Queueing Network Controls via Deep Reinforcement Learning

J. G. Dai, Mark Gluzman

Model: Discrete-time MDP with Long-run Average Cost Objective

- an MDP with countable state space X, finite action space A, one-step cost $g(x) \geq 0$ and the transition function P(y|x,a)

- Consider a class of randomized Markovian policies $\pi_\theta, \theta \in \Theta$. Under the policy π_θ, the transition matrix $P_\theta(y \mid x) = \sum_{a \in \mathscr{A}} \pi_\theta(a \mid x) P(y \mid x, a)$ for $x, y \in \mathscr{X}$

- Assume each Markov chain P_θ is irreducible and aperiodic

- Find θ to minimise the long-run average cost

$$\limsup_{N \to \infty} \frac{1}{N} \mathbb{E}_{\pi_\theta} \left[\sum_{k=0}^{N-1} g\left(x^{(k)}\right) \right]$$

Which is independent of the initial state, $x^{(k)}$ is the state of the Markov chain P_θ after k time steps

Remark

1. On going operations: long-run average cost is appropriate while most algorithm focus on discount
2. System load can be high, leading to large or infinite state space
3. One-step cost function g can be unbounded, leading to many complexity analyses invalid
4. Heavy load leads to long regenerative cycles, variance reduction techniques are required

Contributions

1. The Lyapunov function theory of Meyn-Tweedie is an important tool to justify the PPO algorithm for long-run
2. average cost problems with infinite state space and unbounded one-step cost

Poisson equation

- Assume that the Markov chain P_θ has the stationary distribution, which is denoted by μ_θ
- Long-run average cost is $\mu_\theta^T g$
- Assume that Poisson equation has a solution $h_\theta = h$

$$g(x) - \mu_\theta^T g + \sum_{y \in \mathcal{X}} P_\theta(y \mid x) h(y) - h(x) = 0 \quad \text{for each } x \in \mathcal{X}$$

- In discount setting, this is just Bellman equation
- An advantage function $A_\theta : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ of policy π_θ

$$A_\theta(x, a) := \mathop{\mathbb{E}}_{y \sim P(\cdot|x,a)} \left[\underbrace{g(x) - \mu_\theta^T g + h_\theta(y)}_{Q(x,a)} - \underbrace{h_\theta(x)}_{V(x)} \right]$$

- When X is finite, both assumptions are satisfied.

Trust region policy optimisation (Shulman et al., 2015)

Kakade-Langform (2001): discount factor $\gamma \in [0,1)$

$$c_\gamma(\pi_\theta) - c_\gamma(\pi_\eta) = \frac{1}{(1-\gamma)^2}\mathbb{E}_{x\in d_{\pi_\theta}, a\sim\pi_\theta(\cdot|x)}\left[A_{\pi_\eta}(x,a)\right]$$
$$= \frac{1}{(1-\gamma)^2}\mathbb{E}_{x\sim d_{\pi\eta}, a\sim\pi_\eta(\cdot|x)}\left[\frac{\pi_\theta(x,a)}{\pi_\eta(x,a)}A_{\pi_\eta}(x,a)\right] + \Delta(\pi_\eta,\pi_\theta)$$
$$\equiv L(\pi_\eta,\pi_\theta) + \Delta(\pi_\eta,\pi_\theta)$$

Schulman et al. (2015)

$$c_\gamma(\pi_\theta, s) - c_\gamma(\pi_\eta, s) \le \underbrace{L(\pi_\eta,\pi_\theta) + \frac{4\epsilon}{(1-\gamma)^2}\alpha^2}_{\text{surrogate objective}}$$

where $\epsilon = \max_{(x,a)\in\mathcal{X}\times\mathcal{A}}\left|A_{\pi_\eta}(x,a)\right|$ and $\alpha = d_{\text{TV}}^{\max}(\pi_\eta,\pi_\theta)$

Remark:

- Although long-run problem can be approximate by discount one, the coefficient $\dfrac{4\epsilon}{(1-\gamma)^2}$ would become extreme large and force the update to be small
- When the one-step cost function g is unbounded, ϵ would be problematic
- Trust region algorithm: minimise $L(\eta,\theta)$ subject to $d_{KL}^{max}(\eta,\theta) \le \delta$, where $\delta > 0$ is a hyper-parameter

When the state space is infinite: drift condition

Lemma 1. *Consider an irreducible Markov chain on a countable state space $\mathcal{X}$ with a transition matrix P on $\mathcal{X} \times \mathcal{X}$. Assume there exists a vector $\mathcal{V} : \mathcal{X} \to [1, \infty)$ such that the following drift condition holds for some constants $b \in (0,1)$ and $d \geq 0$, and a finite subset $C \subset \mathcal{X}$:*

$$\sum_{y \in \mathcal{X}} P(y|x)\mathcal{V}(y) \leq b\mathcal{V}(x) + d\mathbb{I}_C(x), \quad \text{for each } x \in \mathcal{X}, \tag{3.1}$$

where $\mathbb{I}_C(x) = 1$ if $x \in C$ and $\mathbb{I}_C(x) = 0$ otherwise. Here, $P(y|x) = P(x,y)$ is the transition probability from state $x \in \mathcal{X}$ to state $y \in \mathcal{X}$. Then (a) the Markov chain with the transition matrix P is positive recurrent with a unique stationary distribution μ; and (b) $\mu^T \mathcal{V} < \infty$, where for any function $f : \mathcal{X} \to \mathbb{R}$ we define $\mu^T f$ as

$$\mu^T f := \sum_{x \in \mathcal{X}} \mu(x) f(x).$$

(Meyn-Tweedie) Assume policy π_η is such that P satisfies the drift condition with Lyapunov function $V \geq 1$.

Lemma 2. *Consider a $\mathcal{V}$-uniformly ergodic Markov chain with transition matrix P and the stationary distribution μ. For any cost function $g : \mathcal{X} \to \mathbb{R}$ satisfying $|g| \leq \mathcal{V}$, Poisson equation (3.3) admits a fundamental solution*

$$h^{(f)}(x) := \mathbb{E}\left[\sum_{k=0}^{\infty} \left(g(x^{(k)}) - \mu^T g\right) \mid x^{(0)} = x\right] \quad \text{for each } x \in \mathcal{X}, \tag{3.4}$$

where $x^{(k)}$ is the state of the Markov chain after k timesteps.

The drift condition (3.1) is sufficient and necessary for an irreducible, aperiodic Markov chain to be V-uniformly ergodic

for any g : X → R with | g(x) | ≤ V(x) for x ∈ X, there exist constants R < ∞ and r < 1 such that

$$\left| \sum_{y \in \mathcal{X}} P^n(y \mid x) g(y) - \mu^T g \right| \leq R\mathcal{V}(x) r^n \qquad \text{for any } x \in X \text{ and } n \geq 0;$$

Main Results

Lemma 4. *Fix an $\eta \in \Theta$. We assume that drift condition (3.1) holds for P_η. Let some $\theta \in \Theta$ satisfies,*

$$\|(P_\theta - P_\eta)Z_\eta\|_\mathcal{V} < 1,$$

then the Markov chain with transition matrix P_θ has a unique stationary distribution μ_θ.

