How Should I Trade?

Stationary Nash, Self-Confirming, and Learned Equilibria in Kiyotaki–Wright Exchange Economies

Thomas J. Sargent* Ziyue Yang†

June 15, 2026

Abstract

We study a pure exchange economy with bilateral anonymous trade and random matching. There are no markets, no credit, and no auctioneer, and double coincidences of wants are rare. Many profitable trades therefore require accepting a good for resale rather than for immediate consumption, giving rise to a *medium of exchange*. Kiyotaki and Wright [1989] characterize the steady-state Nash equilibria in which agents are *born knowing* equilibrium distributions. We work with the corresponding stationary Markov Nash equilibria in an average-reward formulation and ask whether such an equilibrium can *emerge* when agents do not observe equilibrium distributions and instead learn from bilateral encounters. We compare three equilibrium concepts along a spectrum: a *stationary Nash equilibrium*, a *restricted-information self-confirming equilibrium* [Fudenberg and Levine, 1993], and *agents who learn from past play*. Relative value iteration computes stationary Nash equilibria. A restricted-information fixed-point solver computes self-confirming equilibria. A tabular MuZero-inspired self-play algorithm [Schrittwieser et al., 2020] represents agents who plan with empirical models built from records of realized encounters.

Keywords: equilibrium, policy initialization, learning, law of large numbers, self-confirming equilibrium, search.

JEL classification: C63, C73, D51, D83, E40.

*New York University.
†Australian National University.

1 Introduction

> *Many newcomers to reinforcement learning may be disappointed to see that the theory and algorithms in this book are far removed from the dream portrayed in the popular media: reinforcement learning is often described as an "agent" interacting in a physical environment, and maturing as it gains experience. Unfortunately, given today's technologies, the process of "learning from scratch while you control" is unlikely to succeed outside of very special cases such as online advertising.*
>
> — Meyn [2022, p. 2]

"Money" serves as a unit of account, a store of value, and a medium of exchange; the medium-of-exchange function takes center stage in this paper. We consider a pure exchange economy in which agents meet pairwise and at random, with no markets and no credit. Double coincidences of wants are rare, so gains from trade require an agent to accept a good she does not herself want, trusting that she can later exchange it for one she does want.

Kiyotaki and Wright [1989] study steady-state Nash equilibria in an economy with three storable goods, where agents specialize in both consumption and production. In equilibrium, agents know others' trading strategies and an associated steady-state holding distribution; one or more goods emerge endogenously as a medium of exchange. In our average-reward formulation, which embeds Marimon et al. [1990]'s long-run-average-payoff criterion, the natural counterpart is a stationary Markov Nash equilibrium with an invariant holding distribution. We call this a stationary Nash equilibrium.

We study processes through which a stationary Nash equilibrium could emerge. We start from an initial situation whose economic environment and feasible actions are identical to Kiyotaki and Wright's. It differs in agents' information. We assume that agents are born without knowledge of Kiyotaki and Wright's equilibrium distributions and that they observe less about their trading partners than Kiyotaki and Wright assume. We ask whether a stationary Nash equilibrium can emerge when agents observe only bilateral encounters and learn from them.

This decentralized learning problem is challenging.[1] Any theory of statistical learning rests on a law of large numbers, whose guarantee — that an empirical average of payoffs converges to its true expectation — applies only to situations that are visited *infinitely often*. In our setting, an agent's experience is a sequence of random bilateral encounters whose distribution depends on the strategies and holdings of all other agents — objects that an agent neither observes nor controls. Every pre-trade state is a pair (k, j) of own and partner holdings, and the long-run frequency with which any such pair is visited is endogenous, depending on the agent's own policy and on the population-level holding distribution that her policy, together with everyone else's, generates. Situations in which the agent rarely holds a particular good, or rarely meets a partner carrying a particular good, accumulate too little statistical evidence. This matters because an agent's choices inside an equilibrium are shaped by her subjective beliefs about the consequences of actions that the equilibrium never calls on her to take. This creates an *exploration–exploitation* trade-off. An agent who always exploits her current best estimate never revisits and corrects wrong estimates for rarely encountered situations. An agent who explores uniformly wastes interactions on well-learned situations. Hence, the algorithm must steer attention toward under-visited situations in proportion to the uncertainty they carry.

These considerations motivate the computational tools we adopt. Silver et al. [2016] introduced AlphaGo, which combined a neural position evaluator with Monte Carlo tree search (MCTS) guided by a predictor upper-confidence-tree (PUCT) selection rule. The rule allocates simulations toward actions that are either highly valued or seldom tried, and its exploration component fades as visits accumulate (the exact form is given in Section 6.3). Silver et al. [2018] showed in AlphaZero that the same self-play loop, starting from random play and without any human data, could reach superhuman performance across chess, shogi, and Go. Self-play generates a non-stationary sequence of opponents, so the agent encounters situations its current policy cannot yet handle.

Schrittwieser et al. [2020] extended this architecture to settings in which an

[1] Like some games described by Kreps [1998], our setting is one in which agents cannot easily deduce an equilibrium *a priori* just from a description of the environment.

agent does not know transition dynamics. MuZero learns a latent-state dynamics model jointly with value and policy estimates, plans inside the learned model using MCTS with PUCT selection, and updates all components from self-play experience. This is natural in our setting: the partner-good distribution and reciprocal-acceptance probabilities that determine each agent's transition law are endogenous objects depending on everyone else's behavior, and are not known to any individual agent. We adapt a *tabular* version of MuZero's policy-improvement operator to our bilateral-matching economy. Because our state and action spaces are small enough for exact tabular representation, every value estimate, visit count, and policy weight is directly accessible. At each trade and consumption root, a PUCT bandit allocates simulations across actions. The visit counts update the tabular policy tables, while the maximum root action values update the value table across iterations. The policy is improved toward the action recommended by the completed search. This loop plays the same role as the Holland classifier's strength-updating rule used by Marimon et al. [1990]. Both learn average rewards from realized bilateral experience, but the PUCT-based loop handles the rare-visit problem more carefully.

A recurring theme is the role of *prompting*, by which we mean the initial policy and beliefs each agent starts with before any iteration of the solver or any round of learning begins. We use prompting in two ways.

A *directional prompt* directs agents toward an equilibrium they would otherwise miss. One example arises when an economy has more than one Nash equilibrium. From the default initial policy the solver always converges to the same equilibrium; an alternative equilibrium is reached only when the initial policy already prescribes the offers that the alternative would require. Such prompting leads agents to a different equilibrium that, like the first, admits no profitable deviations.

A *misdirectional prompt* has the opposite effect. We initialize agents' beliefs so that certain trade offers would be refused by their intended partners. Agents who hold this belief have no reason to make those offers, so they collect no data on whether the offers would in fact have been accepted. The no-trade pattern they settle into is consistent with their beliefs simply because nothing in the data refutes them. This is the structure of a self-confirming equilibrium: beliefs about events the

agent never observes are unconstrained. Such misdirection leads agents to outcomes that admit profitable deviations: an agent who experimented with one of the refused offers would discover that some are in fact accepted, and would gain by making them more often.

Neither prompt is arbitrary. We design them so that together they map out the boundary between outcomes that decentralized learning can reach and outcomes it cannot. We use them to show how the same physical economy can support different monetary conventions depending on what agents initially believe.

The computational results are organized around five economies. Three-good barter economies (A1, A2, B) test whether the learner recovers commodity-money conventions and, when multiple Nash equilibria coexist, which branch it selects. Economy C adds a fiat object — intrinsically valueless, non-consumable, held in fixed aggregate supply — whose acceptance is justified only by the expectation that others will accept it. Economy D is the five-type barter economy of Marimon et al. [1990], for which no analytic Kiyotaki–Wright characterization exists. In Economies A1 and B, the fully empirical learner reaches a commodity-money Nash equilibrium. In Economy A2 it reaches the fundamental holding distribution, but its policy has profitable deviations. In Economy C it reaches no Nash equilibrium: the reciprocal-spending links that sustain a fiat-money cycle lie off the equilibrium path of a competing self-confirming equilibrium, so they are never tested and never revised, and the learner settles on a low-trade non-Nash outcome instead. In Economy D the fully empirical learner converges to a stable non-Nash plateau. The source is the time-averaged update rule, which folds early, out-of-equilibrium holding distributions into beliefs — a fictitious-play-like structure [Brown, 1951, Robinson, 1951, Shapley, 1964].

We therefore study a second learner, the *known-environment* variant. It supplies each agent with the deterministic one-period consequences of her own actions — the payoff to consuming, the production good that follows consumption, the holding that results when a trade is refused — while still requiring her to learn the endogenous market objects: partner-good frequencies, reciprocal acceptance probabilities, and continuation values. The variant also evaluates the agent's own future decisions

greedily inside the planning horizon and centers search-derived value targets. These three changes together suffice for convergence to the Nash equilibrium in all five economies, including the fiat-money and five-good cases where the fully empirical learner fails. The comparison shows what this package of changes can do: a learner does not need to be told an equilibrium target, but it helps to know the deterministic rules of the physical environment and to plan in a way that is compatible with the average-reward criterion.

The contrast between our two learners echoes a distinction at the center of Camerer and Ho [1999]'s experience-weighted attraction (EWA) framework, which unifies reinforcement learning and belief-based learning in a single rule. Under EWA, an action's attraction rises with the payoff received when the action is taken and, with weight $\delta \in [0, 1]$, with the *foregone* payoff it would have earned had it been taken instead. At $\delta = 0$ only the realized payoff of the chosen action counts, as in choice reinforcement. At $\delta = 1$ foregone and realized payoffs carry equal weight, as in belief-based learning and weighted fictitious play. This distinction matters in our setting because the foregone payoff is not observable. An agent who declines to offer good k for good j never learns whether the partner would have accepted, nor what the resulting holding would later be worth. That counterfactual is itself an endogenous, off-path object. The fully empirical learner thus has no foregone-payoff information for the offers it does not make. The known-environment learner recovers some of it, because knowing the deterministic one-period rules lets it evaluate the immediate consequences of unchosen offers inside its planning model.

The rest of the paper is organized as follows. Section 2 describes the exchange environment and the individual actions. Section 3 gives the stationary Bellman equations and the laws of motion for holding distributions. Section 4 defines the restricted-information SCE concept. Section 5 lists the economies. Section 6 describes the equilibrium solvers and the tabular search learner. Section 7 reports the computational results. Section 8 offers concluding remarks.

2 Model Setup

2.1 Agents, Goods, and Actions

Let G be a finite set of storable goods, and let $I = \{1, \dots, N\} \subseteq G$ be the set of agent types, where $N = |I|$ is the number of types, with population shares $\nu_i \geqslant 0$ satisfying $\sum_{i \in I} \nu_i = 1$. Each type $i \in I$ is also the label of its consumption good: a type-i agent derives utility only from consuming good i, so the type index and the consumption-good index coincide. A production map $f : I \to G$ assigns to each type-i agent a good $f(i) \in G$ that she can produce, with $f(i) \neq i$. Immediately after consuming, a type-i agent produces one unit of good $f(i)$, which she carries into the next period. Goods are indivisible. An agent can store exactly one unit of exactly one good from one period to the next. Matching is random, bilateral, and independent of types.

Let $s : G \to \mathbb{R}_+$ be a one-period storage-cost function. Let $\bar{u} = (\bar{u}_i)_{i \in I}$ be a vector of nonnegative real numbers. The utility that a type-i agent gets from consuming good h is

$$u_i(h) = \bar{u}_i \cdot \mathbb{1}\{h = i\}.$$

Thus, a type i agent gets positive utility only from consuming good i. The indicator $\mathbb{1}\{h = i\}$ is well-defined because $h \in G$ and $i \in I \subseteq G$ are elements of the same set.

After trade, the agent decides whether to consume the good she carries. As in Marimon et al. [1990], agents can consume any carried good $h \in G$ and receive utility $u_i(h)$. The agent then produces $f(i)$. Such goods (those for which $u_i(h) = 0$ for all $i \in I$) may be consumed, traded, and stored.

Let $\mathcal{A}$ denote the (large) population of agents and let $i : \mathcal{A} \to I$ be the type assignment, with the population share of type i equal to ν_i. At each date t, the population $\mathcal{A}$ is partitioned into pairs uniformly at random, independently across t and of all agents' states. We write $p_t : \mathcal{A} \to \mathcal{A}$ for the resulting matching, so $p_t(a)$ is agent a's partner at date t. By construction, $p_t(a) \neq a$ (no self-matches) and $p_t(p_t(a)) = a$ (matching is symmetric). At the beginning of period t, agent a holds good $x_{at} \in G$ and is matched with partner $p_t(a)$. Sections 3–4 work in the continuum-of-agents limit, in which the partner's holding $x_{p_t(a),t}$ is independent of agent a's own state and is drawn from the aggregate holding distribution μ defined in Section 2.2.

The finite-population learning simulations of Section 6.3 use the same matching protocol with a population of size N.

The *pre-trade state* is

$$z_{at} = \left(x_{at}, x_{p_t(a),t}\right).$$

The first within-period decision is whether to offer trade:

$$\lambda_{at} = \begin{cases} 1, & \text{if agent } a \text{ offers to trade,} \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

Trade occurs if and only if both matched agents offer it. Hence post-trade holdings are

$$x_{at}^{+} = \begin{cases} x_{p_t(a),t}, & \text{if } \lambda_{at} = 1 \text{ and } \lambda_{p_t(a),t} = 1, \\ x_{at}, & \text{otherwise.} \end{cases} \tag{2}$$

When the two agents hold the same good, the exchange leaves both holdings unchanged. In the learning simulations, same-good matches are recorded as no-change transitions and are excluded from reciprocal-acceptance counts.

After the trade stage, the agent chooses whether to consume the good she exits with:

$$\gamma_{at} = \begin{cases} 1, & \text{if agent } a \text{ consumes } x_{at}^{+}, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

When an agent consumes, she immediately produces $f(i(a))$. Therefore

$$x_{a,t+1} = \begin{cases} f\left(i(a)\right), & \text{if } \gamma_{at} = 1, \\ x_{at}^{+}, & \text{if } \gamma_{at} = 0. \end{cases} \tag{4}$$

Equations (1)–(4) together describe the one-period law of motion of an individual agent's state.

The type-specific state spaces are

$$\underbrace{Z_i = G \times G,}_{\text{pre-trade}} \qquad \underbrace{H_i = G}_{\text{post-trade}} .$$

There are $|G|^2$ pre-trade states and $|G|$ post-trade states per type. For the economies studied in this paper, these state spaces are small enough for exact fixed-point computation.

Exchange and consumption action spaces are

$$\Lambda_i(k,j) \subseteq \{0,1\}, \qquad \Gamma_i(h) \subseteq \{0,1\},$$

for all $k, j, h \in G$. By default, $\Lambda_i(k,j) = \Gamma_i(h) = \{0,1\}$. Specific economies may restrict these sets (e.g. goods that cannot be consumed have $\Gamma_i(h) = \{0\}$). Section 5 describes specifications for the economies that we shall study in this paper.

If an agent of type i does *not* consume, she carries the current good into the next period and pays its storage cost. If she *does* consume, she receives utility $u_i(h)$ (which equals $\bar{u}_i$ when $h = i$ and zero otherwise), produces one unit of good $f(i)$, and pays the storage cost $s(f(i))$. The corresponding period reward for a type-i agent who exits trade with good h and makes consumption decision $\gamma \in \{0,1\}$ is

$$r_i(h,\gamma) = \gamma\big[u_i(h) - s\big(f(i)\big)\big] + (1-\gamma)\big[-s(h)\big]. \tag{R}$$

2.2 Policies and Distributions

To describe a stationary Nash equilibrium, let

$$\alpha_i(k,j) \in [0,1]$$

be the probability that a type-i agent offers to trade when she holds k and meets a trading partner who holds j. Let

$$\beta_i(h) \in [0,1]$$

be the probability that a type-i agent consumes when she exits trade with h (we use β_i rather than β to avoid confusion with the discount factor in Kiyotaki and Wright). The objects α and β are our Markov-policy versions of the trading rules in Kiyotaki and Wright [1989] and of the exchange and consumption classifiers in Marimon et al. [1990]. An equilibrium also pins down the induced cross-sectional holding

distributions and meeting probabilities, which we define next. Define a beginning-of-period holding distribution for type i as

$$\mu_i(k) := \Pr\big(\text{type-}i \text{ agent holds } k \text{ at start of period}\big).$$

The probability $\mu_i(k)$ is the fraction of type-i agents who hold good k. Because μ_i is a probability distribution over holdings for type i, it satisfies

$$\mu_i(k) \geqslant 0, \qquad \sum_{k \in G} \mu_i(k) = 1 \qquad \text{for each } i \in I.$$

The aggregate holding distribution is

$$\mu(k) := \sum_{i \in I} \nu_i \, \mu_i(k),$$

where ν_i is the population share of type i agents. The fraction of holders of good k who are type i consumers is

$$\eta_i(k) := \Pr(i \mid k) = \frac{\nu_i \, \mu_i(k)}{\mu(k)}. \tag{B1}$$

In the notation of Marimon et al. [1990], these objects correspond to

$$\mu_i(k) \equiv \pi_i^h(k), \qquad \mu(k) \equiv \pi^h(k), \qquad \eta_i(k) \equiv \pi(i \mid k).$$

They also correspond to the type-specific and aggregate stationary holding probabilities denoted by p_{ij} and their aggregates in Kiyotaki and Wright [1989]. The specific values of ν_i for each economy are given in Section 5.[2]

If a type-i agent holds k and meets someone holding j, then her offer $k \to j$ to exchange good k for good j is reciprocated with probability

$$Q(k, j) = \sum_{\ell \in I} \eta_\ell(j) \, \alpha_\ell(j, k). \tag{B2}$$

When $\mu(j) = 0$, we set $Q(k, j) = 0$ by convention. An agent does not observe her trading partner's type, only the good j that the partner holds, so $Q(k, j)$ averages

[2] When $\mu(k) = 0$, good k is not held by any agent in the stationary distribution; in that case $\eta_i(k)$ is undefined but does not affect any on-path expectation, since every term involving $\eta_i(k)$ is weighted by $\mu(k) = 0$.

willingnesses to trade $j \to k$ across types conditional on holding j. The probability defined in equation (B2) is the stationary reciprocal-trade probability implicit in the Kiyotaki–Wright equilibrium calculation and in the classifier simulations of Marimon et al. [1990].

3 Stationary Nash Equilibrium

We restrict our attention to stationary Markov strategies and evaluate them under a long-run average reward criterion. Kiyotaki and Wright [1989] characterize steady-state Nash equilibria in trading strategies under discounted expected utility. In our average-reward formulation, the corresponding benchmark is a stationary Markov Nash equilibrium with an invariant holding distribution. For brevity, we call this object a stationary Nash equilibrium (SNE); it is this paper's rational-expectations benchmark, in which agents optimize against the objective stationary environment. Marimon et al. [1990] keep the same physical exchange environment but replace fully rational agents by Holland classifier systems of competing if-then statements whose strengths are designed to approximate long-run average rewards.[3] We say that a type-i agent's policy is optimal if no alternative stationary policy yields higher long-run average reward against the stationary environment generated by the interactions of all agents.

Let Σ_i denote the set of stationary Markov policies for type i. A policy $\sigma_i \in \Sigma_i$ is a pair

$$\sigma_i = \big(\sigma_i^e, \sigma_i^c\big),$$

where σ_i^e maps a pre-trade state $(k, j) \in Z_i$ into a probability distribution over exchange actions and σ_i^c maps a post-trade state $h \in H_i$ into a probability distribution over consumption actions. Writing $\Delta(B)$ for the set of probability distributions on a finite action set B, this means

$$\sigma_i^e(\cdot \mid k, j) \in \Delta\big(\Lambda_i(k, j)\big), \qquad \sigma_i^c(\cdot \mid h) \in \Delta\big(\Gamma_i(h)\big).$$

[3] For presentations of Holland classifiers, see Holland [1975, 1986] and Goldberg [1989].

Thus, $\sigma_i^e(\lambda \mid k, j)$ is the probability of choosing exchange action λ after observing the pre-trade state (k, j), and $\sigma_i^c(\gamma \mid h)$ is the probability of choosing consumption action γ after exiting trade with good h.

In this paper, we use the convention that

$$\alpha_i(k, j) \equiv \sigma_i^e(1 \mid k, j), \qquad \beta_i(h) \equiv \sigma_i^c(1 \mid h).$$

When $\Gamma_i(h) = \{0\}$ for some good h (e.g. a non-consumable good), this automatically gives $\beta_i(h) = 0$.

