Data Analysis and Modelling
Exam

Evidence for Lunar Cycles in Ocean Patterns

Joseph Pritchard - Adelaide University

June 5, 2020

Introduction

Evidence for cyclic behavior in ocean state on the scale of the siderial (27.3 days) and synodic (29.5 days) lunar orbital periods is searched for in sea level, water temperature, air temperature, barometric pressure, wind direction, wind gust and wind speed. Data is sourced from annual ocean measurements provided by the Australian Bureau of Meteorology (BOM) for the years spanning (2000-2002). A Wavelet Transform (WT) is used for analysis (Gaussian, Mexixan Hat, Morlet) and a random white noise filter applied as a test for statistical significance.

Wavelet Analysis Method

A function (Wavelet) with time and frequency localised features is selected and the overlap computed at every time point of our data. Mathematically this is given by

$$\psi(t) = \psi\left(\frac{t - t_0}{s}\right)$$

where ψ is our chosen Wavelet, t the current comparison time, t_0 the time localisation of our Wavelet and s the scale factor. s is proportional to frequency and may be converted using the appropriate formula for the chosen Wavelet [2].

By this approach we are thus able to distinguish the periodicities present in our data and correspondingly the time at which they occur. Time localisation provides an advantage compared with traditional Fourier processes [1]. Time vs frequency clarity is a tradeoff attributable to the chosen wavelet - oscillatory wavelets (Morlet) provide greater frequency resolution at the expense of time clarity when compared to time localised Wavelets (Gaussian) [2].

In search for oscillatory behaviour on the scale of 27.3 days we choose a Morlet Wavelet for analysis to provide strong frequency resolution. All analysis is carried out with use of MATLAB software, the script for which may be viewed in the appendix.

Statistical Significance

A random noise time signal is generated by allocating a random number selected from a uniform distribution to every time point. The signal is then scaled by the standard deviation of our data up to a sigma level of our choosing. Noise is Wavelet Transformed and all WT ocean data below the corresponding noise level is set to zero, leaving only data above the specified significance level. Through virtue of the uniform distribution sampled, the noise is white in nature.

In testing it is found a sigma level of 2 standard deviations provides rejection strong enough to obscure background features whilst preserving the clarity of features of interest.

Data Drop-outs

Throughout the supplied data there exist periods of erroneous measurement where values of -9999.0 are recorded for several variables. To overcome this clear error, these values are set to zero and wavelet analysis conducted as normal. It should be noted this data fault provides limitations on the clarity of the resulting transformation plots, especially in the case of air temperature where from late 2000 through late 2002 this feature is present. To limit these negative effects, WT is performed for a time duration of 100 days beggining at the start of 2000, to limit the inclusion of such points.

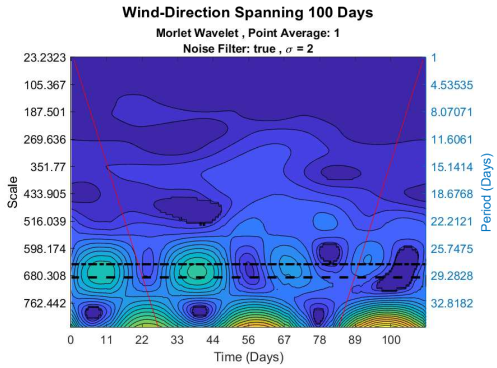
It is suspected this error in measurement is attributable to measurement device failure and maintenance periods.

