Analytic Energy-Guided Policy Optimization for Offline Reinforcement Learning

Jifeng Hu, Sili Huang, Zhejian Yang, Shengchao Hu,
Li Shen, Hechang Chen, Lichao Sun, Yi Chang, Dacheng Tao

Jifeng Hu
2025

School of Artificial Intelligence, Jilin University, Changchun, China

1. Background

Conditional Diffusion Model

Classifier-guided Diffusion Model

$$p(x_{t-1} \mid x_t, \mathcal{O}) \propto q(x_{t-1} \mid x_t) p(\mathcal{O} \mid x_t)$$

$$p(x_{t-1} \mid x_t, \mathcal{O}) = \mathcal{N}(x_t; \mu_t + \Sigma_t \cdot \nabla \log p(\mathcal{O} \mid x_t), \Sigma_t)$$

Class-free Diffusion Model

$$\mathbb{E}_{x_0 \sim q(x_0), t \sim U(0,T), b \sim \mathcal{B}(\lambda)} \left[\| \epsilon_\theta(x_t, b * \mathcal{C}, t) - \epsilon \|_2^2 \right]$$

$$\hat{\epsilon} = \epsilon_\theta(x_t, t, \varnothing) + \omega(\epsilon_\theta(x_t, t, \mathcal{C}) - \epsilon_\theta(x_t, t, \varnothing))$$

Usually, classifier guidance and classifier-free guidance need paired data, i.e., samples and the corresponding conditioning variables, to train a controllable diffusion model. However, it is difficult to describe the conditioning variables for each transition in RL. We can only evaluate the value for the transitions with a scalar and continuous function.

Score-Based Generative Modeling through Stochastic Differential Equations

1. Relation of RL and energy guided diffusion model

	Energy-guided diffusion model	Constrained RL policy optimization
Problem definition	$\min_p \quad \mathbb{E}_{x\sim p(x)}\mathcal{E}(x),$ $s.t. \quad D_{KL}(p(x)\,\|\,q(x)) < \epsilon,$	$\max_\pi \quad \mathbb{E}_{s\sim D^\mu}\left[\mathbb{E}_{a\sim\pi(\cdot\|s)}Q(s,a)\right],$ $s.t. \quad D_{KL}(\pi(\cdot\,\|\,s)\,\|\,\mu(\cdot\,\|\,s)),$
Optimal solution	$p(x) \propto q(x)e^{-\beta\mathcal{E}(x)}$	$\pi^*(a\,\|\,s) \propto \mu(a\,\|\,s)e^{\beta Q(s,a)}$

1. Background

Action-value function guided diffusion policy = Unconditional diffusion policy (obtained through imitation learning) + Guidance term (a function of Q)

$$\nabla_{a_t} \log \pi_t(a_t \mid s) = \nabla_{a_t} \log \mu_t(a_t \mid s) + \nabla_{a_t} \mathcal{E}_t(s, a_t)$$

Since the diffusion model has multiple steps, the above relationship is only satisfied in the first step. In order to make the above relationship hold for any step

(Intermediate energy) $$\mathcal{E}_t(s, a_t) = \begin{cases} \beta Q(s, a_0), & t = 0 \\ \log \mathbb{E}_{\mu_{0|t}(a_0|a_t, s)}[e^{\beta Q(s, a_0)}], & t > 0 \end{cases}$$

2. Method: imitation+guidance

Obviously, the optimal distribution (guided distribution) can be formed by combining two terms. If we want to sample from the optimal distribution, it is only necessary to compute the two terms separately

$$\nabla_{a_t} \log \pi_t(a_t \mid s) = \nabla_{a_t} \log \mu_t(a_t \mid s) + \nabla_{a_t} \mathcal{E}_t(s, a_t)$$

Unconditional diffusion model

$$\mathcal{L}_{diff} = \mathbb{E}_{x_0 \sim q(x_0), t \sim U(0,T)} \left[\| \epsilon_\theta(x_t, t) - \epsilon \|_2^2 \right]$$

$$\nabla_{a_t} \log \mu_t(a_t \mid s) = -\frac{\epsilon_\theta(s, a_t, t)}{\sigma_t}$$

