Jazwinski, A. Stochastic process and filtering theory, academic press. *A subsidiary of Harcourt Brace Jovanovich Publishers*, 1970.

Kamalapurkar, R., Klotz, J. R., Walters, P., and Dixon, W. E. Model-based reinforcement learning in differential graphical games. *IEEE Transactions on Control of Network Systems*, 5(1):423–433, 2016.

Karatzas, I. and Shreve, S. Brownian motion and stochastic calculus (graduate texts in mathematics), 1991.

Kushner, H. Numerical approximations for stochastic differential games. *SIAM J. Control Optim.*, 41:457–486, 2002.

Kushner, H. and Chamberlain, S. On stochastic differential games: Sufficient conditions that a given strategy be a saddle point, and numerical procedures for the solution of the game. *Journal of Mathematical Analysis and Applications*, 26:560–575, 1969.

Liu, I.-J., Yeh, R. A., and Schwing, A. G. Pic: permutation invariant critic for multi-agent deep reinforcement learning. In *Conference on Robot Learning*, pp. 590–602. PMLR, 2020.

Lyle, C., van der Wilk, M., Kwiatkowska, M., Gal, Y., and Bloem-Reddy, B. On the benefits of invariance in neural networks. *arXiv preprint arXiv:2005.00178*, 2020.

Ma, J., Zhang, J., et al. Representation theorems for backward stochastic differential equations. *Annals of Applied probability*, 12(4):1390–1418, 2002.

Mahajan, A., Rashid, T., Samvelyan, M., and Whiteson, S. Maven: Multi-agent variational exploration. In *Advances in Neural Information Processing Systems*, pp. 7613–7624, 2019.

Mataramvura, S. and Øksendal, B. Risk minimizing portfolios and hjbi equations for stochastic differential games. *Stochastics An International Journal of Probability and Stochastic Processes*, 80(4):317–337, 2008.

Osborne, M. J. and Rubinstein, A. A course in game theory cambridge. *MA: MIT Press [Google Scholar]*, 1994.

Pereira, M. A., Wang, Z., Exarchos, I., and Theodorou, E. A. Learning deep stochastic optimal control policies using forward-backward sdes. In *Robotics: science and systems*, 2019.

Prasad, A. and Sethi, S. P. Competitive advertising under uncertainty: A stochastic differential game approach. *Journal of Optimization Theory and Applications*, 123(1): 163–185, 2004.

Raissi, M. Forward-backward stochastic neural networks: Deep learning of high-dimensional partial differential equations. *arXiv preprint arXiv:1804.07010*, 2018.

Ramachandran, K. M. and Tsokos, C. P. *Stochastic differential games. Theory and applications*, volume 2. Springer Science & Business Media, 2012.

Rashid, T., Samvelyan, M., De Witt, C. S., Farquhar, G., Foerster, J., and Whiteson, S. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. *arXiv preprint arXiv:1803.11485*, 2018.

Rozen, O., Shwartz, V., Aharoni, R., and Dagan, I. Diversify your datasets: Analyzing generalization via controlled variance in adversarial datasets. *arXiv preprint arXiv:1910.09302*, 2019.

Schmidt, L., Santurkar, S., Tsipras, D., Talwar, K., and Madry, A. Adversarially robust generalization requires more data. In *Advances in Neural Information Processing Systems*, pp. 5014–5026, 2018.

Shi, G., Hönig, W., Shi, X., Yue, Y., and Chung, S.-J. Neural-swarm2: Planning and control of heterogeneous multirotor swarms using learned interactions. *arXiv preprint arXiv:2012.05457*, 2020.

Simon, D. *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. John Wiley & Sons, 2006.

Son, K., Kim, D., Kang, W. J., Hostallero, D. E., and Yi, Y. Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning. *arXiv preprint arXiv:1905.05408*, 2019.

Sunehag, P., Lever, G., Gruslys, A., Czarnecki, W. M., Zambaldi, V. F., Jaderberg, M., Lanctot, M., Sonnerat, N., Leibo, J. Z., Tuyls, K., et al. Value-decomposition networks for cooperative multi-agent learning based on team reward. In *AAMAS*, pp. 2085–2087, 2018.

Tan, M. Multi-agent reinforcement learning: Independent vs. cooperative agents. In *Proceedings of the tenth international conference on machine learning*, pp. 330–337, 1993.

Theodorou, E., Tassa, Y., and Todorov, E. Stochastic differential dynamic programming. In *Proceedings of the 2010 American Control Conference*, pp. 1125–1132. IEEE, 2010.

Wang, L., Yang, Z., and Wang, Z. Breaking the curse of many agents: Provable mean embedding q-iteration for mean-field reinforcement learning. *arXiv preprint arXiv:2006.11917*, 2020.

Wang, Z., Lee, K., Pereira, M. A., Exarchos, I., and Theodorou, E. A. Deep forward-backward SDEs for min-max control. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 6807–6814. IEEE, 2019a.

Wang, Z., Pereira, M. A., and Theodorou, E. A. Deep 2fbsdes for systems with control multiplicative noise. *arXiv preprint arXiv:1906.04762*, 2019b.

Yang, Y., Tutunov, R., Sakulwongtana, P., Ammar, H. B., and Wang, J. α^{α}-rank: Scalable multi-agent evaluation through evolution. *arXiv preprint arXiv:1909.11628*, 2019.

Yong, J. and Zhou, X. Y. *Stochastic controls: Hamiltonian systems and HJB equations*, volume 43. Springer Science & Business Media, 1999.

Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R. R., and Smola, A. J. Deep sets. In *Advances in neural information processing systems*, pp. 3391–3401, 2017.

Zhou, M., Chen, Y., Wen, Y., Yang, Y., Su, Y., Zhang, W., Zhang, D., and Wang, J. Factorized q-learning for large-scale multi-agent systems. In *Proceedings of the First International Conference on Distributed Artificial Intelligence*, pp. 1–7, 2019.

Supplementary Material

A. Multi-agent HJB Derivation

Applying Bellman's principle to the value function (3) yields,

$$\begin{aligned}
V^i(t, \boldsymbol{X}(t)) &= \inf_{U_i \in \mathcal{U}_i} \mathbb{E}\left[V^i(t + \mathrm{d}t, \boldsymbol{X}(t + dt)) + \int_t^{t+dt} C^i \mathrm{d}\tau \right] \\
&= \inf_{U_i \in \mathcal{U}_i} \mathbb{E}\big[C^i \mathrm{d}t + V^i(t, \boldsymbol{X}(t)) + V_t^i(t, \boldsymbol{X}(t))\mathrm{d}t \\
&+ V_x^{i\mathrm{T}}(t, \boldsymbol{X}(t))\mathrm{d}\boldsymbol{X} + \frac{1}{2}\mathrm{tr}(V_{xx}(t, \boldsymbol{X}(t)\Sigma\Sigma^{\mathrm{T}})\mathrm{d}t \Big] \\
&= \inf_{U_i \in \mathcal{U}_i} \mathbb{E}\big[C^i \mathrm{d}t + V^i(t, \boldsymbol{X}(t)) + V_t^i(t, \boldsymbol{X}(t))\mathrm{d}t \\
&+ V_x^{i\mathrm{T}}(t, \boldsymbol{X}(t))((f + G\boldsymbol{U})\mathrm{d}t + \Sigma \mathrm{d}\boldsymbol{W}) + \frac{1}{2}\mathrm{tr}(V_{xx}^i(t, \boldsymbol{X}(t))\Sigma\Sigma^{\mathrm{T}})\mathrm{d}t \Big] \\
&= \inf_{U_i \in \mathcal{U}_i} \big[C^i \mathrm{d}t + V^i(t, \boldsymbol{X}(t)) + V_t^i(t, \boldsymbol{X}(t))\mathrm{d}t \\
&+ V_x^{i\mathrm{T}}(t, \boldsymbol{X}(t))((f + G\boldsymbol{U})\mathrm{d}t) + \frac{1}{2}\mathrm{tr}(V_{xx}^i(t, \boldsymbol{X}(t))\Sigma\Sigma^{\mathrm{T}})\mathrm{d}t \Big] \\
\Rightarrow 0 &= V_t^i(t, \boldsymbol{X}(t)) + \inf_{U_i \in \mathcal{U}_i} \big[C^i + V_x^{i\mathrm{T}}(t, \boldsymbol{X}(t))(f + G\boldsymbol{U}) \big] + \frac{1}{2}\mathrm{tr}(V_{xx}^i(t, \boldsymbol{X}(t))\Sigma\Sigma^{\mathrm{T}})
\end{aligned} \tag{20}$$

