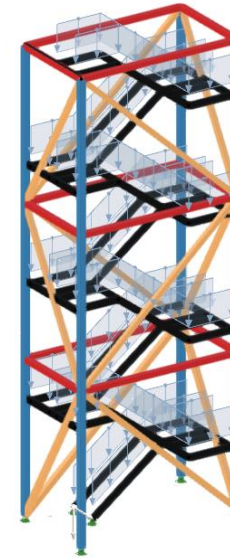


Parametrisk design och miljöoptimering av industritrapptorn



Sebastian Ståhl & Victor Teinert

Avdelningen för Konstruktionsteknik
Lunds Tekniska Högskola
Lunds Universitet, 2024

Avdelningen för Konstruktionsteknik

Lunds Tekniska Högskola

Box 118

221 00 LUND

Division of Structural Engineering

Faculty of Engineering, LTH

P.O. Box 118

S-221 00 LUND

Sweden

Parametrisk design och miljöoptimering av industritrapptorn

Parametric design and environmental optimization of industrial staircase towers

Sebastian Stål & Victor Teinert

2024

Rapport TVBK- 5302

ISSN 0349-4969

ISRN: LUTVDG/TVBK- 24/5302-SE

Examensarbete

Handledare: Ivar Björnsson (LTH), Samir Filander (Sweco)

Maj 2024

Förord

Detta examensarbete sätter punkt för vår civilingenjörsutbildning inom Väg- och Vattenbyggnad vid Lunds Tekniska Högskola. Arbetet har varit ett samarbete mellan avdelningen för konstruktionsteknik på LTH och region Industri på Sweco, där arbetet också har genomförts.

Vi vill rikta ett stort tack till våra handledare Samir Filander och Baptiste Woerli på Sweco som tålmodigt och engagerat väglett oss genom arbetet. Även vår handledare på LTH, Ivar Björnsson, ska ha ett stort tack för vägledning och feedback under arbetets gång.

Stockholm, Maj 2024.

Sebastian Ståhl & Victor Teinert

Sammanfattning

Att byggsektorn står inför ett rådande klimatnödläge är ett faktum. Idag investeras hundratals miljarder kronor för att i framtiden kunna producera fossilfritt stål; trots det läggs lite fokus på miljöoptimering av konstruktioner. Syftet med arbetet är att utvärdera hur parametrisk design kan användas som verktyg för att miljöoptimera stålstommar i industritrapptorn samt undersöka vilken profiltyp som ger minst klimatavtryck.

Parametrisk design innebär att man med hjälp av variabler och designparametrar bygger upp geometrin för konstruktioner. Genom att ändra de parametriserade variablerna kan man enkelt ändra en konstruktions geometri. Trapptorn är konstruktioner med hög redundans och geometrin har en enkel matematisk beskrivning, vilket gör det relativt enkelt att använda parametrisk design för den typen av konstruktion.

Arbetet tar enbart hänsyn till miljöpåverkan av trappornets bärande stålstomme och dess rostskyddsmålning. De profiltyper som analyseras är kvadratiska kallformade konstruktionsrör (KKR), kvadratiska varmformade konstruktionsrör (VKR) och HEA-profiler. HEA-profiler har ett lägre klimatavtryck per kg stål jämfört med konstruktionsrören men har en större mantelarea, vilket ökar åtgången av rostskyddsmålning.

Huvuddelen av arbetet gick ut på att skapa ett skript i det parametriska verktyget Grasshopper. Skriptet använder parametrisk design för att automatisera uppbyggandet av trappornsmodeller. Användaren anger indata som till exempel våningshöjder och lutning på trappan, därefter bygger skriptet upp en trappornsgemetri som förhåller sig till relevanta standarder och byggregler.

Grasshopper-skriptet innehåller även en koppling till FEM-programvaran RFEM 6 samt den genetiska optimeringsalgoritmen Galapagos. All data som krävs för att utföra en FEM-analys skapas i Grasshopper och exporteras sedan till RFEM 6 där analyserna utförs. Resultatet importerar sedan tillbaka till Grasshopper och utvärderas av Galapagos. Galapagos kan därefter påbörja en ny optimeringsiteration genom att ändra på de parametriserade variablerna och skapa en ny trappornskonstruktion med annorlunda geometri och tvärsnitt. Denna slutna process fortlöper tills Galapagos har hittat en lösning med så lågt klimatavtryck som möjligt, utan att utnyttjandegraden överstiger 100%.

För att utvärdera skriptets funktionalitet görs ett försök att återskapa ett redan projekterat och dimensionerat projekt. Skriptet lyckades återskapa trappornet med nästintill identisk geometri och stålvikt. Sedermera optimerades konstruktionen med tre olika profiltyper, KKR, VKR och HEA. Resultatet från optimeringen visar att en trappornsstomme med stängda profiler blir både lättare och får mindre miljöpåverkan. Mellan de stängda profilerna är skillnaden nästan försumbar. Skriptet lyckades även med hjälp av optimeringen minska referensprojektets utsläpp av koldioxidekvivalenter med drygt 40%.

Abstract

The fact that the construction sector faces a prevailing climate emergency is undeniable. Today, hundreds of billions of kronor are being invested to develop fossil-free steel; yet little focus is placed on environmental optimization of constructions. The purpose of this work is to evaluate how parametric design can be used as a tool for environmental optimization of steel staircase towers, as well as to investigate which type of profile provides the least climate impact.

Parametric design means that geometry of the structure is built up using variables and design parameters. By changing the parametrized variables, the geometry can be easily altered. Staircase towers are constructions with high redundancy, and the geometry has a simple mathematical description. This makes it relatively easy to use parametric design for this type of structure.

The work only considers the environmental impact of the load bearing elements and its corrosion protection paint. The profile types analysed are cold-formed structural tubes (CFRHS), square hot-formed structural tubes (HFRHS), and HEA profiles. HEA profiles have a lower climate footprint per kilogram of steel compared to the structural tubes but have a larger surface area, which increases the amount of corrosion protection paint required.

The main part of the work involved creating a script in the parametric tool Grasshopper. The script uses parametric design to automate the creation of calculation models for the towers. The user enters data such as floor heights and the inclination of the stairs, and then the script builds up a stair tower geometry that adheres to relevant standards and building regulations.

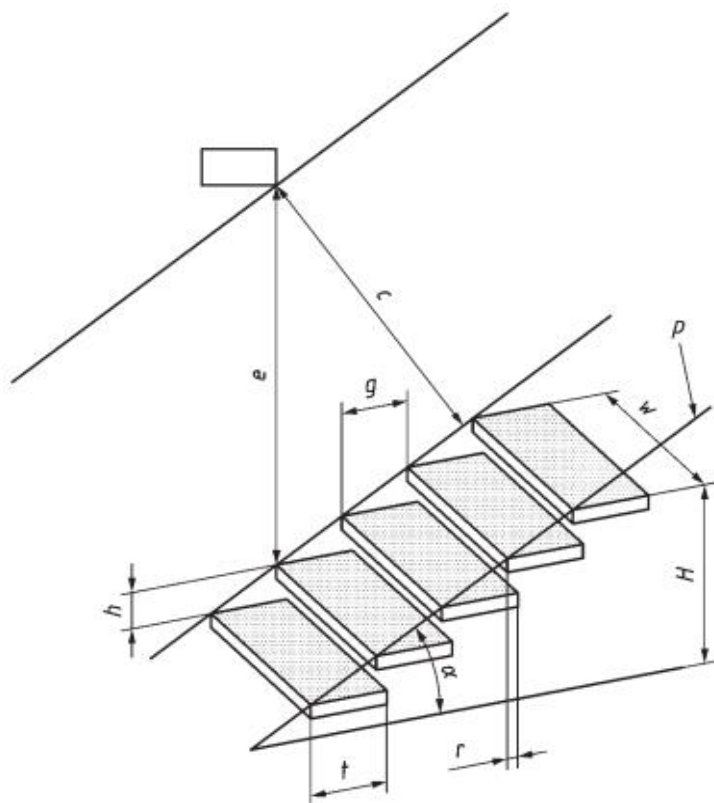
The Grasshopper script also includes a link to the FEM software RFEM 6 and the genetic optimization algorithm Galapagos. All the data required to perform an FEM analysis is created in Grasshopper and then exported to RFEM 6, where the analyses are performed. The result is then imported back to Grasshopper and evaluated by Galapagos. Galapagos can then start a new optimization iteration by changing the parametrized variables and creating a new model with different geometry and cross-section. This closed process continues until Galapagos finds a solution with minimized climate footprint, without exceeding the load bearing capacity.

To validate the script, it is applied to an existing construction project of a steel staircase tower. The script succeeded in recreating the stair tower with almost identical geometry and steel weight as the original project. Subsequently, the structures were optimized with three different profile types, CFRHS, HFRHS, and HEA. The results from the optimization show that a structure with closed profiles becomes both lighter and has less environmental impact. The difference between the solution with closed profiles is almost negligible. The script also succeeded, with the help of optimization, in reducing the reference project's emissions of carbon dioxide equivalents by just over 40%.

Förkortningar och variabler

KKR	Kallformade konstruktionsrör
VKR	Varmformade konstruktionsrör
GWP	Global Warming Potential
Π_{GWP}	Summa GWP
W_s	Stålvikt [kg]
A_m	Mantelarea [m ²]
CO_{2e}	Koldioxidekvivalenter [kgCO _{2e}]
$\text{CO}_{2e,s}$	Koldioxidekvivalenter per kg stål
$\text{CO}_{2e,r}$	Koldioxidekvivalenter per m ² rostskydd
U_i	Utnyttjandegrad
LCA	Livscykelanalys
E_d	Dimensionerande last
R_d	Hållfasthet
$w_{\text{tot},i}$	Deformation i godtycklig riktning
w_{lim}	Deformationskrav

Trappans olika delar och benämningar



- H – trapphöjd
- g – stegdjup
- e – fri höjd
- h – steghöjd
- r – stegöverlappning
- α – lutningsvinkel
- w – stegbredd
- p – stigningslinje
- t – stegdjup på trappsteg
- c – fritt utrymme

trapplöp – oavbruten serie trappsteg mellan två viloplan

fri höjd – minsta vertikala avstånd ovanför stigningslinjen

viloplan – horisontell avsats som avslutar ett trapplöp

stegöverlappning – skillnad mellan djupet på trappsteget och stegdjup

stigningslinje – teoretisk linje som knyter ihop trappstegen

lutningsvinkel – vinkel mellan stigningslinje och dess projektion på horisontalplanet

steghöjd – höjd mellan två varandra följande trappsteg, mätt från en stegyta till en annan

trappsteg – horisontell yta på vilken man placerar foten på

vangstycke – sidodel som bär upp trappstegen

stegbredd – fritt avstånd mellan de yttre kanterna på ett trappsteg

stegdjup på trappsteg – fritt avstånd mellan trappstegets främre och bakre kant

fritt utrymme – minsta fria avstånd mellan varje hinder och stigningslinje

Innehållsförteckning

Förord	I
Sammanfattning	II
Abstract	III
Förkortningar och variabler.....	IV
Trappans olika delar och benämningar	V
1. Inledning.....	1
1.1 Bakgrund	1
1.2 Parametrisk design och optimering	2
1.3 Trapporn inom industri	3
1.4 Syfte	4
1.5 Avgränsningar	4
1.6 Disposition	4
2. Ett trapporns miljöpåverkan.....	7
2.1 Val av profiler	7
2.2 Rostskyddsmålning	9
2.3 Miljöoptimering	12
3. Optimeringsteori.....	13
3.1 Problemställning.....	13
3.2 Straffmetoden	13
3.3 Designvariabler och optimeringsalgoritmer.....	14
3.4 Naturinspirerade algoritmer	15
4. Parametrisk modellering och optimering i Grasshopper.....	17
4.1 Modellering i Grasshopper.....	17
4.2 Finit elementanalys i Grasshopper	18
4.3 Optimering i Grasshopper	18
5. Parametrisk design av trapporn.....	23

5.1	Trapptornets geometri	23
5.1.1	Bärande stomme	23
5.1.2	Trappornas utformning	26
5.2	RFEM 6 indata	27
5.2.1	Geometri och tvärsnitt	27
5.2.2	Laster	29
5.2.3	Ståldesign	31
5.3	Miljöoptimering av trapptorn	32
5.3.1	Designvariabler	32
5.3.2	Formulering av optimeringsproblemet	33
6.	Utvärdering av skriptet och optimeringsresultat	35
6.1	Garpenberg-projektet	35
6.2	Optimeringsresultat	38
7.	Diskussion och slutsats	41
7.1	Parametrisk design som verktyg vid miljöoptimering	41
7.2	Profilerna ur ett miljöperspektiv	41
7.3	Förbättringspotential och framtida forskning	42
	Referenser	44
	Bilagor	47
	Bilaga A	47
	Bilaga B	48
	Bilaga C	50
	Bilaga D	53
	Bilaga E	54

1. Inledning

Detta examensarbete ämnar utforska hur optimering kan samverka med parametrisk design för att minska stålkonstruktioners miljöavtryck. Den centrala delen av arbetet innefattar utvecklingen av ett skript för att automatisera projektering- och dimensioneringsprocessen av ett industritrapptorns bärande stålstomme. Skriptet, som drivs av en optimeringsalgoritm, har i uppgift att identifiera de mest miljöeffektiva dimensionerna på stålprofiler för konstruktionen. Detta möjliggör i sin tur en utvärdering av vilken typ av stålprofiler i trappornets stomme som är mest miljövänlig.

Idén till arbetet grundar sig i branschens varierande syn på öppna- och slutna stålprofiler där det råder blandade meningar om vilken profiltyp som är mest miljövänlig. Anledningen till denna oenighet tros bero på att svenska stålleverantörer enbart presenterar utsläpp från profilernas produktion på deras hemsidor, vilket inte rättvist reflekterar verkligheten. Arbetets tillvägagångssätt är inspirerat av Baptiste Woerlis (2019) tidigare examensarbete, där han med hjälp av ett skript framgångsrikt kombinerade parametrisk design med viktoptimering av rörbryggor.

1.1 Bakgrund

Byggsektorn står idag för 21% av världens koldioxidutsläpp där en betydande del kommer från stålanvändning (GlobalABC, 2024). Sveriges satsning att minska utsläppen från ståltillverkning är en aktuell debatt och det kommer investeras hundratals miljarder kronor för att i framtiden kunna producera fossilfritt stål¹ (Rawet, 2023). Om fossilfritt stål är en bra lösning på det rådande klimatnödläge byggsektorn står inför låter vi vara osagt, men oavsett är stålproduktion en energikrävande process (Jernkontoret, 2022). Det är därför viktigt att optimera stålkonstruktioner för att undvika onödig användning av stål.

Optimering av konstruktioner är ingen ny företeelse och antalet vetenskapliga artiklar inom ämnet har ökat explosionsartat det senaste decenniet. En litteraturstudie från University of Singapore analyserade 196 artiklar om strukturell optimering, av dessa fokuserade 62% på kostminimering och enbart 2% på minimering av miljöpåverkan (Mei & Wang, 2021). Trots byggsektorns betydande klimatavtryck och omfattande investeringar i klimatneutrala byggmaterial läggs få resurser på miljöoptimering och minimering av byggnaders klimatavtryck.

För att bedöma en konstruktions miljöpåverkan används oftast begreppet Global Warming Potential (GWP). GWP är ett mått på hur mycket en viss mängd av en växthusgas bidrar till global uppvärmning över en given tidsperiod, vanligtvis mätt i koldioxidekvivalenter (CO₂e). Det används för att jämföra olika växthusgaser och bedöma deras relativa påverkan på klimatförändringar. Koldioxid används som referenspunkt med ett GWP på 1 över en tidsperiod på 100 år. Andra växthusgaser, som metan och lustgas, har högre GWP-värden, vilket innebär att de har en större påverkan på uppvärmningen per enhet av utsläpp jämfört med koldioxid (USEPA, 2023). För att bedöma en produkts totala klimatpåverkan kan man utföra en livscykelanalys.

¹ Fossilfritt stål, eller grönt stål är stål som tillverkas utan användning av fossila bränslen.

Livscykelanalys (LCA), är ett instrument som bidrar till att främja miljömässig hållbarhet i byggsektorn. Genom LCA erhåller man insikter om en produkts miljöeffekter i flera kategorier, där klimatpåverkan (CO₂e) står i störst fokus för att bedöma en produkts miljöbelastning. En livscykelanalys är uppdelad i olika livscykelfaser och informationsmoduler, vilket illustreras i Figur 1.1. Denna uppdelning möjliggör att aktörer kan tillhandahålla LCA-data för specifika segment av byggprocessen eftersom tillverkare och leverantörer av byggmaterial och tjänster bidrar med specifik LCA-information för sina produkter. I sin tur kan den informationen sedan användas av den part som ansvarar för att sammanställa en heltäckande LCA för ett byggnadsprojekt. På så sätt kan detaljerad LCA-data för varje delkomponent integreras för att skapa en omfattande översikt över byggnadens miljöpåverkan under hela dess livslängd (Erlandsson, 2019).

Livscykelinformation byggnad														Återvinning – sammans- bedömning		
A 1-3 Produktskedet			A 4-5 Byggproduktion -skedet		B 1-7 Användningsskedet							C 1-4 Slutskedet			D Övrig miljöinfo	
A1 – Råvaruförsörjning	A2 - Transport	A3 - Tillverkning	A4 - Transport	A5 – Bygg- och installationsprocessen	B1 - Användning	B2 - Underhåll	B3 - Reparation	B4 - Utbyte	B5 – Renovering	B6 - Energianvändning	B7 - Vattenanvändning	C1 – Demotering, rivning	C2 - Transport	C3 - Restproduktshantering	C4 - Bortskaffning	Återanvändnings-, Återvinnings- & Materialåtervinnings- potential

Figur 1.1 Livscykelanalysens olika faser (IVL Svenska Miljöinstitutet AB, 2024).

- Produktskedet (A 1–3) tar hänsyn till komponentens miljöpåverkan vid tillverkning.
- Byggproduktionsskedet (A 4–5) representerar komponentens utsläpp vid bygg- och installationsprocessen.
- Användningsskedet (B 1–7) visar komponentens miljöavtryck under användning av byggnaden.
- Slutskedet (C 1–4) representerar komponentens utsläpp vid rivning av byggnaden.
- Övrig miljöinfo (D) presenterar övrig miljöinfo, som till exempel komponentens potential till återanvändning och återvinning.

En miljöoptimering av stålkonstruktioner skulle kunna innebära att man under projekteringen tar hänsyn till olika stålprofilers livscykel och minimerar konstruktionens totala GWP . De relevanta faserna att ta i beaktning till blir således produktskedet (fas A), slutskedet (fas C) och övrig miljöinfo (fas D). För att tidseffektivt kunna utföra denna optimering kan man i projekteringsfasen använda parametrisk design.

1.2 Parametrisk design och optimering

Parametrisk design är en metod inom design och arkitektur där man använder variabler och designparametrar för att påverka formen och funktionen hos ett objekt. Med hjälp av parametrisk design kan till exempel arkitekter och konstruktörer skapa, ändra och analysera

komplexa geometrier inom rimliga tidsramar. Dessa geometriska former kan sedan bilda konstruktioner som optimeras för specifika krav, såsom vikt, och dimensioneras där utefter (Toker, 2022).

En framstående förebild för parametrisk design var den spanske arkitekten Antoni Gaudí. Med hjälp av hängande kedjor belastade med vikter skapade Gaudí fysiska modeller där varje element utsätts för ren dragkraft. Med hjälp av en spegel kunde Gaudí invertera modellen och se hur konstruktionen skulle se ut när den enbart utsätts för ren tryckkraft. Genom att ändra de fysiska parametrarna, så som vikterna kunde Gaudí generera andra versioner av modellen. Gaudís arbete visar hur parametrisk design kan tillämpas fysiskt för att skapa konstnärliga och funktionella strukturer (Makert & Alves, 2016). Idag har de fysiska modellerna ersatts med datorprogram där man snabbt och effektivt kan bygga upp sina parametriska modeller. Dessa parametriska modeller kan i sin tur kopplas till optimeringsalgoritmer som genom att ändra på valda variabler kan optimera konstruktionen.

Parametrisk design och optimering av den parametriska modellen är särskilt användbar för strukturer med hög redundans där den matematiska beskrivningen är relativt enkel. Det minskar mängden parametriserade variabler vilket är fördelaktigt då fler variabler leder till en större sökrymd. Det innebär att det finns fler potentiella lösningar, vilket kan öka tiden det tar att hitta lösningar (Parker & Rardin, 1988). Det är därför essentiellt att begränsa de parametrar som har störst inverkan på designresultatet för att skapa en balans mellan modellens komplexitet och beräkningsbarhet. En konstruktion med hög redundans, relativt enkel matematisk beskrivning och få variabler vid optimering är trapporn inom industri.

1.3 Trapporn inom industri

Trapporn i industrisammanhang är en konstruktion som består av en serie trappor och plattformar som är konstruerade för att ge tillgång till olika nivåer inom en industriell anläggning. Ett trapporn kan vara en permanent struktur som är integrerad med en byggnad, eller en fristående enhet som kan flyttas vid behov. Permanenta trapporn har utöver trappor oftast en hiss för att underlätta transporten av material, verktyg och personal. Figur 1.2 presenterar ett trapporn under byggnation.

