

Bepalen rotatiecentra van wervelbewegingen in nekmodel

Citation for published version (APA):

Verstegen, J. H. J. (1995). *Bepalen rotatiecentra van wervelbewegingen in nekmodel*. (DCT rapporten; Vol. 1995.046). Technische Universiteit Eindhoven.

Document status and date:

Gepubliceerd: 01/01/1995

Document Version:

Uitgevers PDF, ook bekend als Version of Record

Please check the document version of this publication:

- A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
- The final author version and the galley proof are versions of the publication after peer review.
- The final published version features the final layout of the paper including the volume, issue and page numbers.

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy

If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

**BEPALEN ROTATIECENTRA
VAN WERVELBEWEGINGEN IN
NEKMODEL**

Auteur: J.H.J. Verstegen
Stageverslag

Begeleiders : Marko de Jager
Fons Sauren

WFW rapportnr : 95.046

Eindhoven, april 1995

Samenvatting

Om het gedrag van de menselijke nek te kunnen voorspellen tijdens ongevalsituaties is er een mathematisch model van de nek ontwikkeld. Met dit model is het bijvoorbeeld mogelijk om krachten en versnellingen te berekenen op de verschillende nekwerfels en spieren.

Er zijn een aantal mogelijkheden om het model te valideren, een van deze mogelijkheden is het bepalen van rotatiecentra van de wervels en te vergelijken met de werkelijke positie van deze rotatiecentra. Het rotatiecentrum van een lichaam is gedefinieerd als het punt waarom een lichaam roteert. Dit rotatiecentrum bestaat voor iedere beweging behalve een zuivere rotatie. Bij vrijwillige bewegingen is het rotatiecentrum bekend. Voor ongevalsituaties wordt vermoedt dat deze op dezelfde positie liggen.

Om deze rotatiecentra te bepalen is er een programma geschreven wat met behulp van de gegevens uit het mathematische model de rotatiecentra bepaalt. Door nu deze rotatiecentra te vergelijken met de werkelijke rotatiecentra is het mogelijk om het model te valideren.

Om de werking van het programma te controleren zijn er een tweetal andere modellen doorgerekend. Met de gegevens van deze modellen, een daarvan is het twopivot model, de andere is een fictief model dat uit een vijftal lichamen bestaat, zijn de rotatiecentra bepaalt. Uit grafische controle blijkt dat het programma voor deze modellen met vaste scharnierpunten goed werkt. Bij het testen van het nekmodel was alleen de uitvoer van het programma van belang en niet de resultaten van het nekmodel, met andere woorden de test was niet bedoelt om het model te evalueren maar het programma. Ook bij het nekmodel bleek het programma op een juiste manier de rotatiecentra te berekenen.

Inhoudsopgave

1.	Inleiding	1
2.	Methodes voor het bepalen van het rotatiecentrum	2
2.1	Methode van Reuleaux	2
2.2	De rigid body methode	3
2.3	Methode met twee cirkels.	5
2.4	Methode van Bowyer en Woodwark	7
3.	Evaluatie van de verschillende methodes	8
4.	Conclusie ten aanzien van de berekeningsmethodes	9
5.	Werking van het programma	12
5.1.	Uitvoering van de rigid body methode in FORTRAN	12
5.2.	Relatieve en absolute positie	13
5.3.	Globale werking van het programma	15
5.3.1.	Invoer/uitvoer van CR	16
5.3.2.	Gebruik van direct access files	17
6.	Testen van het programma	20
6.1.	Resultaten met twopivotmodel	20
6.2.	Resultaten met fivepivotmodel	21
6.3.	Resultaten met nekmodel	22
7.	conclusies	23
	Literatuurlijst	24

1. Inleiding

Er is een wiskundig model ontwikkeld van de menselijke nek om het gedrag van het hoofd-nek systeem in ongevalsituaties te bestuderen. Het model bestaat uit 10 starre lichamen voor het hoofd, de werfels C1 tot en met C7, T1 en de torso. De lichamen zijn verbonden door een lineair visco-elastisch model voor de intervertebrale gewrichten. Daarnaast zijn de belangrijkste nekspieren opgenomen.

Dit model is te valideren door middel van het bepalen van rotatiecentra van de verschillende werfels. Het rotatiecentrum is het virtuele draaipunt van een wervel ten opzichte van de onderliggende wervel. Het is mogelijk om hiermee zowel de verandering van de positie alsmede de oriëntatie van de bovenste wervel ten opzichte van de onderste te beschrijven in het sagittale vlak. Aangetoond is dat de momentane ligging van het rotatiecentrum slechts een geringe variatie kent gedurende de gehele flexie-extensie beweging van het hoofd en de nek. Dit geldt in ieder geval voor vrijwillige flexie-extensie bewegingen, het is niet zeker of het ook voor gedwongen bewegingen (bijvoorbeeld botsingen) zal gelden, het vermoeden is dat het wel geldig is. Als het mogelijk is om dit centrum te bepalen en daarmee aan te tonen dat het ongeveer op de zelfde positie (op de onderste wervel) ligt dan geeft dat een eerste indicatie of de beweging van een wervel ten opzichte van de lager gelegen wervel realistisch is.

Het bepalen van de rotatiecentra wordt gedaan aan de hand van resultaten van het wiskundige model (in MADYMO). Dit model levert de positie van de werfels ten opzichte van de onderliggende.

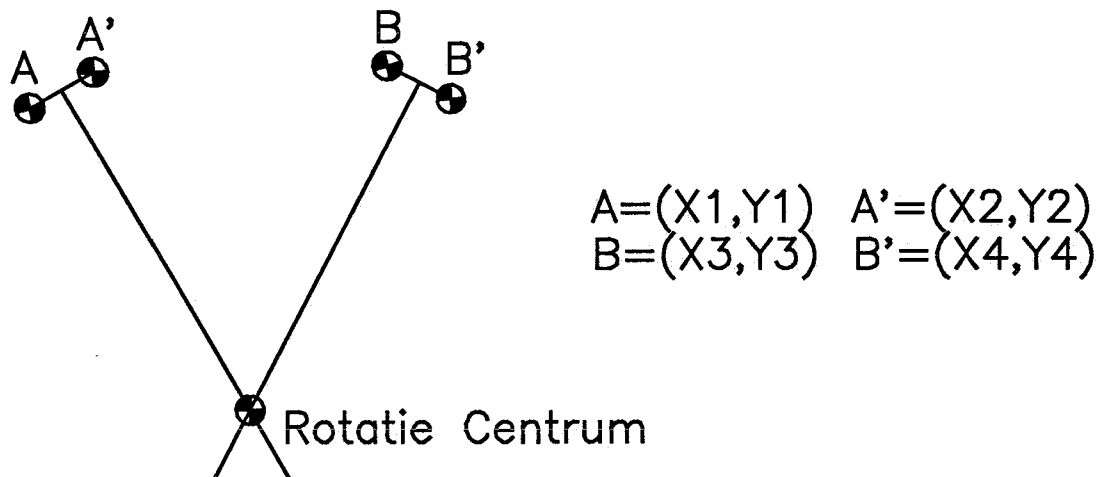
Het rotatiecentrum is te bepalen met behulp van een viertal verschillende methodes. In dit rapport worden deze vier methodes uitgelegd en wordt er een keuze gemaakt voor een van de methodes. Verder wordt de gekozen methode uitgewerkt in FORTRAN en zal er besproken worden hoe dit globaal in zijn werk gaat, welke problemen er optreden en wat de eisen voor de invoer en de uitvoer zijn. Daarna zal er een voorbeeld uitgewerkt worden aan de hand van het twopivot model, dat tevens gebruikt wordt om de software te testen. Het is hierna de bedoeling om het model te onderwerpen aan de software en te kijken of het ook daadwerkelijk voldoet aan de werkelijkheid. Het model wordt dan doorgerekend voor een aantal flexie-extensie bewegingen van het hoofd en de nek.

2. Methodes voor het bepalen van het rotatiecentrum

Een verplaatsing van een lichaam is altijd te beschrijven door een rotatie met een translatie. In het platte vlak is er altijd voor iedere verplaatsing (behalve zuivere translatie) één punt te vinden waarmee de verplaatsing als een enkele rotatie beschreven kan worden, dit is het rotatiecentrum. Voor het bepalen van het rotatiecentrum zijn er een viertal berekeningsmethodes bekeken en geëvalueerd.

2.1 Methode van Reuleaux

Dit is een grafische methode die als volgt werkt. Men definieert twee punten op het lichaam (A en B), zie fig 1. Van deze punten wordt de positie bepaald. Na rotatie bepaalt men de positie van de geroteerde punten (A' en B'). Het rotatiecentrum is het snijpunt van de middelloodlijnen van lijnstukken AA' en BB'. Deze methode stamt uit 1885 en wordt nog steeds gebruikt. Het is mogelijk om deze methode analytisch uit te werken zodat het met de uitvoer van MADYMO mogelijk is om de punten te berekenen. Dit is nauwkeuriger dan de grafische bepaling.



Afbeelding 1: Methode van Reuleaux

2.2 De rigid body methode

Deze methode is beschreven door Spiegelman en Woo [3]. Bij deze methode wordt wederom gebruik gemaakt van de punten A, B, A' en B'. Echter er wordt niet gebruik gemaakt van een grafische methode maar van een puur mathematische methode. Als een star lichaam een rotatie ondergaat om een punt, dan ondergaat ieder punt van het lichaam de transformatie

$$x' = Bx \quad (1)$$

met x de originele positie en x' de positie na rotatie en B de rotatie matrix

$$B = \begin{bmatrix} \cos(\phi) & \sin(\phi) \\ -\sin(\phi) & \cos(\phi) \end{bmatrix} \quad (2)$$

Als vervolgens de oorsprong van het coördinaten stelsel willekeurig verschoven wordt via vector D dan wordt

$$X = D + x \quad (3)$$

en

$$X' = D + x' \quad (4)$$

Indien vergelijking 3 en 4 worden gesubstitueerd in vergelijking 1 ontstaat:

$$X' - D = B (X - D) \quad (5)$$

Deze vergelijking beschrijft de rotatie van een lichaam in een willekeurig coördinaten stelsel. Door de positie van de punten $A=(x_1, y_1)$, $A'=(x_2, y_2)$, $B=(x_3, y_3)$ en $B'=(x_4, y_4)$ in te vullen (zie fig. 1) in vergelijking 5 en het elimineren van de vector D is het mogelijk om de rotatiematrix te bepalen door middel van.

$$\cos(\phi) = \frac{S' S + T' T}{S^2 + T^2} \quad (6)$$

$$\sin(\phi) = \frac{S' T - T' S}{S^2 + T^2} \quad (7)$$

Met in deze vergelijkingen:

$$S = X_1 - X_3 \quad (8)$$

$$S' = X_2 - X_4 \quad (9)$$

$$T = Y_1 - Y_3 \quad (10)$$

$$T' = Y_2 - Y_4 \quad (11)$$

Als deze rotatiematrix bekend is, is het mogelijk om met de positie van A en A' of B en B' de positie van het rotatiecentrum te bepalen.

$$C_x = X_1 + \frac{(Y_2 - U)}{\sin(\phi)} - \frac{\cos(\phi)(Y_1 - U)}{\sin(\phi)} \quad (12)$$

$$C_y = Y_1 - \frac{(X_2 - V)}{\sin(\phi)} + \frac{\cos(\phi)(X_1 - V)}{\sin(\phi)} \quad (13)$$

Waarin

$$U = \frac{\sin(\phi)(X_1 - X_2)}{2(1 - \cos(\phi))} + \frac{Y_1 + Y_2}{2} \quad (14)$$

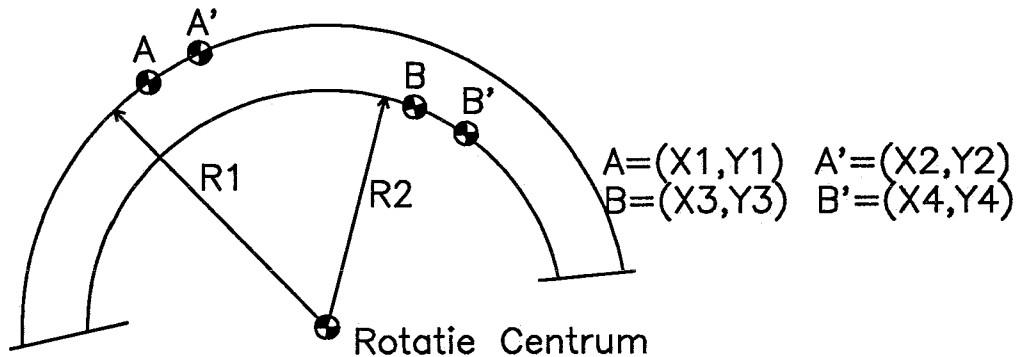
$$V = -\frac{\sin(\phi)(Y_1 - Y_2)}{2(1 - \cos(\phi))} + \frac{X_1 + X_2}{2} \quad (15)$$

Deze methode wordt uitgebreid beschreven en getest in een artikel van Spiegelman en Woo [3]. De conclusies die Spiegelman en Woo trekken worden vergeleken met die van de methode van Reuleaux zie daarvoor hoofdstuk 4.

