

VRIJE UNIVERSITEIT BRUSSEL

Ontwerpmethodologie

4WE, 4TT, 4LR, 2.2LR, 2.2WB, 3.2LR

Prof. S. Vanlanduit



Vrije Universiteit Brussel

Voorwoord

Ongeveer 30% van het Bruto Nationaal Product is afkomstig van de verkoop van producten die de laatste vijf jaren werden ontwikkeld en 90% van de producten die we zullen gebruiken binnen de 10 jaar bestaan nog niet [18]. Naast dit rechtstreekse economisch draagvlak biedt het product ontwerp door innovatie nog tal van andere toegevoegde waarden voor het socio-economisch klimaat in Vlaanderen. Bovendien is productontwikkeling een bedrijvigheid die zich manifesteert over verschillende sectoren en op verschillende schalen. Gezien de productie van producten meer en meer naar lage loonlanden verschuift wordt het behoudt van de belangrijke product kennis die het ontwerp inhoud steeds belangrijker. Deze decentralisatie en de bijhorende herverdeling van de rol van de verschillende teams (ontwerp en productie) zorgt ervoor dat de organisatie en communicatie van het ontwerp een zeer belangrijke taak binnen het ontwikkelingsproces wordt. Door de hogere eisen van de consumenten naar kwalitatieve maar goedkopere producten wordt het ontwikkelingsproces alsmaar complexer. Daarnaast komt nog de groeiende concurrentie (o.a. van China, Japan, Oost-Europa) die bovenop de kwaliteits- en prijsdruk ook een steeds kortere ontwerptijd nodig maakt. Voorgaande argumenten tonen de noodzaak van een methodologische aanpak van het ontwerpproces van een product - zoals in deze cursus beschreven wordt - aan. In het bijzonder zullen volgende vraagstellingen rond het ontwerpproces aan bod komen:

- Wat zijn de verschillende stadia van het ontwerpproces, en welke tools zijn er voorhanden om de resultaten in de verschillende stadia in kaart te brengen?
- Hoe kunnen de ontwerp projecten op een efficiënte manier gepland worden (opvolging van de taken, deadlines, allocatie van resources)?
- Wanneer zijn de ontwerp inspanningen rendabel? Hoe bereken ik de kost van een product?

- Welke doelgerichte strategieën kunnen aangewend worden om bepaalde doelstellingen te realiseren (vb. milieuvriendelijk ontwerp)?
- Wat zijn de mogelijkheden en beperkingen van computer ondersteuning bij het ontwerpen.

Deel 1 in de cursus handelt over de organisatie van het ontwerpproces en is voornamelijk gebaseerd op de referentiewerken [16, 13] (de economische aspecten komen aan bod in [3]). De informatie rond Deel 2 - doelgericht ontwerp aspecten - is overgenomen uit [12]. Voor de materie met betrekking tot Deel 3 - computerondersteund ontwerpen werden de werken [5, 4] en de handleiding van het SolidWorks software pakket gebruikt.

Inhoudsopgave

Voorwoord	i
I Organisatie van het mechanische ontwerp proces	1
1 Algemeenheden	3
1.1 Inleiding	3
1.2 Variabelen in het ontwerpproces	4
1.3 Concurrent engineering	7
2 Het mechanisch ontwerpproces	15
2.1 Terminologie in het ontwerpproces	15
2.2 Het ontwerp team	17
2.2.1 Informatie verwerking	17
2.2.2 Teams	18
2.3 Het ontwerpproces	21
2.3.1 Overzicht	21
2.3.2 Project definitie en project planning	21
2.3.3 Specificatie in het ontwerpproces	25
2.3.4 Het conceptueel ontwerp	30
2.3.5 De product ontwikkelingsfase in het ontwerpproces	38
2.3.6 Lanceren en ondersteunen van het product	45
3 Project management	57
3.1 Definities	57
3.2 De projectomgeving en de project fasen	59
3.3 Project integratie management	61
3.4 Project draagwijdte management	62
3.5 Project tijdsmanagement	62
3.6 Project kost	65
3.7 Projectkwaliteit	66

3.8	Human resources	67
3.9	Project communicatie	68
3.10	Project risico management	70
3.11	Projectaanwinst management	71
3.12	Introductie Microsoft Project	71
4	Economische aspecten in het ontwerpproces	73
4.1	Economische beslissingen nemen	73
4.1.1	Actualiseren van kosten en inkomsten	73
4.1.2	Kost vergelijkingen	75
4.1.3	Afschrijvingen	78
4.1.4	Belastingen	79
4.1.5	Rentabiliteit	80
4.1.6	Andere rentabiliteitsfactoren	83
4.1.7	Break-even analyse	85
4.2	Kost	86
4.2.1	Kost categorieën	87
4.2.2	Kost schattingsmethodes	90
4.3	Value engineering - Value analyse (VE - VA)	93
II	Doelgerichte ontwerpaspecten	103
5	Design for use (DFU)	105
5.1	Inleiding	105
5.1.1	Wat is ergonomie?	105
5.1.2	Het belang van ergonomie	106
5.1.3	Het Interactie-model	106
5.1.4	Ergonomie en ontwerpen	108
5.2	Gebruikers	110
5.2.1	Soorten gebruikers	110
5.2.2	Het gebruik van percentielen	110
5.2.3	Mannikins	112
5.2.4	Menselijke kracht	112
5.2.5	De zintuigen	115
5.2.6	Horen	116
5.2.7	Zien	116
5.3	Interactie	117
5.3.1	Logische bediening	118
5.3.2	Compatibiliteit	118
5.3.3	Aangeleerde codes	120

5.3.4	Terugkoppeling	121
5.3.5	Redundantie	121
5.3.6	Intelligente producten	121
5.3.7	Grafische informatie	122
5.3.8	Gebruiksaanwijzing	124
5.3.9	Productveiligheid	126
5.3.10	Gebruiksonderzoek	126
5.4	Producten	127
5.4.1	Ruimten	127
5.4.2	Ondersteuning	128
5.4.3	Handgrepen	129
5.4.4	Gereedschap	130
5.4.5	Bedieningspanelen	131
6	Design for assembly (DFA)	137
6.1	Doel: kostenbesparing	137
6.2	De methode	137
6.2.1	Opdrachtfase	138
6.2.2	Informatiefase	138
6.2.3	Visualisatiefase	138
6.2.4	Analysefase	139
6.2.5	Provocatiefase	145
6.2.6	Creatieve fase	147
6.2.7	Evaluatiefase	147
6.2.8	Invoeringsfase	147
7	Design for manufacturability (DFM)	151
7.1	Inleiding fabricagegericht ontwerpen	151
7.2	Inleiding productietechniek	154
7.2.1	Indeling bewerkingsprocessen	154
7.2.2	Ontwerprichtlijnen voor de verschillende bewerkingspro- cessen	157
7.3	Richtlijnen voor fabricagegericht ontwerpen	160
7.3.1	Algemene DFM richtlijnen	160
7.3.2	Materiaalkeuze voor fabricagegericht ontwerpen	161
7.3.3	Checklist voor de ontwerpfase	162
7.3.4	Checklist voor de vervaardigingsfase	164
8	Materiaalkeuze in het ontwerp	167
8.1	Inleiding	167
8.2	Moment van keuze	168

8.3	Denk in het materiaal	168
8.4	Kennis van de materialen	169
8.5	Wettelijke bepalingen en eisen	171
8.6	Indeling in groepen en eigenschappen, de diagrammen van Ashby	172
9	Green design	189
9.1	Milieukundig ontwerpen	189
9.1.1	Het productontwikkelingsproces	189
9.1.2	Projectorganisatie	190
9.1.3	Strategiebepaling	192
9.1.4	Wet- en regelgeving	192
9.2	Methoden voor verbetering van het milieuprofiel	194
9.2.1	Ontwerpstrategieën	194
9.2.2	Materiaalkeuze	200
9.2.3	Energiebewust ontwerpen	203
9.2.4	Recycling	205
III	Computerondersteund ontwerpen	207
10	Computer Aided Design	209
10.1	Werking en opbouw van solid modelers	209
10.1.1	Conventioneel 2D tekenen	209
10.1.2	3D-CAD	210
10.1.3	Data opslag van solid models	216
10.1.4	Werking en opbouw van parametrische solid modelers .	219
10.1.5	'Bottom Up' en 'Top Down' modelleren	226
10.1.6	Wijzigen van parts	231
10.1.7	High end en mid range 3D-pakketten	233
10.1.8	Keuzecriteria	234
10.1.9	Solid modeling versus 2D-CAD	235
10.1.10	Communicatie van CAD gegevens	237
10.2	Product Data Management	243
10.2.1	Productgegevens en PDM	244
10.2.2	Productbeschrijving	244
10.2.3	Ontwerp-rationele gegevens	244
10.2.4	Bibliotheken	245
10.2.5	Beheer van productgegevens	246
10.2.6	Plaatsing van PDM binnen het bedrijf	247
10.2.7	Werking van het PDM-systeem	248

11 Ontwerp optimalisatie	251
11.1 De fundamenteën van structurele optimalisatie	251
11.2 Optimalisatie algoritmen	256
11.2.1 Problemen zonder constraints	257
11.2.2 Problemen met constraints	262
11.3 Optimalisatie strategieën	263
11.3.1 Multicriteria optimaliteit	264
11.3.2 Vorm optimalisatie	266
11.4 Voorbeeld van een optimaal ontwerp	269
11.4.1 Probleemstelling	269
11.4.2 Optimalisatie in MATLAB	272
11.4.3 Ontwerp optimalisatie in ANSYS	273
A Veertig Inventieve TRIZ principes [1]	281
B Lijst van veelgebruikte materialen	293
C Introductie MS Project	309
C.1 De fundamenteën	309
C.2 Invoeren en schedulen van de taken lijst	311
C.3 Toewijzen van resources en kosten	314
C.4 Bekijken van de project database	316
C.5 Updaten van projecten en bekijken van vooruitgang	316
D Introductie SolidWorks	321
D.1 Maken van een PART in SolidWorks	321
D.2 Maken van een ASSEMBLY in SolidWorks	324
D.3 Maken van tekening (DRAWING)	326

Lijst met afkortingen

ANOM: Analysis of means

ANOVA: Analysis of variance

AOA: Activity on Arrow

AON: Activity on Node

BE: Break-even

BCG: Boston Consultancy Group

BOM: Bill of Materials

CAD: Computer Aided Design

CPM: Critical Path Method

CRF: Capital Recovery Factor

DCF: Discounted Cash Flow

DFA: Design For Assembly

DFM: Design For Manufacturability

DFU: Design For Use

DIN: Deutsche Industrie Norm

DOE: Design Of Experiments

DSA: Design Sensitivity Analysis

ECO: Engineering Change Orders

EV: Earned Value

- FDM:** Fused Deposition Modelling
- FORM:** First Order Reliability Method
- GPS:** Geometrische productspecificatie
- IRR:** Internal Rate of Return
- LiDS:** Lifecycle Design Strategy
- LOM:** Laminated Object Manufacturing
- MPO:** Milieugerichte Productontwikkeling
- MPS:** Materiële productspecificatie
- PDF:** Probability density function
- PDM:** Product Data Management
- PERT:** Project Evaluation and Review Technique
- PO:** Productontwikkeling
- QFD:** Quality Function Deployment
- QLF:** Quality Loss Function
- RAM:** Responsibility Assignment Matrix
- RSM:** Response Surface Methodologie
- SFF:** Sink Fund Factor
- SLA:** Stereolithografie
- SLS:** Selective Laser Sintering
- SNR:** Signal-to-noise ratio
- SPS:** Structurele productspecificatie
- TM:** Transformation Method
- VA:** Value Analysis, Value Engineering
- WA:** Waarde Analyse
- WBS:** Work Breakdown Structure Template

Hoofdstuk 9

Green design

9.1 Milieukundig ontwerpen

In dit hoofdstuk komen de verschillen tussen Milieugerichte Productontwerpen (MPO) en de gebruikelijke productontwikkeling aan de orde, de afwijkende project-organisatie, het bedrijfsmilieubeleid en de wet- en regelgeving op het terrein van het milieu. Vervolgens zal veel aandacht worden besteed aan de methoden die ontwikkeld zijn voor het meten en beoordelen van de milieubelasting en hulpmiddelen die dit vergemakkelijken. Daarna wordt ingegaan op diverse methoden voor het verbeteren van de milieuconsequenties en het creëren van duurzame oplossingen.

9.1.1 Het productontwikkelingsproces

Ook voor de milieugerichte productontwikkeling is het ontwerpproces model dat in Hoofdstuk 2 besproken werd een getrouw beeld van het gebruikelijke ontwerpproces: de 'M' van 'MPO' verandert de structuur van het productontwikkelingsproces als zodanig niet. In de fase van bedenken, genereren en selecteren van alternatieve oplossingen voor het ontwerpprobleem moeten milieuvorwaarden in de afweging betrokken worden. Dit is echter niet eenvoudig, gezien het vaak niet mogelijk is een concrete en toetsbare milieudoelstelling te formuleren:

- Er is veel informatie nodig waar voorheen nooit om gevraagd werd, en die is bovendien moeilijk te verkrijgen (zoals de exacte hoeveelheid en soort materialen die toegepast zijn in een inkoopdeel).
- Als we de compositie dan kennen, dan weten we nog vaak niet waar de grondstoffen vandaan komen. Denk maar eens aan de vele vindplaatsen en fabrieken voor verwerking. Dit wordt verergerd door de

onbekende transport(om)wegen van veel hulp- en grondstoffen voordat ze verwerkt respectievelijk toegepast worden in een product. Wat zijn de milieu-ingrepen van deze materialen tegen de tijd dat uw product in productie gaat? Zal de voorraad koper inderdaad uitgeput raken? Is dit dramatisch of zullen er op tijd geschikte alternatieven worden ontwikkeld? Wordt de ozonlaag inderdaad steeds verder aangetast en komt dat door onze westerse levenswijze of toch voornamelijk door vulkaanuitbarstingen?

- Bij het beoordelen van het milieuprofiel van een product en de verbeterdoelstelling moeten we aannamen doen en ook waardeoordelen uitspreken. Vinden we het broeikas effect erger dan aantasting van de ozonlaag? Of verminderde biodiversiteit erger dan landschapaantasting? De situatie doet zich namelijk regelmatig voor dat een product ten opzichte van een bepaald milieuprobleem verbetert, maar daardoor tegelijkertijd aan een ander milieuprobleem bijdraagt. Over deze waardeoordelen moet consensus ontstaan, wat niet een taak is voor een individuele ontwerper, maar voor mondiale overheden, gesteund door diverse onderzoeksinstituten.
- De ontwerper wordt bij MPO gedwongen vanuit een veel breder perspectief dan voorheen de eisen voor zijn product op te stellen en te realiseren. Ten eerste moet hij, als hij werkelijk streeft naar integraal ketenbeheer, gaan overleggen met partijen in de keten waar hij vroeger niet aan behoefde te denken: de instanties die het product gaan terugnemen, herverwerken en hergebruiken. Ten tweede moet hij gaan nadenken over de rechten van toekomstige generaties en van mensen aan de andere zijde van de wereld. Zijn horizon wordt in zowel tijds- als geografische zin enorm verbreed.

9.1.2 Projectorganisatie

MPO blijft niet geïsoleerd tot de ontwerpafdeling, tot een min of meer extra activiteit van een groep of persoon. Het vraagt een bijdrage van alle afdelingen binnen een bedrijf, van management, marketing en inkoop. Van productie, logistiek en onderhoud. Elk lid van de groep heeft een bepaalde rol te vervullen.

- De taak van de ontwerper is het uitvoeren van het ontwikkelingsproces en het rekening houden met de milieuaspecten waar het team belang aan hecht. Hij is degene die met creatieve constructieve oplossingen komt voor productverbetering.

- De directeur is vooral degene die individuele leden overstijgende beslissingen moet nemen. Denk hierbij aan imago, concurrentie, werkgelegenheid, financiering en dergelijke. Daarnaast zijn er soms bepaald tegenstrijdige doelstellingen te realiseren waar een directeur stelling in kan nemen. Wat bijvoorbeeld te doen als een milieukundige verbetering door verminderd verbruik van energie een hogere kostprijs betekent. Of een duurzamere uitvoering langedureffecten heeft op de bedrijfseconomie.
- Een inkoper kent de grondstoffenmarkt en is op de hoogte van trends en verwachte ontwikkelingen. Inkoopers komen in contact met leveranciers en vernemen daarvan welke milieumaatregelen er genomen zijn en de consequenties voor hun producten. Hij kent daarmee de mogelijkheden en beperkingen van milieubewuste inkoop. Doelbewuste sturing van de inkoop kan reeds tot drastische milieuverbeteringen leiden.
- De tegenhanger van de inkoper is de verkoper of marketeer. Deze is expert op het gebied van afnemerwensen. Enerzijds resulteert dit in aanwijzingen voor productaanpassing, anderzijds vertegenwoordigt hij de consument voor het taxeren van marktkansen. Milieuverbeteringen kunnen marktkansen vergroten mits deze benut en gepresenteerd worden. Anderzijds kan een onjuiste milieuclaim het bedrijf grote schade berokkenen. Koketteren met niet juiste of niet waar te maken verdiensten is af te raden.
- Een milieukundige is de persoon die voor de beoordeling van de milieuaspecten verantwoordelijk is. Vaak zal deze ingehuurd worden omdat het nog geen algemeen specialisme betreft. Belangrijk is dan dat de persoon affiniteit heeft met de productgroep en over aantoonbare ervaring beschikt in de industriële productontwikkeling. Hij moet in staat zijn met een positieve instelling de mate en diepgang van zijn adviezen af te stemmen op de behoefte van het projectteam en de fase van uitvoering van het pilotproject. Hoewel het onderwerp vaak een emotionele lading heeft moet hij een objectieve voorstelling van zaken kunnen geven en tot een neutrale afweging in staat zijn. Zo niet, dan botst dit snel met de rationeel, economisch ingestelde bedrijfscultuur en is het pilotproject tot mislukken gedoemd. Een realistische instelling, vandaar die noodzakelijke ervaring, is een absolute noodzaak.

9.1.3 Strategiebepaling

Het is duidelijk dat we, vóór we in een project milieueisen mee gaan nemen, goed stil moeten staan bij de keuze waar de MPO zich op zal concentreren. MPO wordt in de praktijk nog vaak vertaald als 'Design for Disassembly / Recycling', wat neerkomt op een voor dat doel geschikte materiaalkeuze en een slimme demontabele constructie van het product. Natuurlijk is het een stap in de goede richting en in sommige gevallen is het ook vanuit het milieu gezien de beste strategie. Maar soms zijn andere richtingen met een preventief karakter beter, zoals minimaal energiegebruik of het simpelweg verlagen van het materiaalgebruik. Omdat bedrijven geen onbeperkte tijd hebben voor MPO en omdat de milieueisen onderling soms strijdig zijn, ontstaan dilemma's en moeten er keuzen ten aanzien van de MPO-prioriteiten gemaakt worden. Welke set MPO-strategieën zal worden toegepast moet de onderneming helder maken.

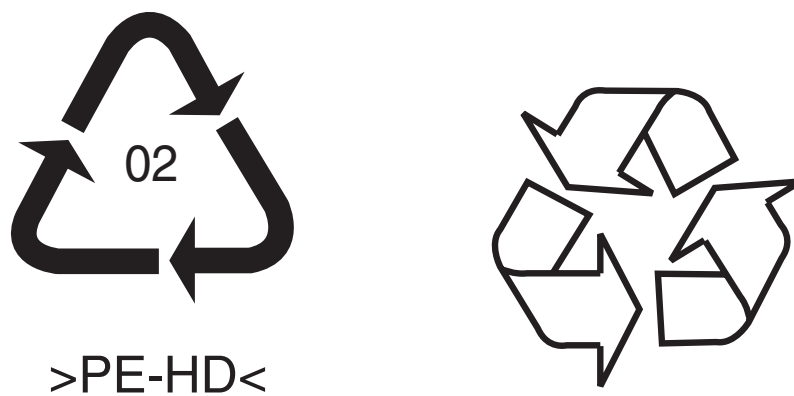
9.1.4 Wet- en regelgeving

Op Europees niveau bestaat een keurmerk met als doel voor bepaalde producten aan het publiek duidelijk te maken dat deze producten aan hoge milieueisen voldoen. Is een bedrijf eenmaal in het bezit van dit Ecolabel, zie Figuur 9.1, dan hoeft het niet meer in elk land apart een milieukeur aan te vragen.

Behalve deze keurmerken zijn er enkele tekens in gebruik die verwarring wekken doordat ze associatie oproepen met milieukeurmerken of verkeerd geïnterpreteerd worden. Voorbeelden hiervan zijn de tekens in Figuur 9.2. Het eerste teken is bedoeld om kunststoffen zodanig te merken zodat het sorteren na afdanking van het product gemakkelijk gaat. Het bestaat uit een driehoekig recyclingsteken met daarin een nummer. Dat nummer typeert de gebruikte kunststof. Onder het teken kan een nadere verduidelijking staan. Dit laatste is genormeerd in DIN-norm 54840 en ISO 11469. Het tweede teken is bedoeld als aanmoediging om de verpakking waar het teken opstaat na gebruik in de bijbehorende recyclingbak te deponeren.



Figuur 9.1: Ecolabel van de Europese unie.

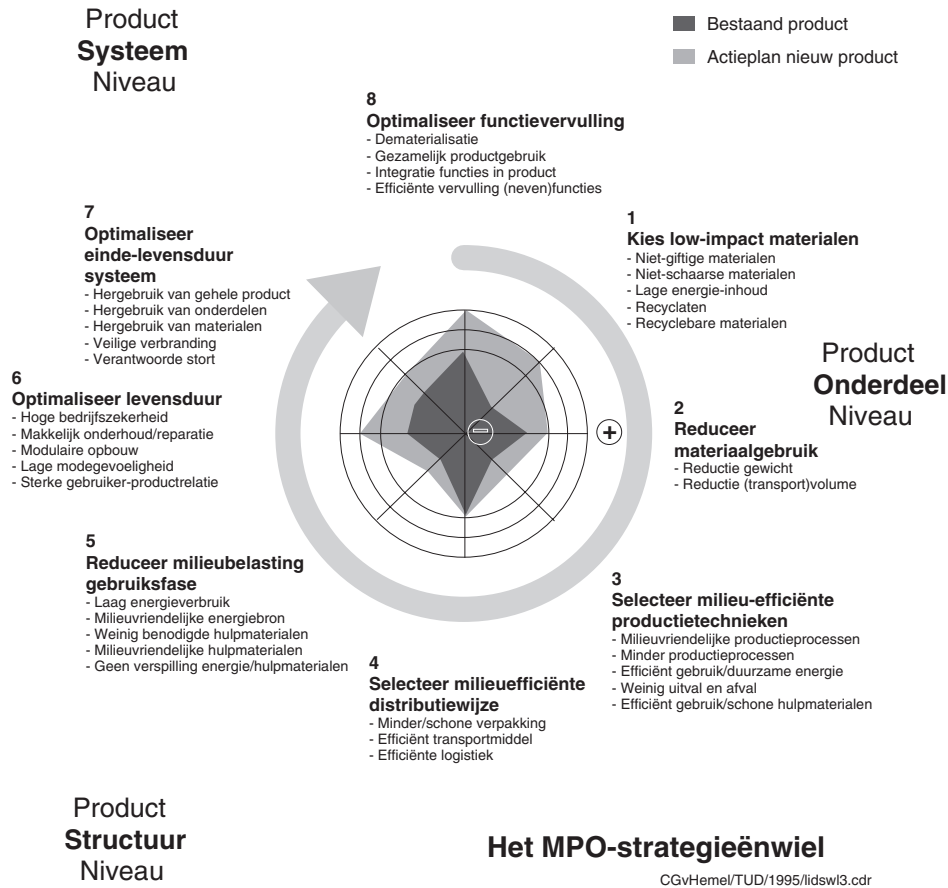


Figuur 9.2: Links: herkenningstekens voor kunststoffen, rechts: recyclingstekens voor verpakkingen.

9.2 Methoden voor verbetering van het milieuprofiel

9.2.1 Ontwerpstrategieën

Alle vuistregels ('MPO-principes') zijn te clusteren in acht 'MPO-strategieën', (Eng.: *Lifecycle Design Strategies*) zoals weergegeven in Figuur 9.3.



Figuur 9.3: Classificatie van de lifecycle design strategie (het zogenaamde LiDS wiel).

De clustering van de vuistregels in acht MPO-strategieën is zodanig dat de clusters elk een 'beslispunt' in het productontwikkelingsproces vormen. Daarbij is gestreefd naar zo min mogelijk overlapping. De strategieën zijn uitgedrukt als 'besluitvormingsacties' die leiden tot een 'positief resultaat'.

Dus niet 'Materiaalkeuze' maar 'Selecteer materialen met een lage milieubelasting'.

De clusters zijn gerangschikt volgens verschillende principes:

- De meest abstracte en verstrekkende strategie 'Optimaliseer functie-
vervulling' heeft ter onderstreping van zijn afwijkende karakter het
nummer '8' gekregen.
- Kloksgewijs kunnen we in strategie 1 tot en met 7 de levensloop van het
product herkennen, van materiaalwinning en -verwerking, productie,
gebruik tot het zogeheten eindelevensduurtraject.
- Tegen de klok in kunnen we ruwweg de volgorde van het produc-
tontwikkelingsproces onderscheiden, met name van de opeenvolgende
beslispunten.

Het LiDS-wiel kan voor verschillende doeleinden en, zoals in Figuur 9.3 te zien is, op verschillende momenten in het ontwerpproces gebruikt worden:

1. Als denkmodel bij de MPO-strategiebepaling. In de praktijk wordt MPO nog vaak eenzijdig ingevoerd. Om dit te voorkomen is het nuttig het LiDS-wiel als denkmodel te gebruiken waarin het geheel aan MPO-strategieën op een logische manier is weergegeven;
2. Als presentatiemiddel, ter visualisatie van het huidige, het gewenste en het gerealiseerde MPO-profiel van een product. In het wiel kan een productmanager aangeven op welke strategieën hij zich wel of niet wil focussen, op kortere of langere termijn. Op deze manier ontstaat een actieplan dat ervoor zorgt dat alle leden van het productontwikkelingsteam weten welke zaken aangepakt gaan worden. Aan het eind van het project kan het resultaat ter evaluatie weer naar deze MPO-doelstelling teruggekoppeld worden;
3. Als creativiteitstechniek. Het wiel is waardevol voor ideegeneratie, om op systematische wijze milieukundige verbeteropties te bedenken.

Welke zijn de MPO-strategieën?

De strategieën 1 tot en met 7 kunnen gezien worden als 'verbeterrichtingen', ter realisatie op korte of middellange termijn (evolutie). De verdergaande verbeteringen, meestal pas op lange termijn te realiseren, worden gevat onder het kopje 'Optimaliseer functievervulling', MPO-strategie 8. Men moet niet lukraak een aantal van deze principes gaan toepassen! Om te

voorkomen dat men zich richt op een verkeerde richting met als resultaat verbeteropties die zowel qua milieuwinst als aansluiting bij bedrijfsdoelstellingen slecht scoren, is een weloverwogen MPO-prioriteitstelling onontbeerlijk.

LiDS 1: Kies materialen met lage milieubelasting ('Selection low-impact materials') Principes:

- Kies niet-giftige materialen ('Non-hazardous materials').
- Vermijd schaarse materialen ('Non-exhaustable materials').
- Kies materialen met een lage energie-inhoud ('Low energy content').
- Kies waar mogelijk een recyclaat ('Recycled materials').
- Kies recyclebare materialen ('Recyclable materials').

Milieudoel: Te vermijden materialen zijn onder andere zware metalen, asbest en CFKs. Andere moeten beperkt worden omdat ze schaars zijn of niet voldoende snel aangevuld worden. Voorbeelden zijn aardolie, koper, tin, zink en platina. Het is zinnig om materialen indien mogelijk opnieuw te gebruiken. Sommige materialen hebben een grotere 'energie-intensiteit' dan andere, wat wil zeggen dat de winning en productie van het materiaal erg energie-intensief is. Deze materialen zijn alleen gerechtvaardigd als hun bijzondere eigenschappen ook werkelijk in het product benut worden. Zo is aluminium, dat toch een hoge energie-intensiteit heeft bij primaire winning, geschikt voor een product dat veel vervoerd wordt, omdat het relatief licht is. Bovendien is het goed recycleerbaar.

LiDS 2: Reduceer materiaalgebruik ('Reduction Material') Streef naar zo min mogelijk materiaal door niet te overdimensioneren en slim te construeren. Streef daarbij naar een klein volume, zodat het product tijdens transport minder ruimte vergt. Principes:

- Reduceer het gewicht ('Reduction of weight').
- Reduceer het (transport)volume ('Reduction of (transport) volume').

Milieudoel: Door minder materiaal te gebruiken werkt men echt aan preventie. Er is simpelweg minder materiaal nodig. Verder wordt de milieubelasting tijdens transport verlaagd doordat het product lichter en/of kleiner is.

LiDS 3: Selecteer milieuefficiënte productietechnieken ('Optimization production techniques') Bij het kiezen van productietechnieken moeten we streven naar technieken met een lage milieubelasting, dat wil zeggen een laag en niet schadelijk hulpstoffenverbruik, weinig verliezen aan energie en grondstoffen en weinig afval. Principes:

9.2. METHODEN VOOR VERBETERING VAN HET MILIEUPROFIEL¹⁹⁷

- Kies milieuvriendelijke productieprocessen (Alternative production processes).
- Minder productieprocessen (Few production processes).
- Efficiënt gebruik van liefst duurzame energie (Low/clean energy consumption).
- Weinig af- en uitval bij productie (Little production waste).
- Efficiënt gebruik van liefst schone hulpmaterialen (Few/clean production consumables).

Milieudoel: Selecteren we productietechnieken met een lage milieubelasting, dan streven we naar een laag energieverbruik, naar weinig benodigde hulp-materialen die bovendien een niet-schadelijke samenstelling hebben. Opmerkingen: Deze strategie vertegenwoordigt het streven naar 'schoner produceren', waar bedrijven vaak al bekend mee zijn als onderdeel van interne milieuzorg. Echter, bij deze strategie moeten men zich niet beperken tot het eigen bedrijf, maar ook stimuleren dat overige bedrijven in de productketen hun processen verbeteren (ketenbeheer). Zo zijn er bedrijven die van hun toeleveranciers eisen dat zij beschikken over een bedrijfsintern milieuzorgsysteem.

LiDS 4: Gebruik milieuefficiënte distributie ('Selection efficient distribution system') Zorg dat het product op een efficiënte wijze vervoerd wordt van fabriek naar detaillist en gebruiker. Dit betreft zowel de verpakking, als transportvorm en transportorganisatie (logistiek). Principes:

- Gebruik minder en liefst schone verpakkingen ('Little/clean packaging').
- Kies het juiste transportmiddel ('Efficiënt transportmode').
- Optimaliseer logistiek ('Efficiënt logistics').

Milieudoel: Hier is sprake van preventie. Hoe minder verpakking nodig is, hoe meer materiaal en transportgewicht (dus transportenergie) wordt bespaard. De milieubelasting door transport met het vliegtuig is veel groter dan met de boot, wat de keuze van transportmiddel moet beïnvloeden. Verder reduceert optimale belading van het transportmiddel en efficiënte distributielogistiek ook milieubelasting.

LiDS 5: Reduceer milieubelasting in de gebruiksfase ('Reduction users-impact'). In de gebruiksfase zijn hulpstoffen (energie, water, wasmiddel, koffie) en ook producten (batterijen, navulcassettes, filters) nodig om het

product te laten functioneren. Ditzelfde geldt voor het onderhoud en de reparatie van het product. Ontwerp het product dusdanig dat de gebruiker deze zaken niet verspilt en/of zoek hiervoor milieuefficiëntere alternatieven.