Trapporn används i många olika industriella miljöer, som till exempel raffinaderier, kraftverk, fabriker och lagerbyggnader. De är ofta tillverkade av stål för att tåla tuffa arbetsförhållanden och väderförhållanden då de ofta är placerade utomhus. För att uppfylla säkerhetsstandarder för personalen regleras trappornas utformning av byggnormer vilket gör att trapporn oftast har en liknande utformning.

Den bärande stommen i ett trapporn kan utföras med både slutna och öppna tvärsnitt. Fördelarna med slutna tvärsnitt, som till exempel kvadratiska konstruktionsrör är en enkelhet i design, mindre svets och mindre mantelyta vid rostskyddsmålning. Öppna tvärsnitt däremot,



Figur 1.2. Utanpåliggande industritrapporn under byggnation. Bild hämtad från Sweco.

som till exempel H-profiler, har fördelen att ha både lägre kostnad och miljöpåverkan per kilogram stål.

1.4 Syfte

Examensarbetet utforskar potentialen för parametrisk design som ett verktyg för miljöoptimering av stålkonstruktioner. Genom att använda denna metodik ämnar arbetet att utvärdera om öppna eller slutna tvärsnittsprofiler i ett industritrapptorns stomme är mer fördelaktiga ur ett miljöperspektiv med avseende på CO_2 -ekvivalenter från stålet och rostskyddsmålningens livscykel.

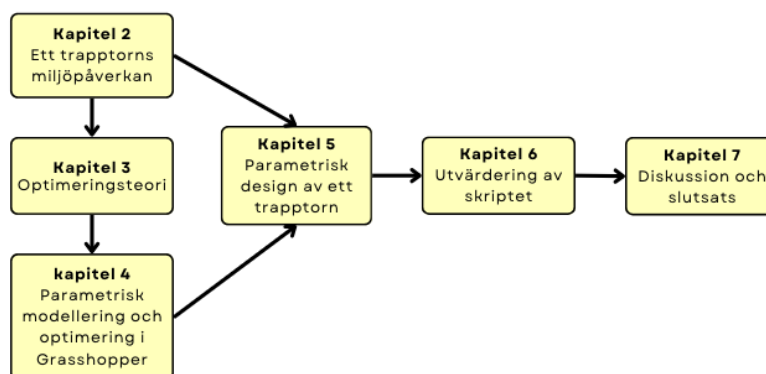
1.5 Avgränsningar

För att arbetet ska kunna genomföras inom tidsramarna har vissa avgränsningar tagits:

- Arbetet fokuserar på inklädda industritrapptorn med hisschakt som är placerade utomhus.
- Ekonomiska och tidsmässiga aspekter som kan påverka konstruktionerna kommer inte att beaktas.
- Enbart den bärande stålstommen och rostskyddsmålning kommer tas i beaktning vid miljöoptimeringen.
- De livscykel-faser som används vid beräkning av stommens utsläpp är produktskedet (fas A), slutskedet (fas C) och övrig miljöinfo (fas D).
- Urvalet av tvärsnittsprofiler som är tillgängliga för användning i optimeringsprocessen kommer att begränsas.
- FEM-analyserna kommer kontrollera brottgränstillståndet (ULS) och bruksgränstillståndet (SLS). Krav på jämvikt, som beräknas i enlighet med EQU, kommer inte att tas med i dessa beräkningar.

1.6 Disposition

För att förstå rapportens disposition och hur arbetets kommande kapitel hänger ihop presenteras ett flödesschema i figur 1.3 samt en övergriplig förklaring till varje kapitel nedan.



Figur 1.3. Flödesschema på hur rapportens kapitel hänger ihop.

Det nästkommande kapitlet (kapitel 2) presenterar vilka faktorer som påverkar trappornens miljöpåverkan. Det ger läsaren en inblick i vilka val en konstruktör står inför vid projektering av ett trapporn och hur dessa val påverkar konstruktionens GWP. Kapitlet avslutas med att koppla dessa designbeslut till konceptet miljöoptimering och diskuterar metoder för att minska trappornets GWP.

Kapitel 3 introducerar grundläggande optimeringsteori. Läsaren får bekanta sig med konceptet optimering och hur det fungerar för att skapa förståelse för miljöoptimeringen och hur det kommer att utföras senare i arbetet.

I det fjärde kapitlet ligger fokus på hur modelleringen och optimeringen fungerar i det parametriska verktyget Grasshopper. Först får läsaren en introduktion i hur Grasshopper och dess komponenter fungerar, samt hur man kan utföra finita elementanalyser i programmet. Därefter används optimeringsteorin från kapitel 3 för att förklara hur den inbyggda optimeringsalgoritmen Galapagos fungerar.

Kapitel 5 presenterar parametrisk design av ett trapporn och lägger fokus på hur Grasshopper-skriptet fungerar. Kapitlet binder samman informationen från kapitel 2 och 4 och presenterar följande:

- Hur skriptet bygger upp Trappornets geometri
- Hur skriptet utför finita elementanalyser
- Hur skriptet utför miljöoptimering av trappornet

Det näst sista kapitlet, kapitel 6, har i syfte att besvara frågeställningarna:

- Går parametrisk design att använda som verktyg vid miljöoptimering av stålkonstruktioner?
- Vilken profiltyp i en trappornsstomme resulterar i lägst GWP?

Dessa frågor kommer besvaras med hjälp av ett referensprojekt. Här skapas först referensprojektet med hjälp av skriptet för att utvärdera hur det fungerar. Sedan presenteras resultaten från miljöoptimeringar för de olika profiltyperna med hänsyn till GWP.

I det sjunde och sista kapitlet förs en diskussion om arbetet, fortsatt forskning, och en slutsats presenteras.

2. Ett trappornas miljöpåverkan

Ett trappornas miljöpåverkan styrs av många olika faktorer och varierar från fall till fall. Detta arbete kommer enbart fokusera på konstruktionens bärande element och inkluderar stålet och rostskyddsmålningen av det. Tornets totala miljöpåverkan uttryckt i GWP beskrivs här med följande ekvation:

$$GWP_{tot} = W_s \cdot \frac{CO_{2e,s}}{kg} + A_m \cdot \frac{CO_{2e,r}}{m^2} \quad (2.1)$$

Kapitlet ämnar förklara hur val av stålprofiler påverkar de båda termerna i ekvationens högerled. Relevanta GWP-värden från svenska leverantörer presenteras för olika stålprofiler och rostskyddsmålningen samt hur optimering kan minska trappornas miljöavtryck.

2.1 Val av profiler

Den bärande stommen i ett trapporn kan konstrueras av olika typer av stålprofiler, och det finns två huvudsakliga val en konstruktör måste göra:

- Öppna eller slutna profiler
- Varmformade eller kallformade profiler

För öppna profiler i trappornsstommen är I- eller H-balkar vanliga alternativ, medan slutna profiler ofta refererar till konstruktionsrör med kvadratisk eller rektangulärt tvärsnitt. De öppna profilerna har liknande byggnadsmekaniska egenskaper, och skiljer sig mycket i jämförelse till ett konstruktionsrör. En I- eller H-balk har högt tröghetsmoment runt sin starka axel, och är särskilt styv och effektiv mot böjbelastningar i den riktningen. Nackdelen är att tröghetsmomentet är betydligt lägre runt den vecka axeln. Konstruktionsrör har ett mer enhetligt tröghetsmoment runt båda sina axlar, särskilt i fallet med kvadratiska tvärsnitt där dessa moment är likvärdiga. Dock är detta moment generellt lägre jämfört med den starka axeln hos öppna profiler. Värt att nämna är också att slutna profiler har betydligt högre vridstyvhet än öppna (Isaksson & Mårtensson, 2008).

Valet mellan varmformade och kallformade profiler är enbart aktuellt för konstruktionsrör då kallformade I- och H-balkar är näst intill icke-existerande och lagerförs inte hos de stora stålleverantörerna. Konstruktionsrör finns däremot i både varm- och kallformat utförande vilket ger profilerna olika egenskaper som måste beaktas vid dimensionering (BE Group Sverige AB, 2022). När profiler kallformas skapas mer egenspanningar än vid varmformning, vilket påverkar knäckningskapaciteten negativt. I Eurokod 3 tar man hänsyn till dessa egenspanningar genom att dimensionera profilerna utifrån olika knäckningskurvor (Andersson, 2016).

Stålprofilerna har inte bara olika byggnadsmekaniska egenskaper, utan också olika stor miljöpåverkan. Skillnaden i miljöpåverkan grundar sig i vilken ståltillverkningsprocess profilen genomgått. I Sverige tillverkas stål på två olika sätt:

- Malmbaserad ståltillverkning
- Skrotbaserad ståltillverkning

Den malmbaserade ståltillverkning innebär kortfattat att man med hjälp av en masugn reducerar järnmalm till råjärn med koldioxid som restprodukt. Råjärnet går sedan i flytande form vidare till stålverken. Skrotbaserad ståltillverkning använder i stället återvunnet stålskrot som smälts ned i eldrivna ljusbågsugnar och producerar nytt stål (Jernkontoret, 2019).

Tillverkningsprocessen är avgörande för hur stort utsläpp den slutliga stålprofilen får, där den malmbaserade ståltillverkningen med koldioxid som restprodukt har ett betydligt större miljöavtryck än den skrotbaserade. Enligt stålleverantören Tibnors² (2020) miljövarudeklarationer är deras konstruktionsrör malmbaserade medan de öppna profilerna är skrotbaserade, vilket ger följande utsläpp under produktskedet A1-A3 enligt figur 1.6:

Tabell 2.1. Ståltillverkningsutsläpp under produktskedet (Tibnor, 2020). Hela miljövarudeklarationen hittas i bilaga A.

Typ av profil	A1-A3 [kgCO _{2e} /kg]
Öppna varmformade profiler	1.15
Varmformade konstruktionsrör (VKR)	2.60
Kallformade konstruktionsrör (KKR)	2.58

Vid en första anblick verkar öppna profiler mer miljövänliga då slutna släpper ut mer än dubbla mängden CO_{2e} under produktskedet. Innan den slutsatsen kan dras måste dock två saker tas i beaktning:

1. Mängden stål som återvinns täcker inte efterfrågan.
2. Stål är ett cirkulärt material som kan återvinnas.

Idag täcker allt återvunnet stål ungefär 25–30% av världens behov, och år 2050 kommer hälften av allt stål behöva vara malmbaserat (Olofsson, 2018). Den skrotbaserade tillverkningen är alltså i högsta grad beroende av den malmbaserade.

För att ta återvinningsaspekten i beaktning kan man kolla på stålets sista fas i livscykeln, övrig miljöinfo (fas D). Här kan CO_{2e} vinsten från att återvinna det nyproducerade stålet dras bort från produktens totala utsläpp. Dock kan det enbart göras för stål som inte redan har återvunnits (då det redan är borträknat i produktskedet, fas A1). En betydande del av utsläppen från de malmbaserade profilerna kan därför dras bort, och resultatet för profilernas hela livscykel, presenteras i tabell 2.2 nedan (Tibnor, 2020).

Tabell 2.2. Profilers utsläpp under hela livscykeln (Tibnor, 2020). Hela miljövarudeklarationen hittas i bilaga A.

Typ av profil	A1-D [kgCO _{2e} /kg]
Öppna varmformade profiler	1.18
Varmformade konstruktionsrör (VKR)	1.41
Kallformade konstruktionsrör (KKR)	1.35

² Tibnor är en svensk väletablerad stålleverantör som ägs av SSAB.

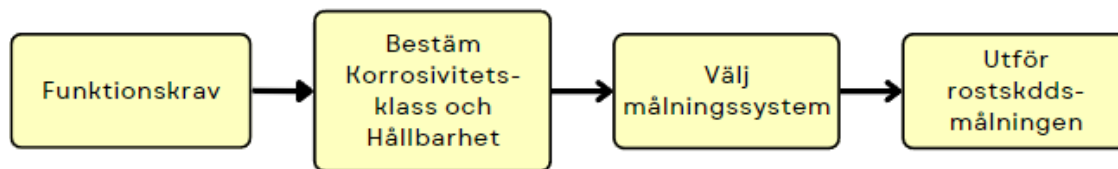
2.2 Rostskyddsmålning

En annan faktor som kommer tas i beaktning är rostskyddsmålning av stålprofilerna och dess bidrag till trappornets totala GWP. Det som avgör hur stor mängd färg som går åt är dels profilernas mantelyta, dels hur tjockt lager som målas. Öppna tvärsnitt har en större mantelyta än stängda. En jämförelse mellan mantelytor görs i tabell 2.3 nedan, där resultatet visar att en HEA-profil har ungefär 1.65 gånger större mantelyta än ett kvadratisk konstruktionsrör med motsvarande vikt.

Tabell 2.3. Korrelation mellan typ av tvärsnitt och mantelyta.

Tvärsnitt	Vikt [kg/m]	Mantelyta [m ² /m]
HEA 100	16.7	0.56
KR 100x100x6	17.0	0.34

Vilken typ av färg och hur tjockt lager som ska appliceras beror på målningsystemet, som i sin tur beror på funktionskravet. Flödesschemat från funktionskrav till rostskyddsmålning introduceras i figur 2.1 (SIS, 2017).



Figur 2.1. Flödesschema på funktionskrav till rostskyddsmålning (SIS, 2017).

Enligt bilaga F i SS-EN-1090-2 (SIS, 2018) kan målningsystemet väljas med hjälp av ett funktionskrav. Vid användning av funktionskrav behöver enbart två uppgifter bestämmas, förväntad livslängd för rostskyddet och korrosivitetsklass. Förväntad livslängd på rostskyddet ska föreskrivas som något av följande enligt ISO 12944-1 (SIS, 2017):

- Låg (L), upp till 7 år
- Medium (M), 7–15 år
- Hög (H), 15–25 år
- Väldigt hög (VH), över 25 år

Korrosivitetsklassen bestäms beroende på vilken korrosivitetsmiljö stålet utsätts för. Tabell 2.4 nedan, presenterar de olika korrosivitetsklasserna och respektive korrosivitetsmiljö.

Tabell 2.4. Korrosivitetssklasserna och respektive korrosivitetssmiljö, hämtad från ISO-12944-2 (SIS, 2017).

Korrosivitetsklass	Miljöns korrosivitet	Utomhus	Inomhus
C1	Mycket låg	-	Uppvärmade utrymmen med torr luft och obetydlig mängd föroreningar.
C2	Låg	Låga halter luftföroreningar, lantliga områden	Icke uppvärmda utrymmen med växlande temperatur och luftfuktighet.
C3	Måttlig	Måttlig mängd luftföroreningar. Stadsområden och lätt industrialiserade områden.	Måttlig fuktighet och viss mängd luftföroreningar.
C4	Hög	Påtagliga mängder luftföroreningar eller måttlig mängd salt. Industri och kustområden.	Hög fuktighet och stor mängd luftföroreningar.
C5	Mycket hög	Industriella områden med aggressivt klimat och hög luftfuktighet. Kust- och offshoreområden med stora mängde salt	Permanent fukt-kondensation och stor mängd luftföroreningar.
CX	Extrem	Extrema industriell, kust- och offshoreområden i subtropiska och tropiskt klimat med salt och föroreningar, skvalpzoner.	Extrem fukt-kondensation och extremt aggressivt klimat.

När funktionskravet har bestämts ska målningsystem väljas. Målningsystemen presenteras i ISO-12944-5 (SIS, 2019) och det finns oftast flera alternativ att uppnå funktionskravet..

För att kunna ta hänsyn till rostskyddsmålning i kommande miljöoptimering måste en uppskattning av färgens GWP göras. Relevant för detta arbete är korrosivitetssklass C2 då trapptornen antas vara i en icke uppvärmd inomhusmiljö och livslängden på rostskyddet antas vara hög. De system som uppfyller nämnt funktionskrav är C2.03, C2.05 och C2.07 (se tabell 2.5 nedan). NDFT står för "nominal dry film thickness" och syftar till färglagrets tjocklek, och det högra NDFT-värdet under "paint system" är den slutgiltiga tjockleken på rostskyddsmålningen.

Tabell 2.5. Olika målningsystem för korrosivitetsklass C2. Hämtad från ISO-12944-5 (SIS, 2019).

System No.	Priming coat				Subsequent coat(s)	Paint system		Durability			
	Binder	Type of primer	No. of coats	NDFT in μm		Binder type	Total no. of coats	NDFT in μm	l	m	h
C2.01	AK, AY	Misc.	1	40 to 80	AK, AY	1 to 2	80	X			
C2.02	AK, AY	Misc.	1	40 to 100	AK, AY	1 to 2	100	X	X		
C2.03	AK, AY	Misc.	1	60 to 160	AK, AY	1 to 2	160	X	X	X	
C2.04	AK, AY	Misc.	1	60 to 80	AK, AY	2 to 3	200	X	X	X	X
C2.05	EP, PUR, ESI	Misc.	1	60 to 120	EP, PUR, AY	1 to 2	120	X	X	X	
C2.06	EP, PUR, ESI	Misc.	1	80 to 100	EP, PUR, AY	2	180	X	X	X	X
C2.07	EP, PUR, ESI	Zn (R)	1	60	—	1	60	X	X	X	
C2.08	EP, PUR, ESI	Zn (R)	1	60 to 80	EP, PUR, AY	2	160	X	X	X	X

Olika färgtyper har olika mycket utsläpp och täckmängd. Färgleverantören Tikkurila (2024) har följande produkter med respektive täckmängd vid en tjocklek på 60 μm och GWP per liter färg. Produkternas miljövarudeklarationer finns i bilaga B:

Tabell 2.6. Tikkurilas produkter med respektive täckmängd och GWP per liter. Presenteras i bilaga B.

Produkt	Färgtyp	Täckmängd vid 60 μm [m^2/l]	GWP [$\text{kgCO}_2\text{e}/\text{l}$]
Fontecryl SC50	Alkyd AK (Misc)	7.5	4.49
Fontecoat FD20	Epoxy EP (Misc)	6.7	6.99
Temadur 10	Polyuretan PUR (Misc)	7.5	7.42
Fontezink 85	Zinkrik epoxy, EP (Zn)	8.7	16.60

För att jämföra GWP:n mellan de tre systemen som uppfyller C2 och en hög livslängd används indata från tabell 2.5 ovan och en godtycklig area (1m^2) och beräknas enligt:

$$GWP_{system} = \frac{Area}{Täckmängd} \cdot GWP_{produkt} \quad (2.2)$$

För system C2.03 väljs enbart 160 μm Fontecryl SC50, där täckmängden blir 2.5 m^2/l .

$$GWP_{C2.03} = \frac{1}{2.5} \cdot 4.49 = 1.80 \text{ kgCO}_2\text{e}/\text{m}^2 \quad (2.3)$$

För system C2.05 kan man välja mellan både epoxy- och polyuretanfärgen. Här väljs epoxyfärgen Fontecoat FD20 då den har både minst åtgång och GWP i jämförelse med polyuretanfärgen. Täckmängden för 120 μm blir 3.35 m^2/l .

$$GWP_{C2.05} = \frac{1}{3.35} \cdot 6.99 = 2.09 \text{ kgCO}_2\text{e}/\text{m}^2 \quad (2.4)$$

System C2.07 har enbart ett 60 μm tjockt lager Fontezink 85.

$$GWP_{C2.07} = \frac{1}{8.7} \cdot 16.60 = 1.91 \text{ kgCO}_2\text{e}/\text{m}^2 \quad (2.5)$$

Det går inte att veta vilket av dessa system som entreprenören väljer att använda, men genom att ta ett snittvärde av de tre system kan ett rimligt GWP-värde per kvadratmeter uppskattas för det relevanta funktionskravet.

$$GWP_{mätning} = \frac{1.80+2.09+1.913}{3} = 1.93 \text{ kgCO}_2\text{e/m}^2 \quad (2.6)$$

2.3 Miljöoptimering

Sammanfattningsvis har öppna profiler lägre GWP per kg stål än slutna. Däremot har öppna profiler en större mantelyta per kg stål än vad stängda har, vilket ökar mängden rostskyddsfärg som krävs för att måla konstruktionen. Utöver det har profilerna också olika byggmekaniska egenskaper. Allt detta gör att det krävs en optimering av konstruktionen för att bestämma vilken typ av bärande profiler i trappornet som ger lägst GWP.

Optimering av konstruktioner är inget nytt fenomen och antalet vetenskapliga studier om strukturell optimering har ökat avsevärt över åren (Mei & Wang, 2021). Kortfattat handlar strukturell optimering att dimensionera konstruktioner så att de uppnår önskade utnyttjandegrader på det mest materialeffektiva sättet, oftast ur ett vikt- eller kostnadsperspektiv. Den optimering som detta arbete ämnar utföra kommer i stället optimera konstruktionens GWP, där grunden till optimeringen är konstruktionens totala miljöpåverkan enligt ekvation 2.1.

3. Optimeringsteori

I detta avsnitt kommer grundläggande optimeringsteori att presenteras. Detta innefattar processen att formulera ett optimeringsproblem, strategier för att behandla begränsningar, samt hur olika designvariabler påverkar valet av optimeringsalgoritm.

