

# Määrittelydokumentti

## Kuvaus

Tarkoituksena on toteuttaa reitinhaku algoritmi, joka etsii kahden solmun välisen parhaan reitin verkossa. Parhaalla reitillä tarkoitetaan ”halvinta” polkua, jossa kahden solmun välinen kustannus lasketaan niiden välisenä euklidisena etäisyytenä. Verkon solmut ovat koordinaatteja  $R^n$  euklidisessa avaruudessa.

Ohjelman syötteeksi annetaan verkko ja lähtösolmu ja maalisolmu. Ohjelma antaa tulosteena parhaan polun ja sen kustannuksen.

Työn ohjelmointikieli on Java.

## A\* haku algoritmi

A\* algoritmi muistuttaa Dijkstran algoritmia, mutta arvioi heuristisesti tutkittavien solmujen kautta kulkevien polkujen kustannukset ja käy parhaalta vaikuttavat vaihtoehdot ensin läpi. Arvio polkukustannuksesta on todellisen kustannuksen lähtösolmusta tutkittavaan solmuun  $N$  ja heuristisen arvion kustannuksesta  $N$  solmusta maalisolmuun välinen summa.

Oleellista on, että heuristiikka antaa enintään yhtä suuren arvion kuin todellinen kustannus, muuten algoritmi ei välttämättä tuota optimaalisinta ratkaisua. Työssä käytettävä heuristiikkafunktio palauttaa solmujen välisen euklidinen etäisyys, joka onkin solmujen välinen pienin mahdollinen etäisyys.

A\* algoritmin aikavaativuus on pahimmillaan  $O((|V| + |E|)\log|V|)$ , jossa  $V$  on verkon solmujen joukko ja  $E$  kaarten joukko. Käytännössä aikavaativuus kuitenkin pyritään saaman pienemmäksi heuristiikkafunktion avulla.

A\* algoritmin tilavaativuus on  $O(|V|)$ .

## Tietorakenteet

A\* algoritmin toiminta vaatii prioriteettijonon, johon tutkittavia solmuja pidetään niiden kustannusarvioiden mukaisessa järjestyksessä. Prioriteettijono toteutetaan mininimikekona.

## Lähteet:

[http://en.wikipedia.org/wiki/A\\*\\_search\\_algorithm](http://en.wikipedia.org/wiki/A*_search_algorithm)

<http://www.cs.helsinki.fi/u/floreen/tira2014/tira.pdf>