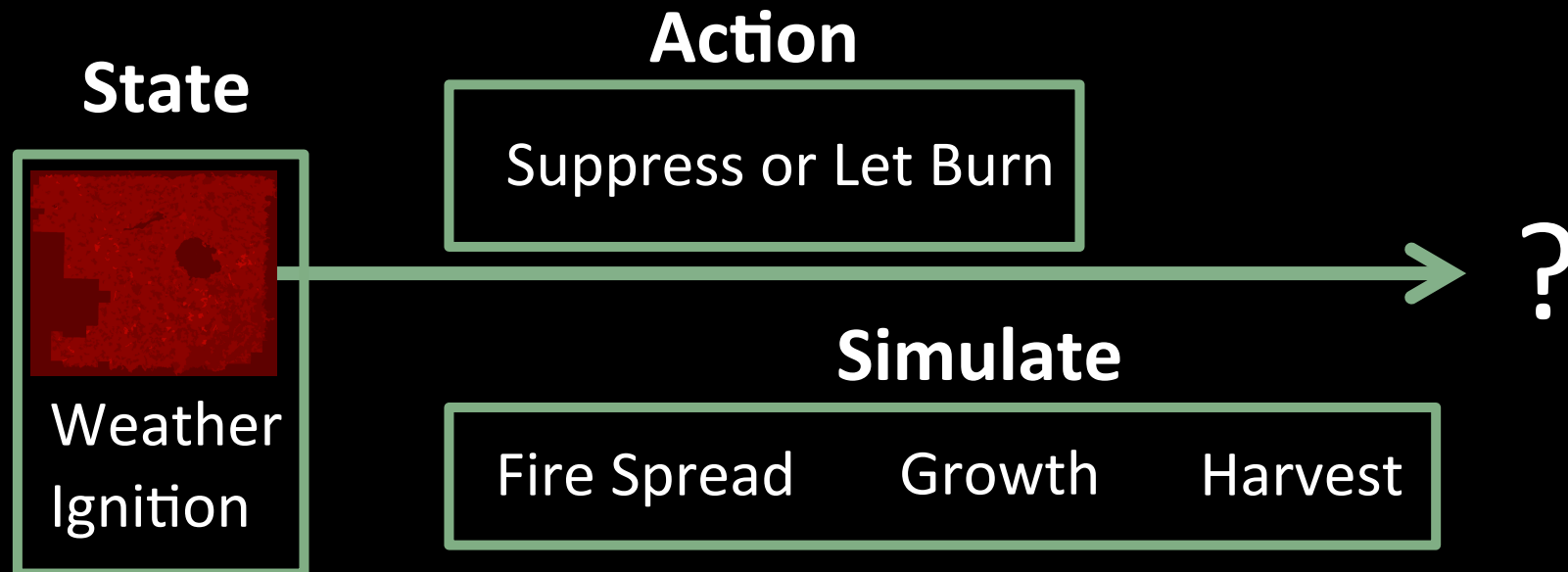


Fast Simulation for Computational Sustainability Sequential Decision Making Problems

Sean McGregor, Rachel Houtman, Hailey
Buckingham, Claire Montgomery, Ronald
Metoyer, Thomas Dietterich



Sequential Optimization in Wildfire Suppression Decisions



Simulate 100 Years of Decisions

Two Tasks

1. Optimizing policies
2. Validating policies

How do We Validate?

1. The optimization algorithm produced acceptable results
2. Simulation specification is correct



How do We Validate?

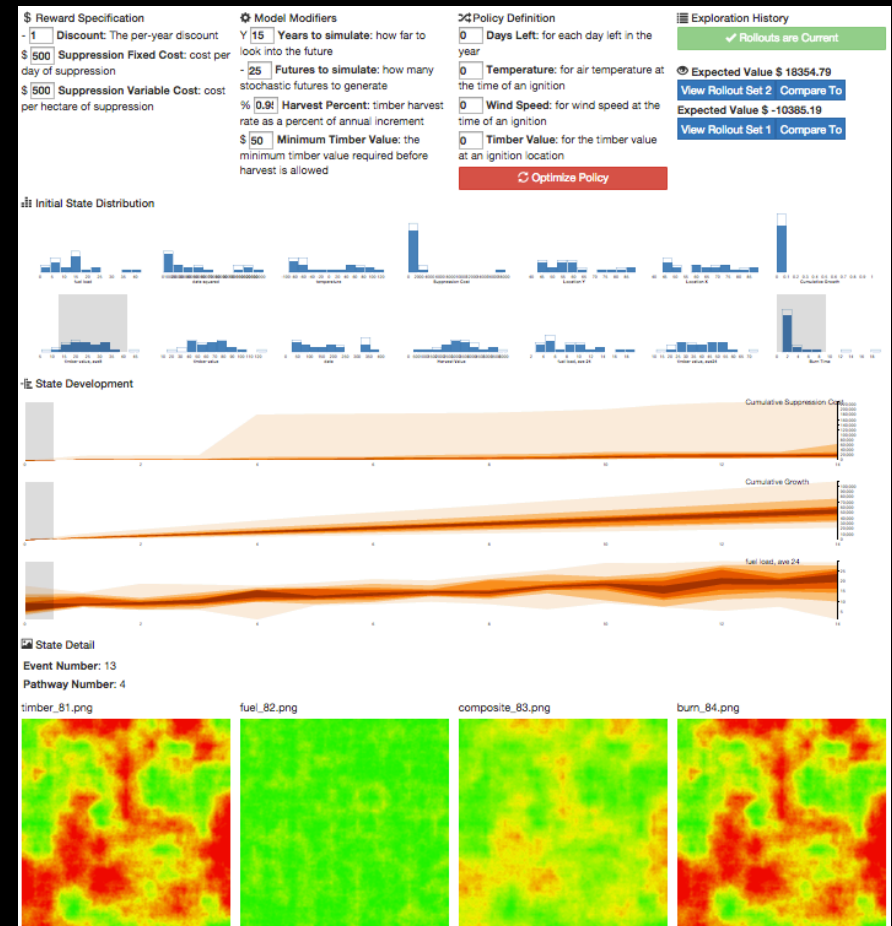
1. The optimization algorithm produced acceptable results
2. Simulation specification is correct



Visualization

Connect simulator to a visualization

1. Manually change parameters
2. Generate trajectories
3. Explore trajectories



How do We Validate?

1. The optimization algorithm produced acceptable results
2. Simulation specification is correct

Wildfire Reward Function

Timber Price * Timber Harvest



Reward = Timber Revenue + Ecology "Revenue" + Suppression Expenses



Ecological "Price" * Ecological State

Wildfire Reward Function

Suppress Everything!



Reward = Timber Revenue + Ecology "Revenue" + Suppression Expenses



Let Everything Burn!