Intermediate guidance

The gradient of intermediate energy

$$\mathcal{E}_t(s, a_t) = \log \mathbb{E}_{\mu_{0|t}(a_0 \mid a_t, s)} [e^{\beta Q(s, a_0)}], \mathrm{t} > 0$$

Intractable term due to the log expectation

2. Method: dependence conversion

- 1) How to obtain the solution of the intermediate guidance $\nabla_{a_t}\mathcal{E}_t(s,a_t)$

  - Convert the implicit dependence on the action in the exponential term to explicit dependence

$$Q(s,a_0) \approx Q(s,a_0)|_{a_0=\bar{a}} + \frac{\partial Q(s,a_0)}{\partial a_0}^{\top}|_{a_0=\bar{a}} *(a_0-\bar{a})$$

  - Obtain the further results

$$\log \mathbb{E}_{a_0\sim\mu(a_0|a_t,s)} e^{\beta Q(s,a_0)} \approx \beta Q(s,\bar{a}) - \beta Q'(s,\bar{a})^{\top}\bar{a} + \log\left\{\mathbb{E}_{a_0\sim\mu(a_0|a_t,s)}[e^{\beta Q'(s,\bar{a})^{\top}a_0}]\right\}$$

The only unknow term is the third term

$$\log\left\{\mathbb{E}_{a_0\sim\mu(a_0|a_t,s)}[e^{\beta Q'(s,\bar{a})^{\top}a_0}]\right\}$$

2. Method: moment generating function

- 2) How to obtain the solution of $\log\left\{\mathbb{E}_{a_0\sim\mu(a_0|a_t,s)}[e^{\beta Q'(s,\bar{a})^\top a_0}]\right\}$

  - Suppose that $\mu_{0|t}=\mathcal{N}(\tilde{\mu}_{0|t},\tilde{\Sigma}_{0|t})$

  - According to the properties of moment generating function

$$\mathbb{E}_{x\sim\mathcal{N}(v,\Sigma)}[e^{a^\top x}]=e^{a^\top v+\frac{1}{2}a^\top\Sigma a}$$

  - After derivation we get

$$\log\mathbb{E}_{a_0\sim\mu(a_0|a_t,s)}e^{\beta Q(s,a_0)}\approx\beta Q(s,\bar{a})+\beta Q'(s,\bar{a})^\top(\tilde{\mu}_{0|t}-\bar{a})+\frac{1}{2}\beta^2 Q'(s,\bar{a})^\top\tilde{\Sigma}_{0|t}Q'(s,\bar{a})$$

  - The intermediate energy can be represented by

$$\log\mathbb{E}_{a_0\sim\mu(a_0|a_t,s)}e^{\beta Q(s,a_0)}\approx\beta Q(s,\bar{a})+\beta Q'(s,\bar{a})^\top(\tilde{\mu}_{0|t}-\bar{a})+\frac{1}{2}\beta^2 Q'(s,\bar{a})^\top\tilde{\Sigma}_{0|t}Q'(s,\bar{a})$$

2. Method: posterior approximation

- 3) How to obtain the approximation of $\tilde{\mu}_{0|t}, \tilde{\Sigma}_{0|t}$

- Obviously, the mean vector satisfy

$$\tilde{\mu}_{0|t} = \frac{1}{\alpha_t}(a_t - \sigma_t \epsilon_\theta(s, a_t, t))$$

- The covariance matrix can be approximated through two methods

$$\tilde{\Sigma}_{0|t}(a_t) = \mathbb{E}_{\mu_{0|t}(a_0|a_t,s)}\left[(a_0 - \tilde{\mu}_{0|t})(a_0 - \tilde{\mu}_{0|t})^\top\right]$$

$$\tilde{\Sigma}_{0|t}(a_t) = \frac{1}{\alpha_t^2}\mathbb{E}_{\mu_{0|t}(a_0|a_t,s)}\left[(a_t - \alpha_t a_0)(a_t - \alpha_t a_0)^\top\right] - \frac{\sigma_t^2}{\alpha_t^2}\epsilon_\theta \epsilon_\theta^\top$$

$$\tilde{\Sigma}_{0|t}(a_t) = \mathbb{E}_{\mu_{0|t}(a_0|a_t,s)}\left[(a_0 - u_0)(a_0 - u_0)^\top\right] - (\tilde{\mu}_{0|t} - u_0)(\tilde{\mu}_{0|t} - u_0)^\top$$

