

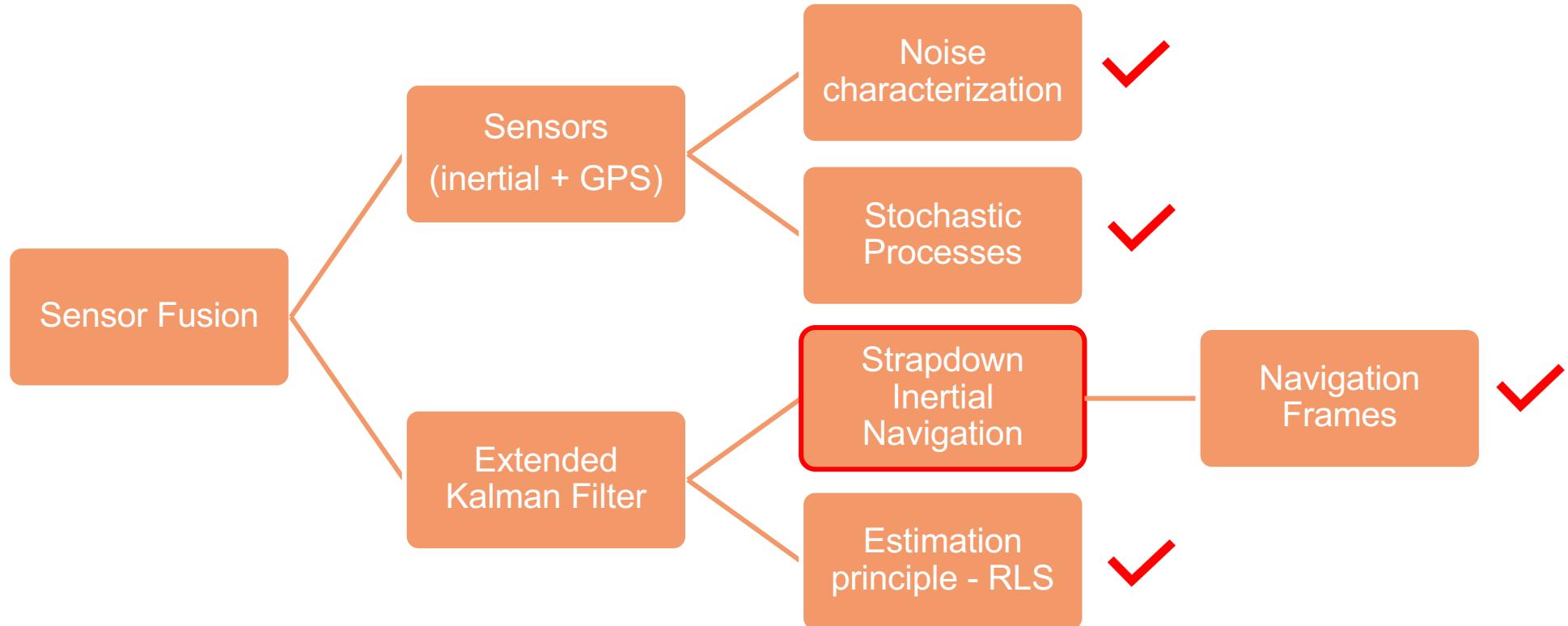
Sensor Orientation Initial Attitude

Jan SKALOUD

Cockpit view of SO course's topics

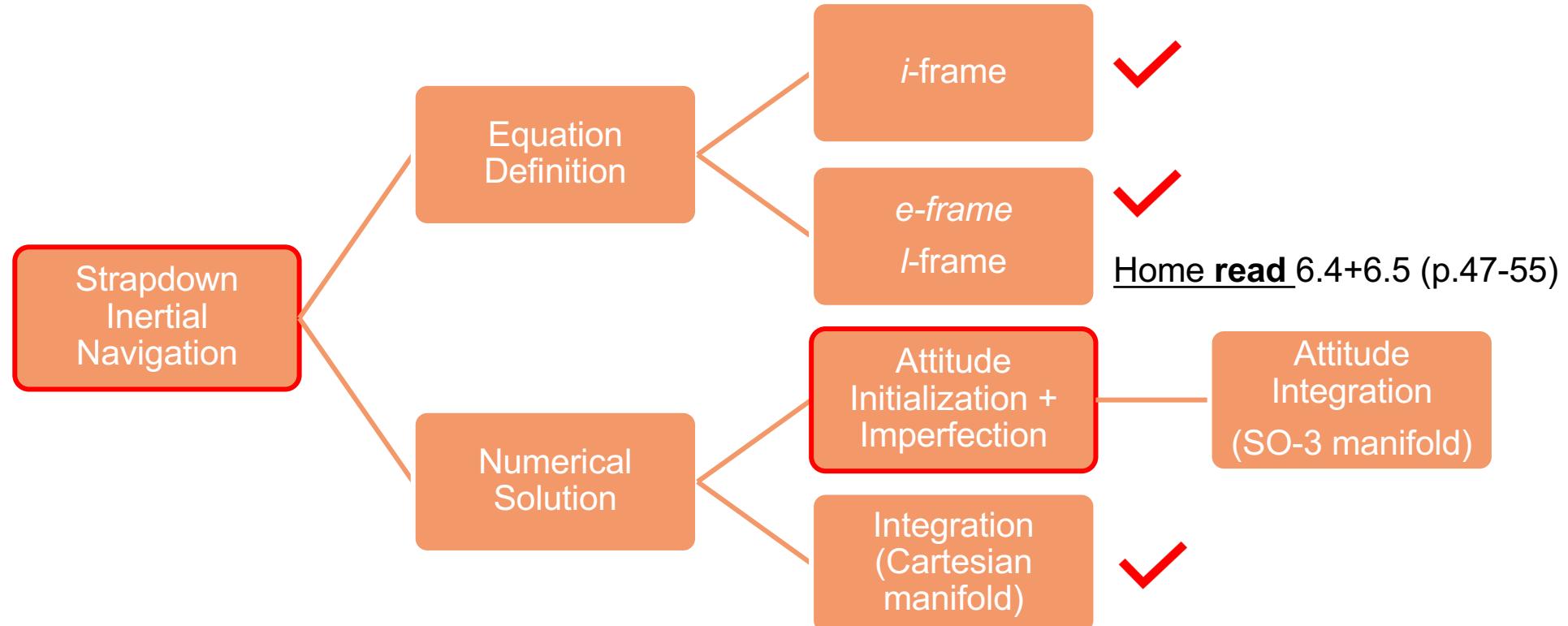
How to reach *integrated* sensor orientation?

■ Sensor orientation



Cockpit view of inertial navigation

Prerequisite for reaching *integrated* sensor orientation



Inertial navigation – agenda

Navigation equations

- *i-frame* (Week 5)
- *e-frame* & SHOW CASE
- *l-frame (local-level)* – **polycopié (6.4)** to read for next week!

Attitude (Week 6)

- Initialization – how ?
- Initialization – imperfections & impact

Strapdown inertial navigation (Week 7)

- Attitude solution in 3D
- Review of differences *e-frame*, *l-frame*
- Strapdown inertial navigation – **polycopié (6.5)** to read for next week!
- Impact of error accumulation

