

61A Project 1 Contest Results

Friday, November 18

A Long Time Ago in a Project Far, Far Away

A Long Time Ago in a Project Far, Far Away

Pig Contest Rules:

A Long Time Ago in a Project Far, Far Away

Pig Contest Rules:

- The score for an entry is the sum of win rates against every other entry

A Long Time Ago in a Project Far, Far Away

Pig Contest Rules:

- The score for an entry is the sum of win rates against every other entry
- All strategies must be deterministic, pure functions of the current player scores

A Long Time Ago in a Project Far, Far Away

Pig Contest Rules:

- The score for an entry is the sum of win rates against every other entry
- All strategies must be deterministic, pure functions of the current player scores
- Winning entries will receive (a little) extra credit

A Long Time Ago in a Project Far, Far Away

Pig Contest Rules:

- The score for an entry is the sum of win rates against every other entry
- All strategies must be deterministic, pure functions of the current player scores
- Winning entries will receive (a little) extra credit



<http://images.clipartof.com/thumbnails/1049431-Royalty-Free-RF-Clip-Art-Illustration-Of-3d-Gold-Mosaic-Dice-Rolling.jpg>

A Long Time Ago in a Project Far, Far Away

Pig Contest Rules:

- The score for an entry is the sum of win rates against every other entry
- All strategies must be deterministic, pure functions of the current player scores
- Winning entries will receive (a little) extra credit



Pig Rules:

<http://images.clipartof.com/thumbnails/1049431-Royalty-Free-RF-Clip-Art-Illustration-Of-3d-Gold-Mosaic-Dice-Rolling.jpg>

A Long Time Ago in a Project Far, Far Away

Pig Contest Rules:

- The score for an entry is the sum of win rates against every other entry
- All strategies must be deterministic, pure functions of the current player scores
- Winning entries will receive (a little) extra credit



Pig Rules:

- Roll a 6-sided die until you either *hold* or *roll a 1* (pig)

<http://images.clipartof.com/thumbnails/1049431-Royalty-Free-RF-Clip-Art-Illustration-Of-3d-Gold-Mosaic-Dice-Rolling.jpg>

A Long Time Ago in a Project Far, Far Away

Pig Contest Rules:

- The score for an entry is the sum of win rates against every other entry
- All strategies must be deterministic, pure functions of the current player scores
- Winning entries will receive (a little) extra credit



Pig Rules:

- Roll a 6-sided die until you either *hold* or *roll a 1* (pig)
- First person to reach 100 points wins

<http://images.clipartof.com/thumbnails/1049431-Royalty-Free-RF-Clip-Art-Illustration-Of-3d-Gold-Mosaic-Dice-Rolling.jpg>

A Long Time Ago in a Project Far, Far Away

Pig Contest Rules:

- The score for an entry is the sum of win rates against every other entry
- All strategies must be deterministic, pure functions of the current player scores
- Winning entries will receive (a little) extra credit



Pig Rules:

- Roll a 6-sided die until you either *hold* or *roll a 1* (pig)
- First person to reach 100 points wins
- 61A Variant: When scores sum to 7, roll a 4-sided die instead

<http://images.clipartof.com/thumbnails/1049431-Royalty-Free-RF-Clip-Art-Illustration-Of-3d-Gold-Mosaic-Dice-Rolling.jpg>

Effective Strategy Ideas

Effective Strategy Ideas

Use the techniques described in the project

Effective Strategy Ideas

Use the techniques described in the project

- Quit while you're ahead (don't roll when you've already won)

Effective Strategy Ideas

Use the techniques described in the project

- Quit while you're ahead (don't roll when you've already won)
- Stop earlier using a 4-sided die

Effective Strategy Ideas

Use the techniques described in the project

- Quit while you're ahead (don't roll when you've already won)
- Stop earlier using a 4-sided die
- Take more risks when you're losing

Effective Strategy Ideas

Use the techniques described in the project

- Quit while you're ahead (don't roll when you've already won)
- Stop earlier using a 4-sided die
- Take more risks when you're losing

Force your opponent to roll a 4-sided die

Effective Strategy Ideas

Use the techniques described in the project

- Quit while you're ahead (don't roll when you've already won)
- Stop earlier using a 4-sided die
- Take more risks when you're losing

Force your opponent to roll a 4-sided die

- You know when holding would give them a 4-sided die

Effective Strategy Ideas

Use the techniques described in the project

- Quit while you're ahead (don't roll when you've already won)
- Stop earlier using a 4-sided die
- Take more risks when you're losing

Force your opponent to roll a 4-sided die

- You know when holding would give them a 4-sided die
- It's your turn! Your score: 14, Their score: 20, Turn: 15

Effective Strategy Ideas

Use the techniques described in the project

- Quit while you're ahead (don't roll when you've already won)
- Stop earlier using a 4-sided die
- Take more risks when you're losing

Force your opponent to roll a 4-sided die

- You know when holding would give them a 4-sided die
- It's your turn! Your score: 14, Their score: 20, Turn: 15

Combine strategies together

Effective Strategy Ideas

Use the techniques described in the project

- Quit while you're ahead (don't roll when you've already won)
- Stop earlier using a 4-sided die
- Take more risks when you're losing

Force your opponent to roll a 4-sided die

- You know when holding would give them a 4-sided die
- It's your turn! Your score: 14, Their score: 20, Turn: 15

Combine strategies together

- Each idea can be parameterized by a few constants

Effective Strategy Ideas

Use the techniques described in the project

- Quit while you're ahead (don't roll when you've already won)
- Stop earlier using a 4-sided die
- Take more risks when you're losing

Force your opponent to roll a 4-sided die

- You know when holding would give them a 4-sided die
- It's your turn! Your score: 14, Their score: 20, Turn: 15

Combine strategies together

- Each idea can be parameterized by a few constants
- Finding just the right set of constants can help a lot

Computing Win Rates Exactly

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

(me,0,0,0): roll

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

(me,0,0,0): roll

(me,0,0,2): roll

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

(me,0,0,0): roll

(me,0,0,2): roll

...

