**AvaxSportsBook**

Sports Betting Contract--Preliminary

Eric Falkenstein

V1.0 8/14/2023

**Abstract.** AvaxSportsBook is a decentralized blockchain contract for betting on weekend sporting events. Token holders administer an oracle contract that posts weekly events, odds, and results to a betting contract. Users can either bet or provide liquidity to accommodate residual imbalances. Cross-margining allows a small number of liquidity providers to support unlimited betting and diversify across events.

Contents

[1 Introduction 2](#_Toc142913146)

[Simplicity from Limited Focus 3](#_Toc142913147)

[Oracle Incentive Compatibility 3](#_Toc142913148)

[2 Contract Basics 5](#_Toc142913149)

[2.1 Outline 5](#_Toc142913150)

[2.2 Schedule and Start Times 6](#_Toc142913151)

[2.3 Odds 7](#_Toc142913152)

[2.4 Betting Capacity and Cross Margining 7](#_Toc142913153)

[2.5 Betting and Redeeming 8](#_Toc142913154)

[2.7 Liquidity Providers (LPs) 8](#_Toc142913155)

[2.8 Emergency Functions 8](#_Toc142913156)

[3 Oracle Incentives 9](#_Toc142913157)

[3.1 Sending and Validating Oracle Data 9](#_Toc142913158)

[3.3 How Oracle Token Holders Claim Oracle's Eth Revenue 10](#_Toc142913159)

[3.4 Tests 10](#_Toc142913160)

[4 Conclusion 10](#_Toc142913161)

[Appendix 12](#_Toc142913162)

[Odds Translation 12](#_Toc142913163)

[Redeeming a Bet 13](#_Toc142913164)

[Function Restrictions 13](#_Toc142913165)

[LP Rewards 14](#_Toc142913166)

[LP Eth to LP Shares to LP revenue 14](#_Toc142913167)

[Oracle avax Revenue 15](#_Toc142913168)

[Margin Adjustment for New Bet 16](#_Toc142913169)

[Gas for transactions 18](#_Toc142913170)

[Settlement Detail 18](#_Toc142913171)

# 1 Introduction

Sports betting is ideally suited for a completely on-chain smart contract. Consensus odds for simple bets on major events are well-known, statistically accurate, and stable. For American football and mixed martial arts, the weekly schedule allows for a consistent process of that that is straightforward to manage. A contract targeting a prominent portion of this multibillion dollar market is small enough to manage and big enough to matter.

There are three types of ASB contract users: bettors, liquidity providers, and the oracle collective. Bettors can take either side of any regular bet offered, subject to a size constraint based on the amount of free liquidity provider (LP) capital. Odds are generally posted by Wednesday morning, allowing bettors to then bet until gametime. The weekend’s events are settled the following Monday, allowing withdraws early Tuesday morning.

LPs earn a positive return for the risk they take given a built-in vig of 4.5%, the standard betting cost implicit in even-money odds of -110.[[1]](#footnote-1) While sportsbook odds are efficient, in that it is difficult to make a profit betting, week-to-week the book could lose money due to small sample variation, and so the LP returns are for bearing risk. The ratio of LP capital relative to the amount of betting determines the return on equity, so the amount of LP capital to bet volume will equilibrate this market (if the return is too low, capital will leave, raising the expected LP returns).

Bets are automatically cross margined so that the capital required is minimized. For example, 10 AVAX collateralizes a single bet paying out 10 AVAX, and a contest where the winning payout is 510 AVAX if one team wins and 500 AVAX if its opponent wins.[[2]](#footnote-2) The required LP capital on any single event is a function of the maximum net payout, not the notional bet amount. LP capital is applied to an entire slate of up to 32 events, and an adjustable parameter limits how much of this capital can be applied to any one event.

An exclusive oracle provides and validates data sent to the betting contract. Submissions are sent by a single token holder of sufficient size, and the collective has at least 11 hours to vote before it can be sent to the betting contract. The token holders only receive their full fee payment if they vote at least twice a week, which requires them to post their tokens on the oracle contract. If a token holder votes less than twice a week, their oracle revenue is reduced proportionately (e.g., making one vote per week, which is half of the target, would entitle them to only 50% of their payout). The reduced amount is reallocated to the other token holders in the oracle contract.

Initially, I distributed 40% of the immutable lifetime supply of oracle tokens to three people I found able and willing and able to administer this contract. I created this, but I have no control or financial interest.[[3]](#footnote-3) The fact that I must give it away is likely a major reason why no one else has created such a contract. The other 60% of the supply is set aside as rewards for LPs, so that each week LPs can get a pro-rata share of the token rewards available that week. If all the LPs claim rewards each week, the incentive program will last 20 weeks, and no further tokens will be created.

The standard 4.5% vig in sport book odds has been stable for decades. As this is a competitive market, it reflects an equilibrium balancing the demands of bettors and bookies as opposed to monopoly power. By taking this vig as a given, we remove tactics that create delays or hack surfaces. The standard odds advertised online on major fights and football games are stable and efficient, so simply using these odds is also efficient. The hassle-free ability to provide liquidity and take bets is sufficient to make it a dominant alternative for many bettors.

