Feedback capacity of channels with memory via Reinforcement Learning and Graph-based auxiliary random variable

Oron Sabag, Haim Permuter and Ziv Aharoni

Ben Gurion university

Workshop on Probability and Information Theory

Two main ideas

1. Graph-based auxiliary random variable
2. Reinforcement learning for computing feedback capacity

Auxiliary random variable (r.v.)

- Auxiliary r.v. plays an important role in multi-users problems

Auxiliary random variable (r.v.)

- Auxiliary r.v. plays an important role in multi-users problems

Gelfand-Pinsker: $$C = \max_{P(u|s)P(x|u,s)} I(U;Y) - I(U;S)$$

Wyner-Ziv: $$R = \min_{P(u|x)} I(X;U|Y)$$

Auxiliary random variable (r.v.)

- Auxiliary r.v. plays an important role in multi-users problems

$$\text{Gelfand-Pinsker:} \qquad C = \max_{P(u|s)P(x|u,s)} I(U;Y) - I(U;S)$$

$$\text{Wyner-Ziv:} \qquad R = \min_{P(u|x)} I(X;U|Y)$$

- Auxiliary r.v. converts multi-letter into single-letter

Auxiliary random variable (r.v.)

- Auxiliary r.v. plays an important role in multi-users problems

$$\text{Gelfand-Pinsker:} \quad C = \max_{P(u|s)P(x|u,s)} I(U;Y) - I(U;S)$$

$$\text{Wyner-Ziv:} \quad R = \min_{P(u|x)} I(X;U|Y)$$

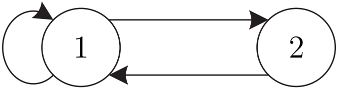
- Auxiliary r.v. converts multi-letter into single-letter
- Auxiliary r.v. are i.i.d.

Graph-based auxiliary r.v.

- Non i.i.d auxiliary r.v.

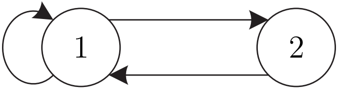
Graph-based auxiliary r.v.

- Non i.i.d auxiliary r.v.
- The auxiliary is represented by a graph



Graph-based auxiliary r.v.

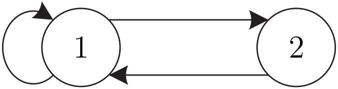
- Non i.i.d auxiliary r.v.
- The auxiliary is represented by a graph



- The graph induces a Markov process

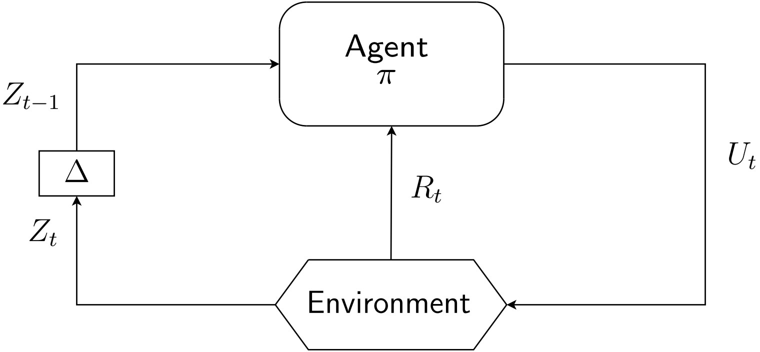
Graph-based auxiliary r.v.

- Non i.i.d auxiliary r.v.
- The auxiliary is represented by a graph



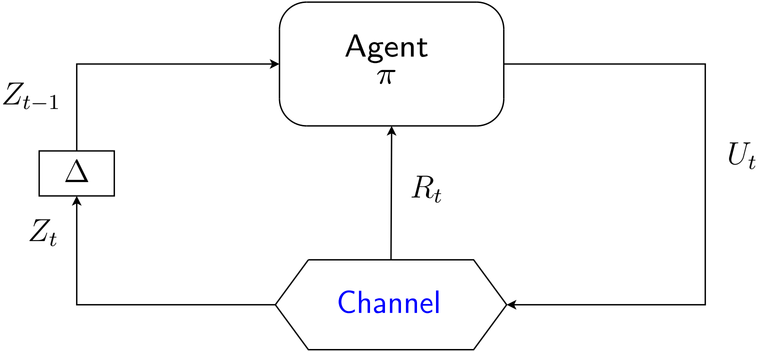
- The graph induces a Markov process
- The single-letter expression is evaluated with the stationary distribution

Reinforcement Learning



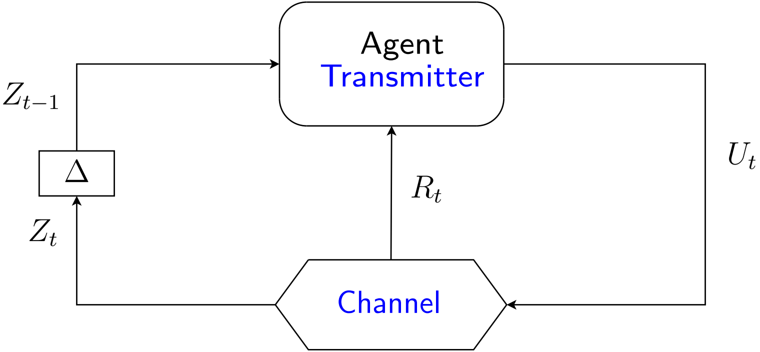
- Z_{t-1} - current state
- U_t - action
- R_t - reward
- Z_t - next state

Reinforcement Learning



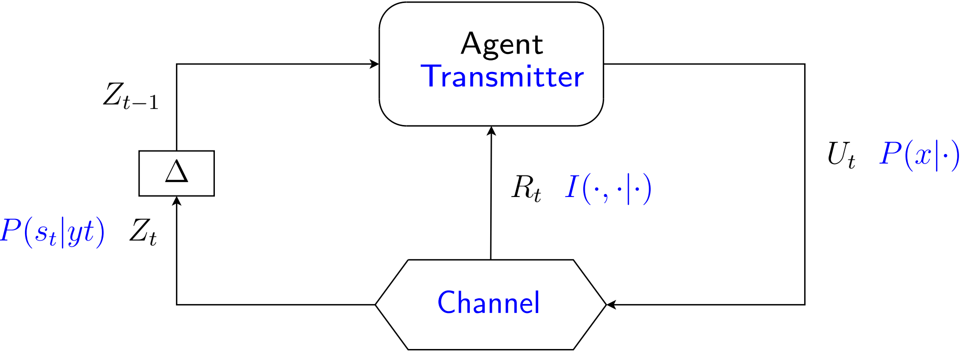
- Z_{t-1} - current state
- U_t - action
- R_t - reward
- Z_t - next state

