

How optic flow based state estimation may explain the insects' wobbles and their ups & downs?

Franck Ruffier

CNRS Research Director
Institute of Movement Sciences

CNRS / Aix-Marseille University, France

franck.ruffier@univ-amu.fr

www.ism.univ-amu.fr/ruffier



Thanks to :

Lucia Bergantin (PhD student)

Xavier Daïni (PhD student)

Charles Coquet (Postdoc)

José Castillo (Postdoc)

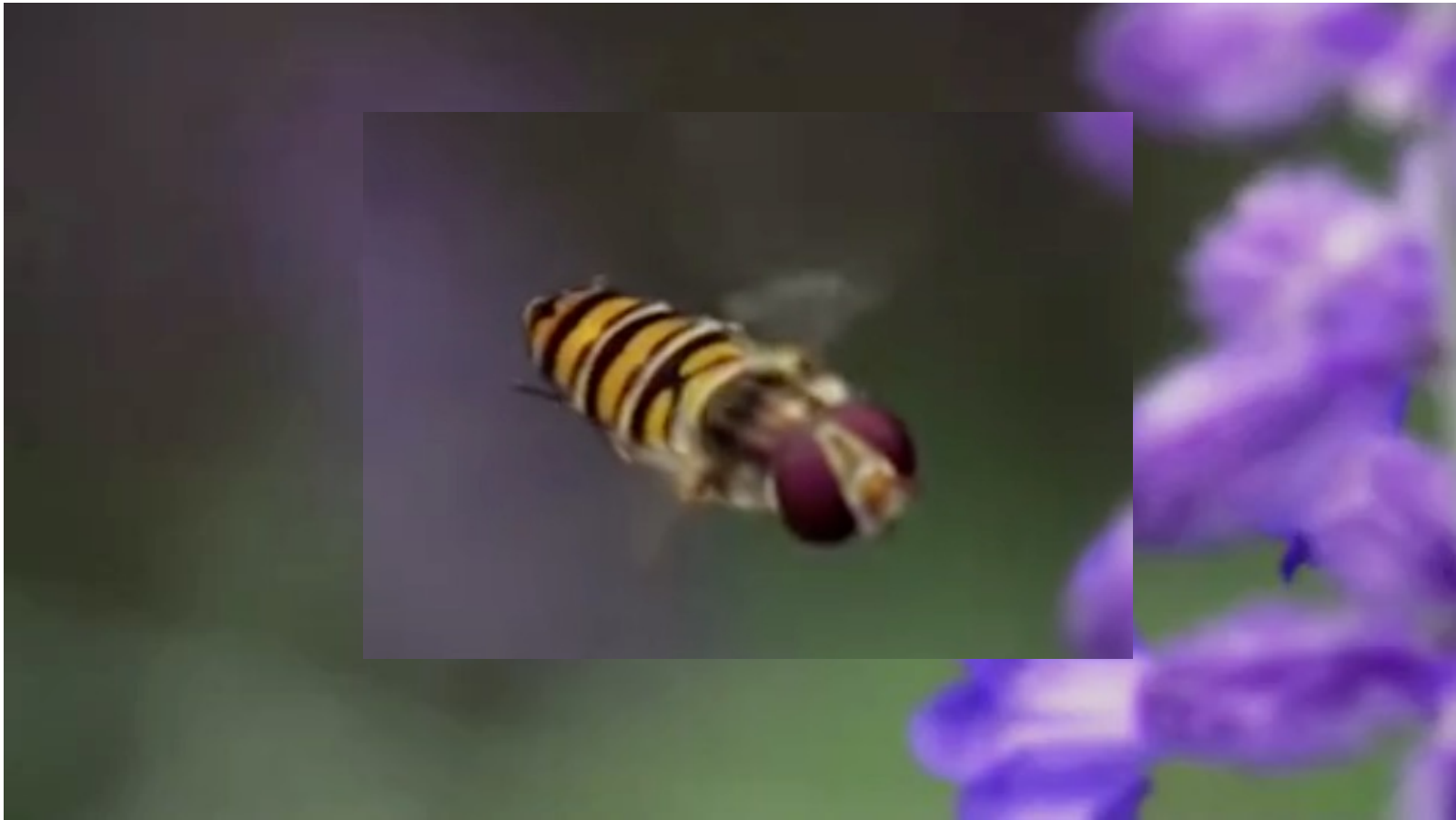
J.-M. Ingargiola (Staff Engineer)



Insects oscillations

- Flying insects' attitude wobbles
- Flying insects' ups & downs

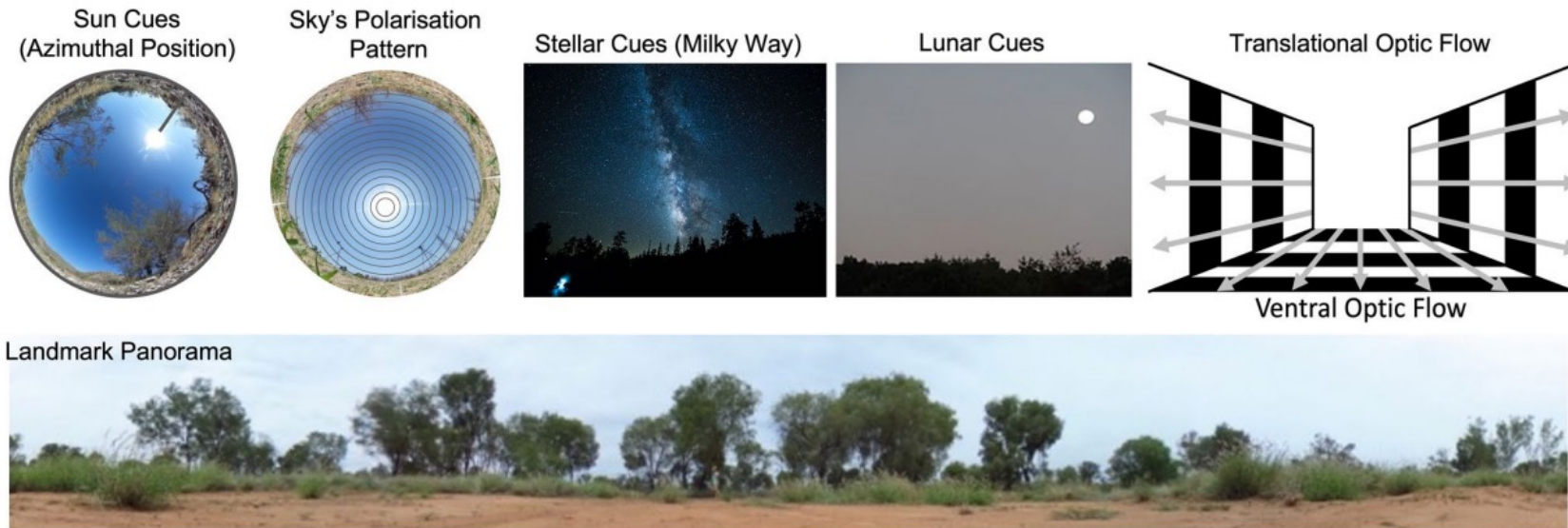
Syrphid flies



Conquest of the skies (BBC)

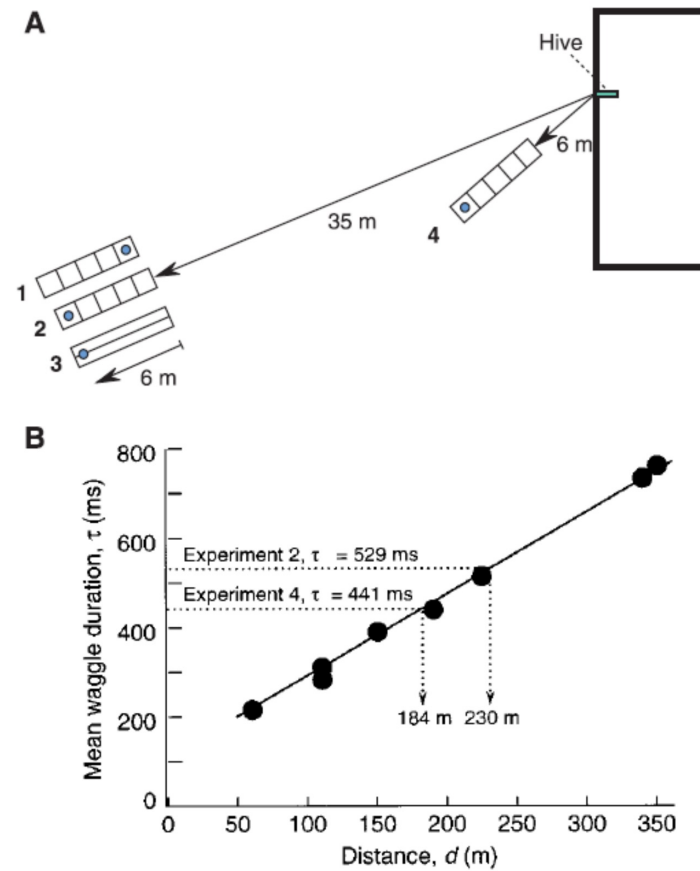


Insects navigate by vision



[Freas and Spetch, 2022]

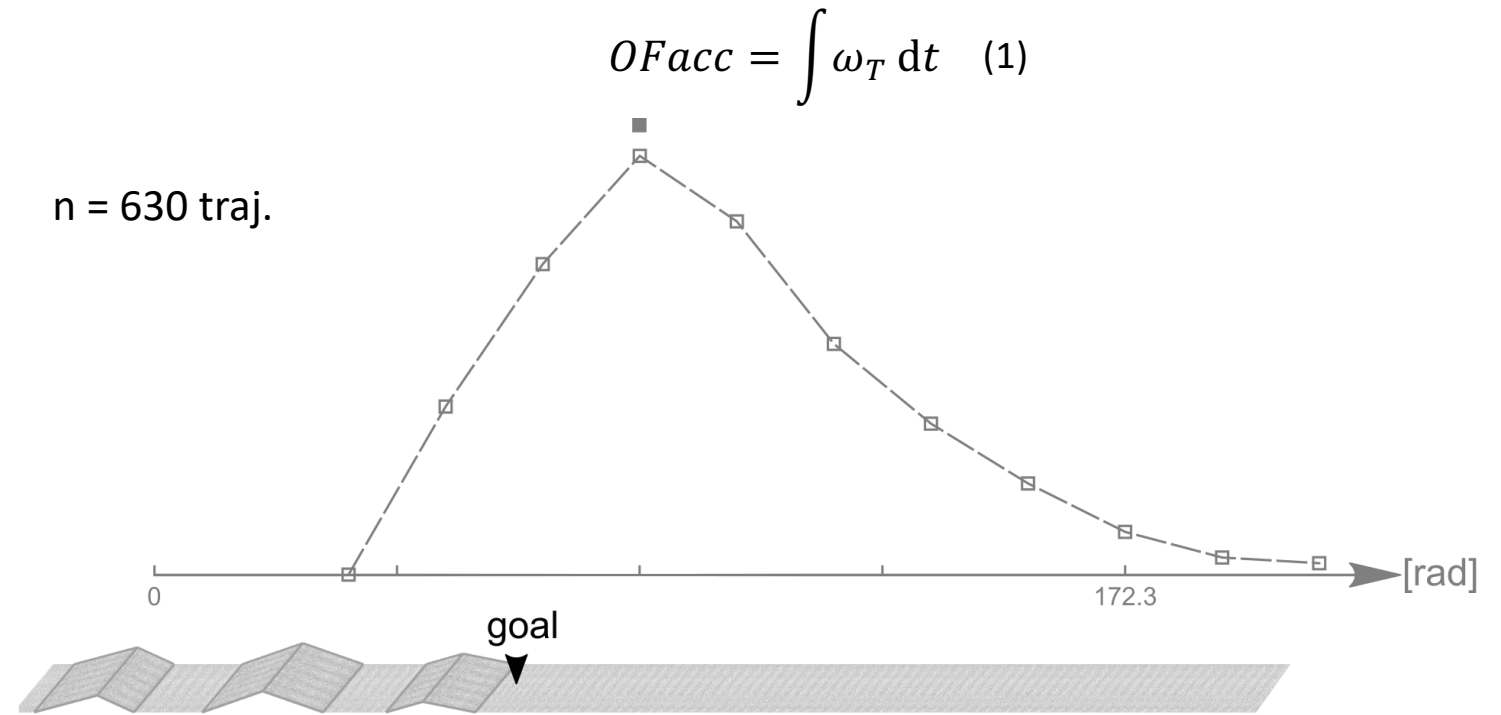
Honeybees perceive their distance travelled