The relative stability of sporting event odds make the odds amalgamation simpler than most applications. Consider that your average daily stock price volatility of 2.5% is 16 times greater than the average bid-ask spread of 0.15%. A market maker who adjusted their bid-ask spread once daily would be exposed as a 'money pump' by arbitrageurs, in that if the price moves up 2.5%, the market maker will almost certainly have sold on the way up, generating real-time losses.

In contrast, the implicit spread on money line bets is 2.5% in terms of a win probability. One needs a 2.5% edge in predicting which team wins to beat the house. The daily volatility of these odds—translated into a win probability—is significantly less than half of that, implying the book would make money even if it had day-old odds and the new odds were moving in the right direction.[[4]](#endnote-1)

Sports odd stability eliminates the adverse selection problem in high-latency centralized markets, which allows for a novel price-setting mechanism: post the widely available standard odds for a slate of upcoming contests. A singular oracle token holder submission grabbed *in toto* from a large sports book will be sufficiently close to the optimal odds to prevent bettors from arbitraging the LPs. A single human can easily find web pages that concisely present the odds or results of high-profile matches presented on ASB. There is no need for price discovery on high-profile sporting events.

## Simplicity from Limited Focus

Football and MMA are almost exclusively weekend events. A focused set of weekly win/lose events makes incentivizing the oracle easier. In contrast, if we targeted a standard centralized sportsbook, it would cover diverse events on most days of the week, and then many obscure bets that are less statistically robust. Only a subset of the oracle token holders could evaluate such data, creating edge cases where a minority of token holders can take advantage of inattentive oracle token holders, a major attack surface. This also makes the oracle easier to monitor, in that the historical event logs refer to standard bets on weekend events.

# Oracle Incentive Compatibility

Incentive compatibility is vital to low-cost enforcement of contracts, and historically this mechanism centered on reputation, not contract law administered by the state.[[5]](#footnote-4) The blockchain's transparency, immutability, and pseudonymity make reputation much easier to monitor. When agents have incentives aligned with their counterparties, we minimize indirect costs (delay and spread), making it more attractive for bettors. If the bettors and LPs want to cheat, they need to collude with the oracle; if the oracle does not cheat, bettors and LPs cannot cheat. The focus is to ensure honest and timely reporting of odds and results is always the oracle collective's value-maximizing act.

Creating a game where honesty is an oracle's best strategy is straightforward; the keys are simplicity and a repeated game, which leads to easy monitoring and a strong incentive towards punishing cheaters. Adding superfluous parties, tokens, and scope increases cost, complexity, and delay. By putting players into a repeated game, an oracle cheat becomes dominated by cooperating because the one-time gain is less than the present value of future revenue foregone. Incentive compatibility is critical to low-cost enforcement of contracts, and historically this centered on reputation, not contract law administered by the state.[[6]](#endnote-2)

Consider the following cost-benefit analysis for ASB's oracle. A conservative equity price/earnings (P/E) ratio is 10. Assume a betting contract has 100 AVAX bets on its books, both long and short. As the oracle's fee is about half of the vig, this would average about 2.5 AVAX in weekly revenue. Given 50 settlement events over the year, this annualizes to 125 AVAX. Given a 10 price/earnings ratio, this values the oracle collective at 1,250 AVAX. The maximum potential cheating revenue in this example is 100 AVAX, requiring the hacker to make all the book's net exposure on his pre-ordained picks, so the LPs have net exposure to the wrong side of every bet.

Such a scam would be conspicuous in the readable event logs, and no rational person would use this contract again, making the value of the oracle token zero. A voting majority's oracle token has a present value of 625 AVAX. Honest reporting is the dominant strategy in this improbable worst-case scenario, in that 625 >> 100.[[7]](#footnote-5)

The oracle voters have, literally, all day to evaluate a data submission that can be evaluated in a couple of minutes. A majority 'no' vote among token holders cast penalizes the proposer of the data.[[8]](#footnote-6) The limited focus of events, timing of oracle submissions remove any plausible deniability for the oracle cheat action in any single event each week. I provide a spreadsheet where a noob can input the weekend’s matches and generate the relevant data into a text file, which a python program then sends to the contract. As there is only one submission allowed per day, and a single oracle token holder sends the data, the average oracle token holder merely evaluates the day’s submission for obvious fraud, and then votes with a click. The website generates event log data in readable form, so one does not need specialized knowledge of hash functions to evaluate the oracle’s behavior.

While it is simple enough to incent the oracle properly, this only protects the contract against insiders. In contrast, decentralization defends this contract against outsiders. Powerful institutions have always used centralized power to prevent competition, often using disingenuous rationales emphasizing safety. Such an attack needs a choke point, prevented if a collective of pseudonymous accounts worldwide administers the oracle. If the oracle is profitable, an effectively infinite number of people will replace oracle token holders captured by outside attackers, as their job does not require more than the 5 or 10 minutes needed to evaluate the weekend’s event schedule, odds, etc.

Initially, the oracle will be relatively centralized in its initial handful of token holders. I chose three people, who do not know each other, to give 13.3% of the lifetime supply of tokens for a couple of reasons. First, one of them could become incapacitated for various reasons. Secondly, if one decides to go rogue, the other two will probably discipline him (they would reject his data submission, so it would never reach the betting contract to affect anything). The other 60% of tokens will be distributed to the LPs over the initial 20 weeks of the contract’s existence.

