POISSON-SAMPLED BEST-RESPONSE DYNAMICS: RATES FOR THE LYAPUNOV FUNCTION AT ALL TIMES, AND AN IMPLEMENTATION FOR THE SWARM DECEPTION ENVIRONMENT

Fouad Lbakali[1], Mohammed El Houcine Ayoubi[1], Alexandre Reiffers-Masson[1*]
[1]*IMT Atlantique, Department of Computer Science, Brest, France*

BACKGROUND

We consider a continuous zero-sum game defined by compact convex strategy sets X, Y and a continuous payoff function $U : X \times Y \to \mathbb{R}$, which is concave in x and convex in y. The key quantity in analyzing the convergence of such games is the Lyapunov function $V(x, y)$, defined as the difference between the maximum potential gain of Player 2 and the minimum potential loss of Player 1:

$$A(y) = \max_{x \in X} U(x, y), \qquad B(x) = \min_{y \in Y} U(x, y), \qquad V(x, y) = A(y) - B(x).$$

V is convex and nonnegative, satisfying $V(x, y) = 0$ if and only if (x, y) is a saddle point [2].

The classical continuous-time Best-Response (BR) dynamics is described by the differential inclusion:

$$\dot{x} \in \mathrm{BR}_1(y) - x, \quad \dot{y} \in \mathrm{BR}_2(x) - y.$$

Solutions to this system satisfy $V'(t) \leq -V(t)$, guaranteeing exponential convergence $V(t) \leq e^{-t}V(0)$ for all $t \geq 0$. However, this theoretical result relies on the assumption that agents can compute and update their strategies continuously at every instant. In practical multi-agent systems and robotics, this is computationally prohibitive. This limitation motivates the need for a dynamics model based on discrete, asynchronous updates.

AIM

This work analyzes a Poisson-sampled Best-Response dynamic. We model the update times as a stochastic process rather than a continuous flow. The process is defined as follows:

1. Updates occur at jump times $0 < T_1 < T_2 < \dots$ generated by a homogeneous Poisson process $N(t)$ with rate λ.
2. At each jump T_k, agents compute a "frozen" best-response direction based on the current state:
$$\alpha_k \in \mathrm{BR}_1(y(T_k^-)) - x(T_k^-), \quad \beta_k \in \mathrm{BR}_2(x(T_k^-)) - y(T_k^-).$$
3. Between jumps ($T_k \leq t < T_{k+1}$), agents evolve deterministically along these directions:
$$\dot{x}(t) = h(t - T_k)\alpha_k, \quad \dot{y}(t) = h(t - T_k)\beta_k,$$
where $h : [0, \infty) \to \mathbb{R}_+$ is a decreasing and integrable function. This function modulates the momentum of the frozen update, ensuring that the impact of an old decision fades over time.

The primary aim of this research is to establish a convergence rate for this piecewise-deterministic Markov process. Specifically, we seek to answer: Does this practical, sampled system retain the exponential convergence of the ideal continuous dynamics? We aim to derive a bound for $\mathbb{E}[V(x(t), y(t))]$ that exhibits exponential decay up to a controllable error term dependent on the sampling rate λ.

METHODS

To prove the convergence bound, we treat the Poisson-sampled dynamics as a Stochastic Approximation (SA) of the ideal differential inclusion (DI). Our methodology consists of three logical steps:

1. **Sampled Trajectories as SA:** We interpret the piecewise evolution of the agents as an Euler discretization with random step sizes $s_k = \int_0^{T_{k+1}-T_k} h(u)du$. Under standard hypotheses (Lipschitz continuity of the vector field and bounded growth), Gast-Gaujal's Theorem 1 [1] ensures that the interpolated trajectory converges in probability to the solution of the DI as $\lambda \to \infty$.

