

Marcin Apostoluk
Wydział Informatyki I Zarządzania
Politechnika Wrocławska

Wrocław, 02.04.2006

Sztuczna inteligencja sterowana potrzebami

Abstract

Większość z podejść do tworzenia inteligentnych przeciwników w grach komputerowych, choć wykorzystuje wiele z technik sztucznej inteligencji, wymaga oprogramowania zachowania przeciwnika w różnorodnych sytuacjach. Przykładowo boty w grach typu FPP (First Person Perspective) programowane są by inteligentnie omijały wystrzelone w ich kierunku rakiety, czy też przewidywały pozycję gracza na kilka sekund do przodu, by mądrze dobrać broń i z niej skorzystać. Pewien czas temu pojawiła się jednak koncepcja sztucznej inteligencji sterowanej potrzebami, która próbuje ominąć dostosowanie zachowań przeciwników do różnych sytuacji wprowadzając ogólny szablon zachowania. To właśnie podejście zostanie pokrótce opisane w niniejszym artykule.

Wprowadzenie

Artykuł ten opiera się na pracy Björn Bergström'a ([1]), który przedstawił podejście zaproponowane przez Hansjoerg Malthaner'a dotyczące sztucznej inteligencji postaci w grach komputerowych. Podejście to oparte jest na behawioryzmie, który bada reakcje organizmów na zadane bodźce. Może się to wydawać niewystarczające, gdyż nie każda reakcja musi być zależna od zadanych bodźców, lecz jak twierdzi Björn Bergström, metoda ta powinna być w pełni wystarczająca dla potrzeb sztucznej inteligencji przeciwnika w grze komputerowej.

1. Główne założenia

Zachowanie postaci w grze komputerowej powinno być oparte na 3 głównych krokach:

1. Obserwuj i oceń środowisko, w którym się znajdujesz.
2. Wybierz najbardziej niezbędną potrzebę, bazując na ocenie środowiska oraz własnych potrzeb.
3. Wykonaj niezbędne działania, aby zrealizować potrzebę.

Kroki te zostaną opisane w dalszej części pracy.

2. Obserwacja i ocena środowiska

Postać w grze komputerowej, aby ocenić środowisko, w którym się znajduje, powinna wziąć pod uwagę wszystkie obiekty oraz postacie, które są w zasięgu jej wzroku. Dla każdego z obiektów i postaci powinna przypisać współczynnik strachu, nienawiści lub przyjaźni.

Do obliczeń tych wartości przydatne są następujące parametry:

- **postrzegana siła postaci** – na jak silną dana postać wygląda. Wartość ta powinna być obliczana na podstawie tężyzny i siły fizycznej postaci, nie zaś przedmiotów, które posiada

- **postrzegalne niebezpieczeństwo postaci** – na jak niebezpieczną dana postać wygląda. Wartość ta powinna być obliczana na podstawie jakości przedmiotów, które posiada (broni, zbroi itp.)

Wartość strachu lub przyjaźni bazuje na powyżej wymienionych parametrach oraz na odległości od postaci. Dla przyjaciół rozpatrujemy wartość przyjaźni, dla wrogów zaś – wartość strachu. Wartości te mogą być traktowane wspólnie jako jeden współczynnik, gdy np. dla wartości przyjaźni rozpatrywać będziemy wartości dodatnie, dla wrogów – wartości ujemne. Współczynnik ten może odnosić się także do przedmiotów (przykładowo skarb – wartość dodatnia).

Wartość nienawiści odzwierciedla wielkość tego uczucia do danej postaci. Jest to współczynnik różny od współczynnika strachu, wykorzystywany przykładowo do określenia, których postaci, dany przeciwnik bardziej nienawidzi (gdyż są przykładowo innej rasy).

Kolejnym krokiem jest obliczenie środka ciężkości powyższych współczynników.

.....
..1...x...2..
.....
.....
.....
....A.....

Przykład obliczania środka ciężkości współczynników

Źródło: [1]

Powyżej załączono przykład obliczeń środka ciężkości dla współczynnika nienawiści. Poprzez 'A' oznaczono postać obliczającą współczynnik, poprzez '1' oraz '2' oznaczono przeciwników o identycznych współczynnikach nienawiści, poprzez 'x' oznaczono wynikowy środek ciężkości dla współczynnika nienawiści. Jak widać obliczenia nie są złożone, a co za tym idzie czasochłonne.

