Optimal Play of the All Yellow Zombie Dice Game

Todd W. Neller, John C. Llano, Minh Q. Vu Dinh, and Clifton G.M. Presser

Overview

- Rules
- Optimality Equations
- Solution Method
- Optimal Play Visualization
- Human-Playable Policies
  - Fixed hold-at
  - Minh cases
  - Llano cases
  - Neller cases
- Conclusions

Zombie Dice

- Dice game for 2 or more players. (Here we consider 2 player only.)
- First published in 2010 by Steve Jackson
  - Jeopardy dice game in the Ten Thousand dice game family, e.g. Farkle, Cosmic Wimpout
- Jeopardy Dice Game – Primary mechanic: Roll/hold decisions where holding *secures* turn progress, whereas rolling *risks* all turn progress for potentially greater turn progress. “Push your luck.”

Zombie Dice Rules

Color	Number of Dice	Brain Sides	Shotgun Sides	Footprint Sides
Green	6	3	1	2
Yellow	4	2	2	2
Red	3	1	3	2

- 2 or more players using 13 non-standard (d6) dice (distribution above)
- Players will have the same number of turns. A turn consists of a sequence of player rolls 3 dice are drawn as random and rolled, and rolled shotguns and brains are set aside. Footprint rolls are included in the next 3 dice if the player continues rolling.
- The turn ends when either the player
  - decides to hold (i.e. stop rolling) and score the total number of brains rolled, or
  - has rolled three or more shotguns, ending the turn with no score change.
- A round consists of each player taking one turn in sequence.
- Any player ending their turn with a goal score of 13 or more causes that to be the last round of the game, unless tie(s) triggers additional tiebreaker round(s).
- At the end of the last round, the player with the highest score wins.

Zombie Dice Example Round

Player	Dice Drawn	Dice Rolled	Result [Decision]
1	G, G, Y	F, F, S	One shotgun set aside, two G retained for reroll [**roll**] S
1	G	B, S, F	One brain set aside, one shotgun set aside (2 total), one G retained for reroll [**roll**] B, S, S
1	R, R	B, S, S	One brain set aside, two shotguns set aside (4 total), ≥ three shotguns → **turn ends** with no score gain B, B, S, S, S, S
2	G, G, G	B, B, F	Two brains set aside, one G retained for reroll [**roll**] B, B
2	G, Y	B, B, F	Two brains set aside (4 total), one G retained for reroll [**roll**] B, B, B, B
2	Y, R	B, S, S	One brain set aside (5 total), two shotguns set aside (2 total) [**hold**] → **turn ends** with a score gain of 5 B, B, B, B, B, S, S

All Yellow Zombie Dice

- Optimal play for *maximizing expected turn score* is known for Zombie Dice. Optimal play for *maximizing expected win probability* is **unknown** for Zombie Dice.
- We here compute optimal winning play for All Yellow Zombie Dice (AYZD) where all dice are yellow.

Nonterminal states are described as the 5-tuple (p, i, j, b, s), where p is the current player number (1 or 2), i is the current player score, j is the opponent score, b is the turn total (number of brains set aside), and s is the number of rolled shotguns set aside. $P(p, i, j, b, s)$ will denote the probability of player p winning in state (p, i, j, b, s) under the assumption of optimal play, i.e. each player plays so as to maximize one's own expected win probability.

Probabilities of Roll Outcomes

Let $P_{\text{roll}}(b, s)$ be the probability of rolling b brains and s shotguns (and thus $3 - b - s$ footprints) on a roll of 3 dice:

$$P_{\text{roll}}(b, s) = \frac{\binom{3}{b}\binom{3-b}{s}}{3^3} = \frac{2}{9b!s!(3-b-s)!}$$

Probability of Winning with a Roll

The probability of winning with a roll $P_{\text{roll}}(p, i, j, b, s)$ under the assumption of optimal play thereafter is:

$$P_{\text{roll}}(p, i, j, b, s) = \sum_{s^+=0}^{2-s} \sum_{b^+=0}^{3-s^+} (P_{\text{roll}}(b^+, s^+) P(p, i, j, b + b^+, s + s^+)) \\ + \sum_{s^+=3-s}^{3} \sum_{b^+=0}^{3-s^+} (P_{\text{roll}}(b^+, s^+)(1 - P(3 - p, j, i, 0, 0)))$$

where b^+ and s^+ denote the number of additional brains and shotguns rolled.

