Distilling Policy Distillation [Appendix]

Wojciech M. Czarnecki
DeepMind

Razvan Pascanu
DeepMind

Simon Osindero
DeepMind

Siddhant M. Jayakumar
DeepMind

Grzegorz Świrszcz
DeepMind

Max Jaderberg
DeepMind

A Experimental details

A.1 MDPs

The MDPs used in this study are $W \times H$ grid worlds, meaning that the state space is $\mathcal{S} = \{s_{i,j}\}_{i,j=1}^{W,H} \cup \{s_{term}\}$. s_{term} is a special state, to which an agent is moved with probability 0.01 after each action, ensuring finite length of the experiments considered. There is one initial state placed in the centre of the grid, $\mathcal{S}_1 = \{s_{\lceil W/2 \rceil, \lceil H/2 \rceil}\}$. There are four possible actions $\{L, R, U, D\}$, each of them has an associated desired effect, namely $L(s_{i,j}) = s_{i-1,j}$, $R(s_{i,j}) = s_{i+1,j}$, $D(s_{i,j}) = s_{i,j-1}$, $U(s_{i,j}) = s_{i,j+1}$. Some transitions are invalid, as they would lead to leaving the state space, thus we define

$$z(s_{i,j}, a) = \begin{cases} a(s_{i,j}) & \text{if } a(s_{i,j}) \in \mathcal{S} \\ s_{i,j} & \text{otherwise} \end{cases}$$

The transition dynamics are defined as:

$$T(a, s_{i,j}) = \begin{cases} z(s_{i,j}, a) & \text{with probability } 1-\eta, \\ z(s_{i,j}, L) & \text{with probability } \eta/4, \\ z(s_{i,j}, R) & \text{with probability } \eta/4, \\ z(s_{i,j}, U) & \text{with probability } \eta/4, \\ z(s_{i,j}, D) & \text{with probability } \eta/4, \end{cases}$$

where $\eta = 0.1$ is the transition noise.

Rewards are associated with some states, and are fully deterministic.

Some states are terminal, which cause the episode to end, and bring the agent back to the initial state.

We considered partial, and fully observable versions of these environments. In the fully observable environments, the agent is given the state index as an observation, while in the partially observable environments a concatenated sequence of $(2k+1) \times (2k+1)$ objects, namely $o_k(s_{i,j})$ is represented as

$$\begin{aligned} &(o(s_{i-k,j-k}), ..., o(s_{i-k,j}), ..., o(s_{i-k,j+k}), ..., \\ &\ o(s_{i-k+1,j-k}), ..., o(s_{i-k+1,j}), ..., o(s_{i-k+1,j+k}), ..., \\ &\ ... \\ &\ o(s_{i+k,j-k}), ..., o(s_{i+k,j}), ..., o(s_{i+k,j+k})) \end{aligned}$$

where $o(s)$ is *wall* if $s \notin \mathcal{S}$ and a pair $(\text{reward\_value}(s), \text{is\_terminating}(s))$ otherwise. For example, if the state provides reward 10 and is terminating, then it will be observed as *(10, True)*.

In all partially observable experiments, we use observations which are concatenations of 9×9 squares of vision, centered in an agent position. We experimented with visual extents ranging from 5×5 to full observability and found that this does not effect the qualitative results of the paper, thus the choice of the particular visual extent is not crucial.

A.2 Distribution over MDPs

In all experiments where we sample multiple MDPs we use the following procedure:

1. We create $\mathcal{S}$ as described in the previous section.
2. For each $s_{ij} \in \mathcal{S}$, starting in the upper left corner and traversing first horizontally and then vertically:
   (a) With probability p_w we remove s_{ij} from $\mathcal{S}$, which we call putting a wall in; if we modified a state, we go back to step 2 and continue the loop.
   (b) With probability p_{+10} we put a reward of $+10$ in s_{ij} and make it terminal; if we modified a state, we go back to step 2 and continue the loop.
   (c) With probability p_{+5} we put a reward of $+5$ in s_{ij} and make it terminal; if we modified a

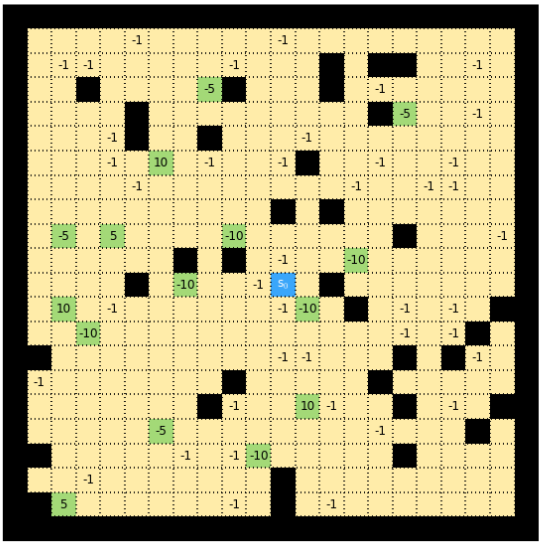

Figure 7: Example 20×20 grid world MDP, with initial state coloured blue, terminal states coloured green, and rewards on various states. Black squares are *wall*s.

state, we go back to step 2 and continue the loop.

