Fluid[1], Python[2] procedures for atmospheric and ocean sciences

Guilherme P. Castelao[3] Roberto A. F. de Almeida[4]

July 21, 2005

[1]http://fluid.oceanografia.org
[2]http://www.python.org/pypi
[3]RSMAS–MPO, U. Miami, guilherme@castelao.net
[4]IO–USP ...

Abstract

Python is a program language that have been reaching space in scientific community during last years. Packages to plot data and apply statistical analysis in Python language have been developed but nothing that resolve basic transformations and calculus related to atmospheric and atmospheric–ocean interface. This package is developed intending to fill that role, offering functionalities as estimative of saturation vapor pressure and long wave heat balance at ocean surface. No new theory is proposed, the main focus is group those functions in a coherent set optimizing the program time for scientific community.

This project was partially funded by CAPES–Brasil. Thanks to Prof. Dr. Bruce Albrecht for stimulate to finally start this project planned for a long time.

Contents

List of Figures

Chapter 1

Introduction

Python is a program language that have been carving its niche inside scientific community. Some reasons for that is interpreted, interactive, object-oriented program ing language the multi–platform capability, fast programming, ability to work with faster compiled languages as C and FORTRAN and interaction with consecrated scientific packages as R statistical language or Generic Mapping Tools. Many scientific packages have been developed in Python, but nothing yet that reunify basic functions related to ocean and atmosphere. This work propose to cove this fault, with routines, for example, to convert vapor pressure to mixing ratio, estimate equivalent neutral wind for different heights or even estimate latent heat flux trough ocean surface by bulk formulae.

The theory related to those relations isn't in general mode hard complex, but a development of a solid package that reunify all those could considerably improve the work of many researchers optimizing the programing spent time. Another advantage is to made easy to keep scientific community updated and check for programming errors. This package was strongly inspired, and sometimes is a simple translate, of COARE (Fairall et al., 1996) and Air–Sea (Pawlowicz et al., 2001).

This package is freely distributed by Gnu Program's License and have a fast review of theory related to each function. This document was created planning to be more like an initial manual or technical reference of Fluid package than an article, as the reader would see. Therefore, comments and suggestions are welcome, please contact the author.

Chapter 2

Functions

This initial version is still based on groups of functions as described below. A conversion to classes is planned for future versions.

Functionalities are grouped by great themes: Atmosphere, Interaction and Commons. A group of oceanographic related functions, called Ocean, is planned to be included soon.

2.1 Variables

Variables used are defined as below:

e**:** Vapor pressure

e_s**:** Saturation vapor pressure

Le**:** Latent heat of vaporization

q**:** Specific humidity

Q_{sw}**:** Short wave heat flux

Q_{sw_i}**:** Short wave heat flux incident

Q_{lw}**:** Long wave heat flux

R_r**:** Reynolds roughness

R_t**:** Reynolds roughness for temperature

R_q**:** Reynolds roughness for humidity

RH**:** Relative humidity

T_a**:** Air temperature

T_s**:** Sea temperature

T_v**:** Virtual temperature

T_{ss}**:** Sea temperature at surface

u_***:** Friction velocity

U**:** Horizontal wind magnitude

U_{10}**:** Horizontal wind magnitude at 10 meters

w**:** Mixture ratio

w_s**:** Mixture ratio

z**:** height

z_0**:** Momentum roughness length

z_{0t}**:** Temperature roughness length

z_{0q}**:** Humidity roughness length

Ψ**:** Stratification function

Ψ_t**:** Stratification function for temperature

Ψ_q**:** Stratification function for humidity

ρ_{air}**:** Air density

θ**:** Potential Temperature

2.2 Atmosphere

2.2.1 Air density

$$\rho_{air} = \frac{p}{R_{gas} T_v} \tag{2.1}$$

2.2.2 Air viscosity

$$\begin{aligned} \nu &= 1.326 \times 10^{-5} \left(1 + 6.542 \times 10^{-3}\ T_a + \right. \\ & \left. + 8.301 \times 10^{-6}\ T_a^2 - 4.84 \times 10^{-9}\ T_a^3 \right) \end{aligned} \tag{2.2}$$

