

S-38.3148 Tietoverkkojen simulointi

11.9.2006

1

Teoria

- Johdanto simulointiin
 - Simulointi menetelmänä
 - Erilaiset simulointimallit
 - Mallinnus, toteutus ja validointi
- Simuloinnin kulku -- prosessin realisaatioiden tuottaminen
- Satunnaismuuttujan arvonta annetusta jakaumasta
- Tulosten keruu ja analyysi
- Varianssinreduktiotekniikoista

11.9.2006

2

Mitä simulointi on?

- Tietokoneen avulla jäljitellään eli simuloidaan jonkin todellisen laitteiston tai prosessin toimintaa
- Laitteistoa tai prosessia kutsutaan järjestelmäksi
- Voidaksemme tutkia järjestelmää tieteellisesti, joudumme tekemään olettamuksia siitä, miten järjestelmä toimii
- Nämä olettamukset muodostavat järjestelmää kuvaavan matemaattisen mallin
 - olettamukset ovat tavallisesti matemaattisia tai loogisia relaatioita

11.9.2006

3

Mitä simulointi on (jatkoa)?

- Jos mallin sisältämät relaatiot ovat kyllin yksinkertaisia, saattaa olla mahdollista matemaattisen analyysin kautta saada eksaktia tietoa järjestelmästä
- Useimmat reaali maailman järjestelmät ovat kuitenkin liian monimutkaisia, jotta realistisia malleja kyettäisiin käsittelemään analyttisesti
- Tällaisissa tapauksissa voidaan turvautua simulointiin
 - simulointi on tietyssä mielessä “viimeinen keino”
 - usein se on ainoa vaihtoehto
- Simuloinnissa tietokonetta käytetään mallin numeeriseen jäljittelyyn
- Simuloinnin aikana kerätään tietoa haluttujen suureiden estimoimiseksi

11.9.2006

4

Tietoliikenneverkkojen simuloinnin tavoitteet

- Järjestelmän suorituskyvyn parantaminen
 - viiveet, läpäisy, jne
- Varmistuminen suorituskykytavoitteiden toteutumisesta ennen hankintapäätöstä
- "Paikallisten" muutosten koko verkkoon kohdistuvien vaikutuksen arviointi
- Pullonkaulojen identifiointi ennen järjestelmän käyttöönottoa
- Kustannusten vähentäminen
- Järjestelmän kehittämisajan lyhentäminen

11.9.2006

5

Tyypillisiä kysymyksiä, joihin etsitään vastausta

- Liikenteelliset kysymykset:
 - Miten verkko toimii kuormaa lisättäessä?
 - Kuinka suuret puskurit tarvitaan verkon reitittimissä?
 - Riittääkö web-palvelimen teho hoitamaan kuorman kasvun siten että vasteajat eivät kasva liikaa?
 - Mikä TCP "versio" antaa parhaan suorituskyvyn?
- Verkon topologia:
 - Mikä on edullisin verkon rakenne?
 - Pitäisikö yrityksen verkko aliverkottaa kasvaneen työasemamäärän takia?
 - Montako satelliittia tarvitaan tietyn tasoisen palvelun saamiseksi kahden maa-aseman välillä?
- Verkon luotettavuus:
 - Mitä seuraa linkin vikaantumisesta?
 - Kuinka nopeasti reititys konvergoituu vikatilanteen jälkeen?

11.9.2006

6

Simuloinnin kohteena usein esiintyviä suureita

- Läpäisy (esim. Mbit/s)
- Päästä-päähän -viive
- Viive verkon kahden pisteen, A ja B, välillä
- Pakettien lukumäärä puskurissa
- Solmujen ja linkkien käyttöaste
- Yhteyden estotodennäköisyys
- Katkeavien puheluiden (hand-overin epäonnistumisen) todennäköisyys mobiiliverkossa
- Pakettien törmäys- ja vetäytymistodennäköisyys (LAN)

11.9.2006

7

Simuloinnin etuja

- Voidaan tutkia monimutkaisia järjestelmiä
- Voidaan tutkia vasta suunnitteilla olevan järjestelmän toimintaa rakentamatta sitä
 - verrata suunnitteluvaihtoehtoja tai erilaisia operointipolitiikkoja
 - joskus pelkkä simulointimallin tekeminen on tärkeämpää kuin itse simulointi
- Voidaan tutkia olemassaolevan järjestelmän toimintaa uusissa käyttöolosuhteissa
 - simuloinnin avulla voidaan helpommin ja tarkemmin säädellä "koeparametreja" kuin todellisessa järjestelmässä
- Simuloinnin avulla voidaan tutkia järjestelmän käyttäytymistä pitkien ajanjaksojen yli (nopeutetussa aikaskaalassa)
 - tai kääntäen hyvin nopeita ilmiöitä hidastetussa aikaskaalassa

