ATM 2106 TA Class

April 10, 2019

Department of Atmospheric Sciences, Yonsei University
Air-Sea Modeling Laboratory

Last time

- Stability
- Geopotential height
- Equation of fluid motion

Today

- The equations of fluid motion with rotation
- Geostrophic motion
- Practice of python HW#2

- **The equation of motion**
  - Net force = sum of all forces

$$\rho \;\delta x\; \delta y\; \delta z\; \frac{D\mathbf{u}}{Dt} = \mathbf{F}_{gravity} + \mathbf{F}_{pressure} + \mathbf{F}_{fric}$$

$$\frac{D\mathbf{u}}{Dt} = -g\hat{\mathbf{z}} - \frac{1}{\rho}\nabla p + \mathscr{F}$$

Equations of motion for a non-rotating fluid

임의의 함수 $\phi(x, y, z, t)$ 에 대해서

$$\delta\phi = \frac{\partial\phi}{\partial t}\delta t + \frac{\partial\phi}{\partial x}\delta x + \frac{\partial\phi}{\partial y}\delta y + \frac{\partial\phi}{\partial z}\delta z$$

$$\frac{d\phi}{dt} = \lim_{\delta t \to 0}\frac{\delta\phi}{\delta t} = \boxed{\frac{\partial\phi}{\partial t}} + u\frac{\partial\phi}{\partial x} + v\frac{\partial\phi}{\partial y} + w\frac{\partial\phi}{\partial z}$$

↳ x, y, z가 고정

$$\frac{d\phi}{dt} = \frac{\partial\phi}{\partial t} + \vec{U}\,\nabla\cdot\phi$$

Lagrangian
이동점에서의 총 변화율

Eulerian
고정점에서의 총 변화율

Advection
이류 · 장의 변화

Equations of motion for a non-rotating fluid

- 유체에서 사용하는 6개의 변수

$u \quad v \quad w \quad \rho \quad T \quad P$ ⇒ 총 6개의 식이 필요

유체를 예측하기 위해서는

$\frac{du}{dt}, \frac{dv}{dt}, \frac{dw}{dt}, \frac{d\rho}{dt}, \frac{dT}{dt}, \frac{dP}{dt}$ 를 알아야 한다.

이중 $P=\rho RT$ 이상기체 방정식을 사용하여 5개의 변수로 줄여준다.

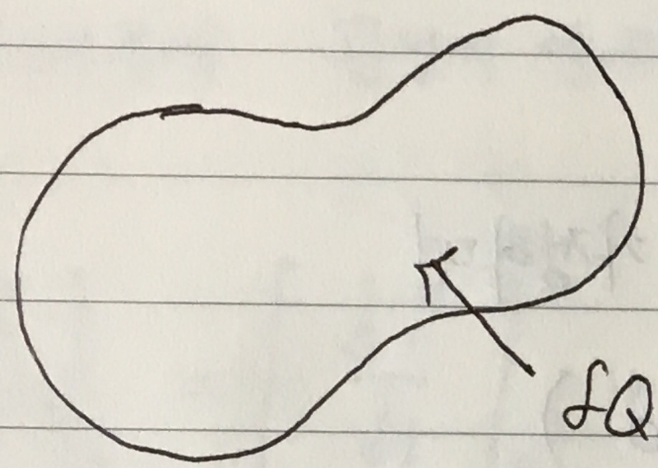
Equations of motion for a non-rotating fluid

$$\frac{D\mathbf{u}}{Dt} + \frac{1}{\rho}\nabla p + g\hat{\mathbf{z}} = \mathcal{F}$$

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}(\rho u) + \frac{\partial}{\partial y}(\rho v) + \frac{\partial}{\partial z}(\rho w) = 0$$

$$\frac{DQ}{Dt} = c_p \frac{DT}{Dt} - \frac{1}{\rho}\frac{Dp}{Dt}$$

$\boxed{3}$ $\frac{dT}{dt}$ 열역학 방정식 (Thermodynamic eq.)



외부에서 임의의 공기괴에 δQ 만큼 열을 주면 공기괴는 내부에 열을 에너지로 저장하거나, 공기를 팽창 시키는 (외부에 일을 하는) 일을 한다.

$$\delta Q = \boxed{C_v\, dT} + \boxed{p\, \delta\alpha}$$

① 내부에 저장 (온도에 비례하는 항)

② 공기를 팽창 (외부에 일든 항)

$$\frac{\delta Q}{\delta t} = C_v \frac{dT}{\delta t} + P \frac{\delta\alpha}{\delta t} \Rightarrow \dot{Q} = \frac{dQ}{dt} = C_v \frac{dT}{dt} + P\frac{d\alpha}{dt}$$

* 이상기체 방정식 미분

$P = \rho R T \qquad P\alpha = RT \qquad \Rightarrow \qquad P\,d\alpha + \alpha\,dP = R\,dT$

