Multiscale Homogenization of Markov Decision Problems

Jake Bouvrie

Duke University
Department of Mathematics

Allerton - October 2 2012

Joint work with Mauro Maggioni

Goals

Given a stochastic decision making problem, i.e.

- planning / reinforcement learning
- stochastic control

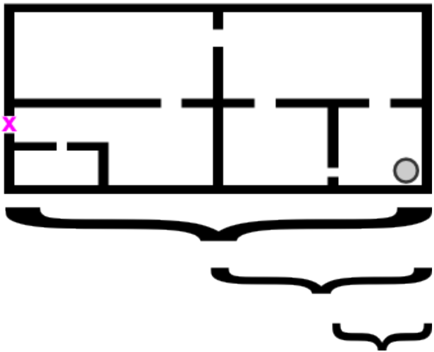
exploit multiscale structure, in order to:

- find a solution efficiently
  - localize computation
  - improve conditioning
- systematize knowledge transfer (see paper).

Cartoon

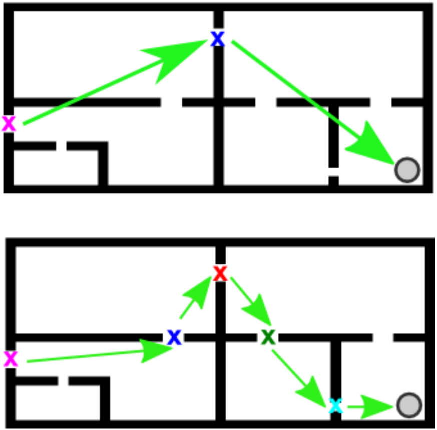
A multiscale planning problem: get from x to ○, with min. effort.

actions : $\{\texttt{left},\texttt{right},\texttt{up},\texttt{down}\}$; $\mathbb{P}(\text{action fails}) > 0$; Markov.



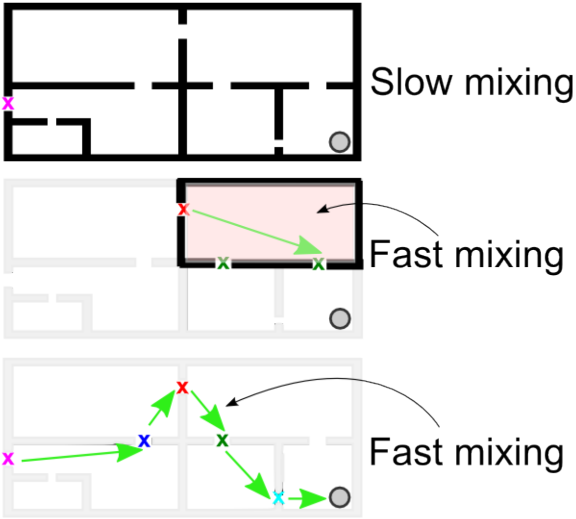
Cartoon - Hierarchical Decomposition

Localize computation by decomposing into small, independent sub-problems:



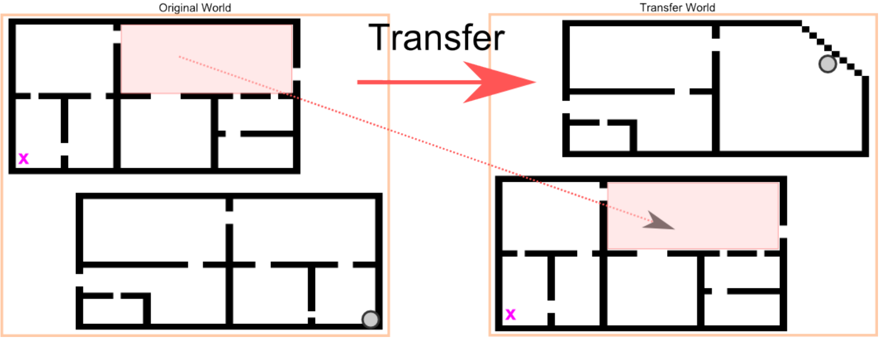
Cartoon - Conditioning

Improve conditioning:



