Renewal Monte Carlo:

Renewal theory based reinforcement learning

Jayakumar Subramanian and Aditya Mahajan

57th IEEE Conference on Decision and Control, Miami Beach, FL, USA,
December 17-19, 2018

RL has achieved considerable success…

Image credit: MIT Technology review

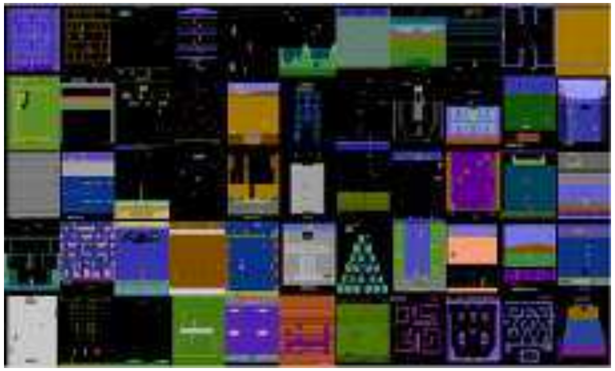


Image credit: Towards Data Science

Image credit: Popular Science

RL has achieved considerable success...

Image credit: MIT Technology review

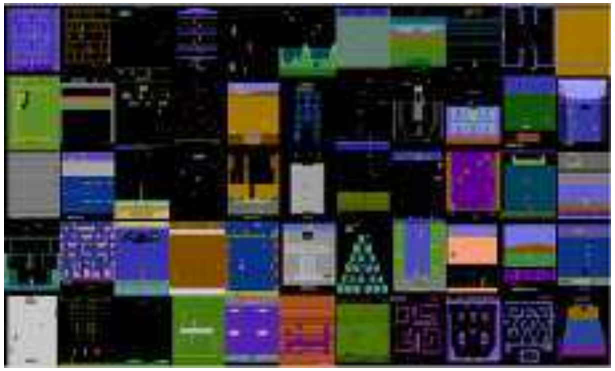


Image credit: Towards Data Science

Image credit: Popular Science

Salient features

- ⊕ Model-free method
- ⊕ Use policy search

RL has achieved considerable success...

Image credit: MIT Technology review

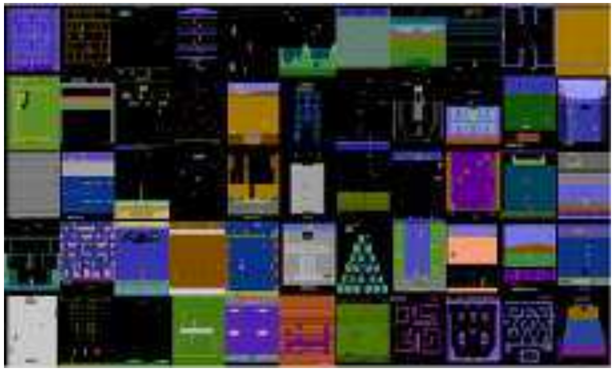


Image credit: Towards Data Science

Image credit: Popular Science

Salient features

⊕ Model-free method

⊕ Use policy search

Limitation

⊖ Learning is slow (takes ~ 10^9 to 10^15 iterations to converge)

RL has achieved considerable success…

Image credit: MIT Technology review

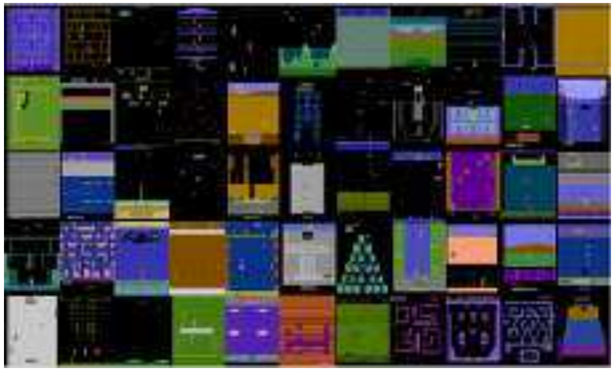


Image credit: Towards Data Science

Image credit: Popular Science

Salient features

- ⊕ Model-free method
- ⊕ Use policy search

Limitation

- ⊖ Learning is slow (takes ~ 10^9 to 10^15 iterations to converge)

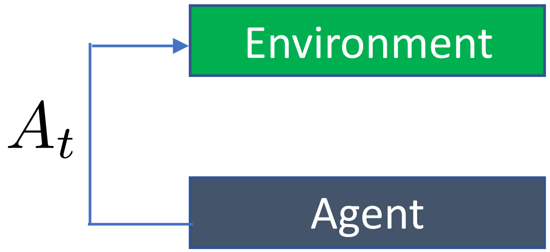
- ⊕ Can we exploit features of the model to make it learn faster? …
- ⊕ Without sacrificing generality?

An RL problem can be formulated as…

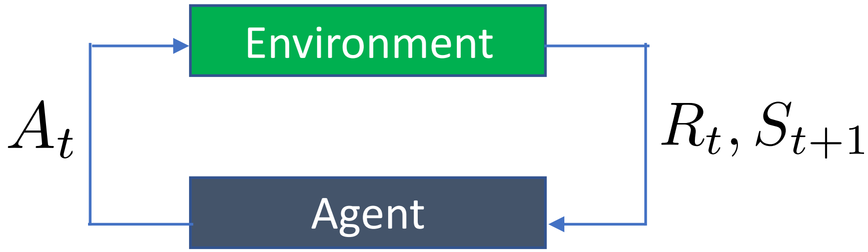
An RL problem can be formulated as…

Agent

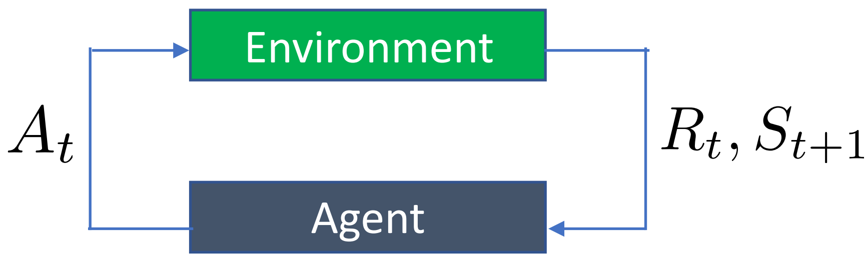
An RL problem can be formulated as…



An RL problem can be formulated as…

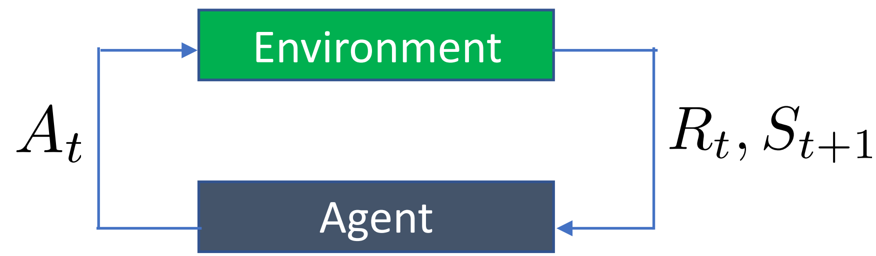


An RL problem can be formulated as…



Infinite horizon Markov decision process (MDP) Model	
State space	$S_t \in \mathcal{S}$
Action space	$A_t \in \mathcal{A}$
Transition probability	$\mathbb{P}(S_{t+1} \vert S_t, A_t) = [P(A_t)]_{S_t, S_{t+1}}$
Per-step reward	$R_t = r(S_t, A_t, S_{t+1})$

An RL problem can be formulated as…



Infinite horizon Markov decision process (MDP) Model	
State space	$S_t \in \mathcal{S}$
Action space	$A_t \in \mathcal{A}$
Transition probability	$\mathbb{P}(S_{t+1} \vert S_t, A_t) = [P(A_t)]_{S_t, S_{t+1}}$
Per-step reward	$R_t = r(S_t, A_t, S_{t+1})$

Unknown in RL

Policy parametrization

Policy parametrization

μ_θ is a parametrized policy

Policy parametrization

μ_θ is a parametrized policy

Gibbs (softmax) policy

$$\mu_\theta(a|s) = \frac{\exp(\tau\theta(s,a))}{\sum_{a'} \exp(\tau\theta(s,a))}$$

Policy parametrization

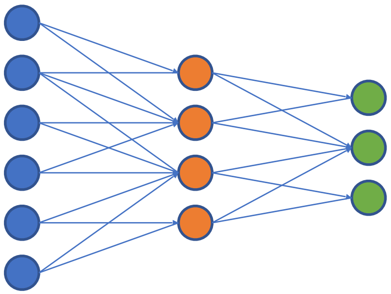
μ_θ is a parametrized policy

Gibbs (softmax) policy

$$\mu_\theta(a|s) = \frac{\exp(\tau\theta(s,a))}{\sum_{a'}\exp(\tau\theta(s,a))}$$

Neural network (NN) policy

$$\mu_\theta(a|s) =$$



θ: weights of NN

Policy gradient

Policy gradient

Performance Gradient Estimate

Policy gradient

Performance Gradient Estimate

$$J_\theta = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t R_t \,|\, S_0 = s_0, A_t \sim \mu_\theta(S_t)\Big]$$

Policy gradient

Performance Gradient Estimate

$$J_\theta = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t R_t \,|\, S_0 = s_0, A_t \sim \mu_\theta(S_t)\Big]$$

G_θ is an estimate of $\nabla_\theta J_\theta$

Policy gradient

Performance Gradient Estimate

$$J_\theta = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t R_t \big| S_0 = s_0, A_t \sim \mu_\theta(S_t)\Big]$$

G_θ is an estimate of $\nabla_\theta J_\theta$

Stochastic Gradient Ascent

Policy gradient

Performance Gradient Estimate

$$J_\theta = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t R_t \,|\, S_0 = s_0, A_t \sim \mu_\theta(S_t)\Big]$$

G_θ is an estimate of $\nabla_\theta J_\theta$

Stochastic Gradient Ascent

$$\theta_{k+1} = [\theta_k + \alpha_k G_{\theta_k}]_\Theta$$

Policy gradient

Performance Gradient Estimate

$$J_\theta = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t R_t \big| S_0 = s_0, A_t \sim \mu_\theta(S_t)\Big]$$

G_θ is an estimate of $\nabla_\theta J_\theta$

Stochastic Gradient Ascent

$$\theta_{k+1} = [\theta_k + \alpha_k G_{\theta_k}]_\Theta$$

$$\sum_{k=0}^{\infty} \alpha_k = \infty \quad \text{and} \quad \sum_{k=0}^{\infty} \alpha_k^2 < \infty$$

Policy gradient

Performance Gradient Estimate

$$J_\theta = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t R_t \,|\, S_0 = s_0, A_t \sim \mu_\theta(S_t)\Big]$$

G_θ is an estimate of $\nabla_\theta J_\theta$

How do we estimate this?

Stochastic Gradient Ascent

$$\theta_{k+1} = [\theta_k + \alpha_k G_{\theta_k}]_\Theta$$

$$\sum_{k=0}^{\infty} \alpha_k = \infty \quad \text{and} \quad \sum_{k=0}^{\infty} \alpha_k^2 < \infty$$

How to estimate $\nabla_\theta J_\theta$?

How to estimate $\nabla_\theta J_\theta$?

Monte Carlo estimate (REINFORCE)

$$G_\theta = \sum_{t=0}^{\infty} \Big[\nabla_\theta \log(\mu_\theta(A_t | S_t)) \gamma^t \Big(\sum_{n=0}^{\infty} \gamma^n R_n \Big) \Big]$$

How to estimate $\nabla_\theta J_\theta$?

Monte Carlo estimate (REINFORCE)

$$G_\theta = \sum_{t=0}^{\infty} \Big[\nabla_\theta \log(\mu_\theta(A_t|S_t)) \gamma^t \Big(\sum_{n=0}^{\infty} \gamma^n R_n \Big) \Big]$$

Actor Critic estimate (Temporal difference / SARSA)

$$G_\theta = \sum_{t=0}^{\infty} \Big[\nabla_\theta \log(\mu_\theta(A_t|S_t)) \gamma^t Q(S_t, A_t) \Big]$$

How to estimate $\nabla_\theta J_\theta$?

