Green Simulation Assisted Reinforcement Learning with Model Risk for Biomanufacturing Learning and Control

Hua Zheng[1] Wei Xie[1] M. Ben Feng[2]

[1]Department of Mechanical and Industrial Engineering
Northeastern University

[2]Department of Statistics and Actuarial Science
University of Waterloo

Green Simulation Assisted RL

In this paper, we proposed a green simulation assisted Bayesian reinforcement learning (GS-RL) to guide dynamic decision making,

- Motivation:

Green Simulation Assisted RL

In this paper, we proposed a green simulation assisted Bayesian reinforcement learning (GS-RL) to guide dynamic decision making,

- Motivation:
    - **theoretical**: model-free RL is known for its sample inefficiency; “learning from limited interaction remains a key challenge” (Schwarzer et al. 2020)

Green Simulation Assisted RL

In this paper, we proposed a green simulation assisted Bayesian reinforcement learning (GS-RL) to guide dynamic decision making,

- Motivation:
  - **theoretical**: model-free RL is known for its sample inefficiency; "learning from limited interaction remains a key challenge" (Schwarzer et al. 2020)
  - **theoretical**: "imperfect dynamics model degrades the performance of the learning algorithm" (Buckman et al. 2018)

Green Simulation Assisted RL

In this paper, we proposed a green simulation assisted Bayesian reinforcement learning (GS-RL) to guide dynamic decision making,

- Motivation:
    - **theoretical**: model-free RL is known for its sample inefficiency; "learning from limited interaction remains a key challenge" (Schwarzer et al. 2020)
    - **theoretical**: "imperfect dynamics model degrades the performance of the learning algorithm" (Buckman et al. 2018)
    - **application**: Biomanufacturing process is complex and limited in sample, and has high inherent stochastic uncertainty

Green Simulation Assisted RL

In this paper, we proposed a green simulation assisted Bayesian reinforcement learning (GS-RL) to guide dynamic decision making,

- Motivation:
    - **theoretical**: model-free RL is known for its sample inefficiency; "learning from limited interaction remains a key challenge" (Schwarzer et al. 2020)
    - **theoretical**: "imperfect dynamics model degrades the performance of the learning algorithm" (Buckman et al. 2018)
    - **application**: Biomanufacturing process is complex and limited in sample, and has high inherent stochastic uncertainty
- Benefit:

Green Simulation Assisted RL

In this paper, we proposed a green simulation assisted Bayesian reinforcement learning (GS-RL) to guide dynamic decision making,

- Motivation:
  - **theoretical**: model-free RL is known for its sample inefficiency; "learning from limited interaction remains a key challenge" (Schwarzer et al. 2020)
  - **theoretical**: "imperfect dynamics model degrades the performance of the learning algorithm" (Buckman et al. 2018)
  - **application**: Biomanufacturing process is complex and limited in sample, and has high inherent stochastic uncertainty
- Benefit:
  - improve sample efficiency by smartly reusing past experience (Green Simulation)

Green Simulation Assisted RL

In this paper, we proposed a green simulation assisted Bayesian reinforcement learning (GS-RL) to guide dynamic decision making,

- Motivation:
    - **theoretical**: model-free RL is known for its sample inefficiency; "learning from limited interaction remains a key challenge" (Schwarzer et al. 2020)
    - **theoretical**: "imperfect dynamics model degrades the performance of the learning algorithm" (Buckman et al. 2018)
    - **application**: Biomanufacturing process is complex and limited in sample, and has high inherent stochastic uncertainty
- Benefit:
    - improve sample efficiency by smartly reusing past experience (Green Simulation)
    - account for model risk (Bayesian dynamics/transition model)

Green Simulation Assisted RL

In this paper, we proposed a green simulation assisted Bayesian reinforcement learning (GS-RL) to guide dynamic decision making,

- Motivation:
  - **theoretical**: model-free RL is known for its sample inefficiency; "learning from limited interaction remains a key challenge" (Schwarzer et al. 2020)
  - **theoretical**: "imperfect dynamics model degrades the performance of the learning algorithm" (Buckman et al. 2018)
  - **application**: Biomanufacturing process is complex and limited in sample, and has high inherent stochastic uncertainty
- Benefit:
  - improve sample efficiency by smartly reusing past experience (Green Simulation)
  - account for model risk (Bayesian dynamics/transition model)
  - **faster convergence!**

Biomanufacturing Challenges

The biomanufacturing faces the critical challenges including

- Biotherapeutics are manufactured in living cells whose biological processes are complex and have highly variable outputs (green simulation);

