Finanziato dall'Unione europea NextGenerationEU
Ministero dell'Università e della Ricerca
Italiadomani PIANO NAZIONALE DI RIPRESA E RESILIENZA
TOR VERGATA UNIVERSITÀ DEGLI STUDI DI ROMA

Learning how to stir oblate particles in complex flows

Luca Biferale
biferale@roma2.infn.it

Department of Physics & INFN, University of Rome Tor Vergata, 00133 Rome, Italy

acknowledgements

Dr. Lorenzo Piro
(Rome UTOV)

Dr. Jingran Qiu
(U. Gothenbhurg)

Dr. Massimo Cencini
(Rome - CNR)

Prof. Bernhard Mehlig
(U. Gothenbhurg)

Prof. Kristian Gustavsson
(U. Gothenbhurg)

erc

Finanziato dall'Unione europea NextGenerationEU

Ministero dell'Università e della Ricerca

Italiadomani
PIANO NAZIONALE DI RIPRESA E RESILIENZA

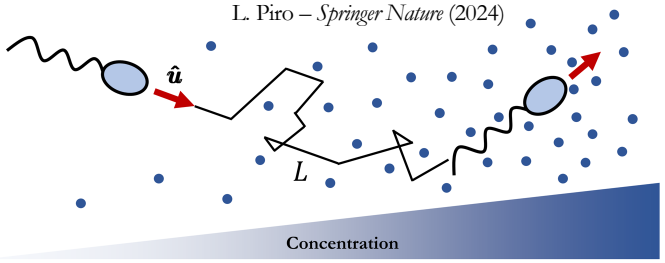
TOR VERGATA
UNIVERSITÀ DEGLI STUDI DI ROMA

INFN

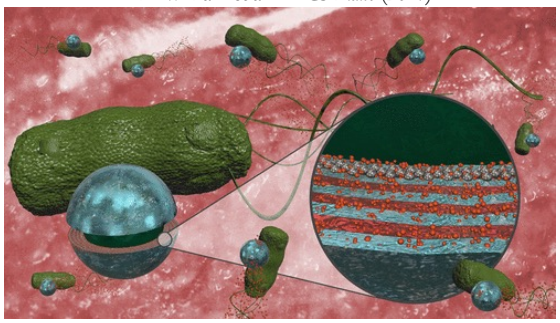
optimal navigation

- **Finding efficient path** is important:
  - Food tracking
  - Escape from harmful regions
- **To navigate** in a complex environment, microswimmers exploit **local cues**:
  - Flow velocity gradients
  - Light gradients (phototaxis)
  - Chemical concentration gradients (chemotaxis)
- **Applications** in the design of **artificial swimmers**:
  - Targeted delivery of drugs
  - Environmental monitoring
  - **Smart Lagrangian Control of Turbulence**

L. Piro – *Springer Nature* (2024)

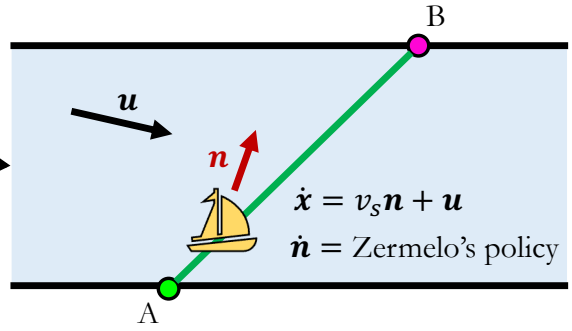


B.-W. Park et al. – *ACS Nano* (2017)



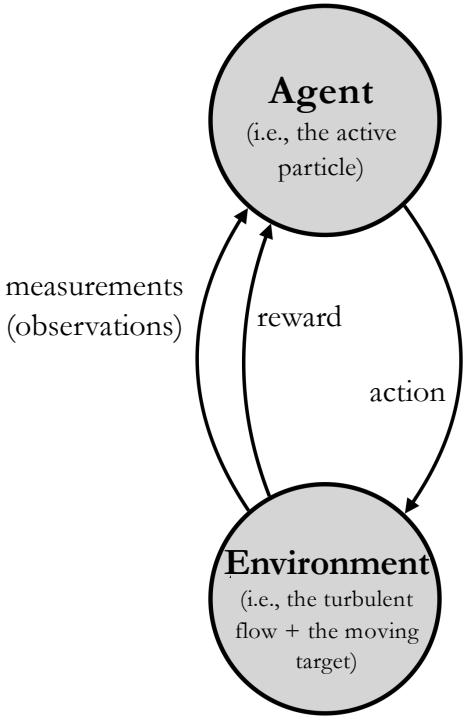
- **First formulation** of the problem by Zermelo (1931):

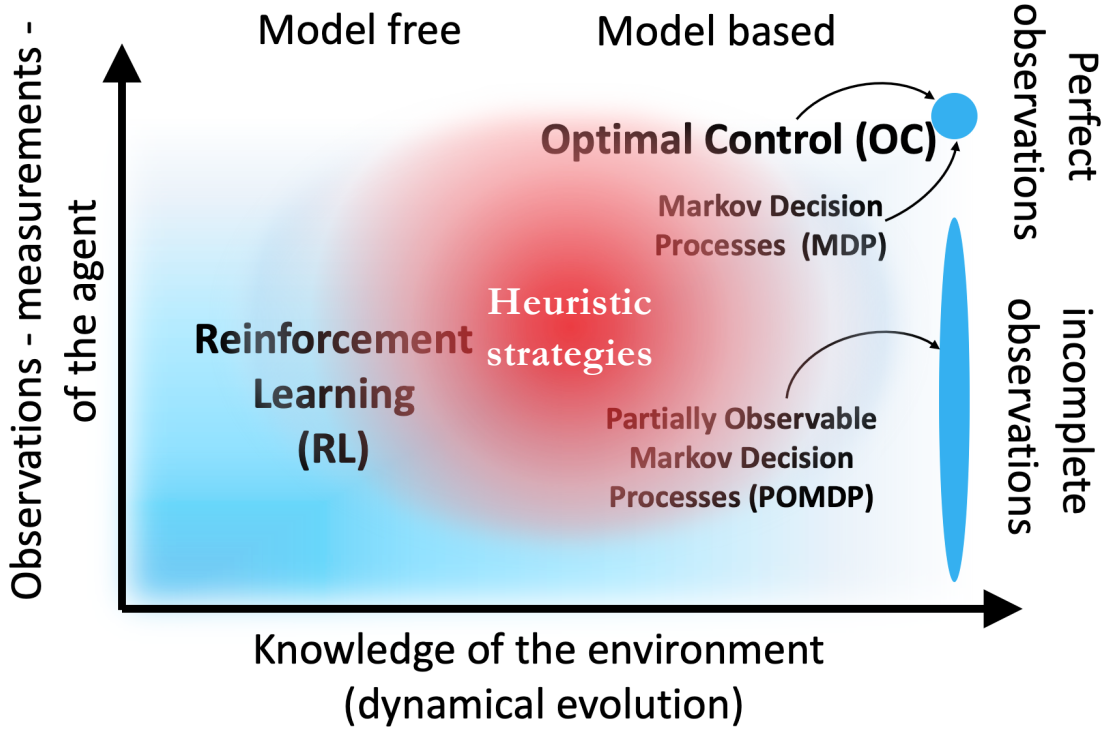
"How must a **ship be steered** to go from a **point to a given goal** in the **shortest time?"**