Wildfire Reward Function

Selecting ecological reward selects policy

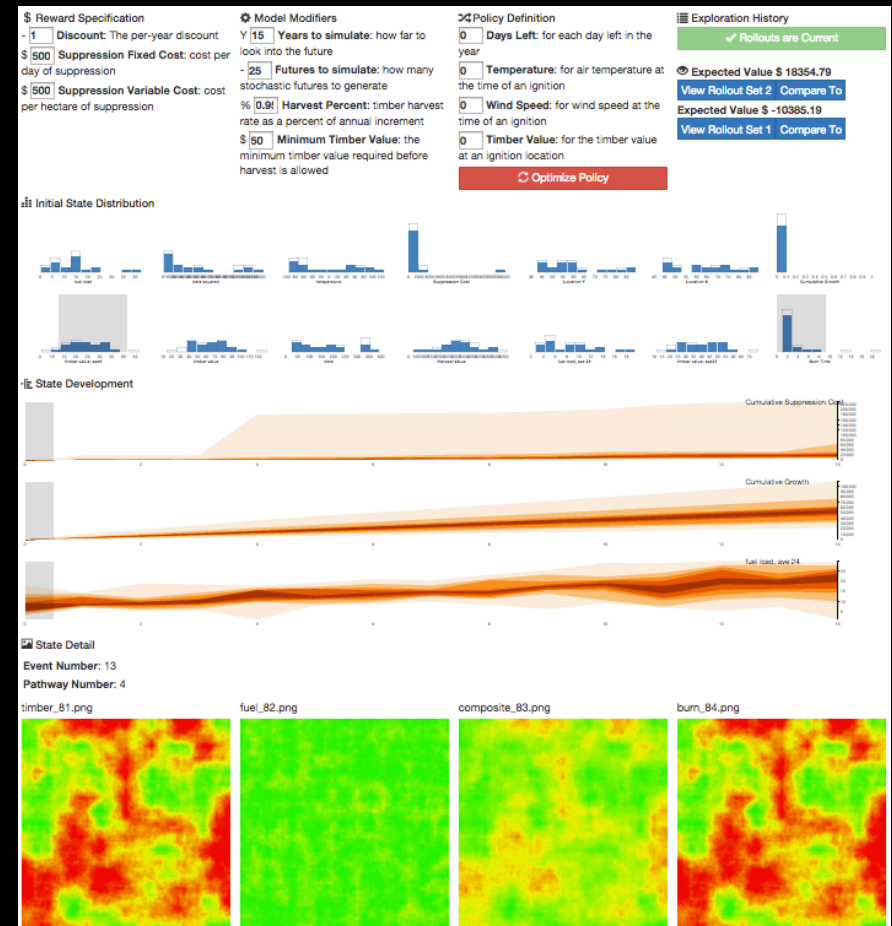
Visualization

Connect simulator to a visualization

1. Manually change parameters
2. Generate trajectories
3. Explore trajectories

Connect optimizer to visualization

1. Manually change parameters
2. Optimize policy
3. Generate trajectories
4. Explore trajectories



Problem

CompSust problems are (often) expensive to simulate

100 Years of Wildfire Simulation: *several hours*

Optimizing to 100 year horizon: *many days!*

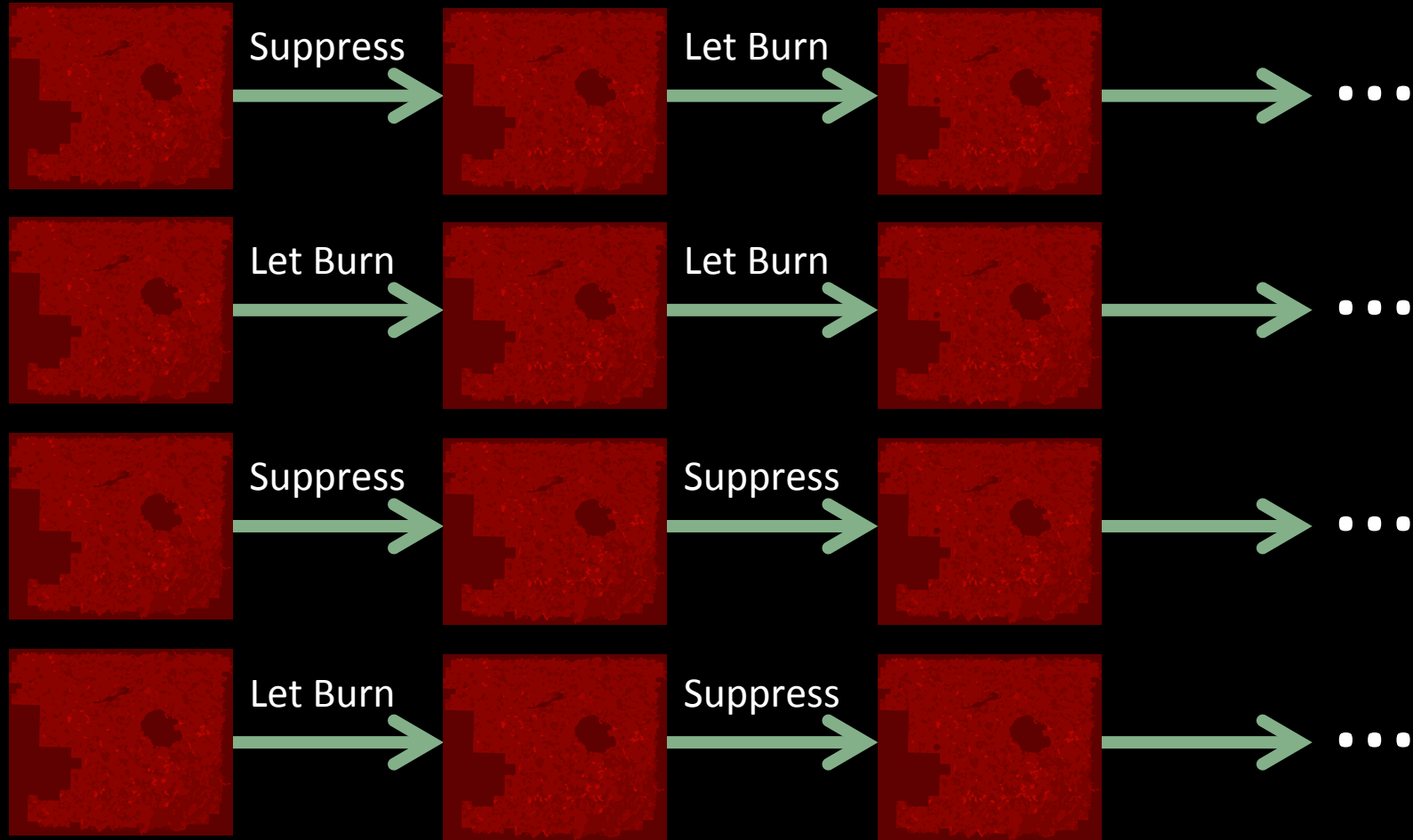
Solution

Trajectory Synthesis: Creating trajectories from a database of pre-computed state transitions