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

(me,0,0,0): roll

(me,0,0,2): roll

...

(me,0,0,18): roll

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
```

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

```
(me,0,0,0): roll  
(me,0,0,2): roll  
...  
(me,0,0,18): roll  
(me,0,0,19): roll  
(me,0,0,20): hold
```

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
```


Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
```

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
```

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
```

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
```

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

(me,99,99,0)

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

```
(me,99,99,0)      1
```


Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

(you,97,99,0)

(me,99,99,0)

1

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

```
(you,97,99,0)      0
(me,99,99,0)      1
```

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

```
(me,96,99,4+)
(you,97,99,0)      0
(me,99,99,0)      1
```

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

```
(me,96,99,4+)      1
(you,97,99,0)     0
(me,99,99,0)      1
```

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

```
(me,96,99,3)
(me,96,99,4+)      1
(you,97,99,0)      0
(me,99,99,0)       1
```

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

```
(me,96,99,3)
(me,96,99,4+) 1
(you,97,99,0) 0
(me,99,99,0) 1
```

Computing Win Rates Exactly

A *state* in the game:

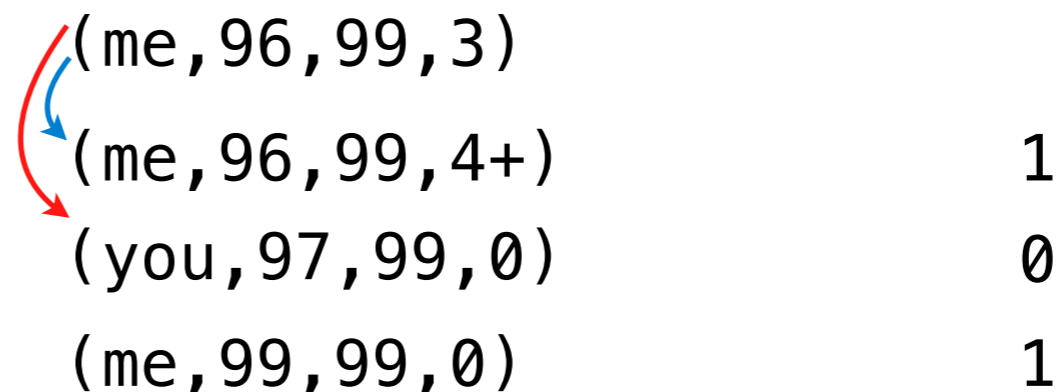
(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

```
(me,96,99,3)
(me,96,99,4+) 1
(you,97,99,0) 0
(me,99,99,0) 1
```



Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

```
(me,96,99,3)           5/6
  ↙
(me,96,99,4+)         1
  ↘
(you,97,99,0)         0
(me,99,99,0)          1
```


Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

Each state has a chance to win

```
(me,96,99,2)
(me,96,99,3)      5/6
(me,96,99,4+)    1
(you,97,99,0)    0
(me,99,99,0)     1
```

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

```
(me,96,99,2)
(me,96,99,3)      5/6
(me,96,99,4+)    1
(you,97,99,0)    0
(me,99,99,0)     1
```

Computing Win Rates Exactly

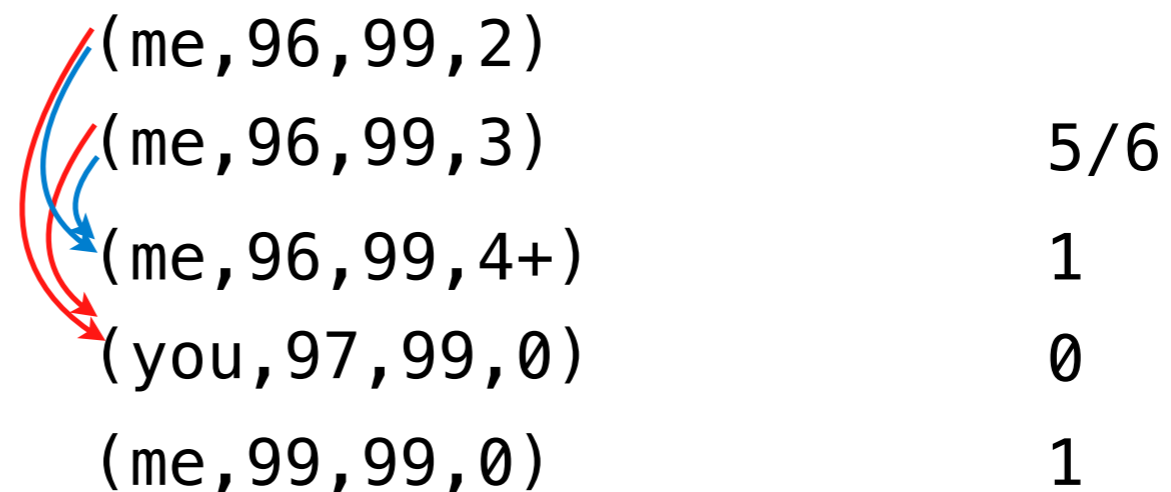
A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```



Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

(me,96,99,2)	5/6
(me,96,99,3)	5/6
(me,96,99,4+)	1
(you,97,99,0)	0
(me,99,99,0)	1

Computing Win Rates Exactly

A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

Each state has a chance to win

```
...
(me,96,99,0)
...
(me,96,99,2)           5/6
(me,96,99,3)           5/6
(me,96,99,4+)         1
(you,97,99,0)         0
(me,99,99,0)          1
```

Computing Win Rates Exactly

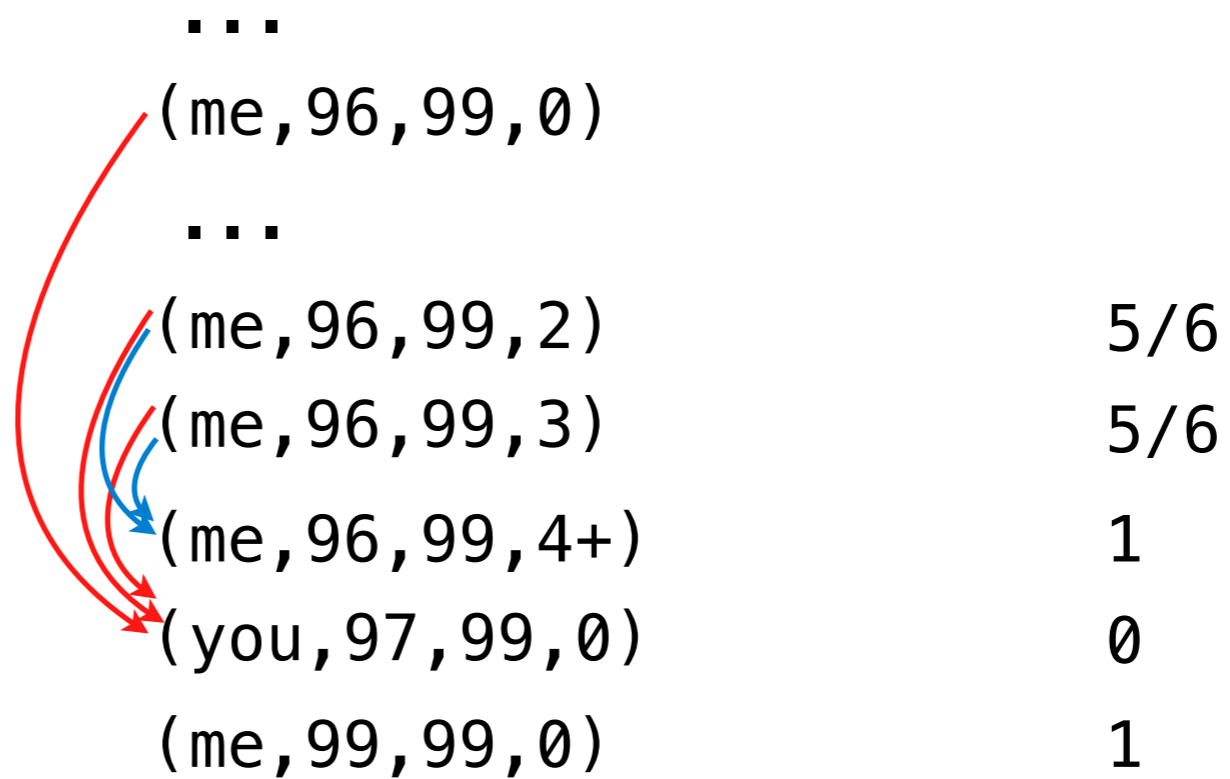
A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```



Computing Win Rates Exactly

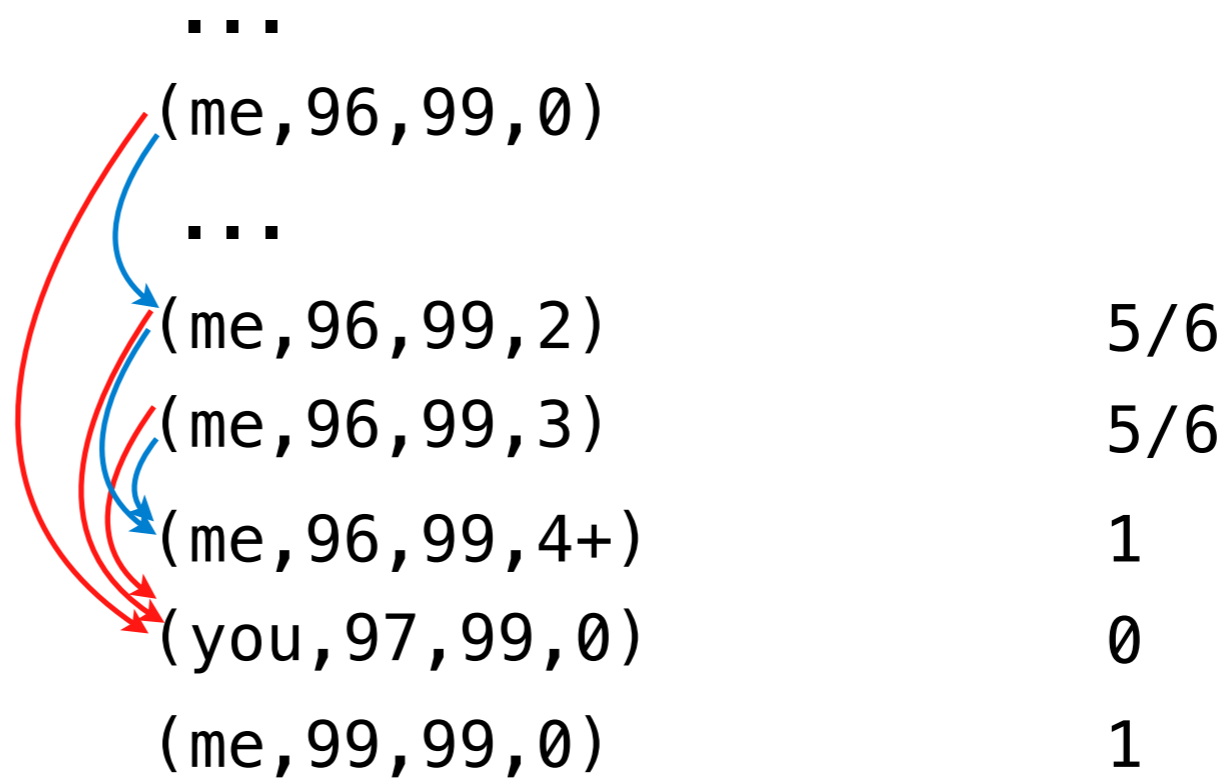
A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```



