Video Game AI: Lecture 2
History of Game AI

Nathan Sturtevant
COMP 3705

UNIVERSITY OF DENVER

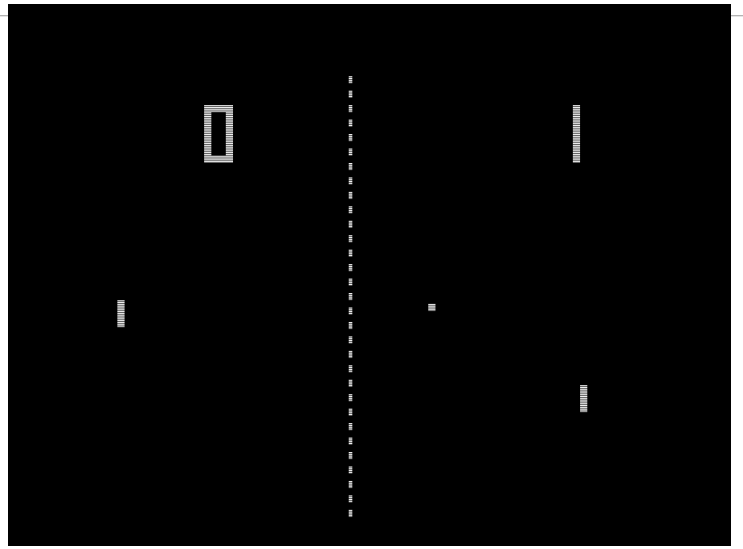
Today's Topics

- Brief history of video game AI
- PacMan Discussion / Quiz
  - Design
  - What role do ghosts play?
  - How could ghosts be changed?
- AI Restrictions / Evaluating AI

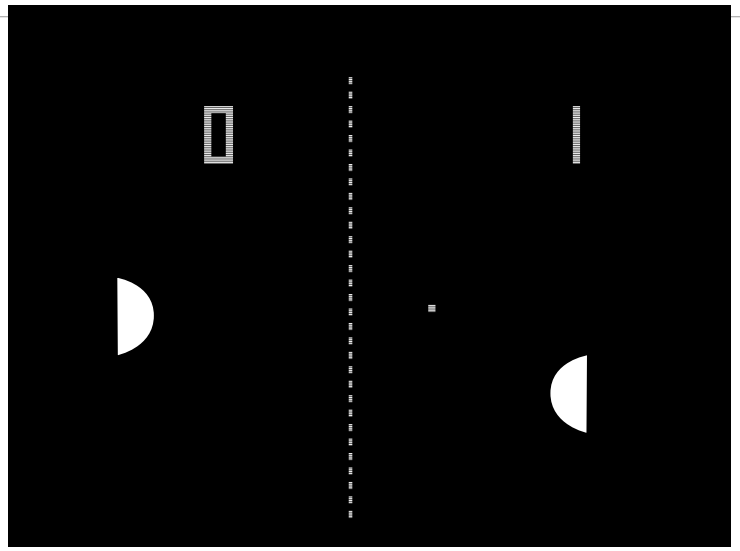
A selective history of video game AI

UNIVERSITY OF DENVER

Pong (1972)

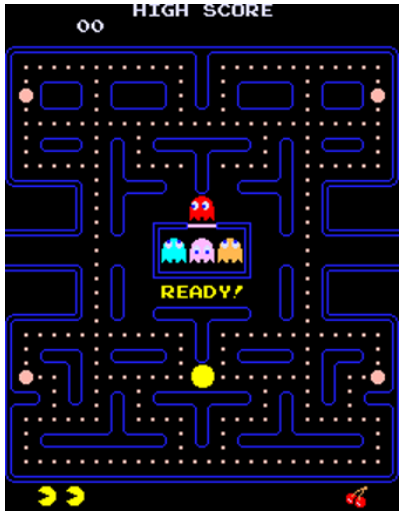


Pong (1972)



Pac-Man (1980)

- Everything you ever want to know about Pacman:
  - http://home.comcast.net/~jpittman2/pacman/pacmandossier.html



AI Behavior - 3 states

- **Chase**
  - Hunt down pac-man using different
- **Scatter**
  - Head for home areas for each ghost
- **Frightened**
  - Wander aimlessly for a few seconds

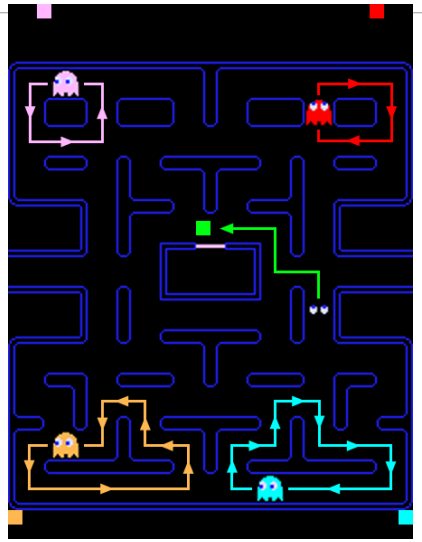
- Ghosts can only change direction when changing state