Results

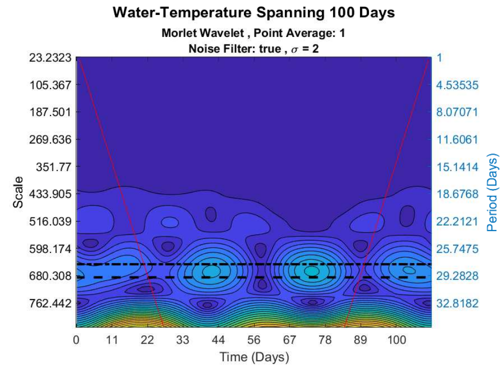
Mixed Lunar Cycle Behaviour

Periodicity at roughly 28 days is observed in WTs of wind direction, water temperature, air temperature, pressure and sea level. This value lies halfway between the siderial (27.3) and synodic (29.5) lunar periods. This behavior sits above the 2σ level of random noise giving us confidence in its presence (Figure 1 (a)-(e)).

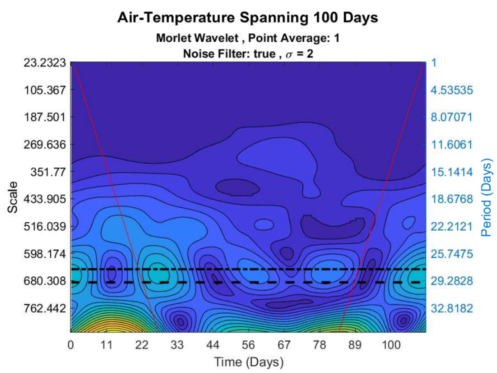
Mixed Lunar Cycle Behavior



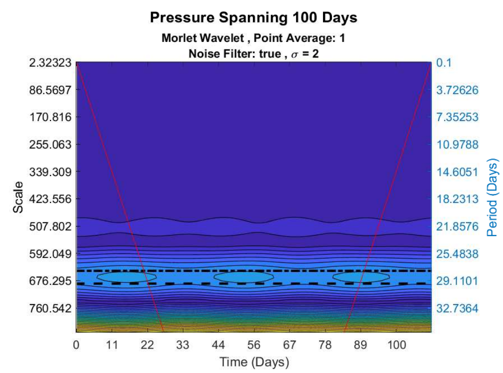
(a) Wind Direction



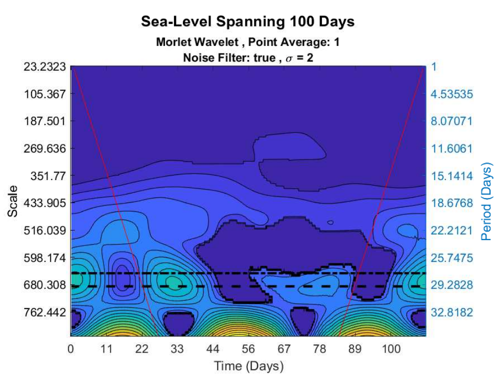
(b) Water Temperature



(c) Air Temperature



(d) Pressure



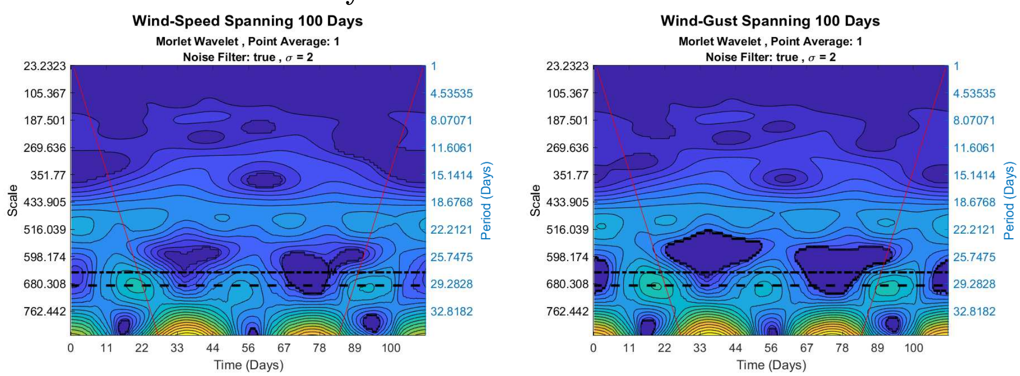
(c) Sea Level

Figure 2
Upper and lower dashed lines mark siderial and synodic periodicities respectively. All data sits above the 2σ level. Contours are linear in magnitude.