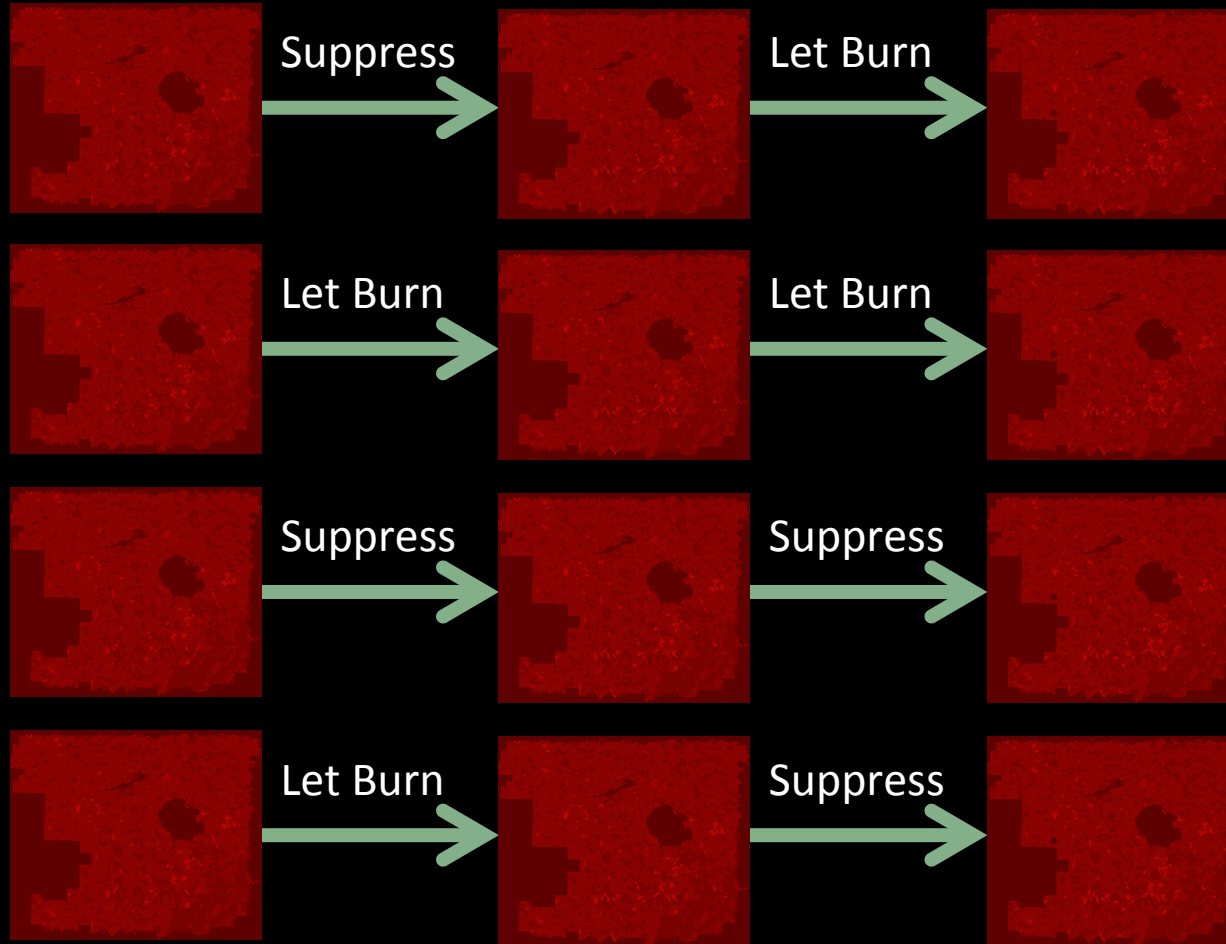
Benefits:

- Perform database queries at visualization time instead of expensive simulations
- Very sample efficient for exogenous variables

