A Gaussian Process Based Technique of Efficient Sensor Selection for Transmitter Localization

Aritrik Ghosh, Arani Bhattacharya

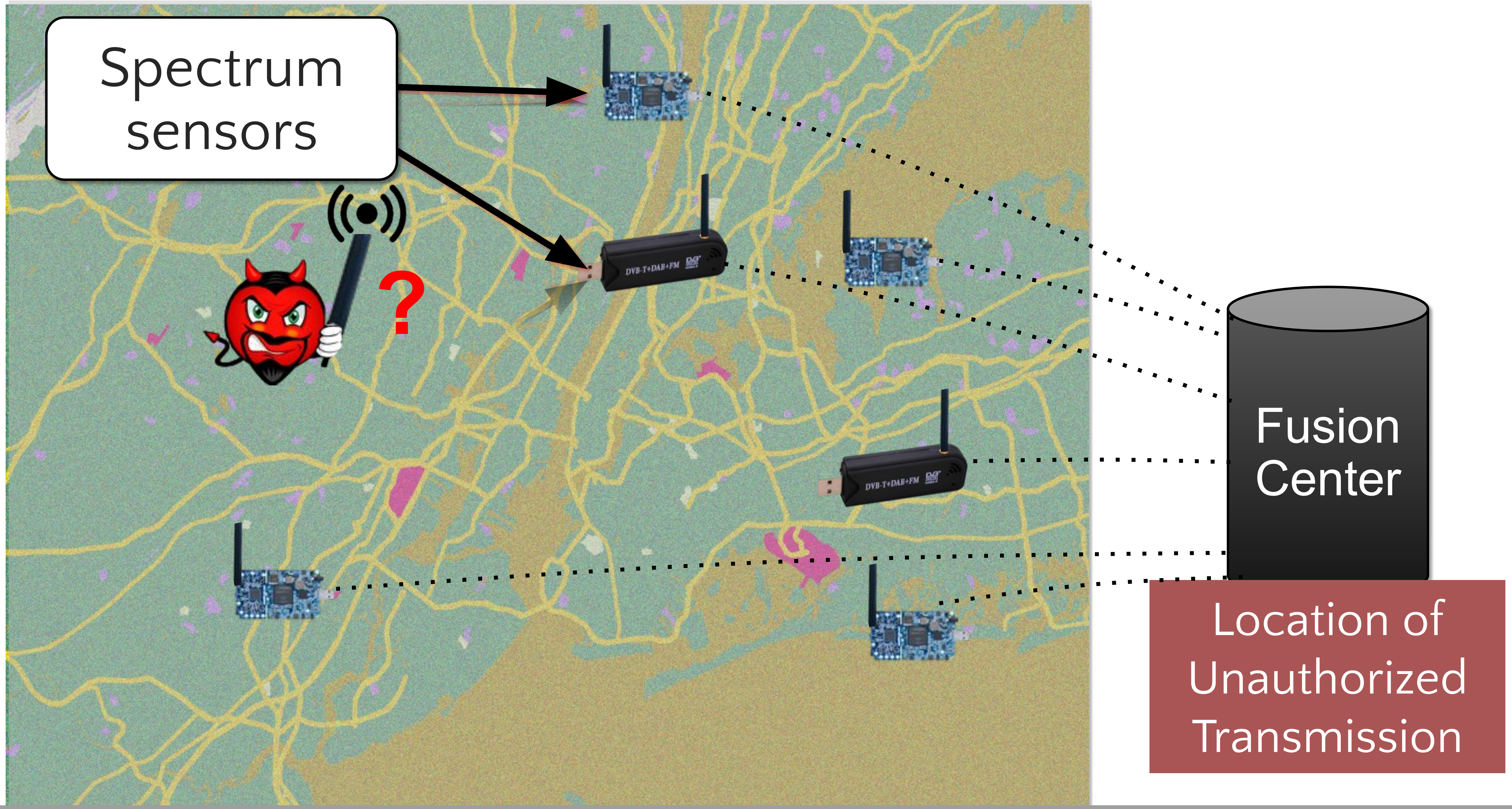
Content

- Motivation for sensor selection
- Problem formulation
- Algorithms used
- Evaluation

Rising Threat of Unauthorized Transmission

How can we monitor spectrum?

A Distributed Spectrum Monitoring System



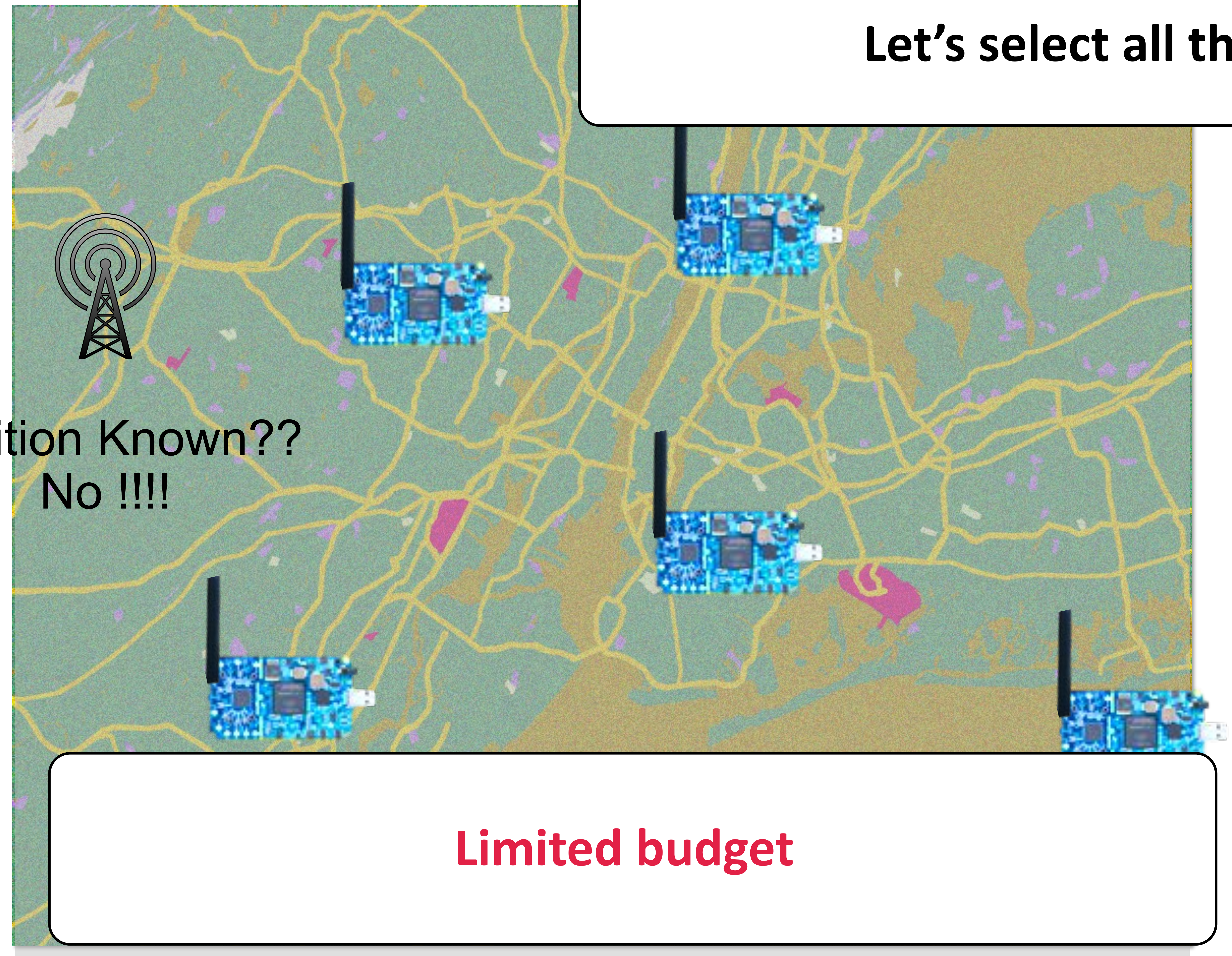
Deploy large number of cheap but noisy spectrum sensors; utilize robust localization to reduce impact of noise

Content

- Motivation for sensor selection
- Problem formulation
- Algorithms used
- Evaluation