3. Wybór najważniejszej potrzeby

Postać korzystająca z inteligencji sterowanej potrzebami powinna mieć przypisane pewne akcje, które może wykonywać, przykładowo: poruszanie się, atak, czy otwarcie drzwi. Przed wyborem kolejnej akcji do wykonania, dla każdej z akcji obliczana jest waga mówiąca o tym, jak konieczna jest dana akcja w danym momencie. Wykonywana jest ta akcja, która posiada największą wagę.

Dla każdej z akcji metoda obliczająca jej wagę musi być oddzielnie zaimplementowana. Przykładowo waga akcji otwierania drzwi w danym momencie może być zależna od różnych czynników, Zależności te muszą być zaimplementowane najlepiej w taki sposób, aby wyliczanie wagi dla akcji otwierania drzwi nie wpływało w znacznym stopniu na sposób wyliczania wagi dla innych akcji, np. ataku.

Samo wykonanie akcji odpowiedzialnej za spełnienie danej potrzeby nie powinno stanowić problemu i zapewne ograniczy się do wywołania jednej metody.

4. Dalsze urozmaicenia

Istnieje wiele sposobów na wprowadzanie dodatkowych elementów do rozgrywki. Różnym postaciom w grze można przypisywać różne formy obliczeń współczynników strachu, przyjaźni i nienawiści. Manipulując odpowiednio współczynnikiem przyjaźni można sprawić, że postacie będą poruszały się w grupach i bały od nich odłączyć. Także różnorodne cechy fizyczne będą mieć wpływ na całą rozgrywkę, przykładowo odległość oraz kąt widzenia może być różna dla różnych potworów w grze, co również wpłynie na ich inteligencję i postrzeganie otoczenia.

Jeśli dla potwora strach opada szybciej wraz z odległością niż nienawiść, potwór taki będzie podążał za bohaterem, lecz bał się zbliżyć do niego, być może poczeka na współtowarzyszy. Bez problemu można rozwiązać problem postaci atakujących z dystansu, dla których ustalając współczynnik strachu bardzo wysoki dla krótkich odległości i niski dla większych, uzyska się efekt, że postacie te będą uciekać od bohatera i atakować z dystansu, gdy opadnie już strach.

5. Metoda idealna?

Czy więc sztuczna inteligencja sterowana potrzebami jest rozwiązaniem znacznie lepszym od pozostałych? Choć sama koncepcja wydaje się bardzo ciekawa i daje ogromne możliwości, istnieją głosy krytyki mówiące, że rozwiązanie to jest niewystarczające. Postać atakująca dwa potwory, których środek ciężkości nienawiści znajduje się pomiędzy nimi, będzie musiała zdecydować się na pewną taktykę po dojściu do tego punktu, a funkcjonalność ta będzie musiała być manualnie zaimplementowana. Z drugiej strony, wyobraźmy sobie bota w grze FPP, który omija rakietę, gdyż jest to obiekt, przed którym czuje ogromny strach, który np. chowa się za ścianą nie mając już amunicji, gdyż uważa to za najbezpieczniejsze posunięcie.

Rozwiązanie to jest na pewno bardzo ciekawe i warte zainteresowania twórców gier komputerowych. Choć może okazać się niewystarczające do implementacji całej sztucznej inteligencji przeciwnika komputerowego, będzie z pewnością bardzo przydatne przy jego tworzeniu i może rzeczywiście uprościć wiele aspektów i zrealizować zarazem sporo pomysłów, które wymagałyby ręcznej implementacji. Bot odczuwający strach lub nienawiść będzie bardziej przypominał człowieka, a co za tym idzie dobrze sprawdzi się jako przeciwnik gracza komputerowego, należy bowiem pamiętać, że przeciwnik idealny to przeciwnik zachowujący się jak człowiek, nie zaś bot, który za każdym razem trafia w gracza i z którym gra to żadna przyjemność.

Bibliografia

- [1] <http://home.swipnet.se/dungeondweller/development/dev00055.htm>
- [2] Joseph Swing „A better monster AI”