

Grao en Enxeñaría Informática  
Universidade de Santiago de Compostela

Autor: Rubén Osorio López

Titor: Manuel Mucientes Molina  
Cotitor: Pablo Rodríguez Mier

21 de xullo de 2017

Esto é a defensa da memoria do traballo de fin de grado nombrado **Intelixencia Artificial aplicada a Videoxogos Top-Down en tempo real**, eu son o autor, Rubén Osorio López, e os tutores son Manuel Mucientes Molina e Pablo Rodríguez Mier

## Táboa de contidos

- 1 Introdución
- 2 Videoxogo baseado en axentes
- 3 Análise de requisitos
- 4 Xestión do proxecto
- 5 Arquitectura
- 6 Validación e probas
- 7 Conclusións

2017-07-19

## Intelixencia Artificial aplicada a Videoxogos Top-Down en tempo real

### └ Táboa de contidos

Durante esta presentación seguiremos unha estrutura similar á memoria, centrándonos máis en algúns aspectos concretos do proxecto que expliquen en que consistiu o traballo realizado.

#### Táboa de contidos

- 1 Introdución
- 2 Videoxogo baseado en axentes
- 3 Análise de requisitos
- 4 Xestión do proxecto
- 5 Arquitectura
- 6 Validación e probas
- 7 Conclusións

- Proxecto que aborda a creación dun **videoxogo** con necesidades de **comportamento complexo** por parte do inimigo.

## Loita 1 contra 1, Top-Down en dúas dimensións

Capaz de percibir e actuar sobre o **entorno competitivo** do  
videoxogo mediante **sensores** e **actuadores**

2017-07-19

## └ Introducción

De forma xeral, búscase a creación dun videoxogo que requira un inimigo con comportamento complexo. O axente que representará o inimigo necesita ser un competidor capaz, para o que se realizou unha etapa de entrenamento na que optivo a información que necesitaba.

**Loita 1 contra 1** significa que soamente dous perxonaxes competirán entre eles contando ambos coas mesmas capacidades, accións posibles e atributos. **Top-Down** refírese ó plano picado utilizado para visualizar o combate. Por outra parte que sea en **dúas dimensións** implica que todo o contido do videoxogo son imaxes planas debuxadas unha a unha, sen que existan modelos en tres dimensións.

**Un axente** é aquilo capaz de percibir o entorno mediante **sensores** e actuar sobre o mesmo en consecuencia mediante **actuadores**, ambos son proporcionados pola súa interface co videoxogo. Ademáis atoparase nun entorno competitivo o que implica que buscará maximizar o seu **rendemento** mentres se minimiza o do contrincante.

- Proyecto que aborda a creación dun **videoxogo** con necesidades de **comportamento complexo** por parte do inimigo.

Videorego

Loita 1 contra 1. Top-Down en dues dimensions

## Article

Capaz de percibir e actuar sobre o entorno competitivo do videoxogo mediante sensores e actuadores

## Objetivos

- Implementación do videoxogo: Xa que necesitaremos unha plataforma que nos permita que o axente e o xogador interactúen seguindo unha serie de regras comúns para competir entre eles.
- Implementación do axente: Necesitarase implementar o axente capaz de desenvolverse correctamente durante a competición.
- Realizar o entrenamento do axente: O axente necesitará obter a información necesaria para logo comportarse adecuadamente gracias ó aprendido durante a etapa de entrenamento.
- Obter datos sobre as capacidades do axente: Obteráanse datos sobre o rendemento do axente contra outras implementacións máis sinxelas.
- Analizar os resultados obtidos: Coa información obtida durante todo o proxecto, e especialmente na etapa anterior, realizarase un análise que describa o que conseguiu o axente.