Sensor Selection

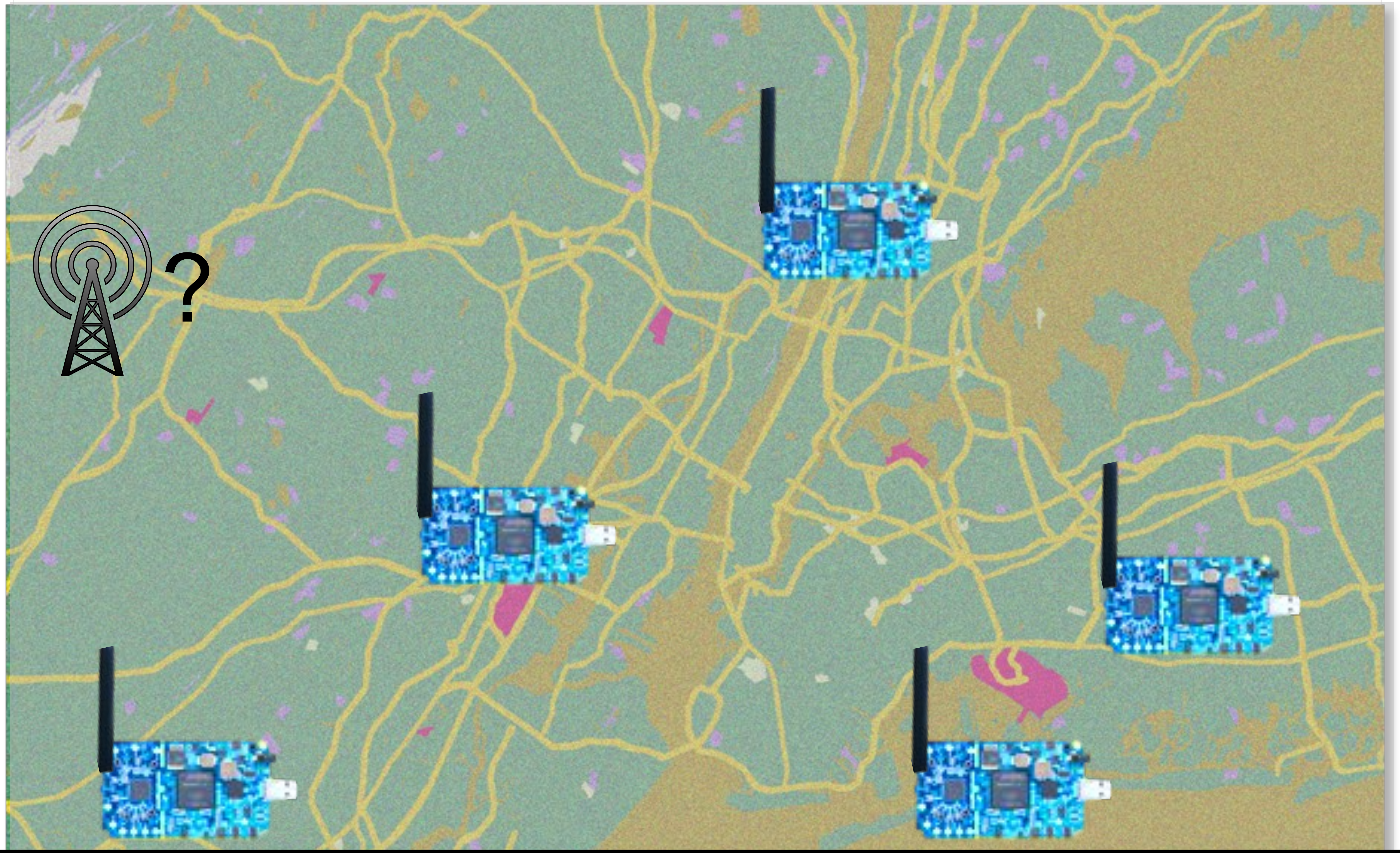
Let's select all the sensors



Position Known??
No !!!!

Limited budget

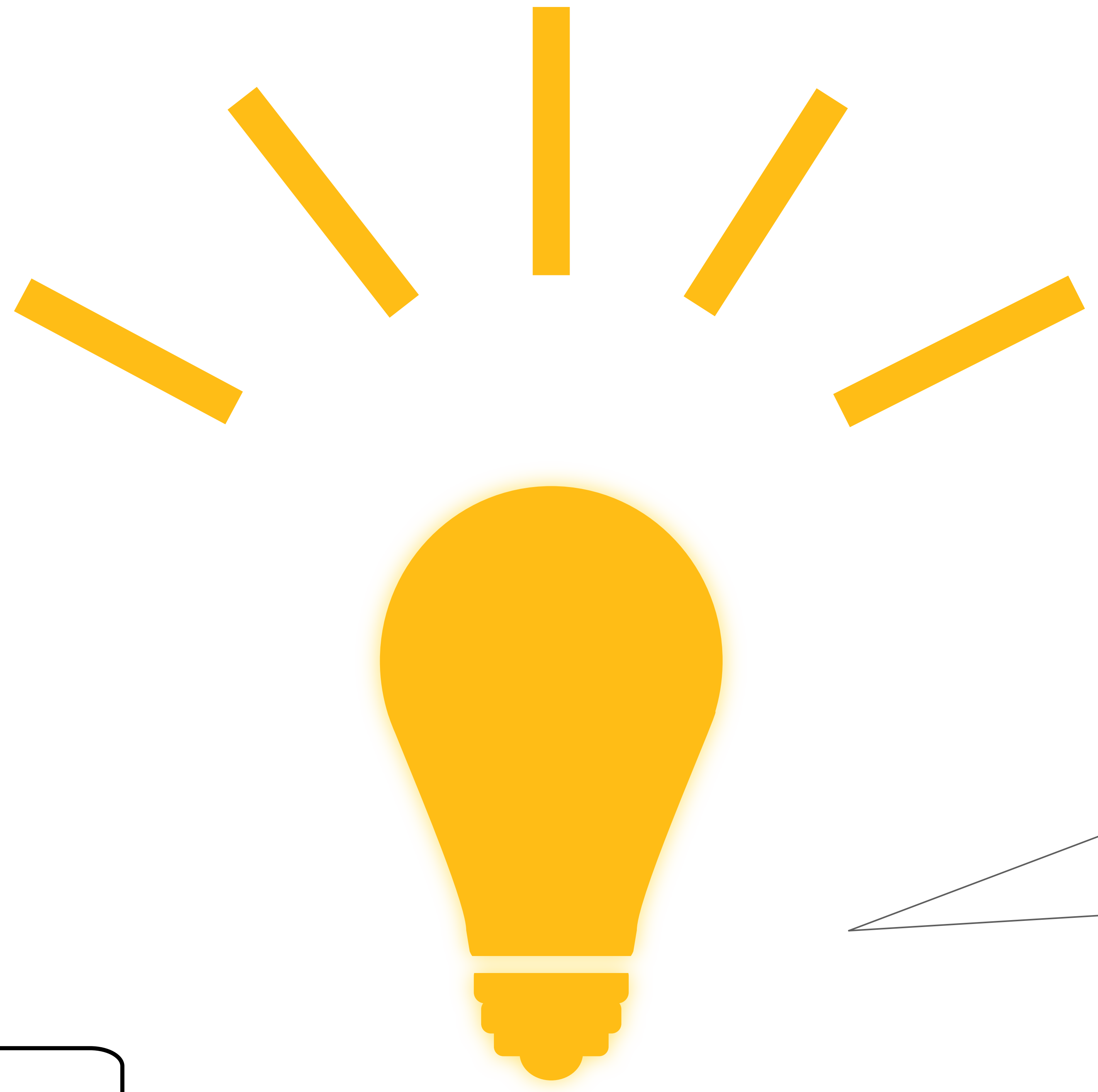
A optimization problem



Maximum Accuracy of localization is subject to # Sensors ≤ Budget

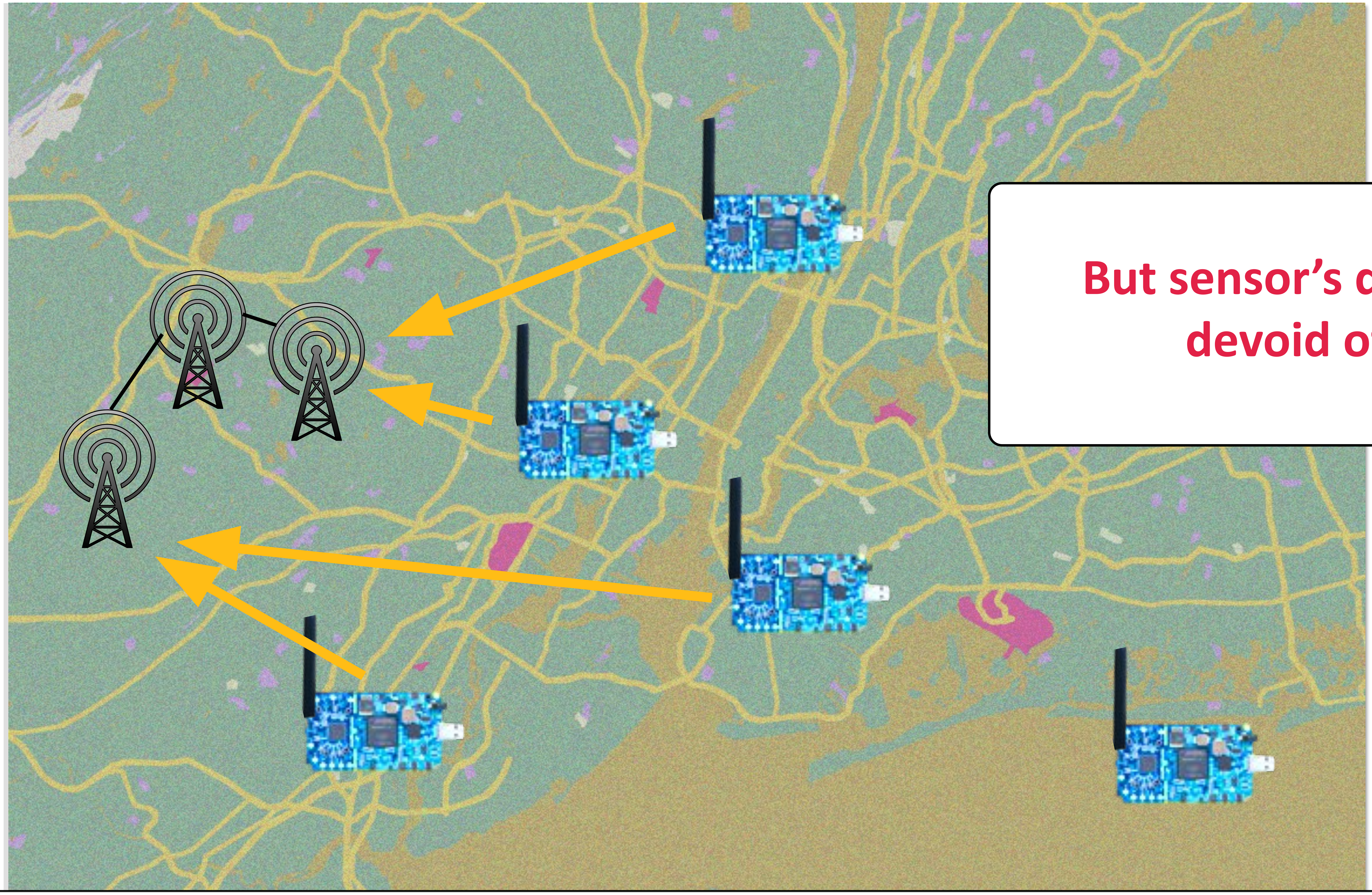
How?!!!

