Analysis of Strategies and Game Balance in Two-Player Ascension

Dominant strategies, equilibria and how to make the game balanced

Damon Doucet, Predrag Gruevski, Rumen Hristov

14.12 Final Project, December 2014

Executive Summary

In this paper we use game-theoretic and empirical methods to analyze the popular board game Ascension: Chronicle of the Godslayer. We first build a simplified model of the game to show the existence of a dominant strategy equilibrium and establish dominance relationships between three classes of strategies. We then calculate a sufficient change to one of the game parameters to effectively remove the dominant strategy equilibrium condition. Finally, we verify our findings by building a simulator for the full Ascension game, implementing the necessary strategies and playing 10,000 games between every pair of strategies, confirming that the empirical results match our predictions based on the simplified model.

Our source code can be found at https://github.com/obi1kenobi/ascension-bot .

Table of Contents

What is Ascension?

Ascension: Chronicle of the Godslayer is a 2-4 player “deck-building” game with the goal of having the most victory points at the end of the game. The game ends when the pool of victory points runs out (which implies that it is a variable-length game).

There are 100 (mostly unique) cards that comprise the center deck; at any given time there will always be six of these face up in the center (when one is removed from the center, another is immediately flipped face up to replace it). Depending on the card’s type, a card in the center can either be defeated (using “**power**”) or acquired (using “**runes**”). Defeating a card provides at least one effect (specified by the card); some effects allow you to interact with opponents, e.g., by forcing them to discard a card). Acquiring a card places it into your deck for later use.

Each player starts with a deck of 10 basic cards. A turn involves drawing five cards from your own deck, playing any or all of these cards for their effects, and possibly acquiring or defeating cards in the center row. Card effects include gaining runes or power, drawing more cards from your deck, or permanently removing (“**banishing**”) a card from your deck.

Acquirable cards can either be **heroes**, which are played once and then are put in the player’s discard pile, or **constructs**, which, once played, remain in play until a card’s effect removes them (for example, an opponent defeating a certain monster in the center row). When there are no cards left in a player’s deck, the player’s discard pile is shuffled and becomes the new deck.

Each acquirable card falls into one of four categories (“**factions**”), each of which has specific strengths:

- **Void**: power and deck control (removing unwanted cards from your deck)
- **Lifebound**: runes and victory points every time a card is played
- **Enlightened**: drawing more cards
- **Mechana**: high-value cards that synergize well with each other to provide lots of runes and power, but are expensive and take a long time to ramp up

Victory points in Ascension are called **honor** and can be gained in several different ways. The most straightforward way to get honor is by defeating monsters using power – depending on the difficulty of the monster, defeating one gives 1-5 honor in addition to other bonuses, such as runes or drawing a card. Certain heroes and constructs, usually of the Lifebound faction, provide honor every time they are played or in exchange for runes. Acquired cards in a player's deck also convert to a fixed amount of honor when the game has ended. The amount of honor depends on the card.

The game ends after the turn in which the total honor gained by all players (not counting honor from acquired cards) reaches or exceeds the limit of 30 times the number of players.

A game of Ascension can have between two and four players. In this paper, we only consider the two-player version of Ascension, for two reasons:

- With fewer players, we expect the analysis will be somewhat simpler.
- Ascension is most frequently played with four players, meaning that most game balance decisions were made with four players in mind. This means that the two-player Ascension game is more likely to have balance issues, and is therefore more likely to have interesting game-theoretic results.

Why is this an interesting problem?

Ascension is a popular board game with a large fan base, and much player lore has been developed about which strategies work well, how to counter an opponent believed to be playing a particular strategy, and the general balance and pace of the game. We can use this lore to both guide our analysis in interesting directions and verify our findings by comparing our results against prevailing lore claims.

The game is sufficiently complex to make it infeasible to statically solve via exhaustive search, and the combinatorial number of synergies between cards makes it unpredictable enough to make it difficult to evaluate the payoff of an individual move. However, the contents (though not the order of cards) of each player's deck and each player's hand during their move are public knowledge. This lends itself well to game-theoretic analysis, because by knowing the contents of an opponent's deck, we can construct beliefs about their strategy, attempt to predict what they are likely to in the future, and work to use that information to improve our final payoff.

Approaches to winning the game

To win a game of Ascension, a player needs to have more total honor when the game ends than all other players. As discussed earlier, there are three different ways to gain honor – acquiring cards, defeating monsters and playing cards that give honor when played. Let us consider the high-level implications of relying on each of these approaches to getting honor in the game.

Honor from acquired cards (Mechana strategies)

As cards are acquired using runes, this approach effectively converts runes into honor. Note that cards have widely different rune costs and honor values. Furthermore, the ratio between runes spent and honor value varies between 0.25 and 1, depending on the card. The amount of honor gained using this approach is directly proportional to the number of runes spent and the conversion ratio between runes and honor, so a player would like to maximize the number of runes available each turn, and acquire cards that have a high runes-to-honor ratio.