3.1 Problemställning

Inom alla optimeringsmetoder måste man omvandla sin intention till en matematisk formulering. Enligt Yang (2010) kan de flesta optimeringsproblem beskrivas matematiskt på följande form:

$$\text{minimera } f_i(\mathbf{x}), i = 1, 2, \dots, M \quad M \in \mathbb{N} \quad (3.1)$$

Med hänsyn till begränsningarna:

$$g_j(\mathbf{x}) \leq 0, j = 1, \dots, J \quad J \in \mathbb{N} \quad (3.2)$$

$$h_k(\mathbf{x}) = 0, k = 1, \dots, K \quad K \in \mathbb{N} \quad (3.3)$$

Där

$$\mathbf{x} = (x_1, x_2, \dots, x_d)^T \quad \mathbf{x} \in [\mathbf{x}_{min}, \mathbf{x}_{max}] \subset \mathbb{R}^d \quad (3.4)$$

Detta kan beskrivas med ord på följande sätt: Hitta det \mathbf{x} som minimerar funktionen $f(\mathbf{x})$ med hänsyn till begränsningarna $h_j(\mathbf{x})$ och $g_k(\mathbf{x})$. I denna beskrivning är $f(\mathbf{x})$ den så kallade målfunktionen. Målfunktionen representerar det objekt som ska minimeras eller maximeras. M representerar antal funktioner som ska optimeras. Optimeringsproblem med ett mål medför $M=1$ medan för mångobjektiva optimeringsproblem blir $M>1$. Målfunktionens definitionsmängd, x_i brukar kallas för designvariabler och är värdena som maximerar eller minimerar resultatet av målfunktionen. Dessa omges av området \mathbb{R}^d , vilket brukar kallas för sökområdet eller designområdet.

Likheterna och olikheterna $g_j(\mathbf{x})$ och $h_k(\mathbf{x})$ kallas för begränsningar. Begränsningarna sätter upp krav och restriktioner som reducerar antalet möjliga lösningar genom att begränsa sökområdet. När dessa är linjära samt målfunktionen är linjär klassas det som ett linjärt optimeringsproblem. Vanligtvis är dock begränsningarna samt målfunktionen olinjär vilket ger ett olinjärt optimeringsproblem. Det finns även fall då optimeringsproblem är helt obegränsade, det vill säga $J=K=0$, vilket kallas för ett obegränsat optimeringsproblem (Yang, 2010).

3.2 Straffmetoden

Optimeringsproblem som innehåller olinjära begränsningar är avsevärt mer komplexa att hantera jämfört med obegränsade eller linjärt begränsade problem. En metod för att hantera denna komplexitet är att omvandla det begränsade problemet till ett obegränsat genom att införa den så kallade straffmetoden (eng. penalty method). Straffmetoden innebär att man lägger till strafftermer till målfunktionen baserat på överträdelsen av begränsningarna. Genom

att omvandla målfunktionen till en strafffunktion, kan optimeringsalgoritmen vägledas i rätt riktning. Detta sker genom att ett straff adderas till målfunktionen när en begränsning inte är uppfylld, vilket resulterar i högre funktionsvärden och därigenom styrs algoritmen mot att uppfylla alla begränsningar. Målfunktionen $f(\mathbf{z})$ kan modifieras till en straffad målfunktion på följande sätt:

$$\Pi(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\nu}) = f(\mathbf{x}) + P(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\nu}) \quad (3.5)$$

Där $P(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\nu})$ är strafftermen vilken definieras på följande sätt enligt (Yang, 2014):

$$P(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\nu}) = \sum_{j=1}^J \mu_j H_j[g_j(\mathbf{x})] g_j^2(\mathbf{x}) + \sum_{k=1}^K \nu_k H_k[h_k(\mathbf{x})] h_k^2(\mathbf{x}) \quad (3.6)$$

Faktorerna $H_j[g_j(\mathbf{x})]$ och $H_k[h_k(\mathbf{x})]$ är indikationsfaktorer vilka är 0 när en begränsning är uppfylld och 1 om den inte är uppfylld. Matematisk kan de definieras som:

$$H_j[g_j(\mathbf{x})] = \begin{cases} 0 & \text{om } g_j(\mathbf{x}) \leq 0 \\ \text{annars} & 1 \end{cases} \quad (3.7)$$

$$H_k[h_k(\mathbf{x})] = \begin{cases} 0 & \text{om } h_k(\mathbf{x}) = 0 \\ \text{annars} & 1 \end{cases} \quad (3.8)$$

Med hjälp av definitionerna i 3.7 och 3.8 visas att när en begränsning är uppfylld kommer straffet till funktionen vara noll och när ett krav inte är uppfyllt kommer funktionen att straffas. Storleken på straffet beror på strafffaktorerna μ_j och ν_k är upp till designern att välja själv och kan skilja sig beroende på optimeringsproblem. Det är viktigt att tänka på att välja en mängd som är rimlig för storleken på problemet. Ett väldigt högt straff till en funktion med låga värden kan ge en överstraffad (eng. overpenalty) funktion och på samma sätt kan funktion med höga värden som tilldelas för låga straff bli understraffad (eng. underpenalty). Enligt Yang (2014), kan strafftermerna för de flesta tillämpningarna ligga mellan 10^3 och 10^{15} .

Strafftermen som tillämpas i detta arbete använder enbart en begränsning och skiljer sig därför något från exemplet som presenteras i ekvation 3.6. Begränsningen baseras på stålelementens utnyttjandegrad, som inte får överstiga 1.0, och blir således följande:

$$P(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\nu}) = \sum_{j=1}^J \mu_j H_j[g_j(\mathbf{x})] g_j^2(\mathbf{x}) \quad (3.9)$$

$$H_j[g_j(\mathbf{x})] = \begin{cases} 0 & \text{om } g_j(\mathbf{x}) < 1.0 \\ \text{annars} & 1 \end{cases} \quad (3.10)$$

Mer om hur arbetet applicerar denna straffterm i optimeringsproblemet förklaras mer ingående i kapitel 5.

3.3 Designvariabler och optimeringsalgoritmer

Designvariablerna inom optimeringsteori brukar vanligen delas in i två olika typer: Diskreta och kontinuerliga variabler. Diskreta variabler kan endast anta specifika diskreta värden. I ett byggsammanhang kan detta exempelvis vara antal våningar eller antal element i en konstruktion. Kontinuerliga variabler kan däremot anta vilket värde som helst inom ett visst intervall och är inte begränsade till särskilda fasta punkter. I det här sammanhanget kan det till

exempelvis vara geometriska variabler som längd, bredd eller höjd som vanligen kan definieras med stor noggrannhet. När ingenjörer arbetar med design och dimensionering stöter de ofta på kataloger där de måste välja komponenter som har vissa egenskaper. Dessa komponenter brukar vanligtvis klassificeras som kategoriska variabler, vilka är en form av diskreta variabler, men de följer vanligen inte något tydligt mönster (Brown & Hwang, 1993).

I optimeringssammanhang påverkar valet av designvariabel vilken typ av optimeringsalgoritm som är mest lämplig. När kontinuerliga variabler används, är sökområdet också kontinuerligt, vilket gör att derivatabaserad optimeringstekniker kan användas så som Newton-metoden och gradientmetoden eller andra numeriska metoder för att hitta globala minimum. När designvariablerna däremot är kategoriska funkar detta inte då derivering inte är möjlig. När kategoriska variabler används, består sökområdet av diskreta punkter, där varje punkt representerar en möjlig lösning. Detta resulterar i ett stort antal potentiella lösningar som måste utvärderas, vilket gör optimeringsproblemet mer utmanande och tidskrävande (Brown & Hwang, 1993).

En ytterligare kategorisering som kan göras inom optimeringsteorin, men för optimeringsalgoritmer, är om de är stokastiska eller deterministiska. En deterministisk algoritm är en typ av algoritm som för en given indata alltid producerar samma utdata och genomgår samma sekvens av steg i beräkningsprocessen, medan stokastiska algoritmer integrerar ett element av slump i sökprocessen för att finna lösningar. På grund av detta kommer de stokastiska algoritmerna inte alltid uppvisa samma lösning, medan de deterministiska gör det. Detta medför att stokastiska algoritmer kan missa den optimala lösningen vid varje iteration, vilket resulterar i en mer tidskrävande eller till och med misslyckad process (Chalouhi, 2022).

3.4 Naturinspirerade algoritmer

En typ av algoritm som har visat sig passa bra för att lösa olinjära optimeringsproblem är naturinspirerade metaheuristiska algoritmer. Dessa algoritmer faller under kategorin stokastiska metoder.

Heuristiska algoritmer förlitar sig på en process av försök och fel (eng. trial and error) för att komma fram till lösningar av hög kvalitet, men utan en garanti för att uppnå den absolut bästa lösningen. Metaheuristiska algoritmer tar detta ett steg längre genom att kombinera slumpmässig utforskning med strategisk lokal sökning, vilket ofta resulterar i bättre prestanda jämfört med enkla heuristiska tekniker. Även om metaheuristiska algoritmer inte kan garantera att de hittar den optimala lösningen varje gång, är deras förmåga att snabbt hitta acceptabla lösningar inom praktiska tidsramar varför de är värdefulla i många tillämpningsområden (Yang, 2010).

Många av dagens metaheuristiska algoritmer är naturinspirerade. Dessa algoritmer är designade efter att efterlikna biologiska processer, naturliga system eller beteendemönster som observeras i naturen. Några väl kända exempel på sådana algoritmer är bland annat Particle Swarm Optimization (PSO), Simulated Annealing (SA) och Genetiska Algoritmer (GA). Dessa algoritmer delar en gemensam strategi; de använder metaheuristiska metoder för att navigera i sökandet efter optimala lösningar. PSO hämtar inspiration från kollektiva beteendemönster hos djur, såsom fågelflockar eller fiskstim, där grupper koordinerar sina rörelser för att lokalisera de mest fördelaktiga områdena i en sökrymd. Genetiska Algoritmer

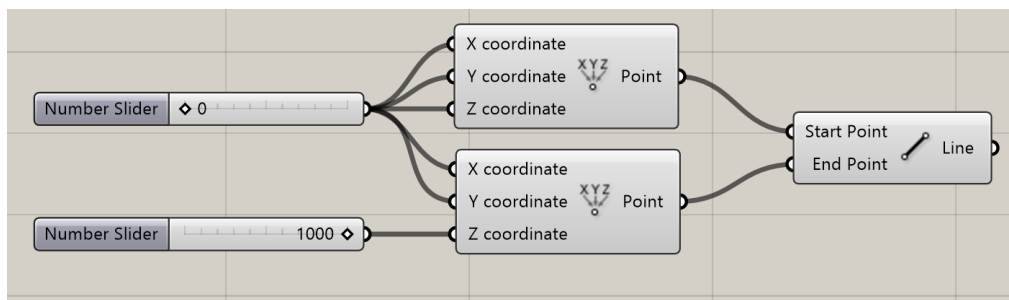
är baserade på de principer som finns i naturligt urval och evolutionsteori, vilket innebär att de bästa lösningarna överlever och förädlas över generationer (Yang, Nature - Inspired Metaheuristic Algorithms, 2010). Simulated Annealing, eller simulerad glödning, är inspirerad av termiska behandlingsprocesser inom materialvetenskapen. Denna process efterliknar hur partiklar omorganiserar sig under kylning efter att ha blivit upphettade, vilket leder till strukturer med förbättrade egenskaper (Van Laarhoven & Aarts, 1987). Alla dessa algoritmer utnyttjar komplexa metaheuristiska tekniker för att på effektivt sätt utforska och identifiera optimala lösningar inom deras respektive sökrumder.

4. Parametrisk modellering och optimering i Grasshopper

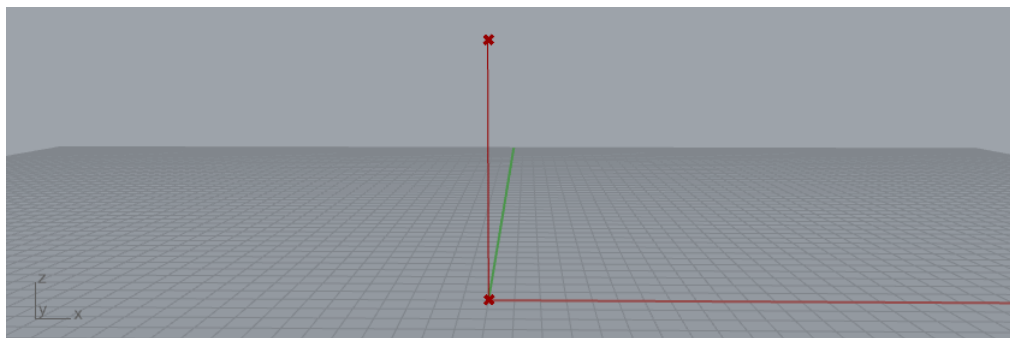
I syfte att integrera parametrisk design och kunna skapa, analysera och optimera önskvärda trapptronskonstruktioner har Grasshopper valts som verktyg. Grasshopper är en etablerad programvara bland arkitekter och konstruktörer och är fullt integrerad med CAD-plattformen Rhinoceros (Rhino3D) (Sawant, 2021).

4.1 Modellering i Grasshopper

Grasshopper är ett visuellt programmeringsspråk som förenklar skapandet av algoritmer genom att koppla samman logiska element. Dessa element, kallade komponenter, fungerar som funktionella block och placeras på en arbetsyta. Det går även att skapa egna komponenter med vars funktioner kan kodas med hjälp av programmeringsspråken C# och Python. Genom att ansluta ledningar från block till block kan de bearbeta en uppsättning data, som rör sig från vänster till höger. Geometri kan antingen importeras från Rhino3D och användas som inmatning eller skapas direkt i Grasshopper. Den visas och uppdateras i Rhino3D:s vyport när komponenter läggs till eller parametrar ändras. Det går till exempel att med hjälp av koordinater skapa två punkter och sedan dra en linje mellan dessa, figur 4.1 illustrerar komponenterna i Grasshopper och figur 4.2 visar resultatet i Rhino3D:s vyport. Genom att ändra det nedre sifferreglaget som är kopplat till ena punktens z-koordinat kan användaren flytta punkten i z-led vilket i sin tur ändrar längden på linjen.



Figur 4.1. Komponenterna i Grasshopper som bildar två punkter och en linje mellan dem.



Figur 4.2. Resultatet av komponenterna i figure 4.1 i Rhino3D:s vyport.

4.2 Finit elementanalys i Grasshopper

Finita elementanalyser (FE-analyser) är ett viktigt verktyg i konstruktörens verktygslåda och är avgörande i designprocessen för en mängd olika strukturer, från små komponenter till storskaliga konstruktioner. Genom att kombinera FE-analys med system- och kontrollberäkningar undersöks hur en konstruktion kan stå emot de påfrestningar och belastningar som den kommer att utsättas för under sin livstid. FE-analysen är därför ett viktigt hjälpmedel för att garantera strukturens säkerhet och funktion. Detta kapitel kommer inte gå in i detalj på hur beräkningarna i en FE-analys utförs, för den nyfikne finns mängder av litteratur tillgängligt, till exempel *Introduction to the finite element method* (Ottosen & Petersson, 1992) . Kort beskrivet delas strukturen upp i mindre, enklare delar, finita element, och för varje element löses ekvationer som beskriver dess beteende.

Idag finns det många programvaror som kan utföra FE-analyser, och ett av dessa är RFEM 6. RFEM 6 är en kraftfull 3D-programvara för FE-analys utvecklad av Dlubal. I programmet finns det möjlighet att genomföra detaljerade beräkningar som till exempel deformationer, interna krafter, spänningar och reaktionskrafter. Med hjälp av en inbyggd Eurokod-modul kan dessa resultat användas för att kontrollera konstruktionselementens utnyttjandegrader (Dlubal, 2024).

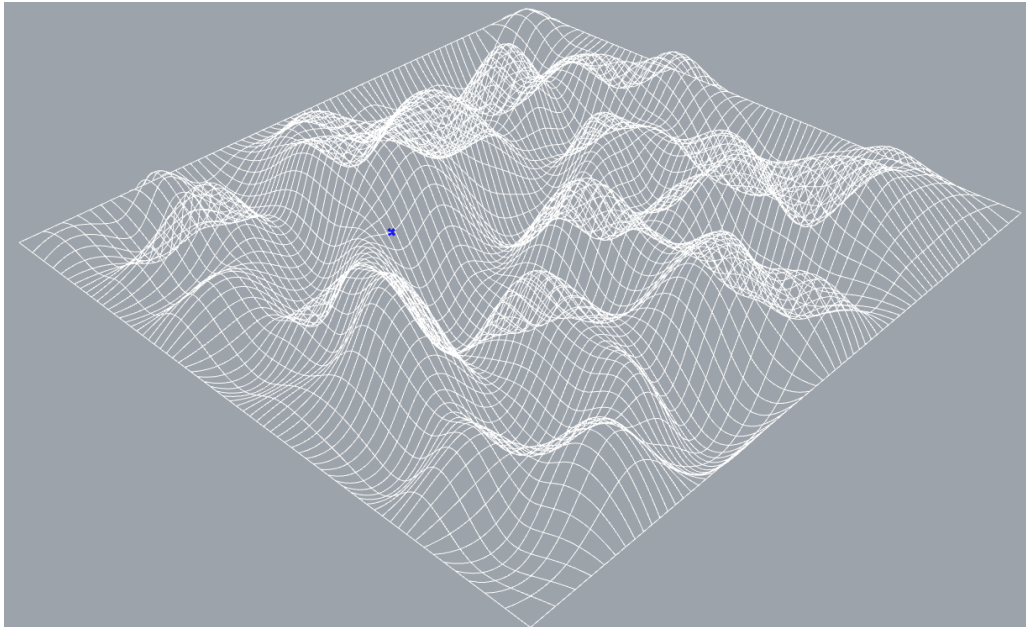
Dlubal har nyligen utvecklat både en Application Programming Interface (API) mellan RFEM 6 och Grasshopper samt Grasshopper-komponenter. Detta möjliggör för programvarorna att kommunicera samt att redan i Grasshopper skapa all data som krävs för att göra en FE-analys och kontrollberäkningar enligt Eurokod 3. Med dessa komponenter kan användaren effektivt skapa modellens geometri, definiera material, upplag, tvärsnitt, laster, lastfall och lastkombinationer. Denna data exporteras till RFEM 6 där modellen automatiskt genereras. En FE-analys utförs, beräkningar på elementens kapacitet genomförs enligt Eurokod 3 och resultaten importeras tillbaka till Grasshopper (Dlubal, 2024).

4.3 Optimering i Grasshopper

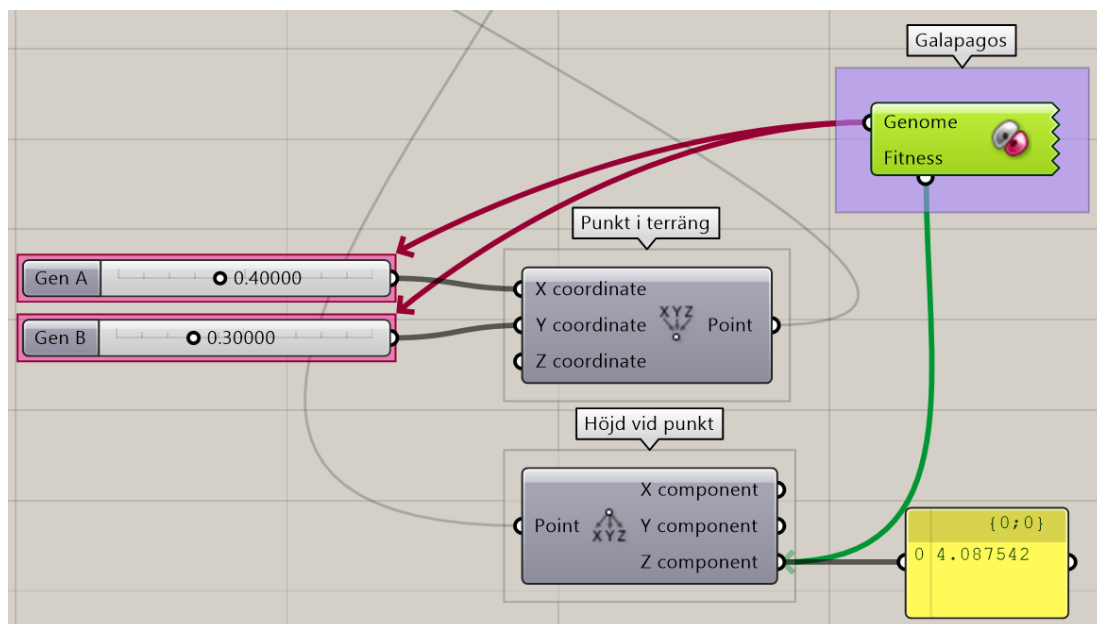
Grasshopper innehåller ett inbyggt optimeringsverktyg som kallas för Galapagos, vilket erbjuder två olika optimeringsalgoritmer: Simulated Annealing och en genetisk algoritm. I Woerlis (2019) examensarbete utfördes jämförande tester på dessa algoritmer, vilket visade att SA inte var lika tidseffektiv som den genetiska algoritmen. Därför har vi valt att använda den genetiska algoritmen i detta arbete.

