

# Bivariate Trajectory–Undercurrent Theory (BTUT): Simplified Logic

Intuitive and Variable–Defined Edition

## 1. The Core Idea

BTUT is a scalable form of game theory. Instead of using heavy differential equations, it models interactions between many agents through simple local rules that spread like heat across a network.

Each agent only looks at its neighborhood, decides how to act based on what it sees, and updates smoothly over time. This makes the whole system computable in  $\mathcal{O}(\underline{N})$  time to be linear with population size.

## 2. The World: Agents and Network

We have a network of  $\underline{N}$  agents (players), represented as a graph  $G = (V, E)$ . Each agent  $i$  has:

- a strategy  $\underline{s}_i \in \{A, B\}$  - a degree  $\underline{k}_i$  (number of connections) - and a connection weight  $\underline{w}_i$  that measures how influential it is.

The weight depends on how “hub-like” the agent is:

$$w_i = \left( \frac{k_i}{k_{\max}} \right)^{\tau},$$

where  $\tau$  (tau) controls how much more important big hubs are. If  $\tau = 0$ , all agents are equal; higher  $\tau$  means hubs dominate.

## 3. Local Payoffs (Mini-Games)

Each connection (edge) between two agents represents one of three basic game types:

- Prisoner’s Dilemma (PD)
- Hawk–Dove (HD)
- Stag Hunt (SH)

Each game type  $g$  has: -  $\underline{u}_A^{(g)}, \underline{u}_B^{(g)}$ : base payoffs for players  $A$  and  $B$  -  $\underline{c}_A^{(g)}, \underline{c}_B^{(g)}$ : cost coefficients -  $\underline{f}^{(g)}(\underline{p}_t)$ : how cooperation strength depends on the global fraction of  $A$ -players  $\underline{p}_t$

Each agent's expected payoff under game  $g$  is:

$$U_i^{(g)} = \frac{1}{2} \left[ u_s^{(g)} + (p_t u_A^{(g)} + (1 - p_t) u_B^{(g)}) \right] f^{(g)}(p_t) - c_s^{(g)} u_s^{(g)}.$$

## 4. Kernel Coupling: How Influence Spreads

Each agent is connected to others through a kernel  $K_{ij}$ , which says how strongly  $i$  is influenced by  $j$ :

$$K_{ij} = e^{-\lambda d(i,j)}.$$

Here: -  $d(i, j)$ : distance between agents -  $\lambda$ : decay rate (how quickly influence fades)

If two agents are close,  $K_{ij}$  is near 1 (strong influence). If far apart,  $K_{ij}$  is near 0.

## 5. Edge Accumulation (Information Gathering)

Each agent sums up what it feels from its neighbors:

$$\Phi_i = \sum_{j \in \mathcal{N}(i)} K_{ij} \log(1 + \eta U_j),$$

where: -  $\Phi_i$ : accumulated influence at node  $i$  -  $\eta$ : scaling factor controlling how much of a neighbor's payoff is felt -  $\log(1 + \eta U_j)$ : smooths rapid changes so big swings don't explode

This is called the log-intensity kernel which converts complex feedback into stable local information.

## 6. Decision and Update

Every agent compares two global utilities:

$$\tilde{U}_A = \sum_i w_i k_i U_i(A, p_t), \quad \tilde{U}_B = \sum_i w_i k_i U_i(B, p_t).$$

Then, the fraction of  $A$ -players updates as:

$$p_{t+1} = \frac{1}{2} p_t + \frac{1}{2} \mathbb{E}[\tilde{U}_A > \tilde{U}_B].$$

This means:

- Take half of your current belief ( $p_t$ ),
- Add half of the new observed tendency (how often  $A$  performs better than  $B$ ).

It's a momentum-like update to be slow, smooth, and stable.

## 7. Hidden Dynamics (Trajectory and Undercurrent)

To describe slow invisible shifts, BTUT tracks two latent variables:

$$\theta_t \text{ (trajectory)}, \quad \psi_t \text{ (undercurrent)}.$$

They evolve as:

$$\begin{aligned}\theta_{t+1} &= \theta_t + \kappa_\theta(p_{t+1} - p_t), \\ \psi_{t+1} &= \psi_t + \kappa_\psi \log\left(\frac{\tilde{U}_A}{\tilde{U}_B}\right).\end{aligned}$$

## 8. Equilibrium and Interpretation

Eventually, the system settles to a steady state:

$$p_\star = \lim_{t \rightarrow \infty} p_t.$$

We know:

$$\frac{\partial p_\star}{\partial \gamma} > 0 \quad \text{and} \quad \frac{\partial p_\star}{\partial c_A^{SH}} < 0.$$

Cooperation grows when rewards rise ( $\gamma$ ) and falls when costs rise ( $c_A^{SH}$ ).

## 9. Why It Scales

Because every term ( $K_{ij}$ ,  $\Phi_i$ ,  $U_i$ ) depends only on local neighbors and additive kernels, the total computation grows linearly with the number of agents:

$$\text{Complexity: } \mathcal{O}(N).$$

No exponential explosion, making BTUT a practical framework for scalable game theory.