Achieving sub-linear regret in infinite horizon average reward constrained MDP with Linear Function Approximation

(Joint work with
Xingyu Zhou, Wayne State University,
Ness Shroff, The Ohio State University)

Arnob Ghosh,
The Ohio State University, Dept. of Electrical and Computer Engineering,
Research Scientist at the NSF AI-Edge Institute

Constrained MDP

Constrained MDP

- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).

    - Provably-efficient algorithm for episodic case in linear CMDP [Ghosh et al'22] (reward r and utility g)

Constrained MDP

- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).
    - Provably-efficient algorithm for episodic case in linear CMDP [Ghosh et al'22] (reward r and utility g)
- In many applications, agents need to take action continuously (e.g., safe controller, routing decisions): Infinite-horizon model is preferable.

Average reward: $J_r^{\pi}(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_{\pi}\left(\sum_{t=1}^{T} r(x_t, a_t) \,|\, x_1 = x\right)$

Average utility: $J_g^{\pi}(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_{\pi}\left(\sum_{t} g(x_t, a_t) \,|\, x_1 = x\right),$

Constrained MDP

- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).
  - Provably-efficient algorithm for episodic case in linear CMDP [Ghosh et al'22] (reward r and utility g)
- In many applications, agents need to take action continuously (e.g., safe controller, routing decisions): Infinite-horizon model is preferable.

Average reward: $J_r^{\pi}(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_{\pi}\left(\sum_{t=1}^{T} r(x_t, a_t) \,|\, x_1 = x\right)$

Average utility: $J_g^{\pi}(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_{\pi}\left(\sum_{t} g(x_t, a_t) \,|\, x_1 = x\right),$

maximize $J_r^{\pi}(x)$ **subject to** $J_g^{\pi}(x) \geq b$

Constrained MDP

- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).
  - Provably-efficient algorithm for episodic case in linear CMDP [Ghosh et al'22] (reward r and utility g)
- In many applications, agents need to take action continuously (e.g., safe controller, routing decisions): Infinite-horizon model is preferable.

Average reward: $J_r^{\pi}(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_{\pi}\left(\sum_{t=1}^{T} r(x_t, a_t) \,|\, x_1 = x\right)$

Average utility: $J_g^{\pi}(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_{\pi}\left(\sum_{t} g(x_t, a_t) \,|\, x_1 = x\right),$

maximize $J_r^{\pi}(x)$ **subject to** $J_g^{\pi}(x) \geq b$

- Theoretical results: only for tabular case and model-based [Chen et al.'22]

Constrained MDP

- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).
  - Provably-efficient algorithm for episodic case in linear CMDP [Ghosh et al'22] (reward r and utility g)
- In many applications, agents need to take action continuously (e.g., safe controller, routing decisions): Infinite-horizon model is preferable.

Average reward: $J_r^\pi(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_\pi\left(\sum_{t=1}^{T} r(x_t, a_t) \,|\, x_1 = x\right)$

Average utility: $J_g^\pi(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_\pi\left(\sum_{t} g(x_t, a_t) \,|\, x_1 = x\right),$

maximize $J_r^\pi(x)$ **subject to** $J_g^\pi(x) \geq b$

- Theoretical results: only for tabular case and model-based [Chen et al.'22]

Constrained MDP

- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).
    - Provably-efficient algorithm for episodic case in linear CMDP [Ghosh et al'22] (reward r and utility g)
- In many applications, agents need to take action continuously (e.g., safe controller, routing decisions): Infinite-horizon model is preferable.

Average reward: $J_r^\pi(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_\pi\left(\sum_{t=1}^{T} r(x_t, a_t) \,|\, x_1 = x\right)$

Average utility: $J_g^\pi(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_\pi\left(\sum_{t} g(x_t, a_t) \,|\, x_1 = x\right),$

maximize $J_r^\pi(x)$ **subject to** $J_g^\pi(x) \geq b$

- Theoretical results: only for tabular case and model-based [Chen et al.'22]

Can we achieve sub-linear regret and constraint violation for infinite-horizon linear CMDP using model-free RL?

Constrained MDP

- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).
  - Provably-efficient algorithm for episodic case in linear CMDP [Ghosh et al'22] (reward r and utility g)
- In many applications, agents need to take action continuously (e.g., safe controller, routing decisions): Infinite-horizon model is preferable.

Average reward: $$J_r^\pi(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_\pi\left(\sum_{t=1}^{T} r(x_t, a_t) \,|\, x_1 = x\right)$$

Average utility: $$J_g^\pi(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_\pi\left(\sum_{t} g(x_t, a_t) \,|\, x_1 = x\right),$$

maximize $J_r^\pi(x)$ **subject to** $J_g^\pi(x) \geq b$

- Theoretical results: only for tabular case and model-based [Chen et al.'22]

Can we achieve sub-linear regret and constraint violation for infinite-horizon linear CMDP using model-free RL?

Constrained MDP

- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).
  - Provably-efficient algorithm for episodic case in linear CMDP [Ghosh et al'22] (reward r and utility g)
- In many applications, agents need to take action continuously (e.g., safe controller, routing decisions): Infinite-horizon model is preferable.

Average reward: $J_r^{\pi}(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_{\pi}\left(\sum_{t=1}^{T} r(x_t, a_t) \,|\, x_1 = x\right)$

Average utility: $J_g^{\pi}(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_{\pi}\left(\sum_{t} g(x_t, a_t) \,|\, x_1 = x\right)$,

maximize $J_r^{\pi}(x)$ **subject to** $J_g^{\pi}(x) \geq b$

- Theoretical results: only for tabular case and model-based [Chen et al.'22]

Can we achieve sub-linear regret and constraint violation for infinite-horizon linear CMDP using model-free RL?

Constrained MDP

- In many practical applications, agent needs to satisfy constraints (e.g., safe navigation by robots, takes decision under limited resource).
  - Provably-efficient algorithm for episodic case in linear CMDP [Ghosh et al'22] (reward r and utility g)
- In many applications, agents need to take action continuously (e.g., safe controller, routing decisions): Infinite-horizon model is preferable.

Average reward: $J_r^{\pi}(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_{\pi}\left(\sum_{t=1}^{T} r(x_t, a_t) \,|\, x_1 = x\right)$

Average utility: $J_g^{\pi}(x) = \lim_{T->\infty} \frac{1}{T}\mathbb{E}_{\pi}\left(\sum_{t} g(x_t, a_t) \,|\, x_1 = x\right),$

maximize $J_r^{\pi}(x)$ **subject to** $J_g^{\pi}(x) \geq b$

- Theoretical results: only for tabular case and model-based [Chen et al.'22]

Linear in feature $\phi(x, a)$

Can we achieve sub-linear regret and constraint violation for infinite-horizon linear CMDP using model-free RL?

Algorithm

Algorithm

- **Basic Assumption:** $J_r^*(x) = J_r^*, J_g^*(x) = J_g^* \forall x$ (satisfied by weakly communicating MDP for tabular case)

Algorithm

- **Basic Assumption:** $J_r^*(x) = J_r^*, J_g^*(x) = J_g^* \forall x$ (satisfied by weakly communicating MDP for tabular case)
- Seek to reduce $\text{Regret}(T) = \sum_t (J_r^* - r(x_t, a_t)), \text{Violation}(T) = \sum_t (b - g(x_t, a_t))$.

Algorithm

- **Basic Assumption:** $J_r^*(x) = J_r^*, J_g^*(x) = J_g^* \forall x$ (satisfied by weakly communicating MDP for tabular case)
- Seek to reduce $\text{Regret}(T) = \sum_t (J_r^* - r(x_t, a_t)), \text{Violation}(T) = \sum_t (b - g(x_t, a_t))$.
- First try: Open the loop

Algorithm

- **Basic Assumption:** $J_r^*(x) = J_r^*, J_g^*(x) = J_g^* \forall x$ (satisfied by weakly communicating MDP for tabular case)
- Seek to reduce $\text{Regret}(T) = \sum_t (J_r^* - r(x_t, a_t)), \text{Violation}(T) = \sum_t (b - g(x_t, a_t))$.
- First try: Open the loop
  - Divide T in K episodes (episode length: $H = T/K$) —> employ algorithm for episodic case from Ghosh et al.'22.

Algorithm

- **Basic Assumption:** $J_r^*(x) = J_r^*, J_g^*(x) = J_g^* \forall x$ (satisfied by weakly communicating MDP for tabular case)
- Seek to reduce $\text{Regret}(T) = \sum_t (J_r^* - r(x_t, a_t)), \text{Violation}(T) = \sum_t (b - g(x_t, a_t))$.
- First try: Open the loop
  - Divide T in K episodes (episode length: $H = T/K$) —> employ algorithm for episodic case from Ghosh et al.'22.
- **Challenge**: The optimal solution of the original problem must be feasible to the episodic case.

