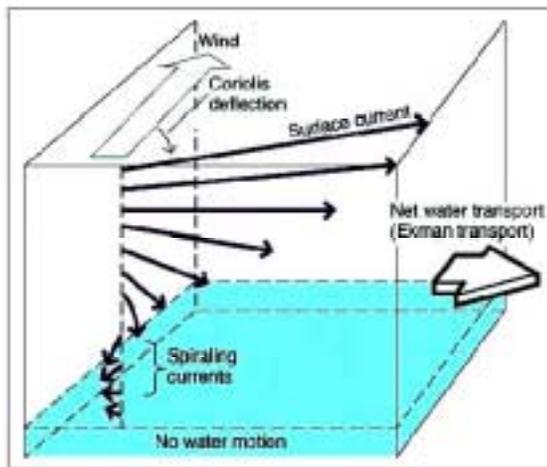


**Wind Stress and Ekman Mass Transports
along CalCOFI Lines: 67,70 and 77**



**OC 3570
Lora Egley
16SEP01**

Wind Stress and Ekman Mass Transports along CalCOFI Lines: 67,70 and 77

Introduction

Wind driven currents from Ekman's hypothesis can be explained as the frictional stress that balances Coriolis force with a resultant mass transport at right angles to the wind direction (1905). The Ekman theory has been studied and tested directly and indirectly since 1905. The objective of this paper is to analyze and evaluate shipboard wind data and calculate the Ekman mass transport using indirect methods.

Prior Studies

An appropriate review of the literature was pertinent for this study to aid into understanding the complexity of the topic in terms of spatial and temporal variability. Furthermore, the research methods were examined. Just to point out a few results, include a paper by Price et. al.,(1987) who calculated transport from the mean current which resulted in "...an estimate of wind driven transport consistent with the transport estimated from in situ current measurements... (p. 1538). Additionally, Chereskin(1995) provided direct confirmation for an Ekman balance in California Current using moored acoustic Doppler

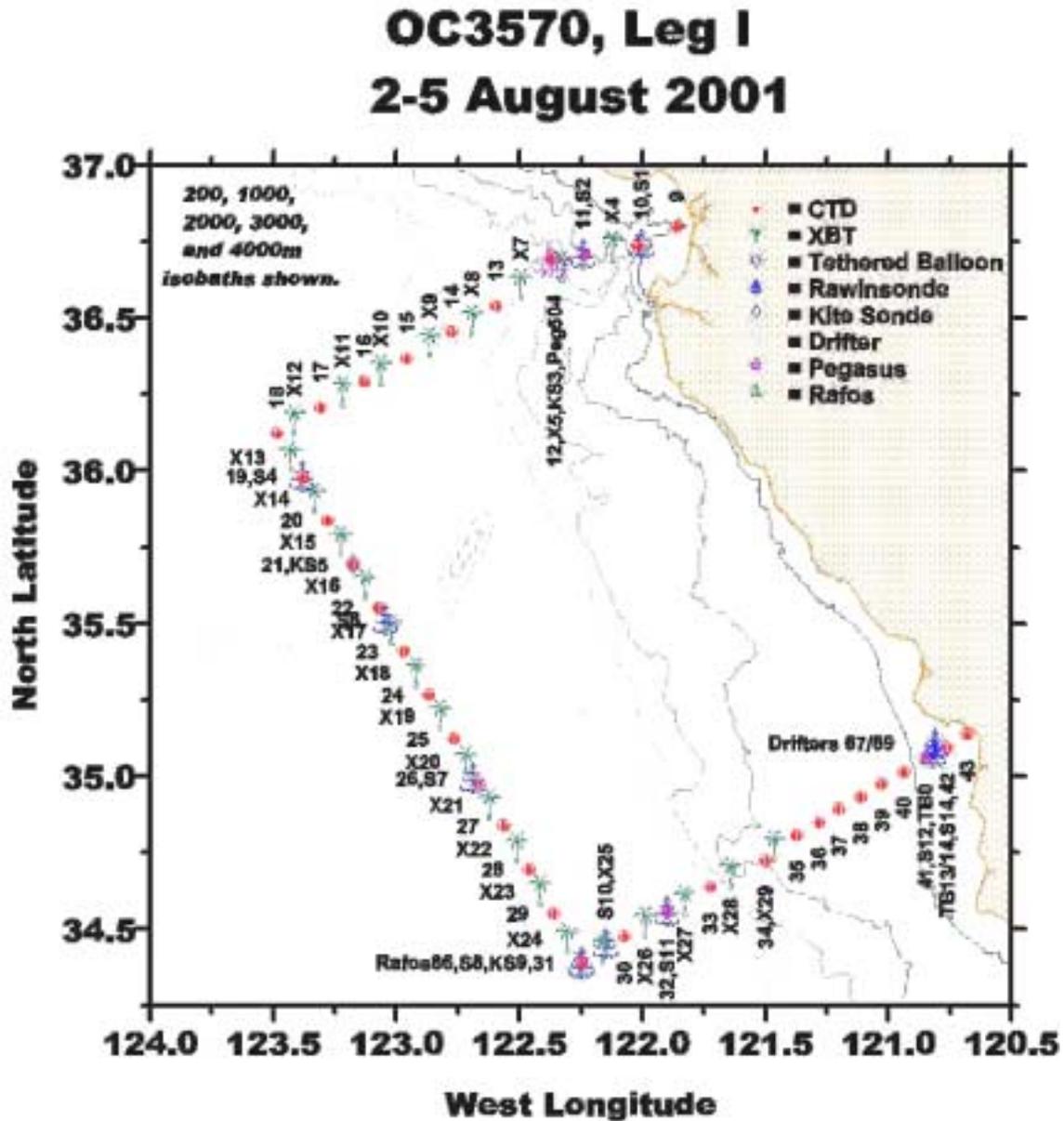
current profiler velocity estimates and buoy wind observations. Cooperative winds provided Chereskin's (1995) study "a 4 month period of unidirectional winds (southward) off Northern California..." which led to the finding of a net westward transport in the surface layer (p. 18,261). Most recently, Bograd et. al. (2001) showed inter-annual and seasonal variability in transports linked to low frequency changes in the wind patterns.

Data and Methods

Underway meteorological and surface oceanographic observations were obtained from the oceanographic cruise on the Research Vessel Point Sur. The data was collected from a 4-day period on the following dates: 2 - 5 August, 2001. Data utilized for this study consisted of wind direction, wind speed, position, time, air temperature, relative humidity, and sea surface temperature. These parameters were obtained from the Point Sur's Science Data Acquisition System referred to as the SAIL data. The raw data sample rate was approximately every 54 seconds. The location of the study was off the central coast of California within the California current. A coastal box was developed which encompassed California Cooperative Oceanic Fisheries Investigations (CalCOFI) line 67 hydrography to station 67-70, alongshore to station 77-70, and then inshore

along CalCOFI line 77 hydrography to Port San Luis. See figure 1 for cruise track and stations.

Figure 1:



A raw data analysis was completed. Parameters were plotted versus time to assess the data in terms of obvious errors. No

observable errors were noted. This analysis also provided a comparison between the port and starboard wind anemometers. There were a few outliers but nothing appreciable was noticed. The starboard anemometer was chosen for the mass transport calculations. Refer to raw data plots with respect to wind direction, wind speed, air temperature, pressure, humidity, and sea surface plots versus time given figures 1 through 13 given in appendix A of this report. The wind data illustrates a steady wind towards the southeast for the first 3 days.

The momentum balance for Geostrophic and Ekman flow develops into the following equations.

$$\text{Total Current : } u = u_E + u_G \quad (1)$$

$$\text{Geostrophic flow : } 2\Omega \times u_G = -\frac{1}{\rho} \nabla p \quad (2)$$

$$\text{Frictional Wind Driven (Ekman Flow) : } 2\Omega \times u_E = -\frac{1}{\rho} \frac{\partial \tau}{\partial z} \quad (3)$$

$$\text{Boundary Conditions : } \tau(z) = \tau_0 \text{ @ } z = 0 \quad (4)$$

$$\tau(z) = 0 \text{ @ } z \leq -H$$

The raw SAIL data was divided into hourly segments. The bulk formulas from Smith's (1988) paper "Coefficients of sea surface wind stress, heat flux, and wind profiles as a function of wind speed and temperature" were used to calculate the momentum and heat fluxes. A program written by Guest (1997) carried out these surface flux calculations. The program involved the following input of wind speed, air temperature, sea surface temperature, relative humidity, pressure, and height.

The essential outputs were surface friction (u^*) and wind stress (τ). The total stress was calculated for every data point. Note matlab programs are given in appendix C.

Equations 5a and 5b cannot be directly calculated because it would be very difficult to precisely measure the perturbed motion (u' and v'). Therefore, the indirect method is used by first solving for the surface friction velocity (u_*^2) based on the drag coefficient (cd) and the mean velocity ($\overline{u_{14}}$). It is important to note the reference height of the anemometer was 14 meters. Furthermore, the coriolis parameter set to a constant value of .0001/s and density (ρ) was set to a constant value of $1024 \frac{kg}{m^3}$. Refer to equations 5 through 8.

$$\text{Vertical Momentum Flux : } \tau = -\overline{\rho u' w'} \quad (5)$$

$$\tau = -\overline{\rho v' w'}$$

$$\text{Stress(Momentum Flux): } \tau = \rho c d u_*^2 \quad (6)$$

$$\text{Ekman Mass Transport : } M_x = \int_{-H}^0 u E \partial z = \frac{\tau_0^y}{\rho g} \quad (7a)$$

$$M_y = \int_{-H}^0 v E \partial z = -\frac{\tau_0^x}{\rho g} \quad (7b)$$

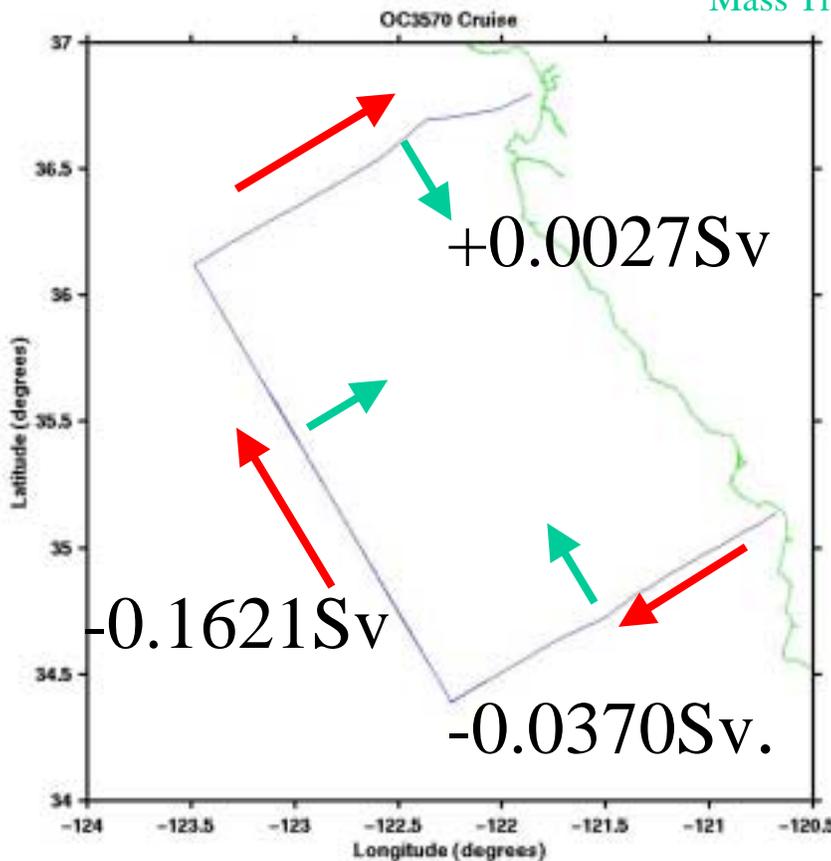
$$\text{Surface Friction Velocity : } u_*^2 = c d \overline{u_{14}} \quad (8)$$

The vector wind stress was then converted into its x (taux) and y (tauy) components. An hourly average was calculated for taux and each CalCOFI line was oriented so that the taux was computed along coastal box. Also to note a smaller average was done around the corners (approximately a 30 minute average). Then the each respective Ekman transport component was calculated specifically into or out of the box. Refer to figure 2 for spatial orientation. Additionally, tables 1 through 3 are provided for the hourly-calculated taux and mass transport values.