11.9.2006

8

... ja rajoituksia

- Esteenä simuloinnin käytölle saattaa olla, että hyvin monimutkaisten järjestelmien kysymyksessä ollen
 - myös simulointimallin luominen voi olla hyvin työlästä
 - simulointiohjelman kirjoittaminen saattaa olla erittäin suuri tehtävä
 - tehtävän helpottamiseksi on kehitetty erilaisia simulointikieliä ja simulointityökaluja
 - simulointiohjelman ajaminen voi viedä pitkän ajan
- Simulointi ei ole yksinomaan eikä ensi sijassa ohjelman kirjoittamista
 - tärkeää on itse malli
 - miten simulointia käytetään johtopäätösten tekemiseen

11.9.2006

9

Suorituskyvyn arviointi: analyysi vs. simulointi

- Simulointi on (liikenneteorian kannalta) eräs tilastollinen menetelmä tarkasteltavan järjestelmän suorituskyvyn arvioimiseksi
- Se sisältää neljä eri vaihetta
 - järjestelmän (olemassa olevan tai kuvitteellisen) mallinnus dynaamisena (ajassa kehittyvänä) stokastisena prosessina
 - prosessin realisaatioiden tuottaminen (“todellisuuden havainnointi”)
 - tietojen keruu (“mittaus”)
 - kerättyjen tietojen tilastollinen analyysi ja johtopäätösten teko
- Matemaattiseen analyysi on vaihtoehtoinen lähestymistapa
- Se sisältää vain kaksi vaihetta
 - järjestelmän mallinnus ajassa kehittyvänä stokastisena prosessina
 - mallin analyttinen ratkaisu
- Järjestelmän mallinnusvaihe on kummallekin yhteinen
 - tosin mallin tarkkuudella voi olla suuriakin eroja: tosin kuin simulointi, matemaattinen analyysi edellyttää yleensä hyvinkin rajoittavien oletusten tekoa

11.9.2006

10

Suorituskyvyn arviointi: analyysi vs. simulointi (jatkoa)

- Matemaattisen analyysin haitat
 - Asettaa rajoittavia ehtoja mallinnukseen
=> malli yleensä liian yksinkertainen
=> monimutkaisten järjestelmien suorituskyvyn arviointi lähes mahdotonta
 - Rajoittavien ehtojenkin vallitessa analyysi itsessään yleensä vaikeaa; tarvitaan pitkälle erikoistuneita ekspertejä
 - Tulokset saattavat
 - rajoittua vain tasapainotilaan
 - koskea vain keskiarvoja
- Matemaattisen analyysin edut
 - Tulosten tuottaminen nopeaa
 - Tulokset tarkkoja
 - Antaa näkemystä
 - Optimointi usein mahdollista (vaikkakin saattaa olla vaikeaa)

11.9.2006

11

Suorituskyvyn arviointi: analyysi vs. simulointi (jatkoa)

- Simuloinnin edut
 - Ei rajoittavia ehtoja mallinnusvaiheessa
=> mahdollistaa monimutkaistenkin järjestelmien suorituskyvyn arvioinnin
 - Mallinnus yleensä hyvin suoraviivaista
- Simuloinnin haitat
 - Tulosten tuottaminen yleensä työlästä (simulointiajot vaativat paljon prosessori-aikaa)
 - Tulokset epätarkkoja (tosin tarkentuvia: mitä enemmän ajoja, sitä tarkemmat tulokset)
 - Kokonaisnäkemys saaminen vaikeampaa
 - Optimointi mahdollista vain hyvin rajoitetusti (esim. muutaman erilaisen "parametrikombinaation" tai ohjausperiaatteen vertailu)