The primary approach to maximizing the number of runes available is acquiring Mystics, a card that gives two runes every time it is played. However, this card is relatively expensive to acquire at 3 runes, and only yields 1 honor at the end of the game. This means that using runes to acquire Mystics must be carefully balanced against using runes to acquire cards with better runes-to-honor ratio in order to maximize the amount of honor at the end of the game. The correct balance depends on the number of rounds until the game ends, which makes this approach even more tricky.

A player relying on this method will generally try to acquire cards from the Mechana faction, as Mechana cards tend to have a runes-to-honor ratio close to 1. Many Mechana cards also provide benefits to acquiring other Mechana cards, such as discounts on the number of runes needed when acquiring Mechana cards. These discounts increase the effective number of runes available to the player and improve the total honor gained with this approach. This is the reason why strategies that use this approach are commonly referred to as **Mechana strategies** among Ascension players.

Honor from defeating monsters (power strategies)

A player requires power to defeat monsters, so this approach effectively converts power into honor. Ascension has a special monster called Cultist that yields 1 honor for 2 power for a 0.5 power-to-honor ratio. The Cultist is always available to be defeated, and can be defeated multiple times in a single player's turn, so its 0.5 power-to-honor ratio is an absolute lower bound on the power-to-honor ratio that can be achieved by a player. More difficult monsters have a better power-to-honor ratio than the Cultist, and sometimes also provide other beneficial effects such as allowing more cards to be drawn. The amount of honor gained with this approach is directly proportional to the amount of power spent and the conversion ratio between power and honor, so a player would like to maximize the number of power available each turn, and defeat monsters with a high power-to-honor ratio.

The primary way to increase the amount of power available each turn is to use runes to acquire Heavy Infantry, a card that costs 2 runes and provides 2 power each time it is played. Heavy Infantry is worth 1 honor at the end of the game, which results in a reasonable runes-to-honor ratio of 0.5, which is better than or equal to 54% of acquirable center-row cards, and better than Mystics' ratio. Note that unlike converting runes into honor, this approach does not require careful balancing of the acquisitions of the main resource card – runes in this approach are only used to generate more power, so the player can afford to keep acquiring more Heavy Infantry until the end of the game with little downside.

More advanced power strategies may seek the help of certain Acquirable cards that provide more power than the Heavy Infantry or provide other beneficial effects for power strategies. One such card is the Voidthirster, a construct which gives one power every turn, as well as one honor once per turn if defeating a monster in the center row.

Note that these strategies require runes to acquire the more powerful cards, so the runes in the deck must be balanced against the power as in the Mechana strategies.

Strategies that depend on defeating monsters with power are commonly referred to as **power strategies** among Ascension players.

Using cards that give honor when played (Lifebound strategies)

A set of cards, most of which are from the Lifebound faction, give honor every time they are played. This approach gets honor by acquiring such cards and then attempting to cycle through the deck in as few rounds as possible, so those cards are played as frequently as possible. To cycle through the deck quickly, this strategy heavily benefits from cards that have effects that allow the player to draw more cards or permanently remove unwanted cards from the deck. Some such cards are relatively expensive, requiring the acquisition of Mystics – but owning too many Mystics will increase the number of rounds needed to cycle through the deck, thereby decreasing the honor gained. The number of Mystics therefore needs to be carefully balanced similarly to the Mechana strategies, and depends on the rounds remaining in the game.

Strategies that predominantly use this approach to gain honor are commonly referred to as **Lifebound strategies** among Ascension players.

Combinations of the above approaches (Hybrid strategies)

A game of Ascension can also be won by combining the above approaches to form a strategy commonly referred to as a **hybrid strategy** among Ascension players. However, hybrid strategies are notoriously difficult to play because they require exquisite balance between runes, power and deck cycle length, and often heavily depend on particular rare cards appearing in the center row sufficiently early in the game to matter.

For example, the most common type of hybrid – the Mechana-power combination – is only really possible if the player is able to acquire the Hedron Cannon card. However, the Hedron Cannon is the rarest (1 out of 100 cards) and most expensive card in the game (8 runes), making it very difficult to acquire. Because of its great power, most opponents will also attempt to eliminate the card from the center row via a card effect that allows this, further diminishing the odds of success.

As hybrid strategies are so fragile and dependent on center row conditions, hybrid strategies almost exclusively occur when players who started out playing a non-combined strategy observe favorable conditions and transition to playing a hybrid strategy in response. Because of these facts, we will omit analysis of hybrid strategies and instead focus our efforts on non-combined strategies.