Theorem 1. *Suppose that the Markov chain with transition matrix P_η is an irreducible chain such that the drift condition (3.1) holds for some function $\mathcal{V} \geq 1$ and the cost function satisfies $|g| < \mathcal{V}$.*

For any $\theta \in \Theta$ such that

$$D_{\theta,\eta} := \|(P_\theta - P_\eta)Z_\eta\|_\mathcal{V} < 1 \tag{3.8}$$

the difference of long-run average costs of policies π_θ and π_η is bounded by:

$$\mu_\theta^T g - \mu_\eta^T g \ \leq \ N_1(\theta,\eta) + N_2(\theta,\eta), \tag{3.9}$$

where $N_1(\theta,\eta)$, $N_2(\theta,\eta)$ are finite and equal to

$$N_1(\theta,\eta) := \mu_\eta^T(g - (\mu_\eta^T g)e + P_\theta h_\eta - h_\eta), \tag{3.10}$$

$$N_2(\theta,\eta) := \frac{D_{\theta,\eta}^2}{1 - D_{\theta,\eta}}\left(1 + \frac{D_{\theta,\eta}}{(1 - D_{\theta,\eta})}(\mu_\eta^T\mathcal{V})\|I - \Pi_\eta + P_\eta\|_\mathcal{V}\|Z_\eta\|_\mathcal{V}\right)\|g - (\mu_\eta^T g)e\|_{\infty,\mathcal{V}}(\mu_\eta^T\mathcal{V}), \tag{3.11}$$

where, for a vector ν on $\mathcal{X}$, $\mathcal{V}$-norm is defined as

$$\|\nu\|_{\infty,\mathcal{V}} := \sup_{x\in\mathcal{X}} \frac{|\nu(x)|}{\mathcal{V}(x)}. \tag{3.12}$$

- Denote: $H(\theta) = N_1(\theta,\eta) + N_2(\theta,\eta)$, If $H(\theta) < 0$, π_θ is a strict improvement over policy π_η
- Minimization of $H(\theta)$ is difficult
- When $D_{\theta,\eta}$ is small, $L(\theta,\eta) = O(D_{\theta,\eta})$ and $\delta(\theta,\eta) = O(D_{\theta,\eta}^2)$. Thus, if $L(\theta,\eta) < 0$, the surrogate function is likely negative
- Conservative update: minimise $L(\theta,\eta)$ while keeping $D_{\theta,\eta}$ is small

- When $D_{\theta,\eta}$ is small, $L(\theta,\eta) = O(D_{\theta,\eta})$ and $\delta(\theta,\eta) = O(D^2_{\theta,\eta})$. Thus, if $L(\theta,\eta) < 0$, the surrogate function is likely negative

$$
\begin{aligned}
|N_1(\theta,\eta)| :&= \left|\mu_\eta^T(g - (\mu_\eta^T g)e + P_\theta h_\eta - h_\eta)\right| \\
&\leq (\mu_\eta^T \mathcal{V})\|g - (\mu_\eta^T g)e + P_\theta h_\eta - h_\eta\|_{\infty,\mathcal{V}} \\
&= (\mu_\eta^T \mathcal{V})\|(P_\theta - P_\eta)h_\eta\|_{\infty,\mathcal{V}} \\
&= (\mu_\eta^T \mathcal{V})\|(P_\theta - P_\eta)Z_\eta\left(g - (\mu_\eta^T g)e\right)\|_{\infty,\mathcal{V}} \\
&\leq (\mu_\eta^T \mathcal{V})\|g - (\mu_\eta^T g)e\|_{\infty,\mathcal{V}} D_{\theta,\eta}.
\end{aligned}
$$

Lemma 5.

$$
D_{\theta,\eta} \leq \|Z_\eta\|_{\mathcal{V}} \sup_{x\in\mathcal{X}} \sum_{a\in\mathcal{A}} |r_{\theta,\eta}(a|x) - 1|\, G_\eta(x,a),
$$

where $G_\eta(x,a) := \frac{1}{\mathcal{V}(x)} \sum_{y\in\mathcal{X}} \pi_\eta(a|x) P(y|x,a)\mathcal{V}(y)$.

The lemma says that $D_{\theta,\eta}$ is small when the ratio $r_{\theta,\eta}(a\,|\,x)$ is close to 1

$$
\begin{aligned}
N_1(\theta,\eta) &= \mu_\eta^T(g - (\mu_\eta^T g)e + P_\theta h_\eta - h_\eta) \\
&= \mathop{\mathbb{E}}_{\substack{x\sim\mu_\eta \\ a\sim\pi_\theta(\cdot|x) \\ y\sim P(\cdot|x,a)}} \left[g(x) - (\mu_\eta^T g)e + h_\eta(y) - h_\eta(x)\right] \\
&= \mathop{\mathbb{E}}_{\substack{x\sim\mu_\eta \\ a\sim\pi_\theta(\cdot|x)}} A_\eta(x,a) \\
&= \mathop{\mathbb{E}}_{\substack{x\sim\mu_\eta \\ a\sim\pi_\eta(\cdot|x)}} \left[\frac{\pi_\theta(a|x)}{\pi_\eta(a|x)} A_\eta(x,a)\right] = \mathop{\mathbb{E}}_{\substack{x\sim\mu_\eta \\ a\sim\pi_\eta(\cdot|x)}} \left[r_{\theta,\eta}(a|x) A_\eta(x,a)\right],
\end{aligned}
$$

where we define an advantage function $A_\eta : \mathcal{X}\times\mathcal{A} \to \mathbb{R}$ of policy π_η, $\eta\in\Theta$ as:

$$
A_\eta(x,a) := \mathop{\mathbb{E}}_{y\sim P(\cdot|x,a)} \left[g(x) - \mu_\eta^T g + h_\eta(y) - h_\eta(x)\right].
$$

Following Schulman et al., solve an unconstrained optimisation problem by minimising the clipped surrogate objective over θ

$$L^{\epsilon}(\theta, \eta) := \mathop{\mathbb{E}}_{x \sim \mu_\eta(\cdot|x), a \sim \pi_\eta(\cdot|x)} \max \left[r_{\theta,\eta}(a \mid x) A_\eta(x, a), \operatorname{clip}\left(r_{\theta,\eta}(a \mid x), 1-\epsilon, 1+\epsilon \right) A_\eta(x, a) \right]$$

where $\epsilon > 0$ is a hyper-parameter

$$\operatorname{clip}(c, 1-\epsilon, 1+\epsilon) := \begin{cases} 1-\epsilon, & \text{if } c < 1-\epsilon, \\ 1+\epsilon, & \text{if } c > 1+\epsilon, \\ c, & \text{otherwise,} \end{cases}$$