Principes:

- Reduceer energieverbruik product ('Low energy consumption').
- Kies duurzame energiebron ('Clean energy source').
- Reduceer benodigde hoeveelheid hulpmaterialen ('Few consumables').
- Kies milieuvriendelijke hulpmaterialen ('Clean consumables').
- Voorkom verspilling energie en hulpmaterialen ('Spill prevention by design').

Milieudoel: Het verlagen van energieverbruik door directe energieconsumptie en door minder materiaal te gebruiken door efficiëntere energieomzetting te realiseren. Verder kan het product zo ontworpen worden dat de gebruiker aangemoedigd wordt tot efficiënt productgebruik, waardoor verspilling wordt verminderd. Opmerkingen: Door het uitvoeren van milieuanalyses is gebleken dat als een product energie verbruikt in de gebruiksfase, dit vaak de fase is waar de grootste milieubelasting ontstaat. Wordt in een project een bepaald hulpproduct of hulpmateriaal verbeterd, dan moeten deze weer worden gezien als afzonderlijke producten, met elk een eigen levensketen (systeembenadering). Ook daarvoor kan de LiDS-benadering gebruikt worden voor het verbeteren van elke keten apart.

LiDS 6: Optimaliseer de levensduur/eerste gebruiksduur ('Optimization initial lifetime') Verleng de technische levensduur (de tijd dat het product goed functioneert) en de esthetische levensduur (de tijd dat de gebruiker het product mooi vindt), ofwel de eerste gebruiksduur van het product, zodat het product langer in zijn oorspronkelijke functie gebruikt zal worden. Principes:

- Kies het materiaal in overeenstemming met de gebruiksduur.
- Verhoog de bedrijfszekerheid ('Reliability and durability').
- Vergemakkelijk onderhoud en reparatie ('Easy maintenance and repair').
- Zorg voor modulariteit ('Modular product structure').
- Verlaag de modegevoeligheid ('Classic design').
- Intensiveer de gebruiker-productrelatie ('Intensify caring for product').

Milieudoel: Al deze principes streven het doel na de milieubelasting te minimaliseren door de producten langer te gebruiken en minder onnodig te af te danken. Door te kiezen voor repareerbaarheid, modulariteit of aanpasbaarheid kan een product dat in technisch of esthetisch opzicht niet meer voldoet, worden 'opgewerkt' en zodoende nog steeds voldoen aan de (gewijzigde) behoefte van de gebruiker.

Opmerking:

Soms is het beter de levensduur niet te verlengen, maar juist te verkorten; als de technische levensduur veel langer is dan de esthetische, dan moet een nieuwe balans gevonden worden. De technische levensduur moet dan korter en/of de esthetische langer, waarbij de laatste optie natuurlijk de voorkeur krijgt. Een kortere levensduur is te verkiezen als er nieuwe, energiezuinigere alternatieven worden ontwikkeld.

LiDS 7: Optimaliseer het einde-levensduur systeem ('Optimization end-of-life system') Verleng de levenscyclus van het product door aan de eerste levensduur nieuwe cycli toe te voegen, door hergebruik van het product, van onderdelen en/of van materialen. Kan van kringloopsluiting geen sprake zijn, garandeer dan veilige verbranding. Principes:

- Stimuleer hergebruik van het gehele product ('Reuse').
- Stimuleer hergebruik van onderdelen ('Remanufacturing of refurbishing').
- Stimuleer hergebruik van de toegepaste materialen ('Recycling').
- Stimuleer veilige verbranding ('Clean incineration').

Milieudoel: Het sluiten van kringlopen, door hergebruik van het product, onderdelen of materialen beoogt de milieubelasting te verlagen doordat de reeds geïnvesteerde materialen en energie die nodig waren voor het produceren van het product nogmaals kunnen worden ingezet. Zo wordt de milieubelasting in de vorm van materiaalbeslag, energiegebruik en schadelijke emissies voor het nieuwe product verlaagd. Hoe meer het product in zijn originele vorm behouden blijft, hoe meer milieuwinst dit oplevert, mits er een milieuefficiënt terugneme- en herverwerkingsysteem voor bestaat of wordt ontwikkeld. Opmerkingen: Voor onderdelengebruik en recycling geldt:

- Maak een product goed demonteerbaar.
- Pas zo min mogelijk verschillende materiaalsoorten toe.
- Pas voor onderdelen die niet scheidbaar zijn 'compatibele' materialen toe, dat wil zeggen, van één materiaalfamilie, zodat het recycleat van gelijke kwaliteit blijft.

In de praktijk is 'Recycling' een erg populaire strategie, met name omdat de vereisten daartoe relatief weinig tijd en investeringen vergen: maak het product demontabel en gebruik de juiste (combinatie) materialen.

LiDS 8: Optimaliseer functievervulling ('New concept development') Met deze strategie worden nieuwe wegen gezocht voor het vervullen van behoeften. Men richt zich niet op verbetering van het huidige, materiële product, maar zoekt verder naar de achterliggende functie en behoeftevervulling van het product. Vandaaruit tracht men alternatieven te ontwikkelen voor het huidige product of productgebruik. Principes:

- dematerialisatie (Dematerialization);
- gezamenlijk productgebruik (Shared use of the product);
- integratie van verschillende functies tot één product (Integration of functions);
- efficiënte vervulling (neven)functies (Functional optimization of product(components)).

Milieudoelen: Deze verbeterrichting duidt op het verlagen van de milieubelasting door andere wegen in te slaan. Vervangen van een materieel product door een immateriële behoeftenvervuller, bijvoorbeeld een dienst. Ook wordt er van uitgegaan dat als mensen gezamenlijk gebruik maken van een product zonder het zelf te bezitten, er op een efficiëntere wijze omgegaan wordt met het product. Als verschillende functies of producten in één product geïntegreerd kunnen worden, bespaart dat veel materiaal en ruimtebeslag. Een voorbeeld hiervan is de toepassing van geheugenmetaal dat gelijktijdig sensor en actuatorfuncties kan combineren, waardoor aanzienlijke componentenreductie optreedt.

9.2.2 Materiaalkeuze

In een vroeg stadium van ontwikkeling is de detaillering per definitie nog te vaag om een milieuanalyse te maken. Op welke gronden kun je dan een geschikt materiaal kiezen? De keuze wordt dan vaak gemaakt worden op grond van ervaring en aannames. Dat hoeft niet fout te zijn, maar het leidt ook gemakkelijk tot verkeerde conclusies; en wel om twee redenen. Ten eerste moet de vergelijking op de juiste functionele eenheid betrokken zijn. Dat is niet altijd de milieukosten per kg, maar milieukosten voor een bepaalde functievervulling over zoveel jaar. Door de tijd er in te betrekken wordt gezorgd dat de duurzaamheid van de oplossing meegenomen wordt. Voor een goed ontwerp geldt dat het materiaal optimaal wordt gebruikt. Voor bekleding

bijvoorbeeld is dan de minimale milieubelasting per hoeveelheid volume het keuzecriterium. Voor trekbelaste delen en buigbelaste delen daarentegen is de Eco-indicator per sterkte respectievelijk stijfheid het keuzecriterium. Het werken met kengetallen ligt dus voor de hand. Daarbij moet bedacht worden dat rekening moet worden gehouden met de integrale milieubezwaren. Waar het om gaat is het principe van de keten-gedachte. De volledige levenscyclus moet in de beoordeling worden betrokken. Desnoods moet daarom een schatting gemaakt worden van de bewerkingswijze, de gebruiks- (onderhouds)aspecten en de afdanking. De Eco-indicator is daar een geschikte hulpmiddel voor, want van een groot aantal materialen, processen en afdankingsmogelijkheden zijn er Eco-indicators. Voor elk materiaal is hiermee de hypothetische levenscyclus Eco-indicator te construeren die voor de materiaalkeuze in het vroege ontwerpstadium gebruikt kan worden. Ter illustratie is in Figuur 9.4 voor een aantal materialen kengetallen getoond voor een product dat wordt afgedankt met het huishoudelijke afval en verbrand zal worden met 100% warmtebenutting. De Eco-indicator voor de hypothetische levenscyclus is dan de som van Eco-indicator (productie + bewerking + afdanking). De afleiding van de kengetallen is als volgt. Stel we belasten een staaf van een materiaal met de eigenschappen ρ , σ en Eco , met lengte L axiaal met een kracht F tot de maximaal toelaatbare spanning σ is bereikt. De doorsnede A die dan nodig is wordt:

1. Schrijf de vergelijking op voor de spanning in de op trek belaste constructie: $A = \frac{F}{\sigma}$
2. Bepaal de uitdrukking voor het gewicht van die staaf: $G = A\rho L$.
3. Bepaal de totale Eco-indicator voor die staaf: $Q = G \times Eco = A \times \rho \times L \times Eco$.
4. Combineer stap 1 met stap 3: $\sigma = \frac{f \times \rho \times L \times Eco}{Q}$.
5. Breng in deze vergelijking de materiaaleigenschappen aan één kant: $\frac{\rho \times Eco}{\sigma} = \frac{Q}{FL}$.

Voor verplaatsingsbelasting gaat de procedure op dezelfde manier. De resulterende kentallen (dus voor eendimensionale spanningstoestanden) zijn:

$$\frac{\rho \times Eco}{\sigma} = cte \quad \text{en} \quad \frac{\rho \times Eco}{E} = cte \quad (9.1)$$

Voor een sterke, stijve, en tevens energiebewuste oplossing moet dus gezocht worden naar een materiaal waarvoor deze kentallen de laagste waarden hebben. Om het zoeken hiernaar te vergemakkelijken zijn de kentallen

in tabellen op te nemen. Iedereen kan dit voor een eigen set materialen met eigen verwerkingsgegevens en afdankscenarios opstellen ter ondersteuning van de materiaalkeuze. Voor buiging van staven en balken (zoals klikvingers) zijn op overeenkomstige wijze kentallen af te leiden. Die kentallen luiden (Figuur 9.4):

$$\frac{\rho \times Eco}{\sigma^{2/3}} = cte \quad \text{en} \quad \frac{\rho \times Eco}{E^{1/2}} = cte \quad (9.2)$$

<i>Materiaal</i>	1	2	3	4	5	6	7	8	9
	<i>Indicator productie</i>	<i>Indicator Bewerking</i>	<i>Indicator afdanking (Incin.)</i>	<i>Indicator Totaal</i>	<i>Dichtheid ρ</i>	<i>Sterkte σ</i>	<i>Stijfheid E</i>	<i>Trek</i>	<i>Buiging</i>
	<i>Eco/kg</i>	<i>Eco/kg</i>	<i>Eco/kg</i>	<i>Eco/kg</i>	<i>kg/dm³</i>	<i>N/mm²</i>	<i>N/mm²</i>	<i>100× $\rho Eco/\sigma$</i>	<i>100× $\rho Eco/\sqrt{\sigma}$</i>
LDPE	3,8	0,5	1,8	6	0,92	8	400	70	28
HDPE	2,9	0,5	1,8	5	0,96	35	1000	14	16
PP	3,3	0,5	1,8	6	0,90	30	1200	17	15
PUR	8,7	0,3	1,8	11	1,25	30	100	45	135
PPE	5,8	0,5	1,8	10	1,10	55	2300	16	19
ABS	9,3	0,5	1,8	13,5	1,06	40	2000	31	28
PVC	4,2	1,1	6,9	14	1,38	50	3000	34	31
PA	13,0	0,5	1,8	157,5	1,14	80	1400	22	47
PET	7,1	0,5	1,8	12,5	1,35	60	3000	21	23
PC	13,0	1,1	1,8	17,5	1,20	60	2400	32	39
PA (30% glas)	9,73	0,5	1,8	111,5	1,55	140	6000	13	24
PC (30% glas)	9,73	0,5	1,8	14,5	1,59	90	6000	21	25
Staal	4,1	0,5	1,8	7,5	7,80	360	210000	14	11
RVS	17,0	0,5	1,8	28	7,70	500	200000	30	33
Aluminium	18,0	2,0	1,8	~28	2,70	180	70000	33	22
Koper	60,0	0,4	1,8	~37	8,90	220	120000	252	160
Hout	0,7	0,5	0,6	2,2	0,7	90	11000	1	1

Figuur 9.4: Eco-indicator kengetallen voor volume, sterkte en buigstijfheid voor enkele constructiematerialen. Bron: Rapport De Eco-indicator 95, Pré Consultants, Amersfoort 1995.

Voor buiging van plaatvormige delen zijn de kentallen iets anders. Bij andere belastingen, bijvoorbeeld bij wringing en bij gecombineerde belastingen zijn de kentallen wéér anders. Het verschil met de andere kentallen is de waarde van de exponent in de noemer van de kentallen. Onveranderlijk blijft echter gelden dat de waarde van de kentallen klein zo mogelijk moet zijn. Met andere woorden, als de materialen gerangschikt worden naar waarde

van de kentallen voor trekbelasting dan blijft de rangschikking gelijk. In de tabel is voor enkele veel toegepaste materialen de kentallen voor sterkte, volume en stijfheid weergegeven. Soortgelijke kengetallen zijn ook af te leiden voor andere toepassingen, bijvoorbeeld voor warmte-isolatie of thermal shock bestendigheid. De procedure is nu duidelijk en biedt daarbij hulp. Echter, een materiaal zal als regel aan meer dan een bepaalde eis moeten voldoen en daarom is dit hulpmiddel, dat slechts naar één eis optimaliseert, slechts een steun bij het kiezen.

9.2.3 Energiebewust ontwerpen

Voor elk van de in een product energievragende processen kan getracht worden het energiegebruik te verminderen. Dit leidt tot een groot aantal vuistregels waarvan enkele hieronder gegeven worden.

Ontstaan

- Probeer zo min mogelijk materiaal toe te passen.
- Probeer materiaal toe te passen met een lage energie-inhoud.
- Probeer gerecycled materiaal toe te passen.
- Kies voor zo min mogelijk onderdelen en bewerkingen.
- Kies voor energiezuinige bewerkingen.
- Probeer afval en uitval tot een minimum te beperken.
- Beperk verspanende bewerkingen.

Distributie

- Streef naar een minimaal productgewicht en -volume.
- Streef naar een minimaal verpakkingsgewicht en -volume.
- Pas zo min mogelijk verpakkingsmateriaal toe.
- Pas materiaal toe met een lage energie-inhoud.
- Probeer gerecycled materiaal toe te passen.
- Probeer meermalige verpakkingen toe te passen; deze zijn bijna altijd energiezuiniger, met name als het aantal gebruikskerens groot en de transportafstanden klein zijn.

- Streef naar gewichtsreductie van de vervoermiddelen.
- Probeer de transportafstanden te beperken.
- Probeer producten en (meermalige) verpakkingen nestbaar en stapelbaar te maken.

Gebruik

- Probeer producten te ontwerpen die door menskracht worden aangedreven.
- Probeer producten te ontwerpen die gebruik maken van duurzame energiebronnen.
- Vermijd het toepassen van een hoogwaardige energiebron (elektriciteit) voor een laagwaardige toepassing (de opwekking van warmte).
- Probeer energiegebruikende producten zo zuinig mogelijk te ontwerpen.
- Probeer koelende en verwarmde apparaten goed te isoleren.
- Ontwerp producten met een zo laag mogelijk vermogen en een zo hoog mogelijk rendement.
- Ontwerp onderhoudsarme producten.
- Ontwerp producten die goed te repareren zijn.
- Ontwerp producten die geen (of minder) hulpmaterialen nodig hebben (bijvoorbeeld meermalig filter voor koffiezetapparaten).
- Probeer de levensduur van een product te optimaliseren; hierbij valt te denken aan:
 - duurzame materialen en duurzame constructies;
 - tijdloze vormgeving;
 - aan te passen vormgeving;
 - goed vervangbare onderdelen;
 - alle onderdelen dezelfde levensduur.
- Probeer transport voor service en reparatie te vermijden of te verminderen.

Afdanken

- Probeer meermalig te gebruiken producten en onderdelen te ontwerpen.
- Ontwerp goed demonteerbare producten.
- Ontwerp producten met zo min mogelijk verschillende materialen.
- Probeer goed recyclebare materialen toe te passen.
- Voorkom dat producten gestort moeten worden (de geïnvesteerde energie gaat geheel verloren).
- Vermijd energie-intensieve demontageprocessen.
- Probeer laminaten, laklagen, composieten en andere toevoegingen die recyclen lastig of onmogelijk maken te vermijden.
- Probeer transport in de afdankfase te verminderen.

9.2.4 Recycling

Recycling is de populaire naam voor het wedergebruik van eerder gebruikt materiaal. In feite is recycling een van de mogelijkheden bij afdanking en valt daarmee onder de noemer end-of-life. Zijn er algemene vuistregels bij end-of-life optimalisatie? Ja, deze luiden:

1. Streef naar hergebruik van het hele product voor dezelfde of een nieuwe toepassing van het product (Reuse).
2. Streef naar hergebruik van subassemblies en componenten (Remanufacturing/ refurbishing).
3. Tracht materiaalrecycling in een hoogwaardiger toepassing te realiseren (upgrading).
4. Streef naar materiaalrecycling in de oorspronkelijke toepassing (Primary recycling).
5. Streef naar materiaalrecycling in een laagwaardiger toepassing (Secondary recycling).
6. Streef naar materiaalrecycling van kunststoffen tot grondstoffen, door middel van thermisch of chemische ontleding. Hierbij worden lange kunststofmoleculen afgebroken tot elementaire grondstoffen, die weer gebruikt worden in raffinaderijen of bij de productie van petrochemicaliën (Tertiary recycling, ook wel 'Feedstock recycling' genoemd, door middel van pyrolyse, hydrogenation, vergassing, chemolyse).

7. Verbrand niet herbruikbare materialen met energieopwekking en goede rookgasreiniging (Quaternary recycling, soms ook 'Thermische recycling' genoemd).
8. Verbrand niet-herbruikbare materialen zonder energieopwekking met rookgasreiniging.
9. Stort de restmaterialen op gecontroleerde wijze als vast afval.

Deel III

**Computerondersteund
ontwerpen**

Hoofdstuk 10

Computer Aided Design

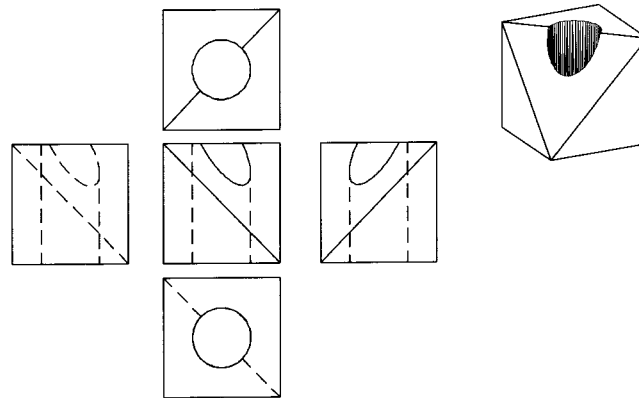
In dit hoofdstuk komen enkele algemeenheden met betrekking tot Computer Aided Design aan bod. Specifieke informatie in verband met het praktisch gebruik van het CAD pakket SolidWorks is gegeven in Appendix D waar een beknopte 'tutorial' terug te vinden is.

10.1 Werking en opbouw van solid modelers

10.1.1 Conventioneel 2D tekenen

In de beginfase van het ontwerpproces vindt bij de ontwerper de eerste beeldvorming van een product in ruimtelijke beelden plaats. De ontwerper maakt eventueel een 3D-schets van het product. Daarna legt de tekenaar het product in 2D-werktekeningen (mono's en samenstellingen) op papier of op een beeldscherm vast. Door het tweedimensionaal vastleggen van ruimtelijke voorwerpen vindt op twee plaatsen een conversie plaats. Allereerst moet de ruimtelijke voorstelling die de ontwerper in gedachten heeft worden vertaald naar een aantal 2D-aanzichten en doorsneden. Als het ontwerp is uitgewerkt moeten derden weer een extra inspanning doen om uit de doorsneden en aanzichten een ruimtelijk beeld van het ontwerp te vormen. Door een product tweedimensionaal vast te leggen, wordt dus tweemaal een vertaalslag gevraagd. Dit is niet alleen een mogelijke bron van informatieverlies of van menselijke fouten, maar vereist ook opleiding en training in het lezen en interpreteren van 2D tekeningen. Figuur 10.1 toont een voorbeeld van een eenvoudig onderdeel, waaruit blijkt dat het interpreteren van een product vanuit alleen voor-, zij-, boven- en onder-aanzichten lastig kan zijn. Tevens wordt opgemerkt dat in 2D de aanzichten 'los' van elkaar worden getekend waardoor snel fouten worden gemaakt, vooral bij wijzigingen. Veel bedrijven, met

name in het midden- en kleinbedrijf, benutten de CAD-software alleen voor het aanmaken van 2D-tekeningen. Als CAD alleen wordt gebruikt voor het aanmaken van conventionele 2D-tekeningen, dan is het terecht dat er wordt gestreefd naar tekeningen die zo goed mogelijk lijken op wat voorheen met potlood en gom werd aangemaakt. Vaak wordt de mogelijkheid om CAD-data rechtstreeks (elektronisch) door te sluizen naar andere pakketten, bijvoorbeeld een CAM-pakket of een logistiek pakket, niet benut. De gebruiker, die het CAD-pakket alleen gebruikt voor het aanmaken van 2D-tekeningen, is al een heel eind als hij weet dat het commando LINE de vroegere liniaal vervangt, ARC de passer en ERASE het gometje. In de discussie over de voor- en nadelen van 2D-CAD worden als voordelen genoemd: het sneller kunnen wijzigen, de handige manipuleerfuncties, zoals COPY, MOVE en MIRROR, het benutten van layers en de kwalitatief goed ogende tekeningen. Het verlies aan overzicht bij het maken van de tekeningen wordt als een nadeel gezien. Bij het overgaan naar 3D-CAD is deze discussie niet meer relevant, omdat de gebruiker de toch nog gewenste 2D-tekeningen niet meer zelf aanmaakt, maar deze rechtstreeks onttrekt aan het 3D-model.



Figuur 10.1: Object aangemaakt met Solid Modelling.

10.1.2 3D-CAD

Bij 3D-CAD wordt onderscheid gemaakt in:

- Wireframe models (draadmodellen).
- Surface models (oppervlakte modellen).
- Solid models (massieve modellen).

Aanvankelijk konden met de 3D-CAD-software-pakketten alleen wireframe en surface models worden aangemaakt. Op dit moment hebben de meeste pakketten ook de mogelijkheid van Solid Modeling

Wireframe model

In een 3D-wireframe model worden van de contouren van een 3D-object de lijnen en curven getekend. Het verschil met een 2D-tekening is dat van de ingevoerde punten niet alleen de x- en y-, maar ook de z-coördinaten worden opgeslagen. Nadelen van Wireframe models zijn:

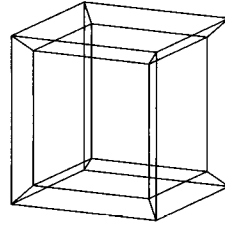
- Er is geen hidden line removal mogelijk omdat er geen vlakken bekend zijn, dat wil zeggen dat niet kan worden aangegeven wat nietzichtbare lijnen zijn en wat wel.
- Het model is niet altijd eenduidig (wordt met het draadmodel van Figuur 10.2 één van de vormen van Figuur 10.3 bedoeld of misschien nog iets anders?).
- Complexe constructie-onderdelen zijn moeilijk te interpreteren.

Surface model

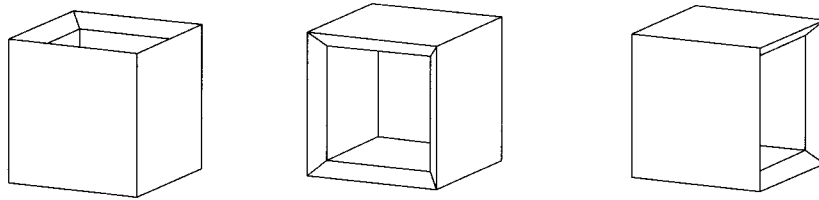
In een surface model wordt het model opgebouwd met diverse typen oppervlakken. Vrijwel alle CAD-pakketten hebben commando's voor het aanmaken van:

- Tabulated surfaces: een oppervlak dat wordt verkregen door een curve (in Figuur 10.4a de polyline A) langs een rechte lijn 'omhoog' te trekken.
- Revolved surfaces: een oppervlak dat ontstaat door een curve (in Figuur 10.4b de polyline B) te wentelen om een lijn.
- Ruled surfaces: een oppervlak dat gevormd wordt door twee curven (de polylines C en D in Figuur 10.4c) te verbinden met rechte lijnen.
- Coons surfaces: een oppervlak dat wordt verkregen door vier met elkaar verbonden curven (de polylines E, F, G en H, Figuur 10.4d); het oppervlak wordt gevormd door zogenaamde Coons patches.

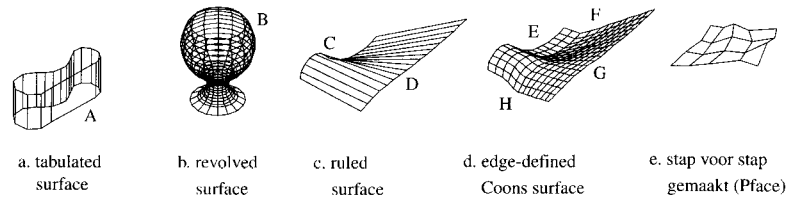
Naast deze gedefinieerde surfaces bestaan er commando's om willekeurige surfaces op te bouwen, zie Figuur 10.4.



Figuur 10.2: Draadmodel.



Figuur 10.3: Solid modellen.

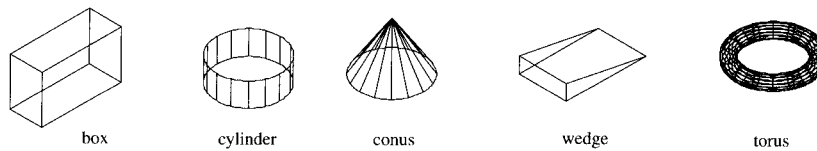


Figuur 10.4: Surface modellen.

Solid model

Aan de hand van de Solid Model-faciliteiten, opgenomen in AutoCAD wordt de basis van Solid Modeling behandeld. Daarna komen de professionelere Solid Modelers aan de orde. Een reden om de basis te behandelen met de begrippen en commando's van AutoCAD is dat er vanuit wordt gegaan dat vele lezers AutoCAD 2D-ervaring hebben. Bij het aanmaken van een Solid Model beginnen we met het plaatsen van zogenaamde primitieven of basic shapes. Dit kunnen meegeleverde basic shapes zijn, zoals BOX, CYLINDER, CONUS, WEDGE of TORUS (Figuur 10.5) of zelf aangemaakte basic shapes met EXTRUDE (Figuur 10.6) of REVOLVE (Figuur 10.7). Men plaatst deze

basisvormen en realiseert vervolgens het gewenste model met Boolean algebra bewerkingen zoals optellen en aftrekken.



Figuur 10.5: Basic shapes.

Bij het commando `EXTRUDE` wordt het doorsnedeprofiel getekend (in Figuur 10.6 de cirkel in de linker tekening). Daarna zijn er twee mogelijkheden om de doorsnede te extruderen: a. Een rechte lijn aangeven, waarlangs het profiel wordt 'getrokken'; er kan ook nog een zogenaamde taper angle worden opgegeven, zodat een conische vorm ontstaat. b. Een pad (path) tekenen (in de linker tekening van 10.6 de omhooggaande en vervolgens afbuigende polyline), waarlangs de doorsnede wordt geextrudeerd; deze wijze van extruderen wordt ook wel sweep genoemd.

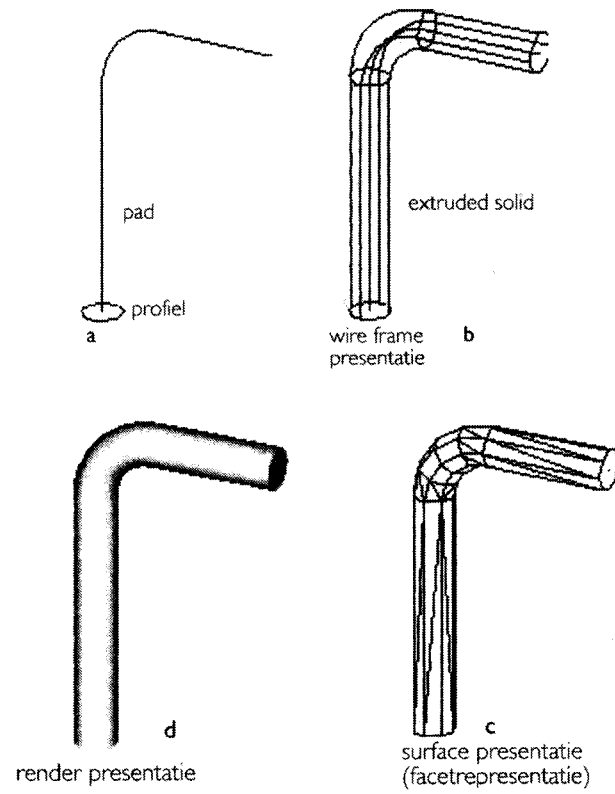
Een Solid Model kan op het beeldscherm worden gepresenteerd als wireframe (zie Figuur 10.6b), als surface met hidden line removal (zie Figuur 10.6c) en gerenderd (Figuur 10.6d).

Bij het commando `REVOLVE` wordt ook het doorsnedeprofiel getekend en deze wordt vervolgens gewenteld om een (hart)lijn over een op te geven hoek, zie Figuur 10.7.

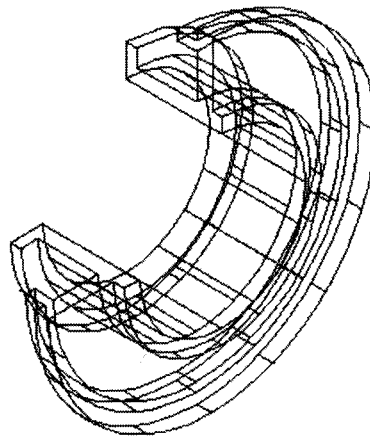
Na het aanmaken van de basic shapes kan met Boolean operaties, zoals `UNION` en `SUBTRACT`, uit de basic shapes het Solid Model worden gegenereerd, zie Figuren 10.8 en 10.9.

Met `FILLET` en `CHAMFER` kunnen afrondingen en afkantingen worden aangebracht. In Figuur 10.10a is op twee rand en van een box `FILLET` toegepast. In Figuur 10.10b op twee randen `CHAMFER`. In Figuur 10.10c is op de bovenste doorsnijding van het met `UNION` verkregen onderdeel `FILLET` toegepast. Zo'n filleting op de doorsnijdingskromme van gekromde oppervlakken wordt ook wel een blending genoemd.

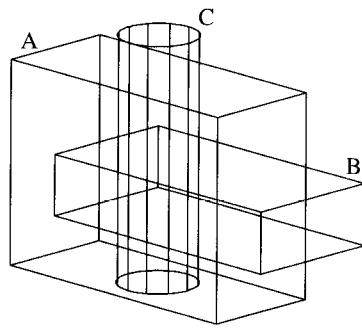
Figuur 10.11 toont nog een met AutoCAD aangemaakt onderdeel. Voor het nog beter visualiseren van het onderdeel is in Figuur 10.12 een kwart uit het onderdeel weggesneden (`SUBTRACT` van een `BOX`, waarvan een zijde samenvalt met de hartlijn). In Figuur 10.13 is aangegeven hoe het onderdeel is gecreëerd. Met het commando `MASSPROP` kan van het onderdeel



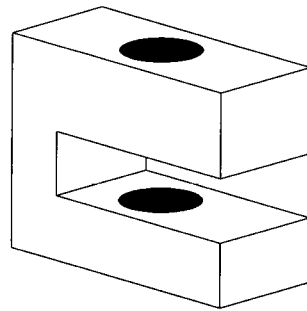
Figuur 10.6: EXTRUDE via een pad (sweep).



