

Een eindige elementenanalyse van een stroming rond een cilinder en visualisatie met AVS

Citation for published version (APA):

de Crom, P. J. J. (1993). *Een eindige elementenanalyse van een stroming rond een cilinder en visualisatie met AVS*. (DCT rapporten; Vol. 1993.129). Technische Universiteit Eindhoven.

Document status and date:

Gepubliceerd: 01/01/1993

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.

EEN EINDIGE ELEMENTENANALYSE VAN
EEN STROMING ROND EEN CYLINDER
EN VISUALISTIE MET AVS

P.J.J. DE CROM

nr : 93129

EEN EINDIGE ELEMENTENANALYSE VAN EEN STROMING ROND EEN CYLINDER EN VISUALISTIE MET AVS

P.J.J. DE CROM

SAMENVATTING

Gebruik makend van de eindige elementen methode wordt een benaderingsoplossing van een 2-dimensionale, instationaire stroming om een cylinder geanalyseerd.

Teneinde een zogenaamde wervelstraat van Von Karman te initiëren, is de stationaire stroming rond de cylinder gedurende één tijdstap door een pulsbeving van de cylinder, loodrecht op de stromingsrichting, verstoord

Voor de stroming is een vloeistof met een Reynolds-getal van 100 genomen. De resultaten die zijn verkregen, zijn gevisualiseerd met behulp van het visualisatie pakket AVS.

INLEIDING

Met behulp van de eindige elementen methode wordt een stroming om een cylinder gemodelleerd. De cylinder krijgt op een gegeven moment een puls, waardoor er in de stroming achter de cylinder een Von Karman straat wordt ontwikkeld. De modellering wordt uitgevoerd met behulp van het programma Sepran, waarna de uitvoergegevens verder worden verwerkt met het visualisatie pakket AVS.

In AVS wordt de stroming gevisualiseerd, waardoor de veranderingen in het verloop van de tijd goed kunnen worden waargenomen.

In dit verslag wordt de numerieke methode besproken die is gebruikt. Hierin wordt iets over de Eindige Elementen Methode gezegd, gevolgd door de resultaten en de conclusies die hieruit kunnen worden getrokken.

In een bijlage wordt tevens een summiere handleiding gegeven over het werken met AVS. Hierin worden de modules en de netwerken, die in deze studie zijn gebruikt, beschreven.

DE GEBRUIKTE VERGELIJKINGEN

De vergelijkingen die gebruikt zijn bij het oplossen van het stromingsprobleem zijn de :

* instationaire Navier Stokes vergelijking :

$$\rho \frac{\partial u_i}{\partial t} + \sum_{j=1}^2 \rho u_j \frac{\partial u_i}{\partial x_j} = \rho f_i + \sum_{j=1}^2 \frac{\partial \sigma_{ij}}{\partial x_j} \quad (i = 1, 2) \quad (1)$$

Hierbij geldt voor Newtonse media het constitutief verband:

$$\sigma_{ij} = -p\delta_{ij} + \eta \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (i, j = 1, 2) \quad (1a)$$

* continuïteitsvergelijking :

$$\sum_{j=1}^2 \frac{\partial u_j}{\partial x_j} = 0 \quad (2)$$

De symbolen staan voor :

- ρ = dichtheid
- σ_{ij} = de componenten van de Cauchy spanningstensor
- p = druk
- δ_{ij} = Kroneckerdelta
- η = dynamische viscositeit
- u_i = de i^{de} component van de snelheidsvector
- f = uitwendige belastingsvector

DE EINDIGE ELEMENTEN METHODE

De instationaire Navier-Stokes vergelijkingen worden gediscrètiseerd met behulp van de Galerkin methode. De snelheid en de druk worden benadert door een lineaire combinatie van onafhankelijke basisfuncties, ϕ en ψ

$$\tilde{u}_i = \sum_{n=1}^M u_{in} \phi_{in} \quad (i = 1, 2) \quad (3)$$

$$\tilde{p} = \sum_{m=1}^M p_m \psi_m \quad (4)$$

Bij de methode van Galerkin worden de Navier-Stokes vergelijkingen met een weegfunctie vermenigvuldigt en vervolgens geïntegreert over het gebied Ω . Als weegfunctie voor de impulsvergelijking wordt de basisfunctie ϕ_{ik} en voor de continuïteitsvergelijking de basisfunctie ψ_m gebruikt.

De vergelijkingen (1) en (2) worden nu :

$$\int_{\Omega} \rho \frac{\partial u_i}{\partial t} \varphi_{ik} d\Omega + \sum_{j=1}^2 \int_{\Omega} \rho u_j \frac{\partial u_i}{\partial x_j} \varphi_{ik} d\Omega = \int_{\Omega} \rho f_i \varphi_{ik} d\Omega + \sum_{j=1}^2 \int_{\Omega} \frac{\partial \sigma_{ij}}{\partial x_i} \varphi_{ik} d\Omega \quad (5)$$

$$\sum_{j=1}^2 \int_{\Omega} \frac{\partial u_j}{\partial x_j} \psi_m d\Omega \quad (i = 1, 2, k = 1, \dots, N, m = 1, \dots, M)$$

De laatste term van de impulsvergelijking, wordt na partieel integreren en toepassen van de stelling van Gauss :

$$\sum_{j=1}^2 \int_{\Omega} \frac{\partial \sigma_{ij}}{\partial x_j} \varphi_{ik} d\Omega = - \sum_{j=1}^2 \int_{\Omega} \sigma_{ij} \frac{\partial \varphi_{ik}}{\partial x_j} d\Omega + \sum_{j=1}^2 \int_{\Gamma} \sigma_{ij} \varphi_{ik} n_j d\Gamma \quad (6)$$

(k = 1, ..., N, i = 1, 2)

In deze vergelijking (6) is Γ de rand van het gebied Ω met een naar buiten gerichte normaal n . Als in de eerste term van het rechterlid van (6) de constitutieve vergelijking (1a) wordt ingevuld, wordt vergelijking (5) :

$$\int_{\Omega} \rho \frac{\partial u_i}{\partial t} \varphi_{ik} d\Omega + \int_{\Omega} \sum_{j=1}^2 \rho u_j \frac{\partial u_i}{\partial x_j} \varphi_{ik} d\Omega = \int_{\Omega} \rho f_i \varphi_{ik} d\Omega + \int_{\Omega} p \frac{\partial \varphi_{ik}}{\partial x_i} d\Omega +$$

$$- \int_{\Omega} \sum_{j=1}^2 \eta \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \frac{\partial \varphi_{ik}}{\partial x_j} d\Omega + \int_{\Gamma} \sum_{j=1}^2 \sigma_{ij} \varphi_{ik} n_j d\Gamma \quad (7)$$

$$\int_{\Omega} \sum_{j=1}^2 \frac{\partial u_j}{\partial x_j} \psi_m d\Omega = 0 \quad (i = 1, 2, k = 1, \dots, N, m = 1, \dots, M)$$

Dit wordt verder per component van de vergelijkingen uitgewerkt, waarbij steeds eerst de benadering voor de snelheid (3) of de druk (4) wordt ingevuld.

* De instationaire term $\int_{\Omega} \rho \frac{\partial \tilde{u}_i}{\partial t} \varphi_{ik} d\Omega =$

$$\sum_{n=1}^N \dot{u}_{in} \int_{\Omega} \rho \varphi_{in} \varphi_{ik} d\Omega \quad (i = 1, 2 \text{ en } k = 1, \dots, N)$$

Dit kan als volgt in matrixvorm worden geschreven :

$$\underline{M} \dot{\underline{u}} \quad \text{met} \quad M = \begin{bmatrix} M^{11} & M^{12} \\ M^{21} & M^{22} \end{bmatrix} \quad \text{waarin geldt :}$$

$$M^{ij}(k, n) = \int_{\Omega} \rho \delta_{ij} \varphi_{ik} \varphi_{jn} d\Omega \quad \text{met } k = 1, \dots, N, n = 1, \dots, N$$

$$\text{en } \underline{u} = [\underline{u}_1, \underline{u}_2]^T \quad \text{waarin geldt dat } \underline{u}_i = [\dot{u}_{i1}, \dots, \dot{u}_{iN}]$$

M wordt de massamatrix genoemd.

* De convectieve term $\int_{\Omega} \sum_{j=1}^2 \rho \tilde{u}_j \frac{\partial \tilde{u}_i}{\partial x_j} \varphi_{ik} d\Omega =$

$$\int_{\Omega} \rho \sum_{j=1}^2 \left(\sum_{n=1}^N u_{jn} \varphi_{jn} \sum_{l=1}^N u_{il} \frac{\partial \varphi_{il}}{\partial x_j} \right) \varphi_{ik} d\Omega \quad (i = 1, 2 \text{ en } k = 1, \dots, N)$$

Dit is een niet-lineaire term, die wordt gelineariseerd met de iteratiemethode van Newton. Bij de Newton iteratie wordt uitgegaan van de aanname $\underline{u}^I = \underline{u}^{I-1} + \delta \underline{u}^{I-1}$. Hier is \underline{u}^{I-1} een reeds bekende snelheid (uit vorige iteratieslag).

De convectieve term wordt gelineariseerd door $\underline{u}^I \cdot \nabla \underline{u}^I = (\underline{u}^{I-1} + \delta \underline{u}^{I-1}) \cdot \nabla (\underline{u}^{I-1} + \delta \underline{u}^{I-1})$ uit te schrijven, waarbij de term $\delta \underline{u}^{I-1} \cdot \nabla \delta \underline{u}^{I-1}$ wordt verwaarloosd. Dit levert de volgende term

$$: \underline{u}^I \cdot \nabla \underline{u}^I = \underline{u}^{I-1} \cdot \nabla \underline{u}^I + \underline{u}^I \cdot \nabla \underline{u}^{I-1} - \underline{u}^{I-1} \cdot \nabla \underline{u}^{I-1}$$

Hiermee wordt de volgende matrixnotatie verkregen :

$$N(\underline{u}^I) \underline{u}^I = C(\underline{u}^{I-1}) \underline{u}^I + D(\underline{u}^{I-1}) \underline{u}^I - N(\underline{u}^{I-1}) \underline{u}^{I-1} \\ = J(\underline{u}^{I-1}) \underline{u}^I - N(\underline{u}^{I-1}) \underline{u}^{I-1}$$

$$\text{met } J(\underline{u}^{I-1}) = \begin{bmatrix} J^{11}(\underline{u}^{I-1}) & J^{12}(\underline{u}^{I-1}) \\ J^{21}(\underline{u}^{I-1}) & J^{22}(\underline{u}^{I-1}) \end{bmatrix}$$

waarin $J^{ij}(k, l) = K^i(k, n) \delta_{ij} + C^{ij}(k, l)$

$$K^i(k, n) = \sum_{p=1}^N \sum_{j=1}^2 u_{jp}^{I-1} \int_{\Omega} \rho \varphi_{jp} \varphi_{ik} \frac{\partial \varphi_{in}}{\partial x_j} d\Omega \quad (k = 1, \dots, N, n = 1, \dots, N)$$

$$C^{ij}(k, l) = \sum_{p=1}^N u_{ip}^{I-1} \int_{\Omega} \rho \varphi_{jl} \varphi_{ik} \frac{\partial \varphi_{ip}}{\partial x_j} d\Omega \quad (l = 1, \dots, N)$$

Voor verdere uitwerking zie Appendix A.
N(u) is de convectieve matrix.

* De externe krachten $\int_{\Omega} \rho f_i \varphi_{ik} d\Omega$.

Deze levert de krachtenvector :

$$\underline{F} = \begin{bmatrix} \underline{F}^1 \\ \underline{F}^2 \end{bmatrix} \quad \text{met} \quad F^i_k = \int_{\Omega} \rho f_i \varphi_{ik} d\Omega \quad (i = 1, 2 \text{ en } k = 1, \dots, N)$$

* De drukterm $-\int_{\Omega} \tilde{p} \frac{\partial \varphi_{ik}}{\partial x_i} d\Omega$.