Theoretical Analysis

The full game of Ascension is too complex to analyze using the game theory tools we studied in class, due to its complex inter-card interactions, the large number of cards and the exponential size of the state space. In order to provide a theoretical basis for our empirical testing, we chose to create a simplified model of the Ascension game and use it to evaluate the three classes of non-hybrid strategies – power, Lifebound and Mechana.

Simplified Ascension Model

Our Ascension model made the following simplifications:

- The only card effects are producing runes or power, gaining honor every turn, gaining honor for owning the card, and drawing more cards.
- All cards in the center row (i.e. all cards excluding Mystics and Heavy Infantry) are replaced with a single type of "center row" card that has the appropriate per-card values for each effect, calculated as expectations of the value of a random card drawn from the full Ascension deck.
- Acquiring fractional cards is allowed; they produce the appropriate fraction of each effect.
- Each turn, each player gets the number of runes and power that is calculated as a function of the expected runes and power in their hand and amount of draw-cards effect they have acquired.
- Draw-card effect is a multiplier for the number of runes and power the player has, and scales exponentially. This is because in real Ascension, the more

draw-card effect cards one has in their deck, the more likely it is that drawing a card will result in getting another draw-card effect card.

We calculated all appropriate coefficients by calculating their expected values in the real Ascension game – you can find the details for all calculations in the calculator program we wrote, in `src/tools/expectation_calculator.py`.

We structured the simplified game as follows: there are two players who at the start of the game can choose between playing power, Lifebound-T or Mechana-T, where T is a small (< 50) non-negative integer parameter. We modeled the three moves to the correspond to the non-hybrid strategies in the real Ascension game, as follows:

- Power: every turn, spend all available power to gain honor by defeating the center deck card as many times as possible, and spend all available runes on acquiring Heavy Infantry.
- Lifebound-T: every turn before turn number T, spend all available runes on acquiring Mystics, to increase the amount of runes available in the deck. Every turn at or after turn number T, spend all available runes on acquiring as many center-deck cards as possible, getting draw-card effect, honor-per-turn effect and a flat amount of honor for owning the card.
- Mechana-T: every turn before turn number T, spend all available runes on acquiring Mystics, to increase the amount of runes available in the deck. Every turn at or after turn number T, spend all available runes on acquiring as many center-deck cards as possible, getting draw-card effect, and a flat amount of honor for owning the card. The center-deck card provides a larger flat amount of honor instead of providing draw-card effect, as Mechana cards have a better runes-to-honor ratio than Lifebound cards.

Once each player has chosen a move, the game proceeds by iteratively simulating each player’s turn until the total amount of honor gained from monsters or honor-per-turn effect reaches or exceeds 60 honor points. This ending condition is the same as in real Ascension.

When the game ends, each of the players gets a payoff equal to the fraction their total honor represents of the total honor gained in the game. For example, if player 1 gained X total honor, while player 2 gained Y total honor, the payoffs for the game are (X/(X+Y), Y/(X+Y)). We chose to use payoffs as fractions of 1 as that captures the winner of the game, by how much the winner won, and also allows fair comparisons between games with low total honor and games with high total honor. Using the absolute amount of honor gained as the payoff would not have captured these properties, and maximizing payoffs would produce a player that would rather gain more honor and lose than gain less honor and win, which is incorrect.

We calculated the model coefficients given in the table below as the appropriate values based on the real game of Ascension:

Coefficient name	**Value**
Starting runes per turn	4.0
Starting power per turn	1.0
Maximum runes per turn	10.0
Maximum power per turn	10.0
Honor tokens when game ends	60 or 90
Initial draw-card effect multiplier	1.0
Honor per power when defeating monsters	0.700855
Power: honor per rune when acquiring cards	0.5
Power: power increase per rune when acquiring Heavy Infantry	0.428571
Mechana: runes increase per rune when acquiring Mystics	0.454545
Mechana: honor per rune when acquiring Mystics	0.333333
Mechana: honor per rune when acquiring center cards	0.6
Mechana: rune cost per center cards	2.357.143
Mechana: draw-card effect multiplier per card	1.038.089
Lifebound: runes increase per rune when acquiring Mystics	0.454545
Lifebound: honor per rune when acquiring Mystics	0.333333
Lifebound: honor per rune when acquiring center cards	0.487179
Lifebound: rune cost per center cards	2.357.143
Lifebound: draw-card effect multiplier per card	1.038.089

Figure 1 – Table of simplified model coefficients

Results and Analysis

For ease of analysis, we implemented the game described above in Python (source code can be found in `src/tools/strategy_simulator.py`). We then used the simulator to calculate the payoffs for every pair of moves. The full payoff matrix is too large to reproduce here – it is 101x101 in size – but it can be generated by running the code above. Instead, we will discuss our findings based on the calculated payoff matrix.

