# Projektrapport

# Simulering og optimering af afkastforløb



DMS10 Gruppe 38 Aalborg Universitet • Institut for Maskinteknik Pontoppidanstræde 103 • 9220 Aalborg Ø



Det Teknisk-Naturvidenskabelige fakultet

Institut for Maskinteknik

Pontoppidanstræde 101 9220 Aalborg Øst Tlf.: +45 96359297 Fax: +45 98151675 http://www.ime.aau.dk

Titel: Simulering og optimering af afkastforløb Tema: Industrielt udviklingsarbejde Projektperiode: 1. Februar – 1. Juni 2007 Projektgruppe: DMS10 - 38

# Synopsis Projektet omha

Torben Stougaard

**Deltagere:** 

Alf Søe-Knudsen

Søren Emil Sørensen

#### Vejledere:

Lektor Michael Rygaard Hansen Adjunk Morten Kjeld Ebbesen Projektet omhandler simulering og optimering af afkastforløb gennem et sliskedesign. Der er konstrueret et simuleringsværktøj i from af et program. Simuleringsprogrammet tager udgangspunkt i stivlegemedynamik. Bevægelsen af legemerne beskrives ud fra Newtons tre love ved to bevægelsesligninger, der løses numerisk vha. 4.orden Runge-Kutta. Der er i simuleringsprogrammet lagt vægt på beskrivelsen af friktion, hvor der er implementeret tre forskellige friktionsmodeller, i form af Coulomb, Dahl og LuGre. Kontakten mellem legemerne tager udgangspunkt i en relaksation af de stive legemer, hvor reaktionskræfterne mellem legemerne bestemmes ud fra et fjedermassesystem. Til bestemmelse af kontakt mellem legemerne er der konstrueret en kontaktalgoritme. Algoritmen består af en grov og en fin kontaktbestemmelse. Den grove kontaktbestemmelse tager udgangspunkt i et grænsevolumenhierarki, mens den fine tager udgangspunkt i diskritiseringen af legemerne. Der er udført en verificering af simuleringsprogrammet for at anskueliggøre præcisionen af simuleringerne i forhold til reelle afkastforløb. Til verificeringen er der udført forsøg på et testsorteringsanlæg, hvor pakkernes position gennem en sliske er bestemt vha. infrarød stereobilledgenkendelse. Ud fra simuleringsprogrammet er der konstrueret et designværktøj, hvor sliskedesignet optimeres mht. pakkernes afkastforløb. Til designværktøjet er der anvendt complexmetoden der ud fra en evolutionel udvikling af de-

Oplagstal:	7
Sideantal:	163
Appendiks:	21
Bilag:	2
Andet:	Vedlagt Kildekode
	Vedlagt CD-Rom, DVD-1 og DVD-2
Afsluttet d.:	1. Juni 2007

Rapportens indhold er frit tilgængeligt, men offentliggørelse (med kildeangivelse) må kun ske efter aftale med forfatterne.

signs, finder et optimum.

# Forord

Dette projekt tager udgangspunkt i et projektforslag givet af FKI Logistex<sup>®</sup> omhandlende: *Simulering af afkastforløb*. Projektet er en videreførelse af: *Simulering af afkastforløb* fra efterårssemesteret 2006.

Rapporten er udarbejdet af gruppe 38 DMS10 Aalborg Universitet i løbet af forårssemesteret 2007 under vejledning af Lektor Michael Rygaard Hansen og Adjunkt Morten Kjeld Ebbesen. Rapporten henvender sig til vejledere og censor, FKI Logistex<sup>®</sup> samt medstuderende.

Gennem projektet har der været en løbende kontakt med Per Nielsen, Product Manager, og Erik Steen Petersen, Engineering Manager fra FKI Logistex<sup>®</sup>.

## Læsevejledning

Til siderne fra forside til og med resumé er der anvendt romertal til sidenummerering, til selve rapporten anvendes arabiske tal. Det vedlagte Appendiks, nummereres med romertal og Bilag nummereres med store latinske bogstaver. Den vedlagte kildekode er nummereret med arabiske tal.

På vedlagte CD-Rom findes rapporten *Simulering af afkastforløb* produceret af gruppen på efterårssemesteret 2006, samt rapporten *Simulering og optimering af afkastforløb* produceret af gruppen på forårssemesteret 2007. Ligeledes findes der på vedlaget CD-ROM det færdige simulerings- optimeringsprogram. Det er muligt at køre programmet *EmneSimulering.exe*, hvis mappen med programmet fra vedlagte CD kopieres over på computer. Vedlagt programmet findes visualiseringsfilerne *Graphic.m, Kost\_graf.m* og energiplot *Energy\_Plots.m.* Visualiseringen forudsætter at MATLAB R2006a er installeret.

På vedlagte DVDer findes data for den udførte verificering.

Kildelisten findes bagerst i rapporten og der er anvendt GOST til kildehenvisninger.

Ved gentagende beregninger af samme karakter henvises der til sammenhørende appendiks. Der anvendes højrehåndsreglen hvor momenter regnes positiv mod urs. Der benyttes "." som decimaltalsseparator.

# Indholdsfortegnelse

Abstract		.IX
Resumé		XII
1. Ind	ledning	1
2. Pro	blemanalyse	3
2.1.	Beskrivelse af sorteringssystemer	3
2.2.	Variation af emner	5
2.3.	Sliskedesign	5
2.4.	Generelle designparametre	7
3. Pro	blemformulering	9
4. Afg	grænsning	9
5. Pro	blembeskrivelse	.11
5.1.	Beskrivelse af afkastforløb	.11
5.2.	Variation af pakker	.12
5.3.	Bakkebevægelse	.13
6. Sim	nuleringsprogram	.21
6.1.	Valg af simuleringsmetode	.21
6.2.	Beskrivelse af simuleringsprogram	.21
6.3.	Kinematik	.22
6.4.	Inddeling af legemer	.26
7. Dyı	namik	.27
7.1.	Bevægelsesligningerne	.27
7.2.	Løsning af bevægelsesligningerne	.30
7.3.	Kontaktmodel	. 34
7.4.	Fjeder og dæmperkonstanter samt tidsstep	. 39
7.5.	Bevægelige legemer	.42
8. Fril	ktion	.45
8.1.	Friktionsmodeller	.45
8.2.	Eksperimentel bestemmelse af fiktionsparametrene	. 50
8.3.	Implementering af friktionsmodeller	. 52
9. Koi	ntaktbestemmelse	. 59
9.1.	Valg af kontaktformulering	. 59
9.2.	Diskretisering af geometri	. 59
9.3.	Opbygning af kontaktbestemmelse	. 60
9.4.	Grov kontaktbestemmelse	.61
9.5.	Fin kontaktbestemmelse	.67
9.6.	Reduktion i antallet at kontaktsøgninger	.74
9.7.	Implementering i simuleringsprogrammet	.75
10. Ir	ndledende databehandling	. 79
10.1.	Inputfil	. 80

10.	2.	Behandling af geometri	81
10.	3.	Masseinertimoment	83
10.	4.	Energiregnskab	84
10.	5.	Belastning af pakker	86
10.	6.	Visualisering	87
11.	Tes	t af simuleringsprogram	89
11.	1.	Verificering af dynamik, kontakt og friktion	89
11.	2.	Simulering af afkastforløb	91
11.	3.	Eftervisning af jam	92
11.	4.	Eftervisning af selvstart	93
11.	5.	Vurdering af simuleringsprogram	94
12.	Ver	rifikation af simuleringsprogram	95
12.	1.	Forsøgsbeskrivelse	95
12.	2.	Afgrænsning af forsøg	97
12.	3.	Forsøgsopstilling	97
12.	4.	Forsøgsudførsel	100
12.	5.	Forsøgsresultater	100
12.	6.	Databehandling af film	101
12.	7.	Test af målemetode	113
12.	8.	Resultatbehandling	114
12.	9.	Verifikationssimuleringer	116
12.	10.	Kurvetilpasning	117
12.	11.	Vurdering af verifikationssimuleringer	118
12.	12.	Tilpasning af materialeparametre	120
12.	13.	Vurdering af optimerede verifikationssimuleringer	122
12.	14.	Valgt af friktionsmodel	126
12.	15.	Delkonklusion	126
13.	Opt	timering af sliskedesign	129
13.	1.	Optimerings terminologi	129
13.	2.	Valg af optimeringsrutine	130
13.	3.	Complexmetoden	131
13.	4.	Implementering af optimeringsprogram	135
13.	5.	Input til optimeringsprogram	136
13.	6.	Optimeringsrutine	136
13.	7.	Parametrisk model	138
13.	8.	Kostfunktion	144
13.	9.	Optimeringsresultater	149
13.	10.	Vurdering optimeringsresultater	157
14.	Koi	nklusion	159
15.	Kil	deliste	161

# Appendiks indholdsfortegnelse

Appendiks I	Tandhjuls rotation ved vippebevægelse	3
Appendiks II	Kinematik	7
Appendiks III	Dæmpningsberegninger	17
Appendiks IV	Beskrivelse af friktionsfænomener	21
Appendiks V	Friktionsforsøg	29
Appendiks VI	Forsøgsrapport	57
Appendiks VII	Måledynamik	63
Appendiks VIII	Bogholderi	71
Appendiks IX	Beskrivelse af inputfiler	75
Appendiks X	Test af simuleringsprogram	81
Appendiks XI	Metoder til bestemmelse af afkastforløb	109
Appendiks XII	Pakker til verifikationsforsøg	115
Appendiks XIII	Kalibrering af kamera	123
Appendiks XIV	Præcisionstest af visionsystem	135
Appendiks XV	Plot af måledata	143
Appendiks XVI	Verifikationssimuleringer	149
Appendiks XVII	Resultater af kurvetilpasninger	153
Appendiks XVIII	Måledata og verifikationssimuleringer	155
Appendiks XIX	Måledata og optimerede verifikationssimuleringer	173
Appendiks XX	Opbygning af sliskegeometri til optimering	193
Appendiks XXI	Optimeringsresultater	197
Bilag A	Projektforslag	205
Bilag B	Liros katalog	207

# Kildekode indholdsfortegnelse

Simulering
Emne_Simulering
NumInt9
Dynamic15
GeoHandling24
Friktion
InitialCalc
FuncCall
ReadInput
Optimering
Complex
InitalOpti141
Parametric_Model
Object_Function
Optim_Kost
ReadInput_Opt190
Andet
DataTypes198
Output

# Abstract

This project is based on the problem given by FKI Logistex concerning simulation of unloading sequence through chutes. FKI Logistex is specialized in development and manufacturing of sorter systems used in various applications such as luggage handling, postal distribution etc. Along the length of the sorter systems multiple chutes are mounted which purpose is to guide packages from the sorter to a new destination. A chute design is constructed by means of experience and knowledge from previous designs. This empirical approach only provides an idea of the cute designs functionality, which involves the necessity of carrying out experiments.

Due to the above-mentioned FKI Logistex wants a computer based tool which can simulate the behavior of the packages through a chute. This simulation tool should help minimize cost and time in the development of new chute designs.

In this project a simulation tool has been developed. The simulation tool is made with an external link to SolidWorks where various geometries can be prepared for consecutive simulation. The output of the simulation is a visual representation of the interaction between the objects on basis of which the chute design is evaluated.

The simulation tool is based on rigid body dynamics. From Newton's low of motion, a rigid body can be described by two independent second order differential equations. By solving these equations numerically, the position and orientation of the rigid bodies can be determined.

The interactions between the unconstrained bodies are determined by use of a penalty method. The penalty method is based on a relaxation of the rigid body assumption, where local deformation is permitted. Based on the relaxation a contact layer is created, surrounding the bodies. When two bodies are in contact, reaction force is determined form the deformation of the contact layer.

To determine if two bodies are in contact, a contact algorithm is developed. The contact algorithm is graduated into a rough and a fine contact determination. The rough contact search is based on a boundary volume formulation whose purpose is to determine whether two volumes may come into contact. The fine contact search is based on the discretization of the geometries, which is divided into polyeder composed of surfaces, lines and vertexes. Given these elements, the fine contact search is conducted by comparing surfaces vs. vertexes, lines vs. lines and lines vs. vertexes between the bodies.

The motion of the packages through the chute is highly dependent of the friction between surfaces. Because of this dependence, it is chosen to implement three different friction models with three levels of complexity to determine which model that represents the friction behavior best and at the same time is cheap to calculate. The chosen frictions models are Coulomb, Dahl and LuGre.

To determine the friction parameters for the three models, experimental test have been carried out.

The simulation tool is verified by experimental tests. The verification is performed by comparing simulations with real measured unloading sequences through a chute at a full-scale test facility. To find the position and orientation of the packages through the chute, infrared stereo triangulation is used. Based on the verification, good correlation between the simulations and reality has been found. From the obtained measurements, an optimization of the materiel parameters was performed to obtain a better consistency concerning position, orientation, and time. The result of the optimized material parameters gave an improved accuracy.

In connection with the verification, each friction model is tested and compared to find the model that best replicates the real unloading sequence. The test showed no decisive difference between the three friction models. Nevertheless, it is recommended to use Dahl's friction model in the further use of the simulation tool. Dahls friction model is selected due to its simple correlation and good consistency with the optimized materiel parameters.

By combining the simulation tool with an optimization method, the program has become a design tool. The design tool uses the complex optimization method, that is based on an evolutionary development of a design population. The evaluation of the chute design are carried out through a simulation where a sequence of varies packages can be used.

When an optimum is reached, the optimized chute design is written to a file that can be imported to SolidWorks.

Six optimization tests of a chute design were performed to determine if the design tool finds a plausible solution to a given problem. The results of the optimizations gave an improved chute design.

# Resumé

Dette projekt tager udgangspunkt i problembeskrivelsen givet at FKI Logistex, omhandlende simulering af afkastforløb. FKI Logistex har specialiseret sig i udvikling og fremstilling af sorteringssystemer, der anvendes i forskellige sammenhænge så som, bagagehåndtering, postdistribution ol. Langs sorteringssystemet er der placeret slisker, hvis formål er at guide pakker fra sorteren ud til en ny destination. Sliskedesignet er konstrueres på baggrund af erfaringer fra tidligere designs. Denne empiriske tilgang til designopgaven giver kun en idé om sliskedesignet funktionalitet, hvormed det er ofte er nødvendigt at udføre prototypeforsøg. Pga. ovenstående problemstilling, ønsker FKI Logistex et computerbaseret værktøj, som kan simulere pakkers afkastforløb ned gennem en sliske. Simuleringsværktøjet skal medvirke til at minimere omkostningerne samt tiden anvendt i forbindelse med udvikling af nye slisker.

I dette projekt er der udviklet et simuleringsværktøj. Simuleringsværktøjet er fremstillet med et eksternt link til SolidWorks, hvor forskellige geometrier kan importeres til simuleringerne. Resultatet af simuleringerne er en visuel repræsentation af interaktionen mellem objekterne, hvorudfra sliskedesignet kan evalueres. Simuleringsværktøjet tager udgangspunkt i stivlegemedynamik. Ud fra Newtons love er bevægelserne af de stive legemer beskrives vha. to uafhængige differentialligninger der løses numerisk, for at bestemme legemernes position og orientering.

Interaktionen mellem legemerne bestemmes vha. strafmetoden. Strafmetoden tager udgangspunkt i en relaksation af de stive legemer. Ud fra relaksationen påføres legemerne et kontaktlag, hvorudfra reaktionskræfterne bestemmes.

Kontakten mellem legemerne bestemmes ud fra en konstrueret kontaktalgoritme. Kontaktalgoritmen er inddelt i en grov og en fin kontaktbestemmelse. Den grove kontaktbestemmelse tager udgangspunkt i et grænsevolumenhierarki, hvis formål er, hurtig, at afgøre om to legemer er i kontakt. Den fine kontaktsøgning tager udgangspunkt i en diskretiseringen af geometrierne, der er inddelt flader, linjer og knuder. Ud fra disse udføres den fine kontaktbestemmelse, mellem legemerne, ved at sammenligne flader mod knuder, linjer mod linjer og linjer mod knuder.

Pakkernes opførsel ned gennem sliskerne er stærkt afhængig af friktionen mellem overfladerne. På baggrund af denne afhængighed er det valgt at implementere tre forskellige friktionsmodeller, med tre niveauer af kompleksitet. Dette er gjort for at finde den model der repræsentere friktionsopførslen bedste og på samme tid er billig at beregne. De valgte friktionsmodeller er Coulomb, Dahl og LuGre.

Til at bestemme friktionsparametrene til de tre modeller er der udført friktionsforsøg. Pålideligheden af simuleringsprogrammet er bestemt ved en verifikation. Verifikationen er udført ved at sammenligne simuleringer med reelle afkastforløb. Til at bestemme pakkernes position og orientering, gennem sliskerne, er der anvendt infrarød stereotriangulering. Ud fra verificeringen er der fundet en god overensstemmelse mellem simuleringerne og de reelle afkastforløb.

Ud fra verifikationsforsøgene er der udført optimeringer af materialeparametrene, til simuleringsprogrammet, for at opnå en bedre præcision af positionen, orienteringen og tid. Optimeringen af materialeparametrene gav et forbedret resultat.

I forbindelse med verifikationen er de enkelte friktionsmodeller sammenlignet, for at finde den model der bedst beskriver det reelle afkastforløb. Testen viste ingen afgørende forskel mellem de tre implementerede friktionsmodeller. Det er dog ud fra den udførte verifikation valgt at anbefalet Dahls friktionsmodel i forbindelse med en videre anvendelse af simuleringsprogrammet. Dahls model er anbefalet pga. dens simple opbygning samt på baggrund af de ensartede resultater, opnået ved verificeringen.

Ved at samle simuleringen med en optimeringsmetode er programmet udvidet til et designværktøj. Designværktøjet anvender complex optimeringsmetoden, hvilket er baseret på en evolutionel udvikling af en population af designs. Evalueringen af sliskedesignet udføres gennem en simulering, hvor en repræsentativ variation af pakker kan anvendes. Når et optimum er fundet, udskrives det optimerede sliskedesign til en fil, der kan importeres til SolidWorks.

Der er udført seks optimeringer af et sliskedesign for at bestemme om designværktøjet finder frem til et plausibelt resultat, til et givet problem. Resultater af optimeringerne gav et forbedret sliskedesign.

# 1. Indledning

FKI Logistex er en global koncern med divisioner over hele verden. Virksomheden har på verdensplan 3000 medarbejdere, hvor 480 er ansat i Danmark. FKI har specialiseret sig i integrerede materialehåndteringsløsninger til distribution, bagagehåndtering, fragt og virksomhedslogistik ol. (FKI). Kundesegmentet dækker virksomheder som lufthavne, postvæsner, varehuse, tekstilfabrikker, biblioteker ol. Divisionen i Danmark varetager udvikling og fremstilling af sorteringsanlæg, der indgår som en del af FKI Logistex's produktsortiment. Dette projekt er givet af FKI Logistex og omhandler simulering af pakkers afkastforløb. I forbindelse med design af slisker, hvis formål er udsortering af emner til en givet destination, er sliskedesignets parametre bestemt ud fra erfaring. Erfaringerne bygger på de ansattes viden om tidligere sliskers udformning, samt hvilke emner, overflader og lignende der tidligere er anvendt. Denne empiriske tilgang giver kun en ide om sliskens funktionalitet, hvilket betyder at forsøg er nødvendig for at sikre det færdige designs funktionalitet. Hele design- og testfasen tager omkring én måned og kan derfor anses som en omkostningstung proces.

Selvom test er udført på slisken, sikrer det ikke, at forskellige parametre ændrer sig ved den endelige opsætning af anlægget. Eksempler på disse, er parametre som temperatur- og fugtighed, hvilket kan resultere i forandringer i friktionskoefficienten mellem sliske og emne. Disse ændringer medfører i værste fald at samtlige slisker skal redesignes, efter anlægget er opstillet.

Kapacitet og skyggearealet er vigtige parametre i designet af sorteringsanlæg, hvorfor det altid forsøges at optimere disse. Dette indebærer stadig stigende krav til sliskernes design, bl.a. ønsket om at modtage emner med større og større hastighed, og på samme tid minimere sliskernes indløbsbredde, for at kunne få plads til flere slisker på samme areal.

## 1.1.1. Ønsker og krav

I forbindelse med overstående designopgave er det FKI's ønske at få et værktøj der kan simulere afkastforløbet af et emne. Dette værktøj skal give en større sikkerhed i valg af sliskedesign og give en ide om en mere optimeret sliskebredde.

Værktøjet skal være CAD baseret, eller der skal være et link til deres CAD system, således at slisker kan implementeres direkte fra deres tegningsdatabase.

Følgende parametre skal kunne justeres.

- Emnets størrelse
- Emnets vægt
- Emnets tyngdepunkt
- Friktionen (gerne inkl. indflydelse fra temperatur/fugtighed)
- Hastighed og retning, hvormed emnet leveres til sliskeindløbet.

Resultat af simuleringen skal være, at det kan eftervises.

- at pakker glider rigtigt ned og evt. vender på vejen.
- om et givet design giver jam.
- at slisken er selvstartende i alle områder. Dvs. en pakke placeret vilkårligt på slisken skal begynde af glide.

Ønsker, ikke krav

- Ved akkumulering, hvor stort et tryk er der på de enkelte pakker.
- Deformation af pakker.
- Indikere impact zoner på konstruktion, gerne med en angivelse af impact størrelsen.

# 2. Problemanalyse

Denne analyse har til formål, ud fra opgaveformuleringen, at klarlægge opgavens omfang. Dette anskueliggøres ved at beskrive principperne bag FKI's sorteringssystemer, samt variationen af disse. Derefter beskrives de emnetyper anlægget sorterer og hvilke minimale og maksimale emner anlægget kan betjene. Efterfølgende beskrives variationen af sliskedesigns, samt hvilke overflade- og materialesammensætninger der anvendes. Problemanalysen skal give anledning til en efterfølgende problemformulering og eventuelt en problemafgrænsning.

Følgende oplysninger er hentet fra FKI's hjemmeside (FKI), samt gennem kommunikation med FKI Logistex.

# 2.1. Beskrivelse af sorteringssystemer

FKI's sorteringssystemer er overordnet opdelt i to typer. Den ene er en omdirigering af emnet til et andet transportbånd, dvs. et sporskifte, mens den anden er en sortering af emner ud til flere destinationer. Da projektoplægget omhandler sidstnævnte sorteringstype vil følgende beskrivelse omhandle disse.

Det generelle kendetegn ved sorteringssystemer af denne type er at emner afkastes gennem én sliske.

Hos FKI Logistex, Danmark, fremstilles der fire forskellige udsorteringssystemer:

- S-3000M Tilt-Tray Sorter
- S-3000E Tilt-Tray Sorter
- S-3000CB Cross-Belt
- S-3000DT Double Tray

Årsagen til den store variation i sorteringssystemer er ønsket om at kunne håndtere en meget bred variation af emner, bl.a. mht. skrøbelighed, geometri og vægt.

## 2.1.1. S-3000 serien

S-3000 systemet består af en lukket bane, hvor to C-profiler fungere som skinner. På én bane kan der køre op til 5000 vogne. Vognene er sat sammen så de danner en kæde, der passer med banens længde. Vognene er udført i tilpassede aluminiumsprofiler, se Figur 2.1. På enderne af tværprofilet er der monteret to hjul, som kører inde i C-profilerne. Banen er modulopbygget, hvor modulerne findes med forskellige hældninger og sving. Modulopbygningen betyder at banens udformning kan varieres efter den enkelte kundes behov. Vognene trækkes rund vha. flere lineære elektriske motorer, fordelt jævnt rundt på banen. Sorteringssystemerne er ofte placeret højt over gulvet for at udnytte fabriksarealet optimalt. Det der adskiller de enkelte typer i S-3000 serien er systemet ovenpå de enkelte vogne.



Figur 2.1: S-3000E vogn.

#### S-3000M

Emnernes transport rundt på banen foregår på en bakke af bøgefiner. Når emnet kastes af, foregår det ved hjælp af en mekanisk vippeanordning, placeret mellem bakken og vognen. Vippeanordningen kan vippe 35 grader til begge sider i forhold til kørselsretningen. Sorteringsbåndet er begrænset af en maksimal hastighed på 2.5 *m/s*, hvor højere hastighed vil resultere i at emnet falder af bakken pga. de radielle accelerationskræfter i sving. S-3000M har en kapacitet, på emner, med en maksimal længde, ved anvendelse at én bakke, på 1000 *mm* eller 1475 *mm* ved anvendelsen af to bakker. Den maksimale emnebredde er 1000 *mm* og den maksimale vægt er 50 *kg*.

#### S-3000E

De overordnede principper for S-3000E serien er de samme som ved S-3000M, dog med den forskel at den mekaniske vippemekanisme er erstattet af en elektrisk. Forskellen medføre at bakken kan vippes mellem 0 og 45 grader til begge sider. Denne mulighed udnyttes, ved at vippe bakkerne ind mod centrum, rundt i sving, hvilket medfører at anlægget kan opnå en hastighed på 3 m/s.

#### S-3000CB

Ved Cross-Belt systemet er der på vognene monteret et plant transportbånd. Idet båndet aktiveres transporteres emnet ud på tværs af kørselsretningen. Cross-Belt systemet er designet til at sorterer de emner de øvrige sorteringssystem har problemer med at håndtere, dvs. skrøbelige emner og emner med en høj friktionsmodstand. Systemet kan håndtere emner med en maksimal længde på 1200 *mm*, bredde 600 *mm* og en vægt på maksimalt 30 *kg*.

#### S-3000DT

Double-Tray ligner det tidligere nævnte Tilt-Tray system, hvor der i stedet for en bakke er monteret to uafhængige bakker pr. vogn. Udsorteringen af emnerne sker ved at bakkerne kan vippe til hver deres respektive sider, hvormed der opnås et højere materialeflow pr. sorteringsanlæg.

Systemet håndterer emner med en maksimal længde på 555 *mm*, bredde 530 *mm* og en vægt på 25 *kg*.

## 2.2. Variation af emner

FKI specialiserer sig i kundespecifikke anlæg. Derfor kan emner der håndteres af sorteringsanlæggene, dog med begrænsninger på størrelse og vægt, variere endeløst. Nogle af de emner der behandles er kufferter, sportstasker, breve, papkasser, plastkasser, div. fødevarer, tøj ol. For emner der ikke kan stå stabilt på bakkerne, placeres disse i en kasse under transporten. Alt efter hvilken type sorteringsanlæg der anvendes kan variationen af emnets størrelse og vægt variere fra nul til en maksimal længde og bredde på 1475 *mm* · 1000 *mm* samt en maksimal vægt på 50 *kg*.

Med så stor en emnevariation kan en generel beskrivelse af emnerne ikke udføres. Derimod kan nogle overordnede parametre for emnevariationen opstilles.

- Størrelse
- Vægt
- Tyngdepunkt
- Overflademateriale
- Geometri

En nærmere undersøgelse af variation af disse parametre gennemgås for det valgte emne i afsnit 5.1.

# 2.3. Sliskedesign

Langs de forskellige sorteringsanlæg er der placeret slisker, hvis formål er, at lede emnerne i en given retning. Sliskerne kan enten føre emnerne videre til et nyt transportbånd eller de kan akkumulere emnerne til en videre pakning. Eftersom FKI leverer meget kundespecifikke produkter, er det ofte nødvendigt, at designe slisker tilegnet den enkelte kunde. Dette betyder at FKI med tiden har opbygget et betydeligt sliskesortiment, hvis omfang vil være formålsløst af beskrive. Det er valgt er fokuser på tre overordnede designprincipper, se Figur 2.2.



Figur 2.2: Generelle sliskedesigns. ØV: Trugsliske. ØH: Kaskadesliske. NV: Vindelsliske.

#### Trugsliske

En trugsliske er udformet som et trug udført i bukket stålplade. Trugformen har til formål at guide emnerne mod midten af udløbet. Slisken anvendes ofte, hvor det ønskes at ændre emnets retning 90 grader i forhold til sorteringsanlægget. Endvidere anvendes trugslisken ofte i forbindelse med en rullebane samt hvor der ønskes en skånsom behandling af emnet.

#### Kaskadesliske

Kaskadeslisken er som vist på Figur 2.2 en plan sliske med tilhørende vinkelrette sider. Kaskadeslisken er designet til akkumulering af emner og er derfor udført med en undersænkning ca. 1 *m* fra indløbet. Undersænkningen gør det muligt at opbygge flere lag af emner under akkumulering. Slisken anvendes ofte til postpakker, hvor der er et ønske om akkumulering, men ingen krav til skånsom behandling af emnerne.

#### Vindelsliske

Vindelslisken anvendes når emnet skal transporteres fra et højre niveau og ned til et lavere niveau, på et lille skyggeareal. Vindelslisken er opbygget omkring en centerstolpe, hvor flere bukkede plader er monteret, for at danne spiralformen. Indløbet er udformet således at emnerne guides ind mod spiralens centrum. Hældningen på spiralen varierer alt efter emne-type med enten 400 *mm* eller 500 *mm* for hver 90 grader. Vindelslisken er designet således

at emnerne forlader slisken med ca. samme hastighed. Dette opnås ved hjælp af centrifugalkraften det tvinger de tunge emner ud i en større radius og dermed en længere bane.

#### Generelle designparametre for slisker

Sliskernes overordnede design har en stor variation, men enkelte designparametre er gennemgående. Sliskerne er fortrinsvis udført i stålplader malet med en friktionslak. Friktionslakken har en overflade med samme friktionskarakteristisk som blank stål. Dette medfører en ensartet friktionsopførsel og dermed et ensformig afkastforløb gennem hele sliskens levetid.

Designet af de forskellige slisketyper foregår, som beskrevet, ud fra erfaring, hvor der bliver justeret på de enkelte sliskers parametre indtil emnernes forløb er som ønsket. For Trugslisker kunne disse parametre være, som vist på Figur 2.3.



Figur 2.3: Designparametre for trugsliske.

Bredden af indløbet og indløbsvinkelen bestemmes ud fra hastigheden af sorteringsanlægget, samt størrelsen og udformningen af emnerne. Ved høje hastigheder er det nødvendig med et stort indløb for at guide emnet ind i slisken. Hældningen og længden af slisken bestemmer udgangshastigheden på emnerne. Hældningens størrelse bestemmes som et kompromis mellem skyggeareal og belastningen på emnerne. Sliskebredden afhænger af emnets størrelse og holdes på et minimum.

## 2.4. Generelle designparametre

For at opnå et stabilt fald af emnet, tilstræbes det at placere sliskerne på steder, hvor sorteringsbåndet er plant og lige. Dermed undgås gyroskopiske og vertikale kræfter på emnet i en afkastsituation. I enkelte tilfælde, ofte pga. begrænset plads, kan det være nødvendigt at placere slisker i sving eller steder med hældning på båndet.

Pga. toleranceusikkerheder er sliskernes indløbskant placeret 60 *mm* vertikalt og 10 *mm* horisontalt i forhold til de vippede bakker.

# 3. Problemformulering

I hht. opgaveformuleringen og problemanalysen er følgende problem formuleret. I forbindelse med udsortering af emner gennem slisker, er designet af nye slisker baseret på erfaring fra tidligere sliskedesigns. Denne empiriske tilgang til sliskedesignet medfører en tidskrævende og omkostningstung designfase, hvor optil flere prototypeforsøg er nødvendige. Foruden den omkostningstunge designfase, er kapacitet og skyggeareal vigtige parametre, der ønskes minimeret, ved fremtidige designs. Disse parametre medfører krav til nye slisker om at kunne modtage emner med stadig større hastighed og på samme tid fylde mindst muligt.

Med udgangspunkt i ovenstående, er følgende problemstillingen formuleret:

Hvorledes fremstilles et computerbaseret værktøj, der kan beskrive emners forløb ned gennem en sliske og derigennem hjælpe med udviklingen af et optimalt sliskedesign?

Problemstillingen løses ved at konstruere et simuleringsværktøj der kan vise emners afkastforløb, fra sorteringsbåndet, ned gennem én sliske. Der lægges vægt på beskrivelsen af friktion med henblik på at opnå en realistisk simulering. For at øge designværktøjet anvendelighed fokuseres der på at reducere beregningstiden af en simulering.

Simuleringsværktøjet verificeres ved at sammenligne målinger af reelle afkastforløb. Derigennem ønskes simuleringsværktøjet præcision eftervist, hvilket skal give tillid til en videre anvendelse af simuleringsværktøjet.

På baggrund af simuleringsværktøjet konstrueres et designværktøjet, der ud fra simuleringer, finder et optimalt sliskedesign. Designværktøjet skal give anledning til færre prototypeforsøg, for derigennem at reducere omkostningerne ved udvikling af nye sliskedesigns.

# 4. Afgrænsning

Det er valgt at afgrænse projektet mht. opgaveformuleringen på følgende områder:

- Der tages udgangspunkt i sorteringsanlægget S-3000E serien.
- Sortimentet af emner begrænses til kun at omhandle pakker i form af papkasser.
- Temperatur og fugtighed holdes konstant, for at reducere forsøgsudførslen og databehandlingen.
- Der anvendes samme friktionsegenskaber til sliske og bakke.
- Designværktøjet begrænses til kun at omhandle trugsliskedesignet.

# 5. Problembeskrivelse

Dette Kapitel indeholder en nærmere analyse af det afgrænsede problem. Indledningsvis analyseres det valgte sorteringsanlæg og sliskedesigns, mht. hvilken kontaktflader og bevægelser der indgår i afkastforløbet. Herefter beskrives variationen af pakkernes størrelser, vægt, tyngdepunkt, mm., hvorefter bakkens vippebevægelse beskrives.

## 5.1. Beskrivelse af afkastforløb

Selvom et større sorteringsanlæg kan indeholde adskillige slisker og bakker kan beskrivelsen af afkastforløbet reduceres til kun at omhandle afkast fra én bakke til én sliske. Afkastet af emnet starter idet bakken begynder at vippe, hvorefter emnet begynder at glide. Herefter mister emnet kontakten med bakken og falder ned gennem indløbet af slisken. Simuleringsværktøjet skal, i hht. problemformuleringen, vise emnets forløb gennem slisken, og det er derfor ikke relevant at inddrage alle komponenterne i sorteringsanlægget, da disse ikke har nogen direkte indflydelse på forløbet gennem slisken. Derfor udføres der i det følgende en reduktion af de implicerede dele, så problemstillingen indeholder et minimum af komponenter.

## 5.1.1. Sorteringsanlæg

Sorteringsanlægget består af flere komponenter, hvor et udsnit, i form af en vognsektion, er vist på Figur 5.1. Konstruktionen kan inddeles i vogn, vippemotor og bakke.



Figur 5.1: Vognsektion fra S-3000E inddelt i vogn, vippemotor og bakke.

Sammenlagt udfører de tre dele den bevægelse, der får pakken til at glide af bakken, hvor vognen giver en vandret og vippemotoren en roterende bevægelse.

Det er muligt at fjerne alle delene fra simuleringen og i stedet bestemme hastighed og retning, hvormed emnet leveres til slisken. Det vil dog ikke være hensigtsmæssigt eftersom bakken giver anledning til en stor variation i afkastet pga. friktionen mellem bakken og pakken. Det er derfor valgt at inddrage bakken i simuleringerne, således at et mere realistisk afkastforløb opnås.

Bakkerne er fremstillet i lakeret bøgefiner. Da bakkerne indeholder flere bolthuller og rundinger der ikke har indflydelse på pakkens afkast, er geometrien forsimplet ved at fjerne disse.

### 5.1.2. Sliske

Emnets forløb gennem slisken er kun influeret af de indadvendte flader. Derfor har de resterende flader i geometrien ingen indflydelse på simuleringens resultat, og kan derfor undlades. Ligeledes kan alle huller og rundinger der ikke har indflydelse på emnets forløb fjernes. På Figur 5.2 er en trugsliske med hovedmål illustreret.



Figur 5.2: Trugsliske med hovedmål.

## 5.2. Variation af pakker

Det er valgt at reducere emnerne til kun at omhandle pakker. Variationen af pakkernes geometri har en stor betydning for de enkelte pakkers afkastforløb og kan derfor betragtes som en designparameter.

Emnerne i form af pakker består af en genstand, der er pakket ind i pap for at beskytte genstanden under transport. Da genstanden inde i pakken kan have en arbitrær geometri kan indpakningen ligeledes være af arbitrær form. Generelt anvendes standardiserede udformninger af pakkerne, i form af kasser eller cylindere. Det er valgt at reducere pakkernes geometri til kun at være rektangulære kasser. Ligeledes ses der bort fra tape og andet der kan være påsat kassen. I Tabel 5.1 er data fra FKI, for pakkernes minimale og maksimale længde, højde og bredde givet.

Pakker	Minimum [ <i>mm</i> ]	Maksimum [mm]
Længde	150	1200
Bredde	100	700
Højde	20	700

Tabel 5.1: Maksimale og minimale størrelser på pakker.

Vægten af pakkerne varierer fra 1 til 31.5 kg, hvor tyngdepunktets placering er bestemt af pakkens indhold. Der findes ingen data for variationen af tyngdepunktets placering i emnerne, men det forventes at variere i et vist omfang.

## 5.3. Bakkebevægelse

Bakken der transporterer og afkaster pakken har en translatorisk samt en roterende bevægelse. Disse to bevægelser er til enhver tid uafhængige af hinanden.

Det valgte sorteringsanlæg, S-3000E, har en maksimal translatorisk hastighed på 3 *m/s*. Ved afkast af emnet vippes bakken således at den opnår en vinkel af 42°, hvorefter den vipper tilbage til udgangspositionen. Vippebevægelsen foregår efter en speciel kurve, hvor rotationspunktet bevæger sig vertikal mens bakken vipper. Vippebevægelsen foregår omkring en omdrejningsakse der er parallel med bevægelsesretningen, se Figur 5.3



Figur 5.3: Vognen med vippemekanisme, hvorpå bakken er monteret.

Tandkransen, der styrer afkastet, er udformet som to halvcirkler forbundet af et lige stykke, se Figur 5.4. Motoren der trækker tandhjulet er fastmonteret på vognbunden.



Figur 5.4: Tandkransen der styrer vippebevægelsen. Tandkransens geometri består af to halvcirkler forbundet af en fælles tangent.

Det er valgt at beskrive bakkens vippebevægelse vha. 2D kinematik, hvor det antages at motorens moment er så stort at systemets dynamik kan negligeres. Den bestemte vippebevægelse anvendes i simuleringsprogrammet til beskrivelse af afkastforløbet.

### 5.3.1. Beskrivelse af vippebevægelse

Det er valgt at opdele vognens vippemekanisme i to legemer, *Legeme 1* det drivende tandhjul og *Legeme 2* tandkransen, se Figur 5.5. De øvrige legemer følger bevægelsen af disse to og kan derfor undlades. For at reducer det kinematiske systems frihedsgrader til én, opstilles der fem tvangsbindinger, hvormed det er muligt at beskrive vippebevægelsens opførsel på baggrund af tandhjulets rotation.



Figur 5.5: Tandkrans og tandhjul, med de lokale og globale koordinatsystemers placering.

Det er valgt at opdele den kinematiske analyse i to situationer alt efter om det drivende tandhjul bevæger sig på det lige, eller den cirkelformede del af tandkransen.

#### Situation 1

På Figur 5.6 er en skitsetegning af det kinematiske system for den lige del af tandkransen illustreret.



Figur 5.6: Kinematisk system for den lige del af tandkransen.

Eftersom tandkransen er tangent til tandhjulet, skal det gælde at vektoren **d** er vinkelret på enhedsvektoren langs  $\xi_2$ .

$$\Phi_1 = \mathbf{t}^T \mathbf{d} = 0 \tag{5.1}$$

Hvor

$$\mathbf{t} = \mathbf{A}_2 \begin{bmatrix} 1\\ 0 \end{bmatrix}$$
(5.2)

Hvor **d** er vektoren fra tandhjulets centrum til kontaktpunktet *P*, mellem tandhjulet og tandkransen.

$$\mathbf{d} = \mathbf{r}_2 + \mathbf{A}_2 \, \mathbf{s}_2^{P'} + \mathbf{r}_1 \tag{5.3}$$

Eftersom kontaktpunktet P flytter sig med vippebevægelsen er den lokale vektor  $\mathbf{s}_{2}^{P'}$  givet ved.

$$\mathbf{s}_{2}^{P'} = \begin{bmatrix} (\theta_{1} - \theta_{2}) \cdot R_{1} \\ 0 \end{bmatrix}$$
(5.4)

For at sikre kontakt mellem tandhjulet og tandkransen, skal vektoren **d** have længden  $R_1$ .

$$\Phi_2 = \mathbf{d}^T \mathbf{d} - R_1^2 = 0 \tag{5.5}$$

Eftersom rotationspunktet *A* kun må flytte sig vertikalt i det globale koordinatsystem, skal det gælde at.

$$\Phi_3 = \left(r_2 + \mathbf{A}_2 \mathbf{s}_2^{\mathcal{A}'}\right)^T \cdot \begin{bmatrix} 1\\0 \end{bmatrix} = 0$$
(5.6)

Tandhjulet låses i x- og y-retningen ved bindingerne  $\Phi_{4.5}$ .

$$\mathbf{\Phi}_{4,5} = \mathbf{r}_1 + \mathbf{s}_1^G = \mathbf{0} \tag{5.7}$$

Driverligning  $\Phi_6^d$  for tandhjulets rotation er givet ved.

$$\Phi_6^d = \theta_1 - \dot{\theta}_1 \cdot t = 0 \tag{5.8}$$

Ovenstående tvangsbindinger anvendes indtil punktet *P* passerer punktet *K*. Herefter anvendes tvangsbindingerne givet i situation 2.

#### Situation 2

På Figur 5.7 er en skitsetegning af det kinematiske system for den cirkelformede del af tandkransen illustreret.



Figur 5.7: Kinematisk system for den cirkelformede del af tandkransen.

I situation 2 indføres vinklen  $\theta_3$  med følgende relation til  $\theta_1$ .

$$\theta_3 = \left(\frac{R_1}{R_2 - R_1}\right) \cdot \left(\theta_1 - \theta_K\right) \tag{5.9}$$

Hvor  $\theta_K$  er tandhjulets rotation hen til punktet *K*. Tilsvarende situation 1 gælder det af tandhjulet og tandkransen skal have fælles tangent i kontaktpunktet *P*.

$$\Phi_1 = \mathbf{t}^T \mathbf{d} = 0 \tag{5.10}$$

Hvor d tilsvarende er givet ved.

$$\mathbf{d} = \mathbf{r}_2 + \mathbf{A}_2 \, \mathbf{s}_2^{P'} + \mathbf{r}_1 \tag{5.11}$$

Den lokale vektor  $\mathbf{s}_{2}^{P'}$  fra punktet *O* til kontaktpunktet *P* er givet ud fra  $\theta_{3}$  ved.

$$\mathbf{s}_{2}^{P'} = \mathbf{s}_{2}^{C'} + R_{2} \cdot \begin{bmatrix} -\sin(\theta_{3}) \\ \cos(\theta_{3}) \end{bmatrix}$$
(5.12)

Tangentvektoren t er givet ved den partielle afledte af d mht.  $\theta_3$ .

$$\mathbf{t} = \frac{\partial}{\partial \theta_3} \mathbf{d} = \mathbf{A}_2 R_2 \begin{bmatrix} -\cos(\theta_3) \\ -\sin(\theta_3) \end{bmatrix}$$
(5.13)

For at kontakten mellem tandkransen og tandhjulet et opfyldt skal følgende være gældende.  $\Phi_2 = \mathbf{d}^T \mathbf{d} - R_1^2 = 0$  (5.14)

Ligeledes må rotationspunktet *A* kun have en translatorisk bevægelse langs det globale koordinatsystems y-akse.

$$\Phi_3 = \left(r_2 + \mathbf{A}_2 \mathbf{s}_2^{\mathcal{A}'}\right)^T \cdot \begin{bmatrix} 1\\ 0 \end{bmatrix} = 0$$
(5.15)

Tandhjulet låses translatorisk i x- og y-retningen ved bindingerne  $\Phi_{4.5}$ .

$$\boldsymbol{\Phi}_{4,5} = \mathbf{r}_1 + \mathbf{s}_1^G = \mathbf{0} \tag{5.16}$$

Driverligning  $\Phi_6^d$  for tandhjulets rotation er givet ved.

$$\Phi_6^d = \theta_1 - \dot{\theta}_1 \cdot t = 0 \tag{5.17}$$

#### Løsning af det kinematisk system

Tvangsbindinger giver et ulineært ligningssystem som kan skrives på formen.

$$\boldsymbol{\Phi} = \boldsymbol{\Phi}(\mathbf{q}) = 0 \tag{5.18}$$

$$\mathbf{\Phi}^d = \mathbf{\Phi}(\mathbf{q}, t) = 0 \tag{5.19}$$

Hvor  $\Phi^d$  indeholder driverligningen og  $\Phi$  indeholder de resterende fem ligninger. De seks ubekendte, i koordinatvektoren **q** skal til ethvert tidspunkt, *t*, overholde tvangsbindingerne. Ligninger løses, for hvert tidsstep, ved hjælp af Newton-Raphton metoden givet ved (Nikravesh, 1988 s. 67).

$$\mathbf{q}^{t+1} = \mathbf{q}^{t} - \mathbf{\Phi}_{\mathbf{q}}^{-1}\left(\mathbf{q}^{t}\right) \mathbf{\Phi}\left(\mathbf{q}^{t}\right)$$
(5.20)

Hvor  $\mathbf{q}^{t+1}$  beskriver koordinatvektoren til den næste iteration,  $\mathbf{q}^{t}$  beskriver koordinatvektoren i det nuværende gæt.  $\mathbf{\Phi}_{q}^{-1}$  er den inverterede Jacobimatice og  $\mathbf{\Phi}$  er vektoren med tvangsbindingerne. Jacobimatricen er defineret som de partielle afledte af tvangsbindingerne mht. vektoren  $\mathbf{q}$ .

$$\mathbf{\Phi}_q = \frac{d\mathbf{\Phi}}{d\mathbf{q}} \tag{5.21}$$

For hver iteration kontrolleres det, om gættet overholder tvangsbindingerne, ved at undersøge om ligningerne (5.18) og (5.19) er overholdt.

