

## AVARUUSRISTIKON GEOMETRIAN OPTIMOINNISTA

Sami Holopainen

Rakenteiden mekaniikka, Vol. 37  
No. 2, 2004, ss. 34 - 46

### TIIVISTELMÄ

Artikkelissa tarkastellaan usein kirjallisuudessa esiintyvän avaruusristikon yhdistettyä poikkileikkauspinta-alojen ja geometrian optimointia. Tavoitteena on siis hakea laskennallisesti ristikon sauvojen optimikoko ja niiden solmuille optimaalinen sijainti. Kohdefunktiona on käytetty ristikon materiaalityyppiä (massaa) siten, että sallittuja jännityksiä ei saa ylittää. Optimointimenetelmä on selostettu yleisellä tasolla.

### JOHDANTO

Rakenteen geometrian optimointi on eräs kantavien rakenteiden optimoinnin ongelmatyyppi [5]. Siitä käytetään myös nimitystä *layout*-optimointi, mikä käsittää myös topologian ja sauvojen poikkipintasuureiden optimoinnin (*sizing*). Kyseessä on sidottu optimointi, sillä tässä tapauksessa sauvojen jännitykset ovat rajoitettuja. Optimointiongelma on epälineaarinen (*nonlinear programming* NLP), koska siinä olevat funktiot ovat suunnittelumuuttujien suhteen epälineaarisia. Suunnittelumuuttujina olevat solmukoordinaatit ja poikkileikkauspinta-alat ovat tässä tapauksessa jatkuvia. Joskus esim. sauvan poikkipintasuureet ovat olosuhteiden pakosta (saatavuus) kuitenkin diskreettejä. Kuvatut optimointiongelmat ratkaistaan jollakin robustilla optimointimenetelmällä, jonka sopivuus ko. ongelman ratkaisuun perustuu usein hyviin käytännön kokemuksiin. Menetelmän tehokkuus perustuu ennen kaikkea siihen, että kriteeri(e)n ja rajoitusten herkkyydet tunnetaan. Ko. ongelma lineaarisena statiikan tehtävänä on sellainen, että herkkyydet voidaan laskea, jopa analyttisesti. Kuitenkin juuri herkkyydlaskenta on usein ratkaisun haastavin osa, koska se on suunnittelumuuttujina olevien solmukoordinaattien suhteen hankalaa. Ongelman ratkaisu voidaan jakaa kolmeen osaan: Statiikan ratkaisu (FEM), herkkyydlaskenta kuormitetulle rakenteelle (*design sensitivity analysis*, DSA) ja varsinainen optimointi jollakin robustilla optimointimenetelmällä. Menetelmä voi olla kokemukseen ('insinöörijatteluun') perustuva epäsuora, iteratiivinen menetelmä, esim. täysi jännitystekniikka (*fully stressed design*, FSD) tai optimaalisuuskriteerien menetelmä (*optimality criteria*, OC). Menetelmänä voi olla myös matemaattinen optimointialgoritmi, kuten peräkkäisten kvadraattisten/lineaaristen approksimaatioiden menetelmä (*sequential quadratic/linear programming*, SQP/SLP), täydennetyn Lagrangen funktion menetelmä (*augmented lagrangian method* ALM), MMA (*method of moving asymptotes*) tai kolmanneksi jokin heuristinen menetelmä, esim. simuloitu jäähdytys

(*simulated annealing*, SA), tabu-haku (*tabu search*, TS) ja geneettiset algoritmit. Ensimmäiset menetelmät ovat tyypillisesti hyvin tehtäväsidoonaisia, mutta suppenevat melko hyvin. Matemaattiset algoritmit suppenevat usein hyvin ja ovat suhteellisen yleispäteviä. Jälkimmäiset menetelmät suppenevat gradienttipohjaisia matemaattisia menetelmiä hitaammin ja vaativat paljon analyysyjä. Niitä kuitenkin käytetään, kun luotettavaa gradientti-informaatiota ei ole saatavilla esim. stabiilisuuteen liittyvissä rajoitusehdoissa. Tässä artikkelissa optimointimenetelmänä on nk. täydennetyt *Lagrangen* funktion menetelmä (ALM), jota on selostettu yleispiirteisesti.

## OPTIMOINTIONGELMAN RATKAISUSTA

ALM-menetelmä on nk. sakkofunktiomenetelmä [2], [9]. Perusajatuksena on muodostaa jono vapaita ääriarvotehtäviä, joiden ratkaisut konvergoivat kohti sidotun tehtävän ratkaisua. Konvergenssi saavutetaan sakottamalla käyvän alueen (pisteet, jotka eivät riko rajoitusehtoja) ulkopuolella kulkemisesta. Sakkofunktiona käytetään tässä neliöllistä funktiota (jatkuva, positiivinen). Optimiratkaisu  $\mathbf{x}^*$  voidaan määrittellä s.e.  $\mathbf{x}^* = \lim_{r \rightarrow \infty} \mathbf{x}(r)$  eli sakkokerrointa  $r$  kasvatetaan iteraatioittain. Sopiva sakkokerroin valitaan s.e. saatu ratkaisu olisi lähellä alkuperäisen ongelman käypää aluetta ( $r$  riittävän suuri) ja toisaalta niin, että täydennetty *Lagrangen* funktio ei käyttäydy numeerisesti huonosti ( $r$  riittävän pieni). Käytetty ALM-algoritmi ratkaisee periaatteessa ongelmaa [9]:

$$\begin{aligned} & \min f(\mathbf{x}) \\ & \text{s.e. } \mathbf{h}(\mathbf{x}) = \mathbf{0}, \\ & \mathbf{x}^l \leq \mathbf{x} \leq \mathbf{x}^u. \end{aligned} \tag{1}$$

