



Ville Katisko

Graafisen algoritmieditorin käyttö mallintamisessa

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Rakennustekniikka

Insinöörityö

28.4.2024

Tiivistelmä

Tekijä:	Ville Katisko
Otsikko:	Graafisen algoritmieditorin käyttö mallintamisessa
Sivumäärä:	33 sivua + 0 liitettä
Aika:	28.4.2024
Tutkinto:	Insinööri (AMK)
Tutkinto-ohjelma:	Rakennustekniikka
Ammatillinen pääaine:	Rakennetekniikka
Ohjaajat:	Joni Pirhonen Lehtori, Metropolia Sami Katisko Suunnittelupäällikkö, Alekski Katisko Rakennesuunnittelija, Tekova Oy

Insinööriyön aiheena oli tutkia parametrisen suunnittelun soveltuvuutta rakennesuunnittelussa ja kehittää algoritmipohjaa Grasshopper-ohjelmaa käyttäen. Työ toteutettiin Tekova Oy:lle, joka toimii toimitilatoteuttajana ympäri Suomea.

Työn tavoitteena oli luoda algoritmipohja, joka mahdollistaa rakenteiden mittojen ja muotojen nopean mukauttamisen suunnitteluvaiheessa, parantaen näin suunnittelu-prosessin tehokkuutta ja joustavuutta.

Työ toteutettiin hyödyntämällä aiempaa hallirakennusprojektia. Algoritmipohjan luomisessa määriteltiin ensin parametriset muuttujat, jotka vastaavat hallin rakenteiden mittoja, mukaan lukien pilarit, palkit ja anturat. Työn kuluessa käytettiin Grasshopper-komponentteja, jotka linkittyivät Tekla Structures -ohjelmiston kanssa mahdollistaen mallin reaaliaikaisen päivittymisen.

Olellaisina tuloksina havaittiin, että algoritmipohjaisella suunnittelulla voidaan merkittävästi nopeuttaa rakenteellisten muutosten tekoa ja parantaa mallin tarkkuutta. Lisäksi työssä todettiin, että parametrisen mallinnuksen avulla voidaan vähentää suunnitteluun liittyviä inhimillisiä virheitä, kun muutokset mallissa päivittyvät automaattisesti kaikkiin liittyviin komponentteihin.

Tulokset osoittavat, että parametrisen suunnittelun ja algoritmipohjaisten työkalujen avulla voidaan merkittävästi edistää rakennusalan siirtymistä kohti digitaalista ja automatisoitua suunnittelua. Työn tulokset ovat hyödynnettävissä laajasti rakennus-alalla, erityisesti sellaisten suunnittelu-prosessien kehittämisessä, joissa vaaditaan korkeaa mukautuvuutta ja nopeaa reagointikykyä suunnittelumuutoksiin.

Avainsanat: mallintaminen, Grasshopper, Tekla Structures

Tämän opinnäytetyön alkuperä on tarkastettu Turnitin Originality Check -ohjelmalla.

Abstract

Author: Ville Katisko
Title: Use of Graphical Algorithm Editor in Modelling
Number of Pages: 33 pages + 0 appendices
Date: 28 April 2024

Degree: Bachelor of Engineering
Degree Programme: Civil Engineering
Professional Major: Structural Engineering
Supervisors: Joni Pirhonen, Senior Lecturer, Metropolia UAS
Sami Katisko Planning Manager, Alekski Katisko Structural Engineer, Tekova Oy

The subject of the engineering thesis was to explore the applicability of parametric design in structural engineering and to develop an algorithmic foundation using Grasshopper software. The study was carried out for Tekova Oy, which operates as a premises contractor throughout Finland.

The aim of the thesis was to create an algorithmic base that allows for the rapid customization of the dimensions and shapes of structures during the design phase, thereby improving the efficiency and flexibility of the design process.

The study was implemented by utilizing a previous hall construction project. In creating the algorithmic base, the parametric variables that correspond to the dimensions of the hall's structures, including columns, beams, and footings, were first defined. During the work, Grasshopper components that linked with Tekla Structures software were used, enabling real-time updates of the model.

Significant results showed that algorithmic design can greatly speed up the making of structural changes and improve the accuracy of the model. In addition, the work found that parametric modelling can reduce human errors related to design as changes in the model automatically update all related components.

The results demonstrate that parametric design and algorithm-based tools can significantly advance the construction industry's move towards digital and automated design. The outcomes of the work are widely applicable in the construction industry, especially in the development of design processes that require high adaptability and quick response to design changes.

Keywords: modelling, Grasshopper, Tekla Structures

Sisällys

Lyhenteet ja selitteet

1	Johdanto	7
2	Parametrinen mallinnus ja käytetyt ohjelmistot	8
2.1	Parametrinen suunnittelu	8
2.2	Rhino	9
2.3	Grasshopper	12
2.4	Tekla Structures	14
2.5	Grasshopper-Tekla Live Link	15
3	Grasshopperin käyttö rakennesuunnittelussa	16
3.1	Parametrisen algoritmin luominen ja muokkaaminen	16
3.2	Ensimmäiset askeleet Grasshopperissa	17
3.3	Tuulipilarit ja nurkkapilarit	17
3.4	Keskipilarit	18
3.5	Pääpilarit	19
3.6	Liimapuupalkki	21
3.7	Ulkoseinäelementit	24
3.8	Anturat	25
3.9	Koko halli	26
4	Havaitut ongelmat	29
5	Haasteet	30
6	Tulokset ja jatkokehitys	31
7	Yhteenveto	32
	Lähteet	33

Lyhenteet ja selitteet

2-laivainen: Rakennuksen kehässä kaksi pääkantavaa rakenteellista elementtiä.

Auto C-Plane:

C-Plane (Construction Plane) tarkoittaa rakennustasoa tai työskentelytasoa, jolla käyttäjä työskentelee. Auto C-Plane-toiminto vaihtaa automaattisesti aktiivisen rakennustason valitun pinnan suunnitukseksi.

BIM: Building Information Modeling, eli tietomalli, jonka avulla hallitaan rakennusprojektin eri vaiheita.

Gumball: Gumball on visuaalinen työkalu 3D-ohjelmistoissa, joka tarjoaa interaktiivisen ja intuitiivisen tavan siirtää, kiertää tai skaalata valittuja kohteita.

NURBS: *Non-Uniform Rational B-Splines*. Matemaattinen malli, jota käytetään tietokoneavusteisessa suunnittelussa ja 3D-mallinnuksessa luomaan käyriä ja pintoja.

Spline: Spline on matemaattinen funktio, joka määritellään palasittain polynomeilla eli spline-käyrä muodostuu useista polynomisista osista, jotka liittyvät toisiinsa ja muodostavat käyrän.

Push-Pull: Push-Pull-ominaisuus mahdollistaa pintojen, reunojen ja muiden geometristen elementtien helposti muokkaamisen vetämällä tai työntämällä niitä.

QuadRemesh:

QuadRemesh on työkalu, jota käytetään 3D-geometrian uudelleenverkottamiseen, jolloin kohteesta syntyy nelisivuisista (quad) polygoneista koostuva verkkomainen malli.

ShrinkWrap:

ShrinkWrap on ominaisuus tai työkalu, joka luo uuden verkkopinnan olemassa olevan 3D-mallin ympärille siten, että uusi verkkopinta kuituu alkuperäisen mallin muotoon.

1 Johdanto

Tämän opinnäytetyön tarkoituksena on tutkia ja kehittää algoritmipohjaista suunnittelua, erityisesti keskittyen parametriseen mallinnukseen Grasshopper-ohjelman avulla. Parametrinen suunnittelu on noussut merkittäväksi innovaatioksi rakennusosalalla, tarjoten uudenlaisen tavan lähestyä rakenteiden suunnittelua ja optimointia. Tämä työ hyödyntää suunnitteluprosessissa Rhinoceros 3D -ohjelmistoa ja sen lisäosaa Grasshopperia.

Työ tehdään Tekova Oy:lle. Tekova on toimitilatoteuttaja. Yritys rakentaa erilaiset liike-, toimi-, logistiikka- ja urheilutilat sekä hoitaa suunnittelun, toteutuksen, luovutuksen ja takuuajan. Tekova palvelee asiakkaita ympäri suomea.

Lähtökohtana algoritmipohjan kehittämiseksi otettiin yrityksen aiemmin toteuttama halliprojekti, jonka todelliset mitat ja sijainnit tarjosivat arvokasta tietoa parametrisen mallin räätälöimiseksi. Tavoitteena on luoda käyttäjäystävällinen algoritmi ja nopeuttaa suunnitteluprosessia. Tämän avulla voidaan tuottaa Tekla Structures -ohjelmistoon tarkasti määritellyt rakenteet, kuten runko, pilari- ja palkkirakenteet, perustukset ja seinäelementit, sekä vaadittavat tukirakenteet.

Opinnäytetyön keskeisenä tavoitteena on myös osoittaa, miten parametrinen suunnittelu voi parantaa rakennusprojektien tehokkuutta, tarkkuutta ja joustavuutta. Tämä tapahtuu kehittämällä ja testaamalla algoritmeja, jotka kykenevät mukautumaan muuttuviin suunnitteluparametreihin ja projektin vaatimuksiin reaaliajassa. Tutkimuksen aihe valittiin, koska haluttiin tutkia ja saada ymmärrystä siitä, miten algoritmipohjainen suunnittelu voi muuttaa rakennusalan käytäntöjä ja edistää alaa kohti entistä digitaalisempaa tulevaisuutta.

2 Parametrinen mallinnus ja käytetyt ohjelmistot

2.1 Parametrinen suunnittelu

Parametrinen suunnittelu on mullistanut tapamme lähestyä rakennusten ja rakenteiden suunnittelua. Toisin kuin perinteinen 3D-mallinnus ja 2D-CAD, parametrinen suunnittelu nojaa ennalta määrättyihin sääntöihin ja parametreihin, jotka ohjaavat suunnitteluprosessia. Tämä mahdollistaa suunnittelijoiden luoda monimutkaisia muotoja ja rakenteita, jotka olisivat muutoin olleet vaikeasti saavutettavissa. [1.]

