Figgie: Rules, Game Theory, Bayesian Model, and Portable Figgie Notation (PFN)

Spaghetti Software Inc*

February 15, 2025

Abstract

Figgie is a dynamic, incomplete-information card trading game in which players buy and sell suits over a short time period to compete for ownership of a "goal suit" that is only revealed at the end. This paper provides (i) an overview of the Figgie rules and mechanics, (ii) a description of its game-theoretic structure (including multi-round dynamics), (iii) a high-level Bayesian framework for modeling incomplete-information aspects, and (iv) a proposed text-based notation (*Portable Figgie Notation*, or PFN) that logs deck setup, deals, trading records, and end results—inspired by Portable Bridge Notation (PBN).

Contents

*Email: `info@softwarespaghetti.com`

1 Introduction

Figgie is a card game in which each player aims to maximize their final bankroll by collecting cards of the hidden "goal suit." Cards and suits are traded in real time, with partial information regarding which suit contains 12 cards (the "dummy" suit) and which same-color suit is actually valuable. In contrast to turn-based auctions, Figgie allows multiple players to place bids and offers simultaneously, with each trade resetting the quotes on all suits.

Although the trading period is often described as a single interval (e.g., four minutes), *each executed trade effectively begins a new subgame* in which players adjust their strategies and update their beliefs. Thus, the entire trading phase can be modeled as a *discrete* sequence of trade events, where real clock time continues to pass but the key strategic "ticks" happen whenever a trade occurs.

This paper is divided as follows:

- Section 2 reviews the basic rules of Figgie.
- Section 3 casts Figgie in a game-theoretic perspective, including a discussion of how to model the trading phase as a sequence of subgames.
- Section 4 outlines how a Bayesian model might be used to track hidden information and update beliefs over time (including how to specify the probability of observing each trade).
- Section 5 introduces a draft specification for a *Portable Figgie Notation* (PFN), loosely inspired by PBN in the game of Bridge.

2 Rules Overview

2.1 Deck Composition

- A Figgie deck consists of exactly 40 cards across four suits: spades (♠), clubs (♣), hearts (♡), and diamonds (♢).
- One suit has 12 cards, two suits have 10 cards each, and one suit has 8 cards.
- The identity of which suit is 12, 10, or 8 is unknown until the end.

2.2 Goal Suit and Color Matching

- The color of the 12-card suit (black or red) determines the color of the goal suit.
- The goal suit is the *other* suit of that same color and contains either 8 or 10 cards.

2.3 Setup and Starting Conditions

- Typically, there are N players (4 or 5 in standard Figgie), each starting with \$350 or similar.
- Players are dealt some subset of the 40 cards, giving them private information (how many of each suit they personally hold).

2.4 Real-Time Trading Phase

- There is no turn-taking. Players can post bids/offers or transact at any time during a short (e.g., 4-minute) trading period.
- A *trade* involves exactly one card of a chosen suit at a mutually agreed price.
- All outstanding quotes (bids/offers) *reset* to null immediately after each single-card trade.
- Players can continue until the timer runs out.

While this description suggests one continuous trading phase, it is helpful to think of it as a *sequence* of subgames, each triggered by a trade event. That is, each executed trade (a single-card transaction) not only updates money balances but also supplies new public information about who traded what and at what price. We can label these discrete events as "Trade 1," "Trade 2," and so on. In this sense, "trading time" advances in discrete steps whenever a trade occurs.

2.5 End of Round and Scoring

- Once time expires, the 12-card suit and corresponding goal suit are revealed.
- Each player's final holdings of the *goal suit* determine who gets the pot (any money pool to be distributed).
- If multiple players tie for the most goal-suit cards, the pot is split among them.

3 Game-Theoretic Model

3.1 Players, Suits, and Strategies

- Let $I = \{1, 2, \ldots, N\}$ be the set of players.
- The four suits are $S = \{\spadesuit, \clubsuit, \heartsuit, \diamondsuit\}$.
- Each player i may execute any trading action (bid, offer, buy, sell) at any time.

A strategy for each player depends on their private information (their initial hand) and the public history (past trades and quotes). Formally, if Θ_i is the set of private types for player i (i.e., possible card distributions in their hand), then:

$$\sigma_i : \Theta_i \times \mathcal{H}_t \to A_i,$$

where $\mathcal{H}_t$ is the observed history (public) at time t, and A_i is the set of possible actions (bidding, offering, buying, selling, or passing).

