

Opdracht 1

Rollercoaster Tycoon 2

Rollercoaster Tycoon 2 is een spel dat draait om het runnen van een pretpark. Om het spel te kunnen winnen moet de speler zich constant bezig houden met het welzijn van zijn of haar klanten (in het spel heten klanten "Guests"). Omdat de Guests zo'n prominente rol spelen in het onderhouden van een welvarend pretpark, leek RCT2 (Rollercoaster Tycoon 2) mij het ideale onderwerp voor dit onderzoek.

Variabelen

Naast Guests, zijn er ook andere AI in het spel te vinden, zoals klusjesmannen en schoonmakers. Maar aangezien de Guests het meest uitgebreid en interessant zijn zal ik het alleen over de Guests hebben.

De Guests in RCT2 lijken op het eerste oog vrij simpel. Ze lopen een beetje rond, gaan een attractie in en verlaten vervolgens het park. Maar naarmate je spel meer speelt, kom je er al snel genoeg achter dat ze toch wat ingewikkelder in elkaar zitten.

Een Guest heeft een aantal verschillende variabelen die bij worden gehouden. De meeste hiervan worden gecommuniceerd naar de speler, maar er zijn ook een aantal verstopte variabelen waar ik het later over zal hebben.

Op deze afbeelding zie je de variabelen die naar de speler worden gecommuniceerd. Zo zie je voor: **Blijheid**, **Energie**, **Honger**, **Dorst**, **Misselijkheid** en hoe nodig ze naar het **Toilet** moeten een balk die met behulp van kleuren aangeeft of ze positief of negatief zijn.

Onder de balken zie je wat extra informatie waar spelers vaak overheen kijken. De bovenste waarde geeft aan hoe **intens** een achtbaan moet zijn voor de Guest om er in te gaan. De waarde daaronder geeft aan hoe snel een Guest **misselijk** word.



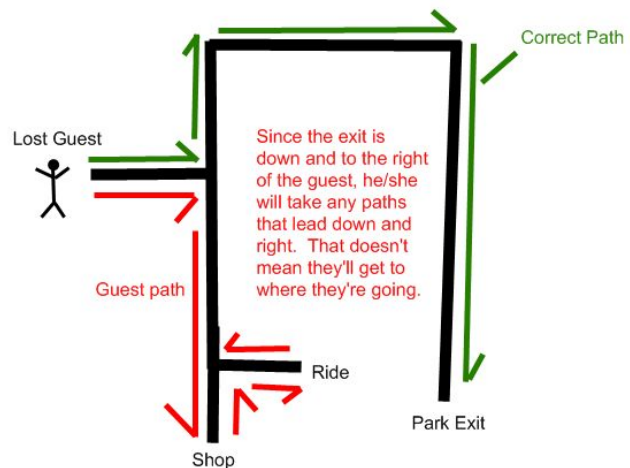
Deze variabelen zijn de fundatie voor het gedrag van een Guest. Een hoop van de keuzes van een Guest worden gemaakt aan de hand van deze variabelen.

States

Een Guest kan zich in verschillende states bevinden. Ik zal ze proberen allemaal kort toe te lichten.

Lopen

Een guest heeft 2 verschillende manieren van lopen. Willekeurig **rondstruinen** of op een **doelwit** aflopen. Als een guest naar een specifiek doelwit probeert te komen checkt bij elk knooppunt welke stap hem dichterbij het doelwit brengt. Dit is een extreem primitieve manier van pathfinding en brengt ook een aantal problemen met zich mee. Onder de juiste condities kan een Guest namelijk verdwaald raken (de onderstaande tekening legt uit hoe dit werkt). Hoewel dit in het geval van RCT2 een probleem is, kan het ook als een interessante game mechanic gebruikt worden. Zo heb ik een tijdje terug hetzelfde pathfinding systeem gebruikt voor de spookjes in een Pacman game. Wanneer de speler hier bewust van is kan de speler, door slim te spelen, de spoken afschudden.



Als een Guest vrij rondloopt en bij een knooppunt komt kiest de Guest willekeurig een kant uit om op te gaan (hoewel er door een hoop spelers is vastgesteld dat ze een voorkeur hebben om vooruit te lopen).

A guest that is "walking" i.e not going anywhere in particular, at a junction (which could be a path junction or it could be the entrance to a ride, shop etc), will make a decision at random about which path to take. They are biased towards going straight on- I've found that ride entrances that connect to a path at right angles get far fewer guests.

Evaluatie van attractie of winkel

Als een Guest voor een winkel of de ingang van een attractie staat vergelijkt hij een aantal dingen. Hij vergelijkt de prijs van de attractie of winkel met het geld dat hij heeft en wat hij bereid is om uit te geven. Onderliggend hebben ze vast een formule om hun maximale budget per attractie of winkel uit te rekenen, maar hier ziet de speler niks van.

