

## Referaatti artikkelista A Comprehensive Study on Pathfinding Techniques for Robotics and Video Games

Santeri Martikainen  
20.9.2016

Kirjoittajat Zeyad Abd Algfoor, Mohd Shahrizal Sunar ja Hoshang Kolivand käsittelevät artikkelissaan reitinetsintäalgoritmeja, joita sovelletaan muun muassa GPS-navigoinnin, videopelien, robotiikan ja ihmisjoukkojen liikkeen simuloinnin saralla. Aihetta lähestytään artikkelissa lähinnä videopelien näkökulmasta ja siinä siinä käsitellään lähinnä erilaisia toimintaympäristön mallinnustekniikoita, sekä niihin erityisesti sopivia algoritmeja.

Reitinetsintäalgoritmien tutkimus keskittyy nykyisellään tehokkuuteen ja realististen reittien löytämiseen annetussa ympäristössä. Algoritmien toimintaympäristö ja niille asetetut vaatimukset voivat vaihdella paljonkin.

Ei ole olemassa yleispätevää reitinetsintäalgoritmia, joka sopisi kaikkiin tilanteisiin, joten eri tarpeisiin ja ympäristöihin on kehitetty erilaisia ratkaisuja. Eri algoritmeilla on erilaisia resurssitarpeita ajan- ja muistinkäytön suhteen. Niiden soveltuvuuteen halutuun tehtävään vaikuttaa voimakkaasti toimintaympäristön topologia ja se onko ympäristö staattinen vai dynaaminen.

Reitinetsintä kokonaisuutena jakaantuu kahteen vaiheeseen: Toimintaympäristöstä muodostetaan ensin yksinkertaistettu malli, minkä jälkeen sitä käydään läpi jollakin algoritmilla reittien löytämiseksi.

Toimintaympäristön abstraktointi voidaan jakaa kahteen pääluokkaan: Verkkoihin (grid) ja hierarkkisiin tekniikoihin.

### **Verkot, ruudukot**

Yksi tapa yksinkertaistetun mallin luomiseen on muodostaa käsiteltävästä ympäristöstä graafi. Graafin solmuja käsitellään sitten sijainteina joihin voi siirtyä ja joiden avulla muodostetaan haluttu reitti kohteeseen. Solmun naapurisolmut voivat olla yksinkertaisesti lähimpiä paikkoja mihin nykyisestä sijainnista voi siirtyä, tai sitten niitä voidaan käsitellä myös siten, että kahden toisiinsa liittyvän solmun välillä katsotaan olevan suora näköyhteys ja esteetön

kulku.

Toinen yleinen tapa käsitellä topologiaa on jakaa alue soluihin. Nämä ovat muodoltaan yleensä polygoneja, useimmiten joko kolmioita, neliöitä tai kuusikulmioita. Näiden kolmen muodon etuna on, että niitä käyttämällä voidaan kattaa koko alue yhtenäisenä ruudukkona ilman väliinjääviä katvealueita.

Verkkoihin perustuvissa malleissa on kuitenkin kaksi perustavanlaatuista ongelmaa: Reitit muodostuvat lukuisista vierekkäisistä solmuista mikä rajaa mahdollisuuksia päästä optimaalisiin tuloksiin. Toisekseen, mikäli solmuja on hyvin paljon, tämä johtaa varsin suureen muistinkäyttöön.

## **Hierarkkiset tekniikat**

Reitinetsintään luodun topologian ei välttämättä tarvitse olla symmetrinen, vaan esimerkiksi esteiden lähellä voidaan käyttää hienojakoisempaa jaottelua, siinä missä vaikkapa esteettömillä aukeilla riittää suurpiirteisempi lähestymistapa.

Quadtree on eräs tätä ajattelua seuraavista ratkaisumalleista. Siinä alue on jaettu neliöihin, ja mikäli jonkin neliön sisällä on jonkinlainen este, kyseinen neliö jaetaan neljään pienempään neliöön jne, kunnes joko saadaan neliöitä joiden sisällä ei ole mitään, tai ollaan saavutettu pienin sallittu neliökoko.

## **Rinnakkainen reitinhaku**

Oman ongelmansa tuo videopeleissä hyvin yleinen tilanne, missä saman ympäristön jakaa useampi toimija. Kyse voi olla esimerkiksi samaan päämäärään suuntaavista hahmoista, joiden kaikkien tulee löytää kohteeseen toisiaan mahdollisimman vähän häiriten. Niinikään mahdollisesti liikkuvaan kohteeseen hakeutuvan toimijan tulee pystyä muuttamaan reittiään dynaamisesti sitä mukaa kun tilanne muuttuu.