[1] H.C. Berg - *Springer Science* (2008)
[2] J. D. Wheeler et al. - *Annu. Rev. Cell Dev. Biol.* (2019)
[3] G. Jékely - *Philos. Trans. R. Soc. B* (2009)
[4] H. C. Berg, D. A. Brown - *Nature* (1972)
[5] A. C. H. Tsang et al. - *Adv. Intell. Syst.* (2020)
[6] E. Zermelo - *J. Applied Math. Mechanics* (1931)

Control Theory





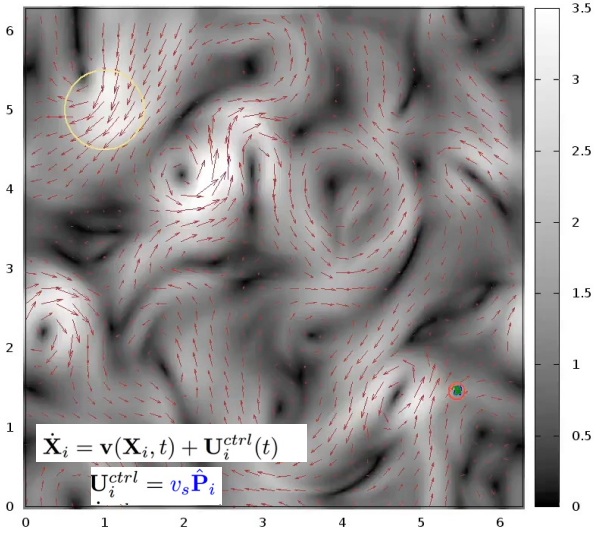
https://hdl.handle.net/20.500.11767/84114

Sutton, R. S., & Barto, A. G. **Reinforcement learning: An introduction**. MIT press (2018).

Bryson, A. E. and Ho, Y. **Applied optimal control: optimization, estimation and control** (New York: Routledge, 1975).

Examples of Optimal navigation problems in the literature

Zermelo's problem. Path planning for minimizing navigation time [1]

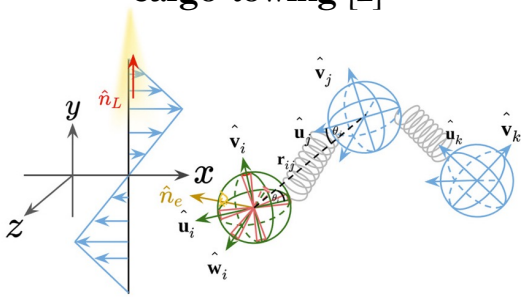


Reinforcement Learning (blue)

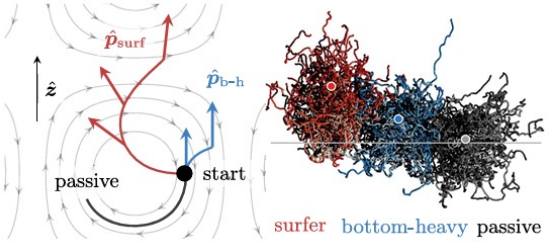
vs

Trivial Policy (green)

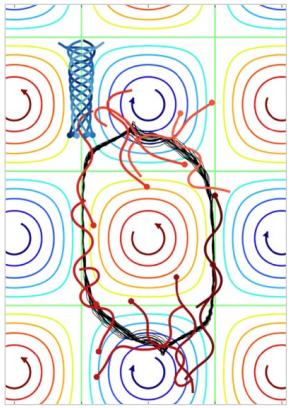
Optimal navigation for cargo-towing [2]



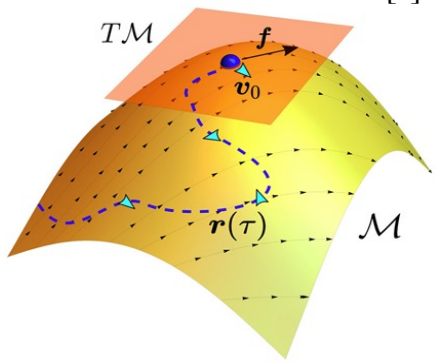
Path planning in 3d turbulent flows [5]



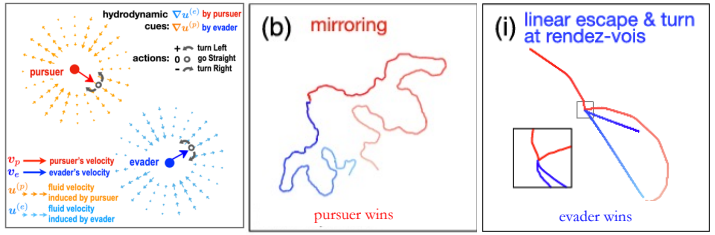
Optimal navigation for undulatory micro-swimmers [3]



Optimal navigation in curved manifolds [4]



Pursuing-escape strategies [6]



[1] Biferale et al. *Chaos* **29**, 103138 (2019)
[2] Sankaewtong et al. *Phys. Rev. Research* **6**, 033305 (2024)
[3] El Khiyati et al. *Eur. Phys. J. E*, **46**, 43 (2023)
[4] Piro et al. *Phys. Rev. Research* **3**, 023125 (2021)
[5] Monthiller et al. *Phys. Rev. Lett.* **129**, 064502 (2022)
[6] Borra et al. *Phys. Rev. Fluids* **7**, 023103 (2022)

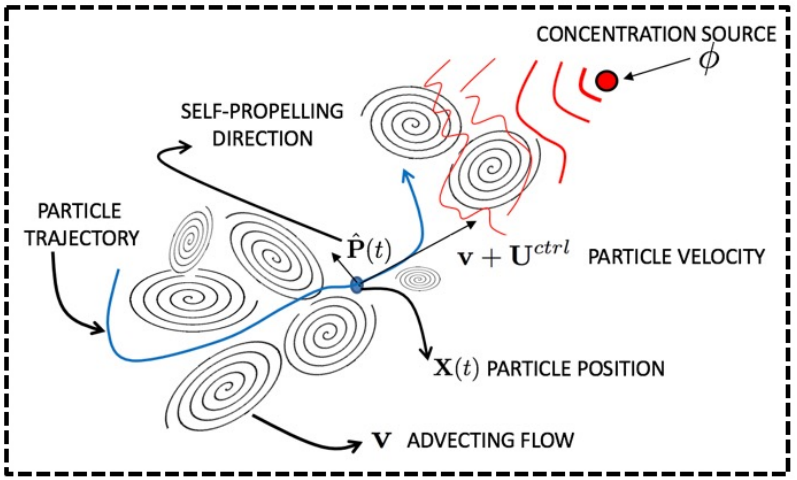
$$\begin{cases} \partial_t \mathbf{v} + \mathbf{v}\cdot\partial_{\mathbf{x}}\mathbf{v} + \partial_{\mathbf{x}}P - \nu\Delta\mathbf{v} = 2\mathbf{v}\times\boldsymbol{\Omega} + \mathcal{S}\mathbf{v} + \alpha g\hat{\mathbf{z}}T & (1) \\ \partial_t T + \mathbf{v}\cdot\partial_{\mathbf{x}}T - \chi\Delta T = \mathcal{G}v_z & (2) \\ \partial_t c + \mathbf{v}\cdot\partial_{\mathbf{x}}c - \chi_c\Delta c = \phi & (3) \end{cases}$$