# 2 Contract Basics

## 2.1 Outline

Event data, including odds, are sent to the contract on Tuesday, allowing people to bet on weekend events from Wednesday to game time (which is loaded in UTC with the odds). The subsequent Monday, the oracle sends the results to the betting contract, settling that week's bets, and the contract then repeats the process.

Bettors and LPs need only interact only with the betting contract, while the token holders only interact with the oracle contract. All transactions with the betting contract are denominated in AVAX, including providing liquidity and betting.

The LP capital backstops residual imbalances in the book. The LP's total capital is available for all contests that week, and there is a limiting mechanism on how much AVAX can be allocated to any single contest. This diversifies the LPs, reducing the chance that a single contest outcome could extinguish LP capital. The betting contract contains all the methods for bettors and LPs: betting and redemption for bettors, investing, and withdrawal for LPs.

A singular oracle token holder, with at least 10% of the oracle tokens outstanding, proposes the relevant data: the upcoming schedule (who plays whom, when), odds, results, and some technical parameters (e.g., paused matches, the diversification parameter). Each submission is then subject to an evaluation period subject to a majority vote: send or reject. A successful data submission is sent to the betting contract after a voting period of 11 hours, giving the oracle collective sufficient time to veto a fraudulent or incompetent submission. To discourage misbehavior, token holders submitting failed data submissions are charged a modest fee.

odds, results, etc

avax fees

Oracle Contract

Equity Token Holders

Betting Contract

data, votes

avax

avax

tokens

Bettors, LPs

**Weekly Schedule**

Standard Oracle Actions

|  |  |  |
| --- | --- | --- |
| **Day** | **GMT Hour** | **Oracle Action** |
| Tuesday | 12 | post Schedule, start times, odds |
| Wednesday | <12 | process |
| Wednesday | 12 | post odds update |
| Thursday | <12 | process |
| Thursday | 12 | post odds update |
| Friday | <12 | process |
| Friday | 12 | post odds update |
| Saturday | <12 | process |
| Monday | 12 | post results of weekend matches |
| Tuesday | <12 | process |

Odds updates are optional.

The data submitter automatically votes for his submission, so if no one votes, it will succeed, as it is a simple majority vote that determines success or fail. Posting data can only occur in the 60-minute window where the GMT hour is 12: between 12:00:00 and 13:00:00 GMT. Voting takes place between the GMT noon posting and processing the post, which can take place at any time, by anyone, at a GMT hour less than 12. This gives the oracle at least 11 hours to evaluated each data submission.

If a settlement or initial post are rejected, then a settlement or initial post, must be posted again. This effectively delays the contract by a day. Thus, the days listed above can be changed if a post is rejected. For odds updates, these are optional and can be omitted. Odds updates cannot occur after Friday night, and settlement can only be posted the following Monday, so the oracle token holders do not have to do anything until settlement (indeed, they cannot do anything).

Each week, aka ‘epoch,’ the MMA, boxing and US football games that weekend are sent to the oracle contract. The data are sent to the betting contract if most oracle token holders vote yes. When the betting contract has data for the next weekend, bettors can bet up to the time of the various games that weekend, as each game is given a starting time. Odds can be updated, but only once a day, and only in the same 60-minute window each day.

After the weekend, the outcomes are sent to the betting contract, which settles all the week's bets. Once the settlement is sent to the betting contract, which can occur at midnight GMT, bettors can redeem their bets. The contract has no ability to seize neglected funds, so as long as the blockchain exists, users will be able to safely let unredeemed money sit in the contract. LPs cannot withdraw or deposit during the period between the start of the first game and settlement, as otherwise, they could game the contract by anticipating unusual losses or winnings.

## 2.2 Schedule and Start Times

Each betting period will contain up to 32 events and target a weekend (e.g., Friday night through Sunday night). Each contest is slotted into an array that can be unambiguously linked to its outcome via event logs that expose what events odds were on the contract. The schedule array contains a string with the sport (NFL, MMA, etc.), the two opponents, and the starting time. The initial favorite will be listed first and the underdog second, though the odds can change over the week while the ordering of the contestants cannot.[[9]](#footnote-7)

## 2.3 Odds

The contract generates odds with an all-in vig of approximately 4.5%, the standard vig at major betting books. The ASB.co frontend presents all odds in terms of payout to the bettor, the 'all-in' odds that include the fee to the oracle. Thus, when a bettor sees 2.000 decimal odds in the ASB GUI they can be sure they will receive 2.0 AVAX if they bet 1.0 AVAX, etc. The oracle may update odds over the week, but the odds posted on the contract at the time of the bet are applied.

## 2.4 Betting Capacity and Cross Margining

Margin rules make sure all bets are fully collateralized. Unlike in futures markets, the margin is not derived from a probabilistic risk, such as an instant 20% price movement. Such a rule exposes a contract to insolvency, as there is a possibility of price movements greater than 20%, which becomes increasingly probable over time. In betting on binary outcomes, the worst-case scenario for the house is assumed, which is like assuming a 100% loss on each of their positions. The contract will always be fully collateralized on all bets, as this is enforced at the time of each bet. There is no insolvency risk.

As bettors take the opposite side of a contest, it is a waste of capital to require the LPs to collateralize both sides independently. The solution involves netting exposure.