Probleeman täydennetty *Lagrangen* funktio

$$A(\mathbf{x}, \boldsymbol{\lambda}, r) = f(\mathbf{x}) + \boldsymbol{\lambda}^T \mathbf{h}(\mathbf{x}) + \frac{r}{2} \mathbf{h}(\mathbf{x})^T \mathbf{h}(\mathbf{x}), \quad \mathbf{x} \in \mathbb{R}^n; \boldsymbol{\lambda}, \mathbf{h} \in \mathbb{R}^m; r > 0.$$

ALM:ssä alkuperäinen sidottu ongelma korvataan vapaalla tehtävällä:  $\min_{\mathbf{x}, \boldsymbol{\lambda}} A(\mathbf{x}, \boldsymbol{\lambda}, r)$ .

Minimipisteessä  $\nabla A(\mathbf{x}, \boldsymbol{\lambda}, r) = \mathbf{0}$ , josta seuraa, että

$$\begin{cases} \nabla f(\mathbf{x}) + \mathbf{h}'(\mathbf{x})^T \boldsymbol{\lambda} = \mathbf{0}, \\ \mathbf{h}(\mathbf{x}) = \mathbf{0}. \end{cases} \text{ missä} \tag{2}$$

$$\nabla f(\mathbf{x}) = f'(\mathbf{x})^T, \quad [\nabla f(\mathbf{x})]_i = \frac{\partial f}{\partial x_i}, \quad \mathbf{h}'(\mathbf{x})^T = \nabla \mathbf{h}(\mathbf{x}), \quad [\nabla \mathbf{h}(\mathbf{x})]_{ij} = \frac{\partial h_i}{\partial x_j}$$

Edellä  $f: \mathbb{R}^{q_1} \rightarrow \mathbb{R}$  on kohdefunktio (objekti),  $\boldsymbol{\lambda}$  *Lagrangen* kerroinvektori ja  $\mathbf{h}$ :n komponentteina ovat rajoitusyhtälöt  $h_i: \mathbb{R}^{q_1} \rightarrow \mathbb{R}$ . Sakkofunktiona on termi  $\mathbf{h}^T \mathbf{h}$ . Käytetyn täydennetyt *Lagrangen* funktion minimipiste on alkuperäisen ongelman *Karush-Kuhn-Tucker*-piste (KKT-piste). Ts. ehto (2) toteutuu.

Käytännössä algoritmi etenee iteratiivisesti:

1. Valitse alkuarvot  $\boldsymbol{\lambda}$  ja sakkokerroin  $r$
2. Ratkaise minimointitehtävä  $\min A_{\mathbf{x},\boldsymbol{\lambda}}(\mathbf{x},\boldsymbol{\lambda},r)$ , ratkaisuna saadaan iteraatti  $\mathbf{x}^k$
3. Jos  $\mathbf{h}(\mathbf{x})=\mathbf{0}$  (sakkofunktio riittävän pieni), lopeta. Muuten päivitä  $\mathbf{x}^k, \boldsymbol{\lambda}^k$ , valitse  $r^{k+1} > r^k$  ja jatka askeleesta 2.

Alkuperäinen ongelma on siis yhtälörajoitteinen, jonka KKT-piste saadaan Lagrangen funktion (täydennetty Lagrangen funktio ilman sakkotermiä) gradientin nollakohdasta. Tällöin rajoitutaan tarkastelemaan vain käypiä pisteitä, ts.  $A(\mathbf{x},\boldsymbol{\lambda}) = f(\mathbf{x})$ . Käytetyissä algoritmissa ratkaisua haetaan iteratiivisesti. Jokaisella iteraatiokierröksellä  $k$  ratkaistaan lineaarisesti rajoitettua problemaa, jossa kohdefunktiona on täydennetty Lagrangen funktio (kohta 2):

$$\begin{aligned} \min A(\mathbf{x}, \lambda, r) &= f(\mathbf{x}) + \lambda^T \mathbf{h}(\mathbf{x}) + \frac{r}{2} \mathbf{h}(\mathbf{x})^T \mathbf{h}(\mathbf{x}) \\ \text{s.e. } \mathbf{h}'(\mathbf{x}_k)(\mathbf{x} - \mathbf{x}_k) + \mathbf{h}(\mathbf{x}_k) &= \mathbf{0}, \\ \mathbf{x}^l \leq \mathbf{x} \leq \mathbf{x}^u \end{aligned} \quad (3)$$

Ratkaisu on approksimatiivinen, jossa kohdefunktiota  $f$  approksimoidaan toisen kertaluvun kvadraattisella approksimaatiolla ja Hessen matriisia  $A''$  päivitetään BFGS-tekniikalla (kirjaimet tulevat kehittäjien Broyden, Fletcher, Goldfarb ja Shanno nimestä). Ratkaisulta ei vaadita suurta tarkkuutta, sillä approksimatiivista problemaa ratkaistaan kunnes ongelma (3) toteutuu (ulompi iteraatio) ja optimi saavutetaan. Ts. alkuperäisen ongelman KKT-ehdot toteutuvat. Merkitään  $\mathbf{I}^k = \boldsymbol{\lambda}^{k+1} - \boldsymbol{\lambda}^k$  Uusi hakusuunta  $\mathbf{s}^k$  saadaan Newtonin menetelmällä seuraavasti:

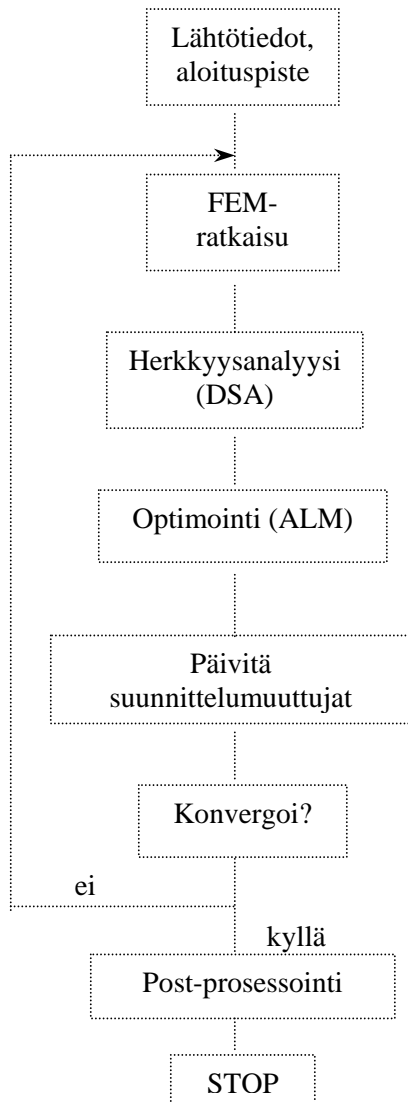
$$A''(\mathbf{z}^k)\mathbf{s}^k = -\nabla A(\mathbf{z}^k). \quad (4)$$

Edellä  $\mathbf{z}^k = [\mathbf{x}^k, \boldsymbol{\lambda}^k]^T$  on nykyinen iteraatiopiste ja  $\mathbf{s}^k = [\mathbf{d}^k, \mathbf{I}^k]^T$  hakusuunta. Hessen matriisin komponentit ovat seuraavat:

$$[A'']_{ij} = \frac{\partial^2 A}{\partial x_i \partial x_j}.$$

Uusi iteraatiopiste  $\mathbf{z}^{k+1} = \mathbf{z}^k + \mathbf{s}^k$  ja  $\mathbf{x}^{k+1} = \mathbf{x}^k + \mathbf{d}^k$ . Kuinka ryhmä (4) ratkaistaan lopulta, selviää esim. artikkelista [4]. ALM:ää on käsitelty myös lähteessä [2]. Mukana voi olla myös epäyhtälörajoituksia, kun niihin lisätään pelivara- ja ylijäämämuuttujat s.e. ongelma palautuu yhtälörajoitteiseksi tehtäväksi. Myös muuttujille voidaan asettaa rajat, joiden käsittely vaihtelee algoritmeittain. Algoritmin idea on siis se, että käyvän alueen ulkopuolella ”liikkuminen” on ”kallista” ja aiheuttaa siis ”kustannuksia”. Hyvin asetetut optimointiparametrit (mm. lopetus, sakkokerroin iteraatioittain jne.) saavat aikaan sen, että optimi on käyvällä alueella. Algoritmi konvergoi, jos KKT-ehdot (2) toteutuvat. Lopetuskriteereinä on tavallisesti kohdefunktion ja suunnittelumuuttujien muutokset. Ei-konveksissa ongelmassa (joko kohdefunktio tai rajoitukset ei-konvekseja) ei ole takeita siitä, päädytäänkö globaaliin vai kenties johonkin lokaaliin optimiin (mikä sekun

on jo parannus sinänsä). Optimointialgoritmi saa tarvittavan siirtymä- ja jännitys-informaation FEM-ohjelmasta. Näitä tietoja tarvitaan kohdefunktion ja rajoitusten arvojen laskemiseen laskentapisteessä sekä niiden herkkyyksien laskennassa. Ratkaisuun käytetyn ohjelman vuokaavio on periaatetasolla esitetty kuvassa 1.



**Kuva 1 Ratkaisuvuokaavio.**

## HERKKYYSLASKENTA

Optimiratkaisu toteuttaa KKT-ehdot (2). Niiden laskennassa tarvitaan kohdefunktion ja rajoitusten herkkyydet suunnittelumuuttujien suhteen. Herkkyyksilaskenta tai -analyysi (*Design Sensitivity*

*Analysis DSA*) on analyyttisesti usein hyvin hankalaa, mahdotontakin, jolloin on tyydyttävä laskemaan funktion  $g:R^q \rightarrow R$  muutosta, kun sitä hieman häiritään muuttamalla suunnittelumuuttujien ( $x_i \in R$ ) arvoja esim. differenssimenetelmällä tai kokeilemalla. Matemaattisessa mielessä DSA ja häiriöherkkyystarkastelu (Imperfection Sensitivity Analysis ISA) ovat ekvivalentit. Tämä artikkeli käsittelee staattisesti kuormitettujen ristikoiden optimointia. Tässä tapauksessa myös analyyttinen herkkyyksilaskenta onnistuu. Herkkyyksianalyysiä on käsitelty melko kattavasti aiheeseen liittyen mm. lähteessä [7]. Varsinainen analyysi, voidaan tehdä periaatteessa kahdella eri tavalla:

- Varioidaan kontinuumimallin tasapainoehto suunnittelumuuttujien suhteen ja ratkaistaan saatu lineaarinen ryhmä e. NR-iteroinnilla tai diskretoimalla ensin FEM-muotoon.
- Muodostamalla suunnitteluderivaatat diskreetille (FEM) mallille.