EULERIAN

LAGRANGIAN

$$\dot{\mathbf{X}}_i = \mathbf{v}(\mathbf{X}_i, t) + \mathbf{U}_i^{ctrl}(t) \quad (4)$$

$$\begin{cases} \mathbf{U}_i^{ctrl} = v_s\hat{\mathbf{P}}_i & (5) \\ \dot{\mathbf{U}}_i^{ctlr} = -\mathbf{U}_i^{ctrl}/\tau + \beta D_t\mathbf{v} + g\beta\hat{\mathbf{z}} + \mathbf{f}^{ctrl} & (6) \end{cases}$$



ACTIONS: ONE-WAY CONTROL

$$\begin{cases} v_s & \to v_s[\mathbf{x}, c, \partial_{\mathbf{x}}c, \partial_{\mathbf{x}}\mathbf{v}, \dots] & (7) \\ \hat{\mathbf{P}}_i & \to \hat{\mathbf{P}}_i[\mathbf{x}, \mathbf{w}, T, c, \partial_{\mathbf{x}}c, \{\hat{\mathbf{P}}_j\}_1^{N_p}] & (8) \\ \beta = \frac{2(\rho_f-\rho)}{\rho_f+2\rho} & \to \frac{2(\rho_f-\rho[\mathbf{w},T,c,\dots])}{\rho_f+2\rho[\mathbf{w},T,c,\dots]} & (9) \\ \tau = \frac{r^2}{3\nu\beta} & \to \frac{r^2[\mathbf{w},T,c,\dots]}{3\nu\beta[\mathbf{w},T,c,\dots]} & (10) \end{cases}$$

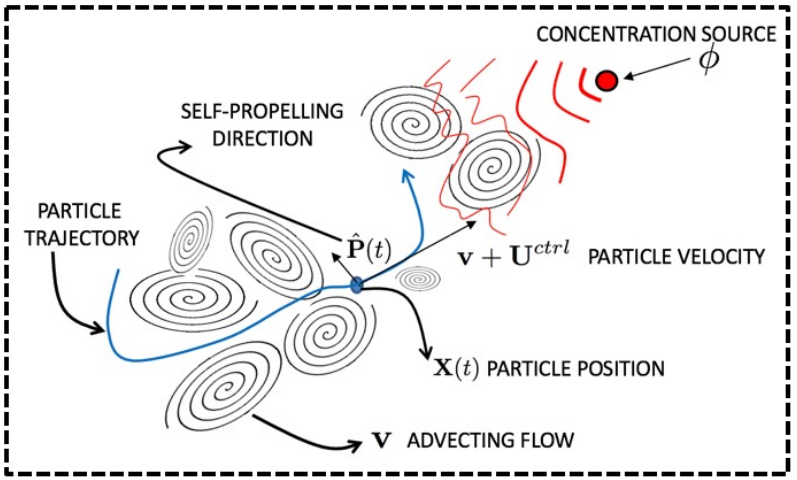
$$\begin{cases} \partial_t \mathbf{v} + \mathbf{v}\cdot\partial_{\mathbf{x}}\mathbf{v} + \partial_{\mathbf{x}}P - \nu\Delta\mathbf{v} = 2\mathbf{v}\times\boldsymbol{\Omega} + \mathcal{S}\mathbf{v} + \alpha g\hat{\mathbf{z}}T + \sum_{i=1}^{N_p}\delta(\mathbf{x}-\mathbf{X}_i(t))\mathcal{F} & (1) \\ \partial_t T + \mathbf{v}\cdot\partial_{\mathbf{x}}T - \chi\Delta T = \mathcal{G}v_z + \sum_{i=1}^{N_p}\delta(\mathbf{x}-\mathbf{X}_i(t))\mathcal{L} & (2) \\ \partial_t c + \mathbf{v}\cdot\partial_{\mathbf{x}}c - \chi_c\Delta c = \phi & (3) \end{cases}$$

EULERIAN

LAGRANGIAN

$$\dot{\mathbf{X}}_i = \mathbf{v}(\mathbf{X}_i,t) + \mathbf{U}_i^{ctrl}(t) \quad (4)$$

$$\begin{cases} \mathbf{U}_i^{ctrl} = v_s\hat{\mathbf{P}}_i & (5) \\ \dot{\mathbf{U}}_i^{ctlr} = -\mathbf{U}_i^{ctrl}/\tau + \beta D_t\mathbf{v} + g\beta\hat{\mathbf{z}} + \mathbf{f}^{ctrl} & (6) \end{cases}$$



ACTIONS: ONE-WAY CONTROL

$$\begin{cases} v_s & \to v_s[\mathbf{x},c,\partial_{\mathbf{x}}c,\partial_{\mathbf{x}}\mathbf{v},\dots] & (7) \\ \hat{\mathbf{P}}_i & \to \hat{\mathbf{P}}_i[\mathbf{x},\mathbf{w},T,c,\partial_{\mathbf{x}}c,\{\hat{\mathbf{P}}_j\}_1^{N_p}] & (8) \\ \beta = \frac{2(\rho_f-\rho)}{\rho_f+2\rho} & \to \frac{2(\rho_f-\rho[\mathbf{w},T,c,\dots])}{\rho_f+2\rho[\mathbf{w},T,c,\dots]} & (9) \\ \tau = \frac{r^2}{3\nu\beta} & \to \frac{r^2[\mathbf{w},T,c,\dots]}{3\nu\beta[\mathbf{w},T,c,\dots]} & (10) \end{cases}$$

ACTIONS: TWO-WAY CONTROL

$$\begin{cases} \mathcal{F} \to \mathcal{F}[\mathbf{w},\epsilon,\mathcal{H},\dots] & (11) \\ \mathcal{L} \to \mathcal{L}[c,T,v_z,\dots] & (12) \end{cases}$$

THE CASE OF ISOTROPIC HELICOIDS

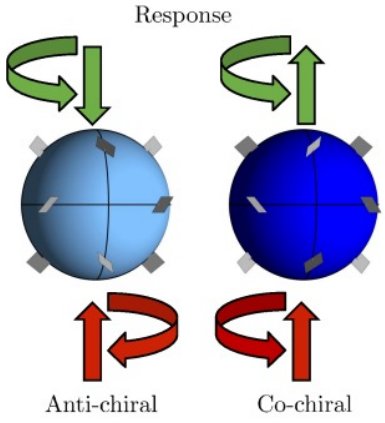
$$\dot{\boldsymbol{v}} = \frac{1}{\tau_\mathrm{p}}\left[\boldsymbol{u}(\boldsymbol{x}(t),t) - \boldsymbol{v} + \frac{2\tilde{a}}{9}C_0(\boldsymbol{\Omega}(\boldsymbol{x}(t),t) - \boldsymbol{\omega})\right]$$

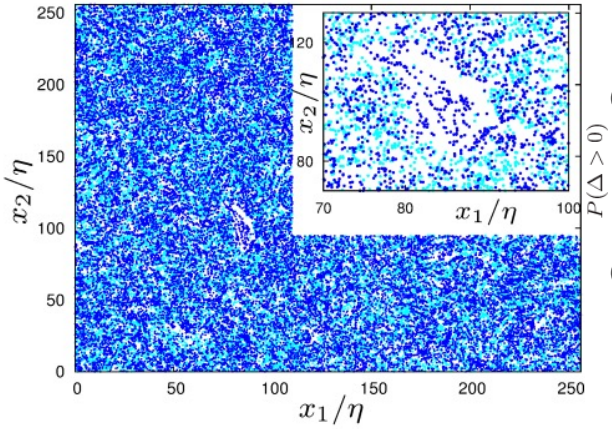
$$\dot{\boldsymbol{\omega}} = \frac{1}{\tau_\mathrm{p}}\left[\frac{10}{3}S(\boldsymbol{\Omega}(\boldsymbol{x}(t),t) - \boldsymbol{\omega}) + \frac{5}{9\tilde{a}}C_0(\boldsymbol{u}(\boldsymbol{x}(t),t) - \boldsymbol{v})\right].$$

