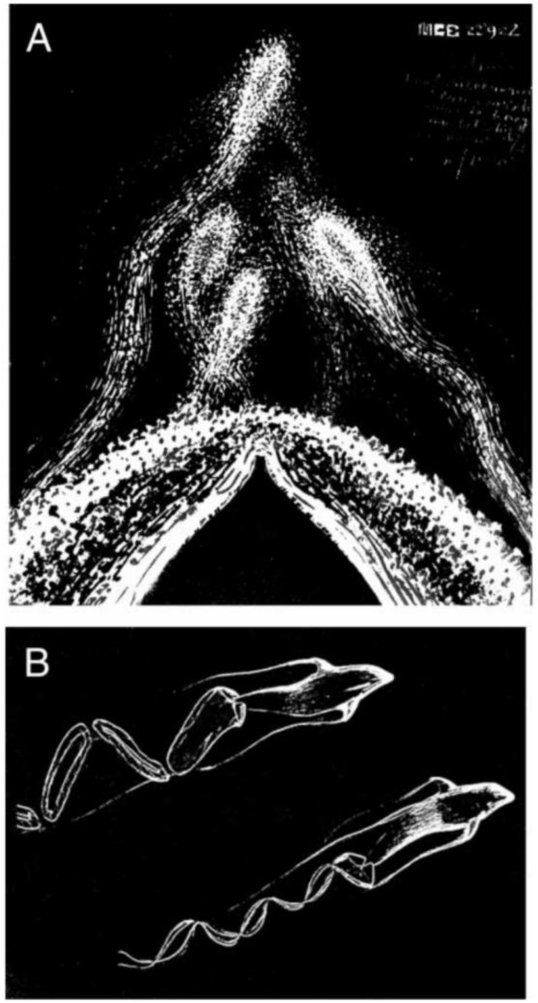
Bioluminescence modeling for deep sea experiments

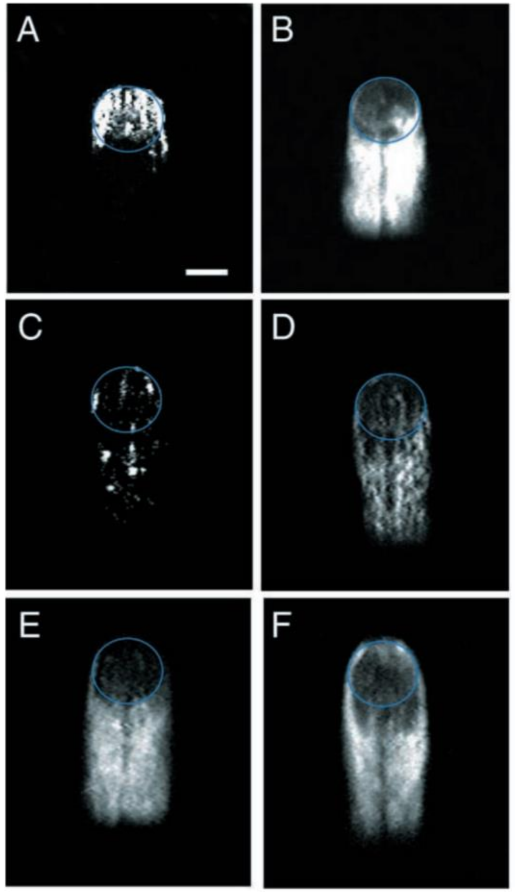
P-ONE virtual collaboration meeting, December