Given the cost function assumption (the cost function is quadratic w.r.t control variable.), the infimum can be obtained explicitly using optimal control $U_i^* = -R^{-1}(G_i^{\mathrm{T}} V_x^i + Q_i^{\mathrm{T}} \boldsymbol{X})$. With that we can obtain the final form of the HJB PDE as

$$V_t^i + h + V_x^{i\mathrm{T}}(f + G\boldsymbol{U}_{0,*}) + \frac{1}{2}\mathrm{tr}(V_{xx}^i \Sigma\Sigma^{\mathrm{T}}) = 0, \quad V^i(T, \boldsymbol{X}) = g(\boldsymbol{X}(T)). \tag{21}$$

B. Multi-agent FBSDE Derivation

Given the HJB PDE in equation 5, one can apply the non-linear Feynman-Kac lemma (Karatzas & Shreve, 1991) to obtain a set of FBSDE as

$$\begin{aligned}
\mathrm{d}\boldsymbol{X}(t) &= (f + G\boldsymbol{U}_{0,*})\mathrm{d}t + \Sigma \mathrm{d}\boldsymbol{W}, \quad \boldsymbol{X}(0) = \boldsymbol{x}_0 \quad \text{(FSDE)} \\
\mathrm{d}V^i &= -h\mathrm{d}t + V_x^{i\mathrm{T}} \Sigma \mathrm{d}\boldsymbol{W}, \quad V(\boldsymbol{X}(T)) = g(\boldsymbol{X}(T)). \quad \text{(BSDE)}
\end{aligned} \tag{22}$$

The backward process is derived by applying Ito's lemma on V^i

$$\begin{aligned}
\mathrm{d}V^i &= V_t^i \mathrm{d}t + V_x^{i\mathrm{T}} \mathrm{d}\boldsymbol{X} + \frac{1}{2}\mathrm{tr}(V_{xx}^i \Sigma\Sigma^{\mathrm{T}})\mathrm{d}t \\
&\quad \text{Plug in eq.21 into} V_t^i \text{ term, and eq.1into d}\boldsymbol{X} \text{ term} \\
&= (-h - V_x^{i\mathrm{T}}(f + G\boldsymbol{U}_{0,*}) - \frac{1}{2}\mathrm{tr}(V_{xx}^i \Sigma\Sigma^{\mathrm{T}}))\mathrm{d}t + V_x^{i\mathrm{T}}((f + G\boldsymbol{U}_{0,*})\mathrm{d}t + \Sigma \mathrm{d}\boldsymbol{W}) + \frac{1}{2}\mathrm{tr}(V_{xx}^i \Sigma\Sigma^{\mathrm{T}})\mathrm{d}t \\
&= -h\mathrm{d}t + V_x^{i\mathrm{T}} \Sigma \mathrm{d}\boldsymbol{W}.
\end{aligned}$$

C. Metrics

In the test set, we randomly select B initial states, and $B \times T$ noise $\boldsymbol{W}$, where B is Batch size and T is time horizon. We evaluate the performance of models based on three different metrics. All losses (for comparison) are computed by averaging the last 10 stages and over 3 random seeds for fair comparison.

C.1. Relative Square Error(RSE)

The RSE is a metric applied in test phase which is defined as following:

$$\mathcal{L}_{RSE} = \frac{\sum_{1\leq j\leq B}^{i\in\mathbb{I}}(\hat{Y}^i(0,\boldsymbol{X}^j(0)) - Y^i(0,\boldsymbol{X}^j(0)))^2}{\sum_{1\leq j\leq B}^{i\in\mathbb{I}}(\hat{Y}^i(0,\boldsymbol{X}^j(0)) - \bar{Y}^i(0,\boldsymbol{X}^j(0)))^2}, \tag{23}$$

Where Y^i is the analytical solution of value function for ith agents at initial state $\boldsymbol{X}^j(0)$. The initial state $\boldsymbol{X}^j(0)$ is new batch of data sampled from same distribution as $\boldsymbol{X}(0)$ in the training phase. The batch size B is 256 for all inter-bank simulations. $\hat{Y}^i$ is the approximated value function for ith agent by FBSDE controller, and $\bar{Y}^i$ is the average of analytical solution for ith agent over the entire batch.

C.2. Evaluation/training Loss

The evaluation loss is same as training loss which is defined as the mean square error between true terminal value evaluated on the terminal state and the value propagated by the BSDE,

$$\mathcal{L}(\hat{Y}_T^i, Y_T^i) = \frac{1}{B}||\hat{Y}_T^i - Y_T^i||_2^2 \tag{24}$$

C.3. Cumulative Loss

The cumulative loss is computed explicitly from the objective function of optimal control (2) in the test phase. Here we restate it for completeness.

$$\begin{aligned}\mathcal{L}_{cum} &:= J_t^i(\boldsymbol{X}, U_{i,m}; \boldsymbol{U}_{-i,m-1}) \\ &= \mathbb{E}\left[g(\boldsymbol{X}_T) + \int_0^T C^i(\boldsymbol{X}_\tau, U_i(\boldsymbol{X}_\tau); \boldsymbol{U}_{-i})\mathrm{d}\tau\right],\end{aligned} \tag{25}$$

D. Missing Derivation and Proof in Section 3

D.1. Assumption 2

Here we state the assumption 2 in detail.

Consider a general FBSDE system,

$$\begin{aligned}\boldsymbol{X}_T^{t,\boldsymbol{x}} &= \boldsymbol{x} + \int_t^T \mu_s \mathrm{d}s + \int_t^T \Sigma_s \mathrm{d}\boldsymbol{W}_s \qquad \text{(FSDE)}, \\ Y_t^{T,\boldsymbol{x}} &= g(\boldsymbol{X}_T^{t,\boldsymbol{x}}) - \int_t^T H_s \mathrm{d}s + \int_t^T Z_s \mathrm{d}\boldsymbol{W}_s \quad \text{(BSDE)},\end{aligned} \tag{26}$$

We use $|\cdot|$ and $||\cdot||_F$ to denote the L^2 norm and Frobenius norm respectively. For terminal loss $g(\cdot)$, drift function $\mu(\cdot,\cdot,\cdot)$, $H(\cdot,\cdot,\cdot)$, diffusion function $\Sigma(\cdot,\cdot)$, and control function $\boldsymbol{u}(\cdot,\cdot)$, they are Lipschitz continuous with respect to their arguments:

$$\begin{aligned}|g(t,\boldsymbol{x}_1) - g(t,\boldsymbol{x}_2)|^2 &\leq g_x|\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ |H(t,\boldsymbol{x}_1,\boldsymbol{z}_1) - H(t,\boldsymbol{x}_2,\boldsymbol{z}_2)|^2 &\leq H_x|\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 + H_z||\boldsymbol{z}_1 - \boldsymbol{z}_2||_F^2, \\ ||\Sigma(t,\boldsymbol{x}_1) - \Sigma(t,\boldsymbol{x}_2)||_F^2 &\leq \Sigma_x|\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ ||\boldsymbol{u}(t,\boldsymbol{x}_1) - \boldsymbol{u}(t,\boldsymbol{x}_2)||^2 &\leq u_x|\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ |\mu(t,\boldsymbol{x}_1,\boldsymbol{u}_1) - \mu(t,\boldsymbol{x}_2,\boldsymbol{u}_2)|^2 &\leq \mu_x|\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 + \mu_u|\boldsymbol{u}_1 - \boldsymbol{u}_2|^2,\end{aligned} \tag{27}$$

Here $\mu_x, \mu_u, g_x, H_x, H_z, \Sigma_x, u_x$ are all positive constants.

