

ELEMENTIT JA SOLMUT TERÄSRUNGON TUOTEMALLISSA

Markku Heinisuo
Juha Hyvärinen

Rakenteiden Mekaniikka, Vol. 29
Nro 2, 1996, s. 3-18

TIIVISTELMÄ

Artikkelissa tarkastellaan kirjallisuudessa esitettyjen tuotemallien niitä osia, jotka käsittelevät elementtimenetelmän solmujen ja elementtien välisiä relaatioita. Tuotemalleja ja niiden määrittelyyn käytettyjä tuotetietomalleja arvioidaan ja esitetään mahdollisimman yksinkertaisen esimerkin avulla, että tuotemallit eivät ole yksikäsitteisiä sekä millaisia valintoja niihin voidaan kytkeä. Tärkein asia tuotemallissa ei ole se, kuinka se on kuvattu vaan se, että sen lukeminen ja kirjoittaminen johtaa yksikäsitteiseen tulokseen mahdollisimman vähällä tietokoneajalla. Tuotemallin rakenteella voidaan vaikuttaa sen käyttökelpoisuuteen.

JOHDANTO

Rakennusprojektin suunnittelijat, osien valmistajat, kuljettajat, asentajat sekä tuotteen loppukäyttäjät ja viimein purkajat käyttävät nykyään työssään mitä kirjavampia atk-työkaluja (hardware ja software). Mikään atk-työkalu ei ole saavuttanut monopoliasemaa eikä tulle sitä saavuttamaan. Tällaisessa ympäristössä kohdistuu projektiin osallistijille paineita tiedon siirron toteuttamisessa osapuolelta toiselle. Paineita syntyy jo yritysten sisällä. Esimerkiksi rakenteen geometria hallitaan usein eri ohjelmalla kuin sen lujuusopillinen analyysi ja mitoitus, sillä saatavilla ei juurikaan ole ohjelmia, joissa nämä suunnittelun työvaiheet olisi integroituna.

Tuotemallin avulla ratkaistaan em. käytännön ongelmia. Tuotemallin tietosisältö kuvataan tuotetietomallissa. Tuotemalli (product model) on tuotteen yksittäisen ilmentymän (esim. tietyn rakennuksen rungon) tiedot tallennettuna tuotetietomallin mukaisesti jäsennettynä esim. CAD-ohjelman käyttämässä tietokannassa tai STEP-siirtotiedostossa (ISO-10303-21). Tuotetietomalli (product data model) eli tuotetietojen käsitelmä jäsentää jollakin formaalilla, eksaktilla mallintamismenetelmällä (esim. EXPRESS, EXPRESS-G, NIAM, IDEF1X) tuotteen (esim. rakennuksen) tiedot ja tietojen väliset yhteydet käsitteellisellä tasolla. Tuotetietomallin perusteella voidaan toteuttaa ATK-sovellutuksia (esim. CAD-ohjelmia ja tietokantoja tai siirtotiedostoja), joissa käsitellään yksittäisten tuotteiden tietoja (tuotemalleja).

Seuraavassa tarkastellaan yksinomaan tiedonsiirtoa teräsrunkojen suunnittelussa, mutta tulokset ovat osittain yleispäteviä. Esimerkkinä käsitellään FEM-laskennasta tuttuja olioita **elementtejä ja solmuja** ja näiden välisten riippuvuuksien määrittelyä tuotetietomallissa. Tavoitteena on osoittaa, että tuotetietomalli voidaan "rakentaa" usealla tavalla. Parasta vaihtoehtoa ei ilmeisesti ole vielä löytynyt. Vaikka sama tietosisältö voidaan määrittellä usealla eri tavalla, niin lopputuloksen tulee olla yksikäsitteinen. Kirjallisuudessa malleista ei näe useinkaan syvällisempää analyysiä. Ongelma asetetaan yleensä niin, että esitetään probleema ja sille ratkaisu eli tuotetietomalli kaaviona. Käytännön työn kannalta ratkaisevaa on mutkikkaiden mallien testaus laajan testimateriaalin avulla. Riittävän laajan testimateriaalin avulla on testattava tuotemallin lukemista ja sinne kirjoittamista. Seuraavassa malleja tarkastellaan teoreettisesti. Tarkastellut mallit ovat voimakkaasti yksinkertaistettuja. Tavoitteena on myös esittää yksinkertaisilla esimerkeillä, kuinka tuotemallin sisältöä voidaan muuttaa.