Monte Carlo estimate (REINFORCE)

$$G_\theta = \sum_{t=0}^{\infty} \left[\nabla_\theta \log(\mu_\theta(A_t | S_t)) \gamma^t \left(\sum_{n=0}^{\infty} \gamma^n R_n \right) \right]$$

Actor Critic estimate (Temporal difference / SARSA)

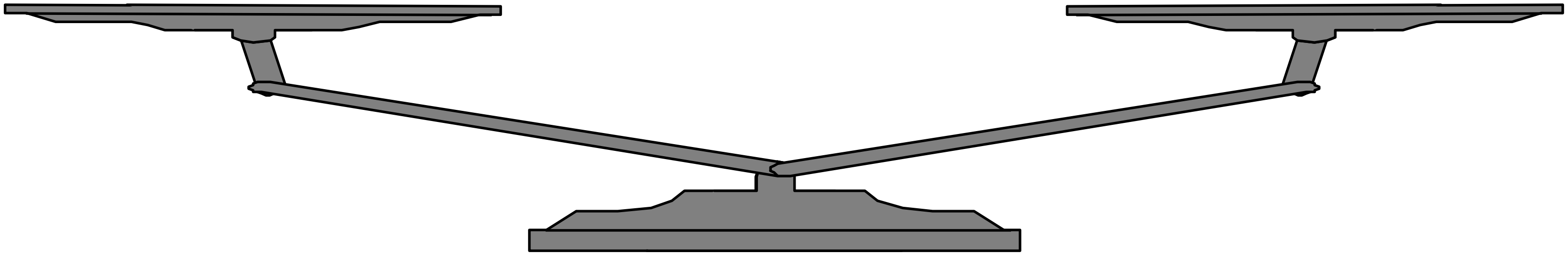
$$G_\theta = \sum_{t=0}^{\infty} \left[\nabla_\theta \log(\mu_\theta(A_t | S_t)) \gamma^t Q(S_t, A_t) \right]$$

Actor Critic with eligibility traces estimate (SARSA-λ)

$$G_\theta = \sum_{t=0}^{\infty} \left[\nabla_\theta \log(\mu_\theta(A_t | S_t)) \gamma^t Q^\lambda(S_t, A_t) \right]$$

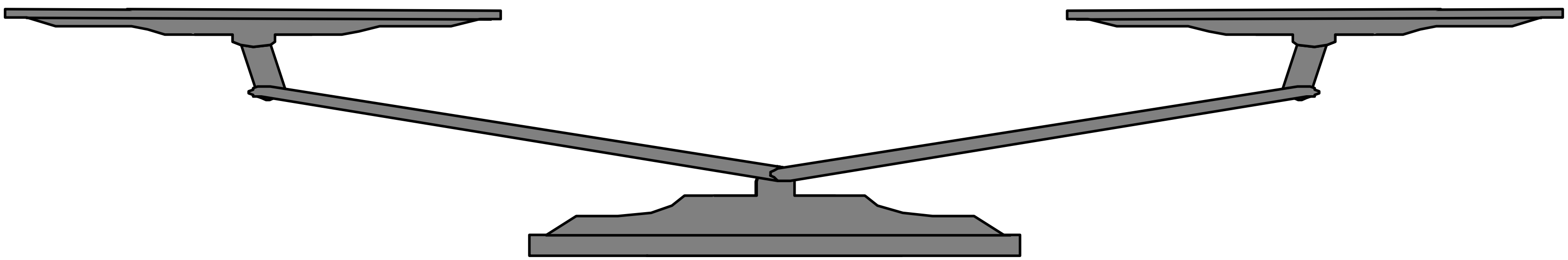
MC vs. TD

MC	TD



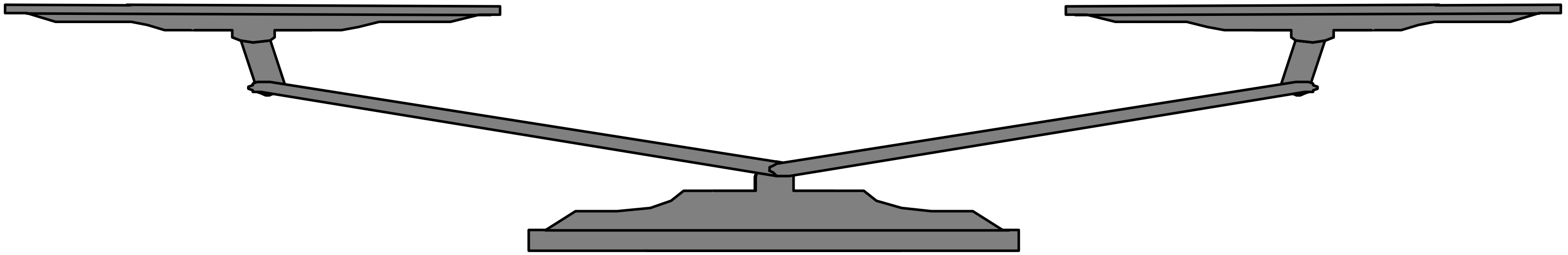
MC vs. TD

MC	TD
⊕ Unbiased ⊕ Simple & easy to implement ⊕ Discounted & average reward cases ⊖ High variance ⊖ End-of-episode updates ⊖ Not asymptotically optimal for inf. hor.	



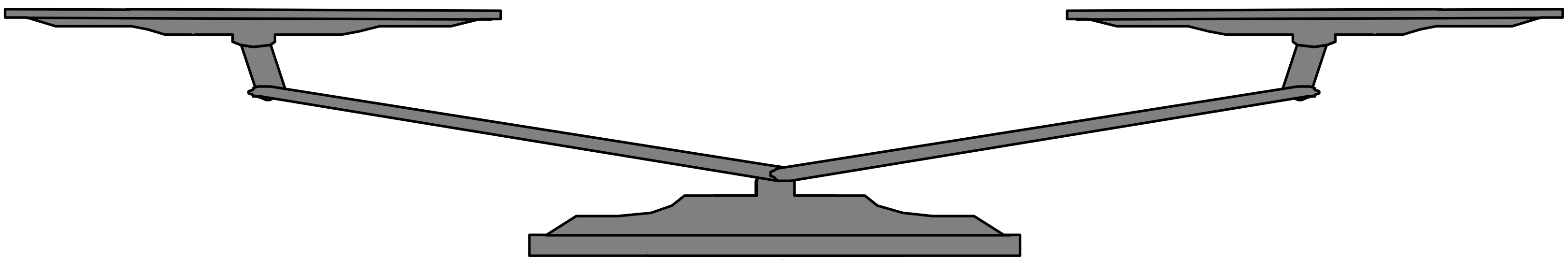
MC vs. TD

MC	TD
⊕ Unbiased	⊕ Low variance
⊕ Simple & easy to implement	⊕ Per-step updates
⊕ Discounted & average reward cases	⊕ Asymptotically optimal for inf. hor.
⊖ High variance	⊖ Biased
⊖ End-of-episode updates	⊖ Often requires function approximation
⊖ Not asymptotically optimal for inf. hor.	⊖ Additional effort for average reward



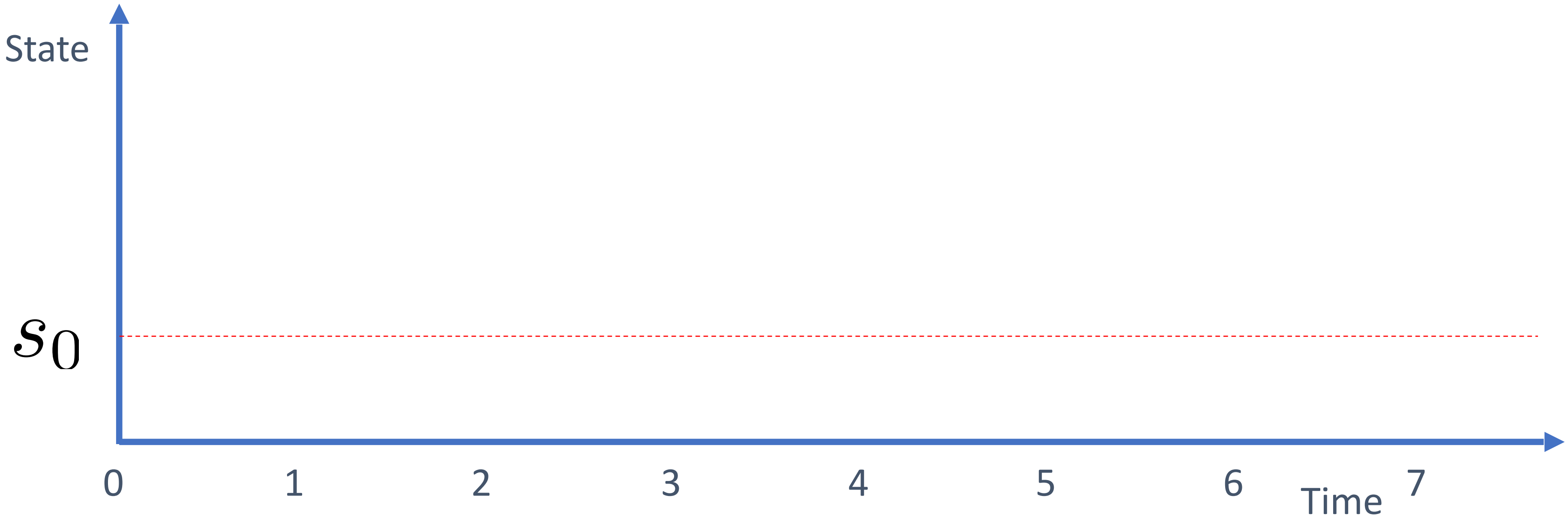
MC vs. TD

MC	TD
⊕ Unbiased	⊕ Low variance
⊕ Simple & easy to implement	⊕ Per-step updates
⊕ Discounted & average reward cases	⊕ Asymptotically optimal for inf. hor.
⊖ High variance	⊖ Biased
⊖ End-of-episode updates	⊖ Often requires function approximation
⊖ Not asymptotically optimal for inf. hor.	⊖ Additional effort for average reward

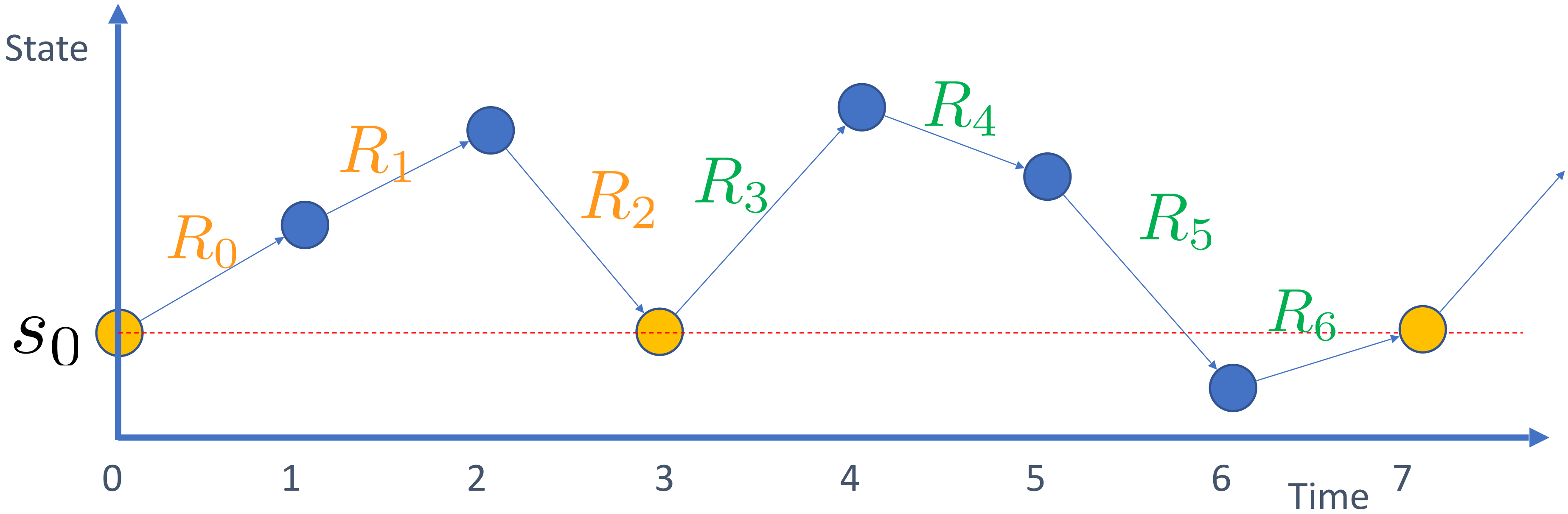


Can we get the best of both worlds?

