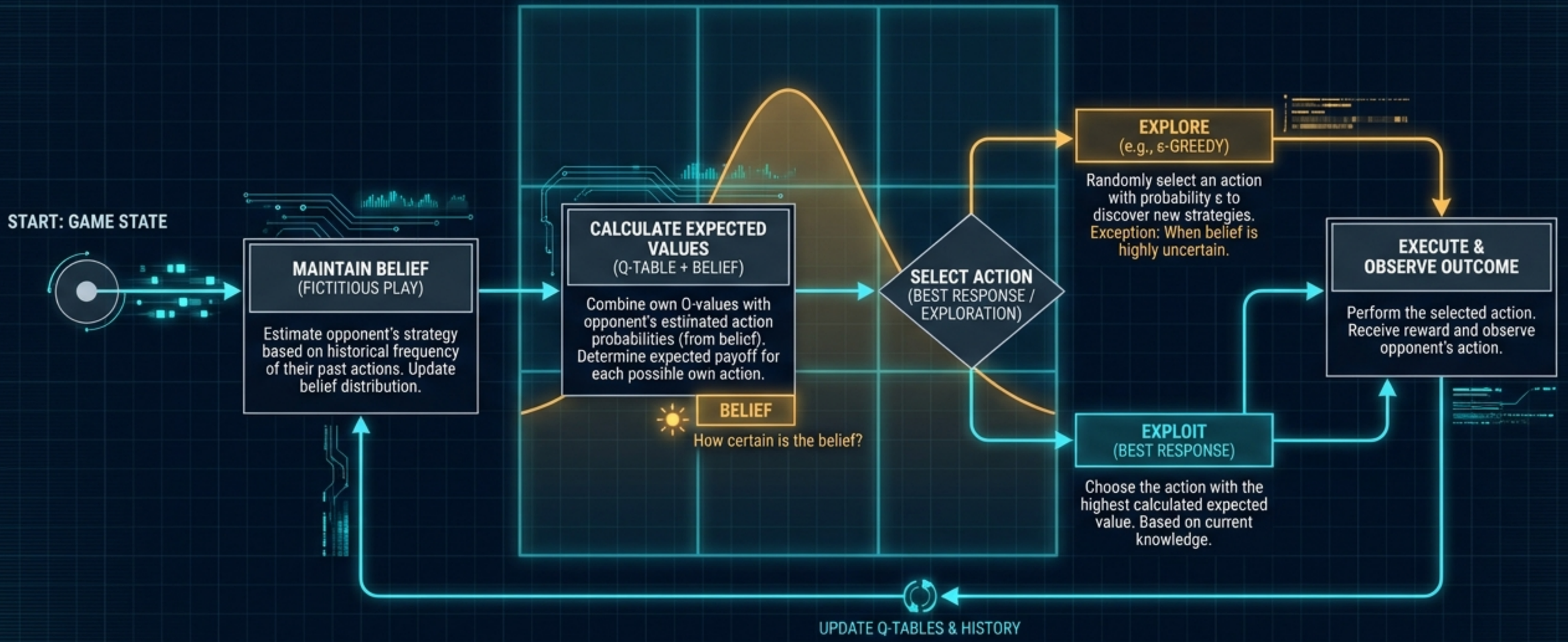
ACTION SELECTION IN JOINT Q-LEARNING

A STEP-BY-STEP BLUEPRINT FOR OPENING MOVES USING FICTITIOUS PLAY

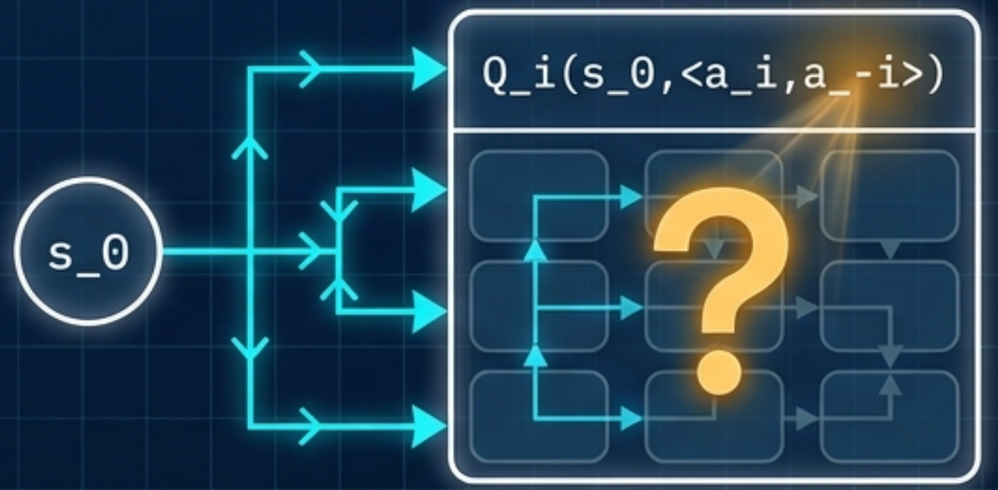


THE OPENING DILEMMA AT STATE s_0

STANDARD Q-LEARNING

At t=0, the agent observes state s_0 and simply selects the action with the highest estimated value in a vacuum.

JOINT Q-LEARNING



The agent maintains values for combinations of actions. To choose an optimal opening move, it must anticipate what the other agents (a_-i) will do simultaneously.

THE CHALLENGE: How does an agent select a_i when the outcome depends entirely on the unknown a_-i?

THE JAL-AM COGNITIVE PIPELINE

Before executing a move at t=0, the agent executes a rapid four-step internal synthesis:

1. Model the Opponent

Formulate a belief via Fictitious Play ($\hat{\pi}$).

2. Evaluate Joint Outcomes

Retrieve the Joint Q-Table (Q_i).

3. Calculate Expected Values