Tuotetietomallit kuvataan tässä artikkelissa standardin ISO-10303 (STEP) EXPRESS (10303-11) kuvauskieltä käyttäen. Tuotemallien fyysinen formaatti on standardin osan ISO-10303-21 mukainen. Artikkelin liittyy TTKK:ssa tehtävään Suomen Akatemian rahoittamaan tutkimukseen "Asiantuntijajärjestelmät teräsrakenteiden suunnittelussa".

Esimerkit ovat syntyneet kehitettäessä asiantuntijajärjestelmän avulla generoitavaa analyysin tuotemallia (Heinisuo, 1996) teräsrungolle, kun lähtökohtana on rungon geometrian tuotemalli (Hyvärinen, 1994, Heinisuo, Hyvärinen, 1995).

Tavoite on, että teräsrungon geometrian sisältävästä tuotemallista saadaan tuotemalli rungon analyysiä varten ilman suunnittelijan merkittävää lisätyötä. Analyysissä voidaan asettaa mm. kriittisiin liitoksiin sauvamallia "järeämpi" analyysimalli. Alustavia pilottisovellutuksia on laskettu ja kehitettävän asiantuntijajärjestelmän perusideoita on esitelty (Heinisuo, Hyvärinen, 1995b, Heinisuo, 1996b, Hyvärinen 1996b). Sovellettavan asiantuntijajärjestelmän kehitys on työn alla (Hyvärinen, 1996). Atk-avusteisia järjestelmiä, joissa on integroitu geometria- ja analyysimalli, on olemassa suljetuissa systeemeissä mm. teräsrungoille (Heinisuo et al, 1991) ja puurakenteille (Mikkola, 1985).

KIRJALLISUUDEN TUOTEMALLEJA

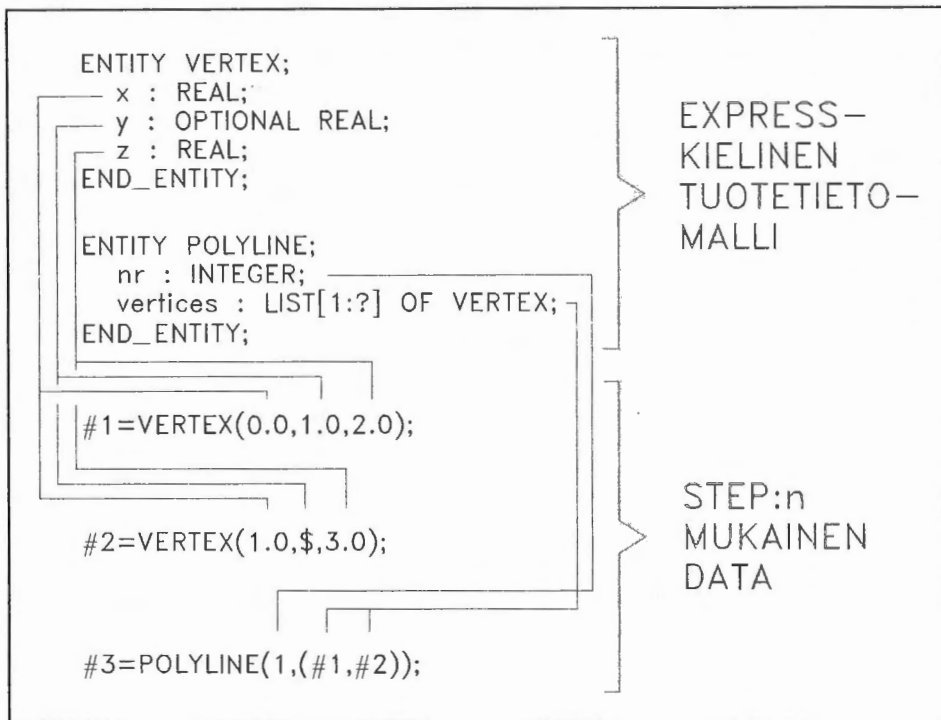
Teräsrunkojen tuotemalleja, joihin kuuluu analyysimalli, on esitetty ainakin lähteissä (Haller, 1994) ja (CIMsteel, 1995). Näiden analyysimallit perustuvat (Bernoulli-Euler-tyyppisiin) sauvaelementteihin. Analyysimallille (FEM) on esitetty sovellutusriippumaton "standardiluonnos" ISO-10303-104. Olio-ajatteluun perustuvia FEM-ohjelmia on tehty (Hededal, 1995). Seuraavat solmun ja elementin tehtävät on vapaasti käännetty lähteestä (Hededal, 1995, s. 25).