D.2. Proof of Lemma 1

Proof. Denote $(\boldsymbol{X}_s^{t,\boldsymbol{x}}, Y_s^{t,\boldsymbol{x}}, Z_s^{t,\boldsymbol{x}})_{t\le s\le T}$ as the solution for the FBSDE system for the ith agent:

$$\begin{aligned}\boldsymbol{X}_T^{t,\boldsymbol{x}} &= \boldsymbol{x} + \int_t^T \mu_s \mathrm{d}s + \int_t^T \Sigma_s \mathrm{d}\boldsymbol{W}_s \qquad \text{(FSDE)},\\ Y_t^{T,\boldsymbol{x}} &= g(\boldsymbol{X}_T^{t,\boldsymbol{x}}) - \int_t^T H_s \mathrm{d}s + \int_t^T Z_s \mathrm{d}\boldsymbol{W}_s \quad \text{(BSDE)},\end{aligned} \tag{28}$$

for any $(t_0, \boldsymbol{x}) \in [t_0, T] \times \mathcal{X}$. Then for any $t \in [t_0, T]$ and $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}$, let $(\boldsymbol{X}_t^j, Y_t^j, Z_t^j)$ be the short notation of $(\boldsymbol{X}_t^{t_0,\boldsymbol{x}_j}, Y_t^{t_0,\boldsymbol{x}_j}, Z_t^{t_0,\boldsymbol{x}_j})$, where $j \in \{1, 2\}$, $t \in [0, T]$.
Here we define $\delta\boldsymbol{X}_t, \delta Y_t, \delta Z_t, \delta\Sigma_t, \delta H_t$ as:

$$\begin{aligned}\delta\boldsymbol{X}_t &= \boldsymbol{X}_t^1 - \boldsymbol{X}_t^2,\\ \delta Y_t &= Y_t^1 - Y_t^2,\\ \delta Z_t &= Z_t^1 - Z_t^2,\\ \delta\mu_t &= \mu_t(t, \boldsymbol{X}_t^1, \boldsymbol{U}_t^1) - \mu(t, \boldsymbol{X}_t^2, \boldsymbol{U}_t^2),\\ \delta H_t &= H(t, \boldsymbol{X}_t^1, Z_t^1) - H(t, \boldsymbol{X}_t^2, Z_t^2),\\ \delta\Sigma_t &= \Sigma(t, \boldsymbol{X}_t^1) - \Sigma(t, \boldsymbol{X}_t^2),\end{aligned} \tag{29}$$

Then we have:

$$\begin{aligned}\mathrm{d}\delta\boldsymbol{X}_t &= \mathrm{d}\boldsymbol{X}_t^1 - \mathrm{d}\boldsymbol{X}_t^2\\ &= (\mu_t(t, \boldsymbol{X}_t^1, \boldsymbol{U}_t^1)\mathrm{d}t + \Sigma(t, \boldsymbol{X}_t^1)\mathrm{d}W_t) - (\mu_t(t, \boldsymbol{X}_t^2, \boldsymbol{U}_t^2)\mathrm{d}t + \Sigma(t, \boldsymbol{X}_t^2)\mathrm{d}W_t)\\ &= (\delta\mu_t)\mathrm{d}t + \delta\Sigma_t \mathrm{d}W_t,\\ \mathrm{d}\delta Y_t &= -\delta H_t \mathrm{d}t + \delta Z_t^{\mathrm{T}} \mathrm{d}W_t.\end{aligned} \tag{30}$$

By applying Itô lemma to $\mathrm{d}\delta\boldsymbol{X}_t$ and $\mathrm{d}\delta Y_t$,

$$\begin{aligned}\mathrm{d}|\delta\boldsymbol{X}_t|^2 &= (2\delta\mu_t\delta\boldsymbol{X}_t + \frac{1}{2}\cdot 2\delta||\Sigma_t||_F^2)\mathrm{d}t + 2(\delta\boldsymbol{X}_t)^{\mathrm{T}}\delta\Sigma_t \mathrm{d}W_t\\ &= (2\delta\mu_t\delta\boldsymbol{X}_t + \delta||\Sigma_t||_F^2)\mathrm{d}t + 2(\delta\boldsymbol{X}_t)^{\mathrm{T}}\delta\Sigma_t \mathrm{d}W_t\\ \mathrm{d}|\delta Y_t|^2 &= (-2\delta H_t\delta Y_t + \delta||Z_t^{\mathrm{T}}||_F^2)\mathrm{d}t + 2(\delta Z_t\delta Y_t)^{\mathrm{T}}\mathrm{d}W_t\end{aligned} \tag{31}$$

By taking the expectation on both sides of $\mathrm{d}|\delta\boldsymbol{X}_t|^2$, it will yield:

$$\begin{aligned}\mathbb{E}[|\delta\boldsymbol{X}_t|^2] &= |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 + \int_{t_0}^t \mathbb{E}\left[2\delta\mu_s\delta\boldsymbol{X}_s + \delta||\Sigma_s||_F^2\right]\mathrm{d}s\\ &\le |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 + \int_{t_0}^t \mathbb{E}\left[(\mu_x + \mu_u u_x)^{-1}|\delta\mu_s|^2 + (\mu_x + \mu_u u_x)|\delta\boldsymbol{X}_s|^2 + \delta||\Sigma_s||_F^2\right]\mathrm{d}s\\ &\le |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 + \int_{t_0}^t \mathbb{E}\left[(\mu_x + \mu_u u_x)^{-1}(\mu_x|\delta\boldsymbol{X}_s|^2 + \mu_u|\delta\boldsymbol{u}|^2) + (\mu_x + \mu_u u_x)|\delta\boldsymbol{X}_s|^2 + \delta||\Sigma_s||_F^2\right]\mathrm{d}s\\ &\le |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 + \int_{t_0}^t \mathbb{E}\left[(\mu_x + \mu_u u_x)^{-1}(\mu_x|\delta\boldsymbol{X}_s|^2 + \mu_u u_x|\delta\boldsymbol{X}_s|^2) + (\mu_x + \mu_u u_x)|\delta\boldsymbol{X}_s|^2 + \delta||\Sigma_s||_F^2\right]\mathrm{d}s\\ &\le |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 + \int_{t_0}^t \mathbb{E}\left[|\delta\boldsymbol{X}_s|^2 + (\mu_x + \mu_u u_x)|\delta\boldsymbol{X}_s|^2 + \Sigma_x|\delta\boldsymbol{X}_s|^2\right]\mathrm{d}s\\ &= |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 + (I + \mu_x + \mu_u u_x + \Sigma_x)\int_{t_0}^t \mathbb{E}|\delta\boldsymbol{X}_s|^2 \mathrm{d}s\\ &\text{by Gronwall's inequality}\\ &\le e^{(I+\mu_x+\mu_u u_x+\Sigma_x)(t-t_0)}|\boldsymbol{x}_1 - \boldsymbol{x}_2|^2\\ &= e^{\xi(t-t_0)}|\boldsymbol{x}_1 - \boldsymbol{x}_2|^2\end{aligned} \tag{32}$$

Where $\xi = I + \mu_x + \mu_u u_x + \Sigma_x$.