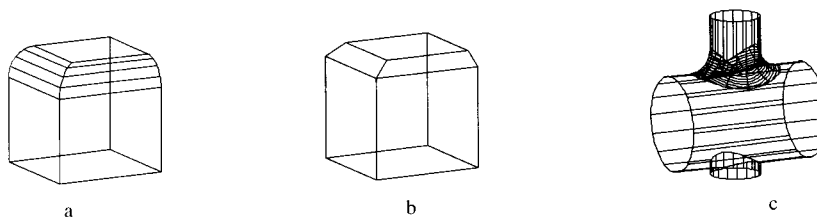
Figuur 10.7: REVOLVE, draadmodelrepresentatie.



Figuur 10.8: Drie primitieve plaatsen.

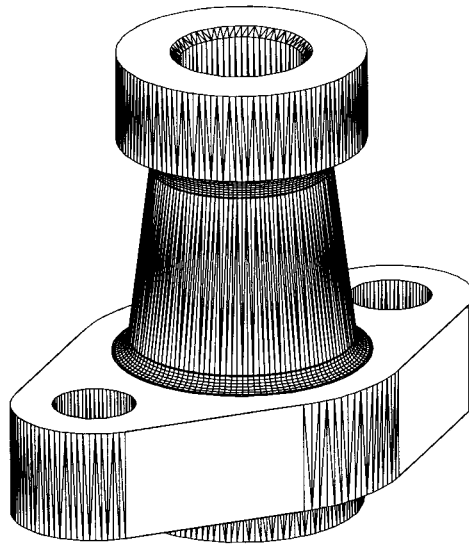


Figuur 10.9: SUBTRACT A-B-C.

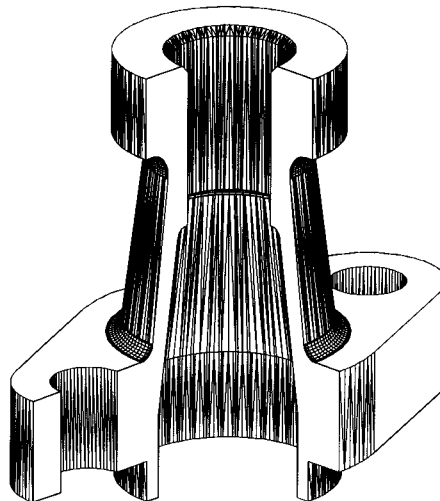


Figuur 10.10: FILLET en CHAMFER.

de massa, het oppervlak, het zwaartepunt en de massatraagheidsmomenten worden berekend.



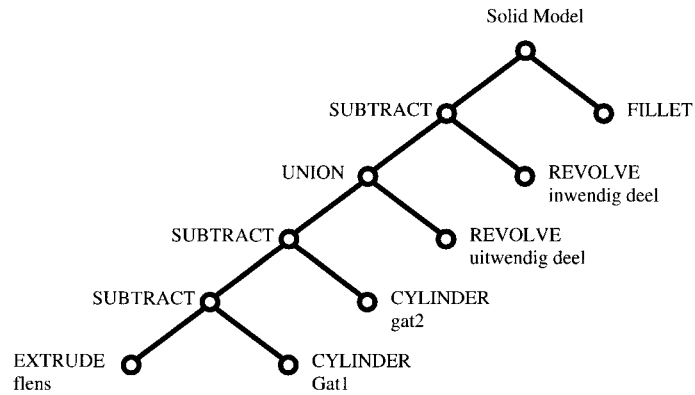
Figuur 10.11: Een met binaire operaties verkregen representatie.



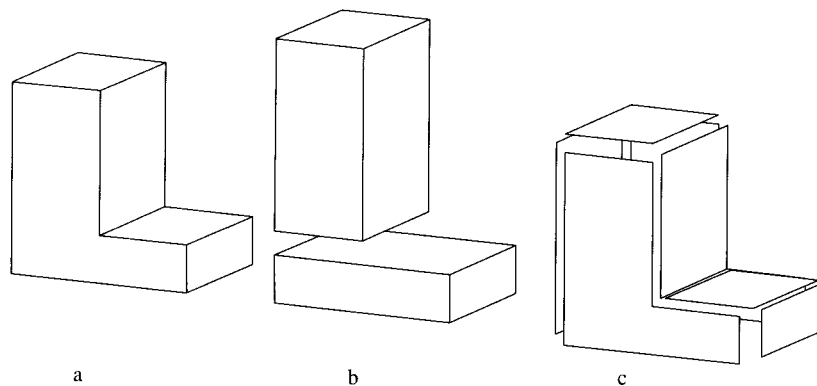
Figuur 10.12: Opengesneden onderdeel.

10.1.3 Data opslag van solid models

Tot nu toe is alleen ingegaan op hoe Solid Models worden opgebouwd. Er zal nu iets nader worden ingegaan op hoe de gegenereerde CAD-data wordt



Figuur 10.13: Constructive Solid Geometry model van het onderdeel in Figuur 10.11.



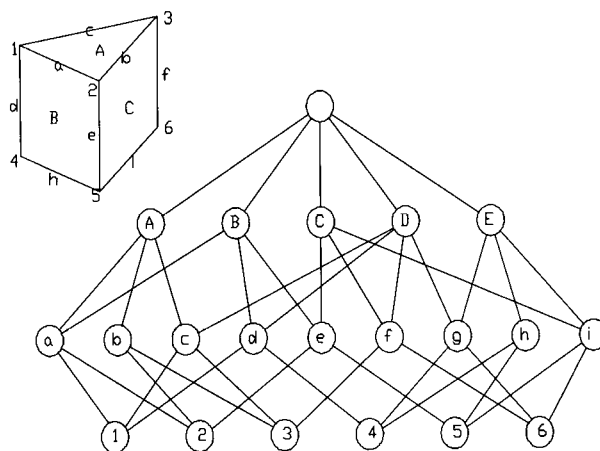
Figuur 10.14: .

opgeslagen. De wijze van opslaan van een Solid Model is voor nogal wat zaken van belang, zoals bijvoorbeeld: het wijzigen van het Solid Model (CSG of B-rep, zie hierna); de snelheid van de beeldopbouw; het onttrekken van informatie voor andere softwarepakketten, zoals een EEM-, een CAM-, een logistiek- of een stereolithografie-pakket.

De meest gebruikte datamodellen voor het opslaan van de geometrie en de topologie van een Solid Model zijn:

- Constructive Solid Geometry (CSG)
- Boundary representation (B-rep)

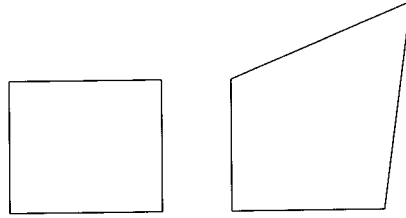
Bij CSG worden de basic shapes (de primitieven) en de bewerkingen erop (Boolean operaties) opgeslagen in een zogenaamde CSG-tree (zie Figuur 10.13 en 10.14). Om een Solid Model opgeslagen in een CSG-tree op het scherm af te beelden is het noodzakelijk om de vlakken (faces), de randen (edges) en de punten (vertices) van de eindvorm te bepalen vanuit de CSG-tree. Daarvoor is een stuk software nodig, de zogenaamde boundary evaluator. De hele ontstaansgeschiedenis wordt bij het oproepen van de geometrie steeds opnieuw doorlopen. Bij B-rep worden na elke (Booleaanse) bewerking de begrenzingen van het model berekend en opgeslagen (Figuur 10.14 en 10.15). De ontstaansgeschiedenis gaat bij B-rep verloren, alleen het eindresultaat wordt vastgelegd. CSG wordt wel de 'unevaluated representation' en B-rep de 'evaluated representation' genoemd.



Figuur 10.15: Dataopslag van een B-rep.

In AutoCAD wordt B-rep toegepast. Figuur 10.14 illustreert dat bij B-rep alleen de beschrijving van de boundaries van een solid wordt opgeslagen. Op het met UNION uit twee boxen aangemaakte Solid Model is het commando EXPLODE toegepast. Door de B-rep-dataopslag leidt EXPLODE niet tot de oorspronkelijke twee boxen, Figuur 10.14b, zoals wellicht verwacht, maar tot afzonderlijke vlakken van de begrenzing van de solid (te controleren met LIST). Om dit te illustreren zijn al deze vlakken met MOVE iets op zij gezet. In het geval van B-rep-dataopslag zijn alle primitieven, waaruit het model is opgebouwd, verloren.

Voor een CAD-datamodel is het onderscheid tussen topologie en geometrie (zie Figuur 10.16) van belang. De topologie van een object legt de relaties tussen de verschillende elementen van het object vast, bijvoorbeeld



Figuur 10.16: Topologie hetzelfde geometrie verschillend.

wat met wat is verbonden. Van het object in Figuur 10.16 is de topologie: vier rechte lijnen en hoe deze met elkaar zijn verbonden (de coördinaten van de hoekpunten of de lengte van de lijnstukken zijn nog niet vastgelegd). In de geometrische beschrijving worden getalwaarden vastgelegd, bijvoorbeeld van het object van Figuur 10.16 de coördinaten van de vier hoekpunten.

10.1.4 Werking en opbouw van parametrische solid modelers

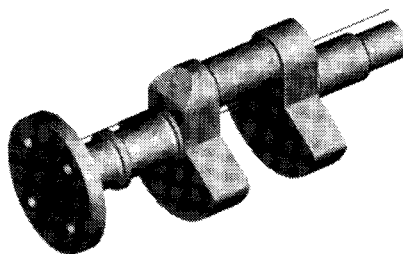
Alle huidige Solid Modelers onderscheiden drie delen:

1. Parts: onderdelen van een ontwerp als Solid Model aanmaken.
2. Assemblies: samenstellingen maken door geometrische relaties tussen de parts aan te brengen.
3. Drawings: 2D-tekeningen maken van de parts (aanzichten en doorsneden van de parts automatisch genereren, inclusief bemating) en 2D-tekeningen maken van de samenstelling (inclusief softwarematige ondersteuning bij het plaatsen van posnummers en stuklijsten).

Op deze drie delen gaan we nader in. De toelichting daarbij is voornamelijk gebaseerd op de ervaring opgedaan met Mechanical Desktop, SolidWorks en Solid Edge.

Parts

In vrijwel alle professionele Solid Modelers wordt gesproken over het opbouwen van een part met features (vormelementen). Als eerste wordt de base feature aangemaakt en vandaar uit kan de ontwerper met het invoeren van nieuwe features materiaal toevoegen (met bijvoorbeeld het commando EXTRUDE JOIN) of weghalen (EXTRUDE CUT). Er wordt onderscheid



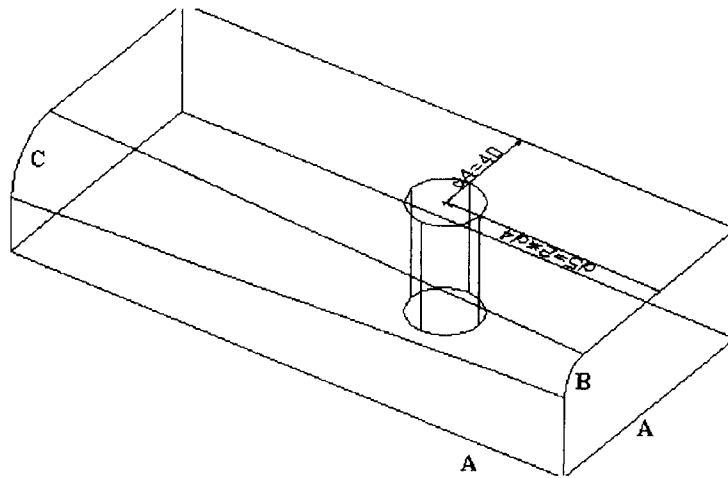
Figuur 10.17: Een solid model.

gemaakt in profiled features, waarbij het feature ontstaat door een aangeemaakt profile (een 2D-contour) te extruderen of te roteren, en applied features, zoals holes, fillets en chamfers. De functionaliteit van het part-deel van de diverse Solid Modelers verschilt niet veel en groeit ook steeds meer naar elkaar toe. Als bijvoorbeeld pakket X 'fillets met variabele radius' heeft (in Figuur 10.18 van B naar C), dan zal pakket Y in een volgende release ook zoiets hebben opgenomen. De gebruikte terminologie kan wel sterk verschillen. De meeste pakketten spreken wel over features en profiles, maar de hulpvlakken, die benut worden voor het aanbrengen van een profile verschillen vaak van naam.

Zo noemt Mechanical Desktop zo'n vlak work plane, SolidWorks construction plane, Solid Edge reference plane en Unigraphics en Catia datum plane.

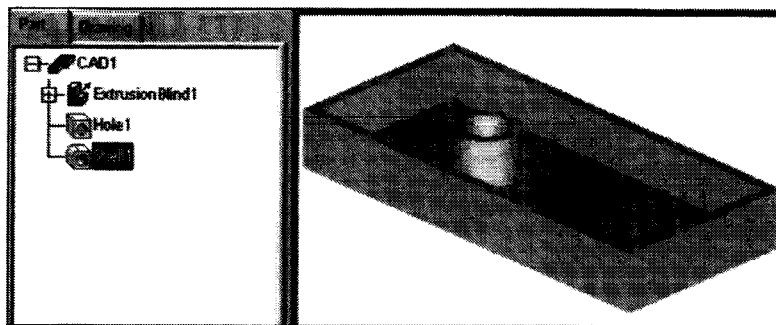
Een verschil tussen de Solid Model-faciliteiten van AutoCAD en de professionele Solid Modelers, zoals Mechanical Desktop, ligt in het wijzigen van een Solid Model. Stel dat men de plaats van het gat in Figuur 10.18 wilt wijzigen. Omdat in AutoCAD, vanwege het B-rep datamodel, alleen de begrenzingen worden opgeslagen en er dus niets bekend is over de ontstaansgeschiedenis, moet het gat eerst worden gedicht en vervolgens op de gewenste plaats een nieuwe CYLINDER met SUBTRACT worden aangebracht.

In de professionele solid model pakketten kan met een commando als EDIT FEATURE wel een maat worden aangeklikt en gewijzigd. Het gat kan hiermee op een simpele wijze worden verplaatst. Na zo'n wijziging moet de boundary representation van het Solid Model opnieuw worden berekend en dit vergt bij modellen met veel features nogal wat rekentijd. De wijzigingen kunnen worden aangebracht via de dimensions waarmee de feature is vastgelegd, de zogenaamde parametric dimensions. Iedere parametrische dimensie krijgt in de dataopslag een unieke naam, in Mechanical Desktop dO, dl, d2, et cetera. Tussen de parametric dimensions kunnen relaties wor-



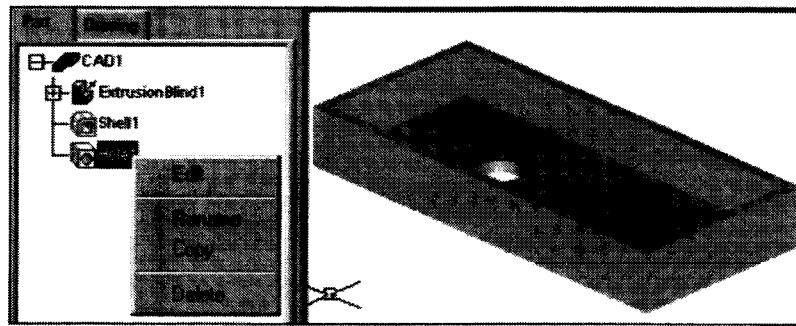
Figuur 10.18: Een gat verplaatsen.

den gelegd. In Figuur 10.18 is bijvoorbeeld d_5 gerelateerd aan d_4 middels $d_5=2*d_4$. Afmetingen parametrisch kunnen vastleggen is essentieel in de huidige Solid Modelers.



Figuur 10.19: Feature manager.

De meeste Solid Modelers kunnen de feature-opbouw op het scherm tonen (linker kolom in Figuur 10.19), een zogenaamde feature manager. Dit lijkt op een CSG-tree. Naast het overzicht dat de feature manager geeft, kan het ook handig worden benut voor het aanbrengen van wijzigingen. Het Solid Model van Figuur 10.19 is aangemaakt met Catia V5. Uit de feature manager, in Mechanical Desktop de Browser genoemd, blijkt dat dit als volgt is gebeurd: de base feature (het rechthoekige blok, in de Browser Extrusion Blind1),



Figuur 10.20: De volgorde van het aanmaken van de shell en hole omgedraaid.

daarna een HOLE en tot slot met het commando SHELL (in andere pakketten ook wel THIN WALL genoemd) het uiteindelijke Solid Model, waar, bij de uitvoering van het SHELL-commando, is aangegeven dat de bovenzijde opengemaakt moest worden. Het wijzigen van de volgorde (eerst de SHELL en daarna de HOLE) is mogelijk door in de feature manager de SHELL aan te klikken en met slepen boven de HOLE te plaatsen (zie de feature manager van Figuur 10.20). Nadat het model opnieuw is doorgerekend ontstaat het Solid Model van Figuur 10.20. Een doordenkertje.

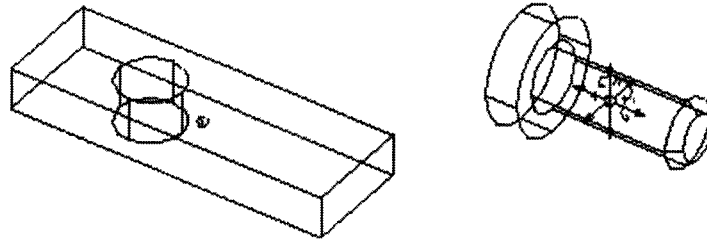
Figuur 10.20 toont het pop-up-menu dat wordt verkregen als met de rechter muisknop een feature wordt aangeklikt. Alle Solid Modelers hebben zoiets, en het is een handig menu voor wijzigen.

Assemblies

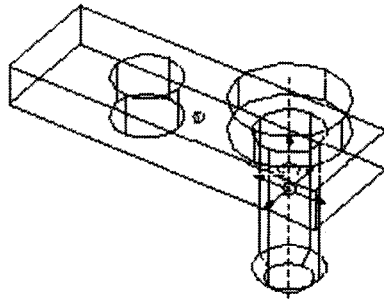
Bij assembleren moeten de nog erg ens willekeurig op het beeldscherm staande parts op de gewenste wijze ten opzichte van elkaar worden geplaatst.

In Figuur 10.21 zijn de parts een strip en een bout. De bout moet in het gat van de strip worden geplaatst. Iedere part heeft zes vrijheidsgraden (Degrees Of Freedom, DOF's), namelijk drie translaties en drie rotaties. Assembleren houdt in het reduceren van het aantal DOF's van iedere part.

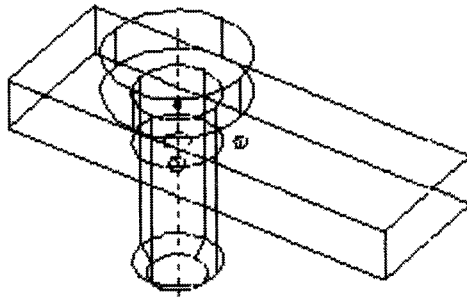
De strip is als eerste geplaatst in de samenstellingstekening. Vanuit het eerst geplaatste part wordt de samenstelling opgebouwd en daarom heeft de strip geen DOF's meer. De bout heeft nog zes DOF's. In Figuur 10.21 wordt met een DOF-symbool het aantal DOF's van een part getoond. Het DOF symbool staat in het zwaartepunt van een part. De pijlen geven de drie translaties aan, de drie cirkelbogen om de translaterichtingen de drie rotaties.



Figuur 10.21: Twee parts geplaatst.



Figuur 10.22: Mate plane.



Figuur 10.23: Mate line.

Om de bout in het gat te plaatsen moeten assembly constraints worden aangebracht via commando's als MATE PLANE en MATE LINE. Met MATE PLANE laten we de onderkant van de boutkop en het bovenzvlak van de strip samenvallen. De bout heeft nu nog drie DOF's: twee translaties in

het bovenvlak en de rotatie om de hartlijn, zie in Figuur 10.22 de twee translatiepijlen en de enige nog overgebleven rotatieboog. Daarna is met MATE LINE, waarbij de hartlijn van het gat en van de bout is geselecteerd, Figuur 10.23 aangemaakt. Er is nu nog een DOF over, namelijk rotatie van de bout om zijn hartlijn en dat mag.

Interference check

Alle Solid Modelers hebben in het assembly-deel de mogelijkheid van interference checking, dat is controleren of er overlap van parts (samenvallend materiaal) optreedt.



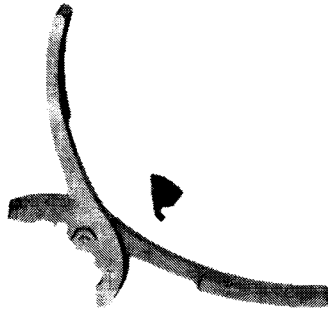
Figuur 10.24: Assembly van een tang.

Figuur 10.24 toont een met Mechanical Desktop geassembleerde tang. Om te bepalen wat de maximale openingshoek van de tang is, dus om te bepalen bij welke openingshoek interference optreedt, kan met een commando als MATE ANGLE de hoek tussen de twee tangdelen worden gevarieerd en kan bij iedere hoek worden nagegaan of er interference optreedt. Bij de openings hoek van Figuur 10.25 treedt interference op. Mechanical Desktop maakt van het samenvallend materiaal een part. In Figuur 10.25 is dit part met MOVE buiten de tang geplaatst.

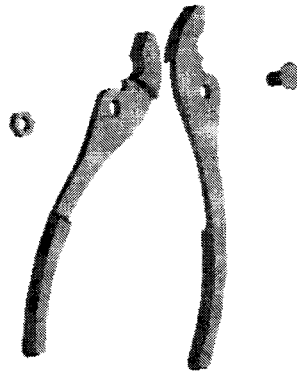
Exploded view Alle Solid Modelers hebben in het assembly-deel ook de mogelijkheid van het maken van een exploded view. Van de tang is een exploded view getoond in Figuur 10.26.

Drawings

Alle Solid Modelers hebben een drawing-deel waarmee van een part een 2D-monotekening en van een assembly een 2D-samenstellingstekening kan worden onttrokken. Bij het aanmaken van een 2D-monotekening hoeft de gebruiker alleen aan te geven welke aanzichten en doorsneden hij wil en waar ze geplaatst moeten worden. Het CAD-pakket zorgt voor de uitlijning en de bemating. De maten die door het CAD-pakket in de tekening worden geplaatst zijn de maten waarmee de gebruiker het Solid Model heeft gedimensioneerd, de parametric dimensions, ook wel de driving dimensions genoemd.



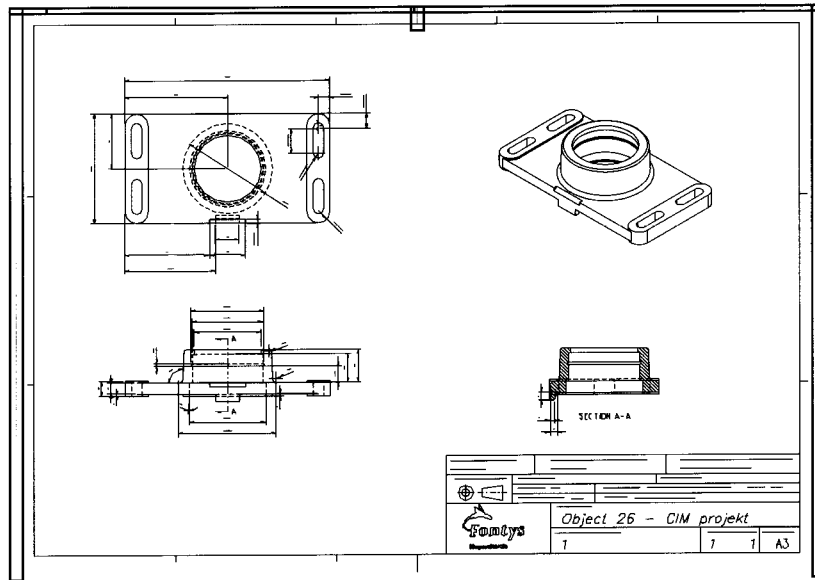
Figuur 10.25: Interference.



Figuur 10.26: Exploded view.

Om een goed ogende 2D-tekening te krijgen zal de gebruiker de be mating nog wel iets opschonen, bijvoorbeeld iets verplaatsen. Ook kan de gebruiker nog maten toevoegen. Dit worden reference dimensions of driven dimensions genoemd. In feite zijn deze maten niet nodig, omdat bij een volle dig geparameteriseerd model het model eenduidig vastligt. Het kan echter zijn dat de ontwerper bij het bemaaten niet aan de werkvoorbereider heeft gedacht en derhalve de tekening heeft bemaat op een wijze die voor de werkvoorbereiding niet is gewenst. Ook zullen vaak voor een 2Dproductie-tekening nog zaken als plaats- en vormtoleranties, lassymbolen, passingen en dergelijke moeten worden toegevoegd.

Ook 2D-samenstellingstekeningen worden vrijwel geheel automatisch ontrokken aan de assembly. Voor zaken als het plaatsen van posnummers (balloons) en het maken van stuklijsten (Bill Of Materials, BOM's) geeft de



Figuur 10.27: 2D monotekeningen onttrokken aan een solid model.

software veel ondersteuning.

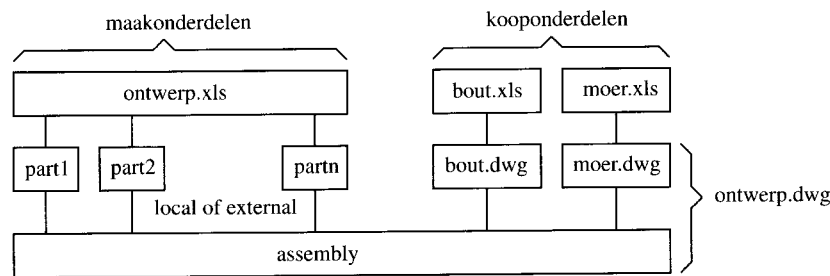
10.1.5 'Bottom Up' en 'Top Down' modelleren

Een ontwerp bestaat uit diverse parts. In zo'n ontwerp zijn referentie- of hoofdmaten te onderscheiden. Dit zijn de maten waarmee de parts aan elkaar zijn gerelateerd; de kernmaten. De vraag is hoe in Solid Modelers wordt omgegaan met samenhangende afmetingen van de diverse parts in een ontwerp. Deze problematiek is nauw verwant met Bottom Up en Top Down modelleren. Bij Bottom Up wordt begonnen met alle parts afzonderlijk te modelleren. Bij Top Down worden vanuit een samenstelling, met alleen nog maar de hoofdmaten, de parts gemodelleerd. Alle Solid Modelers hebben uiteraard de mogelijkheid om via Bottom Up een ontwerp op te zetten. De vraag is wat de Top Down faciliteiten zijn, oftewel hoe er met kernmaten wordt omgegaan.

Kernmaten opnemen in een spreadsheet

Vrijwel alle CAD-pakketten hebben de mogelijkheid om een part te linken (koppelen) aan een spreadsheet-programma, in een Windows-omgeving doorgaans EXCEL. Door te beginnen met het vaststellen wat de kernmaten van

een ontwerp zijn en deze in een EXCEL-file op te nemen kan een ontwerp gestructureerd worden opgezet.



Figuur 10.28: Een gestructureerde opzet van een ontwerp.

De werkwijze, geïllustreerd in Figuur 10.28, is:

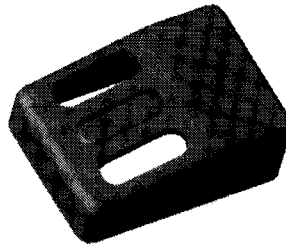
1. Plaats in een EXCEL-werkblad (in Figuur 10.28 ontwerp.xls) kernmaten van het gehele ontwerp (Figuur 10.29). Dit zijn de onafhankelijke afmetingen van het ontwerp. Alle andere afmetingen zijn hieraan gerelateerd of zijn constant. Iedere kernmaat krijgt een unieke naam, bij voorbeeld de hartafstand tussen twee verbindingsgaten vastleggen met `steekx` en `steeky`. In de tabel wordt aan iedere kernmaat een waarde toegekend. Als er meerdere versies (varianten) worden ontworpen, dan in het EXCEL-werkblad voor iedere versie waarden toekennen aan de onafhankelijke variabelen, Figuur 10.29.
2. De parts waarvan de afmetingen samenhangen met de kernmaten, 'de maakonderdelen', worden gekoppeld aan de EXCEL-file. Bij het aanmaken van zo'n part worden de gegevens uit de EXCEL-file benut, bijvoorbeeld aan een afmeting van een part wordt de waarde $\text{lagerhoogte} + \text{steekx}/2 + 10$ toegekend.
3. Bij het assembleren worden ook de koop- of normdelen toegevoegd. Via de aan zo'n normdeel gekoppelde EXCEL-file kan worden aangegeven welke versie moet worden gebruikt, bijvoorbeeld een bout M8 met steellengte 35. Op deze wijze kan een ontwerp zeer gestructureerd worden opgezet. Voor het onderhoud van een ontwerp is dit van belang. De EXCEL-file is de leidraad van het ontwerp. De ontwerper legt door de keuze van de referentiematen zijn ontwerpplan vast, en eventueel wordt via commentaar zijn gedachtengang ook nog toegelicht in het EXCEL-werkblad.

	A	B	C	D	E
		steekx	steeky	lagerhoogte	lagerdia
versie1		80	70	90	10
versie2		150	50	40	30
versie3		150	60	70	20

Figuur 10.29: Deel van een EXCEL werkblad ONTWERP.XLS.

Voorbeelden van parts met samenhangende afmetingen

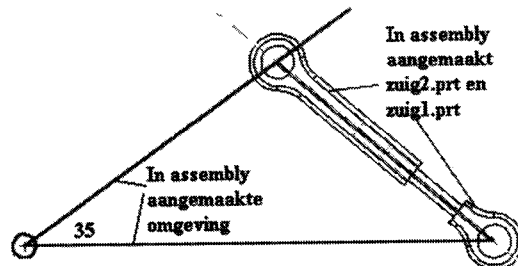
Aan de hand van drie voorbeelden wordt nader ingegaan op hoe in CAD pakketten afmetingen van verschillende parts aan elkaar kunnen worden gerelateerd.



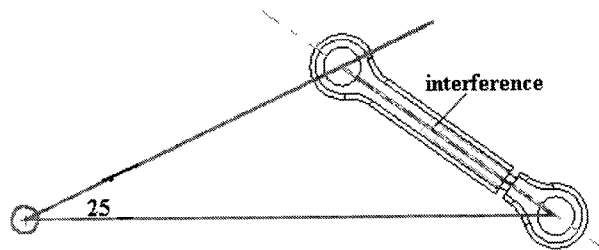
Figuur 10.30: Een partfile van een knop aanmaken via de assembly.

Voor de computermuis in Figuur 10.30 is begonnen met de behuizing te modelleren. Dit part wordt in een samenstellingstekening geplaatst. Voor het ontwerpen van de knop wordt in de samenstelling een New part geopend. In sommige pakketten, zoals SolidWorks, kunnen de afmetingen van het profiel van de knop rechtstreeks worden gerelateerd aan de knopopening van het huis, waarbij in dit voorbeeld aan de afmetingen van de knop nog een kleine offset wordt gegeven. Dit is een vorm van Top Down modelleren. Er zijn pakketten, zoals Mechanical Desktop, die niet de mogelijkheid van Top Down modelleren hebben, tenminste als daaronder wordt verstaan de mogelijkheid van rechtstreeks relateren van een afmeting van een part aan een afmeting van een ander part. In Mechanical Desktop kan, als aan een part wordt gewerkt, niet iets van een ander part worden geselecteerd; de andere parts in de samenstelling zijn niet actief aanwezig, ook al staan ze wel op het beeldscherm. Toch is het in Mechanical Desktop ook mogelijk de afmetingen van de diverse parts aan elkaar te relateren, namelijk via de

hiervoor genoemde EXCEL-file. Uit het oogpunt van onderhoud, wijzigen en gestructureerd werken is het alle en kunnen relateren van afmetingen van de diverse parts via een EXCEL-file te prefereren boven de mogelijkheid van rechtstreeks relateren.



