Appendix

A. Proofs

A.1. Lipschitz MDP

Lemma 1. *Let $\mathcal{M}$ be (K_R, K_P)-Lipschitz and let π be any policy with the property that $\forall s_1, s_2 \in \mathcal{S}$,*

$$|V^\pi(s_1) - V^\pi(s_2)| \leqslant \max_{a \in \mathcal{A}} |Q^\pi(s_1, a) - Q^\pi(s_2, a)|$$

then π is $\frac{K_\mathcal{R}}{1-\gamma K_\mathcal{P}}$-Lipschitz-valued.

Proof. Let

$$K_Q = \inf \left\{ L \in \mathbb{R} \cup \{\infty\} \;\middle|\; \forall s_1, s_2 \in \bar{\mathcal{S}}, |Q^\pi(s_1, a) - Q^\pi(s_2, a)| \leqslant L d_\mathcal{S}(s_1, s_2) \right\}. \tag{9}$$

First note that the assumption of the Lemma implies that $K_V \leqslant K_Q$, where K_V is the Lipschitz norm of the value function.

We will show that K_Q is bounded by a recurrence relationship. The derived recurrence will have a finite fixed point, thus proving that K_Q is finite. See that,

$$|Q^\pi(s_1, a) - Q^\pi(s_2, a)| \leqslant |\mathcal{R}(s_1, a) - \mathcal{R}(s_2, a)| + \gamma \left| \int_{\bar{\mathcal{S}}} (\bar{\mathcal{P}}(s'|s_1, a) - \mathcal{P}(s'|s_2, a)) V^\pi(s', a')\, ds' \right|.$$

The first term of the RHS may be bounded above by $K_\mathcal{R} \|s_1 - s_2\|_2$. For the second term of the RHS we apply the definition of the Wasserstein metric (Section 1).

$$\begin{aligned} |Q^\pi(s_1, a) - Q^\pi(s_2, a)| &\leqslant K_\mathcal{R} d_\mathcal{S}(s_1, s_2) + \gamma K_V W\left(\mathcal{P}(\cdot|s_1, a), \mathcal{P}(\cdot|s_2, a)\right) \\ &\leqslant (K_\mathcal{R} + \gamma K_V K_\mathcal{P}) d_\mathcal{S}(s_1, s_2) \\ &\leqslant (K_\mathcal{R} + \gamma K_Q K_\mathcal{P}) d_\mathcal{S}(s_1, s_2) \end{aligned}$$

This recurrence has a finite fixed point given by $\frac{K_R}{1-\gamma K_P}$. Thus the conditions of Lipschitz-Valued policies (Definition 3) are satisfied, completing the proof. □

Corollary 1. *Let $\mathcal{M}$ be (K_R, K_P)-Lipschitz, then π^* is $\frac{K_\mathcal{R}}{1-\gamma K_\mathcal{P}}$-Lipschitz-Valued.*

Proof.

$$\begin{aligned} |V^\pi(s_1) - V^\pi(s_2)| &= |\max_{a_1 \in \mathcal{A}} Q^\pi(s_1, a_1) - \max_{a_2 \in \mathcal{A}} Q^\pi(s_2, a_2)| \\ &\leqslant \max_{a \in \mathcal{A}} |Q^\pi(s_1, a) - Q^\pi(s_2, a)| \end{aligned}$$

Thus the condition for Lemma 1 holds and the result follows. □

A.2. Global DeepMDP

Lemma 2. *Let $\mathcal{M}$ and $\overline{\mathcal{M}}$ be an MDP and DeepMDP respectively, with an embedding function ϕ and global loss functions $L^\infty_\mathcal{R}$ and $L^\infty_\mathcal{P}$. For any $K_{\bar{V}}$-Lipschitz-valued policy $\bar{\pi} \in \bar{\Pi}$ the value difference can be bounded by*

$$\left|Q^{\bar{\pi}}(s, a) - \bar{Q}^{\bar{\pi}}(\phi(s), a)\right| \leqslant \frac{L^\infty_\mathcal{R} + \gamma K_{\bar{V}} L^\infty_\mathcal{P}}{1 - \gamma},$$

Proof. The proof consists of showing that the supremum $\sup_{s,a} \left|Q^{\bar{\pi}}(s, a) - \bar{Q}^{\bar{\pi}}(\phi(s), a)\right|$ is bounded by a recurrence

relationship.

$$\begin{aligned}
\max_{s,a}\left|Q^\pi(s,a)-\bar{Q}^\pi(\phi(s),a)\right| &\leqslant \max_{s,a}\left|\mathcal{R}(s,a)-\bar{\mathcal{R}}(\phi(s),a)\right| + \gamma\max_{s,a}\left|\mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)}V^\pi(s') - \mathop{\mathbb{E}}_{\bar{s}'\sim\bar{\mathcal{P}}(\cdot|\phi(s),a)}\bar{V}^\pi(\bar{s}')\right| \\
&= L^\infty_{\bar{\mathcal{R}}} + \gamma\max_{s,a}\left|\mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)}\left[V^\pi(s')-\bar{V}^\pi(\phi(s'))\right] + \mathop{\mathbb{E}}_{\substack{\bar{s}'\sim\bar{\mathcal{P}}(\cdot|\phi(s),a)\\ s'\sim\mathcal{P}(\cdot|s,a)}}\left[\bar{V}^\pi(\phi(s'))-\bar{V}^\pi(\bar{s}')\right]\right| \\
&\leqslant L^\infty_{\bar{\mathcal{R}}} + \gamma\max_{s,a}\left|\mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)}\left[V^\pi(s')-\bar{V}^\pi(\phi(s'))\right]\right| + \gamma\max_{s,a}\left|\mathop{\mathbb{E}}_{\substack{\bar{s}'\sim\bar{\mathcal{P}}(\cdot|\phi(s),a)\\ s'\sim\mathcal{P}(\cdot|s,a)}}\left[\bar{V}^\pi(\phi(s'))-\bar{V}^\pi(\bar{s}')\right]\right| \\
&\leqslant L^\infty_{\bar{\mathcal{R}}} + \gamma\max_{s,a}\left|\mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)}\left[V^\pi(s')-\bar{V}^\pi(\phi(s'))\right]\right| + \gamma K_{\bar{V}}\max_{s,a}W\left(\phi\mathcal{P}(\cdot|s,a),\bar{\mathcal{P}}(\cdot|\phi(s),a)\right) \\
&= L^\infty_{\bar{\mathcal{R}}} + \gamma\max_{s,a}\left|\mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)}\left[V^\pi(s')-\bar{V}^\pi(\phi(s'))\right]\right| + \gamma K_{\bar{V}}L^\infty_{\bar{\mathcal{P}}} \\
&\leqslant L^\infty_{\bar{\mathcal{R}}} + \gamma\max_{s,a}\mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)}\left|\left[V^\pi(s')-\bar{V}^\pi(\phi(s'))\right]\right| + \gamma K_{\bar{V}}L^\infty_{\bar{\mathcal{P}}} \text{ Using Jensen's inequality.} \\
&\leqslant L^\infty_{\bar{\mathcal{R}}} + \gamma\max_{s,a}\left|\left[V^\pi(s)-\bar{V}^\pi(\phi(s))\right]\right| + \gamma K_{\bar{V}}L^\infty_{\bar{\mathcal{P}}} \\
&\leqslant L^\infty_{\bar{\mathcal{R}}} + \gamma\max_{s,a}\left|\left[Q^\pi(s,a)-\bar{Q}^\pi(\phi(s),a)\right]\right| + \gamma K_{\bar{V}}L^\infty_{\bar{\mathcal{P}}}
\end{aligned}$$

Solving for the recurrence relation over $\max_{s,a}\left|Q^\pi(s,a)-\bar{Q}^\pi(\phi(s),a)\right|$ results in the desired result. □

Theorem 1. *Let $\mathcal{M}$ and $\overline{\mathcal{M}}$ be an MDP and DeepMDP respectively, with an embedding function ϕ and global loss functions $L^\infty_{\bar{\mathcal{R}}}$ and $L^\infty_{\bar{\mathcal{P}}}$. For any $K_{\bar{V}}$-Lipschitz-valued policy $\bar{\pi}\in\bar{\Pi}$ the representation ϕ guarantees that for any $s_1, s_2\in\mathcal{S}$ and $a\in\mathcal{A}$,*

$$\left|Q^{\bar{\pi}}(s_1,a)-Q^{\bar{\pi}}(s_2,a)\right| \leqslant K_{\bar{V}}\left\|\phi(s_1)-\phi(s_2)\right\|_2 + 2\frac{\left(L^\infty_{\bar{\mathcal{R}}}+\gamma K_{\bar{V}}L^\infty_{\bar{\mathcal{P}}}\right)}{1-\gamma}$$

Proof.