However, in order to capture *multi-round* structure, we can label each discrete trade as a "subgame." If trade event $t-1$ has just occurred, the subgame at time t is characterized by the updated money balances, card holdings, and any inferred information. We can thus say

$$\sigma_i(\theta_i, \mathcal{H}_t)$$

provides player i's action in subgame t. Once a single card is traded, subgame t ends, and the game transitions to subgame $t+1$. At the end of the total allowed clock time (e.g., after 4 minutes or when no more trades happen), the 12-card suit is revealed and payoffs are realized.

3.2 Hidden Distribution and Goal Suit

- One suit has 12 cards, two suits have 10 each, and one suit has 8.
- If the 12-card suit is black, the other black suit is the goal; if it is red, the other red suit is the goal.

Thus the total possible $\delta \in \mathcal{D}$ (assignments of $\{12, 10, 10, 8\}$ to suits, plus the color constraint) represent a space of *hidden states*.

3.3 Payoffs

Each player seeks to maximize final bankroll, which is:

$$u_i = B_i(T) + (\text{share of pot if the player holds the most goal-suit cards}).$$

When multiple players tie, they split the pot equally.

3.4 Multi-Round / Subgame-Perfect Reasoning

In principle, one could explore a (Perfect) Bayesian Equilibrium (PBE) or Subgame Perfect Equilibrium (SPE) approach where the game is "unwrapped" into a series of discrete subgames. Each single-card trade forms a boundary between consecutive subgames, with updated public information. At each subgame node, players use strategies that (1) incorporate their beliefs about the likely 12-card (and thus goal) suit, and (2) optimize given the finite time horizon (e.g., 4 minutes) which limits how many total trades can occur.

Analyzing such an equilibrium typically involves:

- *Backward induction (if the horizon is finite)*: The final "subgame" (the last possible trade before time expires) is easier to analyze because there is minimal future to consider. If acquiring or dumping cards changes your expected payoff, you do so; otherwise, you hold.
- *Belief updating*: After each trade, players infer something about *why* that trade took place at that price. This can shift posterior probabilities over which suit is 12 and which is the goal suit.
- *Potential multiple equilibria*: Real-time trading with incomplete information often admits multiple equilibria (e.g., "fast trade" vs. "wait-and-see" behaviors).

While a full solution for all subgames may be complex, the key insight is that each trade redefines the public state. Thus, from a theoretical standpoint, the single 4-minute phase can be decomposed into a *sequence* of decision nodes ("Trade 1," "Trade 2," and so on). This perspective aligns well with how players actually adapt their strategies in real time.

3.5 What if All Players Follow an Optimal Bayesian-Nash Strategy?

Even though one can, in principle, define a perfect or optimal Bayesian-Nash equilibrium (BNE) for Figgie, characterizing it in closed form is extraordinarily challenging due to the game's high-dimensional state space and the real-time, multi-player environment. Still, we can discuss in broad terms what would happen in such an equilibrium:

- **Prices Reflect Updated Beliefs.** At any moment, each player's posterior over which suit is the 12-card suit (and thus which other suit is the goal) would be fully rational, incorporating:

  1. Their private cards,
  2. Observed trade history, and
  3. Common knowledge that all players are rational and updating in the same way.

  As a result, any bid or offer in an equilibrium would be set at a price that fairly reflects the expected final payoff of holding that suit, given the current posterior probabilities.

- **Trades Convey Information.** In many incomplete-information games, a single trade (or even a posted quote) reveals players' beliefs. A player who aggressively buys spades at a high price might signal a strong belief that spades is the goal suit. In an optimal BNE, players factor this signaling effect into their decisions.

- **Potential for Multiple Equilibria (Including No-Trade Scenarios).** A core result in some incomplete-information market models is the *no-trade theorem*: fully rational, risk-neutral players with common priors might not trade at all because an attempt to buy or sell can update the counterparty's beliefs in a way that negates any perceived profit. In Figgie, there may indeed be equilibria with little or no trading. On the other hand, the deck composition (one suit having 12 cards, etc.) and the payoff structure (trying to collect the goal suit) can also support equilibria where significant trading occurs if players need to rebalance suits. Which outcome obtains depends on precise priors, initial deals, and the exact logic of the subgame-perfect strategies.

  Case: No Trades Occur. In some Figgie sessions, it is entirely possible that no one executes a trade throughout the entire (e.g., 4-minute) window. From a game-theoretic perspective, this can arise if all players fear revealing information through bids or sales. Because any move may signal private beliefs or holdings to opponents—and thus allow the opponents to adjust their own strategies—*fully rational* players might conclude that the risks of "giving away" information outweigh the potential benefits of acquiring or offloading goal-suit cards. If no trades occur, each player simply keeps their initial hand, the 12-card suit is revealed at the end, and final payoffs are determined accordingly. In extreme cases of conservative or risk-averse equilibrium strategies, the game can thus end exactly where it began, with no price discovery.

