Fast Simulation for Computational Sustainability Sequential Decision Making Problems

Sean McGregor,[1] Rachel Houtman,[2] Hailey Buckingham,[2]
Claire Montgomery,[2] Ronald Metoyer,[3] and Thomas G. Dietterich[1]
School of Electrical Engineering and Computer Science, Oregon State University[1]
College of Forestry, Oregon State University[2]
Department of Computer Science and Engineering, University of Notre Dame[3]

Abstract

Solving sequential decision making problems in computational sustainability often requires simulators of ecology, weather, fire, or other complex phenomena. The extreme computational expense of these simulators stymie optimization and interactive visualization of decision rules (policies). This work presents our results in creating an interactive visualization for a wildfire management problem whose simulator normally takes several hours to run. We successfully generate visualizations for a landscape's development over 100 year time spans within 17 seconds, when the original simulator took several hours.

1 Introduction

Many computational sustainability problems require making decisions through time, including invasive species eradication [1], and wildfire management [6]. Solving these problems involves maximizing expected reward by finding a policy that selects the best actions for configurations of the world. For example, in invasive species problems we perform the "eradication" and "restore" actions, then receive a reward proportionate to the number of native and invasive species on the landscape. In the wildfire management domain, which we use as an example throughout this paper, we select suppression decisions over 100 year time spans and receive rewards proportionate to the timber production of the landscape.

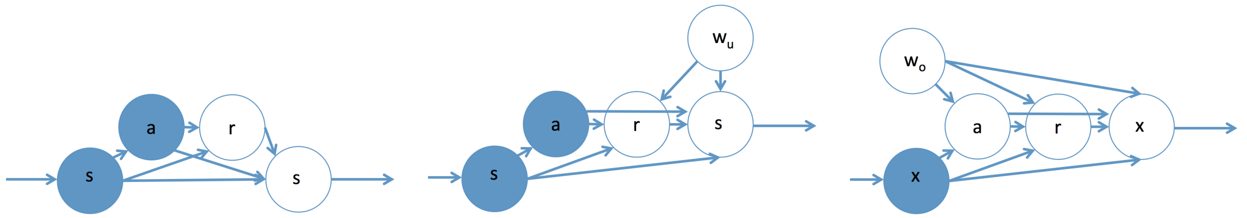
In machine learning we formalize these problems as Markov Decision Processes (MDPs), which describe the world in terms of a finite set of states (S), a finite set of possible actions that can be taken in each state (A), a function that gives the probability of entering a state after applying an action in a state (P), and a function providing rewards for taking actions in states ($R(s, a)$). Since ecological systems typically have more configurations than can be stored explicitly in a table, the function P is implemented as a simulator that generates hypothetical futures (trajectories) subject to a policy. These simulators are often sufficiently expensive to run that we must find ways to minimize their use when optimizing policies.

Many algorithms economize simulator expense by constructing trajectories for a policy based on the pre-computed results of a different policy. This *off-policy* policy evaluation is particularly important for supporting interactive visualization [7], which allow ecologists, land managers, developers, and policy makers to validate the assumptions incorporated into simulators.

Fonteneau et al.'s [2]'s Model Free Monte Carlo (MFMC) method is one approach for off-policy policy evaluation. MFMC synthesizes trajectories by piecewise *stitching* state transitions together from a database of previously-simulated state transitions. Since MFMC replaces simulations with a series of database queries, it can generate trajectories without waiting for the simulator to run.

Our work currently under review for the 2016 conference on Neural Information Processing Systems (NIPS) addresses two "curses of dimensionality" that make MFMC impractical for large state space MDPs that are typical of computational sustainability problems. First, it is difficult to sample sufficient state transitions to gain database coverage of high dimensional state-action spaces [5]. Second, it is difficult to determine similarity among state-action samples in higher dimensions. We solve these curses of dimensionality with a new algorithm, MFMCi, for problems with exogenous state variables, such as weather in a wildfire management problem.

4th International Conference on Computational Sustainability (CompSust 2016), Ithaca, New York.



(a) The standard MDP transition.

(b) MDP transition with unobservable exogenous variables (w_u).

(c) MDP transition with *observable exogenous* variables (w_o).

Figure 1: MDP probabilistic graphical models.