## Objetivos

- Implementación do videoxogo

2017-07-19

# Inteligencia Artificial aplicada a Videoxogos Top-Down en tempo real

- └ Introducción
- └ Obxectivos
- └ Obxectivos

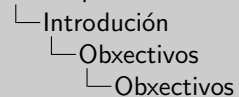
- Implementación do videoxogo: Xa que necesitaremos unha plataforma que nos permita que o axente e o xogador interactúen seguindo unha serie de regras comúns para competir entre eles.
- Implementación do axente: Necesitarase implementar o axente capaz de desenvolverse correctamente durante a competición.
- Realizar o entrenamento do axente: O axente necesitará obter a información necesaria para logo comportarse adecuadamente gracias ó aprendido durante a etapa de entrenamento.
- Obter datos sobre as capacidades do axente: Obteráanse datos sobre o rendemento do axente contra outras implementacións máis sinxelas.
- Analizar os resultados obtidos: Coa información obtida durante todo o proxecto, e especialmente na etapa anterior, realizarase un análise que describa o que conseguiu o axente.

## Objetivos

- Implementación do videoxogo
- Implementación do axente

2017-07-19

# Inteligencia Artificial aplicada a Videoxogos Top-Down en tempo real



- Implementación de videojogo
- Implementación de agente

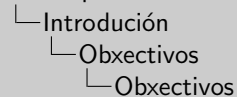
- Implementación do videoxogo: Xa que necesitaremos unha plataforma que nos permita que o axente e o xogador interactúen seguindo unha serie de regras comúns para competir entre eles.
- Implementación do axente: Necesitarase implementar o axente capaz de desenvolverse correctamente durante a competición.
- Realizar o entrenamento do axente: O axente necesitará obter a información necesaria para logo comportarse adecuadamente gracias ó aprendido durante a etapa de entrenamento.
- Obter datos sobre as capacidades do axente: Obteráanse datos sobre o rendemento do axente contra outras implementacións máis sinxelas.
- Analizar os resultados obtidos: Coa información obtida durante todo o proxecto, e especialmente na etapa anterior, realizarase un análise que describa o que conseguiu o axente.

## Objetivos

- Implementación do videoxogo
- Implementación do axente
- Realizar o adestramento do axente

2017-07-19

# Inteligencia Artificial aplicada a Videoxogos Top-Down en tempo real



- Implementación do videocxogo
- Implementación do axente
- Realizar o adestramento do axente

- Implementación do videoxogo: Xa que necesitaremos unha plataforma que nos permita que o axente e o xogador interactúen seguindo unha serie de regras comúns para competir entre eles.
- Implementación do axente: Necesitarase implementar o axente capaz de desenvolverse correctamente durante a competición.
- Realizar o entrenamento do axente: O axente necesitará obter a información necesaria para logo comportarse adecuadamente gracias ó aprendido durante a etapa de entrenamento.
- Obter datos sobre as capacidades do axente: Obteráanse datos sobre o rendemento do axente contra outras implementacións máis sinxelas.
- Analizar os resultados obtidos: Coa información obtida durante todo o proxecto, e especialmente na etapa anterior, realizarase un análise que describa o que conseguiu o axente.

## 2017-07-19

### Objetivo

- Implementación do videoxogo
- Implementación do axente
- Realizar o adestramento do axente
- Obter datos sobre as capacidades do axente

- Implementación do videoxogo
- Implementación do axente
- Realizar o adestramento do axente
- Obter datos sobre as capacidades do axente

- Implementación do videoxogo: Xa que necesitaremos unha plataforma que nos permita que o axente e o xogador interactúen seguindo unha serie de regras comúns para competir entre eles.
- Implementación do axente: Necesitarase implementar o axente capaz de desenvolverse correctamente durante a competición.
- Realizar o entrenamento do axente: O axente necesitará obter a información necesaria para logo comportarse adecuadamente gracias ó aprendido durante a etapa de entrenamento.
- Obter datos sobre as capacidades do axente: Obteráanse datos sobre o rendemento do axente contra outras implementacións máis sinxelas.
- Analizar os resultados obtidos: Coa información obtida durante todo o proxecto, e especialmente na etapa anterior, realizarase un análise que describa o que conseguiu o axente.

