TAMPEREEN TEKNILLINEN YLIOPISTO

**JUHO VASTAPUU**

# TERÄSRAKENTEISEN HALLIN PITUUSSUUNTAISTEN JÄYKISTERAKENTEIDEN OPTIMOINTI

Diplomityö

TkT Kristo Mela lähetetty tarkastukseen 6.6.2016.

# TIIVISTELMÄ

**JUHO VASTAPUU**: Teräsrakenteisen hallin pituussuuntaisen jäykistysjärjestelmän optimointi
Tampereen teknillinen yliopisto
Diplomityö, x sivua, x liitesivua
Joulukuu 2018
Rakennustekniikan diplomi-insinöörin tutkinto-ohjelma
Pääaine: Rakennesuunnittelu
Tarkastaja: TkT Kristo Mela

Avainsanat:

# ALKUSANAT

# SISÄLLYSLUETTELO

# 1 JOHDANTO

## 1.1 Tutkimuksen tausta

## 1.2 Työn tavoitteet

## 1.3 Työn rajaukset

# 2 OPTIMOINTI

## 2.1 Optimointitehtävän matemaattinen määrittely

Matemaattisella optimoinnilla tarkoitetaan prosessia, jolla löydetään jollekkin funktiolle paras mahdollinen arvo sille asetetut reunaehdot huomioiden. Asettamalla optimoitava kohde sekä halutut rajoite-ehdot matemaattiseen muotoon, voidaan optimoimalla löytää matemaattisin keinoin paras käypä ratkaisu. Käyvällä ratkaisulla tarkoitetaan ratkaisua, joka kuuluu annettujen rajoite-ehtojen joukkoon.

Matemaattisesti optimoinnissa on tavoitteena etsiä funktiolle käyvästä joukosta minimi- tai maksimiarvo. Optimointitekniikoita ja algoritmeja on kehitetty lukuisia ja kukin niistä soveltuu käytettäväksi eri tavalla eri optimointitehtäviin. Optimointi ja erilaisten optimoitimenetelmien tutkiminen on yksi matemaattisen operaatiotutkimuksen osa-alueista. Optimoinnista voidaan joissain yhteyksissä käyttää myös nimitystä matemaattinen ohjelmointi (mathematical programming), jolla viitataan matemaattisten algoritmien kehittämiseen ja ohjelmoimista optimointitarkoituksiin. (Rao 1999, s. 1)

Optimointitehtävä kirjoitetaan matemaattisesti seuraavanlaisessa muodossa.

$$\text{Etsi } \mathbf{x} = \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{Bmatrix} \text{ joka minimoi } f(\mathbf{X})\text{, siten että}$$

$$
\begin{aligned}
g_i(\mathbf{x}) &\le 0, \quad j = 1, 2, \cdots, m \\
h_j(\mathbf{x}) &= 0, \quad j = 1, 2, \cdots, p
\end{aligned}
\tag{1}
$$

missä $\mathbf{x}$ on vektori, joka sisältää n-kappaletta suunnittelumuuttujia, f($\mathbf{x}$) on tavoitefunktio, $g_i(\mathbf{x})$ ja $h_j(\mathbf{x})$ ovat rajoite-ehtoja. Rajoite-ehdot voivat olla yhtälön mukaisesti joko epäyhtälö- tai yhtälömuotoisesti ilmoitettuja. Suunnittelumuuttujien lukumäärä (n) sekä rajoite-ehtojen lukumäärä (m ja/tai p) eivät ole riippuvaisia toisistaan. Tällaista optimointitehtävää kutsutaan rajoitetuksi optimointiongelmaksi. Optimointiongelman ei kuitenkaan tarvitse olla rajoitettu, vaan se voidaan ilmoittaa myös rajoittamattomana. Kaavassa (1) on esitetty optimointitehtävän standardimuotoinen asettelu (standard design optimization model). (Rao 1999, s. 6)