In het artikel van Spiegelman en Woo is een fout geslopen. Er is een minteken bij de formule waarmee V berekend wordt weggelaten of vervallen (pag 717 [3], vierde formule, zie ook formule 15, in dit rapport).

2.3 Methode met twee cirkels.

Bij deze methode wordt wederom gebruikt gemaakt van de punten A en A' en B en B'. Indien er twee punten om een punt roteren is het mogelijk om, om het middelpunt een cirkel te tekenen door de punten A en A' en een cirkel door de punten B en B'.



Afbeelding 2: Methode met twee cirkels

Deze punten hebben hetzelfde middelpunt (zie fig. 2). Door de vergelijkingen van deze cirkels op te stellen en de punten A en A' in te vullen en de stralen aan elkaar gelijk te stellen krijgt men een vergelijking met twee onbekenden (de x- en y-coördinaat van het rotatiecentrum). Als hetzelfde gedaan wordt met de punten B en B' wordt een tweede vergelijking met dezelfde twee onbekenden verkregen. Hieruit volgt:

$$(x-C_x)^2 + (y-C_y)^2 = R^2 \quad (16)$$

Als hier de positie van A en A' ingevuld worden ontstaan er de volgende vergelijkingen

$$(X_1-C_x)^2 + (Y_1-C_y)^2 = R_1^2 \quad (17)$$

$$(X_2-C_x)^2 + (Y_2-C_y)^2 = R_1^2 \quad (18)$$

Als deze aan elkaar gelijk gesteld worden ontstaat er een vergelijking met twee onbekenden:

$$(X_1-C_x)^2 + (Y_1-C_y)^2 = (X_2-C_x)^2 + (Y_2-C_y)^2 \quad (19)$$

Wordt er vervolgens voor gezorgd dat de twee onbekenden, C_x en C_y , uit de kwadratische termen gehaald worden dan wordt de volgende vergelijking verkregen:

$$A - 2(X_1 - X_2)C_x = -2(Y_1 - Y_2)C_y \quad (20)$$

met daarin

$$A = X_1^2 + Y_1^2 - X_2^2 - Y_2^2 \quad (21)$$

Hiermee kan een vergelijking voor C_y als functie van C_x worden bepaald:

$$C_y = \frac{A - 2(X_1 - X_2)C_x}{-2(Y_2 - Y_1)} \quad (22)$$

Met de punten B en B' kan hetzelfde gedaan worden, waarna het mogelijk is om door substitutie C_x , C_y te bepalen:

$$C_x = \frac{B - \frac{Y_4 - Y_3}{X_2 - X_1} A}{2(X_3 - X_4) - \frac{2(Y_4 - Y_3)(X_1 - X_2)}{Y_2 - Y_1}} \quad (23)$$

Met daarin

$$B = X_3^2 + Y_3^2 - X_4^2 - Y_4^2 \quad (24)$$

Dit resultaat invullen in vergelijking 22 en het rotatiecentrum is bekend.

2.4 Methode van Bowyer en Woodwark

Bij deze methode wordt gebruik gemaakt van drie punten in de tijd. Hij is gebaseerd op het feit dat er altijd een cirkel is die door deze drie punten gaat, mits deze niet alle drie op een lijn liggen. Deze methode wordt voornamelijk gebruikt in de CAD-wereld om cirkels of cirkelbogen te kunnen construeren door drie punten en is niet geschikt voor het onderhanden liggende probleem. Voor het berekenen van het rotatiecentrum is de positie van een punt op drie tijdstippen t_1 , t_2 en t_3 nodig. Omdat de positie van het rotatiecentrum tussen tijdstip t_1 en t_2 kan verschillen van die tussen t_2 en t_3 kan deze methode afwijkingen ten gevolge kunnen hebben. In het boek van Bowyer en Woodwark [2] wordt slechts een heel summiere beschrijving gegeven van de berekeningen, de nu volgende vergelijkingen zijn daar rechtstreeks uit overgenomen.

$$X_{CC} = \frac{\begin{vmatrix} (X_{LK}^2 + Y_{LK}^2) & Y_{LK} \\ (X_{MK}^2 + Y_{MK}^2) & Y_{MK} \end{vmatrix}}{2 \begin{vmatrix} X_{LK} & Y_{LK} \\ X_{MK} & Y_{MK} \end{vmatrix}} \quad (25)$$

$$Y_{CC} = \frac{\begin{vmatrix} X_{LK} & (X_{LK}^2 + Y_{LK}^2) \\ X_{MK} & (X_{MK}^2 + Y_{MK}^2) \end{vmatrix}}{2 \begin{vmatrix} X_{LK} & Y_{LK} \\ X_{MK} & Y_{MK} \end{vmatrix}} \quad (26)$$

Met daarin

$$\begin{aligned} X_{LK} &= X_L - X_K, & X_{MK} &= X_M - X_K \\ Y_{LK} &= Y_L - Y_K, & Y_{MK} &= Y_M - Y_K \end{aligned} \quad (27)$$

Bij deze formules is ervan uitgegaan dat de punten K,L en M heten, CC is het middelpunt van de cirkel door K, L en M. De berekende positie van het rotatiecentrum is relatief ten opzichte van punt K, om de absolute positie te kunnen bepalen is het nodig om bij deze relatieve verplaatsing de absolute positie van punt K op te tellen.

3. Evaluatie van de verschillende methodes

Zoals al gezien bij het verklaren van de werking van de verschillende methodes is het bij voorbaat al mogelijk om de methode van Bowyer en Woodwark af te laten vallen.

Bij het beschouwen van de overige methodes is in eerste instantie gekeken naar het artikel van Spiegelman en Woo. Aan de hand van hun conclusies valt ook de methode van Reuleaux af. Spiegelman en Woo komen tot de volgende voordelen van de rigid body methode ten opzichte van de methode van Reuleaux:

- Markers kunnen geplaatst worden op de radiale as of aan beide zijden van het rotatiecentrum.
- De richting van een lijn door de markers ten opzichte van het rotatiecentrum heeft geen invloed op de nauwkeurigheid mits de afstanden niet gelijk zijn.
- De nauwkeurigheid van de berekening van de hoekverdraaiing wordt niet beïnvloed door de grootte van de hoekverdraaiing.
- Voorkennis van het rotatiecentrum is niet nodig.
- Een optimaal resultaat wordt bereikt als de markers zover mogelijk uit elkaar liggen.

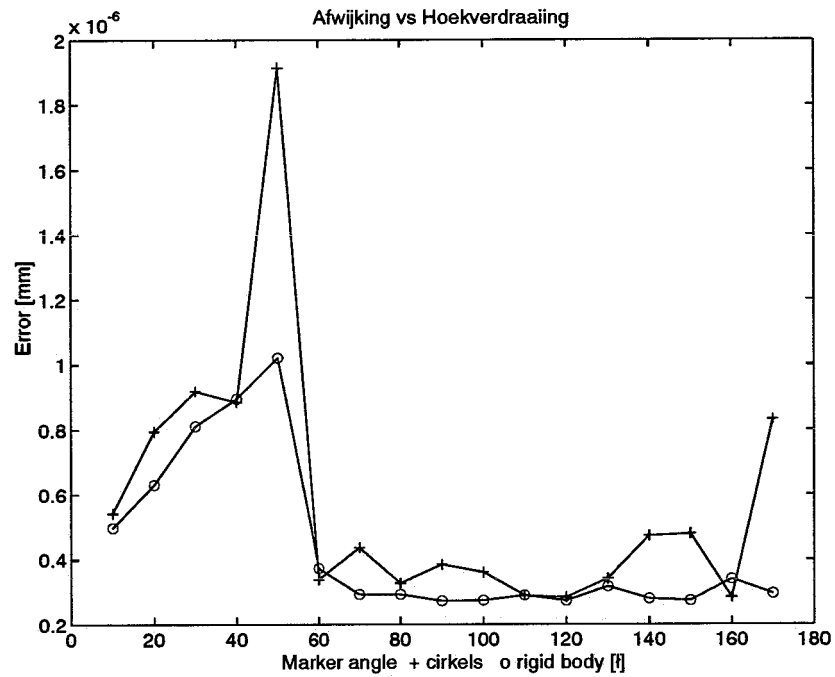
Na deze keuzes blijven alleen de rigid-body methode en de methode met twee cirkels over. Deze zijn op een soortgelijke manier vergeleken als de methode van Reuleaux en de rigid body methode met elkaar vergeleken zijn door Spiegelman en Woo, namelijk door middel van een numerieke uitwerking van een testprobleem.

Met behulp van een CAD-pakket (AUTOCAD) zijn de posities van punten A en B bepaald. Deze punten lagen op een cirkel met een straal van 30 mm onder een onderlinge hoek van 10 tot 180 graden (marker-angle) met tussen-stappen van 10 graden. Vervolgens zijn deze punten geroteerd om het middelpunt van de cirkel over een hoek van 1 graad. Daarmee ontstaan er 18 sets van coördinaten waarmee het middelpunt berekend kan worden. Vervolgens is er van deze positie de bekende positie afgetrokken en is er een afwijking geconstateerd. Deze afwijkingen in x- en y-coördinaat zijn vervolgens gekwadrateerd en opgeteld en daarna is daar de wortel van getrokken. De resultaten hiervan zijn te vinden in fig. 3. In fig. 4 zijn de fouten van beide methodes van elkaar afgetrokken het resultaat van deze aftrekking geeft een indicatie over welke methode de kleinste afwijking geeft. Hetzelfde is gedaan voor de hoekverdraaiing. De resultaten daarvan zijn te vinden in fig. 5 en 6. Bij zowel fig. 3, 4, 5 als 6 zijn de waarden bij een hoek van 180 graden weggelaten. Dit komt omdat de methode met de twee cirkels daar een zo grote afwijking gaf dat de rest van de waarden niet meer te onderscheiden zijn.

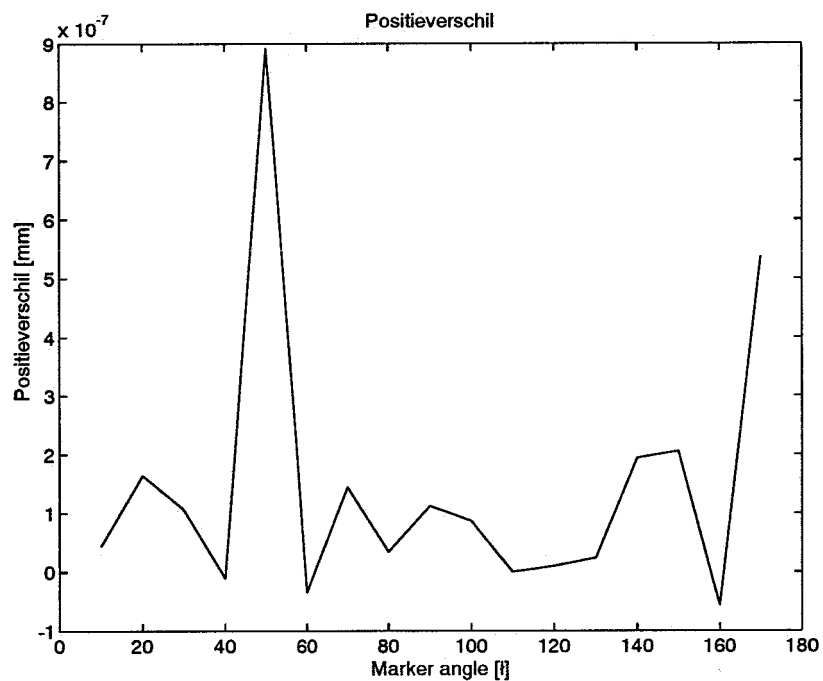
Een deel van de fouten wordt geïntroduceerd door de eindige precisie waarmee de computerpakketten werken. Deze precisie is 8 decimalen, bij de uitvoer van madymo wordt gewerkt met een precisie van 6 decimalen. Indien de waarden van Autocad afgerond worden naar 6 decimalen blijkt dat de nauwkeurigheid van beide methoden omlaag gaat. Dan blijft de nauwkeurigheid van de rigid body methode over een groter gebied groter.

