Implementing the Gaussian puff atmospheric dispersion model and using it to estimate methane emission rates

William Daniels and Ryker Fish

Department of Applied Mathematics and Statistics

Colorado School of Mines

COLORADO SCHOOL OF MINES

Agenda

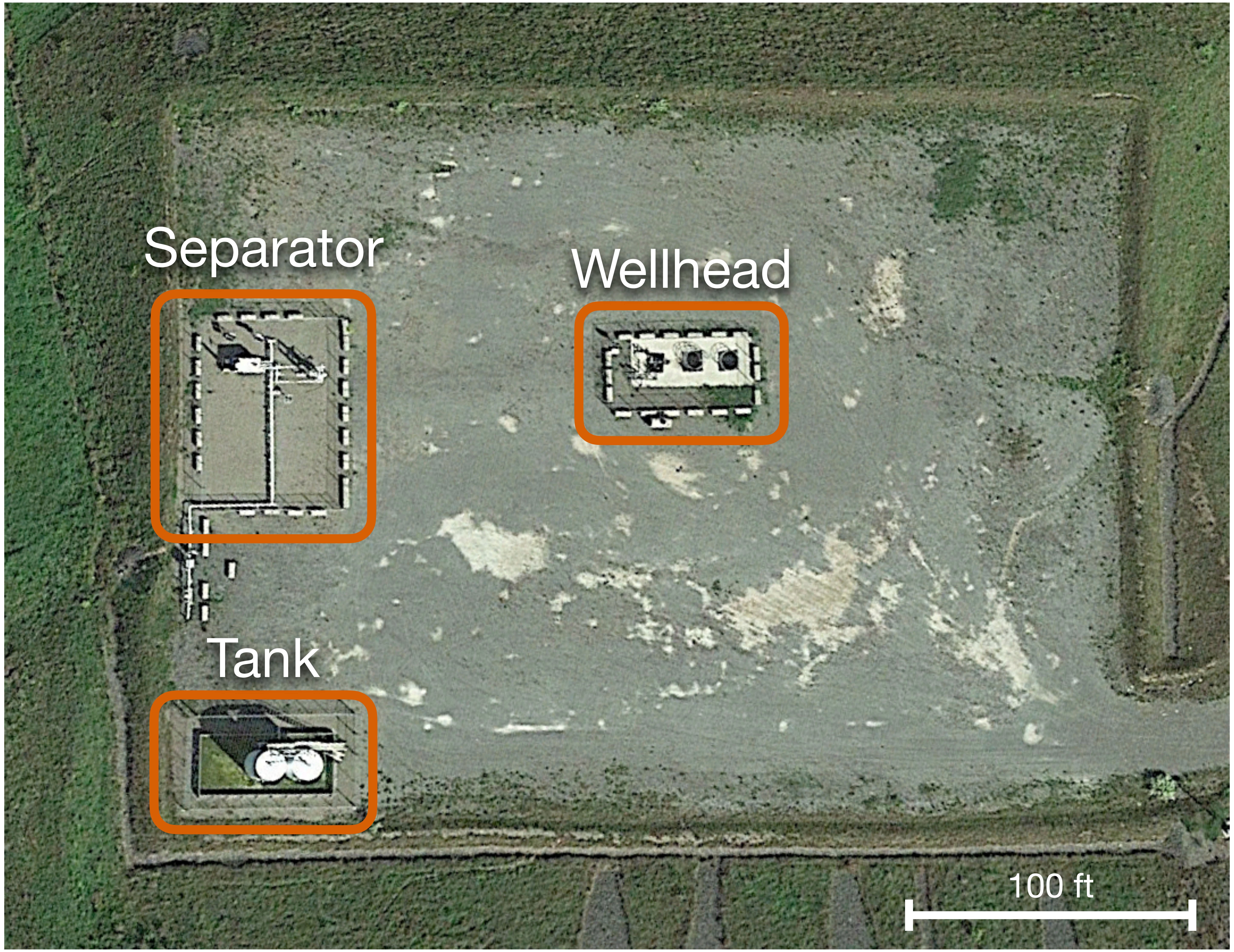
Talk (~30 min):

- Overview of the Gaussian plume and Gaussian puff models
- Mathematical relationship between the Gaussian plume and puff models
- Using the Gaussian puff model to estimate emission source and rate
- Fast implementation of the Gaussian puff model

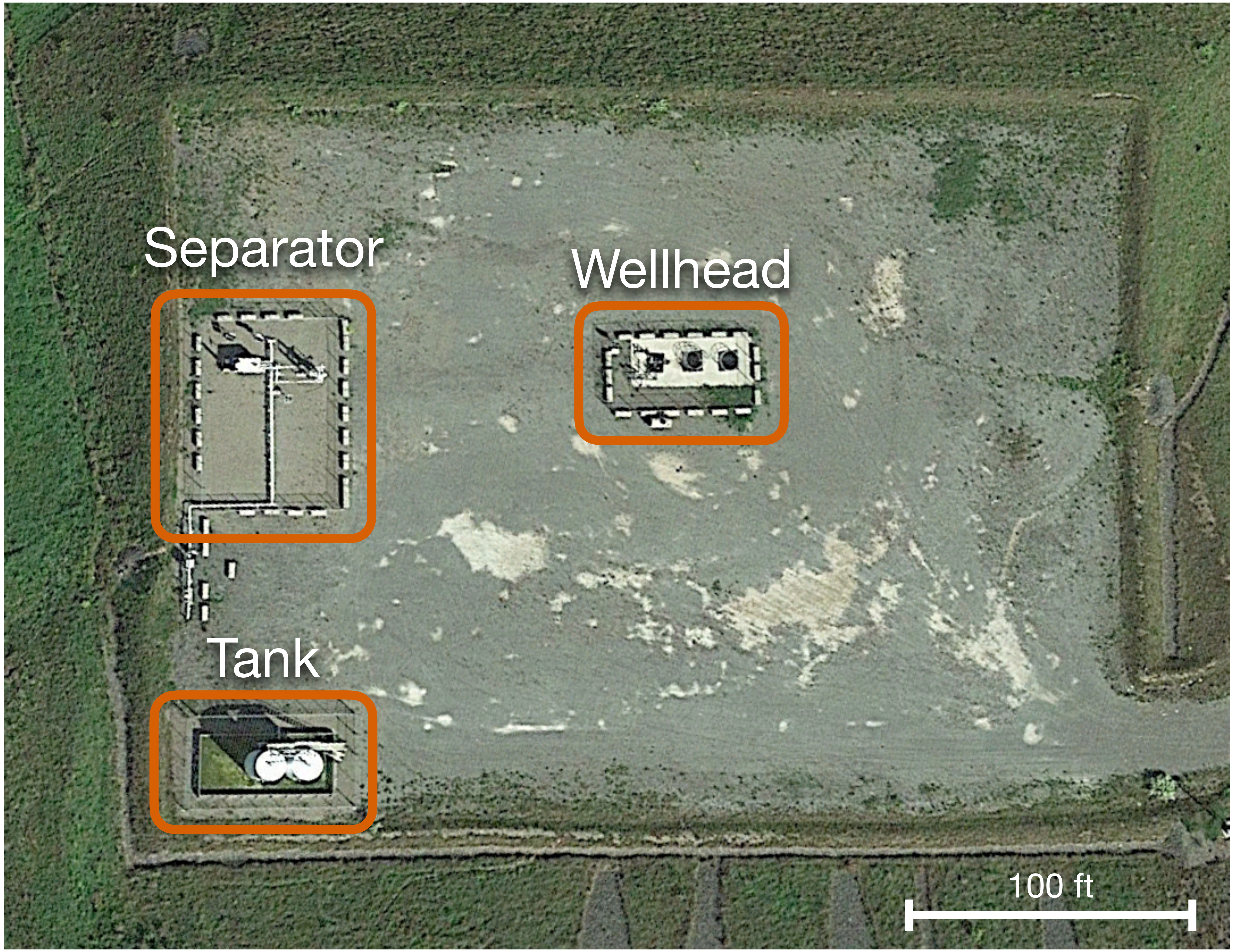
Workshop (~60 min):

- Running the Gaussian puff
- Running the inversion to estimate source and rate
- Work time and questions

Overview

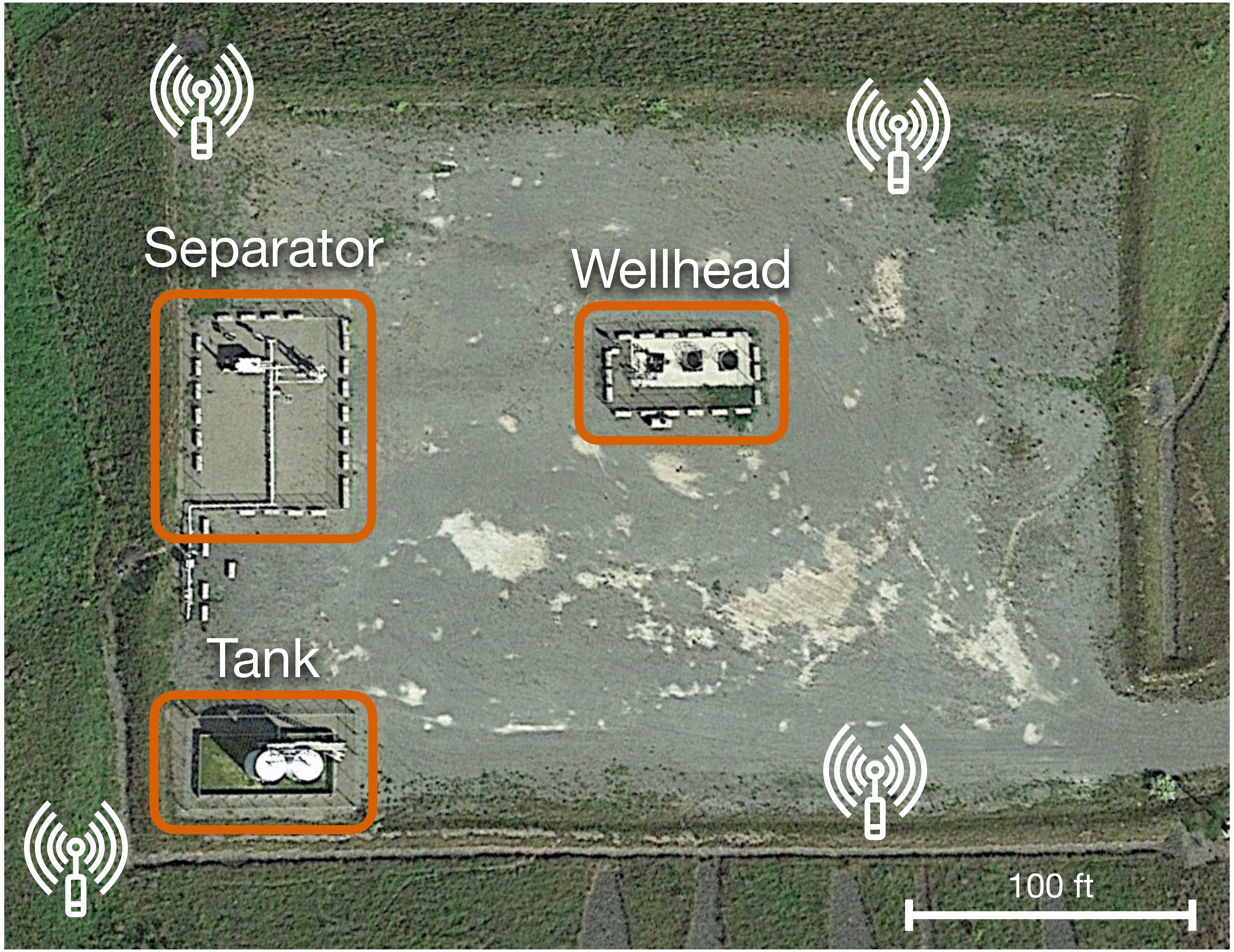


Overview



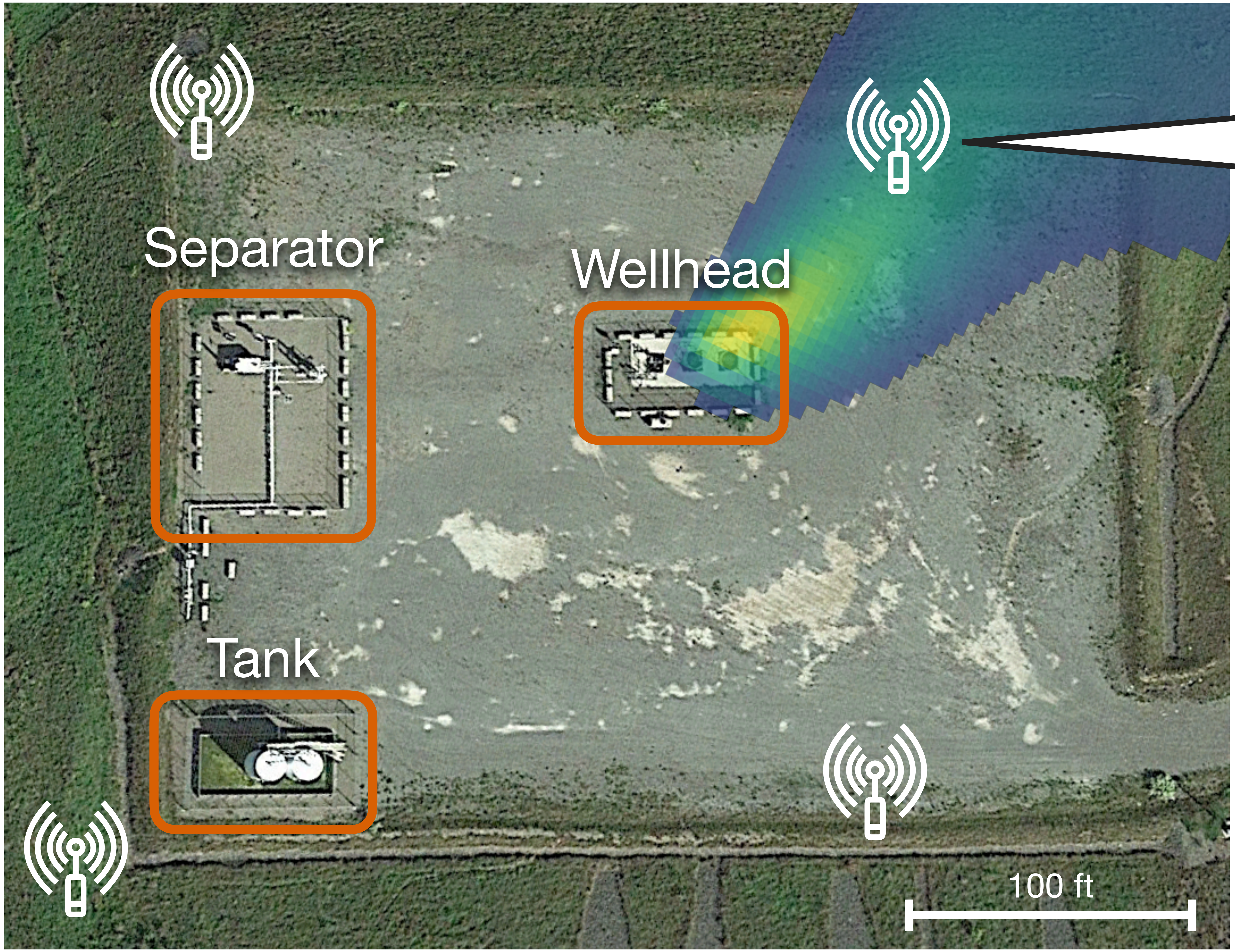
Continuous monitoring system (CMS)

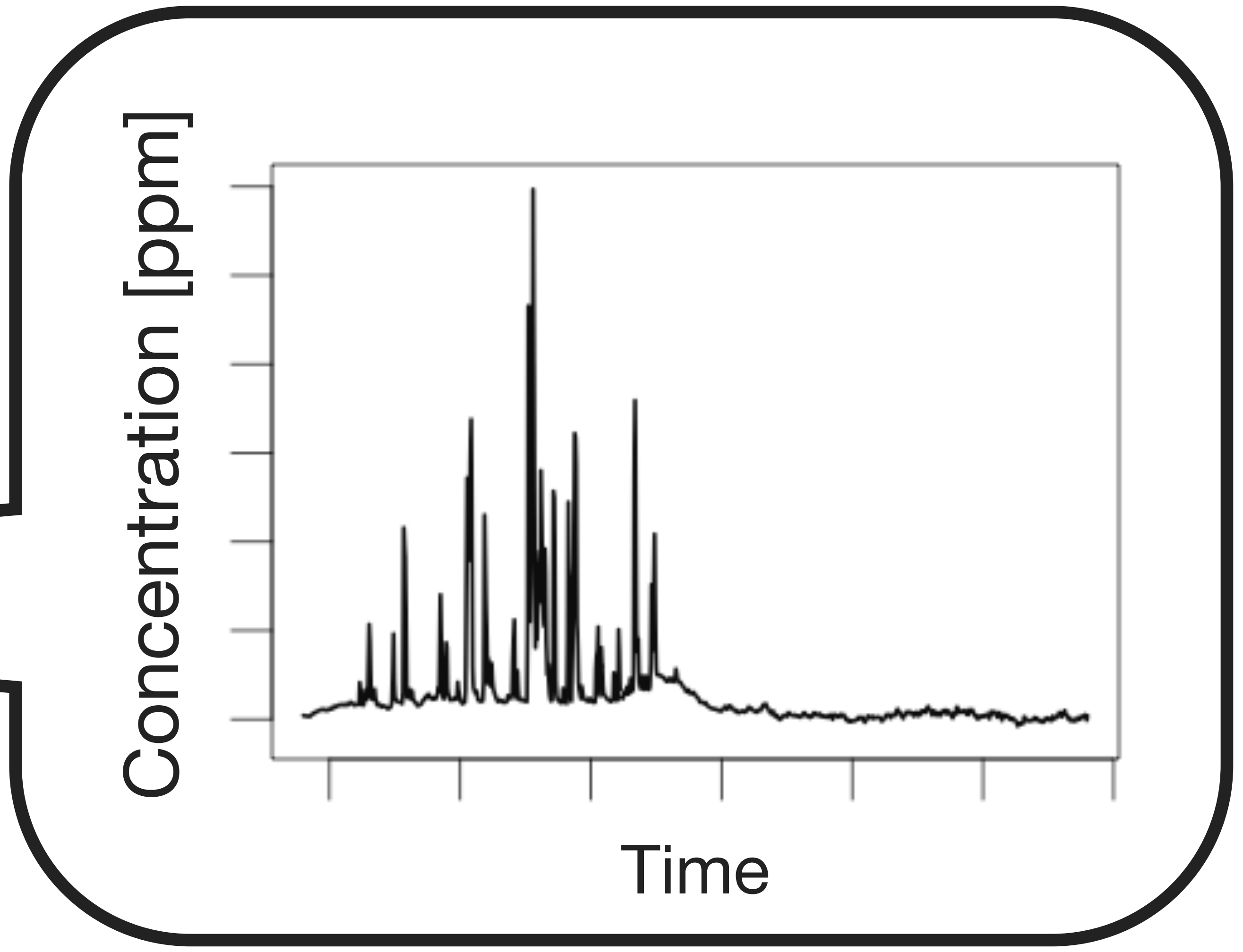
Overview



Continuous monitoring system (CMS)

Overview





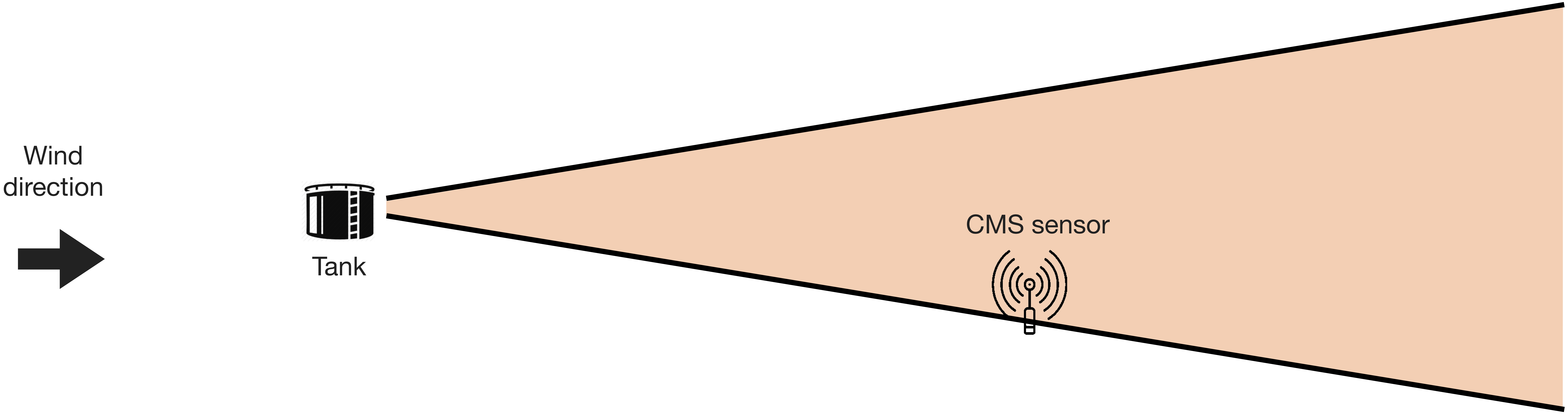
Atmospheric transport models (like the Gaussian puff) simulate the transport of a trace gas (like methane) from a source to a receptor

Gaussian plume model:

models the transport of methane by assuming that everything is steady state

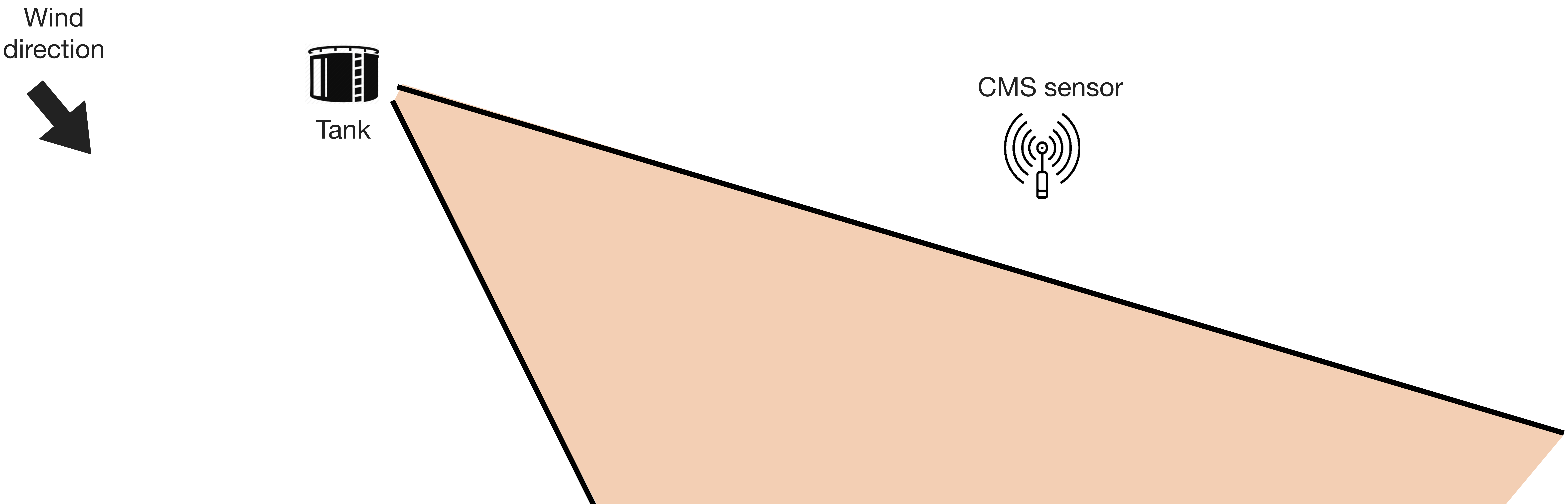
Gaussian plume model:

models the transport of methane by assuming that everything is steady state



Gaussian plume model:

models the transport of methane by assuming that everything is steady state

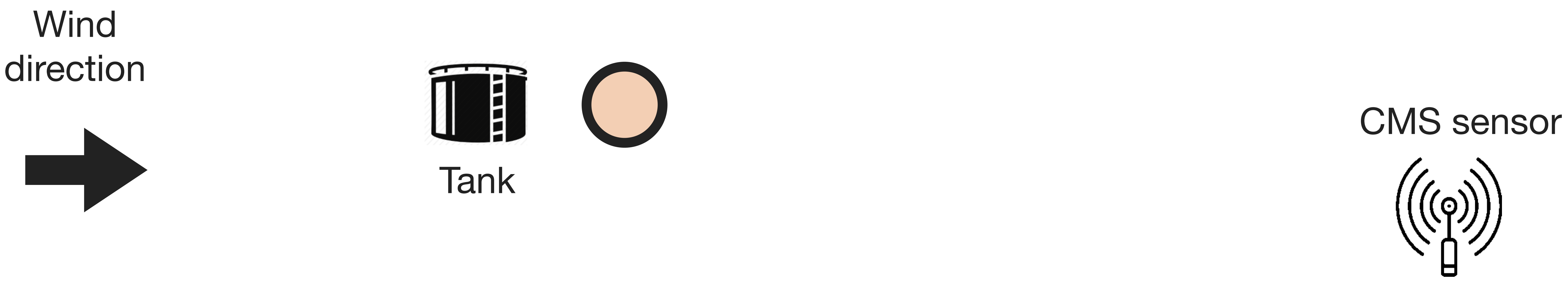


Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”

