

Flow Matching Policy Optimization with Mirror Descent and Entropy Constraints

Ting Gao, Stavros Orfanoudakis, Nan Lin, Winnie Daamen, Serge Hoogendoorn, Elvin Isufi
Delft University of Technology · T.Gao-1@tudelft.nl

TL;DR

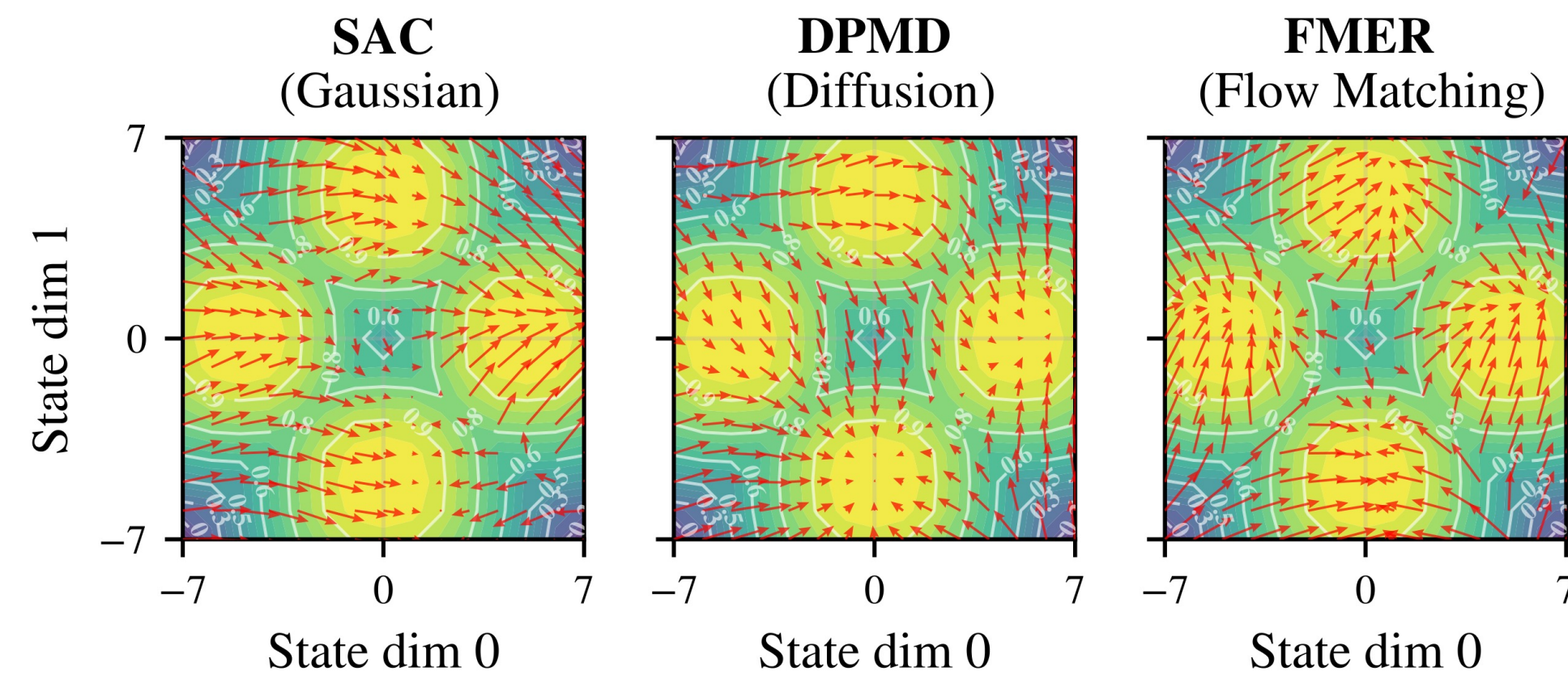
Diffusion (SDE) policies have intractable entropy and unstable gradients. FMER uses **Flow Matching (ODE)** as the policy generator, turns policy mirror descent into a **weighted flow-matching regression** and adds an **exact, tanh-corrected entropy constraint**, giving the best results on sparse-reward FrankaKitchen.



