Practicumverslag voor Advectie-Dispersie transport

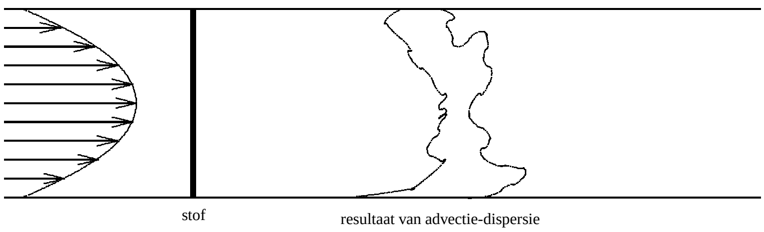
Remko Moerkerken, 9575336, remkom@ch.twi.tudelft.nl
Sebastiaan, 9121164, sebastia@ch.twi.tudelft.nl

23 juni 2003

1 Inleiding

Wanneer er in een stromend medium (bijvoorbeeld een rivier) een oplosbare stof wordt geloosd dan is deze stof onderhevig aan 2 fenomenen: advectie en dispersie. Zie voor een schematische voorstelling figuur 1. De advectie zorgt ervoor dat de stof meegevoerd wordt met de stroming van de rivier. Dit kan volgens een stromingsprofiel: in een rechte rivier stroomt het water in het midden over het algemeen sneller dan aan de kant.

Het tweede fenomeen is dispersie, wiskundig gezien hetzelfde als diffusie. Dispersie zorgt ervoor dat de stof zich niet alleen in de stromingsrichting verplaatst maar zich over de gehele nabijgelegen vloeistof verspreid.



Figuur 1: Advectie en dispersie van een stof in een rivier

2 Theorie

Voor dit practicum beschouwen wij het 2-dimensionale model van de advectie-dispersie vergelijking:

$$\frac{\partial C}{\partial t} + \frac{\partial (UC)}{\partial x} + \frac{\partial (VC)}{\partial y} = D_x \frac{\partial^2 C}{\partial x^2} + D_y \frac{\partial^2 C}{\partial y^2} \tag{1}$$

Hierin zijn D_x en D_y de dispersiecoëfficienten en U en V de snelheden in de $x-$ en $y-$richting, gemiddeld over de diepte. We noemen $\frac{\partial (UC)}{\partial x}$ en $\frac{\partial (VC)}{\partial y}$ de advectieve termen. De rol van de dispersie is afhankelijk van de grootte van D_x en D_y. Zijn deze coëfficienten klein dan zal de concentratieverdeling veel op het snelheidsprofiel lijken. We versimpelen het model nog verder door aan te nemen dat de transversale snelheid V zo klein is dat deze verwaarloosd kan worden.

3 Model

Het oplossen van de vergelijking 1 gebeurt zelden analytisch. Wanneer wij het verloop van de stof in de rivier willen volgen kunnen we overgaan op een deeltjesmodel. We delen de stof op in N deeltjes (die, voor het gemak, elk voor $\frac{1}{N}$de van de totale massa staan). We geven een deeltje aan met $P_t(X_t, Y_t)$. We nemen aan dat een deeltje zich onafhankelijk in de $x-$ en $y-$richting kan bewegen.

Om de dispersie te modelleren gaan we uit van een stochastische wandeling. We beschouwen in de afleiding alleen de $x-$richting. Voor de $y-$richting gaat het analoog.

$$X_{t+\Delta t} = X_t + \Delta X + Z \tag{2}$$

Hierbij is ΔX het advectieve transport, hier kunnen we $\Delta X = U \Delta t$ nemen. Z stelt de dispersie voor als willekeurige verplaatsing.

De analytische oplossing van de eendimensionale transportvergelijking is

$$C(x,t) = \frac{C_0}{\sqrt{4\pi D_x t}} e^{-\frac{(x-Ut)^2}{4D_x t}} \tag{3}$$

Hierin is C_0 de beginconcentratie en D_x de dispersiecoëfficient. We zien dat deze oplossing eenzelfde vorm heeft als de functie van een normale verdeling

$$f_X(x) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{4}$$

met de volgende substituties:

$$\begin{aligned} f_X(x) &= \frac{C(x,t)}{C_0} \\ \mu &= Ut \\ \sigma &= \sqrt{2D_x t} \end{aligned} \tag{5}$$

We kunnen nu de dispersie zien als een normaal verdeelde stochast (deze verdeling wordt door de advectie meegevoerd met snelheid U dus heeft de dispersie gemiddelde 0). De variantie van de dispersie is dus gelijk aan $2D_x t$. Dit geeft aanleiding om de dispersie Z te beschouwen als

$$Z = \mathscr{N}(0, 2D_x t) = \sqrt{2D_x t}\,\mathscr{N}(0,1) \tag{6}$$

De deeltjes bewegen dan alsvolgt door de rivier heen:

$$\begin{aligned} X_{t+\Delta t} &= X_t + U\Delta t + \sqrt{2D_x \Delta t}\,\mathscr{N}(0,1) \\ Y_{t+\Delta t} &= Y_t + \sqrt{2D_y \Delta t}\,\mathscr{N}(0,1) \end{aligned} \tag{7}$$

Wanneer een deeltje tegen de kade opbotst (de nieuwe positie dreigt buiten de oevers te liggen) dan leggen we het deeltje op de kant. Bij de volgende stap kan het dan weer meegenomen worden door de dispersie en de stroming. Dit is niet helemaal volgens de realiteit (het effect is minimaal) maar om de implementatie niet onnodig ingewikkeld te maken hebben we het hierbij gelaten.