Parametrinen suunnittelu tehostaa mallinnusprosessia antamalla suunnittelijoille mahdollisuuden määrittellä loogisia sääntöjä. Tämä automatisoitu lähestymistapa vähentää manuaalista työtä ja inhimillisen virheen riskiä, mikä parantaa projektien tuottavuutta ja laatua. [1.]

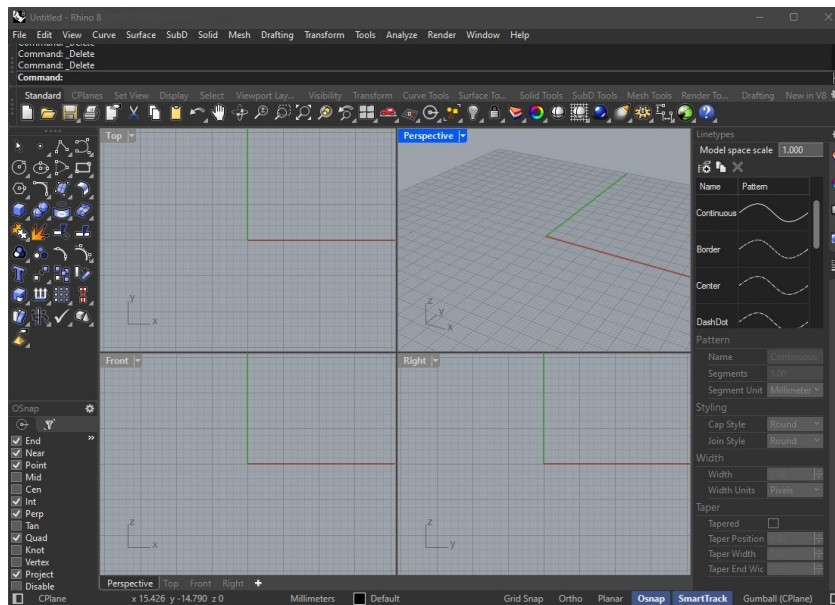
Parametrinen suunnittelu tuo joustavuutta, joka mahdollistaa suunnittelijoiden muutoksien tekemisen nopeasti ilman, että koko mallia tarvitsee uudelleen työstää. Tämä nopeus muutosten hallinnassa on erityisen arvokasta monimutkaisissa projekteissa, joissa vaatimukset voivat nopeasti muuttua. [1.]

Kun koko suunnitelma voidaan luoda ja hallita kokonaisuutena. Suunnitteluprosessi avaa ovet uudelle luovuudelle. Parametrinen suunnittelu mahdollistaa rakenteiden optimointia sekä tehokkaampia ratkaisuja, antaen suunnittelijoille vapauden tutkia erilaisia muotoja ja rakenteita. [1.]

Parametrin suunnittelun avulla voidaan tuottaa korkealaatuisia malleja tehokkaammin. Se soveltuu erinomaisesti niin suuriin kuin pieniinkin projekteihin, mutta on erityisen hyödyllinen monimutkaisissa rakenteissa, joissa sen avulla voidaan optimoida suunnitteluprosessi ja saavuttaa parempia tuloksia. [1.]

2.2 Rhino

Rhinoceros 3D, lyhyemmin Rhino on monipuolinen 3D-mallinnusohjelmisto (Kuva 1), joka yhdistää mallinnusvapauden äärimmäiseen tarkkuuteen. Rhino on suunniteltu tukemaan erittäin tiukkoja toleransseja tuotannossa ja tiedonsiirrossa muihin CAD-ohjelmiin. Sen käyttämä NURBS-geometria takaa mallien tarkan eheyden. [2.]



Kuva 1. Rhino 8 -ohjelmisto.

Rhino erottuu erityisesti kyvyssään luoda ja kuvailla monimutkaisia 3D-muotoja. Sen kehittyneet työkalut antavat käyttäjälle täyden hallinnan muotojen hallintaan ja helpottavat vierekkäisten pintojen yhdistämistä toisiinsa. Mallinnushistoria-toiminto mahdollistaa pintojen säätämisen niin, että viereiset pinnat mukautuvat automaattisesti säilyttäen asetetut arvot. Analyysityökalujen avulla pinnat voidaan tutkia visuaalisesti tai matemaattisesti. Näin varmistetaan esteettisten ja valmistuksen vaatimusten täytyminen. [2.]

Rhinoon on lisätty SubD-työkalusetti (kuva 2), joka on spline-pohjainen. Spline on matemaattinen funktio, joka määritellään palasittain polynomeilla eli spline-käyrä muodostuu useista polynomisista osista, jotka liittyvät toisiinsa ja

muodostavat käyrän. Tämä mahdollistaa orgaanisten muotojen helpon luomisen ja säätämisen, jotka voivat olla haasteellisia perinteisellä NURBS-geometrialla. [2.]



Kuva 2. SubD-työkalut Rhinossa.

Rhino on myös erinomainen ohjelmisto suoraviivaisen perusgeometrian mallintamiseen sekä tarjoaa monipuolisen mallinnustyökaluvalikoiman. Uusin Rhino 8 versio lisäsi Push-Pull muokkauksen. Tämä ominaisuus mahdollistaa pintojen, reunojen ja muiden geometrinen elementtien helposti muokkaamisen vetämällä tai työntämällä niitä. Auto C-Plane toiminnon. C-Plane (Construction Plane) tarkoittaa rakennustasoa tai työskentelytasoa, jolla käyttäjä työskentelee. Auto C-Plane-toiminto vaihtaa automaattisesti aktiivisen rakennustason valitun pinnan suuntaiseksi. Enemmän Gumball muokkausominaisuuksia. Gumball on visuaalinen työkalu 3D-ohjelmistoissa, joka tarjoaa interaktiivisen ja intuitiivisen tavan siirtää, kiertää tai skaalata valittuja kohteita. Nämä ominaisuudet yksinkertaistavat ja nopeuttavat kiinteiden osien muokausprosessia. [2.]

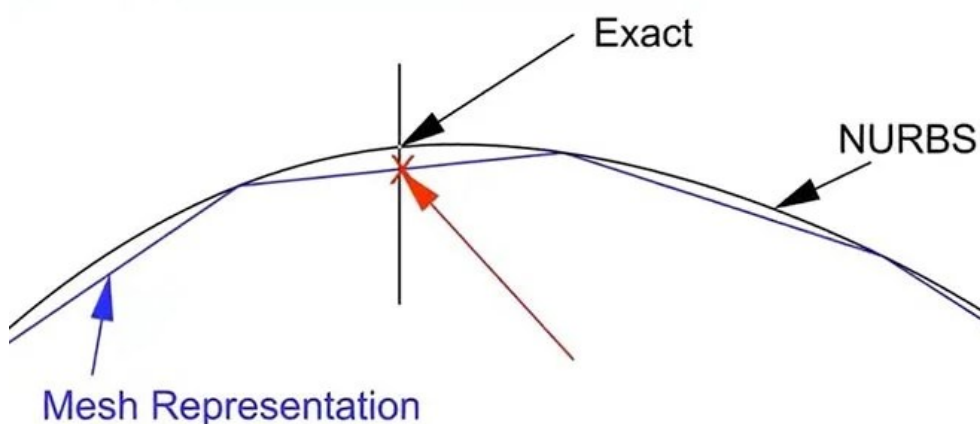
Rhino sisältää myös käytännöllisiä työkaluja, jotka mahdollistavat käyttökelpoisten mallien luomisen heikkolaatuisesta geometriasta. Tällaisia työkaluja ovat esimerkiksi QuadRemesh. Tämä on työkalu, jota käytetään 3D-geometrian uudelleenverkottamiseen, jolloin kohteesta syntyy nelisivuisista (quad) polygoneista koostuva verkkomainen malli. Ja ShrinkWrap. Tämä on ominaisuus tai työkalu, joka luo uuden verkkopinnan olemassa olevan 3D-mallin ympärille siten, että uusi verkkopinta kutistuu alkuperäisen mallin muotoon. [2.]

Rhino on yhteensopiva satojen erilaisten CAD-, CAM-, CAE-, renderointi- ja animaatio-ohjelmistojen kanssa. OpenNURBS-kirjastot mahdollistavat sen, että

muut sovellukset voivat lukea ja kirjoittaa Rhinon omia 3DM-tiedostoja. Tämä mahdollistaa laajan yhteistyön ja tiedonvaihdon eri suunnittelu- ja simulointiohjelmien välillä, tehostaen monialaista projektityöskentelyä. [3.]

NURBS, eli Non-Uniform Rational B-Splines, on matemaattinen esitysmuoto käyriä ja pintoja varten. Tämä tekniikka on olennainen osa Rhino-ohjelmiston kehitystä, sillä se mahdollistaa monimutkaisten muotojen ja pintojen luonnin suurella tarkkuudella. Nämä käyrät ja pinnat määritellään joukolla kontrollipisteitä, painoarvoja ja solmuvektoreita. Kontrollipisteet ohjaavat käyrän tai pinnan yleistä muotoa, kun taas painoarvojen avulla voidaan hienosäätää jokaisen kontrollipisteen vaikutusta muodon määräytymisessä. [4.]