[Srinivasan et al., 2000]

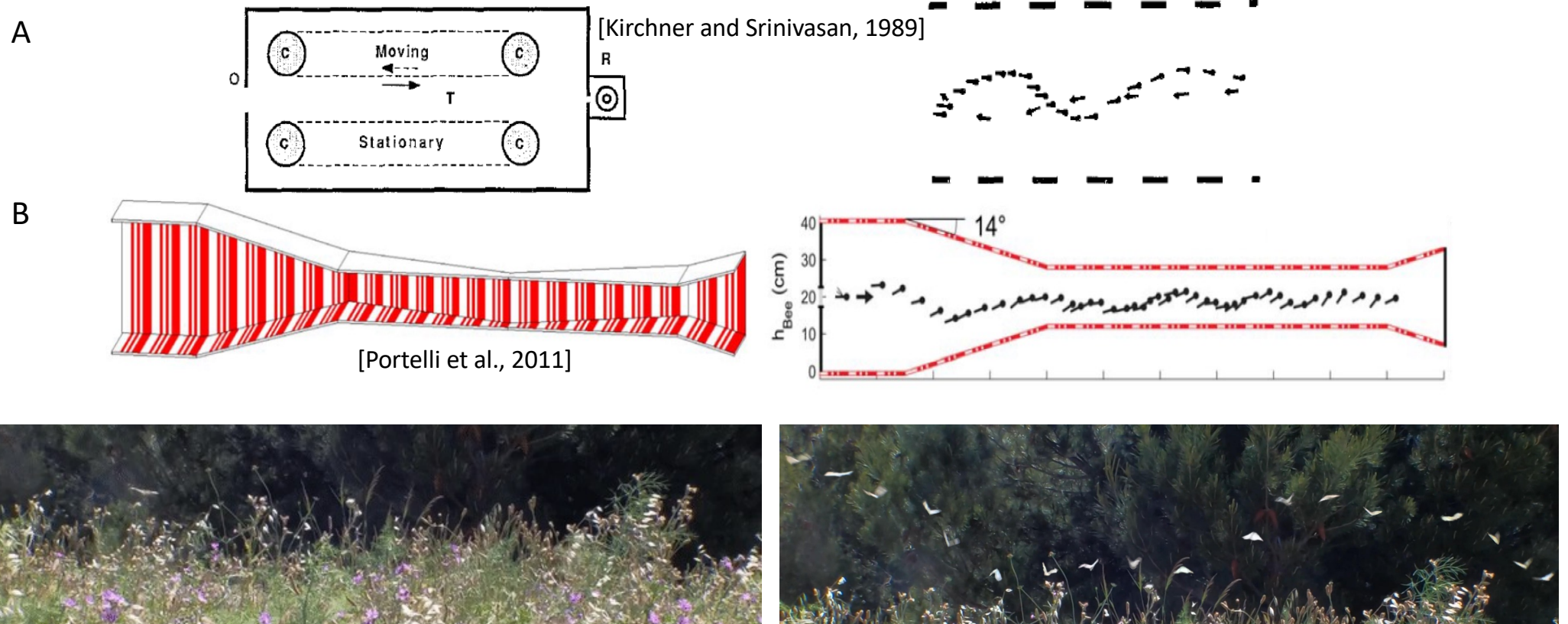
Measuring the distance travelled

In the literature: accumulation of raw translational optic flow ω_T (*OFacc* model)



[Bergantin et al., 2021]

Oscillations in flying insects



Optic flow cues

Translational optic flow:

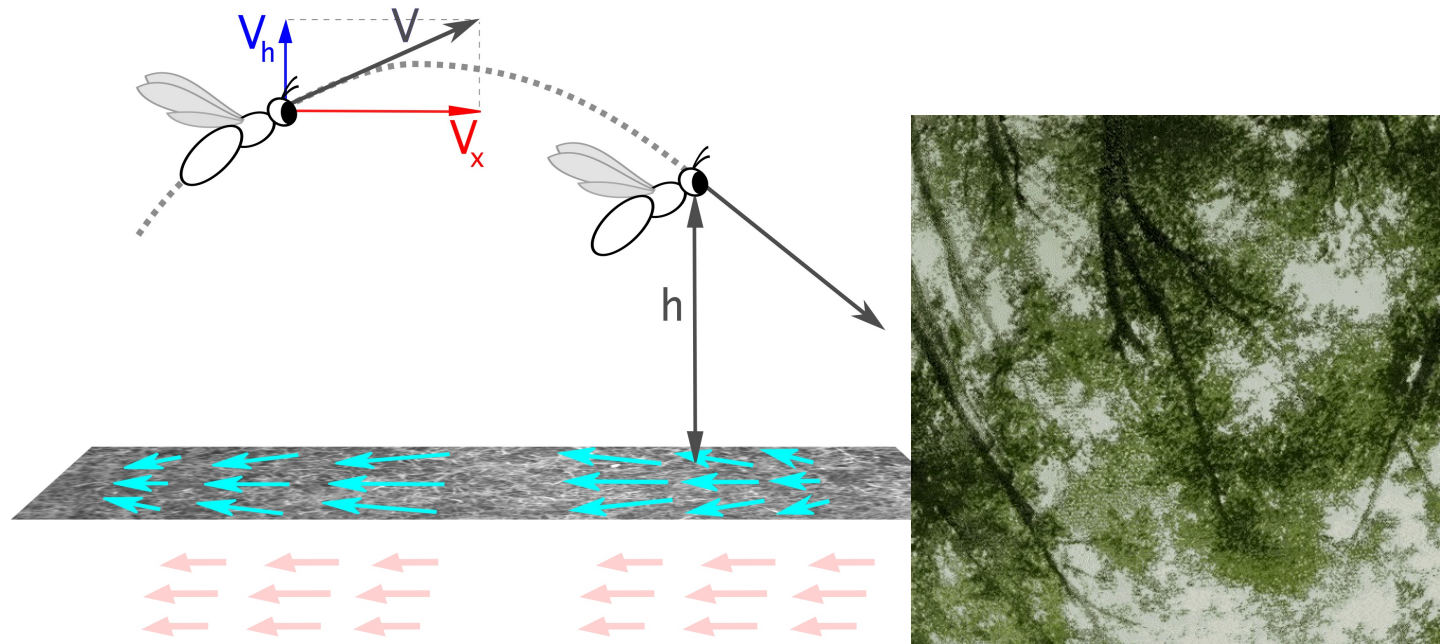
$$\omega_T^{th} = \frac{v_x}{h} \quad (2)$$

Optic flow divergence:

$$\omega_{div}^{th} = \frac{v_h}{h} \quad (3)$$

used to observe the state
Vector [Ho et al., 2017]

$$X = [h; v_h] \quad (4)$$



Optic flow cues

Translational optic flow:

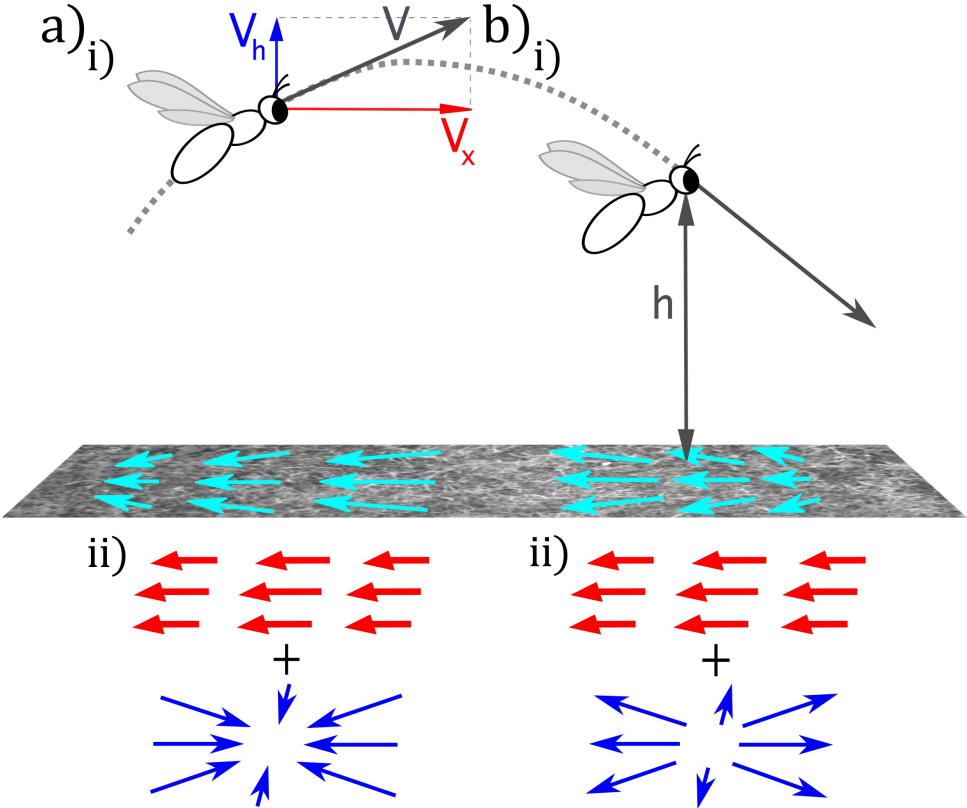
$$\omega_T^{th} = \frac{v_x}{h} \quad (2)$$

Optic flow divergence:

$$\omega_{div}^{th} = \frac{v_h}{h} \quad (3)$$

used to observe the state vector [Ho et al., 2017]

$$X = [h; v_h] \quad (4)$$



[Bergantin et al., 2021]