David Rutten är utvecklaren av Galapagos vars funktionsbeskrivning finns tillgänglig på hans blogg *I Eat Bugs For Breakfast* (2011). För att illustrera Galapagos arbetsprocess använder Rutten ett exempel där programmet ska hitta den högsta punkten i ett landskap format av bergstoppar. Denna virtuella terräng är definierad av ett rutnät bestående av x- och y-koordinater. I detta exempel betecknas x-koordinaten som gen A och y-koordinaten som gen B, se figur 4.4 och 4.5 nedan. Kombinationen av gen A och B ger upphov till en specifik ”fitness”, vilket är ett mått på prestanda eller lämplighet i optimeringssammanhang. I det här exemplet representeras prestandan av en punkt som följer höjden av landskapet med avseende de koordinater som gen A och B skapar. Målet med optimeringen är att lokalisera den kombination av generna som ger den högsta toppen i landskapet.

När Galapagos initieras har den ingen information om terrängens utformning, så det initiala tillvägagångssättet är att "befolka" landskapet med ett antal slumpmässigt placerade individer, eller "genomer". Dessa genomer representeras av punkter med unika x- och y-koordinater, som exempelvis $P=(x=0.4, y=0.3)$. Programmet utvärderar sedan varje punkts prestanda genom att jämföra höjden på deras positioner, där punkter på högre höjd får en bättre ranking än de på lägre.

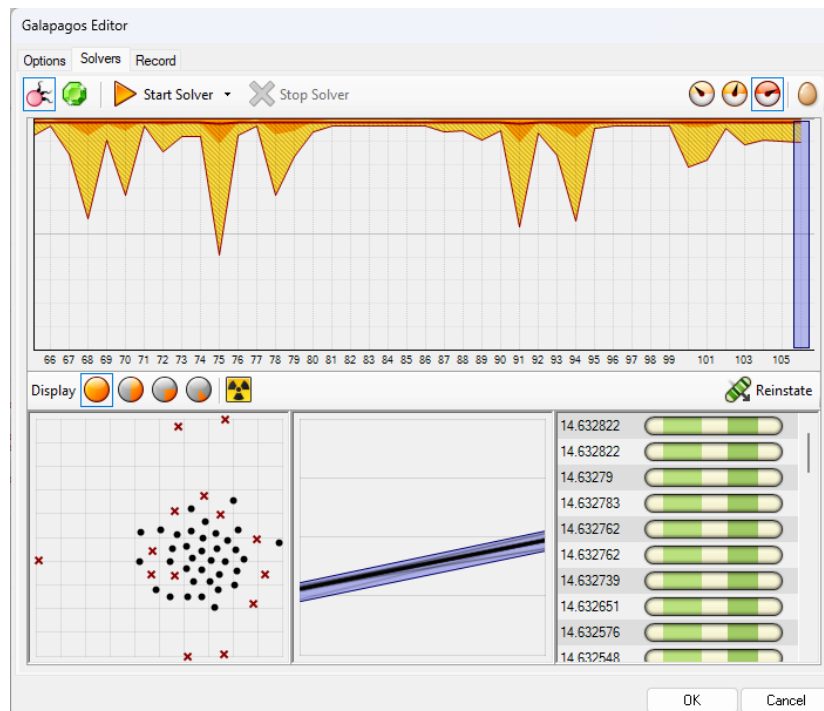


Figur 4.3. Visar slumpmässigt genererat landskap i Rhinoceros. Den blå punkten representerar startpunkten i landskapet $P=(x=0.4, y=0.3)$ som ska optimeras.

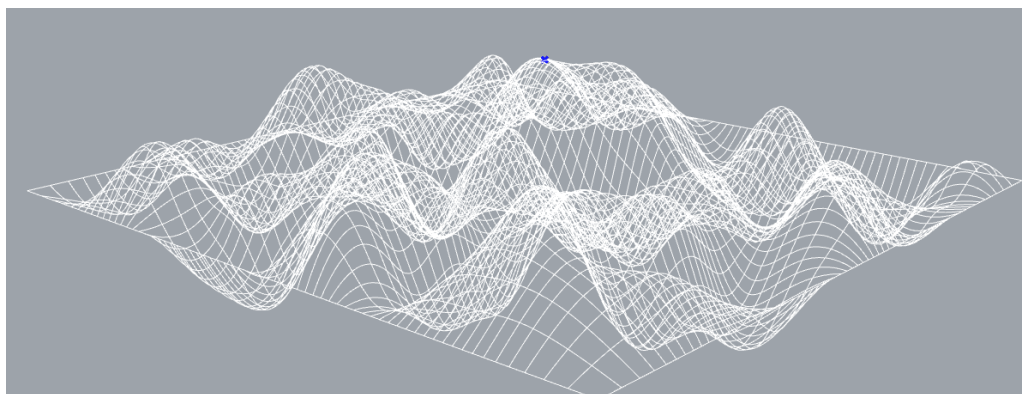


Figur 4.4. Visar vy i Grasshopper. De röda pilarna pekar på gen A och gen B. Den gröna pilen är fitnessen vilket är höjdkoordinaten som ska optimeras.

Efter denna första utvärdering favoriserar Galapagos de genomer som befinner sig på högre höjd och eliminerar de med lägre prestanda. De bäst presterande genomerna avlas sedan, vilket skapar nya genomer som är en genetisk kombination av sina föräldrar. Denna nya generation kan hamna på ännu utforskade positioner i landskapet, någonstans mellan föräldrarna. Denna process upprepas – med selektion av de bästa och eliminering av de sämsta – tills det att den optimala punkten, det vill säga den högsta toppen, har hittats (Rutten, 2011).



Figur 4.5. Visar optimeringsprocessen i Galapagos, den gröna listan representerar punkter som Galapagos hittar, rangordnat från bästa till sämsta.



Figur 4.6. Den blå punkten representerar den högsta punkten Galapagos kunde hitta.

Innan man börjar använda Galapagos kan man göra vissa inställningar för att påverka optimeringsprocessen, dessa visas i figur 8 nedan. Dessa inställningar inkluderar att definiera antalet individer i populationen och det maximala antalet stagnationer. En individ i en population kan betraktas som en möjlig lösning. I detta fall utgör 50 individer en population. Dessutom kan man justera en "initial boost" – eller inledande skjuts – vilket innebär att den första populationen kan göras större än de följande för att utforska en bredare del av sökrymden. I detta exempel är den första generationen dubbelt så stor. Stagnation inträffar när en algoritm slutar göra framsteg mot det globala optimumet, trots att populationen fortfarande är varierad och inte konvergerar mot en lokal optimalpunkt (Lampinen & Zelinka, 2000). Med andra ord, om Galapagos har utfört flera optimeringsförsök utan framsteg, kommer algoritmen att stanna, exempelvis efter 50 stagnationer enligt figur 8 nedan.

I Galapagos finns även alternativet "maintain", som bestämmer hur stor del av den ursprungliga populationen som bevaras från en generation till nästa under optimeringsprocessen. I fallet nedan är inställningen satt till 5%, vilket innebär att de 5% bäst presterande individerna – i detta sammanhang punkter – från varje generation behålls utan att förändras. Detta är för att bevara mångfalden inom populationen och undvika tidig konvergens mot lokala optimum. "Inbreeding" påverkar antalet avkommor som tillåts mellan individerna från samma population. Det är också ett verktyg för att minska risken att algoritmen fastnar vid lokala optimum.

Tabell 7. Visar inställningar för Galapagos

Max. stagnant	50
Population	50
Initial Boost	2 x
Maintain	5%
Inbreeding	75%

5. Parametrisk design av trapptorn

Huvuddelen av detta examensarbete går ut på att utveckla ett Grasshopper-skript som med hjälp av RFEM 6 utför en miljöoptimering av trapptorn. Skriptet kan delas in i 5 steg och utförs i följande ordning vid varje iteration av optimeringsprocessen:

1. Skapande/förändring av geometri i Grasshopper
2. Val av tvärsnitt
3. Exportering av data till RFEM 6
4. Fe-analys, systemberäkningar och kontrollberäkningar i RFEM 6
5. Importering av kontrollberäkningar från RFEM 6 till Grasshopper
6. Tillbaka till punkt 1

Detta kapitel kommer först klargöra hur skriptet med hjälp av de parametriserade variablerna skapar trapptornets geometri. Sedermera förklaras vilken data som exporteras till RFEM 6 och hur analysen utförs. Kapitlets sista avsnitt beskriver hur optimeringsprocessen går till.

5.1 Trapptornets geometri

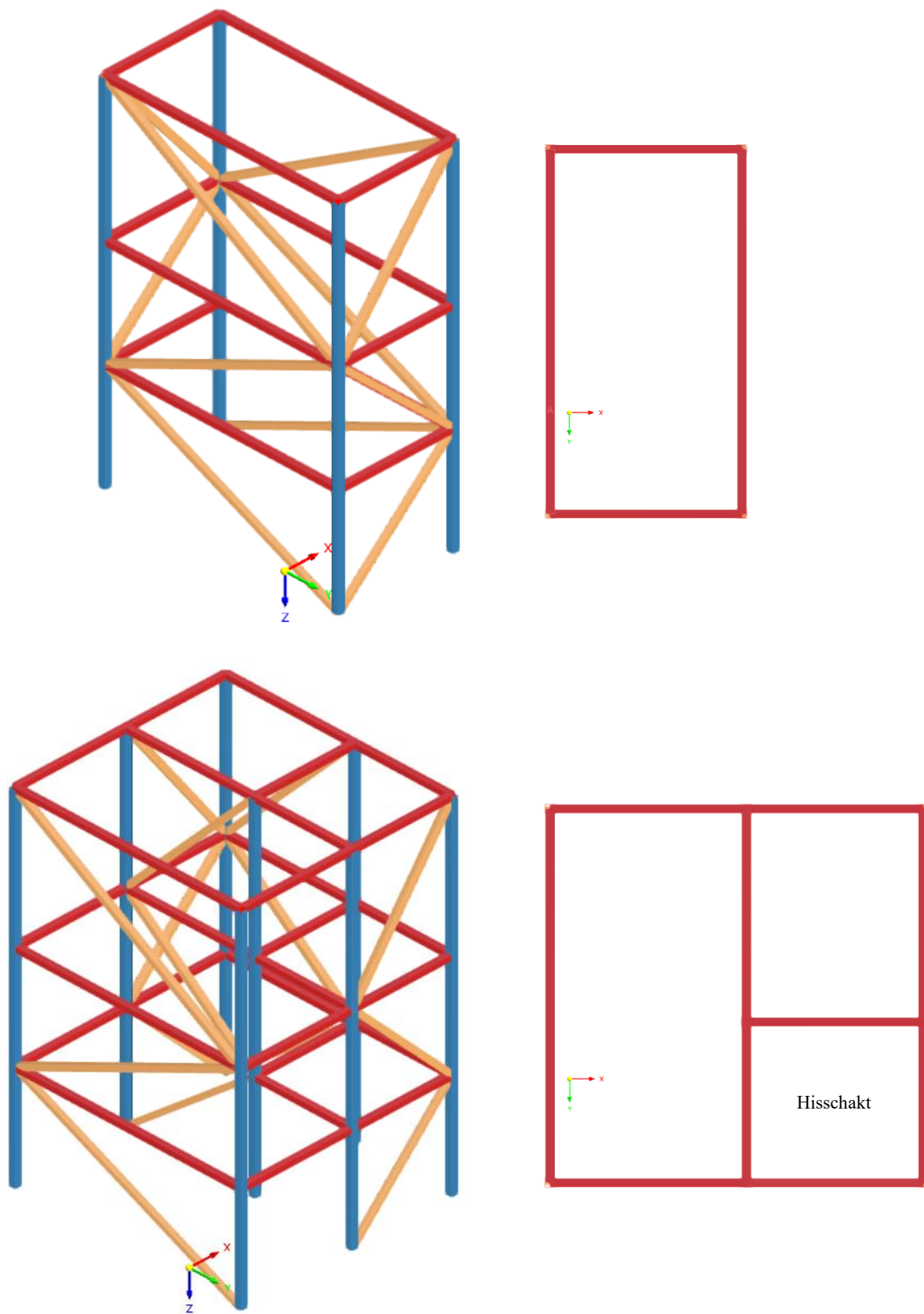
Den typ av trapptorn som skriptet ämnar skapa är en permanent konstruktion där geometrin av den bärande stommen inte påverkas av trappan. Detta skapar större flexibilitet i konstruktionen eftersom trappans utformning, till skillnad från den bärande stommen är strikt styrt av byggnormer. Skapandet av trapptornets geometri kan därför delas upp i två delar; den bärande stommen och trappornas utformning.

5.1.1 Bärande stomme

Trapptornets bärande stomme består av pelare, balkar och vindsträvor vars uppgift är att säkerställa stabilitet och föra ned last till grunden. Lasterna som trapptornet utsätts för är framför allt:

- Egentyngd
- Nyttiglast
- Snölast
- Vindlast

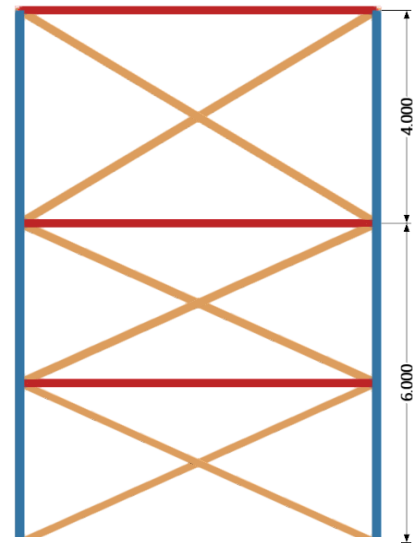
Balkarna tar upp snölast, nyttiglast och egentyngd från trapporna och för den vidare ut till pelarna. Vindlasten tas upp av diagonalerna. Det system som skriptet skapar av pelare, balkar och diagonaler följer ett enhetligt mönster oavsett storlek på tornet, och påverkas enbart av ett eventuellt hisschakt och dess önskade placering. I figur 5.1 visas en 3D-vy och en planvy för två system, ett med hisschakt och ett utan.



Figur 5.1. 3D- och planvy för system med och utan hisschakt.

Ett trappstorn utan hisschakt består av fyra pelare, diagonaler i alla sidor samt balkar fördelade jämnt i höjdlid. Om trappstornet har ett hisschakt får stommen i stället åtta pelare och utöver diagonaler på varje sida får konstruktionen även diagonaler i väggen mellan hisschaktet och trappan. Skriptet bygger upp stommens geometri på följande sätt:

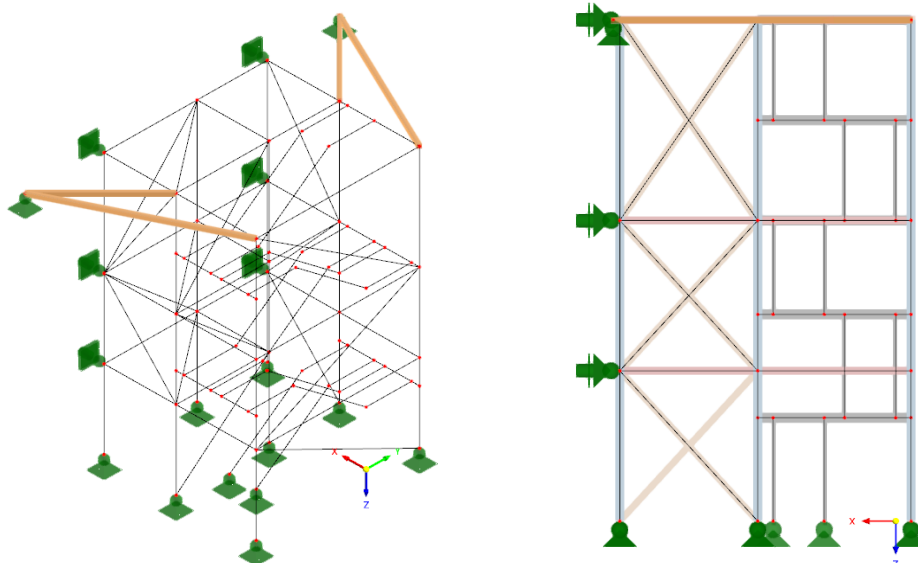
1. Användaren anger höjden på ett eller fler stannplan³
2. Pelare skapas mellan nivå 0 och de angivna höjderna
3. Användaren anger det högsta tillåtna avståndet i höjdlid mellan balkarna
4. Pelarna delas upp med jämna avstånd enligt förgående krav
5. Balkar skapas vid varje pelarslut och förbinds av diagonaler



Figur 5.2. Exempel på balkfördelning.

Om höjderna 6 och 10 meter väljs som stannplan, och den maximalt tillåtna höjden mellan balkarna är satt till 4 m blir resultatet som i figur 5.2 till höger.

Trappstorn har en tendens att vara höga, lätta och smala konstruktioner, vilket gör dem särskilt utsatta för fenomen som stjälpning och vridning. För att förhindra detta är det vanligt att de stabiliseras genom att förankras mot en existerande byggnad. Denna förankring kan genomföras på två huvudsakliga sätt; antingen direkt mot byggnaden vid varje balkram eller genom att använda balkar som löper diagonalt och ansluter till byggnaden i en vinkel. Inom ramen för skriptet ges användaren möjlighet att välja hur denna stagning ska konfigureras. Figur 5.3 illustrerar en kombination av båda dessa stagningssystem.



Figur 5.3. Exempel på en kombination av de två systemen. Upplagen som är vinklade horisontellt representerar förankring vid varje balkram. De beigea balkarna visar diagonaler som ansluter till byggnaden i en vinkel.

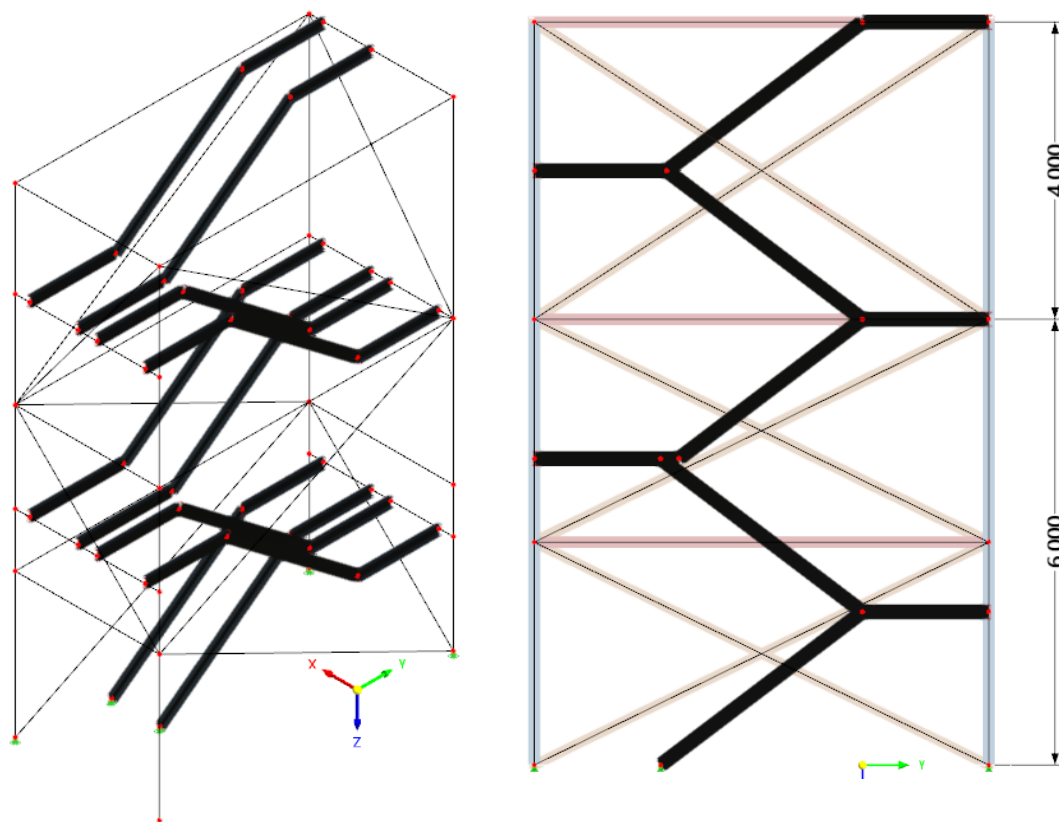
³ Stannplan är en höjd där hissen stannar och trapplopp avslutas, ofta på samma höjd som utgångar.

5.1.2 Trappornas utformning

Trappornas huvudsakliga uppgift är att ge människor säker och bekväm tillgång till de olika stannplanen. För att säkerställa att detta syfte uppfylls på ett tillförlitligt sätt är trappornas utformning strikt reglerade av standarder som till exempel SS-EN ISO 14122–3:2016. Standarden specificerar detaljer som trappans lutning, steghöjd, stegdjup samt andra viktiga säkerhets- och tillgänglighetskrav. Genom att följa dessa normer skapar man funktionella, bekväma och säkra trappor för alla som behöver använda dem. Vid skapandet av trappans geometri har följande krav från SS-EN ISO 14122–3:2016 tagits i beaktning:

- Trapphöjden får inte överstiga 3 meter
- Förhållandet mellan stegdjup (d) och steghöjd (h) uppfyller kravet $600 < 2h \cdot d < 660$
- Stegöverlappning får inte underskrida 10 mm

För att skapa trappornas geometri har en Python-komponent programmerats. Koden tar hänsyn till kraven listade ovan och med hjälp av indata i form av lutning på trappan och höjderna på stannplan bygger den upp trappans geometri. För den intresserade presenteras koden i bilaga C. I figur 5.4 nedan är stannplanen på höjderna 6 och 10 meter.



