TRAFFIC FLOW RECONSTRUCTION

Corso di: Big Data Architectures

Dott. Marco Fanfani

DISIT Lab, Sistemi Distribuiti e Tecnologie Internet

http://www.disit.dinfo.unifi.it/

Dipartimento di Ingegneria dell'Informazione

Università degli Studi di Firenze

CC BY SA

INTRODUCTION

Smart Mobility is a tool to achieve **sustainable** development of cities.

Mobility solutions depend on **traffic status** in the road network.

The knowledge of the **real-time traffic flow status in each segment** of a whole road network in a city or area is becoming fundamental for a large number of smart services such as:

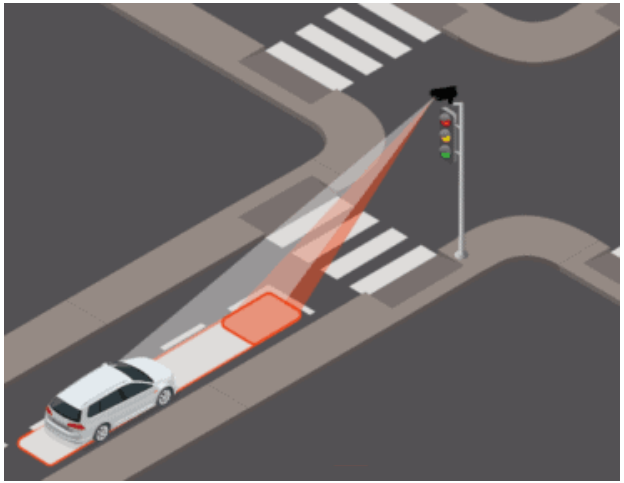
- routing,
- planning,
- dynamic tuning services,
- healthy walk,
- etc.

SENSORS ARE LIMITED

Traffic density measures are typically obtained by **stationary sensors on fixed positions**, and they are usually of different kinds: TV cameras, road spires, etc.

Due to sustainability reasons, the number of deployed sensors is **limited**.

Reconstruction algorithms are required to obtain the traffic flow condition in each road segment of the city.



DEFINITION

*The **Traffic Flow Reconstruction**, TFR, is the process to estimate dense traffic density (flow) – e.g., vehicle per meter (or vehicles per second) – for each road segment within the road network by starting from a limited number of traffic flow sensors having fixed positions in the network at the same time instant.*

METHODS OVERVIEW

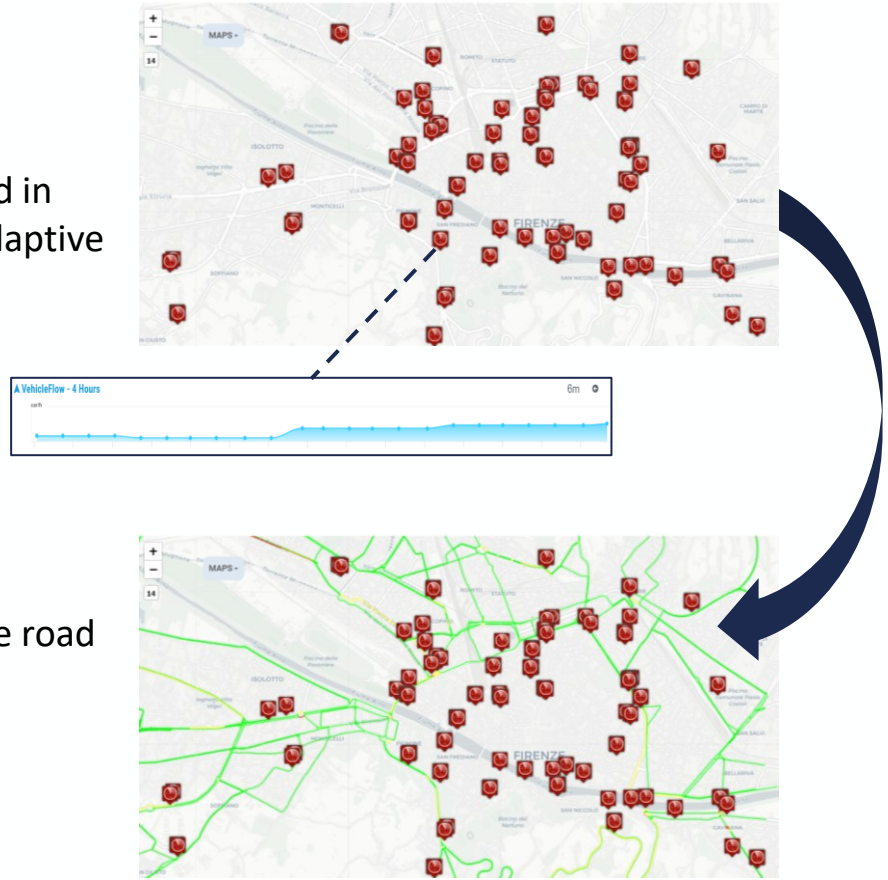
Goal: Obtain a **traffic reconstruction** in every road in **real-time**, at **low-cost**, using a general and self-adaptive model.