Observability analysis

Simplified honeybee vertical dynamics [Portelli et al., 2010]:

$$\left\{ \begin{array}{l} \dot{h} = v_h \\ \dot{v}_h = -\frac{v_h}{\tau_z} + \frac{K_z}{\tau_z} \Delta\Phi \\ \omega_{div} = \frac{v_h}{h} \end{array} \right. \quad (5) \longrightarrow \left\{ \begin{array}{l} \dot{x}(t) = f(x(t), \Delta\Phi(t)) = \begin{bmatrix} \dot{h}(t) \\ \dot{v}_h(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{1}{\tau_z} \end{bmatrix} \begin{bmatrix} h \\ v_h \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K_z}{\tau_z} \end{bmatrix} \Delta\Phi \\ y(t) = g(x(t)) = \frac{v_h}{h} = \frac{\dot{h}}{h} = \omega_{div} \end{array} \right. \quad (6)$$

Observability matrix:

$$O = \begin{bmatrix} L_f^0(g(x(t))) \\ L_f^1(g(x(t))) \end{bmatrix}$$

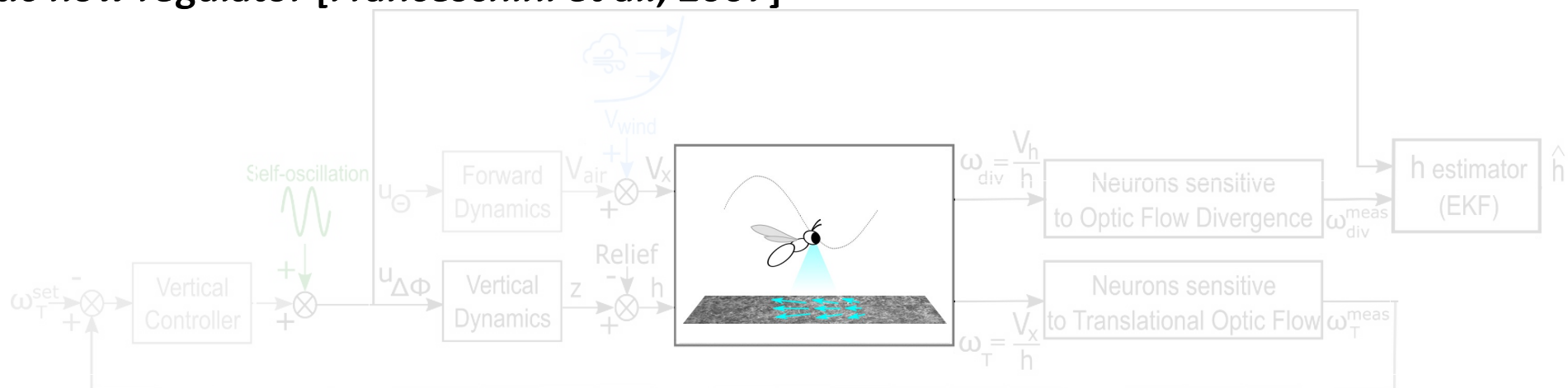
—————> Locally observable if
 $u \neq 0, h \neq 0, v_h \neq 0$
as in [Ho et al., 2017]

SOFla Simulations

- Self-oscillations [Kirchner and Srinivasan, 1989; Portelli et al., 2011]
- Honeybee dynamics [Portelli et al., 2010]
- Neurons sensitive to translational optic flow [Ibbotson, 1991; Stone et al., 2017]
- Neurons sensitive to optic flow divergence [Bidwell and Goodman, 1993]
- Optic flow regulator [Franceschini et al., 2007]

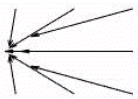
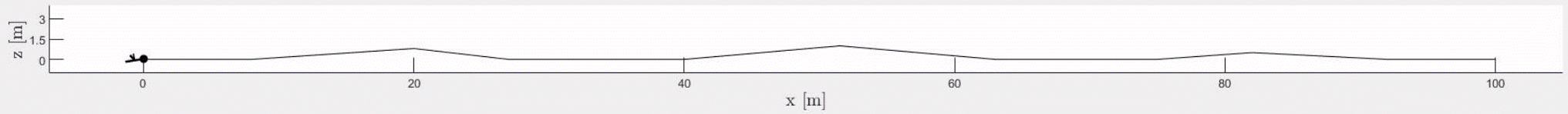
$$\dot{x}(t) = f(x(t), \Delta\Phi(t)) = \begin{bmatrix} \dot{h}(t) \\ \dot{v}_h(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & -\frac{1}{\tau_z} \end{bmatrix} \begin{bmatrix} h \\ v_h \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K_z}{\tau_z} \end{bmatrix} \Delta\Phi \quad (6)$$

$$y(t) = g(x(t)) = \frac{v_h}{h} = \frac{\dot{h}}{h} = \omega_{div}$$



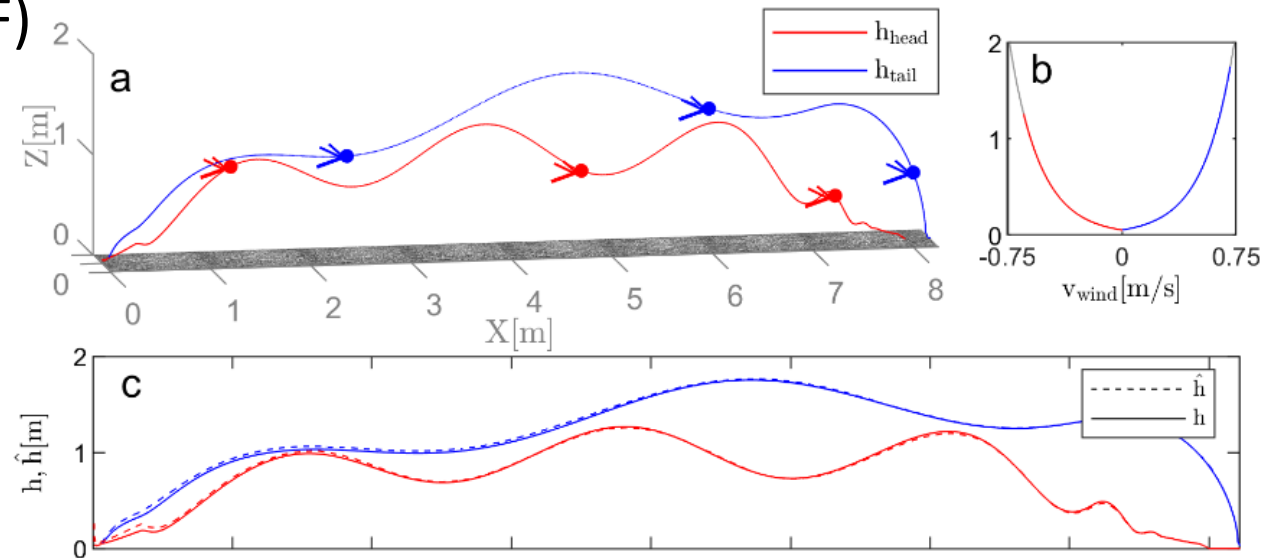
[Bergantin et al., 2021]

SOFla Simulations



Estimation of the ground height

Estimation of the ground height with an **Extended Kalman Filter (EKF)**



→ \hat{h} can be used as a scaling factor

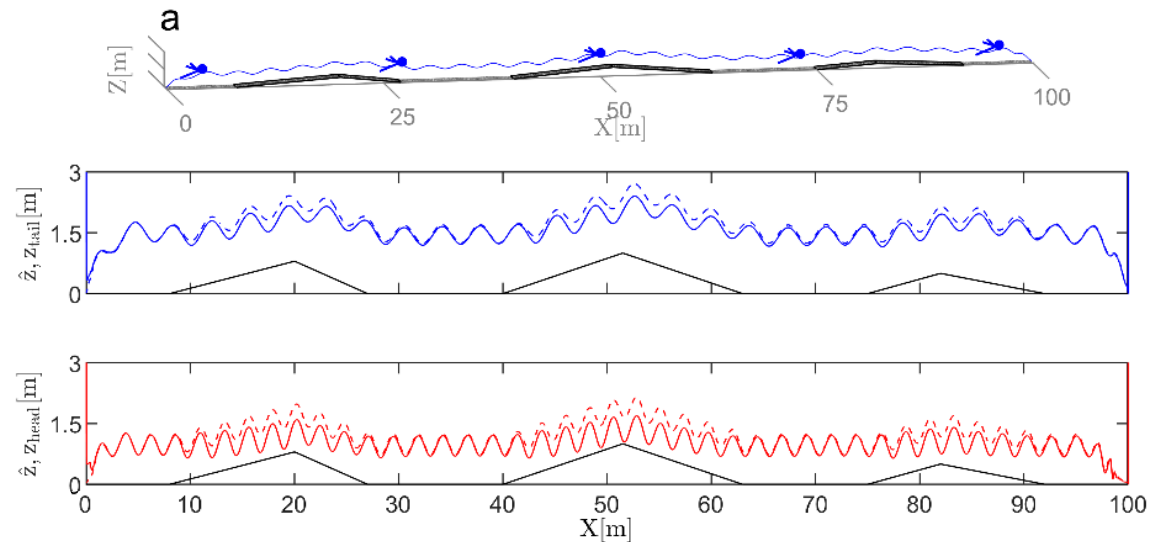
The *SOFIa* odometer model (Self-scaled Optic Flow Integration)

$$\hat{X}_{SOFIa} = \int \omega_T^{meas} \cdot \hat{h} dt \quad (8)$$

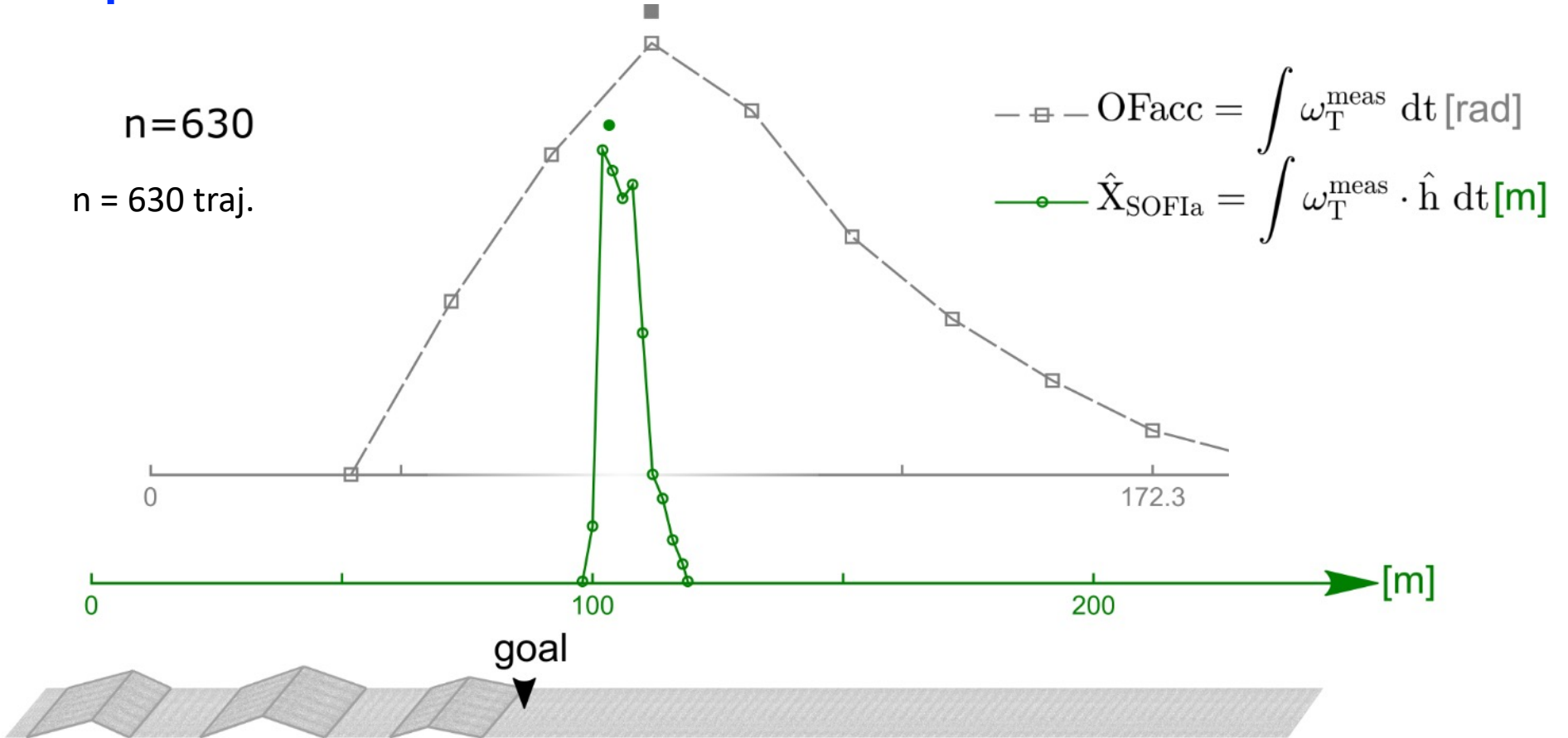
Final % error ($\hat{X}_{SOFIa} - X_{gt}$)