Figur 5.4. Exempel på trappornas geometri med stannplan på 6 och 10 meter.

Som tidigare påpekats i kapitlet är trappans design oberoende av den bärande strukturen. Detta beror på att de horisontella vangstyckena i varje trappsektion kan monteras på valfri plats längs pelarna med hjälp av upplagsbalkar. Dessa upplagsbalkar är fästa på insidan av pelarna, vilket möjliggör en flexibel placering inom ramverket av den övergripande konstruktionen.

5.2 RFEM 6 indata

Som tidigare nämnts i rapporten kommer konstruktionsberäkningar och analyser utföras i FEM-programvaran RFEM 6. Detta kapitel ämnar beskriva arbetsprocessen i RFEM 6, inklusive de nödvändiga antagandena, de laster som konstruktionen belastas med och hur dessa implementeras i programmet, samt design av stålelement. I RFEM 6 följer man en stegvis arbetsmetod, där man progressivt arbetar igenom faser av geometri, materialval, tvärsnittsdata, lastapplikation, designkontroller och slutligen analys av resultat. Denna process kan även genomföras i Grasshopper genom att använda RFEM-komponenter som är tillgängliga i Grasshopper. I stället för att definiera elementen direkt i RFEM 6, utförs konfigurationen inom Grasshopper.

5.2.1 Geometri och tvärsnitt

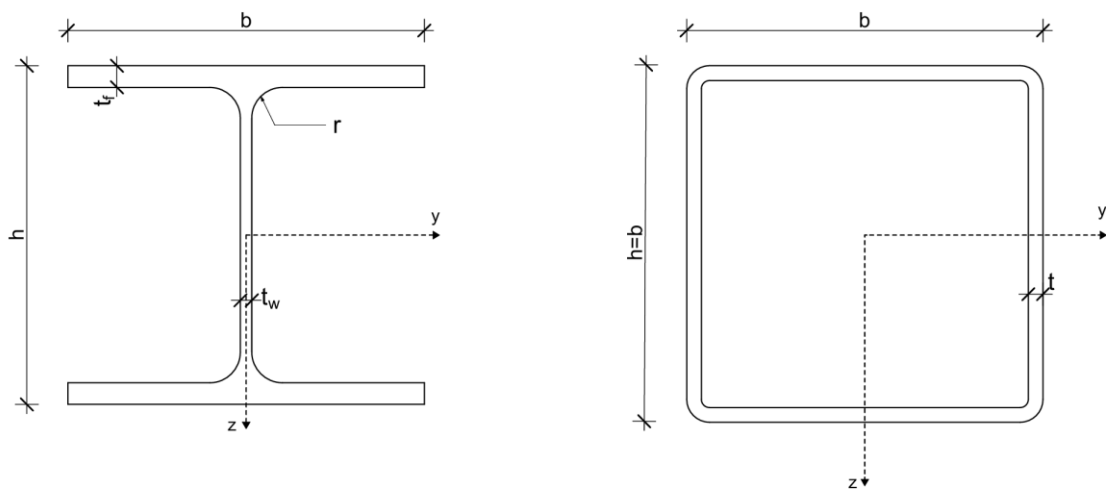
Det inledande steget att skapa geometri utfördes i Grasshopper och utgörs i grund och botten av en modell bestående av linjer utan några egenskaper. Nästa steg blir att applicera ståltvärsnitt på de linjer som har definierats i geometrin. Detta görs med hjälp av RFEM 6 som har en inbyggd databas med standardtvärsnitt. Eftersom modellen är del av ett optimeringsskript och ska hitta profilen med lägsta miljöpåverkan samtidigt som den uppfyller alla designkrav är det nödvändigt för programmet att kunna välja mellan ett urval av olika tvärsnitt. Det är dock inte önskvärt att låta varje individuellt element i modellen variera eftersom det skulle vara opraktiskt och leda till en komplex byggprocess. För att hantera detta grupperas konstruktionselementen i olika kategorier, där variationen av tvärsnittsprofiler begränsas inom varje kategori. Kategorierna delas in i fem olika grupper av konstruktionselement och visas i tabell 5.1 nedan.

Indelningen baseras på två tidigare trapporn som Sweco projekterat samt vägledning från handledare. Vindsträvorna delas in i två grupper eftersom dess belastning varierar mycket med höjden, vilket medför att större dimensioner kan behövas närmare marken. Vindsträvor 1 är de vindsträvor belägna under ungefär en tredjedel av trappornets höjd, och vindsträvor 2 är belägna på de resterande två tredjedelarna.

Tabell 5.1. Visar gruppindelning av konstruktionselement.

Nr.	Grupp
1	Pelare
2	Balkar
3	Vindsträvor 1
4	Vindsträvor 2
5	Vangstycken

Rapporten innehåller som tidigare nämnt en analys som gör en jämförelse mellan den miljömässiga påverkan från öppna och slutna tvärsnittsprofiler. I vår undersökning har vi valt att fokusera på kallformade konstruktionsrör (KKR) och varmformade konstruktionsrör (VKR) för de slutna tvärsnitten och på HEA-balkar för de öppna tvärsnitten, dessa visualiseras i figur 5.5 nedan. HEA-balkar väljs på grund av dess goda knäckningsegenskaper, vilket gör dem lämpliga att använda både som pelare och balkar. Trappans vangstycken och upplag har konsekvent valts till UPE200-profiler i enlighet med branschstandarder. Dessa element kommer inte att ingå i optimeringsprocessen.



Figur 5.5. Visar ritning för öppet- och slutet tvärsnitt

För att effektivisera optimeringsprocessen har urvalet av tvärsnittsdimensioner som varje grupp av konstruktionselement kan välja mellan även begränsats. Precis som vid indelningen av profilgrupper baseras tvärsnittsurläget på två tidigare trapporn samt handledning av Sweco. Detta gav oss en preliminär uppfattning om lämpliga storlekar på för de olika elementen. Med denna informations som grund skapades ett intervall för tvärsnittsdimensioner vilket gör att modellen kan optimeras för trapporn av olika storlekar. Varje grupp av konstruktionselement har nio olika dimensioner att välja mellan. För konstruktionsrören kan godstjockleken variera mellan 5,8 och 10 mm. Vid varje dimensionsökning ökar tvärsnittshöjden med 20 mm. Tabell 5.2 nedan visar intervallerna för varje grupp tvärsnitten.

Tabell 5.2. Intervallindelning för olika konstruktionselement.

Grupp	Slutna (VKR & KKR)	Öppna
1. Pelare	140-180x5, 8, 10	HEA120-280
2. Balkar	120-160x5, 8, 10	HEA100-260
3. Vindsträva 1	120-160x5, 8, 10	HEA100-260
4. Vindsträva 2	100-140x5, 8, 10	HEA100-260
5. Vangstycken	-	UPE200

5.2.2 Laster

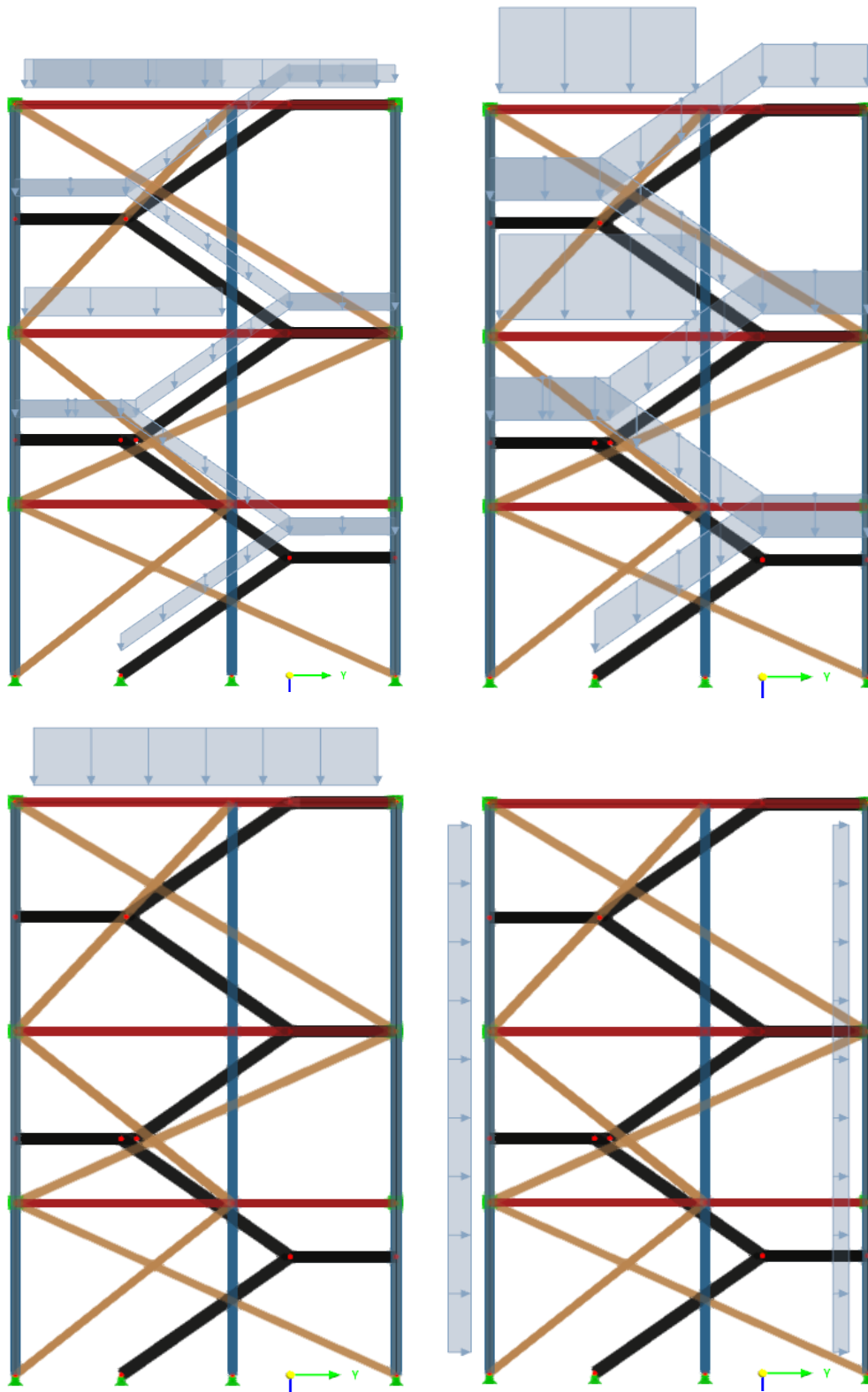
För att kunna dimensionera stålprofilerna i trappornet krävs att rätt lastkonfigurationer är upprättade för byggnaden. Trappornen som används i denna rapport brukar generellt belastas med sex olika lastfall, dessa redovisas i tabell 5.3 nedan med referens till eurokod.

Tabell 5.3. Lastfall, dess beteckning och referens

Lastfall	Beteckning	Referens
Egentyngd Stål	LF1	SS-EN 1991-1-1
Egentyngd från väggar, golv och tak	LF2	SS-EN 1991-1-1
Nyttig last	LF3	SS-EN 1991-1-1
Snö	LF4	SS-EN 1991-1-3
Vind X	LF5	SS-EN 1991-1-4
Vind Y	LF6	SS-EN 1991-1-4

Vanligtvis kan egentyingden från väggar, golv och tak betraktas som ett och samma lastfall, eftersom de tenderar att ha en jämförbar storlek av last per kvadratmeter. Lasterna som orsakas av väggarna fördelas längs konstruktionens yttre delar, där de isolerande väggelementen är installerade. Golvets egentyngd appliceras över trappans vangstycken samt vid de stannplan där hisschakt finns, det vill säga där golvytor är planerade att finnas. Egentyingden från taket appliceras, naturligt nog, endast på det översta våningsplanet. Den nyttiga lasten läggs till på samma platser som golvets egentyngd. Snölasten tillämpas uteslutande på taket och vindlasten tas i beaktning i både x- och y-led. Storleken på dessa laster är det upp till användaren av skriptet att ta fram då de varierar på grund av trappornets position, storlek, krav från beställare med mera. För att visualisera dessa visas lastfall 2, 3, 4 och 6 i figur 5.6 nedan.

Beräkningskontrollerna utförs i RFEM 6 med 12 olika lastkombinationer varav 6 är i brottgränstillståndet och 6 i bruksgränstillståndet. Beräkningskontrollerna i brottgränstillståndet görs med lastkombination STR 6.10b och säkerhetsklass 3 (SIS, 2010), där huvudlasten varierar för de olika kombinationerna. I bruksgränstillståndet används karakteristisk lastkombination 6.14b (SIS, 2010), där elementen kontrolleras för nedböjning. Samtliga lastkombinationer redovisas i bilaga D och för information om vilka brottmoder som analyseras för de specifika elementen, se bilaga E.



Figur 5.6. Lastfall visualiserade.
 Övre raden: Lastfall 2, egentyngd väggar, golv och tak. lastfall 3, nyttig last.
 Undre raden: Lastfall 4, snölast. lastfall 6, Vind i y-riktning.

5.2.3 Ståldesign

Strukturella analyser och kontroller utförs i RFEM 6 med hjälp den tidigare nämnda Eurokod-modulen. Denna modul baseras på SS-EN 1993-1-1:2005, vilket är standarden för dimensionering av stålkonstruktioner. För att kunna utföra dimensionering av element måste vissa antaganden definieras i RFEM 6. Detta görs i Grasshopper med hjälp av RFEM-komponenter. Antagandena är följande:

- Trapporn modelleras med stång- och balkelement. Kategoriseringen av dessa element ser ut enligt tabell 5.4 nedan.

Tabell 5.4. Indelning av elementtyp.

Nr.	Grupp	Elementtyp
1	Pelare	Balk
2	Balkar	Balk
3	Diagonaler 1	Stång
4	Diagonaler 2	Stång
5	Vangstycken	Balk

- För stabilitetskontroller måste knäcklängder och randvillkor definieras. Knutpunkterna mellan elementen antas vara ledade vilket ger knäcklängden $L \cdot \beta$ med $\beta = 1.0$ för samtliga element.
- Vålvningseffekter från vridning kan försummas för stängda tvärsnitt och för öppna tvärsnitt kan effekter från St.Venant vridning försummas enligt SS-EN 1993-1-1:2005 sida 52 avsnitt 6.2.7.
- Vangstyckena till trapporna är bestämda till UPE200 profiler enligt branschstandard och kommer ej dimensioneras.
- Designkontroller i bruksgränstillstånd behandlar deformationer med hänsyn till deformationskrav $L/300$ i godtycklig riktning med karakteristisk lastkombination 6.14b.
- Vindlast beräknas som jämnt utbredd i höjdlängd över hela konstruktionen, det vill säga utan hänsyn till avsnitt 7.2.2 i SS-EN 1991-1-4:2005.

Resultatet från FEM-analysen kontrolleras sedan hjälp av olika Eurokod-modulen för ståldesign, vilket inkluderar alla brottmoder.

5.3 Miljöoptimering av trapptorn

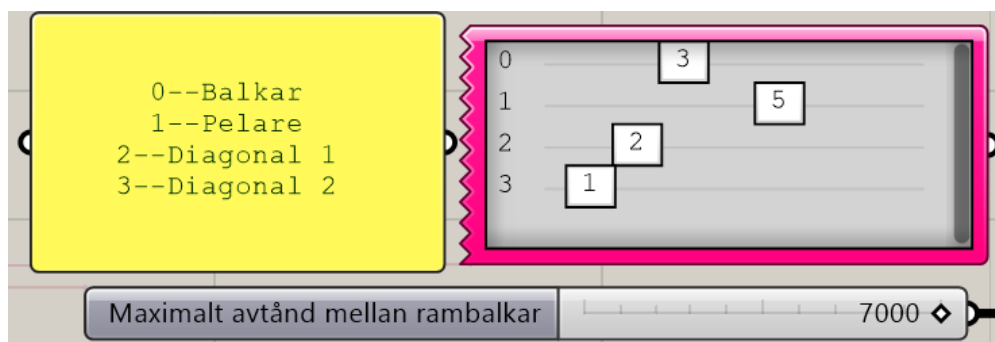
5.3.1 Designvariabler

Det sista steget i designprocessen är att starta optimeringsprocessen för modellen med målet att hitta lösningen som ger så lite CO_2e som möjligt. Som tidigare nämnt i avsnitt 3.1 utgör designvariablerna det som påverkar resultatet av målfunktionen i optimeringsproblemet. Detta optimeringsproblem består av 10 variabler:

- 1 kontinuerlig geometrisk variabel som styr det maximala avståndet mellan balkramarna, vilket i sin tur påverkar antalet balkramar, vinkeln på diagonalerna och pelarnas längd
- 9 kategoriska designvariabler som är antalet möjliga tvärsnitt som respektive grupp kan välja mellan

I Grasshopper representeras dessa variabler av komponenter för sifferreglage (eng. number slider). För den geometriska variabeln "maximalt avstånd i höjddled mellan balkar" utgör sifferreglaget ett intervall för avstånd som kan justeras mellan 4000 och 7000 mm. Denna lösning är inte optimal ur ett optimeringsperspektiv eftersom trapptornets geometri enbart ändras när avståndet når en brytpunkt. Detta diskuteras vidare i kapitel 7.

Designvariablerna är organiserade i en så kallad "genepool", vilket är en samling av sifferreglage som kan anta ett heltal mellan 0 och 9 och tillåter kombinationer av olika parametrar. Varje siffra i genepoolen representerar en specifik typ av tvärsnittsdimension, där värdena rangordnas från större till mindre dimensioner. De tvärsnitt som nämns i avsnitt 5.2.2 är sparade i denna komponent, som visas i figur 5.7.



Figur 5.7. Visar Nummerreglering för maximalt avstånd mellan balkar och en genepool (rosa ram) som representerar specifika tvärsnittsdimensioner.

5.3.2 Formulering av optimeringsproblemet

Med hjälp av optimeringsteorin i kapitel 3 samt ekvation 2.1 kan arbetets optimeringsproblem formuleras enligt följande:

Minimera

$$\Pi_{\text{GWP}} = W \cdot CO_{2e,s} + A \cdot CO_{2e,r} + \sum_{i=1}^n k \cdot P(U_i) \cdot U_i^2 \quad i = 1, 2, \dots, n \quad (5.1)$$

Med hänsyn till begränsningarna

$$\text{Utnyttjandegrad ULS: } \frac{E_d}{R_d} = U_i < 1.0 \quad (5.2)$$

$$\text{Utnyttjandegrad SLS: } \frac{w_{tot,i}}{w_{lim}} = U_i < 1.0 \quad (5.3)$$

$$\text{Där } \sum_{i=1}^n k \cdot P(U_i) \cdot U_i^2 \text{ är strafftermen.} \quad (5.4)$$

Ekvation 5.1 motsvarar optimeringsproblemet målfunktionen (Π_{GWP}) och representerar summan av CO_2 -ekvivalenter från ett trapptors stål och rostskyddsmålning. Syftet med optimeringen är att försöka minimera denna så mycket som möjligt.

U_i representerar utnyttjandegraden för respektive element i konstruktionen. Dessa kommer som utdata från kontrollberäkningarna i RFEM 6 som görs både i brottgränstillstånd (ULS) och i bruksgränstillstånd (SLS). I ULS är utnyttjandegraden baserad på förhållandet mellan den dimensionerande lasten (E_d) för olika brottmoder och elementets dimensionerande hållfasthet (R_d). I SLS är utnyttjandegraden förhållandet mellan total nedböjning för ett element ($w_{tot,i}$) och dess deformationskrav (w_{lim}). Se bilaga D och E för lastkombinationer och de brottmoder som kontrolleras för respektive element.

$P(U_i)$ är indikationsfaktorn relaterad till utnyttjandegraden. Om utnyttjandegraden i ett element är större än 1.0 (100%) adderas ett straff på målfunktionen. Matematiskt kan detta definieras som:

$$P(U_i) = \begin{cases} 1, & U > 1.0 \\ 0, & U \leq 1.0 \end{cases} \quad (5.5)$$

För varje element som överskrider sin utnyttjandegrad adderas en straffterm. Straffaktorn definieras som k i ekvation 5.4, vilken har fastställts till värdet 10^4 . Valet av detta baserats på att storleken på målfunktionen är ungefär i storleken $10^4 - 10^5$. Detta gör att straffet för utnyttjandegrader mycket nära 1.0 inte blir så stora. Syftet är att uppmuntra den genetiska algoritmen att utforska lösningar som närmar sig 1.0.

Sammanfattningsvis kommer Galapagos försöka minimera målfunktionen Π_{GWP} . Om en lösning innehåller element som överskrider en utnyttjandegrad på 100% adderas ett väsentligt straff på målfunktionen. Dessa lösningar kommer rangordnas lägre än de utan straff. Galapagos strävar således mot lösningar som förhåller sig till begränsningarna.