(d) With probability p_{-1} we put a reward of -1 in s_{ij}; if we modified a state, we go back to step 2 and continue the loop.

(e) With probability p_{-5} we put a reward of -5 in s_{ij} and make it terminal; if we modified a state, we go back to step 2 and continue the loop.

(f) With probability p_{-10} we put a reward of -10 in s_{ij} and make it terminal; if we modified a state, we go back to step 2 and continue the loop.

3. We check if there exists a path between the initial state and the +10 state, and if this is not true, we repeat the process.

Unless otherwise stated in the text, we use $W = H = 20$, $\frac{1}{10}p_w = p_{+10} = \frac{1}{2}p_{+5} = \frac{1}{10}p_{-1} = p_{-5} = p_{-10} = 0.01$.

A.3 Actor Critic

We use a basic actor critic method, where we sample one full episode under the student policy, $\tau \sim \pi_\theta$, and then update the parameters according to either the single sample Monte Carlo estimated return:

$$\nabla_\theta \log \pi_\theta(a_t|\tau_t)[\sum_t \gamma^{t-1} r_t - V_\theta(s_t)]$$

or, in the TD(1) case, with bootstrapped estimates

$$\nabla_\theta \log \pi_\theta(a_t|\tau_t)[r_t + \gamma V_\theta(s_{t+1}) - V_\theta(s_t)].$$

In all experiments we used $\gamma = 0.99$, but we obtained qualitatively similar results with other values too ($\gamma = 0.95$ and $\gamma = 0.999$).

After each update we use the same Monte Carlo or TD value to fit the baseline function, using the L^2 loss:

$$\nabla_\theta (V_\theta(s_t) - \gamma^{t-1} r_t)^2$$

or

$$\nabla_\theta (V_\theta(s_t) - (r_t + \gamma V_\theta(s_{t+1}))^2$$

in the case of TD learning, where $V_\theta(s_{t+1})$ is treated as a constant. All Vs are initialised to 0s. The learning rate used is 0.1.

A.4 Q-Learning

We use the standard Q-Learning update rule of

$$Q(a_t, s_t) := (1-\lambda)Q(a_t, s_t) + \lambda(r_t + \gamma \max_a Q(a, s_{t+1}))$$

applied after each visited state. All Qs are initialised to 0s. The learning rate was set to $\lambda = 0.01$. The policy was trained for 30k iterations.

When treating the Q-Learned policy as a teacher, depending on the temperature T reported (by default 0) it was either a greedy policy (if the temperature is 0)

$$\hat{\pi}(a|s) = 1 \text{ iff } Q(a,s) = \max_{b \in \mathcal{A}} Q(b,s)$$

$$\pi(a|s) = \frac{\hat{\pi}(a|s)}{\sum_{b \in \mathcal{A}} \hat{\pi}(b|s)}$$

or a Boltzman policy computed as:

$$\pi(a|s) = \frac{\exp(Q(a,s)/T)}{\sum_{b \in \mathcal{A}} \exp(Q(b,s)/T)}$$

A.5 Policy parametrisation during distillation

Policies are represented as logits of each action, for each unique observation. Consequently for each observation o, and for action space $\mathcal{A}$ the policy for Actor Critic is parameterised as $\pi_\theta(a|o) = \frac{\exp(\theta_{a,o})}{\sum_{b \in \mathcal{A}} \exp(\theta_{b,o})}$.

Similarly, value functions are represented simply as one float per observation: $V_{\pi_\theta}(o) = \theta^V_o$, and Q-values $Q_{\pi_\theta}(a,o) = \theta^Q_{a,o}$.

B Extended figures

We include extended versions of various figures. Fig. 8 is an extended version of Fig. 5 including experiments with an A2C teacher.