* Corresponding author. E-mail: alexandre.reiffers-masson@imt-atlantique.fr

2. **Uniform Error Bound (OSL):** To quantify the distance between the sampled trajectory $X^{\mathrm{SA}}(t)$ and the ideal trajectory $x^{\mathrm{DI}}(t)$, we utilize the One-Sided Lipschitz (OSL) property of the Best-Response dynamics. Using Theorem 4 in [1], we derive an explicit bound on the deviation:

$$\mathbb{P}\left(\sup_{t\in[0,T]} \|X^{\mathrm{SA}}(t) - x^{\mathrm{DI}}(t)\| \geq C_T\sqrt{\gamma} + \delta\right) \leq \frac{c}{\gamma},$$

where $\gamma = 1/\lambda$.

3. **Lyapunov Transfer:** Finally, we transfer the exponential decay property from the ideal system to the sampled one. Since V is Lipschitz continuous on compact sets, the error in the trajectory $\|X^{\mathrm{SA}} - x^{\mathrm{DI}}\|$ translates linearly to an error in the Lyapunov value. This allows us to bound $V(X^{\mathrm{SA}}(t))$ by the ideal decay $e^{-t}V(0)$ plus a noise term derived from the stochastic approximation error.

EXPECTED RESULTS

Theoretical Convergence

We successfully derived an explicit bound for the expected value of the Lyapunov function. By decomposing the expectation based on the probability of the tracking error exceeding a threshold $\mathcal{E}$, and bounding $\mathbb{P}(\mathcal{E}) \leq \frac{\gamma bT}{\varepsilon^2}$, we obtained:

$$\mathbb{E}[V(Y^{SA}(t))] \leq \underbrace{e^{-t}V(0)}_{\text{Ideal decay}} + L_V e^{LT}\sqrt{\text{Error}(\gamma)} + V_{\max}\frac{\gamma bT}{\varepsilon^2}, \qquad t < T$$

Here, L_V is the Lipschitz constant of V. This result confirms that for a sufficiently high sampling rate (small γ, high λ), the Poisson-sampled dynamics tend to have the same bound as the ideal solution, but with a noise floor determined by γ.

Application: Swarm Deception

We apply this theoretical framework to a Swarm Deception environment. This scenario is modeled as a Zero-Sum game where:

- Player 1 (The Swarm): Controls N drones attempting to reach a target while masking their true intent/positions.
- Player 2 (The Observer): Attempts to correctly estimate the positions/intent of the swarm based on partial observations.

RL Implementation of Poisson Updates

To formulate this work in the RL framework instead of assuming continuous-time learning, where both players continuously adjust their strategies toward instantaneous best responses, we introduce a *stochastic update mechanism* based on a homogeneous Poisson process. This process serves as a random update clock that determines the discrete instants at which best-response updates occur. Meaning Instead of standard step-by-step updates, we use a Poisson-triggered policy:

1. The environment maintains a global Poisson clock.
2. Agents act using "frozen" action primitives between clock ticks.
3. Policy updates (gradient steps) and decision changes occur only at the discrete Poisson events.

Let X and Y denote the compact strategy spaces of the two players introduced earlier. In RL terminology, the players' stochastic policies at stage t are denoted by

$$\pi_t^1(\cdot \mid s_t, o_{0:t-1}) \quad \text{and} \quad \pi_t^2(\cdot \mid o_{0:t}),$$

where s_t represents the (possibly aggregated) state available to Player 1, and $o_{0:t}$ is the observation history available to Player 2.

Model Architecture and Environment

The environment uses a high-fidelity physics engine (see Figure 1). The state space includes drone velocities (linear/angular), orientation (quaternions), and relative target vectors. The reward structure reflects the zero-sum competition:

$$r_t = r_t^{\text{nav}} - \text{MSE}(s_t^{\text{estim}}, s_t)$$

where the Swarm gains reward from navigation r_t^{nav} and high estimation error (MSE), while the Observer is penalized by the MSE.

Figure 2 illustrates the training architecture. We expect simulation results to demonstrate that agents trained with Poisson sampling converge to robust deceptive strategies (Nash equilibria) similar to continuous-time agents, but with significantly reduced computational overhead regarding decision frequency.

