AI For Generated Worlds

John Abercrombie, Epic Games
Damián Isla, The Molasses Flood
Anatomy of TFITF

THE GAME STRUCTURE - "GO WITH THE FLOW"

"TOOBIN' MEETS OREGON TRAIL"
The Flame in the Flood

AI Ecology Inspired by BioShock

- “AI live in the environment and do what they do.”

So where’s the “encounter” at?

- Resource Spawning
- AI Spawning
- AI territories
- AI Spatial behavior
Anatomy of an “Explore Island”

- Closed Arena
- Single entrance
- “Sockets” filled with randomized “locations” (max 4 per island)
  - Primary: Church, Gas Station, Clinic, Campground, etc.
  - Secondary: Bus (for sleeping in), Water source, Quest cache, Bear den, etc.
- Additional Randomized geometry
- Standing water (sometimes)
- Heat sources (e.g. campfires)
Design Goal

From the Halo days:

“the harder you push, the harder the encounter pushes back.”

In the case of *The Flame in the Flood*

- The farther into the explore island you go, the greater the danger
- But the greater the reward
We don’t know much

But we do know this:

- Traversable space
- Single exit
- Primary location
- Standing water
- Heat sources (e.g. campfire)
Feature Maps

Discretize space into high-resolution square grid

Using grid + Dijkstra’s, create feature maps
  - Distance-to-edge
  - Distance-to-exit
  - Distance-to-water
  - Distance-to-territory

Spatial analysis for spawning/allocation similar or identical to Spatial analysis for runtime AI positioning
Distance Features are waaaaaay more useful than binary features

i.e. rather than say
- “Is this point in boar territory?”
Say
- “How far is this point from boar territory?”
Plants: What types do we spawn?

- Plant types determined by a higher-level scheduler
- Defined with \texttt{min}, \texttt{max} and \texttt{average} interval between appearances
  - Cattails: ~ every 2 islands but never more than 4
  - Saplings: ~ every 3 islands but never more than 6
  - Devils trumpet: ~ every 5 islands, never more than 10

More on this in “Forging the River” talk on Thurs
Vegetation Spawning

- Placement parameters
  - Water distance min/max
  - Distance to edge min/max
- Rarity → Distance from Entrance
  - Common plants spawned near entrance
  - Rare/powerful plants far from entrance
  - “The farther in you go, the greater the reward”
Devil’s Trumpet
Yucca
Dandelion
Saplings
AI Spawning

- Generate a coarser grid of spawn points to evaluate for spawn locations
  - Jittered Hex Grid

- Generally also have a “no-go” margin to prevent other AI-types from spawning too near them

- Spawn AI in priority order
<table>
<thead>
<tr>
<th>Animal</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rabbit</td>
<td>Spawn near cluster of 3-5 burrows</td>
</tr>
<tr>
<td>Wolf</td>
<td>Spawn around primary location</td>
</tr>
<tr>
<td>Bear</td>
<td>Spawn at “bear den” location.</td>
</tr>
<tr>
<td>Boar</td>
<td>Generate a territory using spatial analysis.</td>
</tr>
</tbody>
</table>
Boar territory is always defined by a “front”

- Boar spawned near the front
- Anything beyond the front is boar’s territory
- Remember, high-value loot is always spawned back there
Boar territory is always defined by a “front”

- Boar spawned near the front
- Anything beyond the front is boar’s territory
- Remember, high-value loot is always spawned back there
Summary

1. Define spatial features in terms of what you do know about the space
   a. Distance Features >> Binary Features

2. Define Spawn criteria and spatial behavior in terms of the spatial features
   a. Analysis system for Spawning can/should be the same as for runtime positioning

3. Dijkstra is your friend

4. Exploit serendipity + player’s tendency to interpret things as meaningful/intentional.
So how do you generate encounters without designing encounters?

We ARE still designing encounters.

Designing them in terms of abstract spatial features
Thank you!

Thank you!

@damian_isla
@molassesflood