Game Theory and U.S. Senate Campaigns

Tim Wetzel

Mathematics Department
California Polytechnic State University
San Luis Obispo, California

March 20, 2019

Approval Page

Title: Game Theory and U.S. Senate Campaigns

Author: Tim Wetzel

Date: March 20, 2019

Senior Project Advisor

Department Chair

Advisor Signature

Chair Signature

Abstract

A mathematical model is developed for deciding when a candidate running for a United States Senate seat should spend campaign funds. The Senate race is modeled as a zero-sum game. This requires finding player strategies and generating player utilities based on those strategies.

Applying least-squares regression to FEC spending and fundraising data from the 2016 U.S. Senate race, a fundraising return function is defined which is used to generate possible strategies. Several different utility functions are then calculated, using FEC spending data from the 2018 U.S. Senate race and straw poll data collected by the website FiveThirtyEight.com.

The strategy space turns out to be quite large, and it was not possible to solve the largest games directly using the standard methods from linear programming. Therefore the model was tested using a combination of linear programming and Monte-Carlo methods. These simulations yield strategy solutions for both players.

1 Introduction

When determining optimal campaign spending strategies, most strategists focus on location, demographics, and medium; much time and paper has been spent researching these aspects of spending. An often overlooked aspect of a campaign, and the main focus of this project, is *when* in the campaign to spend. A model is created using FEC senate campaign fundraising and expenditure receipts, straw poll data from FiveThirtyEight, game theory, and linear regression to create a weighted Blotto model of campaign spending, focusing on which quarters to spend.

2 Model Discussion

2.1 Game Theory

The following definitions are used throughout the report:

- **Pure Strategy** — A pure strategy is a strategy that requires no randomization or deviation from a single strategy. A pure strategy in poker, for example, is to fold every hand and never bet. This strategy might look like $[1.0, 0.0]$, with 1.0 referring to the proportion of hands a player should fold and 0.0 referring to the proportion of hands a player should bet on.

- **Mixed Strategy** — A mixed strategy is one that calls for the employment of more than one strategy, and specifies how often each strategy should be used. One mixed strategy in poker, for example, is to fold 30% of hands while betting on the other 70%. This strategy might look like $[0.3, 0.7]$, with 0.3 referring to the percentage of hands a player should fold and 0.7 referring to the percentage of hands a player should bet on. The sum of a mixed strategy's elements should be 1.

- **Strategy Vector** — A strategy vector is a vector in which *each element* corresponds to one strategy a player can employ; these may be any combination of **mixed strategies** and **pure strategies**. To use the above examples, a strategy vector for poker containing the two vectors above would look like

$$\begin{bmatrix} [1.0, 0.0] & [0.3, 0.7] \end{bmatrix}.$$

In this report, each strategy in the strategy vector will contain four elements. These refer to the amount a player will spend in each quarter of a simulated campaign season. The strategy vectors, then, will be of size $n \times 4$, where n refers to the number of strategies, and 4 refers to the four quarters of the simulated election season. Each player will have a corresponding strategy vector.

- **Zero-Sum Game** — This is a game in which the sum of all payoffs for all players is zero. In a two player zero-sum game, the payoffs can be represented by a single number. This single number x will mean that the first player receives a payoff of x and the second player receives a payoff of $-x$.

- **Value of a Game** — The value of a two player zero sum game is the expected payoff of the game when both players play optimal strategies.

- **Payout Matrix** — A payout matrix the matrix with rows indexed by the first players strategies, columns indexed by the second players strategies, and (row, column) entry equal to the payoff when (row,column) is played by the first and second players.

  Consider a payout matrix M, corresponding to a game between players P1 and P2:

$$\begin{array}{cc} & \begin{bmatrix} t_1 & t_2 & t_3 \end{bmatrix} \\ M = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix} & \begin{bmatrix} -1 & 2 & -1 \\ 1 & 0 & 0 \\ -1 & 1 & 1 \end{bmatrix}. \end{array}$$

  This matrix is 3×3, since each player has 3 possible strategies. Rows of the matrix correspond to possible payouts given any strategy of P1 (s_1, s_2, or s_3 in this case) while columns correspond to possible payouts given any strategy of P2 (t_1, t_2, and t_3 in this case). Therefore, if $m_{j,k}$ refers to an element of M, then $m_{j,k}$ refers to the value of playing these current strategies. A positive number is good for P1 and bad for P2.

- **Optimal strategies** A strategy S is optimal if a player who uses S will earn a least the value of the game, regardless of the actions taken by his opponent. Optimal solutions can be found using **linear programming**.

The next section outlines the basics of linear programming, and shows how it can be used to find optimal strategies.

2.2 Finding a Game Value

Linear programming can be used to find a solution to a matrix game.

For example, consider the following strategy vectors and associated payout matrix M, of a zero-sum game between P1 and P2:

$$\begin{array}{cc} & \begin{bmatrix} t_1 & t_2 & \dots & t_j & \dots \end{bmatrix} \\ \begin{bmatrix} s_1 \\ s_2 \\ \dots \\ s_i \\ \dots \end{bmatrix} & \begin{bmatrix} m_{1,1} & m_{1,2} & \dots & m_{1,j} & \dots \\ m_{2,1} & m_{2,2} & \dots & m_{2,j} & \dots \\ \dots & \dots & \dots & \dots & \dots \\ m_{i,1} & m_{i,2} & \dots & m_{i,j} & \dots \\ \dots & \dots & \dots & \dots & \dots \end{bmatrix} \end{array}$$

The **s** vector denotes every possible strategy P1 can employ, while the **t** vector denotes every possible strategy P2 can employ. Row i gives the payoffs when P1 plays strategy i and column j gives the payoffs when P2 plays strategy j.

If P1 plays the pure strategy s_i and P2 plays the pure strategy t_j, then the value of the game is simply the corresponding matrix element, $m_{i,j}$. These pure strategies can be represented as vectors with a 1 in the played strategy and 0's elsewhere. The value of the game can now be solved using matrix multiplication, by finding $s^{*T} M t^*$. In this example, the resulting equation would look like

$$v = \begin{bmatrix} 0.00 & 0.00 & ... & 1.00 & ... \end{bmatrix} \begin{bmatrix} m_{1,1} & m_{1,2} & ... & m_{1,j} & ... \\ m_{2,1} & m_{2,2} & ... & m_{2,j} & ... \\ ... & ... & ... & ... & ... \\ m_{i,1} & m_{i,2} & ... & m_{i,j} & ... \\ ... & ... & ... & ... & ... \end{bmatrix} \begin{bmatrix} 0.00 \\ 0.00 \\ ... \\ 1.00 \\ ... \end{bmatrix} = m_{i,j}$$

The value of a game in which players employ mixed strategies can be calculated in a similar way. Suppose, instead of only playing strategy s_i, P1 played each of their strategies in different proportions. Then s^* would be populated with decimals denoting, proportionally, how often P1 chose to employ each strategy. For each strategy s_i, denote the proportion of turns P1 uses strategy s_i as p_i. Similarly, denote the proportion of turns P2 uses strategy t_j as q_j. The value of the game is

$$v = \begin{bmatrix} p_1 & p_2 & ... & p_i & ... \end{bmatrix} \begin{bmatrix} m_{1,1} & m_{1,2} & ... & m_{1,j} & ... \\ m_{2,1} & m_{2,2} & ... & m_{2,j} & ... \\ ... & ... & ... & ... & ... \\ m_{i,1} & m_{i,2} & ... & m_{i,j} & ... \\ ... & ... & ... & ... & ... \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ ... \\ q_j \\ ... \end{bmatrix}$$

2.3 Linear Programming

Linear programming can be used to find optimal strategies and the value of a matrix game M. Consider, as an example, the following payout matrix:

$$\begin{bmatrix} t_1 & t_2 & t_3 \end{bmatrix}$$

$$M = \begin{bmatrix} s_1 \\ s_2 \\ s_3 \end{bmatrix} \begin{bmatrix} -1 & 2 & -1 \\ 1 & 0 & 0 \\ -1 & 1 & 1 \end{bmatrix}$$

In this matrix, P1 employs strategies corresponding to rows of M, P2 employs strategies corresponding to columns of M. P1's goal is to maximize the game's value, while P2's goal is to minimize it. The matrix is written as follows:

M			
	t_1	t_2	t_3
s_1	-1	2	-1
s_2	1	0	0
s_3	-1	1	1

To find the solution to the matrix game, the Pivot method may be used. To begin the algorithm, since there are no initial constraints on the strategies, arbitrary constraints will be assigned. The purpose of this is to track what happens to each row and column, which will ultimately solve for how often to use each strategy. To do this, create a column of 1's on the far right side of M, and a row of -1's on the bottom of M. Additionally, add a zero in the new bottom corner of the matrix. This creates the initial tableau:

M				
	t_1	t_2	t_3	
s_1	-1	2	-1	1
s_2	1	0	0	1
s_3	-1	1	1	1
	-1	-1	-1	0

In order to use the simplex algorithm to solve the game, the value of the game must be positive. Therefore, add a constant to each element to ensure this. In this case, add 2, so that every value is positive. Don't forget this step was taken; it's crucial to fix it at the end.

M				
	t_1	t_2	t_3	
s_1	1	4	1	1
s_2	3	2	2	1
s_3	1	3	3	1
	-1	-1	-1	0

The next step is called a pivot operation. First, a pivot value must be picked. Any number in M (the original game matrix, not along the added edges) may be picked, as long as it satisfies the following constraints:

1. The value in the added bottom row corresponding to the pivot's column MUST be negative. (This is trivial in the first step, since all the numbers on the bottom are -1.)

2. The pivot itself must be positive.

3. The pivot row must be chosen to minimize the ratio: $\frac{\text{Number in added column corresponding to pivot's row}}{\text{Pivot}}$ among the positive pivots in that column

For example, suppose a pivot is being picked for the t_1 column (this is possible because the value at the bottom of the t_1 column is -1, which is negative). It may be any of 1, 3, or 1, since these are the positive values in this column. Since the values on the very right column for all these numbers are all 1, compare the ratios $\frac{1}{1}$, $\frac{1}{3}$, and $\frac{1}{1}$. The smallest of these is $\frac{1}{3}$; therefore, 3 is the first pivot. Circle the number 3 to indicate this.

M				
	t_1	t_2	t_3	
s_1	1	4	1	1
s_2	3	2	2	1
s_3	1	3	3	1
	-1	-1	-1	0

Denote each element of M as $m_{i,j}$, where i refers to the row and j refers to the column. Let i_p and j_p denote the row and column of the pivot; in this case, $i = 1$ and $j = 1$. First, for each $m_{i,j}$ were $i \neq i_p$ and $j \neq j_p$ (all elements not in the same row or column as the pivot), replace $m_{i,j}$ with

$m_{i,j} - \frac{m_{i,j_p} m_{i_p,j}}{p}$. This would look like:

M				
	t_1	t_2	t_3	
s_1	1	$\frac{10}{3}$	$\frac{1}{3}$	$\frac{2}{3}$
s_2	3	2	2	1
s_3	1	$\frac{7}{3}$	$\frac{7}{3}$	$\frac{2}{3}$
	-1	$-\frac{1}{3}$	$-\frac{1}{3}$	$\frac{1}{3}$

Then replace all elements m_{i,j_p} in the pivot column (except the pivot itself) with $-\frac{m_{i_p,j}}{p}$:

M				
	t_1	t_2	t_3	
s_1	$-\frac{1}{3}$	$\frac{10}{3}$	$\frac{1}{3}$	$\frac{2}{3}$
s_2	3	2	2	1
s_3	$-\frac{1}{3}$	$\frac{7}{3}$	$\frac{7}{3}$	$\frac{2}{3}$
	$\frac{1}{3}$	$-\frac{1}{3}$	$-\frac{1}{3}$	$\frac{1}{3}$

Next, replace all elements $m_{i_p,j}$ in the pivot row (except the pivot itself) with $\frac{m_{i_p,j}}{p}$:

M				
	t_1	t_2	t_3	
s_1	$-\frac{1}{3}$	$\frac{10}{3}$	$\frac{1}{3}$	$\frac{2}{3}$
s_2	3	$\frac{2}{3}$	$\frac{2}{3}$	$\frac{1}{3}$
s_3	$-\frac{1}{3}$	$\frac{7}{3}$	$\frac{7}{3}$	$\frac{2}{3}$
	$\frac{1}{3}$	$-\frac{1}{3}$	$-\frac{1}{3}$	$\frac{1}{3}$

Then replace the pivot p with $\frac{1}{p}$:

M				
	t_1	t_2	t_3	
s_1	$-\frac{1}{3}$	$\frac{10}{3}$	$\frac{1}{3}$	$\frac{2}{3}$
s_2	$\frac{1}{3}$	$\frac{2}{3}$	$\frac{2}{3}$	$\frac{1}{3}$
s_3	$-\frac{1}{3}$	$\frac{7}{3}$	$\frac{7}{3}$	$\frac{2}{3}$
	$\frac{1}{3}$	$-\frac{1}{3}$	$-\frac{1}{3}$	$\frac{1}{3}$

Last, switch the labels in the same row and column as the pivot (in this case, s_1 and t_1):

M				
	s_2	t_2	t_3	
s_1	$-\frac{1}{3}$	$\frac{10}{3}$	$\frac{1}{3}$	$\frac{2}{3}$
t_1	$\frac{1}{3}$	$\frac{2}{3}$	$\frac{2}{3}$	$\frac{1}{3}$
s_3	$-\frac{1}{3}$	$\frac{7}{3}$	$\frac{7}{3}$	$\frac{2}{3}$
	$\frac{1}{3}$	$-\frac{1}{3}$	$-\frac{1}{3}$	$\frac{1}{3}$

Step 1 is now complete! Repeat this process, picking a new pivot, until there are no more negative values on the added bottom row. Once this is complete, the only thing that matters is the added row and column, which indicate optimal strategies for each player. When this matrix is solved, it looks like:

M				
	s_2	s_1	s_3	
t_2	$-\frac{2}{21}$	$\frac{1}{3}$	$-\frac{1}{21}$	$\frac{4}{21}$
t_1	$\frac{3}{7}$	0	$-\frac{3}{7}$	$\frac{1}{7}$
t_3	$-\frac{1}{21}$	$-\frac{1}{3}$	$\frac{10}{21}$	$\frac{2}{21}$
	$\frac{2}{7}$	0	$\frac{1}{7}$	$\frac{3}{7}$

To simplify the solution, here's a more straightforward version of the matrix:

M				
	s_2	s_1	s_3	
t_2	*	*	*	$\frac{4}{21}$
t_1	*	*	*	$\frac{1}{7}$
t_3	*	*	*	$\frac{2}{21}$
	$\frac{2}{7}$	0	$\frac{1}{7}$	$\frac{3}{7}$

The only numbers needed from this matrix are those on the added bottom row corresponding to s values, those on the added far right column corresponding to t values, and the value of the game on the bottom right. Any s or t strategy that did not switch axes would now be set to zero (there were none here, but that's not always the case).

But what do these numbers mean? To find out, divide each number by the value at the very bottom right ($\frac{3}{7}$, in this case). After this, subtract the 2 that was added to the value at the beginning of the game. This gives

M				
	s_2	s_1	s_3	
t_2	*	*	*	$\frac{4}{9}$
t_1	*	*	*	$\frac{1}{3}$
t_3	*	*	*	$\frac{2}{9}$
	$\frac{2}{3}$	0	$\frac{1}{3}$	$-\frac{11}{7}$

The number across from each strategy now denotes what percentage of turns a player should employ that strategy. The value at the bottom right of the matrix now denotes the average value of the game, if each player plays optimally. Therefore:

- To play optimally, P1 should play s_1m 0% of the time, s_2 66.67% of the time, and s_3 33.33% of the time.
- To play optimally, P2 should play t_1m 33.33% of the time, t_2 44.44% of the time, and t_3 22.22% of the time.
- If both players employ these strategies over an infinite number of games, the average value of these games will be $-\frac{11}{7}$ (in favor of P2).

For more information on why this algorithm works, see [3].

2.4 Colonel Blotto Problem

The campaign trail is often compared to a battlefield because of its zero-sum nature and the tactics involved in winning. It makes sense, then, that games used to model battle strategies can also be used to model campaign strategies.

One model is given by the Colonel Blotto Problem. There lives an acclaimed strategist named Colonel Blotto (actually there doesn't, but for some reason that name has stuck in the math world). Colonel Blotto is in the final dregs of a battle against his rival, Commodore Hopper [9] (the name of his opponent varies by author). To win the battle, Colonel Blotto has to defeat his opponent in the last 2 remaining battlefields, f_1 and f_2. To do this, he had to find a way to allot his 4 armies between the two battlefields. His opponent, Commodore Hopper, has a similar charge — however, Commodore Hopper only has 3 armies.

The utility function was defined as follows. Let b_1 be the number of armies Colonel Blotto allots to f_1, and b_2 be the number of armies he allots to f_2. Similarly, let h_1 be the number of armies Commodore Hopper allots to f_1 and h_2 be the number of armies she allots to f_2. This yields different strategy vectors for each player,

$$b = \{(4,0),(3,1),(2,2),(1,3),(0,4)\}, \tag{1}$$

$$h = \{(3,0),(2,1),(1,2),(0,3)\}, \tag{2}$$

where b is the strategy vector for Colonel Blotto and h is the strategy vector for Commodore Hopper. Each element of the strategy vector is a **tuple** (or finite, ordered list) of size 2, consisting of (b_1, b_2) for Colonel Blotto and (h_1, h_2) for Commodore Hopper. In the most basic Blotto model, the utility function $m_{i,j}$ is defined as 1 if Blotto wins the most battles, -1 if Hopper wins the most battles, and 0 if they win the same number of battles. Informally, this can be written:

$$m_{i,j} = \begin{cases} 1, & \text{if Blotto wins more battles than Hopper} \\ 0, & \text{if Blotto and Hopper win the same number of battles} \\ -1, & \text{if Hopper wins more battles than Blotto} \end{cases}$$

This can be written more mathematically as well, as follows:

$$m_{i,j} = \begin{cases} 1, & \text{if } (b_1 - h_1) + (b_2 - h_2) > 0 \\ 0, & \text{if } (b_1 - h_1) + (b_2 - h_2) = 0 \\ -1, & \text{if } (b_1 - h_1) + (b_2 - h_2) < 0 \end{cases}$$

This yields the following payout matrix, with rows denoting Blotto's strategies and columns denoting Hopper's:

$$\begin{array}{cc} & \begin{bmatrix} (3,0), (2,1), (1,2), (0,3) \end{bmatrix} \\ \begin{bmatrix} (4,0) \\ (3,1) \\ (2,2) \\ (1,3) \\ (0,4) \end{bmatrix} & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}.$$

In this case, Hopper does not win in any case; however, even with fewer troops, she can tie Blotto overall by winning one of the two battles. While this specific payout matrix has no negative values, this is not necessarily true in a more generalized game.

This game can be written in a similar but more nuanced way. Rather than using the utility function as above, payouts can be weighted with the differential by which a battle was won. This gives more points for a strong win, rather than a single value for all battles won. This problem is called a **Weighted Blotto Model**, and its utility function is

$$m_{i,j} = (b_1 - h_1) + (b_2 - h_2)$$

If (1) and (2) are used, m_i, j is equal to 1 for every element of the matrix (since $(b_1 - h_1) + (b_2 - h_2) = (b_1 + b_2) - (h_1 - h_2) = b - h = 5 - 4 = 1$), giving a relatively uninteresting game. However, suppose the two leaders are given more constraints in the ways they arrange their armies - perhaps Blotto can send at most 2 armies to f_1 due to the difficulty of moving troops, and Hopper can gain two extra armies from an ally if she sends all of her troops to f_2. These new strategy vectors would be

$$b = \{(2,2), (1,3), (0,4)\}$$

$$h = \{(3,0), (2,1), (1,2), (0,5)\}$$

This yields the payout matrix

$$\begin{array}{cc} & \begin{bmatrix} (3,0), (2,1), (1,2), (0,5) \end{bmatrix} \\ \begin{bmatrix} (2,2) \\ (1,3) \\ (0,4) \end{bmatrix} & \begin{bmatrix} 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 \end{bmatrix} \end{array}.$$

As the strategy vectors become more complex, the payout matrix has the potential to become more interesting as well.

While all these examples divide armies into different locations, this is not the only way to create strategies. In political campaigns, dollars spent can replace armies allocated, and fiscal quarters can replace battlefields. The utility function, then, outputs the percentage of votes a candidate wins over the course of the campaign, and is a sum of differentials in poll numbers over the course of the campaign. This will be explained in more detail in Section 4: **Methods**.

2.5 Monte-Carlo Method

Some matrices, though finite and solvable through linear programming, are so large that they can't be solved in any reasonable amount of time. In these cases, a very close estimate of the correct answer is acceptable, and much more achievable. These answers can be solved using subsets of the original strategy vectors and many simulations. This method, named after the casino in Monaco [6], is called the Monte-Carlo method. This employs the following process:

- For each simulation:
  1. Select a new subset of strategies for each player
  2. Create a payout matrix using only these strategies
  3. Solve this small payout matrix using linear programming
  4. Record optimal strategies on this matrix
  5. Add each of these proportions to an array corresponding to every possible strategy for each player, where each proportion goes in its corresponding spot
- Repeat the above steps many, many times
- There now exists an array, with high values for strategies that won a lot and low numbers for strategies that did not. Divide each of these numbers by the number of repetitions to find the approximate proportion of times each strategy should be used in optimal play.

As an example, consider the following payout matrix:

M			
	t_1	t_2	t_3
s_1	-1	2	-1
s_2	1	0	0
s_3	-1	1	1

One way to solve this is direct linear programming, outlined in Section 2.3: **Linear Programming**. Another way, however, is to test many different subsets of s and t against each other, and see which strategy wins the most. For this example, use 2 as a subset size. First, create empty arrays s and t, whose elements correspond to how often each strategy should be played:

$$s = [s_1\ wins, s_2\ wins, s_3\ wins] = [0, 0, 0]$$

$$t = [t_1\ wins, t_2\ wins, t_3\ wins] = [0, 0, 0]$$

Next, pick two strategies randomly for each player, and construct a payout matrix using only these strategies.

M		
	t_2	t_3
s_1	2	-1
s_3	1	1

Next, solve this matrix using linear programming. This yields the following optimal strategies:

- P1: Never play s_1, always play s_3
- P2: Play t_2 on $\frac{2}{3}$ of turns, play t_3 on $\frac{1}{3}$ of turns

Add these proportions to the corresponding spots in (2.5) and (2.5):

$$s = [0, 0, 1]$$

$$t = [0, 0.6667, 0.3333]$$

Now repeat the process. Pick another two strategies randomly for each player, and construct and solve a payout matrix using these strategies:

M		
	t_1	t_3
s_1	-1	-1
s_2	1	0

This yields the following optimum strategies:

- P1: Never play s_1, always play s_2
- P2: Never play t_1, always play t_3

Add these to s and t:

$$s = [0, 1, 1]$$

$$t = [0, 0.6667, 1.3333]$$

After 1,000,000 simulations like this, s and t will look something like this:

$$s = [247.3980, 666698.3446, 333054.2574]$$

$$t = [333498.3152, 444330.4690, 222171.2158]$$

Dividing each element of s and t by 1,000,000 yields

$$s = [0, 0.667, 0.333]$$

$$t = [0.333, 0.444, 0.222]$$

Do these look familiar? They should! They're the same answers derived from direct linear programming of the entire matrix (see Section 2.3: **Linear Programming**). Sometimes these numbers are slightly different than the correct ones, but increasing the number of simulations almost always takes care of this.

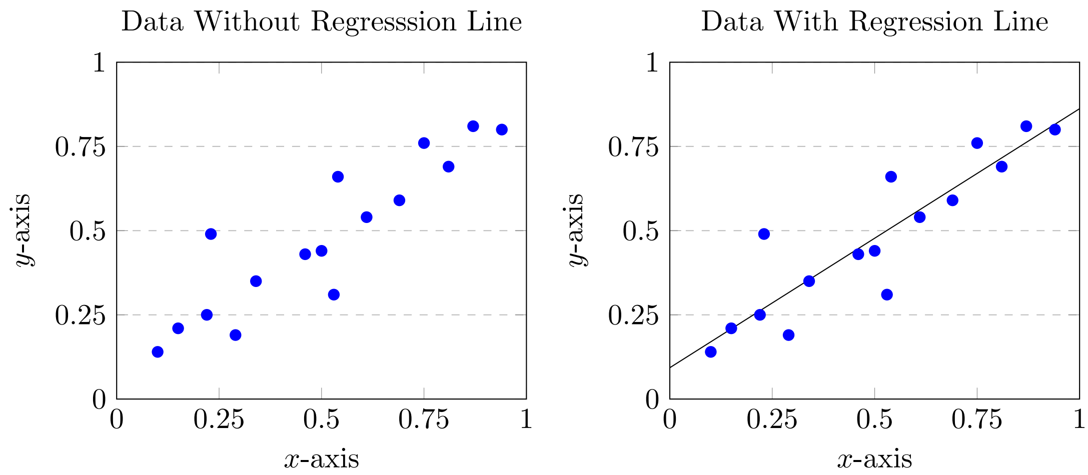


Figure 1: Theoretical data set, with and without regression line.

2.6 Linear Regression

Linear regression is a method used mostly in statistics to determine linear trends in a discrete dataset. Consider a dataset D, populated by 2-dimensional data (**Figure 1**. The goal of linear regression is to find a single linear function to describe the trend of the data over the x-axis in terms of the y-axis. In simpler terms, it's a line that shows a trend in the data. Doing this requires a simple optimization equation.

First, consider the set of data points in **Figure 1**. Call these points $\{(x_i, y_i)|i \in [1, 16] \cap \mathbb{N}\}$ (this is just a way to index the 16 points in the set). The goal is to find a line that follows the trend of the data as closely as possible. Plotting any line will create an error between the line and the data; therefore, to get the most accurate line, one must minimize that error equation.

To find the regression line, the first step is to calculate a simple mean of the x_i and y_i values. The equations for the means of this dataset, $\bar{x}$ and $\bar{y}$ respectively, are

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} \qquad \bar{y} = \frac{\sum_{i=1}^{n} y_i}{n}$$

where n is the number of points in the data set. Next, to calculate the difference between each x and y value and their respective means, use the equations

$$a_i = x_i - \bar{x} \qquad b_i = y_i - \bar{y}$$

From these variables, a slope m can be calculated. This value indicates how strongly the data is correlated - an m close to 1 indicates a strong direct correlation between the variables, while an m close to -1 indicates a strong inverse correlation. m values close to 0 indicate extremely weak correlation between the variables, or none at all. This m value is calculated by

$$m = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x - \bar{x})^2}$$

There are other ways to calculate this m value, which won't be covered in this paper.

For the data in **Figure 1**, this r_{xy} value turns out to be 0.769. This implies that a line with slope 0.769 is the the simple linear function which most closely models the data. This value can be calculated much more easily on Microsoft Excel, using the `SLOPE()` funciton [12]. Normally when using regression, a y-intecept value is also calculated. However, in this model, only the slope was used.

3 The Game

3.1 Initial Assumptions

Before even beginning to model the election, some initial assumptions are required. These serve as the basis for the rules of the game, since they allow abstract concepts to be quanitfied as variables or equations. These assumptions are as follows:

1. A typical U.S. Senate campaign starts as early as January 1^{st} of the election year.
2. Trends in the spending and fundraising data from the 2016 and 2018 United States Senate races are representative of trends in spending and fundraising data for all modern U.S. Senate races
3. Because spending and fundraising reports for U.S. Senate races are due on the first day of January, April, July, and October [13], these dates can be reasonably chosen to denote the end of a fiscal quarter. Furthermore, spending and fundraising strategies can be categorized into these three-month increments:
   - Quarter 1 (Q1) of a given year is January 1^{st} - March 31^{st} of that year
   - Quarter 2 (Q2) of a given year is April 1^{st} - June 30^{th} of that year
   - Quarter 3 (Q3) of a given year is July 1^{st} - September 30^{th} of that year
   - Quarter 4 (Q4) of a given year is October 1^{st} - December 31^{st} of that year
4. The spending and fundraising strategies of a U.S. Senate campaign are comprehensively reported on the FEC website. Any PAC or committee not listed as an approved committee on the FEC website is not considered in spending and fundraising strategies, nor are they considered in calculating the fundraising return function.
5. U.S. Senate candidates in the 2016 election who also ran for President collected votes and money using strategies fundamentally different than other U.S. Senate candidates, and these races were therefore not considered in any calculation.
6. How much a candidate will fundraise in any given quarter may be reasonably predicted as a linear function of past spending and fundraising.
7. Trends in a candidate's popularity in any given quarter may be reasonably predicted as a function of that candidate's entire last quarter of spending, their opponent's entire last quarter of spending, and differences in the results of independently collected straw polls.
8. U.S. Senate candidates spend their entire budget by the end of the last quarter of their campaign.

Obviously, many of these assumptions are *at least* somewhat removed from reality. Assumption 1, for example, wrongly assumes that candidates do not begin campaigning before January 1 of an election year. Assumptions 5, 6, 7, and 8 are also extremely presumptive. To calculate any sort of reasonable model in the time allotted, however, these assumptions had to be made. As further outlined in Section 6: **Critiques**, the model has immense room for improvement in many aspects.

Throughout the report, the utility vectors take the form

$$[a, b, c, d] \cdot 2 \times 10^{-5}.$$

In the code, spending amounts were calculated in increments of \$200,000, rather than in dollar amounts. Therefore, utility vectors which in the report which would have appeared as

$$[0.00001242, 0.00000143, -0.00001119, 0.00003519]$$

would be written in the code as

$$[2.484, 0.286, -2.238, 7.038].$$

In other words, every element of the utility vectors in the code are multiplied by 200,000 (or 2×10^5). As it turns out, these vectors are much easier to read and interpret. Therefore, rather than writing them in the report in their original form, each number is left multiplied by 200,000, and the scalar 2×10^{-5} is left outside the matrix to indicate this change.

With these assumptions in place, a proper game may be constructed. Like any good game, this model utilizes a nice set of rules, outlined below.

3.2 The Rules

1. The Race for U.S. Senate is a zero-sum game consisting of two players, A and B.

2. Each player has an initial budget, y_A and y_B. A player may never spend more in any quarter than their budget holds at the beginning of that quarter, after fundraising is accounted for.

3. There are 4 rounds in the game, corresponding to 4 quarters (3-month blocks) outlined in Section 3.1: **Assumptions**. Candidates play each round simultaneously.

4. Each round consists of two phases: Players receive fundraising, and players choose an amount to spend in the current month.

   - Fundraised money is earned in phase 1 as a function of a player's last fundraised amount and most recently spent amount. This money is added to their budget before they pick another spending strategy.
   - On the first turn, a player receives no fundraising in addition to their initial budget. Therefore, fundraising added in round two is given as a function of that player's Q1 spending and their initial budget.
   - After fundraising is added to each player's budget, they may choose to spend any amount of money between zero and their current budget, as long as the amount is a whole number.

5. The fundraising return function for each player in quarter i is given by

   $$f_i(f_{i-1}, s_i) = 0.1124s_i + 1.1057f_{i-1}$$

   where s_i denotes the last player's amount spent in quarter i, and f_{i-1} denotes their amount fundraised in the previous quarter. This function is not implemented in the first turn. At the beginning of the second turn, f_{i-1} is replaced by the player's initial budget y_0.

6. The game also includes an unspecified, finite, and constant number of voters. Each player's objective for the game is to maximize the *percentage* of these voters who will vote for them at the end of the last round.

7. Players must only spend money in multiples of \$200,000 each round, except in the last round.

8. Players must spend their entire remaining budget in the last round of the game.
9. The **cost of votes** is defined by how much **more** a player must spend than their opponent in that quarter to move 1% of votes (for P1, this move is positive; for P2, this move is negative). The utility function for each quarter, then, is defined by percentage of votes a player gains by spending \$1 more than their opponent in that round.
10. The utility function is unique to each round.
    - In Round 1, the utility function is defined by
    $$u(s_1, t_1) = (0.611974882 \cdot 2 \times 10^5) \cdot (s_1 - t_1)$$
    where s_1 is the amount P1 spends in Round 1, t_1 is the amount P2 spends in Round 1.
    - In Round 2, the utility function is defined by
    $$u(s_2, t_2) = (0.214168992 \cdot 2 \times 10^5) \cdot (s_2 - t_2)$$
    where s_2 is the amount P1 spends in Round 2, t_2 is the amount P2 spends in Round 2.
    - In Round 3, the utility function is defined by
    $$u(s_3, t_3) = (0.234584564 \cdot 2 \times 10^5) \cdot (s_3 - t_3)$$
    where s_3 is the amount P1 spends in Round 3, t_3 is the amount P3 spends in Round 3.
    - In Round 4, the utility function is defined by
    $$u(s_4, t_4) = (2.111203038 \cdot 2 \times 10^5) \cdot (s_4 - t_4)$$
    where s_4 is the amount P1 spends in Round 4, t_4 is the amount P2 spends in Round 4.
11. A negative value at the end of the game denotes a win for P1. A positive value at the end of the game denotes a win for P2. The absolute value of the game value itself denotes
    $$\mid v \mid = \mid v_1 - v_2 \mid,$$
    where v_1 denotes the percentage of voters who voted for P1, and v_2 denotes the percentage of voters who voted for P2. In other words, $\mid v \mid$ denotes the point differential of the winner over the loser.

4 Methods

Construction of the game matrix consisted of three discrete parts: a **fundraising return function**, a **weighted payout vector**, and each player's **strategy vector**.

- The **fundraising return function** models how candidates can make money later in a campaign by spending their money on fundraising events and advertising
- The **weighted payout vector** essentially models how expensive it is to gain votes in different quarters of a campaign
- The **strategy vectors** model all possible spending strategies a player can employ, given a specified budget and number of quarters

4.1 Fundraising Return Function

Building the fundraising function required, first of all, lots of data on how past candidates raised and spent their funds. This data can be found on the FEC website [2]. Data was downloaded for the two most popular Senate candidates in each 2016 race; this was determined by simple comparison of final poll numbers posted on the FEC website. The data was then grouped into different time increments of spending and fundraising, including months and quarters.

The next step was finding a the fundraising return function, using spending and fundraising as parameters. The output was the amount of money a candidate would gain from fundraising in a current time increment, dependent on their past spending and fundraising. This function was found using least-squares regression, and was calculated using both month-long and quarter-long time increments, with quarters of each year being defined as the following:

- Q1 defined as January 1^{st} - March 31^{st} of a single year
- Q2 defined as April 1^{st} - June 30^{th} of a single year
- Q3 defined as July 1^{st} - September 30^{th} of a single year
- Q4 defined as October 1^{st} - December 31^{st} of a single year

Note that in Q4 of an election year, candidates generally only spend money up until November 6^{th}, when the election takes place.

The challenge was finding which parameters most accurately predicted a candidate's fundraising. After consultation with subject matter experts (Max Hokit, Staff Assistant to California Congressman Salud Carbajal [4] and Jacqueline Wetzel, Financial Director to Texas State Senator Beverly Powell [17]), it was determined that any number of factors could affect campaign fundraising for a given quarter; however, the most influential contributing factors would be the most recent fundraising amounts the most recent spending amounts. Using FEC data on the 2016 Senate race, multiple possible formulas were explored for creating this return function. This function has immense room for improvement, but in order to simplify the model, it was solved as a function of the past quarter's spending and the past quarter's fundraising. To solve for the final function, data was gathered for each Senate candidate on how much they spent and fundraised each quarter of 2017 and 2018. These amounts were determined by accessing the FEC Campaign Finance Database [2], and including the fundraising and expenditure receipts from the candidate's FEC profile, as well as all authorized committees associated with that profile. As will be expanded upon in Section 7: **Critiques**, the greatest confounding variables in this function are likely:

- Committees not authorized by the candidate who spent and fundraised independently for the campaign, which creates neither expenditure nor fundraising receipts
- Volunteer work done for the campaign, which creates no expenditure receipts
- Candidates whose fundraising was based more on social media messaging (and other free forms of messaging) than it was on tactics which create expenditure receipts
- News stories in the media that either curbed or increased fundraising for a given quarter, which may have indirectly increased or decreased spending in the campaign to make up for lost points
- Campaigns that incorrectly report expenditures

None of these possibilities could be accounted for in this simplistic model of fundraising, creating one of the major weaknesses in the model. This weakness is touched upon again later, in Section 6: **Critiques**.