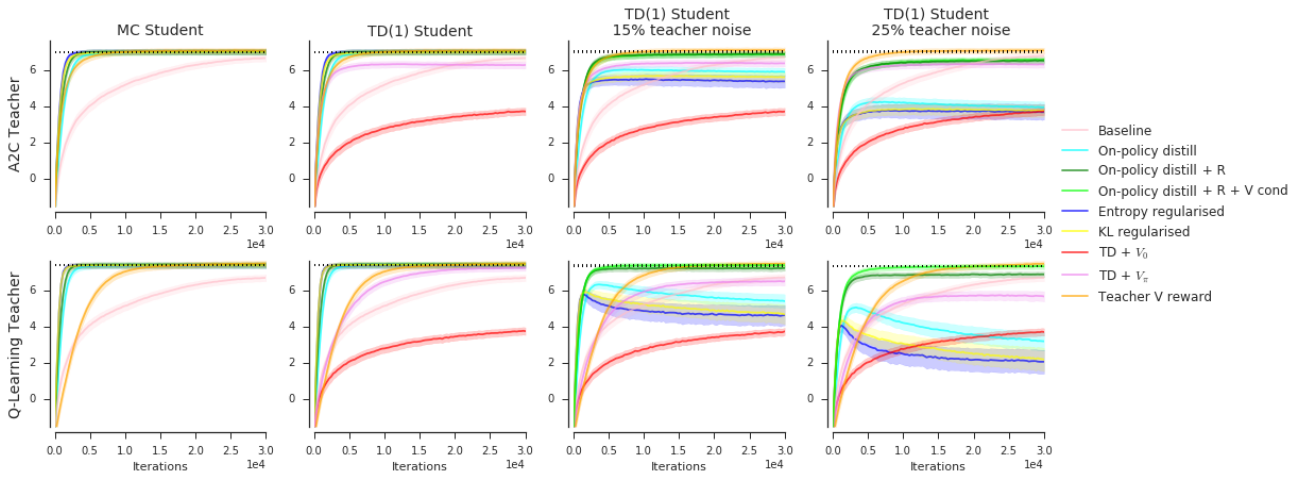

Figure 8: Learning curves obtained from averaging 1k training runs on randomly generated MDP grid worlds. The gradual decrease in reward when distilling from a sub-optimal Q-Learning teacher with distillation methods that enforce full *policy cloning* comes from the fact that the teacher is purely deterministic – while being closer to it initially helps, once the student replicates all the wrong state decisions perfectly its reward start to decrease. Extended version of Fig. 5 including A2C teacher.

Fig.9 is an extended version of Fig. 6, including more sizes of the corridor environment.

Fig.10 is an extended version of Fig. 4, including additional agents.

C Proofs for Section 3 (Policy distillations)

Theorem 1. *Let us assume that $g(\theta) = \mathbb{E}_{\pi_\theta}[\nabla_\theta \ell(\tau|\theta)]$ is differentiable and there does not exist $\alpha_\tau \in \mathbb{R}$ such that $\nabla_\theta \ell(\tau|\theta) = \alpha_\tau \nabla \pi_\theta(\tau)$ almost everywhere. Then $g(\theta)$ is not a gradient vector field of any function.*

Proof. If gradient of some f is differentiable then f's Hessian exists and is a symmetric matrix:

$$\frac{\partial}{\partial x}\left(\frac{\partial}{\partial y} f(x,y)\right) = \frac{\partial}{\partial y}\left(\frac{\partial}{\partial x} f(x,y)\right).$$

Consequently, if some function is a gradient vector field, then its Jacobian has to be symmetric. We will show that for g this is not true in general, by focusing on two arbitrary indices ij and ji. We use notation $f[i]$ to denote the ith output of the multivariate function f. Using the log derivative trick we obtain that $\frac{\partial}{\partial \theta_j} g(\theta)[i]$ equals

$$\begin{aligned}
&\frac{\partial}{\partial \theta_j}\mathbb{E}_{\pi_\theta}\left[\frac{\partial}{\partial \theta_i}\log \pi_\theta(\tau)\ell(\tau,\theta)\right] \\
=\ &\mathbb{E}_{\pi_\theta}\left[\frac{\partial}{\partial \theta_j}\log \pi_\theta(\tau)\frac{\partial}{\partial \theta_i}\log \pi_\theta(\tau)\ell(\tau,\theta)\right] \\
&+\mathbb{E}_{\pi_\theta}\left[\frac{\partial}{\partial \theta_j}[\frac{\partial}{\partial \theta_i}\log \pi_\theta(\tau)\ell(\tau,\theta)]\right] \\
=\ &\mathbb{E}_{\pi_\theta}\left[\frac{\partial}{\partial \theta_j}\log \pi_\theta(\tau)\frac{\partial}{\partial \theta_i}\log \pi_\theta(\tau)\ell(\tau,\theta)\right] \\
&+\mathbb{E}_{\pi_\theta}\left[\frac{\partial}{\partial \theta_i \theta_j}\log \pi_\theta(\tau)\ell(\tau,\theta)\right] \\
&+\mathbb{E}_{\pi_\theta}\left[\frac{\partial}{\partial \theta_i}\log \pi_\theta(\tau)\frac{\partial}{\partial \theta_j}\ell(\tau,\theta)\right]
\end{aligned}$$

thus $\frac{\partial}{\partial \theta_i} g(\theta)[j] - \frac{\partial}{\partial \theta_j} g(\theta)[i]$ equals

$$\begin{aligned}
&\mathbb{E}_{\pi_\theta}\left[\frac{\partial}{\partial \theta_j}\log \pi_\theta(\tau)\frac{\partial}{\partial \theta_i}\ell(\tau,\theta) - \frac{\partial}{\partial \theta_i}\log \pi_\theta(\tau)\frac{\partial}{\partial \theta_j}\ell(\tau,\theta)\right] \\
&\quad = \int_\tau \left[\frac{\partial}{\partial \theta_j}\pi_\theta(\tau)\frac{\partial}{\partial \theta_i}\ell(\tau,\theta) - \frac{\partial}{\partial \theta_i}\pi_\theta(\tau)\frac{\partial}{\partial \theta_j}\ell(\tau,\theta)\right] d\tau
\end{aligned}$$

