
**Projektna dokumentacija iz predmeta
Inteligentni multiagentski sustavi
"Pacman"**

*Stjepan Antivo Ivica (0036448356)
Vanessa Županović (0036449759)*

Zagreb, Lipanj 2013.

Sadržaj

Uvod:.....	3
Model.....	3
Svijet:	3
Agent Pacman.....	4
Politika odlučivanja agenta.....	4
Funkcija korisnosti.....	4
Agent duh.....	5
Politika odlučivanja.....	5
Testiranje uspješnosti implementacije.....	5
Slučaj 1:.....	6
Slučaj 2:.....	6
Slučaj 3:.....	7
Slučaj 4:.....	7
Slučaj 5:.....	8
Zaključak.....	8
Literatura.....	8

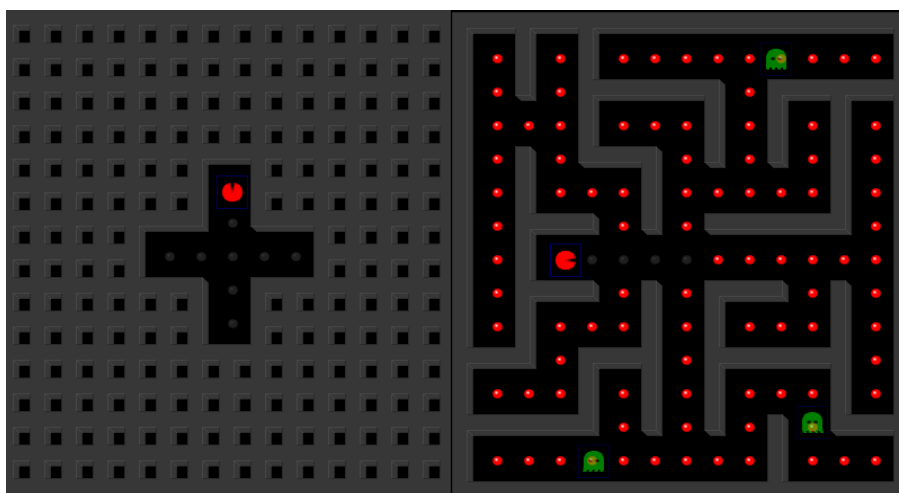
Uvod:

Cilj ovog projekta bio je nadograditi postojeću implementaciju igre "Pacman" na način da se izrade odgovarajući modeli autonomnih racionalnih agenata Pacmana i duhova s ciljem omogućavanja igre bez intervencije korisnika (što je uspješno ostvareno) te mogućeg postizanja stanja pobjede od strane agenta Pacman u ovisnosti o uspješnosti implementacije samog agenta. Za programsku realizaciju igre korištena je grafička biblioteka za razvoj igara ClanLib te je ista ostvarena u programskom jeziku C++. Raspon okoline, veličina mape i njena povezivost, broj duhova i njihova pozicija te početna pozicija agenta Pacman jesu promjenjive, no budući da se parametri zadaju kao funkcije moguće ih je mijenjati isključivo u kodu unutar datoteke "application.cpp" te tako izraditi specifične i po potrebi vezane testove.

Model

Svijet:

Svi agenti nalaze se u labirintu koji ujedno predstavlja njihovu okolinu. Okolina je iz perspektive svakog od agenata dinamična, tj. mijenjaju je drugi agenti. Važne karakteristike samog svijeta igre jesu veličina te povezivost. Važno je naglasiti da se za generiranje svijeta igre koristi gotov generator (dio inicijalne implementacije), stoga stohastičko generiranje svijeta katkad rezultira vrlo lošom mapom. Mapa sadrži objekte: kolač, veliki kolač, zid, praznina i trag, te agente duhove (broj duhova moguće je postaviti po volji te inicijalno iznosi 1) i Pacmana koji za razliku od agenata duhova posjeduje znanje (memoriju). Količina i raspodjela velikih kolača, koji Pacmanu mijenjaju svojstva, su fiksne, tj. 4 velika kolača su smještene po jedan u svaki od kutova mape, nadalje, niti jedan od agenata nije u mogućnosti prolaziti kroz zidove. Objekte kolač i veliki kolač može "pojesti" isključivo Pacman i to na način da zauzme polje na kojem se nalaze. Ukoliko agent Pacman "pojede" veliki kolač moći će kroz neko izvjesno vrijeme "pojesti" i agenta duha čime u spomenutom vremenu mijenja strategiju agenta duha (detaljnije kasnije). Ukoliko "pojede" kolač Pacman za sobom ostavlja trag kojeg agenti duhovi mogu pratiti (agenti duhovi "jedu" trag) što im omogućava promjenu stanja nasumičnog gibanja s ciljem lociranja Pacmana, sam Pacman neometano prolazi kroz vlastiti trag. Krajnji cilj svakog agenta duha je uloviti Pacmana, dok Pacman nastoji ostvariti pobjedu na način da pojede sve kolače koji se nalaze na mapi. Valja primijetiti da agenti duhovi i Pacman imaju suprotne strategije, te da su međusobno heterogeni. Između istih nema nikakve komunikacije niti pregovaranja.



Slika 2.1 Primjer izgleda svijeta. Lijevo – loše generirani svijet, desno – dobro generirani svijet s 4 autonomna agenta (3 duha i Pacman).

Agent Pacman

Osnovi cilj agenta Pacman je pojesti sve raspoložive kolače na mapi, a da pri tom ne biva pojeden od strane agenta duha. Ukoliko mu to uspije, smatra se da je ostvario pobjedu. Agent ima memoriju u obliku skupa (set) koordinata u kojoj se inicijalno nalaze 4 kuta mape (položaji velikih kolača), stoga agent na početku igre zna da postoje 4 kuta mape koje još nije posjetio a gdje se nalaze veliki kolači. Početno je broj kolača koje agent mora pojesti jednak broju lokacija labirinta koje Pacman treba posjetiti. Koordinate svakog pojedenog kolača dodaju se u iznad spomenutu strukturu. Gibanje agenta se može opisati kroz sljedeća stanja:

- LEFT : agent se giba u lijevo
- RIGHT : agent se giba u desno
- UP : agent se giba prema gore
- DOWN : agent se giba prema dolje
- RANDOM : agent se giba nasumično
- STOP: agent se ne giba, čeka

Agentu je u svakom trenutku raspoloživ (vidljiv) proizvoljno zadan pravokutan dio diskretnog prostora. Primjerice, ukoliko je agent u točki (0,0) i "vidi" do točke (3,3) onda njegov vidokrug zapravo čini kvadrat dimenzija 3x3.

Sam agent Pacman modeliran je sljedećim stanjima:

- SEARCH_FLOURISH
- SEARCH_POOR
- SEARCH_DESPERATE
- FLEE
- STALL
- KILL_EM_ALL

Politika odlučivanja agenta

Politika odlučivanja agenta definirana je na sljedeći način: Prvo se provjerava ukoliko je agent pod utjecajem velikog kolača, tj. ukoliko agent pojede veliki kolač ulazi u stanje KILL_EM_ALL s ciljem da "pojede" sve duhove. Nadalje, nakon navedenog, utvrđuje se je li agent u opasnosti. Ukoliko se uspostavi da to nije slučaj agent u miru nastavlja pretraživati prostor stanja u potrazi za hranom počevši sa SEARCH_FLOURISH, tj. gleda postoji li do njega takvo polje na kojem se nalazi kolač. Agent u tom stanju pohlepno odabire bilo koji smjer gibanja uz prethodno opisani uvjet. Unutar stanja SEARCH_POOR agent odabire smjer koji pripada najkraćem putu prema kolačiću koji mu je unutar vidljivosti, agent u ovo stanje ulazi ukoliko postoji nepojedeni kolač koji jest u vidljivosti no nije na niti jednom od susjednih polja. Ukoliko ne postoji kolač u vidokrugu Pacman ulazi u stanje SEARCH_DESPERATE, onda je odabrani smjer gibanja jednak najkraćem putu prema jednom od kolača kojeg je zapamtio. Ako takav kolač ne postoji, giba se nasumično. U stanju FLEE agent je u opasnosti te vidi duha (unutar vlastitog vidokruga) i bježi u smjeru u kojem može postići najviše koraka a da ga pri tom duh ne ulovi, dok u stanju STALL vidi veliki kolač te gleda postoji li duh u njegovoj vidljivosti. Ako postoji i nije preblizu (unutar polovine vidljivosti), stani i čekaj (zavlači igru). U stanje KILL_THEM_ALL Pacman ulazi ukoliko pojede veliki kolač što mu daje "moći" da razazna pozicije duhova te se "zaleti" prema njima s namjerom da ih "ubije", no pri tom ne odabire sebi najbližeg već onog prvog stvorenog. U SEARCH_POOR i SEARCH_DESPERATE i KILL_EM_ALL agent pretražuje prostor BFS-om, dok se u stanju FLEE rabi min-max algoritam.