Using least-squares approximation in MATLAB, many possible solutions to this fundraising equation were calculated. These solutions used many different combinations of past spending and past fundraising; however, with little hard evidence backing a specific solution, the simplest equation was picked. This equation, solving for the fundraising f_i garnered in month i, was ultimately calculated as a function of the previous quarter's fundraising (f_{i-1}) and the total amount spent in month i (s_i). Least-squares approximation yielded the following equation:

$$f_i(f_{i-1}, s_i) = 0.1124 s_i + 1.1057 f_{i-1}$$

In this equation, the f_{i-1} mandates that a candidate fundraise more each consecutive quarter, which is typically (but not universally) true. In addition, spending more also allows a candidate to fundraise more effectively that quarter, which is also typically true.

4.2 Weighted Utility Vector

The next step was creating some sort of payout vector, using techniques from Weighted Blotto Models. First, spending data was gathered for the 2018 Senate race, using the same methods used in creating the fundraising return function. The vector still needed some sort of weighted payout per quarter; it was determined that straw polls taken throughout the race were the only way to calculate this payout. The main challenge was finding straw polls that were both comprehensive and accurate. Two sites were considered for these polls: Real Clear Politics [1] (a political news and poll conglomeration website), and FiveThirtyEight [16] (a news company focusing on statistical analysis). To check the biases of each polling site, Media Bias Fact Check [10] was consulted to give comprehensive bias-check reviews of them. The website gives the following ratings for each:

- **Real Clear Politics: Right-Center Bias** "In review, the website features selected political news stories and op-eds from various news publications in addition to commentary from its own contributors. Though their own political views lean conservative, the sites founders say their goal is to give readers ideological diversity in its commentary section. In reviewing their political news and opinions there are slightly more that are published from right leaning sources, however both sides are represented. Real Clear Politics is perhaps best known for their RCP Polling Average, which combines all polling data to create a statistical average. Most of the news content on Real Clear Politics is aggregated from other sources such as: Washington Post, New York Post, Salon, Fox News, The Federalist and National Review.

Several of the sources used by Real Clear Politics are listed as Mixed factual due to failed fact checks. In reviewing original Real Clear Politics articles, there is a right leaning bias in wording and story selection such as this: Gov. Jerry Moonbeam Browns Warning to Fellow Democrats. Although this is an opinion piece, it completely lacks sourcing of any kind. In general, the majority of stories from Real Clear Politics comes from a right leaning perspective. Overall, we rate RCP as Right-Center biased based on source selection that leans right and Mixed for factual reporting due to use of multiple sources who have failed fact checks."

- **FiveThirtyEight: Left-Center Bias** "FiveThirtyEight, sometimes referred to as 538, is a website that focuses on opinion poll analysis, politics, economics, and sports blogging. The website, which takes its name from the number of electors in the United States electoral college, was founded on March 7, 2008, as a polling aggregation website with a blog created by analyst Nate Silver. FiveThirtyEight typically relies on its methodology and not opinion for its reports, however they do publish new stories that have a slight left-center bias in coverage."

Due to a higher number of polls reaching further back in the race, transferable data organization, and Tim Wetzel's undying love and respect for Nate Silver, FiveThirtyEight was chosen as the source for this polling data. The data was next organized by state, and then by the first day of data collection. The first day was chosen because any other date used would require manual entry to assign a date to each poll. Polls were then categorized into quarters, using the first day of data collection.

Next, each race was analyzed individually. For each quarter, poll numbers were arranged in a 2-dimensional array, with the x-axis denoting which day in the quarter the poll's data collection began, and the y-axis denoting the differential between the two candidates' poll ratings. After the data was organized in this way for each race and quarter, a simple regression was calculated in Microsoft Excel, using the SLOPE function [12]. This regression was designed to give a number approximating the number of percentage points the polling differential changed between the two candidates, with an upward trend denoting a change toward the "positively-designated" candidate and a downward trend denoting a change toward the "negatively-designated" candidate.

It should be noted that these regression lines generally had many more data points at the end of the election than at the beginning, due to the nature of election news coverage. This likely created the largest errors in the first two quarters of 2018. In addition, Q4 2018 consists almost exclusively of data from October 1, 2018 - November 6, 2018, since November 6 is election day.

In order to determine the best utility function, a variety of options were explored for the model:

1. Function calculated directly from polling numbers (Unweighted)
2. Functions where polls with higher sample sizes were weighted more heavily (Weighted by Sample)
3. Functions where polls were weighted more heavily for having a higher sample size-to-state population ratio (Weighted by State)
4. Functions where individual negative values that composed a race's utility function were set to zero, and positive values were left unedited (Floored)

This yielded six different utility functions, and in all likelihood, none of them are even close to correct. However, they do offer interesting ideas of what a utility function over the last year of a Senate election might look like. Each vector has four values, each corresponding to a different fiscal quarter of an election year ([Q1, Q2, Q3, Q4] respectively). The values themselves represent how many points the polls would move in a player's favor if that player were to spend \$1. For example, a value of 0.00000113 indicates that spending \$1 creates a .0.00000113% change in voters' favor of that player; a value of -0.00000009 indicates a 0.00000009% change in favor of that player's opponent. However, as previously mentioned, a constant 2×10^{-5} is placed outside the vector, to make the numbers more readable. Without this constant, the numbers would indicate the effect in poll points a \$200,000 spend would create (see Section **3.1: Initial Assumptions**). The equations are as follows:

Utility Vectors Considered				
Name of Utility Vector	**Q1**	**Q2**	**Q3**	**Q4**
Unweighted, Unfloored	$6.30872096 \cdot 2 \times 10^{-5}$	$2.549945091 \cdot 2 \times 10^{-5}$	$0.401012477 \cdot 2 \times 10^{-5}$	$-1.653903518 \cdot 2 \times 10^{-5}$
Weighted by Sample, Unfloored	$0.26559308 \cdot 2 \times 10^{-5}$	$0.242170061 \cdot 2 \times 10^{-5}$	$0.124087002 \cdot 2 \times 10^{-5}$	$-0.419994208 \cdot 2 \times 10^{-5}$
Weighted by State, Unfloored	$0.353833059 \cdot 2 \times 10^{-5}$	$-0.460278059 \cdot 2 \times 10^{-5}$	$0.204454969 \cdot 2 \times 10^{-5}$	$-0.480740837 \cdot 2 \times 10^{-5}$
Unweighted, Floored	$0.611974882 \cdot 2 \times 10^{-5}$	$0.214168992 \cdot 2 \times 10^{-5}$	$0.234584564 \cdot 2 \times 10^{-5}$	$2.111203038 \cdot 2 \times 10^{-5}$
Weighted by Sample, Floored	$0.167306442 \cdot 2 \times 10^{-5}$	$0.089473901 \cdot 2 \times 10^{-5}$	$0.11420262 \cdot 2 \times 10^{-5}$	$2.841187593 \cdot 2 \times 10^{-5}$
Weighted by State, Floored	$0.243860933 \cdot 2 \times 10^{-5}$	$0.282414783 \cdot 2 \times 10^{-5}$	$0.14780057 \cdot 2 \times 10^{-5}$	$3.216185277 \cdot 2 \times 10^{-5}$

Some omissions were made in this part of the model. Specifically, for any fiscal quarter for a race, if there were not at least two polls taken at different dates in that quarter, a regression line could not be calculated. This eliminated many different quarters for many states, and it eliminated Maryland, Nebraska, Washington, and Wyoming's races altogether. Additionally, any poll made using SurveyMonkey was not considered, due to poor collection methods and the overwhelming presence of this data in several races.

4.3 Generating Strategies

With a fundraising function and payout vector in place, the next step was generating strategies two generalized players P1 and P2, given two initial budgets. This was more complicated than a simple iteration, since the fundraising return function created extra strategies in Q2, Q3, and Q4. Therefore, strategies had to be created iteratively, one quarter at a time. For each candidate, Q1 strategies were simply a list of numbers from zero to the maximum whole number that was less than or equal to the candidate's current budget. These data points were listed in conjunction with the candidate's budget after spending that amount.

Next, fundraising from the previous quarter had to be determined. Recall from equation [CITE] that this function uses parameters (previous quarter's fundraising return) and (previous quarter's spending). However, due to the recursive nature of this function, initial budget y_0 was substituted into the function in lieu of previous fundraising return. This yielded

$$f_1(y_0, s_1) = 0.1124s_1 + 1.1057y_0$$

as the function for the first quarter's fundraising.

At this point, the strategy array contained strategies only for the first quarter, with a remaining budget y_i for each candidate in quarter i calculated by

$$y_i = y_{i-1} - s_i + f(f_{i-1}, s_i)$$

The process was then repeated, this time iterating through each number from 0 to the highest integer less than or equal to the strategy's remaining budget. This was repeated once more for the third quarter, then each remaining budget was spent in the fourth quarter. P1's strategy vector will be denoted s, while t's strategy vector will be denoted t.

After these strategies were generated for both players, dominated strategies were eliminated in order to make solving the payout matrix more efficient. To do this, a function was written to compare elements of each strategy vector to each other (recall that elements of these strategy arrays are vectors of length 4). If every element of one vector was greater than or equal to every element of another vector, the latter vector is considered a **dominated strategy** and was eliminated from its strategy vector.

4.4 Solving the Payout Matrix

With all dominated strategies eliminated, a payout matrix M could be constructed. Like in **Section 2.2**, positive values were assigned to P1, and negative values were assigned to P2. A matrix was constructed with rows corresponding to each element of s and columns corresponding to each element of t. Each element of M was calcuated using the utility function constructed in **Section 6.2**.

It would make the most sense, from a mathematical point of view, to solve this matrix constructed from every possible strategy of each player. However, in practice, this was nearly impossible to do. As initial budgets increased, the size of the matrix increased exponentially, and the code would not run in an even remotely reasonable amount of time. To address this, a Monte-Carlo approach to solving the matrix was required.

First, each player's strategies were generated the same as before. Then, 100 random subsets were generated from s and t. 100,000 matrices were created in this way, pitting pairs of strategies against each other. This generated optimal strategies for 100,000 games between subsets of s and t. Each strategy belonging to an optimized strategy vector was added to a "winners list," then each instance of the strategy was counted. This yielded a vector consisting of tuples, where each tuple contained indices of every strategy that was used in an optimized strategy vector, and the number of times it was used. Last, the latter of these numbers was divided by 100,000 to give the proportion of games that strategy should be used.

Example Initial Budgets								
Year	State	P1 Name	Party	Initial Budget	P2 Name	Party	Initial Budget	Budget Differential
2018	CA	Dianne Feinstein	D	$7,540,299	Kevin DeLeon	D	$1,189,460	$6,350,838
2018	PA	Bob Casey	D	$5,857,855	Lou Barletta	R	$83,772	$5,774,083
2018	MO	Claire McCaskill	D	$5,491,965	Josh Hawley	R	$1,156,672	$4,335,292
2018	MT	Jon Tester	D	$2,601,442	Matthew Rosendale	R	$315,399	$2,286,043
2018	IN	Mike Braun	R	$2,845,151	Joe Donnelly	D	$1,214,857	$1,630,293
2018	VA	Tim Kaine	D	$1,652,252	Corey Stewart	R	$39,334	$1,612,917
2018	AZ	Kyrsten Sinema	D	$2,139,420	Martha McSally	R	$630,040	$1,509,380
2018	WI	Tammy Baldwin	D	$1,712,842	Leah Vukmir	R	$210,657	$1,502,184
2018	NV	Jacky Rosen	D	$1,621,476	Dean Heller	R	$317,237	$1,304,238
2018	MI	Debbie Stabenow	D	$1,637,430	John James	R	$405,181	$1,232,249
2018	ND	Heidi Heitkamp	D	$1,375,695	Kevin Cramer	R	$213,441	$1,162,254
2018	RI	Sheldon Whitehouse	D	$781,534	Robert Flanders	R	$84,280	$697,253
2018	TN	Marsha Blackburn	R	$1,485,492	Phil Bredesen	D	$808,984	$676,507
2018	ME	Angus King	I	$614,868	Eric Brakey	R	$47,671	$567,197
2018	NM	Martin Heinrich	D	$572,439	Mick Rich	R	$19,943	$552,496
2018	WV	Joe Manchin	D	$814,799	Patrick Morrisey	R	$641,753	$173,045
2018	WY	John Barrasso	R	$424,225	Gary Trauner	D	$254,031	$170,194
2018	CT	Chris Murphy	D	$18,350	Matthey Corey	R	$3,001	$15,349
2018	NE	Debbie Fischer	R	$306,930	Jane Raybould	D	$302,889	$4,041
2016	WI	Russ Feingold	D	$3,433,425	Ron Johnson	R	$888,443	$2,544,981
2016	CA	Loretta Sanchez	D	$2,060,321	Kamala Harris	D	$66,0586	$1,399,734
2016	OH	Rob Portman	R	$1,434,749	Ted Strickland	D	$879,287	$555,462
2016	NH	Margaret Hassan	D	$2,086,756	Kelly Ayotte	R	$1,536,590	$550,165
2016	MO	Roy Blunt	R	$1,146,922	Jason Kander	D	$837,161	$309,761
2016	PA	Pat Toomey	R	$1,359,343	Katie McGinty	D	$1,079,492	$279,851
2016	NV	Catherine Cortez Masto	D	$765,080	Joe Heck	R	$633,667	$131,413
2016	NC	Richard Burr	R	$639,766	Deborah Ross	D	$583,201	$56,564
2016	AZ	John McCain	R	$368,348	Ann Kirkpatrick	D	$332,292	$36,055

Figure 2: Real-world examples of starting budgets of top U.S. Senate candidates. Initial budgets calculated by subtracting money spent in Q4 the year before the race from money fundraised in Q4 the year before the race. Initial budgets less than or equal to zero omitted.

5 Results

The first step in getting actual results was determining which initial budgets to assign to players. There are infinitely many possible initial budgets to assign to each player, creating infinite possible games. Therefore, reasonable budgets had to be picked. To do this, real-world examples of "initial budgets" calculated for January of election years 2016 and 2018 were collected as reference (see **Figure 2**. These "initial budgets" were calculated by summing each candidate's amount fundraised in Q4 of the year before the election (Q4 2015 and Q4 2017, respectively) and subtracting the amount that candidate spent that same quarter. However, many of these races had huge discrepancies in budgets, which would lead to very uninteresting races if plugged into the model. Therefore, the example budgets in **Figure 4** were instead used. For testing, $800,000 was picked as the budget for each lower-budget player (P1). This number was picked to ensure that each lower-budget player had at least 4 different strategy options in Q1 of the race, since $200,000 was picked as the increment in which players could spend. To test qualitatively different races, the budget of the higher-budget player was determined by multiplying the budget of P1 by 1.00, 1.125, 1.25, 1.375, 1.5, 1.675, 1.75, and 2.00 (see **Figure 4**). This covered many theoretical races, ranging from equally-matched races to those in which one player had double the budget of the other.

Next, this set of budgets were tested on the utility vector calculated in Section 4.2: **Weighted Utility Vector**.

5.1 Unweighted, Floored Utility Vector

Recall that the Unweighted, Floored Utility Vector was calculated by setting negative regression slopes in individual races to zero, then taking a simple average of these slopes for each fiscal quarter. This yielded the vector

$$[0.611974882, 0.214168992, 0.234584564, 2.111203038] \cdot 2 \times 10^{-5}. \tag{3}$$

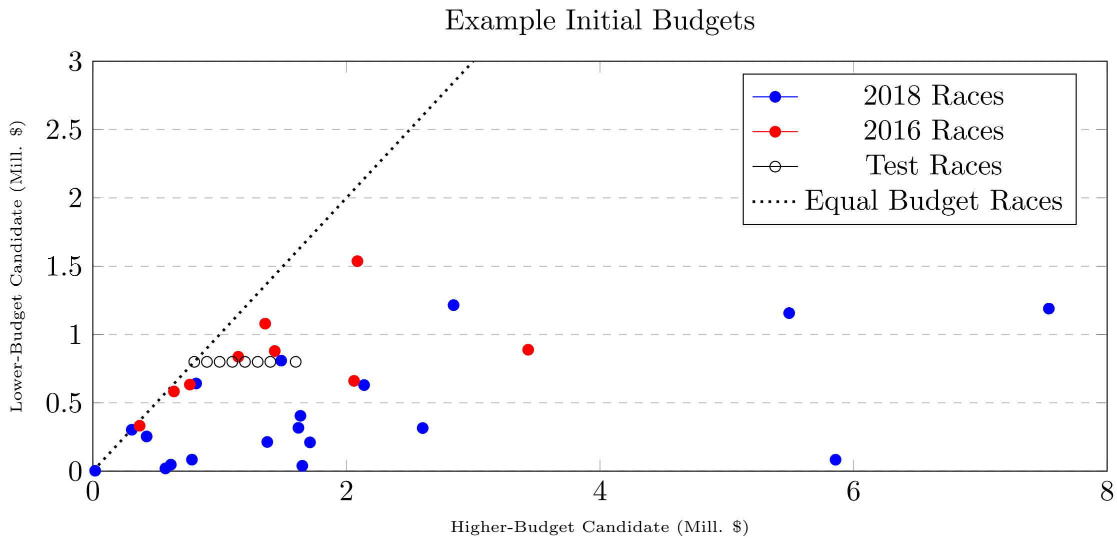


Figure 3: Discrepancies between initial budgets of 2016 and 2016 top Senate candidates listed in **Figure 2**. Blue dots indicate races in 2018, and red dots indicate races in 2016. Note that, in most races, one candidate has a significantly higher initial budget than the other. No points occur over the Equal Budget Races line (though some occur exactly on the line), since the higher-budget candidate was always plotted on the x-axis.

Tested Initial Budgets					
Example	P1 Initial Budget	P2 Initial Budget	Initial Budget Differential	$\frac{P1\ Initial\ Budget}{P2\ Initial\ Budget}$	$\frac{P2\ Initial\ Budget}{P1\ Initial\ Budget}$
Budget Test 1	$800,000	$800,000	$0	1.0000	1.000
Budget Test 2	$800,000	$900,000	$100,000	0.8889	1.1250
Budget Test 3	$800,000	$1,000,000	$200,000	0.8000	1.2500
Budget Test 4	$800,000	$1,100,000	$300,000	0.7273	1.3750
Budget Test 5	$800,000	$1,200,000	$400,000	0.6667	1.5000
Budget Test 6	$800,000	$1,300,000	$500,000	0.6154	1.6750
Budget Test 7	$800,000	$1,400,000	$600,000	0.5714	1.7500
Budget Test 8	$800,000	$1,600,000	$800,000	0.5000	2.0000

Figure 4: All sets of theoretical budgets, tested against each other in pairs.

P1 $800,000 vs. P2 $800,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,676	29.245%	$0	$0	$0	$3,640,676	29.620%
$200,000	$0	$0	$3,512,868	20.670%	$200,000	$0	$0	$3,512,868	20.822%
$0	$200,000	$0	$3,485,636	14.810%	$0	$200,000	$0	$3,485,636	14.721%
$400,000	$0	$0	$3,385,061	10.478%	$400,000	$0	$0	$3,385,061	10.365%
$0	$0	$200,000	$3,440,676	7.340%	$0	$0	$200,000	$3,440,676	7.346%
$200,000	$200,000	$0	$3,357,828	5.210%	$200,000	$200,000	$0	$3,357,828	5.125%
$600,000	$0	$0	$3,257,253	3.651%	$600,000	$0	$0	$3,257,253	3.588%
$200,000	$0	$200,000	$3,312,868	2.571%	$200,000	$0	$200,000	$3,312,868	2.556%
$0	$400,000	$0	$3,330,596	1.854%	$0	$400,000	$0	$3,330,596	1.748%
$400,000	$200,000	$0	$3,230,021	1.304%	$400,000	$200,000	$0	$3,230,021	1.243%
Other				2.867%	Other				2.866%
Game Value:			**-0.01**		**Winner:**			**P2**	

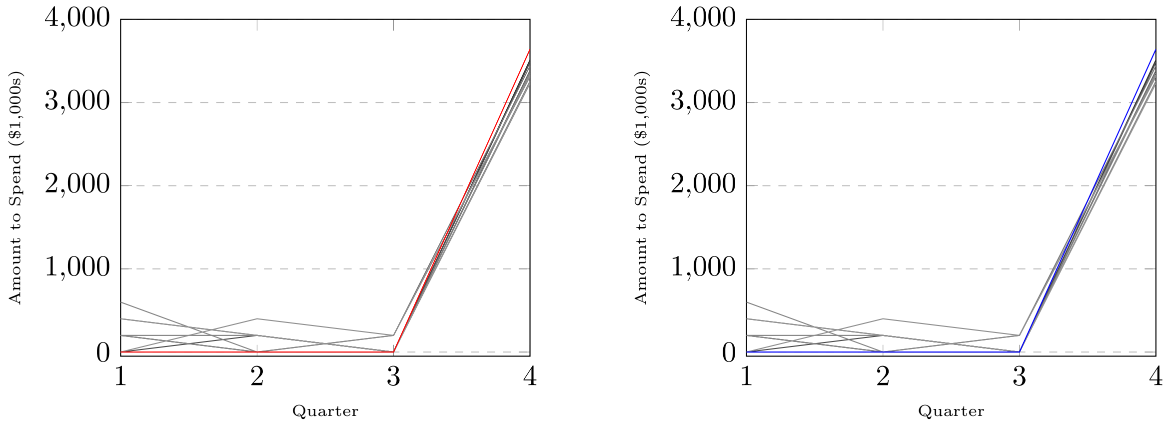
Figure 5: Mixed strategy output, when P1 and P2 are both given initial budgets of $800,000 and the Unweighted, Floored Utility Vector is used.

Using this utility vector yielded the following results. In every case, the top strategy for each player was to save all their money until the last quarter, then spend it all. This makes sense, since the last quarter was weighted so heavily in the utility vector.

In fact, as lesser-used strategies are examined, they reveal a common trend: spending as little as possible before Q4, then dumping the rest of the budget. The Q4 strategy column is arranged in almost exact descending order, denoting the importance of this quarter; the few exceptions to this rule are strategies which favor Q1, which has the next-highest voter return. It's likely this trend continues deep into the least-used strategies; however, in order to reduce runtime, strategies used less than 1% of the time were not calculated.

Another interesting trend in this model is the tendency for higher-budget candidates to have a wider range of strategies, and the tendency of lower-budget candidates to have more concentrated strategies. This is likely due to the way in which the matrices were solved. Because Monte-Carlo only plays random subsets of strategies against each other, small and large wins are counted equally. If the strongest strategy a player may employ is not present in this subset, the algorithm solves the matrix with only the strategies it has. While this may result in a lower-valued game, it still counts as a win for the game winner. As the budget ratio between players increases, the higher-budget player has many more strategies that beat their opponent. This takes what may be a concentrated strategy and dilutes it (see **Figure 21**).

Lastly, note that P2 wins in every case. While the game value in **Figure 5** should be zero, since P1 and P2 have identical starting budgets, the errors caused in the Play % column create a small error in the game value, resulting in a win for P2 in every case. This makes sense because players are never punished for spending more when using this vector, so a higher-budget player will always have a clear advantage.

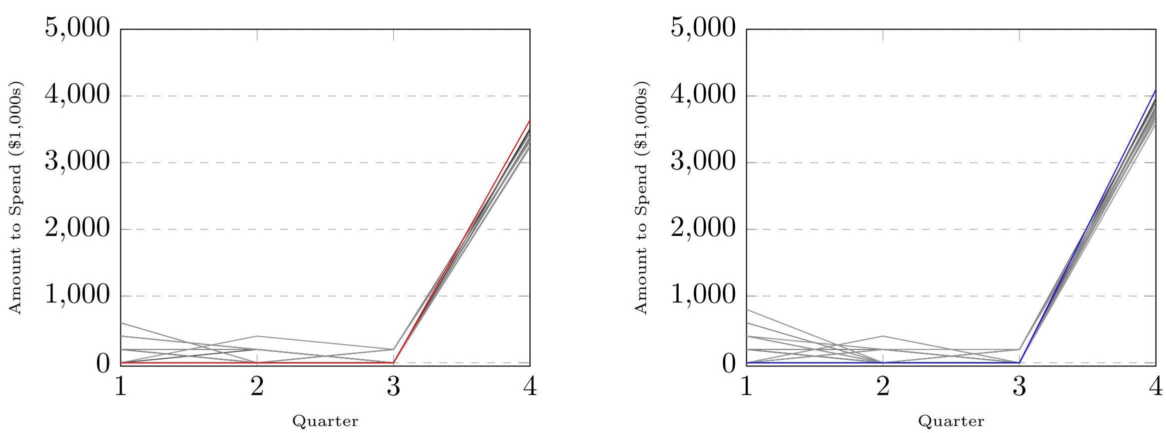


a P1: $800,000 Initial Budget **b** P2: $800,000 Initial Budget

Figure 6: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $900,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,676	29.451%	$0	$0	$0	$4,095,760	22.894%
$200,000	$0	$0	$3,512,868	20.936%	$200,000	$0	$0	$3,967,953	17.618%
$0	$200,000	$0	$3,485,636	14.660%	$0	$200,000	$0	$3,940,720	13.742%
$400,000	$0	$0	$3,385,061	10.457%	$400,000	$0	$0	$3,840,145	10.448%
$0	$0	$200,000	$3,440,676	7.413%	$0	$0	$200,000	$3,895,760	8.264%
$200,000	$200,000	$0	$3,357,828	5.133%	$200,000	$200,000	$0	$3,812,913	6.258%
$600,000	$0	$0	$3,257,253	3.549%	$600,000	$0	$0	$3,712,337	4.717%
$200,000	$0	$200,000	$3,312,868	2.570%	$200,000	$0	$200,000	$3,767,953	3.700%
$0	$400,000	$0	$3,330,596	1.765%	$0	$400,000	$0	$3,785,680	2.867%
$400,000	$200,000	$0	$3,230,021	1.264%	$400,000	$200,000	$0	$3,685,105	2.222%
					$800,000	$0	$0	$3,584,530	1.717%
					$0	$200,000	$200,000	$3,740,720	1.288%
					$400,000	$0	$200,000	$3,640,145	1.035%
Other				2.802%	Other				3.230%
Game Value:			**-4.55**		**Winner:**			**P2**	

Figure 7: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $900,000, and the Unweighted, Floored Utility Vector is used.

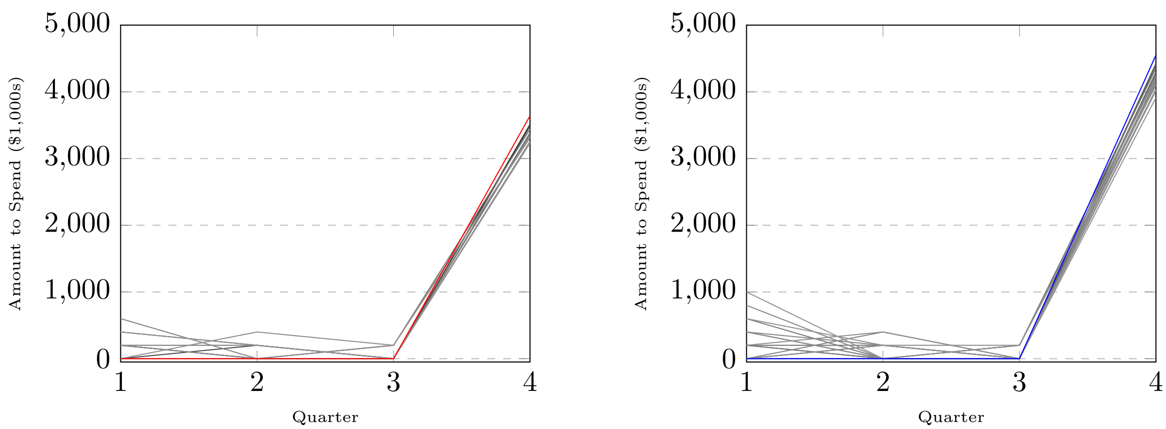


a P1: $800,000 Initial Budget **b** P2: $900,000 Initial Budget

Figure 8: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,000,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,676	29.451%	$0	$0	$0	$4,550,845	16.364%
$200,000	$0	$0	$3,512,868	20.936%	$200,000	$0	$0	$4,423,037	13.499%
$0	$200,000	$0	$3,485,636	14.660%	$0	$200,000	$0	$4,395,805	11.347%
$400,000	$0	$0	$3,385,061	10.457%	$400,000	$0	$0	$4,295,230	9.589%
$0	$0	$200,000	$3,440,676	7.413%	$0	$0	$200,000	$4,350,845	8.102%
$200,000	$200,000	$0	$3,357,828	5.133%	$200,000	$200,000	$0	$4,267,997	6.565%
$600,000	$0	$0	$3,257,253	3.549%	$600,000	$0	$0	$4,167,422	5.659%
$200,000	$0	$200,000	$3,312,868	2.570%	$200,000	$0	$200,000	$4,223,037	4.658%
$0	$400,000	$0	$3,330,596	1.765%	$0	$400,000	$0	$4,240,765	3.994%
$400,000	$200,000	$0	$3,230,021	1.264%	$400,000	$200,000	$0	$4,140,190	3.324%
					$800,000	$0	$0	$4,039,614	2.674%
					$0	$200,000	$200,000	$4,195,805	2.371%
					$400,000	$0	$200,000	$4,095,230	1.906%
					$200,000	$400,000	$0	$4,112,957	1.579%
					$600,000	$200,000	$0	$4,012,382	1.406%
					$1,000,000	$0	$0	$3,911,806	1.117%
Other				3.002%	Other				5.846%
Game Value:			**-9.02**		**Winner:**			**P2**	

Figure 9: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,00,000, and the Unweighted, Floored Utility Vector is used.



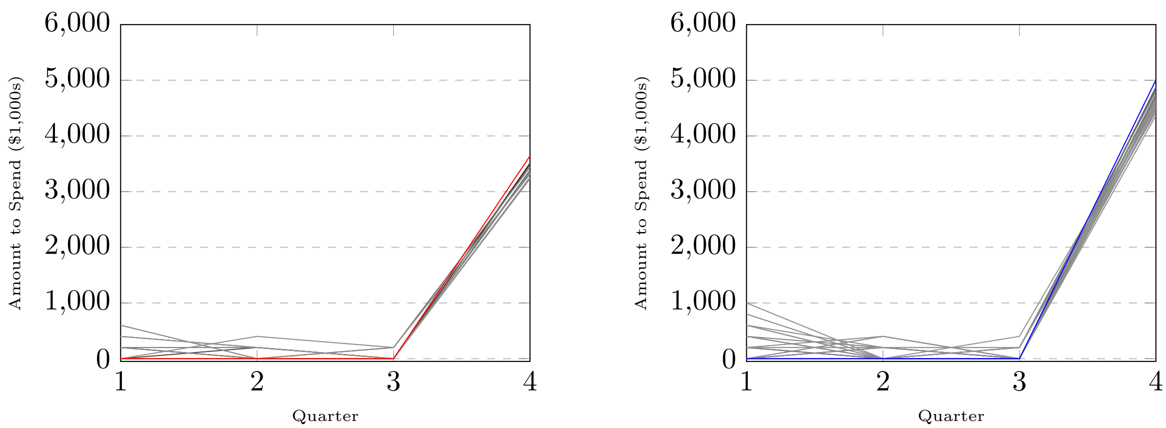
a P1: $800,000 Initial Budget

b P2: $1,000,000 Initial Budget

Figure 10: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,100,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,676	29.349%	$0	$0	$0	$5,005,929	12.834%
$200,000	$0	$0	$3,512,868	20.920%	$200,000	$0	$0	$4,878,122	11.530%
$0	$200,000	$0	$3,485,636	14.729%	$0	$200,000	$0	$4,850,889	9.950%
$400,000	$0	$0	$3,385,061	10.269%	$400,000	$0	$0	$4,750,314	8.451%
$0	$0	$200,000	$3,440,676	7.297%	$0	$0	$200,000	$4,805,929	7.421%
$200,000	$200,000	$0	$3,357,828	5.368%	$200,000	$200,000	$0	$4,723,082	6.453%
$600,000	$0	$0	$3,257,253	3.580%	$600,000	$0	$0	$4,622,506	5.599%
$200,000	$0	$200,000	$3,312,868	2.582%	$200,000	$0	$200,000	$4,678,122	4.941%
$0	$400,000	$0	$3,330,596	1.822%	$0	$400,000	$0	$4,695,849	4.211%
$400,000	$200,000	$0	$3,230,021	1.247%	$400,000	$200,000	$0	$4,595,274	3.711%
					$800,000	$0	$0	$4,494,699	3.293%
					$0	$200,000	$200,000	$4,650,889	2.863%
					$400,000	$0	$200,000	$4,550,314	2.424%
					$200,000	$400,000	$0	$4,568,042	2.139%
					$600,000	$200,000	$0	$4,467,466	1.879%
					$1,000,000	$0	$0	$4,366,891	1.580%
					$0	$0	$400,000	$4,605,929	1.400%
					$200,000	$200,000	$200,000	$4,523,082	1.185%
					$600,000	$0	$200,000	$4,422,506	1.111%
Other				2.837%	Other				7.025%
Game Value:			**-13.58**		**Winner:**			**P2**	

Figure 11: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,100,000, and the Unweighted, Floored Utility Vector is used.

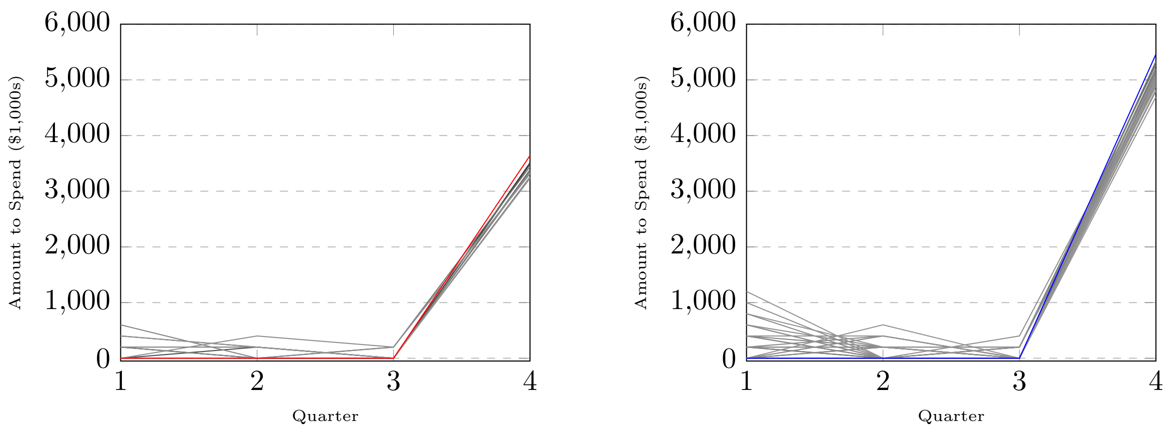


a P1: $800,000 Initial Budget **b** P2: $1,100,000 Initial Budget

Figure 12: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,200,000									
Player 1 (P1)					Player 2 (P2)				
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,676	29.539%	$0	$0	$0	$5,461,014	9.641%
$200,000	$0	$0	$3,512,868	20.915%	$200,000	$0	$0	$5,333,206	8.965%
$0	$200,000	$0	$3,485,636	14.501%	$0	$200,000	$0	$5,305,974	7.997%
$400,000	$0	$0	$3,385,061	10.416%	$400,000	$0	$0	$5,205,399	7.071%
$0	$0	$200,000	$3,440,676	7.322%	$0	$0	$200,000	$5,261,014	6.313%
$200,000	$200,000	$0	$3,357,828	5.270%	$200,000	$200,000	$0	$5,178,166	5.938%
$600,000	$0	$0	$3,257,253	3.714%	$600,000	$0	$0	$5,077,591	5.259%
$200,000	$0	$200,000	$3,312,868	2.558%	$200,000	$0	$200,000	$5,133,206	4.830%
$0	$400,000	$0	$3,330,596	1.733%	$0	$400,000	$0	$5,150,934	4.203%
$400,000	$200,000	$0	$3,230,021	1.231%	$400,000	$200,000	$0	$5,050,359	3.854%
					$800,000	$0	$0	$4,949,783	3.526%
					$0	$200,000	$200,000	$5,105,974	3.169%
					$400,000	$0	$200,000	$5,005,399	2.788%
					$200,000	$4,000,00	$0	$5,023,126	2.601%
					$600,000	$200,000	$0	$4,922,551	2.365%
					$1,000,000	$0	$0	$4,821,975	2.180%
					$0	$0	$400,000	$5,061,014	1.890%
					$200,000	$200,000	$200,000	$4,978,166	1.718%
					$600,000	$0	$200,000	$4,877,591	1.583%
					$0	$600,000	$0	$4,995,894	1.382%
					$400,000	$400,000	$0	$4,895,319	1.253%
					$800,000	$200,000	$0	$4,794,743	1.179%
					$1,200,000	$0	$0	$4,694,168	1.036%
Other				2.801%	Other				9.259%
Game Value:			**-18.06**		**Winner:**			**P2**	

Figure 13: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,200,000, and the Unweighted, Floored Utility Vector is used.