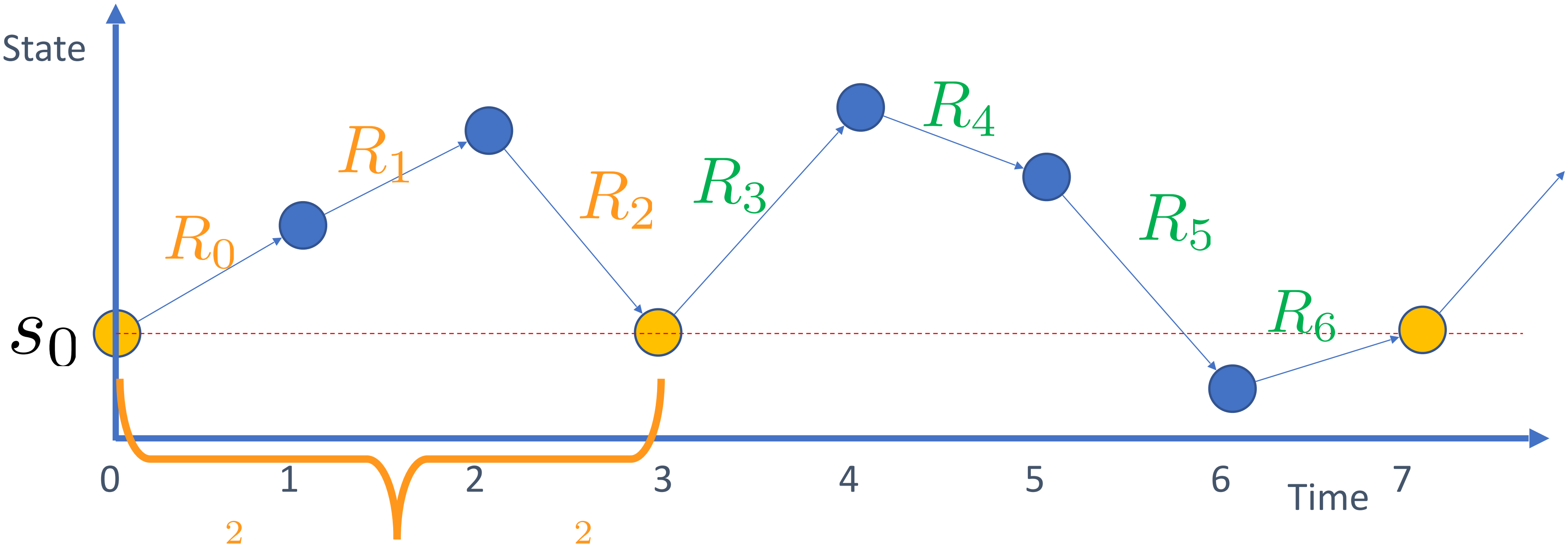
Renewal Monte Carlo



Renewal Monte Carlo

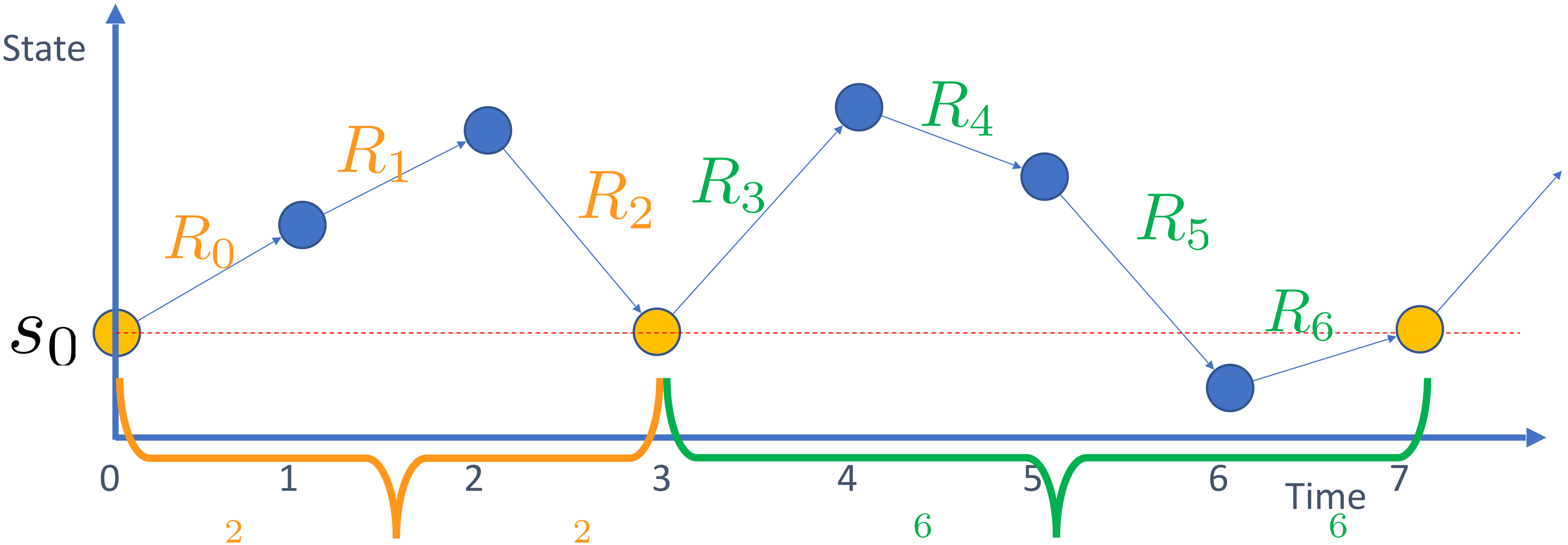


Renewal Monte Carlo



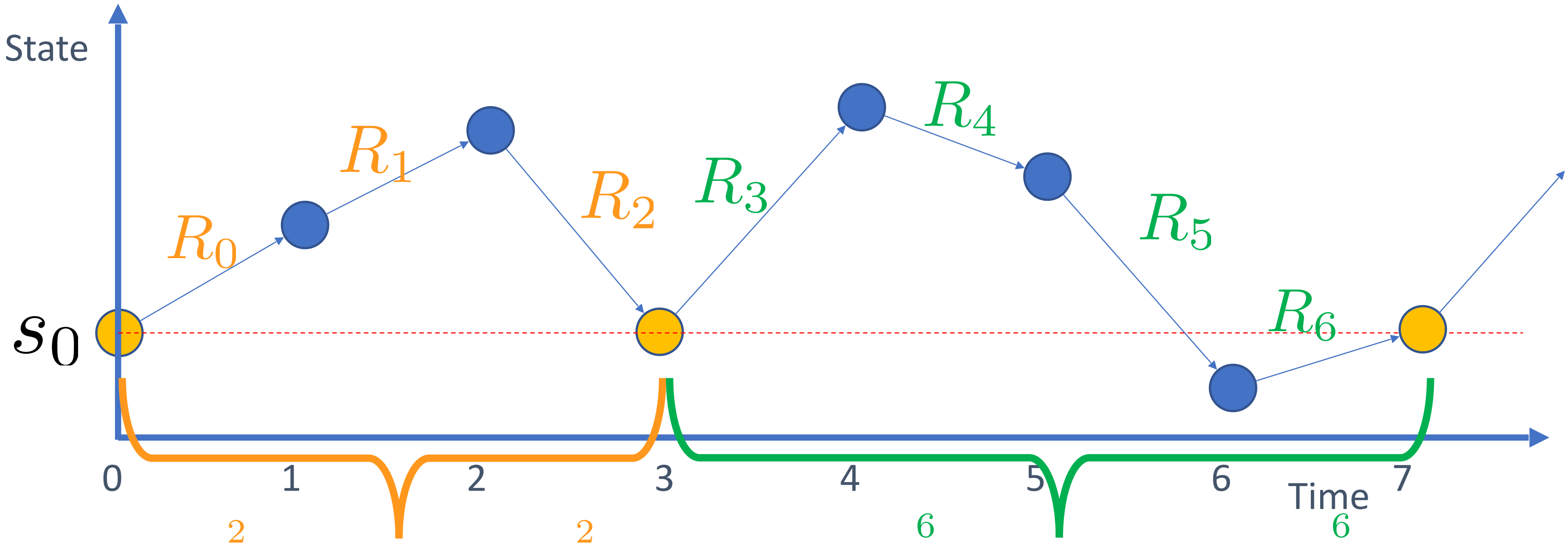
$$\mathsf{R}^{(1)} = \sum_{t=0}^{2} \gamma^t R_t, \mathsf{T}^{(1)} = \sum_{t=0}^{2} \gamma^t$$

Renewal Monte Carlo



$$\mathsf{R}^{(1)} = \sum_{t=0}^{2} \gamma^t R_t, \mathsf{T}^{(1)} = \sum_{t=0}^{2} \gamma^t \qquad \mathsf{R}^{(2)} = \sum_{t=3}^{6} \gamma^{t-3} R_t, \mathsf{T}^{(2)} = \sum_{t=3}^{6} \gamma^{t-3}$$

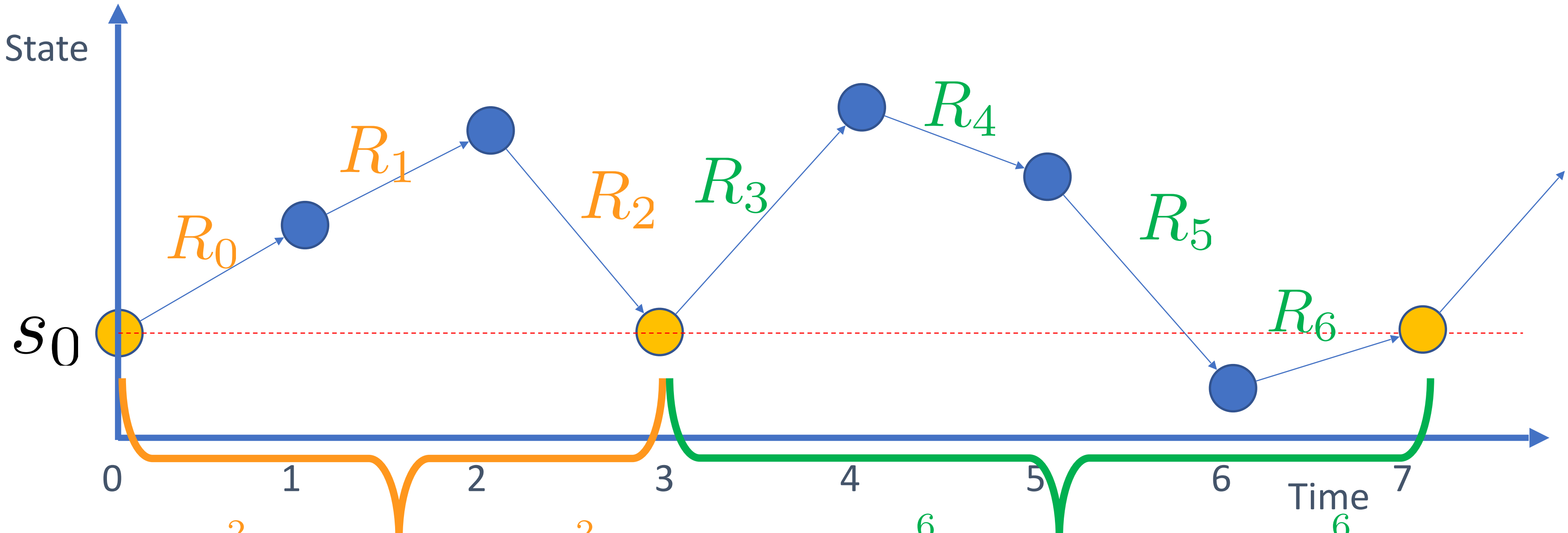
Renewal Monte Carlo



$$\mathsf{R}^{(1)} = \sum_{t=0}^{2} \gamma^t R_t, \mathsf{T}^{(1)} = \sum_{t=0}^{2} \gamma^t \quad \mathsf{R}^{(2)} = \sum_{t=3}^{6} \gamma^{t-3} R_t, \mathsf{T}^{(2)} = \sum_{t=3}^{6} \gamma^{t-3}$$

$\mathsf{R}^{(1)}, \mathsf{R}^{(2)} \ldots$ are i.i.d and $\mathsf{T}^{(1)}, \mathsf{T}^{(2)} \ldots$ are i.i.d

Renewal Monte Carlo

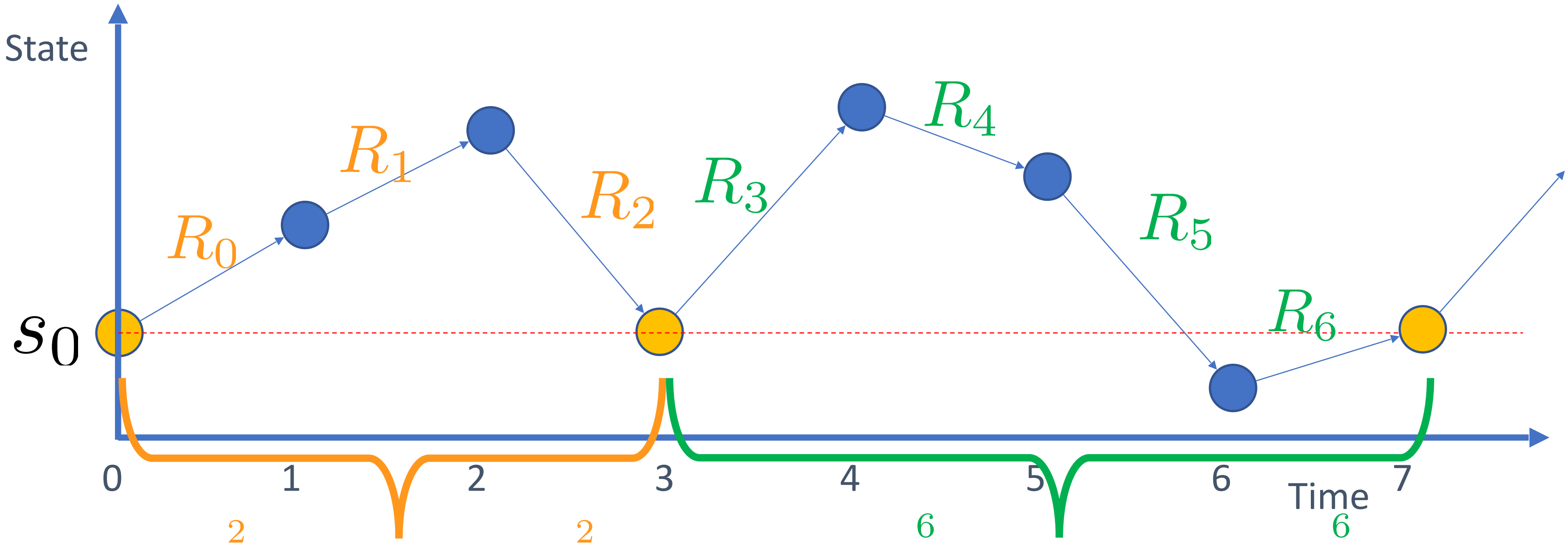


$$\mathsf{R}^{(1)} = \sum_{t=0}^{2} \gamma^t R_t, \mathsf{T}^{(1)} = \sum_{t=0}^{2} \gamma^t \qquad \mathsf{R}^{(2)} = \sum_{t=3}^{6} \gamma^{t-3} R_t, \mathsf{T}^{(2)} = \sum_{t=3}^{6} \gamma^{t-3}$$

$\mathsf{R}^{(1)}, \mathsf{R}^{(2)} \ldots$ are i.i.d and $\mathsf{T}^{(1)}, \mathsf{T}^{(2)} \ldots$ are i.i.d

$\mathsf{R}_\theta = \mathbb{E}[\mathsf{R}^{(n)}]$ and $\mathsf{T}_\theta = \mathbb{E}[\mathsf{T}^{(n)}]$

Renewal Monte Carlo



$$\mathsf{R}^{(1)} = \sum_{t=0}^{2} \gamma^t R_t, \mathsf{T}^{(1)} = \sum_{t=0}^{2} \gamma^t \quad \mathsf{R}^{(2)} = \sum_{t=3}^{6} \gamma^{t-3} R_t, \mathsf{T}^{(2)} = \sum_{t=3}^{6} \gamma^{t-3}$$

$\mathsf{R}^{(1)}$, $\mathsf{R}^{(2)}$... are i.i.d and $\mathsf{T}^{(1)}$, $\mathsf{T}^{(2)}$... are i.i.d

$\mathsf{R}_\theta = \mathbb{E}[\mathsf{R}^{(n)}]$ and $\mathsf{T}_\theta = \mathbb{E}[\mathsf{T}^{(n)}]$ estimated by $\hat{R}, \hat{T}$

RMC based policy gradient

RMC based policy gradient

Performance Gradient Estimate

$$J_\theta = \frac{\mathsf{R}_\theta}{(1-\gamma)\mathsf{T}_\theta}$$

RMC based policy gradient

Performance Gradient Estimate

$$J_\theta = \frac{\mathsf{R}_\theta}{(1-\gamma)\mathsf{T}_\theta} \quad ; \quad \nabla_\theta J_\theta = \frac{H_\theta}{(1-\gamma){\mathsf{T}_\theta}^2}$$