Overall Game Balance – Dominant Strategies

A key property of a good board game is that it is "**balanced**" – that no dominant strategy exists that will achieve at least a payoff of 0.5 against any other strategy. Ascension lore claims that Ascension is somewhat unbalanced for 2 players – we wanted to put this claim to the test. Our payoff matrix showed that **playing power is a dominant strategy** that always achieves a payoff of at least 0.5 in the 2 player case, as the following table shows:

P2 strategy when P1 plays power	Payoffs for (P1, P2)
Overall best response: power	(0.5, 0.5)
Best Mechana response: Mechana-5	(0.561, 0.439)
Best Lifebound response: Lifebound-3	(0.557, 0.443)

Figure 2 – Payoffs of various strategies when P1 is playing power

The following figures show the total honor at the end of each turn for the power strategy versus the best Mechana response and the best Lifebound response:

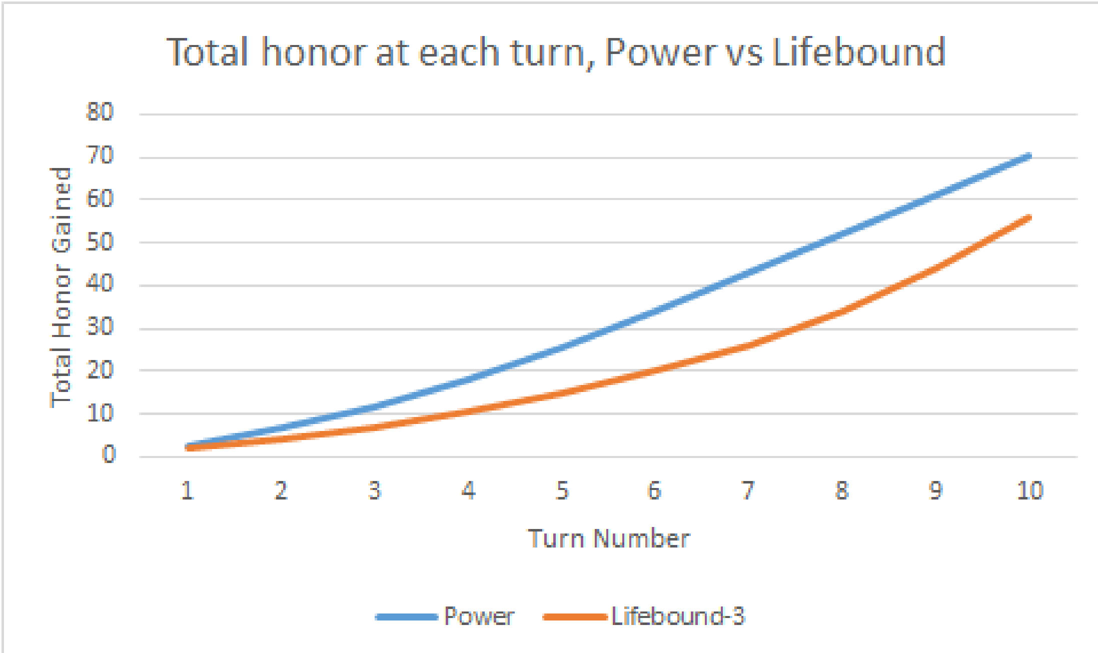

Figure 3 – The total honor after each game turn for the power and Lifebound-3 strategies. The game ended after turn number 10.