Vektori $\mathbf{x}$ sisältää optimointitehtävän kaikki suunnittelumuuttujat (design variables). Muuttamalla jonkin suunnittelumuuttujan $x_i$ arvoa, muuttuu myös tavoitefunktion f($\mathbf{x}$) arvo. Suunnittelumuuttujista voidaan käyttää myös nimitystä optimointimuuttujat tai vapaat muuttujat, eli niiden arvoja voidaan muutella vapaasti kun haetaan tavoitefunktiolle arvoa. Toisistaan riippumattomien eli itsenäisten suunnittelumuuttujien lukumäärä on optimointiongelman vapausasteluku (design degree of freedom). Yleisesti ottaen suunnittelumuuttujien tulee olla toisistaan riippumattomia, mutta joissain tapauksissa niiden määrä voi olla ongelman vapausastelukua suurempi. Tämä on perusteltua esimerkiksi silloin, kun kohdefunktion määrittely pelkillä itsenäisillä suunnittelumuuttujilla olisi hankalaa. Jokaiselle suunnittelumuuttujalle täytyy myös pystyä asettamaan jokin numeerinen lähtöarvo, jotta optimointitehtävä pystytään suorittamaan.

Kohde- tai tavoitefunktio f($\mathbf{x}$) (objective function) on optimointitehtävän matemaattinen muoto ilmoitettuna suunnittelumuuttujavektorin $\mathbf{x}$ funktiona. Optimointitehtävän tavoitteena on joko minimoida tai maksimoida kohdefunktion arvo. Mikäli optimointitehtävässä on useampi kuin yksi kohdefunktio, käytetään tehtävästä nimitystä monitavoiteoptimointi (multiobjective design optimization). Tällöin kohdefunktio ilmaistaan matemaattisesti kohdefunktioiden joukkona

$$
\mathbf{f}(\mathbf{x}) = \begin{bmatrix} f_1(\mathbf{x}) & f_2(\mathbf{x}) & \cdots & f_p(\mathbf{x}) \end{bmatrix}, \tag{2}
$$

jossa jokainen kohdefunktio koostuu kuitenkin samasta suunnittelumuuttujavektorista $\mathbf{x}$.

Optimoitavalle kohteelle asetettavat rajoite-ehdot esitetään rajoitefunktioina $g_i(\mathbf{x})$ ja $h_j(\mathbf{x})$. Optimointialgoritmi ratkaisee optimointitehtävän siten, että kohdefunktion arvo toteuttaa rajoite-ehdot. Rajoite-ehtojen muodostamaa joukkoa kutsutaan täten optimointiongelman käyväksi joukoksi (feasible region). Mitä tahansa käyvässä joukossa olevaa kohdefunktion arvoa kutsutaan käyväksi ratkaisuksi (feasible design) huolimatta siitä onko kyseessä optimiratkaisu. Kaksiuloitteisessa tapauksessa käypää joukkoa voidaan havainnollistaa piirtämällä rajoitefunktioiden käyrät koordinaatistoon. Käypä joukko muodostuu näiden käyrien rajoittamana alueena. Käyvän joukon negaatiota kutsutaan ei-käyväksi joukoksi (unfeasible region). Sekä tavoitefunktion f($\mathbf{x}$), että rajoite-ehtojen $g_i(\mathbf{x})$ ja $h_j(\mathbf{x})$ on oltava toisitaan joko implisiittisesti tai eksplisiittisesti riippuvia. Mikäli riippuvuutta funktioiden välillä ei ole, ei optimointitehtävää voi muodostaa eikä varsinaista optimointiongelmaa voi edes osoittaa. (Arora 2004, s. 43)

Kuten optimointitehtävän määrittelevä kaava (1) osoittaa, optimointiongelmalle voidaan asettaa rajoite-ehtoja sekä yhtälö- että epäyhtälömuodossa. Epäyhtälörajoitteita kutsutaan käypään joukkoon nähden toispuoleisiksi rajoite-ehdoiksi (unilateral constraints tai one-sided constraints). Epäyhtälörajoitteiden rajoittama käypä joukko on täten paljon laajempi kun verrataan yhtälörajoitteista käypää joukkoa. Esimerkiksi kakisuloitteisessa tapauksessa yhtälörajoite tarkoittaisi, että käypä ratkaisu löytyisi rajoitefunktion käyrältä. (Arora 2004, s. 16–18)