11.9.2006

12

Karikoita simuloinnin käytössä

- Simulointiin ryhdyttäessä ei ole selviä tavoitteita
 - järjestelmää tai olennaisia kysymyksenasetteluja ei ymmärretä
 - “eikun simuloimaan ehkä se siitä selviää”
- Projektin kuluessa ei pidetä riittävästi yhteyttä toimeksiantajiin
- Mallin sopimaton tarkkuustaso (liikaa / liian vähän yksityiskohtia)
 - liian tarkalla kuvaustasolla simulointiohjelma saadaan helposti tukehtumaan
 - liian karkea malli riippuu lähtötiedoista, joita ei ehkä ole saatavilla
- Pääpainon paneminen simulointiohjelman luomiseen
 - ikään kuin tehtävä olisi vain vaativa ohjelmointiongelma
- Riittävän asiantuntemuksen puute
 - kohdejärjestelmästä, suorituskykyanalyysistä, tilastollisesta analyysistä
- Väärien työkalujen käyttö
- Liiallinen luottaminen kaupallisiin simulaattoreihin
 - luullaan että “kuka vain” voi käyttää simulaattoria
 - animaatioiden väärinkäyttö

11.9.2006

13

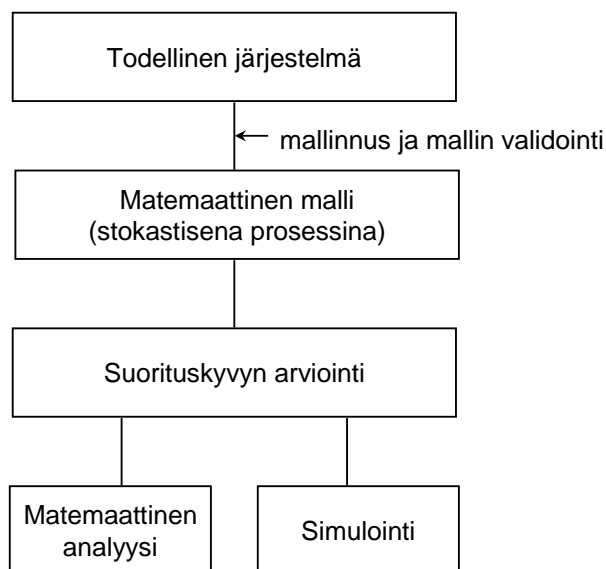
Karikoita simuloinnin käytössä (jatkoa)

- Puutteellinen / virheellinen järjestelmän satunnaisuuksien huomioonottaminen
 - lähtöjakaumat valitaan mielivaltaisesti tai perusteettomasti (esim. normaalijakauma)
- Tulosten analysointi yhdestä ajosta
 - simulointi pitää toistaa useita kertoja tulosten tarkkuuden selvittämiseksi
 - yhtä pitkää ajoa käytettäessä pitää huomioida tilastolliset riippuvuudet
- Väärin käsitellyt alkuehdot
 - simuloinnin alkuvaihe (lähtee annetusta tilasta) ei ole edustava ja pitää hylätä
- Liian lyhyt simulointi
- Heikko satunnaislukugeneraattori
- Virheellinen siemenlukujen käyttö
 - tietyssä simuloinnissa eri tarkoitukseen käytettyjä satunnaislukusekvenssejä ei saa aloittaa samalla siemenluvulla (esim. 0:lla), koska tämä johtaa virheellisiin riippuvuuksiin muuttujien kesken
 - mieluiten sekvenssit eivät ole miltään osin päällekkäisiä

11.9.2006

14

Liikenneteoreettisen järjestelmän suorituskyvyn arviointi



11.9.2006

15

Stokastisen prosessin simuloinnin vaiheet

- Järjestelmän mallinnus ajassa kehittyvänä stokastisena prosessina
- Prosessin realisaatioiden tuottaminen
 - satunnaislukujen generointi
 - tapahtumaohjattu simulointi
 - usein simuloinnilla tarkoitetaan pelkästään tätä vaihetta (liikenneteorian kannalta se on kuitenkin simulointia suppeammassa mielessä)
- Tietojen keruu
 - transientti vaihe vs. tasapainotila
- Tilastollinen analyysi ja johtopäätökset
 - piste-estimaattorit
 - luottamusvälit

11.9.2006

16

Teoria

- Johdanto simulointiin
 - Simulointi menetelmänä
 - Erilaiset simulointimallit
 - Mallinnus, toteutus ja validointi
- Simuloinnin kulku -- prosessin realisaatioiden tuottaminen
- Satunnaismuuttujan arvonta annetusta jakaumasta
- Tulosten keruu ja analyysi
- Varianssinreduktiotekniikoista