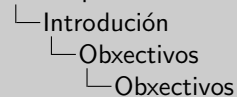


## Objetivos

- Implementación do videoxogo
- Implementación do axente
- Realizar o adestramento do axente
- Obter datos sobre as capacidades do axente
- Analizar os resultados obtidos

2017-07-19

# Inteligencia Artificial aplicada a Videoxogos Top-Down en tempo real



- Implementación de videoxogo
- Implementación do axente
- Realizar o adestramento do axente
- Obter datos sobre as capacidades do axente
- Analizar os resultados obtidos

- Implementación do videoxogo: Xa que necesitaremos unha plataforma que nos permita que o axente e o xogador interactúen seguindo unha serie de regras comúns para competir entre eles.
- Implementación do axente: Necesitarase implementar o axente capaz de desenvolverse correctamente durante a competición.
- Realizar o entrenamento do axente: O axente necesitará obter a información necesaria para logo comportarse adecuadamente gracias ó aprendido durante a etapa de entrenamento.
- Obter datos sobre as capacidades do axente: Obteráanse datos sobre o rendemento do axente contra outras implementacións máis sinxelas.
- Analizar os resultados obtidos: Coa información obtida durante todo o proxecto, e especialmente na etapa anterior, realizarase un análise que describa o que conseguiu o axente.

## 2017-07-19

- └ Introducción
- └ Obxectivos
- └ Obxectivos

## Objetivos

- Implementación do videoxogo
- Implementación do axente
- Realizar o adestramento do axente
- Obter datos sobre as capacidades do axente
- Analizar os resultados obtidos

- Implementación do videoxogo
- Implementación do axente
- Realizar o adestramento do axente
- Obter datos sobre as capacidades do axente
- Analizar os resultados obtidos

- Implementación do videoxogo: Xa que necesitaremos unha plataforma que nos permita que o axente e o xogador interactúen seguindo unha serie de regras comúns para competir entre eles.
- Implementación do axente: Necesitarase implementar o axente capaz de desenvolverse correctamente durante a competición.
- Realizar o entrenamento do axente: O axente necesitará obter a información necesaria para logo comportarse adecuadamente gracias ó aprendido durante a etapa de entrenamento.
- Obter datos sobre as capacidades do axente: Obteráanse datos sobre o rendemento do axente contra outras implementacións máis sinxelas.
- Analizar os resultados obtidos: Coa información obtida durante todo o proxecto, e especialmente na etapa anterior, realizarase un análise que describa o que conseguiu o axente.

# Mecánicas

## Movimento

Movimento libre nunha habitación rectangular.

## Ataque

Permítese atacar a zona que se atopa cada onde o personaxe está mirando.

## Defensa

Posibilidade de defenderse dun ataque permitindo atacar se a defensa ten éxito

2017-07-19

# Inteligencia Artificial aplicada a Videoxogos Top-Down en tempo real

- Videoxogo baseado en axentes

– Mecánicas

– Mecánicas

## Mecánicas

## Movements

Movimiento libre nunha habitación rectangular.

## Ataque

Permite-se atacar a zona que se atopa cada onde o personaxe está mirando.

## Defens

Posibilidade de defenderse dun ataque permitindo atacar se a  
defensa ten éxito

**Movement** libre nunha habitación rectangular que suma a complexidade de evitar situacións nas que non se poida escapar do contrincante por estar ó lado dunha parede ou unha esquina. Ademáis a única maneira de mirar cara unha dirección é mirar cara ela.

**Como** solo se permite atacar a zona directamente enfrente do personaxe é importante ter en conta cada donde se está mirando. Isto favorece unha actitude agresiva pois hai que moverse na dirección do enemigo antes de atacalo.

**Pódese** realizar unha manobra defensiva de alto risco e alta recompensa que permite evitar un ataque. Se se evita con éxito poderase realizar un ataque propio pero se non serase vulnerable durante uns instantes.

**Esto fai que non exista unha estratexia idónea** pois un estilo agresivo perde contra un defensivo que á súa vez perde contra xogadores que busquen a contra do movemento defensivo, este último ademáis perde contra o xogador agresivo. Esta fórmula de **pedra, papel, tesoiras** demostrou ser amplamente utilizada en diseño de videoxogos.

Primeira implementación realizada con **Unity3D**, estándar de facto para videoxogos de este tamaño.

## Problemas de simulación

Impossibilidade de escalar o tempo sem romper o funcionamento do videoxogo.