2.2.3 Saturation vapor pressure

For values of temperature in Celsius degree between −30 and 50.

$$e_s = 6.112\ \exp\left(\frac{17.67\ T_a}{T_a + 243.5}\right) \tag{2.3}$$

2.2.4 Mixing ratio

$$w = \epsilon \frac{e}{p - e} \tag{2.4}$$

2.2.5 Saturation Mixing ratio

From Wallace and Hobbs (1977), page 73:

$$w_s = \epsilon \frac{e_s}{p - e_s} \tag{2.5}$$

2.2.6 Potential temperature

From Wallace and Hobbs (1977), page 69:

$$\theta = T \left(\frac{p_0}{p} \right)^{R/c_p} \tag{2.6}$$

2.2.7 Relative Humidity

From Wallace and Hobbs (1977), page 73:

$$RH = 100 \frac{w}{w_s} \tag{2.7}$$

2.2.8 Equivalent potential temperature

From Wallace and Hobbs (1977), page 79:

$$\theta_e = \theta \, \exp \left(\frac{L \; w_s}{c_p \; T} \right) \tag{2.8}$$

2.2.9 Specific Humidity

$$q = \frac{w}{1 + w} \tag{2.9}$$

2.2.10 Vapor pressure

From Wallace and Hobbs (1977), page 71:

$$e = \frac{w}{w + \epsilon} p \tag{2.10}$$

2.2.11 Virtual temperature

From Wallace and Hobbs (1977), page 52:

$$T_v = \frac{T}{1 - \frac{e}{p}(1 - \epsilon)} \tag{2.11}$$

Alternative solution on page 72:

$$T_v = T \frac{\epsilon + w}{\epsilon(1 + w)} \tag{2.12}$$

neglecting terms in w^2 and higher orders

$$T_v = T\left(1 + \frac{1 - \epsilon}{\epsilon} w\right) = T(1 + 0.61w)$$

where $\epsilon = 0.622$

2.2.12 RH2w

Convert relative humidity to mixing ratio

$$w = RH * w_s \tag{2.13}$$

2.2.13 RH2e

Convert relative humidity to vapor pressure by first section 2.2.12 than section 2.2.10.

2.2.14 q2w

Convert specific humidity to mixing ratio.

$$w = \frac{q}{1 - q} \tag{2.14}$$

2.3 Common

2.3.1 C2K

Convert Celsius to Kelvin,

$$K = T + 273.15 \tag{2.15}$$

2.3.2 K2C

Convert Kelvin to Celsius,

$$T - 273.15 \tag{2.16}$$

2.4 Ocean

2.4.1 Latent Heat of vaporization

$$L_e = (2.501 - 0.00237 T_{sea})\, 10^6 \tag{2.17}$$

2.5 Interaction

These procedures are related to the interaction between air–sea along or near its interface.

2.5.1 Equivalent Wind

Knowing the wind speed at certain height, considering a logaritmic profile, could be estimated the equivalent wind at a different height by,

$$U_z = \frac{u_*}{\sqrt{\frac{\kappa}{\log\left(\frac{z}{z_0}\right)}}} \tag{2.18}$$

where u_* is the friction velocity (2.5.5), κ the Von Karman constant, z_0 the known wind level and z the height of desired wind.

2.5.2 Roughness Length

From Smith (1988)

$$z_c = \alpha_c \frac{u_*^2}{g}$$

$$z_s = R_r \frac{\nu}{u_*}$$

$$z_0 = z_c + z_s \tag{2.19}$$

which leads to a continuum solution for high and low wind speeds regime.

2.5.3 Reynolds Roughness

$$R_r = \frac{z_0 u_*}{\nu} \tag{2.20}$$

2.5.4 Drag coefficient

$$\sqrt{C_D} = \kappa \left[\ln\left(\frac{z}{z_0}\right) - \Psi_u\right]^{-1} \tag{2.21}$$

2.5.5 Friction velocity estimative

As

$$u_* = \sqrt{C_D} U \tag{2.22}$$

u_* is estimated by an iteration process. The initial value is simple figured out, with which z_0 is estimated as described on section 2.5.2, which permit estimate the C_D value as section 2.5.4, which finally leads to u_* again by equation 2.22. This cycle is redone until the initial and final values converges. This last value is choosed as the best coherent with true measured data.