Nu de benadering voor p invullen : $-\int_{\Omega} \sum_{m=1}^M p_m \psi_m \frac{\partial \varphi_{ik}}{\partial x_i} d\Omega$

Dit levert de gradiëntmatrix L^F :

$$L = (L^1 \ L^2) \quad \text{met} \quad (L^i)^T(m, k) = - \int_{\Omega} \psi_m \frac{\partial \varphi_{ik}}{\partial x_i} d\Omega$$

met $i = 1, 2$, $k = 1, \dots, N$ en $m = 1, \dots, M$.

$$* \text{ De diffusie-term } \int_{\Omega} \sum_{j=1}^2 \eta \left(\frac{\partial \tilde{u}_i}{\partial x_j} + \frac{\partial \tilde{u}_j}{\partial x_i} \right) \frac{\partial \varphi_{ik}}{\partial x_j} d\Omega = .$$

$$\int_{\Omega} \left[\sum_{j=1}^2 \sum_{n=1}^N \eta \left(u_{in} \frac{\partial \varphi_{in}}{\partial x_j} + u_{jn} \frac{\partial \varphi_{jn}}{\partial x_i} \right) \frac{\partial \varphi_{ik}}{\partial x_j} \right] d\Omega =$$

$$\sum_{n=1}^N \sum_{j=1}^2 u_{jn} \delta_{ij} \int_{\Omega} \sum_{\alpha=1}^2 \eta \frac{\partial \varphi_{jn}}{\partial x_{\alpha}} \frac{\partial \varphi_{ik}}{\partial x_{\alpha}} d\Omega + \sum_{j=1}^2 \sum_{n=1}^N u_{jn} \int_{\Omega} \frac{\partial \varphi_{jn}}{\partial x_i} \frac{\partial \varphi_{ik}}{\partial x_j} d\Omega$$

($i = 1, 2$, $k = 1, \dots, N$)

Dit levert de diffusiematrix S volgens :

$$S = \begin{bmatrix} S^{11} & S^{12} \\ S^{21} & S^{22} \end{bmatrix} \text{ met } S^{ij}(k, n) = \int_{\Omega} \eta \left\{ \sum_{\alpha=1}^2 \left(\frac{\partial \varphi_{ik}}{\partial x_{\alpha}} \frac{\partial \varphi_{jn}}{\partial x_{\alpha}} \right) \delta_{ij} + \frac{\partial \varphi_{ik}}{\partial x_j} \frac{\partial \varphi_{jn}}{\partial x_i} \right\} d\Omega$$

met $i = 1, 2$, $j = 1, 2$, $k = 1, \dots, N$ en $n = 1, \dots, N$.

$$* \text{ De randspanningen } \int_{\Gamma} \sum_{j=1}^2 \sigma_{ij} \varphi_{ik} n_j d\Gamma .$$

Dit levert de randspanningsvector B :

$$B = \begin{bmatrix} B^1 \\ B^2 \end{bmatrix} \text{ met } B^i(k) = \int_{\Gamma} \sum_{j=1}^2 \sigma_{ij} n_j \varphi_{ik} d\Gamma \quad (k = 1, \dots, N)$$

Na discretisatie worden dus de volgende vergelijkingen verkregen :

$$M \dot{\underline{u}} + [S + N(\underline{u})] \underline{u} + L^T \underline{p} = \underline{f} + \underline{b} \quad (8a)$$

$$L \underline{u} = \underline{0} \quad (8b)$$

De vector \underline{u} bevat $2N$ snelheidsparameters u_{in} ($i = 1, 2$; $n = 1, \dots, N$) en \underline{p} is een vector waarin m drukparameters p_m ($m = 1, \dots, M$) staan.

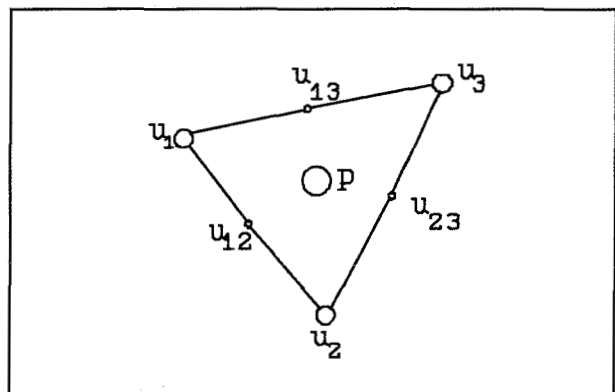
HET UITGEBREIDE CROUZEIX-RAVIART ELEMENT

De 2-dimensionale ruimte (Ω) wordt onderverdeeld in elementen. Er is in deze studie gebruik gemaakt van de gemodificeerde Crouzeix-Raviart element ($P_2^+ - P_1$). Dit element is afgeleid van de uitgebreide kwadratische element dat bestaat uit 7 knooppunten, waarbij elk knooppunt 2 snelheidswaarden bevat en alleen het centrale knooppunt de druk onbekende bevat (p , dp/dx , dp/dy). Dit element heeft dus in totaal 17 onbekenden. Door dit grote aantal vrije parameters vergt het veel reken-tijd.

Het is echter mogelijk om de snelheidstermen en de afgeleide van de druk van het centrale knooppunt te elimineren, waardoor er nog maar 13 onbekende overblijven.

Het element ziet er als volgt uit :

Door deze eliminatie wordt de nauwkeurigheid niet aange-tast, maar de rekentijd wel verkort. (Van de Vosse, 1986)



FIGUUR 1: Het gemodificeerde Crouzeix-Raviart element

DE TIJDINTEGRATIE

De tijdintegratie in de discrete Navier-Stokes vergelijkingen kunnen benaderd worden door een eindige differentie Θ -methode.

Beschouw hiervoor de vergelijking : $\dot{\underline{u}} = \underline{A}\underline{u} + \underline{f}$

waarvan de benadering wordt gedefinieerd door :

$$\frac{\underline{u}^{n+1} - \underline{u}^n}{\Delta t} = \Theta \underline{A} \underline{u}^{n+1} + (1-\Theta) \underline{A} \underline{u}^n + \underline{f}^{n+\Theta}$$

$$\text{met } \underline{f}^{n+\Theta} = \Theta \underline{f}^{n+1} + (1-\Theta) \underline{f}^n$$

Er kunnen 2 methoden worden onderscheiden, namelijk :

* De Euler Impliciet (ei) methode waar $\Theta=1$

* De Crank-Nicolson (cn) methode met $\Theta=0.5$

De ei-methode heeft een nauwkeurigheid van $O(\Delta t)$, terwijl de cn-methode een nauwkeurigheid heeft van $O(\Delta t^2)$.

Door berekeningsfouten of fouten in de beginvoorwaarde kunnen er oscillaties ontstaan. Het blijkt dat cn deze oscillaties versterkt (versterkingsfactor naar -1 voor grote negatieve eigenwaarden), terwijl ei juist deze oscillaties uitdempt (versterkingsfactor naar 0). Daarom worden in deze studie de

eerste stappen in de iteratie met ei berekend en daarna overgestapt op cn, zodat men geen last heeft van de oscillaties ten gevolge van de beginvoorwaarde.

DE BOETEFUNCTIE-METHODE

Na toepassing van de discretisatie met behulp van de eindige elementen methode, worden de niet-lineaire instationaire Navier-Stokes vergelijkingen als volgt geschreven :

$$M\dot{\underline{u}} + [S+N(\underline{u})]\underline{u} + L^T \underline{p} = \underline{f} + \underline{b} \quad (8a)$$

$$L\underline{u} = \underline{0} \quad (8b)$$

Toepassen van tijdsintegratie volgens de Θ -methode levert :

$$\begin{aligned} (M + \Theta\Delta t [S + N(\underline{u}^{n+1})])\underline{u}^{n+1} + \Theta\Delta t L^T \underline{p}^{n+1} &= \Delta t (\underline{f}^{n+\Theta} + \underline{b}^{n+\Theta}) + \\ (M - (1-\Theta)\Delta t [S + N(\underline{u}^n)])\underline{u}^n - (1-\Theta)\Delta t L^T \underline{p}^n & \\ L\underline{u}^{n+1} = \underline{0} & \end{aligned} \quad (9)$$

Dit stelsel kan direct worden opgelost, maar heeft als groot nadeel dat er dan nul-elementen op de hoofddiagonaal van de coëfficiënten-matrix voorkomen, waardoor het oplossen veel tijd en geheugen vergt. Het verschijnen van de nul-elementen op de diagonaal is het gevolg van niet voorkomen van de druk in de continuïteitsvergelijking.

Dit kan verholpen worden met behulp van de boete functie methode. Hierbij wordt in plaats van de continuïteitsvergelijking $\text{div}\underline{u} = 0$, $\text{div}\underline{u} = -\varepsilon p$ gebruikt. Hierbij is de boete functie parameter ε klein.

Dit discretiseren met behulp van Galerkin levert via :

$$\sum_{j=1}^2 \frac{\partial u_j}{\partial x_j} = -\varepsilon p \rightarrow \sum_{j=1}^2 \int_{\Omega} \sum_{n=1}^N u_{jn} \frac{\partial \varphi_{jn}}{\partial x_j} \psi_m d\Omega = -\varepsilon \int_{\Omega} \sum_{l=1}^M p_l \psi_l \psi_m d\Omega$$

de vergelijking : $L\underline{u} = \varepsilon M_p \underline{p}$ met $M_p = \int_{\Omega} \psi_l \psi_m d\Omega$

met $l = 1, \dots, M$ en $m = 1, \dots, M$

Dit in vergelijking (9) invullen levert het stelsel :

$$(M + \Theta\Delta t [S + N(\underline{u}^{n+1}) + \frac{1}{\varepsilon} L^T M_p^{-1} L])\underline{u}^{n+1} =$$

$$\Delta t (\underline{f}^{n+\Theta} + \underline{b}^{n+\Theta}) + (M - (1-\Theta)\Delta t [S + N(\underline{u}^n) + \frac{1}{\varepsilon} L^T M_p^{-1} L])\underline{u}^n$$

$$\underline{p}^{n+1} = \frac{1}{\varepsilon} M_p^{-1} L \underline{u}^{n+1}$$

Dit stelsel is niet lineair. Gebruik maken van de Newton linearisatie : $N(\underline{u}^{n+1})\underline{u}^{n+1} = J(\underline{u}^n)\underline{u}^{n+1} - N(\underline{u}^n)\underline{u}^n$ (zie convectieve term) leidt dan tot het volgende stelsel :

$$\{M + \Theta \Delta t [S + J(\underline{u}^n) + \frac{1}{\varepsilon} L^T M_p^{-1} L]\} \underline{u}^{n+1} = \Delta t (\underline{f}^{n+\Theta} + \underline{b}^{n+\Theta}) +$$

$$\{M - (1-\Theta) \Delta t [S + \frac{1}{\varepsilon} L^T M_p^{-1} L]\} \underline{u}^n - (1-2\Theta) \Delta t N(\underline{u}^n) \underline{u}^n$$

$$\underline{p}^{n+1} = \frac{1}{\varepsilon} M_p^{-1} L \underline{u}^{n+1}$$

Substitutie van $\underline{u}^{n+1} = 1/\Theta [\underline{u}^{n+\Theta} - (1-\Theta) \underline{u}^n]$ leidt tot een oplossingsprocedure die uit 2 stappen bestaat :

$$1) \left[\frac{1}{\Theta \Delta t} M + S + J(\underline{u}^n) + \frac{1}{\varepsilon} L^T M_p^{-1} L \right] \underline{u}^{n+\Theta} = \frac{1}{\Theta \Delta t} M \underline{u}^n + \underline{f}^{n+\Theta} + \underline{b}^{n+\Theta} + \Theta N(\underline{u}^n) \underline{u}^n$$

$$\underline{p}^{n+\Theta} = \frac{1}{\varepsilon} M_p^{-1} L \underline{u}^{n+\Theta}$$

en

$$2) \underline{u}^{n+1} = \frac{1}{\Theta} [\underline{u}^{n+\Theta} - (1-\Theta) \underline{u}^n]$$

$$\underline{p}^{n+1} = \frac{1}{\Theta} [\underline{p}^{n+\Theta} - (1-\Theta) \underline{p}^n]$$

De eerste stap is een Euler Impliciete stap naar tijdnivo $n+\Theta$, welke onvoorwaardelijk stabiel is. De tweede stap is voorwaardelijk stabiel, want deze bevat de instabiele effecten van de Θ -methode.

In de eerste stap worden de effecten van mogelijke numerieke oscillaties in de snelheid en de druk tot een minimum beperkt. Omdat de druk op tijdnivo $n+1$ niet nodig is om de tijdintegratie voort te zetten, kan men voor de druk de tweede stap weglaten.