Reinforcement Learning



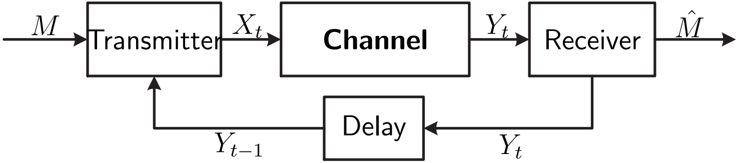
- Z_{t-1} - current state
- U_t - action
- R_t - reward
- Z_t - next state

Reinforcement Learning



- Z_{t-1} - current state
- U_t - action
- R_t - reward
- Z_t - next state

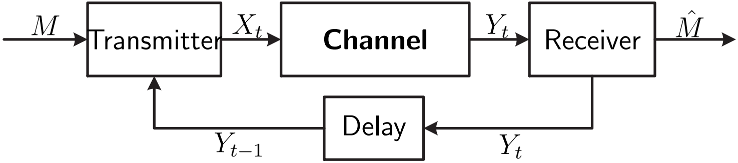
Communication with Feedback



- Unifilar finite state channel (FSC):

$$p\left(y_t | x_t, s_{t-1}\right)$$
$$s_t = f(x_t, y_t, s_{t-1})$$

Communication with Feedback

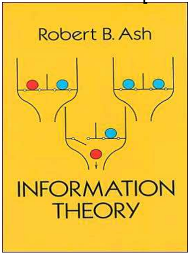


- Unifilar finite state channel (FSC):

$$p\left(y_t | x_t, s_{t-1}\right)$$
$$s_t = f(x_t, y_t, s_{t-1})$$

- **The goal:** compute the capacity and coding scheme

Trapdoor Channel [Blackwell61]



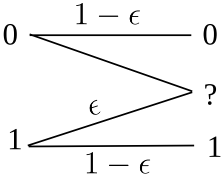
Ising Channel [Berger90]

$$y_i = \begin{cases} x_i, & \text{with prob. } \frac{1}{2} \\ x_{i-1}, & \text{with prob. } \frac{1}{2} \end{cases}$$

Dicode Erasure Channel [Pfister08]

$$y_i = \begin{cases} x_i - x_{i-1}, & \text{w/ prob. } \bar{\epsilon} \\ ?, & \text{w/ prob. } \epsilon \end{cases}$$

Erasure Channel with no repeated 1's



The Capacity

Theorem (P-Cuff-Van Roy-Weissman'08, P-Weissman-Goldsmith'09)

The **feedback capacity** *of unifilar FSC*

$$C_{fb} = \lim_{n\to\infty} \max_{\{p(x_i|s_{i-1},y^{i-1})\}_{i=1}^n} \frac{1}{n} I\left(X^n \to Y^n\right)$$

- The directed information (Massey 1990)

$$I\left(X^n \to Y^n\right) = \sum_{i=1}^{n} I(X^i; Y_i | Y^{i-1})$$

- This is a multi-letter expression

The Q-graph

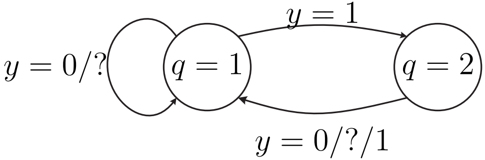
- A *Q-graph* is an irreducible directed graph.

The Q-graph

- A *Q-graph* is an irreducible directed graph.
  Each node has $|\mathcal{Y}|$ different labelled edges.

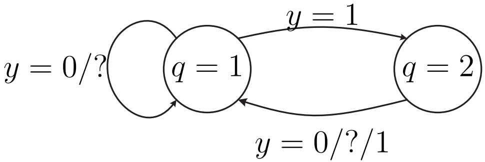
The Q-graph

- A Q-*graph* is an irreducible directed graph. Each node has $|\mathcal{Y}|$ different labelled edges.
- For instance, if $\mathcal{Y} = \{0, 1, ?\}$:



The Q-graph

- A *Q-graph* is an irreducible directed graph. Each node has $|\mathcal{Y}|$ different labelled edges.
- For instance, if $\mathcal{Y} = \{0, 1, ?\}$:

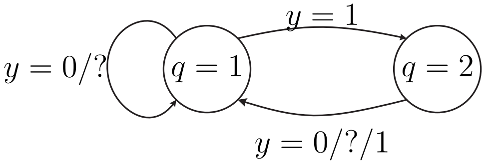


- The Q-graph defines a mapping:

$$\Phi_{i-1} : \mathcal{Y}^{i-1} \to \mathcal{Q} \quad (\text{or}, \quad g : \mathcal{Q} \times \mathcal{Y} \to \mathcal{Q})$$

The Q-graph

- A Q*-graph* is an irreducible directed graph.
  Each node has $|\mathcal{Y}|$ different labelled edges.
- For instance, if $\mathcal{Y} = \{0, 1, ?\}$:



- The Q-graph defines a mapping:

$$\Phi_{i-1} : \mathcal{Y}^{i-1} \to \mathcal{Q} \quad (\text{or}, \quad g : \mathcal{Q} \times \mathcal{Y} \to \mathcal{Q})$$

- Each outputs sequence is Q-uantized

Feedback capacity

Theorem [Sabag/P./Pfister17]

The feedback capacity of a unifilar FSC is bounded by

$$C_{fb} \le \sup_{p(x|s,q)} I(X,S;Y|Q), \qquad \forall Q\text{-graph}$$

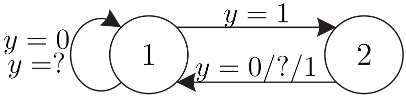
Feedback capacity

Theorem [Sabag/P./Pfister17]

The feedback capacity of a unifilar FSC is bounded by

$$C_{fb} \leq \sup_{p(x|s,q)} I(X,S;Y|Q), \qquad \forall Q\text{-graph}$$

Q-graph defines a mapping: $(\mathcal{Q}_{i-1}, \mathcal{Y}_i) \to \mathcal{Q}_i$



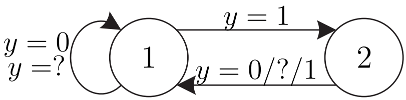
Feedback capacity

Theorem [Sabag/P./Pfister17]

The feedback capacity of a unifilar FSC is bounded by

$$C_{fb} \leq \sup_{p(x|s,q)} I(X,S;Y|Q), \qquad \forall Q\text{-graph}$$

Q-graph defines a mapping: $(\mathcal{Q}_{i-1}, \mathcal{Y}_i) \to \mathcal{Q}_i$