Probability of Winning with a Hold

The probability of winning with a hold $P_{\text{hold}}(p, i, j, b, s)$ under the assumption of optimal play thereafter is:

$$P_{\text{hold}}(p, i, j, b, s) = (1 - P(3 - p, j, i + b, 0, 0))$$

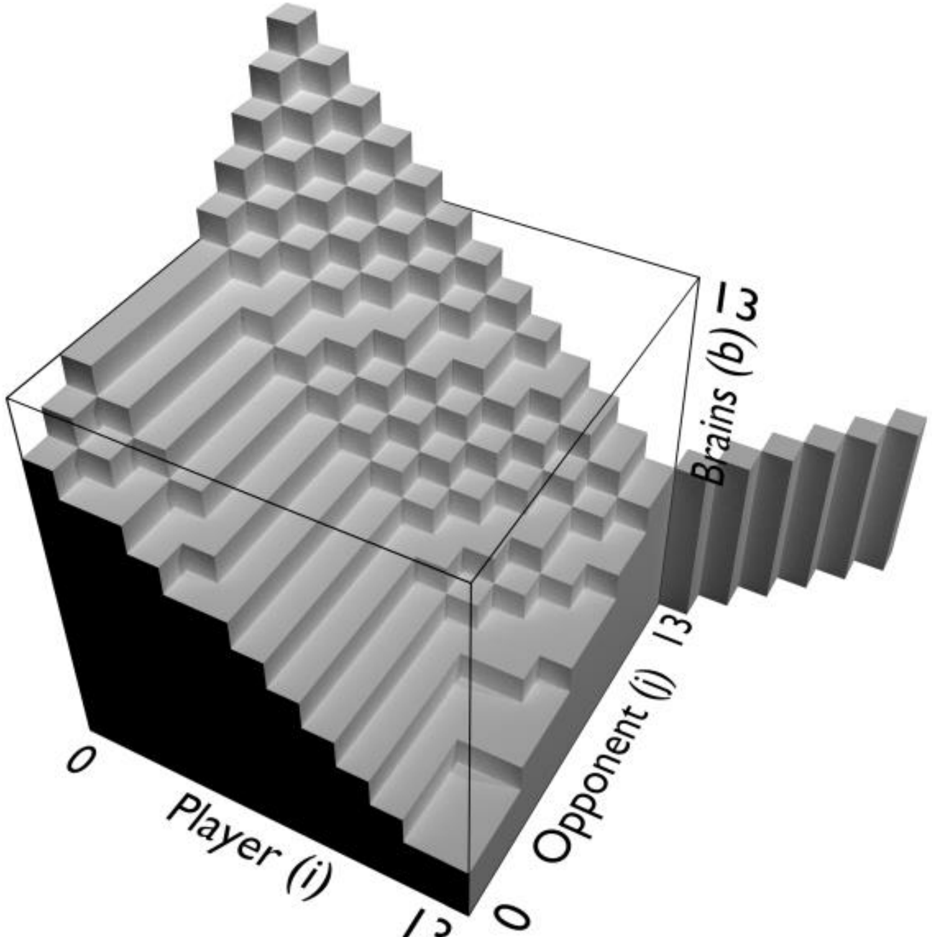
Then the probability of winning $P(p, i, j, b, s)$ under the assumption of optimal play is:

$$P(p, i, j, b, s) = \max(P_{\text{roll}}(p, i, j, b, s), P_{\text{hold}}(p, i, j, b, s))$$