4 Simulatie

We hebben dit model geïmplementeerd voor een rivier van 30 meter breedte met 1000 deeltjes en het verloop 40000 seconden gevolgd. Voor onze implementatie kan gekozen worden welke dispersiecoëfficienten gebruikt worden. Ook kunnen we kiezen voor het niet-uniforme stromingsprofiel waarin $U = \frac{3y(b-y)}{b^2}$, zie tabel 2. In het begin zijn alle deeltjes uniform over de breedte van de rivier verdeeld.

Voor alle gevallen kunnen we twee plaatjes maken, één die de rivier weergeeft met daarin de locatie van de deeltjes. In de andere is de concentratie van de deeltjes gemiddeld over de $y-$richting over een breedte van 300 meter te zien. We hebben 4 tijdstippen weergegeven, zie tabel 1. Het programma is ook in staat om een filmpje te maken of om het verloop van de deeltjes direct te volgen. Zo'n filmpje is te downloaden op *http://zwartoog.kaioken.nl/*.

rood	100 s
groen	2000 s
magenta	20000 s
blauw	40000 s

Tabel 1: Kleuren en tijdstippen

D_x $\left(\frac{m^2}{s}\right)$	D_y $\left(\frac{m^2}{s}\right)$	Uniform	figuur	pagina
10^{-2}	0	Ja	2	6
10^{-2}	10^{-4}	Ja	3	6
10^{-2}	0	Nee	4	7
10^{-2}	10^{-4}	Nee	5	7
10^{-1}	0	Ja	6	8
10^{-1}	10^{-2}	Ja	7	8
10^{-1}	0	Nee	8	9
10^{-1}	10^{-2}	Nee	9	9
10^{-1}	10^{-3}	Nee	10	10
10^{-2}	10^{-3}	Nee	11	10
10^{-3}	10^{-3}	Nee	12	11
10^{-4}	10^{-4}	Nee	13	11
0	10^{-4}	Nee	14	12

Tabel 2: Bekeken parameters

Wanneer we een uniform stroomprofiel hebben met alleen dispersie in de stromingsrichting dan zien we dat de deeltjes zich slechts matig dispersiëren, zie figuur 2.

Als we ook dispersie in de $y-$richting geven dan zien we nog weinig verschil, zie figuur 3. Door de hoge concetratie van de deeltjes is niet te zien dat sommige een verplaatsing ondergaan hebben in de transversale richting.

Met stromingsprofiel zonder dispersie in de $y-$richting krijgen we figuur 4. We zien hier duidelijk het stromingsprofiel in terug. De deeltjes in het midden van de rivier hebben een grotere snelheid dan aan de rand.

Pas wanneer we ook dispersie in de $y-$richting toelaten dan zien we een natuurlijk te verwachten advectie-dispersie patroon, figuur 5. Nog steeds heeft de deeltjewolk duidelijk het stromingsprofiel, maar we zien de oneffenheden door de dispersie in beide richtingen. De hoogste concentratie blijft zich op de kop bevinden, de rest van de deeltjes wordt rustig naar de kant gewerkt waar ze minder snelheid hebben.

In figuren 6-9 hebben we beide dispersiecoëfficienten fors verhoogt om het effect hiervan te onderzoeken. Het blijkt dat de invloed van D_x niet voor een grote verandering in het plaatje zorgt. Dit komt doordat de stroming in deze richting dominant is.

In figuur 9 daarentegen hebben we een ook relatief grote dispersie in de $y-$richting gekozen. Hierdoor blijven de deeltjes niet lang aan de rand plakken zodat ze verder naar het midden van de rivier dispersiëren en zodoende verder door de stroming meegevoerd worden. De deeltjes die zich in het midden van de stroming bevinden worden door dezelfde reden vertraagd. Het stromingsprofiel wordt als het ware uitgemiddeld.

We zagen in figuur 9 een grote verspreiding van de deeltjes door de vloeistof en wezen de oorzaak hiervan voornamelijk toe aan D_y. In figuur 10 hebben we D_y kleiner genomen. We zien nu het stromingsprofiel weer duidelijk terug.

Met figuur 11 is te zien dat de invloed van D_x klein is aangezien dit figuur sterk lijkt op de vorige.