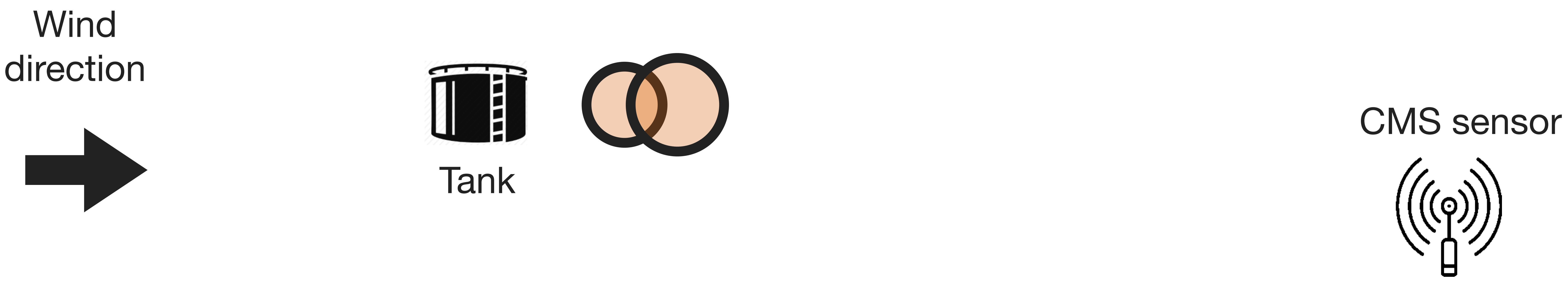
Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”



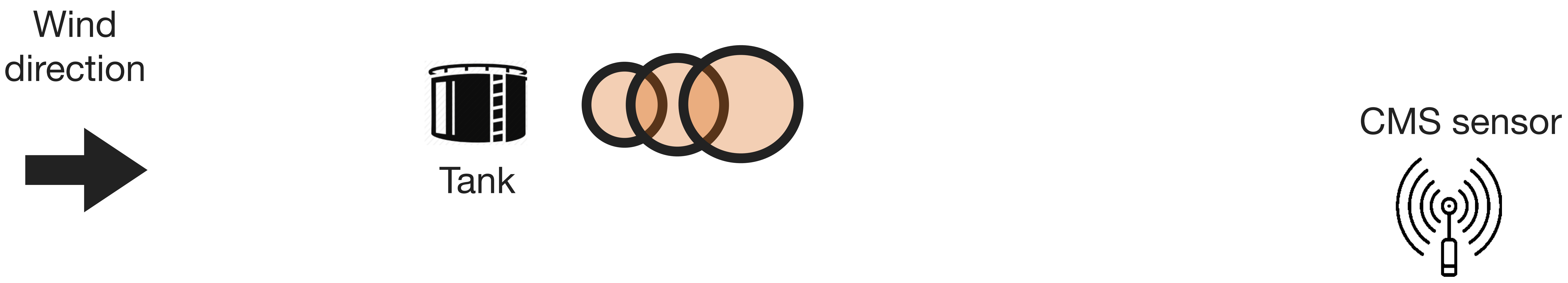
Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”



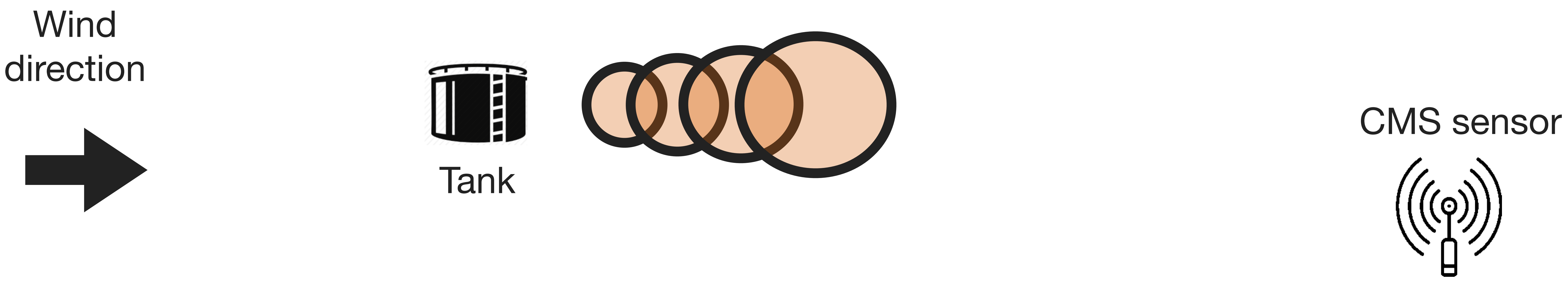
Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”



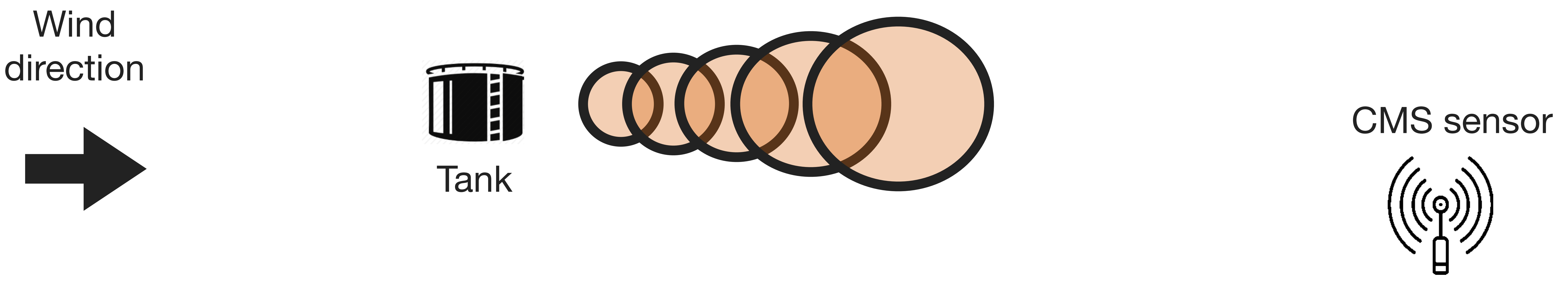
Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”



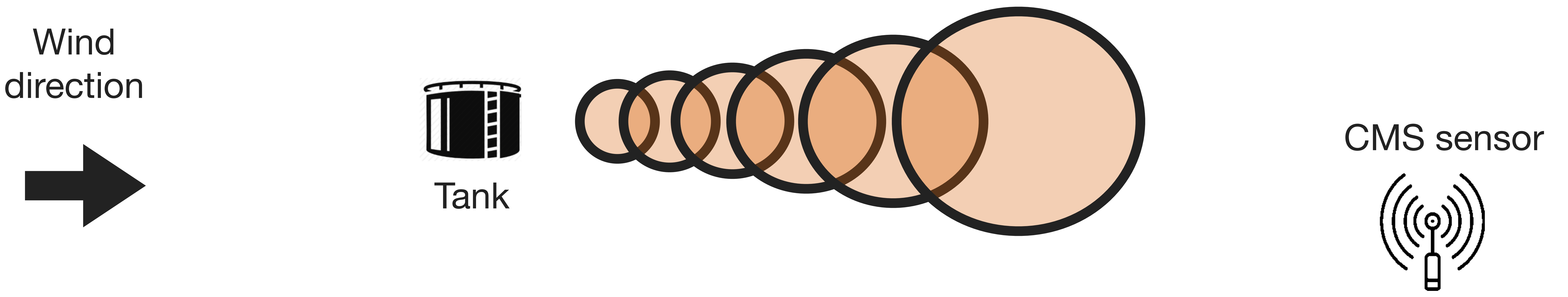
Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”



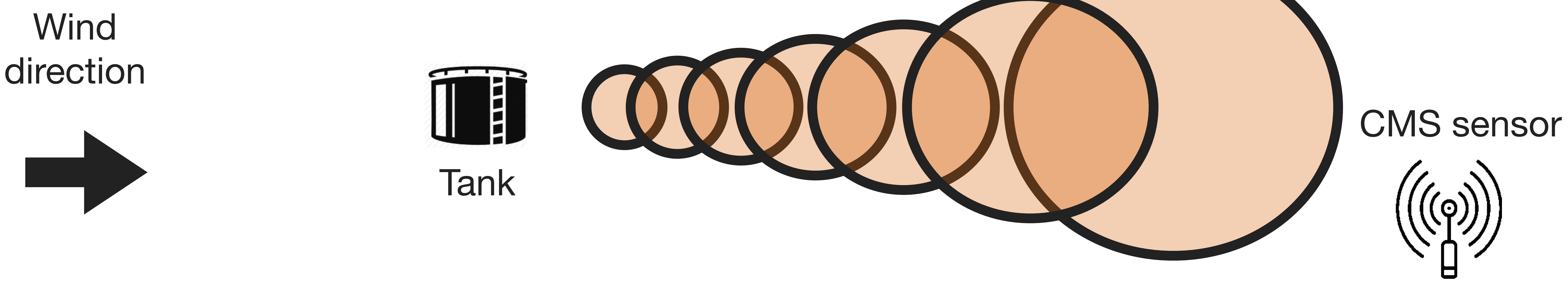
Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”



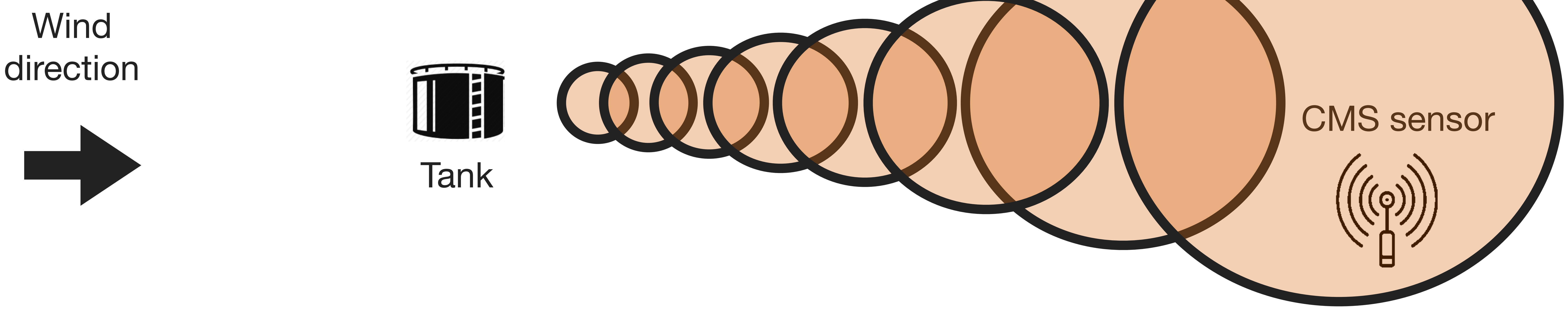
Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”



Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”

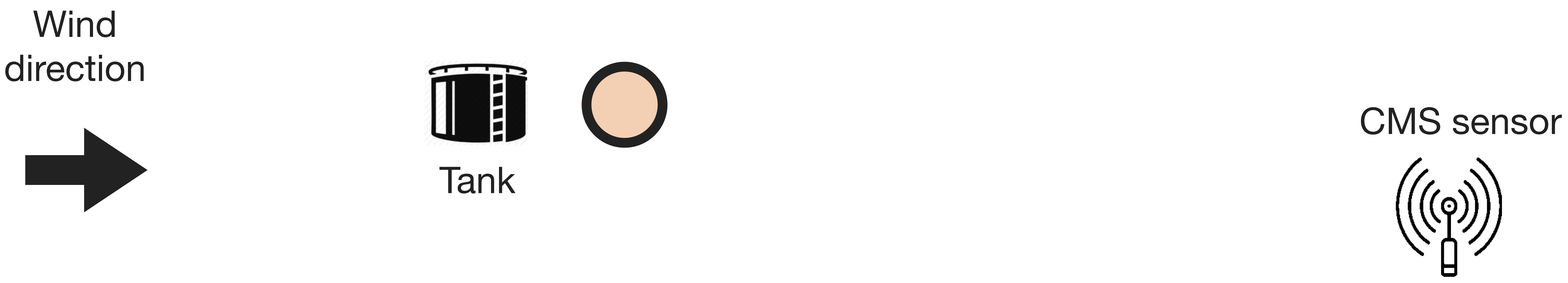


Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”

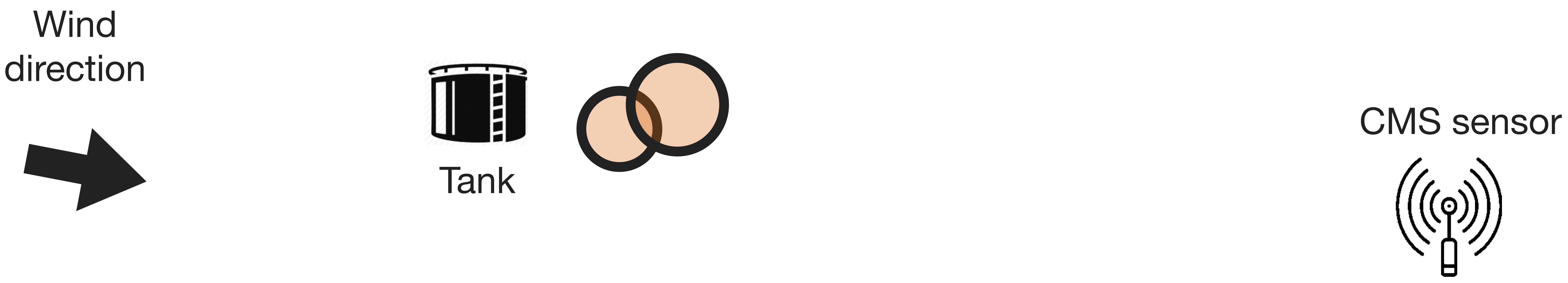
Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”



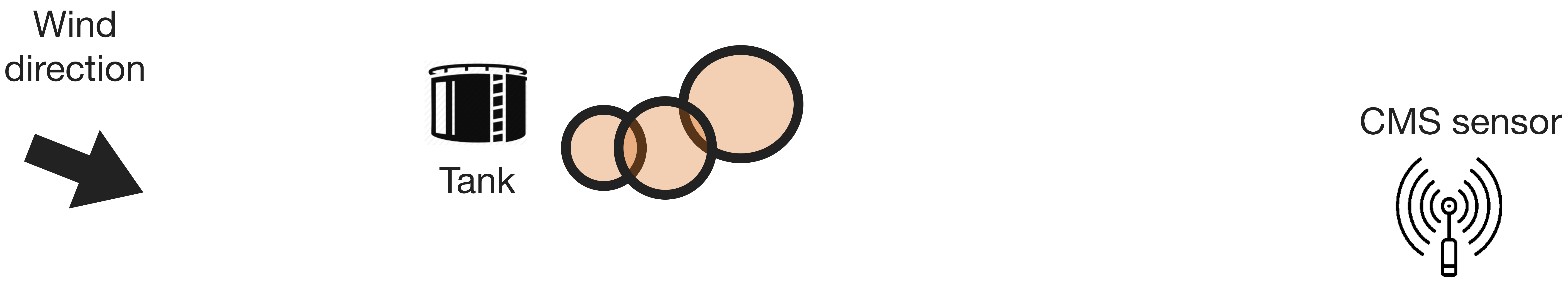
Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”



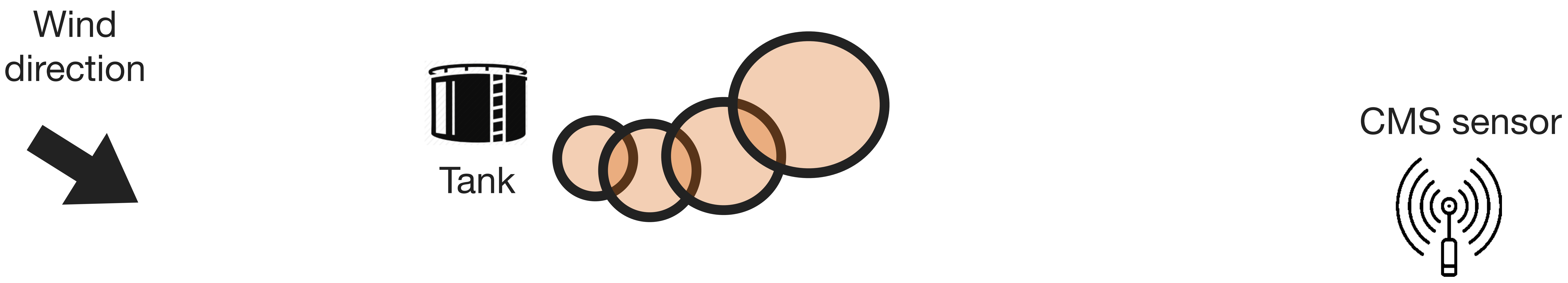
Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”



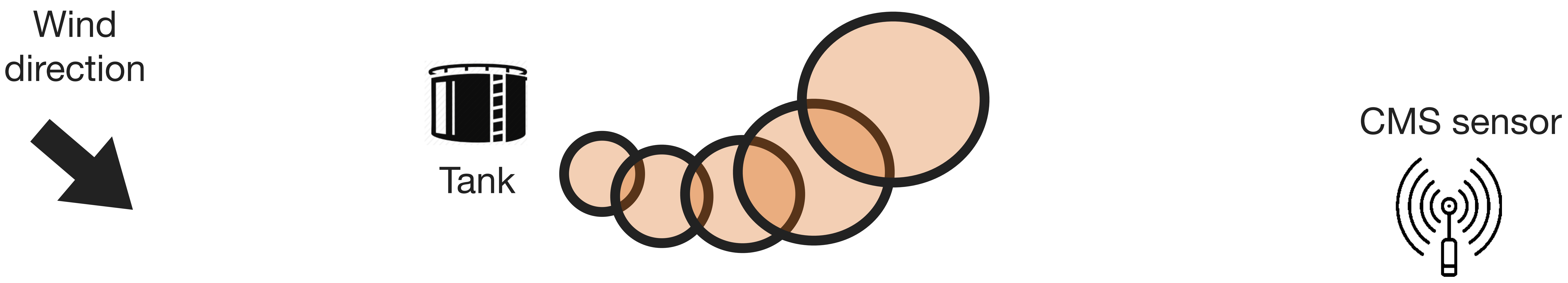
Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”



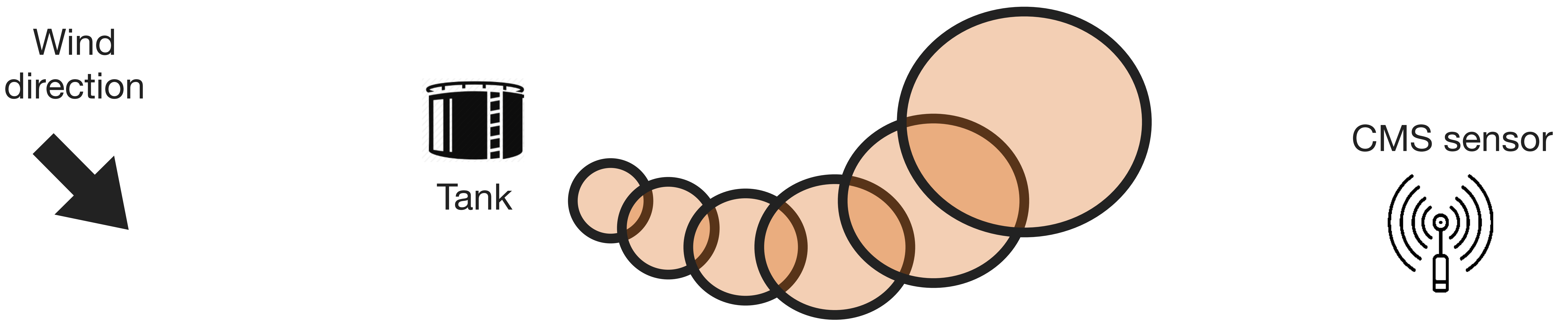
Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”



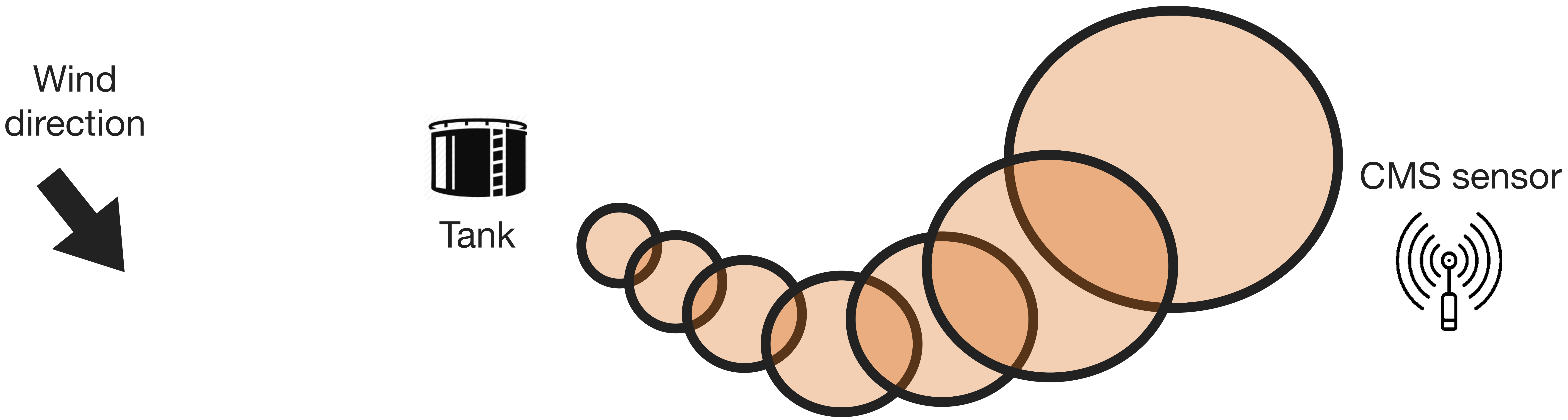
Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”