Time limits for each mode

Mode	Level 1	Levels 2–4	Levels 5+
Scatter	7	7	5
Chase	20	20	20
Scatter	7	7	5
Chase	20	20	20
Scatter	5	5	5
Chase	20	1033	1037
Scatter	5	1/60	1/60
Chase	indefinite	indefinite	indefinite

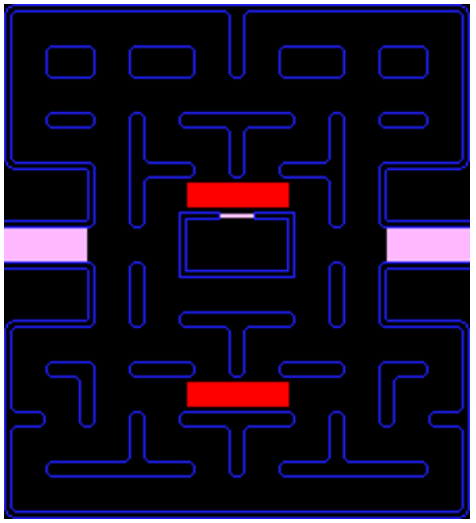
UNIVERSITY OF DENVER

Scatter locations



UNIVERSITY OF DENVER

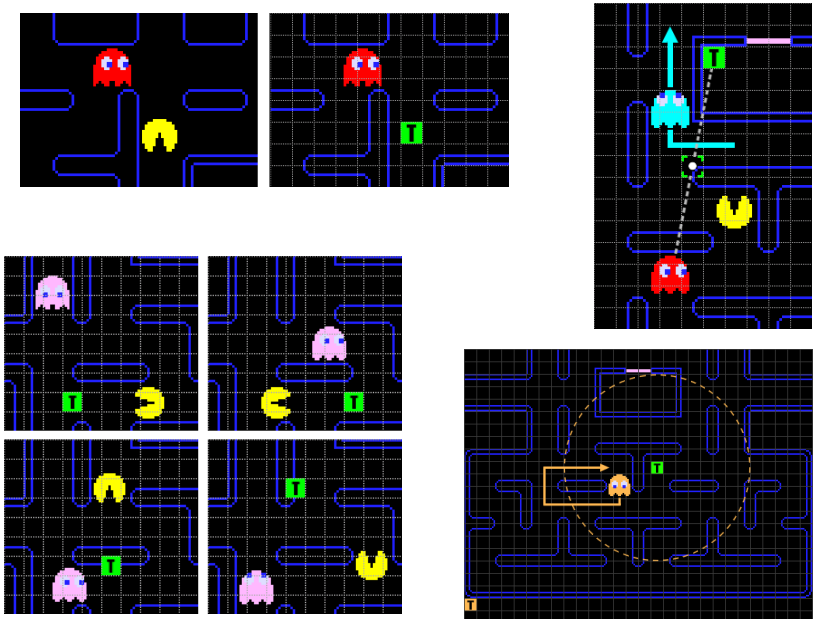
Exploits



UNIVERSITY OF DENVER

Video

http://www.gdcvault.com/play/1014631/Classic-Game-Postmortem-PAC



UNIVERSITY OF DENVER

Karateka (1984)

UNIVERSITY OF DENVER

Karateka (1984)

UNIVERSITY OF DENVER

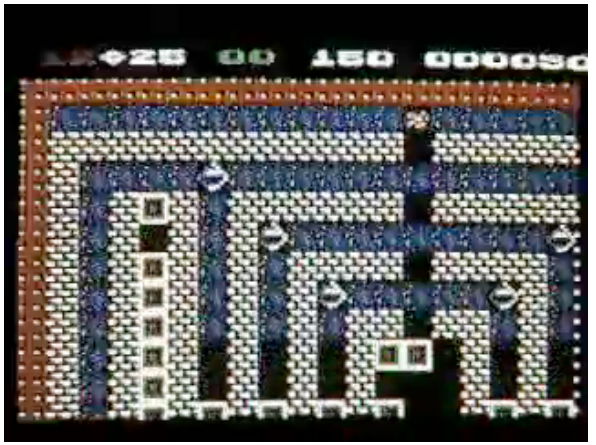
Boulder Dash (1984)

- Simple predictable behavior

UNIVERSITY OF DENVER

Boulder Dash (1984)