Yhtälömuotoisten rajoite-ehtojen määrän tulee olla maksimissaan suunnittelumuuttujien määrä, toisin sanottuna optimointitehtävän kaavan (1) tulee toteuttaa ehto

$$p \leq n. \tag{3}$$

Tapaus, jossa yhtälömuotoisia rajoite-ehtoja on annettuja suunnittelumuuttujia enemmän, on kyseessä ylimääritetty yhtälöryhmä (overdetermined system). Tällaisessa tapauksessa rajoite-ehtojen joukossa on ylimääräisiä eli redundatteja ehtoja, jotka toteuttavat suoraan jonkun muun rajoite-ehdon, eikä niiden ilmoittaminen täten ole tarpeellista. Triviaalitapauksessa jossa suunnittelumuuttujien määrä ja yhtälömuotoisten rajoite-ehtojen määrä on yhtäsuuri, löytyy tehtävälle ratkaisu analyyttisin keinoin eikä optimointi ole tarpeellista. Kaksiuloitteisessa tapauksessa tämä tarkoittaisi kahden käyrän leikkauspistettä.

Standardimuotoisessa optimointitehtävässä epäyhtälörajoitteet ilmoitetaan aina kaavan (1) osoittamassa muodossa, eli siten että rajoite-ehto on pienempi tai yhtäsuuri kuin nolla ($\leq 0$). Tästä huolimatta voidaan optimointitehtävässä käsitellä myös $\geq$ -tyyppisiä rajoite-ehtoja. Standardimuotoista tehtävää aseteltaessa nämä voidaan muuttaa $\leq$ -muotoon yksinkertaisesti kertomalla ehto luvulla -1. Epäyhtälömuotoisten rajoite-ehtojen määrää ei ole rajoitettu, toisin kuin yhtälömuotoiset rajoitteet kaavassa (3). Niiden määrää ei ole siis rajoitettu. (Arora 2004, s. 43)

## 2.2 Optimointitehtävän luokittelu

Rajoitefunktioiden perusteella tehtävä voidaan luokitella joko rajoitetuksi- tai ei-rajoitetuksi tehtäväksi. Mikäli tehtävällä on yksikin rajoitefunktio, on kyseessä rajoitettu optimointitehtävä. Luokittelua voidaan tarkentaa edelleen osittain rajoitetuksi tai täysin suljetuksi systeemiksi. Suljetulla systeemillä tarkoitetaan tilannetta, jossa rajoitefunktiot muodostavat äärellisen kokoisen käyvän joukon.

Suunnittelumuuttujien perusteella tehtävä voidaan jakaa staattiseksi tai dynaamiseksi tehtäväksi. Staattisessa tai parametrisessa tehtävässä kohdefunktio on määritelty suunnittelumuuttujien suhteessa ja tehtävänä on ratkaista suunnittelumuuttujat. Kaavan (1) määrittelyssä kyseessä on staattinen optimointitehtävä, jossa siis etsitään suunnittelumuuttujille arvo. Dynaamisessa tehtävässä kohdefunktio muodostuu puolestaan funktioista, jotka on määritelty jonkun tietyn parametrin suhteen, kuten esimerkiksi seuraavasti.

$$\text{Etsi } \mathbf{x}(t) = \begin{Bmatrix} b(t) \\ d(t) \end{Bmatrix} \text{ joka minimoi } f[\mathbf{X}(t)]. \tag{4}$$

Dynaamisessa tehtävässä etsitään siis kohdefunktioon sijoitettavien suunnittelumuuttujien sijasta funktiot, jotka esitetään jonkin parametrin suhteen.

## 2.3 Rakenteiden optimointi

# VIITTEET

Arora, J. S. (2004). Introduction to optimum design. 2. painos. ISBN: 0-12-064155-0.

Rao, S. S. (1999). Engineering optimization : theory and practice. 4. painos. ISBN: 978-0-470-18352-6.