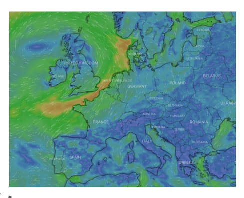
State estimation of urban air pollution with statistical, physical, and super-learning graph models

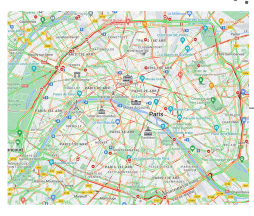
Matthieu Dolbeault[*], Olga Mula[†], and Agustin Somacal[‡]
[*]Institute for Geometry and Practical Mathematics of RWTH.
[†]TU Eindhoven, Department of Mathematics and Computer Science.
[‡] Laboratoire Jacques-Louis Lions, Sorbonne Université.

Abstract: We consider the problem of real-time reconstruction of urban air pollution maps. The task is challenging due to the heterogeneous sources of available data, the scarcity of direct measurements, the presence of noise, and the large surfaces that need to be considered. In this work, we introduce different reconstruction methods based on posing the problem on city graphs. Our strategies can be classified as fully data-driven, physics-driven, or hybrid, and we combine them with super-learning models. The performance of the methods is tested in the case of the inner city of Paris, France.

Available data and preprocessing

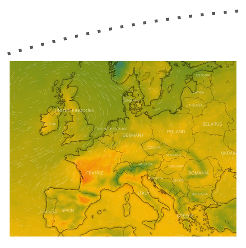


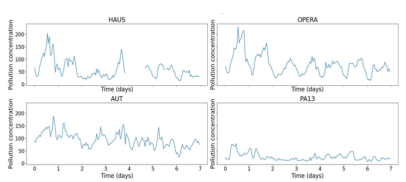
Google traffic screenshot



Project traffic-colored-pixels on street city graph

OpenStreetMap city graph





Wind
w

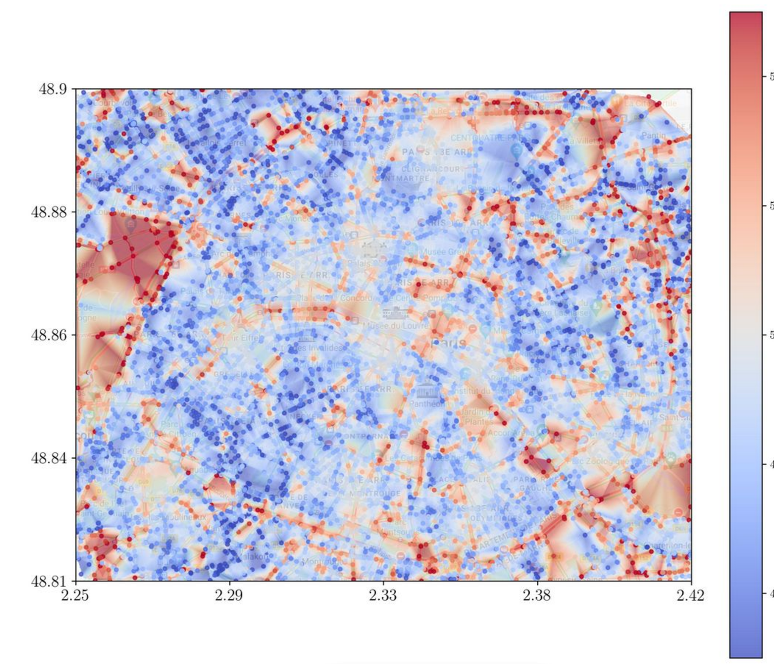
Traffic on nodes
q^v

Temperature
θ

NO_2 measurements
z

State estimation model A

Pollution Map



Best Linear Unbiased Estimator (BLUE)

The best we can do (with a linear method) if we had the full statistics of every target point.

As it is a pure statistical method, it will be used as a **lower error bound baseline.**

Objective: Estimate pollution at a station point r_i.

Data needed:
- Statistical information given by the history (z_i^t) of the station at previous times t.
- NO_2 observations z_j^t on other stations j≠i in the present and the past.

$$A_{\text{blue}}(r_i) = \langle z_i \rangle + \sum_{j\neq i} c_j (z_j - \langle z_j \rangle)$$
$$c_j = K_{jk}^{-1} K_{ki} \qquad j,k \neq i$$
$$K_{ij} = Cov(z_i, z_j)$$
$$Cov(z_i - A_{\text{blue}}(r_i), z_j) = 0$$

Spatial Average

The simplest we can do, just average NO_2 measurements at present time.

It will be used as a **high error bound baseline.**

Objective: Estimate pollution at a given point r.

Data needed:
- NO_2 observations z_i on available stations in the present.

$$A_{\text{avg}}(r) = \bar{z} = \frac{1}{m}\sum_{i=1}^{m} z_i$$

Source

Linear method for traffic on nodes fitted to locally improve predictions given by Spatial Average.

Objective: Estimate pollution at a given node v.

Data needed:
- NO_2 observations z_i^t on available stations in the present and the past.
- Traffic density vector q_c^v on node v
- Temperature θ
- Wind w

$$A_{\text{src}}(v) = A_{\text{avg}}(v) + \alpha_\theta \theta + \alpha_w w + \sum_{c\in\texttt{colors}} \alpha_c (q_c^v - \bar{q})$$
$$\texttt{colors} := \{\texttt{green}, \texttt{orange}, \texttt{red}, \texttt{dark-red}\}$$

Kriging

Linear method using a distance dependent surrogate for approximating the missing statistics.

Objective: Estimate pollution at a given point r.

Data needed:
- NO_2 observations z_i^t on available stations in the present and the past.
- Location r_i of the known stations.