RMC based policy gradient

Performance Gradient Estimate

$$J_\theta = \frac{\mathsf{R}_\theta}{(1-\gamma)\mathsf{T}_\theta} \quad ; \quad \nabla_\theta J_\theta = \frac{H_\theta}{(1-\gamma){\mathsf{T}_\theta}^2}$$

$$H_\theta = \mathsf{T}_\theta \nabla_\theta \mathsf{R}_\theta - \mathsf{R}_\theta \nabla_\theta \mathsf{T}_\theta$$

RMC based policy gradient

Performance Gradient Estimate

$$J_\theta = \frac{\mathsf{R}_\theta}{(1-\gamma)\mathsf{T}_\theta} \quad ; \quad \nabla_\theta J_\theta = \frac{H_\theta}{(1-\gamma)\mathsf{T}_\theta{}^2}$$

$$H_\theta = \mathsf{T}_\theta \nabla_\theta \mathsf{R}_\theta - \mathsf{R}_\theta \nabla_\theta \mathsf{T}_\theta$$ with estimate: $\widehat{H}_\theta$

RMC based policy gradient

Performance Gradient Estimate

$$J_\theta = \frac{\mathsf{R}_\theta}{(1-\gamma)\mathsf{T}_\theta} \quad ; \quad \nabla_\theta J_\theta = \frac{H_\theta}{(1-\gamma){\mathsf{T}_\theta}^2}$$

$H_\theta = \mathsf{T}_\theta \nabla_\theta \mathsf{R}_\theta - \mathsf{R}_\theta \nabla_\theta \mathsf{T}_\theta$ with estimate: $\widehat{H}_\theta$

Stochastic Gradient Ascent

$$\theta_{k+1} = [\theta_k + \alpha_k \widehat{H}_{\theta_k}]_\Theta$$

$\sum_{k=0}^{\infty} \alpha_k = \infty$ and $\sum_{k=0}^{\infty} \alpha_k^2 < \infty$

RMC based policy gradient

Performance Gradient Estimate

$$J_\theta = \frac{\mathsf{R}_\theta}{(1-\gamma)\mathsf{T}_\theta} \quad ; \quad \nabla_\theta J_\theta = \frac{H_\theta}{(1-\gamma)\mathsf{T}_\theta{}^2}$$

$H_\theta = \mathsf{T}_\theta \nabla_\theta \mathsf{R}_\theta - \mathsf{R}_\theta \nabla_\theta \mathsf{T}_\theta$ with estimate: $\widehat{H}_\theta$

$\widehat{\mathsf{R}}_\theta, \widehat{\mathsf{T}}_\theta$ estimated using MC / TD ; $\nabla_\theta \widehat{\mathsf{R}}_\theta, \nabla_\theta \widehat{\mathsf{T}}_\theta$ using RL policy gradient

Stochastic Gradient Ascent

$$\theta_{k+1} = [\theta_k + \alpha_k \widehat{H}_{\theta_k}]_\Theta$$

$\sum_{k=0}^{\infty} \alpha_k = \infty$ and $\sum_{k=0}^{\infty} \alpha_k^2 < \infty$

Convergence

Convergence

$\widehat{\mathsf{R}}_k, \widehat{\mathsf{T}}_k, \widehat{\nabla}\mathsf{R}_k, \widehat{\nabla}\mathsf{T}_k$ unbiased estimators of $\mathsf{R}_{\theta_k}, \mathsf{T}_{\theta_k}, \nabla\mathsf{R}_{\theta_k}, \nabla\mathsf{T}_{\theta_k}$

Convergence

$\widehat{\mathsf{R}}_k, \widehat{\mathsf{T}}_k, \widehat{\nabla}\mathsf{R}_k, \widehat{\nabla}\mathsf{T}_k$ unbiased estimators of $\mathsf{R}_{\theta_k}, \mathsf{T}_{\theta_k}, \nabla\mathsf{R}_{\theta_k}, \nabla\mathsf{T}_{\theta_k}$

$\widehat{\mathsf{T}}_k \perp \widehat{\nabla}\mathsf{R}_k$ and $\widehat{\mathsf{R}}_k \perp \widehat{\nabla}\mathsf{T}_k$

Convergence

$\widehat{\mathsf{R}}_k, \widehat{\mathsf{T}}_k, \widehat{\nabla}\mathsf{R}_k, \widehat{\nabla}\mathsf{T}_k$ unbiased estimators of $\mathsf{R}_{\theta_k}, \mathsf{T}_{\theta_k}, \nabla\mathsf{R}_{\theta_k}, \nabla\mathsf{T}_{\theta_k}$

$\widehat{\mathsf{T}}_k \perp \widehat{\nabla}\mathsf{R}_k$ and $\widehat{\mathsf{R}}_k \perp \widehat{\nabla}\mathsf{T}_k$

$\widehat{H}_k = \widehat{\mathsf{T}}_k \widehat{\nabla}\mathsf{R}_k - \widehat{\mathsf{R}}_k \widehat{\nabla}\mathsf{T}_k$ is an unbiased estimator of H_{θ_k}

Convergence

$\widehat{\mathsf{R}}_k, \widehat{\mathsf{T}}_k, \widehat{\nabla}\mathsf{R}_k, \widehat{\nabla}\mathsf{T}_k$ unbiased estimators of $\mathsf{R}_{\theta_k}, \mathsf{T}_{\theta_k}, \nabla\mathsf{R}_{\theta_k}, \nabla\mathsf{T}_{\theta_k}$

$\widehat{\mathsf{T}}_k \perp \widehat{\nabla}\mathsf{R}_k$ and $\widehat{\mathsf{R}}_k \perp \widehat{\nabla}\mathsf{T}_k$

$\widehat{H}_k = \widehat{\mathsf{T}}_k \widehat{\nabla}\mathsf{R}_k - \widehat{\mathsf{R}}_k \widehat{\nabla}\mathsf{T}_k$ is an unbiased estimator of H_{θ_k}

+

H_θ is continuous; $\widehat{H}_k$ has bounded variance and

Convergence

$\widehat{\mathsf{R}}_k, \widehat{\mathsf{T}}_k, \widehat{\nabla}\mathsf{R}_k, \widehat{\nabla}\mathsf{T}_k$ unbiased estimators of $\mathsf{R}_{\theta_k}, \mathsf{T}_{\theta_k}, \nabla\mathsf{R}_{\theta_k}, \nabla\mathsf{T}_{\theta_k}$

$\widehat{\mathsf{T}}_k \perp \widehat{\nabla}\mathsf{R}_k$ and $\widehat{\mathsf{R}}_k \perp \widehat{\nabla}\mathsf{T}_k$

$\widehat{H}_k = \widehat{\mathsf{T}}_k \widehat{\nabla}\mathsf{R}_k - \widehat{\mathsf{R}}_k \widehat{\nabla}\mathsf{T}_k$ is an unbiased estimator of H_{θ_k}

+

H_θ is continuous; $\widehat{H}_k$ has bounded variance and

$\dot{\theta} = H_\theta$ has locally asymptotically stable isolated limit points

Convergence

$\widehat{\mathsf{R}}_k, \widehat{\mathsf{T}}_k, \widehat{\nabla}\mathsf{R}_k, \widehat{\nabla}\mathsf{T}_k$ unbiased estimators of $\mathsf{R}_{\theta_k}, \mathsf{T}_{\theta_k}, \nabla\mathsf{R}_{\theta_k}, \nabla\mathsf{T}_{\theta_k}$

$\widehat{\mathsf{T}}_k \perp \widehat{\nabla}\mathsf{R}_k$ and $\widehat{\mathsf{R}}_k \perp \widehat{\nabla}\mathsf{T}_k$

$\widehat{H}_k = \widehat{\mathsf{T}}_k \widehat{\nabla}\mathsf{R}_k - \widehat{\mathsf{R}}_k \widehat{\nabla}\mathsf{T}_k$ is an unbiased estimator of H_{θ_k}

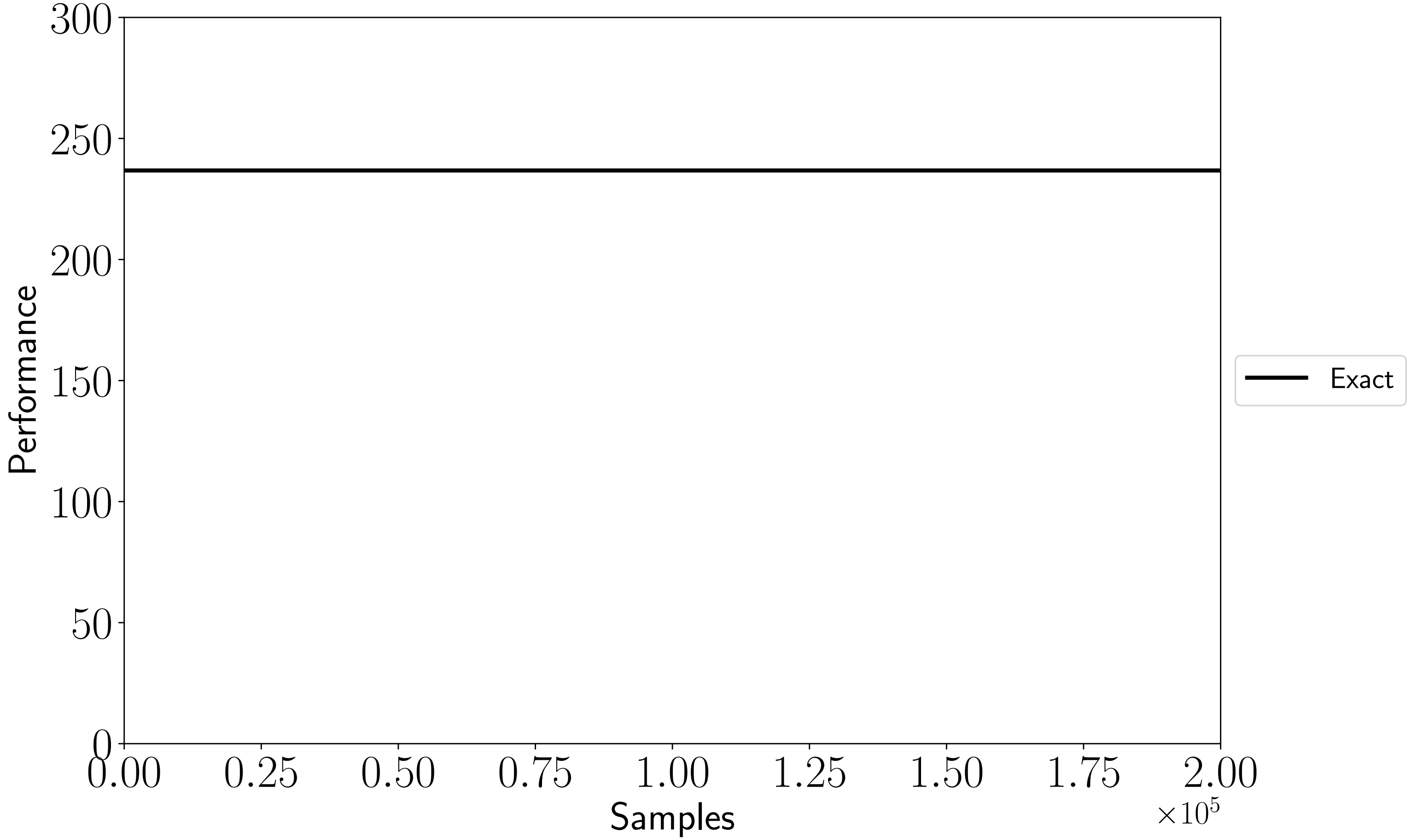
+

H_θ is continuous; $\widehat{H}_k$ has bounded variance and

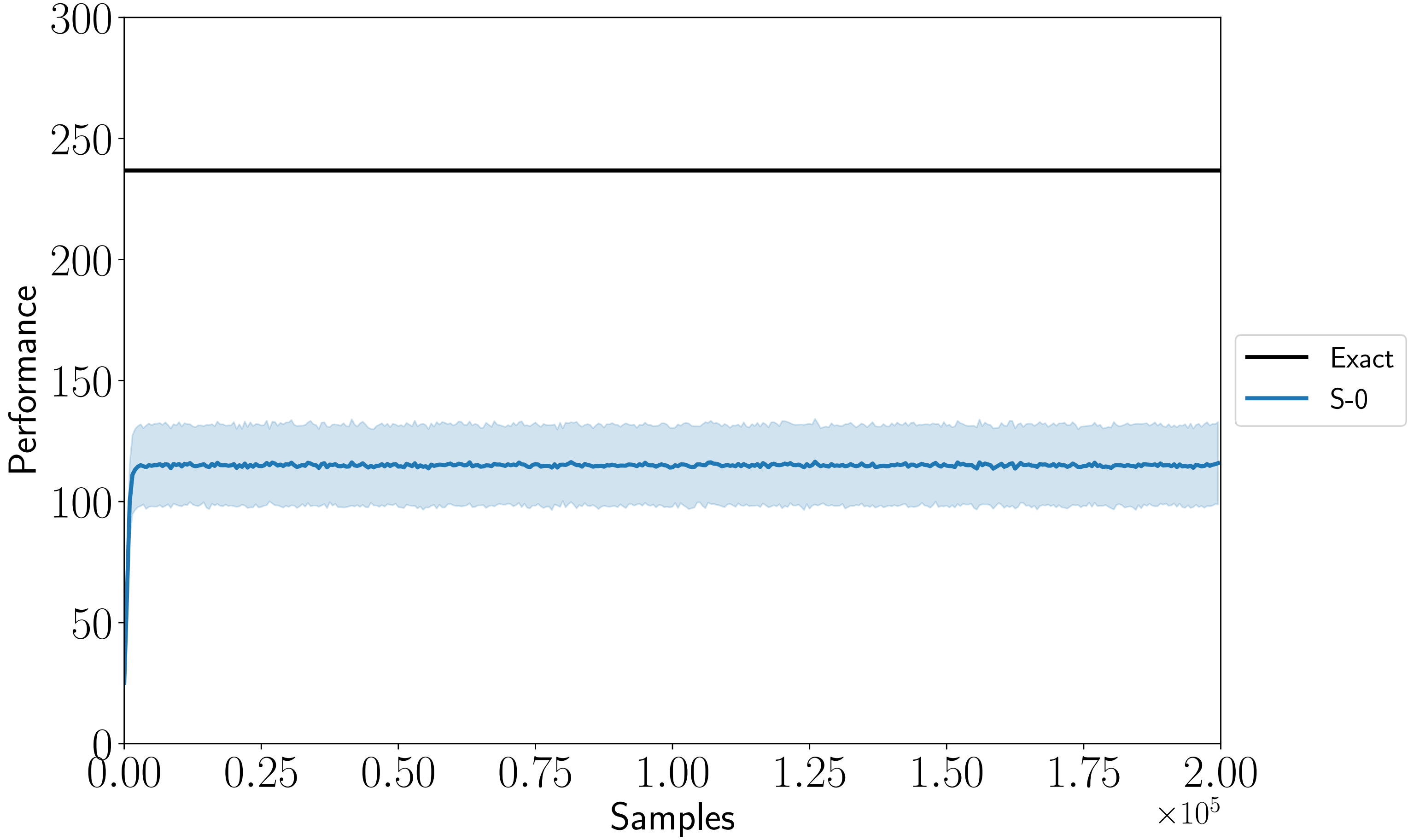
$\dot{\theta} = H_\theta$ has locally asymptotically stable isolated limit points