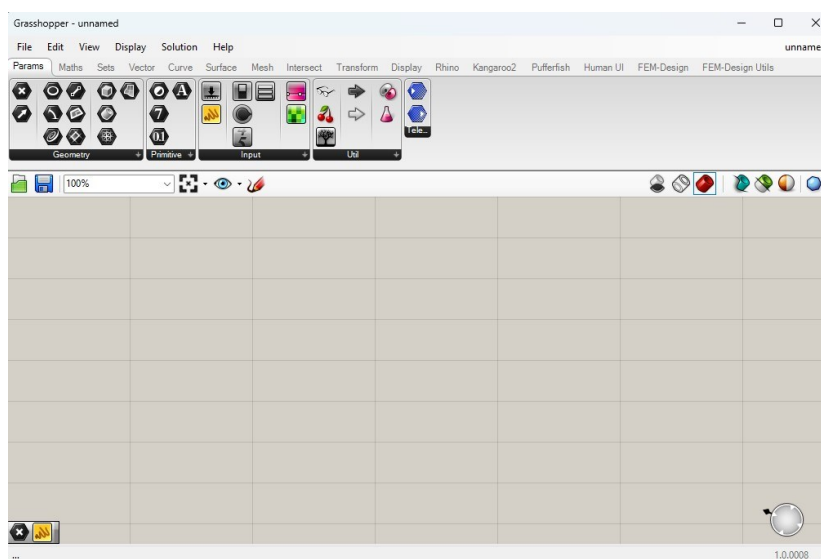
Tämä joustavuus hallintapisteiden ja painotusten avulla mahdollistaa tarkan kontrollin käyrän tai pinnan muotoilussa, mikä on teollisuudenaloilla, kuten auto-teollisuudessa, teollisessa muotoilussa ja arkkitehtuurissa, erittäin arvokasta. Kuvassa 3 nähdään, että Mesh-geometria näyttää läheltä katsottuna sahalaitaiselta ja epätasaiselta, kun NURBS-geometria tarjoaa sileän ja jatkuvan pinnan. Mesh-geometria on myös vaikeampi muokata ja hallita, mutta toisaalta verkko-geometrian käyttö johtaa yleensä pienempiin tiedostokokoihin. NURBS-muodot ovat interaktiivisia ja niitä voidaan muokata reaaliajassa, mikä lisää niiden soveltuvuutta monimutkaisten muotojen suunnitteluun, aina yksinkertaisista kaarista monimutkaisiin kaksoiskäyriin pintoihin. [4.]



Kuva 3. Ero Mesh ja NURBS -geometrioiden välillä [3].

2.3 Grasshopper

Grasshopper on visuaalinen ohjelmointiympäristö (kuva 4), joka on integroitu osana Rhino-ohjelmistoa ja se sisältyy suoraan ohjelman asennuspakettiin, joten erillistä latausta tai asennusta ei tarvita. Grasshopper ei vaadi ohjelmointi- tai koodaustaitoja. Ohjelma tarjoaa intuitiivisen käyttöliittymän, jossa käyttäjät voivat rakentaa algoritmeja visuaalisesti liittämällä erilaisia komponentteja toisiinsa, mikä mahdollistaa monimutkaisten muotojen ja rakenteiden luomisen ilman perinteistä koodausta. [4.]

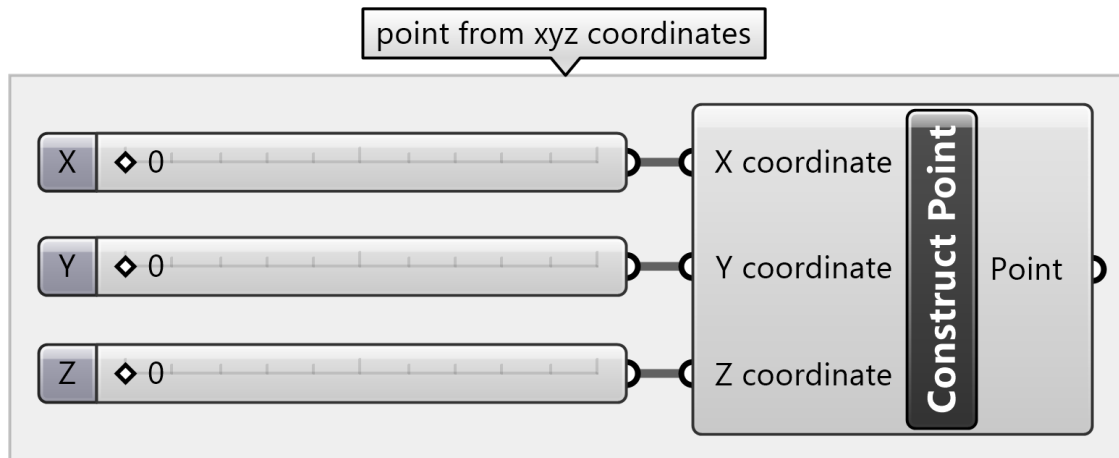


Kuva 4. Grasshopper ympäristö.

Grasshopper tukee myös monipuolista lisäosakirjastoa, joka laajentaa sen toiminnallisuutta ja soveltuvuutta erilaisiin suunnittelu- ja analyysitehtäviin. Nämä lisäosat tarjoavat työkaluja muun muassa simulaatioihin, rakenneanalyysiin, ympäristöanalyysiin ja digitaaliseen valmistukseen, tehden Grasshopperista monipuolisen työkalun eri suunnittelualoille. [5.]

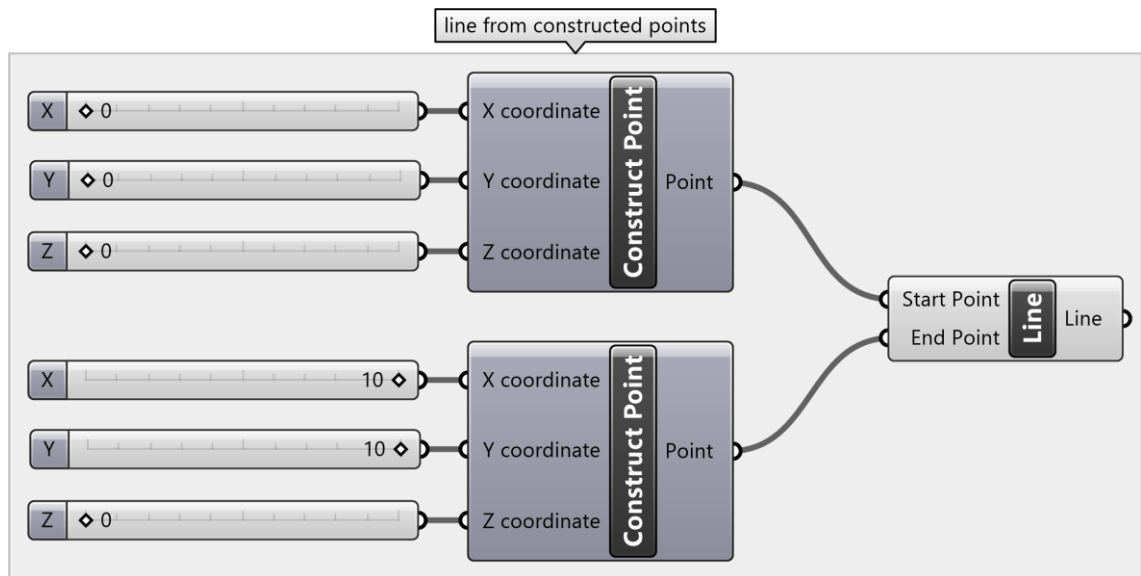
Grasshopperia käytetään geometrinen objektien, kuten pisteiden ja viivojen, luomiseen ja muokkaamiseen. Grasshopperissa pisteitä luodaan käyttäen x-, y- ja z-koordinaatteja. Kuvassa 5 pisteiden määrittelyyn käytetään Construct Point komponenttia, joka löytyy Vektori-välilehden Pistepaneelistä. Asettamalla tämä komponentti kankaalle haluttuun kohtaan ja lisäämällä syötetieto x-, y- ja z-

parametreille käyttämällä Number slider -komponentteja, jotka löytyvät Parametrit-välilehden Syötepaneelistä. Yhdistämällä jokaisen Number slider oikeanpuoleisen pisteen lähtö vastaavaan pisteeseen Construct Point komponentin vasemmalla puolella. Säättämällä Number slider komponentin kahvaa asetetaan x-, y- ja z-arvot pisteelle tai kirjoittamalla Number slider komponenttiin haluttu arvo. [6.]



Kuva 5. Number slider ja Construct point komponentit. [6.]

Grasshopperissa viivat voidaan määritellä lähtö- ja päätepisteillä, lähtöpisteellä, suunnalla ja pituudella. Kuvassa 6 lähtö- ja päätepisteet voidaan asettaa luomalla pisteitä Number slider ja Construct point -komponentteja käyttäen, tekemällä kaksi eri kohdassa olevaa pistettä ja yhdistämällä nämä Line-komponentin start point ja end point -kohtiin. [6.]



Kuva 6. Viivan tekeminen Line komponentilla. [6.]

2.4 Tekla Structures

Tekla Structures on monipuolinen rakennesuunnittelun BIM (Building Information Modelling) ohjelmisto, joka mahdollistaa rakennesuunnittelijoiden, valmistajien, urakoitsijoiden ja projektinjohtajien luoda, yhdistää, hallita ja jakaa tarkkoja 3D-malleja projekteihin. Ohjelmisto tukee monipuolisten materiaalien käyttöä ja edistää yhteistyötä, päätöksentekoa sekä auttaa yrityksiä saavuttamaan poikkeuksellisia tuloksia. Tekla Structuresia voidaan käyttää koko projektin ajan, aina rakennusten ja infrastruktuurin suunnittelusta valmistukseen, rakentamiseen ja ylläpitoon. Ohjelmisto tukee suunnittelua, yksityiskohtaista mallintamista ja tietojen hallintaa. [7.]

Tekla Structures on nopea, luotettava ja mullistava ohjelmisto, joka mahdollistaa tarkat 3D-mallit ilman komplikaatioita. Sen avulla rakennusalan ammattilaiset voivat ylittää perinteiset rajoitukset rakennusprojektin jokaisessa vaiheessa. Ohjelmisto mahdollistaa tietojen luomisen, yhdistämisen, hallinnan ja jakamisen tehokkaasti. [7.]

2.5 Grasshopper-Tekla Live Link

Grasshopper-Tekla Live Link mahdollistaa Grasshopperin ja Tekla Structuren yhdistämisen. Grasshopper-Tekla Live Link antaa kokoelman Grasshopper-komponentteja (kuva 7), jotka pystyvät luomaan ja vuorovaikuttamaan Tekla Structures -ohjelmasta löytyvien kohteiden kanssa reaaliajassa. Näin voidaan tehdä algoritmista mallinnusta Tekla Structures -ohjelmistoon käyttäen Rhino/Grasshopper ympäristöä. Tämä integraatio tarjoaa ainutlaatuisen kyvyn suunnittelijoille ja insinööreille mallintaa rakenteita ja rakennuselementtejä visuaalisesti koodaamalla Grasshopperissa, ja nähdä välittömästi tulokset Tekla Structures -ohjelmassa. Näin yhdistetään Grasshopper/Rhino ohjelmiston geometrinen joustavuus ja Tekla Structuren rakennetekninen tarkkuus, mahdollistaen monimutkaisten rakenteiden ja niiden osien suunnittelun ja optimoinnin. [8.]