11.9.2006

17

Järjestelmä

- Järjestelmä on joukko keskenään vuorovaikuttavia entiteettejä
- Järjestelmän tila on joukko muuttujia, jotka tietyllä hetkellä riittävät kuvaamaan järjestelmän tutkimuksen kannalta riittävällä tarkkuudella
 - varattuina olevien johtojen lukumäärä, pakettien lukumäärä puskureissa jne
- Diskreetti järjestelmä
 - tilamuuttujien arvot muuttuvat hyppäyksittäin diskreeteillä ajanhetkillä
 - "digitaaliset" järjestelmät
- Jatkuva järjestelmä
 - tilamuuttujat muuttuvat jatkuvasti ajan funktiona
 - niiden välisiä riippuvuuksia kuvaavat usein differentiaaliyhtälösysteemit
 - "analogiset" järjestelmät
- Monasti valinta diskreetin ja jatkuvan järjestelmäkuvauksen välillä on mallin suunnittelussa tarkoituksenmukaisuuskysymys
 - riippuu siitä, minkä aikaskaalatason tapahtumista ollaan kiinnostuneita
 - diskreettiä järjestelmää voidaan karkeammalla tasolla katsottuna pitää jatkuvana järjestelmänä (esim. "nestejonot")

11.9.2006

18

Simulointimallit

- Staattinen simulointimalli
 - koskee tiettyä ajanhetkeä tai systeemin tasapainotilaa -- ajalla ei ole mitään merkitystä
 - tunnuslukujen arvioiminen tunnetusta tasapainojakaumasta
 - moniulotteisten integraalien numeerinen integrointi ns. Monte-Carlo-menetelmällä
- Dynaaminen simulointimalli
 - kuvaa systeemin kehittymistä ajassa
- Deterministinen malli
 - ei sisällä mitään satunnaisuutta
 - kun tulosuureet on annettu, lähtösuureet määräytyvät yksikäsitteisesti (mahdollisesti hyvin monimutkaisten relaatioiden kautta)
- Stokastinen malli
 - ottaa huomioon, että reaali maailma sisältää aina satunnaisuutta (puhelujen saapumishetket ja -kestot)
 - annetuilla tulosuureilla lähtösuureet ovat stokastisia
 - simuloinnin avulla voidaan kartoittaa niiden jakaumaa ja muodostaa estimaatteja tunnusluvuille (esim. odotusarvoille)

11.9.2006

19

Simulointimallit (jatkoa)

- Termin Monte Carlo -simulointi käytössä on horjuvuutta
- Useimmiten sillä tarkoitetaan *staattista* simulointia
 - klassillinen esimerkki on Buffonin neula:
 - d:n pituinen neula heitetään umpimähkään lattialle, jossa lautojen leveys on d
 - todennäköisyys, että neula osuu lautojen raon päälle, on $2/\pi$
 - toistamalla heittoja saadaan estimaatti $2/\pi$:lle
- Toisinaan Monte Carlo -simuloinnilla tarkoitetaan ylipäätään *stokastista* simulointia (siis myös dynaamista stokastista simulointia)
 - tämä on termin alkuperäinen käyttö
 - termin ottivat käyttöön J. von Neumann ja S. Ulam toisen maailmansodan aikana koodinimenä työlle, jossa simuloitiin neutronien kulkeutumista ydinmateriaalissa
 - eräät simuloinneissa käytettävät ideat ovat peräisin tältä ajalta

11.9.2006

20

Simulointimallit (jatkoa)

- Tietoliikennejärjestelmien (liikenteellisessä) suorituskyvyn simuloinnissa ollaan enimmäkseen tekemisissä
 - diskreetin
 - dynaamisen
 - stokastisensimulointimallin kanssa
- Tämä kurssi käsittelee pääasiassa tällaisia järjestelmiä ja simulointimalleja
- Stokastisuuden huomioonottamisessa simuloinnissa on kaksi vaihtoehtoa
 - luodaan satunnaisuuksia koskeva malli (esim. tietty stokastinen prosessi) ja tuotetaan simuloinnin aikana satunnaisuutta tämän mallin mukaisesti
 - käytetään todellisesta järjestelmästä rekisteröityjä mittausjaksoja lähtötietoina simuloinnissa, trace-driven simulointi

