

Määrittelydokumentti

Kuvaus

Tarkoituksena on toteuttaa tekoäly pelille kivi-sakset-paperi ja sen mahdollisille laajennuksille kuten kivi-sakset-paperi-lisko-spock. Pelin Nashin tasapaino strategia olisi pelata satunnaisesti, niin että kaikkien vaihtoehtojen todennäköisyys tulla valituksi on yhtä suuri, mutta tämä strategia johtaa pitkällä aikavälillä ainoastaan tasapeliin ja eikä hyväksikäytä vastustajan mahdollisia virheitä tai ennustettavuutta. Pyritään toteuttamaan tekoäly, joka oppii ennustamaan vastustajan käytöstä tämän aiemman toiminnan mukaan. Ohjelmointikielenä toimii java.

Markovin ketju

Tekoälylle toteutetaan Markovin ketjujen ideaan perustuvalla algoritmilla. Prosessi on Markovin ketjun mukainen, jos prosessin tila tulevaisuudessa riippuu vain edeltävästä tilasta. Kivi-paperi-sakset pelissä edellinen tila voidaan ajatella vastustajan edellisten päätösten muodostamana tietyn pituinen "päätosketjuna". Ohjelma pitää kirjaa vastustajan pelihistoriasta ja katsoo mikä vastustajan seuraava liike todennäköisemmin on sen perusteella mitä päätöksiä hän on menneisyydessä tehnyt tilanteissa, joissa hänen päätösketjunsä on ollut vastaava. Jos esimerkiksi kivi, kivi, paperi ketjun jälkeen vastustaja on valinnut 60% ajasta sakset, tekoäly valitsee kiven (tai ainakin painottaa) vastaavassa tilanteessa. Pelin alkua lukuun ottamatta tekoäly ei siis valitse omaa päätöstään satunnaisesti.

Työssä tullaan kokeilemaan tekoälyä eripituisilla päätösketjuilla.

Tietorakenteet

Työssä tullaan käyttämään hakupuuta apuna algoritmin etsiessä vastustajan todennäköistä seuraavaa päätöstä hänen tekemän päätösketjun perusteella. Jokainen solmu sisältää tiedon kuinka monta kertaa solmuun on päädytty ja mitä valintaa solmu edustaa (kivi, paperi vai sakset). Koska jokaista päätöstä on mahdollista seurata kolmella eri valinnalla (viidellä, jos lisko ja spock lisätään mukaan) on jokaisella solmulla kolme lasta. Puun tilavaativuus on siis $O(3^n)$, jossa n on päätösketjun pituus ja puun korkeus. Solmun hakeminen vie $O(n)$ aikaa.

Puun solmuihin voidaan lisätä myös tieto kuinka monta pistettä vastustaja on saanut kyseisellä päätösketjulla, jolloin tekoälylle voidaan toteuttaa painotuksia, oletuksella, että vastustaja todennäköisemmin valitsee päätöksiä, jotka ovat toimineet hänelle menneisyydessä.

Käytetään pinoa vastustajan sen hetkisen päätösketjun tallentamiseen.

Lähteet:

http://en.wikipedia.org/wiki/Markov_chain

<http://1.618034.com/roshambo-markov-chains-and-a-teaspoon-of-lulz/>

<http://www.cs.helsinki.fi/u/floreen/tira2013syksy/tira.pdf>

