Supplementary material for: TibGM: A Transferable and Information-Based Graphical Model Approach for Reinforcement Learning

Tameem Adel [1] **Adrian Weller** [1 2]

7. Implementation Details

Whenever feasible, and for the sake of comparing on common grounds, we have tried to imitate the hyperparameter settings of (Haarnoja et al., 2018b) and (Haarnoja et al., 2018a). After each learning iteration of the `TibGM` stochastic policy, the learnt policy is embedded along with the dynamics. The dynamics are sampled from the sample-efficient off-policy soft actor-critic (SAC Haarnoja et al., 2018b). Hyperparameter values are displayed in Tables 2 and 3.

Table 2. Shared hyperparameter values.

Parameter	Value
replay buffer size	10^6
Training began after collecting:	2,000 samples
encoder	2-hidden layer NN with 512 ReLUs, each
decoder	2-hidden layer NN with 512 ReLUs, each
dimension of $\mathbf{z}$	256
dimension of $\mathbf{h_t}$	128
optimizer	Adam (Kingma & Ba, 2015)
learning rate	$3 \cdot 10^{-4}$
batch size	128
discount	0.99

Table 3. Benchmark specific hyperparameter values.

Parameter	Swimmer	Hopper	Walker2d	HalfCheetah	Ant	Humanoid
reward scale	100	1	3	1	3	3
action dimension	2	3	6	6	8	21
state dimension	4	6	12	12	16	42

Regarding normalizing flows (NFs), in addition to Sylvester normalizing flows (SNFs, van den Berg et al., 2018), we performed some initial experiments using alternative NF formulations, namely vanilla planar NFs, radial NFs and inverse auto-regressive NFs. SNFs considerably outperformed all the other types of NFs.

In Section 5.1 and Figure 2 of the main document, we showed how `ExTibGM` and `TibGM` achieve better results than the baselines based on the number of time steps (the x-axis). Comparisons in terms of wall-clock run-time lead to very

[1]Department of Engineering, University of Cambridge, UK [2]The Alan Turing Institute, UK. Correspondence to: Tameem Adel <tah47@cam.ac.uk>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

similar results. In Table 4, we show a highlight of these comparisons with LSP (most relevant since it is also latent-space, PGM-based) by listing the return values after a specific number of minutes (100 minutes):

Table 4. Total expected return on 6 benchmark tasks after 100 wall-clock minutes.

Task / Algorithm	**`TibGM`**	**`ExTibGM`**	LSP
Swimmer	1019	1542	351
Hopper	2217	2922	2178
Walker2d	4910	4869	3184
HalfCheetah	15290	16187	12093
Ant	5109	6055	3921
Humanoid	7823	8318	5596

Overall, `ExTibGM` and `TibGM` do significantly better (in 6 and 5 tasks, respectively).

References

Haarnoja, T., Hartikainen, K., Abbeel, P., and Levine, S. Latent space policies for hierarchical reinforcement learning. *International Conference on Machine Learning (ICML)*, 2018a.

Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *International Conference on Machine Learning (ICML)*, 2018b.

Kingma, D. and Ba, J. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations (ICLR)*, 2015.

van den Berg, R., Hasenclever, L., Tomczak, J., and Welling, M. Sylvester normalizing flows for variational inference. *Uncertainty in Aritifical Intelligence (UAI)*, 2018.