$$\begin{aligned}
\left|Q^{\bar{\pi}}(s_1,a)-Q^{\bar{\pi}}(s_2,a)\right| &\leqslant \left|\bar{Q}^{\bar{\pi}}(s_1,a)-\bar{Q}^{\bar{\pi}}(s_2,a)\right| + \left|Q^{\bar{\pi}}(s_1,a)-\bar{Q}^{\bar{\pi}}(s_1,a)\right| + \left|Q^{\bar{\pi}}(s_2,a)-\bar{Q}^{\bar{\pi}}(s_2,a)\right| \\
&\leqslant \left|\bar{Q}^{\bar{\pi}}(s_1,a)-\bar{Q}^{\bar{\pi}}(s_2,a)\right| + 2\frac{\left(L^\infty_{\bar{\mathcal{R}}}+\gamma K_V L^\infty_{\bar{\mathcal{P}}}\right)}{1-\gamma} \text{ Applying Lemma 2} \\
&\leqslant K_V\left\|\phi(s_1)-\phi(s_2)\right\|_2 + 2\frac{\left(L^\infty_{\bar{\mathcal{R}}}+\gamma K_V L^\infty_{\bar{\mathcal{P}}}\right)}{1-\gamma} \text{ Using the Lipschitz property of } \bar{Q}^{\bar{\pi}}
\end{aligned}$$

□

Theorem 2. *Let $\mathcal{M}$ and $\overline{\mathcal{M}}$ be an MDP and a (K_R, K_P)-Lipschitz DeepMDP respectively, with an embedding function ϕ and global loss functions $L^\infty_{\bar{\mathcal{R}}}$ and $L^\infty_{\bar{\mathcal{P}}}$. For all $s\in\mathcal{S}$, the suboptimality of the optimal policy $\bar{\pi}^*$ of $\overline{\mathcal{M}}$ evaluated on $\mathcal{M}$ can be bounded by,*

$$V^*(s) - V^{\bar{\pi}^*}(s) \leqslant 2\frac{L^\infty_{\bar{\mathcal{R}}}+\gamma K_{\bar{V}}L^\infty_{\bar{\mathcal{P}}}}{1-\gamma}$$

Where $K_{\bar{V}} = \frac{K_{\bar{\mathcal{R}}}}{1-\gamma K_{\bar{\mathcal{P}}}}$ is an upper bound to the Lipschitz constant of the value function $\bar{V}^{\bar{\pi}^*}$, as shown by Corollary 1.

Proof. For any $s\in\mathcal{S}$ we have

$$\left|V^*(s)-V^{\bar{\pi}^*}(s)\right| \leqslant \left|\bar{V}^*(\phi(s))-V^{\bar{\pi}^*}(s)\right| + \left|V^*(s)-\bar{V}^*(\phi(s))\right|. \tag{10}$$

Using the result given by Lemma 2 and Corollary 1, we may bound the first term of the RHS by $\frac{L^\infty_{\bar{\mathcal{R}}}+\gamma K_V L^\infty_{\bar{\mathcal{P}}}}{1-\gamma}$.

To complete the proof, we want to show that for any $s \in \mathcal{S}, a \in \mathcal{A}$, we have,

$$|Q^*(s,a) - \bar{Q}^*(\phi(s),a)| \leqslant \frac{L^\infty_{\bar{\mathcal{R}}} + \gamma K_V L^\infty_{\bar{\mathcal{P}}}}{1-\gamma}, \tag{11}$$

$$|V^*(s) - \bar{V}^*(\phi(s))| \leqslant \frac{L^\infty_{\bar{\mathcal{R}}} + \gamma K_V L^\infty_{\bar{\mathcal{P}}}}{1-\gamma}. \tag{12}$$

We prove the validity of Equation 11 similarly to Lemma 2:

$$\begin{aligned}
\max_{s,a} \left|Q^*(s,a) - \bar{Q}^*(\phi(s),a)\right| &\leqslant \max_{s,a} \left|\mathcal{R}(s,a) - \bar{\mathcal{R}}(\phi(s),a)\right| + \gamma \max_{s,a} \left| \mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)} V^*(s') - \mathop{\mathbb{E}}_{\bar{s}'\sim\bar{\mathcal{P}}(\cdot|\phi(s),a)} \bar{V}^*(\bar{s}') \right| \\
&= L^\infty_{\bar{\mathcal{R}}} + \gamma \max_{s,a} \left| \mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)} [V^*(s') - \bar{V}^*(\phi(s'))] + \mathop{\mathbb{E}}_{\substack{\bar{s}'\sim\bar{\mathcal{P}}(\cdot|\phi(s),a) \\ s'\sim\mathcal{P}(\cdot|s,a)}} [\bar{V}^*(\phi(s')) - \bar{V}^*(\bar{s}')] \right| \\
&\leqslant L^\infty_{\bar{\mathcal{R}}} + \gamma \max_{s,a} \left| \mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)} [V^*(s') - \bar{V}^*(\phi(s'))] \right| + \gamma \max_{s,a} \left| \mathop{\mathbb{E}}_{\substack{\bar{s}'\sim\bar{\mathcal{P}}(\cdot|\phi(s),a) \\ s'\sim\mathcal{P}(\cdot|s,a)}} [\bar{V}^*(\phi(s')) - \bar{V}^*(\bar{s}')] \right| \\
&= L^\infty_{\bar{\mathcal{R}}} + \gamma \max_{s,a} \left| \mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)} [V^*(s') - \bar{V}^*(\phi(s'))] \right| + \gamma K_{\bar{V}} L^\infty_{\bar{\mathcal{P}}} \\
&\leqslant L^\infty_{\bar{\mathcal{R}}} + \gamma \max_{s} \left| V^*(s) - \bar{V}^*(\phi(s)) \right| + \gamma K_{\bar{V}} L^\infty_{\bar{\mathcal{P}}} \text{ Using Jensen's inequality.} \\
&= L^\infty_{\bar{\mathcal{R}}} + \gamma \max_{s} \left| \max_a Q^*(s,a) - \max_a \bar{Q}^*(\phi(s),a) \right| + \gamma K_{\bar{V}} L^\infty_{\bar{\mathcal{P}}} \\
&\leqslant L^\infty_{\bar{\mathcal{R}}} + \gamma \max_{s,a} \left| Q^*(s,a) - \bar{Q}^*(\phi(s),a) \right| + \gamma K_{\bar{V}} L^\infty_{\bar{\mathcal{P}}}
\end{aligned}$$

Solving for the recurrence proves Equation 11. Now to show the validity of Equation 12, we derive,

$$\begin{aligned}
|V^*(s) - \bar{V}^*(\phi(s))| &= |\max_a Q^*(s,a) - \max_{a'} \bar{Q}^*(\phi(s),a')| && (13) \\
&\leqslant \max_a |Q^*(s,a) - \bar{Q}^*(\phi(s),a)| && (14) \\
&\leqslant \frac{L^\infty_{\bar{\mathcal{R}}} + \gamma K_V L^\infty_{\bar{\mathcal{P}}}}{1-\gamma}. && (15)
\end{aligned}$$

as desired. This completes the proof. □

A.3. Local DeepMDP Proofs

Lemma 3. *Let $\mathcal{M}$ and $\bar{\mathcal{M}}$ be an MDP and DeepMDP respectively, with an embedding function ϕ. For any $K_{\bar{V}}$-Lipschitz-valued policy $\bar{\pi} \in \bar{\Pi}$, the expected value function difference can be bounded using the local loss functions $L^{\xi_{\bar{\pi}}}_{\bar{\mathcal{R}}}$ and $L^{\xi_{\bar{\pi}}}_{\bar{\mathcal{P}}}$ measured under $\xi_{\bar{\pi}}$, the stationary state action distribution of $\bar{\pi}$.*

$$\mathop{\mathbb{E}}_{s,a\sim\xi_{\bar{\pi}}} \left|Q^{\bar{\pi}}(s,a) - \bar{Q}^{\bar{\pi}}(\phi(s),a)\right| \leqslant \frac{\left(L^{\xi_{\bar{\pi}}}_{\bar{\mathcal{R}}} + \gamma K_{\bar{V}} L^{\xi_{\bar{\pi}}}_{\bar{\mathcal{P}}}\right)}{1-\gamma},$$

Proof.