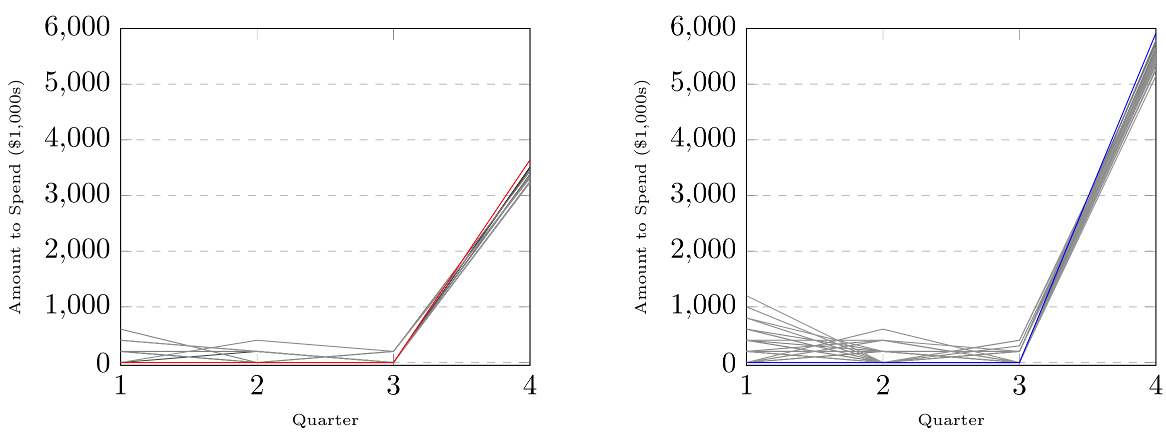


a P1: $800,000 Initial Budget **b** P2: $1,200,000 Initial Budget

Figure 14: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,300,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,676	29.363%	$0	$0	$0	$5,916,098	8.105%
$200,000	$0	$0	$3,512,868	20.734%	$200,000	$0	$0	$5,788,291	7.406%
$0	$200,000	$0	$3,485,636	14.613%	$0	$200,000	$0	$5,761,058	6.841%
$400,000	$0	$0	$3,385,061	10.410%	$400,000	$0	$0	$5,660,483	6.268%
$0	$0	$200,000	$3,440,676	7.476%	$0	$0	$200,000	$5,716,098	5.739%
$200,000	$200,000	$0	$3,357,828	5.097%	$200,000	$200,000	$0	$5,633,251	5.248%
$600,000	$0	$0	$3,257,253	3.657%	$600,000	$0	$0	$5,532,675	4.777%
$200,000	$0	$200,000	$3,312,868	2.555%	$200,000	$0	$200,000	$5,588,291	4.494%
$0	$400,000	$0	$3,330,596	1.777%	$0	$400,000	$0	$5,606,018	4.167%
$400,000	$200,000	$0	$3,230,021	1.257%	$400,000	$200,000	$0	$5,505,443	3.826%
					$800,000	$0	$0	$5,404,868	3.458%
					$0	$200,000	$200,000	$5,561,058	3.346%
					$400,000	$0	$200,000	$5,460,483	2.987%
					$200,000	$400,000	$0	$5,478,211	2.781%
					$600,000	$200,000	$0	$5,377,635	2.532%
					$1,000,000	$0	$0	$5,277,060	2.297%
					$0	$0	$400,000	$5,516,098	2.113%
					$200,000	$200,000	$200,000	$5,433,251	1.943%
					$600,000	$0	$200,000	$5,332,675	1.822%
					$0	$600,000	$0	$5,450,978	1.621%
					$400,000	$400,000	$0	$5,350,403	1.452%
					$800,000	$200,000	$0	$5,249,828	1.412%
					$1,200,000	$0	$0	$5,149,252	1.282%
					$200,000	$0	$400,000	$5,388,291	1.170%
					$0	$400,000	$200,000	$5,406,018	1.065%
Other				3.061%	Other				11.848%
Game Value:			**-22.68**		**Winner:**			**P2**	

Figure 15: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,300,000, and the Unweighted, Floored Utility Vector is used.

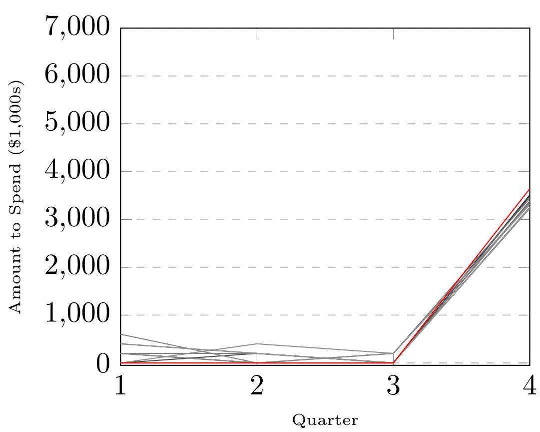


a P1: $800,000 Initial Budget **b** P2: $1,300,000 Initial Budget

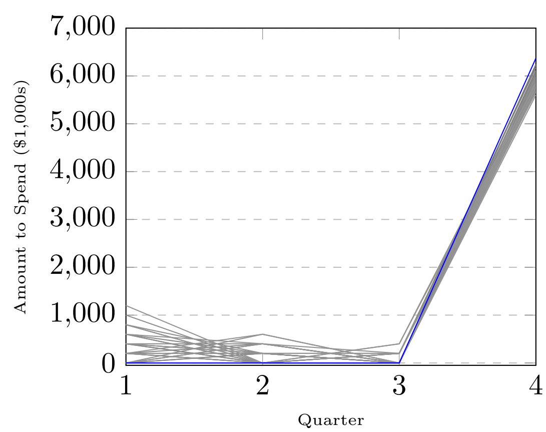
Figure 16: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,400,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,676	29.308%	$0	$0	$0	$6,371,183	6.379%
$200,000	$0	$0	$3,512,868	20.763%	$200,000	$0	$0	$6,243,375	5.883%
$0	$200,000	$0	$3,485,636	14.774%	$0	$200,000	$0	$6,216,143	5.563%
$400,000	$0	$0	$3,385,061	10.474%	$400,000	$0	$0	$6,115,568	5.112%
$0	$0	$200,000	$3,440,676	7.391%	$0	$0	$200,000	$6,171,183	4.854%
$200,000	$200,000	$0	$3,357,828	5.150%	$200,000	$200,000	$0	$6,088,335	4.597%
$600,000	$0	$0	$3,257,253	3.692%	$600,000	$0	$0	$5,987,760	4.299%
$200,000	$0	$200,000	$3,312,868	2.565%	$200,000	$0	$200,000	$6,043,375	4.007%
$0	$400,000	$0	$3,330,596	1.725%	$0	$400,000	$0	$6,061,103	3.779%
$400,000	$200,000	$0	$3,230,021	1.249%	$400,000	$200,000	$0	$5,960,528	3.593%
					$800,000	$0	$0	$5,859,952	3.269%
					$0	$200,000	$200,000	$6,016,143	3.028%
					$400,000	$0	$200,000	$5,915,568	2.895%
					$200,000	$400,000	$0	$5,933,295	2.620%
					$600,000	$200,000	$0	$5,832,720	2.553%
					$1,000,000	$0	$0	$5,732,144	2.355%
					$0	$0	$400,000	$5,971,183	2.231%
					$200,000	$200,000	$200,000	$5,888,335	2.171%
					$600,000	$0	$200,000	$5,787,760	2.022%
					$0	$600,000	$0	$5,906,063	1.811%
					$400,000	$400,000	$0	$5,805,488	1.719%
					$800,000	$200,000	$0	$5,704,912	1.585%
					$1,200,000	$0	$0	$5,604,337	1.478%
					$200,000	$0	$400,000	$5,843,375	1.407%
					$0	$400,000	$200,000	$5,861,103	1.353%
					$400,000	$200,000	$200,000	$5,760,528	1.228%
					$800,000	$0	$200,000	$5,659,952	1.116%
					$200,000	$600,000	$0	$5,778,255	1.104%
					$600,000	$400,000	$0	$5,677,680	1.049%
Other				2.909%	Other				14.940%
Game Value:			**-27.20**		**Winner:**			**P2**	

Figure 17: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,400,000, and the Unweighted, Floored Utility Vector is used.



a P1: $800,000 Initial Budget

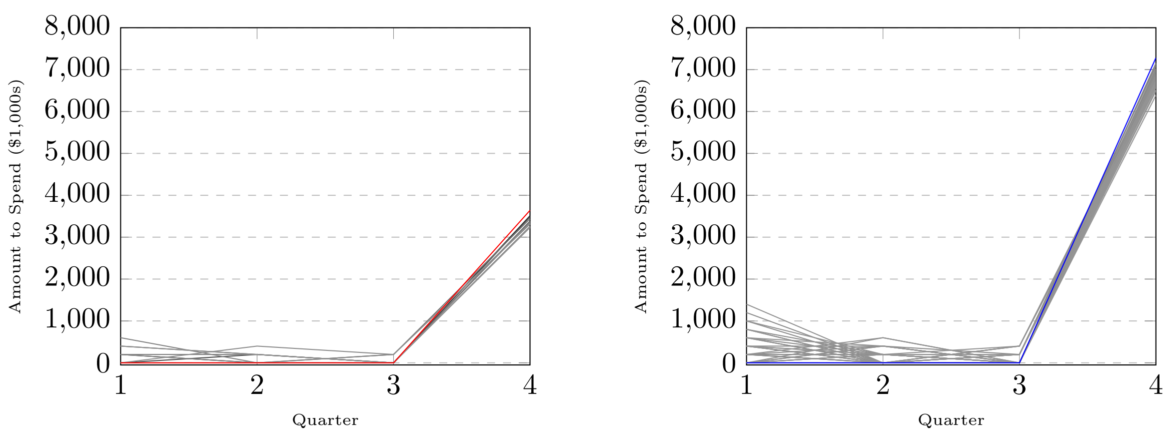


b P2: $1,400,000 Initial Budget

Figure 18: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,600,000									
Player 1 (P1)					Player 2 (P2)				
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,676	29.447%	$0	$0	$0	$7,281,352	4.358%
$200,000	$0	$0	$3,512,868	20.949%	$200,000	$0	$0	$7,153,544	4.223%
$0	$200,000	$0	$3,485,636	14.781%	$0	$200,000	$0	$7,126,312	3.993%
$400,000	$0	$0	$3,385,061	10.64%	$400,000	$0	$0	$7,025,737	3.813%
$0	$0	$200,000	$3,440,676	7.162%	$0	$0	$200,000	$7,081,352	3.608%
$200,000	$200,000	$0	$3,357,828	5.038%	$200,000	$200,000	$0	$6,998,504	3.401%
$600,000	$0	$0	$3,257,253	3.58%	$600,000	$0	$0	$6,897,929	3.259%
$200,000	$0	$200,000	$3,312,868	2.561%	$200,000	$0	$200,000	$6,953,544	3.159%
$0	$400,000	$0	$3,330,596	1.753%	$0	$400,000	$0	$6,971,272	3.069%
$400,000	$200,000	$0	$3,230,021	1.226%	$400,000	$200,000	$0	$6,870,697	2.893%
					$800,000	$0	$0	$6,770,121	2.852%
					$0	$200,000	$200,000	$6,926,312	2.591%
					$400,000	$0	$200,000	$6,825,737	2.578%
					$200,000	$400,000	$0	$6,843,464	2.458%
					$1,000,000	$0	$0	$6,642,313	2.315%
					$600,000	$200,000	$0	$6,742,889	2.306%
					$0	$0	$400,000	$6,881,352	2.123%
					$200,000	$200,000	$200,000	$6,798,504	2.064%
					$0	$600,000	$0	$6,816,232	2.010%
					$600,000	$0	$200,000	$6,697,929	1.854%
					$400,000	$400,000	$0	$6,715,657	1.826%
					$800,000	$200,000	$0	$6,615,081	1.693%
					$1,200,000	$0	$0	$6,514,506	1.691%
					$200,000	$0	$400,000	$6,753,544	1.592%
					$0	$400,000	$200,000	$6,771,272	1.540%
					$400,000	$200,000	$200,000	$6,670,697	1.423%
					$200,000	$600,000	$0	$6,688,424	1.368%
					$800,000	$0	$200,000	$6,570,121	1.363%
					$1,000,000	$200,000	$0	$6,487,273	1.274%
					$600,000	$400,000	$0	$6,587,849	1.236%
					$0	$200,000	$400,000	$6,726,312	1.136%
					$1,400,000	$0	$0	$6,386,698	1.101%
					$400,000	$0	$400,000	$6,625,737	1.043%
Other				2.863%	Other				22.787%
Game Value:			**-36.45**		**Winner:**			**P2**	

Figure 19: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,600,000, and the Unweighted, Floored Utility Vector is used.



a P1: $800,000 Initial Budget

b P2: $1,400,000 Initial Budget

Figure 20: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

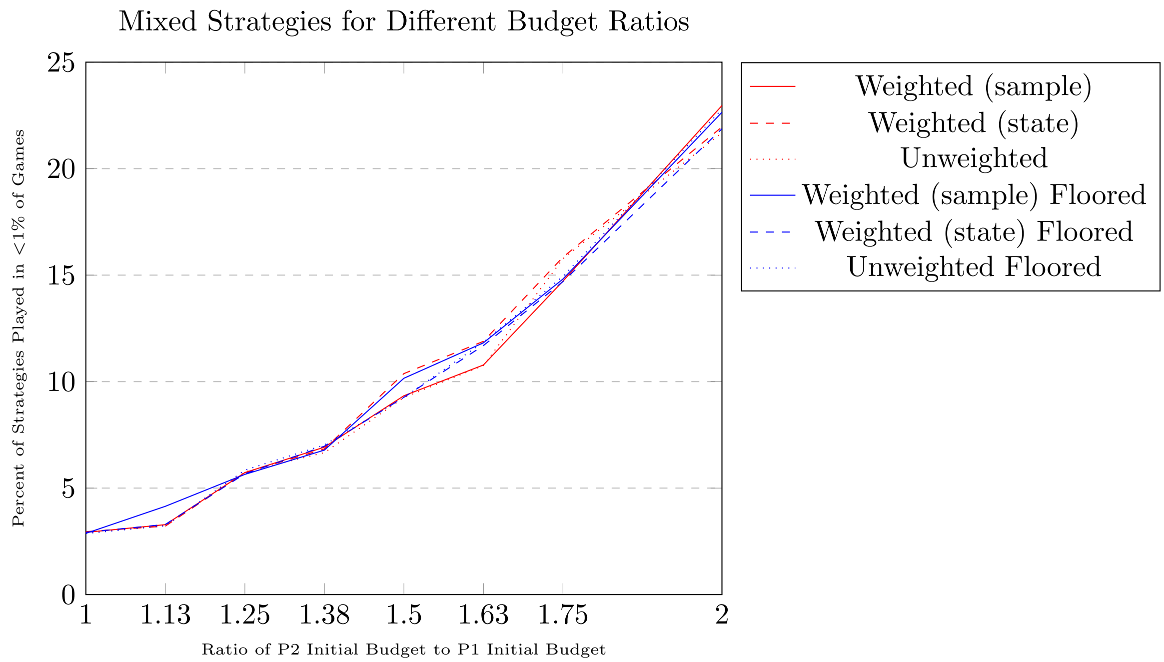


Figure 21: Games in which a player uses strategies designated for <1% of iterations, tracked as the ratio between budgets of players increases. As one player gains a clear budget advantage, that player's strategy set becomes more uniform.

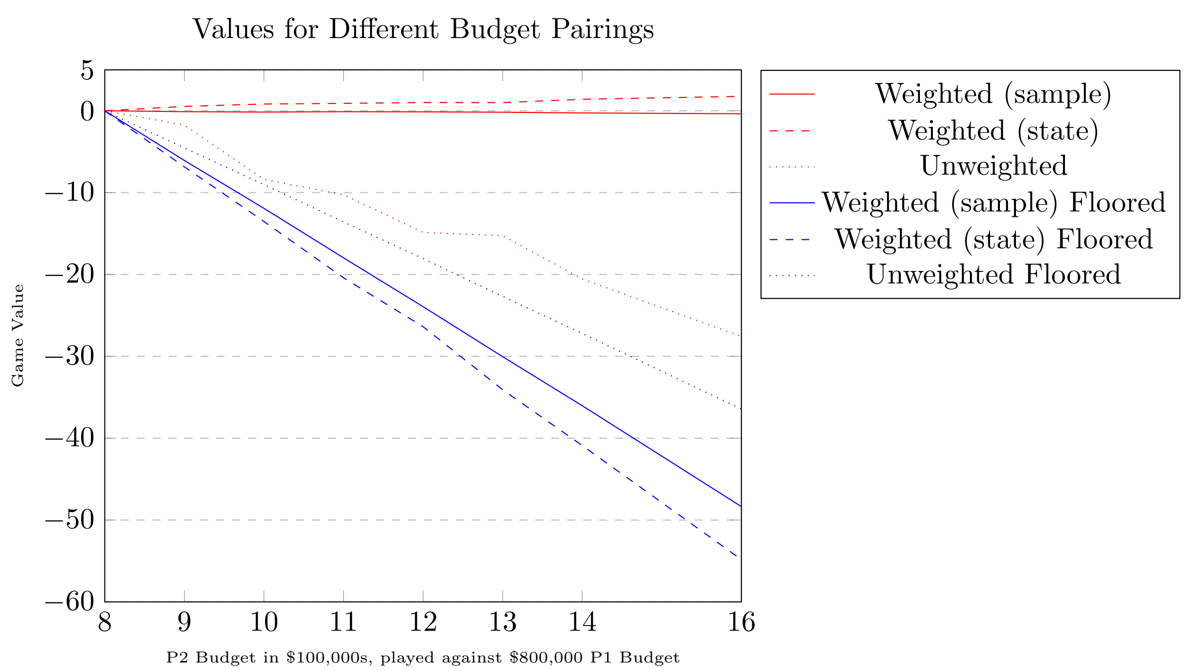


Figure 22: Game values using different utility vectors, using different budgets for P2 against an $800,000 P1 budget. The fairest games are closest to the x-axis, while unfair games have average slopes with high absolute values. The fairest game in this graph is created by the Weighted by Sample, Unfloored Utility Vector, while the most unfair is created by the Weighted by State, Floored Utility Vector.

6 Exploration

While the Unweighted, Floored Utility Vector was chosen as the most realistic, it's important to explore different results given by different utility vectors. It's especially important to explore different vectors in this report, since all the utility vectors are probably very inaccurate. This section contains theoretical results, had different utility vectors been chosen.

6.1 Weighted by State, Floored Utility Vector

This utility vector was calculated very similarly to the original vector used in the game rules (Unweighted, Floored Utility Vector). This is because the vector was calculated using methods very similar to that used to calculate the original vector. In fact, for each individual race, the regression slope was calculated exactly the same, and floored in the same way. The main difference in this calculation was finding the final value for each quarter. While the original was a simple average of the regression slopes calculated for each race in a quarter, the Weighted by State, Floored Utility Vector was a weighted average.

To calculate weights, the state population for each state with enough data for each quarter was determined using U.S. Census estimates. Note that, since not all states have a Senate race in 2018, and some were not covered closely enough to yield straw polls, each quarter used data from a different subset of states. After state population was determined for each state, the sum of populations for all state races used that quarter was calculated. No state in 2018 had more than one Senate race, so no state population was counted twice. The weight given to each race was calculated by

$$w_{state} = \frac{pop_{state}}{pop_{sum}}, \tag{4}$$

where w_{state} denotes the weight assigned to each state's senate race, pop_{state} denotes the population of any state with a 2018 Senate race, and pop_{sum} denotes the sum of all pop_{state} values that were calculated. The final value for each quarter, then, was given by

$$v_{q_k} = \sum w_{state} \mid state \in K, \tag{5}$$

where q_k denotes the value of the vector in quarter k, and K denotes the set of states with enough data to calculate an individual regression slope in quarter K.

Recall that the original game vector took the form

$$[0.611974882, 0.214168992, 0.234584564, 2.111203038] \cdot 2 \times 10^{-5}. \tag{6}$$

The Weighted by State, Floored Utility Vector, by contrast, takes the form

$$[0.243860933, 0.282414783, 0.14780057, 3.216185277] \cdot 2 \times 10^{-5}. \tag{7}$$

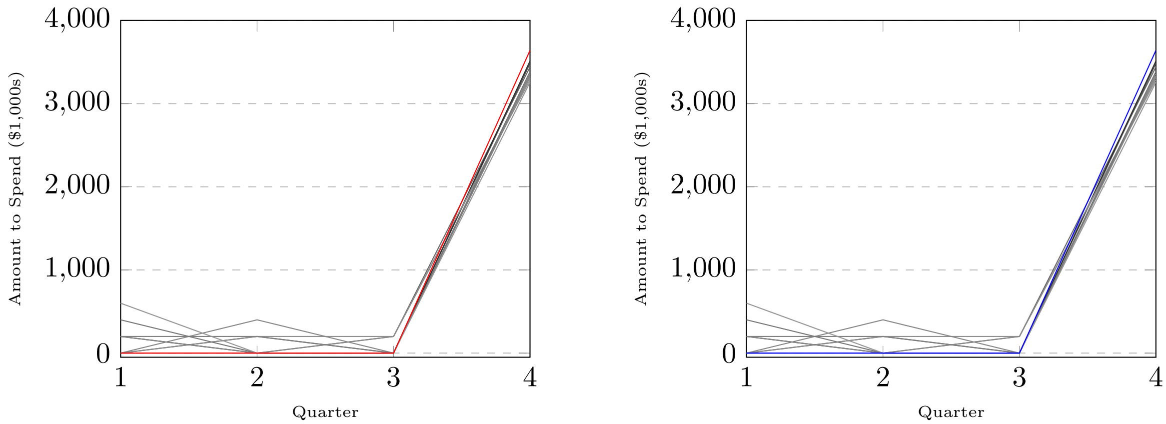
Note that when compared to the original utility vector, the Weighted by State, Floored Utility Vector grants a higher reward for spending in Q2 and Q4, and a lower reward for spending in Q1 and Q3. As a result, the top strategy for this vector is the same as for the original vector — each player spends no money until the final quarter, and then spends their entire budget. In fact, the first difference in strategies between these two vectors is the fourth-most-used strategy. In the original vector, this strategy — for either player, regardless of budget — is to spend \$400,000 in Q1, then the rest in Q4. When using this alternative vector, however, the fourth-most-used strategy is to spend \$200,000 in Q3. This is the result of a much lower voter return in Q1; while in the original vector the return was about 9.79, in this alternative vector, the return is 3.901.

Similar to the original utility vector, this vector necessitates optimal strategies that spend the majority of their budget in Q4. The strategies are ordered, almost without exception, by the amount spent in Q4. The first exception to this rule in any pair of budgets occurs in the player with a larger budget, and always occurs in a strategy used less than 5% of the time.

Note that, like the original utility vector, this vector yields only wins for P2. This happens for the same reason as before — because players are never punished for spending more, a higher-budget candidate always has a clear advantage.

P1 $800,000 vs. P2 $800,000									
Player 1	(P2)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.234%	$0	$0	$0	$3,640,675	29.591%
$200,000	$0	$0	$3,512,868	20.754%	$200,000	$0	$0	$3,512,868	20.791%
$0	$200,000	$0	$3,485,635	14.811%	$0	$200,000	$0	$3,485,635	14.686%
$0	$0	$200,000	$3,440,675	10.323%	$0	$0	$200,000	$3,440,675	10.404%
$400,000	$0	$0	$3,385,060	7.451%	$400,000	$0	$0	$3,385,060	7.319%
$200,000	$200,000	$0	$3,357,828	5.240%	$200,000	$200,000	$0	$3,357,828	5.085%
$0	$400,000	$0	$3,330,595	3.570%	$0	$400,000	$0	$3,330,595	3.651%
$200,000	$0	$200,000	$3,312,868	2.611%	$200,000	$0	$200,000	$3,312,868	2.566%
$0	$200,000	$200,000	$3,285,635	1.811%	$0	$200,000	$200,000	$3,285,635	1.773%
$600,000	$0	$0	$3,257,252	1.260%	$600,000	$0	$0	$3,257,252	1.210%
Other				2.935%	Other				2.924%
Game Value:			**-0.01**		**Winner:**			**P2**	

Figure 23: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $800,000, and the Weighted by State, Floored Utility Vector is used.

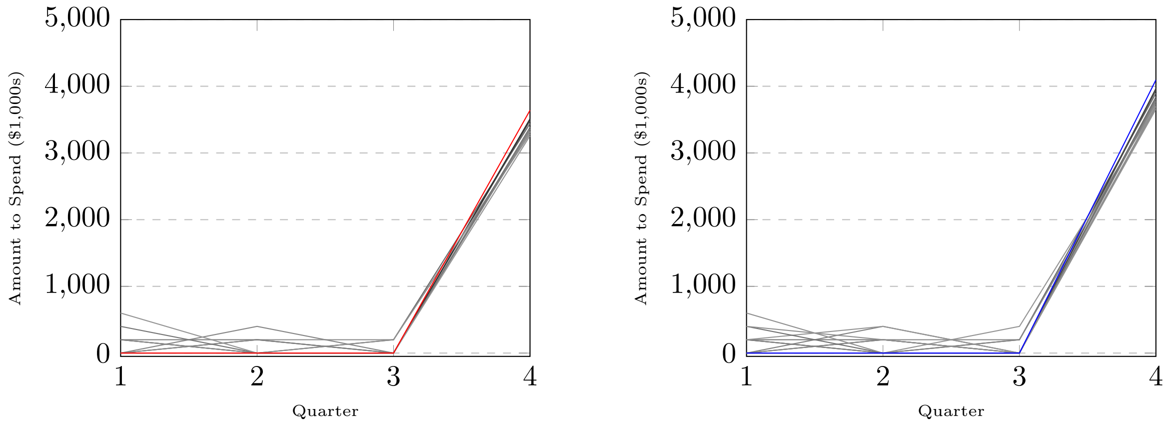


a P1: $800,000 Initial Budget **b** P2: $800,000 Initial Budget

Figure 24: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $900,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.371%	$0	$0	$0	$4,095,760	23.150%
$200,000	$0	$0	$3,512,868	20.968%	$200,000	$0	$0	$3,967,952	17.473%
$0	$200,000	$0	$3,485,635	14.613%	$0	$200,000	$0	$3,940,720	13.577%
$0	$0	$200,000	$3,440,675	10.487%	$0	$0	$200,000	$3,895,760	10.488%
$400,000	$0	$0	$3,385,060	7.308%	$400,000	$0	$0	$3,840,145	8.122%
$200,000	$200,000	$0	$3,357,828	5.157%	$200,000	$200,000	$0	$3,812,912	6.270%
$0	$400,000	$0	$3,330,595	3.671%	$0	$400,000	$0	$3,785,680	4.791%
$200,000	$0	$200,000	$3,312,868	2.467%	$200,000	$0	$200,000	$3,767,952	3.654%
$0	$200,000	$200,000	$3,285,635	1.802%	$0	$200,000	$200,000	$3,740,720	2.915%
$600,000	$0	$0	$3,257,252	1.310%	$600,000	$0	$0	$3,712,337	2.239%
					$400,000	$200,000	$0	$3,685,105	1.743%
					$0	$0	$400,000	$3,695,760	1.263%
					$200,000	$400,000	$0	$3,657,872	1.022%
Other				2.846%	Other				3.293%
Game Value:			**-6.85**		**Winner:**			**P2**	

Figure 25: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $900,000, and the Weighted by State, Floored Utility Vector is used.



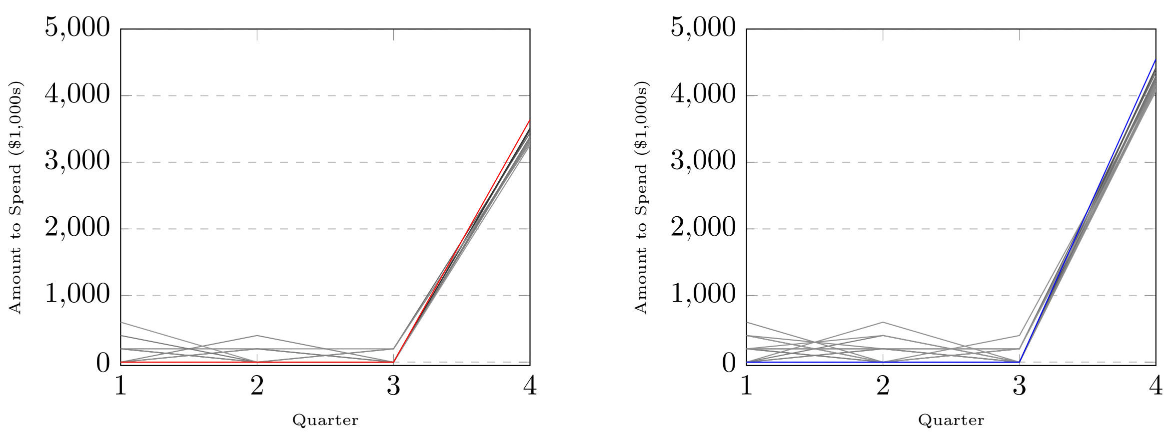
a P1: $800,000 Initial Budget

b P2: $900,000 Initial Budget

Figure 26: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,000,000									
Player 1 (P1)					Player 2 (P2)				
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.295%	$0	$0	$0	$4,550,844	16.087%
$200,000	$0	$0	$3,512,868	20.912%	$200,000	$0	$0	$4,423,037	13.585%
$0	$200,000	$0	$3,485,635	14.838%	$0	$200,000	$0	$4,395,804	11.431%
$0	$0	$200,000	$3,440,675	10.437%	$0	$0	$200,000	$4,350,844	9.493%
$400,000	$0	$0	$3,385,060	7.216%	$400,000	$0	$0	$4,295,229	8.080%
$200,000	$200,000	$0	$3,357,828	5.230%	$200,000	$200,000	$0	$4,267,997	6.785%
$0	$400,000	$0	$3,330,595	3.575%	$0	$400,000	$0	$4,240,764	5.596%
$200,000	$0	$200,000	$3,312,868	2.507%	$200,000	$0	$200,000	$4,223,037	4.715%
$0	$200,000	$200,000	$3,285,635	1.770%	$0	$200,000	$200,000	$4,195,804	3.918%
$600,000	$0	$0	$3,257,252	1.303%	$600,000	$0	$0	$4,167,421	3.203%
					$400,000	$200,000	$0	$4,140,189	2.859%
					$0	$0	$400,000	$4,150,844	2.384%
					$200,000	$400,000	$0	$4,112,957	1.919%
					$0	$600,000	$0	$4,085,724	1.675%
					$400,000	$0	$200,000	$4,095,229	1.449%
					$200,000	$200,000	$200,000	$4,067,997	1.149%
Other				2.917%	Other				5.672%
Game Value:			**-13.53**		**Winner:**			**P2**	

Figure 27: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,000,000, and the Weighted by State, Floored Utility Vector is used.



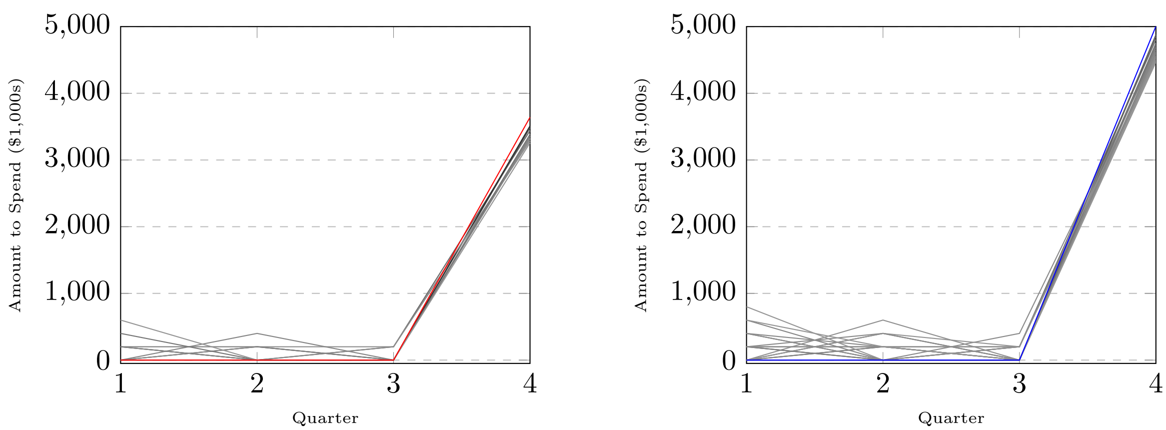
a P1: $800,000 Initial Budget

b P2: $1,000,000 Initial Budget

Figure 28: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,100,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.610%	$0	$0	$0	$5,005,929	13.086%
$200,000	$0	$0	$3,512,868	20.664%	$200,000	$0	$0	$4,878,121	11.437%
$0	$200,000	$0	$3,485,635	14.628%	$0	$200,000	$0	$4,850,889	9.759%
$0	$0	$200,000	$3,440,675	10.406%	$0	$0	$200,000	$4,805,929	8.420%
$400,000	$0	$0	$3,385,060	7.226%	$400,000	$0	$0	$4,750,314	7.302%
$200,000	$200,000	$0	$3,357,828	5.157%	$200,000	$200,000	$0	$4,723,081	6.499%
$0	$400,000	$0	$3,330,595	3.691%	$0	$400,000	$0	$4,695,849	5.637%
$200,000	$0	$200,000	$3,312,868	2.587%	$200,000	$0	$200,000	$4,678,121	4.878%
$0	$200,000	$200,000	$3,285,635	1.795%	$0	$200,000	$200,000	$4,650,889	4.343%
$600,000	$0	$0	$3,257,252	1.311%	$600,000	$0	$0	$4,622,506	3.811%
					$400,000	$200,000	$0	$4,595,274	3.230%
					$0	$0	$400,000	$4,605,929	2.884%
					$200,000	$400,000	$0	$4,568,041	2.518%
					$0	$600,000	$0	$4,540,809	2.144%
					$400,000	$0	$200,000	$4,550,314	1.869%
					$200,000	$200,000	$200,000	$4,523,081	1.584%
					$800,000	$0	$0	$4,494,698	1.396%
					$0	$400,000	$200,000	$4,495,849	1.181%
					$600,000	$200,000	$0	$4,467,466	1.056%
Other				2.925%	Other				6.966%
Game Value:			**-20.40**		**Winner:**			**P2**	

Figure 29: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,100,000, and the Weighted by State, Floored Utility Vector is used.

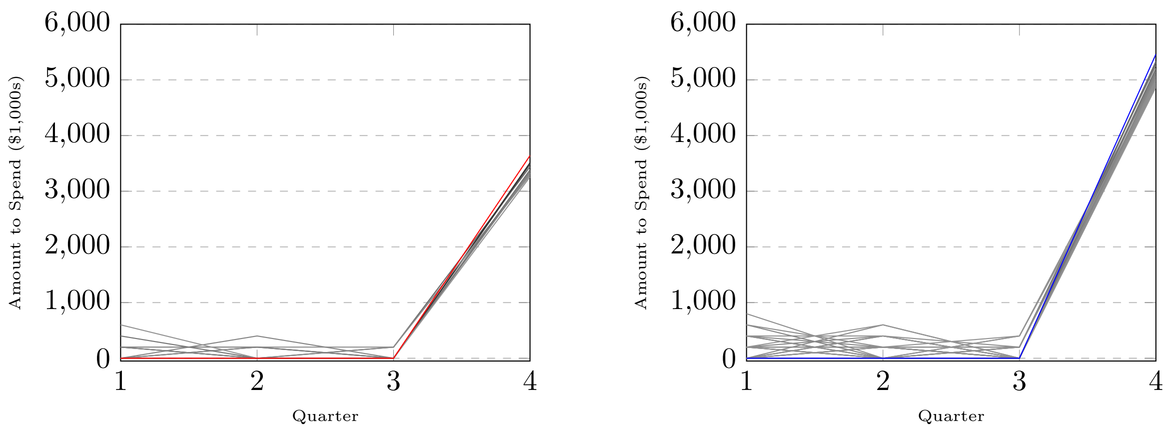


a P1: $800,000 Initial Budget **b** P2: $1,100,000 Initial Budget

Figure 30: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,200,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.290%	$0	$0	$0	$5,461,013	9.873%
$200,000	$0	$0	$3,512,868	20.772%	$200,000	$0	$0	$5,333,206	8.695%
$0	$200,000	$0	$3,485,635	14.696%	$0	$200,000	$0	$5,305,973	7.907%
$0	$0	$200,000	$3,440,675	10.443%	$0	$0	$200,000	$5,261,013	6.977%
$400,000	$0	$0	$3,385,060	7.354%	$400,000	$0	$0	$5,205,398	6.534%
$200,000	$200,000	$0	$3,357,828	5.212%	$200,000	$200,000	$0	$5,178,166	5.893%
$0	$400,000	$0	$3,330,595	3.686%	$0	$400,000	$0	$5,150,933	5.355%
$200,000	$0	$200,000	$3,312,868	2.538%	$200,000	$0	$200,000	$5,133,206	4.833%
$0	$200,000	$200,000	$3,285,635	1.816%	$0	$200,000	$200,000	$5,105,973	4.430%
$600,000	$0	$0	$3,257,252	1.336%	$600,000	$0	$0	$5,077,590	3.905%
					$400,000	$200,000	$0	$5,050,358	3.461%
					$0	$0	$400,000	$5,061,013	3.094%
					$200,000	$400,000	$0	$5,023,126	2.801%
					$0	$600,000	$0	$4,995,893	2.636%
					$400,000	$0	$200,000	$5,005,398	2.369%
					$200,000	$200,000	$200,000	$4,978,166	2.099%
					$800,000	$0	$0	$4,949,783	1.901%
					$0	$400,000	$200,000	$4,950,933	1.706%
					$600,000	$200,000	$0	$4,922,550	1.507%
					$200,000	$0	$400,000	$4,933,206	1.409%
					$400,000	$400,000	$0	$4,895,318	1.186%
					$0	$200,000	$400,000	$4,905,973	1.116%
					$200,000	$600,000	$0	$4,868,086	1.049%
Other				2.857%	Other				9.264%
Game Value:			**-26.38**		**Winner:**			**P2**	

Figure 31: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,200,000, and the Weighted by State, Floored Utility Vector is used.