2.5.6 Wind level correction

The friction velocity is estimated according section 2.5.5, and drag coefficient from section 2.5.4 considering a neutral layer ($\Psi_u = 0$) and $z = 10$ to estimate wind at 10 meters. From equation 2.22,

$$U_{10} = \frac{u_*}{\sqrt{C_D}} \tag{2.23}$$

2.5.7 Noon Solar Altitude

According to Liou (1980),

$$\cos(\theta_0) = \sin(\lambda)\ \sin(\delta) + \cos(\lambda)\ \cos(\delta)\ \cos(h)$$

where θ_0 is the sun zenithal angle, λ is the observer latitude, δ is the sun declination, and h hour angle. For the meridian sun passage $h = 0$, therefore,

$$\cos(\theta_0) = \sin(\lambda)\ \sin(\delta) + \cos(\lambda)\cos(\delta) \tag{2.24}$$

From Liou (1992) cames the relation,

$$\sin(\delta) = \sin(\epsilon)\ \sin(\phi) \tag{2.25}$$

From (2.25),

$$\cos(\delta) = \pm\sqrt{1\,(\sin(\epsilon)\ \sin(\phi))^2} \tag{2.26}$$

Applying (2.25) and (2.26) in (2.24),

$$\begin{aligned} \cos(\theta_0) &= \sin(\lambda)\ \sin(\epsilon)\ \sin(\phi) + \\ &\quad + \cos(\lambda)\left(\pm\sqrt{1\,(\sin(\epsilon)\ \sin(\phi))^2}\right) \end{aligned} \tag{2.27}$$

2.5.8 Wind Stress

$$\tau = \rho_{air} C_D u^2 \tag{2.28}$$

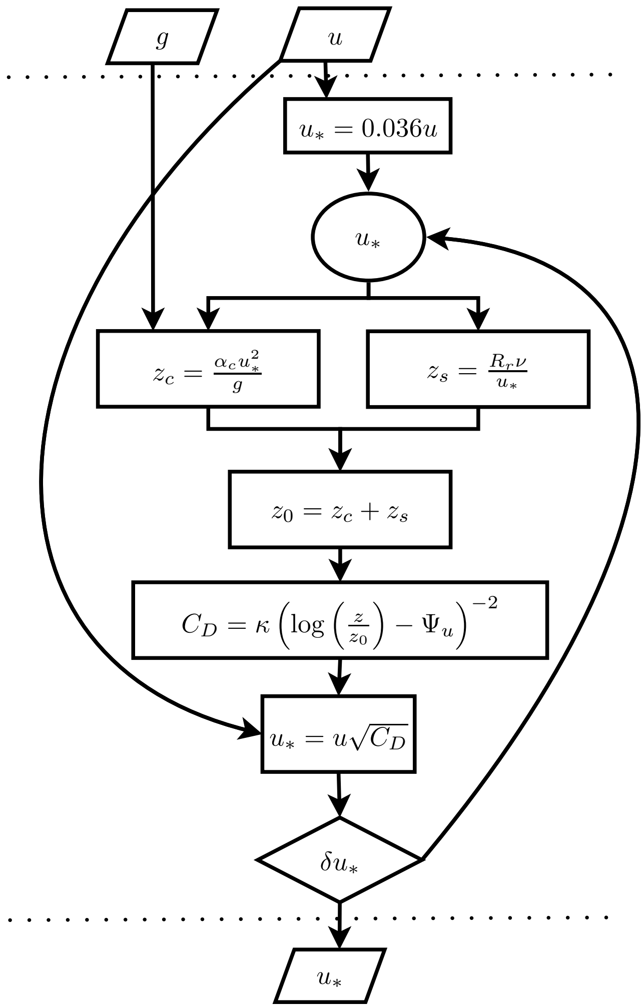

Figure 2.1: Iteration process to estimate friction velocity (u_*).