Li Ruohan, Golo Wimmer, **SMB**

Contents

- Current Status
- What We've Done
- New Results
- Plans for the Future

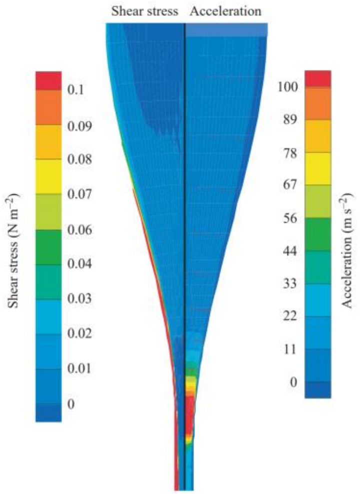


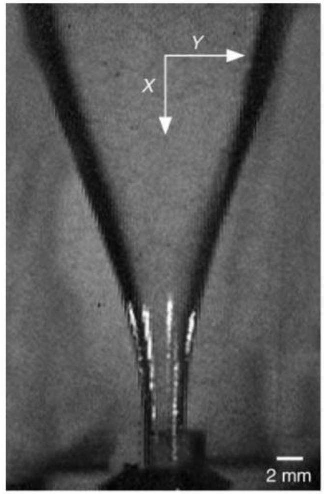


https://jeb.biologists.org/content/jexbio/201/9/1447.full.pdf

Current Status I

- Multiple studies exist
  - J. Rohr et al. (1998) (Dolphins)
  - M. I. Latz et al. (2004) (Nozzle)
  - …
  - For specific cases





Correlation between stress/acceleration and emission!

- For neutrino telescopes
  - I.G. Priede et al. (2008) (KM3Net, ANTARES)
  - J. Craig et al. (2009)
  - …

Strong assumptions

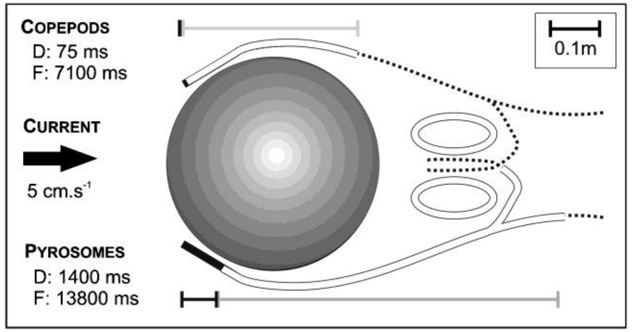
Current Status II

- The neutrino telescope studies exclusively assumed collision responses

$$Impacts\ s^{-1} = \pi\left(\frac{\phi_{sphere}}{2} + \frac{\phi_{animal}}{2}\right)^2 \times v \times \rho$$

- For $\rho \in [0.01, 0.05]m^{-3}$ the expected number of flashes is approx. 1 per hour

Predict more light downstream due to shear stress



-> No calculation

What we've done I: A new simulation Framework

Fourth Day

- Population Modeling
- Light Propagation
- Detector Response

Firedrake

- Water Current Modeling
- Navier-Stokes solver
- Added discretization and stabilization

Python package
Monte Carlo simulation
Modular
https://github.com/MeighenBergerS/fourth_day

Together model realistic populations and water flows

Interface included in Fourth Day
https://www.firedrakeproject.org/

What we've done II

Solve for Reynold's numbers $R_e \in [300, 30000]$

$$\partial_t \boldsymbol{u} + (\boldsymbol{u}\nabla)\boldsymbol{u} + \nabla \rho - \nu \Delta \boldsymbol{u} = 0$$

$$\langle \boldsymbol{w}, \boldsymbol{u}_t \rangle + \langle \boldsymbol{w}, (\boldsymbol{u}\nabla)\boldsymbol{u} \rangle + P(\boldsymbol{u}, p; \boldsymbol{w}) + \langle \tau(\boldsymbol{u}\nabla)\boldsymbol{w}, u_{res} \rangle = 0 \;\; \forall \boldsymbol{w} \in V_u$$

Space-discretized

Cauchy stress tensor

SUPG stabilization

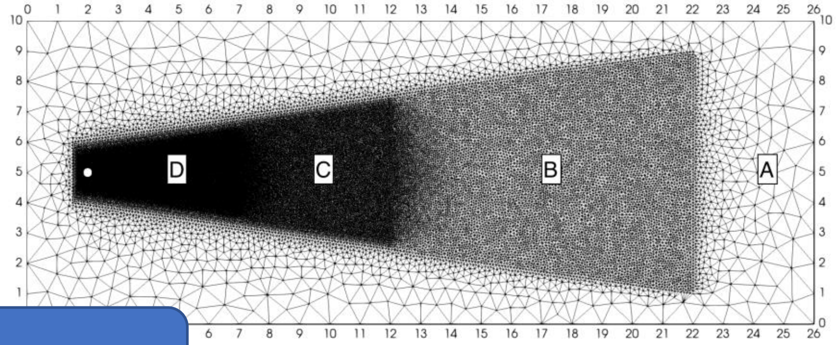
$$u_{res} = \partial_t \boldsymbol{u} + (\boldsymbol{u}\nabla)\boldsymbol{u} - \nabla \sigma(\boldsymbol{u}, p)$$