11.9.2006

21

Trace driven simulointi

Etuja

- Uskottavuus
 - todelliset lähtötiedot
 - tarkka kuorma
- Mallin oikeellisuuden tarkastus
 - verrataan mitattuihin arvoihin
- Tarkat vertailut mahdollisia
 - koska kuorma on tarkka (todellinen), voidaan luotettavasti verrata pieniäkin eroja järjestelmässä tai algoritmeissa
- Vähemmän satunnaisuutta
 - trace on deterministinen
 - tuloksiin tulee vaihtelua vain järjestelmämallin muiden osien stokastisuuden ansiosta

Haittoja

- Kompleksisuus
 - vaatii järjestelmän tarkemman simuloinnin
 - vie paljon aikaa
- Edustavuus
 - yhdestä järjestelmästä otettu trace ei ehkä ole edustava toisen järjestelmän kannalta
- Vain yksi vertailukohta
 - järjestelmä, joka optimoidaan yhden tracen perusteella, ei välttämättä ole optimaalinen toisella tracella
- Rajallisuus
 - tracen tallettaminen vie paljon tilaa ja joudutaan rajoittumaan suhteellisen lyhyeen jaksoon
- Kuorman vaihtelu vaikeaa
 - yksi trace edustaa vain yhtä kuormaa
 - herkkyytarkastelut ovat vaikeita tai vaativat traceja eri kuormitusilanteista

11.9.2006

22

Teoria

- Johdanto simulointiin
 - Simulointi menetelmänä
 - Erilaiset simulointimallit
 - Mallinnus, toteutus ja validointi
- Simuloinnin kulku -- prosessin realisaatioiden tuottaminen
- Satunnaismuuttujan arvonta annetusta jakaumasta
- Tulosten keruu ja analyysi
- Varianssinreduktiotekniikoista

11.9.2006

23

Simulointimallin luominen

- Simuloinnin tarkoituksena on toimia korvikkeena todellisella järjestelmällä suoritetuille kokeille
- Mallin pitää olla mahdollisimman *oikea*
 - mallista tehtävien johtopäätösten tulee olla yhtäpitäviä todellisesta järjestelmästä tehtävien johtopäätösten kanssa
- Mallin tulee olla *uskottava*
 - ainoastaan siten oikeankaan mallin tuottamilla tuloksilla on vaikutusta päätöksentekoon
 - pahin vaihtoehto kuitenkin on, että malli on uskottava mutta virheellinen

11.9.2006

24

Periaatteita mallin luomisessa

- Määrittele huolella, mitä halutaan tutkia
 - mitä suureita halutaan mitata
 - miten mallia käytetään
 - vaihtoehtoiset järjestelmäkonfiguraatiot, joita halutaan vertailla
- Valitse tarkoituksenmukainen yksityiskohtaisuuden taso
 - käytä asiantuntijoita herkkyysanalyysien tekoon: mitkä järjestelmän osat ovat tulosten kannalta kriittisimpiä
 - tai käytä “karkeaa” simulointia tai analyttisiä tuloksia herkkyysanalyysiin
 - aloita vain “kohtuulliselta” yksityiskohtaisuuden tasolta, lisää detaljeja tarvittaessa
 - älä sisällytä lopputulosten kannalta turhia yksityiskohtia
 - detaljitason pitää vastata käytettävissä olevien lähtötietojen tasoa

11.9.2006

25

Mallin validointi

- Halutaan varmistua siitä, että malli todella kuvaa tutkittavaa järjestelmää
 - oletukset
 - syöttöparametrien arvot ja jakaumat
 - ulostulosuureet ja johtopäätökset
- Varmistukseen on käytettävissä periaatteessa kolme eri keinoa
 - asiantuntijan intuitio
 - vertailu todellisesta järjestelmästä tehtyihin mittauksiin
 - teoreettiset tulokset

11.9.2006

26

Mallin validointi (jatkoa)

- Asiantuntijan intuitio
 - yleinen ja käytännöllinen menetelmä
 - “aivoriihi”-tilaisuus järjestelmää tuntevien henkilöiden kanssa järkevien oletusten ja alkuarvojen määrittelemiseksi
 - asiantuntija voi usein helposti tunnistaa “mahdottomat” tulokset
- Vertailu todellisesta järjestelmästä tehtyihin mittauksiin
 - paras ja luotettavin menetelmä
 - usein vaikea toteuttaa: järjestelmää ei ole, mittaukset liian kalliita ...
- Teoreettiset tulokset
 - joissakin tapauksissa järjestelmää tai sen osaa kyetään mallintamaan yksinkertaistavien oletusten vallitessa
 - simulointi voidaan ajaa samoilla oletuksilla, ainakin tällöin tulosten tulisi täsmätä
 - ei kuitenkaan vielä takaa, että pitää yhtä todellisuuden kanssa