Figure 2:

Central Coast of California: Northwest Pacific

Taux along Calcofi line 67, 70, 77
Mass Transport y into the box



CalCOFI Line 67 (course 240)

Table 1 Total distance traveled: 175.4907km

Date	Time (GMT)	taux (kg/ms ²)	Mass transport (m ² /s)
			positive into the box
			negative out of box
2-Aug-01	1545-1645	-0.000421348	-0.004114731
	1645-1745	0.009808415	0.0957853
	1745-1845	0.019540112	0.19082141
	1845-1945	0.045809104	0.44735453
	1945-2045	0.026765457	0.26138141
	2045-2145	0.03454597	0.33736299
	2145-2245	0.060113384	0.58704477
	2245-2345	0.056671739	0.55343495
3-Aug-01	2345-0045	0.041320006	0.40351569
	0045-0145	0.033538254	0.32752201
	0145-0245	0.017145405	0.1674356
	0245-0345	0.01460597	0.14263643
	0345-0445	-0.003052682	-0.029811344
	0445-0545	-0.001891748	-0.018474098
	0545-0645	-0.002706774	-0.026433342
	0645-0745	-0.004395159	-0.042921473
	0745-0845	7.36656E-05	0.00071939
	0845-0945	0.011279892	0.11015519
	0945-1045	0.011508875	0.11239135
	1045-1145	0.015913583	0.15540608
	1145-1245	0.007933146	0.077472133
	1245-1319	0.003421414	0.033412249

CalCOFI Line 70 (course 150)

Table 2

Total distance traveled: 230.1515km

Date	Time (GMT)	taux (kg/ms ²)	Mass transport (m ² /s)
			positive into the box
			negative out of box
3-Sep-01	1545-1645	-0.057813977	-0.56458962
	1645-1745	-0.049824167	-0.48656413
	1745-1845	-0.055986148	-0.54673973
	1845-1945	-0.033853541	-0.33060099
	1945-2045	-0.02582014	-0.2521498
	2045-2145	-0.029436837	-0.28746911
	2145-2245	-0.048084241	-0.46957266
	2245-2345	-0.05396344	-0.52698672
4-Sep-01	2345-0045	-0.053814994	-0.52553705
	0045-0145	-0.073485771	-0.71763448
	0145-0245	-0.088272768	-0.86203875
	0245-0345	-0.082924682	-0.80981135
	0345-0445	-0.096809022	-0.94540061
	0445-0545	-0.090334828	-0.88217606
	0545-0645	-0.075028845	-0.73270356
	0645-0745	-0.067771813	-0.66183411
	0745-0845	-0.067809456	-0.66220172
	0845-0945	-0.064216215	-0.62711147
	0945-1045	-0.1155945	-1.1288526
	1045-1145	-0.11277117	-1.101281
	1145-1245	-0.1234209	-1.2052822
	1245-1345	-0.091856073	-0.89703197
	1345-1445	-0.10855121	-1.0600704
	1445-1513	-0.063088509	-0.61609872

CalCOFI Line 77 (course 060)

Table 3

Total distance traveled: 213.6358km

Date	Time (GMT)	taux (kg/ms ²)	Mass transport (m ² /s)
			positive into the box
			negative out of box
4-Aug-01	1745-1845	-0.006984435	-0.068207371
	1845-1945	-0.005646643	-0.055142998
	1945-2045	-0.010314589	-0.10072841
	2045-2145	-0.013594226	-0.13275612
	2145-2245	-0.016181432	-0.15802179
	2245-2345	-0.023246144	-0.22701313
5-Aug-01	2345-0045	-0.045039542	-0.43983927
	0045-0145	-0.044399682	-0.43359064
	0145-0245	-0.055743099	-0.5443662
	0245-0345	-0.051783433	-0.50569758
	0345-0445	-0.041248559	-0.40281796
	0445-0545	-0.02168842	-0.21180098
	0545-0645	-0.016720044	-0.16328168
	0645-0745	-0.016575388	-0.16186902
	0745-0845	-0.02481698	-0.24235332
	0845-0945	-0.040277841	-0.3933383
	0945-1045	-0.003542014	-0.034589979
	1045-1145	0.010559223	0.10311741
	1145-1245	0.001677763	0.016384405
	1245-1249	-0.002564989	-0.025048719

The distance traveled each hour for each stress component was deduced, so that it could be applied to the mass transport. This final calculation gives the volume transport. Refer to volume transport table 4. Note negative values correspond transports going out of the box and positive values correspond to values going into the box.

Table 4

CalCOFI	Net volume into	Net volume out of	Net total
Line	the box	the box	
67	0.0027381Sv.		
70		-0.16214Sv.	
77		-0.03698Sv.	
Total	0.0027381Sv.	-0.19912Sv.	-0.19638Sv.

Noise/Errors

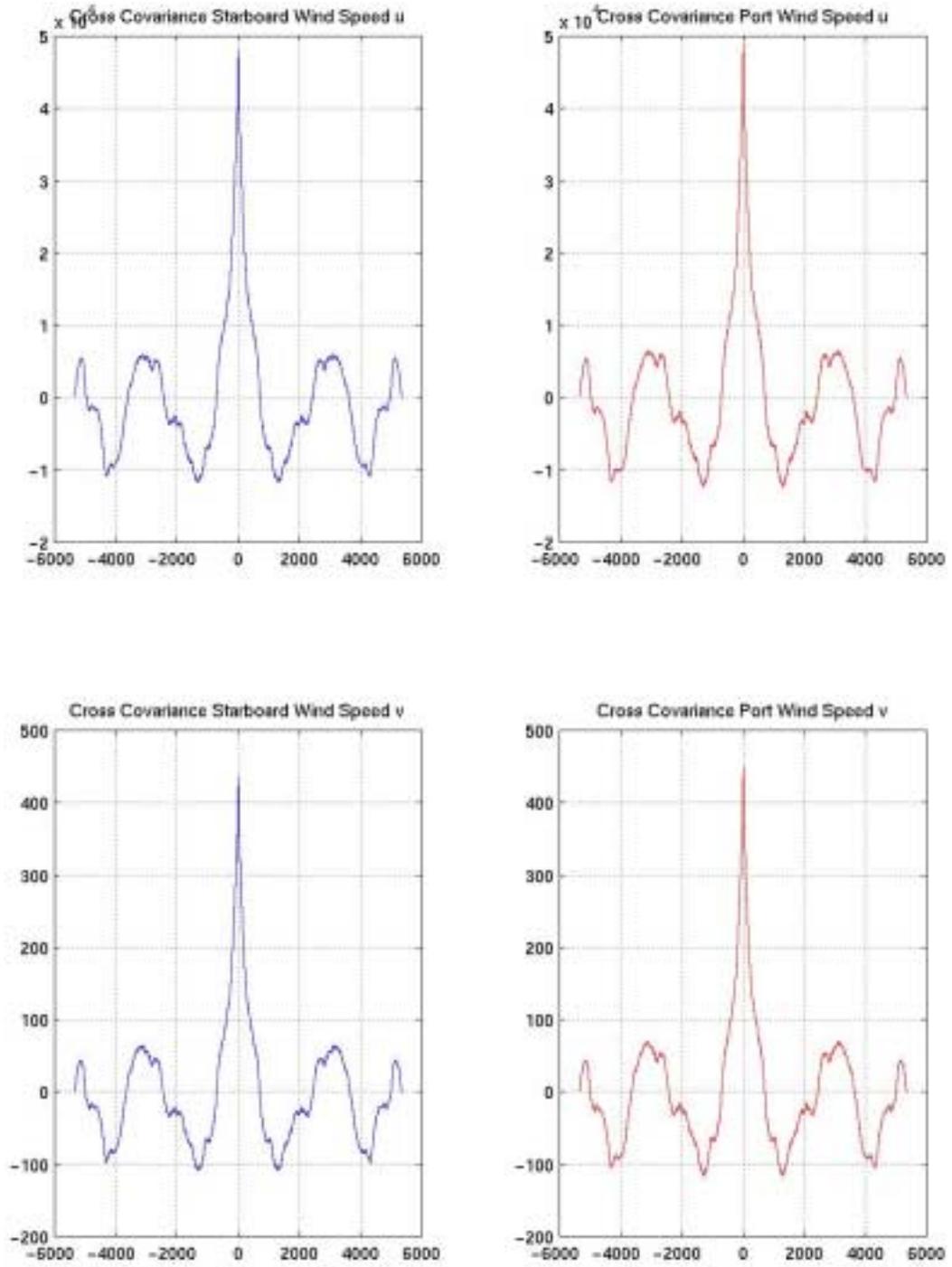
Averaging can reduce a relatively large random error. As averaging time record becomes large, the bias becomes large. There were approximately 67 samples in a one-hour of data. Computing hourly averages removed some of the higher frequency variability. Chereskin (1995) paper also addresses this problem by stating wind driven signal being masked by pressure driven current noise, and notes that statistics can provide some confidence however wind direction is highly variable. This

paper had the fortune of persistent winds to the southeast for the majority of the 4 days.

Time Scales

A hanning filter was used on the wind data to remove discontinuities at the start and end of records (Bendat and Piersol 2000). The matlab function `filtfilt` was utilized to provide a zero phase forward and reverse digital filtering. These filters were used to determine the sample rate. A plot of wind speed utilizing the cross covariance function indicates correlation up to 84 hours. See figure 3. Also, an integral time scale of 20 hours was calculated using a autocorrelation function to depict the random process's scale characteristics. For matlab code see appendix B14. Based on the above information, it was determined hourly sampling was sufficient to represent the Ekman transport. Also to note, from Chereskin's(1995) research, the wind driven flow is stated to be in Ekman balance on daily time scales.