Solmu on elementtien liittymäkohta ja solmun tehtävä on hallita probleeman primäärisiä suureita: vapausasteita ja kuormituksia. Solmuihin diskretoidaan koko analyysi. Esikäsitellyssä solmut lähettävät ratkaistaville yhtälöille vapausasteet ja kuormitukset. Jälkikäsitellyssä solmut ottavat vastaan analyysin tulokset. Jokainen vapausaste ja vastaava kuormitus muodostaa duaalisen parin, jossa jompikumpi on ennalta määritelty, jotta tehtävällä olisi yksikäsitteinen ratkaisu. Solmu saattaa määritellä myös pisteen

fysikaalisessa kappaleessa. Elementin tehtävä on huolehtia jakautuneista ominaisuuksista. Elementillä on tehtäviä esi- ja jälkikäsitellyssä. Esikäsitellyssä elementti kokoaa geometria- ja materiaaliominaisuudet kontinuumista (tarvittavat oliot ominaisuudet ja materiaalit) ja tiivistää informaation solmuihin. Jälkikäsitellyssä elementti jakaa diskreetin solmuinformaation kontinuumiin. Elementtimenetelmän keskeiset olioluokat ovatkin **solmut**, **elementit**, **materiaalit** ja elementtien geometriset **ominaisuudet**.

Seuraavassa käsitellään vain solmuja ja elementtejä. Aluksi esitellään näiden tuotetietomallien EXPRESS-kieliset määritelmät. Määritelmistä on riisuttu pois kaikki muu paitsi elementtimenetelmässä tarvittavat elementtien ja solmujen keskinäisiä riippuvuuksia kuvaavat attribuutit. Täsmällisemmät olioiden määritelmät löytyvät lähteistä. Koostumushierarkia saadaan aikaan EXPRESS-kielillä asettamalla olion attribuutin tietotyyppi toinen olio.

Kuvassa 1 on esimerkki (Haller, 1994) EXPRESS-kielisestä tuotetietomallin kuvauksesta ja vastaavasta tuotemallista. Kuvan 1 esimerkissä on kaksi olioluokkaa nimeltään VERTEX ja POLYLINE sekä näillä attribuutit nimeltään **x**, **y**, **z** ja **nr**, **vertices**. Attribuutin **y** arvo tuotemallissa on vapaaehtoinen (OPTIONAL => \$-merkki datassa) ja attribuutin **vertices** tietotyyppi on yksi tai useampi VERTEX-luokan olio. LIST-aggregaatti selvitetään myöhemmin. Olioluokan määrittäminen alkaa merkillä ENTITY ja loppuu merkeihin END_ENTITY ja ;.



Kuva 1. Esimerkki tuotetietomallista ja tuotemallista

Väitöskirjan (Haller, 1994) oliot ELEMENT ja NODE ja niiden välinen riippuvuus on määriteltä seuraavasti (määrittely no 1)

```

ENTITY ELEMENT;
  n : LIST[2:?] OF NODE;
END_ENTITY;

```

```
ENTITY NODE;  
END_ENTITY;
```

Nyt on huomattava, että eo määrittelyistä on riisuttu muut määrittelyt pois, paitsi seuraavassa tarvittavat. Esimerkiksi solmujen attribuutteina ovat mm. koordinaatit. Merkintä LIST[2:?] OF NODE tarkoittaa sitä, että olioon ELEMENT liittyy aina kaksi tai useampia attribuutteja nimeltään n, tietotyypiltään LIST OF NODE ja niiden (NODEjen) paikkaa ei saa vaihtaa (järjestetty) ja lisäksi sama NODE voi olla useasti samassa elementissä (voi olla duplikaatteja).

EUREKA-hankkeessa (CIMsteel, 1995) on seuraavat määrittelyt (määrittely no 2)

```
ENTITY ELT_NODE_CONNECTIVITY;  
elt_node_con_number : INTEGER;  
connecting_elt : ELEMENT;  
connecting_node : NODE;  
END_ENTITY;
```

Attribuutti nimeltään elt_node_con_number (tietotyyppi INTEGER) ilmoittaa solmujen järjestyksen elementissä. Jos on kyseessä "alkusolmu", niin attribuutin arvo on 1, "loppusolmulla" on arvo 2.