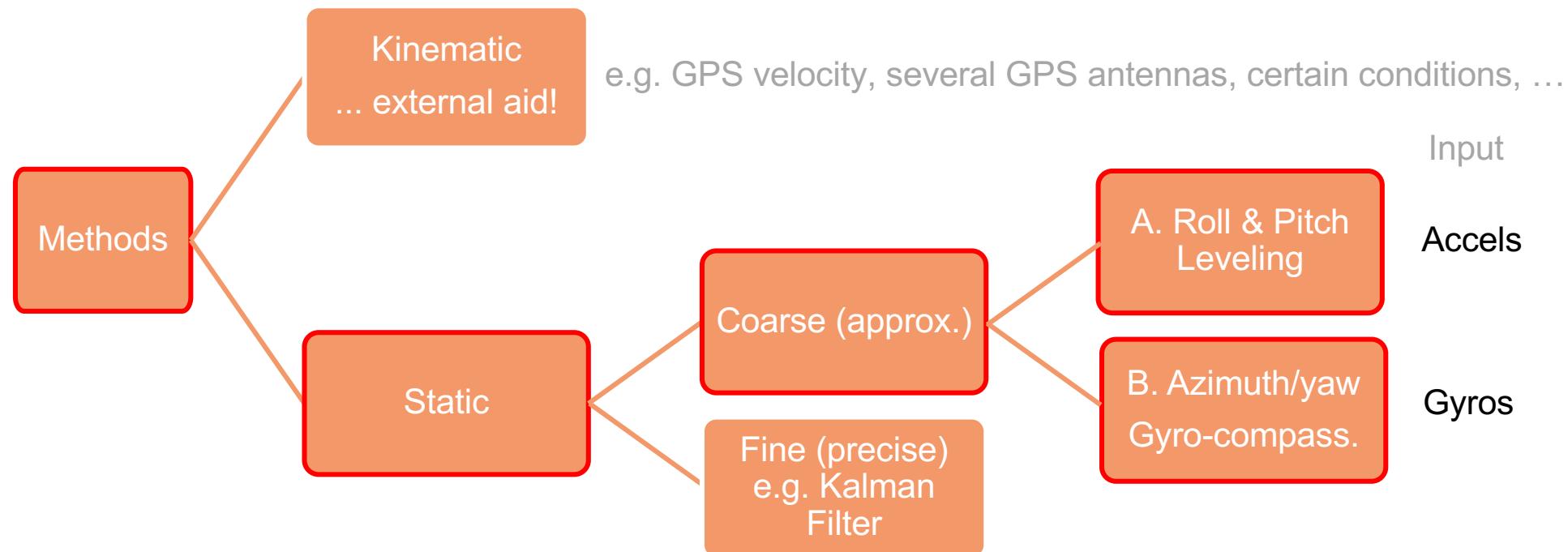
Attitude initialization (in a navigation jargon = Initial Alignment Ch.7)

- Goal: to determine

$$\mathbf{R}_b^e \text{ or } \mathbf{R}_b^\ell$$

since $\mathbf{R}_b^e = \mathbf{R}_\ell^e(\varphi, \lambda) \cdot \mathbf{R}_b^\ell$

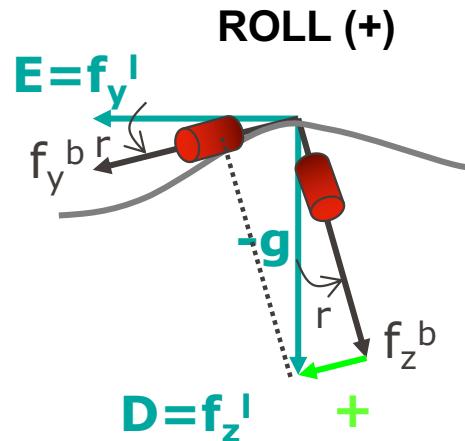
recall $\mathbf{R}_\ell^e(\varphi, \lambda) = [\mathbf{n}^e, \mathbf{e}^e, \mathbf{d}^e] = \dots$ is known!



Attitude initialization – A. Leveling

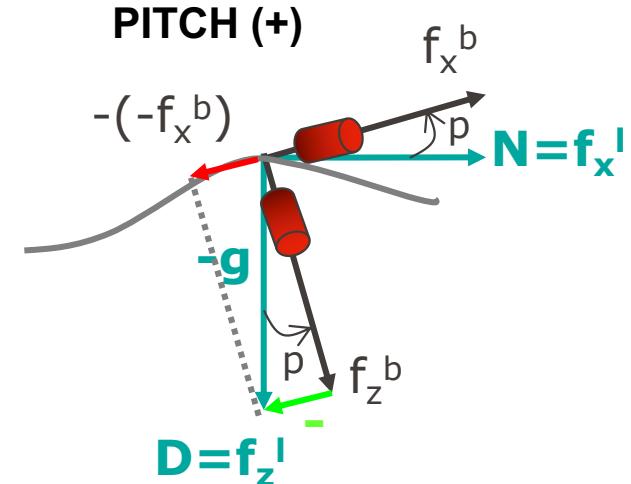
(1) approximation⁽²⁾ for small angles

- ☐ Intuitive determination of roll (r) and pitch (p) angles from **static triad** of accelerometers, through the **projection of local gravity (g) to the sensed specific force (f)**



$$\tan(r) = \frac{-f_y}{-f_z}$$

$$\sin(r) = \frac{f_y}{\|f\|}$$



$$\sin(p) = \frac{f_x}{\|f\|}$$

Notes:

1. for obtaining a correct sign it is better/safer to use function *atan2(nominator, denominator)*
2. approximative because the drawing does not consider a rotational sequence!