Isn’t it same like multi-arm bandit.



Well each sensor have a fixed probability of being closest to the transmitter. And with limited budget we have to identify some of those sensors which has maximum probability of being closest to transmitter.

Okay! That make sense



But sensor's data are not devoid of noise

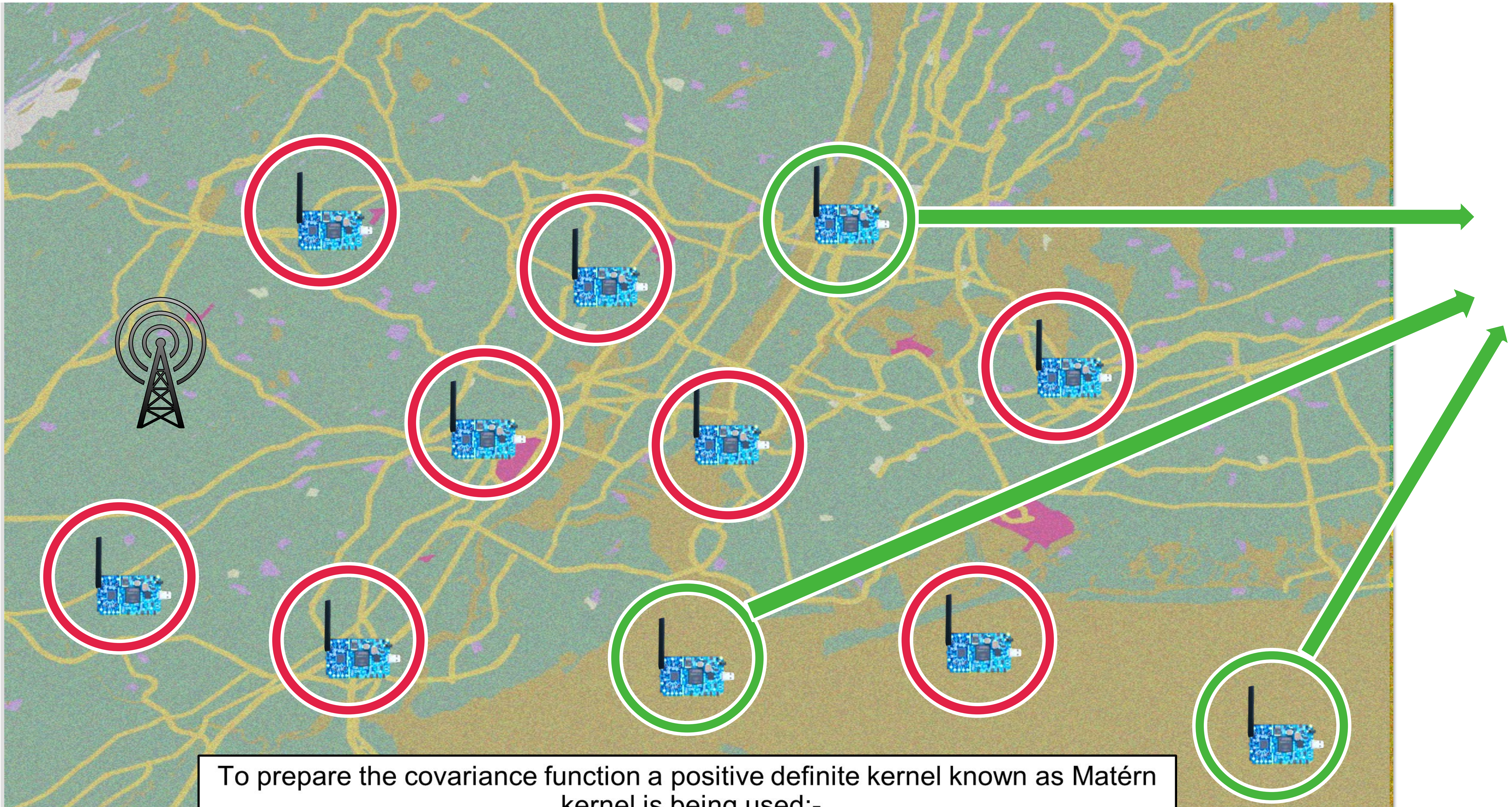
$$S_r = A_r + N$$

S_r - Received Signal, A_r - Actual transmitted Signal, N - Noise

Intuition for Gaussian process optimization.

- We want to find the best fit of our function of finding closest sensors.
- To find this peak, we will fit a Gaussian Process to our observed points and pick our next best point where we believe the maximum will be.
- This next point is determined by an acquisition function - that tradeoff exploration and exploitation
- A kernel describes the covariance of the Gaussian process random variables. Together with the mean function the kernel completely defines a Gaussian process.

Preparing for Gaussian Process



Select few random sensors and observe their reading.

Using the observation

mean -> $\mu(x)$

Co-variance -> $K(x, x^*)$

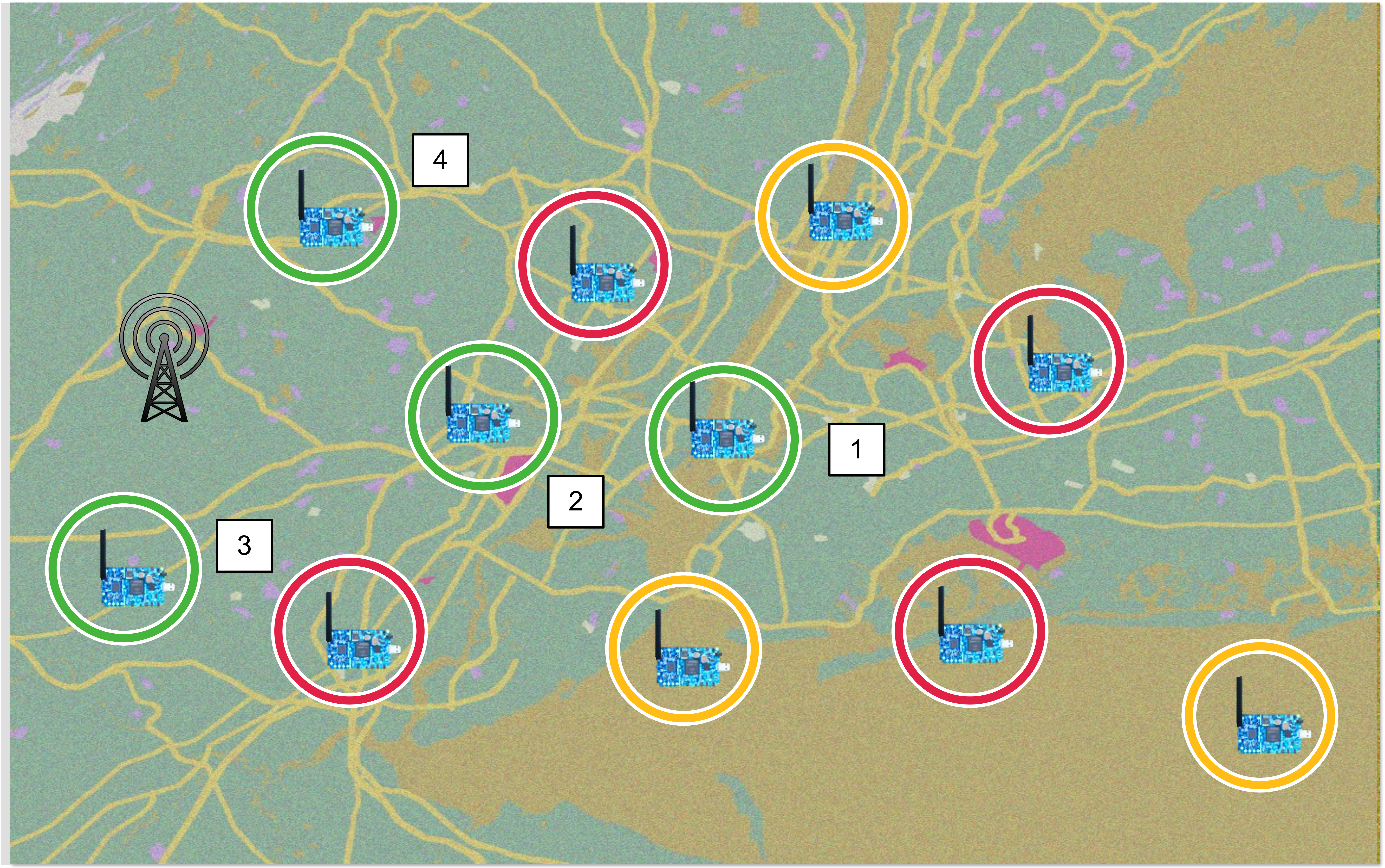
Variance -> $\sigma(x)^2$

To prepare the covariance function a positive definite kernel known as Matérn kernel is being used:-