Kuva 7. Tekla-osion Grasshopper-komponentit.

3 Grasshopperin käyttö rakennesuunnittelussa

3.1 Parametrisen algoritmin luominen ja muokkaaminen

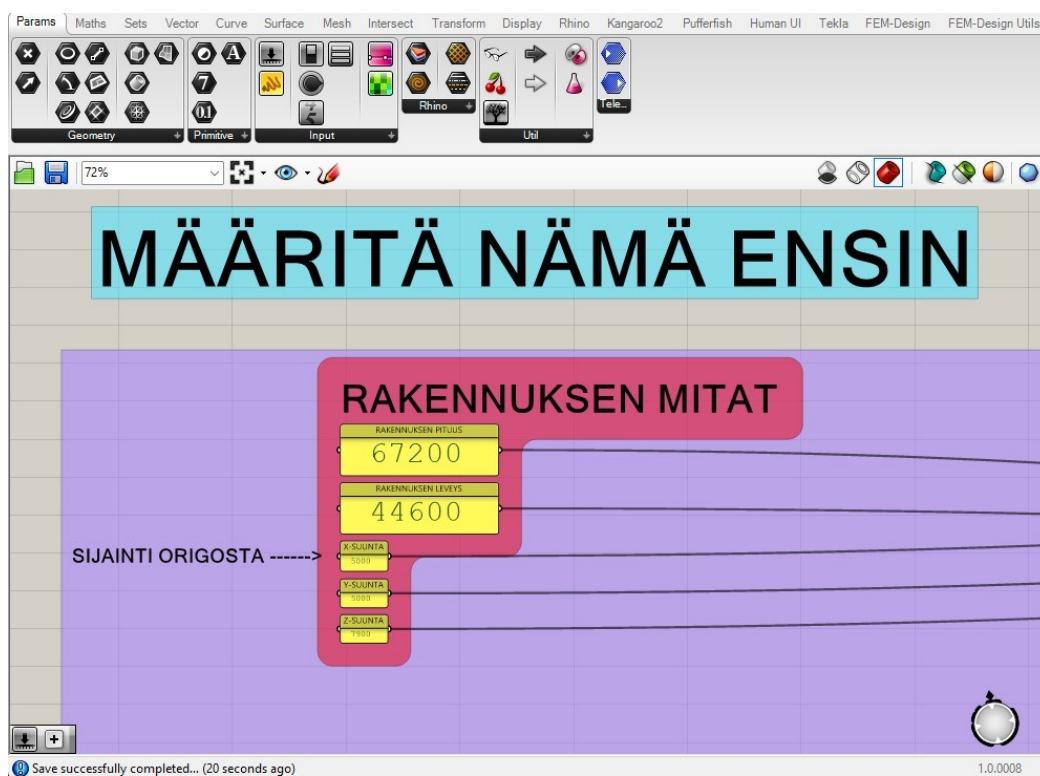
Algoritmpohjan tekemiseen on otettu malliksi yrityksen aiemmin toteuttama halliprojekti, jonka avulla saatiin hallin rakenteiden todellisia mittoja ja sijainteja. Tämä aikaisempi projekti auttaa luomaan loogisen ja toimivan järjestyksen algoritmpohjaan. Tämän lähestymistavan ansiosta voidaan varmistaa, että Grasshopperissa tehdyt komponentit kuten viivat ja pisteet asettuvat tarkasti ja oikeille kohdilleen Rhinon 3D-ympäristöön.

Parametrisen algoritmin kehityksessä ensimmäisenä arvioitiin, mitkä rakennuksen osat halutaan määrittää muuttuviksi parametreiksi. Pohdimme, voiko rakennuksen muodon tehdä joustavaksi, jolloin se voisi sisältää kaarteita ja muunnelmia, vai rajoittuisiko se yhteen perusmuotoon, jonka kokoa voitaisiin muuttaa suuremmaksi tai pienemmäksi. Tarkastelun jälkeen päätettiin, että yksi perusmuoto riittää ja hallin pituuden, leveyden ja korkeuden tulisi olla säädettävissä. Algoritmin tavoitteena oli luoda sellainen parametrinen rakennusmallin pohja, joka mahdollistaisi kaksilaivaisen rungon sekä rakennuksen keskeisten rakenteiden: pilareiden, palkkien, kattopalkkien, anturoiden, sokkelin, alapohjan laatan ja seinäelementtien mallintamisen ja gridit Tekla Structureen.

Eryteisesti haluttiin hallin kummankin pään reunimmaisiet moduulien välit muokattavaksi itsenäisesti, kun taas hallin pituus ja leveyssuunnassa sivujen keskellä olevien moduulien välimatka mukautuu yhden annetun mitan mukaan. Selkeyden ja käytettävyyden parantamiseksi päätimme rajoittaa parametrien määrää. Tavoitteenamme oli estää, että suunnittelija kohtaisi liiallisen määrän säädettäviä parametreja, mikä voisi johtaa sekavuuteen mittatietoja syötettäessä. Tällä ratkaisulla varmistetaan, että algoritmi on selkeä ja helppokäyttöinen, samalla kun se mahdollistaa joustavuuden rakennuksen mitoituksessa.

3.2 Ensimmäiset askeleet Grasshopperissa

Grasshopperissa aloitetaan asettamalla kuvassa 8 hallirakennukselle pituus ja leveys sekä sijainti origosta, jotka on arkkitehti määrittellyt. Tämä vaihe on algoritmin perusta, sillä nämä määrittävät koko rakennuksen parametriset rajoitukset. Tarkat mittatiedot eivät ole välttämättömiä sillä näitä voi myöhemminkin muuttaa. Algoritmi ankkuroituu näihin annettuihin mittoihin ja luo sekä skaalaa jatkossa luotavat geometriset muodot.

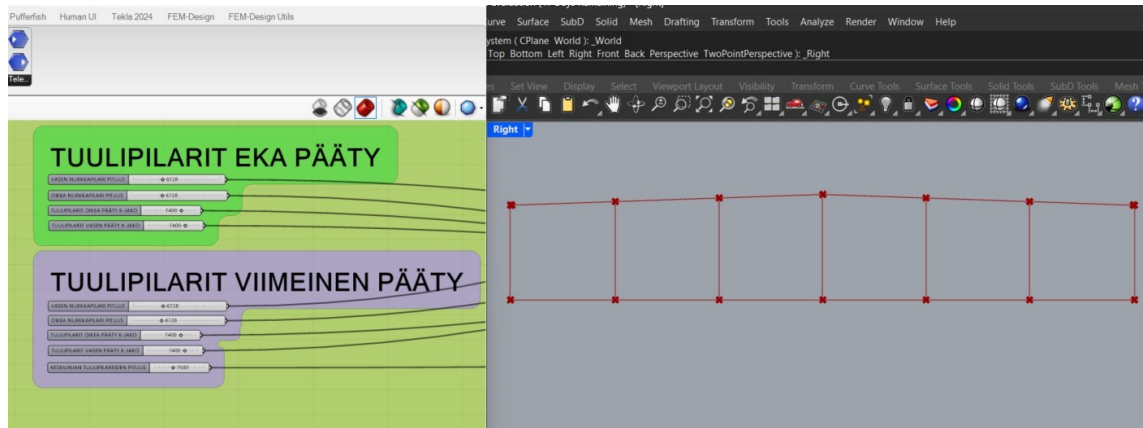


Kuva 8. Rakennukset mitat asetettuna Grasshopperissa.

3.3 Tuulipilarit ja nurkkapilarit

Kuvassa 9 määritetään hallirakennuksen päätyjen pilareiden pituudet ja moduulivälit. Kun asetamme nurkkapilareiden ja keskilinjalla sijaitsevan pilarin pituudet, Grasshopper-ohjelma luo algoritmin avulla automaattisesti välissä olevien tuulipilarien oikeat mitat. Ohjelma ottaa huomioon mahdolliset pituserot

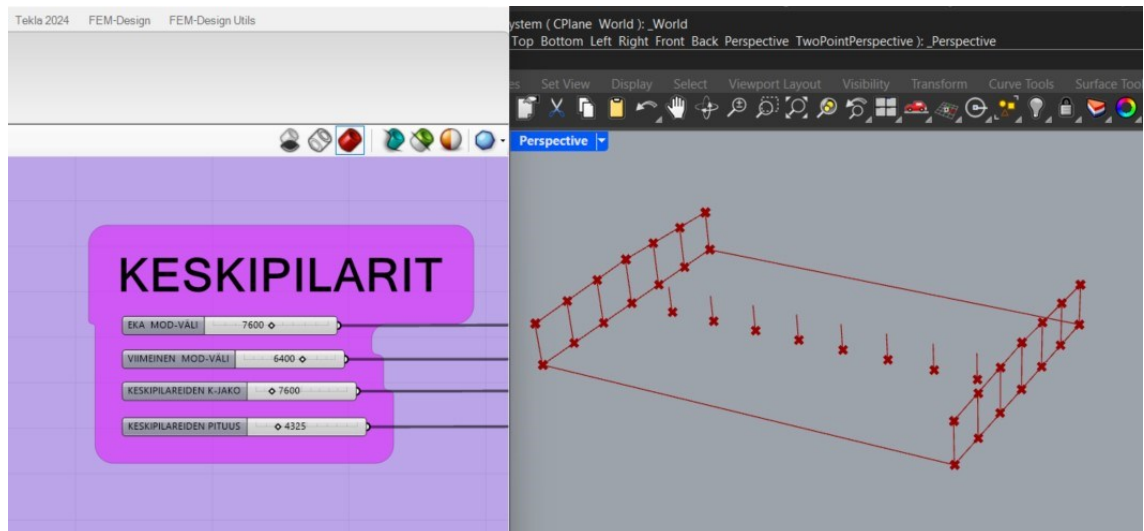
nurkkapilarin ja keskilinjan pilarin välillä, säätäen tuulipilarien pituuksia vastamaan tätä kulmaa, jolloin koko pilaririvistö asettuu tarkasti suunnitellun mukaisesti.