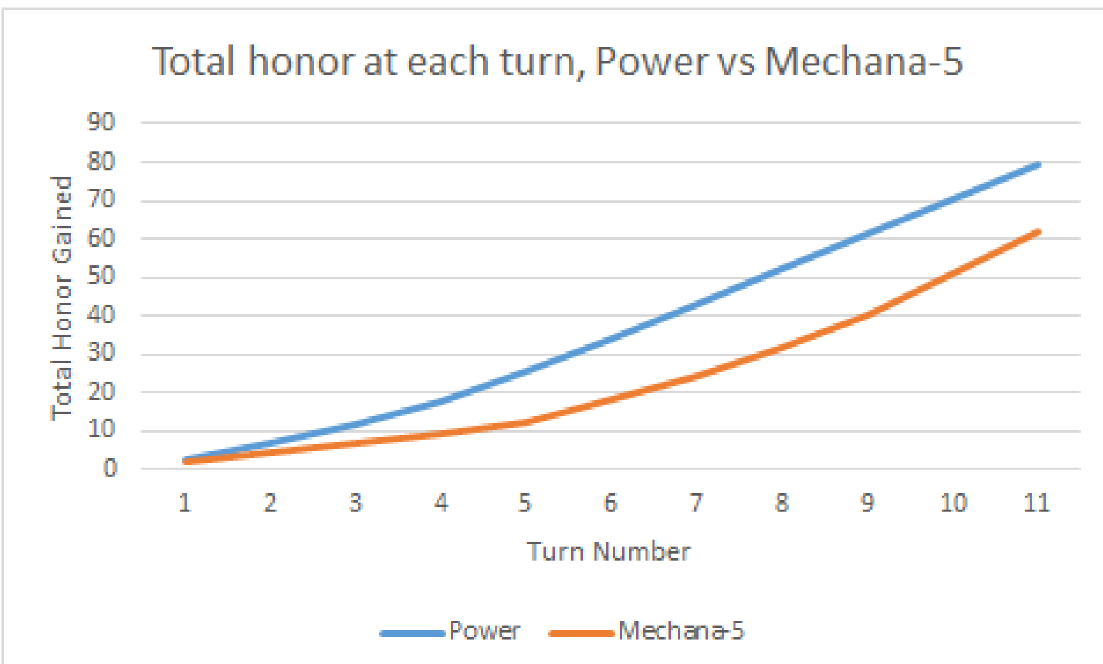

Figure 4 – The total honor after each game turn for the power and Mechana-5 strategies. The game ended after turn number 11.

Our model therefore showed that we should expect that Ascension is **unbalanced** for two players, as playing power is expected to result in a tie or win against any strategy. To maximize payoffs, a rational Ascension player would therefore always play power. Ascension lore further suggests that power is weakened in longer games, as Mechana and Lifebound get progressively more and more honor every turn whereas power's amount of honor gained every turn flattens out relatively early. We used this insight to perform a parameter search for an amount of honor needed for the game to end such that the game becomes as close as possible to being balanced. Our parameter search showed that increasing this parameter from 60 to 90 would make the game essentially balanced, with the following payoffs:

P2 strategy when P1 plays power	Payoffs for (P1, P2)
Overall best response: power	(0.5, 0.5)
Best Mechana response: Mechana-6	(0.517, 0.483)
Best Lifebound response: Lifebound-6	(0.5004, 0.4996)

Figure 5 – Payoffs of various strategies when P1 is playing power, with 90 total honor required before the game ends

The following figures show the total honor at the end of each turn with the above parameter modification, for the power strategy versus the best Mechana response and the best Lifebound response:

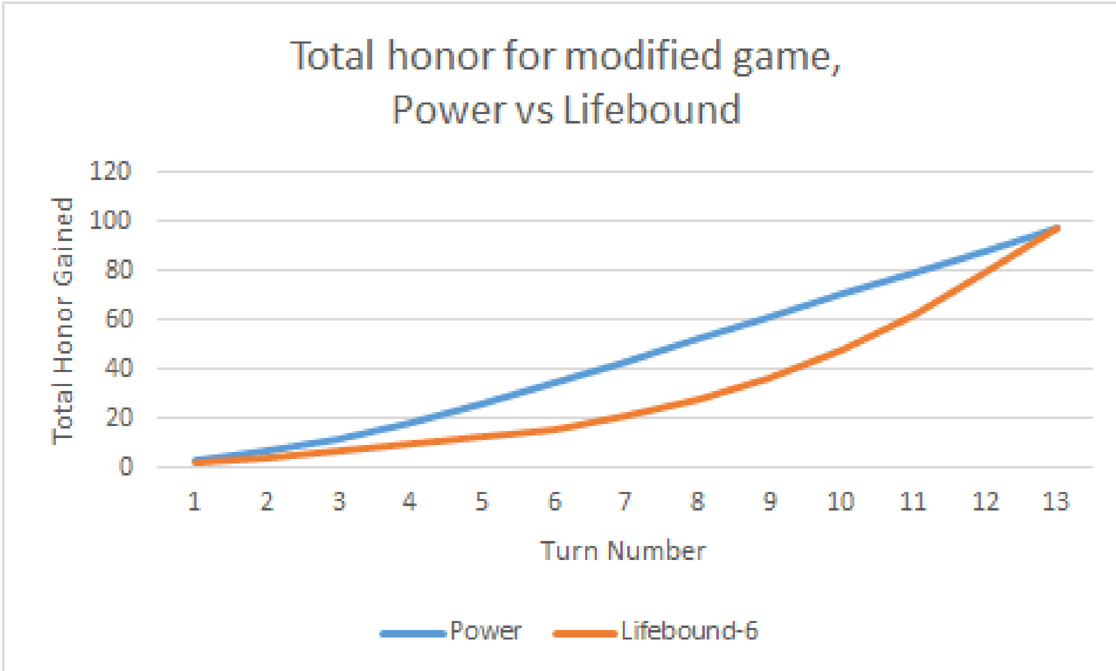

Figure 6 – The total honor after each game turn, with the parameter modification, for the power and Lifebound-3 strategies. The game ended after turn number 13.