Residual

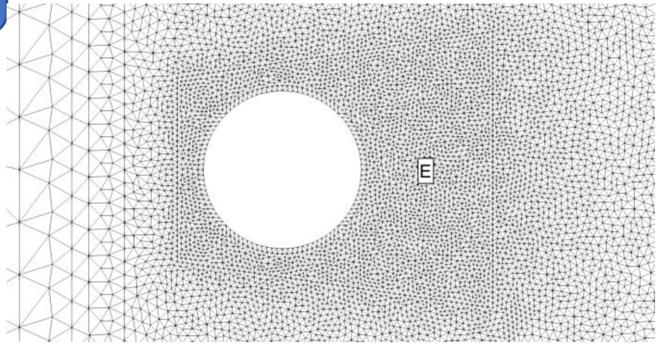
$$\tau = \frac{1}{2}\left(\left(\frac{2\kappa(x)}{\Delta t}\right)^2 + \left(\frac{2|\boldsymbol{u}|}{\Delta x}\right)^2 + 9\left(\frac{4\nu}{\Delta x^2}\right)^2\right)^{-1/2}$$

Stabilization parameter

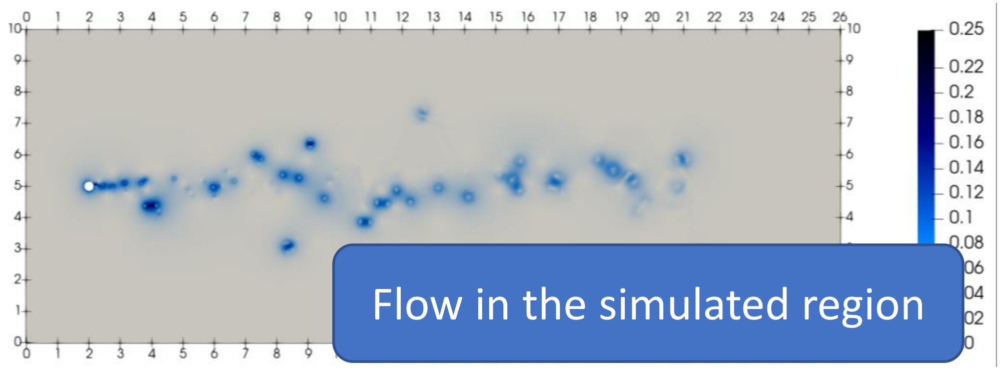
$$\kappa(x) = 1, x < 2.75\, m; \;\; \kappa(x) = \frac{1}{6} \text{ else}$$



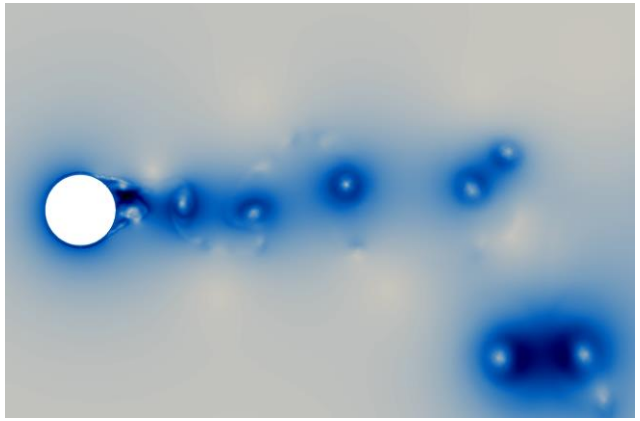
Mesh we use



What we've done III



Flow in the simulated region



Flow near the detector

$$P_{emission} = \alpha \nabla \boldsymbol{u}$$

Linear emission probability

Each organism has a host of properties

Model movement and emission using a MC scheme

species	Paraphyllina ransoni Russell
pos_x	0.5042
pos_y	0.682592
velocity	0.00409511
angle	2.27632
radius	0.00105776
energy	1
observed	True
max_emission	27.5055
emission fraction	0.5
regeneration	0.001
is_emitting	False
emission_duration	-200
encounter photons	0
shear photons	0
photons	0

