

ShakkiAI - Toteutusdokumentti

Antero Vainio

January 12, 2014

Ohjelman yleisrakenne Varsinainen harjoitustyön osuus koko ohjelmakoodista kuuluu projektikansion ai-pakkaukseen, joskin joitain ai-pakkauksen luokkia käytetään myös hyväksi pelilogiikassa, kuten esimerkiksi shakin mattitilanteiden tarkistamisessa.

Ohjelma hyödyntää mahdollisten siirtojen läpikäymisessä n.s. Nappulahallinta-luokkaa, joka koostuu luokan Siirettava alkioista muodostetusta pinosta. Kukin Siirettava-alkio koostuu puolestaan kyseisen nappulan lähtökoordinaateista, ja pinosta mahdollisia maalikoordinaatteja.

Saavutetut aika- ja tilavaativuudet Nappulahallinta-luokan alkio kootaan 'skannaamalla' senhetkisestä kentästä halutun pelaajan pelinappulat, ja jokaisen nappulan kohdalla skannaamalla samasta kentästä lailliset maalikoordinaatit, joiden pohjalta voidaan koota kaikki mahdolliset siirrot. Huom. koska pelikenttä on aina 8*8 taulukko, tapahtuu tällaisen operaation tekeminen aina vakioaikavaativuusluokassa, mutta älykkäällä skannauksella voisi karsia tutkittavia peliruutuja pois ainakin tiettyjen pelinappuloiden kohdalla.

Itse siirronvalinta-algoritmi toimii, kuten määrittelydokumentissa 'tavoiteltua', ajassa $O(2^n)$, eikä siihen siis ole tällä hetkellä tiedossa tehokkaampaa ratkaisua. Kuitenkin optimoimalla, ja karsimalla pelipuun tarpeettomia alipuita, saisi siirronvalintaa nopeutettua, kun se koostuu kuitenkin enimmillään kolmen, neljän, tai viiden korkuisesta pelipuusta (kuudenkin korkuisen pelipuun tutkiminen olisi mahdollista hyvin karsitulla algoritmilla).

Muita puutteita ja parannusehdotuksia Koneoppimisen soveltaminen shakkibottiin on jäänyt täysin huomioimatta, ja sillä saisi paitsi varmasti älykkäämmän tekoälyn, myös mielenkiitoisempia botti vastaan botti -shakkipelejä, tällä hetkellä kun samoilla asetuksilla, kaikki botti vastaan botti -pelit ovat identtisiä.