

Veikka Vikstedt

TERÄSRAKENTEEN PARAMETRINEN MALLINTAMINEN

S1 Huikun liittopalkkisiltaan

TERÄSRAKENTEEN PARAMETRINEN MALLINTAMINEN

S1 Huikun liittopalkkisiltaan

Veikka Vikstedt
Opinnäytetyö
Kevät 2024
Rakennustekniikan tutkinto-ohjelma
Oulun ammattikorkeakoulu

TIIVISTELMÄ

Oulun ammattikorkeakoulu

Rakennustekniikan tutkinto-ohjelma, talonrakennustekniikan suuntautumisvaihtoehto

Tekijä: Veikka Vikstedt

Opinnäytetyön nimi: Teräsrakenteen parametrinen mallintaminen

Opinnäytetyön englanninkielinen nimi: Parametric modeling of the steel structure

Työn ohjaaja(t): DI Juho-Martti Vinkki (OAMK), DI Tuomo Järvenpää (A-Insinöörit) ja DI Aleksi Mäki-Mantila (A-Insinöörit)

Työn valmistuslukukausi ja -vuosi: Kevät 2024

Sivumäärä: 35

Tietomallintamisella tarkoitetaan rakennushankkeesta tehtävää 3D-mallia, joka sisältää tietoa rakennettavasta kohteesta. Tietomallintaminen on yksi tärkeä osa suunnitteluprosessia, jonka tekemisessä voi mennä huomattava määrä resursseja, riippuen hankkeen koosta. Tietomallintamisen parametrisoinnin avulla voidaan kehittää tietomallintamisesta tehokas osa suunnitteluprosessia.

Tämän opinnäytetyön aiheena on kehittää, jo aiemmin tehtyä algoritmipohjaa Rhinoceros 7 -ohjelmiston Grasshopper-lisäosaa hyödyntäen. Tavoitteena on lisätä algoritmiin uusia osia, joiden avulla saadaan teräsrakenne mallinnettua S1 Huikun sillan tietomalliin. Opinnäytetyö tehtiin toimeksiantona A-Insinöörit Suunnittelu Oy:lle.

Opinnäytetyö koostuu teoriaosuudesta liittyen parametriseen tietomallintamiseen, sekä toteutusosasta, jossa rakennetaan algoritmipohja. Työssä havaittiin, että Grasshopperin avulla saadaan tehtyä toimiva algoritmipohja ja sen avulla saadaan säästettyä resursseja S1 Huikun -projektin tietomallinnuksessa.

Asiasanat: parametrinen tietomallintaminen, algoritmiavusteinen suunnittelu, Grasshopper, Tekla

ABSTRACT

Oulu University of Applied Sciences
Degree Program in Civil Engineering, Option of House Building Engineering

Author: Veikka Vikstedt

Title of thesis: Parametric modeling of the steel structure

Supervisors: MSc Juho-Martti Vinkki (OAMK), MSc Tuomo Järvenpää (A-Insinöörit) ja MSc Aleksi Mäki-Mantila (A-Insinöörit)

Term and year when the thesis was submitted: Spring 2024

Number of pages: 35

BIM-modeling is one important part of the design process, which can take a considerable number of resources, depending on the size of the project. With the help of BIM-modeling parameterization, information modeling can be developed into an effective part of the design process.

The topic of this thesis is to develop an already made algorithm using the Grasshopper add-on of the Rhinoceros 7 software. The goal is to add new parts to the algorithm, which the steel structure can be modeled in the BIM-model of the S1 Huikku bridge. The thesis was commissioned by A-Insinöörit Suunnittelu Oy.

The thesis consists of a theory part related to parametric information modeling, and an implementation part where the algorithm is built. In the work, it was discovered that with the help of Grasshopper a functional algorithm can be made and with it, resources can be saved for BIM-modeling related to the S1 Huikku project.

Keywords: Parametric modelling, Algorithms-Aided Design, Grasshopper, Tekla

ALKULAUSE

Kiitokset A-Insinöörit Suunnittelu Oy:lle tämän opinnäytetyön mahdollistamisesta. Lisäksi haluan esittää kiitokseni opinnäytetyöni ohjaajille: Juho-Martti Vinkille ja Tuomo Järvenpäälle, jotka ovat antaneet arvokasta ohjausta työni kehittämisessä ja omalta osaltaan mahdollistaneet työni toteutumisen. Erityisen kiitollinen olen Aleksi Mäki-Mantilalle, olet ollut korvaamaton apu aina työni alusta sinetöintiin asti. Kiitokset myös SITA-yksikön Jarkko Savolaiselle, joka auttoi opinnäytetyöni alussa Grasshopperin käytössä.

Suurikiitokset kaikille läheisilleni, jotka ovat tukeneet minua tämän työn aikana. Kiitän lämpimästi myös työyhteisöäni, joka on kannustanut ja auttanut minua opinnäytetyöni tiellä. Haluan myös kiittää kaikkia opiskelukavereitani yhteisistä opiskeluaajoista ja unohtumattomista hetkistä opiskeluaikana. Viimeiset kiitokset menevät tyttöystävälleni Ellalle jatkuvasta ja jokapäiväisestä tuesta niin tämän työn kuin opintojeni aikana.

Oulussa 14.3.2024

Veikka Vikstedt

Veikka Vikstedt

SISÄLLYS

1	JOHDANTO	7
2	TIETOMALLINTAMINEN.....	8
2.1	Tietomallinnuksen historiaa.....	8
2.2	Tietomallinnus nykypäivänä.....	9
3	PARAMETRINEN TIETOMALLINTAMINEN.....	10
3.1	Parametrisessa tietomallinnuksessa käytettävät ohjelmat	11
3.2	Parametrisen tietomallinnuksen edut.....	11
3.3	Parametrisen tietomallinnuksen haasteet.....	13
4	RHINOCEROS 3D JA GRASSHOPPER	14
4.1	Rhinoceros 3D	14
4.2	Grasshopper	14
4.3	Grasshopper-Tekla Live Link	16
5	HUIKUN SILTA.....	18
6	TERÄSRAKENTEEN PARAMETRINEN MALLINTAMINEN.....	19
6.1	Teräsrakenteiden mittaparametrien automatisointi.....	20
6.2	Pystyristikon mallintaminen.....	22
6.3	Vaarnatappien mallintaminen pääkannattimen ylälaippaan	23
6.4	Väliristikon mallintaminen.....	25
6.5	Kiinnityslevyt	26
6.6	Vaakadiagonaalit	29
6.7	Valmis teräsrakenne.....	31
7	POHDINTA	32
	LÄHTEET.....	34

1 JOHDANTO

Tietomallintaminen on yksi merkittävä osa rakennushankkeiden suunnitteluprosessia. Sen avulla saadaan välitettyä tietoa rakennettavasta kohteesta tehokkaasti eteenpäin rakennushankkeiden toimijoiden välillä. Tietomallinnuksen tarkoituksena on tuottaa kattava 3D-malli, jossa on tarvittavat tiedot kohteesta sen rakentamiseksi. Tietomallintaminen yleistyi 2000-luvun alkupuolella sillanrakennushankkeissa ja se on jatkanut kehittymistään näihin päiviin asti.

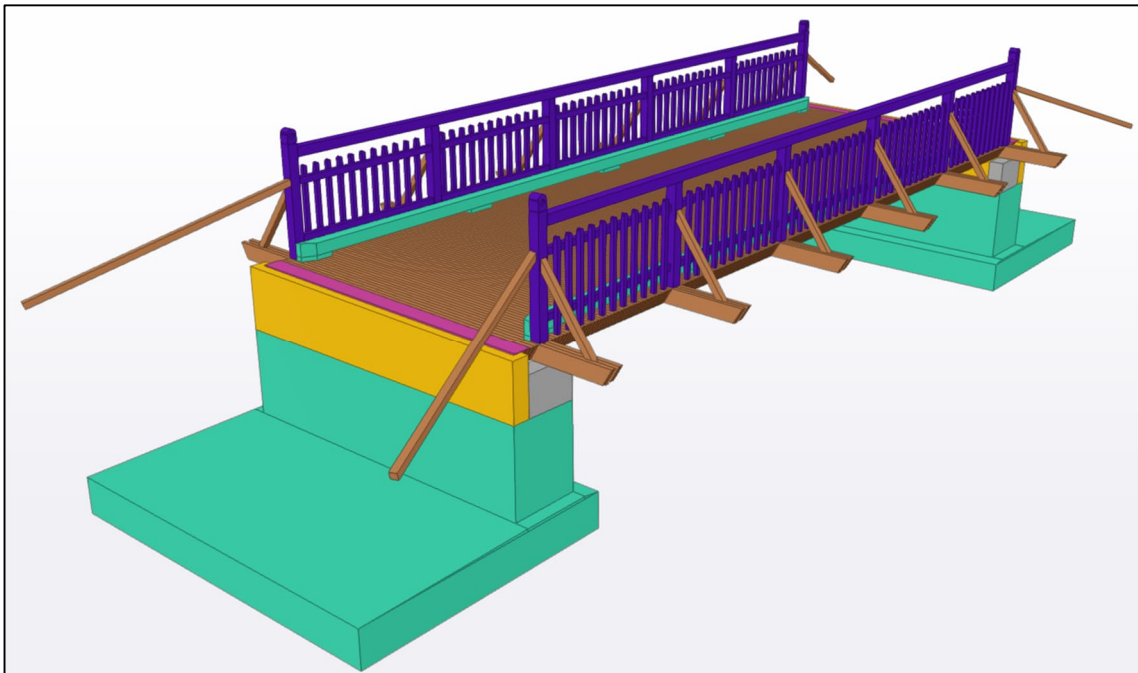
Parametrisella tietomallintamisella rakennetaan tietomalli käyttäen parametreja, jotka määräävät mallin geometrian tai toiminnan. Mallissa olevia objektien määrää, muotoja tai muita ominaisuuksia voidaan siis muokata muuttamalla annettuja parametreja. Tämä mahdollistaa suunnitteluprosessia joustavuutta muutoksille ja muodostaa tietomallintamisesta tehokkaan osan suunnitteluprosessia.

Isoissa rakennushankkeissa tietomallintamisen määrä myös kasvaa ja se vie resursseja suunnitteluprosessista. Yksi suurimpia haasteita projekteissa tietomallintamisen suhteen on muutokset, joita tulee suunnittelun edetessä. Esimerkiksi rakenteiden mitoitusprosessien edetessä huomataan, että rakenteen mitat tai muut ominaisuudet eivät ole riittäviä kestävyyskannalta. Myös toisteisuutta erilaisten objektien mallintamiseen voi tulla huomattavasti.

Tämän opinnäytetyön tarkoituksena on tutkia sekä ratkaista näitä ongelmia rakentamalla Grasshopper-ohjelmistolla algoritmi, jonka avulla mahdolliset muutokset hankkeessa on helpompi toteuttaa. Lisäksi opinnäytetyössä käsitellään yleisesti parametrilla mallintamista, sekä pohditaan sen tuomia etuja kuin myös haasteita tietomallintamiseen liittyen. Työn toimeksiantajana on A-Insinöörit Suunnittelu Oy ja työ toteutetaan Hailuodon kiinteän yhteyden S1 Huikun sillan liittopalkkisillan. Opinnäytetyö on osa sillan suunnitteluprosessia ja tehtävänä on muodostaa algoritmi sillan teräsrakenteelle.

2 TIETOMALLINTAMINEN

Tietomalli tarkoittaa rakennushankkeesta tehtyä tietokoneella suunniteltua 3D-mallia, joka sisältää tietoa rakennettavasta kohteesta. Tietomallin tarkoituksena on havainnollistaa rakennushanketta paremmin sekä tarjota tarvittavat tiedot rakennettavasta kohteesta sekä sen tulevaisuuden ylläpidosta. (Väylävirasto 2020.) Kuvassa 1 näkyy kevyenliikenteensillasta rakennettu tietomalli, josta voidaan esimerkiksi ottaa erilaisia mittoja rakenteille sekä tarkastella rakenteiden materiaalitietoja.



KUVA 1. Sillan tietomalli

2.1 Tietomallinnuksen historiaa

Ennen tietomallien yleistymistä, projekteja tehtiin CAD-ohjelmistoilla, jonka nimi tulee sanoista Computer-aided Design. CAD-ohjelmistot kehitettiin auttamaan suunnittelijoita piirustusten teossa. Ennen tietokoneita piirustukset rakennushankkeissa tehtiin käsin kynällä. CAD-piirtäminen oli ensin kaksiulotteisia, ja se yleistyi suunnittelussa 1980-luvulla ja 1990-luvulla se syrjäytti käsin tehdyt piirustukset. (Nova Design 2024.)