The Q-graph and $P(x|s,q)$ induces

$$p(s,q,x,y) = \pi(s,q)p(x|s,q)p(y|s,x)$$

Feedback capacity

Theorem [Sabag/P./Pfiser16]

The feedback capacity of a unifilar FSC is bounded by

$$C_{fb} \leq \sup_{p(x|s,q)} I(X,S;Y|Q), \qquad \forall Q\text{-graph}$$

- If $|\mathcal{Q}|$ unbounded then its also achievable, i.e.,

$$C_{fb} = \inf_{Q} \sup_{p(x|s,q)} I(X,S;Y|Q)$$

Feedback capacity

Theorem [Sabag/P./Pfiser16]

The feedback capacity of a unifilar FSC is bounded by

$$C_{fb} \leq \sup_{p(x|s,q)} I(X,S;Y|Q), \qquad \forall Q\text{-graph}$$

- If $|\mathcal{Q}|$ unbounded then its also achievable, i.e.,

$$C_{fb} = \inf_{Q} \sup_{p(x|s,q)} I(X,S;Y|Q)$$

- For all known cases the upper bound is tight $|\mathcal{Q}| \leq 4$,

Sketch Proof

$$C_{fb} = \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} I(X^n \to Y^n)$$

Sketch Proof

$$\begin{aligned} C_{fb} &= \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} I(X^n \rightarrow Y^n) \\ &\triangleq \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X^i; Y_i | Y^{i-1}) \end{aligned}$$

Sketch Proof

$$
\begin{aligned}
C_{fb} &= \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} I(X^n \to Y^n) \\
&\triangleq \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X^i; Y_i|Y^{i-1}) \\
&= \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X^i, S_{i-1}(X^{i-1}, Y^{i-1}); Y_i|Y^{i-1})
\end{aligned}
$$

Sketch Proof

$$
\begin{aligned}
C_{fb} &= \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} I(X^n \to Y^n) \\
&\triangleq \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X^i; Y_i | Y^{i-1}) \\
&= \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X^i, S_{i-1}; Y_i | Y^{i-1})
\end{aligned}
$$

Sketch Proof

$$\begin{aligned}
C_{fb} &= \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} I(X^n \to Y^n) \\
&\triangleq \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X^i; Y_i|Y^{i-1}) \\
&= \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X_i, S_{i-1}; Y_i|Y^{i-1})
\end{aligned}$$

Sketch Proof

$$
\begin{aligned}
C_{fb} &= \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} I(X^n \to Y^n) \\
&\triangleq \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X^i; Y_i | Y^{i-1}) \\
&= \max_{P(x_i|s_{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X_i, S_{i-1}; Y_i | Y^{i-1})
\end{aligned}
$$

Sketch Proof

$$
\begin{aligned}
C_{fb} &= \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} I(X^n \to Y^n) \\
&\triangleq \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X^i; Y_i | Y^{i-1}) \\
&= \max_{P(x_i|s_{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X_i, S_{i-1}; Y_i | Y^{i-1}) \\
&\leq \max_{P(x_i|s_{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X_i, S_{i-1}; Y_i | Q_{i-1}(Y^{i-1}))
\end{aligned}
$$

Sketch Proof

$$
\begin{aligned}
C_{fb} &= \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} I(X^n \to Y^n) \\
&\triangleq \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X^i; Y_i|Y^{i-1}) \\
&= \max_{P(x_i|s_{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X_i, S_{i-1}; Y_i|Y^{i-1}) \\
&\leq \max_{P(x_i|s_{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X_i, S_{i-1}; Y_i|Q_{i-1})
\end{aligned}
$$

Sketch Proof

$$
\begin{aligned}
C_{fb} &= \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} I(X^n \to Y^n) \\
&\triangleq \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X^i; Y_i | Y^{i-1}) \\
&= \max_{P(x_i|s_{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X_i, S_{i-1}; Y_i | Y^{i-1}) \\
&\leq \max_{P(x_i|s_{i-1},q_{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X_i, S_{i-1}; Y_i | Q_{i-1})
\end{aligned}
$$

Sketch Proof

$$
\begin{aligned}
C_{fb} &= \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} I(X^n \to Y^n) \\
&\triangleq \max_{P(x_i|x^{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X^i; Y_i | Y^{i-1}) \\
&= \max_{P(x_i|s_{i-1},y^{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X_i, S_{i-1}; Y_i | Y^{i-1}) \\
&\leq \max_{P(x_i|s_{i-1},q_{i-1})} \frac{1}{n} \sum_{i=1}^{n} I(X_i, S_{i-1}; Y_i | Q_{i-1}) \\
&= \max_{P(x|s,q)} I(X, S; Y | Q)
\end{aligned}
$$

Examples

Theorem [Sabag/P./Pfister17]

$$C_{fb} \leq \sup_{p(x|s,q)} I(X,S;Y|Q), \qquad \forall Q\text{-graph}$$

Ex1: Memoryless channel, $|\mathcal{S}| = 1$. Choose Q constant.

$$C_{fb} \leq \sup_{p(x|s,q)} I(X,S;Y|Q) = \sup_{p(x)} I(X;Y)$$

Examples

Theorem [Sabag/P./Pfister17]

$$C_{fb} \leq \sup_{p(x|s,q)} I(X,S;Y|Q), \qquad \forall Q\text{-graph}$$

Ex1: Memoryless channel, $|\mathcal{S}| = 1$. Choose Q constant.

$$C_{fb} \leq \sup_{p(x|s,q)} I(X,S;Y|Q) = \sup_{p(x)} I(X;Y)$$

Ex2: State known at the decoder and encoder. Choose $Q = S$

$$C_{fb} \leq \sup_{p(x|s,q)} I(X,S;Y,S|Q) = \sup_{p(x|s)} I(X;Y|S)$$

Upper bound

A unifying capacity formula

For all *solved* channels in the literature,

$$C_{\mathsf{fb}} = \max I(X, S; Y|Q).$$

Upper bound

A unifying capacity formula

For all *solved* channels in the literature,

$$C_{\mathsf{fb}} = \max I(X, S; Y|Q).$$

1. Discrete memoryless channels $C_{\mathsf{fb}} = \max I(X;Y)$
2. Markov channels $C_{\mathsf{fb}} = \max_{p(x|s)} I(X;Y|S)$

Upper bound

A unifying capacity formula

For all *solved* channels in the literature,

$$C_{\mathsf{fb}} = \max I(X,S;Y|Q).$$

1. Discrete memoryless channels $C_{\mathsf{fb}} = \max I(X;Y)$
2. Markov channels $C_{\mathsf{fb}} = \max_{p(x|s)} I(X;Y|S)$
3. Trapdoor channel $C_{\mathsf{fb}} = \log\left(\frac{1+\sqrt{5}}{2}\right)$