#### 5.3.2. Input til motoren

Inputtet til driverligningen tager udgangspunkt i inputtet til motoren. Dette input er givet fra FKI i form af en rampefunktion. Rampefunktionen for vippebevægelsen fra 0° til 42° og tilbage til udgangsposition er givet ved syv step, se Figur 5.8.



Figur 5.8: Rampefunktion for vippebevægelsen beskrevet ved tidsstep fra  $t_1$  til  $t_8$ .

Start og slut tiderne, for de enkelte step, samt de tilhørende accelerationer er givet i Tabel 5.2. Beregningerne af vinkelaccelerationerne er givet i Appendiks I.

t [ <i>sek</i> .]	$\dot{\omega}$ [rad/s <sup>2</sup> ]
$0.0 \le t \le 0.5$	-30.33
$0.5 < t \le 0.6$	0.00
$0.6 < t \le 0.7$	151.62
$0.7 < t \le 1.7$	0.00
$1.7 < t \le 1.8$	151.62
$1.8 < t \le 1.9$	0.00
$1.9 < t \le 2.4$	-30.33

Tabel 5.2: Tider og accelerationer for rampefunktionen.

Rampefunktionen er den samme for alle afkast, ved anvendelse af S-3000E sorter.

#### 5.3.3. Resultater

Beregningerne af vippebevægelsen er foretaget i MATLAB, hvor der er anvendt en numerisk bestemt Jacobimatricen. Der er til beregningerne anvendt et tidsstep på 0.0005 *sek*. Programmet anvendt til beregningerne kan findes på vedlagte CD-ROM i mappen *Bakkebevægelse*. På Figur 5.9 er en billedserie af den beregnede vippebevægelsen illustrere, hvor den samlede rotation og position af tandkransens koordinatsystem, i forhold til det globale koordinatsystem, er illustreret på Figur 5.10.



Figur 5.9: Tre trin af vippebevægelsen.



Figur 5.10: Tandkransens koordinatsystems position og rotation i forhold til det globale koordinatsystem.

Den beregnede vippebevægelse er udskrivet til en tekstfil, hvor der for hvert tidsstep er givet en position og rotation, af tandkransen, i forhold til det globale koordinatsystem. I tekstfilen er det globale koordinatsystem og tandkransens lokale koordinatsystem flytte op til oversiden af bakken, pga. den senere anvendelse i simuleringsprogrammet.

# 6. Simuleringsprogram

Dette kapitel giver en indledende beskrivelse af det konstruerede simuleringsprogram. Indledningsvis argumenteres for valget af den metode, der ligger til grund for simuleringsprogrammet. Herefter gives en kort beskrivelse af simuleringsprogrammet, hvorefter den i programmet anvendte kinematik beskrives.

Det er i rapporten valgt ikke at fremstille kildekoden for simuleringsprogrammet. I forbindelse med beskrivelsen af den anvendte teori, henvises der i stedet til vedlagte kildekode. I enkelte tilfælde suppleres teorien med et pseudokodeeksempel.

# 6.1. Valg af simuleringsmetode

Ud fra problemformuleringen ønskes et generelt simuleringsværktøj, hvor inputtet skal kunne varierer alt efter hvilke legemer, slisker og sorteringssystem der anvendes. Da simuleringen skal give et dynamisk output i form af emnets bevægelse er det valgt at simuleringen skal omfatte teorien for stivlegemedynamik.

Stivlegemedynamik tager udgangspunkt i legemer der er uendelig stive, som enten bevæger sig frit i rummet eller er bundet sammen af en række tvangsbindinger. Med antagelsen om uendelige stive legemer kan ethvert punkt i legemet beskrives ud fra ét referencekoordinatsystem.

Valget af stivlegemedynamik medfører at legemernes deformation ikke medregnes, hvilket giver anledning til fejl. Dette kan ifølge (Meriam, et al., 2003) accepteres, hvis ændringen af legemets geometri er meget mindre end legemets samlede flytning.

Et alternativt til stivlegemedynamik er Finit Element (FE), hvor fleksibiliteten af legemerne inddrages. Anvendelsen af FE giver anledning til et mere kompliceret simuleringsprogram, hvor beregningerne vil blive tunge pga. store ligningssystemer.

## 6.2. Beskrivelse af simuleringsprogram

Dette afsnit giver en kort gennemgang af den teori der ligger til grund for det konstruerede simuleringsprogrammet, hvorefter den nærmere teori er beskrevet i de efterfølgende kapitler.

Generelt kan opbygningen af simuleringsprogrammet inddeles i to dele i form af en dynamisk og en kontaktbestemmende del, se Figur 6.1.



*Figur 6.1: Generel inddeling af indholdet i simuleringsprogrammet i form af en dynamik og en kontaktbestemmende del.* 

Dynamikken tager udgangspunkt i beskrivelsen af legemernes orientering, position og bevægelse i rummet, mens kontaktbestemmelsen giver den geometriske sammenhæng mellem legemerne. Under beregning af afkastforløbet er der en datastrøm mellem de to blokke, hvor kontaktbestemmelsen anvender oplysninger som position og orientering af legemerne. Dynamikken får informationer om evt. kontakt mellem legemerne.

I dynamikdelen bestemmes legemernes position og hastighed ud fra bevægelsesligningerne. Bevægelsen af legemerne er bestem ud fra kontakten med andre legemer samt tyngdeaccelerationen. Ved kontakt påtrykkes legemerne en kontakt-og en friktionskraft.

## 6.3. Kinematik

Dette afsnit giver en beskrivelse af den, i simuleringsprogrammet, anvendte kinematik. I Appendiks II er en nærmere beskrivelse givet.

## 6.3.1. Definitioner

#### Stivlegeme

Teorien for stivelegemer tager udgangspunkt i partikeldynamik, i form af Newtons love, samt antagelsen om at afstanden mellem samtlige partikler, i et stift legeme, forbliver den samme til enhver tid (Shabana, 1998 s. 10). I følgende tekst anvendes betegnelsen legeme underforstået, at det er et stivlegeme.

#### Koordinatsystemer

I det følgende anvendes betegnelsen global koordinatsystem som et fast koordinatsystem (reference koordinatsystem) bestående af akserne *x*, *y* og *z*, hvis position ikke ændres. Ligeledes anvendes betegnelsen lokalt koordinatsystem som et koordinatsystem, placeret i hvert legeme, med koordinatakserne  $\xi$ ,  $\eta$  og  $\zeta$ .
# 6.3.2. Koordinattransformation

Punktet *P* i legemet *i* kan mht. legemets lokale koordinatsystem, beskrives ved vektoren  $\mathbf{s}^{P'}$ , se Figur 6.2.



*Figur 6.2: Legeme i med punktet P, beskrevet ved vektoren*  $\mathbf{r}^{P}$ *.* 

På baggrund af definitionen af stive legemer kan punktet *P*'s placering beskrives i det globale koordinatsystem ved.

$$\mathbf{r}^{P} = \mathbf{r} + \mathbf{A}\mathbf{s}^{P'} \tag{6.1}$$

Hvor vektoren **r** beskrive legemets placering mht. det globale koordinatsystem. Matricen **A** indeholder ni retningscossinuser som beskriver orienteringen af det lokale koordinatsystems akser  $\xi \eta \zeta$  i forhold til det globale koordinatsystems xyz.

# 6.3.3. Euler parametre

I simuleringsprogrammet er det valgt at anvende Euler parametre til at bestemme orienteringen af legemernes lokale koordinatsystem i forhold til det globale koordinatsystem. Euler parametre tager udgangspunkt i Eulers teorem, der siger at; et legemes flytning, hvor ét punkt er fastholdt, kan beskrives ved en rotation om én akse. Figur 6.3 illustrerer koordinatsystemerne *xyz* og  $\xi\eta\zeta$ , hvis orientering, i forhold til hinanden, kan beskrives ved rotation $\phi$ omkring én akse givet ved enhedsvektoren  $\vec{u}$ .



Figur 6.3: Sammenhængen mellem to koordinatsystemer kan beskrives som rotation  $\phi$  omkring vektoren u.

På baggrund af vektoren  $\vec{u}$  og rotationen  $\phi$  er Euler parametrene  $e_0, e_1, e_2, e_3$  defineret ved.

$$e_0 = \cos\frac{\phi}{2} \tag{6.2}$$

$$\vec{e} = \vec{u}\sin\frac{\phi}{2} \tag{6.3}$$

Hvor

$$\vec{e} = [e_1, e_2, e_3]^T$$
 (6.4)

Ud fra de fire Euler parametre er transformationsmatricen **A** givet ved (Nikravesh, 1988 s. 159).

$$\mathbf{A} = \left(2e_0^2 - 1\right)\mathbf{I} + 2\left(\mathbf{e}\mathbf{e}^T + e_0\tilde{\mathbf{e}}\right)$$
(6.5)

eller

$$\mathbf{A} = 2 \begin{bmatrix} e_0^2 + e_1^2 - \frac{1}{2} & e_1 e_2 - e_0 e_3 & e_1 e_3 - e_0 e_2 \\ e_1 e_2 - e_0 e_3 & e_0^2 + e_2^2 - \frac{1}{2} & e_2 e_3 - e_0 e_1 \\ e_1 e_3 - e_0 e_2 & e_2 e_3 - e_0 e_1 & e_0^2 + e_3^2 - \frac{1}{2} \end{bmatrix}$$

Hvor  $\tilde{\mathbf{e}}$  er skævmatricen af vektoren  $\mathbf{e}$ , defineret ved.

$$\tilde{\mathbf{e}} = \begin{bmatrix} 0 & -e_3 & e_2 \\ e_3 & 0 & -e_1 \\ -e_2 & e_1 & 0 \end{bmatrix}$$
(6.6)

## 6.3.4. Vinkelhastighed

Roterer legemets koordinatsystem  $\xi \eta \zeta$  i forhold til det globale *xyz*, kan vinkelhastigheden beskrives ved vinkelhastighedsvektoren  $\vec{\omega}$ , der udgør aksen hvorom legemet roterer, samt ændringen af rotationen. Vinkelhastighedsaksen skal ikke forveksles med orienteringsrotationsaksen. Fastlåses vinkelhastigheden, vil orienteringsrotationsaksen være den akse, hvorom en rotation vil kunne gøre det lokale koordinatsystem sammenfaldende med det globale, se Figur 6.4.



Figur 6.4: Vinkelhastighedsaksen og orienteringsrotationsaksen mellem det lokale  $\xi \eta \zeta$  og globale xyz koordinatsystem.

Vektoren  $\mathbf{s}^{P'}$ er givet i det globale koordinatsystem ved

$$\mathbf{s}^p = \mathbf{A}\mathbf{s}^{p'} \tag{6.7}$$

Differentieres ligning (6.7) mht. tiden fås.

$$\dot{\mathbf{s}}^{p} = \dot{\mathbf{A}}\mathbf{s}^{p'} + \mathbf{A}\dot{\mathbf{s}}^{p'} \tag{6.8}$$

Da  $\mathbf{s}^{P'}$  er en fast vektor i det lokale koordinatsystem, gælder det at  $\mathbf{A}\dot{\mathbf{s}}^{P'} = 0$ . Det kan vises at der eksistere en sammenhæng mellem transformationsmatricen **A** og den afledte  $\dot{\mathbf{A}}$  givet ved (Nikravesh, 1988 s. 173).

$$\dot{\mathbf{A}} = \tilde{\boldsymbol{\omega}} \mathbf{A} \tag{6.9}$$

Hvor  $\omega$  er vinkelhastighedsvektoren med hensyn til globale koordinater. Ligeledes kan det vises at (Nikravesh, 1988 s. 173).

$$\dot{\mathbf{A}} = \mathbf{A}\tilde{\boldsymbol{\omega}}' \tag{6.10}$$

For et legeme der både har en vinkelhastighed samt en translation relativt til det globale koordinatsystem kan positionen af ét punkt *P* bestemmes ved.

$$\mathbf{r}^{p} = \mathbf{r} + \mathbf{A}\mathbf{s}^{P'} \tag{6.11}$$

Punktets hastighed bestemmes ved den tidsafledte af (6.11), hvilket er givet ved.

$$\dot{\mathbf{r}}^{P} = \dot{\mathbf{r}} + \mathbf{A}\tilde{\boldsymbol{\omega}}'\mathbf{s}^{P'} \tag{6.12}$$

# 6.3.5. Tidsafledte af Euler parametre

Der er i (Nikravesh, 1988 s. 173) givet følgende sammenhæng mellem vinkelhastigheden og de tidsafledte til Euler parametrene.

$$\boldsymbol{\omega}' = 2\mathbf{L}\dot{\mathbf{p}} \tag{6.13}$$

Hvor de afledte Euler parametre kan isoleres ved.

$$\dot{\mathbf{p}} = \frac{1}{2} \mathbf{L}^T \boldsymbol{\omega}' \tag{6.14}$$

Ligeledes er der givet at.

$$\boldsymbol{\omega} = 2\mathbf{G}\dot{\mathbf{p}} \tag{6.15}$$

Hvor de afledte Euler parametre kan isoleres ved.

$$\dot{\mathbf{p}} = \frac{1}{2} \mathbf{G}^T \boldsymbol{\omega} \tag{6.16}$$

G og L er defineret ved.

	$\left[-e_{1}\right]$	$e_0$	$-e_{3}$	$e_2$		$\left[-e_{1}\right]$	$e_0$	$e_3$	$-e_2$	
<b>G</b> =	$ -e_2 $	$e_3$	$e_0$	$-e_1$	L =	$ -e_2 $	$-e_{3}$	$e_0$	$e_1$	(6.17)
	$\left\lfloor -e_{3}\right\rfloor$	$-e_2$	$e_1$	$e_0$		$\left\lfloor -e_{3}\right\rfloor$	$e_2$	$-e_1$	$e_0$	

Ved løsning af bevægelsesligningerne anvendes sammenhængen mellem de afledte Euler parametre og vinkelhastigheden ligning (6.14). Dette er nærmere beskrevet i afsnit 7.1.5.

# 6.4. Inddeling af legemer

Det er valgt i simuleringsprogrammet at inddele legemerne i tre typer, i form af: *Fri, Bevæ-gelige* og *Faste* legemer. Inddelingen specificeres i inputfilen til simuleringen, som er nærmere beskrevet i afsnit 10.1. Denne inddeling er foretaget for at opnå mere simple beregninger, samt for at reducere konktaktbestemmelsen mellem legemerne. I forbindelse med kontaktbestemmelsen undersøges der ikke for kontakt mellem de bevægelige og faste legemer.

## Frie legemer

De frie legemer kan frit bevæge sig i rummet og er kun under indflydelse af tyngdeaccelerationen og reaktionskræfter, i forbindelse med kontakt. I forbindelse med simulering af afkastforløb svarer pakkerne til de frie legemer.

## Bevægelige legemer

De bevægelige legemer kan påtrykkes en foruddefineret hastighed og bevægelse. De bevægelige legemer er ikke influeres af påtrykte kræfter under simuleringen. I forbindelse med simulering af et afkastforløb svarer bakken til de bevægelige legemer.

## Faste legemer

De faste legemer defineres ved en fast position og orientering i simuleringsprogrammet. Denne position er uændret gennem hele simuleringen også selvom kræfter påtrykkes legemeterne. I forbindelse med én simulering svarer slisken til et fast legeme.

# 7. Dynamik

I dette kapitel beskrives den, i simuleringen, anvendte dynamik. I simuleringsprogrammet er de dynamiske beregninger inddelt i tre moduler som vis på Figur 7.1.



Figur 7.1: Moduler anvend i simuleringsprogrammet.

Modulerne aktiveres skiftevis i en løkke, hvor legemernes position og hastighed bestemmes til hvert tidsstep. Bevægelsesmodulet beregner accelerationen ud fra legemernes påtrykte kræfter og momenter, som bestemmes i reaktionsmodulet. I det følgende beskrives den teori der ligger til grund for de enkelte moduler. Indledningsvis beskrives bevægelsesligningerne for hhv. translationen og rotationen af legemerne, hvorefter den numeriske integration af bevægelsesligningerne forklares. Efterfølgende beskrives den valgte metode til modellering af kontakt mellem legemerne. Afslutningsvis beskrives hvordan fjeder- og dæmperkonstanterne for legemerne bestemmes samt hvorledes bakkebevægelsen er implementeret.

# 7.1. Bevægelsesligningerne

Bevægelsesligningerne beskriver sammenhængen mellem kræfter og accelerationer, hvilket giver anledning til en bevægelse af et legeme. Kinetikken tager udgangspunkt i Newtons tre love for partikler.

- 1. En partikel som ikke er påvirket af en resulterende kraft, vil enten være i hvile eller foretage en jævn retlinet bevægelse.
- 2. En partikel med massen *m*, der påvirkes af en resulterende kraft *F*, vil have en acceleration *a*, som opfylder:  $F = m \cdot a$
- 3. Hvis en partikel *a* påvirkes af en kraft fra partikel *b*, vil *b* blive påvirket med en lige stor modsat rettet kraft.

For en partikel *i*, hvis masse  $m_i$  ingen rummelig udbredelse har, er bevægelsesligningen, givet ved Newtons 2. lov.

$$m_i \ddot{\mathbf{r}}_i = \mathbf{f}_i \tag{7.1}$$

Hvor kraften  $\mathbf{f}_i$  og accelerationen  $\ddot{\mathbf{r}}_i$  er påtrykt partiklen.

#### 7.1.1. Bevægelsesligningen for et legeme

I det følgende udvides kinetikken fra én partikel, til flere interagerende partikler, der tilsammen danner et legeme. I hht. definitionen på stivelegemer forbliver positionen af partiklerne, relativt til hinanden, uændret. Med bevægelsesligningerne for samtlige partikler i legemet, kan det vises at legemets bevægelsesligning er givet ved (Nikravesh, 1988 s. 211).

$$m\ddot{\mathbf{r}} = \mathbf{f} \tag{7.2}$$

Hvor kraften **f** er påført i legemets massecenter, hvormed partiklernes interne kræfter ophæver hinanden.

#### 7.1.2. Rotationsbevægelsesligningen

I modsætning til en partikel har et legeme en masseudbredelse, hvormed kræfter der ikke er påført i massecenteret ligeledes påtrykker legemet et moment. Momentet **n** er givet ved krydsproduktet mellem den påtrykte kraft og afstandsvektoren **s** fra massecenteret ud til angrebspunktet.

$$\mathbf{n} = \tilde{\mathbf{s}}\mathbf{f} \tag{7.3}$$

Med udgangspunkt i en partikel, forskudt fra massecentrum, påtrykt en kraft, kan det vises at rotationsbevægelsesligningen er givet ved (Nikravesh, 1988 s. 217).

$$\mathbf{J}\dot{\boldsymbol{\omega}} + \tilde{\boldsymbol{\omega}}\mathbf{J}\boldsymbol{\omega} = \mathbf{n} \tag{7.4}$$

Hvor  $\boldsymbol{\omega}$  og  $\dot{\boldsymbol{\omega}}$  er vinkelhastigheden og vinkelaccelerationen og **J** er masseinertimomentet givet ved.

$$\mathbf{J} = -\int_{(v)} \tilde{\mathbf{s}} \tilde{\mathbf{s}} dm \tag{7.5}$$

#### 7.1.3. Inertimoment

Masseinertimomentet ligning (7.5) er givet i globale koordinater, hvormed den er en funktion af positionen. Hvis legemets position ændrer sig i forhold til det globale koordinatsystem skal masseinertimomentet opdateres. For at undgå unødvendigt komplicerede beregninger, anvendes i stedet masseinertimomentet J', givet i legemets lokale koordinater. Ved anvendelse af lokale koordinater opnås et masseinertimoment der er invariant med tiden, hvormed beregningen kun skal udføres én gang. Masseinertimomentet i det lokale koordinatsystem er defineret ved.

$$\mathbf{J}' = -\int \tilde{\mathbf{s}}' \tilde{\mathbf{s}}' dm \tag{7.6}$$

Bestemmelsen af masseinertimomenterne for legemerne i simuleringsprogrammet er beskrevet i afsnit 10.3.

#### 7.1.4. Rotationsbevægelsesligningen i lokale koordinater

Med masseinertimomentet givet i lokale koordinater foretages en omskrivning af rotationsbevægelsesligningen.

Indsættes  $\tilde{\mathbf{s}} = \mathbf{A}\tilde{\mathbf{s}}'\mathbf{A}^T$  i ligning (7.5) kan masseinertimomentet, i globale koordinater, skrives som

$$\mathbf{J} = -\int \mathbf{A}\tilde{\mathbf{s}}'\mathbf{A}^{T}\mathbf{A}\tilde{\mathbf{s}}'\mathbf{A}^{T}dm$$
(7.7)

Da  $\mathbf{A}^{T}\mathbf{A} = \mathbf{I}$ , reduceres (7.7) til.

$$\mathbf{J} = \mathbf{A} \left( -\int \tilde{\mathbf{s}}' \tilde{\mathbf{s}}' dm \right) \mathbf{A}$$
  
$$\mathbf{J} = \mathbf{A} \mathbf{J}' \mathbf{A}^{T}$$
  
(7.8)

Med sammenhængende  $\mathbf{n} = \mathbf{An}'$ ,  $\boldsymbol{\omega} = \mathbf{A}\boldsymbol{\omega}'$  og  $\dot{\boldsymbol{\omega}} = \dot{\mathbf{A}}\boldsymbol{\omega}' + \mathbf{A}\dot{\boldsymbol{\omega}}' = \mathbf{A}\ddot{\boldsymbol{\omega}}'\boldsymbol{\omega}' + \mathbf{A}\dot{\boldsymbol{\omega}}' = \mathbf{A}\dot{\boldsymbol{\omega}}'$ , givet i (Nikravesh, 1988 s. 219), kan ligning (7.4) formuleres i lokale koordinater ved.

$$\mathbf{A}\mathbf{J}'\dot{\boldsymbol{\omega}}' + \mathbf{A}\tilde{\boldsymbol{\omega}}'\mathbf{J}'\boldsymbol{\omega}' = \mathbf{A}\mathbf{n}'$$

Ved at multiplicere med  $\mathbf{A}^{T}$ , opnås rotationsbevægelsesligningen givet i det lokale koordinatsystem.

$$\mathbf{J}'\dot{\boldsymbol{\omega}}' + \tilde{\boldsymbol{\omega}}'\mathbf{J}'\boldsymbol{\omega}' = \mathbf{n}' \tag{7.10}$$

For et ikke-tvangsbundet legeme i rummet, der kun er i kontakt med andre legemer gennem kraftpåvirkninger, kan legemets fysiske opførsel beskrives ved bevægelsesligningerne (7.2) og (7.10).

#### 7.1.5. Implementering af bevægelsesligningerne

I simuleringsprogrammet er bevægelsesligningerne givet i subrutinen *CalcAcc* i modulet *Dynamic*. I *CalcAcc* bestemmes for hvert legeme de afledte af positionen og hastigheden, hvilket anvendes i forbindelse med den numeriske integration af bevægelsesligningerne, beskrevet i afsnit 7.2. Til bestemmelse at de afledte positioner og hastigheder anvendes bevægelsesligningerne (7.2) og (7.10) samt sammenhængen mellem legemets lokale vinkelhastighed og de afledte Euler parametre, fra ligning 6.14. Eftersom legemerne er uafhængige af hinanden beregnes bevægelsesligningerne individuelt. Implementeringen af bevægelsesligningerne er beskrevet med følgende pseudokode.

Pseudokode:								
$\mathbf{q}_{\mathrm{in}}$	Matrix med positioner og hastigheder for hvert legeme							
n	Antal legemer							
G	Global array							
$\mathbf{q}_{out}$	Matrix med hastigheder og accelerationer for hvert legeme							
r	Positionsvektor							
р	Euler parametre							
ω′	Lokalvinkelhastighed							
т	Masse							
J′	Masseinertimoment							
n'	Lokalmomentvektor							
f	Kraftvektor							
function $\mathbf{q}_{out}$	$=f(\mathbf{q}_{in},\mathbf{G})$							
$\mathbf{q}_{in} = \left[\mathbf{r}^{T}, \mathbf{p}^{T}, \dot{\mathbf{r}}^{T}, \boldsymbol{\omega}^{\prime T}\right]_{n}$								
doi=1	l, <i>n</i>							
	$\ddot{\mathbf{r}}_i = \frac{1}{m_i} \cdot \mathbf{f}_i$							
	$\dot{\boldsymbol{\omega}}_i' = \mathbf{J}_i'^{-1}\mathbf{n}_i' - \mathbf{J}_i'^{-1}\widetilde{\boldsymbol{\omega}}_i'\mathbf{J}_i'\boldsymbol{\omega}_i'$							
	$\dot{\mathbf{p}}_i = 0.5 \cdot \mathbf{L}_i^T \mathbf{\omega}_i'$							
end do	0							
$\mathbf{q}_{out} =$	$\begin{bmatrix} \dot{\mathbf{r}}^T, \dot{\mathbf{p}}^T, \ddot{\mathbf{r}}^T, \dot{\boldsymbol{\omega}}'^T \end{bmatrix}_n$							
end function								

Det globale array **G** indeholder flere oplysninger om bl.a., reaktionskraften **f**, momentet **n**', massen *m*, samt den inverterede og ikke inverterede masseinertimoment  $\mathbf{J}'^{-1}$  og  $\mathbf{J}'$  for hvert legeme. Transformationsmatricen **L** kaldes som en funktion fra modulet *FuncCall*.

# 7.2. Løsning af bevægelsesligningerne

Bevægelsesligningerne (7.2) og (7.10) er et sæt 2.ordens ordinære differentialligninger, hvis løsning bestemmer legemets hastighed og position med hensyn til tiden. Det ideelle ville være at løse bevægelsesligningerne på lukket form. Dette er dog ikke praktisk muligt pga. bevægelsesligningernes kompleksitet mht. kraft og momentpåvirkningerne som afhænger af den, i simuleringen, implementerede geometri. Derfor anvendes en numerisk løser til løsning af bevægelsesligningerne.

De numeriske løsere har generelt en høj fleksibilitet, idet ændringer i differentialligningen, ikke har nogen indflydelse på, hvorvidt ligningen kan løses. Derimod er nøjagtigheden af resultatet afhængig af, hvilken numerisk løser der anvendes og hvor lang en strækning der integreres over.

#### 7.2.1. 2.ordens differentialligninger

Én 2.ordens differentialligning kan opdeles i to 1.ordens ligninger ved (Kreyszig, 1999 s. 957).

$$\ddot{y}_{1} = f(t, y_{1}, \dot{y}_{1}) 
\downarrow 
\dot{y}_{1} = y_{2} 
\dot{y}_{2} = f(t, y_{1}, y_{2})$$
(7.11)

På tilsvarende vis kan de to bevægelsesligninger (7.2) og (7.10) opdeles i to 1.ordens differentialligninger som individuel kan løses numerisk.

#### 7.2.2. 4.ordens Runge-Kutta

4.ordens Runge-Kutta er en meget brugt metode, der tager udgangspunkt i et begyndelsesværdiproblem, til løsning af første ordens differentialligninger (Press, et al., 1997 s. 702). Runge-Kutta er en metode der propagerer en løsning, over et interval, ved at kombinere information fra fire Euler lignende skridt, der evalueres med en højere ordens Taylorrække (Press, et al., 1997 s. 702). Med udgangspunkt i et begyndelsesværdiproblem, hvor værdierne for  $y_n$  er kendt til tiden  $t_n$  løser Runge-Kutta differentialligningen til tiden  $t_{n+1}$  således at  $y_{n+1}$  bestemmes.

Afstanden mellem punktet  $y_{n+1}$  til tiden  $t_{n+1}$  og punktet  $y_n$  til tiden  $t_n$ , bestemmes ved integralet af 1.ordens differentialligningen  $\dot{y}(t)$ , hvor *n* angiver step nummeret.

$$y(t_{n+1}) - y(t_n) = \int_{t_n}^{t_{n+1}} \dot{y}(t) dt$$
(7.12)

Anvendes Simpson's regel, der tager udgangspunkt i Lagrange kvadratiske interpolation (Kreyszig, 1999 s. 872), kan ligning (7.12) skrives som.

$$y(t_{n+1}) - y(t_n) \approx \frac{h}{6} \left[ \dot{y}(t_n) + 4\dot{y}\left(t_n + \frac{h}{2}\right) + \dot{y}(t_{n+1}) \right]$$
 (7.13)

Hvor *h* er tidssteppet givet ved  $t_{n+1} - t_n$ . Ligning (7.13) omskrives til.

$$y(t_{n+1}) \approx y(t_n) + \frac{h}{6} \left[ \dot{y}(t_n) + 2\dot{y}\left(t_n + \frac{h}{2}\right) + 2\dot{y}\left(t_n + \frac{h}{2}\right) + \dot{y}(t_{n+1}) \right]$$
(7.14)

Af de fire led i ligning (7.14), hvor  $\dot{y}$  indgår, er det kun den afledte til tiden  $t_n$  der er kendt, hvilket formuleres ved følgende udtryk.

$$k_1 = f\left(t_n, y_n\right) \tag{7.15}$$

Eftersom den afledte til tiden  $t_n + \frac{h}{2}$  ikke er kendt, estimeres denne ud fra den afledte til tiden  $t_n$ , dvs.  $k_1$ , ved at lægge afstanden  $\frac{1}{2}hk_1$  til  $y_n$ .

$$k_2 = f\left(t_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_1\right)$$
(7.16)

På tilsvarende vis estimeres det tredje led, ved hjælp af den afledte til tiden  $t_n + \frac{h}{2}$ , ud fra den foregående  $k_2$ .

$$k_3 = f\left(t_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_2\right)$$
(7.17)

Tilsvarende estimeres den afledet til tiden  $t_{n+1}$ ud fra  $k_3$ .

$$k_4 = f(t_n + h, y_n + hk_3)$$
(7.18)

På Figur 7.2 er de fire step  $k_1$ ,  $k_2$ ,  $k_3$  og  $k_4$  illustreret.



Figur 7.2: For 4.ordens Runge-Kutta bestemmes de afledte fire gange for hvert tidsstep. En gang i startpunktet  $y_n$ , to gange i midten og en gang i slutpunktet  $y_{n+1}$ . Hvert punkt estimers ud fra forrige punkts hældning. Af de fire afledte bestemmes den endelige funktionsværdi.

Ligning (7.14) kan ved indsættelse af (7.15), (7.16), (7.17) og (7.18) skrives som.

$$y_{n+1} \approx y_n + \frac{h}{6} (k_1 + 2k_2 + 2k_3 + k_4)$$
 (7.19)

Det kan vises at Runge-Kutta er  $h^4$  ordens nøjagtig (Kreyszig, 1999 s. 951), hvormed (7.19) kan skrives som.

$$y_{n+1} = y_n + \frac{h}{6} (k_1 + 2k_2 + 2k_3 + k_4) + O(h^4)$$
(7.20)

Da ligning (7.20) gælder for samtlige differentialligninger kan den formuleres på algebraiskform ved.

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \frac{h}{6} \left( \mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4 \right) + O(h^4)$$
(7.21)

Hvormed *m*-differentialligninger løses.

En ulempe ved Runge-Kutta i forhold til andre tilsvarende numeriske løsere, er at differentialligningen skal evalueres fire gange for hver step, hvilket gør beregningerne tunge. Dette opvejes af at den er let at implementere samt at det er en stabil løser (Press, et al., 1997 s. 702).

Ved løsningen af bevægelsesligningerne anvendes ligning (7.11) og (7.21).

Størrelsen af tidssteppet h, der er anvendt i simuleringsprogrammet, er nærmere beskrevet i afsnit 7.4.3.

# 7.2.3. Implementering af Numerisk integration

Den numeriske integration er i simuleringsprogrammet givet ved subrutinen rk4 i modulet *NumInt*. Rutinen kan betragtes som kernen i simuleringsprogrammet, da det er her tiden styres og det er herfra de resterende rutiner til simuleringsberegningerne kaldes. Rutinen tager udgangspunkt i en løkke, hvor tidssteppet kører indtil den anvendte simuleringstid opnås. For hvert tidsstep gemmes legemernes position, hastighed og acceleration i matricerne *Pos, Has* og *Acc*. Subrutinen *For\_Mom\_Calc* kaldes, hvormed alle kræfter og momenter påtrykt legemerne bestemmes i hht. legemernes pågældende positioner. Denne rutine er nærmer beskrevet i afsnit 9.7. Herefter bestemmes  $\mathbf{k}_1$ , hvor funktionen *CalcAcc* kaldes. Denne proces gentages for bestemmelsen af  $\mathbf{k}_2$ ,  $\mathbf{k}_3$  og  $\mathbf{k}_4$ , hvor legemernes reaktionskræfter og momenter, opdateres for hver iteration. Derefter bestemmes legemernes nye position og hastighed  $\mathbf{q}_{n+1}$ . Følgende pseudokode ligger til grund for implementeringen af *rk4*.

Pseudokode:	
G	Global array
q	Matrix med position og hastighed for samtlige legemer
n	Stepnummer
t	Simuleringstid
$\Delta t$	Tidsstep
Pos	Positionsmatrice for hvert tidsstep for alle legemer
Has	Hastighedsmatrice for hvert tidsstep for alle legemer
Acc	Accelerationsmatrice for hvert tidsstep for alle legemer
subroutine rk4	$4(\mathbf{G},\mathbf{q})$
$n = \frac{t}{\Delta t}$	
do $t = 1$	1, <i>n</i>
	$Pos(t,:,:) = \mathbf{q}_n(:,1:7)$
	$Has(t,:,:) = \mathbf{q}_n(:,8:13)$
	<i>call</i> For_Mom_Calc $(G, \mathbf{q}_n, \Delta t)$
	$Acc(t,:,:) = \text{CalcAcc}(t,\mathbf{q}_n,\mathbf{G})$
	$\mathbf{k}_1 = \Delta t \cdot \operatorname{CalcAcc}(t, \mathbf{q}_n, \mathbf{G})$
	:
	<i>call</i> For_Mom_Calc ( <b>G</b> , <b>q</b> <sub>n</sub> + <b>k</b> <sub>3</sub> , $\Delta t$ )
	$\mathbf{k}_{4} = \Delta t \cdot \operatorname{CalcAcc}(t + \Delta t, \mathbf{q}_{n} + \mathbf{k}_{3}(i, :), \mathbf{G})$
	$\mathbf{q}_{n+1} = \mathbf{q}_n + \frac{\Delta t}{6} \left( \mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4 \right)$
end do	
end subroutine	

# 7.3. Kontaktmodel

Kontakt mellem to legemer er et kompleks fænomen, der involverer lokal deformation, akustisk bølgeudbredelse, revnevækst, energitab ved intern friktion, plastisk deformation osv. Der eksister ingen komplet og elegant formulering til at beskrive disse fænomener, hvorfor alle simuleringsmodeller anvender en idealiseret approksimation af den sande opførsel (Goyal, et al., 1994 I).

Der findes flere forskellige kontaktmodeller, som hver især har styrker og svagheder. Nogle af modellerne lægger vægt på fysisk realisme og andre på beregningshastigheder. De to mest anvendte modeller er: Strafmetoden og den impulsbaserede metode. I det følgende givers en kort beskrivelse af de to simuleringsmodeller.

#### Strafmetoden

Strafmetoden tager udgangspunkt i en relaksation af de stive legemer, ved at tillade lokale deformationer, hvormed en lokal strafkraft kan bestemmes ud fra deformationen (Goyal, et al., 1994 I). Reaktionskraften påtrykkes begge legemer modsatrettet hinanden. I de fleste tilfælde beregnes strafkraften ud fra Hooks lov for fjedre, hvor den lokale deformation svarer til en fjederdeformation.

En af problematikkerne ved anvendelse af strafmetoden er at deformationen og kontaktpunktet skal bestemmes, hvilket ofte medfører tunge kontaktalgoritmer. Ligeledes er numerisk stabilitet et problem, hvor kraftige sammenstød mellem legemer eller meget høje fjederstivheder kan medfører stive differentialligninger (Erleben, 2004), (Goyal, et al., 1994 I). For at opnå numerisk stabilitet kræver det ofte små tidsstep, hvilket gør beregningstiden stor. Strafmetoden egner sig både til kollision og statisk kontakt mellem legemer (Goyal, et al., 1994 I).

#### Den impulsbaserede metode

Den impulsbaserede metode simulerer alle fysiske interaktioner mellem legemer som en kollisionsimpuls (Erleben, 2004). Impulsmetoden tager udgangspunkt i relationen mellem en kollisionskraft og ændringen i bevægelsesmængden givet ved.

$$\int_{0}^{t} \mathbf{f} \cdot dt = \mathbf{P}(t) - \mathbf{P}(0)$$
(7.22)

Ved kollision mellem to 100 % stive legemer bliver kraftpåvirkningen uendelig stor, i et uendelig lille tidsrum. Derfor anvendes kraften ikke til beskrivelse af reaktionen, men i stedet anvendes ligning (7.22) i et subdomæne. Ud fra Stronge's hypotese er det heraf muligt at finde kollisionsopførslen ved at bestemme legemernes reaktionshastighed, ud fra den relative kollisionshastighed (Mirtich, 1996).

I modsætning til stafmetoden kræver den impulsbaserede metoden ingen lokal deformation, hvormed kontaktalgoritmen er simpleres. Statisk kontakt er ved den impulsbaseret metode moduleret som en serie af små microkollisioner, der foregår ved en meget høj frekvens. Dette giver anledning til at legemerne aldrig opnår fuldstændig stilstand (Erleben, 2004). Impulsbaseret kontakt er beregningsteknisk effektiv, for systemer som har mange legemer, der bevæger sig med høj hastighed og er derfor et godt valg for realtids simuleringer.

# 7.3.1. Valg af simuleringsmetode

Det er valgt at anvende stafmetoden, eftersom den impulsbaserede metode ikke giver mulighed for en direkte implementering af de valgte friktionsmodeller, beskrevet i kapitel 8. Dette skyldes at impulsmetoden tager udgangspunkt i en energiformulering, hvorimod strafmetoden tager udgangspunkt i kontaktnormalkræfterne. Ligeledes er det ikke muligt, med den impulsbaseret metode, at opnår en fuldstændig statisk kontakt mellem legemerne. Anvendelsen af strafmetoden giver anledning til en længere beregningstid, sammenlignet med den impulsbaserede, men eftersom der ikke er krav om realtids simuleringer er det af mindre betydning. Det tilstræbes dog generelt at reducere beregningstiden.

# 7.3.2. Anvendelse af strafmetoden

Ved kollision mellem to legemer er der en periode med deformation, hvor elastisk energi er ophobet i legemet. Denne periode er efterfulgt af en restitutionsperiode, hvor noget af denne energi bliver returneret som kinetisk energi. Ved en friktionsløs kontakt mellem to legemer moduleres dette ved at påtrykke legemerne to lige store modsatrettede kræfter. Disse kræfter påtrykkes i samme retning som enhedsnormalvektoren til tangentplanet, udspændt af kon-taktpunktet, se Figur 7.3 (Meriam, et al., 2003 s. 215). Indføres friktionen mellem legemerne, vil der ligeledes være to modsatrettede friktionskræfter, parallelt med tangentplanet, modsatrettet den relative hastighed i kontaktpunktet.



Figur 7.3: Ved en kollision mellem to legemer påtrykkes legemerne, i kontaktpunktet, to lige store modsatrettede kræfter. Har legemerne en relativ hastighed i forhold til hinanden påtrykkes legemerne ligeledes en friktionskraft.

Samtlige legemer i simuleringen tildeles en fjederstivhed, hvormed strafkræfterne mellem legemerne bestemmes vha. et fjedermassesystem i kontaktpunktet. Eftersom fjederkraften afhænger af en deformation, relakseres de stive legemer ved at dække overfladen med et tyndt masseløst lag, kontaktlag. Det masseløse lag lægges uden på den oprindelige geometri, hvorefter det stive legeme reduceres, i størrelse, således at kontaktlaget har samme tykkelse overalt, se Figur 7.4. Ved en kollision mellem legemerne deformeres kontaktlaget lokalt, hvilket giver en deformationen af fjedersystemet. Da kollision mellem legemer giver et energitab i form af intern friktion, lydudbredelse, mm., indføres en viskøs dæmper. Størrelsen af kontaktlagets tykkelse, samt fjeder- og dæmperkonstanterne er beskrevet i afsnit 7.4.



Figur 7.4:Kontakt mellem legemerne i og j, hvor de stiplede linjer udgør det masseløse kontaktlag. Kontakten mellem legemerne moduleres som et fjedermassesystem.

Figur 7.5 viser legemet i og j, der til tidspunktet t, er i kontakt, i punktet P. Strafkræfterne normalt til tangentplanet bestemmes ved at summere fjeder- og dæmperkræfterne.

$$\mathbf{f}_{i,w} = \mathbf{f}_{i,k} + \mathbf{f}_{i,c}$$

$$\mathbf{f}_{j,w} = \mathbf{f}_{j,k} + \mathbf{f}_{j,c}$$

$$(7.23)$$

Hvor indeks k angiver fjederkraften og indeks c er dæmperkraften. Kontaktpunktet P er defineret ud fra den korteste afstand mellem legemernes stive kerner, svarende til afstanden l, se Figur 7.5.



Figur 7.5:Legemerne i og j, i kontakt, i punktet P. Den korteste afstand mellem de stive legemer, l giver en lokal deformation  $\Delta u$ .

Kontakten mellem *i* og *j* medfører en lokal deformation, illustreret ved en penetrering mellem de to legemers kontaktlag. Har legemerne et kontaktlag med tykkelsen  $\delta_i$  og  $\delta_i$ , er den lokale deformation mellem legemerne givet ved.

$$\Delta u = \left(\delta_i + \delta_j\right) - l \tag{7.24}$$

Fjederkræfterne påtrykt i kontaktpunktet, normalt til tangentplanet, bestemmes ud fra den lokale deformation og den ækvivalente fjederstivhed.

$$\mathbf{f}_{i,k} = -\mathbf{f}_{j,k} = \mathbf{u}_{\mathbf{w}} \left( k_{Eq} \cdot \Delta u \right) \tag{7.25}$$

Hvor  $\mathbf{u}_{\mathbf{w}}$  er enhedsnormalvektoren til tangentplanet i kontaktpunktet. Den ækvivalente fjederstivhed  $k_{Eq}$  bestemmes ud fra legemernes fjederkonstanter.

$$k_{Eq} = \frac{k_i \cdot k_j}{k_i + k_j} \tag{7.26}$$

Dæmperkræfterne i kontaktpunktet bestemmes ved den ækvivalente dæmperkonstant multipliceret med den relative hastighed,  $\dot{\mathbf{r}}_{P,w}$ . Hastigheden bestemmes ved at projicere legemernes relative hastighed ind i normalretningen til tangentplanet.

$$\mathbf{f}_{i,c} = -\mathbf{f}_{j,c} = \mathbf{u}_{\mathbf{w}} \left( c_{Eq} \cdot \left| \dot{\mathbf{r}}_{P,w} \right| \right)$$
(7.27)

Hvor den ækvivalente dæmperkonstant bestemmes ved

$$c_{Eq} = \frac{c_i \cdot c_j}{c_i + c_j} \tag{7.28}$$

Legemernes relative hastighed bestemmes ved forskellen mellem legemernes hastighed i kontaktpunktet.

$$\dot{\mathbf{r}}_{p} = \dot{\mathbf{r}}_{p_{i}} - \dot{\mathbf{r}}_{p_{j}}$$
$$= \dot{\mathbf{r}}_{i} + \tilde{\boldsymbol{\omega}}_{i} \mathbf{s}_{i} - \dot{\mathbf{r}}_{i} - \tilde{\boldsymbol{\omega}}_{i} \mathbf{s}_{i}$$
(7.29)

Hvor vektorerne  $\mathbf{s}_i$  og  $\mathbf{s}_j$  er afstanden fra legemernes koordinatsystemer til kontaktpunktet, i lokale koordinater, se Figur 7.5. Den projicerede hastighed bestemmes ved.

$$\dot{\mathbf{r}}_{P,w} = \frac{\mathbf{W} \cdot \dot{\mathbf{r}}_{P}}{\left|\mathbf{W}\right|^{2}} \mathbf{W}$$
(7.30)

Friktionskræfterne  $\mathbf{f}_{i,f}$  og  $\mathbf{f}_{j,f}$  afhænger af den relative hastighedsforskel i tangentplanet, normalkraften, samt legemernes overflader. Den relative hastighed i tangentplanet kan bestemmes ved.

$$\dot{\mathbf{r}}_{P,t} = \dot{\mathbf{r}}_P - \dot{\mathbf{r}}_{P,w} \tag{7.31}$$

Bestemmelsen af friktionskræfterne er nærmere beskrevet i kapitel 8. Kontakt- og friktionskræfterne giver et moment omkring legemernes massecenter i globale koordinater. Da rotationsbevægelsesligningerne er givet i lokale koordinater, multipliceres momentet den transponerede transformationsmatrice.

$$\mathbf{n}_{i}' = \mathbf{A}^{T} \tilde{\mathbf{s}}_{i} \left( \mathbf{f}_{i,w} + \mathbf{f}_{i,f} \right)$$
  
$$\mathbf{n}_{j}' = \mathbf{A}^{T} \tilde{\mathbf{s}}_{j} \left( \mathbf{f}_{j,w} + \mathbf{f}_{j,f} \right)$$
  
(7.32)

Er et legeme udsat for flere forskellige kontakter, på samme tid, bestemmes kontaktkræfterne individuelt, hvorefter momenter og kræfter, påtrykt det enkelte legeme, summeres.