- **Unraveling vs. Partial Information.** In a finite-horizon Bayesian game, backward induction can cause *unraveling*, where every move early on reveals enough information that eventually all players infer the true goal suit. If that happens, remaining trades would simply be re-allocative, matching cards to the players who value them most. Alternatively, if the game is short or if equilibrium behavior steers players away from revealing crucial signals, the game can end with persistent uncertainty about the goal suit.

- **Computational Complexity.** Solving for a BNE typically requires enumerating vast game trees. Each single-card trade is a state transition, and each state includes beliefs about the hidden suit distribution, plus money/card holdings. For a 40-card deck and multiple players, this space becomes huge. Hence, while the game *does* have an equilibrium in theoretical terms, finding it exactly is usually intractable in practice.

Key Takeaway. If players are indeed fully rational and follow an optimal BNE, any trades (and the prices of those trades) will *correctly reflect* the underlying probabilities of each suit being the goal. However, depending on how the equilibrium "shakes out," the game might end with very few trades (as per some no-trade arguments) or conversely might involve substantial "unraveling" of information if that is the mutually preferred equilibrium path. *Crucially, in certain equilibria,* no *trades might ever occur, leaving all players with their initial hands and forcing the game to resolve purely on initial card distributions.* The possible presence of *multiple equilibria* means there is no single inevitable trading trajectory.

4 Bayesian Model

4.1 State Space and Types

- Ω is the sample space of all possible ways the deck might be distributed:

$$\omega \in \Omega \implies \omega = (\delta, X, Y, H)$$

  where δ is the suit size assignment, X is the 12-card suit, Y is the goal suit, and H is the dealing of the 40 cards among the N players.

- Player i's *type* $\theta_i = H_i$, their private hand.

4.2 Posterior Updating

Players update their beliefs $\Pr(\omega \mid H_i, \mathcal{H}_t)$ upon seeing new trades or price movements. If spades suddenly see high demand, that suggests many players might believe spades is the goal suit, or at least that black is the favored color.

Formally, after observing a new trade event e_k:

$$\Pr\bigl(\omega \bigm| \theta_i,\ e_1,\dots,e_k\bigr) \;=\; \frac{\Pr\bigl(\omega \bigm| \theta_i,\ e_1,\dots,e_{k-1}\bigr) \;\times\; P\bigl(e_k \bigm| \omega,\ e_1,\dots,e_{k-1}\bigr)}{\displaystyle\sum_{\omega' \in \Omega_i} \Pr\bigl(\omega' \bigm| \theta_i,\ e_1,\dots,e_{k-1}\bigr) \times P\bigl(e_k \bigm| \omega',\ e_1,\dots,e_{k-1}\bigr)},$$

where $P(e_k \mid \omega)$ is the *likelihood* of observing that trade event if the true state is ω.

4.3 Expected Value of a Suit

At any time, a player estimates the expected value of holding one additional card of suit s by summing over the relevant posterior probabilities:

$$\mathrm{EV}(\text{card of suit } s) = \sum_{\omega} \Pr(\omega \mid H_i, \mathcal{H}_t) \times V_i(s \mid \omega),$$

where $V_i(s \mid \omega)$ is the marginal utility of having that suit card in state ω.

Note: Each trade event t updates $\mathcal{H}_t$, so a multi-round Bayesian model breaks down the entire trading window into a sequence of smaller Bayesian updates. Over real clock time, many bids/offers might be posted, but once a single trade is executed, we treat that as the "observation" that triggers belief revision.

4.4 Specifying the Probability of an Observed Trade via a Behavioral-Finance Microstructure Model

A key challenge is defining the likelihood term $P(e_k \mid \omega)$, i.e. the probability of observing a specific trade (who traded which suit/card at what price) given the hypothesized state ω. We can borrow ideas from standard microstructure and behavioral finance models, where each player's *valuation* for a card depends on its expected payoff under ω, but trades also have a stochastic / logistic element to capture noise, risk preferences, or strategic uncertainty.

- **Valuation.** Let $v_i(s \mid \omega)$ be the marginal utility (valuation) that player i assigns to one additional card of suit s under state ω. For example, if ω indicates that suit s is likely the goal suit, then $v_i(s \mid \omega)$ is high.