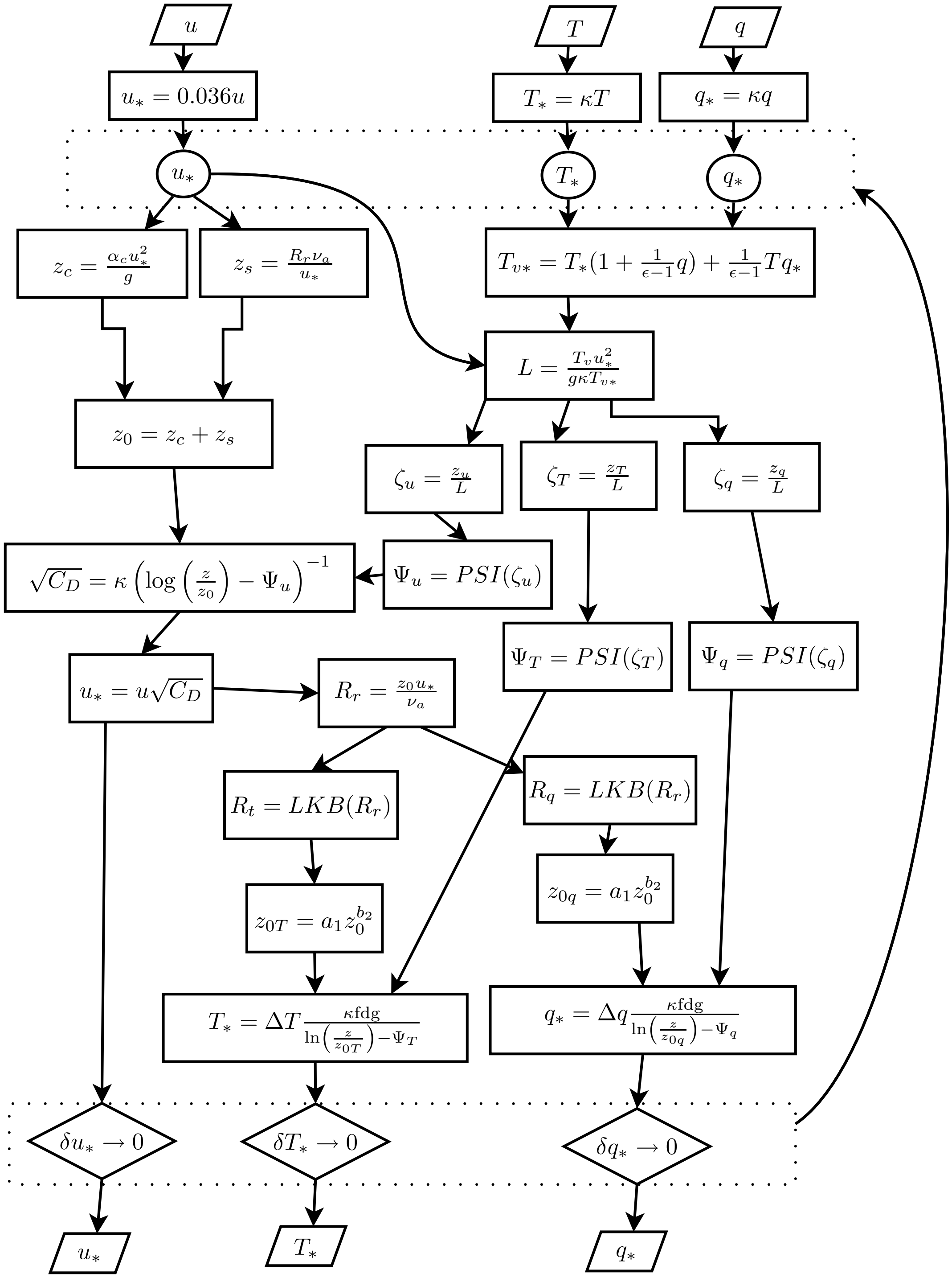
u
$u_* = 0.036u$
T
$T_* = \kappa T$
q
$q_* = \kappa q$
u_*
T_*
q_*
$z_c = \frac{\alpha_c u_*^2}{g}$
$z_s = \frac{R_r \nu_a}{u_*}$
$T_{v*} = T_*(1 + \frac{1}{\epsilon - 1} q) + \frac{1}{\epsilon - 1} T q_*$
$L = \frac{T_v u_*^2}{g \kappa T_{v*}}$
$z_0 = z_c + z_s$
$\zeta_u = \frac{z_u}{L}$
$\zeta_T = \frac{z_T}{L}$
$\zeta_q = \frac{z_q}{L}$
$\sqrt{C_D} = \kappa \left(\log\left(\frac{z}{z_0}\right) - \Psi_u \right)^{-1}$
$\Psi_u = PSI(\zeta_u)$
$\Psi_T = PSI(\zeta_T)$
$\Psi_q = PSI(\zeta_q)$
$u_* = u\sqrt{C_D}$
$R_r = \frac{z_0 u_*}{\nu_a}$
$R_t = LKB(R_r)$
$R_q = LKB(R_r)$
$z_{0q} = a_1 z_0^{b_2}$
$z_{0T} = a_1 z_0^{b_2}$
$T_* = \Delta T \frac{\kappa \mathrm{fdg}}{\ln\left(\frac{z}{z_{0T}}\right) - \Psi_T}$
$q_* = \Delta q \frac{\kappa \mathrm{fdg}}{\ln\left(\frac{z}{z_{0q}}\right) - \Psi_q}$
$\delta u_* \to 0$
$\delta T_* \to 0$
$\delta q_* \to 0$
u_*
T_*
q_*