$$
\begin{aligned}
\mathop{\mathbb{E}}_{s,a\sim\xi_\pi}\left|Q^\pi(s,a)-\bar{Q}^\pi(\phi(s),a)\right| &\leqslant \mathop{\mathbb{E}}_{s,a\sim\xi_\pi}\left|\mathcal{R}(s,a)-\bar{\mathcal{R}}(\phi(s),a)\right| + \gamma \mathop{\mathbb{E}}_{s,a\sim\xi_\pi}\left|\mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)} V^\pi(s') - \mathop{\mathbb{E}}_{\bar{s}'\sim\bar{\mathcal{P}}(\cdot|\phi(s),a)} \bar{V}^\pi(\bar{s}')\right| \\
&= L^{\xi_\pi}_{\bar{\mathcal{R}}} + \gamma \mathop{\mathbb{E}}_{s,a\sim\xi_\pi}\left|\mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)}[V^\pi(s')-\bar{V}^\pi(\phi(s'))] + \mathop{\mathbb{E}}_{\substack{\bar{s}'\sim\bar{\mathcal{P}}(\cdot|\phi(s),a)\\ s'\sim\mathcal{P}(\cdot|s,a)}}[\bar{V}^\pi(\phi(s'))-\bar{V}^\pi(\bar{s}')]\right| \\
&\leqslant L^{\xi_\pi}_{\bar{\mathcal{R}}} + \gamma \mathop{\mathbb{E}}_{s,a\sim\xi_\pi}\left|\mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)}[V^\pi(s')-\bar{V}^\pi(\phi(s'))]\right| + \gamma \mathop{\mathbb{E}}_{s,a\sim\xi_\pi}\left|\mathop{\mathbb{E}}_{\substack{\bar{s}'\sim\bar{\mathcal{P}}(\cdot|\phi(s),a)\\ s'\sim\mathcal{P}(\cdot|s,a)}}[\bar{V}^\pi(\phi(s'))-\bar{V}^\pi(\bar{s}')]\right| \\
&\leqslant L^{\xi_\pi}_{\bar{\mathcal{R}}} + \gamma \mathop{\mathbb{E}}_{s,a\sim\xi_\pi}\left|\mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)}[V^\pi(s')-\bar{V}^\pi(\phi(s'))]\right| + \gamma K_{\bar{V}} \mathop{\mathbb{E}}_{s,a\sim\xi_\pi} W\left(\phi\mathcal{P}(\cdot|s,a), \bar{\mathcal{P}}(\cdot|\phi(s),a)\right) \\
&= L^{\xi_\pi}_{\bar{\mathcal{R}}} + \gamma \mathop{\mathbb{E}}_{s,a\sim\xi_\pi}\left|\mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)}[V^\pi(s')-\bar{V}^\pi(\phi(s'))]\right| + \gamma K_{\bar{V}} L^{\xi_\pi}_{\bar{\mathcal{P}}} \\
&\leqslant L^{\xi_\pi}_{\bar{\mathcal{R}}} + \gamma \mathop{\mathbb{E}}_{s,a\sim\xi_\pi}\mathop{\mathbb{E}}_{s'\sim\mathcal{P}(\cdot|s,a)}\left|[V^\pi(s')-\bar{V}^\pi(\phi(s'))]\right| + \gamma K_{\bar{V}} L^{\xi_\pi}_{\bar{\mathcal{P}}} \text{ Using Jensen's inequality.} \\
&\leqslant L^{\xi_\pi}_{\bar{\mathcal{R}}} + \gamma \mathop{\mathbb{E}}_{s,a\sim\xi_\pi}\left|[V^\pi(s)-\bar{V}^\pi(\phi(s))]\right| + \gamma K_{\bar{V}} L^{\xi_\pi}_{\bar{\mathcal{P}}} \text{ Applying the stationarity property.} \\
&\leqslant L^{\xi_\pi}_{\bar{\mathcal{R}}} + \gamma \mathop{\mathbb{E}}_{s,a\sim\xi_\pi}\left|[Q^\pi(s,a)-\bar{Q}^\pi(\phi(s),a)]\right| + \gamma K_{\bar{V}} L^{\xi_\pi}_{\bar{\mathcal{P}}}
\end{aligned}
$$

Solving for the recurrence relation over $\mathbb{E}_{s,a\sim\xi_{\bar{\pi}}}\left|Q^{\bar{\pi}}(s,a)-\bar{Q}^{\bar{\pi}}(\phi(s),a)\right|$ results in the desired result. □

Theorem 3. *Let $\mathcal{M}$ and $\overline{\mathcal{M}}$ be an MDP and DeepMDP respectively, with an embedding function ϕ. Let $\bar{\pi}\in\bar{\Pi}$ be any $K_{\bar{V}}$-Lipschitz-valued policy with stationary distribution $d_{\bar{\pi}}(s)$ and let $L^{\xi_{\bar{\pi}}}_{\bar{\mathcal{R}}}$ and $L^{\xi_{\bar{\pi}}}_{\bar{\mathcal{P}}}$ be the local loss functions measured under $\xi_{\bar{\pi}}$, the stationary state action distribution of $\bar{\pi}$. For any two states $s_1, s_2\in\mathcal{S}$, the local representation similarity can be bounded by*

$$
\begin{aligned}
|V^{\bar{\pi}}(s_1)-V^{\bar{\pi}}(s_2)| \leqslant K_{\bar{V}}\left\|\phi(s_1)-\phi(s_2)\right\|_2 \\
+ \frac{L^{\xi_{\bar{\pi}}}_{\bar{\mathcal{R}}}+\gamma K_{\bar{V}} L^{\xi_{\bar{\pi}}}_{\bar{\mathcal{P}}}}{1-\gamma}\left(\frac{1}{d_{\bar{\pi}}(s_1)}+\frac{1}{d_{\bar{\pi}}(s_2)}\right)
\end{aligned}
$$

Proof. Using the fact that $\left|V^{\bar{\pi}}(s)-\bar{V}^{\bar{\pi}}(s)\right| \leqslant d^{-1}_{\bar{\pi}}(s)\mathbb{E}_{s\sim d_{\bar{\pi}}}\left|V^{\bar{\pi}}(s)-\bar{V}^{\bar{\pi}}(s)\right|$,

$$
\begin{aligned}
\left|V^{\bar{\pi}}(s_1)-V^{\bar{\pi}}(s_2)\right| &\leqslant \left|\bar{V}^{\bar{\pi}}(s_1)-\bar{V}^{\bar{\pi}}(s_2)\right| + d^{-1}_{\bar{\pi}}(s_1)\mathop{\mathbb{E}}_{s\sim d_{\bar{\pi}}}\left|V^{\bar{\pi}}(s_1)-\bar{V}^{\bar{\pi}}(s_1)\right| + d^{-1}_{\bar{\pi}}(s_2)\mathop{\mathbb{E}}_{s\sim d_{\bar{\pi}}}\left|V^{\bar{\pi}}(s_2)-\bar{V}^{\bar{\pi}}(s_2)\right| \\
&\leqslant \left|\bar{V}^{\bar{\pi}}(s_1)-\bar{V}^{\bar{\pi}}(s_2)\right| + \frac{L^{\xi_{\bar{\pi}}}_{\bar{\mathcal{R}}}+\gamma K_V L^{\xi_{\bar{\pi}}}_{\bar{\mathcal{P}}}}{1-\gamma}\left(d^{-1}_{\bar{\pi}}(s_1)+d^{-1}_{\bar{\pi}}(s_2)\right), \text{ Applying Lemma 3} \\
&\leqslant K_V\left\|\phi(s_1)-\phi(s_2)\right\|_2 + \frac{L^{\xi_{\bar{\pi}}}_{\bar{\mathcal{R}}}+\gamma K_V L^{\xi_{\bar{\pi}}}_{\bar{\mathcal{P}}}}{1-\gamma}\left(d^{-1}_{\bar{\pi}}(s_1)+d^{-1}_{\bar{\pi}}(s_2)\right)
\end{aligned}
$$

□

A.4. Connection to Bisimulation

Lemma 4. *Let $\mathcal{M}$ be a $K_{\mathcal{R}}$-$K_{\mathcal{P}}$-Lipschitz MDP, with metric between states $d_{\mathcal{S}}$. Then the bisimulation metric $\widetilde{d}$ is also Lipschitz.*

$$
\widetilde{d}(s_1,s_2) \leqslant \frac{(1-\gamma)K_{\mathcal{R}}}{1-\gamma K_{\mathcal{P}}} d_{\mathcal{S}}(s_1,s_2) \tag{16}
$$

Thus, all close states in $d_{\mathcal{S}}$ are also close under the bisimulation metric (although the converse need not be true).