In general this term is zero iff $\nabla\ell(\tau,\theta) = \alpha_\tau \nabla \pi_\theta(\tau)$ almost everywhere, which can not be true due to assumptions. Consequently, $g(\theta)$ is not a gradient vector field of any function. □

Proposition 1. *Using an update rule of the form $\mathbb{E}_{\pi_\theta}[\sum_{t=1}^{|\tau|} \nabla_\theta \ell(\pi(\tau_t), \pi_\theta(\tau_t))]$ for a strongly stochastic[4] student policy, with episodic finite state-space MDPs and tabular policies, provides convergence to the teacher policy over all reachable states for the loss function ℓ, provided the optimiser used can minimise*

[4]Meaning that each for each action a, parameters θ and state s, $\pi_\theta(s)[a] > 0$.

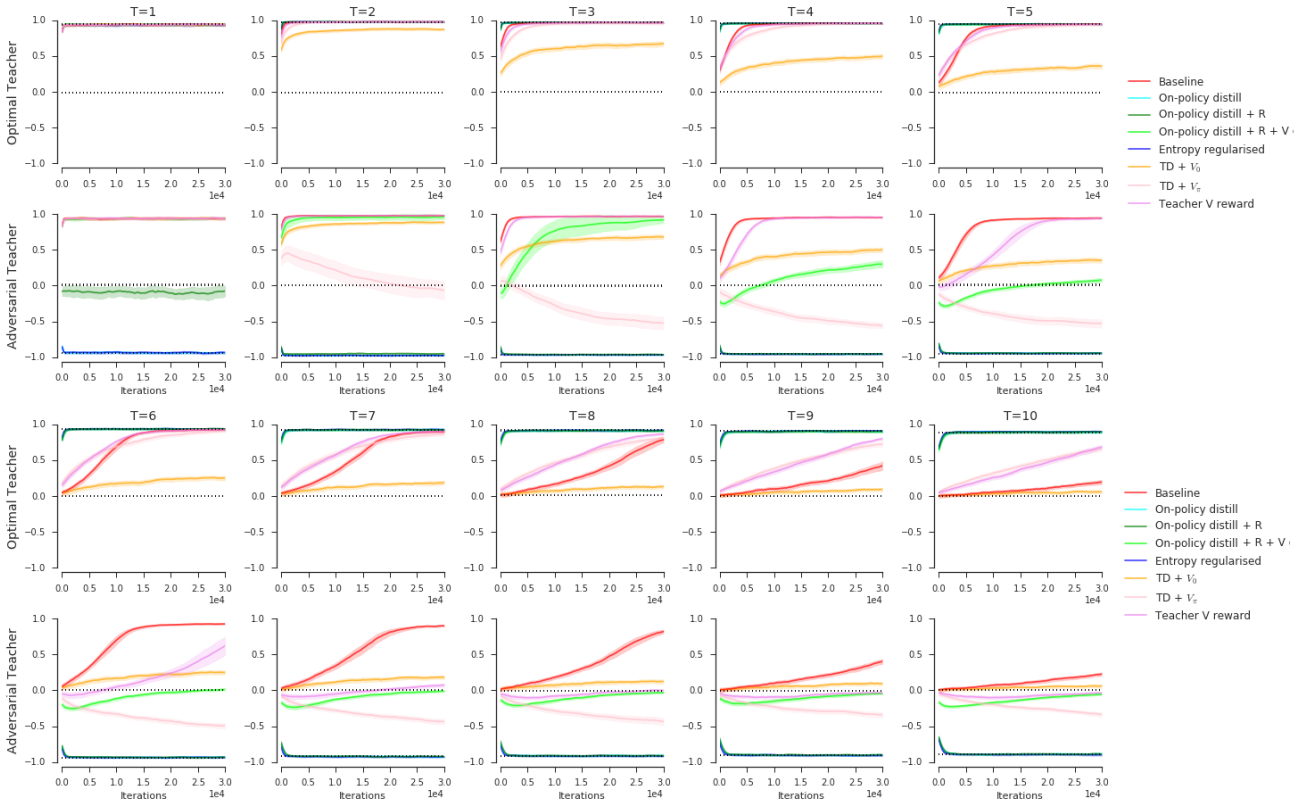

Figure 9: Results of distilling the optimal teacher and adversarial (minimising reward) teacher in the chain-structured MDP. Extended version of Fig. 6.

$\ell(a,b)$ wrt. b, for any a in the domain of ℓ, and $\ell(a,b)$ reaches minimum at $\ell(a,a)$.