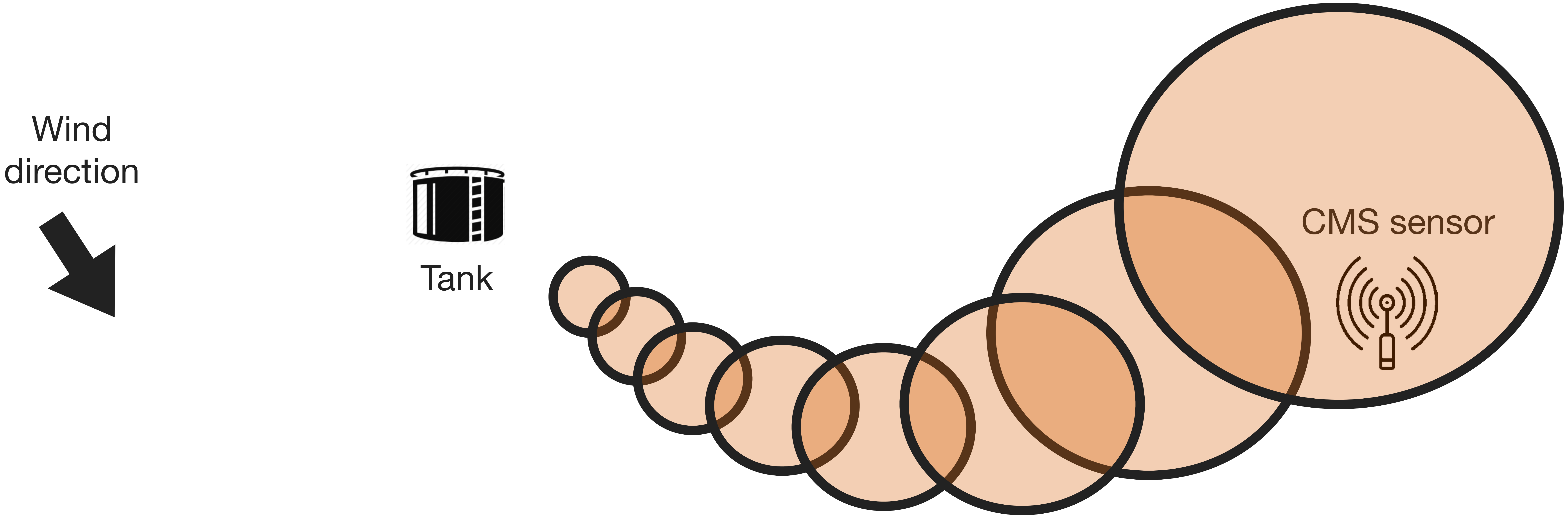
Gaussian puff model:

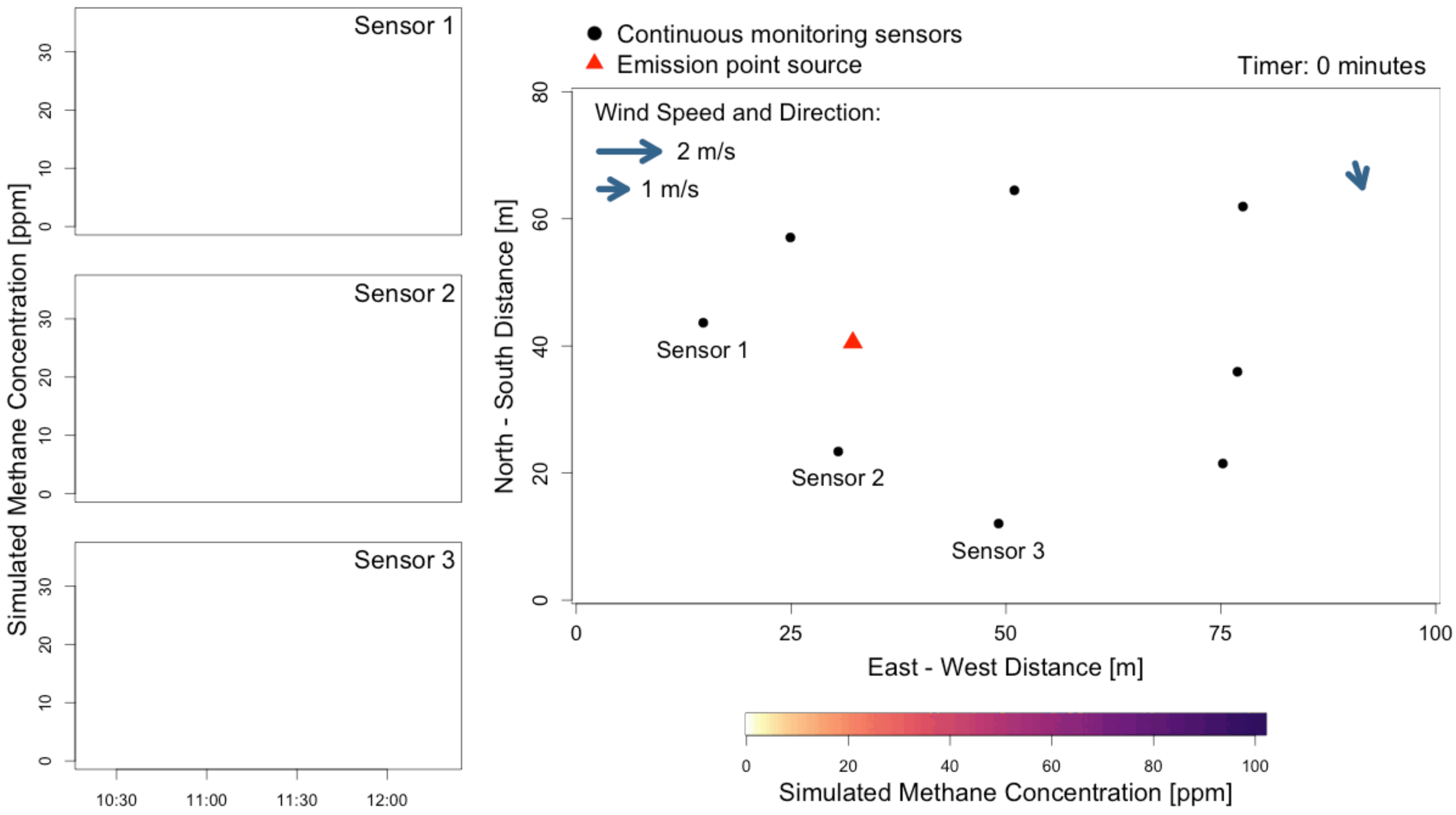
approximates a continuous release of methane as a sum of many small “puffs”



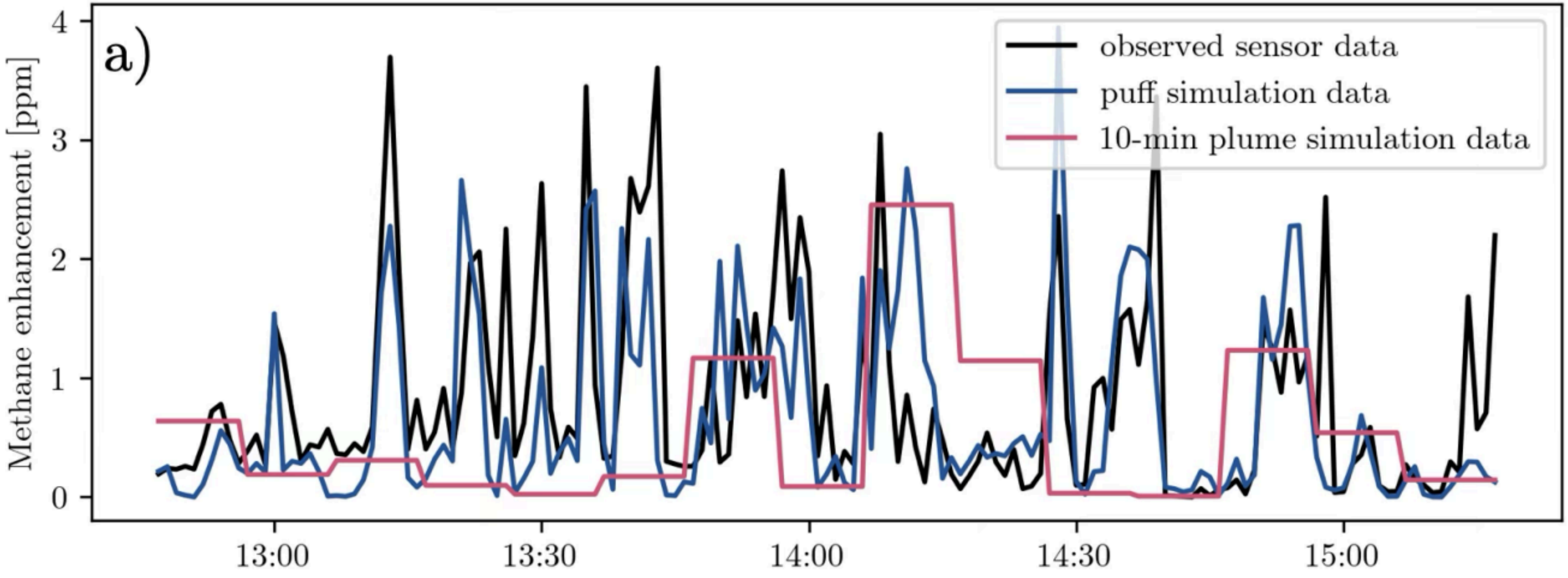
Gaussian puff model:

approximates a continuous release of methane as a sum of many small “puffs”



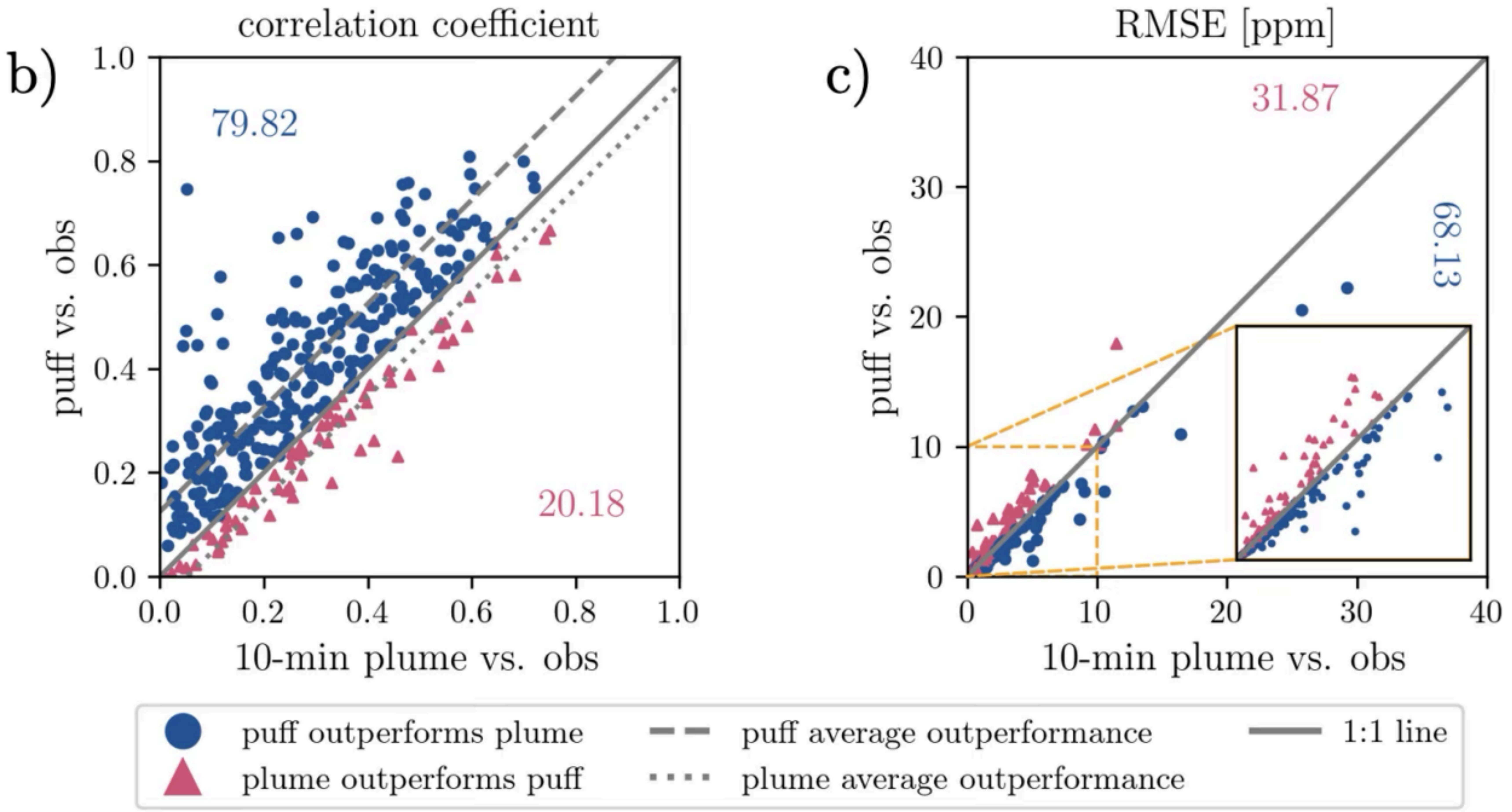

Sensor 1
Sensor 2
Sensor 3
Simulated Methane Concentration [ppm]
10:30
11:00
11:30
12:00
Continuous monitoring sensors
Emission point source
Timer: 0 minutes
Wind Speed and Direction:
2 m/s
1 m/s
North - South Distance [m]
East - West Distance [m]
Simulated Methane Concentration [ppm]

The Gaussian puff can leverage high frequency wind data, while the Gaussian plume requires a temporal average.



Meng Jia, Ryker Fish, et al. A fast and lightweight implementation of the Gaussian puff model for near-field atmospheric transport of trace gasses. *Scientific Reports (in press)*, 2025.

The Gaussian puff can leverage high frequency wind data, while the Gaussian plume requires a temporal average.



Meng Jia, Ryker Fish, et al. A fast and lightweight implementation of the Gaussian puff model for near-field atmospheric transport of trace gasses. *Scientific Reports (in press)*, 2025.

Gaussian plume model: mathematical definition

Set up coordinate system so that source is at (0,0,H) and positive x-axis aligns with downwind vector

Exponential decay as you move away from the downwind vector in the vertical dimension

Emission rate

$$c(x,y,z) = \frac{Q}{2\pi u \sigma_y \sigma_z} \exp\left(-\frac{y^2}{2\sigma_y^2}\right)\left[\exp\left(-\frac{(z-H)^2}{2\sigma_z^2}\right) + \exp\left(-\frac{(z+H)^2}{2\sigma_z^2}\right)\right]$$

Predicted methane concentration at sensor location (x,y,z)

Exponential decay as you move away from the downwind vector in the horizontal plane

Gaussian puff model: mathematical definition

Set up coordinate system so that source is at (0,0,H) and positive x-axis aligns with downwind vector

Total volume of methane contained in puff p

$$c_p(x, y, z, t, Q) = \frac{Q}{(2\pi)^{3/2}\sigma_y^2\sigma_z} \exp\left(-\frac{(x-ut)^2+y^2}{2\sigma_y^2}\right)\left[\exp\left(-\frac{(z-H)^2}{2\sigma_z^2}\right)+\exp\left(-\frac{(z+H)^2}{2\sigma_z^2}\right)\right]$$

Predicted methane concentration at sensor location (x,y,z) and time t from puff p

Exponential decay in concentration in horizontal plane (x, y)

Exponential decay in concentration in vertical dimension (z)

Gaussian puff model: mathematical definition

Set up coordinate system so that source is at (0,0,H) and positive x-axis aligns with downwind vector

Total concentration at (x, y, z, t)

$$c(x, y, z, t, Q) = \sum_{p=1}^{P} c_p(x, y, z, t, Q)$$

Total volume of methane contained in puff p

$$c_p(x, y, z, t, Q) = \frac{Q}{(2\pi)^{3/2}\sigma_y^2\sigma_z} \exp\left(-\frac{(x-ut)^2 + y^2}{2\sigma_y^2}\right)\left[\exp\left(-\frac{(z-H)^2}{2\sigma_z^2}\right) + \exp\left(-\frac{(z+H)^2}{2\sigma_z^2}\right)\right]$$

Predicted methane concentration at sensor location (x,y,z) and time t from puff p

Exponential decay in concentration in horizontal plane (x, y)

Exponential decay in concentration in vertical dimension (z)

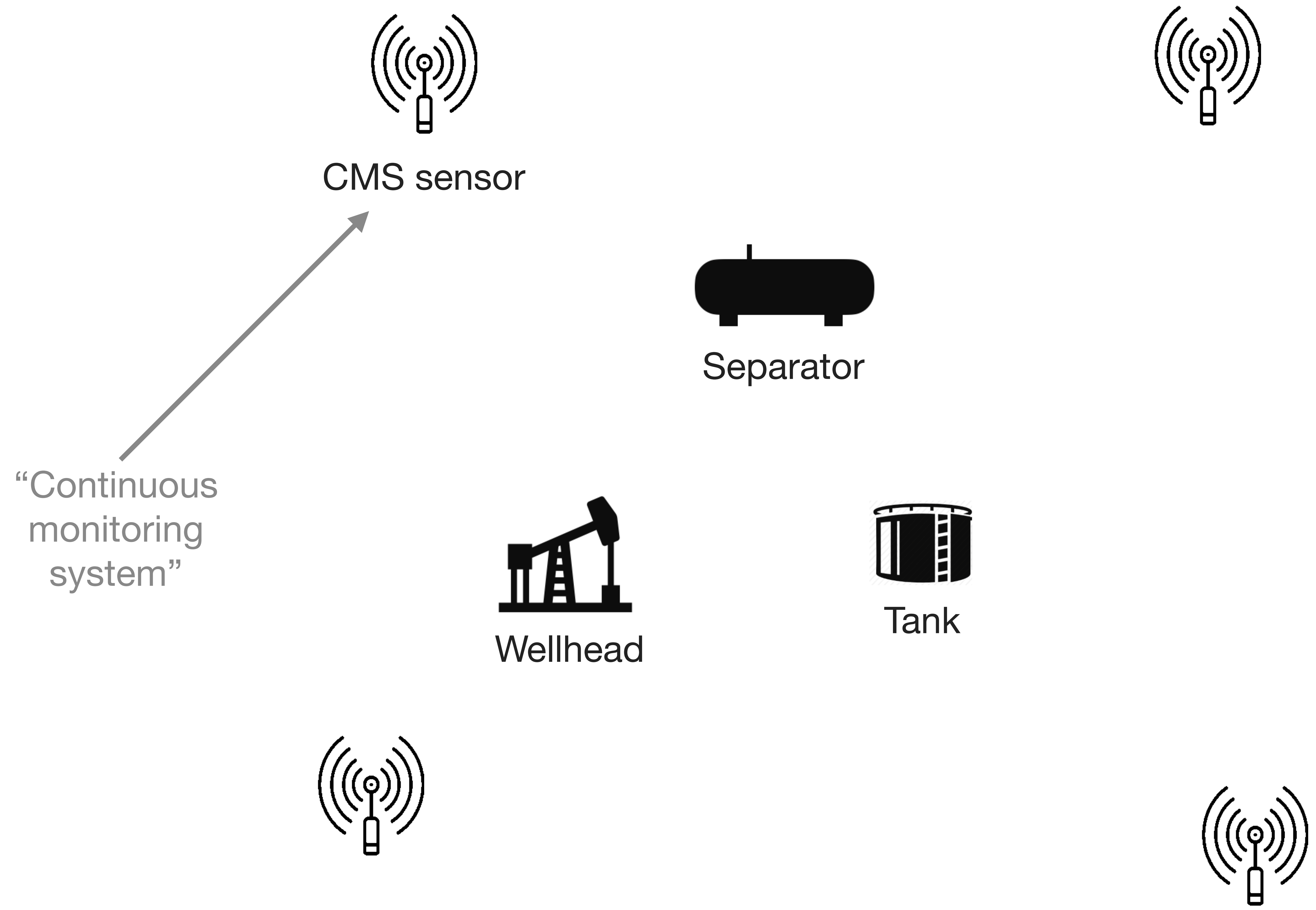
Agenda

Talk (~30 min):

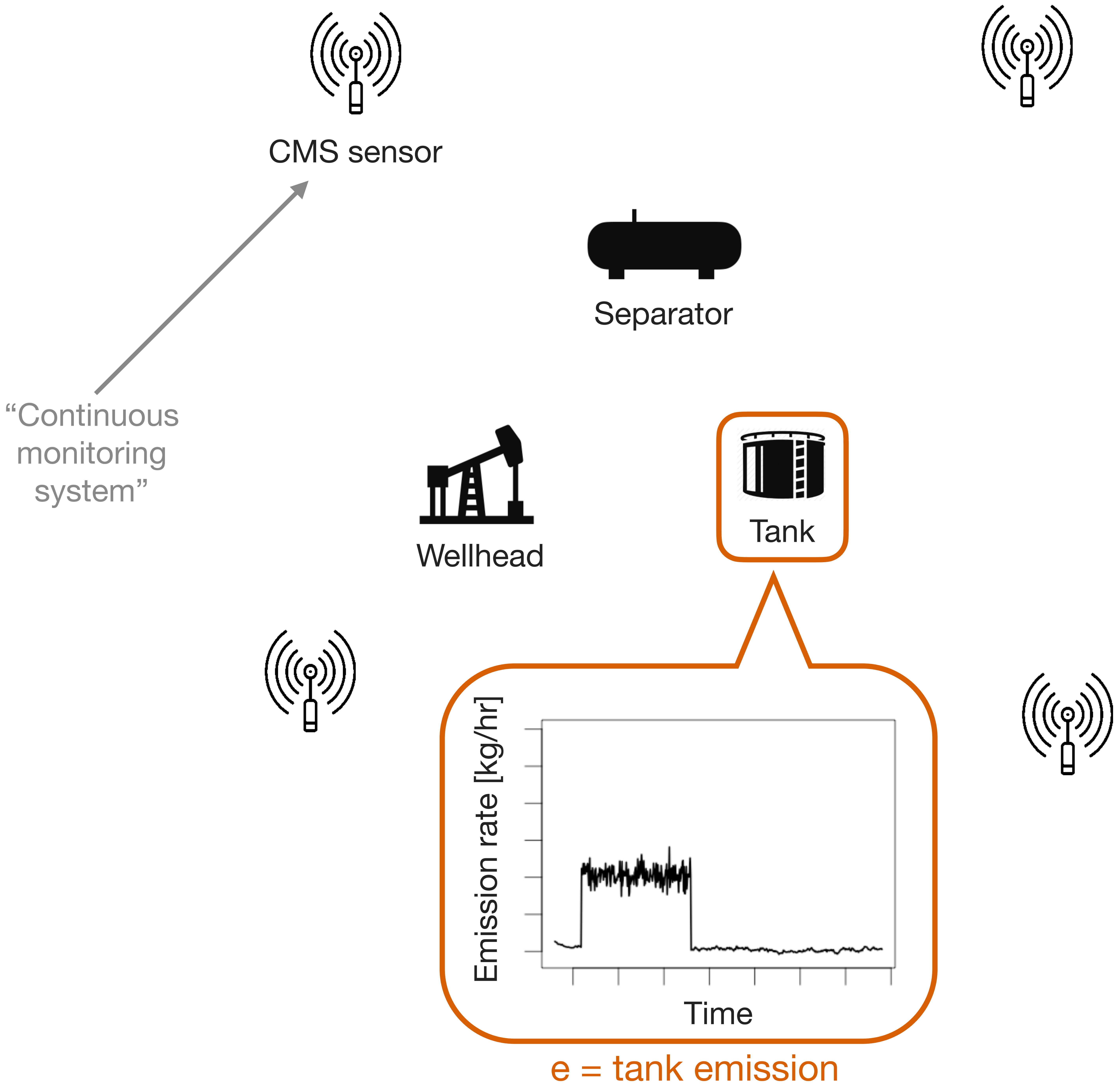
- ~~Overview of the Gaussian plume and Gaussian puff models~~
- ~~Mathematical relationship between the Gaussian plume and puff models~~
- Using the Gaussian puff model to estimate emission source and rate
- Fast implementation of the Gaussian puff model