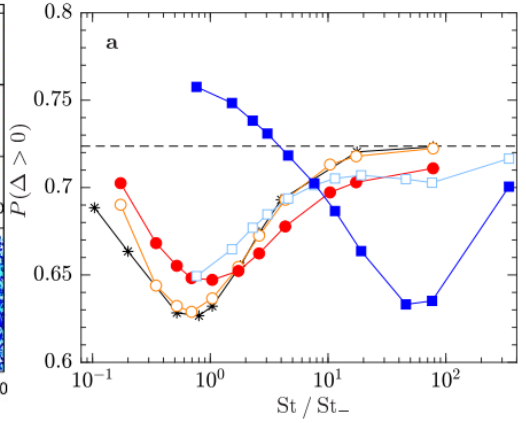
$$\begin{pmatrix}\dot{v}_i \\ \dot{\omega}_i\end{pmatrix} = \mathbb{D}\begin{pmatrix}u_i - v_i \\ \Omega_i - \omega_i\end{pmatrix}, \qquad \mathbb{D} = \frac{1}{\mathrm{St}}\begin{pmatrix}1 & \frac{2\bar{a}C_0}{9} \\ \frac{5C_0}{9\bar{a}} & \frac{10}{3}\end{pmatrix}$$

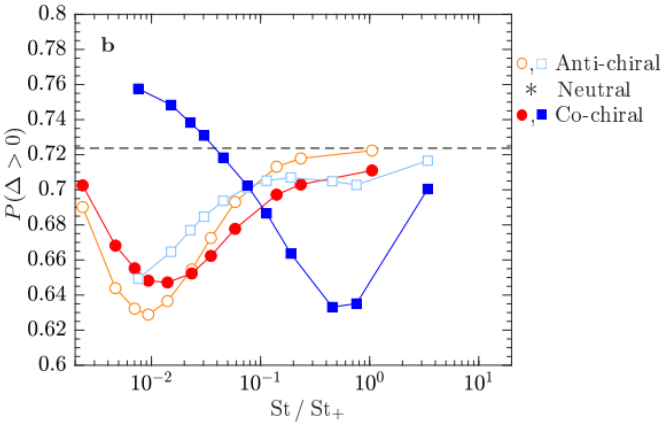
$$\boldsymbol{\nabla}\cdot\boldsymbol{v} \sim -\frac{\mathrm{St}}{27S - C_0^2}\left(27S\mathrm{Tr}[\boldsymbol{A}^2] - \frac{9aC_0}{5}\mathrm{Tr}[\boldsymbol{AV}]\right) + o(\mathrm{St}),$$

$$A_{ij} = \partial_j u_i \text{ and } V_{ij} = \partial_j \Omega_i.$$







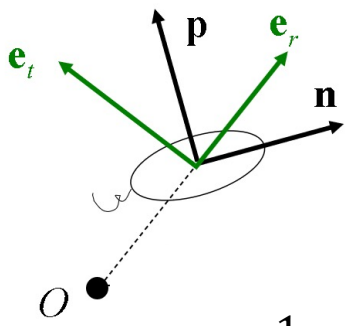


Lord Kelvin, Phil. Mag. 42 (1871) 362
K. Gustavsson and L.B. Physical Review Fluids , vol. 1, 054201 (2016)
L.B. , K. Gustavsson and R. Scatamacchia Journal of Fluid Mechanics 869, 646-673 (2019)

Shape ‘strongly’ affects navigation performance

(J. Qiu, L Piro, L. Biferale, M. Cencini, B. Mehlig, K. Gustavsson unpublisehd)

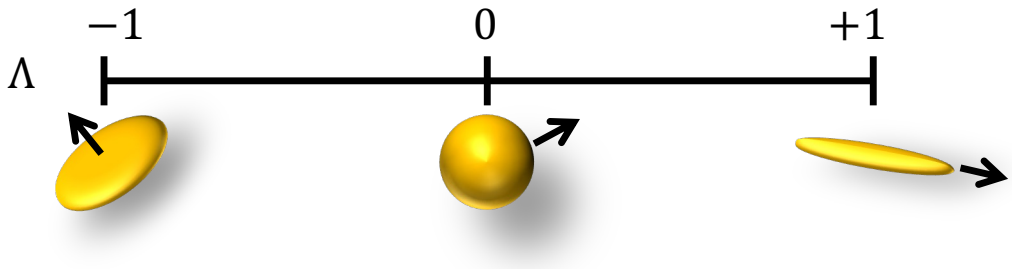
- Swimmers with definite size and shape ⟶ **Flow-induced** torque[7]
- Axisymmetric **microswimmers dynamics** at low Re:

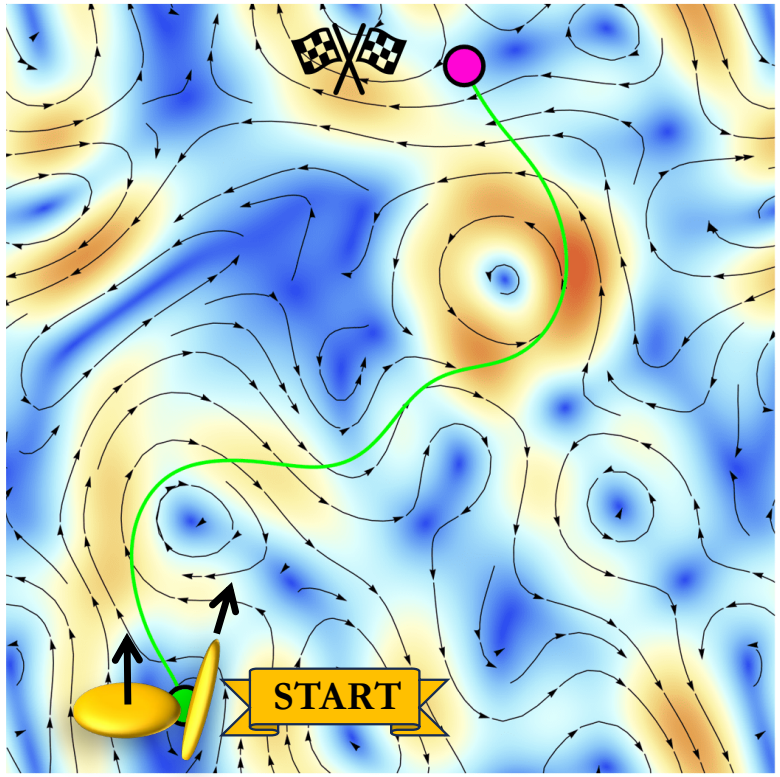


Swimming speed

$$\begin{cases} \dot{\boldsymbol{x}} = v_s \boldsymbol{n} + \boldsymbol{u} \\ \dot{\boldsymbol{n}} = \frac{1}{2}\boldsymbol{\omega}\times\boldsymbol{n} + \Lambda\, \boldsymbol{p}\cdot\mathbb{S}\cdot\boldsymbol{n} \equiv \boldsymbol{\omega}_{\mathbf{f}}(\Lambda) \end{cases}$$