PPO: Markov Chain Monte Carlo

Under policy π_η an episode is generated:

$$E = \{x^0, a^0, x^1, a^1, \cdots, x^{K-1}, a^{K-1}\}$$

Based on the generated episode, the Monte-Carlo estimate of $L^\epsilon(\theta, \eta)$ is

$$\hat{L}\left(\theta, \eta, D^{(0:N-1)}\right) = \frac{1}{N}\sum_{k=0}^{N-1} \max\left[\frac{\pi_\theta\left(a^{(k)} \mid x^{(k)}\right)}{\pi_\eta\left(a^{(k)} \mid x^{(k)}\right)}\hat{A}_\eta\left(x^{(k)}, a^{(k)}\right), \mathrm{clip}\left(\frac{\pi_\theta\left(a^{(k)} \mid x^{(k)}\right)}{\pi_\eta\left(a^{(k)} \mid x^{(k)}\right)}, 1-\epsilon, 1+\epsilon\right)\hat{A}_\eta\left(x^{(k)}, a^{(k)}\right)\right]$$

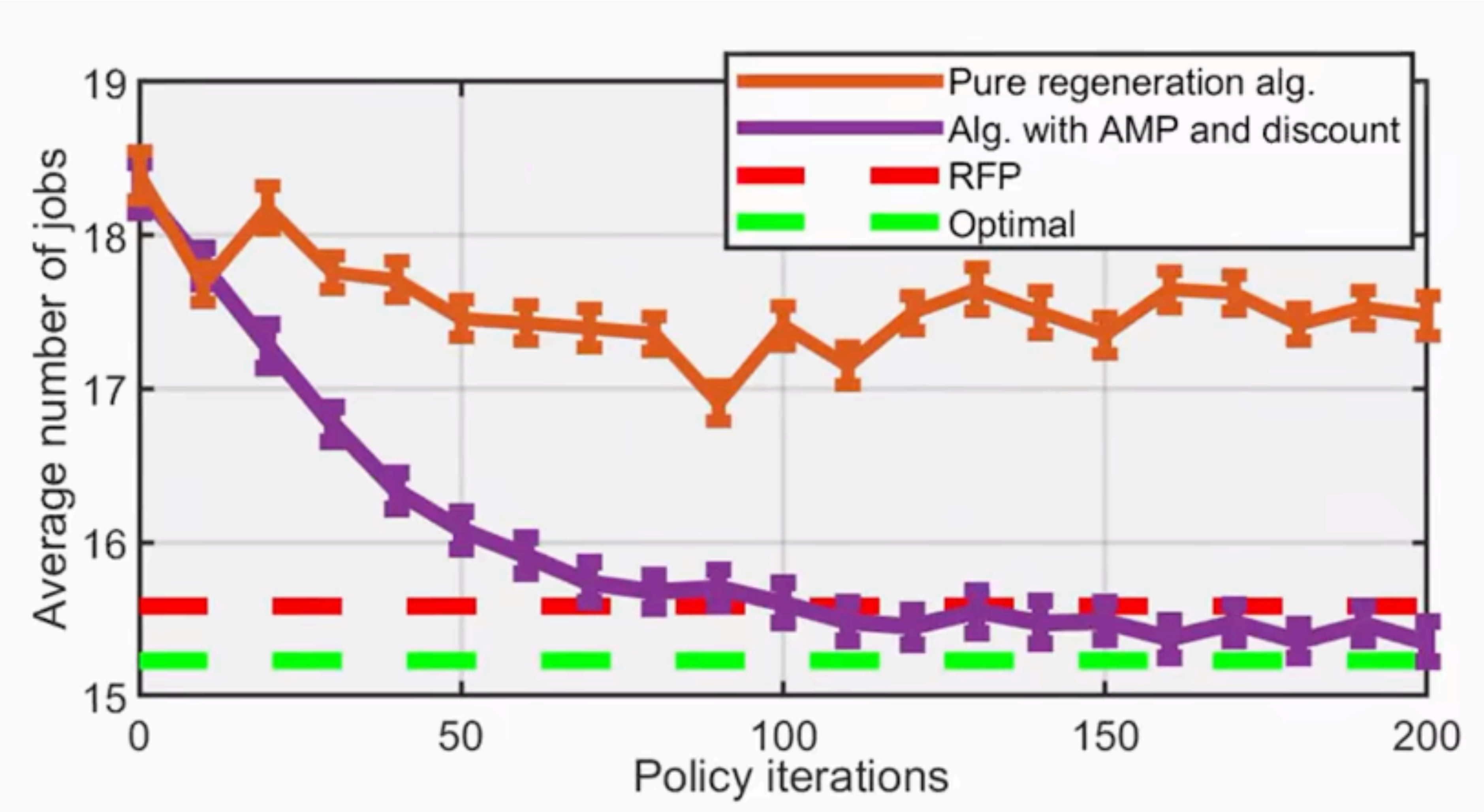
$$A_\eta(x, a) := \mathop{\mathbb{E}}_{y \sim P(\cdot|x,a)}\left[g(x) - \mu_\eta^T g + h_\eta(y) - h_\eta(x)\right]$$