Figuur 10.31: Een zuiger in zijn werkomgeving ontwerpen.

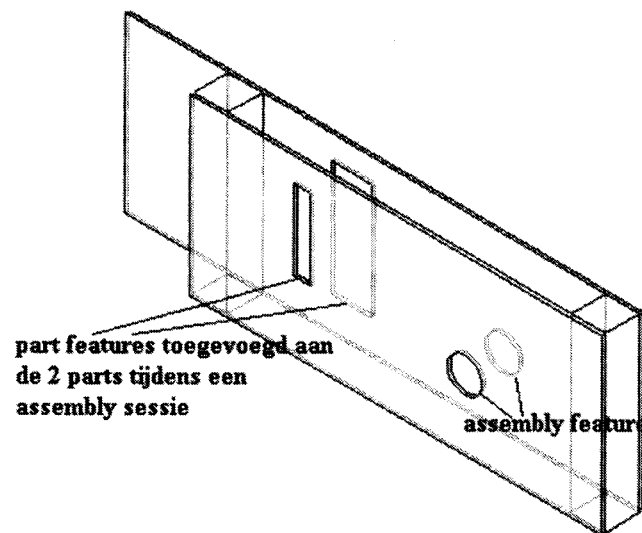


Figuur 10.32: Interference-check.

Het tweede voorbeeld betreft het ontwerpen van een zuigerconstructie. Figuur 10.31 toont de met SolidWorks vanuit een assembly gemodelleerde constructie. De omgeving waarin de zuiger moet werken was bekend (de 'stangenconstructie' AB en AC) en is als eerste in zijn uiterste stand (35 graden) in een assembly geplaatst. Daarna is een New part in de assembly geopend. Daarin is, gerelateerd aan de omgevingschets, zuig1.prt ontworpen. Vervolgens is in een New part zuig2.prt gemaakt, waarbij de dimensies zijn gerelateerd aan zuig1.prt. Om te controleren of er interference optreedt, is daarna in de omgevingschets de draaihoek veranderd in zijn minimale stand (25 graden). Na het commando UPDATE en een lange rekenpartij werd Figuur 10.32 verkregen. Hieruit blijkt dat de zuigerstang in die stand niet in de zuigerboring past, oftewel dat de boring dieper moet worden gemaakt.

Een edit zuig2.prt moet dus nog worden toegepast. Deze zuigerconstructie kan in Mechanical Desktop als volgt worden gemodelleerd: de omgeving (de 'stangen' AB en AC) als parts modelleren en wel als twee parts om de rotatie tussen 35 en 25 graden te kunnen realiseren in de samenstelling; de kernafmetingen van de twee zuigerparts in een EXCEL-file opnemen en met behulp daarvan de twee zuigerparts modelleren.

Vervolgens kunnen de vier parts worden geassembleerd (de twee zuigerparts kunnen met MATE LINE in lijn worden gebracht). De standen van 35 en 25 grad en kunnen met de assembly constraint ANGLE worden gerealiseerd, waarna de interference check kan worden uitgevoerd en, indien nodig, de boring dieper kan worden gemaakt door waarden in de EXCEL-file te wijzigen of door het desbetreffende zuigerpart actief te maken en te wijzigen



Figuur 10.33: Assembly features en part features aanmaken in een assembly.

Het laatste voorbeeld betreft de mogelijkheid om aan een 'halffabrikaat' features toe te voegen. De constructie, Figuur 10.32, bestaat uit vier parts, namelijk twee platen en twee steunen. Stel dat het geheel een onderdeel is van een mechatronisch ontwerp waarbij de platen dienen voor het ondersteunen van allerlei onderdelen. De sleuven in de twee platen dienen daarvoor aan elkaar te worden gerelateerd. Nadat de platen tijdens de fabricage aan elkaar zijn bevestigd, moet er als eindbewerking nog een gat dwars door de voor- en achterplaat worden geboord. Daarbij is het de bedoeling dat de gat-

informatie niet in de 'monotekeningen' voor de toeleverancier terecht komt. De toeleverancier moet de platen met al zijn sleuven afleveren, maar niet met het gat dat tijdens de eindbewerking moet worden aangebracht.

Er zal weer nagegaan worden hoe dit in SolidWorks en in Mechanical Desktop kan worden gerealiseerd. Het pakket SolidWorks maakt vanuit een assembly onderscheid in: features aan een part toevoegen; features aan een assembly toevoegen (assembly features).

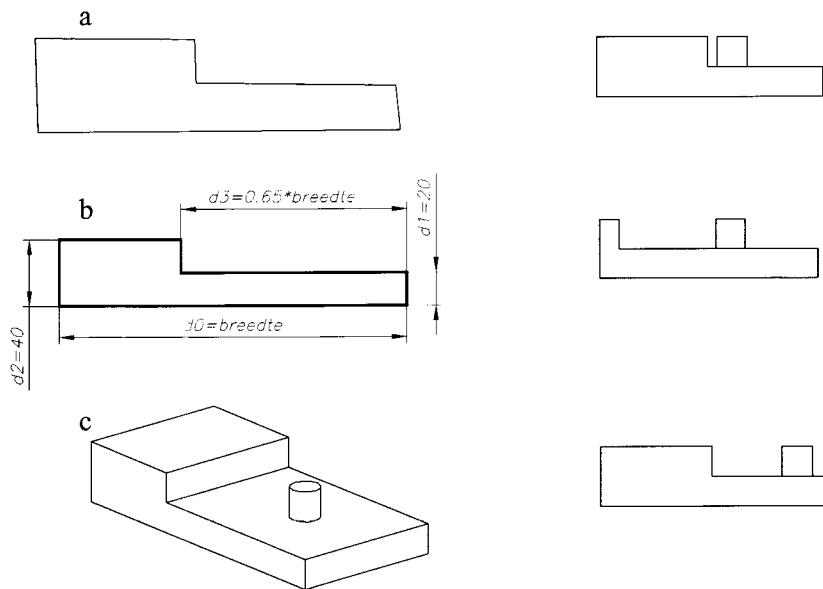
Nadat de platen, nog zonder sleuven, en de steunen zijn geassembleerd kan met EDIT PART een sleuf in de voorplaat worden aangemaakt. Daarna moet men de achterplaat selecteren en opnieuw EDIT PART activeren. De afmetingen van de sleuf in de achterplaat kunnen, zoals eerder opgemerkt, in SolidWorks rechtstreeks aan de afmetingen van de sleuf in de voorplaat worden gerelateerd. In Mechanical Desktop moet dit via de in een EXCEL-file opgenomen kernmaten. In SolidWorks wordt met het commando ASSEMBLY FEATURE een feature aangemaakt die alleen in de assembly komt, dus alleen in de .asm file en niet in de .prt files. Met dit commando wordt in SolidWorks het gat voor de eindbewerking aangebracht, zie Figuur 10.32. Om in Mechanical Desktop features aan een halffabrikaat toe te voegen, moeten eerst de afzonderlijke parts worden samengevoegd tot een nieuw part. Dit gaat met het commando AMCOMBINE. Het halffabrikaat wordt dus een nieuw part. In dit nieuwe part kan dan op de gebruikelijke wijze de feature HOLE worden aangebracht.

10.1.6 Wijzigen van parts

De resultaten van het wijzigen van een part zijn sterk afhankelijk van hoe het part is gemodelleerd; hier zal nog nader op worden ingegaan. Stel dat iemand de oorspronkelijke breedte van het part van Figuur 10.34 wijzigt van 210 in 150, welk model in de rechter figuur wordt dan verkregen? Dit hangt af van hoe er gemodelleerd is. Tekenaars die van oorsprong met 2D-CAD werken, hebben de neiging om het profiel (Figuur 10.34b) van de EXTRUDE feature een op een te tekenen, dat wil zeggen mete en op maat, en dan meteen het commando EXTRUDE te geven. Dit is in de meeste Solid Modelers toegestaan; het pakket neemt dan de een op een getekende maten over. Echter, met deze een op een werkwijze is niet van te voren te voorspellen waarin een wijziging van een afmeting zal resulteren. Slechts met trial en error is daar achter te komen. Deze een op een werkwijze is dan ook af te raden en is ook duidelijk niet in de geest van hoe in een parametric Solid Modeler een profiel moet worden aangemaakt.

In een Solid Modeler zoals Mechanical Desktop, waarmee Figuur 10.34 is aangemaakt, wordt bij het aanmaken van een EXTRUDE feature begonnen

met een sketch van het profiel, zie Figuur 10.34a. Deze schets legt alleen de topologie van het profiel vast en niet de geometrie. Na het schetsen wordt de geometrie vastgelegd. In Mechanical Desktop gebeurt dit met ADD CONSTRAINTS en ADD DIMENSIONS. Bij de overgang van schets naar profiel maakt Mechanical Desktop van lijnen die nagenoeg horizontaal of verticaal zijn geschetst automatisch exacte horizontale en verticale lijnen, dus de gebruiker hoeft deze constraints niet meer aan te brengen. Om het profiel van figuur 34b eenduidig vast te leggen (fully constrained te maken), moeten vier afmetingen met ADD DIMENSIONS worden ingevoerd. De gebruiker beslist welke, zie Figuur 10.34b. Uit deze figuur blijkt dat de maat breedte in een EXCEL-file is opgenomen. In deze EXCEL-file is aan de maat breedte de waarde 210 toegekend.



Figuur 10.34: Een maat wijzigen.

Als iemand later de breedte van de plaat van 210 in 150 wil veranderen, toont het commando EDIT FEATURE de afmetingen van Figuur 10.34b en kan hij zien hoe het part oorspronkelijk is gedimensioneerd. Hij ziet dan dat voor het veranderen van de breedte van de plaat in de bijbehorende EXCEL-file de waarde van 210 moet worden veranderd in 150 en wat daarvan het effect is op de rest. Geen van de in de rechter figuur getoonde modellen zal worden verkregen, omdat de oorspronkelijke ontwerper de breedte van de inkeping heeft gekoppeld aan de kernmaat breedte ($0.65 * \text{breedte}$, zie Figuur 10.34b). Indien gewenst kan hij ook deze relatie wijzigen of verwijderen. Als

het profiel een op een was getekend, dan zal het commando EDIT FEATURE geen enkele maat op het scherm tonen, omdat er met ADD DIMENSION geen enkele maat is aangebracht. Zoals al gezegd is een andere reden om de maten met ADD DIMENSION aan te brengen dat alleen de met ADD DIMENSION ingevoerde maten automatisch in een 2D-tekening worden gezet.

10.1.7 High end en mid range 3D-pakketten

De eerste 3D-CAD pakketten, zoals UNIGRAPHICS en CATIA, waren surface modelers en werkten alleen op werkstations. Specifieke industrieën, zoals de automobiel- en luchtvaartindustrie, maakten gebruik van deze pakketten. Een aantal jaren geleden kwamen de Solid Modelers op de markt, onder andere Pro Engineer. Pro Engineer is vanaf de basis ontwikkeld als parametric solid modeler. De concurrentie kon niet meteen volgen, omdat ze gebonden waren aan de eerder gekozen technologie. Door de ontwikkelingen op pc-gebied kwamen ook Solid Modelers op pc's beschikbaar, bijvoorbeeld SolidWorks, Solid Edge en Autodesk Mechanical Desktop. De benodigde investeringen voor soft- en hardware zijn voor deze pakketten aanmerkelijk lager dan de door de gevestigde 3D-CAD leveranciers geleverde Solid Modelers.

Men noemt de pc-pakketten wel de mid range en de workstation-pakketten de high end pakketten. Dit onderscheid is voor wat betreft de modelleermogelijkheden van de Solid Modelers niet terecht. De verschillen zijn gering. Wel een verschil is dat in de high end pakketten allerlei modules, zoals een CAM-module, zijn geïntegreerd met voor alle modules een gemeenschappelijke database. Een ander verschil is dat er een goed functionerende interface is met bij voorbeeld een EEM-pakket. Mid range pakketten worden daarom ook wel aangeduid met CAD-centric en de high end pakketten met proces-centric. Voor de overlevingskansen van de genoemde mid range pakketten is waarschijnlijk van belang dat de makers van deze Solid Modelers zorgen voor een solide interfacing met andere pakketten.

Voor het uitwisselen van CAD-data tussen verschillende pakketten zijn er verschillende mogelijkheden: directe vertalers; uitwisseling gebaseerd op de softwarekernel van een pakket; een neutraal file-formaat.

Voor directe vertalers is de onderhoudsinspanning die nodig is voor de diverse vertalers een probleem, bijvoorbeeld als wordt overgegaan op een nieuwe release van een pakket. Uitwisseling gebaseerd op de softwarekernel komt in aanmerking als een bepaalde softwarekernel in vele pakketten wordt toegepast. Vele Solid Modelers en dan met name de mid range pakketten hebben een softwarekernel die door een derde partij zijn ontwikkeld. De meest gangbare softwarekernels zijn ACIS (Mechanical Desktop, Solid Designer) en

Parasolid (Unigraphics, Solid Edge, SolidWorks). Voor CAD-gebruikers zou het een goede zaak zijn dat alle CAD-pakketten zijn gebaseerd op eenzelfde softwarekernel, zodat de uitwisseling van geometrische gegevens daarop kan worden geënt. Een tijdlang leek dit ACIS te worden met het bijbehorende SAT-uitwisselingsformaat. Helaas voor de gebruikers wijzen de huidige marktontwikkeling van CAD-pakketten niet meer in de richting van een de facto standaard softwarekernel. Het gangbaarst op dit moment is de uitwisseling via het neutrale formaat IGES, terwijl uitwisseling via STEP in opkomst is. In de praktijk is voor veel bedrijven het uitwisselen van CAD-data een grote zorg. Met veel uitproberen en vervolgens gedegen afspraken maken tussen de twee uitwisselingspartijen kunnen redelijke resultaten worden verkregen, maar het is gewenst dat de leveranciers van CAD-pakketten meer oog zouden hebben voor de uitwisselingsproblematiek.

10.1.8 Keuzecriteria

Een aantal aspecten van Solid Modelers is belicht. Een antwoord geven op de vraag 'Welk pakket kun je het beste aanschaffen' is moeilijk te geven. Vele zaken spelen daarbij een rol. Van bedrijf tot bedrijf zal aan ieder aspect een andere gewichtsfactor worden toegekend. Een aantal keuzecriteria zijn:

1. Interfacing met andere pakketten (CAD, CAM, EEM, et cetera). Het is aan te raden om van een voor aanschaf in aanmerking komend pakket de IGES- en STEP-faciliteiten te testen.
2. Specifieke mogelijkheden, zoals het uitslaan van plaatconstructies (sheet metal).
3. Parts met dubbel gekromde oppervlakken (met REVOLVE kunnen geen dubbel gekromde oppervlakken worden aangemaakt).
4. Parts met complexe afrondingen (fillets).
5. Afleveren van drawings die zonder veel ingrijpen aan de gestelde eisen voldoen (zowel de mono-2D-tekeningen als de samenstellings-2D-tekeningen).
6. Assemblies met Top Down faciliteiten.
7. De kernmaten via een EXCEL-file koppelen aan het ontwerp.

8. De gebruikersvriendelijkheid (voor een beginner van belang, voor een ervaren gebruiker minder; voor een veelvuldig gebruiker van het CAD-pakket zijn de mogelijkheden van het pakket, de functionaliteit, belangrijker dan hoe die mogelijkheden moeten worden benut)
9. Het marktperspectief van het pakket.

10.1.9 Solid modeling versus 2D-CAD

Als voordelen van een Solid Model kunnen worden genoemd:

- een betere visuele beoordeling;
- minder kans op tekenfouten, omdat de 2D-werktekeningen eenduidig uit het Solid Model worden onttrokken;
- massa- en massatraagheidsmomenten-berekeningen door het CADpakket;
- een samenstelling (assembly) van het ontwerp uit de als Solid Model gegenereerde onderdelen aanmaken;
- vanuit de samenstelling een exploded view genereren;
- interference checking, materiaal-overlap van onderdelen in de samenstelling controleren;
- montage van onderdelen vanuit de samenstelling beoordelen;
- het hergebruik van aangemaakte parts;
- optimalisatie met betrekking tot sterkte en stijfheid via een EEM-berekeningen (indien een solide koppeling met een EEM-pakket beschikbaar is);
- voor rapid proto typing (Layer Manufacturing Technologies) de benodigde STL-file aanmaken;
- rechtstreekse overdracht van geometrie naar CAM;
- de maakbaarheid van het ontwerp nagaan (bij voorbeeld voor freesbewerkingen als een koppeling met een CAM-pakket beschikbaar is).

Deze opsomming zal weinig aanleiding geven tot discussie. Daarom tot slot nog een aantal wat krassere beweringen:

- Solid Modeling heeft geen nadelen, hooguit kun je zeggen dat:

- de ene Solid Modeler meer functionaliteit heeft dan een andere
- op dit moment het uitwisselen van Solid Model data tussen de diverse Solid Modelers via bijvoorbeeld IGES of STEP nog veel problemen oplevert, maar dat te verwachten is dat, net als met 2D-data dit probleem op den duur aanmerkelijk minder wordt.
- Met betrekking tot 2D-tekeningen zitten we in een overgangstijd. Op den duur zullen ze niet meer nodig zijn.
- De huidige gebruikers van 2D-tekeningen, zoals werkvoorbereiders, zullen te zijner tijd op een beeldscherm het Solid Model van het product voor zich hebben en kunnen, indien nodig, met een commando als DISTANCE alle gewenste maten in het Solid Model opvragen. Het is zelfs de vraag of dit dan nog wel nodig is, omdat het ter beschikking staande CAM-pakket zelf alle benodigde informatie uit het Solid Model onttrekt.
- De primaire taak van 2D-tekenregels is het kunnen interpreteren van een 3D-object, ofwel het kunnen krijgen van een correct beeld van het in werkelijkheid driedimensionale object. Het driedimensionale object uit zulke 2D-tekeningen halen is een hele kunst en vergt veel oefening. Met de komst van de Solid Modelers vervalt deze primaire taak, want iedereen die vanuit het Solid Model 2D-tekeningen aanmaakt zal in die 2D-tekening ook een isometrisch aanzicht plaatsen, eventueel meerdere als dat voor het interpreteren van het 3D-object gewenst is.
- In de overgangstijd moeten in werktuigbouwkundige opleidingen nog wel het leren lezen van een 2D-tekening worden opgenomen, maar is het zelf leren aanmaken van 2D-tekeningen tijdverspilling. Wel is van veel belang dat wordt geleerd hoe aan een uit het Solid Model onttrokken 2D-tekening voor een productieafdeling informatie wordt toegevoegd, bijvoorbeeld passingen, maat- en vormtoleranties, etc. Het is te verwachten dat dit soort informatie op den duur via genormaliseerde 3D-tekenregels aan een Solid Model kan worden toegevoegd.
- In werktuigbouwkundige opleidingen moet meteen vanaf het begin worden begonnen met Solid Modeling.
- Het kunnen maken van 3D-handschetsen blijft zeer belangrijk (bij het opzetten van een Solid Model moet daar ook altijd mee worden begonnen).

- De softwareproducenten van Solid Modelers besteden alleen maar veel aandacht en dus tijd aan het drawing-deel van hun product omdat de huidige markt er nog om vraagt.
- Invoering van gestandaardiseerde (ISO) 3D-tekenregels is gewenst.
- Het is de vraag of het aanmaken van plots nog wel zo nodig of zelfs nog gewenst is. Bij een presentatie van het product via een beeldscherm blijft de verbinding met de tekeningdatabase, dus de echte data, in stand. Bij een plot is deze verbinding verbroken. Inconsistentie en redundantie van productdata dreigt daardoor. Bij het opzetten van productdata-informatie is juist een eerste vereiste het voorkomen van inconsistente data binnen een bedrijf. Toch zal het voorlopig voor werkbesprekingen nog veel waarde hebben om de beschikking te hebben over plots met daarop onder andere de belangrijkste maten. De tijd van werkoverleg met een groot scherm of iedereen met een plat beeldscherm voor zich, waarop een pc de tekeningen toont, zal nog wel even duren.
- Een ander punt met betrekking tot plots is, dat op dit moment een plot van een product-ontwerp het enige is dat als een juridisch document kan worden aangemerkt. Dit is ongewenst. In een tijd dat tegelijkertijd aan eenzelfde ontwerp in Nagano en Nederweert kan worden gewerkt, dient een file voorzien van een elektronische handtekening een juridisch document te kunnen zijn.
- Soms wordt, hoewel twijfelachtig, als een nadeel van Solid Modeling genoemd dat het aanmaken van de producttekeningen langer duurt, zeker in het begin. Bekijken we echter het gehele proces, dan zal, op grond van de hierboven genoemde voordelen van Solid Modelers, toch meestal een behoorlijke besparing in de productontwikkelingstijd kunnen worden verkregen.

10.1.10 Communicatie van CAD gegevens

Productgegevens hebben een complexe structuur, zeker wanneer het gaat om complete productmodellen of geometrische modellen. Het is niet eenvoudig om productgegevens te communiceren naar een partij waarvan de informatiebehoefte deels of geheel onbekend is. Dit is typisch het geval bij de uitwisseling van CAD-modellen van het ene systeem naar het andere. Waarschijnlijk zijn de intenties van de ontvangende partij wel bekend, maar niet hoe precies zijn CAD-systeem (of andere softwareapplicatie) zal reageren op de verstuurd gegevens.

Wanneer is uitwisseling van CAD-modellen nodig?

De situaties waarin het versturen van CAD-gegevens nodig is vallen uiteen in drie groepen:

1. Er moet afstand overbrugd worden. Om puur logistieke redenen kan dit voorkomen, bijvoorbeeld omdat een ontwerper verscheidene werkplekken heeft en 'zijn' model steeds meeneemt naar een andere locatie. Ook kunnen twee of meer personen afwisselend aan hetzelfde model werken, ofwel gebruik makend van hetzelfde werkstation ofwel geografisch verspreid werken.
2. Er moet tijd overbrugd worden. Deze situatie wordt in principe door een database systeem afgedekt, maar de database van een CAD-systeem is meestal pakkeetspecifiek. Dit is een nadeel voor langere termijn archivering, men gebruikt dan liever een extern gedefinieerd formaat, bijvoorbeeld dat van een uitwisselingsstandaard. Er is hier dus sprake van gegevensuitwisseling, zij het dat de ontvangende partij (nog) niet bekend is.
3. Er moet functionaliteit toegevoegd worden. Dit is de meest voorkomende reden voor het uitwisselen van productgegevens. In de eerste plaats denken we aan overbrengingsmomenten in het primaire proces in de ontwerpcyclus. De belangrijkste hiervan (maar zeker niet de enige) zijn (a) de overdracht van modellen van conceptueel ontwerp naar engineering, (b) van engineering naar fabricagevoorbereiding (Computer-Aided Manufacturing), (c) van engineering (en soms ook conceptueel ontwerp) naar analysepakketten en prototypingfaciliteiten. Het betreft hier informatiestromen in de downstream-richting, die van oudsher bestaan in de productontwikkeling. Juist door de toepassing van digitale technieken wordt het steeds aantrekkelijker om gegevens ook te versturen in de stroomopwaartse richting, bijvoorbeeld van analyse terug naar ontwerp, of tussen twee of meer ontwerpende partijen onderling, bijvoorbeeld ter ondersteuning van concurrent engineering. Uiteraard gebeurde dit altijd al op informele, dat wil zeggen niet-digitale, wijze. Het blijkt echter dat er grote problemen ontstaan zodra CAD-systemen (of andere softwarepakketten) die een ongelijke functionaliteit hebben, gaan communiceren.

Voordat we op de zojuist genoemde problematiek ingaan, eerst een kort overzicht van de techniek van gegevensuitwisseling tussen CAD-systemen. Het principe is tot op zekere hoogte vergelijkbaar met de uitwisseling van

tekstdocumenten tussen verschillende PCs. Als twee PCs dezelfde tekstverwerker hebben, dan vergt de communicatie niet meer acties dan respectievelijk 'bestand opslaan als...' en 'bestand openen ...', waarbij de fysieke communicatie via diverse media (schijf, netwerk, CD etcetera) kan verlopen. Op dezelfde wijze kan een CAD-model van systeem A naar systeem B worden gestuurd. Zolang op beide systemen exact dezelfde CAD-software wordt gebruikt treden er technisch geen problemen op. Dit noemt men rechtstreekse koppeling via systeem-specifieke files, zie Figuur 10.35a. Dit soort bestanden worden Binary Large Objects (BLOB) genoemd, omdat het objecten in binaire vorm zijn en in vergelijking tot andere (niet-CAD) gegevens veel opslaggeheugen gebruiken. Let op dat er los hiervan wel organisatorische vraagstukken kunnen liggen, op het gebied van versiebeheer en autorisatie. Technisch gezien vereist een rechtstreekse koppeling dat de interne database van systeem B precies gelijk is aan de database van systeem A. Alle modeleringfuncties die met A mogelijk waren moeten ook op systeem B mogelijk zijn, met andere woorden op systeem B kan zonder extra beperkingen worden doorgewerkt met het model dat van A afkomstig was. Wanneer de systemen A en B van verschillend type zijn, dan zal een file afkomstig van A in het algemeen niet zonder meer te gebruiken zijn op B. Er zal een conversieslag moeten plaatsvinden met speciaal voor dat doel gemaakte software, zie Figuur 10.35b. Het grote probleem doet zich nu voor dat afhankelijk van de kwaliteit van die conversieslag het model meer of minder bruikbaar zal zijn. In veel gevallen is een zinvolle conversie zelfs uitgesloten. Neem als extreem voorbeeld dat systeem A een 2D tekenpakket is en systeem B een 3D solid modeler. Het model uit systeem A bevat te weinig informatie om door systeem B zelfs maar als ontwerpobject herkend te worden. Als systeem B echter ook het creëren van 3D objecten vanuit 2D objecten ondersteunt, dan zou communicatie in principe mogelijk zijn. In de praktijk doen zich de meeste problemen juist voor bij de uitwisseling van modellen tussen twee CAD-systemen die 'ongeveer' op elkaar lijken. Dit zal vaak het geval zijn bij toepassingen waar het de bedoeling is dat het werk op systeem A wordt voortgezet op systeem B.

Standaarden voor de uitwisseling van productgegevens

Er ontstond behoefte aan uitwisselingsstandaarden voor productgegevens, als alternatief voor de systeemspecifieke bestanden. De belangrijkste motieven voor de ontwikkeling van standaarden zijn:

1. De in- en uitvoersoftware voor het ontvangen en versturen van CAD-gegevens wordt eenvoudiger, vooral de invoersoftware. Deze hoeft nu

Figuur 10.35: Drie veel voorkomende manieren om een CAD-model van het ene naar het andere systeem uit te wisselen: a) rechtstreeks, b) met conversie om de gegevens geschikt te maken voor het ontvangende systeem, c) gebruikmakend van een uitwisselingsstandaard (in de tekening IGES genoemd, maar dat kan door de naam van een andere standaard vervangen worden).

alleen geschikt te zijn voor bestanden in standaardformaat, en niet voor velerlei systeemspecifieke formaten.

2. De kwaliteit van de communicatie wordt minder gevoelig voor veranderingen van de betrokken systemen. Als een nieuwe softwareversie wordt genstalleerd op systeem C dan zullen (de gebruikers van) de systemen A en B daar in principe geen rekening mee te houden; alle wijzigingen moeten door de in- en uitvoersoftware van systeem C opgevangen worden; het formaat van de uitgewisselde files verandert niet.
3. De inhoud van de uitgewisselde bestanden is beter gedocumenteerd; de betekenis van de objecten en attributen zijn in de standaard gepubliceerd. Daardoor wordt ook het maken van de in- en uitvoersoftware vergemakkelijkt.
4. Samenhangend met voorgaande punt levert een standaard een voordeel op voor archivering van productgegevens. De kans dat na langere tijd standaardfiles nog interpreteerbaar zijn is groter dan die van systeemspecifieke files.

Als extra voordeel kan genoemd worden het feit dat er door standaardisatie een forum ontstaat van CAD-leveranciers en gebruikers, waardoor er gelegenheid is om ook op andere punten dan het uitwisselingsformaat naar elkaar toe te groeien. Dit heeft er wellicht toe bijgedragen dat thans zowel de user-interfaces alsook basisonderdelen van de software, de zogenaamde kernels, van verschillende CAD-pakketten gelijk getrokken zijn. Dit is zeker in het belang van degenen die modellen uitwisselen tussen verschillende systemen. Op het ogenblik zijn vooral de standaarden IGES (Initial Graphics Exchange Specification) en DXF (Data eXchange Format) van praktisch

belang, alsmede de nieuwe ISO-standaard IS10303, of STEP. Het principe van het gebruik van een standaard voor het versturen van CAD-modellen is aangegeven in Figuur 10.35c. Het is de bedoeling dat de leverancier van een CAD-systeem zorgt voor in- en uitvoersoftware voor IGES en/of andere standaarden. Deze software wordt soms aangeduid met respectievelijk de term postprocessor en preprocessor. Een preprocessor (laten we zeggen van CAD-systeem A) maakt van een CAD-model een file in het formaat voorgeschreven door de standaard. Deze file kan verstuurd worden naar systeem B, alwaar het eerst verwerkt wordt door de postprocessor. Deze vertaalt de standaard file naar een CAD-model geschikt voor systeem B. Deze vertaalslag houdt meer in dan een syntactische omzetting; het kan voorkomen dat de standaard file gegevenstypen bevat die op systeem B niet gedefinieerd zijn. Dit is in wezen dezelfde situatie die optreedt bij uitwisseling zonder standaard. Vandaar dat in Figuur 10.35c is aangegeven dat conversie zal moeten plaatsvinden, ondanks het gebruik van een standaard. Opgemerkt kan worden dat ook tijdens de uitvoerfase een vertaalslag plaatsvindt van de interne DB van het CAD-systeem naar het standaardformaat. In veel gevallen kan dit beperkt blijven tot een syntactische omzetting in het geval de standaard redundantie in datatypen biedt. Een cirkel, bijvoorbeeld, kan in het ene CAD-systeem d.m.v. middelpunt en straal, maar in een ander CAD-systeem d.m.v. een NURBS curve gedefinieerd zijn. Voor het schrijven van deze gegevens naar een IGES-file, is in beide gevallen geen conversie nodig omdat beide representatievormen ondersteund worden door de IGES-standaard. Bij het inlezen zal er conversie nodig zijn als systeem B de cirkel anders representeert dan systeem A.