$$\frac{dQ}{dt} = C_v \frac{dT}{dt} + P\frac{d\alpha}{dt}$$

$$= C_v \frac{dT}{dt} + R\frac{dT}{dt} - \alpha\frac{dP}{dt}$$

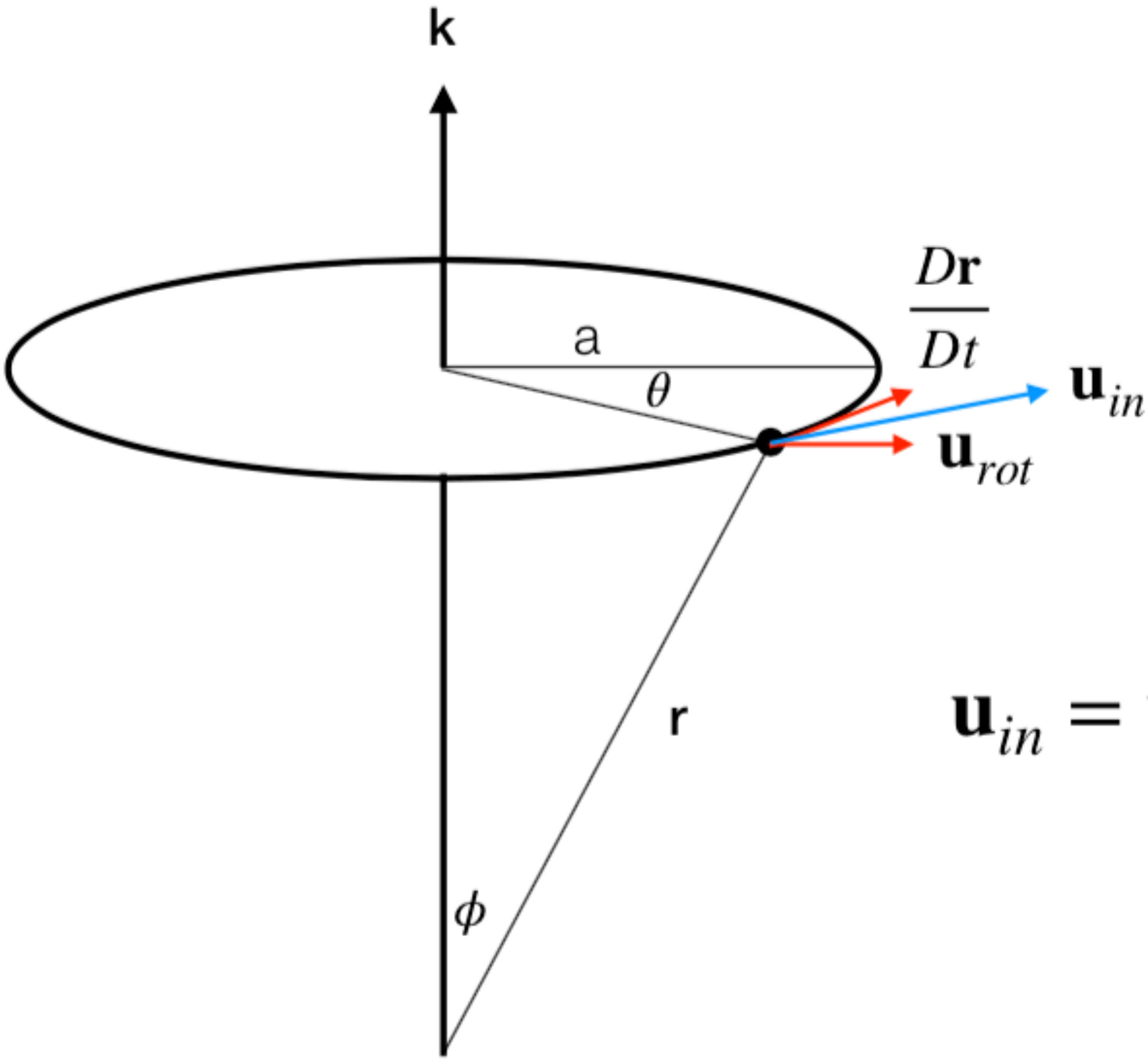
정적비열

$$= C_p \frac{dT}{dt} - \alpha\frac{dP}{dt}$$

정압비열

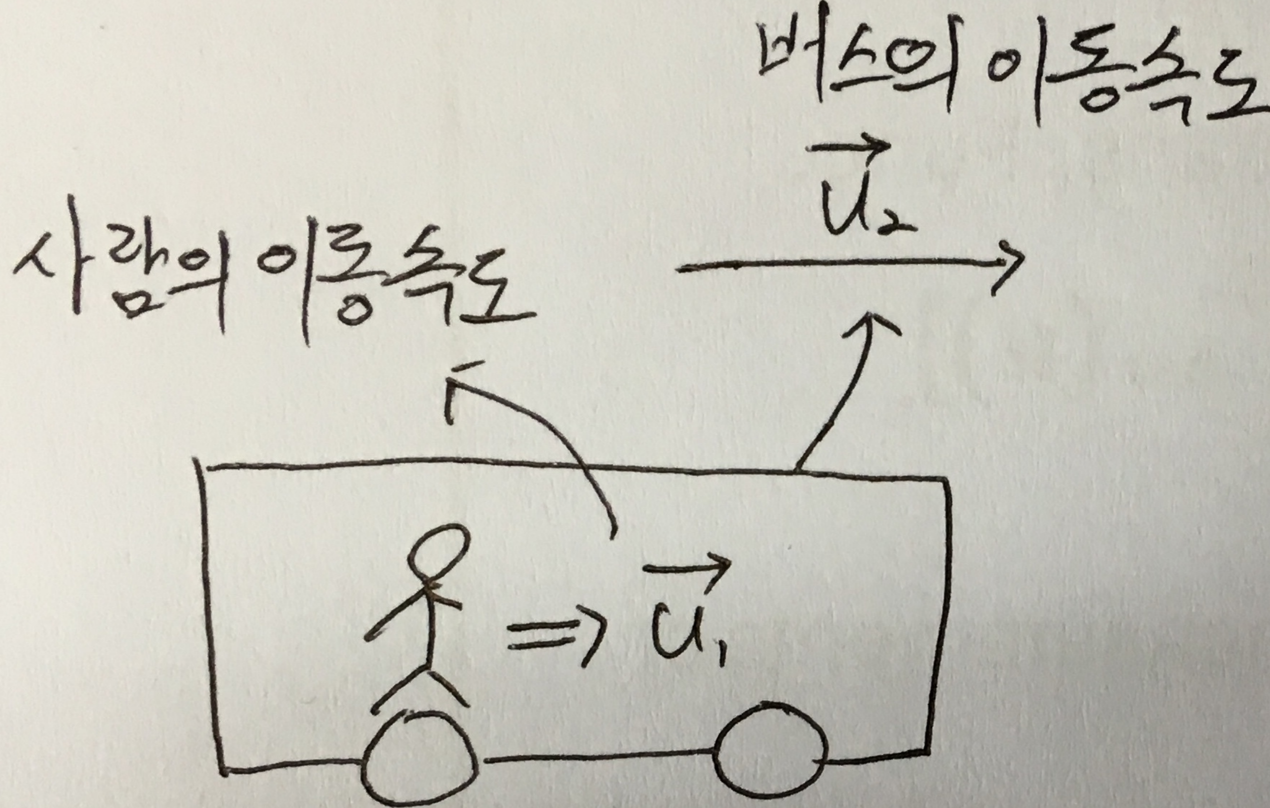
$C_p = C_v + R$

Equations of motion for a rotating fluid



$$\mathbf{u}_{in} = \mathbf{u}_{rot} + \boldsymbol{\Omega} \times \mathbf{r}$$

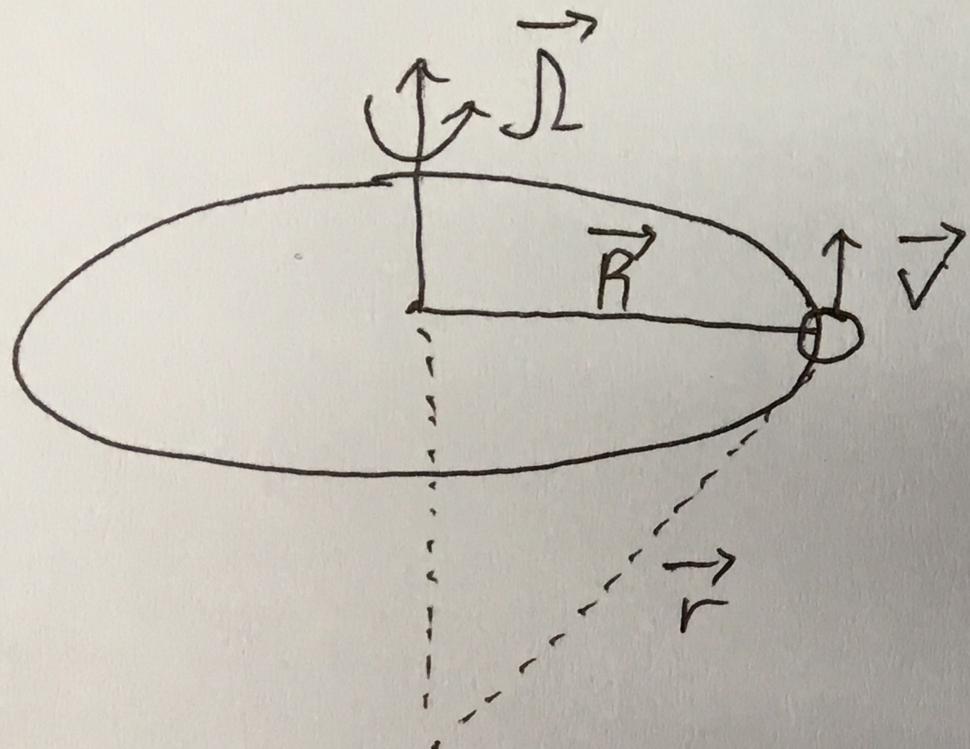
Equations of motion for a rotating fluid



외부에서 바라볼 때 사람의 이동속도($\vec{u}$)는

$$\vec{u} = \vec{u}_1 + \vec{u}_2$$

Equations of motion for a rotating fluid



선속도 $\vec{V} = \vec{\Omega} \times \vec{R}$

$= \vec{\Omega} \times \vec{r}$

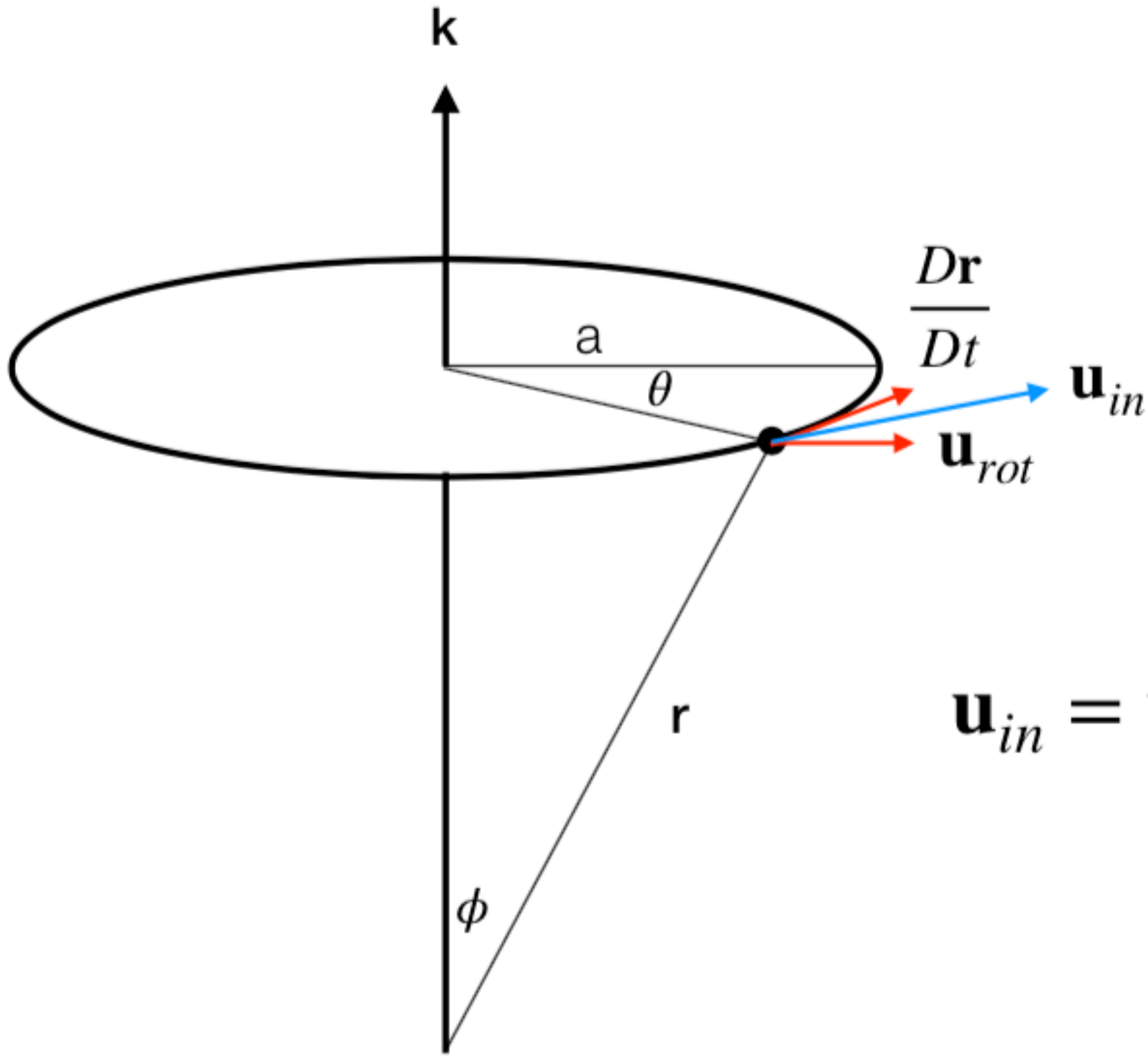
원래는 $\vec{\Omega} \times \vec{R}$ 이지만
$\vec{\Omega}$에 수직되는 항만 계산되므로
$\vec{\Omega} \times \vec{r}$로 써도 결과값은 같다.

$$\vec{U}_I = \vec{U}_A + \vec{\Omega} \times \vec{R}$$

$\vec{U}_I$: 외부에서 바라볼때

$\vec{U}_A$: 회전하는 계 안에서 바라볼때

Equations of motion for a rotating fluid



$$\mathbf{u}_{in} = \mathbf{u}_{rot} + \boldsymbol{\Omega} \times \mathbf{r}$$

A vector, $\mathbf{A}$, can be written with any three independent unit vectors.

$$\mathbf{A} = A_i\ \hat{\mathbf{i}} + A_j\ \hat{\mathbf{j}} + A_k\ \hat{\mathbf{k}} \longrightarrow \text{in the absolute frame}$$

$$= A_x\ \hat{\mathbf{x}} + A_y\ \hat{\mathbf{y}} + A_z\ \hat{\mathbf{z}} \longrightarrow \text{in the rotating frame}$$

Then, let's find out the Lagrangian differentiation of $\mathbf{A}$ with respect to the absolute (inertial) frame

$$\left(\frac{D\mathbf{A}}{Dt}\right)_{in} = \frac{D}{Dt}\left(A_i\ \hat{\mathbf{i}}\right) + \frac{D}{Dt}\left(A_j\ \hat{\mathbf{j}}\right) + \frac{D}{Dt}\left(A_k\ \hat{\mathbf{k}}\right)$$

$$= \hat{\mathbf{i}}\frac{DA_i}{Dt} + \hat{\mathbf{j}}\frac{DA_j}{Dt} + \hat{\mathbf{k}}\frac{DA_k}{Dt}$$