Figure 2.2: .

2.5.9 Stability Parameter

$\zeta_u, \zeta_T, \zeta_q$

2.6 Heat Flux

2.6.1 Short wave radiation

$$Q_{sw} = (1 - \alpha) Q_{sw_i} \tag{2.29}$$

where Q_{sw_i} is the incident Short wave radiation measured and α the albedo defined according to Payne (1972).

2.6.2 Short wave radiation for clear sky

From Reed (1977), the mean daily insulation ($Q_{cs}[W\ m^{-2}]$) is defined as

$$\begin{aligned} Q_{cs} &= A_0 + A_1 \cos(\phi) + B_1 \sin(\phi) + \\ &\quad + A_2 \cos(2\phi) + B_2 \sin(2\phi) \end{aligned} \tag{2.30}$$

where coefficients are defined by table 2.1, $\phi = (t - 21)(360/365)$, t is day of year and L the latitude.

	Latitudes 20°S to 40°N
A_0	$15.82 + 326.87\cos(L)$
A_1	$9.63 + 192.44\cos(L + 90)$
B_1	$3.27 + 108.70\sin(L)$
A_2	$0.64 + 7.80\sin(2(L - 45))$
B_2	$0.5 + 14.42\cos(2(L - 5))$
	Latitudes 40°N to 60°N
A_0	$342.61 - 1.97L - 0.018L^2$
A_1	$52.085.86L + 0.043L^2$
B_1	$-4.80 + 2.46L - 0.017L^2$
A_2	$1.08 - 0.47L + 0.011L^2$
B_2	$-38.79 + 2.43L - 0.034L^2$

Table 2.1: Coefficients of Smithsonian formula for the latitudes ranges Reed (1977).

2.6.3 Cloud cover

Fraction of cloud cover, varies between 0 to 1. Reed (1977) describes the ratio between observed short wave radiation and expected one for clear sky as

$$\frac{Q_{sw}}{Q_{cs}} = 1 - n_c C + 0.0019\alpha$$

where $n_c = 0.62$, limited to values of C between 0.3 and 1.0. Rewriting this equation in respect of C,

$$C = \left(1 - \frac{Q_{sw}}{Q_{cs}} + 0.0019\alpha\right)\frac{1}{n_c} \tag{2.31}$$

2.6.4 Long wave heat flux

The Long wave heat flux is resolved parameterizing the Radiative Transfer Models (RTMs). Those bulk formulae uses air–sea interface samples as cloudiness, sea surface temperature and near surface air temperature and humidity to estimate the net infrared radiation emitted (Fung et al., 1984). Two methods are available on *Fluid*, called here as Clark (Clark et al., 1974) and Bunker (Bunker, 1976).