RESULTATEN

INLEIDING

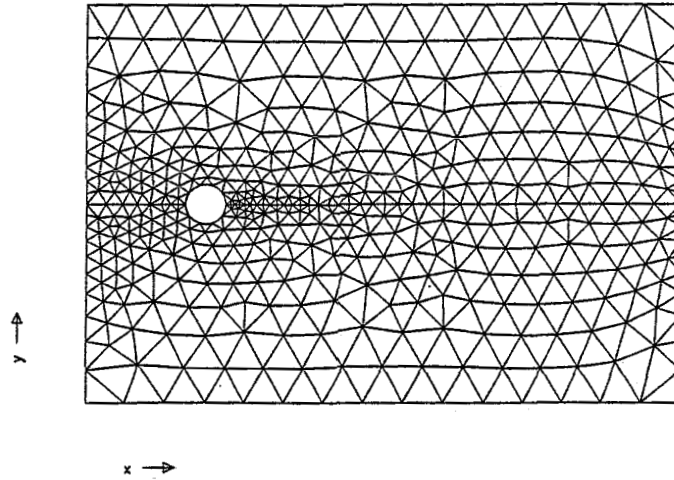
Zoals vermeld is, is in deze studie een stroming om een cylinder gemodelleerd. De stroming rond de cylinder wordt gekarakteriseerd door twee dimensieloze getallen :

$$- Re = \frac{DU_0}{\nu} = 100$$

$$- St = \frac{fD}{U_0}$$

Van de stroming, die hier numeriek gesimuleerd wordt, is experimenteel door Tritton [4] vastgesteld dat het Strouhal-

getal voor de stroming met $Re = 100$ gelijk is aan 0,16. Voor de numerieke berekeningen is een eindige elementen mesh gegenereerd, welke in figuur 2 is geplott. Omdat het te verwachten is dat er in de buurt van de cylinder de meeste veranderingen in de stroming optreden, is daar elementen verfijning toegepast.



FIGUUR 2 : de elementenverdeling

Op de "randen" en de ingang wordt een dimensieloze horizontale snelheid van 1 voorgeschreven, terwijl op de uitgang een spanningsloze situatie (σ_n en σ_t) wordt aangenomen.

In eerste instantie wordt de stationaire stroming (in 10 iteratiestappen) berekend, zie figuur 3 op de volgende pagina.

Daarna wordt de stroming verstoord doordat de cylinder op $t=0$ een snelheidspuls van 1 omhoog loodrecht op de stroomrichting krijgt. Na deze verstoring wordt de instationaire stroming berekend. De tijdsinterval waarin de berekeningen worden uitgevoerd, bestaat uit 80 dimensieloze tijdstappen. Hiervan zijn de eerste 70 stappen met grootte eenheid 1 en de laatste met $0,25 * \text{eenheidsstap}$ grootte genomen. Dit is gedaan, omdat de verwachting is dat na verloop van tijd de stroming ontwikkeld is, zodat er een periodieke stroming is ontstaan.

Door kleinere tijdstappen te nemen is dit beter waar te nemen. De eerste 10 tijdstappen wordt de Euler-Impliciete methode toegepast, daarna wordt overgegaan op de Crank-Nicolson methode.

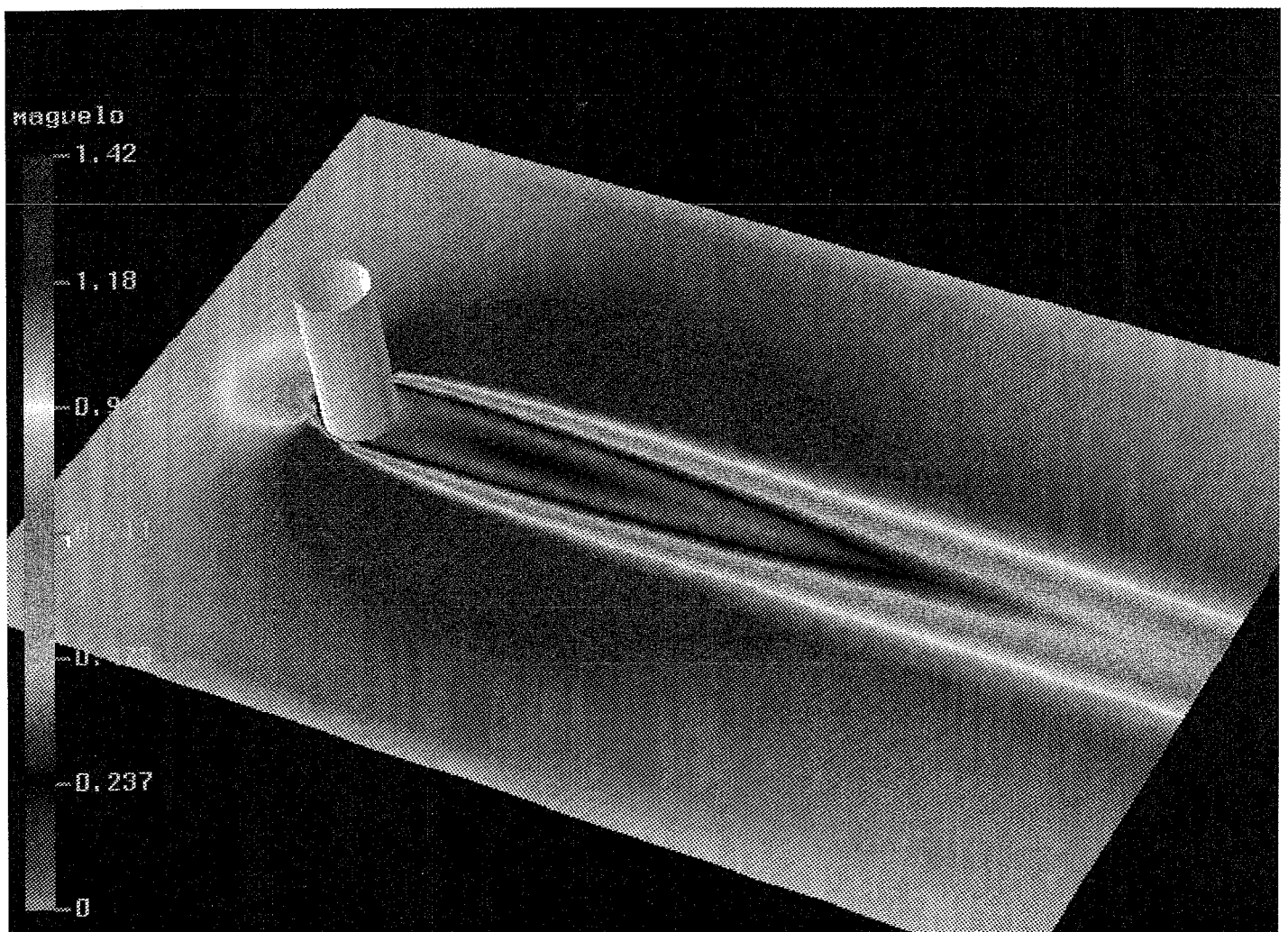
Voor iedere tijdstap worden de snelheid, druk en de vortichtheid berekend. De boete functieparameter wordt voor alle berekeningen gelijk aan $1D-6$ genomen.

RESULTATEN

De numerieke simulatie van de stroming om een cylinder is gevisualiseerd met AVS, waarvan een video presentatie is gemaakt. De bespreking van de resultaten die hierna volgt is aan de hand van deze video gedaan.

Er is in deze studie gekeken naar de snelheid in vectorvorm, de stroomlijnen, de vortichtheid en de druk. Er is tevens gekeken of er een periodieke stroming ontstaat.

FIGUUR 3. STATIONNAIRE STROMING



stroomlijnen

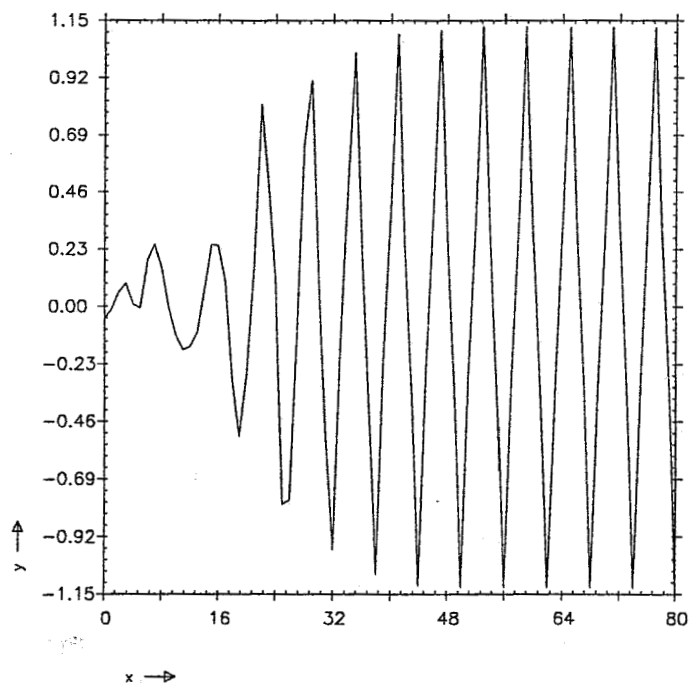
Bij de stroomlijnen is goed waar te nemen hoe de stroming gaat oscilleren achter de cylinder. De laatste 6 hele tijdstappen is de stroming volledig ontwikkeld en zou er eventueel een periodieke stroming kunnen optreden. (De laatste 6 hele tijdstappen komt overeen met 24 berekende tijdstappen).

Bij het maken van de video zijn deze 24 stappen 4 keer achterelkaar gezet, zodat goed kan worden waargenomen of er sprake is van een periodieke stroming. Als hiernaar gekeken wordt dan blijkt dat er inderdaad een vloeiende overgang is tussen de begin- en eindtijdstap bij de stroomlijnen. Dus hier kan de conclusie worden getrokken dat er een periodieke stroming is ontstaan. Op de volgende pagina's staan 6 stappen uit de stroming weergegeven onder de kop STROOMFUNCTIE.

De periodetijd kan worden afgelezen uit de figuur waar de snelheid ten opzichte van de tijd staat uitgezet (figuur 4). Hieruit valt af te lezen dat één periode 6 dimensieloze seconden in beslag neemt. De frequentie van de stroming is dus :

$$f = \frac{1}{\Delta T} = \frac{1}{6} = 0,167$$

FIGUUR 4 :
snelheid-tijd grafiek



Met de frequentie wordt het Strouhal-getal : $St = 0,167$
Dit komt goed overeen met de waarde die Tritton gevonden heeft.

vectorgrootte

In AVS kan van de snelheid de grootte van de vectoren worden berekend. Bij de visualisatie hiervan is goed te zien waar de wervels zich ontwikkelen en hoe ze verder voortbewegen. Hier is onder andere te zien dat er om beurten boven en onder de horizontale symmetrielijns achter de cylinder een wervel ontstaat.