(määrittely no 2 jatkuu)

```
ENTITY ELEMENT;  
INVERSE  
  connectivity : SET[1:?] OF ELT_NODE_CONNECTIVITY  
  FOR connecting_elt;  
END_ENTITY;
```

Aggregaatti SET ilmoittaa, että liittyvät oliot ovat vapaassa järjestyksessä ja

duplikaatteja ei sallita (LIST oli järjestetty ja duplikaatit sallittiin). INVERSE selvitetään seuraavassa.

(määrittely no 2 jatkuu)

```
ENTITY NODE;
```

```
INVERSE
```

```
  connecting_ends : SET[0:?] OF ELT_NODE_CONNECTIVITY
```

```
  FOR connecting_node;
```

```
END_ENTITY;
```

Olioluokan NODE määritelmä tarkoittaa sitä, että nollan tai useamman olion ELT_NODE_CONNECTIVITY attribuutin connecting_node attribuuttina on NODE eli käytännössä yksistään tällä määrittelyllä ei NODE (eikä ELEMENT, vrt. alkuosa määrittelyä no 2) näy lopullisessa tuotemallissa.

"Standardiluonnoksessa" ISO-10303-104 on annettu seuraavat määritelmät (määrittely no 3)

```
ENTITY ELEMENT;
```

```
required_node_list : LIST[1:?] OF NODE;
```

```
END_ENTITY;
```

```
ENTITY NODE;
```

```
END_ENTITY;
```

Edellä esitellyt EXPRESS-kieliset tuotetietomallien osat tuottavat tuotemalliin seuraavannäköistä dataa (ASCII-file, ISO-10303-21) kuvan 2 esimerkkitapauksessa.

NODE-----NODE
ELEMENT

Kuva 2. Yksi elementti ja kaksi solmua

Määrittely no 1 ja 3:

```
#1=ELEMENT((#2,#3));  
#2=NODE;  
#3=NODE;
```

Määrittely no 2:

```
#1=ELT_NODE_CONNECTIVITY(1,#2,#3);  
#2=ELEMENT;  
#3=NODE;  
#4=ELT_NODE_CONNECTIVITY(2,#2,#5);  
#5=NODE;
```

INVERSE ei näy tuotemallissa, joten tuotemallissa ei näy olioilla #2 (elementti), #3 ja #5 (solmut) attribuutteja tässä yksinkertaistetussa tapauksessa. Arvioidaan seuraavaksi eo. määrittelyjä.

MALLIEN ARVIOINTIA

Määrittelyssä no 1 lähtökohtana on sauvaelementit, joten solmujen lukumäärä on vähintään kaksi. Yllättävää on se, että saman solmun sallitaan esiintyvän useasti (LIST

sallii duplikaatit) samassa elementissä. Jos LIST korvataan SET määrittelyllä määrittelyssä no 1, niin tällöin duplikaatteja ei sallita, mutta solmujen järjestys voi vaihdella. Jos solmujen järjestystä voi vaihdella elementin määrittelyssä, niin on odotettavissa vaikeuksia. Esimerkiksi kolmesolmuisella kaarevalla sauvaelementillä on tärkeää tietää, missä järjestyksessä solmut esiintyvät elementissä. Tietotyyppi, joka on järjestetty ja ei salli duplikaatteja sisältyy seuraavaan olioluokan määrittelyyn (UNIQUE on optiona standardissa LIST aggregaatille)

```
ENTITY ELEMENT;  
nodes : LIST[2:?] OF UNIQUE NODE;  
END_ENTITY;
```

Määrittely no 3 muistuttaa määrittelyä no 1, paitsi että solmuja voi olla yksi tai useampi. Yksi solmu voi olla esimerkiksi jousielementissä. Kummastakaan määrittelystä (no 1 tai no 3) ei seuraa tuotemallin NODE-olioon informaatiota siitä, mihin elementteihin solmu liittyy. Kun katsotaan vastaavia tuotemalleja, niin nähdään, että tämä tieto on niistä löydettävissä. Tuotemalleista voidaan etsiä elementit ja tehdä vastaavat taulut, joista tieto löytyy. Tyypillisesti olioita on tietokannassa kymmeniä tuhansia tai satoja tuhansia tai miljoonia riippuen mallin tarkkuudesta. Koko tietokannan selaus kestää kauan, joten hakua nopeuttaa, jos solmuissa olisi valmiina viittaus elementteihin. Toisaalta, jos tieto on molemmissa, niin voidaan päätyä ristiriitaan. Tietokannan ollessa iso ja mutkikas, ei koskaan voi olla liian varma ratkaisun yksikäsitteisyydestä.