k(x, x^*) = (2^{1-v}/Γ(v))$r^v B_v$(r),

r = ($2v^{0.5}$/l)) ‖x − x^*‖

Gaussian process sensor selection



Select few random sensors and observe their reading.

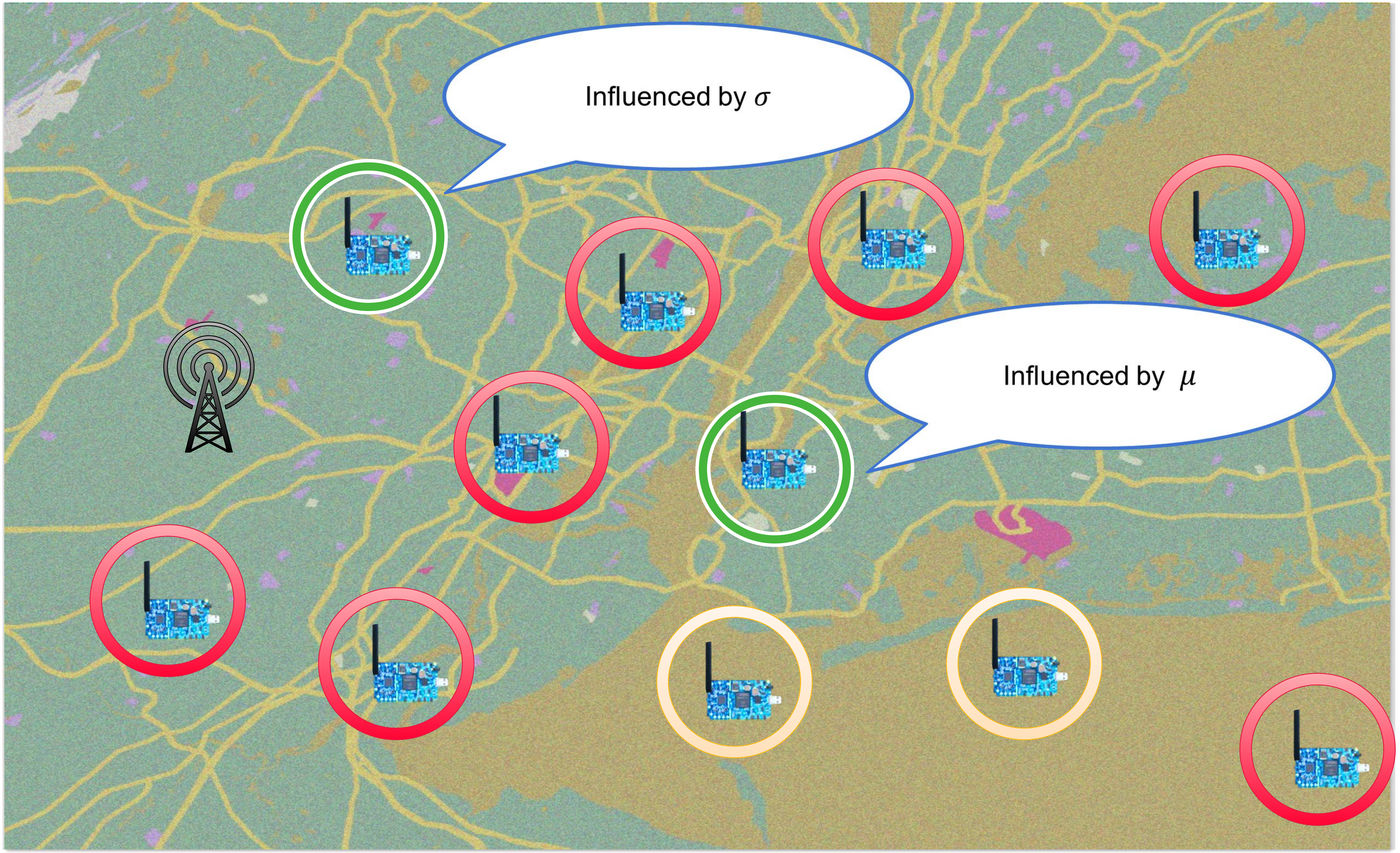
Using the observation
mean -> $\mu(x)$
Co-variance -> $K(x, x^*)$
Variance -> $\sigma(x)^2$
Is made.

Select sensors sequentially using the rule

$$x_{new} = argmax_{x \in sensor\ set}(\mu_{prev} + \beta_{prev}^{0.5} * \sigma_{prev})$$

Use Bayesian update to obtain new mean

Exploitation vs. Exploration



Content

- Motivation for sensor selection
- Problem formulation
- Algorithms used
- Evaluation

GP-UCB

β controls the exploration vs. exploitation set up.

$$\beta_t = 2\log\left(\frac{|D|t^2\pi^2}{6\delta}\right), \delta \in (0,1)$$

Regret measures loss by not knowing the optimal solution.

Start with -

1. Sensor decision set "D".
2. Gaussian Process prior mean μ_0
3. Variance σ_0^2
4. Kernel function $K(.,.)$

↓

Find a new sensor using the rule : -

$$x_t = argmax_{x \in sensor\ set}(\mu_{t-1} + \beta_t^{0.5} * \sigma_t)$$

↓

Sample :- $y_{observed} = f_{signal_t} + error_t$

↓

Use Bayesian update to obtain new mean and variance

↓

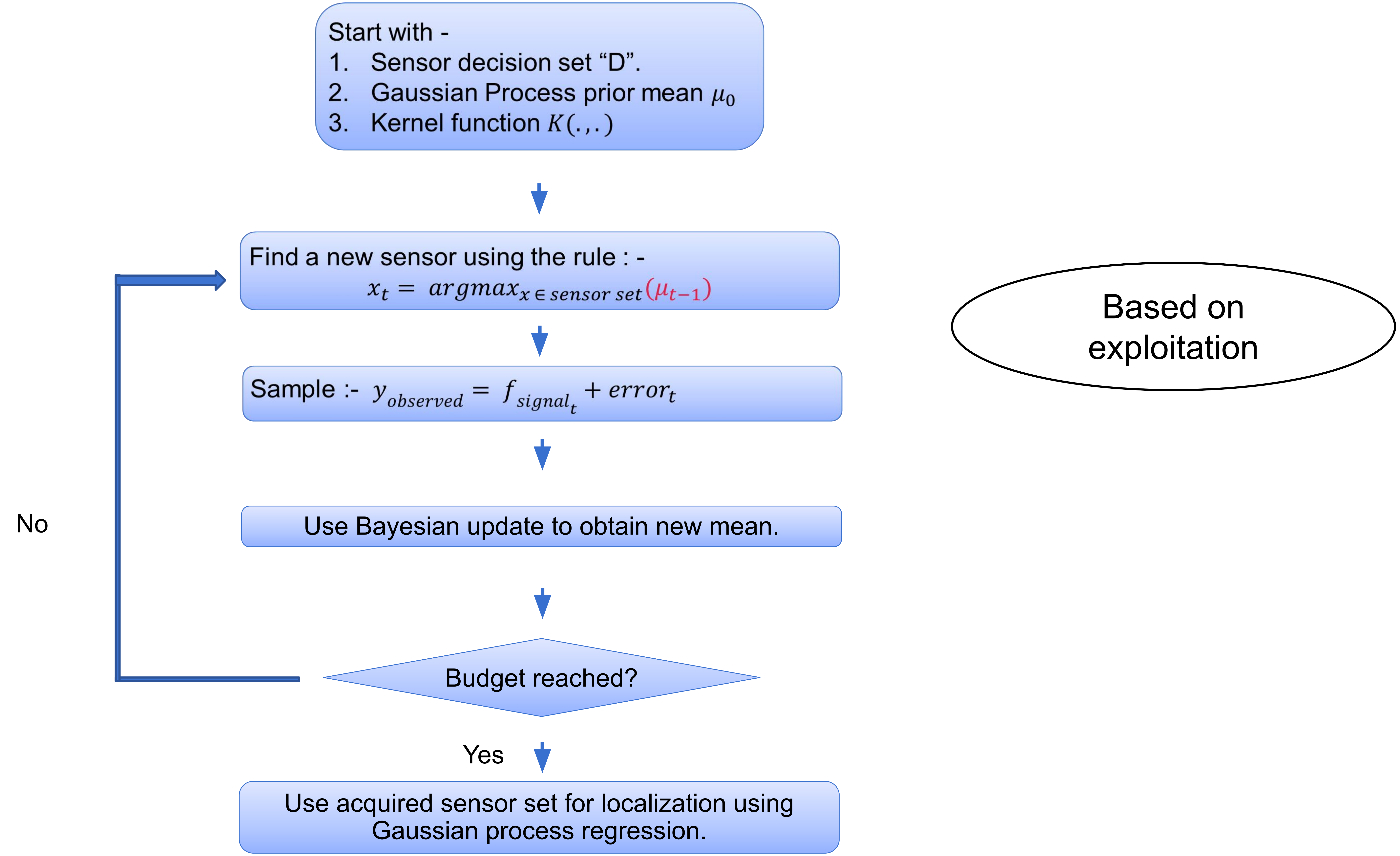
Budget reached?

