

Käytettävät algoritmit ja tietorakenteet

Keko (minimikeko)

A*

Dijkstra / BMF (testausta varten)

Ongelma ja sen ratkaisu

Harjoitustyön ongelmana on reitin löytäminen kartalla pisteestä A pisteeseen B. Esimerkiksi peleissä tähän ongelmaan törmätään hyvinkin usein, ja siksi yritänkin simuloida ohjelmallani pelissä mahdollisesti vastaan tulevaa tilannetta. Karttaa mallinnetaan kaksiulotteisena taulukkona (verkkona), joka koostuu ruuduista (solmu). Jos harjoitustyön tekemiseen käytettävissä oleva aika riittää, tulen käyttämään staattisen kartan sijasta koko ajan vierittyvää ja generoituvaa karttaa, jolla reittiä pitää jatkuvasti laskea uudestaan. Ongelman ratkaisuna tulen käyttämään hyväksi todettua A*-algoritmia, mahdollisesti myös muita vaihtoehtoja kokeillen. A*:n pitäisi olla keskimääräisesti esim. Dijkstraa tehokkaampi, ja heuristiikkaa säätämällä algoritmi taipuneekin myös jatkuvaan lennosta tehtävään reitinlaskentaan.

Ohjelman syötteet ja niiden käyttö

Ohjelman ytimenä tulee toimimaan reitin löytämiseen käytettävä algoritmi, joka saa parametrina solmuista koostuvan taulukon. Solmuja hyväksikäyttämällä lasketaan lyhyin mahdollinen reitti, tallennetaan se kekkoon ja visualisoidaan käyttäjälle kyseinen reitti.

Tavoitteet aika/tilavaatimusten osalta

A*:n aikavaatimus riippuu käytettävästä heuristiikasta, mutta pahimmassa tapauksessa algoritmin pitäisi toimia $O((|E|+|V|)\log|V|)$ ajassa. Jos pääsen työssä vaiheeseen jossa reittiä pitää aktiivisesti päivittää, uskon heuristiikan vaikuttavan olennaisesti suoritussykyyn. Tilavaatimus on pahimmillaan $O(|V|^2)$, vaikkakin tämä skenaario tuntuu epätodennäköiseltä kartalla, jossa algoritmi tulee törmäämään väistämättä johonkin esteeseen joka ohjaa reitin laskentaa niin, ettei kaikkia mahdollisuuksia tarvitse käydä läpi.

Lähteet

Tietorakenteet ja algoritmit -kurssin kalvot, Floreen 2013

A* Pathfinding for beginners (<http://www.policyalmanac.org/games/aStarTutorial.htm>)

A* Algorithm tutorial (<http://heyes-jones.com/astar.php>)