stroomfunctie

fig. 1

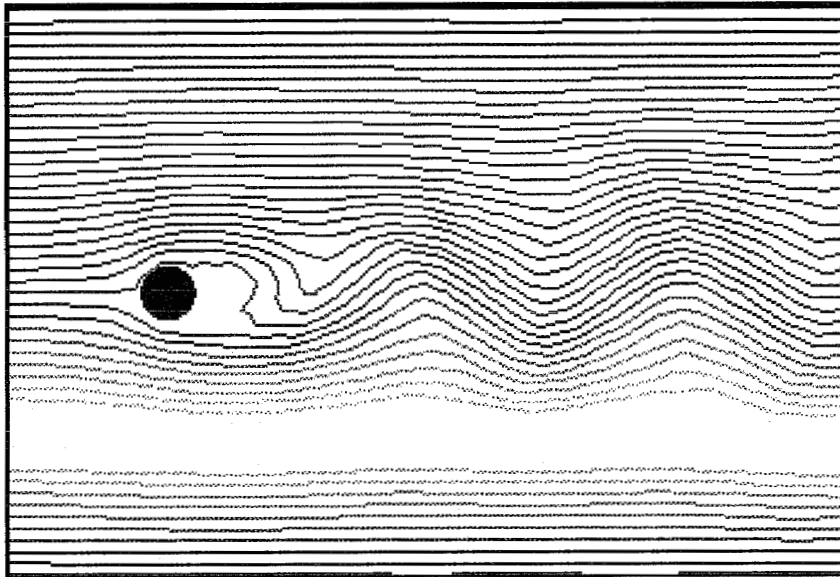


fig. 2

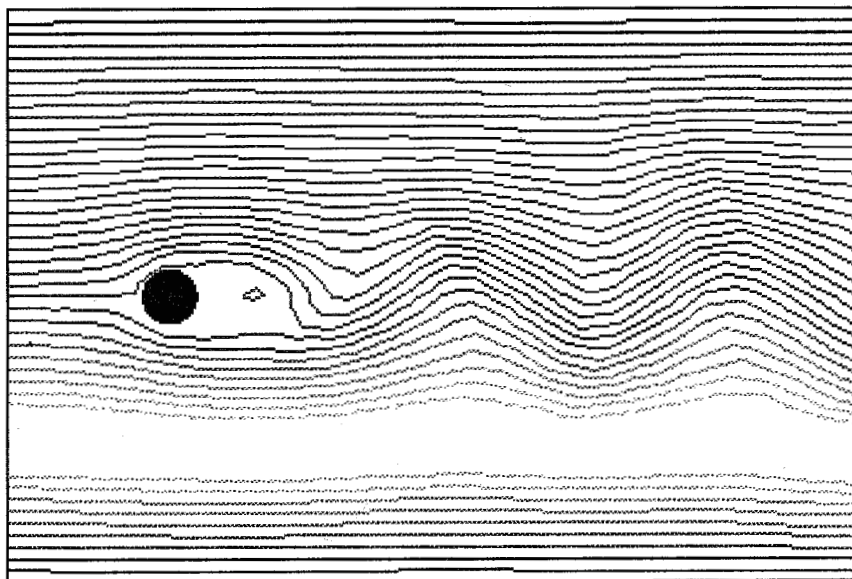


fig. 3

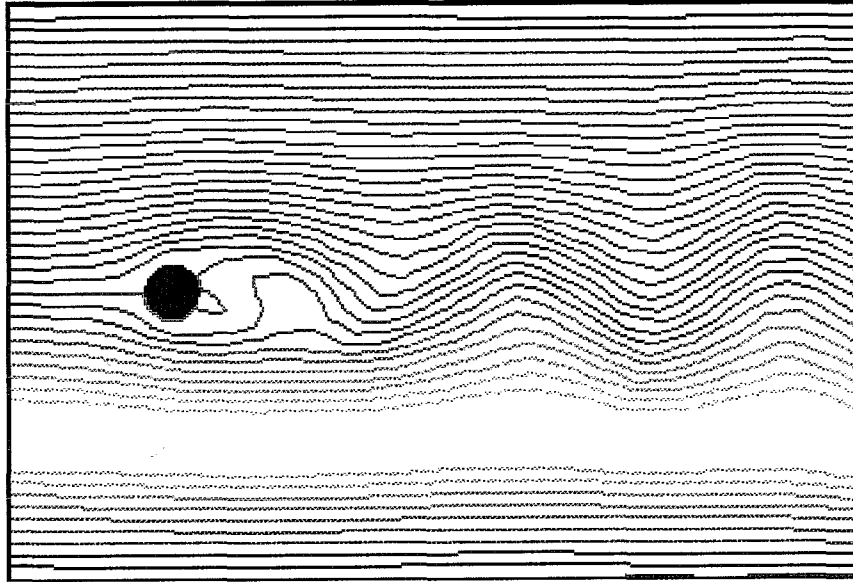


fig. 4

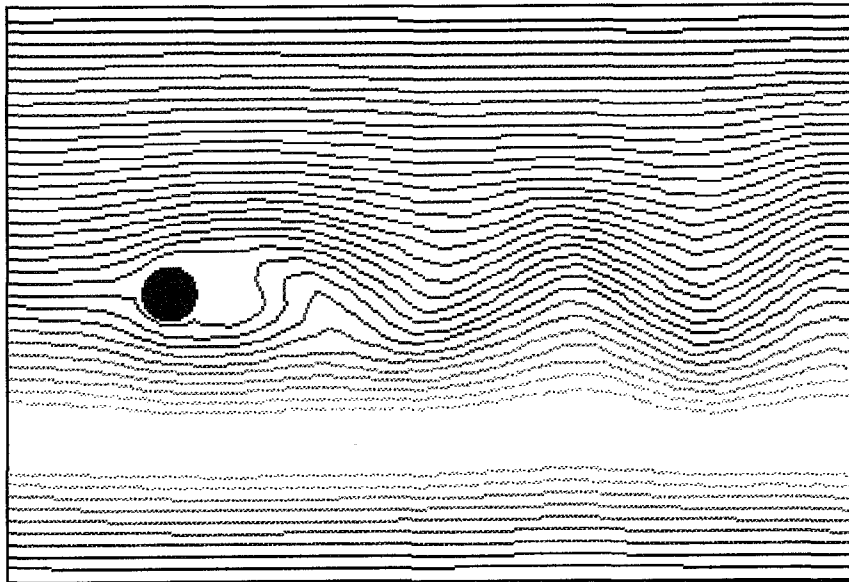


fig. 5

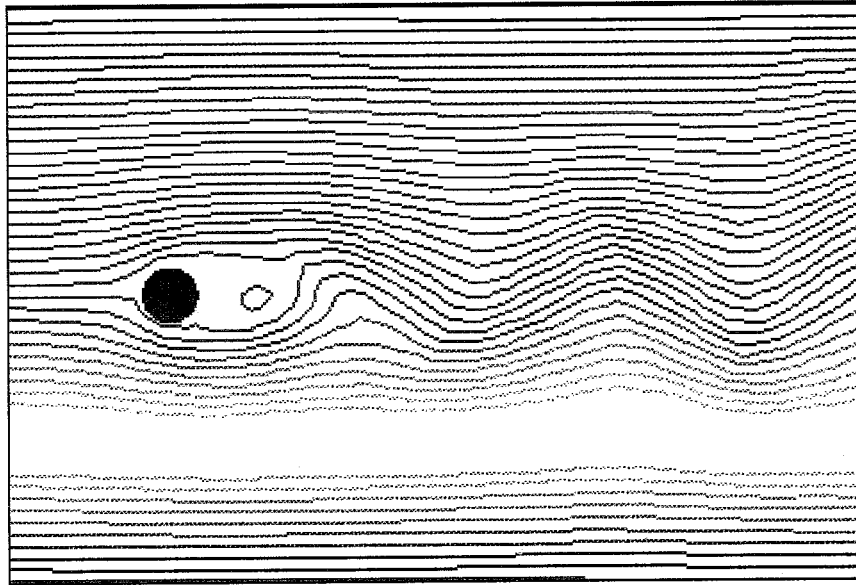
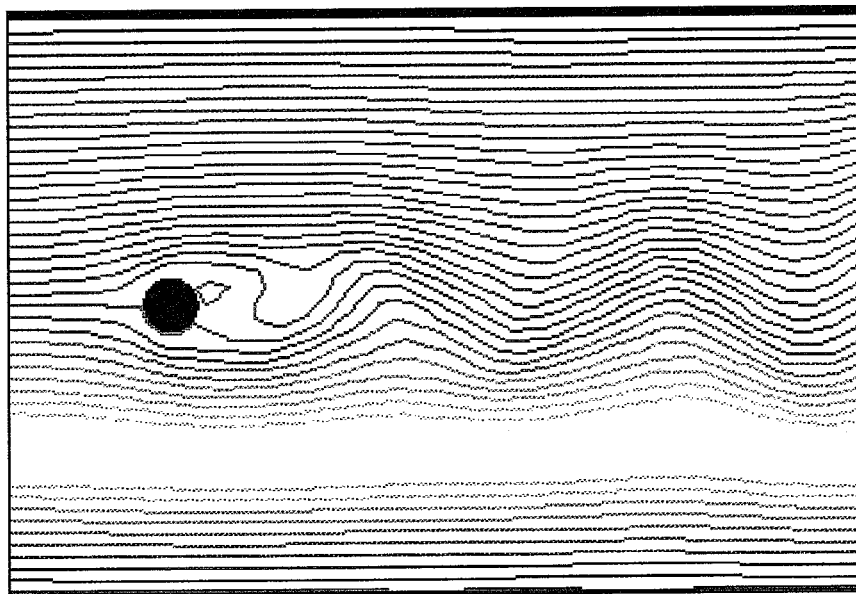


fig. 6



vorticiteit

De vorticiteit is een maat voor de wervelsterkte (rotatie) in de stroming. Bij de vorticiteit kan men dus goed de ontwikkeling en de voortbeweging van de wervels waarnemen. Van de laatste 24 tijdstappen zijn weer 6 stappen genomen die afgedrukt zijn op de volgende pagina's (Fig.1 t/m 6). Er is hier te zien dat de wervels die elkaar volgen van teken verschillen. Dit wil zeggen dat een wervel met positieve rotatie wordt gevolgd door een wervel met negatieve rotatie. Dit is te zien, doordat de achtereenvolgende wervels steeds van rood/paars naar groen/blauw (of andersom) wisselen.

druk

De druk gaat oscilleren met een frequentie van 0,167 als er een verandering in de grootte van de tijdstap optreedt. Schijnbaar is de cn-tijdsintegratie nog niet stabiel genoeg en zouden na de tijdstapveranderingen eigenlijk enige ei-stappen nodig zijn om de verstoring die hierdoor wordt gegenereerd te dempen.

CONCLUSIES

- De methode is goed voor een simulatie van de Von Karman wervelstraat.
Er ontstaat een stroming met $Sr = 0,167$, welke goed overeenkomt met de waarde gevonden door Tritton.
- De drukterm is zeer gevoelig bij de boetefunctie methode. Er treden numerieke oscillaties op na de tijdstap wijziging
- AVS is een goed hulpmiddel voor de visualisatie van een stroming

Fig.1

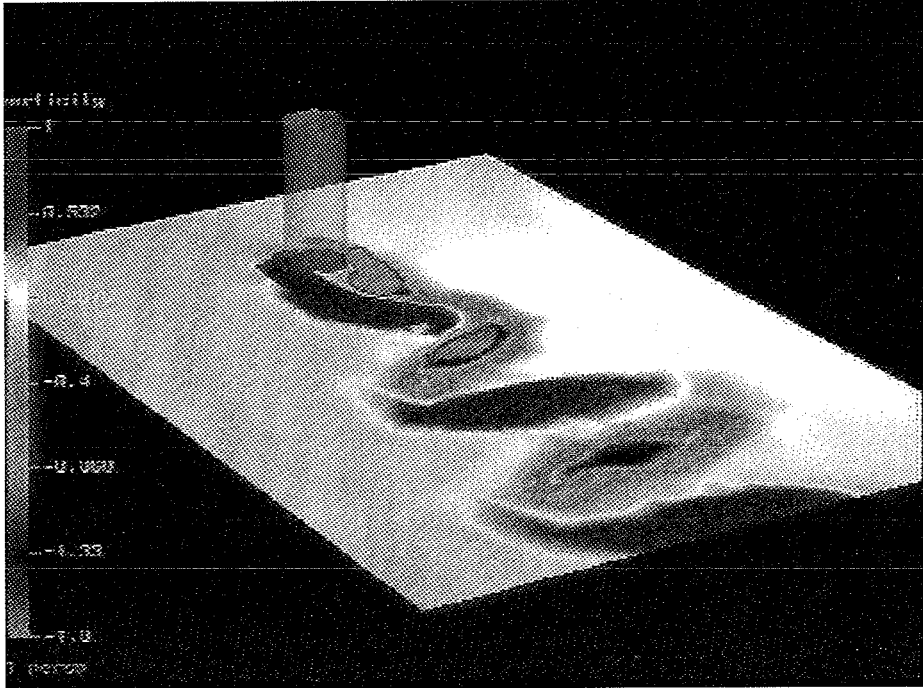
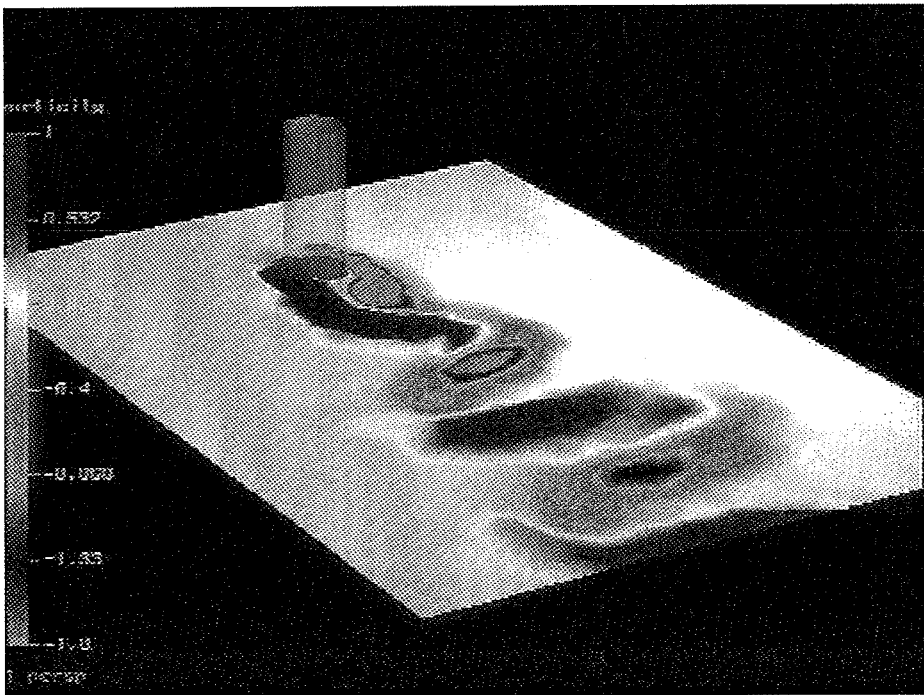