- Simple predictable behavior

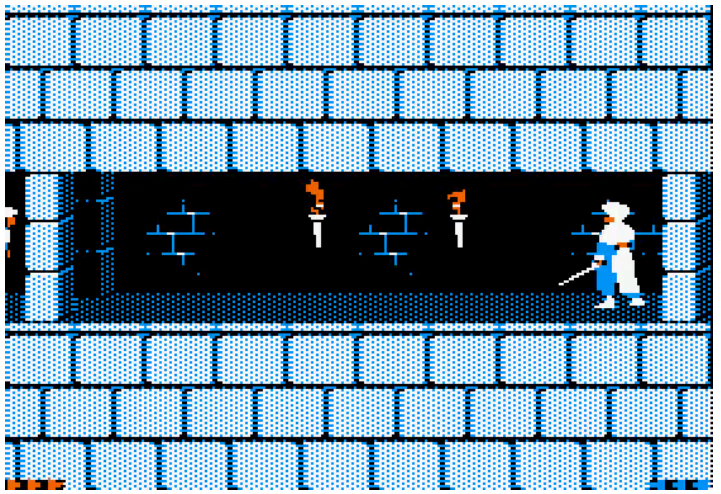


UNIVERSITY OF DENVER

Prince of Persia (1989)

UNIVERSITY OF DENVER

Prince of Persia (1989)

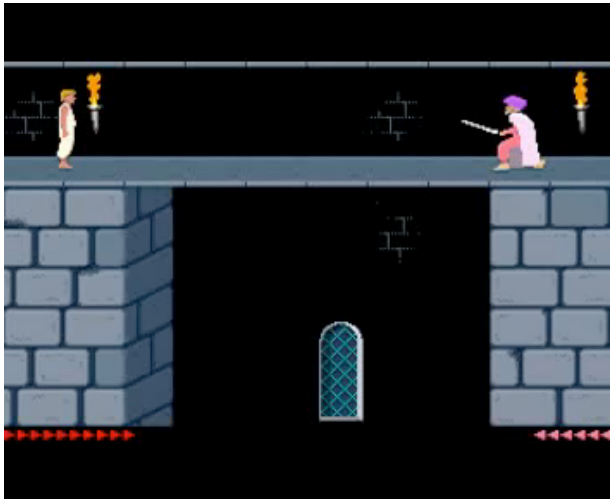


UNIVERSITY OF DENVER

Prince of Persia (1989)

UNIVERSITY OF DENVER

Prince of Persia (1989)



UNIVERSITY OF DENVER

Black and White (2001)

UNIVERSITY OF DENVER

Black and White (2001)

UNIVERSITY OF DENVER

Black and White (2001)

- AI designed by Richard Evans
  - Also did AI for The Sims
  - Now working at a startup on a new title
- Slap or stroke a creature to give negative/positive reinforcement
  - Creatures learn from your feedback
    - If you understand what you're doing

UNIVERSITY OF DENVER

Case Study: Halo

- GDC 2002 talk covering Halo AI
  - Jaime Griesemer & Chris Butcher
- How did they design the AI?
  - Avoid heavy scripting
  - Avoid masses of enemies

UNIVERSITY OF DENVER

Case Study: Halo

- GDC 2002 talk covering Halo AI
  - Jaime Griesemer & Chris Butcher
- How did they design the AI?
  - Avoid heavy scripting
  - Avoid masses of enemies

UNIVERSITY OF DENVER

Case Study: Halo

- Building a good AI is a mix of design and programming
  - Designers worked on long-term interactions

    (~3 minutes)
  - Program/script the short-term behaviors

    (run from grenade, etc)
- Give the AI the same capabilities as player

UNIVERSITY OF DENVER

Case Study

- Predictability
  - Want enemies to be predictable…
    - …give player the joy of beating them
    - Added “breaking point” change of behavior
      - When AI is almost dead, drastically change behavior
- Unpredictability
  - Random enemies too unpredictable
  - Try to make human random
    - AI becomes more unpredictable

UNIVERSITY OF DENVER

Player Feedback on AI

- Stronger enemies perceived as smarter

UNIVERSITY OF DENVER

Player Feedback on AI

- Stronger enemies perceived as smarter

Weak Enemy Playtest

Too hard	12%
About right	52%
Too easy	**36%**

Very Intelligent	**8%**
Somewhat Intelligent	72%
Not Intelligent	20%

UNIVERSITY OF DENVER

Player Feedback on AI

- Stronger enemies perceived as smarter

Weak Enemy Playtest

Too hard	12%
About right	52%
Too easy	**36%**

Very Intelligent	**8%**
Somewhat Intelligent	72%
Not Intelligent	20%

Tough Enemy Playtest

Too hard	7%
About right	92%
Too easy	**0%**

Very Intelligent	**43%**
Somewhat Intelligent	57%
Not Intelligent	0%

UNIVERSITY OF DENVER

Level Design

- Design levels to show off AI
  - Not much AI needed to fight in a long hallway
- Make sure visual cues are obvious