2. Method: posterior approximation

- To simplify the problem of calculating the exact covariance matrix, we adopt two strategies:

✓ The isotropic Gaussian assumption $\tilde{\Sigma}_{0|t} = \tilde{\sigma}_{0|t}^2 * I$

✓ The marginalization over a_t $\tilde{\Sigma}_{0|t} = \mathbb{E}_{\mu_t(a_t|s)} \tilde{\Sigma}_{0|t}(a_t)$

- The approximated covariance matrix can be simplified to

$$\tilde{\sigma}_{0|t}^2 = \frac{\sigma_t^2}{\alpha_t^2}\left[1 - \frac{1}{d}\mathbb{E}_{\mu_t(a_t|s)}\left[\| \epsilon_\theta(a_t, t) \|_2^2\right]\right]$$

$$\tilde{\sigma}_{0|t}^2 = Var(a_0) - \frac{1}{d}\mathbb{E}_{\mu_t(a_t|s)}[\| \tilde{\mu}_{0|t} - u_0 \|_2^2]$$

- The intermediate energy can be represented by

$$\mathcal{E}_t(s, a_t) \approx \beta Q(s, \bar{a}) + \beta Q'(s, \bar{a})^\top (\tilde{\mu}_{0|t} - \bar{a}) + \frac{1}{2}\beta^2 \tilde{\sigma}_{0|t}^2 * \| Q'(s, \bar{a}) \|_2^2$$

2. Method: neural network estimation

- 4) How to obtain the gradient of intermediate energy $\nabla_{a_t}\mathcal{E}_t(s,a_t)$
  - First train a neural network for intermediate energy estimation

$$\mathcal{L}_{IE}=\mathbb{E}\left[\|\mathcal{E}_\Theta(s,a_t,t)-\mathcal{E}_t(s,a_t)\|_2^2\right]$$

  - Then use the well-trained intermediate energy to obtain the gradient

$$\nabla_{a_t}\mathcal{E}_t(s,a_t)\approx\nabla_{a_t}\mathcal{E}_\Theta(s,a_t,t)$$

  - Guidance rescale: make inference more stable

$$\nabla_{a_t}\log\pi_t(a_t\mid s)=\frac{\nabla_{a_t}\log\pi_t(a_t\mid s)}{\|\nabla_{a_t}\log\pi_t(a_t\mid s)\|}*\|\nabla_{a_t}\log\mu_t(a_t\mid s)\|$$

  - The Q function can be obtained through many strategies, such as IQL

$$\mathcal{L}_V=\mathbb{E}_{(s,a)\sim D_\mu}\left[L_2^\tau(V_\phi(s)-Q_{\bar{\psi}}(s,a))\right]$$

$$\mathcal{L}_Q=\mathbb{E}_{(s,a,s')\sim D_\mu}\left[\|r(s,a)+\gamma V_\phi(s')-Q_\psi(s,a)\|_2^2\right]$$

$$L_2^\tau(y)=|\tau-1(y<0)|\,y^2$$

3. Experiments

Experiments:

Gym-MuJoCo (9 tasks), Pointmaze (6 tasks), Locomotion (6 tasks), and Adroit (12 tasks)

Evaluation Metrics:

$$E_{norm} = \frac{E - E_{random}}{E_{expert} - E_{random}} * 100$$

Baselines:

- Diffusion-based methods: DiffuserLite, HD-DA, IDQL, AdaptDiffuser, Consistency-AC, HDMI, TCD, QGPO, SfBC, DD, Diffuser, and D-QL, etc.
- Traditional RL methods: AWAC, and BC, etc.
- Model-based methods: TAP, MOReL, MOPO, and MBOP, etc.
- Constraint-based methods: CQL, BCQ, BEAR, etc.
- Uncertainty-based methods: PBRL, TD3+BC, and IQL, etc.
- Transformer-based methods: BooT, TT, and DT, etc.