Equations of motion for a rotating fluid

$\left(\frac{D\mathbf{A}}{Dt}\right)_{in}$ can also be written as

$$\frac{D\mathbf{r}}{Dt} = \mathbf{\Omega} \times \mathbf{r}$$

$$\left(\frac{D\mathbf{A}}{Dt}\right)_{in} = \frac{D}{Dt}\left(A_x\, \hat{\mathbf{x}}\right) + \frac{D}{Dt}\left(A_y\, \hat{\mathbf{y}}\right) + \frac{D}{Dt}\left(A_z\, \hat{\mathbf{z}}\right)$$

$$= \hat{\mathbf{x}}\frac{DA_x}{Dt} + \hat{\mathbf{y}}\frac{DA_y}{Dt} + \hat{\mathbf{z}}\frac{DA_z}{Dt} + A_x\frac{D\hat{\mathbf{x}}}{Dt} + A_y\frac{D\hat{\mathbf{y}}}{Dt} + A_z\frac{D\hat{\mathbf{z}}}{Dt}$$

$$= \hat{\mathbf{x}}\frac{DA_x}{Dt} + \hat{\mathbf{y}}\frac{DA_y}{Dt} + \hat{\mathbf{z}}\frac{DA_z}{Dt} + \mathbf{\Omega} \times \left(A_x\hat{\mathbf{x}} + A_y\hat{\mathbf{y}} + A_z\hat{\mathbf{z}}\right)$$

$$= \left(\frac{D\mathbf{A}}{Dt}\right)_{rot} + \mathbf{\Omega} \times \mathbf{A}$$

Equations of motion for a rotating fluid

$$\delta i = \frac{\partial i}{\partial \lambda}\delta\lambda + \frac{\partial i}{\partial \phi}\delta\phi + \frac{\partial i}{\partial z}\delta z$$

강체회전의 경우 $\delta\lambda = \Omega\delta t$, $\delta\phi = 0$, $\delta z = 0$이기 때문에 $\delta i/\delta t = (\partial i/\partial\lambda)(\partial\lambda/\delta t)$로 쓸 수 있으며, $\delta t \to 0$의 극한을 취하면 다음과 같이 된다.

$$\frac{D_a i}{Dt} = \Omega\frac{\partial i}{\partial \lambda}$$

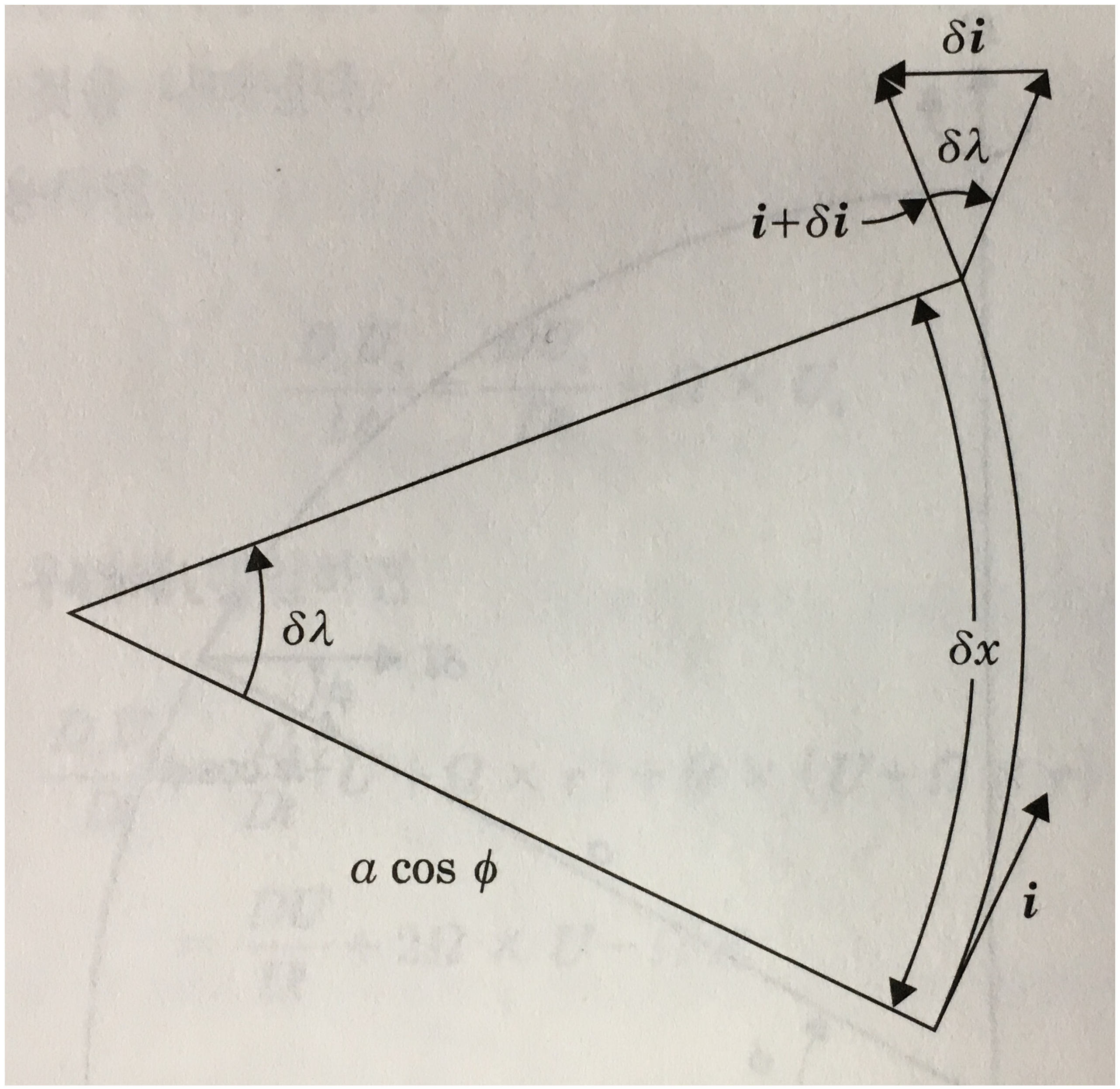
한편, 그림 2.1과 그림 2.2로부터, i의 동서미분은 아래와 같이 표현된다.

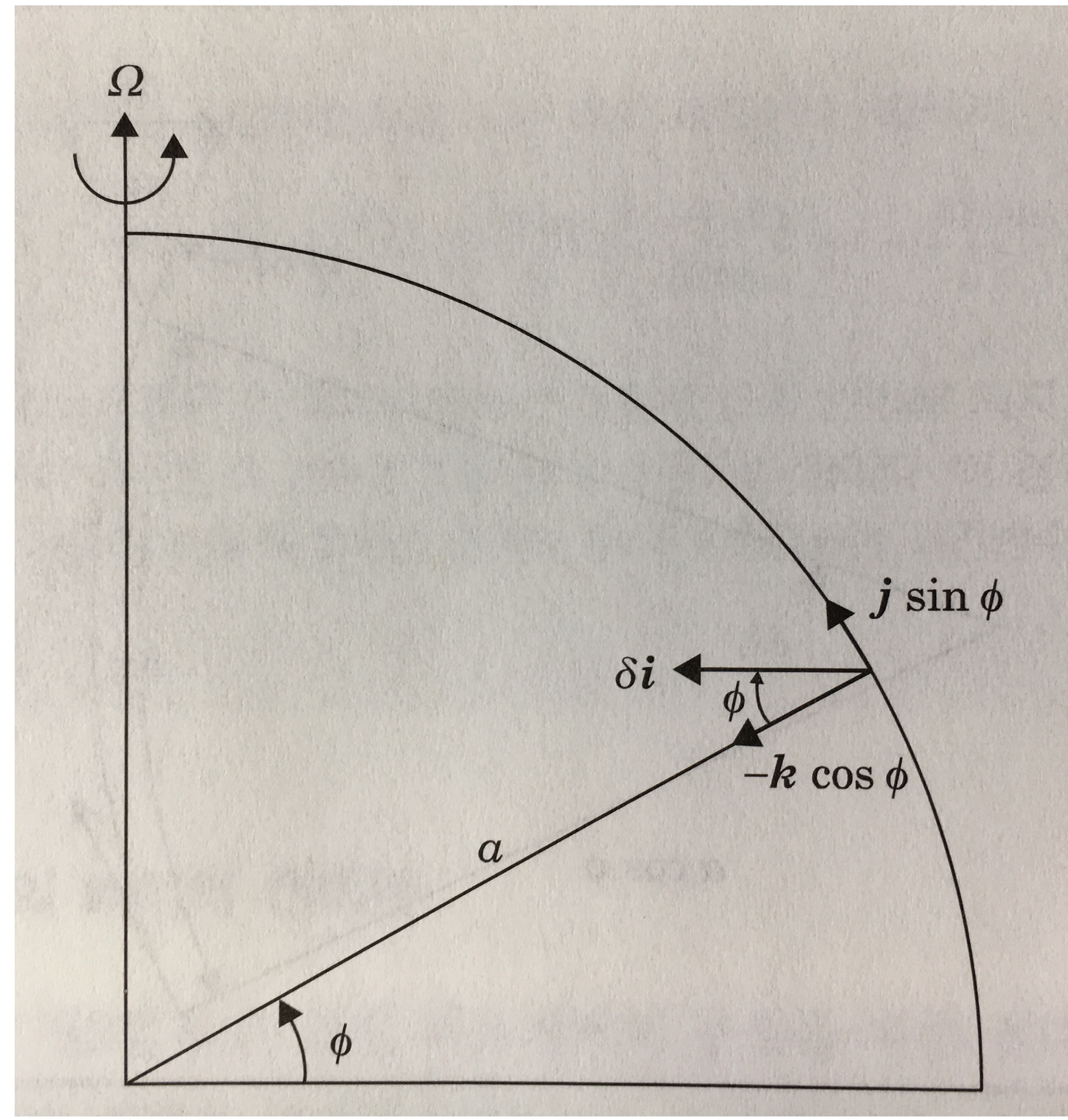
$$\frac{\partial i}{\partial \lambda} = \boldsymbol{j}\sin\phi - \boldsymbol{k}\cos\phi$$

그러나 $\boldsymbol{\Omega} = (0,\ \Omega\cos\phi,\ \Omega\sin\phi)$이므로

$$\frac{D_a i}{Dt} = \Omega(\boldsymbol{j}\sin\phi - \boldsymbol{k}\cos\phi) = \boldsymbol{\Omega} \times \boldsymbol{i}$$

Equations of motion for a rotating fluid





Equations of motion for a rotating fluid

$$A_x \frac{D\hat{x}}{Dt} = \vec{\Omega} \times A_x \hat{x}$$

(holton 책에선 $\hat{x} \to \hat{i}$ 로 표현)

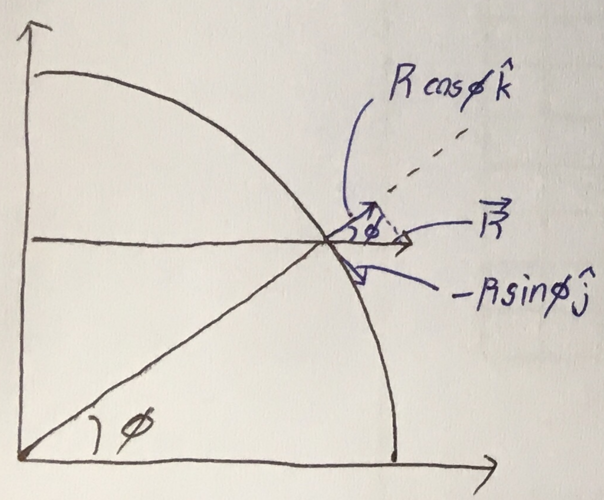
$$\delta i = \frac{\partial i}{\partial \lambda}\delta\lambda + \frac{\partial i}{\partial \phi}\delta\phi + \frac{\partial i}{\partial z}\delta z$$

spherical coordinate 에서의 $\hat{i}$의 변화율.