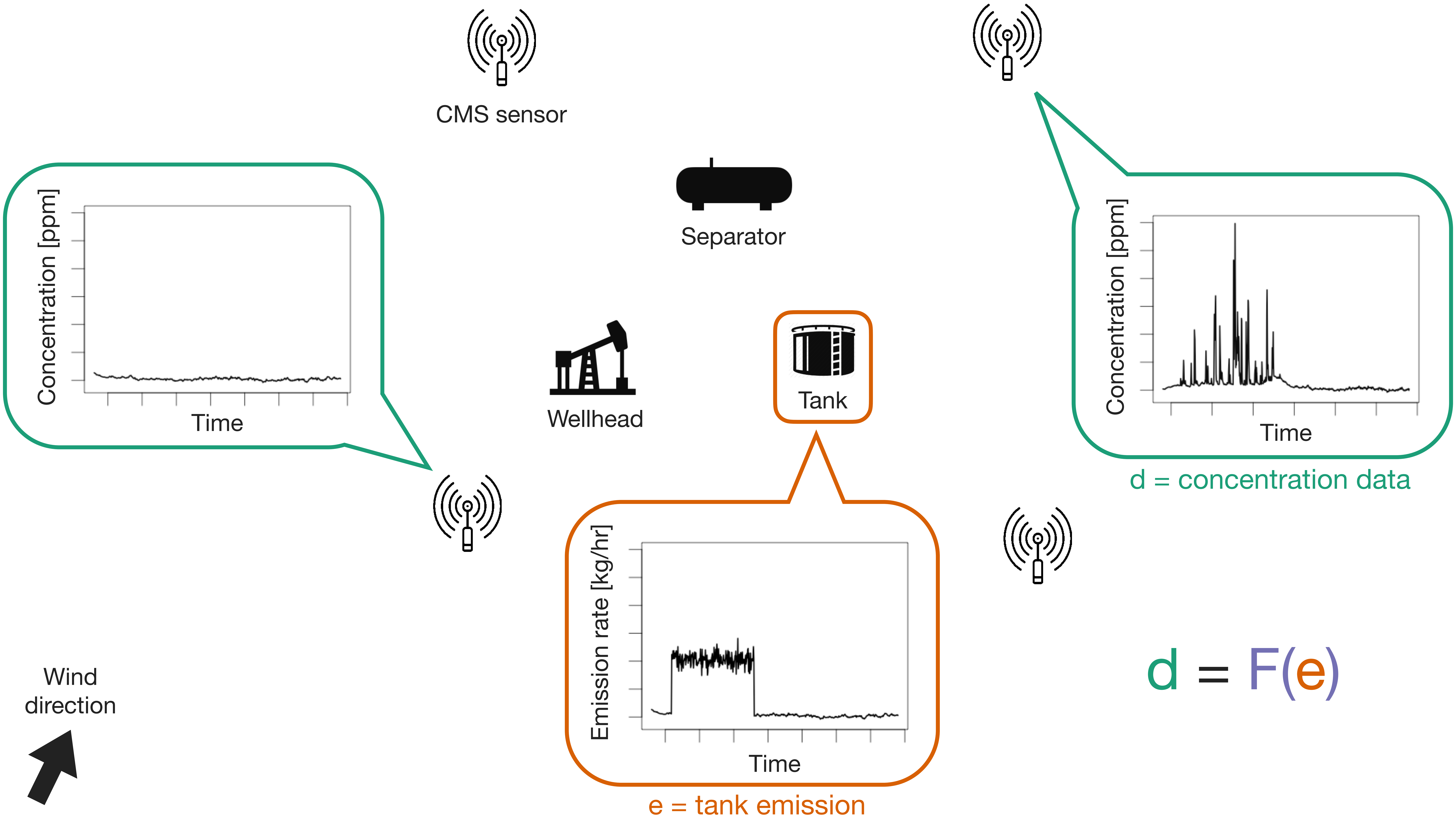
Workshop (~60 min):

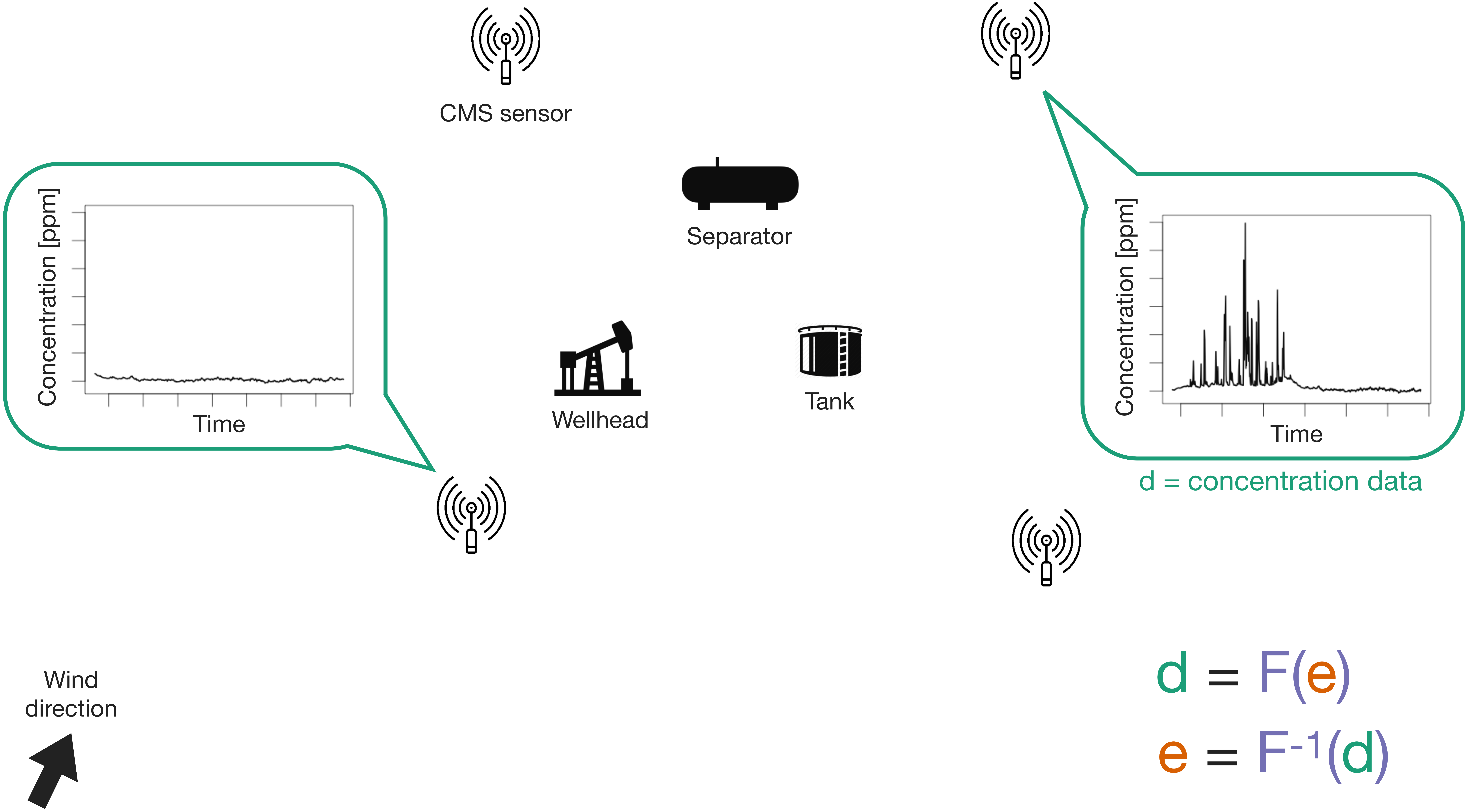
- Running the Gaussian puff
- Running the inversion to estimate source and rate
- Work time and questions



The continuous monitoring inverse problem


CMS sensor
Separator
"Continuous monitoring system"
Wellhead
Tank
Emission rate [kg/hr]
Time
e = tank emission


CMS sensor
Separator
Concentration [ppm]
Time
Wellhead
Tank
Concentration [ppm]
Time
d = concentration data
Emission rate [kg/hr]
Time
e = tank emission
Wind
direction
d = F(e)


CMS sensor
Concentration [ppm]
Time
Separator
Wellhead
Tank
Concentration [ppm]
Time
d = concentration data
Wind
direction

$$d = F(e)$$

$$e = F^{-1}(d)$$

Open source framework for solving inverse problem

STEP 1:
Background removal
and event detection

→

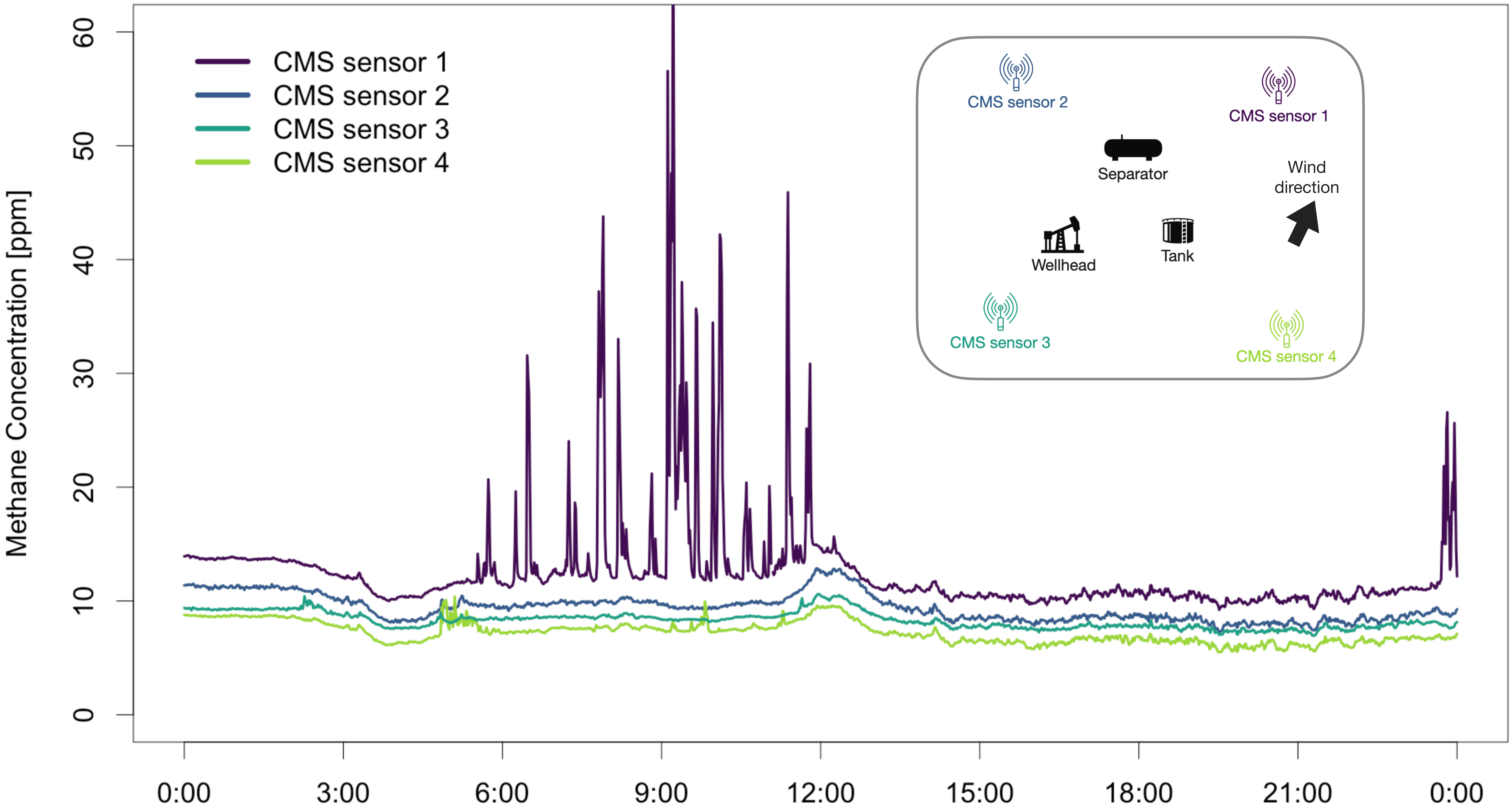
STEP 2:
Simulation

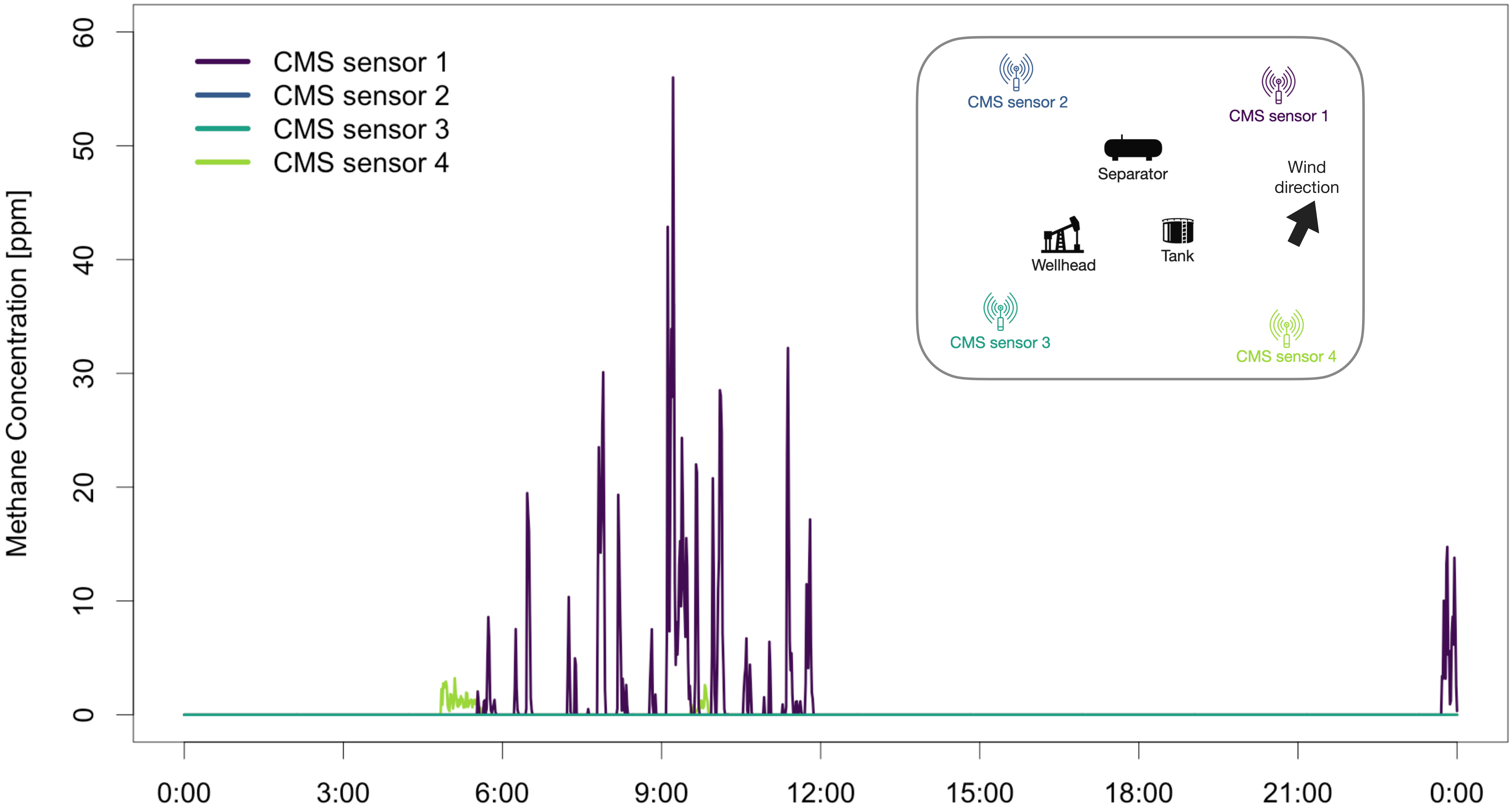
→

STEP 3:
Localization

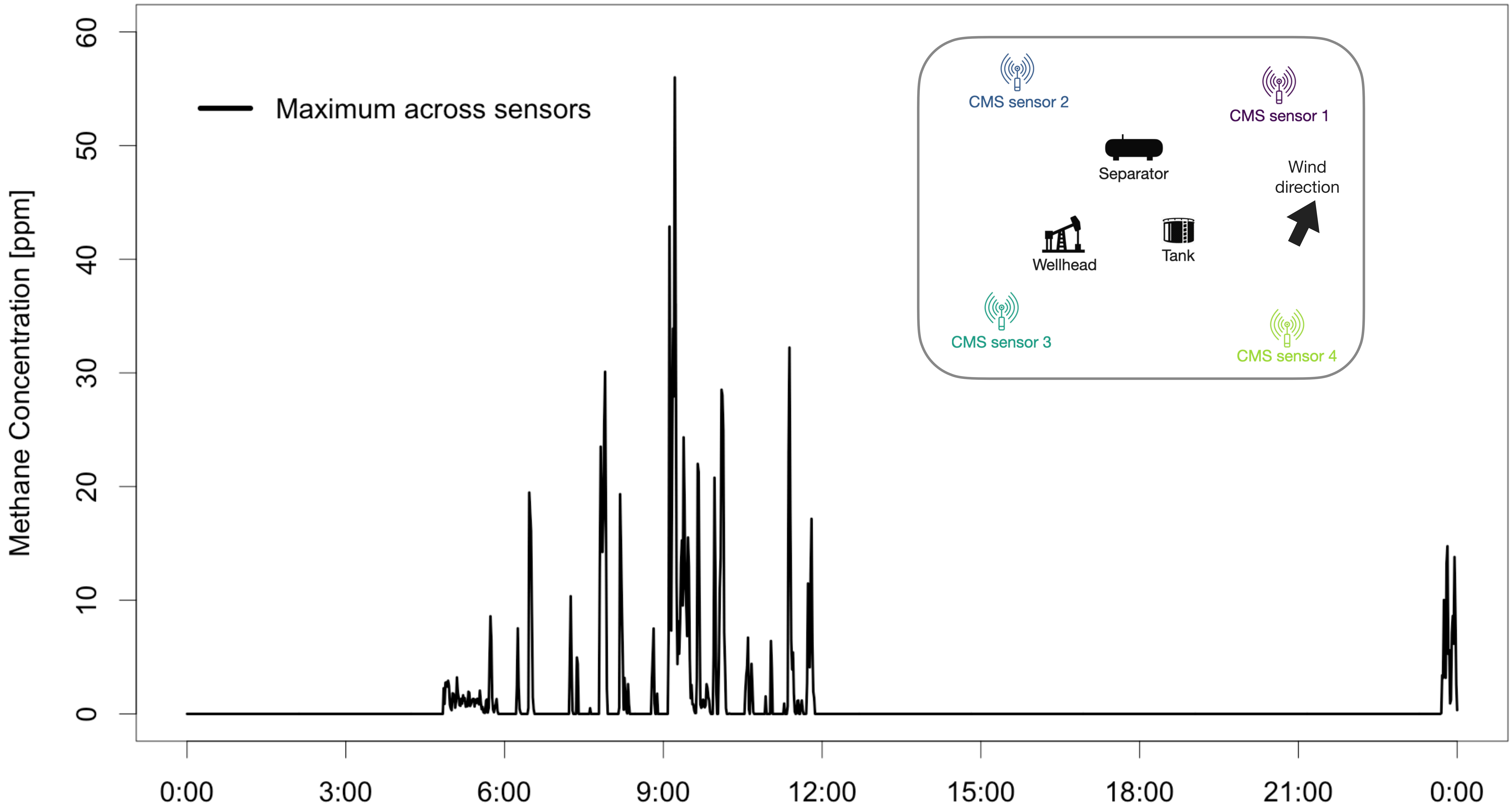
→

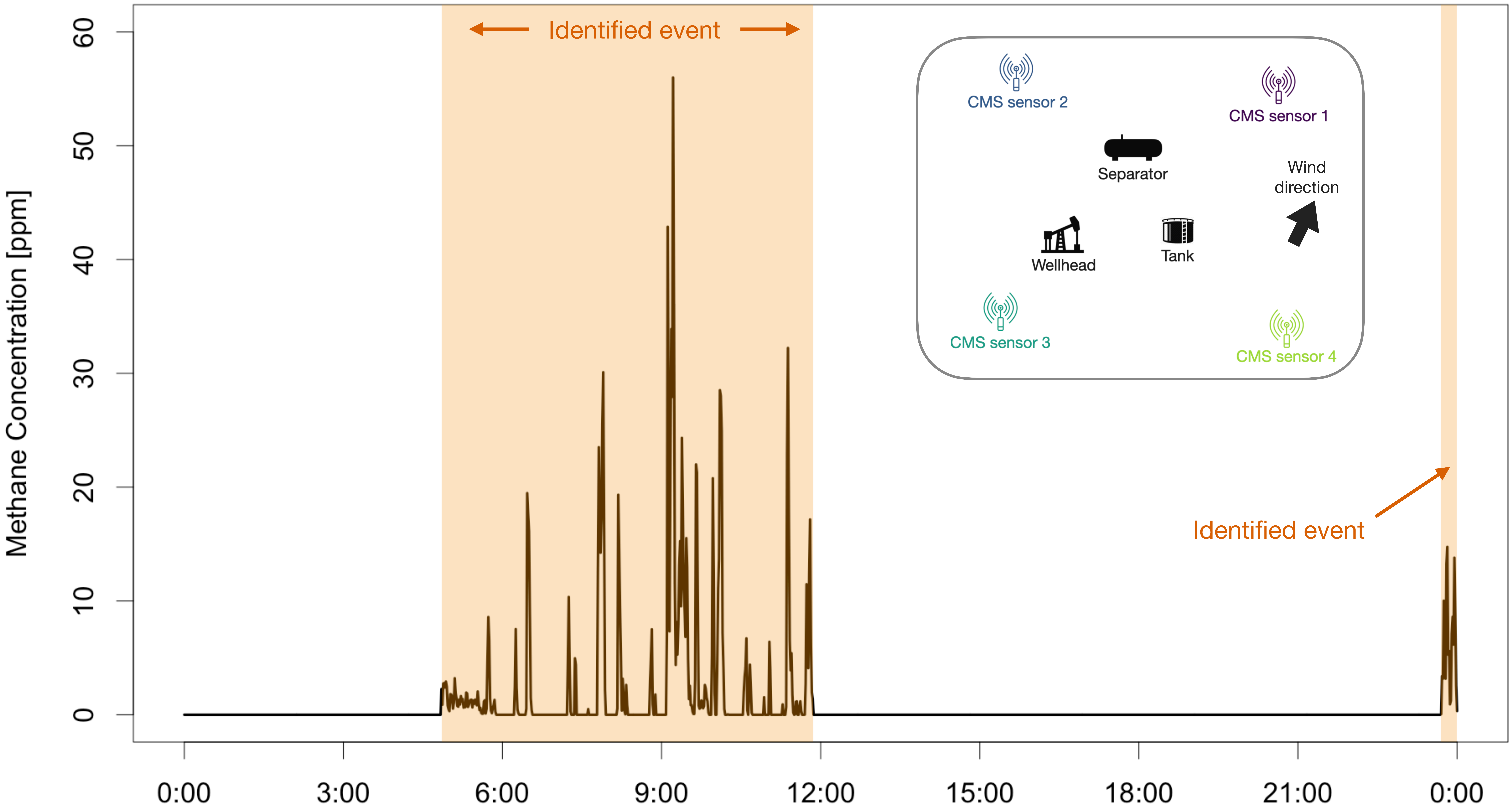
STEP 4:
Quantification


CMS sensor 1
CMS sensor 2
CMS sensor 3
CMS sensor 4
Methane Concentration [ppm]
0
10
20
30
40
50
60
0:00
3:00
6:00
9:00
12:00
15:00
18:00
21:00
0:00
CMS sensor 2
CMS sensor 1
Separator
Wind direction
Wellhead
Tank
CMS sensor 3
CMS sensor 4