Upper bound

A unifying capacity formula

For all *solved* channels in the literature,

$$C_{\mathsf{fb}} = \max I(X, S; Y|Q).$$

1. Discrete memoryless channels $C_{\mathsf{fb}} = \max I(X;Y)$
2. Markov channels $C_{\mathsf{fb}} = \max_{p(x|s)} I(X;Y|S)$
3. Trapdoor channel $C_{\mathsf{fb}} = \log\left(\frac{1+\sqrt{5}}{2}\right)$
4. B. Symmetric Ising channel $C_{\mathsf{fb}} = \max \frac{2H_2(a)}{3+a}$

Upper bound

A unifying capacity formula

For all *solved* channels in the literature,

$$C_{\mathsf{fb}} = \max I(X, S; Y|Q).$$

1. Discrete memoryless channels $C_{\mathsf{fb}} = \max I(X;Y)$
2. Markov channels $C_{\mathsf{fb}} = \max_{p(x|s)} I(X;Y|S)$
3. Trapdoor channel $C_{\mathsf{fb}} = \log\left(\frac{1+\sqrt{5}}{2}\right)$
4. B. Symmetric Ising channel $C_{\mathsf{fb}} = \max \frac{2H_2(a)}{3+a}$
5. Input-constrained BEC $C_{\mathsf{fb}}(\epsilon) = \max_a \frac{H_2(a)}{a+\frac{1}{1-\epsilon}}$

Upper bound

A unifying capacity formula

For all *solved* channels in the literature,

$$C_{\mathsf{fb}} = \max I(X, S; Y|Q).$$

1. Discrete memoryless channels $C_{\mathsf{fb}} = \max I(X;Y)$
2. Markov channels $C_{\mathsf{fb}} = \max_{p(x|s)} I(X;Y|S)$
3. Trapdoor channel $C_{\mathsf{fb}} = \log\left(\frac{1+\sqrt{5}}{2}\right)$
4. B. Symmetric Ising channel $C_{\mathsf{fb}} = \max \frac{2H_2(a)}{3+a}$
5. Input-constrained BEC $C_{\mathsf{fb}}(\epsilon) = \max_a \frac{H_2(a)}{a+\frac{1}{1-\epsilon}}$
6. B. Ising channels $C_{\mathsf{fb}}(p) = H_2\left(\frac{1}{2^{H_2(p)}+1}\right) - \frac{H_2(p)}{2^{H_2(p)}+1}$

Upper bound

A unifying capacity formula

For all *solved* channels in the literature,

$$C_{\text{fb}} = \max I(X, S; Y|Q).$$

1. Discrete memoryless channels $C_{\text{fb}} = \max I(X;Y)$
2. Markov channels $C_{\text{fb}} = \max_{p(x|s)} I(X;Y|S)$
3. Trapdoor channel $C_{\text{fb}} = \log\left(\frac{1+\sqrt{5}}{2}\right)$
4. B. Symmetric Ising channel $C_{\text{fb}} = \max \frac{2H_2(a)}{3+a}$
5. Input-constrained BEC $C_{\text{fb}}(\epsilon) = \max_a \frac{H_2(a)}{a+\frac{1}{1-\epsilon}}$
6. B. Ising channels $C_{\text{fb}}(p) = H_2\left(\frac{1}{2^{H_2(p)}+1}\right) - \frac{H_2(p)}{2^{H_2(p)}+1}$
7. Input-constrained BSC $\max_a \frac{H_2(a)+aH_2\left(\frac{\alpha(1-\alpha)}{a}\right)}{1+a} - H_2(\alpha)$

Upper bound

A unifying capacity formula

For all *solved* channels in the literature,

$$C_{\mathsf{fb}} = \max I(X, S; Y|Q).$$

1. Discrete memoryless channels $C_{\mathsf{fb}} = \max I(X;Y)$
2. Markov channels $C_{\mathsf{fb}} = \max_{p(x|s)} I(X;Y|S)$
3. Trapdoor channel $C_{\mathsf{fb}} = \log\left(\frac{1+\sqrt{5}}{2}\right)$
4. B. Symmetric Ising channel $C_{\mathsf{fb}} = \max \frac{2H_2(a)}{3+a}$
5. Input-constrained BEC $C_{\mathsf{fb}}(\epsilon) = \max_a \frac{H_2(a)}{a+\frac{1}{1-\epsilon}}$
6. B. Ising channels $C_{\mathsf{fb}}(p) = H_2\left(\frac{1}{2^{H_2(p)}+1}\right) - \frac{H_2(p)}{2^{H_2(p)}+1}$
7. Input-constrained BSC $\max_a \frac{H_2(a)+aH_2\left(\frac{\alpha(1-\alpha)}{a}\right)}{1+a} - H_2(\alpha)$

For these channels, capacity is attained with $|\mathcal{Q}| \leq 4$

Markov Decision Process (MDP) Formulation

Theorem (P-Cuff-Van Roy-Weissman'08)

The feedback capacity of a unifilar FSC can be formulated as a Markov decision process

Markov Decision Process (MDP) Formulation

Theorem (P-Cuff-Van Roy-Weissman'08)

The feedback capacity of a unifilar FSC can be formulated as a Markov decision process

- MDP's state: $p(s_t|y^t)$

Markov Decision Process (MDP) Formulation

Theorem (P-Cuff-Van Roy-Weissman'08)

The feedback capacity of a unifilar FSC can be formulated as a Markov decision process

- MDP's state: $p(s_t|y^t)$

Solution methods:

- Dynamic programming - value iteration algorithm
  Effective only for binary alphabet

Markov Decision Process (MDP) Formulation

Theorem (P-Cuff-Van Roy-Weissman'08)

The feedback capacity of a unifilar FSC can be formulated as a Markov decision process

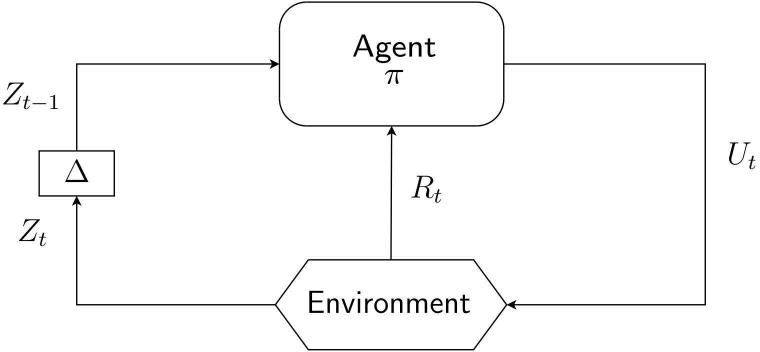
- MDP's state: $p(s_t|y^t)$

Solution methods:

- Dynamic programming - value iteration algorithm
  Effective only for binary alphabet

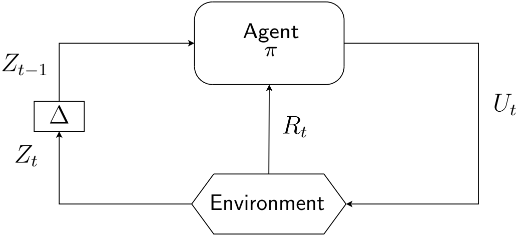
- Reinforcement learning (RL)
  Effective for large alphabets

Reinforcement Learning



- Z_{t-1} - current state
- U_t - action
- R_t - reward
- Z_t - next state

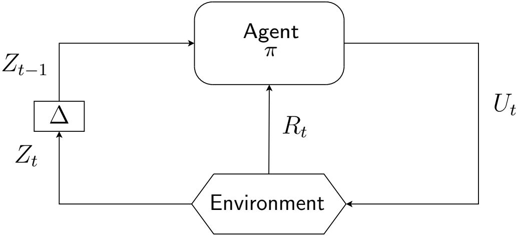
Reinforcement Learning



- The goal: maximize the expected average reward

$$\mathbb{E}_{\pi}\left[G\right] = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}_{\pi}\left[R_t\right]$$

Reinforcement Learning



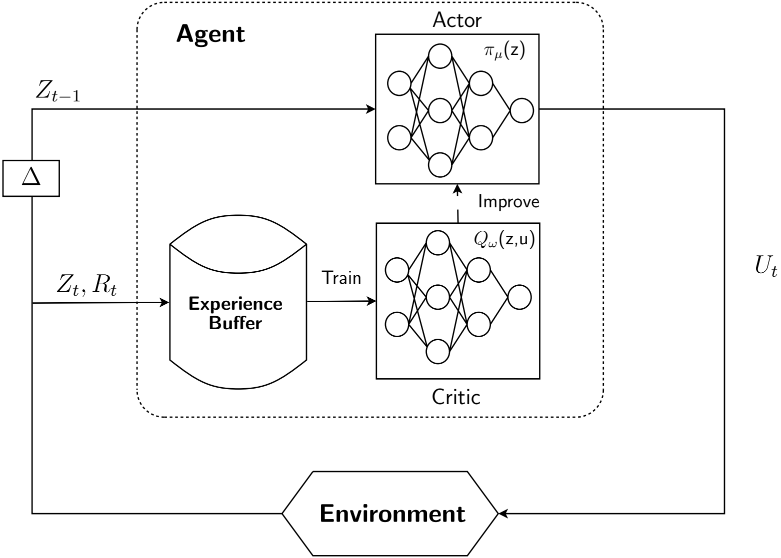
- The goal: maximize the expected average reward

$$\mathbb{E}_\pi\left[G\right] = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}_\pi\left[R_t\right]$$

- The state-action value function

$$Q_\pi(z,u) = \mathbb{E}_\pi\left[G | Z_1 = z, U_1 = u\right]$$

Q-Learning Approach



Deep Deterministic Policy Gradient, (Lillicrap et al.'16)

- Draw N interactions from experience (z_i, u_i, r_i, z_i')
- Train critic: minimize by ω

$$\frac{1}{N}\sum_{i=1}^{N}\left[Q_\omega\left(z_i, u_i\right) - \left[r_i - \rho_\mu + Q_\omega\left(z_i', \pi_\mu(z_i')\right)\right]\right]^2$$

- Improve actor: maximize by μ

$$\frac{1}{N}\sum_{i=1}^{N}\nabla_u Q_\omega\left(z_i, u\right)\big|_{u=\pi_\mu(z_i)}\nabla_\mu \pi_\mu\left(z_i\right)$$

The Ising Channel

- Defined by Berger and Bonomi (1990):

$$Y_i = \begin{cases} S_{i-1} & \text{, w.p. } 0.5 \\ X_i & \text{, w.p. } 0.5 \end{cases}, \quad S_{i-1} = X_{i-1}$$

- Models channel with ISI, magnetic recording

The Ising Channel

- Defined by Berger and Bonomi (1990):

$$Y_i = \begin{cases} S_{i-1} & \text{, w.p. } 0.5 \\ X_i & \text{, w.p. } 0.5 \end{cases}, \quad S_{i-1} = X_{i-1}$$

- Models channel with ISI, magnetic recording

- Solved the binary case (Elischo-P'14, Sharov-Roth'16)

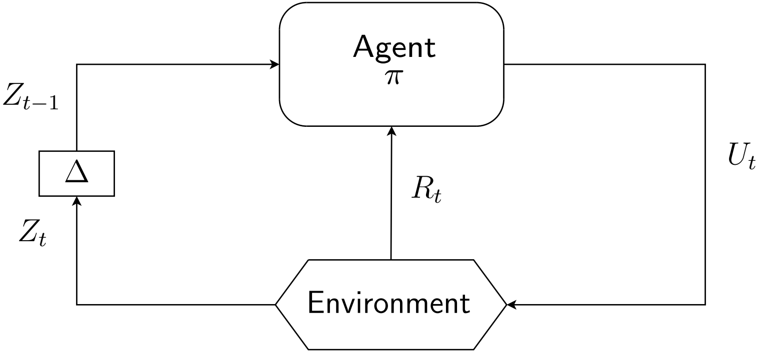
The Ising Channel

- Defined by Berger and Bonomi (1990):

$$Y_i = \begin{cases} S_{i-1} & \text{, w.p. } 0.5 \\ X_i & \text{, w.p. } 0.5 \end{cases}, \quad S_{i-1} = X_{i-1}$$

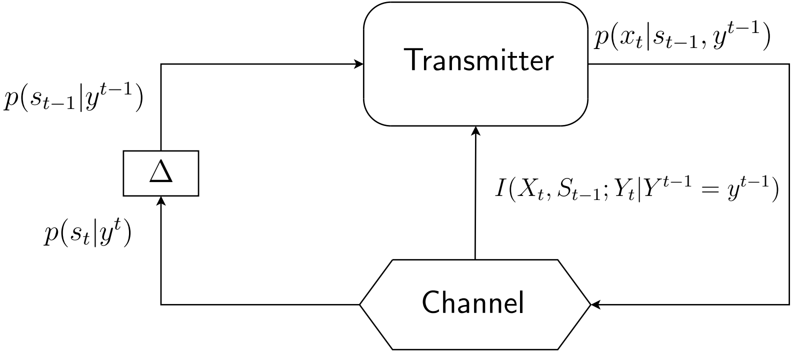
- Models channel with ISI, magnetic recording
- Solved the binary case (Elischo-P'14, Sharov-Roth'16)
- **The goal:** apply RL methodology to larger alphabets