Funkcija korisnosti

Još jednu od vrlo važnih stavki modela ovog agenta predstavlja funkcija korisnosti, koja

je definirana na način: ukoliko agent pojede kolač, to mu donosi 125 poena, nadalje za pojeden veliki kolač agent dobiva 200 poena, ukoliko pak pojede duha, to mu donosi 1000. Prolaz vremena setira agentu broj bodova na 0, te naposljetku, ako agenta pojede duh, isti dobiva -5000 (ukupni broj bodova ne može poprimiti negativne vrijednosti, tj. ukoliko je suma bodova < 0 , setira se na 0.)

Čitav proces odlučivanja s iznad opisanim stanjima može se vidjeti unutar "gameobject_pacman.cpp" koji zbog svoje veličine nije priložen u dokumentaciji.

Agent duh

Trajni cilj svakog agenta duha je uloviti Pacmana. Duhovi su u osnovi "jednostavni" homogeni agenti, te se po mapi kreću nasumično sve dok ne nalete na trag agenta Pacman kojeg "jedu" te pri tom mijenjaju boju, dakle, modelirani su isključivo s dva stanja: random i prati trag. Valja napomenuti da duhovi u osnovi nikad ne znaju gdje se nalazi Pacman te nikad ne prate trag ukoliko je Pacman pojeo veliki kolač. Kada duh pojede trag postaje "užaren" te prati trag dok ga ne izgubi.

Politika odlučivanja

Politika odlučivanja ovog agenta može se ukratko opisati na način: duh se inicijalno giba nasumično sve dok ne naleti na trag agenta Pacman. Ukoliko je agent Pacman pojeo veliki kolač, duh ne prati trag, već dokle god je to stanje Pacmana aktivno, isti se vodi za promijenjenom strategijom te bježi od agenta Pacman, u protivnom duh mijenja boju u crvenu i nastavlja pratiti trag.

Čitav proces odlučivanja dotupan je unutar "gameobject_ghost.cpp", no budući da ga nije bilo potrebno modificirati (fokus projekta je na izradi uspješnog agenta Pacman) te zbog svoje veličine nije priložen u dokumentaciji.

Testiranje uspješnosti implementacije

Početno valja napomenuti da se pretpostavlja kako je Pacman "besmrtni", tj. ukoliko duh dohvati Pacmana, Pacman se vraća na inicijalnu poziciju, no igra se ne prekida, cilj testiranja je ustvrditi uspješnost implementacije na način da se razmatra u koliko je pokretanja igre Pacman uspio očistiti mapu prije nego li ga je ulovio neki od duhova. Kroz testiranje varirat će se parametri: broj duhova, veličina mape i vidljivost. Broj duhova u svakom pokretanju uvećavat će se za jedan. Maksimalan broj duhova za kojeg će se implementacija testirati jest 4. Inicijalno je broj duhova postavljen na 1. Veličina svijeta, početno se postavlja na 7x7 diskretnih jedinica, te se u svakom sljedećem pokretanju uvećava s korakom 4. Preostali parametar, vidljivost svijeta, inicijalno je postavljen na 1, te se također uvećava s korakom 1. Suviše velika vrijednost ovog parametra mogla bi zbuniti agenta stoga ju je potrebno držati u razumnim granicama. Pri svakom potezu postoji određena šansa da Pacman zapamti postojanje točno jednog kolača iz svoje okoline, u svim testnim primjerima izuzev zadnjeg, ista je postavljena na 70%.