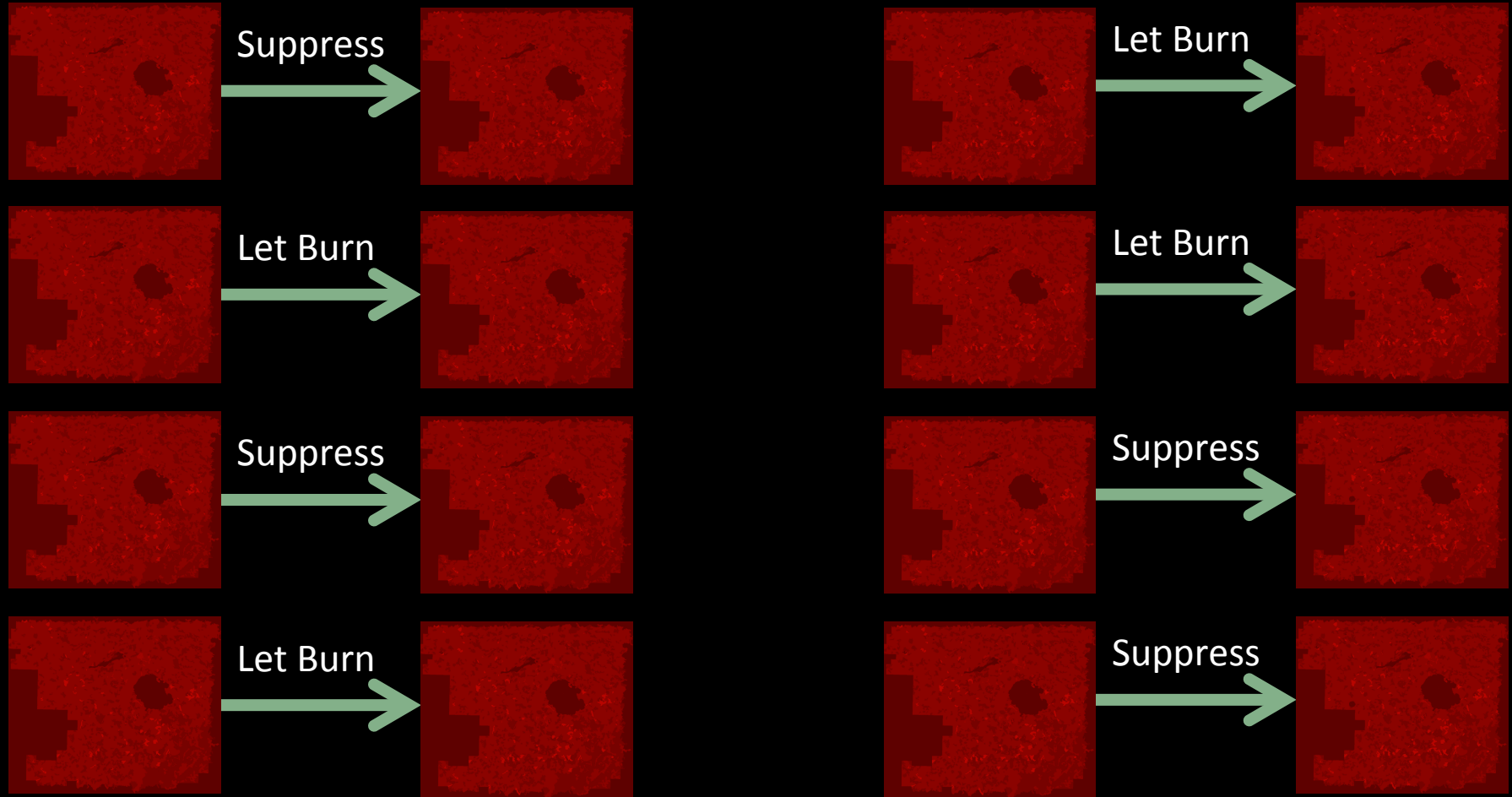
Constructing Database



Constructing Database

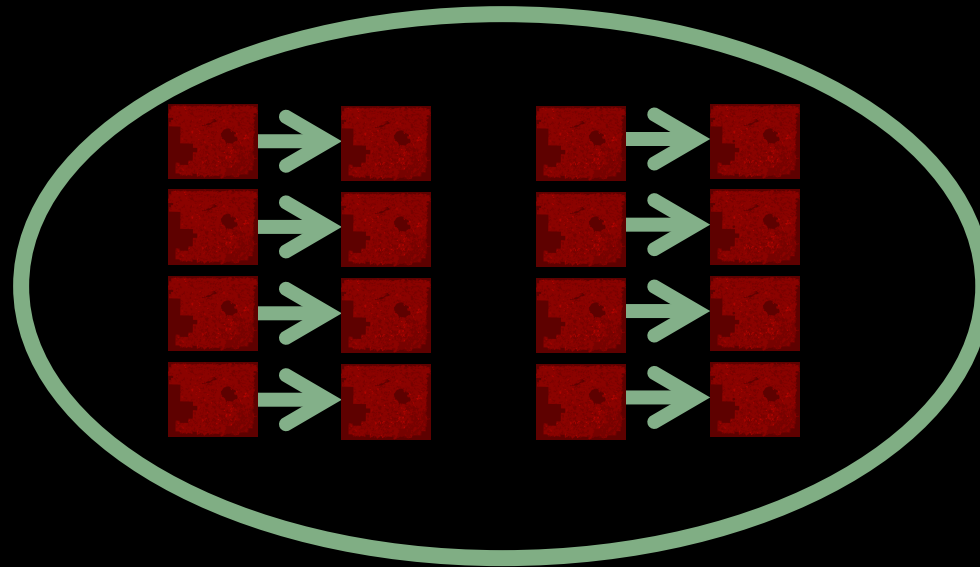


Constructing Database

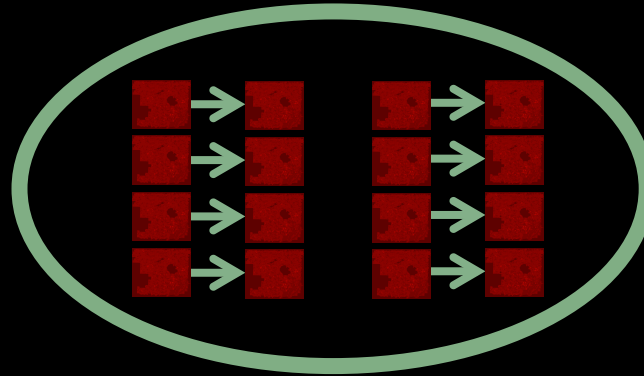


Constructing Database

Database



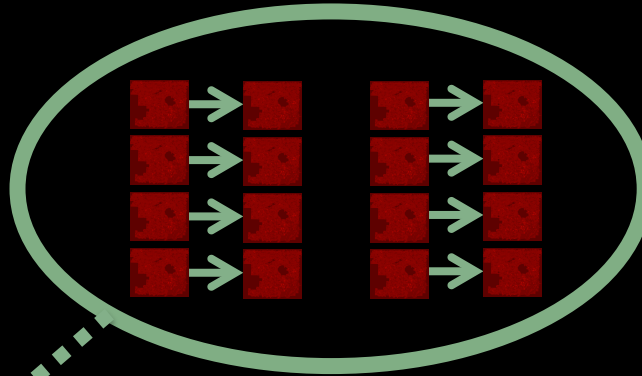
Fast Simulation



Draw Initial State



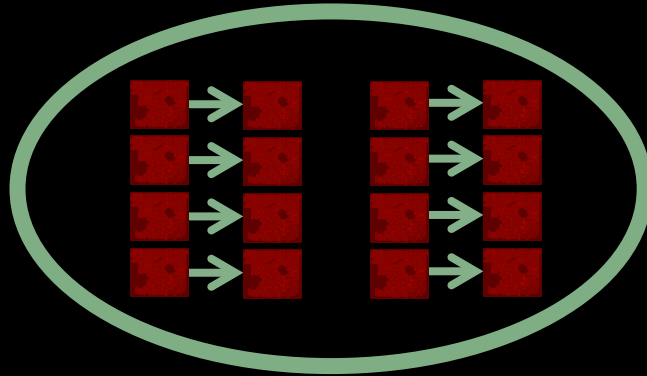
Fast Simulation



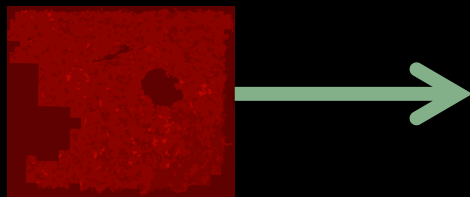
Switch to the most similar state in the database