Kuva 9. Grasshopper mitat ja Rhino rakennuksen pääty.

3.4 Keskipilarit

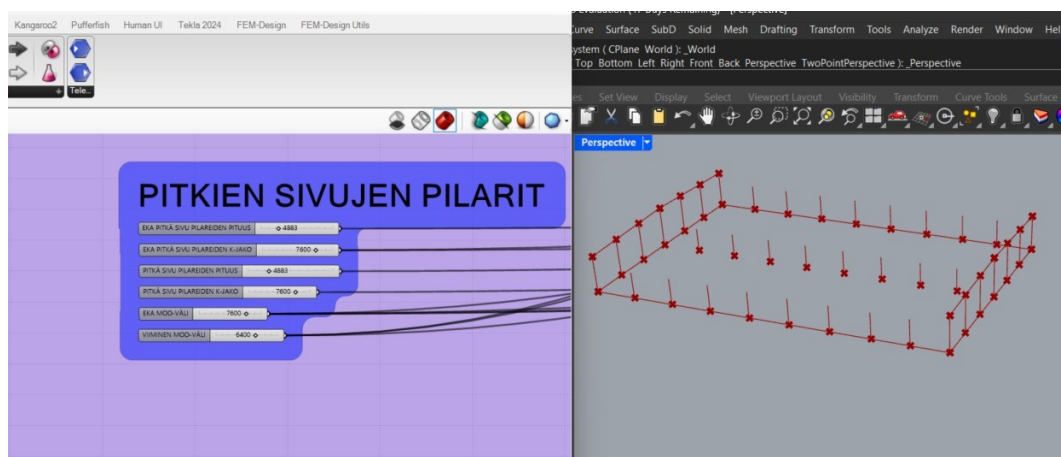
Kuvassa 10 näkyy keskipilareiden määrittely Grasshopper-algoritmissa, missä säädetään ensimmäisen ja viimeisen moduulien välit sekä keskipilareiden jakoväli ja pituus. Keskipilarit jakautuvat tasaisesti koko hallin pituuteen päädyt huomioiden. Tasainen välitys sen takia, jotta parametreja ei tulisi liikaa ja algoritmi pysyisi siistinä sekä yleensä pilarit ovat tasajaolla.



Kuva 10. Grasshopper-parametrit ja Rhinoon lisätty rakennuksen keskipilarit.

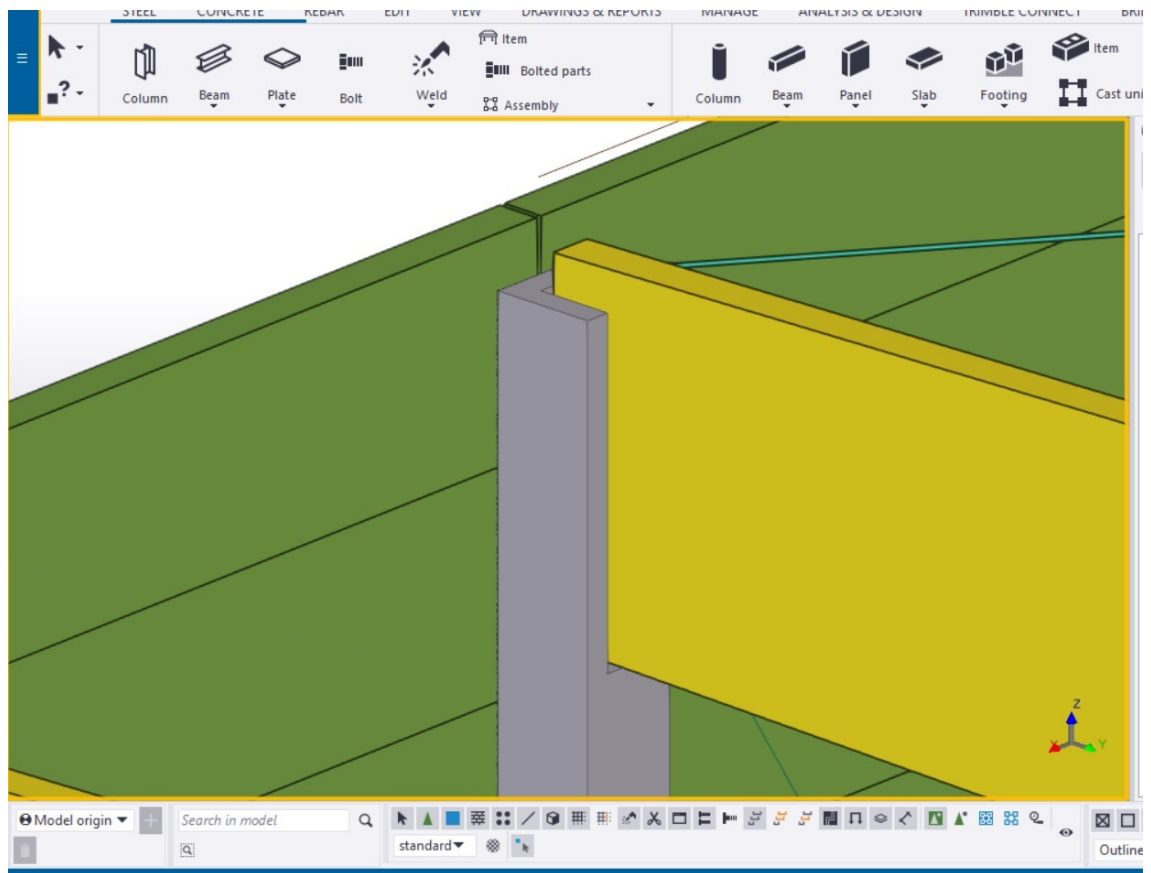
3.5 Pääpilarit

Hallin pääpilarit asetetaan kuvassa 11 Grasshopperin algoritmilla. Parametreilla määritellään ensin pääpilarien korkeudet ja tämän jälkeen moduulivälit, jotka määräävät pilaririvistön jakautumisen hallin pitkille sivuille. Ensimmäinen ja viimeinen moduuliväli on säädettävissä erikseen omilla parametreilla. Jäljelle jääneet pilarit määritetään yhdellä parametrilla, jolle kaikille tulee yhtä suuri etäisyys toisistaan.



Kuva 11. Grasshopper-parametrit ja Rhinoon lisätty rakennuksen pääpilarit.

Algoritmin avulla on mahdollista muuttaa pilarityypit vaatimuksien mukaiseksi. Materiaaliksi voidaan valita puu, betoni tai teräs. Kun tyyppiä valitaan betonipilari, on otettava huomioon sen ja liimapuupalkin liitos. Kuvassa 12 betonipilarin ja liimapuupalkin liitos, joka eroaa merkittävästi puupilarille suunnitelluista liitoksista. Algoritmi tarjoaa joustavuuden suunnitella ja toteuttaa erilaisia liitoksia ottaen huomioon pilarimateriaalin valinnan. Tämä tarkoittaa, että käyttäjä voi säätää algoritmin parametreja niin, että se määrittää liitostyyppin betonipilarin ja liimapuupalkin välille.



Kuva 12. Betonipilarin ja puupalkin liitos.

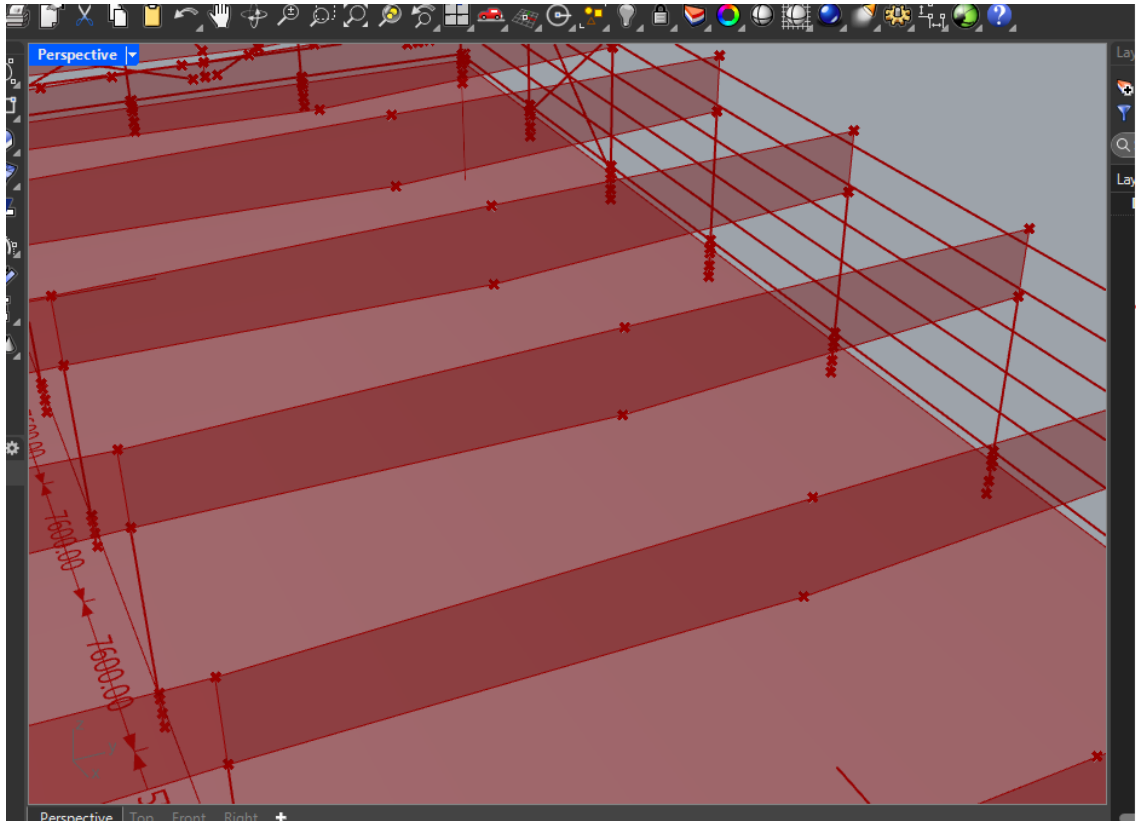
3.6 Liimapuupalkki

Liimapuupalkkien määrittäminen Grasshopperissa aloitetaan asettamalla liukusäätimiin (kuva 13.) halutut mitat. Käyttäjä voi säätää palkin pituutta, etäisyyttä ja korkeutta helposti liikuttamalla säädintä tai kirjoittamalla säätimeen haluttu mitta. Liimapuupalkin molempien päiden tukikorkeuksia sekä keskikorkeus on säädettävissä. Liukusäätimiin on asetettu minimi- ja maksimiarvot, jotta niihin ei voi syöttää algoritmin hajottavia arvoja. Liukusäätimistä lähtevät viivat yhdistyvät algoritmin seuraavaan osioon.