CAD-ohjelmistojen kehityttyä tuli tarve CNC-koneistuksen kautta myös 3D-malleille. Ensimmäiset 3D-mallit käyttivät niin sanottuja rautalankamalleja ja ne kehitettiin 1970-luvulla, mutta yleistyivät vasta 1980-luvulla. (Ball 2013.)

2.2 Tietomallinnus nykypäivänä

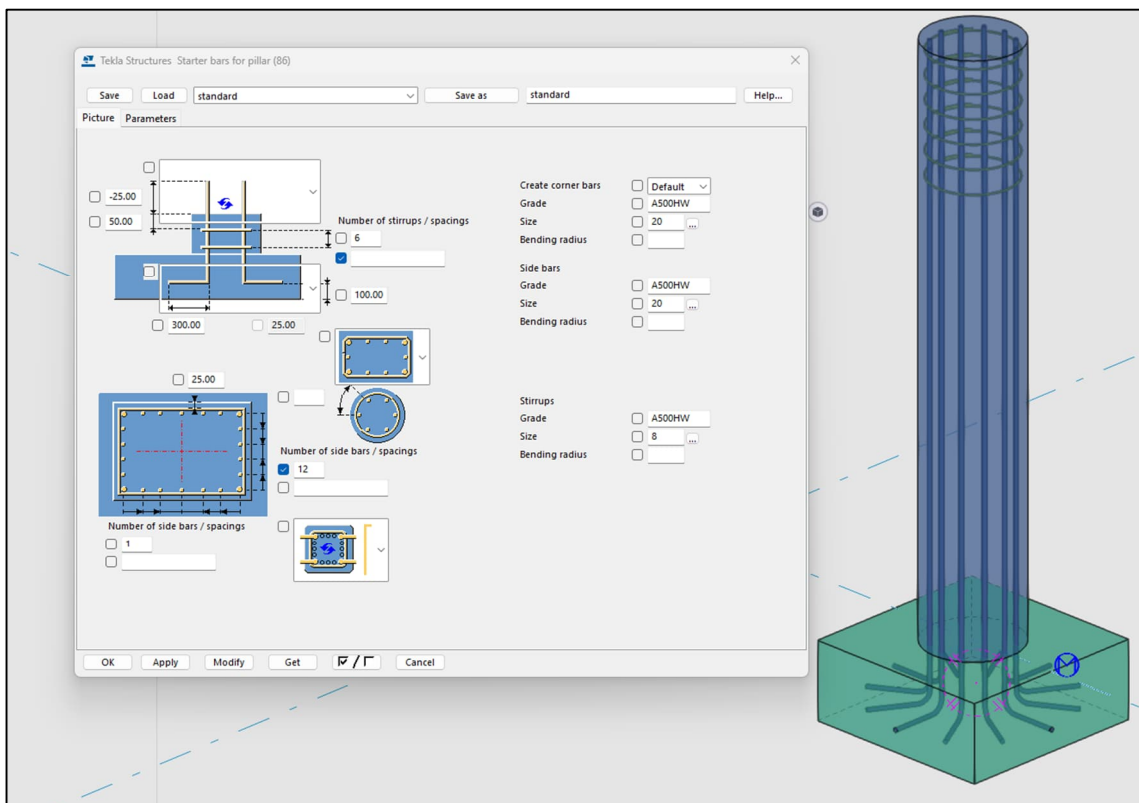
Tietomallinnuksessa käytettäviä ohjelmia ovat muun muassa Trimblen tarjoama Tekla Structures ja Autodeskin Revit. Tässä opinnäytetyössä tietomallinukseen käytetään Tekla Structures ohjelmistoa. Vuonna 1966 yritys nimeltä Teknillinen Laskenta Oy perustettiin Suomessa. Tästä tuleekin nimi Tekla. Se perustettiin kehittämään sen ajan tietokoneille toimivaa ohjelmistoa, jonka avulla voitiin suorittaa laskuja esimerkiksi tiesuunnitteluun liittyen. (Tamminen 2006, 11). Yhdysvaltalainen Trimble inc. osti Teklan 2010-luvulla ja on jatkanut Teklan kehitystä eteenpäin.

Tietomallien hyödyntäminen toden teolla alkoi 2000-luvun alkupuolella. Tietomallit yleistyivät silta-rakentamisessa, koska väylävirasto sitoutui käyttämään hankkeissaan tietomalleja. Tietomalleista on todella paljon hyötyä tulevaisuutta ajatellen, sillä tietomallista löytää kaikki tarvittavat tiedot esimerkiksi sillan rakenteista ja sen osista. Täten voidaan hyvin ylläpitää myös sillan kuntoa. (Väylävirasto 2020.)

3 PARAMETRINEN TIETOMALLINTAMINEN

Parametrinen tietomallintaminen on suhteellisen uusi suuntaus tietomallintamisessa. Vasta viime vuosikymmenellä alan ammattilaiset ovat alkaneet hyödyntämään sitä toden teolla (Davis 2013). Parametrisellä tietomallintamisella tarkoitetaan tietomalliin luotujen geometrinen objektien välistä riippuvuussuhdetta algoritmipohjaan, jota ohjataan parametrien avulla. Parametri on muuttuja, joka ohjaa algoritmiprosessia. Esimerkiksi kuvassa 2 näkyvä pilarin korkeus voisi olla yksi parametri. (Tanska & Österlund 2014, 13.)

Teklassa on itsessään valmiina parametrisia komponentteja, joiden avulla voidaan mallintaa joitain objekteja. Esimerkiksi kuvan 2 pilarin raudoitteet mukautuvat automaattisesti pilarin koon muuttuessa. Teklan omat komponentit ovat rajallisia eivätkä välttämättä sovellu kunnolla kaikkiin tilanteisiin mallinnettaessa.



KUVA 2. Parametrisen komponentin esimerkki Teklassa

3.1 Parametrisessa tietomallinnuksessa käytettävät ohjelmat

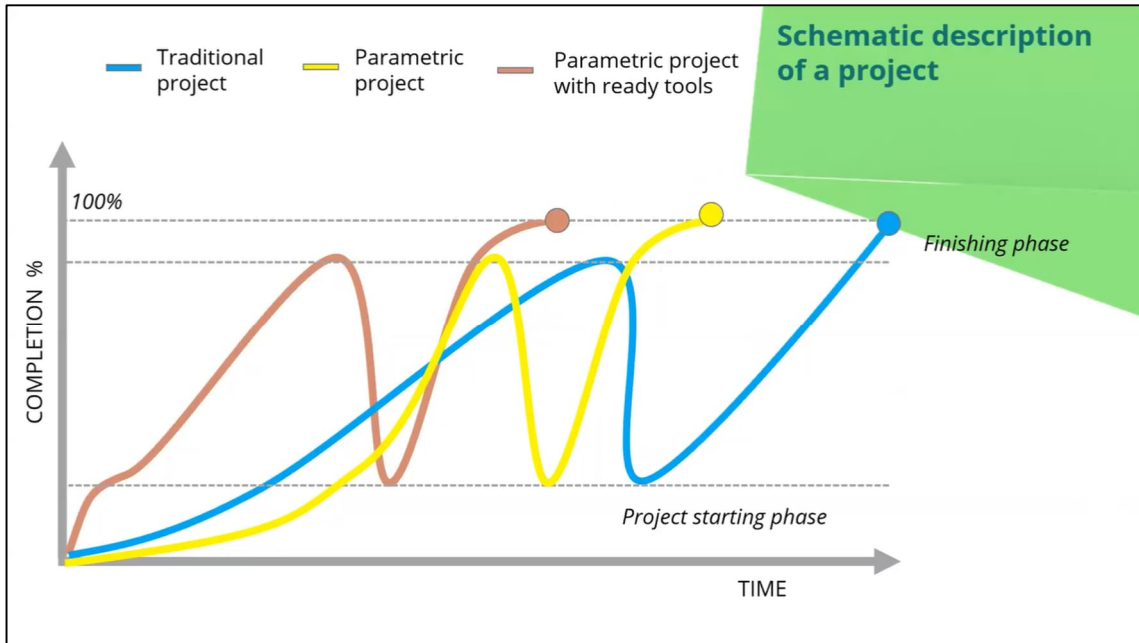
Parametriseen tietomallintamiseen on tarjolla erilaisia ohjelmistoja käytettäväksi. Tässä opinnäytetyössä käytettävä ohjelma on Rhinoceros 7, josta tässä työssä käytetään lyhennettä "Rhino". Rhinon tarjoama lisäpaketti Grasshopper on ohjelmisto, jolla rakennetaan tarvittavia algoritmeja mallintamiseen. Muita ohjelmia, joita käytetään, ovat esimerkiksi Bentley Generative Components ja Autodesk Dynamo (Tanska & Österlund 2014, 13.)

3.2 Parametrisen tietomallinnuksen edut

Parametristä tietomallinnusta voidaan käyttää mallin automatisointiin eli pyritään tekemään tietomallinnuksesta helpompaa ja nopeampaa parametrisen suunnittelutyökalun avulla. Rakentamalla kerralla hyvä algoritmi, jolla mallinnetaan joku silloissa yleisesti esiintyvä osa, säästetään aikaa tulevilla hankkeilla. (Tekla 2024.)

Silloissa on paljon tyyppiirustuksia, ja monista silloista löytyy samankaltaisia rakennelmia, kuten esimerkiksi siipimuurit. Siipimuurit ovat sillan maatuissa sijaitsevat seinämät, joiden tehtävä on pitää maa-aines paikoillaan siirtymälaatan alla. Grasshopperin avulla voidaan suunnitella työkalu, jonka kautta voidaan helposti muuttaa siipimuurin mittoja, ja samalla myös siipimuurin raudoitteet muuttuvat itse siipimuurin muuttuessa. Tämä säästää aikaa tietomallinnuksessa, ja rakennettua työkalua voidaan käyttää hyväksi myös tulevilla hankkeilla.

Rakennushankkeissa tulee suunnitteluvaiheen aikana useimmiten vastaan erilaisia muuttujia. Nämä muuttujat voivat olla jonkun lähtötiedon muuttuminen tai vaikka mitoituksen aikana huomataan, että tarvitaan jokin toinen ratkaisu, jotta saadaan rakennelma kestäväksi. Muuttujat kuitenkin aiheuttavat lisää töitä ja sitä kautta projektiin voi kulua aikaa odotettua enemmän. Parametrisen tietomallintamisen avulla saadaan näistä muuttujista aiheutuvia ongelmia korjattua nopeammin. Kuvassa 3 on esimerkki kolmesta eri projektityylistä ja niissä tapahtuvista muuttujista sekä muuttujien vaikutuksesta projektiin kulkuun. (Tekla 2024).



KUVA 3. Projektien ja parametrusten projektien aikajana kuvattuna muutosten kanssa (Wojslaw 2023)

Kuvasta 3 voidaan havaita, että perinteinen projekti vie aikaa enemmän verrattuna muihin. Grasshopperia käytettäessä saadaan parametrisonnin avulla säästettyä aikaa verrattuna perinteiseen projektiin, kun vastaan tulee muutos hankkeessa. Kaikista tehokkain tapa mallintamiseen on, kun Grasshopperilla tehtyjä valmiita algoritmipohjia hyödynnetään hankkeissa.

Perinteinen tapa on tällä hetkellä luovuttaa tilaajalle valmiit piirustukset 2D-muodossa. Malliin perustuvissa projekteissa ei tehdä erikseen 2D-kuvia, jotka toimitetaan tilaajalle. Tarvittavat tiedot kohteen rakentamiseen tulee siis suoraan mallista. Malliin perustuvissa projekteissa parametriseistä tietomallintamisesta on todellakin höytyä tietomallia rakentaessa, ja sen avulla voidaan tehdä entistä monimutkaisempia kohteita. Kun malliin perustuvat projektit ovat helpompia toteuttaa parametriseistä tietomallinnuksen avulla, säästetään myös aikaa, kun ei tarvitse erikseen tehdä 2D-kuvia työmaalle. (Wojslaw 2023.)

3.3 Parametrisen tietomallinnuksen haasteet

Parametrinen tietomallintaminen käsitteenä on uudehko alalla ja se avaa uusia mahdollisuuksia toteuttaa rakennushankkeita. Parametrisessä tietomallinnuksessa on myös omat haasteensa. Parametrisen tietomallinnuksen ohjelmistot ovat käyttöjärjestelmiltään suhteellisen normaaleja käyttää. Haasteena on tietää, mitä ohjelmistoilla tekee. Toimivan työkalun suunnittelu ohjelmiston avulla vaatii laaja-alaista ymmärrystä ja tietoa ohjelmiston käytöstä. (Thet Hnin 2022; Davis 2013.)

Haasteina on myös käyttäjille se, että ohjelmisto tarjoaa valtavan määrän uusia työkaluja. Mahdollisuudet luoda uusia ratkaisuja ovat laajat. Tämä voi aiheuttaa varsinkin ohjelmistoihin tutustuville helposti sekavan tunteen, kun näytöllä näkyy monta kuvaketta, josta voi painaa ja tuottaa erilaisia ratkaisuja. (Thet Hnin 2022.)