Hoewel STEP (STandard for the Exchange of Product model data) momenteel nog niet de meest gebruikte standaard is voor uitwisseling van productgegevens besteden we er hier toch extra aandacht aan. Omstreeks 1984 is begonnen met de ontwikkeling van een internationale standaard, dus onder auspiciën van ISO (International Organization of Standardization). Dit gebeurde uit onvrede van zowel CAD-leveranciers als CAD-gebruikers, die inzagen dat de vele toen bestaande nationale standaarden moesten worden vervangen door een wereldstandaard. Bovendien had men lering getrokken uit de ervaringen opgedaan met het gebruik van IGES en andere standaarden. Er zijn daarom voor STEP procedures gemaakt voor een systematische ontwikkeling en zonodig aanpassing van de standaard. In sterk vereenvoudigde vorm is die procedure in Figuur 10.36 geschetst. Enkele principes die we bij het opzetten van een database tegenkwamen vinden we ook nu weer terug, namelijk de stappen analyse van de informatiebehoeften, het opstellen van een gegevensmodel (de entiteiten) en het specificeren van een dataschema. Dit laatste gebeurt in STEP met behulp van de taal EXPRESS. Dit is een

codering die enigszins lijkt op een programmeertaal, maar tegelijkertijd ook voor mensen goed leesbaar is. Dit betekent dat ISO 10303 niet alleen een beschrijving is van de STEP-standaard, maar tevens digitale bestanden omvat met de dataschemas. Deze bestanden worden EXPRESS files genoemd, die uiteraard onderscheiden moeten worden van STEP-files. In EXPRESS kunnen de objecttypen en attributen worden gespecificeerd die met STEP-files uitgewisseld kunnen gaan worden. Deze objecttypen hoeven niet per se in alle CAD-systemen ook te bestaan, als het maar mogelijk is om ieder mogelijk productmodel van het CAD-systeem te vertalen in STEP-objecten. Een groot voordeel van STEP is dat een deel van de in- en uitvoersoftware automatisch vanuit EXPRESS-files kan worden gegenereerd. Dat betekent dat als STEP gewijzigd of uitgebreid wordt, de inspanning om de software aan te passen relatief klein is. Tenslotte worden ook binnen de standaard testprocedures beschreven en voorgeschreven voor de in- en uitvoersoftware. De belangrijkste onderdelen (Parts) van STEP zijn:

- Part 1, algemene inleiding tot ISO 10303.
- Part 11, definitie van de EXPRESS-taal. Deze taal heeft een formele syntax en is interpreteerbaar door mensen en door computers.
- Part 21, het STEP file-formaat. Dit beschrijft het formaat waarin STEP-objecten gecodeerd worden. Ook deze taal heeft een formele syntax en is uiteraard door de computer leesbaar.
- Parts 41 49, algemene gegevensstructuren (objecttypen), beschreven in EXPRESS.
- Parts 201 230, Application Protocols. Dit zijn applicatie-specifieke of bedrijfstak-specifieke beschrijvingen van modellen die men via STEP wil uitwisselen. Deze APs worden voor een belangrijk deel eveneens in EXPRESS gedefinieerd, waarbij gerefereerd wordt naar de Parts 4149.

In 1994 is de "Initial Edition of STEP" door ISO geregistreerd als standaard. Veel CAD-leveranciers hebben STEP op een of andere wijze aan hun softwareproduct toegevoegd. Hoewel de ISO-norm het officieel nog niet toelaat wordt al druk geëxperimenteerd met STEP data sharing in plaats van file exchange. De gedachte hierachter is dat onder andere concurrent engineering een gemeenschappelijk productmodel vereist. Binnen ISO wordt nu gewerkt aan de SDAI (Standard Data Access Interface) die het mogelijk maakt een applicatie te laten werken met verscheidene repositories. Er wordt dan open gelaten of een repository een database management systeem, een file-systeem of een intern werkformaat is. Wel wordt exact gespecificeerd hoe een d.m.v. EXPRESS gedefinieerd object moet worden opgeslagen en opgevraagd.

Figuur 10.36: Schematische weergave van de STEP-methodologie. Er zijn twee informatiestromen te onderscheiden. (a) de communicatie van gegevens tussen twee CAD/CAM-systemen via een STEP exchange file en (b) (in stippellijnen) de richtlijnen voor het vervaardigen van de STEP invoer- en uitvoersoftware..

10.2 Product Data Management

In het voorgaande is besproken hoe productinformatie met behulp van computers wordt gegenereerd, vastgelegd en uitgewisseld. Veel van deze digitale informatie is essentieel om op de juiste plaats en op het juiste moment beslissingen omtrent productontwikkeling of bedrijfsvoering te kunnen nemen. Naast de digitalisering is er een grote stijging in het volume van productgegevens. Dit wordt veroorzaakt door ontwikkelingen als engineering-on-order, een toenemende complexiteit van producten en kortere levenscycli van producten. Het beheer van gegevens en met name productgegevensbeheer (Product Data Management, PDM) is dan ook een groeiend vakgebied. In deze paragraaf zullen we gegevensbeheer dan ook vanuit het oogpunt van PDM bekijken. Zoals bekend zorgt internet voor een vergrote verspreiding van gegevens. Informatie over bijvoorbeeld leveranciers, onderdelen en materialen kan van het internet worden gehaald.

Ontwerpteamen die bestaan uit meerdere groepen of personen die op verschillende locaties werken, gebruiken het internet om beter te kunnen samenwerken. Dit gaat niet vlekkeloos; verschillen in bijvoorbeeld cultuur zorgen ervoor dat gegevens op een andere manier worden vastgelegd of dat er een andere waarde aan wordt toegekend. De taken van de verschillende groepen moet op elkaar worden afgestemd om te voorkomen dat er dubbel werk wordt gedaan. We hadden al gezien dat verschillen in computerprogramma's vereisen dat er enige afspraken zijn omtrent de manier waarop en in welke vorm gegevens worden gebruikt. Deze communicatieproblemen bij samenwerkingsverbanden leggen dus ook nadruk op het belang van gegevensbeheer.

10.2.1 Productgegevens en PDM

Alvorens te beschrijven wat PDM inhoudt, wordt er eerst gekeken naar de gegevens die beheerd dienen te worden: productinformatie of productgegevens, waarbij we onderscheid zullen maken tussen de gegevens die een product omschrijven, aan de ene kant en beheersgegevens aan de andere kant. Hoe de productgegevens er in werkelijkheid uitzien is grotendeels afhankelijk van het product en de kennis rond het product en het bedrijf. De gegevens die worden vastgelegd zijn deels ook afhankelijk van de hoeveelheid en diversiteit van personen die in het ontwerpproject participeren. Deze mensen genereren, afhankelijk van hun taak, verschillende typen gegevens en gebruiken en communiceren deze gegevens van respectievelijk met elkaar. Om een wat helderder beeld te scheppen, is er een provisorische scheiding in deze gegevens aangebracht.

10.2.2 Productbeschrijving

Met productbeschrijving wordt hier bedoeld op de gegevens die de verschijningsvorm van het product vastleggen. Productontwerpen worden in computers over het algemeen opgeslagen met behulp van CAD-systemen. Deze systemen bieden de mogelijkheid om ontwerpen in de vorm van een technisch tekening of 3D model te bewerken en op te slaan (zie vorige paragraaf). Voor andere vormen waarin een ontwerp is beschreven, zoals schetsen, kleurstudies, schuimmodelletjes, textuurproeven, etc. bestaan er maar beperkte opslagmogelijkheden in een computer: deze gegevens kunnen worden omgezet in en opgeslagen als afbeelding. Hoewel de inhoudelijke kwaliteit met deze vertaalslag vaak achteruit gaat, maakt het wel de communicatie van de gegevens over datanetwerken, zoals het internet, mogelijk. Wanneer dergelijke gegevens met bijvoorbeeld grafische illustratie-programmas op de computer zijn gemaakt, treedt dit kwaliteitsverschil niet op: in dat geval kunnen de gegevens in het formaat dat bij de programmas hoort of in een neutraal formaat worden opgeslagen. Een andere vorm waarin ontwerpen worden beschreven zijn presentaties in de vorm van renderings en video-animaties. Dit zijn meestal visualisaties van de geometrische beschrijvingen uit schetsen, technische tekeningen en CAD-modellen.

10.2.3 Ontwerp-rationele gegevens

Bij ontwerp-rationele gegevens doelen we op gegevens die de ontwikkeling van een ontwerp sturen en bepalen. Dergelijke kennis stuurt de ontwerper bij zowel de divergerende als convergerende gedeelten in het ontwerpproces. In

het convergerende gedeelte kan de kennis bijdragen bij ontwerpbeslissingen. Hier valt te denken aan een programma van eisen of een kostprijsberekening. Kennis draagt ook bij in het creëren van ontwerpvarianties, oftewel in de divergerende trajecten in het ontwerpproces. Zo valt uit een analyse van een product een morfologische kaart af te leiden, die op zijn beurt kan helpen bij het genereren van productvarianten. Andere voorbeelden van kennis omtrent het product zijn:

- De meeste Design for X technieken (Design for Assembly, Design for Manufacture, etc.) leveren kennis op die gebruikt kan worden om zowel nieuwe, innovatieve oplossingen te bieden, evenals om de prestatie van een ontwerp op een gebied X te toetsen,
- Marktkennis draagt bij aan het programma van eisen,
- Simulatie helpt bij het maken van ontwerpbeslissingen. De opslag van dergelijke kennis kan tegenwoordig plaatsvinden in zogenaamde kennisystemen. Kennissystemen bezitten de gave om kennis die wordt gebruikt te herkennen, op te slaan en te modelleren. Andere, meer gangbare vormen om kennis in op te slaan zijn databases, spreadsheets, etc.

10.2.4 Bibliotheken

Bibliotheken bevatten herbruikbare onderdelen, ontwerpen of kennis met een algemene geldigheid. Voorbeelden van bibliotheken zijn:

- Standaardvarianties op problemen, zoals bekende constructieoplossingen: voor mechanische overbrengingen kunnen we kegeltandwielen, wormwielen, tandwielen met ketting etc. onderscheiden,
- Bibliotheken met standaardonderdelen, features, symbolen,
- Bibliotheken of handboeken met normen en standaards rondom het product, het bedrijf of de productiefaciliteiten,
- Informatie over concurrerende producten,
- Corporate Design handboek.

Bij procesgegevens valt te denken aan gereedschap-ontwerpen, werk-instructies, kwaliteitscertificering van de productie (ISO 9000 implementatie) en dergelijke. Onder projectgegevens vallen gegevens rond de planning van het ontwerpproject. Hierbij valt denken aan een projectplan, taakbeschrijvingen

en planningen voor prototypebouw en tests. De voorgaande paragrafen hebben laten zien dat gegevens in het productontwikkelingsproces erg divers zijn in hun voorkomen. Deze diversiteit maakt het gegevensbeheer in de consumenten/fabricage-industrie veel complexer dan in een bedrijf dat alleen met tekstuele of getalsmatige gegevens werkt. Veel functies in het PDM-systeem trachten orde te scheppen in deze gegevens. In de volgende paragraaf wordt het productgegevensbeheer toegelicht.

10.2.5 Beheer van productgegevens

PDM vindt vooral ingang bij grote ontwerpprojecten waar veel personen bij zijn betrokken. Des te meer personen in een ontwerpproject zijn betrokken, des te belangrijker wordt het om op basis van kennis met elkaar te kunnen communiceren en des te meer gegevens worden er aangemaakt. Het probleemgebied PDM speelt dan ook sterker in grote organisaties, zoals de automotieve industrie, dan in kleine organisaties, zoals ontwerpbureaus. Toch zullen ook deze kleine organisaties zich rekenschap moeten geven van de problemen en mogelijkheden rond productgegevensbeheer. Ze krijgen steeds meer te maken met (complexe) samenwerkingsverbanden die de behoefte aan productgegevensbeheer doet groeien. Niet alleen het aantal personen maakt dus dat productgegevensbeheer belangrijk wordt, ook het aantal groepen in het ontwerpproject en de complexiteit van de communicatie tussen die groepen legt een grote druk op PDM. PDM richt zich in eerste plaats op productgegevens, maar bij de opslag en het beheer van productgegevens moet ook rekening worden gehouden met de dynamische omgeving waarin productgegevens worden gebruikt. De scheidslijn tussen informatie die bij het product hoort en de marketing-, productie-, bedrijfsinformatie is weliswaar strikt, maar in de praktijk wordt deze productinformatie door diverse personen in een bedrijf gebruikt. Andere gegevens hebben hun invloed op productgegevens en deze beïnvloeden op hun beurt de andere gegevens weer. Informatie zoals de mogelijkheden van het machinepark en de marktgegevens, beïnvloedt bijvoorbeeld de keuze tussen ontwerpvarianten. In z'n dynamische omgeving is het van belang dat wordt bijgehouden waar de gegevens toe dienen. PDM richt zich op het beheer van productgegevens met als doel deze gegevens zo optimaal mogelijk te kunnen gebruiken. Voordat er wordt ingegaan op de taken van PDM, zal er eerst getracht worden de plaats van PDM binnen de gegevens en gegevensstromen binnen het bedrijf te beschrijven.

10.2.6 Plaatsing van PDM binnen het bedrijf

PDM heeft raakvlakken met andere gegevens binnen de organisatie. Bij veel bedrijven vindt de productontwikkeling plaats in de afdeling 'Engineering'. Binnen de bedrijfsfunctie Engineering is gegevensbeheer nodig en daarnaast is er gegevensbeheer nodig dat niet is gebonden aan een bedrijfsfunctie, maar over alle bedrijfsfuncties heen loopt en soms nog verder naar andere bedrijven, denk aan marketing, productie, verkoop, onderhoud etcetera. EDM (Engineering Data Management) behandelt alle gegevens, die tijdens de engineering fase relevant zijn, gericht op het optimaal laten verlopen van het engineering traject. EDM heeft een overlapping met onder andere PDM. Deze overlapping betreft alleen de productinformatie die in het engineering traject van belang is. PDM behandelt de productgegevens die gebonden zijn aan het product- en procesontwerp over de gehele productlevenscyclus, van eerste marketing concept (ontstaan) tot beindiging van het gebruik van het product. PDM heeft qua gegevens die beheerd moeten worden dus niet alleen een overlapping met EDM, maar met alle andere gegevensbeheergebieden. Met andere woorden: EDM gaat in de diepte (veel soorten gegevens, klein deel van de productlevenscyclus) en PDM gaat in de breedte (klein deel van de gegevens, over de gehele productlevenscyclus). Doel van gegevensbeheer Zoals in de inleiding al is aangegeven, wordt de druk op goed gegevensbeheer steeds groter. Het PDM-systeem tracht met een goed beheer van productgegevens het bedrijf daarbij te ondersteunen. De doelen die daarvoor worden nagestreefd zijn:

- Gegevensstromen optimaliseren: het vermijden van bottlenecks in de stroom van gegevens door een bedrijf, waardoor de ontwikkeltijd kort blijft,
- Toestaan dat meerdere personen gelijktijdig aan een ontwerp werken: de gegevens zodanig kunnen verdelen over het ontwerpteam dat meerdere personen aan een ontwerp kunnen werken zonder dat de kwaliteit van het ontwerp er onder lijdt,
- Gegevens hergebruiken: productgegevens hoeven nog maar n keer te worden vastgelegd,
- Groter bereik van gegevens: gebruik van productgegevens voor bijvoorbeeld gebruikers- en service-handleidingen,
- Overzicht en inzicht scheppen in grote aantallen gegevens.

Deze doelen dienen op zichzelf weer hogere, bedrijfskundige, doelen zoals een kortere time-to-market van producten en engineering-on-order.

10.2.7 Werking van het PDM-systeem

Deze doelen tracht men te vervullen met behulp van het zogenaamde PDM-systeem. Voordat wordt ingegaan op de functies van het PDM-systeem, zal er eerst uitgelegd worden wat de onderliggende techniek van een PDM-systeem is. Een PDM-systeem kan in bijna geen enkel geval in een bedrijf worden gecomplementeerd zonder dat systeemontwikkelaars de mogelijkheden van het systeem aanpassen aan de behoeften van het bedrijf. Daarom is het belangrijk om enig inzicht in de werking van een PDM-systeem te hebben. Een typische aanpassing is het maken van aansluitingen op legacy-programma's (legacy-programma's zijn oude programma's die niet meer worden gebruikt, maar waarvan de gegevens nog wel toegankelijk moeten zijn). Bibliotheken met normdelen en symbolen moeten vaak ook nog in het PDM-systeem worden ingevoegd. Technisch gezien is een PDM-systeem een toepassing die rond een DBS gebouwd is. Het belangrijkste kenmerk van een database in verband met PDM is het vermogen om de gegevens weer te geven zonder dat de gebruiker iets van de fysieke locatie van de gegevens hoeft te weten. Zo kan er een grote hoeveelheid gegevens worden gebruikt, verspreid over meerdere computers, terwijl de gebruiker het als n groot geheel ziet. De indeling van de gegevens gebeurt tevens op een voor de gebruiker logische wijze. Daarmee wordt bedoeld dat de gegevens bijvoorbeeld worden ingedeeld naar productstructuur of functie. Deze weergave (view) van de gegevens kan tevens worden aangepast aan de specifieke behoeften: met n druk op de knop kan worden overgeschakeld van indeling naar productstructuur naar indeling naar functie. Een dergelijke weergave van de aanwezige gegevens naar de behoefte van de gebruiker wordt een view genoemd. Zoals we eerder hebben gezien vereist iedere view zijn eigen dataschema voor de DB. We wijzen ook op paragraaf 4.2 (gegevensoverdracht); gegevensoverdracht kan gezien worden als het verschaffen van de gewenste view op het product. De dictionary van het PDM-systeem kent alle beheersvormen en de daarbij behorende productgegevensvormen. De beheersvormen worden beschreven door logica die relaties tussen de productgegevensvormen legt. Wanneer een productgegeven wordt opgeslagen in het PDM-systeem wordt het onder het beheer van een profiel geplaatst. Het profiel kent alle beheersvormen uit de dictionary. Dit profiel houdt bij wie het productgegeven heeft gemaakt en welke personen of groepen er wijzigingsrechten hebben. Het profiel bij een CAD-model van een onderdeel van een product weet bijvoorbeeld ook bij welke samenstelling het onderdeel hoort. De mogelijkheden van het PDM-systeem kunnen worden uitgebreid door de profielen voor de productgegevens aan te passen. Dankzij deze structurering kan het PDM-systeem productgegevens rangschikken voor een view, meta-data aan productgegevens verbinden en op trefwoorden de

database doorzoeken en andere functies uitvoeren, zoals in de volgende paragrafen wordt uitgelegd.

Hoofdstuk 11

Ontwerp optimalisatie

De inhoud van dit hoofdstuk is gebaseerd op het referentiewerk [5].

11.1 De fundamenteën van structurele optimalisatie

Met oog op het ontwikkelen en construeren van componenten en systemen rijst de vraag naar welke maatregelen moeten genomen worden om de kost te minimaliseren en de kwaliteit en betrouwbaarheid te verbeteren. Dit wordt beoogt in structurele optimalisatie, waar een rationele verwezenlijking van een structureel ontwerp wordt betracht dat het best is van alle mogelijke ontwerpen m.b.t. een vooraf gedefinieerd objectief en gegeven een set van geometrische en/of mechanische beperkingen.

In gewone *structurele analyse* problemen is het structureel ontwerp gegeven tesamen met de relevante eigenschappen van de te gebruiken materialen en de randvoorwaarden van de structuur. Tevens is zijn de belastingscondities gegeven (deterministisch of stochastisch). Met deze gegevens worden dan spanningstoestand, rek en uitwijking, eigenfrequenties, elastische instabiliteit e.d. berekend.

In *sensitiviteitsanalyse* wordt een deel van het ontwerp, materialen, of randvoorwaarden gevarieerd, en wordt het gevolg van deze veranderingen op de structurele respons berekend. In een conventionele ontwerp strategie worden een aantal herhaaldelijke veranderingen van de structurele parameters gevolgd door een analyse uitgevoerd tot een structuur is gevonden die het vereiste structurele gedrag vertoont binnen een redelijke kost.

Structurele optimalisatie duidt op een type ontwerp waar de structurele parameters worden onderverdeeld in ontwerp parameters (*Eng.: design variables*) en voorgedefinieerde parameters (*Eng.: preassigned parameters*), en

waar de optimale waarden van de ontwerp variabelen worden gezocht die een criterium genaamd de *objective function* (of ook kost functie) minimaliseren (of maximaliseren) terwijl voldaan wordt aan geometrische of mechanische beperkingen (*constraints* genoemd).

Afhankelijk van de manier waarop de design variables worden verondersteld af te hangen van de ruimtelijke variabelen kunnen structurele optimalisatie problemen onderverdeeld worden in:

- Continue of verdeelde parameter optimalisatie problemen,
- Discrete optimalisatie problemen

Gewoonlijk variëren de ontwerp variabelen van structurele elementen als bogen, platen, schalen, etc. continu over de lengte of domein van het element terwijl frames, trusses, e.d. gewoonlijk discreet worden geoptimaliseerd.

optimalisatie problemen kunnen geclassereerd worden in functie van hun ontwerp variabelen:

Constructieve layout Het bepalen van de meest geschikte layout is in principe enkel mogelijk door het bestuderen van alle bestaande types en door het vergelijken van de berekende optima (zie Figuur 11.1).

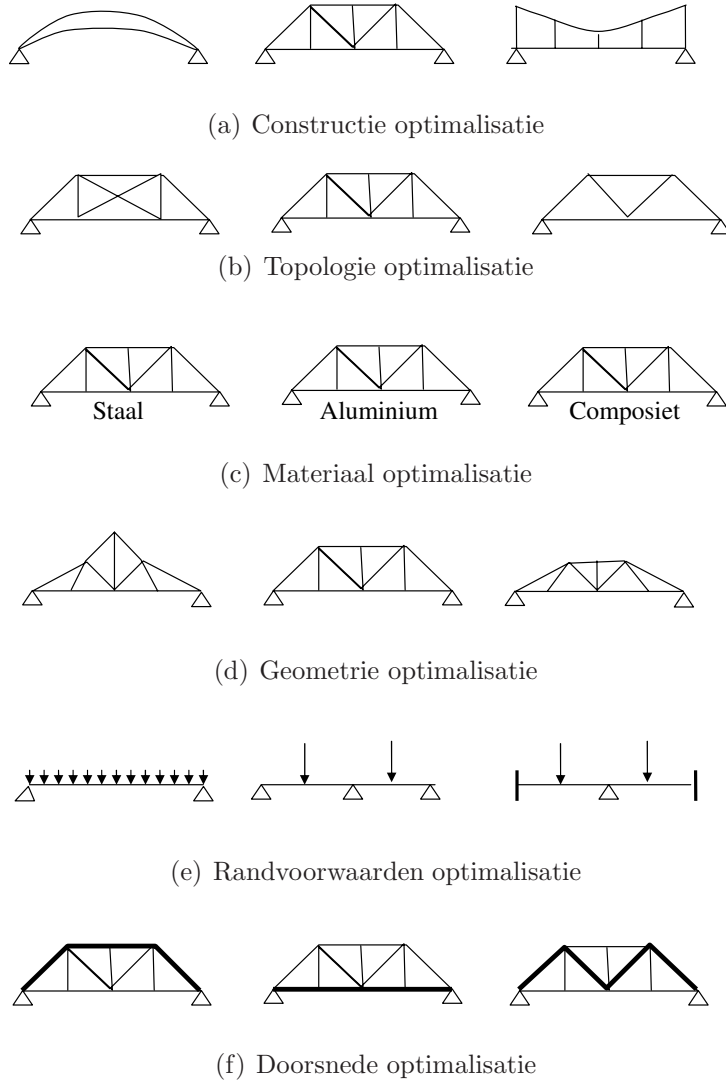
Topologie De topologie van de elementen in een structuur wordt vaak beschreven door parameters die kunnen gewijzigd worden in discrete stappen (vb. aantal trusses, aantal secties). Andere topologieën kunnen ook bekomen worden door elimineren van knooppunten en linken.

Materiaaleigenschappen Conventionele materialen bezitten variabelen zoals het specifiek gewicht, de Youngs modulus, mechanische sterkte eigenschappen, etc.

Geometrie en vorm De geometrie van frames of trusses wordt beschreven door nodale coördinaten of lengtes van stangen terwijl bij vlakke of gekromde oppervlakken (platen, schalen) de geometrie gegeven wordt door curvaturen en dikteverdelingen. Deze variabelen zijn gewoonlijk continu.

Randvoorwaarden De locatie, het aantal en het type randvoorwaarde of externe kracht kan gevarieerd worden om een effectiever ontwerp te bekomen (deze categorie is zowel discreet als continu).

cross-sectie Dit type van ontwerp variabelen wordt het vaakst gebruikt in optimalisatie problemen.



Figuur 11.1: Voorbeeld van de zes klassen van optimalisatie problemen gedefinieerd in functie van hun ontwerp variabelen.

De i -de design variabele wordt x_i genoteerd en alle n design variabelen vormen een vector \mathbf{x} die de design ruimte vormen (een Euclidische ruimte).

De constraints kunnen volgende vorm aannemen:

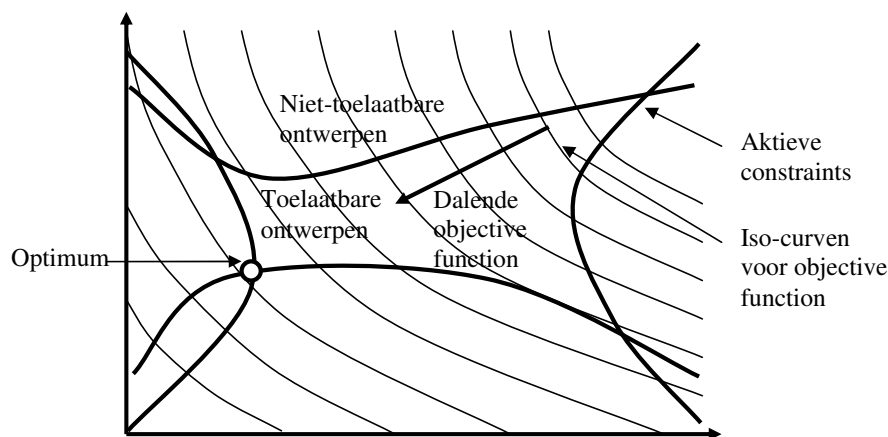
Geometrische constraints restricties expliciet opgelegd op de design variabelen door consideraties zoals fabricage limieten, fysische toepasbaarheid, esthetica, etc. Constraints van dit type zijn typisch ongelijkheden die boven en ondergrenzen van de design variabelen opleggen.

Mechanische constraints deze zijn gewoonlijk niet-lineair en impliciet in functie van de ontwerp variabelen.

Twee types constraints kunnen onderscheiden worden: gelijkheden $h_i(\mathbf{x}) = 0$ $i = 1, \dots, q$ en ongelijkheden $g_j(\mathbf{x}) \leq 0$ $j = 1, \dots, p$ (i.f.v. de respons van de structuur).

Elke ongelijkheid constraint wordt voorgesteld door een oppervlak in de design ruimte dat alle punten \mathbf{x} bevat waarvoor de gelijkheid constraint $g_j(\mathbf{x}) = 0$ met $j = 1, \dots, p$ voldaan is. Men onderscheid toelaatbare en niet-toelaatbare ontwerpen naargelang de designs in een deelruimte liggen gedefinieerd door de constraints of niet (*Eng. feasible en non-feasible constraints*). De actieve (of bindende) constraints t.o.v. een design variabele \mathbf{x} noemt men die constraints waarvoor $h_i(\mathbf{x}) = 0$ of $g_j(\mathbf{x}) = 0$ (d.w.z. waarvoor \mathbf{x} op de rand van het toelaatbaar gebied ligt).

De objective function (vaak ook kost functie of criterion function genoemd) moet uitgedrukt worden in functie van de design variabelen op een manier dat de waarde kan bepaald worden voor elk punt in de design ruimte (vb. de spanning, resonantie frequentie, flutter snelheid, etc.). De objective function is meestal een scalaire functie f van de design variabelen (vb. gewicht, kost, etc.).



Figuur 11.2: Twee-dimensionele design ruimte voor structurele optimalisatie.

Het optimaal ontwerp probleem kan geformuleerd worden als volgend

mathematisch probleem:

$$\min_{\mathbf{x} \in \mathbb{R}^n} \{f(\mathbf{x}) \mid \mathbf{h}(\mathbf{x}) = 0, \mathbf{g}(\mathbf{x}) \leq 0\} \quad (11.1)$$

met $f(\mathbf{x})$ de objective function, $\mathbf{g}(\mathbf{x})$ de vector met p ongelijkheidsconstraints en $\mathbf{h}(\mathbf{x})$ de vector met p gelijkheidsconstraints. Verder definieert met $X = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{h}(\mathbf{x}) = 0, \mathbf{g}(\mathbf{x}) \leq 0\}$ het toelaatbaar domein.

Hierbij wordt gezocht naar een globaal minimum $x^* \in X$, m.a.w.:

$$f(\mathbf{x}^*) \leq f(\mathbf{x}) \quad \forall \mathbf{x} \in X$$

Dit is moeilijker te verwezenlijken dan het vinden van een lokaal minimum, waarvoor: $x^* \in X$, m.a.w.

$$f(\mathbf{x}^*) \leq f(\mathbf{x}) \quad \forall \mathbf{x} \in X \cap U_\epsilon(\mathbf{x}^*)$$

waar $U_\epsilon(\mathbf{x}^*)$ een omgeving is van het punt \mathbf{x}^* .

Stelling 1 Voor (een lokale) oplossing van het optimalisatieprobleem 11.1 zonder constraints geldt: $\nabla f(\mathbf{x}^*) = 0$. (ZONDER BEWIJS)

Stelling 2 Indien voor een \mathbf{x}^* van het optimalisatieprobleem 11.1 zonder constraints geldt dat: $\mathbf{H}(\mathbf{x}^*) = \frac{\partial^2 f}{\partial x_i \partial x_j} \Big|_{\mathbf{x}^*}$ strikt positief definit is dan is \mathbf{x}^* een lokaal minimum. Deze \mathbf{H} noemt men de Hessiaan. (ZONDER BEWIJS)

Voor het constrained optimalisatie probleem worden Lagrange multipliatoren α_i en β_j gedefinieerd:

$$L(\mathbf{x}, \alpha, \beta) = f(\mathbf{x}) + \sum_{i=1}^q \alpha_i h_i(\mathbf{x}) + \sum_{j=1}^p \beta_j g_j(\mathbf{x}) \quad (11.2)$$

Stelling 3 Voor (een lokale) oplossing van het optimalisatieprobleem 11.1 met constraints geldt: $\nabla L(\mathbf{x}^*) = \nabla f(\mathbf{x}^*) + \sum_{i=1}^q \alpha_i^* \nabla h_i(\mathbf{x}^*) + \sum_{j=1}^p \beta_j^* \nabla g_j(\mathbf{x}^*) = 0$ en $h_i(\mathbf{x}^*) = 0$, $g_j(\mathbf{x}^*) \leq 0$ en $\beta_j^* \leq 0$. Dit noemt men de Kuhn-Tucker voorwaarden. (ZONDER BEWIJS)

Definitie 1 Een convex functie $f(\mathbf{x})$ is een continue functie waarvoor geldt: $\forall \mathbf{x}_0, \mathbf{x}_1 : f((1-\theta)\mathbf{x}_0 + \theta\mathbf{x}_1) \leq (1-\theta)f(\mathbf{x}_0) + \theta f(\mathbf{x}_1) \quad \forall \theta \in [0, 1]$. Een concave functie is een functie $f(\mathbf{x})$ waarvoor $-f(\mathbf{x})$ convex is.

Het optimalisatieprobleem (in Vergelijking 11.1) is convex indien de objective functie f convex is en de constraints concaaf zijn.

