

KangarooDeFi: AMM-Native Prediction Markets for Real-Time Conviction Pricing

KangarooDeFi Research Team (support@kangaroodefi.com)

Abstract—Prediction markets are designed to aggregate distributed information by allowing participants to express beliefs about future outcomes through economic commitments. However, many existing implementations remain limited by either orderbook fragmentation or static pooled-betting structures. Orderbooks require active counterparties and sufficient depth, while pooled systems often fail to generate continuous price discovery during the market lifecycle. This paper introduces KangarooDeFi, an automated-market-maker-native framework for short-duration prediction rounds. For each round, the protocol creates two ERC20 outcome tokens representing directional exposure, UP and , and initializes two AMM pairs against the protocol wrapped-collateral token, kwBNB. The router mints matched initial kwBNB and outcome-token balances into the pairs, calls `sync()`, and opens a buy-side conviction market in which participant demand changes AMM reserve ratios and therefore visible sentiment indicators. The current implementation prevents direct sell-back of outcome tokens into the same round pair, so the system is more accurately described as a buy-side AMM conviction market with settlement-based exit rather than a fully symmetric secondary market. The paper formalizes the round lifecycle, liquidity baseline, price impact, payout allocation, expected value, oracle error, and manipulation risks of this model. The results suggest that KangarooDeFi represents a DeFi-native market primitive in which belief is tokenized, priced through AMM mechanics, and resolved through deterministic round settlement.

Keywords—automated market maker, decentralized finance, kwBNB, prediction markets, smart contracts.

I. INTRODUCTION

Prediction markets have long been recognized as mechanisms for aggregating dispersed information. By allowing participants to attach economic value to future outcomes, these systems can produce collective signals that may be more responsive than surveys or informal opinion aggregation. In practice, however, many prediction-market implementations struggle to deliver continuous liquidity, clear pricing, and repeat engagement.

Orderbook-based prediction markets require active counterparties. When depth is limited, spreads widen and prices become unreliable. Pooled betting systems avoid some orderbook problems by aggregating capital into outcome pools, but they often do not produce meaningful price movement while the market is active. They record commitment, but they do not continuously price conviction.

Decentralized finance has already solved part of this problem through automated market makers (AMMs). AMMs replace order matching with deterministic pricing functions and reserve-based liquidity. KangarooDeFi applies this infrastructure to prediction markets by creating round-specific outcome tokens and AMM pools for each directional outcome.

The objective is not simply to let users bet on UP or DOWN; it is to transform directional belief into a priced ERC20 position. The current system is intentionally round based. Each fifteen-minute market operates as a self-contained micro-market with its own outcome tokens, initial liquidity baseline, trading phase, lock phase, and settlement phase. This design allows the protocol to isolate risk, compute round-specific capital flows, and display real-time sentiment derived from AMM reserves.

II. RELATED WORK

D. Prediction Market Theory

Prediction markets have been studied as information-aggregation mechanisms. Hanson introduced combinatorial information markets, while later work explored market scoring rules, incentive compatibility, and decentralized oracle systems. These models often assume sufficient liquidity and rational participation. Real deployments frequently fail because those assumptions do not hold in early or niche markets.

B. Orderbook-Based Systems

Orderbook markets are efficient when there are many active participants on both sides. In prediction markets, many events have sparse participation. This produces thin books, wide spreads, and poor execution quality. For short-duration markets, these issues become more severe because liquidity must arrive quickly.

C. AMMs and DeFi Liquidity

Automated market makers replace centralized order matching with deterministic reserve functions. The most common invariant is the constant-product model:

$$x \cdot y = k.$$

Here, x and y are the reserves of the two assets in the pool, and k is the invariant. This model provides continuous buy-side availability as long as reserves exist. Angeris et al. show that constant-function market makers can provide predictable price impact under reserve-based constraints.

D. Existing Hybrid Platforms

Hybrid prediction platforms improve user experience but often depend on off-chain infrastructure, fragmented liquidity, or market-specific liquidity providers. KangarooDeFi differs by creating outcome tokens and protocol-initialized AMM liquidity for every round.

III. SYSTEM DESIGN

A. Round-Based Market Structure

Each market is defined over a time interval $[t_0, t_T]$, during which participants acquire directional exposure on an underlying asset A , such as BTC/USDT. A round can be represented as:

$$r = (A, t_0, t_T, P_0, P_T),$$

where r denotes the prediction round, A is the underlying asset, t_0 is the opening timestamp, t_T is the closing timestamp, P_0 is the opening oracle price, and P_T is the closing oracle price.