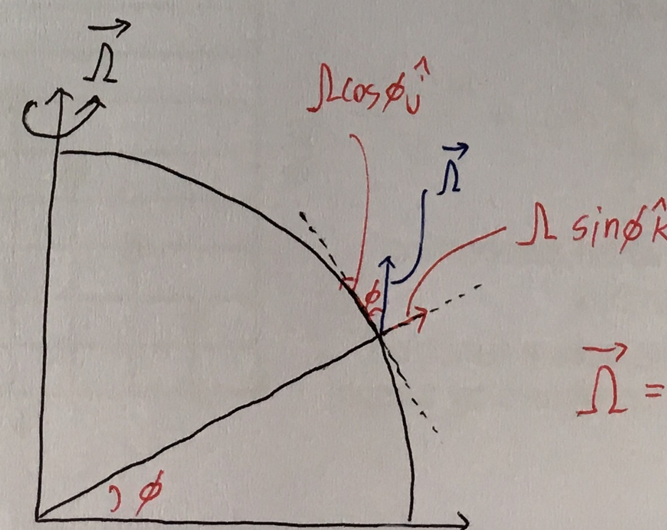
아래 그림을 보면

δi 에 의한 $\delta\phi, \delta z = 0$ 이므로

$$\frac{D\hat{i}}{Dt} = \frac{\partial i}{\partial\lambda}\frac{\partial\lambda}{\partial t} = \Omega\frac{\partial i}{\partial\lambda} = \Omega\frac{|\hat{i}|\,\delta\lambda\,(-\vec{R}/R)}{\delta\lambda} = \Omega(\sin\phi\,\hat{j} - \cos\phi\,\hat{k}) = \vec{\Omega}\times\hat{i}$$



$\vec{R} = (0, -R\sin\phi, R\cos\phi)$



$\vec{\Omega} = (0, \Omega\cos\phi, \Omega\sin\phi)$

정리해보면

$$\left[\frac{d\vec{A}}{dt}\right]_I = \left[\frac{d\vec{A}}{dt}\right]_R + \vec{\Omega}\times\vec{A} \qquad (1)$$

절대계에서의 변화율

관성계에서의 변화율

지구(관성계)가 회전하여 발생하는 변화율

$$\vec{U}_I = \vec{U}_R + \vec{\Omega}\times\vec{r} \qquad (2)$$

(1) 식의 $\vec{A}$에 (2) 식 $\vec{U}_I$를 대입하여 풀어주면

Equations of motion for a rotating fluid

$$\left[\frac{d\vec{U}_I}{dt}\right]_I = \left[\frac{d\vec{U}_I}{dt}\right]_R + \vec{\Omega} \times \vec{U}_I$$

$$= \left[\frac{d}{dt}\left(\vec{U}_R + \vec{\Omega} \times \vec{r}\right)\right]_R + \vec{\Omega} \times \left(\vec{U}_R + \vec{\Omega} \times \vec{r}\right)$$

$$= \left[\frac{d\vec{U}_R}{dt}\right]_R + \left[\underset{0}{\cancel{\frac{d\vec{\Omega}}{dt}}} \times \vec{r}\right]_R + \left[\vec{\Omega} \times \frac{d\vec{r}}{dt}\right]_R + \vec{\Omega} \times \vec{U}_R + \vec{\Omega} \times \vec{\Omega} \times \vec{r}$$

$$\therefore \left[\frac{d\vec{U}_I}{dt}\right]_I = \left[\frac{d\vec{U}_R}{dt}\right]_R + 2\vec{\Omega} \times \vec{U}_R + \vec{\Omega} \times \vec{\Omega} \times \vec{r}$$

Equations of motion for a rotating fluid

$$\left(\frac{D\mathbf{u}_{in}}{Dt}\right)_{in} = \left(\frac{D\mathbf{u}_{in}}{Dt}\right)_{rot} + \boldsymbol{\Omega} \times \mathbf{u}_{in} \qquad \mathbf{u}_{in} = \mathbf{u}_{rot} + \boldsymbol{\Omega} \times \mathbf{r}$$

$$= \left(\frac{D\left(\mathbf{u}_{rot} + \boldsymbol{\Omega} \times \mathbf{r}\right)}{Dt}\right)_{rot} + \boldsymbol{\Omega} \times \left(\mathbf{u}_{rot} + \boldsymbol{\Omega} \times \mathbf{r}\right)$$

$$= \left(\frac{D\mathbf{u}_{rot}}{Dt}\right)_{rot} + 2\boldsymbol{\Omega} \times \mathbf{u}_{rot} + \boldsymbol{\Omega} \times \left(\boldsymbol{\Omega} \times \mathbf{r}\right)$$

Equations of motion for a rotating fluid

Momentum equation we did last time can be applied to the parcel in the absolute frame

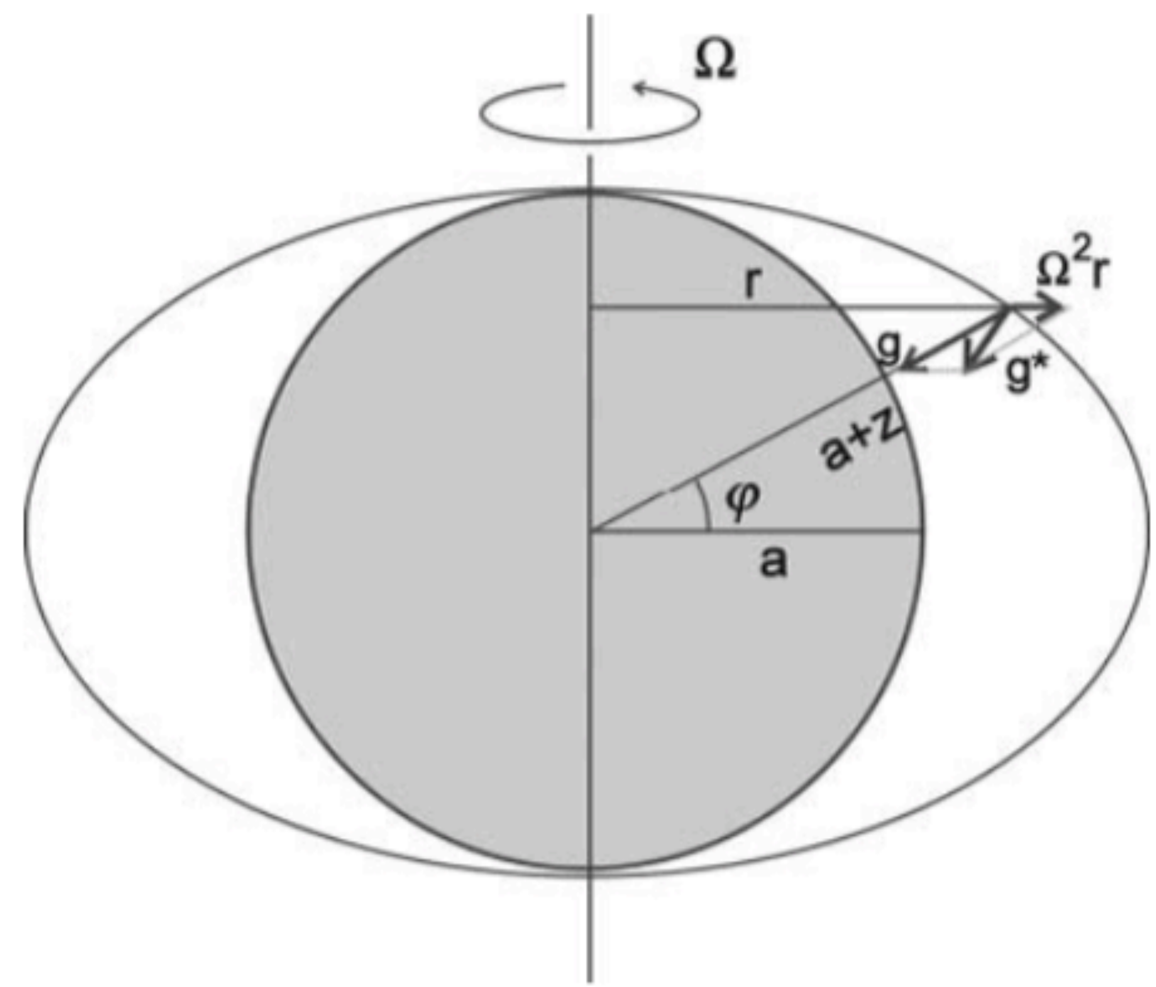
$$\left(\frac{D\mathbf{u}_{in}}{Dt}\right)_{in} + \frac{1}{\rho}\nabla p + g\hat{\mathbf{z}} = \mathscr{F}$$

$$\downarrow$$

$$\left(\frac{D\mathbf{u}_{rot}}{Dt}\right)_{rot} + \frac{1}{\rho}\nabla p + g\hat{\mathbf{z}} = \underbrace{-\,2\boldsymbol{\Omega}\times\mathbf{u}_{rot}}_{\text{Coriolis acceleration}}\;\underbrace{-\,\boldsymbol{\Omega}\times\left(\boldsymbol{\Omega}\times\mathbf{r}\right)}_{\text{Centrifugal acceleration}} + \mathscr{F}$$