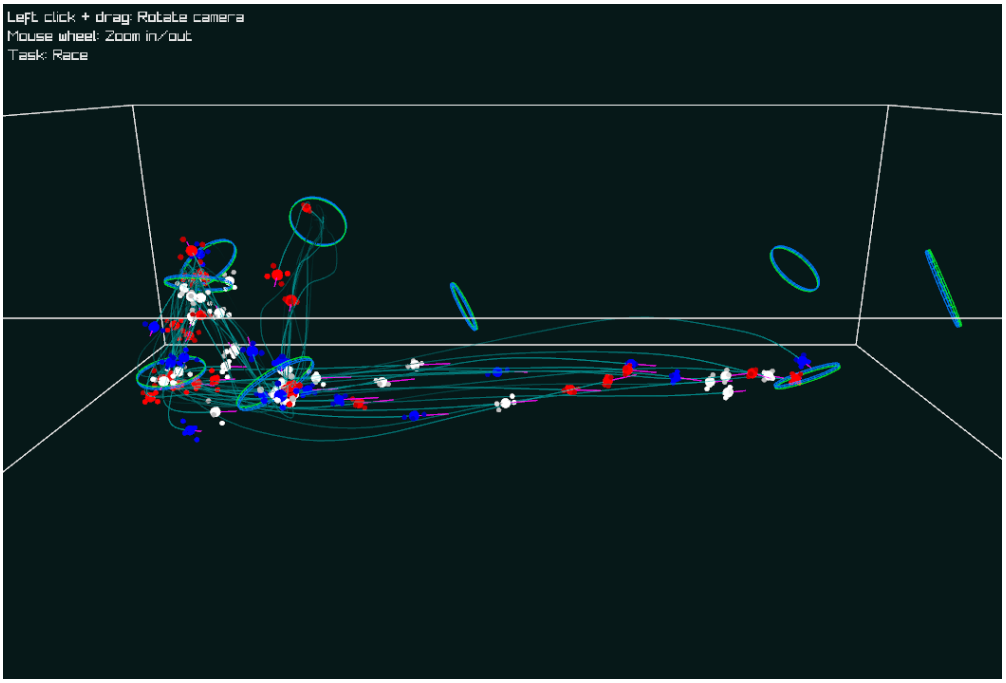

Figure 1. The Swarm Deception Environment. Drones must navigate complex 3D pathways (colored markers) while minimizing detection by an observer.

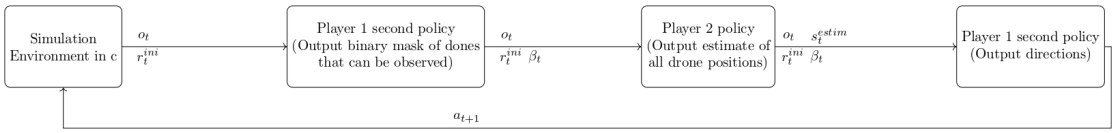

Figure 2. RL Training Architecture. The diagram highlights the interaction between the Swarm Policy, the Estimator Policy.

- o_t is the observation vector for each drone in the swarm.
- r_t^{ini} is the initial reward computed for task completion: $r_t^{\text{ini}} = \text{current\_observation\_score} - \text{last\_obs\_score}$.
- β_t is a binary mask indicating which parts of the observation space are visible to player 2.
- s_t^{estim} is the estimate of the positions of all the swarm drones.
- r_t is the total reward.
- a_t represents the actions that should be taken.

References

[1] N. Gast and B. Gaujal. Markov chains with discontinuous drifts have differential inclusion limits. *Performance Evaluation*, 69(12):623–642, 2012.

[2] J. Hofbauer and S. Sorin. Best response dynamics for continuous zero-sum games. *Discrete and Continuous Dynamical Systems Series B*, 6(1):215, 2006.