Iteration for θ_k converges a.s. to a value where $\nabla_\theta J_\theta = 0$

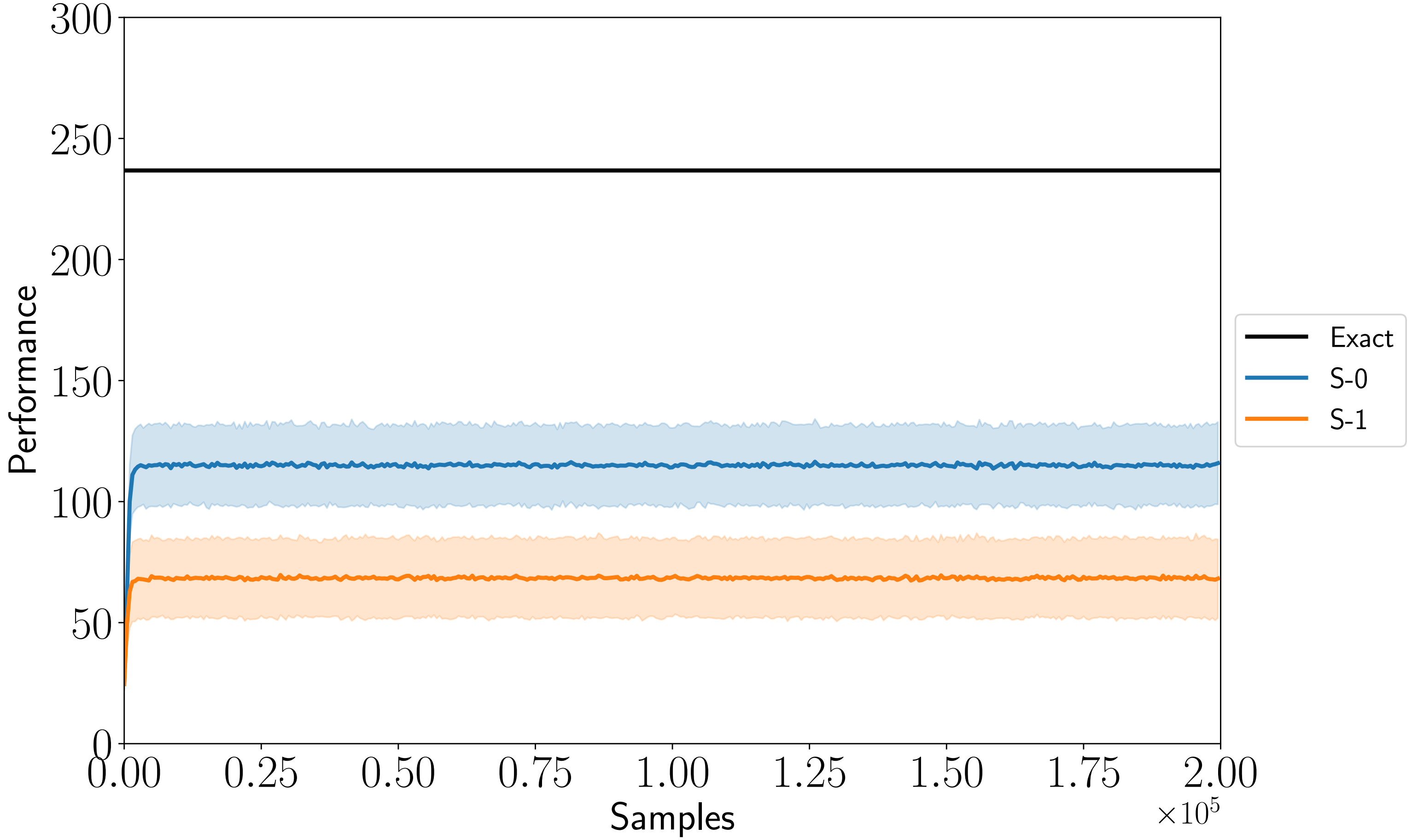
E.g. – Randomly generated MDP



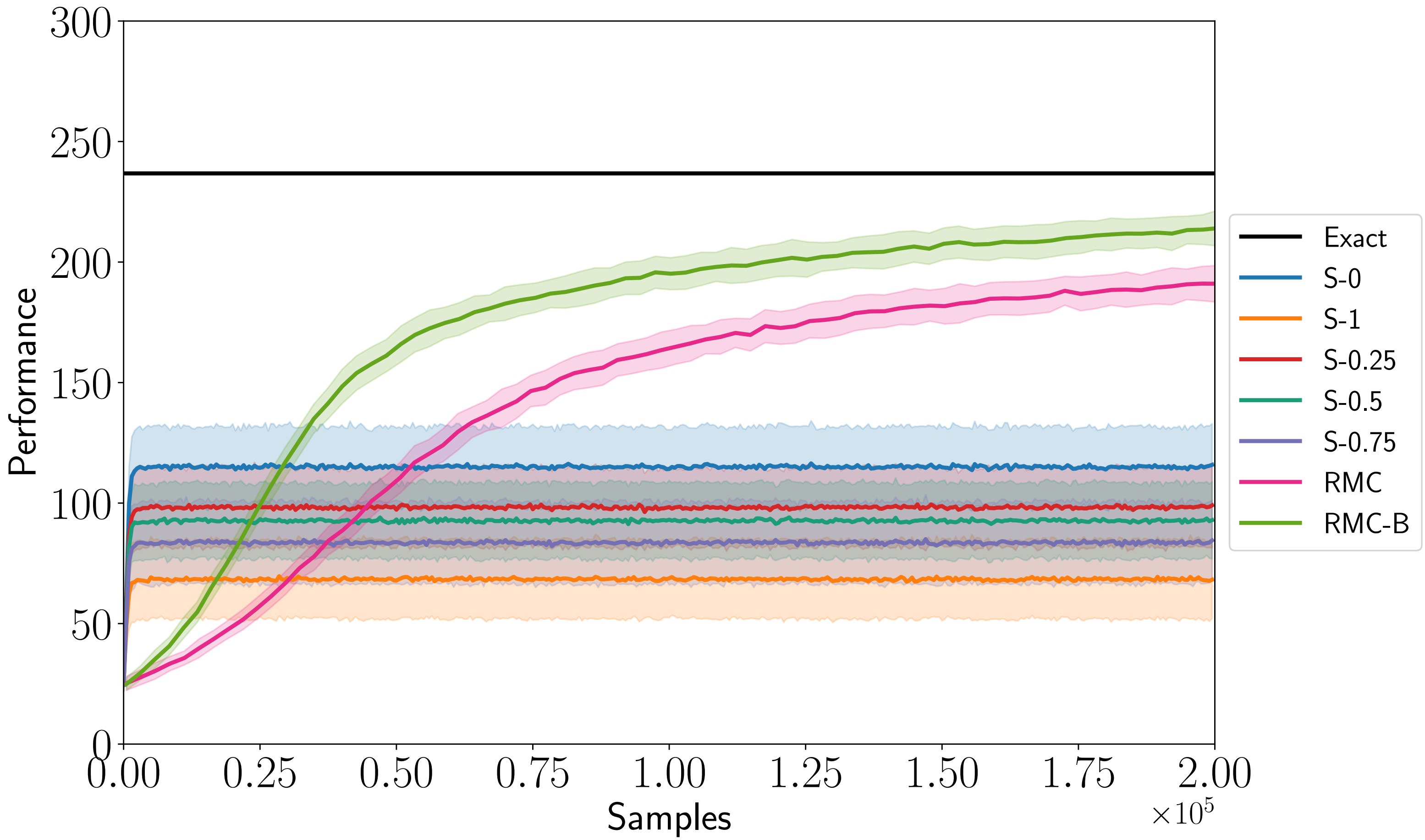
E.g. – Randomly generated MDP



E.g. – Randomly generated MDP



E.g. – Randomly generated MDP



Related work

Related work

- Simulation optimization [Glynn 1986, 1990]:
  - Assume known probability law of the primitive random variables and its weak derivate

Related work

- Simulation optimization [Glynn 1986, 1990]:
  - Assume known probability law of the primitive random variables and its weak derivate
- Sensitivity analysis for MDPs [Xi-Ren Cao, 1997]:
  - Average reward criterion
  - Known and unknown system models

Related work

- Simulation optimization [Glynn 1986, 1990]:
  - Assume known probability law of the primitive random variables and its weak derivate
- Sensitivity analysis for MDPs [Xi-Ren Cao, 1997]:
  - Average reward criterion
  - Known and unknown system models
- Renewal theory for RL: [Marbach & Tsitsiklis 2001, 2003]
  - Average reward criterion
  - Relative value function for average reward

Limitation of RMC

Limitation of RMC

- ⊖ Renewal could take a long time

Limitation of RMC

⊖ Renewal could take a long time

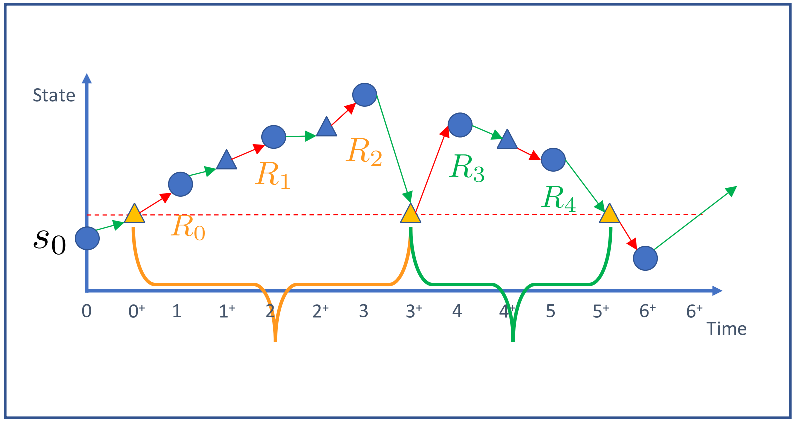
⊕ Two techniques to overcome this:

Limitation of RMC

⊖ Renewal could take a long time

⊕ Two techniques to overcome this:

Post-decision state model

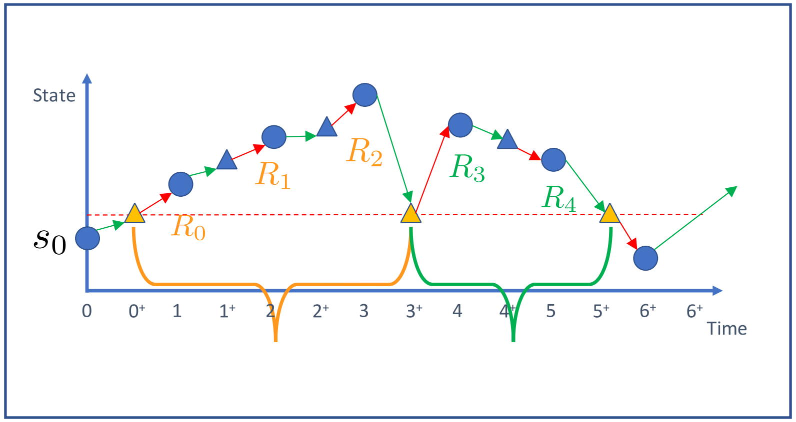


Limitation of RMC

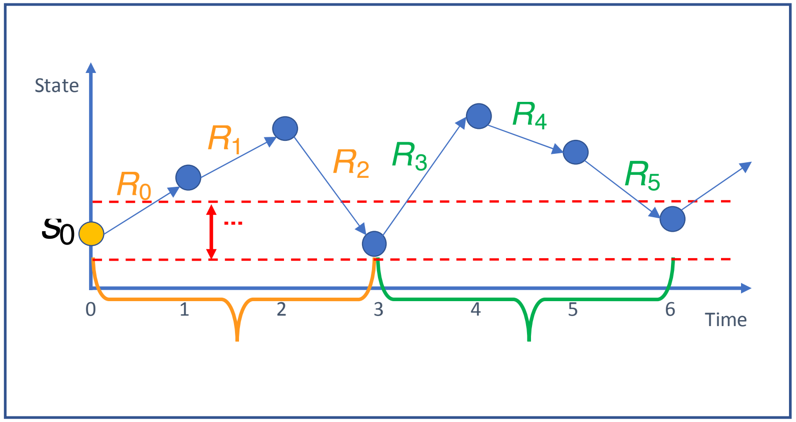
⊖ Renewal could take a long time

⊕ Two techniques to overcome this:

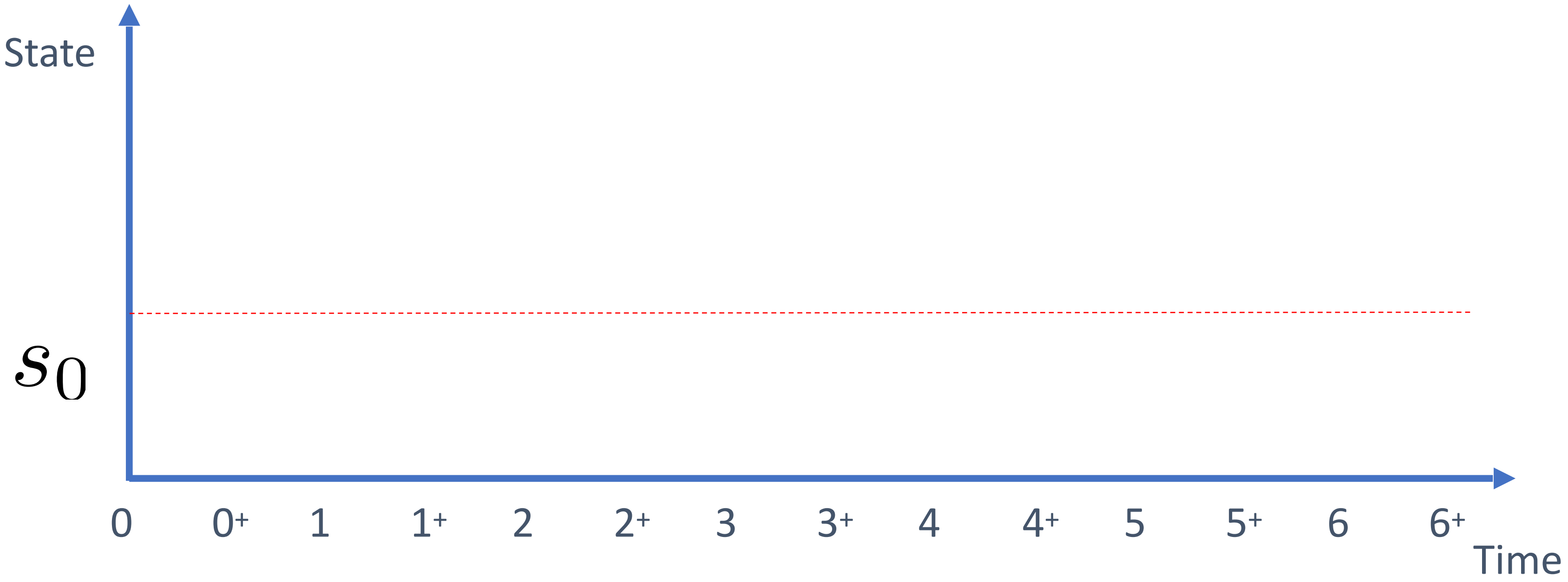
Post-decision state model



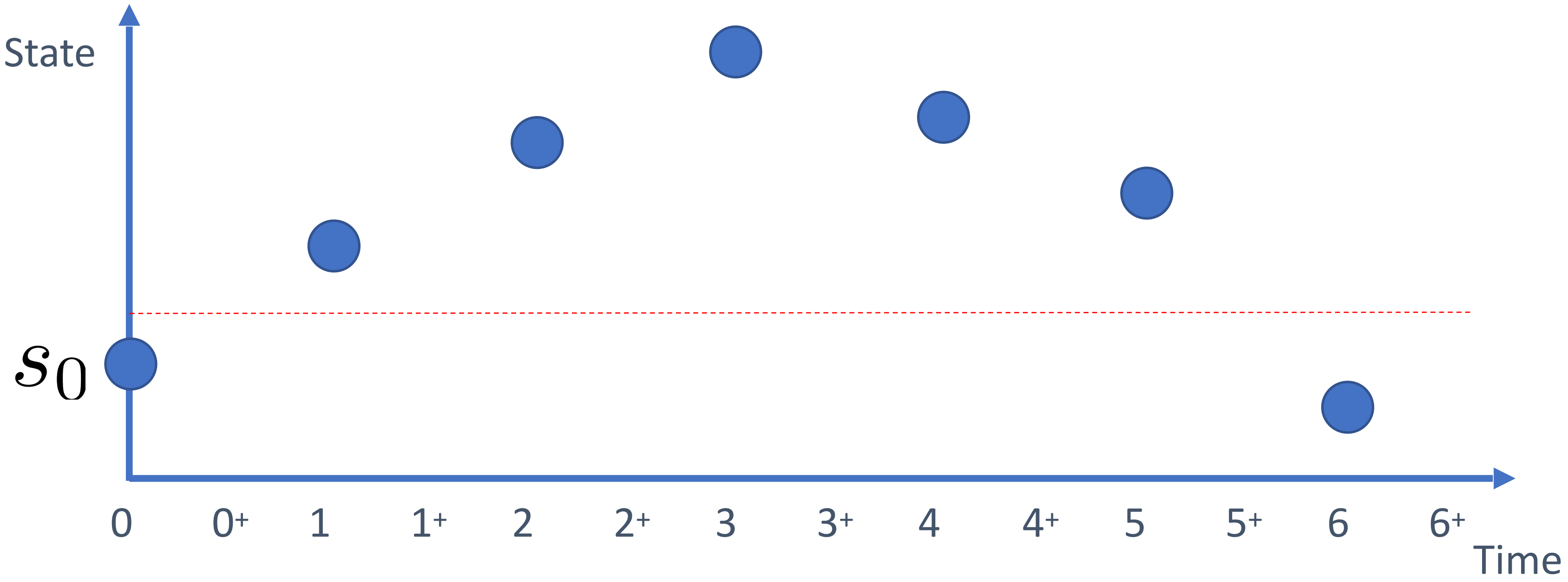
Approximate renewal model



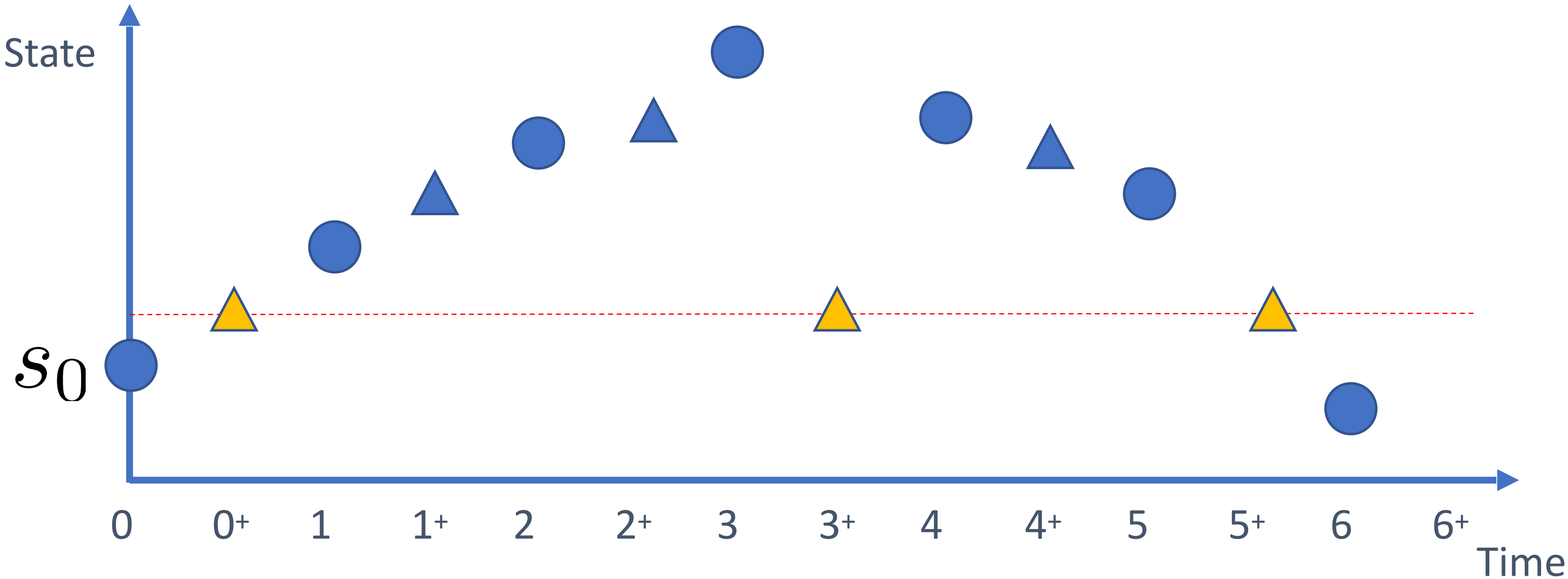
Post-decision state model



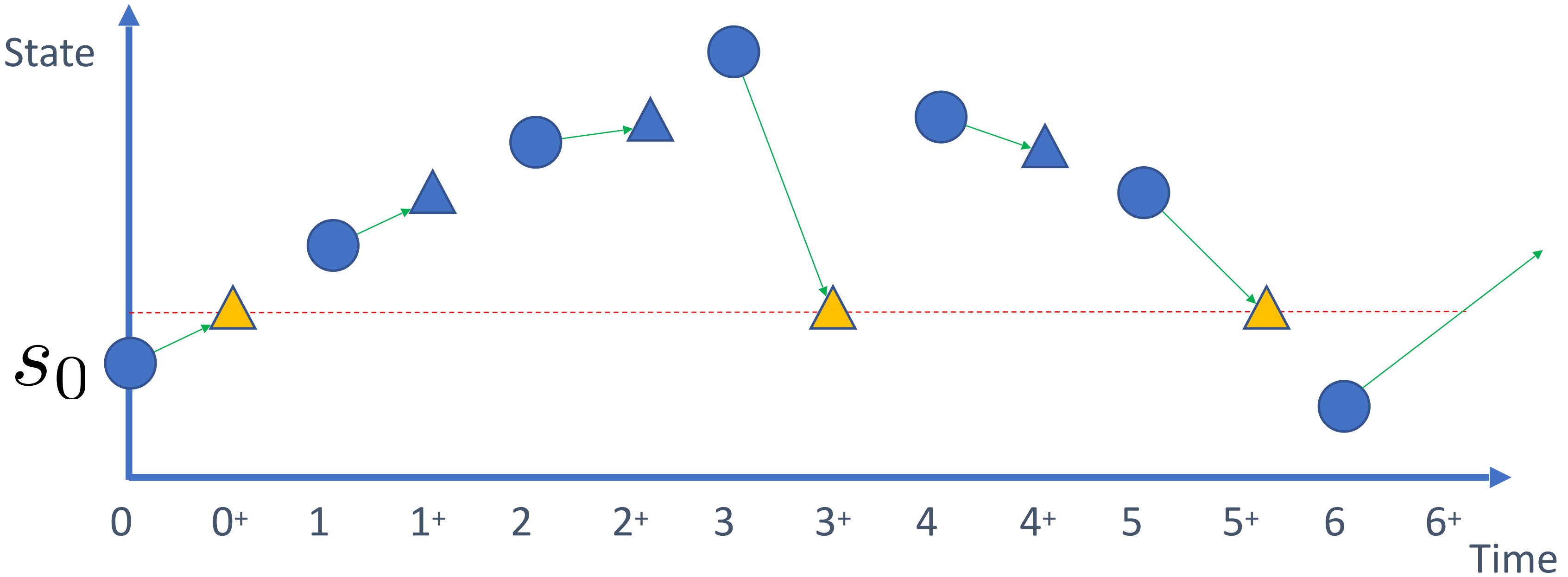
Post-decision state model



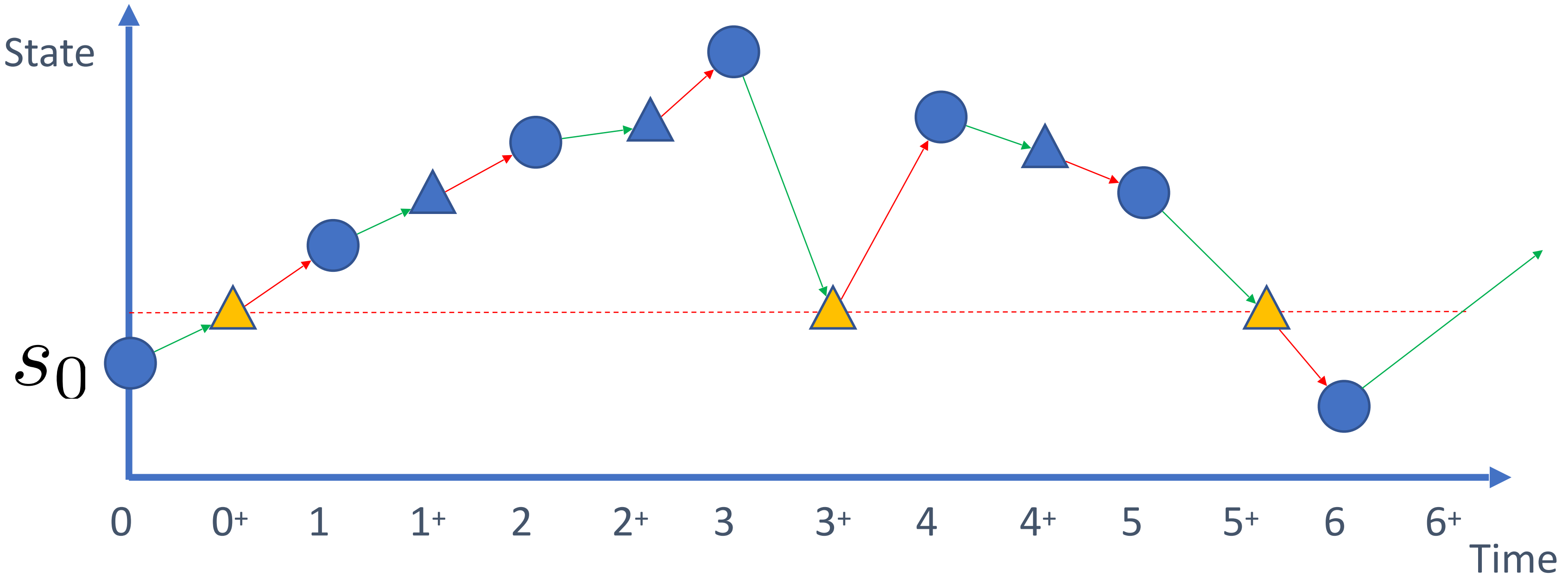
Post-decision state model



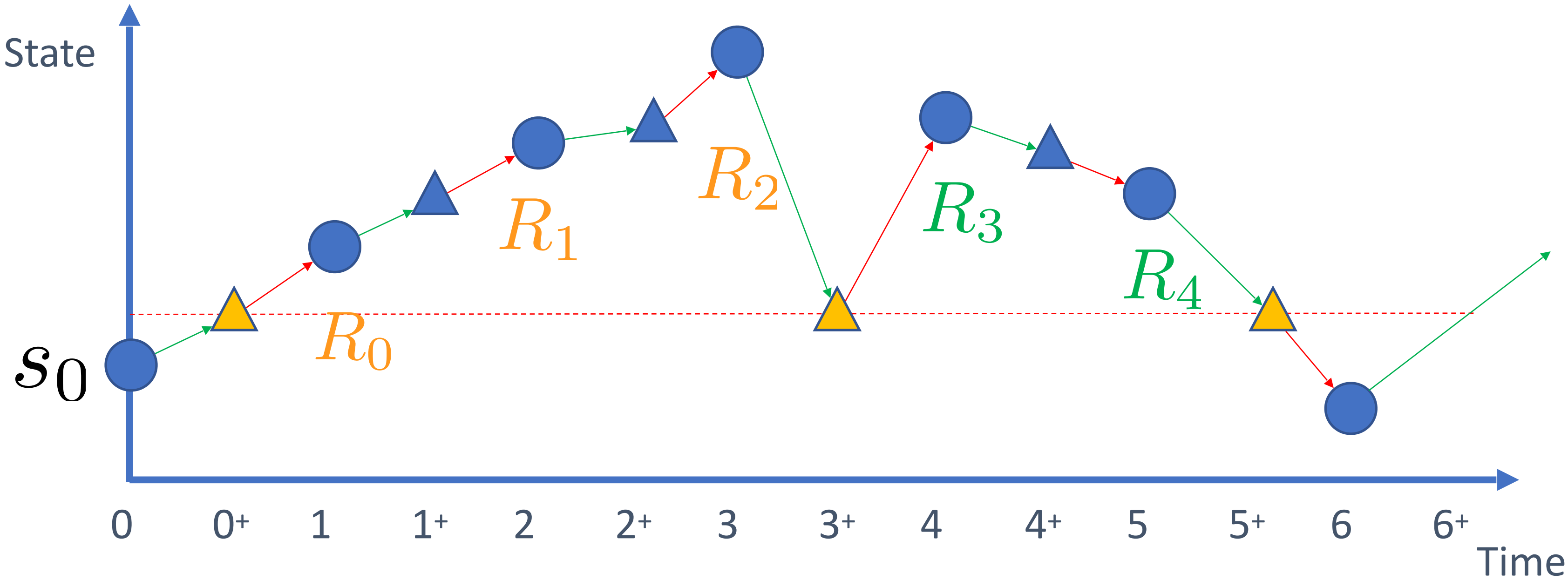
Post-decision state model



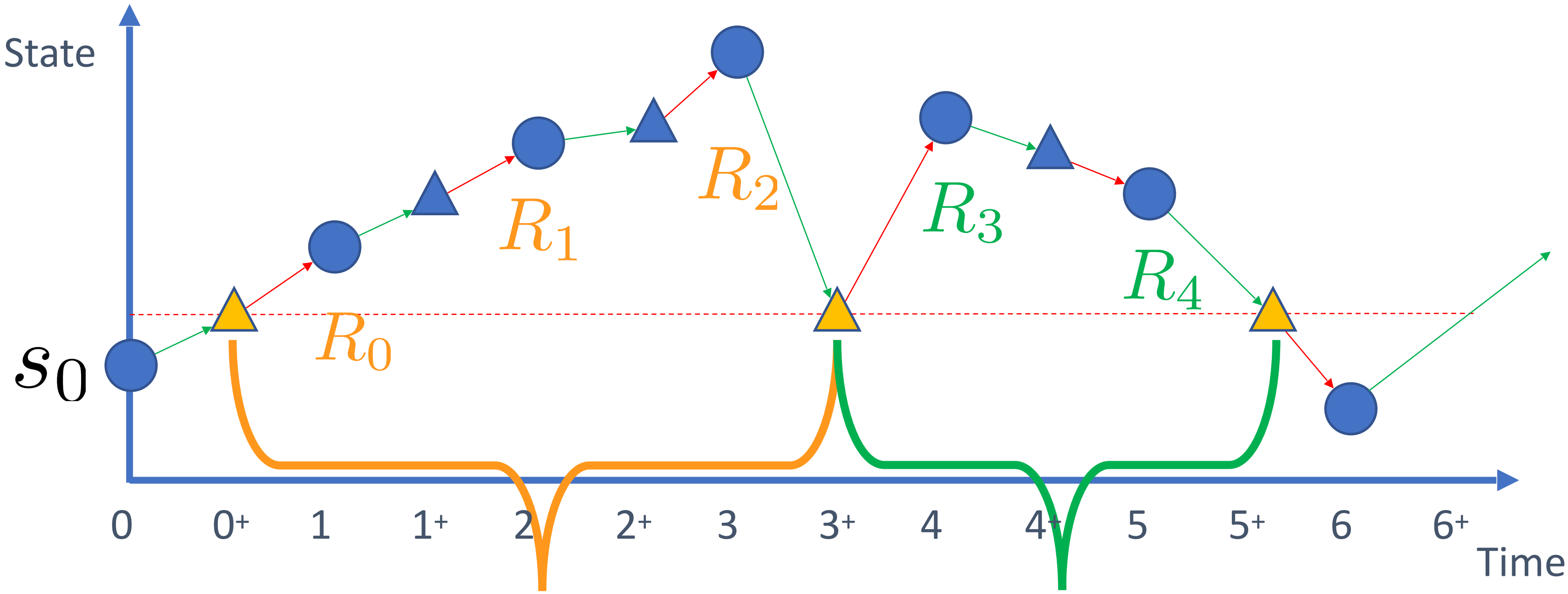
Post-decision state model