Similarly, we can have,

$$\begin{aligned}
\mathbb{E}[|\delta \boldsymbol{Y}_t|^2] &= \mathbb{E}|\delta \boldsymbol{Y}_T|^2 + \int_t^T \mathbb{E}\left[2\delta H_s \delta Y_s - \delta||\boldsymbol{Z}_s^{\mathrm{T}}||_F^2\right] \mathrm{d}s \\
&= \mathbb{E}|\boldsymbol{g}(T, \boldsymbol{x}_1) - \boldsymbol{g}(T, \boldsymbol{x}_2)|^2 + \int_t^T \mathbb{E}\left[2\delta H_s \delta Y_s\right] - \mathbb{E}|||\boldsymbol{Z}_s||_F^2 \mathrm{d}s \\
&\leq g_x \mathbb{E}|\delta \boldsymbol{X}_T|^2 + \int_t^T H_z \mathbb{E}|\delta Y_s|^2 + H_z^{-1}\mathbb{E}|\delta H_s|^2 - \mathbb{E}||\boldsymbol{Z}_s||_F^2 \mathrm{d}s \\
&\leq g_x \mathbb{E}|\delta \boldsymbol{X}_T|^2 + \int_t^T H_z \mathbb{E}|\delta Y_s|^2 + H_z^{-1}\mathbb{E}\left[H_x|\delta \boldsymbol{X}_s|^2 + H_z|\delta \boldsymbol{Z}_s|^2\right] - \mathbb{E}||\boldsymbol{Z}_s||_F^2 \mathrm{d}s \\
&= g_x \mathbb{E}|\delta \boldsymbol{X}_T|^2 + \int_t^T H_z \mathbb{E}|\delta Y_s|^2 + H_z^{-1} H_x \mathbb{E}|\delta \boldsymbol{X}_s|^2 \mathrm{d}s \\
&\leq \left[g_x e^{\xi(t-t_0)} + H_x \frac{e^{\xi(T-t)} - e^{\xi(t-t_0)}}{H_z \Sigma_x}\right] |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 + H_z \int_t^T \mathbb{E}|\delta Y_s|^2 \mathrm{d}s \\
&\quad \text{by Gronwall's inequality} \\
|\delta \boldsymbol{Y}_t|^2 &\leq e^{H_z(T-t)} \left[g_x e^{\xi(T-t_0)} + H_x \frac{e^{\xi(T-t_0)} - e^{\xi(t-t_0)}}{H_z \Sigma_x}\right] |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2
\end{aligned} \tag{33}$$

When $t_0 = 0$, one can have:

$$\begin{aligned}
|\delta Y_T|^2 &\leq g_x e^{\xi T} |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\
&= L_1 |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\
|\delta Y_0|^2 &\leq e^{H_z T} \left[g_x e^{\xi T} + H_x \frac{e^{\xi T} - 1}{H_z \Sigma_x}\right] |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\
&= L_2 |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2
\end{aligned} \tag{34}$$

Where

$$\begin{aligned}
L_1 &= g_x e^{\xi T} \\
L_2 &= e^{H_z T} \left[g_x e^{\xi T} + H_x \frac{e^{\xi T} - 1}{H_z \Sigma_x}\right] \\
\xi &= I + \mu_x + \mu_u u_x + \Sigma_x
\end{aligned} \tag{35}$$

□

D.3. Lemma.2 with Proof

Lemma 2. *Denote $(\boldsymbol{X}_s^{t,\boldsymbol{x}}, Y_s^{t,\boldsymbol{x}}, Z_s^{t,\boldsymbol{x}})_{t \leq s \leq T}$ as the solution for the FBSDE system with importance sampling (12, 13) satisfying assumptions 1 and 2. Denote the difference of Y component at two different states $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ as:*

$$\delta \boldsymbol{X}_t = \boldsymbol{X}_t^{t_0,\boldsymbol{x}_1} - \boldsymbol{X}_t^{t_0,\boldsymbol{x}_2}, \delta Y_t = Y_t^{t_0,\boldsymbol{x}_1} - Y_t^{t_0,\boldsymbol{x}_2}. \tag{36}$$

Then we can have:

$$\begin{aligned}
|\delta Y_T|^2 &\leq \tilde{L}_1 |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2, \\
|\delta Y_{t_0}|^2 &\leq \tilde{L}_2 |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2,
\end{aligned} \tag{37}$$

Where L_1 and L_2 are defined as:

$$\begin{aligned}
\tilde{L}_1 &= g_x e^{\tilde{\xi}} \\
\tilde{L}_2 &= e^{2(H_z + k_z)(T-t_0)} \left[g_x e^{\tilde{\xi}(T-t_0)} + H_x (H_z^{-1} + k_z^{-1}) \frac{e^{\tilde{\xi}(T-t_0)} - 1}{2\Sigma_x}\right], \\
\tilde{\xi} &= I + \tilde{\mu}_x + \tilde{\mu}_u u_x + \Sigma_x,
\end{aligned} \tag{38}$$

Where $\mu_x, \mu_u, \Sigma_x, H_x, H_z, g_x, u_x$ are Lipschitz constant defined in Assumption.2. The definition of Lipschitz constant $\tilde{\mu}_x, \tilde{\mu}_u, k_z$ and proof can be found in the following proof.

Proof. Similar to the proof of lemma.1, now we first analyze the forward process with IS. Inspired by the success of (Exarchos & Theodorou, 2018), we select the control computed from the last run as the importance sampling term. Then the IS term in FSDE is defined as $m_s := \Sigma K = GU_{*,0}$. New drift term is modified as $\tilde{\mu}_s = \mu_s + m_s$ with Lipschitz constant $\tilde{\mu}_x$ and $\tilde{\mu}_u$. In the BSDE, Then IS term is written as $k_s = Z_s K_s = Z_s \Gamma U_{*,0} = V_x G U_{*,0}$, and the modified $\tilde{H}_s = H_s + k_s$. Here we formally write the Lipschitz constant for IS terms in FSDE and BSDE are:

$$\begin{aligned} |m_s(t, \boldsymbol{x}_1, \boldsymbol{u}_1) - m_s(t, \boldsymbol{x}_2, \boldsymbol{u}_2)|^2 \leq m_x |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 + m_u |\boldsymbol{u}_1 - \boldsymbol{u}_2|^2, \\ |k_s(t, \boldsymbol{x}_1, \boldsymbol{z}_1) - k_s(t, \boldsymbol{x}_2, \boldsymbol{z}_2)|^2 \leq k_x |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 + k_z |\boldsymbol{z}_1 - \boldsymbol{z}_2|^2, \end{aligned} \tag{39}$$

Similar to the proof (D.2), we can have,

$$\begin{aligned} |\delta \boldsymbol{X}_t|^2 &\leq e^{(I + \tilde{\mu}_x + \tilde{\mu}_u u_x + \Sigma_x)(t - t_0)} |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ &= e^{\tilde{\xi}(t - t_0)} |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \end{aligned} \tag{40}$$

Where $\tilde{\xi} = I + \tilde{\mu}_x + \tilde{\mu}_u u_x + \Sigma_x$.

And for Y term we will have,

$$\begin{aligned} \mathbb{E}[|\delta \boldsymbol{Y}_t|^2] &= \mathbb{E}|\delta \boldsymbol{Y}_T|^2 + \int_t^T \mathbb{E}\left[2(\delta H_s \delta Y_s + \delta k_s \delta Y_s) - \delta ||Z_s^{\mathrm{T}}||_F^2\right] \mathrm{d}s \\ &\leq g_x \mathbb{E}|\delta \boldsymbol{X}_T|^2 + \int_t^T 2H_z \mathbb{E}|\delta Y_s|^2 + (2H_z)^{-1} \mathbb{E}|\delta H_s|^2 + 2k_z \mathbb{E}|\delta Y_s|^2 + (2k_z)^{-1} \mathbb{E}|\delta k_s|^2 - \mathbb{E}||Z_s||_F^2 \mathrm{d}s \\ &\leq g_x \mathbb{E}|\delta \boldsymbol{X}_T|^2 + \int_t^T (2H_z + 2k_z) \mathbb{E}|\delta Y_s|^2 + (2H_z)^{-1} \mathbb{E}\left[H_x |\delta \boldsymbol{X}_s|^2 + H_z ||\delta Z_s||_F^2\right] \\ &+ (2k_z)^{-1} \mathbb{E}\left[k_x |\delta \boldsymbol{X}_s|^2 + k_z ||\delta Z_s||_F^2\right] - \mathbb{E}||Z_s||_F^2 \mathrm{d}s \end{aligned}$$