Figure 3:

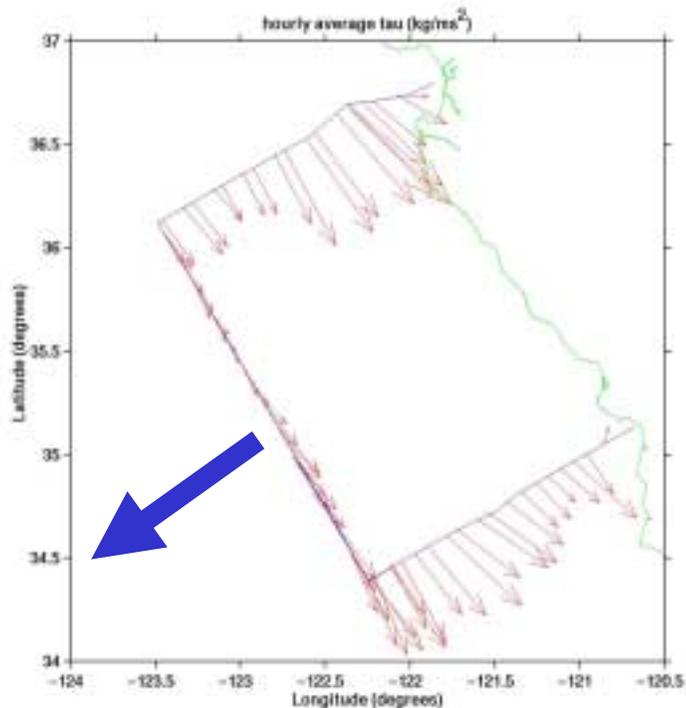


Results and Discussion

Refer to figure 4 and appendix C for hourly wind stress diagrams (τ) depicted along each respective CalCOFI line. Figures illustrate all three CalCOFI lines with the wind stress towards the southeast. Overall, CalCOFI line 67, the mass transport calculations give a net transport into the box and CalCOFI lines 70 and 77 show a net transport out of the box. Again the positive values correlate to a transport to in to the box, vice versa as indicated in tables 1 through 4. The overall net volume transport is approximately -0.2Sv to the southwest.

Figure 4:

Southwest
Transport



Concerns

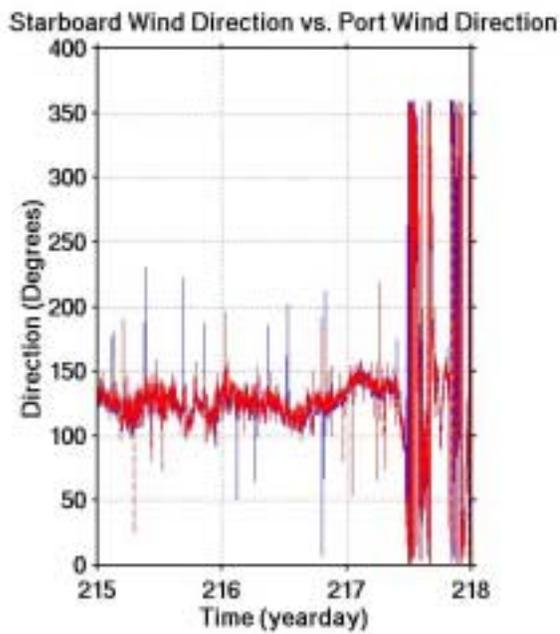
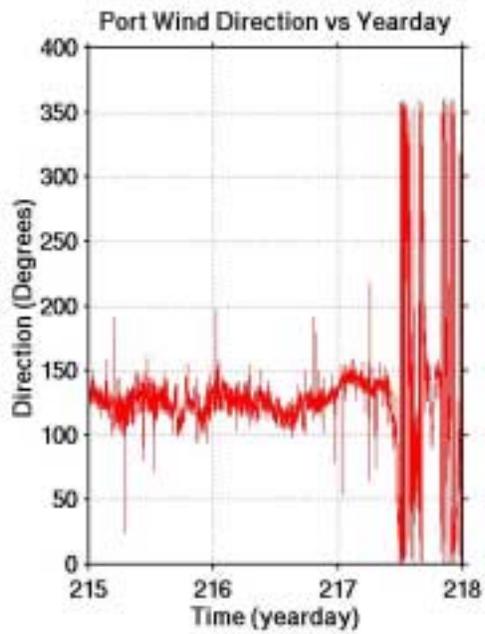
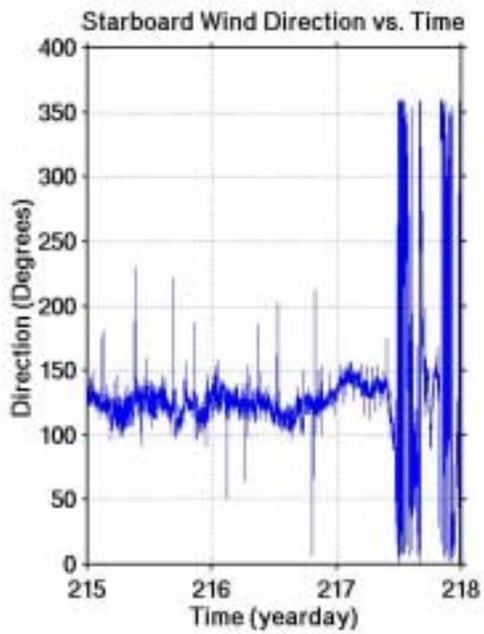
First, it is difficult to separate Ekman and Geostrophic components. Therefore, an indirect method must be used, which leads to errors based on assumptions. Secondly, it is difficult to obtain the perfect environment to meet all of the Ekman assumptions, which are steady state, closure, vertical homogenous ocean, and away from horizontal boundaries. Wind direction is usually a highly variable parameter. Thirdly, the space and time scales are very complex.

Summary and Conclusions

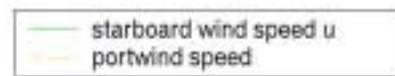
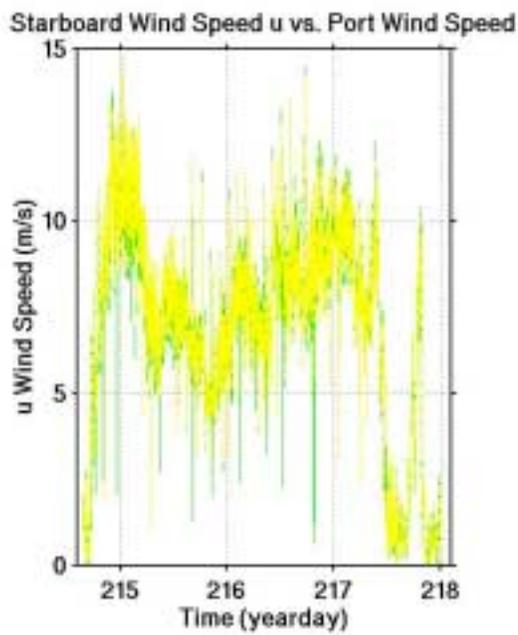
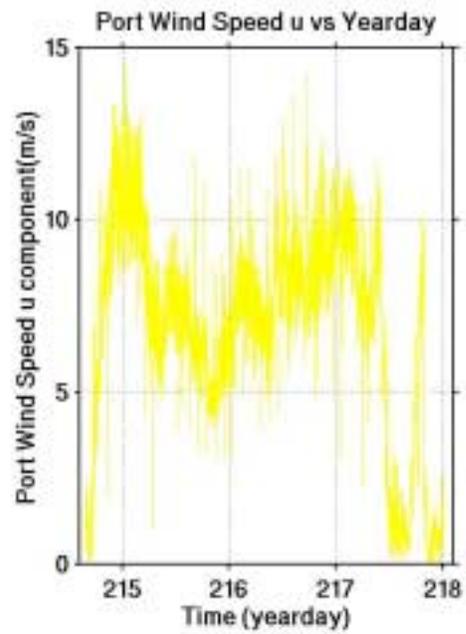
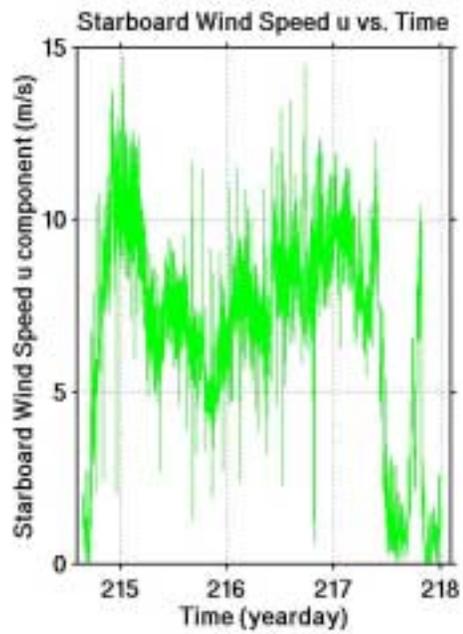
The dynamics of momentum through the surface layer is complex, due to the difficulty of making measurements. From the persistently northwesterly winds for the 4 days a net mass transport to the southwest was predicted based on Ekman theory. Uniquely this paper was able to show a net mass transport predominately to the southwest. Therefore, the outcome of this paper supports the Ekman theory. Also, based on the conclusion of a net loss out of the box, suggests there is upwelling occurring within the California Current region.

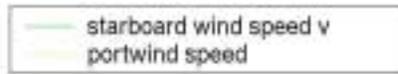
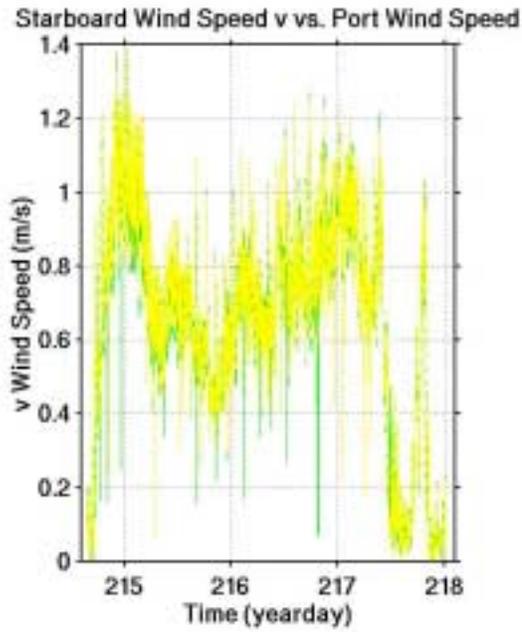
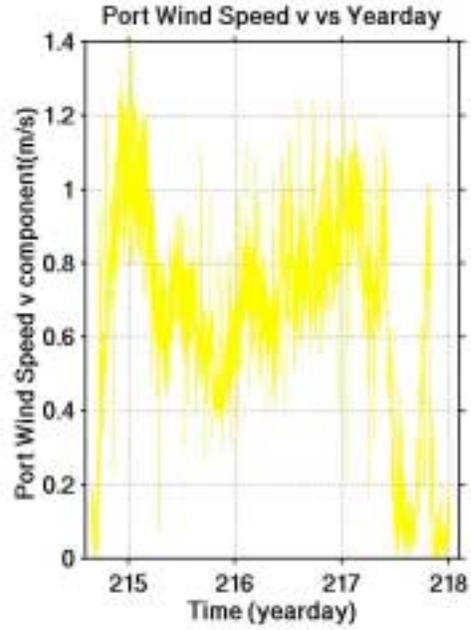
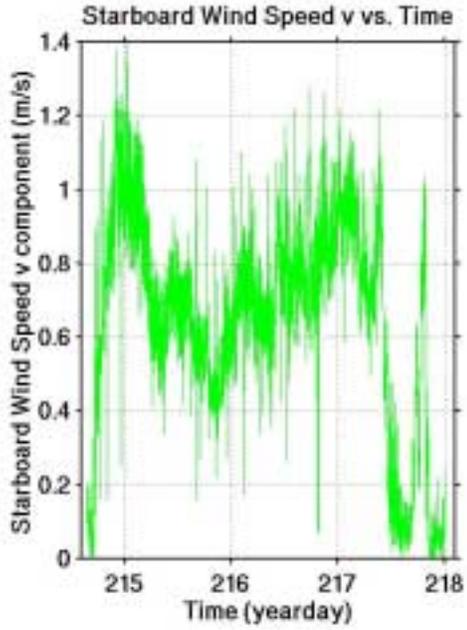
Works Cited