**Cross margining example:** Assume two teams are given even odds so that for either team, a 1 AVAX bet pays the winner 2 AVAX. If there is 10 AVAX on team A, and 10 AVAX on its opponent, team B, it would be a 'flat' book in that the LPs have no exposure to this game; payoffs are funded by betting counterparties, not the LPs. A new bet that pushes the book to have a net exposure would necessitate LP funds as collateral, so a bet of 2 AVAX on team A would move 2 AVAX from the LP's free margin to the LP's locked margin. Given a total of 12 on team A, and 10 on team B, a bet of 2 AVAX on team B would move 2 AVAX *out* of the required margin to the free margin because the resulting book would be flat again.

Adjusting the net required LP margin involves 'linear programming' where the LP's net game exposure is the maximum liability of either team winning. The margin adjustment is applied at the time of a bet, so there must be sufficient free LP collateral to accommodate a bet adding to LP exposure.

A contract parameter prevents an overconcentration of LP capital on one event. For example, 123 AVAX in total LP capital and a concentration parameter of 10 implies a maximum of 12.3 AVAX LP exposure for any event. Thus, if the current LP liability for team 0 winning was 10.0 AVAX, it could only accommodate an additional payout of up to 2.3 AVAX on team 0. In contrast, a bet on team 1 could accommodate a bet payoff of 22.3 AVAX. This concentration parameter can be adjusted over time by large oracle holders.

The concentration parameter and a limited LP pool limit the damage to stale odds. If a contest offered odds significantly deviating from the true odds, the LPs are limited on their exposure to that one contest. The concentration parameter and the amount of LP capital not currently used as required collateral determine the maximum bet size on any contest and can be seen on the front end.

## 2.5 Betting and Redeeming

All ties and 'no contest' games give bettors their initial bet back. Winners receive their bet amount plus the payoff implied by their bet odds. When the week's results are sent to the betting contract, all bets are settled, and the oracle payment is sent to the oracle contract. Settlement creates a mapping allowing bettors with a win or tie outcomes to redeem their bets.

Bettors redeem all of their outstanding bets at once. The redeem function loops through up to 16 resolved bets, and sends the winnings to the bettor’s account. Redemption can only be processed if there are no active bets in the account, so bettors must wait until settlement to redeem old bets if they have a new, active bet. If an account has 16 unredeemed bets, it must redeem them before it can place another bet.

Bettors and LPs do not use specialized tokens, just native AVAX. All bets are fully collateralized, so if one has access to the account used for sending the bet or LP investment, a user's funds are safe in the contract. Unclaimed bets will reside in the contract forever, as mechanisms to sweep neglected funds to LPs or oracle token holders would introduce attack surfaces.

## 2.7 Liquidity Providers (LPs)

To become an LP, one sends AVAX to the betting contract, which then credits the LP with shares representing their pro-rata ownership of the LP pool. For example, if there is 10 AVAX in the LP book, and 10 shares, adding 1 AVAX would give a new LP a 1/11th share of the new pool of 11 AVAX, keeping the AVAX/share the same. This LP claim exists only within the betting contract and is tied to the initial LP AVAX account address. It is not transferable to other AVAX addresses, and so is not represented by a token. The size of the LP capital should adjust to the volume and degree of cross-margined trading, which will determine the expected pnl. The relation of an LP's share value will be equilibrated by its net capital, which makes creating an LP share token problematic.

LPs can only withdraw during the inactive period if margin is available. A bet backed by LP margin locks this margin until settlement, or bettors take the other side, freeing the LP capital. Since there is at least a 24-hour window each week after settlement (~Monday 7 PM ET) before new bets are offered (~ Tuesday 7PM ET), LPs are sure to be able to withdraw at least once each week. More practically, there will be free margin available for marginal LPs to withdraw over much of every week, as bettors will probably not max out the bookie's free margin in the first days of the week.

LPs must also have their AVAX in the contract for at least two settlements. If LPs could withdraw after only one settlement, people could add large amounts of capital just before the weekend when the pool has little net risk and then take it out right after settlement. For example, if all bets were fully collateralized by offsetting bettors one week, the LPs would receive a certain profit given the vig built into the odds. Outsiders could provide superfluous liquidity just before the weekend and withdraw immediately after settlement, generating a certain profit. This would dilute the profits of LPs providing 'real' liquidity. Therefore, an LP must expose herself to at least one betting period before withdrawing.

## 2.8 Emergency Functions

There is no outside adjudicator to rectify problems, as this would delay payments and complicate the contract (how to incent the adjudicator?). All problems must be solved on-chain within these contracts.

If a hacker could sneak in bad odds that enabled a cheat, the oracle collective could nullify this action by posting a result of a 'tie' regardless of the outcome. This allows the LPs and bettors to get their money back as if nothing happened, and the incorrect (but fair) tie result should be clearly explained by the event logs showing the earlier hack. This would be an extreme scenario, like a fork in a blockchain, but it is helpful to anticipate.

Suppose off-chain odds change quickly and significantly, exposing the LPs to arbitrage. In that case, oracle token holders can immediately pause up to two bets. This action does not require the usual 12-hour vetting period to allow oracle token voting. It does not expose LPs or bettors to losses but prevents new bets, which would be allowed again if new odds are posted. This method is restricted to large token holders to avoid mischievous trolls who might want to annoy users at little cost.