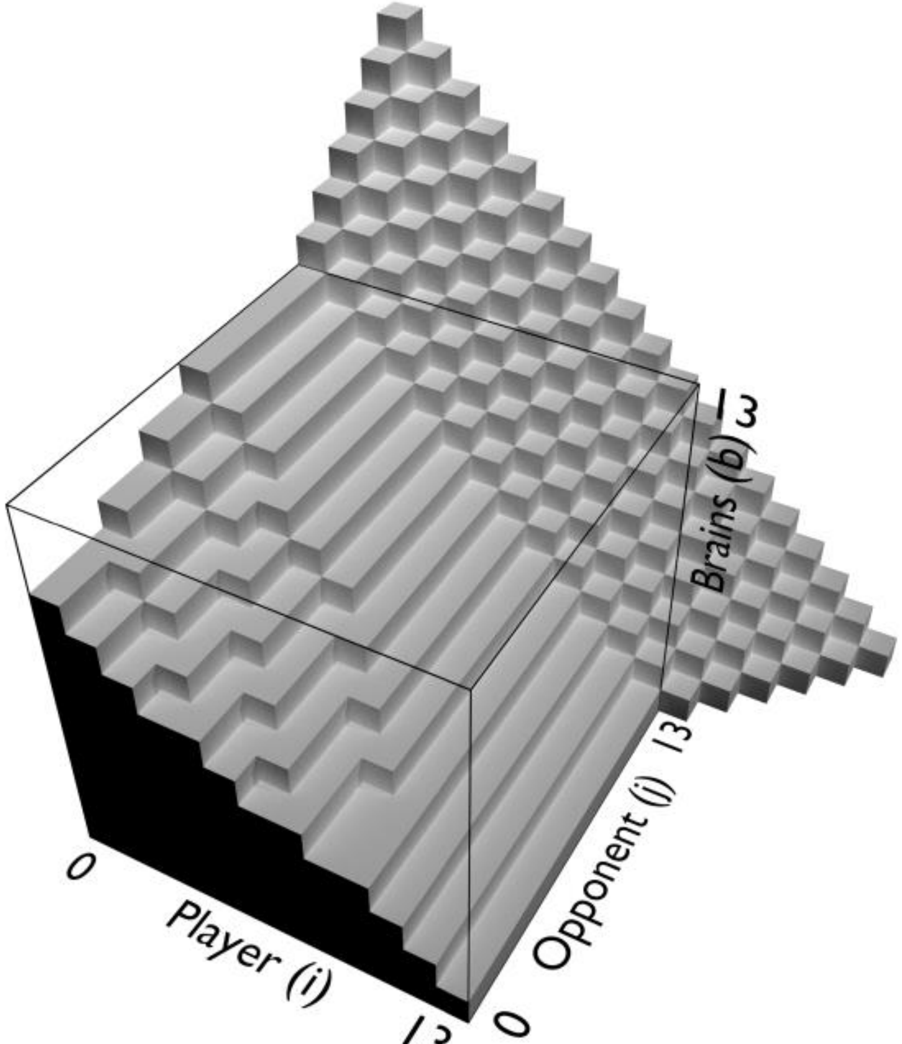
Solving Optimality Equations

- $P_{roll}(b,s)$ is precomputed.
- Cyclic, recursive $P(p,i,j,b,s)$ is solved through a variation of Value Iteration:
  - From initial arbitrary P estimates, substitute estimates in equation right-hand sides.
  - Compute the left-hand side P values as new, better estimates.
  - Terminate iterations of previous steps when the maximum change to a P estimate is $\leq 1\times10^{-14}$.
- Since possible scores are unbounded, we created a high artificial upper scoring bound and verified that probabilities and policies were unchanged for higher bounds.

Optimal Play (zero shotguns set aside)

