

Konzept für die zweite Bachelorarbeit

Vorläufiger Titel:	Performancgegenüberstellung von A-Stern und Ray-cast basierter AI-Agenten -Agent Navigation in sich dynamisch verändernden virtuellen Umgebungen-
StudierendeR:	Josua Kucher
Matrikelnummer:	Mt121050
BetreuerIn:	Wagner Markus
Vertiefungsrichtung:	Interaktive Medien

Kurzbeschreibung

Pfadfindalgorithmen wie beispielsweise A-Star sind in vielerlei Hinsicht eine sehr effiziente Lösung um AI gesteuerte Agenten in einer virtuellen Umgebung fortzubewegen. In der Regel besteht er aus zwei Teilen dem Buildalgorithmus der außerhalb von Runtime durchgeführt wird und dem Searchalgorithmus der die Suchen in Runtime übernimmt. Es gibt Varianten (DHPA* etc.) in denen der Buildalgorithmus dynamisch in Runtime ausgeführt wird um Änderungen in der Umgebung zu erkennen und für zukünftige Suchen zu berücksichtigen. Dies benötigt natürlich Rechenleistung in Runtime oder führt bei Nichtimplementierung zu Problemen mit dynamischen Umgebungen. (Kring, Champandard, & Samarin, 2010, p. 1)

In dieser Arbeit werden Teile dieser Rechenleistung direkt auf die AI-Agenten ausgelagert und ihnen mittels Raycasts zu ermöglichen ihre Umgebung wahrzunehmen. Hierzu wird eine pyramidenförmige Form vor dem Agenten in regelmäßigen Abständen mit Raycasts gefüllt. Dies erfolgt in einer Matrix von 5X4 Raycast welche vom Agenten ausgehend auf Punkte mit konstantem Abstand projiziert wird (Abbildung 1). Die Raycasts übergeben dann beim Auftreffen die Eigenschaften und Abmessungen der Objekte. Mit diesen Informationen sind die Agenten dann in de Lage eigenständig die Entscheidung zu treffen wie sie das potenzielle Hindernis umgehen/überwinden sollen.

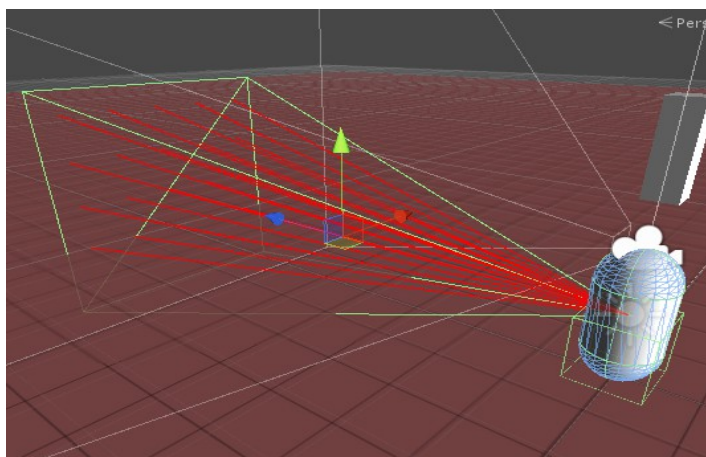


Abbildung 1: Raycasting in Pyramidenform

Forschungsfrage/Problembenennung

A-Star ist für dynamische drei dimensionale Umgebungen kaum geeignet und führt zu Performanceeinbusen in diesen Umgebungen, wenn man den Buildalgorithmus in Runtime wiederholt ausführen muss um dynamischer Änderungen den AI Agenten mitzuteilen.

In dieser Arbeit ist das Ziel herauszufinden wie die Lösung von dynamischen AI Agents im Vergleich zu A-Star im Punkt Performance abschneidet, wenn AI Agenten ein Hindernis von einem Startpunkt zu einem definierten Endpunkt durchqueren müssen.

Forschungsfrage:

Wie groß ist der Performanceunterschied zwischen dem A* Pfadfindalgorithmus und Raycastgestützten AI Agenten in statischen und dynamischen dreidimensionalen Umgebungen ?

Forschungsstrategie

Um diese Fragen zu beantworten, ist es notwendig einen Hindernisparkour zu konstruieren, der definierte Ziel und Endpunkte hat und einmal mit dynamischen AI Agenten und einmal mit A-Star gestützten Agenten angesiedelt wird die diesen durchqueren müssen.

Anschließend werden Performancetests mithilfe der Unity3D Performancetools durchgeführt und vergleiche zu den beiden Lösungen gezogen. Diese empirischen Daten werden anschließend statistisch ausgewertet um einen Vergleich beider Ansätze zu ermöglichen. Bsp.(Lazar, Feng, & Hochheiser, 2010)

Das Ziel der Arbeit ist dann erreicht, wenn die dynamischen Agenten in der Lage sind, die Karte ohne weitere Hilfe außer der Zielkoordinate zu durchqueren und alle Objekte die sie antreffen in eigenständiger Natur umgehen.

Um dies zu erreichen, muss zuerst ein Agent mit Raycasting und AI ausgestattet werden, welcher als Prefab für alle nachfolgenden Agenten dient.

Des Weiteren muss die Karte erstellt und mit dem A-Star Algorithmus eingelesen werden, um spätere vergleiche zu ziehen.

Als Letztes müssen Performancetests mithilfe der Unity3D Tools durchgeführt werden.

Um diese Schritte zu erreichen, muss Recherche im Bereich A-Star, AI Programmierung, Statistik und Raycasting betrieben werden. Hierzu werden Quellen aus der Fachliteratur zugezogen.

Literatur

Julian Togelius Georgios, N. Yannakakis (2013). A Panorama of Artificial and Computational Intelligence in Games. IEEE Transactions on Computational Intelligence and AI in Games

Kring, A., Champandard, A. J., & Samarin, N. (2010). DHPA* and SHPA*: Efficient Hierarchical Game Worlds. Conference on Artificial Intelligence and Interactive Digital Entertainment.

Malik Ghallab, Dennis G. Allard (1983). Ae — AN EFFICIENT NEAR ADMISSIBLE HEURISTIC SEARCH ALGORITHM. Laboratoire d'Automatique et d'Analyse des Systemes

Yee Chia Hui, Edmond C. Prakash, Narendra S. Chaudhari (2004). Game AI: artificial intelligence for 3D path finding. TENCON 2004. 2004 IEEE Region 10 Conference

Thomas P. Hartley and Qasim H. Mehdi (2011). In-Game Tactic Adaptation for Interactive Computer Games. 2011 16th International Conference on Computer Games (CGAMES)

Thomas P. Hartley and Qasim H. Mehdi (2011). In-Game Tactic Adaptation for Interactive Computer Games. 2011 16th International Conference on Computer Games (CGAMES)

Akenine-Möller, T., Haines, E., & Hoffman, N. (2008). Real-time Rendering (3. Auflage). Wellesley, Mass: Taylor & Francis Ltd.

Lazar, J., Feng, J. H., & Hochheiser, H. (2010). Research Methods in Human-Computer Interaction (1. Aufl.). Chichester, West Sussex, U.K: Wiley.

Weitere Anmerkungen

--