Equations of motion for a rotating fluid

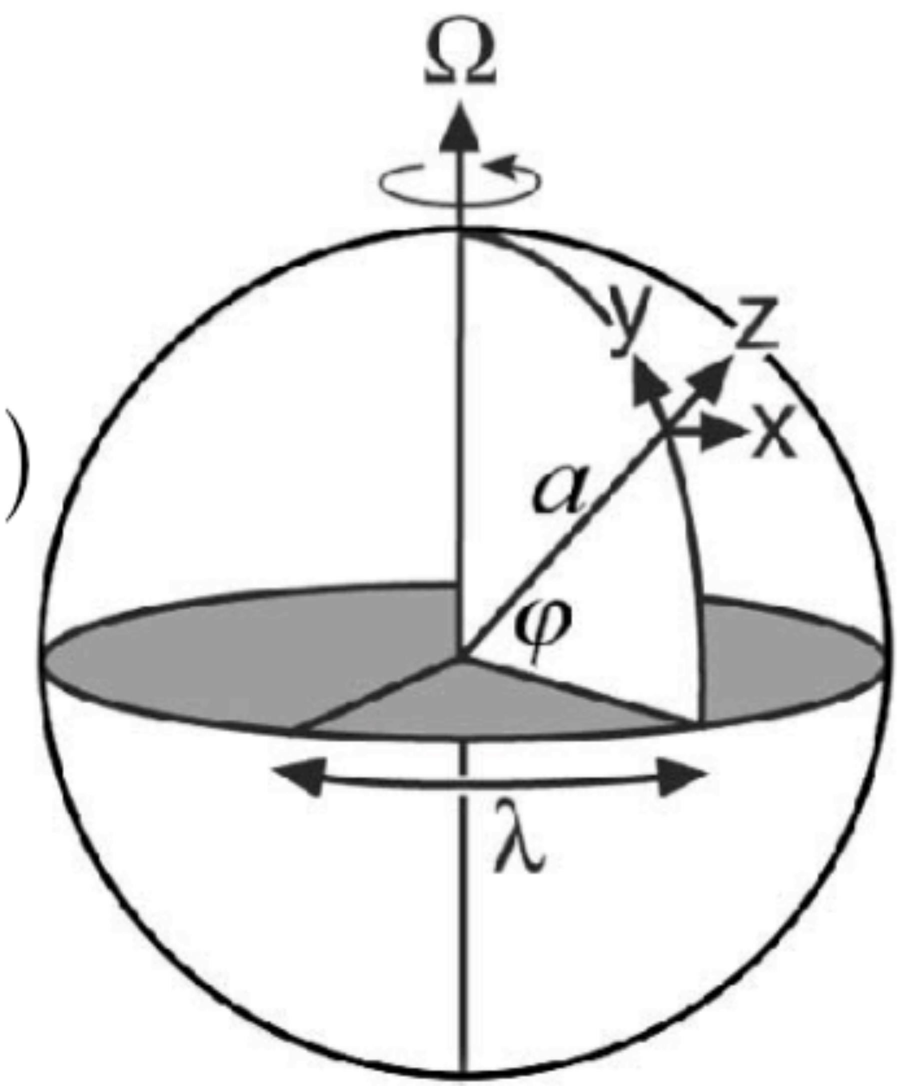
- Centrifugal acceleration: $-\mathbf{\Omega} \times (\mathbf{\Omega} \times \mathbf{r})$
  - Modifies gravity acceleration
  - Effective gravity : g^*
  - $\mathbf{\Omega} \sim 7.27 \times 10^{-5}\ s^{-1}$ makes centrifugal acceleration small.
  - Geoid : The surface perpendicular to the effective gravity
  - We may use this surface to make centrifugal acceleration to disappear.



- Coriolis force: $-2\ \mathbf{\Omega} \times \mathbf{u}$
  - $\mathbf{\Omega} = \left(0,\ \Omega \cos\phi,\ \Omega \sin\phi\right)$
  - $\mathbf{\Omega} \times \mathbf{u} = \left(\Omega \cos\phi\ w - \Omega \sin\phi\ v,\ \Omega \sin\phi\ u,\ -\Omega \cos\phi\ u\right)$
  - w is smaller than other terms.
  - Ωu is smaller than g.
  - $-2\mathbf{\Omega} \times \mathbf{u} \approx -\left(-2\Omega \sin\phi\ v,\ 2\Omega \sin\phi\ u,\ 0\right)$

$$= f\hat{\mathbf{z}} \times \mathbf{u}$$

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



https://youtu.be/RrWKSOvqV-0

Equations of motion for a rotating fluid

$$\frac{D\mathbf{u}}{Dt} + \frac{1}{\rho}\nabla p + g^*\hat{\mathbf{z}} + f\hat{\mathbf{z}} \times \mathbf{u} = \mathscr{F}$$

$$\frac{Du}{Dt} + \frac{1}{\rho}\frac{\partial p}{\partial x} - fv = \mathscr{F}_x$$

$$\frac{Dv}{Dt} + \frac{1}{\rho}\frac{\partial p}{\partial y} + fu = \mathscr{F}_y$$

$$\frac{1}{\rho}\frac{\partial p}{\partial z} + g = 0$$

- Hydrostatic approximation
- Vertical component of the frictional force is negligible compared with gravity

$$\frac{Du}{Dt} + \frac{1}{\rho}\frac{\partial p}{\partial x} - fv = \mathscr{F}_x$$

$$\frac{Dv}{Dt} + \frac{1}{\rho}\frac{\partial p}{\partial y} + fu = \mathscr{F}_y$$

$$\frac{1}{\rho}\frac{\partial p}{\partial z} + g = 0$$

Are all of these terms important every time?

$$\overset{U/T}{\overset{\uparrow}{\frac{\partial u}{\partial t}}} + \overset{U^2/L}{\overset{\uparrow}{\mathbf{u}\cdot\nabla u}} + \frac{1}{\rho}\frac{\partial p}{\partial x} - \overset{fU}{\overset{\uparrow}{fv}} = 0$$

$$\underset{1/fT}{\underset{\downarrow}{\frac{\partial v}{\partial t}}} + \underset{U/(fL)}{\underset{\downarrow}{\mathbf{u}\cdot\nabla v}} + \frac{1}{\rho}\frac{\partial p}{\partial y} + \underset{1}{\underset{\downarrow}{fu}} = 0$$

$$R_o = \frac{U}{fL} \longrightarrow$$ **Rossby number**

- First, we consider flows where friction can be neglected.
- Then, let's guess how big each term can be.
- **For typical large-scale flows in the atmosphere:**

  - U ~ 10 m s^{-1} (horizontal velocity scale)
  - W ~ 1 cm s-1 (vertical velocity scale)
  - L ~ 10^6 m (length scale)
  - T ~ 10^5 s (time scale)
  - f ~ 10^{-4} s^{-1}
  - $1/\rho\ \partial p/\partial x \sim 10^{-3}$