Noticing that the drift term in BSDE w/o IS is $H_s = C^{i*} + V_x G U_{*,0}$ which is a lipschitz continous function, while IS term is $k_s = V_x G U_{*,0}$. then we can have $k_x \leq H_x$. By replacing k_x by H_x, it yields:

$$\begin{aligned} &\leq g_x \mathbb{E}|\delta \boldsymbol{X}_T|^2 + \int_t^T (2H_z + 2k_z) \mathbb{E}|\delta Y_s|^2 + (2H_z)^{-1} \mathbb{E}\left[H_x |\delta \boldsymbol{X}_s|^2 + H_z ||\delta Z_s||_F^2\right] \\ &+ (2k_z)^{-1} \mathbb{E}\left[H_x |\delta \boldsymbol{X}_s|^2 + k_z ||\delta Z_s||_F^2\right] - \mathbb{E}||Z_s||_F^2 \mathrm{d}s \\ &\leq g_x \mathbb{E}|\delta \boldsymbol{X}_T|^2 + \int_t^T (2H_z + 2k_z) \mathbb{E}|\delta Y_s|^2 + \frac{H_x}{2} (H_z^{-1} + k_z^{-1}) \mathbb{E}|\delta \boldsymbol{X}_s|^2 + \mathbb{E}||\delta Z_s||_F^2 - \mathbb{E}||Z_s||_F^2 \mathrm{d}s \\ &= g_x \mathbb{E}|\delta \boldsymbol{X}_T|^2 + \int_t^T (2H_z + 2k_z) \mathbb{E}|\delta Y_s|^2 + \frac{H_x}{2} (H_z^{-1} + k_z^{-1}) \mathbb{E}|\delta \boldsymbol{X}_s|^2 \mathrm{d}s \end{aligned}$$

by Gronwall's inequality

$$|\delta \boldsymbol{Y}_t|^2 \leq e^{2(H_z + k_z)(T - t)} \left[g_x e^{\tilde{\xi}(T - t_0)} + H_x (H_z^{-1} + k_z^{-1}) \frac{e^{\tilde{\xi}(T - t_0)} - e^{\tilde{\xi}(t - t_0)}}{2\Sigma_x} \right] |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \tag{41}$$

When $t_0 = 0$, we have,

$$\begin{aligned} |\delta Y_T|^2 &\leq g_x e^{\tilde{\xi} T} |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ &= \tilde{L}_1 |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ |\delta Y_0|^2 &\leq e^{2(H_z + k_z)T} \left[g_x e^{\tilde{\xi} T} + H_x (H_z^{-1} + k_z^{-1}) \frac{e^{\tilde{\xi} T} - 1}{2\Sigma_x} \right] |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ &= \tilde{L}_2 |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \end{aligned} \tag{42}$$

Where $\tilde{\xi} = I + \tilde{\mu}_x + \tilde{\mu}_u u_x + \Sigma_x$. Following arguments in (Ma et al., 2002), one further has,

$$||Z_t||_S^2 \leq ||\Sigma||_S^2 ||\nabla_x Y_t||_S^2 \leq M_\Sigma \tilde{L}_2 \tag{43}$$

□

D.4. Proof of Theorem 2

According to the result in Lemma.1 with the assumption that the initial state dataset $\mathcal{D}$ are identical for FBSDE w/ and w/o importance sampling.

$$\begin{aligned} |\delta Y_T|^2 &\leq g_x e^{\xi T} |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ &= L_1 |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ |\delta Y_0|^2 &\leq e^{H_z T} \left[g_x e^{\xi T} + H_x \frac{e^{\xi T} - 1}{H_z \Sigma_x} \right] |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ &= L_2 |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ \xi &= I + \mu_x + \mu_u u_x + \Sigma_x \end{aligned} \tag{44}$$

Similarly, According to Lemma.2, one have,

$$\begin{aligned} |\delta Y_T|^2 &\leq g_x e^{\tilde{\xi} T} |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ &= \tilde{L}_1 |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ |\delta Y_0|^2 &\leq e^{2(H_z + k_z)T} \left[g_x e^{\tilde{\xi} T} + H_x (H_z^{-1} + k_z^{-1}) \frac{e^{\tilde{\xi} T} - 1}{2\Sigma_x} \right] |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ &= \tilde{L}_2 |\boldsymbol{x}_1 - \boldsymbol{x}_2|^2 \\ \tilde{\xi} &= I + \tilde{\mu}_x + \tilde{\mu}_u u_x + \Sigma_x \end{aligned} \tag{45}$$

We have $\tilde{\mu}_x = \mu_x + m_x \geq \mu_x$ and $\tilde{\mu}_u = \mu_u + m_u \geq \mu_u$, where m_x and m_u are the Lipschitz constants for m_s w.r.t. $\boldsymbol{x}$ and $\boldsymbol{u}$ defined in equation.39 . Then we have $\tilde{\xi} \geq \xi$ which leads to $\tilde{L}_1 \geq L_1$. Noticing that the drift term in BSDE w/o IS is $H_s = C^{i*} + V_x G U_{*,0}$, while IS term is $k_s = V_x G U_{*,0}$. then we can have $k_x \leq H_x$, and $k_z \leq H_z$ which leads to

$$\frac{1}{2}\left(\frac{1}{H_z} + \frac{1}{k_z}\right) > \frac{1}{H_z} \tag{46}$$

We have known that $\tilde{\mu}_x \geq \mu_x$ and $\frac{1}{2}(\frac{1}{H_z} + \frac{1}{k_z}) > \frac{1}{H_z}$. Then we can have $\tilde{L}_1 \geq L_1$ and $\tilde{L}_2 \geq L_2$ strictly.

E. Invariant Layer Introductions and Implementation Techniques

E.1. Invariant Mapping

A function f maps its domain from $\mathcal{X}$ to $\mathcal{Y}$. Domain $\mathcal{X}$ is a vector space $\mathbb{R}^d$ and $\mathcal{Y}$ is a continuous space $\mathbb{R}$. Assume the function takes a set as input: $\mathbb{X} = \{x_1 ... x_N\}$, then the function f is invariant if it satisfies property E.1.

Property 1. A function $f : \mathcal{X} \to \mathcal{Y}$ defined on sets is permutation invariant to the order of objects in the set. i.e. For any permutation function π: $f(\{x_1 ... x_N\}) = f(\{x_{\pi(1)} ... x_{\pi(N)}\})$

In this paper, we discuss the case when f is a neural network only.

Theorem 3. *(Zaheer et al., 2017) $\boldsymbol{X}$ has elements from countable universe. A function $f(\boldsymbol{X})$ is a valid permutation invariant function, i.e invariant to the permutation of $\boldsymbol{X}$, iff it can be decomposed in the form $\rho(\sum_{x \in \boldsymbol{X}} \phi(x))$, for appropriate functions ρ and ϕ.*

In the symmetric multi-agent system, each agent is not distinguishable. This property gives some hints about how to extract the features of the $-i$th agents using a neural network. The states of the $-i$th agents can be represented as a set: $\boldsymbol{X} = \{X_1, X_2, ..., X_{i-1}, X_{i+1}, ..., X_N\}$. We want to design a neural network f which has the property of permutation invariance. Specifically, ϕ is represented as a one layer neural network and ρ is a common nonlinear activation function, and the invariant layer module is shown in Fig.11.