For a given stationary environment (μ, Q) that confronts agents and a given policy profile $\sigma = (\sigma_i, \sigma_{-i})$, our first task is to define the one-period expected flow utility generated by σ_i. If the agent exits the trade stage with good h, the expected one-period reward from the consumption decision is

$$\bar{r}_i^{\sigma_i^c}(h) := \sum_{\gamma \in \Gamma_i(h)} \sigma_i^c(\gamma \mid h)\, r_i(h, \gamma).$$

If the pre-trade state is (k, j), then the expected flow utility under policy σ_i is

$$\bar{r}_i^{\sigma}(k, j) := \sum_{\lambda \in \Lambda_i(k,j)} \sigma_i^e(\lambda \mid k, j) \left[(1 - \lambda)\, \bar{r}_i^{\sigma_i^c}(k) + \lambda \Big(Q(k, j)\, \bar{r}_i^{\sigma_i^c}(j) + \big(1 - Q(k, j)\big)\, \bar{r}_i^{\sigma_i^c}(k) \Big) \right].$$

This expression integrates over an agent's exchange decision, the probability that an offer is reciprocated, and the post-trade consumption decision.

The representative type-i agent chooses a stationary Markov policy σ_i to maximize long-run average flow utility. Let $Z_t = (X_t, J_t) \in G \times G$ be her pre-trade state at date t: X_t is her own beginning-of-period holding, and J_t is the partner's holding. Under stationary random matching, J_t is drawn from the aggregate holding distribution μ independently of X_t. Given σ_i and the cross-sectional environment (μ, Q), the within-period transitions (1)–(4) make X_t a Markov chain on G. We write $\mathbb{P}^{\sigma,\mu,Q}$ for the law of $\{Z_t\}_{t \geqslant 0}$ under this construction and $\mathbb{E}^{\sigma}$ for the corresponding expectation, suppressing the (μ, Q) dependence. The agent's objective is

$$\max_{\sigma_i \in \Sigma_i} \bar{U}_i\big(\sigma_i;\, \mu, Q\big) := \liminf_{T \to \infty} \frac{1}{T}\, \mathbb{E}^{\sigma} \left[\sum_{t=0}^{T-1} \bar{r}_i^{\sigma}(Z_t) \right]. \tag{O}$$

Given the stationary environment (μ, Q), type i faces a finite-state average-reward Markov decision problem over holdings. For a fixed stationary policy σ_i, we work under the assumption that the induced type-i holding chain is unichain, that is, has a unique recurrent set, so the long-run average reward does not depend on the initial holding.[4] Under the unichain condition there exists a scalar *gain* g_i^σ for the fixed policy and a *bias function*, also called the *relative value function*, $V_i^\sigma : G \to \mathbb{R}$, that is unique up to an additive constant. The gain g_i^σ is the long-run average reward per period, and the bias V_i^σ records state-by-state deviations from that average. We fix the additive constant by choosing a reference good $k_i^{\text{ref}} \in G$ and imposing

$$V_i^\sigma(k_i^{\text{ref}}) = 0.$$

For a post-trade holding $h \in G$, the expected continuation value under σ_i is

$$W_i^\sigma(h) := \sum_{\gamma \in \Gamma_i(h)} \sigma_i^c(\gamma \mid h) \Big[r_i(h,\gamma) + \gamma V_i^\sigma\big(f(i)\big) + (1-\gamma) V_i^\sigma(h) \Big].$$

For an initial holding $k \in G$, the average-reward value under σ_i for a type-i agent is

$$g_i^\sigma + V_i^\sigma(k) = W_i^\sigma(k) + \sum_{j \in G} \mu(j)\, \sigma_i^e(1 \mid k, j)\, Q(k,j) \left[W_i^\sigma(j) - W_i^\sigma(k) \right].$$

Because g_i^σ does not depend on the initial holding, it equals the criterion $\bar{U}_i(\sigma_i; \mu, Q)$ defined in (O), and the two symbols are used interchangeably throughout the paper. Equivalently,

$$V_i^\sigma(k) = W_i^\sigma(k) + \sum_{j \in G} \mu(j)\, \sigma_i^e(1 \mid k, j)\, Q(k,j) \left[W_i^\sigma(j) - W_i^\sigma(k) \right] - g_i^\sigma.$$

To deduce Bellman equations that are satisfied by optimal policies, we replace policy weights with point-wise maximization over feasible actions. Let g_i denote the optimal gain and $V_i(k)$ the optimal bias, or relative value, when the agent enters the period holding good k. The optimal post-trade value is

$$W_i(h) = \max_{\gamma \in \Gamma_i(h)} \Big\{ \gamma \big[u_i(h) - s\big(f(i)\big) + V_i\big(f(i)\big) \big] + (1-\gamma) \big[-s(h) + V_i(h) \big] \Big\}. \tag{B3}$$

[4] We verify the unichain property numerically for the Nash benchmarks computed in Section 6. Some self-confirming policies reported in Section 7 induce chains with several recurrent classes. In those cases the reported stationary distribution places mass on only one class, and the consistency requirement that defines the equilibrium is likewise restricted to that support.

Then the average-reward Bellman equation is

$$g_i + V_i(k) = \sum_{j \in G} \mu(j) \max_{\lambda \in \Lambda_i(k,j)} \Big\{ (1-\lambda) W_i(k) + \lambda \big[Q(k,j)\, W_i(j) + \big(1 - Q(k,j)\big)\, W_i(k) \big] \Big\}. \quad \text{(B4)}$$

The Bellman optimality conditions (B3)–(B4) characterize point-wise best responses, reducing the search over actions to value comparisons once V_i and the environment (μ, Q) are known. For exchange, when both actions are feasible (i.e. $\Lambda_i(k,j) = \{0,1\}$),

$$\alpha_i(k,j) \in \begin{cases} \{1\}, & \text{if } Q(k,j)\big[W_i(j) - W_i(k)\big] > 0, \\ \{0\}, & \text{if } Q(k,j)\big[W_i(j) - W_i(k)\big] < 0, \\ [0,1], & \text{if } Q(k,j)\big[W_i(j) - W_i(k)\big] = 0, \end{cases} \quad \text{(BR-e)}$$

with $\alpha_i(k,j)$ pinned to the unique feasible action whenever $\Lambda_i(k,j)$ is a singleton. For consumption,

$$\beta_i(h) \in \begin{cases} \{1\}, & \text{if } u_i(h) - s\big(f(i)\big) + V_i\big(f(i)\big) > -s(h) + V_i(h), \\ \{0\}, & \text{if } u_i(h) - s\big(f(i)\big) + V_i\big(f(i)\big) < -s(h) + V_i(h), \\ [0,1], & \text{if } u_i(h) - s\big(f(i)\big) + V_i\big(f(i)\big) = -s(h) + V_i(h). \end{cases} \quad \text{(BR-c)}$$

with the extra feasibility restriction $\beta_i(h) = 0$ for any good h with $\Gamma_i(h) = \{0\}$.

Given policies (α, β) and the beginning-of-period holding distribution μ, the within-type exchange flow from k to j is

$$F_i(k,j) = \mu_i(k)\, \mu(j)\, \alpha_i(k,j)\, Q(k,j), \qquad F_i(k,k) = 0. \quad \text{(F1)}$$

The post-trade holding distribution for type i is

$$\mu_i^+(g) = \mu_i(g) - \sum_{j \neq g} F_i(g,j) + \sum_{k \neq g} F_i(k,g). \quad \text{(F2)}$$

After the consumption decision, next period's beginning-of-period holding distribution satisfies

$$\mu_i'(g) = \mathbb{1}\{g = f(i)\} \sum_{h \in G} \mu_i^+(h)\, \beta_i(h) + \mu_i^+(g)\, \big(1 - \beta_i(g)\big). \quad \text{(F3)}$$

Stationarity requires

$$\mu_i'(g) = \mu_i(g) \qquad \text{for all } i \in I \text{ and } g \in G. \tag{F4}$$

Define the law-of-motion map

$$F_\sigma : \prod_{i \in I} \Delta(G) \to \prod_{i \in I} \Delta(G), \qquad F_\sigma(\mu) = \mu'.$$

Given $\mu = (\mu_i)_{i \in I}$ and the policy profile $\sigma = (\alpha, \beta)$, the components μ_i' are computed from μ_i via (F1)–(F3) (use the aggregate $\mu(k) = \sum_i \nu_i \mu_i(k)$ and $\eta_i(k) = \nu_i \mu_i(k)/\mu(k)$). Then (F4) is the fixed-point equation $F_\sigma(\mu) = \mu$.

As in Kiyotaki and Wright [1989], we impose this stationarity restriction on the holding distribution. Marimon et al. [1990] also require the probability collection Π_t to be time-invariant and induced by the agents' classifier decisions together with random matching.

Definition 3.1 (Stationary Nash equilibrium)**.** A stationary Nash equilibrium is a collection

$$\Big(\{\mu_i\}_{i \in I}, \{\alpha_i\}_{i \in I}, \{\beta_i\}_{i \in I}, Q \Big)$$

such that:

(i) the reciprocal-trade probability Q is induced by (μ, α) through (B1)–(B2);

(ii) (α_i, β_i) solves the type-i Bellman problem (B3)–(B4) given the stationary environment (μ, Q);

(iii) μ satisfies the laws of motion (F1)–(F4);

(iv) the conditional type distribution is consistent: $\eta_i(k) = \dfrac{\nu_i \, \mu_i(k)}{\mu(k)}$ for $\mu(k) > 0$. (This condition follows from condition (i) and is included for reference.)

This is the stationary Markov version of the Kiyotaki–Wright Nash equilibrium. Marimon et al. [1990] represent the same stationary behavior with a probability collection Π and fixed classifier lists $\{D_a\}$.

4 Self-Confirming Equilibria

We take the stationary Nash equilibria described in Definition 3.1 as "complete-information, correct beliefs" benchmarks against which to measure outcomes under alternative assumptions about information and beliefs. In an SNE, a type-i agent knows the stationary environment summarized by the aggregate holding distribution μ and the reciprocal trade probability $Q(k, j)$. The agent solves the finite-state average-reward Bellman problem in Section 3. This section considers a different information structure, suited to settings in which agents have less information than a stationary Nash equilibrium requires.[5] Agents know their own types, their own holdings, their current trading partner's holdings, their own actions, their own realized trade outcomes, and their own realized rewards. They do not know the cross-sectional distribution μ, the conditional type distribution η, other agents' policies, or the actual reciprocal trade probability Q. We adopt self-confirming equilibrium (SCE) in the sense of Fudenberg and Levine [1993]; connections between self-confirming and Nash equilibrium resemble those analyzed by Fudenberg and Kreps [1995].

A stationary restricted-information policy for type i is still a pair

$$\sigma_i = \big(\sigma_i^e, \sigma_i^c\big),$$

where

$$\sigma_i^e(\cdot \mid k, j) \in \Delta\big(\Lambda_i(k, j)\big), \qquad \sigma_i^c(\cdot \mid h) \in \Delta\big(\Gamma_i(h)\big).$$

The difference from the complete-information benchmark is not the domain of the policy, but the information used to evaluate actions.

Before the exchange decision, the agent observes only

$$\mathcal{I}_{a,t}^e = \big(i(a),\, x_{a,t},\, x_{p_t(a),t}\big) \in I \times G \times G,$$

and before the consumption decision she observes

$$\mathcal{I}_{a,t}^c = \big(i(a),\, x_{a,t}^+\big) \in I \times G.$$

[5] Bray and Kreps [1987] describe their conception of the difference between learning "about" and learning "within" an equilibrium.

When an agent declines to trade ($\lambda = 0$), she cannot tell whether the partner would have reciprocated had she offered, so her belief about the reciprocal-acceptance probability at that pre-trade state is not disciplined by data. This is the off-path-belief gap that the SCE concept formalizes.

4.1 Subjective stationary environments

A subjective stationary environment for type i is

$$\theta_i = (m_i, q_i) \in \Theta_i \equiv \Delta(G) \times [0,1]^{G \times G}.$$

Here $m_i(j)$ is type i's subjective probability of meeting a partner holding good j, and $q_i(k,j)$ is type i's subjective probability that an offer to exchange k for j is reciprocated. Both beliefs are indexed by i because each type learns from its own bilateral encounters; under random matching the objective marginal is μ for every type, but the SCE concept allows m_i to differ across types and only collapses to $m_i = \mu$ in the rational-expectations limit. The agent knows her own payoff function, her own production good $f(i)$, and the feasible action correspondences Λ_i and Γ_i.

For $h, k, j \in G$ and $\lambda \in \{0,1\}$, define a subjective post-trade transition kernel

$$T_q(h \mid k, j, \lambda) = \begin{cases} \mathbb{1}\{h = k\}, & \lambda = 0, \\ \big(1 - q(k,j)\big)\mathbb{1}\{h = k\} + q(k,j)\,\mathbb{1}\{h = j\}, & \lambda = 1. \end{cases}$$

Let

$$c_i(h, \gamma) = \begin{cases} f(i), & \gamma = 1, \\ h, & \gamma = 0, \end{cases}$$

be the next-period holding induced by post-trade holding h and consumption action γ.

Given a subjective environment $\theta_i = (m_i, q_i)$ and a stationary policy $\tilde{\sigma}_i$, the agent believes that her beginning-of-period holding evolves according to

$$\begin{aligned} P_i^{\tilde{\sigma}_i, \theta_i}(k' \mid k) = \sum_{j \in G} m_i(j) \sum_{\lambda \in \Lambda_i(k,j)} \tilde{\sigma}_i^e(\lambda \mid k, j) \sum_{h \in G} T_{q_i}(h \mid k, j, \lambda) \\ \times \sum_{\gamma \in \Gamma_i(h)} \tilde{\sigma}_i^c(\gamma \mid h)\, \mathbb{1}\big\{k' = c_i(h, \gamma)\big\}. \end{aligned} \tag{S1}$$

The corresponding subjective one-period expected reward is

$$R_i^{\tilde{\sigma}_i,\theta_i}(k) = \sum_{j\in G} m_i(j) \sum_{\lambda\in\Lambda_i(k,j)} \tilde{\sigma}_i^e(\lambda \mid k,j) \sum_{h\in G} T_{q_i}(h \mid k,j,\lambda) \sum_{\gamma\in\Gamma_i(h)} \tilde{\sigma}_i^c(\gamma \mid h)\, r_i(h,\gamma).$$

The type-i subjective objective is

$$\bar{U}_i\big(\tilde{\sigma}_i;\theta_i\big) := \liminf_{T\to\infty} \frac{1}{T}\, \mathbb{E}^{\tilde{\sigma}_i,\theta_i}\left[\sum_{t=0}^{T-1} R_i^{\tilde{\sigma}_i,\theta_i}(X_t)\right],$$

where $X_t \in G$ is the agent's beginning-of-period holding and the expectation is taken under the subjective Markov law generated by $(\tilde{\sigma}_i,\theta_i)$, with X_t evolving according to the transition kernel $P_i^{\tilde{\sigma}_i,\theta_i}$ in (S1).

Imposing the same unichain condition as in Section 3, now under the subjective Markov law $P_i^{\tilde{\sigma}_i,\theta_i}$, subjective optimality is described by an average-reward Bellman equation. There exists a scalar subjective gain g_i^θ and a bias function $V_i^\theta : G \to \mathbb{R}$, normalized as in Section 3 by $V_i^\theta(k_i^{\mathrm{ref}}) = 0$, such that

$$W_i^\theta(h) = \max_{\gamma\in\Gamma_i(h)} \Big\{ r_i(h,\gamma) + V_i^\theta\big(c_i(h,\gamma)\big) \Big\}, \tag{S2}$$

and

$$g_i^\theta + V_i^\theta(k) = \sum_{j\in G} m_i(j) \max_{\lambda\in\Lambda_i(k,j)} \sum_{h\in G} T_{q_i}(h \mid k,j,\lambda)\, W_i^\theta(h). \tag{S3}$$

A stationary policy σ_i is subjectively optimal under θ_i if $\sigma_i^c(\cdot \mid h)$ assigns probability only to maximizers in (S2) and $\sigma_i^e(\cdot \mid k,j)$ assigns probability only to maximizers in the exchange maximization in (S3).

Given a stationary policy profile σ, let the true reciprocal trade probability be

$$Q^\sigma(k,j) = \sum_{\ell\in I} \eta_\ell(j)\, \sigma_\ell^e(1 \mid j,k),$$

with the same convention as before when $\mu(j) = 0$.

Definition 4.1 (Restricted-information stationary self-confirming equilibrium)**.** A restricted-information stationary self-confirming equilibrium is a tuple

$$(\mu,\sigma,\theta) = \Big(\{\mu_i\}_{i\in I},\ \{\sigma_i\}_{i\in I},\ \{\theta_i\}_{i\in I}\Big), \qquad \theta_i = (m_i, q_i),$$

satisfying the following three conditions:

(i) Stationarity.[6] For each type i and each $k' \in G$,

$$\mu_i(k') = \sum_{k \in G} \mu_i(k)\, P_i^{\sigma_i,(\mu,Q^\sigma)}(k' \mid k),$$

where $P_i^{\sigma_i,(\mu,Q^\sigma)}$ denotes the objective transition kernel obtained from (S1) by setting $m_i = \mu$ and $q_i = Q^\sigma$. Thus, the laws of motion (F1)–(F4) hold.[7]

(ii) Subjective optimality. For each type i,

$$\sigma_i \in \arg\max_{\tilde{\sigma}_i \in \Sigma_i} \bar{U}_i\big(\tilde{\sigma}_i; \theta_i\big),$$

so that agents optimize within their subjective stationary environment.

(iii) On-path belief consistency. For each type i,

$$m_i(j) = \mu(j) \qquad \text{for all } j \in G \text{ with } \mu(j) > 0.$$

Condition (iii) forces meeting beliefs to be correct. It is stated only on the support of μ because goods outside the support are never observed, but m_i and μ are both probabilities that sum to one, so agreeing on the support pins them down everywhere with $m_i = \mu$ on all of G. When a reported SCE policy has more than one recurrent class, the reported distribution μ selects the recurrent class generated by the stated construction. The stationarity and belief-consistency conditions are then read on the support of that selected distribution. The complete-information best-response gain reported below is a separate diagnostic against the full-information benchmark.

[6] Under the unichain assumption, the Cesàro time average of the type-i marginal converges to μ_i from any initial distribution; if the chain is also aperiodic, marginals also converge to the stationary distribution. This is the natural learning-theoretic interpretation, since the agent's beliefs in condition (iii) are disciplined by the long-run distribution of her observations.

[7] These laws of motion describe how the cross-sectional distribution of holdings evolves over time under the policy σ. Equations (F1)–(F3) take the period-t distribution μ and produce the period-$(t+1)$ distribution μ' by accounting for trade flows and within-period consumption-and-production. Equation (F4) is the stationarity requirement that this period-to-period map fix μ. They are the distributional counterpart of the agent-level Bellman equations (B3)–(B4): the laws of motion describe the time path of the population, while the Bellman equations determine each agent's value function given that path.

Moreover, for every pre-trade decision state (i, k, j) at which the offer action $\lambda = 1$ is chosen with positive objective probability,

$$\mu_i(k)\,\mu(j)\,\sigma_i^e(1 \mid k, j) > 0 \quad \Longrightarrow \quad q_i(k, j) = Q^\sigma(k, j).$$

When

$$\mu_i(k)\,\mu(j)\,\sigma_i^e(1 \mid k, j) = 0,$$

observed data impose no restrictions on $q_i(k, j)$.

Stationary Nash equilibrium is the special case of self-confirming equilibrium in which all agents' subjective environments coincide with the objective one at every pre-trade state. If $(\{\mu_i\}, \{\alpha_i\}, \{\beta_i\}, Q)$ is a stationary Nash equilibrium and we set $m_i = \mu$ and $q_i(k, j) = Q(k, j)$ at *every* (i, k, j), then the resulting (μ, σ, θ) satisfies conditions (i)–(iii) of Definition 4.1. Conversely, for any restricted-information stationary self-confirming equilibrium in which $m_i = \mu$ and $q_i(k, j) = Q^\sigma(k, j)$ hold at *all* (i, k, j), not only on the support of trade, the policy-distribution pair (μ, σ) is a stationary Nash equilibrium. This mirrors the logic of Fudenberg and Kreps [1995]: self-confirming equilibrium requires optimality given subjective beliefs together with correct beliefs only along the equilibrium path.

We now specify the finite exchange economies used in the computations.

5 Economy-by-Economy Parameterizations

In this section we parameterize the five economies studied computationally in Section 7. Economies A1, A2, B, and C are parameterizations of the Kiyotaki and Wright [1989] exchange environments. Economy D is the five-good extension introduced in [Marimon et al., 1990].