Ensimmäisen menetelmän etu on siinä, että mahdollisesti rajoitusehtoina olevien jännitysten (ja venymien) herkkyyksilaskenta sisältyy variointiin, sillä varioitavat funktio(naali)t sisältävät ne. Kontinuumiperusteista herkkyyksianalyysiä käytetään usein epälineaarissa mekaniikan ongelmissa. Jälkimmäinen menetelmä on luonnollisesti suoravii-vaisempi esittää, sillä se on jo valmiiksi ratkaistavassa muodossa ja tässä artikkelissa siksi käytetty.

Herkkyysanalyysi voidaan suorittaa suoraan derivoimalla tai nk. adjungoidulla menetelmällä seuraavasti:

$$\frac{dg}{dx_i} = \frac{\partial g}{\partial x_i} + \frac{\partial g}{\partial \mathbf{q}} \frac{\partial \mathbf{q}}{\partial x_i} = \frac{\partial g}{\partial x_i} + \mathbf{z}^T \frac{\partial \mathbf{q}}{\partial x_i}, \quad i = 1, \dots, n, \quad \text{missä} \quad (5)$$

$$\mathbf{z} = \frac{\partial g}{\partial \mathbf{q}} \text{ ja } \mathbf{q} \in \mathbb{R}^p \text{ (solmuisiirtymävektori).}$$

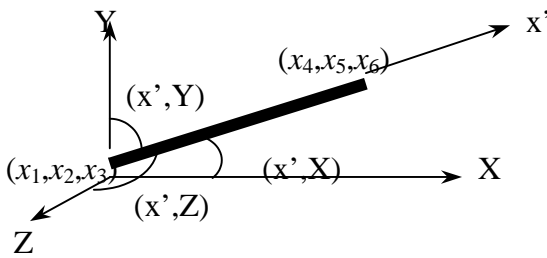
Tässä artikkelissa objektina on ristikon materiaalitulavuus (kuormittamattomana) ja rajoitusehtoina ovat sauvojen jännitykset. Suunnittelumuuttujina ovat sauvojen pinta-alat ja niiden solmukoordinaatit (ennen kuormittamista alkutilassa). Rajoitusehtojen herkkyudet

$$\frac{d\sigma_e}{dA_e} = \frac{\sigma_e}{|\sigma_e|} \mathbf{z}^T \frac{\partial \mathbf{q}}{\partial A_e}. \quad (6)$$

Oletetaan, että muodonmuutokset ovat pieniä ja materiaali on lineaarisesti kimmoista (vakio kimmomoduuli  $E$ ), jolloin todelliselle jännitykselle pätee

$$\mathbf{z} = E\mathbf{b} \text{ eli } \mathbf{z} = E/L_e [-l, -m, -n, l, m, n], \quad \text{missä} \quad (7)$$

$L_e$  = elementin  $e$  alkuperäinen pituus,  
 $l = \cos(x', X)$ ,  
 $m = \cos(x', Y)$ ,  
 $n = \cos(x', Z)$ .



**Kuva 2 Suuntakosinit.**

Ts.  $\mathbf{z}$ :n muut komponentit ovat nollia. Tasapainotilassa potentiaalienergia  $\Pi$  saavuttaa miniminsä eli  $\partial\Pi/\partial\mathbf{q} = 0$ . Derivoimalla tasapainoehto saadaan  $\mathbf{K}\partial\mathbf{q}/\partial A_e = -\partial\mathbf{K}/\partial A_e \mathbf{q}$ , kun kuormitus ei riipu suunnittelumuuttujista (eikä siirtymistä). Edellä  $\mathbf{K} = \partial\mathbf{R}/\partial\mathbf{q}$  eli elementin jäykkyysmatriisi ja  $\mathbf{R}$  = sisäisten voimien vektori.

Merkitään  $\mathbf{K}\boldsymbol{\lambda} = \mathbf{z}$ . Silloin

$$\begin{aligned}\frac{d\sigma_e}{dA_e} &= \frac{\sigma_e}{|\sigma_e|} \lambda^T \mathbf{K} \frac{\partial \mathbf{q}}{\partial A_e} = -\frac{\sigma_e}{|\sigma_e|} \lambda^T \frac{\partial \mathbf{K}}{\partial A_e} \mathbf{q} \quad \text{nyt ristikolle} \\ \frac{d\sigma_e}{dA_e} &= -\frac{\sigma_e}{|\sigma_e|} \lambda_e^T \frac{1}{A_e} \mathbf{K}_e \mathbf{q}_e, \text{ missä } (\bullet)_e \text{ liittyy elementtiin } e.\end{aligned}\quad (8)$$

