function H = RS(sequence,isplot)

%

% 'RS' estimate the hurst parameter of a given sequence with R/S method.

%

% Inputs:

% sequence: the input sequence for estimate

% isplot: whether display the plot. without a plot if isplot equal to 0

% Outputs:

% H: the estimated hurst coeffeient of the input sequence

% Author: Chu Chen

% Version 1.0, 03/10/2008

% chen-chu@163.com

%

if nargin == 1

isplot = 0;

end

N = length(sequence);

dlarge = floor(N/5);

dsmall = max(10,log10(N)^2);

D = floor(logspace(log10(dsmall),log10(dlarge),50));

D = unique(D);

n = length(D);

x = zeros(1,n);

y = zeros(1,n);

R = cell(1,n);

S = cell(1,n);

for i = 1:n

d = D(i);

m = floor(N/d);

R{i} = zeros(1,m);

S{i} = zeros(1,m);

matrix\_sequence = reshape(sequence(1:d\*m),d,m);

Z1 = cumsum(matrix\_sequence);

Z2 = cumsum(repmat(mean(matrix\_sequence),d,1));

R{i} = (max(Z1-Z2)-min(Z1-Z2));

S{i} = std(matrix\_sequence);

if min(R{i})==0 || min(S{i}) ==0

continue;

end

x(i) = log10(d);

y(i) = mean(log10(R{i}./S{i}));

end

% fit a line with middle part of sequence

index = x~=0;

x = x(index);

y = y(index);

n2 = length(x);

cut\_min = ceil(3\*n2/10);

cut\_max = floor(9\*n2/10);

X = x(cut\_min:cut\_max);

Y = y(cut\_min:cut\_max);

p1 = polyfit(X,Y,1);

Yfit = polyval(p1,X);

H = (Yfit(end)-Yfit(1))/(X(end)-X(1));

if isplot ~= 0

figure,hold on;

bound = ceil(log10(N));

axis([0 bound 0 0.75\*bound]);

temp = (1:n).\*index;

index = temp(index);

for i = 1:n2

plot(x(i),log10(R{index(i)}./S{index(i)}),'b.');

end

x = linspace(0,bound,10);

y1 = 0.5\*x;

y2 = x;

h1 = plot(x,y1,'b--','LineWidth',2);

h2 = plot(x,y2,'b-.','LineWidth',2);

plot(X,Yfit,'r-','LineWidth',3);

legend([h1,h2],'slope 1/2','slope')

xlabel('log10(m)'),ylabel('log10(R/S)'),title('R/S Method');

end