

Opdracht 3

1

Unity is speciaal ontwikkeld voor video games. De andere tools zijn meer gericht op educatie. Daardoor heeft Unity een grote hoeveelheid aan features. Dit kan handig zijn bij het maken van een simulatie. Vooral als je de simulatie er mooi uit wilt laten zien. Je kan met Unity je Agents zelf vorm geven waarmee je alles kan simuleren wat je maar kan bedenken. Wel zorgt de grote hoeveelheid aan features ervoor dat het lastig is om het overzicht te houden en maakt het dat een vrij simpele agent al moeilijk is om te implementeren. Je bent het grootste gedeelte van de tijd bezig om alles te snappen en werkend te krijgen. Het agent gedeelte raakt dan verloren in al de andere dingen die moeten gebeuren om een goede demonstratie te kunnen geven van een agent based simulatie.

Mijn code is agent based maar door de complexiteit van unity en de tutorial die niet altijd werkt vond ik het lastig om alles werkend te krijgen. De manier van denken is niet heel lastig maar het implementeren in unity bleek voor mij haast onmogelijk. Het was de bedoeling dat mijn AI partikels liet zien zodra mijn Playertank was gespot maar het lukte me niet om dat te implementeren.

2

1. De agent begint met een start state "i0" die in de set I zit. Aan de hand van acties die getriggerd worden door de perceptie wordt de interne staat van de agent veranderd.
2. De functie S geeft de agent mogelijkheden om zijn omgeving waar te nemen. In het geval van dit unity project gebruiken we de S functie touch. Met deze functie kan de agent er achter komen of hij wordt "aangeraakt". Als de S functie dan iets voelt dan kan de agent dat opmerken met de functie P.
3. Op dit punt als de agent wat heeft waargenomen gaat de agent over op een actie die zijn interne staat verandert. In het geval van het voorbeeld geeft de agent partikels af en vlucht weg van de Playertank. (als het me was gelukt om het werkend te krijgen)
4. Nadat de agent de omgeving heeft waargenomen en een actie heeft gedaan wordt de interne staat geupdate en gaat hij weer verder met zijn waarneming staat i0 en kan de hele loop weer opnieuw beginnen.

In het kort gezegd de agent neemt waar, hij doet wat en neemt weer waar.

3

1. Mijn omgeving is Accessibele wand in unity zouden mijn agents alles kunnen weten wat er in de omgeving afspeelt. We weten alle informatie van de omgeving.
2. De omgeving is ook deterministisch want het is zeker wat er gaat gebeuren bij het uitvoeren van acties.
3. Het is episodisch want er wordt alleen rekening gehouden met de huidige staat.
4. De omgeving is wel dynamisch want er kan interactie plaatsvinden die de situatie verandert. In de omgeving kan de target van de playertank door de gebruiker worden veranderd.
5. De omgeving is continuus wat er is geen endstate de omgeving gaat als de AI de playertank niet meer detecteert gewoon verder totdat de playertank weer wordt gedetecteerd. Dat kan oneindig verdergaan.

1. Accessible vs inaccessible. De omgeving zou inaccessible gemaakt kunnen worden door in plaats van de agent interactie te laten hebben met objecten gemaakt door unity zouden we een camera boven een krijspunt kunnen hangen en de agent laten proberen om de weggebruikers te kunnen laten detecteren. Dit zou ervoor zorgen dat we niet alle informatie meer hebben de omgeving weet immers niet of er wegebruikers aankomen of niet. Want die worden nu niet meer door de omgeving bestuurd.
2. Deterministic vs non-Deterministic. Om de agent non-Deterministic te maken kunnen we hem meerdere acties geven waaruit gekozen kan worden om te doen zodra de agent de playertank waarneemt. Inplaats van partikels te toen zou de agent kunnen vluchten van de playertank of misschien er rondjes rond heen vliegen. Omdat we nu niet meer met zekerheid kunnen zeggen wat voor actie er gaat gebeuren is het nu non-Deterministic geworden.
3. Discrete vs continuous. Als we een eindactie aan de agent toevoegen, bijvoorbeeld als de playertank word gedetecteerd laat je partikels zien en daarna delete je jezelf kunnen we er een discrete systeem van maken. Er zit nu een duidelijk einde aan de loop van de agent.