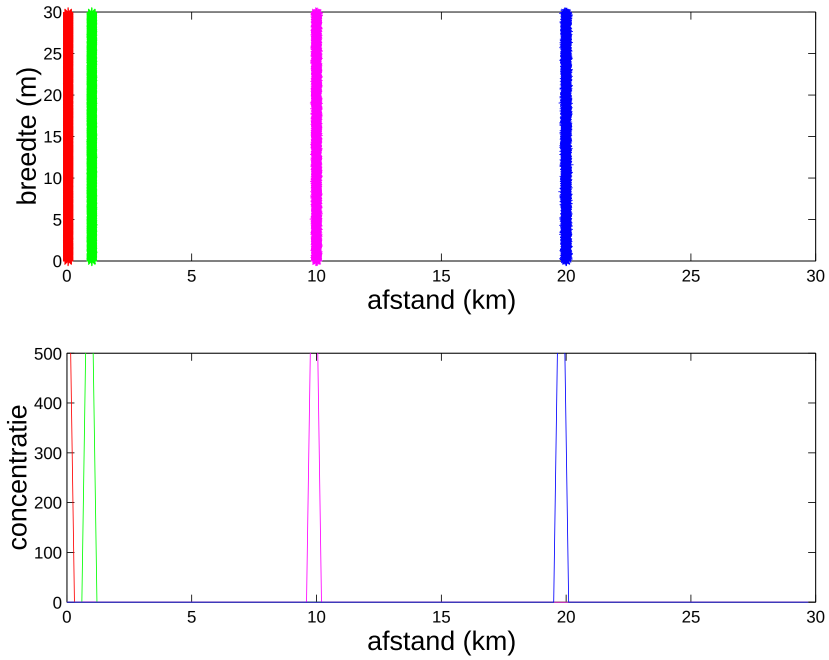
Wij zien geen reden om aan te nemen waarom dispersie in de ene richting groter zou zijn dan in de andere richting. Het resultaat wanneer we beide dispersiecoëfficienten op $10\frac{(cm)^2}{s}$ en $1\frac{(cm)^2}{s}$ zetten zijn gegeven in figuur 12 en figuur 13 respectievelijk. We zien dat figuur 13 sterk lijkt op ons eerdere figuur 5. Dit is ook weer te wijten aan de geringe invloed van D_x.

Wanneer we geen dispersie veronderstellen in de $x-$riching dan krijgen we het verloop zoals in figuur 14.

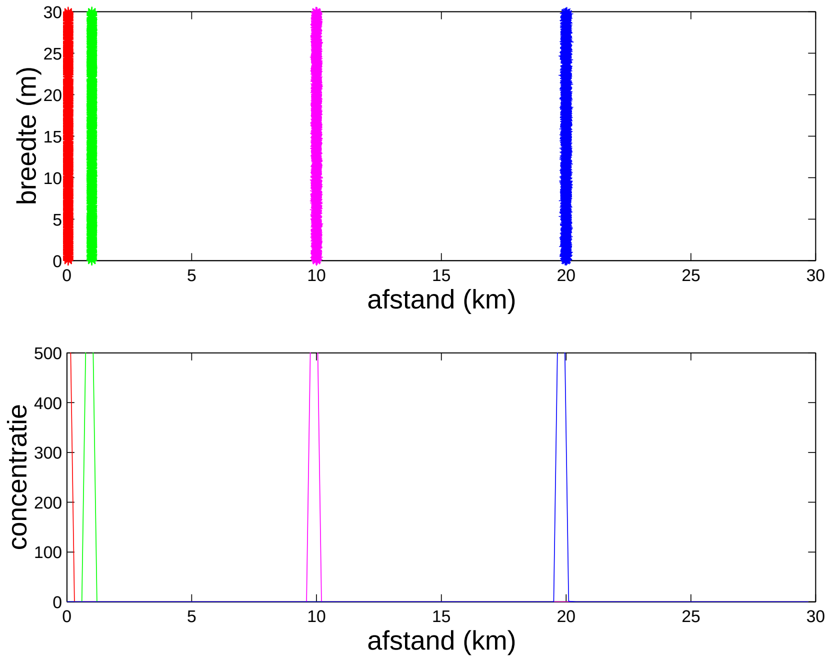
5 Conclusie

We hebben gezien dat de dispersie in de $x-$richting de breedte van de wolk beïnvloed maar niet tot grote verandering van de vorm zorgt. De dispersie in de $y-$richting geeft nog minder zichtbare verandering. Alle deeltjes blijven redelijk goed bij elkaar.

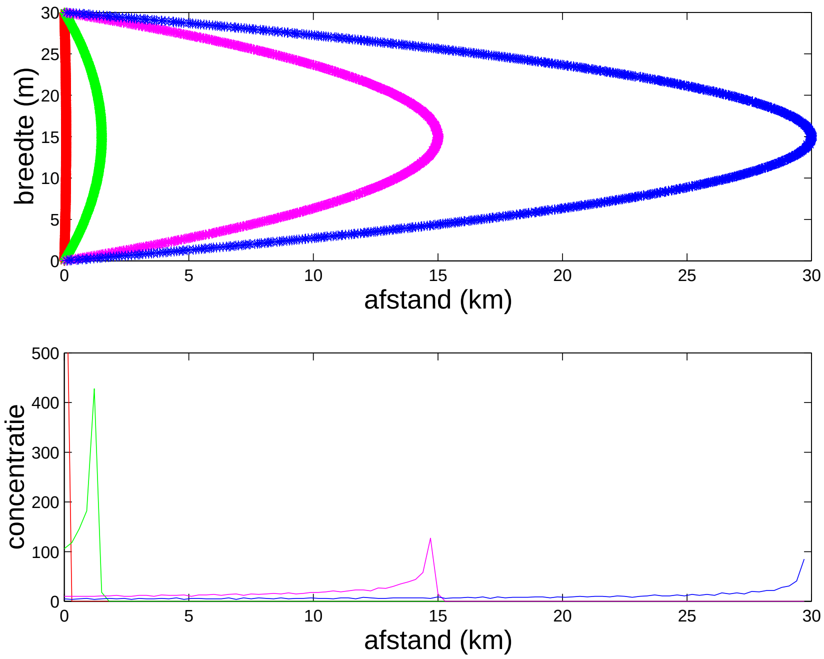
Een parabolisch stromingsprofiel geeft echter een heel ander beeld. Doordat de snelheid niet overal gelijk is bewegen de deeltjes niet in alle lagen van de stroming even snel. De dispersie in de $y-$richting zorgt nu dus voor snelheidsveranderingen van de deeltjes in de $x-$richting. Dit leidt tot een echte grote deeltjeswolk. De dispersie in de $x-$richting maakt de wolk wel iets breder, maar dit effect valt in het niet ten opzichte van het wisselvallige snelheidspatroon van de deeltjes.