Attitude initialization – A. Leveling

(2) starting relation (accelerometers)

- Recall from the previous lecture

$$\cancel{\dot{\mathbf{v}}^e = \mathbf{R}_b^e \mathbf{f}^b - 2\Omega_{ie}^e \mathbf{v}^e + \mathbf{g}^e} \quad \dots \text{is a function of time}$$

- Under static (non-moving) conditions $\mathbf{v}^e = 0$; $\dot{\mathbf{v}}^e = 0$, the above eq. reduces:

$$0 = \mathbf{R}_b^e \mathbf{f}^b + \mathbf{g}^e$$

- By multiplying the above eq. from left by $\mathbf{R}_e^\ell = (\mathbf{R}_\ell^e)^T$ (see slide 7)

$$-\mathbf{R}_e^\ell \mathbf{R}_b^e \mathbf{f}^b = \mathbf{R}_e^\ell \mathbf{g}^e$$

- The above term is equivalent to

$$-\mathbf{R}_b^\ell \mathbf{f}^b = \mathbf{g}^\ell$$

$$-\mathbf{f}^b = \mathbf{R}_\ell^b \mathbf{g}^\ell$$

■ Sensor orientation
 readings from 3
 accelerometers

?

known
 from a model, e.g. Eq. 6.45

$$\mathbf{g}^\ell = [0, 0, \gamma(\varphi, h)]^T$$

Attitude initialization – A. Leveling

(3) final relation (accelerometers)

- Expressing $-\mathbf{f}^b = \mathbf{R}_\ell^b \mathbf{g}^\ell$ in the individual components, while considering yaw=0:

$$-\begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} = \mathbf{R}_1(r) \mathbf{R}_2(p) \mathbf{I}_3(\text{yaw} = 0) \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix}$$

- After substituting $\mathbf{R}_1(r)$, $\mathbf{R}_2(p)$ matrices and performing a multiplication of the right side :

$$-\begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} = \begin{bmatrix} -g \sin(p) \\ g \sin(r) \cos(p) \\ g \cos(r) \cos(p) \end{bmatrix} \quad \begin{array}{l} (1) \\ (2) \\ (3) \end{array}$$

- We obtain 3 equations for two unknowns, several possibilities:

$$(1)/[(2)^2 + (3)^2]$$

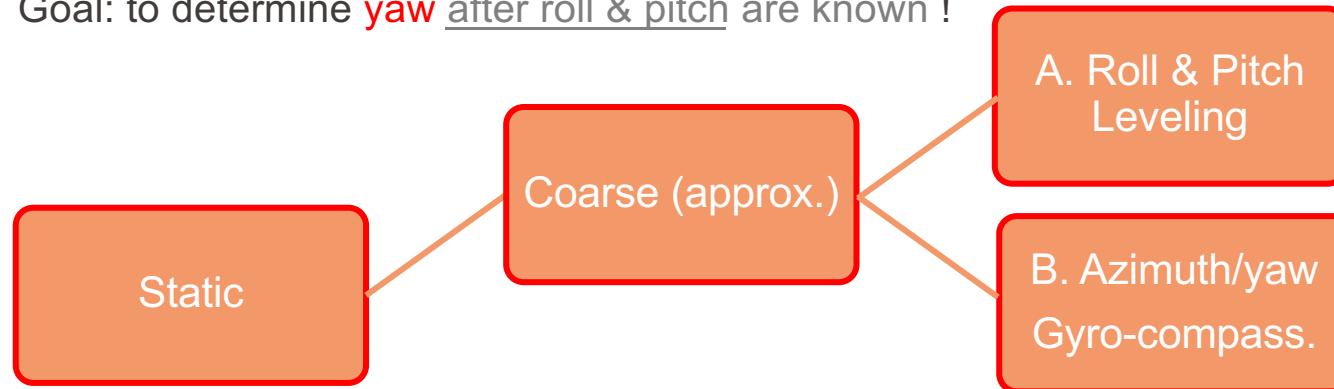


$$(2)/(3) : \tan(r) = \frac{-f_y}{-f_z} \quad (1) : \sin(p) = \frac{f_x}{\|f\|} \quad \tan(p) = \frac{f_x}{\sqrt{f_y^2 + f_z^2}}$$

Note: absolute knowledge of gravity is not needed – all is known from the accelerometers !

Attitude initialization – B. gyro compassing (1)

- Goal: to determine **yaw** after roll & pitch are known !



- How?
 1. *Project gyro observations to “leveled” plane (e.g. plane where $r = p = 0$)*
 2. *Determine yaw from the projected gyro data (formulas via I. drawing, vs. II. equations)*
- Transformation to “leveled” plane :

$$\mathbf{R}_b^{\bar{\ell}} = [\mathbf{R}_1(r) \mathbf{R}_2(p)]^T$$

Note: in such a plane z axis – points down, but x , y axes do not necessary align with North, East

Attitude initialization –

B. gyro compassing (2)

- 1. Projection of gyro readings to a “leveled” plane
(using previously determined roll & pitch):

$$\omega_{ib}^{\bar{\ell}} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}^{\bar{\ell}} = \mathbf{R}_b^{\bar{\ell}} \omega_{ib}^b \quad \mathbf{R}_b^{\bar{\ell}} = [\mathbf{R}_1(r) \mathbf{R}_2(p)]^T$$

Note: in a static (non-rotating with respect to Earth) case

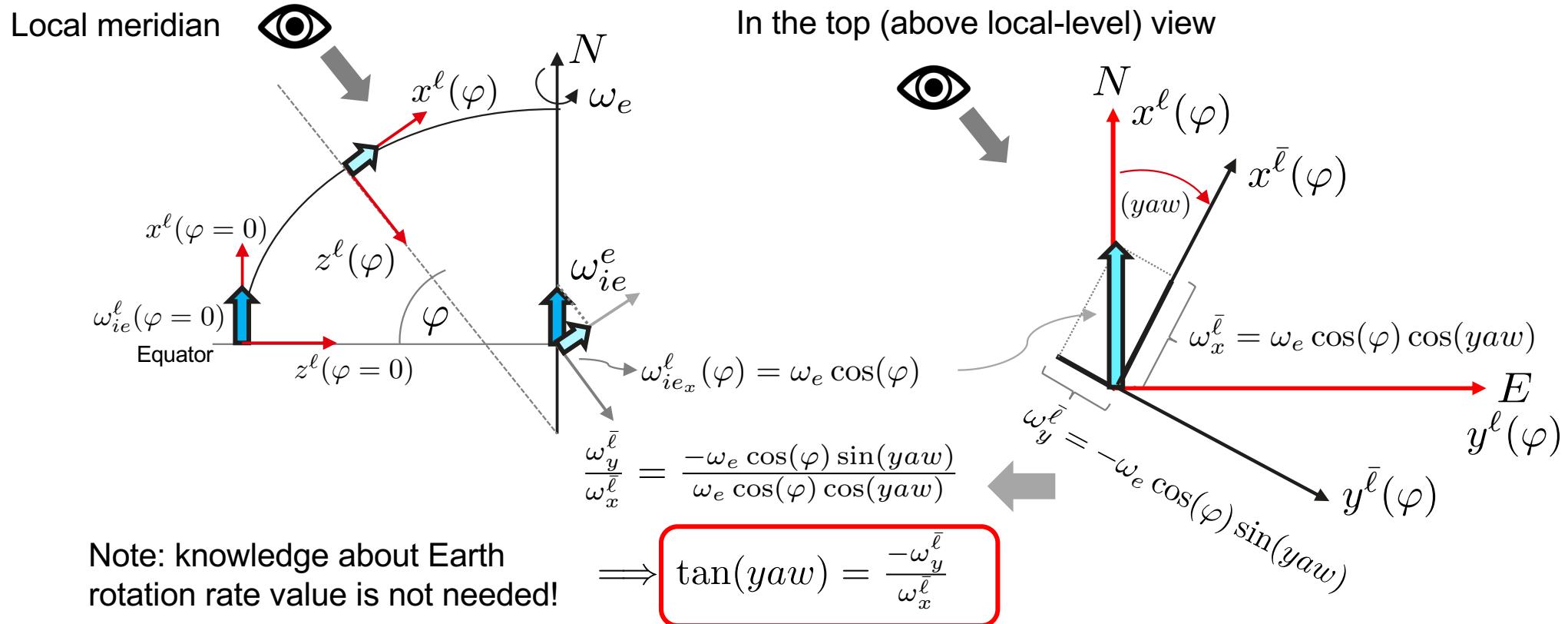
$$\omega_{ib}^b = \omega_{ie}^b + \underbrace{\omega_{eb}^b}_{=0}$$