Flow vorticity | **Shape factor** | Flow strain





- **Exact** result[8]: $\boldsymbol{\omega}_{\mathbf{f}}(\Lambda=-1)$ ⟺ Zermelo’s policy ⟹ **Disk-like** swimmers follow **Zermelo’s path**

[7] G. B. Jeffery - *Proc. R. Soc. London A* (1922)
[8] L. Piro et al. – *Phys. Rev. Research* (2024)

Problem: maximize single-particle diffusion in turbulence

- **Stochastic flow model** (2D) [9]: $\boldsymbol{u}(\boldsymbol{x},t) = (\partial_y \Psi, -\partial_x \Psi)$, where

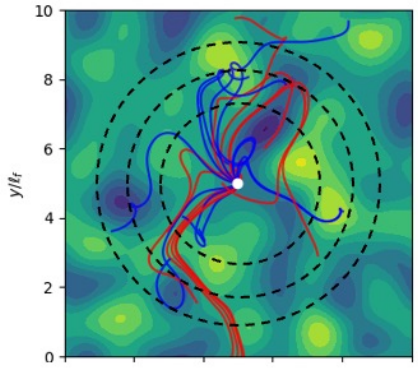
$$\begin{cases} \Psi(x,t) = \sum_{i=0}^{N} c_i(t)\Psi_i(\mathrm{x}) \\ \langle \Psi_i(\boldsymbol{x}), \Psi_j(\boldsymbol{x}') \rangle = \delta_{ij} u_{\mathrm{rms}}^2 \ell_f^2 \exp\left[-\frac{(\boldsymbol{x}-\boldsymbol{x}')^2}{2\ell_f^2} \right] \\ \langle c_i(t), c_j(t') \rangle = \frac{\delta_{ij}}{N} \exp\left[-\frac{|t-t'|}{\tau_f} \right] \leftarrow \text{Ornstein-Uhlenbeck process} \end{cases}$$

➡ Flow dynamics identified by single **dimensionless parameter**:

$$\boxed{\mathrm{Ku} = \frac{u_{\mathrm{rms}} \tau_f}{\ell_f}}$$

[Kubo number]

➡ **Slower flow** evolution at **larger Ku**
(Ku → ∞ : time-independent flow)

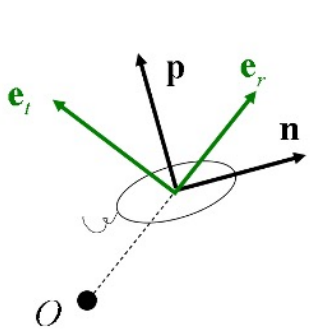


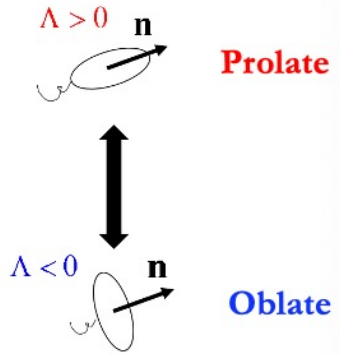
- **Microswimmer model** (self-propelled spheroid):

Dimensionless swimming speed = $v_s/u_{\mathrm{rms}} = 1/2$

$$\begin{cases} \dot{\boldsymbol{x}} = \Phi_s \boldsymbol{n} + \boldsymbol{u} \\ \dot{\boldsymbol{n}} = \left[\frac{1}{2}\boldsymbol{\omega} \times \boldsymbol{n} + \Lambda(\mathrm{t})\, \boldsymbol{p} \cdot \mathbb{S} \cdot \boldsymbol{n} \right] \xrightarrow{\text{2D}} \dot{\theta} = \left[\frac{1}{2}\omega + \Lambda(\mathrm{t})\, S_{\mathrm{np}} \right] \end{cases}$$

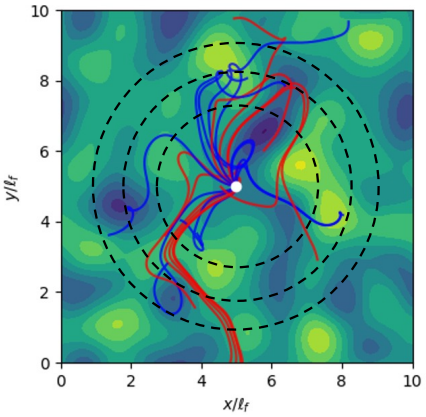
Flow vorticity — Flow strain — **CONTROL**





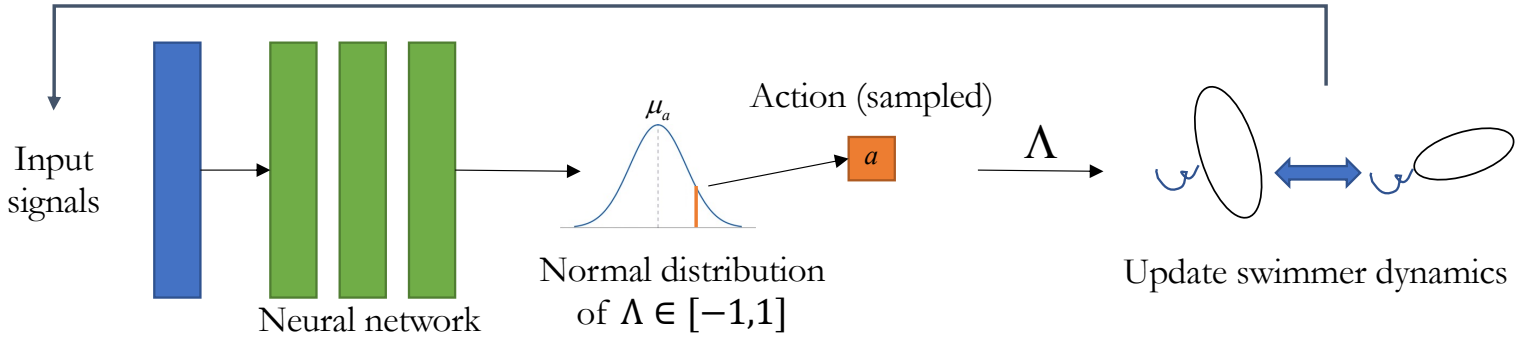
[9] J. Bec et al. - *Annu. Rev. Fluid Mech.* (2024)

Reinforcement Learning setup



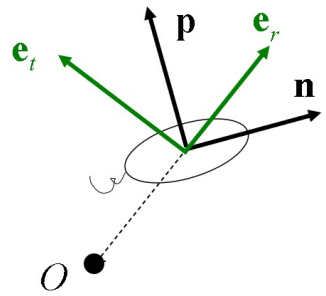
Goal → **Maximize distance** from origin within a time T [**optimal escape** problem]

- Proximal Policy Optimization (**PPO**) algorithm:



- **Random initial** position and orientation.
- **Reward = distance from origin within time *T*.**
- Continuous **input signals**: n_r, n_t, S_{nn}, S_{np}, ω.