Fig.2



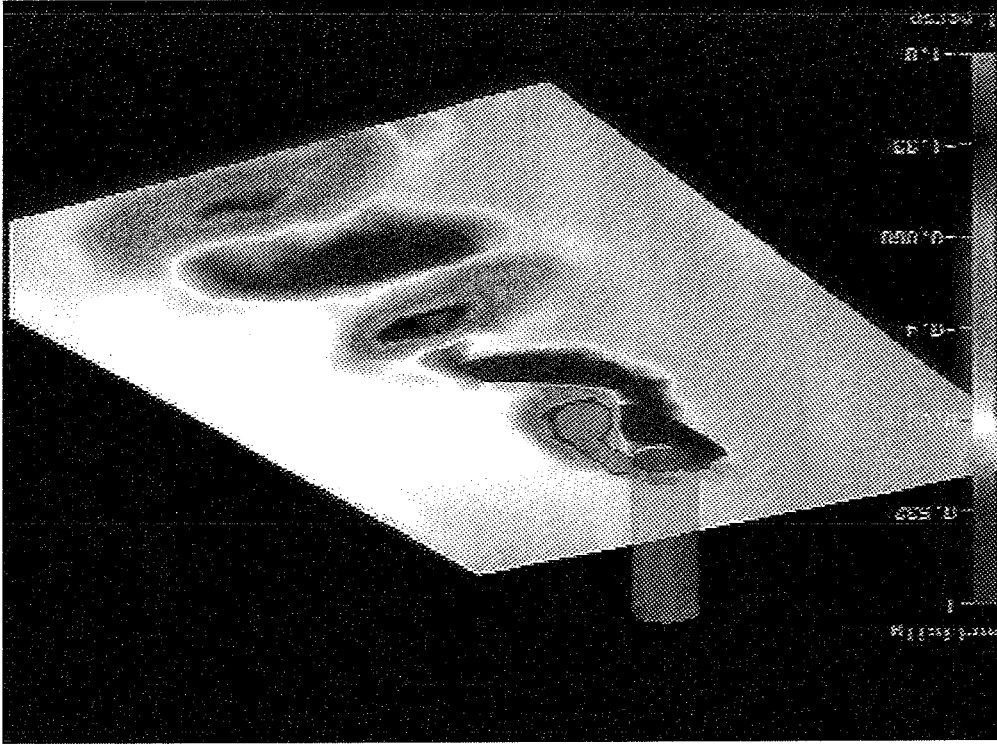


Fig.4

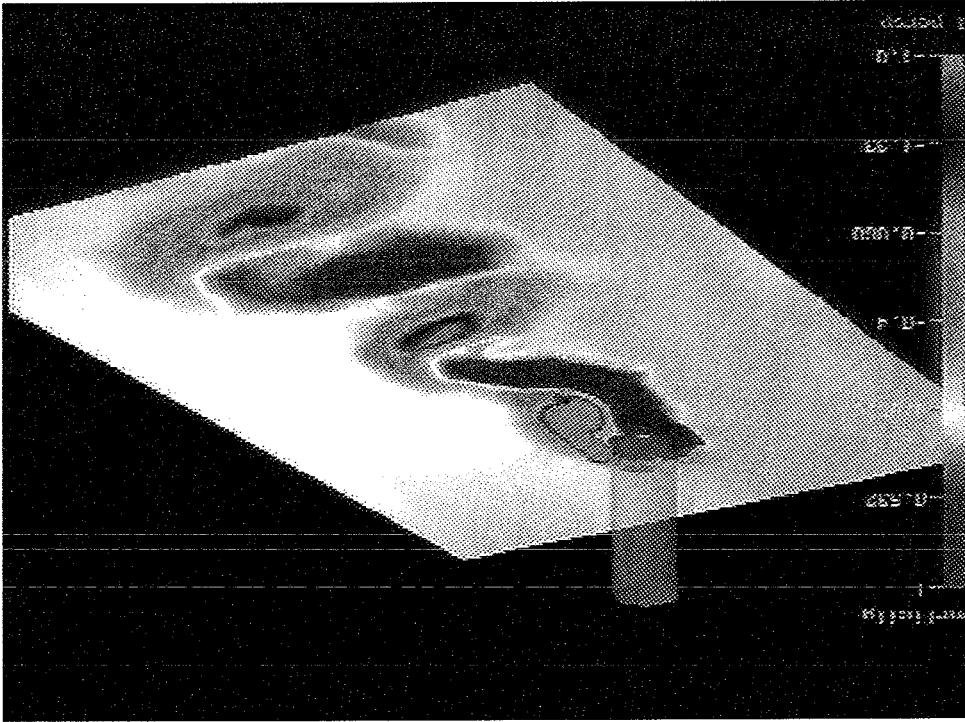


Fig.3

Fig.5

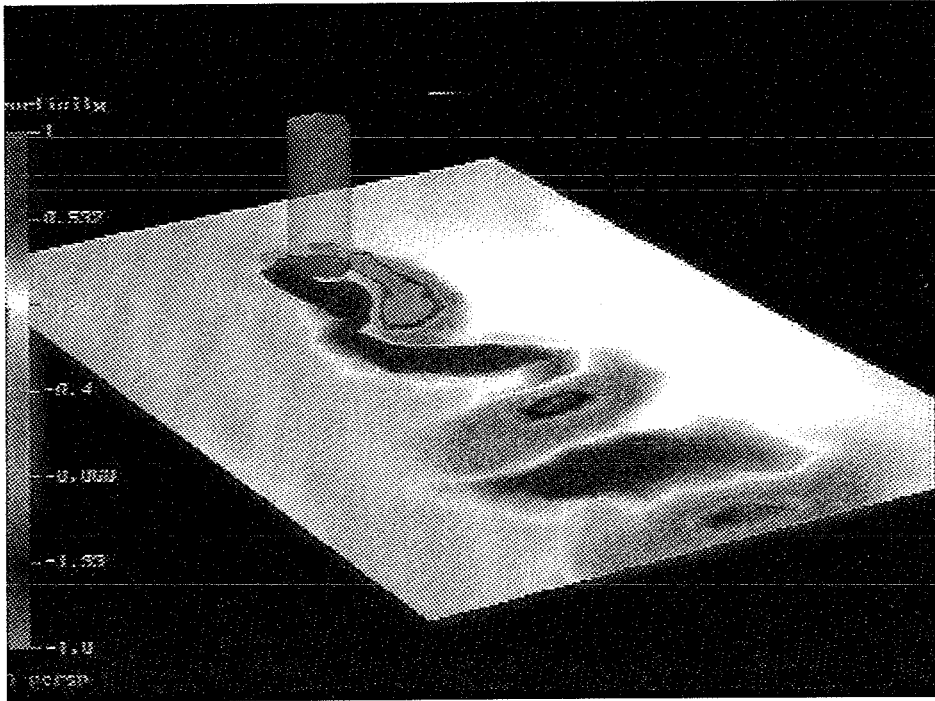
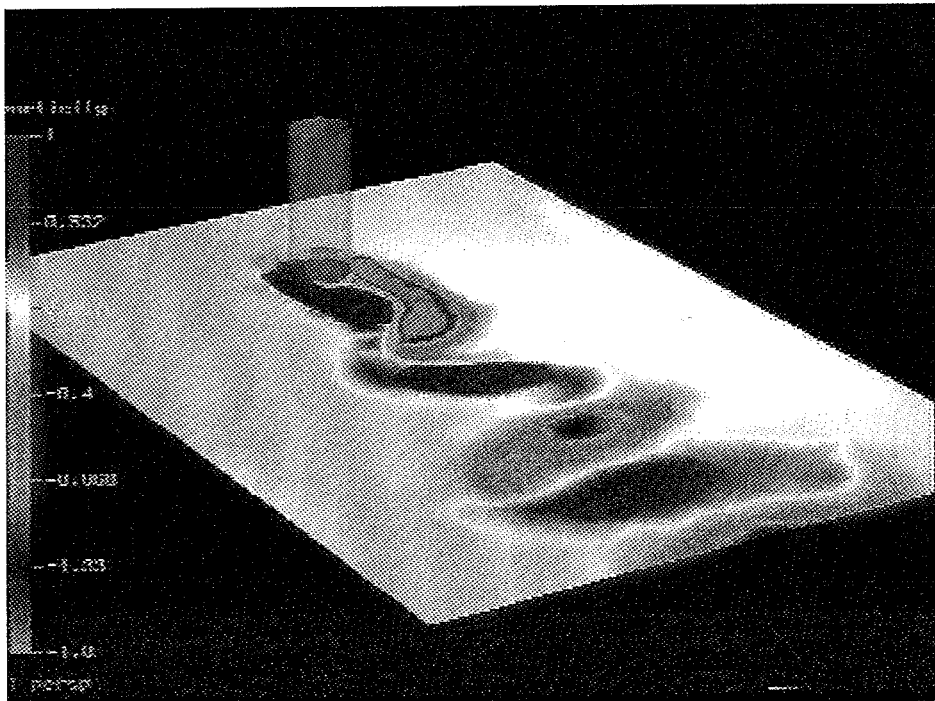


Fig.6



APPENDIX A

De convectie term $N(\underline{u})\underline{u}$ is niet lineair en heeft daarom een speciale behandeling om de vergelijkingen op te kunnen lossen. De iteratie methode van Newton wordt toegepast. Deze levert de volgende vergelijking op :

$$\underline{u}^I \cdot \nabla \underline{u}^I = \underline{u}^{I-1} \cdot \nabla \underline{u}^I + \underline{u}^I \cdot \nabla \underline{u}^{I-1} - \underline{u}^{I-1} \cdot \nabla \underline{u}^{I-1} \quad (A1)$$

De termen uit deze vergelijking heeft de volgende structuur :

$$\sum_{j=1}^2 u_j \frac{\partial u_i}{\partial x_j} \quad i=1,2 \quad (A2)$$

De eerste term van het rechterlid in vergelijking A1 is u_j de reeds bekende snelheid, terwijl in de tweede term u_i de bekende snelheid is. Er wordt aangenomen dat de bekende snelheid op dezelfde manier wordt benaderd :

$$a_i = \sum_{n=1}^N a_{in} \phi_{in} \quad (A3)$$

Hier is $a = u^{I-1}$

Dit geeft voor de eerste term in het rechterlid van A1 :

$$\sum_{j=1}^2 \rho a_j \frac{\partial u_i}{\partial x_j} = \int_{\Omega} \sum_{j=1}^2 \sum_{p=1}^N \rho a_{jp} \phi_{jp} \sum_{n=1}^N u_{in} \frac{\partial \phi_{in}}{\partial x_j} \phi_{ik} d\Omega \quad (i = 1,2, k = 1, \dots, N)$$

$$\text{met } K^i(k, n) = \sum_{p=1}^N \sum_{j=1}^2 a_{jp} \int_{\Omega} \rho \phi_{jp} \phi_{ik} \frac{\partial \phi_{in}}{\partial x_j} d\Omega \quad (n = 1, \dots, N)$$

welk de volgende matrix oplevert :

$$\begin{bmatrix} K^1(a_j) & 0 \\ 0 & K^2(a_j) \end{bmatrix} = D(\underline{u}^{I-1}) \underline{u}^I \quad (A4)$$