Therefore, the norm

$$||\omega_{ib}^{\bar{\ell}}|| = ||\omega_{ib}^b|| = \omega_e$$

Attitude initialization – B. gyro compassing (3)

- 2. Determine yaw from the projected gyro data via I. deduction from drawing



Attitude initialization – B. gyro compassing (4)

- 2. Determine yaw from the projected gyro data via II. equations

Here: main idea – details in polycopié (Ch. 7)

$$\omega_{ib}^b = \mathbf{R}_\ell^b \omega_{ie}^\ell = \mathbf{R}_\ell^b \mathbf{R}_e^\ell \omega_{ie}^e$$

Multiplying the above equation from left by \mathbf{R}_b^ℓ

Decomposition:

$$\mathbf{R}_b^\ell \omega_{ib}^b = \mathbf{R}_e^\ell \omega_{ie}^e \quad \text{Eq. 3.19}$$

$$\mathbf{R}_3^T \mathbf{R}_2^T \mathbf{R}_1^T \omega_{ib}^b = \begin{bmatrix} \dots & \dots & \cos \varphi \\ \dots & \dots & 0 \\ \dots & \dots & -\sin \varphi \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \omega_e \end{bmatrix}$$

$$\underbrace{\mathbf{R}_3^T \mathbf{R}_2^T \mathbf{R}_1^T}_{\omega_{ib}^\ell} = \begin{bmatrix} \cos(y) & -\sin(y) & 0 \\ \sin(y) & \cos(y) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \begin{bmatrix} \omega_x^\ell \\ \omega_y^\ell \\ \omega_z^\ell \end{bmatrix}$$

$$\begin{bmatrix} \cos(y) & -\sin(y) & 0 \\ \sin(y) & \cos(y) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \omega_x^\ell \\ \omega_y^\ell \\ \omega_z^\ell \end{bmatrix} = \begin{bmatrix} \omega_e \cos \varphi \\ 0 \\ -\omega_e \sin \varphi \end{bmatrix} \quad \begin{array}{l} (1) \\ (2) \\ (3) \end{array}$$

from 2nd Eq.:

$$\tan(y) = \frac{-\omega_y^\ell}{\omega_x^\ell}$$

Accel. leveling – limiting factors

Q1 – What is the expected accuracy derived from leveling process?

- Since (in NED):

$$\sin(r) = -f_y/g \rightarrow \text{roll}$$

$$\sin(p) = f_x/g \rightarrow \text{pitch}$$

- The roll, pitch accuracy is governed by accel's. accuracy (mainly bias b). For small angles:

$$\Delta r = -b(f_y)/g$$

$$\Delta p = b(f_x)/g$$

- Example: 10 mg bias \rightarrow leveling error of ...?