Since our goal is to support interactive MDP visualization, we use visual properties of the MDP visualization MDPvis [7] to evaluate the performance of MFMCi. We use the unitless measurement of "visual fidelity error," which is a measure of how similar MDPvis looks under MFMCi when compared to the visualization generated from the ground truth simulator.

We demonstrate MFMCi on a computationally expensive wildfire, timber, vegetation, and weather simulator that takes hours to generate single trajectories. The aim of the wildfire management simulator is to inform wildfire suppression policies that determine whether the US government will suppress a wildfire.

The fire simulator spreads fire spatially from an ignition point according to the surrounding pixel layers and the hourly weather sampled from 26 historical weather years. Weather variables include hourly wind speed, wind direction, cloud cover, minimum temperature, maximum temperature, temperature, humidity, and precipitation. We use MFMCi to synthesize trajectories by modeling the weather time series and ignition locations as exogenous variables. The weather is exogenous because, to a first approximation, neither the actions not the landscape influence the weather. Ignition location is exogenous to the landscape because tree cover does not affect the ignition's spatial probability distribution. Additionally, timber harvest and vegetation growth are deterministic functions of the landscape, which means every state transition contains their results.

In the next section we provide additional theoretical background on our algorithm leveraging independencies of exogenous state variables. In the results section we describe its performance in terms of our computationally expensive wildfire management problem domain.

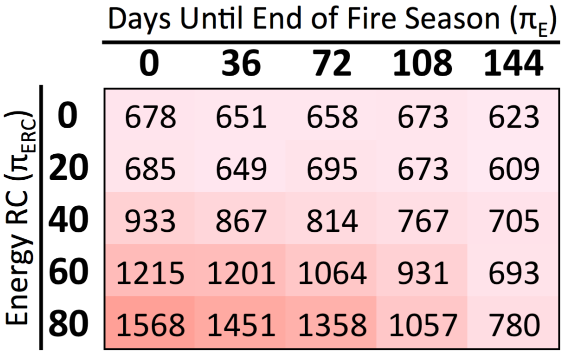
2 Methods

We address MFMC's dimensionality issue by exploiting *transition independencies* that factor the state space into *Markovian* variables that transfer between time steps, and *exogenous* variables that are combined with the Markovian variables at every time step. For example, in the wildfire domain, the state of the trees from one time step to another is Markovian, but we make decisions in response to exogenous weather events such as rain, wind, and lightning. By factoring out exogenous variables, we can synthesize trajectories from the (lower dimensional) Markovian state space.

We can define the transition independencies in terms of probabilistic graphical models in Figure 1. The standard MDP transition is in Figure 1a. Figure 1b shows the setting of prior work [2, 5] that model "unobservable random disturbances" (w_u). We call these *unobservable exogenous variables*, which are distinct from the *observable* exogenous variables (w_o in Figure 1c) of interest in this work.

MFMC normally selects state transitions from a database by matching a state action pair (s, a) to the first two elements (s', a') in the 4-tuple (s', a', r', s_{result}). It then adopts r' as the one step reward and s_{result} as the resulting state. In our approach, we show how to decompose s into (x, w_o) and then only match x against x' in the 5-tuple $(x', w'_o, a', r', x_{result})$. Our algorithm, MFMCi, then adopts w'_o as the instance of the exogenous random variable, r' as the reward, and x_{result} as the resulting state. In order for this to work correctly, two conditions must hold. First, the values of w_o in each state must be independent and identically distributed. Second, the database must contain a separate tuple $(x', w'_o, a, r', x_{result})$ for each possible action a (we refer to sets of tuples containing the same x and w, but different a as *transition sets*) so that we can look up the action corresponding to both the x and w_o. Since we guarantee an action consistent with the policy will be in the nearest tuple, we can reduce the dimensionality of the search for the nearest tuple to a distance in the space of x from the (s, a) space.

Fonteneau et al. [2] adopt Lipschitz continuity assumptions on the transition, reward, and policy functions to prove bounds on the bias and variance of the estimated return of a policy. Their bounds depend on the Lipschitz constants, the number of generated trajectories, and the sparsity of the database of transitions. We employed a similar set of assumptions, but by reducing the dimensionality of the database's space, we tighten the bounds derived from these assumptions.