Opdezelfde manier wordt de tweede term van het rechterlid van A1 bepaald met behulp van de methode van Galerkin, wat dan oplevert :

$$\sum_{j=1}^2 \rho u_j \frac{\partial a_i}{\partial x_j} = \int_{\Omega} \sum_{j=1}^2 \sum_{l=1}^N \rho u_{jl} \phi_{jl} \sum_{p=1}^N a_{ip} \frac{\partial \phi_{ip}}{\partial x_j} \phi_{ik} d\Omega$$

$$\text{met } C^{ij}(k, l) = \sum_{p=1}^N a_{ip} \int_{\Omega} \rho \phi_{jl} \phi_{ik} \frac{\partial \phi_{ip}}{\partial x_j} d\Omega \quad (i = 1,2, k = 1, \dots, N)$$

welk de volgende matrix oplevert :

$$\begin{bmatrix} C^{11}(a_j) & C^{12}(a_j) \\ C^{21}(a_j) & C^{22}(a_j) \end{bmatrix} = C(\underline{u}^{I-1}) \underline{u}^I \quad (A5)$$

Met behulp van A4 en A5 is vergelijking A1 te schrijven als :
 $N(\underline{u}^I) \underline{u}^I = D(\underline{u}^{I-1}) \underline{u}^I + C(\underline{u}^{I-1}) \underline{u}^I - N(\underline{u}^{I-1}) \underline{u}^{I-1}$

LITERATUURLIJST

- [1] NUMERICAL ANALYSIS OF CAROTID ARTERY FLOW
F.N. van de Vosse (proefschrift)

- [2] A FINITE ELEMENT AND EXPERIMENT ANALYSES OF STEADY AND
UNSTEADY FLOW OVER A STEP
F.N. van de Vosse (interafdelingsproject)

- [3] DE TOEPASSING VAN DE EINDIGE ELEMENTENMETHODE BIJ
STROMINGSPROBLEMEN
A.A. van Steenhoven (dictaat 4K030)

- [4] J. FLUID MECHANICS
D.J. Tritton
gedeelte : Experiments on the flow past a circular cylinder
at low Reynolds numbers

- [5] DE AVS-HANDLEIDING
Stardent

BIJLAGE : GEBRUIK VAN AVS

INLEIDING

Om de uitvoer van een sepran programma goed en overzichtelijk te tonen, kan men gebruik maken van het visualisatie pakket AVS (Application Visualisation System).

Met AVS kunnen bijvoorbeeld de verschillende data uit de tijdstappen achter elkaar worden gezet en als een video worden afgespeeld. Hierdoor krijgt men een goed inzicht in wat er precies in de stroming gebeurt.

Voor het probleem dat in deze studie gevisualiseerd moest worden, is gebruik gemaakt van UCD (Unstructured Cell Data). Hiermee is het mogelijk om een geometrisch voorwerp te definiëren dat bestaat uit discrete elementen met daar aan gekoppelde data. De UCD definities zijn dus uitermate geschikt voor visualisatie van EEM analyses.

In deze bijlage volgt een summiere inleiding in het gebruik van AVS en worden er enkele modules beschreven. Voor de beschrijvingen is gebruik gemaakt van de gedeeltes Module Reference en de AVS user's guide uit de AVS-handleiding [5]. Verder worden er aan het eind van deze appendix nog voorbeelden van netwerken, die in deze studie zijn gebruikt, met een korte beschrijving gegeven

DE NETWORK EDITOR

Om de data met AVS te visualiseren, is het nodig om een netwerk te maken, dat de data kan verwerken. Hiervoor dient de network editor. De network editor kan via het hoofdmenu, welke men na het opstarten van AVS te zien krijgt, gestart worden. Er verschijnt nu een groot werkveld met er boven de modules die kunnen worden gebruikt om het netwerk mee op te bouwen.

De modules zijn onderverdeeld in 4 categoriën, namelijk :

- DATA INPUT : Deze modules introduceren nieuwe data in een AVS netwerk. Sommige lezen de data van de disk, andere creëren data aan de hand van hun input parameters.
- FILTERS : Deze categorie transformeren een numerieke data set in een andere data set.
- MAPPERS : Deze voeren de visualisatie-stap uit. Ze zetten numerieke data om tot een beschrijving van één of meer toonbare objecten, ofwel geometriën ofwel images.
- DATA OUTPUT : Produceren de uiteindelijke uitvoer van het visualisatie-proces.

De netwerken worden in het werkveld opgebouwd uit de verschillende modules. Een module wordt naar het werkveld gebracht, door met de muis op de gewenste module te gaan staan. Door de linkerknop in te drukken en ingedrukt te houden, terwijl met "de muis" naar het werkveld wordt geschoven. Na het loslaten van de knop verschijnt op die plaats in het werkveld de geselecteerde module. Als naderhand de module nog verplaatst moet worden, dan gebeurt dit op dezelfde manier.

Op de module staan gekleurde vlakjes. De vlakjes aan de boven-

kant van de module zijn de ingangspoorten, waarlangs de data wordt ingevoerd, en aan de onderkant zijn de uitgangspoorten, waarlangs de door de module verwerkte data wordt uitgevoerd.

De poorten hebben niet allemaal dezelfde kleur. Het is alleen mogelijk om een uitgangspoort met een ingangspoort te koppelen als ze dezelfde kleur hebben. Door op een uitgangspoort te gaan staan met behulp van de muis en de middelste knop in te drukken, verschijnen alle mogelijke verbindingen met de andere modules in het werkveld. Om de gewenste verbinding te verkrijgen, moet de muis met ingedrukte knop naar de gewenste verbindinglijn worden bewogen. Als deze lijn oplicht, kan de knop losgelaten worden en zal de geselecteerde verbinding blijven staan.

Een verbinding tussen twee modules verbreken gaat op dezelfde manier alleen moet met de rechter knop van de muis worden gewerkt.

Om een module van het veld te verwijderen, moet men deze verplaatsen naar de hamer rechtsonderin het veld.

MODULES UIT AVS

DATA INPUT

READ UCD : Deze module leest de UCD data van de disk. Als de read UCD wordt gekozen, dan verschijnt er een window links op het scherm waarin kan worden aangegeven welke file moet worden ingelezen. De naam van deze file heeft een verplichte extensie ".inp".

Verder kan het een ASCII of binary file zijn. De vorm van de ASCII-file moet als volgt zijn :

- Op de eerste regel moeten achtereenvolgens het aantal knooppunten en elementen en het aantal datawaarden per knooppunt, het aantal data per element en het aantal data per model worden weggezet.
- Op de volgende regels worden onder elkaar de coördinaten van de knooppunten weergegeven.
- Daarna moeten de elementtype en de connectiviteit van de knooppunten die een element vormen worden opgegeven. Dit gebeurt door op een regel het volgende te zetten :
elementnummer---materiaaldata (meestal voor ieder element gelijk (1))---elementtype---volgorde van de knooppunten op een element.
- Op de regel na de coördinaten wordt opgegeven uit welke vorm de verschillende data bestaat. (vectoren of scalars) Dit wordt als volgt genoteerd : Eerst wordt opgegeven hoeveel grootheden in de knooppunten zijn gedefinieerd. Daarna wordt per grootheid aangegeven uit hoeveel componenten deze bestaat. Bijvoorbeeld als de data bestaat uit 1 vector en 3 scalars dan zijn er 4 grootheden per knooppunt. Het volgende cijfer is een 3 om aan te geven dat de eerste 3 componenten samen een vector vormen. Daarna volgen nog 3 énen om aan te geven dat de volgende 3 array's alle een scalaire grootheid zijn.

P.S. read UCD kent alleen 3-D vectoren

- Daarna staan onder elkaar de namen van de data componenten met hun eenheid erachter vermeld. Voorbeeld :
velocity, m/s
distance,m
- Tot slot worden de datacomponenten naast elkaar ingevoerd.

Een voorbeeld van een ASCII-file :

```
1732 832 6 0 0
```

(1732 knoopp.,832 elem.,6 datawaarden per knpnt, geen data per element of model)

```
1 x y z
```

(knppntnr,coördinaten)

```
1732 x y z
```

```
1 1 tri 1 2 3 4 5 6
```

(elemnr, mat.data, elem.type, connectiviteit knooppunten)

```
832 1 tri 126 127 128 137 138 139
```

```
4 3 1 1 1
```

(4 datacomponenten, 1 vector, 3 scalars)

```
velocity,m/s          (naam vector/scalar, eenheid)
strfunct,m2/s
pressure,N/m2
vorticity,/s
```

```
1 0.2 0.3 0.0 100 400 12
```

*(knppntnr, snelheidsvector, stroomfunctie, druk en vortici-
teit)*

```
1732 1.0 0.0 0.0 50 130 0
```

GENERATE COLORMAP : Zorgt voor de uitvoer van de AVS kleurenkaart. De data structuur van de kleuren kaart kan gebruikt worden door modules die de invoer data in kleurwaarden transformeert (b.v. "UCD contour"). De uitvoer van generate colormap wordt gebaseert op de status van de kleurenkaart, welke wordt aangeroepen met behulp van de edit button in de controle paneel. Het bovenste gedeelte van de controle paneel dient voor het koppelen van de invoerwaarden aan de uitvoer kleuren. Standaard wordt de grootste waarde aan blauw en de kleinste waarde aan rood gekoppeld. (Of andersom, dit is in te stellen met de inverse-button). Een andere kleurenpatroon is in te voeren door b.v. een gedeelte van het bereik (FROM TO) op te geven en eventueel ook de kleureninstelling te veranderen. Standaard loopt deze van 0.0 naar 1.0 (d.w.z. van rood naar blauw) te veranderen in b.v. 0.0 naar 0.2 (d.w.z. van fel rood naar oranje). Na gebruik van Do Interpolation wordt het opgegeven gedeelte in de nieuwe kleuren verandert, waardoor er meer van de stroming is te zien.

Voorbeeld : Als de invoerwaarden reiken van 0 tot 175, maar de meeste waarden liggen tussen 160-165, dan geeft dit in het plaatje een vlak die voor het grootste gedeelte eenzelfde kleur heeft. Er zal in eerste instantie worden geconcludeerd dat er weinig in de stroming gebeurd. Dan is het wenselijk om de waarden in het gebied met een contrasterende kleur ten opzichte van zijn omgeving te kleuren. Dit wordt op de volgende manier verkregen :

- * zet muis op From en tik 160 in
- * ga daarna naar To en zet deze op 165

Om het contrasterende patroon te verkrijgen :

- * die van From value op b.v. 0.0 (fel rood)
- * die van To value op 0.1 (rood).

Terwijl dit gebied eerst b.v. groen gekleurd was. Druk daarna op de Do Interpolation button.

READSEQANIM : Zorgt ervoor dat read UCD meerdere files (unsupported) automatisch inleest, zodat het netwerk die kan verwerken. Bij het in gebruik nemen van de module verschijnt er meteen een window. Waarin, bovenin, kan worden aangegeven welke file als eerste moet worden ingelezen. Onderin staan 2 wijzerplaten. In de rechterdial kan worden aangegeven hoeveel frames er moeten worden ingelezen (iedere file wordt daarbij omgezet tot een frame). In de linker dial wordt tijdens het achterelkaar verwerken van de files, aangegeven hoeveel files "read UCD" al heeft ingelezen. Het inlezen wordt gestart door op de button Anim. te drukken. Readseqanim kan alleen werken met files waarvan de namen de volgende vorm hebben : Naam.00X.inp tot Naam.XXX.inp..

Omdat "Read UCD" geen ingangspoort heeft, moeten, alvorens "readseqanim" kan worden verbonden met "read UCD", eerst de volgende handelingen worden verricht :

- * De muis op het vierkantje in module "read UCD" zetten en de rechter knop van de muis indrukken.
- * Er verschijnt een window. Daarin zie je onder andere Parameters en eronder read file staan. Je moet read file aan klikken.

- * Er verschijnt weer een nieuw window. Als je daar Port Visible aanklikt, zodat deze oplicht, creëer je op de module een ingangspoort. Je kunt nu beide windows sluiten.

Je kunt nu readseqanim verbinden met read UCD.

FILTERS

UCD EXTRACT : Leest een UCD-structuur in, die verschillende grootheden op de knooppunten heeft en voert een structuur uit met maar 1 grootte op elke knooppunt. Elk van deze grootheden kan een vector of scalar zijn. UCD extract "isoleert" elke afzonderlijke grootte van de knooppuntdata, of het nu een vector of een scalar is. Als hij een vectorcomponent selecteert dan neemt hij de gehele vector en niet een enkele component uit een vector. Je selecteert met de node-data welke grootte je wilt zien.