In these economies, outcomes converge either to a *fundamental* equilibrium or to a *speculative* equilibrium, as discussed by Kiyotaki and Wright [1989]. In a fundamental equilibrium, agents accept a non-consumption good only when its acceptance is justified by storage costs and consumption values, i.e., by the prospect of a quicker or cheaper route to one's own consumption good. Depending on the economy, this

can support a single commodity-money convention (Economy A1), a two-good circulation pattern (Economy B), or a fiat-money convention in which a zero-utility object circulates because of correct beliefs that others will accept it (Economy C). A *speculative* equilibrium is one in which agents additionally accept a higher-storage-cost good they do not wish to consume, solely because they expect another type to accept it in turn.

Economy A1. Economy A1 has three types with equal shares $\nu_i = \frac{1}{3}$, production map

$$f(1) = 2, \qquad f(2) = 3, \qquad f(3) = 1,$$

storage costs $(s_1, s_2, s_3) = (0.1,\ 1,\ 20)$, and utilities $\bar{u}_i = 100$. These parameters lead to the *fundamental* equilibrium in which the lowest-storage-cost good serves as commodity money.

Economy A2. Economy A2 is identical to A1 except that utilities are raised to $\bar{u}_i = 500$ for all i. Kiyotaki and Wright [1989] show that type 1 plays the fundamental strategy if and only if

$$s_3 - s_2 \;>\; \left(\pi_3^h(1) - \pi_2^h(1)\right) \tfrac{1}{3}\, \bar{u}_1,$$

where $\pi_3^h(1)$ and $\pi_2^h(1)$ are the stationary probabilities that types 3 and 2 hold good 1 in the conjectured equilibrium. Evaluated at the fundamental allocation, $\pi_3^h(1) = 1$ and $\pi_2^h(1) = \frac{1}{2}$, so the right-hand side equals $\frac{1}{6}\bar{u}_1$. With $s_3 - s_2 = 19$, the inequality is satisfied at $\bar{u}_1 = 100$ ($\frac{1}{6} \cdot 100 \approx 16.7 < 19$) and is violated at $\bar{u}_1 = 500$ ($\frac{1}{6} \cdot 500 \approx 83.3 > 19$). Hence, in Economy A2, the *speculative* branch is sustainable. The results below distinguish that branch from the fundamental holding allocation.

Economy B. For Economy B, types and shares are the same as in A1, but the production map changes to

$$f(1) = 3, \qquad f(2) = 1, \qquad f(3) = 2,$$

with $(s_1, s_2, s_3) = (1,\ 4,\ 9)$ and $\bar{u}_i = 100$. This economy has two stationary equilibria: a *fundamental* one in which goods 1 and 2 serve as media of exchange, and a *speculative* one in which goods 2 and 3 do.

Economy C. Economy C augments the A1 production cycle with a fiat-money "good", good 0, keeping the equal type shares $\nu_i = \frac{1}{3}$:

$$G = \{0, 1, 2, 3\}, \qquad f(1) = 2, \quad f(2) = 3, \quad f(3) = 1.$$

Following Kiyotaki and Wright [1989], good 0 yields no consumption utility, has zero storage cost, and cannot be consumed; that is, $\Gamma_i(0) = \{0\}$ for every type i. The storage costs are $(s_0, s_1, s_2, s_3) = (0,\ 9,\ 14,\ 29)$ and $\bar{u}_i = 100$. The aggregate fiat-money share is fixed at the exogenous level $M_0 = 0.32$, imposed as $\sum_{i \in I} \nu_i \, \mu_i(0) = M_0$. The SNE benchmark is the *fundamental* fiat-money equilibrium in which good 0 is the unique fiat medium of exchange, even though commodity goods may also change hands on the equilibrium path.

Economy D. Economy D is the five-good model of Marimon et al. [1990]. It has five consumer types with $\nu_i = \frac{1}{5}$, production map

$$f(1) = 3, \quad f(2) = 4, \quad f(3) = 5, \quad f(4) = 1, \quad f(5) = 2,$$

storage costs $(s_1, \ldots, s_5) = (1,\ 4,\ 9,\ 20,\ 30)$, and utilities $\bar{u}_i = 200$. In Economy D, every good is simultaneously a consumption good for one type and a production good for another, so double coincidences of wants can arise. Three forces shape trade: storage costs favor low-numbered goods, consumption value attaches to each type's own good, and speculation generates a further incentive as a good may be accepted solely because another type is expected to accept it later. Economy D poses the same speculative problem as the three-good cases, scaled to a larger network with more off-path and low-probability trade states. Because Economy D lacks a closed-form equilibrium characterization, Marimon et al. [1990] report only a classifier-derived trading pattern, described as nearly fundamental with some speculative moves.

In the next section, we construct three comparable outcomes from these primitives: the stationary Nash benchmark, a restricted-information SCE, and the tabular learning trajectory.

6 Computations

Each economy is studied under three procedures. The first procedure computes a stationary Nash equilibrium with complete information and correct beliefs: the resulting policy satisfies the Bellman equations (B3)–(B4), the holding distribution satisfies stationarity (F1)–(F4), and the conditional type distribution satisfies (B1). When multiple equilibria exist, the solution reached depends on the *prompting* that sets the initial policy. The second procedure reports restricted-information SCE results: agents optimize against subjective environments $\theta_i = (m_i, q_i)$, with beliefs constrained to be correct only on state-action pairs that arise under the candidate policy. The formal fixed-point and residual conditions are stated in Section 6.2.

The third procedure is a finite-population tabular learning algorithm. A population of $\mathcal{N}$ agents proceeds through a sequence of learning iterations, and the algorithm maintains three sets of shared tables. The empirical model $\hat{M}^n$ records meeting frequencies, transition kernels at trade and consumption stages, and average rewards, all as cumulative count ratios; $\hat{M}^n$ denotes the table at the start of iteration n, to which new records are appended online as the iteration proceeds. Policy-prior tables $\alpha_i^n(k, j)$ and $\beta_i^n(h)$ store trade and consumption priors. A terminal-value table $v_i^n(h)$ holds continuation values used at the end of each planning horizon. Each learning iteration consists of R random-matching rounds. Before every trade or consumption decision, the acting agent runs a PUCT search [Rosin, 2011] of fixed horizon against the current empirical model, with (α^n, β^n) as action priors and v^n as terminal values, then draws an action according to the resulting search counts. Because the state space is finite and small, the search enumerates every chance outcome under the empirical model exactly rather than drawing random rollouts as in standard Monte Carlo tree search. Each realized transition and reward is appended to the empirical model immediately, so searches at roots not yet visited within the same iteration see the updated table; the average search-count policies update α and β, and the maximum root action values update v, at the end of the iteration. The detailed algorithm is described in Section 6.3.

We report two versions of this learning procedure. In the *fully empirical* case agents estimate everything from data, including the deterministic one-period rules

such as the consumption reward, the production of $f(i)$ after consumption, and the holding that results when a trade is refused. This is a demanding benchmark for the algorithm, because finite-sample noise in objects that are in principle deterministic can affect the learning of the distributional objects. Yet, it also represents a more general learning problem that does not rely on the agent's prior knowledge of the primitives.

The *known-environment* case, defined formally in Section 6.4, treats those deterministic rules as primitives that the agent already knows, leaving her to learn only the endogenous market objects, namely which goods her partners hold, which of her offers are reciprocated, and the continuation values and policies these imply. This case also evaluates the agent's own future decisions greedily inside the planning horizon and normalizes search-derived value targets as average-reward bias values.

6.1 Stationary Nash computation

The relative value iteration (RVI) solver iterates over stationary policies $\sigma = (\alpha, \beta)$. At iteration m, it computes the invariant population law by iterating the law-of-motion map (F1)–(F4), yielding the objective stationary environment (μ^m, Q^m) from (B2). Holding this environment fixed, each type's average-reward control problem under correct beliefs is solved by RVI on (B3)–(B4), with the bias normalized at a reference good. The current policy is then updated toward the greedy best-response policy through a damped update. The iteration stops when both the stationary-distribution residual and the best-response gain fall below tolerance.

We initialize the exchange probabilities to $\alpha_i^0(k, j) = \frac{1}{2}$, while consumption is initialized at $\beta_i^0(i) = 1$ and $\beta_i^0(h) = 0$ for $h \neq i$. To reach the speculative equilibria in Economies A2 and B, the solver instead starts from a pure speculative initial policy that includes the key speculative links. In A2 these include type 1's offers of good 2 for good 3 and of good 3 for good 1; in B they include type 2's offers of good 1 for good 3 and of good 3 for good 2. This initial policy is a full pure policy pattern, not just a change at those two states. Economy C starts from a money-acceptance initial policy, setting $\alpha_i^0(k, 0) = 1$ and $\alpha_i^0(k, i) = 1$ for all types i and holdings k, and zero elsewhere. This is an example of prompting that guides the solver to a

Algorithm 1 Relative value iteration for stationary Nash equilibrium

Require: Initial policy $\sigma^0 = (\alpha^0, \beta^0)$, damping factor $\rho_m \in (0, 1]$, tolerances $\varepsilon_{\text{dist}}, \varepsilon_{\text{BR}} > 0$

for $m = 0, 1, \ldots$ **do**
  Iterate (F1)–(F4) under σ^m to obtain (μ^m, Q^m)
  for all types i **do**
    Run RVI in (μ^m, Q^m) to obtain $(g_i^{\star,m}, V_i^m, W_i^m)$
    Choose $\bar{\sigma}_i^m = (\bar{\sigma}_i^{e,m}, \bar{\sigma}_i^{c,m})$ satisfying, for all $h \in G$ and $k, j \in G$,

$$\bar{\sigma}_i^{c,m}(\cdot \mid h) \in \Delta\left(\operatorname*{argmax}_{\gamma \in \Gamma_i(h)} \{r_i(h,\gamma) + V_i^m(c_i(h,\gamma))\}\right),$$

$$\bar{\sigma}_i^{e,m}(\cdot \mid k, j) \in \Delta\left(\operatorname*{argmax}_{\lambda \in \Lambda_i(k,j)} \lambda\, Q^m(k,j)\left[W_i^m(j) - W_i^m(k)\right]\right).$$

    Evaluate σ_i^m in (μ^m, Q^m) to obtain $g_i^{\sigma,m}$ and set $e_i^m \leftarrow g_i^{\star,m} - g_i^{\sigma,m}$
  end for
  Set $e^m \leftarrow \max_i e_i^m$ and $d^m \leftarrow \|F_{\sigma^m}(\mu^m) - \mu^m\|_\infty$
  if $e^m \leqslant \varepsilon_{\text{BR}}$ and $d^m \leqslant \varepsilon_{\text{dist}}$ **then**
    return (σ^m, μ^m, Q^m)
  end if
  Update $\sigma^{m+1} \leftarrow (1 - \rho_m)\sigma^m + \rho_m \bar{\sigma}^m$
end for

particular equilibrium when multiple equilibria exist. The detailed algorithm is listed in Algorithm 1.

6.2 Restricted-information SCE computation

The SCE benchmark uses the same economic environment but restricts only on-path beliefs. Each candidate policy is evaluated against the subjective environment $\theta_i = (m_i, q_i)$ from Section 4.1: m_i must match the stationary distribution of partner goods that type i observes, and $q_i(k, j)$ must match the true reciprocal probability only at states where type i holds k, meets j, and offers trade with positive probability. Beliefs at unoffered states are unrestricted by Definition 4.1.

Algorithm 2 states the Bellman/RVI fixed-point calculation associated with this definition. At iteration m, let Q^m denote the objective reciprocal-trade probability implied by (σ^m, μ^m) via (B2), and define the on-path set

$$O^m = \left\{(i, k, j) : \mu_i^m(k)\, \mu^m(j)\, \sigma_i^{e,m}(1 \mid k, j) > 0\right\}.$$

Given θ_i, the algorithm computes the type's subjective best response via the Bellman equation in Section 4.1, recomputes the objective stationary distribution, sets $m_i \leftarrow \mu$, and sets $q_i(k, j) \leftarrow Q^\sigma(k, j)$ at every on-path triple $(i, k, j) \in O^m$, leaving q_i unchanged at off-path triples. Convergence is verified by inspecting three residuals, all evaluated at the current iterate $(\sigma^m, \mu^m, \theta^m)$. Let $g_i^{\theta,m}$ be type i's optimal subjective average reward under θ_i^m, and let $g_i^{\sigma,\theta,m}$ be the subjective average reward from σ_i^m under θ_i^m. The subjective optimality residual is

$$s^m = \max_i \left\{g_i^{\theta,m} - g_i^{\sigma,\theta,m}\right\}.$$

The on-path belief residual and stationarity residual are

$$b^m = \max_{(i,k,j)\in O^m} \left|q_i^m(k, j) - Q^m(k, j)\right|, \qquad d^m = \left\|F_{\sigma^m}(\mu^m) - \mu^m\right\|_\infty,$$

with $b^m = 0$ if O^m is empty. The solver stops when $\max\{s^m, b^m\} \leqslant \varepsilon_{\text{SCE}}$ and $d^m \leqslant \varepsilon_{\text{dist}}$, returning the current iterate $(\sigma^m, \mu^m, \theta^m)$.

We report non-Nash SCEs computed from the fixed-point calculation with skeptical off-path beliefs — misdirectional prompts that close off paths to improvement

through trade. In Section 7, the reported object satisfies the subjective-optimality, on-path-belief, and stationarity residuals above, and its complete-information best-response gain is reported separately to show that it is not Nash.

6.3 Tabular learning search

A finite population learns a model of the meeting environment from observed play and uses it to plan each trade and consumption decision. The estimates accumulate counts over the entire history of play, so the planning model is a history-weighted average of past environments.

Let $\mathcal{N}$ denote the population size, distinct from the number of types $N = |I|$. At iteration n, the population state is $\mathcal{P}^n_{\mathcal{N}} = \{(i_a, h^n_a)\}_{a=1}^{\mathcal{N}}$, with each agent's type $i_a \in I$ and current holding $h^n_a \in G$. Bilateral meetings advance the population and add experience to the empirical model $\hat{M}^n$ that is shared among agents. This model holds empirical frequencies, transition laws, and reward averages defined in Section 6.3.1. Before each trade or consumption decision, an agent runs finite-horizon PUCT using $\hat{M}^n$, the policy priors $\alpha^n = (\alpha^n_i(k, j))$ and $\beta^n = (\beta^n_i(h))$, and the terminal-value table $\upsilon^n = (\upsilon^n_i(h))$. The terminal table enters only when the planning horizon is reached. It is revised at the end of each iteration from the search values themselves, as described in Section 6.3.3.

Given $(\hat{M}^n, \alpha^n, \beta^n, \upsilon^n)$, PUCT computes the pre-trade and post-trade search-value functions $\hat{V}^n_i(h, d)$ and $\hat{W}^n_i(h, d)$ from the root-planning recursions (L2) and (L3). Each call returns a root policy on $\{0, 1\}$ obtained by normalizing the PUCT search counts as in (P2), together with the maximum root action value associated with the completed search. The root policies and maximum root action values accumulated within iteration n provide the update targets for $(\alpha^{n+1}, \beta^{n+1}, \upsilon^{n+1})$, defined in Section 6.3.3. The following subsections develop the algorithm in detail.

Before turning to the details, it helps to situate the algorithm relative to the problem MuZero was designed to solve. MuZero targets environments in which the agent's observations are rich enough to support a generative model of the underlying dynamics, such as a board configuration in Go or a stack of pixel frames in Atari, so that planning, executed as rollouts inside the learned model, produces faithful coun-

Algorithm 2 Restricted-information SCE fixed-point calculation

Require: Initial policy and beliefs (σ^0, θ^0) with $\theta_i^0 = (m_i^0, q_i^0)$, initial distribution μ^0, damping factor $\rho_m \in (0, 1]$, tolerances $\varepsilon_{\mathrm{SCE}}, \varepsilon_{\mathrm{dist}} > 0$

for $m = 0, 1, \ldots$ **do**
  for all types i **do**
    Find $(g_i^{\theta,m}, V_i^{\theta,m}, W_i^{\theta,m})$ with $V_i^{\theta,m}(k_i^{\mathrm{ref}}) = 0$ satisfying, for all $h, k \in G$,

$$W_i^{\theta,m}(h) = \max_{\gamma \in \Gamma_i(h)} \{ r_i(h,\gamma) + V_i^{\theta,m}(c_i(h,\gamma)) \},$$

$$g_i^{\theta,m} + V_i^{\theta,m}(k) = \sum_{\ell \in G} m_i^m(\ell) \max_{\lambda \in \Lambda_i(k,\ell)} \sum_{h' \in G} T_{q_i^m}(h' \mid k, \ell, \lambda)\, W_i^{\theta,m}(h').$$

    Choose $\bar{\sigma}_i^m = (\bar{\sigma}_i^{e,m}, \bar{\sigma}_i^{c,m})$ satisfying, for all $h \in G$ and $k, j \in G$,

$$\bar{\sigma}_i^{c,m}(\cdot \mid h) \in \Delta\left(\operatorname*{argmax}_{\gamma \in \Gamma_i(h)} \{ r_i(h,\gamma) + V_i^{\theta,m}(c_i(h,\gamma)) \} \right),$$

$$\bar{\sigma}_i^{e,m}(\cdot \mid k, j) \in \Delta\left(\operatorname*{argmax}_{\lambda \in \Lambda_i(k,j)} \sum_{h \in G} T_{q_i^m}(h \mid k, j, \lambda)\, W_i^{\theta,m}(h) \right).$$

    Evaluate σ_i^m under θ_i^m to obtain $g_i^{\sigma,\theta,m}$ and set $s_i^m \leftarrow g_i^{\theta,m} - g_i^{\sigma,\theta,m}$
  end for
  Compute Q^m from (σ^m, μ^m) via (B2)
  Compute the subjective gap $s^m \leftarrow \max_i s_i^m$, the on-path belief error

$$b^m \leftarrow \max_{(i,k,j) \in O^m} \left| q_i^m(k,j) - Q^m(k,j) \right|, \qquad O^m := \left\{ (i,k,j) : \mu_i^m(k)\, \mu^m(j)\, \sigma_i^{e,m}(1 \mid k, j) > 0 \right\},$$

  with the convention $b^m \leftarrow 0$ if $O^m = \varnothing$, and the stationarity residual $d^m \leftarrow \| F_{\sigma^m}(\mu^m) - \mu^m \|_\infty$
  if $\max\{s^m, b^m\} \leqslant \varepsilon_{\mathrm{SCE}}$ and $d^m \leqslant \varepsilon_{\mathrm{dist}}$ **then**
    return $(\sigma^m, \mu^m, \theta^m)$
  end if
  Update $\sigma^{m+1} \leftarrow (1 - \rho_m)\sigma^m + \rho_m \bar{\sigma}^m$
  Iterate (F1)–(F4) under σ^{m+1} to obtain μ^{m+1} and compute Q^{m+1} from (B2)
  for all types i and goods j **do**
    Set $m_i^{m+1}(j) \leftarrow \mu^{m+1}(j)$
  end for
  for all (i, k, j) **do**
    if $\mu_i^{m+1}(k)\, \mu^{m+1}(j)\, \sigma_i^{e,m+1}(1 \mid k, j) > 0$ **then**
      Set $q_i^{m+1}(k,j) \leftarrow Q^{m+1}(k,j)$
    else
      Keep $q_i^{m+1}(k,j) \leftarrow q_i^m(k,j)$
    end if
  end for
end for

terfactuals of the true environment. In our bilateral-matching economy, an agent observes only her own type and holding and her partner's holding at the moment of a meeting. She does not observe the cross-sectional holding distribution, other agents' types or policies, or any history of meetings in which she did not participate. What she learns, namely partner-good frequencies and reciprocal acceptance rates, is not the law of motion of the underlying economy. It is an empirical summary of an equilibrium object generated by the joint play of everyone else. Holding others' behavior fixed, her empirical model faithfully describes what she will meet on the equilibrium path, but it says nothing about how others' behavior would change in response to her own.

By contrast, the computational burden in our setting is smaller. The state and action spaces of the bilateral-matching economy are small enough for exact tabular representation. A meeting state is the finite triple (i, k, j) and the post-trade consumption state is the pair (i, h). Where MuZero needs neural networks to compress high-dimensional observations into a usable state, ours is already small enough to enumerate. We therefore keep MuZero's PUCT-based policy improvement but replace its neural approximators and Monte Carlo tree search with empirical counts, tabular transition laws, and exact enumeration of chance outcomes.