No

Yes

Use acquired sensor set for localization using Gaussian process regression.

Mean Only



Start with -
1. Sensor decision set "D".
2. Gaussian Process prior mean μ_0
3. Kernel function $K(.,.)$
Find a new sensor using the rule : -
$x_t = argmax_{x \in sensor\ set}(\mu_{t-1})$
Sample :- $y_{observed} = f_{signal_t} + error_t$
Use Bayesian update to obtain new mean.
Budget reached?
No
Yes
Use acquired sensor set for localization using Gaussian process regression.
Based on exploitation

Variance only

Start with -

1. Sensor decision set "D".
2. Gaussian Process prior mean μ_0
3. Variance σ_0^2
4. Kernel function $K(.,.)$

Find a new sensor using the rule : -

$$x_t = argmax_{x \in sensor\ set}(\sigma_t)$$

Sample :- $y_{observed} = f_{signal_t} + error_t$

Use Bayesian update to obtain variance

Budget reached?

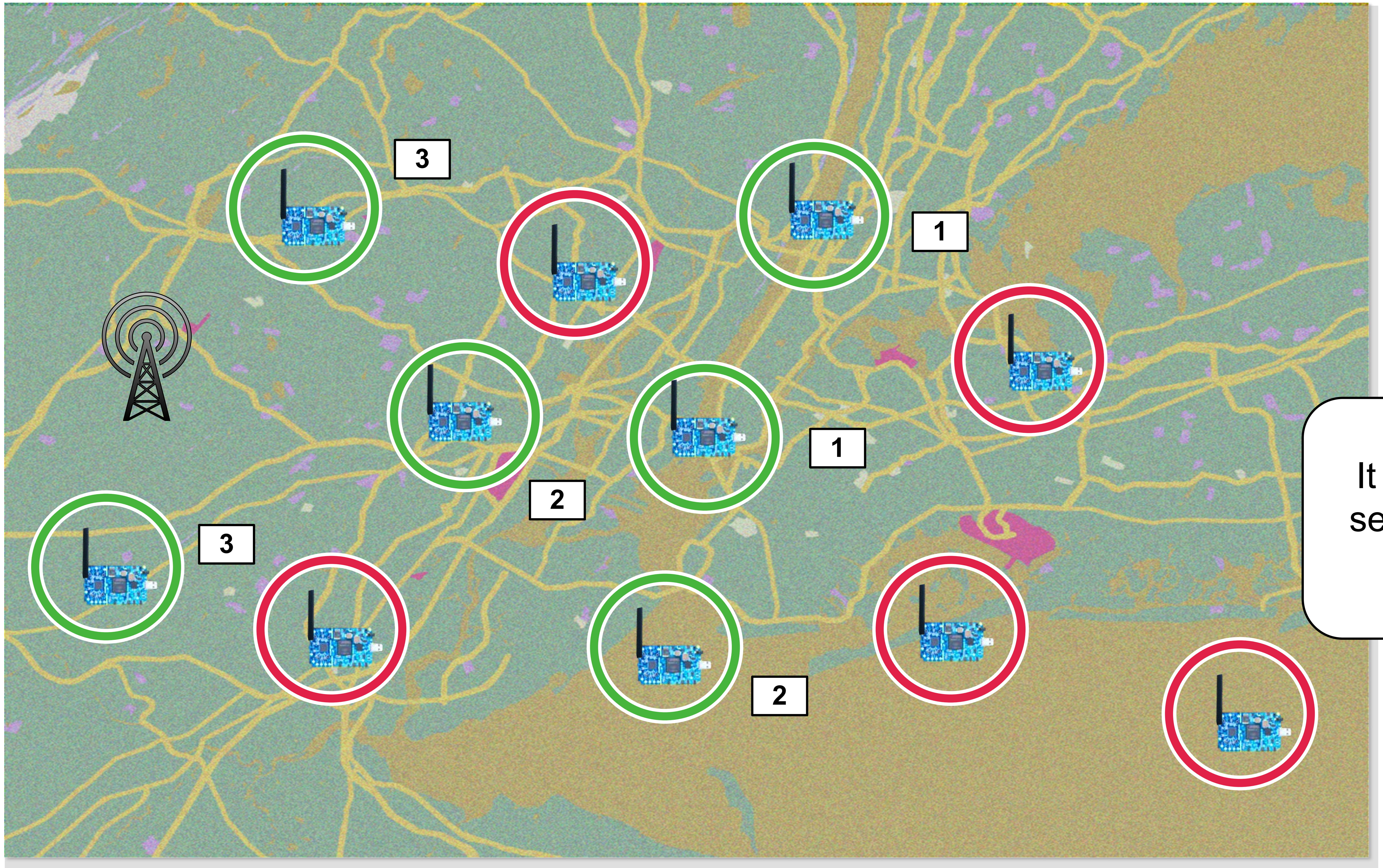
No

Yes

Use acquired sensor set for localization using Gaussian process regression.

Based on exploration

Batched Process



It is possible to select multiple sensors parallelly in batches in each iteration.

Feedback

$$fb: N \rightarrow \{N, 0\}$$

This fd is a mapping such as

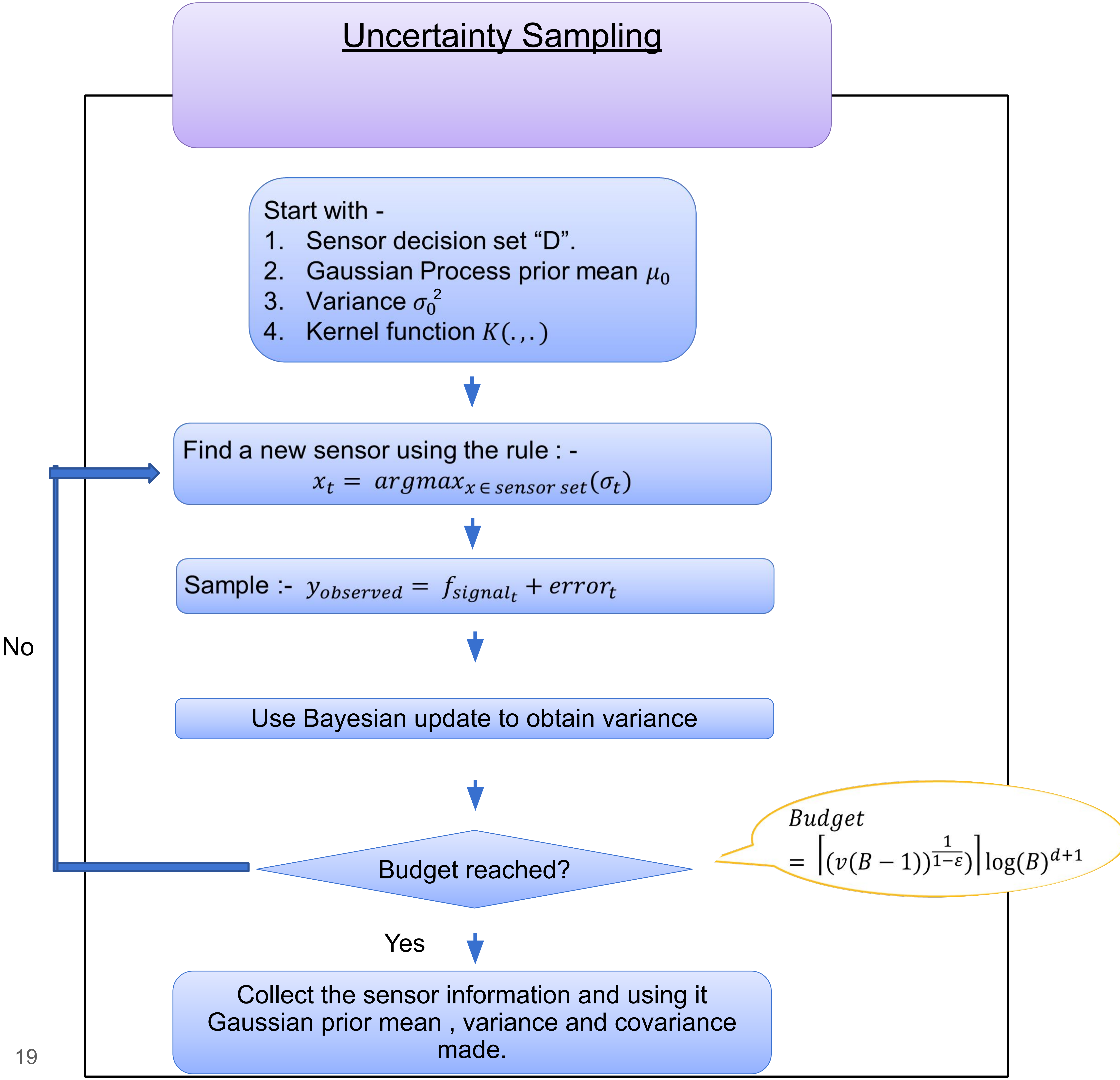
$$fd[t]_{batschsize=B} = \lfloor (t-1)/B \rfloor B$$

i.e.