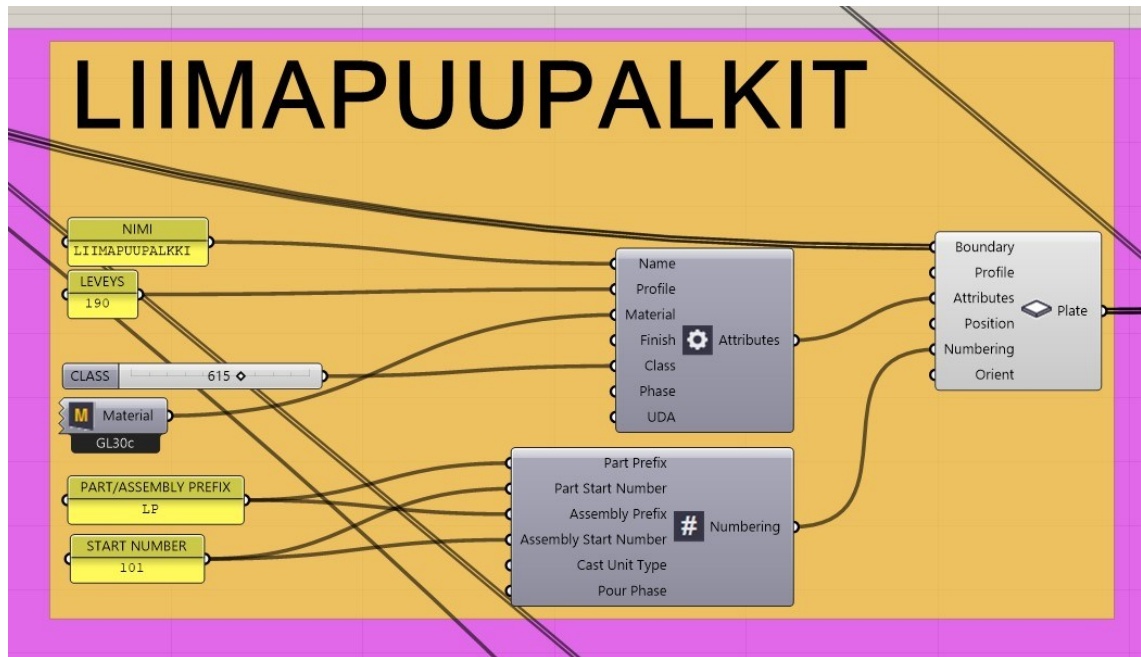
Kuva 13. Grasshopperin liukusäätimet mitoille.

Liimapuupalkkien geometria on visualisoitu Rhino-ohjelmistossa, jonka avulla päästään käsitykseen palkkien koosta. Kuvassa 14 näkyy parametrien avulla generoitu liimapuupalkin muoto. Tässä nähdään reaaliajassa, mitä liukusäätimen arvon muuttaminen tekee.



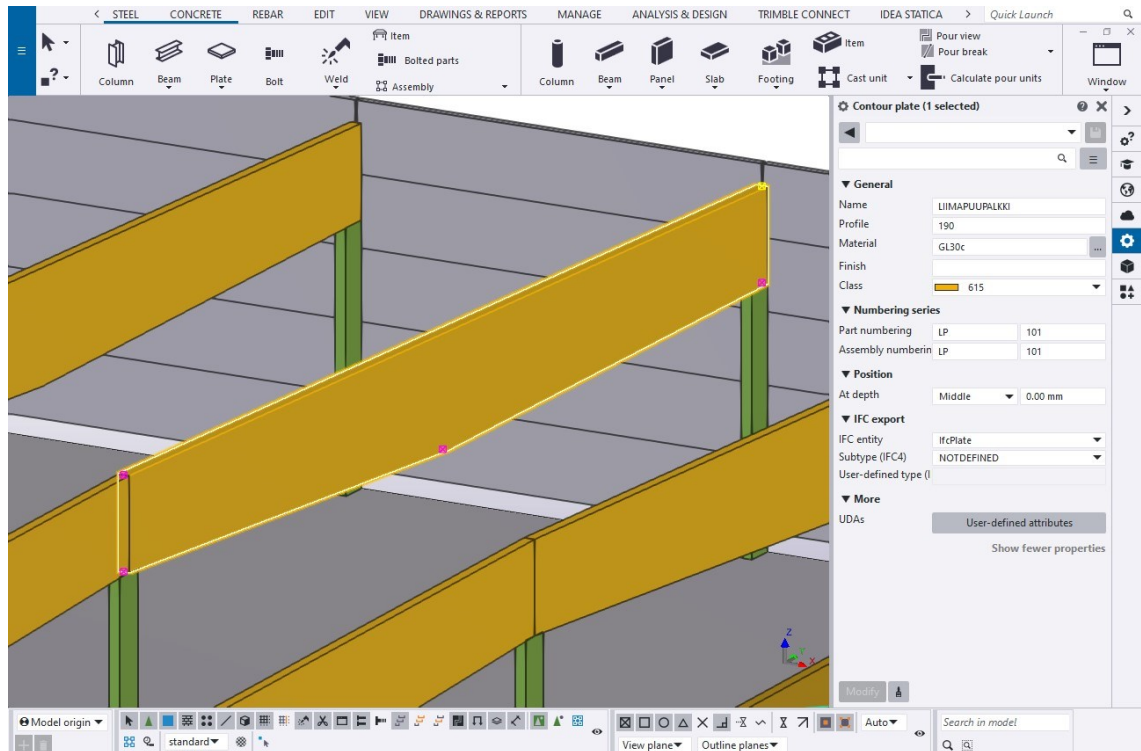
Kuva 14. Rhinossa liimapuupalkkien geometria.

Grasshopperissa tehdystä algoritmista on saatu geometria, joka yhdistetään Tekla Live Linkistä saatujen komponenttien avulla (kuva 15.) Tekla Structures -mallinnusohjelmistoon. Grasshopperissa määritellään liimapuupalkin attribuutit, kuten nimi, paksuus, class ja materiaali. Numbering-kohtaan määritellään myös halutut tiedot. Yleensä näitä tietoja ei tarvitse muuttaa paitsi esimerkiksi profiilia voidaan juotua hieman muuttamaan, tässä onkin laitettu jokainen kohta muutettavaksi, jos tarve tulee.



Kuva 15. Tekla-komponentit Grasshopperissa

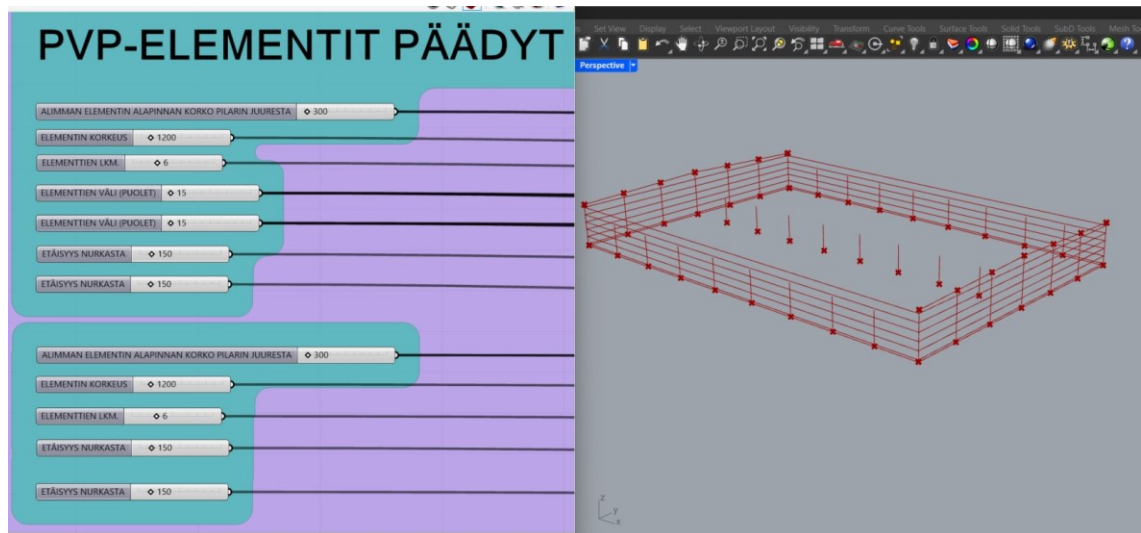
Kuvassa 16 liimapuupalkki on mallinnettu Tekla Structures -ohjelmistoon, johon tiedot on tuotu suoraan Grasshopperin algoritmin kautta käyttämällä Tekla Live Link toimintoa. Näin saadaan Teklaan jokaiselle liimapuupalkille tarkat mitat ja tiedot. Tämä prosessi nopeuttaa suunnittelutyötä ja parantaa tietojen sekä rakenteiden yksityiskohtien tarkkuutta.



Kuva 16. Liimapuupalkki mallinnettuna Tekla Structures -ohjelmistossa.

