



COMP702: Classical Artificial Intelligence

2: Authored Behaviour

Rule-based AI

Rule-based AI

- ▶ Generally **reactive** to the state of the world
- ▶ Based on **if-then** triggers, basic **calculations**, etc.
- ▶ Generally hand-coded and only modifiable by a programmer

Case study: Ghosts in Pac-Man

- ▶ Full details: <http://gameinternals.com/understanding-pac-man-ghost-behavior>
- ▶ Each ghost has 3 states
 - ▶ Chase: head for a specific position (see next slide)
 - ▶ Scatter: head for a specific corner of the level
 - ▶ Frightened: move randomly

Ghost “personalities”

- ▶ Red ghost: aim for Pac-Man
- ▶ Pink ghost: aim for 2 spaces ahead of Pac-Man
- ▶ Blue ghost: aim for position on the line between red ghost and 2 spaces ahead of Pac-Man
- ▶ Orange ghost: aim for Pac-Man until 8 spaces away, then aim for corner

Ghost movement

- ▶ No pathfinding — greedily move towards target
- ▶ Can only change direction at an intersection
- ▶ Can't reverse or stay still
- ▶ Therefore can't get stuck, despite imperfect pathfinding

Ghost behaviour

- ▶ Behaviour rules are very simple
- ▶ However, the combination of them leads to interesting gameplay and illusion of personality

Design lessons

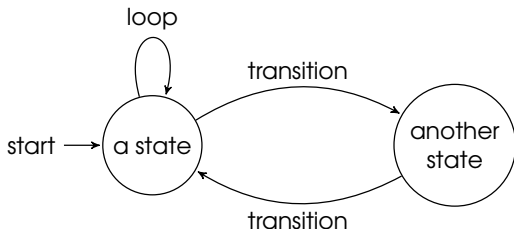
- ▶ AI doesn't have to be complicated
- ▶ Simple AI, when interacting with a player and each other, can give engaging results
- ▶ Bugs in AI don't always matter...

Finite state machines

Finite state machines

- ▶ A **finite state machine (FSM)** consists of:
 - ▶ A set of **states**; and
 - ▶ **Transitions** between states
- ▶ At any given time, the FSM is in a **single state**
- ▶ **Inputs** or **events** can cause the FSM to transition to a different state

State transition diagrams

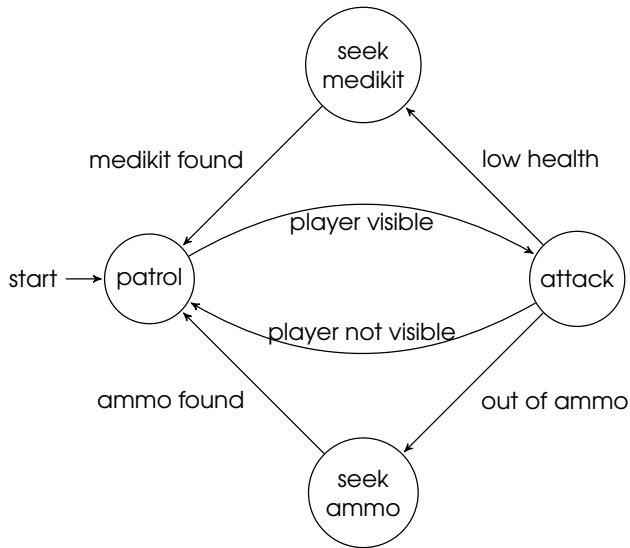


- ▶ FSMs are often drawn as **state transition diagrams**
- ▶ Reminiscent of **flowcharts** and certain types of **UML diagram**

FSMs for AI behaviour

The next slide shows a simple FSM for the following AI behaviour, for an enemy NPC in a shooter game:

- ▶ By default, patrol (e.g. along a preset route)
- ▶ If the player is spotted, attack them
- ▶ If the player is no longer visible, resume patrolling
- ▶ If you are low on health, run away and find a medikit.
Then resume patrolling
- ▶ If you are low on ammo, run away and find ammo.
Then resume patrolling



Other uses of FSMs

As well as AI behaviours, FSMs may also be used for:

- ▶ Animation
- ▶ UI menu systems
- ▶ Dialogue trees
- ▶ Token parsing
- ▶ ...

Beyond FSMs

Some topics for you to research, for when plain old FSMs aren't enough...

- ▶ Hierarchical FSMs
- ▶ Nested FSMs
- ▶ Stack-based FSMs
- ▶ Hierarchical task networks
- ▶ ...

Behaviour Trees

Behaviour trees (BTs)

- ▶ A **hierarchical** model of decision making
- ▶ Allow **complex behaviours** to be built up from **simple components**
- ▶ Allow for **more complex** behaviours than FSMs
- ▶ First used in Halo 2 (2005), now used extensively
- ▶ Also used in robotics and other non-game AI applications

Using BTs

- ▶ Fairly easy to implement; plenty of resources online
- ▶ **Unreal**: an advanced BT system is built in
- ▶ **Unity**: numerous free and paid options on the Asset Store e.g. Behavior Machine, Behavior Designer, Behave, RAIN

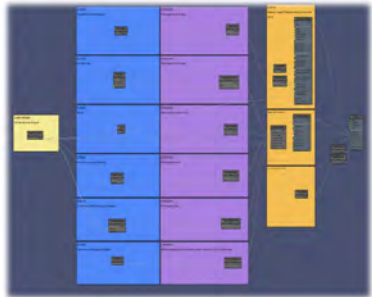
BT basics

- ▶ A BT is a **tree** of **nodes**
- ▶ On each game update (i.e. each frame), the root node is **ticked**
 - ▶ When a node is ticked, it might cause some or all of its **children** to tick as well
 - ▶ So ticks propagate down the tree from the root
- ▶ A ticked node returns one of three **statuses**:
 - ▶ Success
 - ▶ Running
 - ▶ Failure
- ▶ “Running” status allows nodes to represent operations that **last multiple frames**

Blackboard

- ▶ It is often useful to **share** data between nodes
- ▶ A **blackboard** (sometimes called a **data context**) allows this
- ▶ Blackboard defines **variables**, which can be **read** and **written** by nodes
- ▶ Blackboard can be **local** to the AI agent, **shared** between several agents, or **global** to all agents
- ▶ (Shared blackboards mean that your AI has “telepathy” — this may or may not be desirable!)

BTs in The Division



[http://www.gdcvault.com/play/1023382/
AI-Behavior-Editing-and-Debugging](http://www.gdcvault.com/play/1023382/AI-Behavior-Editing-and-Debugging)