstraction for MDPs. In: *Proceedings of the Ninth International Symposium on Artificial Intelligence and Mathematics*. pp 531–539.
- Lee I, Lau H (2004) Adaptive state space partitioning for reinforcement learning. *Engineering applications of artificial intelligence* 17: 577–588.
- Porta J, Vlassis N, Spaan M, Poupart P (2006) Point-based value iteration for continuous POMDPs. *JMLR* 7: 2329–2367.
- Erez T, Smart W (2010) A scalable method for solving high-dimensional continuous pomdps using local approximation. In: *Proceedings of UAI-2010*.
- White B (2005) An economic analysis of ecological monitoring. *Ecological Modelling* 189: 241–250.
- Chadès I, McDonald-Madden E, McCarthy M, Wintle B, Linkie M, et al. (2008) When to stop managing or surveying cryptic threatened species. *PNAS* 105: 13936–13940.
- Tomberlin D (2010) Decision support for environmental monitoring and restoration: Application of the partially observable Markov decision process. In: *Proc. of the 5th iEMSs Congress on Environmental Modelling and Software*.
- Yoccoz NG, Nichols JD, Boulinier T (2001) Monitoring of biological diversity in space and time. *TREE* 16: 446–453.
- Laidre K, Jameson R, Jeffries S, Hobbs R, Bowlby C, et al. (2002) Estimates of carrying capacity for sea otters in Washington state. *Wildlife Society Bulletin* 30: 1172–1181.
- Lance M, Richardson S, Allen H (2004) Washington state recovery plan for the sea otter. Technical report, Washington Department of Fish and Wildlife, Olympia, WA.
- Puterman M (2005) *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley Series in Probability and Statistics. Wiley-Interscience.
- McCallum A (1995) Reinforcement learning with selective perception and hidden state. Ph.D. thesis, Dept. of Computer Science, University of Rochester, Rochester, NY.
- Chernova S, Veloso MM (2006) Tree-based policy learning in continuous domains through teaching by demonstration. Technical Report WS-06-13, AAAI.
- Bellman R (1957) *Dynamic Programming*. Princeton NJ: Princeton University Press.
- Gibbons J, Chakraborti S (2003) The general two-sample problem: the Kolmogorov-Smirnov two-sample test. In: *Nonparametric statistical inference*, New York, USA: Marcel Dekker Inc. 4th edition. pp 239–247.
- Kaelbling L, Littman M, Cassandra A (1998) Planning and acting in partially observable stochastic domains. *Artificial Intelligence* 101: 99–134.
- Cassandra A Partially observable Markov decision processes: pomdp-solver software (version 5.3). [online] <http://www.pomdp.org/pomdp/index.shtml>. Accessed 14 April 2010.

Supporting Information

File S1 Sea otter management: a sensitivity analysis.
(PDF)

File S2 MATLAB M-files to implement CU-Tree for the sea otter problem.
(TAR)

Acknowledgments

The authors wish to thank Alain Dutech and two anonymous reviewers for their useful comments on an early version of our manuscript.

Author Contributions

Conceived and designed the experiments: SN IC. Performed the experiments: SN. Analyzed the data: SN IC. Contributed reagents/materials/analysis tools: SN IC. Wrote the paper: SN IC.