approximated by $\dot{\boldsymbol{x}} \simeq \dot{\boldsymbol{x}}^{(1)} + \dot{\boldsymbol{x}}^{(2)}$ and $\dot{\boldsymbol{y}} \simeq \dot{\boldsymbol{y}}^{(1)} + \dot{\boldsymbol{y}}^{(2)}$;

$$\dot{\boldsymbol{x}}^{(1)} = +x^*(1-x^*)(\boldsymbol{u}\cdot\mathbf{1}_{\mathsf{z}})\boldsymbol{p}^* \circ \boldsymbol{\epsilon}, \tag{12}$$

$$\dot{\boldsymbol{y}}^{(1)} = -y^*(1-y^*)(\boldsymbol{u}\cdot\mathbf{1}_{\mathsf{z}})\boldsymbol{p}^* \circ \boldsymbol{\delta}, \tag{13}$$

$$\begin{aligned}\dot{\boldsymbol{x}}^{(2)} = &-(x^*-\tilde{x}^*)(\boldsymbol{u}\cdot\mathbf{1}_{\mathsf{z}})\boldsymbol{\delta}\circ\boldsymbol{\epsilon}\circ\boldsymbol{p}^* \\ &+ x^*\tilde{x}^*(\boldsymbol{u}\cdot\mathbf{1}_{\mathsf{z}})\{(\boldsymbol{\delta}\cdot\boldsymbol{p}^*)\boldsymbol{\epsilon}\circ\boldsymbol{y}^*\circ\mathbf{1}_{\mathsf{x}} \\ &+ (\boldsymbol{\epsilon}\cdot\boldsymbol{p}^*)\boldsymbol{\epsilon}\circ\boldsymbol{x}^*\circ\mathbf{1}_{\mathsf{y}} + (\boldsymbol{\delta}\circ\boldsymbol{\epsilon}\circ\boldsymbol{y}^*\cdot\mathbf{1}_{\mathsf{x}})\boldsymbol{p}^*\},\end{aligned} \tag{14}$$

$$\begin{aligned}\dot{\boldsymbol{y}}^{(2)} = &+(y^*-\tilde{y}^*)(\boldsymbol{u}\cdot\mathbf{1}_{\mathsf{z}})\boldsymbol{\delta}\circ\boldsymbol{\epsilon}\circ\boldsymbol{p}^* \\ &- y^*\tilde{y}^*(\boldsymbol{u}\cdot\mathbf{1}_{\mathsf{z}})\{(\boldsymbol{\delta}\cdot\boldsymbol{p}^*)\boldsymbol{\delta}\circ\boldsymbol{y}^*\circ\mathbf{1}_{\mathsf{x}} \\ &+ (\boldsymbol{\epsilon}\cdot\boldsymbol{p}^*)\boldsymbol{\delta}\circ\boldsymbol{x}^*\circ\mathbf{1}_{\mathsf{y}} + (\boldsymbol{\delta}\circ\boldsymbol{\epsilon}\circ\boldsymbol{x}^*\cdot\mathbf{1}_{\mathsf{y}})\boldsymbol{p}^*\},\end{aligned} \tag{15}$$

with $\boldsymbol{x}^* := (x^*, x^*, \tilde{x}^*, \tilde{x}^*)$, $\boldsymbol{y}^* := (y^*, \tilde{y}^*, y^*, \tilde{y}^*)$, $\boldsymbol{p}^* := \boldsymbol{x}^* \circ \boldsymbol{y}^*$, $\mathbf{1}_{\mathsf{x}} := (+1, +1, -1, -1)$, $\mathbf{1}_{\mathsf{y}} := (+1, -1, +1, -1)$, and $\mathbf{1}_{\mathsf{z}} := \mathbf{1}_{\mathsf{x}} \circ \mathbf{1}_{\mathsf{y}}$. Eqs. (12)-(15) are derived by considering small changes in the stationary condition $\boldsymbol{p}^{\mathrm{st}} = \boldsymbol{M}\boldsymbol{p}^{\mathrm{st}}$ for deviations of $\boldsymbol{\delta}$ and $\boldsymbol{\epsilon}$ (see Appendix B.1 and B.2 for the detailed calculation). By this, we can avoid a direct calculation of $\boldsymbol{p}^{\mathrm{st}}$, which is hard to be obtained.

5 Experimental Findings

5.1 Simulation and Low-Order Approximation

From the obtained dynamics, i.e., Eqs. (12)-(15), we interpret the learning dynamics in detail. In the first-order dynamics, multi-memory learning is no more than a simple extension of the zero-memory one. Indeed, the zero-memory learning draws an elliptical orbit given by Hamiltonian as the conserved quantity [30, 14]. Eqs. (12) and (13) mean that the multi-memory dynamics also draw similar elliptical orbits for each pair of x_i and y_i. In other words, the dynamics are given by a linear flow on a four-dimensional torus. Because no interaction occurs between the pair of i and i' such that $i \neq i'$, the dynamics of the multi-memory learning for each state are qualitatively the same as learning without memories. Fig. 2 shows the time series of the multi-memory learning dynamics near the Nash equilibrium in an example of a two-action zero-sum game, the penny-matching game ($u_1 = u_4 = 1$, $u_2 = u_3 = -1$). The experimental trajectories are generated by the Runge-Kutta fourth-order method of Eq. (10) (see Appendix B.3 for details), while the approximated trajectories are by the Runge-Kutta fourth-order method for the first- (Eqs. (12) and (13)), the second- (Eqs. (14) and (15)), and the third-order approximations (in Appendix B.2). The step-size is 10^{-2} in common. The top-left panel in the figure shows that the dynamics roughly draw a circular orbit for each state and are well approximated by the first-order dynamics of Eqs. (12) and (13). However, the top-right panel, where a sufficiently long time has passed, shows that the dynamics deviate from the circular orbits.

Such deviation from the circular orbits is given by higher-order dynamics than Eqs. (12) and (13). In the second-order dynamics given by Eqs. (14) and (15), the multi-memory learning is qualitatively different from the zero-memory one. Indeed, Eqs. (14) and (15) obviously mean that interactions occur between the pair of i and i' such that $i \neq i'$. Thereby, the dynamics of multi-memory learning become much more complex than that of zero-memory learning. In practice, no Hamiltonian function, denoted by $H^{(2)}$, exists in the second-order dynamics, as different from the first-order one. One can check this by calculating $\partial \dot{x}_i^{(2)}/\partial \epsilon_{i'} + \partial \dot{y}_{i'}^{(2)}/\partial \delta_i \neq 0$ for i and $i' \neq i$, if assuming that Hamiltonian should satisfy $\dot{\boldsymbol{x}}^{(2)} = +\partial H^{(2)}/\partial \boldsymbol{\epsilon}$ and $\dot{\boldsymbol{y}}^{(2)} = -\partial H^{(2)}/\partial \boldsymbol{\delta}$. Thus, the multi-memory dynamics might not have any conserved quantities and not draw any closed trajectory. Indeed, the right panels in Fig. 2 show that the dynamics tend to diverge from the Nash equilibrium. This divergence from the Nash equilibrium is surprising because zero-memory learning in zero-sum games always has a closed trajectory and keeps the Kullback-Leibler divergence from the Nash equilibrium constant [31, 14]. Here, note that we need the third-order dynamics to fit the experimental dynamics well, as seen by comparing the middle-right and lower-right panels in Fig. 2. The error between