Cartoon - Multiscale Transfer Learning

Identify transfer opportunities, encode knowledge, transfer knowledge:



Background: Discrete-Time Markov Decision Processes

MDP: a tuple (S, A, P, R, Γ) consisting of:

- A state space S (finite)
- An action (or "control") set A (finite)
- For $s, s' \in S, a \in A$, a transition probability tensor $P(s, a, s')$
- Reward function $R(s, a, s')$
- Collection of discount factors $\Gamma(s, a, s') \in (0, 1)$

$\mathcal{P}(A)$: set of all discrete probability distributions on A.

A **stationary stochastic policy** $\pi : S \to \mathcal{P}(A)$ is a function mapping states into distributions over the actions.

A policy (control law) specifies how to behave in the environment.

The Markov Chain P^π

Consider the stochastic state sequence $(s_t)_{t\geq 0}$ given by choosing controls $a_t \sim \pi(s_{t-1})$.

$(s_t)_{t\geq 0}$ is a homogeneous Markov chain with transition law

$$P^\pi(s, s') := \mathbb{E}_{a\sim\pi(s)}[P(s, a, s')]$$

MDP: Discounted Infinite-Horizon Value Functions

A **value function** $V^\pi : S \to \mathbb{R}$ assigns to each state s the expected sum of discounted rewards collected over an infinite horizon by running the policy π starting in s.

$$V^\pi(s) = \mathbb{E}\left[R(s_0, a_1, s_1)\right] + \mathbb{E}\left[\sum_{t=1}^{\infty}\left\{\prod_{\tau=0}^{t-1}\Gamma(s_\tau, a_{\tau+1}, s_{\tau+1})\right\} R(s_t, a_{t+1}, s_{t+1}) \;\middle|\; s_0 = s\right]$$

The expectation is taken over all sequences of state-action pairs $\{(s_t, a_t)\}_{t\geq 1}$,with $a_t \sim \pi(s_{t-1})$.

MDP: Discounted Infinite-Horizon Value Functions

Lemma

$$V^\pi(s) = \sum_{s',a} P(s,a,s')\pi(s,a)\big[R(s,a,s') + \Gamma(s,a,s')V^\pi(s')\big], s \in S.$$

In matrix-vector form,

$$V^\pi = \big(I - (\Gamma \circ P)^\pi\big)^{-1} r$$

where $r := (P \circ R)^\pi \mathbf{1}$.

The matrix $\big(I - (\Gamma \circ P)^\pi\big)^{-1}$ will be referred to as the *potential operator*.

MDP: Optimality

Goal is to find a policy (plan) that maximizes reward, given any starting state:

Optimal Solution

$$\pi^* := \arg\sup_{\pi \in \Pi} V^\pi$$

$$V^* := V^{\pi^*}$$

Π: Stochastic, stationary, Markov policies.

MDP: Solution Algorithms

Solving with off the shelf dynamic programming based methods:

- is expensive,
- scales poorly.

Example: Solve a sequence of $|S| \times |S|$ linear systems of the form $V^{\pi_k} = \left(I - (\Gamma \circ P)^{\pi_k}\right)^{-1} r^{\pi_k}$, $k = 0, 1, \ldots$

Multiscale Markov Decision Processes

Solving a problem with a **multiscale MDP hierarchy** consists of the following steps:

Step 1 **Partition** the state-space into subsets of states ("clusters") connected via "bottleneck" states.

Step 2 **Compress** or **homogenize** the MDP into another, smaller and *coarser* MDP, whose state space is the set of bottlenecks, and whose actions are given by following certain policies within clusters ("subtasks").

Repeat steps above with the compressed MDP as input, until desired number of compression steps, obtaining a hierarchy of MDPs.

Step 3 **Solve the hierarchy of MDPs** from the top-down (coarse to fine) by pushing solutions of coarse MDPs down to finer MDPs.