Back to the Feedback Capacity



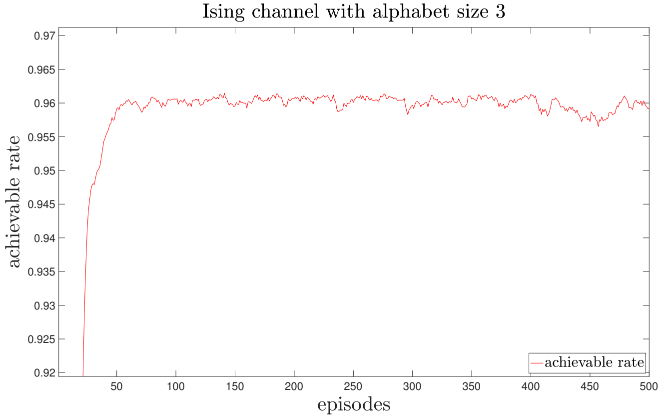
The goal: maximize **average reward**

Back to the Feedback Capacity



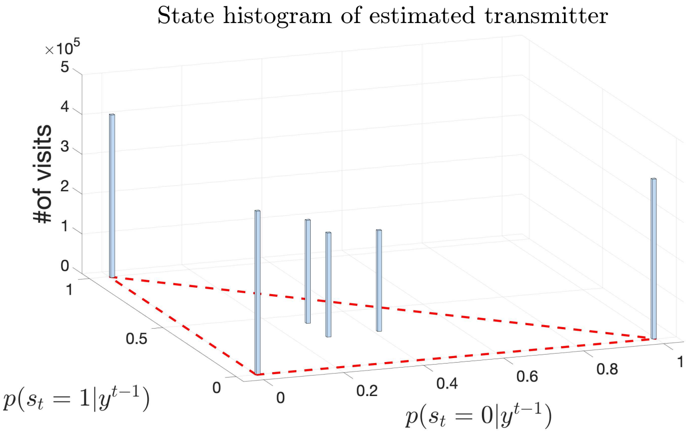
The goal: maximize **achievable rate**

Numerical results - Achievable Rate

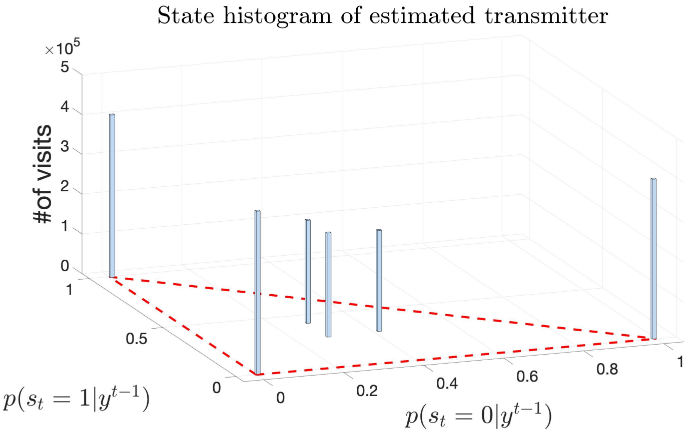


- Reveal the **structure** of the optimal solution

Properties of the Estimated Solution

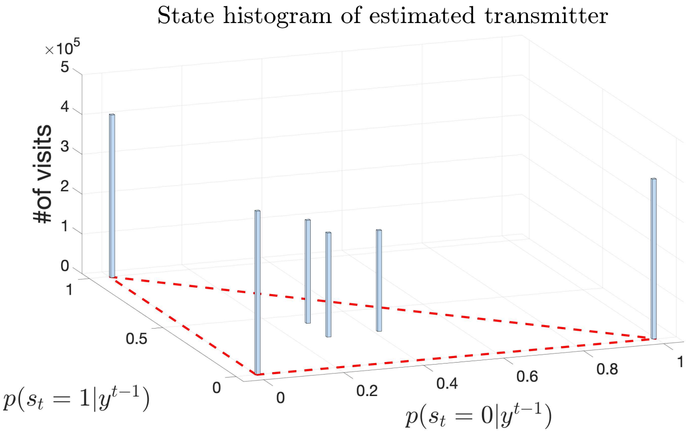


Properties of the Estimated Solution



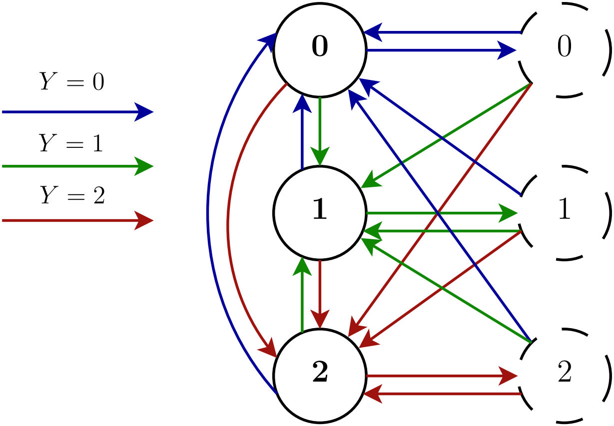
- Optimal input distribution structure

Properties of the Estimated Solution

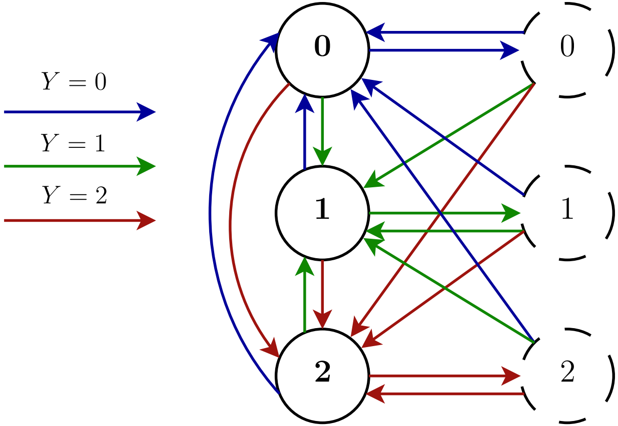


- Optimal input distribution structure
- Transitions between states as function of channel's output