Synodic Behaviour

Observe in the displayed results (Figure 2 (a)-(b)) the intersection of the lower dashed line (marking synodic period) with the central point of the contour peaks. This is consistent across the displayed peaks, all of which sit above the 2σ level giving us confidence in the validity of the link to the synodic cycle.

Synodic Behavior



(a) Wind Speed

(b) Wind Gust

Figure 2
Upper and lower dashed lines mark siderial and synodic periodicities respectively. All data sits above the 2σ level. Contours are linear in magnitude.

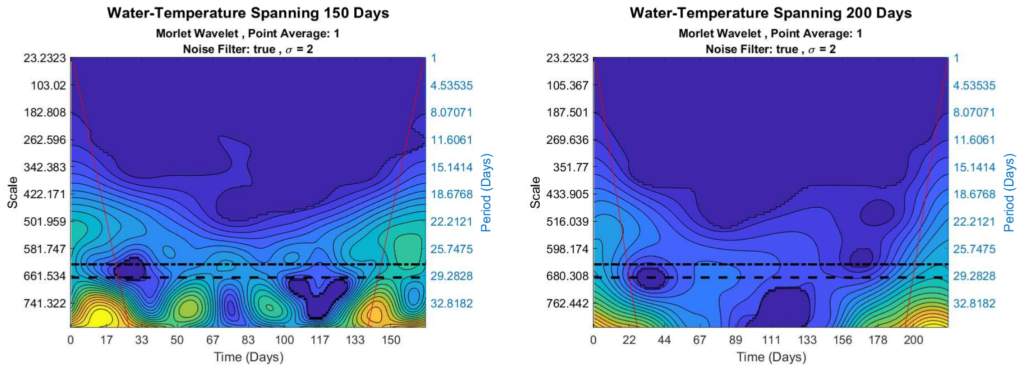
Discussion

The effectiveness of the wavelet technique is highly dependent on the size of data sampled. Sampling for large periods of time - specifically greater than 100 days - causes amplification in the detection of high period peaks, to the extent that lunar orbital periods become obscured. This is showcased by sampling for periods of 150 and 200 days (Figure 3 (a)-(b)).

This restriction in maximum sample duration provides limitations in determining the temporal duration of period signals present within the data. This drawback is remedied by performing WT on subsequent year data (2001, 2002), which show the same periodicities present with over 2σ accuracy (Appendix Item 1)).

Choice of period sampling range (set by scale factor s) provide the possibility of drowning the desired observations, again as higher period behaviours cause the relative size of lower period peaks to vanish (Figure 4 (a)-(b)).

Time Limiting Effects

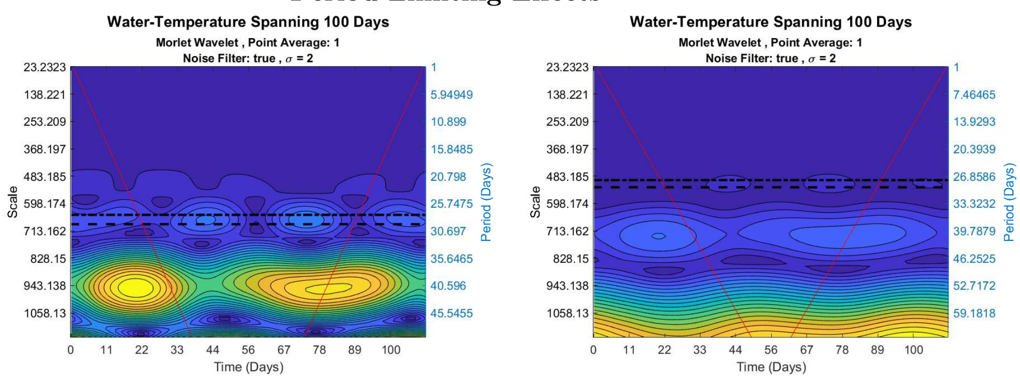


(a) 150 Day Sample

(b) 200 Day Sample

Figure 3
Showcase of the drowning effects of extended sampling periods.