#### Implementering i simuleringsprogram

Ovenstående ligninger er implementeret i simuleringsprogrammet i forbindelse med kontaktalgoritmerne og er nærmere beskrevet i afsnit 9.7.

# 7.4. Fjeder og dæmperkonstanter samt tidsstep

Bestemmelsen af fjeder og dæmperkonstanterne samt tidssteppet, hænger sammen og har en betydning for den samlede performance af simuleringsprogrammet. Formålet med tidsstepsestimeringen er ønsket om at løse simuleringen med så stort et tidsstep som muligt og samtidigt undgå stive differentielligninger. Eftersom tidssteppet i høj grad afhænger af fjeder og dæmperkonstanterne, bestemmes disse indledningsvis. De påførte fjedre og dæmpere har til formål at beskrive den lokale deformation, legemerne udsættes for, under kontakt. Ved en reel situation vil den lokale deformation afhænge af, hvor og hvordan legemerne kollidere. Dette betyder at deformationen for et legeme i kontakt, vil variere over emnet. Eftersom det ikke er muligt at lave en entydig beskrivelse af stivheden i det eksakte kontaktpunkt, antages det at stivheden er konstant over hele emnet. Til estimering af fjeder og dæmperkonstanterne, findes der to udbredte metoder. Den ene metode beskrevet af (Hippman, 2004), anvender materialeparametre som elasticitetsmodulet og poissons forhold. Den anden metode beskrevet af (Goyal, et al., 1994), anvender et energiudtryk som f.eks. den maksimale kinetiske energi og restitutionskoefficienten. Det er valgt at anvende den sidstnævnte metode, da elasticitetsmodulet og poissons forhold ikke kan bestemmes for det anvendte materiale.

## 7.4.1. Fjederkonstant

Bestemmelse af fjederstivheden tager udgangspunkt i det legeme, der inden simuleringen starter, er i besiddelse af det største energiniveau. Energiniveauet bestemmes ud fra en sum af legemernes potentielle og kinetiske energier.

$$E_{maks} = E_{kin,tra} + E_{kin,rot} + E_{pot}$$
(7.33)

Hvor

$$E_{kin,tra} = \frac{1}{2} \cdot \boldsymbol{m} \cdot \mathbf{v}^{T} \mathbf{v}$$
(7.34)

$$E_{kin,rot} = \frac{1}{2} \boldsymbol{\omega}'^T \mathbf{J}' \boldsymbol{\omega}'$$
(7.35)

$$E_{pot} = (h \cdot m \cdot g) \cdot p \tag{7.36}$$

Da legemerne ikke nødvendigvis falder den samlede højde h, indføres vægtningsfaktoren p, som er et tal mellem nul og et. Indledningsvis er p fastsat til 0.2 ud fra forventningen om at én afkastet pakke ikke falder mere end 20 %, af den samlede starthøjde h, før den kommer i kontakt med slisken. Værdien for p kan specificeres i indputfilen.

Ved anvendelse af den maksimale energi, sikres det for samtlige legemer at der ikke forekommer fuldstændig penetrering ved kontakt.

Tykkelsen af legemernes kontaktlag,  $\delta$  bestemmes ud fra legemet med den mindste dimension. Det er valgt at fastsætte  $\delta$  til 5 % af denne. Ud fra kontaktlagets tykkelse og det maksimale energiniveau, bestemmes fjederstivheden for samtlige legemer ved.

$$E_{maks} = \frac{1}{2} k \cdot \delta^{2}$$

$$(7.37)$$

$$k = \frac{2E_{maks}}{\delta^{2}}$$

#### 7.4.2. Dæmperkonstant

Til bestemmelse af dæmperkonstanten c, er det valgt at tage udgangspunkt i den mængde energi der bliver restitueret, til kinetisk energi, efter kontakt. Det antages at alt tabet af energi foregår gennem en viskøs dæmper, både ved sammentrykning og udvidelse. Den energi der restitueres, betegnes  $E_r$  og kan bestemmes på flere forskellige måder (Goldsmith, 2001 s. 142). I det følgende er det valgt, at anvende, forskellen i kinetisk energi før og efter kontakt, hvormed restitutionskoefficienten findes ved.

$$E_{r} = \frac{\frac{1}{2}m \cdot v_{a}^{2}}{\frac{1}{2}m \cdot v_{b}^{2}} = \frac{v_{a}^{2}}{v_{b}^{2}} = e^{2}$$
(7.38)

Hvor  $v_a$  er hastigheden af legemet lige før kontakt og  $v_b$  er hastigheden lige efter. For at bestemme dæmpningskoefficienten *c* ud fra restitutionskoefficienten *e*, anvendes fjedermassesystem som vist på Figur 7.6.



Figur 7.6: Fjedermassesystemet anvendt til bestemmelse af dæmperkonstanten c.

Bevægelsesligningen for systemet vist på Figur 7.6 er givet ved.

$$m\ddot{x} + c \cdot \dot{x} + k \cdot x = 0 \tag{7.39}$$

Det antages at fjedermassesystemet er underdæmpet, hvormed den generelle løsning er givet ved (Rao, 2004 s. 142).

$$x(t) = c_1 e^{\left(-\zeta + \sqrt{\zeta^2 - 1}\right) \cdot \omega_n \cdot t} + c_2 e^{\left(-\zeta - \sqrt{\zeta^2 - 1}\right) \cdot \omega_n \cdot t}$$
(7.40)

Hvor  $\zeta$  er dæmpningsforholdet givet ved.

$$\zeta = \frac{c}{c_c} = \frac{c}{2 \cdot m \cdot \omega_n} \tag{7.41}$$

Og den naturlige frekvens er.

$$\omega_n = \sqrt{\frac{k}{m}} \tag{7.42}$$

Konstanterne  $c_1$  og  $c_2$  findes ud fra randbetingelserne givet idet kuglen slipper overfladen, dvs. x(0) = 0 og  $\dot{x}(0) = v_b$ . Dermed reduceres den generelle løsning til.

$$x(t) = e^{\{-\zeta \cdot \omega_n \cdot t\}} \cdot \left(\frac{v_b}{\omega_d} \cdot \sin(\omega_d \cdot t)\right)$$
(7.43)

Hvor  $\omega_d = \sqrt{1 - \zeta^2} \cdot \omega_n$  er den dæmpede egenfrekvens. Kontakten mellem kuglen og fladen brydes når F(t) = 0, dvs. når  $-c_n \cdot \dot{x} - k_n \cdot x = 0$ , hvormed ligning (7.43) løses med hensyn til  $\omega_d \cdot t$ .

$$\tan\left(\omega_{d}\cdot t\right) = \frac{2\cdot\zeta\cdot\sqrt{1-\zeta^{2}}}{1-2\cdot\zeta^{2}}$$
(7.44)

Ved at differentiere ligning (7.43) med hensyn til tiden fås at.

$$\dot{x}(t) = \frac{v_b}{\omega_d} \cdot \left(-\zeta \cdot \omega_n \cdot t \cdot e^{\{-\zeta \cdot \omega_n \cdot t\}} \cdot \sin(\omega_d \cdot t) + e^{\{-\zeta \cdot \omega_n \cdot t\}} \cdot \omega_d \cdot \cos(\omega_d \cdot t)\right)$$
(7.45)

Hastigheden  $v_a$  til tiden t = 0 er givet ved.

$$v_a = \dot{x}(0) = v_b \cdot e^{-\zeta \omega_n t} \tag{7.46}$$

Og da

$$E_r = \frac{v_a^2}{v_b^2} = e^{-2\cdot\zeta\cdot\omega_n\cdot t} \Longrightarrow t = -\frac{\ln(E_r)}{2\cdot\zeta\cdot\omega_n}$$
(7.47)

Indsættes (7.44) i (7.47) fås at.

$$\ln\left(E_r\right) = -\frac{4\cdot\zeta}{\sqrt{1-\zeta^2}} \cdot \tan^{-1}\left(\frac{\sqrt{1-\zeta^2}}{\zeta}\right)$$
(7.48)

/ \_\_\_\_\_ \

Af ligning (7.48) og (7.41) kan dæmperkoefficienten c isoleres, hvormed den kan bestemmes ud fra legemets fjederstivhed, masse og restitutionskoefficienten. Eftersom de bevægelige og faste legemer i simuleringsprogrammet ikke har nogen masse, bestemmes deres viskøse dæmper på baggrund af det frie legeme med den største masse. Resultatet sammen med en komplet gennemgang af beregningerne findes i Appendiks III. Restitutionskoefficienten, for pap mod pap, er bestemt til 0.5, ud fra en vurdering af andre materialers restitutionskoefficienter givet i (Goldsmith, 2001 s. 258).

#### 7.4.3. Tidsstep

Ud fra fjeder og dæmperkonstanterne udføres en estimering af tidssteppet, h, der anvendes i forbindelses med den numeriske integration. Ved estimeringen af tidssteppet bestemmes det størst mulige tidsstep, der er i stand til at beskrive svingningskarakteristikken ved kontakt. Til bestemmelse af tidsstepestimatet anvendes de beregnede fjeder og dæmperkonstanter. På baggrund heraf beregnes en periodetid for den frie svingning samt en indsvingningstid ved (Rao, 2004 s. 51, s. 143).

$$\tau = \frac{2 \cdot \pi}{\omega_n} \tag{7.49}$$

$$\tau = \frac{1}{\zeta \cdot \omega_n} \tag{7.50}$$

Pga. den numeriske diskretisering kan kontaktpunktet mellem to legemer skride (drifte) fra dets ligevægtsposition, hvilket giver en ekstra periodetid (Goyal, et al., 1994) givet ved.

$$\tau = \frac{c_1 + c_2}{k_1 + k_2} \tag{7.51}$$

Ligning (7.49), (7.50) og (7.51) beregnes for samtlige legemer i simuleringen, hvorefter der tages udgangspunkt i den mindste værdi  $\tau_{min}$  for  $\tau$ . For at opnå tilstrækkelig med punkter til at bestemme svingningskarakstatikken, ved en kollision mellem legemerne, skal tidssteppet være mindre end den mindste værdi  $\tau_{min}$ .

$$\tau_{\min} > h \tag{7.52}$$

Det har via konvergensforsøg med simuleringsprogrammet vist sig at 8 målepunkter pr. svingning giver et resultat, hvor systemet energiregnskab bliver konstant, hvormed tidssteppet er bestemt ved.

$$h = \frac{\tau_{\min}}{8} \tag{7.53}$$

I simuleringsprogrammet er bestemmelsen af fjederstivheder, dæmperkonstanter og tidsstep udført i subrutinen *Spring damper dt Calc* under modulet *InitialCalc*.

## 7.5. Bevægelige legemer

I inputfilen kan de bevægelige legemers bevægelse specificeres ved at give en hastighedsvektor og/eller en vektor for vinkelrotationen. Ud fra de specificerede hastigheder bestemmes de bevægelige legemers position vha. 4.ordens Runge-Kutta.

Ud over den retlinjede bevægelser kan legemet gives en yderligere bevægelse. Denne bevægelse specificeres i inputfilen, i form af en tekstfil, indeholdende sekvenstiden samt positioner og orienteringer af legemet.

#### 7.5.1. Implementering af bakkebevægelse

Den genererede tekstfil, beskrevet afsnit 5.3, for bakkebevægelsen omformuleres fra 2D til 3D ved at definere hvilken lokal koordinatakse bevægelsen udføres om. Tekstfilen for bakkebevægelsen indlæses i simuleringsprogrammet gennem simuleringsinputfilen. Den beregnede bakkebevægelse, i det lokale  $\xi_2 \eta_2 \zeta_2$  koordinatsystem, er givet i forhold til bakkens retlinjede bevægelse, i det lokale  $\xi_1 \eta_1 \zeta_1$  koordinatsystem, se Figur 7.7. På baggrund heraf ønskes det at bestemme bakkens nye position og orientering ift. det globale koordinatsystem.



Figur 7.7: Bakkebevægelsen er givet i forhold til bakkens retlinjede bevægelse.

Punktet *P* er givet i det lokale  $\xi_2 \eta_2 \zeta_2$  koordinatsystem og kan beskrives i det globale koordinatsystem ved.

$$\mathbf{r}^{p} = \underbrace{\mathbf{A}_{1}\mathbf{A}_{12}}_{\mathbf{A}_{2}}\mathbf{s}_{2}^{p'} + \underbrace{\mathbf{A}_{1}\mathbf{s}_{1}^{2'} + \mathbf{r}_{1}}_{\mathbf{r}_{2}}$$
(7.54)

Hvor  $A_{12}$  er transformationsmatricen mellem koordinatsystemerne  $\xi_1 \eta_1 \zeta_1$  og  $\xi_2 \eta_2 \zeta_2$ . Bakkens nye position og orientering er dermed givet ud fra vektoren  $\mathbf{r}_2$  og transformationsmatricen  $A_2$ . Implementeringen af bakkebevægelsen er foretaget i subrutinen *Move* i modulet *Dynamic*.

# 8. Friktion

Et legeme der er påvirket af en kraft eller er i bevægelse hen over en overflade, påvirkes af en fysisk reaktion i den modsatte retning, af kraften eller bevægelsen. Reaktionen skyldes kontakten mellem legemerne, hvor mikroskopiske ujævnheder i overfladerne griber ind i hinanden og på den måde bremser bevægelsen. Denne reaktion kaldes friktion og findes alle steder hvor legemer er i kontakt.

Til beskrivelse at friktionen mellem to legemer er det i simuleringsprogrammet muligt at vælge mellem tre forskellige friktionsmodeller. Friktionsmodellerne repræsenterer tre forskellige kompleksiteter i beskrivelsen af friktion

I dette afsnit beskrives indledningsvis teorien for de tre friktionsmodeller. Herefter gives en kort beskrivelse af de udførte friktionsforsøg til bestemmelse af friktionsparametrene til friktionsmodellerne. Herefter er implementeringen af friktionsmodellerne beskrevet. I afsnittet anvendes der en række udtryk forbundet med friktionsfænomener, hvor en uddybende forklaring af disse er givet i Appendiks IV.

# 8.1. Friktionsmodeller

I forskellige problemstillinger har det relevans at modulere friktion, og der findes derfor flere forskellige friktionsmodeller. Især indenfor kontrolsystemer, hvor det ønskes at kompensere for den afvigelse friktionen medfører. Eksempler på anvendelsen af friktionskompensering i kontrolsystemer er højpræcisions servomekanismer, robotter, pneumatik, hydrauliksystemer mm. Ligeledes er friktion af stor interesse indenfor geologi, hvor modeller bl.a. anvendes til at beskrive jordskælv.

Til simuleringsprogrammet er det valgt at fokusere på implementeringen af flere friktionsmodeller. Formålet med at implementere flere modeller er at finde én model der bedst mulig beskriver pakkernes opførsel, ved et afkastforløb, og på samme tid er beregningsmæssig billig.

De fleste friktionsmodeller kan inddeles i tre kategorier: modeller der bygger på den klassiske Coulomb friktionsteori, modeller der bygger på Dahls teori, samt en gruppe der er udviklet til brug ved beskrivelse af geologiske fænomener.

Det er valgt ikke at implementere geologiske modeller, da disse primært er udviklet med henblik på beskrivelse af stick-slip fænomenet (Dupont, et al., 1993).

# 8.1.1. Coulombs model

Den klassiske Coulombs friktionsteori bygger på den empiriske lov om sammenhængen mellem normalkraft og friktionskraft. Denne sammenhæng kan i mange tilfælde med god tilnærmelse beskrives ved en friktionskoefficient  $\mu$ .

$$F = N \cdot \mu \tag{8.1}$$

Friktionskoefficienten  $\mu$  kan enten være bestemt ud fra en statisk tilstand  $\mu_s$ , hvor koefficienten er bestemt ud fra Break-away kraften, eller ved en dynamisk tilstand, hvor koefficienten  $\mu_d$  bestemmes ved den dynamiske friktionskraft.

#### **Coulombs friktionsmodel**

Coulombs friktionsmodel tager udgangspunkt i den dynamiske friktionskraft og er delt op i to dele. Er der bevægelse mellem kontaktfladerne bestemmes friktionskraften med normalkraften og den dynamiske friktionskoefficient  $\mu_d$ , se Figur 8.1. Ved stilstand kan friktions-

kraften mellem fladerne være mellem nul og den dynamiske friktionskraft.



Figur 8.1: Coulombs friktionsmodel uafhængig af hastigheden v.

Friktionsmodellens to betingelser er givet ved.

$$\begin{cases} F = N \cdot \mu_d \cdot (-sign(v)) & v \neq 0 \\ F < N \cdot \mu_d & v = 0 \end{cases}$$
(8.2)

Coulombs friktionsmodel, som illustreret på Figur 8.1, er en meget simpel model. Pga. den simple sammenhæng, anvendes denne model ofte i forbindelse med beregninger, hvor præcisionen af friktion har en mindre betydning.

#### 8.1.2. Dahls model

Dahls model blev udviklet i forbindelse med beskrivelsen af friktion i lejer. Modellen er udviklet ud fra den antagelse at normalkraften er konstant samt at bevægelsen foregår i én dimension.

Dahls model tager udgangspunkt i en fysisk tolkning af det der sker når kontaktfladerne bryder forbindelsen. Dahl beskrev denne opførsel ved at anvende fjedre monteret mellem friktionsfladerne, se Figur 8.2 (A).

Påføres legemerne en belastning strækkes fjedrene (B), svarende til pre-sliding forskydning. Opnår kraften, Coulombs friktionskraft, brydes nogle af fjedrene og der forekommer en relativ hastighed mellem fladerne (C). Idet nogle af fjedrene brydes gendannes nye i en ikke deformeret tilstand. På den måde vil fjedrene ved den relative bevægelse mellem kontakt-fladerne, kontinuert blive brudt og gendannet, ud fra et tilfældighedsprincip. Stoppes den ydre påvirkning vil friktionskraften mindskes og de eksisterede fjedre trækker sig sammen (D) (Dahl, 1968).



Figur 8.2: A) Kontakten mellem to flader modeleret ved fjederforbindelser. B) Ved små forskydninger vil fjedrene kun strækkes og derefter finde tilbage til udgangspunktet når belastningen ophører. C) Når forskydningskraften mellem de to flader bliver større en Break-away kraften, vil fjedrene brydes. Nye fjederforbindelser vil kontinuerligt blive dannet og brudt. D) Ved aflastning vil overfladerne følge de nye fjederforbindelsers relaksation.

Dahls fjedermodel giver en hystereseopførsel, som kan sammenlignes med en elastisk ideal plastisk materialeopførsel (Dahl, 1975). Ud fra dette kan friktionsopførslen beskrives ud fra en arbejdskurve, tilsvarende et elastisk ideal plastisk materiale, givet ved:

$$\frac{dF}{dx} = \sigma_0 \cdot \left| 1 - \frac{F}{F_c} \cdot sign(v) \right|^{t} \cdot sign\left( 1 - \frac{F}{F_c} \cdot sign(v) \right)$$
(8.3)

*F* er friktionskraften,  $\sigma_0$  er fjederstivheden,  $F_C$  er Coulombs friktionskraft, *x* er den relative forskydning, *v* er den relative hastighed og *i* beskriver formen på forskydnings-kraftkurven. Dette udtryk omskrives til at beskrive friktionskraftens opførsel med hensyn til tiden.

Løses denne 1.ordens differentialligning, bestemmes friktionskraften F.

Dahls model kan bedst anses som en generalisering af Coulombs friktionsmodel, da Coulomb er et specialetilfælde til Dahl (Dahl, 1968). Forskellen mellem de to modeller, er at der i Dahls model er indbygget en mulighed for pre-sliding forskydning. Dahls model kan ikke beskrive den statiske friktion, Stribeck effekten samt viskøse effekter, se Appendiks IV.

### 8.1.3. Børstemodellen

Denne model antager at friktionsfladerne i kontakt, kan beskrives som to børstebelagte overflader, hvor kontakten mellem børsterne giver en friktionsmodstand, se Figur 8.3 (Olsson, et al., 1997).



Figur 8.3 Overfladekontakten beskrevet ved en børstekontakt. Børsternes evne til at fjedre medfører, at de ved en given udbøjning glider af hinanden.

Princippet for børstemodellen er at det enkelte børstehår virker som en bladfjeder. Når to børstehår kommer i kontakt og begynder at udbøje, giver det anledning til en friktionsmodstand mellem fladerne. Børsternes udbøjning fortsætter indtil den bliver så stor, at børsterne glider af hinanden, hvormed kontakten brydes.

Den børstebelagte overflades opførsel antages at kunne beskrives ud fra ét gennemsnitligt børstehår. Friktionskraften *F*, der opstår ved en given gennemsnitlige udbøjning af børsthåret, *z*, kan beskrives ved en sammenhæng mellem *z* og stivheden,  $\sigma_0$ .

$$F = \sigma_0 \cdot z \tag{8.5}$$

Den initielle børsteudbøjningen, z følger den relative forskydning mellem overfladerne. Dette medfører at den afledte  $\dot{z}$  vil følge den relative forskydningshastighed. Når Børstehårets modstand ikke kan opnå en ligevægt, med legemets påvirkning, vil den begynde at glide og dens udbøjning z, bliver konstant, hvormed den afledte  $\dot{z}$  bliver nul. Ud fra denne opførsel er følgende differentialligning opstillet (Canudas de Wit, et al., 1995).

$$\dot{z} = v - \frac{|v|}{g(v)} \cdot z \tag{8.6}$$

Hvor funktionen g(v) er et positiv udtryk, der indeholder informationer omkring materialeegenskaber, smøring, temperatur, mm. Ved en tilstand, hvor den relative hastighed mellem friktionsfladerne er positiv og konstant, vil den gennemsnitlige udbøjning ligeledes være konstant,  $z = z_{ss}$  samt dens afledte lig nul. I dette tilfælde er det muligt at beskrive funktionen g(v) ved.

$$\dot{z} = 0 = v - \frac{|v|}{g(v)} z_{ss}$$

$$\ddagger z_{ss} = \frac{v}{|v|} g(v)$$

$$\Downarrow$$

$$\downarrow z_{ss} = g(v)$$
(8.7)

Ud fra ligning (8.5) og (8.7) beskrives friktionskraften, ved konstant hastighed.

Ud fra ligningerne (8.5), (8.6) og (8.8) kan følgende sammenhæng mellem friktionskraften og hastigheden mellem de to overflader opstilles.

. .

$$F = \sigma_{0} \cdot \dot{z}$$

$$\hat{F} = \sigma_{0} \cdot \left( v - \frac{|v|}{g(v)} \cdot z \right)$$

$$\hat{F} = \sigma_{0} \cdot v \left( 1 - \frac{|v|}{v \cdot \frac{F_{C}}{\sigma_{0}}} \cdot \frac{F}{\sigma_{0}} \right)$$

$$\hat{F} = \sigma_{0} \cdot v \left( 1 - \frac{F_{C}}{F_{C}} \cdot sign(v) \right)$$
(8.9)

Af ligning (8.9) fremgår det at børstemodellen svarer til Dahls model i ligning (8.4), hvor *i* er sat til én. Børstemodellen er derfor et specialtilfælde af Dahls model.

#### 8.1.4. LuGres model

LuGres model er en videreudvikling af børstemodellen, hvor dæmpning og viskøse effekter er implementeret. Ved at udvide ligning (8.5) med to led, der beskriver dæmpning og viskositet, opnås LuGres model (Canudas de Wit, et al., 1997).

$$F = \sigma_0 \cdot z + \sigma_1 \dot{z} + \alpha_2 \cdot v \tag{8.10}$$

Hvor  $\sigma_1$  beskriver dæmpningen og  $\alpha_2$  beskriver viskositeten. Ved samme steady-state betragtning som ligger til grund for ligningerne (8.5) til (8.8) opnås følgende udtryk, ud fra ligning (8.10).

$$F_{ss} = \sigma_0 \cdot g(v) + \alpha_2 \cdot v \tag{8.11}$$

Hvor  $F_{ss}$  er friktionskraften ved steady-state tilstanden, dvs.  $\dot{z} = 0$  og hastigheden er positiv, dvs. sign(v) = 1. Det er givet fra (Canudas de Wit, et al., 1995), at følgende udtryk for  $\sigma_0 \cdot g(v)$ , giver en god approksimation af Stribeck effekten.

$$\sigma_0 \cdot g(v) = \alpha_0 + \alpha_1 \cdot e^{-\left(\frac{v}{v_0}\right)^2}$$
(8.12)

Hvor konstanten  $\alpha_0$  svarer til Coulomb friktionskraften  $F_c$ , og  $\alpha_1$  svarer til  $F_s - F_c$ , hvor  $F_s$  er Break-away kraften.  $v_0$  beskriver, hvordan  $\sigma_0 \cdot g(v)$  varierer mellem Coulomb kraften og den statiske friktionskraft (Canudas de Wit, et al., 1995).

LuGres model beskriver samme fænomener som børstemodellen. Desuden beskriver modellen Break-away kraften og Stribeck effekten samt viskøse effekter, som friction lag og hysterese.

# 8.2. Eksperimentel bestemmelse af fiktionsparametrene

Bortset fra de dynamiske parametre,  $\sigma_0$  og  $\sigma_1$ , er friktionsparametrene til de tre anvendte modeller bestemt ud fra eksperimentalt opsamlede data.

Forsøgsudførelsen tager udgangspunkt i afkastet af en pakke ned af en stålsliske. Det er derfor valgt at undersøge friktionen mellem stål og pap. De eksperimentelle data er opsamlet under friktionsforsøg udført mellem en blank tør stålplade og papkasser.

Selve forsøgsopstillingen er udført som en 0.39 m bred og 5.5 m lang glidebane af stålplader, placeret mellem to skinner, se Figur 8.4. Banens længde er valgt ud fra de eksisterende skinner i laboratoriet. I den ene ende af banen er der monteret et spil. Spillet trækker vha. en træksnor en målevogn der kører på skinnerne. Målevognen trækker en slæde bestående af en papkasse, påført en forud defineret vægt. Friktionskraften måles med en transducer placeret mellem målevogn og slæde.



Figur 8.4: Gearmotoren trækker vha. træksnoren målevognen, som gennem en måleforbindelse trækker slæden. Slæden glider hen over glidebanen, som har en bredde på 0.39 m og en længde på 5.5 m.

En nærmere gennemgang af de udførte forsøg og den efterfølgende databehandling findes i Appendiks V og Appendiks VI.

# 8.2.1. Parametre til LuGres model

LuGres friktionsmodel indeholder seks parametre, to dynamiske,  $\sigma_0 \circ \sigma_1$ , samt fire steadystate parametre,  $\alpha_0, \alpha_1, \alpha_2 \circ \sigma_0$ .

De to dynamiske parametre kan beskrives som en fjeder og en dæmper der modellerer presliding forskydning. Værdierne for disse to dynamiske parametre er bestemt ved et estimat, da bestemmelsen af disse kræver yderligere forsøgsarbejde, hvilket ligger uden for projektets rammer. De to dynamiske parametre er estimeret på baggrund af simuleringsresultaterne og størrelsen på pre-sliding forskydning.

- $\sigma_0 = 120000 \ N/m$
- $\sigma_l = 10 N \cdot s/m$

Det er gennem optimering muligt at bestemme størrelsen på de fire steady-state parametre, der bringer steady-state kurven, for LuGres model, tættest på de i forsøgene målte værdier.

# 8.2.2. Parametre til Dahls model

Dahls model indeholder to parametre. En steady-state parameter svarende til Coulomb friktionskraften,  $F_c$  og en dynamisk,  $\sigma_0$ .  $F_c$  sættes til den samme værdi som  $\alpha_0$  i LuGres model, da den fysiske fortolkning af  $\alpha_0$  er Coulomb friktionkraften. Den dynamiske parameter  $\sigma_0$  svarer, ligeledes som ved LuGres model, til en fjederstivhed og sættes til 120000 N/m.

# 8.2.3. Parametre til Coulomb

Den fysiske defination af  $\alpha_0$ , i LuGres model, svare til Coulomb friktionskraften.

#### 8.2.4. Bestemmelse af steady-state koefficienterne

Under den antagelse at der er en ligefrem proportional sammenhæng mellem normalkraften N og de tre steady-state parametre,  $\alpha_0$ ,  $\alpha_1$  og  $\alpha_2$ , er de tre steady-state koefficenter,  $a_0$ ,  $a_1$  og  $a_2$ , defineret på følgende måde.

$$a_0 = \frac{\alpha_0}{N}, \quad a_1 = \frac{\alpha_1}{N}, \quad a_2 = \frac{\alpha_2}{N}$$

Ved hver forsøgsudførsel er steady-state kræfterne ved forskellige hastigheder og normalkræfter bestemt. På baggrund af ligning (8.11) og (8.12) er der gennemført en kurvetilpasning ud fra steady-state kræfterne, mht. de fire steady-state koefficienter. Ud fra de eksperimentelt fundne data er størrelsen af steady-state koefficienterne givet i Tabel 8.1.

$a_0$	$a_1$	$v_0[m/s]$	$a_2[s/m]$		
0.332	-0.070	0.235	0.028		

Tabel 8.1: De steady-state parametrene anvendt i programmet.

En nærmere beskrivelse af de bestemte koefficienter kan findes i Appendiks V.

# 8.3. Implementering af friktionsmodeller

I forbindelse med implementeringen af friktionsmodellerne har det været nødvendigt at lave nogle antagelser, for at modellerne passer ind i simuleringsprogrammet. I det følgende er disse antagelser beskrevet, samt hvordan modellerne er implementeret.

## 8.3.1. Coulombs friktionsmodel

Coulombs model er opdelt i to dele. En del med en relativ bevægelse mellem kontaktfladerne, og en del uden. Er der ikke en relativ bevægelse, er det nødvendig at anvende en kraftligevægt til at beregne den stationære friktionskraft. For at denne kraftligevægt kan bestemmes er det nødvendigt at låse fladerne i forhold til hinanden. Fremgangsmåden ved implementering vil således være: Først beregnes den stationære friktionskraft. Er denne større end den maksimale friktionskraft, er der bevægelse. Dermed er kraften givet ved den dynamiske friktionskraft og låsningen mellem fladerne ophæves (Garcia de Jalón, et al., 1993 s. 338). I stedet for at anvende ovenstående formulering er det valgt at anvende en tillempning, af den statiske friktionskraft, således at den er en funktion af en hastighed mellem nul og  $v_{\varepsilon}$ . For hastigheder større end  $v_{\varepsilon}$  sættes friktionskraften til  $\alpha_0$ . Friktionskraften vil derfor kunne beskrives ved følgende gaffelfunktion (Hippman, 2004).

$$F = \begin{cases} \alpha_0 \cdot \frac{v}{v_{\varepsilon}} \cdot \left(2 - \frac{v}{v_{\varepsilon}}\right) & v < v_{\varepsilon} \\ \alpha_0 & v \ge v_{\varepsilon} \end{cases}$$
(8.13)

På Figur 8.5 er sammenhængen mellem de to funktioner i ligning (8.13) illustreret.



Figur 8.5: Coulombs friktionsmodel anvendt i simuleringsprogrammet.

Da friktionskraften er en funktion af hastigheden vil en kraftligevægt, hvor friktionskraften er forskellig fra nul, medføre en bevægelse. Denne formulering medfører at friktionskraften aldrig alene kan bringe et legeme til stilstand på en skrå flade. Ved implementeringen af denne friktionsmodel i simuleringsprogrammet, er det valgt at sætte  $v_{\varepsilon}$  til 0.001 *m/s*. I simuleringsprogrammet er Coulombs model givet i subrutinen *RegCoulomb* i modulet *Friction*. Implementeringen af Coulombs friktionsmodel er beskrevet ved følgende pseudokode.

Pseudokode:	
$\dot{\mathbf{r}}_{\mathrm{P,t}}$	Relative hastighed i tangentplanet
$\mathbf{f}_n$	Normalkraften
$\alpha(N)$	Vektor med de anvendte friktionsoparametre
h	Tidstep
$\mathbf{f}_{f}$	Vektor for friktionskraften
function $\mathbf{f}_f = \mathbf{f}_f$	$f(\dot{\mathbf{r}}_{p},\mathbf{f}_{n})$
$N =  \mathbf{f}_n $	
$\alpha_0 = \alpha$	(N)
$v =  \dot{\mathbf{r}}_P $	
$v_{\varepsilon} = 0.$	001
if(v <	$v_{\varepsilon}$ )
	$F = N \cdot \alpha_0 \cdot \frac{v}{v_{\varepsilon}} \cdot \left(2 - \frac{v}{v_{\varepsilon}}\right)$
else	
	$F = N \cdot \alpha_0$
end	
$\mathbf{f}_f = F$	$\frac{-\dot{\mathbf{r}}_{P,t}}{\left \dot{\mathbf{r}}_{P,t}\right }$
end function	

#### 8.3.2. Dahls model

Ifølge Dahls model kan friktionskraften beskrives ud fra følgende udtryk.

$$\dot{F} = \sigma_0 \cdot v \cdot \left| 1 - \frac{F}{\alpha_0} \cdot sign(v) \right| \cdot sign\left( 1 - \frac{F}{\alpha_0} \cdot sign(v) \right)$$
(8.14)

Udtrykket bygger på den endimensionelle beskrivelse af fjedermodellen. Ved den endimensionelle beskrivelse kan fjederforlængelsen kun skifte retning, hvis den relative hastighed mellem friktionsfladerne har været nul.

Simuleringen tager udgangspunkt i legemer, hvis overflader er beskrevet ved plane trekanter. For at kunne beskrive friktionen mellem disse legemer, er det nødvendigt, at udvide Dahls model til at kunne beskrive friktion i et plan. Den todimensionelle beskrivelse, udføres ved at overføre den endimensionelle opførsel af fjedrene til en todimensional fjedermodel.

Forskellen mellem den en- og todimensionelle model, er at det i den todimensionelle model er muligt at ændre retning, uden at den relative hastighed mellem friktionsfladerne har været nul. Derfor antages det at fjedrene ingen torsionsstivhed har, hvormed det er muligt for fjedrene at rotere frit omkring deres fastgørelsespunkt, uden at dette bidrager til friktionskraften. Baggrunden for denne antagelse er, at den distance den enkelte fjeder strækkes, før den brydes, er lille sammenlignet med den krumning der kan forekommer ved en relativ bevægelse mellem to flader. Den enkelte fjeder vil derfor, tilnærmelsesvis, kun opleve en endimensionel deformation, hvor fjedrene altid bliver stukket i bevægelsesretningen.

Ved den todimensionelle beskrivelse er det, for at finde friktionskraften, kun nødvendigt at kende fem informationer: Længden af den i planet projicerede hastighedsvektor  $\dot{\mathbf{r}}_p$ , de tre parametre til modellen,  $\alpha_0$ ,  $i \circ g \sigma_0$ , samt forrige tidssteps friktionskraft. Friktionskraften bestemmes ved at integrere gradienten, fra kontaktens påbegyndelse  $t_0$  til tidspunktet  $t_i$ , hvor friktionskraften ønskes bestemt.

$$F = \int_{t_0}^{t_i} \dot{F} dt \tag{8.15}$$

Integrationen er udført numerisk ved hjælp af 4.ordens Runge-Kutta. Eftersom friktionskraften er modsatrettet bevægelsesretningen er vektoren for friktionskraften  $\mathbf{f}_f$  givet ved følgende ligning.

$$\mathbf{f}_{f} = F \cdot \frac{-\dot{\mathbf{r}}_{P,t}}{\left|\dot{\mathbf{r}}_{P,t}\right|} \tag{8.16}$$

Bestemmelsen af friktionskraften i simuleringsprogrammet er som følgende: Ved en kontakt mellem to legemer, dannes fjedrene. Er der en relativ hastighed mellem legemerne, vil fjedrenes forlængelse påbegyndes, hvormed friktionskraften, *F*, vokser. Er friktionskraften ikke stor nok til at sikrer en ligevægt, mellem legemerne, sker der en bevægelse mellem overfladerne, og friktionskraften bliver konstant. Denne bevægelse vil fortsætte indtil der opstår en ligevægtssituation, hvor fjedrene aflastes og friktionskraften falder, eller legemerne skilles og friktionskraften nulstilles. I det tilfælde, hvor den relative hastighed mellem overfladerne blive nul, bestemmes friktionskraftens retning ud fra hastighedsvektoren, for den sidste udførte bevægelse. Værdierne for hastigheden og friktionskraften gemmes for hvert tidsstep, sammen med relationen til de involverede knuder og linjer i den pågældende kontakt. I forbindelse med implementeringen af Dahls model, har det vist sig nødvendigt, at sætte en grænse for værdien af gradienten  $\dot{F}$ . Dette skyldes at store gradienter kan forekomme i forbindelse med en lille normalkraft og en høj hastighed.

I simuleringsprogrammet er Dahls friktionsmodel givet i subrutinen *Dahl* i modulet *Friction*. Implementeringen af Dahls model er beskrevet ved følgende pseudokode.

> Pseudokode:  $\dot{\mathbf{r}}_{p,t}$ Relative hastighed i tangentplanet  $\mathbf{f}_n$ Normalkraften  $\alpha(N)$ Vektor med de anvendte friktionsoparametre h Tidstep  $\mathbf{f}_{f}$ Vektor for friktionskraften F Friktionskraften beregnet i forrige tidstep function  $\left| \mathbf{f}_{f}, \dot{F} \right|^{T} = f\left( \dot{\mathbf{r}}_{p,t}, \mathbf{f}_{n}, F, h \right)$  $N = |\mathbf{f}_n|$  $v = \left| \dot{\mathbf{r}}_{p,t} \right|$  $\begin{vmatrix} \alpha_0 & \sigma_0 \end{vmatrix}^T = \boldsymbol{\alpha}(N)$  $\dot{F} = \sigma_0 \cdot v \cdot \left| 1 - \frac{F}{\alpha_0} \right|^i \cdot sign\left( 1 - \frac{F}{\alpha_0} \right)$  $if(\dot{F} > gradient grænse)$  $\dot{F} = gradient grænse$ end  $F_n = F + \dot{F} \cdot h$  $\mathbf{f}_f = F_n \cdot \frac{-\dot{\mathbf{r}}_{p,t}}{|\dot{\mathbf{r}}_{n,t}|}$ end function

#### 8.3.3. LuGres model

Friktionskraften kan ifølge LuGres model beskrives ud fra følgende udtryk.

$$F = \sigma_0 \cdot z + \sigma_1 \cdot \dot{z} + \alpha_2 \cdot v \tag{8.17}$$

Det er muligt at bestemme friktionskraften, hvis tre parametre er kendt: Den gennemsnitlige børste deformation z hastigheden hvormed børsten deformeres  $\dot{z}$  og hastigheden hvormed fladerne forskydes i forhold til hinanden v.

LuGres model er ligesom Dahls, kun beskrevet endimensionel. For at udvide LuGres model til to dimensioner antages det at børsterne frit kan rotere omkring deres egen akse, se Figur 8.6. De udbøjede børstehår vil derfor altid vende modsat overfladens relative bevægelses-retningen.



Figur 8.6: Børsterne kan frit rotere i deres fastgørelse, og de udbøjer modsat bevægelsesretningen.

Ved den todimensionelle beskrivelse er det, for at finde friktionskraften, nødvendigt at kende otte informationer: Længden af den i planet projicerede hastighedsvektor, de seks parametre til modellen, samt den gennemsnitlige børsteudbøjning, z. Børsteudbøjningen bestemmes ved at integrere gradienten  $\dot{z}$ , fra kontaktens påbegyndelse  $t_0$  til tidspunktet  $t_i$ , hvor friktionskraften ønskes bestemt.

$$z = \int_{t_0}^{t_i} \dot{z} dt \tag{8.18}$$

I simuleringsprogrammet er integration udført vha. 4.orendes Rungen-Kutta. Børstehårets opførsel, i modellen, vil være som følger: Børstehårets deformation vil påbegynde, når kontakten mellem legemerne opstår. Er friktionskraften ikke stor nok til at sikre systemets statiske ligevægt, vil der ske glidning mellem kontaktfladerne. Denne glidning vil fortsætte indtil en ligevægtssituationen opstår, eller fladerne bevæger sig fra hinanden, hvormed *z* nulstilles.

I det tilfælde, hvor den relative hastighed mellem overfladerne er nul, vil modellen ikke have kendskab til i hvilken retning friktionskraften skal vende. Det er valgt i disse situationer at anvende den sidst forekomne bevægelsesretning. Tilsvarende som for Dahls model gemmes værdierne for hastigheden og børstedeformationerne, for hvert tidsstep, med relation til de, i kontakten, involverede knuder og linjer.

Ligesom for Dahls model, er det nødvendigt at sikre stabilitet i beregningerne ved, at sætte en grænse for størrelsen af  $\dot{z}$ .

I simuleringsprogrammet er friktionsmodellen givet i subrutinen *LuGre* i modulet *Friction*. Implementeringen af LuGres model er beskrevet ved følgende pseudokode.

Pseudokode:								
$\dot{\mathbf{r}}_{\mathrm{p,t}}$	Relative hastighed i tangentplanet							
$\mathbf{f}_n$	Normalkraften							
$\alpha(N)$	Vektor med de anvendte friktionsoparametre							
h	Tidstep							
$\mathbf{f}_{f}$	Vektor for friktionskraften							
Z	Børsteudbøjningen							
function $\begin{bmatrix} \mathbf{f}_f, \dot{z} \end{bmatrix}^T = f(\dot{\mathbf{r}}_{p,t}, \mathbf{f}_n, z, h)$								
$N =  \mathbf{f} $	1							
$v =  \dot{\mathbf{r}}_{p,t} $	$v = \left  \dot{\mathbf{r}}_{\mathrm{p,t}} \right $							
$\lfloor \alpha_0 \rfloor$	$\begin{bmatrix} \alpha_0 & \alpha_1 & \alpha_2 & \sigma_0 & \sigma_1 & v_0 \end{bmatrix}^T = \boldsymbol{\alpha}(N)$							
$g = \alpha_0$	$_{0}+\alpha_{1}\cdot e^{-\left(\frac{\nu}{\nu_{0}}\right)^{2}}$							
$\dot{z} = v -$	$-\frac{\sigma_0}{g} \cdot z \cdot v$							
$if(\dot{z} >$	gradient grænse)							
	ż = gradient grænse							
end								
z = z +	$z = z + h \cdot \dot{z}$							
$F = \sigma_0$	$\sigma_0 \cdot z + \sigma_1 \cdot \dot{z} + \alpha_2 \cdot v$							
$\mathbf{f}_f = F$	$\frac{\mathbf{\dot{r}}_{\mathrm{p,t}}}{\left \mathbf{\dot{r}}_{\mathrm{p,t}}\right }$							
end function								

### 8.3.4. Bogholderi

Dahl og LuGres friktionsmodeller anvender oplysninger fra sidste tidsstep i form af friktionskraften og børsteudbøjningen z. Eftersom der kan være flere kontakter, mellem legemerne, giver det anledning til et større bogholderi i simuleringsprogrammet.