6.3.1 State, policy, and empirical model

For a realized meeting, let i be the focal agent's type, k her current holding, and j the partner's holding. The trade and consumption decisions are represented as binary actions,

$$\lambda = 0 \quad \text{refuse}, \qquad \lambda = 1 \quad \text{offer},$$

$$\gamma = 0 \quad \text{keep}, \qquad \gamma = 1 \quad \text{consume},$$

with infeasible consumption actions excluded as in Section 2.

At iteration n, the algorithm maintains population-level policy prior tables with one entry for each trade and consumption root,

$$\alpha_i^n(k, j) \in [0, 1], \qquad \beta_i^n(h) \in [0, 1],$$

where $\alpha_i^n(k,j)$ is the prior probability of offering at trade root (i,k,j) and $\beta_i^n(h)$ is the prior probability of consuming at consumption root (i,h). All agents of the same type and state share the same prior. These priors enter the PUCT search as inputs rather than as final policies. The learning initialization is separate from the RVI initialization in Section 6.1. In the learning algorithm, we set $\alpha_i^0(k,j) = 1/2$ at every trade root, $\beta_i^0(h) = 1/2$ at feasible consumption roots and zero at infeasible ones, and $v_i^0(h) = 0$. Each agent initially holds her type's production good. In Economy C, the fixed fiat-money stock is distributed evenly across types at initialization, with the remaining mass placed on each type's production good.

The algorithm also maintains a single shared tabular model $\hat{M}^n$, estimated from the collective experience of the entire population. For each type i, the partner-good distribution is

$$\hat{m}_i^n(j) = \frac{C_i^n(j)}{\sum_{\ell \in G} C_i^n(\ell)},$$

where $C_i^n(j)$ counts the meetings in which a type-i agent faced a partner holding good j, summed across all rounds and iterations through iteration n and the initial count value. A second empirical object summarizes reciprocal acceptance. Let

$$\hat{q}^n(k,j) = \frac{A_{\text{acc}}^n(k,j)}{O^n(k,j)},$$

where $O^n(k,j)$ counts the offers made by any agent holding good k to a partner holding good j and $A_{\text{acc}}^n(k,j)$ counts those offers that were reciprocated. The counts are pooled across the offerer's type. Conditional on the partner holding good j, the relevant object is the average acceptance rate among the types who hold good j and face an offer of good k. The role of $\hat{q}^n$ in planning depends on the learning variant, as discussed in Section 6.4.

The model also records empirical transition laws at the trade and consumption stages. Let $N_i^{T,n}(k,j,\lambda,h')$ be the sum of initial count values and observed counts for trade transitions from root (i,k,j) under action λ to next holding h', and let $N_i^{C,n}(h,\gamma,h')$ be the analogous count for consumption transitions. Then the learned transition probabilities are

$$\hat{P}_i^{T,n}(h' \mid k,j,\lambda) = \frac{N_i^{T,n}(k,j,\lambda,h')}{\sum_{\ell \in G} N_i^{T,n}(k,j,\lambda,\ell)},$$

and

$$\hat{P}_i^{C,n}(h' \mid h, \gamma) = \frac{N_i^{C,n}(h, \gamma, h')}{\sum_{\ell \in G} N_i^{C,n}(h, \gamma, \ell)}.$$

The empirical one-period rewards are sample averages. Let $B_i^{T,n}(k, j, \lambda)$ be the number of trade-stage records, including the initial count value, used to average rewards at root (i, k, j) under action λ. Let $B_i^{C,n}(h, \gamma)$ be the analogous number of consumption-stage records used to average rewards. Let $S_i^{T,n}(k, j, \lambda)$ and $S_i^{C,n}(h, \gamma)$ be the corresponding cumulative reward sums. Then the average rewards are

$$\hat{r}_i^{T,n}(k, j, \lambda) = \frac{S_i^{T,n}(k, j, \lambda)}{B_i^{T,n}(k, j, \lambda)}, \qquad \hat{r}_i^{C,n}(h, \gamma) = \frac{S_i^{C,n}(h, \gamma)}{B_i^{C,n}(h, \gamma)}. \tag{L1}$$

These tables are updated directly from realized self-play records. A trade record specifies the type, pre-trade holding, partner holding, trade action, post-trade holding, and realized trade reward, $(i, k, j, \lambda, h^+, r^T)$. It updates

$$\begin{aligned} N_i^{T,n}(k, j, \lambda, h^+) &\leftarrow N_i^{T,n}(k, j, \lambda, h^+) + 1, \\ B_i^{T,n}(k, j, \lambda) &\leftarrow B_i^{T,n}(k, j, \lambda) + 1, \\ S_i^{T,n}(k, j, \lambda) &\leftarrow S_i^{T,n}(k, j, \lambda) + r^T. \end{aligned}$$

A consumption record specifies the type, holding, consumption action, next holding, and realized consumption reward, (i, h, γ, h^+, r^C). It updates

$$\begin{aligned} N_i^{C,n}(h, \gamma, h^+) &\leftarrow N_i^{C,n}(h, \gamma, h^+) + 1, \\ B_i^{C,n}(h, \gamma) &\leftarrow B_i^{C,n}(h, \gamma) + 1, \\ S_i^{C,n}(h, \gamma) &\leftarrow S_i^{C,n}(h, \gamma) + r^C. \end{aligned}$$

Agents receive no reward at the trade stage, so $r^T = 0$ and $\hat{r}_i^{T,n}$ is always zero. The consumption reward is $r^C = r_i(h, \gamma)$, defined in (R).

Together, these objects define the empirical model

$$\hat{M}^n = \left(\{\hat{m}_i^n\}_{i \in I},\ \hat{q}^n,\ \{\hat{P}_i^{T,n}, \hat{P}_i^{C,n}, \hat{r}_i^{T,n}, \hat{r}_i^{C,n}\}_{i \in I} \right),$$

which is the probability-and-reward table derived from the cumulative experience of the population through iteration n:

$$\left(C_i^n,\ O^n,\ A_{\text{acc}}^n,\ N_i^{T,n},\ N_i^{C,n},\ B_i^{T,n},\ B_i^{C,n},\ S_i^{T,n},\ S_i^{C,n}\right)_{i \in I}.$$

The count and averaging-denominator arrays $C_i^n, O^n, N_i^{T,n}, N_i^{C,n}, B_i^{T,n}$, and $B_i^{C,n}$ start from the positive initial value κ_0. The accepted-offer counts A_{acc}^n are initialized at $\kappa_0/2$, so that the prior reciprocal-acceptance probability $\hat{q}^0(k,j) = A_{\text{acc}}^0(k,j)/O^0(k,j)$ equals $1/2$ at every (k,j). The reward-sum arrays $S_i^{T,n}$ and $S_i^{C,n}$ are initialized consistently with the initial reward averages. Every component of $\hat{M}^n$ is therefore defined before its corresponding cell has been observed. As agents interact, the realized observations move these ratios toward the empirical frequencies and sample rewards.

6.3.2 Root planning

A *decision node* is the state at which the agent chooses a binary action. At the trade stage this state is the triple (i,k,j) formed by the agent's type, her current holding, and the partner's holding. At the consumption stage it is the pair (i,h) formed by the agent's type and current holding. The planning step uses the empirical model $\hat{M}^n$ defined in Section 6.3.1. The learner knows the *tree skeleton*, i.e., the state variables, the feasible actions (offer/refuse, consume/keep), and the list of possible next holdings, but the transition probabilities and rewards on each branch are looked up from $\hat{M}^n$. In the fully empirical case these are the sample averages $(\hat{P}_i^{T,n}, \hat{P}_i^{C,n}, \hat{r}_i^{T,n}, \hat{r}_i^{C,n})$.

At each decision node, the agent runs PUCT before selecting an action. PUCT combines two information sources: the action prior from α or β and the action values computed by projecting the empirical model H steps forward. Rather than taking the greedy action, PUCT runs a fixed number of simulations. Each simulation selects an action, updates its running value estimate, and increments a visit count. The normalized distribution of these counts is the agent's policy at that node.

For a trade root (i,k,j), let

$$\widehat{\Psi}_i^{T,n}(k,j,\lambda;d) = \hat{r}_i^{T,n}(k,j,\lambda) + \sum_{h' \in G} \hat{P}_i^{T,n}(h' \mid k,j,\lambda)\, \hat{W}_i^n(h',d) \tag{L2}$$

denote the learned value of trade action $\lambda \in \{0,1\}$ with d remaining search steps. Equation (L2) and equation (L3) below are a mutually recursive pair: $\widehat{\Psi}^{T,n}$ uses the continuation $\widehat{\Psi}^{C,n}$ (via $\hat{W}_i^n$), and $\widehat{\Psi}^{C,n}$ uses $\widehat{\Psi}^{T,n}$ (via $\hat{V}_i^n$); both recursions terminate at the scalar table $v_i^n(h)$ when d reaches zero. The continuation value $\hat{W}_i^n(h,d)$ is the expected value accruing to a type-i agent who exits the trade stage holding h with d

search steps remaining, equivalently the value immediately before the consumption decision,

$$\hat{W}_i^n(h,d) = \big(1-\beta_i^n(h)\big)\,\widehat{\Psi}_i^{C,n}(h,0;d) + \beta_i^n(h)\,\widehat{\Psi}_i^{C,n}(h,1;d),$$

where the current consumption prior $\beta_i^n(h)$ weights the two consumption-action values. The pre-trade continuation value $\hat{V}_i^n(h,d)$ then averages the trade-root values over the learned partner-good distribution,

$$\hat{V}_i^n(h,d) = \sum_{j\in G} \hat{m}_i^n(j)\Big[\big(1-\alpha_i^n(h,j)\big)\,\widehat{\Psi}_i^{T,n}(h,j,0;d) + \alpha_i^n(h,j)\,\widehat{\Psi}_i^{T,n}(h,j,1;d)\Big].$$

At search depth zero, both recursions terminate at the same scalar table,

$$\hat{V}_i^n(h,0) = v_i^n(h), \qquad \hat{W}_i^n(h,0) = v_i^n(h).$$

The Bellman equations in Section 3 keep the pre-trade bias V_i and the post-trade value W_i separate. To simplify the algorithm, the learner collapses them into a single continuation-value table v_i^n.

For a consumption root (i,h), the learned action value for consumption action $\gamma \in \Gamma_i(h)$ is

$$\widehat{\Psi}_i^{C,n}(h,\gamma;d) = \hat{r}_i^{C,n}(h,\gamma) + \sum_{h'\in G} \hat{P}_i^{C,n}(h' \mid h,\gamma)\,\hat{V}_i^n(h',d-1). \tag{L3}$$

Infeasible consumption actions are masked before normalization and selection. When $\Gamma_i(h) = \{0\}$, the consumption policy is fixed at "keep." Figure 1 summarizes the forward-search structure that PUCT navigates at a trade root.

At a given root, write p_x for the prior probability of root action $x \in \{0,1\}$ and $\bar{L}_x$ for the running value estimate. The estimate starts at the model-implied action value,

$$\bar{L}_x \leftarrow \begin{cases} \widehat{\Psi}_i^{T,n}(k,j,x;H), & \text{trade root,} \\ \widehat{\Psi}_i^{C,n}(h,x;H), & \text{consumption root,} \end{cases}$$

where H is the training horizon. Each action also carries a search count N_x initialized to zero. Before the exploration bonus is added, the two running value estimates are

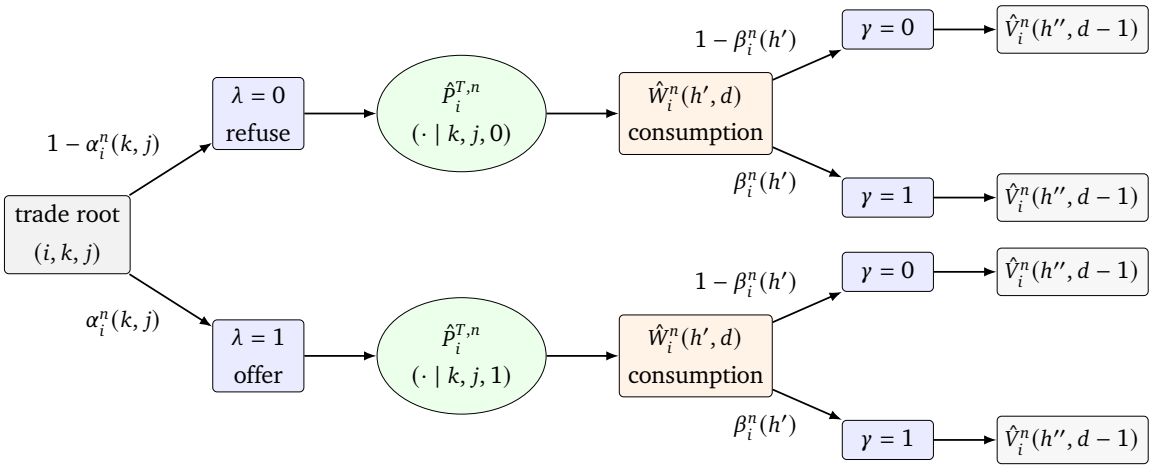

Figure 1: Root-level forward search used by PUCT at a trade root. The root branches over the binary trade action, applies the learned trade transition, then evaluates the consumption continuation under the current consumption prior β_i^n, and finally evaluates the next pre-trade continuation value. Here h' denotes a post-trade holding reached under $\hat{P}_i^{T,n}(\cdot \mid k,j,\lambda)$, and h'' denotes the next beginning-of-period holding after the consumption transition.

normalized to a unit-free score,

$$S_x(\bar{L}) = \begin{cases} \dfrac{\bar{L}_x - \min_y \bar{L}_y}{\max_y \bar{L}_y - \min_y \bar{L}_y}, & \text{if } \max_y \bar{L}_y > \min_y \bar{L}_y, \\ \dfrac{1}{2}, & \text{otherwise.} \end{cases}$$

Let $N_{\text{tot}} = N_0 + N_1$. In simulation s, the planner selects

$$x_s \in \operatorname*{argmax}_{x\in\{0,1\}} \left\{ S_x(\bar{L}) + c_{\text{puct}}\, p_x \frac{\sqrt{N_{\text{tot}}+1}}{1+N_x} \right\}, \tag{P1}$$

and updates the selected action's statistics by

$$N_{x_s} \leftarrow N_{x_s} + 1, \qquad \bar{L}_{x_s} \leftarrow \bar{L}_{x_s} + \frac{\ell_{x_s} - \bar{L}_{x_s}}{N_{x_s}},$$

where ℓ_{x_s} is the same model-implied action value $\widehat{\Psi}$ used to initialize $\bar{L}_x$, now evaluated at the selected action, namely $\widehat{\Psi}_i^{T,n}(k,j,x_s;H)$ at a trade root and $\widehat{\Psi}_i^{C,n}(h,x_s;H)$

at a consumption root. Within a single PUCT search $\hat{M}^n$ is held fixed and queried as a tabular look-up, so ℓ_{x_s} does not depend on the simulation index s: every visit to root action x returns the same value. The first visit sets $\bar{L}_x$ to that value, and because $\ell_{x_s} = \bar{L}_{x_s}$ from then on, the running-average update $\bar{L}_x \leftarrow \bar{L}_x + (\ell_{x_s} - \bar{L}_{x_s})/N_{x_s}$ leaves it unchanged. The simulation budget within a search is therefore spent on allocating visit counts N_x across actions, not on refining values. Across decisions and iterations, by contrast, $\hat{M}^n$ is updated from realized self-play observations as they arrive, so the model-implied values seen by later searches can change. Selection in (P1) is therefore a predictor-plus-UCB rule with fixed action values $\widehat{\Psi}$ and an exploration term that diminishes as the action is visited more often. Since the exploration term is proportional to the prior p_x, the action favored by the current policy prior (α_i^n at a trade node, β_i^n at a consumption node) receives more search visits and therefore more weight in the resulting policy. After the PUCT simulations, the root policy is the normalized search counts,

$$\pi(x) = \frac{N_x^{1/\tau}}{\sum_{y\in\{0,1\}} N_y^{1/\tau}}, \qquad x \in \{0,1\}, \tag{P2}$$

where τ controls concentration on the action with the largest search count. During learning, agents draw from π rather than always picking the most-visited action, and that randomization preserves exploration.

6.3.3 Self-play data and policy update

Let $\mathcal{D}_n$ denote the iteration-n dataset, i.e., the multiset of realized trade and consumption records collected during iteration n (each record has the form (i,k,j,λ,h^+,r^T) or (i,h,γ,h^+,r^C) introduced in Section 6.3.1); $\mathcal{D}_n$ is the input to the empirical-model update operator used in Algorithms 3–4. Let $\zeta_n \in (0,1)$ be the policy/value learning rate used at iteration n; we let ζ_n decay geometrically to a positive lower bound, as discussed below.

Each learning iteration consists of R random-matching rounds. In each round agents are randomly paired, plan from their type and current holding, and independently draw trade actions. An exchange is completed only when both agents

offer and the two goods differ. Same-good matches ($k = j$) therefore leave holdings unchanged and do not update the acceptance-count tables $A^n_{\text{acc}}(k, j)$ and $O^n(k, j)$; the trade-stage transition with $h^+ = k$ is still appended to $\mathcal{D}_n$ so that $\hat{P}_i^{T,n}(\cdot \mid k, k, \cdot)$ remains well-defined. For different-good matches, every offer updates $O^n(k, j)$ and every reciprocated offer updates $A^n_{\text{acc}}(k, j)$.

After the trade stage in each round, every agent runs consumption-root PUCT, samples a consumption action, and records the realized reward and next holding in the shared consumption tables. The empirical model is therefore estimated from the same population process that drives the policy updates. In the reported runs, $\mathcal{N} = 1800$ and $R = 35$, so one self-play iteration contains 35×900 pair meetings and 35×1800 consumption decisions.

For each trade root (i, k, j) visited at iteration n, let $\bar{\pi}_i^{T,n}(\cdot \mid k, j)$ be the average across all PUCT calls at that root of the PUCT policies computed from normalized search counts. At unvisited roots, set $\bar{\pi}_i^{T,n}(1 \mid k, j) = \alpha_i^n(k, j)$. Define $\bar{\pi}_i^{C,n}(\cdot \mid h)$ analogously at consumption roots, with $\bar{\pi}_i^{C,n}(1 \mid h) = \beta_i^n(h)$ at unvisited feasible roots and $\bar{\pi}_i^{C,n}(\cdot \mid h) = (1, 0)$ when consumption is infeasible. The policy priors then update as[8]

$$\alpha_i^{n+1}(k, j) = \text{clip}_\varepsilon\left[(1 - \zeta_n)\alpha_i^n(k, j) + \zeta_n \bar{\pi}_i^{T,n}(1 \mid k, j)\right],$$

$$\beta_i^{n+1}(h) = \text{clip}_\varepsilon\left[(1 - \zeta_n)\beta_i^n(h) + \zeta_n \bar{\pi}_i^{C,n}(1 \mid h)\right].$$

The terminal-value table is updated by the same convex combination. Let $\overline{V}_i^n(h)$ be the average over the PUCT calls in iteration n whose root has type i and own holding h (the trade roots (i, h, j) and the consumption root (i, h)) of the maximum root action value returned by PUCT, or $v_i^n(h)$ if no such root is visited. Then

$$v_i^{n+1}(h) = (1 - \zeta_n)\, v_i^n(h) + \zeta_n\, \overline{V}_i^n(h),$$

so unvisited value entries also remain unchanged.

[8] The clipping operator is $\text{clip}_\varepsilon[x] = \min(1 - \varepsilon, \max(\varepsilon, x))$, the projection onto $[\varepsilon, 1 - \varepsilon]$ for feasible binary actions; $\text{clip}_\varepsilon[x] = 0$ for infeasible consumption actions (i.e., when $1 \notin \Gamma_i(h)$ in fiat-money economies).

6.3.4 Stopping rule and diagnostics

Let $\mathcal{U}_n^T$ denote the trade roots (i,k,j) at which at least one PUCT call occurred in iteration n, and $\mathcal{U}_n^C$ the nontrivial consumption roots (i,h) at which at least one PUCT call occurred. The mean policy update used for stopping is

$$\bar{\Delta}_n = \frac{\sum_{(i,k,j)\in\mathcal{U}_n^T} |\alpha_i^{n+1}(k,j) - \alpha_i^n(k,j)| + \sum_{(i,h)\in\mathcal{U}_n^C} |\beta_i^{n+1}(h) - \beta_i^n(h)|}{\max\{1, |\mathcal{U}_n^T| + |\mathcal{U}_n^C|\}}. \tag{D1}$$

Learning stops at the first iteration $n \geqslant n_{\min}$ for which $\bar{\Delta}_n \leqslant \delta$ for p consecutive iterations, or at $N_{\max}$, whichever comes first. In the reported runs, $n_{\min} = 100$ is the minimal number of iterations before the stopping rule is checked, $\delta = 0.003$ is the threshold, and $p = 8$ is the window. After convergence, the deployment-policy extraction runs PUCT with the longer horizon H_{dep} and lower count temperature τ_{dep}, yielding the deployment policy $(\alpha^\star, \beta^\star)$ that we report.