Post-decision state model

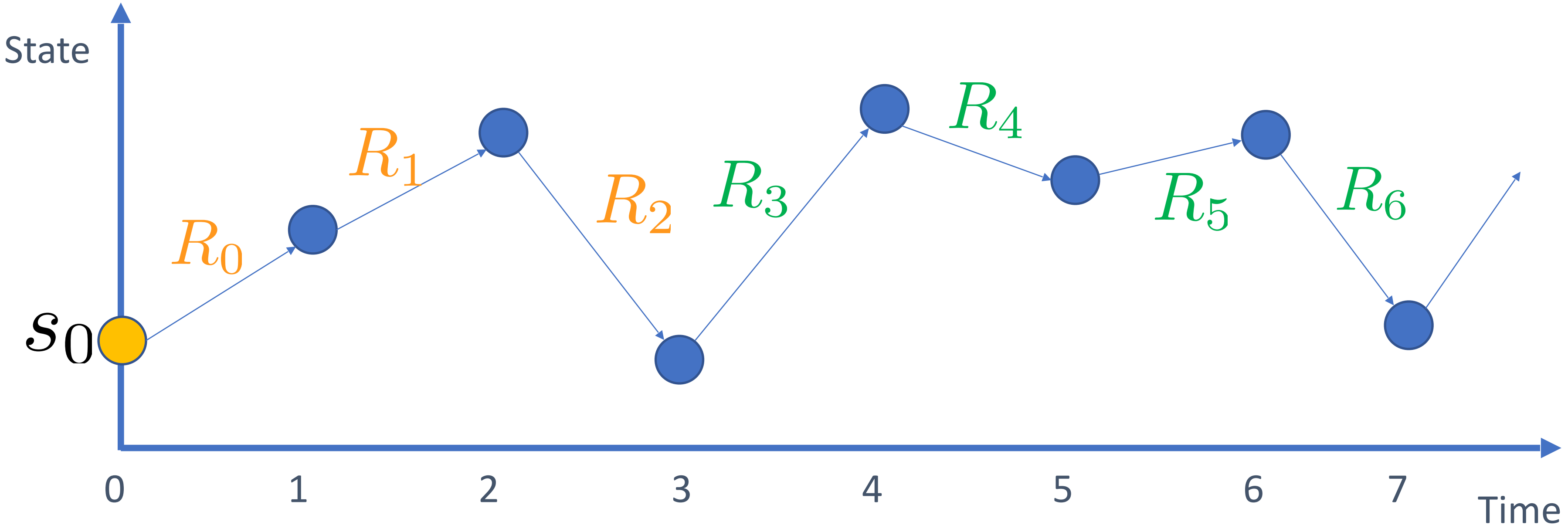


Post-decision state model

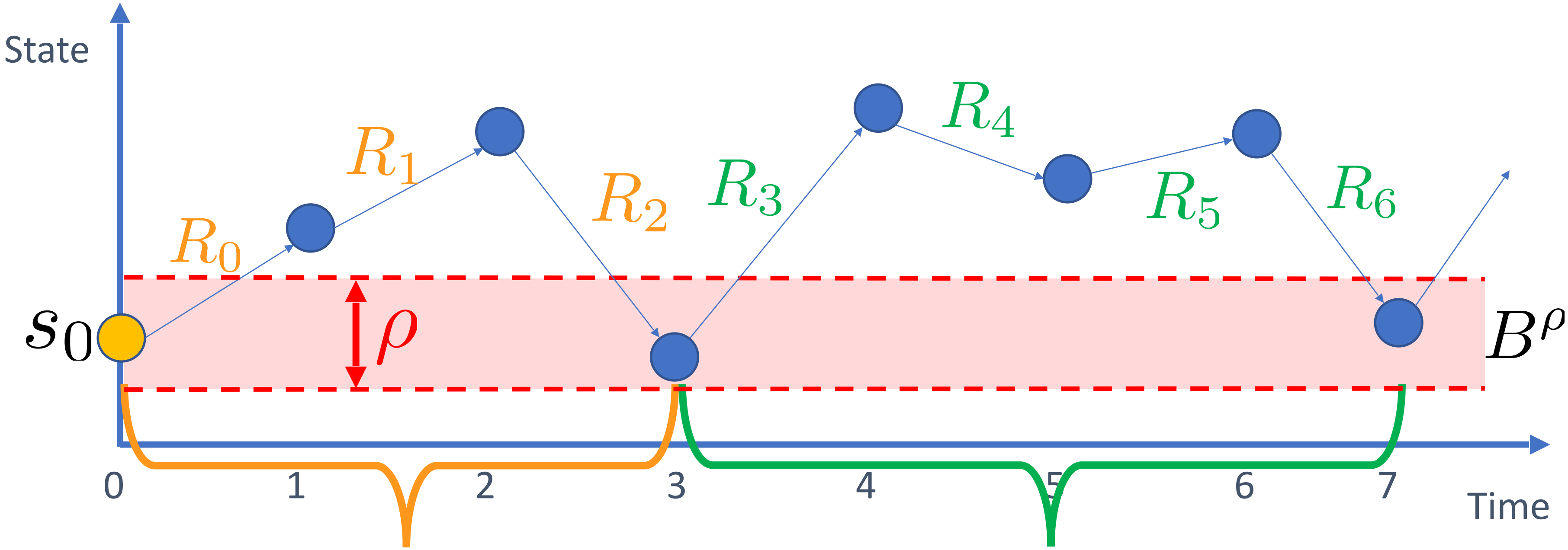


Renewals defined in terms of post-decision states

Approximate RMC

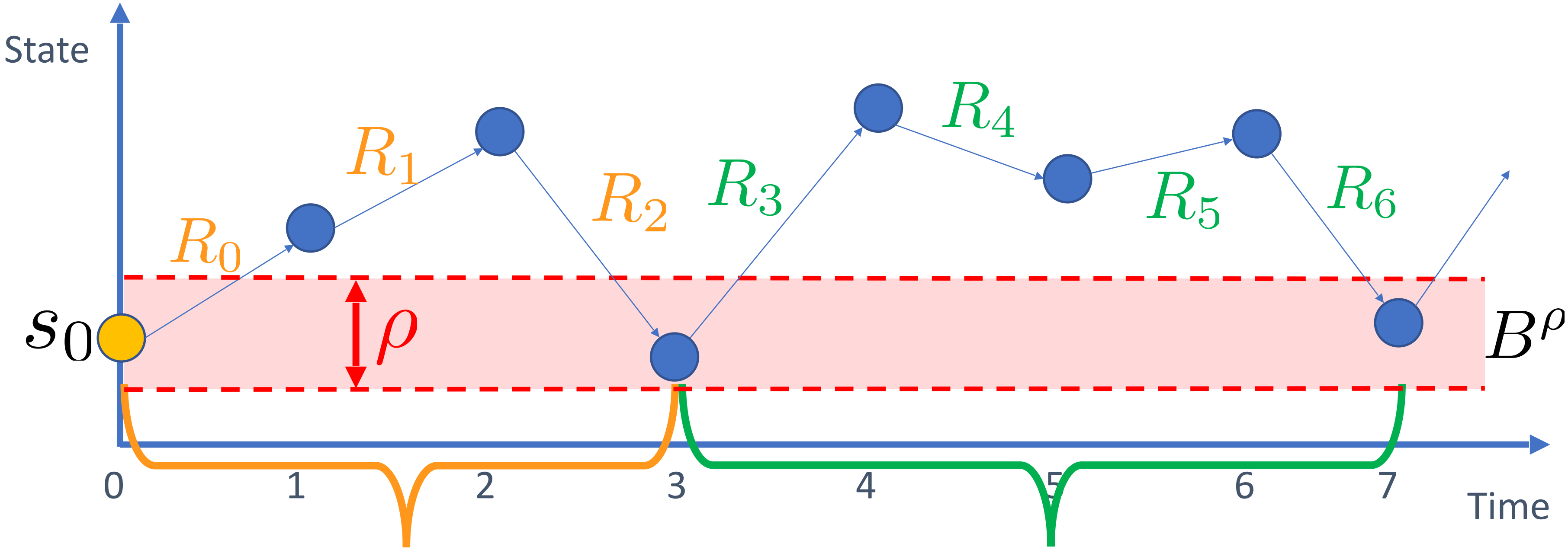


Approximate RMC



$\mathsf{R}^{\rho,(1)} = \sum_{t=0}^{2} \gamma^t R_t, \mathsf{T}^{\rho,(1)} = \sum_{t=0}^{2} \gamma^t$ $\mathsf{R}^{\rho,(2)} = \sum_{t=3}^{6} \gamma^{t-3} R_t, \mathsf{T}^{\rho,(2)} = \sum_{t=3}^{6} \gamma^{t-3}$