Yksi haaste myös Grasshopperin käytössä on, että se on kolmannen osapuolen sovellus. Tämän takia Grasshopperilla voi olla yhteensopivuusongelmia tiettyjen Tekla-versioiden suhteen tai joitain rajoituksia käytössä, joita ei voida ratkaista. (Wojslaw 2023.)

4 RHINOCEROS 3D JA GRASSHOPPER

4.1 Rhinoceros 3D

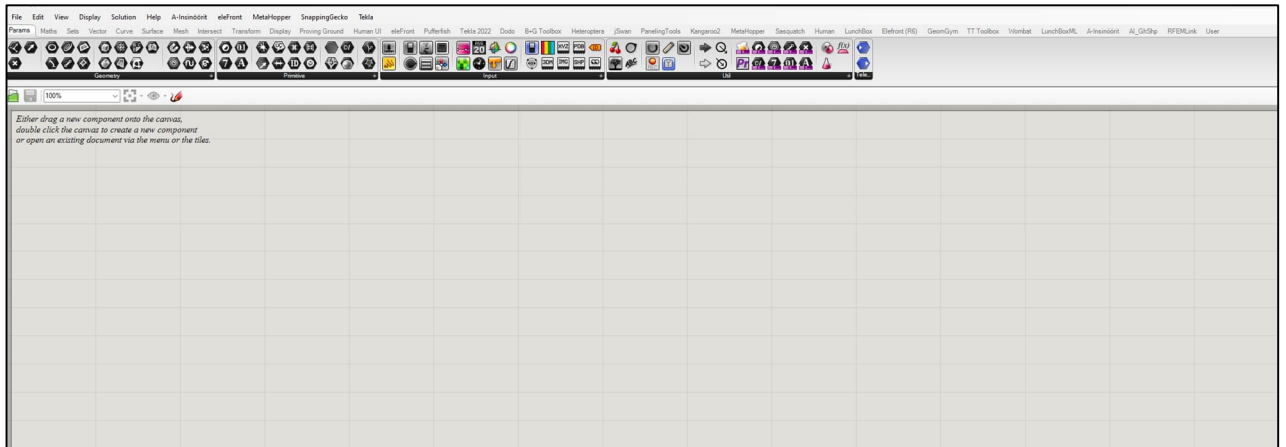
Rhinoceros 3D, josta tässä opinnäytetyössä käytetään lyhennettä Rhino, on Robert McNeel & Associatesin luoma NURBS-pintojen ja -viivojen 3D-mallinnusohjelma (Tanska & Österlund 2014, 30). NURBS on vaihtoehtoinen esitystapa geometrisen mallin mesh-pinnoille.

NURBS-pinnat perustuvat hyvin tarkkaan matemaattiseen laskentaan ja täten mallinnustarkkuudesta ei tarvitse huolehtia, koska jokainen piste näin luotavassa geometriassa on mittatarkka (Tanskanen & Österlund 2014, 13). Tässä opinnäytetyössä keskitytään käyttämään enemmän Rhinon lisäosaa Grasshopperia.

4.2 Grasshopper

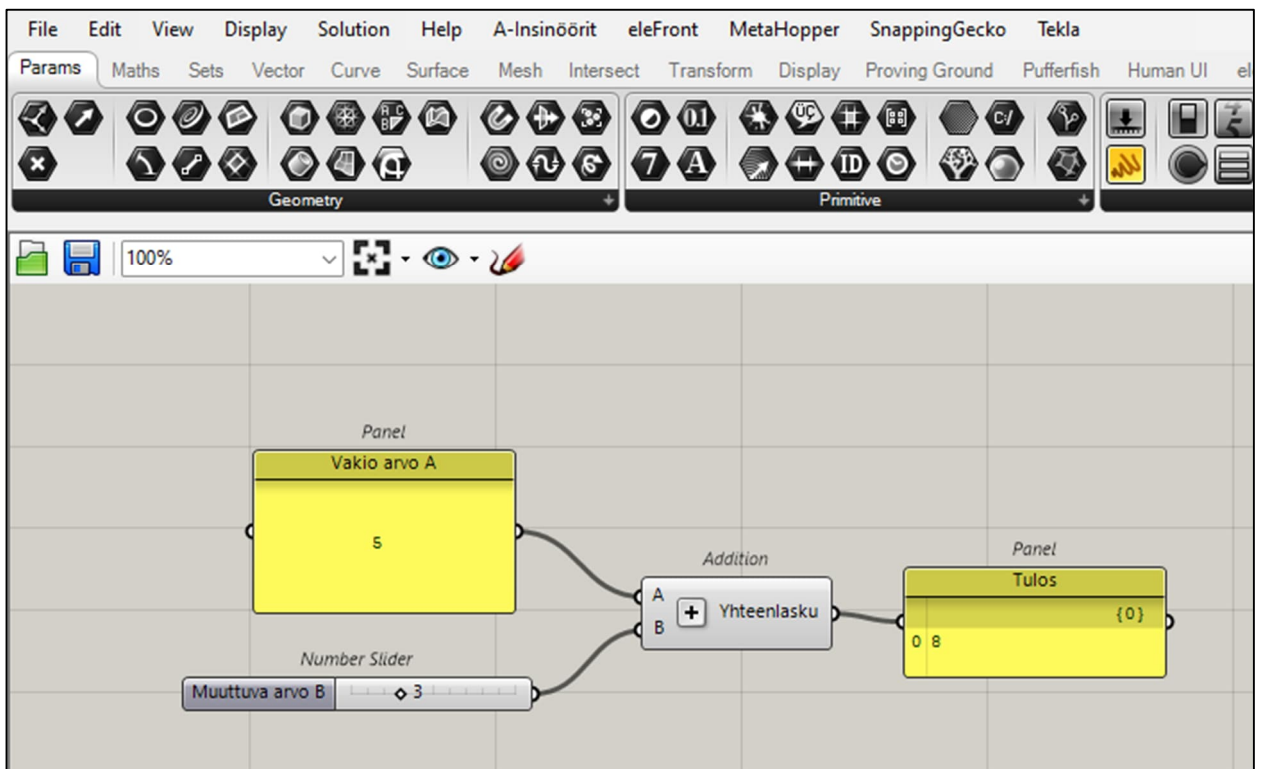
Tietomallintamisen ja suunnittelun murros johdatti McNeel & Associatesin luomaan uuden visuaalisen sovelluksen, joka perustuu algoritmeihin. Tämä uusi visuaalinen sovellus ei tarvinnut perinteistä koodausta. Sen ensimmäinen versio oli nimeltään "Explicit History" ja lopulta sen nimeksi vakiintui Grasshopper. Ohjelmien kehittämisen takana oli David Rutten. (Tedeschi 2011.)

Grasshopperin ideana on tuottaa suunnittelijoille lisää työkaluja ratkaisuihin suunnittelutoimistolla. Perinteinen koodaaminen voi tuntua hyvin vaikealle sellaisille henkilöille, jotka eivät asiaa paremmin tunne. Grasshopper toimii visuaalisena koodausohjelmana, jonka tarkoitus on helpottaa suunnittelijoiden työtä. Kuvassa 4 on Grasshopperin perusnäkyminen ohjelmisto avatessa. (Tanska & Österlund 2014, 30–31.)



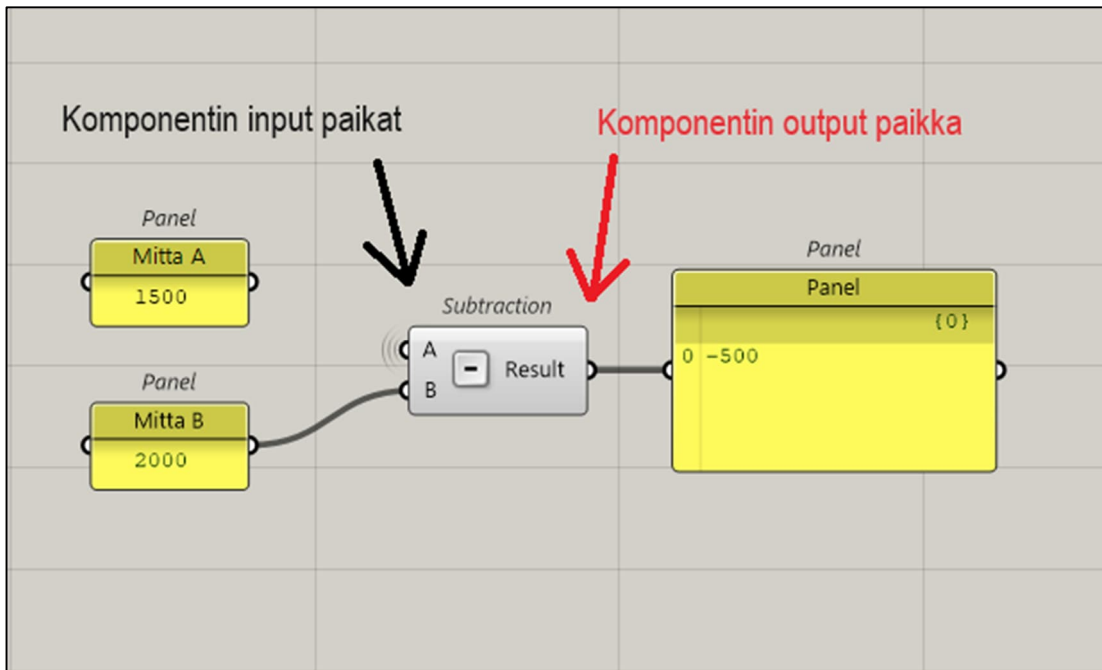
KUVA 4. Grasshopperin käyttöliittymä

Grasshopperin käyttöliittymä toimii siten, että käyttäjä raahaa käyttöliittymän työpöydälle erilaisia työkaluja. Nämä työpöydällä olevat työkalut kytketään toisiinsa langoilla. Kuvassa 5 on esitetty yksinkertainen esimerkki Grasshopperilla toteutetusta parametrisoidusta laskutoimituksesta. Lähtöarvoja muuttamalla saadaan muutettua algoritmin lopputulosta. (Hirvikoski & Karjalainen 2021.)



KUVA 5. Esimerkki kahden lähtöarvon yhteenlasku

Grasshopper -komponenteilla on output- ja input-paikat, johon voidaan vetää lankoja. Lankojen näkyvyyttä voidaan muuttaa asetuksista ja ne kertovat erilaisia tietoja käyttäjälle komponenttiin menevästä langasta. Langan voi myös halutessaan piilottaa, jolloin tehdystä algoritmista voi tulla selvempi. Toisaalta jos lankoja piilottaa liikaa, voi olla vaikea hahmottaa algoritmin tiedon kulkua. Kuvassa 6. näkyy komponentin output- ja input-paikat sekä esimerkki langattomasta ja langallisesta yhteydestä. (Hirvikoski & Karjalainen 2021.)



KUVA 6. Komponentin in- ja output-paikat

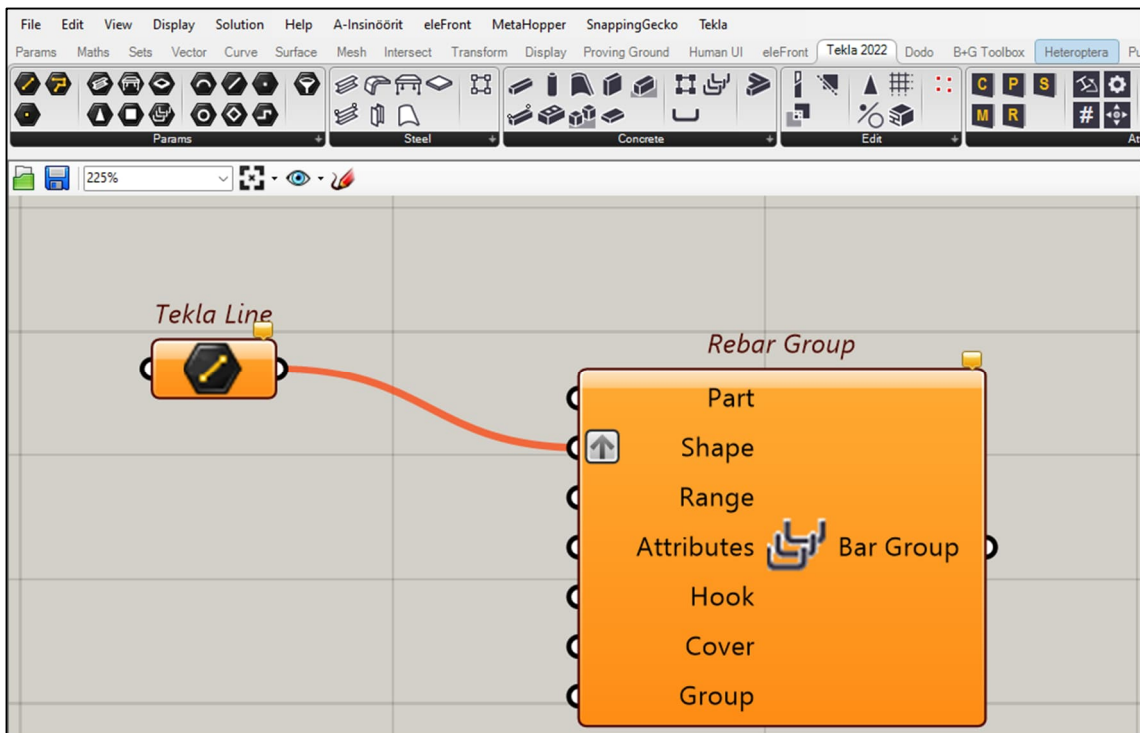
Kun komponentit raahataan valkokankaalle, antavat ne eri värikoodin. Värikoodi kertoo käyttäjälle tiedon komponentin tilasta. Oranssi väri kertoo komponentin olevan ilman dataa. Punainen väri kertoo virheestä. Vihreä väri kertoo, että komponentti on valittuna. Harmaa väri kertoo, että komponenttiin tulee dataa, ja se toimii normaalisti. (Hirvikoski & Karjalainen 2021.)