Input:

- road graph
- sensors' measurements

Methods:

- Exploit a **fluid-dynamic model** adapted for the road network
- Use **Graph Neural Network** for computing embedding of roads and sensors



FLUID-DYNAMIC MODEL

- The traffic sensor detections are interpreted as **sources of traffic** leading into the outcoming roads of the nodes where sensors are located.
- We consider a **mathematical model for fluid dynamic flows on networks** which is based on conservation laws.
- Road network is studied as a **directed graph** composed by arcs that meet at some nodes, corresponding to **junctions**.

Bilotta & Nesi, *Traffic flow reconstruction by solving indeterminacy on traffic distribution at junctions*, FGCS, 2021



MATHEMATICAL MODEL

Roads are modelled as if they were water pipelines.

Crossroads are modelled as if they were pipeline junctions.

The flow of the vehicles is modelled as if it was a water flow.

The **law of conservation of the flow (of the vehicles, LWR model)** applies:

$$\frac{\partial \rho(t,x)}{\partial t} + \frac{\partial f\big(\rho(t,x)\big)}{\partial x} = 0$$

- $\rho(t,x)$ is the **vehicular density**, with values from 0 to $\rho_{\max}$
- $f(\rho(t,x)) = \rho(t,x)v(t,x)$ is the **vehicular flux**, and
- $v(t,x)$ is the **local speed** of the vehicles.

MATHEMATICAL MODEL

Roads are modelled as if they were water pipelines.

Crossroads are modelled as if they were pipeline junctions.

The flow of the vehicles is modelled as if it was a water flow.

The **law of conservation of the flow (of the vehicles, LWR model)** applies:

Partial derivative of the **density** w.r.t. the **time**

$$\frac{\partial \rho(t,x)}{\partial t} + \frac{\partial f\left(\rho(t,x)\right)}{\partial x} = 0$$

Partial derivative of product of the **density** and the **velocity** w.r.t. the **space**

- $\rho(t,x)$ is the **vehicular density**, with values from 0 to $\rho_{\max}$
- $f(\rho(t,x)) = \rho(t,x)v(t,x)$ is the **vehicular flux**, and
- $v(t,x)$ is the **local speed** of the vehicles.

MATHEMATICAL MODEL

The **law of conservation of the flow (of the vehicles, LWR model)** applies:

$$\frac{\partial \rho(t,x)}{\partial t} + \frac{\partial f\left(\rho(t,x)\right)}{\partial x} = 0$$

- $\rho(t,x)$ is the **vehicular density**, with values from 0 to $\rho_{\max}$
- $f(\rho(t,x)) = \rho(t,x)v(t,x)$ is the **vehicular flux**, and
- $v(t,x)$ is the **local speed** of the vehicles.

*The **variation over time in the number of vehicles** present on a stretch of road is determined by the **difference** between the **flow of vehicles entering and leaving** the stretch itself.*

In simple terms: vehicles are not created and disappear along the road.

If more vehicles enter a stretch than exit, the **density increases**; conversely,

if more come out than enter, the **density decreases**.

MATHEMATICAL MODEL

- $\rho(t,x)$ denotes the car density which admits values from 0 to $\rho_{\max}$, where $\rho_{\max} > 0$ is the **maximal vehicular density** on the road.
- The function $f(\rho(t,x))$ is the **vehicular flux** which is defined as the product $\rho(t,x)v(t,x)$, where $v(t,x)$ is the **local speed** of the cars.
- If we assume that $v(t,x)$ is a decreasing function, only depending on the density, then the corresponding **flux is a concave function**

FUNDAMENTAL DIAGRAM

- We consider the local cars' speed as

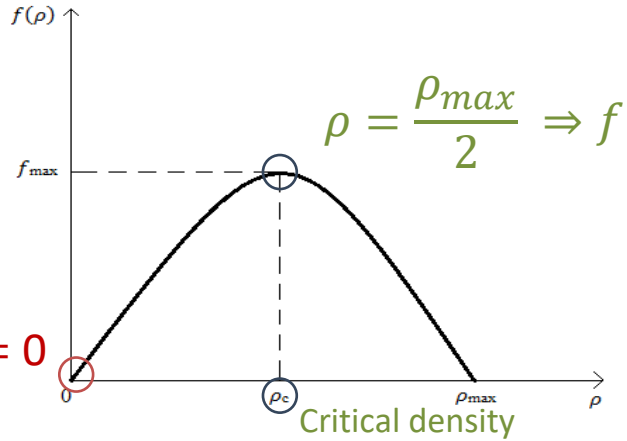
$$v(\rho) = v_{max}(1 - \frac{\rho}{\rho_{max}})$$

$$\begin{cases} \rho = 0 \Rightarrow v(\rho) = v_{max} \\ \rho^{\uparrow} \Rightarrow v(\rho)^{\downarrow} \\ \rho = \rho_{max} \Rightarrow v(\rho) = 0 \end{cases}$$