Period Limiting Effects



(a) 50 Day Maximum Period

(b) 200 Day Maximum Period

Figure 4
Showcase of the drowning effects of extended period (scale) axis.

Alternate Wavelets

Alternative Wavelets are trialed but yield non-definitive results (Figure 5). Given the sensitivity of sampling duration and period axis settings in the clarity of Morlet transform results, it is believed this is the likely cause for lack of clarity in the alternate WT plots. Of the three wavelets trialed, Morlet provides the greatest frequency localisation and least time localisation. In switching to a wavelet with weaker frequency clarity, it is suspected the prominent higher period peaks (Figure 4) delocalise in frequency and obscure the lunar peaks.

This belief is supported by noting the oscillatory Mexican Hat wavelet provides clearer low period peaks than the time localised Gaussian Wavelet (Figure 5). For succinctness water temperature plots are provided only but it should be noted the behaviour of interest is ubiquotess across measurement categories.

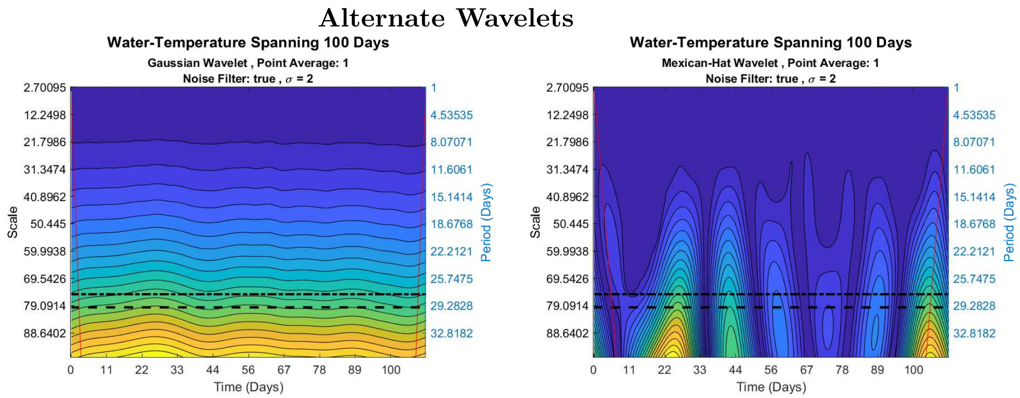


Figure 5
Results for Gaussian and Mexican Hat wavelet analysis. Observe lack of frequency space clarity attributable to the time localised nature of the wavelets.

All plots feature cone of influence (CoI) markings (red diagonal lines). These mark regions beyond which data may be contaminated by edge effects inherent in the WT process. This limitation consistently provides no issue as statistically significant results lie within these boarders for all attributes investigated. The extent of CoI presence is dependent on the Wavelet chosen [2].

Conclusion

Statistically significant results for the presence of synodic (29.5 day) and mixed (28 day) lunar cycles within sea level, water temperature, air temperature, barometric pressure, wind direction, wind gust and wind speed are located within the first 100 days of the year 2000. These results are confirmed to the same significance levels in subsequent years 2001 and 2002 indicating a consistent link between lunar period and the analysed ocean features.

