6 APPENDIX

6.1 NETWORK SPECIFICATIONS, HYPERPARAMETERS

The observation encoder and decoder architectures are taken from Ha & Schmidhuber (2018), the representation and transition functions are jointly implemented using an RSSM Hafner et al. (2019b), reward predictor, policy and value are implemented as 3-layer MLPs with 200 hidden units and ELU activations Clevert et al. (2015). The stochastic component of the hidden state is modeled as a 30-dimensional diagonalized Gaussian, the deterministic component has 200 components. The policy outputs actions parametrized by a tanh mean scaled by a factor of 5, the standard deviation is computed as a softplus deviation of a Gaussian distribution and then transformed via a tanh Hou et al. (2020)

We draw 50 batches of length 50 on each training iteration. The model, policy and value components are trained using Adam Kingma & Ba (2014), with learning rates 6×10^{-4}, 8×10^{-5}, 8×10^{-5}, respectively, and gradient normalization set to 100. On experiments with contrastive augmentation, the momentum parameter on the contrastive representation was set to $\alpha = 0.95$. Images from the environment were rendered at 80×80 pixels and were cropped down to 64×64 (images are rendered at 64×64 if no contrastive augmentation is used). We use a latent imagination horizon of $H = 15$, the parameters for the discounted V_λ targets where $\gamma = 0.99$, $\lambda = 0.95$. We tested $\lambda_{\ell_1} \in [1.0, 0.1, 0.01]$ and $\lambda_C \in [1.0, 0.1, 0.01]$, with the best performing pair across all experiment variants being $\lambda_{\ell_1} = \lambda_C 0.1$.

The prioritized episodic replay buffer is initialized with 5 seed episodes using random actions. The collection interval was set to $C = 100$. Exploration noise was drawn from a $\mathcal{N}(0, 0.3)$ distribution. Action repeat for all environments were set to 2.

6.2 BISIMULATION DISTANCE

We recall from Ferns & Precup (2014) that for any two states s_i, s_j, and for any $c = [0, 1)$, we can find a bisimulation pseudometric that obeys

$$d(s_i, s_j) = \max_{a \in \mathcal{A}} (1 - c)||r(s_i) - r(s_j)||_1^1 + c\mathbb{W}_1(p(s_{i+1} \mid a, s_i), p(s_{j+1} \mid a, s_j), d)$$

Where $r(s_i)$ is the expected reward of state s_i, $p(s_{i+1} \mid a, s_i)$ is the state transition function, and $\mathbb{W}_1 = \inf_{\gamma' \in \Gamma(P \times P)} \int_{S \times S} d(s_i, s_j) \mathbf{d}\gamma'(s_i, s_j)$ is the Wasserstein-1 metric.

To implement bisimulation metric learning in imagined trajectories, we defined a bisimulation policy and value function $a \sim \pi_b(s_i, s_j)$, $V_b(|s_i - s_j|)$ that operate on state pairs. For any pair of states s_i, s_j and action $a \sim \pi_b(s_i, s_j)$, we compute the future state distribution $p(s_{i+1} \mid a, s_i) \times p(s_{j+1} \mid a, s_j)$ from the model. the bisimulation policy and value models are trained on the following objectives

$$\begin{array}{ll} q_b(s_i, s_j, a) & = (1 - c)||r(s_i) - r(s_j)||_1^1 + c\mathbb{W}_2(p(s_{i+1} \mid a, s_i), p(s_{j+1} \mid a, s_j), L^2) \\ \mathcal{L}_{\pi_b}(s_i, s_j) & \qquad\qquad\qquad\qquad\qquad = q_b(s_i, s_j, a) \\ \mathcal{L}_{V_b}(s_i, s_j) & \qquad\qquad\qquad = ||V_b(|s_i - s_j|) - q_b(s_i, s_j, a)||_1^1 \end{array}$$

That is, we relax the Wasserstein-1 distance to Wasserstein-2, similarliy to Zhang et al. (2020), since it has a closed form formula for diagonal Gaussian distributions, we train a specialized policy to maximize this distance in one-step imagination, and we train a value model so that the bisimulation metric is deducible based on the absolute distance between states $|s_i - s_j|$. The gradients of the bisimulation value model are propagated to the model parameters, with the intention of making distance between latent states informative w.r.t. differences in future states and rewards.

The architecture for the bisimulation policy and value function are identical to the standard policy and value models, we chose $c = 0.5$ to compute the model losses, state pairs were chosen by randomly pairing latent states in the 50×50 training batch. This technique did not improve sample efficiency, but may be of use in situations similar to the ones studied in Zhang et al. (2020). Results on the same 8 environments used for the ablation study are presented in table 3

6.3 EXPLORATION VIA LATENT DISAGREEMENT

We follow the intrinsic reward by latent disagreement proposed in Sekar et al. (2020) and use this intrinsic model uncertainty as a reward bonus for our policy. The reasoning behind this was to allow the data collection process to collect experience that was near optimality in terms of true environment reward, but tilted towards model refinement (by acquiring samples where the model is uncertain). We added this intrinsic reward with a 0.1 scaling factor. Results on the same 8 environments used for the ablation study are presented in table 3. This technique did not increase sample efficiency

Table 3: Episodic reward average for Dreamer, ReaPER, Bisimulation (Bisim), Exploration via latent disagreement (ExP), and the intermediary components of ReaPER as a function of environment steps. Rewards are averaged across the cartpole balance, cartpole swingup, reacher easy, cup catch, finger spin, walker walk, walker run and cheetah run environments in DMControl. ReaPER consistently outperforms the other options.

Steps	Dreamer	Bisim	Contrast	L1	L1Contrast	PER	Exp	ExpReaPER	ReaPER
$100K$	309	280	349	358	262	263	346	353	**374**
$200K$	549	510	663	563	619	452	581	628	**670**
$300K$	636	590	714	693	691	562	689	743	**756**
$400K$	682	621	732	752	758	595	724	769	**780**
$500K$	760	684	760	774	741	611	740	770	**787**