Proof. Because of strong stochasticity of π_θ, the distribution of states visited under this policy covers entire state space $S = (s_1, \ldots, s_N)$ reachable from the initial state. We use notation $\ell^i_\theta := \ell(\pi(s_i), \pi_\theta(s_i))$. Consequently the update

$$g(\theta) := \mathbb{E}_{\pi_\theta}[\sum_{t=1}^{|\tau|} \nabla_\theta \ell(\pi(\tau_t), \pi_\theta(\tau_t))]$$

can be rewritten as

$$g(\theta) = \begin{bmatrix} p_\theta(s_1)\nabla_{\theta_1}\ell^1_\theta & \ldots & p_\theta(s_N)\nabla_{\theta_N}\ell^N_\theta \end{bmatrix}^T,$$

where $p_\theta(s)$ is the probability of agent being in state s when following policy π_θ and we use the independence of parametrisation of the policy in each state (which comes from the tabular assumption – θ_i is the parametrisation of policy in state s_i).

Let us denote by $g^*(\theta)$ gradient of a an expected loss under teacher policy

$$\begin{aligned} g^*(\theta) :&= \nabla_\theta[\mathbb{E}_\pi \sum_{t=1}^{|\tau|} \ell(\pi(\tau_t), \pi_\theta(\tau_t))] \\ &= \mathbb{E}_\pi \nabla_\theta[\sum_{t=1}^{|\tau|} \ell(\pi(\tau_t), \pi_\theta(\tau_t))] \\ &= \begin{bmatrix} p(s_1)\nabla_{\theta_1}\ell^1_\theta & \ldots & p(s_N)\nabla_{\theta_N}\ell^N_\theta \end{bmatrix}^T. \end{aligned}$$

where again $p(s)$ is the probability of sampling state s under π.

It is easy to notice that these two update directions have a non-negative cosine:

$$\langle g(\theta), g^*(\theta) \rangle = \sum_{i=1}^{N} p(s_i) p_\theta(s_i) \|\nabla_{\theta_i} \ell^i_\theta\|^2 \geq 0.$$

Furthermore, because for all s, $p(s) \geq 0, p_\theta(s) > 0$, the cosine is zero if and only if for each state s_i either $\|\nabla_{\theta_i}\ell^i_\theta\|^2 = 0$ (teacher and student policies match) or $p(s_i) = 0$ (state is not reachable by π). This means that for every state, reachable by π, the corresponding update rule coming from $g(\theta)$ is guaranteed to be stricly descending as long as it is not in the minimum.

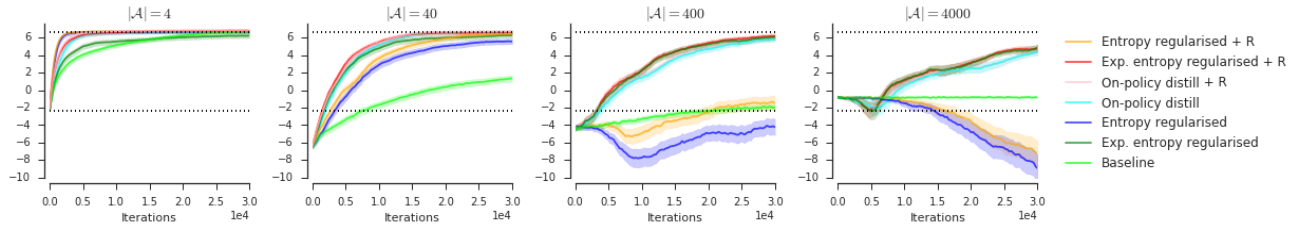

Figure 10: Learning curves averaged over 1k random MDPs with $|\mathcal{A}|$ actions, out of which 4 are movement actions and the remaining ones do not affect the movement of the agent, but simply make exploration hard. Plots show the failure mode of the intrinsic reward only based distillation, and how their expected version fixes it. Extended version of Fig. 4

Due to assumptions about $\ell(a, \cdot)$ having a unique minimum and optimiser being able to find it, we obtain that $\pi_\theta(s_i)$ will converge to $\pi(s_i)$ for each $s_i \in \mathcal{S}$ where $p(s_i) > 0$.

Consequently we have shown, that the update direction is a strict descent direction wrt. expected loss under the teacher policy and thus student policy converges to the teacher one over all reachable states.

Using Monte Carlo estimates for the $g(\theta)$ estimation can be analysed analogously to how Stochastic Gradient Descent generalises Gradient Descent. □