- obtaining that

$$f(\rho) = v_{max}\left(1 - \frac{\rho}{\rho_{max}}\right)\rho$$

- where v_{max} is the **speed limit**

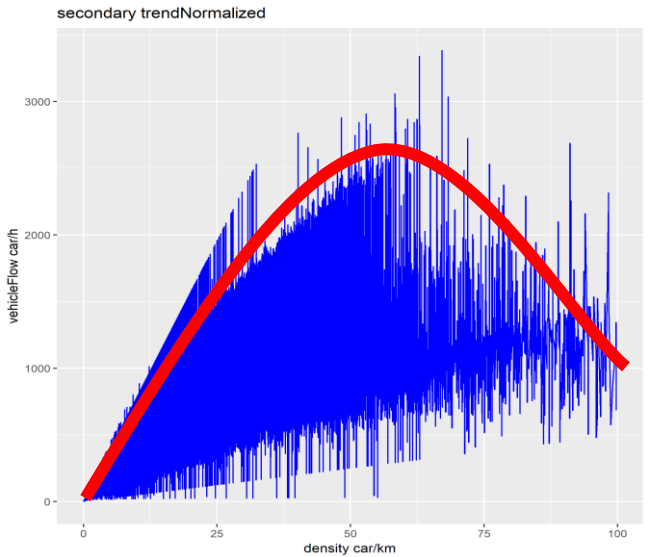


$$\rho = \frac{\rho_{max}}{2} \Rightarrow f(\rho) = f_{max}$$

$$\rho = 0 \Rightarrow f(\rho) = 0$$

FUNDAMENTAL DIAGRAM: EXAMPLE FROM DATA

Observation from traffic sensors (Florence, year 2019)



DISCRETIZATION SCHEME

The following **discretization** and simplification of the model is operated:

- Each road is partitioned in segments Δx long.
- The time is partitioned in intervals Δt long.

Denote (h,m) a **bounded time-space region (cell)** of duration h and length m.

Let $u_m^h = u(t_h, xm) = u(\mathrm{h},)$ be a continuous function defined on (h,m).

Denote *F* the numerical flux. Then, the **vehicular density** results from:

Density at time *h+1* in segment *m*

$$u_m^{h+1} = u_m^h - \frac{\Delta t}{\Delta x}\left(F\left(u_m^h, u_{m+1}^h\right) - F\left(u_{m-1}^h, u_m^h\right)\right)$$

Changes due to neighborhood segments

Density at time *h* in segment *m*

SENSORS' MEASUREMENTS

The measured data sensor is interpreted as the **source of traffic** leading into the outcoming roads of the considered junction.

Suppose to assign a condition at the incoming boundary for $x = 0$ as $\rho(t, 0) = \rho_b^{inc}(t)$.

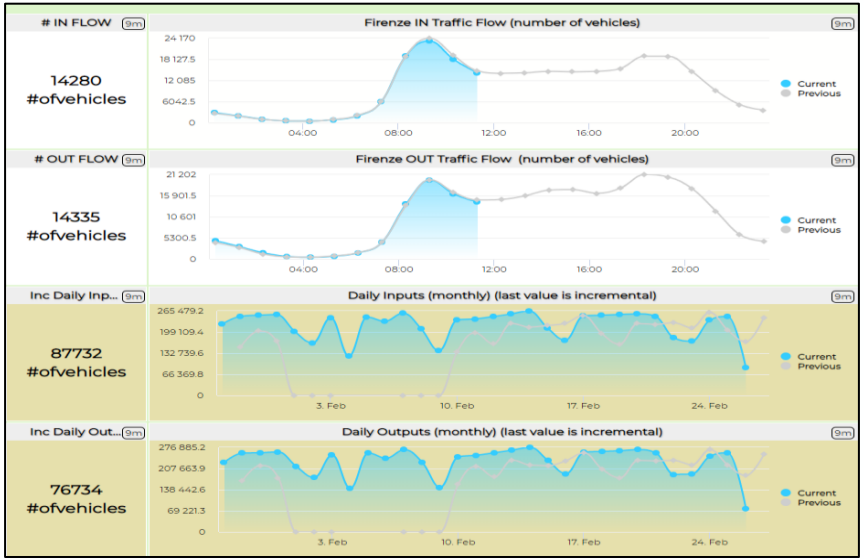
We proceed by inserting an **incoming ghost cell** and the discretization becomes

$$u_0^{h+1} = u_0^h - \frac{\Delta t}{\Delta x}\left(F\left(u_0^h, u_1^h\right) - F\left(v_{(inc)}^h, u_0^h\right)\right)$$

where $v_{(inc)}^h = \frac{1}{\Delta t}\int_{t_h}^{t_h+1} \rho_b^{inc}(t)\, dt$ replaces the ghost value u_{-1}^h

