

Lentokenttä simulaattori  
Tietorakenteet ja algoritmit

Antti Venetjoki

PROJEKTI

Maaliskuu 2024

SISÄLLYS

[1 Tavoiteltu arvosana 3](#_Toc161960860)

[2 Todisteet toiminnasta 4](#_Toc161960861)

[3 Koodiselostus 9](#_Toc161960862)

[3.1 Yksi kiitotie 9](#_Toc161960863)

[3.2 Tiukka kaksinkertainen kiitotie 9](#_Toc161960864)

[3.3 Dynaaminen kaksinkertainen kiitotie 9](#_Toc161960865)

[3.4 Kolminkertainen kiitotie 9](#_Toc161960866)

[3.5 Yksi kiitotie polttoaineen etusijalla 9](#_Toc161960867)

[3.6 Yleistä 10](#_Toc161960868)

[4 11](#_Toc161960869)

[4.1 Tiukka kaksinkertainen kiitotie vs. Yksi kiitotie 11](#_Toc161960870)

[4.1.1 Tulosten vertailu 11](#_Toc161960871)

[4.1.2 Analyysi 11](#_Toc161960872)

[4.2 Dynaaminen kaksinkertainen kiitotie vs. Yksi kiitotie 11](#_Toc161960873)

[4.2.1 Tulosten vertailu 11](#_Toc161960874)

[4.2.2 Analyysi: 11](#_Toc161960875)

[5 Projektiin käytetty aika 13](#_Toc161960876)

# Tavoiteltu arvosana

Tavoite arvosana on viisi (5).

s-posti: [antti.venetjoki@tuni.fi](mailto:antti.venetjoki@tuni.fi)

puh. Ota yhteyttä sähköpostitse

# Todisteet toiminnasta

A black background with white text

Description automatically generated

Kuva 1 Todiste onnistuneesta kääntämisestä

A screenshot of a computer program

Description automatically generated

Kuva 2 Todiste onnistuneesta ajosta P1

A screenshot of a computer program

Description automatically generated

Kuva 3 Todiste onnistuneesta ajosa P2

**A screenshot of a computer program

Description automatically generated**

Kuva 4 Todiste onnistuneesta ajosta P3

A screenshot of a computer program

Description automatically generated

Kuva 5 Todiste onnistuneesta ajosta P4

A screenshot of a computer program

Description automatically generated

Kuva 6 Todiste onnistuneesta ajosta P5

# Koodiselostus

Suunnittelu kattaa useita erilaisia lentokenttäsimulaation ohjelmaversioita. Jokainen versio vastaa erilaisia kiitoteitä ja toimintastrategioita. Versiot sisältävät:

## Yksi kiitotie

Tässä versiossa simuloidaan lentokenttää, jossa on vain yksi kiitotie, jolla tapahtuvat sekä saapumiset että lähdöt. Se kuvaa perus lentokenttätoimintoja yhdellä jonotusjärjestelmällä sekä laskeutumisille että lentoonlähdöille.

## Tiukka kaksinkertainen kiitotie

Tässä lentokentällä on kaksi erillistä kiitotietä, joista toinen on omistettu vain lähdöille ja toinen vain saapumisille. Tämä erottelu optimoi liikenteen virtausta estämällä konflikteja saapuvien ja lähtevien lentokoneiden välillä.

## Dynaaminen kaksinkertainen kiitotie

Samanlainen kuin tiukka kaksinkertainen kiitotieversio, tässä suunnittelussa on kaksi kiitotietä, mutta dynaaminen allokaatiotaktiikka. Yleensä toinen kiitotie on varattu lähdöille ja toinen saapumisille, mutta sitä muutetaan liikennetilanteen perusteella tehokkuuden maksimoimiseksi.

## Kolminkertainen kiitotie

Tässä versiossa lentokentällä on kolme kiitotietä, joista yksi on varattu sekä laskeutumisille että lentoonlähdöille, ja kaksi muuta ensisijaisesti laskeutumisille. Jos kuitenkin ensisijaisen kiitotien laskeutumisjono on tyhjä, toinen lisäkiitotiistä voidaan käyttää lähtöihin ruuhkan lieventämiseksi.