6.3.5 Tabular learning algorithm

Algorithms 3–4 give pseudocode for the learning iteration described above. The first algorithm advances the population through one learning iteration and records root policies, maximum root action values, and realized transitions. The second uses those records to update the empirical model, policy priors, and value table, and then computes the deployment policy. The index n labels one learning iteration that involves R matching rounds and the associated consumption decisions. The rounds within iteration n are indexed by r. The population state $\mathcal{P}_{\mathcal{N}}^n = \{(i_a, h_a^n)\}_{a=1}^{\mathcal{N}}$ records each agent's type and holding at the start of iteration n. The holding paths in the figures use the empirical distribution after iteration n, when $\mathcal{P}_{\mathcal{N}}^{n+1}$ has been formed. Within an iteration, $\tilde{h}_a$ denotes the current holding, h_a^+ the post-trade holding, and h_a^{++} the post-consumption holding. The random matching $\mathcal{B}_{n,r}$ is the set of unordered pairs in round r, and

$$\chi = \mathbb{1}\{k \neq j, \lambda_a = \lambda_b = 1\}$$

is the trade-success indicator for a matched pair.

The operators PUCT_T and PUCT_C are the trade- and consumption-node planners from Section 6.3. The dataset $\mathcal{D}_n$ contains the realized records from iteration n,

and $\mathrm{Emp}(\hat{M}, \xi)$ denotes the one-record empirical-model update described in Section 6.3.1, where ξ is a single realized record. These updates are applied as records arrive, so the model table can change during an iteration. The root-record multisets are Π_n^T (the visit policies returned by PUCT_T at the trade roots encountered in iteration n), Π_n^C (the visit policies returned by PUCT_C at the consumption roots encountered in iteration n), and $\mathcal{V}_n$ (the maximum root action values associated with each of those calls). The operator $\mathrm{Match}(\cdot)$ returns a uniformly random partition of the finite population into unordered pairs. The PUCT planner is created once per iteration and memoizes the root policy and maximum root action value for each queried trade root (i, k, j) and consumption root (i, h). The operator Fix $F_{\sigma^\star}$ returns the invariant distribution of the law of motion (F1)–(F4) under the policy $\sigma^\star$.

The notation $\mathrm{PUCT}_{\mathrm{dep}}$ denotes the deployment call to PUCT after stopping. It uses the empirical model and policy-prior tables at index $n_\star$, with horizon H_{dep} and count temperature τ_{dep}. It computes $(\alpha^\star, \beta^\star)$ but does not add records or update $\hat{M}$.

Algorithm 3 Self-play data collection in iteration n

Require: Population $\mathcal{P}_{\mathcal{N}}^n$, tables $(\alpha^n, \beta^n, \upsilon^n, \hat{M}^n)$, matching rounds R, training horizon H, temperature τ, and PUCT coefficient c_{puct}.

Ensure: Updated population $\mathcal{P}_{\mathcal{N}}^{n+1}$, updated model $\hat{M}^{n+1}$, and root records $(\Pi_n^T, \Pi_n^C, \mathcal{V}_n)$.

$\mathcal{D}_n, \Pi_n^T, \Pi_n^C, \mathcal{V}_n \leftarrow \varnothing$; $\tilde{h}_a \leftarrow h_a^n$ for all agents.
for $r = 1, \ldots, R$ **do**
  Draw random matching $\mathcal{B}_{n,r} \sim \mathrm{Match}(\{(i_a, \tilde{h}_a)\}_{a=1}^{\mathcal{N}})$.
  for all $(a, b) \in \mathcal{B}_{n,r}$ **do**
    Observe partner goods; update meeting-frequency counts in $\hat{M}^n$.
    Run PUCT_T at roots $(i_a, \tilde{h}_a, \tilde{h}_b)$ and $(i_b, \tilde{h}_b, \tilde{h}_a)$ against $\hat{M}^n$.
    Sample the two offer decisions from the root visit policies.
    Exchange holdings iff both agents offer and the goods differ; otherwise leave holdings unchanged.
    Record trade root policies, maximum root action values, transitions, and reciprocal-acceptance observations; append them to $\hat{M}^n$ online.
  end for
  for $a = 1, \ldots, \mathcal{N}$ **do**
    Run PUCT_C at root $(i_a, \tilde{h}_a)$ against $\hat{M}^n$ and sample the consumption decision.
    Apply the physical consumption/production transition, record the root policy and maximum root action value, and append them to $\hat{M}^n$ online.
  end for
end for
Set $\hat{M}^{n+1}$ to the resulting empirical table and return $\mathcal{P}_{\mathcal{N}}^{n+1} = \{(i_a, \tilde{h}_a)\}_{a=1}^{\mathcal{N}}$, $\hat{M}^{n+1}$, and root records.

Table 1 lists the common numerical settings for both learning cases. The fully em-

Table 1: Parameters used in the learning runs.

Parameter	Value
Population size $\mathcal{N}$	1800
Maximum training iterations $N_{\max}$	180
Matching rounds per iteration R	35
Minimum iterations $n_{\min}$	100
Stability tolerance δ	0.003
Patience window p	8
PUCT simulations per training root	32
Training PUCT horizon H	10
Evaluation PUCT simulations per root	96
Evaluation PUCT horizon H_{dep}	14
Fixed-policy evaluation rounds	120
Evaluation record interval	5
PUCT exploration coefficient c_{puct}	1.5
Training count temperature τ	1.0
Evaluation count temperature τ_{dep}	0.35
Initial policy learning rate ζ_0	0.45
Learning-rate decay	0.94
Minimum policy learning rate	0.01
Empirical-model prior count κ_0	2.0
Exploration floor ε	0.03

Algorithm 4 Tabular heterogeneous learning and reporting

Require: Initial population $\mathcal{P}_N^0 = \{(i_a, h_a^0)\}_{a=1}^N$, persistent tables $(\alpha^0, \beta^0, \upsilon^0, \hat{M}^0)$, iteration limits $(N_{\max}, R)$, horizons (H, H_{dep}), temperatures $(\tau, \tau_{\text{dep}})$, learning rates (ζ_n), projection floor ε, PUCT coefficient c_{puct}, and stopping parameters $(n_{\min}, \delta, p)$.

Ensure: Deployment policy $(\alpha^\star, \beta^\star)$, stationary distribution $\mu^\star$, and acceptance probabilities $Q^\star$.

$n_\star \leftarrow N_{\max}$

for $n = 0, \dots, N_{\max} - 1$ **do**

  Run Algorithm 3 to obtain $\mathcal{P}_N^{n+1}$, $\hat{M}^{n+1}$, and $(\Pi_n^T, \Pi_n^C, \mathcal{V}_n)$.

  Compute rootwise means $\bar{\pi}_i^{T,n}$, $\bar{\pi}_i^{C,n}$, and $\overline{V}_i^n$.

  Update table entries:

$$\alpha_i^{n+1}(k,j) \leftarrow \text{clip}_\varepsilon\Big[(1-\zeta_n)\alpha_i^n(k,j) + \zeta_n \bar{\pi}_i^{T,n}(1 \mid k,j)\Big],$$

$$\beta_i^{n+1}(h) \leftarrow \text{clip}_\varepsilon\Big[(1-\zeta_n)\beta_i^n(h) + \zeta_n \bar{\pi}_i^{C,n}(1 \mid h)\Big], \quad \upsilon_i^{n+1}(h) \leftarrow (1-\zeta_n)\upsilon_i^n(h) + \zeta_n \overline{V}_i^n(h).$$

  Compute $\bar{\Delta}_n$ from (D1).

  if $n \geqslant n_{\min} + p - 1$ and $\bar{\Delta}_t \leqslant \delta$ for all $t = n - p + 1, \dots, n$ **then**

    $n_\star \leftarrow n + 1$ and **break**.

  end if

end for

$(\alpha^\star, \beta^\star) \leftarrow \text{PUCT}_{\text{dep}}(\hat{M}^{n_\star}, \alpha^{n_\star}, \beta^{n_\star}, \upsilon^{n_\star}, H_{\text{dep}}, \tau_{\text{dep}}, c_{\text{puct}})$

$\mu^\star \leftarrow \text{Fix } F_{\sigma^\star}$ with $\sigma^\star = (\alpha^\star, \beta^\star)$

Set $\mu^\star(j) \leftarrow \sum_{i \in I} \nu_i \mu_i^\star(j)$ and, for $\mu^\star(j) > 0$, compute $\eta_\ell^\star(j) \leftarrow \nu_\ell \mu_\ell^\star(j)/\mu^\star(j)$ with $\eta_\ell^\star(j) \leftarrow 0$ when $\mu^\star(j) = 0$.

$Q^\star(k,j) \leftarrow \sum_{\ell \in I} \eta_\ell^\star(j) \alpha_\ell^\star(j,k)$

pirical case estimates all one-period rules from play. The known-environment case, defined in Section 6.4, keeps these numerical settings but supplies the deterministic one-period economic rules, evaluates the agent's own future decisions greedily, and centers search values as bias-style targets.

6.4 Known-environment learner

The known-environment learner is a variant of the algorithm in Section 6.3 that gives agents the deterministic one-period rules of the economy as primitives, while still requiring them to learn what depends on others' play. The variant is motivated by the fictitious-play-like behavior of the fully empirical case in Economy D, as we show in Section 7.5. It changes three components of the algorithm: the one-step planning model, the interior-node backup, and the form of the value target.

First, the planning model changes. To keep the model objects distinct, $\hat{M}^n$ in Sec-

tion 6.3.1 denotes the maintained empirical table of cumulative count ratios; $\hat{M}^n_{\text{emp}}$ and $\hat{M}^n_{\text{rules}}$ below are the two planning models derived from it. $M(\sigma)$, introduced in Section 7.5, denotes the one-iteration population environment generated by policy σ. The fully empirical agent uses

$$\hat{M}^n_{\text{emp}} = \big(\{\hat{m}^n_i\}, \{\hat{r}^{T,n}_i, \hat{P}^{T,n}_i, \hat{r}^{C,n}_i, \hat{P}^{C,n}_i\}_{i\in I}\big),$$

in which every reward and transition is a sample average. The known-environment agent uses

$$\hat{M}^n_{\text{rules}} = \big(\{\hat{m}^n_i\}, \hat{q}^n, \{0,\ P^T_{\hat{q}^n},\ r^C_i,\ P^C_i\}_{i\in I}\big),$$

in which only the partner-good distribution $\hat{m}^n_i$ and the reciprocal-acceptance probability $\hat{q}^n(k,j)$ are still learned from meetings. Everything else is plugged in from the primitives: trade-stage rewards are zero; $P^T_{\hat{q}^n}$ is the kernel T_q from Section 4.1 evaluated at $q = \hat{q}^n$; $r^C_i(h,\gamma) = r_i(h,\gamma)$ is the reward in (R); and P^C_i is induced by the next-holding map $c_i(h,\gamma)$ defined before (S1). The agent still receives no information about the equilibrium — she sees only partner-good frequencies and the empirical reciprocation rate — but search can now evaluate counterfactual actions with the correct one-step consequences even when the current population rarely takes them.

In the terms of Camerer and Ho [1999], the two planning models differ in how much foregone-payoff information they supply to the search. Under $\hat{M}^n_{\text{emp}}$, an offer the population rarely makes is backed by almost no realized experience, so the search has little basis on which to value it. This is close to the case that EWA associates with $\delta \approx 0$, where attractions move only with payoffs actually received. Under $\hat{M}^n_{\text{rules}}$, the deterministic one-period rules of any trade are known, so the search can value it even on states the population avoids. This is closer to the EWA case $\delta > 0$, in which foregone payoffs are weighted alongside realized ones. A known-environment agent is handed enough of the physical environment to evaluate trades she does not currently attempt.

Second, the agent's own future actions inside the planning horizon are evaluated greedily, while her current action is still drawn from the normalized PUCT visit counts. The planner therefore stops treating its future self as committed to yesterday's mixed strategy.

Third, if $\overline{V}_i^n(h)$ is the average scalar value recorded at iteration n for type i holding h, the value update is

$$v_i^{n+1}(h) \leftarrow (1-\zeta_n)v_i^n(h) + \zeta_n\left[\overline{V}_i^n(h) - \overline{V}_i^n(k_i^{\text{ref}})\right], \qquad v_i^{n+1}(k_i^{\text{ref}}) = 0,$$

which uses the same iteration index n and learning rate ζ_n as the policy update in Section 6.3.3. The subtraction normalizes the target relative to a chosen reference holding, removing the level without requiring a full policy evaluation.

7 Computational Results

We apply the algorithms of Section 6 to compute the stationary Nash equilibrium, the non-Nash restricted-information SCE, and the tabular learning outcomes. We first report results for the fully empirical case. The last subsection compares the fully empirical case to the known-environment case defined in Section 6.4. The SNE is the complete-information, correct-belief reference. It identifies which goods can sustain a medium-of-exchange convention and, when multiple equilibria exist, characterizes the alternatives.

We measure how far the learned policy is from a stationary Nash equilibrium by its *complete-information best-response gain*: the largest increase in long-run average payoff that any type can obtain by replacing the learned policy with its complete-information stationary best response, evaluated at the invariant distribution generated by the learned policy. The gain is zero at a Nash equilibrium and positive whenever some type can profitably deviate.

We report the holding distributions under the different equilibrium concepts and under the learned policy, then the holding distribution observed in self-play simulations, and lastly the learned trade policy.

Each holding figure has four panels: an SNE, an SCE that is not a Nash equilibrium, the stationary law of the learned policy, and the terminal distribution observed in self-play simulations. Economies A2 and B add a fifth speculative panel because those economies have two Nash equilibrium branches reached by different initial policies. The simulated-terminal panel is computed by simulating the converged de-

ployment policy $(\alpha^\star, \beta^\star)$ for 120 random-matching rounds and recording the holding distribution every five rounds; no PUCT calls are made during this fixed-policy evaluation.

The holding-dynamics panel reports aggregate holding shares observed during the learning simulations. The solid lines are population shares during learning, and the dashed horizontal lines are aggregate shares under the SNE. These paths record how the holding distribution evolves while the model and policy are still being revised.

Before turning to the per-economy results, we define the convergence diagnostic used below to distinguish points on the learning path from a stationary equilibrium. At each iteration t, the diagnostic records the current type-by-good distribution μ_t^{sp} and policy σ_t, where the superscript "sp" denotes the self-play path, and computes the stationarity residual and the complete-information best-response gain,

$$d_t^{\mathrm{sp}} = \left\| F_{\sigma_t}(\mu_t^{\mathrm{sp}}) - \mu_t^{\mathrm{sp}} \right\|_\infty, \qquad b_t^{\mathrm{sp}} = \max_i \{ g_i^{\mathrm{BR}}(\mu_t^{\mathrm{sp}}, \sigma_t) - g_i^{\sigma}(\mu_t^{\mathrm{sp}}, \sigma_t) \}.$$

Here $g_i^{\mathrm{BR}}(\mu, \sigma)$ is the type-i complete-information best-response average reward in the stationary environment (μ, Q^σ) induced by (μ, σ) through (B1)–(B2), obtained by solving the Bellman problem (B3)–(B4) for type i at (μ, Q^σ); $g_i^\sigma(\mu, \sigma)$ is the type-i average reward under the current policy σ_i evaluated in the same environment. The learning path is said to have settled near a stationary Nash equilibrium only when both residuals are small.

7.1 Economy A1

Recall that Economy A1 is the three-good Kiyotaki–Wright environment. The storage costs are $s_1 = 0.1$, $s_2 = 1$, and $s_3 = 20$, and the production function is given by $f(1) = 2$, $f(2) = 3$, and $f(3) = 1$. Figures 2 and 3 report the results.

The SNE is the commodity-money allocation in which good 1 serves as the universal medium of exchange. Type 3, who produces good 1, holds it with probability 1 while seeking the consumption good 3. Type 1 holds the production good 2 with probability 1 while seeking good 1. Type 2 splits evenly between good 1 and the

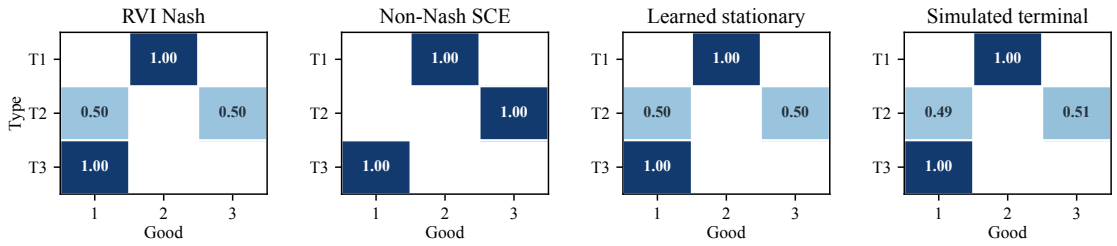

Figure 2: Economy A1. Stationary holdings.

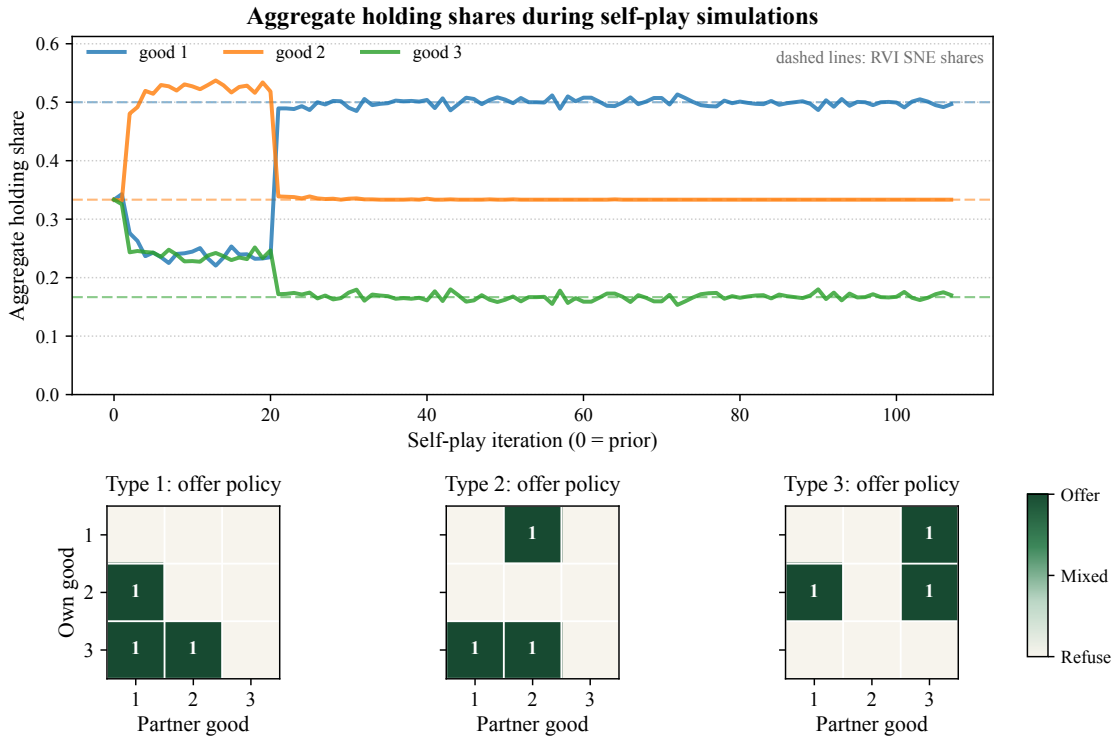

Figure 3: Economy A1. Holding shares during self-play simulations and final reported trade policy.

production good 3 (0.50/0.50). Type 2 spends half the time holding good 3, just produced or recently received, and the other half holding commodity money (good 1) acquired by trading away good 3 while en route to consuming good 2. The aggregate holding distribution assigns roughly half of all holdings to good 1, because it circulates as the common medium.