4.3 Grasshopper-Tekla Live Link

Rhinon hyvä ominaisuus on se, että Rhinon voi yhdistää lukuisiin suunnittelualan eri ohjelmistoihin. Grasshopperiin voi Teklan nettisivuilta ladata Tekla Live Linkin, jonka avulla saadaan yhdistettyä Grasshopper ja Tekla. (Wojslaw 2023.)

Huomioitavaa on se, että ladatun Tekla Live Linkin pitää vastata sitä Tekla-versioita, joka käyttäjällä on ladattu tietokoneelle. Esimerkiksi Tekla 2022 -versioon tarvitaan Tekla Live Link 2022 -versio. Tekla 2022 -version kanssa ei siis voi käyttää Live Link 2021 -versiota. (Wojslaw 2023.)

Kun yhdistää Teklan ja Grasshopperin live linkin avulla, pitää ensimmäisenä avata Tekla. Tämän jälkeen, kun haluttu malli Teklasta on avattu, voidaan avata Rhino. Mikäli Rhinon avaa ennen Teklaa ei Live Link toimi. Kun Tekla Live Link on ladattu Grasshopperiin, luo Grasshopper uuden Tekla-välilehden. Tekla-välilehdellä on valmiina parametriset työkalut, jonka avulla voidaan mallintaa suoraan Tekla-malliin objekteja Grasshopperin avulla. Kuvassa 6. on esimerkki Teklan parametrisestä työkalusta, joka hakee tiedot Tekla-mallista. (Wojslaw 2023.)



KUVA 7. Tekla Line -komponentti, joka hakee Tekla-mallista referenssiivian

5 HUIKUN SILTA

Opinnäytetyön tutkimusosa tehtiin Hailuodon kiinteän yhteyden allianssihankeeseen Huikun siltaan. Hankkeessa itsessään on kaksi siltaa, jotka ovat S1 Huikun ja S2 Riutun sillat. Siltojen välillä on pengertie, jonka pituus on 6,9 kilometriä. Hankkeen rakentamisen on tarkoitus alkaa alkukesästä 2024. (Väylävirasto 2024.)

Kuvassa 8 nähdään suunnitellun Huikun sillan havainnekuva. Huikun sillan suunnittelee A-Insinöörit ja se sijaitsee Hailuodon puolella hankkeessa. Sen kokonaispituus on 767 metriä, mikä tekee siitä Suomen kuudenneksi pisimmän sillan. Kannen hyödyllinen leveys on 9,5 metriä ja alikulukorkeus on 18 metriä laivaväylän kohdalla. Silta on teräsbetonikantinen jatkuva liittopalkkisilta, jossa on tasakorkeat palkit. Rakenneteräsmäärä on noin 2,3M kg ja kansi valetaan yhdellä valulla. (Väylävirasto 2024.)

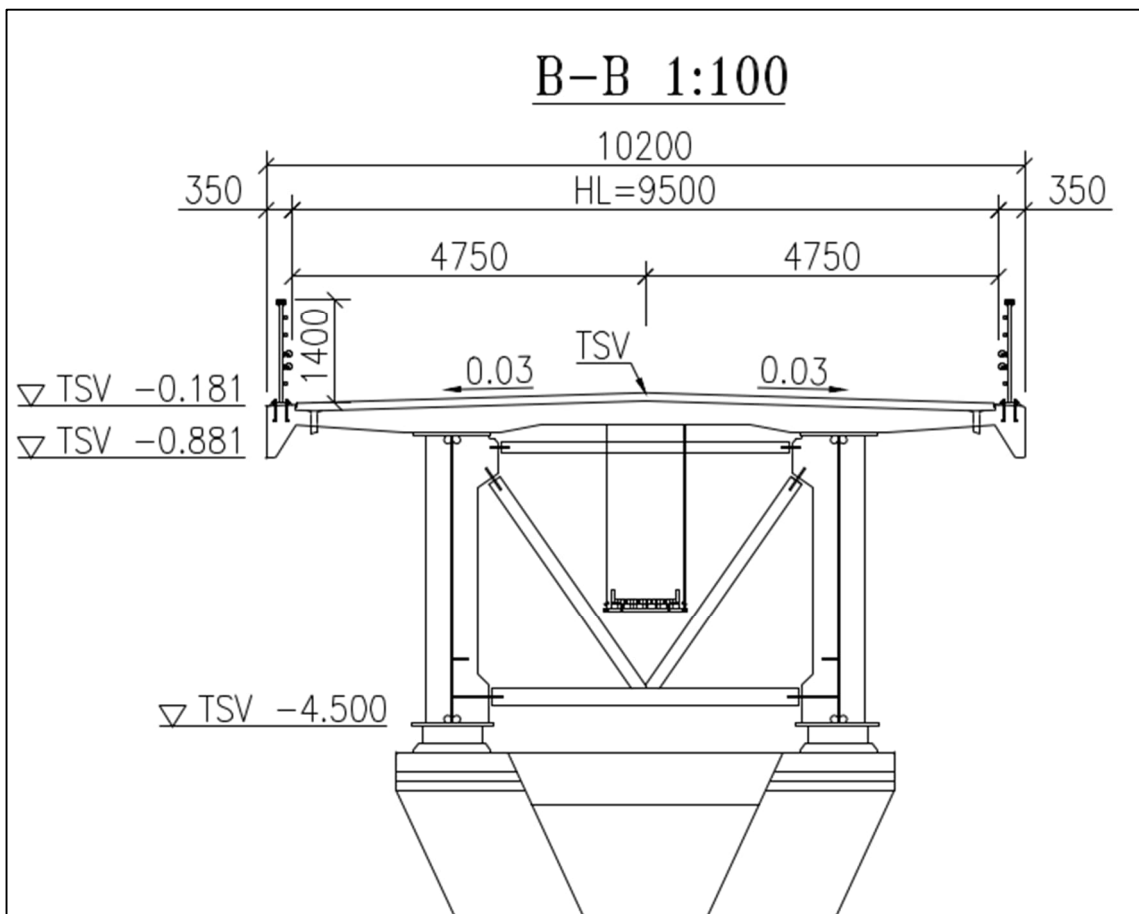


KUVA 8. Huikun sillan visualisointikuva (Afy 2023)

6 TERÄSRAKENTEEN PARAMETRINEN MALLINTAMINEN

Tämän opinnäytetyön lähtökohtana oli jatkaa kehitystyötä Huikun sillan teräsrakenteen parametri-
sessa mallinnuksessa. Teräsrakenteen pääkannattimista oli valmiina Grasshopperilla tehty algo-
ritmi ja kehitystyön ideana oli tuottaa teräsrakenteen muut osat parametrisesti mallintaen.

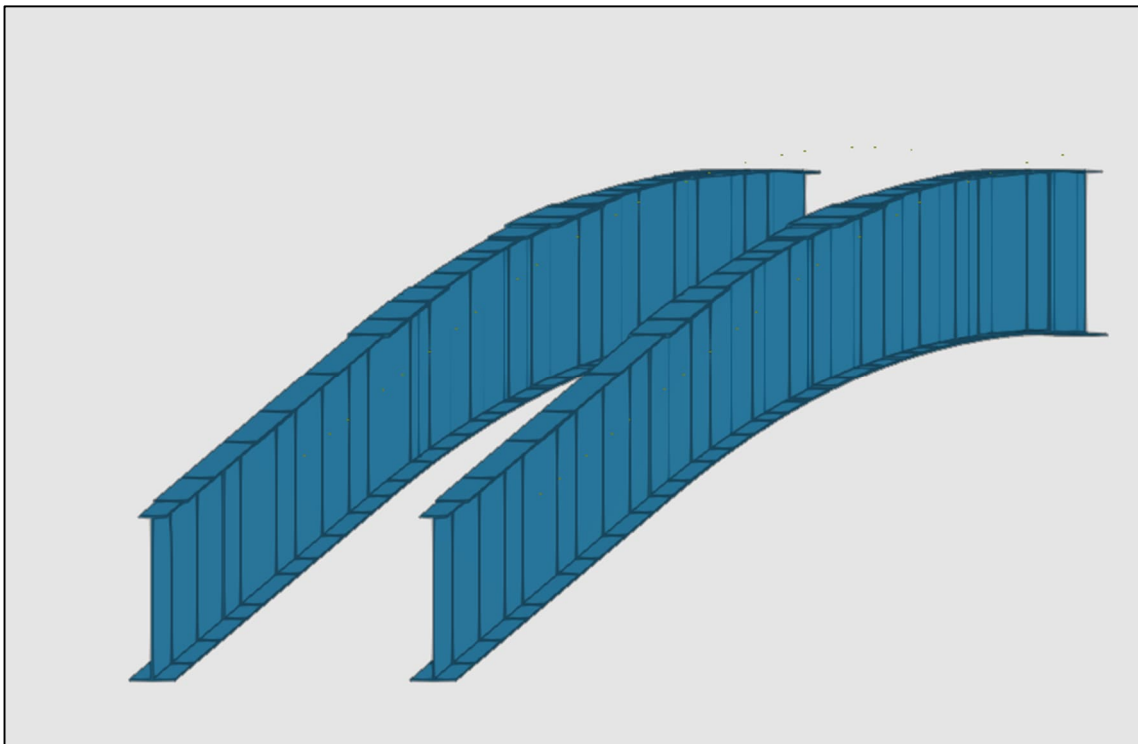
Teräsrakenteen manuaalinen mallinnus on hyvin pitkä ja virhealtis prosessi. Manuaalinen mallin-
nus sitoo merkittävän määrän resursseja, kuten suunnittelijoita ja aikaa. Grasshopperin avulla mal-
linnettu teräsrakenne säästää resursseja. Teräsrakenteesta pitää tehdä kaksi eri tietomallia. Toi-
nen tietomalli on sillan varsinainen tietomalli, jossa näkyy kaikki tarvittavat tiedot valmiista sillasta.
Toinen tietomalli on konepajapiirustuksia varten, jossa teräsrakenne on mallinnettu jännityksetö-
mään muotoon. Kuvassa 9 nähdään Huikun sillan poikkileikkaus, josta ilmenee mallinnettava te-
räsrakenne.



KUVA 9. S1 Huikun sillan poikkileikkaus

6.1 Teräsrakenteiden mittaparametrien automatisointi

Pääkannattimista tehty algoritmi toimii siten, että se mallintaa kannattimet puoliksi sillan pituussuuntaan nähden. Tämä ratkaisu toimii, koska sillan teräsrakenne on symmetrinen. Algoritmin idea on mallintaa pääkannattimet täysin valmiiksi, jonka jälkeen valmiin sillan puolikkaan voi Teklan eri ominaisuuksia hyödyntäen kopioida peilikuvana paikalleen. Kuvassa 10 nähdään jo aiemmin valmiina olevalla algoritmilla mallinnetut pääkannattimet.