UNIVERSITY OF DENVER

Level Design

- Design levels to show off AI
  - Not much AI needed to fight in a long hallway
- Make sure visual cues are obvious

"In Halo the Grunts run away when an Elite is killed. Initially nobody noticed so we had to keep adding clues to make it more obvious. By the time we shipped we had made it so not only does **every single** Grunt run away **every single** time an Elite is killed but they all have an outrageously exaggerated panic run where they wave their hands above their heads they scream in terror and half the time one of them will say 'Leader Dead, Run Away!' I would still estimate that less than a third of our users made the connection"

UNIVERSITY OF DENVER

Design Decisions

- Can handle 20-25 units, 2-4 vehicles
- AI can't track everyone around them
  - Only track 3-5 important players
- Use sound and animation to convey internal state of character

UNIVERSITY OF DENVER

Technologies

- Build a model of the world
  - Emotional state of units
  - Complex perceptions of world
  - Implemented in a Finite State Machine
- Ray-casting for lines of view
  - 60% of AI code

The Canonical Encounter

The Canonical Encounter

Two-stage fallback

The Canonical Encounter

Two-stage fallback

- Enemies occupy a territory

The Canonical Encounter

Two-stage fallback

- Enemies occupy a territory
- Pushed to “fallback” point

The Canonical Encounter

Two-stage fallback

- Enemies occupy a territory
- Pushed to “fallback” point
- Pushed to “last-stand” point

The Canonical Encounter

Two-stage fallback

- Enemies occupy a territory
- Pushed to “fallback” point
- Pushed to “last-stand” point
- Player “breaks” them

The Canonical Encounter

Two-stage fallback

- Enemies occupy a territory
- Pushed to “fallback” point
- Pushed to “last-stand” point
- Player “breaks” them
- Player finishes them off

The Canonical Encounter

Two-stage fallback

- Enemies occupy a territory
- Pushed to “fallback” point
- Pushed to “last-stand” point
- Player “breaks” them
- Player finishes them off

... plus a little “spice”

The Canonical Encounter

Two-stage fallback

- Enemies occupy a territory
- Pushed to “fallback” point
- Pushed to “last-stand” point
- Player “breaks” them
- Player finishes them off

... plus a little “spice”

- snipers

The Canonical Encounter

Two-stage fallback

- Enemies occupy a territory
- Pushed to “fallback” point
- Pushed to “last-stand” point
- Player “breaks” them
- Player finishes them off

... plus a little “spice”

- snipers
- turrets

The Canonical Encounter

Two-stage fallback

- Enemies occupy a territory
- Pushed to “fallback” point
- Pushed to “last-stand” point
- Player “breaks” them
- Player finishes them off

... plus a little “spice”

- snipers
- turrets
- dropships

The Canonical Encounter

Two-stage fallback

- Enemies occupy a territory
- Pushed to “fallback” point
- Pushed to “last-stand” point
- Player “breaks” them
- Player finishes them off

... plus a little “spice”

- snipers
- turrets
- dropships

The Canonical Encounter

Two-stage fallback

- Enemies occupy a territory
- Pushed to “fallback” point
- Pushed to “last-stand” point
- Player “breaks” them
- Player finishes them off

... plus a little “spice”

- snipers
- turrets
- dropships

The Canonical Encounter

Two-stage fallback

- Enemies occupy a territory
- Pushed to “fallback” point
- Pushed to “last-stand” point
- Player “breaks” them
- Player finishes them off

... plus a little “spice”

- snipers
- turrets
- dropships

Task

Task

The mission designers' language for telling the AI what it should be doing

Halo:

- Territory
- Behavior
  - aggressiveness
  - rules of engagement

Task

The mission designers' language for telling the AI what it should be doing

Halo:

- Territory
- Behavior
  - aggressiveness
  - rules of engagement
  - player following

Task

The mission designers' language for telling the AI what it should be doing

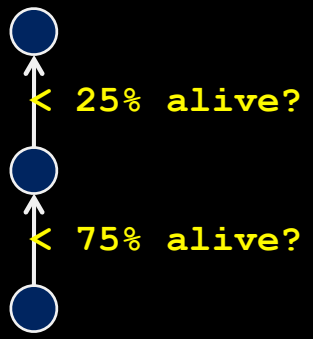
Halo:

- Territory
- Behavior
  - aggressiveness
  - rules of engagement
  - player following

Changing task moves AI around the encounter space

The Imperative Method

Give the designers an FSM construction tool



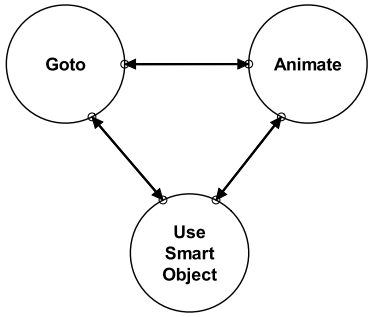
UNIVERSITY OF DENVER

F.E.