FLOW CONSERVATION

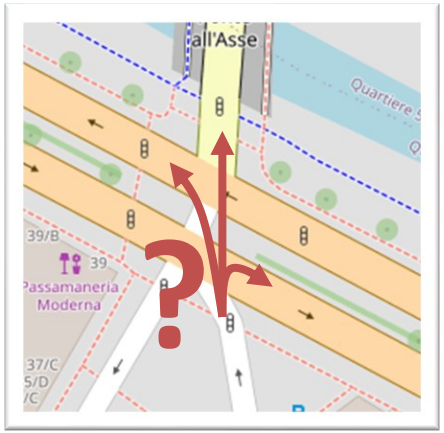
The assessment has been performed verifying in real-time the conservation of the flow in the area. Figure reports the real-time dashboard for controlling the conservation of flow



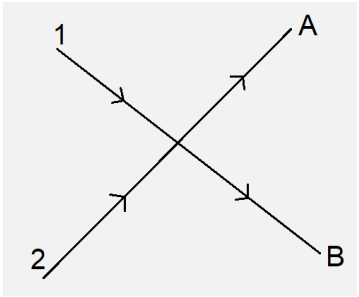
https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=MTc2MQ==

HOW TO HANDLE CROSSROADS?

The vehicular traffic flow is propagated in the network according to the fluid dynamic model



The distribution of the traffic at crossroads is governed by a **Traffic Distribution Matrix** whose coefficients are based on the **weights** of the segments of roads that make the crossroad.



$$\begin{pmatrix} d_{1A} & d_{2A} \\ d_{1B} & d_{2B} \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} = \begin{pmatrix} f_A \\ f_B \end{pmatrix}$$

TDM COMPUTATION: INITIALIZATION

TDM Weights are **initialized** based on the following:

- **Road type**: motorway, trunk, primary, secondary, tertiary, unclassified, residential, service;
- **Lanes**: how many lanes are drawn on the asphalt, also considering possible restrictions (e.g. lanes reserved to public transport);
- **Traffic restrictions**: examples are mandatory/forbidden directions at crossroads, speed limits, limited traffic zones.

TDM COMPUTATION: STOCHASTIC LEARNING

It has been observed that:

- The way how vehicles distribute at crossroads varies depending on the day of the week, and of the time of the day;
- A random variation of some weights is very likely to lead to an improved accuracy;
- If no improvements are achieved after *n* attempts, it is reasonable to move anyway to the best of the last *n* configs.

An offline process is run, based on the above, that leads to time-based weight adjustments, aimed at an improved accuracy.

LEAVE-ONE-OUT CROSS VALIDATION

This is performed by computing the solution which excludes data from each different sensor (all of them), so as to estimate the deviation from the calculated traffic density $\rho_c(t)$ in the road where the selected sensor is located, with respect to the density $\rho_M(t)$ measured by the sensor, for each time t.

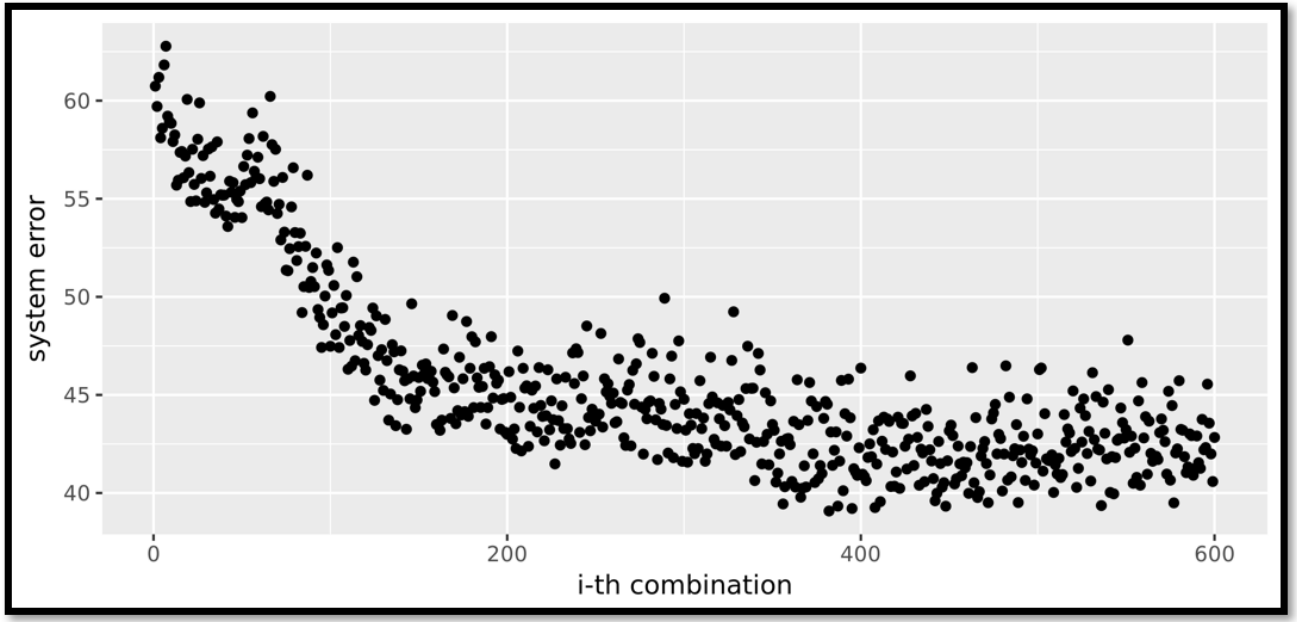
At a given location the **RMSE** is estimated as $\sqrt{\frac{\Sigma_{t=1}^{T}(\rho_c(t)-\rho_M(t))\hat{}2}{T}}$, where T is the total number of observations