## Yksi kiitotie polttoaineen etusijalla

Tämä versio laajentaa yhden kiitotien simulointia ottamalla käyttöön polttoaineen etusijajärjestelmän. Matalalla polttoainetasolla olevat lentokoneet saavat etusijan laskeutumiseen, mikä varmistaa turvallisuuden estämällä polttoaineen loppumisen tapaukset.

## Yleistä

Jokainen versio käyttää tiettyjä kiitotiekonfiguraatioita ja toimintaperiaatteita, jotka on räätälöity lentokentän tehokkuuden ja turvallisuuden optimoimiseksi erilaisissa olosuhteissa. Suunnittelut käsittelevät erilaisia liikennemääriä, kiitotien kapasiteettia ja jononhallintastrategioita simuloidakseen realistisia lentokenttätilanteita.

# Vertailu

Eri lentokenttäsimulaatioiden suorituskykyä voidaan analysoida vertaamalla kunkin simulointiajon tuloksia ja vertaamalla lentokentän kapasiteettia ja tehokkuutta erilaisissa konfiguraatioissa. Tarkastellaan useampia kiitoteitä sisältäviä vaihtoehtoja verrattuna yhden kiitotien versioon:

## Tiukka kaksinkertainen kiitotie vs. Yksi kiitotie

### Tulosten vertailu

Tiukan kaksinkertaisen kiitotien version tulisi teoriassa tarjota suurempi kapasiteetti, koska saapumiset ja lähdöt erotetaan kahdelle erilliselle kiitotielle.

Tulosten tarkastelun tulisi osoittaa suurempi lentokoneiden läpimenoaika tiukan kaksinkertaisen kiitotien version ja yhden kiitotien version välillä.

### Analyysi

Kapasiteetin kasvu tiukassa kaksinkertaisessa kiitotieversiossa johtuu siitä, että saapumiset ja lähdöt on eroteltu omille kiitoteilleen, mikä vähentää mahdollisia konflikteja ja mahdollistaa laskeutumisen ja lentoonlähdön samanaikaisen suorittamisen.

Toimintojen erottaminen kasvattaa lentokentän kokonaiskäyttötehoa, mikä vähentää viiveitä ja parantaa läpimenoaikaa.

## Dynaaminen kaksinkertainen kiitotie vs. Yksi kiitotie

### Tulosten vertailu

Samoin kuin tiukan kaksinkertaisen kiitotien versiossa, myös dynaamisella kaksinkertaisella kiitotieversiolla tulisi olla suurempi kapasiteetti verrattuna yhden kiitotien versioon, koska kiitoteitä voidaan käyttää dynaamisesti liikennetilanteen mukaan.

Tuloksissa näkyy muuttuva kiitoteiden käyttö, jossa lentokenttä käyttää tehokkainta konfiguraatiota reaaliaikaisen liikennetilanteen perusteella.

### Analyysi:

Kapasiteetin kasvu dynaamisessa kaksinkertaisessa kiitotieversiossa johtuu siitä, että kiitoteitä voidaan käyttää dynaamisesti liikennetilanteen mukaan, mikä optimoi kiitoteiden käytön ja vähentää viiveitä.

Kiitoteiden allokoimisen säätäminen mahdollistaa lentokentän tehokkaamman reagoinnin liikennemäärien vaihteluihin, mikä maksimoi läpimenokyvyn ja tehokkuuden.

Vertaamalla näiden varianttien tuloksia voimme havaita, kuinka useiden kiitoteiden käyttö lisää lentokentän kapasiteettia ja tehokkuutta. Toimintojen erottaminen ja kiitoteiden dynaaminen allokoiminen mahdollistavat lentokentän käsittelykyvyn kasvattamisen samalla kun operatiivinen tehokkuus säilyy.

# Projektiin käytetty aika

Projektiin ensimmäiseen osioon (P1) käytettiin huomattavasti eniten aikaa koska annettu lähdekoodi oli osittain väärin, huonosti jaoteltu ja erittäin epäselvä.

Projektin muihin osioihin meni vähemmän aikaa koska minulla oli jonkinlainen käsitys alkuperäisen lähdekoodin toiminnasta. Yhteensä käytin projektin ohjelmointiin n. kymmenen tuntia. Tarkkaa tuntikirjanpitoa en tehnyt.