- **Willingness to Buy/Sell.** In a standard microstructure model, a *trade* between buyer i and seller j at price p occurs if buyer's net surplus $v_i(s \mid \omega) - p$ and seller's net surplus $p - v_j(s \mid \omega)$ are both positive enough to motivate the transaction.

- **Logistic Acceptance.** Instead of a strict threshold, we add a logistic function to capture bounded rationality or behavioral noise:

$$P_i^{(\text{buy})}(p \mid \omega) \;=\; \sigma\Big(\alpha\left[v_i(s \mid \omega) - p\right]\Big), \quad P_j^{(\text{sell})}(p \mid \omega) \;=\; \sigma\Big(\beta\left[p - v_j(s \mid \omega)\right]\Big),$$

  where $\sigma(x) = \frac{1}{1+e^{-x}}$ and $\alpha, \beta > 0$ are sensitivity parameters (larger = more "rational").

- **Matching Probability.** Let γ_{ij} denote the probability (per unit time, or per "encounter") that buyer i and seller j actually meet or that the buyer's posted bid and seller's posted ask line up.

- **Trade Likelihood.** If the observed trade event e_k says "Player i buys a card of suit s from player j at price p\$," then under state ω:

$$P(e_k \mid \omega) \;=\; \gamma_{ij} \;\times\; P_i^{(\text{buy})}(p \mid \omega) \;\times\; P_j^{(\text{sell})}(p \mid \omega).$$

  This yields a *numerical* probability of that exact trade happening, given ω. A higher valuation gap between i and j (i.e. $v_i(s) \gg v_j(s)$) makes the trade more probable at any feasible price p.

Hence, to implement a Bayesian update, one inserts $P(e_k \mid \omega)$ from this microstructure-based formula into Bayes' rule. As trades occur, each player reweights their candidate states ω by how likely that trade was, given ω. This approach—especially if combined with a particle filter or importance sampling—provides a workable, real-time way to model *why* a particular trade took place.

5 Portable Figgie Notation (PFN)

Inspired by *Portable Bridge Notation (PBN)*, we introduce an informal *Portable Figgie Notation (PFN)* that allows recording of:

- **Deck setup**: which suit has 12 cards, which have 10 or 8, and which is the goal suit.
- **Initial deal**: who started with which cards.
- **Trade history**: each card traded, the buyer, the seller, and the price.
- **Result**: final bankrolls, the revealed 12-card suit, the actual goal suit, and who won or split the pot.

Optionally, one may also label trades by discrete "trade index" (i.e., the kth trade in the sequence), as well as real-time timestamps. This is consistent with the notion that each trade forms a boundary for a new subgame.

5.1 General Structure

A PFN file can be split into key sections, each delineated by square brackets (similar to PBN):

- `[FiggieGame]`: Overall metadata (title, game ID, date, number of players).
- `[DeckSetup]`: Suit distribution, the color of the 12-card suit, and the goal suit.
- `[Deal]`: Which cards each player initially holds.
- `[Trades]` (optional): Chronological record of single-card trades.
- `[Result]`: Final outcome after time expires.

5.2 Section-by-Section Description

`[FiggieGame]` Section Contains basic metadata:

```
[FiggieGame]
Title: Example Round
GameID: 2025-02-15-1
Players: 4
Date: 2025-02-15
```

`[DeckSetup]` Section Specifies which suit is 12, which is 8 or 10, and the color logic:

```
[DeckSetup]
Distribution: S=10 C=12 H=10 D=8
GoalSuitColor: Black
GoalSuit: Spades
```

Optionally, you might add a `DeckOrder` line listing each card in some fixed ordering if you want full reproducibility.

[Deal] Section Lists each player's *initial* cards:

```
[Deal]
P1: S1,S2,S3,C2,C7,H5,H6,H7,D8
P2: S4,S5,S6,C1,C8,H3,H9,D2
P3: S9,S10,H1,H2,H4,D3,D4,D5
P4: C3,C4,C5,C6,H8,H10,D6,D7
```

Here, S1, S2, . . . are labels for spade cards; the actual rank is irrelevant in Figgie, so indexing them is sufficient.

[Trades] Section (optional) Records each single-card transaction:

```
[Trades]
1: T=15.3 Buyer=P2 Seller=P1 Suit=Spades Card=S2 Price=20
2: T=33.0 Buyer=P3 Seller=P2 Suit=Diamonds Card=D2 Price=5
3: T=45.5 Buyer=P1 Seller=P4 Suit=Hearts   Card=H8 Price=10
```

- T= indicates the timestamp (e.g., seconds into the trading round).
- Buyer / Seller indicates which players transacted.
- Suit and Card show what was traded.
- Price is the agreed payment.