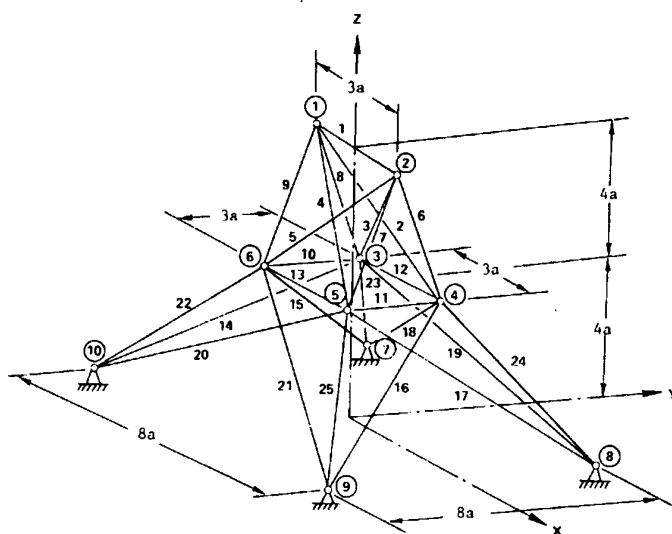
Kohdefunktion  $f$  herkkyys sauvan pinta-alan  $A_e$  suhteen eli  $df/dA_e = L_e$  (sauvaelementin  $e$  pituus). Herkkyuden laskenta solmukoordinaattien suhteen tapahtuu periaatteessa samalla tavalla, mutta suora derivointi  $\partial \mathbf{K}/\partial x_j$  on hankalampaa: vain ko. solmuun liittyvien elementtien jäykkyysmatriisit riippuvat ko. muuttujasta. Lisäksi sauvan jännitys riippuu myös eksplisiittisesti solmukoordinaateistaan (kaavan (5) termi  $\partial g/\partial x_j \neq 0$ ). Kohdefunktiossa derivointi kohdistuu ko. solmuun liittyvien elementtien  $k$  pituuksiin:

$$\frac{df}{dx_j} = \sum_k A_k \frac{\partial L_k}{\partial x_j}, \quad k \in I \text{ (ko. solmuun liittyvät elementit)} \quad (9)$$

## AVARUUSRISTIKON LAYOUT-OPTIMOINTI

Tavoitteena on siis hakea laskennallisesti ristikon sauvojen optimikoko ja niiden solmuille optimaalinen sijainti. Optimointikriteerinä on käytetty ristikon materiaalilavuutta (massaa) siten, että sallittuja jännityksiä ei saa ylittää. Muita rajoituksia olisivat esim. ristikon jousto (myös kohdefunktiona), siirtymät, globaali ja lokaalit nurjahdusehdot. Näistä tavallisesti helpoimmin käsiteltäviä (numeeriset ongelmat, herkkyyslaskenta) ovat siirtymät ja jousto. Suunnittelumuuttujina ovat ristikon tiettyjen solmujen koordinaatit ja sauvojen pinta-alat ja ne ovat jatkuvia. Kyseessä on siis epälineaarinen optimointiongelma (*Nonlinear programming*, NLP) ja sen instanssina avaruusristikon *layout*-optimointi. Tässä esimerkissä pinta-alalle on asetettu alaraja, jolla vältetään nk. ”singulaarisuusongelma”, joka usein esiintyy suunnittelumuuttujien nollakohdissa käytettäessä jännitysrajoituksia (ongelma KKT-ehtoihin perustuvissa algoritmeissa). ”Singulaarisuusongelma” voidaan ottaa huomioon nk. relaksointitekniikalla (Cheng & Guo 1997) tai olla luottamatta gradientti-perusteiseen tekniikkaan sellaisenaan soveltamalla esim. primaali-duaalimenetelmää (Achtzinger 2000). Yhdistetyssä *sizing* ja *layout*-optimoinnissa alarajan asettaminen on osoittautunut suurissa ongelmissa (*large-scale*) lähes välttämättömäksi, ellei optimointia suoriteta monitasoisesti (esim. erikseen *sizing* ja *layout*-optimointi peräkkäisinä vaiheina) tai käyttämällä heuristisia optimointimenetelmiä [3]. Tyypillisesti avaruusristikoissa on suuri määrä osia, joista jopa kaikki on optimoitava. Tällöin optimointialgoritmin on myös kyettävä tunnistamaan kussakin iteraatiopisteessä aktiiviset rajoitusehdot ts. tässä tapauksessa niiden sauvojen jännitykset, jotka ovat lähellä sallittua jännitystä. Muut rajoitusehdot voidaan jättää tarkastelun ulkopuolelle

ko. vaiheessa. Tarkasteltava rakenne on symmetrinen ja käsittää 25 sauvaelementtiä (kuva 3) [1], [2].