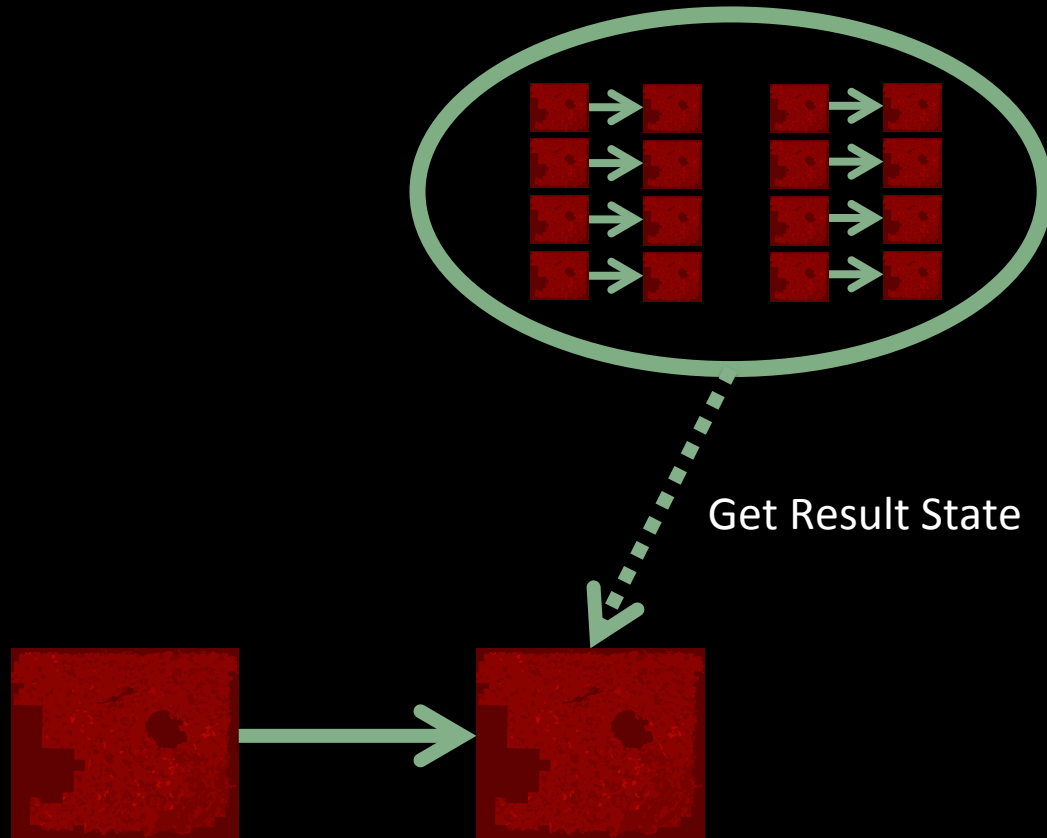
Fast Simulation



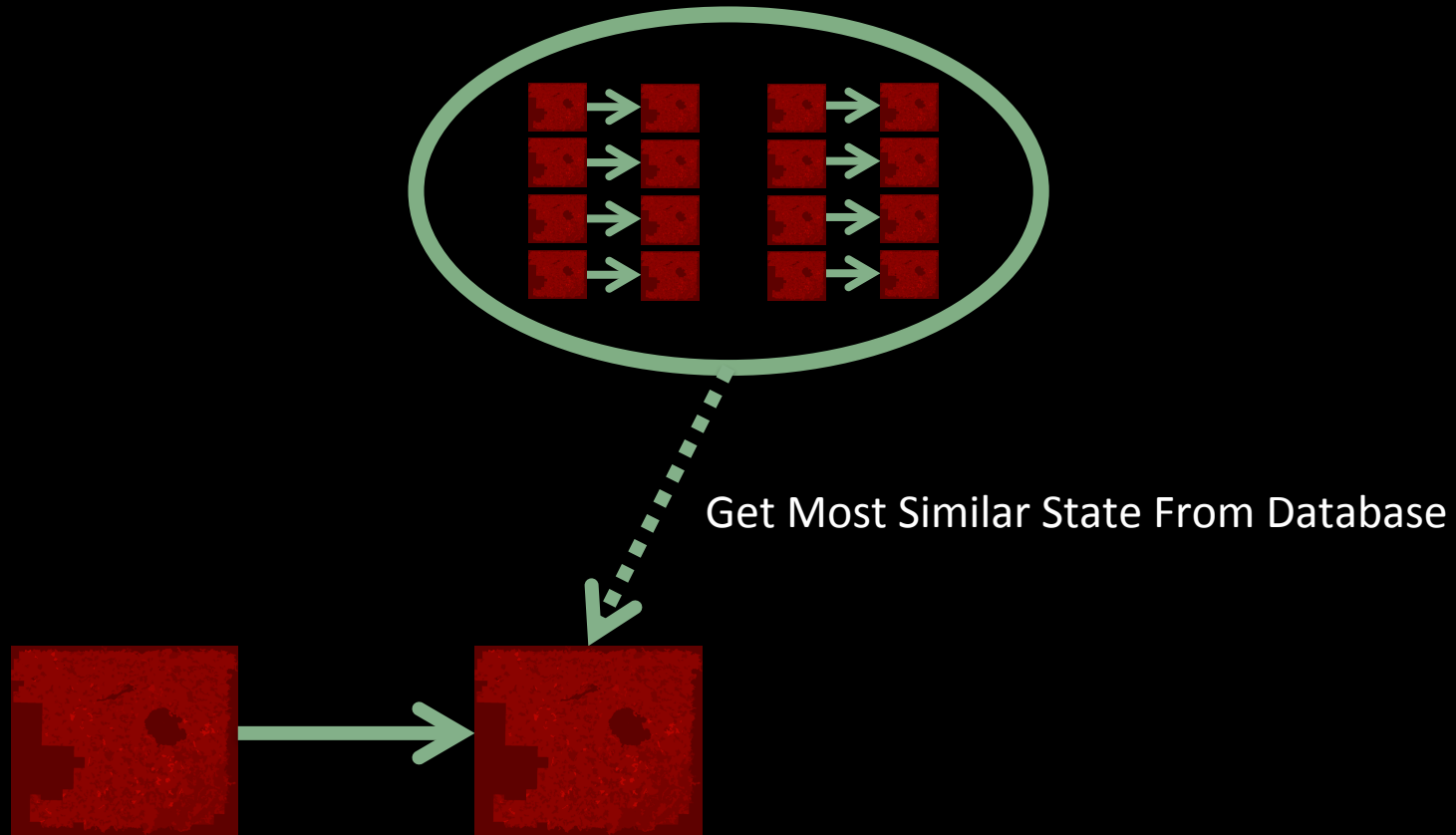
Apply Current Policy



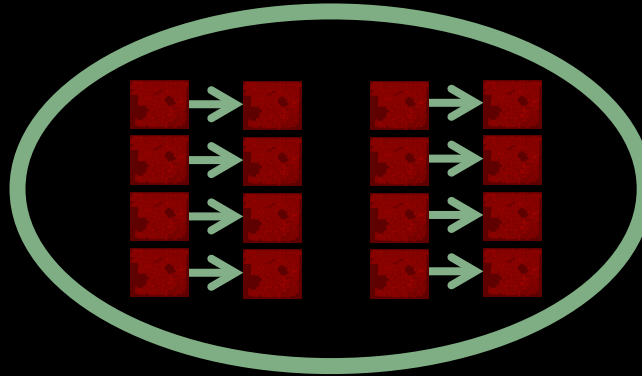
Fast Simulation