Intermission - How to run a simulation

```python
# Module imports
from fourth_day import Fourth_Day, config
# Some example settings
config['scenario']['population size'] = 100
config['scenario']['duration'] = 6000
config['scenario']['organism movement'] = True
config['scenario']['injection']['rate'] = 1
config['scenario']['injection']['y range'] = [0., 10.]
# Organisms
config['organisms']['emission fraction'] = 0.1
config['organisms']['alpha'] = 2.
config['organisms']['photon yield'] = 1e10
config["organisms"]["filter"] = 'depth'
config["organisms"]["depth filter"] = 100000  # Note the absurd depth. This is used to exclusively simulate custom organisms
# Water
config['water']['model']['time step'] = 0.1
# Detector
config["geometry"]["detector properties"]["Custom"] = {
    "x_pos": 2.,
    "y_pos": 5.,
    "det num": 12, #12 pmts numbered by position
    "x_offsets": np.array(
        [0.1,0.,-0.1,0., 0.12,-0.12,-0.12,0.12, 0.2,-0.04,-0.2,0.04]
    ) / 2., #test radius 0.3 meter, real radius 0.15 meter
    "y_offsets": np.array(
        [0.,0.1,0.,-0.1,  0.12,0.12,-0.12,-0.12, 0.04,0.2,-0.04,-0.2]
    ) / 2.,
    "angle offset": 90.,  # In which direction the detector points
    "opening angle": 25.,  # 25., # from dark box rotation test result: +-25 degrees
    "wavelength acceptance": [ #position number,center wavelength,quantum efficiency
        [395., 405.,0.26], #0,400
        [505., 515.,0.16], #1,510
        [420., 430.,0.28], #2,425
        [465., 475.,0.23], #3,470
        [300., 600.,1.], #4,no filter
        [487., 497.,0.1], #5,492
        [540., 560.,0.1], #6,550
        [515., 535.,0.13], #7,525
        [475., 485.,0.2], #8,480
        [445., 455.,0.2], #9,450
        [455., 465.,0.23], #10,460
        [325., 375.,0.3], #11,350
    ],
}
# Creating a fourth_day object
fd = Fourth_Day()
# Launching solver
fd.sim()
```

Options for the desired scenario

Two types of output

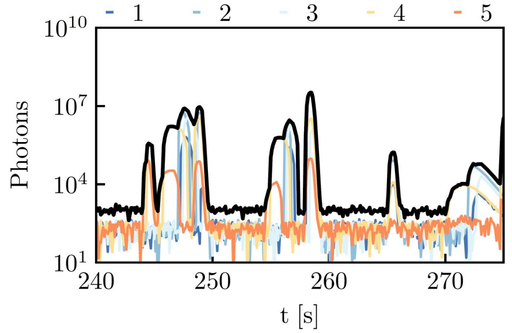
1) Population statistics

```python
fd.statistics[-1]
```

	species	pos_x	pos_y	velocity	angle	radius	energy	observed	max_emission	emission fraction	regeneration	pulse mean
0	test short pulse 2	14.996553	6.777046	0.0	0.0	0.001111	1.0	True	19.624966	0.1	0.0001	3.0
1	test short pulse 1	20.934872	4.254415	0.0	0.0	0.001487	1.0	True	10.621293	0.1	0.0001	3.0
2	test short pulse 1	12.965971	5.216831	0.0	0.0	0.001286	1.0	True	21.749651	0.1	0.0001	3.0
3	test short pulse 2	18.350581	0.416827	0.0	0.0	0.001158	1.0	True	6.333908	0.1	0.0001	3.0
4	test medium pulse 1	15.422526	9.020421	0.0	0.0	0.000609	1.0	True	14.241045	0.1	0.0001	30.0
...	...	...	...	...	...	...	...	...	...	...	...	...