**Kuva 3 Optimoitava ristikko**

### OPTIMOINNIN LÄHTÖTIEDOT

Ajoa varten annetut alkutiedot on esitetty taulukossa 1:

**Taulukko 1. Lähtötiedot.**

Materiaali	teräs
Kimmomoduuli	$2e5$ MPa
Minimiala	$150$ mm <sup>2</sup>
Mitta $a$	$1000$ mm

Tässä optimoidaan sauvojen 10,11,...,25 poikkileikkausaloja ja vaaditaan, että sauvojen 1,2,...,9 pinta-alat ovat  $150$  mm<sup>2</sup>. Lisäksi vaaditaan, että vaakasuuntaiset sauvat 10, 11, 12 ja 13 ovat yhtäsuuret eli ongelmassa on yhteensä 13 pinta-alamuuttujaa. Alussa kaikkien pinta-alamuuttujien arvoiksi asetetaan  $200$  mm<sup>2</sup>. Geometriaa optimoidaan s.e. suunnittelu-, tarkemmin paikkamuuttujina ovat solmujen 1 ja 2 kaikki koordinaatit (yht. 6 kpl). Paikkamuuttujien arvot alussa on esitetty taulukossa 2:

**Taulukko 2. Paikkamuuttujat alussa.**

	X	Y	Z
<b>Solmu 1</b>	-1500	0	8000
<b>Solmu 2</b>	1500	0	8000

Optimointiongelma on siten seuraava:

$$\min f = V(\mathbf{A}, \mathbf{x}) = A_i L_i(\mathbf{x}), \quad i = 1, 2, \dots, 25, \quad (10)$$

$$\sigma_{sall\_a} \leq \sigma_i(\mathbf{x}_j, \mathbf{u}_k(\mathbf{A}, \mathbf{x})) \leq \sigma_{sall\_y}, \quad i = 1, 2, \dots, 25, j \in S \text{ (ko. sauvan solmukoordinaatit)}, \quad (11)$$

$$k = 1, 2, \dots, 6, \text{ ts. sauvan vapausasteet (DOF).}$$

$$A_{sall\_a} \leq A_m \leq A_{sall\_y}, \quad m = 1..13, x_{sall\_a,n} \leq x_n \leq x_{sall\_y,n}, \quad n = 1, 2, \dots, 6. \quad (12)$$

Kaikkien sauvojen sallitut jännitykset ovat 130 MPa vetoa ja 13 MPa puristusta. Suunnittelumuuttujina ovat siis sauvojen 10, 14, 15, ..., 25 poikkileikkausalat ja solmujen 1 ja 2 koordinaatit. Sauva ei saa hävitä, joten solmujen ei sallita mennä päällekkäin. Edellä sallitut paikkamuuttujien arvot on esitetty taulukossa 3 (mm) ja kuormitus (symmetrinen) taulukossa 4:

**Taulukko 3. Paikkamuuttujien rajat (mm).**

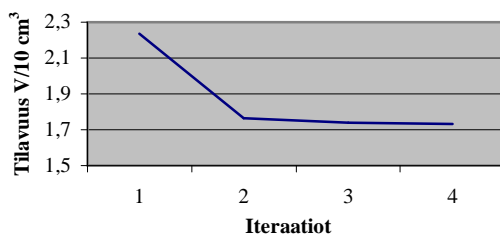
	$x_{sall\_a}$	$x_{sall\_y}$	$y_{sall\_a}$	$y_{sall\_y}$	$z_{sall\_a}$	$z_{sall\_y}$
<b>Solmu 1</b>	-4000	<0	-1000	1000	5000	8000
<b>Solmu 2</b>	>0	4000	-1000	1000	5000	8000

**Taulukko 4. Kuormitus (kN).**

	<b>X</b>	<b>Y</b>	<b>Z</b>
<b>Solmu 1</b>	0	0	-5
<b>Solmu 2</b>	0	0	-5

## RATKAISU

Tilavuus iteraatioittain\* on esitetty kuvassa 4:



**Kuva 4. Tilavuus iteraatioittain.**

\* Vain nk. pääiteraatiot (uloin laskentasilmutka), jotka tarvitaan, että kohdefunktio optimissa ei enää muutu sallittua enemmän edellisen laskentakierroksen optimista. Kohdefunktion sallittu muutos on  $1 \cdot 10^{-3}$ . Uusi kierros alkaa edellisen kierroksen optimista ("lämmen käynnistys").



Optimirakenne on siis symmetrinen, ja solmut 1 ja 2 yhtyvät! Suunnittelumuuttujat optimissa:

$$A_i^* = 150 \text{ mm}^2 \text{ (alarajalla), } i = 1..13,$$

$$\mathbf{x}_1^* = [ >0, 0, 8000 ]^T \text{ (solmu 1) ja}$$

$$\mathbf{x}_2^* = [ >0, 0, 8000 ]^T \text{ (solmu 2).}$$

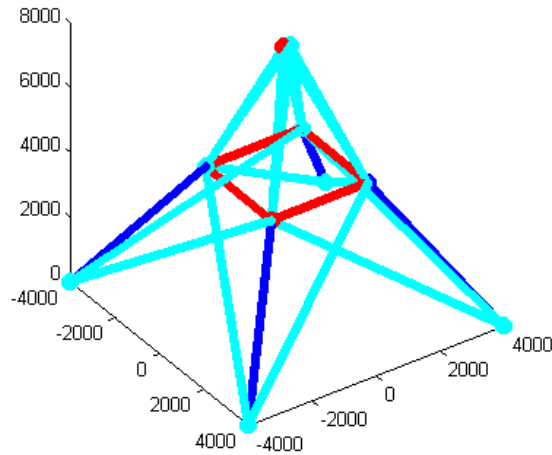
Tehtävä oli asetettu siten, että solmut eivät saa mennä päällekkäin. Numeroarvot jännitysrajoitusten, Lagrangen kertoimien ja solmuniirymien osalta on esitetty taulukossa 5.