Transitions of states by a Q-graph

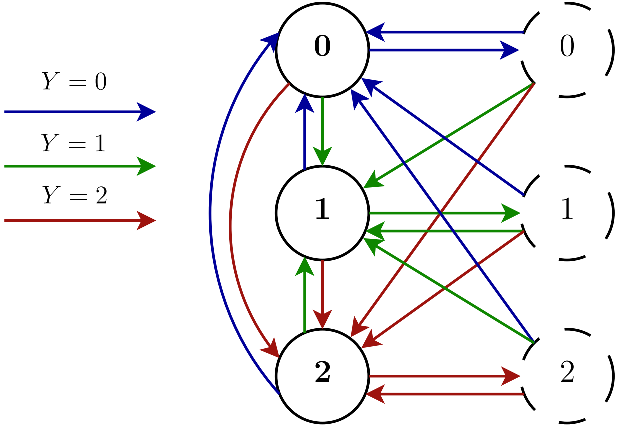


Transitions of states by a Q-graph



- Design coding scheme

Transitions of states by a Q-graph

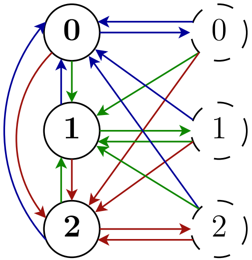


- Design coding scheme
- Prove upper-bound

Coding scheme

- **Pre-transmission:** generate an information sequence s.t.

$$x_i = \begin{cases} x_{i-1} & \text{, w.p } p \\ \text{Unif}[\mathcal{X} \backslash x_{i-1}] & \text{, w.p } 1-p \end{cases}$$

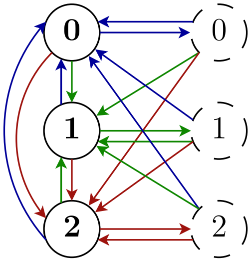


Coding scheme

- **Pre-transmission:** generate an information sequence s.t.

$$x_i = \begin{cases} x_{i-1} & \text{, w.p } p \\ \text{Unif}[\mathcal{X} \backslash x_{i-1}] & \text{, w.p } 1-p \end{cases}$$

  - At the beginning: assume decoder knows s

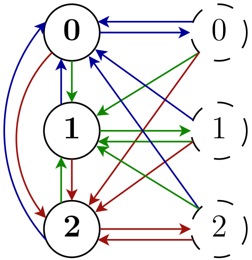


Coding scheme

- **Pre-transmission:** generate an information sequence s.t.

$$x_i = \begin{cases} x_{i-1} & \text{, w.p } p \\ \text{Unif}[\mathcal{X} \backslash x_{i-1}] & \text{, w.p } 1-p \end{cases}$$

  - At the beginning: assume decoder knows s
  - **Encoder:** Transmit x, if $y = s$, repeat x

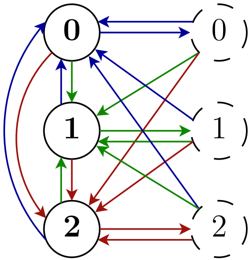


Coding scheme

- **Pre-transmission:** generate an information sequence s.t.

$$x_i = \begin{cases} x_{i-1} & \text{, w.p } p \\ \text{Unif}[\mathcal{X} \backslash x_{i-1}] & \text{, w.p } 1-p \end{cases}$$

  - At the beginning: assume decoder knows s
  - **Encoder:** Transmit x, if $y = s$, repeat x
  - **Decoder:** Receive y, if $y \neq s$ store y, else store next output

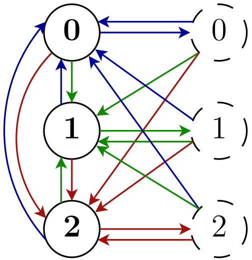


Coding scheme

- **Pre-transmission:** generate an information sequence s.t.

$$x_i = \begin{cases} x_{i-1} & \text{, w.p } p \\ \text{Unif}[\mathcal{X} \backslash x_{i-1}] & \text{, w.p } 1-p \end{cases}$$

  - At the beginning: assume decoder knows s
  - **Encoder:** Transmit x, if $y = s$, repeat x
  - **Decoder:** Receive y, if $y \neq s$ store y, else store next output



- **The rate of the scheme:**

$$R(\mathcal{X}) = \max_{p \in [0,1]} 2 \frac{H_2(p) + (1-p)\log(|\mathcal{X}|-1)}{p+3}$$

Upper-bound

Theorem (Sabag-P-Pfister'17)

For any choice of Q-graph

$$C_{fb} \leq \max_{p(x|s,q) \in \mathcal{P}_\pi} I(X,S;Y|Q)$$

Upper-bound

Theorem (Sabag-P-Pfister'17)

For any choice of Q-graph

$$C_{fb} \leq \max_{p(x|s,q)\in\mathcal{P}_\pi} I(X,S;Y|Q)$$

Theorem (Duality bound)

For any FSC channel and $T_{Y|Q}$

$$C_{fb} \leq \lim_{n\to\infty} \max_{\max_{x^n}} \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[D\left(P_{Y|X=X_i,X^-=X_{i-1}} \| T_{Y|Q=Q_{i-1}}\right)\right]$$

The feedback capacity

Theorem

For all $|\mathcal{X}| \leq 8$, the feedback capacity of the Ising channel is given by

$$C_{fb}(\mathcal{X}) = \max_{p\in[0,1]} 2\frac{H_2(p) + (1-p)\log(|\mathcal{X}|-1)}{p+3}$$

The feedback capacity

Theorem

For all $|\mathcal{X}| \leq 8$, the feedback capacity of the Ising channel is given by

$$C_{fb}(\mathcal{X}) = \max_{p\in[0,1]} 2\frac{H_2(p) + (1-p)\log(|\mathcal{X}|-1)}{p+3}$$

- What happens for $|\mathcal{X}| \geq 9$?