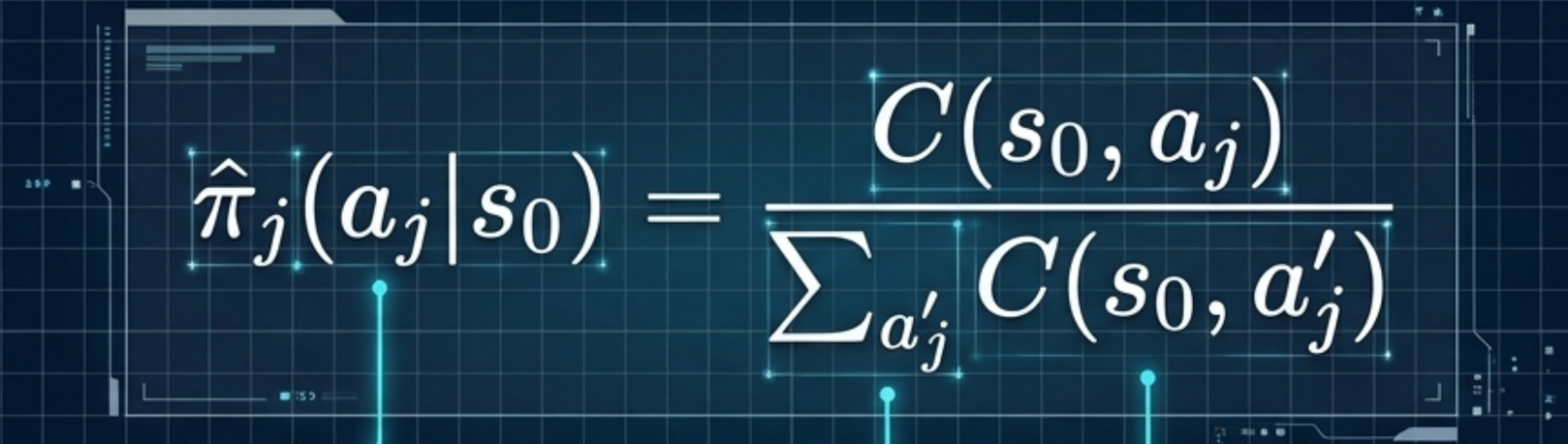
Marginalize probabilities to find Action Values (AV_i).

4. Execute Best Response

Make the final ϵ-greedy decision (a_i).

Step 1: Modeling the Opponent (Fictitious Play)

The agent asks: “What is my opponent likely to do in state s_?” It answers this by calculating the empirical distribution of the opponent's past actions.



$$\hat{\pi}_j(a_j|s_0) = \frac{C(s_0, a_j)}{\sum_{a'_j} C(s_0, a'_j)}$$

The Belief: The estimated probability that opponent j will take action a_j in state s_0

The Specific Count: How many times the opponent has historically taken this exact action in this exact state.

The Total Count: The total number of times this state has been visited.

The “Episode 1” Initialization

Established Training (History Exists)

$C > 0$. The belief $\hat{\pi}_j$ is derived from rich historical data, allowing the agent to accurately predict opponent tendencies.