Algorithm

- **Basic Assumption:** $J_r^*(x) = J_r^*, J_g^*(x) = J_g^* \forall x$ (satisfied by weakly communicating MDP for tabular case)
- Seek to reduce $\text{Regret}(T) = \sum_t (J_r^* - r(x_t, a_t)), \text{Violation}(T) = \sum_t (b - g(x_t, a_t))$.
- First try: Open the loop
    - Divide T in K episodes (episode length: $H = T/K$) —> employ algorithm for episodic case from Ghosh et al.'22.
- **Challenge**: The optimal solution of the original problem must be feasible to the episodic case.
    - **Solution:** Relax the constraint Hb to $Hb - \kappa$

Algorithm

- **Basic Assumption:** $J_r^*(x) = J_r^*, J_g^*(x) = J_g^* \forall x$ (satisfied by weakly communicating MDP for tabular case)
- Seek to reduce $\text{Regret}(T) = \sum_t (J_r^* - r(x_t, a_t)), \text{Violation}(T) = \sum_t (b - g(x_t, a_t))$.
- First try: Open the loop
  - Divide T in K episodes (episode length: $H = T/K$) —> employ algorithm for episodic case from Ghosh et al.'22.
- **Challenge**: The optimal solution of the original problem must be feasible to the episodic case.
  - **Solution:** Relax the constraint Hb to $Hb - \kappa$
- Both Regret and violation bound $(T/H)O(1)$+ $\tilde{\mathcal{O}}(\sqrt{d^3H^2T})$

Algorithm

- **Basic Assumption:** $J_r^*(x) = J_r^*, J_g^*(x) = J_g^* \forall x$ (satisfied by weakly communicating MDP for tabular case)
- Seek to reduce $\text{Regret}(T) = \sum_t (J_r^* - r(x_t, a_t)), \text{Violation}(T) = \sum_t (b - g(x_t, a_t))$.
- First try: Open the loop
  - Divide T in K episodes (episode length: $H = T/K$) —> employ algorithm for episodic case from Ghosh et al.'22.
- **Challenge**: The optimal solution of the original problem must be feasible to the episodic case.
  - **Solution:** Relax the constraint Hb to $Hb - \kappa$
- Both Regret and violation bound $(T/H)O(1)$+ $\tilde{\mathcal{O}}(\sqrt{d^3H^2T})$

Too Small H —> episodic case would not resemble infinite-horizon
Too Large H —> no effect of breaking in episodes

Algorithm

- **Basic Assumption:** $J_r^*(x) = J_r^*, J_g^*(x) = J_g^* \forall x$ (satisfied by weakly communicating MDP for tabular case)
- Seek to reduce $\text{Regret}(T) = \sum_t (J_r^* - r(x_t, a_t)), \text{Violation}(T) = \sum_t (b - g(x_t, a_t))$.
- First try: Open the loop
  - Divide T in K episodes (episode length: $H = T/K$) —> employ algorithm for episodic case from Ghosh et al.'22.
- **Challenge**: The optimal solution of the original problem must be feasible to the episodic case.
  - **Solution:** Relax the constraint Hb to $Hb - \kappa$
- Both Regret and violation bound $(T/H)O(1)$+ $\tilde{\mathcal{O}}(\sqrt{d^3H^2T})$

Algorithm

- **Basic Assumption:** $J_r^*(x) = J_r^*, J_g^*(x) = J_g^* \forall x$ (satisfied by weakly communicating MDP for tabular case)
- Seek to reduce $\text{Regret}(T) = \sum_t (J_r^* - r(x_t, a_t)), \text{Violation}(T) = \sum_t (b - g(x_t, a_t))$.
- First try: Open the loop
  - Divide T in K episodes (episode length: $H = T/K$) —> employ algorithm for episodic case from Ghosh et al.'22.
- **Challenge**: The optimal solution of the original problem must be feasible to the episodic case.
  - **Solution:** Relax the constraint Hb to $Hb - \kappa$
- Both Regret and violation bound $(T/H)O(1)$+ $\tilde{\mathcal{O}}(\sqrt{d^3H^2T})$
  - $H = O(d^{-3/4}T^{1/4})$

Algorithm

- **Basic Assumption:** $J_r^*(x) = J_r^*, J_g^*(x) = J_g^* \, \forall x$ (satisfied by weakly communicating MDP for tabular case)
- Seek to reduce $\text{Regret}(T) = \sum_t (J_r^* - r(x_t, a_t)), \text{Violation}(T) = \sum_t (b - g(x_t, a_t))$.
- First try: Open the loop
  - Divide T in K episodes (episode length: $H = T/K$) —> employ algorithm for episodic case from Ghosh et al.'22.
- Challenge: The optimal solution of the original problem must be feasible to the episodic case.
  - **Solution:** Relax the constraint Hb to $Hb - \kappa$
- Both Regret and violation bound $(T/H)O(1)$+ $\tilde{\mathcal{O}}(\sqrt{d^3H^2T})$
  - $H = O(d^{-3/4}T^{1/4})$
- **Final result:** with high prob. Regret and violation bound $\tilde{\mathcal{O}}(d^{3/4}T^{3/4})$

Algorithm 2

- Can we do it better?
  - Fit the Q to Bellman equation (solve regularized least square) for $\diamond = r, g$
    $$\sum (\diamond(x_k, a_k) - J^*_\diamond + \phi(x,a)^T w_\diamond - v_\diamond(x_{k+1})) + \lambda ||w_r||_2^2$$
    - **Challenge:** Do not know J_r, J_g; $v_\diamond$ depends on $\pi, w_\diamond$.
  - Naive oracle-based algorithm with optimism (computationally inefficient) provides the parameter $w_\diamond$

$$\max_{\pi, w_r, w_g, J_r, J_g, b_r, b_g} J_r$$

subject to $||w_\diamond|| \leq C, ||b_{\diamond,k}||_{\Lambda_t^{-1}} \leq \beta, J_g \geq b$

$$w_\diamond = \Lambda_k^{-1}\left(\sum_{t=1}^{k-1} \diamond(x_k, a_k) - J_\diamond + v_\diamond(x_{t+1}) + b_{\diamond,k}\right)$$

Algorithm 2

- Can we do it better?
  - Fit the Q to Bellman equation (solve regularized least square) for $\diamond = r, g$
    $$\sum(\diamond(x_k, a_k) - J_\diamond^* + \phi(x,a)^T w_\diamond - v_\diamond(x_{k+1})) + \lambda ||w_r||_2^2$$
    - **Challenge:** Do not know J_r, J_g; $v_\diamond$ depends on π, $w_\diamond$.
  - Naive oracle-based algorithm with optimism (computationally inefficient) provides the parameter $w_\diamond$

$$\max_{\pi, w_r, w_g, J_r, J_g, b_r, b_g} J_r$$

subject to $||w_\diamond|| \leq C, ||b_{\diamond,k}||_{\Lambda_t^{-1}} \leq \beta, J_g \geq b$

$$w_\diamond = \Lambda_k^{-1}(\sum_{t=1}^{k-1} \diamond(x_k, a_k) - J_\diamond + v_\diamond(x_{t+1}) + b_{\diamond,k})$$

Algorithm 2

- Can we do it better?
  - Fit the Q to Bellman equation (solve regularized least square) for $\diamond = r, g$
    $\sum(\diamond(x_k, a_k) - J_\diamond^* + \phi(x,a)^T w_\diamond - v_\diamond(x_{k+1})) + \lambda ||w_r||_2^2$
    - **Challenge:** Do not know J_r, J_g; $v_\diamond$ depends on π, $w_\diamond$.
  - Naive oracle-based algorithm with optimism (computationally inefficient) provides the parameter $w_\diamond$

$$\max_{\pi, w_r, w_g, J_r, J_g, b_r, b_g} J_r$$

subject to $||w_\diamond|| \leq C, ||b_{\diamond,k}||_{\Lambda_t^{-1}} \leq \beta, J_g \geq b$

$$w_\diamond = \Lambda_k^{-1}\left(\sum_{t=1}^{k-1} \diamond(x_k, a_k) - J_\diamond + v_\diamond(x_{t+1}) + b_{\diamond,k}\right)$$

Algorithm 2

- Can we do it better?
  - Fit the Q to Bellman equation (solve regularized least square) for $\diamond = r, g$

    $$\sum (\diamond (x_k, a_k) - J^*_\diamond + \phi(x,a)^T w_\diamond - v_\diamond(x_{k+1})) + \lambda ||w_r||_2^2$$

    - **Challenge:** Do not know J_r, J_g; $v_\diamond$ depends on π, $w_\diamond$.
  - Naive oracle-based algorithm with optimism (computationally inefficient) provides the parameter $w_\diamond$

$$\max_{\pi, w_r, w_g, J_r, J_g, b_r, b_g} J_r$$

subject to $||w_\diamond|| \le C, ||b_{\diamond,k}||_{\Lambda_t^{-1}} \le \beta, J_g \ge b$

$$w_\diamond = \Lambda_k^{-1}(\sum_{t=1}^{k-1} \diamond (x_k, a_k) - J_\diamond + v_\diamond(x_{t+1}) + b_{\diamond,k})$$

Bonus term

Algorithm 2

- Can we do it better?
  - Fit the Q to Bellman equation (solve regularized least square) for $\diamond = r, g$
    $$\sum (\diamond(x_k, a_k) - J^*_\diamond + \phi(x,a)^T w_\diamond - v_\diamond(x_{k+1})) + \lambda ||w_r||_2^2$$
    - **Challenge:** Do not know J_r, J_g; $v_\diamond$ depends on π, $w_\diamond$.
  - Naive oracle-based algorithm with optimism (computationally inefficient) provides the parameter $w_\diamond$

$$\max_{\pi, w_r, w_g, J_r, J_g, b_r, b_g} J_r$$

subject to $||w_\diamond|| \leq C, ||b_{\diamond,k}||_{\Lambda_t^{-1}} \leq \beta, J_g \geq b$

$$w_\diamond = \Lambda_k^{-1}(\sum_{t=1}^{k-1} \diamond(x_k, a_k) - J_\diamond + v_\diamond(x_{t+1}) + b_{\diamond,k})$$

Bonus term

Algorithm 2

- Can we do it better?
  - Fit the Q to Bellman equation (solve regularized least square) for $\diamond = r, g$
    $$\sum(\diamond(x_k, a_k) - J^*_\diamond + \phi(x,a)^T w_\diamond - v_\diamond(x_{k+1})) + \lambda ||w_r||_2^2$$
    - **Challenge:** Do not know J_r, J_g; $v_\diamond$ depends on π, $w_\diamond$.
  - Naive oracle-based algorithm with optimism (computationally inefficient) provides the parameter $w_\diamond$

$$\max_{\pi, w_r, w_g, J_r, J_g, b_r, b_g} J_r$$

subject to $||w_\diamond|| \le C, ||b_{\diamond,k}||_{\Lambda_t^{-1}} \le \beta, J_g \ge b$

$$w_\diamond = \Lambda_k^{-1}\left(\sum_{t=1}^{k-1} \diamond(x_k, a_k) - J_\diamond + v_\diamond(x_{t+1}) + b_{\diamond,k}\right)$$