The non-Nash SCE is autarkic. Every type holds only its production good. Type 1 holds good 2, type 2 holds good 3, and type 3 holds good 1. No exchange takes place. The outcome is self-confirming because agents who never offer trade have no data from which to learn that trading would raise their payoffs. Off-path beliefs about whether a partner would reciprocate a trade offer are never tested and so never revised. The autarky convention is individually rational given these untested beliefs. When an agent believes her partner will refuse, it is optimal to refuse in turn, and the refusal is never contradicted by experience.

The learned invariant holding distribution reproduces the commodity-money convention. The learned stationary matrix matches the RVI benchmark entry by entry to displayed precision. The learning path shows how the convention is built up. The population begins near equal aggregate shares. During learning the holding distribution first overshoots, carrying too much good 2, and then settles close to the commodity-money distribution. At convergence, good 1's aggregate share is about 0.49 and good 3's share is about 0.18.

Measured in aggregate holdings, iterations 1–20 lie closer to the autarkic non-Nash SCE than to the SNE, and iterations 21–107 lie closer to the SNE. Repeated interaction eventually pulls the population from the autarkic neighborhood to the commodity-money convention.

7.2 Economy A2

Economy A2 inherits A1's production cycle and storage costs but raises all consumption utilities to $\bar{u}_i = 500$. The higher payoff changes the economics of speculative acceptance. At A1's utility level, carrying good 3 (storage cost 20) in anticipation of resale is too expensive relative to the consumption gain. At A2's utility level, the gain from eventually reaching one's consumption good is large enough that accept-

ing good 3 as a speculative intermediary becomes worthwhile when other agents are expected to accept it as well. The Kiyotaki–Wright condition of Section 5 is violated at $\bar{u} = 500$, so the speculative branch is relevant and the policy at the speculative entry state (type 1's offer of good 2 for good 3) matters. Figure 4 reports the fundamental and speculative equilibria, and Figure 5 reports the self-play path and trade-policy diagnostic.[9]

The two RVI benchmarks computed from different initial policies are distinct in structure and economic interpretation. The fundamental Nash equilibrium is identical to A1's. Type 3 holds good 1, type 1 holds good 2, and type 2 splits evenly between good 1 and good 3. In the speculative Nash equilibrium, type 1 now holds good 2 (0.71) *and* good 3 (0.29), because type 3—whose consumption good is good 3—will accept it, creating a speculative resale opportunity for type 1. Type 2 similarly holds good 1 (0.59) and good 3 (0.41). Good 3 circulates far more widely than in the fundamental branch. Sustaining the speculative equilibrium requires all types to participate in this chain of trade, and the RVI algorithm finds this branch by initializing from the speculative offers.

The non-Nash SCE in the benchmark panel has the same holding allocation as the fundamental SNE. It has the same stationary holding matrix to displayed precision: type 1 holds good 2, type 3 holds good 1, and type 2 splits between goods 1 and 3. However, it is not an SNE because the policy is not a complete-information best response. The recurrent commodity-money trades coincide between the SCE and the fundamental SNE. The two diverge at the speculative entry state: type 1's offer of good 2 for good 3. In the reported SCE, type 1 believes that such an offer will be rejected and therefore sets $\alpha_1(2,3) = 0$. Under the objective acceptance kernel at the same holding distribution, however, a type 2 agent holding good 3 would in fact accept good 2. Type 1's complete-information best response is therefore $\alpha_1(2,3) = 1$, using good 3 as an intermediary on the way to her consumption good. This deviation raises type 1's average payoff by about 21.4[10], so the SCE is self-confirming but not

[9]For comparison, Figure 12 in Section 7.6 shows the offer policy obtained by the known-environment learner.

[10]The gain of 21.3 reported in Table 2 is the same deviation evaluated at the slightly different learned policy.

Nash. Relative to the speculative RVI Nash equilibrium, $\alpha_1(2,3) = 0$ is precisely the missing policy link: the offer that would route good 3 into type 1's holdings and sustain the speculative chain.

The learned invariant holding distribution coincides with the fundamental SNE, but the reported deployment policy is close to the non-Nash SCE. The learning path also shows a long plateau before the final policy selection stabilizes. From iterations 2–70, aggregate good 2 stays near one half and aggregate good 1 stays near one quarter. At iteration 40 the aggregate holding vector is approximately $(0.236, 0.524, 0.240)$, and at iteration 70 it is $(0.234, 0.527, 0.239)$. This plateau coincides with neither the speculative Nash equilibrium nor the non-Nash SCE: it is a transient convention in which good 2 temporarily serves as an intermediary, with type 3 giving up good 1 for good 2, type 2 trading good 3 for good 2, and type 1 trading good 2 for good 1. The transition near iterations 85–87 moves the aggregate distribution to the shared holding matrix of the fundamental SNE and the non-Nash SCE.

7.3 Economy B

Economy B has two Nash equilibria that differ in which goods circulate as media of exchange. The learning algorithm settles on the fundamental equilibrium. Good 1 becomes commodity money and good 2 co-circulates, matching the RVI benchmark entry by entry to displayed precision (Figures 6 and 7).

The production map is $f(1) = 3$, $f(2) = 1$, $f(3) = 2$ with storage costs $(1, 4, 9)$. Type 2 produces good 1 (the cheapest), type 3 produces good 2, and type 1 produces good 3 (the most expensive).

In the fundamental Nash equilibrium, good 1 is the commodity money. Type 2 holds good 1 exclusively. Type 2 produces good 1 and carries it until trading for the consumption good 2, which she then consumes immediately. Type 3 holds good 1 (0.59) and good 2 (0.41), accepting good 1 as commodity money from type 2 and holding the production good 2 while seeking good 3. Type 1 holds good 3 (0.71) and good 2 (0.29), mainly carrying the costly production good but accepting the cheaper good 2 as a lower-cost stepping stone toward the consumption good 1. Good 1

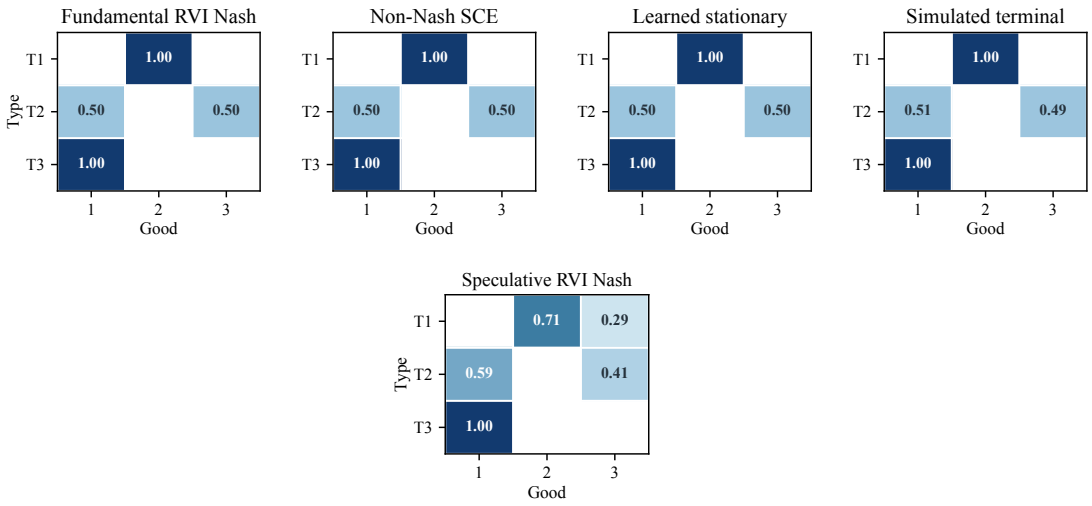

Figure 4: Economy A2. Fundamental and speculative stationary holdings.

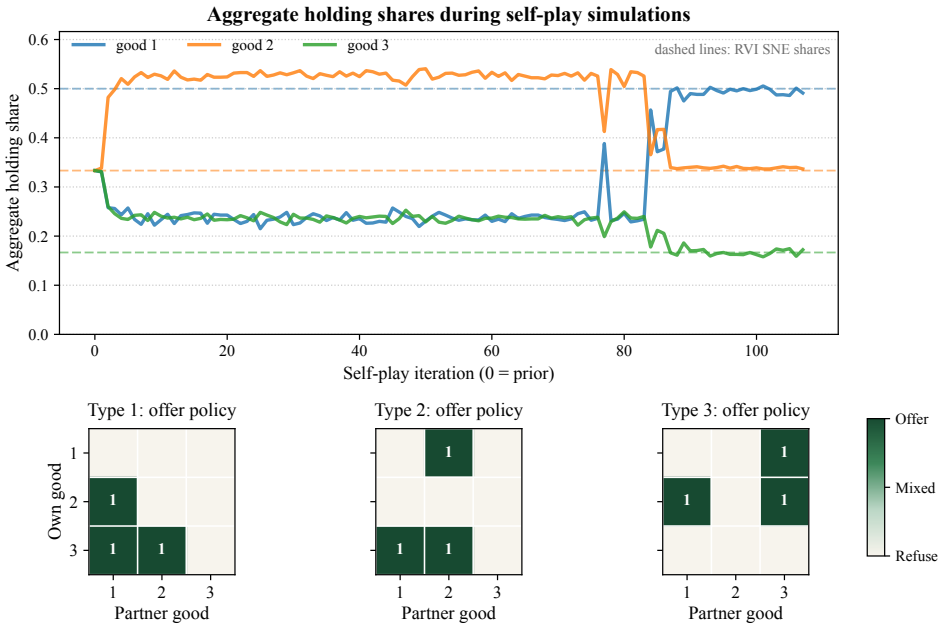

Figure 5: Economy A2. Holding shares during self-play simulations and final reported trade policy.

accounts for about 0.53 of aggregate holdings.

In the speculative Nash equilibrium, goods 2 and 3 circulate instead. Type 3 holds good 2 exclusively (1.00). Type 2 now holds good 1 (0.71) *and* good 3 (0.29), speculatively accepting good 3 because type 3 consumes good 3 and will accept it in exchange. Type 1 holds good 2 (0.59) and good 3 (0.41), accepting good 2 as a stepping stone since type 2 takes good 2 in exchange for good 1. Good 3, the most expensive good to store, circulates as a medium.

An example non-Nash SCE is a third, inferior outcome in which good 2 disappears from circulation entirely. Type 2 and type 3 both pile into good 1. Type 1 splits between good 1 and good 3 (0.50 each). Type 3 gives up her role as the good 2 producer in the circulation network, since she holds only good 1. Both Nash equilibria require good 2 to flow, and in the SCE it does not, so neither exchange chain can form. The outcome is self-confirming. Because no one offers good 2, no one ever learns it would be accepted.

The learning path recovers the fundamental holding distribution. Type 2 holds good 1 (1.00). Type 3 holds good 1 (0.59) and good 2 (0.41). Type 1 holds good 3 (0.71) and good 2 (0.29). All three match the SNE holding matrix to displayed precision. The population moves quickly toward the fundamental aggregate composition. Good 1 is already above 0.51 within the first few learning iterations and ends near 0.52, while goods 2 and 3 each remain near one quarter. As in Economy A2, building the speculative equilibrium requires coordinated acceptance of good 3 before any agent has seen evidence that partners accept it. The learning path does not generate that belief. The fundamental convention forms first, from storage-cost minimization alone, and is never dislodged.

7.4 Economy C

Economy C augments the A1 production cycle with fiat money (good 0). Good 0 has zero storage cost, yields no consumption utility, cannot be consumed, and is held in fixed aggregate supply $M_0 = 0.32$ as a share of the population (in the $\mathcal{N} = 1800$ simulations this corresponds to 576 units). Storage costs rise to $(s_0, s_1, s_2, s_3) = (0, 9, 14, 29)$, with commodity-good costs substantially higher than in A1. The higher

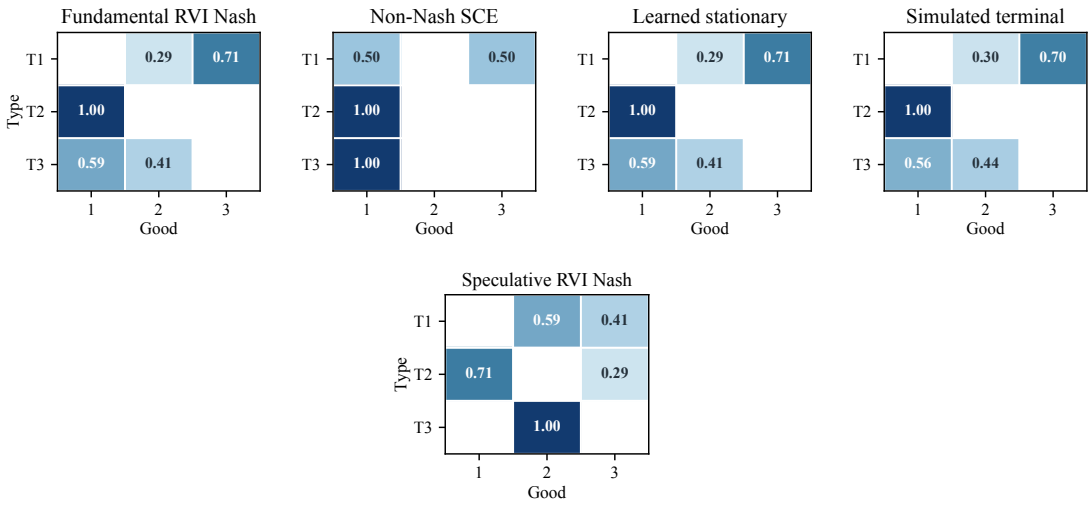

Figure 6: Economy B. Fundamental and speculative stationary holdings.

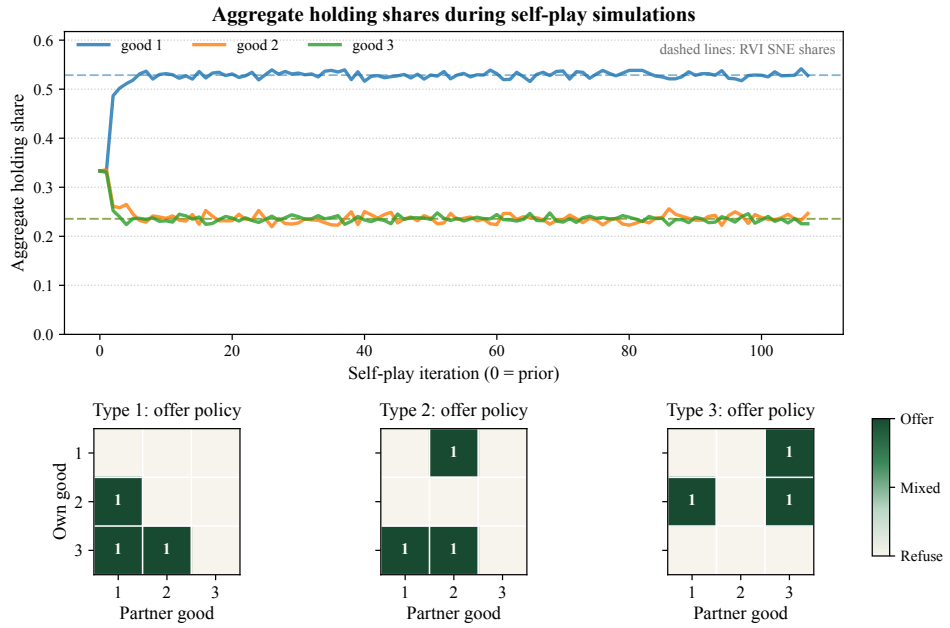

Figure 7: Economy B. Holding shares during self-play simulations and final reported trade policy.

costs make commodity exchange expensive and give every type a strong incentive to economize on storage costs by holding fiat money instead. Unlike commodity money, fiat money has no intrinsic value. It is accepted only because agents expect others to accept it in turn, so its value is entirely expectational. Figures 8 and 9 report the stationary holdings and dynamic-policy diagnostics.

The SNE is the fiat-money equilibrium. All three types hold both fiat money and commodity goods simultaneously. Type 1 holds good 0 (0.26) and good 2 (0.74). Type 2 holds good 0 (0.31), good 1 (0.26), and good 3 (0.42). Type 3 holds good 0 (0.38) and good 1 (0.62). The aggregate fiat-money share is exactly 0.32, pinned by the exogenous supply constraint. No type holds fiat money exclusively. Money circulates through all three types as a common medium, and each type also holds commodity goods reflecting its place in the production cycle. The fiat-money shares across types (0.26 for type 1, 0.31 for type 2, 0.38 for type 3) reflect each type's position in the trade cycle and the storage costs of the commodity goods she carries while en route to her consumption good.

An example non-Nash SCE in Figure 8 is the low-trade holding convention. In that allocation, type 1 holds good 2 with probability 0.999, type 2 holds fiat money with probability 0.932, and type 3 holds good 1 with probability 0.973. In aggregate terms the SCE is approximately 0.320 for fiat money, 0.335 for good 1, 0.333 for good 2, and 0.012 for good 3. Almost the entire fiat stock is held by type 2, while good 3, the good type 2 produces after consuming good 2, is nearly absent. The allocation is self-confirming because the trades that would restart circulation are hidden behind off-path reciprocal-acceptance links. In the constructed SCE, type 1 would accept fiat money for good 2, type 2 would accept good 1 for good 3, and type 3 would accept good 2 for good 1. The corresponding spenders—type 2 offering good 0 for good 2, type 3 offering good 1 for good 3, and type 1 offering good 2 for good 1—do not make these offers in that SCE policy, so the missing links are not tested by experience. With no successful trade and no consumption, the holding law is stationary. Complete-information agents would deviate, so this rest point is self-confirming but not Nash.

The fully empirical learner lands on this low-trade allocation rather than on the

fiat-money SNE. The self-play path initially comes close to the fiat-money benchmark. At iteration 10 the aggregate holding vector is approximately 0.320, 0.298, 0.244, and 0.138 for goods 0 through 3, close to the SNE shares 0.320, 0.293, 0.246, and 0.141, and the path returns to that neighborhood late in training. The jump after iteration 96 moves it away from the SNE and toward the non-Nash SCE allocation. The deployed final policy has stationary aggregate shares 0.320, 0.335, 0.333, and 0.012 for goods 0 through 3, matching the SCE panel to displayed precision, and the post-convergence simulation ends at essentially the same distribution. Thus the money stock remains pinned at 0.32 by the supply constraint, but money is not a common medium of exchange in the SNE sense. The alternating bands in Figure 9 suggest that the monetary convention is fragile under fully empirical learning: changes in learned reciprocal-acceptance rates are associated with switches between a near-fiat-money cycle and the low-trade SCE allocation.

The offer policy identifies the broken links. In the SNE, and in the known-environment run in Section 7.6, type 2 spends fiat money to obtain her consumption good, so $\alpha_2(0,2) \simeq 1$ and $\alpha_2(2,0) \simeq 0$. The final fully empirical policy reverses this link: $\alpha_2(0,2) = 0.001$ and $\alpha_2(2,0) = 1.000$. Type 2 is therefore willing to receive fiat money for good 2 but almost never spends fiat money to obtain good 2. Since type 2 produces good 3 only after consuming good 2, good 3 drains from the population. A second missing reciprocal link traps the other side of the cycle. Type 1 is willing to spend good 2 for good 1, with $\alpha_1(2,1) = 1.000$, but the dominant holders of good 1, type 3, almost never accept good 2 in return: $\alpha_3(1,2) \simeq 0$. The type-level holding matrix is the consequence: type 1 is stuck with good 2, type 2 is stuck with fiat money, and type 3 is stuck with good 1.

7.5 Economy D

Economy D is the five-type pure barter economy of Marimon et al. [1990], which has no analytic Kiyotaki–Wright equilibrium characterization. The production map $f(1) = 3$, $f(2) = 4$, $f(3) = 5$, $f(4) = 1$, $f(5) = 2$ offsets production and consumption. Type 4 produces good 1 (storage cost 1), type 5 produces good 2 (cost 4), type 1 produces good 3 (cost 9), type 2 produces good 4 (cost 20), and type 3 produces

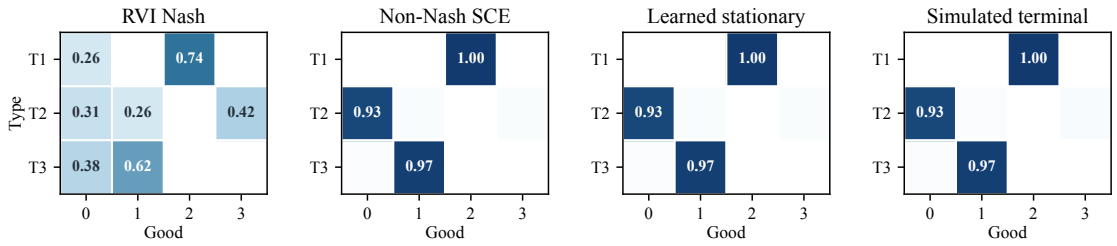

Figure 8: Economy C. Stationary holdings.