What is the problem?

Good behaviour is often multimodal

Gaussian policies pick one of several equally good actions.



2D multi-goal task (bright = goals, arrows = chosen actions). SAC and DPMD drift to favourite goals; FMER heads for the nearest of four equally good goals.

But diffusion (SDE) policies can hit two walls

✗ **Entropy is intractable**
→ exploration relies on heuristics

✗ **Gradients through the denoising chain**
→ expensive, high-variance, unstable

Key insight: switch to an ODE flow-matching policy

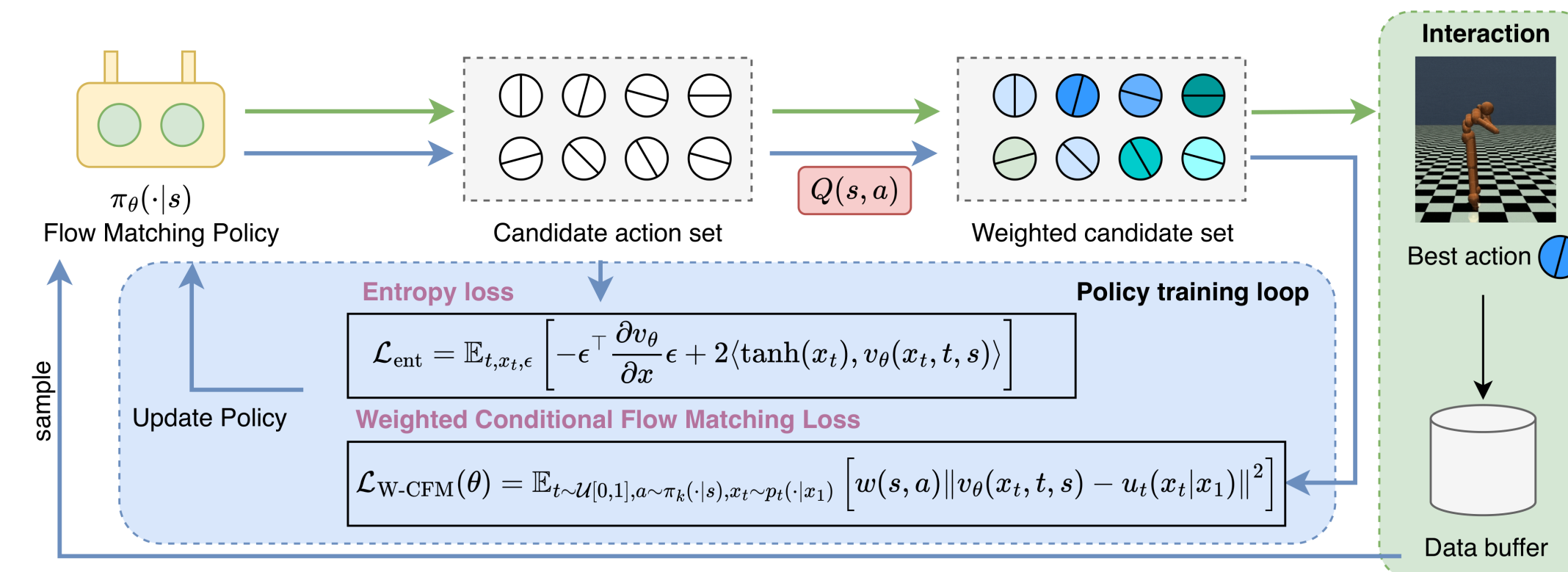
Entropy becomes the divergence of the velocity field, and training becomes plain regression: **no simulation, no backprop through the ODE.**

	Mirror descent	Entropy constraint	Unbiased entropy	Simulation-free
DIPO	✗	✗	✗	✓
DPMD	✓	✗	✗	✓
QVPO	✗	✓	✗	✓
SAC-Flow	✗	✓	✗	✗
FLAME	✗	✓	✗	✗
FMER (ours)	✓	✓	✓	✓

FMER: one simulation-free update

A **policy mirror descent** step under a **minimum-entropy constraint**:

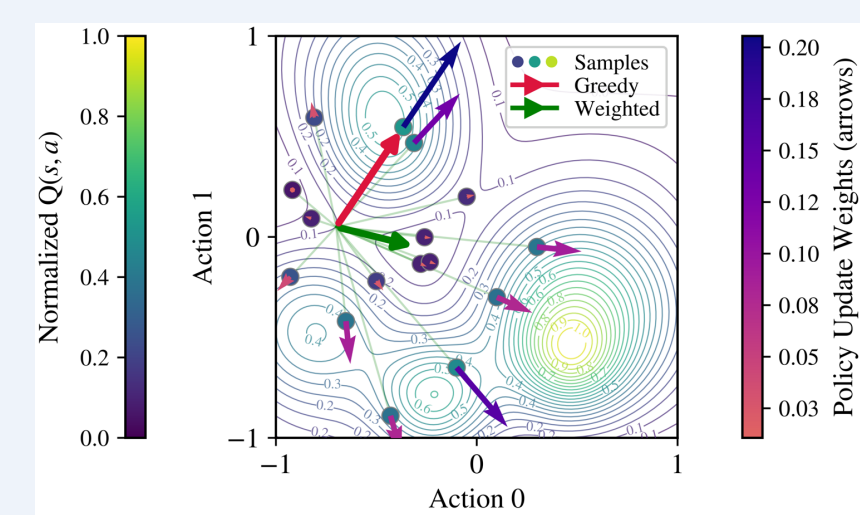
$$\mathcal{L}_{\text{FMER}}(\theta) = \mathcal{L}_{\text{W-CFM}}(\theta) + \alpha \mathcal{L}_{\text{ent}}(\theta)$$



Mirror descent = weighted flow matching

Theorem 1. Minimisers of the weighted CFM loss = mirror-descent target:

$$\mathcal{L}_{\text{W-CFM}} = \mathbb{E}_{t,s,a,x_0} [w(s,a) \|v_\theta(x_t, t, s) - (x_1 - x_0)\|^2]$$



$\Rightarrow \pi_\theta^*(a|s) = \frac{\pi_k(a|s) \exp(A(s,a)/\tau)}{Z(s)}$

- In practice: sample $M = 8$ candidates from π_k ; weight each by softmax of $A(s,a)/\tau$.
- Pure regression: **no backprop through the ODE**
- Soft weights pool all good candidates; greedy top-1 (red) gets stuck

Analytical entropy in a bounded action space

Entropy = integrated divergence of the velocity field; for bounded actions $a = \tanh(x)$, **Prop. 1** adds the exact **tanh correction**:

$$\mathcal{L}_{\text{ent}} = -\mathbb{E}_{t,x_t} [\nabla_x \cdot v_\theta - 2 \langle \tanh(x_t), v_\theta \rangle]$$

Self-tuning trust region

$$\text{ESS}(s) = \frac{1}{\sum_i \hat{w}_i(s)^2} \in [1, M]$$

τ sets the trust-region size; FMER adapts τ to hold Efficient Sample Size (ESS) at a target, so no per-task tuning.

Does it work?