References

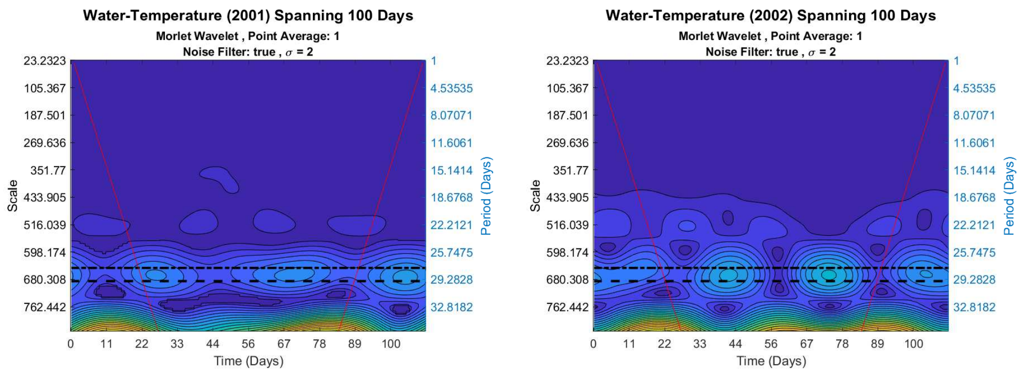
[1] Rowell, G 2020, *Data Analysis and Modelling Course Notes*, Data Analysis and Modelling, University of Adelaide

[2] C. Torrence, G. Compo, 1998, *A Practical Guide to Wavelet Analysis*, Program in Atmospheric and Oceanic Sciences, University of Colorado, Boulder, Colorado

Appendix

Following are additional illustrative plots (Item 1) and the MATLAB software used in the generation of all plots included in this report.

Alternate Sample Year Example



Item 1
Water Temperature is provided here as an illustrative example. Consistent year-to-year features arise for all remaining features and corresponding WT plots are available upon request.

Table of Contents

 1
Wavelet Settings 2
Display Settings 2
Plotting 2


```
%close all
clear all

TEST = importdata('2019_1.txt');
TEST = TEST.data(:,12);
Testf = 24*60;

Data0 = importdata('2002.txt');
data0 = Data0.data(:,:);

Data1 = importdata('2001.txt');
data1 = Data1.data(:,:);

Data2= importdata('2002.txt');
data2 = Data2.data(:,:);

%Connect years
data = cat(1,data0,data1,data2);

%Extract Data
SL = data(:,1);
SL = (SL > -9000) .* SL;

WTemp = data(:,2);
WTemp = (WTemp > -9000) .* WTemp;

ATemp = data(:,3);
ATemp = (ATemp > -9000) .* ATemp;

P = data(:,4);
P = (P > -9000) .* P;

Res = data(:,5);
Res = (Res > -9000) .* Res;

ARes = data(:,6);
ARes = (ARes > -9000) .* ARes;

WDir = data(:,7);
WDir = (WDir > -9000) .* WDir;

WGus = data(:,8);
WGus = (WGus > -9000) .* WGus;
```

```
WS = data(:,9);
WS = (WS > -9000) .* WS;
```

Wavelet Settings

```
%Data to Plot
%SL, WTemp, ATemp, P, Res, ARes, WDir, WGus, WS
Choice = WTemp;

%Wavelet Choice
WN = "MH";

%Set sample frequency points per day
fsamp = 24;

%Set period search range
Tmin = 1;
Tmax = 36;

%Set number of points to sample
nmax = 2^14;

Days = 100;
nmax = Days * fsamp;

%Set noise filter and sigma level
filter = true;
sigma = 2;

%Set Decimation Factor
d = 1;
```

Display Settings

```
%Number of disections for x axis and labels
xdis = 10;

%Specify axis labels
ticks = linspace(1,nmax,xdis);
xlabels = round(ticks/fsamp,0);

%Set title and file name
FILENAME = "Pressure";
```

Plotting

```
%Set frequency range to scan
fmin = 1/(Tmax);
fmax = 1/(Tmin);

%figure
%plot(TEST)
```

```
%WavT(Choice, WN, fsamp, fmin, fmax, nmax, d, filter, sigma, xdis,
 xlabels, FILENAME)
```