At each iteration the RMSE for each sensor has been measured and also the so-called **system RMSE (system error)**, which is the average value of the measured RMSE of all the sensors.

STOCHASTIC LEARNING RESULTS

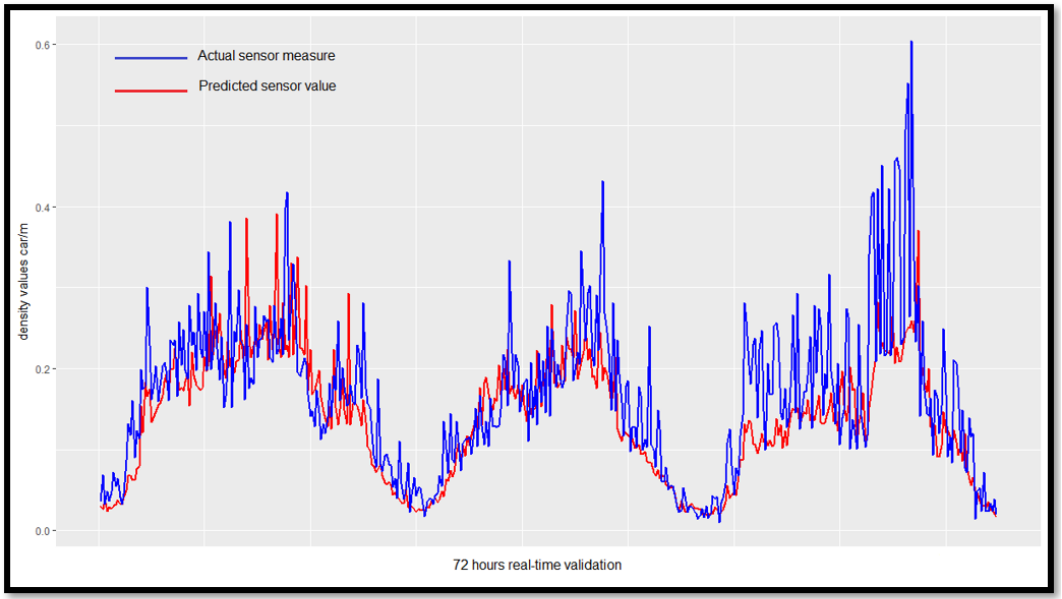
Evaluated on the real sensor measurements:

1. Use TDM to compute the vehicle flow in a segment at a given time
2. Compute the mean error for those segments where there is a sensor
3. Update the TDM weights
4. Return to step 1



In the x axis, the number of the learning iterations. In the y axis, the (decreasing) system error.

LOOCV RESULTS



The diagram refers to one in particular of the sensors, and it displays the predicted vs actual values over the time in the 72 hours validation.

The validation is conducted by means of LOOCV in the case of optimal weights assignments in road graph.

The system error has been computed to be around 30%

LIMITS OF PHYSICS-BASED MODELS

Physics-based reconstruction methods provide strong guarantees of physical consistency, but they **rely on assumptions that may not fully describe real urban traffic systems**.

Main limitations arise from:

- **Model simplification** on traffic dynamic (driver behavior, congestion phenomena, route choice, interactions among vehicles, ...)
- **Incomplete knowledge of the system** like exact turning ratios
- **Limited adaptability:** a model calibrated for one network may require redesign when applied to another city or different traffic conditions.
- **Computational complexity: l**arge-scale urban networks may lead to increasingly complex optimization problems.