For each round r , the protocol generates two ERC20 outcome tokens:

$$UP_r, \text{ DOWN}_r.$$

The UP_r token represents the outcome in which the closing price is greater than or equal to the opening price. The $DOWN_r$ token represents the outcome in which the closing price is lower than the opening price. These tokens do not directly track price movement. They encode settlement rights conditional on the round outcome.

B. Liquidity Pair Formation

The router creates two AMM pairs through a V2-style factory:

$$\text{Pair}_{UP} = (UP_r, \text{kWB NB}) \\ \text{Pair}_{DOWN} = (DOWN_r, \text{kWB NB})$$

The router initializes both pairs by assigning outcome-token supply and minting matching kWB NB collateral into each pair.

For each pool $i \in \{UP, DOWN\}$, the initial reserves are:

$$R_{i,0}^{\text{token}} = S_r, \quad R_{i,0}^{\text{kWB NB}} = S_r.$$

Therefore, the initial invariant is:

$$k_i = R_{i,0}^{\text{token}} R_{i,0}^{\text{kWB NB}} = S_r^2.$$

This is the key implementation-level difference from a normal external-LP workflow. The round does not wait for third-party liquidity providers. It begins with protocol-initialized liquidity denominated in kWB NB and outcome tokens. After the balances are assigned, the router calls `sync()` so the AMM reserves reflect the balances held at the pair addresses.

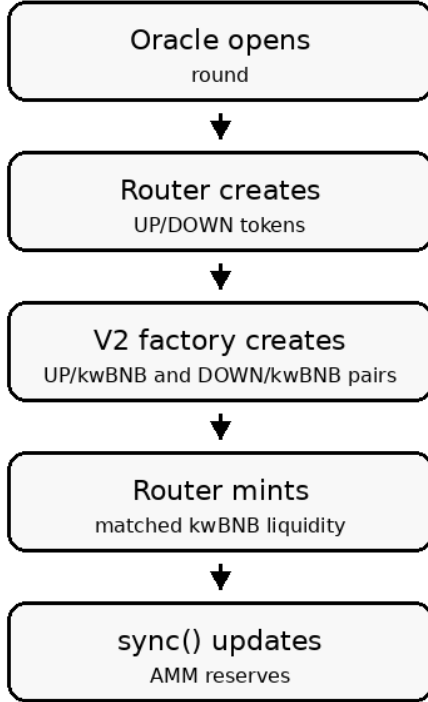


Fig. 1 Router-based kWB NB liquidity lifecycle for one prediction round.

C. Trade Execution and Price Impact

Let x_i denote the kWB NB reserve and y_i denote the outcome-token reserve of pool i . During the active phase, a participant

buys outcome token i by depositing kWB NB into the corresponding pool. The pre-trade marginal price is:

$$P_i = \frac{x_i}{y_i}.$$

If a participant contributes Δx kWB NB and the AMM fee rate is f , the effective input is:

$$\Delta x_{eff} = \Delta x(1 - f).$$

The post-trade kWB NB reserve becomes:

$$x_i' = x_i + \Delta x_{eff}.$$

Under the constant-product invariant, the post-trade token reserve is:

$$y_i' = \frac{x_i y_i}{x_i + \Delta x_{eff}}.$$

The number of outcome tokens received by the participant is therefore:

$$\Delta y_i = y_i - y_i' = y_i - \frac{x_i y_i}{x_i + \Delta x_{eff}}.$$

The post-trade marginal price is:

$$P_i' = \frac{x_i + \Delta x_{eff}}{y_i'} = \frac{(x_i + \Delta x_{eff})^2}{x_i y_i}.$$

Thus, the relative price impact is:

$$\frac{P_i'}{P_i} = \left(1 + \frac{\Delta x_{eff}}{x_i}\right)^2.$$

This formula shows that price impact is nonlinear. Larger inputs express stronger conviction because they move the displayed price more aggressively.

The current implementation intentionally prevents selling outcome tokens back into the same round pair. Transfers to the round pair are rejected through token-transfer rules. Therefore, KangarooDeFi should be described as a buy-side AMM conviction market: users can acquire directional exposure during the active phase, while exit and recovery occur through settlement rather than direct sell-back into the same AMM pair.