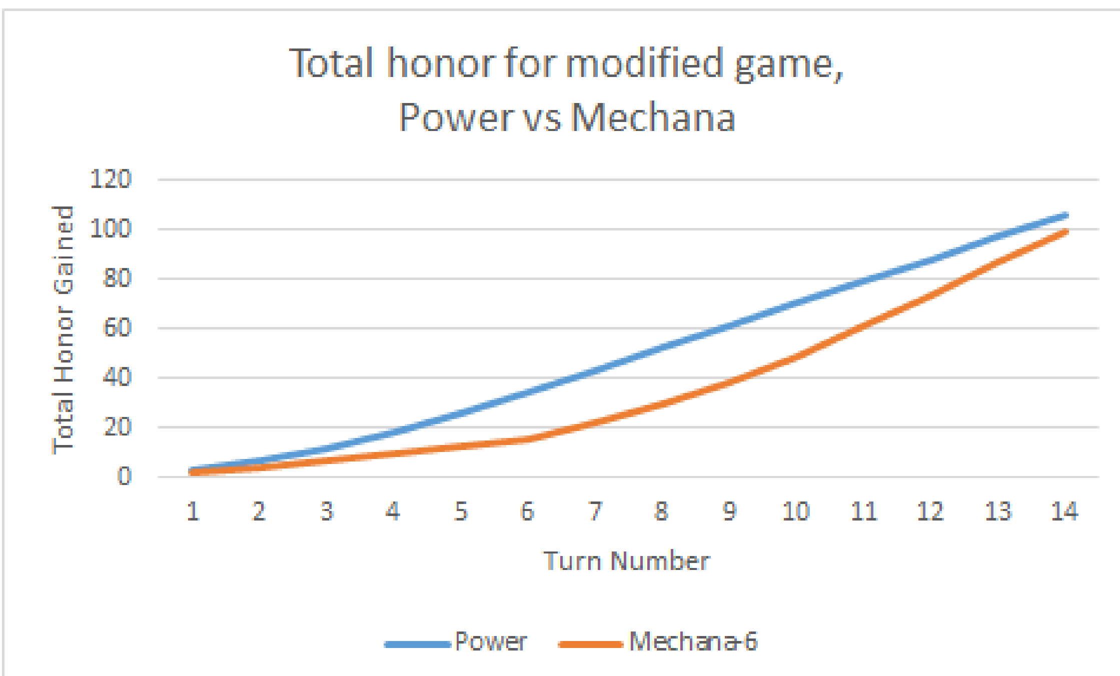

Figure 7 – The total honor after each game turn for the power and Mechana-5 strategies. The game ended after turn number 14.

After we did this analysis, we found that the Ascension game publisher made changes to the game in an expansion pack – changing the amount of honor when the game ends for two players from 60 to 90. This is the exact value our model suggested would balance the game. We believe that this fact demonstrates the soundness of our model.

Balance of Lifebound vs. Mechana – Dominance and Nash Equilibria

We were also interested in determining whether Mechana and Lifebound are balanced – i.e. whether some Nash equilibrium, when one player is forced to play a Mechana move while the other is forced to play a Lifebound move, yields payoffs that are close to being equal, (0.5, 0.5). To do this, we simply used the calculated payoff matrix and looked for Mechana-Lifebound move pairs in which neither player could improve their payoff by changing the T-parameter of their move.

We found that the matrix contained only a **single Nash equilibrium of this kind**, between Mechana-6 and Lifebound-6, with payoffs (0.447, 0.553). This suggests that Lifebound and Mechana are not perfectly balanced either, with playing a Lifebound move being the slightly better choice. This shows that **playing Lifebound dominates playing Mechana.**

Real Ascension

Game State and Transitions

Even though the real Ascension game is much more complicated than the games studied in class, it has similar properties. The game state can be defined by several values and several sets of cards. For each player, we have an amount of honor as well as the set of cards in the player's hand, discard pile, deck and constructs. There is also state common to both players: the cards in the center of the board, the number of Mystics and Heavy Infantry available, what cards are in the board deck and the amount of honor left. These describe the state of the game in its entirety. Note that because all players observe all moves, each player's deck composition is common knowledge.

The game is structured as an iterated game that ends after a number of rounds that is not known ahead of time, but can be estimated during the game. At each iteration, each player draws five cards at random from his deck and chooses a set of actions to perform: play cards from his hand, defeat monsters, acquire cards or activate constructs. Players play in a predefined order and observe each others' moves; played and acquired cards go to the player's discard pile. When a player's deck runs out, his discard pile is shuffled and becomes his deck. Whenever a card from the center row is acquired or defeated, a replacement card is drawn at random from the center deck.

We have a finite (although impractically large) number of states so in theory we could consider all of them and create a game tree. There are two kinds of transitions between states – player moves and random transitions. Player move transitions occur when a player performs their set of actions and ends their turn. Random transitions happen when cards are drawn at random from any player's deck or the center deck.