These limitations motivate approaches capable of **learning hidden traffic patterns directly from historical observations**.

WHY GRAPH NEURAL NETWORK

A graph representation describes the structure of the transportation system but does not automatically provide a mechanism for **learning traffic relationships**.

A **Graph Neural Network** extends graph representation by learning:

- node representations;
- edge relationships;
- spatial dependencies;
- traffic propagation patterns.

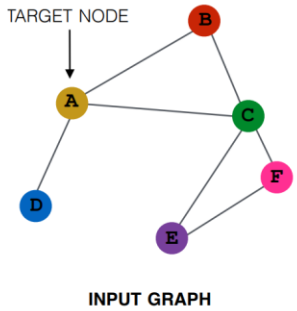
GNN MESSAGE PASSING

The core mechanism of a GNN is *Message Passing*, where each node **updates its representation by aggregating features from its immediate neighbors**.

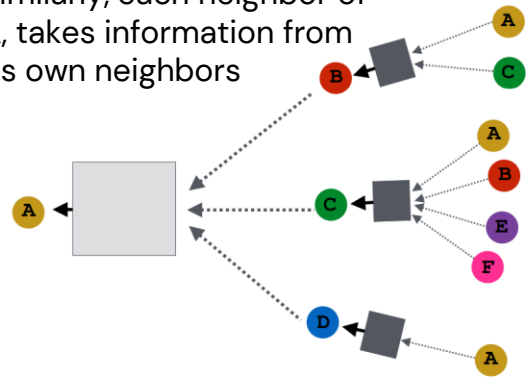
At each layer l, a

- node u receives messages from its neighbours $N(u)$
- combines them with its current state
- passes the result through a non-linear activation function

The target node A takes information from its neighbors B, C, and D



Similarly, each neighbor of A, takes information from its own neighbors



$$h_v^{(l+1)} = \sigma\left(W_l \sum_{u \in \mathrm{N}(v)} \frac{h_u^{(l)}}{|N(v)|} + B_l h_v^{(l)}\right), \forall l$$

PHYSICS-BASED MODELS VS GNNs

Physics-Based Approach	Graph Neural Network Approach
Knowledge explicitly defined	Knowledge learned from data
Conservation equations	Learned message passing
Optimization-based solution	End-to-end prediction
High interpretability	Higher flexibility
Limited by modeling assumptions	Limited by data availability
Limited training phase (TDM)	Requires historical examples

GNN FOR TFR

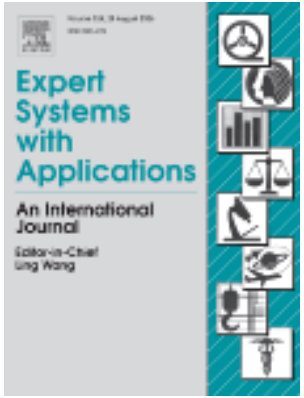
Can a Graph Neural Network automatically learn how traffic information propagates through a road network and reconstruct unobserved traffic states?

Key observations:

- traffic networks are naturally represented as graphs;
- traffic states exhibit strong spatial dependencies;
- historical observations contain implicit information about traffic propagation mechanisms.

The proposed framework replaces manually designed reconstruction rules with a learnable graph-based model.

Acciai, Bilotta, Fanfani, Nesi, *Graph neural network for continuous traffic density estimation on unmonitored roads from very few scattered measurements*, ESWA, 2026.