Voor een attractie is er een tweede waarde waarop gecheckt wordt, namelijk de intensiteit. Als een attractie te intens is voor de Guest gaat hij er niet in. Als hij er wel in gaat ook al is het een intense attractie, kan de Guest na afloop misselijk worden en zelfs kotsen. Als er een prullenbak in de buurt is zal de Guest hier in proberen te kotsen, zo niet, dan heb je een voetpad vol met kots.

In de rij staan

In het spel kun je niet alleen voetpaden maken, maar ook wachtrijen. Guests waar de attractie niet genoeg plek voor heeft gaan hier achter elkaar staan om te wachten tot ze aan de beurt zijn. Als een rij te lang is kan dat een reden zijn voor een Guest om door te lopen. Zodra een Guest eenmaal in de rij staat, kan hij er niet meer uit. Dit zorgt er voor dat de speler bewust keuzes moet maken in de lengte van een wachtrij, want terwijl Guests in de rij staan kunnen ze honger krijgen of naar de wc moeten en zullen daardoor minder tevreden zijn over je park.

In een attractie

Er is enorm veel variatie in attracties. De meest voorkomende zijn die waar de speler een eigen baan maakt (vaak een achtbaan, maar het kan ook iets zijn, zoals een kartbaan). Bij deze vorm van attractie hoort altijd een voertuig (karretje, bootje, shuttle, etc.). Dit voertuig animeert over de baan en komt tot slot bij de uitgang uit. Wanneer een Guest een attractie betreedt, wordt hij als het ware “geparent” aan het voertuig en animeert hij met het voertuig mee. Wanneer het voertuig bij de uitgang komt wordt de Guest weer losgekoppeld van het voertuig en begint hij weer met lopen.

Er zijn ook attracties zoals glijbanen waar helemaal geen voertuig wordt gebruikt, maar waar de Guests gewoon een simpele animatie uitvoeren. Andere uitzonderingen zijn waterfietsen en doolhoven. Bij waterfietsen is er een optie om geen baan te gebruiken. Wanneer de speler dit kiest, zullen de Guests willekeurig in het rond varen voor een bepaalde tijd en uiteindelijk terugkeren naar de uitgang. Bij doolhoven zullen de Guests, via dezelfde methode als het willekeurig rondlopen over paden, op zoek gaan naar de uitgang.

Grimmig genoeg is er ook een optie voor de Guests om te sterven in het spel. Dit komt zelden voor (tenzij je het expres doet), maar gebeurt vaak wanneer een achtbaan te hard gaat en uit de bocht vliegt. Een andere optie is wanneer de speler een brug verwijdert waar Guests op lopen. Dit zorgt er voor dat ze in het water terecht komen en langzaam verdrinken. Yikes.

Wat doen ze goed, en wat niet?

In dit document heb ik hier en daar al wat dingen genoemd die de AI in RTC2 goed doet, of juist niet. Maar wat de AI extreem goed doet, is het simuleren van complex gedrag. Door de simpele regels tijdens het lopen en keuzes maken op basis van een aantal waardes komen de Guests echt tot leven. Dit spel is al redelijk oud en is er met gelimiteerd computergeheugen in geslaagd om honderden Guests tegelijkertijd rond te laten lopen in het spel. Allemaal met hun eigen gedrag en zonder je computer te laten sneuvelen.

Hoewel dat voor die tijd enorm knap was, zijn we tegenwoordig wat meer gewend. Daardoor kan het spel soms te limiterend voelen voor de speler en zelfs tot frustratie leiden. Ik kan me nog goed herinneren dat ik vroeger altijd enorm gefrustreerd raakte wanneer mijn park slecht werd beoordeeld omdat niemand te uitgang wist te vinden. Andersom creëert dit ook een hoop kansen om het spel te exploiteren, door bijvoorbeeld spelers eindeloos je attractie in te laten gaan. Kortom, de pathfinding is iets waar zeker veel verbeterd aan kan worden.

None of this is computationally expensive- four array accesses, a function call (to get a random number), and then a conditional, and it only needs to be evaluated when a guest is at a junction. It is an example of how a few simple rules can result in seemingly complex behavior.

This simplicity can also be exploited to get large numbers of guests on your rides/buying from your shops.

Maar persoonlijk vind ik dat juist wel passen bij het spel. Het spel was voor mij als kind entertainend genoeg, zonder dat ik veel last had van de primitieve pathfinding. Maar juist door deze limitatie ontstaat er een hele andere manier van spelen, namelijk het maken van het meest efficiënte pad. Of het doelbewust maken van een park waar de Guests nooit weg kunnen.

De limitaties in de AI zorgen ervoor dat spelers probleemoplossend gaan denken. En hierdoor ontstaan communities die samenwerken om alles uit het spel te halen.

Sidenote: Wat echt super frustrerend is aan het spel, is dat wanneer je een schoonmaker inhuurt, hij automatisch gras gaat maaien. Grasmaaien is echt het meest nutteloze wat een schoonmaker kan doen en als je veel gras hebt zijn ze er uren mee bezig, terwijl de rest van je park vol ligt met kots. Je kunt dit uitzetten, maar moet dat voor elke schoonmaker individueel doen. Iets wat ze in nieuwere versies van het spel gelukkig gefixt hebben.