Stelling 4 *Voor een convex probleem zijn de Kuhn-Tucker voorwaarden voldoende voorwaarden. Bovendien is een lokale oplossing \mathbf{x}^* tevens globaal. (ZONDER BEWIJS)*

Een optimalisatie probleem vereist een aantal stappen, die gestructureerd zijn zoals weergegeven in Figuur 11.3 :

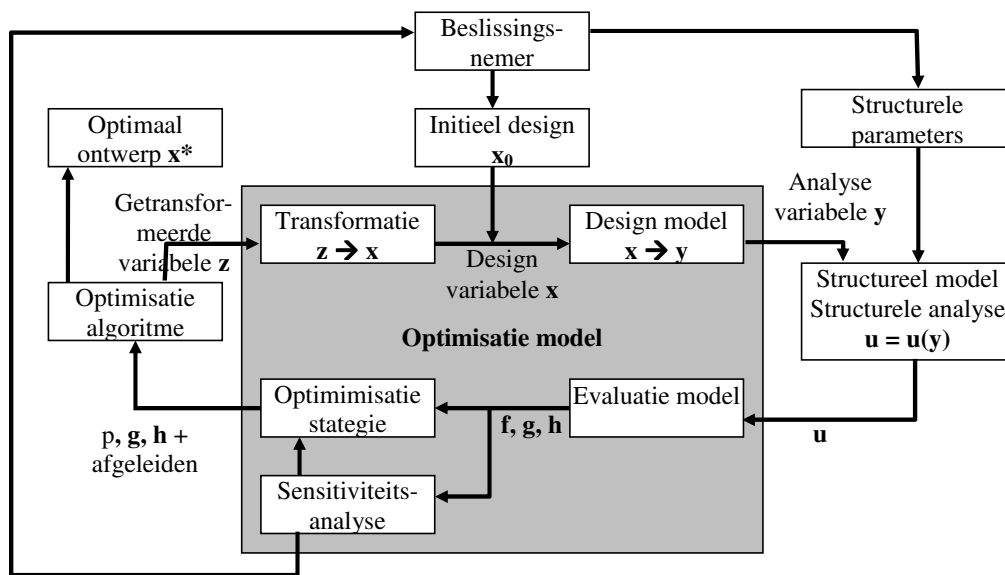
1. Een structureel model van het probleem moet opgesteld worden (mathematische beschrijving van het fysische gedrag van de structuur).
2. optimalisatie algorithmen zoals mathematische programmeren worden gebruikt om het niet-lineair constrained optimalisatie probleem op te lossen.
3. Het optimalisatie model slaat een brug tussen het structureel model en de optimalisatie algorithmen. In het design model beschrijft de relatie tussen de design variabelen en de analyse variabelen uit de structurele analyse. Door de transformatie van de design variabelen in transformatie variabelen kan de efficiëntie en de convergentiesnelheid van de optimalisatie verbeterd worden. Tevens worden d.m.v. het evaluatie model de waarden voor de objective functions en constraints berekend uit de waarden van de state variables. Indien mogelijk worden ook de gevoeligheden van de objective functions en de constraints berekend.

11.2 Optimalisatie algoritmen

De keuze van het meest geschikte optimalisatie algoritme is probleemafhankelijk. Enerzijds onderscheid men algoritmen van de nulde, eerste of tweede orde naargelang respectievelijk alleen functiewaarden, of ook eerste en tweede afgeleiden worden gebruikt (er wordt verondersteld dat de objective function en constraints twee keer continu differentieerbaar zijn).

De meerderheid van de optimalisatie algoritmen zijn iteratief (uitgaande van een startwaarde \mathbf{x}_0 bepaald men opeenvolgende verbeterde vectoren $\mathbf{x}_1, \mathbf{x}_2, \dots$). Indien zowel de objective functie als de constraints lineair zijn dan kunnen zogenaamde 'Lineaire Programmatie (LP) methodes' gebruikt worden. Kwadratisch programmeren (QP) behelst de minimalisatie problemen van een kwadratische objective function met lineaire constraints. Voor zowel LP als QP bestaan numeriek efficiënte algoritmen (vb. de Simplex methode).

Praktische structurele optimalisatie problemen zijn gewoonlijk echter zeer niet-lineair en bijgevolg moeten Niet-lineaire Programmatie methodes aangewend



Figuur 11.3: Structuur van het optimalisatie probleem.

worden. In de volgende paragrafen worden een aantal veelgebruikte optimalisatie algoritmes beschreven (voor een uitgebreide behandeling van het onderwerp zie [6, 7]).

11.2.1 Problemen zonder constraints

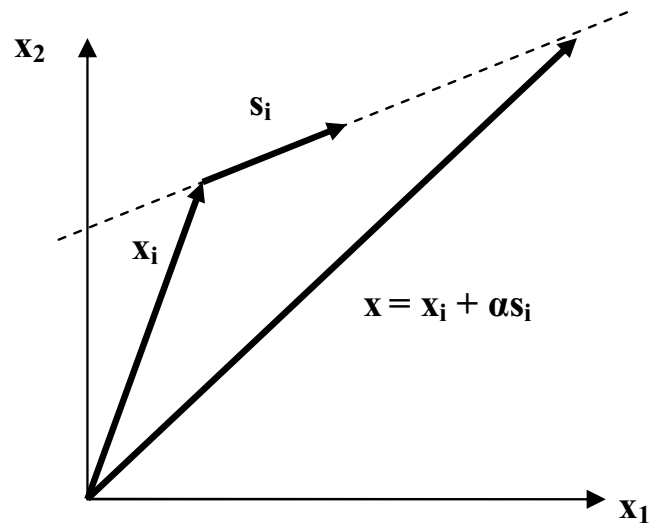
Lijnzoekmethodes

In elke iteratiestap wordt een nieuwe design variabele bekomen : $\mathbf{x}_{i+1} = \mathbf{x}_i + \alpha_i \mathbf{s}_i$.

In het geval van een variable staplengte α : $\mathbf{x}(\alpha) = \mathbf{x}_i + \alpha \mathbf{s}_i$. Bijgevolg wordt een rechte lijn gevolgd in de design ruimte. Dit reduceert het probleem to het zoeken van een ééndimensionaal minimum α_i in een gegeven richting s_i zoals geïllustreerd in Figuur 11.4 (deze klasse van technieken noemt men bijgevolg lijnzoekmethodes).

Volgende lijnzoekmethodes kunnen o.a. gebruikt worden :

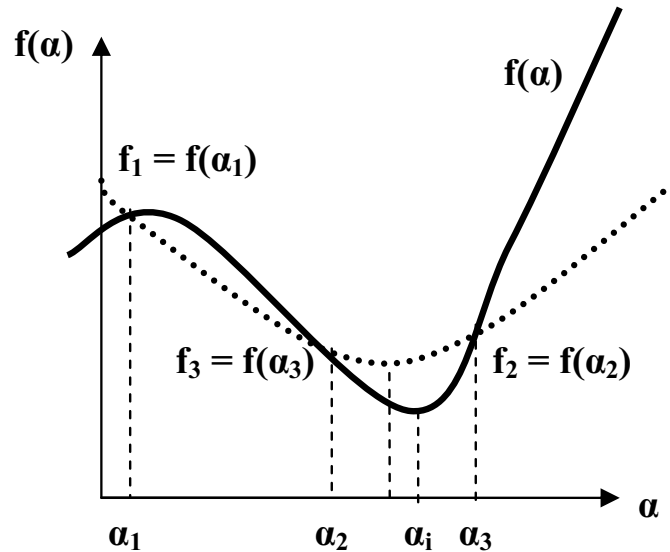
1. **Lagrange interpolatie** (zie Figuur 11.5). Stel α_1, α_2 en α_3 drie punten waarvan de functiewaarden f_1, f_2 en f_3 geëvalueerd worden. Men kan dan de kwadratische polynoom $P_2(\alpha)$ bepalen die deze drie punten interpoleert: $P_2(\alpha) = \sum_{i=1}^3 L_i(\alpha) f_i$ met $L_i(\alpha) = \prod_{j=1, j \neq i}^3 \frac{(\alpha - \alpha_j)}{(\alpha_i - \alpha_j)}$ de



Figuur 11.4: Grafische voorstelling van de lijnzoekmethode in twee dimensies.

zogenaamde Lagrange veeltermen. Daarna itereert men de procedure met α_2, α_3 en α_4 (waarbij α_4 het minimum van de veelterm P_2).

2. **Hermite interpolatie.** Dezelfde procedure als bij de Lagrange interpolatie kan ook worden uitgevoerd d.m.v. interpolatie met een kubische polynoom i.p.v. een kwadratische. Deze methode wordt toegepast wanneer ook afgeleide waarden van de objective functie gekend zijn (m.a.w. twee functiewaarden en de waarde van de eerste afgeleiden in die punten zijn gegeven).
3. **Methode van de intervalreductie.** Stel gegeven een interval α_0, α_1 waarin de functie f unimodaal is (slechts 'één minimum heeft). In dit interval evalueren we de functie f in twee punten: $f(\alpha_2)$ en $f(\alpha_3)$. Indien $f(\alpha_2) > f(\alpha_3)$ dan behouden we interval $[\alpha_2, \alpha_1]$ en indien $f(\alpha_2) < f(\alpha_3)$ dan behouden we interval $[\alpha_0, \alpha_3]$. In het gereduceerde interval kiezen we opnieuw twee punten. Deze procedure wordt geïtereerd tot convergentie wordt bereikt.



Figuur 11.5: Grafische voorstelling van de Lagrange Interpolatie methode.

Methode van de geconjugeerde richtingen (methode van POWELL)

Om het iteratiepunt \mathbf{x}_{i+1} te berekenen wordt informatie over de voorafgaande stappen $\mathbf{x}_1, \dots, \mathbf{x}_i$ gebruikt. Een van de methoden maakt gebruik van geconjugeerde richtingen \mathbf{s}_j en \mathbf{s}_k die worden gedefinieerd door: $\mathbf{s}_j^T \mathbf{H} \mathbf{s}_k = 0$ ($j \neq k$), waarbij \mathbf{H} de Hessiaan zoals voordien gedefinieerd.

Bij de methode van Powell wordt een benadering \mathbf{A} voor de Hessiaan gebruikt. Startende van een initieel punt \mathbf{x}_0 worden n ééndimensionale minimalisatie iteratiestappen uitgevoerd langst n lineair onafhankelijke zoekrichtingen $\mathbf{s}_0, \dots, \mathbf{s}_{n-1}$ (vb. eenheidsvectoren):

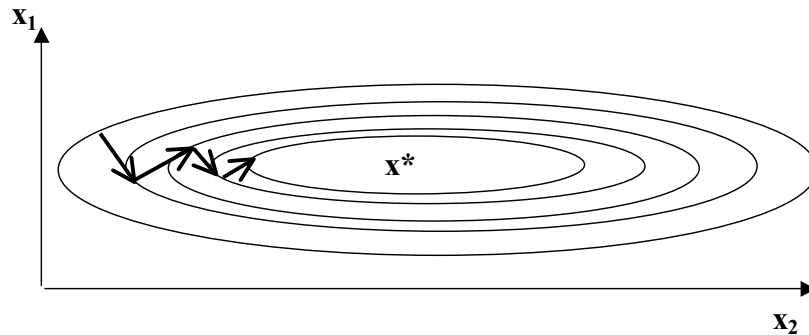
$$\mathbf{x}(\alpha) = \mathbf{x}_i + \alpha \mathbf{s}_i \quad (\text{voor } i = 1, \dots, n) \quad (11.3)$$

De zoekcyclus wordt vervolledigd door een minimalisatie in de richting $\mathbf{s}_n = \mathbf{x}_n - \mathbf{s}_{n-2}$ die leidt tot het punt \mathbf{x}_{n+1} . In de volgende zoekcyclus wordt de eerste zoekrichting \mathbf{s}_0 geëlimineerd en wordt de procedure herhaald op de overblijvende richtingen.

Steepest descent methode

In de Steepest descent (of stijlste daling) methode wordt de als zoekrichting \mathbf{s}_i de richting van de stijlste daling genomen :

$$\mathbf{s}_i = -\nabla f(\mathbf{x}_i) \quad (11.4)$$



Figuur 11.6: Grafische voorstelling van de achtereenvolgende stappen in de Steepest Descent methode (bij een ellipsvormige objectieffunctie waarvan één van de afmetingen veel groter is dan de andere is de convergentie zeer traag).

Methode van de geconjugeerde gradiënten (Fletcher-Reeves methode)

In de Fletcher-Reeves methode wordt eerst een minimum gezocht volgens de steepest descent methode in Vergelijking 11.4 $\mathbf{s}_1 = -\nabla f(\mathbf{x}_1)$ om \mathbf{x}_2 te bekomen. Vervolgens worden nieuwe zoekrichtingen gegenereerd :

$$\mathbf{s}_{i+1} = -\nabla f(\mathbf{x}_{i+1}) + \frac{|\nabla f(\mathbf{x}_{i+1})|^2}{|\nabla f(\mathbf{x}_i)|^2} \mathbf{s}_i \quad (11.5)$$

Speciale Quasi-Newton methode

Bij de quasi-Newton methode wordt het algemeen niet-lineair optimalisatie probleem omgezet in een QP probleem (gebruik makende van de KT voorwaarden) :

$$\min_{\mathbf{x} \in \mathbb{R}} \mathbf{x}^T \mathbf{H} \mathbf{x} + \mathbf{c}^T \mathbf{x} + b \quad (11.6)$$

In deze vergelijking is \mathbf{H} de positief definitie Hessiaan matrix. De optimale oplossing \mathbf{x}^* kan gevonden worden door nodige voorwaarde

$$\nabla f(\mathbf{x}) = 0 \Rightarrow \mathbf{x}^* = -\mathbf{H}^{-1}\mathbf{c} \quad (11.7)$$

In Newton methodes wordt de Hessiaan \mathbf{H} numeriek berekend en wordt dan m.b.v. vorige vergelijking de optimale oplossing bekomen (de berekening van de Hessiaan is echter zeer reken-intensief). Bij quasi-Newton methodes wordt de Hessiaan benaderd door een updating algoritme. Eén van deze updating formules is de Broyden-Fletcher-Goldfarb-Shanno (BFGS) formule:

$$\mathbf{H}_k = \mathbf{H}_{k-1} + \frac{\mathbf{q}_{k-1}\mathbf{q}_{k-1}^T}{\mathbf{q}_{k-1}^T\mathbf{s}_{k-1}} - \frac{\mathbf{H}_{k-1}^T\mathbf{s}_{k-1}\mathbf{s}_{k-1}^T\mathbf{H}_{k-1}}{(\mathbf{s}_{k-1}^*\mathbf{H}_{k-1}\mathbf{s}_{k-1})} \quad (11.8)$$

met $q_{k-1} = \mathbf{x}_k - \mathbf{x}_{k-1}$ en $\mathbf{s}_{k-1} = \nabla f(\mathbf{x}_k) - \nabla f(\mathbf{x}_{k-1})$. Als startwaarde kan elke positief definitie matrix \mathbf{H}_0 gebruikt worden (vb. de éénheidsmatrix). In elke iteratiestap worden volgende zoekrichtingen bepaald:

$$\mathbf{s}_k = -\mathbf{H}_k^{-1}\nabla f(\mathbf{x}_k) \quad (11.9)$$

Om de inversie van de Hessiaan matrices \mathbf{H}_k te vermijden kan ook een updating schema voor de inverse Hessiaan matrices $\mathbf{G}_k = \mathbf{H}_k^{-1}$ opgesteld worden. Een voorbeeld hiervan is de Davidson-Fletcher-Powell (DFP) formule:

$$\mathbf{G}_k = \mathbf{G}_{k-1} + \frac{\mathbf{q}_{k-1}\mathbf{q}_{k-1}^T}{\mathbf{q}_{k-1}^T\mathbf{s}_{k-1}} - \frac{\mathbf{G}_{k-1}^T\mathbf{q}_{k-1}\mathbf{q}_{k-1}^T\mathbf{G}_{k-1}}{(\mathbf{q}_{k-1}^*\mathbf{G}_{k-1}\mathbf{q}_{k-1})} \quad (11.10)$$

Heuristische methodes

Random zoekmethodes

Hierbij wordt design ruimte verkend op een random manier. De methode is zeer tijdsroevend maar kan goede resultaten geven wanneer de objective functie zeer complex is.

Genetische algoritmes

Eén van de tegenwoordig zeer populaire heuristische methodes zijn de zogenaamde 'Genetische Algoritmen (GA)', die gebaseerd zijn op de mechanismen van de natuurlijke selectie om een functie te optimaliseren. Het grote voordeel van deze methodes is dat een globaal optimum wordt bekomen waar klassieke optimalisatie methodes slechts naar een lokaal optimum convergeren (de klasse van toepasbare objective functies is dan ook veel groter en bevat exotische multi-modale niet-continue functies). Het belangrijkste nadeel t.o.v.

traditionele methodes is de vereiste rekentijd (die weliswaar kleiner is dan bij random zoekmethodes).

De volgende stappen worden gebruikt in een genetisch algoritme:

1. De ontwerpen \mathbf{x} in de design ruimte X worden gecodeerd in bijvoorbeeld binaire strings.
2. Er worden populaties gevormd van ontwerpen.
3. Binnen een populatie worden in elke iteratie een aantal operatoren uitgevoerd:
 - (a) Reproductie: de individuele strings worden gecopieerd naar een nieuwe generatie volgens de waarde van de objective functie (strings met een hoger objective functie waarde hebben een grotere kans om geselecteerd te worden voor reproductie).
 - (b) Crossover: twee deelstrings van verschillende ontwerpen worden samen genomen.
 - (c) Mutatie: een random verandering van een van de cijfers in de string (echter met zeer kleine waarschijnlijkheid).
4. Op een iteratieve wijze worden nieuwe populaties gevormd.

Meer informatie omtrent deze methodes kan gevonden worden in [8].

11.2.2 Problemen met constraints

Reductie tot problemen zonder constraints

Het optimalisatie probleem in Vergelijking 11.1 dat p ongelijkheidsconstraints $g_j(\mathbf{x}) \leq 0$ bevat wordt geherformuleerd tot:

$$\Phi_i(\mathbf{x}, R_i) = f(\mathbf{x}) + R_i \sum_{j=1}^p G[g_j(\mathbf{x})] \quad (i = 1, 2, 3 \dots) \quad (11.11)$$

hierbij is $G[g_j(\mathbf{x})]$ de penalty functie en R_i de penalty parameters. De functie G wordt gekozen zodat gedurende de minimizatie voor een reeks waarden R_i de oplossing convergeert naar deze van het originele probleem met constraints. Enkele veelgebruikte voorbeelden:

- Methode van de uitwendige penalty function : $G[g_j(\mathbf{x})] = (\max(0, g_j(\mathbf{x})))^2$ en Φ_i wordt geminimaliseerd voor een stijgende set van waarden R_i (vb. $R_1 = 10^{-3}$, $R_2 = 10^{-2}$, $R_3 = 10^{-1}$, ...). De naam van deze methode

komt van het feit dat de geherformuleerde objective functie gelijk is aan de originele voor de toelaatbare ruimte en wordt gepenaliseerd met niet-negatieve termen voor punten in de niet-toelaatbare ruimte.

- Methode van de inwendige penalty function : $G[g_j(\mathbf{x})] = \left(\frac{-1}{g_j(\mathbf{x})}\right)$.

Algemene methodes

1. Sequentiële linearisatie procedure: Door het introduceren van boven en ondergrenzen voor alle designvariabelen wordt het toepassingsdomein uitgebreid tot problemen waar de oplossingen niet noodzakelijk op de doorsnede van twee constraints liggen, maar op een algemener gekromd hypervlak. De objective functie en de constraints van het niet-lineair probleem worden geëxpandeerd in een Taylor-reeks in de omgeving van een punt \mathbf{x}^k :

$$f(\mathbf{x}^k + \delta\mathbf{x}) = f(\mathbf{x}^k) + \nabla f(\mathbf{x}^k)\delta\mathbf{x} \quad (11.12)$$

en analoog voor h_i (voor $i = 1, \dots, q$) en g_j (voor $j = 1, \dots, p$). Dit gelineariseerd probleem kan dan met klassieke lineaire programmatie methodes opgelost worden (vb. de Simplex methode)

2. Augmented Lagrange functie procedure: hierbij wordt de Lagrange functie zoals gedefinieerd in Vergelijking 11.2 gebruikt om de constraints te vervatten in de objective functie. Omdat het rechtstreeks gebruik van de Lagrange functie L niet noodzakelijk een oplossing garandeert (de Kuhn-Tucker voorwaarden zijn niet voldoende voor algemene optimalisatie problemen, alleen voor convexe problemen) wordt de Lagrange functie uitgebreid:

$$L_a(\mathbf{x}, \alpha, \beta, \mathbf{w}) = L(\mathbf{x}, \alpha, \beta) + w_1P_1 + w_2P_2 + w_3P_3 \quad (11.13)$$

3. Sequential quadratic programming (SQP) Deze methode gebruikt een sequentiële linearisering (i.e. een eerste orde Taylor benadering rond het punt \mathbf{x}) en kwadratische benadering van een niet-lineair probleem. Bij elke iteratie wordt een benadering gemaakt van de Lagrange functie (zie vergelijking 11.2) gebruik makende van een quasi-Newton methode.
4. Method of generalized reduced gradients (GRG)

11.3 Optimalisatie strategieën

Om verschillende types optimalisatie problemen (zoals vorm en topologie optimalisatie) moeten specifieke optimalisatie strategieën geïntegreerd worden

in de aanpak van het optimalisatie probleem zoals weergegeven in de Figuur 11.3. Deze strategieën zijn onderdelen van het optimalisatie model die het probleem transfereren naar zodat het opgelost kan worden m.b.v. een scalaire parameter optimalisatie methode.

11.3.1 Multicriteria optimaliteit

Voor multicriteria optimalisatie worden simultaan twee of meer criteria gebruikt (dit noemt men ook vector optimalisatie of multi-objective optimalisatie). Deze problemen komen in de praktijk zeer vaak voor: bv. optimalisatie van de kost en de sterkte. In analogie met scalaire optimalisatie probleem uit Vergelijking 11.1 definieert men het Vector optimalisatie Probleem (VOP):

$$\min_{\mathbf{x} \in \mathbb{R}^n} \{\mathbf{f}(\mathbf{x}) \mid \mathbf{h}(\mathbf{x}) = 0, \mathbf{g}(\mathbf{x}) \leq 0\} \quad (11.14)$$

waar $\mathbf{f}(\mathbf{x}) = \begin{pmatrix} f_1(\mathbf{x}) \\ \vdots \\ f_m(\mathbf{x}) \end{pmatrix}$ de vector objective functie is.

Gewoonlijk is er bij vector optimalisatie een afweging tussen verschillende criteria: de verbetering van één of meer criteria kan ten koste van de achteruitgang van andere criteria gaan. Een alternatief hiervoor is het gebruik van de zogenaamde Pareto-optimaliteit:

Definitie 2 Een vector $\mathbf{x}^* \in X$ wordt Pareto-optimaal genoemd als en slechts als er geen vector $\mathbf{x} \in X$ bestaat waarvoor geldt:

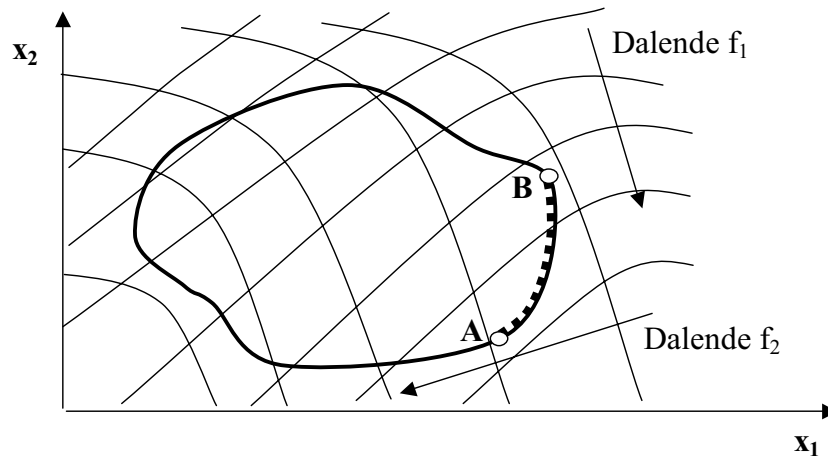
$$\begin{cases} f_j(\mathbf{x}) \leq f_j(\mathbf{x}^*), & \text{voor alle } j \in \{1, \dots, m\} \\ f_j(\mathbf{x}) < f_j(\mathbf{x}^*), & \text{voor tenminste één } j \in \{1, \dots, m\} \end{cases}$$

Een multicriteria optimalisatie probleem kan een oneindig aantal Pareto optima hebben (zie bv. Figuur 11.7). De ontwerper moet het optimaal ontwerp binnen deze Pareto optima kiezen (vb. aan de hand van de fabriceerbaarheid).

Een andere strategie bestaat erin het multiobjective optimalisatie probleem (uit Vergelijking 11.14) om te zetten naar een scalair probleem (Vergelijking 11.1), i.e. men lost volgend substitueert probleem op: $\min_{\mathbf{x} \in \mathbb{R}^n} p(\mathbf{f}(\mathbf{x}))$, waar men p de *preference functie* of gescalariseerde objective functie noemt. Enkele mogelijke preference functies zijn:

Som van de gewogen objectieven

$$p(\mathbf{f}(\mathbf{x})) = \sum_{j=1}^m w_j f_j(\mathbf{x}) \text{ met } \mathbf{x} \in \mathbb{R}^n \quad (11.15)$$



Figuur 11.7: Illustratie van de ligging van de Pareto optima (boog AB).

met de wegingsfactoren w_j die voldoen aan $0 \leq w_j \leq 1$ en $\sum_{j=1}^m w_j = 1$ en die gekozen worden door de ontwerper.

Som van de afstandsfuncties

$$p(\mathbf{f}(\mathbf{x})) = \left(\sum_{j=1}^m |f_j(\mathbf{x}) - \bar{y}_j|^r \right)^{\frac{1}{r}} \quad \text{met } \mathbf{x} \in \mathbb{R}^n \quad (11.16)$$

Hierbij is $\bar{\mathbf{y}} \in \mathbb{R}^n$ een vector met gegeven target waarden voor de criteria f_j ($j = 1, \dots, m$). Zowel deze vector als de waarde r van de exponent in de afstandsfunctie wordt door de ontwerper gekozen.

Trade-off methode

$$p(\mathbf{f}(\mathbf{x})) = f_1(\mathbf{x}) \quad , \quad f_j(\mathbf{x}) \leq \bar{y}_j \quad j = 2, \dots, m \quad \mathbf{x} \in \mathbb{R}^n \quad (11.17)$$

met f_1 als het hoofdobjectief en f_2, \dots, f_m als secundaire objectieven (constraints). De waarden \bar{y}_j worden gekozen door de ontwerper.

Min-max formulatie

$$p(\mathbf{f}(\mathbf{x})) = \max_j (z_j(\mathbf{x})) \quad \text{met } \mathbf{x} \in \mathbb{R}^n \quad (11.18)$$

en met $z_j(\mathbf{x}) = \frac{f_j(\mathbf{x}) - \bar{f}_j}{\bar{f}_j}$. De door de ontwerper gespecificeerde waarden \bar{f}_j zijn positief.

11.3.2 Vorm optimalisatie

Onder de term vorm optimalisatie verstaat men het optimaal 'shapen' van componenten door simultaan gegeven vereisten te beschouwen. Dit wordt gedaan door zogenaamde '*shape functions*' te definiëren die de vorm van de structuur beschrijven. In het algemeen leidt het vorm optimalisatie probleem tot het formuleren van *objective functionalen* F en constraint operatoren G . Gezien de vorm continue wordt gevarieerd gedurende de optimalisatie is een nieuwe discretisatie nodig tijdens dit proces. Men onderscheid indirecte en direct vorm optimalisatie methodes.

Indirecte methodes Deze methodes bevatten volgende twee stappen:

1. Afleiden van de optimaliteitscondities als de noodzakelijke condities voor het optimaal ontwerp.
2. Verwezenlijking van de optimaliteitscondities door geschikte oplossingsmethodes.

Beschouw een structuur met rand Γ waarvan het gebied A beschreven wordt door de curvi-lineaire coördinaten ξ^α . De optimale shape functie \mathbf{R} met componenten R_j ($j = 1, 2, 3$) zal worden bepaald. Hierbij wordt het vorm optimalisatie probleem beperkt tot problemen waarbij zowel de constraint als de objective functie uitgedrukt kunnen worden onder de vorm van integralen:

$$\text{Min}F = \min \int_A f(\xi_\alpha, R^j, R_{,\alpha}^j) dA \quad (j = 1, 2, 3) \quad (11.19)$$

waarbij $R_{,\alpha}^j = \frac{\partial R^j}{\partial \xi^\alpha}$ en waar

$$\int_A f_k(\xi_\alpha, R^j, R_{,\alpha}^j) dA = 0 \quad (k = 1, \dots, m) \quad (11.20)$$

$$\int_A f_l(\xi_\alpha, R^j, R_{,\alpha}^j) dA \leq 0 \quad (l = 1, \dots, r) \quad (11.21)$$

verondersteld worden gegeven te zijn.

Om de nodige condities voor het probleem af te leiden worden de ongelijkheidsoperatoren in Vergelijking 11.22 getransformeerd in gelijkheidsoperatoren met behulp van 'slack' variabelen η_l :

$$\int_A f_l(\xi_\alpha, R^j, R_{,\alpha}^j) dA + \eta_l^2 = 0 \quad (l = 1, \dots, r) \quad (11.22)$$

Gebruik makende van Lagrange vermenigvuldigers λ_k , λ_l wordt de Lagrange functionaal I gedefinieerd:

$$I = \int_A \Phi(\xi^\alpha, R^j, R_{,\alpha}^j, \lambda_k, \lambda_l) + \sum_{l=1}^r \lambda_l \eta_l^2 \quad (11.23)$$

waarbij:

$$\Phi(\xi^\alpha, R^j, R_{,\alpha}^j, \lambda_k, \lambda_l) = f(\xi_\alpha, R^j, R_{,\alpha}^j) + \sum_{k=1}^m \lambda_k f_k(\xi_\alpha, R^j, R_{,\alpha}^j) + \sum_{l=1}^r \lambda_l f_l(\xi_\alpha, R^j, R_{,\alpha}^j) \quad (11.24)$$

Met behulp van de Euler vergelijkingen voor het volgende variatie probleem: $\min I(\xi^\alpha, R^j, R_{,\alpha}^j, \lambda_k, \lambda_l, \eta_l)$ kan het vorm optimalisatie probleem opgelost worden. Enerzijds moet $\delta I = 0$ voor een arbitraire variatie R^j en dus: $2\lambda_l \eta_l = 0$ voor $l = 1, \dots, r$. Anderzijds kan volgend Lemma uit de variatierekening gebruikt worden:

$$\frac{\partial \Phi}{\partial R^j} - \left(\frac{\partial \Phi}{\partial R_{,\alpha}^j} \right)_\alpha = 0 \quad \int n^\alpha \frac{\partial \Phi}{\partial R_{,\alpha}^j} \delta R^j d\Gamma = 0 \quad (11.25)$$

Hieruit kunnen de onbekenden R^j , λ_k en λ_l bepaald worden.

Directe methodes Bij de directe vorm optimalisatie methodes worden de problemen getransformeerd naar een parameter optimalisatie probleem dat dan opgelost wordt door één van de optimalisatie methodes uit Paragraaf 11.2.

Men bepaald een optimale shape functie R^* voor welke de objectief functie F een minimum bereikt: $\min_{R \in \Gamma_2} F(\mathbf{R}) \rightarrow F(\mathbf{R}^*)$ met Γ_2 de verzameling van alle shape functies.

Het toelaatbaar variationeel domein wordt gedefinieerd door de constraint operatoren H_j en G_j :

$$\begin{cases} H_i \mathbf{R} = \varphi_i, & (i = 1, \dots, q) \\ G_j \mathbf{R} = \chi_j, & (j = 1, \dots, p) \end{cases} \quad (11.26)$$

De ongekende functies \mathbf{R} worden benaderd door geschikte vorm benaderingsfuncties $\tilde{\mathbf{R}}$ (*Eng.: shape approximation functions*).

De keuze van geschikte benaderingsfuncties voor optimale geometrieën is probleemafhankelijk. De gekozen functie moet de vorm zo goed mogelijk benaderen wat een groot aantal te optimaliseren parameters met

zich meebrengt. Door een aangepaste keuze kan het aantal parameters nodig om de vorm met een bepaalde nauwkeurigheid te benaderen gereduceerd worden.

Enkele van de belangrijkste benaderingsfuncties voor modelleren van geometrische vormen van componenten zijn:

1. Algemene polynoom functies: beschrijft de afhankelijkheid van de vorm functie \tilde{R}^j van de lokale coördinaat ξ als volgt:

$$\tilde{R}^j(\xi, \mathbf{x}) = x_1 + x_2\xi + x_3\xi^2 + \dots + x_n\xi^{n-1} \quad (11.27)$$

Voor een stijgende graad van de polynoom treedt een oscillerend effect op en wordt de conditionering van het probleem zeer slecht. Om de numerieke conditie te verbeteren kunnen Chebychev functies gebruikt worden.