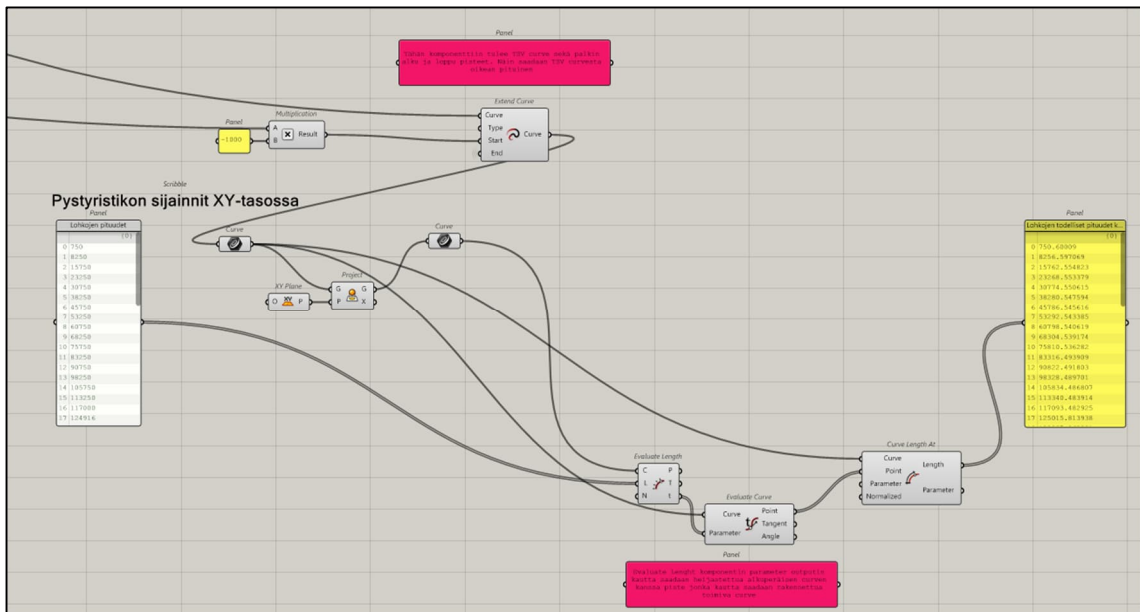


KUVA 10. Tietomallissa olevat pääkannattimet

Grasshopperiin tarvittavat teräsrakenteiden mitat otettiin AutoCAD LT:llä tehdyistä piirustuksista, jossa piirustus sijaitsee XY-koordinaatistossa. Teräsrakenteiden mitat oli ilmoitettu vaakasuoralla matkalla X-suuntaan. Teräsrakenteen kaaren mitta piti siis tarkistaa manuaalisesti käsin, pilkkomalla kaari halutuille lohkoille.

Ensimmäisenä asiana oli tutkia mahdollisuutta saada lohkon mitat Grasshopperin avulla parametrisesti. Rhinoon oli ladattu TSV eli tientasausviiva, joka ilmoittaa ajoradan korkeusaseman. Se oli lyhennetty valmiiksi sopivan mittaiseksi, jotta sitä voisi hyödyntää algoritmin teossa.

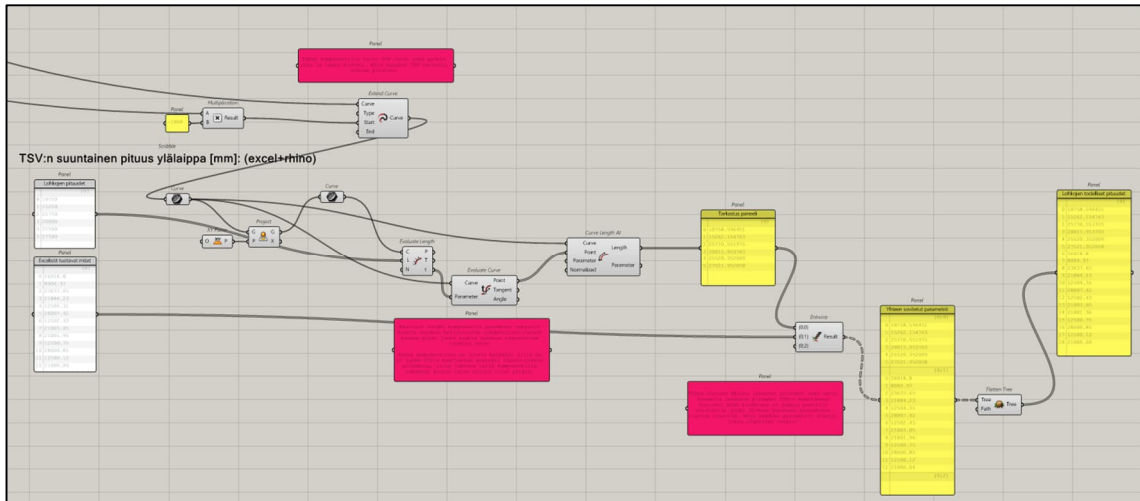
Kun TSV oli sopivan mittainen, piti TSV pilkkoa halutuille lohkoille, jonka mukaan teräsrakenne todellisuudessa rakennetaan. TSV projisoitiin aluksi XY-tasoon, jolloin siitä katosi Z-koordinaatti. Tämän jälkeen annettiin ensimmäinen lohkon mitta, ja liikuttiin projisoidulla TSV:llä tämän mitan verran eteenpäin. Evaluate Length -komponentin avulla saatiin sama piste heijastettua takaisin oikeaan TSV:hen. Kun tiedossa oli kaaren alkupiste ja tuo lohkon loppupiste, saatiin sitä kautta kuljettu matka TSV-kaarella. Tämä kuljettu matka kaarella siis vastasi teräsrakenteen mitta. Kuvassa 11 näkyy edellä mainittu algoritmi, jonka avulla saadaan kaarella kuljetut mitat, kun liikutaan vaakasuoralla matkalla tietty määrä.



KUVA 11. Algoritmi teräsrakenteen mitoille

TSV mittalinja kuitenkin muuttui keskellä siltaa suorasta viivasta kaareksi ja tämä aiheutti Grasshopperin laskevan kaaren pituudet väärin. Tähän piti kehittää Excelillä oma matemaattinen kaava, jonka avulla saatiin laskettua kaaren kohdalla kulkevat lohkon pituudet.

Kuvassa 12 näkyy lopputulemana kehitetty uusi algoritmi, jossa on yhdistettynä Grasshopperin laskemia mittoja suoran viivan matkalta, sekä Excelin laskemia mittoja kaaren osalta. Excelistä saadut mitat syötettiin paneeliin ja Entwine sekä Flatten Tree -komponenttien avulla saatiin sovittua yhteen molemmat parametrit, jossa oli teräsrakenteen tarvitsema data mitoille.



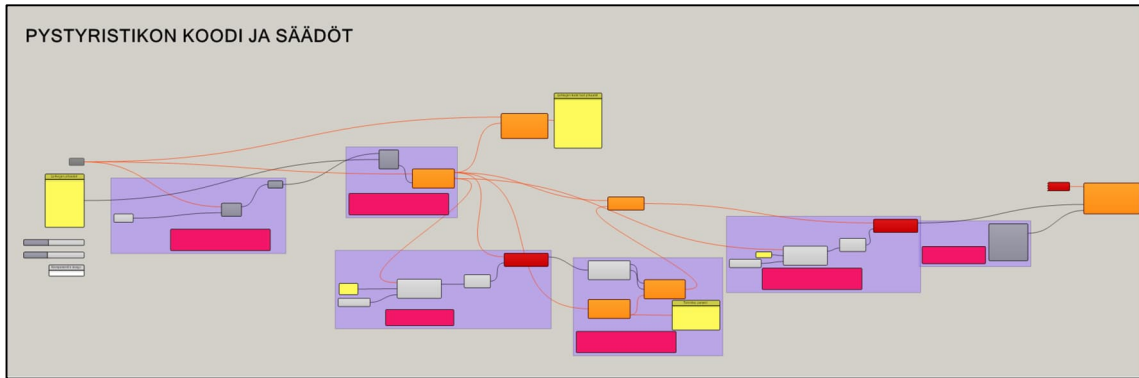
KUVA 12. Algoritmi teräsrakenteen mitoille, jossa on hyödynnetty Excel laskentaa

6.2 Pystyristikon mallintaminen

Kun tiedossa oli toimiva algoritmi teräsrakenteiden mitoille, lähdettiin kehittämään pystyristikon mallintamiseen tarvittavaa algoritmia. Koska opinnäytetyön ideana oli kehittää valmiina olevaa algoritmia toimimaan paremmin ja monipuolisemmin, asetti se myös omat rajauksensa algoritmin jatkokehitykselle. Pystyristikon algoritmin kehittäminen alkoi siitä, että otettiin aiemmin kehitetyllä mittaparametrien automatisointi algoritmilla valmiit lohkojen mitat pystyristikolle.

Pystyristikkoita sijaitsee teräsrakenteessa lähemmäs 100 kappaletta, ja pääkannattimien mitat laippojen ja uumien kohdalta muuttuvat teräsrakenteessa. Tämän takia myös poikkiristikon mitat muuttuvat sillan matkalla. Teräsrakenteessa on kahdenlaisia ristikoita ja niiden välille tulee vaakadiagonaalit, jotka kiinnitetään kiinnityslevyillä ristikoihin.

Kuvassa 9 näkyvää pystyristikon tyyppiä oli aiemmin käytetty onnistuneesti toisessa projektissa, ja siitä oli luotu oma komponentti Teklan avulla. Grasshopperiin voi tuoda Teklassa olevia valmiita komponentteja Live Linkin avulla. Ideana oli siis tuoda toimiva komponentti Teklaan ja laittaa se kulkemaan TSV-kaaren mukaisesti syöttämällä oikeat mitat pystyristikoiden sijainnille kaarella. Kuvassa 13 on pystyristikon mallintamiseen kehitetty algoritmi.



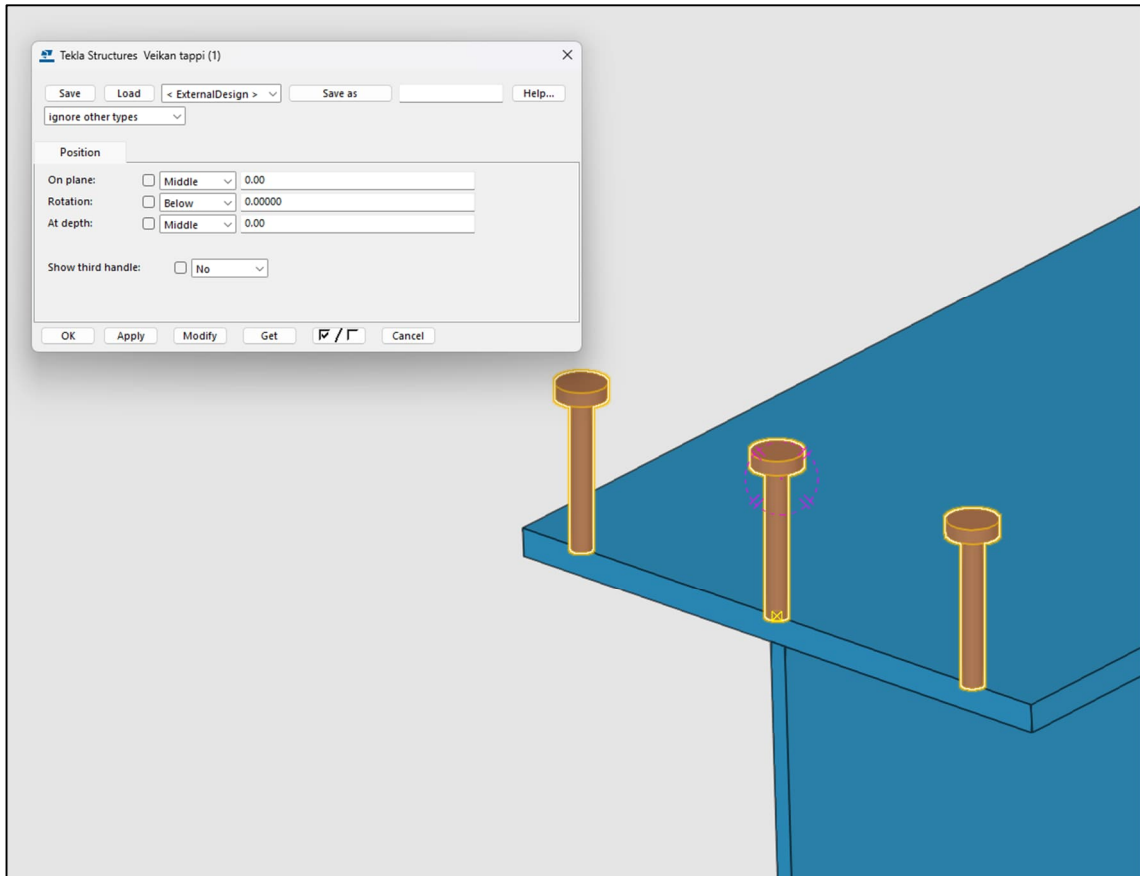
KUVA 13. Pystyristikon mallintamiseen tarvittava algoritmi

Tämä ratkaisu aiheutti kuitenkin manuaalisia töitä, sillä jokainen komponentti piti säätää vielä komponentin asetuksista oikean kokoisiksi, vaikka se sijaitsi kaarella oikeassa kohtaa. Pystyristikon mallintaminen Grasshopperin avulla on mahdollista, mutta se aiheuttaa paljon manuaalista työtä myös valmista algoritmia käytettäessä.