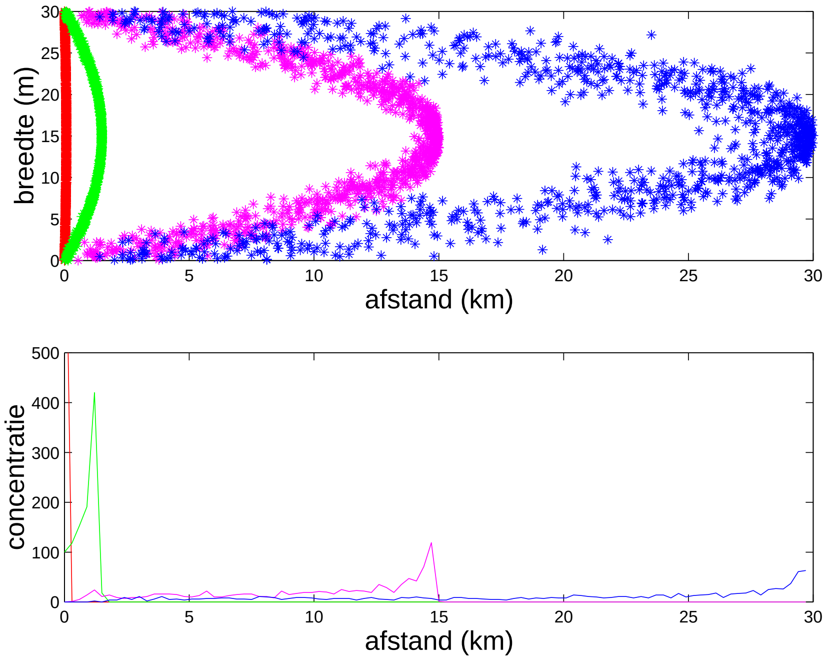
Figuur 2: Alleen dispersie in de $x-$richting, uniform stroomprofiel



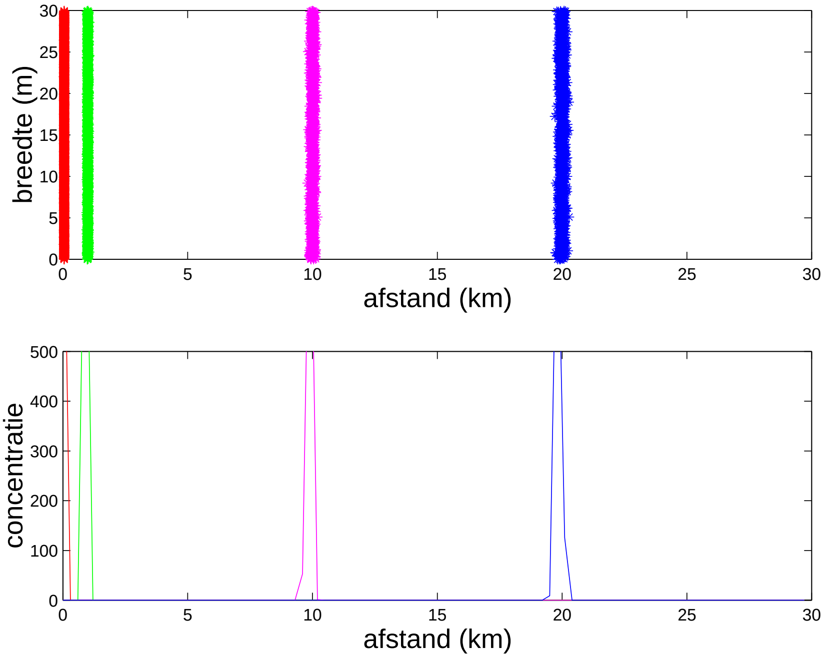
Figuur 3: Dispersie in beide richtingen, uniform stroomprofiel