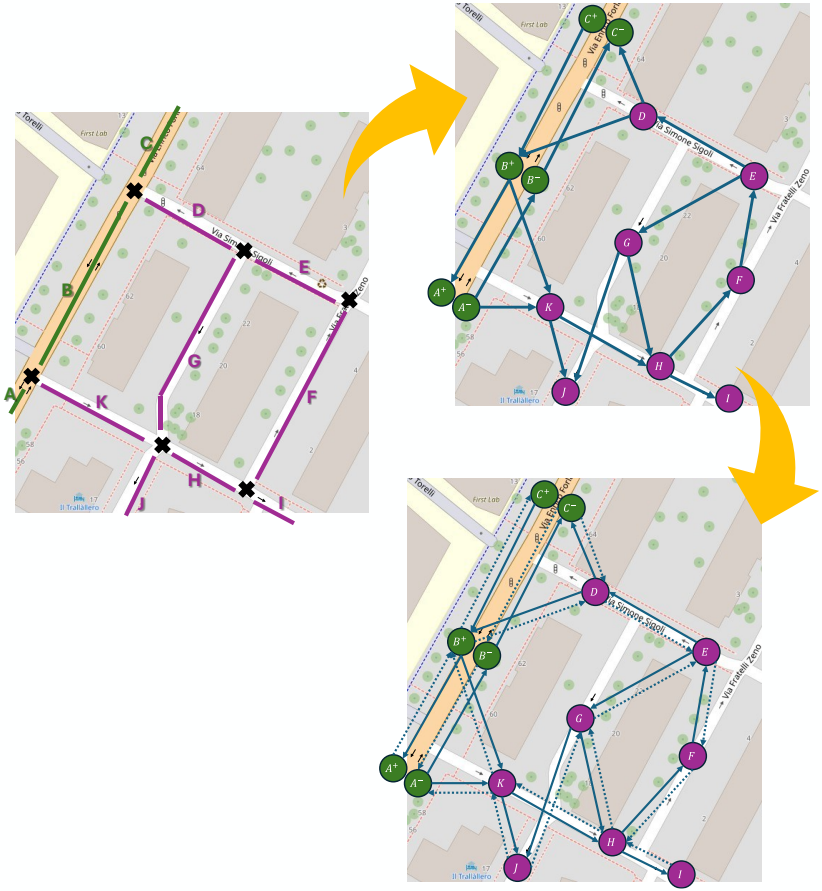


ROAD GRAPH MODELLING

Road graph is transformed to pass from an edge labeling task to a node labeling task

- roads become graph nodes
- junctions become graph edges

To account for traffic backpropagation, virtual edges are introduced



INPUT FEATURES

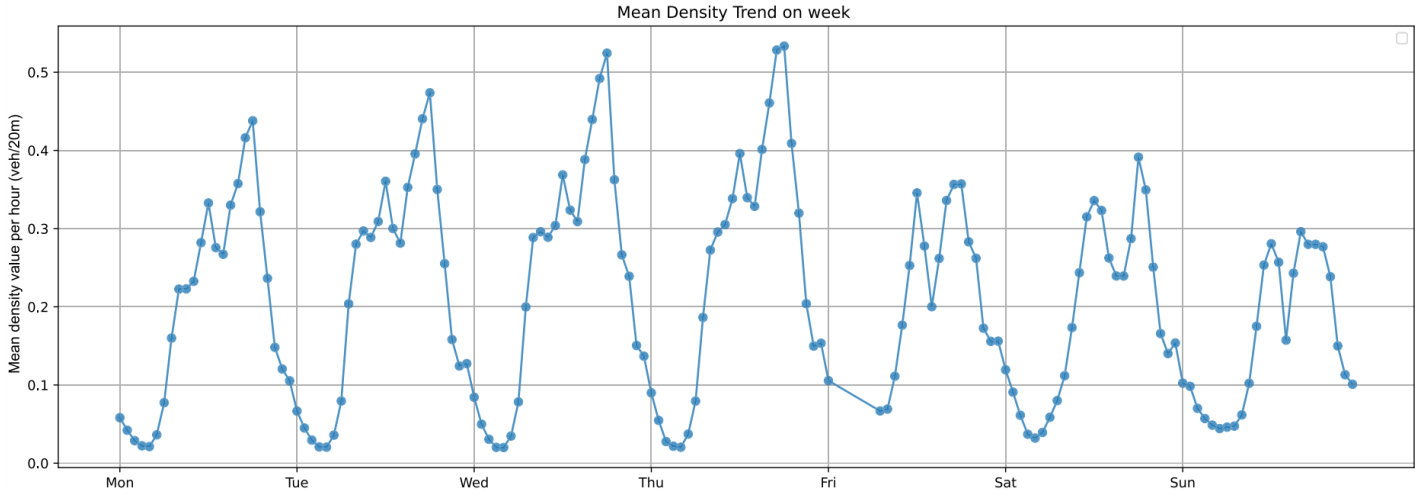
Each node (road) is described with the following features

	Kind	Feature	Description	Type
Road description	Static	lanes	Number of the lanes in the road segment	Integer
		sensor	Indicates whether the segment has a traffic sensor	Binary
		road-type	Encodes the type of road (e.g., primary, secondary)	Integer
Time description	Dynamic (time)	$\cos(\theta_q)$	Cosine of the time, see (1) and (2)	Float
		$\sin(\theta_q)$	Sine of the time, see (1) and (2)	Float
		$\cos(\theta_d)$	Cosine of the weekday, see (3)	Float
		$\sin(\theta_d)$	Sine of the weekday, see (3)	Float
		day-part	Encodes the part of the day (morning, afternoon, evening, night)	Float
		holiday	Indicates whether the day is a workday or holiday	Binary
Traffic data (only available in input for road with a traffic sensor)	Dynamic (density)	δ_t	Traffic density in the segment (node) at time t	Float
		δ_{t-15}	Traffic density in the segment (node) 15 min before t	Float
		δ_{t-30}	Traffic density in the segment (node) 30 min before t	Float
		δ_{t-45}	Traffic density in the segment (node) 45 min before t	Float
		δ_{t-60}	Traffic density in the segment (node) 60 min before t	Float

WHY TIME?