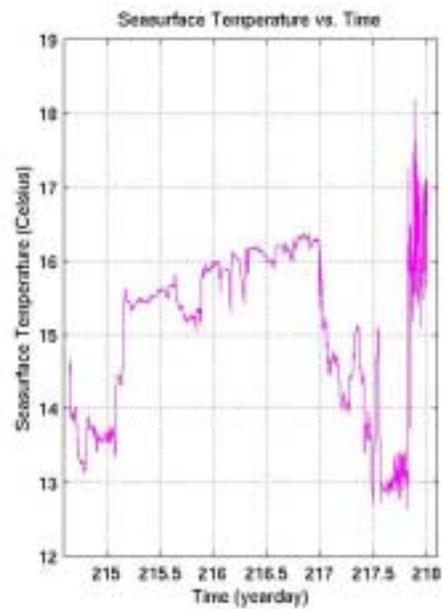
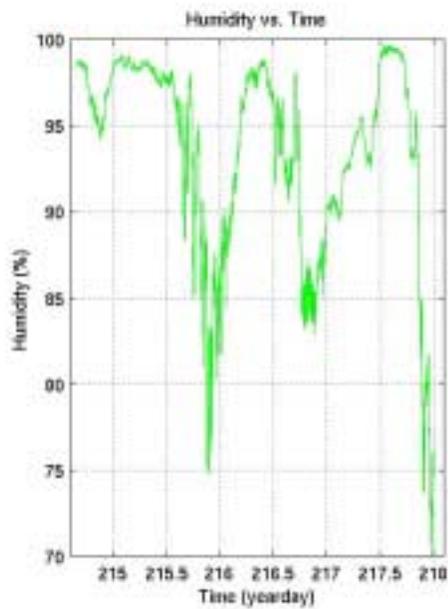
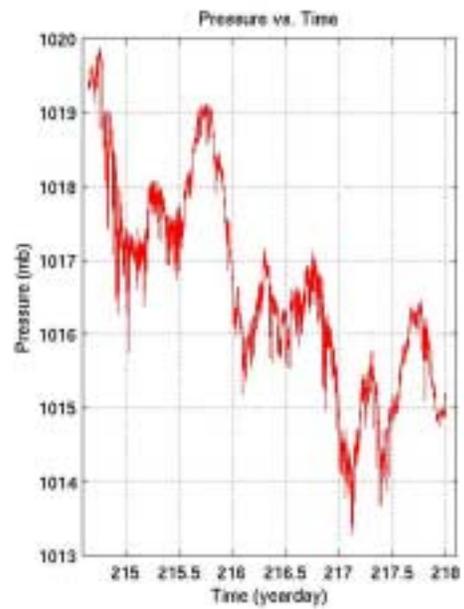
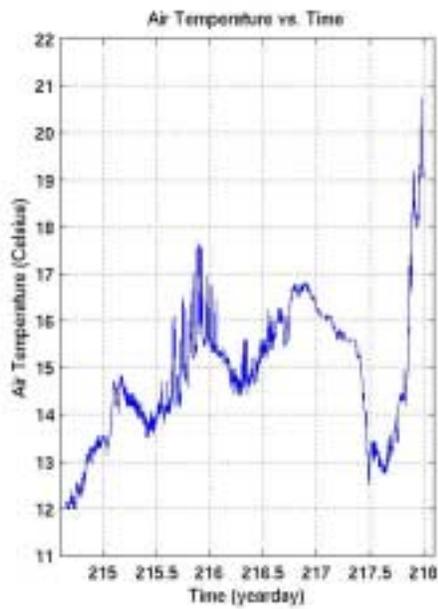
- Bendat, J. S., and A. G. Piersol, *Random Data: Analysis and Measurement Procedures*. John Wiley, New York, 2000.
- Bograd, S. J., Chereskin, T. K., Roemmich D., 2001: Transport of mass, heat, salt, and nutrients in the southern California current system: Annual cycle and interannual variability. *J. Geophys. Res.*, **106**, 9255-9274.
- Chereskin, T. K., 1995: Direct evidence for an Ekman Balance in the California Current. *J. Geophys. Res.*, **100**, 18,261-18269.
- Ekman, V. W., 1905: On the influence of the earth's rotation on ocean-currents. *Ark. Mat. Astron. Fys.*, **2**, 1-52.
- Guest, P. , sfcfluxoc3570.m, 1997. Matlab program.
- Pond, S. and Pickard, G. L., 1983: *Introductory dynamical oceanography*. Butterworth, Oxford, 1983.
- Price J. F., et. al., 1987: Wind-driven ocean currents and Ekman transport. *Science*, 238, 1534-1538.
- Smith, S.D. 1988: Coefficients of sea surface wind stress, heat flux, and wind profiles as a function of wind speed and temperature. *J. Geophys. Res.*, 93, 15,467-15,472.

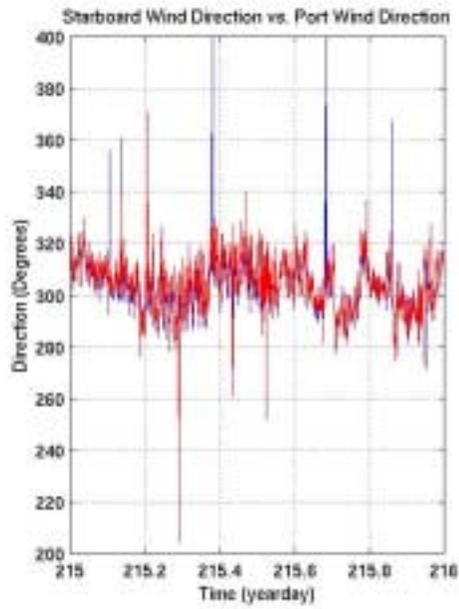
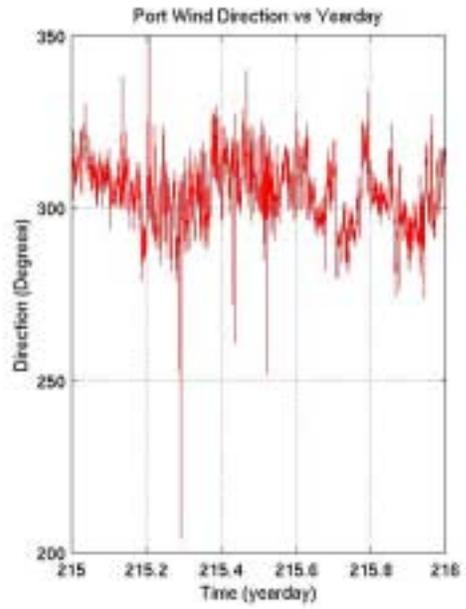
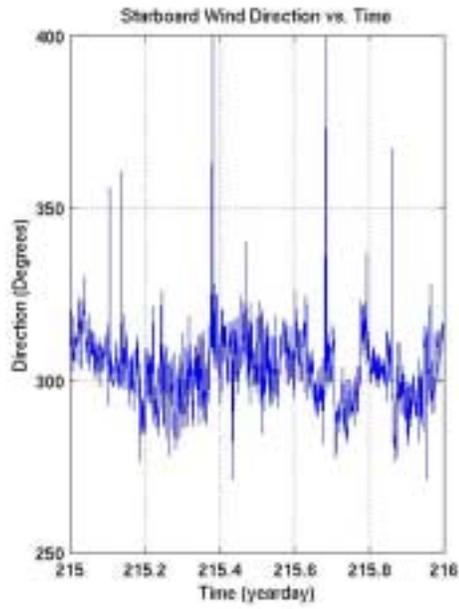


— starboard wind direction
— portwind direction

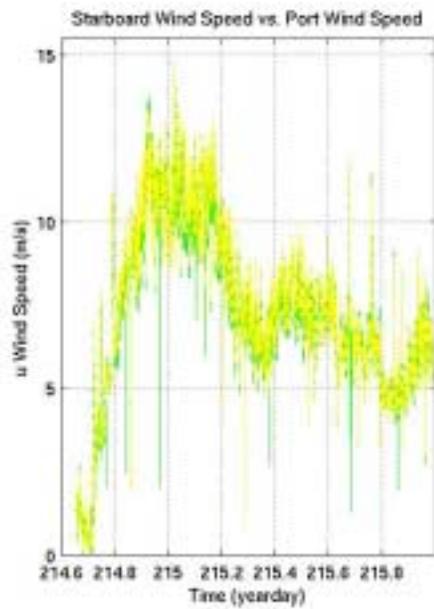
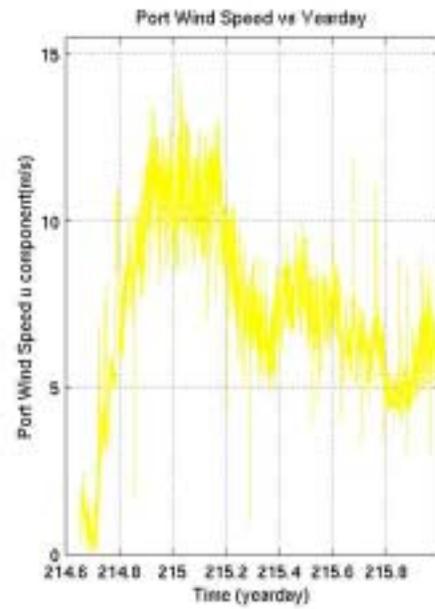
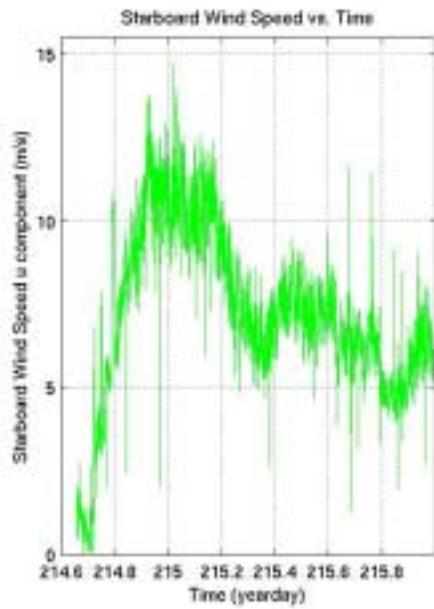




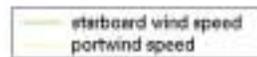
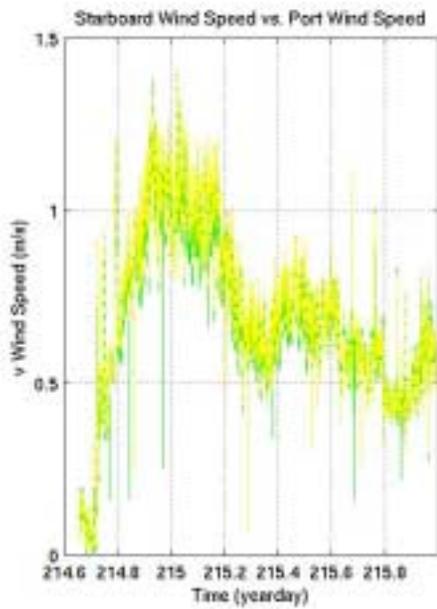
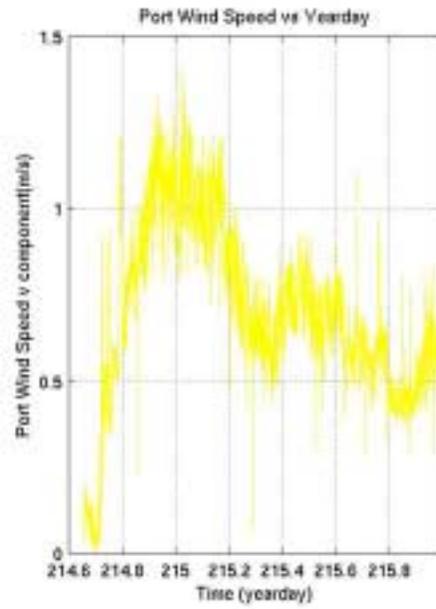
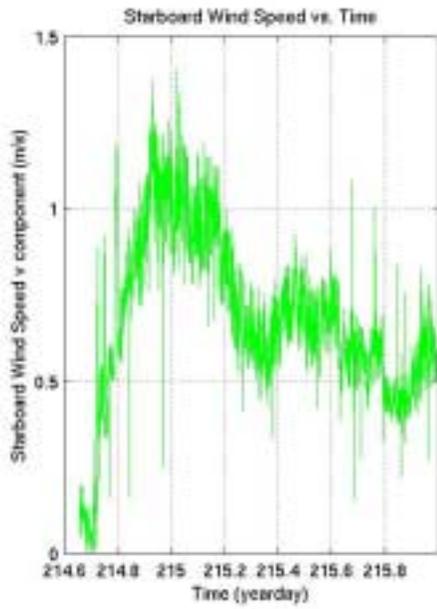