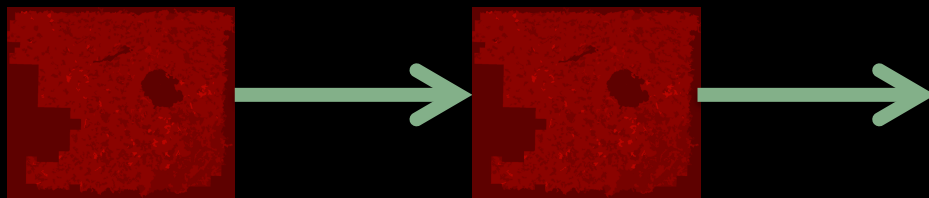
Fast Simulation



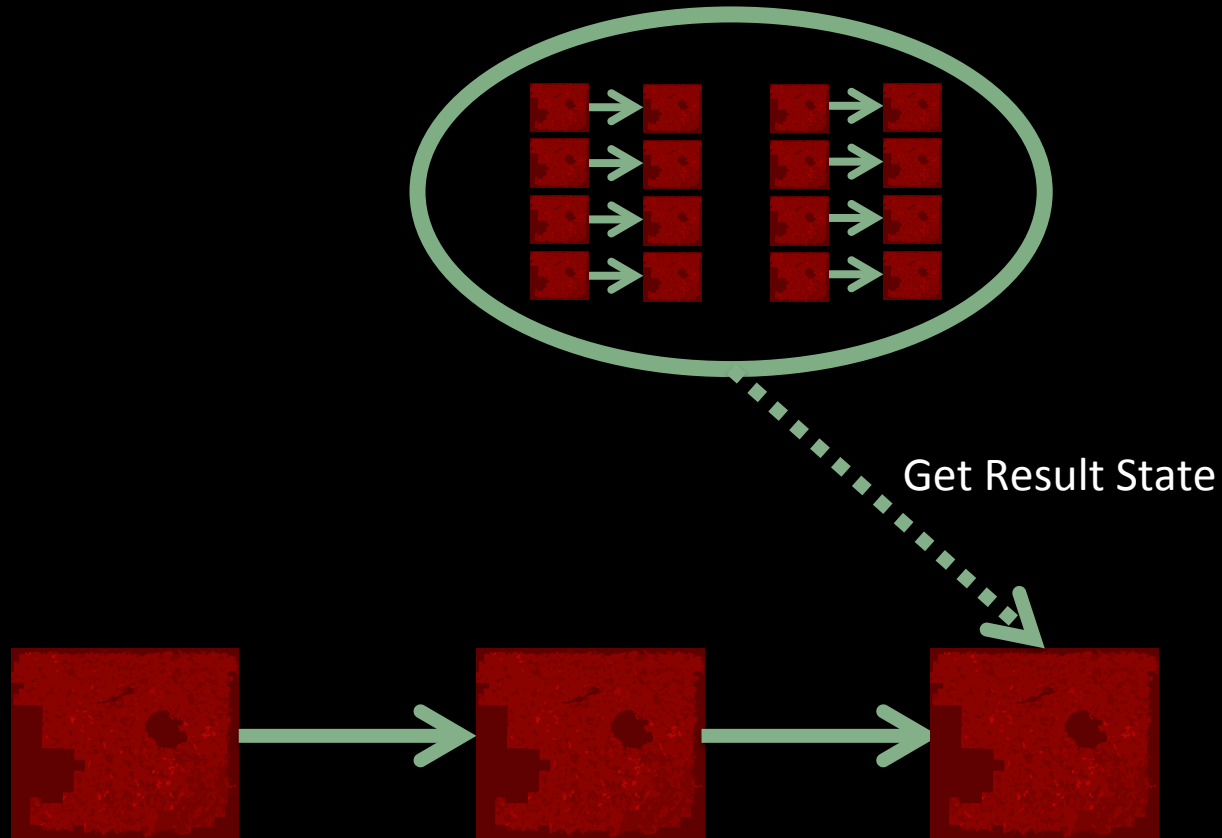
Fast Simulation



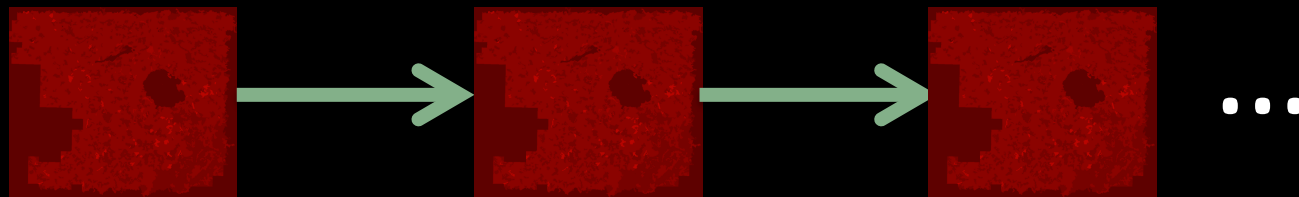
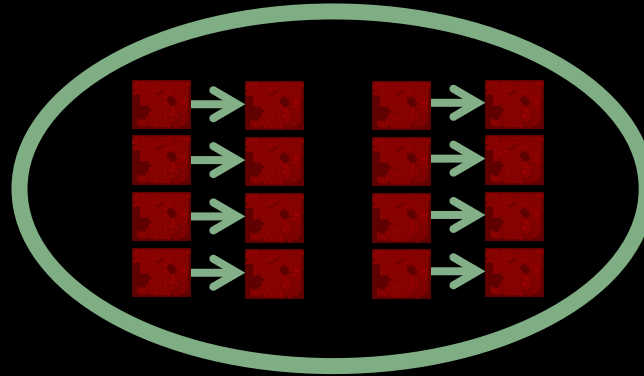
Apply Current Policy