UNIVERSITY OF DENVER

F.E.A.R. States

- Goto
  - Physical movement to a new location
- Animate
  - In-place animation
- Use Smart Object
  - Special-case of animate



UNIVERSITY OF DENVER

AI Psychology in F.E.A.R.

- People will perceive the AI as smarter if they know what it is doing
  - If only one unit remains, say “I need reinforcements”
  - Introduce conversations instead of someone talking to themselves
    - (I’m shot! v. What’s your status?)
  - If AI is stuck, say “I’ve got nowhere to go!”

UNIVERSITY OF DENVER

- “A gamer posting to an internet forum expressed that they he was impressed that the A.I. seem to actually understand each other’s verbal communication. ‘Not only do they give each other orders, but they actually DO what they’re told!’ Of course the reality is that it’s all smoke and mirrors, and really all decisions about what to say are made after the fact, once the squad behavior has decided what the A.I. are going to do.”

UNIVERSITY OF DENVER

Façade (2005)

- Michael Mateas, Andrew Stern

UNIVERSITY OF DENVER

Left For Dead (2008)

Goal: Generate dramatic game pacing

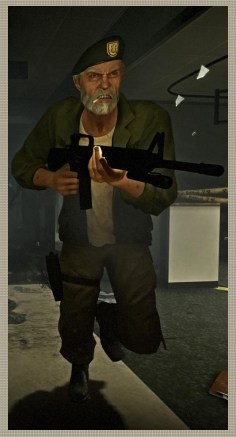
- What do we mean by "dramatic game pacing"?
  - Algorithmically adjusting game pacing on the fly to maximize "drama" (player excitement/intensity)
- Inspired by Observations from Counter-Strike
  - Natural pacing of CS is "spiky", with periods of quiet tension punctuated by unpredictable moments of intense combat
  - Constant, unchanging combat is fatiguing
  - Long periods of inactivity are boring
  - *Unpredictable* peaks and valleys of intensity create a powerfully compelling and replayable experience
  - Same scenario, often the same map, yet different and compelling experience each round

Goal: Generate dramatic game pacing

- Adaptive Dramatic Pacing algorithm
  - Creates peaks and valleys of intensity similar to the proven pacing success of Counter-Strike
  - Algorithm:
    - Estimate the "emotional intensity" of each Survivor
    - Track the max intensity of all 4 Survivors
    - If intensity is too high, remove major threats for awhile
    - Otherwise, create an interesting population of threats
  - Another key system of the **"AI Director"**

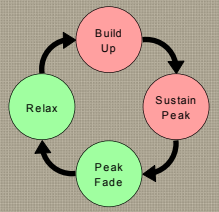
Goal: Generate dramatic game pacing
Adaptive Dramatic Pacing

- Estimating the "emotional intensity" of each Survivor
  - Represent Survivor Intensity as a value
  - Increase Survivor Intensity
    - When injured by the Infected, proportional to damage taken
    - When the player becomes incapacitated
    - When player is pulled/pushed off of a ledge by the Infected
    - When nearby Infected dies, inversely proportional to distance
  - Decay Survivor Intensity towards zero over time
  - Do NOT decay Survivor Intensity if there are Infected actively engaging the Survivor



Goal: Generate dramatic game pacing
Adaptive Dramatic Pacing

- Use Survivor Intensity to modulate the Infected population
  - Build Up
    - Create full threat population until Survivor Intensity crosses peak threshold
  - Sustain Peak
    - Continue full threat population for 3-5 seconds after Survivor Intensity has peaked
    - Ensures minimum "build up" duration.
  - Peak Fade
    - Switch to minimal threat population ("Relax period") and monitor Survivor Intensity until it decays out of peak range
    - This state is needed so current combat engagement can play out without using up entire Relax period. Peak Fade won't allow the Relax period to start until a natural break in the action occurs.
  - Relax
    - Maintain minimal threat population for 30-45 seconds, or until Survivors have traveled far enough toward the next safe room, then resume Build Up.