Proof. Let $B_{\widetilde{d}} = \sup_{s_1,s_2 \in \mathcal{S}} \frac{\widetilde{d}(s_1,s_2}{d_{\mathcal{S}}(s_1,s_2)}$ be (the potentially infinite) Lipschitz constant of the bisimulation metric $\widetilde{d}$ such that:

$$\widetilde{d}(s_1, s_2) \leqslant B_{\widetilde{d}} d_{\mathcal{S}}(s_1, s_2), \forall s_1, s_2 \in \mathcal{S}$$

Thus,

$$\begin{aligned}
T_{\widetilde{d}}(\mathcal{P}(\cdot|s_1,a), \mathcal{P}(\cdot|s_2,a)) &= \sup_{f \in \mathbb{F}_d} \mathop{\mathbb{E}}_{s_1' \sim \mathcal{P}(\cdot|s_1,a)}[f(s_1')] - \mathop{\mathbb{E}}_{s_2' \sim \mathcal{P}(\cdot|s_2,a)}[f(s_2')] \\
&\leqslant B_{\widetilde{d}} \sup_{f \in \mathbb{F}_1} \mathop{\mathbb{E}}_{s_1 \sim P(\cdot)}[f(s_1)] - \mathop{\mathbb{E}}_{s_2 \sim P(\cdot)}[f(s_2)] \\
&= B_{\widetilde{d}} W\left(\mathcal{P}(\cdot|s_1,a), \mathcal{P}(\cdot|s_2,a)\right) \\
&\leqslant B_{\widetilde{d}} K_{\mathcal{P}} d_{\mathcal{S}}(s_1, s_2)
\end{aligned}$$

Then using the fixed point property of all bisimulation metrics,

$$\begin{aligned}
\widetilde{d}(s_1, s_2) &= \max_{a \in \mathcal{A}} (1-\gamma) \left|\mathcal{R}(s_1,a) - \mathcal{R}(s_2,a)\right| + \gamma T_{\widetilde{d}}(\mathcal{P}(\cdot|s_1,a), \mathcal{P}(\cdot|s_2,a)) \\
&\leqslant (1-\gamma) K_{\mathcal{R}} d_{\mathcal{S}}(s_1, s_2) + \gamma B_{\widetilde{d}} K_{\mathcal{P}} d_{\mathcal{S}}(s_1, s_2)
\end{aligned}$$

by letting s_1^*, s_2^* be the states where the supremum of $\frac{\widetilde{d}(s_1^*,s_2^*)}{d_{\mathcal{S}}(s_1^*,s_2^*)}$ is attained (if the supremum is not attained anywhere, let these be limiting points). We can conclude the proof with:

$$\begin{aligned}
B_{\widetilde{d}} = \frac{\widetilde{d}(s_1^*, s_2^*)}{d_{\mathcal{S}}(s_1^*, s_2^*)} &\leqslant \frac{(1-\gamma) K_{\mathcal{R}} d_{\mathcal{S}}(s_1^*, s_2^*) + \gamma B_{\widetilde{d}} K_{\mathcal{P}} d_{\mathcal{S}}(s_1^*, s_2^*)}{d_{\mathcal{S}}(s_1^*, s_2^*)} \\
&\leqslant (1-\gamma) K_{\mathcal{R}} + \gamma B_{\widetilde{d}} K_{\mathcal{P}}
\end{aligned}$$

We can now derive the upper bound $B_{\widetilde{d}} \leqslant \frac{(1-\gamma)K_{\mathcal{R}}}{1-\gamma K_{\mathcal{P}}}$, and by the definition of $B_{\widetilde{d}}$, the desired result trivially follows. □

Lemma 5. *Let $\mathcal{M}$ be an MDP and $\overline{\mathcal{M}}$ be a $K_{\bar{\mathcal{R}}}$-$K_{\bar{\mathcal{P}}}$-Lipschitz MDP with an embedding function $\phi : \mathcal{S} \to \bar{\mathcal{S}}$ and global DeepMDP losses $L^{\infty}_{\mathcal{P}}$ and $L^{\infty}_{\bar{\mathcal{R}}}$.. We can extend the bisimulation metric to also measure a distance between $s \in \mathcal{S}$ and $\bar{s} \in \bar{\mathcal{S}}$ by considering an composed MDP constructed by joining $\mathcal{M}$ and $\overline{\mathcal{M}}$. When an action is taken, each state will transition according to the transition matrix of its corresponding MDP. Then the bisimulation metric between a state s and it's embedded counterpart $\phi(s)$ is bounded.*

$$\widetilde{d}(s, \phi(s)) \leqslant L^{\infty}_{\mathcal{R}} + \gamma L^{\infty}_{\mathcal{P}} \frac{K_{\bar{\mathcal{R}}}}{1 - \gamma K_{\bar{\mathcal{P}}}}$$

Proof. First, note that

$$\begin{aligned}
W_{\widetilde{d}}(\phi\mathcal{P}(\cdot|s,a), \bar{\mathcal{P}}(\cdot|\phi(s),a)) &= \sup_{f \in \mathbb{F}_{\widetilde{d}}} \mathop{\mathbb{E}}_{\bar{s}_1' \sim \phi\mathcal{P}(\cdot|s,a)}[f(\bar{s}_1')] - \mathop{\mathbb{E}}_{\bar{s}_2' \sim \bar{\mathcal{P}}(\cdot|\phi(s),a)}[f(\bar{s}_2')] \\
&\leqslant \frac{(1-\gamma)K_{\mathcal{R}}}{1-\gamma K_{\mathcal{P}}} \sup_{f \in \mathbb{F}_1} \mathop{\mathbb{E}}_{\bar{s}_1' \sim \phi\mathcal{P}(\cdot|s,a)}[f(\bar{s}_1')] - \mathop{\mathbb{E}}_{\bar{s}_2' \sim \bar{\mathcal{P}}(\cdot|\phi(s),a)}[f(\bar{s}_2')] \quad \text{(Using Theorem 4)} \\
&= \frac{(1-\gamma)K_{\mathcal{R}}}{1-\gamma K_{\mathcal{P}}} W_{\ell_2}(\mathcal{P}(\cdot|s,a), \bar{\mathcal{P}}(\cdot|\phi(s),a)) \\
&\leqslant \frac{(1-\gamma)K_{\mathcal{R}}}{1-\gamma K_{\mathcal{P}}} L^{\infty}_{\bar{\mathcal{P}}}
\end{aligned}$$

Using the triangle inequality of pseudometrics and the previous derivation:

$$\begin{aligned}
\sup_s \widetilde{d}(s, \phi(s)) &= \max_{a \in \mathcal{A}} \left((1-\gamma)\left|\mathcal{R}(s,a) - \bar{\mathcal{R}}(\phi(s),a)\right| + \gamma W_{\widetilde{d}}(\mathcal{P}(\cdot|s,a), \bar{\mathcal{P}}(\cdot|\phi(s),a))\right) \\
&\leqslant (1-\gamma) L^{\infty}_{\bar{\mathcal{R}}} + \gamma \max_{a \in \mathcal{A}} \left(W_{\widetilde{d}}(\mathcal{P}(\cdot|s,a), \phi\mathcal{P}(\cdot|s,a)) + W_{\widetilde{d}}(\phi\mathcal{P}(\cdot|s,a), \bar{\mathcal{P}}(\cdot|\phi(s),a))\right) \\
&\leqslant (1-\gamma) L^{\infty}_{\bar{\mathcal{R}}} + \gamma \frac{(1-\gamma)K_{\mathcal{R}}}{1-\gamma K_{\mathcal{P}}} L^{\infty}_{\mathcal{P}} + \gamma \max_{a \in \mathcal{A}} W_{\widetilde{d}}(\mathcal{P}(\cdot|s,a), \phi\mathcal{P}(\cdot|s,a)) \\
&\leqslant (1-\gamma) L^{\infty}_{\bar{\mathcal{R}}} + \gamma \frac{(1-\gamma)K_{\mathcal{R}}}{1-\gamma K_{\mathcal{P}}} L^{\infty}_{\mathcal{P}} + \gamma \sup_s \widetilde{d}(s', \phi(s))
\end{aligned}$$

Solving for the recurrence leads to the desired result. □

Theorem 5. *Let $\mathcal{M}$ be an MDP and $\overline{\mathcal{M}}$ be a $K_{\bar{\mathcal{R}}}$-$K_{\bar{\mathcal{P}}}$-Lipschitz DeepMDP where we assume metric between deep states is the ℓ_2 distance. Let ϕ be the embedding function and $L^{\infty}_{\bar{\mathcal{P}}}$ and $L^{\infty}_{\bar{\mathcal{R}}}$ be the global DeepMDP losses. The Bisimulation distance in $\mathcal{M}$, $\widetilde{d} : \mathcal{S} \times \mathcal{S} \to \mathbb{R}^+$ can be upperbounded by the ℓ_2 distance in the embedding and the losses in the following way:*

$$\widetilde{d}(s_1, s_2) \leqslant 2\left(L^{\infty}_{\bar{\mathcal{R}}} + \gamma L^{\infty}_{\bar{\mathcal{P}}} \frac{K_{\bar{\mathcal{R}}}}{1 - \gamma K_{\bar{\mathcal{P}}}}\right) + \frac{(1-\gamma)K_{\mathcal{R}}}{1 - \gamma K_{\bar{\mathcal{P}}}} \left\|\phi(s_1) - \phi(s_2)\right\|_2$$