Biomanufacturing Challenges

The biomanufacturing faces the critical challenges including

- Biotherapeutics are manufactured in living cells whose biological processes are complex and have highly variable outputs (green simulation);
- More "personalized" bioprocess requires more advanced manufacturing protocols and automation (optimal policy from reinforcement learning);

Biomanufacturing Challenges

The biomanufacturing faces the critical challenges including

- Biotherapeutics are manufactured in living cells whose biological processes are complex and have highly variable outputs (green simulation);
- More "personalized" bioprocess requires more advanced manufacturing protocols and automation (optimal policy from reinforcement learning);
- analytical testing time required by biopharmaceuticals of complex molecular structure is lengthy, and the process observations are relatively limited. (Bayesian dynamics model)

Biomanufacturing Decision Process

Formulate the bioprocesss control as sequence of decisions rules

- Map current bioprocess state (i.e. critical quality attributes (CQAs) sampled and analyzed off-line or monitored at-line) to control critical process parameters (CPPs)

Biomanufacturing Decision Process

Formulate the bioprocesss control as sequence of decisions rules

- Map current bioprocess state (i.e. critical quality attributes (CQAs) sampled and analyzed off-line or monitored at-line) to control critical process parameters (CPPs)
- Optimal policy maximizes cumulative outcomes (i.e. profit, yield, productivity and cost) if applied to biomanufacturing production.

Biomanufacturing Decision Process

Formulate the bioprocesss control as sequence of decisions rules

- Map current bioprocess state (i.e. critical quality attributes (CQAs) sampled and analyzed off-line or monitored at-line) to control critical process parameters (CPPs)
- Optimal policy maximizes cumulative outcomes (i.e. profit, yield, productivity and cost) if applied to biomanufacturing production.

Formally speaking, at each time step t with $t \in \{1, 2, \ldots, H\}$

Biomanufacturing Decision Process

Formulate the bioprocesss control as sequence of decisions rules

- Map current bioprocess state (i.e. critical quality attributes (CQAs) sampled and analyzed off-line or monitored at-line) to control critical process parameters (CPPs)
- Optimal policy maximizes cumulative outcomes (i.e. profit, yield, productivity and cost) if applied to biomanufacturing production.

Formally speaking, at each time step t with $t \in \{1, 2, \ldots, H\}$

- the bioprocess is in some state s_t, and the decision maker may choose any action a_t by following a policy $\pi_t(a_t|s_t)$.

Biomanufacturing Decision Process

Formulate the bioprocesss control as sequence of decisions rules

- Map current bioprocess state (i.e. critical quality attributes (CQAs) sampled and analyzed off-line or monitored at-line) to control critical process parameters (CPPs)
- Optimal policy maximizes cumulative outcomes (i.e. profit, yield, productivity and cost) if applied to biomanufacturing production.

Formally speaking, at each time step t with $t \in \{1, 2, \ldots, H\}$

- the bioprocess is in some state s_t, and the decision maker may choose any action a_t by following a policy $\pi_t(a_t|s_t)$.
- Then the process responds at the next time step $(t+1)$ by moving into a new state s_{t+1} following transition probability $P(s'|s,a)$, and giving the decision maker a corresponding reward or cost, denoted by $R_t(a_t, s_t)$.

Optimization

- Let $D^{\pi_\theta}_{P_{\omega^c}}(\tau) \equiv p(s_1;\omega^c)\prod_{t=1}^{H-1}\pi^t_{\theta}(a_t|s_t)p(s_{t+1}|s_t,a_t;\omega^c)$ denote the distribution of the trajectory $\tau \equiv (s_1, a_1, \ldots, s_{H-1}, a_{H-1}, s_H)$
- Given historical data $\mathcal{D}_p$, we have objective

$$\max_{\pi_\theta} \mu(\pi_\theta) = \mathbb{E}_{\omega\sim p(\omega|\mathcal{D}_p)}\left[\mathbb{E}_{\tau\sim D^{\pi_\theta}_{P_\omega}(\tau)}\left[\sum_{t=1}^{H-1}\gamma^{t-1}r_t \middle| \pi_\theta, s_1, \omega\right]\right]$$