Expectations

As the game-theoretical analysis of our simplified Ascension model showed that playing power is a dominant strategy, we expect that a power strategy in the full Ascension game would beat both Lifebound and Mechana strategies. Furthermore, the the analysis showed power getting a higher payoff against Mechana than against Lifebound, so we expect a higher win percentage of power against Mechana than against Lifebound in a real game of Ascension. We therefore expect that power will also be a dominant strategy in the full Ascension game. Finally, as the Nash equilibrium between Lifebound and Mechana moves showed a higher payoff for Lifebound than for Mechana, we expect that a Lifebound strategy will, on average, beat the Mechana strategy in the full Ascension game.

Empirical Testing

In order to verify our game-theoretic analysis of the simplified Ascension game, we built a simulator for the real Ascension game and implemented the same strategies we analyzed in the simplified model. In the next subsections, we discuss how the simulator code is implemented, the details of our strategies, our analysis methods and our findings.

Code Overview

We wrote 4227 lines of code for the simulator, all of which is available on Github using the following link: https://github.com/obi1kenobi/ascension-bot

Input Files

First, each card was broken into a set of effects, e.g., gain runes, or banish a card from the center row. The set of all effects is in `src/input/effects.txt`, where some effects have parameters (e.g., gain 1 rune as opposed to gain 2 runes). The cards reference these effects by their line number in the input file.

Next, the cards were broken into two categories: those that can be acquired (heroes and constructs; those purchased with runes), and those that can be defeated (monsters; defeated with power). There is a separate input file for each of these (`src/input/acquirables.csv` and `src/input/defeatables.csv`). The input files describe the name of the card, how much the card costs, and any effects that occur when the card is played or defeated. In addition, for cards that can be acquired, how much honor the card is worth at the end of the game is also defined.

In Ascension, cards' effects can combine in many different ways; some effects may be optional, others require the use of one effect to activate a second, and others require that at most one of several effects be activated. The input files describe the effects using a series of ANDs, ORs, and ?s (representing optional effects). In addition, since many effects require parameters, those are specified in function notation.

An example card is shown below:

```
Shade of the Black Watch,3R,1,Void Hero,AND(5(2),OR(6(1);8(1))?)
```

Here, the "Shade of the Black Watch" costs 3 runes, is worth 1 honor at the end of the game, is a "Void Hero", and has a compound effect. (Note that effect 5 is gaining power, effect 6 is banishing a card from your hand, and effect 8 is banishing a card from your discard pile.) The effect of the card as described in the input line is "gain 2 power and optionally either banish a card from your hand or from your discard pile."

Card Decoder

The `card_decoder` module is responsible for reading all of the input files and parsing the cards and effects into usable pieces for the code. It defines the objects to be used through the rest of the application, broken into several files.

In `src/card_decoder/effects.py` are the definitions for the two types of effects: Simple and Compound. A Compound effect is one that combines at least two effects using AND or OR. A Simple effect is one of the effect indices as described in `src/input/effects.txt`. Both Simple and Compound effects may or may not be optional. In addition, Simple effects may have a parameter. Effects also provide methods useful for generating moves (described later): generating all legal combinations of the effects, in addition to getting the parameter for a given effect index in a compound effect.

In `src/card_decoder/cards.py` are definitions for several objects. First is a dictionary that simply looks up cards by name. The next is the `Card` class, which holds a cost, a card type, and an effect (which may be simple or compound). This class is extended as an `Acquirable` and a `Defeatable` to represent the two types of cards.

Finally, there are two suites of unit tests. One, `src/card_decoder/test_effects.py`, for testing that parsing the compound effects works correctly; and the other, `src/card_decoder/test_decoder.py`, that ensures that the decoder works as intended.

Gameplay

The file `src/game.py` is a very simple wrapper around playing a game. It exports a single `play_game` method, which takes strategies (see below), creates a board, and continually requests them to play moves. It returns the final board state, which includes the winner and how much honor the players had at the end of game.

The `Board` class (`src/board.py`) represents the current board state, including the center row, any banished cards, and the current player states.

Player state is defined in `src/player.py`. This includes the player's current deck state, discard pile, constructs, and hand. It also holds data about specific effects (like having bonuses toward acquiring specific types of cards).

The game state (board and player states) is modified through the use of a `Move`, which is defined in `src/moves.py`. A move can be one of acquiring a hero or construct, defeating a monster, playing a hero or construct, or activating a construct already in play. This file takes holds logic for updating general board state (like paying for cards).

The moves defer any effect activation, however, to `src/effects.py`, which exports a single function, `apply_simple_effect`. This maps each effect index to a function that modifies the board state.

Move Generation

There are two components to move generation: deciding which moves can be played, and deciding the legal targets of effects activated through those moves.

Generating the legal targets for an effect happens in `src/legal_targets.py`. This is similar to `src/effects.py` in that it maps an effect index to a function that reports the legal targets of that effect. Some effects do not have targets, in which case the function returns an empty object. Note, however, that not having a target must be distinct from not being able to be played.