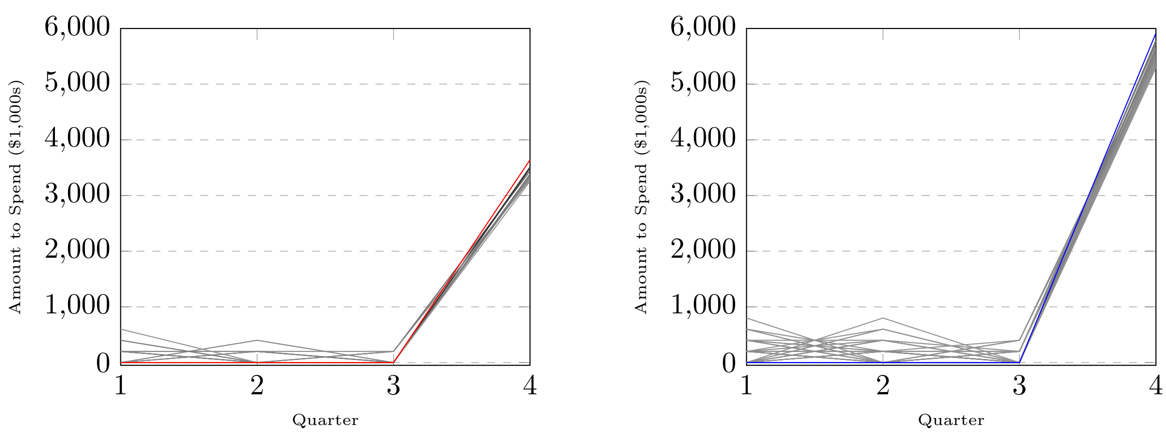


a P1: $800,000 Initial Budget **b** P2: $1,200,000 Initial Budget

Figure 32: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,300,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.488%	$0	$0	$0	$5,916,098	7.910%
$200,000	$0	$0	$3,512,868	20.956%	$200,000	$0	$0	$5,788,290	7.383%
$0	$200,000	$0	$3,485,635	14.687%	$0	$200,000	$0	$5,761,058	6.898%
$0	$0	$200,000	$3,440,675	10.497%	$0	$0	$200,000	$5,716,098	6.328%
$400,000	$0	$0	$3,385,060	7.262%	$400,000	$0	$0	$5,660,483	5.774%
$200,000	$200,000	$0	$3,357,828	5.170%	$200,000	$200,000	$0	$5,633,250	5.447%
$0	$400,000	$0	$3,330,595	3.668%	$0	$400,000	$0	$5,606,018	4.969%
$200,000	$0	$200,000	$3,312,868	2.513%	$200,000	$0	$200,000	$5,588,290	4.484%
$0	$200,000	$200,000	$3,285,635	1.691%	$0	$200,000	$200,000	$5,561,058	4.104%
$600,000	$0	$0	$3,257,252	1.235%	$600,000	$0	$0	$5,532,675	3.889%
					$400,000	$200,000	$0	$5,505,443	3.547%
					$0	$0	$400,000	$5,516,098	3.272%
					$200,000	$400,000	$0	$5,478,210	2.824%
					$0	$600,000	$0	$5,450,978	2.803%
					$400,000	$0	$200,000	$5,460,483	2.442%
					$200,000	$200,000	$200,000	$5,433,250	2.339%
					$800,000	$0	$0	$5,404,867	2.115%
					$0	$400,000	$200,000	$5,406,018	1.967%
					$600,000	$200,000	$0	$5,377,635	1.787%
					$200,000	$0	$400,000	$5,388,290	1.619%
					$400,000	$400,000	$0	$5,350,403	1.479%
					$0	$200,000	$400,000	$5,361,058	1.366%
					$200,000	$600,000	$0	$5,323,170	1.333%
					$600,000	$0	$200,000	$5,332,675	1.145%
					$0	$800,000	$0	$5,295,938	1.083%
Other				2.833%	Other				11.693%
Game Value:			**-34.08**		**Winner:**			**P2**	

Figure 33: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,300,000, and the Weighted by State, Floored Utility Vector is used.

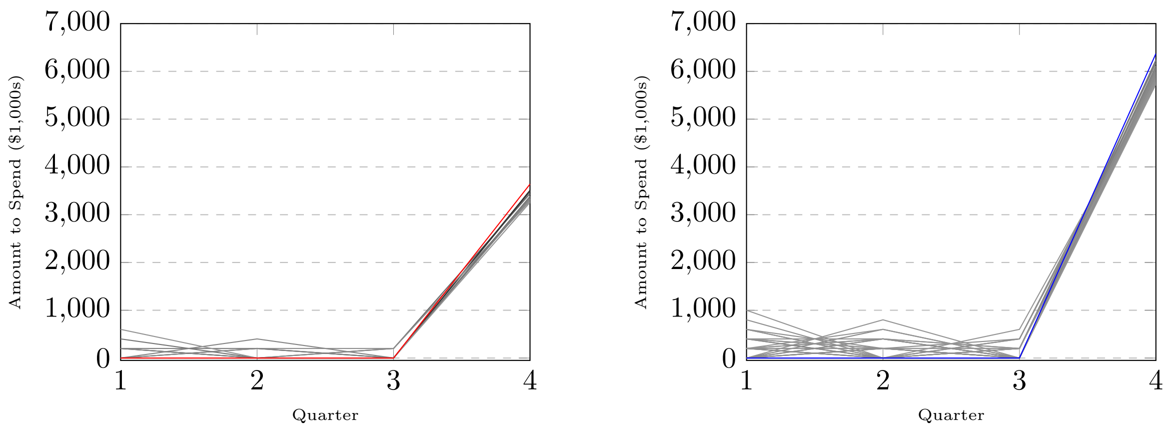


a P1: $800,000 Initial Budget **b** P2: $1,300,000 Initial Budget

Figure 34: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,400,000									
Player 1 (P1)					Player 2 (P2)				
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.600%	$0	$0	$0	$6,371,182	6.444%
$200,000	$0	$0	$3,512,868	20.779%	$200,000	$0	$0	$6,243,375	5.905%
$0	$200,000	$0	$3,485,635	14.723%	$0	$200,000	$0	$6,216,142	5.641%
$0	$0	$200,000	$3,440,675	10.310%	$0	$0	$200,000	$6,171,182	5.240%
$400,000	$0	$0	$3,385,060	7.338%	$400,000	$0	$0	$6,115,567	4.930%
$200,000	$200,000	$0	$3,357,828	5.077%	$200,000	$200,000	$0	$6,088,335	4.464%
$0	$400,000	$0	$3,330,595	3.623%	$0	$400,000	$0	$6,061,102	4.305%
$200,000	$0	$200,000	$3,312,868	2.552%	$200,000	$0	$200,000	$6,043,375	4.048%
$0	$200,000	$200,000	$3,285,635	1.755%	$0	$200,000	$200,000	$6,016,142	3.740%
$600,000	$0	$0	$3,257,252	1.249%	$600,000	$0	$0	$5,987,759	3.449%
					$400,000	$200,000	$0	$5,960,527	3.286%
					$0	$0	$400,000	$5,971,182	3.089%
					$200,000	$400,000	$0	$5,933,295	2.809%
					$0	$600,000	$0	$5,906,062	2.764%
					$400,000	$0	$200,000	$5,915,567	2.529%
					$200,000	$200,000	$200,000	$5,888,335	2.438%
					$800,000	$0	$0	$5,859,952	2.185%
					$0	$400,000	$200,000	$5,861,102	2.073%
					$600,000	$200,000	$0	$5,832,719	2.020%
					$200,000	$0	$400,000	$5,843,375	1.768%
					$400,000	$400,000	$0	$5,805,487	1.756%
					$0	$200,000	$400,000	$5,816,142	1.643%
					$200,000	$600,000	$0	$5,778,255	1.482%
					$600,000	$0	$200,000	$5,787,759	1.430%
					$0	$800,000	$0	$5,751,022	1.3580%
					$400,000	$200,000	$200,000	$5,760,527	1.229%
					$1,000,000	$0	$0	$5,732,144	1.168%
					$0	$0	$600,000	$5,771,182	1.058%
					$200,000	$400,000	$200,000	$5,733,295	1.053%
Other				2.994%	Other				14.696%
Game Value:			**-40.90**		**Winner:**			**P2**	

Figure 35: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,400,000, and the Weighted by State, Floored Utility Vector is used.

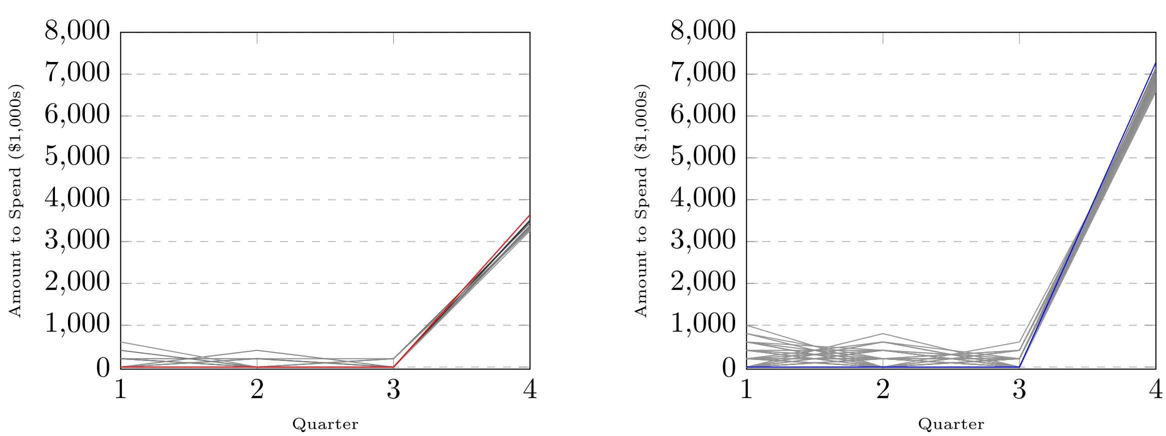


a P1: $800,000 Initial Budget **b** P2: $1,400,000 Initial Budget

Figure 36: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,600,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.361%	$0	$0	$0	$7,281,351	4.402%
$200,000	$0	$0	$3,512,868	20.672%	$200,000	$0	$0	$7,153,544	4.141%
$0	$200,000	$0	$3,485,635	14.740%	$0	$200,000	$0	$7,126,311	3.950%
$0	$0	$200,000	$3,440,675	10.491%	$0	$0	$200,000	$7,081,351	3.931%
$400,000	$0	$0	$3,385,060	7.263%	$400,000	$0	$0	$7,025,736	3.642%
$200,000	$200,000	$0	$3,357,828	5.169%	$200,000	$200,000	$0	$6,998,504	3.461%
$0	$400,000	$0	$3,330,595	3.751%	$0	$400,000	$0	$6,971,271	3.252%
$200,000	$0	$200,000	$3,312,868	2.598%	$200,000	$0	$200,000	$6,953,544	3.173%
$0	$200,000	$200,000	$3,285,635	1.776%	$0	$200,000	$200,000	$6,926,311	3.029%
$600,000	$0	$0	$3,257,252	1.236%	$600,000	$0	$0	$6,897,928	2.918%
					$0	$0	$400,000	$6,881,351	2.740%
					$400,000	$200,000	$0	$6,870,696	2.727%
					$200,000	$400,000	$0	$6,843,464	2.525%
					$0	$600,000	$0	$6,816,231	2.465%
					$400,000	$0	$200,000	$6,825,736	2.314%
					$200,000	$200,000	$200,000	$6,798,504	2.195%
					$800,000	$0	$0	$6,770,121	2.111%
					$0	$400,000	$200,000	$6,771,271	2.036%
					$200,000	$0	$400,000	$6,753,544	1.964%
					$600,000	$200,000	$0	$6,742,888	1.909%
					$400,000	$400,000	$0	$6,715,656	1.749%
					$0	$200,000	$400,000	$6,726,311	1.741%
					$600,000	$0	$200,000	$6,697,928	1.623%
					$200,000	$600,000	$0	$6,688,424	1.613%
					$0	$800,000	$0	$6,661,191	1.494%
					$400,000	$200,000	$200,000	$6,670,696	1.416%
					$1,000,000	$0	$0	$6,642,313	1.406%
					$0	$0	$600,000	$6,681,351	1.321%
					$200,000	$400,000	$200,000	$6,643,464	1.287%
					$800,000	$200,000	$0	$6,615,081	1.272%
					$0	$600,000	$200,000	$6,616,231	1.122%
					$400,000	$0	$400,000	$6,625,736	1.087%
					$600,000	$400,000	$0	$6,587,848	1.064%
					$200,000	$200,000	$400,000	$6,598,504	1.051%
Other				2.943%	Other				21.869%
Game Value:			**-54.83**		**Winner:**			**P2**	

Figure 37: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,600,000, and the Weighted by State, Floored Utility Vector is used.



a P1: $800,000 Initial Budget

b P2: $1,600,000 Initial Budget

Figure 38: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $800,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.374%	$0	$0	$0	$3,640,675	29.455%
$200,000	$0	$0	$3,512,868	20.795%	$200,000	$0	$0	$3,512,868	20.855%
$0	$200,000	$0	$3,485,635	14.709%	$0	$200,000	$0	$3,485,635	14.614%
$0	$0	$200,000	$3,440,675	10.318%	$0	$0	$200,000	$3,440,675	10.320%
$400,000	$0	$0	$3,385,060	7.424%	$400,000	$0	$0	$3,385,060	7.539%
$200,000	$200,000	$0	$3,357,828	5.154%	$200,000	$200,000	$0	$3,357,828	5.056%
$0	$400,000	$0	$3,330,595	3.681%	$0	$400,000	$0	$3,330,595	3.682%
$200,000	$0	$200,000	$3,312,868	2.572%	$200,000	$0	$200,000	$3,312,868	2.551%
$0	$200,000	$200,000	$3,285,635	1.805%	$0	$200,000	$200,000	$3,285,635	1.822%
$600,000	$0	$0	$3,257,252	1.259%	$600,000	$0	$0	$3,257,252	1.232%
Other				2.909%	Other				2.874%
Game Value:			**0.00**		**Winner:**			**P2**	

Figure 39: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $800,000, and the Weighted by Sample, Floored Utility Vector is used.

6.2 Weighted by Sample, Floored Utility Vector

This vector was calculated extremely similarly to the Weighted by State, Floored Utility Vector. Like that vector, each individual race's regression slope was calculated the same as the original vector (Unweighted, Floored Utility Vector). The only difference in this calculation was the way in which the weights were found. In this vector, the weights were calculated by

$$w_{state} = \frac{smp_{state}}{smp_{sum}}, \tag{8}$$

where w_{state} denotes the weight assigned to each state's senate race, $pop_{smp_{state}}$ denotes the total sample size used in straw polls for a given quarter of any state with a 2018 Senate race, and smp_{sum} denotes the sum of all smp_{state} values that were calculated. The final value for each quarter, then, was given by

$$v_{q_k} = \sum smp_{state} \mid state \in K, \tag{9}$$

where q_k denotes the value of the vector in quarter k, and K denotes the set of states with enough data to calculate an individual regression slope in quarter K.

Recall again that the original utility vector was

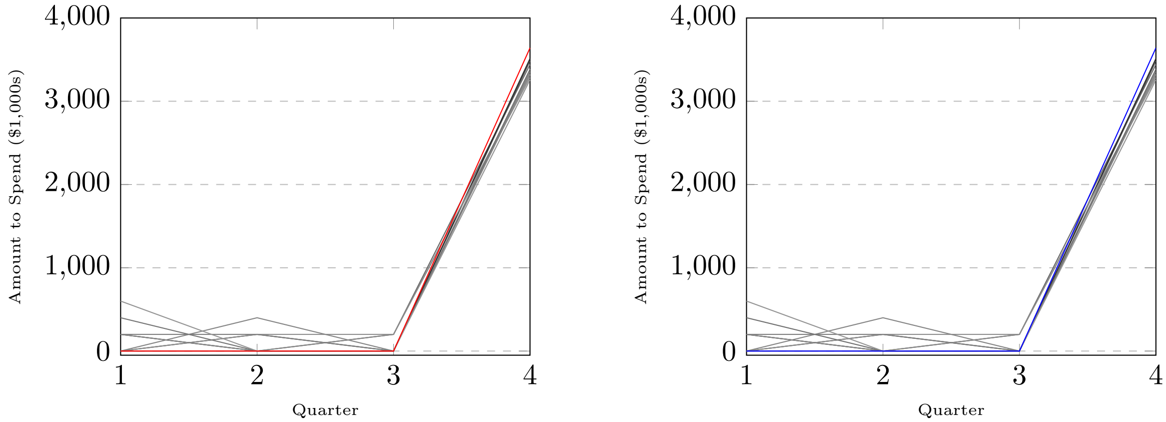
$$[0.611974882, 0.214168992, 0.234584564, 2.111203038] \cdot 2 \times 10^{-5}, \tag{10}$$

while the Weighted by Sample, Floored Utility Vector is

$$[0.167306442, 0.089473901, 0.11420262, 2.841187593] \cdot 2 \times 10^{-5}. \tag{11}$$

This vector places the most value on Q4 out of any vector considered, when comparing proportions of vector values. It's unsurprising that these strategies, too, are ordered almost perfectly in order of Q4 spending values. When viewed in context of the past two vectors, this vector yielded relatively uninteresting strategies. Like the Unweighted, Floored Utility Vector and Weighted by State, Floored Utility Vectors, any strategy that did not follow this order was used less than 5% of the time.

Like the previous two vectors, this vector also yields only wins for P2, since players are never punished for spending money. This gives a clear advantage to higher-budget candidates.

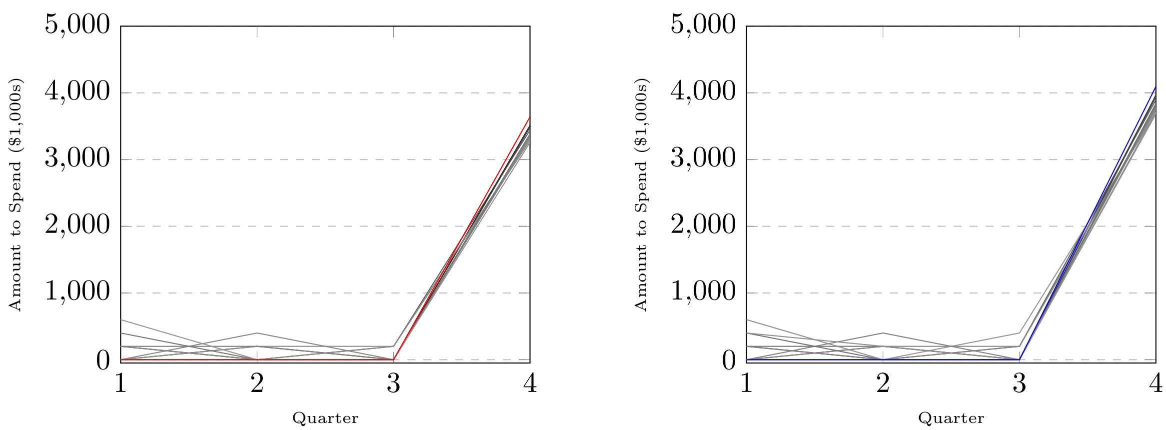


a P1: $800,000 Initial Budget **b** P2: $800,000 Initial Budget

Figure 40: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $900,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.360%	$0	$0	$0	$4,095,760	22.927%
$200,000	$0	$0	$3,512,868	20.772%	$200,000	$0	$0	$3,967,952	17.767%
$0	$200,000	$0	$3,485,635	14.72%	$0	$200,000	$0	$3,940,720	13.639%
$0	$0	$200,000	$3,440,675	10.628%	$0	$0	$200,000	$3,895,760	10.629%
$400,000	$0	$0	$3,385,060	7.255%	$400,000	$0	$0	$3,840,145	8.216%
$200,000	$200,000	$0	$3,357,828	5.098%	$200,000	$200,000	$0	$3,812,912	6.184%
$0	$400,000	$0	$3,330,595	3.714%	$0	$400,000	$0	$3,785,680	4.927%
$200,000	$0	$200,000	$3,312,868	2.607%	$200,000	$0	$200,000	$3,767,952	3.600%
$0	$200,000	$200,000	$3,285,635	1.758%	$0	$200,000	$200,000	$3,740,720	2.790%
$600,000	$0	$0	$3,257,252	1.195%	$600,000	$0	$0	$3,712,337	2.223%
					$400,000	$200,000	$0	$3,685,105	1.680%
					$0	$0	$400,000	$3,695,760	1.273%
Other				2.893%	Other				4.145%
Game Value:			**-6.07**		**Winner:**			**P2**	

Figure 41: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $900,000, and the Weighted by Sample, Floored Utility Vector is used.

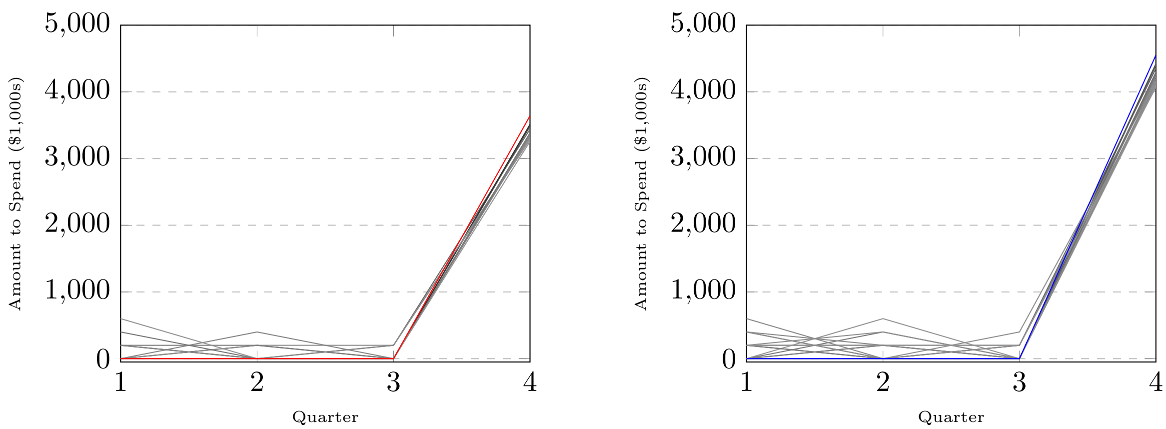


a P1: $800,000 Initial Budget **b** P2: $900,000 Initial Budget

Figure 42: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,000,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.312%	$0	$0	$0	$4,550,844	16.166%
$200,000	$0	$0	$3,512,868	20.898%	$200,000	$0	$0	$4,423,037	13.644%
$0	$200,000	$0	$3,485,635	14.752%	$0	$200,000	$0	$4,395,804	11.331%
$0	$0	$200,000	$3,440,675	10.347%	$0	$0	$200,000	$4,350,844	9.676%
$400,000	$0	$0	$3,385,060	7.333%	$400,000	$0	$0	$4,295,229	8.085%
$200,000	$200,000	$0	$3,357,828	5.208%	$200,000	$200,000	$0	$4,267,997	6.846%
$0	$400,000	$0	$3,330,595	3.675%	$0	$400,000	$0	$4,240,764	5.616%
$200,000	$0	$200,000	$3,312,868	2.544%	$200,000	$0	$200,000	$4,223,037	4.720%
$0	$200,000	$200,000	$3,285,635	1.759%	$0	$200,000	$200,000	$4,195,804	3.835%
$600,000	$0	$0	$3,257,252	1.260%	$600,000	$0	$0	$4,167,421	3.357%
					$400,000	$200,000	$0	$4,140,189	2.751%
					$0	$0	$400,000	$4,150,844	2.255%
					$200,000	$400,000	$0	$4,112,957	1.929%
					$400,000	$0	$200,000	$4,095,229	1.615%
					$0	$600,000	$0	$4,085,724	1.365%
					$200,000	$200,000	$200,000	$4,067,997	1.165%
Other				2.912%	Other				5.644%
Game Value:			**-11.92**		**Winner:**			**P2**	

Figure 43: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,000,000, and the Weighted by Sample, Floored Utility Vector is used.



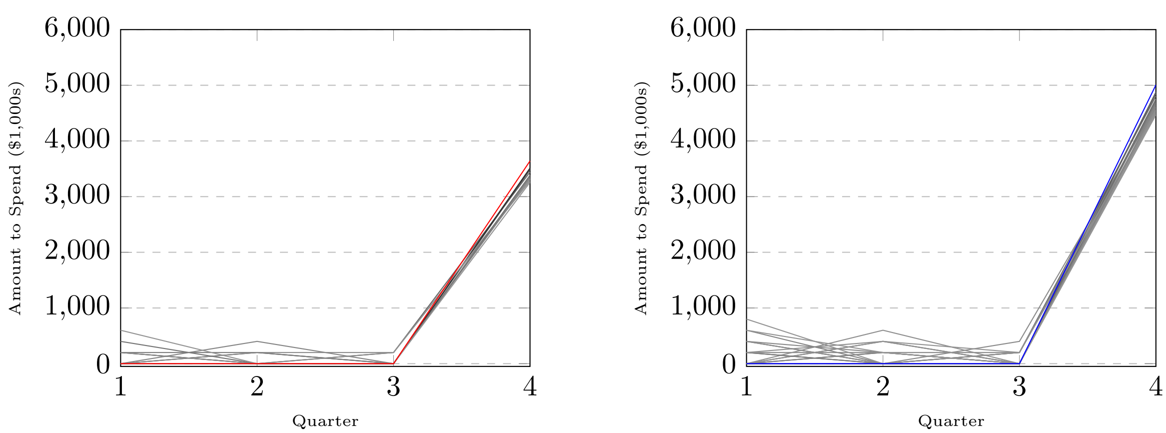
a P1: $800,000 Initial Budget

b P2: $1,000,000 Initial Budget

Figure 44: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,100,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.475%	$0	$0	$0	$5,005,929	12.809%
$200,000	$0	$0	$3,512,868	20.767%	$200,000	$0	$0	$4,878,121	11.402%
$0	$200,000	$0	$3,485,635	14.858%	$0	$200,000	$0	$4,850,889	9.86%
$0	$0	$200,000	$3,440,675	10.415%	$0	$0	$200,000	$4,805,929	8.652%
$400,000	$0	$0	$3,385,060	7.340%	$400,000	$0	$0	$4,750,314	7.600%
$200,000	$200,000	$0	$3,357,828	5.130%	$200,000	$200,000	$0	$4,723,081	6.502%
$0	$400,000	$0	$3,330,595	3.627%	$0	$400,000	$0	$4,695,849	5.708%
$200,000	$0	$200,000	$3,312,868	2.434%	$200,000	$0	$200,000	$4,678,121	5.006%
$0	$200,000	$200,000	$3,285,635	1.784%	$0	$200,000	$200,000	$4,650,889	4.272%
$600,000	$0	$0	$3,257,252	1.272%	$600,000	$0	$0	$4,622,506	3.736%
					$400,000	$200,000	$0	$4,595,274	3.176%
					$0	$0	$400,000	$4,605,929	2.845%
					$200,000	$400,000	$0	$4,568,041	2.387%
					$400,000	$0	$200,000	$4,550,314	2.128%
					$0	$600,000	$0	$4,540,809	1.835%
					$200,000	$200,000	$200,000	$4,523,081	1.620%
					$800,000	$0	$0	$4,494,698	1.407%
					$0	$400,000	$200,000	$4,495,849	1.223%
					$600,000	$200,000	$0	$4,467,466	1.048%
Other				2.898%	Other				6.784%
Game Value:			**-17.96**		**Winner:**			**P2**	

Figure 45: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,100,000, and the Weighted by Sample, Floored Utility Vector is used.

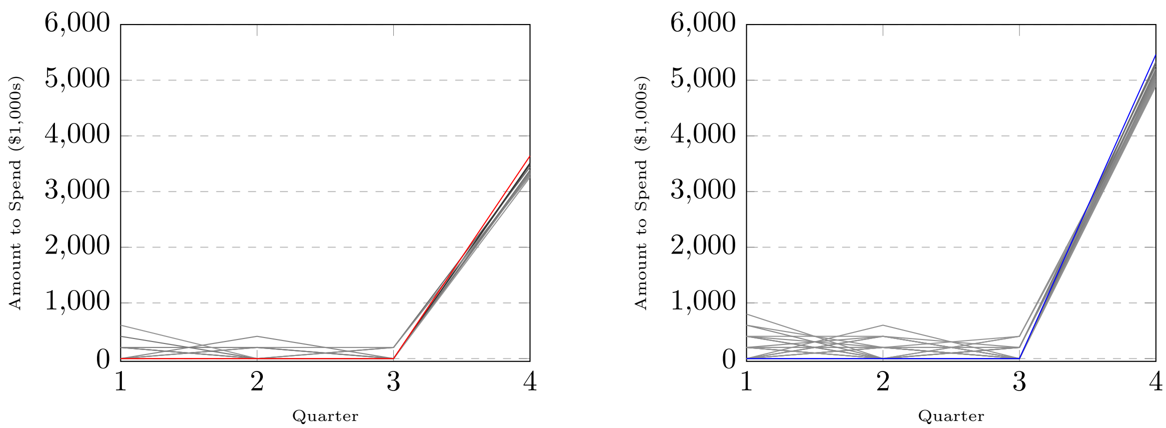


a P1: $800,000 Initial Budget **b** P2: $1,100,000 Initial Budget

Figure 46: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,200,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.231%	$0	$0	$0	$5,461,013	9.667%
$200,000	$0	$0	$3,512,868	20.818%	$200,000	$0	$0	$5,333,206	8.807%
$0	$200,000	$0	$3,485,635	14.767%	$0	$200,000	$0	$5,305,973	8.139%
$0	$0	$200,000	$3,440,675	10.491%	$0	$0	$200,000	$5,261,013	7.061%
$400,000	$0	$0	$3,385,060	7.239%	$400,000	$0	$0	$5,205,398	6.44%
$200,000	$200,000	$0	$3,357,828	5.136%	$200,000	$200,000	$0	$5,178,166	5.835%
$0	$400,000	$0	$3,330,595	3.705%	$0	$400,000	$0	$5,150,933	5.336%
$200,000	$0	$200,000	$3,312,868	2.606%	$200,000	$0	$200,000	$5,133,206	4.797%
$0	$200,000	$200,000	$3,285,635	1.778%	$0	$200,000	$200,000	$5,105,973	4.229%
$600,000	$0	$0	$3,257,252	1.214%	$600,000	$0	$0	$5,077,590	3.861%
					$400,000	$200,000	$0	$5,050,358	3.447%
					$0	$0	$400,000	$5,061,013	3.163%
					$200,000	$400,000	$0	$5,023,126	2.887%
					$400,000	$0	$200,000	$5,005,398	2.633%
					$0	$600,000	$0	$4,995,893	2.381%
					$200,000	$200,000	$200,000	$4,978,166	2.104%
					$800,000	$0	$0	$4,949,783	1.978%
					$0	$400,000	$200,000	$4,950,933	1.701%
					$600,000	$200,000	$0	$4,922,550	1.535%
					$200,000	$0	$400,000	$4,933,206	1.402%
					$400,000	$400,000	$0	$4,895,318	1.302%
					$0	$200,000	$400,000	$4,905,973	1.138%
Other				3.015%	Other				10.157%
Game Value:			**-23.93**		**Winner:**			**P2**	

Figure 47: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,200,000, and the Weighted by Sample, Floored Utility Vector is used.

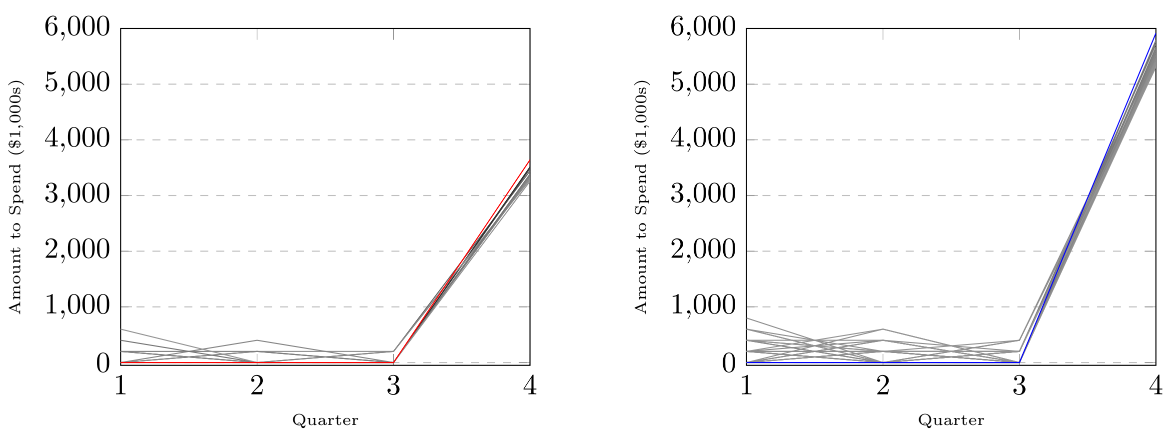


a P1: $800,000 Initial Budget **b** P2: $1,200,000 Initial Budget

Figure 48: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,300,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.470%	$0	$0	$0	$5,916,098	8.306%
$200,000	$0	$0	$3,512,868	20.977%	$200,000	$0	$0	$5,788,290	7.385%
$0	$200,000	$0	$3,485,635	14.591%	$0	$200,000	$0	$5,761,058	6.886%
$0	$0	$200,000	$3,440,675	10.490%	$0	$0	$200,000	$5,716,098	6.375%
$400,000	$0	$0	$3,385,060	7.248%	$400,000	$0	$0	$5,660,483	5.740%
$200,000	$200,000	$0	$3,357,828	5.139%	$200,000	$200,000	$0	$5,633,250	5.217%
$0	$400,000	$0	$3,330,595	3.660%	$0	$400,000	$0	$5,606,018	4.863%
$200,000	$0	$200,000	$3,312,868	2.514%	$200,000	$0	$200,000	$5,588,290	4.501%
$0	$200,000	$200,000	$3,285,635	1.766%	$0	$200,000	$200,000	$5,561,058	4.181%
$600,000	$0	$0	$3,257,252	1.230%	$600,000	$0	$0	$5,532,675	3.752%
					$400,000	$200,000	$0	$5,505,443	3.539%
					$0	$0	$400,000	$5,516,098	3.230%
					$200,000	$400,000	$0	$5,478,210	2.927%
					$400,000	$0	$200,000	$5,460,483	2.702%
					$0	$600,000	$0	$5,450,978	2.481%
					$200,000	$200,000	$200,000	$5,433,250	2.290%
					$800,000	$0	$0	$5,404,867	2.065%
					$0	$400,000	$200,000	$5,406,018	1.933%
					$600,000	$200,000	$0	$5,377,635	1.824%
					$200,000	$0	$400,000	$5,388,290	1.612%
					$400,000	$400,000	$0	$5,350,403	1.450%
					$0	$200,000	$400,000	$5,361,058	1.410%
					$600,000	$0	$200,000	$5,332,675	1.311%
					$200,000	$600,000	$0	$5,323,170	1.128%
					$400,000	$200,000	$200,000	$5,305,443	1.071%
Other				2.915%	Other				11.821%
Game Value:			**-30.04**		**Winner:**			**P2**	

Figure 49: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,300,000, and the Weighted by Sample, Floored Utility Vector is used.