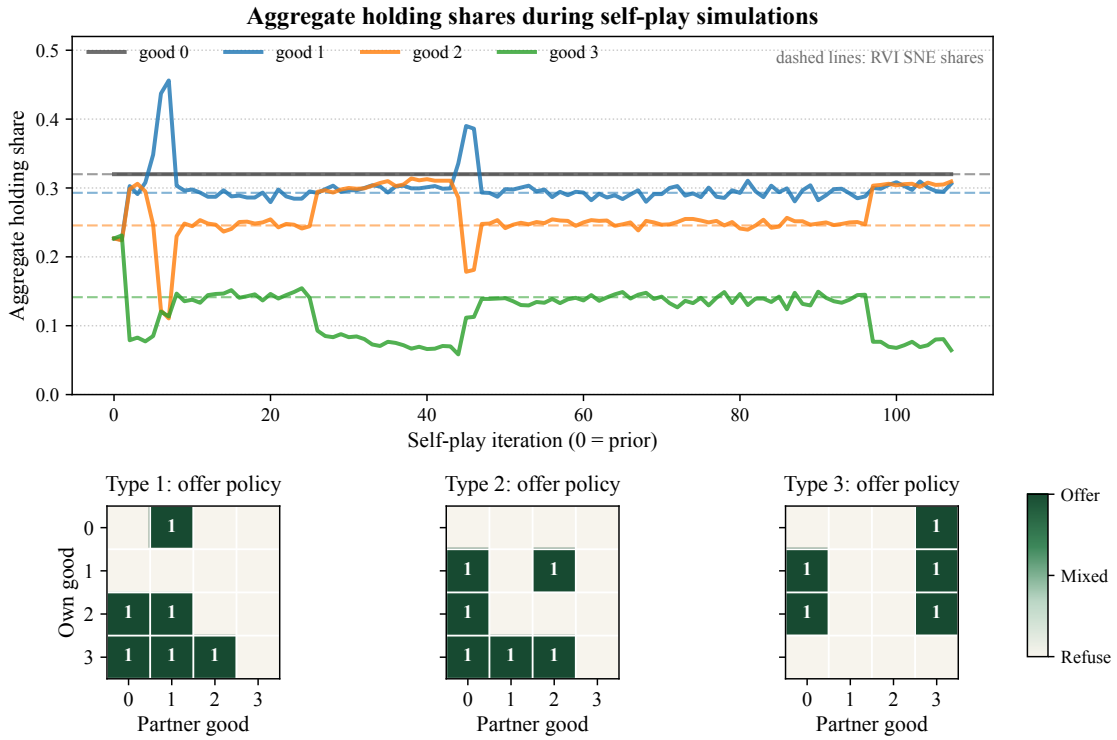

Figure 9: Economy C. Holding shares during self-play simulations and final reported trade policy.

good 5 (cost 30). Storage costs strongly favor good 1 as commodity money, but concentrating holdings on good 1 asks every type to participate in a five-step speculative acceptance chain, a substantially harder coordination problem than in the three-good economies. Figures 10 and 11 report the stationary holdings, aggregate holding dynamics, and offer policies.

In the SNE, good 1 emerges as commodity money and accounts for 0.612 of aggregate holdings. Most types use it as an intermediary. Type 4 holds it exclusively, since type 4 both produces good 1 and uses it as a bridge while waiting to consume good 4. Type 5 and type 2 follow a similar pattern. Each sheds its production good quickly and holds good 1 while seeking its consumption good (type 5 holds good 1 at 0.77, type 2 at 0.73). Type 3 faces a harder problem. Her production good 5 is expensive, so she gets stuck carrying good 5 part of the time (share 0.37), though she still manages to hold good 1 for the majority of the time (0.57). Type 1 has the most difficult path. Her production good 3 is hard to move, so type 1 holds it nearly all the time (0.89) and only occasionally acquires good 2 as a cheaper stepping stone toward good 1.

The non-Nash SCE is simpler. Nearly every type settles on good 1 as the sole store of value. Types 3, 4, and 5 hold good 1 exclusively, and type 2 holds it with probability 0.72. The outlier is type 1, who holds nothing but her production good 3. Nobody accepts good 3, so type 1 has no route to her consumption good. The multi-step exchange chains that would let type 1 trade her way to good 1 never form, because the other types have already decided not to participate. The result is a self-fulfilling collapse. Consensus around good 1 is even tighter than in the SNE, and yet that very consensus leaves type 1 permanently in autarky, making the outcome less efficient overall.

The learned policy falls short of the SNE benchmark. The invariant distribution assigns good 1 an aggregate share of 0.439, against an SNE value of 0.612. The stationary aggregate holding distance is 0.173, or 0.179 in total variation. Several type-level distortions drive the gap. Type 2 holds good 1 with probability 0.411 rather than 0.725. Type 3 holds good 1 with probability 0.323 rather than 0.566 and carries good 5 too often. Type 5 holds good 1 with probability 0.460 rather

than 0.768. Type 1 spreads its holdings over goods 2, 3, and 4 rather than mostly holding good 3. Only type 4 remains close to the SNE value. The learning trajectory tells the same story. Good 1 rises sharply early on, reaches 0.571 at iteration 15, then falls back and ends near 0.447 in the self-play path. The complete-information best-response gain under the invariant learned policy is 1.71, contributed by type 2. The offer-policy heat maps show the same pattern, with each type offering to trade for a wider range of partner goods than the SNE requires.

The learning path converges to a stable convention, but not to the SNE. The reason lies in how the empirical model $\hat{M}^n$ is updated. Each entry of $\hat{M}^n$, whether a partner-good frequency, a trade or consumption transition probability, an average reward, or a reciprocal-acceptance rate, is a ratio of cumulative counts. The numerator records the relevant outcomes accrued since the start of training, and the denominator records the number of opportunities for that outcome. Each iteration adds new records to both numerator and denominator without discarding any earlier ones. For an entry that has already accumulated many records, the iteration-n additions therefore shift the cumulative ratio only by an amount of order $1/n$, regardless of how relevant they are to the environment generated by the current policy. In effect, agents at iteration n plan against a history-weighted average of all past environments rather than against the environment generated by the current policy. This is the defining structure of fictitious play [Brown, 1951, Robinson, 1951, Fudenberg and Levine, 1998]: old, out-of-equilibrium counts persist in the table indefinitely and are diluted only as $1/n$. Holding distributions from early iterations, when policies were diffuse and coordination was absent, become embedded in $\hat{M}^n$ and anchor beliefs away from the SNE.

The numerical results confirm this. The path average is $(0.453, 0.183, 0.174, 0.098, 0.091)$, and the learned partner-good beliefs track it closely. The invariant law under the final learned policy is $(0.439, 0.182, 0.186, 0.096, 0.096)$, well below the SNE share of 0.612 for good 1. The policy is internally stable with the invariant-law residual below 10^{-10} and the mean policy update below 10^{-3}. It is therefore a stable point of the learning process without being a fixed point of the best-response mapping — a pattern known from analyses of fictitious-play and stochastic-fictitious-play dy-

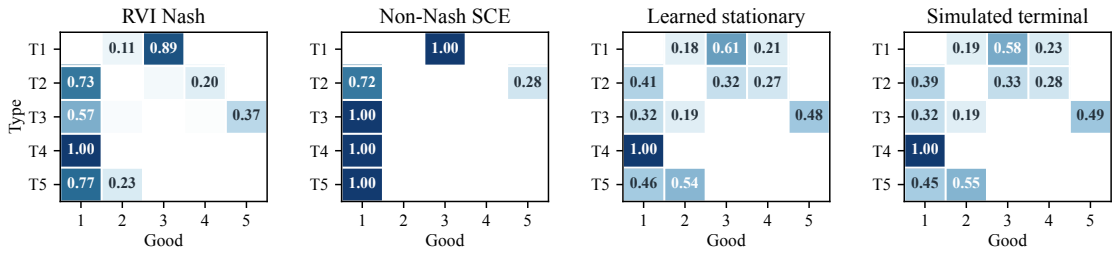

Figure 10: Economy D. Stationary holdings.

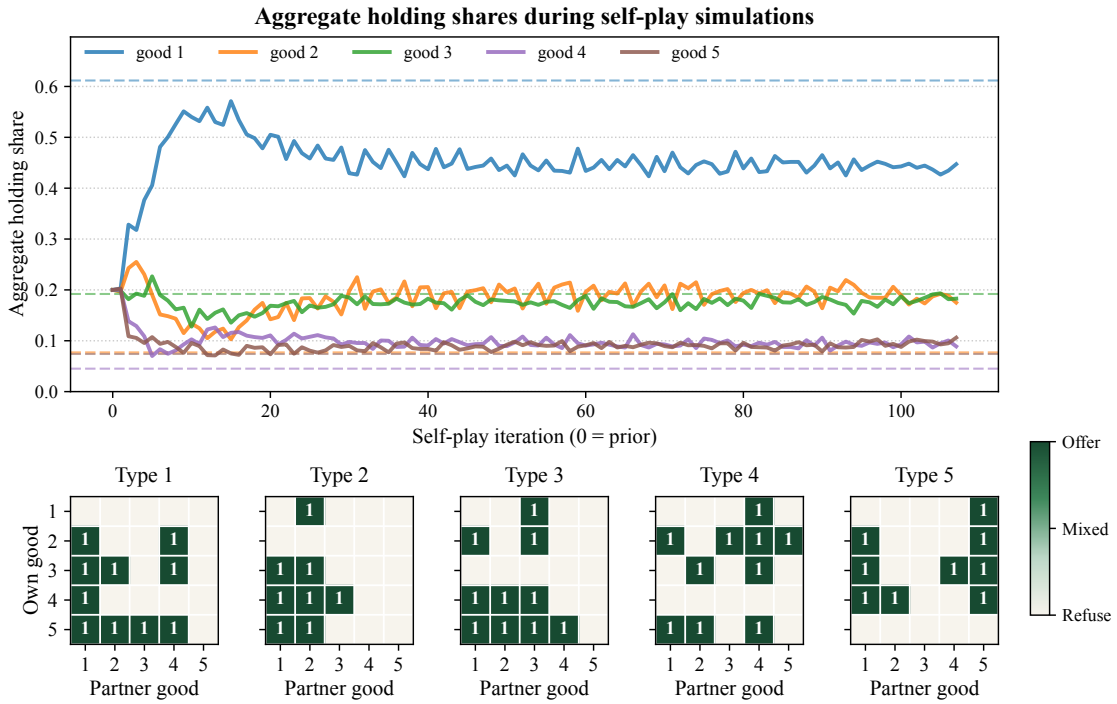

Figure 11: Economy D. Holding shares during self-play simulations and final reported trade policy.

namics [Fudenberg and Kreps, 1993, Hofbauer and Sandholm, 2002, Benaïm et al., 2009].

Once we switch to the known-environment case, we change what the planner has to learn. In the next subsection, agents are given the one-period economic rules as primitives: trade has zero current reward, consumption rewards and storage costs are known, and the post-consumption production map is known. They still learn partner-good frequencies and reciprocal-acceptance rates from meetings, so the equilibrium object is not supplied to them. The change is that search can evaluate rarely chosen counterfactual actions using the correct rules that the economy follows, rather than relying on noisy empirical transition and reward averages. The variant also evaluates the agent's own future decisions inside the planning horizon greedily, while the current action is still drawn from the PUCT visit-count policy, and it uses the average-reward normalization described in Section 6.4. With these changes, the learning path converges to the SNE in every economy, as we report in the next subsection.

7.6 Known-environment learning: results

In this subsection, we report the outcomes of the known-environment variant defined in Section 6.4. The best-response gain falls to zero up to numerical precision in all five economies. Table 2 reports the best-response gain together with two further diagnostics. The *stationary holding distance* is the maximum absolute difference, over goods, between the invariant aggregate holding distribution generated by the learned policy and the nearest SNE aggregate distribution. It measures how close the induced allocation is to the benchmark. The *final trade frequency* is the fraction of matched pairs that trade in the last learning iteration.

Stationary holding distance is an allocation diagnostic, not a Nash diagnostic by itself. Economy A2 shows the distinction: the fully empirical learner almost matches the fundamental holding distribution, but its policy still has a large best-response gain. Final trade frequency can also differ across policies that have the same holding distribution, because the policy is not pinned down at off-path or indifferent trade states.

Table 2: Fully empirical versus known-environment learning. The "Known env.*" columns package three changes to the baseline learner: deterministic one-period rules, greedy interior backup, and bias-style centering of value targets (see Section 6.4). Partner-good frequencies, reciprocal acceptance, the policy, and the continuation values are still learned from play.

	Best-response gain		Stationary holding distance		Final trade frequency	
Economy	Empirical	Known env.*	Empirical	Known env.*	Empirical	Known env.*
A1	1.50×10^{-12}	0	3.52×10^{-9}	3.52×10^{-9}	0.220	0.431
A2	21.3	0	1.40×10^{-6}	7.65×10^{-10}	0.266	0.388
B	0	0	3.59×10^{-9}	3.59×10^{-9}	0.313	0.392
C	12.8	0	0.130	1.74×10^{-8}	0.116	0.326
D	1.71	7.11×10^{-15}	0.173	8.69×10^{-9}	0.161	0.220

Table 2 and Figures 12, 13, and 14 show that the known-environment variant changes both the speed and the limit points of learning. The learned policy determines the transition law, and the transition law determines stationary holdings, but the map is many-to-one. A policy can reproduce an aggregate holding vector while still prescribing a suboptimal action in a counterfactual or strategically pivotal state. The known-environment planner narrows what must be learned. Agents no longer have to estimate storage costs, consumption rewards, production after consumption, or no-trade/refusal transitions from sparse observations. They only have to learn who they meet and how often offers are reciprocated. Greedy continuation evaluation then asks whether the current action leads to a good continuation policy, rather than evaluating it under yesterday's mixed policy. Less uncertainty therefore gives the search a sharper signal about which offers are optimal, especially at states that are rarely reached under the current convention.

Economy A2 is the three-good case where this distinction is already visible. In Figure 12, the known-environment run keeps the holding path at the fundamental commodity-money benchmark and changes the relevant offer policy enough to remove type 1's deviation. Thus the known-environment package matters not because it changes the desired allocation, but because it directs planning toward the offer policy that makes the allocation optimal.

Similarly, in Economy C (Figure 13), the aggregate path stays near the fiat-money benchmark from the beginning, final trade frequency rises to 0.326, and the invariant law under the learned policy matches the SNE allocation.

Economy D shows the same force in a larger state space. In Figure 14, the known-environment path moves toward the commodity-money composition almost immediately. By iteration 5 the aggregate total-variation distance from the SNE shares is about 0.018, by iteration 10 it is about 0.014, and by the last learning iteration it is about 0.007. The invariant policy then has stationary holding distance 8.69×10^{-9} and Nash best-response gain 7.11×10^{-15}. The final heat map reflects this: the relevant offer probabilities line up with the SNE commodity-money policy, while the remaining visible differences occur at low-probability or indifferent states.

8 Concluding Remarks

Relative to Marimon et al. [1990], we make the following contributions:

- We make explicit an average-reward dynamic-programming formulation of the individual exchange problem. Marimon et al. [1990] used classifier strengths as long-run average-reward accounting devices. We write the associated Bellman equations and gain and bias functions, and we compute complete-information stationary Nash benchmarks by relative value iteration (RVI).
- We separate complete-information Nash equilibrium computation from restricted-information learning. We define and compute a self-confirming equilibrium in which beliefs must be correct only for observed bilateral histories.[11]
- We replace Marimon et al.'s Holland classifier accounting system with a transparent tabular MuZero-inspired PUCT self-play procedure that records values, visit counts, learned transition models, and search policies.
- We add formal diagnostics for learning paths. The three points above allow us

[11] Marimon et al. were unaware of work on self-confirming equilibrium by Battigalli [1987], Battigalli and Guaitoli [1988], Battigalli [1993] or of the related subsequent work by Fudenberg, Kreps and Levine.

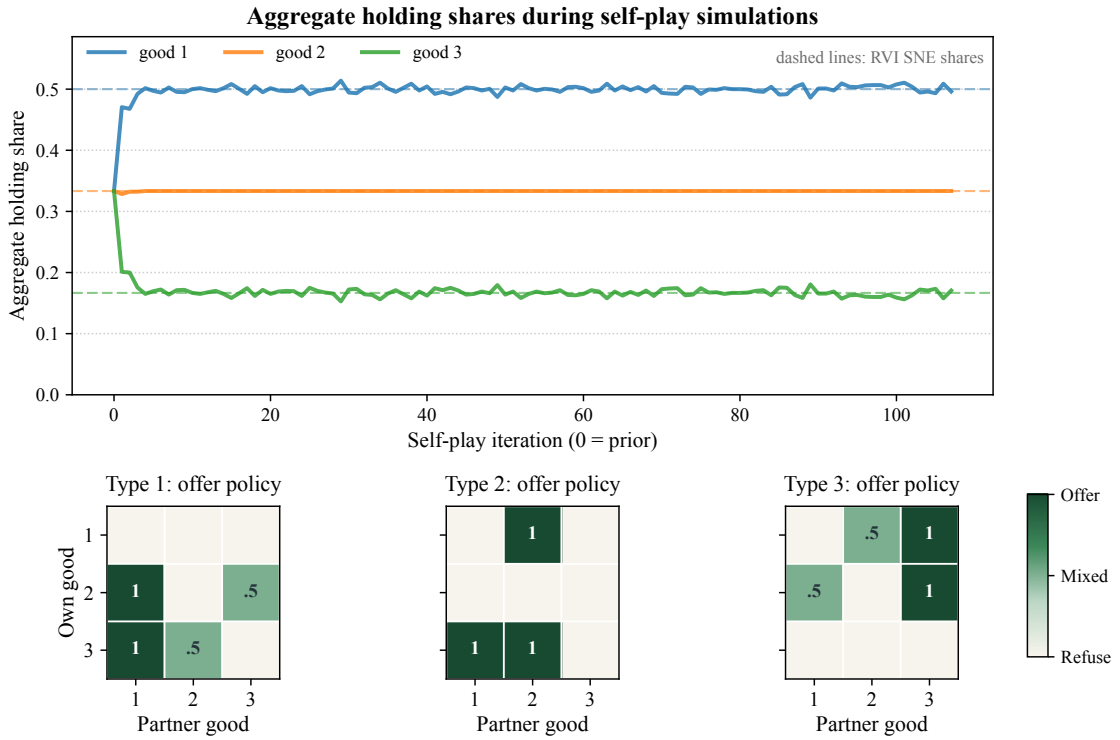

Figure 12: Economy A2. Known-environment learning path. The aggregate holding path starts from the production prior at iteration 0; dashed horizontal lines are the nearest RVI SNE aggregate shares. The lower panels report the final learned trade policy.

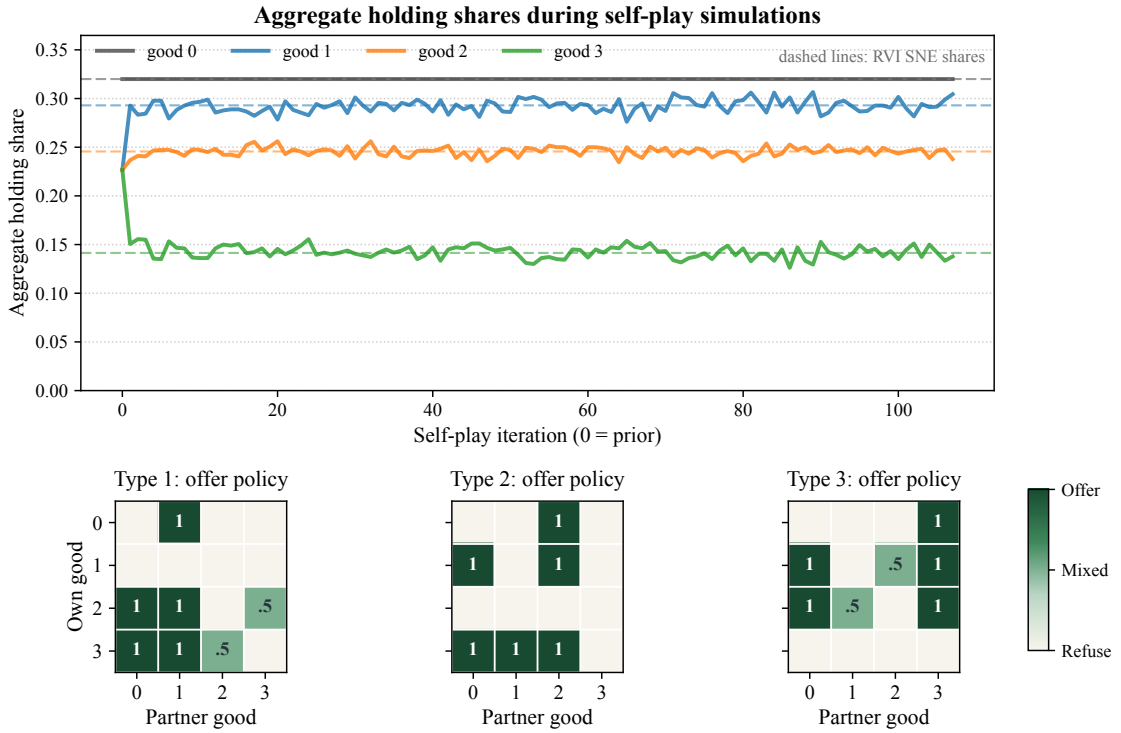

Figure 13: Economy C. Known-environment learning path. The aggregate holding path starts from the production prior at iteration 0; dashed horizontal lines are the nearest RVI SNE aggregate shares. The lower panels report the final learned trade policy.