4. Conclusie ten aanzien van de berekeningsmethodes

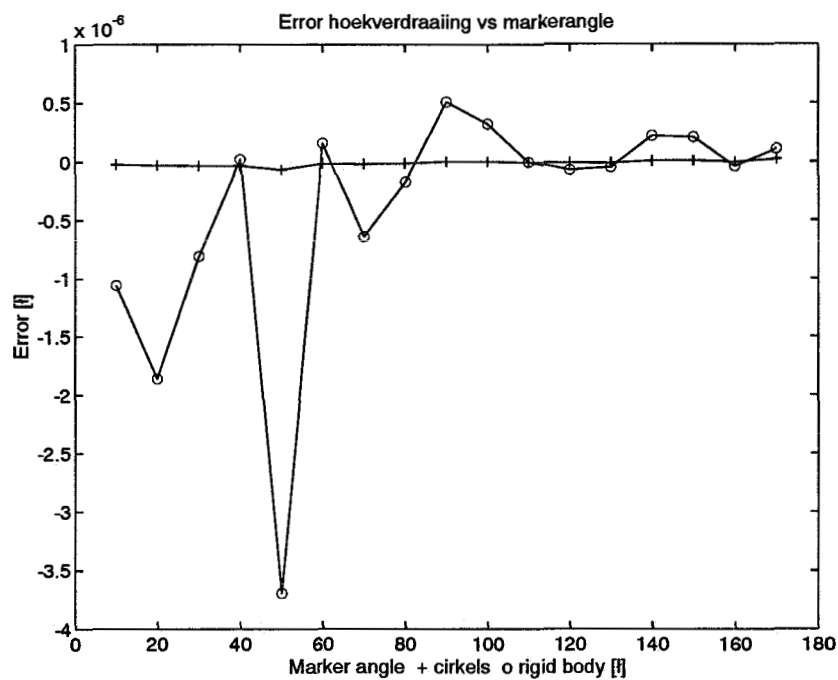
Op basis van de berekeningen blijkt dat de rigid body methode over het algemeen nauwkeuriger is dan de methode met de twee cirkels wat betreft de positie van het rotatie centrum. Wat betreft de rotatiehoek is de methode met de twee cirkels veel nauwkeuriger. Voor het bepalen van de rotatie centra is de positie van groter belang dan de rotatiehoek en daarom ligt de keuze van de rigid body methode voor de hand. Helaas blijkt uit de berekeningen niet wat nu de optimale hoek tussen de punten A en B ten opzichte van het rotatiecentrum is. Echter uit het artikel van Spiegelman en Woo blijkt dat voor de foutgevoeligheid deze hoek te liggen bij 180 graden. Bij de simulatie van de nek zal dan ook getracht moeten worden om deze hoek zo dicht mogelijk bij 180 graden neer te leggen. Het is eventueel ook mogelijk om na het bepalen van het rotatiecentrum de punten A en B te verleggen zodat deze 180 graden wordt en vervolgens de simulatie opnieuw uit te voeren om dan de beste resultaten te verkrijgen. Echter in alle gevallen is de fout erg klein (10^{-6} mm).



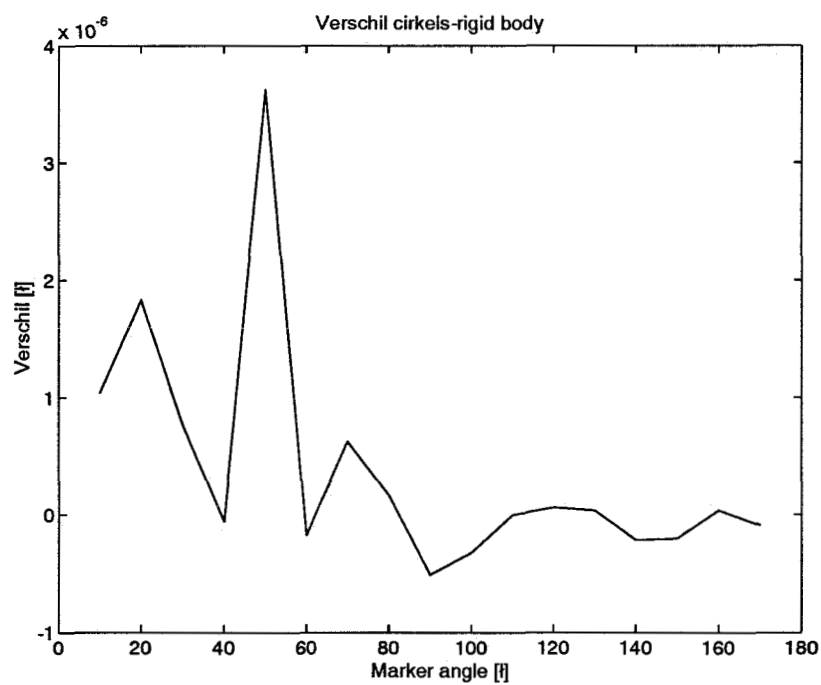
Afbeelding 3: Afwijkingen in positie van het rotatiecentrum



Afbeelding 4: Verschil in afwijking tussen rigid body en cirkelmethode



Afbeelding 5: Hoekverdraaiing van het lichaam



Afbeelding 6: Verschil tussen afwijkingen cirkelmethode en rigid body

5. Werking van het programma

5.1. Uitvoering van de rigid body methode in FORTRAN

Nu de keuze voor de berekeningsmethode vast ligt is het nodig om te gaan kijken hoe deze gerealiseerd wordt en wat daar allemaal bij komt kijken. Voor de uitwerking van de rigid body methode is FORTRAN gekozen, omdat deze programmeertaal een aantal goede eigenschappen heeft. De voornaamste is dat hij voor veel mensen eenvoudig te begrijpen is en in de wetenschappelijke wereld veel gebruikt wordt. Een andere is de eenvoudige manier om formules te vertalen in software.

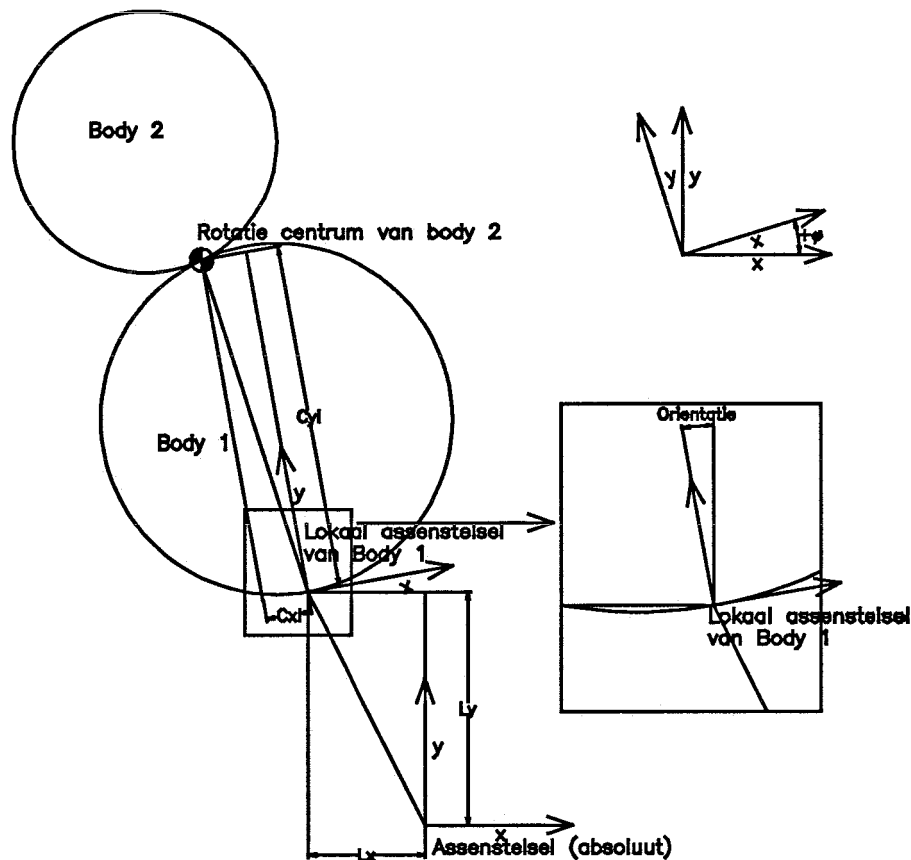
De rigid body methode zoals deze gebruikt gaat worden berekent een rotatiecentrum ten opzichte van de onderliggende wervel, om op deze wijze het wiskundige model te valideren. De berekende positie is dus geen absolute positie maar een relatieve positie op de onderliggende wervel. Om de resultaten grafisch weer te geven in het nek-model binnen MADYMO, is echter een absolute positie noodzakelijk. Nu is het juist deze grafische weergave die het makkelijk maakt om de resultaten te bekijken en te controleren of het rotatie centrum ook werkelijk op de onderliggende wervel ligt. De relatieve posities zullen dus omgerekend moeten worden naar absolute posities.

Voor de grafische weergave wordt gebruik gemaakt van de postprocessor MAPPK. Het woord postprocessor wil in feite niets anders zeggen dan dat de berekende informatie uit MADYMO gebruikt wordt in een ander afzonderlijk programma (MAPPK). Dit houdt in dat het mogelijk is om de berekende data te verwerken voor dat deze zichtbaar gemaakt wordt. Bij het programmeren van de software om de rotatiecentra te berekenen is er een groot voordeel bij de postprocessor, deze kan namelijk twee verschillende grafische uitvoer files gelijktijdig weergeven. Wat op zich weer betekent dat de oorspronkelijke file niet aangepast hoeft te worden en dus ook apart gebruikt kan worden.

MADYMO maakt een aparte file aan voor de grafische weergave, de zogenaamde KIN3 file. Voor de weergave van de rotatiecentra zal een soortgelijke KIN3 file gemaakt moeten worden. Het is dus taak van het programma om zo'n file te maken.

5.2. Relatieve en absolute positie

Nu moet er nog bekeken worden hoe de relatieve positie van de rotatiecentra berekend kunnen worden met behulp van MADYMO en hoe de absolute positie en de oriëntatie van de onderliggende wervel bepaald kunnen worden.



Afbeelding 7: Omrekening van relatieve naar absolute positie

In fig. 7 is in getekende vorm te zien hoe uitgaande van de relatieve positie de absolute positie berekend kan worden. Hierin is $C_1 = (C_{x1}, C_{y1})$ de relatieve positie van het rotatiecentrum in het lokale assenstelsel van het onderliggende lichaam. Oriëntatie is de oriëntatie en $L = (L_x, L_y)$ is absolute positie van het coördinatenstelsel van het onderliggende lichaam. De omrekening van relatief naar absoluut gebeurt als volgt

$$C = BC_L + L \quad (28)$$

Waarin B de rotatiematrix is in het bijbehorende vlak, C is de absolute positie.

$$B_X = \begin{bmatrix} -\cos & -\sin \\ -\sin & \cos \end{bmatrix}, \quad B_Y = \begin{bmatrix} -\cos & \sin \\ \sin & \cos \end{bmatrix}, \quad B_Z = \begin{bmatrix} -\cos & -\sin \\ -\sin & \cos \end{bmatrix} \quad (29)$$

Bepalen relatieve positie rotatiecentrum

Daartoe beschikt MADYMO over de mogelijkheid om posities van punten op lichamen naar een file weg te schrijven. Binnen MADYMO is het mogelijk om door middel van een RELDIS commando in de datafile een file aan te laten maken met daarin de relatieve posities van punten op lichamen ten opzichte van andere willekeurige punten. Voor het berekenen van het rotatiecentra zijn twee punten per lichaam nodig. Deze punten worden op het gewenste lichaam gedefinieerd en weggeschreven ten opzichte van het nulpunt in het lokale assenstelsel van het onderliggende lichaam. Het resultaat hiervan is een file met de extensie .RDS, in het vervolg zal deze file de RELDIS file genoemd worden.