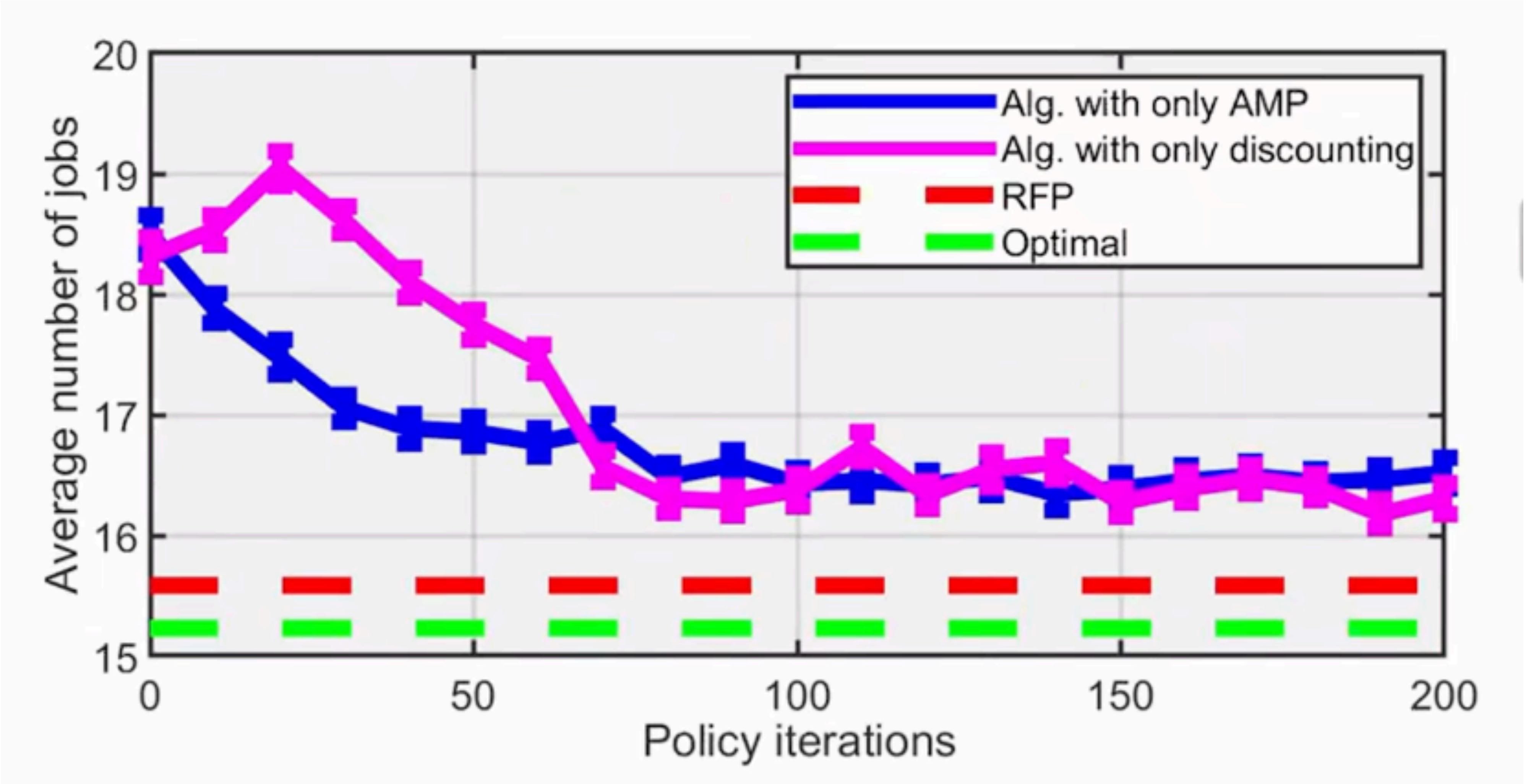
- Use neural network to parametrised the policy π_θ.
- Use neural network to approximate advantage function (actor-critic framework)
- Use pytorch/tensorflow to solve the optimisation problem and update the policy
- The key is to estimate the advance function, i.e., $h(x)$
- Assume the MDP is known, so the expectation can be computed exactly (Model-based)
- Requires one to estimate $h(y)$ for some states that have not been visited in the simulation
- Use Monte Carlo method to estimate $h(y)$ at a selected subset of y's, and then use an approximate $f(y)$ to replace $h(y)$. The latter is standard in learning

Variance reduction in value function estimation

Several variance reduction techniques to estimate h:

- Regenerative simulation
- Discounting
- Approximate martingale process (AMP) control variate, Henderson-Glynn(2002)





Regenerative simulation

$$h^{(f)}(x) := \mathbb{E}\left[\sum_{k=0}^{\infty}\left(g\left(x^{(k)}\right) - \mu^T g\right) \mid x^{(0)} = x\right] \text{ for each } x \in \mathcal{X}$$

where $x^{(k)}$is the state of the Markov chain after k timesteps.

Poisson equation admits infinitely many solutions. Let x^* be a recurrent state. Then Poisson's equation

$$g(x) - \mu^T g + \sum_{y \in \mathcal{X}} P(y \mid x)h(y) - h(x) = 0 \quad \text{for each } x \in \mathcal{X}$$

admits a solution

$$h^{(x^*)}(x) := \mathbb{E}\left[\sum_{k=0}^{\sigma(x^*)-1}\left(g\left(x^{(k)}\right) - \mu^T g\right) \mid x^{(0)} = x\right] \text{ for each } x \in \mathcal{X}$$

assume that an episode consisting of N regenerative cycles, has been generated under policy π η, where x (0)=x^*

$$\left\{x^{(0)}, x^{(1)}, \cdots, x^{(\sigma_1)}, \cdots, x^{(\sigma_N - 1)}\right\}$$

$$A_\eta(x, a) := \mathop{\mathbb{E}}_{y \sim P(\cdot|x,a)}\left[g(x) - \mu_\eta^T g + h_\eta(y) - h_\eta(x)\right]$$

$$\hat{A}_\eta\left(x^{(k)}, a^{(k)}\right) := g\left(x^{(k)}\right) - \widehat{\mu_\eta^T g} + \sum_{y \in \mathcal{X}} P\left(y \mid x^{(k)}, a^{(k)}\right) f_{\psi^*}(y) - f_{\psi^*}\left(x^{(k)}\right)$$

$$\widehat{\mu_\eta^T g} := \frac{1}{\sigma(N)} \sum_{k=0}^{\sigma(N)-1} g\left(x^{(k)}\right)$$

where σ(n) is the nth time when regeneration state x^* is visited.

$$\hat{h}_k := \sum_{t=k}^{\sigma_k - 1}\left(g\left(x^{(t)}\right) - \widehat{\mu_\eta^T g}\right)$$

where $\sigma(k)$= min { t > k | x (t) =x^*} is the first time when the regeneration state x^* is visited after time k

$$\psi^* = \arg\min_{\psi \in \Psi} \sum_{k=0}^{\sigma(N)-1}\left(f_\psi\left(x^{(k)}\right) - \hat{h}_k\right)^2$$

where $x^{(k)}$ is the state of the Markov chain P after k timesteps

and $\sigma\left(x^*\right) = \min\left\{k > 0 \mid x^{(k)} = x^*\right\}$is the first future time when state x^* is visited

Regenerative simulation (parallel version)

$$\left\{x^{(0,q)}, a^{(0,q)}, x^{(1,q)}, a^{(1,q)}, \cdots, x^{(k,q)}, a^{(k,q)}, \cdots, x^{\left(\sigma^q(N)-1,q\right)}, a^{\left(\sigma^q(N)-1,q\right)}\right\}$$

$$\hat{L}\left(\theta, \theta_i, D^{(1:Q),(0:\sigma^q(N)-1)}\right) = \sum_{q=1}^{Q} \sum_{k=0}^{\sigma^q(N)-1} \max\left[\frac{\pi_\theta\left(a^{(k,q)}|x^{(k,q)}\right)}{\pi_{\theta_i}\left(a^{(k,q)}|x^{(k,q)}\right)} \hat{A}_{\theta_i}\left(x^{(k,q)}, a^{(k,q)}\right),\ \operatorname{clip}\left(\frac{\pi_\theta\left(a^{(k,q)}|x^{(k,q)}\right)}{\pi_{\theta_i}\left(a^{(k,q)}|x^{(k,q)}\right)}, 1-\epsilon, 1+\epsilon\right) \hat{A}_{\theta_i}\left(x^{(k,q)}, a^{(k,q)}\right)\right]$$