Bogholderiet er i programmet placeret sammen med beregningerne af friktionskraften. Inden friktionskraften beregnes undersøges det om den pågældende kontakt, mellem legemerne, matcher en fra sidste tidsstep. Er det tilfældet anvendes værdierne for z eller F fra den pågældende kontakt, fra sidste tidsstep. Hvis kontakten ikke kan identificeres med en tidligere kontakt sættes z og F til nul. Efter friktionskraften er beregnet, gemmes oplysningerne for z eller F samt enhedsvektoren for den projicerede hastighedsvektor, for den pågældende kontakt. Hvis det er en ny kontakt oprettes der en ny kontakt i bogholderiet. Er kontakten tidligere anvendt opdateres værdierne for den pågældende kontakt.

Data fra hver iteration gemmes i to matricer, hhv.  $\mathbf{F}_{bog}$  og  $\mathbf{F}_{save}$ .  $\mathbf{F}_{bog}$  er en to dimensionel matrix hvor både rækker og søjler angiver legemenummeret i simuleringen. I matricens indgange angives antallet af kontakter mellem de pågældende legemer. Matricen  $\mathbf{F}_{save}$  er en fire dimensionel matrix, hvor to af dimensionerne angiver kontakten mellem legemerne. De sidste to dimensioner angiver hhv. kontaktnummeret og oplysningerne om kontakten. Oplysningerne om de enkelte kontakter kan opdeles i tre dele, i form af en identifikation-, historik- og en datadel. På Figur 8.7 er et eksempel på delenes opbygning i matricen  $\mathbf{F}_{save}$  for *n* kontakter mellem to legemer med Dahls friktionsmodel vist.

1	$I_1^1$	$I_2^1$	$I_3^1$	$I_4^1$	$It^1$	$Sv^1$	$F^1$	$v^1_{\alpha,x}$	$v^1_{\alpha,y}$	$v^1_{\alpha,z}$	$dF_1^1$	$dF_2^1$	$dF_3^1$	$dF_4^1$
:	÷	÷	÷	÷	:	÷	÷	÷	:	÷	÷	÷	÷	:
n	$I_1^n$	$I_2^n$	$I_3^n$	$I_4^n$	$\underline{It^n}$	$Sv^n$	$F^{n}$	$v_{\alpha,x}^n$	$v_{\alpha,y}^n$	$v_{\alpha,z}^n$	$dF_1^n$	$dF_2^n$	$dF_3^n$	$dF_4^n$
	Indtifikationsnr. His					ist.					Data			

*Figur 8.7: Opbygningen af matricen*  $\mathbf{F}_{save}$ *, for kontakten mellem to legemer.* 

De første fire  $I_1 - I_4$  udgør identifikationsdelen, som anvendes til at identificere de nye kontakter ud fra kontakter fra forrige tidsstep. De næste to, historikdelen, anvendes i forbindelse med opdatering af kontaktlisten. Det første tal, *It*, angiver, hvor mange iterationer den pågældende kontakt ikke har været anvendt. Den anden, *Sv*, angiver om kontakten har været i brug i den pågældende iteration ved at sætte en switch til *on* eller *off*. De efterfølgende pladser, data delen, indeholder oplysninger til bestemmelse af friktionskraften. Første plads angiver sidste tidssteps børsteudbøjning *z* eller friktionskraft *F*. De næste tre pladser indeholder enhedsnormalvektoren for den projicerede hastighedsvektor  $v_{\alpha}$ . De sidste fire pladser anvendes til af gemme værdierne af de afledte *dz* eller *dF*, for hvert af de fire step. Disse værdier anvendes til at opdatere værdierne for *z* eller *F* vha. 4. Rungen Kutta, efter hvert tidsstep. Opdateringen af *z* eller *F* foregår i slutningen af subrutinen *4rk* i modulet *NumInt*. I Appendiks VIII er en mere dybdegående beskrivelse af det implementerede bogholderi givet.
# 9. Kontaktbestemmelse

Kontakt mellem objekter indgår i flere forskellige simuleringssammenhænge, bl.a. anvendes det inden for fysisk modulering, molykulær modulering, computer animationer, robot styringer, ol. (Lin, 1993). Generelt kan kontaktbestemmelse kort beskrives som: Bestemmelse af hvorvidt to legemer kolliderer ud fra legemernes geometriske udformning og orientering i det globale rum.

På tros af denne korte formulering er kontaktalgoritmerne ofte flaskehalsen i simuleringsprogrammer, da det som regel er en beregningsmæssig tung opgave. Eftersom dynamiske værktøjer er af stigende efterspørgsel i forbindelse med, computerspil, fysisk simulering, robotsimuleringer osv., er interessen for kollisionsbestemmelse ligeledes vokset. Denne store interesse har medført en kraftig udvikling, hvor der hele tiden udvikles nye kontaktformuleringer. Der findes et utal af kontaktformuleringer (Lin, 1993), (Mirtich, 1998) som alle er designet til hvert sit formål.

Dette afsnit har til formål at beskrive den, i simuleringsprogrammet, anvendte kontaktformulering. Indledningsvist begrundes valget af kontaktformulering, hvorefter diskretiseringen af geometrierne beskrives. Efterfølgende beskrives de anvendte kontaktalgoritmer, der er inddelt i en grov samt en fin algoritme. Afslutningsvist beskrives kontaktbestemmelsesalgoritmernes implementering i simuleringsprogrammet.

# 9.1. Valg af kontaktformulering

Det er valgt, at konstruere en algoritme der er tilpasset den aktuelle opgave. Dette skyldes følgende årsager: De nyere kontaktformuleringer er formuleret ud fra den impulsbaserede metode og kan ikke anvendes på stafmetoden. Hovedparten af kontaktformuleringerne forudsætter at legemernes geometrier er konvekse, hvor det anvendte sliskedesign altid indbefatter en konkav geometri. Dette problem kan løses ved at inddele den konkave geometri i flere konvekse legemer. Dette er imidlertid uønsket pga. kravet om et direkte link til Solid-Works. Derudover er der ofte, ved de eksisterende formuleringer, lavet simplificeringer i kontaktformuleringerne for at opnå realtidssimuleringer, hvormed præcisionen af kollisionspunktet ikke vægtes højt.

Pga. ovenstående er det valgt at konstruere en kontaktformulering som tilegner sig projektets problematik.

# 9.2. Diskretisering af geometri

I CAD programmer er overflader repræsenteret ved Bézierkurver og NURBS, hvilket gør det muligt at lave rundinger, huller, dobbeltkrumme overflader, osv. En kontaktbestemmelse mellem sådanne geometrier medfører at et sæt af højere ordens ikke-lineære algebraiske ligninger skal løses. Analytisk er dette ofte ikke muligt, da disse er for komplekse, hvormed numeriske metoder skal anvendes (Goyal, et al., 1994 I). For at simplificere beregningerne diskretiseres geometrierne til polyeder, dvs. et trådnet af polygoner udspændt mellem en mængde knuder. Dermed reduceres kompleksiteten, således at kontaktalgoritmen kan formuleres udelukkende vha. lineære ligninger.

I simuleringsprogrammet diskretiseres geometrierne til trekanter. Ved diskretiseringen af geometrien, til trekanter, opstår der diskretiseringsfejl for krumme kurver. Dette medfører at kontakten mellem legemerne ikke er korrekt. For at opnå højere præcision kan en finere diskretisering anvendes. Dette medfører dog tungere beregninger, eftersom antallet af tre-kanter, øges

Det antages at diskretiseringen af sliskerne og pakkerne ikke giver anledning til store diskretiseringsfejl, eftersom sliskerne er udformet i bukkede stålplader med rette flader. Ligeledes er pakkernes facon givet ved kantede geometrier, så som kvadratiske kasser, hvilket ikke giver diskretiseringsfejl.

Diskretiseringen udføres i praksis ved at gemme SolidWorks parten i wrl-format (Virtual Reality Modeling Language).

I forbindelse med diskreteseringen er det vigtigt at være opmærksom på at emnet er en lukket geometri, som ikke indeholder duplikerede punkter.

# 9.3. Opbygning af kontaktbestemmelse

Kontaktbestemmelsesmodulet har til formål at bestemme hvilke kontakter der er mellem legemerne. For hvert kontaktpunkt der bestemmes, skal oplysninger om kontaktpunktets placering og penetreringsdybde sendes til reaktionsmodulet, hvor strafkraften, friktionskraften og momentet beregnes. Generelt består kontaktbestemmelsen af en række betingelser der, hvis de er opfyldt, medfører at to legemer er i kontakt. Kontaktbestemmelsen er inddelt i tre faser i form af en grov kontaktbestemmelse, en fin kontaktbestemmelse samt en bestemmelse af kontaktpunktet og penetreringsdybden, som vist på Figur 9.1.



Figur 9.1: Opbygning af kontaktbestemmelse.

Fra de dynamiske beregninger gives et input i form af positionerne for samtlige legemer, hvilket indledningsvist anvendes i en grov kontaktbestemmelse. Er der ikke kontakt mellem de to legemer, ved den grove algoritme, sammenlignes det næste sæt af legemer. Er der derimod kontakt udføres en finere kontaktbestemmelse mellem de to legemer. Hvis den fine kontaktbestemmelsesalgoritme ikke finder nogen kontakt, undersøges det næste sæt af legemer. Finder den derimod kontakt, bestemmes kontaktpunktets placering og penetreringsdybde. Hele denne procedure udføres mellem samtlige legemer, for hvert tidsstep. I de følgende afsnit er den grove og fine kontaktbestemmelsesalgoritme beskrevet. Bestemmelsen af kontaktpunktets placering og penetreringsdybde er beskrevet sammen med den fine kontaktbestemmelse.

# 9.4. Grov kontaktbestemmelse

Den grove kontaktformulering har til formål hurtigt at bestemme om der evt. er kontakt. Dvs. at algoritmen på så tidligt et stadie som muligt, stopper kontaktsøgningen. Derved begrænses antallet af udførte beregninger, hvormed beregningstiden for simuleringen reduceres.

I simuleringsprogrammet tager den grove kontaktformulering udgangspunkt i en hierarkisk grænsevolumenformulering givet ved (Zachmann, 2000). I stedet for at kontrollere samtlige geometriske elementer i legemerne mod hinanden, omgrænses legemernes geometri af et grænsevolumen. Med den hierarkiske grænsevolumenformulering opsplittes legemerne i en række underniveauer, i form af nye grænsevolumener. Ved kontaktbestemmelse, mellem to legemer, undersøges kontakten mellem grænsevolumener. Er der kontakt, på det øverste niveau, fortsætter kontaktundersøgelsen på et niveau lavere. Er der kontakt på det laveste niveau fortsættes kontaktsøgningen med den fine kontaktformulering.

# 9.4.1. Valg af grænsevolumen

Valget af grænsevolumeners udformning, er et kompromis mellem, hurtigt at kunne beregne om der er kontakt, mod hvor tæt grænsevolumen omslutter legemet (Coutinho, 2001 s. 18). Generelt kan der anvendes to udformninger for grænsevolumen, en kugle eller en kasse. Kuglen giver en simpel bestemmelse af kontakt, baseret på kuglens centrum og dens radius. Derimod medfører kugleformuleringen en mindre fleksibel tilpasning til legemets geometri, hvilket ofte betyder en større afstand fra kuglens periferi og ind til legemets overflade. Specielt ved lange smalle geometrier, giver dette en ringe udnyttelse af kontakthierarkiet. Grænsevolumenerne beskrevet ved kasser giver derimod en bedre tilpasning til legemernes geometri, hvorimod kontaktbestemmelsen er mere beregningstung.

Under udviklingen af simuleringsprogrammet er der udført forsøg med at implementere både den kugle- og kasseformede grænsevolumen. Ved de udførte forsøg gav den kasseformede grænsevolumenformulering de hurtigste beregninger. Det er derfor valgt at anvende den kasseformede grænsevolumen.

#### 9.4.2. Kontaktbestemmelse mellem grænsevolumener

De kasseformede grænsevolumener delfiners på baggrund af legemerne i simuleringsprogrammet. Kassernes sider er parallelle med planer udspændt af legemes lokale koordinatakser, hvor kassens dimensioner bestemmes ved geometriens minimum- og maksimumsværdier, langs de lokale akser.

Når kasserne kontrolleres for kontakt, i det pågældende tidsstep, sker det i det globale koordinatsystem. Ud fra kassen givet i det lokale koordinatsystem generes derfor nye kasse omkring det oprindelige grænsevolumen i det globale koordinatsystem (Zachmann, 2000), se Figur 9.4.



Figur 9.2: Når legemet roterer i forhold til det globale koordinatsystem ændres minimum- og maksimumsværdierne for grænsevolumenet.

Den nye kasse genereres på baggrund af hjørnepunkterne, fra kassen, givet i lokale koordinater.

Ud fra de seks minimum- og maksimumsværdier givet i det lokale koordinatsystem kan koordinaterne på kassens otte hjørner bestemmes. De otte koordinatsæt er givet ved søjlerne i matricen  $\mathbf{H}'$ .

$$\mathbf{H'} = \begin{bmatrix} x'_{\min} & x'_{\min} & x'_{\min} & x'_{\min} & x'_{\max} & x'_{\max} & x'_{\max} & x'_{\max} \\ y'_{\min} & y'_{\min} & y'_{\max} & y'_{\max} & y'_{\min} & y'_{\min} & y'_{\max} & y'_{\max} \\ z'_{\min} & z'_{\max} & z'_{\min} & z'_{\max} & z'_{\min} & z'_{\max} & z'_{\min} & z'_{\max} \end{bmatrix}$$
(9.1)

Koordinaterne i  $\mathbf{H}'$  transformeres til globale koordinater, vha. legemets transformationsmatrix  $\mathbf{A}$ .

$$\mathbf{H} = \mathbf{A}\mathbf{H}' \tag{9.2}$$

Minimum- og maksimumsværdierne for kassen i globale koordinater, bestemmes ud fra Hs minimum og maksimumsværdier, for hver række, samt legemets globale position  $\mathbf{r}_i$ .

$$x_{\max} = Max(\mathbf{H}_{1j}) + r_x \qquad x_{\min} = Min(\mathbf{H}_{1j}) + r_x$$
  

$$y_{\max} = Max(\mathbf{H}_{2j}) + r_y \qquad y_{\min} = Min(\mathbf{H}_{2j}) + r_y \qquad (9.3)$$
  

$$z_{\max} = Max(\mathbf{H}_{3j}) + r_z \qquad z_{\min} = Min(\mathbf{H}_{3j}) + r_z$$

På baggrund af ovenstående bestemmes kontakten mellem kassen i og j ved at undersøge om følgende seks betingelser er opfyldt, se Figur 9.3.

$$\begin{aligned} x_{\max,i} &> x_{\min,j} \quad x_{\max,j} > x_{\min,i} \\ y_{\max,i} &> y_{\min,j} \quad y_{\max,j} > y_{\min,i} \\ z_{\max,i} &> z_{\min,i} \quad z_{\max,i} > z_{\min,i} \end{aligned}$$

$$(9.4)$$

Er samtlige betingelser opfyldt er der kontakt mellem kasserne.



Figur 9.3: Kasserne i og j kontrolleres for kontakt ud fra deres minimum- og maksimumsværdier i det globale koordinatsystem.

For at reducere beregningstiden i simuleringsprogrammet er grænsevolumenerne i forbindelse med de faste legemer, dvs. slisken, kun givet i globale koordinater, eftersom disse ikke ændrer position og orientering gennem simuleringen.

### 9.4.3. Hierarkisk struktur

Den hierarkiske opdeling af grænsevolumenerne tager udgangspunkt i det grænsevolumen der omslutter hele legemet, benævnt *maksboks*. De geometriske elementer inden i dette grænsevolumen, er efterfølgende opdelt i nye grupper ud fra trekanternes tyngdepunkter. Hver af disse grupper omsluttes igen af et nyt grænsevolumen. Disse nye grænsevolumener udgør dermed det næsthøjeste niveau i den hierarkiske struktur. Opdeling af grænsevolumenerne i mindre volumener fortsætter indtil alle grænsevolumenerne på det laveste niveau kun indeholder én trekant, se Figur 9.4. Grænsevolumener som kun indeholder én trekant benævnes *minibokse*.



Figur 9.4: Maksboksen som omgiver hele legemet, deles op i mindre grænsevolumener, indtil der kun er grænsevolumener tilbage indeholdende én trekant.

Antallet af grænsevolumener varierer, fra legeme til legeme, afhængigt af antallet af trekanter det enkelte legeme er diskretiseret med. For et grænsevolumen er informationer omkring, hvilke grænsevolumener, det indeholder, på ét niveau lavere, gemt. Det pågældende grænsevolumen har dermed informationer om, hvilke grænsevolumener, niveauet lavere, der skal anvendes i den videre kontaktbestemmelse.

Er der kontakt mellem to legemer vil den grove kontaktalgoritme reducere kontakten, til et sæt af minibokse. På baggrund af miniboksene fortsætter kontaktsøgningen med den fine kontaktformulering.

#### Generering af den hierarkiske struktur

Til opbygning af den hierarkiske strukturer er det i simuleringsprogrammet valgt at implementere en rutine, der bygger på en kontinuerlig opdeling af maksbokse ud fra trekanternes tyngdepunkter.

For en trekant hvis hjørnepunkter kan beskrives ved stedvektorerne  $\mathbf{p}$ ,  $\mathbf{q}$  og  $\mathbf{r}$ , kan stedvektoren for dens tyngdepunkt findes på følgende måde (Krex, 2004 s. 53).

$$\mathbf{t}_T = \frac{\left(\mathbf{p} + \mathbf{q} + \mathbf{r}\right)}{3} \tag{9.5}$$

For at opnå en fornuftig opdeling af kasserne, i mindre kasser, er det valgt at tage udgangspunkt i et middelgeometrisk tyngdepunkt,  $\mathbf{t}_{G}$ .

$$\mathbf{t}_G = \frac{\sum_{i=1}^{N} \mathbf{t}_T}{N}$$
(9.6)

Hvor N er antallet af trekanter grænsevolumen (Coutinho, 2001 s. 21).

Opdelingen af grænsevolumenet, udføres ved at bestemme placeringen af den enkelte trekants tyngdepunkt i forhold til ét eller flere skæringsplaner gennem det middelgeometriske tyngdepunkt. Det er i simuleringsprogrammet valgt at lade disse skæringsplaner være parallelle med planer udspændt af de lokale koordinatakser. Et grænsevolumen af trekanter kan dermed opdeles i 2, 4 eller 8 nye underrum, afhængigt af om der anvendes 1, 2 eller 3 skæringsplaner, se Figur 9.5.



Figur 9.5: Opdeling med 1, 2 og 3 skæringsplaner, gennem det middelgeometriske tyngdepunkt.

Minimum- og maksimumsværdier for de nye grænsevolumener, defineres ud fra de indeholdte, trekanters geometriske udstrækning.

Et eksempel på dette er vist på Figur 9.6, hvor et legeme opdeles vha. tre skæringsplaner. I de to illustrerede underrum A) er der trekanter hvis udstrækning krydser flere underrum. Trekanterne tildeles det underrum som deres tyngdepunkt er placeret i. Dette medfører at underrummets grænsevolumen B) og C) får en udstrækning som går udover de enkelte underrums grænser. Dermed vil grænsevolumenerne, på det enkelte niveau, overlappe hinanden D).



Figur 9.6: A) Ved de to underrum er der flere trekanter, hvis udstrækning krydser flere underrum. B) og C) Underrummets grænsevolumener får en udstrækning som kan gå udover underrummets grænser. D) Grænsevolumenerne overlapper hinanden.

Anvendes der én eller to skæringsplaner, varieres disse i forhold til de tre koordinatakser, hver gang grænsevolumenniveauet ændres. Dette udføres for at sikre at trekanter med tyngdepunkter beliggende i skæringsplanet, kan adskilles i minibokse.

Mimiboksene indeholder udover én trekant også informationer omkring tilstødende linjer og knuder. Da den enkelte trekant har sine linjer tilfælles med de omkringliggende trekanter, vil den enkelte linje være omgivet af mindst to minibokse. Tilsvarende vil knuderne i geometrien have flere tilstødende trekanter, hvormed de ligeledes er omgivet af flere minibokse. For at undgå at knuder og linjer bliver kontrolleret flere gange, i forbindelse med den fine kontaktbestemmelse, er det valgt kun at tildele den ene af de tilstødende minibokse, den pågældende linje eller knude.

Det er efter forsøg med opdeling af grænsevolumenerne i 2, 4 og 8 nye underrum, valgt at anvende en opdeling på 2 skæringsplaner. Denne opdeling er valgt da det giver den hurtigste beregningstid.

#### Anvendelse af den hierarkiske struktur til kontaktbestemmelse

På baggrund af legemernes grænsevolumenhierarkier undersøges det om legemerne er i kontakt, hvilket sker i to trin. Det første trin udføres ved at maksboksen for legeme i, kontrolleres for kontakt gennem hele legeme j's hierarki. Dvs. først kontrolleres maksboksen for legeme i mod maksboksen for legeme j. Er der ikke kontakt kontrolleres et nyt sæt af legemer. Er der derimod kontakt fortsættes kontaksøgningnen mellem maksboksen for legeme i med næste niveau af grænsevolumener i legeme j. Denne søgning fortsætter så længe mindst én kontakt, mellem grænsevolumenerne, opnås. Hvis der gennem hele kontaktsøgningen med trin to. I trin to søges der med udgangspunkt i de fundne minibokse, i legeme j, på tilsvarende vis som for trin ét, gennem legeme i og legeme j, påbegyndes den fine kontaktbestemmelse, ud fra de trekanter, linjer og knuder der er indeholdt i de fundne minibokse. Et 2D eksempel på anvendelsen af den hierarkiske struktur til bestemmelse af den grove kontakt, er illustreret på Figur 9.7.



Figur 9.7: Anvendelse af hierarkiet ved den grove kontaktbestemmelse. A) Kontakt mellem maksbokse. B) Trin ét. C) Trin to, hvor to kontakter mellem minibokse er identificeret.

Ved simuleringen af afkastforløbet vil der ofte være en stor forskel i legemernes størrelse. Eksempelvis er pakkerne ofte meget mindre end slisken. Ligeledes kan der pga. legemets geometriske udformning og orientering, være en betydelig distance fra periferien af kassen og ind til legemet. Hvis dette er tilfældet vil den tid det tager at erkende, at der ikke er kontakt, være kortest, hvis det er den mindste af maksboksene, der tages udgangs punkt i under trin ét i kontaktbestemmelsen, se Figur 9.8.



Figur 9.8: Til venstre søger legeme (i) ind i legeme (j), til højre omvendt. Til venstre kommer legemerne tætter på hinanden før trin to i den grove kontaktalgoritme aktiveres, og dermed reduceres den tid det tager at erkende at der ikke er kontakt.

## 9.5. Fin kontaktbestemmelse

Den fine kontaktbestemmelse skal, hvis den grove kontaktbestemmelse har fundet mindst to minibokse i kontakt, afgøre om der er kontakt mellem legemer eller ej. Er der kontakt skal algoritmen give et output i form af kontaktpunktets placering samt penetreringsdybden. Outputtet anvendes herefter til at bestemme strafkraften mellem legemerne. I den fine kontaktbestemmelsesalgoritme indgår der, pga. diskritiseringen af geometrien, kun lineære ligninger med tre ubekendte. Dermed er det ikke nødvendigt at anvende numeriske inverteringsalgoritmer, hvilket reducerer beregningstiden betydeligt.

### 9.5.1. Analyse af kontakt

Ved diskritiseringen af geometrien i knuder, linjer og flader kan kontakten mellem to legemer ske gennem følgende seks kombinationer: Knude-Knude, Knude-Linje, Knude-Flade, Linje-Linje, Linje-Flade eller Flade-Flade. I henhold til (Goyal, et al., 1994 I), kan disse inddeles i to typer, hvor alle kontaktformer undtagen Knude-Flade og Linje-Linje betegnes: degenereret kontakt. Er der en degenereret kontakt mellem to legemer, vil kontakten altid kunne konverteres til en række af ikke-degenererede kontakter, dvs. Knude-Flade og Linje-Linje kontakt. På Figur 9.9, er vist tre eksempler på konverteringer af degenererede kontakter til ikke-degenererede. Eksempel A, er en Flade-Flade kontakt, hvor hele den lille kasse hviler på en større kasse. Ved en degenerering kan kontakten formuleres som fire Knude-Flade kontakter. Eksempel B, er en Flade-Flade kontakt, hvor den lille kasse er rykket ud til hjørnet af den store kasse. I dette tilfælde kan kontakten beskrives som to Linje-Linje og én Knude-Flade kontakt. Eksempel C, er en Linje-Flade kontakt der konverteres til én Linje-Linje og én Knude-Flade kontakt.



Figur 9.9: Degenerering af kontakter. A) Flade-Flade bliver til fire Knude-Flade kontakter, B) Flade-Flade bliver til to Linje-Linje og én Knude-Flade kontakt. C) Linje-Flade bliver til én Knude-Flade og én Linje-Linje kontakt.

Eftersom de degenererede kontakttyper altid kan repræsenteres af flere ikke degenererede kontakttyper kan kontaktbestemmelsesalgoritmen reduceres, således at kun Knude-Flade og Linje-Linje kontakt undersøges. Ved konvekse geometrier opstår der dog problemer, hvormed det er nødvendigt at implementere Knude-Linje kontakt. I det følgende beskrives de i simuleringsprogrammet anvendte kontaktalgoritmer for Knude-Flade, Linje-Linje og Knude-Linje kontakt.

#### 9.5.2. Knude-Flade kontakt

Bestemmelsen af Knude-Flade kontakt udføres ved at undersøge knuderne fra det ene legemes minibokse, mod trekanterne i det andet legemes minibokse og efterfølgende modsat. Kontaktbestemmelsen mellem knuder og flader kan formuleres ved en generaliseret undersøgelse af sammenhængen mellem punktet  $P_j$  i legeme *j* og trekanten *T* i legeme *i*, udspænd af punkterne  $P_i$ ,  $Q_i$  og  $R_i$ , se Figur 9.10.



Figur 9.10: Udsnit af legemerne i og j, i form af trekanten T og punktet  $P_j$ . Trekanten T er udspændt af punkterne  $P_i$ ,  $Q_i$ ,  $R_i$  der giver vektorerne **a** og **b** samt normalvektoren **w**.

Hvis en knude og en trekant er i kontakt skal to betingelser være opfyldt. Punktet  $P_j$  skal være indeholdt i trekantens rum, givet ved trekantens flade og normalvektor. Ligeledes skal afstanden vinkelret mellem knuden og trekantens plan være mindre end summen af de to legemers kontaktlag. Er begge betingelser opfyldt er der kontakt mellem legemerne og en strafkraft påtrykkes i kontaktpunktet. Trekanten, *T* vist på Figur 9.10 er givet i globale koordinater, hvor vektorerne **a** og **b** samt trekantens normalvektor **w** bestemmes ved.

$$\mathbf{a} = \overrightarrow{P_i R_i}$$
,  $\mathbf{b} = \overrightarrow{P_i Q_i}$ ,  $\mathbf{w} = \widetilde{\mathbf{a}} \cdot \mathbf{b}$  (9.7)

Med konstanterne  $k_1$ ,  $k_2$  og  $k_3$  kan samtlige punkter  $P_k$  i rummet beskrives ved en parameterfremstilling.

$$P_i + \mathbf{a}k_1 + \mathbf{b}k_2 + \mathbf{u}_{\mathbf{w}}k_3 = P_k \tag{9.8}$$

Hvor  $\mathbf{u}_{w}$  er planets enhedsnormalvektor.

$$\mathbf{u}_{\mathbf{w}} = \frac{\mathbf{w}}{\left|\mathbf{w}\right|}$$

Relationen mellem trekanten T og punktet  $P_j$  findes ved at løse parameterfremstillingen (9.8) med hensyn til punktet.

$$P_i + \mathbf{a}k_1 + \mathbf{b}k_2 + \mathbf{u}_{\mathbf{w}}k_3 = P_i \tag{9.9}$$

Flyttes  $P_i$  over på venstre side kan (9.9) skrives på følgende matrixform.

$$\begin{bmatrix} x_a & x_b & x_u \\ y_a & y_b & y_u \\ z_a & z_b & z_u \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \\ k_3 \end{bmatrix} = \begin{bmatrix} x_j \\ y_j \\ z_j \end{bmatrix} - \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix}$$
(9.10)

Vektoren **k** for konstanterne  $k_1$ ,  $k_2$  og  $k_3$ , bestemmes ved at løse ligningssystemet.

$$[\mathbf{k}] = [\mathbf{V}]^{-1} [\mathbf{r}]$$

Hvis punktet  $P_j$  skal være indeholdt i trekantens rum, skal følgende betingelser for konstanterne  $k_1$  og  $k_2$  være opfyldt.

$$k_1 \ge 0, \ k_2 \ge 0, \ k_1 + k_2 \le 1$$
 (9.12)

Konstanten  $k_3$  svarer til den vinkelrette afstand fra trekanten ud til punktet  $P_j$ , hvor betingelsen for kontakt er givet ved.

$$k_3 = l < (\delta_i + \delta_j) \tag{9.13}$$

Hvor  $\delta$  er kontaktlagets tykkelse og *l* er den korteste vinkelrette afstand mellem trekanten og knuden. Er betingelserne (9.12) og (9.13) opfyldt er der kontakt mellem punktet og trekanten, hvorefter strafkraften bestemmes.

Vektorerne  $\mathbf{s}_i$  og  $\mathbf{s}_j$  fra legemernes origo til kontaktpunktet, givet i globale koordinater, bestemmes ved.

$$\mathbf{s}_{i} = \mathbf{o}_{i} + \mathbf{a}k_{1} + \mathbf{b}k_{2} + \mathbf{u}_{w} \left( \delta_{i} - \frac{k_{Eq} \cdot \Delta u}{k_{i}} \right)$$
  
$$\mathbf{s}_{j} = \mathbf{o}_{j} - \mathbf{u}_{w} \left( \delta_{j} - \frac{k_{Eq} \cdot \Delta u}{k_{j}} \right)$$
  
(9.14)

Hvor  $\mathbf{o}_i$  og  $\mathbf{o}_j$  er vektorerne fra legemernes origo til punkterne  $P_i$  og  $P_j$  givet i globale koordinater. Vektorerne  $\mathbf{s}_i$  og  $\mathbf{s}_j$  anvendes til bestemmelse af den relative hastighed, og momenter i kontaktpunktet, som beskrevet i afsnit 7.3.2.

Er betingelserne i (9.12) og (9.13) ikke opfyldt undersøges den næste sæt af knuder og flader.

#### 9.5.3. Linje-Linje kontakt

Linje-Linje kontaktalgoritmen undersøger for kontakt mellem samtlige linjer indeholdt i de fundne minibokse, for legeme *i*, mod samtlige linjer indehold i miniboksene fundet i legeme *j*. Med udgangspunkt i to linjer for hhv. legeme *i* og legeme *j* vist på Figur 9.11, undersøges det om disse er i kontakt.



Figur 9.11: To linjer for hhv. legeme i og j udspændt af vektoren  $v_i$  og  $v_j$ . Normalvektoren w fra legeme i til j, er ortogonal til de to linjer.

Linjen i legemet *i* er givet ved vektoren  $\mathbf{v}_i$  udspændt af punkterne  $P_j$  og  $Q_j$ , og linjen i legemt *j* udspændt af vektoren er givet ved punkterne  $P_j$  og  $Q_j$ . Ortogonalt til vektorerne  $\mathbf{v}_i$  og  $\mathbf{v}_j$  er normalvektoren  $\mathbf{w}$  givet ved krydsproduktet mellem de to linjevektorer.

$$\mathbf{w} = \tilde{\mathbf{v}}_i \cdot \mathbf{v}_j \tag{9.15}$$

Ved hjælp af en parameterfremstilling er følgende sammenhæng mellem vektorerne  $\mathbf{v}_i$ ,  $\mathbf{u}_w$  og  $\mathbf{v}_j$  givet.

$$P_i + \mathbf{v}_i k_1 - P_j - \mathbf{v}_j k_2 + \mathbf{u}_w k_3 = 0$$
(9.16)

Hvor  $\mathbf{u}_{\mathbf{w}}$  er enhedsnormalvektoren til  $\mathbf{w}$ . Flyttes punkterne  $P_i$  og  $P_j$  på højre side af ligningssystemet, kan ligningssystemet løses med hensyn til konstanterne  $k_1$ ,  $k_2$  og  $k_3$ .

$$\begin{bmatrix} \mathbf{V} \end{bmatrix} \begin{bmatrix} \mathbf{k} \end{bmatrix} = \begin{bmatrix} \mathbf{r} \end{bmatrix}$$

$$\Downarrow$$

$$\begin{bmatrix} \mathbf{k} \end{bmatrix} = \begin{bmatrix} \mathbf{V} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{r} \end{bmatrix}$$
(9.17)

To betingelser skal være opfyldt, for at de to linjer er i kontakt. Den korteste afstand mellem linjerne skal ligge indenfor og vinkelret på vektorerne  $\mathbf{v}_i$  og  $\mathbf{v}_j$ . Den korteste afstand mellem linjerne skal være mindre end summen af de to legemers kontaktlag. Dette formuleres med konstanterne  $k_1$ ,  $k_2$  og  $k_3$ , ved.

$$0 \le k_1 \le 1, \ 0 \le k_2 \le 1, \ k_3 = l < (\delta_i + \delta_j)$$
(9.18)

Er betingelserne i (9.18) opfyldt er der kontakt og strafkræfterne påtrykkes samme retning som enhedsnormalvektoren  $\mathbf{u}_{w}$  i kontaktpunktet. Tilsvarende, som ved Knude-Flade kontakt, bestemmes vektorerne fra legemernes koordinatsystemer ud til kontaktpunktet for de to legemer ved.

$$\mathbf{s}_{i} = \mathbf{o}_{i} + \mathbf{v}_{i}k_{1} + \mathbf{u}_{\mathbf{w}}\left(\delta_{i} - \frac{k_{Eq} \cdot \Delta u}{k_{i}}\right)$$
  
$$\mathbf{s}_{j} = \mathbf{o}_{j} + \mathbf{v}_{j}k_{2} - \mathbf{u}_{\mathbf{w}}\left(\delta_{j} - \frac{k_{Eq} \cdot \Delta u}{k_{j}}\right)$$
  
(9.19)

Hvor  $\mathbf{o}_i$  og  $\mathbf{o}_j$  er vektorerne fra legemernes origo til punkterne  $P_i$  og  $P_j$  givet i globale koordinater.

Er betingelserne i ligning (9.18) ikke opfyldt er der ikke kontakt og det næste sæt af linjer undersøges.

#### Reduktion i antallet af linjer

Visse linjer i legemernes geometri vil aldrig komme i kontakt med andre legemer. Det gælder de linjer, hvor de tilstødende trekanter er i plane med hinanden. Derfor er der implementeret en algoritme der reducere antallet af linjer, for derved at minimere beregningstiden. Denne algoritme er implementeret i simuleringsprogrammet under *Reduce\_no\_of\_Lines* i modulet *InitialCalc*.

### 9.5.4. Kontakt i konvekse hjørner

Med ovenstående kontaktformulering opstår situationer, hvor kontakten mellem to legemer ikke bestemmes korrekt. Dette kan forekomme når to trekanter tilsammen danner et konvekst hjørne, se Figur 9.12.



Figur 9.12: To trekanter som danner et konveks hjørne, hvor det grå område markerer trekanternes rum. Det konvekse hjørne giver anledning til en åbning, hvor kontakt ikke registreres med Linje-Linje eller Knude-Linje kontakt.

De to gule trekanter er en del af geometrien for legeme *i* og er i en vinkel til hinanden, således at de danner et udadvendt hjørne. Det område hvor trekanten vil være i kontakt med et punkt fra legeme *j* er markeret med gråt. Da dette rum peger i trekantens normalretning, bliver der en åbning mellem trekanterne. Det betyder at punktet  $P_j$  fra legeme *j*, ikke påvirkes af en kraft, hvis den befinder sig i mellemrummet, men først kommer i kontakt, når den rammer trekanternes rum eller trænger igennem hjørnet. Dette giver en fejl i kontaktformuleringen. For at undgå dette indføres en Knude-Linje kontaktformulering der dækker åbninger mellem trekanterne i konvekse hjørner.

#### 9.5.5. Knude-Linje kontakt

Ved Knude-Linje kontakt undersøges de knuder indeholdt i de fundne minibokse fra legeme *i* mod linjer indeholdt i de fundne minibokse fra legeme *j*, og efterfølgende modsat. Der tages udgangspunkt i et generelt tilfælde, hvor punktet  $P_j$  fra legeme *j* undersøges mod et konvekst hjørne i legemet *i*. Rummet, trekanterne i det konvekse hjørne ikke dækker, kan beskrives ved linjen  $\mathbf{v}_{ab}$  samt enhedsnormalvektorerne  $\mathbf{u}_a$  og  $\mathbf{u}_b$  til de to tilstødende trekanter, se Figur 9.13.



Figur 9.13: Trekanterne a og b i legemet i, danner et konvekst hjørne, hvis rum kan beskrives af trekanternes enhedsnormalvektorer  $\mathbf{u}_a$  og  $\mathbf{u}_b$ , samt linjen  $\mathbf{v}_{ab}$ . Er punktet  $P_j$  inde i det udspændte rum er der kontakt med linjen  $\mathbf{v}_{ab}$ .

Rummet, markeret med stiplet på Figur 9.13, som udspændes af vektorerne  $\mathbf{v}_{ab}$ ,  $\mathbf{u}_a$  og  $\mathbf{u}_b$ , kan beskrives ved en parameterfremstilling, med konstanterne  $k_1$ ,  $k_2$  og  $k_3$ , hvilket løses med hensyn til punktet  $P_i$ .

$$P_i + \mathbf{v}_{ab}k_1 + \mathbf{u}_a k_2 + \mathbf{u}_b k_3 = P_j \tag{9.20}$$

For at punktet  $P_j$  er inde i rummet udspændt af vektorerne  $\mathbf{v}_{ab}$ ,  $\mathbf{u}_a$  og  $\mathbf{u}_b$ , skal følgende betingelser være opfyldt.

$$0 \le k_1 \le 1$$
  $0 \le k_2$   $0 \le k_3$  (9.21)

Ligeledes skal den korteste afstand mellem punktet og linjen være mindre end summen af de to legemers kontaktlag.

$$l = |\mathbf{l}| < \left(\delta_i + \delta_j\right) \tag{9.22}$$

Hvor I er den vinkelrette afstandsvektor til linjen bestemt ved.

$$\mathbf{l} = P_i + \mathbf{v}_{ab} k_1 - P_j \tag{9.23}$$

Vektorerne fra legemernes koordinatsystem til kontaktpunktet i globale koordinater bestemmes ved.

$$\mathbf{s}_{i} = \mathbf{o}_{i} + \mathbf{v}_{ab} k_{1} + \mathbf{u}_{l} \left( \delta_{i} - \frac{k_{Eq} \cdot \Delta u}{k_{i}} \right)$$
  
$$\mathbf{s}_{j} = \mathbf{o}_{j} - \mathbf{u}_{l} \left( \delta_{j} - \frac{k_{Eq} \cdot \Delta u}{k_{j}} \right)$$
  
(9.24)

Hvor  $\mathbf{u}_{\mathbf{l}}$  er enhedsnormalvektoren for vektoren  $\mathbf{l}$ .

## 9.6. Reduktion i antallet at kontaktsøgninger

Eftersom kontaktbestemmelsen mellem legemerne i simuleringsprogrammet er en beregningsmæssig tung opgave, er det valg ikke at søge efter nye kontakter ved hvert tidsstep. Det er gjort ved at anvende kontakterne fra forrige tidsstep til bestemmelse af reaktionskræfterne til det pågældende tidsstep. Dette giver anledning til en fejl idet nye kontakter ikke indgår i reaktionskræfterne. Det er valgt kun at opdatere kontakterne hvert femte tidsstep. Den manglende opdatering af kontakter er vurderet til ikke at have nogen nævneværdig betydning for simuleringens samlede præcision.

# 9.7. Implementering i simuleringsprogrammet

I det følgende er implementeringen af den grove og den fine kontaktalgoritme beskrevet.

### 9.7.1. Generering af kontakthierarkiet i simuleringsprogrammet

I simuleringsprogrammet genereres kontakthierarkiet med subrutinerne *MiniBox*, *MaxBox* og *CounBox* i modulet *Initialcalc*.

*MiniBox* rutinen generer minimum- og maksimumsværdierne for miniboksene omkring hver enkelt trekant. De fælles knuder og linjer for trekanterne bliver fordelt til de enkelte minibokse, efter først til mølle princippet. Minimum- og maksimumsværdierne for miniboksene er for de bevægelige legemer givet i lokale koordinater og for de faste legemer givet i globale koordinater. Værdierne for miniboksene gemmes i det globale array *G* ved variablerne *G%MiniB* og *G%MiniA*, hvor *G%MiniB* indeholder minimum- og maksimumsværdierne for miniboksene samt positionen af trekantens tyngdepunkt. *G%MiniA* indeholder oplysninger om de pågældende minibokses knuder, linjer samt trekant.

MaxBox rutinen generer maksbokse omkring hele legemet, ud fra de minibokse der er tilknyttet det enkelte legeme, ved at vælge den mindste minimumsværdi og den største maksimumsværdi. Minimum- og maksimumsværdierne for maksboksene gemmes i det globale array *G* under *G%MaxB*.

*CounBox* rutinen genererer alle kasserne i niveauerne mellem, maksboksene og miniboksene, samt generer sammenhængen mellem kasserne. I *G%BoksH* gemmes oplysningerne omkring, hvor mange og hvilke undergrænsevolumener hver enkelt grænsevolumen indeholder. Er nummeret på grænsevolumen negativ er kassen en miniboks.

*CounBox* er bygget op omkring to løkker. Den yderste af løkkerne, gennemløber alle legemerne mens den anden, en *while* løkke, holder opdelingsprocessen i gang indtil det ikke længere er muligt at opdele grænsevolumenerne i flere underniveauer. Når dette er tilfældet, brydes *while* løkken, og opdelingen af det næste legeme påbegyndes. Listerne for de generede grænsevolumener, mellem maksboksen og miniboksene, gemmens i variablen *G%BoxM*.

## 9.7.2. Implementering af grov kontaktalgoritme

Den grove kontaktalgoritme er i simuleringsprogrammet implementeret sammen med bestemmelsen af kræfter og momenter, i rutinen *For\_Mom\_Calc* i *Dynamic* modulet samt subrutinen *BoxContact* i modulet *Geohandling*. I *For\_Mom\_Cal* kontrolleres der for kontakt mellem maksboksene og i *BoxContact* for kontakt mellem kasserne i niveauerne lavere. Koden for *For\_Mom\_Calc* er bygget op omkring to løkker, der begge tæller legemenummeret. Den første betingelse undersøger for kontakt mellem maxboxene. Er der kontakt kaldes *BoxContact* der, gennem relationerne i kontakthierarkiet, kontrollerer for kontakt mellem miniboksene. Er der kontakt mellem miniboksene starter den fine kontaktbestemmelse. Den fine kontaktbestemmelse er givet ved subrutinerne *Trig\_Contact, Line\_Contact* og *Edge\_Edge\_Contact*. Bestemmes der kontakt i en af rutinerne, returneres kontaktkraften og momentet, hvilket efterfølgende summeres i de globale matricer  $\mathbf{f}_{i,G}$  og  $\mathbf{n}'_{i,G}$ . Når løkkerne er gennemløbet samtlige legemer er alle kræfter i systemet bestemt, til det pågældende tidsstep, og de globale matricer for legemernes kræfter og momenter,  $\mathbf{f}_{i,G}$  og  $\mathbf{n}'_{i,G}$ , anvendes i den numeriske integration af bevægelsesligningerne.

### 9.7.3. Implementering af den fine kontaktalgoritme

Kontaktformuleringerne for Knude-Flade, Linje-Linje, Knude-Linje kontakt er givet i simuleringsprogrammet ved rutinerne *Trig\_Contact*, *Edge\_Edge\_Contact* og *Line\_Contact* i modulet *GeoHandeling*.

Følgende pseudokode for Knude-Flade kontakt er implementeret i simuleringsprogrammet, hvor der er anvendt samme nomenklatur som i den fremstillede teori.