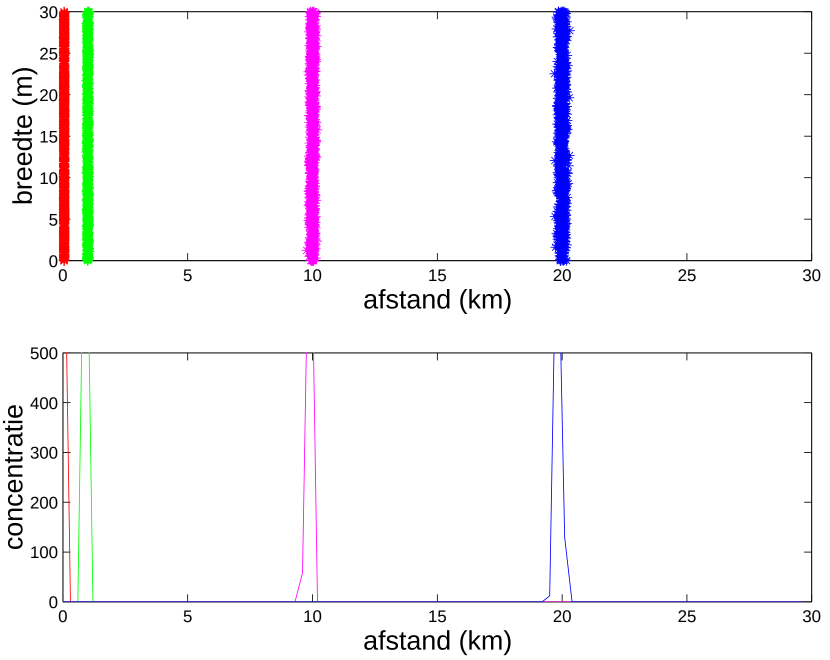
Figuur 4: Alleen dispersie in de $x-$richting



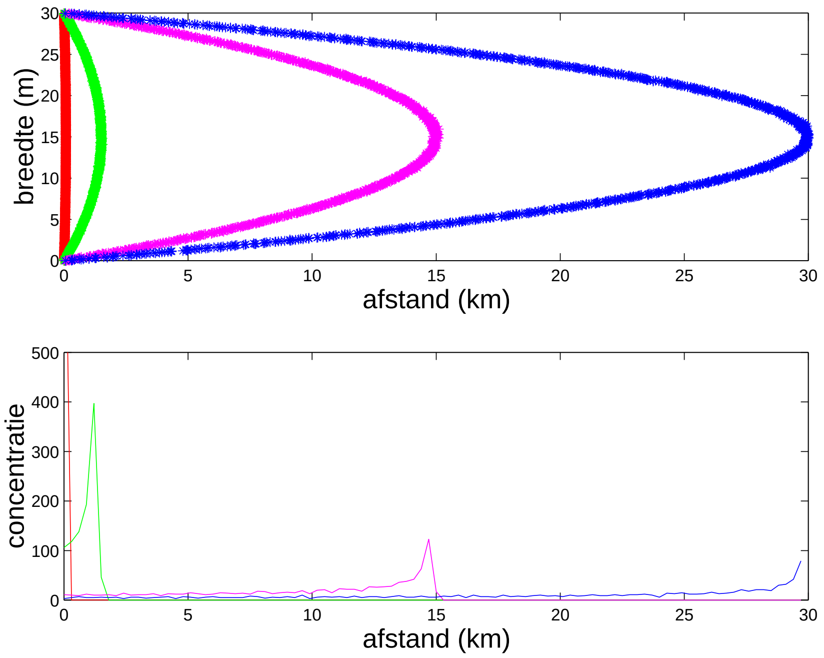
Figuur 5: Dispersie in beide richtingen



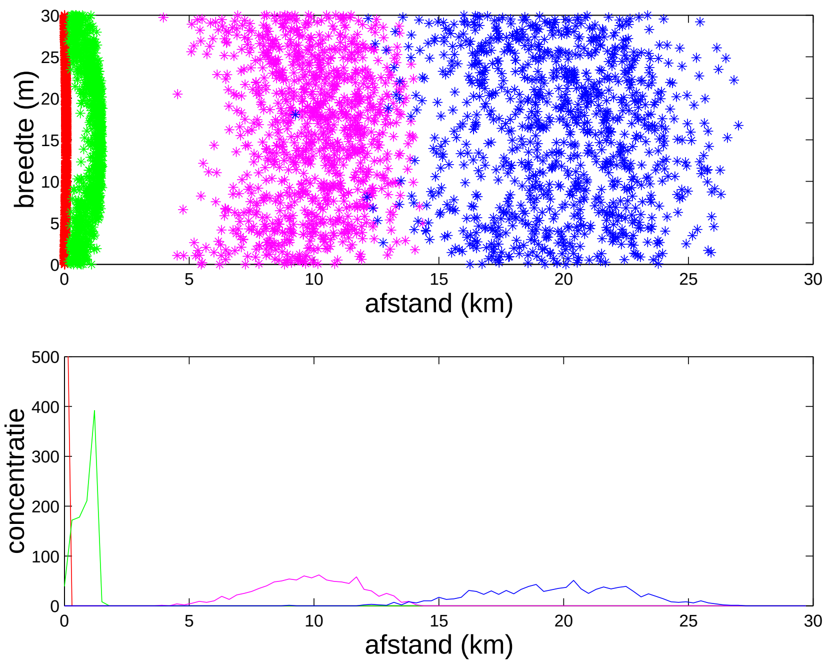
Figuur 6: Vergrote dispersie in $x-$richting