Bepalen absolute positie van het lokale assenstelsel

Voor het bepalen van de absolute positie van de lokale assenstelsels wordt een soortgelijke functie LINDIS gebruikt. Deze functie werkt exact hetzelfde als de RELDIS functie maar hier wordt LINDIS gebruikt omdat er dan een aparte file gecreëerd wordt. Om de absolute positie te kunnen wegschrijven wordt nu het punt (0,0,0) in het lokale assenstelsel van het onderliggende lichaam bepaald ten opzichte van het absolute nulpunt. Bij de LINDIS functie wordt er een file aangemaakt met de extensie .LDS deze zal in het vervolg LINDIS genoemd worden.

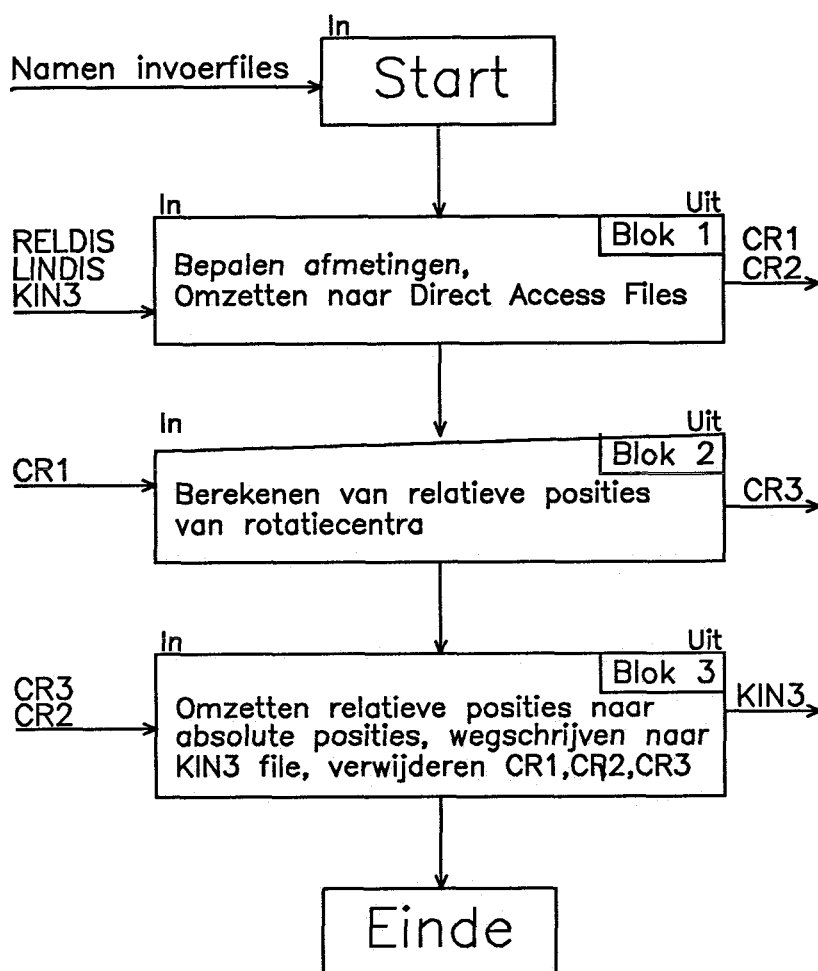
Bepalen oriëntatie van het lokale assenstelsel

De oriëntatie van de assenstelsels is iets ingewikkelder, weliswaar kan MADYMO deze zelf wegschrijven maar dat kan alleen als er gebruikt gemaakt wordt van scharnieren, in het nekmodel wordt er geen gebruik gemaakt van scharnieren. De oriëntatie van de lichamen zal uit de KIN3 file gehaald moeten worden. Deze oriëntaties zijn beschreven met behulp van eulerparameters. Het is hier niet het doel om deze euler parameters uitvoerig te gaan beschrijven, maar voor het gebruik in dit geval (enkel het sagittale vlak) is dit vrij eenvoudig. In het sagittale vlak is de rotatie altijd een rotatie om de y-as. In de uitvoer van MADYMO is dan de waarde van de eerste eulerparameter de hoekverdraaiing. De richting van deze draaiing wordt gehaald uit de derde eulerparameter, welke 1 of -1 is. Dit betekent voor de oriëntatie van het lokale assenstelsel dat het voldoende is om deze te roteren om de y-as.

In de handleiding van MADYMO is een uitgebreide beschrijving van de gebruikte commando's te vinden

5.3. Globale werking van het programma

Het programma is te verdelen in een drietal blokken.



Afbeelding 8: Globale opbouw van het programma

- Blok 1:** in dit gedeelte van het programma wordt bepaald om hoeveel lichamen het gaat en om hoeveel tijdstippen het gaat, dit is nodig om de grootte van de records in de direct access files te bepalen. Vervolgens worden de file omgezet in de direct access files zoals besproken wordt in par 5.3.2
- Blok 2:** in dit gedeelte worden de punten daadwerkelijk berekend en weer opgeslagen in een andere direct access file. De oorspronkelijke direct access file met de RELDIS gegevens wordt hierbij weggegooid.
- Blok 3:** in dit gedeelte worden de relatieve posities van de rotatiecentra omgerekend naar de absolute posities. Vervolgens worden de posities uitgeschreven in de vorm van een KIN3 file. Hierna worden alle direct access files weggegooid.
- In bijlage 1 wordt de complete listing van het programma gegeven.

5.3.1. Invoer/uitvoer van CR

Aan de invoer van het programma worden een aantal eisen gesteld

Zoals in de voorgaande paragrafen vermeld is zijn er een drietal files nodig. Voor de relatieve verplaatsing van de lichamen wordt er een RELDIS gemaakt. In deze file staan de verplaatsingen van de punten op de lichamen vanaf het laagste niet starre lichaam naar het laatste gewenste lichaam. De volgorde van de twee punten per lichaam maken niet uit. Voorbeeld: er zijn 3 lichamen (twopivot model) genummerd 1, 2, 3, waarvan lichaam 1 star is, lichaam 2 scharnierend verbonden is met 1 en 3 scharnierend met 2. De punten op de lichamen 2 en 3 worden dan A2, B2, A3, B3 genoemd, dan ziet de RELDIS file er als volgt uit (afgezien van de header):

```
Time 0
pos. A2      pos. = positie van A of B ten opzichte van het assenstelsel van het lager
pos. B2      gelegen lichaam
pos. A3
pos. B3
Time 1
etc.
```

Voor de absolute verplaatsingen van de lokale assenstelsel moeten de absolute verplaatsingen van de onderliggende lichamen worden bepaald. In het voorbeeld bij de RELDIS file houdt dit in dat de absolute positie van de lichamen 1 en 2 bepaald moeten worden. De LINDIS file komt er dan als volgt uit te zien (afgezien van de header):

```
Time 0
pos lichaam 1      pos. = positie van oorsprong van het assenstelsel van het lichaam
pos lichaam 2      ten opzichte van het referentieassenstelsel
Time 1
etc.
```

Het is voor de werking van het programma noodzakelijk dat er niet minder lichamen in de LINDIS file (meer mag wel) staan dan het aantal gevraagde rotatiecentra.

Voor het bepalen van de oriëntatie van de lichamen is de KIN3 file nodig. De KIN3 file van het hoofd-nek model is opgebouwd uit een tweetal blokken:

Header : Dit blok bestaat uit gegevens omtrent het aantal lichamen, het aantal Kelvin elementen (hier gebruikt voor de modellering van spieren) en nog een aantal kenmerken meer die in dit geval niet van belang zijn. Daarna komen er een aantal regels (afhankelijk van het aantal lichamen) die de grootte en de vorm van de lichamen beschrijven. Verder staat hier ook de naam van het model, en de run in.

Data : Dit blok bevat het tijdstip waarop de onderstaande data geldig is. Dit zijn de posities en de oriëntaties van de lichamen, hierna komt de positie van de kelvin elementen. Dit blok wordt voor het aantal berekende tijdstippen herhaald.

Het programma moet zelf ook zo'n file maken met daarin de posities van de rotatiecentra. Deze file is veel eenvoudiger. De header bevat nu alleen de naam en de run, het aantal lichamen en de grootte en vorm van de lichamen. Het data blok bevat alleen het tijdstip en de posities van de lichamen (dat wil zeggen de rotatiecentra), de oriëntaties van deze lichamen doet er niet toe.

Voor de KIN3 file is er slechts een eis, deze houdt in dat er alleen de posities en de oriëntaties van de lichamen in mogen staan bij het juiste lichaam en dat er geen andere informatie in staat zoals bijvoorbeeld versnellingsopnemers, markers e.d. Indien er meer lichamen in de file staan als dat er rotatiecentra bepaald moeten worden, worden deze gewoon overgeslagen, net zoals het gebeurt met de kelvin elementen.

Voor alle drie de files geldt dat de tijdstippen gelijk moeten zijn. Is dit niet het geval dan zal het programma hierop een foutmelding genereren. Het programma gaat wel verder maar de uitvoer is naar alle waarschijnlijkheid niet bruikbaar. Als laatste is er nog een probleem dat veroorzaakt wordt door UNIX, er moet voor gezorgd worden dat alle files een naam hebben met een lengte van acht karakters (zonder extensie), onder DOS is dit geen probleem.

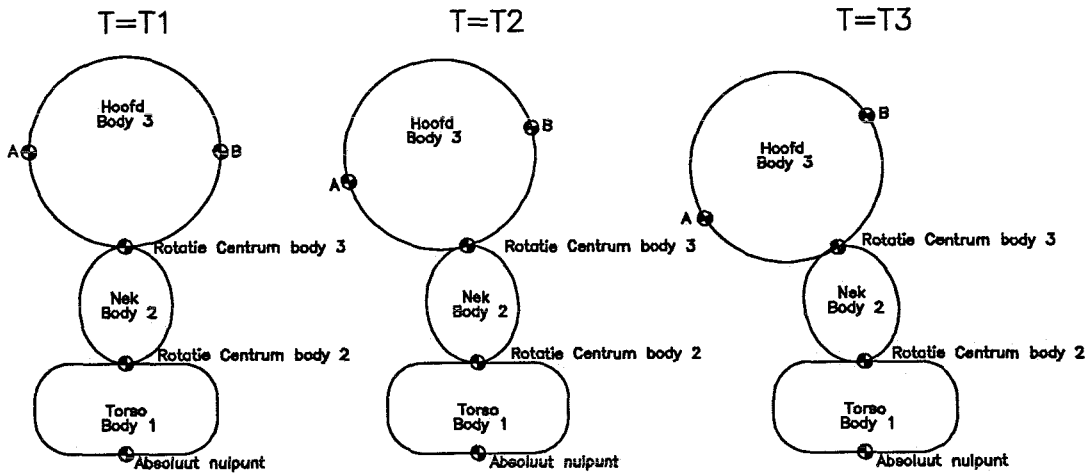
In hoofdstuk 6 wordt een voorbeeld gegeven aan de hand van een twopivot model. In bijlage 2 wordt een handleiding gegeven hoe de datafiles voor MADYMO aangepast moeten worden, daarmee worden deze eisen verduidelijkt.