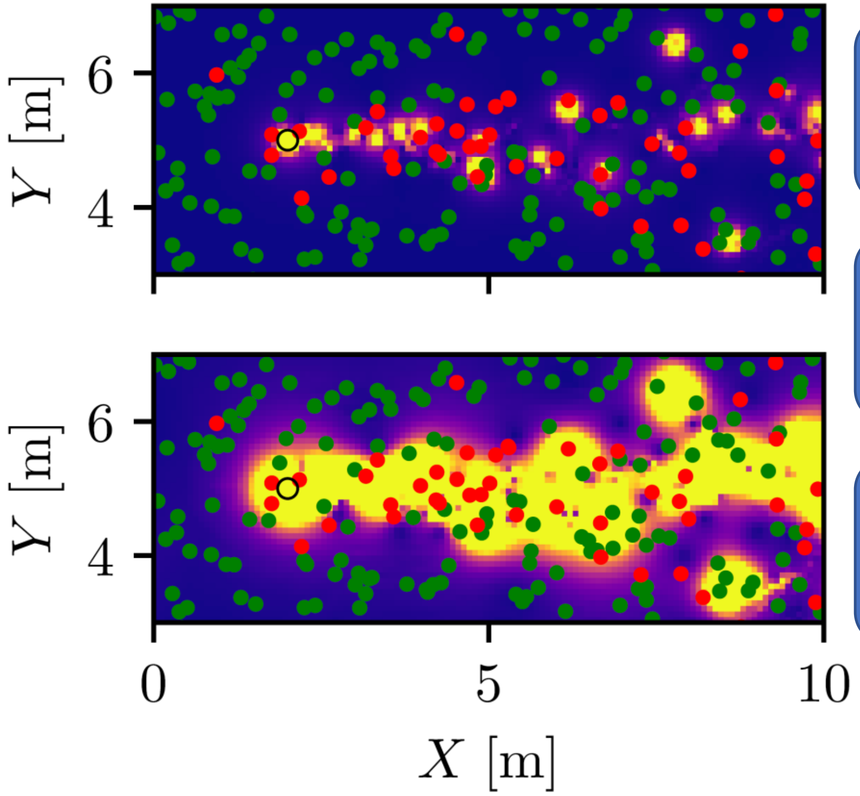
```python
fd.measured["Detector 1"]
```

2) Measured counts



New Results I