Optimization

- Let $D^{\boldsymbol{\pi_\theta}}_{P_{\boldsymbol{\omega}^c}}(\boldsymbol{\tau}) \equiv p(s_1;\boldsymbol{\omega}^c)\prod_{t=1}^{H-1}\pi^t_{\boldsymbol{\theta}}(a_t|s_t)p(s_{t+1}|s_t,a_t;\boldsymbol{\omega}^c)$ denote the distribution of the trajectory $\boldsymbol{\tau} \equiv (s_1, a_1, \ldots, s_{H-1}, a_{H-1}, s_H)$
- Given historical data $\mathcal{D}_p$, we have objective

$$\max_{\boldsymbol{\pi_\theta}} \mu(\boldsymbol{\pi_\theta}) = \mathbb{E}_{\boldsymbol{\omega}\sim p(\boldsymbol{\omega}|\mathcal{D}_p)}\left[\mathbb{E}_{\boldsymbol{\tau}\sim D^{\boldsymbol{\pi_\theta}}_{P_{\boldsymbol{\omega}}}(\boldsymbol{\tau})}\left[\sum_{t=1}^{H-1}\gamma^{t-1}r_t \middle| \boldsymbol{\pi_\theta}, s_1, \boldsymbol{\omega}\right]\right]$$

- Under some regularity conditions, we derived its gradient

$$\nabla_\theta \mu(\boldsymbol{\pi_\theta}) = \underbrace{\mathbb{E}_{\boldsymbol{\omega}}}_{(1)}\left[\underbrace{\mathbb{E}_{\boldsymbol{\tau}\sim D^{\boldsymbol{\pi}_{\bar{\boldsymbol{\theta}}}}_{P_{\bar{\boldsymbol{\omega}}}}}}_{(2)}\left[\frac{D^{\boldsymbol{\pi_\theta}}_{P_{\boldsymbol{\omega}}}(\boldsymbol{\tau})}{D^{\boldsymbol{\pi}_{\bar{\boldsymbol{\theta}}}}_{P_{\bar{\boldsymbol{\omega}}}}(\boldsymbol{\tau})}\sum_{t=1}^{H-1}\nabla_\theta \log(\pi_{\boldsymbol{\theta}}(a_t|s_t))\sum_{t'=t}^{H-1}\gamma^{t'-1}r'_t(a^{(i,j)}_{t'}, s^{(i,j)}_{t'})\right]\right]$$

where (1) accounts for parametric uncertainty (i.e. model risk) and (2) accounts for stochastic uncertainty.

Individual/Mixture Likelihood Ratio (ILR/MLR)

At the k-th iteration, given a posterior sample $\boldsymbol{\omega}_k \sim p(\boldsymbol{\omega}|\mathcal{D}_p)$ and policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_k}$, the *likelihood ratio based policy gradient estimator*,

$$\widehat{\nabla_{\boldsymbol{\theta}}\mu}_{k,\mathbf{n}}^{ILR/MLR} = \frac{1}{k}\sum_{i=1}^{k}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[L_k(\boldsymbol{\tau}^{(i,j)})\sum_{t=1}^{H-1}\nabla_{\boldsymbol{\theta}}\log(\pi_{\boldsymbol{\theta}_k}(a_t^{(i,j)}|s_t^{(i,j)}))\sum_{t'=t}^{H-1}\gamma^{t'-1}r_t'(a_{t'}^{(i,j)},s_{t'}^{(i,j)})\right]$$

where $\boldsymbol{\tau}^{(i,j)} \overset{\text{i.i.d}}{\sim} D_{P_{\boldsymbol{\omega}_i}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_i}}(\boldsymbol{\tau})$ represent the trajectories generated in i-th iteration.

Individual/Mixture Likelihood Ratio (ILR/MLR)

At the k-th iteration, given a posterior sample $\boldsymbol{\omega}_k \sim p(\boldsymbol{\omega}|\mathcal{D}_p)$ and policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_k}$, the *likelihood ratio based policy gradient estimator*,

$$\widehat{\nabla_{\boldsymbol{\theta}}\mu}_{k,\mathbf{n}}^{ILR/MLR} = \frac{1}{k}\sum_{i=1}^{k}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[L_k(\boldsymbol{\tau}^{(i,j)})\sum_{t=1}^{H-1}\nabla_{\boldsymbol{\theta}}\log(\pi_{\boldsymbol{\theta}_k}(a_t^{(i,j)}|s_t^{(i,j)}))\sum_{t'=t}^{H-1}\gamma^{t'-1}r'_t(a_{t'}^{(i,j)},s_{t'}^{(i,j)})\right]$$

where $\boldsymbol{\tau}^{(i,j)} \overset{\text{i.i.d}}{\sim} D_{P_{\boldsymbol{\omega}_i}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_i}}(\boldsymbol{\tau})$ represent the trajectories generated in i-th iteration.