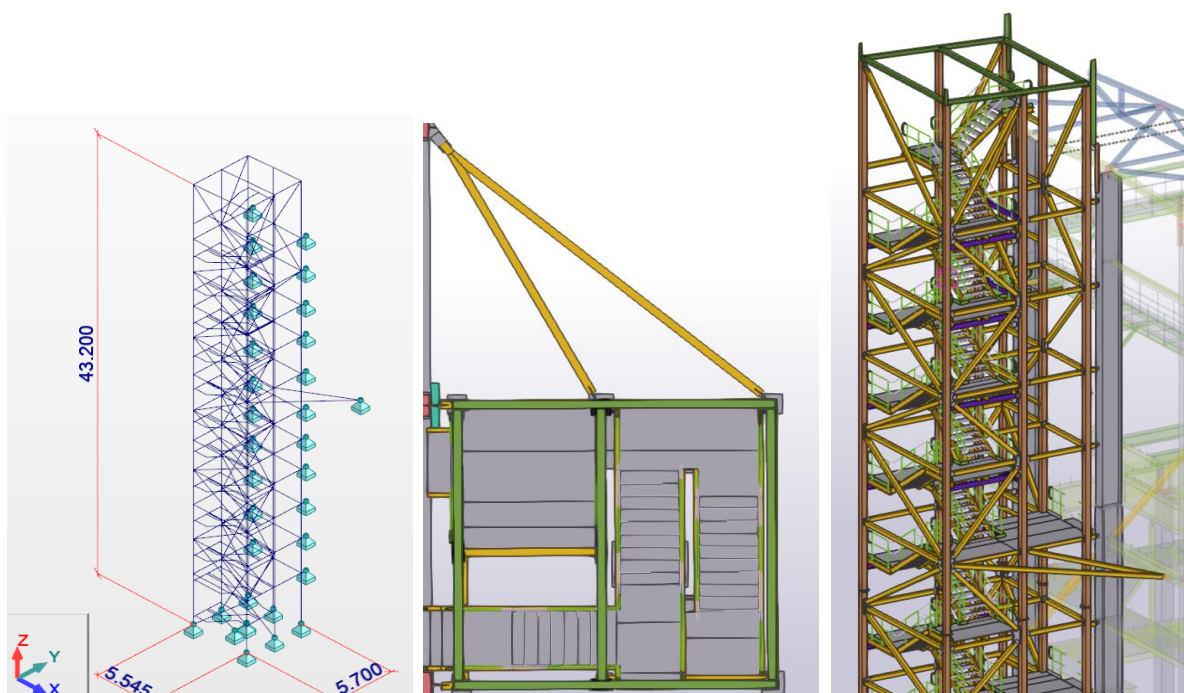
6. Utvärdering av skriptet och optimeringsresultat

För att kunna utvärdera skriptets funktionalitet används ett redan dimensionerat och projekterat trappstorn som referensprojekt. Referensprojektet återskapas med hjälp av skriptet och dess miljöpåverkan beräknas. Därefter används Galapagos för att optimera konstruktionen med tre olika trappstornsstommar, VKR-, KKR-, och HEA-profiler. Slutligen presenteras optimeringsresultaten för de nämnda trappstornsstommarna.

6.1 Garpenberg-projektet

Referensprojektet Garpenberg är ett kombinerat trapp- och hisstorn med dimensionerna 5.545x5.700x43.200 m som konsultföretaget Sweco fick i uppdrag att dimensionera och projektera åt metallföretaget Boliden. Konstruktionen är ett pelar-balk-system som är näst intill identiskt till den geometri som presenterades i kapitel 5.1. Trappstornet har ett hisschakt och består av 8 pelare, 12 horisontella ramar fördelade jämnt i höjddled, samt vindsträvor i väggarna. Konstruktionen stagas mot befintlig byggnad vid varje ram som förhindrar förskjutning i y-led.

Utöver stagning vid varje ram stabiliseras konstruktionen med två horisontella diagonaler som fästs i befintlig byggnad. Dessa är placerade vid konstruktionens halva höjd och tar upp laster i både x- och y-riktning. Beräkningarna för Garpenberg-projektet utfördes av Sweco i FEM-programmet Robot Structural Analysis och kontroller har utförts enligt Eurokod 1993-1-1. Sweco utförde även projekteringen i BIM-programmet Tekla. Modellerna från båda programvarorna presenteras i figur 6.1. Knäcklängden för samtliga pelare och diagonaler är densamma som dess längd och pelarnas upplag är fixlager. Konstruktionen har dimensionerats efter följande laster i tabell 6.1 som är hämtade från beräkningsrapporten.



Figur 6.1. Från vänster till höger: Robot-modell. Planvy på Tekla-modell. 3D-vy på Tekla-modell. Hämtade från Sweco.

Tabell 6.1. Laster som presenteras i beräkningsrapporten för Garpenberg-projektet.

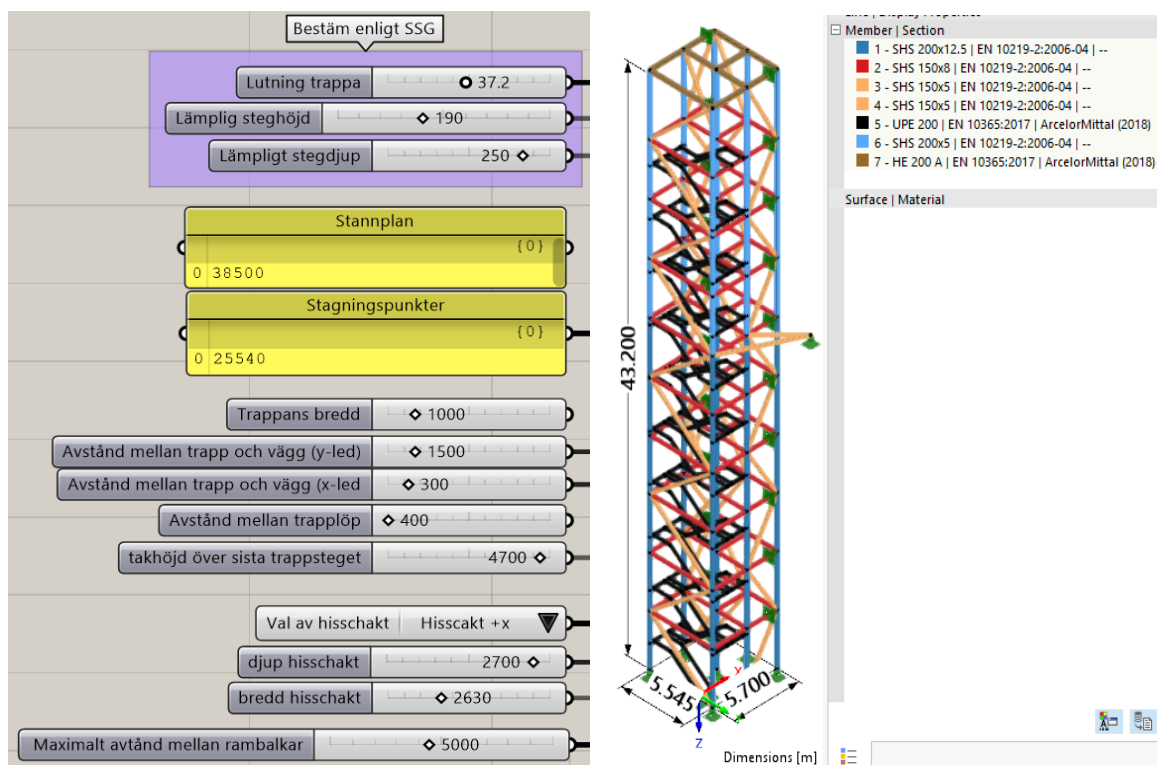
Typ av last	Värde [kN/m ²]
Tak	1.0
Egentyngd trappor	1.0
Nyttig last trappor	3.0
Snö	2.0
Vind lovart	0.74
Vind lä	0.46

Profilerna som användes i stomstålet är uteslutande kallformade konstruktionsrör förutom takbalkarna som är av typ HEA 200. Pelarna är uppdelade på två dimensioner, där den nedre halvan är 200x200x12.5 mm och den övre 200x200x5 mm. Balkarna är genomgående 150x150x8 mm och diagonalerna inklusive stagningsbalkarna är 150x150x5. Alla trappans vangstycken och dess upplag är UPE200. Den modellerade stålmängden för hela konstruktionen är 43 ton. Profilgruppernas högsta utnyttjandegrad presenteras i tabell 6.2 nedan.

Tabell 6.2. Högsta utnyttjandegrad för varje profilgrupp i Robot-modellen av Garpenberg.

Grupp	Utnyttjandegrad
Pelare	33%
Balkar	33%
Diagonal 1	67%
Diagonal 2	23%
Vangstycken	-

I figur 6.2 presenteras indata som används för att återskapa trappornet samt RFEM-modellen som skriptet skapar med den indata. Konstruktionens dimensioner blir 5.54x5.70x43.20m. Profilerna preciserades till desamma som Sweco använde i Garpenberg-projektet och utnyttjandegraderna för respektive profilgrupp presenteras i tabell 6.3.



Figur 6.2. Indata i Grasshopper-skriptet och RFEM-modellen som skriptet skapar.

Tabell 6.3. Högsta utnyttjandegrad för varje profilgrupp i RFEM-modellen som skapats av Grasshopper-skriptet.

Grupp	Utnyttjandegrad
Pelare	30%
Balkar	35%
Diagonal 1	57%
Diagonal 2	33%
Vangstycken	-

Den modellerade stålvikten blir 43,2 ton jämfört med 43 ton som Sweco fick i referensprojektet. Av dessa 43,2 ton motsvarar 7 ton öppna profiler i form av UPE 200 (vangstycken) och resterande kallformade konstruktionsrör. Stålets totala GWP, med värden från hela livscykel (hämtade från kapitel 2.1 tabell 2.2) blir resultatet följande:

$$GWP_{stål} = W_{UPE} \cdot GWP_{öppna} + W_{KKR} \cdot GWP_{KKR} \quad (6.1)$$

$$GWP_{stål} = 7 \cdot 10^3 \cdot 1.18 + 36.2 \cdot 10^3 \cdot 1.35 = 57\,130 \text{ kgCO}_2 \quad (6.2)$$

Konstruktionens täckningsyta är 818.5 m² och med funktionskravet C2 hög hållfasthet som presenterades i kapitel 2.2 bidrar målningen till:

$$GWP_{rostskydd} = \text{täckningsyta} \cdot GWP_{C_2H} \quad (6.3)$$

$$GWP_{rostskydd} = 818.5 \cdot 1.93 = 1\,580 \text{ kgCO}_2e \quad (6.4)$$

Trappornets totala GWP beräknas:

$$GWP_{total} = 57\,130 + 1\,580 = 58\,710 \text{ kgCO}_2e \quad (6.5)$$

6.2 Optimeringsresultat

Vid alla optimeringar användes följande inställningar i Galapagos:

Tabell 4. Visar optimeringsinställningar för Galapagos

Max. stagnant	40
Population	20
Initial Boost	2 x
Maintain	5%
Inbreeding	75%

Resultaten från optimeringarna presenteras i figur 6.3–6.6 och visar tydligt att en stomme med stängda profiler har cirka 11–14% lägre GWP och är 22% lättare än en stomme med HEA-profiler. Optimeringen lyckades även minska Garpenberg-projektets GWP med 45%, från 58 710 kgCO_{2e} till 32 262 kgCO_{2e}.

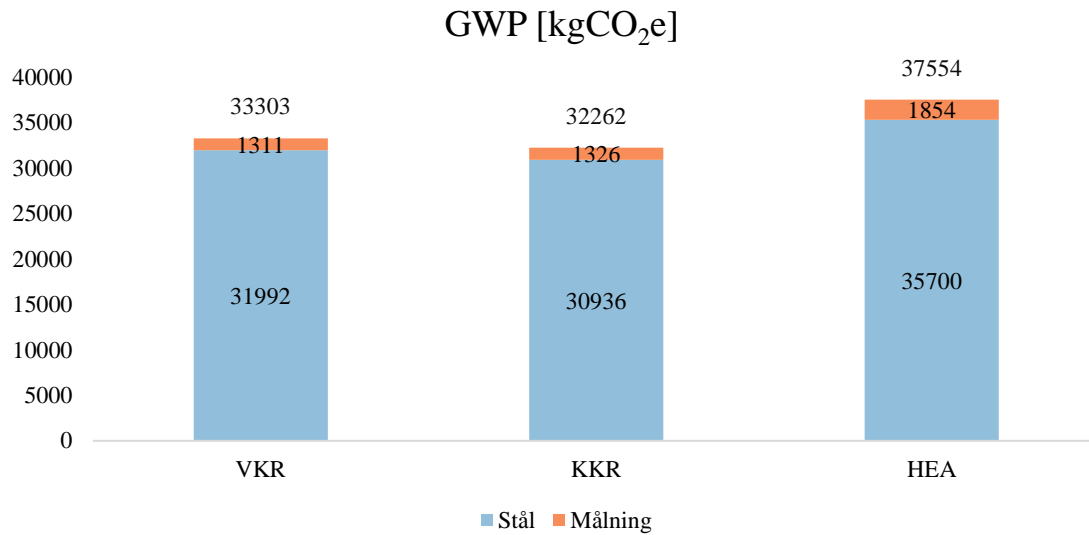
I figur 6.4 presenteras GWP för varje profilgrupp och stomme, där den största skillnaden är HEA-pelarna, som får mellan 2134 och 2732 kgCO_{2e} mer än VKR- och KKR-pelarna. Den minsta skillnaden i GWP återfinns hos balkarna, där VKR-profilerna får högst utsläpp.

En jämförelse i stålvtikt mellan profilgrupp och stomme görs i figur 6.5, där HEA-stommen är tyngst i alla profilgrupper. Återigen är den största skillnaden mellan pelarna, där HEA-pelarna är mellan 3282 och 3364 kg tyngre än de slutna.

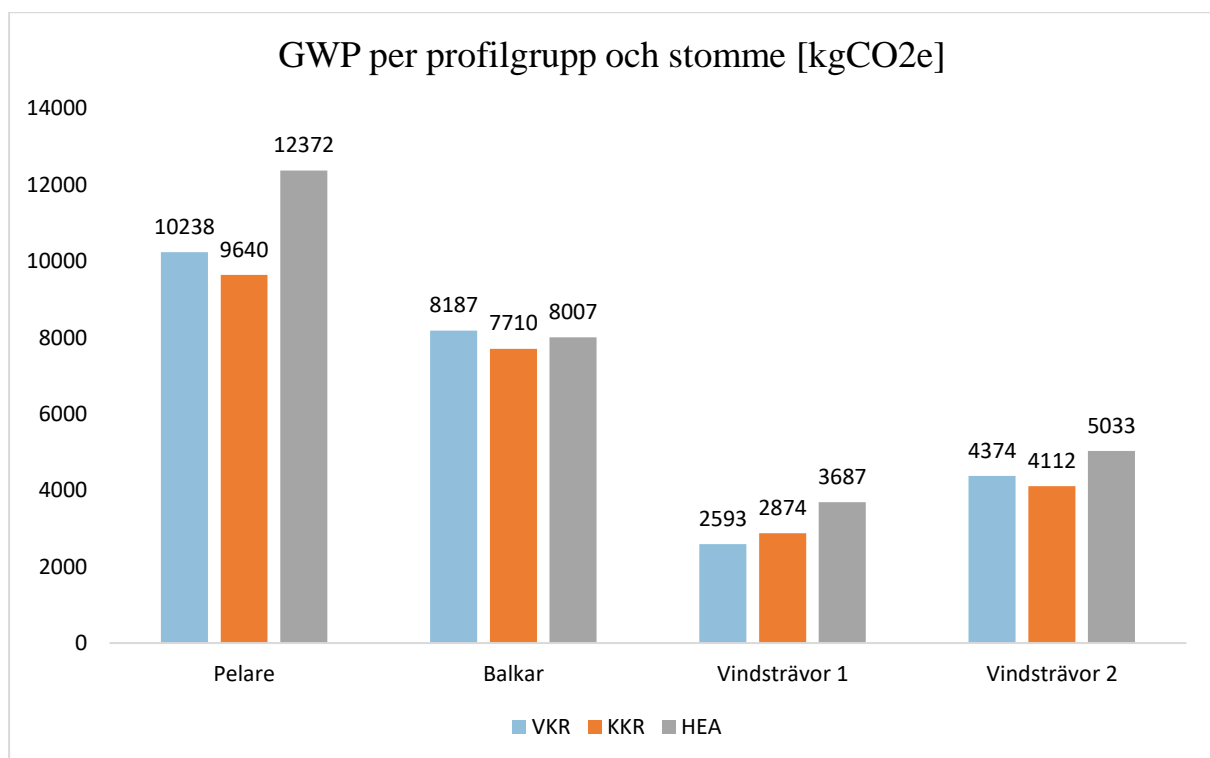
Alla tre trapporn fick samma mängd balkar och därför identisk geometri. I tabell 6.4 presenteras profildimensioner och respektive utnyttjandegrad för de olika profilgrupperna. HEA-stommen uppnår generellt en högre utnyttjandegrad jämfört med de slutna.

Tabell 6.5. Profildimensioner och högsta utnyttjandegrad för olika konstruktionselementgrupper.

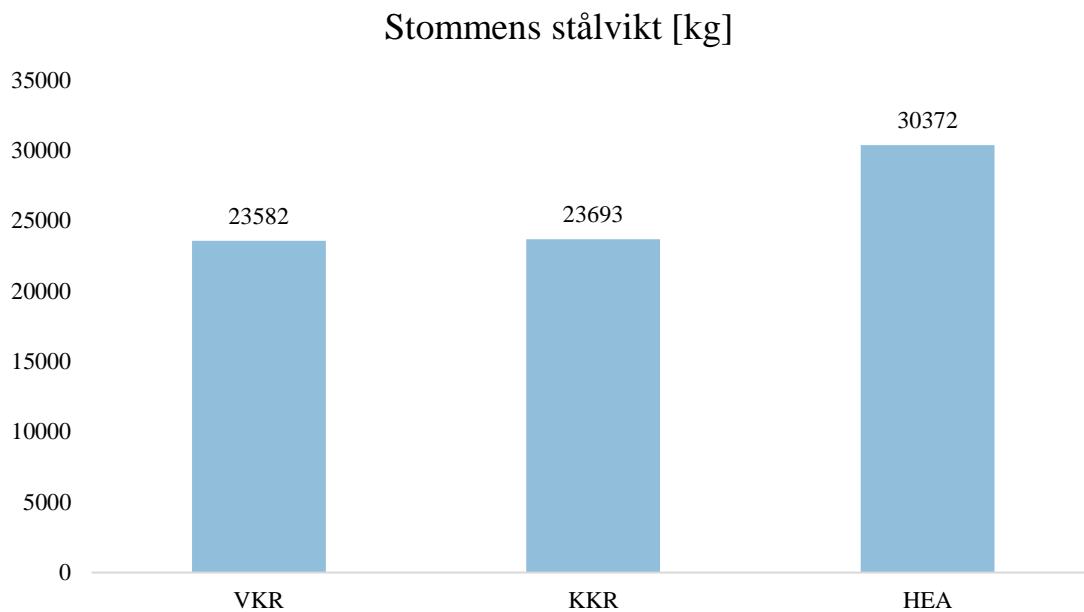
Grupp	VKR		KKR		HEA	
	Dimension	Utnyttjandegrad	Dimension	Utnyttjandegrad	Dimension	Utnyttjandegrad
Pelare	140x5	64%	140x5	78%	HEA160	90%
Balkar	140x5	80%	140x5	81%	HEA140	88%
Diagonal 1	120x5	92%	140x5	72%	HEA160	80%
Diagonal 2	120x5	66%	120x5	79%	HEA140	96%
Vangstycken	UPE200	-	UPE200	-	UPE200	-



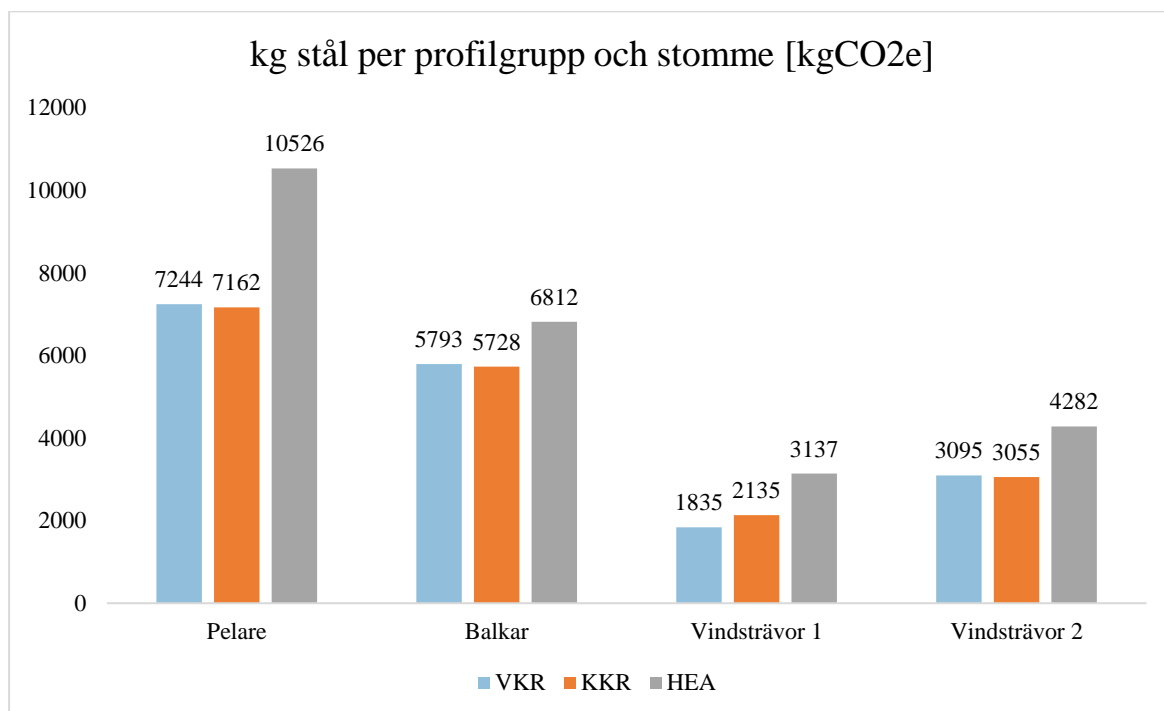
Figur 6.3. GWP-värdet för de olika trapptorsstommarna, visar både stålet och rostskyddsmålningens bidrag.