Energy RC (π_{ERC})	Days Until End of Fire Season (π_E) 0	36	72	108	144
0	678	651	658	673	623
20	685	649	695	673	609
40	933	867	814	767	705
60	1215	1201	1064	931	693
80	1568	1451	1358	1057	780

(a) Visual fidelity error in weighted pixels across the policy space for the factored metric. The darkness of the cell is proportionate to the number of unsuppressed fires for the policy.

Database Size	Policy	μ	σ
$\lvert d\rvert$	$\Pi_{(ERC,E)}$	880	281
Unbiased $\frac{1}{2}\lvert d\rvert$	$\Pi_{(ERC,E)}$	913	273
Biased $\frac{1}{2}\lvert d\rvert$	$\Pi_{(ERC,E)}$	554	113
$\lvert d\rvert$	Π_L	750	172
Unbiased $\frac{1}{2}\lvert d\rvert$	Π_L	894	4
Biased $\frac{1}{2}\lvert d\rvert$	Π_L	1,224	20

(b) Visual fidelity error under the full, half (unbiased and biased) databases. The policy class $\Pi_{(ERC,E)}$ matches the policies of Figure 2a. The policy class Π_L has two ignition location-based policies.

3 Results and Discussion

We define policies in terms of two thresholds: π_E and π_{ERC}. Policy π_E is the maximum time until end of fire season at which the fire will be allowed to burn. π_{ERC} is the Energy Release Component (ERC) at which the fire is suppressed. We seek to visualize trajectories for all combinations of π_E and π_{ERC}.

Our quantitative evaluation showed several interesting results. First, Figure 2a shows our factored metric performs well across the entire policy space. The relatively higher values in the lower left of the chart result from leaving most of the sample's wildfires unsuppressed. Since unsuppressed fires have greater variance in the number of burned crown pixels (tree foliage) and surface pixels, it would take more samples to drive down the Monte Carlo variance of the visualization than is practical.

Table 2b lists visual fidelity error for two policy classes using both the full database, and a database with half as many samples. We constructed the halved databases by either removing all but one transition from the transition sets (biased), or by removing all samples associated with even-numbered trajectories in the database (unbiased). The additional policy class Π_L suppresses all fires on one half of the landscape and otherwise allows them to burn. The results show the transition set is more valuable for synthesizing trajectories for policy classes not found in the transition database. Such novel policies are an important part of exploratory analysis that will be performed by foresters.

MFMCi generates trajectories more than 1200 times faster than invoking the simulator. Without these speedups, interactive visualization and policy exploration of wildfire policies is not possible.

References

[1] T. Dietterich, M. Taleghan, and M. Crowley. PAC Optimal Planning for Invasive Species Management: Improved Exploration for Reinforcement Learning from Simulator-Defined MDPs. *Twenty-Seventh AAAI Conference on Artificial Intelligence*, 2013.

[2] R. Fonteneau, S. A. Murphy, L. Wehenkel, and D. Ernst. Model-Free Monte Carlo-like Policy Evaluation. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS 2010)*, pages 217–224, 2010.

[3] R. Fonteneau, S. a. Murphy, L. Wehenkel, and D. Ernst. Batch Mode Reinforcement Learning based on the Synthesis of Artificial Trajectories. *Annals of Operations Research*, 208(1):383–416, Sep 2013.

[4] R. Fonteneau and L. Prashanth. Simultaneous Perturbation Algorithms for Batch Off-Policy Search. In *53rd IEEE Conference on Conference on Decision and Control*, 2014.

[5] A. Hallak and T. Mann. Off-policy Model-based Learning under Unknown Factored Dynamics. *Proceedings of the 32nd International Conference on Machine Learning*, 37, 2015.

[6] R. M. Houtman, C. A. Montgomery, A. R. Gagnon, D. E. Calkin, T. G. Dietterich, S. McGregor, and M. Crowley. Allowing a Wildfire to Burn: Estimating the Effect on Future Fire Suppression Costs. *International Journal of Wildland Fire*, 22(7):871–882, 2013.

[7] S. McGregor, H. Buckingham, T. G. Dietterich, R. Houtman, C. Montgomery, and R. Metoyer. Facilitating Testing and Debugging of Markov Decision Processes with Interactive Visualization. In *IEEE Symposium on Visual Languages and Human-Centric Computing*, Atlanta, 2015.