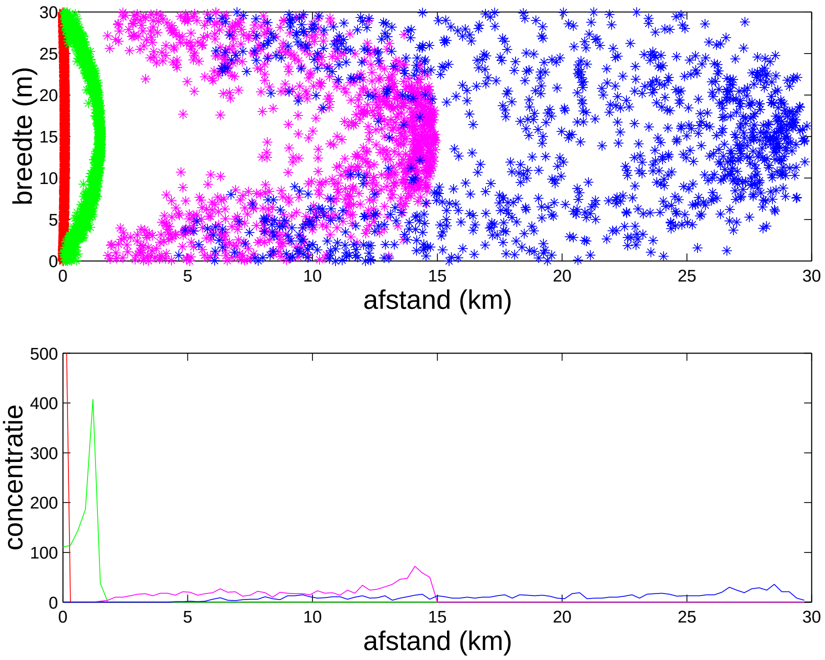
Figuur 7: Vergrote dispersie in $x-$ en $y-$richting



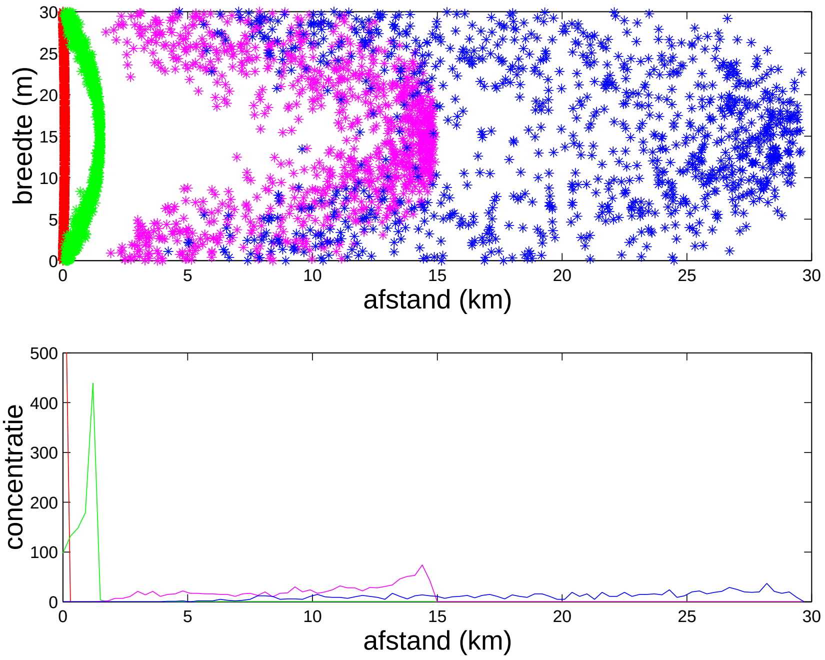
Figuur 8: Vergrote dispersie in $x-$richting



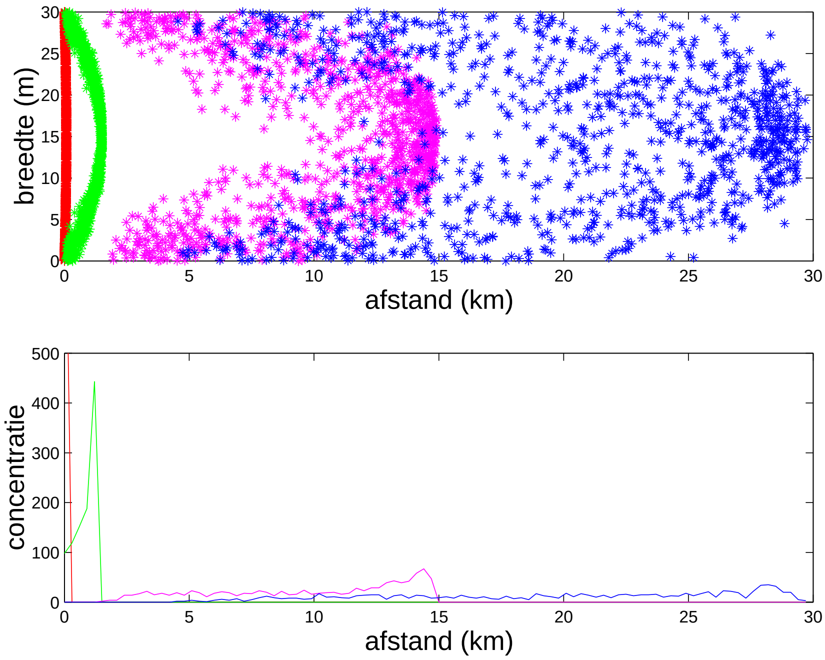
Figuur 9: Vergrote dispersie in $x-$ en $y-$richting