Methane Concentration [ppm]
0
10
20
30
40
50
60
0:00
3:00
6:00
9:00
12:00
15:00
18:00
21:00
0:00
CMS sensor 1
CMS sensor 2
CMS sensor 3
CMS sensor 4
CMS sensor 2
CMS sensor 1
Separator
Wind direction
Wellhead
Tank
CMS sensor 3
CMS sensor 4


Maximum across sensors
CMS sensor 2
CMS sensor 1
Separator
Wind direction
Wellhead
Tank
CMS sensor 3
CMS sensor 4
Methane Concentration [ppm]
0
10
20
30
40
50
60
0:00
3:00
6:00
9:00
12:00
15:00
18:00
21:00
0:00



Identified event
CMS sensor 2
CMS sensor 1
Separator
Wind direction
Wellhead
Tank
CMS sensor 3
CMS sensor 4
Identified event
Methane Concentration [ppm]
0
10
20
30
40
50
60
0:00
3:00
6:00
9:00
12:00
15:00
18:00
21:00
0:00

Open source framework for solving inverse problem

STEP 1:
Background removal
and event detection

STEP 2:
Simulation

STEP 3:
Localization

STEP 4:
Quantification

Gaussian puff atmospheric dispersion model

Total volume of methane contained in puff p

$$c_p(x, y, z, t, Q) = \frac{Q}{(2\pi)^{3/2}\sigma_y^2\sigma_z} \exp\left(-\frac{(x-ut)^2 + y^2}{2\sigma_y^2}\right)\left[\exp\left(-\frac{(z-H)^2}{2\sigma_z^2}\right) + \exp\left(-\frac{(z+H)^2}{2\sigma_z^2}\right)\right]$$

Concentration contribution of puff p

Decay in puff concentration in horizontal plane (x, y)

Decay in puff concentration in vertical dimension (z)

Gaussian puff atmospheric dispersion model

Total volume of methane contained in puff p

Total concentration at (x, y, z, t)

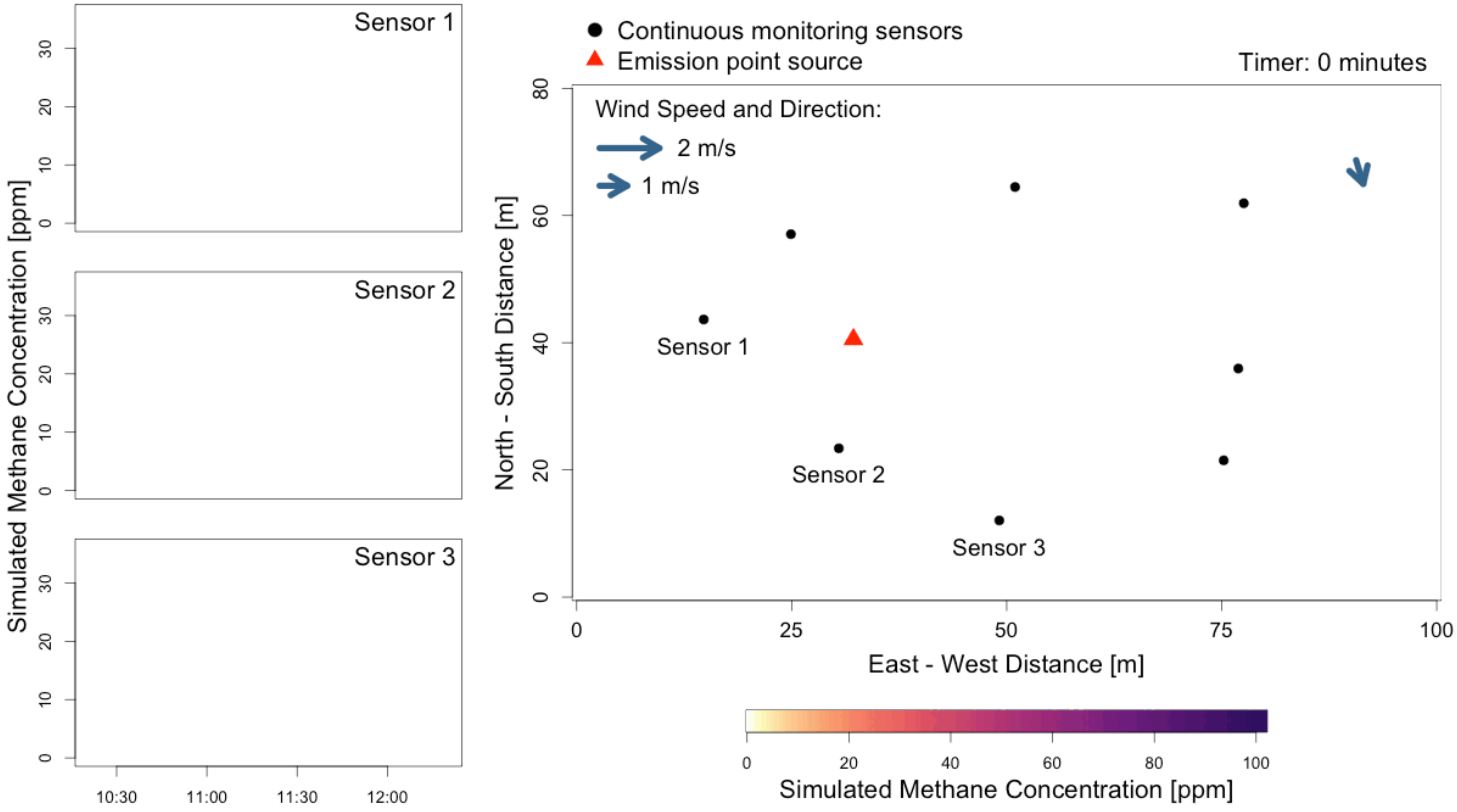
$$c(x, y, z, t, Q) = \sum_{p=1}^{P} c_p(x, y, z, t, Q)$$

$$c_p(x, y, z, t, Q) = \frac{Q}{(2\pi)^{3/2}\sigma_y^2\sigma_z} \exp\left(-\frac{(x-ut)^2 + y^2}{2\sigma_y^2}\right)\left[\exp\left(-\frac{(z-H)^2}{2\sigma_z^2}\right) + \exp\left(-\frac{(z+H)^2}{2\sigma_z^2}\right)\right]$$

Concentration contribution of puff p

Decay in puff concentration in horizontal plane (x, y)

Decay in puff concentration in vertical dimension (z)


Sensor 1
Sensor 2
Sensor 3
Simulated Methane Concentration [ppm]
0
10
20
30
10:30
11:00
11:30
12:00
Continuous monitoring sensors
Emission point source
Timer: 0 minutes
Wind Speed and Direction:
2 m/s
1 m/s
North - South Distance [m]
East - West Distance [m]
0
20
40
60
80
25
50
75
100
Simulated Methane Concentration [ppm]

Repeat this for all other potential sources!

Open source framework for solving inverse problem

STEP 1:
Background removal
and event detection

→

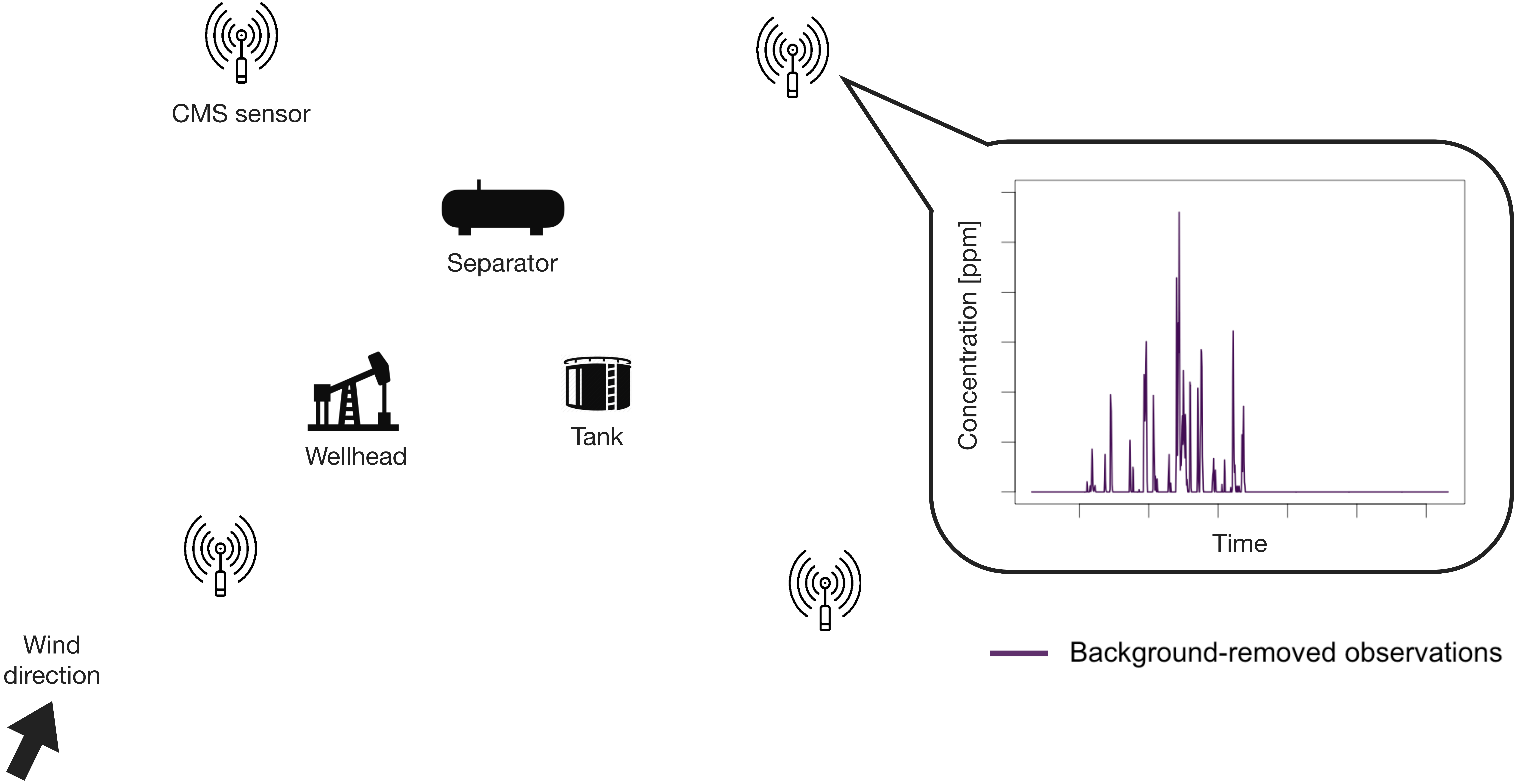
STEP 2:
Simulation

→

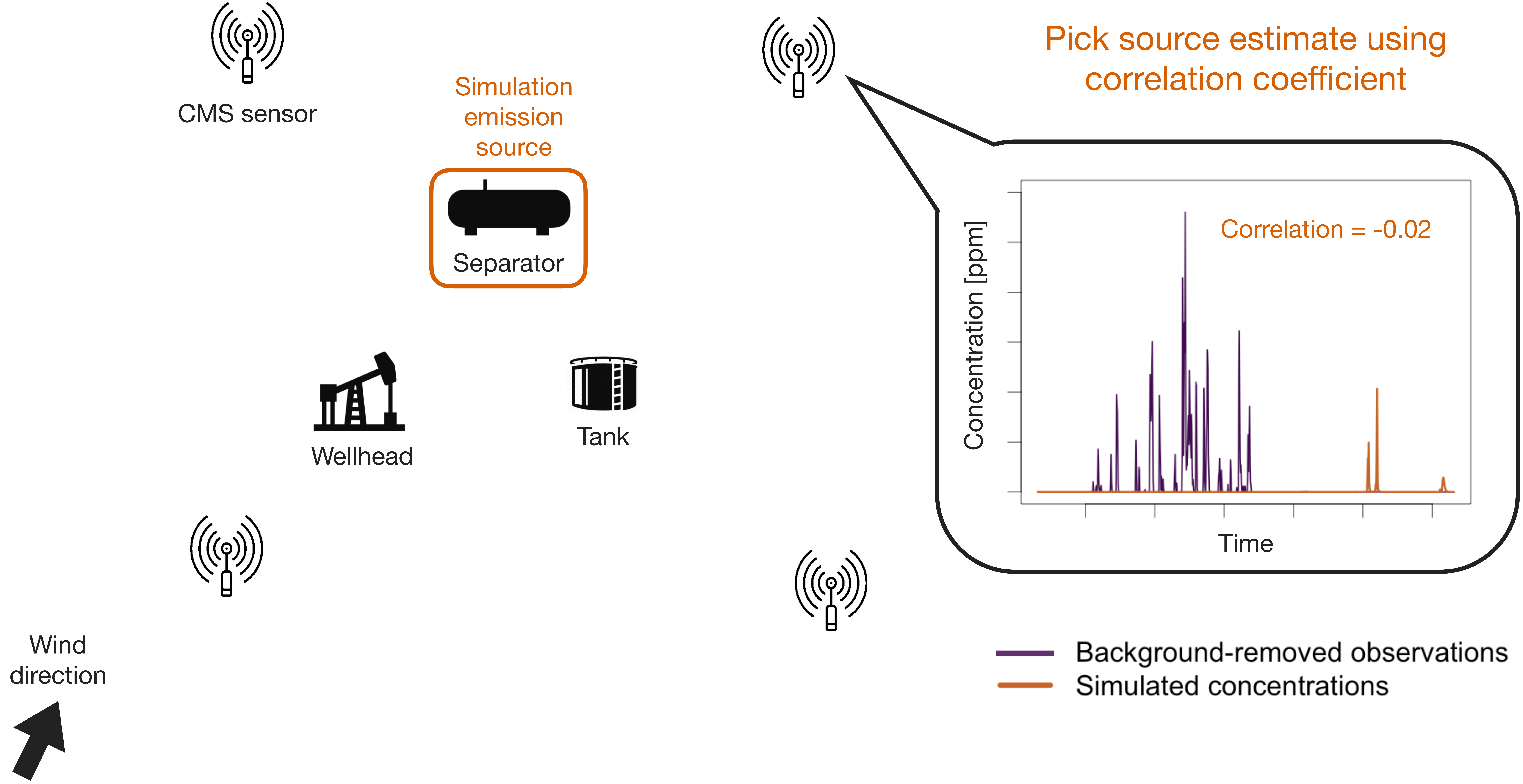
STEP 3:
Localization

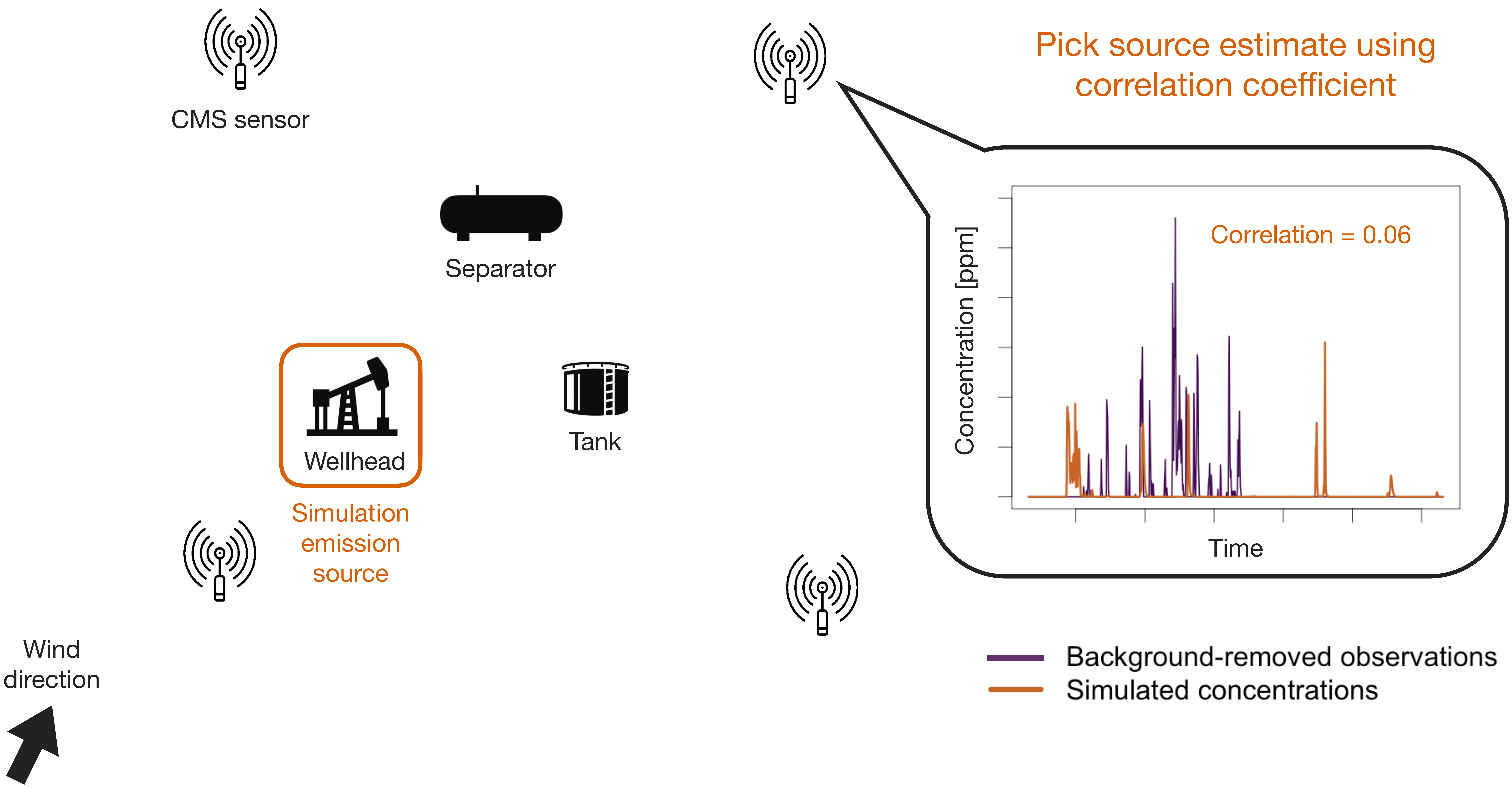
→

STEP 4:
Quantification

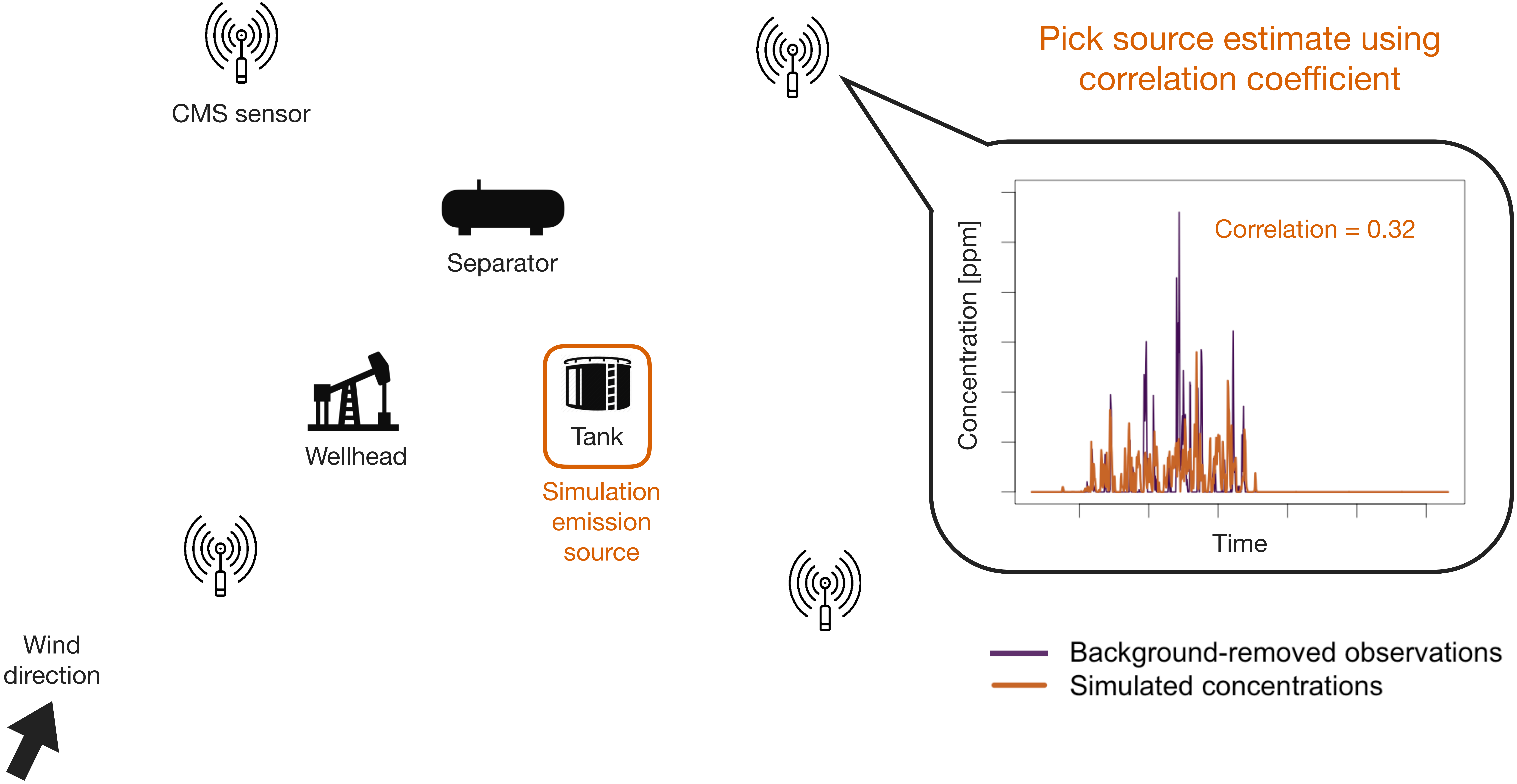

CMS sensor
Separator
Wellhead
Tank
Wind
direction
Concentration [ppm]
Time
Background-removed observations

Pick source estimate using correlation coefficient





CMS sensor
Separator
Wellhead
Tank
Simulation emission source
Wind direction
Pick source estimate using correlation coefficient
Correlation = 0.06
Concentration [ppm]
Time
Background-removed observations
Simulated concentrations



Pick source estimate using correlation coefficient
CMS sensor
Separator
Wellhead
Tank
Simulation emission source
Wind direction
Correlation = 0.32
Concentration [ppm]
Time
Background-removed observations
Simulated concentrations

Open source framework for solving inverse problem

STEP 1:
Background removal
and event detection

STEP 2:
Simulation

STEP 3:
Localization

STEP 4:
Quantification

Simulation is a linear function of emission rate

Volume of methane contained in puff p

$$c_p(x, y, z, t, Q) = Q \frac{1}{(2\pi)^{3/2}\sigma_y^2\sigma_z} \exp\left(-\frac{(x-ut)^2 + y^2}{2\sigma_y^2}\right)\left[\exp\left(-\frac{(z-H)^2}{2\sigma_z^2}\right) + \exp\left(-\frac{(z+H)^2}{2\sigma_z^2}\right)\right]$$

Concentration contribution of puff p

$$c(x, y, z, t, Q) = \sum_{p=1}^{P} c_p(x, y, z, t, Q)$$

Total concentration at (x, y, z, t)