Figur 6.4. GWP-värdet per profilgrupp i varje stomme (rostskyddsmålning ej inräknat).



Figur 6.5. Stålvikten för de olika trappornsstommarna.



Figur 6.6. Stålvikt per profilgrupp i varje stomme.

7. Diskussion och slutsats

Kapitel 7 är arbetets sista kapitel. Här diskuteras resultaten från förgående kapitel för att besvara frågeställningarna:

- Går parametrisk design att använda som verktyg vid miljöoptimering av stålkonstruktioner?
- Vilken profiltyp i en trappornsstomme resulterar i lägst GWP?

Slutligen presenteras möjligheter till förbättring i arbetet och förslag på framtida forskning.

7.1 Parametrisk design som verktyg vid miljöoptimering

Skriptet lyckades återskapa Garpenberg-projektet med näst intill identisk stålvikt. Det är dock värt att nämna att lastappliceringen och geometrin inte är helt identisk mellan Swecos modell och det som skapas med skriptet, vilket leder till viss osäkerhet i resultatet. Trots detta kan osäkerheten i stor utsträckning förbises med hänsyn till att utnyttjandegraderna i Garpenberg-projektet (tabell 6.1) och i RFEM-modellen innan optimeringen utförs (tabell 6.2) är relativt lika. Optimeringen av projekt Garpenberg blev lyckad, där en optimerad KKR-stomme fick 45% mindre GWP jämfört med den ursprungliga KKR-stommen.

Vi konstaterar därmed att skriptet har på ett framgångsrikt sätt byggt upp och optimerat en trappornsstomme. Skriptet skapade en konstruktion med nästan identisk geometri och stålvikt som Garpenberg-tornet och med hjälp av optimering kunde dess miljöpåverkan kunde nästan halveras.

En annan aspekt som är värd att nämna är hur användning av skriptet påverkar konstruktörens arbete. Att projektera trapporn kan vara mycket tidskrävande, i synnerhet utformningen av trapplöpen. En höjdändring på ett stannplan kan påverka hela trappans geometri och utformning. Genom att konstruktören lätt kan bygga upp en geometri med några enkla inställningar öppnar det upp möjligheten att snabbt testa olika alternativ. Detta kan motivera konstruktören att hitta lösningar som är bättre dimensionerade, vilket i slutändan ger en mer miljöeffektiv konstruktion.

Vi drar där med slutsatsen att parametrisk design är ett bra och effektivt sätt att miljöoptimera trapporn och minska dess klimatavtryck.

7.2 Profilerna ur ett miljöperspektiv

Optimeringsresultaten visar att en stomme med KKR-profiler får lägst GWP och minst stålvikt. VKR-stommen får liknande resultat med enbart 3% högre GWP och HEA-stommen får 16,4% högre GWP jämfört med KKR-stommen. Rostskyddsmålningens inverkan på den totala GWP är relativt låg oavsett profilval och blir i denna jämförelse irrelevant eftersom dess bidrag inte påverkar resultatet.

Optimeringsresultaten visar också att HEA-profilerna uppnådde utnyttjandegrader nära 100%, vilket var högre än för de slutna profilerna. Trots detta resulterade användningen av HEA-

profiler i en tyngre stomme jämfört med de slutna profilerna, där pelarna var den största bidragande faktorn.

Baserat på dessa resultat drar vi slutsatsen att en trappornsstomme byggd med slutna tvärsnitt har en lägre miljöpåverkan än en som är byggd med HEA-profiler.

7.3 Förbättringspotential och framtida forskning

I vårt skript tog varje iteration ca 70 sekunder, vilket är en ohållbart lång tid för denna typ av optimering. Den förlängda iterationstiden berodde främst på att beräkningarna i RFEM 6 var tidskrävande och att import- och exportprocessen av data skapade ytterligare fördröjningar. En potentiell lösning för att minska optimeringstiden kan vara att låta beräkningarna utföras i Grasshopper och genomföra dem med hjälp av ett FEM-plugin, som till exempel Karamba3D. En kortare iterationstid skulle ha möjliggjort analyser med större populationer, vilket i sin tur hade tillåtit utvärdering av fler lösningskombinationer.

Ambitionen var från början att göra en koppling med RFEM 6 eftersom det är en programvara som är Sweco använder. Dock, med tanke på målsättningen med detta arbete, skulle en FEM-plugin möjligtvis varit ett bättre alternativ. Integrationen mellan RFEM 6 och Grasshopper är fortfarande i ett tidigt utvecklingsskede och relativt outforskad när det gäller optimering. Med fortsatta förbättringar och utveckling kan potentialen för denna koppling bli mer effektiv i framtiden.

En annan förbättringspunkt hade varit att ändra på hur sifferreglaget ”maximalt avstånd mellan horisontella balkar” fungerar. Som den fungerar idag ändras enbart geometrin när avståndet når en brytpunkt, vilket betyder att väldigt många värden generar likadana geometrier trots att sifferreglaget har ändrats. Ur ett optimeringsperspektiv hade en bättre lösning varit att i stället låta den fungera som en diskret variabel där varje steg på sifferreglaget representerar olika uppdelningar av höjden med balkar. Detta skulle ha minskat antalet kombinationer av lösningar och eventuellt förenklat sökandet efter optimala resultat.

En ytterligare förbättringsmöjlighet skulle vara att integrera en bättre hänsyn till byggbarheten. Det vore önskvärt att införa en begränsning som straffar målfunktionen för lösningar där balkar eller diagonaler har större dimensioner än pelarna. Detta skulle undvika att resultera i konstruktioner med opraktiska anslutningar, vilket är en viktig faktor att beakta för att säkerställa en smidig och effektiv byggprocess.

Framtida arbeten kan fokusera på att implementera nämnda förbättringsmöjligheter och skulle kunna handla om att:

- Kombinera både öppna och slutna tvärsnitt i samma konstruktion
- Implementera byggbarhet
- Ta hänsyn till svetsning och dess miljöpåverkan
- Undersöka anledningen till att en HEA-stomme får högre utnyttjandegrader men ändå är tyngre än en KKR-stomme

Referenser

- Andersson, S. (2016). Varmformade eller kallformade stålrör, vad är mest lönsamt? Bygg & Teknik.
- BE Group Sverige AB. (2022). Handbok om byggstål. BE Group Sverige AB.
- Brown, D., & Hwang, K.-Y. (1993). *Solving Fixed Configuration Problems with Genetic Search*. Salt Lake City: Springer.
- Chalouhi, E. K. (2022). *Optimal design solutions of road bridges considering embedded environmental impact and cost*. Stockholm: Kungliga Tekniska Högskolan.
- Dlubal. (2024). *Bulding Information Modeling (BIM) | Rhino & Grasshopper*. Hämtat från <https://www.dlubal.com/en/solutions/application-areas/building-information-modeling-bim/rfem-and-rhinoceros-grasshopper> den 3 April 2024
- Dlubal. (2024). *RFEM 6 - Structural of All Types*. Hämtat från <https://www.dlubal.com/en/products/rfem-fea-software/rfem/what-is-rfem> den 3 April 2024
- Erlandsson, M. (2019). Vägledning och råd hur olika aktörer kan bidra till klimatförbättrade byggnader. Stockholm: IVL Svenska Miljöinstitutet.
- Erlandsson, M., Malmqvist, T., Francart, N., & Kellner, J. (2018). Minskad klimatpåverkan från nybygda flerbostadshus. Stockholm: IVL Svenska Miljöinstitutet.
- GlobalABC. (Mars 2024). *Global Status Report for Buildings and Construction - Beyond foundations: Mainstreaming sustainable solutions to cut emissions from the buildings sector*. Hämtat från <https://wedocs.unep.org/20.500.11822/45095> den 17 April 2024
- Isaksson, T., & Mårtensson, A. (2008). *Byggkonstruktion. Regel- och formelsamling* (fjärde uppl.). Studentlitteratur.
- IVL Svenska Miljöinstitutet AB. (den 14 Maj 2024). *IVL*. Hämtat från IVL: <https://www.ivl.se/projekt/kunskapsbank-for-klimat-och-bebyggelse/byggnation/klimatberakning-for-byggnation.html>

- Jernkontoret. (2019). *Jernkontoret*. Hämtat från <https://www.jernkontoret.se/sv/stalindustrin/tillverkning-anvandning-atervinning/processer/> den 18 April 2024
- Jernkontoret. (2022). *Jernkontoret*. Hämtat från <https://www.jernkontoret.se/sv/energi--miljo/energi-energiforsorjning/> den 17 April 2024
- Lampinen, J., & Zelinka, I. (2000). *On stagnation of the differential evolution algorithm*. Brno: Brno University of technology.
- Makert, R., & Alves, G. (2016). Between Designer and Design: Parametric Design and Prototyping Considerations on Gaudí's Sagrada Familia. *Periodica Polytechnica Architecture*.
- Mei, L., & Wang, Q. (2021). *Structural Optimization in Civil Engineering: A Literature Review*. Singapore: Department of Building, School of Design and Environment, National University of Singapore.
- Olofsson, P. (2018). *Recyclingnet*. Hämtat från https://www.recyclingnet.se/article/view/629575/varldens_mest_cirkulara_material den 18 April 2024
- Ottosen, N., & Hans, P. (1992). *Introduction to the finite element method* (1:a uppl.). Pearson.
- Parker, R., & Rardin, R. (1988). *Discrete Optimization* (1 uppl.). San Diego: Academic Press.
- Rao, S. S. (2020). *Engineering Optimization Theory and Practice* (5 uppl.). Miami: Wiley.
- Rawet, P. (2023). *Grönt stål – en riskfylld mångmiljardsatsning*. Hämtat från <https://www.svt.se/nyheter/inrikes/gront-stal-en-riskfylld-mangmiljardsatsning--vzs1qs> den 17 April 2024
- Rutten, D. (2011). *I Eat Bugs For Breakfast*. Hämtat från <https://ieatbugsforbreakfast.wordpress.com/2011/03/04/epatps01/> den 15 mars 2024
- Sawant, S. (2021). *Parametric architecture*. Hämtat från <https://parametric-architecture.com/grasshopper-3d-a-modeling-software-redefining-the-design-process/> den 19 April 2024
- SIS. (den 1 Juni 2010). SVENSK STANDARD SS-EN 1990.
- SIS. (den 29 December 2017). SVENSK STANDARD SS-EN ISO 12944-1:2017.

- SIS. (den 29 12 2017). SVENSK STANDARD SS-EN ISO 12944-2:2017.
- SIS. (den 27 Juni 2018). SVENSK STANDARD SS-EN 1090-2:2018.
- SIS. (den 30 Oktober 2019). SVENSK STANDARD SS-EN ISO 12944-5:2019.
- Tibnor. (2020). EPD Beams. Tibnor.
- Tibnor. (2020). EPD Cold-formed Structural hollow sections. Tibnor.
- Tikkurila. (den 14 Maj 2024). *Tikkurila*. Hämtat från Tikkurila EPD:
<https://tikkurila.se/industri/produkter/epd-verifierade-produkter>
- Toker, U. (2022). *Parametric Methods for Beginners: Architecture Applications* (2022 uppl.). Milton : Taylor & Francis Group.
- USEPA. (den 18 April 2023). *EPA*. Hämtat från
<https://www.epa.gov/ghgemissions/understanding-global-warming-potentials> den 25 Februari 2024
- Van Laarhoven, P. J., & Aarts, E. H. (1987). *Simulated Annealing: Theory and Applications*. Dordrecht: Springer.
- Woerli, B. (2019). *Parametric design and optimization of factory pipe bridges*. Stockholm: KTH Royal Institute of Technology School of Architecture and The Built Environment.
- World Green Building Council. (u.d.). *Bringing embodied carbon upfront*. Hämtat från
<https://worldgbc.org/advancing-net-zero/embodied-carbon/> den 23 Februari 2024
- Yang, X. S. (2010). *Nature - Inspired Metaheuristic Algorithms* (2 uppl.). Cambridge: Luniver Press.
- Yang, X. S. (2014). *Nature-Inspired Optimization Algorithms*. London: Elsevier.

Bilagor

Bilaga A

Cold-formed structural hollow sections (KKR):



Environmental performance

Potential environmental impact

Parameter	Unit	A1 -A3	A4	A5	C1	C2	C3	C4	D
Global Warming Potential (GWP)	kg CO2-eq.	2.58E+00	4.64E-02	6.82E-07	1.97E-02	6.90E-03	2.40E-03	6.81E-04	-1.30E+00
Stratospheric ozone depletion potential (ODP)	kg CFC11-eq.	1.08E-08	1.16E-17	1.71E-22	3.23E-18	1.73E-18	7.98E-18	3.75E-18	3.97E-15
Acidification potential of soil and water (AP)	kg SO2-eq.	5.31E-03	1.98E-04	2.93E-09	7.07E-05	2.96E-05	1.68E-05	4.37E-06	-2.52E-03
Eutrophication potential (EP)	kg PO43--eq.	5.63E-04	4.93E-05	7.31E-10	1.68E-05	7.40E-06	4.05E-06	4.92E-07	-1.75E-04
Formation potential for tropospheric ozone (POCP)	kg Ethene-eq.	3.75E-04	-7.31E-05	-1.09E-09	7.04E-06	-1.10E-05	1.86E-06	3.29E-07	-6.09E-04
Potential for abiotic depletion of non-fossil resources (ADPE)	kg Sb-eq.	2.56E-05	4.31E-09	6.34E-14	1.64E-09	6.41E-10	2.73E-09	2.63E-10	-2.20E-05
Potential for abiotic depletion of fossil fuels (ADPF)	MJ	2.58E+01	6.35E-01	9.34E-06	2.68E-01	9.44E-02	4.67E-02	9.67E-03	-1.22E+01

Hot finished structural hollow sections (VKR):



Environmental performance

Potential environmental impact

Parameter	Unit	A1 -A3	A4	A5	C1	C2	C3	C4	D
Global Warming Potential (GWP)	kg CO2-eq.	2.60E+00	5.26E-02	6.82E-07	2.07E-02	6.84E-03	2.32E-03	1.36E-04	-1.26E+00
Stratospheric ozone depletion potential (ODP)	kg CFC11-eq.	2.96E-11	1.32E-17	1.71E-22	3.41E-18	1.72E-18	7.73E-18	7.50E-19	-8.60E-12
Acidification potential of soil and water (AP)	kg SO2-eq.	6.27E-03	2.24E-04	2.93E-09	7.46E-05	2.93E-05	1.63E-05	8.74E-07	-2.45E-03
Eutrophication potential (EP)	kg PO43--eq.	6.05E-04	5.58E-05	7.31E-10	1.77E-05	7.33E-06	3.92E-06	9.84E-08	-1.72E-04
Formation potential for tropospheric ozone (POCP)	kg Ethene-eq.	9.28E-04	-8.29E-05	-1.09E-09	7.42E-06	-1.09E-05	1.80E-06	6.57E-08	-5.90E-04
Potential for abiotic depletion of non-fossil resources (ADPE)	kg Sb-eq.	-5.48E-07	4.89E-09	6.34E-14	1.73E-09	6.35E-10	2.64E-09	5.25E-11	-2.12E-05
Potential for abiotic depletion of fossil fuels (ADPF)	MJ	2.61E+01	7.20E-01	9.34E-06	2.83E-01	9.36E-02	4.52E-02	1.93E-03	-1.18E+01

Beams (HEA):



Environmental performance

Potential environmental impact

Parameter	Unit	A1 -A3	A4	A5	C1	C2	C3	C4	D
Global Warming Potential (GWP)	kg CO2-eq.	1.15E+00	3.95E-02	6.82E-07	2.14E-02	6.84E-03	2.22E-03	1.36E-04	-3.55E-02
Stratospheric ozone depletion potential (ODP)	kg CFC11-eq.	1.28E-09	9.90E-18	1.71E-22	3.52E-18	1.72E-18	7.39E-18	7.50E-19	-1.31E-11
Acidification potential of soil and water (AP)	kg SO2-eq.	2.30E-03	1.69E-04	2.93E-09	7.69E-05	2.93E-05	1.56E-05	8.74E-07	-6.97E-05
Eutrophication potential (EP)	kg PO43--eq.	2.36E-04	4.20E-05	7.31E-10	1.83E-05	7.33E-06	3.75E-06	9.84E-08	-6.31E-06
Formation potential for tropospheric ozone (POCP)	kg Ethene-eq.	3.90E-04	-6.20E-05	-1.09E-09	7.66E-06	-1.09E-05	1.72E-06	6.57E-08	-1.57E-05
Potential for abiotic depletion of non-fossil resources (ADPE)	kg Sb-eq.	4.50E-07	3.66E-09	6.34E-14	1.78E-09	6.35E-10	2.53E-09	5.25E-11	-4.81E-07
Potential for abiotic depletion of fossil fuels (ADPF)	MJ	1.07E+01	5.40E-01	9.34E-06	2.92E-01	9.36E-02	4.32E-02	1.93E-03	-3.36E-01



ENVIRONMENTAL IMPACT DATA

CORE ENVIRONMENTAL IMPACT INDICATORS – EN 15804+A2, PEF

Impact category	Unit	A1	A2	A3	A1-A3	A4	A5	B1	B2	B3	B4	B5	B6	B7	C1	C2	C3	C4	D
GWP – total ¹⁾	kg CO ₂ e	1,39E+01	4,34E-01	7,79E-01	1,51E+01	7,15E-02	8,32E-03	MND	MND	MND	MND	MND	MND	MND	0,00E+00	0,00E+00	0,00E+00	1,55E+00	-1,24E-01
GWP – fossil	kg CO ₂ e	1,38E+01	4,34E-01	7,85E-01	1,50E+01	7,15E-02	3,77E-03	MND	MND	MND	MND	MND	MND	MND	0,00E+00	0,00E+00	0,00E+00	1,64E+00	-1,27E-01
GWP – biogenic	kg CO ₂ e	9,47E-02	5,14E-06	-6,35E-03	8,83E-02	0,00E+00	4,54E-03	MND	MND	MND	MND	MND	MND	MND	0,00E+00	0,00E+00	0,00E+00	-9,03E-02	2,79E-03
GWP – LULUC	kg CO ₂ e	2,09E-03	1,95E-04	3,83E-04	2,66E-03	3,00E-05	3,74E-06	MND	MND	MND	MND	MND	MND	MND	0,00E+00	0,00E+00	0,00E+00	5,78E-04	-1,84E-05
Ozone depletion pot.	kg CFC ₁₁ e	1,01E-06	9,78E-08	1,36E-07	1,24E-06	1,69E-08	3,69E-10	MND	MND	MND	MND	MND	MND	MND	0,00E+00	0,00E+00	0,00E+00	2,19E-07	-4,94E-09
Acidification potential	mol H ⁺ e	1,81E-01	4,46E-03	3,09E-03	1,89E-01	2,13E-04	1,51E-05	MND	MND	MND	MND	MND	MND	MND	0,00E+00	0,00E+00	0,00E+00	3,85E-03	-5,19E-04
EP-freshwater ²⁾	kg Pe	4,78E-02	2,70E-06	9,02E-05	4,79E-02	5,42E-07	1,06E-07	MND	MND	MND	MND	MND	MND	MND	0,00E+00	0,00E+00	0,00E+00	1,33E-05	-5,22E-06
EP-marine	kg Ne	2,57E-02	1,17E-03	5,94E-04	2,75E-02	4,38E-05	5,12E-06	MND	MND	MND	MND	MND	MND	MND	0,00E+00	0,00E+00	0,00E+00	8,08E-04	-1,06E-04
EP-terrestrial	mol Ne	2,69E-01	1,30E-02	6,51E-03	2,88E-01	4,87E-04	3,94E-05	MND	MND	MND	MND	MND	MND	MND	0,00E+00	0,00E+00	0,00E+00	9,13E-03	-1,24E-03
POCP (“smog”) ³⁾	kg NMVOCe	7,13E-02	3,58E-03	4,97E-03	7,99E-02	1,89E-04	2,48E-01	MND	MND	MND	MND	MND	MND	MND	0,00E+00	0,00E+00	0,00E+00	2,60E-03	-6,32E-04
ADP-minerals & metals ⁴⁾	kg Sbe	3,77E-05	1,28E-06	4,69E-06	4,37E-05	2,55E-07	7,47E-08	MND	MND	MND	MND	MND	MND	MND	0,00E+00	0,00E+00	0,00E+00	6,80E-06	-2,37E-06
ADP-fossil resources	MJ	4,33E+01	6,25E+00	5,89E+00	5,55E+01	1,09E+00	3,56E-02	MND	MND	MND	MND	MND	MND	MND	0,00E+00	0,00E+00	0,00E+00	1,22E+01	-1,21E+00
Water use ⁵⁾	m ³ e depr.	1,59E+00	2,67E-02	5,14E-01	2,13E+00	5,33E-03	6,92E-04	MND	MND	MND	MND	MND	MND	MND	0,00E+00	0,00E+00	0,00E+00	2,47E-01	-2,43E-02