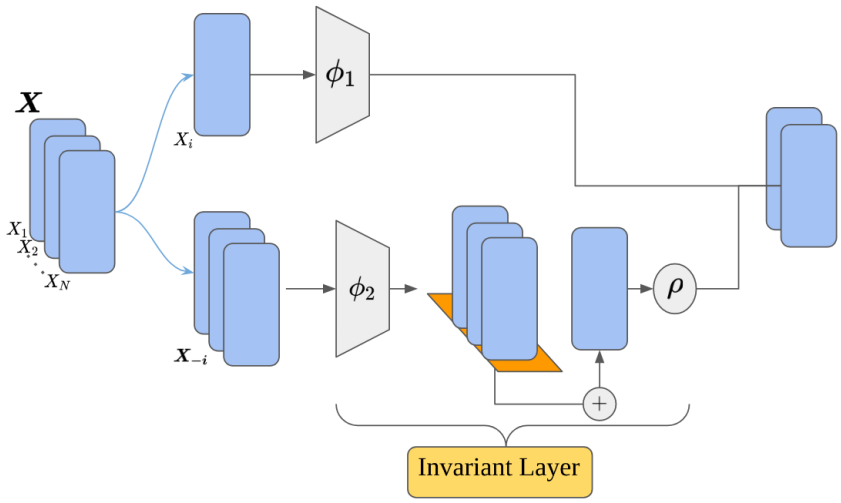

Figure 11. Feature extractor architecture. ϕ represents fully connected neural network. ρ is the ReLU activation function.

E.2. Feature Extractor with Invariant Layer Architecture

The architecture of feature extractor with invariant layer is described in Fig. 11.

E.3. Invariant Layer Techniques

Noticing that all the agents has the access to the global states, we define the state input features of invariant layer for the ith agent as:

$$\boldsymbol{X}_t = \{X_i, X_1, X_2..., X_{i-1}, X_{i+1}, ...X_N\}\,, \tag{47}$$

with shape of $[B, N, N_x]$, where B is the batch size, N_x is the dimension of the observed states. In the other word, **we always put own feature in the first position**. For each agent i, there will exist such a feature tensor, then for the feature extractor, the shape of input is $[BS, N, N, N_x]$. Therefore, the shape of input tensor will become $[BS, N, N-1, N_x]$ for invariant layer. where N is the number of agents. First,we could use neural network to map the observed states to the feature space with dimension N_f. Then the shape of the tensor will become $[BS, N, N-1, N_f]$. After summing up the features of all the element in the set, the dimension of the tensor would reduce to $[BS, N, 1, N_f]$, and we denote this feature tensor as F. However, the memory complexity is $\mathcal{O}(N^2 \times N_f)$ which is not tolerable when the number of agent N increases. Alternatively, we can simply map the tensor $\boldsymbol{X}_t$ whose dimension is $[BS, N, N_f]$ into the desired feature dimension N_f, then the shape of the tensor would become $[BS, N, N_f]$, and we denote this tensor as F'. Now we create another tensor which is the average of features of element in set with size $[BS, 1, N_f]$ and we denote it to be $\bar{F}'$. Then we compute $\hat{F} = (\bar{F}' \times N - F')/(N-1)$ which has size of $[BS, N, N_f]$. We can find that $\hat{F} = F$, and the memory complexity of computing $\hat{F}$ is just $\mathcal{O}(N \times N_f)$. The derivation only holds when the system is symmetric and the agents are not distinguishable. The technique can be extended to higher state dimension for individual agent.

F. Experiment Configurations

This section elaborates the experiment configurations for §4. For all the simulation, the number of SGD iteration is fixed as $N_{SGD} = 100$. We are using Adam as optimizer with 1E-3 learning rate for all simulations.

F.1. Inter-bank Experiments

In section §4.1, For the prediction of initial value function V_0^i, all frameworks are using 2 layers feed forward network with 128 hidden dimension. For the baseline framework, we followed the suggested configuration motioned in (Han et al., 2018). At each time steps, V_x^i is approximated by three layers of feed forward network with 64 hidden dimensions. We add batch norm (Ioffe & Szegedy, 2015) after each affine transformation and before each nonlinear activation function. For Deep

FBSDE with LSTM backbone, we are using two layer LSTM parametrized by 128 hidden state. If the framework includes the invariant layer, the number of mapping features is chosen to be 256. The hyperparameters of the dynamics is listed as following:

$$a = 0.1,\ q = 0.1,\ c = 0.5,\ \epsilon = 0.5,\ \rho = 0.2,\ \sigma = 1, T = 1. \tag{48}$$

In the simulation, the time horizon is separated into 40 time-steps over 1 second by Euler method. Learning rate is chosen to be 1E-3 which is the default learning rate for Adam optimizer. The initial state for each agents are sampled from the uniform distribution $[-\delta_0, \delta_0]$. Where δ_0 is the constant standard deviation of state $\boldsymbol{X}(t)$ during the process as described in (Han & Hu, 2019). In the evaluation, we are using 256 new sampled trajectory which are different from training trajectory to evaluate the performance. The number of stage is set to be 100 which is enough for all framework to converge.

F.2. Belief Space Car Racing

In §4.2, the hyperparameter is listed as following:

$$c_{drag} = 0.01,\ L = 0.1,\ c = 0.5,\ T = 10.0 \tag{49}$$

The observation noise is sampled from Gaussian noise $m \sim \mathcal{N}(0, 0.1\boldsymbol{I})$. The time horizon is enrolled into 100 time-steps by Euler method. In this experiments, the initial value V_i is approximated a single trainable scale and $V_{x,i}(t)$ is approximated by two layers of LSTM parametrized with 32 hidden dimensions. The number of stage is set to be 10.

G. FBSDEs and Analytical Solution for Inter-Bank Borrowing/Lending Problem

G.1. FBSDEs for Inter-Bank Borrowing/Lending Problem

By plugging the running cost (17) to the HJB (5) function, one can have,

$$V_t^i + \inf_{U_i \in \mathcal{U}_i} \left[\sum_{j=1}^{N} [a(\bar{X} - X_j) + U_j^2] V_{x_j} + \frac{1}{2} U_i^2 - q U_i (\bar{X} - X_i) + \frac{\epsilon}{2} (\bar{X} - X_i)^2 \right] + \frac{1}{2} \mathrm{tr}(V_{xx}^i \Sigma \Sigma^{\mathrm{T}}) = 0. \tag{50}$$

By computing the infimum explicitly, the optimal control of player i is:$U_i^*(\boldsymbol{X}, t) = q(\bar{X} - X_i) - V_x^i(\boldsymbol{X}, t)$. The final form of HJB can be obtained as

$$V_t^i + \frac{1}{2} \mathrm{tr}(V_{xx}^i \Sigma \Sigma^{\mathrm{T}}) + a(\bar{X} - X_i) V_x^i + \sum_{j \neq i} [a(\bar{X} - X_j) + U_j] V_x^j + \frac{\epsilon}{2} (\bar{X} - X_i)^2 - \frac{1}{2} (q(\bar{X} - X_i) - V_x^i)^2 = 0 \tag{51}$$

Applying Feynman-Kac lemma to equation 51, the corresponding FBSDE system is

$$\begin{aligned} \mathrm{d}\boldsymbol{X}(t) &= (f(\boldsymbol{X}(t), t) + G(\boldsymbol{X}(t), t)\boldsymbol{u}(t))\mathrm{d}t + \Sigma(t, \boldsymbol{X}(t))\mathrm{d}\boldsymbol{W_t}, \quad \boldsymbol{X}(0) = \boldsymbol{x}_0 \\ \mathrm{d}V^i &= -[\frac{\epsilon}{2} (\bar{X} - X_i)^2 - \frac{1}{2} (q(\bar{X} - X_i) - V_x^i)^2 + U_i]\mathrm{d}t + V_x^{i^{\mathrm{T}}} \Sigma dW, \quad V(T) = g(\boldsymbol{X}(T)). \end{aligned} \tag{52}$$

G.2. Analytical solutions for Inter-Bank Borrowing/Lending Problem

The analytical solution for linear inter-bank problem was derived in (Carmona et al., 2013). We provide them here for completeness. Assume the ansatz for HJB function is described as:

$$V_i(t, \boldsymbol{X}) = \frac{\eta(t)}{2} (\bar{X} - X_i)^2 = \mu(t) \quad i \in \mathbb{I} \tag{53}$$