Computing Win Rates Exactly

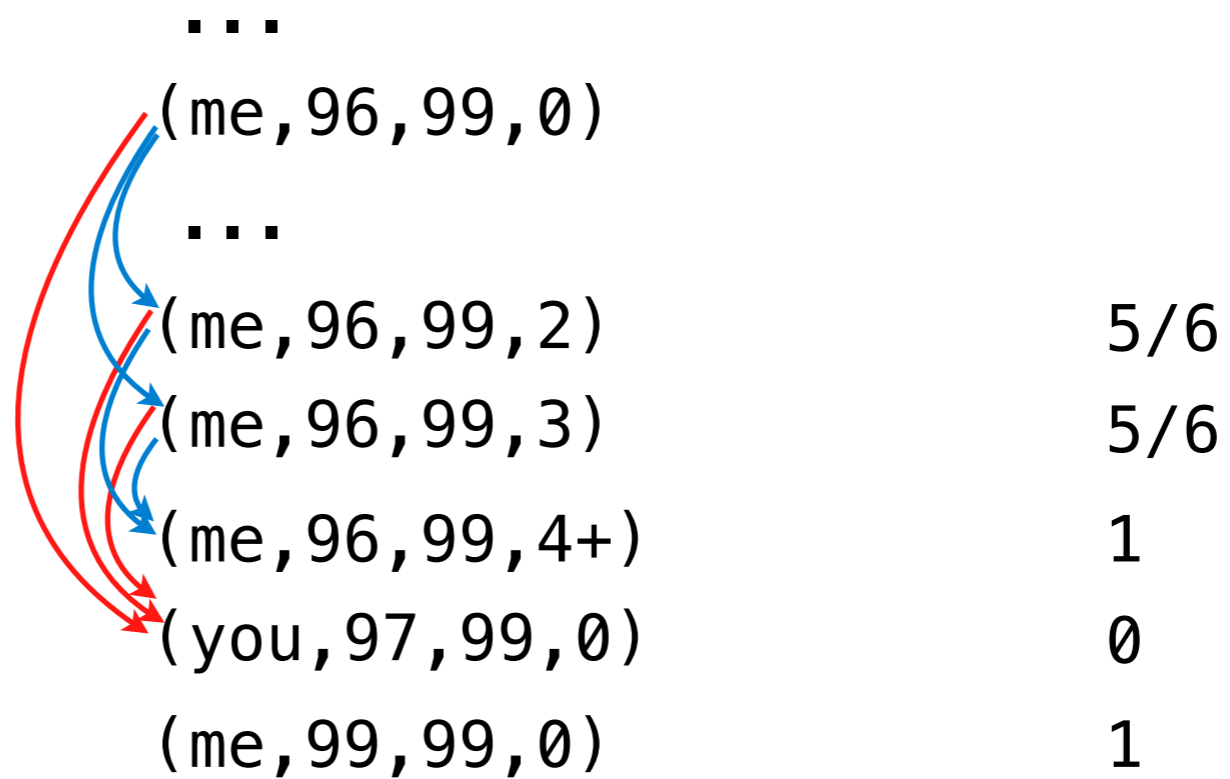
A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```