odometry of example:

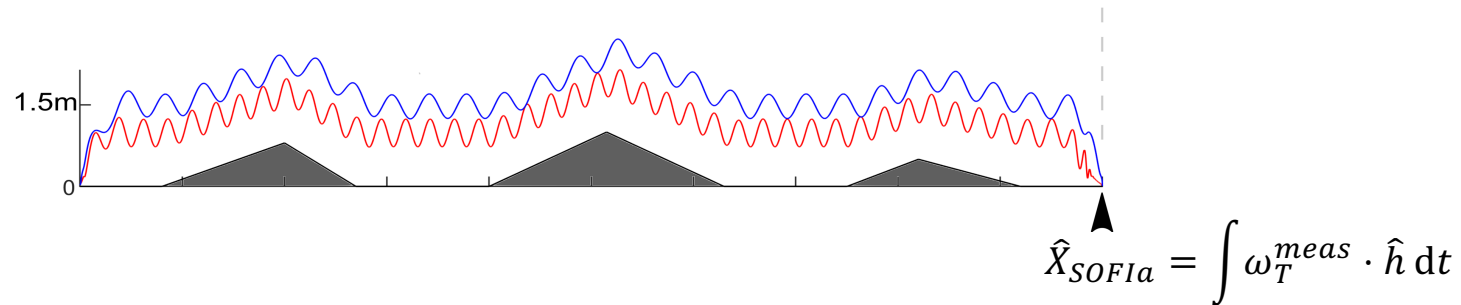
- 0,69% tail wind
- -1,8% head wind



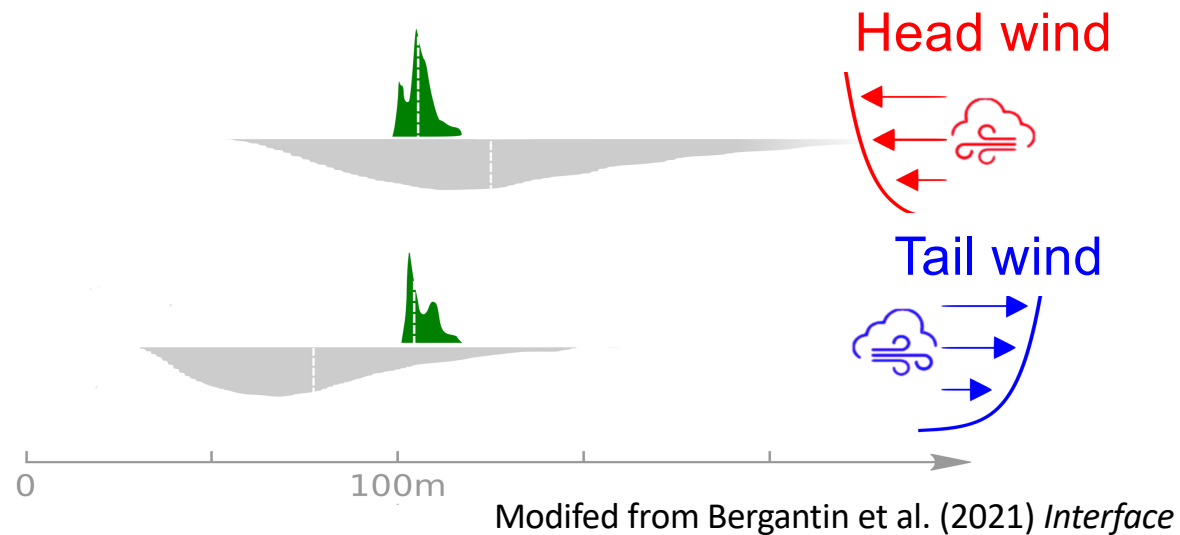
Comparison of models



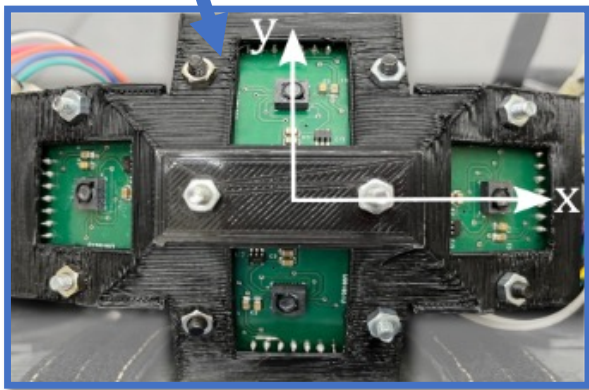
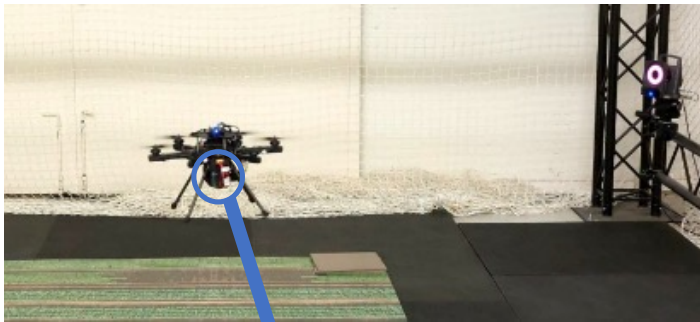
SOFIa results for head and tail wind



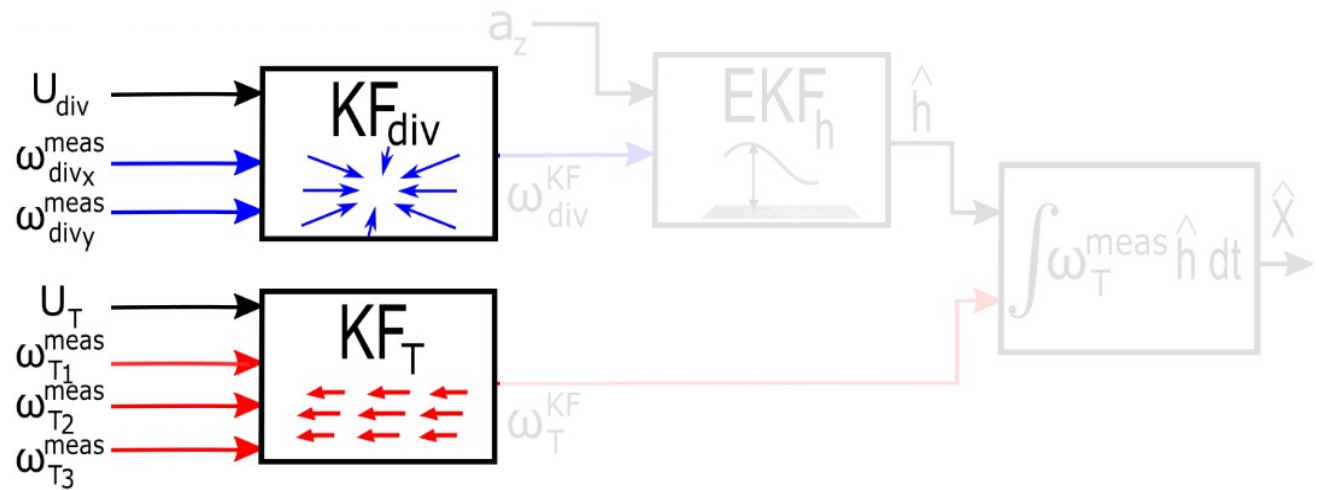
Reliability of the odometer in different winds