Rutinen er bygget op omkring én løkker for antallet af punkter i de aktuelle minibokse. Indledningsvis bestemmes det om der er kontakt mellem trekanten og punktet. For hver trekant i simuleringen er information om vektoren  $\mathbf{a}'$ ,  $\mathbf{b}'$  og  $\mathbf{n}'$  gemt i en matrix i det globale array **G**. For den pågældende trekant transformeres disse til globale koordinater. Derefter bestemmes konstanterne  $k_1$ ,  $k_2$  og  $k_3$ , på tilsvarende vis som beskrevet i afsnit 9.5.2. Herefter kontrolleres betingelserne for kontakt mellem knuden og trekanten. Er der kontakt, bestemmes fjeder og dæmperkraften, på tilsvarende vis beskrevet i afsnit 7.4. Herefter bestemmes friktionskraften ved at kalde subrutinen *FrictionForce*. Denne rutine henviser derefter videre til den valgte friktionsmodel. Efterfølgende summeres kræfterne  $\mathbf{f}_{i,w}$  og  $\mathbf{f}_{j,w}$  med input-

kræfterne  $\mathbf{f}_i$  og  $\mathbf{f}_j$ , hvilket sendes tilbage til rutinen *For\_Mom\_Calc*. Tilsvarende summeres momenterne  $\mathbf{n}'_{i,n}$  og  $\mathbf{n}'_{i,m}$ .

For Linje-Linje og Knude-Linje kontakt er rutinernes generelle opbygning tilsvarende Knude-Flade.

Pseudokode:	
$Box_i$	Miniboks for legeme <i>i</i>
$Box_{j}$	Miniboks for legeme <i>j</i>
n	Punkt nummer
т	Trekant nummer
Ν	Normalretning
Т	I tangentplanet
$Pkt_i$	Antal punkter i $Box_i$
P'	Knude i lokale koordinater
a',b',u'	Lokale vektore for trekanten
subroutine T	$Trig\_Contact(Box_i, Box_j, m, i, j, \mathbf{f}_i, \mathbf{f}_j, \mathbf{n}'_i, \mathbf{n}'_j, \mathbf{r}_i, \mathbf{r}_j, \dot{\mathbf{r}}_i, \mathbf{A}_i, \mathbf{A}_j, \tilde{\boldsymbol{\omega}}_i, \tilde{\boldsymbol{\omega}}_j, \mathbf{G})$
do n	$e = 1, Pkt_i$
	$\mathbf{o}_{i,n} = \mathbf{A}_i^T P_{i,n}'; \ \mathbf{o}_{j,m} = \mathbf{A}_j^T P_{j,m}'$
	$\mathbf{a}_{j,m} = \mathbf{A}_{j}^{T} \mathbf{a}_{j,m}^{\prime}; \ \mathbf{b}_{j,m} = \mathbf{A}_{j}^{T} \mathbf{b}_{j,m}^{\prime}; \ \mathbf{u}_{j,m} = \mathbf{A}_{j}^{T} \mathbf{u}_{j,m}^{\prime}$
	$\mathbf{r} = \mathbf{r}_i + 0_{i,m} - 0_{j,m} - \mathbf{r}_j$
	$\mathbf{V} = [\mathbf{a}_{j,m}, \mathbf{b}_{j,m}, \mathbf{u}_{j,m}]$
	$\mathbf{k} = \mathbf{V}^{-1}\mathbf{r}$
	<i>if</i> $(k_1 \ge 0 \& k_2 \ge 0 \& k_1 + k_2 \le 1 \& (\delta_i + \delta_j) > k_3)$ <i>then</i>
	$\Delta u = \left(\delta_i + \delta_j\right) - \left k_3\right $
	$\mathbf{f}_{i,n} = \mathbf{u}_{j,m} \cdot k_{eq} \cdot \Delta u$
	$\mathbf{s}_{i,P} = \mathbf{o}_{i,n} + \mathbf{a}_{j,m} k_1 + \mathbf{b}_{j,m} k_2 + \mathbf{u}_{j,m} \left( \delta_i - \frac{k_{Eq} \cdot \Delta u}{k_i} \right)$
	$\mathbf{s}_{j,P} = \mathbf{o}_{j,m} - \mathbf{u}_{j,m} \left( \delta_j - \frac{k_{Eq} \cdot \Delta u}{k_j} \right)$
	$\dot{\mathbf{r}}_{p} = \dot{\mathbf{r}}_{i} + \tilde{\boldsymbol{\omega}}_{i} \mathbf{s}_{i,p} - \dot{\mathbf{r}}_{j} - \tilde{\boldsymbol{\omega}}_{j} \mathbf{s}_{j,p}$
	$\dot{\mathbf{r}}_{P,N} = \frac{\mathbf{u}_{j,m} \cdot \dot{\mathbf{r}}_{P}}{\left \mathbf{u}_{j,m}\right } \mathbf{u}_{j,m}$
	$\mathbf{f}_{i,n} = \mathbf{f}_{i,n} + \mathbf{u}_{j,m} \left( c_{Eq} \cdot \left  \dot{\mathbf{r}}_{P,N} \right  \right)$
	$\dot{\mathbf{r}}_{P,T} = \dot{\mathbf{r}}_P - \dot{\mathbf{r}}_{P,N}$
	<i>call</i> FrictionForce $(i, j, n, m, \mathbf{f}_{i,n}, \dot{\mathbf{r}}_{P,T}, \mathbf{G})$
	$\mathbf{f}_{_{j,m}}=-\mathbf{f}_{_{i,n}}$
	$\mathbf{n}_{i,n}' = \mathbf{A}^T \tilde{\mathbf{s}}_{i,P} \mathbf{f}_{i,n}  ; \ \mathbf{n}_{j,m}' = \mathbf{A}^T \tilde{\mathbf{s}}_{j,P} \cdot \mathbf{f}_{j,m}$
	$\mathbf{f}_i = \mathbf{f}_i + \mathbf{f}_{i,n}$
	÷
	$\mathbf{n}_{i}' = \mathbf{n}_{i}' + \mathbf{n}_{i}'$
	end if
end	do
end subrouti	ne
1	

# 10. Indledende databehandling

Dette kapitel giver en beskrivelse af de indledende beregninger, der foretages i simuleringsprogrammet. På Figur 10.1 er den overordnede struktur af simuleringsdelen af programmet illustreret.

I simuleringsprogrammet består kernen af en dynamisk og en kontaktbestemmende del. Inden simuleringsberegningerne påbegyndes udføres en indledende databehandling, hvor inputfilerne for simuleringen indlæses og behandles. Efter simuleringsberegningerne er udført, udskrives resultatfilerne, i form af positioner, hastigheder, accelerationer og energi, for hvert legeme i simuleringen.



Figur 10.1: Simuleringsprogrammets generelle opbygning, hvor kernen består af en dynamik og en kontaktbestemmende del. Ved eksekvering af programmet indlæses inputfilerne, der databehandles inden simuleringsberegningerne foretages. Efter simuleringsberegningerne, udføres en resultatbehandling og der udskrives outputfiler.

Den indledende databehandling, for simuleringsdelen, foregår i følgende fire trin:

- Indlæsning af inputfiler.
- Behandling af geometri.
- Bestemmelse af masseinertimoment.
- Bestemmelse af fjederstivheder, dæmperkonstanter og tidsstep.

Punkterne er i det følgende beskrevet nærmere, undtagen punkt fire, der er beskrevet i afsnit 7.4.

# 10.1. Inputfil

Som input til programmet er der konstrueret en række inputfiler, hvor data omkring simulering og legemerne er defineret. Alle dataene i inputfilerne indlæses i programmet, hvorefter de anvendes til simuleringsberegningerne. Der er fire forskellige inputfiler, i form af: *materiale, inerti, bevægelse* og *simulering*. Filerne vil i det efterfølgende blive gennemgået i den nævnte rækkefølge. I simuleringsprogrammet indlæses input- samt geometrifilerne med subrutinerne *ReadFile, WRL\_Read, WRL\_Read\_nr* og *WRL\_Read\_PK* i modulet *ReadInput*. En nærmere beskrivelse af inputfilerne findes i Appendiks IX.

### Materialeinputfil

Materialeinputfilen fungerer som en database, hvor informationer om friktionsparametre og restitutionskoefficienter på forskellige materialer, kan defineres. De angivne materialenavne i materialeinputfilen anvendes til at knytte materialernes egenskaber til legemerne, i simule-ringsinputfilen.

### Inertiinputfil

I inertiinputfilen angives inertimomentet for det pågældende legeme, hvis der ønskes en mere præcis angivelse. Filen angives i inputfilen til simuleringsprogrammet

### Bevægelsesinputfil

Bevægelsesinputfilen indeholder en forskreven bevægelse, der kan anvendes til de bevægelige legemer i simuleringsprogrammet. I filen, gives på liste form, positionen og orienteringen af legemets lokale koordinatsystem i forhold til det globale. Filen angives i inputfilen til simuleringsprogrammet

### Simuleringsinputfil

I starten af inputfilen angives det, hvor lang tid simuleringen skal køre, navnet på den anvendte materialefil, samt hvilken friktionsmodel der skal anvendes. Efterfølgende indtastes oplysninger om de, i simuleringen, anvendte legemer. Legemerne er som beskrevet i afsnit 6.4, inddelt i tre typer, i form af, frie, bevægelige og faste legemer. For de frie legemer skal følgende oplysninger gives:

- Antal frie legemer
- Navn på legemernes geometrifil
- Legemernes masse
- Legemernes materiale
- Geometri anvendt til bestemmelse af masseinerti eller navn på inertiinputfilen
- Tyngdepunktets placering
- Koncentrationen af massen i legemet
- Start position
- Start hastighed

For de faste legemer skal oplysninger som navnet på geometrifilen, materiale, samt position af legemerne indføres.

Ved de bevægelige legemer skal følgende oplysninger angives:

- Antal bevægelige legemer
- Navn på legemernes geometrifil
- Legemernes materiale
- Navn på bevægelsesfil
- Starttidspunkt for eksekvering af bevægelsesfil
- Start position
- Start hastighed

Geometrierne anvendt til simuleringsprogrammet tegnes i SolidWortk som en part, hvorefter de gemmes som en wrl-fil. Diskretiseringsgraden af de enkelte parts bestemmes i menuen for Egenskaber i SolidWorks. Ved indlæsning af filen importeres data om partens knuder samt hvilken relationer der er mellem disse.

# 10.2. Behandling af geometri

Efter indlæsning af geometrierne i simuleringsprogrammet behandles disse på en række punkter. De frie legemer behandles separat på to punkter. Indledningsvis flyttes de frie legemers lokale koordinatsystem, givet i SolidWorks, ind til legemets geometriske centrum. Herefter flyttes det lokale koordinatsystem ud i legemets tyngdepunkt ud fra vektoren angivet i inputfilen til simuleringen. Herefter påføres de frie legemer et kontaktlag. I det følgende er teorien anvendt i forbindelse med generering af kontaktlaget beskrevet.

## 10.2.1. Kontaktlag

I hht. kontaktformuleringen beskrevet i afsnit 7.3 er legemerne omsluttet af et kontaktlag til bestemmelse af reaktionskræfterne. For at kompensere for det pålagte kontaktlag, offsettes legemets overflader med kontaktlagets tykkelse.

I simuleringsprogrammet er det valgt at anvende metoden, givet af (Xiuzhi, et al., 2003) der tager udgangspunkt i et offset af legemernes knuder, hvormed den oprindelige relation mellem knuderne kan bibeholdes. Metoden tager udgangspunkt i en vægtet sum af normalvektorerne, for den enkelte knudes tilstødende trekanter, hvilket giver en flytning og en retning af knuden.

Knuden *P* er givet mht. legemets lokale koordinatsystem med vektoren  $\mathbf{p}_i$ . Flytningen af knuden  $\mathbf{p}_i^g$  i legemet *i* til punktet beskrevet ved vektoren  $\mathbf{p}_i^n$ , er illustreret på Figur 10.2.



Figur 10.2: Knuden P flyttes mht. legemets lokale koordinatsystem ud fra offsetvektoren og kontaktlagets tykkelse.

Vektoren  $\mathbf{p}_i^n$  er givet ved følgende sammenhæng.

$$\mathbf{p}_i^n = \mathbf{p}_i^g + \mathbf{v}_i^o \cdot \delta \tag{10.1}$$

Hvor  $\delta$  er kontaktlagets tykkelse. Vektoren  $\mathbf{v}_i^o$  bestemmes ud af normalevektorerne til de tilstødende trekanter, samt en vægtningsfaktor  $w_i$ , tilknyttet hver enkelt trekant.

$$\mathbf{v}_i^o = \sum_{j=1}^N w_j \cdot \mathbf{u}_{i,j}$$
(10.2)

Hvor *N* er antallet af tilstødende trekanter. Det antages at kontaktlagets tykkelse  $\delta$  er konstant for samtlige tilstødende trekanter, samt at samtlige trekanters planer skærer i punktet  $\mathbf{p}_{i}^{n}$ , hvormed det gælder at.

$$\left(\mathbf{p}_{i}^{n}-\mathbf{p}_{i}^{g}\right)^{T}\mathbf{u}_{i,j}=\delta$$
(10.3)

Indsætte ligning (10.3) i ligning (10.1) fås.

$$\left(\mathbf{v}_{i}^{0}\right)^{T}\mathbf{u}_{i,j}=1$$
(10.4)

Indsættes ligning (10.2), i (10.4) fås

$$\left(\sum_{j=1}^{N} w_j \cdot \mathbf{u}_{i,j}\right)^T \mathbf{u}_{i,k} = 1 \qquad , k = 1...N$$
(10.5)

Hvilken kan skrives på matrixform.

$$\begin{bmatrix} \left(\mathbf{u}_{i,1}\right)^{T} \mathbf{u}_{i,1} & \left(\mathbf{u}_{i,2}\right)^{T} \mathbf{u}_{i,1} & \cdots & \left(\mathbf{u}_{i,N}\right)^{T} \mathbf{u}_{i,1} \\ \left(\mathbf{u}_{i,1}\right)^{T} \mathbf{u}_{i,2} & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ \left(\mathbf{u}_{i,1}\right)^{T} \mathbf{u}_{i,N} & \cdots & \left(\mathbf{u}_{i,N-1}\right)^{T} \mathbf{u}_{i,N} & \left(\mathbf{u}_{i,N}\right)^{T} \mathbf{u}_{i,N} \end{bmatrix} \begin{bmatrix} w_{1} \\ w_{2} \\ \vdots \\ w_{N} \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}$$
(10.6)

Løses ligning (10.6) bestemmes vægtningsfaktorerne,  $w_j$  for samtlige tilstødende trekanter, hvormed vektoren  $\mathbf{p}_i^n$  kan bestemmes.

#### Særtilfælde

I forbindelse med anvendelsen af ovenstående offsetmetode, er der to særtilfælde.

I geometrien kan det ikke udelukkes at flere trekanters enhedsnormalvektorer en ens. Er det tilfældet eliminere duplikater således at det kun er én enhedsnormalvektor der repræsenterer disse.

Det andet tilfælde forekommer hvis der er mere end tre tilstødende trekanter til en knude, efter at evt. duplikerede enhedsnormalvektorer er fjernet. Eftersom der ved fire eller flere trekanter ikke er nogen garanti for at der forekommer et fælles skæringspunkt, ved flytning af knuden, vil der med ovenstående offsetformulering være situationer, hvor der ikke kan findes en løsning for ligning (10.6), (Xiuzhi, et al., 2003).

For at afhjælpe dette, bliver de *N* enhedsnormalvektorer inddelt i *N* undergrupper, bestående af tre enhedsnormalvektorer givet ved.

 $(\mathbf{u}_{i,1} \ \mathbf{u}_{i,2} \ \mathbf{u}_{i,3}), (\mathbf{u}_{i,2} \ \mathbf{u}_{i,3} \ \mathbf{u}_{i,4}) \dots (\mathbf{u}_{i,N} \ \mathbf{u}_{i,1} \ \mathbf{u}_{i,2})$  (10.7)

Undergrupperne giver *N* offsetvektorer,  $\mathbf{v}^{o}_{1\cdots N}$ , hvorudfra  $\mathbf{v}^{o}$  bestemmes som middelværdivektoren af disse. Præcisionen af  $\mathbf{v}^{o}$  afhænger af om de fundne offsetvektorer har et fælles skæringspunkt. Er det tilfældet vil gennemsnittet giver en eksakt løsning, hvorimod resultat bliver en approksimation, hvis det ikke er tilfældet (Xiuzhi, et al., 2003).

### Implementering af offsetalgoritmen

Offsetalgoritmen er implementeret i subrutinen *Make\_Contactlayer* under modulet *Initial-Calc*.

Det er i simuleringsprogrammet valgt at pålægge de frie legemer det dobbelte kontaktlag. Herved undgås det at anvende offsetalgoritmen på de faste og bevægelige legemer. Dette er fordelagtigt da de frie legemer oftest har den simpleste geometri, og risikoen for offsetfejl derved reduceres. Det dobbelte kontaktlag medfører, ved kontakt, en fejl i kontaktpunktets placering. Kompensationen for dette udføres ved at flytte kontaktpunktet, længden svarende til det ene legemes sammenpressede fjeder, modsat fjederkraftens retning.

## 10.3. Masseinertimoment

I forbindelse med rotationsbevægelsesligningen anvendes masseinertimomentet til bestemmelse af vinkelaccelerationerne. Eftersom masseinertimomentet er givet i lokale koordinater er det tilstrækkeligt at beregne disse én gang, indledningsvis, i programmet.

Masseinertimomentet beskriver fordelingen af legemets masse og afhænger derfor både af geometrien samt variationen af densiteten.

Masseinertimomentet kan beregnes eksternt og indlæses i simuleringsprogrammet via inputfilen til simuleringen. Som et alternativ til indlæsning af en ekstern fil er der implementeret to forsimplede formuleringer. Det er i de forsimplede formuleringer valgt at approksimere masseinertimomentet ved hjælp af en kugle eller et rektangel, der begge har en jævn fordelt masse. Alt efter hvor koncentreret massen er fordelt i pakken, kan størrelsen af massegeometrien variere. Koncentrationen af massen kan angives ved tre størrelser i form at *Smale*, *Mean* eller *Large*, hvilket svarer til 100 %, 66 % eller 33 % af hhv. *r* eller *l*·*h*·*b*.

Ved anvendelse af en kugle sættes radius til den afstand, der omslutter geometrien. Med ovenstående formulering er der tre inputparametre til simuleringsprogrammet i form af en generel geometri, tyngdepunktets placering og koncentrationen af massen. Beregning af masseinertimomentet udføres i *InvMassInerti* i modulet *InitialCalc*.

## 10.4. Energiregnskab

I simuleringsprogrammet er der implementeret et energiregnskab der har til formål at angive om systemet opfylder termodynamikkens første lov. Med udgangspunkt i at energien i simuleringsprogrammet kan beskrives som et regnskab, skal følgende sammenhæng være gældende:

Energien i systemet + Tabet af energi - Tilført energi = Konstant

$$\downarrow \qquad (10.8)$$

$$E_{mek} + E_D + E_F - E_M = \text{Konstant}$$

Hvor  $E_{mek}$  er den mekaniske energi,  $E_D$  er energitab på grund af viskøse dæmpere,  $E_F$  er energitab på grund af friktion og  $E_M$  er energi tilført til systemet.

Regnskabet udføres ved at beregne, hvad der bliver tilført og tabt af energi, til det enkelte legeme.

### 10.4.1. Energien i det mekaniske system

Energien i det mekaniske system kan defineres som summen af systemets potentielle og kinetiske energi. Den kinetiske,  $E_{kin}$  og den potentielle energi,  $E_{pon}$  er kun afhængige af tilstanden, henholdsvis hastighed og position, og ikke af vejen frem til denne tilstand. På grund af dette vil den mekaniske energi altid kunne bestemmes uafhængigt af systemets historie. Påvirkes systemet ikke af energitab eller tilførsel, vil den samlede energi altid være konstant.

$$E_{mek} = E_{pot} + E_{kin} \tag{10.9}$$

#### Kinetisk energi

Den energi der er i et legeme, som følge af dens bevægelsesmængde, benævnes legemets kinetiske energi. Da bevægelse kan opdeles i translation og rotation, vil den samlede kinetiske energi være beskrevet som summen af disse,  $E_{kin,T}$  og  $E_{kin,R}$ .

$$E_{kin,T} = \frac{1}{2}m\left|\dot{\mathbf{r}}\right|^2 \tag{10.10}$$

$$E_{kin,R} = \frac{1}{2} \boldsymbol{\omega}'^T \mathbf{J}' \boldsymbol{\omega}' \tag{10.11}$$

#### **Potentiel energi**

Potentiel energi er den energi som et legeme er i besiddelse af pga. dennes placering i rummet. Er et system påvirket af f.eks. tyngdefeltet eller fjedre, vil det være i besiddelse af potentiel energi. Legemets potentielle energi, på grund af jordens tyngdefelt,  $E_{pot,g}$  er givet ved.

$$E_{pot,g} = h \cdot m \cdot g \tag{10.12}$$

Hvor *h* er højden af legemet over nulplanet, *m* er massen af legemet og g er tyngdeaccelerationen, fastsat til 9.82  $m/s^2$ . Den potentielle energi fra en lineær fjeder,  $E_{pot,g}$  er givet ved.

$$E_{pot,S} = \frac{1}{2}k \cdot \delta^2 \tag{10.13}$$

k er fjederkonstanten og  $\delta$  er den relative ændring af fjederens længde. Den potentielle energi er for alle legemer i simuleringsprogrammet beregnet ud fra massecenteret.

#### 10.4.2. Energitab

Energitab skyldes at systemet udfører et arbejde. De i simuleringsprogrammet medregnede energitab skyldes viskøse dæmpere og friktionskræfternes arbejde.

#### Viskøs dæmpning

Modstandskraften i en lineær viskøs dæmper er afhængig af hastigheden og dæmperkoefficienten, c, hvor energitabet,  $E_D$  kan bestemmes ved.

$$E_D = \int_{t_1}^{t_2} \mathbf{r}_w^T \mathbf{r}_w \cdot c \cdot dt \qquad (10.14)$$

Hvor  $\mathbf{r}_w$  er hastighedsvektoren i dæmperens retning.

#### Friktion

Friktionen mellem to legemer giver anledning til et tab af energi i form af varme. Dette energitab,  $E_F$  kan beregnes ved.

$$E_F = \int_{t_1}^{t_2} \mathbf{f}_f^T \mathbf{r}_t dt \tag{10.15}$$

Hvor  $\mathbf{r}_t$  er den i kontaktplanet projicerede hastighed og  $\mathbf{f}_f$  er friktionskraften.

#### 10.4.3. Energitilførsel

Energitilførsel skyldes at noget udefra, udfører et arbejde på systemet. I forbindelsen med simuleringsmodellen vil de motorer der driver bakken, tilføre systemet energi. Bakken er moduleret uden masse, det vil derfor kun værre pakken, der bliver tilført energi. Det er valgt at se bort fra denne energitilførsel, i energiregnskabet, da pakken kun står på bakke i starten af simuleringsforløbet. Energiregnskabet vil derfor ikke være konstant, indtil pakken har sluppet bakken.

### 10.4.4. Implementering i simuleringsprogrammet

Simuleringsprogrammet opdaterer energiregnskabet for hvert tidsstep, hvor ligning (10.14) og (10.15) opdateres i forbindelse med 4.ordens Runge-Kutta. Beregningerne af de enkelte energier udføres i flere rutiner, hvorefter en opdatering sker i subrutinen *rk4* i modulet *Nu-mInt*. Legemets mekaniske energi, med undtagelse af fjederenergien, beregnes i forbindelse med løsning af bevægelsesligningerne i rutinen *For\_Mom\_Calc* i modulet *Dynamic*. Fjederenergien og energitabet fra dæmperen beregnes i kontaktbestemmelsesalgoritmerne i modulet *GeoHandeling*. Energitabet pga. friktion bestemmes i hver friktionsmodel i modulet *Friction*.

### 10.5. Belastning af pakker

I hht. opgaveformuleringen er der et ønske om at bestemme impacktstørrelsen på pakkerne. I forbindelse med et afkastes ned gennem en sliske, er den evt. skade pakke påføres afhængig af den hastighed hvormed kræfterne påtrykkes.

Eftersom pakkernes fjederstivheder er bestemt ud fra et maksimalt energiniveau, svarer disse ikke til pakkematerialernes reelle stivheder. Dette betyder at den påvirkning pakkerne udsættes for svare overens med virkeligheden. Det er derfor ikke muligt i simuleringsprogrammet at bestemme størrelsen af pakkernes reelle belastning, men kun at beskrive belastningskarakterstikken.

Eftersom påvirkningerne, fra reaktionskræfterne, kan være påtrykt et arbitrær sted på pakkernes overflade er det ikke hensigtsmæssigt at anvende disse til beskrivelse af pakkens belastning. Det er derfor valgt at anvende en samlet værdi i form af raten, hvormed kontaktkræfterne udfører et arbejde på pakkerne. Denne værdi, effekten *P* er givet ved.

$$P = \frac{dU}{dt} \tag{10.16}$$

Hvor U er arbejdet udført af kontaktkræfterne, givet ved.

$$U = E_{pot,S} + E_D + E_F$$
 (10.17)

I simuleringsprogrammet bestemmes den effekt pakkerne afsætter, ud fra en numerisk differentiation af energien U givet ved (Kreyszig, 1999 s. 880).

$$f_2' \approx \frac{1}{12h} \left( f_0 - 8f_1 + 8f_3 - f_4 \right)$$
(10.18)

Belastningen af pakkerne beregnes i modulerne dynamic og output.

## 10.6. Visualisering

Efter simuleringsberegningerne udskrives en række outputfiler til visualisering af afkastforløbet.

Indledningsvist udskrives to filer for hvert legeme, indeholdende hhv. geometriens knudekoordinater samt koordinaternes sammenhæng.

Inden programmet afsluttes, udskrives filer for position, hastighed, acceleration, energi og belastninger, for hvert legeme. I positionsfilerne er positionen for legemet givet 50 gange pr. sek. med de tre positionskoordinater og de fire Euler parametre. Tilsvarende indeholder hastighed- og accelerationsfilerne oplysninger om legemets hastighed og acceleration, givet 50 gange pr. sek. Energifilerne indeholder information om legemernes energi gennem hele simuleringen. Efter at alle filerne er udskrevet afsluttes programmet. Visualiseringen af den beregnede simulering foretages i MATLAB ved at køre filen *Graphic.m.* Under visualiseringen er pakkernes belastning illustreret ved en ændring af pakkens farve, hvor hvid angiver ingen belastning og rød angiver pakkens maksimale belastning. Udskrivningen af filerne sker i modulet *Output*.

# 11. Test af simuleringsprogram

I dette kapitel bliver de tests der er foretaget af simuleringsprogrammet præcenteret. Disse tests er foretaget af både programmets enkeltdele, som kontakt og friktions, samt for det samlede simuleringsprogram. De valgte tests er opdelt således at simuleringsprogrammets enkeltdele først bliver testet mht. dynamik, kontakt og friktion, hvorefter en test foretages af interaktionen mellem enkeltdelene, gennem en simulering af et afkastforløb. Herefter eftervises det om programmet kan simulere jam, hvilket udføres for samtlige friktionsmodeller. Afslutningsvis eftervises det om en sliske er selvstartende.

# 11.1. Verificering af dynamik, kontakt og friktion

Alle delfunktionerne af simuleringsprogrammet er testet separat, ved at udføre nogle udvalgte tests. Disse test er udvalgt således at de isolerer de dele af programmet det ønskes at teste. Opdelingen er foretaget for at resultatet bliver gennemskueligt og entydigt. Det er i dette afsnit valgt kun at beskrive resultaterne af testene, hvor den komplette gennemgang er givet i Appendiks X.

### Dynamik

Til test af simuleringsprogrammets dynamiske del er der udvalgt tre forskellige test. Formålet med disse tests er at vise at simuleringsprogrammets dynamiske enkeltdele fungerer korrekt. De tre tests er en gyroskopisk test, en test af bevægelse i det frie rum og en test af impulsbevarelsen.

Den gyroskopiske test er foretaget for at eftervise at de gyroskopiske kræfter bliver simuleret korrekt. Testen er foretaget ved at lade en kube stå på et hjørne og snurre, hvor kuben har en lille starthældning, der giver en ubalance. Resultatet er at kuben ikke vælter men prøver at rette op, hvilket kun er muligt hvis de gyroskopiske kræfter virker. Testen viser at de gyroskopiske kræfter virker og at systemets energiregnskab forbliver konstant gennem hele simuleringen

Test af bevægelse i det frie rum bliver udført ved at en pakke bliver frigivet i 16 meters højde, hvorefter det kontrolleres om den følger teorien, dvs.

$$s = \frac{1}{2} \cdot g \cdot t^2 \tag{11.1}$$

Hvor *s* er den afstand pakket skal være nået, til tiden *t*, med en gravitation *g* på 9.82  $m/s^2$ . Testen viser at pakken bevæger sig korrekt igennem det frie rum og at energiregnskabet forbliver konstant gennem hele simuleringen.

Impulsbevarelsen er en test der bliver foretaget uden friktion og dæmpning i systemet. Impulsbevarelsen bliver testet ved at den lille pakke på med en masse på 5 kg, se Figur 11.1, har en starthastighed i x-aksens retning på 1 m/s.



Figur 11.1: Impulsbevarelsestest, hvor udgangspunktet er vist til venstre og resultatet efter 2.5 sek. til højre.

Den lille pakke kolliderer med den store pakke, på ligeledes 5 kg, hvorved resultatet skal være at den store pakke opnår den lille pakkes initialhastighed på l m/s og at den lille pakke efterfølgende forbliver stationær. Forsøget viste at resultatet bliv som forventet, men at energien ikke er helt konstant som vist på Figur 11.2 (nederst).



Figur 11.2: ØV: Energiplot fra den lille kube. ØH: Energigrafen for den store kube. Nederst det samlede energiregnskab.

Denne afvigelse skyldes reduktionen i antallet af kontaktsøgninger, samt størrelsen af tidssteppet. Disse er afpasset således at energiregnskabet ved en simulering er konstant, med dæmper i kontaktpunktet. Ved denne test er der ingen dæmper i systemet, hvilket har gjort systemet numerisk ustabilt. Denne ustabilitet, se Figur 11.2 ØH, medfører at systemets energi stiger. Ustabiliteten er accepteret da systemet til enhver tid er stabilt når der er påført en dæmper.

### Kontakt

Det er valgt i simuleringsprogrammet at anvende tre forskellige kontakter dvs. Knude-Flade, Knude-Linje og Linje-Linje kontakt. Disse kontaktkombinationer er kontrolleret for at se om der opnås en stabil og realistisk kontakt, med konstant energi. Alle kontakter virker korrekt og resultatet er som forventet.

#### Friktion

Der er implementeret tre forskellige friktionsmodeller, der er kontrolleret ved Knude-Flade og Linje-Linje kontakt. Det er fravalgt at komme med eksempler på friktion under ren knude-linje kontakt, eftersom det ikke har været muligt at lave en entydig test med denne kontaktform. Disse test er foretaget på to forskellige slisker, én med en hældning på 11°, hvor pakken skal blive stående og én med en hældning på 31°, hvor pakken skal glide, se Figur 11.3. Der er foretaget 12 forskellige friktionsforsøg, der alle viser at friktionsmodellerne virker som forventet. Ud fra friktionstestene, på den stejle sliske, ses en forskel på, hvor langt de tre pakker kommer ned af slisken med de tre friktionsmodeller, se Figur 11.3.



Figur 11.3: Den samme test med de tre friktionsmodeller. TV: Dahl. IM: LuGre TH: Coulomb

Pakkerne med Dahl og LuGres friktionsmodel opnår ca. samme distance efter 0.6 sek. i simuleringerne, hvorimod pakken med Coulombs friktionsmodel afviger fra de to andre.

## 11.2. Simulering af afkastforløb

For at teste om de forskellige moduler dvs. dynamik, friktion og kontakt kan arbejde sammen er der foretaget en test af simuleringsprogrammet. Resultatet af simuleringen er efterfølgende vurderet mht. realisme, samt om det samlede energiregnskab stemmer. Seks billeder fra simuleringen, hvor der er anvendt Coulombs friktionsmodel, er illustreret på Figur 11.4.



Figur 11.4: Simulering af afkast med en pakke, hvor der er anvendt Coulombs friktionsmodel.

Energiregnskabet for simuleringen stemmer, og resultatet af simuleringens forløb er vurderet til at være realistisk. Ud fra simuleringen fremgår det at de enkelte moduler kan arbejde sammen. En mere uddybende gennemgang af simuleringerne kan findes i Appendiks X. En vurdering af simuleringernes realisme, er undersøgt i forbindelse med en verificering i kapitel 12.

## 11.3. Eftervisning af jam

Det er et ønske fra FKI at simuleringsprogrammet kan vise om et design giver jam. Denne eftervisning er foretaget ved en simulering, hvor startopstillingen er som vist på Figur 11.5



Figur 11.5: Startopstilling af pakker til eftervisning af jam.

Simuleringen foretages ved at de 31 kuber frigives, således at de alle forsøger at komme igennem slisken omtrent samtidigt, hvormed der opstår jam.

Der er foretaget tre ens simuleringer med hver af de tre friktionsmodeller. De tre simuleringer er foretaget for at kunne sammenligne om det har nogen indvirkning på jam, hvilken friktionsmodel der bliver anvendt. Resultatet af de tre simuleringer viser at der ikke kan opserveres nogen forskel på de tre friktionsmodellers opførsel, se Figur 11.6



Figur 11.6: Simuleringer af jam, hvor der efter 6,5 sek. har været stationaritet i 3.5 sek. TV: Dahl, IM: LuGre og TH: Coulomb

De tre figurer viser resultatet efter 6.5 *sek.*, hvor simuleringen har været stabil i ca. 3.5 *sek.* Det er vurderet at simuleringen til eftervisning af jam udviser en realistisk opførsel og at simuleringsprogrammet kan eftervise om der vil forekomme jam i en afkastsituation. En mere uddybende gennemgang af simuleringerne kan findes i Appendiks X.

## 11.4. Eftervisning af selvstart

Det er et ønske fra FKI at simuleringsprogrammet skal kunne eftervis et sliskedesigns evne til selvstart. Til at eftervise dette er der udført en simulering, hvor tre pakker er placeret stationært på en sliske, se Figur 11.7 ØV. Pladen foran slisken bliver fjernet efter 7 *sek.*, hvorefter pakkerne skal glide ud af slisken.



Figur 11.7: Fire billeder fra simulering, hvor tre pakker akkumuleret i trugsliske, bliver frigivet. Alle pakker glider herefter ud af slisken.

Simuleringen viser at de to første pakker hurtigt glider ud af slisken, hvorimod den tredje pakke er tæt på at blive hængende på det øverst flade stykke af slisken. På baggrund af simuleringen er det vist at simuleringsprogrammet kan eftervise en sliskes evne til selvstart.

## 11.5. Vurdering af simuleringsprogram

De udførte tests har vist at simuleringsprogrammet virker efter hensigten og at programmet kan eftervise de ønsker og krav der er givet i opgaveformuleringen. I forbindelse med test af friktionsmodeller udviser de tre modeller ikke den samme opførsel, hvor især Coulombs model afviger fra de to andre. En vurdering af simuleringernes realisme mht. de tre friktionsmodeller er givet i forbindelse med verificeringen i kapitel 12.
# 12. Verifikation af simuleringsprogram

Vha. simuleringsprogrammet er det muligt at simulere et afkastforløb af en vilkårlig pakke ned gennem en sliske. Det er dog ikke alene, på baggrund af simuleringen, muligt at vurdere om simuleringen udviser en korrekt opførsel, i forhold til et reelt afkastforløb. Derfor er det valgt at udføre en verificering af simuleringsprogrammet.

Verificeringen har til formål at vise simuleringsprogrammets pålidelighed, således at der opnås tiltro til simuleringerne i forbindelse med design af nye slisker. Ligeledes har verificeringen til formål at undersøge, hvilken af det tre implementeret friktionsmodeller der udviser den bedste opførsel mht. pakkernes afkastforløb.

Verificeringen udføres ved at sammenligne flere reelle afkastforløb med tilsvarende simuleringer. Ud fra de reelle afkast justeres simuleringsprogrammet ved at afstemme de forskellige materialeparametre.

Dette kapitel giver indledningsvis en beskrivelse af de forsøg der er udførte i forbindelse med verificeringen. Herefter er databehandlingen af de opnåede forsøgsresultater beskrevet. Afslutningsvis vurderes simuleringsprogrammet mht. de målte data for afkastforløbet. På baggrund af vurderingen laves en tilpasning af materialeparametrene til simuleringerne. Ud fra materialetilpasningen vurderes simuleringerne på ny.

## 12.1. Forsøgsbeskrivelse

Forsøget har til formål at kortlægge pakkernes bevægelse ned af slisken, for efterfølgende at have et verificeringsgrundlag til simuleringsprogrammet. Forsøgene er udføres på FKI Logistex testanlæg i Århus.

Til bestemmelse af kassernes position, er der i forbindelse med den pågældende sliske, placeret en måleopstilling.

## 12.1.1. Målemetoder

Dette afsnit har til formål at beskrive de målemetoder der kan anvendes til bestemmelse af pakkernes position og orientering.

Der findes et begrænset antal metoder til bestemmelse af positionen af pakken, eftersom:

- Pakken bevæger sig over en relativ stor afstand under et afkast.
- Måleudstyret i kassen må ikke influere på pakkens bevægelse.
- Midlerne til indkøb af udstyr er begrænsede.
- Målemetoden skal have en tilfredsstillende præcisions.

Der er på baggrund af ovenstående punkter fundet fire relevante målemetoder, som alle ligger inden for en overskuelige økonomiske og tidsmæssige rammer. I det følgende gives en kort beskrivelse af de fire målemetoder. En mere dybdegående beskrivelse, med skitsetegninger og vurderinger er givet i Appendiks XI.

#### Billedgenkendelse med highspeedkamera

Et highspeedkamera placeres, over slisken, således at billedplanet, se Figur 12.6, er parallel med slisken. Oven på pakkerne placeres et koordinatsystem, hvor koordinatakserne har forskellige farver. Vha. billedgenkendelse bestemmes positionen af pakkens koordinatsystems i forhold til en referencekoordinatsystem, i billedplanet. Metoden bestemmer kun pakkernes position i et plan og giver derfor ikke informationer om bevægelse og rotation ud af planet.

#### Stereobilledgenkendelse

Over slisken placeres to kameraer, således at begge kameraerne dækker hele slisken. Oven på pakkerne placeres et koordinatsystem, hvor koordinatakserne har forskellige farver. Ud fra billedgenkendelse bestemmes placeringen af pakkens koordinatsystems på begge kameras billedplaner. Vha. stereotriangulering bestemmes pakkens position og orientering. Præcisionen af stereobilledgenkendelse afhænger af kameraernes opløsning og kvalitet. En præcision på  $\pm 100 \ mm$  kan forventes ved en afstand på 3 m fra kameraerne (Aitenbichler, et al., 2003).

#### Accelerometer og gyrometer

Tre accelerometre og gyrometre, placeres inde i pakken, således at samtlige bevægelsesfrihedsgrader kan måles. Under afkastet samples accelerationer og vinkelhastigheder. De målte accelerationer og vinkelhastigheder differentieres hhv. en og to gange, for at bestemme position og orientering. Acceleromerene og gyrometrene giver, over tid, anledning til en eksponentiel stigende fejl pga. afrundings- og trugeringsfejl. Hastigheden hvormed fejlen accelerer afhænger af kvaliteten af udstyret.

#### Infrarød stereobilledgenkendelse

Infrarød stereobilledgenkendelse minder om ovenstående stereobilledgenkendelse. I stedet for at placerer et farvet koordinatsystem oven på pakkerne, monteres infrarøde dioder. Ligeledes placeres der foran kameralinserne infrarøde filtre, således at kun infrarød lys fra dioderne registreres. Ud fra billedgenkendelse bestemmes diodernes placering på begge kameraers billedplaner. Vha. stereotriangulering bestemmes diodernes position i forhold til hinanden, hvormed det er muligt at bestemme pakkens position og orientering. Præcisionen af IR stereobilledgenkendelse er ligeledes  $\pm 100 mm$ .

### 12.1.2. Valg af målemetode

Det er valgt at anvende infrarød stereobilledgenkendelse til bestemmelse af pakkernes position og orientering. Denne metode er valgt, da det forventes at kunne opnår en højere præcision på målingerne end ved brug af highspeedkameraet eller accelerometre. Highspeedkameraet giver kun mulighed for position og orientering i ét plan, hvormed information om rotation og bevægelser ud af planet går tabt. Stereobilledgenkendelsen giver mulighed for at bestemme position og orientering af pakken i rummet. Præcisionen af de målte data ændre sig ikke over den tid der samples, i modsætning til målinger med accelerometre, men er bestemt ud fra kameraernes kvalitet og kalibrering.

Den infrarøde stereobilledgenkendelse er valg frem for stereobilledgenkendelse, da billedgenkendelsen af lyset fra de infrarøde dioder er nemmere at bestemme, frem for at skal genkende et farvet koordinatsystem.

Der findes flere forskellige kommercielle produkter der anvender billedgenkendelse og infrarødt lys til positionsbestemmelse. Disse produkter er fravalgt primært pga. økonomiske årsager, men også pga. en faglig interesse fra gruppens side ifm. anvendelse af stereobilledgenkendelse.

## 12.2. Afgrænsning af forsøg

I forbindelse med udsorteringen har følgende punkter indflydelse på pakkens forløb ned gennem en sliske.

- Pakkens vægt
- Pakkens tyngdepunkt
- Pakkens størrelse og udformning
- Sliskedesignet
- Hastighed på sorteringsbåndet
- Placering af pakken på bakken

Det vil være mest hensigtsmæssigt at verificere simuleringsprogrammet ud fra forsøg, hvor samtlige ovenstående punkter varieres. Dette er dog ikke praktisk muligt eftersom det giver anledning til et uhensigtsmæssigt stort forsøgsarbejde og databehandling. Det er derfor valgt at begrænse verifikationen til 16 forskellige forsøg, hvor der anvendes,

- to forskellige slisker, hhv. trugsliske og vindelsliske
- to forskellige hastigheder på sorteringsbåndet, hhv. 2.5 m/s og 1.8 m/s.
- to forskellige tyngdepunktsplaceringer, hhv. hvor massen er jævnt fordelt, og placeret koncentreret, forskudt i forhold til pakkens centrum.

• to forskellige geometrier, hhv. en kvadratisk pakke og en aflang pakke.

Variationen af vægten på pakken er fravalgt på baggrund af antagelsen om at friktionsparametrene er uafhængige af normalkraften, som beskrevet i Appendiks V. Ligeledes er det fravalgt at variere pakkens placering på bakken. De 16 forsøg gentages seks gange for at have et tilstrækkeligt datagrundlag.

# 12.3. Forsøgsopstilling

Der anvendes et S3000E test sorteringsanlæg hos FKI Logistex. På den ene langside at sorteringsbåndet er vindelslisken placeret, mens trugsliske er placeret på modsatte side. Hen over sorteringsbåndet er der bygget et stillads, hvor de to kameraer er monteret ca. 2 meter over sorteringsbåndets bakker, således at deres billeder dækker hovedeparten af slisken, se Figur 12.1.



Figur 12.1: Forsøgsopstilling. Kameraerne er monteret ca. 2 meter over bakkerne og peger ned mod slisken.

For at undgå forstyrrelser i billedgenkendelsen er udefrakommende sollys reduceret ved at tildække ét af ovenlysvinduerne, samt at hænge presenninger op rundt om forsøgsopstillingen. I det følgende er en nærmere beskrivelse af det anvendte forsøgsudstyr givet.

### 12.3.1. Grabber udstyr

Til at dataopsamle filmene, anvendes et FALCON Quattro grabberkort (IDS), der kan grabbe film fra flere kameraer synkront. Der er anvendt det medfølgende software til dataopsamling.

## 12.3.2. Kamera

Der er anvendt to analoge s/h TVCCD-118 kameraer (Alpha Sound) med en opløsning på 512x384 pixel. Kameraerne er monteret i to skråtstillede plastik bøsninger, i en vinkel af 14 grader, således at deres sigtelinje krydser ca. to meter fra kameraerne, se Figur 12.2. Bøsningerne er monteret på en lægte 900 *mm* fra hinanden. Mellem kameralinsen og det yderste glas er der placeret et Kodak Wratten 80C high pass filter (Kodak), så kun infrarødt lys registreres på pilledeplanet.



Figur 12.2: Kameraerne er monteret i to bøsninger der er placeret 900 mm fra hinanden. Bøsningerne er skråtstillede så kameraerne peger i en 14° vinkel ind mod hinanden.

### 12.3.3. Diodeplader

Til hver pakke anvendes der tre 3 *mm* infrarøde dioder IR204-A (Ecolight.dk). Dioderne er monteret på en krydsfinerplade, så de tilsammen danner hjørnerne i en retvinklet trekant, hvor den ene katete er 150 *mm* og den anden 250 *mm*, se Figur 12.3. Den vinkelrette trekant udgør et lokalt koordinatsystem hvor origo er defineret som dioden i katetehjørnet. Den korte katete udgør  $\xi$ -aksen og den længste katete  $\eta$ -aksen. Dioderne er parallelt koblet til et 1.5 *v aaa* batteri.