Proof.

$$\begin{aligned}
\widetilde{d}(s_1, s_2) &\leqslant \widetilde{d}(s_1, \phi(s_1)) + \widetilde{d}(s_2, \phi(s_2)) + \widetilde{d}(\phi(s_1), \phi(s_2)) \\
&\leqslant 2\left(L^{\infty}_{\bar{\mathcal{R}}} + \gamma L^{\infty}_{\bar{\mathcal{P}}} \frac{K_{\bar{\mathcal{R}}}}{1 - \gamma K_{\bar{\mathcal{P}}}}\right) + \widetilde{d}(\phi(s_1), \phi(s_2)) && \text{(Using Theorem 5)} \\
&\leqslant 2\left(L^{\infty}_{\bar{\mathcal{R}}} + \gamma L^{\infty}_{\bar{\mathcal{P}}} \frac{K_{\bar{\mathcal{R}}}}{1 - \gamma K_{\bar{\mathcal{P}}}}\right) + \frac{(1-\gamma)L^{\infty}_{\bar{\mathcal{R}}}}{1 - \gamma K_{\bar{\mathcal{P}}}} \left\|\phi(s_1) - \phi(s_2)\right\|_2 && \text{(Applying Theorem 4)}
\end{aligned}$$

Completing the proof. □

A.5. Quality of $\bar{\Pi}$

Lemma 6. *Let d_f and d_g be the metrics on the space χ, with the property that for some $\epsilon \geqslant 0$ it holds that $\forall x, y \in \chi, d_f(x, y) \leqslant \epsilon + d_g(x, y)$. Define the sets of 1-Lipschitz functions $\mathbb{F} = \{f : |f(x) - f(x)| \leqslant d_f(x, y), \forall x, y \in \chi\}$ and $\mathbb{G} = \{g : |g(x) - g(x)| \leqslant d_g(x, y), \forall x, y \in \chi\}$. Then for any $f \in \mathbb{F}$, there exists one $g \in \mathbb{G}$ such that for all $x \in \chi$,*

$$|f(x) - g(x)| \leqslant \frac{\epsilon}{2}$$

Proof. Define the set $\mathbb{Z} = \{z : |z(x) - z(y)| \leqslant \epsilon + d_g(x, y), \forall x, y \in \chi\}$. Then trivially, any function $f \in \mathbb{F}$ is also a member of $\mathbb{Z}$. We now show that the set $\mathbb{Z}$ can equivalently be expressed as $z(x) = g(x) + u(x)$, where $g \in \mathbb{G}$ and $u(x) \in (\frac{-\epsilon}{2}, \frac{\epsilon}{2})$, is (non Lipschitz) bounded function.

$$\begin{aligned}
|z(x) - z(y)| &= |g(x) + u(x) - g(y) - u(y)| \\
&\leqslant |g(x) - g(y)| + |u(x) - u(y)| \\
&\leqslant d_g(x, y) + \epsilon
\end{aligned}$$

Note how both inequalities are tight (there is a g and u for which the equality holds), together with the fact that the set $\mathbb{Z}$ is convex, it follows that any $z \in \mathbb{Z}$ must be expressible as $g(x) + u(x)$.

We now complete the proof. For any $z \in \mathbb{Z}$, there exist a $g \in \mathbb{G}$ s.t. $z(x) = g(x) + u(x)$. Then:

$$|z(x) - g(x)| = |u(x)| \leqslant \frac{\epsilon}{2}$$

□

Theorem 4. *Let $\mathcal{M}$ be an MDP and $\overline{\mathcal{M}}$ be a ($K_{\bar{\mathcal{R}}}$, $K_{\bar{\mathcal{P}}}$)-Lipschitz DeepMDP, with an embedding function ϕ, and global loss functions $L^{\infty}_{\bar{\mathcal{R}}}$ and $L^{\infty}_{\bar{\mathcal{P}}}$. Denote by $\bar{\Pi}_K$ the set of K-**Lipschitz deep policies** $\{\bar{\pi} : \bar{\pi} \in \bar{\Pi}, |\bar{\pi}(a|s_1) - \bar{\pi}(a|s_2)| \leqslant K \left\|\phi(s_1) - \phi(s_2)\right\|_2, \forall s_1, s_2 \in \mathcal{S}, a \in \mathcal{A}\}$. Finally define the constant $C = \frac{(1-\gamma)K_{\mathcal{R}}}{1-\gamma K_{\bar{\mathcal{P}}}}$. Then for any $\widetilde{\pi} \in \widetilde{\Pi}_K$ there exists a $\bar{\pi} \in \bar{\Pi}_{CK}$ which is close to $\widetilde{\pi}$ in the sense that, for all $s \in \mathcal{S}$ and $a \in \mathcal{A}$,*

$$|\widetilde{\pi}(a|s) - \bar{\pi}(a|s)| \leqslant L^{\infty}_{\bar{\mathcal{R}}} + \gamma L^{\infty}_{\bar{\mathcal{P}}} \frac{K_{\bar{\mathcal{R}}}}{1 - \gamma K_{\bar{\mathcal{P}}}}$$

Proof. The proof is based on Lemma 6. Let $\chi = \mathcal{S}$, $d_f(x, y) = K\widetilde{d}(x, y)$, $d_g(x, y) = KC \left\|\phi(x) - \phi(y)\right\|_2$ and $\epsilon = 2\left(L^{\infty}_{\bar{\mathcal{R}}} + \gamma L^{\infty}_{\bar{\mathcal{P}}} \frac{K_{\bar{\mathcal{R}}}}{1-\gamma K_{\bar{\mathcal{P}}}}\right)$. Theorem 5 can be used to show that the condition $d_f(x, y) \leqslant \epsilon + d_g(x, y)$ holds. Then the application of Lemma 6 provides the desired result. □

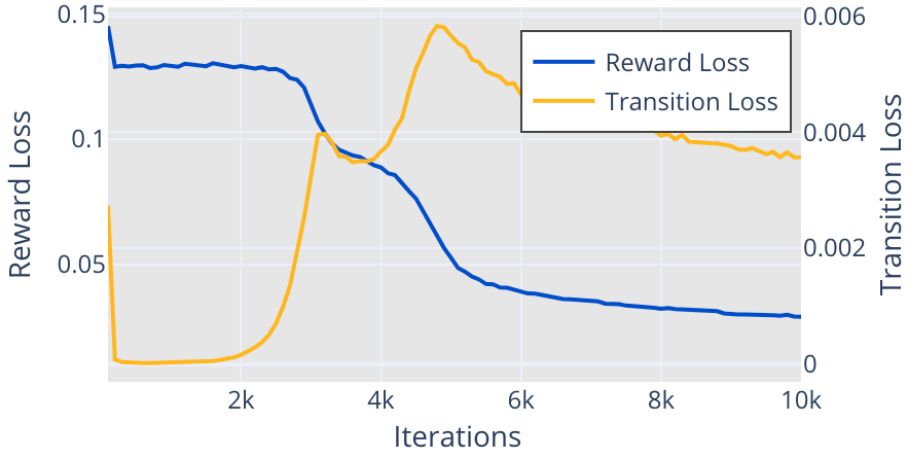

Figure 4. Due to the competition between reward and transition losses, the optimization procedure spends significant time in local minima early on in training. It eventually learns a good representation, which it then optimizes further.

B. DonutWorld Experiments

B.1. Environment Specification

Our synthetic environment, DonutWorld, consists of an agent moving around a circular track. The environment is centered at (0,0), and includes the set of points whose distance to the center is between 3 and 6 units away; all other points are out-of-bounds. The distance the agent can move on each timestep is equal to the distance to the nearest out-of-bounds point, capped at 1. We refer to the regions of space where the agent's movements are fastest (between 4 and 5 units away from the origin) as the "track," and other in-bounds locations as "grass". Observations are given in the form of 32-by-32 black-and-white pixel arrays, where the agent is represented by a white pixel, the track by luminance 0.75, the grass by luminance 0.25, and out-of-bounds by black. The actions are given as pairs of numbers in the range (-1,1), representing an unnormalized directional vector. The reward for each transition is given by the number of radians moved clockwise around the center.

Another variant of this environment involves four copies of the track, all adjacent to one another. The agent is randomly placed onto one of the four tracks, and cannot move between them. Note that the value function for any policy is identical whether the agent is on the one-track DonutWorld or the four-track DonutWorld. Observations for the four-track DonutWorld are 64-by-64 pixel arrays.

B.2. Architecture Details

We learn a DeepMDP on states and actions from a uniform distribution over all possible state-action pairs. The environment can be fully represented by a latent space of size two, so that is the dimensionality used for latent states of the DeepMDP.