6.3 Vaarnatappien mallintaminen pääkannattimen ylälaippaan

Kun pystyristikon algoritmi oli kehitetty toimivaksi, siirryttiin sen jälkeen vaarnatappien algoritmi kehittämiseen. Vaarnatapit ovat hitsattavat pultit teräsrakenteen ylälaipan yläpintaan, joiden tehtävänä on kiinnittää betoninen kansi teräsrakenteeseen. Teräsrakenteen yleispiirustusta katsoessa havaittiin, että vaarvoja kulkee koko sillan matkalla kolmessa rivissä ja ne ovat samankokoiset koko sillan matkalla. Ainoastaan lohko- ja jakoväli muuttuivat.

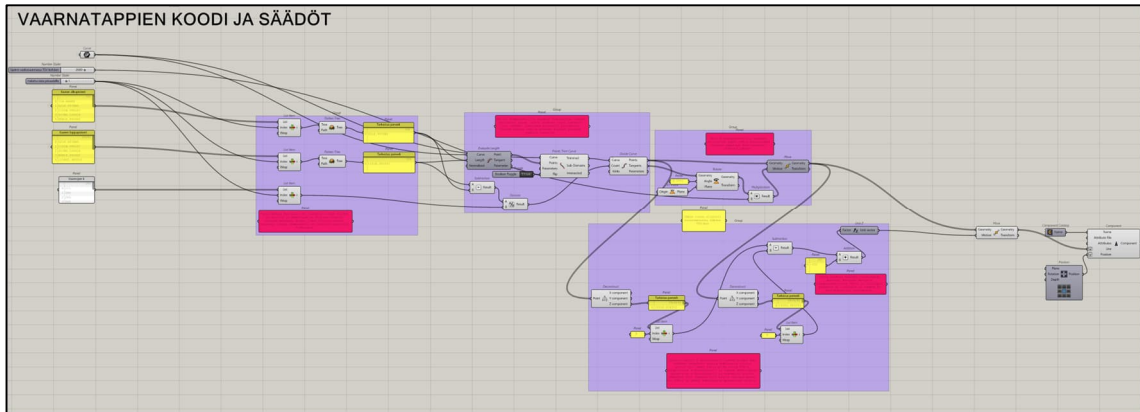
Grasshopperissa itsessään ei ole mahdollisuutta mallintaa pultteja, jotka kuvastaisivat vaarvoja. Tämän takia Teklaan mallinnettiin kaksi beam-objektia, jotka muotoiltiin näyttämään vaarnatapeilta. Kun vaarnatappi oli muotoiltu, kopioitiin niitä kolme vierekkäin oikeilla väleillä, jonka jälkeen tehtiin niistä oma komponentti. Kuvassa 14 nähdään vaarnatapeista tehty komponentti.



KUVA 14. Vaarnatapeista tehty komponentti

Vaarnatapeista tehty komponentti tuotiin Grasshopperiin Live Linkin avulla. Vaarnatapeille katsottiin mittaparametrien automatisointi algoritmilla oikeat lohkot kohdilleen. Algoritmi toimi vaarnoille yksittäin, eikä lohkojen oikeisiin mittoihin tarvittu Exceliä avuksi.

Lohkoille etsittiin kaarelta alkupiste ja siirryttiin lohkon mitan verran eteenpäin kaarella, jolloin saatiin loppupiste lohkoille. Teräsrakenteen yleispiirustuksesta saatu vaarnatappien jakoväli tuotiin lohkon alku- ja loppupisteiden kanssa Grasshopperiin omille paneeleilleen. List Item -komponentin avulla saatiin Number Slider -komponenttia käyttäen haettua oikeat sarakkeet yksittäisen vaarnalohkojen tekoon. Vaarnatappien jakovälin data jaettiin lohkon pituudella, jolloin saatiin oikea määrä vaarnoja kulkemaan lohkoilla. Kuvassa 15 nähdään vaarnatapeille tehty algoritmi.



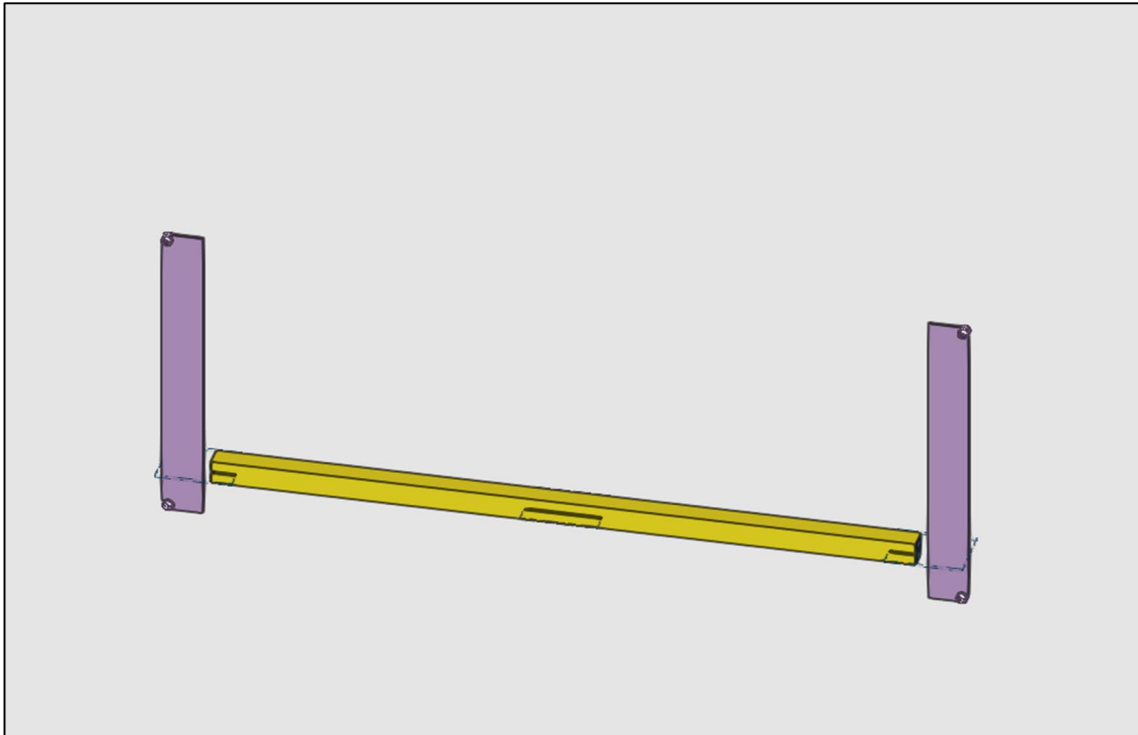
KUVA 15. Vaarnatappien algoritmi

Isoin haaste oli saada vaarnalohkot kulkemaan oikeassa paikassa, sekä oikealla korkeudella. Vaarnatappien sijainti määrättiin siirtämällä sitä kohtisuorassa TSV:tä nähden. Jotta vaarnatappi saatiin kulkemaan kohtisuorassa, piti TSV projisoida XY-tasoon ja siirtää TSV kohtisuoraksi.

Kun vaarnatappi saatiin liikkumaan kohtisuorasti TSV:tä nähden huomattiin, että tappien korko muuttuu merkittävästi niitä liikuttaessa. Oikea korkeus vaarnatappille saatiin ottamalla TSV:n kohdalla kulkevasta lohkoista Z-koordinaatti sekä siirretystä lohkoista Z-koordinaatti. Nämä vähentämällä toisistaan sekä vähentämällä mitta TSV:stä ylälaipan pintaan saatiin vaarnatappi paikalleen.

6.4 Välistikoiden mallintaminen

Ristikkoja on kaksi erilaista teräsrakenteessa. Välistikosta oli myös valmis komponentti aiemmasta projektista, jota hyödynnettiin tämän algoritmin teossa. Välistikko eroaa poikkiristikosta siten, että se on paljon kevyempi rakenteeltaan. Esimerkiksi välistikossa ei ole ollenkaan diagonaaleja, kun taas poikkiristikossa on. Kuvassa 16 nähdään Teklan Custom Component -työkalulla tehty välistikko.

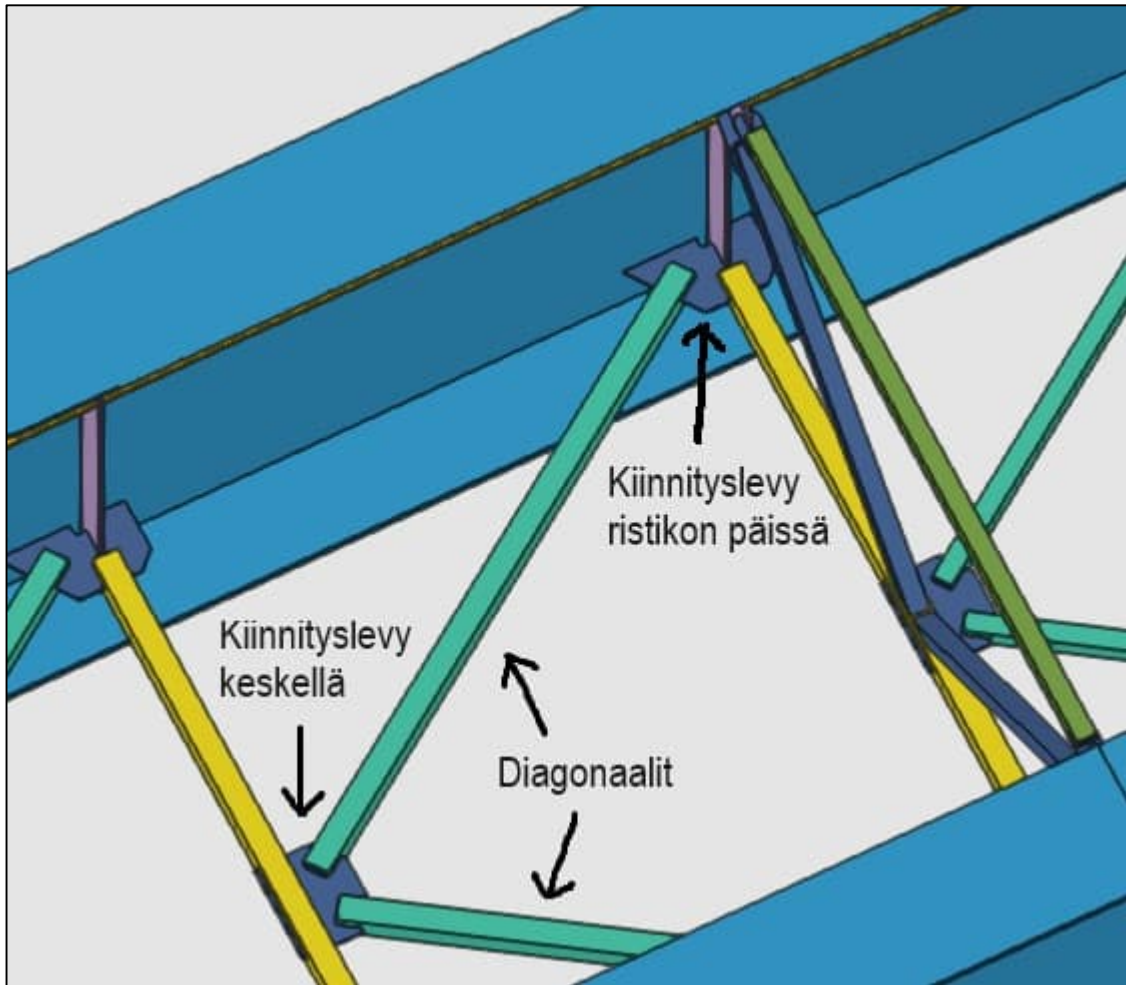


KUVA 16. Väliristikko

Väliristikolle haettiin myös omat mitat teräsrakenteen yleispiirustuksesta. Väliristikon algoritmi it-sessään ei eroa kummemmin poikkiristikoon tehdystä algoritmista. Joitain pieniä parannuksia algoritmin tehtiin, jotta sitä olisi helpompi ymmärtää.