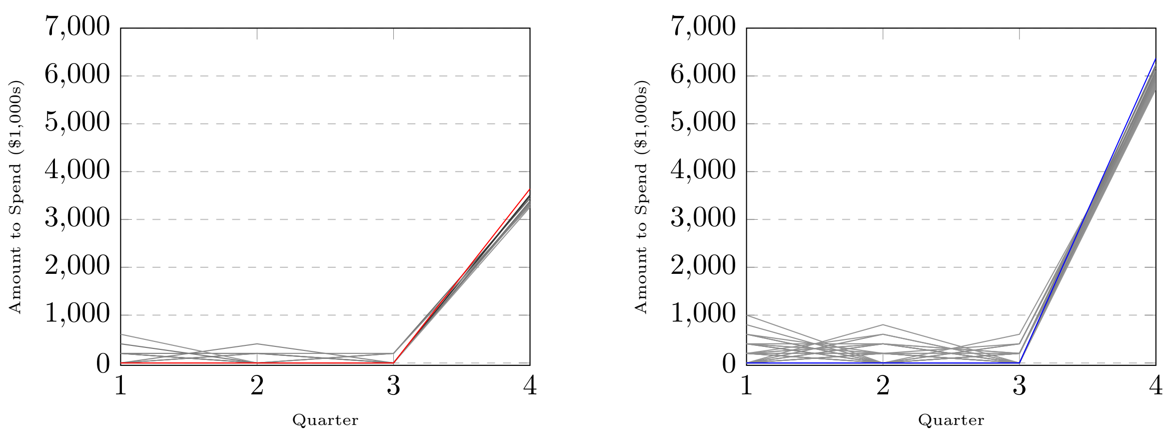


a P1: $800,000 Initial Budget　　**b** P2: $1,300,000 Initial Budget

Figure 50: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,400,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.352%	$0	$0	$0	$6,371,182	6.343%
$200,000	$0	$0	$3,512,868	21.08%	$200,000	$0	$0	$6,243,375	5.891%
$0	$200,000	$0	$3,485,635	14.754%	$0	$200,000	$0	$6,216,142	5.58%
$0	$0	$200,000	$3,440,675	10.140%	$0	$0	$200,000	$6,171,182	5.178%
$400,000	$0	$0	$3,385,060	7.270%	$400,000	$0	$0	$6,115,567	4.969%
$200,000	$200,000	$0	$3,357,828	5.191%	$200,000	$200,000	$0	$6,088,335	4.701%
$0	$400,000	$0	$3,330,595	3.588%	$0	$400,000	$0	$6,061,102	4.272%
$200,000	$0	$200,000	$3,312,868	2.649%	$200,000	$0	$200,000	$6,043,375	4.069%
$0	$200,000	$200,000	$3,285,635	1.816%	$0	$200,000	$200,000	$6,016,142	3.667%
$600,000	$0	$0	$3,257,252	1.244%	$600,000	$0	$0	$5,987,759	3.524%
					$400,000	$200,000	$0	$5,960,527	3.329%
					$0	$0	$400,000	$5,971,182	3.113%
					$200,000	$400,000	$0	$5,933,295	2.946%
					$400,000	$0	$200,000	$5,915,567	2.718%
					$0	$600,000	$0	$5,906,062	2.535%
					$200,000	$200,000	$200,000	$5,888,335	2.446%
					$800,000	$0	$0	$5,859,952	2.161%
					$0	$400,000	$200,000	$5,861,102	2.151%
					$600,000	$200,000	$0	$5,832,719	1.891%
					$200,000	$0	$400,000	$5,843,375	1.758%
					$400,000	$400,000	$0	$5,805,487	1.711%
					$0	$200,000	$400,000	$5,816,142	1.567%
					$600,000	$0	$200,000	$5,787,759	1.501%
					$200,000	$600,000	$0	$5,778,255	1.416%
					$400,000	$200,000	$200,000	$5,760,527	1.308%
					$1,000,000	$0	$0	$5,732,144	1.184%
					$0	$0	$600,000	$5,771,182	1.178%
					$0	$800,000	$0	$5,751,022	1.062%
					$200,000	$400,000	$200,000	$5,733,295	1.017%
Other				2.916%	Other				14.814%
Game Value:			**-36.03**		**Winner:**			**P2**	

Figure 51: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,400,000, and the Weighted by Sample, Floored Utility Vector is used.

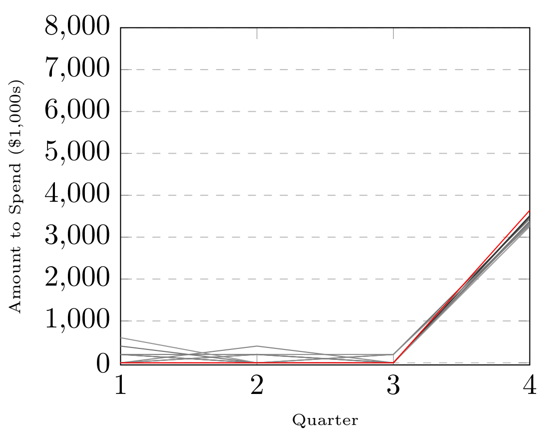


a P1: $800,000 Initial Budget **b** P2: $1,400,000 Initial Budget

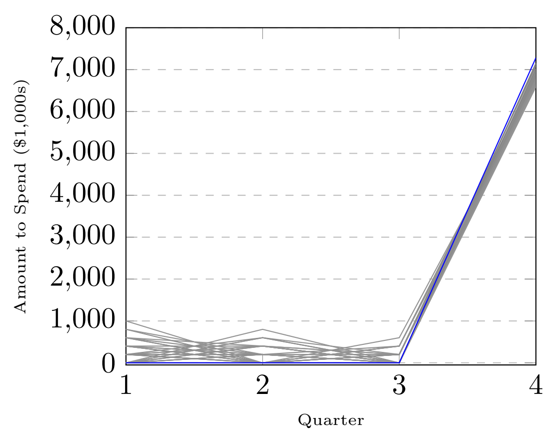
Figure 52: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,600,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$0	$0	$3,640,675	29.405%	$0	$0	$0	$7,281,351	4.386%
$200,000	$0	$0	$3,512,868	20.81%	$200,000	$0	$0	$7,153,544	4.176%
$0	$200,000	$0	$3,485,635	14.639%	$0	$200,000	$0	$7,126,311	4.085%
$0	$0	$200,000	$3,440,675	10.369%	$0	$0	$200,000	$7,081,351	3.793%
$400,000	$0	$0	$3,385,060	7.400%	$400,000	$0	$0	$7,025,736	3.562%
$200,000	$200,000	$0	$3,357,828	5.174%	$200,000	$200,000	$0	$6,998,504	3.493%
$0	$400,000	$0	$3,330,595	3.638%	$0	$400,000	$0	$6,971,271	3.279%
$200,000	$0	$200,000	$3,312,868	2.577%	$200,000	$0	$200,000	$6,953,544	3.274%
$0	$200,000	$200,000	$3,285,635	1.804%	$0	$200,000	$200,000	$6,926,311	3.026%
$600,000	$0	$0	$3,257,252	1.232%	$600,000	$0	$0	$6,897,928	2.999%
					$400,000	$200,000	$0	$6,870,696	2.845%
					$0	$0	$400,000	$6,881,351	2.649%
					$200,000	$400,000	$0	$6,843,464	2.569%
					$400,000	$0	$200,000	$6,825,736	2.503%
					$0	$600,000	$0	$6,816,231	2.406%
					$200,000	$200,000	$200,000	$6,798,504	2.153%
					$800,000	$0	$0	$6,770,121	2.083%
					$0	$400,000	$200,000	$6,771,271	2.036%
					$600,000	$200,000	$0	$6,742,888	2.000%
					$200,000	$0	$400,000	$6,753,544	1.873%
					$400,000	$400,000	$0	$6,715,656	1.774%
					$0	$200,000	$400,000	$6,726,311	1.736%
					$600,000	$0	$200,000	$6,697,928	1.645%
					$200,000	$600,000	$0	$6,688,424	1.557%
					$0	$0	$600,000	$6,681,351	1.487%
					$400,000	$200,000	$200,000	$6,670,696	1.483%
					$1,000,000	$0	$0	$6,642,313	1.379%
					$0	$800,000	$0	$6,661,191	1.297%
					$200,000	$400,000	$200,000	$6,643,464	1.259%
					$800,000	$200,000	$0	$6,615,081	1.227%
					$400,000	$0	$400,000	$6,625,736	1.198%
					$600,000	$400,000	$0	$6,587,848	1.062%
					$0	$600,000	$200,000	$6,616,231	1.06%
Other				2.952%	Other				22.646%
Game Value:			**-48.36**		**Winner:**			**P2**	

Figure 53: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,600,000, and the Weighted by Sample, Floored Utility Vector is used.



a P1: $800,000 Initial Budget



b P2: $1,600,000 Initial Budget

Figure 54: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

6.3 Unweighted, Unfloored Utility Vector

Until this point, only floored utility vectors have been examined. Floored vectors are likely the most realistic, since it would go against much election literature and common practice if spending money did not help with polls [11] [17] [4]. However, it is possible that spending money actually hurts a campaign. This could be caused by voters getting tired of hearing a candidate's name, or by scrutiny of policy issues that accompanies name recognition. In all likelihood, the negative numbers are almost definitely the direct result of a poorly-constructed utility function. As Section 7: **Critiques** will outline, a more accurate utility vector is definitely possible, but was not explored in this report.

The Unweighted, Unfloored Utility Vector is the first of these vectors. It takes the form

$$[6.30872096, 2.549945091, 0.401012477, -1.653903518] \cdot 2 \times 10^{-5}. \tag{12}$$

Note that this utility vector rewards players for spending at the beginning of the campaign, but these rewards get progressively worse until finally a player is punished for spending in the Q4. What makes this utility vector epecially interesting, though, is the way it interacts with the fundraising return function and the rules of the game. Recall that the fundraising return function takes the form

$$f_i(f_{i-1}, s_i) = 0.1124s_i + 1.1057f_{i-1},$$

where f_k denotes the amount a candidate fundraised in quarter k and s_k denotes the amount a candidate spent in quarter k. This yields higher and higher fundraising returns each quarter, and gives an extra boost for previous spending. Now, recall also that a player *must* spend their entire remaining budget in Q4. This rule makes more sense when working with floored vectors; however, to lift it would drastically increase the run time of the code, since it would require generating strategies for every spending option for Q4, rather than tacking on a single calculated number to each strategy (see Section 4.3: **Generating Strategies**).

While this rule is unrealistic, it generates some exciting and unexpected optimal strategy sets. Regardless of budget, every game seems to reward an s-curve-like top strategy, where the difference in spending between Q1 and Q2 is notably less than that between Q2 and Q3. This shape is likely generated because players are incentivized to spend their entire remaining budget in Q3, since spending in Q4 is punished. While this does give players a higher fundraising return in Q4, the amount they gain in fundraising from spending this amount in Q3 is less than the amount they would retain by not spending it.

Another interesting aspect of these graphs is the tendency not to spend as strongly early in the game; specifically, in Q2. However, note that the fundraising return is a function of both the past quarter of spending *and* the past quarter of fundraising. Rather than being written recursively, the fundraising function can be written as

$$f(q_k) = \begin{cases} 0 & k = 1 \\ 0.1124s_1 + 1.1057y_0 & k = 2 \\ 0.1124s_2 + 1.1057(0.1124s_1 + 1.1057y_0) & k = 3 \\ 0.1124s_3 + 1.1057(0.1124s_2 + 1.1057(0.1124s_1 + 1.1057y_0)) & k = 4 \end{cases}. \tag{13}$$

Simplified,

$$f(q_k) = \begin{cases} 0 & k = 1 \\ 0.1124s_1 + 1.1057y_0 & k = 2 \\ 0.1124s_2 + 0.1243s_1 + 1.2223y_0 & k = 3 \\ 0.1124s_3 + 1.1057s_2 + 0.1374s_1 + 1.3518y_0 & k = 4 \end{cases}, \tag{14}$$

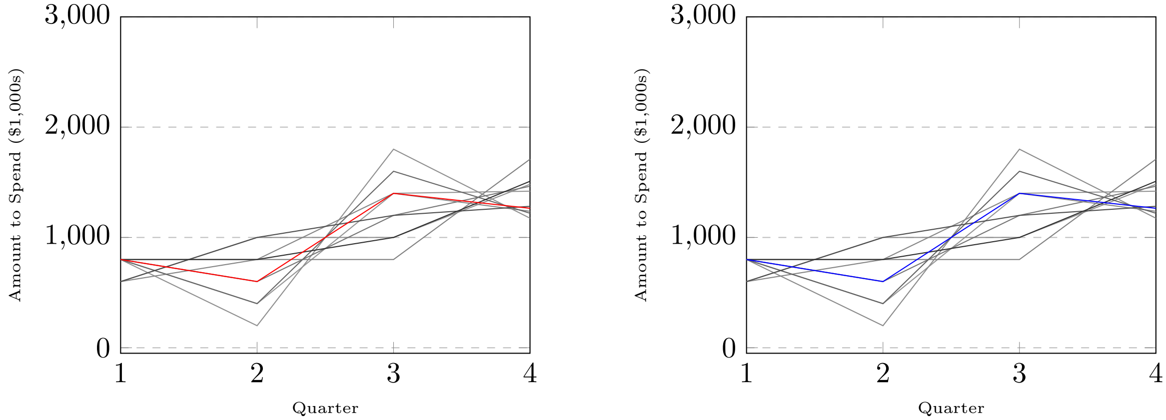
P1 \$800,000 vs. P2 \$800,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
\$800,000	\$600,000	\$1,400,000	\$1,264,325	29.263%	\$800,000	\$600,000	\$1,400,000	\$1,264,325	29.368%
\$800,000	\$800,000	\$1,000,000	\$1,509,285	20.772%	\$800,000	\$800,000	\$1,000,000	\$1,509,285	20.676%
\$600,000	\$1,000,000	\$1,200,000	\$1,282,052	14.836%	\$600,000	\$1,000,000	\$1,200,000	\$1,282,052	14.793%
\$800,000	\$400,000	\$1,600,000	\$1,219,365	10.407%	\$800,000	\$400,000	\$1,600,000	\$1,219,365	10.531%
\$800,000	\$600,000	\$1,200,000	\$1,464,325	7.458%	\$800,000	\$600,000	\$1,200,000	\$1,464,325	7.267%
\$800,000	\$800,000	\$800,000	\$1,709,285	5.206%	\$800,000	\$800,000	\$800,000	\$1,709,285	5.199%
\$600,000	\$800,000	\$1,400,000	\$1,237,092	3.523%	\$600,000	\$800,000	\$1,400,000	\$1,237,092	3.633%
\$800,000	\$200,000	\$1,800,000	\$1,174,405	2.496%	\$800,000	\$200,000	\$1,800,000	\$1,174,405	2.608%
\$600,000	\$1,000,000	\$1,000,000	\$1,482,052	1.832%	\$600,000	\$1,000,000	\$1,000,000	\$1,482,052	1.736%
\$800,000	\$400,000	\$1,400,000	\$1,419,365	1.232%	\$800,000	\$400,000	\$1,400,000	\$1,419,365	1.270%
Other				2.975%	Other				2.919%
Game Value:			**0.00**		**Winner:**			**P1**	

Figure 55: Mixed strategy output, when P1 is given an initial budget of \$800,000, P2 is given an initial budget of \$800,000, and the Unweighted, Unfloored Utility Vector is used.

where $f(q_k)$ denotes the amount a player fundraises immediately before quarter k, s_k denotes the amount a player spends in quarter k, and y_0 indicates a player's initial budget. This results in a lower spend amount in Q2. However, in Q1, this incentive not to increase later fundraising amounts is outweighed heavily by the high utility of spending that quarter.

In every budget pairing, another highly used strategy is spending a relatively even amount each quarter. The benefit of this strategy is keeping the fundraising return low relative to other strategies, since spending a lot in a single quarter creates an abnormally high coefficient for that s value in (14). In lesser-used strategies, however, spending amounts become more and more variable in Q2 and Q3. The purpose of this is likely to create large differentials in the voter utility function to maximize voters won in higher-spending months, while minimizing the amount fundraises — and therefore must spend — for Q4. These graphs are especially interesting because lesser-used strategies for each quarter are not bounded by the top strategy, but are rather both higher and lower than it, depending on the strategy.

This vector only yields wins for P2. This is more significant than previous vectors, however, because this result is no longer trivial. Since the vector punishes players for spending in the last quarter, it's feasible that P1 could win different simulations than in previous games. For this reason, players are highly discouraged from saving their entire budget for the last quarter, which in previous games has been a very popular dominant strategy.



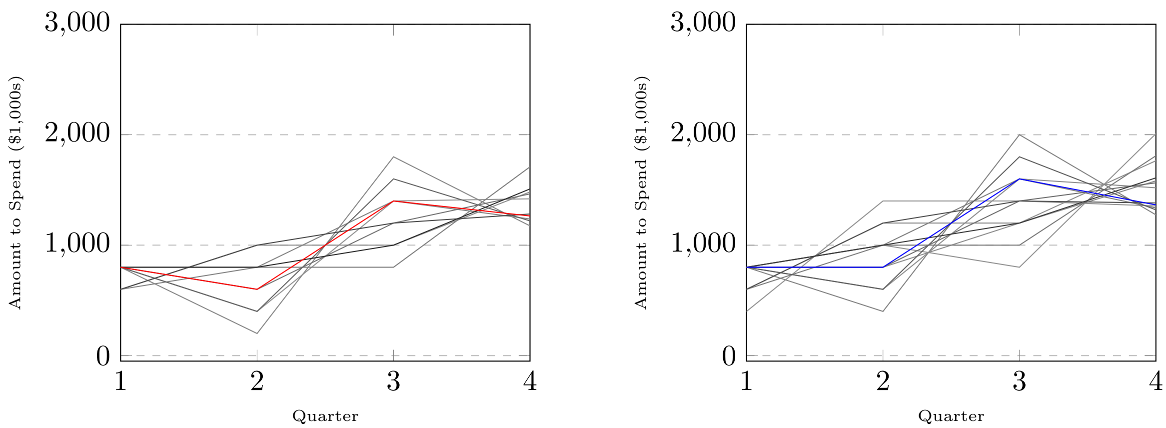
a P1: $800,000 Initial Budget

b P2: $800,000 Initial Budget

Figure 56: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $900,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$600,000	$1,400,000	$1,264,325	29.587%	$800,000	$800,000	$1,600,000	$1,364,369	22.952%
$800,000	$800,000	$1,000,000	$1,509,285	20.626%	$800,000	$1,000,000	$1,200,000	$1,609,329	17.574%
$600,000	$1,000,000	$1,200,000	$1,282,052	14.726%	$600,000	$1,200,000	$1,400,000	$1,382,097	13.698%
$800,000	$400,000	$1,600,000	$1,219,365	10.363%	$800,000	$600,000	$1,800,000	$1,319,409	10.681%
$800,000	$600,000	$1,200,000	$1,464,325	7.345%	$800,000	$800,000	$1,400,000	$1,564,369	8.102%
$800,000	$800,000	$800,000	$1,709,285	5.182%	$800,000	$1,000,000	$1,000,000	$1,809,329	6.202%
$600,000	$800,000	$1,400,000	$1,237,092	3.598%	$600,000	$1,000,000	$1,600,000	$1,337,137	4.783%
$800,000	$200,000	$1,800,000	$1,174,405	2.551%	$800,000	$400,000	$2,000,000	$1,274,449	3.631%
$600,000	$1,000,000	$1,000,000	$1,482,052	1.851%	$600,000	$1,200,000	$1,200,000	$1,582,097	2.888%
$800,000	$400,000	$1,400,000	$1,419,365	1.248%	$800,000	$600,000	$1,600,000	$1,519,409	2.204%
					$800,000	$800,000	$1,200,000	$1,764,369	1.694%
					$800,000	$1,000,000	$800,000	$2,009,329	1.289%
					$400,000	$1,400,000	$1,400,000	$1,354,865	1.069%
Other				2.923%	Other				3.233%
Game Value:			**-1.75**		**Winner:**			**P2**	

Figure 57: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $900,000, and the Unweighted, Unfloored Utility Vector is used.



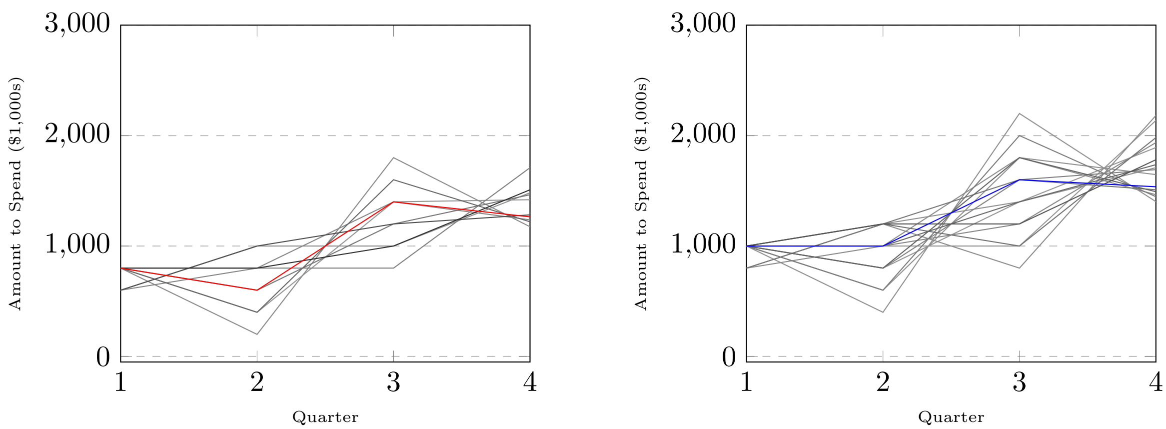
a P1: $800,000 Initial Budget

b P2: $900,000 Initial Budget

Figure 58: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,000,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$600,000	$1,400,000	$1,264,325	29.373%	$1,000,000	$1,000,000	$1,600,000	$1,536,606	16.190%
$800,000	$800,000	$1,000,000	$1,509,285	20.941%	$1,000,000	$1,200,000	$1,200,000	$1,781,566	13.529%
$600,000	$1,000,000	$1,200,000	$1,282,052	14.749%	$1,000,000	$800,000	$1,800,000	$1,491,646	11.424%
$800,000	$400,000	$1,600,000	$1,219,365	10.276%	$1,000,000	$1,000,000	$1,400,000	$1,736,606	9.513%
$800,000	$600,000	$1,200,000	$1,464,325	7.398%	$1,000,000	$1,200,000	$1,000,000	$1,981,566	7.899%
$800,000	$800,000	$800,000	$1,709,285	5.096%	$800,000	$1,200,000	$1,600,000	$1,509,374	6.819%
$600,000	$800,000	$1,400,000	$1,237,092	3.663%	$1,000,000	$600,000	$2,000,000	$1,446,686	5.540%
$800,000	$200,000	$1,800,000	$1,174,405	2.583%	$1,000,000	$800,000	$1,600,000	$1,691,646	4.719%
$600,000	$1,000,000	$1,000,000	$1,482,052	1.837%	$1,000,000	$1,000,000	$1,200,000	$1,936,606	3.950%
$800,000	$400,000	$1,400,000	$1,419,365	1.240%	$1,000,000	$1,200,000	$800,000	$2,181,566	3.265%
					$800,000	$1,000,000	$1,800,000	$1,464,414	2.764%
					$1,000,000	$400,000	$2,200,000	$1,401,726	2.408%
					$800,000	$1,200,000	$1,400,000	$1,709,374	1.985%
					$1,000,000	$600,000	$1,800,000	$1,646,686	1.709%
					$1,000,000	$800,000	$1,400,000	$1,891,646	1.390%
					$1,000,000	$1,000,000	$1,000,000	$2,136,606	1.134%
Game Value:			**-8.39**		**Winner:**			**P2**	
Other				2.844%	Other				5.762%

Figure 59: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,000,000, and the Unweighted, Unfloored Utility Vector is used.



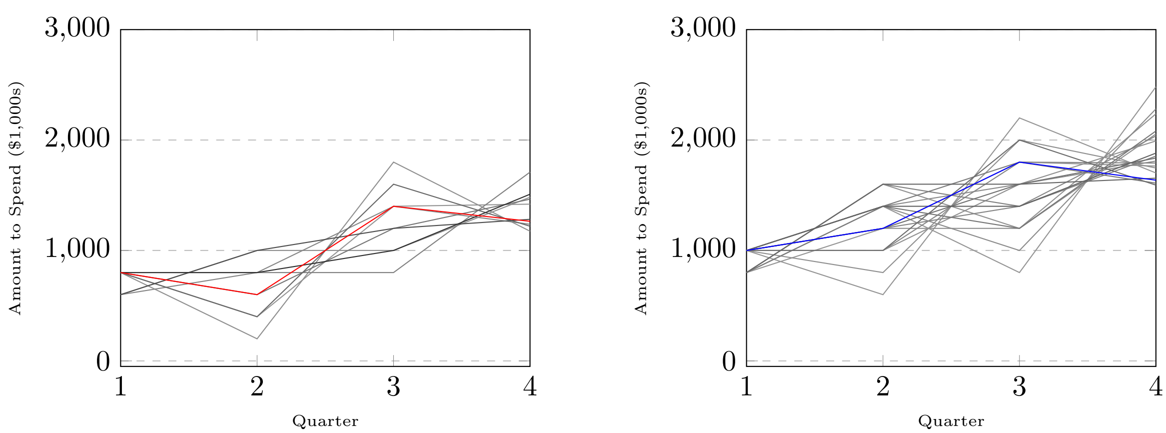
a P1: $800,000 Initial Budget

b P2: $1,000,000 Initial Budget

Figure 60: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,100,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$600,000	$1,400,000	$1,264,325	29.356%	$1,000,000	$1,200,000	$1,800,000	$1,636,650	12.95%
$800,000	$800,000	$1,000,000	$1,509,285	20.991%	$1,000,000	$1,400,000	$1,400,000	$1,881,610	11.353%
$600,000	$1,000,000	$1,200,000	$1,282,052	14.699%	$800,000	$1,600,000	$1,600,000	$1,654,378	9.862%
$800,000	$400,000	$1,600,000	$1,219,365	10.481%	$1,000,000	$1,000,000	$2,000,000	$1,591,690	8.709%
$800,000	$600,000	$1,200,000	$1,464,325	7.205%	$1,000,000	$1,200,000	$1,600,000	$1,836,650	7.605%
$800,000	$800,000	$800,000	$1,709,285	5.168%	$1,000,000	$1,400,000	$1,200,000	$2,081,610	6.572%
$600,000	$800,000	$1,400,000	$1,237,092	3.744%	$800,000	$1,400,000	$1,800,000	$1,609,418	5.659%
$800,000	$200,000	$1,800,000	$1,174,405	2.478%	$800,000	$1,600,000	$1,400,000	$1,854,378	4.980%
$600,000	$1,000,000	$1,000,000	$1,482,052	1.805%	$1,000,000	$1,000,000	$1,800,000	$1,791,690	4.304%
$800,000	$400,000	$1,400,000	$1,419,365	1.233%	$1,000,000	$1,200,000	$1,400,000	$2,036,650	3.752%
					$1,000,000	$1,400,000	$1,000,000	$2,281,610	3.15%
					$800,000	$1,400,000	$1,600,000	$1,809,418	2.909%
					$1,000,000	$800,000	$2,000,000	$1,746,730	2.325%
					$800,000	$1,600,000	$1,200,000	$2,054,378	2.123%
					$1,000,000	$1,000,000	$1,600,000	$1,991,690	1.853%
					$1,000,000	$1,200,000	$1,200,000	$2,236,650	1.582%
					$1,000,000	$1,400,000	$800,000	$2,481,610	1.376%
					$800,000	$1,200,000	$1,800,000	$1,764,458	1.223%
					$1,000,000	$600,000	$2,200,000	$1,701,770	1.049%
Other				2.84%	Other				6.664%
Game Value:			**-10.25**		**Winner:**			**P2**	

Figure 61: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,100,000, and the Unweighted, Unfloored Utility Vector is used.



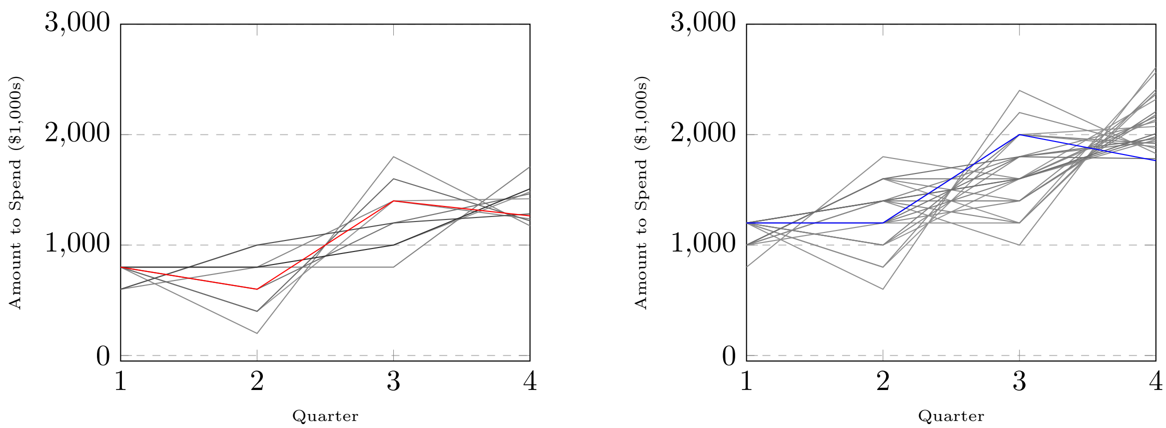
a P1: $800,000 Initial Budget

b P2: $1,100,000 Initial Budget

Figure 62: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,200,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$600,000	$1,400,000	$1,264,325	29.447%	$1,200,000	$1,200,000	$2,000,000	$1,763,927	9.798%
$800,000	$800,000	$1,000,000	$1,509,285	20.903%	$1,200,000	$1,400,000	$1,600,000	$2,008,887	8.736%
$600,000	$1,000,000	$1,200,000	$1,282,052	14.619%	$1,000,000	$1,600,000	$1,800,000	$1,781,655	7.889%
$800,000	$400,000	$1,600,000	$1,219,365	10.312%	$1,200,000	$1,200,000	$1,800,000	$1,963,927	7.106%
$800,000	$600,000	$1,200,000	$1,464,325	7.583%	$1,200,000	$1,400,000	$1,400,000	$2,208,887	6.442%
$800,000	$800,000	$800,000	$1,709,285	5.209%	$1,000,000	$1,600,000	$1,600,000	$1,981,655	5.862%
$600,000	$800,000	$1,400,000	$1,237,092	3.605%	$1,200,000	$1,000,000	$2,000,000	$1,918,967	5.218%
$800,000	$200,000	$1,800,000	$1,174,405	2.571%	$1,200,000	$1,200,000	$1,600,000	$2,163,927	4.781%
$600,000	$1,000,000	$1,000,000	$1,482,052	1.727%	$1,200,000	$1,400,000	$1,200,000	$2,408,887	4.324%
$800,000	$400,000	$1,400,000	$1,419,365	1.242%	$1,000,000	$1,400,000	$1,800,000	$1,936,695	3.917%
					$1,200,000	$800,000	$2,200,000	$1,874,007	3.485%
					$1,000,000	$1,600,000	$1,400,000	$2,181,655	3.182%
					$1,200,000	$1,000,000	$1,800,000	$2,118,967	2.836%
					$1,200,000	$1,200,000	$1,400,000	$2,363,927	2.585%
					$1,200,000	$1,400,000	$1,000,000	$2,608,887	2.278%
					$800,000	$1,800,000	$1,600,000	$1,954,423	2.201%
					$1,000,000	$1,200,000	$2,000,000	$1,891,735	1.918%
					$1,200,000	$600,000	$2,400,000	$1,829,047	1.727%
					$1,000,000	$1,400,000	$1,600,000	$2,136,695	1.576%
					$1,200,000	$800,000	$2,000,000	$2,074,007	1.424%
					$1,000,000	$1,600,000	$1,200,000	$2,381,655	1.313%
					$1,200,000	$1,000,000	$1,600,000	$2,318,967	1.128%
					$1,200,000	$1,200,000	$1,200,000	$2,563,927	1.020%
Other				2.782%	Other				9.254%
Game Value:			**-14.86**		**Winner:**			**P2**	

Figure 63: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,200,000, and the Unweighted, Unfloored Utility Vector is used.

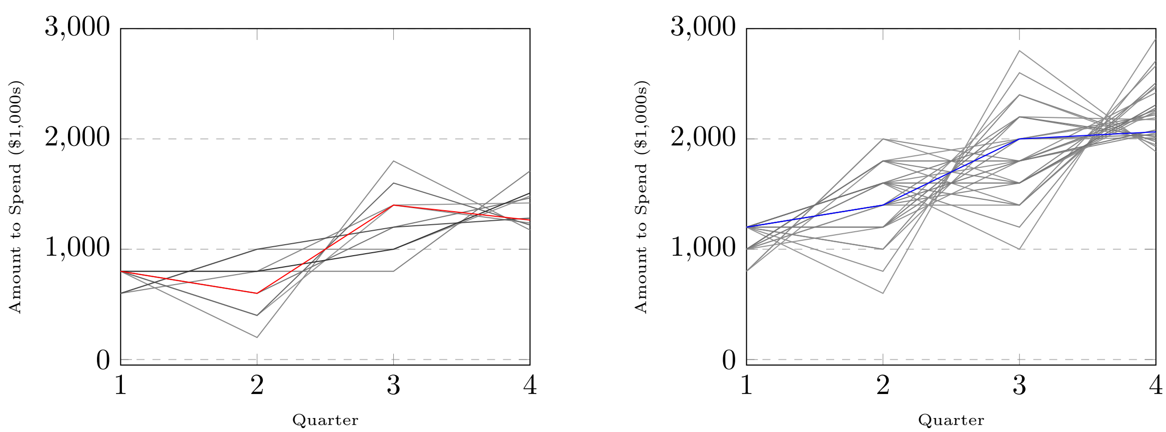


a P1: $800,000 Initial Budget **b** P2: $1,200,000 Initial Budget

Figure 64: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,300,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$600,000	$1,400,000	$1,264,325	29.358%	$1,200,000	$1,400,000	$2,000,000	$2,063,972	8.018%
$800,000	$800,000	$1,000,000	$1,509,285	20.738%	$1,200,000	$1,600,000	$1,600,000	$2,308,932	7.5%
$600,000	$1,000,000	$1,200,000	$1,282,052	14.760%	$1,000,000	$1,800,000	$1,800,000	$2,081,699	6.740%
$800,000	$400,000	$1,600,000	$1,219,365	10.455%	$1,200,000	$1,200,000	$2,200,000	$2,019,012	6.415%
$800,000	$600,000	$1,200,000	$1,464,325	7.331%	$1,200,000	$1,400,000	$1,800,000	$2,263,972	5.886%
$800,000	$800,000	$800,000	$1,709,285	5.162%	$1,200,000	$1,600,000	$1,400,000	$2,508,932	5.434%
$600,000	$800,000	$1,400,000	$1,237,092	3.677%	$1,000,000	$1,600,000	$2,000,000	$2,036,739	4.853%
$800,000	$200,000	$1,800,000	$1,174,405	2.512%	$1,200,000	$1,000,000	$2,400,000	$1,974,052	4.399%
$600,000	$1,000,000	$1,000,000	$1,482,052	1.815%	$1,000,000	$1,800,000	$1,600,000	$2,281,699	3.990%
$800,000	$400,000	$1,400,000	$1,419,365	1.258%	$1,200,000	$1,200,000	$2,000,000	$2,219,012	3.875%
					$1,200,000	$1,400,000	$1,600,000	$2,463,972	3.545%
					$1,200,000	$1,600,000	$1,200,000	$2,708,932	3.200%
					$800,000	$2,000,000	$1,800,000	$2,054,467	2.992%
					$1,000,000	$1,400,000	$2,200,000	$1,991,779	2.764%
					$1,200,000	$800,000	$2,600,000	$1,929,092	2.488%
					$1,000,000	$1,600,000	$1,800,000	$2,236,739	2.312%
					$1,200,000	$1,000,000	$2,200,000	$2,174,052	2.076%
					$1,000,000	$1,800,000	$1,400,000	$2,481,699	1.980%
					$1,200,000	$1,200,000	$1,800,000	$2,419,012	1.768%
					$1,200,000	$1,400,000	$1,400,000	$2,663,972	1.576%
					$1,200,000	$1,600,000	$1,000,000	$2,908,932	1.471%
					$800,000	$1,800,000	$2,000,000	$2,009,507	1.425%
					$1,000,000	$1,200,000	$2,400,000	$1,946,819	1.232%
					$1,200,000	$600,000	$2,800,000	$1,884,132	1.191%
					$800,000	$2,000,000	$1,600,000	$2,254,467	1.076%
					$1,000,000	$1,400,000	$2,000,000	$2,191,779	1.041%
Other				2.934%	Other				10.753%
Game Value:			**-15.26**		**Winner:**			**P2**	

Figure 65: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,300,000, and the Unweighted, Unfloored Utility Vector is used.