Simulation is a linear function of emission rate

Volume of methane contained in puff p

$$c_p(x, y, z, t, Q) = Q \frac{1}{(2\pi)^{3/2}\sigma_y^2\sigma_z} \exp\left(-\frac{(x-ut)^2 + y^2}{2\sigma_y^2}\right)\left[\exp\left(-\frac{(z-H)^2}{2\sigma_z^2}\right) + \exp\left(-\frac{(z+H)^2}{2\sigma_z^2}\right)\right]$$

Concentration contribution of puff p

$$c(x, y, z, t, Q) = \sum_{p=1}^{P} c_p(x, y, z, t, Q)$$

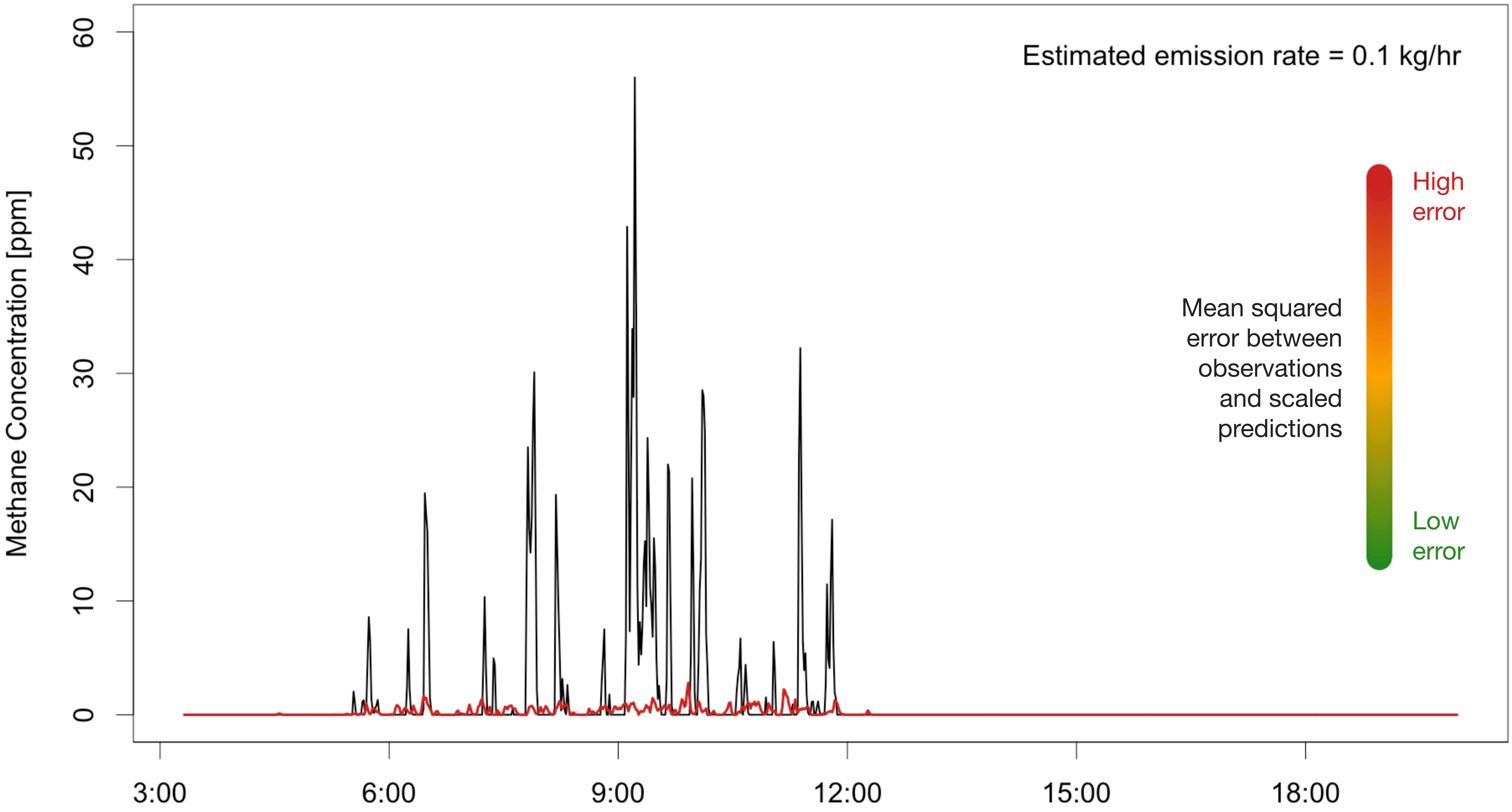
Total concentration at (x, y, z, t)

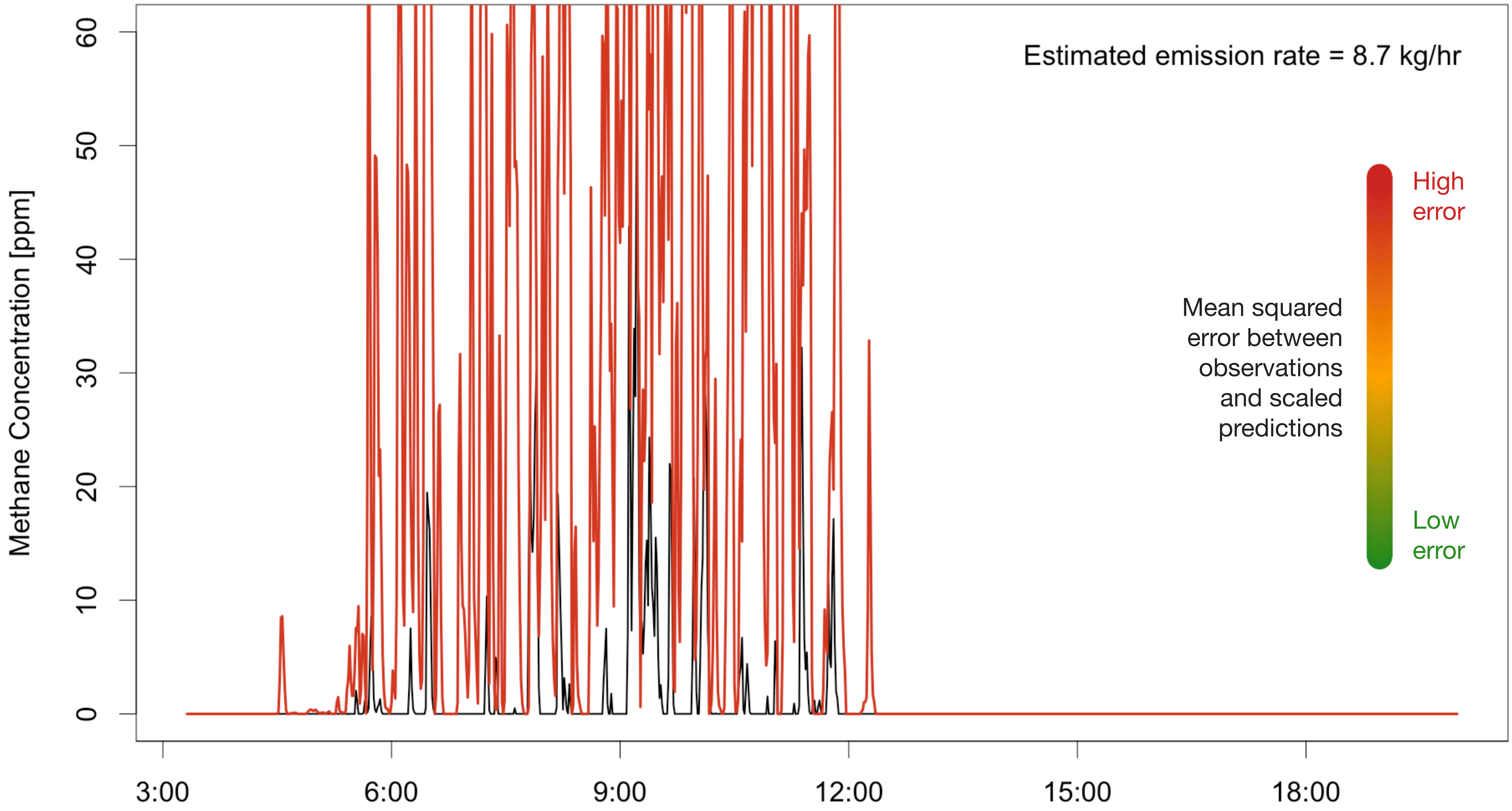
Concentration data

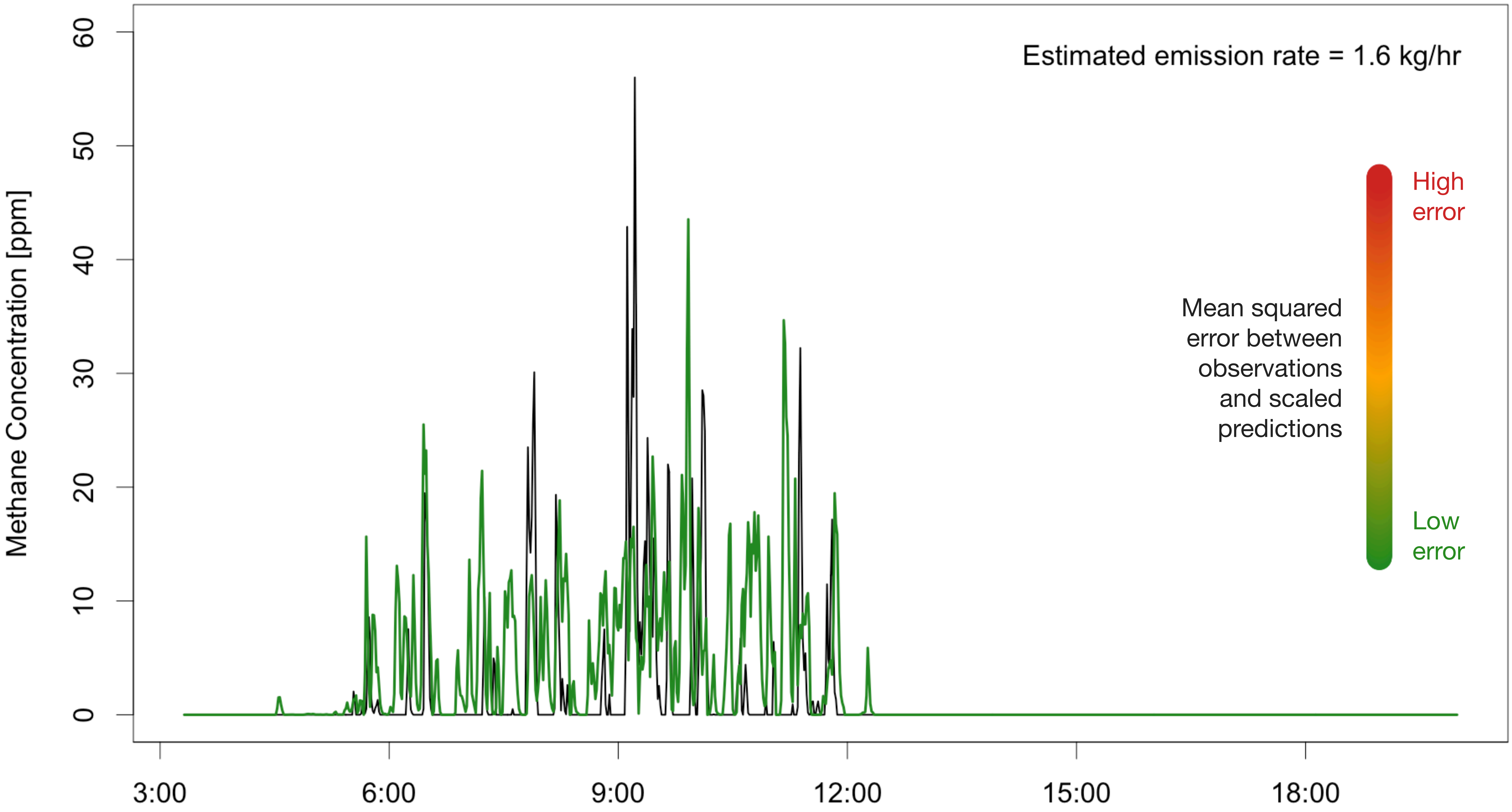
$$\hat{Q} = \underset{Q}{\operatorname{argmin}} \left\{ \frac{1}{n} \sum_{t=1}^{n} \big(d(x, y, z, t) - c(x, y, z, t, Q)\big)^2 \right\}$$

Emission rate estimate

Simulated concentrations

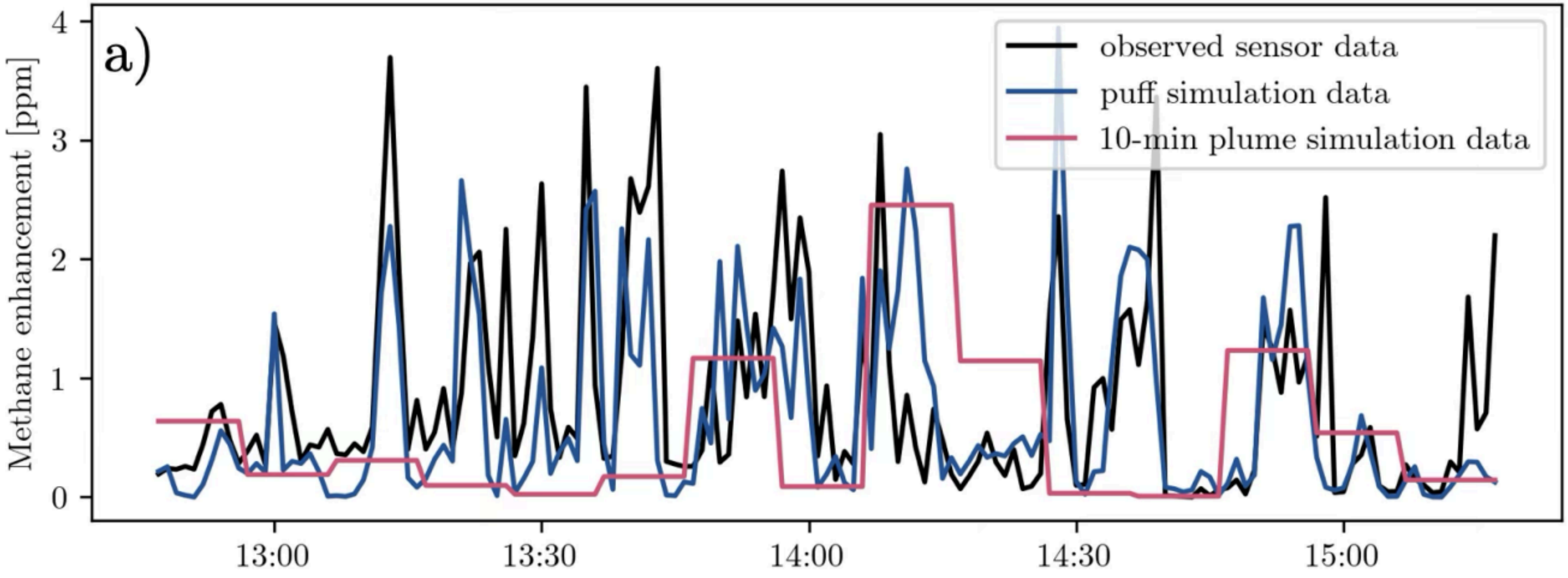

Estimated emission rate = 0.1 kg/hr
Methane Concentration [ppm]
0
10
20
30
40
50
60
3:00
6:00
9:00
12:00
15:00
18:00
Mean squared error between observations and scaled predictions
High error
Low error


Estimated emission rate = 8.7 kg/hr
Methane Concentration [ppm]
0
10
20
30
40
50
60
3:00
6:00
9:00
12:00
15:00
18:00
Mean squared error between observations and scaled predictions
High error
Low error



Estimated emission rate = 1.6 kg/hr
Methane Concentration [ppm]
0
10
20
30
40
50
60
3:00
6:00
9:00
12:00
15:00
18:00
Mean squared error between observations and scaled predictions
High error
Low error

The Gaussian puff can leverage high frequency wind data, while the Gaussian plume requires a temporal average.



Meng Jia, Ryker Fish, et al. A fast and lightweight implementation of the Gaussian puff model for near-field atmospheric transport of trace gasses. *Scientific Reports (in press)*, 2025.

Agenda

Talk (~30 min):

- ~~Overview of the Gaussian puff model~~
- ~~Relationship between the Gaussian puff and the Gaussian plume models~~
- ~~Using the Gaussian puff model to estimate emission source and rate~~
- Fast implementation of the Gaussian puff model

Workshop (~60 min):

- Code overview
- Running the Gaussian puff
- Running the inversion to estimate source and rate
- Work time and questions

FastGaussianPuff implementation

Ryker Fish and William Daniels

Gaussian Puff Model

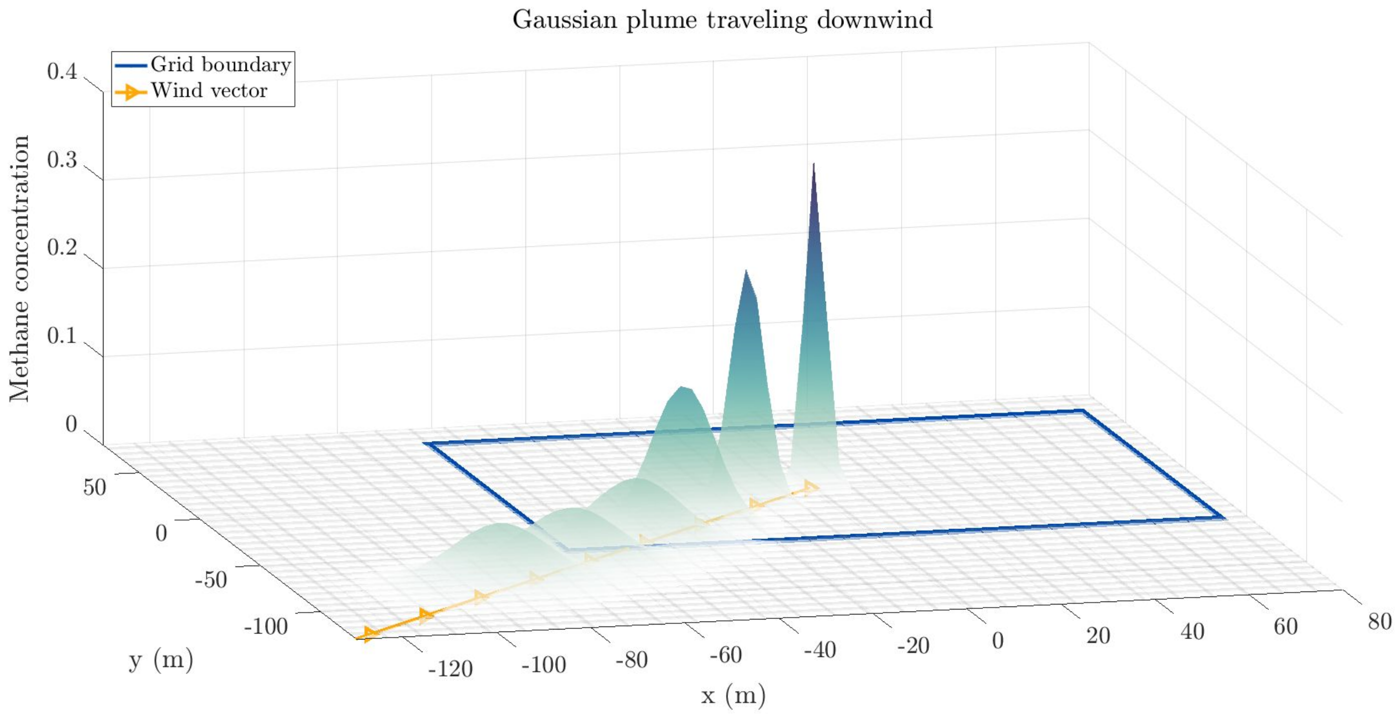
Concentration per puff:

$$c_p(x,y,z,t) = \frac{Q}{(2\pi)^{3/2}\sigma_y^2\sigma_z} \exp\left(-\frac{(x-ut)^2+y^2}{2\sigma_y^2}\right)\left[\exp\left(-\frac{(z-H_0)^2}{2\sigma_z^2}\right)+\exp\left(-\frac{(z+H_0)^2}{2\sigma_z^2}\right)\right]$$

Overall concentration:

$$c(x,y,z,t) = \sum_{p\in\mathcal{S}_t} c_p(x,y,z,t)$$

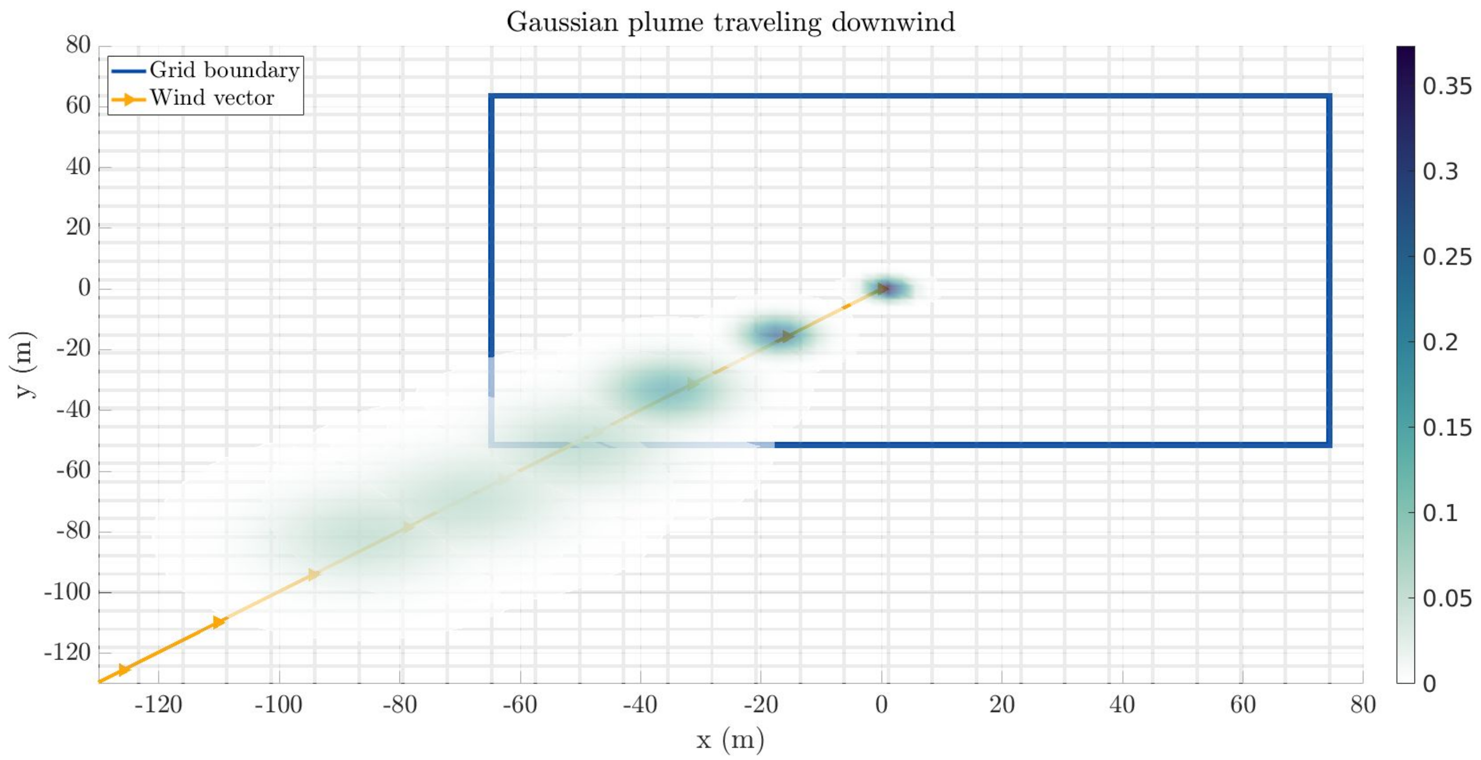
Goal: have model that can be run quickly on 3D grids or for long times on sparse sensor locations