Bonus term

Algorithm 2

- Can we do it better?
  - Fit the Q to Bellman equation (solve regularized least square) for $\diamond = r, g$
    $\sum (\diamond(x_k, a_k) - J_\diamond^* + \phi(x,a)^T w_\diamond - v_\diamond(x_{k+1})) + \lambda ||w_r||_2^2$
    - **Challenge:** Do not know J_r, J_g; $v_\diamond$ depends on π, $w_\diamond$.
  - Naive oracle-based algorithm with optimism (computationally inefficient) provides the parameter $w_\diamond$

$$\max_{\pi, w_r, w_g, J_r, J_g, b_r, b_g} J_r$$

subject to $||w_\diamond|| \le C, ||b_{\diamond,k}||_{\Lambda_t^{-1}} \le \beta, J_g \ge b$

$$w_\diamond = \Lambda_k^{-1}\left(\sum_{t=1}^{k-1} \diamond(x_k, a_k) - J_\diamond + v_\diamond(x_{t+1}) + b_{\diamond,k}\right)$$

Algorithm 2

- Can we do it better?
  - Fit the Q to Bellman equation (solve regularized least square) for $\diamond = r, g$
    $\sum (\diamond (x_k, a_k) - J^*_\diamond + \phi(x,a)^T w_\diamond - v_\diamond(x_{k+1})) + \lambda ||w_r||_2^2$
    - **Challenge:** Do not know J_r, J_g; $v_\diamond$ depends on $\pi, w_\diamond$.
  - Naive oracle-based algorithm with optimism (computationally inefficient) provides the parameter $w_\diamond$

$$\max_{\pi, w_r, w_g, J_r, J_g, b_r, b_g} J_r$$

$$\text{subject to } ||w_\diamond|| \leq C, ||b_{\diamond,k}||_{\Lambda_t^{-1}} \leq \beta, J_g \geq b$$

$$w_\diamond = \Lambda_k^{-1}(\sum_{t=1}^{k-1} \diamond (x_k, a_k) - J_\diamond + v_\diamond(x_{t+1}) + b_{\diamond,k})$$

  - **Is it done? Not yet:** need smoothness in policy since one needs to show uniform concentration bound for both reward and utility for model-free algorithms;

Algorithm 2

- Can we do it better?
  - Fit the Q to Bellman equation (solve regularized least square) for $\diamond = r, g$
    $\sum (\diamond (x_k, a_k) - J^*_\diamond + \phi(x,a)^T w_\diamond - v_\diamond(x_{k+1})) + \lambda ||w_r||_2^2$
    - **Challenge:** Do not know J_r, J_g; $v_\diamond$ depends on $\pi, w_\diamond$.
  - Naive oracle-based algorithm with optimism (computationally inefficient) provides the parameter $w_\diamond$

$$\max_{\pi, w_r, w_g, J_r, J_g, b_r, b_g} J_r$$

subject to $||w_\diamond|| \le C, ||b_{\diamond,k}||_{\Lambda_t^{-1}} \le \beta, J_g \ge b$

$$w_\diamond = \Lambda_k^{-1}\left(\sum_{t=1}^{k-1} \diamond (x_k, a_k) - J_\diamond + v_\diamond(x_{t+1}) + b_{\diamond,k}\right)$$

  - **Is it done? Not yet:** need smoothness in policy since one needs to show uniform concentration bound for both reward and utility for model-free algorithms;
    - $\pi \in \Pi$: class of smooth policies (such as soft-max)

Algorithm 2

- Can we do it better?
  - Fit the Q to Bellman equation (solve regularized least square) for $\diamond = r, g$
    $$\sum (\diamond (x_k, a_k) - J^*_\diamond + \phi(x,a)^T w_\diamond - v_\diamond(x_{k+1})) + \lambda ||w_r||_2^2$$
    - **Challenge:** Do not know J_r, J_g; $v_\diamond$ depends on $\pi, w_\diamond$.
  - Naive oracle-based algorithm with optimism (computationally inefficient) provides the parameter $w_\diamond$

$$\max_{\pi, w_r, w_g, J_r, J_g, b_r, b_g} J_r$$
subject to $||w_\diamond|| \leq C, ||b_{\diamond,k}||_{\Lambda_t^{-1}} \leq \beta, J_g \geq b$
$$w_\diamond = \Lambda_k^{-1}\left(\sum_{t=1}^{k-1} \diamond (x_k, a_k) - J_\diamond + v_\diamond(x_{t+1}) + b_{\diamond,k}\right)$$

  - **Is it done? Not yet:** need smoothness in policy since one needs to show uniform concentration bound for both reward and utility for model-free algorithms;
    - $\pi \in \Pi$: class of smooth policies (such as soft-max)
  - **Regret and violation bound:** $\tilde{\mathcal{O}}(\sqrt{d^3 T})$ first such result for Linear CMDP.

Algorithm 3

- **Computationally efficient algorithm yet $\tilde{\mathcal{O}}(\sqrt{T})$ regret and violation (first such result for linear CMDP).**
- Additional assumptions:

Finite mixing time— $||\mathbb{P}^{\pi}\nu_1 - \mathbb{P}^{\pi}\nu_2||_{TV} \leq e^{-1/t_{mix}}||\nu_1 - \nu_2||_{TV}$; ν_1, ν_2 any state occupancy measure.

Every policy is exploratory in the feature space (can be relaxed to only one known exploratory policy)—

$$\lambda_{min}\left(\int_{\mathcal{X}} \sum_{a} \pi(a|x)\phi(x,a)\phi(x,a)^T d\nu^{\pi}(x)dx\right) \geq \sigma$$

Algorithm 3

- **Computationally efficient algorithm yet $\tilde{\mathcal{O}}(\sqrt{T})$ regret and violation (first such result for linear CMDP).**
- Additional assumptions:

Finite mixing time— $||\mathbb{P}^\pi \nu_1 - \mathbb{P}^\pi \nu_2||_{TV} \leq e^{-1/t_{mix}} ||\nu_1 - \nu_2||_{TV}$; ν_1, ν_2 any state occupancy measure.

Every policy is exploratory in the feature space (can be relaxed to only one known exploratory policy)—

$$\lambda_{min}\left(\int_{\mathcal{X}} \sum_a \pi(a|x)\phi(x,a)\phi(x,a)^T d\nu^\pi(x)dx\right) \geq \sigma$$

Algorithm 3

- **Computationally efficient algorithm yet $\tilde{\mathcal{O}}(\sqrt{T})$ regret and violation (first such result for linear CMDP).**
- Additional assumptions:

Finite mixing time— $||\mathbb{P}^{\pi}\nu_1 - \mathbb{P}^{\pi}\nu_2||_{TV} \leq e^{-1/t_{mix}}||\nu_1 - \nu_2||_{TV}$; ν_1, ν_2 any state occupancy measure.

Every policy is exploratory in the feature space (can be relaxed to only one known exploratory policy)—

$$\lambda_{min}\left(\int_{\mathcal{X}} \sum_{a} \pi(a|x)\phi(x,a)\phi(x,a)^T d\nu^{\pi}(x)dx\right) \geq \sigma$$

Algorithm 3

- **Computationally efficient algorithm yet $\tilde{\mathcal{O}}(\sqrt{T})$ regret and violation (first such result for linear CMDP).**
- Additional assumptions:

Finite mixing time— $||\mathbb{P}^{\pi}\nu_1 - \mathbb{P}^{\pi}\nu_2||_{TV} \leq e^{-1/t_{mix}}||\nu_1 - \nu_2||_{TV}$; ν_1, ν_2 any state occupancy measure.

Every policy is exploratory in the feature space (can be relaxed to only one known exploratory policy)—

$$\lambda_{min}\left(\int_{\mathcal{X}} \sum_a \pi(a|x)\phi(x,a)\phi(x,a)^T d\nu^{\pi}(x)dx\right) \geq \sigma$$

Algorithm 3

- **Computationally efficient algorithm yet $\tilde{\mathcal{O}}(\sqrt{T})$ regret and violation (first such result for linear CMDP).**
- Additional assumptions:

Finite mixing time— $||\mathbb{P}^\pi \nu_1 - \mathbb{P}^\pi \nu_2||_{TV} \leq e^{-1/t_{mix}} ||\nu_1 - \nu_2||_{TV}$; ν_1, ν_2 any state occupancy measure.

Every policy is exploratory in the feature space (can be relaxed to only one known exploratory policy)—

$$\lambda_{min}\left(\int_{\mathcal{X}} \sum_a \pi(a|x)\phi(x,a)\phi(x,a)^T d\nu^\pi(x)dx\right) \geq \sigma$$

- **Primal-dual adaptation** of MDP-EXP2 [Wei et al.'21]

Algorithm 3

- **Computationally efficient algorithm yet $\tilde{\mathcal{O}}(\sqrt{T})$ regret and violation (first such result for linear CMDP).**
- Additional assumptions:

Finite mixing time— $||\mathbb{P}^\pi \nu_1 - \mathbb{P}^\pi \nu_2||_{TV} \leq e^{-1/t_{mix}} ||\nu_1 - \nu_2||_{TV}$; ν_1, ν_2 any state occupancy measure.

Every policy is exploratory in the feature space (can be relaxed to only one known exploratory policy)—

$$\lambda_{min}\left(\int_{\mathcal{X}} \sum_a \pi(a|x)\phi(x,a)\phi(x,a)^T d\nu^\pi(x)dx\right) \geq \sigma$$

- **Primal-dual adaptation** of MDP-EXP2 [Wei et al.'21]
  - $\pi_k(a|x) \propto \pi_{k-1} \exp(\phi(x,a)^T(w_{r,k} + Y_k w_g^k))$ at epoch k.