Oscillation example Consider a game with seven states, $\{s_0, s_L, s_R, s_{LL}, s_{LR}, s_{RL}, s_{RR}\}$. We start at s_0 and in the first step we decide whether to go to s_L or s_R. If we chose to go to s_L, in step 2 we chose whether to go to s_{LL} or to s_{LR}. Similarly, if we are in s_R after round 1, in step 2 we have a choice whether to go to s_{RL} or s_{RR}. The only rewards are $r(L, s_L) = -1$, $r(R, s_L) = -2$, and $r(R, s_R) = -3$. In the game we use a policy π_θ depending on two parameters θ_x and θ_y as follows. In the first step we go to s_R with probability $\frac{e^{\theta_x}}{1+e^{\theta_x}}$ and to s_L with probability $\frac{1}{1+e^{\theta_x}}$. In step 2 we have two branchings again, if we are in s_L with probalility $\frac{e^{\theta_y}}{1+e^{\theta_y}}$ we go to s_{LL}, and with probability $\frac{1}{1+e^{\theta_y}}$ we go to s_{LR}. Similarly, if we are in s_R we go with probalility $\frac{e^{\theta_y}}{1+e^{\theta_y}}$ to s_{RL}, and with probability $\frac{1}{1+e^{\theta_y}}$ we go to s_{RR}. We choose a penalty function $\ell = \ell(\theta_y) = 4\frac{e^{\theta_y}}{1+e^{\theta_y}} - 4$, living in the state s_R, when we are in s_L in step 2, ℓ is zero. Equivalently one can think of it being a distillation cost with an information potential loss, $\ell(\pi(s)\|\pi_\theta(s)) = 4\sum_a \pi(a|s)\pi_\theta(a|s) - 4$ where the teacher $\pi(R|s_L) = 1$. We have an update rule

$$
\begin{cases} \dot{x} = \frac{\partial}{\partial \theta_x} \mathbb{E}_{\pi_\theta}[\sum_{t=1}^{|\tau|} r_t] \\ \dot{y} = \frac{\partial}{\partial \theta_y} \mathbb{E}_{\pi_\theta}[\sum_{t=1}^{|\tau|} r_t] - \frac{e^{\theta_x}}{1+e^{\theta_x}} \ell'(\theta_y) \end{cases}
$$

$$
\begin{cases} \dot{x} = \frac{e^{\theta_x}(e^{\theta_y}-1)}{(1+e^{\theta_x})^2(1+e^{\theta_y})} \\ \dot{y} = \frac{e^{\theta_y}(1+3e^{\theta_x})}{(1+e^{\theta_x})(1+e^{\theta_y})^2} - 4\frac{e^{\theta_x}e^{\theta_y}}{(1+e^{\theta_x})(1+e^{\theta_y})^2} \end{cases}
$$

$$
\begin{cases} \dot{x} = \frac{e^{\theta_x}(e^{\theta_y}-1)}{(1+e^{\theta_x})^2(1+e^{\theta_y})} \\ \dot{y} = \frac{e^{\theta_y}(1-e^{\theta_x})}{(1+e^{\theta_x})(1+e^{\theta_y})^2}. \end{cases}
$$

This system of equations has a first integral $H(\theta_x, \theta_y) = e^{\theta_x} + e^{-\theta_x} + e^{\theta_y} + e^{-\theta_y}$ (with integrating factor $\frac{e^{\theta_x}e^{\theta_y}}{(1+e^{\theta_x})^2(1+e^{\theta_y})^2}$). Note, that $H(\theta_x, \theta_y) = 4 + {\theta_x}^2 + {\theta_y}^2 + \mathcal{O}({\theta_x}^3, {\theta_y}^3)$, therefore the fixed point $\theta = (0, 0)$ is a center. Therefore, with each policy update the values θ stay on the same closed curve and they keep changing in a cyclic manner, never converging.

Theorem 2. *In order to recover the gradient vector field property for 1-step on-policy distillation updates with any loss $\ell(\pi(\tau_t)\|\pi_\theta(\tau_t))$, one can add an extra reward term $\hat{r}_t = -\ell(\pi(\tau_{t+1})\|\pi_\theta(\tau_{t+1}))$. Analogously if the loss is of the form $\mathbb{E}_{a\sim\pi_\theta}\hat{\ell}(\pi(\tau_t))$ then the correction is of form $-\hat{\ell}(\pi(\tau_{t+1}))$.*

Proof. Consider the following loss $\mathcal{L}(\theta) = \mathbb{E}_{\pi_\theta}[\ell(\tau, \theta)]$ and its gradient:

$$
\begin{aligned}
\nabla_\theta \mathcal{L}(\theta) &= \nabla_\theta \int_\tau \pi_\theta(\tau|\theta)\,[\ell(\tau,\theta)]\,d\tau \\
&= \int_\tau \nabla_\theta(\pi_\theta(\tau|\theta)\,[\ell(\tau,\theta)])d\tau \\
&= \int_\tau [\nabla_\theta \pi_\theta(\tau|\theta)]\ell(\tau,\theta) + \pi_\theta(\tau|\theta)[\nabla_\theta \ell(\tau,\theta)]d\tau
\end{aligned}
$$

using the log-derivative trick $\nabla_\theta f(x) =$

$f(x)\nabla_\theta \log f(x)$ and the above equation we get

$$\begin{aligned}\nabla_\theta \mathcal{L}(\theta) = & \int_\tau [\pi_\theta(\tau|\theta)\nabla_\theta \log \pi_\theta(\tau|\theta)]\ell(\tau,\theta) + \\ & \pi_\theta(\tau|\theta)[\nabla_\theta \ell(\tau,\theta)]d\tau \\ = & \int_\tau [\pi_\theta(\tau|\theta)\nabla_\theta \log \pi_\theta(\tau|\theta)]\ell(\tau,\theta)d\tau + \\ & \int_\tau \pi_\theta(\tau|\theta)[\nabla_\theta \ell(\tau,\theta)]d\tau \\ = & \mathbb{E}_{\pi_\theta(\tau|\theta)}\nabla_\theta \log \pi_\theta(\tau|\theta)\ell(\tau,\theta) + \\ & \mathbb{E}_{\pi_\theta(\tau|\theta)}\nabla_\theta \ell(\tau,\theta)\end{aligned}$$