Määrittelyn no 2 mukaisessa tuotemallissa on elementin ja solmun välinen riippuvuus annettu aina oliossa ELT_NODE_CONNECTIVITY, jolla on attribuuttina yksi solmu ja yksi elementti. Tällaisen vakiomittaisen attribuuttijonon käsittely (selaus ja haku) saattaa olla käytännössä nopeampaa, kuin tuntemattoman mittaisen attribuuttijonon selaus. Ainakin hakuohjelman tekeminen on helpompaa. Nämä väitteet on luonnollisesti testattava käytännössä. Lisäksi olion ELT_NODE_CONNECTIVITY täydellisessä

määrittelyssä (CIMsteel, 1995) on attribuutti fixity, jolla voidaan määritellä kunkin elementin liittymisjäykkyys solmuun. Näin on helppo määritellä esimerkiksi nivel (tai tietty rotaatiojäykkyys) tietyn sauvan päähän solmussa, johon liittyy useita sauvoja. Tämä ominaisuus on hankala määritellä esim. lähteen (Haller, 1994) tuotemallissa. Solmuun liittyviä elementtejä tarvitaan mm. jälkikäsitelyssä.

Duplikaattisolmuja voidaan sallia joissakin nelisolmuisissa elementeissä rajoitetusti. Esimerkiksi sama solmu voi esiintyä kahdesti muodostaen geometrialtaan kolmionmuotoisen elementin, jonka käsittely (esim. jäykkyyssmatriisin laskenta) voidaan suorittaa nelisolmuisen elementin kaavoilla. Jos sama solmu esiintyy kolme kertaa nelisolmuisessa elementissä, niin vaikeuksia on luvassa. Tuotemallien rakentamisessa on syytä pyrkiä estämään tällaiset tilanteet.

Muutetaan seuraavaksi määrittelyssä no 2 INVERSE määritykset EXPLICIT-attribuuteiksi.. Lopputulos on (määrittely no 4)

```
ENTITY ELT_NODE_CONNECTIVITY;  
connecting_elt : ELEMENT;  
connecting_node : NODE;  
END_ENTITY;
```

```
ENTITY ELEMENT;  
connectivity : SET[1:?] OF ELT_NODE_CONNECTIVITY;  
END_ENTITY;
```

```
ENTITY NODE;  
connecting_ends : SET[0:?] OF ELT_NODE_CONNECTIVITY;  
END_ENTITY;
```

Tämän mukainen tuotemalli on kuvan 2 tapauksessa

```

#1=ELT_NODE_CONNECTIVITY(#2,#3);
#4=ELT_NODE_CONNECTIVITY(#2,#5);
#2=ELEMENT((#1,#4));          (tai #2=ELEMENT((#4,#1)); vrt. SET)
#3=NODE((#1));
#5=NODE((#4));

```

Määrittely no 4 ei eksplisiittisesti estä kirjoittamasta esimerkin tuotemallia seuraavasti

```

#1=ELT_NODE_CONNECTIVITY(#2,#3);
#4=ELT_NODE_CONNECTIVITY(#2,#5);
#2=ELEMENT((#1,#4));          (tai #2=ELEMENT((#4,#1)); vrt. SET)
#3=NODE((#4));
#5=NODE((#2));

```

joka selvästi johtaa ei-haluttuun lopputulokseen. Tällaisen virheen estämiseksi voidaan käyttää INVERSE määritystä vaikkakin po. virhe tuntuu erittäin harvinaiselta.

Määrittelyssä no 2 voi olla solmuja, jotka eivät liity mihinkään elementtiin (SET[0:?]). Sen sijaan elementit liittyvät aina vähintään yhteen solmuun (SET[1:?]). Olio ELT_NODE_CONNECTIVITY pitää kirjata siitä, mitkä solmut ja elementit liittyvät toisiinsa, ja solmujen järjestys elementissä ilmoitetaan elt_node_con_number attribuutilla.