We use a convolutional neural net for our embedding function ϕ, which contains three convolutional layers followed by a linear transformation. Each convolutional layer uses 4x4 convolutional filters with stride of 2, and depths are mapped to 2, then 4, then 8; the final linear transformation maps it to the size of the latent state, 2. ReLU nonlinearities are used between each layer, and a sigmoid is applied to the output to constrain the space of latent states to be bounded by (0, 1).

The transition function and reward function are each represented by feed-forward neural networks, using 2 hidden layers of size 32 with ReLU nonlinearities. A sigmoid is applied output of the transition function.

For the autoencoder baseline, we use the same architecture for the encoder as was used for the embedding function. Our decoder is a three-layer feedforward ReLU network with 32 hidden units per layer. The reconstruction loss is a softmax cross-entropy over possible agent locations.

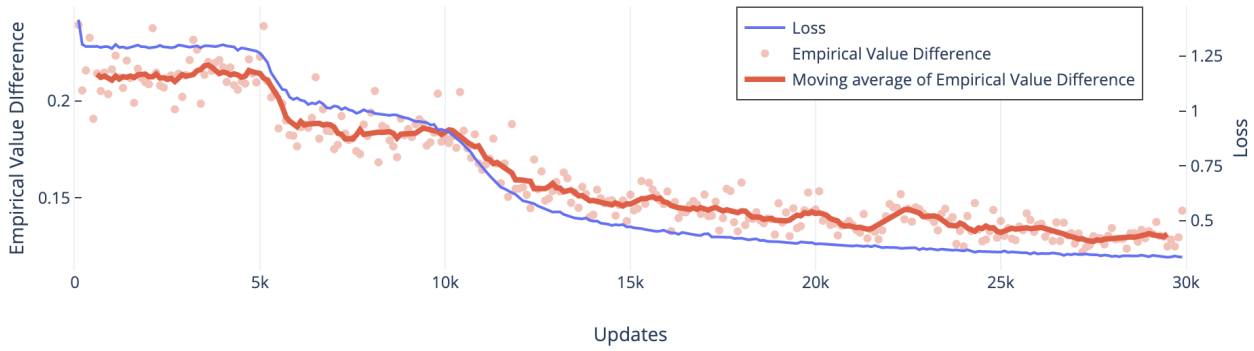

Figure 5. Plot of training curves obtained by learning a DeepMDP on our toy environment. Our objective minimizes both the theoretical upper bound of value difference and the empirical value difference.

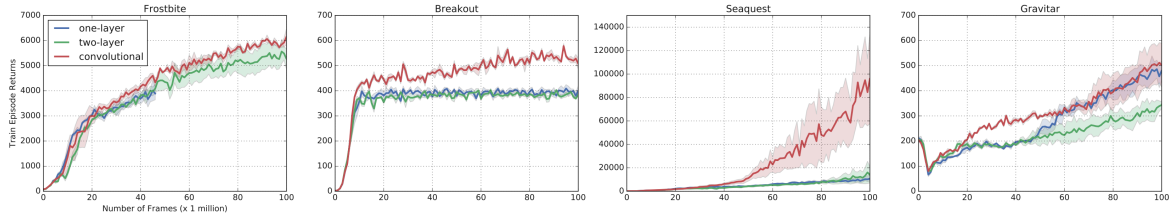

Figure 6. Performance of C51 with model-based auxiliary objectives. Three types of transition models are used for predicting next latent states: a single convolutional layer (convolutional), a single fully-connected layer (one-layer), and a two-layer fully-connected network (two-layer).

B.3. Hyperparameters

All models were implemented in Tensorflow. We use an Adam Optimizer with a learning rate of 3e-4, and default settings. We train for 30,000 steps. The batch size is 256 for DMDPs and 1024 for autoencoders. The discount factor, γ, is set to 0.9, and the coefficient for the gradient penalty, λ, is set to 0.01. In contrast to the gradient penalty described in Gulrajani et al. (2017b), which uses its gradient penalty to encourage all gradient norms to be close to 1, we encourage all gradient norms to be close to 0. Our sampling distribution is the same as our training distribution, simply the distribution of states sampled from the environment.

B.4. Empirical Value Difference

Figure 5 shows the loss curves for our learning procedure. We randomly sample trajectories of length 1000, and compute both the empirical reward in the real environment and the reward approximated by performing the same actions in the DeepMDP; this allows us to compute the empirical value error. These results demonstrate that neural optimization techniques are capable of learning DeepMDPs, and that this optimization procedure, designed to tighten theoretical bounds, is minimized by a good model of the environment, as reflected in improved empirical outcomes.

C. Atari 2600 Experiments

C.1. Hyperparameters

For all experiments we use an Adam Optimizer with a learning rate of 0.00025 and epsilon of 0.0003125. We linearly decay epsilon from 1.0 to 0.01 over 1000000 training steps. We use a replay memory of size 1000000 (it must reach a minimum size of 50000 prior to sampling transitions for training). Unless otherwise specified, the batch size is 32. For additional hyperparameter details, see Table 1 and (Bellemare et al., 2017a).

Hyperparameter	Value
Runner.sticky_actions	No Sticky Actions
Runner.num_iterations	200
Runner.training_steps	250000
Runner.evaluation_steps	Eval phase not used.
Runner.max_steps_per_episode	27000
WrappedPrioritizedReplayBuffer.replay_capacity	1000000
WrappedPrioritizedReplayBuffer.batch_size	32
RainbowAgent.num_atoms	51
RainbowAgent.vmax	10.
RainbowAgent.update_horizon	1
RainbowAgent.min_replay_history	50000
RainbowAgent.update_period	4
RainbowAgent.target_update_period	10000
RainbowAgent.epsilon_train	0.01
RainbowAgent.epsilon_eval	0.001
RainbowAgent.epsilon_decay_period	100000
RainbowAgent.replay_scheme	'uniform'
RainbowAgent.tf_device	'/gpu:0'
RainbowAgent.optimizer	@tf.train.AdamOptimizer()
tf.train.AdamOptimizer.learning_rate	0.00025
tf.train.AdamOptimizer.epsilon	0.0003125
ModelRainbowAgent.reward_loss_weight	1.0
ModelRainbowAgent.transition_loss_weight	1.0
ModelRainbowAgent.transition_model_type	'convolutional'
ModelRainbowAgent.embedding_type	'conv_layer_embedding'

Table 1. Configurations for the DeepMDP and C51 agents used with Dopamine (Castro et al., 2018) in Section 7.3. Note that the DeepMDP is referred to as ModelRainbowAgent in the configs.

C.2. Architecture Search

In this section, we aim to answer: what latent state space and transition model architecture lead to the best Atari 2600 performance of the C51 DeepMDP? We begin by jointly determining the form of $\bar{\mathcal{S}}$ and $\theta_{\bar{\mathcal{P}}}$ which are conducive to learning a DeepMDP on Atari 2600 games. We employ three latent transition model architectures: (1) single fully connected layer, (2) two-layer fully-connected network, and (3) single convolutional layer. The fully-connected transition networks use the 512-dimensional output of the embedding network's penultimate layer as the latent state, while the convolutional transition model uses the $11 \times 11 \times 64$ output of the embedding network's final convolutional layer. Empirically, we find that the use of a convolutional transition model on the final convolutional layer's output outperforms the other architectures, as shown in Figure 6.

C.3. Architecture Details

The architectures of various components are described below. A conv layer refers to a 2D convolutional layer with a specified stride, kernel size, and number of outputs. A deconv layer refers to a deconvolutional layer. The padding for conv and deconv layers is such that the output layer has the same dimensionality as the input. A maxpool layer performs max-pooling on a 2D input and fully connected refers to a fully-connected layer.

C.3.1. ENCODER

In the main text, the encoder is referred to as $\phi : \mathcal{S} \to \bar{\mathcal{S}}$ and is parameterized by θ_e. The encoder architecture is as follows:

Input: observation s which has shape: batch size $\times\ 84 \times 84 \times 4$. The Atari 2600 frames are 84×84 and there are 4 stacked frames given as input. The frames are pre-processed by dividing by the maximum pixel value, 255. Output: latent state $\phi(s)$

In Appendix C.2, we experimented with two different latent state representations. (1) *ConvLayer*: The latent state is the

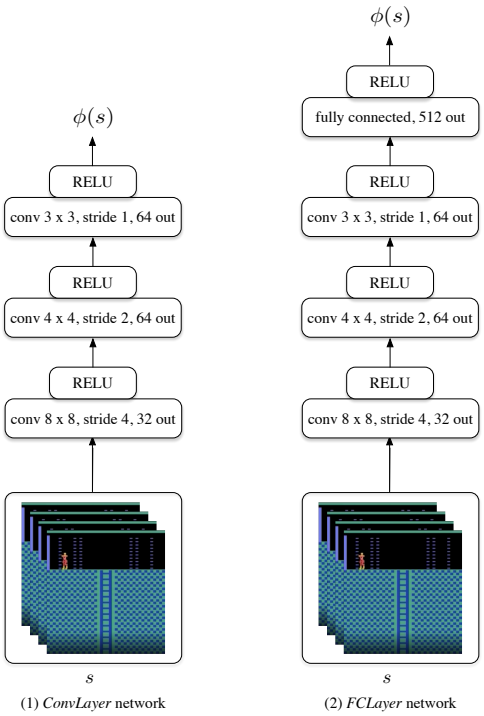

Figure 7. Encoder architectures used for the DeepMDP agent.

output of the final convolutional layer, or (2) *FCLayer*: the latent state is the output of a fully-connected (FC) layer following the final convolutional layer. These possibilities for the encoder architecture are described in Figure 7.