2017-07-19

# Intel·ligència Artificial aplicada a Videojocs Top-Down en temps real

- └ Videoxogo baseado en axentes
  - └ Prototipo de Unity
    - └ Prototipo de Unity

## Ensinar vídeo

Primeira implementación realizada con **Unity3D**, estándar de facto para videoxogos de este tamaño.

### Problemas de simulación

Impossibilidade de escalar o tempo sem romper o funcionamento do vídeo-xogo.

# Implementación de un motor desde cero en C++



2017-07-19

# Intel·ligència Artificial aplicada a Videojocs Top-Down en temps real

- └ Videoxogo baseado en axentes
  - └ Segunda aplicación
    - └ Segunda aplicación



## Algoritmo

```
1 while agent is running do
2   lastState ← currentState;
3   currentState ← getCurrentState();
4   deltaFitness ←
5     calculateFitness(currentState) - calculateFitness(lastState);
6   if lastState ∈ stateActionData then
7     stateActionData .updateWith(lastState,selectedAction,deltaFitness);
8   else
9     stateActionData .insert (lastState,selectedAction,deltaFitness);
10  if currentState ∈ stateActionData then
11    if randomBetween(0,1) <  $\epsilon$  then
12      selectedAction ← randomAction ∈ allPossibleActions;
13    else
14      selectedAction ← action ∈ allPossibleActions |
15        bestWeightedAction(stateActionData,currentState) = action;
16  else
17    selectedAction ← randomAction ∈ allPossibleActions;
```

2017-07-19

## Intelixencia Artificial aplicada a Videoxogos Top-Down en tempo real

- └ Videoxogo baseado en axentes
  - └ Algoritmo
    - └ Algoritmo

Algoritmo

```
1 while agent is running do
2   lastState ← currentState;
3   currentState ← getCurrentState();
4   deltaFitness ←
5     calculateFitness(currentState) - calculateFitness(lastState);
6   if lastState ∈ stateActionData then
7     stateActionData .updateWith(lastState,selectedAction,deltaFitness);
8   else
9     stateActionData .insert (lastState,selectedAction,deltaFitness);
10  if currentState ∈ stateActionData then
11    if randomBetween(0,1) <  $\epsilon$  then
12      selectedAction ← randomAction ∈ allPossibleActions;
13    else
14      selectedAction ← action ∈ allPossibleActions |
15        bestWeightedAction(stateActionData,currentState) = action;
16  else
17    selectedAction ← randomAction ∈ allPossibleActions;
```

## Fitness

```

input : playerHealth, enemyHealth, distance,
        lookingAtEnemy, noWallsNear
output: fitness
1 fitness ← INITIAL_FITNESS_VALUE;
2 fitness ← fitness +(playerHealth *
  MY_HEALTH_MULTIPLIER);
3 fitness ← fitness -(enemyHealth *
  ENEMY_HEALTH_MULTIPLIER);
4 fitness ← fitness -(distance *
  DISTANCE_MULTIPLIER);
5 if lookingAtEnemy then
6 | fitness ← fitness + LOOKING_BONUS;
7 if noWallsNear then
8 | fitness ← fitness + WALL_BONUS;
  
```

Parámetro	Valor
INITIAL_FITNESS_VALUE	1000
MY_HEALTH_MULTIPLIER	100
ENEMY_HEALTH_MULTIPLIER	100
DISTANCE_MULTIPLIER	3
LOOKING_BONUS	200
WALL_BONUS	50

2017-07-19

## Intelixencia Artificial aplicada a Videoxogos Top-Down en tempo real

└ Videoxogo baseado en axentes

└ Algoritmo

└ *Fitness*

Fitness

Parámetro	Valor
INITIAL_FITNESS_VALUE	1000
MY_HEALTH_MULTIPLIER	100
ENEMY_HEALTH_MULTIPLIER	100
DISTANCE_MULTIPLIER	3
LOOKING_BONUS	200
WALL_BONUS	50