2.9 Avalanche

Snoball consensuss

oracle incentives by specialized oracle group: censorship, accountability

# 3 Oracle Incentives

## 3.1 Sending and Validating Oracle Data

A token holder needs at least 10% of the outstanding tokens to submit data to the oracle contract. A power law distribution will always accrue in ownership, and the top owners should find it in their best interest to lead the oracle administration. While submitting data takes some effort, the cost is relatively low given the ease at which relevant data is available and the limited scope and frequency of data submissions. Larger token holders should be sufficiently motivated to send data to the contract promptly.

Tokens must be deposited within the Oracle contract to submit or vote on submissions. This prevents double-voting and forces the token holders to attend to the contract they should be monitoring. The tokens are meant for governance, not speculation, and generate dividends directly proportional to the bet volume. Token holders cannot vote more than once on any data submission, which requires that token holders cannot withdraw tokens while a vote is active.

Initial data proposals must be sent between 12:00 and 13:00 GMT (7-8 AM summer New York), and voting on proposals lasts at least 11 hours. This gives oracle token holders sufficient time each day to see each data submission.[[10]](#footnote-8) Upon submission, the proposal can be submitted to a vote count at any time after 18:00 GMT. A yes vote sends the data to the betting contract, while a no vote burns a fraction of the proposer's bond and resets the state for the next data proposal.[[11]](#footnote-9)

If a majority vote of token holders rejects the data sent, the contract is reset to allow a new submission. One-fifth of the bonding payment is burned, which should be large enough to discourage fraud but small enough to make gratuitous rejections unattractive.

## 3.3 How Oracle Token Holders Claim Oracle's Revenue

Each week the oracle receives 5% of the bettor winnings as a fee for their service. This is consistent with the Oracle receiving 2.5% of bet amounts and the LPs receiving the other 2.5%. While the oracle could receive nothing if all bettors lose, the oracle can never lose money. Token holders must vote on data submissions to get their revenue, which requires they keep their tokens in the oracle contract. For example, if their tokens were in the contract for three settlements, and they voted twice, they would receive one-third of their payment. The other two-thirds would be added back to the pool to go to the other token holders, incenting the token holders to actively monitor the the data submitted.

A single token holder submits the data, and the other token holders have 11 hours to evaluate this data and vote yes or no. As the contract targets high-profile matches, the odds will be available on many websites, and it should take only a few minutes to see if the matches, start times, and results are accurate, or the odds are reasonable.

## 3.4 Tests

On GitHub, there are several test scripts documenting the integrity of the contract methods. These JavaScript files correspond to hypothetical transaction sequences documented in ASBtests.xls, and are described in ASBtests.docx.

# 4 Conclusion

Most sports betting sites touting their crypto functionality are conventional ones accepting crypto. A truly blockchain-based betting dapp upholds Satoshi's vision of *pseudonymity*, *confiscation-proofness*, and *permissionless access*, which requires it to have no off-chain presence. I hope that a focused dapp with good incentives can provide an example of what works on the blockchain. The purpose of the contract is to facilitate betting, not create a new token for speculation. However, unlike most dapps, where tokens have a vague governance role and hypothetical fees, ASB’s token holders have an essential job and get revenue instantly.

One should expect players to always act in their selfish best interest. A sustainable contract creates a repeated game where honesty is always the dominant strategy for every player. Simplicity is crucial in generating good game theory equilibria because the state space grows exponentially in the number of players and actions they can take. An incentive-compatible contract avoids the more costly solution of establishing adjudication procedures and slashing conditions for various infractions. The trust one puts into the ASB Oracle is fundamentally the same as why investors trust miners: the rational self-interested assessment that honesty dominates dishonesty for a hypothetical individual.

Blockchain betting contracts are the perfect application of straightforward rules to a common use case. ASB presents a quick and efficient way to get asset exposure without the many hassles in standard contracts. Sports betting is ubiquitous, but it should be easier. This contract provides a simple way to do that.

# Appendix

## Odds Translation

To convert moneyline odds into Decimal odds, we have the following.

For positive moneyline odds: (Moneyline odds/100) + 1 = Decimal odds

For negative moneyline odds: (100/Moneyline odds) + 1 = Decimal odds

To translate decimal odds into moneyline odds that are prominent on NFL betting sites, we have the following adjustment mechanism:

If decimal odds are greater than 2.0: 100 × (decimal odds – 1) = Moneyline odds

If decimal odds are less than 2.0: -100/(decimal odds -1) = Moneyline odds

To translate moneyline odds to fractional odds:

For positive moneyline odds: Moneyline odds/100 = Fractional odds

For negative moneyline odds: -100/Moneyline odds = Fractional odds

**Odds in the contract**

Odds are available on many betting websites, and arbitrage limits how far these odds can differ. On average, a team's implied probability of winning will change by only 2% over the week, rarely over 5%. All odd postings and updates are recorded in event logs, observable in online queries at sportAVAX.co.

A contest will have a single odds number posted for a contest. These odds are supplied only for the initial favorite using a truncation of preliminary decimal odds. For example, 1.909 would be stored as 909, 2.50 as 1500, etc. This number, however, is just relevant to the team in slot 0, the initial favorite. Further, it needs to be adjusted to reflect the oracle fee that would be assessed. Thus, the betting odds for a favorite where the match odds were 955 would be 910, via

Bettor Odds (favorite) = (contractMatchOdds \* 0.95)/1000 + 1

= 955\*0.95/1000+1=1.907

The odds for its opponent are generated within the contract by the following formula:



This transformation generates odds on the team/player in slot 1, the initial underdog, such that the book has a 2.5% vig.