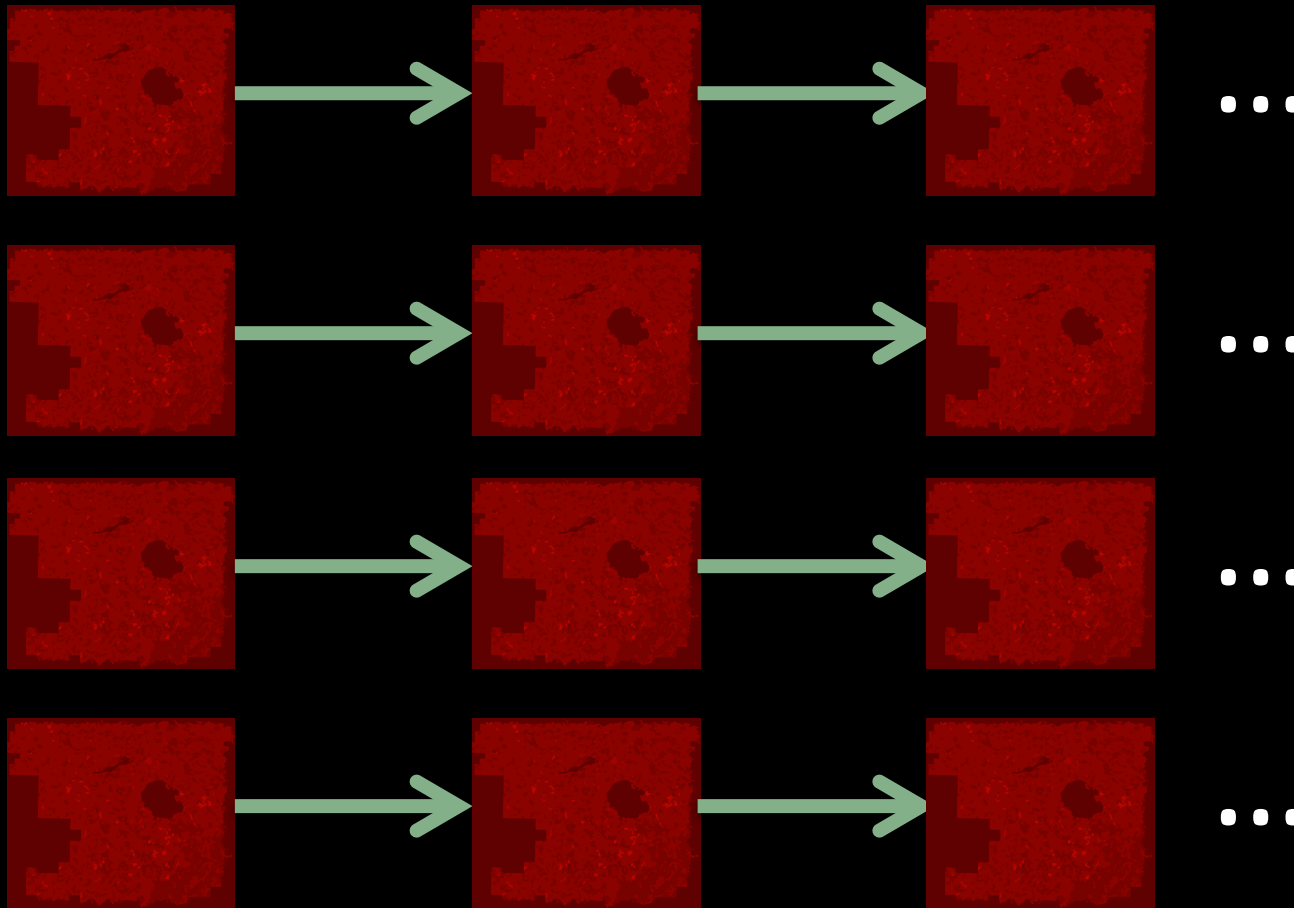
Fast Simulation



Fast Simulation

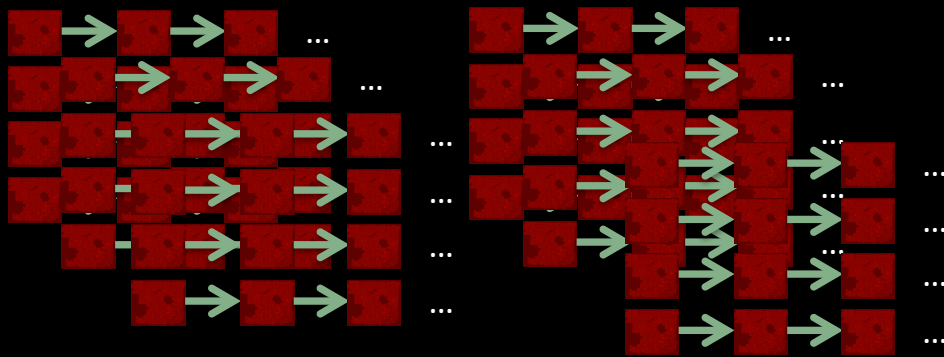


Fast Simulation

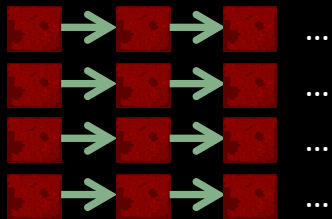


Fast Simulation

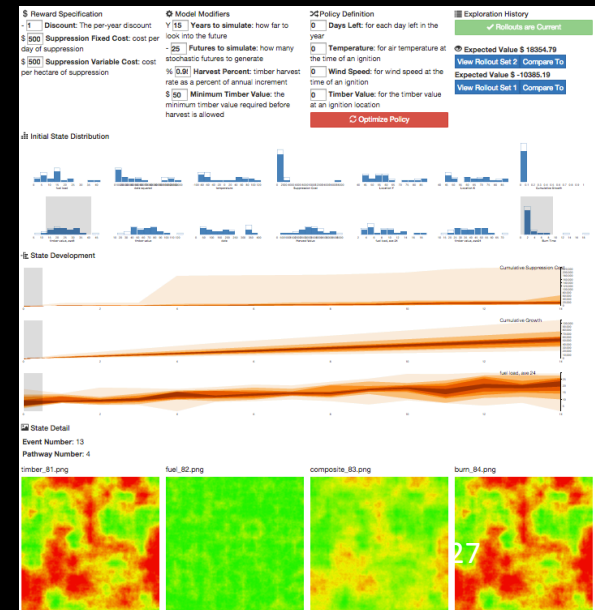
1. Optimize Over Many Sets of Trajectories



2. Generate a Final Set of Trajectories



3. Visualize!



Results

The screenshot shows a web browser window with the address bar set to localhost:8000. The main content is a modal window titled "MDPvis: Connect to a Simulator".

MDPvis: Connect to a Simulator

- Wildfire Simulator**
- Run Your Own Simulator**
 - If you start a server running on your computer that supports [CORS](#), you can connect this visualization to your domain without hosting the visualization code. The server examples we include in [MDPvis's code base](#) include CORS by default.
 - Open Connection to localhost:8938
 - Open Connection to localhost:8000
 - Open Connection to localhost:80
 - Open Connection to localhost:3000
- Custom Server**
 - Input field: `http://example.com:80`
 - Open Connection
- + Add Your Simulator to this List**
- i About**

