**Analysis of OsCaR measurements in Mokbaai tidal basin**

Loading datafiles

close all

clear variables

cd('Mokbaai');

umax = load('umax.txt'); umax = umax./100;

amax = load('amax.txt'); amax = amax./100;

load('grainsize.txt');

load('Benthos.txt');

Numbering sites

umax(:,4) = [1 1 1 1 1 2 2 2 2 2 3 3 3 3 3 4 4 4 4];

Obtaining some means and standard deviations for sites

for i = 1:4

ucrmean(i) = mean(umax(umax(:,1) == i,1));

ucrstd(i) = std(umax(umax(:,1) == i,1));

D50mean(i) = mean(grainsize(grainsize(:,1) == i, 2));

D50sandmean(i) = mean(grainsize(grainsize(:,1) == i, 3));

D90mean(i) = mean(grainsize(grainsize(:,1) == i , 6));

D50std(i) = std(grainsize(grainsize(:,1) == i, 2));

D50sandstd(i) = std(grainsize(grainsize(:,1) == i, 3));

D90std(i) = std(grainsize(grainsize(:,1) == i , 6));

Mudmean(i) = mean(sum(grainsize(grainsize(:,1) == i,4:5),2));

Mudstd(i) = std(sum(grainsize(grainsize(:,1) == i,4:5),2));

end

**Calculating the bottom shear stress from flume measurements**

a = 0.0076;

fw = a.\*amax(:,1).^-0.52;

BSS = 0.5.\*1027.\*fw.\*umax(:,1).^2;

**Calculating the bottom shear stress from Wu et al., 2018**

Inputs

g = 9.81; % gravity

v = 1e-6; % viscosity

for i = 1:length(umax)

D50(i,1) = D50mean(umax(i,4))./10^6; % median grain size (m)

D90(i,1) = D90mean(umax(i,4))./10^6; % 90th prctile grain size(m)

D50sand(i,1) = D50sandmean(umax(i,4))./10^6;

end

T = 5; %wave period (s)

R = 1.65; % suspended sediment density

grainsize(:,6) = grainsize(:,4) + grainsize(:,5);

taucr theory for benthos

for i = 1:length(umax)

Mudprc(i,1) = median(grainsize(grainsize(:,1) == umax(i,4),6));

end

p\_mud = Mudprc./100; % relative mud concentration

p\_sand = 1-p\_mud; % relative sand concentration

BSS\_mud = 3.5;

Dstarsand = D50sand.\*((g.\*R)./v^2).^(1/3); % nondimensional grain size

theta\_cr = (0.30./(1+1.2.\*Dstarsand))+0.055.\*(1-exp(-0.020.\*Dstarsand)); % shields number

BSS\_sand = theta\_cr.\*R.\*1027.\*g.\*D50sand; %critical shear stress

for i = 1:length(umax)

if Mudprc(i) > 5

alpha(i) = 0.42.\*exp(-3.38.\*D50sand(i).\*1000);

phase1(i) = BSS\_sand(i)+1.25.\*(BSS\_mud-BSS\_sand(i)).\*min(Mudprc(i)./100,0.05);

BSS\_theory(i) = phase1(i)+(BSS\_mud-phase1(i)).\*exp(-alpha(i).\*((1-(Mudprc(i)./100))./(Mudprc(i)./100)).^1.2);

elseif Mudprc(i) < 5

BSS\_theory(i) = BSS\_sand(i);

end

end

taucr theory for sediments

clear Dstarsand theta\_cr BSS\_sand alpha phase1 drydens\_mix

Mudprc = grainsize(:,6);

Dstarsand = (grainsize(:,3)./10^6).\*((g.\*R)./v^2).^(1/3);

theta\_cr = (0.30./(1+1.2.\*Dstarsand))+0.055.\*(1-exp(-0.020.\*Dstarsand)); % shields number

BSS\_sand = theta\_cr.\*R.\*1027.\*g.\*(grainsize(:,3)./10^6); %critical shear stress

alpha = 0.42.\*exp(-3.38.\*(grainsize(:,3)./10^6).\*1000);

phase1 = BSS\_sand+1.25.\*(BSS\_mud-BSS\_sand).\*min(grainsize(:,6)./100,0.05);