- individual likelihood ratio (IRL): $L_k(\boldsymbol{\tau}) = \frac{D_{P_{\boldsymbol{\omega}_k}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_{\mathbf{k}}}}(\boldsymbol{\tau})}{D_{P_{\boldsymbol{\omega}_i}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_i}}(\boldsymbol{\tau})}$
- mixture likelihood ratio (MLR): $L_k(\boldsymbol{\tau}) = \frac{D_{P_{\boldsymbol{\omega}_k}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_k}}(\boldsymbol{\tau})}{\sum_{i=1}^{k}\alpha_i^k D_{P_{\boldsymbol{\omega}_i}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_i}}(\boldsymbol{\tau})}$ with $\alpha_i^k = \frac{n_i}{\sum_{i=1}^{k} n_i}$.

Remarks:

Individual/Mixture Likelihood Ratio (ILR/MLR)

At the k-th iteration, given a posterior sample $\boldsymbol{\omega}_k \sim p(\boldsymbol{\omega}|\mathcal{D}_p)$ and policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_k}$, the *likelihood ratio based policy gradient estimator*,

$$\widehat{\nabla_{\boldsymbol{\theta}}\mu}_{k,\mathbf{n}}^{ILR/MLR} = \frac{1}{k}\sum_{i=1}^{k}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[L_k(\boldsymbol{\tau}^{(i,j)})\sum_{t=1}^{H-1}\nabla_{\boldsymbol{\theta}}\log(\pi_{\boldsymbol{\theta}_k}(a_t^{(i,j)}|s_t^{(i,j)}))\sum_{t'=t}^{H-1}\gamma^{t'-1}r'_t(a_{t'}^{(i,j)},s_{t'}^{(i,j)})\right]$$

where $\boldsymbol{\tau}^{(i,j)} \overset{\text{i.i.d}}{\sim} D_{P_{\boldsymbol{\omega}_i}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_i}}(\boldsymbol{\tau})$ represent the trajectories generated in i-th iteration.

- individual likelihood ratio (IRL): $L_k(\boldsymbol{\tau}) = \frac{D_{P_{\boldsymbol{\omega}_k}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_k}}(\boldsymbol{\tau})}{D_{P_{\boldsymbol{\omega}_i}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_i}}(\boldsymbol{\tau})}$
- mixture likelihood ratio (MLR): $L_k(\boldsymbol{\tau}) = \frac{D_{P_{\boldsymbol{\omega}_k}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_k}}(\boldsymbol{\tau})}{\sum_{i=1}^{k}\alpha_i^k D_{P_{\boldsymbol{\omega}_i}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_i}}(\boldsymbol{\tau})}$ with $\alpha_i^k = \frac{n_i}{\sum_{i=1}^{k} n_i}$.

Remarks:

- The likelihood ratio $L_k(\boldsymbol{\tau})$ is larger for the trajectories $\boldsymbol{\tau}$ that are more likely to be generated by the policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_k}$ and transition probabilities $P_{\boldsymbol{\omega}_k}$.

Individual/Mixture Likelihood Ratio (ILR/MLR)

At the k-th iteration, given a posterior sample $\boldsymbol{\omega}_k \sim p(\boldsymbol{\omega}|\mathcal{D}_p)$ and policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_k}$, the *likelihood ratio based policy gradient estimator*,

$$\widehat{\nabla_{\boldsymbol{\theta}}\mu}_{k,\mathbf{n}}^{ILR/MLR} = \frac{1}{k}\sum_{i=1}^{k}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[L_k(\boldsymbol{\tau}^{(i,j)})\sum_{t=1}^{H-1}\nabla_{\boldsymbol{\theta}}\log(\pi_{\boldsymbol{\theta}_k}(a_t^{(i,j)}|s_t^{(i,j)}))\sum_{t'=t}^{H-1}\gamma^{t'-1}r_t'(a_{t'}^{(i,j)}, s_{t'}^{(i,j)})\right]$$

where $\boldsymbol{\tau}^{(i,j)} \overset{\text{i.i.d}}{\sim} D_{P_{\boldsymbol{\omega}_i}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_i}}(\boldsymbol{\tau})$ represent the trajectories generated in i-th iteration.