Multiscale Markov Decision Processes

MMDP Goals

- Localize computation: decompose a complex task into a hierarchy of simpler sub-tasks.
- Improve conditioning:
  - solve small "fast mixing" problems
  - precondition/shape with coarse solution
- Systematize knowledge transfer

Step (1): Statespace Partitioning

Example: Recursive Spectral Partitioning

1. Set $P^\pi_{\text{tel}} = (1-\eta)P^\pi + \eta n^{-1}\mathbf{1}\mathbf{1}^\top$, for some small, positive η.
2. Find the invariant distribution μ satisfying $(P^\pi_{\text{tel}})^\top\mu = \mu$.
3. Let $\Phi = \text{diag}(\mu)$ and compute the symmetrized Laplacian for directed graphs (Chung, '05):

$$L = \Phi - \tfrac{1}{2}\left(\Phi P^\pi + (P^\pi)^\top\Phi\right)$$

4. Find low-conductance cuts from K smallest nontrivial eigenvectors of L.
5. Repeat on resulting subgraphs.

Other possibilities exist: local heat flux, evolving sets, "betweenness",...

Note: Partitioning/bottlenecks *depend on* π. Can be the *diffusion policy* or can encode problem-specific goal information (e.g. reward).

Statespace Partitioning: Clusters

$\mathcal{B}^\pi$: Bottleneck states resulting from cuts, *plus absorbing states*.

Partitioning of $\{S \setminus \mathcal{B}^\pi\}$ is given by $S/\sim$, under

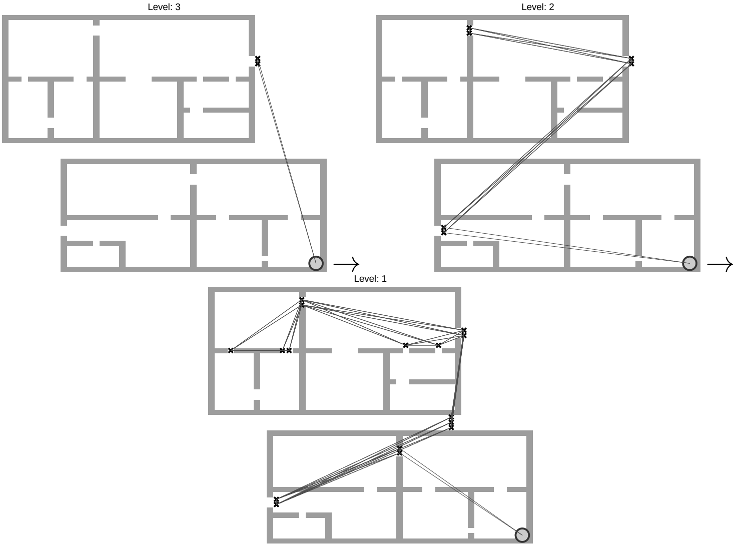
$$s_i \sim s_j, \quad \text{if } s_i, s_j \notin \mathcal{B}^\pi \text{ and there is a path from } s_i \text{ to } s_j$$
$$\text{not passing through any b} \in \mathcal{B}^\pi.$$

A **cluster** is an equivalence class $[s]$ plus any bottleneck states P^π-connected to states in the class.

interior: $\mathring{\text{c}}{:=}\ [s]$
boundary: $\partial\text{c} := $ bottlenecks attached to $[s]$

Clusters of $G = (S, P^\pi)$ only connect to each other via bottlenecks.

Cartoon: Coarse Statespace Graphs



Step (2): Multiscale Compression

Given a policy π_{c} on cluster c, consider the Markov chain $(X_t)_{t\geq 0}$ with transition matrix $P_{\mathsf{c}}^{\pi_{\mathsf{c}}}$, P restricted to c along π_{c}.

Define the hitting times of $\partial\mathsf{c}$:

$$T_m = \inf\{t > T_{m-1} \mid X_t \in \partial\mathsf{c}\}, \qquad m = 1, 2, \ldots$$

with $T_0 = \inf\{t \geq 0 \mid X_t \in \partial\mathsf{c}\}$. $(\mathbb{P}_s(T_m < \infty) = 1, \forall s \in \mathsf{c}, m)$

Intuition

Compression: summarize what happens between successive hitting times.