Published with MATLAB® R2018b

```
% y == data
% WN == Wavelet name, EG "MH", "GW", "MW"
% fsamp == Sampling freq of data
% fmin == minimum frequency to plot
% fmax == maximum frequency to plot
% nmax == maximum number of time data points (Make power of 2 for
 improved
% FT performance
% d == Only use every d'th time data point
% filter == apply a noise filter (boolean)
% xdis == number of disections for xaxis ticks
% xlabels == array of (xdis) timestamps
%            Note: xlabels mark the time at the start of the (xdis)'th
%            degment. EG: if nmax == 10, tmax = 10sec, xdis = 2, then
%            xlabels is a 2-element array: [0,5]
function WavT(y, WN, fsamp, fmin, fmax, nmax, d, filter, sig, xdis,
 xlabels, FILENAME)

    %Define Wavelets
    %Mexican Hat
    if(WN == "MH")
        MH.Func = @(t, t0, s) (1 - ((t-t0)./s).^2).*exp(-0.5*((t-t0)./
s).^2);
        MH.Norm = @(s, Y) (2*pi*s)*Y;
        MH.S_to_f = @(fs, s) (fs) .* ( ( 2*pi / sqrt(2.5) ) .*
 s ).^(-1);
        MH.name = "Mexican-Hat";
        MH.coi = @(scale) sqrt(2)*scale;
        WF = MH;
    end

    %Gaussin Wavelet
    if(WN == "GW")
        GW.Func = @(t, t0, s) pi^(-0.5) * exp(-0.5*((t-t0)./s).^2);
        GW.Norm = @(s, Y) (2*pi*s)*Y;
        GW.S_to_f = @(fs, s) (fs) .* ( ( 2*pi / sqrt(0.5) ) .*
 s ).^(-1);
        GW.name = "Gaussian";
        GW.coi = @(scale) sqrt(2)*scale;
        WF = GW;
    end

    %Morlet Wavelet - w0 must satisfy admissability critereon
    if(WN == "MW")
        w0 = 6;
        C = (1 + exp(-w0^2) - 2*exp((-3/4)*w0^2))^(-1/2);
        MW.Func = @(t, t0, s) C *(1/(pi^0.25)) .* exp(1i*w0.* ((t-
t0)./s) ) .* exp(- ((t-t0)./s).^2 / 2);
        MW.Norm = @(s, Y) (2*pi*s)*Y;
        MW.S_to_f = @(fs, s) (fs) .* ( (4*pi / (w0 + sqrt(2 +
 w0^2))) .* s ).^(-1);
        MW.name = "Morlet";
```

```
        MW.coi = @(scale) sqrt(2)*scale;
        WF = MW;
    end

    %Paul Wavelet
    PW.Func = @(t, t0, s) ((-8)/(sqrt(pi*24))) .* (1 - 1i* ((t-t0)./
s) ).^3;
    PW.Norm = @(s, Y) (2*pi*s)*Y;
    PW.S_to_f = @(fs, s) (fs) .* ( ( 4*pi/5 ) .* s ).^(-1);
    PW.name = "Paul";
    PW.coi = @(scale) scale*sqrt(2);

    %Decimate and truncate data
    %Improves quality of WT
    %WT is best for short time data sets
    y = y(1:d:nmax);
    fSamp = fsamp/d;

    %Convert to power spectrum
    Pyy = (abs(fft(y))).^2;
    Y = fft(y);
    stop = length(Pyy);

    %Set zero point for imaginary FT component
    t = 1:stop;
    t0 = stop/2;

    %Set scale array using fmin and fmax
    smax = WF.S_to_f(fSamp,fmin);
    smin = WF.S_to_f(fSamp,fmax);
    scale = linspace(smin,smax, 100);

    %Perform Transform
    WT = W_T(scale, t, t0, Y, stop, WF);

    %Apply a Noise Filter if filter==true
    if(filter)
        r = sig*std(y)*randn(1,length(y));
        R = fft(r);
        WTN = W_T(scale, t, t0, R', stop, WF);
        %Set signal beow noise to be zero
        WT = Filter(WT, WTN, scale, stop);
    end

    %Plot
    Plot_WT(t, scale, WT, fSamp, WF, filter, xdis, xlabels, FILENAME);

Not enough input arguments.

Error in WavT (line 18)
    if(WN == "MH")
```