Confirm more light downstream; 1000x higher rate than pure collisions



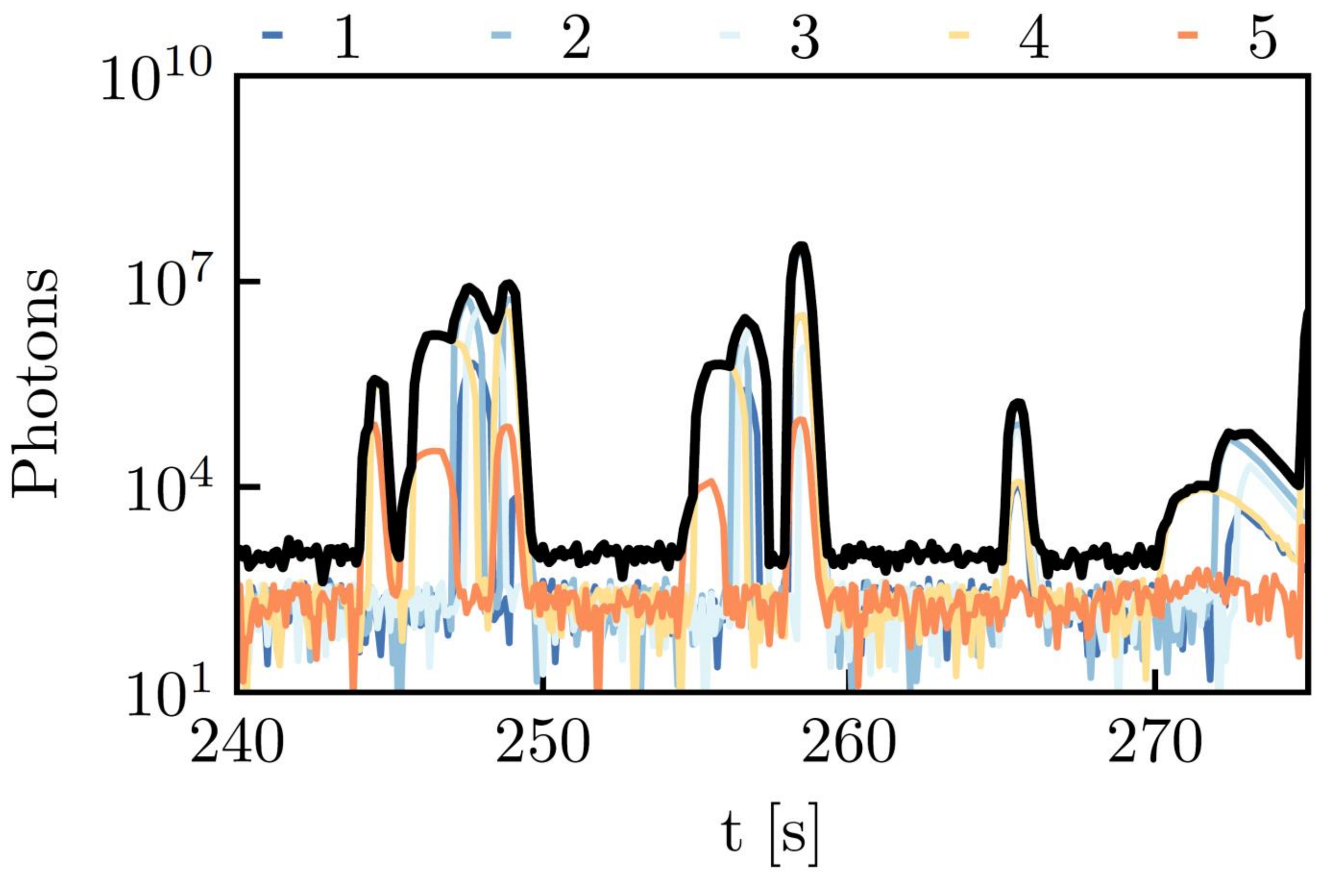
Color scale chosen to emphasize the vortices