$$fd[t]_{B} = \begin{cases} 0 : t \in \{1, \ldots \ldots B\} \\ B : t \in \{B+1, \ldots \ldots 2B\} \\ 2B : t \in \{2B+1, \ldots \ldots 3B\} \\ . \\ . \\ . \end{cases}$$

Hallucination

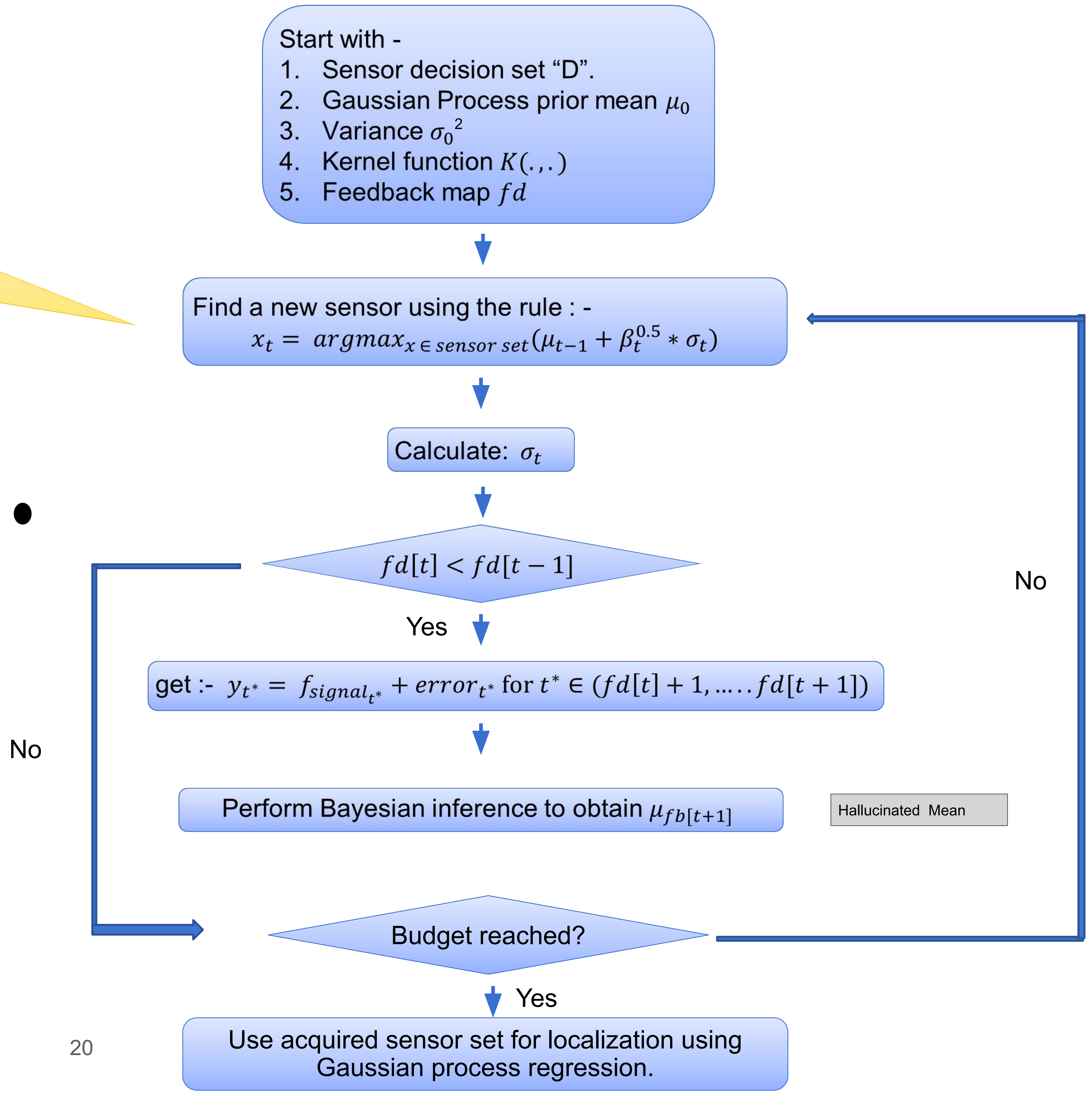
- In GP process, Variance will depend upon the previous sensor which we have observe not the observed value.
- Mean depend upon the actual observation.
- Hallucination of observation done by using most recent posterior mean.



Fixed Batch Process (GP-BUCB)

β controls the exploration vs. exploitation set up.

$$\beta_t = 2\log\left(\frac{|D|t^2\pi^2}{6\delta}\right), \delta \in (0,1)$$



Adaptive Batch Process (GP-AUCB Local)

Start with -
1. Sensor decision set "D".
2. Gaussian Process prior mean μ_0, Variance σ_0^2
3. Constant C
4. Kernel function $K(.,.)$
5. Feedback map fd
6. Maximum batch size B_{max}

Set: $fd[t^*] = 0, \forall t^* > 1$

Choice of 'C' is abstract , it is there to make sure true posterior deviation does not become too small compared to deviation form hallucinated observation.

$t - fd[t] > B_{max}$
or
$\exists x \in D : \dfrac{\sigma_{fd[t]}(x)}{\sigma_{[t-1]}(X)} > \exp(C)$

No

Find a new sensor using the rule : -
$x_t = argmax_{x \in sensor\ set}(\mu_{t-1} + \beta_t^{0.5} * \sigma_t)$

Perform Bayesian inference to obtain σ_t

Budget reached?

No

Yes

Use acquired sensor set for localization using Gaussian process regression.

Yes

get :- $y_{t^*} = f_{signal_{t^*}} + error_{t^*}$ for $t^* \in (fd[t] + 1, \ldots . . fd[t+1])$