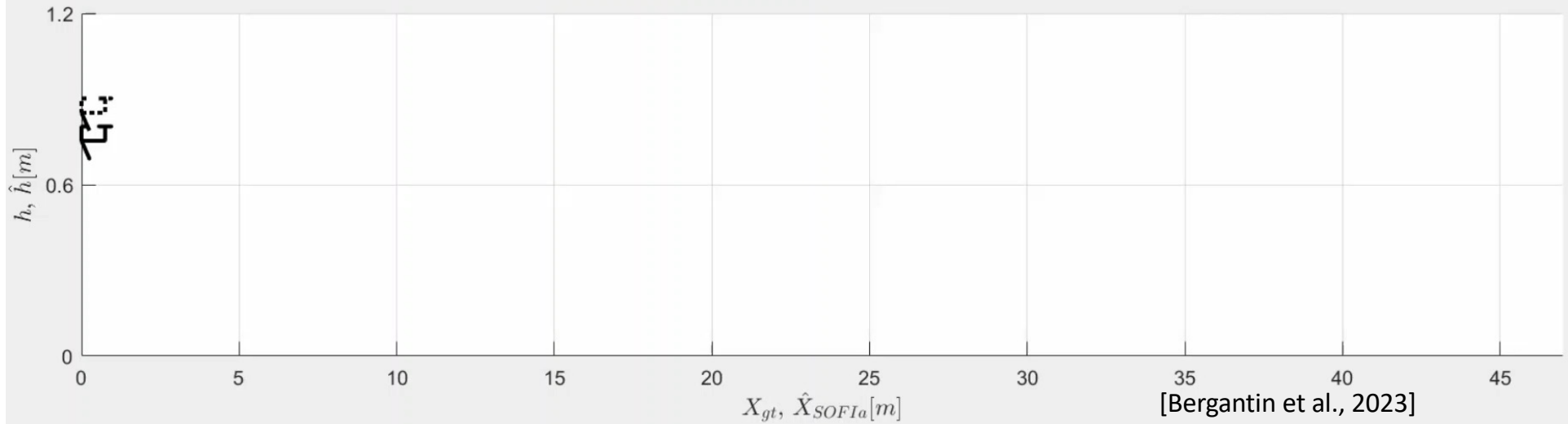
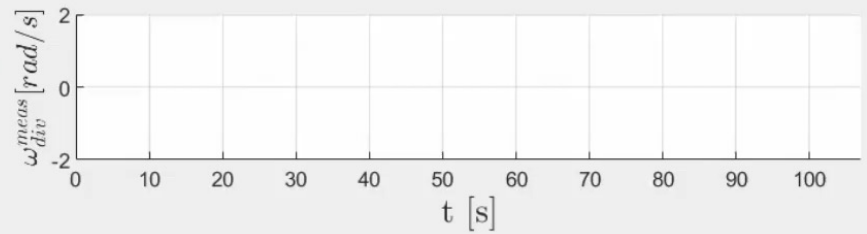
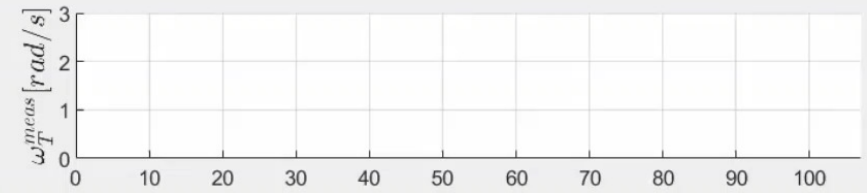
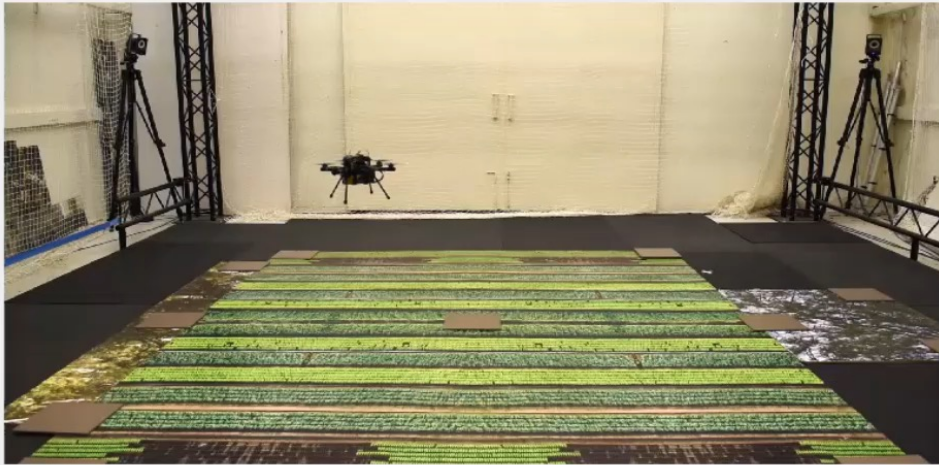
Fusion strategy with 4 optic flow sensors



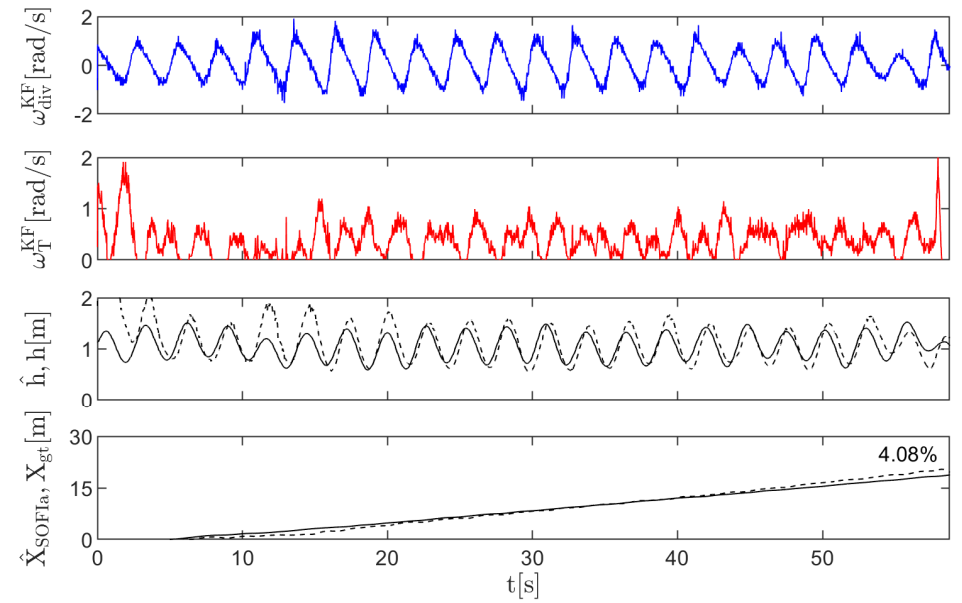
- 3 translational optic flow cues
- 2 optic flow divergence cues



SOFIa onboard a flying robot



Test outdoors



[Bergantin et al., 2023]

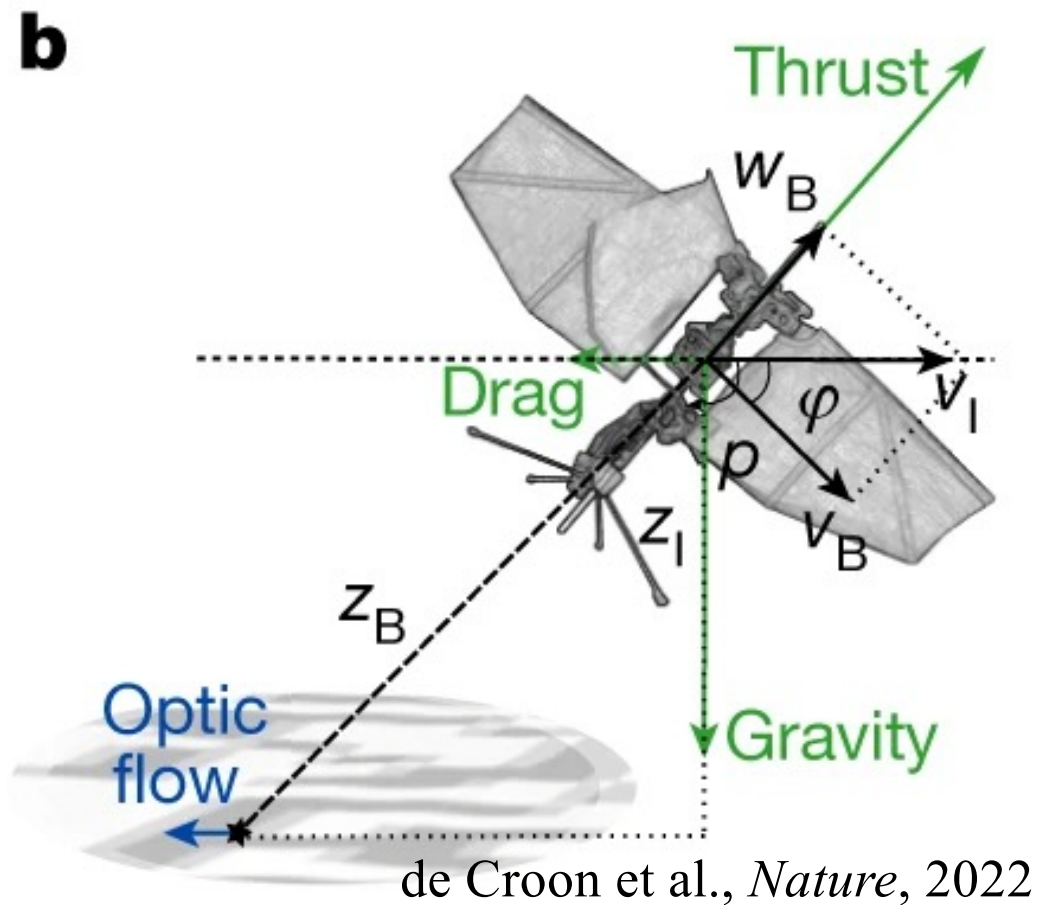
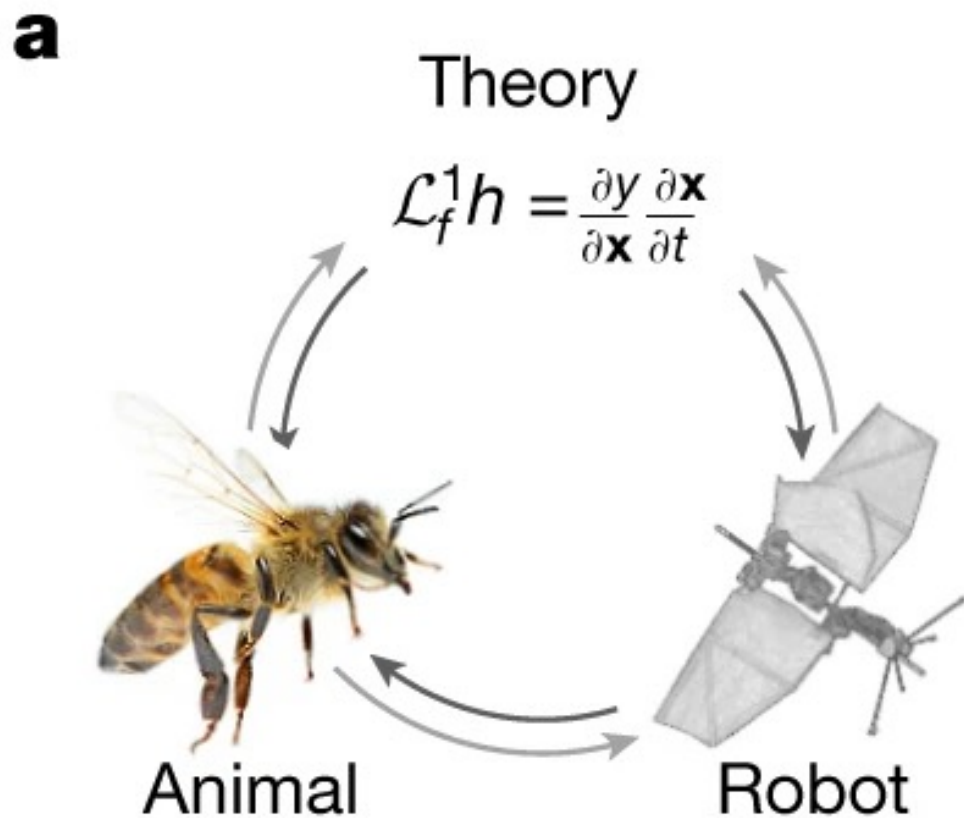
Embracing wobble to control attitude without IMU