D. Interpretation of Price

The AMM price of each outcome token is derived from its reserve ratio:

$$P_i = \frac{R_i^{\text{kWB NB}}}{R_i^{\text{token}}}, \quad i \in \{UP, DOWN\}.$$

A normalized sentiment indicator can be computed as:

$$S_{UP} = \frac{P_{UP}}{P_{UP} + P_{DOWN}}, \quad S_{DOWN} = \frac{P_{DOWN}}{P_{UP} + P_{DOWN}}.$$

These values should be interpreted as AMM-implied sentiment indicators rather than risk-neutral probabilities. Because the market is buy-side during the active phase, displayed sentiment is shaped by capital inflow and relative reserve movement.

E. Market Lifecycle

Each round follows a deterministic lifecycle: initialization, outcome-token creation, pair creation, liquidity bootstrapping, trading, locking, resolution, and settlement. Initialization records P_0 and the round parameters. Token creation deploys or registers UP_r and $DOWN_r$. Pair creation connects those tokens to kWB NB. Liquidity bootstrapping establishes the baseline S_r on both sides. Trading lets users acquire directional tokens. Locking restricts further activity. Resolution compares P_T with P_0 . Settlement distributes committed collateral according to the configured reward logic.

IV. MATHEMATICAL MODEL

A. Outcome Function

The winning outcome W_r is determined by comparing the closing oracle price P_T with the opening oracle price P_0 :

$$W_r = \begin{cases} \text{UP}, & \text{if } P_T \geq P_0, \\ \text{DOWN}, & \text{if } P_T < P_0. \end{cases}$$

This formulation treats a non-decrease as an UP outcome, consistent with the token definition.

B. Liquidity Aggregation

Because the router mints the initial kwBNB liquidity itself, committed capital should be measured as excess collateral above the baseline S_r . Let $B_i^{\text{kwBNB}}(T)$ denote the final kwBNB balance of pair i at the end of the trading phase. The user-committed liquidity is:

$$L_i = \max(B_i^{\text{kwBNB}}(T) - S_r, 0), \quad i \in \{\text{UP}, \text{DOWN}\}.$$

The total committed liquidity is:

$$L_{\text{total}} = L_{\text{UP}} + L_{\text{DOWN}}.$$

This formulation avoids treating the initial protocol-minted baseline as user capital. It also guards against negative committed-liquidity values in edge cases.

C. Reward Allocation

The committed liquidity is distributed across configured economic components. Let α denote the winner share, β the sponsor share, γ the loser recovery or airdrop share, and δ the protocol-fee share:

$$\begin{aligned} W_{\text{pool}} &= \alpha L_{\text{total}}, \\ S_{\text{pool}} &= \beta L_{\text{total}}, \\ R_{\text{loss}} &= \gamma L_{\text{total}}, \\ F_{\text{protocol}} &= \delta L_{\text{total}}, \end{aligned}$$

with the constraint:

$$\alpha + \beta + \gamma + \delta = 1, \quad \alpha, \beta, \gamma, \delta \geq 0.$$

This structure formalizes the multi-layer incentive design of KangarooDeFi. Winners receive the primary reward pool, sponsors receive a configured flow share, losing participants may receive partial recovery or airdrop value, and the protocol receives a fee component.

D. Proportional Payout

Let T_W denote the aggregate eligible supply of the winning outcome token, and let $b_{u,W}$ denote user u 's winning-token balance. The user's proportional payout is:

$$\text{Payout}_u = \frac{b_{u,W}}{T_W} W_{\text{pool}}.$$

The realized profit and return on investment are:

$$\begin{aligned} \text{Profit}_u &= \text{Payout}_u - \text{Cost}_u, \\ \text{ROI}_u &= \frac{\text{Profit}_u}{\text{Cost}_u}. \end{aligned}$$

Here, Cost_u is the amount paid by the user to acquire the position, including the effective AMM entry cost.

E. Expected Value Analysis

For a participant entering outcome i , let q_i denote the participant's subjective probability that outcome i wins. If Recovery_u denotes the residual value received when the position is incorrect, expected value is:

$$EV_u = q_i \cdot \text{Payout}_u + (1 - q_i) \cdot \text{Recovery}_u - \text{Cost}_u.$$

A rational participant enters when:

$$EV_u > 0.$$

This framing makes the system analyzable as an expected-value decision rather than a purely static bet. The existence of a

recovery component reduces downside severity and can increase repeat participation.