UCD EXTRACT SCALAR : Doet hetzelfde als UCD extract, maar deze selecteert scalars (ook uit een vector). Welke component van de vector wordt geselecteerd, wordt gespecificeerd in de module "UCD contour".

UCD EXTRACT VECTOR : Neemt een vectorveld van elke dimensie, coördinaatsysteem of datatype en pakt een gedeelte van de vectorelementen op elk knooppunt. Maakt van drie scalars een vector.

UCD VECMAG : Berekend de grootte van een vectorveld. De module accepteert alleen een vector als input en berekend de grootte per knooppunt. Er is geen controle paneel voor "UCD vecmag".

ANIMATE LINES : Neemt een aantal stroomlijnen, die berekend zijn door "UCD streamlines" (zie MAPPERS) en maakt er een animatie van. Hiervoor voert het een aantal achterelkaar volgende delen van de stroomlijnen uit om een dynamische representatie ervan te maken. Het spreekt voor zich dat dit maar voor één frame per keer kan worden uitgevoerd. Met de dial kun je bepalen hoe lang de lijnsegmenten worden.

TUBE : Zet lijnen om in cilindrische pijpen. De straal van de pijpen kan worden ingesteld. Deze module kan b.v. worden gebruikt om de isolijnen beter te laten zien.

MAPPERS

UCD CONTOUR : Hierdoor wordt een lijst van waarden van kleuren gekoppeld met de UCD. "UCD contour" wordt gebruikt om een kleurencontour van de UCD structuur te creëren. De uitvoer moet naar de module "UCD TO GEOM" worden gestuurd om een gekleurde presentatie van de structuur te produceren. De module kan alleen die waarden van knooppuntscomponenten aan, die in de vorm van scalars worden aangeboden.

Als er zowel vectoriële als scalaire grootheden zijn, zal de module alleen de scalaire grootheden tonen. Met behulp van de "node data" kan men de scalar-data kiezen die men wilt laten zien.

UCD-contour schaaft de colormap naar de range van de waarden van de knooppuntscomponent die is geselecteerd. Met andere woorden, de laagste knooppuntswaarde in de structuur aanwezig krijgt de kleur met de laagste colormap waarde en andersom. Deze instelling kan in "generate colormap" worden veranderd.

UCD HOG : Laat de waarden van een vector in een UCD knooppunt als lijnsegmenten in de 3D-ruimte zien.

De module werkt alleen op vector componenten. Hij interpreteert de 3 waarden van de vector als de x,y,z waarden in de ruimte en toont deze als kleine lijnsegmenten met bepaalde lengte, richting en kleur. Als er meerdere vectoren aanwezig zijn, kan men de gewenste vector selecteren m.b.v. de node data. UCD hog gebruikt de invoer kleurenkaart uit generate colormap om aan elke lijnsegment vector een kleur te geven gebaseerd op de grootte van de vector. UCD hog laat niet standaard de vector op ieder knooppunt zien. Dit wordt geactiveerd door op de button nodes te drukken. Je kunt de grootte van de vector ook zelf bepalen door de schaal te veranderen (door de wijzer te draaien).

UCD ISO : Genereert een iso-oppervlak voor een UCD structuur met scalaire knooppuntdata. De structuur moet op z'n minst 1 scalaire grootheid in zijn knooppuntdata hebben. Met de node data kan men de component selecteren waarvan het iso-oppervlak moet worden getoond. Om het oppervlak te kleuren moet "UCD-iso" een input van de module "generate colormap" ontvangen. Als uit de invoer meer dan één scalaire grootheid wordt geselecteerd, dan kan men de buttons beneden de map scalar gebruiken om de componentwaarden te selecteren die kunnen worden gebruikt voor de bepaling van de kleur voor het oppervlak.

Voorbeeld : 3 scalars : temperatuur, druk, dichtheid
Berekend dan het iso-oppervlak voor een gegeven temperatuur. Intuïtief kleurt men dit gebaseerd op de temperatuurvariabele. Echter, het is ook mogelijk om het temperatuur oppervlak te kleuren met gebruik van de waarden van de druk of de dichtheid, welke dan de druk of de dichtheid aangeeft voor een vaste temperatuur.

UCD ISOLINES : Genereert de isolijnen voor een vlak uit de UCD structuur met scalar knooppuntdata. Met de bovenste dial kan worden aangegeven hoeveel stroomlijnen men in de frame wilt laten zien. Met de onderste dials wordt aangegeven over welke gedeelte van de structuur de isolines worden getoond.

UCD STREAMLINE : Genereert stroomlijnen voor een UCD structuur met de vector knooppunt data. De berekening van de stroomlijnen wordt gestart met de button Start stream. "UCD streamline" werkt alleen bij structuren die een vectoriële grootheid bezitten in hun knooppuntdata. Met de node data kan weer bepaald worden welke grootheid men wil gebruiken. De orde van integratie kan worden opgege-

ven.

UCD TO GEOM : Zet de UCD structuur om in een AVS geometrie. De module neemt zowel de invoer voor de knooppuntcoördinaten als de lijst met de connectiviteit van de knooppunten, die verteld welke knooppunten moeten worden verbonden om een element te vormen. Elk element wordt op deze wijze gedefinieerd en omgezet in de geometrie format, welke wordt toegevoegd aan de geometrie object die de module uitvoert. Men kan zelf bepalen of de externe oppervlakte of de externe randen, dan wel het gehele oppervlakte van de elementen worden getoond. De elementen kunnen ingekrompen worden m.b.v. de shrink factor. Hierdoor worden alle elementen kleiner en komen ze van elkaar af te liggen, waardoor men de vorm van de elementen goed kan zien.

UCD LEGEND : Heeft 2 functies :

- * Ten eerste om de UCD te kleuren. Hiervoor pakt het een AVS kleurenkaart en voert een array van kleuren uit (voor elk knooppunt in de structuur).
- * Daarnaast creëert "UCD legend" een kleuren"legend" widget die de UCD data aan een kleurenschaal relateert. De widget toont de invoer kleurenkaart als een horizontale spectrum. Onder de kleurentabel print UCD legend het bereik van de knooppuntwaarden van de UCD structuur. De kleurenlegend laat dus zien welke kleur elke waarde representeert. UCD legend werkt met modules die een structuur nemen en je toestaan om delen van de data of specifieke waarden in de data te visualiseren (zoals bijv. "UCD iso"). Met UCD legend kan men de subset specificeren met numerieke waarden of met het kleurenbereik (bv. van groen naar blauw). Gebruik van UCD legend is meestal gebaseerd op intuïtie en werkt alleen op scalaire knooppuntsgrootheden.

UCD RSLICE : Snijdt door een UCD structuur langs een willekeurig gekozen snijvlak. "UCD rslice" voert de structuur uit zonder de weggesneden gedeeltes. Om het snijden te starten moet men de Do Slice button indrukken. "UCD rslice" snijdt hierbij dwars door de elementen die in het snijvlak liggen. Elke keer als het snijvlak wordt geheroriënteerd wordt Do Slice afgezet. Het snijvlak kan voor en achterwaarts door de structuur worden bewogen langs de normaal van het vlak door de Distance floating point dial te gebruiken. De oriëntatie van het vlak wordt gecontroleerd door de 2 floating point parameter dials, x-rot en y-rot.

DATA OUTPUT

GEOMETRY VIEWER : Toont de geometrie in een window. Door op het vierkantje in de module met de linker button op de muis te klikken, krijgt men links op het scherm het controlepaneel te zien. Het voert te ver om dat hier te behandelen.

IMAGE VIEWER : Toont de data beelden en geometriën die door het netwerk zijn geproduceerd. Het controle paneel verkrijgt men op dezelfde manier als bij de Geometry viewer. Bij menu selections zijn 5 submenu's aanwezig, die elk hun eigen unieke regelgedeelte hebben. Hier wordt alleen behandeld hoe de verschillende tijdstappen (files) achterelkaar kunnen worden gezet en worden opgeslagen, om daarmee een animatie te krijgen.

Hiervoor moet je submenu Action aanroepen. In het nu verschenen submenu moet store frames aangezet worden. Hierdoor zal iedere frame achterelkaar worden opgeschreven. Men moet hier wel bij oppassen dat iedere verandering aan een frame als aparte frame wordt opgeslagen (Ook als je bv. de frame verplaatst in de window of verkleind, end.) Als alle geselecteerde files zijn afgehandeld, kan men de animatie starten door op continous te drukken. De files worden nu achterelkaar afgelopen, waarbij hij aan het eind automatisch weer van voor af aan begint. (bij bounce loopt de animatie van voor naar achter en weer terug).

IMAGE TO POSTSCRIPT : Maakt een postscript file van de image. De module werkt pas als er een naam is opgegeven. Er is verder een keuzemogelijkheid om de image in kleur dan wel grijstint weg te schrijven. Tevens kan de grootte van de afbeelding, zoals hij moet worden uitgeprint, worden opgegeven.

DE NETWERKEN

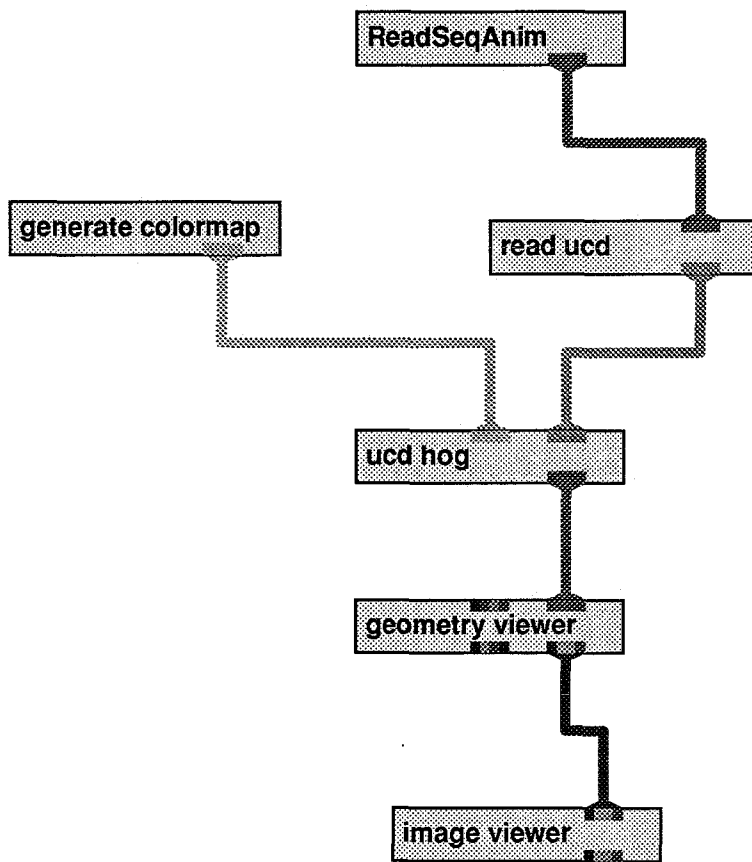
In deze studie zijn 6 netwerken gebruikt om de stroming te visualiseren. Op de volgende pagina's staan ze afgebeeld :

- netwerk 1 : met dit netwerk laat je snelheidsvectoren op de knooppunten zien. Hierbij geven de lengte en de kleur van de vector, de grootte van de snelheid.
- netwerk 2 : hiermee kleur je de geometrie per scalar (ook de vector ontbonden in scalars). Door de keuze kan men de scalars vorticiteit, druk, stroomfunctie bekijken. Tevens kunnen de snelheidscomponenten apart worden bestudeerd in deze studie.
- netwerk 3 : berekent de grootte van de snelheidsvector en voert deze als scalar uit, waarop de geometrie gekleurd wordt.
- netwerk 4 : laat een animatie zien van de stroomlijnen per frame. Hiermee kan men op een bepaald tijdstip zien, hoe de stroming gaat.
- netwerk 5 : laat de isolijnen zien van de scalars. Dat zijn in deze studie, dus de isolijnen van de druk, vorticiteit en de stroomfunctie.