Algorithm 3

- **Computationally efficient algorithm yet $\tilde{\mathcal{O}}(\sqrt{T})$ regret and violation (first such result for linear CMDP).**
- Additional assumptions:

Finite mixing time— $||\mathbb{P}^{\pi}\nu_1 - \mathbb{P}^{\pi}\nu_2||_{TV} \leq e^{-1/t_{mix}}||\nu_1 - \nu_2||_{TV}$; ν_1, ν_2 any state occupancy measure.
Every policy is exploratory in the feature space (can be relaxed to only one known exploratory policy)—

$$\lambda_{min}\left(\int_{\mathcal{X}} \sum_a \pi(a\,|\,x)\phi(x,a)\phi(x,a)^T d\nu^{\pi}(x)dx\right) \geq \sigma$$

- **Primal-dual adaptation** of MDP-EXP2 [Wei et al.'21]
  - $\pi_k(a\,|\,x) \propto \pi_{k-1}\exp(\phi(x,a)^T(w_{r,k} + Y_k w_g^k))$ at epoch k.
    - Divide T in $B = O((\log T)^2 t_{mix}/\sigma$ epochs, every epoch is divided in $2N = O(t_{mix}\log T)$ periods

Algorithm 3

- **Computationally efficient algorithm yet $\tilde{\mathcal{O}}(\sqrt{T})$ regret and violation (first such result for linear CMDP).**
- Additional assumptions:

Finite mixing time— $||\mathbb{P}^\pi \nu_1 - \mathbb{P}^\pi \nu_2||_{TV} \leq e^{-1/t_{mix}} ||\nu_1 - \nu_2||_{TV}$; ν_1, ν_2 any state occupancy measure.

Every policy is exploratory in the feature space (can be relaxed to only one known exploratory policy)—

$$\lambda_{min}\left(\int_{\mathcal{X}} \sum_a \pi(a|x)\phi(x,a)\phi(x,a)^T d\nu^\pi(x)dx\right) \geq \sigma$$

- **Primal-dual adaptation** of MDP-EXP2 [Wei et al.'21]
  - $\pi_k(a|x) \propto \pi_{k-1}\exp(\phi(x,a)^T(w_{r,k} + Y_k w_g^k))$ at epoch k.
    - Divide T in $B = O((\log T)^2 t_{mix}/\sigma$ epochs, every epoch is divided in $2N = O(t_{mix}\log T)$ periods
      - In each epoch, collect rewards (or, utilities) for the last N time-steps (will allow to reach the steady-state distribution in first N time-steps); also allow achieve *unbiasedness* (almost) in estimating q function.

Algorithm 3

- **Computationally efficient algorithm yet $\tilde{\mathcal{O}}(\sqrt{T})$ regret and violation (first such result for linear CMDP).**
- Additional assumptions:

Finite mixing time— $||\mathbb{P}^{\pi}\nu_1 - \mathbb{P}^{\pi}\nu_2||_{TV} \leq e^{-1/t_{mix}}||\nu_1 - \nu_2||_{TV}$; ν_1, ν_2 any state occupancy measure.
Every policy is exploratory in the feature space (can be relaxed to only one known exploratory policy)—

$$\lambda_{min}\left(\int_{\mathcal{X}} \sum_a \pi(a|x)\phi(x,a)\phi(x,a)^T d\nu^{\pi}(x)dx\right) \geq \sigma$$

- **Primal-dual adaptation** of MDP-EXP2 [Wei et al.'21]
    - $\pi_k(a|x) \propto \pi_{k-1}\exp(\phi(x,a)^T(w_{r,k} + Y_k w_g^k))$ at epoch k.
        - Divide T in $B = O((\log T)^2 t_{mix}/\sigma$ epochs, every epoch is divided in $2N = O(t_{mix}\log T)$ periods
            - In each epoch, collect rewards (or, utilities) for the last N time-steps (will allow to reach the steady-state distribution in first N time-steps); also allow achieve *unbiasedness* (almost) in estimating q function.
            - Fit $w_{j,k}$ to the collected reward (or, utility) by solving linear regression.

Algorithm 3

- **Computationally efficient algorithm yet $\tilde{\mathcal{O}}(\sqrt{T})$ regret and violation (first such result for linear CMDP).**
- Additional assumptions:

Finite mixing time— $||\mathbb{P}^{\pi}\nu_1 - \mathbb{P}^{\pi}\nu_2||_{TV} \leq e^{-1/t_{mix}}||\nu_1 - \nu_2||_{TV}$; ν_1, ν_2 any state occupancy measure.
Every policy is exploratory in the feature space (can be relaxed to only one known exploratory policy)—

$$\lambda_{min}\left(\int_{\mathcal{X}} \sum_a \pi(a|x)\phi(x,a)\phi(x,a)^T d\nu^{\pi}(x)dx\right) \geq \sigma$$

- **Primal-dual adaptation** of MDP-EXP2 [Wei et al.'21]
  - $\pi_k(a|x) \propto \pi_{k-1}\exp(\phi(x,a)^T(w_{r,k} + Y_k w_g^k))$ at epoch k.
    - Divide T in $B = O((\log T)^2 t_{mix}/\sigma$ epochs, every epoch is divided in $2N = O(t_{mix}\log T)$ periods
      - In each epoch, collect rewards (or, utilities) for the last N time-steps (will allow to reach the steady-state distribution in first N time-steps); also allow achieve *unbiasedness* (almost) in estimating q function.
      - Fit $w_{j,k}$ to the collected reward (or, utility) by solving linear regression.
  - Update dual-variable $Y_k = Y_{k-1} + \eta(b - \hat{J}_k)$, $\hat{J}_k$: average of utilities collected over total $B - N$

Algorithm 3

- **Computationally efficient algorithm yet $\tilde{\mathcal{O}}(\sqrt{T})$ regret and violation (first such result for linear CMDP).**
- Additional assumptions:

  Finite mixing time— $||\mathbb{P}^{\pi}\nu_1 - \mathbb{P}^{\pi}\nu_2||_{TV} \leq e^{-1/t_{mix}}||\nu_1 - \nu_2||_{TV}$; ν_1, ν_2 any state occupancy measure.

  Every policy is exploratory in the feature space (can be relaxed to only one known exploratory policy)—

  $$\lambda_{min}\left(\int_{\mathcal{X}} \sum_a \pi(a|x)\phi(x,a)\phi(x,a)^T d\nu^{\pi}(x)dx\right) \geq \sigma$$

- **Primal-dual adaptation** of MDP-EXP2 [Wei et al.'21]
  - $\pi_k(a|x) \propto \pi_{k-1}\exp(\phi(x,a)^T(w_{r,k} + Y_k w_g^k))$ at epoch k.
    - Divide T in $B = O((\log T)^2 t_{mix}/\sigma$ epochs, every epoch is divided in $2N = O(t_{mix}\log T)$ periods
      - In each epoch, collect rewards (or, utilities) for the last N time-steps (will allow to reach the steady-state distribution in first N time-steps); also allow achieve *unbiasedness* (almost) in estimating q function.
      - Fit $w_{j,k}$ to the collected reward (or, utility) by solving linear regression.
  - Update dual-variable $Y_k = Y_{k-1} + \eta(b - \hat{J}_k)$, $\hat{J}_k$: average of utilities collected over total $B - N$
  - **Can achieve zero violation** tighten the optimization: $b + \epsilon$

Simulation

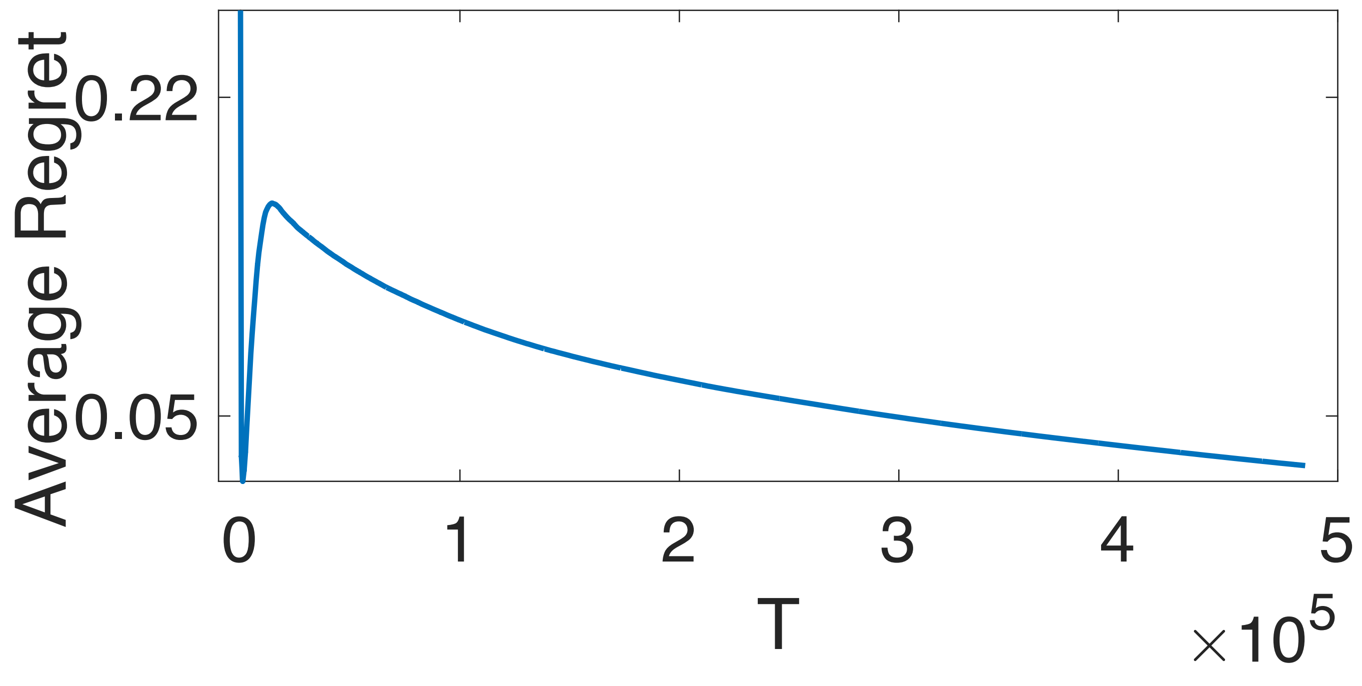
- Transmit a packet $a = 1$ or not $a = 0$
- If transmitted incur cost; **seeks to minimize cost**
- **Also reduces the no. of packets with prob.**
- Multiple packets arrive with certain prob.