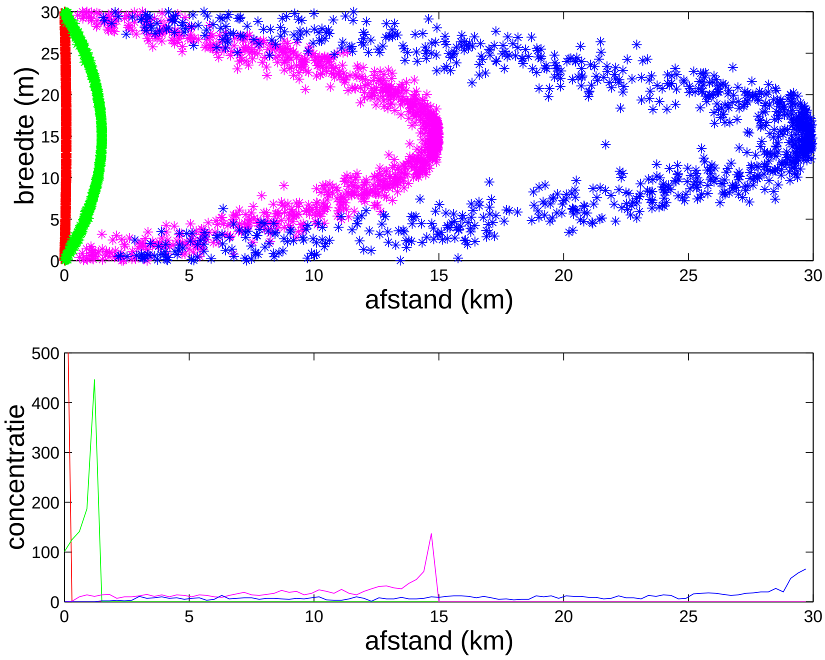
Figuur 10: Vergrote dispersie in $x-$richting en iets groter in y



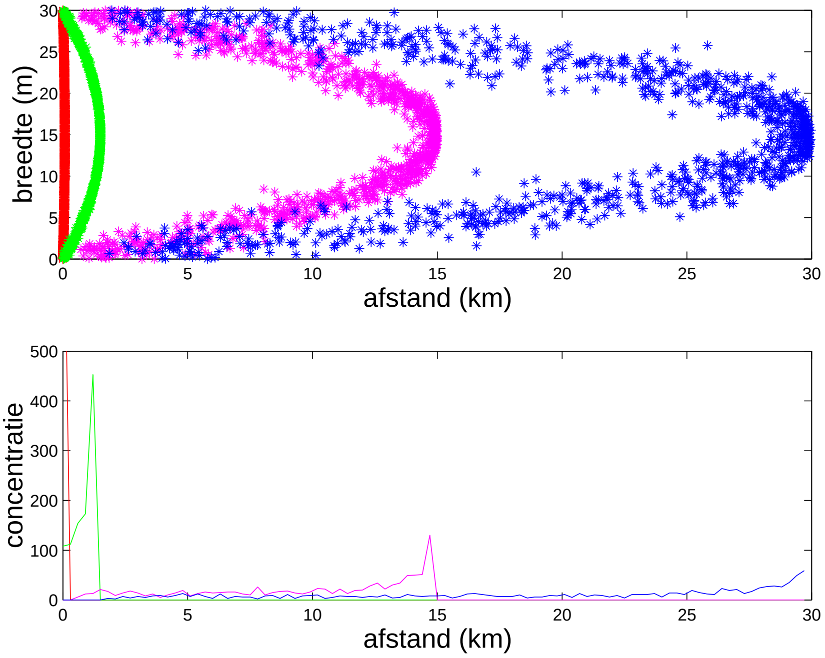
Figuur 11: Alleen grotere dispersie in $y-$richting



Figuur 12: Dispersie in $x-$ en $y-$richting gelijk



Figuur 13: Dispersie in $x-$ en $y-$richting gelijk, iets kleiner



Figuur 14: Geen dispersie in de $x-$richting

A Matlab sources

A.1 Aanroep

```
% vraag 1
advdisp (0.01,0,0)
print -depsc2 pic01.eps

advdisp (0.01,0.0001,0)
print -depsc2 pic02.eps

% vraag 2
advdisp (0.01,0,1)
print -depsc2 pic03.eps

advdisp (0.01,0.0001,1)
print -depsc2 pic04.eps

% eigen  probeersels
% Dx*10, Dy*100
advdisp (0.1,0,0)
print -depsc2 pic05.eps

advdisp (0.1,0.01,0)
print -depsc2 pic06.eps

advdisp (0.1,0,1)
print -depsc2 pic07.eps

advdisp (0.1,0.01,1)
print -depsc2 pic08.eps

% Dx=0.1, Dy=0.001
advdisp (0.1,0.001,1)
print -depsc2 pic09.eps

% Dx=0.01, Dy=0.001
advdisp (0.01,0.001,1)
print -depsc2 pic10.eps

% Dx=Dy=0.001
advdisp (0.001,0.001,1)
```

```
print -depsc2 pic11.eps

% Dx=Dy=0.0001
advdisp (0.0001,0.0001,1)
print -depsc2 pic12.eps

% Dx=0, Dy=0.0001
advdisp (0,0.0001,1)
print -depsc2 pic13.eps

% filmpje
advdisp (0.01,0.0001,1,2)
```