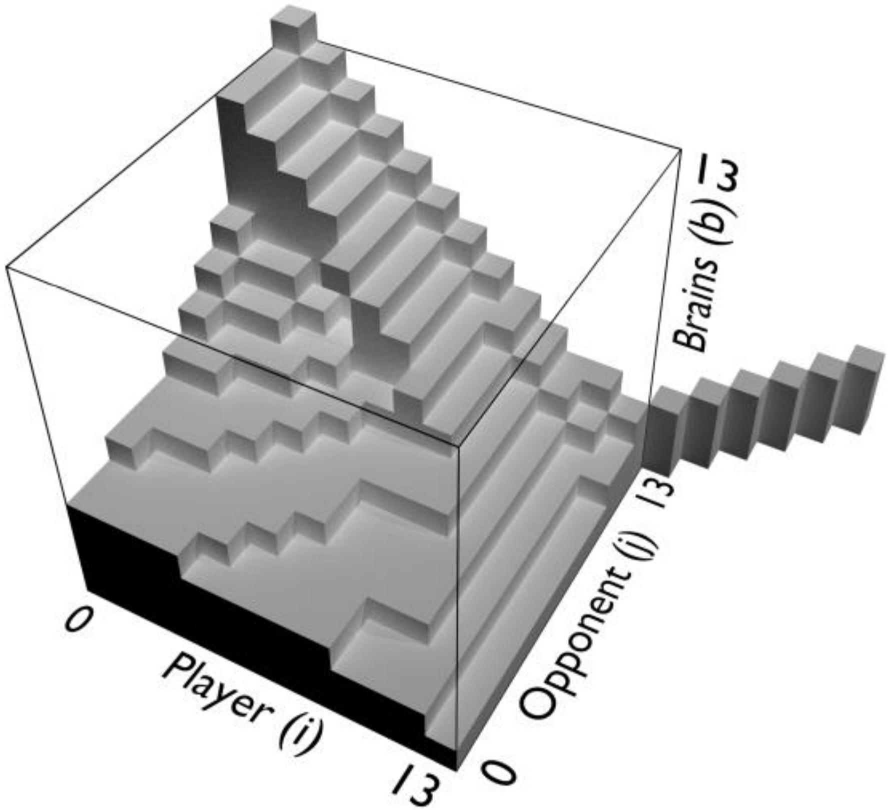


Player 1

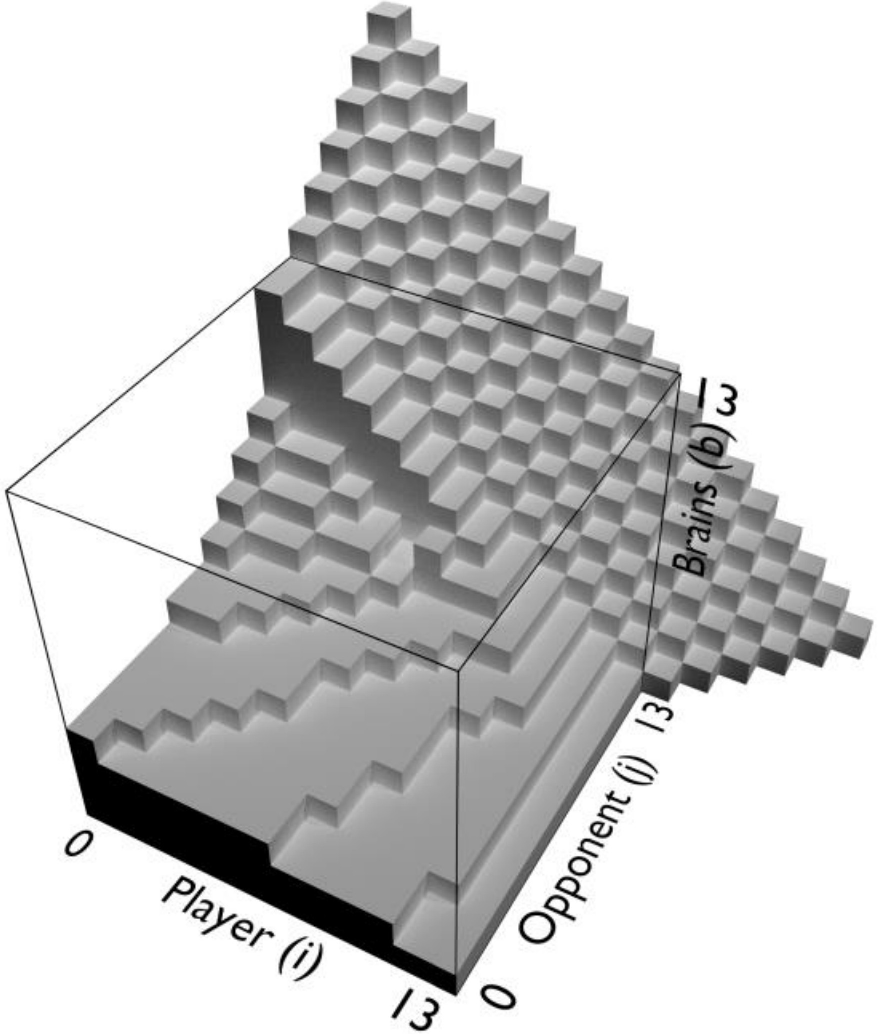


Player 2

Optimal Play (one shotgun set aside)

