Knowledge-based Authentic AI for a Open-World Game

CSC 581: Knowledge and Usability • May 22, 2014
Draft

Ian Dunn
California Polytechnic State University
San Luis Obispo, CA
ian@iondune.com

Abstract

Knowledge representation is a fine candidate for implementing a complex and authentic artificial intelligence system for a game, yet many current game systems avoid knowledge-based systems in favor of more traditional artificial intelligence systems and hacks. We explore the flaws of many simple AI systems and discuss the implementation of a knowledge-based authentic AI system for an open-world sandbox game.

Contents

1 Introduction

As modern games push the envelope by bringing more and more realism to the table, a need for highly sophisticated AI has emerged. However,

1.1 Overview

[To Do]

2 Background

[To Do]

2.1 Theory

[To Do]

2.2 Halo

[To Do]

3 Existing Systems

3.1 World

The herein described artificial intelligence system is meant for use in developing an open-world sandbox game.

Editability

One facet of this game is that random procedural generation is used for creation of the world, environment, and all characters, structures, and objects within the game. The result of this stipulation is that no manual level design controls may be used. The artificial intelligence system must be implemented on an entirely automated platform. On the one hand, this appears to impose an unfortunate limitation on the design of the system. It is actually, however, conducive to our goals. Consider:

- In an open-world/sandbox environment, the player is free to make modifications to the world. Pre-calculated or designed elements would therefore be invalidated. It is easier to formulate a system that establishes all of its system elements given at least partially arbitrary input than to attempt to recalculate on the fly based on similarly arbitrary modification.

- Adding design-ability as a requirement to an artificial intelligence system is actually an encumbrance - see the above seven principles of system design and the Halo AI paper.[Dyckhoff, 2008]

Therefore, as one of our system goals we will not require any manual controls or editors available to designers.

Format

The format of the world will be a volumetric implicit model so that non-functional features such as cliffs, overhangs, and caves can be easily supported. There will be an additional hierarchical model of objects which exist on and in the terrain.

Significant level-of-detail algorithms will be required to support the rendering of such constructs at any significant scale. Similarly, those same level-of- detail aspects will need to be incorporated into the design of the AI system to support the same scale.

Physics and Collision

Obviously a sophisticated system will be needed to support the rigid-body physics and intelligent collision detection required to implement such a game. I will leave the details of such an implementation out of this paper; however, we will assume some required aspects in order to make a discussion about the more interesting AI features possible. Those requirements are:

- Line-of-sight/visibility detection: From arbitrary points in the environment, we should be able to query in reasonable time both a visibility check of a specific object and a general collection of objects which are within line-of-sight.
- Feedback navigation: Controllable actors should specify their movement as a velocity and direction. Continual feedback must be available in the form of a scalar representing movement success. Zero or close-to-zero values represent entirely inhibited movement while large values indicate partially inhibited or entirely uninhibited movement.
- Basic object circumnavigation: If movement of an actor is inhibited by any obstruction which could practically be stepped over or climbed, it should immediately engage such an action, noting the reduce speed in the previously described feedback value.
- Lay-of-the-land survey: Tying into the circumnavigation and line-of-sight principles, the AI system should be able to survey an area from a particular vantage point and make a reasonable estimation of which areas are impassable and which may likely slow down the actor. Note that this is distinct from a path-finding system. Instead, a rough heat-map is generated.

The reader may note that I have specifically left out path-finding from the list of required features. Path-finding is, however, often found as an integrated feature in a physics or collision engine. It is specifically left out from here. I make this omission because I find existing techniques for path-finding fundamentally flawed. See further elaboration in section 4.1.

3.2 Game

It goes without saying that some game system must exist on which to build this AI system. The AI system must be tightly integrated with the game components, and I will therefore be introducing some components as part of the following discourse on the system design.

However, some general ground rules should be established. The game consists of a large number of characters who interact via standard interactions such as talking and trading along with some non-standard interactions such as combat and negotiation. At the core level, each character is mostly represented by its health, position, emotional condition, and whatever knowledge it has about the environment it is in. Obviously the last two items are rather loaded; their specific definition is described later in the paper.

4 Problems with Traditional AI Systems

There a number of issues present in many current game AI systems, even among AAA titles touted for their advanced AI. Not all of the following issues exist in all titles, but they are certainly common issues. I believe these problems have become prevalent largely because of two reasons:

- Hack culture: A lot of the game industry is driven by what I like to call “hack culture”. The need for incredible performance in the face of limited hardware and a feature-hungry customer base makes a lot of game design about not “how well can we do it” but “how well can we fake it”. This is especially prevalent within graphic development.
- Suspension of disbelief: The nature of games often requires a certain level of suspension of disbelief. Nobody ever thinks to question what business a plumber has jumping on turtles and rescuing a mushroom princess from a dinosaur, though nearly everyone plays and enjoys the Super Mario games. Users expect games to follow a very specific set of rules (arguably, that is the definition of a game) rather than exactly mimic something from reality. Arguably, this distinction marks the fine line between simulation and game. However, the introduction of sandbox and open-world games starts to blur this line. Certainly there are some features which should remain fantastical, but there are others where a more realistic touch is appropriate. I believe that AI should be considered among the latter.