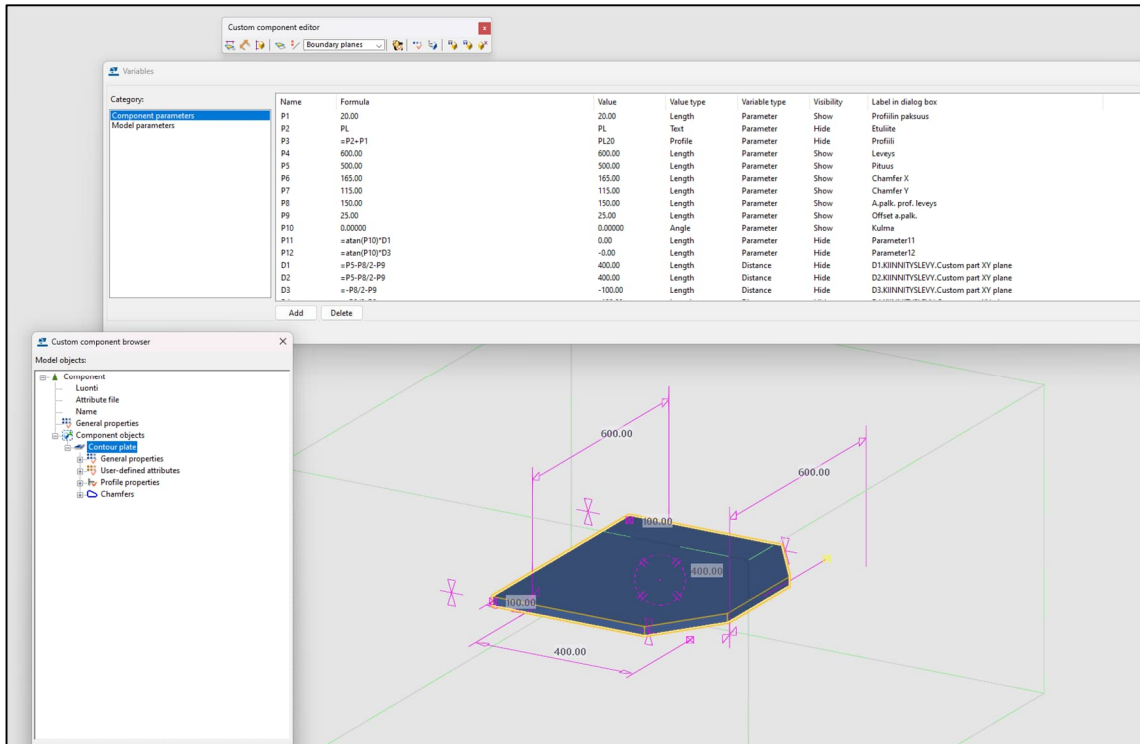
6.5 Kiinnityslevyt

Ristikoiden välille tulee ristikoita yhdistämään vaakadiagonaali terässauvat. Nämä kiinnitetään toisiinsa kiinnityslevyjen avulla. Kiinnityslevyjä on kahdenlaisia teräsrakenteessa. Kuvassa 17 nähdään Vekaransalmen sillassa käytetty teräsristikko, joka on hyvin samanlainen kuin Huikun siltaan tuleva. Kuvassa näkyy kiinnityslevyt ja diagonaalit.



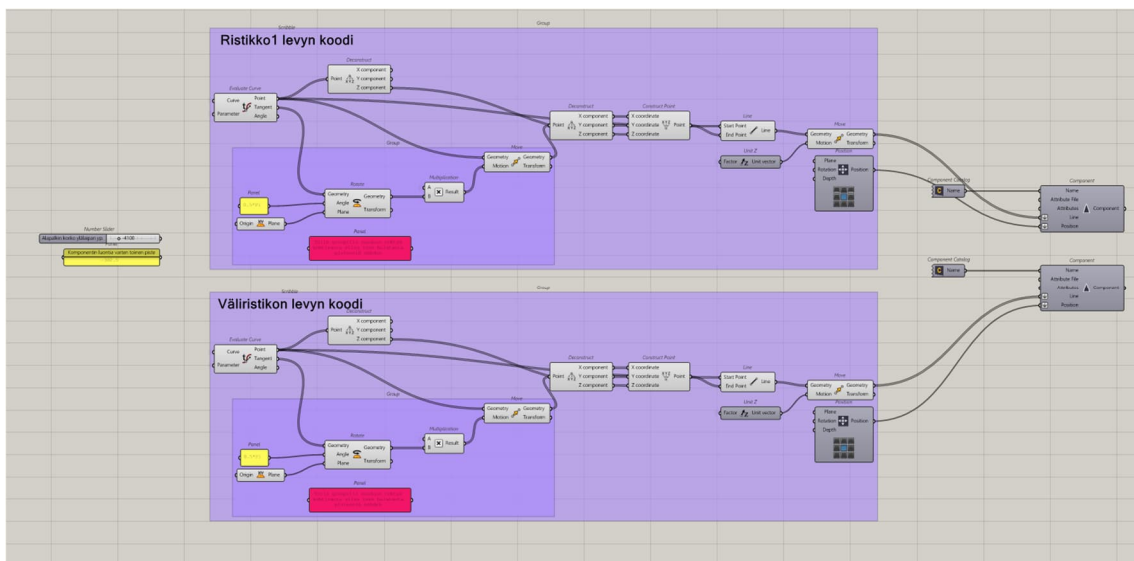
KUVA 17. Vekaransalmen sillan teräsrakenteen osia

Grasshopperilla itsessään voi mallintaa plate-objekteja, mutta niissä on tiettyjä puutteita. Esimerkiksi plate-objektiin ei voi laittaa nurkkapisteisiin viisteitä. Tästä syystä kiinnityslevyistä rakennettiin omat komponentit Teklan Custom Component -työkalulla. Kuvassa 18 nähdään tehty komponentti, joka sisältää erilaisia riippuvuussuhteita ja parametrejä. Esimerkiksi levyn paksuutta ja mittoja voidaan muuttaa.



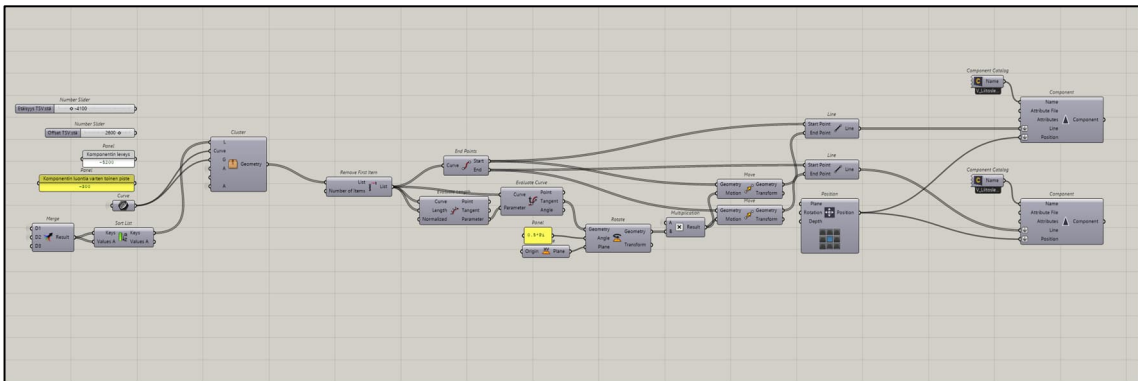
KUVA 18. Teklan Custom Component -työkalulla tehty parametrinen komponentti

Kun kiinnityslevyistä oli tehty valmiit komponentit, alettiin suunnittelemaan niille omaa algoritmia. Kiinnityslevyjen komponentti luotiin siten, että se tarvitsee kaksi pistettä, jonka mukaan levy luodaan. Koska kiinnityslevyt sijaitsevat keskellä ristikköä ja sen päissä, voidaan hyödyntää aiemmin kehitettyjen ristikoiden algoritmeja. Kuvassa 18 nähdään ristikon keskellä sijaitsevan kiinnityslevyn algoritmi.



KUVA 19. Keskellä sijaitsevan kiinnityslevyn algoritmi

Keskellä sijaitsevan kiinnityslevyn luontiin hyödynnettiin poikkiristikon luomisessa käytettyjä pisteitä. Siirtämällä pisteitä sillan poikkisuunnassa ja yhdistämällä pisteet saatiin mallinnettua keskimäinen kiinnityslevy. Kiinnityslevyjen algoritmeihin liittyy pieniä eroavaisuuksia, koska komponenttien sijainti on erilainen verrattuna toisiinsa. Kuvassa 20 nähdään ristikon päädyissä sijaitsevien kiinnityslevyjen algoritmi.

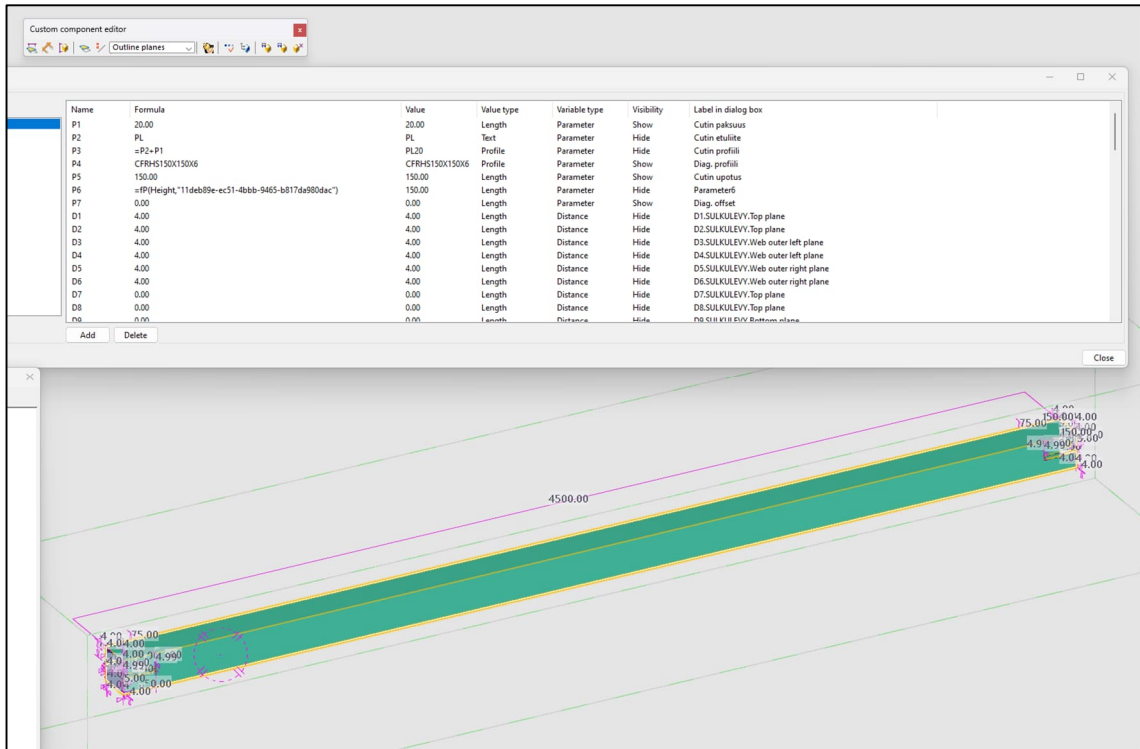


KUVA 20. Päädyissä sijaitsevien levyjen algoritmi

Päädyissä sijaitsevien kiinnityslevyjen algoritmiin käytettiin myös poikkiristikon luontiin käytettyjä komponentteja. Poikkiristikon luontiin käytetystä viivasta otettiin End Points -työkalun avulla viivan alku- ja loppupiste ja niitä liikutettiin hieman sillan pituus suunnassa taaksepäin. Yhdistämällä alkuperäisen viivan pisteet ja siirretyt pisteet, saatiin päädyn kiinnityslevyt mallinnettua.

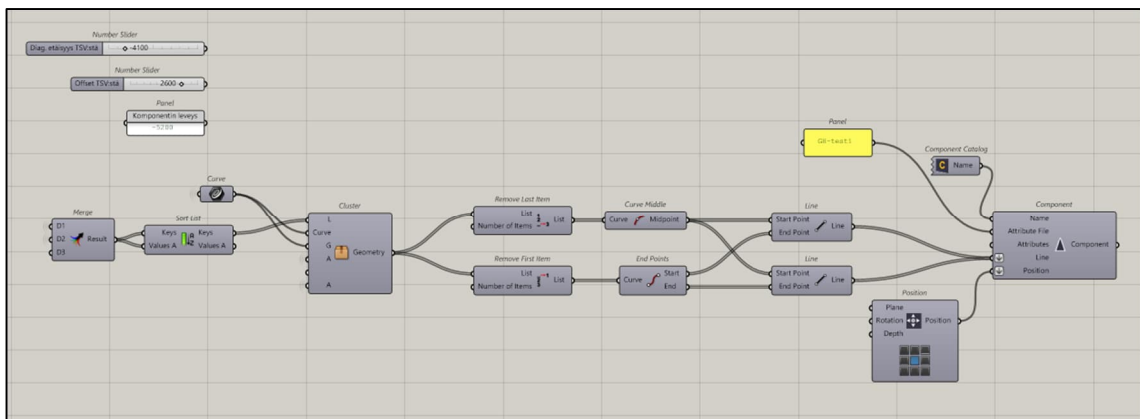
6.6 Vaakadiagonaalit

Kun ristikot mallinnettiin paikalleen ja kiinnityslevyt saatiin myös mallinnettua, olivat seuraavaksi vuorossa diagonaalit. Diagonaaleista tehtiin myös oma komponentti, sillä diagonaalien päässä on pienet levyt, jotka toimivat tulppana ontolle teräsauvalle. Ne piti sitoa kiinni sauvaan, sillä jos sauvan pituus muuttuu, eivät ne seuraa sauvan geometriaa, jos niitä ei ole komponentin avulla sidottu. Kuvassa 21 nähdään diagonaalista tehty komponentti, jossa tulpat on sidottu diagonaalien päihin.