Each state has a chance to win



Computing Win Rates Exactly

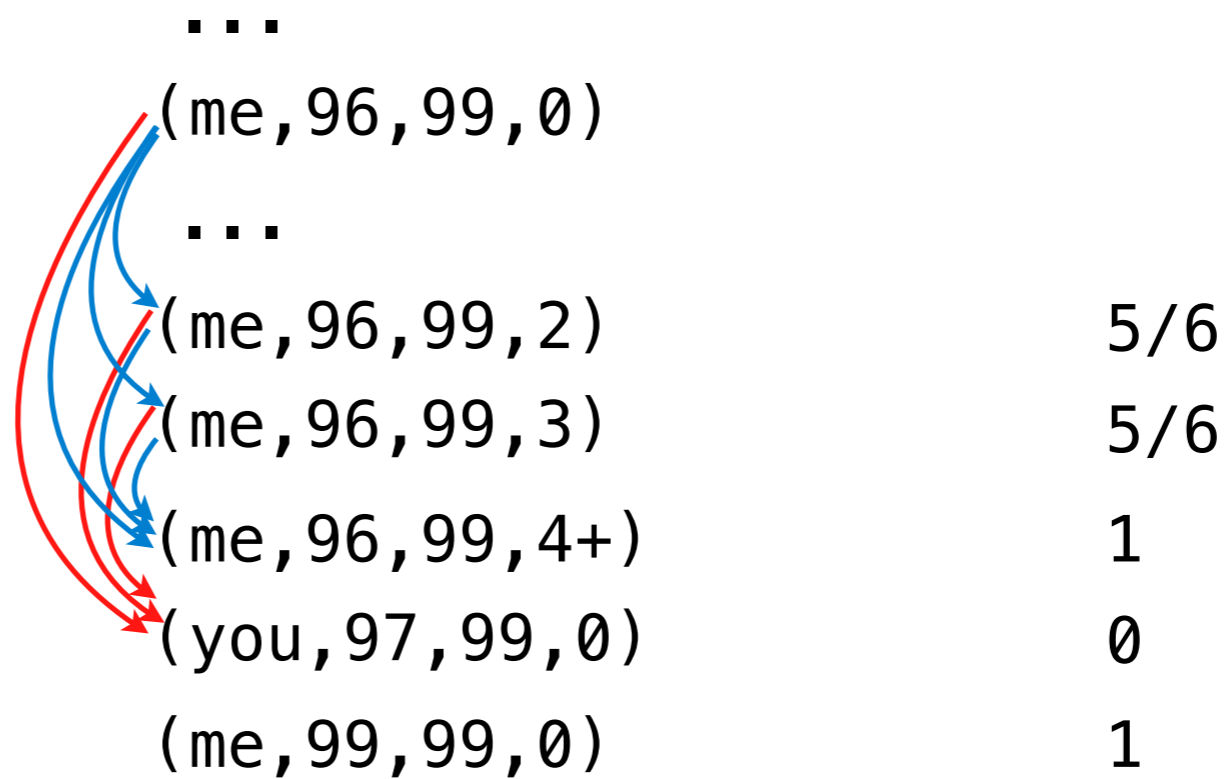
A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll



Computing Win Rates Exactly

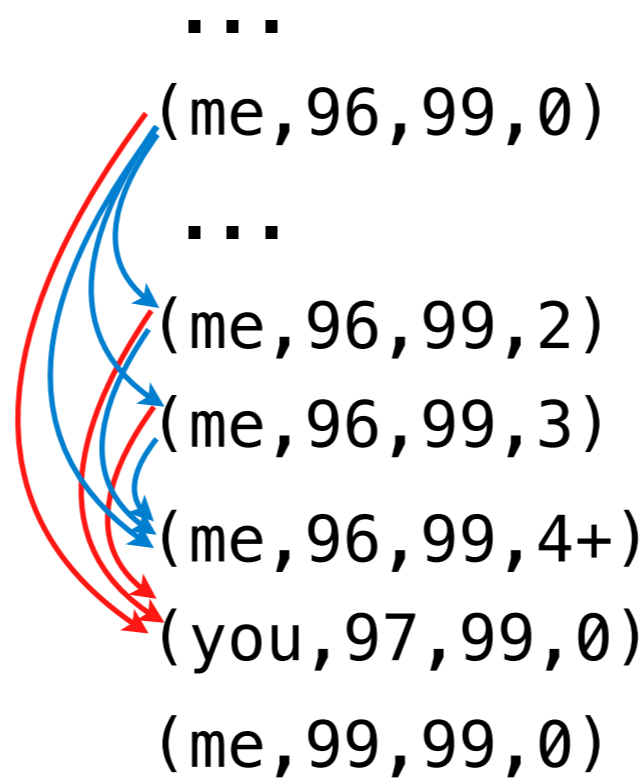
A *state* in the game:

(who rolls next?, player score, opponent score, turn total)

A strategy is a table

Each state has a chance to win

```
(me,0,0,0): roll
(me,0,0,2): roll
...
(me,0,0,18): roll
(me,0,0,19): roll
(me,0,0,20): hold
(me,0,0,21): hold
...
(me,96,99,2): roll
(me,96,99,4): hold
...
(me,99,99,0): roll
```



$$\frac{1}{6} \cdot 0 + \frac{2}{6} \cdot \frac{5}{6} + \frac{3}{6} \cdot 1$$

5/6

5/6

1

0

1

Top Finishers

Top Finishers

5th: Stephen Chen & Winne Yan

53.3%

Top Finishers

5th: Stephen Chen & Winne Yan 53.3%

4th: Michelle Nguyen & WeiJia Jin 53.6%

Top Finishers

5th: Stephen Chen & Winne Yan 53.3%

4th: Michelle Nguyen & WeiJia Jin 53.6%

3rd: Siqi Wu & WeiChih Wang 53.8%

Top Finishers

5th: Stephen Chen & Winne Yan	53.3%
4th: Michelle Nguyen & WeiJia Jin	53.6%
3rd: Siqi Wu & WeiChih Wang	53.8%
2nd: Wonjohn Choi	55.5%

Top Finishers

5th: Stephen Chen & Winne Yan	53.3%
4th: Michelle Nguyen & WeiJia Jin	53.6%
3rd: Siqi Wu & WeiChih Wang	53.8%
2nd: Wonjohn Choi	55.5%
1st: Keegan Mann	55.7%
Yan Duan & Ziming Li	55.7%
Brian Prike & Zhenghao Qian	55.7%
Parker Schuh & Robert Chatham	55.7%