Computations are all local (one cluster at a time)...

Multiscale Compression: Coarse MDP

A homogenized MDP consists of the tuple $(\widetilde{S}, \widetilde{A}, \widetilde{P}, \widetilde{R}, \widetilde{\Gamma})$.

There are a few ways to summarize the fine scale MC:

- analytically (e.g. mean-field approx.), if the model (or an estimate) is known;
- by Monte-Carlo simulations/exploration;
- combinations of the two.

Multiscale Compression: Coarse Statespace

A homogenized MDP consists of the tuple $(\widetilde{S}, \widetilde{A}, \widetilde{P}, \widetilde{R}, \widetilde{\Gamma})$.

- **Statespace** $\widetilde{S}$: The coarse scale statespace $\widetilde{S}$ is the set of bottleneck states $\mathcal{B}$ for the fine scale.

  Note that $\widetilde{S} \subset S$, and we can expect $|\widetilde{S}| \ll |S|$.

Multiscale Compression: Coarse Actions

A homogenized MDP consists of the tuple $(\widetilde{S}, \widetilde{A}, \widetilde{P}, \widetilde{R}, \widetilde{\Gamma})$:

- **Action set** $\widetilde{A}$: A coarse action invoked from b $\in \widetilde{S} = \mathcal{B}$ consists of executing a given fine scale policy $\pi_{\mathsf{c}} \in \boldsymbol{\pi}_{\mathsf{c}}$ within the fine scale cluster c, starting from b $\in \partial$c (at a time that we may reset to 0), until the first positive time at which a bottleneck state in ∂c is hit.

Multiscale Compression: Coarse Transition Kernel

A homogenized MDP consists of the tuple $(\widetilde{S}, \widetilde{A}, \widetilde{P}, \widetilde{R}, \widetilde{\Gamma})$:

- **Coarse scale transition probabilities** $\widetilde{P}(s, a, s')$: If $a \in \widetilde{A}$ is an action executing the policy $\pi_{\mathsf{c}} \in \boldsymbol{\pi}_{\mathsf{c}}$, then $\widetilde{P}(s, a, s')$ is defined as the probability that the Markov chain $P_{\mathsf{c}}^{\pi_{\mathsf{c}}}$ started from $s \in \widetilde{S}$, hits $s' \in \widetilde{S}$ before hitting any other bottleneck.

Multiscale Compression: Coarse Transition Kernel

If P or an estimate of P is known:

Proposition

Let a be the coarse action corresponding to executing a policy π_{c} in cluster c. Then

$$\widetilde{P}(s,a,s') = H_{s,s'}, \qquad \textit{for all } s, s' \in \partial\mathsf{c},$$

where H is the minimal non-negative solution, for each $s' \in \partial\mathsf{c}$, to the linear system

$$H_{s,s'} = P_{\mathsf{c}}^{\pi_{\mathsf{c}}}(s,s') + \sum_{s''\in\mathring{\mathsf{c}}} P_{\mathsf{c}}^{\pi_{\mathsf{c}}}(s,s'')H_{s'',s'}, \quad s \in \mathsf{c}, s' \in \partial\mathsf{c}\,.$$

Multiscale Compression: Coarse Rewards

A homogenized MDP consists of the tuple $(\widetilde{S}, \widetilde{A}, \widetilde{P}, \widetilde{R}, \widetilde{\Gamma})$:

- **Coarse scale rewards** $\widetilde{R}(s, a, s')$: The coarse reward $\widetilde{R}(s, a, s')$ is defined to be the sum of discounted rewards collected along trajectories of the (fine) Markov chain associated to coarse action a, which start at $s \in \widetilde{S}$ and end by hitting $s' \in \widetilde{S}$ before hitting any other bottleneck.