In sections 7.3, 7.4, C.4, and C.5 the latent state of type ConvLayer is used: $11 \times 11 \times 64$ outputs of the final convolutional layer.

C.3.2. Latent Transition Model

In Appendix C.2 there are three types of latent transition models $\bar{\mathcal{P}} : \bar{\mathcal{S}} \to \bar{\mathcal{S}}$ parameterized by $\theta_{\bar{\mathcal{P}}}$ which are evaluated: (1) a single fully-connected layer, (2) a two-layer fully-connected network, and (3) a single convolutional layer (see Figure 8). Note that the first two types of transition models operate on the flattened 512-dimensional latent state (FCLayer), while the convolutional transition model receives as input the $11 \times 11 \times 64$ latent state type ConvLayer. For each transition model, num_actions predictions are made: one for each action conditioned on the current latent state $\phi(s)$.

In sections 7.3, 7.4, C.4, and C.5 the convolutional transition model is used.

C.3.3. Reward Model and C51 Logits Network

The architectures of the reward model $\bar{\mathcal{R}}$ parameterized by $\theta_{\bar{\mathcal{R}}}$ and C51 logits network parameterized by θ_Z depend the latent state representation. See Figure 9 for these architectures. For each architecture type, num_actions predictions are made: one for each action conditioned on the current latent state $\phi(s)$.

In sections 7.3, 7.4, C.4, and C.5 two-layer fully-connected networks are used for the reward and C51 logits networks.

C.3.4. Observation Reconstruction and Next Observation Prediction

The models for observation reconstruction and next observation prediction in Section 7.4 are deconvolutional networks based on the architecture of the embedding function ϕ. Both operate on latent states of type ConvLayer. The architectures are described in Figure 10.

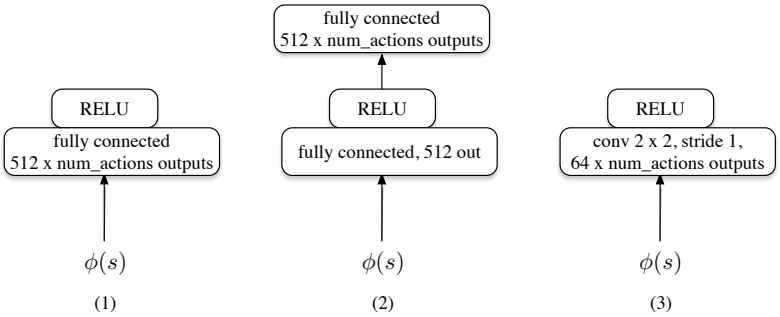

Figure 8. Transition model architectures used for the DeepMDP agent: (1) a single fully-connected layer (used with latent states of type FCLayer), (2) a two-layer fully-connected network (used with latent states of type FCLayer), and (3) a single convolutional layer (used with latent states of type ConvLayer).

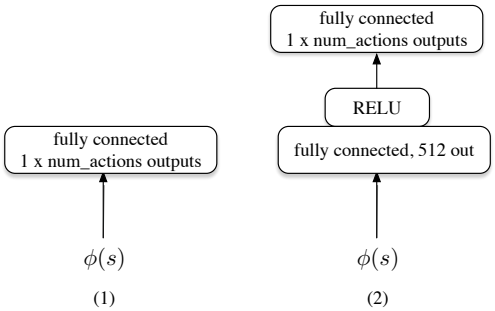

Figure 9. Reward and C51 Logits network architectures used for the DeepMDP agent: (1) a single fully-connected layer (used with latent states of type FCLayer), (2) a two-layer fully-connected network (used with latent states of type ConvLayer).

C.4. DeepMDP Auxiliary Tasks: Different Weightings on DeepMDP Losses

In this section, we discuss results of a set of experiments where we use a convolutional latent transition model and a two-layer reward model to form auxiliary task objectives on top of a C51 agent. In these experiments, we use different weightings in the set $\{0, 1\}$ for the transition loss and for the reward loss. The network architecture is based on the best performing DeepMDP architecture in Appendix C.2. Our results show that using the transition loss is enough to match performance of using both the transition and reward loss. In fact, on Seaquest, using only the reward loss as an auxiliary tasks causes performance to crash. See Figure 11 for the results.

C.5. Representation Learning with DeepMDP Objectives

Given performance improvements in the auxiliary task setting, a natural question is whether optimization of the deepMDP losses is sufficient to perform model-free RL. To address this question, we learn θ_e only via minimizing the reward and latent transition losses. We then learn θ_Z by minimizing the C51 loss but do not pass gradients through θ_e. As a baseline, we minimize the C51 loss with randomly initialized θ_e and do not update θ_e. In order to successfully predict terminal transitions and rewards, we add a terminal reward loss and a terminal state transition loss. The terminal reward loss is a Huber loss between $\bar{\mathcal{R}}(\phi(s_T))$ and 0, where s_T is a terminal state. The terminal transition loss is a Huber loss between $\bar{\mathcal{P}}(s, a)$ and $\mathbf{0}$, where s is either a terminal state or a state immediately preceding a terminal state and $\mathbf{0}$ is the zero latent state.

We find that in practice, minimizing the latent transition loss causes the latent states to collapse to $\phi(s) = 0 \; \forall s \in S$. As (Francois-Lavet et al., 2018) notes, if only the latent transition loss was minimized, then the optimal solution is indeed $\phi : S \to 0$ so that $\bar{\mathcal{P}}$ perfectly predicts $\phi(\mathcal{P}(s, a))$.

We hope to mitigate representation collapse by augmenting the influence of the reward loss. We increase the batch size from 32 to 100 to acquire greater diversity of rewards in each batch sampled from the replay buffer. However, we find that only after introducing a state reconstruction loss do we obtain performance levels on par with our simple baseline. These

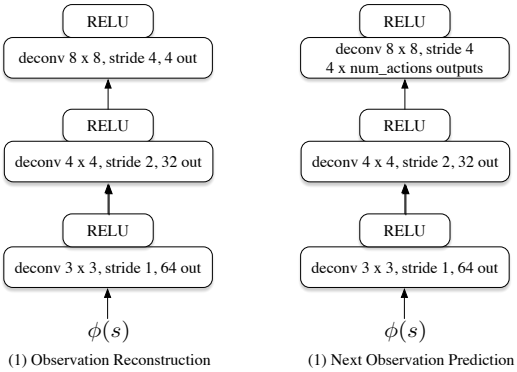

Figure 10. Architectures used for observation reconstruction and next observation prediciton. Both networks take latent states of type ConvLayer as input.

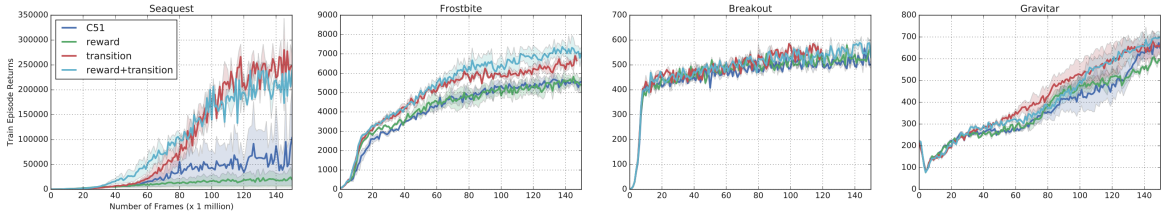

Figure 11. We compare C51 with C51 with DeepMDP auxiliary task losses. The combinations of loss weightings are $\{0, 1\}$ (just reward), $\{1, 0\}$ (just transition), and $\{1, 1\}$ (reward+transition), where the first number is the weight for the transition loss and the second number is the weight for the reward loss.

results (see Figure 12) indicate that in more complex environments, additional work is required to successfully balance the minimization of the transition loss and the reward loss, as the transition loss seems to dominate.