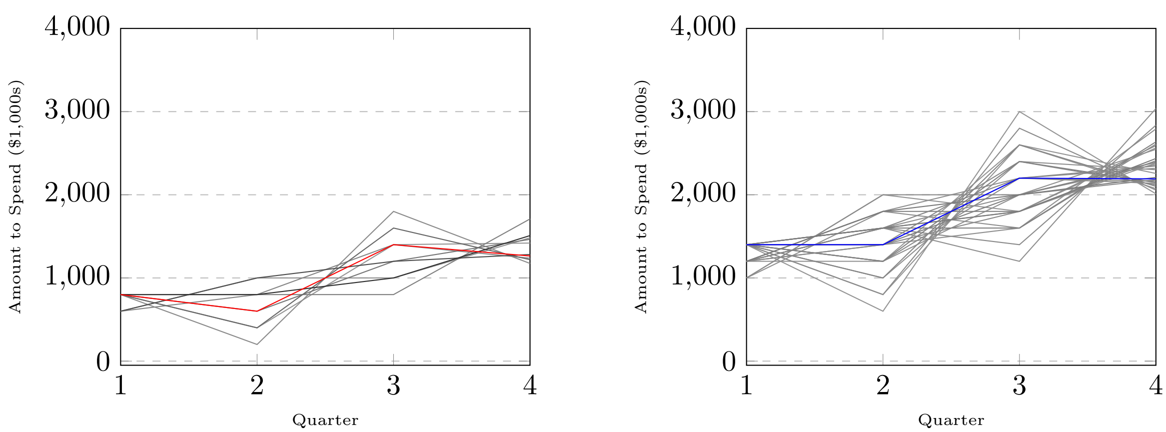


a P1: $800,000 Initial Budget **b** P2: $1,300,000 Initial Budget

Figure 66: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,400,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$600,000	$1,400,000	$1,264,325	29.349%	$1,400,000	$1,400,000	$2,200,000	$2,191,248	6.338%
$800,000	$800,000	$1,000,000	$1,509,285	20.833%	$1,400,000	$1,600,000	$1,800,000	$2,436,208	5.893%
$600,000	$1,000,000	$1,200,000	$1,282,052	14.612%	$1,200,000	$1,800,000	$2,000,000	$2,208,976	5.583%
$800,000	$400,000	$1,600,000	$1,219,365	10.405%	$1,400,000	$1,200,000	$2,400,000	$2,146,288	5.320%
$800,000	$600,000	$1,200,000	$1,464,325	7.466%	$1,400,000	$1,400,000	$2,000,000	$2,391,248	4.977%
$800,000	$800,000	$800,000	$1,709,285	5.085%	$1,400,000	$1,600,000	$1,600,000	$2,636,208	4.505%
$600,000	$800,000	$1,400,000	$1,237,092	3.674%	$1,200,000	$1,600,000	$2,200,000	$2,164,016	4.314%
$800,000	$200,000	$1,800,000	$1,174,405	2.610%	$1,400,000	$1,000,000	$2,600,000	$2,101,328	4.023%
$600,000	$1,000,000	$1,000,000	$1,482,052	1.865%	$1,200,000	$1,800,000	$1,800,000	$2,408,976	3.639%
$800,000	$400,000	$1,400,000	$1,419,365	1.253%	$1,400,000	$1,200,000	$2,200,000	$2,346,288	3.463%
					$1,400,000	$1,400,000	$1,800,000	$2,591,248	3.252%
					$1,400,000	$1,600,000	$1,400,000	$2,836,208	3.101%
					$1,000,000	$2,000,000	$2,000,000	$2,181,744	2.967%
					$1,200,000	$1,400,000	$2,400,000	$2,119,056	2.838%
					$1,400,000	$800,000	$2,800,000	$2,056,368	2.539%
					$1,200,000	$1,600,000	$2,000,000	$2,364,016	2.335%
					$1,400,000	$1,000,000	$2,400,000	$2,301,328	2.275%
					$1,200,000	$1,800,000	$1,600,000	$2,608,976	2.137%
					$1,400,000	$1,200,000	$2,000,000	$2,546,288	1.981%
					$1,400,000	$1,400,000	$1,600,000	$2,791,248	1.902%
					$1,400,000	$1,600,000	$1,200,000	$3,036,208	1.733%
					$1,000,000	$1,800,000	$2,200,000	$2,136,784	1.591%
					$1,200,000	$1,200,000	$2,600,000	$2,074,096	1.489%
					$1,400,000	$600,000	$3,000,000	$2,011,408	1.314%
					$1,000,000	$2,000,000	$1,800,000	$2,381,744	1.282%
					$1,200,000	$1,400,000	$2,200,000	$2,319,056	1.252%
					$1,200,000	$1,600,000	$1,800,000	$2,564,016	1.095%
					$1,400,000	$800,000	$2,600,000	$2,256,368	1.093%
Other				2.848%	Other				15.769%
Game Value:			**-20.55**		**Winner:**			**P2**	

Figure 67: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,400,000, and the Unweighted, Unfloored Utility Vector is used.



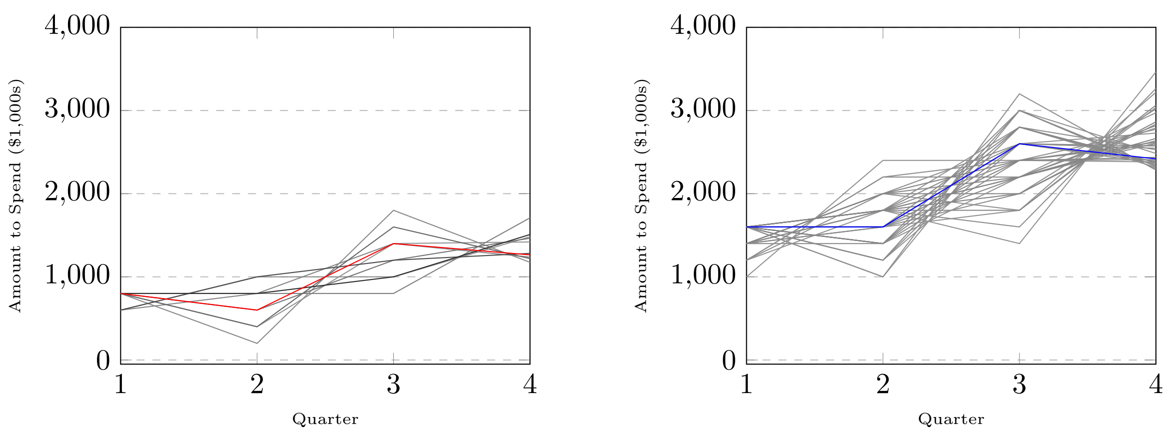
a P1: $800,000 Initial Budget

b P2: $1,400,000 Initial Budget

Figure 68: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,600,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$600,000	$1,400,000	$1,264,325	29.600%	$1,600,000	$1,600,000	$2,600,000	$2,418,570	4.295%
$800,000	$800,000	$1,000,000	$1,509,285	20.765%	$1,600,000	$1,800,000	$2,200,000	$2,663,530	4.092%
$600,000	$1,000,000	$1,200,000	$1,282,052	14.606%	$1,400,000	$2,000,000	$2,400,000	$2,436,297	3.998%
$800,000	$400,000	$1,600,000	$1,219,365	10.306%	$1,600,000	$1,400,000	$2,800,000	$2,373,610	3.756%
$800,000	$600,000	$1,200,000	$1,464,325	7.335%	$1,600,000	$1,600,000	$2,400,000	$2,618,570	3.657%
$800,000	$800,000	$800,000	$1,709,285	5.203%	$1,600,000	$1,800,000	$2,000,000	$2,863,530	3.520%
$600,000	$800,000	$1,400,000	$1,237,092	3.729%	$1,400,000	$1,800,000	$2,600,000	$2,391,337	3.303%
$800,000	$200,000	$1,800,000	$1,174,405	2.571%	$1,600,000	$1,200,000	$3,000,000	$2,328,650	3.233%
$600,000	$1,000,000	$1,000,000	$1,482,052	1.737%	$1,400,000	$2,000,000	$2,200,000	$2,636,297	3.094%
$800,000	$400,000	$1,400,000	$1,419,365	1.277%	$1,600,000	$1,400,000	$2,600,000	$2,573,610	3.031%
					$1,600,000	$1,600,000	$2,200,000	$2,818,570	2.847%
					$1,600,000	$1,800,000	$1,800,000	$3,063,530	2.685%
					$1,400,000	$1,600,000	$2,800,000	$2,346,377	2.595%
					$1,200,000	$2,200,000	$2,400,000	$2,409,065	2.477%
					$1,600,000	$1,000,000	$3,200,000	$2,283,690	2.298%
					$1,400,000	$1,800,000	$2,400,000	$2,591,337	2.158%
					$1,600,000	$1,200,000	$2,800,000	$2,528,650	2.108%
					$1,600,000	$1,400,000	$2,400,000	$2,773,610	2.000%
					$1,400,000	$2,000,000	$2,000,000	$2,836,297	1.999%
					$1,600,000	$1,600,000	$2,000,000	$3,018,570	1.936%
					$1,600,000	$1,800,000	$1,600,000	$3,263,530	1.881%
					$1,200,000	$2,000,000	$2,600,000	$2,364,105	1.646%
					$1,400,000	$1,400,000	$3,000,000	$2,301,417	1.631%
					$1,400,000	$1,600,000	$2,600,000	$2,546,377	1.574%
					$1,200,000	$2,200,000	$2,200,000	$2,609,065	1.558%
					$1,400,000	$1,800,000	$2,200,000	$2,791,337	1.396%
					$1,600,000	$1,000,000	$3,000,000	$2,483,690	1.330%
					$1,400,000	$2,000,000	$1,800,000	$3,036,297	1.327%
					$1,600,000	$1,200,000	$2,600,000	$2,728,650	1.279%
					$1,600,000	$1,400,000	$2,200,000	$2,973,610	1.249%
					$1,600,000	$1,600,000	$1,800,000	$3,218,570	1.132%
					$1,600,000	$1,800,000	$1,400,000	$3,463,530	1.107%
					$1,200,000	$1,800,000	$2,800,000	$2,319,145	1.068%
					$1,000,000	$2,400,000	$2,400,000	$2,381,833	1.058%
Other				2.871%	Other				21.682%
Game Value:			**-27.57**		**Winner:**			**P2**	

Figure 69: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,600,000, and the Unweighted, Unfloored Utility Vector is used.



a P1: $800,000 Initial Budget **b** P2: $1,600,000 Initial Budget

Figure 70: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $800,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$0	$2,000,000	$1,129,445	29.356%	$800,000	$0	$2,000,000	$1,129,445	29.402%
$800,000	$0	$1,800,000	$1,329,445	20.984%	$800,000	$0	$1,800,000	$1,329,445	20.816%
$800,000	$0	$1,600,000	$1,529,445	14.859%	$800,000	$0	$1,600,000	$1,529,445	14.585%
$800,000	$0	$1,400,000	$1,729,445	10.314%	$800,000	$0	$1,400,000	$1,729,445	10.524%
$800,000	$0	$1,200,000	$1,929,445	7.285%	$800,000	$0	$1,200,000	$1,929,445	7.348%
$800,000	$0	$1,000,000	$2,129,445	5.014%	$800,000	$0	$1,000,000	$2,129,445	5.133%
$800,000	$200,000	$1,800,000	$1,174,405	3.715%	$800,000	$200,000	$1,800,000	$1,174,405	3.606%
$800,000	$0	$800,000	$2,329,445	2.601%	$800,000	$0	$800,000	$2,329,445	2.546%
$800,000	$200,000	$1,600,000	$1,374,405	1.761%	$800,000	$200,000	$1,600,000	$1,374,405	1.862%
$800,000	$0	$600,000	$2,529,445	1.229%	$800,000	$0	$600,000	$2,529,445	1.223%
Other				2.882 %	Other				2.955%
Game Value:			**0.00**		**Winner:**			**P1**	

Figure 71: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $800,000, and the Weighted by State, Unfloored Utility Vector is used.

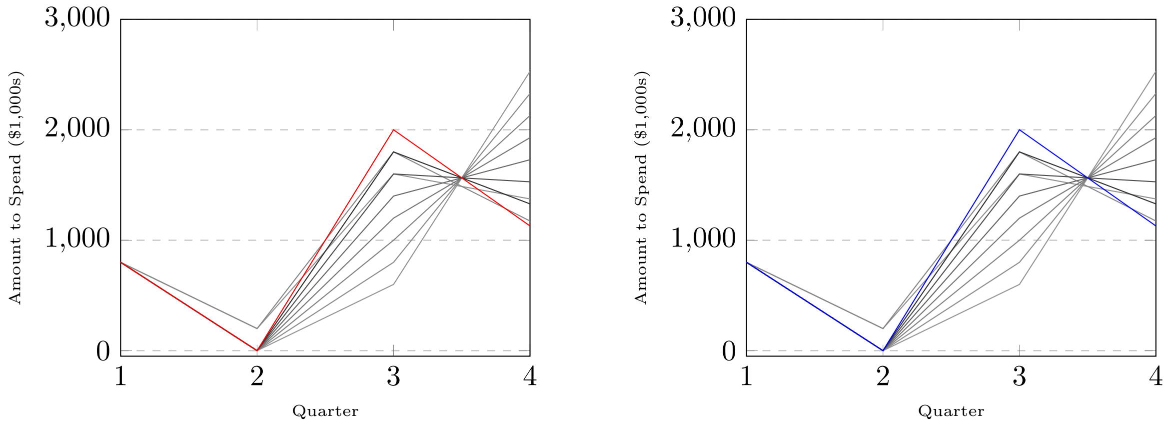
6.4 Weighted by State, Unfloored Utility Vector

The Weighted by State, Unfloored Utility Vector is given by

$$[0.353833059, -0.460278059, 0.204454969, -0.480740837] \cdot 2 \times 10^{-5}. \tag{15}$$

Unlike the Unweighted, Unfloored Utility Vector, this vector's values are all relatively close to each other. Additionally, players get punished in Q2 and Q4, and they are punished more heavily in these quarters than they are rewarded in Q1 and Q3. While this creates a very unrealistic model, it also creates some great graphs. The optimal strategy for any player, regardless of budget, is generally to minimize spending in Q2 and Q4 and maximize spending in Q1 and Q3. In fact, every strategy used ¿1% of the time has a maximum spend in Q1. In most strategies, players spend $0 in Q2, and the Q4 column is ordered roughly descending. These trends are evident from the gradients in the overlay graphs, especially in Q3 and Q4 (**Figures 56, 58, 60, 62, 62, 64, 66, 68, 70**). The optimal strategies create alternately upper and lower bounds for spending strategies — an upper bound for Q1 (and a lower bound, in fact), lower bound for Q2, upper bound for Q3, and a near-lower bound for Q4.

The most exciting part about these simulations, however, is the winner - P1 actually begins winning games for the first time! This is also the first vector in which the negative values outweigh the positive ones. This punishes players more heavily for spending than in any previous game. As a result, players actually have an advantage by starting with a low budget, since it means they don't have to spend as much. The values are very close to zero, but they are definitively positive; P1 definitely has an advantage, for once.

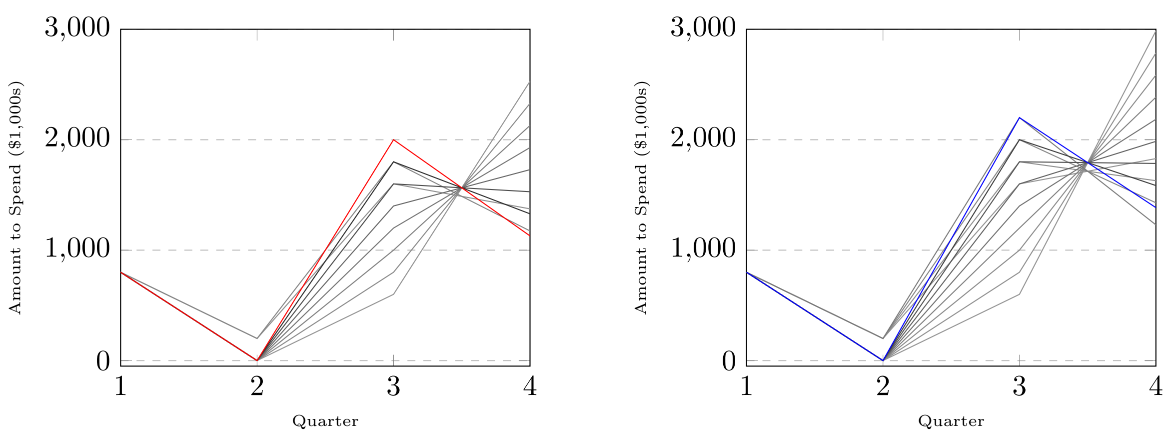


a P1: $800,000 Initial Budget **b** P2: $800,000 Initial Budget

Figure 72: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $900,000									
Player 1 (P1)					Player 2 (P2)				
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$0	$2,000,000	$1,129,445	29.277%	$800,000	$0	$2,200,000	$1,384,529	22.572%
$800,000	$0	$1,800,000	$1,329,445	20.868%	$800,000	$0	$2,000,000	$1,584,529	17.787%
$800,000	$0	$1,600,000	$1,529,445	14.872%	$800,000	$0	$1,800,000	$1,784,529	13.811%
$800,000	$0	$1,400,000	$1,729,445	10.424%	$800,000	$0	$1,600,000	$1,984,529	10.605%
$800,000	$0	$1,200,000	$1,929,445	7.195%	$800,000	$0	$1,400,000	$2,184,529	8.142%
$800,000	$0	$1,000,000	$2,129,445	5.256%	$800,000	$200,000	$2,200,000	$1,229,489	6.234%
$800,000	$200,000	$1,800,000	$1,174,405	3.596%	$800,000	$0	$1,200,000	$2,384,529	4.954%
$800,000	$0	$800,000	$2,329,445	2.554%	$800,000	$200,000	$2,000,000	$1,429,489	3.682%
$800,000	$200,000	$1,600,000	$1,374,405	1.774%	$800,000	$0	$1,000,000	$2,584,529	2.910%
$800,000	$0	$600,000	$2,529,445	1.258%	$800,000	$200,000	$1,800,000	$1,629,489	2.151%
					$800,000	$0	$800,000	$2,784,529	1.631%
					$800,000	$200,000	$1,600,000	$1,829,489	1.261%
					$800,000	$0	$600,000	$2,984,529	1.045%
Other				2.926%	Other				3.215%
Game Value:			**0.53**		**Winner:**			**P1**	

Figure 73: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $900,000, and the Weighted by State, Unfloored Utility Vector is used.

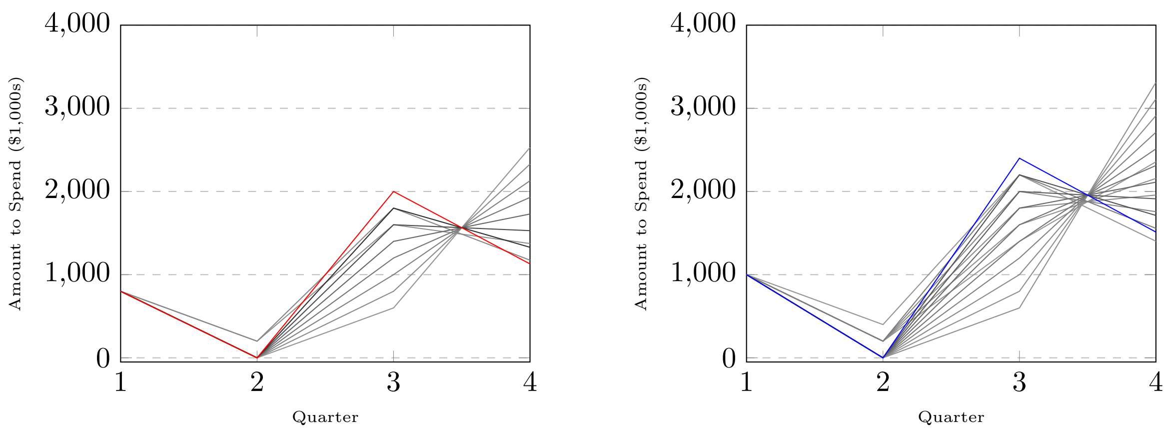


a P1: $800,000 Initial Budget **b** P2: $900,000 Initial Budget

Figure 74: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,000,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$0	$2,000,000	$1,129,445	29.439%	$1,000,000	$0	$2,400,000	$1,511,806	16.177%
$800,000	$0	$1,800,000	$1,329,445	20.974%	$1,000,000	$0	$2,200,000	$1,711,806	13.551%
$800,000	$0	$1,600,000	$1,529,445	14.694%	$1,000,000	$0	$2,000,000	$1,911,806	11.515%
$800,000	$0	$1,400,000	$1,729,445	10.461%	$1,000,000	$0	$1,800,000	$2,111,806	9.556%
$800,000	$0	$1,200,000	$1,929,445	7.257%	$1,000,000	$0	$1,600,000	$2,311,806	7.957%
$800,000	$0	$1,000,000	$2,129,445	5.197%	$1,000,000	$0	$1,400,000	$2,511,806	6.703%
$800,000	$200,000	$1,800,000	$1,174,405	3.567%	$1,000,000	$200,000	$2,200,000	$1,556,766	5.530%
$800,000	$0	$800,000	$2,329,445	2.557%	$1,000,000	$0	$1,200,000	$2,711,806	4.712%
$800,000	$200,000	$1,600,000	$1,374,405	1.742%	$1,000,000	$200,000	$2,000,000	$1,756,766	4.001%
$800,000	$0	$600,000	$2,529,445	1.277%	$1,000,000	$0	$1,000,000	$2,911,806	3.290%
					$1,000,000	$200,000	$1,800,000	$1,956,766	2.792%
					$1,000,000	$0	$800,000	$3,111,806	2.300%
					$1,000,000	$200,000	$1,600,000	$2,156,766	1.903%
					$1,000,000	$0	$600,000	$3,311,806	1.655%
					$1,000,000	$200,000	$1,400,000	$2,356,766	1.475%
					$1,000,000	$400,000	$2,200,000	$1,401,726	1.160%
Other				2.835%	Other				5.723%
Game Value:			**0.82**		**Winner:**			**P1**	

Figure 75: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,000,000, and the Weighted by State, Unfloored Utility Vector is used.



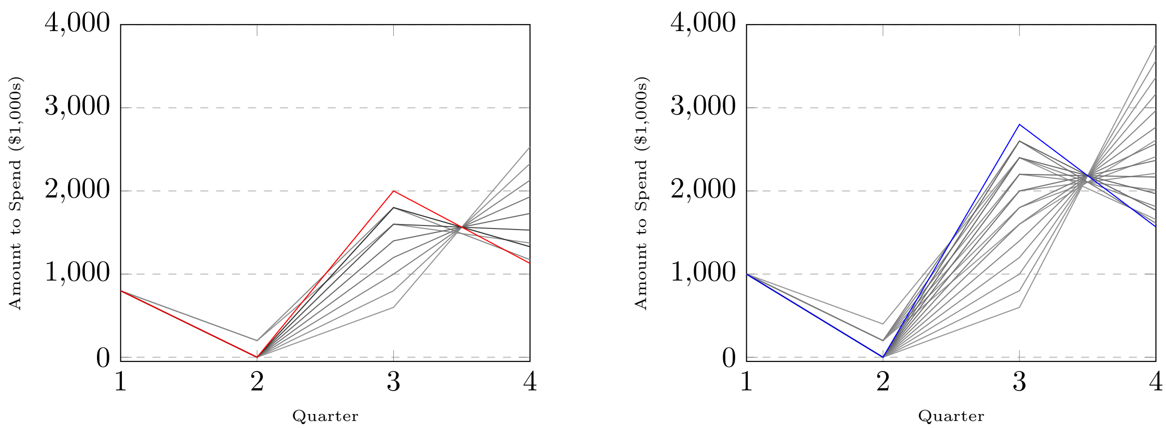
a P1: $800,000 Initial Budget

b P2: $1,000,000 Initial Budget

Figure 76: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,100,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$0	$2,000,000	$1,129,445	29.508%	$1,000,000	$0	$2,800,000	$1,566,890	13.093%
$800,000	$0	$1,800,000	$1,329,445	20.818%	$1,000,000	$0	$2,600,000	$1,766,890	11.256%
$800,000	$0	$1,600,000	$1,529,445	14.522%	$1,000,000	$0	$2,400,000	$1,966,890	9.844%
$800,000	$0	$1,400,000	$1,729,445	10.393%	$1,000,000	$0	$2,200,000	$2,166,890	8.573%
$800,000	$0	$1,200,000	$1,929,445	7.325%	$1,000,000	$0	$2,000,000	$2,366,890	7.500%
$800,000	$0	$1,000,000	$2,129,445	5.295%	$1,000,000	$0	$1,800,000	$2,566,890	6.480%
$800,000	$200,000	$1,800,000	$1,174,405	3.645%	$1,000,000	$200,000	$2,600,000	$1,611,850	5.776%
$800,000	$0	$800,000	$2,329,445	2.460%	$1,000,000	$0	$1,600,000	$2,766,890	5.047%
$800,000	$200,000	$1,600,000	$1,374,405	1.856%	$1,000,000	$200,000	$2,400,000	$1,811,850	4.239%
$800,000	$0	$600,000	$2,529,445	1.281%	$1,000,000	$0	$1,400,000	$2,966,890	3.764%
					$1,000,000	$200,000	$2,200,000	$2,011,850	3.194%
					$1,000,000	$0	$1,200,000	$3,166,890	2.677%
					$1,000,000	$200,000	$2,000,000	$2,211,850	2.431%
					$1,000,000	$0	$1,000,000	$3,366,890	2.092%
					$1,000,000	$200,000	$1,800,000	$2,411,850	1.791%
					$1,000,000	$0	$800,000	$3,566,890	1.641%
					$1,000,000	$200,000	$1,600,000	$2,611,850	1.439%
					$1,000,000	$400,000	$2,400,000	$1,656,810	1.232%
					$1,000,000	$0	$600,000	$3,766,890	1.092%
Other				2.897%	Other				6.839%
Game Value:			**0.90**		**Winner:**			**P1**	

Figure 77: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,100,000, and the Weighted by State, Unfloored Utility Vector is used.

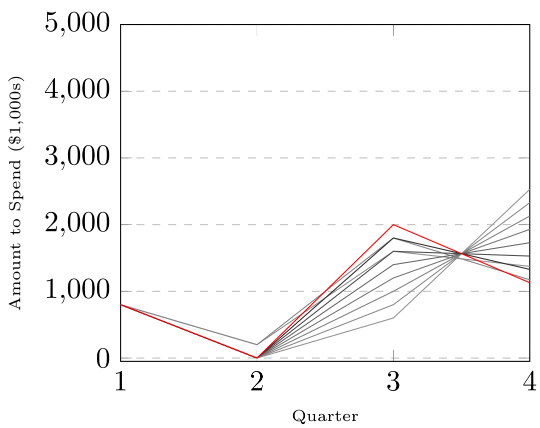


a P1: $800,000 Initial Budget **b** P2: $1,100,000 Initial Budget

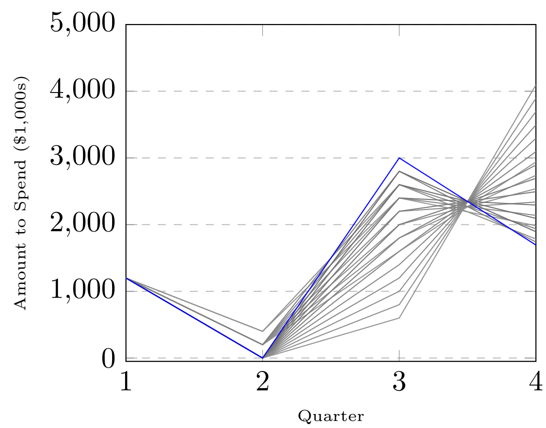
Figure 78: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,200,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$0	$2,000,000	$1,129,445	29.223%	$1,200,000	$0	$3,000,000	$1,694,167	9.734%
$800,000	$0	$1,800,000	$1,329,445	20.952%	$1,200,000	$0	$2,800,000	$1,894,167	8.767%
$800,000	$0	$1,600,000	$1,529,445	14.677%	$1,200,000	$0	$2,600,000	$2,094,167	7.889%
$800,000	$0	$1,400,000	$1,729,445	10.417%	$1,200,000	$0	$2,400,000	$2,294,167	7.119%
$800,000	$0	$1,200,000	$1,929,445	7.284%	$1,200,000	$0	$2,200,000	$2,494,167	6.457%
$800,000	$0	$1,000,000	$2,129,445	5.304%	$1,200,000	$0	$2,000,000	$2,694,167	5.817%
$800,000	$200,000	$1,800,000	$1,174,405	3.730%	$1,200,000	$200,000	$2,800,000	$1,739,127	5.282%
$800,000	$0	$800,000	$2,329,445	2.556%	$1,200,000	$0	$1,800,000	$2,894,167	4.914%
$800,000	$200,000	$1,600,000	$1,374,405	1.846%	$1,200,000	$200,000	$2,600,000	$1,939,127	4.378%
$800,000	$0	$600,000	$2,529,445	1.198%	$1,200,000	$0	$1,600,000	$3,094,167	3.859%
					$1,200,000	$200,000	$2,400,000	$2,139,127	3.458%
					$1,200,000	$0	$1,400,000	$3,294,167	3.163%
					$1,200,000	$200,000	$2,200,000	$2,339,127	2.836%
					$1,200,000	$0	$1,200,000	$3,494,167	2.596%
					$1,200,000	$200,000	$2,000,000	$2,539,127	2.241%
					$1,200,000	$0	$1,000,000	$3,694,167	2.098%
					$1,200,000	$200,000	$1,800,000	$2,739,127	1.946%
					$1,200,000	$400,000	$2,600,000	$1,784,087	1.739%
					$1,200,000	$0	$800,000	$3,894,167	1.514%
					$1,200,000	$200,000	$1,600,000	$2,939,127	1.372%
					$1,200,000	$400,000	$2,400,000	$1,984,087	1.292%
					$1,200,000	$0	$600,000	$4,094,167	1.153%
Other				2.813%	Other				10.376%
Game Value:			**1.01**		**Winner:**			**P1**	

Figure 79: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,200,000, and the Weighted by State, Unfloored Utility Vector is used.



a P1: $800,000 Initial Budget

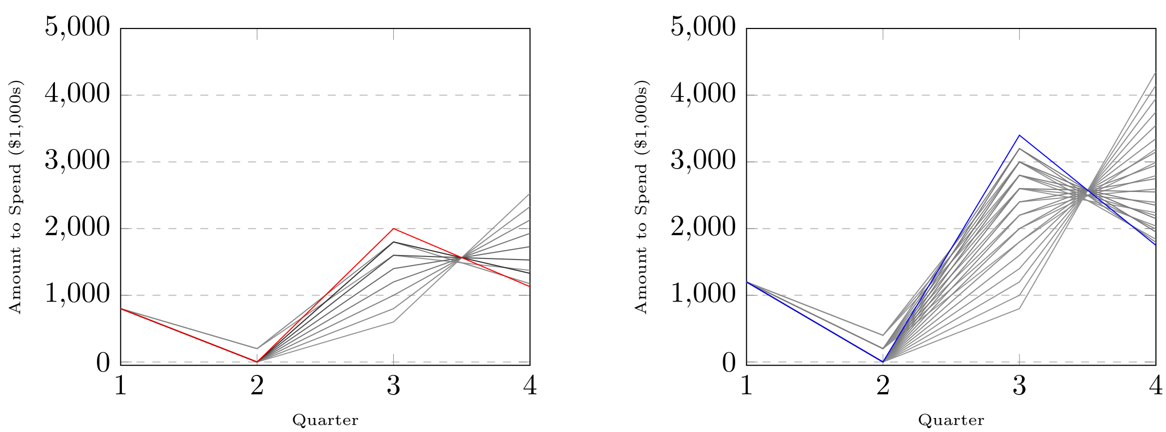


b P2: $1,200,000 Initial Budget

Figure 80: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,300,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$0	$2,000,000	$1,129,445	29.183%	$1,200,000	$0	$3,400,000	$1,749,252	8.103%
$800,000	$0	$1,800,000	$1,329,445	20.896%	$1,200,000	$0	$3,200,000	$1,949,252	7.415%
$800,000	$0	$1,600,000	$1,529,445	14.778%	$1,200,000	$0	$3,000,000	$2,149,252	6.943%
$800,000	$0	$1,400,000	$1,729,445	10.455%	$1,200,000	$0	$2,800,000	$2,349,252	6.272%
$800,000	$0	$1,200,000	$1,929,445	7.367%	$1,200,000	$0	$2,600,000	$2,549,252	5.856%
$800,000	$0	$1,000,000	$2,129,445	5.243%	$1,200,000	$0	$2,400,000	$2,749,252	5.351%
$800,000	$200,000	$1,800,000	$1,174,405	3.587%	$1,200,000	$200,000	$3,200,000	$1,794,212	4.958%
$800,000	$0	$800,000	$2,329,445	2.511%	$1,200,000	$0	$2,200,000	$2,949,252	4.433%
$800,000	$200,000	$1,600,000	$1,374,405	1.866%	$1,200,000	$200,000	$3,000,000	$1,994,212	4.129%
$800,000	$0	$600,000	$2,529,445	1.242%	$1,200,000	$0	$2,000,000	$3,149,252	3.847%
					$1,200,000	$200,000	$2,800,000	$2,194,212	3.360%
					$1,200,000	$0	$1,800,000	$3,349,252	3.292%
					$1,200,000	$200,000	$2,600,000	$2,394,212	2.990%
					$1,200,000	$0	$1,600,000	$3,549,252	2.698%
					$1,200,000	$200,000	$2,400,000	$2,594,212	2.409%
					$1,200,000	$0	$1,400,000	$3,749,252	2.217%
					$1,200,000	$200,000	$2,200,000	$2,794,212	2.001%
					$1,200,000	$400,000	$3,000,000	$1,839,172	1.974%
					$1,200,000	$0	$1,200,000	$3,949,252	1.815%
					$1,200,000	$200,000	$2,000,000	$2,994,212	1.625%
					$1,200,000	$400,000	$2,800,000	$2,039,172	1.517%
					$1,200,000	$0	$1,000,000	$4,149,252	1.356%
					$1,200,000	$200,000	$1,800,000	$3,194,212	1.301%
					$1,200,000	$400,000	$2,600,000	$2,239,172	1.165%
					$1,200,000	$0	$800,000	$4,349,252	1.083%
Other				2.872%	Other				11.89%
Game Value:			**1.00**		**Winner:**			**P1**	

Figure 81: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,300,000, and the Weighted by State, Unfloored Utility Vector is used.