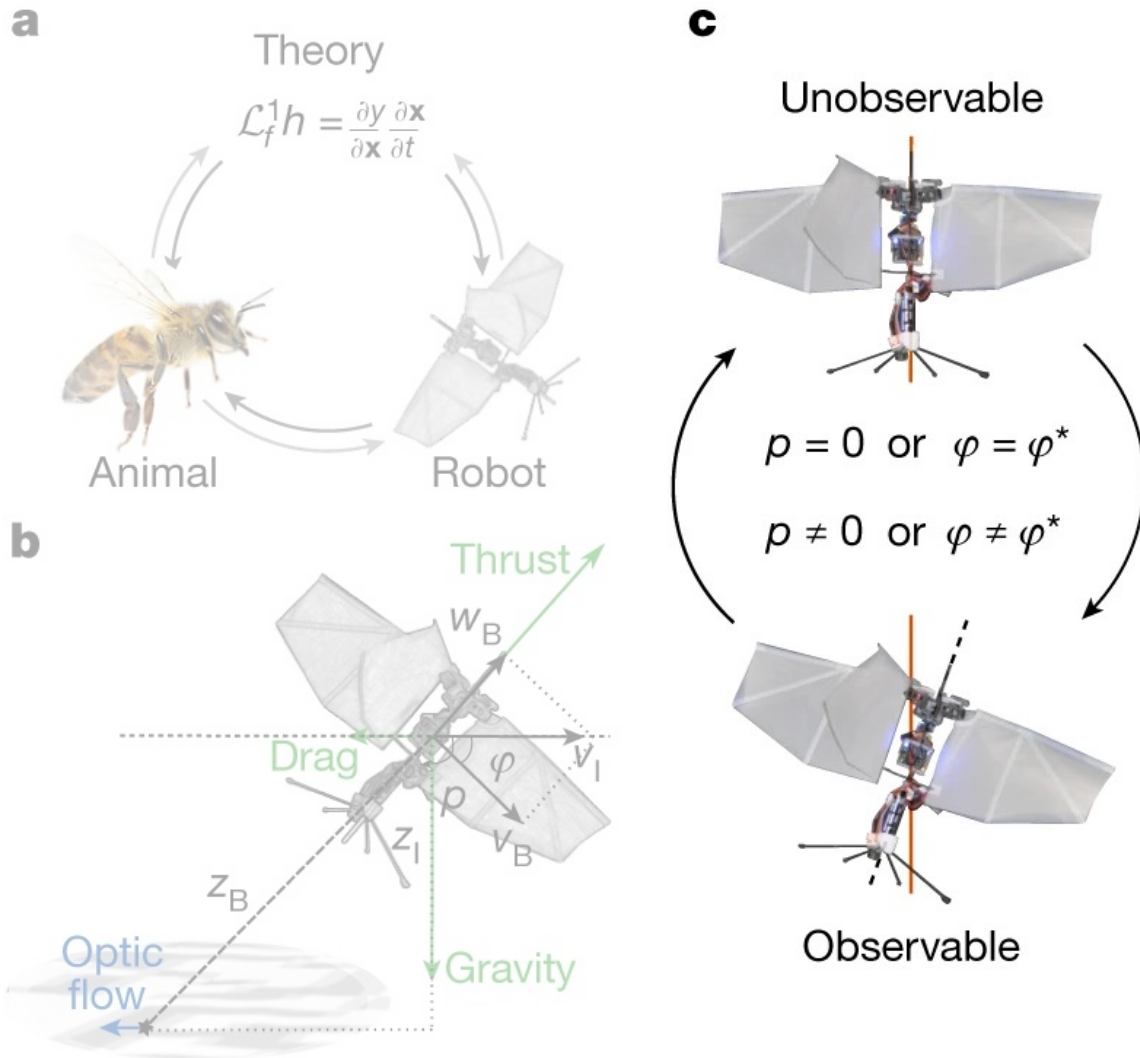
**G.C.H.E. de Croon¹, J.J.G. Dupeyroux¹, C. De Wagter¹,
A. Chatterjee¹, D.A. Olejnik¹, F. Ruffier²**

1. Micro Air Vehicle laboratory, Control and Simulation, Faculty of Aerospace Engineering, Delft University of Technology, the Netherlands

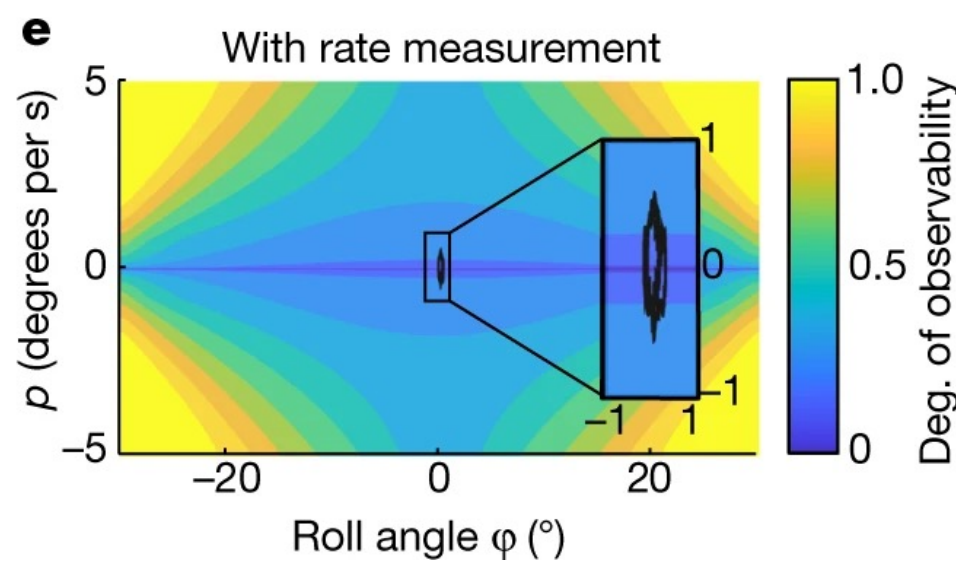
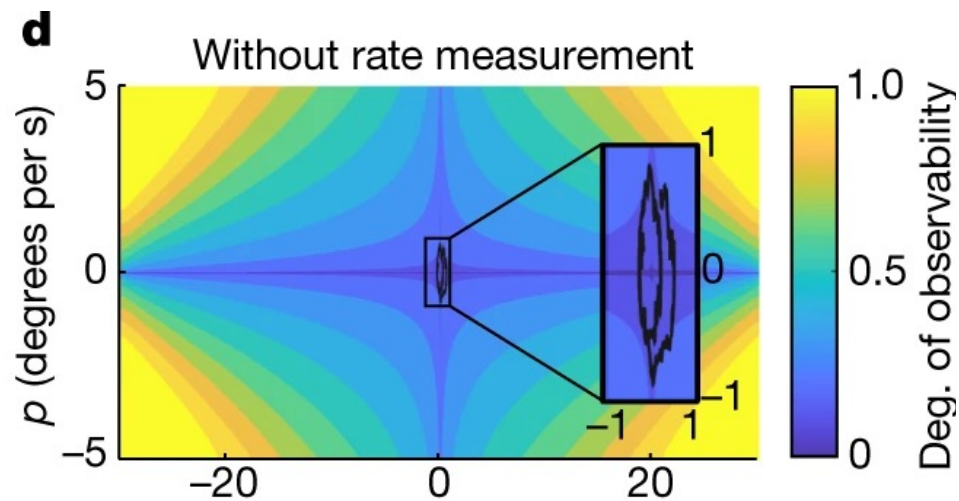
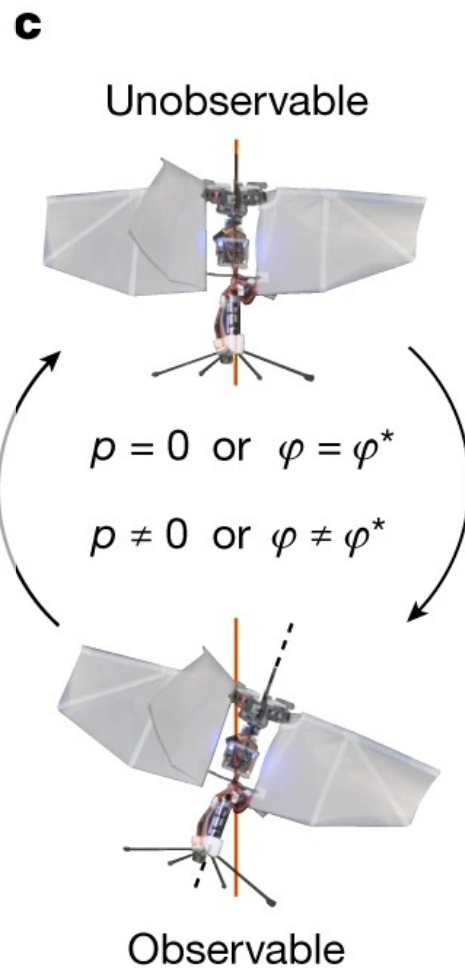
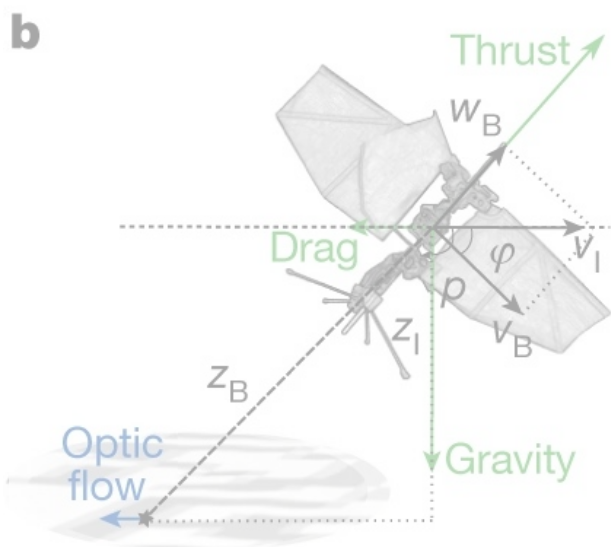
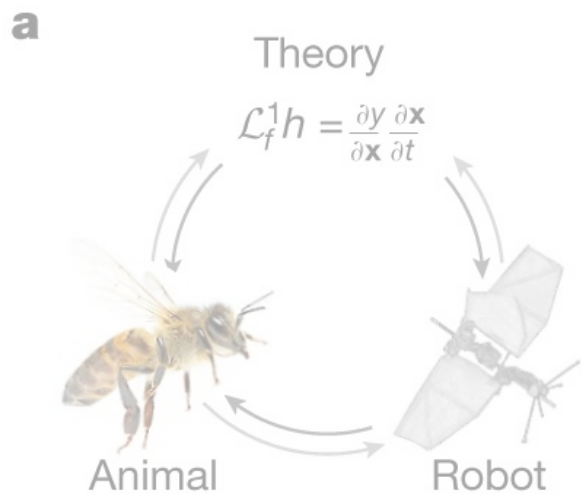
2. Aix Marseille Univ, CNRS, ISM, Marseille, France.