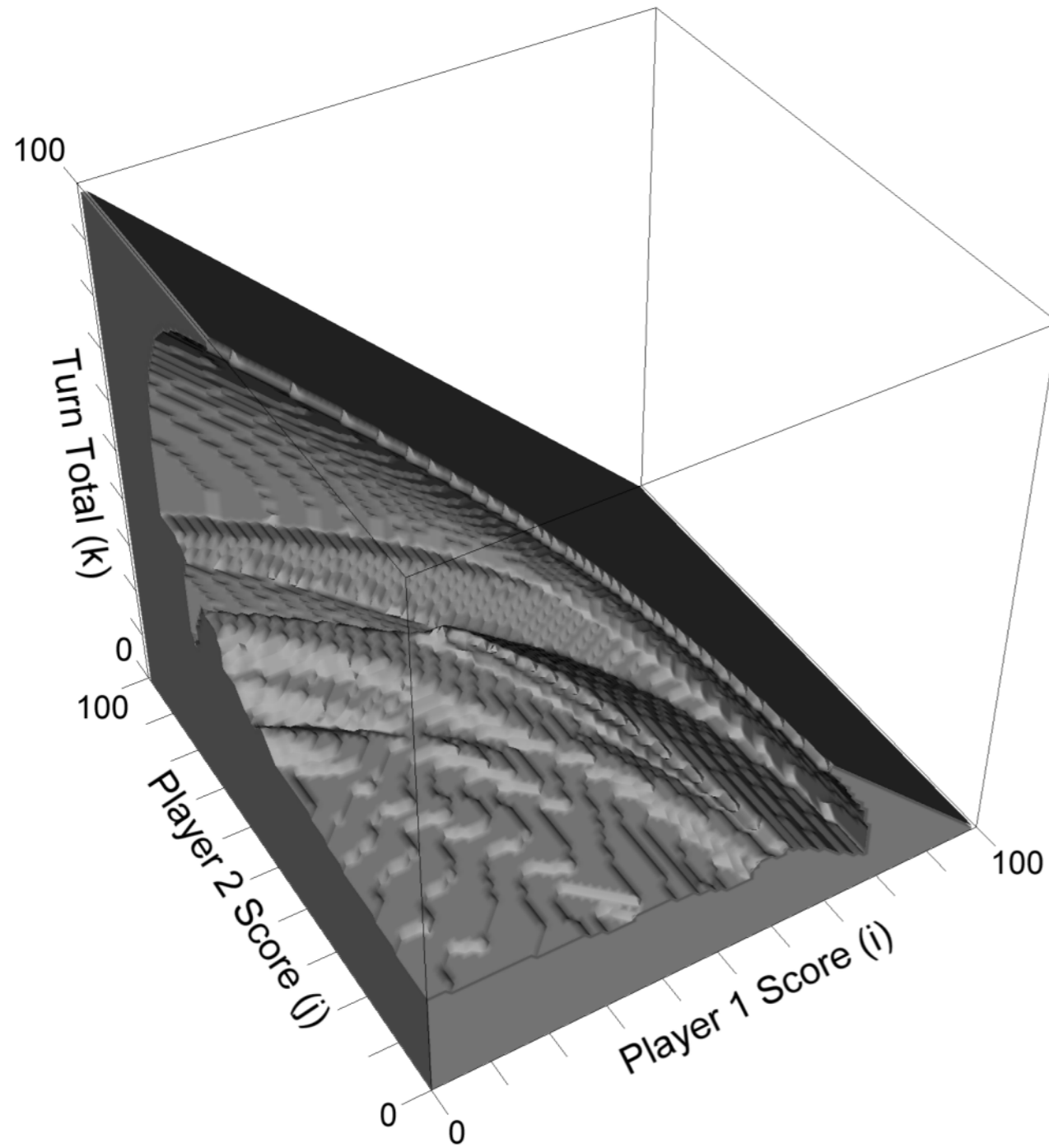
Top Finishers

5th: Stephen Chen & Winne Yan	53.3%
4th: Michelle Nguyen & WeiJia Jin	53.6%
3rd: Siqi Wu & WeiChih Wang	53.8%
2nd: Wonjohn Choi	55.5%
1st: Keegan Mann	55.7%
Yan Duan & Ziming Li	55.7%
Brian Prike & Zhenghao Qian	55.7%
Parker Schuh & Robert Chatham	55.7%

Lots of entries were awesome! Great work.