Algorithm 1: Base proximal policy optimization algorithm for long-run average cost problems

Result: policy π_{θ_I}

Initialize policy π_{θ_0} ;

for *policy iteration* $i = 0, 1, ..., I-1$ **do**

  for *actor* $q = 1, 2, ..., Q$ **do**

    Run policy π_{θ_i} until it reaches Nth regeneration time on $\sigma^q(N)$ step: collect an episode $\{x^{(0,q)}, a^{(0,q)}, x^{(1,q)}, a^{(1,q)}, \cdots, x^{(\sigma^q(N)-1,q)}, a^{(\sigma^q(N)-1,q)}, x^{(\sigma^q(N),q)}\}$;

  end

  Compute the average cost estimate $\widehat{\mu_{\theta_i}^T g}$ by (4.2) (utilizing Q episodes) ;

  Compute $\hat{h}_{k,q}$, the estimate of $h_{\theta_i}(x^{(k,q)})$, by (4.3) for each $q = 1, .., Q$, $k = 0, .., \sigma^q(N)-1$;

  Update $\psi_i := \psi$, where $\psi \in \Psi$ minimizes $\sum\limits_{q=1}^{Q} \sum\limits_{k=0}^{\sigma^q(N)-1} \left(f_\psi(x^{(k,q)}) - \hat{h}_{k,q}\right)^2$ following (4.4) ;

  Estimate th eadvantage functions $\hat{A}_{\theta_i}\left(x^{(k,q)}, a^{(k,q)}\right)$ using (4.5) for each $q = 1, .., Q$, $k = 0, .., \sigma^q(N)-1$:

$$D^{(1:Q),(0:\sigma^q(N)-1)} = \left\{\left(x^{(0,q)}, a^{(0,q)}, \hat{A}_{0,q}\right), \cdots, \left(x^{(\sigma^q(N)-1,q)}, a^{(\sigma^q(N)-1,q)}, \hat{A}_{\sigma^q(N)-1,q}\right)\right\}_{q=1}^{Q}.$$

  Minimize the surrogate objective function w.r.t. $\theta \in \Theta$:

$$\hat{L}\left(\theta, \theta_i, D^{(1:Q),(0:\sigma^q(N)-1)}\right) = \sum_{q=1}^{Q} \sum_{k=0}^{\sigma^q(N)-1} \max\left[\frac{\pi_\theta(a^{(k,q)}|x^{(k,q)})}{\pi_{\theta_i}(a^{(k,q)}|x^{(k,q)})} \hat{A}_{\theta_i}(x^{(k,q)}, a^{(k,q)}),\ \operatorname{clip}\left(\frac{\pi_\theta(a^{(k,q)}|x^{(k,q)})}{\pi_{\theta_i}(a^{(k,q)}|x^{(k,q)})}, 1-\epsilon, 1+\epsilon\right) \hat{A}_{\theta_i}(x^{(k,q)}, a^{(k,q)})\right]$$

  Update $\theta_{i+1} := \theta$.

end

Approximating martingale-process method

Intuition

From the definition of a solution to the Poisson equation

$$g\left(x^{(k)}\right) - \mu_\eta^T g = h_\eta\left(x^{(k)}\right) - \sum_{y\in\mathcal{X}} P_\eta\left(y \mid x^{(k)}\right) h_\eta(y)$$

If the approximation ζ is sufficiently close to h η , then the correlation between

$$g\left(x^{(k)}\right) - \widehat{\mu_\eta^T g} \quad \text{and} \quad \zeta\left(x^{(k)}\right) - \sum_{y\in X} P_\eta\left(y \mid x^{(k)}\right) \zeta(y)$$

is positive and we can use the control variate to reduce the variance

Consider the martingale process starting from an arbitrary state $x^{(k)}$ until the first regeneration time

$$M_{\sigma_k}\left(x^{(k)}\right) = \zeta\left(x^{(k)}\right) + \sum_{t=k}^{\sigma_k-1}\left[\sum_{y\in\mathcal{X}} P_\eta\left(y \mid x^{(t)}\right)\zeta(y) - \zeta\left(x^{(t)}\right)\right]$$

EM_n= 0 for all n ≥ 0

Adding M_{σ_k} estimator to $\hat{h}_k := \sum_{t=k}^{\sigma_k-1}\left(g\left(x^{(t)}\right) - \widehat{\mu_\eta^T g}\right)$

$$\hat{h}_\eta^{AMP(\zeta)}\left(x^{(k)}\right) := \zeta\left(x^{(k)}\right) + \sum_{t=k}^{\sigma_k-1}\left(g\left(x^{(t)}\right) - \widehat{\mu_\eta^T g} + \sum_{y\in\mathcal{X}} P_\eta\left(y \mid x^{(t)}\right)\zeta(y) - \zeta\left(x^{(t)}\right)\right)$$

Adopting a biased estimator through discounting

Regenerative cycles can be long, leading to high variance in estimating h

E.g. cycle length of M/M/1 queue with server utilization $\rho = \lambda/\mu < 1$

$$\mathbb{E}\left[\sigma\left(x^* = 0\right)\right] = O\left(\frac{1}{1-\rho}\right) \text{ as } \rho \uparrow 1$$

Replace

$$h^{(x^*)}(x) := \mathbb{E}\left[\sum_{k=0}^{\sigma(x^*)-1}\left(g\left(x^{(k)}\right) - \mu^T g\right) \mid x^{(0)} = x\right] \text{ for each } x \in \mathcal{X}$$

By

$$r^{(\gamma)}\left(x^*\right) := (1-\gamma)\mathbb{E}\left[\sum_{k=0}^{\infty}\gamma^k g\left(x^{(k)}\right) \mid x^{(0)} = x^*\right]$$

$$V^{(\gamma)}(x) := \mathbb{E}\left[\sum_{k=0}^{\sigma(x^*)-1}\gamma^k\left(g\left(x^{(k)}\right) - r^{(\gamma)}\left(x^*\right)\right) \mid x^{(0)} = x\right] \text{ for each } x \in \mathcal{X}$$

Interpretation

Introducing the discount factor γ can be interpreted as a modification of the original transition dynamics

Consider a modified Markov reward process with transition kernel$\tilde{P}, \gamma$ for each$x \in \mathcal{X}$

$$\begin{cases} \tilde{P}(y \mid x) = \gamma P(y \mid x) & \text{for } y \neq x^* \\ \tilde{P}\left(x^* \mid x\right) = \gamma P\left(x^* \mid x\right) + (1-\gamma) \end{cases}$$