3. Experiments: main results

Dataset	Med-Expert			Medium			Med-Replay			mean score	total score
Env	HalfCheetah	Hopper	Walker2d	HalfCheetah	Hopper	Walker2d	HalfCheetah	Hopper	Walker2d		
AWAC	42.8	55.8	74.5	43.5	57.0	72.4	40.5	37.2	27.0	50.1	450.7
BC	55.2	52.5	107.5	42.6	52.9	75.3	36.6	18.1	26.0	51.9	466.7
MOPO	63.3	23.7	44.6	42.3	28.0	17.8	53.1	67.5	39.0	42.1	379.3
MBOP	105.9	55.1	70.2	44.6	48.8	41.0	42.3	12.4	9.7	47.8	430.0
MOReL	53.3	108.7	95.6	42.1	95.4	77.8	40.2	93.6	49.8	72.9	656.5
TAP	91.8	105.5	107.4	45.0	63.4	64.9	40.8	87.3	66.8	74.8	672.9
BEAR	51.7	4.0	26.0	38.6	47.6	33.2	36.2	10.8	25.3	30.4	273.4
BCQ	64.7	100.9	57.5	40.7	54.5	53.1	38.2	33.1	15.0	50.9	457.7
CQL	62.4	98.7	111.0	44.4	58.0	79.2	46.2	48.6	26.7	63.9	575.2
TD3+BC	90.7	98.0	110.1	48.3	59.3	83.7	44.6	60.9	81.8	75.3	677.4
IQL	86.7	91.5	109.6	47.4	66.3	78.3	44.2	94.7	73.9	77.0	692.6
PBRL	92.3	110.8	110.1	57.9	75.3	89.6	45.1	100.6	77.7	84.4	759.4
DT	90.7	98.0	110.1	42.6	67.6	74.0	36.6	82.7	66.6	74.3	668.9
TT	95.0	110.0	101.9	46.9	61.1	79.0	41.9	91.5	82.6	78.9	709.9
BooT	94.0	102.3	110.4	50.6	70.2	82.9	46.5	92.9	87.6	81.9	737.4
SfBC	92.6	108.6	109.8	45.9	57.1	77.9	37.1	86.2	65.1	75.6	680.3
D-QL@1	94.8	100.6	108.9	47.8	64.1	82.0	44.0	63.1	75.4	75.6	680.7
Diffuser	88.9	103.3	106.9	42.8	74.3	79.6	37.7	93.6	70.6	77.5	697.7
DD	90.6	111.8	108.8	49.1	79.3	82.5	39.3	100.0	75.0	81.8	736.4
IDQL	95.9	108.6	112.7	51.0	65.4	82.5	45.9	92.1	85.1	82.1	739.2
HDMI	92.1	113.5	107.9	48.0	76.4	79.9	44.9	99.6	80.7	82.6	743.0
AdaptDiffuser	89.6	111.6	108.2	44.2	96.6	84.4	38.3	92.2	84.7	83.3	749.8
DiffuserLite	87.8	110.7	106.5	47.6	99.1	85.9	41.4	95.9	84.3	84.4	759.2
HD-DA	92.5	115.3	107.1	46.7	99.3	84.0	38.1	94.7	84.1	84.6	761.8
Consistency-AC	84.3	100.4	110.4	69.1	80.7	83.1	58.7	99.7	79.5	85.1	765.9
TCD	92.7	112.6	111.3	47.2	99.4	82.1	40.6	97.2	88.0	85.7	771.0
D-QL	96.1	110.7	109.7	50.6	82.4	85.1	47.5	100.7	94.3	86.3	777.1
QGPO	93.5	108.0	110.7	54.1	98.0	86.0	47.6	96.9	84.4	86.6	779.2
AEPO	$94.4_{\pm 0.9}$	$111.5_{\pm 1.1}$	$109.3_{\pm 0.5}$	$49.6_{\pm 1.1}$	$100.2_{\pm 0.5}$	$86.2_{\pm 1.1}$	$43.7_{\pm 1.3}$	$101.0_{\pm 0.9}$	$90.8_{\pm 1.5}$	87.4	786.7