V. INCENTIVE MECHANISMS

A. Winner Rewards

Winner rewards provide the core upside for correct directional prediction. The reward pool is distributed proportionally to eligible winning-token balances. This encourages early and confident entry when participants believe the displayed sentiment underestimates their subjective probability.

B. Sponsor Layer

Each round includes a sponsorship layer. Sponsors commit capital for visibility and receive a configured share of round flow. The sponsor allocation is represented by $S_{\text{pool}} = \beta L_{\text{total}}$. In the interface, this creates a visible economic role separate from directional trading.

C. Dynamic Sponsorship Pricing

Sponsorship can be made replaceable through a price-escalation rule. If the current sponsor paid C_s , the next sponsor may be required to pay:

$$C_{s+1} = \lambda C_s, \quad \lambda > 1.$$

Part of the replacement payment may be refunded to the previous sponsor, while the remaining amount contributes to protocol revenue or round economics. This creates competition for visibility and can transform sponsorship into a market-driven attention layer.

D. Loss Recovery Mechanism

Unlike purely zero-sum systems, KangarooDeFi can allocate a partial recovery component to losing participants. This component may be distributed through claimable rewards, airdrop tokens, or other settlement-linked mechanisms. The recovery layer reduces the psychological barrier to participation and supports user retention.

VI. GAME-THEORETIC DYNAMICS

A. Rational Traders

A rational trader participates when the subjective expected value of the position is positive:

$$q_i \cdot \text{Payout}_u + (1 - q_i) \cdot \text{Recovery}_u - \text{Cost}_u > 0.$$

This condition depends on the user's belief, entry cost, expected payout, and recovery assumptions.

B. Momentum Effects

Because purchases increase the kwBNB reserve and reduce the outcome-token reserve, buy pressure mechanically increases the marginal price:

$$\Delta x_i > 0 \Rightarrow P_i' > P_i.$$

Since the sentiment indicator is derived from relative prices, an increase in one outcome price also increases its displayed sentiment:

$$P_i' > P_i \Rightarrow S_i' > S_i.$$

This can create momentum effects in which visible demand attracts further demand.

C. Contrarian Strategies

When AMM-implied sentiment exceeds a trader's subjective probability, the outcome may be considered overpriced from that trader's perspective:

$$S_i > q_i.$$

When sentiment is below subjective probability, the outcome may be considered underpriced:

$$S_i < q_i.$$

Because direct sell-back into the same round pair is restricted, correction is not classical two-sided arbitrage. Instead, correction occurs mainly through asymmetric buy-side pressure toward the relatively underpriced outcome.

D. Market Efficiency

Let $S_i(t)$ denote the AMM-implied sentiment for outcome i at time t , and let $q_i(t)$ denote the representative subjective probability assigned by informed participants. Under sufficient liquidity, heterogeneous beliefs, and active participation, the system may tend toward:

$$S_i(t) \rightarrow q_i(t).$$

This convergence is not guaranteed. Early rounds, low liquidity, concentrated capital, or informational asymmetry can prevent efficient pricing. Nevertheless, AMM-based reserve movement provides a stronger continuous signal than static pooled betting.

VII. RISK MODEL

A. Smart Contract Risk

The system is exposed to smart contract risk, including reentrancy, incorrect state transitions, flawed permissioning, and arithmetic or precision errors. Mitigation requires modular contract design, extensive testing, and independent audit review.

B. Oracle Risk

Outcome determination depends on oracle-reported prices. Let P_t^{oracle} denote the oracle price and P_t^{true} denote the true market reference price. The normalized oracle error is:

$$\epsilon_t = \frac{P_t^{oracle} - P_t^{true}}{P_t^{true}}.$$

Even small deviations may affect binary settlement when P_T is close to P_0 . Mitigation can include multi-source aggregation, time-weighted references, and delayed finalization windows.

C. Liquidity Calibration Risk

The initial liquidity baseline S_r must be calibrated carefully. If S_r is too small, early trades can move prices aggressively and create noisy sentiment. If S_r is too large, user trades may not move prices enough to reveal conviction. The relevant ratio is trade size to baseline reserve:

$$\rho = \frac{\Delta x_{eff}}{S_r}.$$

Higher ρ implies greater price impact and manipulation sensitivity.