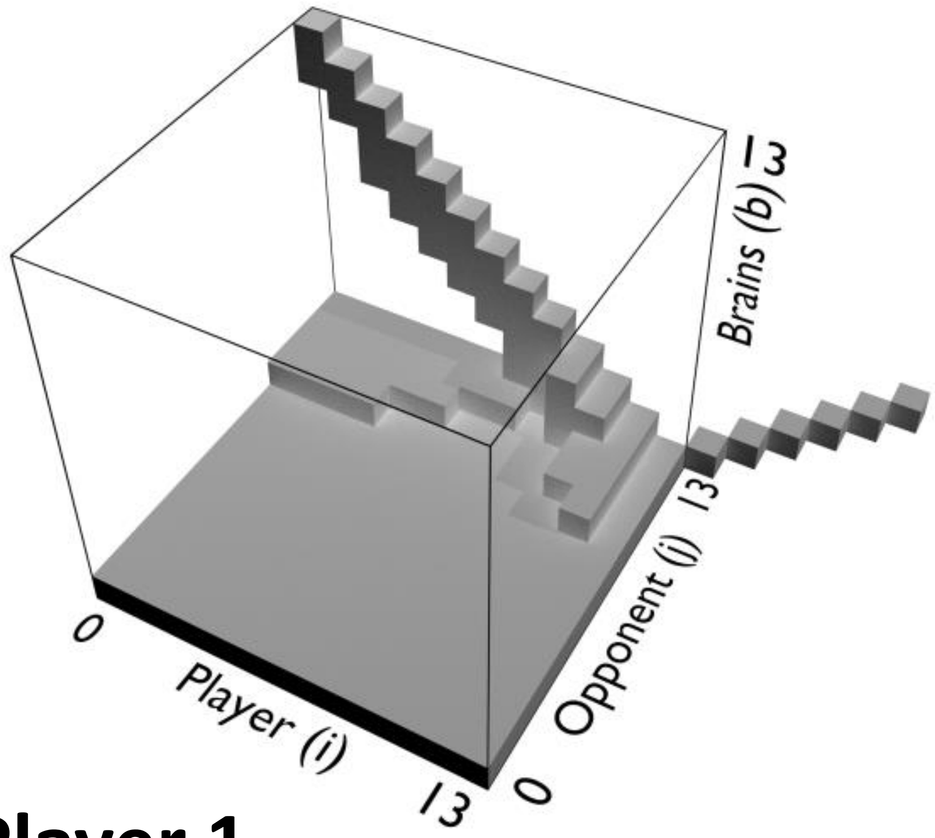


Player 1

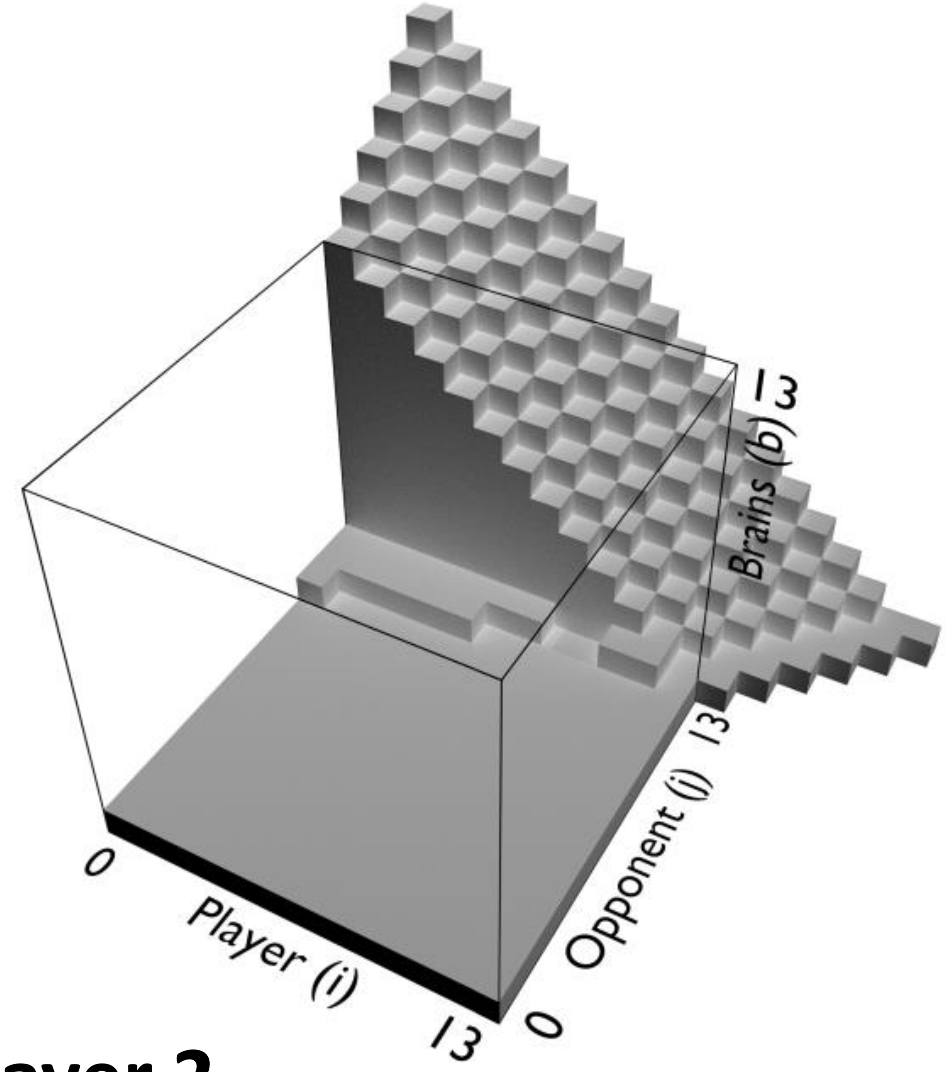


Player 2

Optimal Play (two shotguns set aside)



Player 1



Player 2

Human-Playable Policies

- By human-playable, we mean involving simple mental arithmetic and limited recall of cases and constants.
- We will see a continuum of tradeoffs from simple play policies with modest performance, to complex play policies with excellent performance.

Policy	**Difference**
Fixed Hold-At	-0.0274
Minh Cases	-0.0133
Llano Cases	-0.0118
Neller Cases	-0.0100

Fig. 2: Differences between human-playable and optimal policy win rates

Fixed Hold-At Policy

Algorithm 1: Fixed Hold-At Policy

Input : player p, player score i, opponent score j, turn total b, shotguns rolled s

Output: whether or not to roll

if $p = 2 \wedge j \geq 13 \wedge i + b < j$ **then** // When player 2 with $j \geq goal \ldots$
    return *true* // and holding would lose, roll.
else if $s = 0$ **then** // Keep rolling with 0 shotguns.
    return *true*
else if $s = 1$ **then** // Hold at 4 with 1 shotgun.
    return $b < 4$
else // Hold at 1 with 2 shotguns.