Emitting and dormant organisms in red, green respectively

Color scale chosen to represent region where emissions are likely

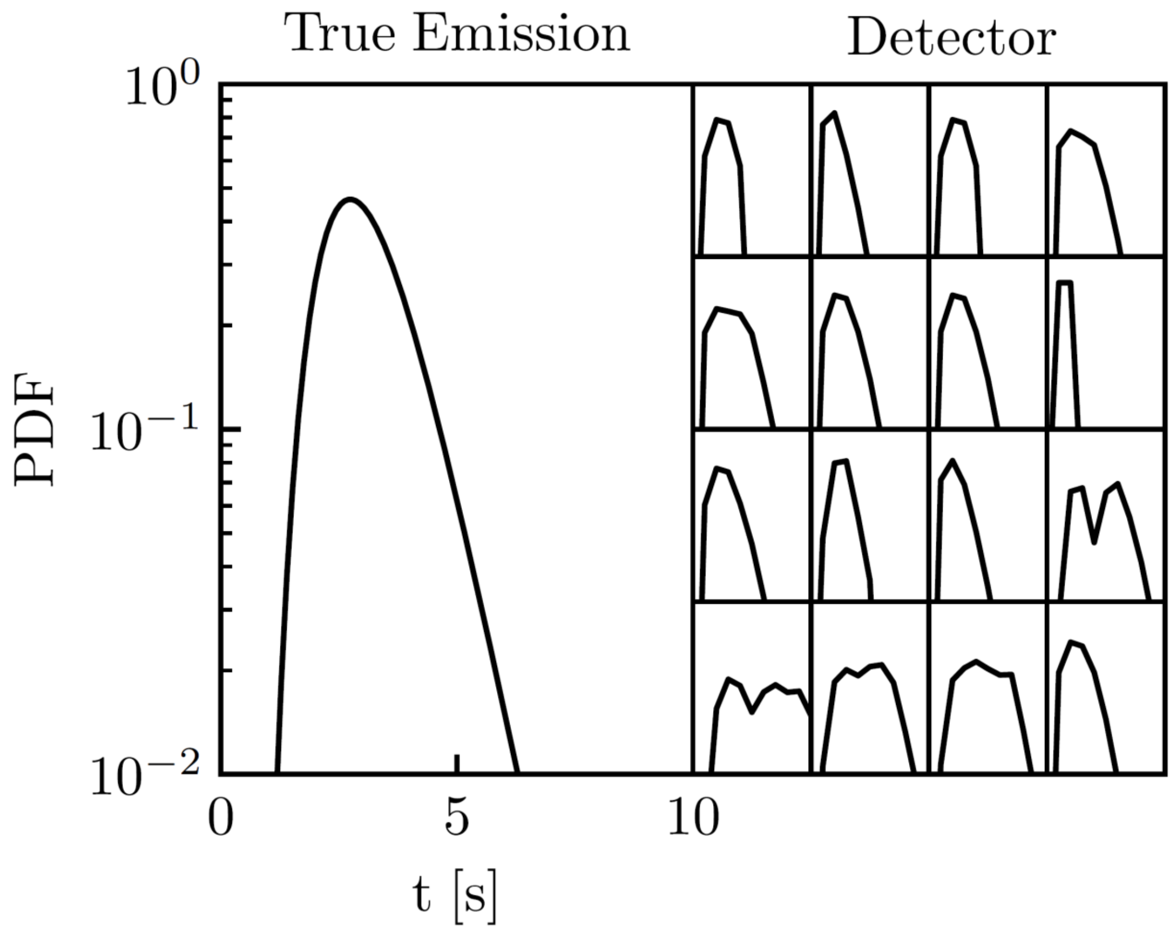
Most flashes happen in the detector's wake. The organisms have more time to respond to the applied shear/acceleration stress!

New Results II



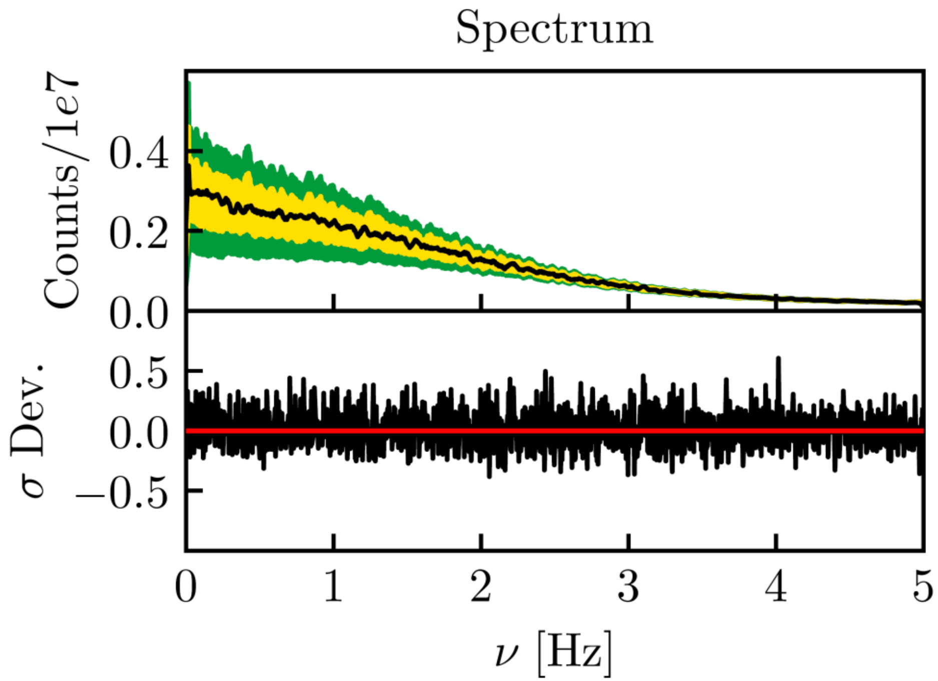
- Photon counts when using a detector like the PMTSpec of STRAW-b
- Each PMT measures different wavelengths and has a different FoV
- Organisms drift from FoV to FoV causing a single emission to appear as multiple
- Effect depends on the geometry and water current