DEPENDENCIES

```
    function Plot_WT(t, scale, WTd, fSamp, WF, filter, xdis, xlabels,
FILENAME)
    %Plot a Wavelet data set WT

        %Final integer specifies the number of contours to render
        %figure
        contourf(t, scale, abs(WTd),20);
        axis ij;
        hold on;

        %Mark Lunar Siderial Period
        Moon = 0*t + WF.S_to_f(fSamp,1/(27.8));
        plot(Moon,'k-.','LineWidth',2)

        %Mark Lunar Synodic Period
        Moon = 0*t + WF.S_to_f(fSamp,1/(29.5));
        plot(Moon,'k--','LineWidth',2)

        %Create scale axis
        %Set s to be multiple of 10 in length
        slen = floor((length(scale)/10)) * 10;

        s = scale(1:slen/10:slen);
        yticks(s);
        yticklabels(s);
        ylabel('Scale');

        %Create frequency axis
        yyaxis right
        f = WF.S_to_f(fSamp,s);
        yticks(s);
        yticklabels(1./f);
        ylabel('Period (Days)');

        %Create time axis
        time = t(1:length(t)/xdis:length(t));
        xticks(time);
        xticklabels(xlabels);
        xlabel('Time (Days)');

        %Define and plot Cone of Influence
        coi = WF.coi(scale);
        t0n = t(1:length(coi));
        plot(scale,coi,'r-');
        plot(max(t) - scale, coi, 'r-');
        axis ij

        %Set labels and axis format
        axis([0 length(t) scale(1) scale(length(scale))])

        if(filter == false)
```

```
            title({'\fontsize{13}' + FILENAME + " Spanning " +
xlabels(end) + " Days",'\fontsize{10}' + WF.name + ' Wavelet , Point
Average: ' + d, "Noise Filter: " + filter});
        else
            title({'\fontsize{13}' + FILENAME + " Spanning " +
xlabels(end) + " Days",'\fontsize{10}' + WF.name + ' Wavelet , Point
Average: ' + d, "Noise Filter: " + filter + " , \sigma = " + sig});
        end

        %Save result
        saveas(gcf, FILENAME, 'epsc');
        saveas(gcf, FILENAME, 'jpeg');
        hold off
    end

    function result = W_T(scale, t, t00, Y, stop, WF)
    %Produce the WT of Y using Wavelet Function WF
        WTr = zeros(length(scale),stop);

        for ii = 1:length(scale)
            %Pick out scale scalar value
            s = scale(ii);
            %Set wavelet function
            w = WF.Func(t, t00, s);
            %Swap 1st and 2nd half of wavelet function in prep. for
ifft
            %Only select 1:stop # of points
            w = [w(stop/2:stop) w(1:(stop/2-1))];
            %Normalize wavelet function wrt s
            W = WF.Norm(s, fft(w));
            %Calculate time domain for current scale value
            WTr(ii,:) = ifft(conj(W').*Y);
        end
        result = WTr;
    end

    function result = Filter(WTtf, WTN, scale, stop)
    %Set all WT value below WTN to ~zero
        for ii = 1:length(scale)
            for jj = 1:stop
                if abs(WTtf(ii,jj)) < abs(WTN(ii,jj))
                    WTtf(ii,jj) = eps;
                end
            end
        end
        result = WTtf;
    end

end
```

Published with MATLAB® R2018b