1.7×

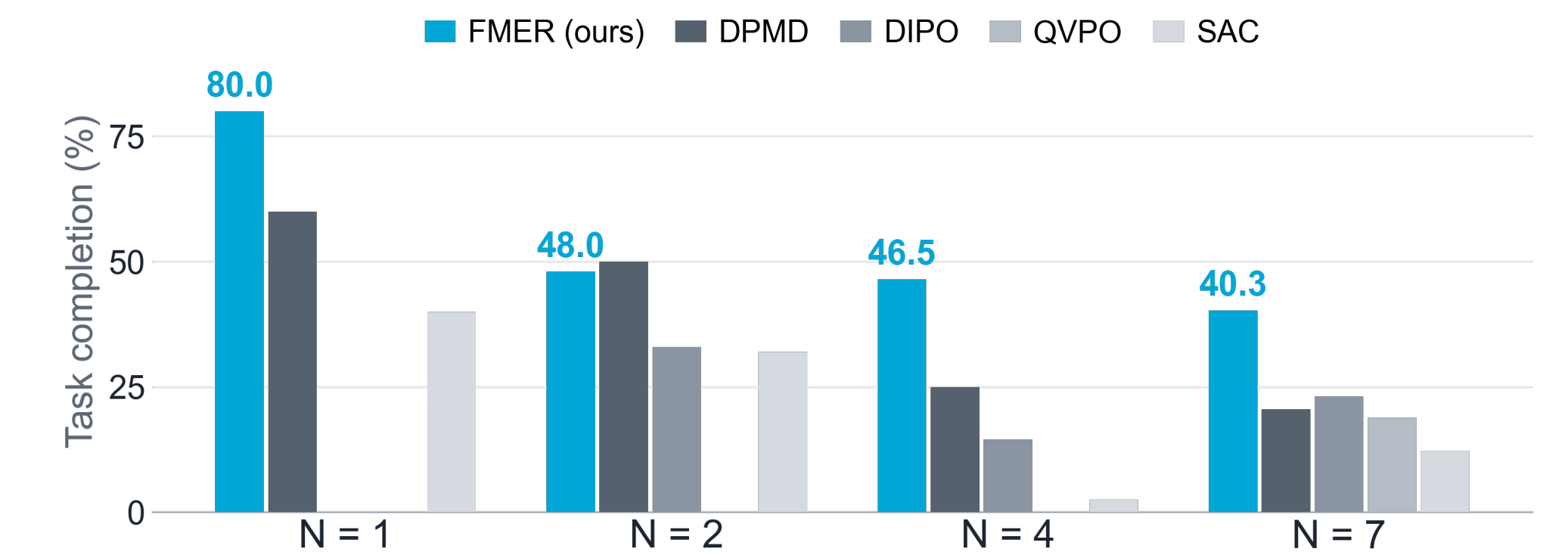
best baseline, FrankaKitchen N = 7
40.3% vs 23.1% (DIPO)

1.9×

best baseline, FrankaKitchen N = 4
46.5% vs 25.0% (DPMD)

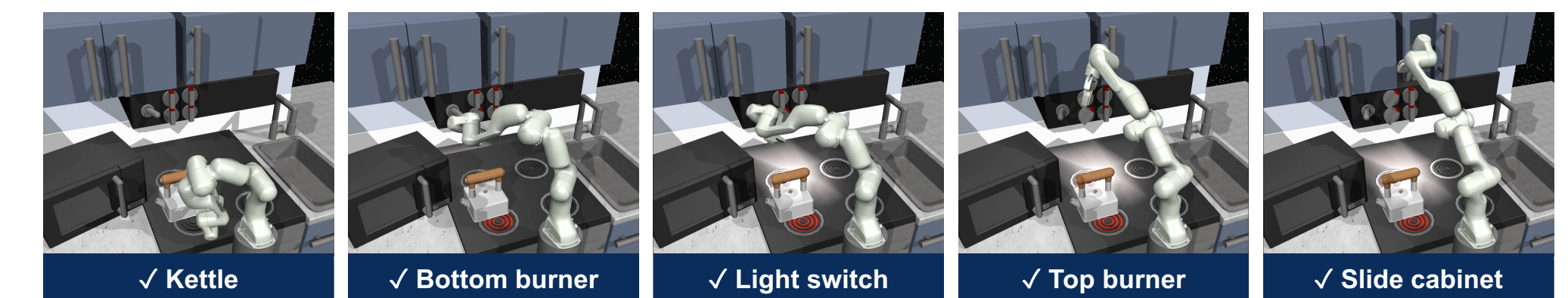
Sparse-reward FrankaKitchen

Reward only on sub-task completion · 280 steps · 1M steps · 5 seeds



Not shown: FLAME, TD3, PPO, SAC-Flow (all $\leq 18.3\%$).

FMER completes **5 tasks** in 280 steps (best baseline DIPO: 3)



Dense-reward MuJoCo

HalfCheetah 12.3k #1 of 9	Walker2d 5.28k #1 of 9	Humanoid 5.29k #2 of 9	Hopper 2.91k #7 of 9
--	-------------------------------------	-------------------------------------	-----------------------------------

Ablation studies (FrankaKitchen N = 7)