The Absolute Beginning (No History)

Before any actions have been observed, historical counts are zero ($C = 0$). The agent initializes its belief as a Uniform Distribution:

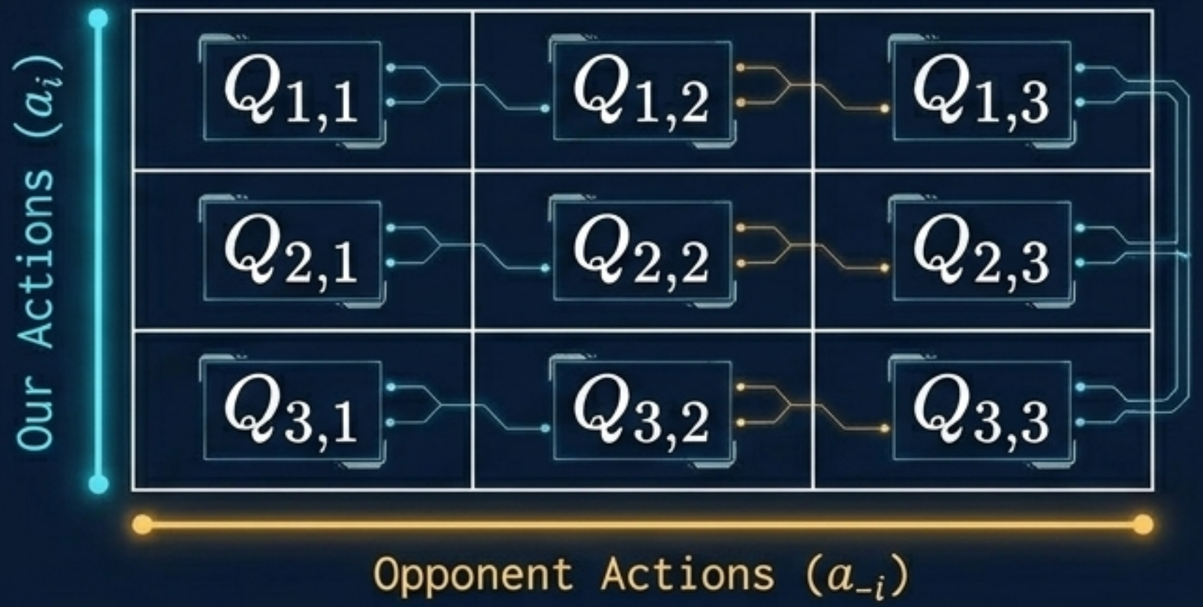
$$\hat{\pi}_j(a_j|s_0) = 1/|A_j|$$

Interpretation: It assumes all opponent actions are equally likely until proven otherwise.

Step 2: Retrieving Joint Value Estimates

Unlike standard Q-learning, which evaluates actions in isolation, the agent consults a matrix of expected returns for every combination of actions:

The Joint Q-Table: $Q_i(s_0, <a_i, a_{-i}>)$



- The agent only controls its own rows (a_i).
- The opponent controls the columns (a_{-i}).
- To succeed, the agent must navigate this matrix using the opponent belief formulated in Step 1.

Step 3: Computing the Expected Action Value (AV_i)

Because the exact column the opponent will choose is unknown, the agent marginalizes the opponent's actions using its Fictitious Play belief.

$$AV_i(s_0, a_i) = \sum_{a_{-i}} \left[Q_i(s_0, \langle a_i, a_{-i} \rangle) \times \prod_{j \neq i} \hat{\pi}_j(a_j | s_0) \right]$$

For each of my available actions, what is my average **expected reward,** weighted by how my opponent usually plays?

Step 4: The ε-Greedy Opening Move

With expected values calculated for every row, the agent makes its final selection for t=0.

In Episode 1, where all AV_i are likely initialized to 0, the first move is effectively effectively random, kickstarting the learning loop.

The Algorithmic Blueprint: From s_0 to a_i

1. Initial State:

Observe s_0 at t=0.

2. Fictitious Play:

Pull historical counts $C(s_0, a_j)$ → formulate opponent belief $\hat{\pi}_j$. (Uniform if Episode 1).

3. Matrix Look-up:

Retrieve Joint Q-values $Q_i(s_0, \langle a_i, a_{-i} \rangle)$.

4. Synthesis:

Marginalize: Multiply Joint Joint Q-values by opponent belief to find expected values AV_i.

5. Execution:

Apply ϵ-greedy selection to AV_i → Execute action a_i.