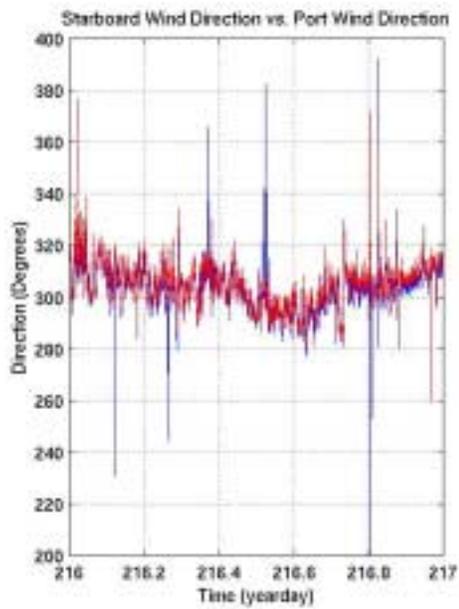
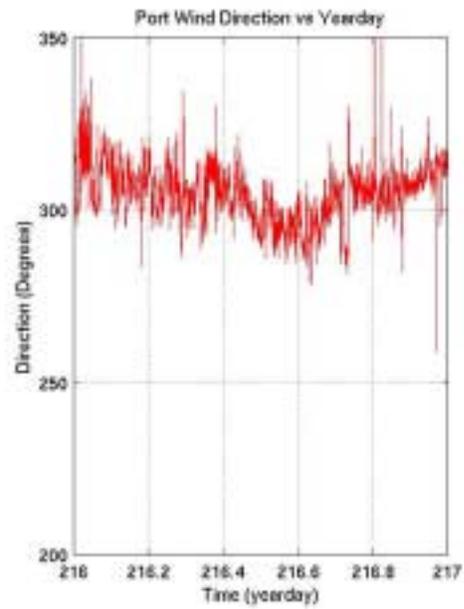
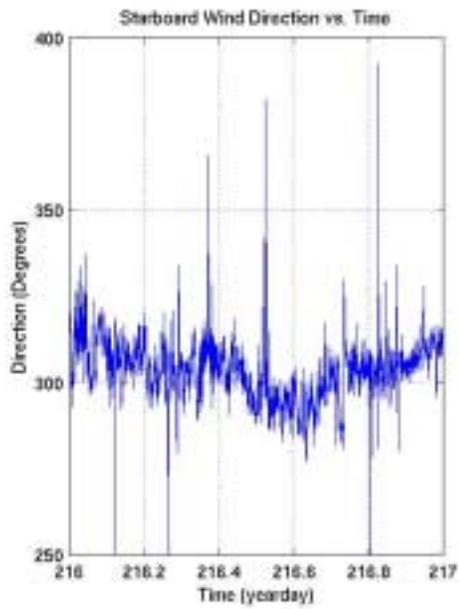


— starboard wind direction
— portwind direction

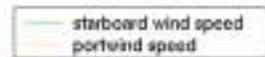
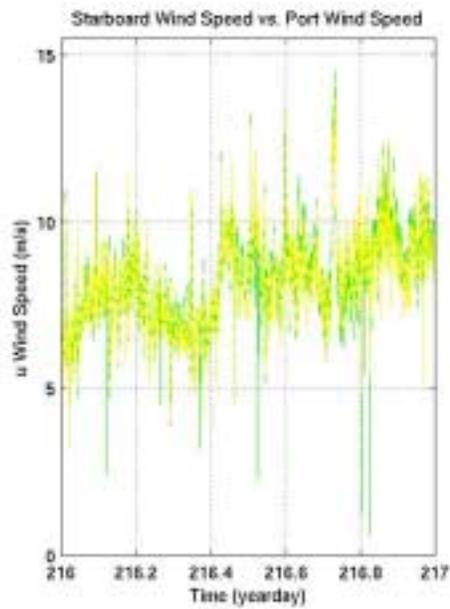
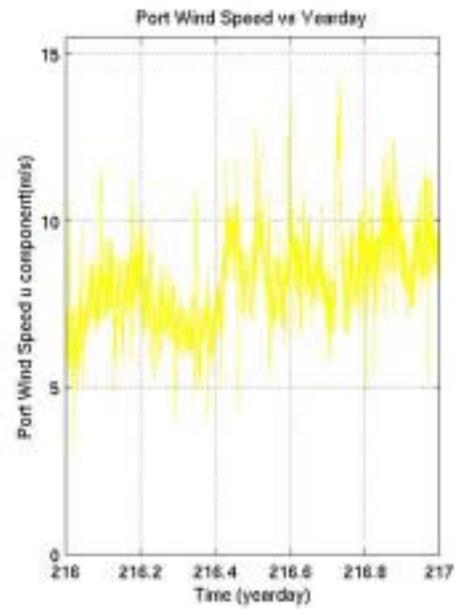
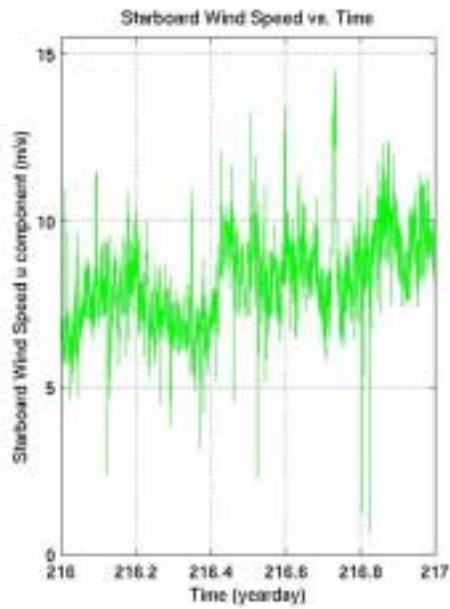


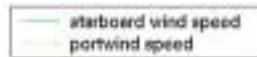
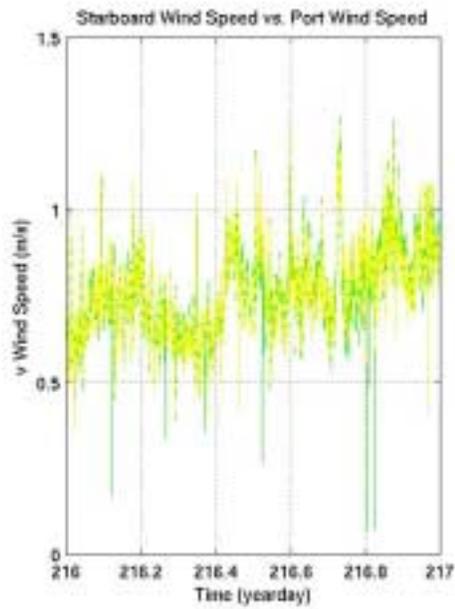
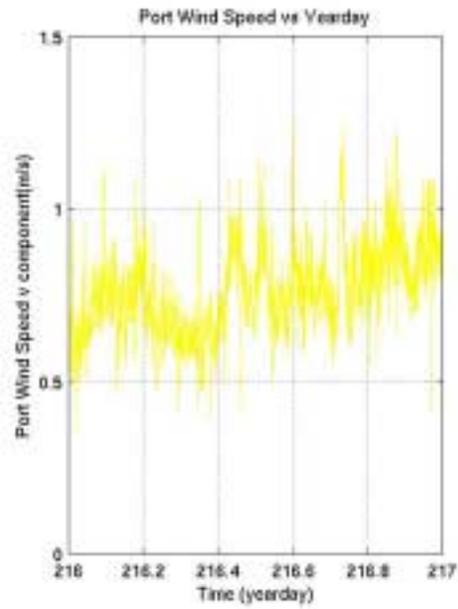
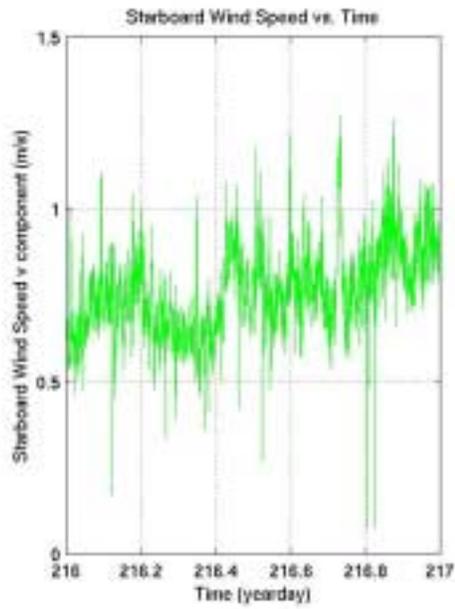
— starboard wind speed
— portwind speed

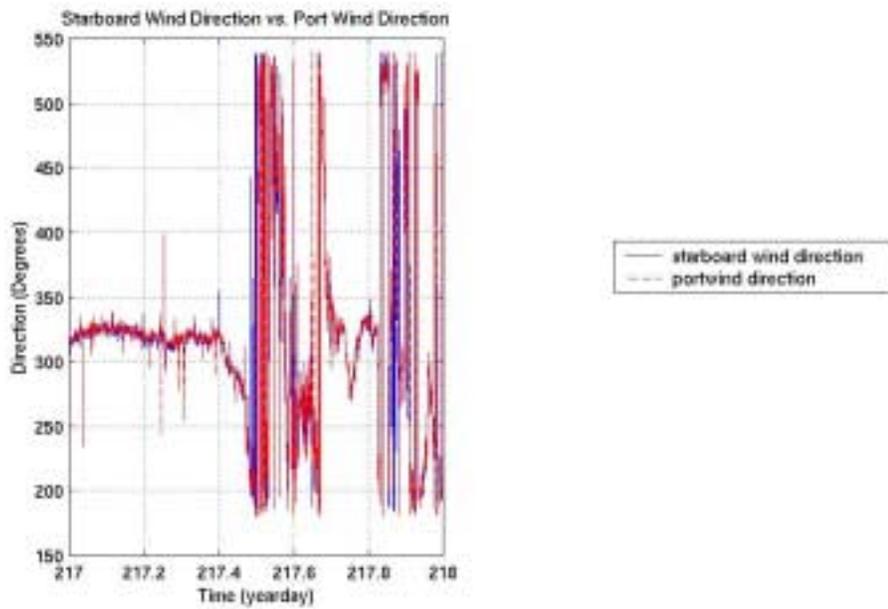
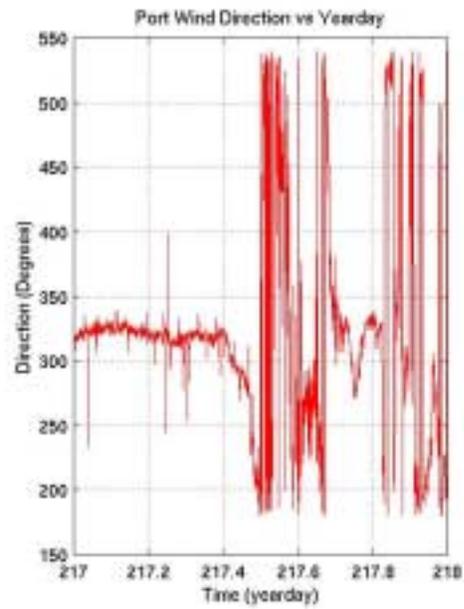
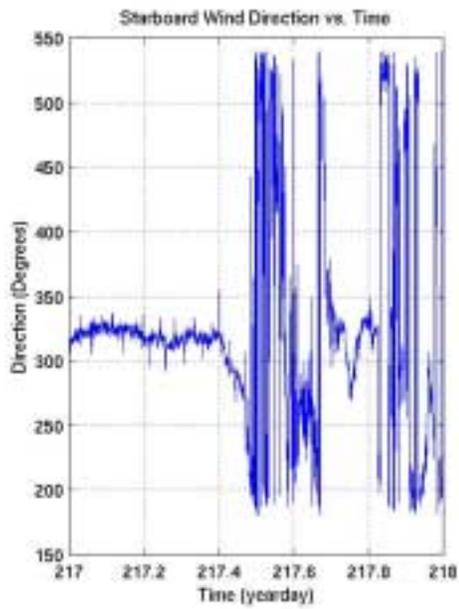