■

Gyro compassing – limiting factors

- Considering NED and small yaw error:

- $\sin(y) \sim y$ and $\cos(y) \sim 1$ in

$$\bar{\omega_x} = \omega_e \cos(\varphi) \cos(y) \sim \omega_e \cos \varphi$$

$$\bar{\omega_y} = -\omega_e \cos(\varphi) \sin(y) \sim \omega_e \cos \varphi \cdot (-y)$$

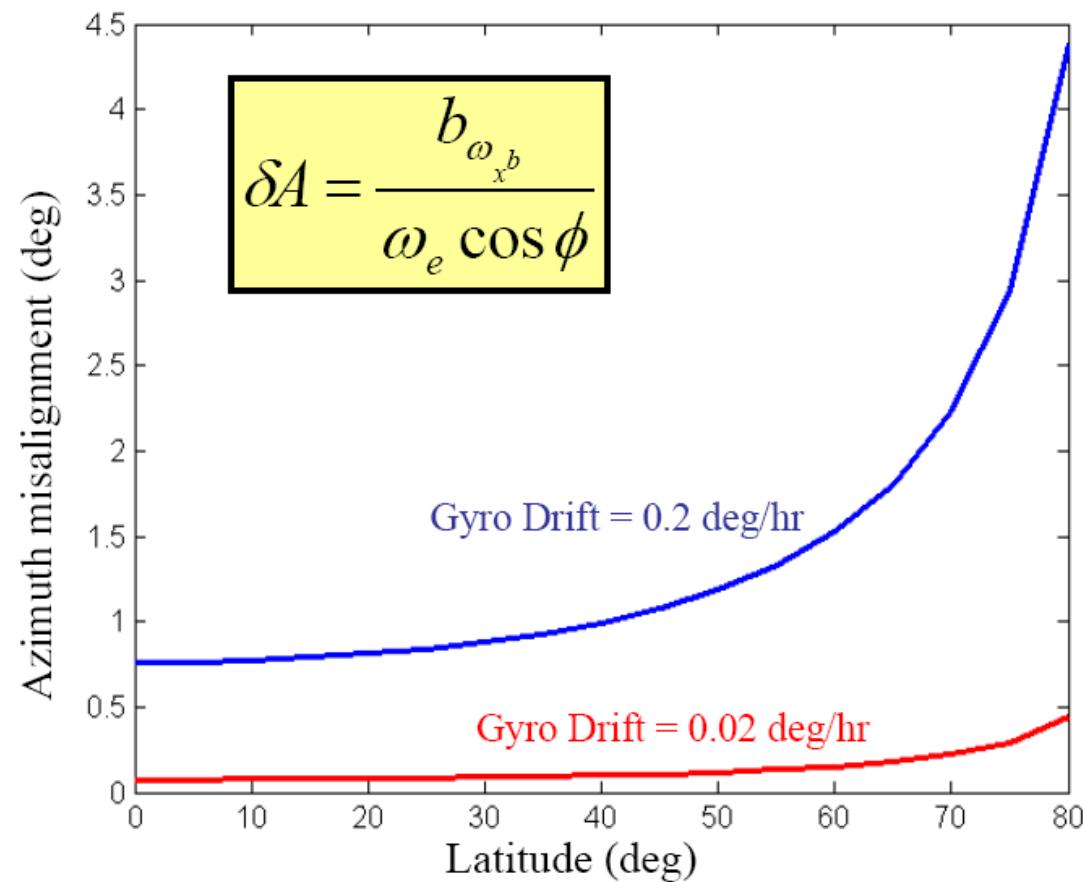
- → Yaw accuracy will mainly depend on the gyro accuracy – bias (b), e.g.:

$$\delta y = \frac{b_{\bar{\omega_y}}}{\omega_e \cos \varphi}$$

- Example: 45 deg latitude, gyro bias 0.2 deg/h

- → Yaw error?

Azimuth (=yaw) accuracy vs. gyro bias (drift)



Azimuth (=yaw) accuracy vs. gyro noise

- Integrated gyro noise produces Angular Random Walk bias (b_{RW})
- a_{RW} depends on noise level and the integration time!

$$\delta y = \frac{b_{RW}}{\omega_e \cos \varphi \sqrt{T}}$$

- -> For a given b_{RW} we can achieve different azimuth alignment accuracy as a function of averaging time T!

■

Effect of gyro angular RW on alignment time