- individual likelihood ratio (IRL): $L_k(\boldsymbol{\tau}) = \frac{D_{P_{\boldsymbol{\omega}_k}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_k}}(\boldsymbol{\tau})}{D_{P_{\boldsymbol{\omega}_i}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_i}}(\boldsymbol{\tau})}$
- mixture likelihood ratio (MLR): $L_k(\boldsymbol{\tau}) = \frac{D_{P_{\boldsymbol{\omega}_k}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_k}}(\boldsymbol{\tau})}{\sum_{i=1}^{k}\alpha_i^k D_{P_{\boldsymbol{\omega}_i}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_i}}(\boldsymbol{\tau})}$ with $\alpha_i^k = \frac{n_i}{\sum_{i=1}^{k} n_i}$.

Remarks:

- The likelihood ratio $L_k(\boldsymbol{\tau})$ is larger for the trajectories $\boldsymbol{\tau}$ that are more likely to be generated by the policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_k}$ and transition probabilities $P_{\boldsymbol{\omega}_k}$.
- ILR is unbiased however its variance could grow exponentially as the horizon H increases, which restricts their applications.

Individual/Mixture Likelihood Ratio (ILR/MLR)

At the k-th iteration, given a posterior sample $\boldsymbol{\omega}_k \sim p(\boldsymbol{\omega}|\mathcal{D}_p)$ and policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_k}$, the *likelihood ratio based policy gradient estimator*,

$$\widehat{\nabla_{\boldsymbol{\theta}}\mu}_{k,\mathbf{n}}^{ILR/MLR} = \frac{1}{k}\sum_{i=1}^{k}\frac{1}{n_i}\sum_{j=1}^{n_i}\left[L_k(\boldsymbol{\tau}^{(i,j)})\sum_{t=1}^{H-1}\nabla_{\boldsymbol{\theta}}\log(\pi_{\boldsymbol{\theta}_k}(a_t^{(i,j)}|s_t^{(i,j)}))\sum_{t'=t}^{H-1}\gamma^{t'-1}r'_t(a_{t'}^{(i,j)},s_{t'}^{(i,j)})\right]$$

where $\boldsymbol{\tau}^{(i,j)} \overset{\text{i.i.d}}{\sim} D_{P_{\boldsymbol{\omega}_i}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_i}}(\boldsymbol{\tau})$ represent the trajectories generated in i-th iteration.

- individual likelihood ratio (IRL): $L_k(\boldsymbol{\tau}) = \dfrac{D_{P_{\boldsymbol{\omega}_k}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_{\boldsymbol{k}}}}(\boldsymbol{\tau})}{D_{P_{\boldsymbol{\omega}_i}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_i}}(\boldsymbol{\tau})}$
- mixture likelihood ratio (MLR): $L_k(\boldsymbol{\tau}) = \dfrac{D_{P_{\boldsymbol{\omega}_k}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_k}}(\boldsymbol{\tau})}{\sum_{i=1}^{k}\alpha_i^k D_{P_{\boldsymbol{\omega}_i}}^{\boldsymbol{\pi}_{\boldsymbol{\theta}_i}}(\boldsymbol{\tau})}$ with $\alpha_i^k = \frac{n_i}{\sum_{i=1}^{k} n_i}$.

Remarks:

- The likelihood ratio $L_k(\boldsymbol{\tau})$ is larger for the trajectories $\boldsymbol{\tau}$ that are more likely to be generated by the policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_k}$ and transition probabilities $P_{\boldsymbol{\omega}_k}$.
- ILR is unbiased however its variance could grow exponentially as the horizon H increases, which restricts their applications.
- MLR is bounded by $\frac{1}{\alpha_i^k}$ and has lower variance than ILR.