Figur 12.3: Diodeplade med tre infrarøde dioder placeret således at de danner en retvinklet trekant.

## 12.3.4. Forsøgspakker

Det er valgt at anvende to forskellige flyttekasser som forsøgspakker. Flyttekasserne har målene 600x390x385 *mm* og 390x345x385 *mm*. Flyttekasserne valgt eftersom de har en flad bund, som svarer overens med kasserne anvendt til friktionsforsøgene. Til flyttekasserne er der konstrueret to forskellige ballasttyper i form af en ballast med en jævn fordelt masse, og en ballast med en koncentreret masse. De jævnt fordelte ballaster har en masse på hhv. 15.4 *kg* for den store pakke og 11.9 *kg* for den lille. De to ballaster med en koncentreret masse er konstrueres ved at indstøbe en stålklods i skummateriale, således at klodsen placeres i det ene hjørne af kassen. Den store og den lille ballast, med en koncentreret masse, vejer hhv.

13.5 kg og 11.3 kg. En detaljeret beskrivelse at forsøgspakkernes ballast, samt information om pakkernes masseinertimoment og tyngdepunkts placering er givet i Appendiks XII. Oven på hver ballast er der i centrum monteret en diodeplade, se Figur 12.4. En detaljeret beskrivelse at diodepladernes placering på forsøgspakkerne er givet i Appendiks XII.



Figur 12.4: Forsøgspakke i form af flyttekassen fyldt med en ballast. Oven på ballasten er diodepladen monteret.

## 12.4. Forsøgsudførsel

Forsøgene udføres ved at anvende samtlige fire forskellige pakkertyper, der afkastes ned gennem en af de to slisker, ved hver forsøgsudførsel. Under de fire afkast filmer de to kameraer, med en samplingsfrekvens på 12.5 *Hz*. Denne procedure udføres seks gange, hvorefter der skiftes til en anden hastighed på sorteringsbåndet. Når der er kørt tilsvarende seks forsøg med den nye hastighed, skiftes der til en anden sliske. De anvendte forsøgspakker placeres med samme rækkefølge med fire bakker mellem hver pakke, så de ikke fremstår samtidig på filmene. Ligeledes placeres pakkerne med samme orientering og placering på bakkerne ved hver forsøgsudførsel. For hver anden forsøgsudførsel udskiftes flyttekasserne med nye, således at slitagen af kasserne ikke bidrager til en variation af afkastforløbet.

## 12.5. Forsøgsresultater

De grabbede film fra de enkelte forsøgsudførsler, er efterfølgende redigeret ved at den enkelte pakkes afkastet er klippet ud. Samtlige film er vedlagt på DVD-2. Under forsøgsudførslen på vindelslisken, med en hastighed på 2.5 *m/s*, var det ikke muligt at få den lille pakke med den koncentreret masse til at stå stabilt på bakken. Pga. hastigheden rykkede pakken sig i det sidste sving inden afkastet, hvormed placeringen af pakken på bakken varierede, kraftigt, mellem de enkelte forsøg. Det er derfor valgt ikke at anvende afkastene, af den lille pakke, med den koncentrerede masse, ved en sorterhastighed på 2.5 *m/s*. Pga. problemerne med stabiliteten, er hastigheden på sorteringsbåndet, under forsøgene på trugslisken, reduceret fra 2.5 *m/s* til 2.2 *m/s*. Dette medførte ingen forbedringer i stabiliteten af den lille pakke. Det er derfor ligeledes valgt ikke at anvende afkastene af den lille pakke med en koncentreret masse ved disse forsøg.

## 12.6. Databehandling af film

Ved en forsøgsudførsel grabbes billederne fra de to kameraer i det tidsrum, hvor pakken falder ned af slisken. Ud fra de to grabbede film for hhv. det højre og venstre kamera udføres der en databehandling, til bestemmelse af pakkernes position og orientering. Det er valgt at databehandle de grabbede film i et program konstrueret i MATLAB. På Figur 12.5 er proceduren for databehandlingen af filmene illustreret.



Figur 12.5: Procedure for behandlingen af billederne fra de to kameraer til bestemmelse af diodernes tre koordinater.

De to film, optaget med hhv. højre og venstre kamera, indeholder en række billeder, som til et givet tidspunkt viser IR-diodernes position, i et forvrænget billedkoordinatsystem. For hver sampling fra det højre og venstre kamera udtages billederne, der efterfølgende behandles synkront. Først filtreres billederne for at fjerne støj og urenheder. Ud fra de filtrerede billeder bestemmes diodelysets centrumskoordinater, i det forvrængede billedkoordinatsystem. Centrumkoordinaterne for det højre og venstre kamera korrigeres derefter for den forvrængning, de to linser i kameraerne giver. Derefter bestemmes 3D-koordinaterne af de kombinerede centrumskoordinater, vha. en stereotriangulering. Ud fra de beregnede 3Dkoordinater bestemmes den kombination, mellem tre punkter, som svarer til placeringen af dioderne på diodepladen. Ud fra de fundne punkter bestemmes diodepladens position og orientering.

I det følgende er teorien anvendt til bestemmelse af pakkernes position og orientering beskrevet. Indledningsvis beskrives den anvendte kameramodel som ligger til grund for det anvendte visionsystem. Efterfølgende beskrives principperne bag den anvendte kalibreringstoolbox. Herefter beskrives proceduren i det konstruerede program.

#### 12.6.1. Interne kameraparametre

Visionsystemer tager udgangspunkt i at der er en geometrisk sammenhæng mellem et objekts position i rummet og det billede kameraet tager af objektet. For at bestemme denne sammenhæng tages der indledningsvis udgangspunkt i et kameras opbygning.

#### Kameraets opbygning

Når et kamera tager et billede af et objekt i rummet, sker det ved at reflekteret lys fra objektet, rammer et fotosensitivt plan inde i kameraet. For at opnå et billede i fokus, dvs. et skarpt billede, skal alle de reflekterede stråler fra ét punkt på objektet ramme ét tilsvarende punkt på billedplanet. Sker det ikke opnås et uskarpt billede, hvor punktet vil komme til at fremstå som en cirkel på billedplanet. For et opnå et billede i fokus findes der to muligheder (Trucco, et al., 1998 s. 18). Enten kan blænden i kameraet reduceres ned i størrelse, til et punkt, betegnet pinhole (knappenålshul). I dette tilfælde er blænden så lille at kun én lysstråle, fra hvert punkt på objektet, slipper ind på billedplanet. Dette giver en uforstyrret og skarp gengivelse af objektet på billedplanet (Trucco, et al., 1998 s. 18). En anden mulighed er at anvende et optisk system, i form af linser, som samler de udefrakommende stråler fra punktet ind på billedplanet. Pga. linsens udformning gengives objektet på billedplanet ikke eksakt, men forvrænges pga. linsens afbøjning af lysstrålerne.

Et kamera har brug for en vis mængede lys, for at det fotosensitive billedplan kan danne et billede. Mængden af lys der danner billedet er styret at blændetiden. Eftersom pinholeblænden kun giver en begrænset mængde lys, ind på billedplanet, er det nødvendigt med en lang blændetid for at opnå et godt billede. Dette er ikke hensigtsmæssigt da f.eks. et filmkamera skal tage flere billeder i sekundet. Et kamera med optik er derimod i stand til at modtage mere lys, alt efter hvilken linse der anvendes, og blændetiden kan derfor reduceres.

#### Pinhole modellen

Pinhole modellen antager at billedet, taget af kameraet, er uforstyrret, dvs. at linseforvrængningerne i billedet kan negligeres.

Modellen er baseret på princippet om kolinaritet, hvor hvert punkt i objektrummet er projiceret ved lige linjer ind på billedeplanet, samt at linjen gennem centeret af linsen ikke brydes af linsen (Heikkilä, et al.), se Figur 12.6.



*Figur 12.6: Punktet* **P** *i rummet projiceres ved lige linjer ind på billedplanet til punktet* **p**.

Billedet på billedplanet er spejlvendt og roteret 180 grader i forhold til omgivelserne, hvilket fremgår af punktet **P**'s projicion på Figur 12.6. Dette billede transformeres automatisk af kameraet, således at det vender korrekt. Transformationen bevirker at billedplanet kan betragtes som liggende foran linseplanet, se Figur 12.7.



Figur 12.7: Billedplanet placeres foran linseplanet.

I kameraet er der placeret to koordinatsystemer, se Figur 12.7. Et kamerakoordinatsystem med origo i linsens centrum, *z*-aksen ud af kameraets sigtelinje og *x*- og *y*-akserne parallelt med billedplanets ramme. Ligeledes er der indlagt et 2D koordinatsystem i billedplanet, med origo placeret i sigtelinjen. Dette koordinatsystem har koordinatakserne  $\tilde{u}$  og  $\tilde{v}$ , der ligeledes er parallel med billedplanets ramme. Foruden disse to koordinatsystemer, er der det billedets koordinatsystem, som har origo i øverste venstre hjørne af billedet, med akserne *u* og *v*. I det følgende betegnes  $\tilde{u}$  og  $\tilde{v}$  som billedkoordinatsystemet og *u* og *v* som det forvrængede billedkoordinatsystem.

Ved pinholemodellen projiceres punkterne  $\mathbf{p}_{k,i}(x_i, y_i, z_i)$ , givet i kamerakoordinatsystemet, ind på billedplanet vha. skaleringsforholdet mellem brændvidden *f* og den vinkelrette afstanden fra kamerakoordinatsystemets *xy*-plan og ud til punkterne, dvs. punkternes *z*koordinat. Sammenhængen mellem kamerakoordinatsystemet og billedplanets koordinatsystem er givet ved.

$$\begin{bmatrix} \tilde{u}_i \\ \tilde{v}_i \end{bmatrix} = \frac{f}{z_i} \begin{bmatrix} x_i \\ y_i \end{bmatrix}$$
(12.1)

Punkternes tilsvarende billedkoordinater  $(u_i, v_i)$ , bestemmes ud fra de projicerede koordinater  $(\tilde{u}_i, \tilde{v}_i)$  ved.

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} D_u \cdot s_u \cdot \tilde{u}_i \\ D_v \cdot \tilde{v}_i \end{bmatrix} + \begin{bmatrix} cc_1 \\ cc_2 \end{bmatrix}$$
(12.2)

Da enheden af  $(u_i, v_i)$  er givet i pixels, svarer koefficienterne  $D_u$  og  $D_v$  til forholdet mellem det metriske mål og pixels.  $S_u$  er en skaleringsfaktor (Heikkilä, 1997 s. 47). Koefficienterne  $cc_1$  og  $cc_2$  svarer til billedplanets centrum og adderes billedkoordinaterne, idet billedkoordinaterne i øverste venstre hjørne.

I forbindelse med den anvendte kalibreringstoolbox, er koefficienterne  $D_u$ ,  $D_v$ ,  $s_u$  og f reduceret til koefficienterne  $fc_1$  og  $fc_2$ , hvormed ligning (12.1) og (12.2) er givet ved.

$$\begin{bmatrix} \tilde{u}_i \\ \tilde{v}_i \end{bmatrix} = \frac{1}{z_i} \begin{bmatrix} x_i \\ y_i \end{bmatrix}$$
(12.3)

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} fc_1 \cdot \tilde{u}_i \\ fc_2 \cdot \tilde{v}_i \end{bmatrix} + \begin{bmatrix} cc_1 \\ cc_2 \end{bmatrix}$$
(12.4)

Pinholemodellen giver kun en approksimation af den reelle kameraprojicion, da modellen ikke tager højde for de forvrængninger kameralinsen giver. Modellen anvendes derfor ofte i sammenhæng, hvor der ikke er krav om en høj præcision, men derimod et ønske om en simpel matematisk sammenhæng. En mere præcis projicion af objektet ind på billedplanet, opnås ved at udvide pinhole modellen med en række korrektionsfaktorer.

#### Korrektion af linseforvrængninger

Den mest almindeligt anvendte korrektion, er korrektion af de radiale linseforvrængninger. De radielle forvrængninger medfører at de reelle billedpunkter bliver forskudt radialt i forhold til centrum af billedplanet. De radiale forvrængninger kan beskrives ved følgende ud-tryk (Heikkilä, 1997).

$$\begin{bmatrix} \delta u_i^{(r)} \\ \delta v_i^{(r)} \end{bmatrix} = \begin{bmatrix} \tilde{u}_i \left( kc_1 \cdot r_i^2 + kc_2 \cdot r_i^4 + kc_5 \cdot r_i^6 \right) \\ \tilde{v}_i \left( kc_1 \cdot r_i^2 + kc_2 \cdot r_i^4 + kc_5 \cdot r_i^4 \right) \end{bmatrix}$$
(12.5)

Hvor  $kc_1$ ,  $kc_2$  og  $kc_5$  er koefficienter for den radiale forvrængning og  $r^2 = \tilde{u}^2 + \tilde{v}^2$ .

Alt efter kameraets kvalitet er det normal tilstrækkeligt, at anvende det første eller de to første led i ligning (12.5) til at korrigere de radiale forvrængninger (Bouguet, 2007). Centeret af linsens krumning er ofte ikke eksakt centreret i et kamera. Det giver anledning til en decentreringsforvrængning som giver både en radial og tangentiel forstyrrelse. Den radielle forvrængning indgår i ligning (12.5), mens de tangentielle forvrængninger kan beskrives ved følgende udtryk (Heikkilä, 1997).

$$\begin{bmatrix} \delta u_i^{(t)} \\ \delta v_i^{(t)} \end{bmatrix} = \begin{bmatrix} 2kc_3 \cdot \tilde{u}_i \cdot \tilde{v}_i + kc_4 \left( r_i^2 + 2\tilde{u}_i^2 \right) \\ kc_3 \left( r_i^2 + 2\tilde{v}_i^2 \right) + 2kc_4 \cdot \tilde{u}_i \cdot \tilde{v}_i \end{bmatrix}$$
(12.6)

Hvor  $kc_3$  og  $kc_4$  er koefficienterne for de tangentielle forvrængninger. Eftersom kameralinserne ofte er temmelig præcist placeret, i forhold til billedplanet, er det ofte ikke nødvendigt at tage de tangentielle korrektionsfaktorer med i den samlede korrektion (Heikkilä, et al.). På Figur 12.8 er konsekvenserne af hhv. radiale og tangentielle forvrængninger illustreret, hvor de stiplede linjer svarer til forvrængningerne af de fyldte linjer.



Figur 12.8: TV: Effekten af radial billedforvrængning. TH: Effekten af tangentiel billedforvrængning. De solide linjer er ikke forvrængede, mens de stiplede linjer svarer til en positiv forvrængning og de prikkede linjer til en negativ forvrængning.

Den radiale og tangentielle forvrængning adderes til de projicerede koordinater i ligning (12.3).

$$\begin{bmatrix} \tilde{u}_i^{(s)} \\ \tilde{v}_i^{(s)} \end{bmatrix} = \begin{bmatrix} \tilde{u}_i + \delta u_i^{(r)} + \delta u_i^{(t)} \\ \tilde{v}_i + \delta v_i^{(r)} + \delta v_i^{(t)} \end{bmatrix}$$
(12.7)

De systematisk forvrængede billedkoordinater bestemmes ud fra ligning (12.4) ved.

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} fc_1 \left( \tilde{u}_i^{(s)} + \alpha_c \cdot \tilde{v}_i^{(s)} \right) \\ fc_2 \cdot \tilde{v}_i^{(s)} \end{bmatrix} + \begin{bmatrix} cc_1 \\ cc_2 \end{bmatrix}$$
(12.8)

Koefficienten  $\alpha_c$  beskriver vinklen mellem *u*- og *v*-akserne, hvis kameraets pixels ikke er kvadratiske (Bouguet, 2007). Parametrene  $kc_{1-5}$ ,  $fc_{1-2}$ ,  $cc_{1-2}$  og  $\alpha$ , betegnes de interne kameraparametre.

#### 12.6.2. Eksterne kameraparametre

De eksterne kameraparametre beskriver sammenhængen mellem et globalt referencekoordinatsystem og kamerakoordinatsystemet. De eksterne parametre er givet ved en flytningsvektor  $\mathbf{t}_{g}$  og transformationsmatricen  $\mathbf{R}_{g}$ , hvor et punkt  $\mathbf{p}_{g}$  med koordinaterne (X, Y, Z) i det globale koordinatsystem er givet i kamerakoordinatsystemet ved.

$$\mathbf{p}_k = \mathbf{R}_g \cdot \mathbf{p}_g + \mathbf{t}_g \tag{12.9}$$

Hvor punktet  $\mathbf{p}_k$  har koordinaterne (x, y, z). Transformationsmatricen  $\mathbf{R}_g$  er givet i Euler vinkler, hvormed der i alt er seks eksterne kameraparametre. For et visionsystem, hvor der indgår flere kameraer, indeholder de eksterne parametre også relationen mellem kameraernes koordinatsystemer. Punktet  $\mathbf{p}_H$  i det højre kamerakoordinatsystem er givet i det venstre kamerakoordinatsystem ved.

$$\mathbf{p}_V = \mathbf{R}_k \mathbf{p}_H + \mathbf{t}_k \tag{12.10}$$

Hvor  $\mathbf{t}_k$  er flytningsvektoren mellem kamerakoordinatsystemerne og  $\mathbf{R}_k$  er transformationsmatricen.

### 12.6.3. Kalibrering

Både de eksterne og interne kameraparametre bestemmes individuelt, for hvert kamera, ved en kalibrering. Efterfølgende anvendes kalibreringsdataene for de to kameraer til en samlet stereokalibrering. Til kalibreringen er det valgt at anvende en kamerakalibreringstoolbox til MATAB lavet af Jean-Yves Bouguet (Bouguet, 2007). I det følgende gives en kort beskrivelse af teorien som ligger til grund for den anvendte toolbox.

#### Kalibrering af kamera

Kalibreringen udføres ved først at grabbe en række billeder af et skakbræt, med det pågældende kamera. Skakbrættet placeres i forskellige vinkler og afstande i forhold til kameraet, så hele kameraets synsfelt dækkes, se Figur 12.9.



Figur 12.9: TV: Kalibreringsbilleder af skakbræt. TH: Koordinatpunkterne på skakbrættet bestemmes mht. det forvrængede billedkoordinatsystem. I øverst venstre hjørne af skakbrættet placeres det globale koordinatsystem.

I hvert billede defineres et globalt koordinatsystem, f.eks. i øverste venstre hjørne af skakbrættet, se Figur 12.9. Da størrelsen af skakbrættets tern er foruddefineret, er alle hjørnepunkernes koordinater i skakbrættet kendt med hensyn til det indlagte globale koordinatsystem. Det giver punkterne  $\mathbf{p}_i(x_i, y_i, z_i)$  givet i det globale koordinatsystem, som har tilsvarende koordinater  $(u_i, v_i)$  i det forvrængede billedkoordinatsystem, hvor i = 1, ..., N er antallet af punkter. Pga. kameraparametrenes ikke-lineære sammenhæng i ligning (12.8), er det ikke muligt at bestemme disse på lukket form. I stedes estimers både de interne og eksterne parametre ved en optimering som minimerer residualet mellem kameramodellen og de Nobservationer. Residualet straffes med kvadratet, og er givet ved følgende kostfunktion.

$$F = \sum_{i=1}^{N} (U_i - u_i)^2 + \sum_{i=1}^{N} (V_i - v_i)^2$$
(12.11)

Hvor  $U_i$  og  $V_i$  er de, ud fra punkterne  $\mathbf{p}_i$ , beregnede koordinater i det forvrængede billedkoordinatsystem. Inden optimeringen udføres der en initialisering. Startgættet bestemmes ved at løse pinhole modellen på lukkede form, hvor linseforvrængningsparametrene ikke inkluderes. Optimeringen udføres med en iterativ gradient metode, hvor Jacobimatricen er givet ved en eksplicit formulering (Bouguet, 2007). De interne parametre giver ni frihedsgrader i form af fokuslængden, billedplanets centrumskoordinater og forvrængningskoefficienterne, mens de eksterne parametre giver seks frihedsgrader, for hvert koordinatsystem i kalibreringsbillederne. For at opnå et tilfredsstillende resultat, af optimeringen, er det nødvendigt med mindst to kalibreringsbilleder i forskellige positioner og orienteringer.

#### Stereokalibrering

Stereokalibreringen har til formål at bestemme relationen mellem de to kameraer, det vil sige vektoren  $\mathbf{t}_k$  og transformationsmatricen  $\mathbf{R}_k$ . Resultatfilerne fra de to individuelle kalibreringer af højre og venstre kamera anvendes, hvor de anvendte kalibreringsbilleder for det højre og venstre kameraer skal passe sammen parvis. På baggrund af data fra de to individuelle kalibreringer udføres der en tilsvarende optimering for at bestemme  $\mathbf{t}_k$  og  $\mathbf{R}_k$ .

#### Kalibreringsresultater

Eftersom de to kameraer aldrig flytter sig i forhold til hinanden, har det under hele forsøgsudførslen kun været nødvendigt at udføre én kalibrering af kameraerne. Til kalibreringen er der anvendt et skakbræt med 176 tern med en størrelse på 50x50 *mm*. Det anvendte skakbræt er filmet synkront med de to kameraer. Under optagelserne er positionen af brættet flyttet, således at kameraernes synsfelt og dybde er dækket. Af de to film er der udtaget 25 billeder fra hvert kamera. De 25 billeder er udtage så skakbrættet fremstå tydeligt på begge billeder og variere i positionen. De anvendte kalibreringsbilleder er vedlagt på DVD-1 i mappen *Kamera Kalibrering*.

Under kalibreringen er fire af kalibreringsbillederne frasorteret, da det gav anledning til er bedre resultat. Resultatet fra de udførte kalibreringer, for det højre og venstre kameraer, er givet i Appendiks XIII. På Figur 12.10 er placeringen af de 21 skakbrætter illustreret i forhold til det højre og venstre kamera.



Figur 12.10: Placering af de 21 skakbrætter fra stereokalibreringen.

#### Globalt koordinatsystem

Det er valgt at bestemme det globale referencekoordinatsystems placering ud fra en stereotriangulering af tre referencepunkter mht. det venstre kamerakoordinatsystem. Ud fra de tre punkter defineres det globale koordinatsystems x- og y-akse.

I forbindelse med de udførte forsøg er der anvendt et globalt koordinatsystem til hhv. trugslisken og vindelslisken. Det er valgt at ligge de globale koordinatsystemer i sliskernes lokale koordinatsystem defineret i CAD tegningerne. Punkterne anvendt til de to globale koordinatsystemer er vist på Figur 12.11.



Figur 12.11: Placering af det globale koordinatsystem på vindel- og trugslisken, samt de punkter hvor diodepladen er placeret.

Punkterne er fundet ved at filme diodepladen i de tre punkter. Filmene er efterfølgende billebehandlet for at finde diodepladens position og orientering. Ud fra de beregnede punkters positioner er det globale koordinatsystem bestemt. På DVD-1 i mappen *Globale koordinatsystemer* er filmene og billedbehandlingsprogrammet, til bestemmelse af de globale koordinatsystemer vedlagt. Beregningerne af stedvektoren og transformationsmatricen, til de globale koordinatsystemer, for trugslisken og vindelslisken, er givet i Appendiks XIII.

## 12.6.4. Billedbehandlingsprogram

Til bestemmelse af pakkernes position og orientering er der konstrueret et billedbehandlingsprogram. Programmet er bygget op omkring én løkke der behandler et billede fra hvert kamera af gangen. I det følgende gives en kronologisk beskrivelse af behandlingen af de to billeder.

#### Filtrering af billede

For at bestemmer diodepladens position i 3D-koordinater, er det nødvendigt at identificere diodelysenes koordinater, i det forvrængede billedkoordinatsystem.

De to billeder, et fra det højre og et fra venstre kamera, importeres til programmet som et gråskalabillede. Det infrarøde filtrer foran kameralinserne medfører at synligt lys filtreres

fra, så kun lys med en bølgelængde over 800 *nm* registreres. Et eksempel på et billede af diodepladen fra det højre og det venstre kamera er illustreret på Figur 12.12.

Figur 12.12: Gråskala billeder med infrarødt filter fra det venstre og højre kamera.

Som det fremgår af Figur 12.12, fremstår infrarødt lys fra andre lyskilder end IR-diodene. Gråskalabillederne konverteres efterfølgende til et binært billede, hvor pixels med en gråtoneværdi over 230, får værdien ét og alt under får værdien nul. Konverteringen af billederne fra Figur 12.12 til binær billede er vist på Figur 12.13.



Figur 12.13: Konverterede billeder fra gråskala til binært billede for venstre og højre kamera.

Af Figur 12.13 fremgår diodelyset som to hvide arealer, sammen med en række andre arealer i form af støj fra det udefrakommende lys. Da støjarealerne giver anledning til flere beregninger i forbindelse med stereotrianguleringen, filtreres billederne yderligere. Fra MAT-LAB *Image Processing Toolbox* behandles billederne med funktionerne *imfill, bwareaopen* og *ismember*. Funktion *imfill* fjerne sorte pixels som befinder sig inde i et hvidt areal. *bwareaopen* fjerner alle hvide arealer som er større end en specificeret størrelse og *ismember* fjerner alle hvide arealer som er mindre end en specificeret størrelse. På Figur 12.14 er store og små arealer fra ovenstående billeder filtreret fra.



Figur 12.14: Arealer større og mindre end arealerne fra diodelyset er sorteret fra.

De arealer som har samme størrelse som diodearealerne kan ikke frasorteres og indgår derfor i de videre beregninger. Ud fra de to filtrerede billeder bestemmes koordinaterne for de hvide arealers centrum, mht. det forvrængede billedkoordinatsystem. Centrumkoordinaterne bestemmes ved at tildele hvert areal en label vha. funktionen *bwlabln*. Herefter bestemmes centrum for hvert areal, ved at anvend funktionen *regionprops*.

#### Billedekorrektion

Inden der udføres en stereotrianguleringen af de fundne centrumskoordinater, transformeres disse fra det forvrængede billedekoordinatsystem til billedplanets koordinatsystem. Den anvendte kameramodel beskrevet i afsnit 12.6.1, beskriver projicionen af et 3D punkt ind på billedplanet mht. det forvrængede billedkoordinatsystem. Ud fra udtrykkene for den radielle og tangentielle forvrængning i ligning (12.7) fremgår det at der ikke findes nogen direkte analytisk løsning til det inverse problem. Anvendes f.eks. samtlige seks korrektionsfaktorer bliver ligning (12.8), hvis den udskrives, et 7. grads polynomium.

$$\begin{bmatrix} u_{i} \\ v_{i} \end{bmatrix} = \begin{bmatrix} fc_{1}\left(\tilde{u}_{i}+\tilde{u}_{i}kc_{1}\cdot r_{i}^{2}+\tilde{u}_{i}kc_{2}\cdot r_{i}^{4}+\tilde{u}_{i}kc_{5}\cdot r_{i}^{6}+2kc_{3}\cdot \tilde{u}_{i}\cdot \tilde{v}_{i}+kc_{4}\left(r_{i}^{2}+2\tilde{u}_{i}^{2}\right)+\alpha_{c}\cdot \tilde{v}_{i}^{(s)}\right)+cc_{1} \\ fc_{2}\cdot\left(\tilde{v}_{i}+\tilde{v}_{i}\left(kc_{1}\cdot r_{i}^{2}+kc_{2}\cdot r_{i}^{4}+kc_{5}\cdot r_{i}^{4}\right)+kc_{3}\left(r_{i}^{2}+2\tilde{v}_{i}^{2}\right)+2kc_{4}\cdot \tilde{u}_{i}\cdot \tilde{v}_{i}\right)+cc_{2} \\ (12.12)$$

For at bestemme  $\tilde{u}_i$  og  $\tilde{v}_i$  ud fra  $u_i$  og  $v_i$ , anvendes den numerisk ligningsløser *Fixed-Point Iterations metode*, givet ved (Kreyszig, 1999 s. 838).

$$x_{n+1} = g(x_n), (n = 0, 1, \cdots)$$
 (12.13)

Eftersom de forvrængede billedkoordinater  $u_i$  og  $v_i$  ligger tæt på billedkoordinaterne  $\tilde{u}_i$  og  $\tilde{v}_i$  anvendes disse som startgæt. Ligning (12.13) kører 20 iterationer, hvorefter det antages at resultatet er konvergeret (Bouguet, 2007). De transformerede koordinater indgår efterfølgende i stereotrianguleringen.

#### Stereotriangulation

Stereotriangulering er evnen til, ud fra trigonometriske sammenhænge, at bestemme et punkts position i rummet vha. to eller flere billeder. Det er valgt at anvende trianguleringsmetoden givet af (Jean-Yves, 1998).

Punktet  $\mathbf{p}_{\mathbf{g}}$ , se Figur 12.15, har koordinaterne  $\mathbf{p}_{k,H}(x_H, y_H, z_H)$  og  $\mathbf{p}_{k,V}(x_V, y_V, z_V)$  i hhv. det højre og venstre kamerakoordinatsystem. Sammenhængen mellem de to koordinater er givet ved.

$$\mathbf{p}_{k,V} = \mathbf{R}_k \mathbf{p}_{k,H} + \mathbf{t}_k \tag{12.14}$$

Ud fra ligning (12.3) er de projicerede punkter ind på billedplanet givet ved.

$$\mathbf{p}_{b,H} = \frac{\mathbf{p}_{k,H}}{z_H} = \left[ u_H \, v_H \, 1 \right]^T \,, \ \mathbf{p}_{b,V} = \frac{\mathbf{p}_{k,V}}{z_V} = \left[ u_V \, v_V \, 1 \right]^T \tag{12.15}$$



Figur 12.15: Ved stereotriangulering bestemmes punktet  $\mathbf{p}_{gs}$  position i rummet ud fra punktets placering på de to billedplaner, samt relation mellem de to kamerakoordinatsystemer.

Det ønskes at bestemme  $\mathbf{p}_{k,H}$  og  $\mathbf{p}_{k,V}$  ud fra  $\mathbf{p}_{b,H}$  og  $\mathbf{p}_{b,V}$ . Ud fra Ligning (12.14) og (12.15) er følgende sammenhæng givet.

$$z_V \mathbf{p}_{b,V} = z_H \mathbf{R}_k \mathbf{p}_{b,H} + \mathbf{t}_k \tag{12.16}$$

Hvormed følgende relation er givet.

$$\begin{bmatrix} -\mathbf{R}_{k}\mathbf{p}_{b,H} \ \mathbf{p}_{b,V} \end{bmatrix} \begin{bmatrix} z_{H} \\ z_{V} \end{bmatrix} = \mathbf{t}_{k}$$
(12.17)

Dette giver tre ligninger med to ubekendte, som løses med mindste kvadratiske metode (Wikipedia), hvor  $z_H$  og  $z_V$  bestemmes ved.

$$\begin{bmatrix} z_H \\ z_V \end{bmatrix} = \left( \mathbf{A}^T \mathbf{A} \right)^{-1} \mathbf{A}^T \mathbf{t}$$
(12.18)

Hvor  $\mathbf{A} = \begin{bmatrix} -\mathbf{R}\mathbf{p}_{b,H} & \mathbf{p}_{b,V} \end{bmatrix}$ . Ud fra  $z_H$  og  $z_V$  bestemmes x- og y-koordinaterne for det højre og venstre kamerakoordinatsystem ved ligning (12.15).

Eftersom programmet ikke ved, hvilke centrumskoordinater der hører sammen, mellem det venstre og det højre billede, trianguleres punkterne i alle kombinationsmuligheder. Det giver  $n_v \cdot n_h = n_{vh}$  kombinationsmulighed, hvor  $n_v$  er antallet af punkter på det venstre billede og  $n_h$  er antallet af punkter på det højre.

#### Identifikation af diodepunkter

Ud fra de triangulerede 3D punkter identificerer programmet, hvilke tre punkter der svarer til diodepladens placering. De tre punkter bestemmes ved at finde den kombination af punkter, hvis indbyrdes afstand og vinkel svarer bedst overens med den reelle afstand og vinkel på diodepladen. De  $n_{vh}$  triangulerede punkter kombineres i sæt af tre, hvilket giver følgende antal kombinationsmuligheder.

$$n_{t} = \frac{n_{vh}!}{3! (n_{vh} - 3)!}$$
(12.19)

Da et punkt fra det venstre eller højre billede ikke må indgå to eller tre gange i et sæt, sorteres disse fra. For hvert sæt bestemmes den indbyrdes afstand mellem punkterne. Mellem punkterne med de to korteste afstande, dvs. kateterne, bestemmes vinklen,  $\theta$ . Ud fra vinklen og de to korteste afstande vægtes, hvert sæt med følgende vægtningsfunktion.

$$P_{V_{acgt}} = abs (250mm - L_{M}) + abs (150mm - L_{K}) + abs (90^{\circ} - \theta)^{2}$$
(12.20)

Hvor  $L_k$  og  $L_m$  er hhv. den korteste og den næst korteste afstand mellem punkterne. Den kombination af punkter som giver den mindste vægtning, svarer til diodernes placering. Hvis den bedste vægtning, der findes i den pågældende sampling, er over 100 vurderes det at lyset fra én eller flere dioder ikke kan observeres på ét af billederne og diodepladen kan ikke identificeres.

#### Bestemmelse af position og orientering af diodepladen

Når de tre diodepunkter er identificeret, bestemmes positionen og orienteringen af diodepladen mht. det globale koordinatsystem. Punkterne er efter stereotrianguleringen givet i det venstre kamerakoordinatsystem, hvilket transformeres over til det globale koordinatsystem vha. ligning (12.9). Da diodepladens origo er placeret i det vinkelrette hjørne af diodepladen, som beskrevet i afsnit 12.6.3, svarer punktet med den korteste afstande til de to andre punkter, til pladens origo. Orienteringen af dioderne bestemmes ud fra vektorerne **a** og **b**, hvor vektoren **a** går fra origo til det nærmeste punkt og vektoren **b** fra origo til det resterende punkt. Vektoren **c** er vinkelret til **a** og **b**, og er givet ved.

$$\mathbf{c} = \mathbf{a} \times \mathbf{b} \tag{12.21}$$

Eftersom vektorerne  $\mathbf{a}$  og  $\mathbf{b}$  ikke nødvendigvis er vinkelrette krydses  $\mathbf{c}$  og  $\mathbf{a}$  til vektoren  $\mathbf{b}'$ .

$$\mathbf{b}' = \mathbf{c} \times \mathbf{a} \tag{12.22}$$

Vektoren **a** er parallel med diodepladens  $\xi$ -akse og **b**' parallel med  $\eta$ -aksen. Ud fra enhedsvektorene af **a**, **b**' og **c** er transformationsmatricen mellem det globale koordinatsystem og diodepladens lokale koordinatsystem givet ved.

$$\mathbf{A} = \begin{bmatrix} u_{\mathbf{a},\xi} & u_{\mathbf{b}',\xi} & u_{\mathbf{c},\xi} \\ u_{\mathbf{a},\eta} & u_{\mathbf{b}',\eta} & u_{\mathbf{c},\eta} \\ u_{\mathbf{a},\zeta} & u_{\mathbf{b}',\zeta} & u_{\mathbf{c},\zeta} \end{bmatrix}$$
(12.23)

Ud fra transformationsmatricen bestemmes Euler parametrene ved (Nikravesh, 1988 s. 160), hvor  $e_0$  bestemmes ved.

$$e_0 = \frac{\left(a_{11} + a_{22} + a_{33}\right) + 1}{4} \tag{12.24}$$

Og  $e_1$ ,  $e_2$  og  $e_3$  bestemmes ved.

$$e_1 = \frac{a_{32} - a_{23}}{4 \cdot e_0}$$
,  $e_2 = \frac{a_{13} - a_{31}}{4 \cdot e_0}$ ,  $e_3 = \frac{a_{21} - a_{12}}{4 \cdot e_0}$  (12.25)

I de billeder, hvor diodepladen ikke kan identificeres sættes positionen og orientering for den pågældende sampling til nul.

#### 12.6.5. Hastighed og accelerationsbestemmelse

Hastigheden og accelerationerne bestemmes vha. numerisk differentiation, givet ved (Burden, et al., 2001).

$$f'(x_1) \approx \frac{1}{h} \left[ -\frac{1}{2} f(x_0) + \frac{1}{2} f(x_2) \right]$$
(12.26)

Det har ved dynamiske forsøg, beskrevet i afsnit 12.7, vist sig at de beregnede hastigheder og accelerationer variere meget i forhold til de reelle hastigheder og accelerationer. Dette skyldes primært at den numerisk differentiation er meget følsomhed over for små unøjagtigheder i de anvendte funktionsværdierne (Kreyszig, 1999 s. 879). På baggrund af de udførte forsøg er det valgt ikke at anvende de beregnede hastigheder i den videre databehandling.

### 12.7. Test af målemetode

Der er udført to test for at bestemme præcisionen af det anvendte målesystem. Den ene test har til formål statisk at bestemme, hvor præcis visionsystemet beregner diodepladens position. Det andet et dynamisk forsøg, hvor positionen bliver bestemt, mens diodepladen er i bevægelse.

Det statiske forsøg er udført ved at flytte diodepladen i en lige linje mellem en række veldefinerede punkter med en indbyrdes afstand på 200 *mm*. De målte data er efterfølgende behandlet for at bestemme afstandsfejlen mellem målepunkterne. Af det statiske forsøg er den største afvigelse fundet til 3.3 *mm*. I denne afvigelse skal der, ud over kameraernes og databehandlingens usikkerheder, medregnes usikkerheder på placeringen af målevognen og diodernes placering på diodepladen. Det dynamiske forsøg er udført ved at bevæge diodepladen i en lige linje med en hastighed på hhv. 1.4 m/s og 2.8 m/s, forbi kameraerne. De udførte forsøg udviser en korrekt tendens, idet målepunkterne er jævnt fordelt langs en lige linje. De beregnede hastigheder afviger op til 0.3 m/s fra de reelle hastigheder.

En mere dybdegående gennemgang samt vurdering af de udførte forsøg findes i Appendiks XIV.

### Interlacing

I forbindelse med de dynamiske forsøg har det vist sig at dioderne, på de optagede film, får en skyggeeffekt, se Figur 12.16. Effekten, interlacing, skyldes signalbehandlingen fra kameraet til grabberkortet, hvor dataene sendes som en streng, Dvs. én linje af gangen fra toppen til bunden. For at øge den visuelle opdateringshastighed, sendes skiftevis de ulige og lige rækker til grabberkortet (Wikipedia). Det medfører at de lige rækker, bliver forskudt i billedet i forhold til de ulige rækker.

Skyggeeffekten medfører at arealet af IR-dioderne bliver aflangt. Dette er ikke hensigtsmæssigt da dette giver anledning til upræcisioner i målingerne. Det er derfor valgt at fjerne alle de lige linjer fra billedet. Dette giver et billede, hvor halvdelen af linjerne er sorte. For at opnå et sammenhængende areal, af diodelyset, gennemløber en stencil hele billedet. Ud fra det oprindelige billede laver stensilen alle omkringliggende pixels hvide, hvis den pågældende pixel er hvid. Dette giver et sammenhængende areal, hvorudfra centrumkoordinaterne kan bestemmes.



Figur 12.16: TV: Når dioderne har en hastighed, kommer der en skyggeeffekt på billedet. TH: For at undgå denne effekt, erstattes alle lige linjer med sort.

## 12.8. Resultatbehandling

Filmene fra de udførte forsøg er databehandlet vha. det beskrevne billedbehandlingsprogram. Det har været nødvendigt i forbindelse med databehandlingen, at specificere i hvilket område diodepladen er placeret i billedet, da mængden af støj fra sollys har været så omfattende at programmet ikke har kunnet håndtere antallet af centrumskoordinater. Ligeledes har mængden af støj medført at grænseværdien, anvendt under konvertering til binært billede, er reguleret mellem hver måleserie, for bedre at kunne identificer diodelyset.

I forbindelse med en sammenligning mellem måledata og simulering, er det valgt at anvende pakkernes centrum som origo. Derfor er samtlige målepunkter flyttet mht. flytningsvektoren fra diodepladen, ned til pakkens centrum. De anvendte flytningsvektorer er givet i Appendiks XII.

De beregnede måleserier for den enkelte pakke, ved en given hastighed og sliske, er sammenlignet. De måleserier som afviger unaturligt fra de andre serier, er forkastet, og indgår derfor ikke i den videre behandling.

På Figur 12.17 er de opnåede måleserier, ved forsøg på trugslisken, med en sorterhastighed på 1.8 *m/s*, for den store pakke, med en jævn masse, illustreret. Orienteringen af pakken i det enkelte målepunkt er illustreret ved, pakkens lokale koordinatsystem, hvor den røde linje svare til  $\xi$ -aksen, den grønne linje til  $\eta$ -aksen og den blå linje  $\zeta$ -aksen.



Figur 12.17: Måledata for den store pakke med en jævn masse, ved forsøg på trugslisken med en sorterhastighed på 1.8 m/s.

I Appendiks XV er graferne for de resterende forsøg illustreret.

## 12.8.1. Vurdering af måleusikkerheder

De enkelte målepunkter er forbundet med en fejl, hvilket skyldes usikkerhed i det anvendte målesystem og forsøgsudførslen. Usikkerhederne er inddelt i systematiske fejl, der er konstante gennem hele forsøgsudførslen og tilfældige fejl der variere gennem forsøgsudførselen. Følgende usikkerheder er forbundet med det udførte forsøg. Systematiske fejl

- Kalibrering
- Placering af dioder på diodeplade
- Placering af diodeplade på forsøgspakker

- Hastighed af sorterbåndet
- Afkasttidspunktet

Tilfældige fejl

- Interlacing
- Støj
- Placering af pakke på bakke
- Slidtage af papkasser

For de systematiske fejl er det især kalibreringen af de to kameraer der har en stor betydning for præcisionen af målingerne. Der er ved kalibreringen af de to kameraer opnået en usikkerhed på (0.17,0.17) og (0.17,0.15) pixels i det forvrængede billedkoordinatsystem for det venstre og højere kamera, se Appendiks XIII. Hvor usikkerheden er givet tre gange standartafvigelsen.

De primære fejlkilder, forbundet med de tilfældige fejl, er interlacing og støj.

Pga. de skyggeeffekter der opstår når dioderne bevæger sig, har det været nødvendigt at fjerne alle de lige linjer fra billedet. Dermed er en del af informationen omkring diodelysets udformning, gået tabt.

Støj fra udefrakommende lys har ligeledes givet anledning til fejl i måledataene. Sollyset har medført at dioderne i nogle billeder ikke har kunnet identificeres, hvormed det pågældende målepunkt er gået tabt. Ligeledes har den store mængde sollys medført at lyset fra dioderne ikke har fremstået tydeligt, da det anvendte kamera er forsynet med *automatic gain controle*, som automatisk regulere lysstyrken. Dette har medført at diodelyset generelt fremstår som en mindre prik på billedet, hvilket giver anledning til en mindre præcision i bestemmelsen af centrumskoordinatet.

Spredningen mellem de opnåede målepunkter er forskellig alt efter hvilken retning de betragtes fra, se Figur 12.17. Betragtes punkterne i det globale xy-plan varierer de værste måleserier ca.  $\pm 150 \ mm$  fra hinanden, hvorimod de varierer op imod  $\pm 500 \ mm$  i z-retningen. Den store variation i z-retningen skyldes trianguleringens følsomhed over for upræcisioner i centrumskoordinaterne.

Eftersom pakkerne er i kontakt med bakken og slisken gennem hele afkastforløbet, accepteres den dårlige præcision af måleresultaterne i z-retningen, da pakkens bevægelse i denne retning er bundet af sliskens overflade. Derudover fremgår det af Figur 12.17 at bevægelsen og orientering i xy-planet giver de primære informationer omkring afkastforløbet.

## 12.9. Verifikationssimuleringer

De beregnede målepunkter har til formål at verificere simuleringsprogrammet, og vil i det følgende indgå i en sammenligning med simuleringer af afkast, der svarer til de udførte forsøg. Ud fra den anvendte forsøgsopstilling og de udførte forsøg, er der konstrueret fire simuleringer. Der er gennem FKI Logistex skaffet CAD modeller af de anvendte slisker og bakke. Under forsøgsudførslen er sliskernes position ift. sorteranlægget, samt afkasttidspunktet, bestemt. På baggrund af disse data er de fire simuleringer konstrueret for hhv. vindelslisken og trugslisken, med de to forskellige hastigheder, se Figur 12.18.



Figur 12.18: Verifikationssimuleringer. TV: Trugslisken. TH: Vindelslisken.

De anvendte pakker i simuleringen er placeret og orienteret på bakken, som ved de udførte forsøg. Pakkerne har samme masse og masseinertimoment som forsøgspakkerne, se Appendiks XII. Simuleringerne udføres med samtlige tre friktionsmodeller, hvor der til friktionsmodellerne anvendes friktionsparametrene bestemt i afsnit 8.2. I Appendiks XVI er de anvendte data til verifikationssimuleringerne nærmere beskrevet. På vedlagte DVD-2 i mappe *Film* er film af de fire verificeringssimuleringer vedlagt.