2017-07-19

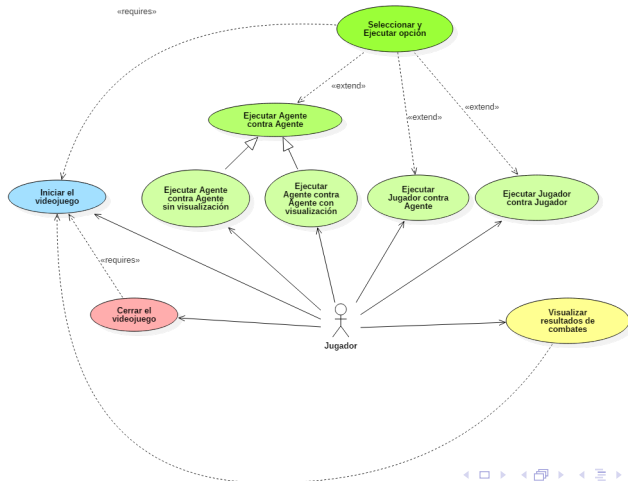
- └ Videoxogo baseado en axentes
  - └ Tipos de adestramento
    - └ Tipos de adestramento

Pretende simular un aprendizaxe contra xogadores reais.

Buscando unha exploración mais extensa de estados que o axente baseado en regras non pode aportar.



## Casos de uso



2017-07-19

- Intelixencia Artificial aplicada a Videoxogos Top-Down en tempo real
  - └─Análise de requisitos
    - └─Casos de uso
      - └─Casos de uso



2017-07-19

- └─Análise de requisitos
  - └─Requisitos
    - └─Requisitos

### Requisitos

- |  |                                     |
|--|-------------------------------------|
| • RF-1/2/3: Funcionalidades do menu        | • RNF-1: Redimensionar da aplicação |
| • RF-4/1/5: Consola com comandos/feedback  | • RNF-2: Velocidade das simulações  |
| • RF-5: Salva a aplicação                  | • RNF-3: Extensibilidade do motor   |
| • RF-6: Entra na ecena de combate          | • RNF-4: Facilidade para depurar    |
| • RF-7/8/9: Mausem/Ataca/Defender          | • RNF-5: Aplicação autônoma         |
| • RF-10: Calar/Poder paratir               | • RNF-6: Extensibilidade de ecenas  |
| • RF-11: Esgotar o tempo de combate        | • RNF-7: Documentação               |
| • RF-12: Volver o menu                     | • RNF-8: Usabilidade da interface   |
| • RF-13: Visualizar combate entre asesores |                                     |
| • RF-14: Simular mltiplos combates         |                                     |

- ◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡

# Metodoloxía

## Contexto do proxecto

- Traballador único
- Duración relativamente corta
- Necesidade de avanzar rapidamente nas etapas iniciais

## Programación Extrema

- Flexibilidade ante cambios
- Evitase utilizar demasiado tempo en tarefas de xestión
- Rápida iteración
- Reunións entre *sprints*

2017-07-19

# Inteligencia Artificial aplicada a Videoxogos Top-Down en tempo real

- └ Xestión do proxecto
  - └ Metodoloxía
    - └ Metodoloxía

## Contexto do proxecto

- Traballador único
- Duración relativamente curta
- Necesidade de avanzar rapidamente nas etapas iniciais

## Programación Extrema

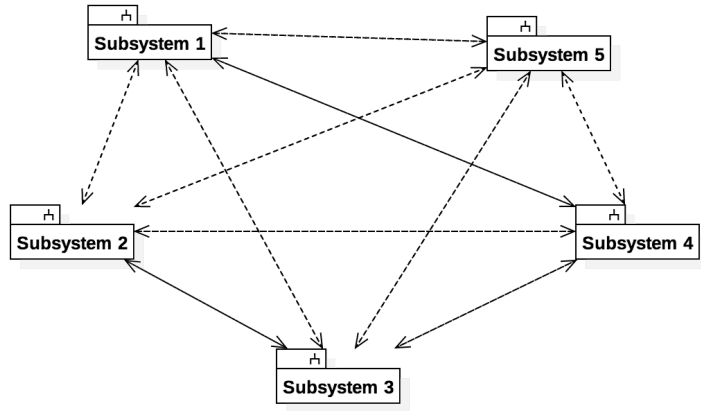
- ✓ Flexibilidad ante cambios
- ✓ Evitase utilizar demasiado tiempo en tareas de gestión
- ✓ Rápida iteración
- ✓ Reuniones entre sprints