Multiscale Compression: Coarse Discount Factors

A homogenized MDP consists of the tuple $(\widetilde{S}, \widetilde{A}, \widetilde{P}, \widetilde{R}, \widetilde{\Gamma})$:

- **Coarse scale discount factors** $\widetilde{\Gamma}(s, a, s')$: The coarse discount factor $\widetilde{\Gamma}(s, a, s')$ is the product of the discounts applied to rewards along trajectories of the Markov chain $P_{\mathsf{c}}^{\pi_{\mathsf{c}}}$ associated to a action $a \in \widetilde{A}$, starting at $s \in \widetilde{S}$ and ending at $s' \in \widetilde{S}$.

Multiscale Compression: Mean-field Approximations

Given stopping times $0 \leq T < T' < \infty$ (a.s.):

$$\Delta_T^{T'} := \prod_{t=T}^{T'-1} \Gamma\big(X_t, a_{t+1}, X_{t+1}\big)$$

$$R_T^{T'} := R(X_T, a_{T+1}, X_{T+1}) + \sum_{t=T+1}^{T'-1} \Delta_T^t R\big(X_t, a_{t+1}, X_{t+1}\big)$$

Approximate $R_{T_0}^{T_1}, \Delta_{T_0}^{T_1}$ by the conditional expectations:
$\mathbb{E}[R_0^{T_1} \mid X_0 = s, X_{T_1} = s'], \qquad \mathbb{E}[\Delta_0^{T_1} \mid X_0 = s, X_{T_1} = s'].$

$\Rightarrow$ **Linear systems.**

$\Rightarrow$ Total cost is at most: $\mathcal{O}\big(|\partial \mathsf{c}||\mathring{\mathsf{c}}|^3 + |\partial \mathsf{c}|^2|\mathring{\mathsf{c}}|\big)$ per cluster.

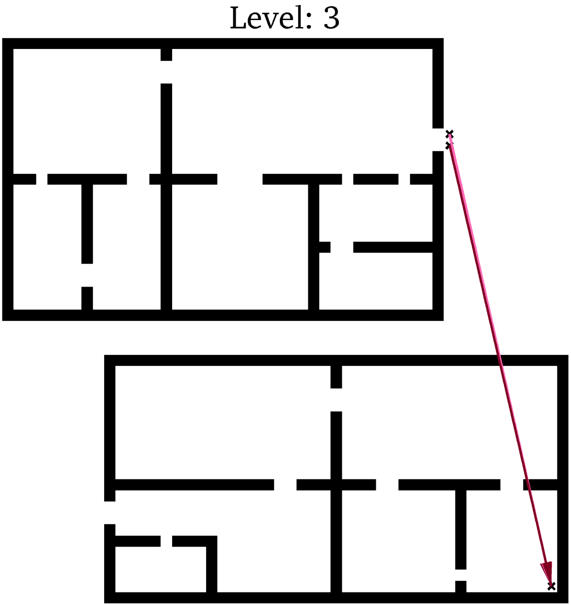
Proof: Doob-*like* h-transforms + strong Markov property.

Multiscale MDPs

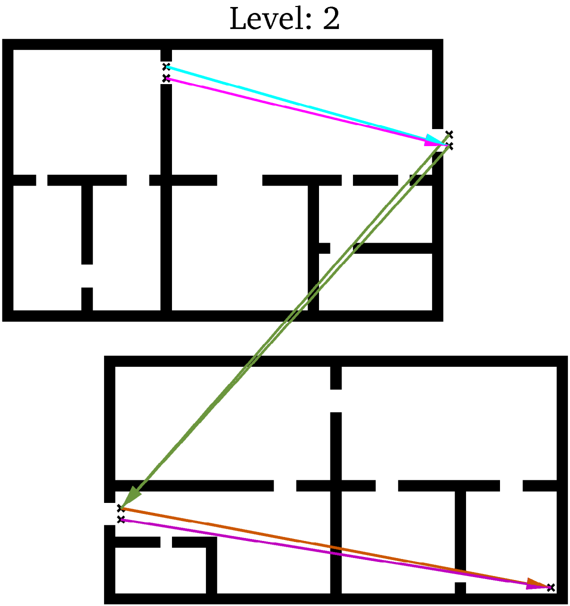
Consider a multiscale hierarchy of MDPs (MMDP) defined in this way:

- The MMDP is *consistent* in the mean across scales.
- Each scale is an *independent, deterministic* MDP, that can be solved using any algorithm.
- Coarse MDPs are *small*.
- Clusters may be interpreted as *sub-tasks*, or macro-actions.