Where $\eta(t), \mu(t)$ are two scalar functions. The optimal control under this ansatz is:

$$U_i^\star(t, \boldsymbol{X}) = \left[q + \eta(t)(1 - \frac{1}{N}) \right] (\bar{X} - X_i) \tag{54}$$

By pluginging the ansatz into HJB function derived in equation (51), one can have,

$$\begin{aligned}\dot{\eta}(t) &= 2(a+q)\eta(t) + (1-\frac{1}{N^2})\eta^2(t) - (\epsilon - q^2), \quad \eta(T) = c,\\ \dot{\mu}(t) &= -\frac{1}{2}\sigma^2(1-\rho^2)(1-\frac{1}{N})\eta(t), \quad \mu(T) = 0.\end{aligned} \tag{55}$$

There exists the analytical solution for the Riccati equation described above as,

$$\eta(t) = \frac{-(\epsilon - q^2)(e^{(\delta^+ - \delta^-)(T-t)} - 1) - c(\delta^+ e^{(\delta^+ - \delta^-)(T-t)} - \delta^-)}{(\delta^- e^{(\delta^+ - \delta^-)(T-t)} - \delta^+) - c(1-1/N^2)(e^{(\delta^+ - \delta^-)(T-t)}) - 1}. \tag{56}$$

Where $\delta^{\pm} = -(a+q) \pm \sqrt{R}$ and $R = (a+q)^2 + (1-1/N^2)(\epsilon - q^2)$

H. Additional Tables and Figures

H.1. Evaluation Loss with different number of agents

Fig.12 shows the comparison of SDFP-FBSDE and baseline by evaluation loss.

Figure 12. Comparison of SDFP-FBSDE and Baseline for inter-bank problem with different number of agents evaluated on evaluation loss(24).

H.2. Superlinear Inter-Bank Plots

Fig.13 demonstrates the performance difference between Baseline and our algorithm. One can find that our algorithm convergence faster and better than baseline. Since in the superlinear case, the influence of control term in the forward dynamics is mitigated, then the final performances are similar.

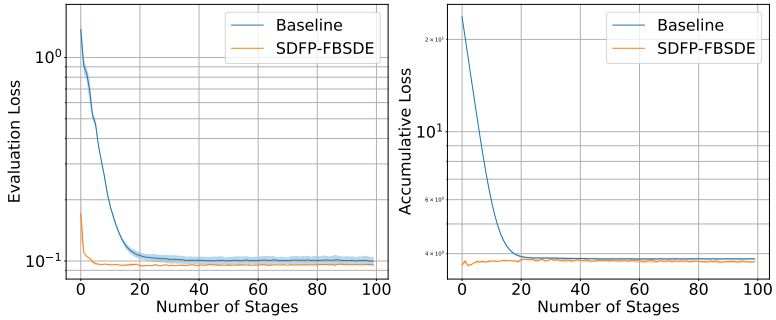

Figure 13. Comparison of SDFP-FBSDE and Baseline for inter-bank problem in superlinear case with evaluation loss (24) and cumulative loss(25)

H.3. Posterior Plot of Car Racing

Fig.14 illustrates the trajectory of single game with posterior estimated by each car. One can find that the variance does not blow up, and both of two cars are staying in the track.

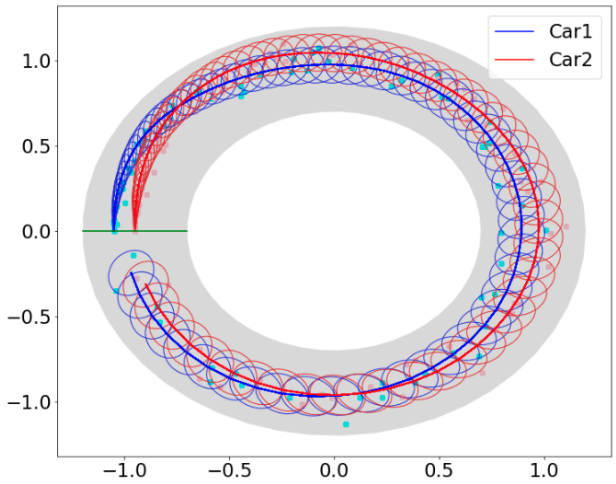

Figure 14. Car racing plot with posterior trained with SDFP-FBSDE. The competition loss is turned off

H.4. DFP-FBSDE Framework

In this subsection, we demonstrate the framework of Deep Fictitious Play FBSDE (DFP-FBSDE) in Fig.15. Each NN (blue box) represents for the FBSDE module shown in Fig.1.

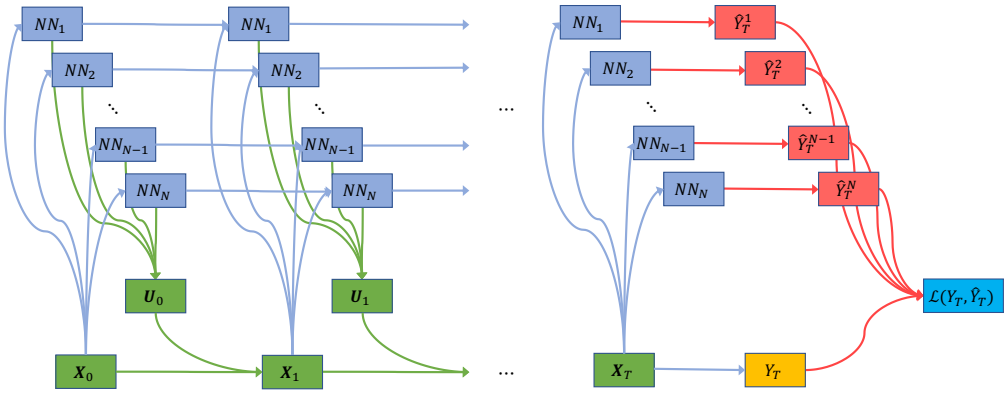

Figure 15. SDFP framework for N Players. Each NN block has the architecture in Fig. 1

I. Belief Car Racing

The framework for the racing problem is trained with batch size of 64, and 100 time steps over a time horizon of 10 seconds.

I.1. Continuous Time Extended Kalman Filter

The Partial Observable Markov Decision Process is generally difficult to solve within infinite dimensional space belief. Commonly, the Value function does not have explicit parameterized form. Kalman filter overcome this challenge by

presuming the noise distribution is Gaussian distribution. In order to deploy proposed Forward Backward Stochastic Differential Equation (FBSDE) model in the Belief space, we need to utilize extended Kalman filter in continuous time (Jazwinski, 1970) correspondingly. Given the partial observable stochastic system:

$$\frac{\mathrm{d}x}{\mathrm{d}t} = f(x, u, w, t), \quad \text{and} \quad z = h(x, v, t) \tag{57}$$

Where f is the stochastic state process featured by a Gaussian noise $w \sim \mathcal{N}(0, Q)$, h is the observation function while $v \sim \mathcal{N}(0, R)$ is the observation noise. Next, we consider the linearization of the stochastic dynamics in eq.(47) represented as follows:

$$A = \frac{\partial f}{\partial x}\bigg|_{\hat{x}}, L = \frac{\partial f}{\partial w}\bigg|_{\hat{x}}, C = \frac{\partial h}{\partial x}\bigg|_{\hat{x}}, M = \frac{\partial h}{\partial v}\bigg|_{\hat{x}}, \tilde{Q} = LQL^{\mathrm{T}}, \tilde{R} = MRM^{\mathrm{T}} \tag{58}$$

one can write the posterior mean state $\hat{x}$ and prior covariance matrix P^- estimation update rule by (Simon, 2006):

$$\begin{aligned} \hat{x}(0)\mathbb{E}[x(0)], \quad P^-(0) &= \mathbb{E}[(x(0) - \hat{x})(x(0) - \hat{x})^{\mathrm{T}}] \\ K &= PC^{\mathrm{T}}\tilde{R}^{-1} \\ \dot{\hat{x}} &= b(\hat{x}, u, w, t) = f(\hat{x}, u, w_0, t) + K[z - h(\hat{x}, v_0, t)] \\ \dot{P}^- &= AP^- + P^-A^{\mathrm{T}} + \tilde{Q} - P^-C^{\mathrm{T}}\tilde{R}^{-1}CP^- \end{aligned} \tag{59}$$