Consequently, we obtain that the valid gradient of the loss considered is composed of two expectations, one being the equivalent of a RL target, but with ℓ being a negation of the reward, and one which is exactly the auxiliary cost of interest. Consequently if we add the reward at time t equal to minus loss at time $t+1$ we will recover proper gradient vector field.

For the case of a loss of the form $\mathbb{E}_{a\sim\pi_\theta}\hat{\ell}(\pi(\tau_t))$ this proof is analogous – simply the correction is not on a state-action pair level, rather a pure state level. □

Cross entropy minima Let us fix a distribution $p(a|s)$, and consider a minima of $\mathrm{H}^\times(p\|q)$ and $\mathrm{H}^\times(q\|p)$ wrt. q. It is easy to see that the minimum of $\mathrm{H}^\times(p\|q)$ is given by p, as by the definition of divergence, the minimum of $\mathrm{KL}^\times(p\|q)$ is given by p, and $\mathrm{KL}^\times(p\|q) = \mathrm{H}^\times(p\|q) + \mathrm{H}(p)$, but for a fixed p, $\mathrm{H}(p)$ is a constant, thus it does not affect the minima. For $\mathrm{H}^\times(q\|p)$ we will show that the minimum is given by the dirac delta distribution in the most probable action a^* in p, denoted as q^*. For simplicity, assuming that this is a unique action, meaning that $\forall_{a\neq a^*} p(a|s) < p(a^*|s)$, then for any $q \neq q^*$

$$\begin{aligned}\mathrm{H}^\times(q\|p) &= -\sum_a q(a|s)\log p(a|s) \\ &> -[\sum_a q(a|s)]\max_b \log p(b|s) \\ &= -[1]\log p(a^*|s) = \mathrm{H}^\times(q^*\|p)\end{aligned}$$

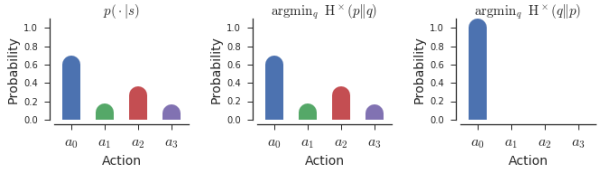

Figure 11: Comparison of various cross-entropies solutions when matching the distribution over finitely many actions.

KL and mean/mode seeking While both $\mathrm{KL}(q\|p)$ and $\mathrm{KL}(p\|q)$ have the same minimum in the space of all distributions, they differ once one constrains the space we are looking over. To be more precise we have that

$$\arg\min_q \mathrm{KL}(q\|p) = \arg\min_q \mathrm{KL}(p\|q) = p$$

but at the same time there exists $C \subset \mathcal{P}$ where $\mathcal{P}$ is the space of all distributions such that

$$\arg\min_{q\in C} \mathrm{KL}(q\|p) \neq \arg\min_{q\in C} \mathrm{KL}(p\|q) \neq p$$

The simplest example is the mixture of multiple Gaussians, which we try to fit with just a single Gaussian. The typical cost of $\mathrm{KL}(p\|q)$ will match the mean of the distribution (thus the name of *mean seeking*), while $\mathrm{KL}(q\|p)$ will cover one of the Gaussians from the mixture, while ignoring the others (thus *mode seeking*), see Fig. 12 and Fig. 13.

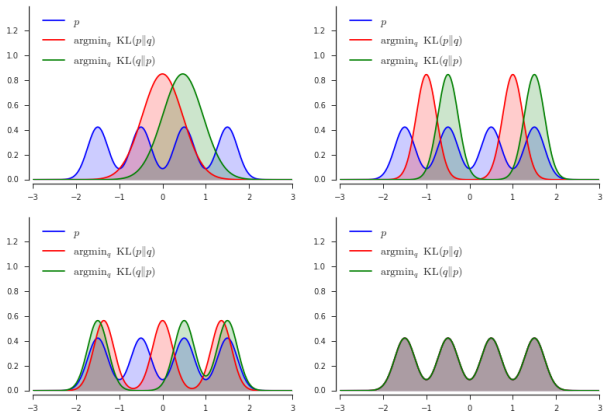

Figure 12: Comparison of various KL variant solutions when matching the distribution over a mixture of 4 Gaussians, using from 1 (upper left) to 4 (lower right) Gaussians. Note how mode seeking KL (green) picks Gaussians to match, while ignoring others, and mean seeking (red) instead puts its Gaussians in between peaks of the original distribution.