KUVA 21. Diagonaalisauvasta tehty komponentti

Kun diagonaaleista oli tehty toimiva komponentti, alettiin niille rakentamaan omaa algoritmia Grasshopperiin. Diagonaalien algoritmi pohjautuu myös jo aiemmin kehitettyjen ristikoiden algoritmiin, sillä ne yhdistyvät ristikoiden välille. Kuvassa 22 nähdään diagonaaleille tehty algoritmi Grasshopperiin.



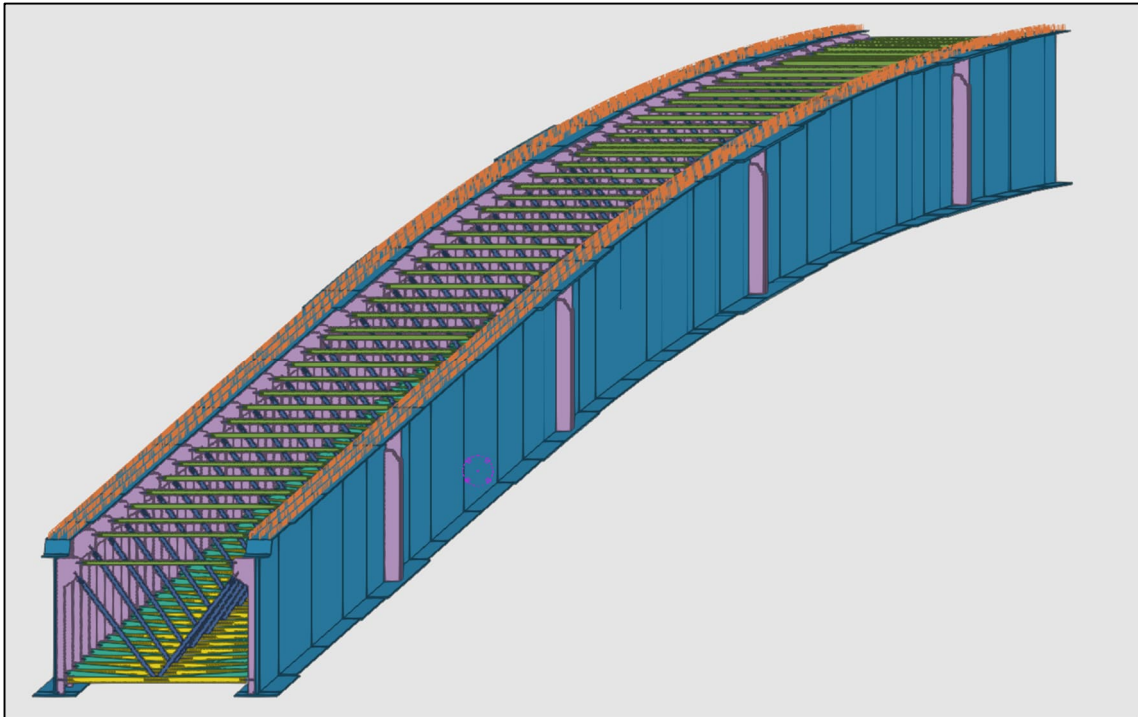
KUVA 22. Diagonaalien algoritmi

Diagonaalien algoritmin kehitys lähti siitä, että molempien ristikoiden data yhdistettiin Merge -työkalun avulla. Sen jälkeen data järjestettiin pituusjärjestykseen TSV-kaaren matkalla. Järjestetty

data syötettiin poikkiristikon luonnissa käytettyyn komponenttiryhmään, jolla saatiin tehtyä poikkiristikon luomisessa käytetyt viivat. Näistä viivoista haettiin sitten End Points -työkalun avulla viivan loppupisteet, ja Curve Middle -työkalulla viivan keskipiste. Kun pisteiden välille luotiin viiva ja se yhdistettiin Component -työkaluun, jossa oli valmiina diagonaalien komponentin data, saatiin mallinnettua diagonaalit paikalleen.

6.7 Valmis teräsrakenne

Kun kaikista tarvittavista osista oli tehty Grasshopperin avulla algoritmi, voidaan mallintaa valmis teräsrakenne. Kuvassa 23 nähdään valmis teräsrakenne, kun valmiit algoritmit on laitettu päälle ja yhdistetty Teklaan.



KUVA 23. Valmis teräsrakenne

Kun malliin oli kaikki tarvittava mallinnettu, tarkistettiin komponenttien toimivuus säätämällä ne oikean kokoisiksi katsomalla mitat teräsrakenteen yleispiirustuksesta. Komponentit toimivat hyvin. Tämän jälkeen valmis teräsrakenne kopioitiin toiselle puolelle Copy To Another Plane -toiminnolla Teklassa.

7 POHDINTA

Opinnäytetyön ideana oli kehittää Hailuodon kiinteä yhteys -projektiin kuuluvan S1 Huikun sillan pääkannattimille tehtyä Grasshopper algoritmia, jonka päätarkoitus on vähentää hankkeeseen käytettävää aikaa tietomallintamisessa. Työ osoittautui haastavaksi, kun käytetään täysin uutta menetelmää tietomallintamisessa.

Grasshopper itse ohjelmistona on suhteellisen helppokäyttöinen. Sen käyttöliittymän oppii tuntemaan kuten kaikkien muidenkin uusien ohjelmistojen käyttöliittymät. Ongelma Grasshopperin käytössä on, että osaa rakentaa toimivan algoritmin, jota voidaan hyödyntää tietomalleissa. Grasshopper tarjoaa hankkeisiin hyvin kattavat resurssit tehdä uusia algoritmeja, mutta tietoa kuinka rakentaa niitä on tarjolla rajoitetusti.

Opinnäytetyön tavoite saavutettiin, ja aiemmin tehtyä algoritmia pääkannattimille saatiin kehitettyä paremmaksi. Sen lisäksi saatiin myös kehitettyä algoritmiin kuusi täysin uutta ominaisuutta. Uusia ominaisuuksia algoritmissa ovat mittaparametrien automatisointi, teräsristikot, vaarnatapit, kiinnityslevyt ja diagonaalit. Algoritmi itsessään on vielä hieman keskeneräinen ja luultavasti tulevaisuudessa sitä tullaan kehittämään pidemmälle.

Kun verrataan Grasshopperin avulla mallintamista manuaalisen mallintamiseen voidaan huomata, että Grasshopperilla tehty mallinnus on tehokkaampi ja tarkempi. Manuaalisesti mallintamiseen käytetty aika on huomattavasti enemmän ja lisäksi se on virhealttiimpaa. Muutosten tekeminen malliin on hitaampaa manuaalisesti kuin Grasshopperilla. Grasshopperin avulla rakennettu malli on hyvin tarkka, koska sen toiminta perustuu matematiikkaan ja algoritmeihin, jotka eivät tee virheitä toisin kuin ihmiset.

Ohjelmiston käyttöönotto vaatii myös organisaatiolta resursseja. Suunnittelutoimisto, joka haluaa pysyä ajan tasalla alan uusista suuntauksista, täytyy olla myös muutoksenhaluinen. Täytyy olla riittävä määrä resursseja, jotta työntekijöitä voidaan kouluttaa käyttämään uusia ohjelmistoja. Ohjelmistot maksavat ja niiden käyttöönotto ja opetteleminen vie aikaa ja samalla rahaa.

Parametrisen suunnittelutyökalun käyttäminen jossain pienemmissä projekteissa ei välttämättä ole järkevää. Taitava ja kokenut tietomallintaja voi onnistua mallintamaan kattavan tietomallin nopeammin verrattuna parametrisesti mallintaan.

Grasshopperin edut tulevat näkyviin siinä vaiheessa, kun hankkeen aikana tulee joitain muutoksia, kuten esimerkiksi mitoituksen kautta saadaan uutta tietoa, ja jotain kohtaa tietomallissa pitää muuttaa. Lisäksi valmiina olevia algoritmeja voidaan hyödyntää tulevissa samankaltaisissa hankkeissa, jolloin resursseja säästyy vielä enemmän, sillä Grasshopperiin itsessään ei tarvitse siinä vaiheessa käyttää niin paljon resursseja. Grasshopperin käytön etuja on myös se, että valmiiden algoritmipohjien käyttäminen ei vaadi käyttäjältä erityistaitoja.

Kaiken kaikkiaan Grasshopperin hyödyntäminen tietomallintamisessa avaa uusia mahdollisuuksia suunnitteluprosesseissa, edistäen innovaatiota sekä helpottaen monimutkaisten projektien hallintaa. On kuitenkin selvää, että vaikka tämä työ olikin haastava, maksaa se vaivansa takaisin tulevissa hankkeissa, joissa hyödynnetään parametrista tietomallintamista.

LÄHTEET

Afry 2023. Kuvakaappaus. S1 Huikun silta. Hakupäivä 2.2.2024.

Ball, Alex 2013. Preserving Computer-Aided Design (CAD). DPC Technology Watch Report. Hakupäivä 2.2.2024 <https://www.dpconline.org/docs/technology-watch-reports/896-dpctw13-02-pdf/file>

Carpó, Mario 2023. A short but believable history of the digital turn in architecture. Hakupäivä 9.1.2024 <https://www.e-flux.com/architecture/chronograms/528659/a-short-but-believable-history-of-the-digital-turn-in-architecture/>

Davis Daniel 2013. A History of Parametric. Hakupäivä 28.2.2024 <https://www.danieldavis.com/a-history-of-parametric/>

Hirvikoski, Matias & Karjalainen, Petteri 2021. Grasshopper-koulutus, Basic. A-insinöörit Oy. Sisäinen koulutus.

Nova Design 2024. A Brief History in Time – The History of Computer Aided Design – CAD. Hakupäivä 9.1.2024 <https://nova-design.co.uk/history-of-computer-aided-design/>

Tamminen, Seppo 2006. Reikäkorteista tuotemallinnukseen – 40 vuotta Teklan ohjelmistoja. Tekla Oyj 1966–2006.

Tanska, Tuulikki & Österlund, Toni 2014. Algoritmit puurakenteissa: menetelmät, mahdollisuudet ja tuotanto. DigiWoodLab, Oulun yliopisto, Arkkitehtuurin tiedekunta.

Tedeschi, Arturo 2011. Parametric architecture with Grasshopper. Italia: Le Penseur.

Tekla 2024. The benefits of parametric design for structural engineers. Hakupäivä 28.2.2024 <https://www.tekla.com/resources/articles/the-benefits-of-parametric-design>

Su Aung, Thet Hnin 2022. 4 Pros and Cons of Parametric Modelling. Novatr.com. Hakupäivä 28.2.2024 <https://www.novatr.com/blog/pros-and-cons-parametric-modeling>

Väylävirasto 2024. Hailuodon kiinteä yhteys. Hakupäivä 15.3.2024 <https://vayla.fi/hailuoto>

Väylävirasto 2024. Hailuodon kiinteä yhteys -hankkeen vesilupa sai lainvoiman – rakentaminen käynnistyy. Hakupäivä 15.3.2024 <https://vayla.fi/-/hailuodon-kiinteä-yhteys-hankkeen-vesilupa-sai-lainvoiman-rakentaminen-kaynnistyy>

Väylävirasto 2020. Mikä on tietomalli? Hakupäivä 22.11.2023 <https://vayla.fi/palveluntuottajat/inf-ramallit/mika-on-tietomalli->

Wojslaw, Krzysztof 2023. Webinaari. Automating Tekla modelling and connecting to structural analysis software for bridges using grasshop. Tekla.com. Hakupäivä 10.1.2024 <https://www.tekla.com/sg/resources/webinars/automating-tekla-modelling-and-connecting-to-structural-analysis-software-for-bridges-using-grasshop#main-content> Kuvakaappaus kohdasta 00:09:33

Wojslaw, Krzysztof 2023. Start with Grasshopper in Tekla – A comprehensive guide. Hakupäivä 29.2.2024 [Start with Grasshopper in Tekla - A comprehensive guide. \(notion.site\)](#)