We follow the notation in (Simon, 2006), where x is the real state, $\hat{x}$ is the mean of state estimated by Kalman filter based on the noisy sensor observation, P^- represents for the covariance matrix of the estimated state, nominal noise values are given as $w_0 = 0$ and $v_0 = 0$, where superscript + is the posterior estimation and $-$ is the prior estimation. Then we can define a Gaussian belief dynamics as $\boldsymbol{b}(\hat{x}_k, P_k^-)$ by the mean state $\hat{x}$ and variance P^- of normal distribution $\mathcal{N}(\hat{x}_k, P_k^-)$

The belief dynamics results in a decoupled FBSDE system as follows:

$$\begin{aligned} \mathrm{d}\boldsymbol{b}_k &= g(\boldsymbol{b}_k, \boldsymbol{U}_k, 0)\mathrm{d}t + \Sigma(\boldsymbol{b}_k, \boldsymbol{U}_k, 0)\mathrm{d}W, \mathrm{d}W \sim \mathcal{N}(0, I) \\ \mathrm{d}V &= -C^{i^\star}\mathrm{d}t + V_x^{i^{\mathrm{T}}}\Sigma\mathrm{d}W \end{aligned} \tag{60}$$

where:

$$\begin{aligned} g(\mathbf{b}_k, \boldsymbol{U}_k) &= \begin{bmatrix} b(t, \boldsymbol{X}(t), U_{i,m}(t); \boldsymbol{U}_{-i,m}) \\ vec(A_kP_k^- + P_k^-A_k^{\mathrm{T}} + \tilde{Q}_k - P_k^-C_k^{\mathrm{T}}\tilde{R}_k^{-1}C_kP_k^-) \end{bmatrix} \\ \Sigma(\mathbf{b}_k, \boldsymbol{U}_k) &= \begin{bmatrix} \sqrt{K_kC_kP_k^-}\mathrm{d}t \\ \mathbf{0} \end{bmatrix} \\ V(T) &= g(\boldsymbol{X}(T)) \\ \hat{\boldsymbol{X}}(0) &= \mathbb{E}[\boldsymbol{X}(0)] \\ P^-(0) &= \mathbb{E}[(\boldsymbol{X}(0) - \hat{\boldsymbol{X}})(\boldsymbol{X}(0) - \hat{\boldsymbol{X}})^{\mathrm{T}}] \end{aligned} \tag{61}$$

In the car racing case, the dynamic function $f(\cdot, \cdot)$ in eq.57 is described as,

$$\begin{aligned} \mathrm{d}\boldsymbol{X} &= (f(\boldsymbol{X}) + G(\mathbf{X})\mathbf{U})\mathrm{d}t + \Sigma(\mathbf{X})\mathrm{d}\boldsymbol{W}, \quad \mathbf{z} = h(\mathbf{X}) + m \\ f(\mathbf{X}) &= \begin{bmatrix} v\cos\theta \\ v\sin\theta \\ -c_{\text{drag}}v \\ 0 \end{bmatrix}, \; G(\mathbf{X}) = \Sigma(\mathbf{X}) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & v/L \end{bmatrix}, \; h(\boldsymbol{X}) = \boldsymbol{x} \end{aligned} \tag{62}$$

Where $\mathrm{d}\boldsymbol{W}$ is standard Brownian motion.

I.2. Cost Functions

We consider the problem of two cars racing on a circular track. The cost function of each car is designed as

$$J_t = \underbrace{\exp\big(\big|\frac{x^2}{a^2} + \frac{y^2}{b^2} - 1\big|\big)}_{\textit{track cost}} + \underbrace{\mathrm{ReLU}\big(-v\big)}_{\textit{velocity cost}} + \underbrace{\exp\big(-d\big)}_{\textit{collision cost}}$$

Where d is Euclidean distance between two cars. We use continuous time extended Kalman Filter to propagate belief space dynamics described in equation 61.

We introduce the concept competitive game by using an additional competition cost:

$$J_{competition} = \exp(-\begin{bmatrix} cos(\theta) \\ sin(\theta) \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} x_1 - x_2 \\ y_1 - y_2 \end{bmatrix})$$

Where x_i, y_i is the x, y position of the ith car. When the ith car is leading, the competition loss will be minor, and it will increase exponentially when the car is trailing.
Thanks to the decoupled BSDE structure, each car can measure this competition loss separately and optimize the value function individually.

J. Hardware

All simulations are run on

1. Nvidia RTX TITAN
2. Nvidia GTX TITAN BLACK

K. Algorithm

Algorithm 1 Scalable Deep Fictitious Play FBSDE

Hyper-parameters:N: Number of players; T: Number of timesteps; M: Number of stages in fictitious play; N_{gd}: Number of gradient descent steps per stage; $\boldsymbol{U}_0$: the initial strategies for players in set $\mathbb{I}$; B: Batch size; Δt: time discretization (Total time/Number of timesteps); π: Permutation function (E.1).
Parameters:ϕ: Network weights for Initial Value (IV) prediction $f_{IV}(\cdot)$; θ: Weights and bias of Backbone and Feature extractor (BF) $f_{BF}(\cdot)$.
Initialize trainable papermeters:θ^0, ϕ^0
for $m \leftarrow 1$ to M **do**
  Generate B sample $\boldsymbol{x}_0$ and $B \times T$ Noise $\Delta \boldsymbol{w} \sim \mathcal{N}(0, \boldsymbol{I}\Delta t)$.
  for $l \leftarrow 0$ to $N_{gd} - 1$ **do**
    for $t \leftarrow 0$ to $T - 1$ **do**
      if t==0 **then**
        Predict value function for ith player: $\hat{y}_0^i = f_{IV}(\boldsymbol{x}_0; \phi^{m \times N_g d + l})$
      else
        Compute network prediction $\hat{z}_i$ for ith player: $\hat{z}_i = \Sigma_i^{\mathrm{T}} f_{BF}(\boldsymbol{x}_t; \theta^{m \times N_{gd} + l})$
      end if
      Compute ith optimal control :$u_i^* = -R^{-1}(\Gamma_i^{\mathrm{T}} z_i + Q_i^{\mathrm{T}} \boldsymbol{x}_t)$
      Infer $-i$th players' network prediction and stop the gradient for them: $\hat{\boldsymbol{z}}_{-i} = \Sigma_{-i}^{\mathrm{T}} f_{BF}(\pi(\boldsymbol{x}_t); \theta^{m \times N_{gd}})$
      Compute $-i$th optimal Control and stop the gradient for them: $\boldsymbol{u}_{-i}^* = -R_{-i}^{-1}(\Gamma_{-i}^{\mathrm{T}} \hat{\boldsymbol{z}}_{-i} + Q_{-i}^{\mathrm{T}} \boldsymbol{x}_t)$
      Propagate FSDE: $\boldsymbol{x}_{t+1} = f_{FSDE}(\boldsymbol{x}_t, u_i^*, \boldsymbol{u}_{-i}^*, \Delta \boldsymbol{w}_t, t)$ (12)
      Propagate BSDE: $\hat{y}_{t+1}^i = f_{BSDE}(\hat{y}_t^i, \boldsymbol{x}_t, u_i^*, \boldsymbol{u}_{-i}^*, \hat{z}_i, \Delta \boldsymbol{w}_t, t)$ (13)
    end for
  end for
  Compute True terminal value $y_T^i = g^i(\boldsymbol{x}_T)$
  Compute loss: $\mathcal{L}(\hat{y}_T^i, y_T^i) = \frac{1}{B}||\hat{y}_T^i - y_T^i||_2^2$ (24)
  Gradient Update: θ^l, ϕ^l
end for