**Taulukko 5. Numerodata optimissa.**

Sauvan nro/DOF <sup>*)</sup>	Jännitykset sauvoissa (MPa)	Lagrangen kertoimet $\lambda^*/1000$	Solmuniirymät (mm)
1	0.5283	-0.0000	-0.0003
2	-9.3362	0.0074	0.0134
3	-9.3678	0.0008	-0.6899
4	-9.2176	0.0008	0.0003
5	-9.1975	0.0010	0.0134
6	-9.6999	0.1852	-0.6903
7	-9.5230	2.4214	-0.0216
8	-9.6605	0.3554	0.0206
9	-9.5434	2.2952	-0.4365
10	2.8703	0.0000	0.0216
11	2.8701	0.0000	0.0205
12	2.8802	0.0000	-0.4365
13	2.8445	0.0000	0.0214
14	-6.8368	-0.0000	-0.0225
15	-6.2619	0.0000	-0.4284
16	-6.4276	0.0000	-0.0213
17	-6.2601	-0.0000	-0.0225
18	-6.3957	0.0000	-0.4285
19	-6.3938	0.0000	
20	-6.2956	0.0000	
21	-6.2974	0.0000	
22	-12.7961	0.0000	
23	-12.9928	-0.0010	
24	-12.9922	0.0001	
25	-12.7967	-0.0000	

<sup>\*)</sup> DOF = solmun vapausasteet s.e. solmun 1 DOF = 1, 2, 3, solmun 2 DOF = 4, 5, 6 jne.

Tulos ei ole tarkka optimi, mutta se täyttää lopetuskriteerit. Optimoitu rakenne on kuvassa 5:



**Kuva 5 Optimoitu rakenne. Solmut 1 ja 2 yhtyvät.**

## ERIKOISTAPAUUS: PUHDAS GEOMETRIAN OPTIMOINTI

Jatketaan testausta. Asetetaan nyt pinta-alat arvoon  $400 \text{ mm}^2$  (kaikki sauvat). Arvo on melko suuri, jolloin geometrialle saadaan enemmän 'pelivaraa'. Optimoidaan kaikkien vapaiden solmujen 1,2,...,6 koordinaatit. Ts. ongelmassa on nyt yhteensä 18 suunnittelutai tarkemmin paikkamuuttujaa. Sallitut paikkamuuttujien arvot on annettu taulukossa 6:

**Taulukko 6. Paikkamuuttujien rajat (mm).**

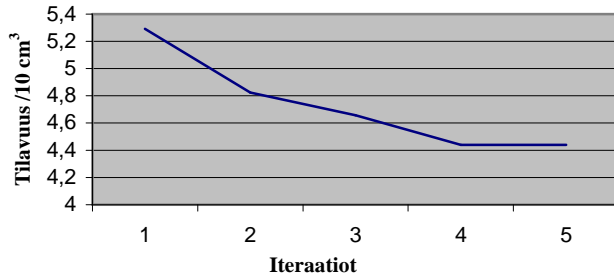
	$x_{sall\ a}$	$x_{sall\ y}$	$y_{sall\ a}$	$y_{sall\ y}$	$z_{sall\ a}$	$z_{sall\ y}$
<b>Solmu 1</b>	-4000	<0	-1000	1000	5000	8000
<b>Solmu 2</b>	>0	4000	-1000	1000	5000	8000
<b>Solmu 3</b>	-4000	-100	100	3000	1000	4500
<b>Solmu 4</b>	100	4000	100	3000	1000	4500
<b>Solmu 5</b>	100	4000	-3000	-100	1000	4500
<b>Solmu 6</b>	-4000	-100	-3000	-100	1000	4500

Lähtörakenne ja -data on muuten sama kuin edellä.

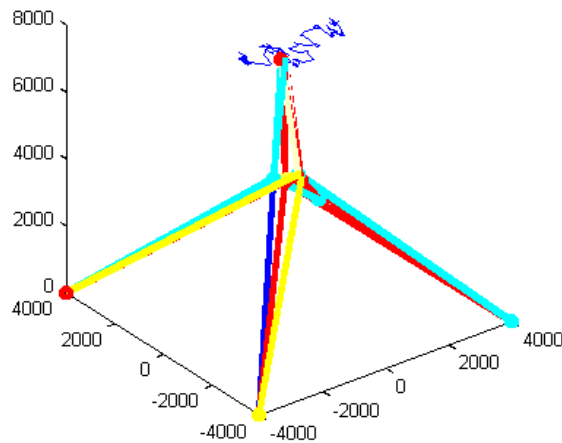
## RATKAISU

Jälleen optimirakenne on symmetrinen ja solmut 1 ja 2 yhtyvät. Vaikka optimirakenne näyttääkin lähes symmetriseltä, jännitykset sauvoissa osoittavat optimirakenteessa olevan kuitenkin epäsymmetriaa. Ts. suunnittelumuuttujat tulisi asettaa s.e. ne noudattavat symmetriaa. On siis vaadittava, että optimoitavat solmut sijaitsevat

symmetrisesti myös optimissa. Tilavuus iteraatioittain on esitetty kuvassa 6 ja optimoitu rakenne kuvassa 7. Iteraatioilla tarkoitetaan tässä samaa kuin edellä kuvassa 4.