Bilaga C

```
from math import tan, radians, atan, pi

target_angle_degrees = x

def optimize_stairs(floor_height_mm, optimal_steps, min_step_height, max_step_height):
    target_angle_radians = radians(target_angle_degrees)
    smallest_difference = float('inf')
    best_fit_configuration = None

    min_step_height_mm = min_step_height * 10000
    max_step_height_mm = max_step_height * 10000

    for step_height_mm in range(min_step_height_mm, max_step_height_mm + 1):
        step_height = step_height_mm / 10000
        required_step_width_mm = step_height / tan(target_angle_radians)
        initial_step_width_mm = 260
        overlapping = initial_step_width_mm - required_step_width_mm

        if overlapping < 9.5:
            step_width_mm = 300
        elif overlapping > 39.5:
            step_width_mm = 230
        else:
            step_width_mm = initial_step_width_mm

        overlapping = step_width_mm - required_step_width_mm
        total_steps_possible = int(floor_height_mm // step_height)
        total_height_current = total_steps_possible * step_height
        difference = abs(floor_height_mm - total_height_current)

        if total_height_current < floor_height_mm:
            extra_steps = (floor_height_mm - total_height_current) // step_height
            total_steps_possible += extra_steps
            total_height_current += extra_steps * step_height
            difference = abs(floor_height_mm - total_height_current)

        if difference < smallest_difference:
            smallest_difference = difference
            best_fit_configuration = {
                'step_height_mm': step_height_mm / 10000,
                'step_height': step_height,
                'step_width': step_width_mm,
                'total_steps': total_steps_possible,
                'total_height': total_height_current,
                'difference': difference,
                'overlapping': overlapping,
                'vinkel': tan(step_height/(step_width_mm-difference))
            }

    if best_fit_configuration:
        flights_with_max_steps = best_fit_configuration['total_steps'] // optimal_steps
        steps_in_last_flight = best_fit_configuration['total_steps'] % optimal_steps
        flights_info = [optimal_steps] * int(flights_with_max_steps)
        if steps_in_last_flight > 0:
            flights_info.append(steps_in_last_flight)

        total_flights = len(flights_info)
        avg_steps_per_flight = best_fit_configuration['total_steps'] // total_flights
        extra_steps = best_fit_configuration['total_steps'] % total_flights
        flights_info = [avg_steps_per_flight + (1 if i < extra_steps else 0) for i in
            range(total_flights)]

        best_fit_configuration.update({
```

```

        'total_flights': total_flights,
        'steps_per_flight': flights_info,
        'diff_height': floor_height_mm - best_fit_configuration['total_height']
    })
    return best_fit_configuration
else:
    return None

# Samla in resultat för varje fixpunktshöjd
results = []
accumulated_height = 0 # Ackumulerad höjd från marken

for i, fixpoint_height in enumerate(floor_heights):
    if i == 0:
        section_height = fixpoint_height
    else:
        section_height = fixpoint_height - floor_heights[i - 1]

    result = optimize_stairs(section_height, optimal_steps, min_step_height,
max_step_height)
    if result:
        accumulated_height += result['total_height'] # Uppdatera den ackumulerade
höjden
        diff_height = fixpoint_height - accumulated_height # Differens mellan fixpunkt
och ackumulerad höjd
        if diff_height > 0.5:
            print("Ingen lösning hittad för avståndet mellan fixpunkterna. (Om avståndet
är större än 2000 mm kommer alltid en lösning att hittas).")
            break
        else:
            results.append(result)
            print("Resultat för fixpunkt {}".format(i + 1))
            print("Beräknad vinkel (grader): {:.2f}".format(atan(result['step_height'] /
(result['step_width'] - result['overlapping']))) * (180/pi)))
            print("Önskad fixhöjd (mm): {}".format(fixpoint_height))
            print("Beräknad fixhöjd (mm): {}".format(accumulated_height))
            print("Differens (mm): {:.2f}".format(diff_height))
            print("Steghöjd (mm): {:.2f}".format(result['step_height_mm']))
            print("Totalt antal steg: {}".format(result['total_steps']))
            print("Antal steg per trapplop: {}".format(result['steps_per_flight']))
            print("Stegbredd (mm): {}".format(result['step_width']))
            print("Överlapp (mm): {:.2f}".format(result['overlapping']))
            print("----- ")
    else:
        print("Ingen lösning hittades för sektionshöjd {}, kontrollera
indata.".format(section_height))

import Rhino.Geometry as rg

def create_stair_points_and_lines(optimize_results, stair_width):
    points = []
    lines_forward_slash = []
    lines_back_slash = []
    lines_forward_slash_parallel = []
    lines_back_slash_parallel = []
    z = 0
    alternate = mirror_stair
    for result in optimize_results:
        if result is not None:
            step_height = result['step_height']
            steps_per_flight = result['steps_per_flight']
            step_width = result['step_width'] - result['overlapping']

            for flight_steps in steps_per_flight:

```

```

        flight_width = flight_steps * step_width

        if alternate:
            # Linje som går / utan x-förflyttning
            start_point = rg.Point3d(stair_width+x_offset, flight_width, z)
            end_point = rg.Point3d(stair_width+x_offset, 0, z + (flight_steps *
step_height))

            lines_forward_slash.append(rg.Line(start_point, end_point))
            # Parallell linje för / utan x-förflyttning
            start_point_parallel = rg.Point3d(stair_width*2+x_offset,
flight_width, z)
            end_point_parallel = rg.Point3d(stair_width*2+x_offset, 0, z +
(flight_steps * step_height))
            lines_forward_slash_parallel.append(rg.Line(start_point_parallel,
end_point_parallel))
        else:
            # Linje som går \ med x-förflyttning från origo
            start_point = rg.Point3d(0, 0, z)
            end_point = rg.Point3d(0, flight_width, z + (flight_steps *
step_height))

            lines_back_slash.append(rg.Line(start_point, end_point))
            # Parallell linje för \ med x-förflyttning
            start_point_parallel = rg.Point3d(stair_width, 0, z)
            end_point_parallel = rg.Point3d(stair_width, flight_width, z +
(flight_steps * step_height))
            lines_back_slash_parallel.append(rg.Line(start_point_parallel,
end_point_parallel))

            # Lägg till punkterna i listan över punkter
            points.extend([start_point, end_point, start_point_parallel,
end_point_parallel])

            # Växla alterneringen för nästa trapplop
            alternate = not alternate

            # Uppdatera z till höjden inför nästa trapplop
            z += (flight_steps * step_height)

    return points, lines_forward_slash, lines_back_slash, lines_forward_slash_parallel,
lines_back_slash_parallel

# Anrop till funktionen, där stair_width nu är den enda parametern
points, lines_forward_slash, lines_back_slash, lines_forward_slash_parallel,
lines_back_slash_parallel = create_stair_points_and_lines(results, stair_width)

# Variabler för utdata
a = points
b = lines_forward_slash
c = lines_back_slash
d = lines_forward_slash_parallel
e = lines_back_slash_parallel

```

Bilaga D

Lastfall	Beteckning	Referens
Egentyngd Stål	LF1	SS-EN 1991-1-1
Egentyngd från väggar, golv och tak	LF2	SS-EN 1991-1-1
Nyttig last	LF3	SS-EN 1991-1-1
Snö	LF4	SS-EN 1991-1-3
Vind X	LF5	SS-EN 1991-1-4
Vind Y	LF6	SS-EN 1991-1-4

Lastkombinationer

CO1 Nyttig last	$1.20 \cdot LF1 + 1.20 \cdot LF2 + 1.50 \cdot LF3 + 1.05 \cdot LF4 + 0.45 \cdot LF6$
CO2 Snö	$1.20 \cdot LF1 + 1.20 \cdot LF2 + 1.05 \cdot LF3 + 1.50 \cdot LF4 + 0.45 \cdot LF6$
CO3 Vind + X	$1.20 \cdot LF1 + 1.20 \cdot LF2 + 1.05 \cdot LF3 + 1.05 \cdot LF4 + 1.50 \cdot LF5$
CO4 Vind -X	$1.20 \cdot LF1 + 1.20 \cdot LF2 + 1.05 \cdot LF3 + 1.05 \cdot LF4 - 1.50 \cdot LF5$
CO5 Vind +Y	$1.20 \cdot LF1 + 1.20 \cdot LF2 + 1.05 \cdot LF3 + 1.05 \cdot LF4 + 1.50 \cdot LF6$
CO6 Vind -Y	$1.20 \cdot LF1 + 1.20 \cdot LF2 + 1.05 \cdot LF3 + 1.05 \cdot LF4 - 1.50 \cdot LF6$
CO7 Nyttig last	$1.0 \cdot LC1 + 1.0 \cdot LC2 + 1.0 \cdot LC3 + 0.7 \cdot LC4 + 0.3 \cdot LF6$
CO8 Snö	$1.0 \cdot LF1 + 1.0 \cdot LF2 + 0.7 \cdot LF3 + 1.0 \cdot LF4 + 0.3 \cdot LF6$
CO9 Vind +X	$1.0 \cdot LF1 + 1.0 \cdot LF2 + 0.7 \cdot LF3 + 0.7 \cdot LF4 + 1.0 \cdot LF5$
CO10 Vind -X	$1.0 \cdot LF1 + 1.0 \cdot LF2 + 0.7 \cdot LF3 + 0.7 \cdot LF4 - 1.0 \cdot LF5$
CO11 Vind +Y	$1.0 \cdot LF1 + 1.0 \cdot LF2 + 0.7 \cdot LF3 + 0.7 \cdot LF4 + 1.0 \cdot LF6$
CO12 Vind -Y	$1.0 \cdot LF1 + 1.0 \cdot LF2 + 0.7 \cdot LF3 + 0.7 \cdot LF4 - 1.0 \cdot LF6$

CO1	6.10b Nyttig last som huvudlast (ULS)
CO2	6.10b Snölast som huvudlast (ULS)
CO3	6.10b Vind i positiv x-riktning (ULS)
CO4	6.10b Vind i negativ x-riktning (ULS)
CO5	6.10b Vind i positiv y-riktning (ULS)
CO6	6.10b Vind i negativ y-riktning (ULS)
CO7	6.14b Nyttig last som huvudlast (SLS)
CO8	6.14b Snölast som huvudlast (SLS)
CO9	6.14b Vind i positiv x-riktning (SLS)
CO10	6.14b Vind i negativ x-riktning (SLS)
CO11	6.14b Vind i positiv y-riktning (SLS)
CO12	6.14b Vind i negativ y-riktning (SLS)

Bilaga E

Skärmbilderna nedan visar brottmoder för olika element från beräkningarna i RFEM 6. Längst till höger i varje rad finns referens till Eurokod.

HEA – Pelare

Design Check Type	Description
SP1100.00	Section Proof Tension acc. to EN 1993-1-1, 6.2.3
SP1200.00	Section Proof Compression acc. to EN 1993-1-1, 6.2.4
SP3100.02	Section Proof Shear in z-axis acc. to EN 1993-1-1, 6.2.6(2) Plastic design
SP3200.02	Section Proof Shear in y-axis acc. to EN 1993-1-1, 6.2.6(2) Plastic design
SP3500.00	Section Proof Shear buckling acc. to EN 1993-1-5, 5.1, 5.2, 5.3 and 5.5
SP4100.03	Section Proof Bending about y-axis acc. to EN 1993-1-1, 6.2.5 Plastic design
SP5100.03	Section Proof Bending about z-axis acc. to EN 1993-1-1, 6.2.5 Plastic design
SP6500.01	Section Proof Biaxial bending, axial force and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
SP6500.02	Section Proof Bending about y-axis, axial force and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
SP6500.03	Section Proof Bending about z-axis, axial force and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
SP6500.04	Section Proof Biaxial bending and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
ST1100.00	Stability Flexural buckling about principal y-axis acc. to EN 1993-1-1, 6.3.1
ST1300.00	Stability Flexural buckling about principal z-axis acc. to EN 1993-1-1, 6.3.1
ST1500.00	Stability Torsional buckling acc. to EN 1993-1-1, 6.3.1
ST3100.00	Stability Bending and buckling about principal axes acc. to EN 1993-1-1, 6.3.3
SE0100.00	Serviceability Negligible deflections
SE1100.00	Serviceability Deflections in z-direction
SE1200.00	Serviceability Deflections in y-direction

HEA – Balkar

Design Check Type	Description
SP0100.00	Section Proof Negligible internal forces
SP1100.00	Section Proof Tension acc. to EN 1993-1-1, 6.2.3
SP1200.00	Section Proof Compression acc. to EN 1993-1-1, 6.2.4
SP3100.02	Section Proof Shear in z-axis acc. to EN 1993-1-1, 6.2.6(2) Plastic design
SP3200.02	Section Proof Shear in y-axis acc. to EN 1993-1-1, 6.2.6(2) Plastic design
SP3500.00	Section Proof Shear buckling acc. to EN 1993-1-5, 5.1, 5.2, 5.3 and 5.5
SP4100.03	Section Proof Bending about y-axis acc. to EN 1993-1-1, 6.2.5 Plastic design
SP5100.03	Section Proof Bending about z-axis acc. to EN 1993-1-1, 6.2.5 Plastic design
SP6500.01	Section Proof Biaxial bending, axial force and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
SP6500.02	Section Proof Bending about y-axis, axial force and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
SP6500.03	Section Proof Bending about z-axis, axial force and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
SP6500.04	Section Proof Biaxial bending and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
ST1100.00	Stability Flexural buckling about principal y-axis acc. to EN 1993-1-1, 6.3.1
ST1300.00	Stability Flexural buckling about principal z-axis acc. to EN 1993-1-1, 6.3.1
ST1500.00	Stability Torsional buckling acc. to EN 1993-1-1, 6.3.1
ST2100.00	Stability Lateral torsional buckling acc. to EN 1993-1-1, 6.3.2
ST3100.00	Stability Bending and buckling about principal axes acc. to EN 1993-1-1, 6.3.3
SE0100.00	Serviceability Negligible deflections
SE1100.00	Serviceability Deflections in z-direction
SE1200.00	Serviceability Deflections in y-direction

HEA – Vindsträvor

Design Check Type	Description
SP1100.00	Section Proof Tension acc. to EN 1993-1-1, 6.2.3
SP1200.00	Section Proof Compression acc. to EN 1993-1-1, 6.2.4
SP3100.02	Section Proof Shear in z-axis acc. to EN 1993-1-1, 6.2.6(2) Plastic design
SP3500.00	Section Proof Shear buckling acc. to EN 1993-1-5, 5.1, 5.2, 5.3 and 5.5
SP4100.03	Section Proof Bending about y-axis acc. to EN 1993-1-1, 6.2.5 Plastic design
SP6500.02	Section Proof Bending about y-axis, axial force and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
ST1100.00	Stability Flexural buckling about principal y-axis acc. to EN 1993-1-1, 6.3.1
ST1300.00	Stability Flexural buckling about principal z-axis acc. to EN 1993-1-1, 6.3.1
ST1500.00	Stability Torsional buckling acc. to EN 1993-1-1, 6.3.1
ST2100.00	Stability Lateral torsional buckling acc. to EN 1993-1-1, 6.3.2
ST3100.00	Stability Bending and buckling about principal axes acc. to EN 1993-1-1, 6.3.3
SE0100.00	Serviceability Negligible deflections
SE1100.00	Serviceability Deflections in z-direction

VKR & KKR - Pelare

Design Check Type	Description
SP0100.00	Section Proof Negligible internal forces
SP1100.00	Section Proof Tension acc. to EN 1993-1-1, 6.2.3
SP1200.00	Section Proof Compression acc. to EN 1993-1-1, 6.2.4
SP2100.00	Section Proof Torsion acc. to EN 1993-1-1, 6.2.7
SP3100.01	Section Proof Shear in z-axis and torsion acc. to EN 1993-1-1, 6.2.7(9) Plastic design
SP3100.02	Section Proof Shear in z-axis acc. to EN 1993-1-1, 6.2.6(2) Plastic design
SP3200.01	Section Proof Shear in y-axis and torsion acc. to EN 1993-1-1, 6.2.7(9) Plastic design
SP3200.02	Section Proof Shear in y-axis acc. to EN 1993-1-1, 6.2.6(2) Plastic design
SP3500.00	Section Proof Shear buckling acc. to EN 1993-1-5, 5.1, 5.2, 5.3 and 5.5
SP4100.03	Section Proof Bending about y-axis acc. to EN 1993-1-1, 6.2.5 Plastic design
SP5100.03	Section Proof Bending about z-axis acc. to EN 1993-1-1, 6.2.5 Plastic design
SP6100.00	Section Proof Axial and shear stress acc. to EN 1993-1-1, 6.2.1(5) Elastic design
SP6500.01	Section Proof Biaxial bending, axial force and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
SP6500.02	Section Proof Bending about y-axis, axial force and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
SP6500.03	Section Proof Bending about z-axis, axial force and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
SP6500.04	Section Proof Biaxial bending and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
ST1100.00	Stability Flexural buckling about principal y-axis acc. to EN 1993-1-1, 6.3.1
ST1300.00	Stability Flexural buckling about principal z-axis acc. to EN 1993-1-1, 6.3.1
ST2100.00	Stability Lateral torsional buckling acc. to EN 1993-1-1, 6.3.2
ST3100.00	Stability Bending and buckling about principal axes acc. to EN 1993-1-1, 6.3.3
SE0100.00	Serviceability Negligible deflections
SE1100.00	Serviceability Deflections in z-direction
SE1200.00	Serviceability Deflections in y-direction
WA5001.00	Warning Torsion is neglected for stability design checks

VKR & KKR – Balkar

Design Check Type	Description
SP0100.00	Section Proof Negligible internal forces
SP1100.00	Section Proof Tension acc. to EN 1993-1-1, 6.2.3
SP1200.00	Section Proof Compression acc. to EN 1993-1-1, 6.2.4
SP2100.00	Section Proof Torsion acc. to EN 1993-1-1, 6.2.7
SP3100.01	Section Proof Shear in z-axis and torsion acc. to EN 1993-1-1, 6.2.7(9) Plastic design
SP3100.02	Section Proof Shear in z-axis acc. to EN 1993-1-1, 6.2.6(2) Plastic design
SP3200.01	Section Proof Shear in y-axis and torsion acc. to EN 1993-1-1, 6.2.7(9) Plastic design
SP3200.02	Section Proof Shear in y-axis acc. to EN 1993-1-1, 6.2.6(2) Plastic design
SP3500.00	Section Proof Shear buckling acc. to EN 1993-1-5, 5.1, 5.2, 5.3 and 5.5
SP4100.03	Section Proof Bending about y-axis acc. to EN 1993-1-1, 6.2.5 Plastic design
SP5100.03	Section Proof Bending about z-axis acc. to EN 1993-1-1, 6.2.5 Plastic design
SP6100.00	Section Proof Axial and shear stress acc. to EN 1993-1-1, 6.2.1(5) Elastic design
SP6500.01	Section Proof Biaxial bending, axial force and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
SP6500.02	Section Proof Bending about y-axis, axial force and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
SP6500.03	Section Proof Bending about z-axis, axial force and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
SP6500.04	Section Proof Biaxial bending and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
ST1100.00	Stability Flexural buckling about principal y-axis acc. to EN 1993-1-1, 6.3.1
ST1300.00	Stability Flexural buckling about principal z-axis acc. to EN 1993-1-1, 6.3.1
ST2100.00	Stability Lateral torsional buckling acc. to EN 1993-1-1, 6.3.2
ST3100.00	Stability Bending and buckling about principal axes acc. to EN 1993-1-1, 6.3.3
SE0100.00	Serviceability Negligible deflections
SE1100.00	Serviceability Deflections in z-direction
SE1200.00	Serviceability Deflections in y-direction
WA5001.00	Warning Torsion is neglected for stability design checks

VKR & KKR – Vindsträvor

Design Check Type	Description
SP1100.00	Section Proof Tension acc. to EN 1993-1-1, 6.2.3
SP1200.00	Section Proof Compression acc. to EN 1993-1-1, 6.2.4
SP3100.02	Section Proof Shear in z-axis acc. to EN 1993-1-1, 6.2.6(2) Plastic design
SP3500.00	Section Proof Shear buckling acc. to EN 1993-1-5, 5.1, 5.2, 5.3 and 5.5
SP4100.03	Section Proof Bending about y-axis acc. to EN 1993-1-1, 6.2.5 Plastic design
SP6500.02	Section Proof Bending about y-axis, axial force and shear acc. to EN 1993-1-1, 6.2.9.1 and 6.2.10 Plastic design
ST1100.00	Stability Flexural buckling about principal y-axis acc. to EN 1993-1-1, 6.3.1
ST1300.00	Stability Flexural buckling about principal z-axis acc. to EN 1993-1-1, 6.3.1
ST2100.00	Stability Lateral torsional buckling acc. to EN 1993-1-1, 6.3.2
ST3100.00	Stability Bending and buckling about principal axes acc. to EN 1993-1-1, 6.3.3
SE0100.00	Serviceability Negligible deflections
SE1100.00	Serviceability Deflections in z-direction