Then to account for the oracle take, the all-in odds for team 1 would be

Bettor Odds (underdog) = (underdogOdds \* 0.95)/1000 + 1

With this method, we can ensure that the set of odds for a contest generates a positive vig, removing a potential attack vector.[[12]](#footnote-10) This formula generates a vig of 2.5% for the LPs via parameter 45 in the above equation, and the 5% take of winnings generates an approximate 2.5% vig for the oracle.

Initial odds presented must be between 1.999 and 1.125. The cap at 2.000 reflect the fact the initial odds apply to the initial favorite, while the 1.125 minimum removes events where the initial odds are greater than 8:1, lopsided contests. Eliminating high payout contests mitigates risk, as such events would invite hacker focus. Further, these events are rare, and so are more difficult to validate historically.

The website sportAVAX.co displays the decimal odds users receive if they win. For example, a user seeing odds of 1.900 will receive back 1.900 times their bet amount.

The most common odds offered for the NFL are presented in moneyline form as ‑110 for both teams. A flat book on such a wager would receive 220 and payout 210. In this way, the 'house' makes money used to pay for various costs and a profit from the house. The implicit profit ('vig') in this case would be 4.55%, 10/220. The general formula for estimating the vig is given by the following formula, where *p* and *q* are decimal payouts (e.g., 1.909 for a standard even money bet) for opposing teams.



The spreadsheet 'ASB.xlsx' presents a page where people can see how these transformations are applied. Those interested in sending odds to the contract will find it a helpful template.

## Redeeming a Bet

Bets are stored in a 16-element array, and after 16 bets, no further bets can be made until they are redeemed. Redemptions can only occur when a bettor has no active bets, so a bettor should redeem his bets after settlement if he anticipates a problem. All bets in the array are settled for the bettor. Each bet is represented by the unique combination of epoch, match, and pick. At settlement, a bets hash refers to a struct containing this information, and a mapping generated at settlement allows redemption. SportAVAX.co provides this data by reading the user's MetaMask address, allowing users to redeem bets by clicking a single button (it is one transaction). However, anyone can log onto the blockchain using the account used to make a bet and submit the bet hash.

## Function Restrictions

The most basic restriction is that the three data submissions can only be applied during the 12th hour on any day. No other submissions can be made until this is processed, and token holders have 11 hours to evaluate the data. After midnight, anyone can then process the vote, and if it passes the data are sent to the betting contract, while if it fails the process starts over. Thus, if a vote fails, this will push back the contract’s starting bets or settlement by at leat one day.

reviewStatus is a variable in the oracle contract that prevents the contract from updating in the wrong order. For example, after sending the initial set of matches, start times, and odds, one can only add an update to the odds, or the outcomes of those matches; one should not send another batch of matches, as this would make any existing bets point to different matches. Further, one cannot send new odds or results until the initial set of matches was processed via a successful vote. Additionally, token holders cannot withdraw while a data submission is under review, as this makes it easy to ensure each token holder votes only once on each data submission.

reviewStatus (RS) variable as control mechanism for constraining the data flow

required output vote succeeds vote fails

initPost 0 10 2 0

updatePost 2 20 2 2

settlePost 2 30 0 2

voteProcess 30 0

withdrawTokens <10

When the initial data are sent, the first start time is used to find the nearest Friday 7 PM ET, and this time prevents LPs from withdrawing until settlement is processed Monday night. This first start time also prevents any odds updates on the weekends, as odds should be sufficiently stable over the next 48 hours, and it allows the oracle token holders to rest. Finally, the outcome cannot be sent until Monday, in the 12th hour GMT. This prevents a surprise data submission that could sneak by the oracle.

Odds sent must be greater than 1.125 and less than 2.00. These are for the initial favorite, so the cap at 2.00 enforces this. Restricting matches where an initial odds are less than 1.125 for the favorite removes long-shot bets that are most subject to fraud, and are sufficiently infrequent to be irrelevant.

## LP Rewards

## LP Eth to LP Shares to LP revenue

LPs own a pro-rata portion of the contract's revenue based on their percentage of LP capital before that week's events. Statistically, the LP capital will grow each settlement due to the vig; this is how LPs make money. As the relevant LP credit/debit occurs at settlement, the LP's AVAX/share value is fixed each week when users can withdraw or invest.

An initial investment generates the following shares:

LPshares = AVAX invested × LpTotalShares / TotalCurrentLPAVAX

For example, assume the contract has 123 AVAX owned by its LPs, who have 100 shares. This AVAX may be sitting free or locked up as collateral for upcoming contests. This implies each LP share is worth 1.23 AVAX.

LP AVAX LP TotalShares avax/Share

123 100 1.23

Suppose Alice wishes to invest 10 AVAX into this pool. The above formula implies she would receive 8.13 shares (10/1.23). This would change the pool's balance sheet to

LP AVAX LP TotalShares avax/Share

133 108.13 1.23

Note the ratio of AVAX/share is the same after Alice's investment, so existing shareholders do not lose or gain money via Alice's new investment.