De uitvoer van het programma is een KIN3 file. In deze KIN3 file zijn alleen de positie van de rotatiecentra opgenomen. Om deze te zien in relatie tot het model is het nodig om beide (de oorspronkelijke KIN3 file en de nieuwe KIN3 file) gelijktijdig door MAPPK te laten weergeven. Omdat er op het eerste tijdstip geen sprake is van een rotatie geeft het programma op dit tijdstip geen uitvoer. Hierdoor moet in de oorspronkelijke KIN3 file de gegevens van het eerste tijdstip worden weggegooid, wordt dit niet gedaan dan zal MAPPK hierdoor een reeks foutmeldingen genereren, MAPPK zal de resultaten wel weergeven.

5.3.2. Gebruik van direct access files

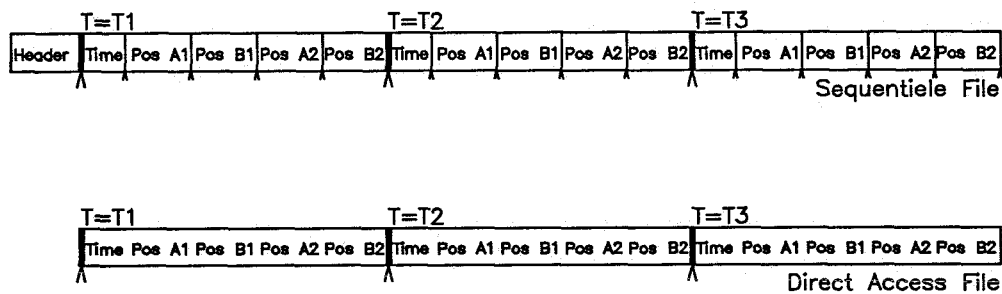
Voor het bepalen van het relatieve rotatiecentrum zijn de gegevens nodig van de posities (relatief ten opzichte van de onderste wervel) op twee tijdstippen. Aan de hand van fig. 8 is te zien dat wanneer het rotatiecentrum op $t=t_2$ berekend moet worden de posities van de

punten op $t=t_1$ en $t=t_2$ nodig zijn. Voor het berekenen van het rotatiecentrum op $t=t_3$ zijn de posities op $t=t_2$ en $t=t_3$ nodig.



Afbeelding 9: Verschillende tijdstippen nodig om het rotatiecentrum te berekenen

Tevens blijkt uit het voorgaande dat er een drietal files nodig zijn om het rotatiecentrum te bepalen. MADYMO schrijft zijn gegevens weg in sequentiële files weg. Deze files zijn gewoon leesbaar en hebben dus het voordeel dat ze direct gecontroleerd kunnen worden op extreme afwijkingen en dergelijke. Het nadeel hiervan blijkt uit het feit dat er voor de berekening van het rotatiecentrum de posities op twee tijdstippen nodig zijn. Dit betekent dat er in de file heen en weer gegaan moet worden tussen verschillende tijdstippen. Omdat dit voor en achteruit gaan in sequentiële files lastig is om dat er veel mogelijke posities zijn is er voor gekozen om een ander type file te maken de zogenaamde direct access file.



Afbeelding 10: Verschil tussen sequentiële en direct access files

In fig. 10 zijn de mogelijke posities in de verschillende soorten files aangegeven met pijlen. Bij de sequentiële file zijn dit er zeer veel, bij de direct access file zijn dit er slechts enkele, alleen die die aan het begin van een record staan.

Tevens is in fig. 10 aangegeven welke posities wel (de grote pijlen) en welke er niet (de kleine pijlen) gebruikt worden. De direct access file bestaat uit een aantal records welke allen een nummer toegewezen hebben gekregen.

Door nu bij het uitlezen het juiste nummer op te geven kan er in een keer de juiste gegevens worden ingelezen en hoeft er niet gespoeld te worden in de file. Het nadeel van dit spoelen is dat er exact bekend moet zijn hoeveel er gespoeld moet worden om op het juiste tijdstip uit te komen.

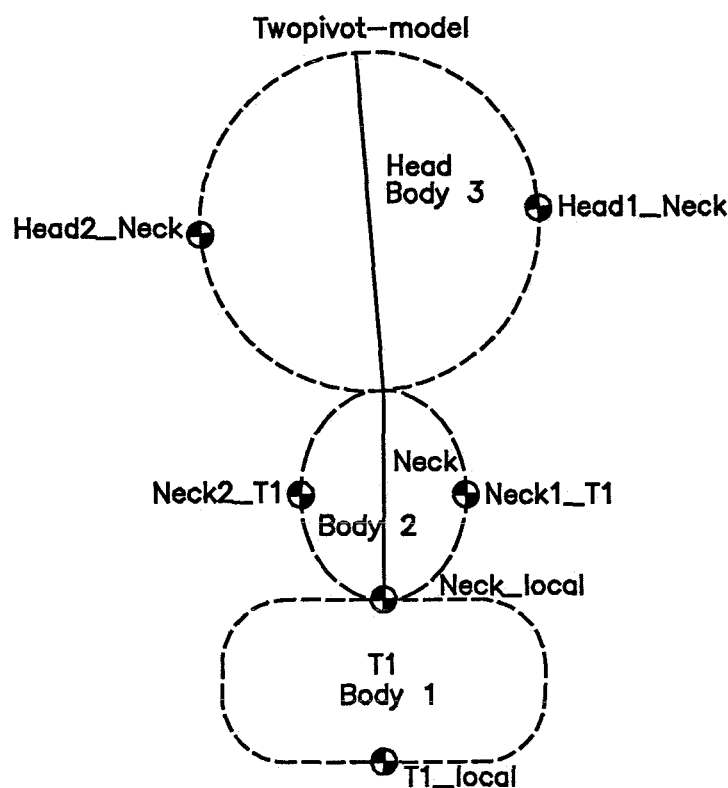
Het omzetten van de file moet wel gebeuren, wat weer een nadeel is. Bij het omzetten van de verschillende files wordt er wel meteen rekening gehouden met wat er nodig is en wat er niet nodig is. Zo wordt er van de RELDIS file alleen het tijdstip overgenomen en worden alleen die coördinaten meegenomen die ook daadwerkelijk gebruikt worden. De RELDIS en de KIN3 file worden gecombineerd tot een file met daarin alleen de absolute positie en de oriëntatie.

6. Testen van het programma

Het is noodzakelijk om het programma aan een aantal tests te onderwerpen, gekozen is om het programma te testen met een drietal verschillende modellen, twee daarvan beschrijven de beweging van het hoofd en de nek, het twopivot model en het nek model. Als extra is er nog een derde model gebruikt dat gebaseerd is op het twopivotmodel maar dat uit vijf scharnieren bestaat.

6.1. Resultaten met twopivotmodel

De werking van het programma en hoe het toegepast moet worden is het beste uit te leggen aan de hand van een eenvoudig en begrijpelijk systeem: het twopivot model. Het is uit observaties gebleken dat de bewegingen van het hoofd beschreven kunnen worden met behulp van een stangenmechanisme, het zogenaamde twopivot model. Dit model bestaat uit een tweetal stangen die scharnierend aan elkaar verbonden zijn (zie fig. 11).

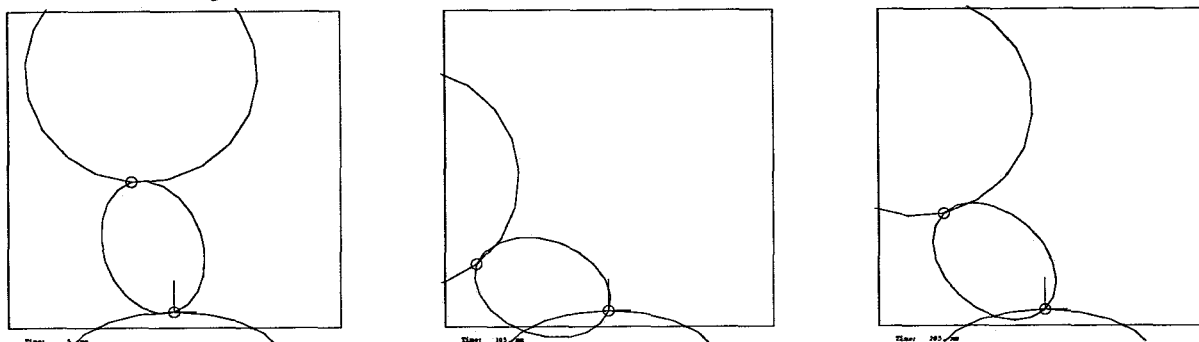


Afbeelding 11: Twopivot model

In MADYMO is het mogelijk om dit model te gebruiken en in plaats van de stangen gebruik te maken van lichamen (in de figuur aangegeven met streeplijnen). Deze lichamen worden dan met behulp van scharnieren aan elkaar verbonden en daarmee is het twopivot

model om de bewegingen te beschrijven klaar. Echter om een enigszins realistisch beeld te kunnen krijgen is het nodig om aan de scharnieren een bepaalde veerijfheid en demping te geven. In bijlage 2 is een beschrijving te vinden hoe de madymo invoer file gemaakt moet worden.

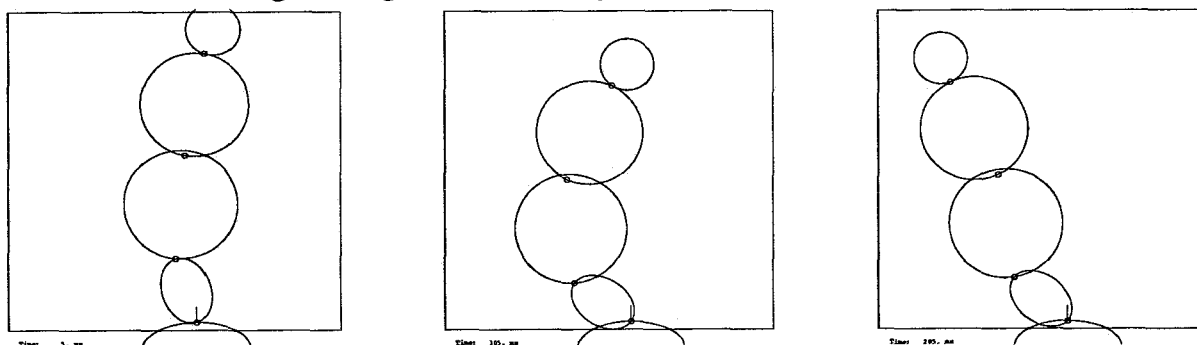
Met behulp van dit model zijn de rotatiecentra bepaald, de resultaten worden hier grafisch gepresenteerd. In fig. 12 zijn de resultaten te vinden op drie verschillende tijdstippen. In deze figuur zijn de rotatiecentra aangegeven door de kleine cirkels, de grote cirkels vormen de lichamen. Uit de figuren blijkt dat het programma voor vaste (bekende) scharnieren het juiste rotatiecentrum berekent.



Afbeelding 12: Resultaten twopivot model op tijdstippen 5, 105 en 205 ms

6.2. Resultaten met fivepivotmodel

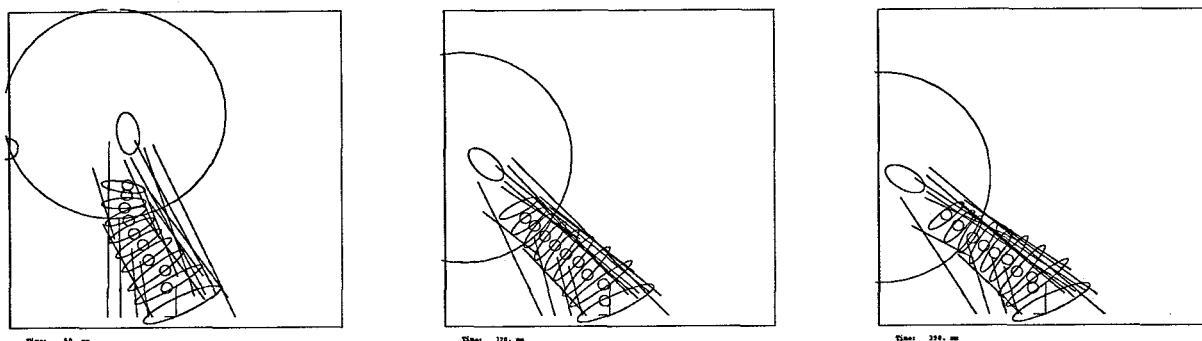
Om te testen of het programma ook goed werkt bij meerdere lichamen is er een five pivot model gemaakt wat geen relatie met de werkelijkheid heeft, maar slechts gebruikt wordt om het programma te testen. In de volgende resultaten, fig. 13, is voor ieder lichaam het rotatiecentrum berekend, tijdens het testen is ook de van mogelijkheid om minder rotatiecentra te berekenen gebruik gemaakt. Ook bij deze testen blijkt het resultaat te kloppen.