3. Experiments: main results

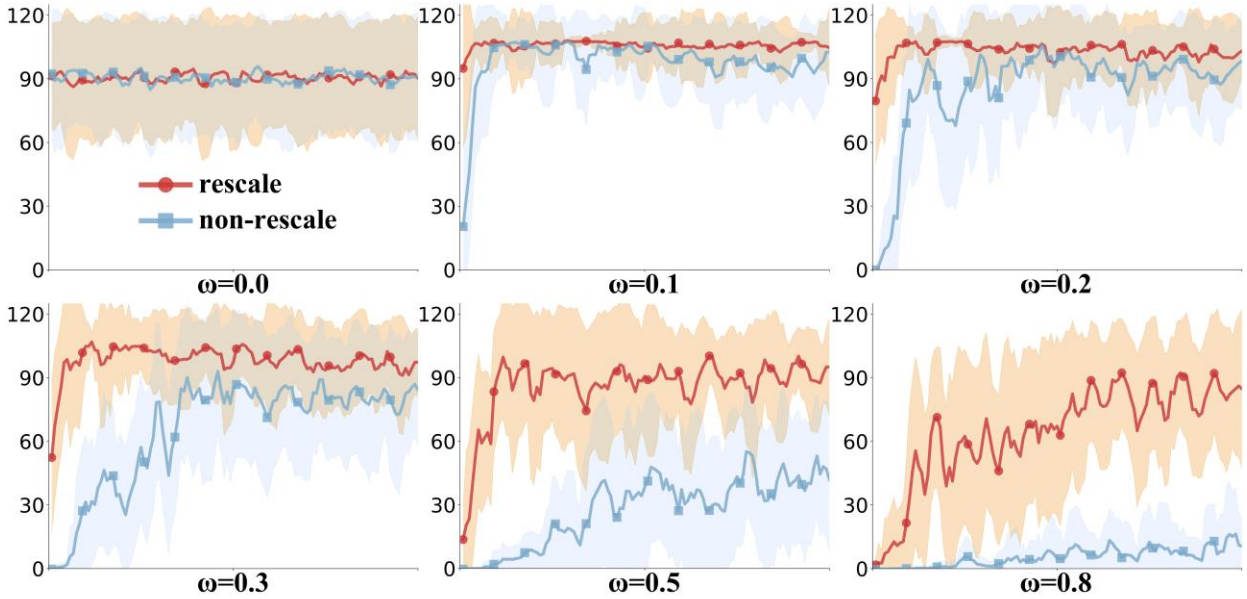
Environment	maze2d-umaze		maze2d-medium		maze2d-large		mean sparse score	mean dense score
Environment type	sparse	dense	sparse	dense	sparse	dense		
DT	31.0	-	8.2	-	2.3	-	13.8	-
BCQ	49.1	-	17.1	-	30.8	-	32.3	-
QDT	57.3	-	13.3	-	31.0	-	33.9	-
IQL	42.1	-	34.9	-	61.7	-	46.2	-
COMBO	76.4	-	68.5	-	14.1	-	53.0	-
TD3+BC	14.8	-	62.1	-	88.6	-	55.2	-
BEAR	65.7	-	25.0	-	81.0	-	57.2	-
BC	88.9	14.6	38.3	16.3	1.5	17.1	42.9	16.0
CQL	94.7	37.1	41.8	32.1	49.6	29.6	62.0	32.9
TT	68.7	46.6	34.9	52.7	27.6	56.6	43.7	52.0
SfBC	73.9	-	73.8	-	74.4	-	74.0	-
SynthER	99.1	-	66.4	-	143.3	-	102.9	-
Diffuser	113.9	-	121.5	-	123.0	-	119.5	-
HDMI	120.1	-	121.8	-	128.6	-	123.5	-
HD-DA	72.8	45.5	42.1	54.7	80.7	45.7	65.2	48.6
TCD	128.1	29.8	132.9	41.4	146.4	75.5	135.8	48.9
DD	116.2	83.2	122.3	78.2	125.9	23.0	121.5	61.5
AEPO	136.0	107.2	128.4	109.9	132.4	165.5	132.3	127.5