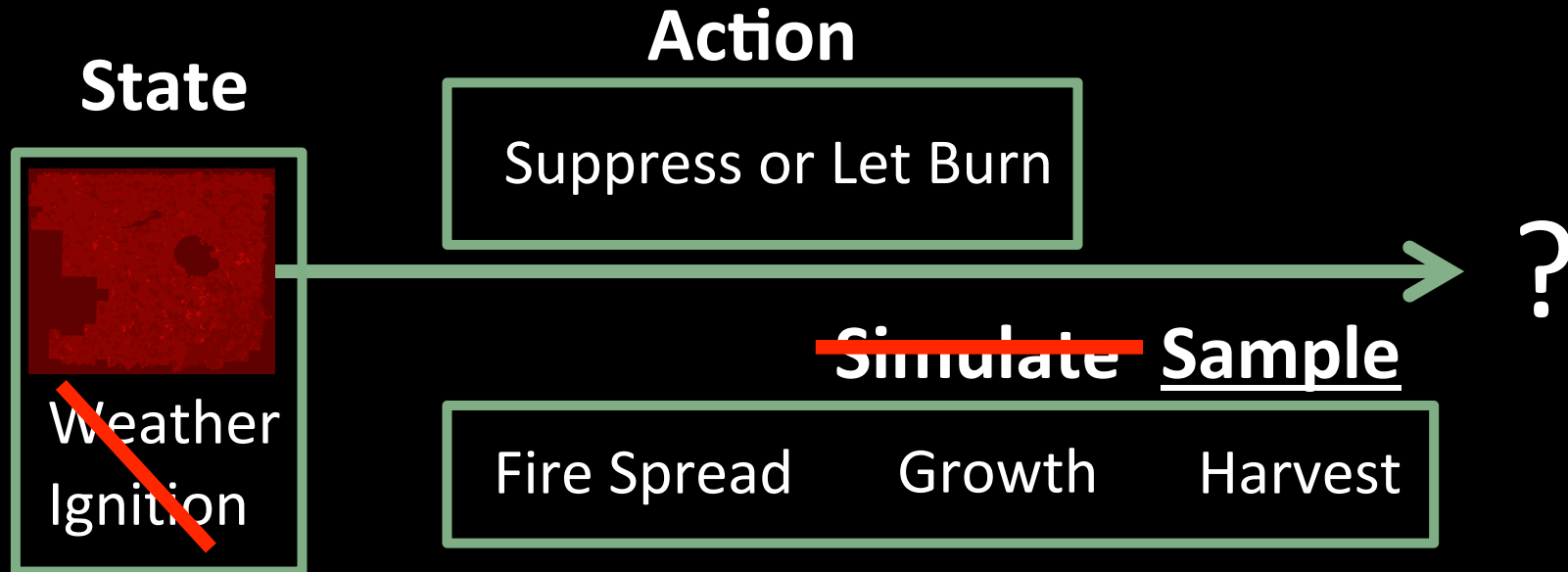
Modeling Trajectory Synthesis

Much simpler than learning a full predictive model

Step 1: Sample a policy space to create a dataset

Step 2: Create a similarity metric for states

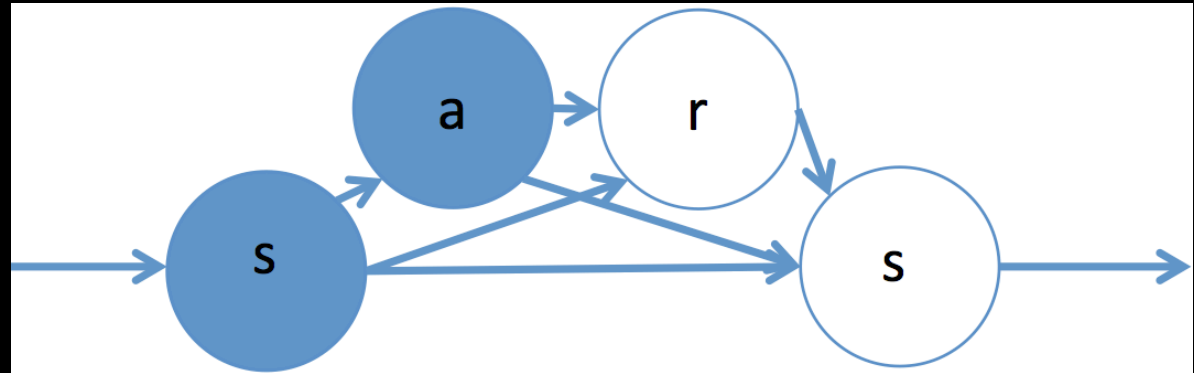
Similarity Metrics



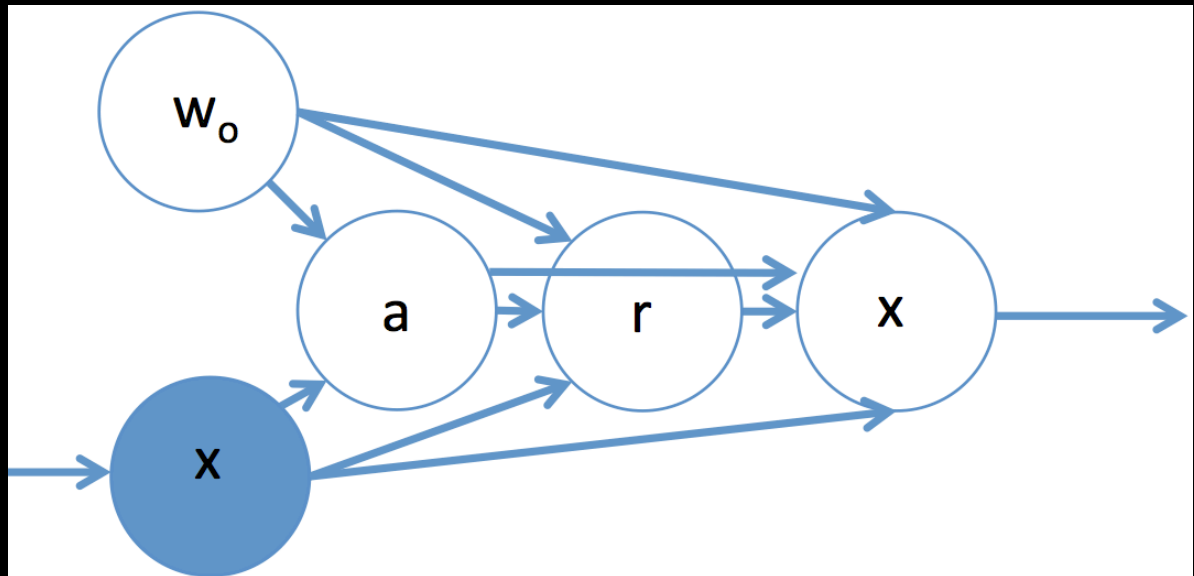
Eliminate Exogenous State!

S: Weather, Ignition Location, and Landscape

Standard State Transition



Factored State Transition



w_0 : Weather and Ignition Location
X: Landscapes

Fast Simulation for Computational Sustainability Sequential Decision Making Problems

Sean McGregor, Rachel Houtman, Hailey Buckingham, Claire Montgomery, Ronald Metoyer, and Thomas G. Dietterich

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 3 seconds, when the original simulator took several hours.

Markov Decision Processes

- A theoretical formulation for sequential decision making subject to uncertainty
 - Wildfire management!
 - Timber harvest planning
 - River flow management
 - Invasive species eradication?

More formally, a Markov Decision Process is

| | |
|------------------------|------------------------------|
| S | All States of the World |
| P_0 | Starting State Distribution |
| A | Available Actions |
| $R(s, a)$ | Rewards |
| $\gamma \in [0, 1]$ | Discount |
| P | State Transition Probability |
| $\pi(s) \rightarrow a$ | Policy |

A wildfire suppression Markov Decision Process is

| | |
|------------------------|--|
| S | All tree and weather configurations |
| P_0 | A snapshot of the current forest, with a random fire |
| A | Suppress or let-burn |
| $R(s, a)$ | Timber harvest, Suppression Expense |
| $\gamma \in [0, 1]$ | 0.96 (Forest Service Standard) |
| P | Several Simulators |
| $\pi(s) \rightarrow a$ | Suppress all fires |

Goal: Visualize and Optimize Decision Rules

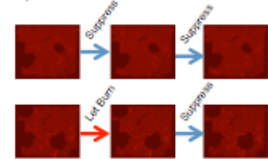
Wildfire: given a wildfire on timber producing lands, how do we balance suppression costs, timber revenues, and ecological services when deciding to suppress a fire or let it burn?

Problem: Simulating Nature is Computationally Expensive

Simulating ecological processes over many decades requires models for weather, climate, fire spread, human encroachment, succession, and more. These models can take hours or days to complete a single scenario!

Solution: Synthesize Trajectories from a Database

1. Generate a dataset of simulations from many different policies



2. When visualizing or optimizing a policy that has not been sampled, use state similarity to "stitch" states together into complete trajectories



3. Visualize^{1,2} or optimize based on the generated trajectories

MDPVis.github.io



State Variables in Computational Sustainability Domains

We model variables in the database differently based on whether they are **persistent** or **exogenous**. Persistent variables are highly correlated from one time step to another, but exogenous variables are independent and identically distributed within every time step.

Persistent

- Plant cover
- Fuel levels
- Species presence/absence
- Elevation, latitude, and longitude

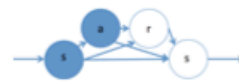
Exogenous

- Weather events
- Wildfire ignitions
- Invasive species introduction
- Timber prices

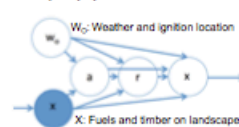
How do We Model Persistent and Exogenous Variables?

We stitch to a state in the database if it is similar to the state we are in. **Similarity does not need to include exogenous variables!** Traditionally, similarity is measured against the complete state and action as highlighted below. In our version of trajectory synthesis, we separate the action and exogenous variables and stitch based solely on the persistent state.

Standard Markov Decision Process (MDP) Transition



Our Trajectory Synthesis Transition



The probabilistic graphical models shown above factorize the state such that we can stitch states based solely on the configuration of a **persistent state**. We don't need to consider similarity of exogenous variables like weather!

Conclusion

We can visualize and optimize policies for computationally expensive sustainability domains with a database of state transitions whose computational cost is independent of the computational cost of the modeled phenomena.

References

- Houtman, R. M., Montgomery, C. A., Gagnon, A. R., Calkin, D. E., Dietterich, T. G., McGregor, S., & Crowley, M. (2015). Allowing a Wildfire to Burn: Estimating the Effect on Future Fire Suppression Costs. *International Journal of Wildland Fire*, 22(7), 871-882.
- Dietterich, T., Houtman, R., & Crowley, M. (2013). PAC Optimal Planning for Invasive Species Management: Improved Exploitation for Reinforcement Learning from Simulator-Defined MDPs. *Twenty-Seventh AAAI Conference on Artificial Intelligence*.
- Fortin, R., Murphy, S. A., Wetzel, L., & Sire, D. (2013). Batch Mode Reinforcement Learning based on the Synthesis of Artificial Trajectories. *Annals of Operations Research*, 208(1), 383-416.
- McGregor, S., Buckingham, H., Dietterich, T. G., Houtman, R., Montgomery, C., & Metoyer, R. (2015). Facilitating Testing and Debugging of Markov Decision Processes with Interactive Visualization. In *IEEE Symposium on Visual Languages and Human-Centric Computing*, Atlanta.
- McGregor, S., Buckingham, H., Houtman, R., Montgomery, C., Metoyer, R., & Dietterich, T. G. (2015). MDPVis: An Interactive Visualization for Testing Markov Decision Processes. In *AAAI Fall Symposium on Sequential Decision Making for Intelligent Agents*.



Oregon State
UNIVERSITY

Thanks!

Collaborators: Rachel Houtman, Hailey Buckingham, Claire Montgomery, Ronald Metoyer, Thomas Dietterich

Funder: NSF

Contact:

CompSust@seanbmgregor.com

Questions?