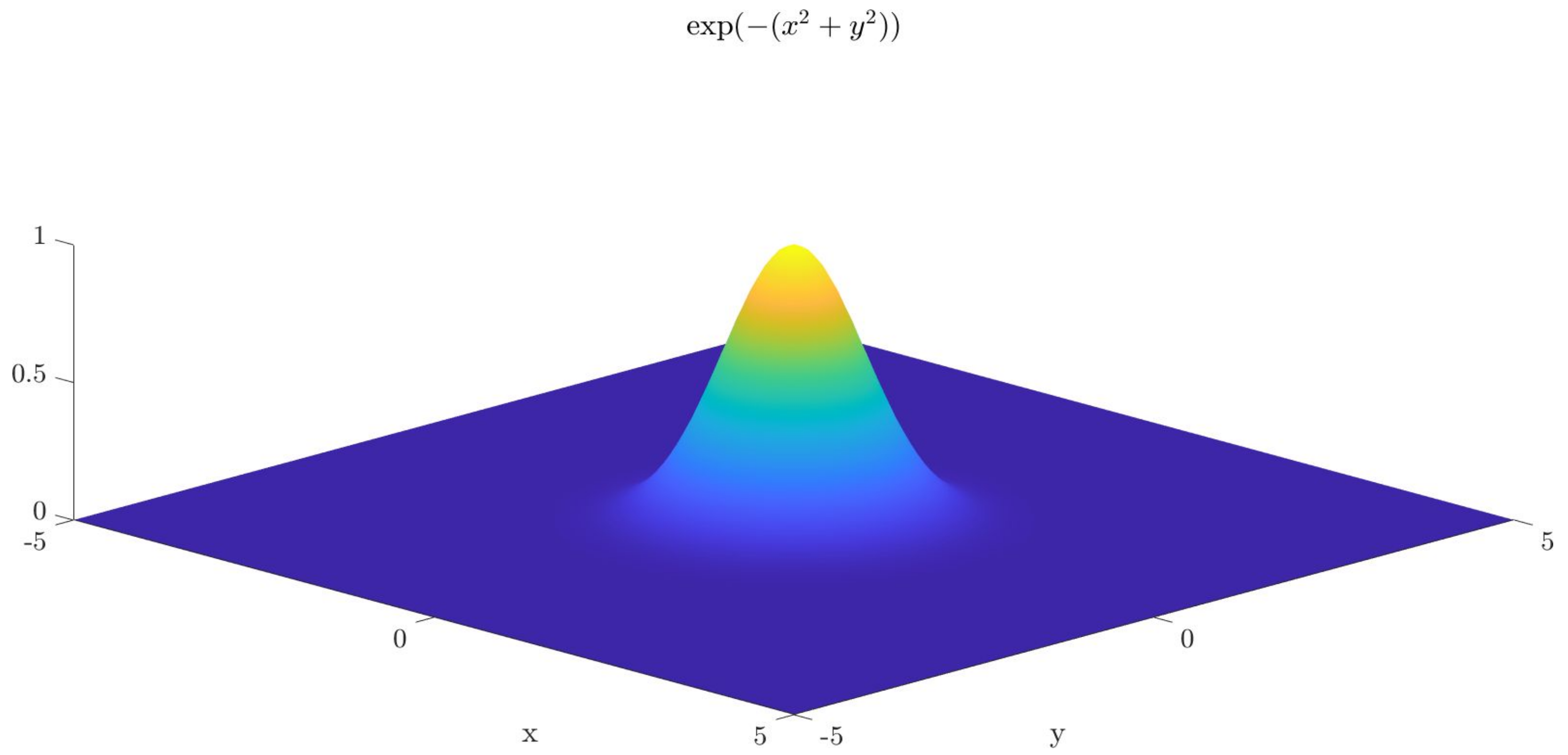
Gaussian Puff Model: Assumptions

1. Wind speed and direction are constant for each puff's lifetime

2. Diffusion coefficients are a function of downwind distance



Spatial Thresholding



- Gaussians decay quickly away from their center
- Idea: only evaluate Gaussians close to their mean
    - What counts as “close” is determined by their variance

Spatial Thresholding

$$c_p(x, y, z, t) \leq \epsilon$$

$$q_{yz} \exp\left(-\frac{(x - wt)^2}{2\sigma_y^2}\right) \exp\left(-\frac{y^2}{2\sigma_y^2}\right) \left[\exp\left(-\frac{(z - z_0)^2}{2\sigma_z^2}\right) + \exp\left(-\frac{(z + z_0)^2}{2\sigma_z^2}\right)\right] \leq \epsilon$$

$$q_{yz} \exp\left(-\frac{(x - wt)^2}{2\sigma_y^2}\right) \leq \epsilon$$

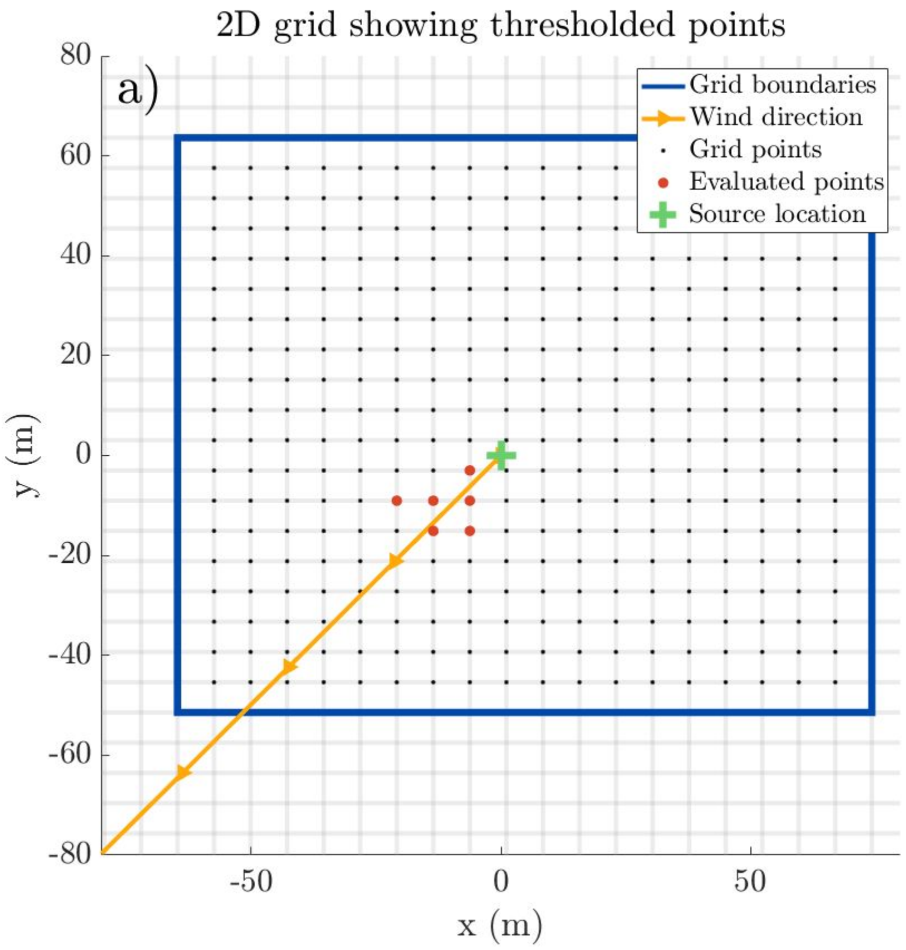
$$q_{yz} \exp\left(-\frac{y^2}{2\sigma_y^2}\right) \leq \epsilon$$

$$q_{yz} \left[\exp\left(-\frac{(z - z_0)^2}{2\sigma_z^2}\right) + \exp\left(-\frac{(z + z_0)^2}{2\sigma_z^2}\right)\right] \leq \epsilon$$

Spatial thresholding: Result

- Only points in red contribute significantly to the concentration, skip all other points

- Still need to check threshold conditions on each black point



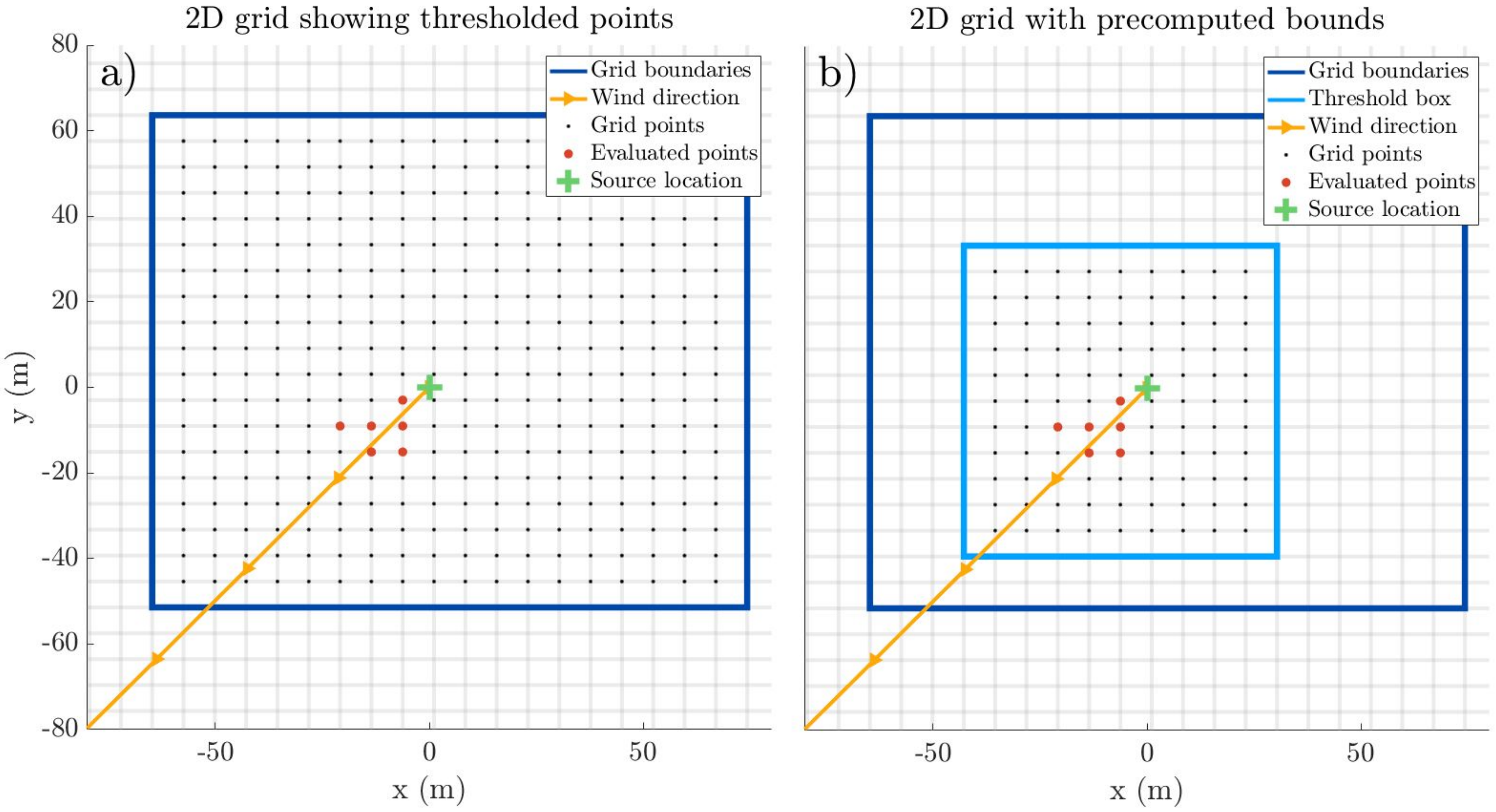
Precomputed Threshold: Result

- Solving for i, j, and k gives upper and lower limits for where to check on the grid (light blue box)

- Ex.

$$\frac{-\mathbf{x}_0^{rt} \cdot \hat{\mathbf{v}} - t_{xy}^{\max}}{\Delta x} < i < \frac{-\mathbf{x}_0^{rt} \cdot \hat{\mathbf{v}} + t_{xy}^{\max}}{\Delta x}$$

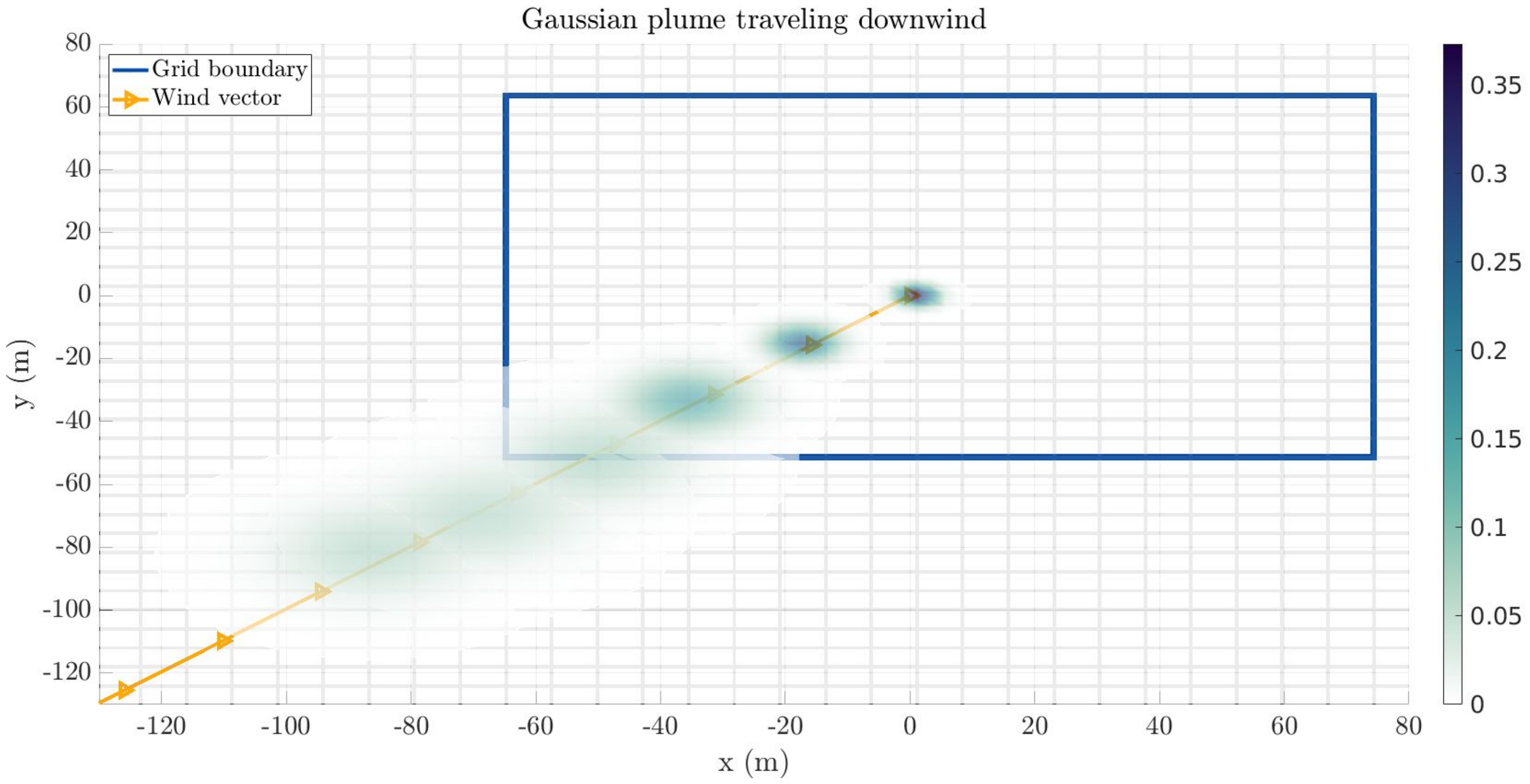
- Bounds computed with largest possible dispersion coefficients



Temporal Thresholding

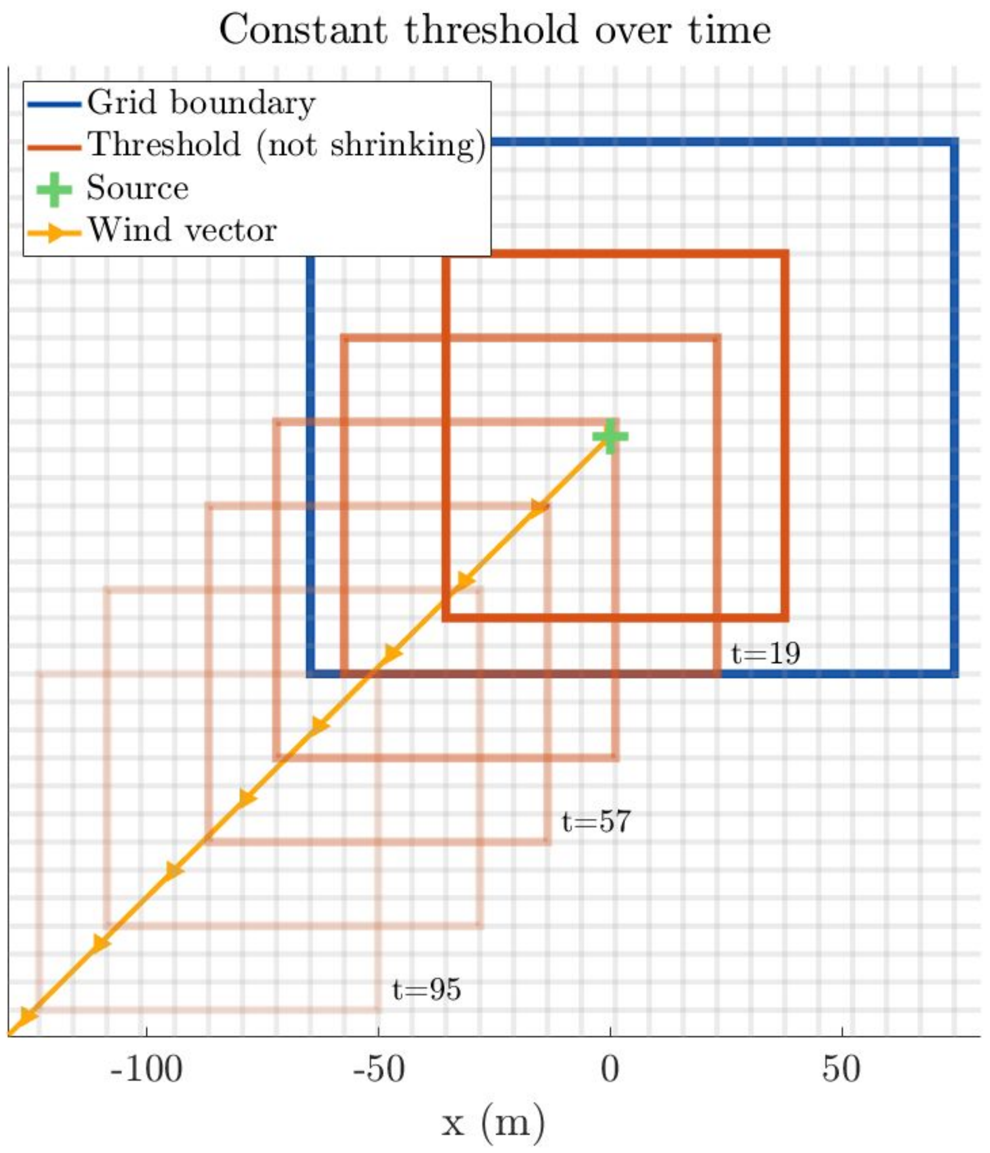
- Model assumption: wind direction and speed is constant for a puff's lifetime

- In simulation: a puff will eventually travel off grid and not return



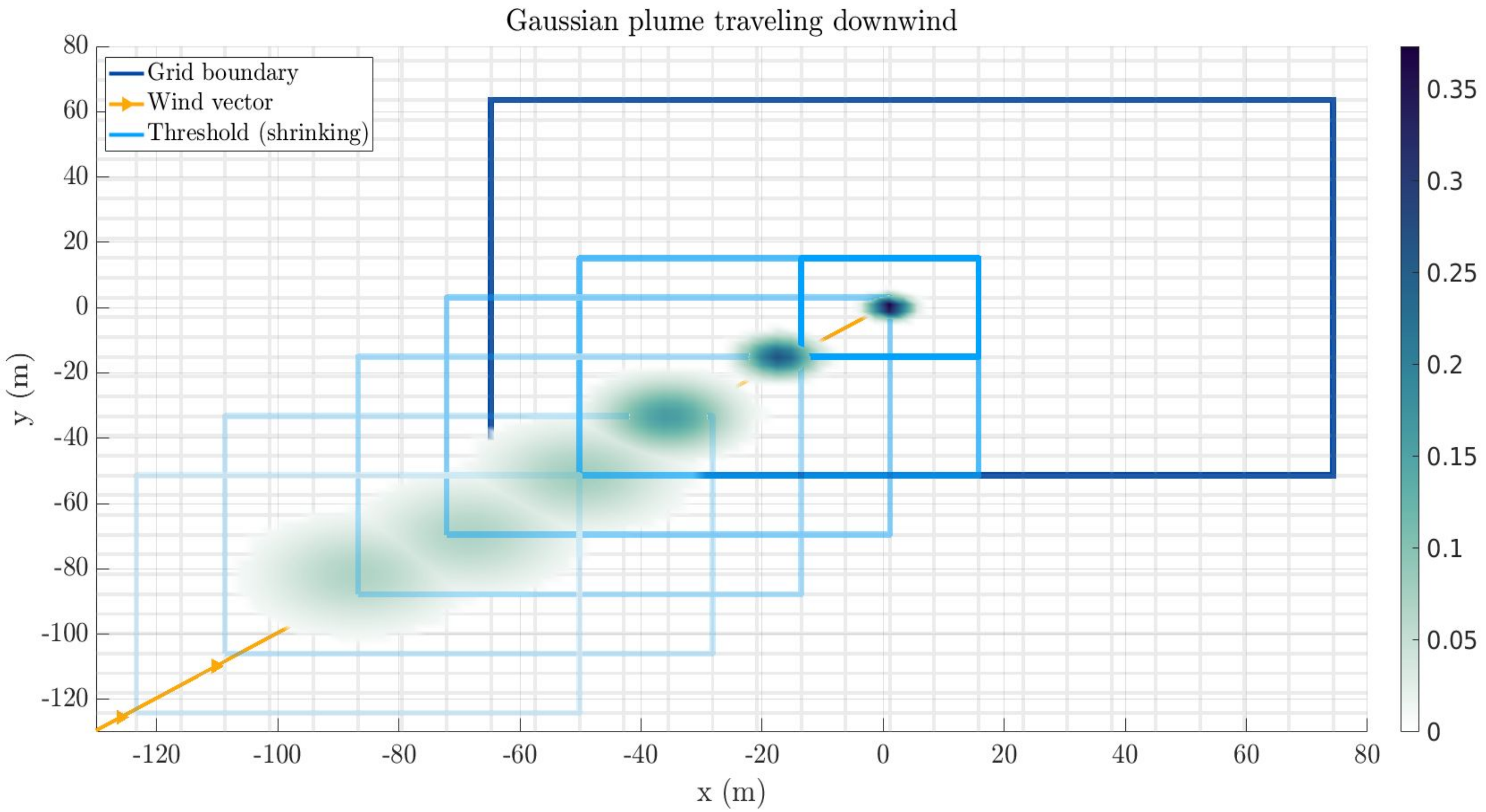
Temporal Thresholding

- Using the largest bounding box and the wind speed gives an estimate of travel time for the puff.