In addition, the leftmost index (1:, 2:, 3:, etc.) can be viewed as a *discrete subgame index*. This clarifies the order of trades in "trading time."

[Result] Section Specifies the end-of-round details:

```
[Result]
Revealed12CardSuit: Clubs
GoalSuit: Spades
P1_FinalBank: 420
P2_FinalBank: 355
P3_FinalBank: 335
P4_FinalBank: 290
Winners: P1
```

This records the final 12-card suit, the real goal suit, each player's final bankroll, and the winner(s) if there is a tie.

5.3 Example PFN File

A complete example might look like this:

```
[FiggieGame]
Title: Figgie Demo Round
GameID: 001
Players: 4
```

```
Date: 2025-02-15

[DeckSetup]
Distribution: S=10 C=12 H=10 D=8
GoalSuitColor: Black
GoalSuit: Spades

[Deal]
P1: S1,S2,S3,C2,C7,H5,H6,H7,D8
P2: S4,S5,S6,C1,C8,H3,H9,D2
P3: S9,S10,H1,H2,H4,D3,D4,D5
P4: C3,C4,C5,C6,H8,H10,D6,D7

[Trades]
1: T=15.3 Buyer=P2 Seller=P1 Suit=Spades   Card=S2 Price=20
2: T=33.0 Buyer=P3 Seller=P2 Suit=Diamonds Card=D2 Price=5
3: T=45.5 Buyer=P1 Seller=P4 Suit=Hearts   Card=H8 Price=10

[Result]
Revealed12CardSuit: Clubs
GoalSuit: Spades
P1_FinalBank: 420
P2_FinalBank: 355
P3_FinalBank: 335
P4_FinalBank: 290
Winners: P1
```

5.4 Potential Extensions

- **Pre-Trade Quotes**: If desired, one could log every posted bid or offer. However, in a fast-paced environment, this can become large very quickly.
- **Hidden/Partial Information**: If the file is meant to be distributed in real time, suits might be withheld or masked until the end.
- **Multiple Rounds or Multiple Deals**: Multiple Figgie rounds can be concatenated in a single PFN file, with each round starting a new `[FiggieGame]` block.
- **Subgame Labeling**: To emphasize the discrete structure, one might group trades by "Round 1," "Round 2," etc., if desired. This could help in analyzing multi-round equilibrium behavior.

6 Conclusion

Figgie blends elements of incomplete information, real-time negotiation, and the potential for strategic bluffing. We provided a game-theoretic framing and a Bayesian model showing how players can update beliefs about the suit configuration based on private and public signals.

A key insight is that the seemingly continuous trading phase can be decomposed into a sequence of discrete subgames, each triggered by a single trade event. This multi-round perspective aligns with Perfect Bayesian Equilibrium notions: strategies may differ at each trade node, and beliefs about the valuable suit evolve with each observed transaction.

In an *optimal Bayesian-Nash equilibrium*, prices (if trading occurs) would fully incorporate players' beliefs about the goal suit. However, depending on the structure of the equilibrium (and the well-known possibility of no-trade theorems in incomplete-information markets), actual trades could range from "complete information unraveling" to "very few or no trades." In fact, entire rounds may finish with *no* transactions if it is an equilibrium for every player to avoid revealing private information. The presence of multiple equilibria means there is no single inevitable path.

Finally, we introduced *Portable Figgie Notation* (PFN) to standardize how rounds are recorded for future study or replay. This lays groundwork for future development of software tools that parse PFN files to automatically reconstruct and analyze each round, enabling deeper research into *Figgie* strategies, AI-driven trading bots, and large-scale data collection for empirical game analysis.

Future directions include developing explicit multi-round equilibrium models (though they may be computationally complex), conducting experiments to see how humans or bots actually trade in real time, refining PFN to capture more nuanced signaling or partial information updates between trades, and exploring how no-trade theorems or high-information "unraveling" might emerge in practice.

Acknowledgments. Thanks to the wider card-game community for inspiring these ideas, and to the creators of Bridge PBN for setting a foundation in standardized notation.

References

[1] Figgie Card Game (n.d.).
`https://www.figgie.com`

[2] Open-source PBN format documentation (n.d.).
`https://www.bridgebase.com/help/5PBN.pdf`

[3] Author, Q. (2025). *Figgie and Speculative Markets*. Journal of Experimental Card Games, 2(3): 45–78.

[4] Milgrom, P. & Stokey, N. (1982). Information, trade and common knowledge. *Journal of Economic Theory*, 26(1): 17–27.