## 12.10. Kurvetilpasning

Eftersom der kan være en stationære fejl mellem måledataene og verifikationssimuleringerne er det valgt at lave en kurvetilpasning, af de simulerede afkast. Da afkasttidspunktet, for hver pakke, er ens under forsøgsudførslen kan simuleringernes afkasttidspunkt ikke flytte sig relativt i forhold til hinanden. Derfor fortages en kurvetilpasningen, ved at flytte samtlige fire kurver, for pakkerne, i forhold til måledataene.

Kurvetilpasningen udføres ved en optimering mht. syv parametre i form af de tre translatoriske frihedsgrader, samt de fire Euler parametre. Optimeringen udføres ved at minimere følgende kostfunktion.

$$f(x_{ij}) = \frac{4}{n} \sum_{j=1}^{n} \left( \frac{1}{k} \sum_{i=1}^{k} x_{ij} \right)$$
(12.27)

Hvor k er antallet af målepunkter for den enkelte pakke og n er antallet af pakker.  $x_{ij}$  er den korteste vinkelrette afstand fra en lineær interpolation mellem simuleringspunkterne, til de enkelte målepunkter. Optimeringen udføres ved at anvende optimeringsrutinen *fminsearch* i MATLAB. Rutinen *fminsearch* tager udgangspunkt i *Nelder-Mead Simplex* metode, der er en ikke gradientbaseret optimeringsrutine givet af (Lagarias, et al., 1998). Kurvetilpasningen er foretaget mht. måledataene for de fire forsøg med de tre friktionsmodeller.

Resultaterne for kurvetilpasningen med Coulomb, Dahl og LuGres friktionsmodeller er givet i Appendiks XVII. De opnåede kurvetilpasninger af verifikationssimuleringerne anvendes i den videre behandling.

## 12.11. Vurdering af verifikationssimuleringer

Vurderingen af verifikationssimuleringerne er foretaget på baggrund af en visuel sammenligning mellem måledataene og simuleringerne mht. position, orientering og tid. Positionerne er vurderet på baggrund af plot af de målte positioner og kurver for, de simulerede afkast.

På Figur 12.19 er måledataene sammen med det simulerede afkast illustreret, for den store pakke, med en jævn masse, ned af trugslisken, med en sorterhastighed på 1.8 m/s. I simuleringen er der anvendt LuGres friktionsmodel.



Figur 12.19: Måledata (blå cirkler) samt simulerede afkast (sort) med LuGres friktionsmodel, med en sorterhastighed på 1.8 m/s ned af trugslisken. TV: Fra siden af slisken og TH: Ovenfra.

I Appendiks XVIII er graferne for de resterende forsøg sammenligner givet. Præcisionen af pakkernes orientering og tid er vurderet ud fra simuleringer, hvor den simulerede pakkes forløb er illustreret, sammen med de målte pakkers position og orientering. Dette er gjort ved indledningsvist at identificer det punkt i simuleringen som ligger tættest på det første punkt i hver måleserie. Dette punkt anvendes som referencepunkt. Det næste punkt i måleserien skal passe sammen med punktet i simuleringen 0.08 *sek.* fra referencepunktet. Dette hænger sammen med at de simulerede pakker er samplet ved 50 *Hz* og de målte data med 12.5 *Hz.* På baggrund af simuleringerne er det muligt at undersøge om tiden mellem simuleringen og de reelle afkast passer sammen.

På Figur 12.20 er en simulering illustreret for den store pakke (rød), med en jævn masse, ned at trugslisken, med en sorterhastighed på 1.8 *m/s*. De hvide pakker illustrerer de målte pakkepositioner til det pågældende tidspunkt. Til simuleringen er der anvendt LuGres friktionsmodel.



Figur 12.20: De målet pakkers (hvide) position og orientering ift. verifikationssimuleringen (rød) på trugslisken med en sorterhastighed på 1.8 m/s, hvor simuleringen anvender LuGres friktionsmodel.

På DVD-2 i mappen *Film* er vedlagt simuleringer med de målte pakkers position ift. verifikationssimuleringerne.

I det følgende er der givet en vurdering af simuleringsprogrammet, på baggrund af kurverne for afkastforløbene samt deres tilsvarende simuleringer.

Generelt er der ingen friktionsmodel der skiller sig ud ved at være bedre, end nogen af de andre modeller. Coulombs model afviger mindst ved forsøgene på trugslisken, mens Dahl og LuGre passer bedre med målingerne på vindelslisken.

I samtlige simuleringer med trugslisken vælter den store pakke, med den koncentrerede masse, når den rammer slisken. Der har under forsøgsudførslen været lignende situationer, hvor pakke med den koncentrerede masse vælter, idet den rammer slisken. Dette skyldes at pakken, har sin masse ud mod trugslisken. Når pakken rammer slisken medfører massens placering at pakken vælter.

Grunden til at pakken vælter i simuleringen er vurderet til at skyldes to årsager. Dels pga. friktionsmodstanden som generel for alle simuleringer giver pakkerne en langsommere bevægelse, ned gennem slisken, end de målte forsøgspakker. Den større modstand medfører at pakken bremses kraftigere idet den rammer slisken, hvilket for pakken, med den koncentrerede masse, bevirker at den vælter. Ligeledes kan pakkens kontaktlag være årsag til at pakken vælter. Pga. kontaktlaget, rykkes pakkens reelle kontaktpunkt, hvilket giver anledning til en lille fejl i det moment der påtrykkes pakken.

Ved simuleringerne på trugslisken med Dahls friktionsmodel vælter både den store og den lille pakke med koncentreret masse.

Ved simuleringerne ned gennem vindelslisken er der ingen pakker der vælter. For pakkerne med de koncentrerede masser skyldes det primært at massen i pakken er placeret modsat ift. vindelslisken.

Af Figur 12.19 samt graferne i Appendiks XVIII fremgår det at der en god overensstemmelse mellem simuleringerne af pakkens afkastforløb, og de målte forløb. Ligeledes passer især orienteringen af de simulerede pakker godt overens med de målte pakkeforløb. Generel er de simulerede pakker langsommere om at komme gennem sliskerne end de målte pakker. Dette er vurderet til at skyldes forskellen mellem sliskernes overflader og overfladerne anvendt til bestemmelse af friktionsparametrene. Ved de udførte friktionsforsøg blev der anvendt blankt stål, mens sliskerne til verifikationsforsøgene er malet.

Ved simuleringer af afkast på trugslisken med 1.8 *m/s* passer afkastet af de store pakker godt overens med målingerne, mens de små pakker har en lille afvigelse. Især ved Dahls og LuGres friktionsmodel er denne afvigelse tydelig.

Den bedste overensstemmelse mellem simuleringerne og måledataene er opnået ved trugslisken med en sorterhastighed på 2.2 *m/s*, hvor den største afvigelse er på ca. 150 *mm*. Ved vindelslisken er der ligeledes en god overensstemmelse mellem de målte afkast og simuleringerne. Generelt drejer pakkerne fra simuleringerne tidligere rundt i vindelen, mens målepunkterne har en mere retlinjet bevægelse. Dette skyldes primært at de simulerede pakker bremses mere i simuleringerne, når de rammer slisken, end pakkerne gør ved de reelle afkast. Ved den store pakke med koncentreret masse har den simulerede pakke en krum bane, hvorimod målepunkterne udgøre en mere retlinjet bevægelse. Dette skyldes at bagenden af pakken under målingerne rammer sliskekanten, hvilket giver pakken en rotation. Den roterende bevægelse bremses idet pakken rammer slisken, hvorefter pakken får en mere ret bevægelse ned gennem slisken. I dette tilfælde er LuGres friktionsmodel den der bedst kompenserer for dette.

## 12.12. Tilpasning af materialeparametre

Pakkerne i verifikationssimuleringerne er generel langsommere om at komme igennem sliskerne end pakkerne fra forsøgsudførslen. Dette er vurderet til primært at skyldes størrelsen af de anvendte friktionsparametre og restitutionskoefficienter. Det er derfor valgt at regulere disse, for at opnå mere korrekte simuleringer af afkastforløbet. Reguleringen er foretaget ved en optimering mht. materialeinputparametrene til simuleringen. Optimeringen er konstrueret i MATLAB, hvor *fminsearch* er anvendt som optimeringsrutine. For hver iteration udføres en simulering med materialeparametrene, specificeret af optimeringsrutinen. Ud fra simuleringsresultatet vurderes pakkernes afkast mht. måledataene ud fra følgende kostfunktion.

$$f\left(\mathbf{v}_{ij},\alpha_{ij},\beta_{ij},\chi_{ij}\right) = \frac{4}{n} \sum_{j=1}^{n} \frac{1}{k} \sum_{i=1}^{k} \left(\left|\mathbf{v}_{ij}\right| \cdot 10\right)^{2} + \frac{1}{3} \left(\left(\frac{\alpha_{ij}\cdot 12}{\pi}\right)^{2} + \left(\frac{\beta_{ij}\cdot 12}{\pi}\right)^{2} + \left(\frac{\chi_{ij}\cdot 12}{\pi}\right)^{2}\right) (12.28)$$

Hvor *k* er antallet af målepunkter for den enkelte pakke og *n* er antallet af pakker.  $\mathbf{v}_{ij}$  er vektoren fra målepunktet til det tilhørende simuleringspunkt og  $\alpha_{ij}$ ,  $\beta_{ij}$  og  $\chi_{ij}$  er vinklerne mellem målepunktets og simuleringspunktets koordinatakser.

Optimeringen er foretaget for samtlige tre friktionsmodeller mht. de udførte forsøg. Det er valgt at anvende samme restitutionskoefficient for både sliske, bakke og pakke for at reducer antallet af parametre. Derimod er det valgt ikke at anvende samme friktionsparametre mellem pakken og bakken, som mellem pakken og slisken. Det giver optimeringen tre frihedsgrader ved anvendelse af Coulombs friktionsmodel og hhv. 5 og 13 frihedsgrader ved Dahls og LuGres friktionsmodellerne. I Tabel 12.1, Tabel 12.2 og Tabel 12.3 er optimeringsresultaterne af de fire forsøg givet, for hhv. Coulomb, Dahl og LuGres friktionsmodel. S-P er friktionsparametrene mellem slisken og pakken og B-S mellem bakken og pakken.

Coulomb	Trugsliske 1.8	Trugsliske 2.2	Vindelsliske	Vindelsliske	Middelværdi
	m/s	m/s	1.8 m/s	2.5 m/s	
е	0.772	0.662	0.627	0.623	0.671
$\alpha_0$ S-P	0.145	0.186	0.235	0.225	0.198
$\alpha_0$ B-P	0.301	0.304	0.311	0.258	0.294
Start kost	47.98	43.69	41.28	54.59	
Slut kost	11.48	12.88	23.69	17.71	

Dahl	Trugsliske 1.8	Trugsliske 2.2	Vindelsliske	Vindelsliske	Middelværdi
	m/s	m/s	1.8 m/s	2.5 m/s	
е	0.540	0.516	0.507	0.481	0.511
$\alpha_0$ S-P	0.330	0.337	0.326	0.324	0.329
$\alpha_0$ B-P	0.326	0.334	0.328	0.352	0.335
$\sigma_0$ S-P	117321	119395	121628	119168	119378
$\sigma_0$ B-P	121744	120861	121505	127701	122953
Start kost	103.74	53.72	32.16	29.20	
Slut kost	25.41	19.20	26.45	21.89	

Tabel 12.1: Optimerede Coulomb friktionsparametre og restitutionskoefficient for de fire forsøg.

Tabel 12.2: Optimerede Dahl friktionsparametre og restitutionskoefficient for de fire forsøg.

LuGre	Trugsliske	Trugsliske	Vindelsliske	Vindelsliske	Middelværdi
	1.8 m/s	2.2 m/s	1.8 m/s	2.5 m/s	
е	0.541	0.505	0.501	0.514	0.515
$\sigma_{_0}$ S-P	123246	120950	121006	121034	121555
$\sigma_0$ B-P	118918	121043	120617	121855	120608
$\sigma_1$ S-P	10.15	10.07	10.07	10.18	10.12
$\sigma_1 B-P$	10.36	10.08	10.07	10.10	10.15
$\alpha_0$ S-P	0.278	0.319	0.329	0.311	0.309
$\alpha_0$ B-P	0.328	0.334	0.334	0.334	0.333
$\alpha_1$ S-P	-0.0726	-0.0706	-0.0710	-0.0713	-0.0714
$\alpha_1$ B-P	-0.0723	-0.0705	-0.0709	-0.0697	-0.0708
$\alpha_2$ S-P	0.0285	0.0282	0.0268	0.0280	0.028
$\alpha_2$ B-P	0.0296	0.0278	0.0283	0.0281	0.028
<i>v</i> <sub>0</sub> S-P	0.233	0.238	0.237	0.236	0.236
<i>v</i> <sub>0</sub> B-P	0.231	0.236	0.236	0.233	0.234
Start kost	79.77	33.57	34.13	36.4	
Slut kost	19.04	20.46	30.72	29.83	

Tabel 12.3: Optimerede LuGre friktionsparametre og restitutionskoefficient for de fire forsøg.

## 12.13. Vurdering af optimerede verifikationssimuleringer

Ud fra de opnåede optimeringsresultater er der foretaget en tilsvarende sammenligning mellem de optimerede simuleringskurver og måledataene, som i afsnit 12.11.

Det fremgår af optimeringsresultaterne at de opnåede værdier for kostfunktionerne har forbedrede sig en del. I nogle tilfælde op til ca. 400 %. For optimeringerne ved trugslisken har resultatet medført af den store pakken, med en koncentreret masse, ikke vælter idet den rammer slisken. Ved Dahls friktionsmodel vælter den lille pakke med den koncentrerede masse dog stadig.

Optimeringen af parametrene til Coulombs friktionsmodel har ved de fire forsøg fundet et godt resultat. Ved Dahl og LuGre har optimeringen haft svært ved at forbedre resultatet i samme omfang. Dette virker modstridende idet flere parametre burde giver anledning til en større tilpasningsgrad, af de simulerede afkastforløb. Der er på baggrund heraf udført optimeringer, hvor startgættet til Dahls friktionsmodel er varieret. Dette gav anledning til andre løsninger, der ikke var bedre end det oprindelige startgæt. Det er derfor valgt at anvende optimeringsresultaterne opnået på baggrund af det oprindelige startgæt.

På Figur 12.21 er det simulerede afkastforløb med de optimerede materialeparametre for LuGre illustreret, for den store pakke med jævn fordelt masse, ned gennem trugslisken med 1.8 m/s.



Figur 12.21: Simuleret afkastforløb med LuGres friktionsmodel af den store pakke med jævn fordelt masse gennem trugslisken med 1.8 m/s.

I Appendiks XIX er givet de resterende kurver for de simulerede afkastforløb, med de optimerede materialeparametre.

Orientering og tiden mellem de simulerede pakker og forsøgspakkerne, er illustreret ved simuleringer, som beskrevet i afsnit 12.11. På Figur 12.22 er den simulerede pakkes position og orientering mht. forsøgspakkernes position illustreret, ned gennem trugslisken, med en sorterhastighed på 1.8 *m/s*. Der anvendes LuGres friktionsmodel med de optimerede materialeparametre givet i Tabel 12.3.



*Figur 12.22: De målte pakkers (hvide) position og orientering ift. det simulerede afkastforløb, med de optimerede materialeparametre (rød) på trugslisken med en sorterhastighed på 1.8 m/s, med LuGres friktionsmodel.* 

På DVD-2 i mappen *Film* er vedlagt simuleringer med de målte pakkers position ift. verifikationssimuleringerne med de optimerede materialeparametre.

Sammenlignes Figur 12.22 med Figur 12.20 fremgår det at pakkernes positioner afviger mindre fra hinanden på Figur 12.22, hvor pakkerne især på de sidste billeder passer bedre sammen.

Der er ved optimeringerne med Coulombs friktionsmodel, opnået de laveste værdier for kostfunktionerne. Generelt er der en god overensstemmelse mellem de simulerede afkast og måledataene, både mht. position, orientering og tid.

På Figur 12.23 er illustreret afkastforløbet med Coulombs friktionsmodel, for den lille pakke med en jævn fordelt masse. Den venstre graf er med materialeparametre bestemt ved de udførte friktionsforsøg og den højre graf, med de optimerede materialeparametre.



Figur 12.23: Simuleret afkastforløb med Coulombs friktionsmodel, af den lille pakke, med jævn masse, gennem trugslisken med 1.8 m/s. TV: Med materialeparametre fra friktionsforsøg. TH: Med optimerede materialeparametre.

De optimerede materialeparametre for Coulombs friktionsmodel afviger mere fra startgættet, ift. de andre friktionsmodeller. Sammenlignes Dahl med Coulomb er den primære forskel mellem de to modeller, givet ved parameteren  $\sigma_0$ . For Dahl medfører  $\sigma_0$  en mere blød opbygning af friktionskraften, i starten af en ny kontakt, i modsætning til Coulomb, hvor friktionskraften er direkte afhængig af normalkraften. De lavere friktionsparametre bestemt ved Coulombs friktionsmodel er vurderet til at skyldes et forsøg på at undgå den pludselig opbygning af friktionskraften. Desuden er den fundne restitutionskoefficient for Coulomb omkring 0.17 større end for Dahl og LuGre. Den store restitutionskoefficient giver anledning til en lille ækvivalent dæmperkonstant, mellem legemerne, hvormed opbygningen af normalkraften bliver mindre. Den mindre normalkraft giver en mindre stødpåvirkningen, hvormed opbygningen af friktionsmodstanden i kontakttidspunktet ligeledes bliver mindre. Den store afvigelse mellem de optimerede materialeparametre og startgættet, samt afvigelsen mellem de enkelte forsøg ved Coulombs friktionsmodel, viser modellens evne til at tilpasse sig det enkelte optimeringsproblem. Derimod er det pga. middelværdiernes store spredning ikke hensigtsmæssigt at anvende disse, som generelle parametre, til afkastsimuleringer.

Ved optimeringen af materialeparametrene med Dahl og LuGre, er kostværdierne ca. den samme. Optimeringen har især forbedret afkastforløbet ned af trugslisken, mens det for vindelslisken ikke har givet nogen mærkbar forbedringer. Afvigelsen mellem de optimerede materialeparametre, for de fire forsøg, er små, og parametrenes middelværdi ligger tæt på startgættet. Den mindre variation mellem måledataene gør middelværdierne mere anvendelige som generelle parametre.

## 12.14. Valgt af friktionsmodel

På baggrund af den udførte verifikation er der valgt en friktionsmodel. Valget skal betragtes som en anbefaling i forbindelse med en videre anvendelse af simuleringsprogrammet. Ud fra den udførte verifikation er der ingen friktionsmodel som er mere fremtrædende en de andre. Det er derfor valgt at anvende Dahls friktionsmodel, af følgende årsager. Der er ikke nogen nævneværdig forskel i verifikationssimuleringerne mellem Dahl og LuGre og det vil derfor ikke være hensigtsmæssigt at anvende LuGre pga. mængden af parametre. Coulombs model giver for de enkelte forsøg et bedre resultater end Dahl. Der er dog ved Coulombs friktionsmodel en stor spredning i de optimerede resultater, hvormed der ikke kan gives et sæt generelle parametre. De optimerede friktionsparametre ved Dahl har derimod en langt mindre spredning, hvilket er mere hensigtsmæssig.

## 12.15. Delkonklusion

Der er på baggrund af målinger af reelle afkastforløb lavet en verificering af simuleringsprogrammet. Målingerne af pakkernes afkastforløb er udført på FKI Logistex testanlæg. Forsøgene er udført med to slisketyper, hvor fire forskellige pakker er afkastet, med to forskellige hastigheder.

Til bestemmelse af pakkernes position er der konstrueret et målesystem, hvor to kameraer vha. stereotriangulering bestemmer positionen af tre infrarøde dioder, placeret på pakkerne. Ud fra de udførte forsøg er afkastforløbet for de fire pakketyper bestemt ned gennem de to slisketyper. De fundne målepunkter for forsøgsudførslen er præget af en del usikkerheder. Pga. interlacing og støj fra udefrakommende lys er især dybdebestemmelsen af diodernes position præget af fejl.

Til verificeringen er der konstrueret fire simuleringer som svarer til de udførte forsøg. Simuleringerne er udført med hver af de tre implementerede friktionsmodeller, hvor friktionsparametrene bestemt i afsnit 8.2 er anvendt. Vurderingen af simuleringerne er foretaget på baggrund af overensstemmelsen mellem simuleringen og de målte data mht. position, orientering og tid.

Der er en generelt god sammenhæng mellem de målte og de simulerede afkastforløb, mht. samtlige tre implementerede friktionsmodeller. De simulerede pakkers position og orientering passer godt sammen. Derimod er pakkerne i simuleringerne generelt langsommere om af komme gennem slisken end de målte pakker.

På baggrund af den gode overensstemmelse mellem de målte og de simulerede afkastforløb, kan det konkluderes at den anvendte metode, til bestemmelsen af friktionsparametrene, beskrevet i afsnit 8.2, fungere efter hensigten.

Der er udført en optimering, mht. materialeparametrene til simuleringerne, for at opnå en bedre overensstemmelse mellem måledataene og simuleringerne.

Ud fra optimeringerne er der opnået en række forbedringer af de simulerede afkast. Især for Coulombs friktionsmodel er forbedringerne mærkbare, men er dog præget af en stor spredning mellem de fire forsøg. Spredningen af friktionsparametrene for de udførte forsøg med Dahl og LuGre er minimale.

Det er på baggrund af det udførte verifikationsforsøg valgt at anvende Dahls friktionsmodel i forbindelse med en videre anvendelse af simuleringsprogrammet.

Det kan konkluderes at det konstruerede simuleringsprogram, ud fra en sammenligning med reelle afkast, udviser en korrekt opførsel, af pakkernes forløb, ned gennem et givet sliskedesign.

### 12.15.1. Evaluering af anvendte målemetode

Den anvendte målemetode har virket efter hensigten og har givet et tilstrækkelig datagrundlag til verifikationen. Målesystemet har et højt potentiale og ville ved en række forbedringer kunne give et endnu bedre resultat. Metoden egner sig til det anvendte formål, hvor pakkerne skal kunne bevæge sig over en relativ stor afstand og på samme tid være upåvirket af målesystemet. Som det fremgår af de opnåede måleresultater har usikkerhederne i det anvendte målesystem givet anledning fejl, hvor især dybdebestemmelsen er præget af usikkerheder. Følgende fire punkter ville forbedre det anvendte målesystem.

- Mindre sollys
- Bedre opløsning
- Højere samplingsfrekvens
- Flere kameraer

Mængden af sollys har stor betydning for kvaliteten af målingerne. Det vil derfor være mest hensigtsmæssigt af udfører forsøg med målesystemet i rum, hvor udefrakommende lys kan fjernes. For at opnå en bedre præcision i bestemmelsen af centrumskoordinaterne, vil det være fordelsagtigt at anvende kameraer med større opløsning. Ligeledes vil en højere samplingsfrekvens være fordelagtig for at opnå flere målepunkter. Følsomheden over for fejl i dybdebestemmelsen, kan reduceres ved at anvende mere end to kameraer, hvilket også evt. vil øge målesystemets måleområde.

# 13. Optimering af sliskedesign

Det er FKI's ønske at kunne designe en ideel sliske til en given opgave, ved brug af færrest mulige resurser. Traditionelt løses denne designopgave gennem en kombination af erfaring og intuition. I de tilfælde hvor erfaringen ikke er tilstrækkeligt, udføres prototypeforsøg, indtil et tilfredsstillende resultat er opnået. For at mindske det nødvendige antal prototypeforsøg og spare på tidsressourcerne er der fremstillet et simuleringsprogram. Dette program virker som et analyseværktøj, hvorigennem de enkelte sliskedesigns kan testes. Et alternativt til denne empiriske indgangsvinkel er anvendelse af et optimeringsværktøj. Et optimeringsværktøj kan betragtes som den inverse til et analyseværktøj, hvor analyseværktøjet giver et designs egenskaber, mens optimeringsværktøjet ud fra ønskede egenskaber giver et design, se Figur 13.1.



Figur 13.1: Ved et analyseværktøj bestemmes et designs egenskaber, mens et optimeringsværktøj giver et design ud fra ønskede egenskaber.

Ved at samle simuleringsprogrammet og en optimeringsalgoritme, skabes et designværktøj, der gennem en iterativ proces simulerer, evaluerer og re-designer slisken, indtil det optimale sliskedesign er fundet

Dette kapitel giver indledningsvist en kort beskrivelse af principperne bag optimering, hvorefter der foretages en udvælgelse af optimeringsmetode. Herefter beskrives den valgte optimeringsmetode, der efterfølgende er implementeret til i et samlet optimeringsprogram. Afslutningsvis præsenteres tre udvalgte optimeringer, hvorudfra optimeringsprogrammet vurderes.

## 13.1. Optimerings terminologi

Optimering er en matematisk metode til at minimere et funktionsudtryk.

$$\min(g_0(\mathbf{x})) \tag{13.1}$$

Funktionsudtrykket  $g_0(\mathbf{x})$  betegnes kostfunktion og beskriver designrummet ud fra parametrene  $\mathbf{x}_1, \mathbf{x}_2 \dots \mathbf{x}_n$ .

Designrummet kan være begrænset af tvangsbindinger som er funktionsudtryk af parametrene. Der skelnes mellem to typer tvangsbindinger implicitte og eksplicitte. De eksplicitte tvangsbindinger kan beskrives på formen

$$x_j \le G_i, \qquad i = 1...m \qquad j = 1...n$$
 (13.2)

Hvor m er antallet af tvangsbindinger og n antallet af parametre. De implicitte tvangsbindinger kan beskrives på følgende form.

$$g_i(x_i, x_k) \le G_i \tag{13.3}$$

På Figur 13.2 er illustreret et eksempel på et designrum, med to parametre, begrænset af en implicit og eksplicit tvangsbinding.



Figur 13.2: Designrum begrænset af en implicit og en eksplicit tvangsbinding.

### 13.2. Valg af optimeringsrutine

Der findes en række forskellige optimeringsmetoder til bestemmelse af et designrums optimum. Disse kan opdeles i to hovedegrupper i form af de gradientbaserede og ikke gradientbaserede. De gradientbaserede optimeringsmetoder anvendes primært i tilfælde, hvor det er billigt at bestemme kostfunktionens gradienter. Eksempler på sådanne optimeringsrutiner er; *Steepest Descent, Conjugated Gradient, Davidson-Fletcher-Powel, Newton* samt for tilfælde hvor kostfunktionen er linear *Simplexmetoden*. I de tilfælde, hvor kostfunktionen ikke er givet analytisk eller gradienterne ikke på anden måde kan genereres billigt, er de ikke gradient baserede metoder et alternativt. Eksempler på sådanne rutiner er; *Monte Carlo, Nelder-Mead Simplex, complex* samt *Response Surface*.

I forbindelse med optimering af et sliskedesign, evalueres designet på baggrund at en simulering. Det er pga. simuleringen ikke muligt at bestemme et analytisk udtryk for kostfunktionens gradienter. Ligeledes vil en numerisk bestemmelse af gradienterne være beregningstung. Det er derfor valgt at anvende en ikke gradient baseret optimeringsrutine. Eftersom det i forbindelse med optimering af sliskedesignet vil indgå tvangsbindinger, er det ikke muligt at anvende Nelder-Mead simplex metode. Monte Carlo metoden er robust, ved problemstillinger med mange parametre, men til gengæld beregningstung. Complexmetoden er robust, og i stand til at løse optimeringsproblemer med et forholdsvis stort antal parametre, op til 36 (Manetsch, 1990). Response surface metoden fungere godt ved problemstillinger med få parametre, men bliver beregningstung når antallet af parametre øges (Madsen et al., 1997). På baggrund af et forventet antal parametre på mellem 10 og 20, er det valgt at anvende complexmetoden til optimering af sliskedesignet.
# 13.3. Complexmetoden

Complex optimeringsmetoden tager udgangspunkt i en evolutionel udvikling af en population af designs, complexen, der drives frem mod et optimum. Denne population af designs, er på nær startgættet, genereret ud fra tilfældigt udvalgte parametersæt.

Det første trin, i complexmetoden, er generering af en startpopulation. Formålet med denne population er at få repræsenteret designrummet. Dette sikres ved at vælge et passende antal parametersæt k. Det første parametersæt er et startgættet der skal opfylde alle tvangsbindingerne. De efterfølgende k-1 parametersæt er genereret tilfældigt. Denne generering sker ved at parametrene vælges tilfældigt inden for deres eksplicitte grænser, vha. en uniformfordeling. Opfylder det fundne parametersæt ikke de implicitte tvangsbindinger, rykkes gættet halvvejs ind mod tyngdepunktet, af complexsens allerede generede parametersæt. Under genereringen af complexen evalueres de enkelte parametersæt.

Efter at complexen er genereret begynder spejlingsprocessen, se Figur 13.3.



*Figur 13.3: Designrummet med optimeringens startpopulation (prikker). Under optimering spejles det dårligste design*  $\mathbf{x}_{old}$  *over de resterende punkters tyngdepunkt*  $\mathbf{x}_{o}$ *, med afstanden*  $\alpha \cdot \Delta$ *.* 

Med udgangspunkt i complexens kostværdier identificeres det parametersæt der har den højeste værdi. Dette parametersæt  $\mathbf{x}_{old}$ , erstattes, gennem en spejlingsproces, med et nyt og bedre parametersæt,  $\mathbf{x}_{new}$ . Dette udføres ved at spejle  $\mathbf{x}_{old}$  hen over tyngdepunktet af de resterende parametersæt  $\mathbf{x}_c$ , hvor hver indgang beregnes ved.

$$x_{i,C} = \frac{1}{k-1} \cdot \sum_{j=1}^{k} (x_{i,j}) - x_{i,\text{old}} \qquad i = 1, 2, \dots, n$$
(13.4)

Hvor *n* er antallet af designparametre. Spejlingen foretages som illustreret på Figur 13.3, givet ved.

$$\mathbf{x}_{new} = \mathbf{x}_C + \alpha \cdot \left( \mathbf{x}_C - \mathbf{x}_{old} \right) \tag{13.5}$$

Hvor  $\alpha$  er en spejlingsfaktor der normalt sættes til en værdi mellem 0.5 og 2 (Manetsch, 1990), oftest 1.3 (Box, 1965).

Før parametersættet  $\mathbf{x}_{new}$  kan erstattes af  $\mathbf{x}_{old}$  foretages følgende kontrol. Først kontrolleres  $\mathbf{x}_{new}$  mod de eksplicitte tvangsbindinger. Bryder  $\mathbf{x}_{new}$  en af disse, rykkes den eller de para-

metre, der overtræder bindingen, lige indenfor tvangsbindingen. Herefter kontrolleres parametersættet mod de implicitte tvangsbindinger. Brydes en af de implicitte tvangsbindinger, rykkes parametersættet halvvejs ind mod tyngdepunktet, hvorefter kontrollen gentages. Når både de implicitte og de eksplicitte tvangsbindinger er overholdt, evalueres  $\mathbf{x}_{new}$  (Box, 1965). Efter evalueringen sammenlignes kostværdien mellem  $\mathbf{x}_{old}$  og  $\mathbf{x}_{new}$ . Er  $\mathbf{x}_{new}$ 's kostværdi lavere end  $\mathbf{x}_{old}$  erstattes  $\mathbf{x}_{old}$  med  $\mathbf{x}_{new}$ . Er værdien ikke lavere, halveres afstanden ind til tyngdepunktet ved.

$$\mathbf{x}_{new}^{N_{rep}} = \frac{1}{2} \left( \mathbf{x}_{new}^{N_{rep}-1} + \mathbf{x}_{c} \right)$$
(13.6)

Hvor  $N_{rep}$  er antallet af evalueringer i den pågældende iterationen. Når det lykkes at finde et nyt og bedre parametersæt har optimeringen gennemført en iteration.



Figur 13.4: Ved at spejle det dårligste design i de resterende designs tyngdepunkt  $\mathbf{x}_c$  vil complexen gradvis samles. Processen fortsættes indtil alle designs er inden for afstanden  $\beta$ .

Efterhånden som rutinen gennemfører flere og flere iterationer, samler populationen sig gradvist omkring et optimum, se Figur 13.4. Når hele populationen er rykket så tæt sammen at den største afstand mellem parametersættene er mindre en end tolerance  $\beta$ , antages det at optimummet er fundet, se Figur 13.4.

# 13.3.1. Den adaptive complexmetode

Complexmetoden kan udvides til den adaptive complexmetode. Ved den adaptive complexmetode bliver optimeringen mere robust ved løsning af problemstillinger med mellem 10 og 36 parametre, i et multimodalt designrum (Manetsch, 1990). En principskitse for den adaptive complexmetode er vist på Figur 13.5.



Figur 13.5: Opbygningen af den adaptive complexmetode.

Udbygningen sørger for at størrelserne på  $\alpha$  og  $\beta$  tilpasses løbende, gennem optimeringsprocessen, samt at der undervejs er mulighed for at indsætte nye tilfældigt genererede parametersæt. Indsættelsen sker ved, at de dårligste sæt i complexen udskiftes med nye.

#### Reguleringen af $\alpha$

For at begrænse antallet af gange  $\alpha$  halveres, gennem én iteration, reguleres størrelsen kontinuert. Reguleringen sker på baggrund af en udjævnet værdi, *S* for antallet af evalueringer der skal til for at gennemføre én iteration. Er *S* lig én, er søgningen for forsigtig og  $\alpha$  er for lille. Hvis *S* derimod er større end 1.5, er  $\alpha$  for stor og søgningen for aggressiv (Manetsch, 1990). Det er af den grund valgt at regulere  $\alpha$ 's størrelse således at det gennemsnitlige antal evalueringer bliver 1.25.

Det udglattede antal evalueringer *S* beregnes ved anvendelse af en eksponentiel udglatning (Manetsch, 1990). Den enkelteksponentielle udglatning er givet ved (NIST).

$$S_{N} = \nu \cdot \sum_{i=1}^{N-2} \left( \left( 1 - \nu \right)^{i-1} \cdot y_{N-i} \right) + \left( 1 - \nu \right)^{N-2} \cdot y_{1}, \qquad 0 < \nu \le 1$$
(13.7)

Hvor *N* er antallet af iterationer, *y* er antallet af evalueringer og *v* er udglatningskonstanten sat til 0.3.  $\alpha$ 's størrelse er begrænset af en  $\alpha_{min}$  og  $\alpha_{max}$ . Med udgangspunkt i overstående kan følgende blokdiagram for sammenhængen mellem *S* og  $\alpha$  opstilles. *q* er en reguleringsfaktor for, hvor hurtigt  $\alpha$  ændrer sig når *S* afviger fra 1.25. I optimeringsprogrammet er *q* sat til 0.1.



Figur 13.6: Blokdiagram for reguleringen af  $\alpha$ .

#### Reduktion af $\beta$

Ved at foretage en løbende konvergenstest af optimeringsresultatet, ud fra størrelsen af  $\beta$ , er det muligt stoppe optimeringen når tyngdepunktets kostværdi varierer mindre end en given tolerance. Konvergenstesten udføres ved at lade  $\beta$ , blive mindre, hver gang complexen har samlet sig inden for <u>dens</u> nuværende størrelse, se Figur 13.7.



Figur 13.7: Kravet til  $\beta$  skærpes efterhånden som populationen samler sig.

Medfører ændringen af  $\beta$ 's størrelse en ændring i tyngdepunktets kostværdi, der er mindre end  $\delta$  gennem  $\gamma_b$  fortløbende iterationer, antages det at  $\beta$  er konvergeret og optimummet fundet (Manetsch, 1990).

#### Etapesøgning

Formålet med etapesøgningen er, gennem en hel optimering, at mindske det antal gange rutinen skal generere tilfældige parametersæt. Dette udføres ved at reducere complexens størrelse, og i stedet, gennem en række etaper, udskifte den ringeste del af complexen, med nye tilfældigt genererede parametersæt. Derved sikres at en fornuftig afsøgning af designrummet for et evt. bedre designs. Udskiftningen foretages når rutinen har fundet et midlertidigt optimum, ud fra ovenstående konvergenskriterier. Etapesøgningen fortsætter indtil, der  $\gamma_e$  gange, i træk, ikke har været en ændring af tyngdepunktets kostværdi der afvigere mere end den valgte tolerance  $\delta$ , se Figur 13.8.



Figur 13.8: Når  $\gamma_e$  etaper er inden for tolerancen  $\delta$ , antages det at rutinen er konvergeret.

Antallet af parametersæt der bibeholdes i complexen, *r*, har betydning for om det er de gamle eller de nye parametersæt der dominerer placeringen af complexens tyngdepunkt. Dette kan have betydning for hvorvidt spejlingsprocessen kan trække complexsen op af et evt. lokal minimum. Ud fra (Manetsch, 1990) skulle en størrelse på complexen, *k* mellem 1.2 og 4, gange antallet af parametre være tilstrækkeligt ved anvendelse af etapesøgning.

# 13.4. Implementering af optimeringsprogram

Optimeringsprogrammet er implementeret som en overbygning af det eksisterende simuleringsprogram. Overbygningen af programmet består af tre elementer, i form af en optimeringsrutine, en parametrisk model samt en kostfunktion, se Figur 13.9.





Indledningsvis indlæses inputfilen til optimeringen, samt et startgæt, i form af en simulering. Efter den indledende databehandling påbegyndes optimeringen. Optimeringen styres af optimeringesrutinen der udfører de enkelte evalueringer. Optimeringsrutinen giver et parametersæt til den parametriske model, hvorudfra et sliskedesign genereres. Det pågældende sliskedesign analyseres gennem en simulering, hvorefter resultatet vurderes i kostfunktionen. Når optimeringsrutinen har fundet et optimum brydes løkken, hvorefter resultaterne behandles. Ud fra resultatbehandlingen udskrives wrl-filen, for det optimerede sliskedesign. I det følgende er de implementere elementer i optimeringsprogrammet nærmere beskrevet.

# 13.5. Input til optimeringsprogram

I forbindelse med en optimering af et sliskedesign, indlæses de nødvendige informationer til optimeringsprogrammet, gennem to inputfiler. Indledningsvis indlæses en almindelig inputfil til en simulering, hvor den første faste geometri anvendes som startgæt til optimeringen. Herefter indlæses endnu en inputfil indeholdende data til optimeringsrutinen, den parametrisk model, samt kostfunktionen. En nærmere beskrivelse af inputfilen til optimeringen er beskrevet i Appendiks IX.

# 13.6. Optimeringsrutine

Det er i programmet valgt at implementere en modificeret version af den adaptive complexmetode, hvor følgende punkter er ændret.

- Spejlingsalgoritmen
- Begrænsning i antallet af spejlingsforsøg
- Stop kriterium for etapen
- Håndtering af eksplicitte tvangsbindinger

I det følgende er modifikationerne nærmere beskrevet, samt de anvendte værdier for  $\alpha$  og  $\beta$ .

### 13.6.1. Spejlingsalgoritme

Det er valgt at anvende et alternativ til ligning (13.6), der i stedet for at fortsætte halvering af afstanden ind til tyngdepunktet, korrigerer gættet mere over mod det bedste parametersæt i complexen  $\mathbf{x}_{best}$  (Andersen et al., 2002). Den modificerede spejlingsalgoritme er givet ved.

$$\mathbf{x}_{new}^{N_{rep}} = 0.5 \cdot \left( \mathbf{x}_{new}^{N_{rep}-1} + \varepsilon \cdot \mathbf{x}_{c} + (1 - \varepsilon) \cdot \mathbf{x}_{best} \right)$$
(13.8)

Hvor  $\varepsilon$  er givet ved følgende udtryk.

$$\varepsilon = \left(\frac{4}{4 + N_{rep} - 1}\right)^{\frac{4 + N_{rep} - 1}{4}}$$
(13.9)

# 13.6.2. Begrænsning i antallet af spejlingsforsøg

Det er valgt at begrænse antallet af spejlingsforsøg, i den enkelte iteration. Efter det femte forsøg, uden et bedre resultat, tildeles parametersættet en tilfældig placering på linjen mellem det sidste gæt og det bedste parametersæt i complexen. Dette udføres selvom det nye parametersæt er dårligere end det erstattede parametersæt. Dette accepteres for at give rutinen muligheden for at komme ud af en fastlåst situation.

#### 13.6.3. Stop kriterium for etapen

Det er, udover stopkriteriet givet i afsnit 13.3.1, valgt at påbegynde en ny etape, hvis værdien for *S* bliver stører end 3.9, da dette, ud fra erfaring, indikere at optimeringen er kørt fast. Et eksempel på en fast kørt situation er illustreret på Figur 13.10, hvor næsten hele complexen er samlet i et lokalt minimum, og enkelte parametersæt har fundet en bedre placering.



Figur 13.10: Situation hvor complexmetoden er kørt fast i et lokalt minimum.

I denne situation ligger  $\mathbf{x}_c$  omkring et lokalt minimum og det ikke muligt, for spejlingsalgoritmen at samle complexen inden for  $\beta$ . Ved at påbegynde en ny etape, gives complexen en ny chance for at samle sig i et bedre optimum.

#### 13.6.4. Eksplicitte tvangsbindinger

Det er i den implementerede version of optimeringsrutinen valgt, at anvende en anden metode til korrigering af parametersæt, der ligger uden for de eksplicitte tvangsbindinger. De parametersæt der bryder den eksplicitte tvangsbinding bliver flyttet indenfor grænsen, parallelt med den oprindelige spejlingsvektor, se Figur 13.11.



Figur 13.11: Bryder et nyt parametersæt en grænse rykkes den inden for den eksplicitte tvangsbinding parallelt med spejlingsvektoren.

Parametersættet flyttes inden for det tilladte område ved hjælp af nedenstående udtryk.

$$\mathbf{x}_{new}^{j} = \mathbf{x}_{C} + \left(\mathbf{x}_{new}^{j-1} - \mathbf{x}_{C}\right) \cdot \frac{x_{i,b} - x_{i,C}}{x_{i,new}^{j-1} - x_{i,C}} \qquad i = 1, 2, \dots n$$
(13.10)

Hvor  $x_{i,b}$  er grænsen for den eksplicitte tvangsbinding og *j* er antallet af korrigeringer.

# 13.6.5. Værdier for $\alpha \log \beta$

Det er valgt at anvende følgende værdier i reguleringen af  $\alpha$ 's størrelse. Som minimumsgrænse er  $\alpha_{min}$  sat til 0.5 og som maksimumsgrænse er  $\alpha_{max}$  sat til 2.0.

Den implementerede reduktion af  $\beta$  udføres ved at decimere værdien de to første gange, hvorefter  $\beta$  bliver halveret de følgende tre gange. Ved en eventuel yderligere reduktion, bliver  $\beta$  reduceret med 10/12. Denne reduktion fortsætter indtil tyngdepunktets kostværdi ændrer sig mindre end delta gennem  $\gamma_b$  fortløbende iterationer.

De initielle værdier for  $\beta$ ,  $\alpha$ ,  $\delta$ , k, r,  $\gamma_b$  og  $\gamma_e$  specificeres i inputfilen til optimeringen.

# 13.6.6. Implementering

Optimeringsrutinen er implementeret i optimeringsprogrammet i modulet *Complex\_op* og indeholdende fire subrutiner. Rutinen *Com\_opti* optiller complexen, hvorefter den starter optimeringen. Efter optimeringen udskriver rutinen det opnåede resultat. Optimeringsrutinen er opbygget på baggrund af Figur 13.5, hvor etapesøgningen er implementeret i subrutinen *Sim\_ComE* og reduktionen af  $\beta$  er implementeret i subrutinen *Sim\_ComB*. Complex søgerutinen samt reguleringen af  $\alpha$  er implementeret *i Sim\_Com*.

# 13.7. Parametrisk model

Den parametriske model har til formål at generere et design på baggrund at et parametersæt, specificeret af optimeringsrutinen. Modellen skal kunne genereres ud fra den i programmet indlæste sliskegeometri.

Antallet af frihedsgrader for den parametriske model, svarer til antallet af parametre i optimeringsrutinen. Antallet af frihedsgrader er derfor et kompromis mellem at skabe det størst mulige designrum, uden at optimeringsopgaven bliver uhensigtsmæssig beregningstung. Det er valgt at begrænse den parametriske model til kun at omhandle trugslisker.

# 13.7.1. Parametrisk model af sliskedesign

På baggrund af overstående problemstilling er der givet følgende fire forslag til en parametriske model af sliskedesignet. Forslagene repræsenteret fire forskellige niveauer af kompleksitet, hvormed den parametriske model kan genereres.