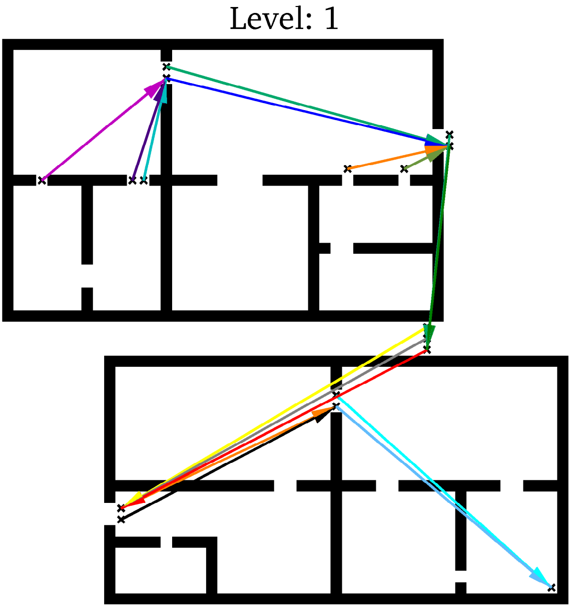
Example coarse policies...


Level: 3

Multiscale Solution of MDPs - Coarse Policies



Multiscale Solution of MDPs - Coarse Policies



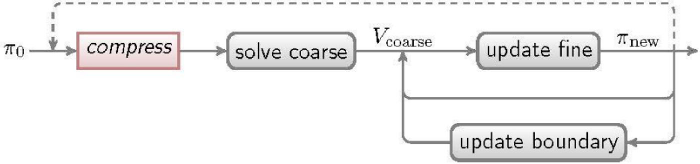
Step (3): Multiscale Solution of MDPs

General Idea

Alternate:

(i) Update fine solution on clusters *independently* given coarse solution (update interiors).

(ii) Update coarse solution given fine solution (update boundary).



Different solution algorithms for solving a pair of coarse/fine MDPs are obtained by iterating over different paths in this flow graph.

Multiscale Solution of MDPs: Interior Updates

Local fine scale update on c **given coarse solution** $V_{\textsf{coarse}}$**:**

For Example: Solve a (Poisson) BVP

Let $(X_t)_{t\geq 0} \sim P_{\mathsf{c}}^{\pi_{\mathsf{c}}}$. We would like to compute

$$V(s) := \mathbb{E}\big[R_0^T + \Delta_0^T V_{\text{coarse}}(X_T) \mid X_0 = s\big], \quad s \in \mathring{\mathsf{c}}$$

where $T := \inf\{n \geq 0 \mid X_n \in \partial\mathsf{c}\}$ is the first passage time of the boundary:

$$V(s) = \begin{cases} V_{\text{coarse}}(s) & \text{if } s \in \partial\mathsf{c} \\ \displaystyle\sum_{s' \in \mathsf{c}, a' \in A} P_{\mathsf{c}}(s,a,s')\pi_{\mathsf{c}}(s,a)\big[R(s,a,s') + \Gamma(s,a,s')V(s')\big] & \text{if } s \in \mathring{\mathsf{c}} \end{cases}$$

$V(s)$ is unique and bounded under mild boundary reachability assumptions.