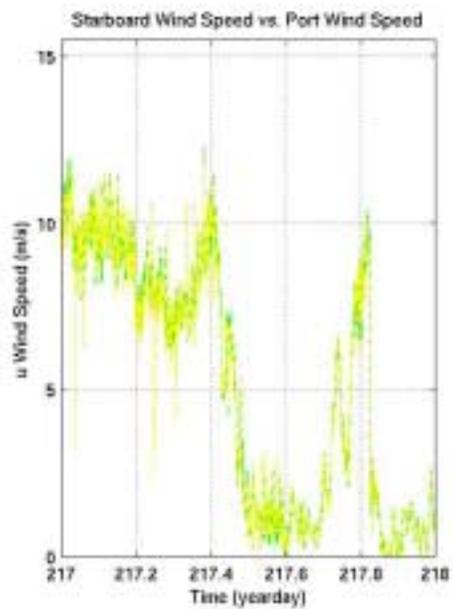
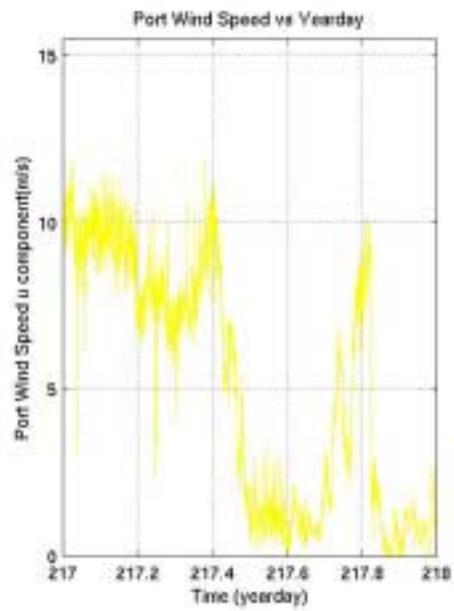
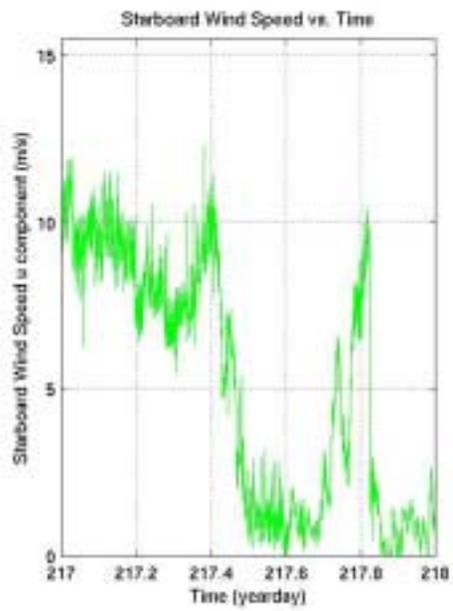


— starboard wind direction
 — portwind direction

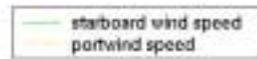
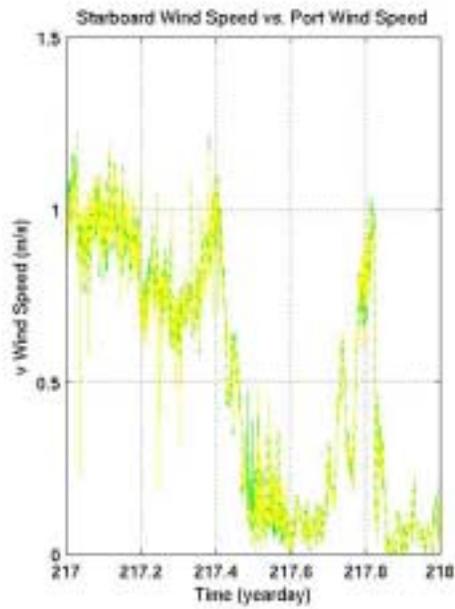
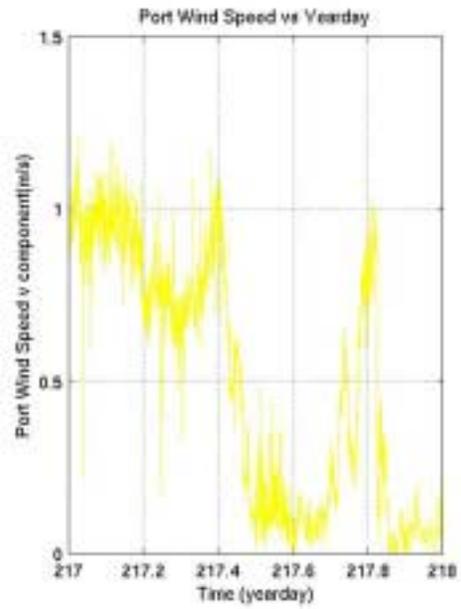
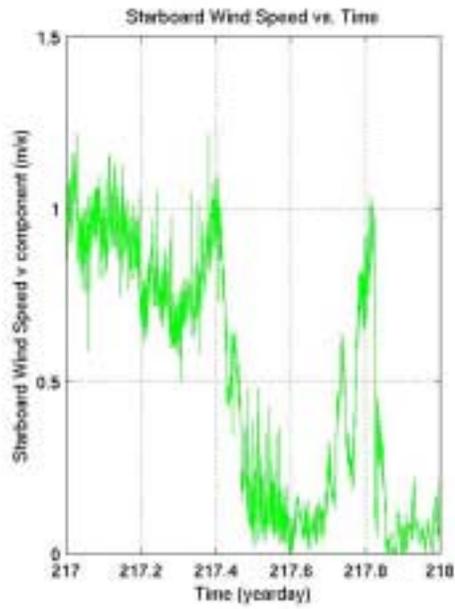








— starboard wind speed
— portwind speed



```

%filename = extract2_5.m
%
%Lora Egley
%29AUG01
%
%Purpose: Extract Position, time, windspeed and
%wind direction, air temperature, pressure, humidity and
%seasurface temperature from the sail data.
%
%Variables
%input:
%  sailyd      = yearday
%  lat         = latitude in decimal degrees
%  lon         = longitude West in decimal degrees
%  swinddir    = starboard wind direction (from)
%  swindspdms  = starboard wind speed m/s
%  pwinddir    = port wind direction(from)
%  pwindspdm   = port wind speed m/s
%  airtemp     = air temperature in Celsius
%  mbar        = pressure (millibars)
%  humidity    = humidity %
%  sst=        = seasurface temperature in Celsius
%
load aug2thru5.dat
sailyd=aug2thru5(:,1);
lat=aug2thru5(:,2);
lon=aug2thru5(:,3);
%Extract wind data.
swinddir=aug2thru5(:,4);
swindspdms=aug2thru5(:,5);
pwinddir=aug2thru5(:,6);
pwindspdms=aug2thru5(:,7);
airtemp=aug2thru5(:,8);
mbar=aug2thru5(:,9);
humidity=aug2thru5(:,10);
sst=aug2thru5(:,11);
output2_5=[sailyd(:),lat(:), lon(:), swinddir(:), swindspdms(:),
pwinddir(:), pwindspdms(:), airtemp(:), mbar(:), humidity(:),
sst(:)];
save aug02_05.dat output2_5 -ascii
save output2_5

```

```
%filename = result2_5.m
%
%Lora Egley
%31AUG01
%
%Purpose: Make .mat files with all parameters need to to
perform an analysis and computation for determining Ekman
Transport.
load output2_5
mathnav
mathoceansdir=mathswinddir+180;
mathoceanpdir=mathpwinddir+180;
result2_5 = [sailyd(:), lat(:), lon(:), mathoceansdir(:),
mathoceanpdir(:), uport(:), ustbd(:), vport(:), vstbd(:),
airtemp(:), mbar(:), humidity(:), sst(:)];
save result2_5.dat result2_5 -ascii
save result2_5 result2_5
```

```

%filename = navmath.m
%
%Lora Egley
%30AUG01
%
%Purpose:  Navigation to Math
%
%Variables
%mathswinddir = stbd math direction degrees
%mathpwinddir = port math directin degrees
%swinddir = stbd navigation direction degrees
%pwinddir = port navigatin direction degrees
%ustbd = stbd velocity x componenet m/s
%vstbd = stbd velocity y componenet m/s
%swindspdms = stbd wind speed m/s
%uport = port velocity x component m/s
%vport = port velocity y component m/s
%pwindspdms = port wind speed
%extract2_5
%Convert wind direction
%TRUE2MATH converts angle in degrees from the true system to math
system.
%   MATHDIR = TRUE2MATH(TRUEDIR) converts angle in degrees

from the true

% convention to the math convention.
% I call the math convention, for lack of a better term, the direction
of a
% vector angle measured in degrees ccw from east, where east = 0
degrees,
% north = 90 degrees, etc.
% true convention is the direction of a vector angle measured in
degrees
% cw from north, where north = 0 degrees, east = 90 degrees, etc.
This is
% also commonly referred to as compass angle.
mathswinddir = true2math(swinddir);
mathpwinddir=true2math(pwinddir);
%Compute starboard wind speed x and y components
%SPDDIR2UV convert speed in degrees and direction to u/v velocities.
%   [U,V] = SPDDIR2UV(SPEED,DIR) converts SPEED/DIR to

U,V. It is assumed

% that direction is in what I call for a lack of a better term the
math
% system. So the direction, DIR, is the vector angle measured in
degrees
% ccw from east, where east = 0 degrees, north = 90 degrees.
% NOTE: Uses the formula
%       u = abs(speed).*cos(radians(dir))
%       v = abs(speed).*sin(radians(dir))
% NOTE: It is assumed that the angles are in degrees 0 < angle <
360.

