

Na początku dokumentu otrzymujemy informację o oprogramowaniu potrzebnym do zainstalowania i prawidłowego działania gry Angry Birds w wersji offline. Potrzebny będzie nam do tego Chrome oraz Java. Dokument posiada już linki, z których można ściągnąć dane oprogramowanie. Następnie szczegółowo opisana jest procedura dodania wtyczki do Chrome dzięki której gra powinna działać oraz wyjaśnienia odnośnie AI Agent. AI Agent analizuje scenerię gry aby podjąć decyzje odnośnie strzału. Mimo że jest on napisany w Javie można go zaimplementować w innym języku takim jak np. python.

Dalej poruszane są takie tematy jak integracja serwera z rozszerzeniem angry birds przez moduł proxy oraz wyszczególnione wiadomości proxy wraz z wyjaśnieniami.

Komunikacja

Port Serwer/Klient komunikują za pomocą agenta i odsyła zwrotne informacje po wykonaniu przez serwer wymaganych działań wskazanych przez komunikaty.

Wiadomości te można podzielić na następujące kategorie:

- konfiguracja
- kwerendy
- akcja podczas gry
- selekcja poziomów

Moduł wizyjny

Moduł wizyjny składa się z dwóch komponentów segmentacji obrazu. Jeden komponent dzieli obraz i wyprowadza listę minimalnych prostokątów ograniczających (MBR) podstawowych obiektów na obrazie. Do podstawowych obiektów należą:

- procy
- czerwony ptak
- żółty ptak
- niebieski ptak
- czarny ptak
- biały ptak
- świnia
- lód
- drewno
- kamień
- TNT
- TrajPoint

Drugi składnik zamiast tego wyprowadza prawdziwe kształty. Oba mogą zostać użyte do przetworzenia zrzutu ekranu.

Obliczenie rekordów minimalnych prostokątów ograniczających typowego scenariusza zajmuje ok. 100 ms. Aby obliczyć prawdziwy kształt potrzeba ok. 300-500 ms. Oprócz tego segmentacja rozpoznaje także wzgórze i ziemię.

Moduł trajektorii szacuje trajektorię, którą ptak będzie podążał, biorąc pod uwagę określony punkt uwolnienia (w stosunku do procy). Używamy stałej prędkości dla każdego strzału na tym samym poziomie i używamy punktu uwolnienia i lokalizacji procy, aby określić kąt startu α . Z nich możemy obliczyć początkowe prędkości poziome i pionowe. Aby ustalić paraboliczny lot ptaka używa się klasycznych praw Newtona. Funkcja predictTrajectory w module trajektorii zwraca listę punktów, które ptak będzie podążał za określonym punktem zwolnienia.

Po zrobieniu zdjęcia rzeczywisty kąt startu α jest zawsze nieco inny niż \arcsin tangens wektora startowego, a różnica zmienia się z poziomu na poziom. Zmienność kąta i prędkości są brane pod

uwagę podczas przewidywania trajektorii i szacowania punktów startowych. Moduł trajektorii zapewnia metodę (`adjustTrajectory`) pomagającą dostosować te dwie zmiany za pomocą informacji z poprzedniego strzału, aby poprawić dokładność następujących ujęć.

Aby pokazać jak powinien wyglądać agent (włączając w to `chrome`, moduł wizji i moduł trajektorii), który będzie odpowiedzialny za oddawanie strzałów został skonstruowany naiwny agent. Jest on nazwany naiwny ponieważ po prostu strzela bezmyślnie w świnie.

Następnie w dokumencie opisane jest jak włączyć stacjonarnie naiwnego agenta, po czym przechodzimy do instrukcji jak stworzyć swojego własnego inteligentnego agenta.

Dla każdego obiektu na rzucie ekranu moduł wizji tworzy nad nim obiekt `java`. Obiekt `java` jest określony w `ABObject.java`, który rozszerza `java.awt.Rectangle`. Klasa `ABObject` ma cztery główne atrybuty, a mianowicie współrzędna `x`, `y` lewego górnego rogu, szerokość i wysokość.

Klasa `ABObject` ma atrybut wskazujący typ obiektu gry. Typy obiektów to: {"Hill", "Sling", "Red Bird", "Yellow Bird", "Blue Bird", "Black Bird", "White Bird", "Pig", "Ice", "Wood", "Kamień", "TNT"}. Atrybut "pusty" wskazuje, czy obiekt jest pusty (ma otwór pośrodku), czy nie.

Segmentacja MBR nie rozróżnia obiektów pustych i pełnych. Kształty klasyfikujemy w trzech grupach `Circle`, `Rect` i `Poly`.

Następnie mamy przedstawiony gotowy kod do robienia rzutu ekranu i segmentacji. Zapewniona została makro wizja, która zarządza dwoma komponentami segmentacji obrazu. Metoda `findPigsRealShape ()` używa segmentacji w kształcie rzeczywistym, aby uzyskać listę świnek, podczas gdy metoda `findPigsMBR ()` używa segmentacji MBR. Żadna z metod nie jest w stanie rozpoznać wszystkich kategorii obiektów.

Jest pięć głównych statusów gry:

- wygrałeś
- przegrałeś
- gra w toku
- wybór poziomu
- ładowanie gry

Kiedy znamy nasz cel, potrzebujemy modułu trajektorii, aby zaplanować strzały. Aby uzyskać zestaw punktów podanych jako cel, możemy wywołać metodę `estimateLaunchPoint (Rectangle, Point)`. Ta metoda wymaga MBR procy i punktu docelowego.

Oddanie strzału

Strzał zawiera sześć parametrów (`x`, `y`, `dx`, `dy`, `t_shot`, `t_tap`) Pierwsze cztery parametry określają ścieżkę myszy od (`x`, `y`) do (`x + dx`, `y + dy`). `T_shot` to czas, w którym ptak ma zostać uruchomiony. `T_tap` określa przerwę między czasem zwalniania a odpowiadającym mu czasem gwintowania. Czas liczony jest w milisekundach.

W dokumencie opisane są też inne przydatne metody takie jak osiągalność, która sprawdza czy cel jest w twoim zasięgu

Na końcu znajdziemy informacje że do kompilacji można używać `ANT`, która jest biblioteka `java` i narzędziem wiersza poleceń.