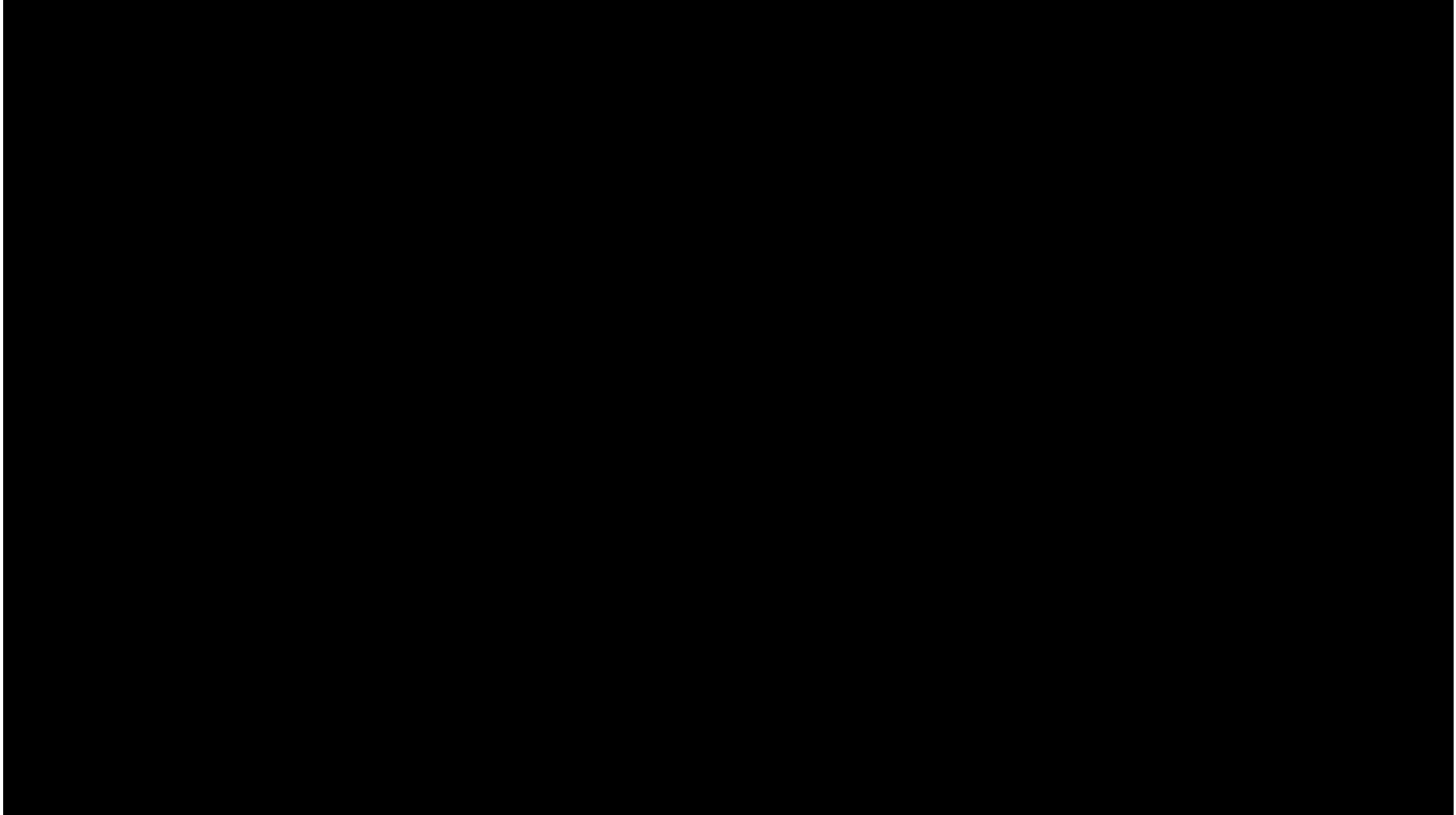


de Croon et al., *Nature*, 2022

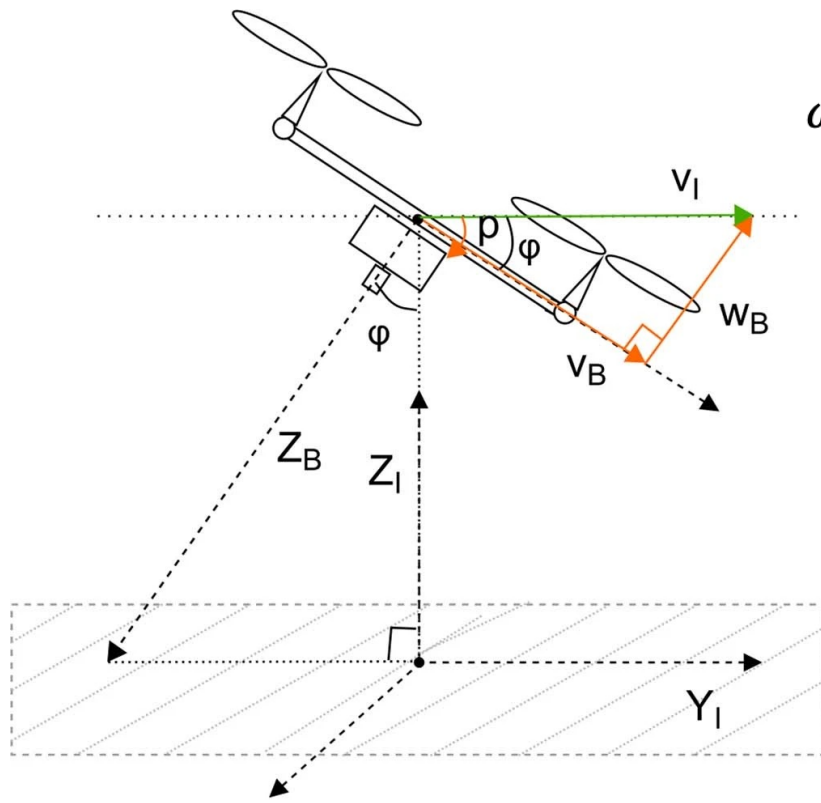


de Croon et al., *Nature*, 2022

Flapper drone flying with optic-flow-based attitude



de Croon et al., *Nature*, 2022



Observation model

$$\omega_y = -\frac{v_B}{Z_B} + p = -\frac{\cos^2(\varphi)v_I}{Z_I} + p \quad (\text{Eq. 1})$$

State update equation

$$f(\vec{x}, u) = \begin{bmatrix} \dot{v}_I \\ \dot{\varphi} \\ \dot{Z}_I \end{bmatrix} = \begin{bmatrix} g \tan(\varphi) \\ p \\ 0 \end{bmatrix} \quad (\text{Eq. 2})$$

Nonlinear observation analysis based on taking subsequent Lie derivatives $\mathcal{L}_f^i h$, which lead to an “observability mapping”

$$H(\vec{x}) = \begin{bmatrix} h \\ \mathcal{L}_f^1 h \\ \mathcal{L}_f^2 h \end{bmatrix} = \begin{bmatrix} y \\ \dot{y} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} -\frac{\cos^2(\varphi)v_I}{Z_I} + p \\ \frac{(2pv_I - g)\sin(2\varphi)}{2Z_I} \\ p \frac{2(pv_I - g)\cos(2\varphi) + g}{Z_I} \end{bmatrix} \quad (\text{Eq. 5})$$

$$y = \omega_y = h(\vec{x})$$

The common way to proceed with (nonlinear) observability analysis is then to perform a local analysis by **differentiating $H(\vec{x})$** with respect to the state to obtain the observability matrix \mathcal{O} :

$$\mathcal{O} = \frac{\partial H(\vec{x})}{\partial \vec{x}} = \begin{bmatrix} \frac{\partial y}{\partial \vec{x}} & \frac{\partial \dot{y}}{\partial \vec{x}} & \frac{\partial \ddot{y}}{\partial \vec{x}} \end{bmatrix} = \begin{bmatrix} \frac{\cos^2(\varphi)}{Z_I} & \frac{p \sin(2\varphi)}{Z_I} & \frac{2 p^2 \cos(2\varphi)}{Z_I} \\ \frac{\sin(2\varphi)v_I}{Z_I} & \frac{(2pv_I - g)\cos(2\varphi)}{Z_I} & 4p \frac{(g - pv_I)\sin(2\varphi)}{Z_I} \\ \frac{\cos^2(\varphi)v_I}{Z_I^2} & \frac{(g - 2pv_I)\sin(2\varphi)}{2Z_I^2} & -p \frac{2(pv_I - g)\cos(2\varphi) + g}{Z_I^2} \end{bmatrix}$$

(Eq. 6)

To **find all conditions in which the matrix \mathcal{O} is not full rank**, we solve for its determinant being zero. We find several solutions. Most are impossible physically.