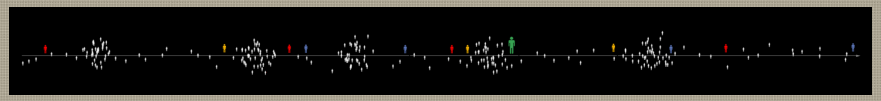


Goal: Generate dramatic game pacing
Adaptive Dramatic Pacing

- Full Threat Population (Build Up)
  - Wanderers
  - Mobs
  - Special Infected
- Minimal Threat Population (Relax)
  - No Wanderers until team is "calm"
  - No Mobs
  - No Special Infected (although existing Specials may attack)
- Boss Encounters NOT affected by adaptive pacing
  - Overall pacing affected too much if they are missing
  - Boss encounters are intended to change up the pacing anyhow

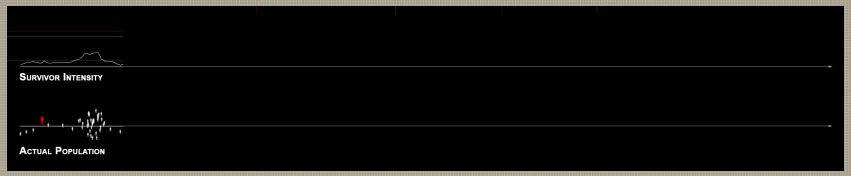
Goal: Generate dramatic game pacing
Adaptive Dramatic Pacing

- An example procedurally generated population



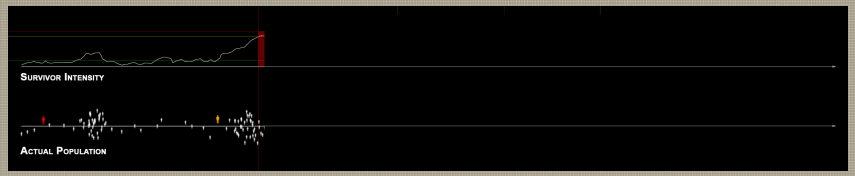
Goal: Generate dramatic game pacing
Adaptive Dramatic Pacing

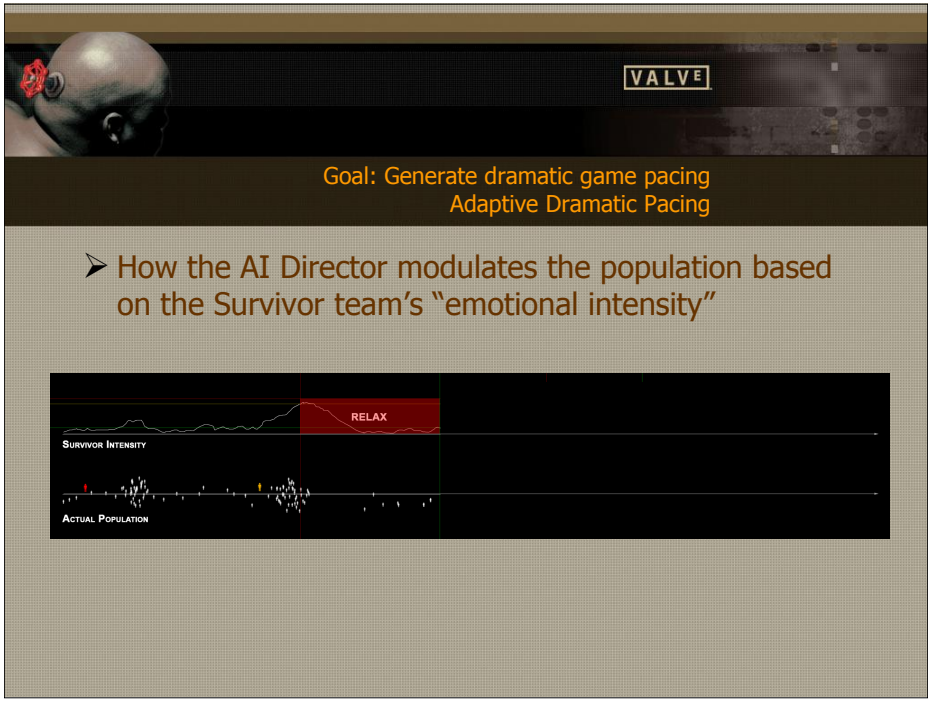
- How the AI Director modulates the population based on the Survivor team's "emotional intensity"