Afbeelding 13: Resultaten fivepivot model op tijdstippen 5, 105 en 205 ms

6.3. Resultaten met nekmodel

Het doel van het programma is het bepalen van de rotatiecentra van een realistisch en gedetailleerd model van de nek. Het model is met de nodige aanpassingen doorgerekent en met de resultaten van de berekening zijn de rotatiecentra bepaald. De hier gepresenteerde figuren (fig. 14) zijn slechts bedoeld ter illustratie van het resultaat en zijn niet bedoeld om het model te valideren.



Afbeelding 14: Resultaten nekmodel op tijdstippen 50, 120 en 190 ms

Uit deze figuren valt te concluderen dat het programma ook werkt met vrij gedefinieerde scharnieren.

7. conclusies

Het doel van het programma is het bepalen van de positie van het rotatiecentrum van nekwerfels tijdens het bewegen van de nek, om met behulp van deze rotatiecentra het nekmodel te valideren.

Voor het bepalen van de rotatiecentra zijn er een viertal verschillende methodes mogelijk. Deze methodes zijn met elkaar vergeleken en aan de hand daarvan is er een keuze gemaakt welke er gebruikt gaat worden. De rigid body methode is daarbij gekozen als methode om de rotatiecentra te bepalen.

Deze methode is vervolgens geïmplementeerd in fortran en getest met behulp van een aantal verschillende modellen. Bij het gebruik van modellen als het twopivot model waarbij de rotatiecentra op een bekende plaats liggen, blijkt dat het programma de juiste waarden berekend voor de rotatiecentra. Het nekmodel is ook gebruikt om te kijken of het programma ook de juiste waarden berekent als de rotatiecentra vooraf niet bekend zijn. Het resultaat hiervan is betrouwbaar, dit wil zeggen dat een visuele controle van de resultaten aangeeft dat de rotatiecentra goed berekend worden.

Literatuurlijst

1. Bowyer A., Woodwark J.K. (1983), A programmer's geometry, London, Butler and tanner, Ltd.
2. Jager de M., Sauren A., Thunnissen J., Wismans J., (1994), A three-dimensional head-neck model: validation for frontal and lateral impacts., Draft manuscript for the 38th Stapp Car Crash Conference 1994.
3. Spiegelman J.J., Woo S.L.Y. (1987), A rigid-body method for finding centers of rotation and angular displacements of planar joint motion, Journal of biomechanics, 20, 715-721.
4. Strang G. (1988), Linear algebra and its applications, San Diego, Harcourt Brace Jovanovich.
5. Wismans J.S.H.M. (1993), Injury biomechanics, Eindhoven, Technische Universiteit Eindhoven

```

PROGRAM CR
CHARACTER RDFILE*12,KNFILE*12,LDFILE*12,OUTFIL*12,INFILE*8
CHARACTER MODEL1*80,MODEL2*80,VRAAG,OUTCR*12
INTEGER SYSTEM,NUMMAR,NUMCR
LOGICAL BEST
REAL X(100),Y(100),Z(100),X1(100),Y1(100),KAPPA(100)
REAL X2(100),Y2(100),CX(100),CY(100),LX(100),LY(100),LZ(100)

PRINT *, 'Input files, without extension           : '
READ(*, '(A8)') INFILE
WRITE (RDFILE, '(2A)') INFILE, '.rds'
INQUIRE (FILE=RDFILE, EXIST=BEST)
IF (.NOT.BEST) THEN
  PRINT *, RDFILE, ' does not exist'
  GOTO 15
ENDIF
WRITE (LDFILE, '(2A)') INFILE, '.lds'
INQUIRE (FILE=LDFILE, EXIST=BEST)
IF (.NOT.BEST) THEN
  PRINT *, LDFILE, ' does not exist'
  GOTO 15
ENDIF
WRITE (KNFILE, '(2A)') INFILE, '.kn3'
INQUIRE (FILE=KNFILE, EXIST=BEST)
IF (.NOT.BEST) THEN
  PRINT *, KNFILE, ' does not exist'
  GOTO 15
ENDIF
10 PRINT *, 'Outputfilename, without extension       : '
READ(*, '(A8)') INFILE
WRITE (OUTFIL, '(2A)') INFILE, '.kn3'
IF (OUTFIL.EQ.KNFILE) THEN
  PRINT *, 'You cannot use ', KNFILE, ' choose another : '
  GOTO 10
ENDIF
INQUIRE (FILE=OUTFIL, EXIST=BEST)
IF (BEST) THEN
11 PRINT *, OUTFIL, ' already exists, overwrite ? (y/n) : '
  READ (*, '(A1)') VRAAG
  IF ((VRAAG.EQ.'n').OR.(VRAAG.EQ.'N')) THEN
    PRINT *, 'Which output file ?, without extension : '
    GOTO 10
  ELSEIF (VRAAG.NE.'y') THEN
    PRINT *, 'y(es) or n(o) please : '
    GOTO 11
  ENDIF
ENDIF
WRITE (OUTCR, '(2A)') INFILE, '.cr'
PRINT *, 'Which coordinate system: 1=XY, 2=YZ, 3=XZ : '
READ(*, '(I1)') SYSTEM
PRINT *, 'How many rotationcenters should be calculated : '
READ(*, '(I3)') NUMCR
PRINT *
CALL RDHEAD(RDFILE, NUMMAR, NUMREC, MODEL1, MODEL2, NUMCR)
CALL CONVER(RDFILE, LDFILE, KNFILE, X, Y, Z, NUMMAR, NUMREC, SYSTEM,
+ LX, LY, LZ, KAPPA, NUMCR)
CALL CALCUL(X1, Y1, X2, Y2, CX, CY, NUMCR, NUMREC)
CALL WRDATA(OUTFIL, CX, CY, NUMCR, NUMREC, SYSTEM, MODEL1, MODEL2,
+ LX, LY, KAPPA, OUTCR)
15 CONTINUE
END

```

```

SUBROUTINE RDHEAD(RDFILE, NUMMAR, NUMREC, MODEL1, MODEL2, NUMCR)
CHARACTER RDFILE*12
CHARACTER MODEL1*80
CHARACTER MODEL2*80
INTEGER NUMMAR, NUMREC, NUMDAT
CHARACTER IDNAME*8
CHARACTER DUMMY2*80
REAL DUMMY

NUMREC=0
PRINT '(A,A12)', 'Reading header from file      : ', RDFILE
OPEN(UNIT=10, FILE=RDFILE, STATUS='OLD')
REWIND(10)
READ(10, '(A80)') MODEL1
PRINT '(A,A)', 'Model                          : ', MODEL1
READ(10, '(A80)') MODEL2
PRINT '(A,A)', 'Description                    : ', MODEL2
READ(10, '(I4,I4)') NUMMAR, NUMDAT
PRINT '(A,I4)', ' Number of points            : ', NUMMAR
IF (NUMCR.GT.NUMMAR/2) THEN
  NUMCR=NUMMAR/2
ENDIF
PRINT '(A,I4)', ' Number of rotationcenters : ', NUMCR
PRINT '(A,I4)', ' Number of data per point  : ', NUMDAT
PRINT *, 'Identifiers of points '
DO 5, J=1, NUMMAR, 1
  READ (10, '(A)') IDNAME
  PRINT *, IDNAME
5  CONTINUE
  READ (10, '(A)') DUMMY2, DUMMY2, DUMMY2, DUMMY2
10  READ (10, '(E14.6E2)', END=30) TIME
  DO 20, J=1, NUMMAR, 1
  READ (10, '(E14.6E2, E14.6E2, E14.6E2, E14.6E2)', END=30) DUMMY,
+    DUMMY, DUMMY, DUMMY
20  CONTINUE
  NUMREC=NUMREC+1
  GOTO 10
30  PRINT '(A,I4)', ' Number of records        : ', NUMREC
  CLOSE(10, STATUS='KEEP')
  END

SUBROUTINE CONVER(RDFILE, LDFILE, KNFILE, X, Y, Z, NUMMAR, NUMREC,
+  SYSTEM, LX, LY, LZ, KAPPA, NUMCR)
C
CHARACTER RDFILE*12, LDFILE*12, KNFILE*12, DUMMY2*80
INTEGER NUMMAR, DUMMY, NUMREC, SYSTEM, NUMBOD, NUMLOC
REAL X(NUMCR*2), Y(NUMCR*2), Z(NUMCR*2), TIME, DUMMY3, KAPPA(NUMCR)
REAL LX(NUMCR), LY(NUMCR), LZ(NUMCR), UX, UY, UZ, HELP
C
OPEN(UNIT=10, FILE=RDFILE, STATUS='OLD')
OPEN(UNIT=11, FILE='CR1.TMP', ACCESS='DIRECT', RECL=NUMCR*16+8)
OPEN(UNIT=12, FILE=LDFILE, STATUS='OLD')
OPEN(UNIT=13, FILE=KNFILE, STATUS='OLD')
OPEN(UNIT=14, FILE='CR2.TMP', ACCESS='DIRECT', RECL=NUMCR*12+8)
REWIND(10)
REWIND(12)
REWIND(13)
C

```

```

C
C Hier wordt de header van de KIN3 file gelezen en het aantal body's
bepaalt
C
  READ(13,'(A)') DUMMY2
  READ(13,'(A)') DUMMY2
  READ(13,'(A)') DUMMY2
  READ(13,'(I4,I4,I4,I4)') NUMBOD,DUMMY,DUMMY,KELVIN
  HELP=INT(NUMBOD/3)
  IF ((NUMBOD/3.0-HELP).GT.0) THEN
    HELP=HELP+1
  ENDIF
  DO 34,J=1,HELP,1
    READ(13,'(A97)') DUMMY2
34  CONTINUE
C
C Einde lezen header van KIN3 file
C
C
C
C Hier wordt de header van de RELDIS file gelezen
C
  READ(10,'(A80)') DUMMY2
  READ(10,'(A80)') DUMMY2
  READ(10,'(I4,I4)') DUMMY,DUMMY
  DO 35,J=1,NUMMAR,1
    READ (10,'(A)') DUMMY2
35  CONTINUE
  READ (10,'(A)') DUMMY2,DUMMY2,DUMMY2,DUMMY2
C
C Einde lezen header van de RELDIS file
C
C
C
C Hier wordt de header van de LINDIS file gelezen
C
  READ(12,'(A80)') DUMMY2
  READ(12,'(A80)') DUMMY2
  READ(12,'(I4,I4)') NUMLOC,DUMMY
  DO 53,J=1,NUMLOC,1
    READ (12,'(A)') DUMMY2
53  CONTINUE
  READ (12,'(A)') DUMMY2,DUMMY2,DUMMY2,DUMMY2
C
C Einde lezen van header LINDIS file
C
C
C Hier wordt de data uit alledrie de files per record gelezen en opgeslagen
in
C CR1.TMP
C
C Hier lezen uit RELDIS file
C
  DO 50,I=1,NUMREC,1
  READ(10,'(E14.6E2)') TIME
  DO 40,J=1,NUMCR*2,1
  READ (10,'(E14.6E2,E14.6E2,E14.6E2,E14.6E2)') DUMMY3,X(J),Y(J)
+  ,Z(J)
40  CONTINUE
  DO 41,J=1,NUMMAR-NUMCR*2,1
  READ(10,'(A)') DUMMY2
41  CONTINUE