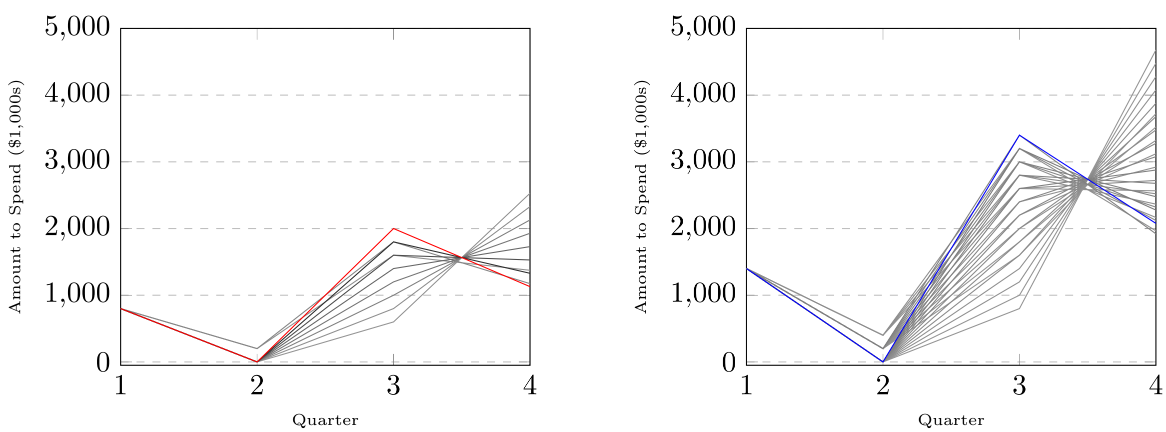


a P1: $800,000 Initial Budget **b** P2: $1,300,000 Initial Budget

Figure 82: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,400,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$0	$2,000,000	$1,129,445	29.458%	$1,400,000	$0	$3,400,000	$2,076,528	6.407%
$800,000	$0	$1,800,000	$1,329,445	20.927%	$1,400,000	$0	$3,200,000	$2,276,528	5.917%
$800,000	$0	$1,600,000	$1,529,445	14.641%	$1,400,000	$0	$3,000,000	$2,476,528	5.480%
$800,000	$0	$1,400,000	$1,729,445	10.332%	$1,400,000	$0	$2,800,000	$2,676,528	5.257%
$800,000	$0	$1,200,000	$1,929,445	7.250%	$1,400,000	$0	$2,600,000	$2,876,528	4.841%
$800,000	$0	$1,000,000	$2,129,445	5.165%	$1,400,000	$200,000	$3,400,000	$1,921,488	4.538%
$800,000	$200,000	$1,800,000	$1,174,405	3.697%	$1,400,000	$0	$2,400,000	$3,076,528	4.291%
$800,000	$0	$800,000	$2,329,445	2.496%	$1,400,000	$200,000	$3,200,000	$2,121,488	4.084%
$800,000	$200,000	$1,600,000	$1,374,405	1.867%	$1,400,000	$0	$2,200,000	$3,276,528	3.742%
$800,000	$0	$600,000	$2,529,445	1.208%	$1,400,000	$200,000	$3,000,000	$2,321,488	3.522%
					$1,400,000	$0	$2,000,000	$3,476,528	3.206%
					$1,400,000	$200,000	$2,800,000	$2,521,488	3.082%
					$1,400,000	$0	$1,800,000	$3,676,528	2.916%
					$1,400,000	$200,000	$2,600,000	$2,721,488	2.664%
					$1,400,000	$0	$1,600,000	$3,876,528	2.616%
					$1,400,000	$200,000	$2,400,000	$2,921,488	2.438%
					$1,400,000	$400,000	$3,200,000	$1,966,448	2.245%
					$1,400,000	$0	$1,400,000	$4,076,528	2.086%
					$1,400,000	$200,000	$2,200,000	$3,121,488	1.991%
					$1,400,000	$400,000	$3,000,000	$2,166,448	1.759%
					$1,400,000	$0	$1,200,000	$4,276,528	1.734%
					$1,400,000	$200,000	$2,000,000	$3,321,488	1.563%
					$1,400,000	$400,000	$2,800,000	$2,366,448	1.519%
					$1,400,000	$0	$1,000,000	$4,476,528	1.438%
					$1,400,000	$200,000	$1,800,000	$3,521,488	1.328%
					$1,400,000	$400,000	$2,600,000	$2,566,448	1.235%
					$1,400,000	$0	$800,000	$4,676,528	1.148%
					$1,400,000	$200,000	$1,600,000	$3,721,488	1.111%
Other				2.959%	Other				15.842%
Game Value:			**1.40**		**Winner:**			**P1**	

Figure 83: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,400,000, and the Weighted by State, Unfloored Utility Vector is used.

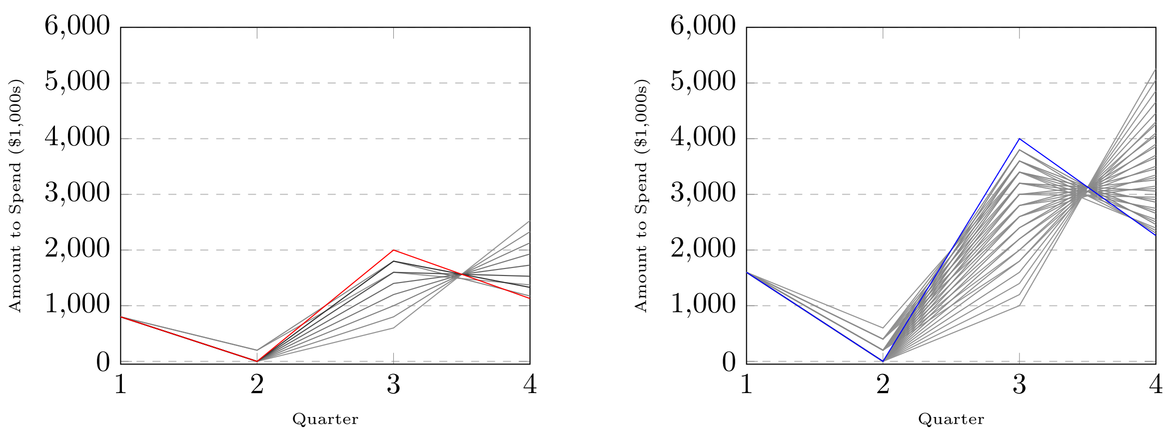


a P1: $800,000 Initial Budget **b** P2: $1,400,000 Initial Budget

Figure 84: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,600,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$800,000	$0	$2,000,000	$1,129,445	29.297%	$1,600,000	$0	$4,000,000	$2,258,890	4.255%
$800,000	$0	$1,800,000	$1,329,445	20.915%	$1,600,000	$0	$3,800,000	$2,458,890	4.107%
$800,000	$0	$1,600,000	$1,529,445	14.686%	$1,600,000	$0	$3,600,000	$2,658,890	3.942%
$800,000	$0	$1,400,000	$1,729,445	10.365%	$1,600,000	$0	$3,400,000	$2,858,890	3.823%
$800,000	$0	$1,200,000	$1,929,445	7.310%	$1,600,000	$0	$3,200,000	$3,058,890	3.620%
$800,000	$0	$1,000,000	$2,129,445	5.255%	$1,600,000	$0	$3,000,000	$3,258,890	3.469%
$800,000	$200,000	$1,800,000	$1,174,405	3.714%	$1,600,000	$200,000	$3,800,000	$2,303,850	3.269%
$800,000	$0	$800,000	$2,329,445	2.549%	$1,600,000	$0	$2,800,000	$3,458,890	3.190%
$800,000	$200,000	$1,600,000	$1,374,405	1.770%	$1,600,000	$200,000	$3,600,000	$2,503,850	3.021%
$800,000	$0	$600,000	$2,529,445	1.249%	$1,600,000	$0	$2,600,000	$3,658,890	2.934%
					$1,600,000	$200,000	$3,400,000	$2,703,850	2.855%
					$1,600,000	$0	$2,400,000	$3,858,890	2.615%
					$1,600,000	$0	$2,200,000	$4,058,890	2.541%
					$1,600,000	$200,000	$3,200,000	$2,903,850	2.511%
					$1,600,000	$200,000	$3,000,000	$3,103,850	2.349%
					$1,600,000	$0	$2,000,000	$4,258,890	2.217%
					$1,600,000	$200,000	$2,800,000	$3,303,850	2.192%
					$1,600,000	$400,000	$3,600,000	$2,348,810	2.134%
					$1,600,000	$0	$1,800,000	$4,458,890	1.952%
					$1,600,000	$200,000	$2,600,000	$3,503,850	1.900%
					$1,600,000	$400,000	$3,400,000	$2,548,810	1.882%
					$1,600,000	$0	$1,600,000	$4,658,890	1.720%
					$1,600,000	$200,000	$2,400,000	$3,703,850	1.633%
					$1,600,000	$400,000	$3,200,000	$2,748,810	1.556%
					$1,600,000	$200,000	$2,200,000	$3,903,850	1.447%
					$1,600,000	$0	$1,400,000	$4,858,890	1.440%
					$1,600,000	$400,000	$3,000,000	$2,948,810	1.378%
					$1,600,000	$0	$1,200,000	$5,058,890	1.330%
					$1,600,000	$200,000	$2,000,000	$4,103,850	1.289%
					$1,600,000	$400,000	$2,800,000	$3,148,810	1.160%
					$1,600,000	$0	$1,000,000	$5,258,890	1.155%
					$1,600,000	$200,000	$1,800,000	$4,303,850	1.087%
					$1,600,000	$400,000	$2,600,000	$3,348,810	1.061%
					$1,600,000	$600,000	$3,400,000	$2,393,770	1.009%
Other				2.89%	Other				21.957%
Game Value:			**1.76**		**Winner:**			**P1**	

Figure 85: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,600,000, and the Weighted by State, Unfloored Utility Vector is used.



a P1: $800,000 Initial Budget **b** P2: $1,600,000 Initial Budget

Figure 86: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

6.5 Weighted by Sample, Unfloored Utility Vector

The Weighted by Sample, Unfloored Utility Vector is most qualitatively similar to the Unweighted, Unfloored Utility Vector, in that it consists of 3 decreasing positive numbers followed by a negative number. It is defined by

$$[0.26559308, 0.242170061, 0.124087002, -0.419994208] \cdot 2 \times 10^{-5}. \tag{16}$$

Like the Unweighted, Unfloored vector, it rewards early spending, but punishes for spending in Q4. Also like the Unweighted, Unfloored vector, this vector has a conflicting interaction with the fundraising return function (which gives players more and more money to spend throughout the campaign) and the rule that mandates players must spend their entire remaining budget in Q4.

However, unlike the Unweighted, Unfloored vector, the results of this vector were so strange that the code inputs and outputs were checked twice. At the lowest budget pairing (**Figure 88**), the budgets are optimized primarily in descending order of Q1 spend amounts — in essence, optimal strategy is spending as much as possible in Q2, then minimizing amounts spend in Q1 and Q3, with seeming disregard for Q4. This behavior is different than that of the games in the Unweighted, Unfloored section, and most likely caused by the smaller ratios between values in the Weighted By Sample, Unfloored Utility Vector. The $800,000 vs. $900,000 game has similar behavior.

When P2's buget is increased to $1,000,000, however, the results change. Rather than a strategy consisting of upper and lower limits, P2's spending for *all four quarters* is somewhere in the middle of its suboptimal strategies. At first glance, it may seem like the optimal strategy changed dramatically. However, the optimal strategy's Q2 spending is the same as the previous game - it's the suboptimal strategies that have changed. While the optimal strategy does fundamentally change in general, Q2 spending remains the same.

The reason Q1 increases before Q2 is relatively simple — Q1 has a higher utility than Q2 for spending money. It may seem at first glance, then that Q2 spending should increase before Q3. Remember, though, that players are punished heavily for spending in Q4. Therefore, P2 has a very high incentive to spend as much as possible in Q3 to decrease their total remaining budget.

This type of strategy is employed for games when P2 has $1,100,000 and $1,200,000 as well. However, when P2 gets $1,300,000 or $1,400,000, the optimal strategy changes again! Presumably, the higher budgets allow for more freedom in P2's optimal strategy.

When P2's budget increases to $1,600,000, the optimal strategy changes yet again. However, examine the difference in percentages between P2's top two strategies in this game — 0.127%. Recall that the Monte-Carlo method was used to determine these percentages. How does one determine whether this change is really a result of a different game, or just a rounding error? In previous games, the P2 strategies followed a relatively intuitive path; however, in this game, there's been enough change to raise the suspicion that this is just an error.

One way to determine whether this is just an error is to run the code again, with more simulations. However, this would be extremely time-intensive; even 100,000 iterations took an immense amount of time. The key to estimating the error lies in the first game using this utility vector - $800,000 vs. $800,000. If calculated directly, these players should have identical strategies. Therefore, a rough error bound may be calculated by taking the maximum of the differences in calculated percentages between identical strategies. In other words, if the strategies for P1 are respectively ordered

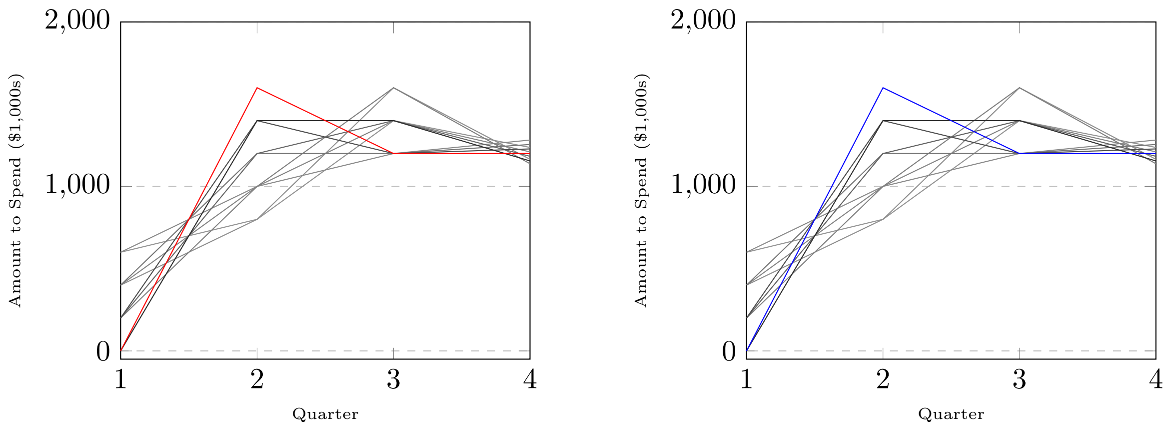
$$p_{1,1}, p_{1,2}, ..., p_{1,k}, p_{2,1}, p_{2,2}, ..., p_{2,k},$$

such that $p_{1,j} = p_{2,j} \forall j \in [1, k] \cap \mathbb{N}$, then the approximate error E can be calculated

$$E_{max} \approx max \mid \{p_{1,j} - p_{2,j}\} \forall j \in [1, k] \cap \mathbb{N}. \tag{17}$$

P1 $800,000 vs. P2 $800,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$1,600,000	$1,200,000	$1,200,355	29.347%	$0	$1,600,000	$1,200,000	$1,200,355	29.462%
$0	$1,400,000	$1,400,000	$1,155,395	20.903%	$0	$1,400,000	$1,400,000	$1,155,395	20.591%
$200,000	$1,400,000	$1,200,000	$1,227,588	14.553%	$200,000	$1,400,000	$1,200,000	$1,227,588	14.670%
$200,000	$1,200,000	$1,400,000	$1,182,628	10.495%	$200,000	$1,200,000	$1,400,000	$1,182,628	10.513%
$400,000	$1,200,000	$1,200,000	$1,254,820	7.304%	$400,000	$1,200,000	$1,200,000	$1,254,820	7.296%
$200,000	$1,000,000	$1,600,000	$1,137,668	5.141%	$200,000	$1,000,000	$1,600,000	$1,137,668	5.182%
$400,000	$1,000,000	$1,400,000	$1,209,860	3.667%	$400,000	$1,000,000	$1,400,000	$1,209,860	3.630%
$600,000	$1,000,000	$1,200,000	$1,282,052	2.577%	$600,000	$1,000,000	$1,200,000	$1,282,052	2.610%
$400,000	$800,000	$1,600,000	$1,164,900	1.836%	$400,000	$800,000	$1,600,000	$1,164,900	1.826%
$600,000	$800,000	$1,400,000	$1,237,092	1.271%	$600,000	$800,000	$1,400,000	$1,237,092	1.299%
Other				2.906%	Other				2.921%
Game Value:			**0.00**		**Winner:**			**Tie**	

Figure 87: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $800,000, and the Weighted by Sample, Unfloored Utility Vector is used.



a P1: $800,000 Initial Budget **b** P2: $800,000 Initial Budget

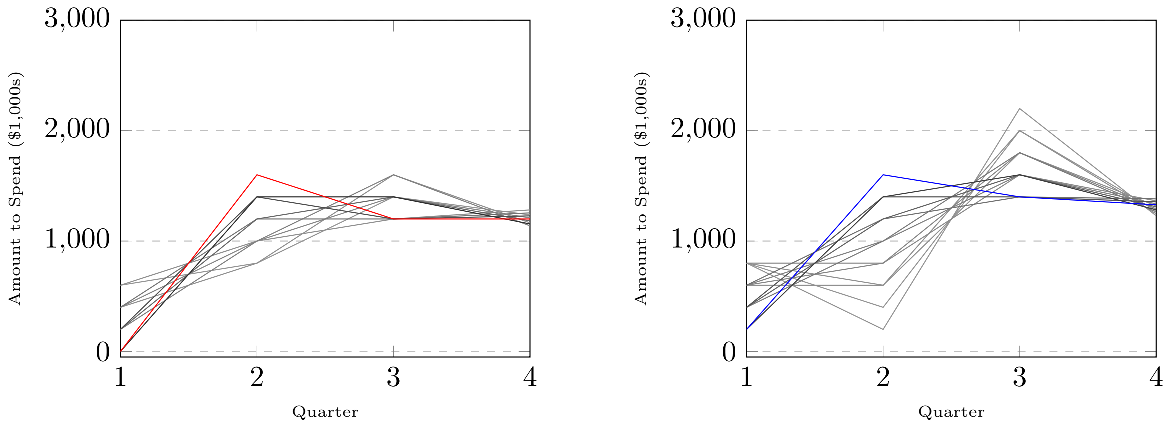
Figure 88: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

In this game, $E_{max} \approx 0.312$. Therefore, it's very well possible that the optimal strategy in the $800,000 vs. $1,600,000 game is chosen over the top suboptimal strategy as a result of error, rather than performance. Note that this is also possible in some previous games as well. This possibility is examined most closely in this game, however, since each of the previous games' optimal strategies follow an intuitive pattern.

Lastly, note that P2 is again the winner in these games. However, the game values have the smallest absolute values of any games simulated in the report. What P1 gains in the final quarter of the game, P2 just barely makes up for in Q1, Q2, and Q3. The game values never break an absolute value of 0.5, making this easily the fairest game in the report.

P1 $800,000 vs. P2 $900,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$1,600,000	$1,200,000	$1,200,355	29.466%	$200,000	$1,600,000	$1,400,000	$1,327,632	22.922%
$0	$1,400,000	$1,400,000	$1,155,395	20.738%	$200,000	$1,400,000	$1,600,000	$1,282,672	17.625%
$200,000	$1,400,000	$1,200,000	$1,227,588	14.691%	$400,000	$1,400,000	$1,400,000	$1,354,865	13.598%
$200,000	$1,200,000	$1,400,000	$1,182,628	10.447%	$400,000	$1,200,000	$1,600,000	$1,309,905	10.613%
$400,000	$1,200,000	$1,200,000	$1,254,820	7.350%	$600,000	$1,200,000	$1,400,000	$1,382,097	8.142%
$200,000	$1,000,000	$1,600,000	$1,137,668	5.297%	$400,000	$1,000,000	$1,800,000	$1,264,945	6.264%
$400,000	$1,000,000	$1,400,000	$1,209,860	3.697%	$600,000	$1,000,000	$1,600,000	$1,337,137	4.806%
$600,000	$1,000,000	$1,200,000	$1,282,052	2.496%	$600,000	$800,000	$1,800,000	$1,292,177	3.770%
$400,000	$800,000	$1,600,000	$1,164,900	1.719%	$800,000	$800,000	$1,600,000	$1,364,369	2.827%
$600,000	$800,000	$1,400,000	$1,237,092	1.209%	$600,000	$600,000	$2,000,000	$1,247,217	2.172%
					$800,000	$600,000	$1,800,000	$1,319,409	1.662%
					$800,000	$400,000	$2,000,000	$1,274,449	1.268%
					$800,000	$200,000	$2,200,000	$1,229,489	1.048%
Other				2.890%	Other				3.283%
Game Value:			**-0.11**		**Winner:**			**P2**	

Figure 89: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $900,000, and the Weighted by Sample, Unfloored Utility Vector is used.

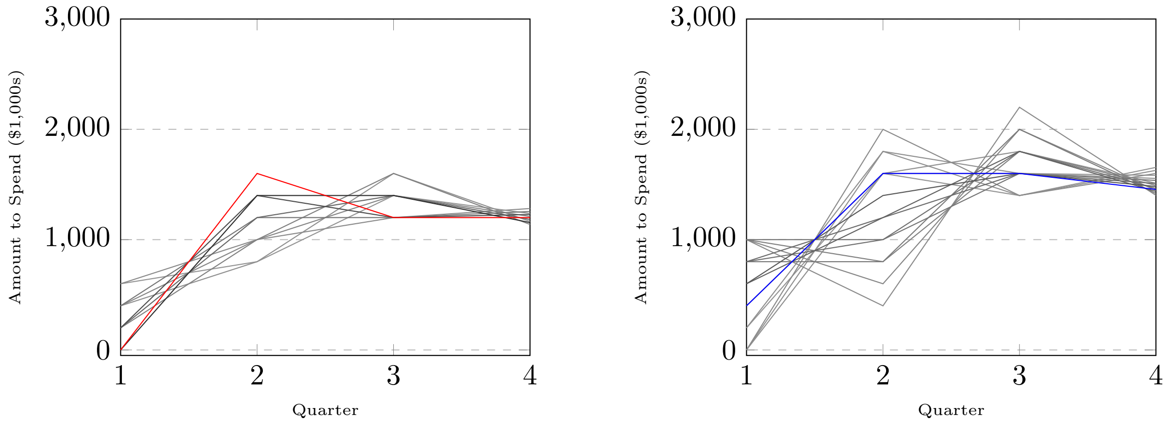


a P1: $800,000 Initial Budget **b** P2: $900,000 Initial Budget

Figure 90: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,000,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$1,600,000	$1,200,000	$1,200,355	29.238%	$400,000	$1,600,000	$1,600,000	$1,454,909	16.299%
$0	$1,400,000	$1,400,000	$1,155,395	20.783%	$600,000	$1,400,000	$1,600,000	$1,482,141	13.418%
$200,000	$1,400,000	$1,200,000	$1,227,588	14.763%	$600,000	$1,200,000	$1,800,000	$1,437,181	11.353%
$200,000	$1,200,000	$1,400,000	$1,182,628	10.452%	$800,000	$1,200,000	$1,600,000	$1,509,374	9.491%
$400,000	$1,200,000	$1,200,000	$1,254,820	7.322%	$800,000	$1,000,000	$1,800,000	$1,464,414	8.004%
$200,000	$1,000,000	$1,600,000	$1,137,668	5.141%	$1,000,000	$1,000,000	$1,600,000	$1,536,606	6.749%
$400,000	$1,000,000	$1,400,000	$1,209,860	3.625%	$800,000	$800,000	$2,000,000	$1,419,454	5.668%
$600,000	$1,000,000	$1,200,000	$1,282,052	2.562%	$1,000,000	$800,000	$1,800,000	$1,491,646	4.770%
$400,000	$800,000	$1,600,000	$1,164,900	1.829%	$1,000,000	$600,000	$2,000,000	$1,446,686	4.019%
$600,000	$800,000	$1,400,000	$1,237,092	1.327%	$1,000,000	$400,000	$2,200,000	$1,401,726	3.342%
					$0	$2,000,000	$1,400,000	$1,600,444	2.744%
					$0	$1,800,000	$1,600,000	$1,555,484	2.336%
					$200,000	$1,800,000	$1,400,000	$1,627,677	1.934%
					$0	$1,600,000	$1,800,000	$1,510,524	1.642%
					$200,000	$1,600,000	$1,600,000	$1,582,717	1.356%
					$400,000	$1,600,000	$1,400,000	$1,654,909	1.148%
Other				2.958%	Other				5.727%
Game Value:			**-0.16**		**Winner:**			**P2**	

Figure 91: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,000,000, and the Weighted by Sample, Unfloored Utility Vector is used.



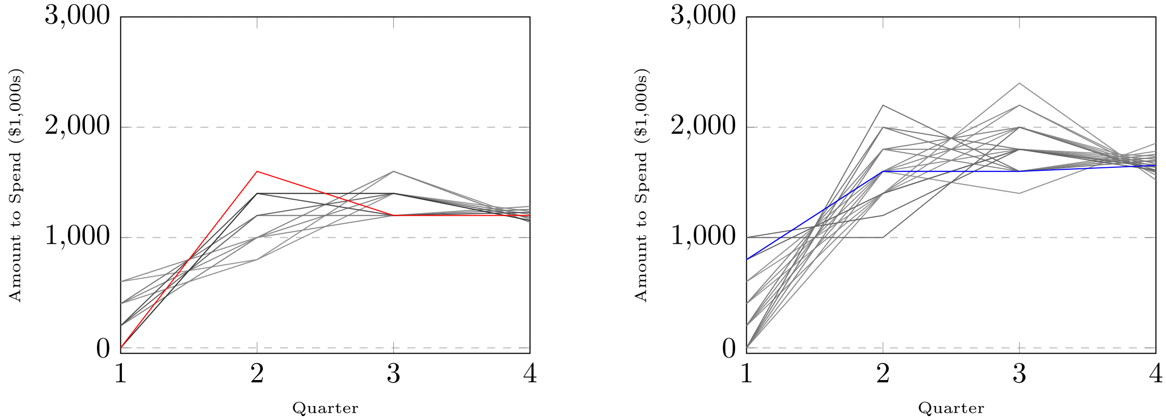
a P1: $800,000 Initial Budget

b P2: $1,000,000 Initial Budget

Figure 92: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,100,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$1,600,000	$1,200,000	$1,200,355	28.970%	$800,000	$1,600,000	$1,600,000	$1,654,378	12.958%
$0	$1,400,000	$1,400,000	$1,155,395	20.864%	$800,000	$1,400,000	$1,800,000	$1,609,418	11.272%
$200,000	$1,400,000	$1,200,000	$1,227,588	14.714%	$1,000,000	$1,200,000	$1,800,000	$1,636,650	9.877%
$200,000	$1,200,000	$1,400,000	$1,182,628	10.518%	$1,000,000	$1,000,000	$2,000,000	$1,591,690	8.560%
$400,000	$1,200,000	$1,200,000	$1,254,820	7.462%	$0	$2,200,000	$1,600,000	$1,700,489	7.535%
$200,000	$1,000,000	$1,600,000	$1,137,668	5.224%	$0	$2,000,000	$1,800,000	$1,655,529	6.517%
$400,000	$1,000,000	$1,400,000	$1,209,860	3.651%	$200,000	$2,000,000	$1,600,000	$1,727,721	5.626%
$600,000	$1,000,000	$1,200,000	$1,282,052	2.595%	$0	$1,800,000	$2,000,000	$1,610,569	4.961%
$400,000	$800,000	$1,600,000	$1,164,900	1.860%	$200,000	$1,800,000	$1,800,000	$1,682,761	4.249%
$600,000	$800,000	$1,400,000	$1,237,092	1.232%	$400,000	$1,800,000	$1,600,000	$1,754,954	3.753%
					$0	$1,600,000	$2,200,000	$1,565,609	3.249%
					$200,000	$1,600,000	$2,000,000	$1,637,801	2.833%
					$400,000	$1,600,000	$1,800,000	$1,709,994	2.428%
					$0	$1,400,000	$2,400,000	$1,520,649	2.109%
					$600,000	$1,600,000	$1,600,000	$1,782,186	1.880%
					$200,000	$1,400,000	$2,200,000	$1,592,841	1.555%
					$800,000	$1,600,000	$1,400,000	$1,854,378	1.413%
					$400,000	$1,400,000	$2,000,000	$1,665,034	1.256%
					$600,000	$1,400,000	$1,800,000	$1,737,226	1.047%
Other				2.910%	Other				6.922%
Game Value:			**-0.11**		**Winner:**			**P2**	

Figure 93: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,100,000, and the Weighted by Sample, Unfloored Utility Vector is used.

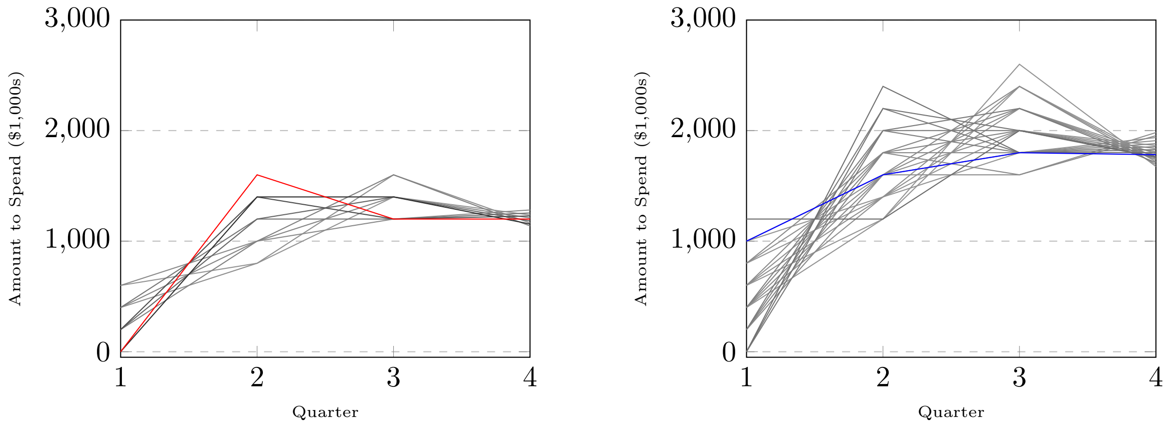


a P1: $800,000 Initial Budget **b** P2: $1,100,000 Initial Budget

Figure 94: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,200,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$1,600,000	$1,200,000	$1,200,355	29.251%	$1,000,000	$1,600,000	$1,800,000	$1,781,655	9.784%
$0	$1,400,000	$1,400,000	$1,155,395	20.739%	$1,200,000	$1,200,000	$2,000,000	$1,763,927	8.798%
$200,000	$1,400,000	$1,200,000	$1,227,588	14.817%	$0	$2,400,000	$1,800,000	$1,800,533	7.833%
$200,000	$1,200,000	$1,400,000	$1,182,628	10.404%	$0	$2,200,000	$2,000,000	$1,755,573	7.184%
$400,000	$1,200,000	$1,200,000	$1,254,820	7.384%	$200,000	$2,200,000	$1,800,000	$1,827,766	6.482%
$200,000	$1,000,000	$1,600,000	$1,137,668	5.197%	$0	$2,000,000	$2,200,000	$1,710,613	5.851%
$400,000	$1,000,000	$1,400,000	$1,209,860	3.714%	$200,000	$2,000,000	$2,000,000	$1,782,806	5.359%
$600,000	$1,000,000	$1,200,000	$1,282,052	2.530%	$400,000	$2,000,000	$1,800,000	$1,854,998	4.894%
$400,000	$800,000	$1,600,000	$1,164,900	1.794%	$200,000	$1,800,000	$2,200,000	$1,737,846	4.189%
$600,000	$800,000	$1,400,000	$1,237,092	1.251%	$400,000	$1,800,000	$2,000,000	$1,810,038	3.904%
					$600,000	$1,800,000	$1,800,000	$1,882,230	3.463%
					$200,000	$1,600,000	$2,400,000	$1,692,886	3.163%
					$800,000	$1,800,000	$1,600,000	$1,954,423	2.866%
					$400,000	$1,600,000	$2,200,000	$1,765,078	2.604%
					$600,000	$1,600,000	$2,000,000	$1,837,270	2.285%
					$800,000	$1,600,000	$1,800,000	$1,909,463	2.074%
					$400,000	$1,400,000	$2,400,000	$1,720,118	1.839%
					$1,000,000	$1,600,000	$1,600,000	$1,981,655	1.661%
					$600,000	$1,400,000	$2,200,000	$1,792,310	1.564%
					$800,000	$1,400,000	$2,000,000	$1,864,503	1.455%
					$400,000	$1,200,000	$2,600,000	$1,675,158	1.226%
					$1,000,000	$1,400,000	$1,800,000	$1,936,695	1.168%
					$600,000	$1,200,000	$2,400,000	$1,747,350	1.019%
Other				2.919%	Other				9.335%
Game Value:			**-0.13**		**Winner:**			**P2**	

Figure 95: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,200,000, and the Weighted by Sample, Unfloored Utility Vector is used.

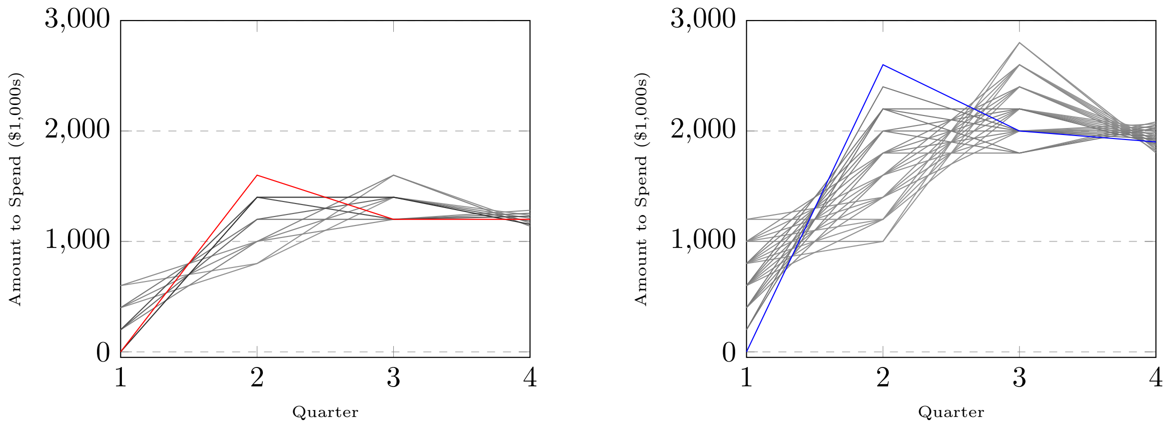


a P1: $800,000 Initial Budget **b** P2: $1,200,000 Initial Budget

Figure 96: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,300,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$1,600,000	$1,200,000	$1,200,355	29.504%	$0	$2,600,000	$2,000,000	$1,900,578	8.061%
$0	$1,400,000	$1,400,000	$1,155,395	20.814%	$200,000	$2,400,000	$2,000,000	$1,927,810	7.393%
$200,000	$1,400,000	$1,200,000	$1,227,588	14.680%	$200,000	$2,200,000	$2,200,000	$1,882,850	6.951%
$200,000	$1,200,000	$1,400,000	$1,182,628	10.272%	$400,000	$2,200,000	$2,000,000	$1,955,043	6.192%
$400,000	$1,200,000	$1,200,000	$1,254,820	7.476%	$600,000	$2,200,000	$1,800,000	$2,027,235	5.796%
$200,000	$1,000,000	$1,600,000	$1,137,668	5.136%	$400,000	$2,000,000	$2,200,000	$1,910,083	5.285%
$400,000	$1,000,000	$1,400,000	$1,209,860	3.629%	$600,000	$2,000,000	$2,000,000	$1,982,275	4.858%
$600,000	$1,000,000	$1,200,000	$1,282,052	2.605%	$800,000	$2,000,000	$1,800,000	$2,054,467	4.529%
$400,000	$800,000	$1,600,000	$1,164,900	1.801%	$400,000	$1,800,000	$2,400,000	$1,865,123	4.156%
$600,000	$800,000	$1,400,000	$1,237,092	1.221%	$600,000	$1,800,000	$2,200,000	$1,937,315	3.898%
					$800,000	$1,800,000	$2,000,000	$2,009,507	3.514%
					$400,000	$1,600,000	$2,600,000	$1,820,163	3.129%
					$1,000,000	$1,800,000	$1,800,000	$2,081,699	3.024%
					$600,000	$1,600,000	$2,400,000	$1,892,355	2.719%
					$800,000	$1,600,000	$2,200,000	$1,964,547	2.487%
					$1,000,000	$1,600,000	$2,000,000	$2,036,739	2.321%
					$600,000	$1,400,000	$2,600,000	$1,847,395	2.090%
					$800,000	$1,400,000	$2,400,000	$1,919,587	1.944%
					$1,000,000	$1,400,000	$2,200,000	$1,991,779	1.793%
					$600,000	$1,200,000	$2,800,000	$1,802,435	1.641%
					$1,200,000	$1,400,000	$2,000,000	$2,063,972	1.525%
					$800,000	$1,200,000	$2,600,000	$1,874,627	1.367%
					$1,000,000	$1,200,000	$2,400,000	$1,946,819	1.282%
					$1,200,000	$1,200,000	$2,200,000	$2,019,012	1.211%
					$800,000	$1,000,000	$2,800,000	$1,829,667	1.051%
					$1,000,000	$1,000,000	$2,600,000	$1,901,859	1.003%
Other				2.862%	Other				10.780%
Game Value:			**-0.18**		**Winner:**			**P2**	

Figure 97: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,300,000, and the Weighted by Sample, Unfloored Utility Vector is used.