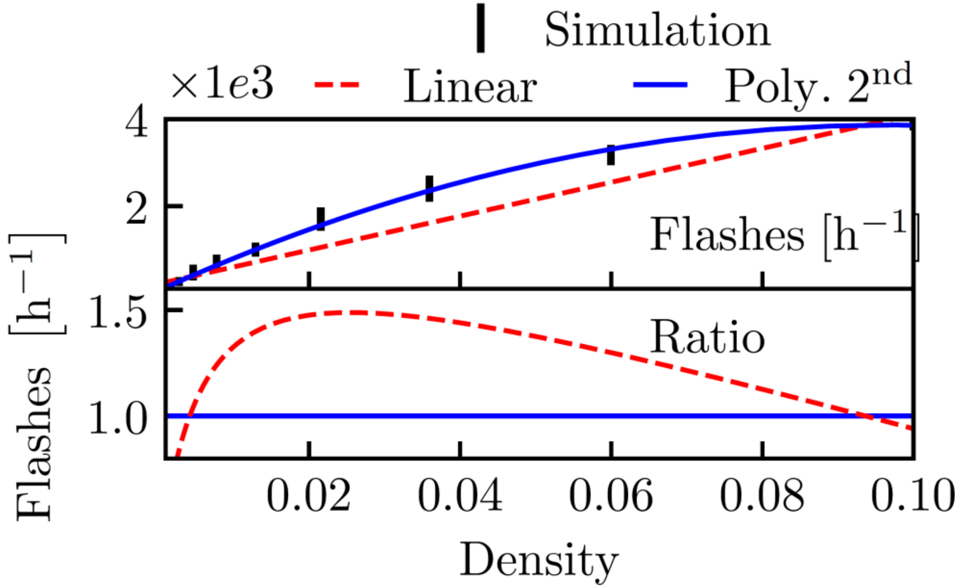
New Results III



- Example how the emission PDF of a single species (left) appears in the detector (right)
- These types of simulations can be used to categorize the organisms flowing past the detector

New Results IV





- Simulated spectrum vs chaotic background spectrum
- No significant difference between chaotic and simulated spectra

- The average number of flashes per hour
- Usually this is modeled using a linear fit
- Not possible in our case, require a Poly 2nd order

Plans for the Future

- Submit a paper
  - Preprint on arxiv in the next few days
- Write documentation
  - Code is easy to use for most cases
- Estimate 3D effects
  - 2D should be enough for most cases
- Cross-check some solver stuff
  - Strouhal number is still a little high

Bioluminescence modeling for deep sea experiments

Meighen-Berger S.[a,1], Ruohan L.[c,3], Wimmer G.[b,2]

[a] *Technische Universität München, James-Franck-Straße, 85748, Garching, Germany*
[b] *Imperial College London*
[c] *Ludwig-Maximilians-Universität München, Schellingstraße 4, 80799, München*

Abstract

We develop a modeling framework for bioluminescence light found in the deep sea near neutrino telescopes. Responding to the turbulence caused by the submerged detectors, the organisms emit light. For this reason, exact modeling of the flow is required. The current is governed by the in-compressible Navier-Stokes equation which needs to be solved for Reynold's numbers between 300 and 30000. We solve the equation by discretisizing the space using the finite element method. Further, we include the Streamline Upwind Petrov Galerkin Method in the velocity equation to avoid instabilities. On top of the flow model, we simulate a population of random microscopic organisms. Their movement and emission are stochastic processes which we model using Monte Carlo methods. We observe unique time-series for the photon counts depending on the flow velocity and detector specifications. This opens up the possibility of categorizing organisms using neutrino detectors. We show that the average light-yield and pulse shapes require exact flow modeling, while the emission timing is chaotic. From this we construct a fast modeling scheme, requiring only a subset of computationally expensive flow and population modeling.

Thank you for your attention!
Questions?