Perform Bayesian inference to obtain $\mu_{[t-1]}$

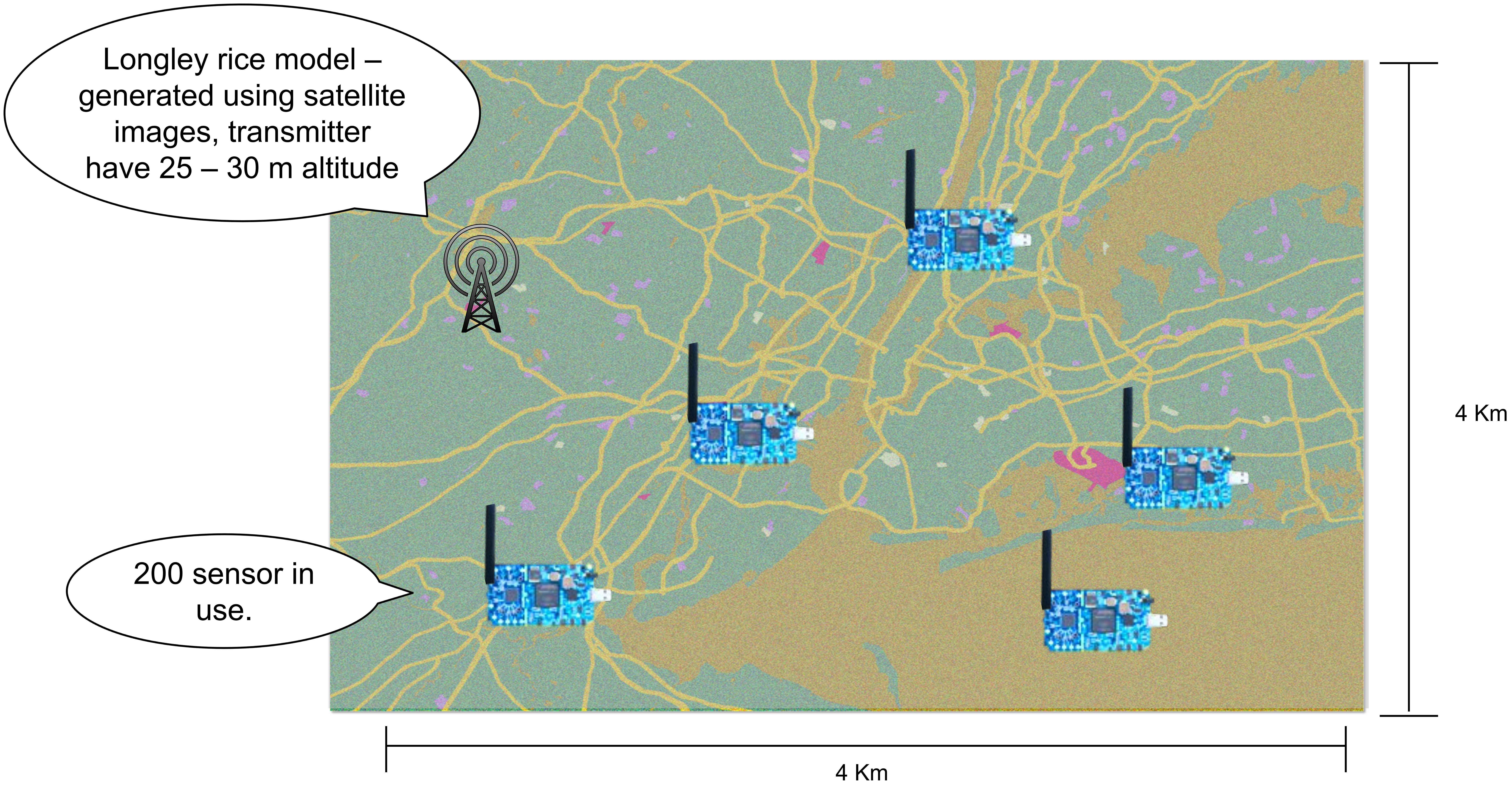
Posterior Mean

Set: $fd[t^*] = t - 1, \forall t^* > t$

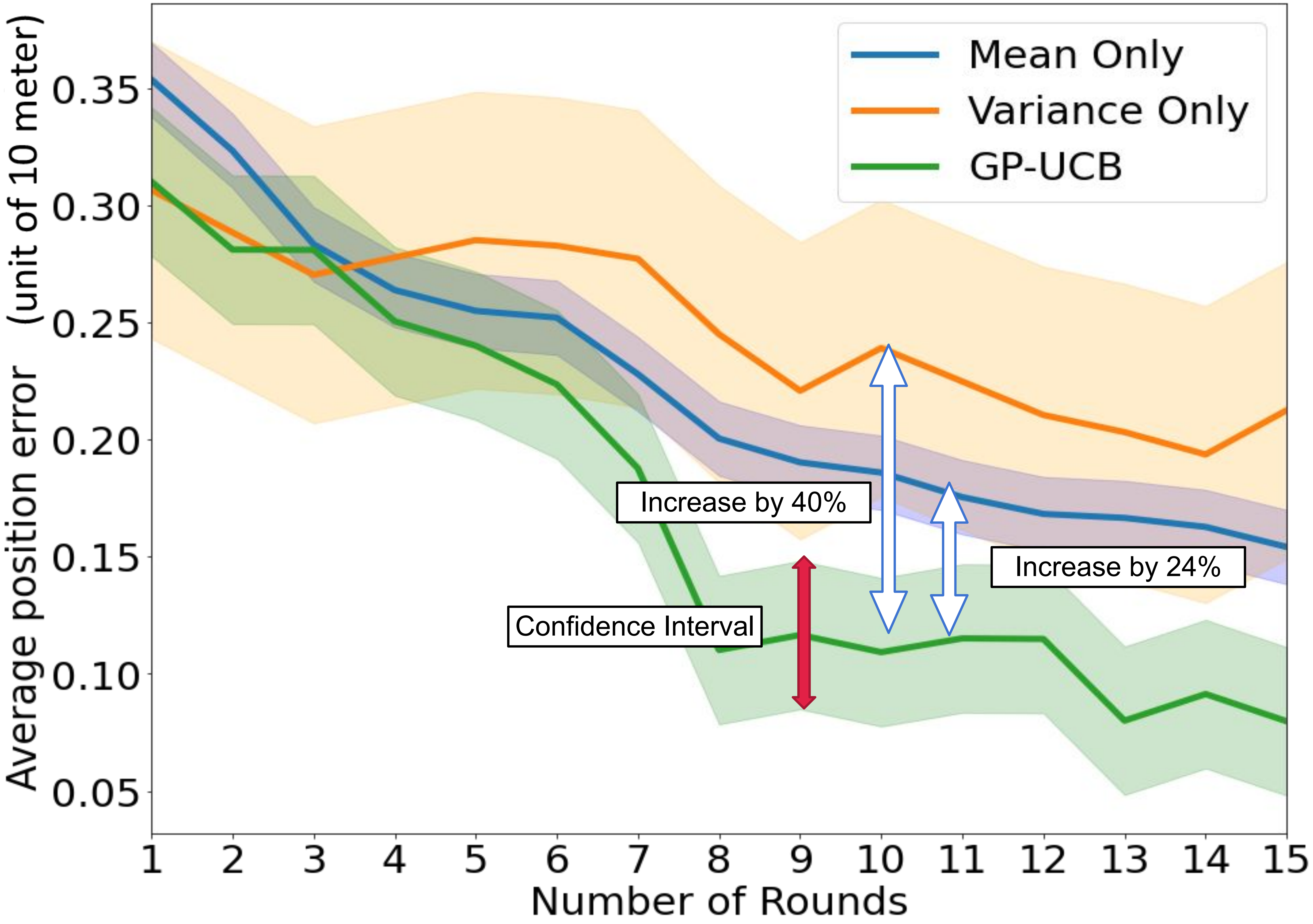
Content

- Motivation for sensor selection
- Problem formulation
- Algorithms used
- Evaluation