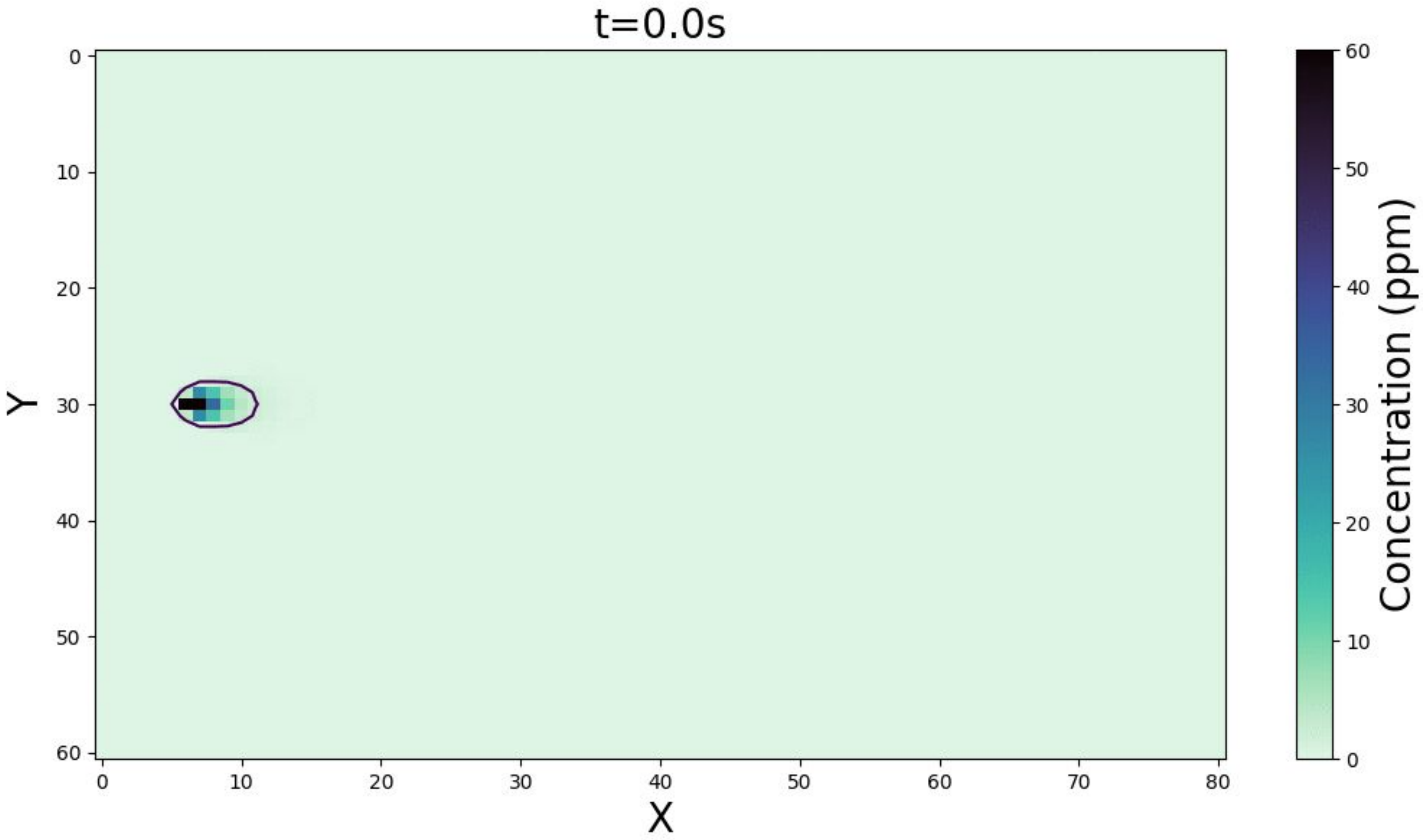


Dynamic Spatial Threshold

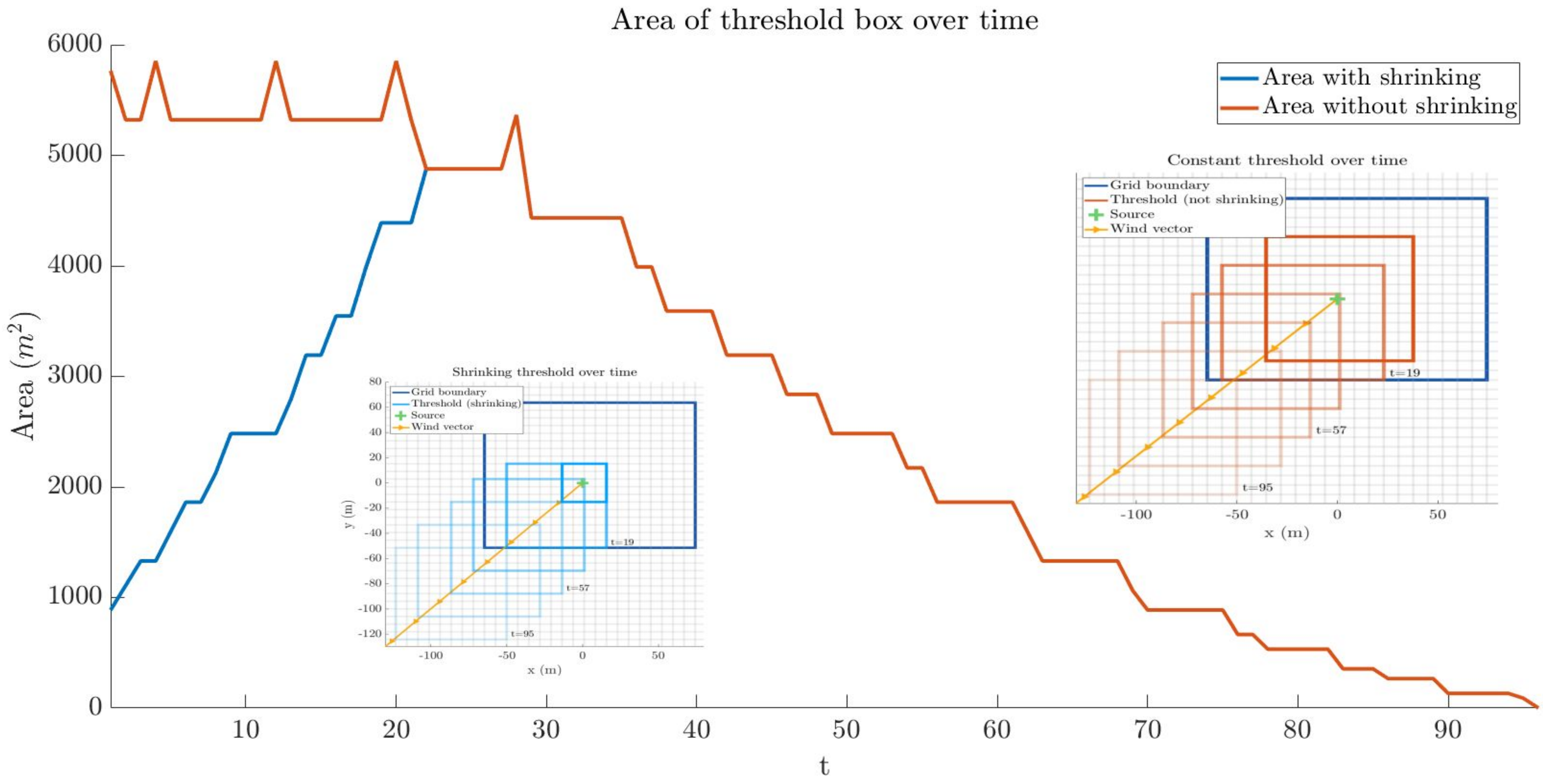
- Dispersion coefficients grow monotonically moving downwind
- Can create a smaller threshold box by carefully using smaller dispersion coefficients closer to the source



Dynamic Spatial Threshold



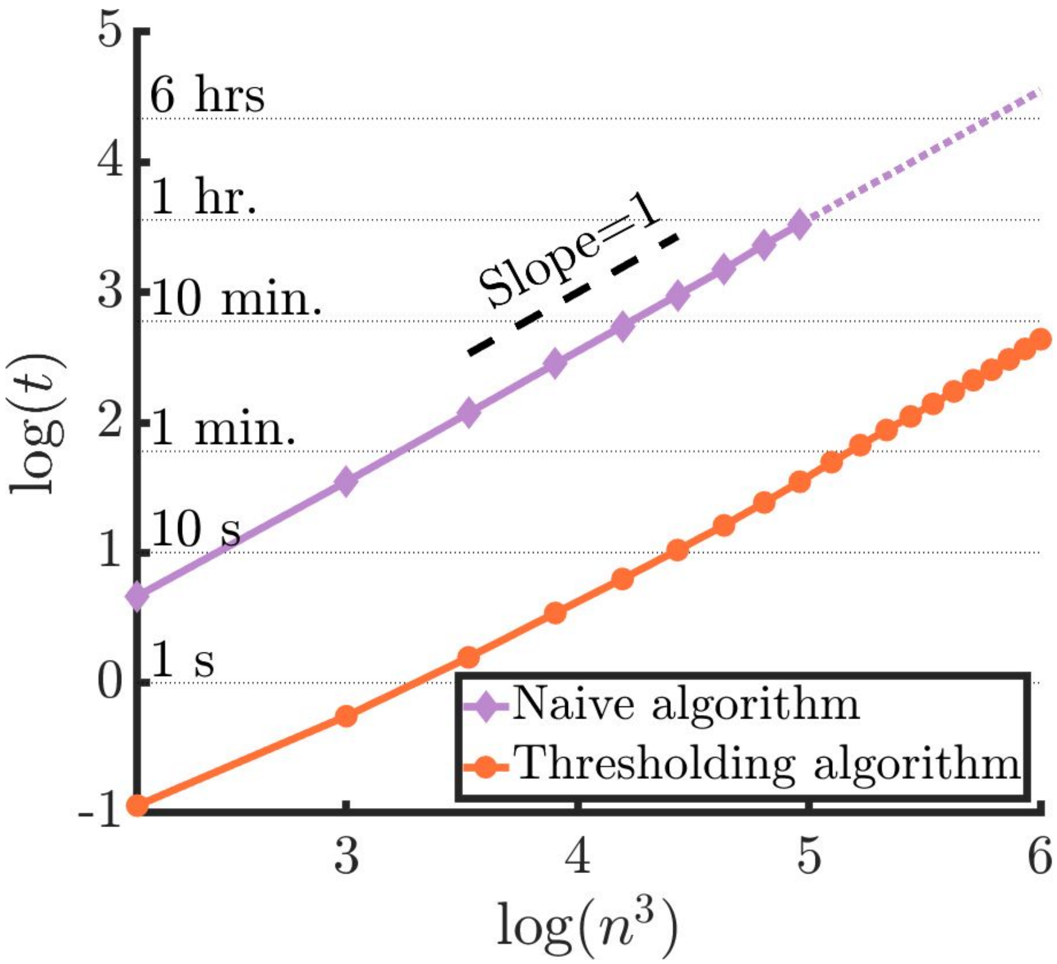
Temporal Thresholding: Shrinking

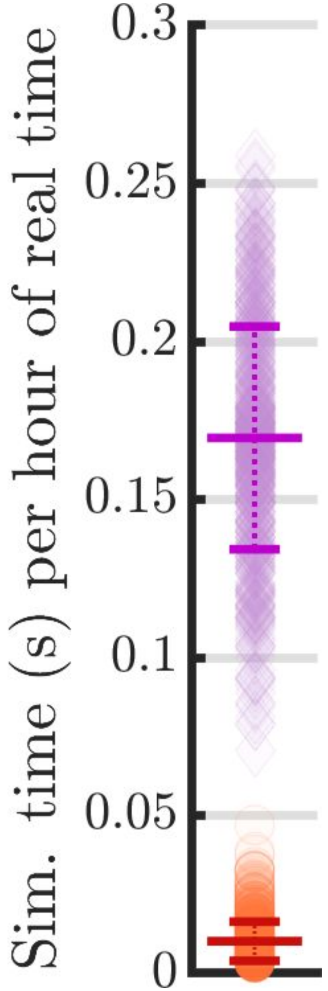


Timings

Left: runtime of the grid version versus grid size.

Right: runtime of the sensor version across the ADED 2022 campaign.

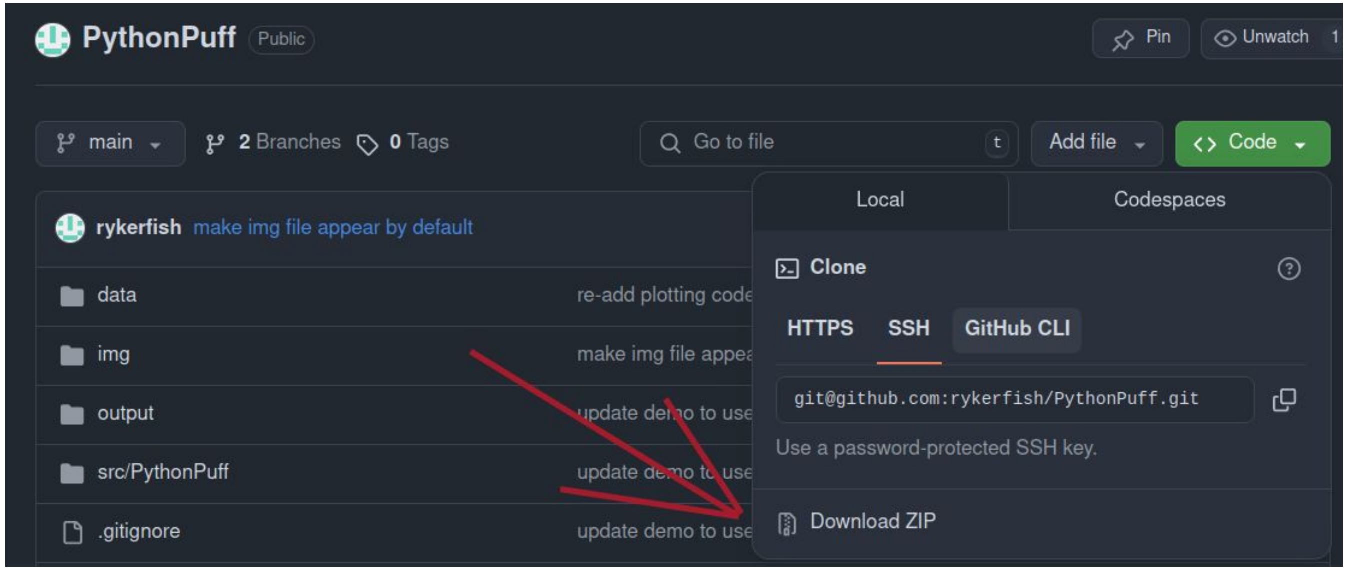




Demo

github.com/rykerfish/PythonPuff

Either clone the repository or download the ZIP file



Install

1. Open the PythonPuff folder in a VSCode, a terminal, or your preferred editor

2. Create and activate conda environment:
    a. *conda env create -f environment.yml*
    b. *conda activate pygp*

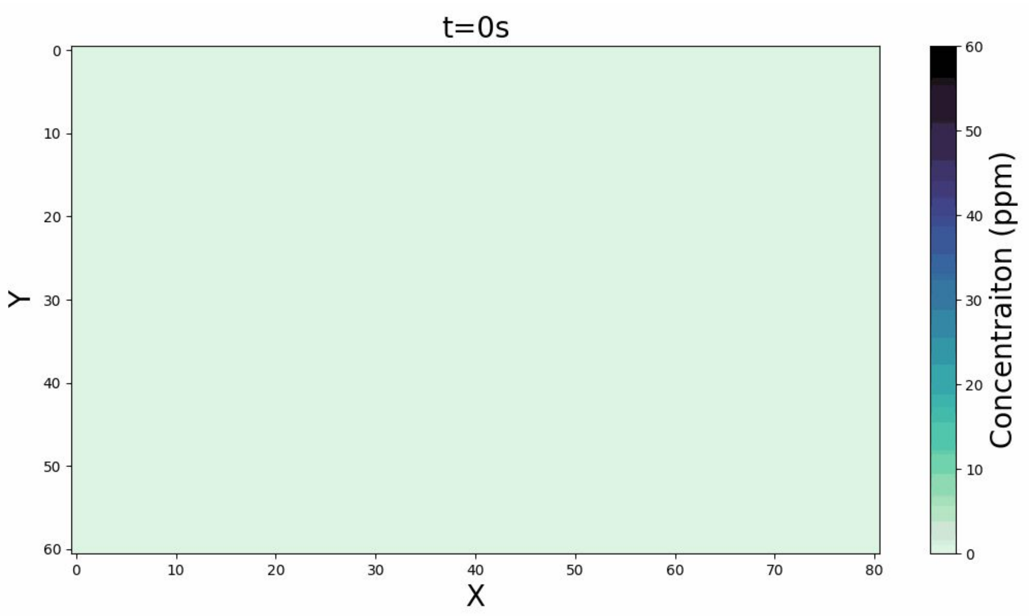
Alternative: install **pandas, numpy, numba,** and **matplotlib** using pip

3. Install the code so it is available everywhere in the conda environment
    a. *pip install .* (from inside the PythonPuff directory)

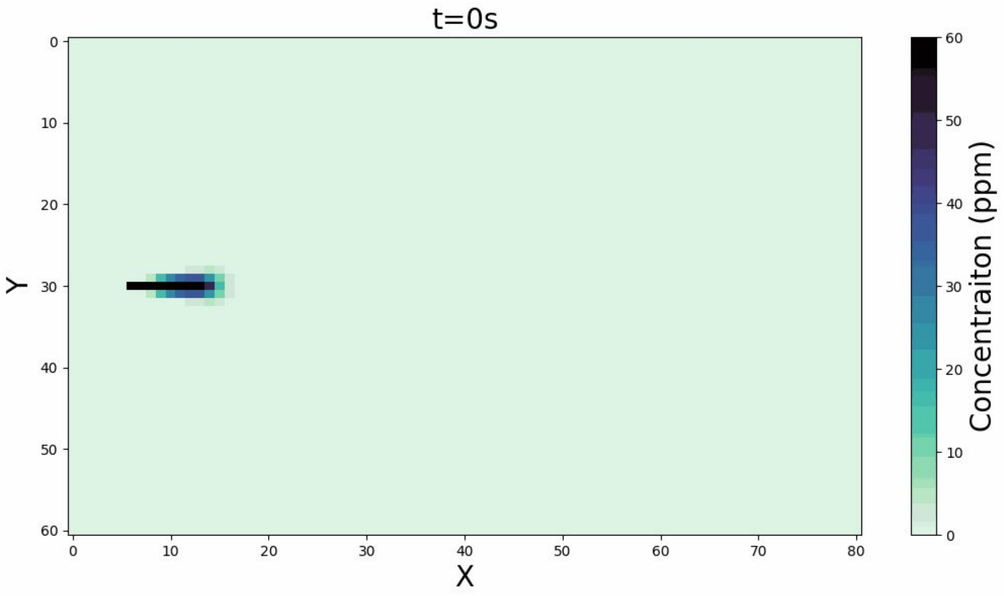
4. Run the demo!
    a. *python demo.py*

Timestep parameters

Simulation timestep, **sim_dt:** how frequently the simulation evaluates a puff so that it travels smoothly



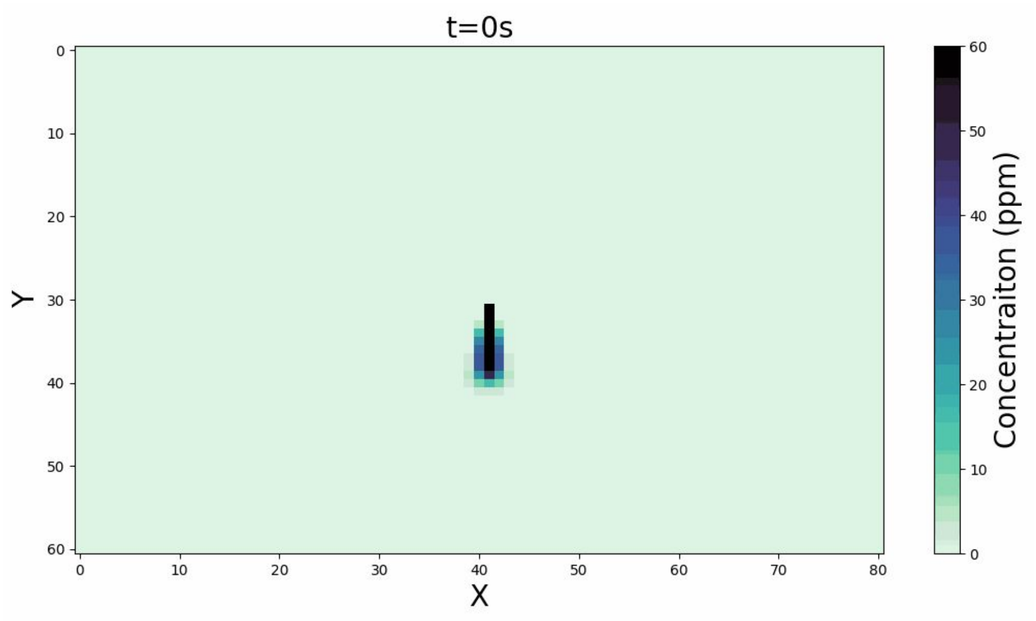
sim_dt being too low causes skipping



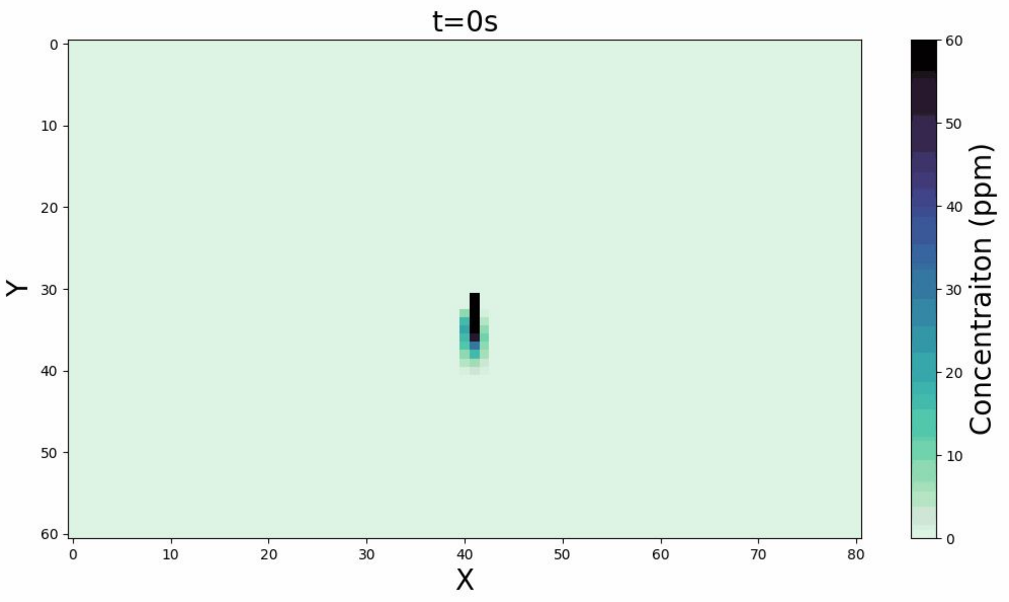
When sim_dt is right, resulting plume will be smooth

Timestep parameters

Puff creation timestep, **puff_dt:** how frequently puffs are created so that a plume is well-resolved.



puff_dt being too low creates a patchy plume



When sim_dt is right, resulting plume will be smooth

Gaussian puff code:

https://github.com/rykerfish/PythonPuff

Inversion code:

https://github.com/wsdaniels/DLQ

Thank you!

COLORADO SCHOOL OF MINES

EEMDL
Energy Emissions Modeling and Data Lab

U.S. DEPARTMENT OF ENERGY