This finding was surprising, since we were able to train a DeepMDP on the DonutWorld environment with no reconstruction loss. Further investigation of the DonutWorld experiments shows that the DeepMDP optimization procedure seems to be highly prone to becoming trapped in local minima. The reward loss encourages latent states to be informative, but the transition loss counteracts this, preferring latent states which are uninformative and thus easily predictable. Looking at the relative reward and transition losses in early phases of training in Figure 4, we see this dynamic clearly. At the start of training, the transition loss quickly forces latent states to be near-zero, resulting in very high reward loss. Eventually, on this simple task, the model is able to escape this local minimum by "discovering" a representation that is both informative and predictable. However, as the difficulty of a task scales up, it becomes increasingly difficult to discover a representation which

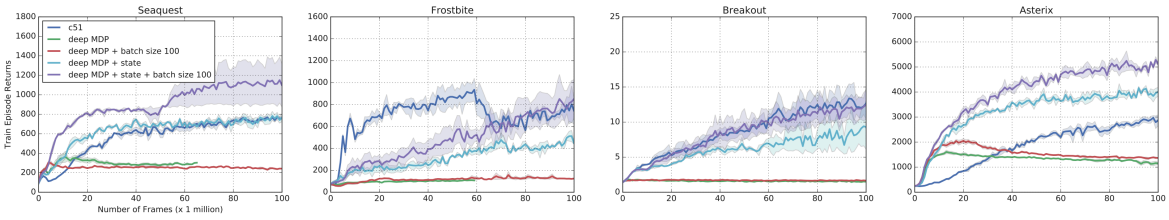

Figure 12. We evaluate the performance of C51 when learning the latent state representation only via minimizing deepMDP objectives. We compare learning the latent state representation with the deepMDP objectives (deep MDP), deepMDP objectives with larger batch sizes (deepMDP + batch size 100), deepMDP objectives and an observation reconstruction loss (deepMDP + state), and deepMDP with both a reconstruction loss and larger batch size (deepMDP + state + batch size 100). As a baselines, we compare to C51 on a random latent state representation (C51).

escapes these local minima by explaining the underlying dynamics of the environment well. This explains our observations on the Arcade Learning Environment; the additional supervision from the reconstruction loss helps guide the algorithm towards representations which explain the environment well.

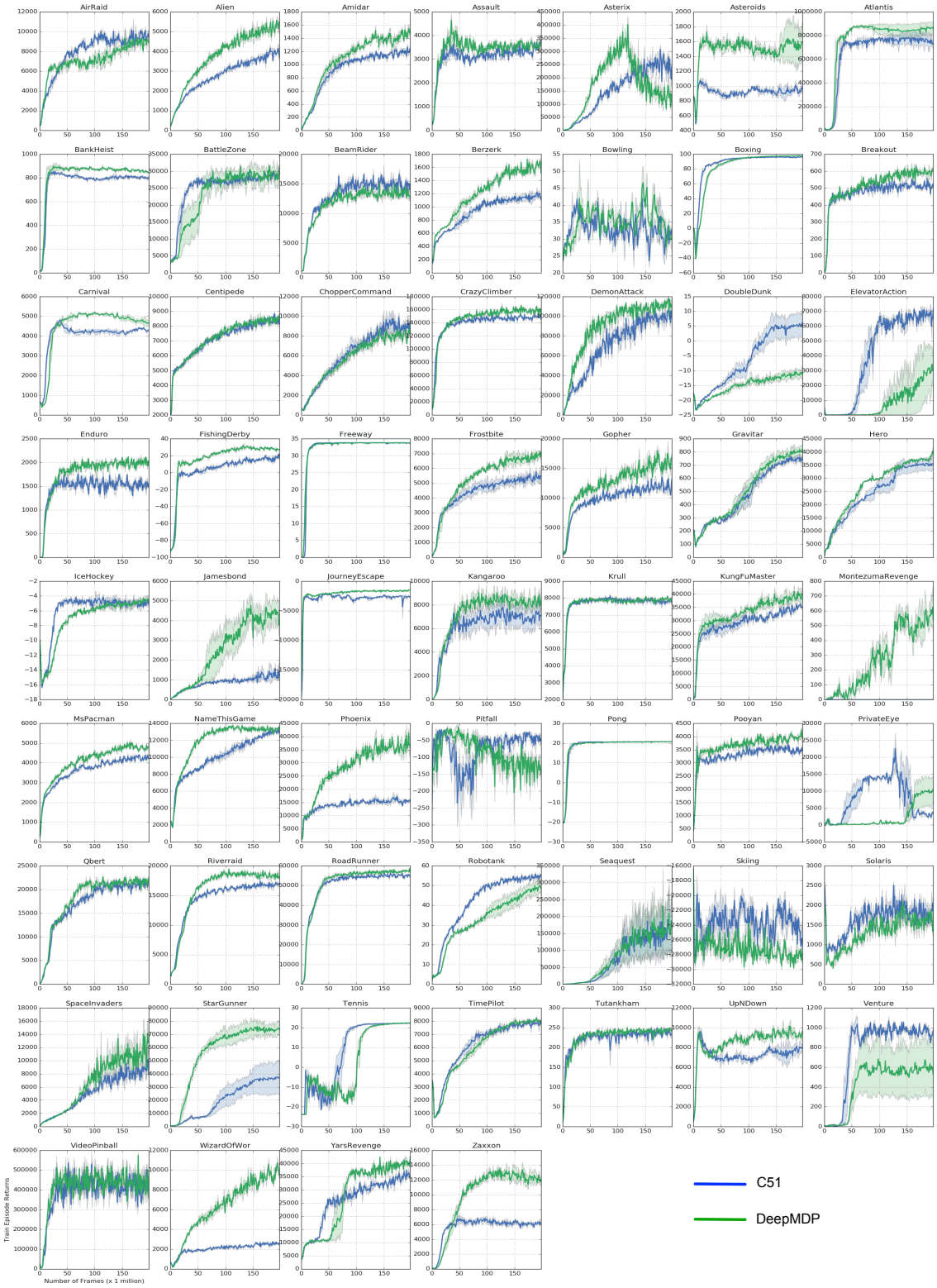

Figure 13. Learning curves of C51 and C51 + DeepMDP auxiliary task objectives (labeled DeepMDP) on Atari 2600 games.

Game Name	C51	DeepMDP
AirRaid	**11544.2**	10274.2
Alien	4338.3	**6160.7**
Amidar	1304.7	**1663.8**
Assault	4133.4	**5026.2**
Asterix	343210.0	**452712.7**
Asteroids	1125.4	**1981.7**
Atlantis	844063.3	**906196.7**
BankHeist	861.3	**937.0**
BattleZone	31078.2	**34310.2**
BeamRider	**19081.0**	16216.8
Berzerk	1250.9	**1799.9**
Bowling	51.4	**56.3**
Boxing	97.3	**98.2**
Breakout	584.1	**672.8**
Carnival	4877.3	**5319.8**
Centipede	**9092.1**	9060.9
ChopperCommand	**10558.8**	9895.7
CrazyClimber	158427.7	**173043.1**
DemonAttack	111697.7	**119224.7**
DoubleDunk	**6.7**	-9.3
ElevatorAction	**73943.3**	37854.4
Enduro	1905.3	**2197.8**
FishingDerby	25.4	**33.9**
Freeway	**33.9**	33.9
Frostbite	5882.9	**7367.3**
Gopher	15214.3	**21017.2**
Gravitar	790.4	**838.3**
Hero	36420.7	**40563.1**
IceHockey	**-3.5**	-4.1
Jamesbond	1776.7	**5181.1**
JourneyEscape	-1856.1	**-1337.1**
Kangaroo	8815.5	**9714.9**
Krull	8201.5	**8246.9**
KungFuMaster	37956.5	**42692.7**
MontezumaRevenge	14.7	**770.7**
MsPacman	4597.8	**5282.5**
NameThisGame	13738.7	**14064.6**
Phoenix	20216.7	**45565.1**
Pitfall	-9.8	**-0.8**
Pong	**20.8**	20.8
Pooyan	4052.7	**4431.1**
PrivateEye	**28694.0**	11223.8
Qbert	23268.6	**23538.7**
Riverraid	17845.1	**19934.7**
RoadRunner	57638.5	**59152.2**
Robotank	**57.4**	51.3
Seaquest	226264.0	**230881.6**
Skiing	**-15454.8**	-16478.0
Solaris	**2876.7**	2506.8
SpaceInvaders	12145.8	**16461.2**
StarGunner	38928.7	**78847.6**
Tennis	22.6	**22.7**
TimePilot	8340.7	**8345.6**
Tutankham	**259.3**	256.9
UpNDown	10175.5	**10930.6**
Venture	**1190.1**	755.4
VideoPinball	**668415.7**	633848.8
WizardOfWor	2926.0	**11846.1**
YarsRevenge	39502.9	**44317.8**
Zaxxon	7436.5	**14723.0**

Table 2. DeepMDP versus C51 returns. For both agents, we report the max average score achieved across all training iterations (each training iteration is 1 million frames).