Approximate RMC



$\mathsf{R}^{\rho,(1)} = \sum_{t=0}^{2} \gamma^t R_t, \mathsf{T}^{\rho,(1)} = \sum_{t=0}^{2} \gamma^t \quad \mathsf{R}^{\rho,(2)} = \sum_{t=3}^{6} \gamma^{t-3} R_t, \mathsf{T}^{\rho,(2)} = \sum_{t=3}^{6} \gamma^{t-3}$

$\mathsf{R}^{(1)}$, $\mathsf{R}^{(2)}$... are i.i.d and $\mathsf{T}^{(1)}$, $\mathsf{T}^{(2)}$... are i.i.d

$\mathsf{R}^{\rho}_{\theta} = \mathbb{E}[\mathsf{R}^{\rho,(n)}]$ and $\mathsf{T}^{\rho}_{\theta} = \mathbb{E}[\mathsf{T}^{\rho,(n)}]$ ▶ estimated by $\hat{R}^\rho, \hat{T}^\rho$

Error bound

Error bound

V_θ is Locally Lipschitz in B^ρ

Error bound

V_θ is Locally Lipschitz in B^ρ

$$|V_\theta(s) - V_\theta(s')| \leq L_\theta d_S(s, s')$$

Error bound

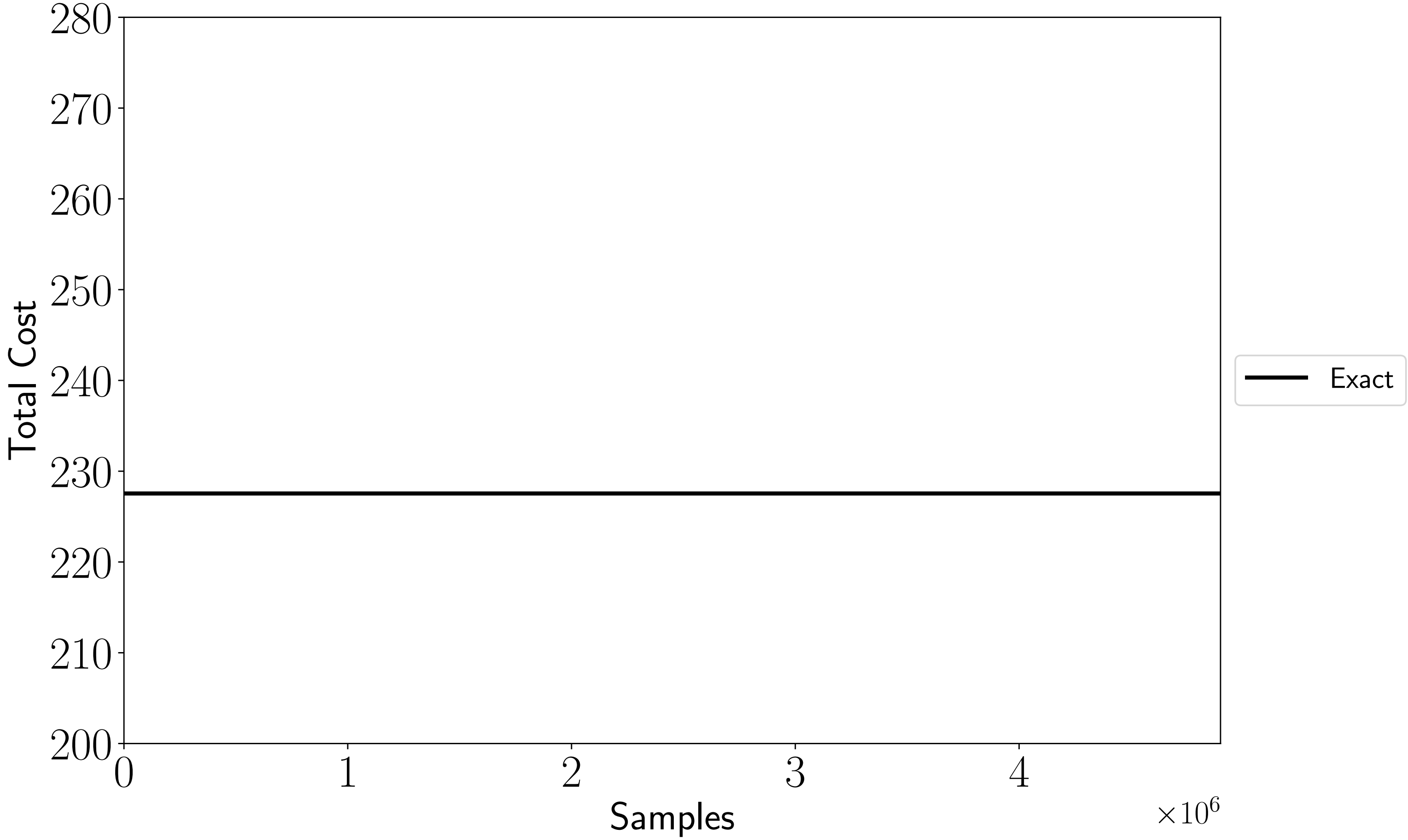
V_θ is Locally Lipschitz in B^ρ

$$|V_\theta(s) - V_\theta(s')| \leq L_\theta d_S(s, s')$$

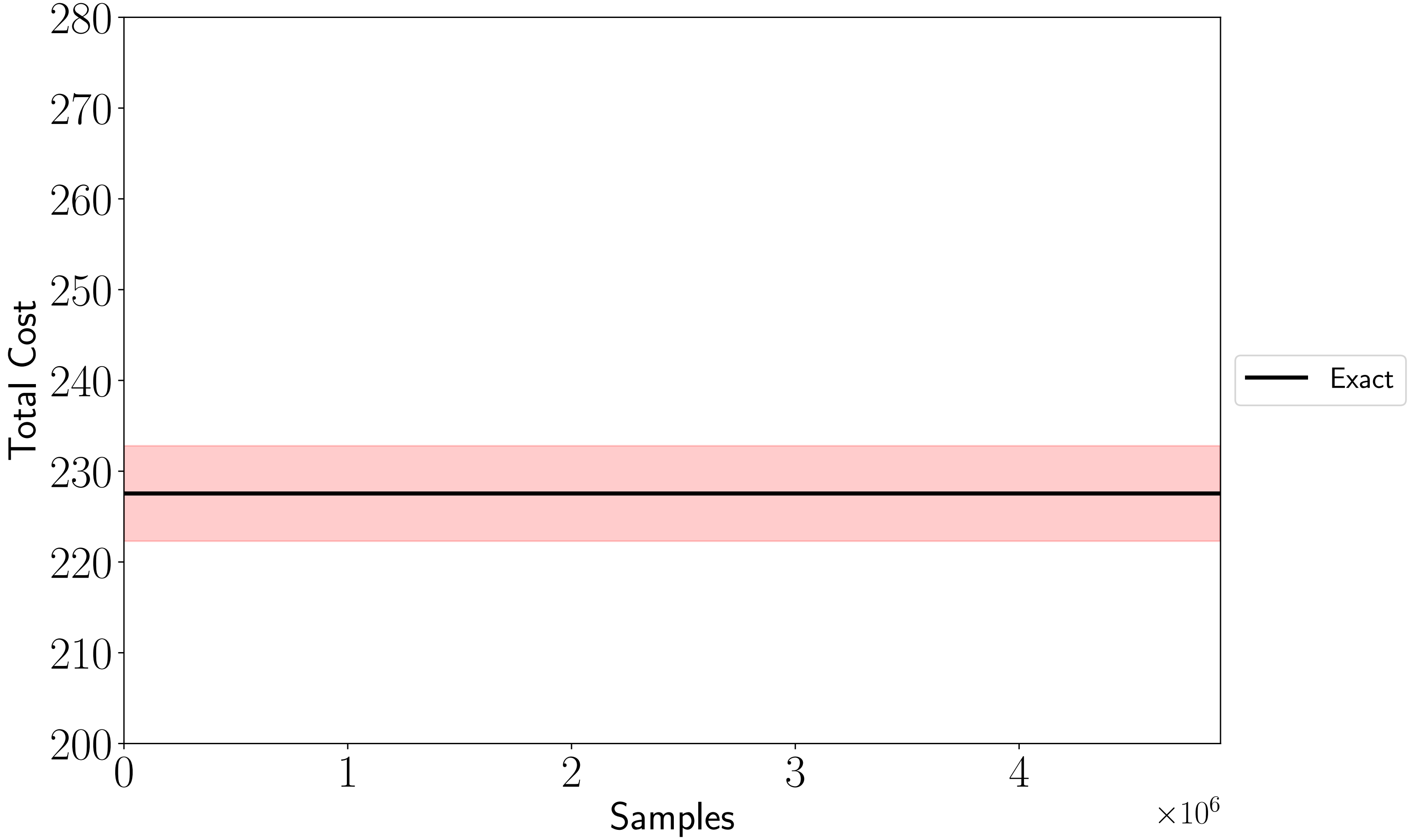
$$J_\theta - J_\theta^\rho \leq \cdots \leq \frac{\gamma}{(1-\gamma)} L_\theta \rho$$

Approximation error bounded by radius of approximation

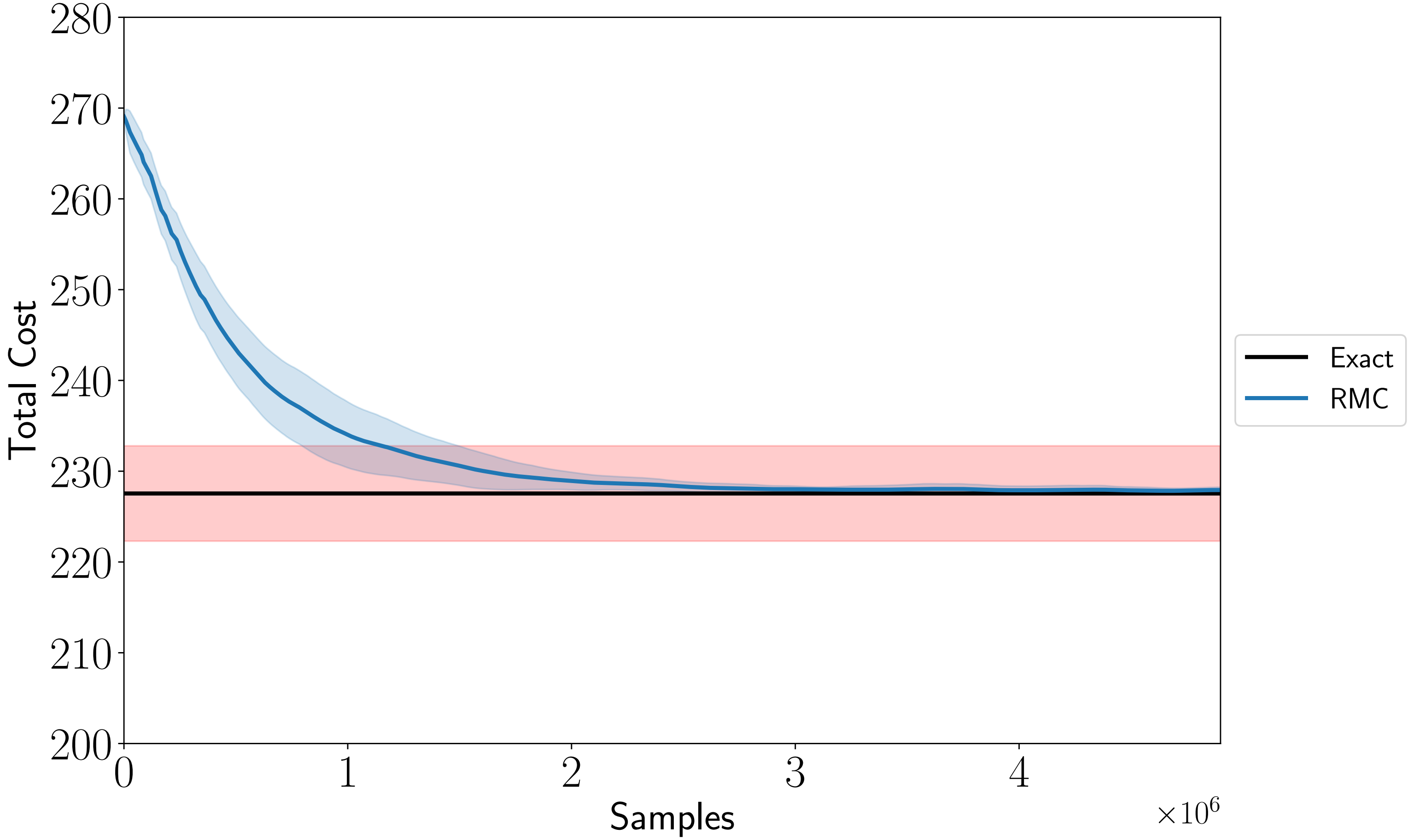
E.g. Inventory management



E.g. Inventory management



E.g. Inventory management



Conclusion

Conclusion

- RMC useful in problems where:
  - renewal time is small
  - structure of optimal policy is known
  - reset actions are present

Conclusion

- RMC useful in problems where:
  - renewal time is small
  - structure of optimal policy is known
  - reset actions are present
- Not so useful in arbitrary high dimensional problems

Conclusion

- RMC useful in problems where:
  - renewal time is small
  - structure of optimal policy is known
  - reset actions are present
- Not so useful in arbitrary high dimensional problems
- In high dimensional problems:
  - RMC can be used as a sub-component of main scheme
  - in the presence of hierarchies, can be used in a level with short renewals

Thank you