3.7 Ulkoseinäelementit

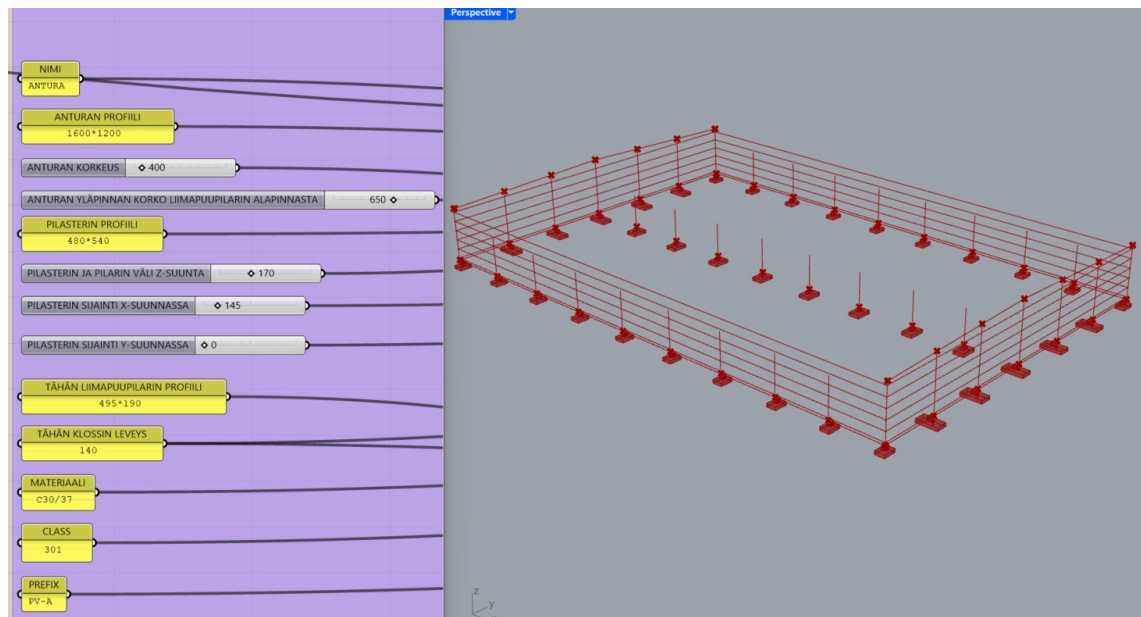
Kuvassa 17 esitetään prosessi, jossa Grasshopperin avulla määritetään hallirakennuksen ulkoseinäelementit. Parametriin syötetään elementtien tärkeimmät mitat: alimman elementin alapinnan korkeus pilarin juuresta, itse elementin korkeus, elementtien lukumäärä sekä etäisyydet nurkista. Nämä mitat varmistavat, että jokainen elementti asettuu oikeaan kohtaan rakenteessa, ja että elementtien kokonaisuus muodostaa suunnitellun ulkoseinän. ulkoseinäelementtien pituus tulee automaattisesti moduulivälin mukaan. Grasshopperissa voidaan myös määrittää haluttu pystysauma väli. Tässä projektissa määritettiin väliksi 30 mm. Tässä algoritmista ei ole mahdollista tehdä seinään aukkoja. Aukot täytyy määrittää Teklassa manuaalisesti.



Kuva 17. Grasshopper-parametrit ja Rhinoon lisätty rakennuksen seinäelementit.

3.8 Anturat

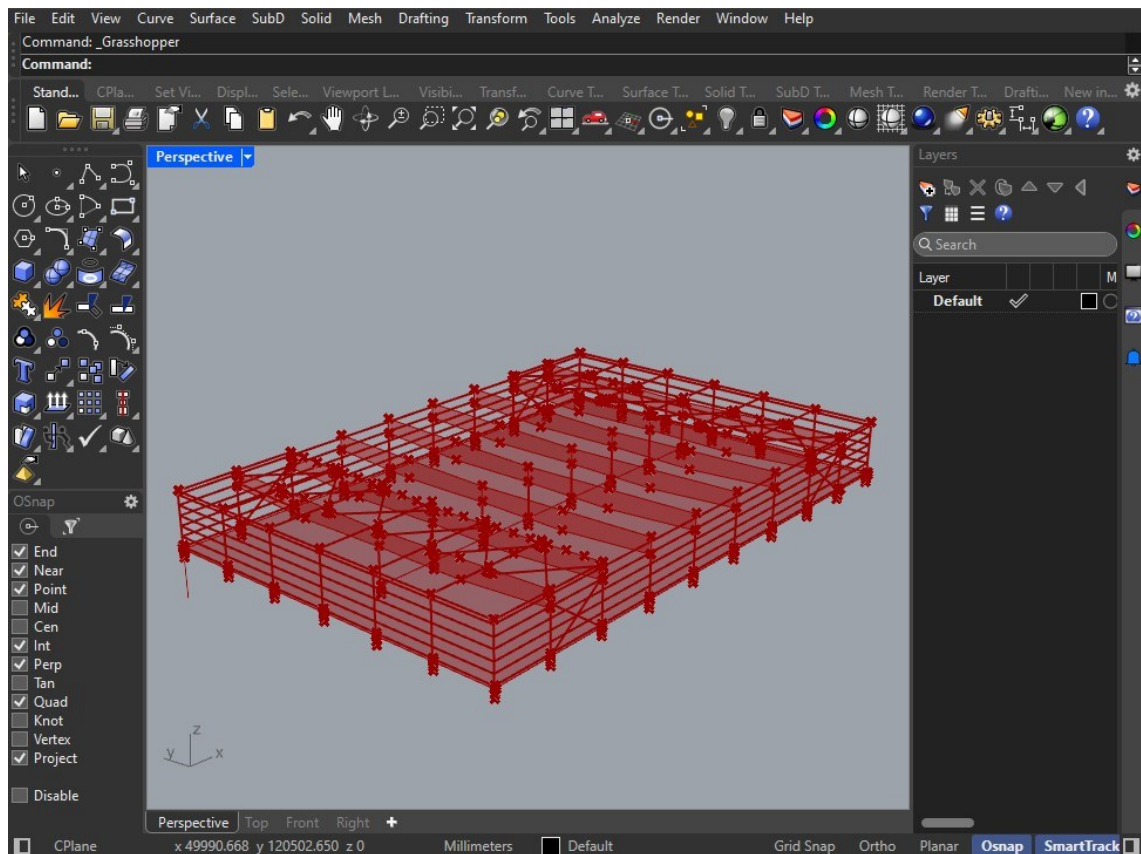
Kuvassa 18 asetetaan Grasshopper-algoritmin avulla anturoiden ja pilasterien tärkeimmät parametrit. Määritetään anturan ja pilasterin profiili sekä korkeus, niiden välinen etäisyys pilarin alareunasta Z-suunnassa, sijainti X- ja Y-suunnassa. Lisäksi betonin lujuusluokka ja class sekä prefix -arvot voidaan syöttää Grasshopperissa. Algoritmissa voidaan määrittää tarvittava tila pilarikenkää varten. Algoritmi hyödyntää syötettyjä tietoja, jotta pilasterit ja anturat voidaan mallintaa tarkasti ja osaksi kokonaisrakennetta.



Kuva 18. Grasshopper-parametrit ja Rhinon lisätty rakennuksen anturat

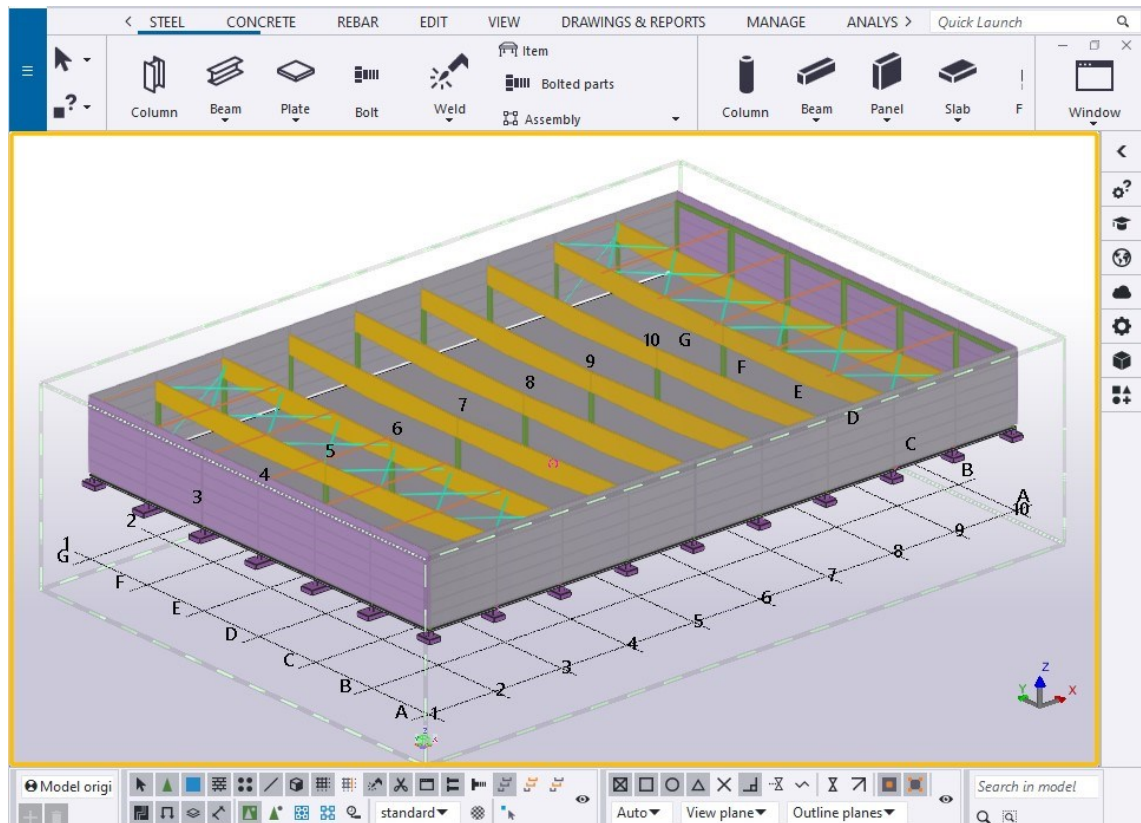
3.9 Koko halli

Grasshopperissa tehdyn algoritmin tuloksena kuva 19 esittää koko hallirakennuksen mallia Rhino 8 -ohjelmistossa. Visualisointi Rhinon 3D-ympäristössä antaa hyvän kuvan algoritmin toiminnasta näyttäen, miten viivat ja pisteet liittyvät toisiinsa. Ilman Rhino-näkymää olisi vaikea hahmottaa algoritmin toimivuutta, kun pelkässä Grasshopperissa ei näe visuaalisesti, mitä se tuottaa.