Environment	antmaze-umaze		antmaze-medium		antmaze-large		mean score	total score
Environment type		diverse	play	diverse	play	diverse		
AWAC	56.7	49.3	0.0	0.7	0.0	1.0	18.0	107.7
DT	59.2	53.0	0.0	0.0	0.0	0.0	18.7	112.2
BC	65.0	55.0	0.0	0.0	0.0	0.0	20.0	120.0
BEAR	73.0	61.0	0.0	8.0	0.0	0.0	23.7	142.0
BCQ	78.9	55.0	0.0	0.0	6.7	2.2	23.8	142.8
TD3+BC	78.6	71.4	10.6	3.0	0.2	0.0	27.3	163.8
CQL	74.0	84.0	61.2	53.7	15.8	14.9	50.6	303.6
IQL	87.5	62.2	71.2	70.0	39.6	47.5	63.0	378.0
QDQ	98.6	67.8	81.5	85.4	35.6	31.2	66.7	466.8
DD	73.1	49.2	0.0	24.6	0.0	7.5	25.7	154.4
D-QL	93.4	66.2	76.6	78.6	46.4	56.6	69.6	417.8
IDQL	93.8	62.0	86.6	83.5	57.0	56.4	73.2	439.3
SfBC	92.0	85.3	81.3	82.0	59.3	45.5	74.2	445.4
QGPO	96.4	74.4	83.6	83.8	66.6	64.8	78.3	469.6
AEPO	100.0	100.0	76.7	83.3	56.7	66.7	80.6	483.4