Geostrophic motion

$$R_o = \frac{U}{fL}$$ $\longrightarrow$ Rossby number ~ 10^{-1}

$$\frac{1}{\rho}\frac{\partial p}{\partial x} - fv = 0$$

$$\frac{1}{\rho}\frac{\partial p}{\partial y} + fu = 0$$

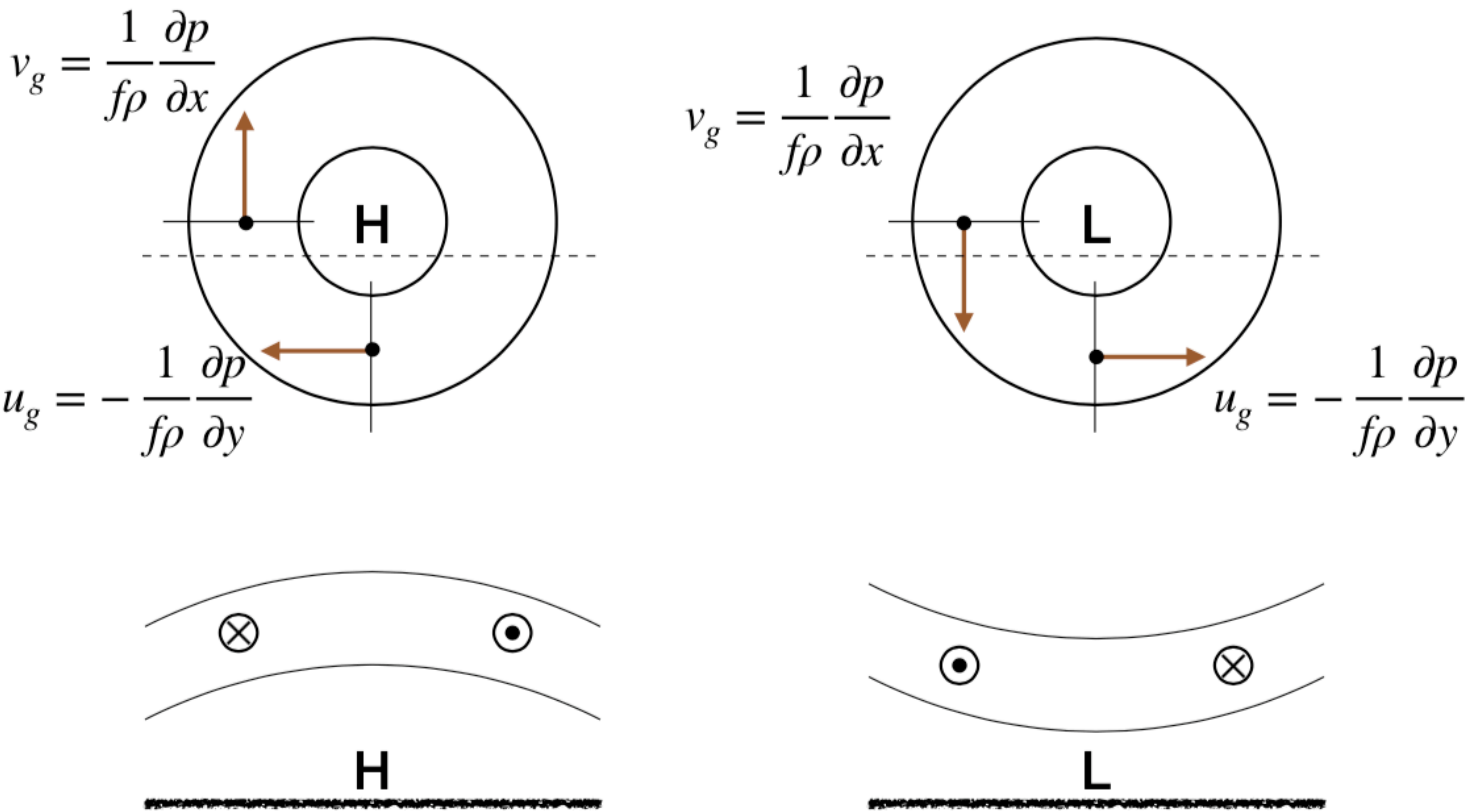
Geostrophic balance

$$u_g = -\frac{1}{f\rho}\frac{\partial p}{\partial y}$$

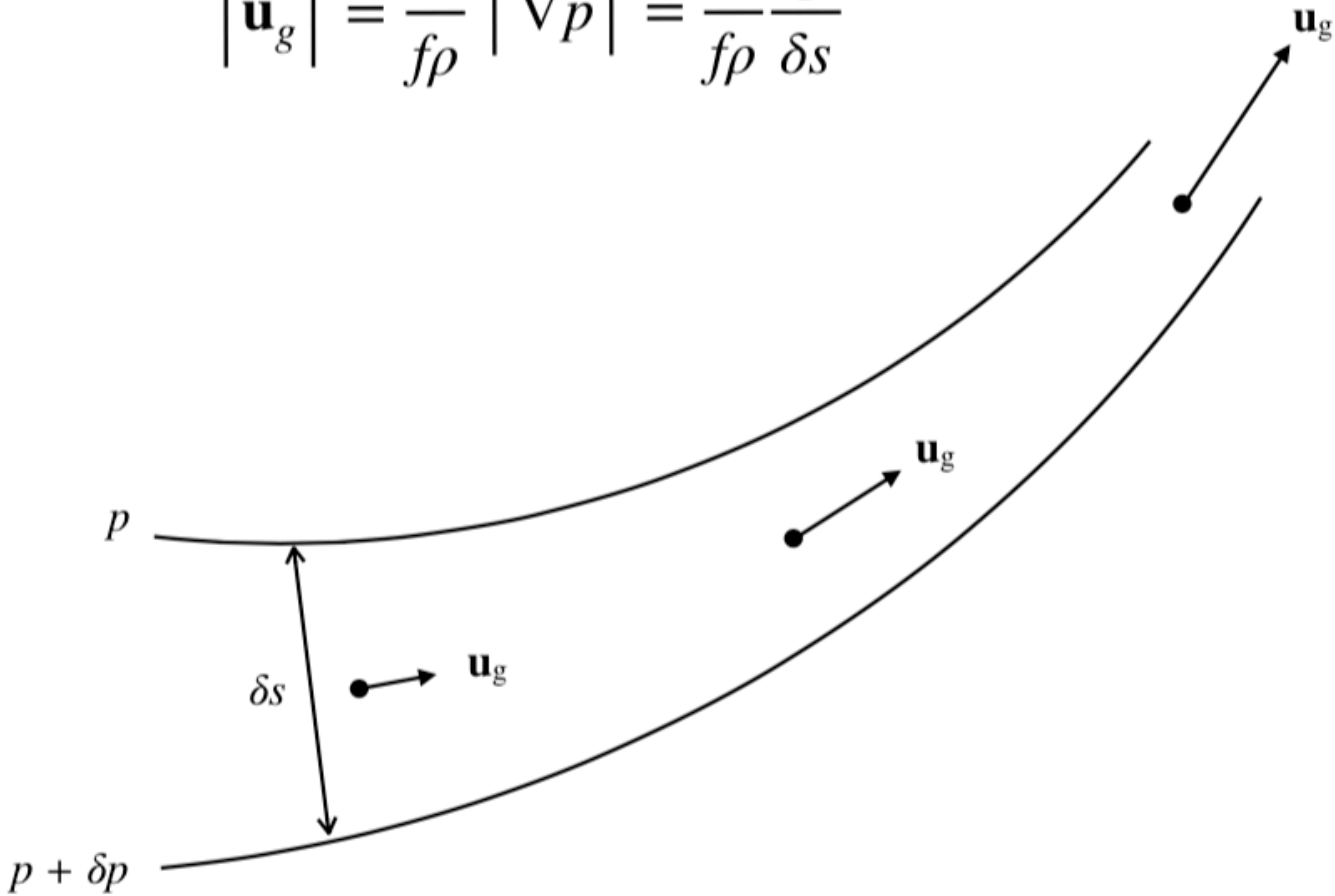
$$v_g = \frac{1}{f\rho}\frac{\partial p}{\partial x}$$

Geostrophic wind

Geostrophic motion



$$\left|\mathbf{u}_g\right| = \frac{1}{f\rho}\left|\nabla p\right| = \frac{1}{f\rho}\frac{\delta p}{\delta s}$$



Geostrophic motion

$$\left(\frac{\partial p}{\partial x}\right)_z = g\rho\boxed{\left(\frac{\partial z}{\partial x}\right)_p}$$

$$\left(\frac{\partial p}{\partial y}\right)_z = g\rho\boxed{\left(\frac{\partial z}{\partial y}\right)_p}$$

$$\longrightarrow$$

$$u_g = -\frac{g}{f}\frac{\partial z}{\partial y}$$

$$v_g = \frac{g}{f}\frac{\partial z}{\partial x}$$

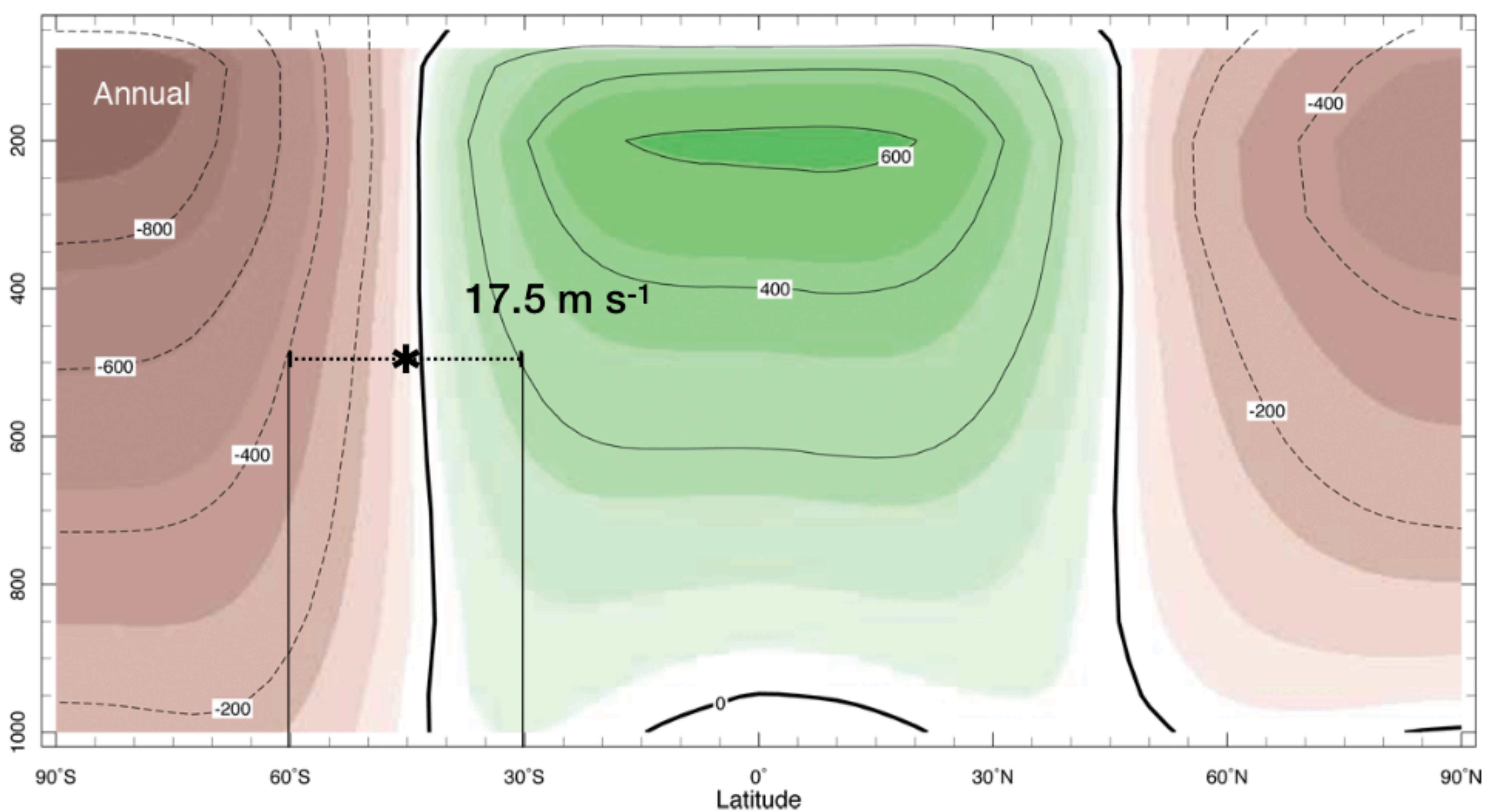
Lateral gradient of Geopotential height

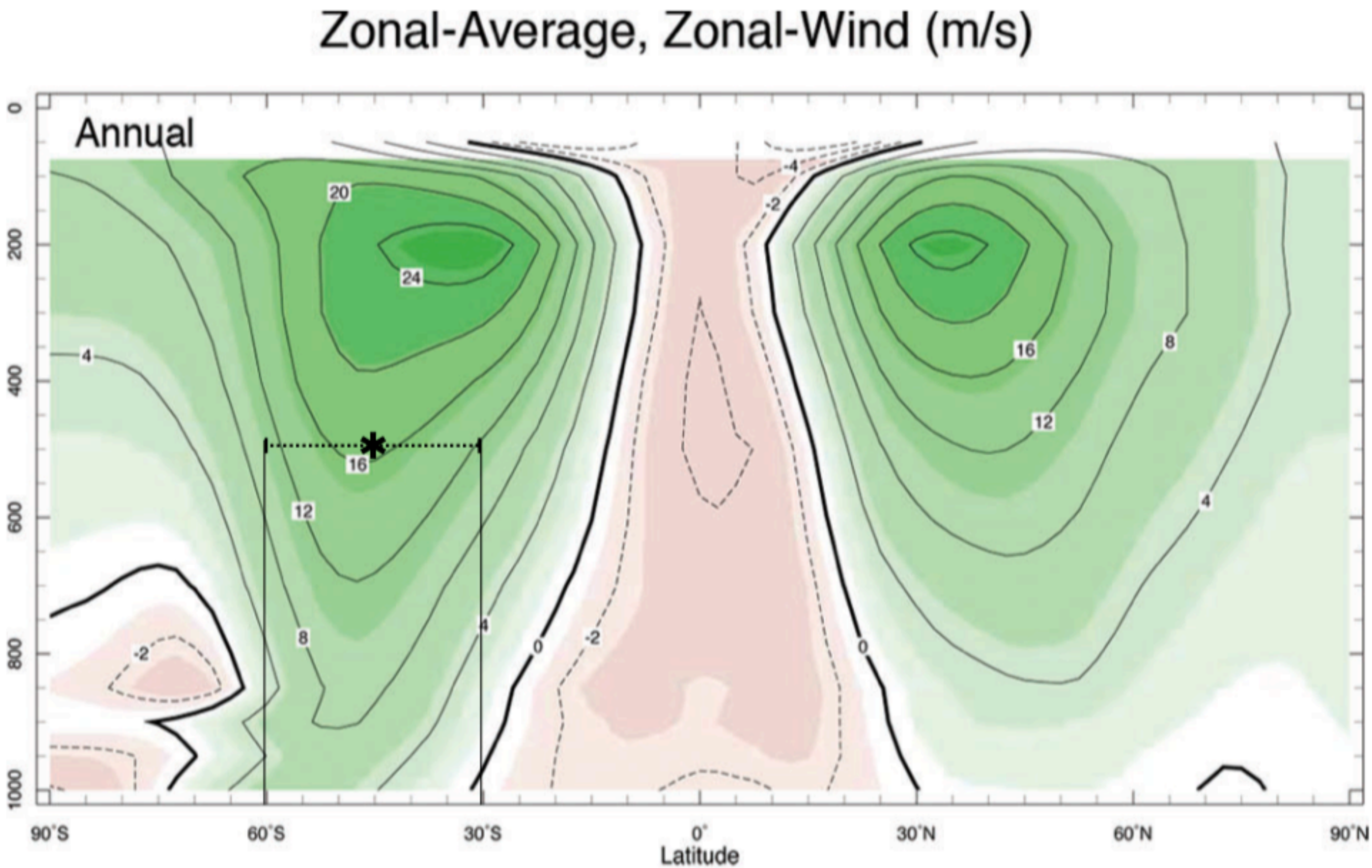
No ρ !