    return $b < 1$
end if

2.74% gap

Minh Cases Policy

Algorithm 2: Minh Cases Policy

Input : player p, player score i, opponent score j, turn total b, shotguns rolled s

Output: whether or not to roll

if $p = 2 \land j \geq 13 \land i + b < j$ **then** // When player 2 with $j \geq goal \ldots$
    return *true* // and holding would lose, roll.
else if $s = 0$ **then** // Keep rolling with 0 shotguns.
    return *true*
else if $s = 1$ **then** // With 1 shotgun,
    if $j \geq 8$ **then** // if opponent's score $j \geq 8$,
        return $i + b < \max(13,\ j + 3)$ // win (with lead of 3 if $j \geq 10$),
    else // else hold at 4.
        return $b < 4$
    end if
else // Hold at 1 with 2 shotguns.
    return $b < 1$
end if

1.33% gap

Minh Cases Policy

- When player 1 has reached/exceeded the goal score, and player 2 would lose when holding, player 2 rolls.
- Otherwise:
  - 0 shotguns set aside: Keep rolling.
  - 1 shotgun set aside:
    - Opponent score ≥ 8: win (with lead of 3 if opponent score ≥ 10)
    - Opponent score < 8: hold at 4
  - 2 shotguns set aside: hold at 1

1.33% gap

Llano Cases Policy (part 1 of 2)