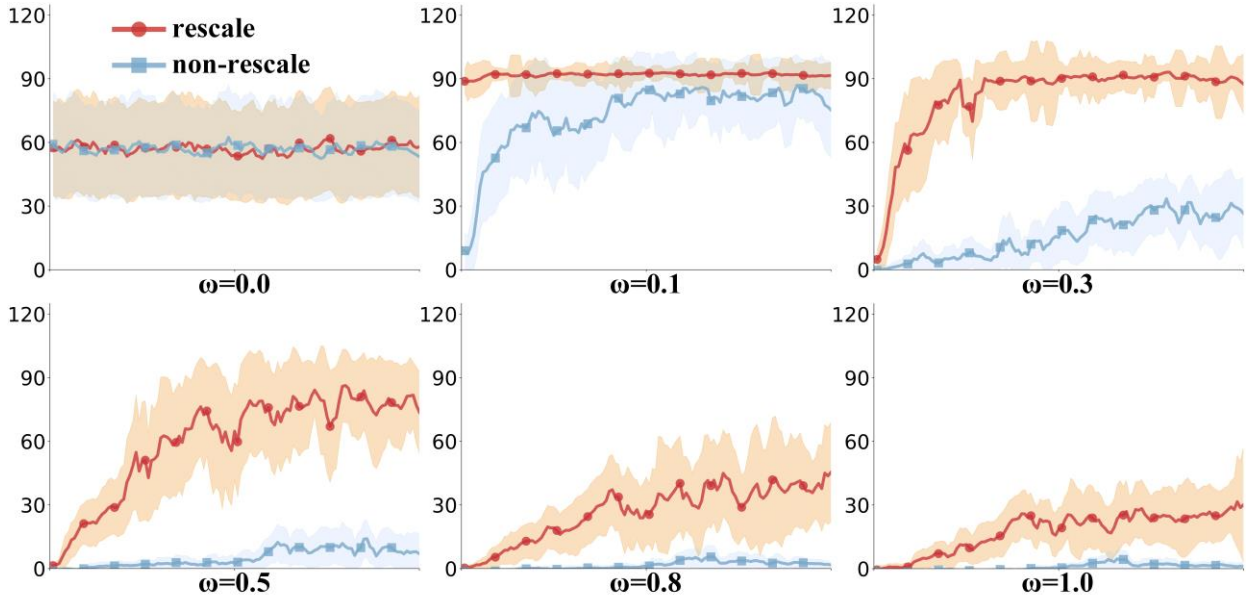
3. Experiments: main results

Task	pen			hammer			door			relocate			mean score	total score
Dataset	human	expert	cloned	human	expert	cloned	human	expert	cloned	human	expert	cloned		
BC	7.5	69.7	6.6	-	-	-	-	-	-	0.1	57.1	0.1	-	-
BEAR	-1.0	-	26.5	-	-	-	-	-	-	-	-	-	-	-
BCQ	68.9	-	44.0	-	-	-	-	-	-	-	-	-	-	-
IQL	71.5	-	37.3	1.4	-	2.1	4.3	-	1.6	0.1	-	-0.2	-	-
TT	36.4	72.0	11.4	0.8	15.5	0.5	0.1	94.1	-0.1	0.0	10.3	-0.1	20.1	240.9
CQL	37.5	107.0	39.2	4.4	86.7	2.1	9.9	101.5	0.4	0.2	95.0	-0.1	40.3	483.8
UWAC	65.0	119.8	45.1	8.3	128.8	1.2	10.7	105.4	1.2	0.5	108.7	0.0	49.6	594.7
TAP	76.5	127.4	57.4	1.4	127.6	1.2	8.8	104.8	11.7	0.2	105.8	-0.2	51.9	622.6
TCD	49.9	35.6	73.3	-	-	-	-	-	-	0.4	59.6	0.2	-	-
HD-DA	-2.6	107.9	-2.7	-	-	-	-	-	-	0.0	-0.1	-0.2	-	-
DiffuserLite	33.2	20.7	2.1	-	-	-	-	-	-	0.1	0.1	-0.2	-	-
DD	64.1	107.6	47.7	1.0	106.7	0.9	6.9	87.0	9.0	0.2	87.5	-0.2	43.2	518.4
HDMI	66.2	109.5	48.3	1.2	111.8	1.0	7.1	85.9	9.3	0.1	91.3	-0.1	44.3	531.6
D-QL@1	66.0	112.6	49.3	1.3	114.8	1.1	8.0	93.7	10.6	0.2	95.2	-0.2	46.1	552.6
QGPO	73.9	119.1	54.2	1.4	123.2	1.1	8.5	98.8	11.2	0.2	102.5	-0.2	49.5	593.9
LD	79.0	131.2	60.7	4.6	132.5	4.2	9.8	111.9	12.0	0.2	109.5	-0.1	54.6	655.5
AEPO	76.7	147.0	69.3	10.3	129.7	6.4	9.0	106.5	3.9	0.8	107.0	0.6	55.6	667.5