Swimmer's **orientation** | **Flow gradients** components



Performance comparison across flow regimes

- **Baseline strategies:**

A. Constant-shape swimmers (Naïve):

- Empirically checked that elongated swimmers perform best:

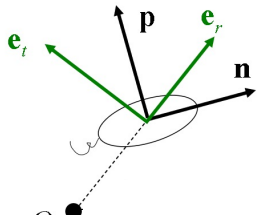
$$\Lambda = \Lambda_{\max} = 0.98$$

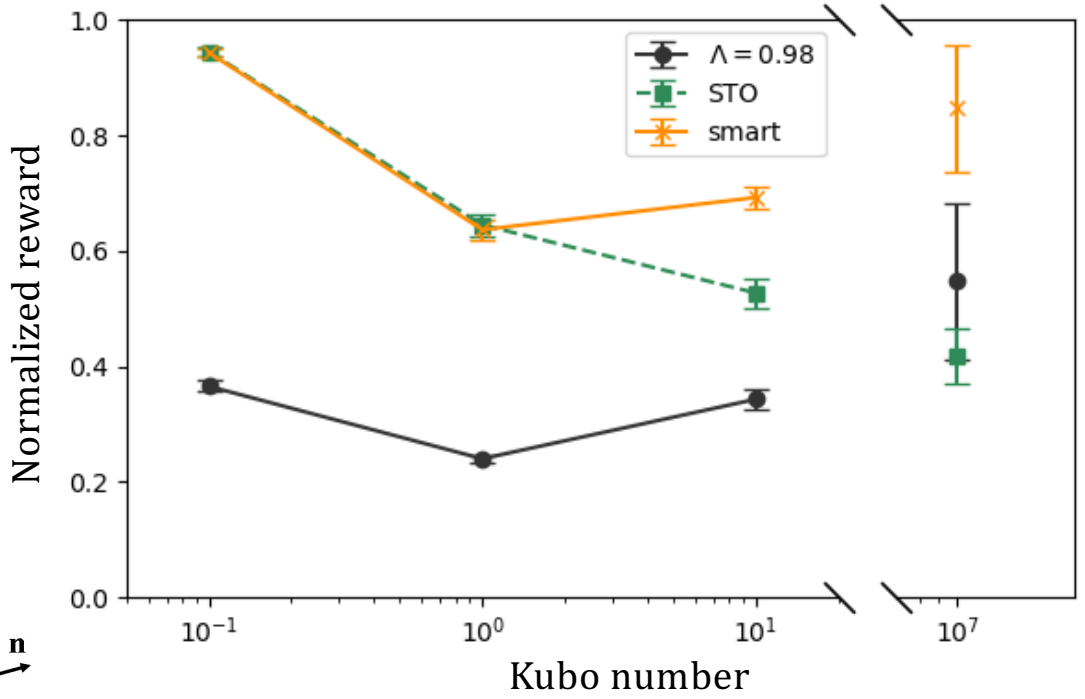
(aspect ratio $= 10$)

B. Short-time optimization (STO) strategy:

- **Underlying mechanism**: choose Λ so that swimmer tends to align to $\boldsymbol{e}_r$
- Obtained by maximizing distance in **short time limit**:

$$\Lambda(t) = -\Lambda_{\max}\text{sign}(n_t)\text{sign}(S_{np})$$

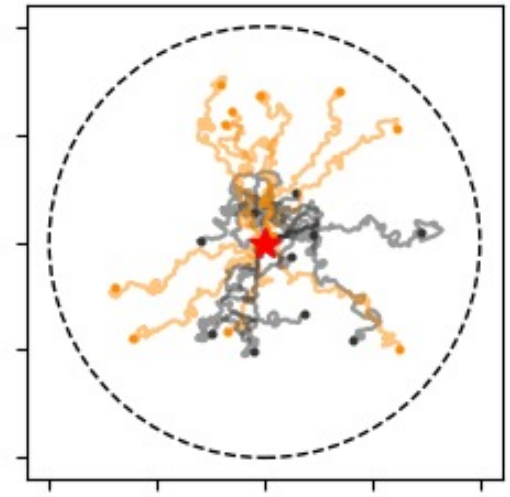




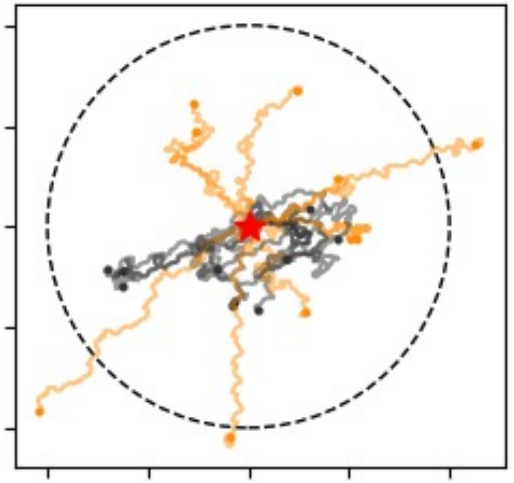
(J. Qiu, L Piro, L. Biferale, M. Cencini, B. Mehlig, K. Gustavsson 'Optimal diffusion by shape adaptation in turbulence', in preparation 2026)

Trajectories

Black: naïve swimmers ($\Lambda = 0.98$)
Yellow: smart swimmers (RL)
Red star: initial position
Black dashed circle: maximal distance that a swimmer can travel without the flow
Time horizon: $T = 200\tau_K$