If we assume the LP collective gained 2 AVAX that week, the new balance sheet after a settlement will look like this:

LP AVAX LP TotalShares avax/Share

135 108.13 1.25

The increase from 133 to 135 reflects a 1.5% profit from that epoch's games. If Alice then sold her shares, she would receive AVAX using a transformation of the above formula:

avax Withdrawal = TotalCurrentLPAVAX × SharesSold / LpTotalShares

Selling 8.13 shares would generate 10.15 AVAX, a 1.5% return on their investment, identical to how much the AVAX LP pool rose over that period.

In this way, any LP investment or withdrawal reflects the percent change in the size of the LP pool over the investment period.

One can see this in the contract tests

## Oracle avax Revenue

Oracle token holders must deposit their tokens in the oracle contract to vote, and then are expected to vote at least twice a week. The two fundamental data submissions are the initial set of contests, odds, and start times, and then the results of those contests. When a weekly settlement transaction is executed, the oracle's 5% fee is applied to the winnings and sent to the oracle contract. The '*feePool'* state variable reflects the lifetime amount of AVAX per token paid to the oracle contract.



When an oracle token holder deposits into the contract, their account notes the current value of *feePool*. When that oracle token holder withdraws or adds to their account, the token holder is sent their entire accrued AVAX using the formula



Having tokens in the oracle is a necessary but insufficient condition for being paid. The contract then takes the total number of tokens







This account's OraclePoughback is then applied to the Oracle feePool as if it were revenue from the epoch’s settlement. This gives an incentive for token holders to wait as long as possible to retreive fees, as unclaimed fees are secure: there is no scenario where the token holders can lose accrued revenue, either due to risk or an oracle hack. is then updated to the currentFeePool, so another immediate add or withdrawal of a token by the same token holder would have CurrentFeePool – UserOldFeePool=0, and receive nothing.

With this method token holders can be sure the contract is in balance, where accounts payable are equal to AVAX in the contract at all times.

## Margin Adjustment for New Bet

There are three types of margin tracked by the contract, all held in the array variable' margin.'

**bookiePool**

This is AVAX owned by the LPs, both free and locked up as collateral.

**bookieLocked**

This is AVAX owned by the LPs that are unavailable for bookie withdrawal. It represents the gross worst-case scenario loss for the LPs.

**bettorFund**.

These are bettor funds applied to outstanding, taken, bets.

LPstruct. Mapping between an address and its share amount and time of investment. The total number of LPStruct.shares will equal sharesOutstanding, reflecting the percent ownership of an address of the bookie capital.

Free LP capital in margin [0] is available for new bets that increase the contract's net liability. New bets that increase the contract's net position will transfer AVAX out of margin [0] into margin[1]. Bets that decrease the bookie's net position will move AVAX from Margin [1] to margin [0]

For a team with standard decimal odds of 1.909. the total payoff for a win can be separated into two components: 1 + 0.909, the latter term representing the bettors net profit, and the former term representing the bettor's initial bet. The amount 1\*betAmount is available from the bettor funds, while 0.909\*betAmount must come from the LPs or bettors taking the other side. Odds are stored such that



For example, the standard odds of 1.909 for an 'even' bet would have odds in the contract stored as 909.

LP Required Margin is the sum of the maximum liability for all the events in an epoch. Each event is independent, so the book is correctly margined by correctly margining all the individual bets. Thus we need merely describe how margining occurs for a single event, knowing these are then summed for determining the overall Required Margin.

The total amount owed if team 0 wins equal the sum of the bet amount and its payoff for all the bets taken on team 0. Let us define two types of capital used to pay bettors, the payout or profit, with must come from someone other than the bettor, and the bettor's initial bet amount, which is returned with his profit:



**betSum0** is the total amount bet on team 0, summing over all the bets on 0. Bettor funds are available for payout but not part of the LP's Required margin (which is in Margin [0] and Margin [1]). **paySum0** is the sum of the bettor's profit if team 0 wins, which requires AVAX from the LPs or bettors taking the other side. As the betSum of team 0's opponent, team 1, is available for paySum0, the trick is monitoring the ability to cover the LP's liability given the amount bet on its opponent. This generates the following maximum liability for the LP (*aka* required capital) for a contest in that it is the maximum liability to either team in a contest:



We add the zero term because the house will have only non-negative liability on every contest. For a new bet long on team 0 playing, the new bet and payout are added to the above max() equation and compared to the extant maximum liability. The difference is the change in the LP's required margin (margin[1]), which is offset by a change in the LP's free Margin (Margin [0]).



This is calculated at the bet time, and the LP's capital is moved between margin[0] (free) and margin[1] (locked) depending on whether the bet increases or decreases the LP collective’s net exposure. For example, an initial bet will increase the required margin, but a subsequent small bet on the opposing team would lower the required margin. A bet could move the book so that the net LP liability switches from team 1 to team 0 or consists of the decrease in the net liability on team 1. In any case, the above function captures the difference in the worst-case scenarios for contract liability.

In this way, the LP's total book exposure is cross-margined so that 1.0 AVAX capital can support many bets via incremental bets on both contestants. At settlement, only money not payable is returned to the bookie's free margin pool.

LP exposure across matches is independent. The assumption for LP exposure is the worst-case scenario, so there will be no chance of an insolvency, as a bet cannot be taken without capital available.