BSS\_theory\_sed = phase1+(BSS\_mud-phase1).\*exp(-alpha.\*((1-(grainsize(:,6)./100))./(grainsize(:,6)./100)).^1.2);

for i = 1:4

BSSmean(i) = mean(BSS(umax(:,4) == i));

BSSstd(i) = std(BSS(umax(:,4) == i));

BSStheorymean(i) = mean(BSS\_theory(umax(:,4) == i));

end

**Analysis of benthos data**

Calculating total metabolic rate folliwing Brey 2010, and splitting this based on functional groups following queiros et al., 2013.

for i = 1:4

for j = 1:4

Itot(i,j) = sum(Benthos(find(Benthos(:,1) == i & Benthos(:,2) == j),3)).\*(1/0.019); % total metabolic rate

Itotstab(i,j) = sum(Benthos(find(Benthos(:,1) == i & Benthos(:,2) == j & Benthos(:,5) == 0),3)).\*(1/0.019); % metabolic rate of stabilizers

ItotS(i,j) = sum(Benthos(find(Benthos(:,1) == i & Benthos(:,2) == j & Benthos(:,5) == 1),3)).\*(1/0.019); % metabolic rate of surface modifiers

ItotUCDC(i,j) = sum(Benthos(find(Benthos(:,1) == i & Benthos(:,2) == j & Benthos(:,5) == 2),3)).\*(1/0.019); % metabolic rate of up / downward converyers

ItotB(i,j) = sum(Benthos(find(Benthos(:,1) == i & Benthos(:,2) == j & Benthos(:,5) == 3),3)).\*(1/0.019); % metabolic rate of burrowers

ItotR(i,j) = sum(Benthos(find(Benthos(:,1) == i & Benthos(:,2) == j & Benthos(:,5) == 4),3)).\*(1/0.019); % metabolic rate of reworkers

end

means and standard deviations

Itotmean(i) = mean(Itot(i,:));

Itotstd(i) = std(Itot(i,:));

Itotstabmean(i) = mean(Itotstab(i,:));

ItotSmean(i) = mean(ItotS(i,:));

ItotUCDCmean(i) = mean(ItotUCDC(i,:));

ItotBmean(i) = mean(ItotB(i,:));

ItotRmean(i) = mean(ItotR(i,:));

Itotstabstd(i) = std(Itotstab(i,:));

ItotSstd(i) = std(ItotS(i,:));

ItotUCDCstd(i) = std(ItotUCDC(i,:));

ItotBstd(i) = std(ItotB(i,:));

ItotRstd(i) = std(ItotR(i,:));

end

Itotmeans = [ItotRmean; ItotSmean; ItotUCDCmean; ItotBmean; Itotstabmean]';

Itotstds(:,:) = [ItotRstd; ItotSstd; ItotUCDCstd; ItotBstd; Itotstabstd]';

%change zeros to low value for log scale

ItotR(ItotR == 0) = 1e-5;

ItotS(ItotS == 0) = 1e-5;

ItotUCDC(ItotUCDC == 0) = 1e-5;

ItotB(ItotB == 0) = 1e-5;

Itotstab(Itotstab == 0) = 1e-5;

**Assessing errors and statistical analysis**

BSSreduction = 1./(BSS./BSS\_theory');

for i = 1:4

meanBSSerror(i)= mean(abs(BSS(umax(:,4) == i)-BSStheorymean(i)));

meanBSSreduction(i) = 1./mean((BSS(umax(:,4) == i)./BSStheorymean(i)));

end

[~,Site1v3BSSreduction] = ttest2(BSSreduction(umax(:,4) == 1),BSSreduction(umax(:,4) == 3));

[~,Site1v3Itot] = ttest2(Itot(1,:),Itot(3,:));

[~,tbl,stats] = anova1(BSSreduction([1:5,11:end],1),umax([1:5,11:end],4));

Tukey\_BSSreduction = multcompare(stats,'CType','tukey-kramer');

[~,Ttest\_Itot] = ttest2(Itot(1,:),Itot(3,:));

**Figures**

figure;

subplot(2,2,[3 4])

hold on

boxplot(ItotR','positions',[0.7:1:3.7],'Colors',[0 0 0],'Widths',0.15);

boxplot(ItotS','positions',[0.85:1:3.85],'Colors',[0 0 0],'Widths',0.15);

boxplot(ItotUCDC','positions',[1:1:4],'Colors',[0 0 0],'Widths',0.15);

boxplot(ItotB','positions',[1.15:1:4.15],'Colors',[0 0 0],'Widths',0.15);

boxplot(Itotstab','positions',[1.3:1:4.3],'Colors',[0 0 0],'Widths',0.15);

colors = [0.3 0.8 0.3; 0.3 0.8 0.3; 0.3 0.8 0.3; 0.3 0.8 0.3; ...