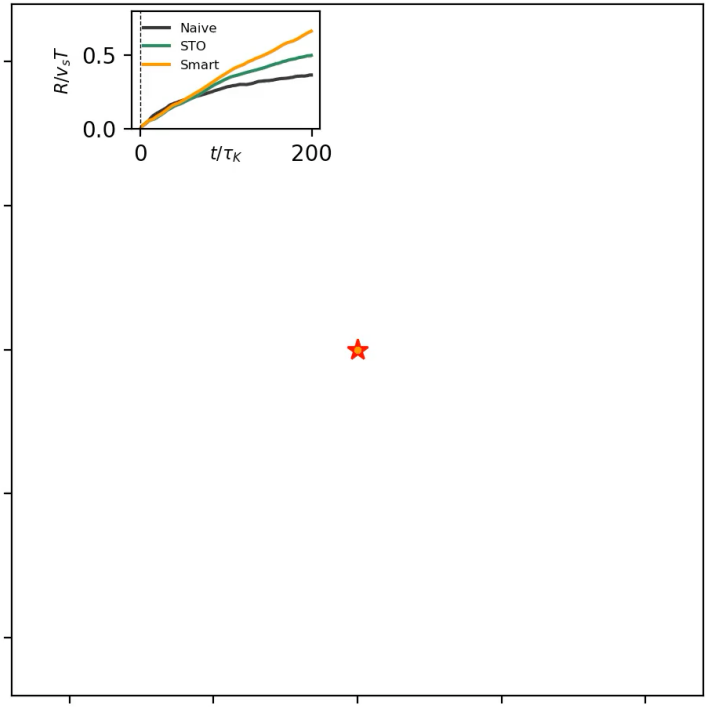


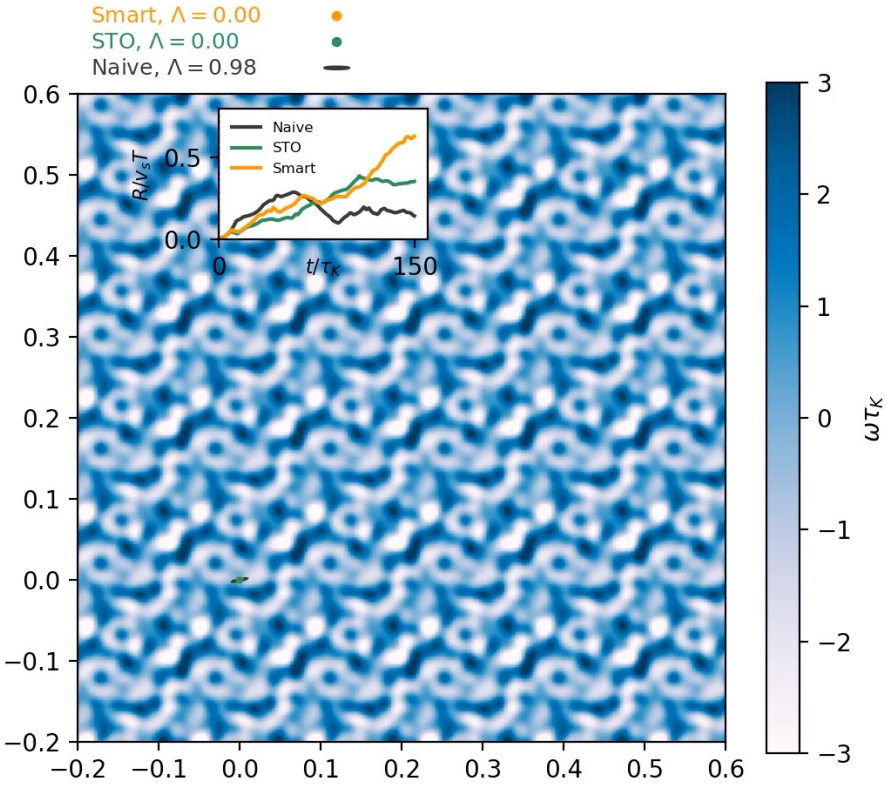
Ku=10



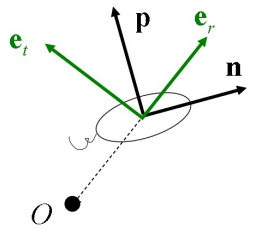
Ku=10^7

Performance comparison: Ku = 10



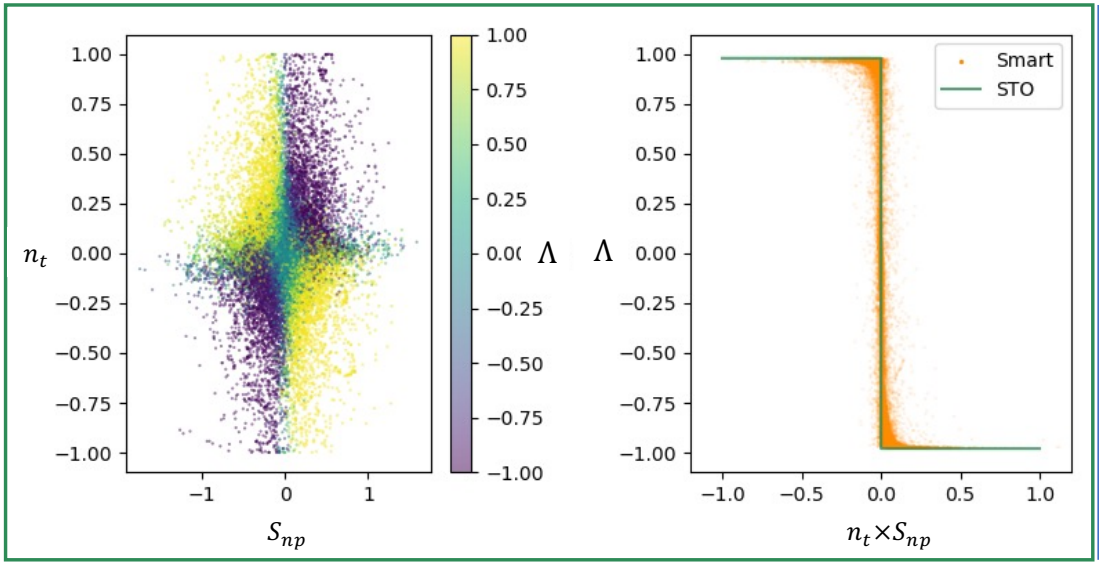


Learned strategies across flow regimes



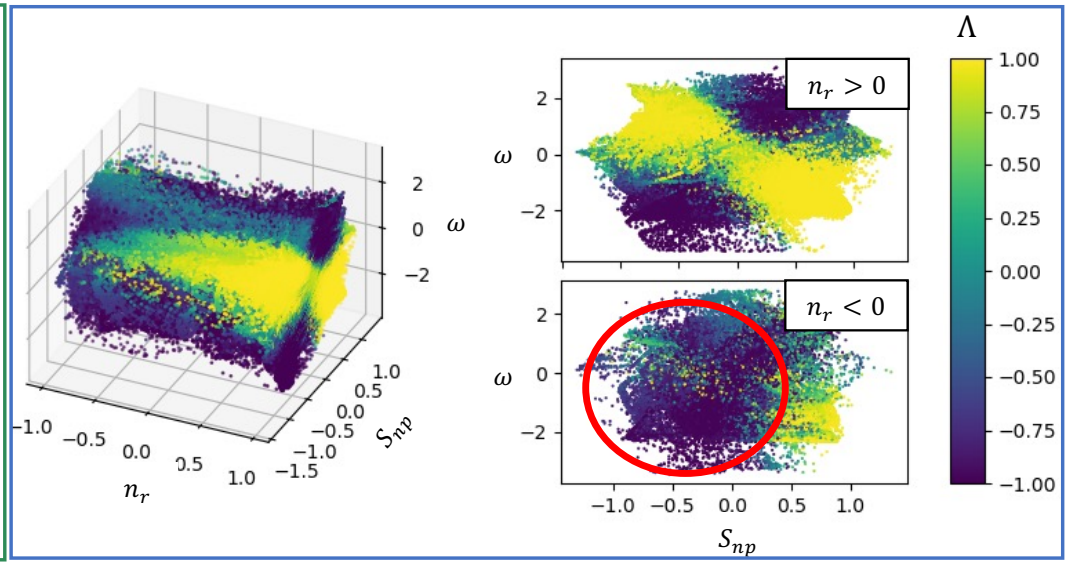
Small Ku

Smart swimmer **recovers STO strategy**
(no long-time planning required)



Large Ku

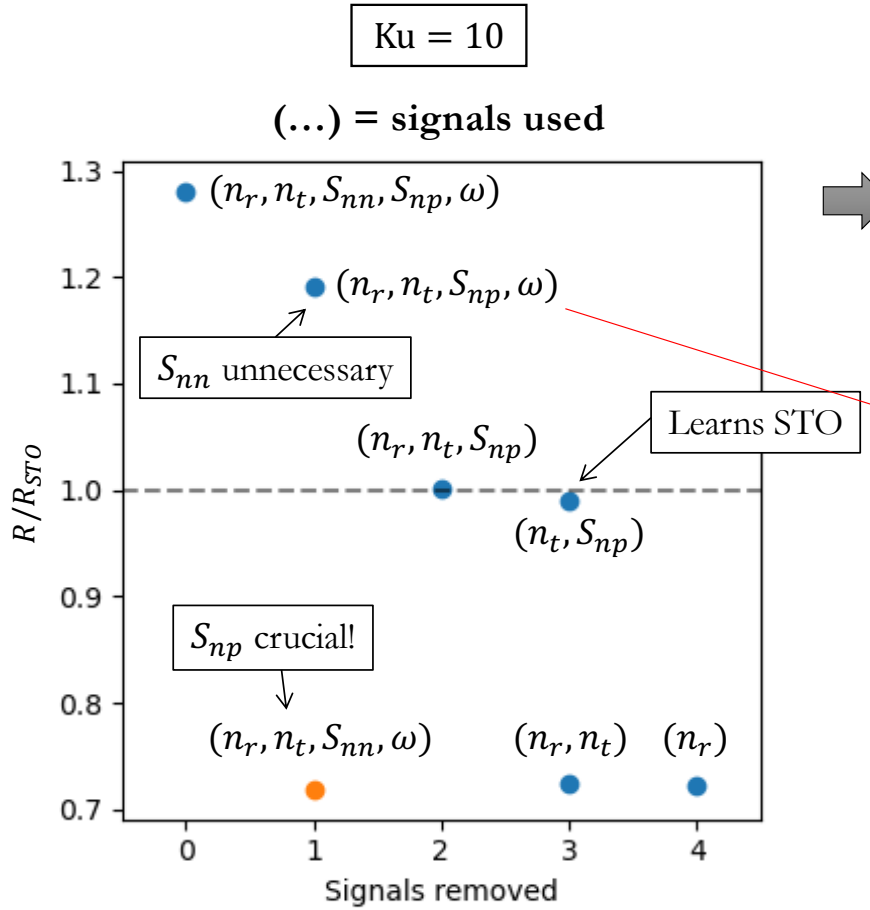
Emergence of $\boldsymbol{S_{np}}$-$\boldsymbol{\omega}$ **relations** producing new Λ-patterns
(learned long-time planning)