## Gas for transactions

**contract function gas**

oracle initialPost 410,000

oracle inital voteProcess 129,000

oracle update Post 112,000

oracle update voteProcess 72,000

oracle settle Post 99,000

oracle settle voteProcess 241,000

oracle vote 37,000

oracle deposit Tokens 52,000

oracle wd Tokens 84,000

bet bet 142,000

bet deposit as LP 54,000

bet deposit as bettor 48,000

bet wd as LP 44,000

bet wd as bettor 33,000

bet LP claims token Rewards 62,000

bet bettor redeems bets 55,000

## Settlement Detail

Each bet creates a struct that contains the team and week of the bet. These two inputs create a hash mapped to a number representing its game outcome: 0 for a loss, 1 for a tie, and 2 for a win. When the array of 32 results is sent to the settlement method, the mapping is created (the mapping is zero for uninitialized hashes, so unless updated, the mapping is 0). This mapping is then used for redemptions, in that a bettor claiming his winnings will need the {epoch, match, team} hash to map to a 1 or 2 to generate a payout.

In addition to creating non-zero hash mappings for non-losing teams, the total payments to all bettors were generated using the results and the paySum and betSum arrays:



Here 1*x* is an indicator function that is 1 if true, 0 else. The *WeeklyWinnings* represents the bettor profit, while the WeeklyPayBack represents the initial bettor funding. The oracle fee of 5% is applied to the bettor winnings, representing about 2.5% of the total bet amount. As individual payouts are less than or equal to the total payouts in any week, this rounding truncations on individual redemptions will not compromise contract solvency; rounding will not prevent redemptions.

At settlement, accounts are adjusted as follows:

Redemption capital = WeeklyPayBack

PayoffPot = WeeklyWinnings \* 95 /100

The bookie's capital then adds the money bet that week minus the payouts for wins and ties.

bookiePool = bookiePool + bettorLocked – redemptionPot – payoffPot

The oracle revenues are then just 5% of the WeeklyWinnings, and are transferred to the oracle contract in the settlement function.

The bettor's money exists in the residual and must be claimed via redemption. At redemption, their bet and its winnings are credited to the bettor's user balance, available for withdrawal or future bets.

After settlement, the bookieLocked margin is set to zero, so all LP funds are available for withdrawal.

1. In moneyline odds, -110, have dec odds of 1.909, generating a vig of 4.55%. This is because if 110 is paid into the book on both sides, and 210 paid out to one winner, the book take is 10/220, which is 4.55% [↑](#footnote-ref-1)
2. A 5-1 contest with zero LP risk would have 5 eth bet on one team for every 1 eth on the other. [↑](#footnote-ref-2)
3. I created something that I would like to use, and if successful will serve as an example, refocusing development away from providing grist for new token scams. [↑](#footnote-ref-3)
4. Closing line odds are actually worse for bettors than the opening line, though not significantly. Regardless, the risk of arbitrage arises whenever the odds on both sides winning implies a probability of less than 1.0, which allows the bettor to make a certain profit regardless of who wins. If the odds were offered by the same book, and that book allowed margin accounting, this would create a money pump. In practice, the opportunity comes from the odds generated by two separate books, which prevents cross margining these bets. Given there is a limit to any one event, and the opening and closing line odds are statistically equivalent, the LPs would make money even if such an odds discrepancy arose. [↑](#endnote-ref-1)
5. E.g., prior to commercial civil law there were courts along trade routes throughout Medieval Europe that enforced commercial laws (the *Lex mercatoria*), and its judgments were accepted not out of any legal authority granted by a state's monopoly on violence, but rather refusal would ruin one's business reputation and thus future revenue. [↑](#footnote-ref-4)
6. In iterated prisoner's dilemma games, the optimal strategy is not to play the Nash strategy of the stage game, but to cooperate and play a socially optimum strategy. An essential part of a strategy in a repeated game is that uncooperative play will reduce the payoff to both players in future periods. A player may choose to act selfishly to increase their own reward rather than play the socially optimum strategy, but if it is known that the other player is following a trigger strategy, then the player expects to receive reduced payoffs in the future if they deviate at this stage. An effective trigger strategy ensures that cooperating has more utility to the player than acting selfishly now and facing the other player’s punishment in the future. This is reciprocal altruism: I play nice because I then expect you to play nice in the future. [↑](#endnote-ref-2)
7. 625 = 50.001% of 1250 [↑](#footnote-ref-5)
8. The reward is done so that, statistically, the submitter should make just enough to cover the costs of the occasional inadvertent (e.g., ‘fat-finger’) errors that may engender a data submission rejection. Otherwise, statistically, the data proposer would have only downside for his work. Nonetheless, we do not want to make proposing data too attractive because that would imply the proposer would eventually acquire a strong majority of the tokens. [↑](#footnote-ref-6)
9. The favorite/underdog refers to the opening line, and so over the week the initial favorite may become the underdog. Nonetheless, the ordering is fixed in the initial event posting. [↑](#footnote-ref-7)
10. Time is set as an offset to GMT, so these hours shift with daylight savings. [↑](#footnote-ref-8)
11. The initial data provider’s tokens are credited as a yes vote, and votes are decided on a simple majority of votes cast. [↑](#footnote-ref-9)
12. A negative vig would allow someone to create positions that would generate arbitrage profits. [↑](#footnote-ref-10)