Algorithm

```
Input: the number of periods $P$ for real-world dynamic data collection; the number of
  iterations $K$; Initialize the set of sample trajectories $\mathcal{E}_1$, the set of transition model
  parameters $\boldsymbol{\Omega}_1$, and the set of policy parameters $\boldsymbol{\Theta}_1$ to be empty set.
for $p = 1, 2, \ldots, P$ (at each new real-world data collection point) do
    for $k = (p-1)K + 1, (p-1)K + 2, \ldots, pK$ do
        1. Generate posterior samples $\boldsymbol{\omega}_k \sim p(\boldsymbol{\omega}|\mathcal{D}_p)$ and build the transition model with
          new parameter $\boldsymbol{\omega}_k$, i.e., $p(s_{t+1}|s_t, a_t, \boldsymbol{\omega}_k)$ for $t = 1, 2, \ldots, H-1$ ;
        2. Generate $n_k$ trajectories following the policy $\pi_{\boldsymbol{\theta}_k}$ and model $\boldsymbol{\omega}_k$;
        3. Calculate $\widehat{\nabla_{\boldsymbol{\theta}} \mu}^{MLR}_{k,\mathbf{n}}$ and update $\boldsymbol{\theta}_{k+1} \leftarrow \boldsymbol{\theta}_k + \eta_k \cdot \widehat{\nabla \mu}^{MLR}_{k,\mathbf{n}}$ ;
        4. Record new generated trajectories $\mathcal{E}_{k+1} = \mathcal{E}_k \cup \{\boldsymbol{\tau}^{(k,j)}|j = 1, 2, \ldots, n_k\}$,
          transition model parameters $\boldsymbol{\Omega}_{k+1} = \boldsymbol{\Omega}_k \cup \{\boldsymbol{\omega}_k\}$ and policy parameters
          $\boldsymbol{\Theta}_{k+1} = \boldsymbol{\Theta}_k \cup \{\boldsymbol{\theta}_k\}$;
    end
    5. Collect new process real-world data $\mathcal{L}_p$ and update the historical data set
     $\mathcal{D}_{p+1} = \mathcal{D}_p \cup \mathcal{L}_p$ and the posterior distribution $p(\boldsymbol{\omega}|\mathcal{D}_{p+1})$.
end
```

A Biomanufacturing Problem

In this paper, we consider a biomanufacturing process control problem and mainly focus on chromatography in the downstream (Martagan et al.[1]).

- **Decision Epoch:** Consider three-step chromatography. Observe measurements and make decisions at each decision epoch $\mathcal{T} = \{t : 1, 2, 3\}$.

A Biomanufacturing Problem

In this paper, we consider a biomanufacturing process control problem and mainly focus on chromatography in the downstream (Martagan et al.[1]).

- **Decision Epoch:** Consider three-step chromatography. Observe measurements and make decisions at each decision epoch $\mathcal{T} = \{t : 1, 2, 3\}$.
- **State Space:** The state s_t at time t, denoted by the protein-impurity-step tuple $s_t \triangleq (p_t, i_t, t)$:

A Biomanufacturing Problem

In this paper, we consider a biomanufacturing process control problem and mainly focus on chromatography in the downstream (Martagan et al.[1]).

- **Decision Epoch:** Consider three-step chromatography. Observe measurements and make decisions at each decision epoch $\mathcal{T} = \{t : 1, 2, 3\}$.
- **State Space:** The state s_t at time t, denoted by the protein-impurity-step tuple $s_t \triangleq (p_t, i_t, t)$:
  - $p_t \in \mathbf{P}$: the amount of protein at $t \in \mathcal{T}$ where $\mathbf{P} \equiv [0, \bar{P}]$

A Biomanufacturing Problem

In this paper, we consider a biomanufacturing process control problem and mainly focus on chromatography in the downstream (Martagan et al.[1]).

- **Decision Epoch:** Consider three-step chromatography. Observe measurements and make decisions at each decision epoch $\mathcal{T} = \{t : 1, 2, 3\}$.
- **State Space:** The state s_t at time t, denoted by the protein-impurity-step tuple $s_t \triangleq (p_t, i_t, t)$:
  - $p_t \in \mathbf{P}$: the amount of protein at $t \in \mathcal{T}$ where $\mathbf{P} \equiv [0, \bar{P}]$
  - $i_t \in \mathbf{I}$: the amount of impurity at $t \in \mathcal{T}$ where $\mathbf{I} \equiv [0, \bar{I}]$.

A Biomanufacturing Problem

In this paper, we consider a biomanufacturing process control problem and mainly focus on chromatography in the downstream (Martagan et al.[1]).

- **Decision Epoch:** Consider three-step chromatography. Observe measurements and make decisions at each decision epoch $\mathcal{T} = \{t : 1, 2, 3\}$.
- **State Space:** The state s_t at time t, denoted by the protein-impurity-step tuple $s_t \triangleq (p_t, i_t, t)$:
    - $p_t \in \mathbf{P}$: the amount of protein at $t \in \mathcal{T}$ where $\mathbf{P} \equiv [0, \bar{P}]$
    - $i_t \in \mathbf{I}$: the amount of impurity at $t \in \mathcal{T}$ where $\mathbf{I} \equiv [0, \bar{I}]$.
- **Action Space:** Let a_t denote the choice of pooling windows.

A Biomanufacturing Problem

In this paper, we consider a biomanufacturing process control problem and mainly focus on chromatography in the downstream (Martagan et al.[1]).