```

```
%Add 180 degrees to mathwinddir to go from the meteorology to
%oceanography convention.
[ustbd,vstbd]=spddir2uv(swindspdms,radians(mathswinddir+180));
[uport,vport]=spddir2uv(pwindspdms,radians(mathpwinddir+180));
```

```

%filename = analysis1.m
%
%Lora Egley
%16SEP01
%
%Purpose: Perform an analysis and computation for %determining Ekman Transport.
load result2_5mod
yd = result2_5mod(:,1);
lat = result2_5mod(:,2);
lon = result2_5mod(:,3);
dirstbd = result2_5mod(:,4);
dirport = result2_5mod(:,5);
uleft = result2_5mod(:,6);
uright= result2_5mod(:,7);
vleft = result2_5mod(:,8);
vright = result2_5mod(:,9);
temp = result2_5mod(:,10);
pressure = result2_5mod(:,11);
hum = result2_5mod(:,12);
sfctemp = result2_5mod(:,13);
%%Direction Plots
figure(1)
subplot(2,2,1)
plot(yd,temp,'b')
grid on
xlabel('Time (yearday)')
ylabel('Air Temperature (Celsius)')
title('Air Temperature vs. Time')
axis([214.6 218.1 11 22])
niceplot
%
subplot(2,2,2)
plot(yd,pressure,'r')
grid on
xlabel('Time (yearday)')
ylabel('Pressure (mb)')
title('Pressure vs. Time')
axis([214.6 218.1 1013 1020])
niceplot
%
subplot(2,2,3)
plot(yd,hum,'g')
grid on
xlabel('Time (yearday)')
ylabel('Humidity (%)')
title('Humidity vs. Time')
axis([214.6 218.1 70 100])
niceplot
%
subplot(2,2,4)
plot(yd,sfctemp,'m')
grid on
xlabel('Time (yearday)')
ylabel('Seasurface Temperature (Celsius)')
title('Seasurface Temperature vs. Time')
axis([214.6 218.1 12 19])
niceplot

```

```

%filename = legla.m
%
%Lora Egley
%6SEP01
%
%Purpose: Extract Position, time, windspeed and
%wind direction, air temperature, pressure, humidity and
%seasurface temperature from the sail data.
%Convert data into appropriate units.
%
%Variables
%input:
%time:                parameters:
% month = month's      swinddir  = starboard wind direction
(from)
% day   = day          swindspdkt = starboard wind speed knots
% year  = year         pwinddir  = port wind direction(from)
% hr    = hour         pwindspdkt = port wind speed knots
% min   = minute      swindspdms = starboard wind speed m/s
(utru)
% sec   = second      pwindspdms = port wind speed m/s
% sailyd = yearday    airtemp   = air temperature in
Celsius(tair)
%
%                mbar       = pressure (millibars) (p)
%                humidity   = humidity (rh)
%                sst=       = seasurface temperature in
Celsius(tsfcsea)
%position used 1st report Furuno:
% latdeg   = latitude degree
% latmin   = latitude minutes
% londeg   = longitude degree
% lonmin   = longitude minutes
% lat      = latitude in decimal degrees
% longitude = longitude in decimal degrees
% lon = longitude West
%
load legla.dat
%Extract
%Extract drifter dates and times.
month=legla(:,1);
day=legla(:,2);
year=legla(:,3);
hr=legla(:,4);
min=legla(:,5);
sec=legla(:,6);
%Convert sail data: year/month/day/hour/min/sec to %yearday.
Local Oceanography function

```

```

sailyd=cal2yd([year,month,day,hr,min,sec]);
%
%Extract Latitude and Longitude from a Sail Data.
%
latdeg=legla(:,10);
latmin=legla(:,11);
londeg=legla(:,12);
lonmin=legla(:,13);
%Convert latitude and longitude to decimal degree.
lat=(latmin/60)+latdeg;
longitude=(lonmin/60)+londeg;
%Convert Longitude to West (negative number)
lon=longitude*(-1);
%
%Extract wind data.
%
swinddir=legla(:,24);
swindspdkt=legla(:,25);
pwinddir=legla(:,28);
pwindspdkt=legla(:,29);
%Convert wind speed from knots to meter/second.
utru=swindspdkt*.514444;
pwindspdms=pwindspdkt*.514444;
tair=legla(:,31);
p=legla(:,32);
rh=legla(:,34);
tsfcsea=legla(:,36);
%
sfcfluxoc3570mod
%
leglaoutput=[sailyd(:),lat(:), lon(:), swinddir(:), utru(:),
pwinddir(:), pwindspdms(:), tair(:), p(:), rh(:),
tsfcsea(:),tau(:)];
save leglaoutput.dat leglaoutput -ascii
save leglaoutput leglaoutput

```

```

% sfcfluxoc3570mod.m
% calculates surface fluxes and other parameters
% used for open water or open lead calculations
% based on ustar, tstar, qstar and L

```

```

% Peter Guest 4/3/97 modified for input 8/1/2001
%
%
% Input:
%      utrue  wind speed (m/s) at z(1)
%      tair   air temperature (Centigrade) at z(2)
%      rh     relative humidity (%) at z(3)
%      tsfcsea surface temperature (Centigrade)
%      p      atmospheric pressure at z(4) (mb)
%      z      measurement level (can be vector)
%
% stars output:
%      ustar  friction velocity (m/s)
%      tstar  scaling temperature (k or c)
%      qstar  scaling specific humidity (g/kg) not (g/g)!!!!
%      L      monin-obukov length scale (m)
%
% sfcflux output:
%
%      shf    sensible heat flux
%      lhf    latent heat flux
%      hf     (total) heat flux
%      wstar  free convection scaling length (needs zi)
%      tau    wind stress
%      Cd     drag coeff
%      Ce     Dalton number
%      Cdn    neutral drag coeff
%      Ch     Stanton number
% plus a bunch of other stuff

%      lv     specific heat set constant at 1004 J kg-1 k-1
%      cp     latent heat of vaporization set constant at 2.50e+6 J kg-1

z = [14];
%file that brings in input parameters

[ustar tstar qstar L] = starsmod(utrue,tair,tsfcsea,rh,p,z);

% Set some stuff again for later calculations

if p == [] % Default pressure
    p = 1012;
end
pbad = find(isnan(p));
p(pbad) = 1012 * ones(size(pbad));

if length(z) == 0 % Default measurement heights
    zu = 14.;
    zt = 14.;
    zq = 14.;
    zp = 14.;
elseif length(z) == 1
    zu = z(1);
    zt = z(1);
    zq = z(1);

```

```

    zp = z(1);
elseif length(z) == 2
    zu = z(1);
    zt = z(2);
    zq = z(2);
    zp = z(2);
elseif length(z) == 3
    zu = z(1);
    zt = z(2);
    zq = z(3);
    zp = z(2);
elseif length(z) == 4
    zu = z(1);
    zt = z(2);
    zq = z(3);
    zp = z(4);
else
    disp ('Incorrect z array')
end

z10=10.0;      % reference height for Obukhov length scale
CHn10=1.0e-3;  % heat and buoyancy flux transfer parameter at z10=10 (Smith,
1988)
CEn10=1.2e-3; % humidity flux transfer parameter at z10=10 (Smith, 1988)
gamma=0.00975; % adiabatic lapse rate
k=0.4;        % Von Karmen's constant
g=9.81;       % gravity

% Calculate mixing ratios
psfc = (p + 0.116*zp); % surface pressure (mb) based on standard atms
ess = esat(tsfcsea);   % saturation vapor pressure wrt flat water
qsat = 622.0.*ess./(psfc - ess); % saturation mixing ratio (g/kg)
%qsfc = qsat;          % mixing ratio at surface assumed saturated
qsfc = qsat .* 0.98;  % for salt water (different from Smith)
esa = esat(tair);     % saturation vapor pressure wrt flat water
esair = (rh./100).*esa;
q = 622.0.*esair./( p - esair); % true mixing ratio at zt (g/kg)
%q = 622.0.*(rh/100).*esa./( p - esa); % mixing ratio at zt (g/kg)
%esai = esati(tair);
%rhi = rh.*esa./esai;

% potential air temperature (K) measured at zt
theta=tair+273.15+gamma.*zt;
thsfc=tsfcsea+273.15;

% virtual potential temp (K) based on Stull (1988)
thetav=theta.*(1.0+0.61e-3*q); % virtual potential air temp
thetavsfc=thsfc.*(1.0+0.61e-3*qsfc); % virt pot temp at sfc

% find roughness lengths
zo= zu.*exp(-utru.*k./ustar - psimsmith(zu,L));
zot = zt.*exp((thsfc-theta).*k./tstar - psitsmith(zt,L));
zoq = zq.*exp((qsfc-q).*k./qstar - psitsmith(zq,L));
zotl = z10.*exp(-k.*k./(CHn10*log(z10./zo))); % Using constant CHn10
(Smith,1988)

```

```

zoq1 = z10.*exp(-k.*k./(CEn10*log(z10./zo))); % Using constant CEn10
(Smith,1988)

% find reference height reference ht 14m (usually 10 meters) values
u10=ustar./k.*(log(z10./zo)-psimsmith(z10,L));
th10=thsfc+(tstar/k).*(log(z10./zot)-psitsmith(z10,L)); % Kelvin
t10=th10-273.16-z10.*gamma; % C
q10=qsfc+(qstar./k).*(log(z10./zoq)-psitsmith(z10,L));

% Calculate surface fluxes
rho=psfc./(2.87.*thetavsfc); % calc density with ideal gas law
shf=-rho.*1004.*ustar.*tstar; % sensible heat flux
lhf=-rho.*2.5e3.*ustar.*qstar; % latent heat flux
cd=(ustar./u10).^2; % Drag Coeff
cdn=(1./(cd).^0.5)+psimsmith(z10,L)./k.^(-2); % neutral Drag Coeff
ce=ustar.*qstar./(u10.*(q10-qsfc)); % Dalton number
cen=k^2./(log(z10./zo).*log(z10./zoq)); % Neutral Z10 m moisture transfer
coeff.
ch=ustar.*tstar./(u10.*(th10-thsfc));
chn=k^2./(log(z10./zo).*log(z10./zot)); % Neutral Z10 m heat transfer
coeff.
z1=z10./L;
tau=rho.*(ustar.^2); %wind stress
taubad=find(isnan(tau)); % replacing any possible complex NAN
tau = markbad(tau,taubad);
%calculating the Richardson Number
Ri=z10./L;
j=find(z10./L>0);
if j~=[];
    Ri(j)=(z10./L(j)).*(0.74+(4.7.*(z10./L(j))))./((1+(4.7
.*(z10./L(j))))).^2);
end;

%calculating the free convection length scale (wstar)
zi = 500;
tstarv=tstar+(0.61e-3.*theta.*qstar);
wstar=(-g./theta.*tstarv.*ustar*zi).^(1/3);
j=find((tstarv) >= 0); % setting stable to NAN (wstar not applicable)
if j~=[];
    wstar = markbad(wstar,j);
end;

%function [taux,tauy] = tauxtauy(tau,swinddir)
%tauxtauy converts tau and direction (meteorology %convention)to
taux and tauy components.