Practice of python HW#2

$$|U_g| = \left|\frac{g}{f}\frac{\partial z}{\partial y}\right| \sim \frac{g}{2\Omega\sin\phi}\frac{\partial z}{\partial y} \sim \frac{9.8}{2\times 7.29\times10^{-5}\times\sin 45^\circ} \times \frac{600\,\mathrm{m}}{2\pi\cdot 6400\mathrm{km}\cdot\frac{36}{360}} \sim 17\,\mathrm{m/s}$$

Zonal-Average Geopotential Height Anomaly (m)




Zonal-Average, Zonal-Wind (m/s)
Annual
20
24
4
16
12
8
4
0
-2
-2
-4
-2
0
16
8
12
4
0
200
400
600
800
1000
90°S
60°S
30°S
0°
30°N
60°N
90°N
Latitude

Practice of python HW#2

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
```

In [5]:
```
!curl https://dl.dropboxusercontent.com/s/skmfwsuc5z50r1g/T_u_inP.npz -o T_u_inP.npz
```

```
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 1397k  100 1397k    0     0  1179k      0  0:00:01  0:00:01 --:--:-- 1179k
```

In [6]:
```
data=np.load('T_u_inP.npz')
```

In [7]:
```
lon=data['lon']
lat=data['lat']
pres=data['pres']
Tair=data['Tair']
uair=data['u']
```

In [8]:
```
print(np.shape(Tair))
```

```
(17, 73, 144)
```

Practice of python HW#2

In [9]:
```
print(np.shape(lon))
print(np.shape(lat))
print(np.shape(pres))
```

```
(144,)
(73,)
(17,)
```

In [10]:
```
umean=np.mean(uair,axis=2)
```

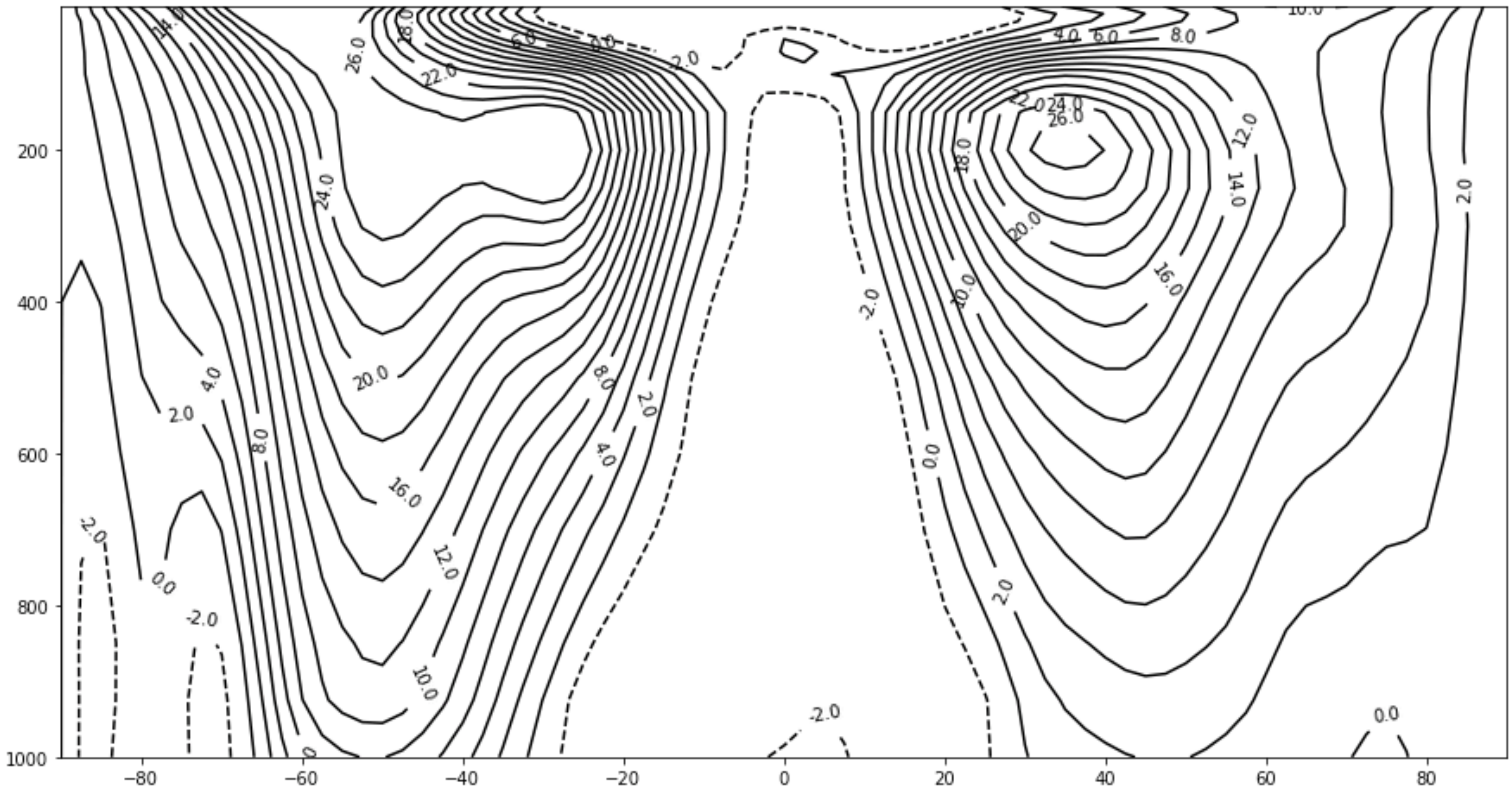
In [11]:
```
umean.shape
```

Out[11]:
```
(17, 73)
```

In [12]:
```
[X,Y]=np.meshgrid(lat,pres)
```

Practice of python HW#2

In [10]:
```
plt.figure(figsize=(15,8))
C=plt.contour(X,Y,umean,range(-2,28,2),colors='black')
plt.clabel(C,fmt="%3.1f")
plt.gca().invert_yaxis()
```



왕초보를 위한 Python 강좌

Lecture 02

T.A 김승기

Content

오늘할 내용은..

그래프 그리기!

Plotting Graph!

1. 2D Graph

2. Contour Graph

(including Guide to HW2)

2D Plot

0. 모듈

```
import matplotlib.pyplot as plt
```

matplotlib.pyplot

1. 명령어

plt.plot(Xdata,Ydata) → X,Y Data는 List 혹은 Array 형태

$$Xdata = (1 \quad 2 \quad 3 \quad 4 \quad 5)$$

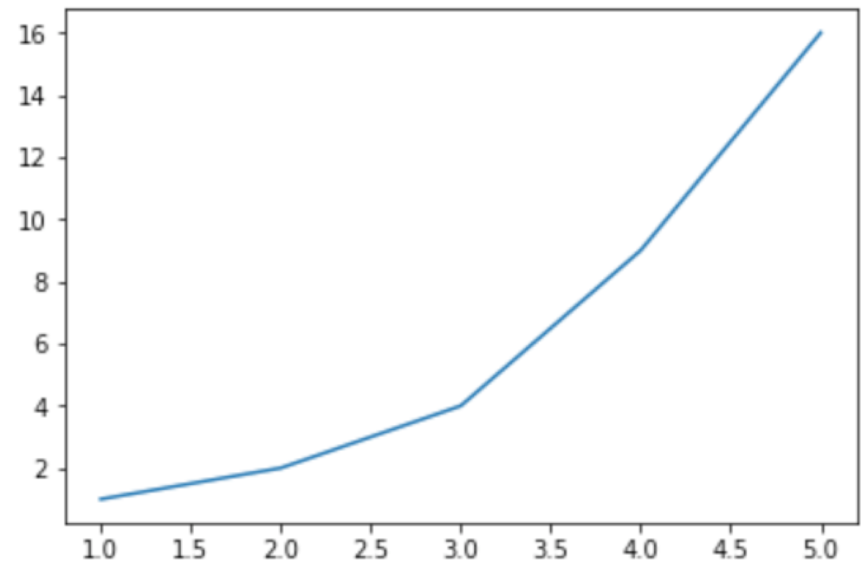
$$Ydata = (1 \quad 2 \quad 4 \quad 9 \quad 16)$$

(1,1) (2,2) (3,4) (4,9) (5,16) → 대응되는 좌표에 점이 찍힌다

2D Plot

2. 예시

```
import numpy as np
import matplotlib.pyplot as plt
plt.plot([1,2,3,4,5], [1,2,4,9,16]) # Xdata : [1,2,3,4,5] , Ydata : [1,2,4,9,16]
plt.show()
```



Python : Lecture 01

2D Plot

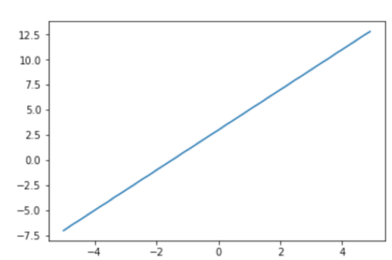
2. 예시

만약 y=2*x+1 그래프를 그리고 싶다면? (x는 -5부터 5까지)

1. Xdata 설정	→	2. Ydata 계산	→	3. Plot(X,Y)
x=[-5.0,-4.9,-4.8,......+5.0]		y=2*x+1		plot(x,y)

Tip) X 행렬을 쉽게 만들수 있는 명령어로 np.linspace(start,end,npoint)를 이용한다.