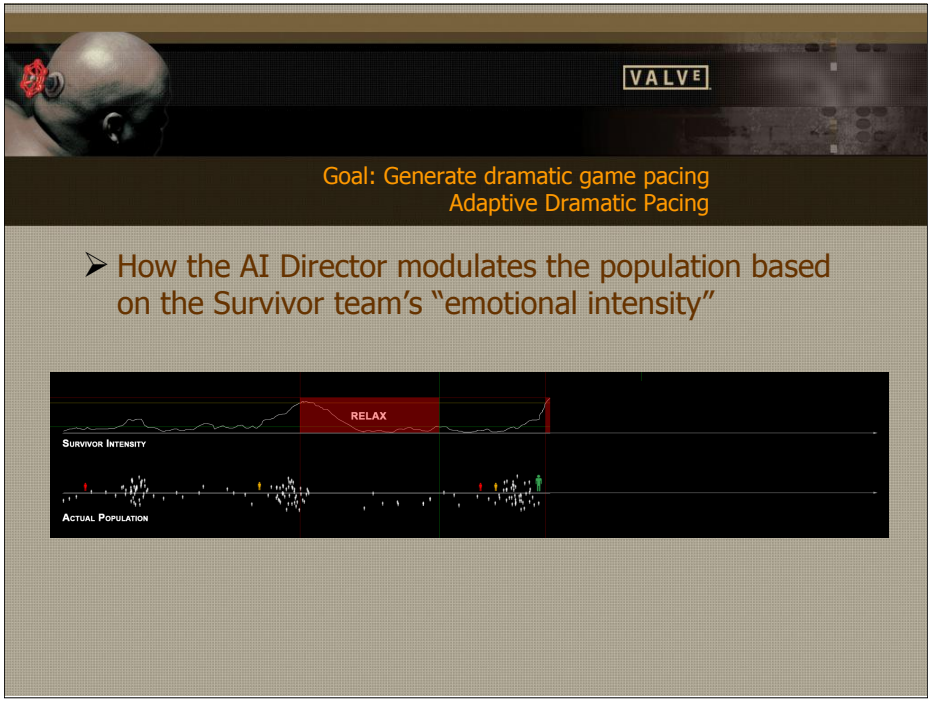


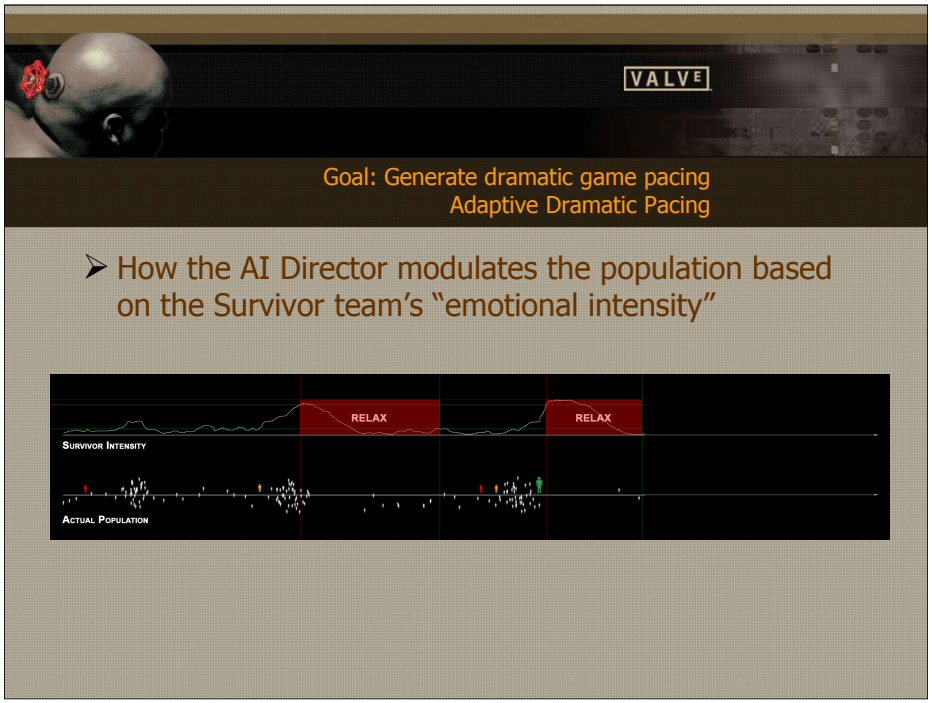
Goal: Generate dramatic game pacing
Adaptive Dramatic Pacing