```

```

C
C Einde lezen data uit RELDIS file
C
C
C Hier data lezen uit LINDIS file
C
  READ(12,'(E14.6E2)') DUMMY3
  IF (DUMMY3.NE.TIME) THEN
    WRITE (*,'(A,I4)') 'LINDIS: Different timesteps in rec: ',I
  ENDIF
  DO 65,J=1,NUMCR,1
  READ(12,'(E14.6E2,E14.6E2,E14.6E2,E14.6E2)') DUMMY3,LX(J),LY(J)
+  ,LZ(J)
65  CONTINUE
  DO 66,J=1,NUMLOC-NUMCR,1
  READ(12,'(A)') DUMMY2
66  CONTINUE
C
C Einde lezen data uit LINDIS file
C
C
C Hier lezen data uit KIN3 file
C
  READ(13,'(F9.2)') DUMMY3
  IF (DUMMY3.NE.TIME) THEN
    WRITE (*,'(A,I4)') 'KIN3: Different timesteps in rec: ',I
  ENDIF
  DO 60,J=1,NUMCR,1
  READ (13,'(7(E12.4E2))') DUMMY3,DUMMY3,DUMMY3,KAPPA(J),UX,UY,UZ
  IF (SYSTEM.EQ.1) THEN
    KAPPA(J)=KAPPA(J)*UZ
  ENDIF
  IF (SYSTEM.EQ.2) THEN
    KAPPA(J)=KAPPA(J)*UX
  ENDIF
  IF (SYSTEM.EQ.3) THEN
    KAPPA(J)=KAPPA(J)*UY
  ENDIF
60  CONTINUE
  DO 64,K=1,(NUMBOD-NUMCR)
  READ (13,'(E12.4E2)') DUMMY3
64  CONTINUE
C Overslaan van de kelvin elementen in de KIN3 file
  DO 61,J=1,KELVIN,1
  READ(13,'(F7.4)') DUMMY3
61  CONTINUE
C
C Einde lezen data uit KIN3 file
C
C
C Hier wegschrijven data naar CR1.TMP en CR2.TMP
C
  IF (SYSTEM.EQ.1) THEN
    WRITE(11,REC=I) TIME,(X(K),Y(K),K=1,NUMCR*2)
    WRITE(14,REC=I) TIME,(LX(K),LY(K),KAPPA(K),K=1,NUMCR)
  ENDIF
  IF (SYSTEM.EQ.2) THEN
    WRITE(11,REC=I) TIME,(Y(K),Z(K),K=1,NUMCR*2)
    WRITE(14,REC=I) TIME,(LY(K),LZ(K),KAPPA(K),K=1,NUMCR)
  ENDIF
  IF (SYSTEM.EQ.3) THEN
    WRITE(11,REC=I) TIME,(X(K),Z(K),K=1,NUMCR*2)

```



```

        WRITE(14,REC=I) TIME,(LX(K),LZ(K),KAPPA(K),K=1,NUMCR)
    ENDIF
C
C Einde wegschrijven naar CR1.TMP
C
50    CONTINUE
      CLOSE(10,STATUS='KEEP')
      CLOSE(11,STATUS='KEEP')
      CLOSE(12,STATUS='KEEP')
      CLOSE(13,STATUS='KEEP')
      CLOSE(14,STATUS='KEEP')
      END

      SUBROUTINE WRDATA(OUTFIL,CX,CY,NUMCR,NUMREC,SYSTEM,MODEL1
+      ,MODEL2,LX,LY,KAPPA,OUTCR)
      CHARACTER OUTFIL*12,OUTCR*12
      CHARACTER MODEL1*80,MODEL2*80
      INTEGER NUMCR,NUMREC,SYSTEM
      REAL TIME,CX(NUMCR),CY(NUMCR),LX(NUMCR),LY(NUMCR)
      REAL KAPPA(NUMCR),CXC,CYC
      OPEN(UNIT=10,FILE=OUTFIL)
      OPEN(UNIT=11,FILE='CR2.TMP',ACCESS='DIRECT',RECL=NUMCR*12+8)
      OPEN(UNIT=12,FILE='CR3.TMP',ACCESS='DIRECT',RECL=NUMCR*8+8)
      OPEN(UNIT=13,FILE=OUTCR)
      PRINT *,'Writing data'
      WRITE(10,'(A)') '#1<KIN3>'
      WRITE(10,'(A)') MODEL1
      WRITE(10,'(A)') MODEL2
      WRITE(10,'(I4,A)') NUMCR,' 0 0 0 0 0 0 0 0 0 0 0 0 0 0'
      DO 70, J=1,NUMCR,1
        WRITE(10,'(A)') '0.0050 0.0050 0.0050 2.0000'
70    CONTINUE
      DO 51, J=2,NUMREC-1,1
      READ(11,REC=J) TIME,(LX(K),LY(K),KAPPA(K),K=1,NUMCR)
      READ(12,REC=J) TIME,(CX(K),CY(K),K=1,NUMCR)
      WRITE(10,'(F8.2)') TIME
      WRITE(13,'(F8.2)') TIME
      DO 51, I=1,NUMCR,1
      IF (SYSTEM.EQ.1) THEN
        CXC=-CX(I)*COS(KAPPA(I))-CY(I)*SIN(KAPPA(I))+LX(I)
        CYC=CY(I)*COS(KAPPA(I))-CX(I)*SIN(KAPPA(I))+LY(I)
        WRITE(10,'(7(E12.4E2))') CXC,CYC,0.0,0.0,1.0,0.0,0.0
        WRITE(13,'(3(E12.4E2))') CX(I),CY(I),0.0
      ENDIF
      IF (SYSTEM.EQ.2) THEN
        CXC=-CX(I)*COS(KAPPA(I))-CY(I)*SIN(KAPPA(I))+LX(I)
        CYC=CY(I)*COS(KAPPA(I))-CX(I)*SIN(KAPPA(I))+LY(I)
        WRITE(10,'(7(E12.4E2))') 0.0,CXC,CYC,0.0,1.0,0.0,0.0
        WRITE(13,'(3(E12.4E2))') 0.0,CX(I),CY(I)
      ENDIF
      IF (SYSTEM.EQ.3) THEN
        CXC=-CX(I)*COS(KAPPA(I))+CY(I)*SIN(KAPPA(I))+LX(I)
        CYC=CY(I)*COS(KAPPA(I))+CX(I)*SIN(KAPPA(I))+LY(I)
        WRITE(10,'(7(E12.4E2))') CXC,0.0,CYC,0.0,1.0,0.0,0.0
        WRITE(13,'(3(E12.4E2))') CX(I),0.0,CY(I)
      ENDIF
51    CONTINUE
      CLOSE(10,STATUS='KEEP')
      CLOSE(11,STATUS='DELETE')

```

```

CLOSE(12,STATUS='DELETE')
CLOSE(13,STATUS='KEEP')
END

```

```

SUBROUTINE CALCUL(X1,Y1,X2,Y2,CX,CY,NUMCR,NUMREC)
INTEGER NUMREC,NUMCR
REAL X1(NUMCR*2),Y1(NUMCR*2),CX(NUMCR)
REAL X2(NUMCR*2),Y2(NUMCR*2),CY(NUMCR)
REAL TIME1,TIME2
REAL S,S1,T,T1,U,V,SINPHI,COSPHI
OPEN(UNIT=11,FILE='CR1.TMP',ACCESS='DIRECT',RECL=NUMCR*16+8)
OPEN(UNIT=12,FILE='CR3.TMP',ACCESS='DIRECT',RECL=NUMCR*8+8)
WRITE (*,'(A,I4,A)') 'Reading DATA ',NUMREC,' Records'
PRINT *,'Calculating'
DO 70,J=3,NUMREC,1
READ(11,REC=J-1) TIME1,(X1(K),Y1(K),K=1,NUMCR*2)
READ(11,REC=J) TIME2,(X2(K),Y2(K),K=1,NUMCR*2)
DO 60,I=2,NUMCR*2,2
S=X1(I-1)-X1(I)
S1=X2(I-1)-X2(I)
T=Y1(I-1)-Y1(I)
T1=Y2(I-1)-Y2(I)
COSPHI=(S1*S+T1*T)/(S*S+T*T)
SINPHI=(S1*T-T1*S)/(S*S+T*T)
IF (SINPHI.EQ.0.0) THEN
    SINPHI=0.5
ENDIF
IF (COSPHI.EQ.1.0) THEN
    COSPHI=0.5
ENDIF
U=SINPHI*(X1(I-1)-X2(I-1))/(2.*(1.-COSPHI))+(Y1(I-1)+Y2(I-1))/2.
V=-SINPHI*(Y1(I-1)-Y2(I-1))/(2.*(1.-COSPHI))+
+ (X1(I-1)+X2(I-1))/2.
CX(I/2)=X1(I-1)+(Y2(I-1)-U)/SINPHI-COSPHI*(Y1(I-1)-U)/SINPHI
CY(I/2)=Y1(I-1)-(X2(I-1)-V)/SINPHI+COSPHI*(X1(I-1)-V)/SINPHI
60 CONTINUE
WRITE (12,REC=J-1) TIME1,(CX(K),CY(K),K=1,NUMCR)
70 CONTINUE
CLOSE(11,STATUS='DELETE')
CLOSE(12,STATUS='KEEP')
END

```

In bijlage 3 staat de volledige listing van de MADYMO invoer file, hieronder volgt het deel dat toegevoegd moet worden om de berekening van de rotatiecentra mogelijk te maken. In de complete listing is dit deel **vet** weergegeven.

```

OUTPUT CONTROL PARAMETERS
0 0 0.005 2 0.005
LINDIS
1 1 0.000 0.000 0.00 -1 1 T1_LOCAL
1 2 0.000 0.000 0.00 -1 1 NECK_LOCAL
END LINDIS
RELDIS
1 2 0.00 -0.03 0.00 1 1 0.00 0.00 0.00 0 NECK1_T1
1 2 0.00 0.02 0.00 1 1 0.00 0.00 0.00 0 NECK2_T1
1 3 0.00 -0.03 0.00 1 2 0.00 0.00 0.00 0 HEAD1_NECK
1 3 0.00 0.02 0.00 1 2 0.00 0.00 0.00 0 HEAD2_NECK
END RELDIS
END OUTPUT

```

Hieronder volgt er een verklaring van het bovenstaande blok. Na deze verklaring die een min of meer vrije interpretatie vormt van de handleiding van MADYMO speciaal gericht op deze toepassing, zal duidelijk gemaakt worden hoe dit blok functioneert bij het bepalen van de rotatiecentra.

OUTPUT CONTROL PARAMETERS:

Deze regel bepaalt wat voor uitvoer er gegenereerd wordt in de KN3 file en op welke tijdstippen de files die onder ieder apart worden beschreven worden geschreven. Deze parameters zijn als volgt opgebouwd.

IOUT IKIN TSKIN OPTKIN TSOUT

IOUT Parameter voor het wel of niet aanmaken van de DEBUG file.

0 - niet aanmaken

ongelijk aan 0 - wel aanmaken

Voor het bepalen van de rotatiecentra maakt deze waarde niet uit.

IKIN Parameter voor het wel af niet aanmaken van een KINEMA file.

0 - niet aanmaken

ongelijk aan 0 - wel aanmaken

Voor het bepalen van de rotatiecentra moet deze waarde 0 zijn.

TSKIN Tijd interval waarmee de KINTWO, KIN3 en FEMESH files gemaakt worden.

OPTKIN Parameter voor de vorm van de uitvoer files KINTWO, KIN3 en FEMESH

- 0 - KINTWO, FEMESH en KIN3 niet aanmaken
- 1 - KINTWO file aanmaken
- 2 - KIN3 file aanmaken
- 3 - KIN3 file aanmaken met lokale coördinatenstelsels in de uitvoer
- 4 - FEMESH file aanmaken
- 5 - KIN3 en FEMESH file aanmaken

Voor het bepalen van de rotatiecentra moet deze waarde 2 zijn.

TSOUT Minimale tijdsinterval waarmee de overige files gegenereerd worden, deze waarde moet voor het bepalen van de rotatiecentra gelijk zijn aan de waarde voor TSKIN.

LINDIS: Dit MADYMO commando zorgt ervoor dat MADYMO een file (.LDS) aanmaakt, waarin de data weggeschreven wordt op tijdstippen die door het OUTPUT CONTROL PARAMETERS blok bepaalt worden. Wat er weggeschreven wordt staat tussen LINDIS en END LINDIS.

Het blok is als volgt opgebouwd.