- Bunker

$$\begin{aligned} Q_{lw} &= 0.022\left[\epsilon\sigma T_a^4\,(11.7 - 0.23e)\,F_C\right] + \\ &+ 4\epsilon\sigma T_a^3(T_s - T_a) \end{aligned} \tag{2.32}$$

  where ϵ (=0.98 as suggested by WGASF (2000)) is the spectrally integrated emittance, e the vapor pressure near surface, σ ($= 5.67\times10^{-8}\mathrm{Wm}^{-2}\mathrm{K}^{-4}$) the Stefan-Boltzmann constant, T_a and T_s are the air and sea temperature in Kelvins. F_C is the Cloudness given by table 2.2.

- Clark

$$\begin{aligned} Q_{lw} &= \epsilon\sigma T_s^4\left(0.39 - 0.05e^{0.5}\right)F_C + \\ &+ 4\epsilon\sigma T_s^3(T_s - T_a) \end{aligned} \tag{2.33}$$

  where the variables are the same defined for equation 2.32.

2.6.5 Transfer Coeficients

2.6.6 Richardson Number

2.6.7 Turbulent Heat fluxes

2.7 Geophysical Fluid Dynamics

2.7.1 Coriolis Parameter f

$$f = 2\Omega\sin(\theta)$$

Latitude	Clark	Bunker
80°		0.84
70°		0.80
60°		0.76
50°	0.73	0.72
40°	0.69	0.68
30°	0.64	0.63
20°	0.60	0.59
10°	0.56	0.52
5°	0.53	0.50
Equator	0.51	
$F(C) =$	$1 - aC^2$	$1 - aC$

Table 2.2: Cloud Factor in function of latitude module for different methods. C is defined by function 2.6.3

2.7.2 Coriolis Parameter h

$$h = 2\Omega \cos(\theta)$$

2.7.3 Coriolis Parameter β

$$\beta = \frac{2\Omega \cos(\theta)}{R}$$

2.7.4 Earth's Angular Velocity

$$\Omega = \frac{2\pi}{\left(1 + \frac{1}{365}\right) \cdot 24 \cdot 60 \cdot 60} = 86637 \quad \left[\frac{\text{Rad}}{\text{s}}\right]$$

2.7.5 Gravity

The gravity is defined in function of latitude by

$$g = \gamma * \left[1.0 + C_1 \sin^2 \phi + C_2 \sin^4 \phi + C_3 \sin^6 \phi + C_4 \sin^8 \phi\right] \tag{2.34}$$

where $\gamma = 9.7803267715$, $C_1 = 0.0052790414$, $C_2 = 0.0000232718$, $C_3 = 0.0000001262$, $C_4 = 0.0000000007$ and ϕ is the latitude in radians.

Based on Fairall et al. (1996).

Chapter 3

Future goals

Goals for future Fluid versions:

0.2 Translate and include seawater package routines.

0.3 Include more available methods to estimate wind profile, as Large and Pond one; Include more methods to long wave radiation heat balance over ocean surface.

0.4 Include stability layer factor Ψ calculus on iteration process; Include turbulent heat fluxes, as Sensible and Latent heat fluxes by bulk formulae by iteration method to estimate needed parameters;

0.5 Translate package to be based on classes, not a group of functions. Biggest advantage is optimization of calculus and share common variables between procedures. The dependences could be easily checked.