```
import numpy as np
import matplotlib.pyplot as plt
X = np.linspace(-5., 5., 101) # linspace(시작점,끝점,개수) => 시작점부터 끝점까지 등간격의 실수 Array를 생성한다.
Y = 2*X+3
plt.plot(X,Y)
plt.show()
```



Contour Plot

(등고선 그래프)

1. 명령어

```
plt.contour(Xdata,Ydata,Zdata)
```

→ Xdata, Ydata, Zdata 모두 2차원 배열

X,Y,Z 각각의 행렬 성분에 대응하는 좌표에 점이 찍힌다.
즉, 원리는 2d plot과 똑같다

$$\text{Xdata}=\begin{pmatrix}1 & 2\\ 3 & 4\end{pmatrix} \quad \text{Ydata}=\begin{pmatrix}1 & 2\\ 3 & 4\end{pmatrix} \quad \text{Zdata}=\begin{pmatrix}10 & 20\\ 30 & 40\end{pmatrix}$$

→ (1,1,10) (2,2,20) (3,3,30) (4,4,40)

Contour Plot

2. 예시

만약 z=x^2+y^2 그래프를 그리고 싶다면? (x,y는 -2부터 2까지)

1. X,Y data 설정 ➡ 2. Zdata 계산 ➡ 3. Plot

1. X,Y data 설정

x=linspace(-2,2,11)

y=linspace(-2,2,11)

➡ x,y는 1차원 이므로 2차원화 해줘야 한다!
How?
[X,Y]=np.meshgrid[x,y]

***Meshgrid**

$$X=\begin{matrix} -2.0 & -1.8 & \cdots & 2.0 \\ -2.0 & -1.8 & \cdots & 2.0 \\ \vdots & \vdots & & \vdots \\ -2.0 & -1.8 & \cdots & 2.0 \end{matrix}_{(11\times11)} \qquad Y=\begin{matrix} 2.0 & 2.0 & \cdots & 2.0 \\ 1.8 & 1.8 & \cdots & 1.8 \\ \vdots & \vdots & & \vdots \\ -2.0 & -2.0 & \cdots & -2.0 \end{matrix}_{(11\times11)}$$

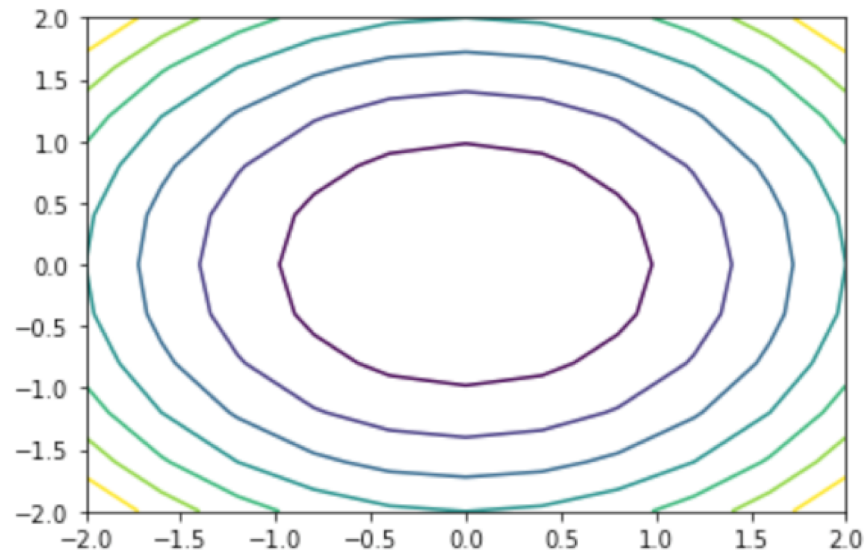
Python : Lecture 01

Contour Plot

2. 예시

만약 z=x^2+y^2 그래프를 그리고 싶다면? (x,y는 -2부터 2까지)

```
import numpy as np
import matplotlib.pyplot as plt
x = np.linspace(-2, 2, 11)
y = np.linspace(-2, 2, 11)
[X,Y]=np.meshgrid(x,y)     # X,Y를 2차원 행렬로 만들어 준다
Z = X**2+Y**2       # meshgrid로 만든 X,Y 2차원 행렬을 이용해서 Z값을 계산해준다.
plt.contour(X,Y,Z)
plt.show()
```



HW2

uair data → **Zonal Mean** → **Contour Plot**

1. 변수

```
lon = data['lon']     # latitude
lat = data['lat']     # longitude
pres = data['pres']   # pressure
Tair = data['Tair']   # annual mean air temperature
uair = data['u']      # annueal mean zonal wind
print np.shape(lon), np.shape(lat), np.shape(pres), np.shape(Tair), np.shape(uair)
```

```
(144,) (73,) (17,) (17, 73, 144) (17, 73, 144)
```

lon	lat	pres	Tair	uair
(144,)	(73,)	(17,)	(17, 73, 144)	(17, 73, 144)

1차원 **3차원** → **pres x lat x lon**

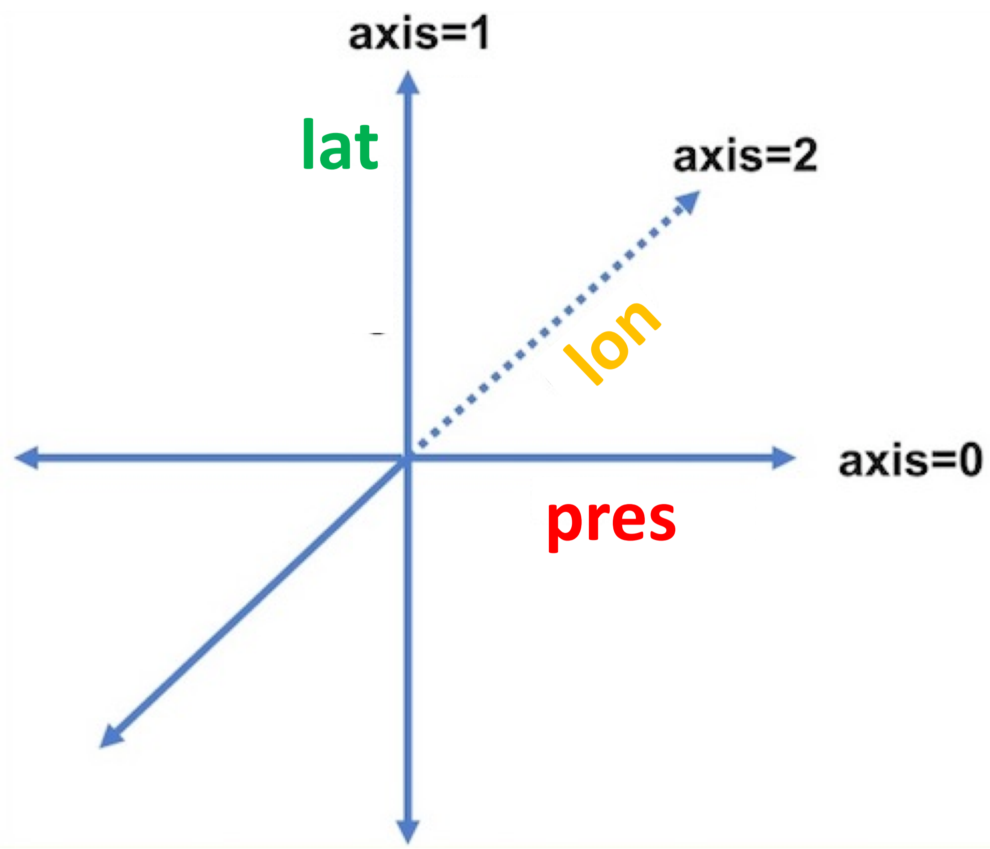
(17) (73) (144)

HW2

uair data ➡ **Zonal Mean** ➡ **Contour Plot**

2. Zonal Mean

uair ➡ **pres** x **lat** x **lon**
(17) (73) (144)



HW2

uair data → Zonal Mean → Contour Plot

3. Contour Plot

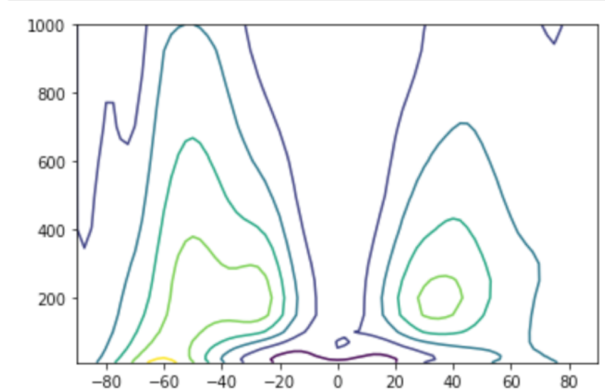
umean → **pres** x **lat**

(17) (73)

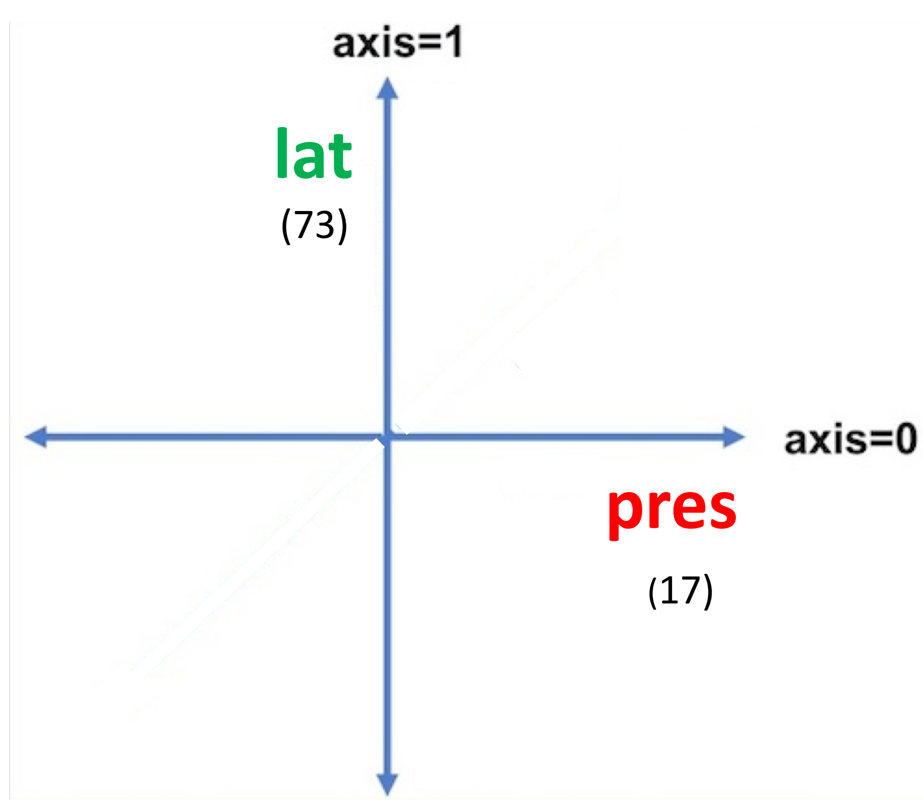
```
[X, Y] = np.meshgrid(lat,pres)
umean = np.mean(uair, axis=2)
print np.shape(X), np.shape(Y), np.shape(umean)
```

```
(17, 73) (17, 73) (17, 73)
```

```
plt.contour(X, Y, umean)
plt.show()
```



-umean



Thank You

- End of the Document -