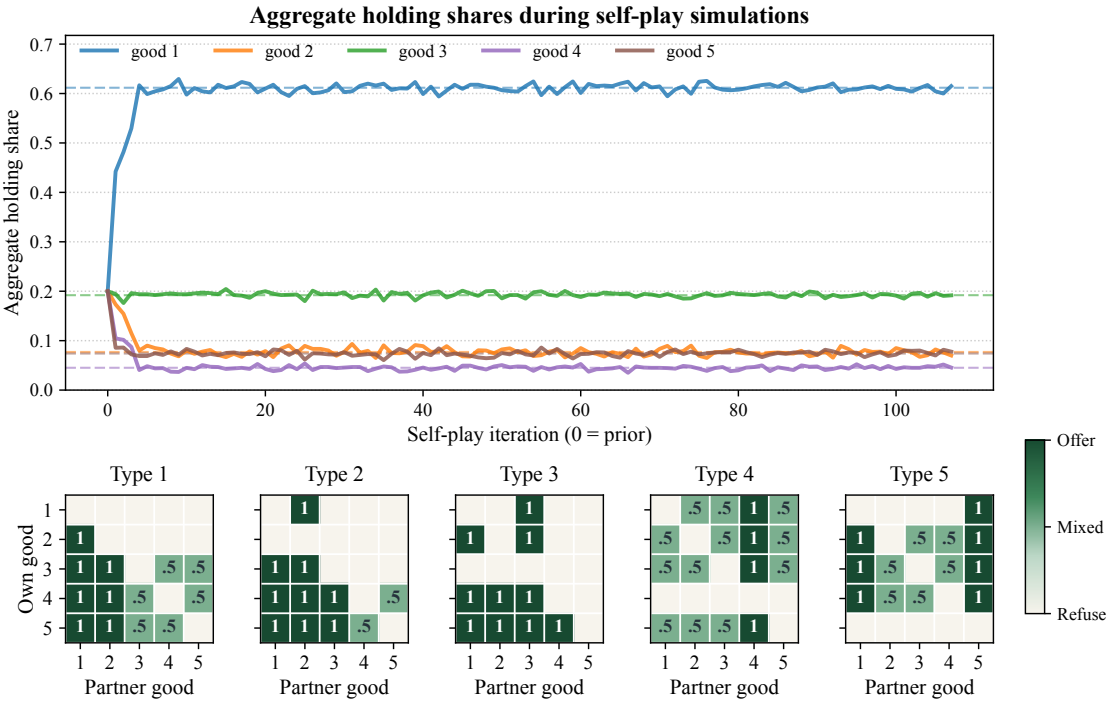

Figure 14: Economy D. Known-environment learning path. The aggregate holding path starts from the production prior at iteration 0; dashed horizontal lines are the nearest RVI SNE aggregate shares. The lower panels report the final learned trade policy.

to characterize the learning path and to distinguish convergence to Nash, convergence to non-Nash SCE, and convergence to fictitious-play history-dependent plateaus.

- For the five-good Economy D, Marimon et al. reported a limit point of learning, described as nearly fundamental with some speculative moves, without fully characterizing the associated equilibrium. We compute an SNE benchmark, diagnose why the baseline learner misses it, and show that a known-environment variant with greedy backup and bias-style centering converges to the SNE in every economy.

The three computational procedures applied in this paper can be aligned on a spectrum from pure equilibrium computation to models of learning. At one end sits the RVI solver. It iterates coupled best-response maps to a fixed point, discards intermediate steps, and reports only the limit. At the other end sits the tabular MuZero self-play algorithm. It simulates a population of agents who update empirical models from bilateral encounters and improve their policies by planning inside those models. Every iteration is a period of activity — agents act, observe, and update — and the entire path of population distributions is part of the object of study. The restricted-information SCE solver occupies an intermediate position: like the RVI solver it describes a fixed point, but it operates under the same information constraints as the learning agents and so it describes limit points that a learning process might approach.

A recurring structural feature of the computations is the role of *prompting* — the initial policy and beliefs with which each agent starts. Prompting operates in two directions. A *directional prompt* is used when multiple Nash equilibria coexist. In Economies A2 and B, the RVI solver and the tabular learner both reach the fundamental commodity-money branch from the uniform prior; only an initial policy that already prescribes the speculative offers leads to the speculative branch. A *misdirectional prompt* instead implants off-path beliefs that lead agents to a non-Nash self-confirming equilibrium that admits profitable deviations. Agents who believe certain offers will be refused never make those offers, never collect the data that would correct the belief, and settle into a convention consistent with their priors. Together,

directional and misdirectional prompts trace the boundary between outcomes that decentralized learning can reach and outcomes it cannot, illustrating how the same physical exchange environment can sustain qualitatively different monetary conventions depending on initial policies and beliefs.

A strand of the macroeconomic learning literature focuses on transient learning dynamics. For example, see Evans and Honkapohja [2001], Cho et al. [2002], and Sargent [2008]. A common design choice in that literature is to use a *constant-gain* algorithm that holds the learning rate fixed rather than driving it to zero. When used this way, constant-gain algorithms do not target exact convergence to a stationary Nash equilibrium: because recent observations always receive positive weight, beliefs typically keep moving and the economy tracks a non-stationary distribution. The resulting ergodic distribution is then not itself a Nash equilibrium; it describes agents with misspecified models, in the sense of Bray and Kreps [1987], who continue to adapt indefinitely.

Glossary of Main Notation

Table 3 collects the main objects used below and aligns them with the corresponding notation in Kiyotaki and Wright [1989] and Marimon et al. [1990]. A dash means that the object has no separate counterpart in that column.

Table 3: Glossary and source-paper notation crosswalk.

This paper	Description	Kiyotaki and Wright [1989]	Marimon et al. [1990]
Primitive exchange environment			
G	Finite set of storable goods.	Goods $1, 2, 3$; with fiat money, also good 0.	Goods $1, 2, 3$ in A1, A2, B; goods $0, 1, 2, 3$ in C; goods $1, \ldots, 5$ in D.
$I = \{1, \ldots, N\}$	Set of agent types; each type's label is also its consumption good.	Types $1, 2, 3$.	Types indexed by i; three types in A–C and five in D.
$N = \lvert I \rvert$	Number of types (distinct from the population size $\mathcal{N}$ below).	Fixed at 3.	Fixed at 3 or 5, depending on the economy.
ν_i	Population share of type i.	Equal type masses, $1/3$.	Finite population shares A_i/A.
i	Agent type and consumption good for that type.	Type i consumes good i.	Same consumption-specialization convention.
$f(i)$	Production map: good produced by type i after consuming.	Production good i^*.	$f(a)$ for an agent a, induced by the type's production pattern.
$s(h)$	One-period storage cost of holding good h.	c_{ij}, storage cost for type i holding good j.	$s(y)$ or the storage-cost vector in each simulated economy.
$\bar{u}_i$, $u_i(h)$	Consumption payoff parameter and utility from consuming holding h.	Corresponds most closely to the net consumption-production payoff $u_i = U_i - D_i$, after normalizing away explicit production disutility D_i.	$u_i(k) = 0$ for $k \neq i$ and $u_i(i) = u_i > 0$.
$\mathcal{A}$, a, $i(a)$	Population, individual agent, and type assignment.	Continuum of agents, not individually indexed.	Agent set $\mathscr{A} = \{1, \ldots, A\}$ and agent index a.
$p_t(a)$	Random bilateral matching map at date t.	Random pairwise matching.	Same notation $p_t(a)$.
$x_{a,t}$	Good held by agent a at the beginning of period t.	Inventory good j.	Same notation x_{at}.
$z_{a,t}$	Pre-trade state $(x_{a,t}, x_{p_t(a),t})$.	Own inventory j and partner inventory k.	Same notation $z_{at} = (x_{at}, x_{p_t(a),t})$.

This paper	Description	Kiyotaki and Wright [1989]	Marimon et al. [1990]
$\lambda_{a,t}$	Binary trade-offer action.	Pure trading strategy $T_i(j,k)$.	Trade decision λ_{at}.
$x^+_{a,t}$, X^+_t	Post-trade, pre-consumption holding (subscript-t form X^+_t used inside expectations).	Exit inventory after a match.	Post-trade holding x^*_{at}.
$\gamma_{a,t}$	Binary consumption action.	Consumption is automatic when the agent obtains good i.	Consumption decision y_{at}.
$Z_i = G \times G$, $H_i = G$	Pre-trade and post-trade state spaces for type i.	Not separately named.	Pre-trade state z_{at} and post-trade state x^*_{at}.
$\Lambda_i(k,j)$, $\Gamma_i(h)$	Feasible exchange and consumption action sets.	Binary trade choices; no separate consumption choice.	Binary trade and consumption classifier actions.
$r_i(h,\gamma)$	One-period reward after the trade stage and consumption decision.	Instantaneous utility net of production and storage costs.	External payoff $U_i(y^*)$ in the classifier accounting system.
Policies, beliefs, and distributions			
Σ_i, σ_i	Stationary Markov policy set and a type-i policy.	Strategy T_i.	Classifier rule lists used to make decisions.
$\alpha_i(k,j)$	Probability type i offers to trade own good k for partner good j.	Pure rule $\tau_i(j,k) \in \{0,1\}$, with j the own good and k the partner good (argument order reversed relative to this paper).	Exchange classifier's winning action; reported as exchange-strategy frequencies.
$\beta_i(h)$	Probability type i consumes after exiting trade with good h (not Kiyotaki–Wright's discount factor β).	–	Consumption classifier's winning action.
$\mu_i(k)$	Stationary probability type i starts a period holding good k.	p_{ij}, the share of type i agents holding good j.	$\pi^h_i(k)$ or empirical holding frequency $n^t_i(k)$.
$\mu(k)$	Aggregate stationary holding distribution.	Aggregate inventory distribution induced by p_{ij}.	$\pi^h(k)$ or aggregate empirical holdings.
$\eta_i(k)$	Conditional probability a holder of good k is type i.	Implicit in matching probabilities.	$\pi(i \mid k)$.
$Q(k,j)$	Objective reciprocal-trade probability for an offer $k \to j$.	Implicit in the equilibrium calculation from others' T rules and p_{ij}.	Empirical reciprocal-acceptance frequency in classifier play.
$\bar{r}_i$, $\bar{U}_i$	Expected flow reward and long-run average objective.	Discounted expected utility.	Classifier strengths are designed to approximate long-run average rewards.

This paper	Description	Kiyotaki and Wright [1989]	Marimon et al. [1990]
g_i, V_i, W_i	Average reward, pre-trade bias, and post-trade value.	Average-reward analogues of the discounted indirect utility $V_i(j) \equiv V_{ij}$; in Kiyotaki–Wright $V_i(j)$ is the discounted value of exiting trade with good j, whereas V_i here is the average-reward bias and W_i the post-trade value.	Reported value measures and classifier strengths, not a separate Bellman object.
k_i^{ref}	Reference good used to normalize the bias.	–	–
$F_i(k, j)$	Within-type flow of successful trades from k to j.	Implied by T_i and p_{ij}.	Implied by realized classifier trades.
μ_i^+, μ_i'	Post-trade and next-period type-i holding laws.	Implied inventory transition.	Implied by the law of motion for $x_{a,t+1}$.
F_σ	Law of motion for holding distributions under policy σ.	Steady-state condition for p.	Time-invariance of the probability collection Π_t.
SNE	Stationary Nash equilibrium: Bellman optimality plus stationarity.	Steady-state Nash equilibrium.	Stationary rational-expectations equilibrium target.
Restricted-information SCE objects			
$\mathcal{I}_{a,t}^e, \mathcal{I}_{a,t}^c$	Restricted-information exchange and consumption states.	–	Classifier inputs z_{at} and x_{at}^*.
$\theta_i = (m_i, q_i)$	Type-i subjective stationary environment.	–	Empirical environment learned by classifier experience, not named as θ_i.
$m_i(j)$	Subjective probability of meeting a partner holding good j.	–	Empirical partner-good distribution.
$q_i(k, j)$	Subjective probability an offer $k \to j$ is reciprocated.	–	Offer-success frequency inferred from realized proposals.
T_q	Subjective post-trade transition kernel.	–	Transition law implicit in the matched-agent law of motion.
$c_i(h, \gamma)$	Next holding after post-trade good h and consumption action γ. The symbol c_i is local to this paper and is unrelated to Kiyotaki–Wright's storage-cost notation c_{ij} or to the Marimon–McGrattan–Sargent consumption-classifier symbol c.	–	Law of motion to $f(a)$ if consuming, otherwise carrying the good.

This paper	Description	Kiyotaki and Wright [1989]	Marimon et al. [1990]
$P_i^{\tilde{\sigma}_i,\theta_i}$	Subjective Markov transition over holdings.	–	Transition frequencies generated by a classifier policy.
$R_i^{\tilde{\sigma}_i,\theta_i}$	Subjective one-period reward.	–	Realized external payoff averaged through classifier accounting.
g_i^θ, V_i^θ, W_i^θ	Subjective gain, bias, and post-trade value.	–	–
Q^σ	True reciprocal-trade probability induced by policy σ.	Implied by equilibrium strategies and inventory shares.	Induced by current classifier decisions and holdings.
SCE	Restricted-information stationary self-confirming equilibrium.	–	Related learning fixed point; not formalized as SCE.
O^m	On-path offered trade states at iteration m.	–	Visited state-action pairs in classifier play.
s^m, b^m, d^m	SCE subjective-optimality, belief, and stationarity residuals.	–	Convergence diagnostics, not with this notation.
Source-paper objects not separately used here			
–	Discount factor in the discounted Kiyotaki–Wright objective; this paper uses average reward instead.	β	–
–	Kiyotaki–Wright indicator variables for consumption, production, and storage in the period payoff.	$I_i^U(t)$, $I_i^D(t)$, $I_{ij}(t)$	–
–	Kiyotaki–Wright money-measure variables: stock, trades per period, velocity, and acceptability.	x_j, t_j, v_j, a_j	–
–	Classifier strings for exchange and consumption rules.	–	e, c
–	Classifier strengths.	–	$S_e^a(t)$, $S_c^a(t)$
–	Matched classifier sets and auction winners.	–	$M_e(z_{at})$, $M_c(z_{at})$, $e_a(z_{at})$, $c_a(z_{at})$
–	Classifier counters, bid indicators, and bid functions.	–	$\tau_e^a(t)$, $\tau_c^a(t)$, $I_e^a(t)$, $I_c^a(t)$, $b_1(e)$, $b_2(c)$
–	Fixed classifier set supporting an equilibrium.	–	D_a

References

Pierpaolo Battigalli. Comportamento razionale ed equilibrio nei giochi e nelle situazioni sociali. 1987.

Pierpaolo Battigalli. *Restrizioni Razionali su Sistemi di Probabilità Soggettive e Soluzioni di Giochi ad Informazione Completa*. EGEA, Milano, 1993.

Pierpaolo Battigalli and Danilo Guaitoli. Conjectural equilibria and rationalizability in a macroeconomic game with incomplete information. Quaderni di Ricerca 1988-6, I.E.P., Università Bocconi, 1988.

Michel Benaïm, Josef Hofbauer, and Ed Hopkins. Learning in games with unstable equilibria. *Journal of Economic Theory*, 144(4):1694–1709, 2009. doi: 10.1016/j.jet.2008.09.003.

Margaret M. Bray and David M. Kreps. Rational learning and rational expectations. In George R. Feiwel, editor, *Arrow and the Ascent of Modern Economic Theory*, pages 597–625. New York University Press, New York, 1987.

George W. Brown. Iterative solution of games by fictitious play. In Tjalling C. Koopmans, editor, *Activity Analysis of Production and Allocation*, Cowles Commission Monograph No. 13, pages 374–376. Wiley, New York, 1951.

Colin F. Camerer and Teck-Hua Ho. Experience-weighted attraction learning in normal form games. *Econometrica*, 67(4):827–874, 1999. doi: 10.1111/1468-0262.00054.

In-Koo Cho, Noah Williams, and Thomas J. Sargent. Escaping Nash inflation. *Review of Economic Studies*, 69(1):1–40, 2002.

George W. Evans and Seppo Honkapohja. *Learning and Expectations in Macroeconomics*. Princeton University Press, Princeton, NJ, 2001.

Drew Fudenberg and David M. Kreps. Learning mixed equilibria. *Games and Economic Behavior*, 5(3):320–367, 1993. doi: 10.1006/game.1993.1021.

Drew Fudenberg and David M. Kreps. Learning in extensive-form games I: Self-confirming equilibria. *Games and Economic Behavior*, 8(1):20–55, 1995. doi: 10.1016/S0899-8256(05)80016-8.

Drew Fudenberg and David K. Levine. Self-confirming equilibrium. *Econometrica*, 61(3):523–545, 1993. doi: 10.2307/2951716.

Drew Fudenberg and David K. Levine. *The Theory of Learning in Games*. MIT Press, Cambridge, MA, 1998.

David E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, 1989.

Josef Hofbauer and William H. Sandholm. On the global convergence of stochastic fictitious play. *Econometrica*, 70(6):2265–2294, 2002.

John H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, 1975.

John H. Holland. Escaping brittleness: The possibilities of general-purpose learning algorithms applied to parallel rule-based systems. In Ryszard S. Michalski, Jaime G. Carbonell, and Tom M. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach*, volume 2. Morgan Kaufmann, 1986.

Nobuhiro Kiyotaki and Randall Wright. On money as a medium of exchange. *Journal of Political Economy*, 97(4):927–954, 1989. doi: 10.1086/261634.

David M. Kreps. Anticipated utility and dynamic choice. In Donald P. Jacobs, Ehud Kalai, Morton I. Kamien, and Nancy L. Schwartz, editors, *Frontiers of Research in Economic Theory: The Nancy L. Schwartz Memorial Lectures, 1983–1997*, pages 242–274. Cambridge University Press, Cambridge, 1998.

Ramon Marimon, Ellen R. McGrattan, and Thomas J. Sargent. Money as a medium of exchange in an economy with artificially intelligent agents. *Journal of Economic Dynamics and Control*, 14(2):329–373, 1990. doi: 10.1016/0165-1889(90)90025-C.

Sean Meyn. *Control systems and reinforcement learning*. Cambridge University Press, 2022.

Julia Robinson. An iterative method of solving a game. *Annals of Mathematics*, 54 (2):296–301, 1951. doi: 10.2307/1969530.

Christopher D. Rosin. Multi-armed bandits with episode context. *Annals of Mathematics and Artificial Intelligence*, 61:203–230, 2011. doi: 10.1007/s10472-011-9258-6.

Thomas J. Sargent. Evolution and intelligent design. *American Economic Review*, 98 (1):5–37, 2008. doi: 10.1257/aer.98.1.5.

Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, Timothy Lillicrap, and David Silver. Mastering Atari, Go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020. doi: 10.1038/s41586-020-03051-4.

Lloyd S. Shapley. Some topics in two-person games. In Melvin Dresher, Lloyd S. Shapley, and Albert W. Tucker, editors, *Advances in Game Theory*, number 52 in Annals of Mathematics Studies, pages 1–28. Princeton University Press, Princeton, NJ, 1964.

David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016. doi: 10.1038/nature16961.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy Lillicrap, Karen Simonyan, and Demis Hassabis. A general reinforcement

learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419):1140–1144, 2018. doi: 10.1126/science.aar6404.