BLUE can not be computed outside know stations because the covariance matrices K_{jk} and K_{ki} are unknown. Here we model the missing statistics by a kernel that decays exponentially with the distance.

$$K(r, r') = C \exp\left(-\frac{\|r - r'\|^2}{2\sigma^2}\right)$$

Physical-Laplacian

Principal Components of graph laplacian and neural-networks.

Objective and **Data needed**: the same requirements as in Source model except that temperature and wind are not used.

Steps of the algorithm:
- **Gaussian smoothing** of the traffic density q_c.
- The traffic density q_c is projected on the space spanned by the first **10** ***leading eigenvectors of graph laplacian***.
- A **cubic model** is applied to the transformed traffic density.
- A **neural-network** is trained to map the cubic interrelations to pollution values on the target node.

Physical-PCA

Principal Components of traffic on nodes with polynomial nonlinearities.

Objective and **Data needed**: the same requirements as in Source model.

Steps of the algorithm:
- **Gaussian smoothing** of the traffic density q_c.
- The smoothed traffic density is projected on the space spanned by the first **10** ***principal components***.
- A **quadratic model** is applied to the transformed traffic density vector incorporating wind and temperature.

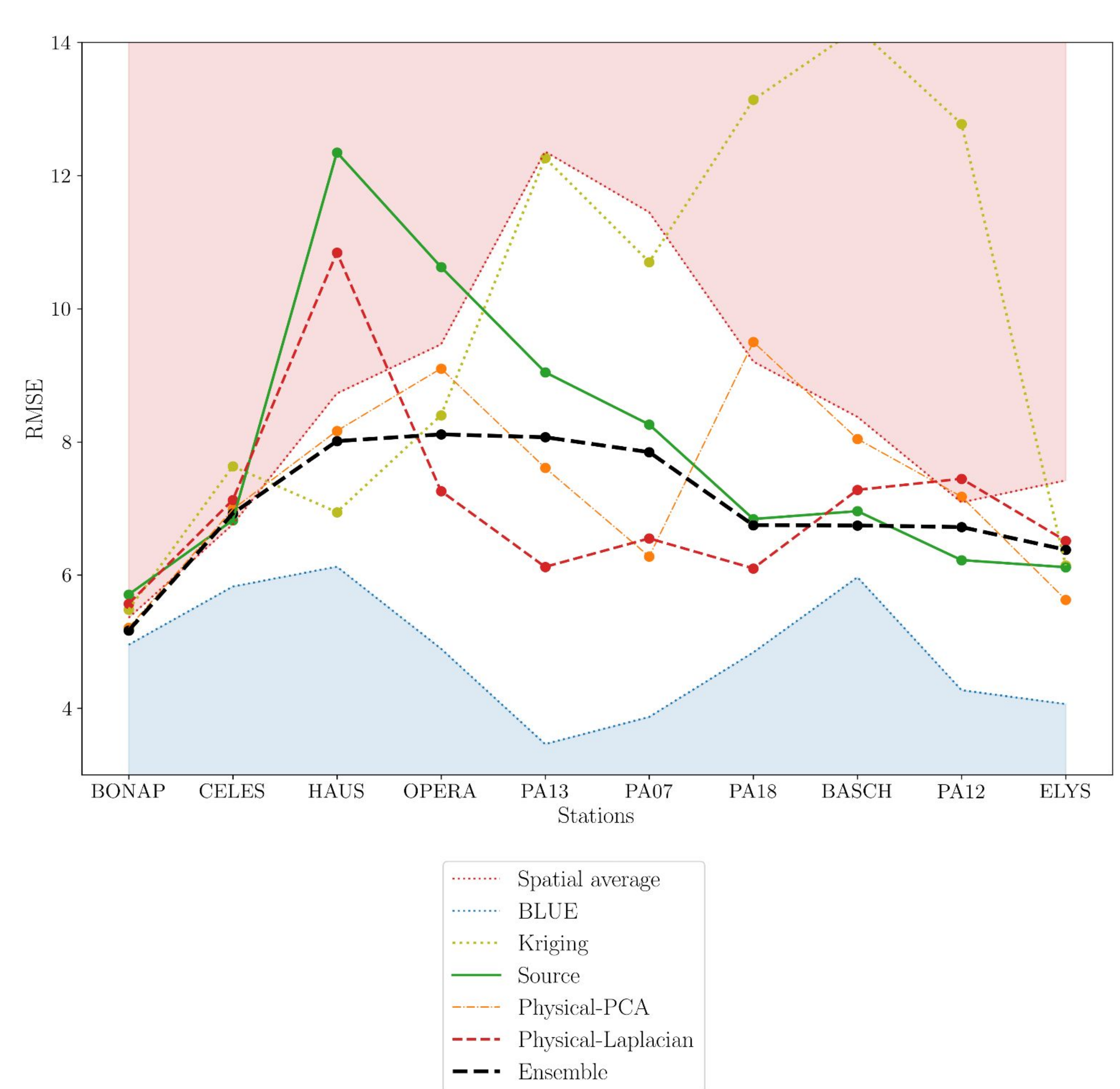
Ensemble

Weighted combination of models.

Objective and **Data needed**: the same requirements as in Source model.

Steps in the algorithm:
- Obtain predictions from Source, Physical-Laplacian and Kriging.
- Combine the models with weights that depend on the distance to known stations favoring Kriging when the target point is near a station.

$$A_{\text{ens}}(r) = \omega(r) A_{\text{krig}}(r) + \frac{1-\omega(r)}{2} A_{\text{src}}(r) + \frac{1-\omega(r)}{2} A_{\text{lapl}}(r)$$
$$\omega(r) = \exp\left(\min_{1\leq i\leq m} |r - v_i^{\text{obs}}| / \delta\right) \quad with \quad \delta = 800\,\text{m}$$



Preprint

Poster