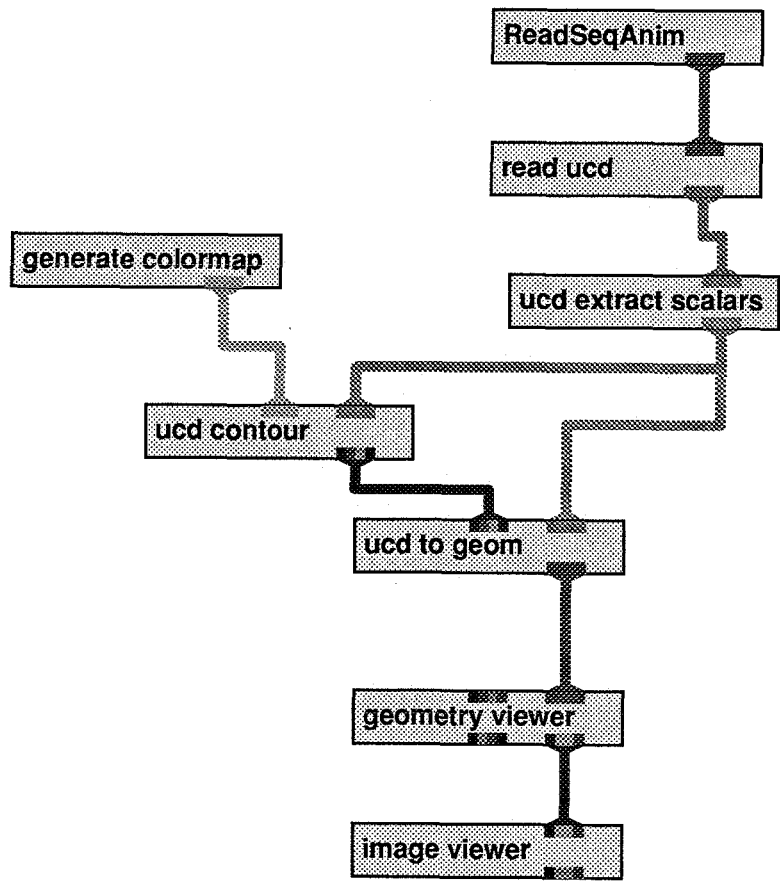
P.S. : Dit netwerk bevat een paar modules die hiervoor niet zijn beschreven. Deze modules zijn gebruikt bij het maken van de video.

- netwerk 6 : Laat hetzelfde zien als netwerk 4, alleen worden nu particles (blokken) en geen delen van de stroomlijnen gebruikt. Hierdoor is het netwerk simpeler.

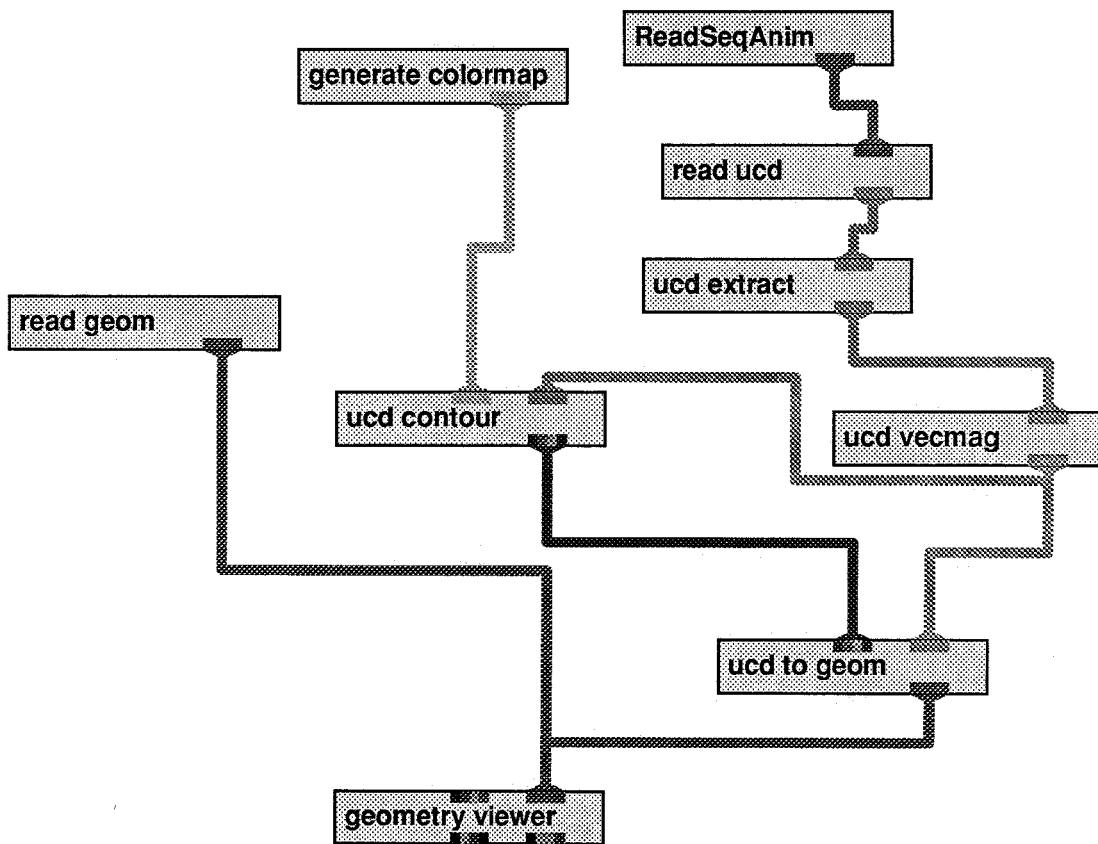
NETWERK 1



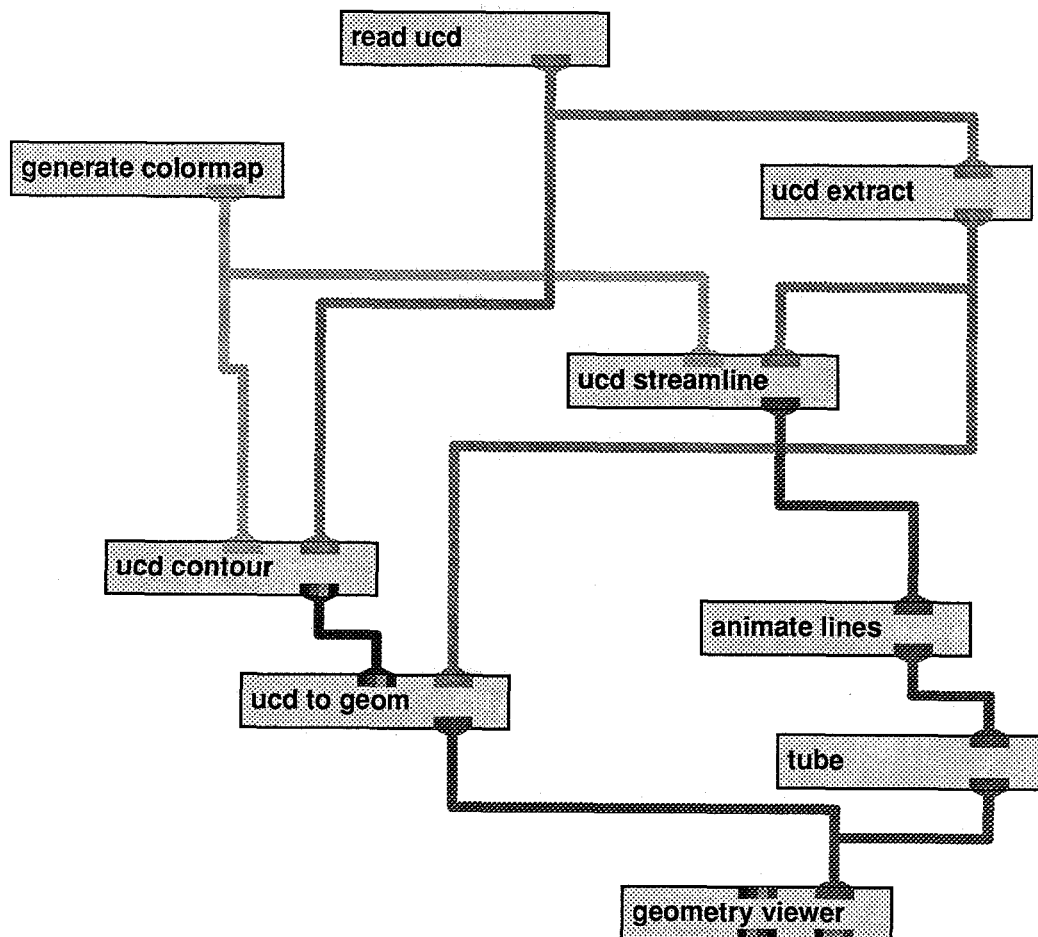
NETWERK 2



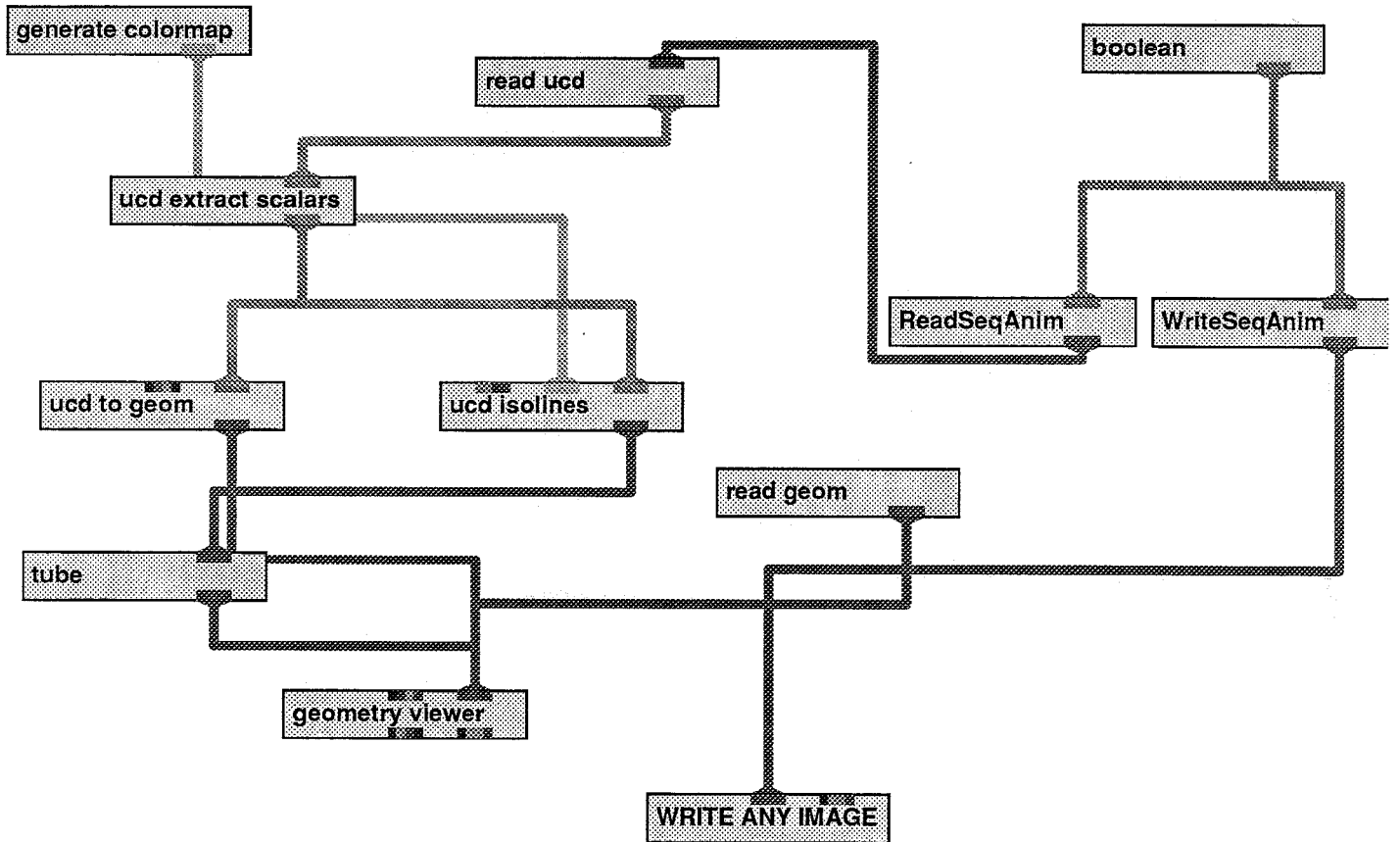
NETWORK 3



NETWORK 4



NETWORK 5



NETWORK 6