4.1 Uncanny Knowledge

Often, AI characters will behave in ways that are only possible through the existing of an omniscient knowledge base. I classify such behaviour as the result of “uncanny knowledge” since it results in characters behaving as though they have incredibly precise knowledge of some things we may expect to be unknown.

Psychic Guards

Perhaps the most notable and plain example of this issue is from *The Elder Scrolls IV: Oblivion*, a 2007 open-world sandbox RPG set in a medieval/fantasy world. An oft-complained topic from the heyday of this game were the so-called "psychic guards". Players would quietly commit a crime in some dark, secluded corner of the world, only to have guards instantly burst into the room to make an arrest.

Path finding

A more commonplace and ubiquitous example of this issue comes in the form of standard path-finding algorithms. Games with good path-finding are touted for the ability of characters to navigate from point A to point B successfully and without getting stuck or otherwise halted. The problem exhibited here is that characters somehow have omniscient knowledge of the environment. The same algorithms that let a character walk through the streets of a well-known town will lead that same character out of an unknown labyrinth as though they had built it themselves.

System Knowledge vs. Character Knowledge

This problem manifests itself from a failure to classify system knowledge as opposed to character knowledge. If properly implemented, the AI system should know equally well how to navigate out of the labyrinth as it does the streets of a city. But from a character's perspective, these two knowledge sets are entirely disparate. This, I believe, is the fundamental flaw with common path-finding systems. Most work by decomposing the entire world into some sort of graph data structure and performing an offline search. This can produce lifelike behaviour when the environment is simple or otherwise expected to be known by a character, but makes little sense in the context of true knowledge representation.

Another name for system knowledge is the "canonical view".

In this described system, we discard the concept of system knowledge entirely. This serves to solve the uncanny knowledge problem, but also comes with an additional added benefit. As the system becomes more complex, it becomes more and more difficult to keep an updated and accurate representation of the system knowledge. Moreover, such a representation becomes increasingly useless when a large number of characters each have their own differing perspectives of the world.

One hallmark argument in favor of system knowledge, however, is memory. Supporting knowledge representation for each and every character is likely to exceed possible memory capacity in short order and is definitely not a scalable solution. Considerable focus on the remainder of this paper will center on establishing data structures and compression methods for reducing the memory requirements of a character knowledge implementation.

4.2 Oblivious Bystander

Another common issues from modern AI systems is what I call the "oblivious bystander" effect. This problem manifests itself in two different ways. First, there are systems in which characters remain entirely unaware or at least unresponsive to events that occur around them as long as they are not specifically targeted. Consider a mugging or assassination occurring in this streets while characters walk idly by as though nothing is happening. Systems exhibiting this behaviour lack a model for determining whether events are visible to characters, or lack a model for determining character responses to arbitrary events.

Even games which seem to implement appropriate bystander AI, however, are often still guilty of a slightly more subtle version of the oblivious bystander effect. As an example, characters in *Grand Theft Auto V* will react by running in fear if they are near to scene of a crime. However, this is really just an example of a system in which characters respond to events that tag them. A discrete crime event occurs, so all characters in a specified area react. Unless the event is of a predetermined and clearly defined sort, characters will not respond to it. Non-scripted events do not produce such reactions. If the player behaves absurdly but in an unscripted manner, character reactions do not occur or are nonspecific.

4.3 Canonical View

As previously discussed as an issue related to uncanny knowledge and path finding, the canonical view or system knowledge is the presence of knowledge which is gathered by the omniscient system and is considered absolute. Such knowledge poses problems in both acquisition and utilization. First, rules must exist for establishing the exact interpretation of the existing world state. Second, rules must exist for applying that data to individual characters with unique perspectives.

It is difficult to determine for a given world how to evaluate such criteria without introducing a natural bias.

5 Solutions

In the following paragraphs, I outline my solutions to the previously described problems and, generally, introduce the principles and systems of the implementation.

5.1 Bigger-Picture Model

The bigger-picture model is the system used to allow scalability in the described AI system. The basic principle of the system is that each character must have its own knowledge model in order to correctly simulate complex scenarios. The quality of such a simulation is directly linked to the size of the knowledge representation utilized. We therefore have a memory problem, and a system which is entirely unscalable.

