Rook Jumping Maze
Design Considerations

Todd Neller¹
Adrian Fisher²
Munyaradzi Choga¹
Samir Lalvani¹
Kyle McCarty¹

¹Gettysburg College
²Adrian Fisher Design Ltd.
Rook Jumping Maze

- **Specification**: grid size, start state (square), goal state, jump numbers for each non-goal state.
- **Jump number**: Move *exactly* that many squares up, down, left, right. (*Not* diagonally.)
- **Objectives**:  
  - Find a path from start to goal.  
  - Find the shortest of these paths.
Rook Jumping Maze History

- Sam Loyd’s 1898 “Back from the Klondike”
  - Queen Jumping Maze
  - Created to defeat Euler’s backtracing method
- 1990: Robert Abbott “Jumping Jim”
- 1991: Adrian Fisher Human-size RJM
- 1997: Robert Abbott “Number Maze”
Puzzle Design as Search

- The number of possible $5 \times 5$ rook jumping mazes configurations with a center goal: $4^{16} \times 3^8 > 2.8 \times 10^{13}$ (a lot)
- The number of possible $n \times n$ mazes is bounded above by $(n-1)^{n^2}$.
- The number of \textit{good} puzzle configurations is considerably less (many needles in a \textit{very} large haystack).
- We can’t generate and test all configurations.
- We can search for a good one.
The Search Problem

1) We need a way to rate the maze relative (un)desirability
   ◦ e.g. penalize if goal not reachable from a state
2) we need a method for looking around:
   ◦ Start with a *random* maze configuration
   ◦ Change a *random* position to a *random* different jump
   ◦ Accept all improvements, reject changes for the worse with high probability
Rook Jumping Maze Generation

- The prime design challenge is to define a good energy function, scoring a maze’s undesirability.
- What are desirable/undesirable characteristics?
  - Goal reachability, reachable states, black holes, white holes, start/goal locations, shortest solution uniqueness, minimum solution path length, forward/backward decisions, initial forced moves, same jump clusters
Penalize No Unique Shortest Solution

- We want to have a solution, and that solution should be uniquely shortest.
- Strongly penalize mazes with no solution.
- Let $|S|$ be the number of states.
- Score: If there is no unique shortest solution, add $|S|^3$ to the undesirability score.
Black holes and white holes

- Black hole: Group of *reachable* states that are *non-reaching*, i.e. can reach from initial state, but can’t reach goal (“forward dead end”)
- White hole: Group of *unreachable* states that are *reaching*, i.e. can’t reach from initial state, but can reach goal (“backward dead end”)
- We penalize black holes, but don’t penalize white holes.
  - Black holes force a restart, encouraging disengagement.
  - White holes increase the difficulty of visual backtracing.
<table>
<thead>
<tr>
<th></th>
<th>3</th>
<th>4</th>
<th>1</th>
<th>3</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>G</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>4</td>
<td>3</td>
<td>2</td>
<td></td>
</tr>
</tbody>
</table>
We want to have all reachable states to have a path to the goal. (No black holes.)
Thus, unreaching states must also be unreachable \(\rightarrow\) wasted maze space.
Score: Add \(|S|^2\) per *unreaching* state, i.e. state with no path to goal.
Penalize Initial Forced Moves

- Initial forced moves worsen the maze design.
- Let $m$ be the number of initial forced moves.
- Score: Add $m^2$.
- Non-corner initial states allow initial forced moves.
  - Restrict initial state to upper-left corner.
  - Allow goal state in any other position for variety.
Reward many decisions

- We prefer decisions over forced moves, working forward \textit{or} working backward.
- $d_f, d_b$ – number of forward, backward decisions along optimal solution path, respectively.
- Score: \textit{Subtract} $\min(d_f, d_b)$. 
A *same jump cluster* is a group of states with the same jump number that are all reachable from each other:

- For each same jump cluster $J$, let $|J|$ be the size.
  - Score: Add $(|J| - 1)^2$.

(b) Maze with 2-jump clusters  
(c) Maze with 3-jump clusters
What Do You Observe?

<table>
<thead>
<tr>
<th>3</th>
<th>3</th>
<th>3</th>
<th>3</th>
<th>3</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>G</td>
<td>3</td>
<td>4</td>
<td>3</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
“Jump Maze” iPhone App

- Free on Apple iStore
- Twofold cognitive challenge
  - Perception of graph topology
  - Memory of past moves
- Hint feature: highlight visited states
- Notes:
  - $5 \times 5$ challenging for average user
  - Checkerboard, colors aid visually
  - 10,000 cached mazes for speed
- Rook Jumping Maze of the Day:
  - [http://tinyurl.com/rjmaze](http://tinyurl.com/rjmaze)
Variations

- Many variations are possible:
  - Use different regular tilings, e.g. triangular or hexagonal.
  - **Topological constraints** may be added (e.g. impassable walls/tiles) or removed (e.g. toroidal wrap-around).
  - **Movement constraints** may be varied as well.
    - Add diagonal moves → Queen Jumping Maze
    - Abbott's "no-U-turn" rule increases state complexity
Summary

- Stochastic local search is a simple, powerful algorithm for finding good configurations in a vast space of configurations, if:
  - One can identify a good “local” step, and
  - One can characterize relative (un)desirability via an energy function.

- We’ve presented a number of features useful for defining a good RJM design energy function.

- Everything you’d ever want to know about RJMs:
  - [http://tinyurl.com/rjmaze](http://tinyurl.com/rjmaze)

- Questions?