Moves are composed of a move type (acquire, defeat, play, activate), a specific card (e.g., which card is being acquired, or played), and a set of targets if the move type is not acquisition. Move generation, for each move type, looks at the possible cards that could have the move type used on them; additionally, if the move type is not acquisition, move generation uses the legal target generation. These two together give us all possible moves a player can play at any given time, which is useful for evaluating moves from a strategy's perspective.

Estimators

There are several estimators at the disposal of the strategies for checking various statistics about the game in order to make better decisions. Metrics tracked include the amount of honor a player's deck is currently worth, how many of each card a player has, a guess at how many turns are left until the end of the game (which helps for thinking about which strategy to run), how many cards a player generally has enter his hand during a turn (e.g., through draw-card effects), and from that, how often the deck cycles. Code for this exists in the `src/estimators` directory.

Strategies

We created several different strategies as part of developing the game code. The first was a basic strategy (`src/strategies/basic_strategy.py`) that simply acquires as many Heavy Infantry as possible and kills the Cultist as many times as possible. This was used to ensure that the basics of the game worked correctly. Next was a random strategy, which simply looked at all legal moves and chose a random one to play. This was used to ensure that the rest of game worked correctly.

Next, we made a control strategy for testing against, the greedy strategy (`src/strategies/greedy_strategy.py`). This plays all of its cards and then acquires or defeats the card worth the most honor.

Our next three strategies were tested against the control strategy and used for generating payoff matrices.

The first is a straightforward power strategy. Just as the strategy from the simplified model, this strategy simply spends all runes to acquire Heavy Infantry and defeats the monsters worth the most honor. This is what is colloquially known as a “rush” strategy, as it tries to simply gain honor from monsters and end the game quickly, before a more complex strategy can gain sufficient honor to win.

The second is a Lifebound strategy, which spends runes on cards that generate honor when played, and cards that make the honor-generating cards be drawn more often. This strategy hopes to cycle through its deck quickly and repeatedly play cards that generate honor to win.

The last is a Mechana strategy, which relies on acquiring valuable and very powerful cards, but requires a longer time to acquire sufficient runes for those cards. The strategy hopes to prolong the game until the synergies between its powerful cards allow it to gain more honor than its opponents.

Autotesting Methods

We wanted to test our game-theoretic conclusions in the simplified model through empirical results in the full game. This was done by automatically simulating 20,000 games for each pair of strategies (10,000 with one as first player; another 10,000 with the other as first player) and observing statistics such as victory percentage, tie percentage, and average margin of victory.

Autotesting Results

Some of the results of the autotesting are shown in the image below:

p1\p2	lifebound	power	mechana
lifebound	(0.544,0.021)	(0.272,0.025)	(0.654,0.014)
power	(0.697,0.026)	(0.527,0.100)	(0.804,0.020)
mechana	(0.328,0.013)	(0.169,0.019)	(0.545,0.015)

Figure 8 – Autotesting results averaged over 10,000 games, formatted as (P1 win ratio, tie ratio)

Remarkably, the power strategy does not have a losing matchup; against Lifebound it wins almost 70% of the time, and against Mechana, it wins 80% of the time. In fact, the best response to power is to play power, suggesting that **the power strategy is a dominant strategy**, and the **only Nash equilibrium of the game** is achieved when **both players play a power strategy**. **The Lifebound strategy also dominates the Mechana strategy**, winning 65% of the time. This is likely caused by the fact that more cards in the deck benefit the Lifebound strategy than the Mechana strategy, as well as the fact that the power strategy can always kill the Cultist to gain honor each turn. These results are **consistent with our findings in the simplified model,** where we predicted that in two-player Ascension, playing power is a dominant strategy and the only Nash equilibrium of the game, and playing the Lifebound strategy dominates playing the Mechana strategy.

Conclusion

In this paper, we showed that the two-player variant of the popular board game of Ascension is not well-balanced. We developed a simplified model of the game of Ascension and used it to draw the following conclusions. Of the three primary ways to get “honor” (victory points), using “power” to defeat monster cards is a dominant strategy and the only Nash equilibrium of the game. Furthermore, a good strategy that uses cards that give “honor” when played (“Lifebound” cards) is dominant compared to a good strategy that buys valuable cards (“Mechana” cards). We were able to verify the validity of our game-theoretic findings by building a full Ascension game simulator and ensuring that the implementations of our strategies indeed perform as predicted by the simplified model.

Finally, we were able to use the theoretical model to derive conditions under which the game balance is significantly improved, allowing strategies based on all three approaches to win with approximately equal probability. After this analysis, we discovered that an update to the rules in a recent expansion to the Ascension game implemented the exact change we calculated would make the game more balanced, which further verifies our results.