Local Events

To address this issue, we utilize what is called the "bigger-picture" model. Consider a simulation of a continent which contains many nations, many cities, and many citizens. Our simulation consists of players who are experiencing and are involved in the world. For the characters nearby the player, we must consider the knowledge model of each individual in order to accurately portray the situation. Two peddlers are making a trade, so we must analyze their motivations and emotions so that an interaction can take place. Perhaps a fight breaks out, and nearby characters must intervene to break it up. All of this simulation must occur at a finite scale so that it can be observed by the player. The thoughts, emotions, and actions of each involved individual are important to the accuracy of the simulation.

Remote Events

Consider now a similar fight taking place three blocks over from the player. The player's interaction with this event is far more limited. Perhaps if the fight becomes loud the player will be able to hear it happening. Maybe the player will go down that street at a later time and hear talk of what occurred. All of these interactions occur at a different level of abstraction than the previously discussed events.

While the events that occur around the player may directly involve them, the events that occur at a slightly more remove level will often only indirectly involve the player. We may consider this a difference of one level of indirection.

Of course it is always possible that these remote events may trigger events with local consequences. It may even be possible that individual interactions are key to determining whether such an escalation takes place.

Consider that the two arguing peddlers may happen to be of two different rival clans. The argument may now escalate into a brawl, and from there into a riot. Soon the player has become directly involved in this interaction since the growing scale of a riot may engulf them.

However, if we can devise a general model for representing these remote events at an abstract level such that only the important elements are observed – either through ambient effects, historic significance, or escalation into a larger event with more widespread effect; then, we have introduced a scheme for reducing the memory and computation overhead of representing extensive knowledge structures on a large scale.

Another way to consider this model is to think in terms of "leak". Interactions are only important in the effects they have on the player. Events that take place in close proximity leak nearly everything since all aspects of the actions are directly observable. We must therefore employ a fully or nearly fully accurate model of simulation to this event. However, a remote event leaks only some information - only the aspects which may directly affect the player or indirectly produce some cause at a later date. We may therefore use a less specific and detailed simulation of the event. Interactions which are event more remote leak less volume and less often. Part of the bigger picture model relies on accurately determining what information leaks from a given interaction.

Lazy Evaluation

Our scheme for implementing the bigger-picture model is similar to the concept from programming language design called "Lazy Evaluation."

Terminology

We devise the following terms for representing the concrete relationships between events.

Interactions: The smallest composable piece of temporal information in the AI system. Consists of any two characters and some exchange of information or material. Note that the exchange of information described can be applied at a highly abstract level. Two characters sharing a glance as they walk through the street is considered an interaction.

Events: A composition of interactions. Some representation for determining related interactions and combining them into a single element. An event is also hierarchical - events can be combined to form a single, larger event.

Level of Indirection: The distinction between events which require one defined abstraction versus a lower or higher abstraction. When multiple events are composed into a larger event, that larger event has a singularly higher level of indirection. This classification is vital to the performance of the system. As level of indirection increases, the level of detail required for simulation decreases, as does the criteria for leaking information.

Translation: The term for leakage of information from an event. Whichever details of an event become important to the application of other events become translated when the information is shared.

Escalation: The term for a significant change in an event which produces either mass translation or a level of indirection change. Note that while the word seems to imply uplifting, escalation actually refers to a reduction in level of indirection.

Locality

Note that by necessity, physical locality is not the only criteria by which events are subdivided.

6 Knowledge Representation

[To Do]

6.1 Experience

Nature vs. Nurture

Experience Trees

It may seem intuitive to encode the set of a characters experience as a linear stream of events sorted chronologically. We choose to instead represent a character's recalled

First, it is a more accurate representation of realistic though processes. People do not remember their lives as a list of chronologically sorted events; instead, memory functions like a tree of related recollections

Second, this representation allows us to utilize the exact same bigger-picture representation and lazy evaluation scheme in representing character experiences.

6.2 Facts

- Basic knowledge
- E.g. path between two locations
- Irrefutable
  - Doesn't mean that it is absolutely correct
  - Rather, it is binary. Either it was and still is correct or it has changed unbeknownst to the character

6.3 Belief

- Like knowledge, but adds confidence
- Effect of confidence governed by character attributes

Resolution

- Method depends on personality attributes
  - May be investigative
  - May be resigned

6.4 Personality

- Discretized into attributes with numeric values

6.5 Emotion

- Similarly discretized into attributes with numeric values

7 Interaction Representation

[To Do]

7.1 Language

- Support multiple languages
- Support dialects
- Simple representation
  - Avoid deep linguistics

8 Conclusions

In conclusion...

A Appendix

[To Do]

References

[Dyckhoff, 2008] Max Dyckhoff. Decision making and knowledge representation in halo 3. *Bungie Publications*, aug 2008.

[Isla, 2005] Damian Isla. Gdc 2005 proceeding: Handling complexity in the halo 2 ai. *GDC*, mar 2005.

[Isla, 2008] Damian Isla. Building a better battle: Halo 3 ai objectives. *Bungie Publications*, mar 2008.