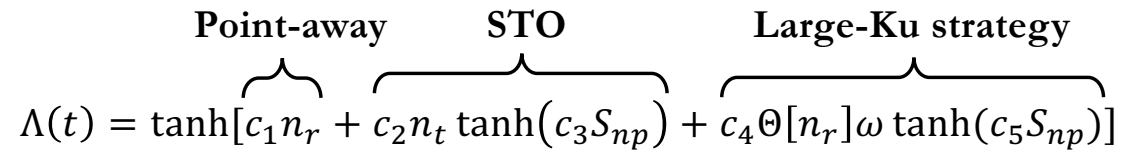


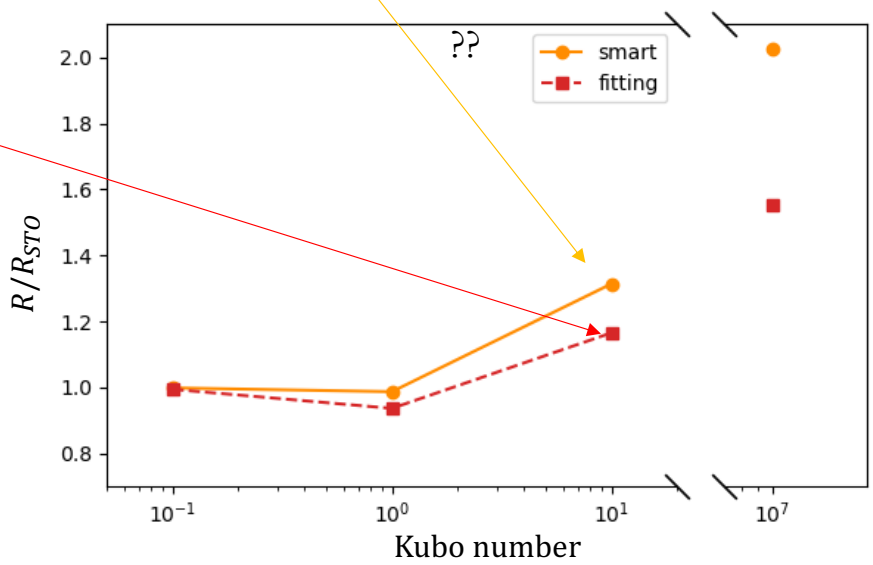
STO: $\Lambda(t) = -\Lambda_{max}\text{sign}(n_t)\text{sign}(S_{np})$

- Use vorticity/strain to stabilize orientation if it points away ($n_r > 0$)
- Become oblate when it points towards the origin ($n_r < 0$)

Signals hierarchy (features ranking) and minimal (interpretable) model

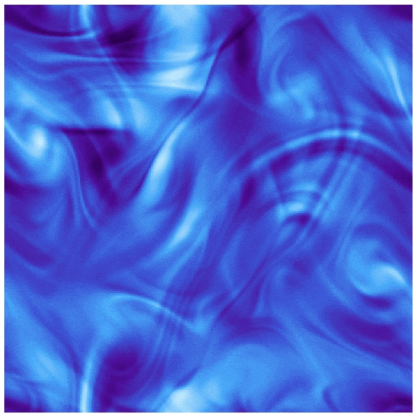






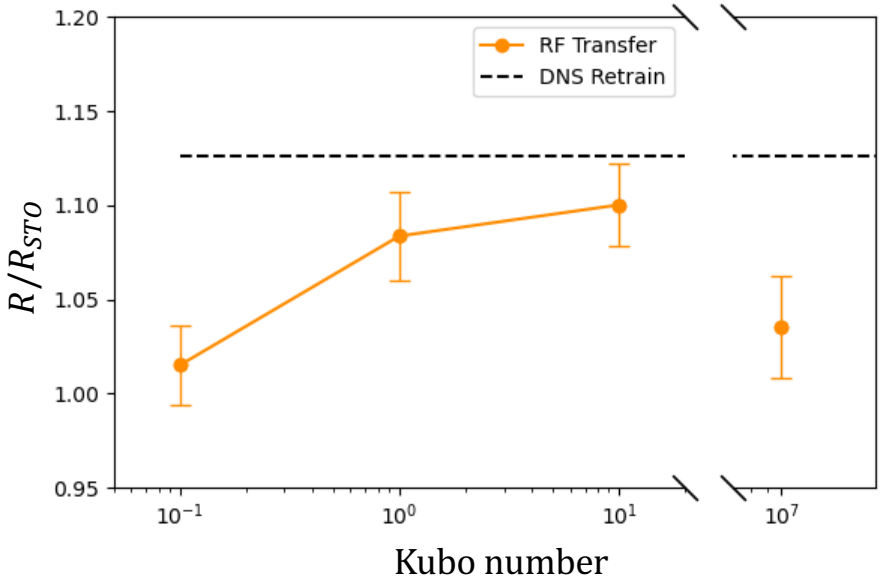
$$\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\nabla p + \nu \Delta \boldsymbol{u} + \boldsymbol{F}$$

Transfer learning to DNS turbulence



- 2D turbulent flow from DNS Navier–Stokes (**direct cascade**).
- **Input signals rescaled** so they have distributions similar to the ones in stochastic flows.
- Strategies learned in stochastic flows successfully capture the **underlying mechanism** which can be transferred to more realistic flows.

Trasnfer learning?



Conclusions

Summary of the results:

- **Task:** escape from a point in a complex flow by **controlling shape**.
- **RL tested against baselines:** constant shape (naïve) or Short-Time-Optimal (STO).
- Smart swimmers **adapt to the temporal structure** of the flow:
  - **Recovers STO** in the small Ku limit.
  - RL discovers **interpretable**, **physics-related** policies.
  - **Robust** to rotational noise, sensing window and time horizon.
- RL strategy **transfers** successfully from synthetic to **real turbulence**.

Possible extensions:

- Include **energetic** costs.
- Navigation in **3D** flows.
- Optimize **different tasks**: stay close to the origin, predator-prey dynamics, optimize feedback on the flow (needs fully differentiable two-way coupled Lagrangian-Eulerian codes)