SYS1 BODY1 X1 Y1 Z1 SYS2 BODY2 ID

- SYS1** Het nummer van het systeem waaraan BODY1 bevestigd is
- BODY1** Het nummer van het systeem waaraan het gevraagde punt bevestigd is.
- X1** De waarden van X1, Y1, Z1 vormen de positie van het gevraagde punt op
- Y1** BODY1 in het lokale assenstelsel van BODY1
- Z1** Voor het bepalen van de rotatiecentra is de oorsprong van de lokale assenstelsels nodig, deze waarden moeten dus alle 0.0 zijn.
- SYS2** system nummer, deze moet -1 zijn voor het bepalen van de absolute positie van de lokale assenstelsels.
- BODY2** body nummer, deze waarde moet 1 zijn. Lichaam 1 vormt dan de positie van het globale assenstelsel.
- ID** Is een string van maximaal 15 karakters waarmee in de file (.LDS) aangegeven wordt hoe het bepaalde punt heet.

RELDIS: Dit MADYMO commando zorgt ervoor dat MADYMO een file (.RDS) aanmaakt, waarin de data weggeschreven wordt op tijdstippen die door het OUTPUT CONTROL PARAMETERS blok bepaalt worden. Wat er weggeschreven wordt staat tussen RELDIS en END RELDIS.

Het blok is als volgt opgebouwd.

- SYS1 Het nummer van het systeem waaraan BODY1 bevestigd is
- BODY1 Het nummer van het systeem waaraan het gevraagde punt bevestigd is.
- X1 De waarden van X1, Y1, Z1 vormen de positie van het gevraagde punt 1 op
- Y1 BODY1 in het lokale assenstelsel van BODY1
- Z1 Deze coördinaten vormen de positie van een van de twee punten die nodig zijn om de positie van het rotatiecentrum van BODY1 te bepalen, van deze waarden moet er minimaal 1 ongelijk zijn aan 0 die in het vlak van beweging ligt. Bij het tweede punt van BODY1 (deze staat in een andere regel) moet deze waarden anders zijn dan in deze regel.
- SYS2 system nummer waaraan BODY2 bevestigd is.
- BODY2 body nummer waaraan punt 2 bevestigd is. Deze waarde moet 1 lichaam lager in de boom zitten.
- X2 De waarden van X2, Y2, Z2 vormen de positie van het gevraagde punt op
- Y2 BODY2 in het lokale assenstelsel BODY2 waarmee de relatieve verplaatsing van punt 1 bepaalt worden, deze waarden moeten allemaal 0 zijn.
- Z2
- CHO Selectie parameter die bepaalt welke verplaatsing er berekend wordt.
- 0 - Dit data blok bepaalt de relatieve verplaatsing van punt 1 ten opzichte van punt 2
- 1 - Dit data blok bepaalt de positie van punt 1 op het huidige tijdstip ten opzichte van de positie van punt 1 op tijdstip t0
- Voor het bepalen van de rotatiecentra moet deze waarden 0 zijn.
- ID Is een string van maximaal 15 karakters waarmee in de file (.RDS) aangegeven wordt hoe het bepaalde punt heet.

END OUTPUT: Deze regel sluit het uitvoer blok af.

Indien dit blok opgenomen wordt in de MADYMO data file kan MADYMO gestart worden. Hierna is het mogelijk om de rotatiecentra te bepalen. Het blok zoals dat boven weergegeven is werkt als volgt:

In de eerste regel van het LINDIS blok wordt een punt gedefinieerd op de oorsprong van het lokale assenstelsel van de torso (T1), dit punt is nodig om het rotatiecentrum van de nek te bepalen. In het RELDIS blok worden in de eerste twee regels twee punten op het lichaam van de nek gedefinieerd. Deze twee punten en het punt in LINDIS horen samen. Voor het hoofd gebeurt in feite hetzelfde, in het LINDIS blok wordt weer de oorsprong van het lokale assenstelsel van het onderliggende lichaam gedefinieerd en daarbij worden in het RELDIS blok weer twee punten gedefinieerd van het bovenliggende lichaam. Opvallend is dus dat indien er een punt in het LINDIS blok staat er twee punten in het RELDIS blok moeten staan. Deze punten horen bij elkaar, de volgorde is dus van groot belang. Voor de werking van het programma CR is het van belang dat er bij het onderste lichaam van het systeem begonnen wordt, en dan systematisch ieder lichaam meegenomen wordt, indien er een lichaam overgeslagen wordt leidt dit tot fouten.

Indien bovenstaande punten in acht genomen worden kan na het draaien van MADYMO het programma CR gestart worden met als invoer de drie al eerder genoemde files.

Het programma CR werkt als volgt:

Na het opstarten van het programma door middel van cr op de prompt in te typen worden er een aantal vragen gesteld:

- Input files, without extension :
Vul hier de naam in van de files, voor de drie files wordt slechts een naam verwacht, zonder extensie, de extensies worden door het programma toegevoegd. De filenaam moet door een eigenschap van UNIX wel acht karakters lang zijn. In het programma zijn een aantal fout afvangingen aangebracht, de respons hierop spreekt voor zich.
- Output filenaam, without extension :
Vul hier de naam van de uitvoer file in, wederom zonder extensie. Als hiervoor dezelfde file wordt gebruikt als de invoerfile zal er een melding van komen met de vraag een andere te kiezen. Bestaat de file al dan zal er gevraagd worden of deze overschreven mag worden.
- Which coordinate system: 1=XY, 2=YZ, 3=XZ :
Hier wordt gevraagd naar het vlak van beweging waarin de rotatiecentra berekend moeten worden.
- How many rotationcenters should be calculated :
Vul in hoeveel rotatiecentra er berekend moeten worden, als er meer rotatiecentra opgegeven worden als dat er punten in de RELDIS file aanwezig zijn dan wordt het maximaal mogelijk aantal genomen (bepaald aan de hand van het aantal punten in de RELDIS file gedeeld door 2).

Hierna zal het programma de punten berekenen. Tijdens dit berekenen worden er op het scherm enige meldingen gedaan, deze meldingen spreken voor zich. Er kunnen ook foutmeldingen op het scherm komen, dit zijn meldingen dat er zich verschillen hebben voorgedaan in de tijdstippen tussen bepaalde files. Deze verschillen worden gerelateerd aan de RELDIS file, dat wil zeggen dat er alleen verschillen op kunnen treden in de LINDIS en de KIN3 file, dit wordt ook aangegeven.

Oorzaken van foutmeldingen.

In de LINDIS file kan dit betekenen dat er te weinig punten in de file staan, minder als dat er in de RELDIS staan en minder dan opgegeven punten. Dit moet aangepast worden in de MADYMO invoerfile, of bij het opgeven van het aantal rotatiecentra. In de KIN3 file betekent dit dat de tijdstippen waarop de RELDIS en LINDIS files gegeneerd worden anders is als de KIN3 file.

Om dit op te lossen moet er voor gezorgd worden dat deze in de MADYMO invoerfile hetzelfde zijn (deze zijn te vinden in het OUTPUT CONTROL PARAMETERS blok).

Om het resultaat zichtbaar te kunnen maken moet er gebruik gemaakt worden van MAPPK. Hiervoor moet er echter nog een verandering in de oorspronkelijke KIN3 file gemaakt worden. Doordat CR het eerste tijdstip niet kan berekenen bestaat deze niet, als beide files gelijktijdig worden afgebeeld zullen er een aantal foutmeldingen over het scherm heen lopen, deze zeggen dat er een verschil in de tijdstippen zit. Om dit probleem op te lossen moet het eerste tijdstip in de oorspronkelijke KIN3 file weggegooid worden. Dit houdt in dat de gegevens van de positie van de lichamen en Kelvin elementen op tijdstip 0 verloren gaan, om deze te behouden moet er eerst een kopie gemaakt worden van de file. In de kopie van de KIN3 file kan dan het eerste blok weggehaald worden.

Het weergegeven gebeurt als volgt: MAPPK INPUTFILE1.KN3 INPUTFILE2.KN3
Het is ook mogelijk om deze met VIEWK te bekijken het commando wordt dan
VIEWK -2 INPUTFILE1.KN3 INPUTFILE2.KN3

Volledige MADYMO 5.1 invoer file voor het twopivotmodel.
Deze listing is enigszins gestyleerd. Het vet weergegeven deel is nodig voor het bepalen van de rotatiecentra

WISMANS MODEL FRONTAL FLEXION
TWO PIVOT MODEL.
MARCH 1993

!Wismans model with upper pivot free range of motion of 30 dgr.
!Joints modelled by means of two revolute joints.
!Dit is een aangepaste versie voor vergelijk met model v. Deng.

0.0000 0.2500
RUKU4 0.0010 0.0010 0.0050
0.1000 1.0000 0.0100 0.1000

SYSTEM 1
HEAD-NECK SYSTEM

CONFIGURATION
3 2 1
END CONFIGURATION

! Average human head-neck data used.
! T1 pos. is (0,-0.031,-0.011) with respect to low. piv.

GEOMETRY
0.0000 0.0000 0.0000 0.0000 0.0000 0.0000 T1
0.0000 0.0000 0.0000 0.0000 0.0023 0.0651 NECK
0.0000 0.0000 0.1290 0.0000 -0.0230 0.0550 HEAD
END GEOMETRY

INERTIA
1.E3 1.E1 1.E1 1.E1
1.22 0.0100 0.0100 0.0100
4.78 0.0293 0.0267 0.0149
END INERTIA

ORIENTATIONS
3 0 1 1 -0.6283
END ORIENTATIONS

JOINTS

1 BRAC

2 REVO

3 REVO

END JOINTS

ORIENTATIONS

2 2 1 1 -0.298

2 1 1 1 0.0

3 2 1 1 0.0

3 3 1 1 0.379

END ORIENTATIONS

STIFFNESS

2 1 0 0 0 0.0 0.0

3 2 3 -172 0.0 0.0

END STIFFNESS

FUNCTIONS

! 1:neck lower joint flexion

3

-2.0 -137.5 0.0 0.0 2.0 137.5

! 2:neck upper joint flexion

7

-1.524 -171.9 -1.024 -85.95 -0.524 -0.01 0.0 0.0 0.524 0.01 1.024 85.95 1.524 171.9

! 3: neck upper joint unloading function (hysteresys model 2)

2

-2.0 200.0 2.0 210.0

END FUNCTIONS

ELLIPSOIDS

1 0.100 0.100 0.0500 0.0 0.0 -0.0500 5 0 0 0 Thorax

2 0.045 0.045 0.0645 0.0 0.0 0.0645 2 0 0 0. NECK

3 0.073 0.105 0.1050 0.0 0.0 0.1050 2 0 0 0. HEAD

END ELLIPSOIDS

INITIAL CONDITIONS

0. 0. 0. 0. 0. 0.

JOINT DOF

2 FREE 0.0 0.0

3 FREE 0.0 0.0

END INITIAL CONDITIONS

END SYSTEM 1

FORCE MODELS

ACCELERATION FIELDS

! prescribed T1 acceleration on the head-neck

! gravity on head-neck

1 1 0 1 2

1 2 0 1 2

1 3 0 1 2

END ACCELERATION FIELDS

FUNCTIONS

! 1:prescribed T1 acceleration according to Deng and Goldsmith

4

0.0 0.0 0.014 -72.6 0.30 0.0 1.0 0.0

! 2: gravity

2

0.0 0.0 0.50 0.0

END FUNCTIONS

END FORCE MODELS

OUTPUT CONTROL PARAMETERS

0 0 0.005 2 0.005

LINDIS

1 1 0.000 0.000 0.00 1 1 NECK_LOCAL

1 2 0.000 0.000 0.00 1 1 HEAD_LOCAL

END LINDIS

RELDIS

1 2 0.00 -0.02 0.00 1 1 0.00 0.00 0.00 0 NECK1_T1

1 2 0.00 0.02 0.00 1 1 0.00 0.00 0.00 0 NECK2_T1

1 3 0.00 -0.02 0.00 1 2 0.00 0.00 0.00 0 HEAD1_NECK

1 3 0.00 0.02 0.00 1 2 0.00 0.00 0.00 0 HEAD2_NECK

END RELDIS

END OUTPUT

END INPUT