2. Chebychev polynomen: deze polynomen T_k worden op recursieve wijze gedefinieerd door:

$$T_0 = 1 \quad T_1 = 2\frac{\xi - \xi_1}{\xi_u - \xi_1} - 1 \quad (11.28)$$

$$T_k = 2T_1T_{k-1} - T_{k-2} \quad \text{voor } k \geq 2 \quad (11.29)$$

met ξ de lokale coördinaat in het interval $[\xi_1, \xi_u]$. De benaderende vorm functie wordt dan gegeven door:

$$\tilde{R}^j(\xi, \mathbf{x}) = x_1T_0(\xi) + x_2T_1(\xi) + x_3T_2(\xi) + \dots + x_nT_{n-1}(\xi) \quad (11.30)$$

3. B-spline functies: dit zijn niet-lineaire functies gedefinieerd door $n + 1$ zogenaamde controle punten (merk op dat deze punten niet op de B-spline curve liggen met uitzondering van het begin en eindpunt). De B-spline curve wordt gedefinieerd door:

$$\mathbf{r}(\xi) = \sum_{i=0}^n \mathbf{p}_i B_{i,k}(\xi) \quad (11.31)$$

met \mathbf{p}_i de vector van het i -de controle punt en $B_{i,k}(\xi)$ de B-spline basis functies van de graad k . Deze basis functies worden net als de Chebychev polynomen op recursieve wijze gedefinieerd:

$$B_{i,1}(\xi) = \begin{cases} 1, & \text{voor } t_1 \leq \xi \leq t_{i+1} \\ 0, & \text{voor alle andere } \xi \end{cases} \quad (11.32)$$

$$B_{i,k}(\xi) = \frac{\xi - t_i}{t_{i+k-1} - t_i} B_{i,k-1}(\xi) + \frac{t_{i+k} - \xi}{t_{i+k} - t_{i+1}} B_{i+1,k-1}(\xi) \quad (11.33)$$

De punten t_i noemt men knooppunten.

4. Bezier curves worden analoog als B-spline curves met $n+1$ controle punten gedefinieerd:

$$\mathbf{r}(\xi) = \sum_{i=0}^n \mathbf{p}_i B_{i,k}(\xi) \quad (11.34)$$

In tegenstelling tot B-spline curves varieert de parameter ξ bij Bezier curves in het interval $[0, 1]$. De basis functies $B_{i,k}$ worden gegeven door:

$$B_{i,k}(\xi) = \frac{k!}{i!(k-i)!} \xi^i (1-\xi)^{k-i} \quad (11.35)$$

5. Ellips functies met variabele exponenten kunnen gebruikt worden om de vorm van vaten te vinden. Deze worden gegeven door:

$$\left(\frac{x^1}{a}\right)^{\kappa_1} + \left(\frac{x^2}{b}\right)^{\kappa_2} = 1 \quad (11.36)$$

met vorm parameters κ_1 en κ_2 en de semi-assen a , b . De parametrische voorstelling (met parameter ϕ) van de ellips luidt dan: $x^1 = a(\sin \phi)^{2/\kappa_1}$, $x^2 = b(\cos \phi)^{2/\kappa_2}$.

11.4 Voorbeeld van een optimaal ontwerp

11.4.1 Probleemstelling

Gegeven

Een stangenstructuur zoals getoond in Figuur 11.8 bestaat uit 13 stangen met een doorsnede A_i $i = 1, \dots, 13$ en 10 knooppunten. Een verticale kracht van $F = 100kN$ werkt in op punt 3.

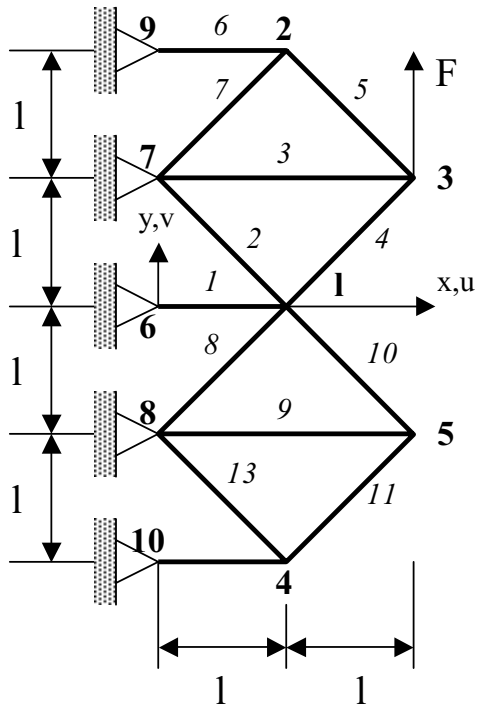
Gevraagd

Bepaal de doorsnedes A_i zodat het gewicht van de structuur minimaal is. Daarbij mogen de aanwezige spanningen in de stangen $150MPa$ in druk en $-100MPa$ in trek niet overtreffen. Verder is gegeven dat $E = 2.110^5 MPa$ en $l = 1.0m$.

Oplossing

De relatie tussen de knooppunt krachten en de knooppunt verplaatsingen wordt bepaald uit de evenwichtsvoorwaarden. Vb. voor knooppunt 2 (zie Figuur 11.9):

$$\begin{cases} F_3 = S_6 + \frac{1}{2}\sqrt{2}S_7 - \frac{1}{2}\sqrt{2}S_5 \\ F_4 = \frac{1}{2}\sqrt{2}S_7 + \frac{1}{2}\sqrt{2}S_5 \end{cases} \quad (11.37)$$



Figuur 11.8: Vlakke stangen structuur (de nummers van de 13 knooppunten zijn aangegeven in vet lettertype, die van de 10 elementen in schuin lettertype).

Verder geldt de elasticiteitswet:

$$\Delta l_6 = \frac{l}{EA_6} S_6 \quad \Delta l_7 = \frac{\sqrt{2}l}{EA_7} S_7 \quad \Delta l_5 = \frac{\sqrt{2}l}{EA_5} S_5 \quad (11.38)$$

En de kinematische wet: $\Delta l_6 = u_2$, $\Delta l_7 = 1/2\sqrt{2}u_2$, $\Delta l_5 = -1/2\sqrt{2}u_2$, $\Delta l_6 = 0$, $\Delta l_7 = 1/2\sqrt{2}v_2$, $\Delta l_5 = 1/2\sqrt{2}v_2$,

Substitutie van deze wetten in de evenwichtsvoorwaarden geeft:

$$\begin{pmatrix} F_3 \\ F_4 \end{pmatrix} = \frac{E}{2\sqrt{2}l} \begin{pmatrix} A_5 + 2\sqrt{2}A_6 + A + 7 & -A_5 + A_7 \\ -A_5 + A_7 & A_5 + A_7 \end{pmatrix} \begin{pmatrix} u_2 \\ v_2 \end{pmatrix} \quad (11.39)$$

Voor andere knooppunten kunnen analoge relaties bepaald worden, zodat

de relatie $\mathbf{f} = \mathbf{K}\mathbf{v}$ bekomen wordt, met

$$\mathbf{K} = \frac{E}{2\sqrt{2}l} \begin{pmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} & \mathbf{K}_{13} & \mathbf{K}_{14} & \mathbf{K}_{15} \\ \mathbf{K}_{21} & \mathbf{K}_{22} & \mathbf{K}_{23} & \mathbf{K}_{24} & \mathbf{K}_{25} \\ \mathbf{K}_{31} & \mathbf{K}_{32} & \mathbf{K}_{33} & \mathbf{K}_{34} & \mathbf{K}_{35} \\ \mathbf{K}_{41} & \mathbf{K}_{42} & \mathbf{K}_{43} & \mathbf{K}_{44} & \mathbf{K}_{45} \\ \mathbf{K}_{51} & \mathbf{K}_{52} & \mathbf{K}_{53} & \mathbf{K}_{54} & \mathbf{K}_{55} \end{pmatrix} \quad (11.40)$$

$$\text{met } \mathbf{K}_{11} = \begin{pmatrix} 2\sqrt{2}A_1 + A_2 + A_4 + A_8 + A_{10} & -A_2 + A_4 + A_8 - A_{10} \\ -A_2 + A_4 + A_8 - A_{10} & A_2 + A_4 + A_8 + A_{10} \end{pmatrix},$$

\mathbf{K}_{22} zoals boven gedefinieerd,

$$\mathbf{K}_{33} = \begin{pmatrix} \sqrt{2}A_3 + A_4 + A_5 & A_4 - A_5 \\ A_4 - A_5 & A_4 + A_5 \end{pmatrix},$$

$$\mathbf{K}_{44} = \begin{pmatrix} A_{11} + 2\sqrt{2}A_{12} + A_{13} & A_{11} - A_{13} \\ A_{11} - A_{13} & A_{11} + A_{13} \end{pmatrix},$$

$$\mathbf{K}_{55} = \begin{pmatrix} \sqrt{2}A_9 + A_{10} + A_{11} & -A_{10} + A_{11} \\ -A_{10} + A_{11} & A_{10} + A_{11} \end{pmatrix},$$

$$\mathbf{K}_{13} = \mathbf{K}_{31} = \begin{pmatrix} -A_4 & -A_4 \\ -A_4 & -A_4 \end{pmatrix},$$

$$\mathbf{K}_{15} = \mathbf{K}_{51} = \begin{pmatrix} -A_{10} & A_{10} \\ A_{10} & -A_{10} \end{pmatrix},$$

$$\mathbf{K}_{23} = \mathbf{K}_{32} = \begin{pmatrix} -A_5 & A_5 \\ A_5 & -A_5 \end{pmatrix},$$

$$\mathbf{K}_{54} = \mathbf{K}_{45} = \begin{pmatrix} -A_{11} & -A_{11} \\ -A_{11} & -A_{11} \end{pmatrix},$$

en alle andere $\mathbf{K}_{ij} = 0$.

Vervolgens worden de spanningen bepaald door $\sigma = \mathbf{R}\mathbf{v}$, met

$$\mathbf{R} = \frac{E}{l} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1/2 & -1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 \\ -1/2 & -1/2 & 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1/2 & 1/2 & 1/2 & -1/2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & 0 \\ -1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & -1/2 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1/2 & -1/2 & 1/2 & 1/2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & -1/2 & 0 & 0 \end{pmatrix} \quad (11.41)$$

De doorsnedes van de stangen vormen de design variabelen (i.e. $\mathbf{x} = [A_1, \dots, A_{13}]$) en de objectief functie is de uitdrukking voor het volume van de structuur. De objective functie is in dit geval een lineaire functie van de design variabelen:

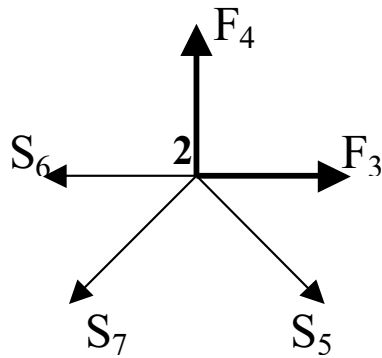
$$f(\mathbf{x}) = \sum_{i=1}^{13} l_i x_i \quad (11.42)$$

waar l_i de lengte van stang i . De constraints zijn gegeven door :

$$g_1(\mathbf{x}) = \sigma_i(\mathbf{x}) - 150MPa \leq 0 \quad (i = 1, \dots, 13) \quad (11.43)$$

$$g_2(\mathbf{x}) = -100MPa - \sigma_j(\mathbf{x}) \leq 0 \quad (j = 1, \dots, 13). \quad (11.44)$$

Verder moet $x_i \geq 0 \quad (i = 1, \dots, 13)$.



Figuur 11.9: Krachten inwerkend op knooppunt 2).

11.4.2 Optimalisatie in MATLAB

Het voorbeeld uit vorige paragraaf kan eenvoudig berekend worden met behulp van MATLAB. De M-code nodig voor de berekening van de objective functie is hieronder gegeven. Gezien het hier gaat om een eenvoudige lineaire objective functie kan ook de afgeleide analytisch berekend worden (dit maakt de optimalisatie routine sneller).

```
function [f,df] = ObjectFunctionTruss(x)
```

```
l = [1 sqrt(2) 2 sqrt(2) sqrt(2) 1 sqrt(2) sqrt(2) 2 sqrt(2) sqrt(2) 1 sqrt(2)];
f = l*x(:);
df = l(:);
```


Op dezelfde manier moet een script `ConstraintTruss` geschreven worden die voor een gegeven vector met doorsnedes x Vergelijkingen 11.43 en 11.44 uitrekent. De optimalisatie routine in MATLAB wordt dan opgeroepen door een M-file met volgende commando's uit te voeren:

```
x0 = 1e-4*ones(13,1); % Startwaarde
options = optimset('LargeScale','off');
options = optimset(options,'GradObj','on','GradConstr','off');
options = optimset(options,'Diagnostics','on','Display');
options = optimset(options,'iter','MaxFunEvals',100,'MaxIter',20);
lb = eps*ones(1,13); ub = Inf.*ones(1,13);
[x,fval] = fmincon('ObjectFunctionTruss',x0,[],[],[],[],lb,ub,'ConstraintTruss',options);
```

De resultaten van de optimalisatie zijn hieronder gegeven:

Iter	F-count	f(x)	max constraint	Step-size	Directional derivative	Procedure
1	14	0.0183137	-0.001	1	-0.0167	
2	30	0.00997437	-0.0005	0.5	-0.0061	
3	45	0.00387518	3.494e+007	1	0.000212	Hessian modified
4	60	0.00408762	5.554e+006	1	-0.000107	Hessian modified
5	75	0.00398052	5.023e+007	1	1.9e-005	Hessian modified
6	90	0.00399947	1.171e+005	1	5.19e-007	Hessian modified
7	105	0.00399999	1.109e+008	1	-7.09e-009	Hessian modified

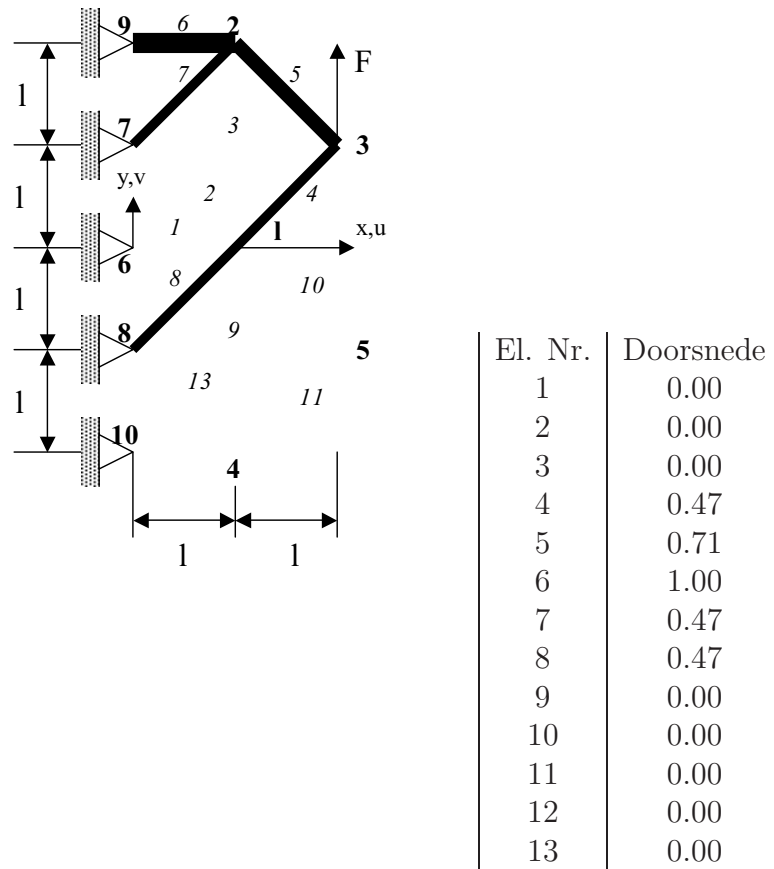
Maximum number of function evaluations exceeded;
increase OPTIONS.MaxFunEvals

De oplossing is gegeven in Figuur 11.10.

11.4.3 Ontwerp optimalisatie in ANSYS

In ANSYS bestaat de design optimalisatie procedure uit volgende stappen:

1. Creëer een 'analyse file' (extensie .lgw) die gebruikt zal worden tijdens de optimalisatie lussen. Deze file moet de volledige analyse voorstellen en moet de volgende stappen bevatten:
 - (a) Bouw het parametrisch model (PREP7).
 - (b) Bereken de oplossing (SOLUTION).
 - (c) Ken de respons kwantiteiten die gebruikt zullen worden als variabelen en objective functies toe aan de parameters (POST1).
2. Initialiseer de optimalisatie analyse:
 - (a) Voer OPT in en specificeer de analyse file (OPT).
 - (b) Declareer de optimalisatie variabelen (OPT).
 - (c) Kies de optimalisatie methode (OPT).
 - (d) Specificeer de optimalisatie lussen controleparameters (OPT).



Figuur 11.10: Geoptimaliseerde stangenstructuur (de dikte van de lijnen van de stangen is evenredig met de doorsnede).

3. Voer de optimalisatie lus uit (OPT).
4. Bekijk de resulterende design sets data (OPT) en post-process de resultaten (POST1).

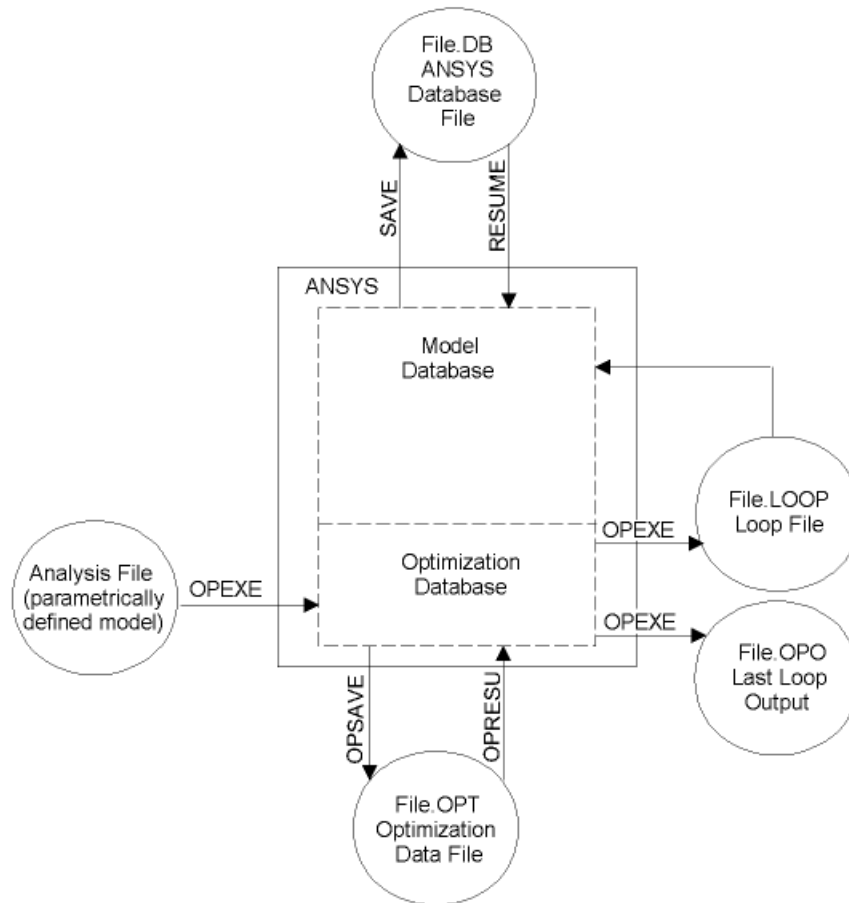
De files die bij deze procedure gecreëerd worden zijn weergegeven in Figuur

Voor het voorbeeld uit Paragraaf 11.4.1 wordt dit als volgt geïmplementeerd:

```

/PREP7
! Initialize DV parameters:
AREA1=0.001
AREA2=0.001
...

```



Figuur 11.11: Schema van de optimalisatie procedure in ANSYS met de bijbehorende files.

```

AREA13=0.001
! Define nodes
N,1,1,2,0,,,
...
N,10,0,0,0,,,
! Define elements
MP,EX,1, 2.1E11
R,1,AREA1, ,
REAL, 1
ET,1,LINK1
EN,1,1,6
...
MP,EX,13, 2.1E11
R,13,AREA13, ,
REAL, 13
ET,13,LINK1
EN,13,4,8

```

```

! LEAVE PREP7
FINISH
! Enter solution mode
/SOLU
! Define boundary conditions
D,9, ,0, , , ,ALL, , , , ,
D,7, ,0, , , ,ALL, , , , ,
D,6, ,0, , , ,ALL, , , , ,
D,8, ,0, , , ,ALL, , , , ,
D,10, ,0, , , ,ALL, , , , ,
F,3,FY,100000
! Solve analysis
SOLVE
FINISH
/POST1
SET,...
ETABLE,VOLU,VOLU          ! VOLU = volume of each element
ETABLE,SMAX_I,LS,1       ! SMAX_I = max. stress of each element
SSUM                      ! Sums the data in each column of the element
                          ! table
*GET,VOLUME,SSUM,,ITEM,VOLU ! Parameter VOLUME = total volume
!
ESORT,ETAB,SMAX_I,0,0     ! Sorts elements based on absolute value
                          ! of SMAX_I
*GET,SMAX,SORT,,MAX      ! Parameter SMAX = max. value of SMAX_I
!
ESORT,ETAB,SMAX_I,1,0    ! Sorts elements based on absolute value
                          ! of SMAX_I
*GET,SMIN,SORT,,MIN     ! Parameter SMIN = max. value of SMAX_I
FINISH

```

Na het uitvoeren van de analyse verschijnt volgend venster dat weergegeven is in Figuur 11.12

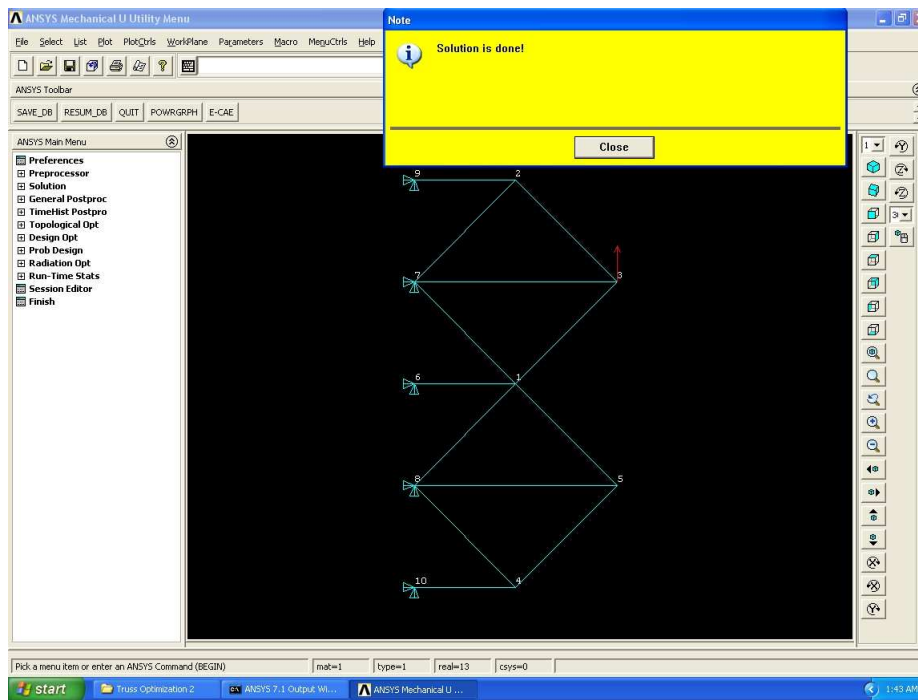
In ANSYS zijn twee standaard optimalisatie methodes beschikbaar:

1. 'subproblem approximation': dit is een geavanceerde zero-order methode die benaderingen van de dependent variabelen (state variabelen en objective functie) gebruikt. Het is een algemene methode die kan gebruikt worden voor een groot aantal problemen.
2. 'first order method': deze methode gebruikt informatie van de afgeleiden en is bijgevolg meer rekenintensief.

Bovendien kan ook een eigen optimalisatie routine gebruikt worden (USEROP).

Verder zijn er nog een aantal hulpmiddelen die dienen om een beter zicht te krijgen over de complexiteit van de design space (gezien optimalisatie niet noodzakelijk een doel hierbij is is het definiëren van een objective functie niet noodzakelijk):

1. Single Loop Run: deze tool voert een iteratie uit en produceert elke keer een FEM analyse (bv. voor verschillende waarden van design variabelen).



Figuur 11.12: ANSYS venster na de analyse.

2. Random Design Generation: meerder loops worden uitgevoerd met random design variabelen bij elke loop.
3. Sweep Generation: startende van een referentie design set worden nieuwe sets gegenereerd door elke design variabele in stappen te laten variëren over het bereik.
4. Factorial Evaluation: Hierbij worden design sets gedefinieerd door extreme combinaties van design variabelen te gebruiken.
5. Gradient Evaluation: berekent de gradient van de objective functie m.b.t. de design variabelen.
6. User-supplied Design Tool: Een externe routine (USEROP) die gebruikt kan worden om de ANSYS procedures te by-passen.

Het uitvoeren van de optimalisatie lussen gebeurt aan de hand van volgend script:

```
/BATCH
```

```

! Loading the analysis file
/INPUT,'Truss Analyse','lgw','D:\ANSYS\Truss Optimization\',, 0
/OPT
OPANL,'Truss Analyse','lgw',' '
! Definition of the 'object function', 'design variables' en 'state variables'
OPVAR, VOLUME, OBJ, , , 1e-9
OPVAR, AREA1, DV, 1e-9, 2e-3 ,
...
OPVAR, AREA13, DV, 1e-9, 2e-3 ,
OPVAR, SMIN, SV, -100000000, 150000000 ,
OPVAR, SMAX, SV, -100000000, 150000000 ,
! Definition of the optimization type
OPTYPE, FIRST
OPFRST, 100, 100,0.001 ,
! Execution of the optimization procedure
OPEXE

```

De resultaten kunnen dan in tabelvorm weergegeven worden (via 'Design Opt', 'Design Sets', 'List') zoals hieronder getoond wordt. De optimale oplossing is aangegeven met '*' tekens.

```

LIST OPTIMIZATION SETS FROM SET 1 TO SET 101 AND SHOW
ONLY OPTIMIZATION PARAMETERS. (A "*" SYMBOL IS USED TO
INDICATE THE BEST LISTED SET)

```

		SET 1 (FEASIBLE)	SET 2 (FEASIBLE)	SET 3 (FEASIBLE)	SET 4 (FEASIBLE)
SMIN	(SV)	-0.78361E+08	-0.96290E+08	-0.96175E+08	-0.99789E+08
SMAX	(SV)	0.86012E+08	0.11088E+09	0.12733E+09	0.12963E+09
AREA1	(DV)	0.10000E-02	0.83138E-03	0.73804E-03	0.72528E-03
AREA2	(DV)	0.10000E-02	0.76141E-03	0.62695E-03	0.60817E-03
AREA3	(DV)	0.10000E-02	0.66300E-03	0.48661E-03	0.46658E-03
AREA4	(DV)	0.10000E-02	0.76237E-03	0.65331E-03	0.64396E-03
AREA5	(DV)	0.10000E-02	0.76067E-03	0.60550E-03	0.58069E-03
AREA6	(DV)	0.10000E-02	0.83559E-03	0.85631E-03	0.83651E-03
AREA7	(DV)	0.10000E-02	0.76067E-03	0.60550E-03	0.57750E-03
AREA8	(DV)	0.10000E-02	0.76184E-03	0.63821E-03	0.62349E-03
AREA9	(DV)	0.10000E-02	0.66248E-03	0.46947E-03	0.44171E-03
AREA10	(DV)	0.10000E-02	0.76134E-03	0.62479E-03	0.60513E-03
AREA11	(DV)	0.10000E-02	0.76134E-03	0.62479E-03	0.60513E-03
AREA12	(DV)	0.10000E-02	0.83124E-03	0.73471E-03	0.72081E-03
AREA13	(DV)	0.10000E-02	0.76134E-03	0.62479E-03	0.60513E-03
VOLUME	(OBJ)	0.18314E-01	0.13763E-01	0.11318E-01	0.10957E-01

...

		SET 89 (FEASIBLE)	SET 90 (INFEASIBLE)	SET 91 (INFEASIBLE)	SET 92 (INFEASIBLE)
SMIN	(SV)	-0.10184E+09	-0.10182E+09	-0.10182E+09	-0.10182E+09
SMAX	(SV)	0.15249E+09	> 0.15305E+09	> 0.15302E+09	> 0.15302E+09
AREA1	(DV)	0.36564E-04	0.35369E-04	0.35300E-04	0.35300E-04
AREA2	(DV)	0.18681E-04	0.18681E-04	0.18681E-04	0.18681E-04
AREA3	(DV)	0.61565E-06	0.61565E-06	0.61565E-06	0.61565E-06
AREA4	(DV)	0.46403E-03	0.46226E-03	0.46216E-03	0.46216E-03
AREA5	(DV)	0.69425E-03	0.69437E-03	0.69438E-03	0.69438E-03
AREA6	(DV)	0.98448E-03	0.98317E-03	0.98309E-03	0.98309E-03
AREA7	(DV)	0.46364E-03	0.46193E-03	0.46202E-03	0.46202E-03
AREA8	(DV)	0.45294E-03	0.45214E-03	0.45207E-03	0.45207E-03
AREA9	(DV)	0.22004E-05	0.22004E-05	0.22004E-05	0.22004E-05
AREA10	(DV)	0.10002E-08	0.10002E-08	0.10002E-08	0.10002E-08
AREA11	(DV)	0.10002E-08	0.10002E-08	0.10002E-08	0.10002E-08



AREA12	(DV)	0.11000E-04	0.11000E-04	0.11000E-04	0.11000E-04
AREA13	(DV)	0.10006E-08	0.10006E-08	0.10006E-08	0.10006E-08
VOLUME	(OBJ)	0.39984E-02	0.39900E-02	0.39898E-02	0.39898E-02


Bijlage D

Introductie SolidWorks

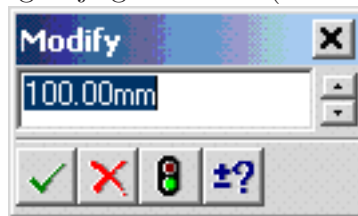
D.1 Maken van een PART in SolidWorks

1. Open een nieuw document.

2. Klik op de sketcher  waarna het venster in Figuur D.1 verschijnt en teken een rechthoek (met willekeurige dimensies) m.b.v. de  tool. Opm: zwartgekleurde lijnen zijn vast t.o.v. een referentie.




3. Maak een 100mm bij 100mm vierkant met behulp van de tool dimensie tool . Selecteer achtereenvolgens de twee overstaande zijden. Merk op dat meer lijnen zwart kleuren doordat ze vast komen te liggen.