The feedback capacity

Theorem

For all $|\mathcal{X}| \leq 8$, the feedback capacity of the Ising channel is given by

$$C_{fb}\left(\mathcal{X}\right) = \max_{p\in[0,1]} 2\frac{H_2(p) + (1-p)\log\left(|\mathcal{X}|-1\right)}{p+3}$$

- What happens for $|\mathcal{X}| \geq 9$?
- Asymptotic better rate

$$R\left(\mathcal{X}\right) \propto \frac{3}{4}\log\frac{|\mathcal{X}|}{2}$$

The DDPG Algorithm

Deep Deterministic Policy Gradient, (Lillicrap et al.'16)

- Draw N interactions from experience (z_i, u_i, r_i, z_i')
- Train critic: minimize by ω

$$\frac{1}{N}\sum_{i=1}^{N}\left[Q_\omega\left(z_i,u_i\right)-\left[r_i-\rho_\mu+\qquad Q_\omega\left(z_i',\pi_\mu(z_i')\right)\right]\right]^2$$

- Improve actor: maximize by μ

$$\frac{1}{N}\sum_{i=1}^{N}\nabla_u Q_\omega\left(z_i,u\right)|_{u=\pi_\mu(z_i)}\nabla_\mu\pi_\mu\left(z_i\right)$$

The DDPG Algorithm with planing

Deep Deterministic Policy Gradient, (Lillicrap et al.'16)

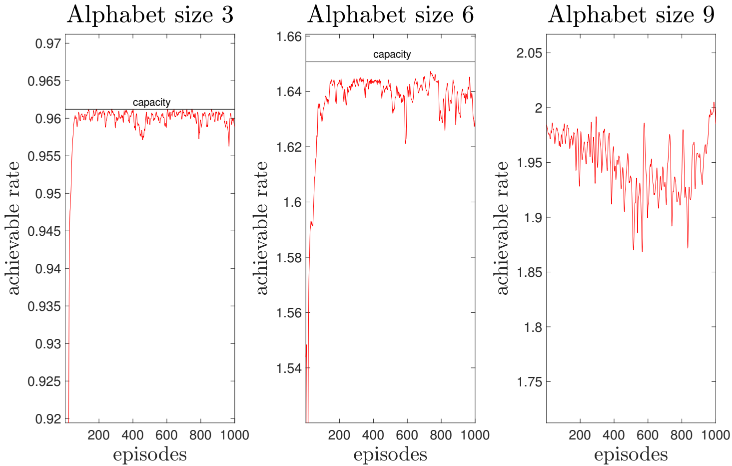
- Draw N interactions from experience (z_i, u_i, r_i, z_i')
- Train critic: minimize by ω

$$\frac{1}{N}\sum_{i=1}^{N}\left[Q_\omega\left(z_i,u_i\right)-\left[r_i-\rho_\mu+\sum_{z'}p(z'|z_i,u_i)Q_\omega\left(z',\pi_\mu(z')\right)\right]\right]^2$$

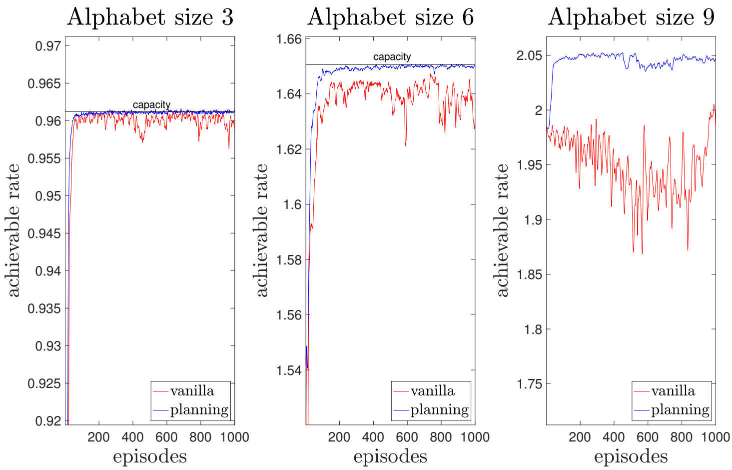
- Improve actor: maximize by μ

$$\frac{1}{N}\sum_{i=1}^{N}\nabla_u Q_\omega\left(z_i,u\right)|_{u=\pi_\mu(z_i)}\nabla_\mu \pi_\mu\left(z_i\right)$$

Improving RL: DDPG without planning



Improving RL: DDPG with planing



Conclusions

- The idea of graph-based auxilary r.v.
- RL methodology for computing feedback capacity
- Problem setting for improving RL

Conclusions

- The idea of graph-based auxilary r.v.
- RL methodology for computing feedback capacity
- Problem setting for improving RL

Future Work:

- What is the solution for $|\mathcal{X}| \geq 9$?

Conclusions

- The idea of graph-based auxilary r.v.
- RL methodology for computing feedback capacity
- Problem setting for improving RL

Future Work:

- What is the solution for $|\mathcal{X}| \geq 9$?
- Use insights of IT to improve RL.

Conclusions

- The idea of graph-based auxilary r.v.
- RL methodology for computing feedback capacity
- Problem setting for improving RL

Future Work:

- What is the solution for $|\mathcal{X}| \geq 9$?
- Use insights of IT to improve RL.
- Graph-based auxiliary r.v. beyond feedback.

Conclusions

- The idea of graph-based auxilary r.v.
- RL methodology for computing feedback capacity
- Problem setting for improving RL

Future Work:

- What is the solution for $|\mathcal{X}| \geq 9$?
- Use insights of IT to improve RL.
- Graph-based auxiliary r.v. beyond feedback.
- Cardinality bound on Q.

Conclusions

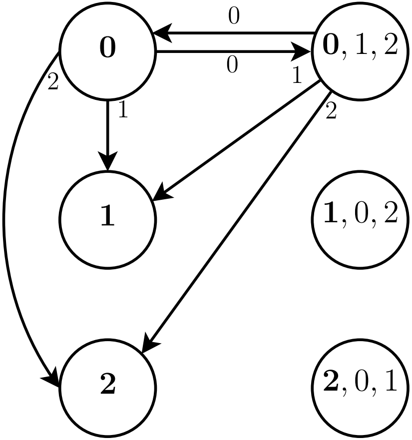
- The idea of graph-based auxilary r.v.
- RL methodology for computing feedback capacity
- Problem setting for improving RL

Future Work:

- What is the solution for $|\mathcal{X}| \geq 9$?
- Use insights of IT to improve RL.
- Graph-based auxiliary r.v. beyond feedback.
- Cardinality bound on Q.

Thank You!

Transitions of states by a directed graph



The DDPG Algorithm

Deep Deterministic Policy Gradient, (Lillicrap et al.'16)

- Draw N interactions from experience (z_i, u_i, r_i, z_i')
- Train critic: minimize by ω

$$\frac{1}{N}\sum_{i=1}^{N}\left[Q_\omega\left(z_i, u_i\right) - \left[r_i - \rho_\mu + \sum_{z'} p(z'|z_i, u_i) Q_\omega\left(z', \pi_\mu(z')\right)\right]\right]$$

- Improve actor: maximize by μ

$$\frac{1}{N}\sum_{i=1}^{N}\nabla_u Q_\omega\left(z_i, u\right)|_{u=\pi_\mu(z_i)}\nabla_\mu \pi_\mu\left(z_i\right)$$