In practice, we are often in this regime, since the teacher and student policies can have different capacities, architectures and priors, thus making perfect replication impossible. Therefore, the choice of *direction* of KL will affect if the agent prefers to just match one, very probable mode (action/behaviour), or if we prefer the agent to look for an averaged action/behaviour.

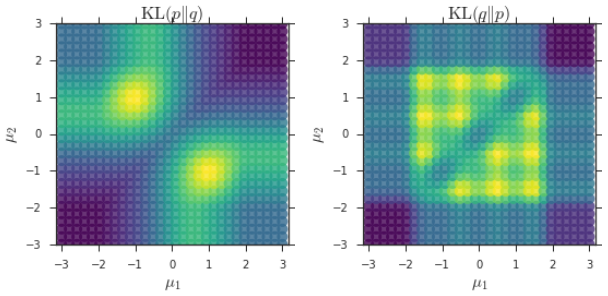

Figure 13: Visualisation of the value of KL (left) and reverse KL (right) parameterised by the location of the mean of two Gaussians, computed with respect to the mixture of 4 Gaussians from Fig. 12. One can see how mean seeking KL prefers to put means in -1 and 1 while the mode seeking attains the minimum for every possible pair, matching means of the original mixture.

D Proofs for Section 4 (Policy distillation from Actor-Critic)

Proposition 2. *For $\mathcal{S}_1$ being a distribution over initial states, if we have $\forall s \in \mathcal{S}\ \ell(\theta^*, s) \leq H(\pi(s))$ then $\mathbb{E}_{s\sim\mathcal{S}_1}[V_{\pi_{\theta^*}}(s)] \geq \mathbb{E}_{s\sim\mathcal{S}_1}[V_\pi(s)]$.*

Proof. Lets assume that the inequality does not hold, meaning that the following teacher's policy gives higher return. This means, that there exists a state s^*, where $V_\pi(s^*) > V_{\pi_{\theta^*}}(s^*)$ but the policies differ, meaning that $\pi(s^*) \neq \pi_{\theta^*}(s^*)$. However, if $V_\pi(s^*) > V_{\pi_{\theta^*}}(s^*)$ then $\ell(\theta, s^*) = \mathrm{H}^\times(\pi(s^*)\|\pi_\theta(s^*))$, and due to the assumption $\ell(\theta, s^*) \leq \mathrm{H}(\pi(s^*))$ for every state, leads to $\pi(s^*) = \pi_{\theta^*}(s^*)$ (as cross entropy is equal to entropy of the first argument only when the argument are the same), which is a contradiction. □

Proposition 3. *Assume that we are given the true value V_π of the teacher policy π, for a finite state size MDP, then optimising using $\mathbb{E}_{\pi_\theta}[\sum_t \nabla \log \pi_\theta(a_t|\tau_t)[r(a_t, \tau_t) + V_\pi(\tau_{t+1})]]$. converges to a policy with $\mathbb{E}_{s\sim\mathcal{S}_1} V_{\pi_\theta}(s) \geq \mathbb{E}_{s\sim\mathcal{S}_1} V_\pi(s)$ for $\mathcal{S}_1$ being the distribution of initial states.*

Proof. First, notice that for all initial states, the update rule provided basically solves the bandit problem, where the value of each action is a sum of an actual reward and the value of the teacher (implying following the teacher policy afterwards). In the worst case scenario it will simply find a distribution matching the teacher's, as it is going to optimise for the reward in the first step, and then fall back to the teacher's policy. Consequently, after enough updates, the policy π_θ will learn to take actions which do not have smaller values than those of the teacher if one was to follow the teacher policy afterwards. Now, using inductive reasoning, if π_θ is already defining a distribution over states visited up to n steps from the initial state which are guaranteed to produce values larger than the teacher, and if we were to follow teacher policy afterwards, then the update will also correct states in distance $n+1$. Given that we assumed that it is a finite state size MDP and updates to different states are independent, then the whole process has to eventually converge. □

Proposition 4. *Let us assume the teacher is an optimal policy for the given MDP, then for each action a_t that would lead to a deviation from the optimal path, it will get an immediate penalty, meaning that $r_t^V < r_t$, while following any of the optimal paths leads to $r_t^V = r_t$.*

Proof. It is easy to notice, that if an agent executes an action a_t which is on the optimal path, we have $V_\pi(\tau_{t+1}) = V_\pi(\tau_t)$, and thus $r_t^V = V_\pi(\tau_{t+1}) - V_\pi(\tau_t) + r_t = 0 + r_t = r_t$. If, on the other hand, it is not on the optimal path, then there exists $\epsilon > 0$ such that $V_\pi(\tau_{t+1}) = V_\pi(\tau_t) - \epsilon$ so $r_t^V = V_\pi(\tau_{t+1}) - V_\pi(\tau_t) + r_t = -\epsilon + r_t < r_t$ □