3. Experiments: rescale ablation

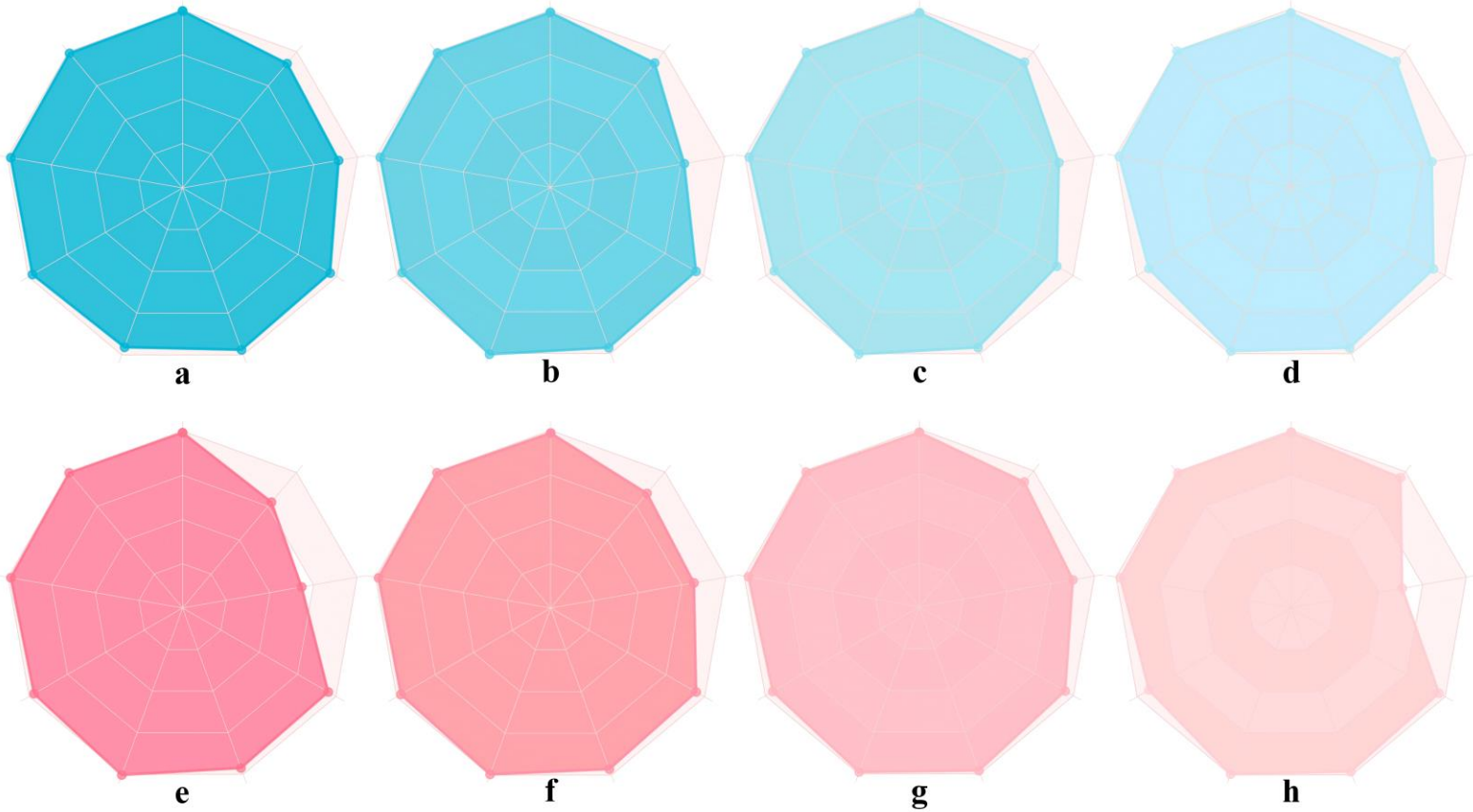


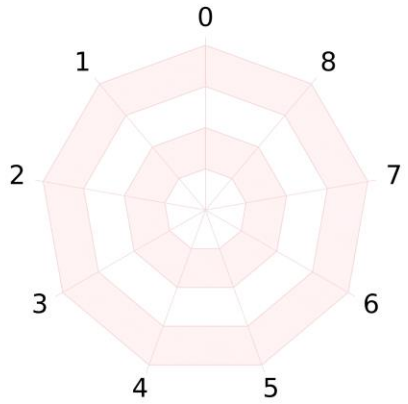
Gym-MuJoCo walker2d-medium-expert task



Gym-MuJoCo halfcheetah-medium-expert task

3. Experiments: parameter sensitivity





a: β=1;τ=0.5
b: β=2;τ=0.5
c: β=4;τ=0.5
d: β=5;τ=0.5
e: β=3;τ=0.8
f: β=3;τ=0.7
g: β=3;τ=0.6
h: β=3;τ=0.5

0: halfcheetah-medium-expert
1: hopper-medium-expert
2: walker2d-medium-expert
3: halfcheetah-medium
4: hopper-medium
5: walker2d-medium
6: halfcheetah-medium-replay
7: hopper-medium-replay
8: walker2d-medium-replay

4. Conclusion

- we theoretically analyze the intermediate energy with a log-expectation formulation, derive the solution of intermediate energy by addressing the posterior integral, and propose a new diffusion-based method called Analytic Energy-guided Policy Optimization (AEPO).

- We investigate the closed-form solution of intermediate guidance that has intractable log-expectation formulation and provide an effective approximation method under the most widely used Gaussian-based diffusion models.

- we conduct sufficient experiments in 4 types, 30+ tasks by comparing with 30+ baselines to validate the effectiveness.

Thanks

School of Artificial Intelligence

Jifeng Hu