D. Manipulation Risk

A large input can move the marginal price nonlinearly:

$$\frac{P_i'}{P_i} = \left(1 + \frac{\Delta x_{eff}}{x_i}\right)^2.$$

Manipulation risk is highest when the reserve x_i is small relative to the trade size. Deeper initial liquidity, participation diversity, and display of both volume and sentiment can reduce the effectiveness of manipulation.

E. Operational Risk

Early-stage deployments may rely on semi-trusted operators for round creation, oracle updates, or settlement execution. Over time, these operations should be minimized, automated, and decentralized where feasible.

VIII. MARKET OPPORTUNITY

The immediate opportunity lies in short-duration crypto-native markets. Crypto users already understand fast feedback loops, visible price movement, and risk-based participation. Fifteen-minute BTC prediction rounds match this behavior by offering rapid settlement and repeated engagement.

KangarooDeFi can also expand into social, behavioral, and community markets. The ability to tokenize outcomes and display real-time sentiment creates a foundation for viral market creation. A future permissionless market layer could allow users to register external tokens, pairs, or event conditions and then create AMM-priced prediction rounds around them.

IX. DISCUSSION

A. From Betting to Market Formation

The core transformation is:

$$\text{Belief} \rightarrow \text{Tokenized Exposure} \\ \rightarrow \text{Market-Priced Conviction}.$$

Instead of recording a static bet, the system converts directional belief into an ERC20 position whose displayed price evolves through AMM reserve dynamics.

B. Composability and DeFi Integration

Because outcome tokens are ERC20 assets, they can be held in wallets, indexed by external services, and used as settlement claims. Future versions may extend this composability into external trading venues or DeFi integrations, subject to transfer restrictions, round lifecycle rules, and settlement safety.

C. Limitations

The current version is not a fully symmetric secondary market because direct sell-back to the same round pair is restricted. This simplifies settlement safety but reduces exit flexibility. Future work can explore controlled sell-back mechanisms, dynamic exit windows, or asymmetric AMM curves that preserve settlement integrity while improving secondary-market behavior.

X. COMPETITIVE POSITIONING

A. Against Orderbook Markets

Orderbook prediction markets require active counterparties and sufficient bid-ask depth. In low-liquidity environments, execution quality deteriorates. KangarooDeFi replaces order matching with deterministic AMM availability, ensuring that each round opens with protocol-initialized buy-side liquidity.

B. Against Pool Betting Systems

Pool betting systems aggregate capital but often fail to generate continuous price signals during the active phase. KangarooDeFi converts directional participation into AMM-priced outcome tokens, allowing conviction to be reflected dynamically throughout the round.

C. Against Hybrid Platforms

Hybrid platforms improve accessibility but may depend on off-chain components, fragmented liquidity, or non-AMM settlement mechanisms. KangarooDeFi positions itself as an AMM-native prediction primitive in which token creation, liquidity initialization, participation, and settlement are coordinated through smart contracts.

XI. FUTURE DIRECTIONS

Future versions may explore adaptive AMM models, volatility-sensitive curves, dynamic fee structures, cross-market liquidity

sharing, controlled exit mechanisms, and permissionless market creation. Outcome tokens could also become inputs for structured products, lending modules, or reputation-weighted market systems.

XII. CONCLUSION

KangarooDeFi introduces an AMM-native approach to decentralized prediction markets. The protocol creates round-specific outcome tokens, initializes liquidity through kwBNB, and lets participant demand reshape reserve ratios during the active phase. The current implementation is best understood as a buy-side conviction market with settlement-based exit. Its central innovation is not merely predicting outcomes; it is turning belief into a market-priced, tokenized position.

REFERENCES

- [1] R. Hanson, “Combinatorial information market design,” *Information Systems Frontiers*, vol. 5, no. 1, pp. 107–119, 2003.
- [2] H. Adams, N. Zinsmeister, M. Salem, R. Keefer, and D. Robinson, “Uniswap v3 Core,” Uniswap Labs, 2021.
- [3] G. Angeris, A. Agrawal, A. Evans, T. Chitra, and S. Boyd, “Constant function market makers: Multi-asset trades via convex optimization,” 2021.
- [4] J. Peterson, J. Krug, M. Zoltu, A. K. Williams, and S. Alexander, “Augur: a decentralized oracle and prediction market platform,” 2018.