U nastavku su navedeni rezultati testiranja za svaki pojedini testni slučaj (odabrani su samo neki od mogućih testnih slučajeva, no isti bi trebali biti dovoljni da demonstriraju rad aplikacije). Broj pokretanja po slučaju iznosi 10. Proizvoljne testove moguće je generirati promjenom vrijednosti netom navedenih parametara unutar "application.cpp" (parametri w – širina, h – visina, g – broj duhova, v – vidljivost).

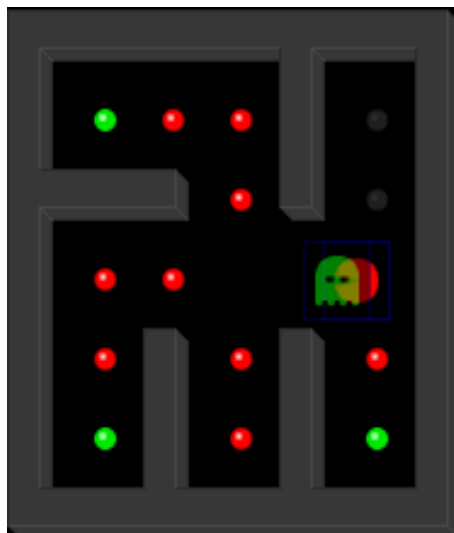
Slučaj 1:

Veličina mape: 7x7

Broj duhova: 1

Vidljivost: 1

Uspješnost: 10/10



Slika 3.1 Elementaran model svijeta s jednim duhom, u ovakvom svijetu Pacman u 100% slučajeva pobjeđuje agenta duga

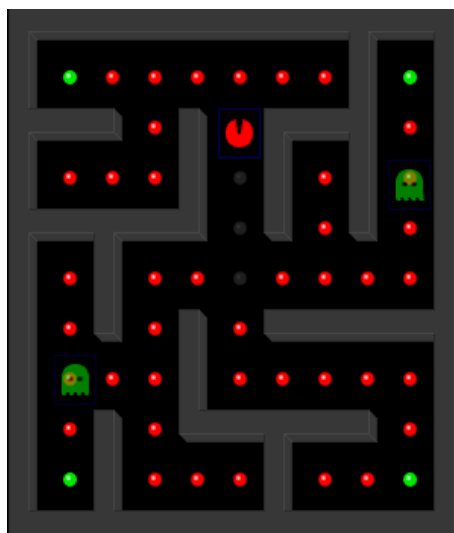
Slučaj 2:

Veličina mape: 11x11

Broj duhova: 2

Vidljivost: 2

Uspješnost: 7/10



Slika 3.2 Svijet sa dva duha. Vidljivost je u ovom slučaju postavljena na 2, Pacman je i dalje relativno uspješan.

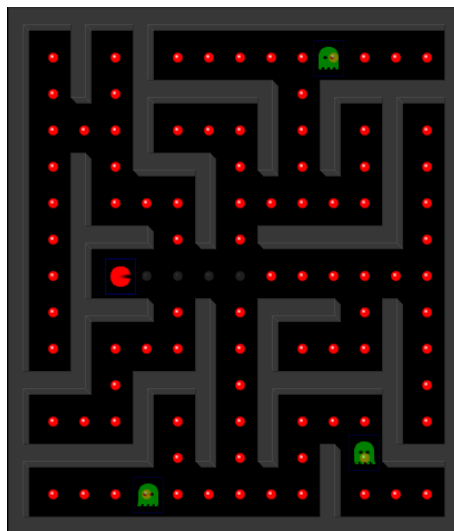
Slučaj 3:

Veličina mape: 15x15

Broj duhova: 3

Vidljivost: 3

Uspješnost: 6/10



Slika 3.3 Svijet sa 3 duha. Uspješnost agenta Pacman u ovom je slučaju nešto niža no u prethodna dva, kako bi se postigle bolje performanse može se smanjiti vidljivost.