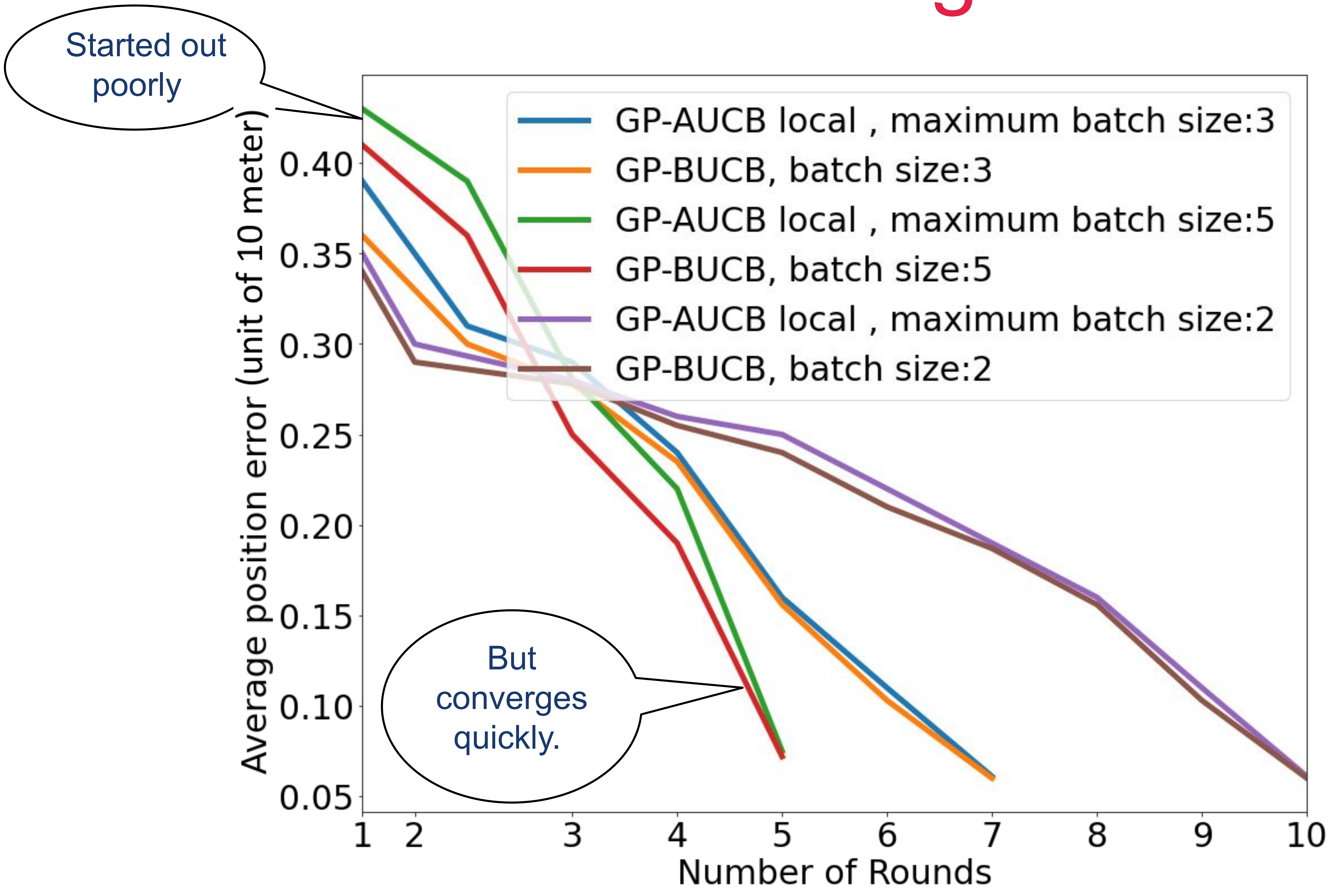
Simulation Scenario



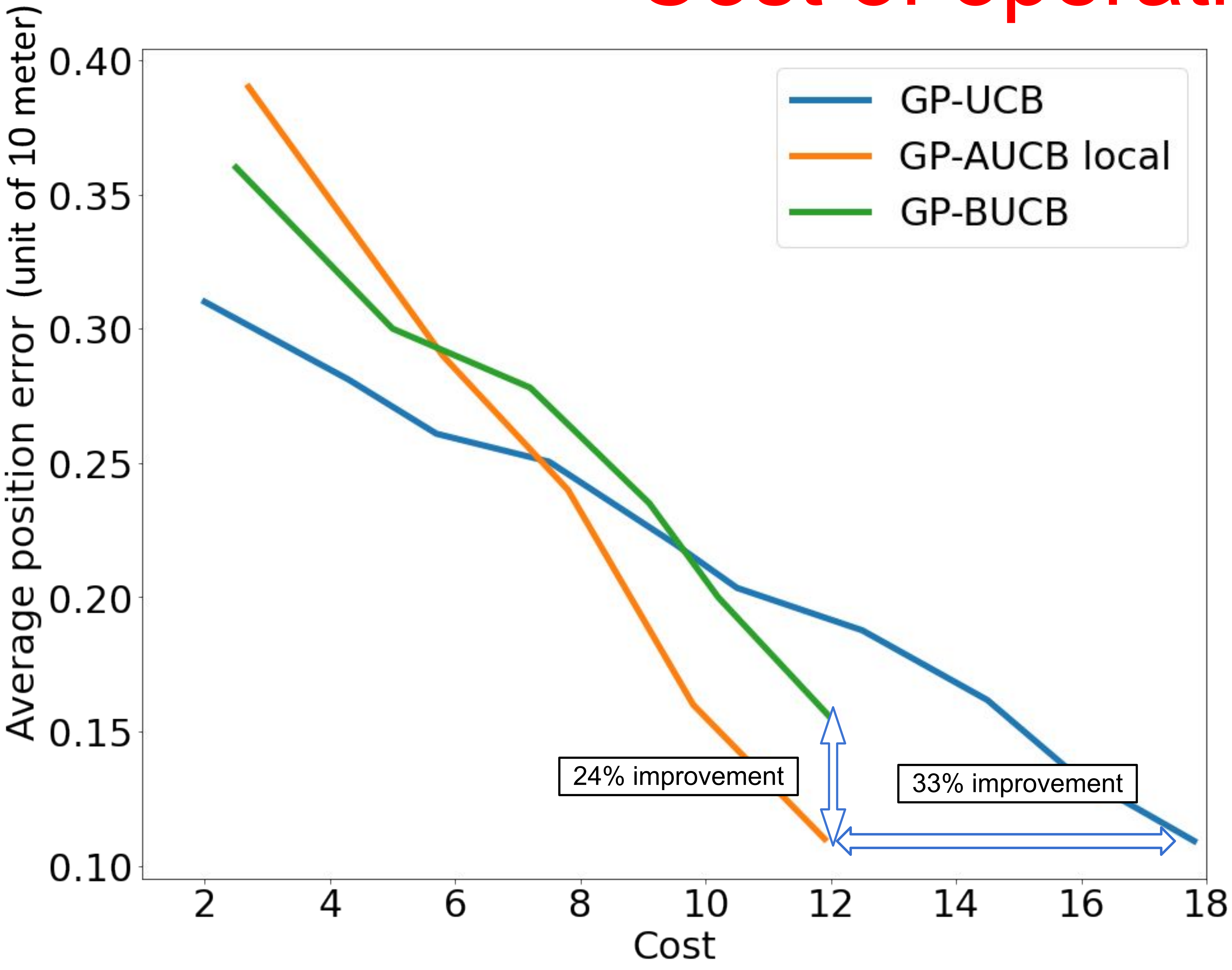
How does sequential algorithm compares mean only and variance only method?



How does batched algorithm fares?



Cost of operation!!

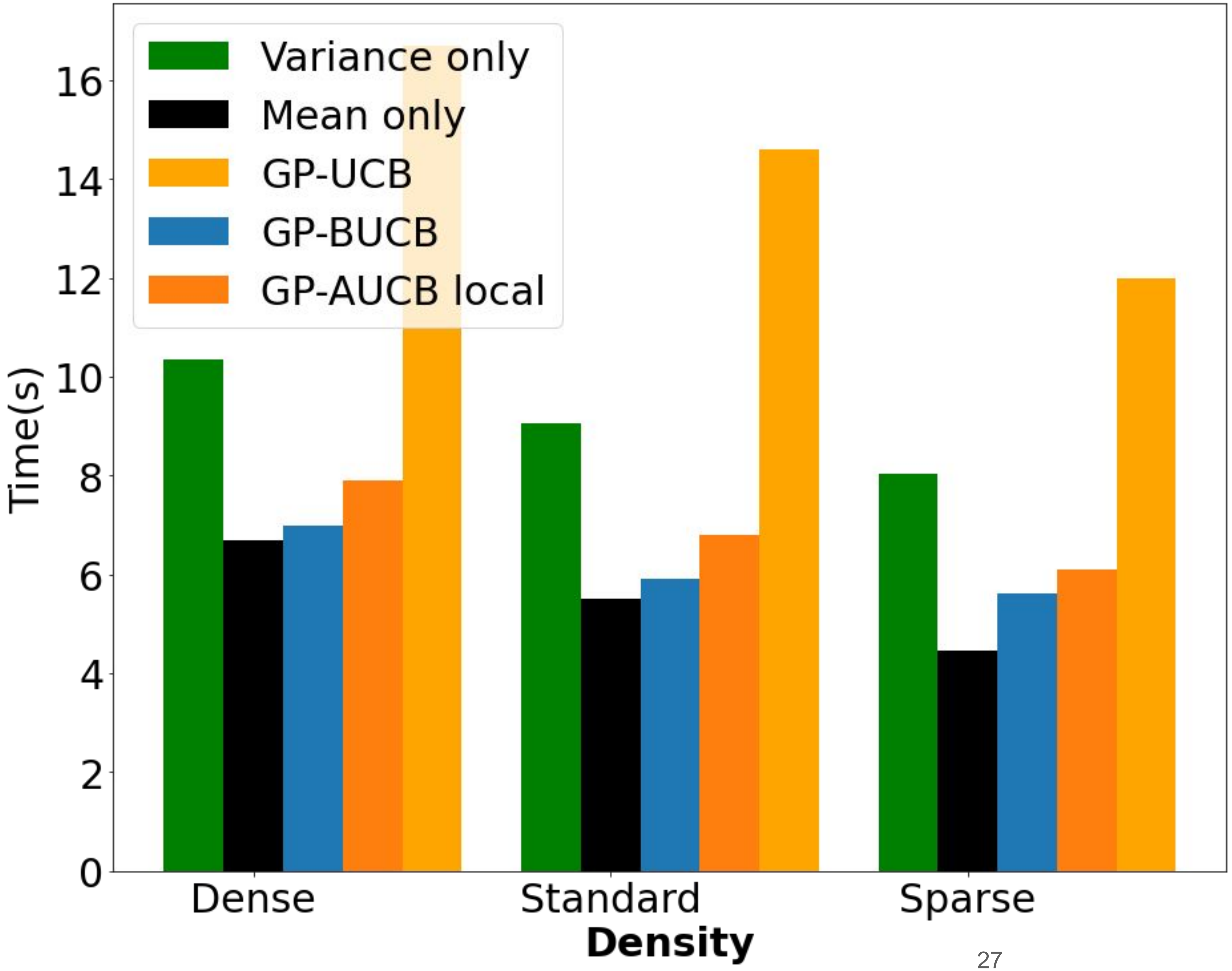


Average time to select each sensor

$$C_n = 0.3 * \sum_{i=1}^{n} (t_i^{bay}) + 0.7 * \sum_{i=1}^{n} (t_i^{sel})$$

Time taken to run the Bayesian update.

How fast are sensors are selected?



A total of 7 sensors were selected.

- Dense : 500 sensors
- Standard : 200 sensors
- Sparse : 50 sensors

Conclusion

- We showed a technique of sequential selection of sensors to localize an unauthorized transmitter.
- We first showed that sequential sensor selection can outperform traditional sensor selection techniques, though at the cost of higher latency.
- We map this to the Gaussian Process multi-armed bandit problem, and by leveraging techniques proposed in existing literature, we utilized the Gaussian Process Upper Confidence Bound to solve it.
- To reduce the latency, we then propose to select the sensors in batches. While this reduces the accuracy, we mitigate the amount of loss of accuracy by using an algorithm that adaptively selects the batch sizes.
- We perform large-scale simulation to validate our approach and show that our approach scales to large number of sensors.