11.9.2006

27

Simulointiohjelman toteutus

- Simulointiohjelma sisältää yleensä kaikki aikaisemmin mainitut vaiheet mallinnusta ja johtopäätöksiä lukuunottamatta, ts.
 - järjestelmän malliksi valitun stokastisen prosessin realisaatioiden tuottamisen
 - tietojen keruun
 - kerättyjen tietojen tilastollisen analyysin
- Simulointiohjelma voidaan toteuttaa
 - kokonaisuudessaan jollakin yleiskäyttöisellä ohjelmointikielellä, esim. C tai C++,
 - käyttäen hyväksi joitakin simulointia tukevia ohjelmakirjastoja, esim. CNCL
 - erityisillä simulointikielillä (SIMULA, GPSS, MODSIM, SES/workbench, SIMSCRIPT)
 - tietyn järjestelmän simulointiin kehitetyillä työkaluilla
 - Tietoliikennesimulaattoreita: QualNet, OPNET, ns2, OMNeT++

11.9.2006

28

Yleiskäyttöiset ohjelmointikielet vs. simulointikielet

Yleiskäyttöiset kielet

(C, C++, Fortran...)

- Useimmat käyttäjät osaavat jo jotain näistä kielistä
- Yleiskielet löytyvät lähes kaikista koneista
- Koodi on helposti siirrettävää
- Ohjelmat ovat halpoja
- Tuotettu koodi on yleensä nopeampaa
- Joustava, voidaan tehdä "mitä vain"
- Vaatii paljon työtä
- On virhealtista -- vaatii ohjelman huolellisen verifiointin
- Työtä voidaan helpottaa erityiskirjastojen avulla

11.9.2006

Simulointikielitet (yleiskäyttöiset)

(SIMULA, GPSS, MODSIM, SES/workbench, SIMSCRIPT)

- Tarjoavat valmiina useita simulointimallin ohjelmoinnissa tarvittavia ominaisuuksia
- Ohjelmointiaika on pienempi -- projektin kokonaiskustannukset pienemmät
- Sisältävät tietoliikennemallinnusta tukevia piiteitä (valmiita moduleita)
- Voidaan käyttää muidenkin järjestelmien, esim. tuotantojärjestelmien simulointiin
- Ohjelmointi tapahtuu käyttäen kielen omia mallinnusrakenteita

Simulointikielitet (tietoliikennesuuntauneet)

(QualNet, OPNET, OMNET++, ns2)

- Eryteisesti tietoliikennejärjestelmiin tarkoitettu
- Sisältää valmiita verkkojen "rakennuspalikoita" (protokollia

29

Tietoliikennesimulaattorit

- Simulaattorit mahdollistavat tiettyyn luokkaan kuuluvien verkkojen simuloinnin ilman ohjelmointia
- Kiinnostava työkalun tukema verkkotyyppi poimitaan valikosta
- Tyypillisiä elementtejä ovat
 - LAN-tyyppi (Ethernet, token ring...)
 - LAN:ien yhdistämiseen käytetyt elementit (sillat, reitittimet...)
 - LAN:eihin yhdistetyt asemat (PC, työasema...)
 - Myös kuljetuskerroksen protokollat on toteutettu (esim. TCP)
- Nopea toteutus
- Helposti tuotettavat graafiset esitykset
- Sidottu työkalun antamiin raameihin
- Voi helposti johtaa simuloinnin väärinkäyttöön

11.9.2006

30

Ohjelman suunnittelu ja verifiointi

- Käytä top-down suunnittelua
- Modulaarinen ohjelman rakenne
 - jaa ohjelma selviin lohkoihin, jotka kommunikoivat hyvin määriteltyjen rajapintojen kautta
- Olisuuntautunut ohjelmointi
 - käytä geneerisiä olioluokkia (järjestelmän komponentteja)
 - testattujen yleisluokkien perusteella voidaan luoda spesifisempiä luokkia
- Debuggaus trace'ien avulla
- Luetuta ohjelma muilla
- Aja simulointiohjelma eri lähtöparametreilla
 - tarkista kussakin tilanteessa tulosten järkevyyt
- Aja simulointiohjelma yksinkertaisissa tapauksissa
 - joissa vertailutulokset on käytettävissä tai helposti laskettavissa
- Tulosten esittäminen animoituina voi paljastaa virheitä