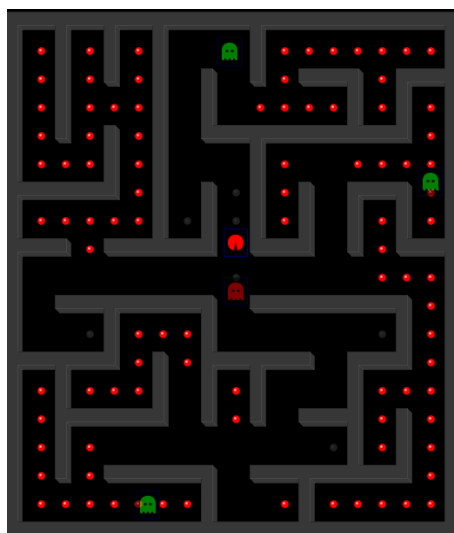
Slučaj 4:

Veličina mape: 19x19

Broj duhova: 4

Vidljivost: 4

Uspješnost: 1/10



Slika 3.4 Svijet sa 4 duha i vidljivošću 4. Agent je u ovom slučaju u potpunosti bezuspješan. Vidljivost je suviše visoka, stoga agent biva zbunjen s količinom informacija koju dobiva. Za bolje performanse potrebno je smanjiti vidljivost.

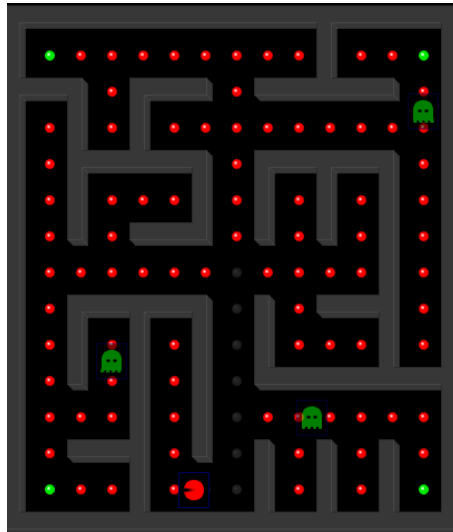
Slučaj 5:

Veličina mape: 15x15

Broj duhova: 3

Vidljivost: 2

Uspješnost: 7/10



Slika 3.5. Ponavljanje testiranja s tri duha, no sa smanjenom vidljivošću i povećanom mogućnosti pamćenja agenta (80%). Sada je agent malo uspješniji, premda valja napomenuti da na uspjeh agenta utječe i inicijalni izgled svijeta (raspored objekata na sceni).

Zaključak

Iz provedenih testova moguće je zaključiti da performanse agenta Pacman opadaju s povećanjem količine duhova što je i očekivano ponašanje. Ukoliko se vidljivost poveća suviše jako, primjerice za 4 duha i veličinu mape 19x19 postavljanje vidljivosti iznad 5 jedinica rezultira vrlo lošim agentom, budući da u tom slučaju agent iz okoline prima previše informacija koje ga zbunjuju prilikom procesa odlučivanja. U takvim slučajevima može se dogoditi da agent zapne, tj. titra u mjestu. Za takva izvođenja, također, su bilježeni negativni poeni. Valja naglasiti da je u pojedinim pokretanjima mapa generirana jako loše (koristi se gotov generator svijeta), stoga se rezultati izvođenja za takve mape nisu uzimali u obzir. Naposljetku, agent se puno bolje ponaša na malim mapama, nego na velikim budući da je fokus tijekom izrade projekta bio upravo na malim mapama i malom broju duhova.

Literatura

[1.] *Fundamentals of Multiagent Systems*, J.M Vidal 2006.