2017-07-19

- └ Xestión do proxecto
  - └ Planificación temporal
    - └ Planificación temporal



## Subsistemas conectados

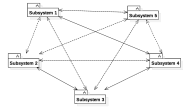


2017-07-19

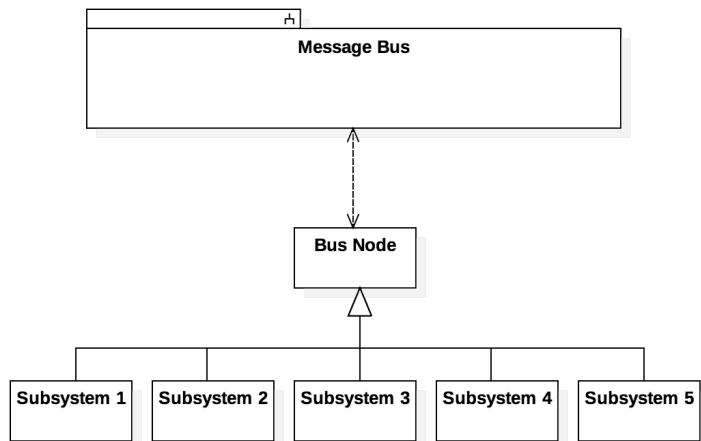
# Intelixencia Artificial aplicada a Videoxogos Top-Down en tempo real

- └─Arquitetura
  - └─Arquitetura do sistema
    - └─Subsistemas conectados

## Subsistemas conectados



## Bus de mensaxes



2017-07-19

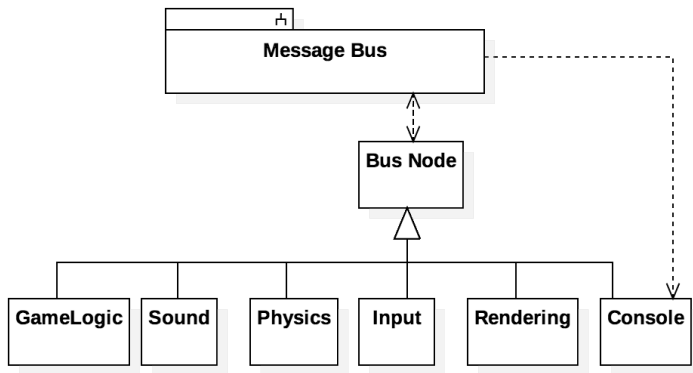
Intelixencia Artificial aplicada a Videoxogos Top-Down  
en tempo real

- └─ Arquitectura
  - └─ Arquitectura do sistema
    - └─ Bus de mensaxes

Bus de mensaxes



## Arquitectura final

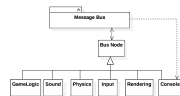


2017-07-19

Intelixencia Artificial aplicada a Videoxogos Top-Down  
en tempo real

- └─ Arquitectura
  - └─ Arquitectura do sistema
    - └─ Arquitectura final

Arquitectura final



2017-07-19

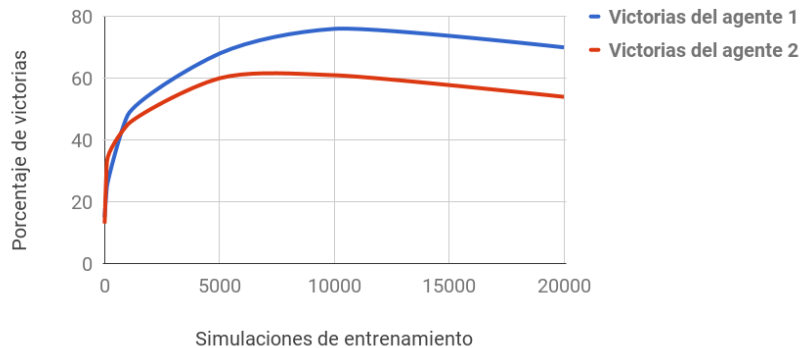
- └ Validación e probas
  - └ Aplicación
    - └ Validación e probas da aplicación

Unha ou mais probas por cada requisito tanto funcional como non funcional superadas na súa totalidade.