**Kuva 6. Tilavuus iteraatioittain.**



**Kuva 7. Optimoitu rakenne. 'Sykkyrä' huipussa osoittaa solmujen 1 ja 2 sijainnin iteraatioittain (sis. myös sisemmät iteraatiot).**

Suunnittelumuuttujat optimissa ovat likimäärin seuraavat (mm)\*:

- $\mathbf{x}_1 \cong [ >0, 0, 7300 ]^T$  (solmu 1),
- $\mathbf{x}_2 \cong [ >0, 0, 7300 ]^T$  (solmu 2),
- $\mathbf{x}_3 \cong [ -300, 100, 3800 ]^T$  (solmu 3),
- $\mathbf{x}_4 \cong [ 300, 100, 3800 ]^T$  (solmu 4),
- $\mathbf{x}_5 \cong [ 300, -100, 3800 ]^T$  (solmu 5),
- $\mathbf{x}_6 \cong [ -300, -100, 3800 ]^T$  (solmu 6).

\* Solmut eivät saa mennä päällekkäin.

## YHTEENVETO

Layout-optimoinnissa eteen tulleita ongelmakohtia ALM-algoritmillä:

- Parametrien asettaminen hankalaa (kaikkea ei voi optimoida).
- Ratkaisu suppenee hitaasti.
- Syntyy numeerista epätarkkuutta.
- Mikäli suppenee liian hitaasti s.e. esim. sallittu ali-iteraatioiden määrä ylittyy, ajaudutaan mahdollisesti käyvän alueen ulkopuolelle, jolloin sisäpistealgoritmi keskeytyy.
- Suuri geometrian muuttuminen alkuperäisestä, jolloin
  - jäsenten 'häviäminen' ja siitä johtuvat singulariteetit
  - rakenteen muuttuminen totaalisesti toiseksi
  - elementtiverkko vaatii päivitystä optimoinnin edetessä (hankalaa kuoret, solidit?).
- Herkkyysanalyysin vaikeus (ja sen implementointi) mm. jännitysrajoitusten osalta
- Herkkyys nimenomaan paikkamuuttujien suhteen suuri.
- Herkkyysanalyysiltä edellytetään hyvää tarkkuutta ja luotettavuutta, ehdoton vaatimus suppenemisen varmistamiseksi.
- Optimointitehtävän (instanssin) asettaminen: kohdefunktio(t), rajoitusehdot. Erilainen tehtävän asettelu johtaa erilaiseen optimaaliseen lopputulokseen, mutta mikä on tarkoituksenmukaisin.

Koska iteraatioita ja analyyseja on tyypillisesti paljon, syntyy numeerista epätarkkuutta. Esim. testiongelmassa iteraatioiden aikana syntyy siirtymää  $y$ -suunnassa symmetriasta huolimatta. Siksi tehtävä tulisi asettaa niin, että se tukee symmetriaa. On esim. vaadittava, että sauvojen 8 ja 9 alat (kuva 3) ovat yhtäsuuret myös suunnittelumuuttujina.

ALM-menetelmän edut kyseisessä tehtävässä:

- Yleensäkin gradienttipohjaisten menetelmien suppeneminen ja saavutettavien tulosten luotettavuus on tyypillisesti hyvä, sikäli kuin menetelmien ja toisaalta instanssin erityisvaatimukset otetaan huomioon.
- Menetelmä ei ole kovin herkkä sakkoparametrin  $r$  valinnasta.
- Sisäpistemenetelmä on turvallinen, tulokset hyväksyttäviä usein myös keskeytystilanteissa.
- Lagrangen kertoimet 'kertovat' paljon: rajoitusehtojen ja/tai optimointimuuttujien merkitys myös keskeytystilanteissa, onko optimi jne.

Mitä seuraavaksi:

- Epälineaaristen rakenteiden geometrian optimointi. Geometrinen ja/tai materiaallinen epälineaarisuus ja
- Gradienttipohjaisten menetelmien mahdollisuudet tällaisten rakenteiden optimoinnissa: herkkyysanalyysi.

## LÄHTEITÄ

- [1] Haftka, R., Gurdal, Z. Elements of Structural Optimization. Third revised and expanded edition. Kluwer academic publisher, 1996, p. 481.
- [2] Vanderplaats, G. Numerical Optimization Techniques for Engineering Design: with Applications. Colorado Springs, CO, 1999, p. 441.
- [3] Bendsøe, M.P., Sigmund O. Topology Optimization. Theory, Methods and Applications. Springer, 2002, p. 370.
- [4] Broyden, C.G. Linear Equations in Optimization, in Algorithms for Continuous Optimization (Emilio Spedicato, ed.), Kluwer, 1994 pp. 25-35.
- [5] Rozvany, G.I.N et al. Optimization of Large Structural Systems. Volume I (Rozvany, G.I.N, ed.), NATO ASI Series, 1991 pp. 1-121.
- [6] Achtziger, W. Optimization with variable sets of constraints and an application to truss design, *Computational Optimization and Applications* **15**(1): 69-96, 2000.
- [7] Kirsch, U. Optimum Structural Design, McGraw-Hill, 1981.
- [8] Cheng G.D., Guo, X.  $\varepsilon$ -relaxed approach in topology optimization, *Structural Optimization* **13**: 258-266, 1997.
- [9] Robinson S.M. A quadratically convergent algorithm for general nonlinear programming problems, *Mathematical Programming* **3**: 145-156, 1972.

Sami Holopainen, dipl.ins.

TTY, Teknillisen mekaniikan ja  
optimoinnin laitos, PL 589  
33101 Tampere  
[sami.holopainen@tut.fi](mailto:sami.holopainen@tut.fi)