4. Door op de maten te dubbelklikken verschijnt volgend venster waarin de maten kunnen gewijzigd worden (creëer hiermee een vierkant van





120 by 120 mm):

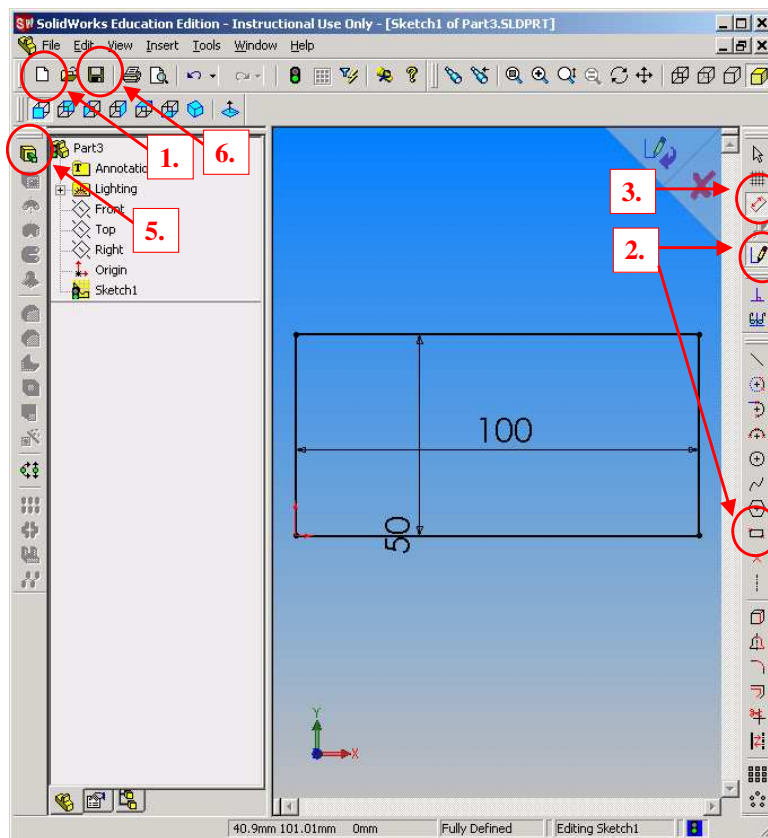
5. Door het extruderen van de sketch wordt een blok gedefinieerd. Klik hiervoor op het extrusie icoon  en voer in het daarop volgende window (zie Figuur D.2) de diepte van de opgetrokken ondedeel in (er kan ook een chamfer opgegeven worden). Klik daarna op OK.

6. Het ontwerp kan opgeslagen worden via het  icon (de file van 'parts' krijgt de extensie '.SLDPRT' terwijl 'assemblies' en 'drawings' respectievelijk met '.SLDASM' en '.SLDDRW' worden aangeduid).
7. Door een van de vlakken te selecteren met  en daarna de 2D sketcher te activeren kunnen additionele features (vb. snedes) gemaakt worden (bijvoorbeeld met de circle tool  in de sketcher een cirkel met straal 35 mm en middelpunt (60,60) te tekenen en deze te 25mm extruderen zoals voordien). Merk op dat voor het selecteren van lijnen, vlakken en punten respectievelijk een van volgende tekens zichtbaar wordt:
- 
- wordt:
8. Een concentrische boring wordt nu aangebracht door een 50mm doorsnede cirkel in het bovenste vlak te tekenen (met de sketcher) en het een relatie te definiëren tussen de middelpunten van de binnenste en buitenste cirkel. Dit wordt gedaan door de  button die een venster opent om de concentrische relatie vast te leggen na het aanklikken van beide cirkels (zie Figuur D.3). Daarna wordt met behulp van de extrude tool  de boring gedefinieerd (de diepte wordt vastgelegd door 'Through All' voor 'Direction 1' te nemen).
9. De hoeken van het onderdeel kunnen afgerond worden door na het selecteren van de desbetreffende zijden (meerdere zijden worden geselecteerd door de 'ctrl' toets in te drukken) de 'fillet' tool  of 'chamfer' tool (rechte afvlakking) te selecteren (zie Figuur D.4 voor een afronding met straal 10mm). Deze afronding is ook mogelijk voor vlakken (kies bijvoorbeeld een afronding van 5mm voor het bovenste basisvlak en 2mm voor het bovenste geextrudeerde vlak).
10. Gebruik de shell knop  om een onderdeel uit te hollen (vb. selecteer het onderste basisvlak en maak een schaal van 2mm zoals getoond in Figuur D.5).
11. De eigenschappen van een extrusie kunnen veranderd worden door: a) in de FeatureManager design tree met de rechtermuisknop 'Edit Defini-

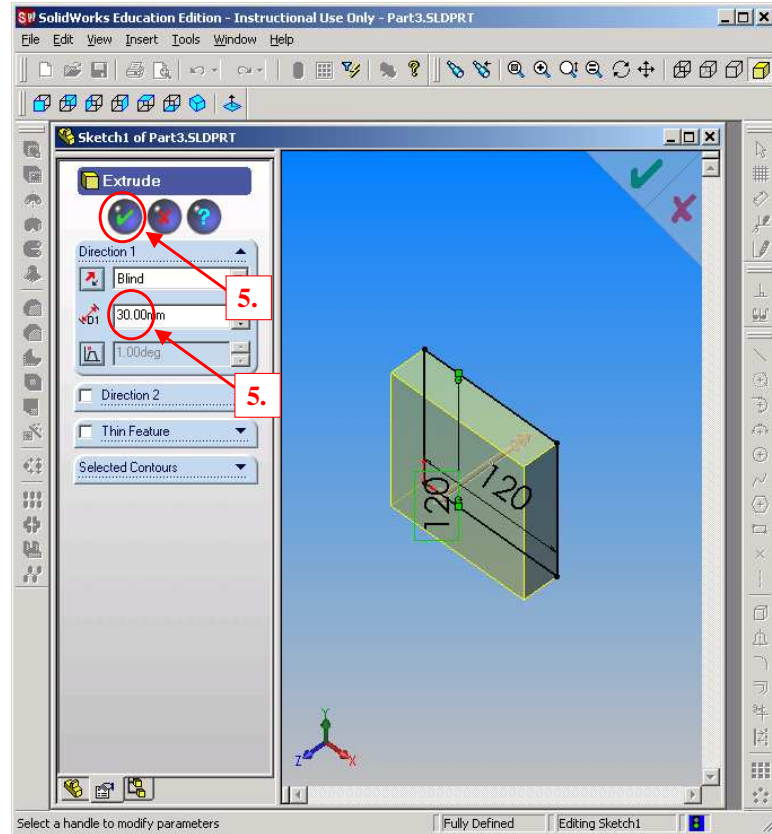
tion' te selecteren bij de gekozen extrusie en de parameters te wijzigen. b) op een dimensie behorende bij een bepaalde extrusie te dubbelklikken en deze dimensie vervolgens te veranderen, c) De Move/Size

features tool  aan te klikken na het selecteren van de extrusie in de design tree, en de handles (zie Figuur D.6) te slepen (de 'Move/Size features' tool moet daarna opnieuw afgezet worden).

12. Met de  tool kunnen secties bekeken worden vanaf een bepaald vlak zoals weergegeven in Figuur D.7 (selecteer de 'Preview' check box om on-line de sectie te kunnen bekijken). Dit wijzigt echter het onderdeel niet.



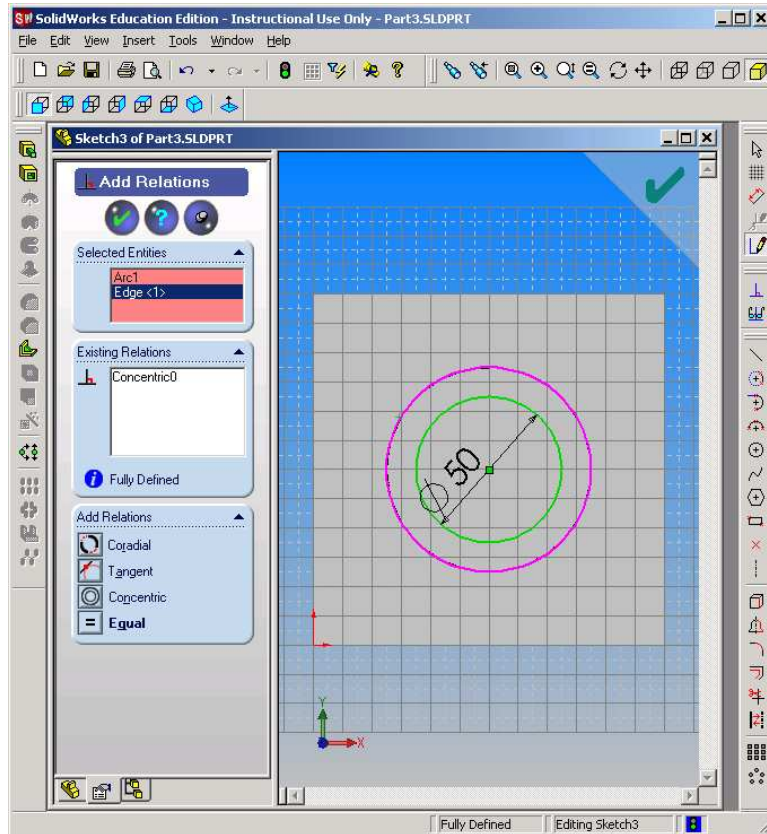
Figuur D.1: De sketcher in SolidWorks.



Figuur D.2: Extrusie van een vlak.


D.2 Maken van een ASSEMBLY in SolidWorks

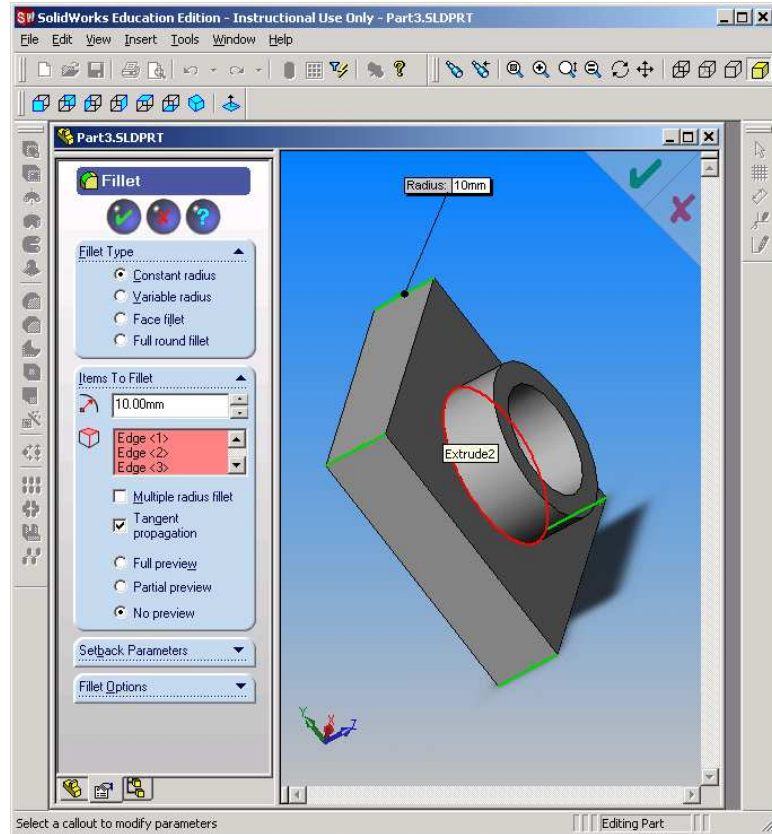
1. Open een nieuw 'part' document.
2. Maak een schaal van 120X120X90mm met 4mm dikte en 10mm afronding, zoals beschreven in vorige paragraaf.
3. Zoom in op een van de hoeken van het onderdeel met de 'zoom to area' tool . Open een nieuwe sketch op het voorste vlak van het onderdeel. In de sketcher, klik op de 'Convert entities'  button (zie Figuur D.8).
4. Klik op de 'Offset entities'  en voer in het daaropvolgende menu



Figuur D.3: Vastleggen van relaties.


2mm als afstand in en vink de 'reverse' checkbox aan. Hierdoor wordt een lijn gecreëerd die mee varieert met de maten van het onderdeel.

5. Voer nu een extruded cut uit met diepte 30mm mbv . Sla het onderdeel op.
6. Open beide aangemaakte onderdelen, en open een nieuwe 'assembly'. Zet alle window horizontaal onder elkaar.
7. Sleep de design tree van de eerste aangemaakte part naar de design tree van de assembly. Op deze manier blijven de oorsprong en de (hulp)vlakken van de parts behouden. Sleep de design tree van de tweede aangemaakte part naar het grafisch venster van de assembly (zie Figuur D.9). Indien de parts niet op de juiste plaats zitten kunnen



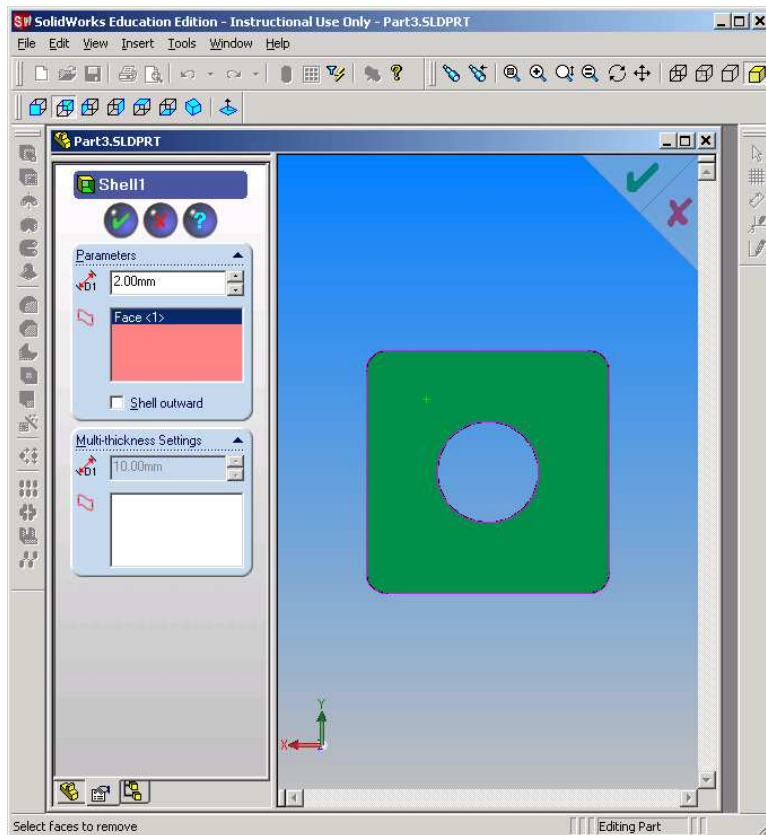
Figuur D.4: Definiëren van afrondingen.

ze later nog bewogen worden met de 'move component' tool  en de 'rotate component' tool.


8. Voor het aligneren van de onderdelen klik op de 'Mating' button  in het assembly venster. Selecteer nu eerst de bovenste rand van onderdeel 1 en de bovenste rand van de lip van onderdeel 2 en klik op OK. Doe daarna hetzelfde voor de twee respectievelijke zijranden (zie Figuur D.10). De assembly ligt nu vast.

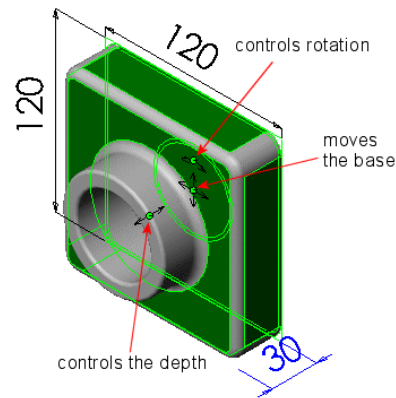
D.3 Maken van tekening (DRAWING)

1. Open een nieuwe tekening: klik op 'new Solidworks document' en daarna op de 'draw' icoon.

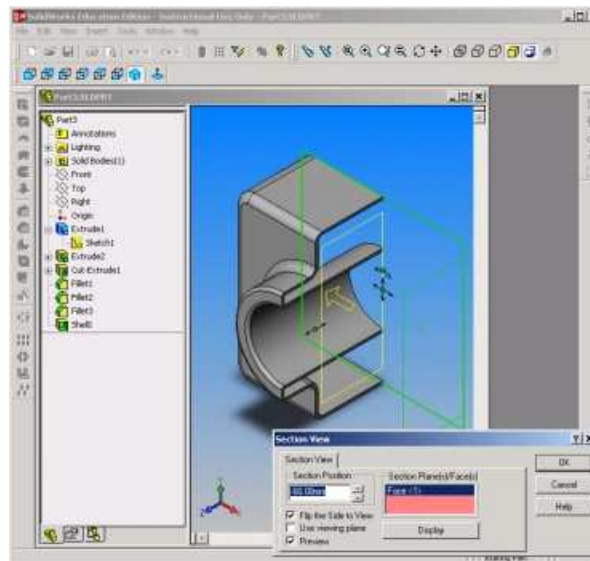


Figuur D.5: Maken van een uitholling of schaal (shell).

2. De eigenschappen van de drawing kunnen veranderd worden door op de rechtermuisknop te klikken en 'Edit Sheet Format' aan te selecteren (de 'font' van de tekst kan ook gekozen worden). Het formaat van de sheet kan apart opgeslagen worden met 'Save sheet format' in het 'file' menu.
3. Ga terug naar de 'Edit Sheet' mode dmv een selectie in het menu dat geactiveerd wordt met rechtermuisknop.
4. Selecteer de 'standard 3 view' dmv de button . Een window met een boodschap die 4 manieren beschrijft om een tekening te maken verschijnt (zie Figuur D.11). Open het eerste onderdeel aangemaakt in de eerste Sectie en selecteer de geopende part vanuit het 'view' menu (een andere mogelijkheid is op de rechtermuisknop de drukken en 'Insert



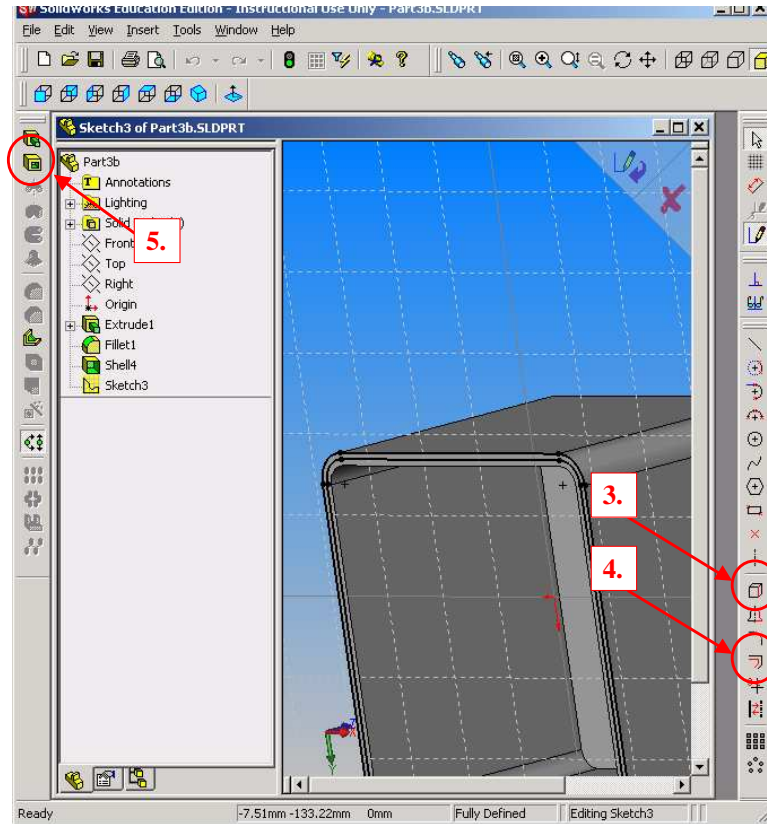
Figuur D.6: Handles voor het wijzigen van extrusies met de 'Move/Size features' tool.



Figuur D.7: Maken van secties.

from file' te selecteren).


5. Klik nu op een willekeurige plaats van het grafisch gebied in het onderdeel venster. Er verschijnt een tekening met drie aanzichten (zoals in Figuur D.12). De aanzichten kunnen afzonderlijk verplaats worden door ze te selecteren en te slepen (echter niet in alle richtingen).

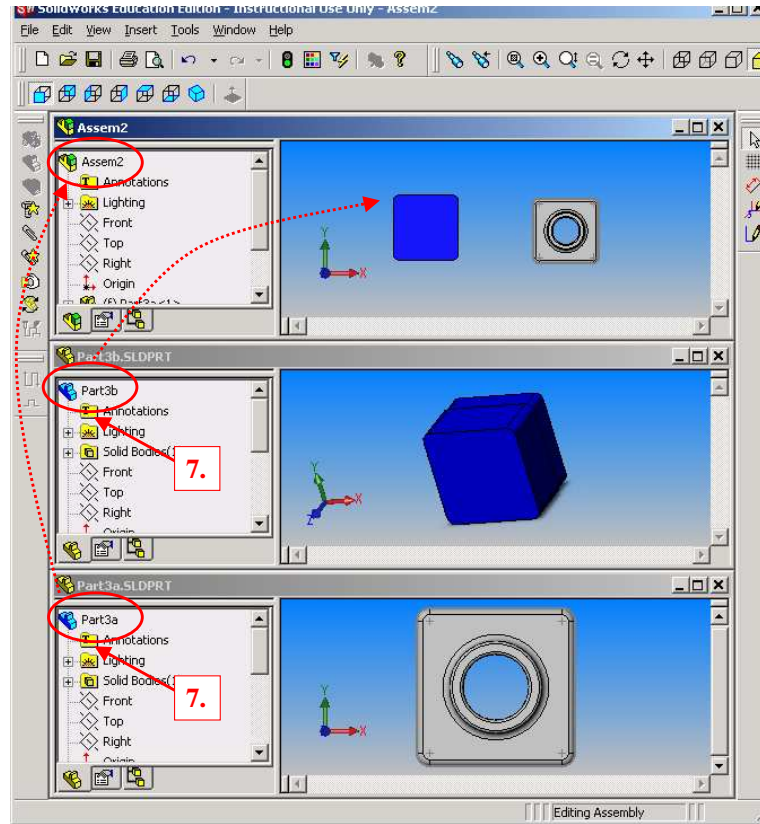


Figuur D.8: Extrusie van een lip.

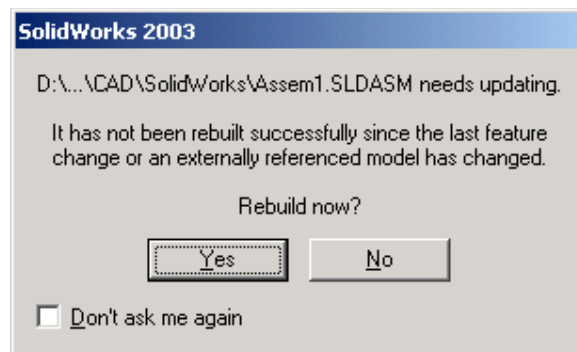
6. Voeg dimensies toe aan de tekening met behulp van 'Model items' in het 'Insert' menu. Selecteer op het daaropvolgend menu 'Dimensions' en 'Import item into all views' (zie Figuur D.13).

7. Door een dimensie aan te passen (dubbel klikken op de dimensie) en

daarna te 're-builden' met behulp van  kan het model vanuit en tekening aangepast worden (en vice versa). Wanneer assemblies worden geopend waarvan de part een onderdeel is vraag Solidworks of de assem-

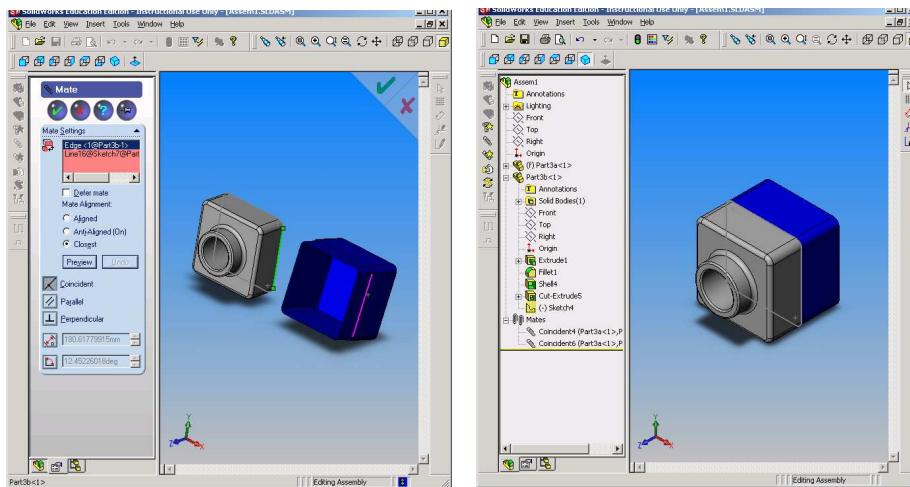


Figuur D.9: Bijeenvoegen van twee parts.

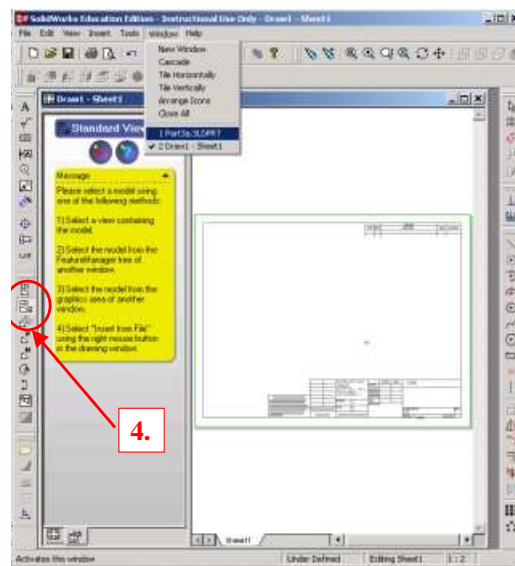


bly mag ge-rebuilt worden:

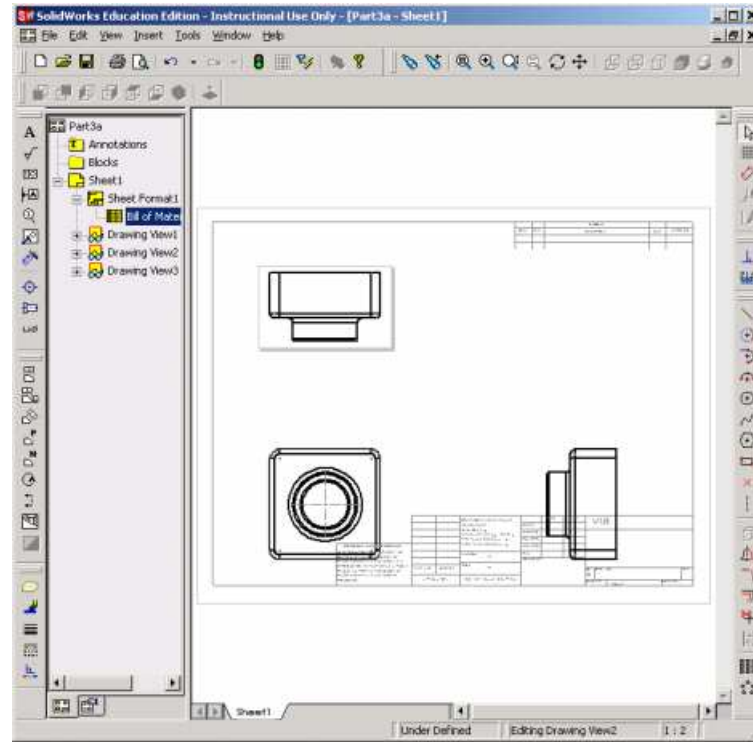
8. De tekening kan nu afgedrukt worden met het 'Print' commando onder het 'File' menu.



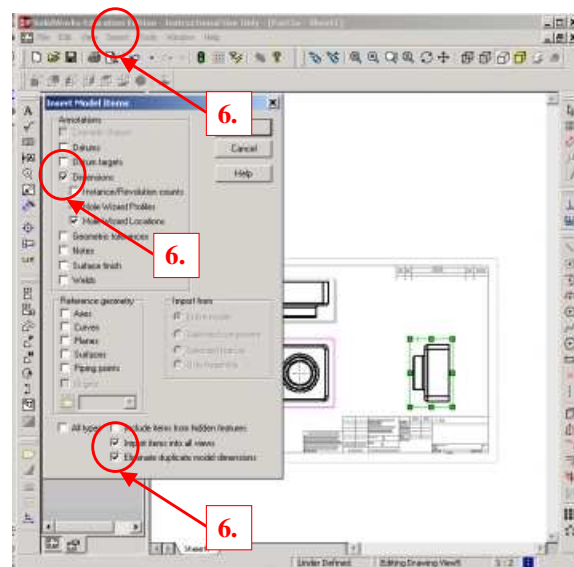
Figuur D.10: 'Maten' van twee parts.



Figuur D.11: Aanmaken van een Standard 3 view tekening.



Figuur D.12: Voorbeeld van een tekening met drie aanzichten.



Figuur D.13: Tonen van de dimensies op de tekeningen.

Bibliografie

- [1] G. S. Altschuller. *Creativity as an exact science: the theory of the solution of inventive problems*. Gordon and Brach Science Publishers, New York.
- [2] D.E. Carter and B.S. Baker. *Concurrent engineering: the product development environment for the 1990s*. Addison-Wesley, Reading, Mass, 1992.
- [3] G. E. Dieter. *Engineering Design, A Materials and Processing Approach*. McGraaw-Hill, New York, 1986.
- [4] S. Donders and H. Van der Auweraer. Engineering approach for robust vibro-acoustic design optimization. In *Proceedings of the 16th ISAAC Seminar (International Seminar on Advanced Techniques in Applied and Numerical Acoustics)*, 2004.
- [5] H. Eschenauer, N. Olhoff, and W. Schnell. *Applied Structural Mechanics*. Springer-Verlag, Berlin, Germany, 1997.
- [6] R. Fletcher. *Practical Methods of Optimization. Volume 1: unconstrained optimization*. John Wiley & Sons, New York, USA, 1980.
- [7] R. Fletcher. *Practical Methods of Optimization. Volume 2: constrained optimization*. John Wiley & Sons, New York, USA, 1980.
- [8] D. Goldberg. *Genetic Algorithms in search, optimization and machine learning*. Addison Wesley, Reading, USA, 1989.
- [9] J. R. Hauser and D. Clausing. The house of quality. *Harvard Business Review*, May-June, 1988.
- [10] N. Kano, N. Seraku, F. Takahashi, and S. Tsuji. Attractive quality and must-be quality. *Journal of the Japanese Society for Quality Control*, pages 39–48, 1984.

- [11] G. E. Moore. Cramming more components onto integrated circuits. *Electronics*, 38(8), 1965.
- [12] NN. *Werktuigbouwkundig Ontwerpen en Construeren*. tenHagenStam uitgevers, Den Haag.
- [13] NN. *A Guide to the Project Management Body of Knowledge*. Project Management Institute, Pennsylvania, USA, 2000.
- [14] NN. Software in design. Cursustext Faculteit Industrieel Ontwerpen, Technische Universiteit Delft, 2004.
- [15] B. Prasad. *Concurrent engineering fundamentals*. Prentice Hall, Englewood Cliffs, N.J., 1996.
- [16] D. G. Ullman. *The Mechanical Design Process*. McGraw-Hill, New York, 3rd edition edition, 2003.
- [17] K. Ulrich and S. A. Pearson. Assessing the importance of design through product archaeology. *Management Science*, 44(3):352–269, 1998.
- [18] Paul Verhaert. *De praktijk van de productontwikkeling*. ACCO, Leuven.
- [19] Akoa Y. New product development and quality assurance - quality deployment system. *Quality Control*, 25(4):7–14, 1972.
- [20] T. Zang, M. Hensch, M. Hilburger, S. Kenny, P. Maghami J. Luckring, S. Padula, and Virginia W. Jefferson Stroud Langley Research Center, Hampton. Needs and opportunities for uncertainty-based multidisciplinary design methods for aerospace vehicles. Technical Report NASA/TM-2002-211462, NASA, Langley Research Center, 1992.
- [21] F. Zwicky. *Discovery, Invention, Research - Through the Morphological Approach*. The MacMillan Company, Toronto.
- [22] F. Zwicky. *The Morfological Method of Analysis and Construction*. Courant Anniversary Volume. Wiley-Interscience, New York.