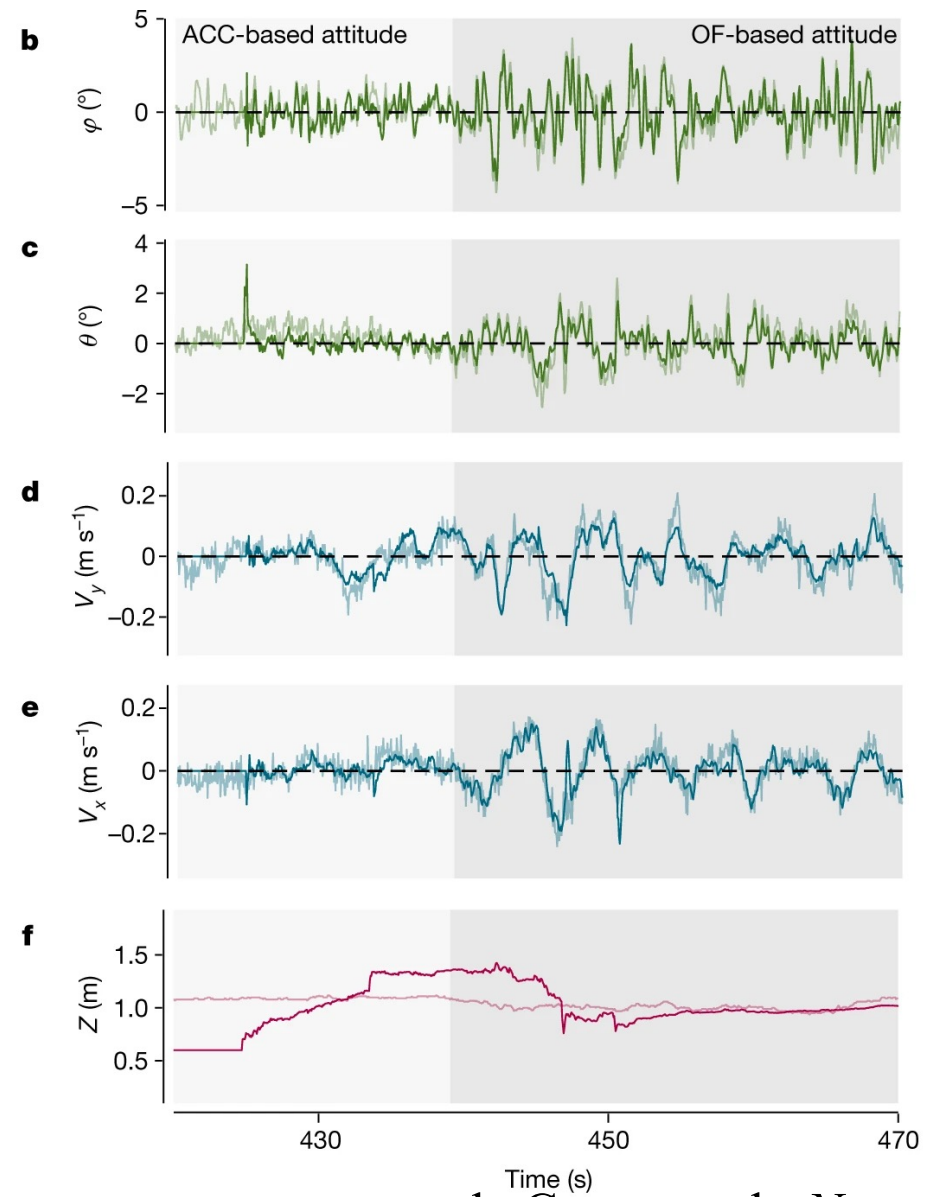
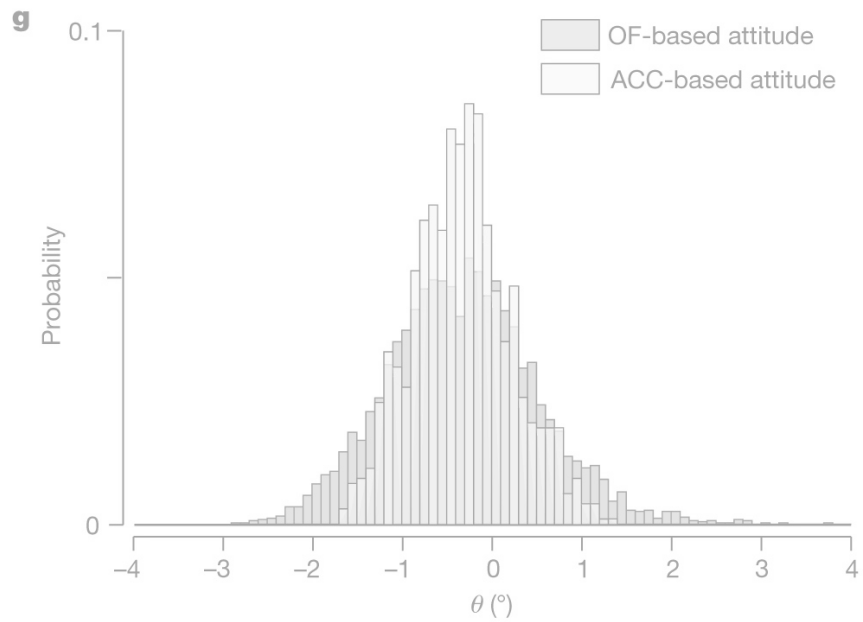
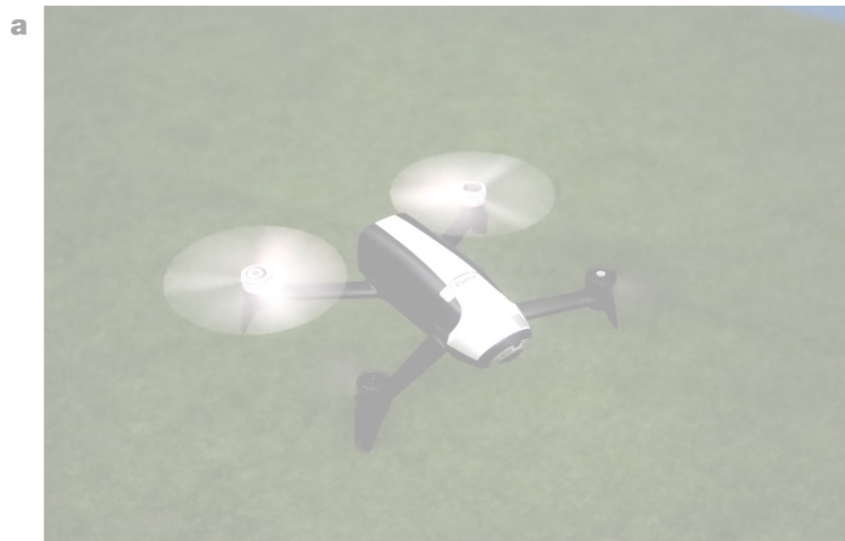
But, the state is not strictly locally observable, as $p = 0$ and $\varphi = 0$ lead to a range of solutions.

=> **Since p is a control input and can be set nonzero**, and $\varphi = 0$ will be a transient state for nonzero p , **the system is likely globally observable**.

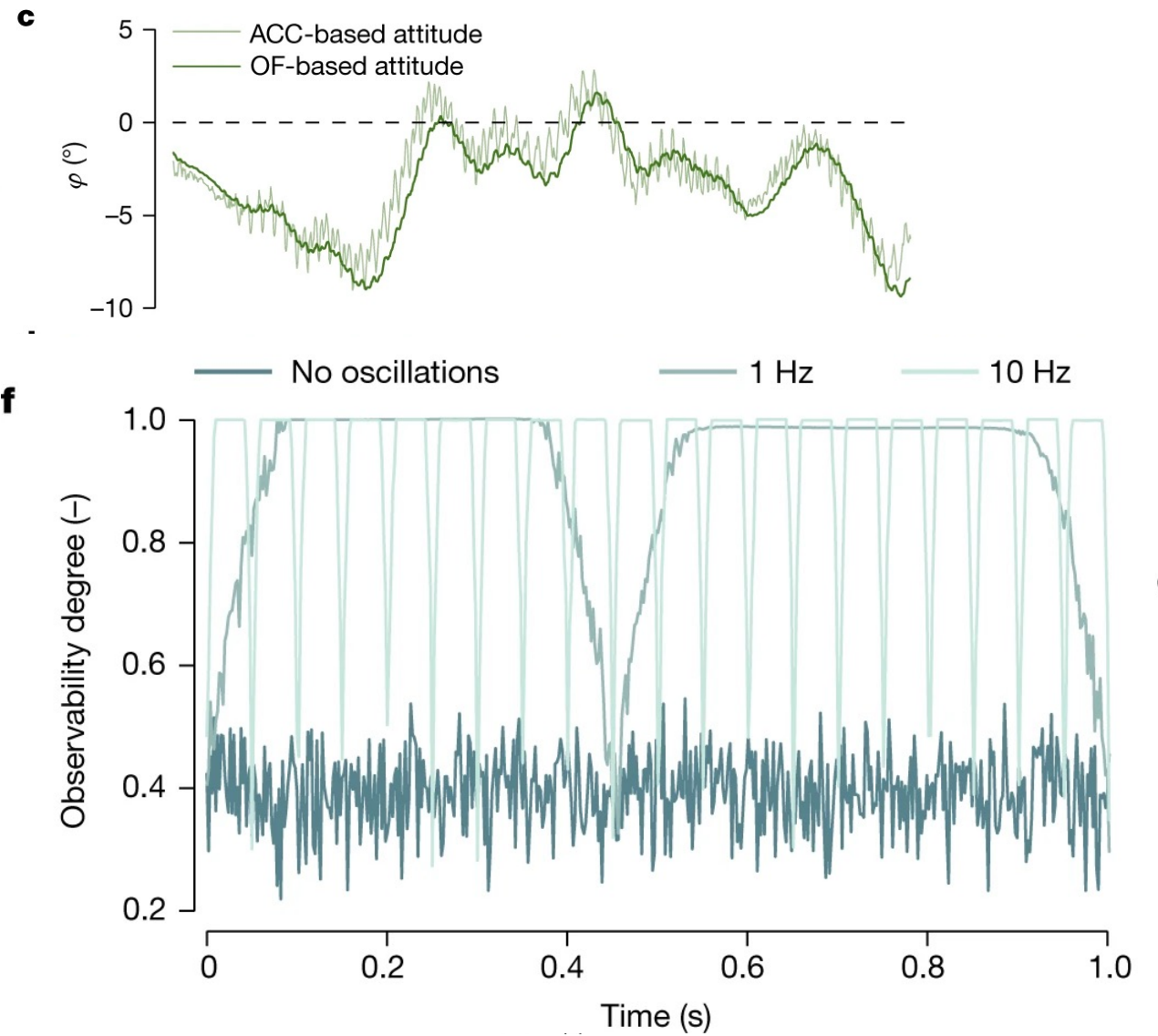
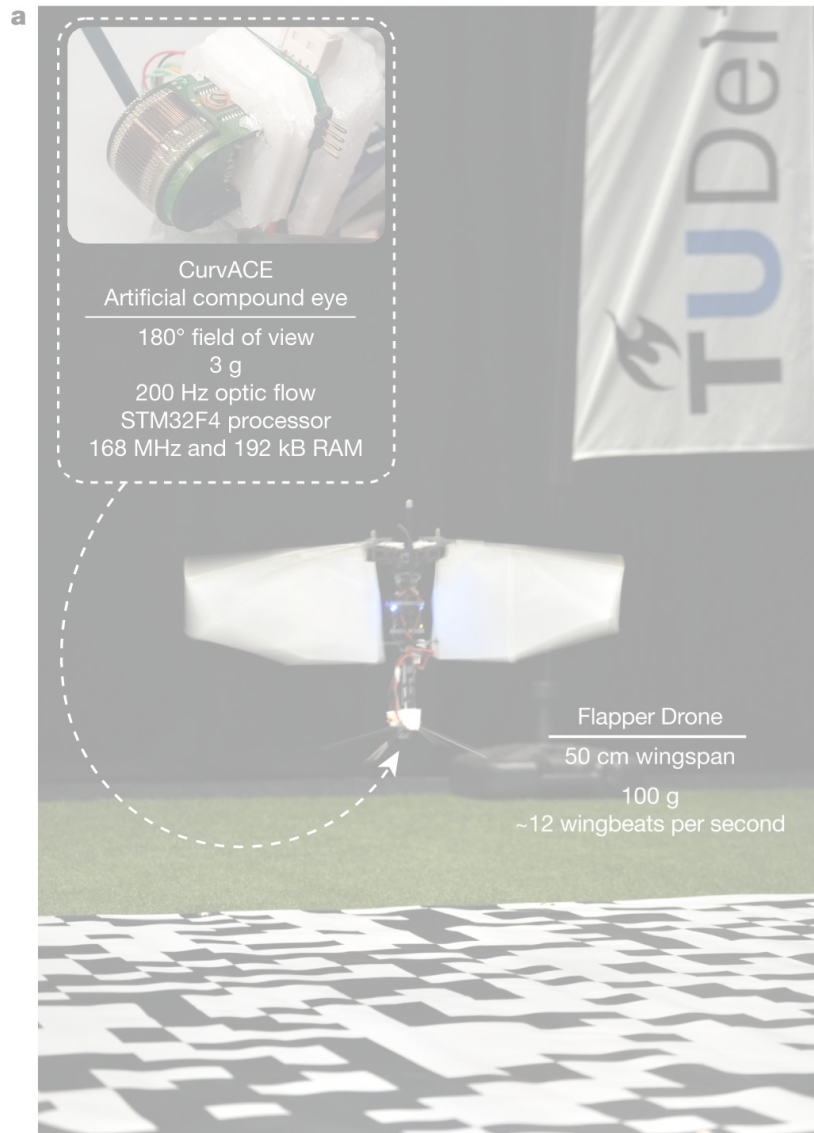
de Croon et al., *Nature*,

=> We have proved the stability of the feedback loop based on this estimate

2022



de Croon et al., *Nature*, 2022



de Croon et al., *Nature*, 2022



Many Thanks !!

Biorobotics Lab.