Simulation

- Transmit a packet $a = 1$ or not $a = 0$
- If transmitted incur cost; **seeks to minimize cost**
- **Also reduces the no. of packets with prob.**
- Multiple packets arrive with certain prob.
- Constraint: seeks to maintain the no. of packets below a threshold.

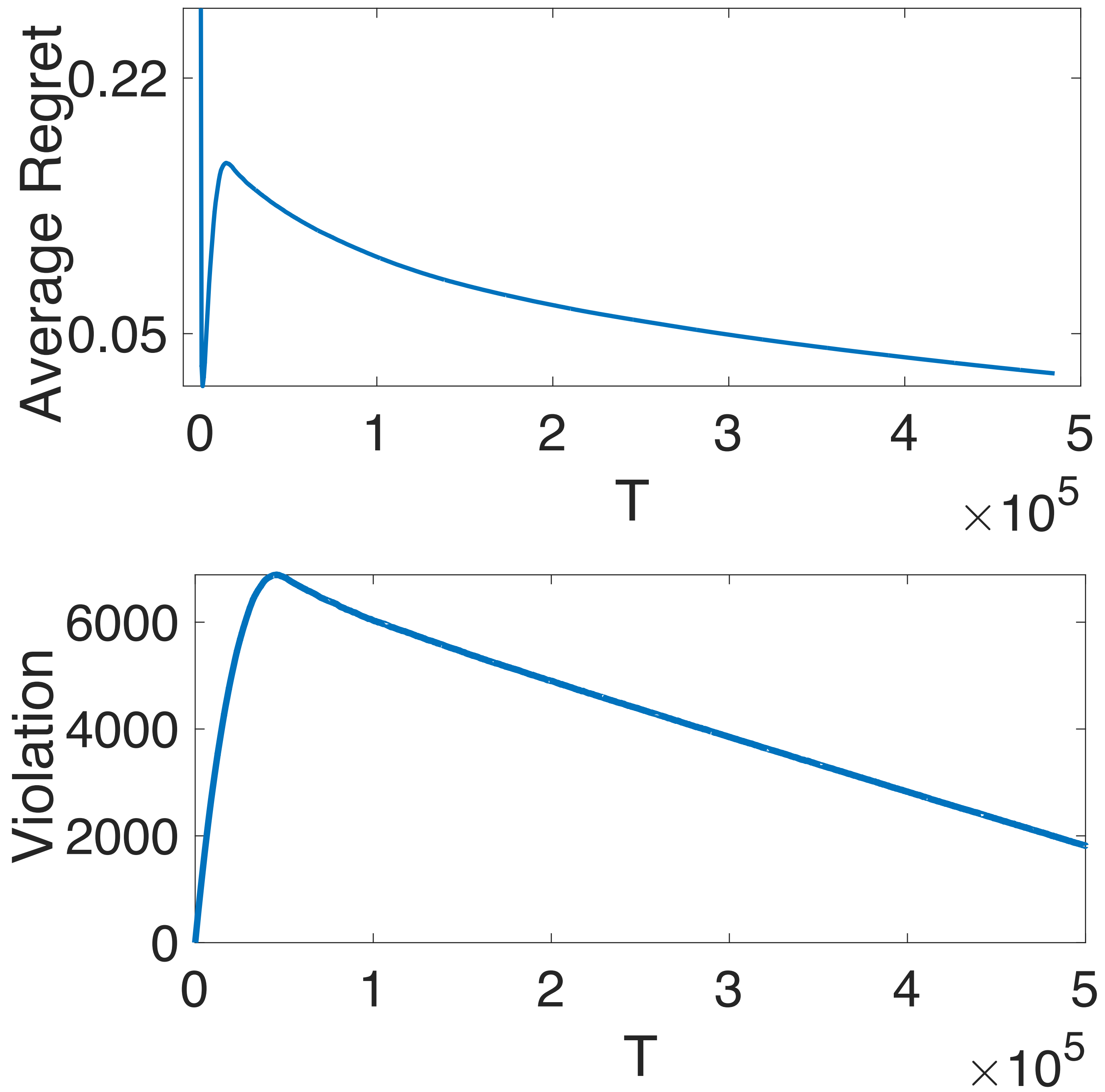
Simulation

- Transmit a packet $a = 1$ or not $a = 0$
- If transmitted incur cost; **seeks to minimize cost**
- **Also reduces the no. of packets with prob.**
- Multiple packets arrive with certain prob.
- Constraint: seeks to maintain the no. of packets below a threshold.