Algorithm 3: Llano Cases Policy

Input : player p, player score i, opponent score j, turn total b, shotguns rolled s

Output: whether or not to roll

$i' \leftarrow i + b$ // i': score after holding

$h = \{6, 3, 1\}$ // h: tiebreaker hold values indexed by shotguns rolled

if $i \geq 13 \vee j \geq 13$ **then** // If either player reached/exceeded goal...

    if $i = j$ **then** // if the scores are even...

        if $p = 1$ **then** // player 1 holds at the appropriate turn score.

            return $b < h[s]$

        else // Player 2 holds when b reaches 1.

            return $b < 1$

        end if

    else if $p = 2 \wedge i < j$ **then** // If player 2 is trailing...

        return $(s < 2 \wedge i' \leq j) \vee (s = 2 \wedge i' < j)$ // match the opponent with 2 shotguns, exceed by 1 otherwise.

    end if

    return *false* // Otherwise hold.

1.18% gap

Llano Cases Policy (part 2 of 2)

```
else                                    // If both players are below goal score...
    if s = 0 then                       // if 0 shotguns rolled...
        if p = 1 then // player 1 goes for the higher of goal or j + 9.
            return i' < max(13, j + 9)
        else                            // Player 2 goes for the goal.
            return i' < 13
        end if
    else if s = 1 then                  // If 1 shotgun rolled...
        if i ≥ 10 ∧ j ≥ 10 then         // and either player has at least 10...
            return i' < 13              // go for the goal.
        else
            return b < 4                // Otherwise, hold at 4.
        end if
    else                                // If 2 shotguns rolled...
        return b < 1                    // Hold at 1.
    end if
end if
```

1.18% gap

Neller Cases Policy (part 1 of 2)

Algorithm 4: Neller Cases Policy

Input : player p, player score i, opponent score j, turn total b, shotguns rolled s

Output: whether or not to roll

```
i′ ← i + b                                   // i′: score after holding
h = {6, 3, 1}   // h: tiebreaker hold values indexed by shotguns rolled
if p = 1 then                                       // If player 1, ...
    if i ≥ 13 ∧ i = j then        // if tied at/above 13, hold at h values.
        return b < h[s]
    else if s = 0 then   // If 0 shotguns, hold at ≥ 13 with ≥ 8 lead.
        return i′ < max(13, j + 8)
    end if
    e ← 8                // Set player 1 end-game score threshold e to 8.
else                                                // Else if player 2, ...
    if j ≥ 13 then          // if player 1 has achieved the goal score, ...
        return (s < 2 ∧ i′ ≤ j) ∨ (s = 2 ∧ i′ < j)  // exceed/meet player 1's
          score with under/exactly 2 shotguns, respectively.
    else if i ≥ 13 ∧ i > j then                    // If holding wins, hold.
        return false
    else if s = 0 then                     // Always roll with no shotguns.
        return true
    end if
    e ← 10              // Set player 2 end-game score threshold e to 10.
end if
```

1.00% gap

Neller Cases Policy (part 2 of 2)

```
if s = 1 then                          // If 1 shotgun rolled, ...
    if i ≥ e ∨ j ≥ e then              // hold at 13 if score(s) ≥ e.
        return i′ < 13
    else                               // Otherwise, hold at 4.
        return b < 4
    end if
else if s = 2 then                     // If 2 shotguns rolled, hold at 1.
    return b < 1
end if
```

1.00% gap

Future Work

- Our next step will be to compute optimal play for the full complexity of 2-player Zombie Dice.
- We will then compare performance of both the optimal and human-playable AYZD policies against optimal Zombie Dice play to see how much dice color distribution matters for play performance.

Conclusions

- Optimal play has been computed for the 2-player All Yellow Zombie Dice game.
- A variety of human playable strategies were presented, including the simple "Minh Cases" strategy that has a 1.33% gap from the optimal win rate.
  - When it's player 2's turn, roll if player 1 would win when player 2 holds. Otherwise,
  - With 0 shotguns set aside, keep rolling.
  - With 1 shotgun set aside,
    - Opponent score ≥ 8, hold at 13 (with lead of 3 if opponent score ≥ 10)
    - Opponent score < 8, hold at 4.
  - With 2 shotguns set aside, hold at 1.