Chapter 4

On use

4.1 Example

```
import numarray
import fluid.common.common
import fluid.common.gravity
import fluid.atmosphere.atmospheric_functions
import fluid.interaction.heat_flux
import fluid.interaction.others


Lat = numarray.array([0,45,59])

Ks = numarray.array([278.,292.,305.])
Ts = Ks-273.15
Ka = numarray.array([280.,290.,300.])
Ta = Ka-273.15
RH = numarray.array([.8,.7,.8])

u = numarray.array([6.,10.,13.])
z_u = numarray.array([4.])
z_T = numarray.array([3.])
z_q = numarray.array([3.])

p = 100800.

e_sat=fluid.atmosphere.atmospheric_functions.saturation_vapor_pressure(Ta)
w_sat = fluid.atmosphere.atmospheric_functions.mixing_ratio(e_sat,p)
q_sat = fluid.atmosphere.atmospheric_functions.specific_humidity(w_sat)
ea = fluid.atmosphere.atmospheric_functions.RH2e(RH,w_sat,p)
w = fluid.atmosphere.atmospheric_functions.mixing_ratio(ea,p)
```

```
q = fluid.atmosphere.atmospheric_functions.specific_humidity(w)
air_nu = fluid.atmosphere.atmospheric_functions.air_viscosity(Ta)
g = fluid.common.gravity.gravity(Lat)

w_sat_sea = fluid.atmosphere.atmospheric_functions.saturation_mixing_ratio(Ts,p)
q_sat_sea = fluid.atmosphere.atmospheric_functions.specific_humidity(w_sat_sea)
q_sea = q_sat_sea*.98

Dq = q-q_sea
DT = Ta - Ts

Kv=fluid.atmosphere.atmospheric_functions.virtual_temperature(Ka,ea,p=p)
Tv = fluid.common.common.K2C(Kv)

# Maybe first u_star gest should be inside find_transfer_coefficients
u_star = fluid.interaction.others.find_u_star(u, air_nu, z_u, g=g)
#print ‘‘u_star’’,u_star
#u_star = 0.036*u
z0=fluid.interaction.others.set_z0(u_star, air_nu)
#CD=fluid.interaction.others.drag_coefficient(z_u,z0)

u_star, T_star, q_star = fluid.interaction.others.find_transfer_coefficients(Ta,
Tv,Ts,q,q_sea,u,z_u,z_T,z_q,air_nu,u_star)


# compute latent heat
Le = fluid.interaction.others

# compute fluxes into ocean
air_rho = fluid.atmosphere.atmospheric_functions.air_density(p,Kv)

Hs, Hl = fluid.interaction.others.turbulent_heat_fluxes(air_rho,u_star,T_star,q_
star,Le)

#% compute transfer coefficients at measurement heights
CD=(u_star/u)**2
CT=u_star*T_star/(u*(DT)) # Stanton number
CQ=u_star*q_star/(u*(Dq))  # Dalton number

stress = fluid.interaction.others.wind_stress(air_rho,CD,u)
#
RI = fluid.interaction.others.richardson_number(Ta,DT,Tv,Dq,u,z_u,g)
```

4.2 Application cases

4.2.1 Wind level correction

The Pilot Research Moored Array in the Tropical Atlantic (PIRATA) is now composed by teen moorings on Tropical Atlantic. Between others sensors the buoys sample the magnitude and direction of the wind at 4m each 10 minutes.

Different wind datasets are normally compared on the standard height of 10m. This is due the expected friction at bottom boundary and consequently a vertical velocity shear. The figure 4.1 exemplify the difference between the actually measured wind magnitude and the estimated for $10m$ according 2.5.6 on PIRATA daily mean data.

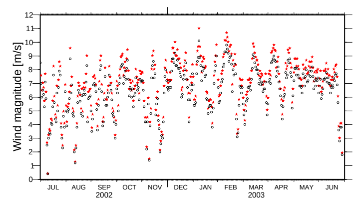

Figure 4.1: Black circles are wind magnitude measured at 4 m height by PIRATA buoy at 15°N 38°W, and red stars the estimated wind for 10 m height.

Considering that the estimative of $10m$ wind presented here is a good approximation of the true wind at that height, the bias is on average near 8.5% and not constant (figure 4.2). Therefore, in procedures where the wind at the standart height of $10m$ is considered, ignore the height correction could lead to different results.