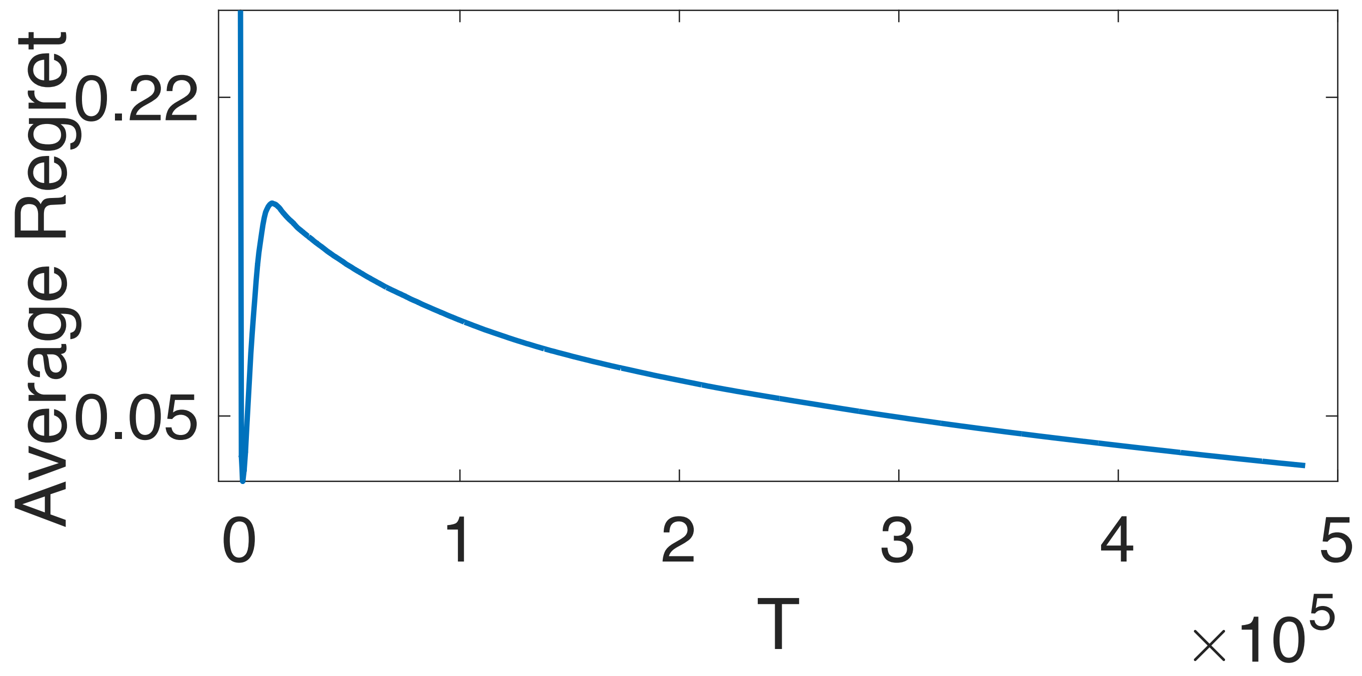
Simulation

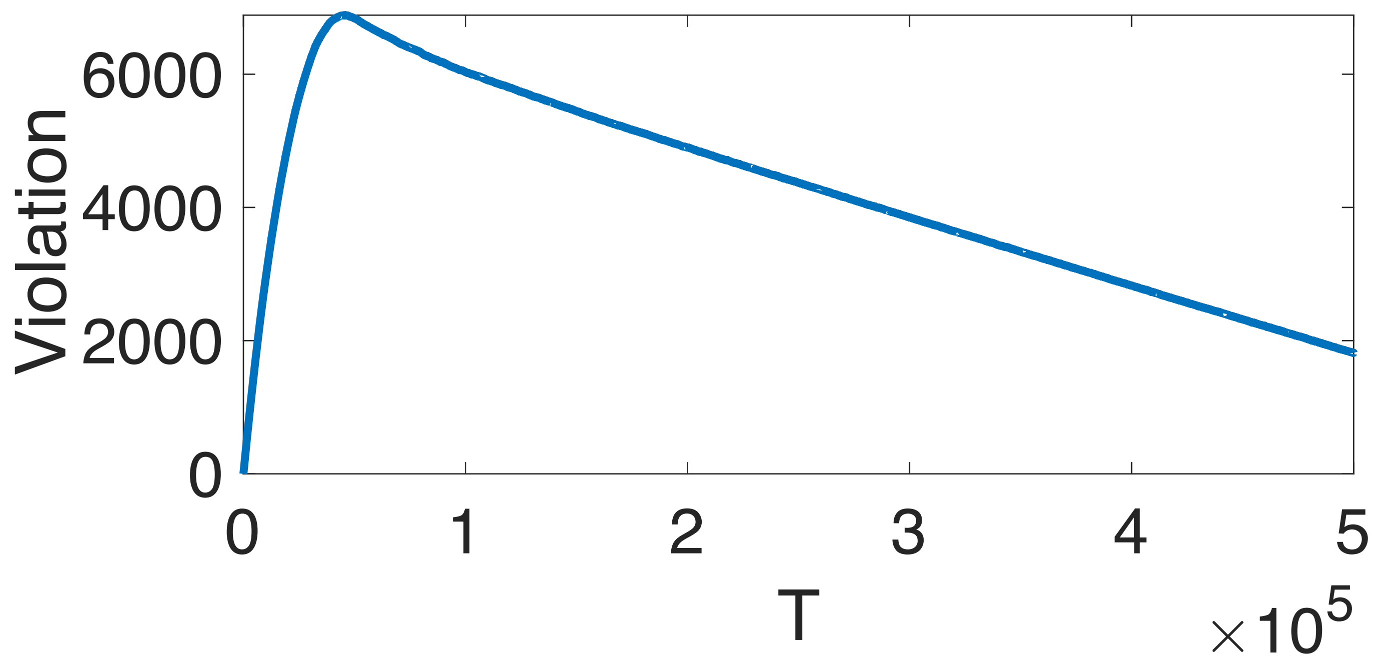
- Transmit a packet $a = 1$ or not $a = 0$
- If transmitted incur cost; **seeks to minimize cost**
- **Also reduces the no. of packets with prob.**
- Multiple packets arrive with certain prob.
- Constraint: seeks to maintain the no. of packets below a threshold.



Simulation

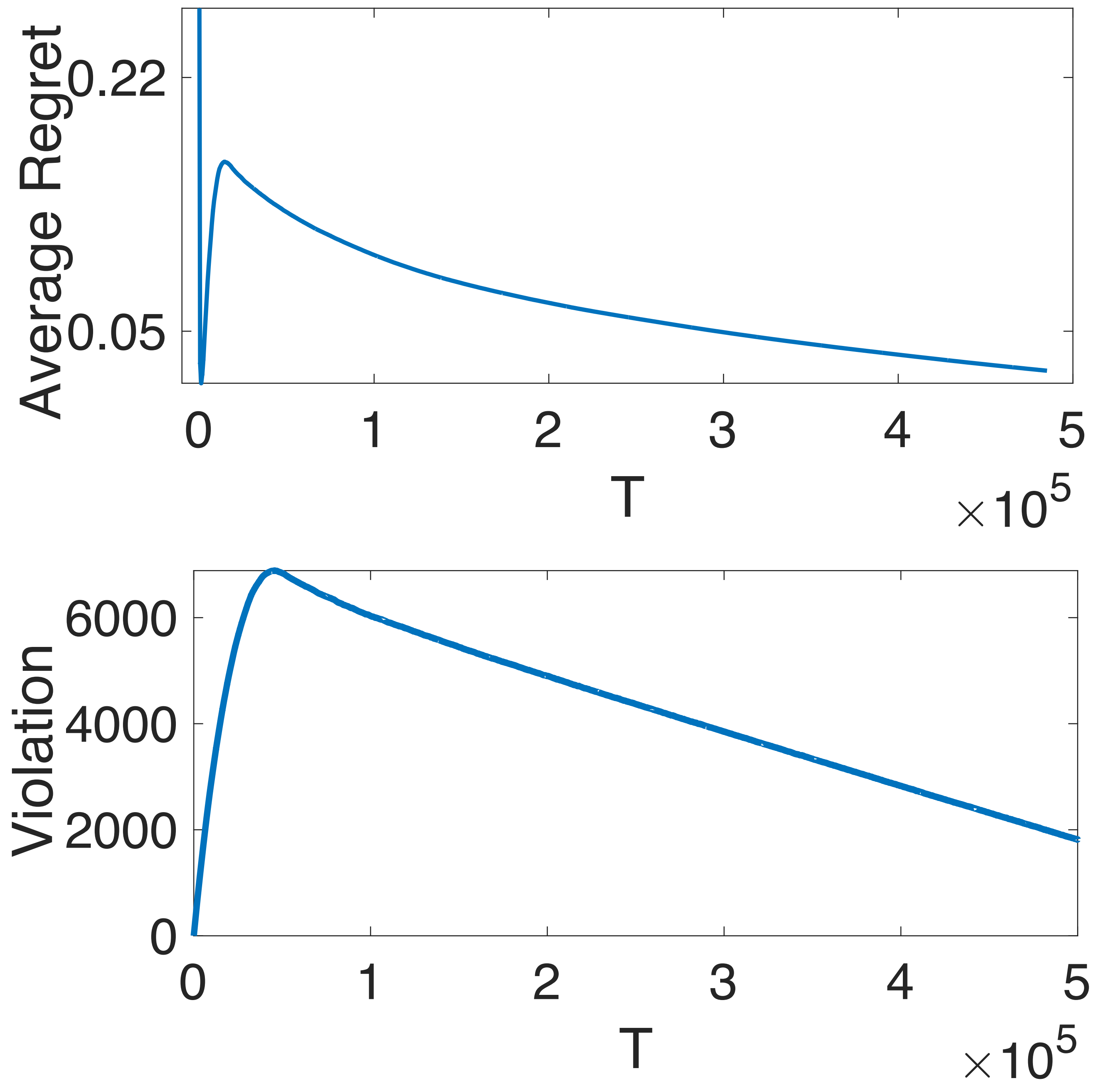
- Transmit a packet $a = 1$ or not $a = 0$
- If transmitted incur cost; **seeks to minimize cost**
- **Also reduces the no. of packets with prob.**
- Multiple packets arrive with certain prob.
- Constraint: seeks to maintain the no. of packets below a threshold.
- **Regret and violation** indeed go to 0.





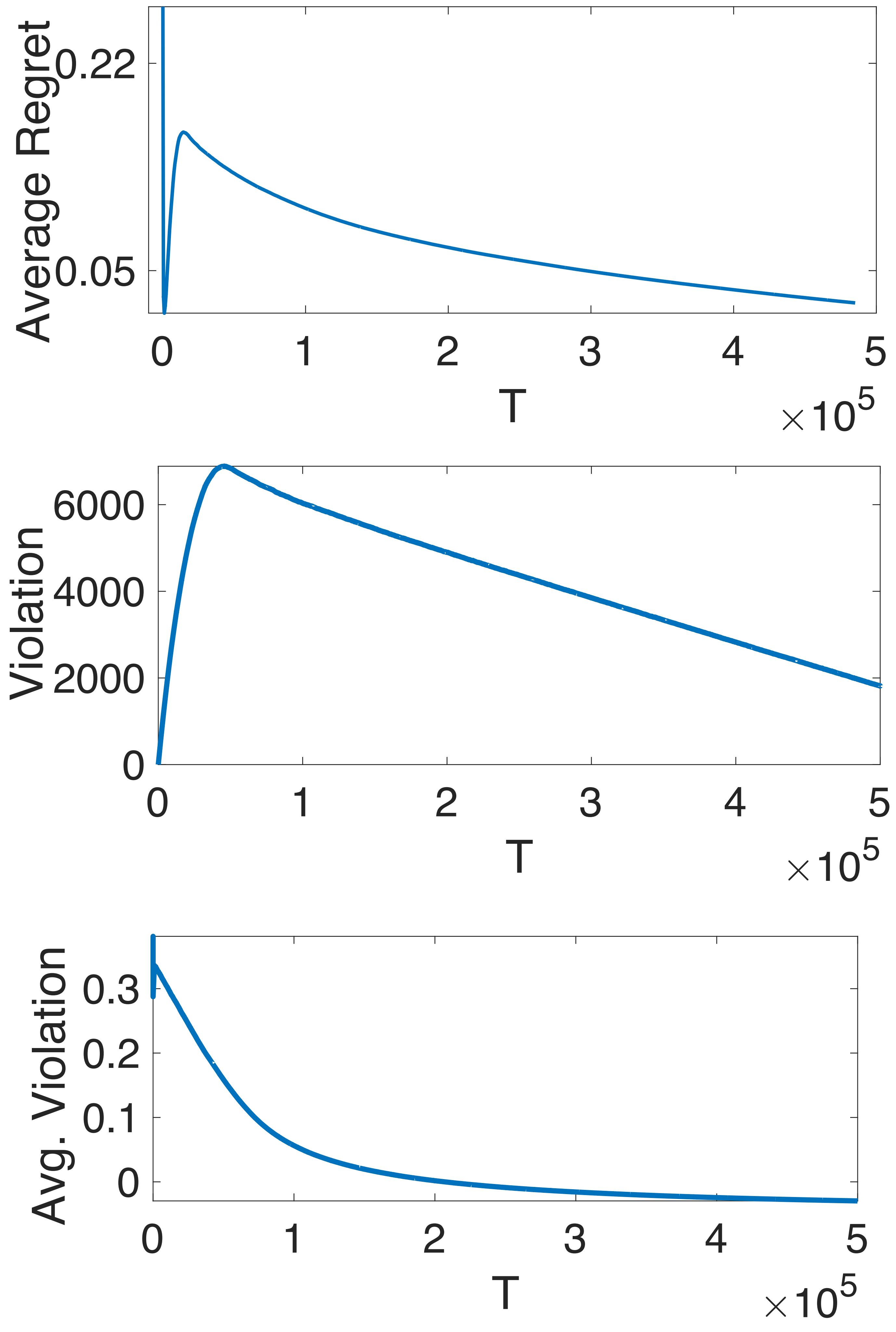
Simulation

- Transmit a packet $a = 1$ or not $a = 0$
- If transmitted incur cost; **seeks to minimize cost**
- **Also reduces the no. of packets with prob.**
- Multiple packets arrive with certain prob.
- Constraint: seeks to maintain the no. of packets below a threshold.
- **Regret and violation** indeed go to 0.
- **Higher ϵ,** violation goes to 0 quicker.