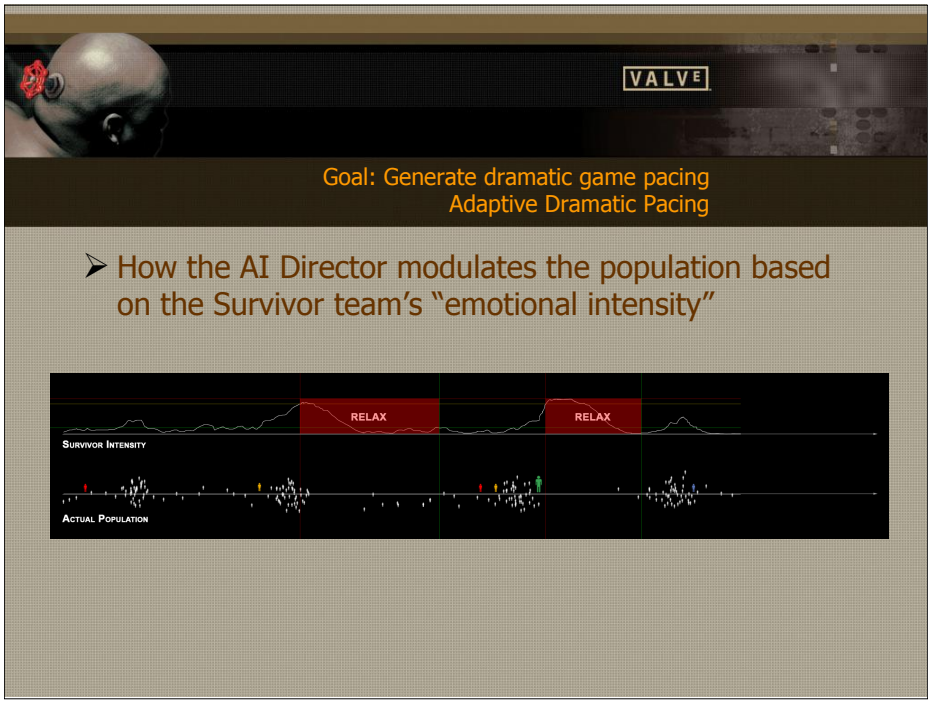
- How the AI Director modulates the population based on the Survivor team's "emotional intensity"




VALVE
Goal: Generate dramatic game pacing
Adaptive Dramatic Pacing
How the AI Director modulates the population based on the Survivor team's "emotional intensity"
RELAX
Survivor Intensity
Actual Population


VALVE
Goal: Generate dramatic game pacing
Adaptive Dramatic Pacing
How the AI Director modulates the population based on the Survivor team's "emotional intensity"
RELAX
Survivor Intensity
Actual Population

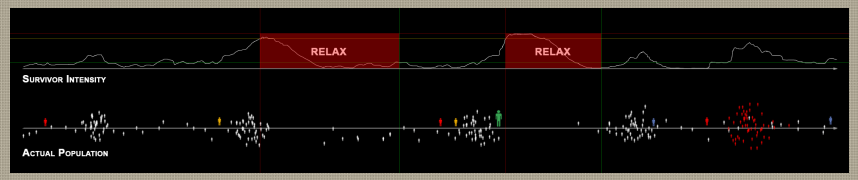

VALVE
Goal: Generate dramatic game pacing
Adaptive Dramatic Pacing
How the AI Director modulates the population based on the Survivor team's "emotional intensity"
RELAX
RELAX
Survivor Intensity
Actual Population


VALVE
Goal: Generate dramatic game pacing
Adaptive Dramatic Pacing
How the AI Director modulates the population based on the Survivor team's "emotional intensity"
RELAX
RELAX
Survivor Intensity
Actual Population

VALVE

Goal: Generate dramatic game pacing
Adaptive Dramatic Pacing

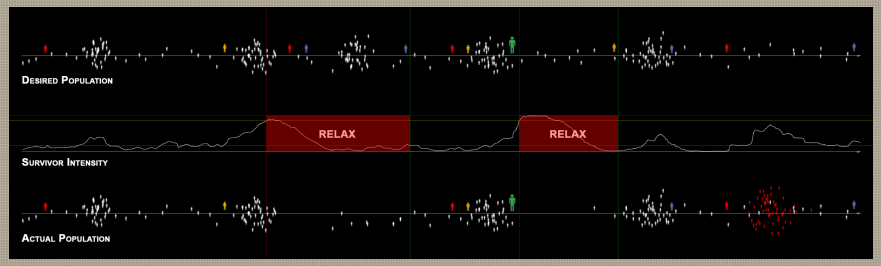
- How the AI Director modulates the population based on the Survivor team's "emotional intensity"



VALVE

Goal: Generate dramatic game pacing
Adaptive Dramatic Pacing

- Comparing population after modulation by the AI Director



VALVE

Goal: Generate dramatic game pacing

- Adaptive Dramatic Pacing reacts to Survivor team
  - Generates reliable peaks of intensity without completely overwhelming the team
  - Because of player variation and procedural threat population, timing and location of peaks will differ each time the game is played
- Algorithm adjusts pacing, not difficulty
  - Amplitude (difficulty) is not changed, frequency (pacing) is
- Simple algorithms can generate compelling pacing schedules
  - Survivor Intensity estimation is crude, yet the resulting pacing works

UNIVERSITY OF DENVER

The Sims *X* (2000-2010)

- How can you ship a title and expansions packs?
  - Sims have set needs
  - Objects have affordances which meet needs
  - Decide which object based on the comparative utility of your actions