Each BVP is independent of the others given $V_{\textsf{coarse}}$**.**

Multiscale Solution of MDPs: Boundary Update

Boundary update on $\mathcal{B}$ given V:

- Local averaging. For $s \in \mathcal{B}$,

$$V_{\textsf{coarse}}(s) \leftarrow \sum_{s' \sim s,a} P(s,a,s')\pi(s,a)\big(R(s,a,s')+\Gamma(s,a,s')V(s')\big)$$

- Value determination
- *Recompression* with respect to a regularized, greedy policy corresponding to current fine V.

Multiscale Solution of MDPs: An Algorithm

Combining these steps,

A Two-Scale Iteration

Compress the fine MDP. Solve the coarse MDP.

1. Solve local boundary value problems, given current π on interior, V on boundary.
2. Update the policy.
3. Update boundary by local averaging, given current π, V.
4. Repeat from (1).

Multiscale Solution of MDPs: Convergence

This particular algorithm is a form of *asynchronous modified policy iteration*.

Theorem

Fix any initial fine-scale (π_0, V^0). *For an appropriate number of bottleneck updates per iteration,*

$$N > \log_{\bar{\gamma}} \tfrac{1}{2}$$

with $\bar{\gamma} := \max_{s,a,s'}\left\{\Gamma(s,a,s')\mathbf{1}_{[P(s,a,s')>0]}\right\}$, *the alternating interior-boundary policy iteration algorithm satisfies*

$$\lim_{k\to\infty} \sup_{s\in S} |V^*(s) - V^k(s)| \to 0$$

and hence converges to the optimal fine scale policy π^*.

Multiscale Solution of MDPs: Complexity

If at a scale j there are r_j clusters of roughly equal size, and n_j states, the solution of the MDP at that scale may be computed in time $\mathcal{O}\big(r_j(n_j/r_j)^3\big)$.

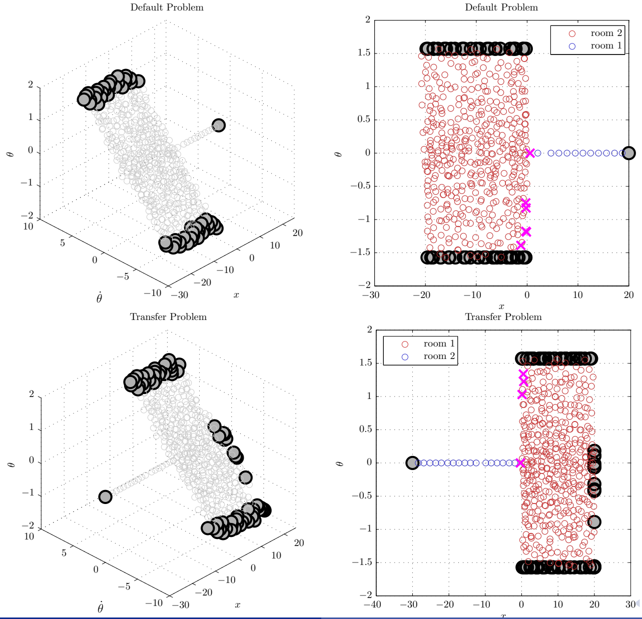
If $r_j = n_j/C$ and $n_j = n/C^j$ (with n the size of the original state space), then the computation time across $\log n$ scales is $\mathcal{O}\big(n \log n\big)$.

Transfer Learning (Briefly)

Given a pair of problems $(\mathrm{MMDP}_{(1)}, \mathrm{MMDP}_{(2)})$, the first of which is solved, transfer to the second.

1. Match sub-tasks at any scale.
2. Transfer a policy, value function, or potential operator between *clusters*.
3. Use transferred data as an initial conditions to solve for remainder of $\mathrm{MMDP}_{(2)}$.

Example: Continuous Control Task



Summary

Overarching themes:

- Multiscale as a unifying, organizational principle:
  - decomposition of tasks into sub-tasks
  - each scale (MDP) may be considered independently of the others; is consistent with others.
- Computational efficiency
  - localization
  - conditioning
- Tight coupling between structure discovery, learning, and planning
- Transfer: MMDPs support multiscale transfer of sub-task solutions between related problems.