```

```
%  
%Convert direction to radians.  
dir=swinddir-240;  
dirr=dir.*(pi./180);  
taux=tau.*cos(dirr);  
tauy=tau.*sin(dirr);  
tauleg1 = [taux(:), tauy(:)];  
save tauleg1.dat tauleg1 -ascii
```

```
%filename = tauxhrave.m  
%Leg 1a  
%Lora Egley  
%8Sep01
```

```

%Variables:
% density = 1024kg/m^3
% f      = coriolis parameter midlat (s^-1)
% taux   = stress component along 240 degrees (kg/ms^2)
% meantauxXXXX = average of taux at beginninng of hr to hr(kg/ms^2)
% MXXXX  = Hourly Mass transport (m^2/s)
%
load final1.dat
density=1024;
f=.0001;
%1
taux1545=final1(1:67,4);
meantaux1545=mean(taux1545);
M1545=meantaux1545./(density*f);
%2
taux1645=final1(68:134,4);
meantaux1645=mean(taux1645);
M1645=meantaux1645./(density*f);

%3
taux1745=final1(135:201,4);
meantaux1745=mean(taux1745);
M1745=meantaux1745./(density*f);

%4
taux1845=final1(202:267,4);
meantaux1845=mean(taux1845);
M1845=meantaux1845./(density*f);

%5
taux1945=final1(268:334,4);
meantaux1945=mean(taux1945);
M1945=meantaux1945./(density*f);

%6
taux2045=final1(335:401,4);
meantaux2045=mean(taux2045);
M2045=meantaux2045./(density*f);

%7
taux2145=final1(402:467,4);
meantaux2145=mean(taux2145);
M2145=meantaux2145./(density*f);

%8
taux2245=final1(468:534,4);
meantaux2245=mean(taux2245);
M2245=meantaux2245./(density*f);

%9
taux2345=final1(535:601,4);
meantaux2345=mean(taux2345);
M2345=meantaux2345./(density*f);

%10
taux0045=final1(602:667,4);
meantaux0045=mean(taux0045);

```

```

M0045=meantaux0045./(density*f);

%11
taux0145=final1(668:734,4);
meantaux0145=mean(taux0145);
M0145=meantaux0145./(density*f);

%12
taux0245=final1(735:801,4);
meantaux0245=mean(taux0245);
M0245=meantaux0245./(density*f);

%13
taux0345=final1(802:867,4);
meantaux0345=mean(taux0345);
M0345=meantaux0345./(density*f);

%14
taux0445=final1(868:934,4);
meantaux0445=mean(taux0445);
M0445=meantaux0445./(density*f);

%15
taux0545=final1(935:1001,4);
meantaux0545=mean(taux0545);
M0545=meantaux0545./(density*f);

%16
taux0645=final1(1002:1067,4);
meantaux0645=mean(taux0645);
M0645=meantaux0645./(density*f);

%17
taux0745=final1(1068:1134,4);
meantaux0745=mean(taux0745);
M0745=meantaux0745./(density*f);

%18
taux0845=final1(1135:1201,4);
meantaux0845=mean(taux0845);
M0845=meantaux0845./(density*f);

%19
taux0945=final1(1202:1267,4);
meantaux0945=mean(taux0945);
M0945=meantaux0945./(density*f);

%20
taux1045=final1(1268:1334,4);
meantaux1045=mean(taux1045);
M1045=meantaux1045./(density*f);

%21
taux1145=final1(1335:1401,4);
meantaux1145=mean(taux1145);
M1145=meantaux1145./(density*f);

```

```

%22
taux1245=final1(1402:1427,4);
meantaux1245=mean(taux1245);
M1245=meantaux1245./(density*f);

Mtransportoutput = [M1545(:), M1645(:), M1745(:), ...
M1845(:), M1945(:), M2045(:), M2145(:), M2245(:), ...
M2345(:), M0045(:), M0145(:), M0245(:), M0345(:),...
M0445(:), M0545(:),M0645(:),M0745(:),M0845(:),...
M0945(:), M1045(:), M1145(:), M1245(:)];
Mtrans = Mtransportoutput';
save Mtrans.dat Mtrans -ascii

meantauxoutput = [meantaux1545(:), meantaux1645(:),...
meantaux1745(:), meantaux1845(:), meantaux1945(:),...
meantaux2045(:), meantaux2145(:), meantaux2245(:),...
meantaux2345(:), meantaux0045(:), meantaux0145(:),...
meantaux0245(:), meantaux0345(:), meantaux0445(:),...
meantaux0545(:), meantaux0645(:), meantaux0745(:),...
meantaux0845(:), meantaux0945(:), meantaux1045(:),...
meantaux1145(:), meantaux1245(:)];
meantauxleg1= meantauxoutput';

meantauxMleg1=[meantauxleg1(:), Mtrans(:)];

save meantauxMleg1.dat meantauxMleg1 -ascii

```

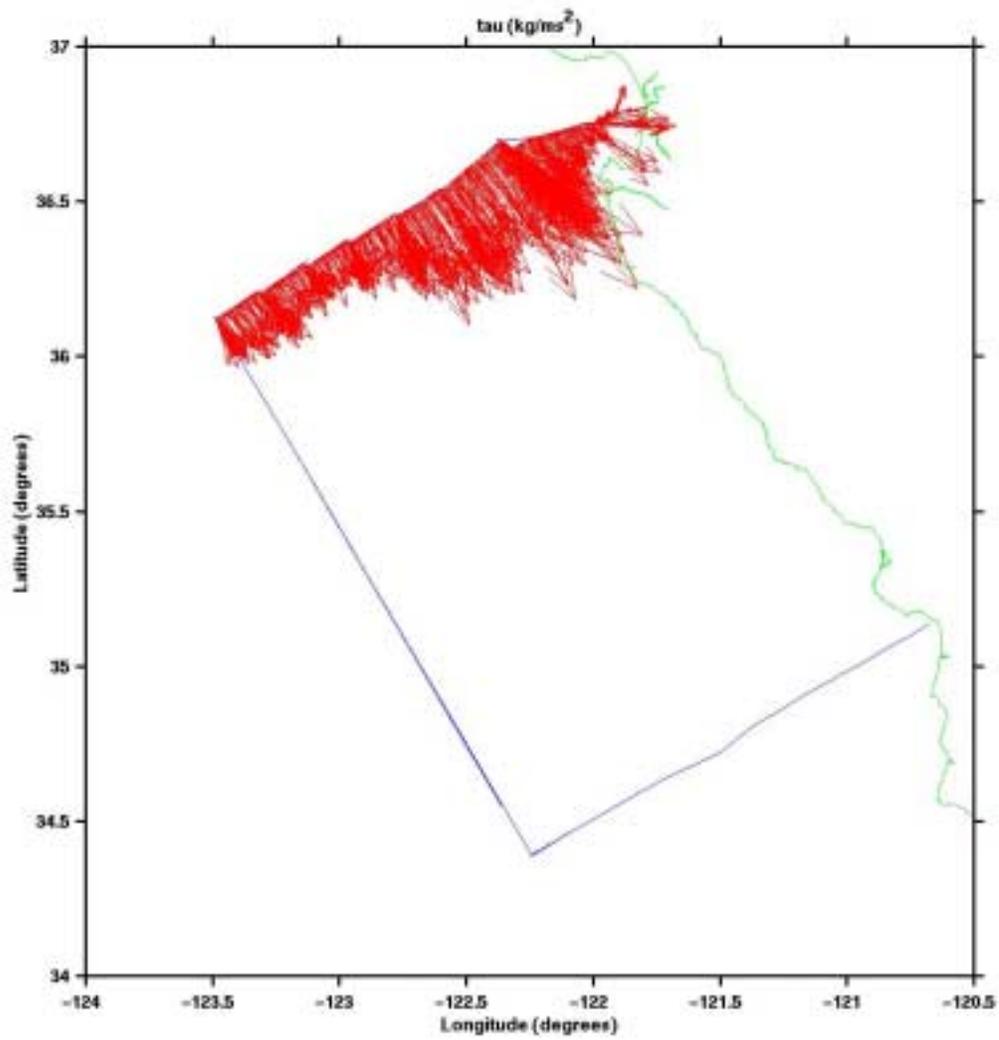
```

%filename=autocorr.m
%
%load windspeed data
load leglaoutput.dat

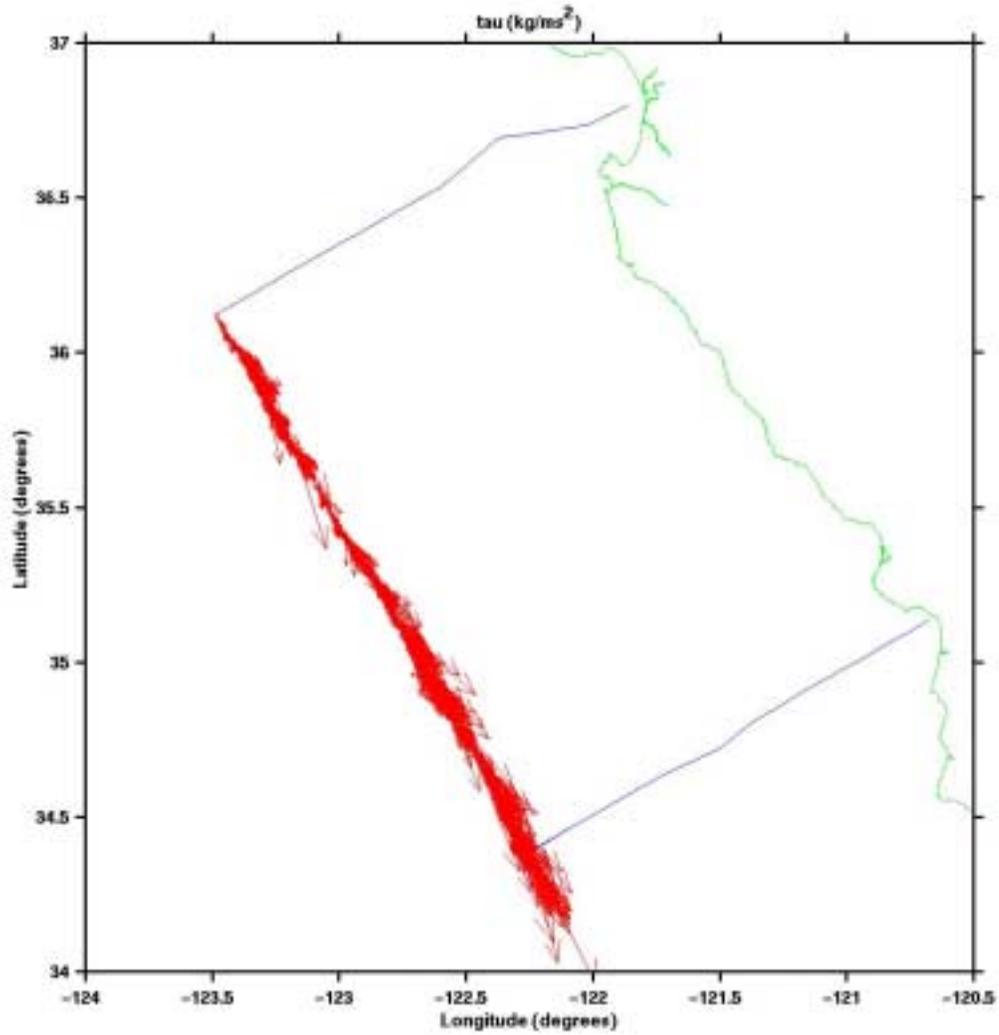
```

```
uttrue=leglaoutput(:,5);
%XCORR(A), when A is a vector,
%is the auto-correlation sequence.
b=xcorr(uttrue);
figure; plot(b)
%integral time scale (I)
%wind data sampled sampled ~every 56sec
%Integrated correlation function and divide by
%correlation function at tau=0
dt=56 %timestep in seconds
I=sum(b)*(56)/max(b)
```

NOTE:
RAW DATA
CalCOFI
Line 67



NOTE:
RAW DATA
USED.
CaCOF1
Line 70



NOTE:
RAW DATA
USED.
CaCO₃
LINE 77