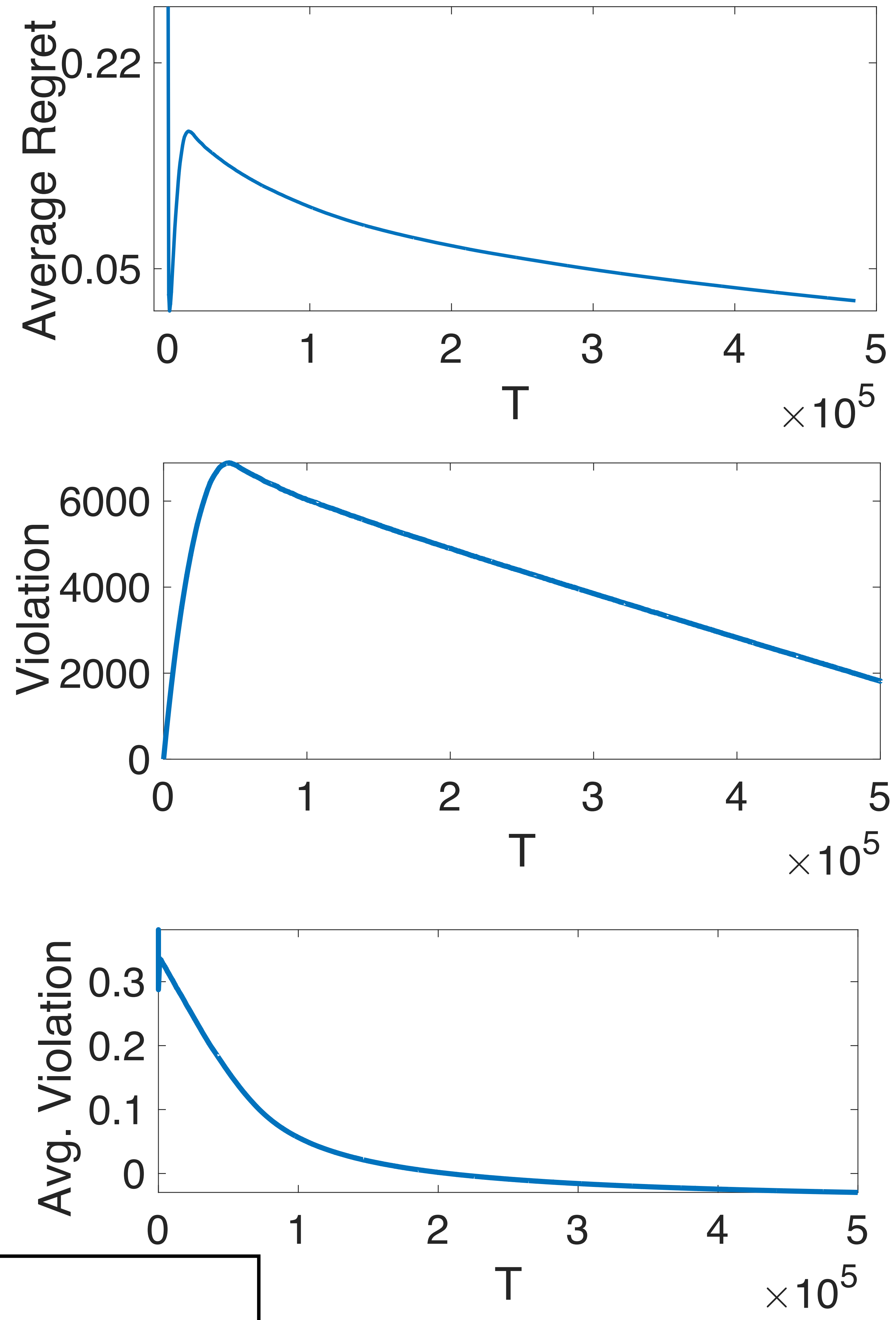
Simulation

- Transmit a packet $a = 1$ or not $a = 0$
- If transmitted incur cost; **seeks to minimize cost**
- **Also reduces the no. of packets with prob.**
- Multiple packets arrive with certain prob.
- Constraint: seeks to maintain the no. of packets below a threshold.
- **Regret and violation** indeed go to 0.
- **Higher ϵ,** violation goes to 0 quicker.



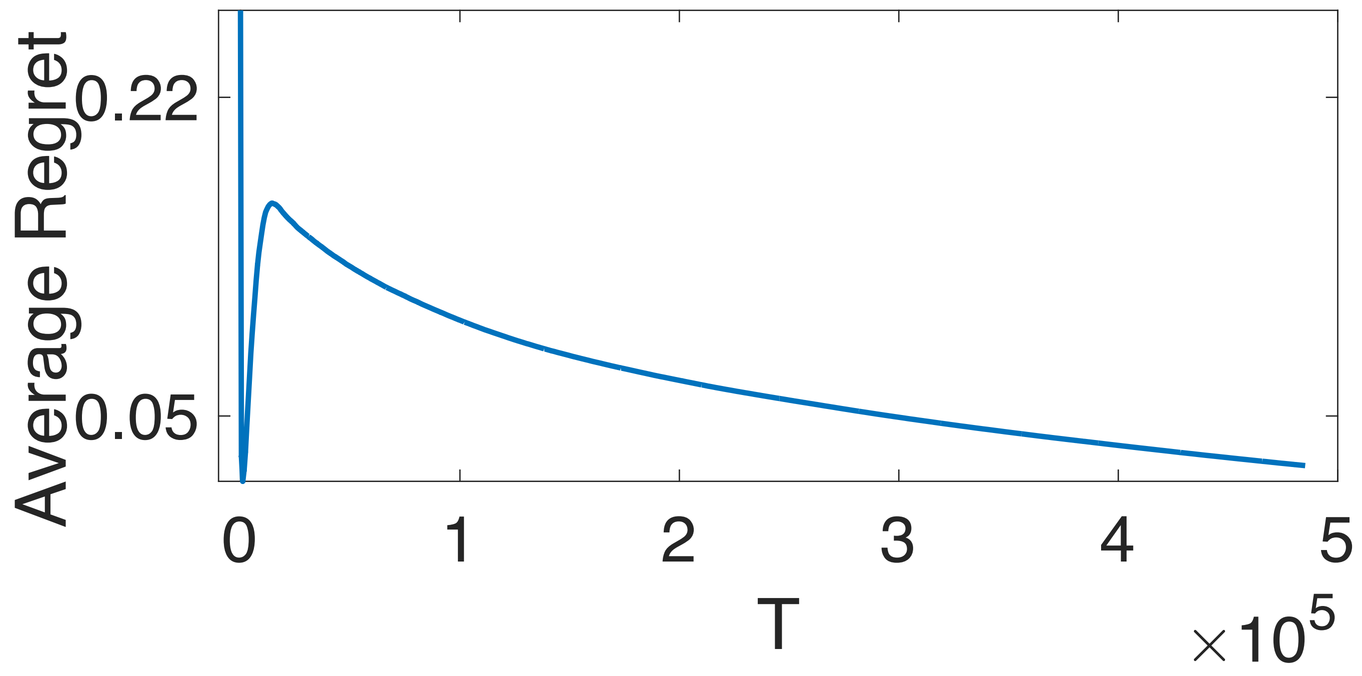
Simulation

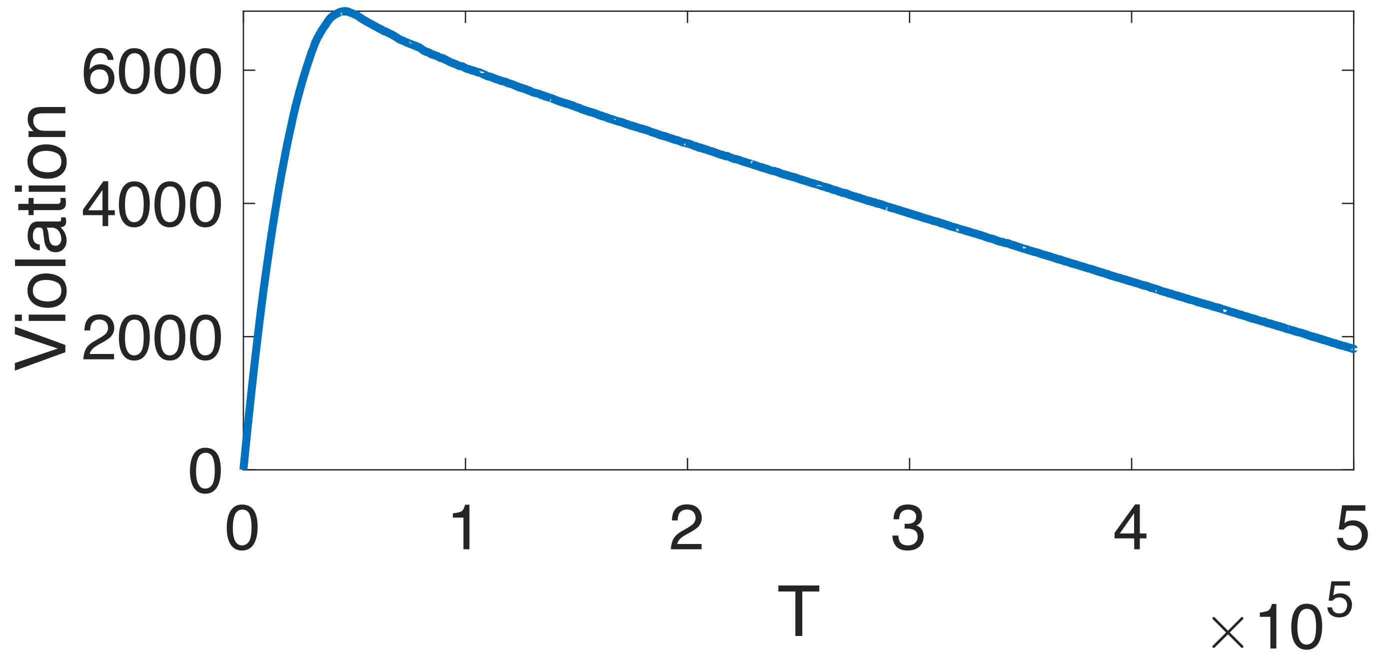
- Transmit a packet $a = 1$ or not $a = 0$
- If transmitted incur cost; **seeks to minimize cost**
- **Also reduces the no. of packets with prob.**
- Multiple packets arrive with certain prob.
- Constraint: seeks to maintain the no. of packets below a threshold.
- **Regret and violation** indeed go to 0.
- **Higher ϵ,** violation goes to 0 quicker.