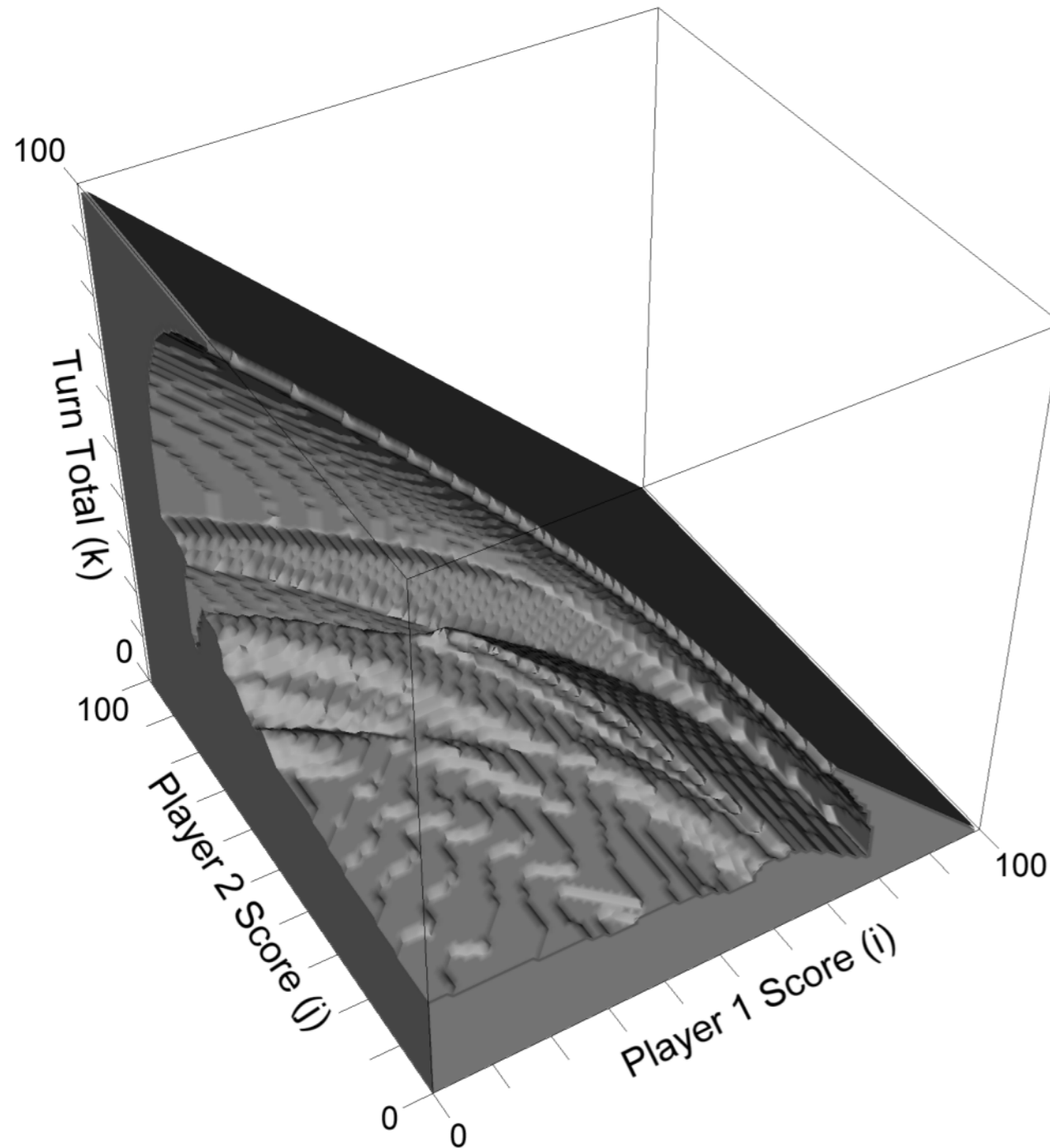
Visualizing Optimal Play in Pig

Visualizing Optimal Play in Pig



Visualizing Optimal Play in Pig

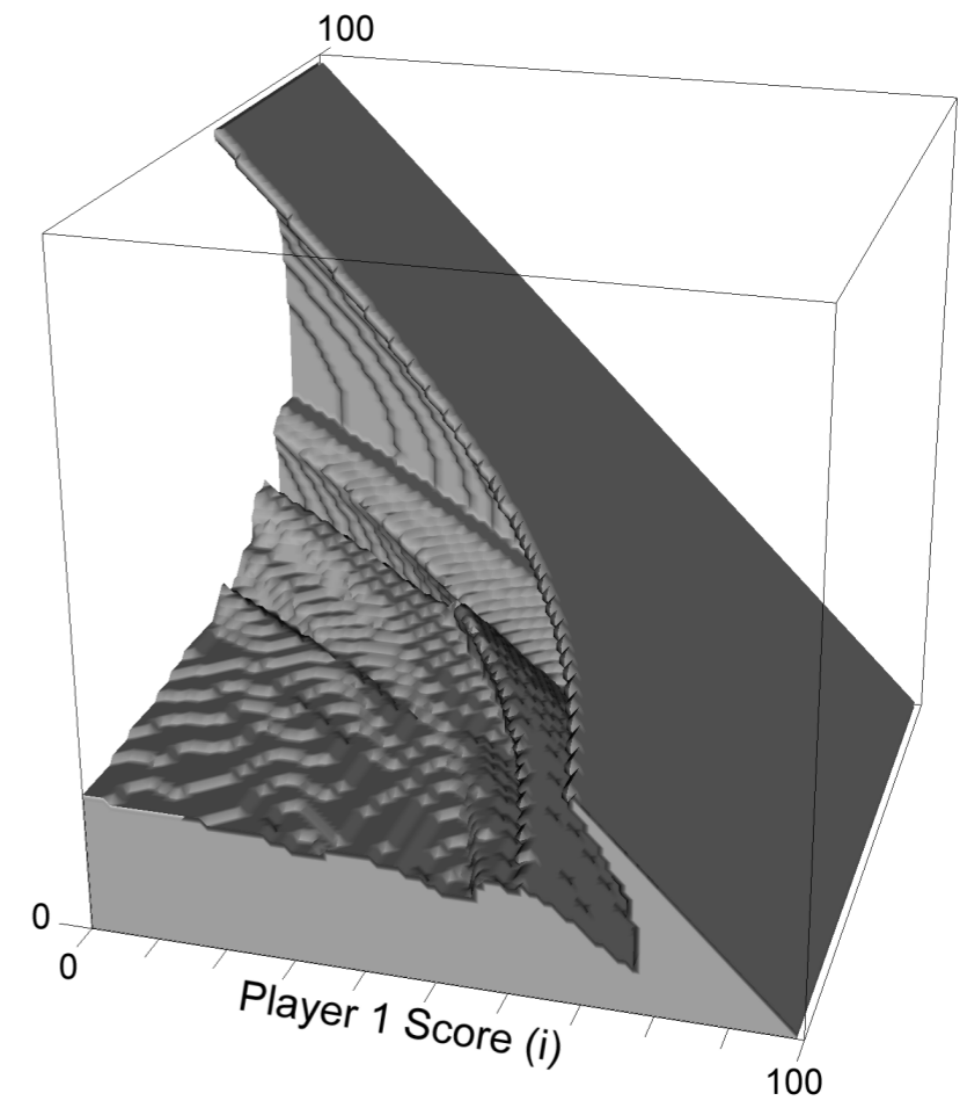
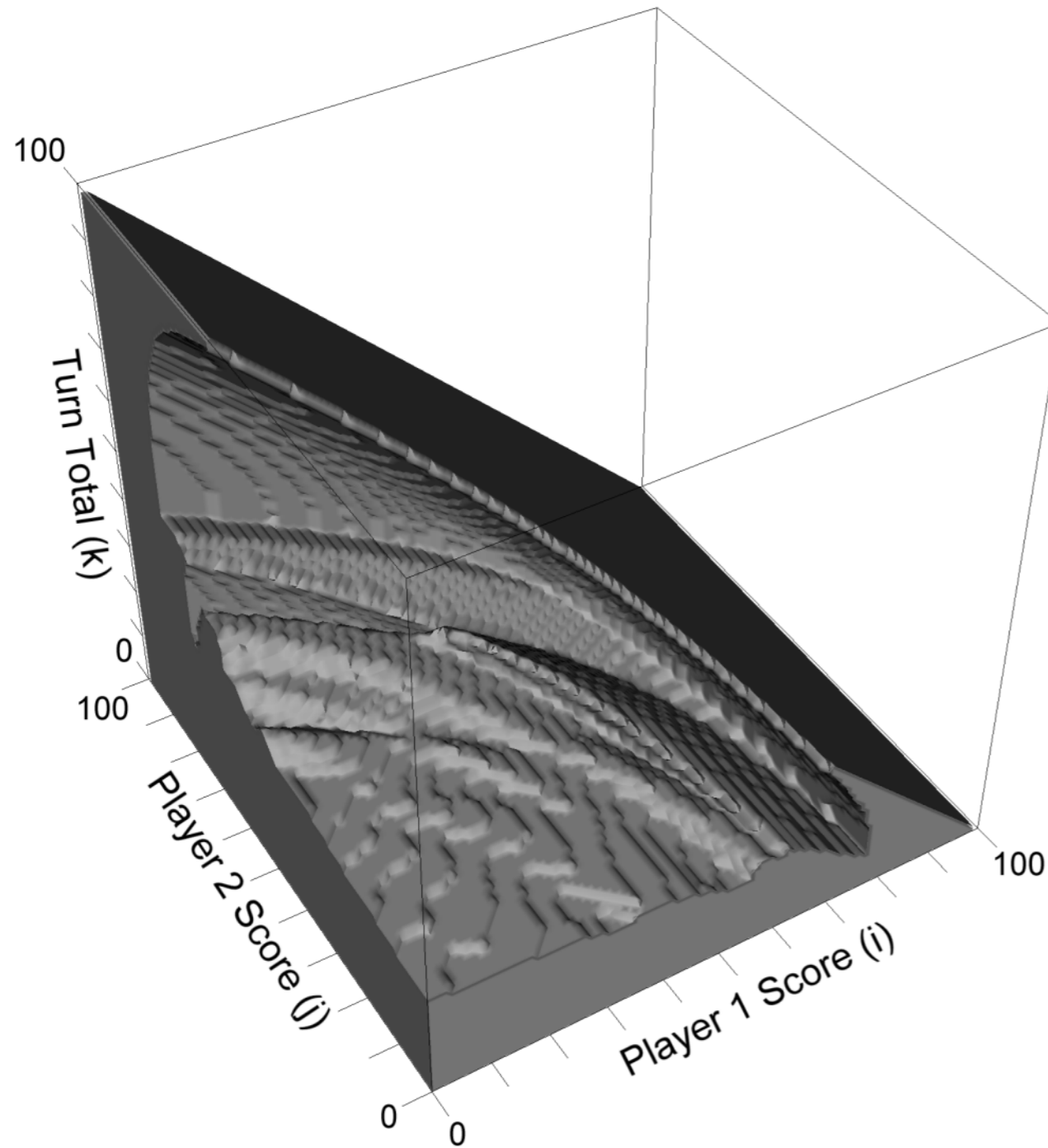
The roll/hold boundary
for optimal Pig play



Todd W. Neller and Clifton G.M. Presser. [Optimal Play of the Dice Game Pig](#), *The UMAP Journal* 25(1) (2004), pp. 25-47

Visualizing Optimal Play in Pig

The roll/hold boundary for optimal Pig play



Todd W. Neller and Clifton G.M. Presser. [Optimal Play of the Dice Game Pig](#), *The UMAP Journal* 25(1) (2004), pp. 25-47