- **Decision Epoch:** Consider three-step chromatography. Observe measurements and make decisions at each decision epoch $\mathcal{T} = \{t : 1, 2, 3\}$.
- **State Space:** The state s_t at time t, denoted by the protein-impurity-step tuple $s_t \triangleq (p_t, i_t, t)$:
    - $p_t \in \mathbf{P}$: the amount of protein at $t \in \mathcal{T}$ where $\mathbf{P} \equiv [0, \bar{P}]$
    - $i_t \in \mathbf{I}$: the amount of impurity at $t \in \mathcal{T}$ where $\mathbf{I} \equiv [0, \bar{I}]$.
- **Action Space:** Let a_t denote the choice of pooling windows.
- **Reward:** Let $r_t = \frac{p_t}{p_t + i_t}$ denote the purity level. At each time step in downstream process, the reward is

$$\mathrm{r}(\mathrm{p}_t, i_t, t = 3) = \begin{cases} -c_f, & \text{if } r_t < r_d, \\ r(p_d), & \text{if } r_t \geq r_d, p_t \geq p_d, \\ r(p_t) - c_l(p_d - p_t), & \text{if } r_t \geq r_d, p_t \leq p_d. \end{cases}$$

$$\mathrm{r}(\mathrm{p}_t, i_t, t) = -\$8 \text{ with } t \in \{1, 2, 3\}$$

A Biomanufacturing Problem

- **State Transitions:** In each step of chromatography, the random proportions of protein and impurity will be removed, which depends on the selection of pooling window a_t.

A Biomanufacturing Problem

- **State Transitions:** In each step of chromatography, the random proportions of protein and impurity will be removed, which depends on the selection of pooling window a_t.
  - $i_{t+1} = (\Psi_t | a_t) i_t$ and $p_{t+1} = (H_t | a_t) p_t$,

A Biomanufacturing Problem

- **State Transitions:** In each step of chromatography, the random proportions of protein and impurity will be removed, which depends on the selection of pooling window a_t.
    - $i_{t+1} = (\Psi_t | a_t) i_t$ and $p_{t+1} = (H_t | a_t) p_t$,
        - ➤ $\Psi_t | a_t \sim \text{Beta}(\psi_t^l | a_t, \psi_t^u | a_t)$ for all $a_t \in \mathcal{A}$ and $t \in \mathcal{T}$;

A Biomanufacturing Problem

- **State Transitions:** In each step of chromatography, the random proportions of protein and impurity will be removed, which depends on the selection of pooling window a_t.
  - $i_{t+1} = (\Psi_t | a_t) i_t$ and $p_{t+1} = (H_t | a_t) p_t$,
    - ➤ $\Psi_t | a_t \sim \text{Beta}(\psi_t^l | a_t, \psi_t^u | a_t)$ for all $a_t \in \mathcal{A}$ and $t \in \mathcal{T}$;
    - ➤ $H_t | a_t \sim \text{Beta}(\eta_t^l | a_t, \eta_t^u | a_t)$ for all $a_t \in \mathcal{A}$ and $t \in \mathcal{T}$;

A Biomanufacturing Problem

- **State Transitions:** In each step of chromatography, the random proportions of protein and impurity will be removed, which depends on the selection of pooling window a_t.
    - $i_{t+1} = (\Psi_t | a_t) i_t$ and $p_{t+1} = (H_t | a_t) p_t$,
        - ➤ $\Psi_t | a_t \sim \text{Beta}(\psi_t^l | a_t, \psi_t^u | a_t)$ for all $a_t \in \mathcal{A}$ and $t \in \mathcal{T}$;
        - ➤ $H_t | a_t \sim \text{Beta}(\eta_t^l | a_t, \eta_t^u | a_t)$ for all $a_t \in \mathcal{A}$ and $t \in \mathcal{T}$;
    - uniform prior $\text{Unif}(0, 300)$ for all parameters.

A Biomanufacturing Problem

- **State Transitions:** In each step of chromatography, the random proportions of protein and impurity will be removed, which depends on the selection of pooling window a_t.
    - $i_{t+1} = (\Psi_t | a_t) i_t$ and $p_{t+1} = (H_t | a_t) p_t$,
        - ➤ $\Psi_t | a_t \sim \text{Beta}(\psi_t^l | a_t, \psi_t^u | a_t)$ for all $a_t \in \mathcal{A}$ and $t \in \mathcal{T}$;
        - ➤ $H_t | a_t \sim \text{Beta}(\eta_t^l | a_t, \eta_t^u | a_t)$ for all $a_t \in \mathcal{A}$ and $t \in \mathcal{T}$;
    - uniform prior $\text{Unif}(0, 300)$ for all parameters.
- **Policy:** We use a 2-layer perceptron (MLP) of $D = 16$ dimensional first layer and 10 dimensional output layer with softmax activation function to parameterize our policy.