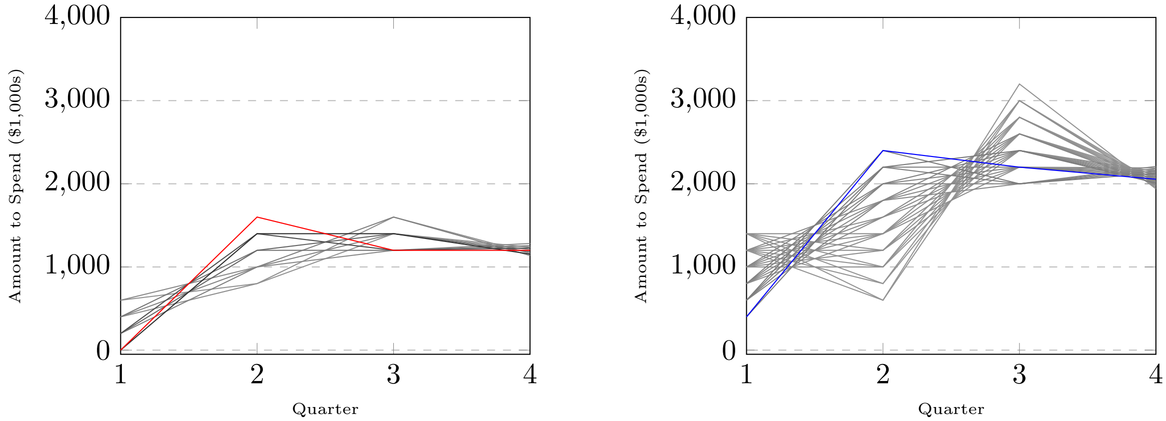


a P1: $800,000 Initial Budget **b** P2: $1,300,000 Initial Budget

Figure 98: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,400,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$1,600,000	$1,200,000	$1,200,355	29.340%	$400,000	$2,400,000	$2,200,000	$2,055,087	6.408%
$0	$1,400,000	$1,400,000	$1,155,395	20.817%	$600,000	$2,400,000	$2,000,000	$2,127,279	5.896%
$200,000	$1,400,000	$1,200,000	$1,227,588	14.757%	$400,000	$2,200,000	$2,400,000	$2,010,127	5.540%
$200,000	$1,200,000	$1,400,000	$1,182,628	10.425%	$600,000	$2,200,000	$2,200,000	$2,082,319	5.272%
$400,000	$1,200,000	$1,200,000	$1,254,820	7.246%	$800,000	$2,200,000	$2,000,000	$2,154,512	4.981%
$200,000	$1,000,000	$1,600,000	$1,137,668	5.142%	$600,000	$2,000,000	$2,400,000	$2,037,359	4.634%
$400,000	$1,000,000	$1,400,000	$1,209,860	3.708%	$800,000	$2,000,000	$2,200,000	$2,109,552	4.280%
$600,000	$1,000,000	$1,200,000	$1,282,052	2.621%	$1,000,000	$2,000,000	$2,000,000	$2,181,744	3.972%
$400,000	$800,000	$1,600,000	$1,164,900	1.772%	$600,000	$1,800,000	$2,600,000	$1,992,399	3.805%
$600,000	$800,000	$1,400,000	$1,237,092	1.268%	$800,000	$1,800,000	$2,400,000	$2,064,592	3.489%
					$1,000,000	$1,800,000	$2,200,000	$2,136,784	3.304%
					$1,200,000	$1,800,000	$2,000,000	$2,208,976	3.067%
					$800,000	$1,600,000	$2,600,000	$2,019,632	2.890%
					$1,000,000	$1,600,000	$2,400,000	$2,091,824	2.689%
					$1,200,000	$1,600,000	$2,200,000	$2,164,016	2.453%
					$800,000	$1,400,000	$2,800,000	$1,974,672	2.412%
					$1,000,000	$1,400,000	$2,600,000	$2,046,864	2.287%
					$1,200,000	$1,400,000	$2,400,000	$2,119,056	2.118%
					$1,400,000	$1,400,000	$2,200,000	$2,191,248	1.913%
					$1,000,000	$1,200,000	$2,800,000	$2,001,904	1.746%
					$1,200,000	$1,200,000	$2,600,000	$2,074,096	1.715%
					$1,400,000	$1,200,000	$2,400,000	$2,146,288	1.629%
					$1,000,000	$1,000,000	$3,000,000	$1,956,944	1.507%
					$1,200,000	$1,000,000	$2,800,000	$2,029,136	1.459%
					$1,400,000	$1,000,000	$2,600,000	$2,101,328	1.251%
					$1,200,000	$800,000	$3,000,000	$1,984,176	1.233%
					$1,400,000	$800,000	$2,800,000	$2,056,368	1.195%
					$1,200,000	$600,000	$3,200,000	$1,939,216	1.071%
					$1,400,000	$600,000	$3,000,000	$2,011,408	1.068%
Other				2.904%	Other				14.716%
Game Value:			**-0.27**		**Winner:**			**P2**	

Figure 99: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,400,000, and the Weighted by Sample, Unfloored Utility Vector is used.

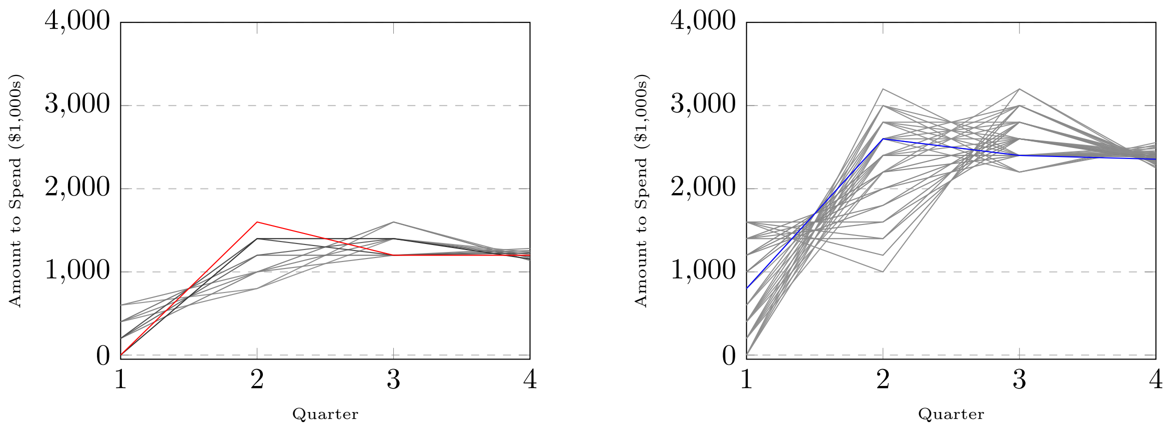


a P1: $800,000 Initial Budget **b** P2: $1,400,000 Initial Budget

Figure 100: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

P1 $800,000 vs. P2 $1,600,000									
Player 1	(P1)				Player 2	(P2)			
Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %	Q1 Strategy	Q2 Strategy	Q3 Strategy	Q4 Strategy	Play %
$0	$1,600,000	$1,200,000	$1,200,355	29.681%	$800,000	$2,600,000	$2,400,000	$2,354,601	4.325%
$0	$1,400,000	$1,400,000	$1,155,395	20.696%	$1,000,000	$2,400,000	$2,400,000	$2,381,833	4.198%
$200,000	$1,400,000	$1,200,000	$1,227,588	14.725%	$1,000,000	$2,200,000	$2,600,000	$2,336,873	4.015%
$200,000	$1,200,000	$1,400,000	$1,182,628	10.256%	$1,200,000	$2,200,000	$2,400,000	$2,409,065	3.735%
$400,000	$1,200,000	$1,200,000	$1,254,820	7.366%	$1,200,000	$2,000,000	$2,600,000	$2,364,105	3.575%
$200,000	$1,000,000	$1,600,000	$1,137,668	5.175%	$1,400,000	$2,000,000	$2,400,000	$2,436,297	3.435%
$400,000	$1,000,000	$1,400,000	$1,209,860	3.721%	$1,200,000	$1,800,000	$2,800,000	$2,319,145	3.343%
$600,000	$1,000,000	$1,200,000	$1,282,052	2.609%	$1,400,000	$1,800,000	$2,600,000	$2,391,337	3.155%
$400,000	$800,000	$1,600,000	$1,164,900	1.764%	$1,400,000	$1,600,000	$2,800,000	$2,346,377	2.972%
$600,000	$800,000	$1,400,000	$1,237,092	1.176%	$1,400,000	$1,400,000	$3,000,000	$2,301,417	2.916%
					$1,600,000	$1,600,000	$2,600,000	$2,418,570	2.909%
					$1,600,000	$1,400,000	$2,800,000	$2,373,610	2.695%
					$1,600,000	$1,200,000	$3,000,000	$2,328,650	2.583%
					$1,600,000	$1,000,000	$3,200,000	$2,283,690	2.484%
					$0	$3,200,000	$2,400,000	$2,400,711	2.314%
					$0	$3,000,000	$2,600,000	$2,355,751	2.259%
					$200,000	$3,000,000	$2,400,000	$2,427,944	2.213%
					$400,000	$3,000,000	$2,200,000	$2,500,136	2.176%
					$0	$2,800,000	$2,800,000	$2,310,791	1.983%
					$200,000	$2,800,000	$2,600,000	$2,382,984	1.818%
					$0	$2,600,000	$3,000,000	$2,265,831	1.702%
					$400,000	$2,800,000	$2,400,000	$2,455,176	1.700%
					$600,000	$2,800,000	$2,200,000	$2,527,368	1.586%
					$200,000	$2,600,000	$2,800,000	$2,338,024	1.525%
					$400,000	$2,600,000	$2,600,000	$2,410,216	1.511%
					$600,000	$2,600,000	$2,400,000	$2,482,408	1.407%
					$200,000	$2,400,000	$3,000,000	$2,293,064	1.406%
					$800,000	$2,600,000	$2,200,000	$2,554,601	1.326%
					$400,000	$2,400,000	$2,800,000	$2,365,256	1.302%
					$600,000	$2,400,000	$2,600,000	$2,437,448	1.212%
					$200,000	$2,200,000	$3,200,000	$2,248,104	1.145%
					$800,000	$2,400,000	$2,400,000	$2,509,641	1.079%
					$400,000	$2,200,000	$3,000,000	$2,320,296	1.035%
Other				2.831%	Other				22.961%
Game Value:			**-0.36**		**Winner:**			**P2**	

Figure 101: Mixed strategy output, when P1 is given an initial budget of $800,000, P2 is given an initial budget of $1,600,000, and the Weighted by Sample, Unfloored Utility Vector is used.



a P1: $800,000 Initial Budget

b P2: $1,600,000 Initial Budget

Figure 102: Overlaid strategies for P1 and P2. Colored line indicates top strategy. Darker gray lines indicate heavily used strategies, while lighter gray lines indicated lightly used strategies.

7 Critiques

7.1 Fundraising Return Function

The most presumptuous aspect of this model is, without question, the fundraising return function. Fundraising in Senate campaigns is an extremely complex equation, and in calculating a fundraising return, one must consider incumbency status, party, state population, previous donors, and a myriad of other parameters not touched in this model. Extensive research has been done to attempt to predict fundraising in public office campaigns [8], and almost none of it is considered in this model. This model's fundraising return function was used purely as a mechanic to create a solvable game, and should not be cited as anything more.

Additionally, the most recent elections were characterized by a paradigm shift in methods of fundraising. With the rise of electronic advertising and a swtich from over-the-phone to internet donations, the cost and methods of fundraising are rapidly and unpredictably shifting [11]. This creates a very short expiration date on any function of advertising, and likely skews even the extremely inaccurate numbers give in **Section 5.1**. On top of this, due to the reporting requirements of the FEC, candidates are not required to be very specific or standard in thier spending and fundraising reporting [13]. This creates a huge error in the fundraising return function, which could not even limit the parameter of spending to purely advertising spending, but rather all expenditures in a campaign.

Even under the invalid assumption that fundraising can be found strictly as a function of past fundraising and past spending, this model's fundraising return function is too simple. Parameters would almost definitely reach much further back than the most recent quarter of spending and fundraising; however, without much more time and research into the field, creating a more nuanced function would create a much more computationally expensive algorithm, with almost no guarantee of a more accurate answer. Therefore, to simplify the model and to make as few faulty assumptions as possible, the simplest fundraising return function was used.

7.2 Utility Function

The first problems with this model's utility function mirror those outlined in the critique of the Fundraising Return Function, since one of its parameters is a candidate's spending. In addition, a different year's data was used for the utility function (2018) than for the fundraising return function (2016). This was done to limit confounding variables present in one year to one of the two functions. Obviously, however, confounding variables exist for both years, and the decision to use different years for each of these functions may have only exacerbated these errors.

A more glaring issue with this function, however, is the polling data used to inform it. Data collected from 538 was neither uniform nor unbiased; most data points corresponded to different polling agencies, and are almost guaranteed to have employed different polling strategies. An ideal dataset would have polls conducted by the same agencies employing the same polling methods, and would be much more standardized in collection methods. However, no such dataset is publicly available. The utility function was made through grasping at straw polls, so to speak. The data sets considered were the only ones even remotely close to the dataset needed to make this equation more accurate, and even these can be described as "less than ideal" at best.

Another issue with the utility function is the way in which the regression lines were calculated. Specifically, independent regression lines were calculated for each quarter and candidate. The idea behind this was to isolate polling errors to the fiscal quarters in which they were taken. However, as a result, the endpoints of these regression lines did not necessarily match up, and so discrepancies between quarters were ignored. Another method to calculate these regression lines would be, after calculating endpoints of regression lines for each quarter individually, to average discrepant endpoints and to set new regression lines between these points. However, this could take large errors in one quarter and affect polling numbers of its adjacent quarters. Picking between these methods is like picking the better of two poor candidate choices. The first option was picked in order to isolate errors to a single quarter.

7.3 Monte-Carlo Method

Solving the matrices was the most time-prohibitive element of this model. It would have been ideal to solve each matrix directly, rather than using the Monte-Carlo method. However, even small initial budgets created huge strategy arrays, which in turn created immense matrices. These matrices could not be solved in a reasonable amount of time; therefore, Monte-Carlo had to be used.

The number of iterations for each matrix may also be criticized. Each matrix was only run 100,000 times, which is relatively low for this sort of problem. However, even running only one utility vector 100,000 times for each set of budgets cost several days of uninterrupted computing. Increasing this iteration size would have created weeks of nonstop computing work. Efficiency of the model has huge room for improvement.

7.4 Error Analysis

Because the Monte-Carlo method always produces an error, it's very useful to estimate this error. This can be done very roughly, as outlined in Section 6.5: **Weighted by Sample, Unfloored Utility Vector**, by calculating the maximum difference in percentages between identical strategies when the two players are given identical budgets. However, there are whole libraries and some careers dedicated entirely to error analysis. While the error analysis used is somewhat useful for this report, it can be calculated much more accurately using more sophisticated methods. This is yet another area in which the report could improve.

8 Summary

While this model has many facets that can and should be improved upon, it's a good starting point from which a more realistic U.S. Senate campaign model may be built. Using data and knowledge provided, this model suggests that the most effective time to spend money on a campaign is the final quarter of that campaign, regardless of initial budget and opponent's initial budget. It also suggests that whether a player has a higher initial budget than their opponent is an extremely strong indicator that player will win the election. While some sub optimal strategies allow for the lower-budget player to win, playing optimally allows the higher-budget player to win in every case. Major areas for improvement of the model lie in the fundraising return function, the utility vector, efficiency of the testing code, and refinement of tests.

References

[1] "Battle for the Senate 2018." RealClearPolitics.com, RealClearPolitics.

[2] "Campaign Finance Database." FEC.gov.

[3] Ferguson, Thomas S. "Game Theory." World Scientific, 2017.

[4] Hokit, Max. Personal interview. 4 December 2018.

[5] Ingham, Sean. "Pareto-Optimality." Encyclopædia Britannica, Encyclopædia Britannica, Inc., 27 Sept. 2017.

[6] Kenton, Will. "Monte-Carlo Simulation." Investopedia, Investopedia, 6 Mar. 2019.

[7] Mendes, Anthony. Count On It. Fall Quarter 2018, California Polytechnic State University in San Luis Obispo, 2018.

[8] Mixon, Franklin G., et al. "Pivotal Power Brokers: Theory and Evidence on Political Fundraising." Public Choice, vol. 123, no. 3/4, 2005, pp. 477493. JSTOR.

[9] Norwood, Arlisha R. "Grace Hopper." WomensHistory.org, National Women's History Museum, 2017.

[10] "Search and Learn the Bias of News Media." Media Bias/Fact Check, mediabiasfactcheck.com/.

[11] Simpson, D., O'Shaughnessy, B., & Schakowsky, J. (2016). Raising Money. In Winning Elections in the 21st Century (pp. 49-69). University Press of Kansas.

[12] "SLOPE Function." Support.Office.com, Microsoft. Microsoft Excel function documentation.

[13] United States, Congress, "Recording Receipts." Help for Candidates and Committees, FEC.

[14] "United States Senate Elections, 2016." Ballotpedia.org, Ballotpedia.

[15] "United States Senate Elections, 2018." Ballotpedia.org, Ballotpedia.

[16] "U.S. Senate Polls." FiveThirtyEight.com, FiveThirtyEight. 5 Nov. 2018.

[17] Wetzel, Jacqueline R. Personal interview. 16 October 2018.

```
"""
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
APPENDIX
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Author: Tim Wetzel
Advisor: Tony Mendes
Paper Title: Game Theory and U.S. Senate Campaigns
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
March 20, 2019
Senior Project Code
California Polytechnic University
San Luis Obispo, CA
Mathematics Department
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Before use, please contact:
*
Tim Wetzel
timawetzel@gmail.com
https://www.linkedin.com/in/timawetzel/
https://timawetzel.wixsite.com/resume
*
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
"""

from itertools import *
from math import *
import numpy as np
import random
import matplotlib.pyplot as plt
from timeit import default_timer as timer
import multiprocessing

def gamesolution(A):
#Solves a payout matrix

      m = len(A)
      n = len(A[0])

      # pad entries to ensure that the value is positive and create
initial tableau
      p = min(A[0] + [0])
      T = [[j-p+1 for j in A[i]]+[0]*i+[1]+[0]*(m-i-1)+[1] for i in
range(m)]
      T += [[-1]*n + [0]*(m+1)]

      # implement simplex algorithm
      while min(T[-1]) < 0:

                  # find pivot column c
```

```
                c = T[-1].index(min(T[-1]))

                # find pivot row r
                ratios = [i[-1]/float(i[c]) if i[c] > 0 else 'inf' for i
in T[:-1]]
                r = ratios.index(min([a for a in ratios if a != 'inf']))

                # perform pivot
                T[r] = [i/float(T[r][c]) for i in T[r]]
                for i in range(m + 1):
                        if i != r:
                            T[i] = [T[i][j] - T[i][c]*T[r][j] for j
in range(m + n + 1)]
                            #watch out for roundoff errors
                            for j in range(m + n + 1):
                                    if abs(T[i][j]) < pow(10,-
8): T[i][j] = 0

        # get strategies from final tableau
        v = 1/float(T[-1][-1])

        row = [v*i for i in T[-1][-m-1:-1]]

        col = [0 for i in A[0]]
        for i in range(m):
                c = list(zip(*T)).index(tuple([0]*i+[1]+[0]*(m-i)))
                if c < len(col):
                        col[c] += v*T[i][-1]

        return [row, col, v+p-1]

def Z(num):
#Used to assist Filter()
        if num < 0:
                return 0
        else:
                return 1

def DOMINATION(strategy_array, weeks):
#Used to eliminate dominated strategies in Test()
        breaker = 0
        dominatingstrategies = [strategy_array[0]] #Only the first strategy
is in there
        i = 0
        while i < len(strategy_array):
                d = 0
                while d < len(dominatingstrategies):
                        j = 0
                        addit = 0
                        while j < weeks:
                            if strategy_array[i][j] >
dominatingstrategies[d][j]:
                                    addit = 1
                                    break
```

```
                              j = j+1
                        d = d+1
                        if addit == 0:
                              break
            if addit == 1:

      dominatingstrategies.append(strategy_array[i])
            i = i + 1
      return dominatingstrategies

def FundReturn(lastquarterspend, lastquarterfundraising):
      #Fundraiaing return function
      return (.1124)*lastquarterspend + (1.1057)*lastquarterfundraising

def CreateAllStrategies(budget, weeks):
      week = 1
      y = budget*1.0
      strategies = []
      strategies2 =[]
      finalstrats = []
      spend = 0
      ifund = 1.0
      while spend <= budget:
            strategies.append([spend, y - spend, y*ifund]) #([how
much you spend, how much you have left, previous fundraising])
            spend = spend + 1
      week = week + 1
      #We now have an array of every spending strategy for week 1
      while len(strategies[0]) < weeks+1: #Starts week 2, ends week n
            for i in strategies: #i is a strategy vector
                  f = FundReturn(i[len(i)-3], i[len(i)-1])
#Extra money we can spend RIGHT NOW
                  y =  i[len(i)-2] +  f #remaining budget plus
fundraising return
                  j = 0 #will cycle through every possible
spend amount for this quarter
                  while j <= y:
                        k = []
                        ticker = 0
                        while len(k)<len(i)-2:
                                    k.append(i[ticker])
                                    ticker = ticker + 1
                        k.append(j)
                        k.append(y-j)
                        k.append(f)
                        strategies2.append(k) #Add new strategy
to new strategy list
                        j = j + 1 #Next spending amount!
            strategies = strategies2 #Should be significantly larger
vector
            strategies2 = []
            week = week + 1
      for i in strategies: #Final spend is already in there, get rid of
last term
```

```
                j = i[:len(i)-2]
                j.append(i[len(i)-1]+i[len(i)-2])
                finalstrats.append(j)
        return finalstrats

def Setupmatrix(s1s, s2s, n, p):
        M = []
        for i in range(len(s2s)):
                N = []
                for j in range(len(s1s)):
                        payout = 0
                        for week in range(n):
                                payout = payout +
p[week]*(s1s[j][week]-s2s[i][week])
                        N.append(payout)
                M.append(N)
        M = [[i for i in a] for a in zip(*M)]
        return M

def Montecarlo(strats1, size1, strats2, size2, howmanytimes, n, p,
indicator1, indicator2):
        winners = [[0 for i in strats1],[0 for i in strats2]]
        if indicator1 and indicator2:
                for simulation in range(howmanytimes):
                        inds1 = random.sample(range(len(strats1)-1),
size1)
                        inds2 = random.sample(range(len(strats2)-1),
size2)
                        subset1 = [strats1[i] for i in inds1] #random
strategies for each player using indices
                        subset2 = [strats2[i] for i in inds2]
                        j = 0
                        M = Setupmatrix(subset1,subset2, n, p)
                        solution = gamesolution(M)
                        #Player 1:
                        while j < len(solution[0]): #for each
proportion:
                                winners[0][inds1[j]] += solution[0][j]
                                j+=1
                        j = 0
                        #Player 2:
                        while j < len(solution[1]): #for each
proportion:
                                winners[1][inds2[j]] += solution[1][j]
                                j+=1
                winners = [[1.0*i/howmanytimes for i in winners[0]],
[1.0*i/howmanytimes for i in winners[1]]]
                return winners
        elif indicator1:
                for simulation in range(howmanytimes):
                        inds1 = random.sample(range(len(strats1)-1),
size1)
                        inds2 = range(len(strats2))
```

```
                        subset1 = [strats1[i] for i in inds1] #random
strategies for each player using indices
                        subset2 = [strats2[i] for i in inds2]
                        j = 0
                        M = Setupmatrix(subset1,subset2, n, p)
                        solution = gamesolution(M)
                        #Player 1:
                        while j < len(solution[0]): #for each
proportion:
                             winners[0][inds1[j]] += solution[0][j]
                             j+=1
                        j = 0
                        #Player 2:
                        while j < len(solution[1]): #for each
proportion:
                             winners[1][inds2[j]] += solution[1][j]
                             j+=1
              winners = [[1.0*i/howmanytimes for i in winners[0]],
[1.0*i/howmanytimes for i in winners[1]]]
              return winners
     elif indicator2:
              for simulation in range(howmanytimes):
                        inds1 = range(len(strats1))
                        inds2 = random.sample(range(len(strats2)-1),
size2)
                        subset1 = [strats1[i] for i in inds1] #random
strategies for each player using indices
                        subset2 = [strats2[i] for i in inds2]
                        j = 0
                        M = Setupmatrix(subset1,subset2, n, p)
                        solution = gamesolution(M)
                        #Player 1:
                        while j < len(solution[0]): #for each
proportion:
                             winners[0][inds1[j]] += solution[0][j]
                             j+=1
                        j = 0
                        #Player 2:
                        while j < len(solution[1]): #for each
proportion:
                             winners[1][inds2[j]] += solution[1][j]
                             j+=1
              winners = [[1.0*i/howmanytimes for i in winners[0]],
[1.0*i/howmanytimes for i in winners[1]]]
              return winners
     else:
              for simulation in range(howmanytimes):
                        inds1 = range(len(strats1))
                        inds2 = range(len(strats2))
                        subset1 = [strats1[i] for i in inds1] #random
strategies for each player using indices
                        subset2 = [strats2[i] for i in inds2]
                        j = 0
                        M = Setupmatrix(subset1,subset2, n, p)
```

```
                              solution = gamesolution(M)
                              #Player 1:
                              while j < len(solution[0]): #for each
proportion:
                                    winners[0][inds1[j]] += solution[0][j]
                                    j+=1
                              j = 0
                              #Player 2:
                              while j < len(solution[1]): #for each
proportion:
                                    winners[1][inds2[j]] += solution[1][j]
                                    j+=1
                  winners = [[1.0*i/howmanytimes for i in winners[0]],
[1.0*i/howmanytimes for i in winners[1]]]
                  return winners

def Subsetfix(budget1, subsetsize1):
      return min(len(CreateAllStrategies(budget1)), subsetsize1)

def Test(utilityvector, budget1, budget2,  subsetsize1, subsetsize2,
iterations):

      n = len(utilityvector)
      y1 = budget1
      y2 = budget2
      p = utilityvector

      s1 = CreateAllStrategies(y1, n)
      s2 = CreateAllStrategies(y2, n)
      f1 = DOMINATION(s1, n)
      f2 = DOMINATION(s2, n)

      subsize1 = Subsetfix(budget1, subsetsize1)
      subsize2 = Subsetfix(budget2, subsetsize2)

      if subsize1 == subsetsize1 and subsize2 == subsetsize2:
                  temp1 = Montecarlo(f1, subsize1, f2, subsize2,
iterations, n, p, True, True)
      elif subsize1 == subsetsize1:
                  temp1 = Montecarlo(f1, subsize1, f2, len(f2), iterations,
n, p, True, False)
      elif subsize2 == subsetsize2:
                  temp1 = Montecarlo(f1, len(f1), f2, subsize2, iterations,
n, p, False, True)
      else:
                  temp1 = Montecarlo(f1, len(f1), f2, len(f2), iterations,
n, p, False, False)
      temp2 = [[],[]]
      j = 0
      while j < len(temp1[0]):
                  if temp1[0][j]>=0.01:
                              temp2[0].append([temp1[0][j],j]) #append
[prop, index]
                  j += 1
```

```
      j = 0
      while j < len(temp1[1]):
                 if temp1[1][j]>=0.01:
                            temp2[1].append([temp1[1][j],j]) #append
[prop, index]
                 j += 1
      print temp2
      return [temp2, f1, f2]


""" """ """ """ """ """ """ """ """ """ """ """ """ """ """ """ """
...................................................................
.....000000000...000000000...000000000...000000000...000000000.....
........000......000.........000............000......000...........
........000......00000.......000000000......000......000000000.....
........000......000...............000......000............000.....
........000......000000000...000000000......000......000000000.....
...................................................................
""" """ """ """ """ """ """ """ """ """ """ """ """ """ """ """ """

print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "Unweighted, floored tests:"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print ""
print "Unweighted, Floored: 4.0 vs 4.0"
start = timer()
Test([9.791598118,3.426703877,3.753353022,33.77924861], 4.0, 4.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Unweighted, Floored: 4.0 vs 4.5"
start = timer()
Test([9.791598118,3.426703877,3.753353022,33.77924861], 4.0, 4.5,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Unweighted, Floored: 4.0 vs 5.0"
start = timer()
```

```
Test([9.791598118,3.426703877,3.753353022,33.77924861], 4.0, 5.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Unweighted, Floored: 4.0 vs 5.5"
start = timer()
Test([9.791598118,3.426703877,3.753353022,33.77924861], 4.0, 5.5,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Unweighted, Floored: 4.0 vs 6.0"
start = timer()
Test([9.791598118,3.426703877,3.753353022,33.77924861], 4.0, 6.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Unweighted, Floored: 4.0 vs 6.5"
start = timer()
Test([9.791598118,3.426703877,3.753353022,33.77924861], 4.0, 6.5,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Unweighted, Floored: 4.0 vs 7.0"
start = timer()
Test([9.791598118,3.426703877,3.753353022,33.77924861], 4.0, 7.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Unweighted, Floored: 4.0 vs 8.0"
start = timer()
Test([9.791598118,3.426703877,3.753353022,33.77924861], 4.0, 8.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
```

```
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "Weighted by state, floored tests:"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print ""
print "Weighted by state, Floored: 4.0 vs 4.0"
start = timer()
Test([0.243860933, 0.282414783, 0.14780057, 3.216185277], 4.0, 4.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by state, Floored: 4.0 vs 4.5"
start = timer()
Test([0.243860933, 0.282414783, 0.14780057, 3.216185277], 4.0, 4.5,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by state, Floored: 4.0 vs 5.0"
start = timer()
Test([0.243860933, 0.282414783, 0.14780057, 3.216185277], 4.0, 5.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by state, Floored: 4.0 vs 5.5"
start = timer()
Test([0.243860933, 0.282414783, 0.14780057, 3.216185277], 4.0, 5.5,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
```

```
print ""
print ""
print "Weighted by state, Floored: 4.0 vs 6.0"
start = timer()
Test([0.243860933, 0.282414783, 0.14780057, 3.216185277], 4.0, 6.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by state, Floored: 4.0 vs 6.5"
start = timer()
Test([0.243860933, 0.282414783, 0.14780057, 3.216185277], 4.0, 6.5,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by state, Floored: 4.0 vs 7.0"
start = timer()
Test([0.243860933, 0.282414783, 0.14780057, 3.216185277], 4.0, 7.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by state, Floored: 4.0 vs 8.0"
start = timer()
Test([0.243860933, 0.282414783, 0.14780057, 3.216185277], 4.0, 8.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "Weighted by sample, floored tests:"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print ""
```

```
print "Weighted by sample, Floored: 4.0 vs 4.0"
start = timer()
Test([0.167306442, 0.089473901, 0.11420262, 2.841187593], 4.0, 4.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by sample, Floored: 4.0 vs 4.5"
start = timer()
Test([0.167306442, 0.089473901, 0.11420262, 2.841187593], 4.0, 4.5,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by sample, Floored: 4.0 vs 5.0"
start = timer()
Test([0.167306442, 0.089473901, 0.11420262, 2.841187593], 4.0, 5.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by sample, Floored: 4.0 vs 5.5"
start = timer()
Test([0.167306442, 0.089473901, 0.11420262, 2.841187593], 4.0, 5.5,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by sample, Floored: 4.0 vs 6.0"
start = timer()
Test([0.167306442, 0.089473901, 0.11420262, 2.841187593], 4.0, 6.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by sample, Floored: 4.0 vs 6.5"
start = timer()
Test([0.167306442, 0.089473901, 0.11420262, 2.841187593], 4.0, 6.5,  100,
100, 100000)
```

```
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by sample, Floored: 4.0 vs 7.0"
start = timer()
Test([0.167306442, 0.089473901, 0.11420262, 2.841187593], 4.0, 7.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by sample, Floored: 4.0 vs 8.0"
start = timer()
Test([0.167306442, 0.089473901, 0.11420262, 2.841187593], 4.0, 8.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "Weighted by sample, unfloored tests:"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print ""
print "Weighted by sample, Unfloored: 4.0 vs 4.0"
start = timer()
Test([0.26559308, 0.242170061, 0.124087002, -0.419994208], 4.0, 4.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by sample, Unfloored: 4.0 vs 4.5"
start = timer()
Test([0.26559308, 0.242170061, 0.124087002, -0.419994208], 4.0, 4.5,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
```

```
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by sample, Unfloored: 4.0 vs 5.0"
start = timer()
Test([0.26559308, 0.242170061, 0.124087002, -0.419994208], 4.0, 5.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by sample, Unfloored: 4.0 vs 5.5"
start = timer()
Test([0.26559308, 0.242170061, 0.124087002, -0.419994208], 4.0, 5.5,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by sample, Unfloored: 4.0 vs 6.0"
start = timer()
Test([0.26559308, 0.242170061, 0.124087002, -0.419994208], 4.0, 6.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by sample, Unfloored: 4.0 vs 6.5"
start = timer()
Test([0.26559308, 0.242170061, 0.124087002, -0.419994208], 4.0, 6.5,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by sample, Unfloored: 4.0 vs 7.0"
start = timer()
Test([0.26559308, 0.242170061, 0.124087002, -0.419994208], 4.0, 7.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
```

```
print "Weighted by sample, Unfloored: 4.0 vs 8.0"
start = timer()
Test([0.26559308, 0.242170061, 0.124087002, -0.419994208], 4.0, 8.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "Unweighted, unfloored tests"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print ""
print "Unweighted, Unfloored: 4.0 vs 4.0"
start = timer()
Test([6.30872096, 2.549945091, 0.401012477, -1.653903518], 4.0, 4.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Unweighted, Unfloored: 4.0 vs 4.5"
start = timer()
Test([6.30872096, 2.549945091, 0.401012477, -1.653903518], 4.0, 4.5,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Unweighted, Unfloored: 4.0 vs 5.0"
start = timer()
Test([6.30872096, 2.549945091, 0.401012477, -1.653903518], 4.0, 5.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Unweighted, Unfloored: 4.0 vs 5.5"
start = timer()
```

```
Test([6.30872096, 2.549945091, 0.401012477, -1.653903518], 4.0, 5.5,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Unweighted, Unfloored: 4.0 vs 6.0"
start = timer()
Test([6.30872096, 2.549945091, 0.401012477, -1.653903518], 4.0, 6.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Unweighted, Unfloored: 4.0 vs 6.5"
start = timer()
Test([6.30872096, 2.549945091, 0.401012477, -1.653903518], 4.0, 6.5,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Unweighted, Unfloored: 4.0 vs 7.0"
start = timer()
Test([6.30872096, 2.549945091, 0.401012477, -1.653903518], 4.0, 7.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "Unweighted, Unfloored: 4.0 vs 8.0"
start = timer()
Test([6.30872096, 2.549945091, 0.401012477, -1.653903518], 4.0, 8.0,  100,
100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - - "
print ""
print ""
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "Weighted by state, unfloored tests:"
```

```
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print "~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~"
print ""
print "Weighted by state, Unfloored: 4.0 vs 4.0"
start = timer()
Test([0.353833059, -0.460278059, 0.204454969, -0.480740837], 4.0, 4.0,
100, 100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by state, Unfloored: 4.0 vs 4.5"
start = timer()
Test([0.353833059, -0.460278059, 0.204454969, -0.480740837], 4.0, 4.5,
100, 100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by state, Unfloored: 4.0 vs 5.0"
start = timer()
Test([0.353833059, -0.460278059, 0.204454969, -0.480740837], 4.0, 5.0,
100, 100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by state, Unfloored: 4.0 vs 5.5"
start = timer()
Test([0.353833059, -0.460278059, 0.204454969, -0.480740837], 4.0, 5.5,
100, 100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by state, Unfloored: 4.0 vs 6.0"
start = timer()
Test([0.353833059, -0.460278059, 0.204454969, -0.480740837], 4.0, 6.0,
100, 100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
```

```
print ""
print ""
print "Weighted by state, Unfloored: 4.0 vs 6.5"
start = timer()
Test([0.353833059, -0.460278059, 0.204454969, -0.480740837], 4.0, 6.5,
100, 100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by state, Unfloored: 4.0 vs 7.0"
start = timer()
Test([0.353833059, -0.460278059, 0.204454969, -0.480740837], 4.0, 7.0,
100, 100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
print "Weighted by state, Unfloored: 4.0 vs 8.0"
start = timer()
Test([0.353833059, -0.460278059, 0.204454969, -0.480740837], 4.0, 8.0,
100, 100, 100000)
end = timer()
print(end - start),"seconds"
print "- - - - - - - - - - - - - -"
print " - - - - - - - - - - - - - "
print ""
print ""
```