Therefore, discounting forces faster regeneration

Further variance reduction: T-step truncation

$$\hat{V}^{(\gamma)}(x) = \sum_{t=0}^{T-1} \gamma^t \left(g\left(x^{(t)}\right) - \widehat{r\left(x^*\right)} \right) + \gamma^T \underbrace{\sum_{t=0}^{\sigma(x^*)-1} \gamma^t \left(g\left(x^{(T+t)}\right) - \widehat{r\left(x^*\right)} \right)}_{\hat{V}^{(\gamma)}(x^T)}$$

Instead of estimating the value at state $x^{(T)}$ by a random roll-out, we can use the value of deterministic approximation function ζ at state $x^{(T)}$

The T-step truncation reduces the variance of the standard estimator but introduces bias unless the approximation

Algorithm 3: Proximal policy optimization with discounting

Result: policy π_{θ_I}

Initialize policy π_{θ_0} and value function $f_{\psi_{-1}} \equiv 0$ approximators ;

for *policy iteration* $i = 0, 1, ..., I-1$ **do**

    for *actor* $q = 1, 2, ..., Q$ **do**

        Run policy π_{θ_i} for $N+L$ timesteps: collect an episode $\{x^{(0,q)}, a^{(0,q)}, x^{(1,q)}, a^{(1,q)}, ..., x^{(N+L-1,q)}, a^{(N+L-1,q)}\}$;

    end

    Estimate the average cost $\widehat{\mu_{\theta_i}^T g}$ by (4.2), the present discounted value $\widehat{r_{\theta_i}(x^*)}$ by (4.21) below;

    Compute $\hat{V}_{k,q}^{AMP(f_{\psi_{i-1}}),(\gamma,\lambda)}$ estimates by (4.20) for each $q = 1, ..., Q$, $k = 0, .., N-1$;

    Update $\psi_i := \psi$, where $\psi \in \Psi$ minimizes $\sum_{q=1}^{Q} \sum_{k=0}^{N-1} \left(f_\psi\left(x^{(k,q)}\right) - \hat{V}_{k,q}^{AMP(f_{\psi_{i-1}}),(\gamma,\lambda)} \right)^2$ following (4.4) ;

    Estimate the advantage functions $\hat{A}_{\theta_i}\left(x^{(k,q)}, a^{(k,q)}\right)$ using (4.5) for each $q = 1, .., Q$, $k = 0, .., N-1$:

$$D^{(0:N-1)_{q=1}^{Q}} = \left\{ \left(x^{(0,q)}, a^{(0,q)}, \hat{A}_{\theta_i}^{(\gamma)}(x^{(0,q)}, a^{(0,q)}) \right), \cdots, \left(x^{(N-1,q)}, a^{(N-1,q)}, \hat{A}_{\theta_i}^{(\gamma)}(x^{(N-1,q)}, a^{(N-1,q)}) \right) \right\}_{q=1}^{Q}$$

    Minimize the surrogate objective function w.r.t. $\theta \in \Theta$:

$$\hat{L}^{(\gamma)}\left(\theta, \theta_i, D^{(0:N-1)_{q=1}^{Q}}\right) = \sum_{q=1}^{Q} \sum_{k=0}^{N-1} \max\left[\frac{\pi_\theta\left(a^{(k,q)}|x^{(k,q)}\right)}{\pi_{\theta_i}\left(a^{(k,q)}|x^{(k,q)}\right)} \hat{A}_{\theta_i}^{(\gamma)}\left(x^{(k,q)}, a^{(k,q)}\right), \right.$$
$$\left. \operatorname{clip}\left(\frac{\pi_\theta\left(a^{(k,q)}|x^{(k,q)}\right)}{\pi_{\theta_i}\left(a^{(k,q)}|x^{(k,q)}\right)}, 1-\epsilon, 1+\epsilon \right) \hat{A}_{\theta_i}^{(\gamma)}\left(x^{(k,q)}, a^{(k,q)}\right) \right];$$

    Update $\theta_{i+1} := \theta$.