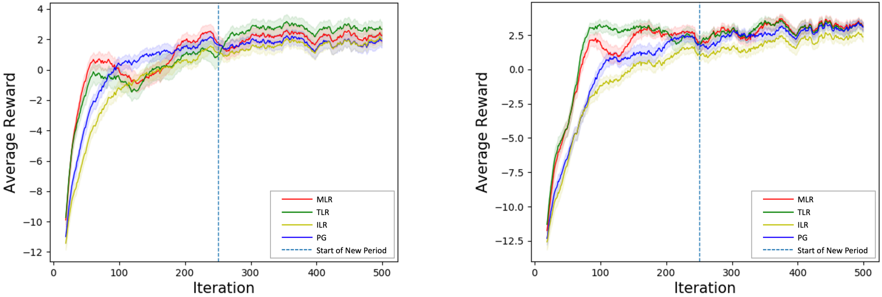
Benchmarks

We compare the performance of proposed green simulation assisted policy gradient with RL (MLR) with

- Likelihood ratio based policy gradient with mixture proposal distribution (MLR)
- Likelihood ratio based policy gradient with true transition model known (TLR)
- Individual likelihood ratio based policy gradient (ILR)
- Empirical policy gradient (PG): classical policy gradient method using the point estimator (mean) of state transition model parameter as the true one

Result: Faster Convergence

With $m = 20$ historical samples and $P = 2$ periods, $r_{test} = 200$ simulation runs and $M = 5$ macro replications, the simulation results shows faster convergence than other algorithms,



(a) $n_i = 50$

(b) $n_i = 25$

Figure: Convergence results of MLR, TLR, ILR and PG.

Conclusion

In this paper, we propose a new green simulation assisted Bayesian reinforcement learning (GS-RL) framework which

- introduces a Bayesian model-based approach into policy search as evolving transition probability;

Conclusion

In this paper, we propose a new green simulation assisted Bayesian reinforcement learning (GS-RL) framework which

- introduces a Bayesian model-based approach into policy search as evolving transition probability;
    - incorporate transition model parametric uncertainty

Conclusion

In this paper, we propose a new green simulation assisted Bayesian reinforcement learning (GS-RL) framework which

- introduces a Bayesian model-based approach into policy search as evolving transition probability;
  - incorporate transition model parametric uncertainty
  - ease the "small sample" challenge for biomanufacturing by incorporating prior knowledge into the control.

Conclusion

In this paper, we propose a new green simulation assisted Bayesian reinforcement learning (GS-RL) framework which

- introduces a Bayesian model-based approach into policy search as evolving transition probability;
    - incorporate transition model parametric uncertainty
    - ease the "small sample" challenge for biomanufacturing by incorporating prior knowledge into the control.
- mixture likelihood ratio provides a new scheme for "experience reply" method for reinforcement learning;

Conclusion

In this paper, we propose a new green simulation assisted Bayesian reinforcement learning (GS-RL) framework which

- introduces a Bayesian model-based approach into policy search as evolving transition probability;
    - incorporate transition model parametric uncertainty
    - ease the "small sample" challenge for biomanufacturing by incorporating prior knowledge into the control.
- mixture likelihood ratio provides a new scheme for "experience reply" method for reinforcement learning;
    - improve sample efficiency: reuse and weight the trajectories depending on its relative importance to current decision processes.

Conclusion

In this paper, we propose a new green simulation assisted Bayesian reinforcement learning (GS-RL) framework which

- introduces a Bayesian model-based approach into policy search as evolving transition probability;
  - incorporate transition model parametric uncertainty
  - ease the "small sample" challenge for biomanufacturing by incorporating prior knowledge into the control.
- mixture likelihood ratio provides a new scheme for "experience reply" method for reinforcement learning;
  - improve sample efficiency: reuse and weight the trajectories depending on its relative importance to current decision processes.
  - reduce policy gradient variance.

[1] Tugce Martagan, Ananth Krishnamurthy, Peter A. Leland, and Christos T. Maravelias. Performance guarantees and optimal purification decisions for engineered proteins. *Operations Research*, 66(1):18–41, January 2018.