Seuraavassa määrittelyssä otetaan huomioon solmujen järjestys elementissä ja duplikaatteja ei voi esiintyä (määrittely no 5)

```

ENTITY ELEMENT;
nodes : LIST[1:?] OF UNIQUE NODE;
END_ENTITY;

```

```
ENTITY NODE;  
elements : SET[0:?] OF ELEMENT;  
END_ENTITY;
```

Tämän mukainen tuotemalli on kuvan 2 tapauksessa

```
#1=ELEMENT((#2,#3));  
#2=NODE((#1));  
#3=NODE((#1));
```

Määrittelyä voidaan täsmentää tarvittaessa INVERSE:llä (määrittely no 6)

```
ENTITY ELEMENT;  
nodes : LIST[1:?] OF UNIQUE NODE;  
END_ENTITY;
```

```
ENTITY NODE;  
INVERSE  
  elements : SET[0:?] OF ELEMENT  
  FOR nodes;  
END_ENTITY;
```

Määrittelyn no 6 mukaisen tuotemallin teko jää kotitehtäväksi lukijalle. Määrittelyistä no 5-6 nähdään, että oliota `ELT_NODE_CONNECTIVITY` ei tarvita solmujen ja elementtien koostumuksen määrittelyä varten. Kuten edellä on todettu, elementin kiinnitys joustavasti solmuun vaatii lisämäärittelyä näissä malleissa. Määrittelyssä no 6 ei ole itse asiassa mitään lisäarvoa (paitsi ehkä lukijaystävällisyys) verrattuna määrittelyyn no 7:

ENTITY ELEMENT;
nodes : LIST[1:?] OF UNIQUE NODE;
END_ENTITY;

ENTITY NODE;
END_ENTITY;

FEM-OHJELMIEN KIRJANPIDOSTA

FEM-ohjelmissa kirjanpito solmujen ja elementtien välillä toteutetaan mm. *ID*-, *IEN*- ja *LM*-taulukoiden avulla (Hughes, 1987). *ID*-taulukosta (2D) löytyy globaalien mallin yhtälöiden numerot eli vapaiden vapausasteiden numerot, jos annetut vapausasteet ovat nollia. Taulukon sarakenumero on globaali solmunumero ja taulukon rivinumero on lokaali vapausasteen numero. *IEN*-taulukossa (2D) sarakenumero on elementtinumero, rivinumero on elementin lokaali solmunumero ja alkioina ovat globaalit solmunumerot. *LM*-taulukosta (3D, dimensio $n_{ed} * n_{en} * n_{el}$, merkinnät ks. (Hughes, 1987)) löytyvät alkioina mallin globaalit yhtälönumerot (vapausasteet), kun alkion ensimmäinen indeksi on paikallinen vapausasteen numero (*i*), toinen lokaali solmunumero (*a*) ja kolmas elementin numero (*e*) eli kaavana

$$LM(i,a,e) = ID(i,IEN(a,e))$$

Tämä esitetään usein kaksiulotteisena (dimensiot $n_{ee} * n_{el}$) taulukkona, jossa käytetään lokaalia yhtälönumeroa (*p*)

$$LM(p,e) = LM(i,e,a), p=n_{ed}(a-1)+i$$

Havainnollisia esimerkkejä kaavojen käytöstä löytyy lähteestä (Hughes, 1987, ks. esim. s. 92 - 94). Näille voidaan kirjoittaa standardin mukaiset tuotetietomallit (myös harjoitustehtävänä lukijalle).

YHTEENVETO

Esimerkeistä nähdään, että tuotetietomalli voidaan "rakentaa" monella tavalla. Kirjallisuudessa esitetyissä tuotemalleissa on tehty erilaisia valintoja. Näitä valintoja kannattaa arvioida uusia malleja tehtäessä. Tärkein asia on tuotemallin (tietosisällön) yksikäsitteisyys. Yksikäsitteisyyttä voidaan varmistaa eri tavoin. Samoin tuotemallin rakenteella voidaan vaikuttaa tuotemallien käyttökelpoisuuteen käytännön sovellutuksissa. Erilaisten tuotemallien rakenteiden vaikutus todellisissa tehtävissä jää testattavaksi tulevaisuudessa.