Kuva 19. Hallirakennus Rhino 8 -ohjelmassa.

Kuvassa 20 sama hallirakennus on Grasshopperin algoritmin avulla mallinnettu automaattisesti Tekla Structures -ohjelmistoon. Mallinnusprosessi mahdollistaa rakenteiden, kuten palkkien ja pilareiden tarkat paikat sekä niiden suhteiden visualisoinnin. Tässä Teklan näkymässä rakennuksen jokainen elementti on esitetty tarkkaan kolmiulotteisesti antaen selkeän kuvan kaikkien rakenteiden toimivuudesta. Teklassa voidaan vielä tarkemmin tarkastella, että kaikki objektit ovat mallintuneet oikealla tavalla.



Kuva 20. Halli Rakennus Tekla Structures -ohjelmassa.

4 Havaitut ongelmat

Grasshopper-ohjelmiston kanssa havaittiin erityinen ongelma liittyen anturoiden mallintamiseen. Kun anturan paikkaa mallinnetaan Grasshopperissa käyttämällä viiva komponenttia, viivan suunnalla havaittiin olevan vaikutusta lopputulokseen. Viivan suunta vaikuttaa, tunnistaako Grasshopperin "Pad Footing" komponentti komennon oikein. Jos viiva vedetään ylhäältä alas, Tekla Structures -ohjelmistossa syntyy virheellisesti teräspalkin elementti anturan sijaan. Tämän ongelman voi kuitenkin korjata kääntämällä viivan suuntaa niin, että se vedetään alhaalta ylös. Tällöin Tekla Structures luo odotetusti anturan. Tämä havainto on tärkeä muistaa suunnitteluprosessissa, ja se korostaa parametrisen suunnittelun tarkkuuden ja yksityiskohtien merkitystä.

Jos Tekla Structures -ohjelmistossa tehdään muutoksia objekteihin mitkä ovat luotu Grasshopperilla, niin seuraavan kerran Grasshopper-algoritmia avattaessa nämä manuaalisesti tehdyt muutokset korvaantuvat Grasshopperin algoritmin määrittelemillä arvoilla. Tämä ominaisuus vaatii erityistä huomiota suunnitteluprosessissa, sillä työvaiheilla on suuri merkitys suunnitelmien hallinnassa pysymiseen.

5 Haasteet

Tämän opinnäytetyön osalta suurimmat haasteet olivat Rhino- ja Grasshopper-ohjelmistojen aiemman käyttökokemuksen puute. Grasshopperin algoritmisen suunnittelun periaatteiden ja toimintalogiikan ymmärtäminen edellytti merkittävää opiskelua ja harjoittelua. Algoritmipohjan rakentaminen Grasshopperilla oli aluksi vaikea prosessi, joka vaati jatkuvaa opiskelua ja kokeilemista. Vaikeuksista huolimatta oppimiskokemus oli palkitseva, ja se tarjosi mahdollisuuden kehittyä jatkuvasti sekä ymmärtää syvällisemmin ohjelmiston tarjoamat mahdollisuudet.

Opinnäytetyön alkuvaiheissa ilmeni useita haasteita liittyen Grasshopper-komponenttien käyttöön. Kokemattomuuden vuoksi algoritmeihin sisällytettiin runsaasti ylimääräisiä komponentteja, mikä johti tarpeettomaan monimutkaisuuteen ja tehottomuuteen. Ajan kuluessa taidot ja ymmärrys Grasshopperista paranivat, minkä myötä ylimääräisten komponenttien tunnistaminen ja poistaminen onnistui paremmin. Turhien komponenttien poisto tekee algoritmista selkeämmän ja tehokkaamman.

6 Tulokset ja jatkokehitys

Tämän opinnäytetyön tuloksena syntyi käyttökelpoinen Grasshopper-algoritmi, joka tehostaa mallinnusta ja rakennesuunnittelua. Algoritmin avulla voidaan automatisoida monia suunnitteluprosessin osa-alueita, vähentäen manuaalisen työn määrää ja parantaen suunnittelun tarkkuutta. Ylimääräisten komponenttien poistaminen algoritmista johti selkeämpään ja virtaviivaisempaan työnkulkuun.

Algoritmi mahdollisti erilaisten rakennusosien, kuten pilarien, palkkien, seinäelementtien ja anturoiden tekemisen sekä mukauttamisen Tekla Structureen. Mallinnuksen joustavuus ja sopeutumiskyky samantyyppisiin projekteihin osoitti algoritmisen suunnittelun potentiaalin ja tämä lähestymistapa todisti kykynsä tuottaa tarkkoja, teknisiä ja helposti muokattavia malleja.

Lisäksi työn tulokset osoittivat, että Grasshopperin käyttö Tekla Structures -ohjelmiston kanssa luo voimakkaan työkalun parametriseen suunnitteluun. Tämä avaa uusia mahdollisuuksia rakennusprojektien hallinnassa ja tuo mukanaan merkittäviä hyötyjä suunnittelun ja toteutuksen eri vaiheissa. Opinnäytetyön tulokset vahvistavat algoritmisen suunnittelun merkittävää roolia rakennusalan tulevaisuudessa ja sen kykyä parantaa suunnittelun laatua ja tehokkuutta.

Jatkokehityksenä mietimme mahdollisuuksia laajentaa algoritmin soveltuvuutta erilaisiin hallirakenteisiin. Tässä työssä keskityttiin 2-laivaisen hallirakennuksen parametrisointiin, mutta tulevaisuudessa olisi hyödyllistä soveltaa algoritmia myös muihin muotoihin, esimerkiksi 1- ja 3-laivaisiin. Lisäksi algoritmin muokkaaminen niin, että se mahdollistaisi pilarien moduuli välejä säätämään laajemmin eikä vain rakennuksen päädyissä. Havaintomme osoittavat, että moduulivälien muokkaaminen koko rakennuksen alueella mahdollistaisi monipuolisemman ja toimivamman algoritmin. Tämä voisi parantaa suunnittelun joustavuutta ja tehokkuutta sekä manuaalisen muokkauksen tarvetta.

Algoritmi pohjan teko, joka mahdollistaa piirustusten luonnin Tekla Structuren piirustuspuolella. Tällä algoritmilla saisi automatisoitua piirustusten tuottamista, joka vähentäisi manuaalisesti aikaa vievän työn teon.

7 Yhteenveto

Tässä insinööriyössä pyrittiin tutkimaan parametrisen suunnittelun soveltuvuutta rakennesuunnittelussa ja luomaan sekä kehittämään Grasshopper-ohjelmaan perustuvaa algoritmia, joka toimii Tekla Structures -ohjelmiston kanssa. Työn keskeinen tavoite oli luoda algoritmi, joka mahdollistaa hallirakennuksen rakenteiden nopean mallintamisen ja mukauttamisen, parantaen näin suunnitteluprosessin tehokkuutta ja joustavuutta.

Työn aikana suunniteltiin ja toteutettiin algoritmi, joka hyödyntää parametrisia muuttujia hallirakennuksen rakenteiden mallintamiseen. Työn tuloksena luotiin käyttäjäystävällinen algoritmi, joka mahdollistaa rakenteiden nopean ja tarkan määrittelyn Tekla Structures -ohjelmistoon. Työssä havaittiin, että algoritmipohjainen suunnittelu nopeuttaa rakenteiden mallinnusta ja muutosten toteutusta sekä vähentää inhimillisten virheiden määrää, koska muutokset päivittyvät automaattisesti kaikkiin liittyviin rakenteisiin.

Vaikka työn tavoitteet saavutettiin pääosin, projektissa ilmeni myös haasteita, kuten anturoiden mallinnus Grasshopperin algoritmin avulla. Nämä ongelmat tarjosivat arvokasta oppia algoritmipohjaisen suunnittelun tarkkuudesta ja sen vaatimasta huolellisuudesta.

Ilman tätä työtä emme olisi saaneet arvokasta tietoa parametrisen suunnittelun hyödyistä ja sen potentiaalista rakennesuunnitteluun. Työ paljasti myös parametrisen mallinnuksen merkittäviä etuja, kuten suunnitteluprosessin nopeuden ja virheiden vähentämisen.

Työtä voisi jatkaa tekemällä algoritmiin liitokset ja komponentit, esimerkiksi peruspultit, pilarikengät ja pilari-palkki-liitos.

Työn tuloksia voidaan hyödyntää laajasti yrityksessä suunnittelu työssä. Tämä insinööriyö vahvistaa algoritmisen suunnittelun roolia rakennusalan tulevaisuudessa ja sen kykyä parantaa suunnittelun laatua ja tehokkuutta.

Lähteet

- 1 The benefits of parametric design for structural engineers. Verkkoaineisto. Tekla. <https://www.tekla.com/resources/articles/the-benefits-of-parametric-design>. Luettu 20.2.2024.
- 2 Major Features and Popular Workflows of Rhino 3D. Verkkoaineisto. SimplyRhino. <https://simplyrhino.co.uk/product-rhino-key-features>. Luettu 4.3.2024.
- 3 Dušan Cvetković. 2023. Rhino 3D Modeling World – The Secrets You Didn't Know. <https://howtorhino.com/blog/software-for-architects/rhino-3d-modeling/>. Luettu 7.3.2024.
- 4 Features. Rhinoceros. <https://www.rhino3d.com/features/>. Luettu 7.3.2024.
- 5 Rhinoceros. <https://www.rhino3d.com/>. Luettu 7.3.2024.
- 6 Brendan Harmon. An Introduction to Grasshopper. <https://baharmon.github.io/basics>. Luettu 14.3.2024.
- 7 Tekla. <https://www.tekla.com/products/tekla-structures>. Luettu 17.3.2024.
- 8 Tekla. https://support.tekla.com/help/tekla-structures/not-version-specific/grasshopper-tekla-link?old_alias_match=1&incoming_url=https%3A%2F%2Fteklastructures.support.tekla.com%2Fnot-version-specific%2Fen%2Fext_grasshopper-tekla-link#top. Luettu 17.3.2024.