#### Princip 1

Dette princip tager udgangspunkt i en skalering og en flytning af hele sliskedesignet. Skaleringen udføres ved enten at trykke eller strække slisken individuelt i de tre koordinataksers retning. Dermed er det muligt at ændre slisken facon samt dens position, se Figur 13.12.



Figur 13.12: Slisken ændrer facon, ved at skaleret den oprindelige geometri langs de tre koordinatakser.

Denne parametriske model har seks frihedsgrader.

#### Princip 2

Princippet tager udgangspunkt i, at den indlæste geometri er styret af n individuelle tværsnit, se Figur 13.13. Det enkelte tværsnit tildeles fem frihedgrader i form af en skalering i bredde og højden, samt flytninger i de tre koordinataksers retning. For at bibeholde et realistisk sliskedesign må tværsnittene ikke bytte plads i forhold til deres oprindelige rækkefølge.



Figur 13.13: Den parametriske model beskrevet ud fra fire tværsnitsplaner.

Denne parametriske model har *m* frihedsgrader.

$$m = n \cdot 5 \tag{13.11}$$

#### Princip 3

Dette princip er en videreudvikling af ovenstående princip, hvor knuderne p i de n tværsnitsplaner frit kan bevæge sig i planerne, se Figur 13.14. Derudover kan planerne ligeledes forskydes i normalernes retning, med den begrænsning, at planerne skal bevare deres oprindelige rækkefølge.



Figur 13.14: Den parametriske model beskrevet ud fra knuderne i tværsnitsplanerne.

For at bevare et realistisk design må knuderne ikke bytte plads ift. naboknuderne. Denne parametrisk model har *m* frihedsgrader givet ved.

$$m = p \cdot 2 + n \tag{13.12}$$

#### Princip 4

Princippet tager udgangspunkt i at de enkelte knuder p, i den indlæste geometri, tildeles samtlige tre frihedsgrader. Den eneste begrænsning i knudeflytningerne, er at den oprindelige relation mellem de enkelte knuder skal være bibeholdt.



Figur 13.15: Parametrisk model, hvor samtlige knuder har tre frihedsgrader.

Denne model har  $m = p \cdot 3$  frihedsgrader.

# 13.7.2. Valg af parametisk model

Det er valgt at anvende princip 2 som udgangspunkt for den paramtriske model. Princippet giver en god fleksibilitet ud fra et begrænset antal frihedsgrader. Derudover passer opdelingen af tværsnittene godt overens med sliskens konstruktion, i form af bukkede pladesektioner.

# 13.7.3. Generering af den parametriske model

Den parametriske model bygger på skaleringer og flytninger af udvalgte tværsnit, defineret på baggrund af sliskens lokale koordinatakser  $\xi, \eta \, \text{og} \, \zeta$ . Tværsnittene er orienteret på en sådan måde at de er normale til den samme lokale koordinatakse, benævnet længdeaksen.

De resterende lokale koordinatakser vil herefter blive benævnt, bredde og højdeaksen, se Figur 13.16.

Det enkelte tværsnitsplans placering defineres på baggrund af de knuder i geometrien, langs længdeaksen, der ligger inden for en given tolerance  $\mu$ . Antallet af tværsnit bestemmes på baggrund af den valgte længdeakse, opdelingstolerancen samt knudernes placering.



Figur 13.16: Har en gruppe knuder samme lokale længdekoordinat, inden for tolerancen µ, udgør de et plan i modellen

Skaleringerne af tværsnitsplanerne, langs bredde og højdeaksen, benævnes  $s_b$  og  $s_h$ , og flytninger langs de lokale koordinat akser,  $f_{\xi}$ ,  $f_{\eta}$  og  $f_{\zeta}$ . Skaleringerne  $s_b$  og  $s_h$ , tager udgangspunktet i legemets lokale origo, og flytningerne  $f_{\xi}, f_n$  og  $f_{\zeta}$  foretaget ud fra tværsnittenes

initiale positioner.

Modellen har fem frihedsgrader, for hvert tværsnitsplan der er identificeret i geometrien. Med udgangspunkt i disse frihedsgrader er det muligt at ændre på sliskens geometri ved at skalere og flytte tværsnitsplanerne, se Figur 13.17.



Figur 13.17: Efter skaleringerne og flytningerne af de fundne tværsnitsplaner, gendannes geometrien på baggrund af knudernes relationer.

Efter skaleringen og flytningerne af de identificerede tværsnitsplaner, gendannes geometrien på baggrund af knudernes relationer. De nye knudekoordinater for knude i i plan j bestemmes ved

$$\begin{aligned} \xi_{j,inew} &= \xi_{j,istart} + f_{\xi,j} \\ \eta_{j,inew} &= \eta_{j,istart} \cdot s_{bj} + f_{\eta,j} \\ \zeta_{j,inew} &= \zeta_{j,istart} \cdot s_{hj} + f_{\zeta,j} \end{aligned}$$
(13.13)

Hvor det antages at  $\xi$ -aksen er længdeaksen og  $\eta$ -aksen er breddeaksen.

Under optimeringen generer optimeringsrutinen et parametersæt, i form af vektoren  $\mathbf{x}$ . På baggrund af denne vektor genererer den parametriske model et nyt sliskedesign. De første fem indgange i vektoren  $\mathbf{x}$  indeholder parametrene for plan nummer ét, de næste fem for plan nummer to, osv.

 $\mathbf{x} = \left[ s_{b,1}, s_{h,1}, f_{\zeta,1}, f_{\eta,1}, f_{\zeta,1}, s_{b,2}, s_{h,2}, f_{\zeta,2}, f_{\eta,2}, f_{\zeta,2}, \dots, s_{b,n}, s_{h,n}, f_{\zeta,n}, f_{\eta,n}, f_{\zeta,n} \right]$ (13.14) I det følgende betegnes det første plan i slisken, langs længdeaksen, indgangsplanet og det

sidste for udgangsplanet, se Figur 13.17.

### 13.7.4. Sliskegeometri til optimering

Startgættet til optimeringen gives i form af en SolidWorks wrlfil. For at få et hensigtsmæssigt startgæt er det vigtigt at knuderne kan grupperes i et passende antal tværsnit. Dette er vigtigt da et unødvendigt stort antal tværsnit vil give et stort antal parametre i optimeringen. I forbindelse med design af slisken, vil eventuelle krumme flader blive approksimeret med trekanter. Dette medfører at der vil blive genereret knuder andre steder end i de ønskede tværsnitsplaner. En nærmere gennemgang af en fremgangsmetode til generering af et fornuftigt design er beskrevet i Appendiks XX.

# 13.7.5. Eksplicitte tvangsbindinger

Til hver enkelt parameter i modellen er der givet et sæt eksplicitte tvangsbindinger. Tvangsbindingerne giver de øvre og nedre grænser, for den enkelte parameterværdi. Disse har til formål at holde geometrien inden for et anvendeligt område.

Med undtagelse af længdeaksen, anvendes de samme ydergrænser for forskydninger og skaleringer, for samtlige planer. Sliskens ydergrænser i længde, højde og bredde defineres i inputfilen til optimeringen. Det enkelte plans grænser langs længdeaksen, defineres på baggrund af planets initiale placering og sliskens ydregrænser, se Figur 13.18.



Figur 13.18: TV: I inputfilen defineres sliskens tolerancegrænser med hensyn til flytninger i længdeaksen. TH: Ud fra tolerancegrænserne genereres grænserne for det enkelte plan.

Med undtagelse af indgangsplanet må tværsnitsplanerne bevæge sig ud til de definerede ydergrænserne defineret i inputfilen. De enkelte planer skal dog bibeholde den oprindelige rækkefølge. Indgangsplanets flytninger begrænses i begge retninger af tolerancegrænsen, hvilket ligeledes defineret i inputfilen.

# 13.7.6. Implicitte tvangsbindinger

For at sikre at planerne forbliver i den rigtige rækkefølge er der givet en række implicitte tvangsbindinger. Rækkefølgen tager udgangspunkt i et referencepunkt i planet, samt planets initielle nummer, i rækkefølgen, se Figur 13.18. For en geometri opdelt i tre planer og med  $\xi$ -aksen som længdeakse, kontrolleres dette ved følgende betingelser.

$$\begin{aligned} \xi_{plan2} - \xi_{plan1} &> 0 \\ \xi_{plan3} - \xi_{plan2} &> 0 \end{aligned}$$
(13.15)

#### 13.7.7. Implementering

Den parametriske model er implementeret i modulet *parametic\_model*. Efter indlæsning af sliskegeometrien genereres den parametriske model i subrutinen *Point\_group*, hvor sliskens tværsnitsplaner identificeres. Ud fra de genererede planer opstilles de eksplicitte tvangsbindinger i rutinen *Constraints\_model1*. Under optimeringen kontrolleres, det pågældende parametersæt, med hensyn til de implicitte tvangsbindinger, i funktionen *Im1\_ok*. Opfylder parametersættet ikke de implicitte bindinger returneres parametersættet til optimeringsrutinen, hvor et nyt sæt genereres. Når parametersættet er godkendt af tvangsbindingerne, genereres det nye sliskedesign i subrutinen *para\_model1*.

# 13.8. Kostfunktion

Kostfunktionen beskriver, på baggrund simuleringsresultaterne, designrummet. Størrelsen af kostværdien afspejler designets evne til at opfylde de ønskede egenskaber, hvor en lav kostværdi er ensbetydende med et godt design. På baggrund af den variation der kan være i de afkastede pakker, er det valgt at kostfunktionen skal kunne vurdere sliskedesignet, ud fra en afkastsekvens med flere forskellige pakketyper. Ved at bedømme sliskedesignet ud fra et kombineret lasttilfælde, sikres det at resultatet af det optimerede sliskedesign ikke bliver specifikt, men at hele pakkevariationen kan repræsenteres. Der er i samarbejde med FKI opstillet 11 forskellige forudsætninger for et godt sliskdesign i form af:

- Stødpåvirkning af emnet
- Pakkens centrering i udløbet
- Pakkens orientering ved udløbet
- Konstant hastighed i det første stykke af slisken
- Konstant hastighed gennem hele slisken
- Ens udløbshastighed mellem pakkerne
- Sliskens kapacitet
- Vinkelhastighed ved udløb
- Vinkelacceleration ved udløb
- Bredden af sliskens udløb
- Længden, bredden og højden af slisken.

I det følgende er ovenstående elementer i kostfunktionen nærmere beskrevet samt, hvorledes de er implementeret i optimeringsprogrammet.

# 13.8.1. Kostfunktion pakker

I dette afsnit beskrives de elementer af kostfunktionen, der afhænger af pakken. Elementerne bliver beregnet enkeltvis for samtlige lasttilfælde, dvs. hver enkelt pakke i en afkastkombination.

#### Stødpåvirkningen

Det ønskes gennem et afkast at behandle de udsorterede pakker så skånsomt som muligt. Stødpåvirkningen bliver beregnet på baggrund den afsatte effekt, som beskrevet i afsnit 10.5. Den enkelte pakke straffes på baggrund af den maksimale stødpåvirkning gennem hele simuleringen.

#### Centrering af emnet ved udløbet

Centrering af emnet ved udløb, sikrer at emnet får en ensartet placering på det transportbånd, eller rullebane, der skal foretage den videre transport af emnet. I optimeringsprogrammet beregnes pakkens afvigelse fra centeret, ud fra afstanden mellem pakkens centrum og slutplanets centrum, idet pakkens centrum passerer slutplanet.

#### Orientering af emnet

De udsorterede pakker skal orienteres, således at de kommer ud igennem udgangsplanet på langs, se Figur 13.19.



Figur 13.19: Pakken i det øjeblik den passere udgangsplanet.

Med udgangspunkt i et plan udspændt af sliskens længde og højdeakse, samt en af pakkens lokale koordinatakser, bestemmes orientering af pakken ved passage af udgangsplanet. Koordinataksen udvælges på baggrund af pakkens dimensioner, hvor den akse normal til planet, udspændt at den længste og den korteste dimension, anvendes. Herved sikres det at pakken kommer ud på langs. Er flere af pakkens dimensioner ens, bestemmes vinklen til sliskeplanet for samtlige vektorer, langs de sider med ens dimensioner, og den mindste vinkel anvendes. Herved sikres det at kubiske kasser ikke kommer ud diagonalt.

#### Konstant hastighed gennem indløbet

For at undgå jam i indløbet, ønskes det at pakken har den samme hastighed, i indløbets normalretning, det første stykke ned gennem slisken. Derved sikres det at slisken hurtigt kan modtagelse en ny pakke, se Figur 13.20.



Figur 13.20: For at sikre kapacitet i indløbet er det et krav at pakkens hastighed er ens mellem indgangsplanet og kontrolplanet.

I inputfilen til optimeringen defineres et kontrolplan, som placeres, en given distance inde på slisken. Ud fra hastighedsforskellen mellem indgangsplanet og kontrolplanet, for den enkelte pakke, straffe designet.

#### Konstant hastighed gennem hele slisken

For at sikre et godt flow gennem systemet ønskes det at hastigheden, langs længderetningen, gennem hele slisken er konstant. Dette vurderes ved at bestemme hastighedsforskellen mellem pakkens passage af indløbsplanet og udløbsplanet. Designet straffes proportionalt med hastighedsforskellen.

#### Ens hastighed ved udløbet

For at sikre en let videre behandling af pakkerne ønskes det at alle pakker forlader slisken med samme hastighed. Dette sikres ved at minimere hastighedsforskellen mellem de forskellige pakker, ved passage af slutplanet. Straffen beregnes ud fra den enkelte pakkes hastighedsafvigelse fra en samlet middelværdi af pakkernes udløbshastighed.

#### Vinkelhastighed ved udløb

Ønsket om at have styr på den videre orientering og transport, medfører at rotationshastigheden ved passage af slutplanet skal være så lille som muligt. I simuleringsprogrammet bestemmes det tidspunkt, hvor slutplanet passeres, hvorefter rotationshastigheden bliver beregnet. Rotationshastigheden bliver straffet ud fra, hvor meget hastigheden afviger fra nul.

#### Vinkelacceleration ved udløb

Ligeledes ønskes det at minimere vinkelaccelerationen ved passage af slutplanet. Denne beregnes ligesom slutvinkelhastigheden og straffes ligeledes ud fra afvigelsen.

# 13.8.2. Sliskens kostfunktion

Kostfunktionerne for slisken kan direkte relateres til sorteringssystemets kapacitet, idet minimering af sliskens skyggeareal giver anledning til et øget antal slisker langs sorteringsbåndet. Slisken vurderes mht. højden, længden og bredden. Højden af slisken, er fra FKI, givet som den primære faktor af de tre sliskedimensioner. Sliskens højde bestemmes ved forskellen mellem det største og mindste globale z-koordinat. Længden af slisken bestemmes ud fra afstanden mellem punkterne med den største og den mindste værdi i det globale koordinatsystems x-retning. Bredden bestemmes på samme måde som de to ovenstående, dog i den globale y-retning.

#### Bredden af sliskeudløbet

Det er valgt at lade bredden af udløbet variere i et begrænset omfang, omkring en foruddefineret størrelse, givet i inputfilen. Denne begrænsede variation sikres ved at straffe afvigelsen af sliskedesignets udløbsbredde, mht. den foruddefinerede bredde, på kvadratet.

# 13.8.3. Sliskekapacitet

Kapaciteten af en sliske relaterer direkte til, hvor mange pakker der kan udsorteres, af den pågældende sliske, uden at der forekommer jam. Målet er, at den i optimeringen anvendte, pakkesekvens kan udsorteres gennem slisken, uden at der ophobes pakker. Dette sikres ved at drive designet hen mod at tidsforskellen, mellem emnerne passerer indgangsplanet, bliver mindst den samme som tidsforskel, mellem emnerne passerer udgangsplanet.

# 13.8.4. Andre kostfunktioner

I forbindelse med optimeringen er der implementeret en række elementer der skal sikre at der ikke findes løsninger, hvor pakken ikke kommer ned gennem slisken.

#### Indgangs- og udgangsareal

I det tilfælde, hvor pakken ikke kommer ind på slisken, har sliskedesignet fejlet. For at drive sliskedesignet væk fra sådanne løsninger, er det valgt at straffe designet, hvis pakken ikke kommer igennem et defineret indgangs og udgang areal i hhv. indgangsplanet og udgangsplanet. Indgangsarealets ydergrænser er givet af to lodrette linjer, igennem de to yderste knuder, samt en vandret linje gennem den nederste knude i indgangsplanets tværsnit, se Figur 13.21.



Figur 13.21: Indgangsarealet defineres på baggrund af yderknuderne i planets tværsnit.

Udgangsarealet begrænses af to lodrette linjer gennem de to yderste knuder i udgangsplanet, se Figur 13.22.



Figur 13.22: Udgangsarealet er defineret på baggrund af yderknuderne i planets tværsnit.

Hvis pakkens tyngdepunkt ikke passere disse arealer pålægges en straf.

#### Antallet af kontakter

Der er gennem optimeringer opnået resultater, hvor det optimerede sliskedesign, gennem en simulering, ikke har været i kontakt med pakken. På baggrund af disse løsninger er det valgt at straffe designet, hvis der ikke har været kontakt mellem pakken og slisken. I kostfunktionen tælles samtlige kontakter, med slisken, fra pakken er passeret indgangsplanet til den har passeret udgangsplanet. Hvis der er mere end tre kontakter i gennemsnit, pr. tidsstep, straffes designet ikke. Er det modsatte tilfældet straffes designet hårdere, jo færre kontakter der registreres.

#### 13.8.5. Normalisering af kostelementerne

For at vægte de enkelte kostdele ens, gennem optimeringen, er der lavet en normalisering af elementerne i kostfunktionen. I forbindelse af normaliseringen er det ikke muligt at bestemme én entydig sammenhæng mellem kostelementernes dimensioner. Det er derfor, på baggrund af en serie af optimeringer, udført en skalering, af det enkelte elementers bidrag, således at elementerne bidrager lige meget til kostværdien, når de alle er vægtet ens.

### 13.8.6. Samlet kostfunktion

Til beregning af den samlede kostværdi er det valgt at anvende en vægtet sum af de enkelte elementers kostværdier. Denne metode er valgt eftersom der er elementer i kostfunktionen der ikke kan opdeles i uafhængige lasttilfælde, dvs. pakkeafkast. Årsagen til dette skyldes bla. at det ikke kan udelukkes at pakkerne interagerer under afkastsekvensen. Den samlede kostfunktion bestemmes ved følgende udtryk.

$$f(g_0) = \frac{1}{n} \sum_{h=1}^{n} \sum_{i=1}^{o} f_P(g_{h,i}) + \sum_{j=1}^{p} f_L(g_j) + \frac{1}{n-1} \sum_{k=1}^{n-1} f_K(g_k) + \frac{1}{n} \sum_{h=1}^{n} \sum_{l=1}^{q} f_R(g_{h,l})$$
(13.16)

Hvor n er antallet af pakker i optimeringen, o er antallet af kostelementer for de enkelte pakker, p er antallet af kostelementer for slisken og q er antallet af de resterende kostelementer.

# 13.8.7. Vægtning af kostelementer

De enkelte elementer i kostfunktionen kan vægtes med en karakter mellem 0 og 5. Dermed er det muligt at prioritere nogle elementer højere eller lavere alt efter hvilken opgave det enkelte sliskedesign skal løse. Vægtningsfaktorerne defineres i inputfilen til optimeringen. I Tabel 13.1 er vægtningerne mellem de enkelte kostelementer angivet. Disse værdier er fundet i samarbejde FKI.

Kostelementer	Vægtning
Stødpåvirkningen	1-5
Centrering	4
Orientering	5
Konstant indløbshastighed	3
Konstant hastighed gennem hele slisken	4
Konstant udløbshastighed	2
Højden af slisken	5
Bredden af slisken	5
Længden af slisken	3
Sliskens kapacitet	3
Vinkelhastighed ved udløb	1
Vinkelacceleration ved udløb	1

Tabel 13.1: Vægtninger af kostelementer bestemt i samarbejde med FKI.

# 13.8.8. Implementering

Kostfunktionen er implementeret i modulet *Optim\_Kost*. I subrutinen *Kost\_Func* summeres alle kostelementerne til en samlet kostværdi for hhv. pakker og sliske. Summeringen foretages ved at kalde funktioner for de enkelte kostelementer.

# 13.8.9. Optimerede sliskedesign

Efter at optimeringen har opnået et optimum genereres der, på baggrund af det bedste parametersæt i complexen, en wrl-fil af det optimerede sliskedesign. Wrl-filen kan herefter importeres i SolidWorks som en solid part. Denne part kan så direkte indgå i det videre konstruktionsarbejde med sliskedesignet.

# 13.9. Optimeringsresultater

For at teste det konstruerede optimeringsprogram er der gennemført to forskellige optimeringer, begge udført med LuGres friktonsmodel. Optimeringerne har to formål, det ene er at undersøge om optimeringsprogrammet giver et plausibelt resultat og derudover undersøge om det er muligt, ved anvendelse af optimeringsprogrammet, at forbedre et allerede eksisterende design.

I det følgende beskrives sliskedesignet og optimeringsparametrene, anvendt til optimeringerne, hvorefter de anvendte pakke- og bakkegeometrier beskrives. Afslutningsvist bliver optimeringsresultaterne med to og tre pakker gennemgået.

#### 13.9.1. Startgæt til optimeringer

Det er i de udførte optimeringer valgt at tage udgangspunkt i et allerede eksisterende sliskedesign, i form af en trugsliske, se Figur 13.23. Denne sliske er inddelt i tre planer, ud fra reglerne beskrevet i afsnit 13.7.3, hvilket giver optimeringen 15 frihedsgrader.

Det er valgt at forhindre slisken i at flytte tættere på bakken, end udgangspositionen. Dette er gjort ved at sætte ydergrænsen  $\xi$ -min til nul, hvormed indgangsplanet låses i længdeaksens retning, se Tabel 13.2.

ξ-Max	ξ–Min	η-Max	η -Min	ζ-Max	ζ-Min	Min	Max
						skalering	skalering
0.1	0.0	0.5	-0.5	0.2	-0.2	0.5	2.0

Tabel 13.2: De anvendte grænser for flytninger og skaleringer af planerne i slisken.

Denne låsning af flytningen, i  $\xi$  retningen, reducerer antallet af frihedsgrader til 14.



Figur 13.23: Det til optimeringerne anvendte startgæt med de identificerede tværsnitsplaner.

Det er ikke i (Manetsch, 1990) givet nogle faste retningslinjer for fastsættelse af de til optimeringen anvendte parametre, hvorved der er anvendt et skøn, fundet ud fra tidligere optimeringer. De anvendte parametre er givet i Tabel 13.3.

k r		δ	$\gamma_e$	$\gamma_b$	
25	10	1.75	3	5	

Tabel 13.3: Anvendte parametre til de udførte optimeringer.

Vægtningsfaktorene anvendt i alle optimeringerne er dem der er givet fra FKI, hvor den maksimale stødpåvirkning er fastsat til fire se Tabel 13.1.

#### 13.9.2. Pakker- og bakkegeometri

Til de to optimeringer anvendes to forskellige pakkegeometrier i form af en rektangulær kasse der ifølge (FKI) normalt ikke er et problem at styre gennem slisken og en kvadratisk pakke der er mest kompliceret at håndtere. Dimensionerne på pakkerne er givet i Tabel 13.4.

	Højde [ <i>mm</i> ]	Længde [mm]	Bredde [mm]
Pakke 1	350	600	350
Pakke 2	200	500	500

Pakkernes masse, tyngdepunkts placering, masseinertimoment samt positionen på bakken varierer mellem de enkelte optimeringer.

### 13.9.3. Optimering med to pakker

Til optimeringerne med to pakker er der anvendt pakke 1 og pakke 2, se Tabel 13.4. Begge pakker har en jævnt fordelt masse på 15 kg. På Figur 13.24 er pakkernes startplacering illustreret. Der er under udførelsen af simuleringerne anvendt en simuleringstid på 5 *sek*, hvilket medførte en beregningstid på ca.14 *sek*. pr evaluering. Optimeringerne er gentaget tre gange for at have et datagrundlag til undersøgelse af optimeringsprogrammets evne til at finde den samme løsning hver gang.



Figur 13.24: Startopstillingen for optimeringen med to pakker.

Som sammenligningsgrundlag for de optimerede sliskedesign er foretaget en simulering af et afkastetforløb, med det eksisterende sliskedesign, se Figur 13.24. Resultatet at denne simulering opnåede en kostværdi på 165.0 og et forløb som vist på Figur 13.25.



Figur 13.25: Simuleringen med valgte pakkesekvens.

#### **Resultater af optimeringerne**

Alle optimeringerne, med to pakker, har samlet sig omkring et enkelt parametersæt, hvormed optimeringen har fundet et minimum. I Tabel 13.5 er kostværdien for de bedste design, samt antallet af evalueringer for de enkelte optimeringer givet.

Optimerings nr.	Kostværdi	Antal evalueringer	Samlet beregningstid [timer]
1	104.2	1211	4.7
2	108.9	826	3.2
3	88.1	1563	6.1

Tabel 13.5: De optimerede sliskers kostværdi, antallet af evalueringer samt beregningstiden for optimeringerne med 2 pakker.

Resultatet af de optimerede sliskedesigns samt pakkernes afkastforløb, gennem sliskerne, er illustreret på Figur 13.26.



Figur 13.26: Resultatet af sliskedesign ved de tre optimeringer med to pakker.

I Tabel 13.6 er det bedste parametersæt fra hver optimering givet. Tegninger med hovedmål på startgættet samt de optimerede slisker er vedlagt i Appendiks XXI.

	Sb	S <sub>h</sub>	$f_{\xi}[m]$	$f_{\eta}[m]$	$f_{\zeta}[m]$
Plan 1, optimering 1	1.077	1.187	0.000	-0.003	0.035
Plan 1, optimering 2	1.032	1.007	0.000	0.019	0.010
Plan 1, optimering 3	0.870	0.850	0.000	0.011	0.013
Plan 2, optimering 1	1.123	1.198	-0.334	-0.103	0.032
Plan 2, optimering 2	1.095	1.131	-0.193	-0.035	0.013
Plan 2, optimering 3	1.033	1.009	-0.635	0.066	-0.005
Plan 3, optimering 1	1.097	1.016	-0.777	-0.024	0.032
Plan 3, optimering 2	1.120	1.090	-0.623	0.088	-0.004
Plan 3, optimering 3	1.042	1.447	-0.952	-0.013	-0.066

			-	-					
$T_{\alpha} h_{\alpha} l_{\alpha}$	124	ς.	Dogultaton	fina	date		ntine onin o or	· · · · · · · · · · · · · · · · · · ·	a makkan
ranei	15.0	):	Resultater	ira.	ae ii	e o	numeringer	· mea i	o bakker.

Film af afkastforløbene er vedlagt på CD-ROM i mappen Film.

#### 13.9.4. Optimering med tre pakker

Der er gennemført tre optimeringer med tre pakker, som illustreret på Figur 13.27. De anvendte pakker har målene som angivet i Tabel 13.4. Samtlige pakke har en jævnt fordelt masse, hvor de to forreste har en masse på 15 kg og den bagerste en masse på 30 kg. Den bagerste pakke er placeret længere tilbage på bakken, hvilket giver en yderligere variation i afkastforløbet. Der er anvendt en simuleringstid på 6 *sek.*, hvilket giver en evalueringstid, med tre pakker, på ca. 25 *sek.* 



*Figur 13.27: Startplaceringen af pakkerne ved optimeringen med tre pakker.* 

På Figur 13.28 er afkastforløbet med startslisken givet. Denne simulering opnåede en kostværdi på 179.0.



Figur 13.28: Simulering med den valgte pakkesekvens og startgæt.

De tre optimeringer har alle fundet et resultat, hvor samtlige parametersæt i complexen har samlet sig i samme minimum. Kostværdierne for de fundne løsninger er givet i Tabel 13.7.

Optimerings nr.	Kostværdi	Antal evalueringer	Samlet beregningstid [timer]
4	97.9	1287	8.9
5	87.6	4011	27.9
6	80.6	4238	29.4

Tabel 13.7: De optimerede sliskers kostværdi, ved optimeringer med tre pakker.

På Figur 13.29 er de optimerede sliskegeometrier illustreret, hvor pakkernes afkastforløb gennem de optimerede sliskegeometrier er vist på Figur 13.30.



Figur 13.29: De optimerede sliskedesigns.



Figur 13.30: Afkastforløbet for resultaterne af optimeringerne men tre pakker.

I Tabel 13.8 er parametersættene til de optimerede sliskegeometrier givet. Tegninger af de optimerede slisker med hovedmål er givet i Appendiks XXI.

	Sb	S <sub>h</sub>	$f_{\xi}[m]$	$f_{\eta}[m]$	$f_{\zeta}[m]$
Plan 1, optimering 4	1.468	0.970	0.000	0.056	0.024
Plan 1, optimering 5	1.170	0.929	0.000	-0.199	-0.007
Plan 1, optimering 6	0.993	0.739	0.000	-0.079	-0.022
Plan 2, optimering 4	1.117	0.894	-0.655	0.071	-0.008
Plan 2, optimering 5	1.093	1.173	-0.850	-0.091	-0.038
Plan 2, optimering 6	0.715	1.359	-0.841	0.183	-0.015
Plan 3, optimering 4	1.034	1.073	-0.710	0.13	0.057
Plan 3, optimering 5	1.009	1.124	-0.801	-0.046	0.096
Plan 3, optimering 6	1.018	1.172	-0.865	-0.250	-0.027

Tabel 13.8: De optimerede parametersæt for skaleringer og forskydninger, for hver af de tre optimeringer.

Film af afkastforløbene er vedlagt på CD-ROM i mappen Film.

# 13.10. Vurdering optimeringsresultater

Ovenstående tests er udført for at undersøge om optimeringsprogrammet kan finde frem til et sliskedesign, der virker efter hensigten, og eventuelt finde en løsning der er bedre til den pågældende opgave end den eksisterende sliske, givet af FKI.

Optimeringsresultaterne viser at de optimerede sliskedesigns alle orienterer pakkerne bedre end den eksisterende sliske, hvilket specielt er tydeligt for simuleringen med tre pakker, hvor det ikke lykkedes for det oprindelige sliskedesign at vende den tunge pakke, inden den når udgangsplanet. Optimeringsresultaterne opfylder ligeledes ønsket om at minimere skyggearealet, hvor samtlige optimerede slisker fylder et mindre areal end startdesignet. En af de ting optimeringerne ligeledes viser er, at der er en stor variation mellem de enkelte optimeringsresultater, hvor det kun er en ud af tre optimeringer der opfylder ønsket om at minimere indløbsbredden. Disse principielt forskellige løsninger antyder at optimeringsproblemet er et semidiskret problem, hvor der findes flere, principielt forskellige løsninger til den samme opgave.

Problemet med sliskens indløbsbredde kunne løses ved at give en maksimal bredde på indløbet, hvilket vil tvinger optimeringen til at finde en løsning med et smallere indløb. Alternativt kunne kostfunktionens straf for at gøre indløbsbredden større, ændres, således at et optimeret design, med et bredere indløb bliver mindre attraktivt. En ensretning af resultaterne, fra optimeringerne, kan forsøges sikret ved at minimere grænserne for optimeringerne. Denne løsning vil medføre en længere beregningstid, uden der er nogen garanti for at optimeringsprogrammet finder mere ensartede løsninger. Alternativt kunne de nuværende grænser bibeholdes, hvilket medfører at der skal køres flere ens optimeringer, før der kan udvælges én løsning.

Testene af optimeringsprogrammet har vist at der kan findes en løsning på et givent problem der både fylder mindre og ensretter pakkerne bedre

# 14. Konklusion

Motivationen for dette projekt er et ønske fra FKI om et CAD baseret simuleringsværktøj, der kan vise emners afkastforløb ned gennem slisker.

Der er konstrueret et simuleringsprogram der er tiltænkt som et indledende værktøj, hvor et sliskedesign kan testes inden en videre konstruktion foretages. Som input til programmet kan CAD geometrier fra SolidWorks anvendes. Ligeledes kan informationer om legemernes materiale, masse, tyngdepunkt og masseinertimoment specificeres. Ud fra en simulering er det muligt vha. en visuel repræsentation at vurdere det pågældende sliskedesignets anvende-lighed.

Gennem en indledende analyse af afkastforløbet, er antallet af komponenter reduceret, hvormed kun de vigtigste komponenter er inddraget i simuleringen. Til beskrivelse af legemernes bevægelse er der anvendt stivlegemedynamik. Med udgangspunkt i Newtons tre love er legemernes bevægelse beskrevet vha. to differentialligninger, der løses numerisk. Der er implementeret tre friktionsmodeller i form at Coulomb, Dahl og LuGre, hvor modellerne udgør tre niveauer af kompleksitet. Friktionskoefficienterne til modellerne er bestemt ved et friktionsforsøg, hvor friktionsmodstanden er bestemt med udgangspunkt i reelle afkast. Der er konstrueret en kontaktalgoritme til bestemmelse af reaktionen mellem legemerne. Algoritmen er inddelt i en grov og en fin kontaktbestemmelse.

De implementerede enkeltdele i simuleringsprogrammet er testet for at eftervise at disse fungerer efter hensigten. Testene er udført mht. dynamik, friktion og kontakt, hvorefter en samlet simulering af et afkastforløb er udført. I henhold til FKI Logistex ønsker er simuleringsprogrammets evne til at eftervise jam og selvstart testet. Alle enkeltdelene og tests virker efter hensigten.

Der er på baggrund af målinger af reelle afkastforløb lavet en verificering af simuleringsprogrammet. Til bestemmelse af pakkernes position er der konstrueret et målesystem, hvor to kameraer vha. stereotriangulering bestemmer positionen af pakkerne.

Der er fundet en god sammenhæng mellem de målte og de simulerede afkastforløb, mht. samtlige tre implementerede friktionsmodeller. De simulerede og målte pakkers position og orientering passer godt sammen. Derimod er pakkerne i simuleringerne generelt langsommere om at komme gennem slisken, end de målte pakker.

Det kan konkluderes, at det konstruerede simuleringsprogram, ud fra sammenligning med reelle afkast, udviser en korrekt opførsel, af pakkernes forløb, ned gennem et givet sliskedesign. Ud fra den udførte verificering, kan det konkludere at den ønskede tillid til simuleringerne er opnået. På baggrund heraf kan simuleringsprogrammet anvendes som et værktøj til test af nye sliskedesigns.

Der er udført en optimering, mht. materialeparametrene, til simuleringerne, for at opnå en bedre overensstemmelse mellem måledataene og simuleringerne. Ud fra optimeringerne er der opnået en række forbedringer, hvor der især er opnået en bedre overensstemmelse mht. tiden. På baggrund af den anvendte parameterestimering er det erfaret at metoden kan anvendes i forbindelse med bestemmelse af nye materialeparametre. Det er ud fra de udførte verifikationsforsøg valgt at anvende Dahls friktionsmodel i forbindelse med en videre anvendelse af simuleringsprogrammet. Dette valg er gjort på baggrund af modellens simple opbygning samt de ensartede resultater, opnået ved verificeringen. Simuleringsprogrammet er udvidet til et designværktøj, hvor det ud fra en række ønskede egenskaber er muligt at bestemme et nyt sliskedesign. Til designværktøjet er der valgt en ikke gradientbaseret optimeringsmetode, i form af complexmetoden. Den optimerede sliskegeometri udskrives efter optimeringen til en fil der, efterfølgende, kan importeres til SolidWorks. Der er udført seks optimeringer for at eftervise designværktøjets evne til at finde nye bedre sliskedesigns. Det kan konkluderes, ud fra resultaterne, at den anvendte optimeringsmetode finder løsninger på baggrund af den parametriske model og kostfunktionen. Det er ud fra optimeringerne opnået resultater, hvor sliskernes skyggeareal er reduceret. De optimerede sliskedesigns varierer i udformning, hvorudfra det er vurderet at der er tale om et semidiskret designrum, hvor der er flere forskellige løsningsprincipper til den samme opgave. Det kan konkluderes at der er udviklet et designværktøj der fungerer efter hensigten.

# 15. Kildeliste

**Aitenbichler Erwin og Mühlhäuser Max** An IR Local Positioning System for Smart Items and Devices [Tidsskrift]. - [s.l.] : Darmstadt University of Technology, 2003.

Alpha Sound Denmark Alpha Teknik Denmark [Online] //

http://www.alphateknik.dk/default.asp?Pri=11&Sek=120&data=190270. - 18. 4 2007.

Andersen et al. Torben Ole, Hansen, Michael Rygaard Elektrohydrauliske systemer [Konference] // LivsLang udannelse. - Aalborg : AAU, 2002.

**Armstrong-Hélouvry B.** Control of machines with friction [Bog]. - [s.l.] : Kluwer Academic Publisher, 1991. - 1. Udgave. - 0-7923-9133-0.

**Bouguet Jean-Yves** http://www.vision.caltech.edu/bouguetj/calib\_doc/ [Online] // Camera Calibration Toolbox for Matlab. - Marts 2007.

**Box M. J.** A New method of constrained optimization and comparison with other methods [Artikel] // Computer journal. - 1965.

**Burden Richard L. og Faires J. Douglas** Numerical analysis [Bog]. - 2001. - 7 : Årg. Burden faires. - 0-534-38216-9.

**Canudas de Wit C. [et al.]** A New Model for Control of Systems with Friction [Artikel] // IEEE Transactions on automatic control. - 1995. - 40. - 3.

**Canudas de Wit C. og Lischinsky P.** Adaptive Friction Compensation with Partially Known Dynamic Friction Model [Artikel] // International Jurnal of Adaptive Control and Signal Processing. - 1997. - 11. - 65-80.

Charles-Augustin de Coulomb [Online] //

http://www.geocities.com/bioelectrochemistry/coulomb.htm.

**Coutinho Murilo G** Dynamic Simulations of Multibody Systems [Bog]. - New York : Springer-Verlag, 2001.

Dahl P., R. A Solid Friction Model [Artikel]. - 1968.

Dahl P., R. Solid Friction Damping of Mechanical Vibrations [Artikel]. - 1976. - 14. - 12.

Dahl P., R. Solid Friction Damping of Spacecraft Oscillations [Artikel]. - 1975.

**Defrancesco S. og Gratton L., M.** A Simple Measurement of the Sliding Friction Coefficient [Artikel]. - 2006.

**Dupont P., E. og Bapna D.** Stability of Sliding Frictional Surfaces with Varying Normal Force [Artikel]. - 1993.

**Ecolight.dk ApS** Ecolight.dk [Online] // http://ecolight.dk/catalog/infrar%C3%B8d-diode-ir204a-p-128.html?osCsid=3c10f9e5c01b3a3349b2afc78e1111b7. - 18. 4 2007.

**Erleben Kenny** Stable, Robust, and Versatile Multibody Dynamics Animation [Artikel]. - 2004.

FKI Logistex [Online] // http://www.fkilogistex.com/scandinavia/.

**Garcia de Jalón Javier og Bayo Eduardo** Kinematic and Dynamic Simulation og Multibody Systems [Bog]. - [s.l.] : Springer, 1993. - 978-0387-94096-0.

**Goldsmith Werner** Impact [Bog]. - [s.l.] : Dover Publication, 2001. - 2. Udgave. - 0-486-42004-3.

**Goyal Suresh, Pinson Elliot, N. og Sindel Frank, W.** Simulation of Dynamics of Interacting Rigid Bodies Including Friction I [Artikel]. - [s.l.] : Springer-Verlag, 1994 I. - I. **Goyal Suresh, Pinson Elliot, N. og Sindel Frank, W.** Simulation of Dynamics of Interacting Rigid Bodies Including Friction II [Artikel]. - [s.l.] : Springer-Verlag, 1994. **Heikkilä Janne** Accurate camera calibration and feature based 3-D reconstruction from monocular image sequences [Rapport]. - Oulu : University of Oulu, 1997.

Heikkilä Janne og Silvén Olli A Four-step Camera Calibration Procedure with Implicit Image Correction [Artikel]. - Oulu, Finland : University of Oulu.

**Hippman Gerhard** An algorithm compliant contact between complexy shapes surfaces in multibody dynamics [Artikel]. - 2004.

IDS IMAGING DEVELOPMENT SYSTEMS IDS [Online] // http://www.ids-

imaging.com/frontend/catalog\_detail.php?pID=479&form\_KatTable=menue2&form\_KatID =69&nav1=22&nav2=69. - 18. 4 2007.

Introduction to Tribology - Friktion [Online] //

http://depts.washington.edu/nanolab/ChemE554/Summaries%20ChemE%20554/Introductio n%20Tribology.htm.

Iocchi Luca Stereo Vision: Triangulation [Online] //

http://www.dis.uniromal.it/~iocchi/stereo/triang.html. - Febuar 2007.

Jean-Yves Bouguet Stereo Triangulation in Matlab [Tidsskrift]. - 1998. - EE CNS148. Kodak Kodak [Online] //

http://www.kodak.com/US/en/motion/support/h1/filtrationP.shtml. - 18. 4 2007.

**Krex H** Maskinståbi [Bog]. - [s.l.] : Nyt Tekniske Forlag, 2004. - 9. Udgave. - 87-571-2547-3.

**Kreyszig Erwin** Advanced Engineering Mathematics [Bog]. - [s.l.] : John Wiley, 1999. - 8. Udgave. - 0-471-33328.

**Lagarias Jeffrey, C [et al.]** Convergence properties of the Nelder-Mead simplex method in low dimensions [Artikel]. - [s.l.] : Society for Industrial and Applied Mathematics, 1998. - 9. - 1.

Lin Ming, C. Efficient Collision Detection for Animation and Robotics [Artikel]. - 1993. Madsen et al. Response SSurface Techbiques for Diffuser Shape Optimization

[Konference] // 28. AIAA Fluid Dynamics Conference. - Snowmass village : AIAA, 1997. **Manetsch T. J.** Toward Efficient Global Optimization in Large Dynamic Systems - The Adaptive Complex Method [Artikel] // IEEE. - 1990.

Meriam J., L. og Kraige L., G. Engineering Mechanics Dynamics [Bog]. - [s.l.] : Wiley, 2003. - 5. Udgave. - 0-471-26606.

Mirtich Brian V-Chip: Fast and Robust Polyhedral Collision Detection [Artikel]. - 1998. Mirtich Brian, Vincent Impulse-based Dynamic Simulation of Rigid Body Systems [Artikel]. - 1996.

**Nikravesh Parviz, E.** Computer-Aided Analysis of Mechanical [Bog]. - [s.l.] : Prentice Hall, 1988. - 0-13-164220-0.

**NIST** Engineering Stasistics Handbook [Online] // NIST/SEMATECH e-Handbook of Statistical Methods. - 02. 04 2007. - http://www.itl.nist.gov/div898/handbook/.

Olsson H. [et al.] Friction Models and Friction Compensation [Artikel]. - 1997.

**Press William, H. [et al.]** Numerical Recipes in Fortran 77 [Bog]. - [s.l.] : Cambridge University Press, 1997. - 2. Udgave. - 0-521-43064-x.

**Rabinowicz E.** The Intrinsic Variables affecting the Stick-Slip Process [Artikel]. - 1957. **Rao Singiresu, S.** Mechanical Vibrations [Bog]. - [s.l.] : Pearson Printice Hall, 2004. - 4. Edition. - 0-13-120768-7.

**Shabana Ahmed, A.** Dynamics of Multibody Systems [Bog]. - [s.l.] : Cambridge Univercity Press, 1998. - 2. Udgave.

**Trucco Emanuele og Verri Alessandro** Introductory Techniques [Bog]. - [s.l.] : Prentice Hall, 1998.

**Walpole R., E. [et al.]** Probability and statistics for engineers and sceientists [Bog]. - 2002. - 7. Udgave. - 0-13-041529.

Wikipedia Encyclopedia [Online] // http://da.wikipedia.org/wiki/Leonardo\_da\_Vinci. Wikipedia Encyclopedia [Online] //

http://en.wikipedia.org/wiki/Charles\_Augustin\_de\_Coulomb.

Wikipedia Encyclopedia [Online] // http://en.wikipedia.org/wiki/Tribology.

Wikipedia Linear least squares [Online] //

http://en.wikipedia.org/wiki/Linear\_least\_squares. - 2. Marts 2007.

Wikipedia Wikipedia [Online] // http://en.wikipedia.org/wiki/Interlace. - 12. 5 2007.

Xiuzhi Qu. og Stucker B. A 3D surface offset method for STL-format models [Artikel] // Rapid Prototyping Jurnal Volumen 9 Number 3. - 2003. - s. 133-141.

**Zachmann** Virtual Reality in Assembly Simulation Collision Detection, Simulation Algorithms, and Interaction Techniques [Rapport]. - Darmstadt : Dem Fachbereich Informatik der Technischen Universität Darmstadt eingereichte, 2000.

Åström K., J. og Canudas C. A New Model for Control of Systems with Friction [Artikel] // IEEE transzction on automatic control. - 1995. - 40. - 3.