Kirjallisuudesta ei löytynyt valmista tuotemallia teräsrunkojen analyysiin, joka sisältäisi tässä tutkimuksessa halutut ominaisuudet (mm. deplanoituvat ja vääristyvät sauvat, ominaisarvon laskenta, levymallien yhdistely sauvamalleihin erityisesti liitosten analyysimalleissa), joten se tulee kehittää seuraavaksi ja se on työn alla projektissa. Kussakin kirjallisuudessa esitetyssä mallissa on tiettyjä hyviä piirteitä, ja kirjallisuuden mallit kannattaa arvioida ennen oman mallin tekemistä.

KIITOKSET

Tekijät kiittävät Suomen Akatemiaa rahallisesta tuesta, DI Antero Miettistä ja prof. Markku Tuomala kriittisistä kommentteista artikkeliin, DI Ari Aaltoa kuvan 1 olemassaolosta ja TRY r.y.:tä CIMsteel dokumentin käyttömahdollisuudesta.

LÄHDEKIRJALLISUUS

CIMsteel Integration Standards (Release One), University of Leeds, Department of Civil Engineering, 23.6.1995

Haller H.-W., Ein Produktmodell für den Stahlbau, Dissertation, Bauingenieur- und Vermessungswesen der Universität Friedericiana zu Karlsruhe (TH), Karlsruhe, 1994

Hededal O., Object Oriented Structuring of Finite Elements, Ph.D.-Thesis, Department of Building technology and Structural Engineering, Aalborg Universitet, AUC, Aalborg, Danmark, Engineering Mechanics, Special Report No 1, 1994

Heinisuo M., Möttönen A., Paloniemi T., Nevalainen P., Automatic design of steel frames in a CAD-system, Proceedings of the 4th Finnish Mechanics Days, Ed. Niemi E., Research Papers 17, Lappeenranta University of Technology, Lappeenranta 1991, pp. 197-204

Heinisuo M., Expert Systems for Design of Steel Structures, Part 5: Hierarchical models of steel skeleton analysis, Tampere University of Technology, Department of Civil Engineering, Structural Mechanics, Report 20, Tampere, 1996 (In preparation)

Heinisuo M., Product model application in analysis of structural steel design, Teräsrakentamisen toiset tutkimus- ja kehityspäivät, Tampere-talo, 11.-12.1.1996, TRY ry, 1996

Heinisuo M., Hyvärinen J., Hierarchical aggregation form of steel skeleton, Product and Process Modelling in the Building Industry, Ed. R.J. Scherer, A.A. Balkema, Rotterdam, 1995, pp. 147-154

Heinisuo M., Hyvärinen J., Steel joint analysis using product model, Eight Nordic Seminar on Computational Mechanics, Chalmers University of Technology, Publication 95:15, Eds. R. Larsson, N.-E. Wiberg, 1995, pp. 1-4

Hughes T.J.R., The Finite Element Method, Linear Static and Dynamic Finite Element Analysis, Prentice-Hall, Inc, Englewoods Cliffs, New Jersey, 1987

Hyvärinen J., Expert Systems for Design of Steel Structures, Part 2: Hierarchical aggregation form of steel skeleton, Tampere University of Technology, Department of Civil Engineering, Structural Mechanics, Report 15, Tampere, 1994

Hyvärinen J., Expert systems for design of steel structures, Licenciate thesis, Tampere University of Technology, 1996 (In preparation)

Hyvärinen J., Intelligent modeling of structural steel joints, Teräsrakentamisen toiset tutkimus- ja kehityspäivät, Tampere-talo, 11.-12.1.1996, TRY ry, 1996

ISO DIS 10303-11: Industrial Automation Systems - Product Data Representation and Exchange, Part 11: The EXPRESS language reference manula, 1992

ISO DIS 10303-21: Industrial Automation Systems - Product Data Representation and Exchange, Part 21: Clear text encoding of the exchange structure, 1993

ISO 10303-104 - FEA Reference Model, ISO TC184/SC4/WG3/P9, Document N 21, 3 January, 1991

Mikkola M., Computer Aided Design of Wood Trusses Using Nail Plates, Proceedings of CIVIL-COMP 85, Volume 1, CIVIL-COMP PRESS, Edinburgh, 1985, pp. 149-154

Markku Heinisuo, Tekn. tri, vanhempi tutkija, Suomen Akatemia c/o TTKK
email: markku2@junior.ce.tut.fi

Juha Hyvärinen, DI, tutkija, Suomen Akatemia c/o TTKK
email: juha@junior.ce.tut.fi