end

Experimental results for multiclass queueing networks: the optimality gap

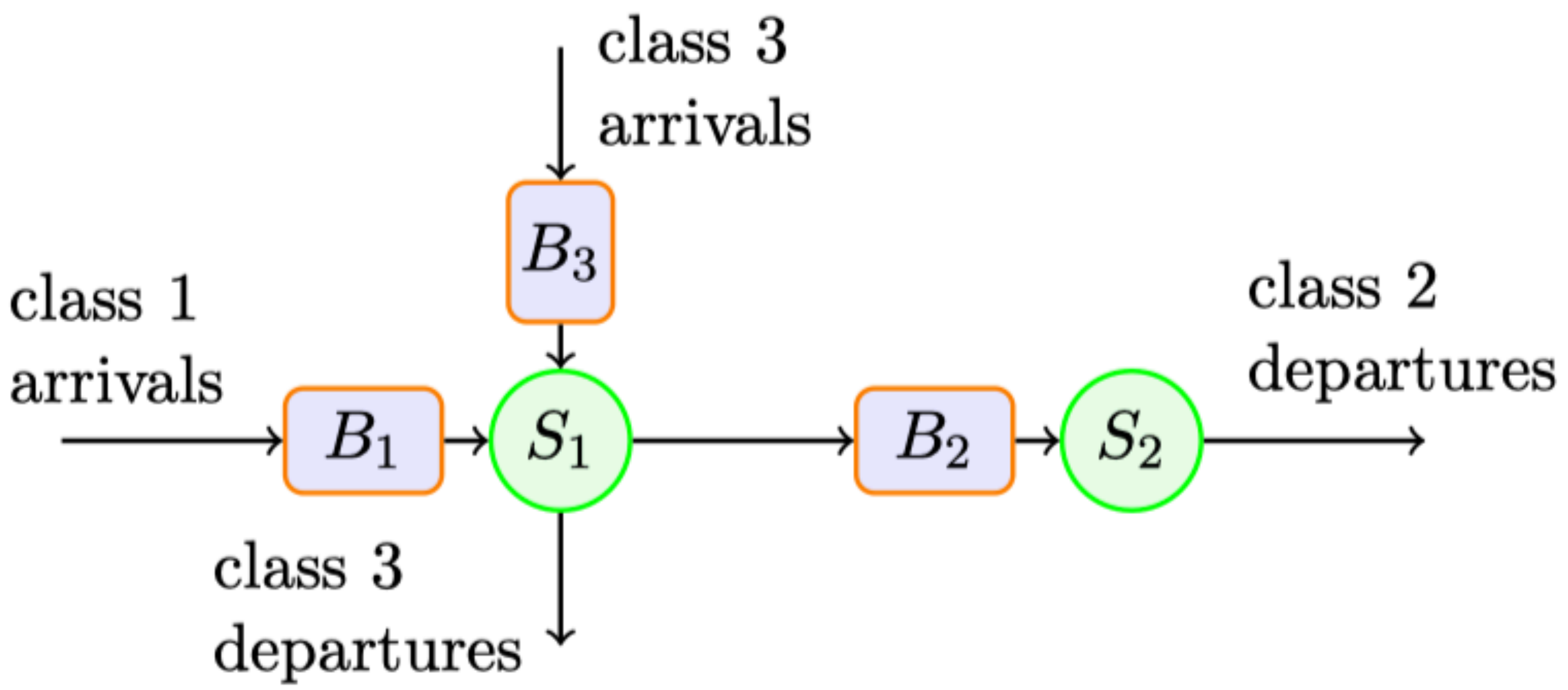


Figure 1: The criss-cross network

Load regime	DP (optimal)	TP	threshold	FP	RFP	PPO (Algorithm 2) with CIs
I.L.	0.671	0.678	0.679	0.678	0.677	0.671 ± 0.001
B.L.	0.843	0.856	0.857	0.857	0.855	0.844 ± 0.004
I.M.	2.084	2.117	2.129	2.162	2.133	2.084 ± 0.011
B.M.	2.829	2.895	2.895	2.965	2.920	2.833 ± 0.010
I.H.	9.970	10.13	10.15	10.398	10.096	10.014 ± 0.055
B.H.	15.228	15.5	15.5	18.430	15.585	16.513 ± 0.140

Table 2: Average number of jobs per unit time in the criss-cross network under different policies. Column 1 reports the variances in the load regimes.

Experimental results for multiclass queueing networks: large state space

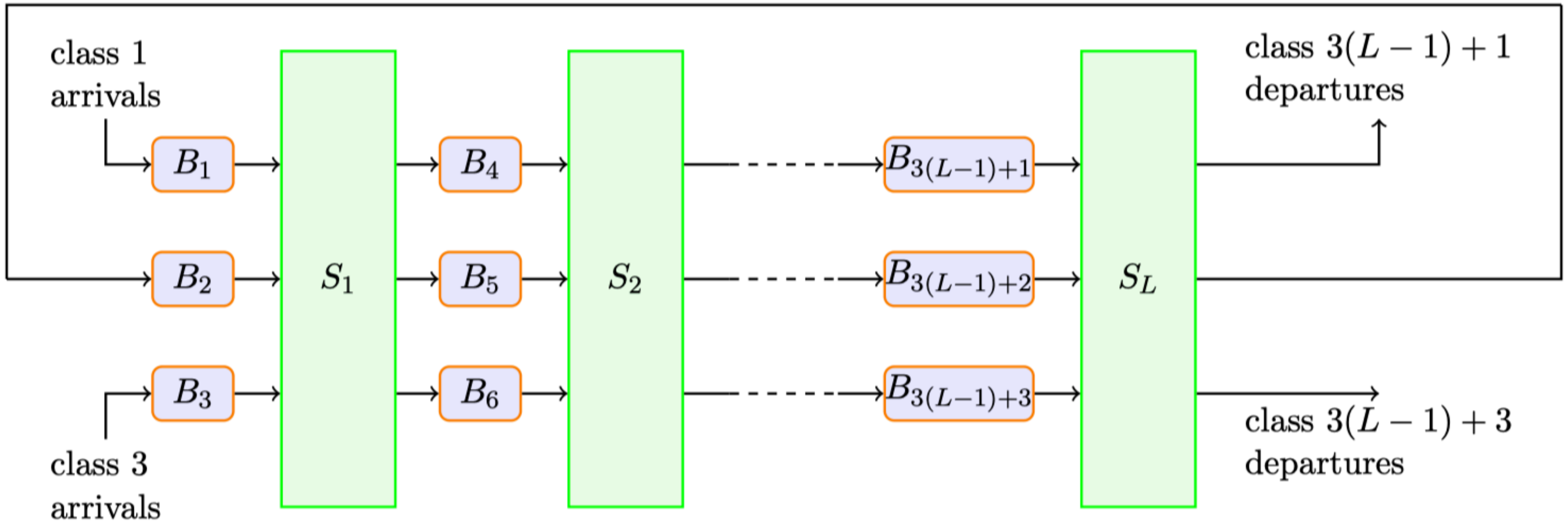


Figure 4: The extended six-class queueing network.

No. of classes 3L	LBFS	FCFS	FP	RFP	PPO (Algorithm 3) with CIs
6	15.749	40.173	15.422	15.286	14.130 ± 0.208
9	25.257	71.518	26.140	24.917	23.269 ± 0.251
12	34.660	114.860	38.085	36.857	32.171 ± 0.556
15	45.110	157.556	45.962	43.628	39.300 ± 0.612
18	55.724	203.418	56.857	52.980	51.472 ± 0.973
21	65.980	251.657	64.713	59.051	55.124 ± 1.807

Table 4: Numerical results for the extended six-class queueing network in Figure 4.