1 0.8 0; 1 0.8 0; 1 0.8 0; 1 0.8 0; ...

0.2 0.5 1; 0.2 0.5 1; 0.2 0.5 1; 0.2 0.5 1; ...

1 0 0.2; 1 0 0.2; 1 0 0.2; 1 0 0.2; ...

0.6 0.2 0.6; 0.6 0.2 0.6; 0.6 0.2 0.6; 0.6 0.2 0.6];

boxes = findobj(gca,'Tag','Box');

for i = 1:length(boxes)

patch(get(boxes(i),'XData'),get(boxes(i),'YData'),colors(i,:),'FaceAlpha',0.5,'EdgeColor','none');

p(i) = patch(NaN,NaN,colors(i,:),'FaceAlpha',0.5,'EdgeColor','none');

end

set(gca,'Xtick',[1:1:4])

set(gca,'XtickLabel',{'Site 1','Site 2','Site 3','Site 4'})

axis([0 5 1e-6 200])

set(gca,'Box','off')

ylabel('I\_{tot} (mW m^{-2})')

legend([p(17) p(13) p(9) p(5) p(1)],'Reworker','Surface modifier','Up/downward conveyer','Burrower','Stabilizer')

legend('BOXOFF')

title('C')

subplot(2,2,1)

boxplot(BSS,BSS\_theory,'Colors',[0 0 0],'Positions',BSS\_theory,'Widths',0.2);

boxes = findobj(gca,'Tag','Box');

colors = [0.1 0.1 0.1;

0.7 0.7 0.2;

0.4 0.4 0.1;

1 1 0];

for i = 1:4

patch(get(boxes(i),'XData'),get(boxes(i),'YData'),colors(i,:),'FaceAlpha',0.5,'EdgeColor','none');

p(i) = patch(NaN,NaN,colors(i,:),'FaceAlpha',0.5,'EdgeColor','none');

end

hold on

plot(BSS\_theory,BSS,'xk')

plot([0 4],[0 4],'-k')

axis([0 3.1 0 3.1])

xticks([0 0.5 1 1.5 2 2.5 3])

xticklabels({'0','0.5','1','1.5','2','2.5','3'})

xlabel('\tau\_{cr,Wu} (N m^{-2})')

ylabel('\tau\_{cr,measured} (N m^{-2})')

text(0.1,0.5,'site 4')

text(2,2.5,'site 3')

text(3,2,'site 1')

text(3,3,'site 2')

set(gca,'Box','off')

title('A')

text(0.25,2,'1:1')

subplot(2,2,2)

hold on

errorbar(Itotmean,BSSmean./BSStheorymean,-BSSstd./BSStheorymean,BSSstd./BSStheorymean,-Itotstd,Itotstd,'.k')

axis([-10 250 0 1.2])

set(gca,'box','off')

xlabel('I\_{tot} (mW m^{-2})')

ylabel('\tau\_{cr,measured} / \tau\_{cr,Wu}')

title('B')

Fitting a curve for subplot B

modelfun = @(b,x)(1-b(1).\*(x.^b(2)));

beta0 = [0 0];

[beta2,Reg,J,CovB,MSE] = nlinfit(Itotmean,BSSmean./BSStheorymean,modelfun,beta0);

[ypred,delta] = nlpredci(modelfun,Itotmean,beta2,Reg,'Covar',CovB,'MSE',MSE,'SimOpt','on');

r = corrcoef(ypred,BSSmean./BSStheorymean);

rsq = r(2)^2;

plot([0:1:250],1-beta2(1).\*([0:1:250].^beta2(2)),'--k')

text(50,0.9,['y = 1 - ',num2str(beta2(1),2),'x^{',num2str(beta2(2),2),'}\newlineR^2 = ',num2str(rsq,2)])