A.2 Advectie-Dispersie

```
% ADVDISP advectie dispersie algoritme
%
% advdisp(Dx,Dy) berekend de locaties en de concentraties van de
% partikels
%
% advdisp(Dx,Dy, snelheidsprofiel ) doet het voorgaande maar dan volgens
% het voorgeprogrammeerde snelheidsprofiel van de stroming
% ( snelheidsprofiel is een boolean)
%
% advdisp(Dx,Dy, snelheidsprofiel , film ) kan een filmpje maken
%      0 = uit
%      1 = movie en AVI maken (mits Matlab versie >= 6)
%      2 = serie plaatjes genereren
%      3 = niets bewaren alleen alles laten zien
%
% Standaard worden er 1000 partikels genomen over een breedte van 30 meter
% met een tijdsbestek van 40000 seconden.
%


function advdisp(Dx,Dy, snelheidsprofiel , film )
T=40000;
N=1000;
dt=1;
b=30;
dx=300;            % sample interval voor concentratie


```

```
if (nargin < 2)
        fprintf('Geef beide dispersiecoefficienten op!\n');
        return
end
if (nargin < 3)
        snelheidsprofiel=0;
end
if (nargin < 4)
        film=0;
end
if (snelheidsprofiel ~= 1)
        U=0.5;
        snelheidsprofiel=0;
end

w0=0;
w1(1)=w0;
w2(1)=w0;

w1=zeros(1,N);
w2=zeros(1,N);

x=zeros(1,N);
y=[0:b/(N-1):b];
frame=0;

clf
set(subplot(211),'XLim',[0 30000],'YLim',[0 b],'NextPlot','add','XTickLabel',[0:5:30],'Box','on');
set(xlabel('afstand (km)'),'FontSize',16);
set(ylabel('breedte (m)'),'FontSize',16);

set(subplot(212),'XLim',[0 30000],'YLim',[0 N/2],'NextPlot','add','XTickLabel',[0:5:30],'Box','on');
set(xlabel('afstand (km)'),'FontSize',16);
set(ylabel('concentratie'),'FontSize',16);

set(gcf,'Position',[8 64 640 480])


if (film==1)
        M=moviein(T/dt-1);
end
```

```
versie=[version];

% in de tijd
for i=2:T/dt
    fprintf('Seconde %.5d\n',i*dt)
    % alle partikels
    for k=1:N
        if snelheidsprofiel
            U=3*(y(k)*(b-y(k)))/(b^2);
        end

        x(k)=x(k)+U*dt+sqrt(2*Dx)*sqrt(dt)*randn;
        y(k)=y(k)+sqrt(2*Dy)*sqrt(dt)*randn;
    end

    % partikels mogen niet buiten de oevers komen, vector implementatie
    y=y.*(y>0);          % als <0 dan y=0
    y=y.*(y<b)+b*(y>b);      % als >b dan y=b


    if (film==0)
        % moet er iets geplot worden?
        if (i*dt==100)
            subplot(211), plot(x,y,'r*')
            c=zeros(30000/dx,1);
            t=1;
            for j=1:dx:30000
                c(t)=sum((x>=j)+(x<j+dx)==2);
                t=t+1;
            end
            subplot(212), plot([1:dx:30000],c,'r')
            shg
        elseif (i*dt==2000)
            subplot(211), plot(x,y,'g*')
            c=zeros(30000/dx,1);
            t=1;
            for j=1:dx:30000
                c(t)=sum((x>=j)+(x<j+dx)==2);
                t=t+1;
            end
            subplot(212), plot([1:dx:30000],c,'g')
            shg
        elseif (i*dt==20000)
```

```
                subplot(211), plot(x,y,'m*')
                c=zeros(30000/dx,1);
                t=1;
                for j=1:dx:30000
                    c(t)=sum((x>=j)+(x<j+dx)==2);
                    t=t+1;
                end
                subplot(212), plot([1:dx:30000],c,'m')
                shg
            elseif (i*dt==40000)
                subplot(211), plot(x,y,'b*')
                c=zeros(30000/dx,1);
                t=1;
                for j=1:dx:30000
                    c(t)=sum((x>=j)+(x<j+dx)==2);
                    t=t+1;
                end
                subplot(212), plot([1:dx:30000],c,'b')
                shg
            end

        elseif (mod(i*dt,25)==0)            % elke 25 stappen een plaatje
            frame=frame+1;

            subplot(211), cla, plot(x,y,'b*')

            c=zeros(30000/dx,1);
            t=1;
            for j=1:dx:30000
                c(t)=sum((x>=j)+(x<j+dx)==2);
                t=t+1;
            end
            subplot(212), cla, plot([1:dx:30000],c,'b')
            shg
            if (film == 1)
                if (versie(1)>=6)
                    M=getframe(gcf,[0, 0, 642, 480]);
                else
                    M(:,frame)=getframe;
                end
            elseif (film == 2)
                print(gcf,'-dbmp16m',sprintf('/mnt/storage/Dragonball/new/irc/filmpje/plaatje_%.5d',frame))
```

```
            end
        end
end

if (film == 1)
        % wegschrijven van filmpjes werkt alleen bij Matlab 6+
        if (str2num(versie(1)) >= 6)
            movie2avi(M,'advectie.avi')
        else
            movie(M)
        end
end
```