Experimental results for multiclass queueing networks: policy

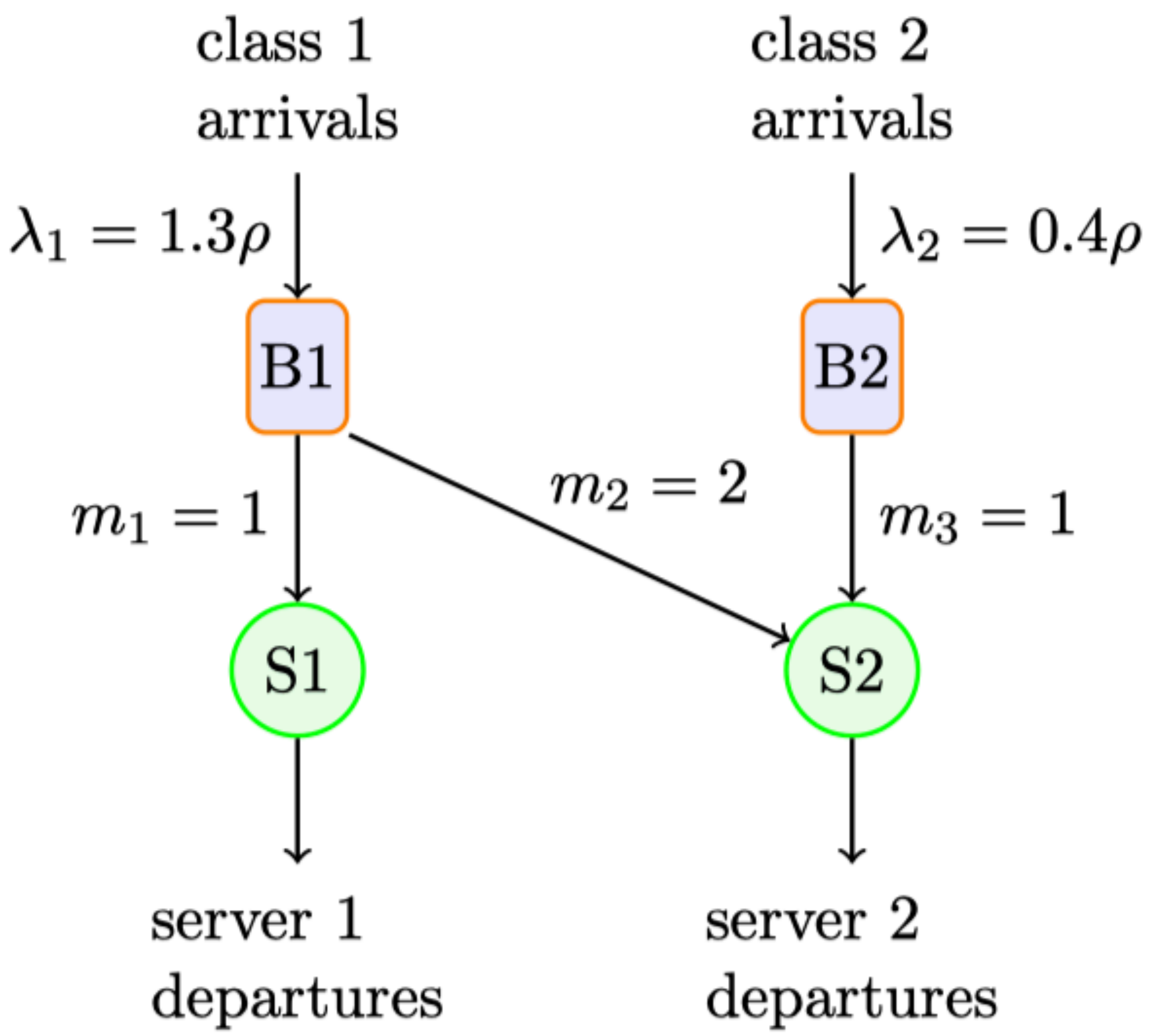
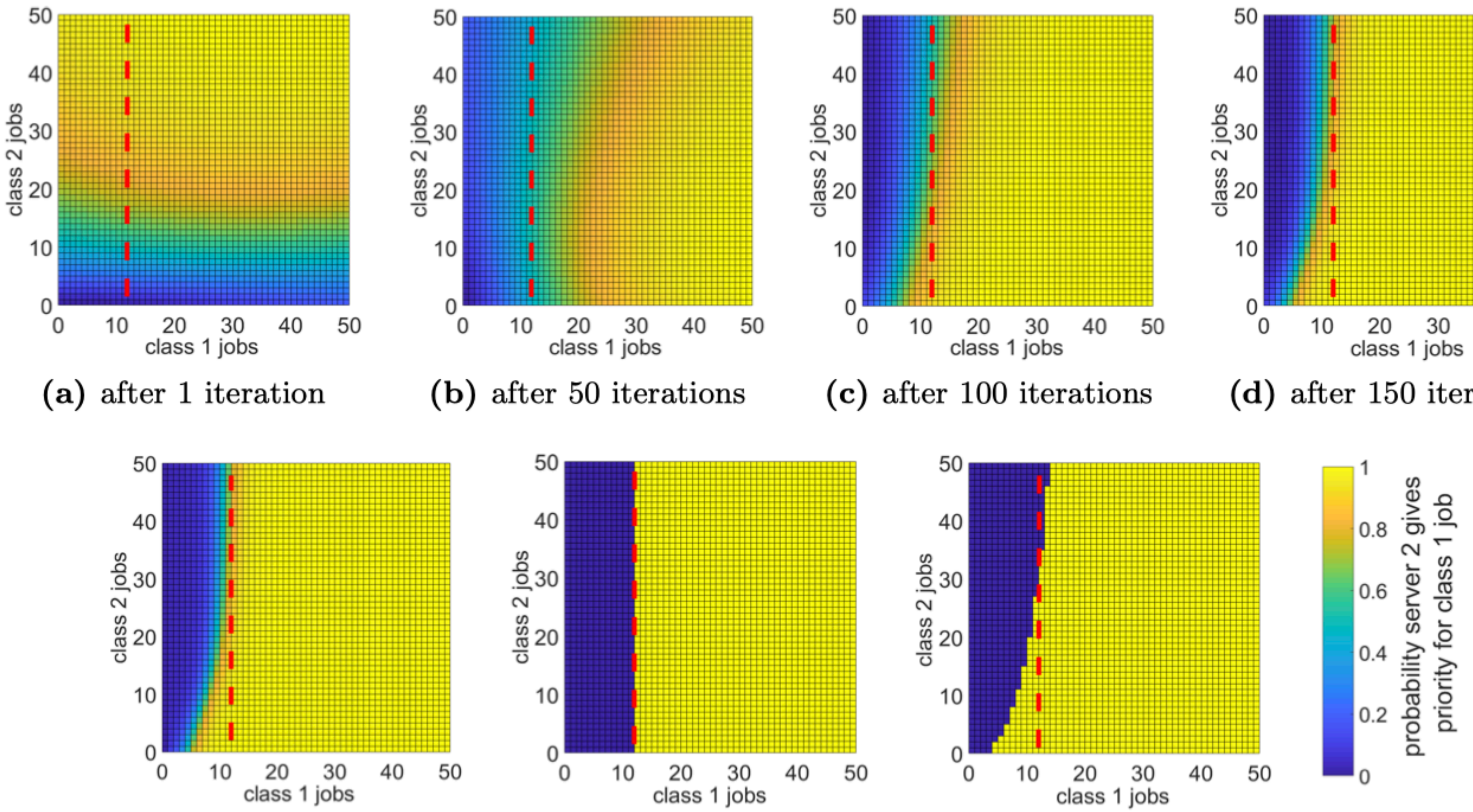


Figure 7: N-model network



(a) after 1 iteration **(b)** after 50 iterations **(c)** after 100 iterations **(d)** after 150 iterations

(e) after 200 iterations **(f)** Threshold policy **(g)** Optimal policy

- Feng, Jiekun, Mark Gluzman, and J. G. Dai. "Scalable Deep Reinforcement Learning for Ride-Hailing." *arXiv preprint arXiv:2009.14679* (2020).

- Oroojlooyjadid, Afshin, et al. "A Deep Q-Network for the Beer Game: A Deep Reinforcement Learning algorithm to Solve Inventory Optimization Problems." *arXiv preprint arXiv:1708.05924* (2017).

- Gijsbrechts, Joren and Boute, Robert N. and Van Mieghem, Jan Albert and Zhang, Dennis, Can Deep Reinforcement Learning Improve Inventory Management? Performance on Dual Sourcing, Lost Sales and Multi-Echelon Problems (October 6, 2020). Available at SSRN: https://ssrn.com/abstract=3302881 or http://dx.doi.org/10.2139/ssrn.3302881

- Chen, Xinyun, Yunan Liu, and Guiyu Hong. "An online learning approach to dynamic pricing and capacity sizing in service systems." *arXiv preprint arXiv:2009.02911* (2020).

https://www.bilibili.com/video/BV1mK4y1E7yC?from=search&seid=16562476630513160936