

AI Pre-Mortem: How to Approach Your AI Problems by Michael Dawe, Rez Graham, Damian Isla, Dave Mark, Brian Schwab

Author: Sebastian Gabl

January 30, 2018

Tags: GDC16, AI, panel, discussion

Track: GDC 2016 - AI Summit

Url: <https://www.gdcvault.com/play/1023395/AI-Pre-Mortem-How-to>

Speaker: Michael Dawe, Rez Graham, Damian Isla, Dave Mark, Brian Schwab, GSN Games, Independent, The Molasses Flood, Intrinsic Algorithm, Magic Leap

Abstract

AI for games can be a complicated area - particularly since there is often more than one approach to solving a problem. Sometimes, the best way to learn about how to approach AI problems is to hear veteran developers work through them. This talk features four AI developers who take on a series of AI issues and talk through how they would think about and handle them. Some of the problems are ones the panelists have encountered, some of them they are provided by the community during the weeks leading up to GDC.

1 Summary of Talk

Dave Mark takes over the role of moderator and designer and together with the other speakers they take a look at common AI problems. The focus of this talk lies on making sure that developing AI does not go wrong in the first place, as opposed to a "Post-Mortem", where you would look at a completed project or task and talk about what went right and what went wrong afterwards. Thinking about how to approach the problem, which information is needed, what the potential pitfalls are and how to implement it in a good way to avoid performance or balancing problems are questions that one should ask themselves beforehand.

1.1 Making believable actors

The first example is about making AI believable in an indie project without a big art budget, meaning no detailed animations or voice acting. Michael Dawe brings up the Heider-Simmel demonstration about moving shapes [1] which is a simple video but still conveys emotion due to the way those shapes move around the screen. The other speakers agree that this isn't really an AI problem and list multiple examples where games managed to convey emotions and feelings without a big art budget, examples being Rimworld, Darkest Dungeon or Prison Architect.

1.2 How do multiple entities solve a new task?

In the second example, a user is asking about making individual entities with each their own tasks cooperate on a new unforeseen task without the use of some sort of squad manager. Rez Graham refers to the way it is done in Sims 3 and suggests a locking mechanism, where entities that care about a particular problem would register themselves for it and wait until the requirements are fulfilled to solve it. Michael Dawe and Damian Isla suggest a similar thing but making the entity use a blackboard system where it would tell others that it is willing to solve the problem and do something else until the requirements are fulfilled.

1.3 Optimal combined attack with minimal loss

This is about figuring out the optimal attack order in a turn based game which looks like a search problem at first glance but all of the speakers soon agree that it probably isn't necessary and might not even be fun for the player, especially if the player has to fight against an AI that optimizes all their moves. There would also be the potential problem of cascading searches, depending on the amount of moves and enemies available. One suggestion is to have a score for each move and work with that, another one is to first see if it even is a problem in-game.

1.4 Quest generator

The fourth example is about a quest generator in a contained environment, that generates quests based on the needs and constraints of that environment and is able to react to unexpected player behaviors. Damian Isla suggests a rule based system while Rez Graham comes up with a score based model where the AI would figure out the most pressing matter by calculating a score and then generating the appropriate quest. Brian Schwab suggests a system with abstracted resources with the possibility to do constraint search in.

1.5 AI director tasks

An AI director should detect emerging combat scenes and start directing it to make it more visually appealing by overriding individual entities behavior. This is similar to the problem presented in 1.2, but this time a squad manager is wanted. The speakers generally agree on solving it in a similar way but Rez Graham notes that one should be aware that the AI only is as smart as the player sees it so it is good to make sure to make the player aware of it somehow.

2 Overview and Relevance

2.1 Things to take home

We often like to over-think or over-complicate AI related tasks when presented with a problem. A lot of the examples presented in the talk actually had pretty simple solutions when looked at in detail. One might think a lot about how to implement certain tasks and put a lot of work and complex code in it when in the end, the player playing the game wouldn't even see the difference and a much simpler solution would have led to the same outcome. It should also be kept in mind that perfect AI is almost never desirable and that a balance between good AI and fun to play with/against should be maintained [2]. In general it is hard to make it look like an AI has made a genuine mistake instead of it appearing to just be broken but it is nevertheless important for an AI that "feels good".

2.2 Future of AI

AI has and will always play an important part in video games and applications and as technology progresses it can get harder to implement it in a convincing way, depending on the purpose. AI researchers have turned towards AI in video games, since they are run inside controllable environments, it is easy to get started and experiments can be run faster and a lot more often than for instance in robotics [3]. But especially as graphics get better and games focus more and more on realism, a common problem occurs more and more often: The Uncanny Valley. This happens when almost everything is visually perfect but something feels off [4], which is not always the result of graphics, but can also be caused by AI. In some cases neural networks can help eliminating the problem, for example for creating an "AI" that selects and blends appropriate movement animations for a character as discussed in the paper by Holden et al. [5]. The network is fed with multiple different motion captures and terrain data and learns how to create convincing character movement. Neural networks will most likely play a big role in future AI given our technology has gotten to a point where simulating such a network can be done in real-time (depending on its size).

3 References and Further Sources

- [1] Heider-Simmel, *Shape animation case study*, 1944.
<https://www.youtube.com/watch?v=VTNmLt7QX8E>
- [2] Jordan Garland, *Why aren't our enemies smarter? Gaming AI's problem of smoke and mirrors*, TechRadar, April 2014.
<http://www.techradar.com/news/gaming/shouldn-t-our-enemies-be-smarter-by-now-gaming-ai-s-trick>
- [3] Julian Togelius, *Why video games are essential for inventing artificial intelligence*, Togelius Blogspot, January 2016.
<http://togelius.blogspot.co.at/2016/01/why-video-games-are-essential-for.html>
- [4] Matthew Carrozo, *How Artificial Intelligence is changing the gaming industry*, Unbabel Blog, 2017.
<https://unbabel.com/blog/ai-changing-gaming-industry/>
- [5] Daniel Holden, Taku Komura, Jun Saito, *Phase-Functioned Neural Networks for Character Control*, ACM Transactions on Graphics (TOG) Volume 36 Issue 4, Article No. 42, New York, July 2017.