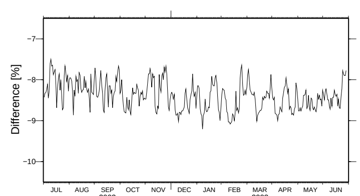

Figure 4.2: Percentual difference between between measured magnitude at 4m and estimated one at 10m presented on figure 4.1.

4.2.2 Short wave

As described on section 2.6.1, the the short wave radiation that is effectively cross the sea surface could be estimated from the incident short wave radiation measured by TOGA/TAO and PIRATA equipments. The figure 4.3 illustrate that result for an annual cycle between 2002 and 2003.

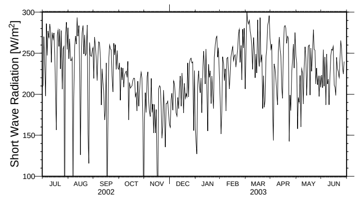

Figure 4.3: Short wave heat flux estimated from data collected by PIRATA mooring at 15°N 38°W.

4.2.3 Long wave

Those moorings measure too the relative humidity (RH), the temperature of air near surface (T_a) and at 1 meter depth (T_s). From section 2.2.7 and 2.2.10 the vapor pressure (e) could be estimated. With the latitude and day of year the short wave radiation for clear sky could be estimated according section 2.6.2, and therefore the cloud cover index from section 2.6.3. Now, following section 2.6.4, with direct measured data and estimated ones the long wave heat balance could be estimated.

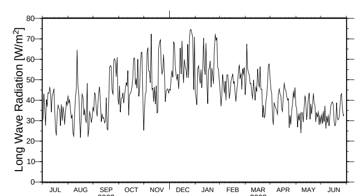

Figure 4.4: Long wave heat flux estimated from data collected by PIRATA mooring at 15°N 38°W.

Bibliography

Bunker, A. F., 1976:. Computations of surface energy flux and annual air-sea interaction cycles of the North Atlantic Ocean. *Mon. Wea. Rev.*, **104**, 1122–1140.

Clark, N. E., L. Eber, R. M. Laurs, J. A. Renner, and J. F. T. Saur, 1974:. Heat exchange between ocean ant atmospheric in the eastern North Pacific for 1961-71. NMFS SSRF 682, NOAA Tech. Rep.

Fairall, C. W., E. F. Brandley, D. P. Rogers, J. B. Edson, and G. S. Young, 1996:. Bulk parametrization of air-sea fluxes for tropical ocean-global atmosphere coupled-ocean atmosphere response experiment. *J. Geophys. Res.*, **101(C2)**, 3747–3764.

Fung, I. Y., D. E. Harrison, and A. A. Lacis, 1984:. On the variability of the net longwave radiation at the ocean surface. *Rev. Geophys.*, **22**, 177–193.

Liou, K.-K., 1980:. *An introduction to atmospheric radiation*, Volume 26 of *International geophysical series*. Academic press, inc, 392 pp.

Liou, K.-K., 1992:. *Radiation and Cloud Process in the Atmosphere*, Volume 20 of *Oxford monographs on geology and geophysics*. Oxford university press, 487 pp.

Pawlowicz, R., B. Beardsley, S. Lentz, E. Dever, and A. Anis, 2001:. Software simplifies air–sea data estimates. *EOS, Transactions, American Geophysical Union*, **82**, 2.

Payne, R. E., 1972:. Albedo of the sea surface. *J. Atmos. Sci.*, **29**, 959–970.

Reed, R. K., 1977:. On estimating insolation over the ocean. *J. Phys. Oceanogr.*, **7**, 482–485.

Smith, S. D., 1988:. Coefficients for 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.

Wallace, J. M. and P. V. Hobbs, 1977:. *Atmospheric science :an introductory survey*. International geophisical series. Academic press, 467 pp.

WGASF, 2000, November). Intercomparison and validation of ocean-atmosphere energy flux fields. Technical report, World Meteorological Organization.