Comproban a integración entre subsistemas e do o axente ca aplicación.



## Comparativa de vitorias

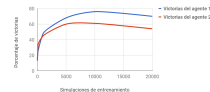


2017-07-19

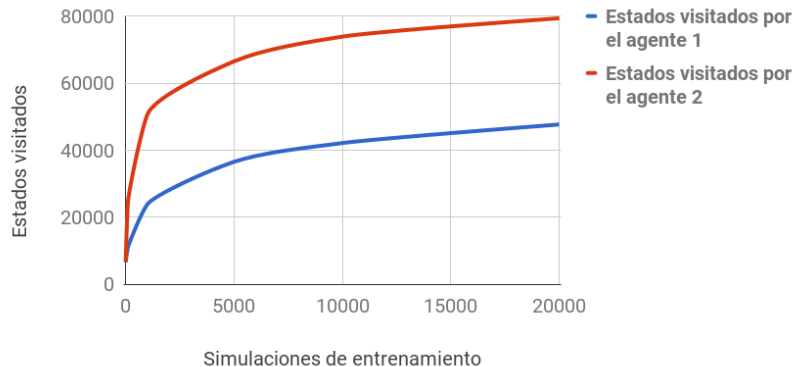
Intelixencia Artificial aplicada a Videoxogos Top-Down en tempo real

- Validación e probas
  - Validación do Axente
    - Comparativa de vitorias

Comparativa de vitorias



## Comparativa de estados visitados

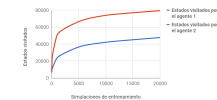


2017-07-19

Intelixencia Artificial aplicada a Videoxogos Top-Down en tempo real

- Validación e probas
  - Validación do Axente
    - Comparativa de estados visitados

Comparativa de estados visitados



## Axente escollido

## Combinación de ámbolos dous métodos de adestramento.

## Contra o axente baseado en regras

Favorece un aprendizaxe moi rápido nas primeiras simulacións.

## Contra él mismo

Aporta unha exploración de estados superior.

2017-07-19

# Inteligencia Artificial aplicada a Videoxogos Top-Down en tempo real

- Validación e pruebas

## Validación do Axente

└ Axente escollido

Axente escollido

#### Combinación de ámbolos dos métodos de adestramento

Contra o axente baseado en regras

Favorece un aprendizaxe moi rápido nas primeiras simulacións.

Contra el mismo

Aporta unha exploración de estados superior.

## Conclusións e leccións aprendidas

### Logros do proxecto

- O comportamento, aspecto e rendemento da aplicación cumpriu as expectativas.
- O axente é capaz de competir contra outras implementacións e contra xogadores humanos.

### Leccións aprendidas

- Importancia de ter en conta os posibles riscos do proxecto o antes posible.
- Utilidade de un deseño flexible previo á implementación.
- Calidade dos resultados de implementacións sinxelas de Intelixencia Artificial.

2017-07-19

## Intelixencia Artificial aplicada a Videoxogos Top-Down en tempo real

### Conclusións

#### Conclusións e leccións aprendidas

#### Conclusións e leccións aprendidas

### Conclusións e leccións aprendidas

#### Logros do proxecto

- O comportamento, aspecto e rendemento da aplicación cumpriu as expectativas.
- O axente é capaz de competir contra outras implementacións e contra xogadores humanos.

#### Leccións aprendidas

- Importancia de ter en conta os posibles riscos do proxecto o antes posible.
- Utilidade de un deseño flexible previo á implementación.
- Calidade dos resultados de implementacións sinxelas de Intelixencia Artificial.

## Posibles ampliaciones

- Melloras de compatibilidade.
- Ampliación do proceso de probas con xogadores humanos.
- Implementación de máis técnicas para o axente.
- Novas mecánicas para o videoxogo.

2017-07-19

# Intel·ligència Artificial aplicada a Videojocs Top-Down en temps real

## – Conclusiones

## Posibles ampliaciones

└ Posibles ampliaciones

- Melloras de compatibilidade.
- Ampliación do proceso de probas con xogadores humanos
- Implementación de máis técnicas para o axente.
- Novas mecánicas para o videoxogo.