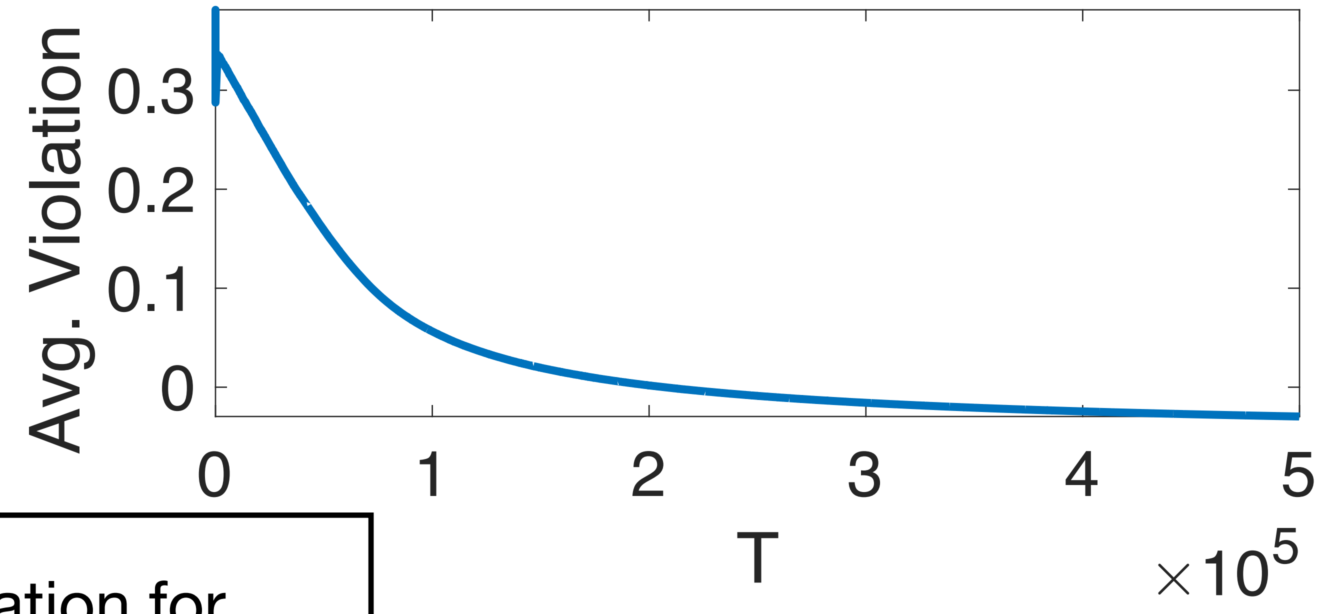


Simulation

- Transmit a packet $a = 1$ or not $a = 0$
- If transmitted incur cost; **seeks to minimize cost**
- **Also reduces the no. of packets with prob.**
- Multiple packets arrive with certain prob.
- Constraint: seeks to maintain the no. of packets below a threshold.
- **Regret and violation** indeed go to 0.
- **Higher ϵ,** violation goes to 0 quicker.



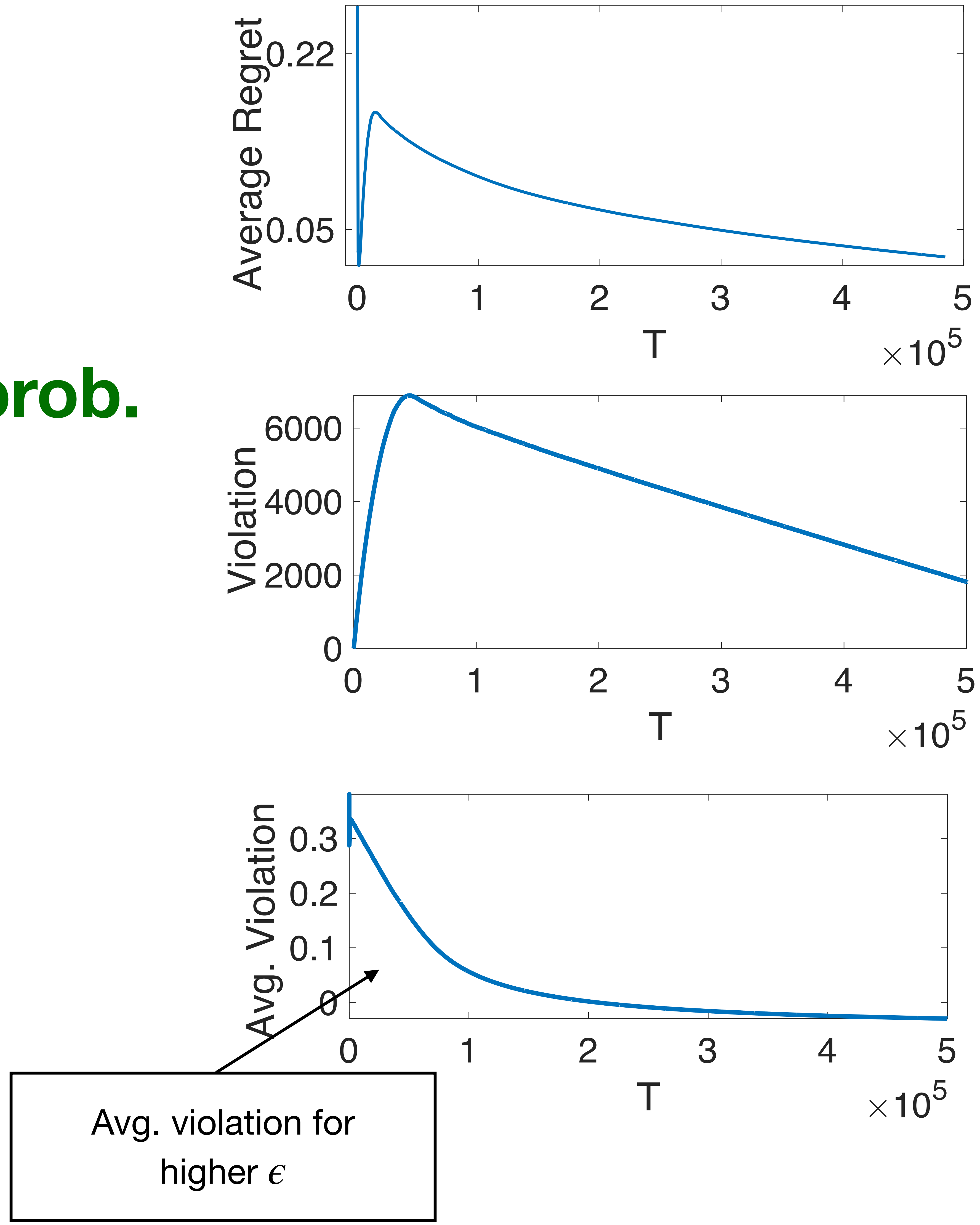




Avg. violation for higher ϵ

Simulation

- Transmit a packet $a = 1$ or not $a = 0$
- If transmitted incur cost; **seeks to minimize cost**
- **Also reduces the no. of packets with prob.**
- Multiple packets arrive with certain prob.
- Constraint: seeks to maintain the no. of packets below a threshold.
- **Regret and violation** indeed go to 0.
- **Higher ϵ,** violation goes to 0 quicker.



Future Research Direction

- Non-linear Function Approximation.

Future Research Direction

- Non-linear Function Approximation.
- Algorithms 1 and 2 depend on span of the optimal value function: can we estimate or eliminate that?

Future Research Direction

- Non-linear Function Approximation.
- Algorithms 1 and 2 depend on span of the optimal value function: can we estimate or eliminate that?
- Will it be possible to achieve $\tilde{\mathcal{O}}(\sqrt{T})$ regret and violation bound under only basic Assumption using **computationally efficient** algorithm (even open for unconstrained case)?

References

Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I Jordan. *Provably efficient reinforcement learning with linear function approximation*. In Conference on Learning Theory, pages 2137– 2143. PMLR, 2020.

Chen-Yu Wei, Mehdi Jafarnia Jahromi, Haipeng Luo, and Rahul Jain. *Learning infinite-horizon average-reward mdps with linear function approximation.* In International Conference on Artificial Intelligence and Statistics, pages 3007–3015. PMLR, 2021.

Liyu Chen, Rahul Jain, and Haipeng Luo. *Learning infinite-horizon average-reward markov decision processes with constraints.* arXiv preprint arXiv:2202.00150, 2022.

Arnob Ghosh, Xingyu Zhou, and Ness Shroff. *Provably efficient model-free constrained rl with linear function approximation.* arXiv preprint arXiv:2206.11889, 2022.

Honghao Wei, Xin Liu, and Lei Ying. *A provably-efficient model-free algorithm for infinite-horizon average-reward constrained markov decision processes.* In AAAI Conference on ArtificialIntelligence, 2022.