Traffic exhibit a periodic behavior

Cos/Sin encoding avoid discontinuities



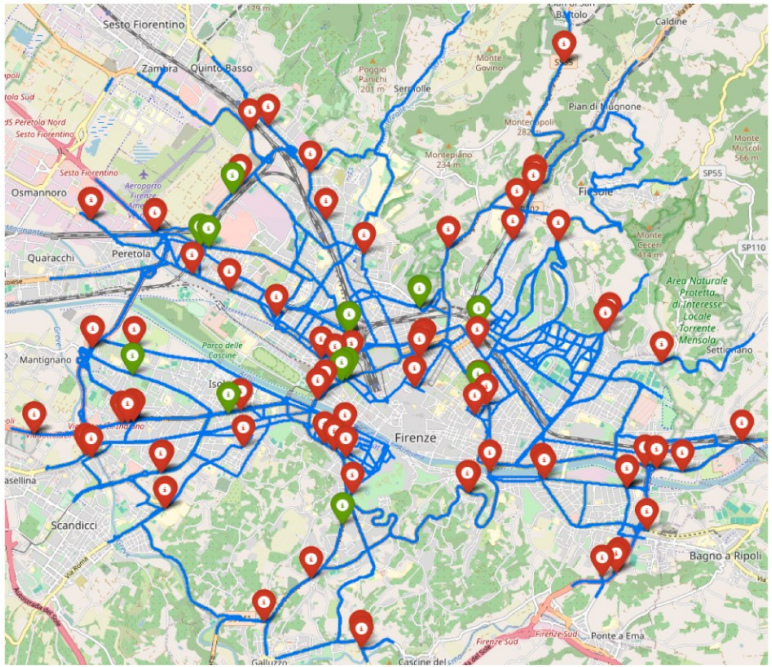
LOSS FUNCTION AND TRAINING ALGORITHM

Roads with traffic sensors (S) are split into two disjoint sets:

- **Reference** S_R including sensors in the most strategic positions
- **Control** S_C including all the other sensors

During training, S_R sensors are used in a ***leave-one-out*** approach:

- for each epoch, $s_o \in S_R$ is randomly selected and its density feature blanked out
- training loss is computed separately on s_o and on all the other sensors ($S \backslash \{s_o\}$)



LOSS FUNCTION AND TRAINING ALGORITHM

The loss function is defined as

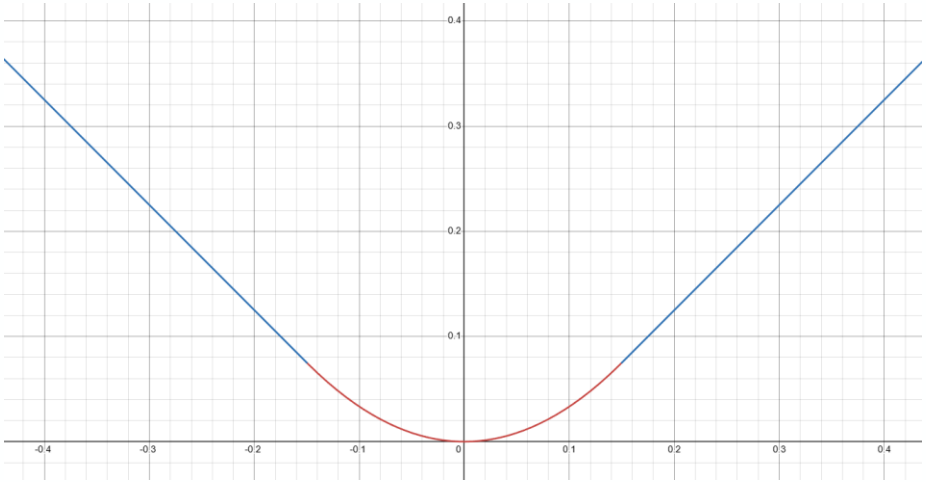
$$\mathcal{L}_{total} = \mathcal{L}_{S\setminus\{s_o\}} + \mathcal{L}_{\{s_o\}}$$

where

$$\mathcal{L}_{\Omega} = \frac{1}{N_{\Omega}} \sum_{i \in \Omega} w_t^i \ell(\hat{\delta}_t^i - \delta_t^i)$$

ℓ is a smooth L1 loss

$$\ell(\epsilon) = \begin{cases} \frac{1}{2\beta}\epsilon^2 & if\ |\epsilon| < \beta \\ |\epsilon| - \frac{\beta}{2} & \text{otherwise} \end{cases}$$



w_t^i is factor used to give more importance to the errors made when the typical density is high

LOSS FUNCTION AND TRAINING ALGORITHM

Observation:

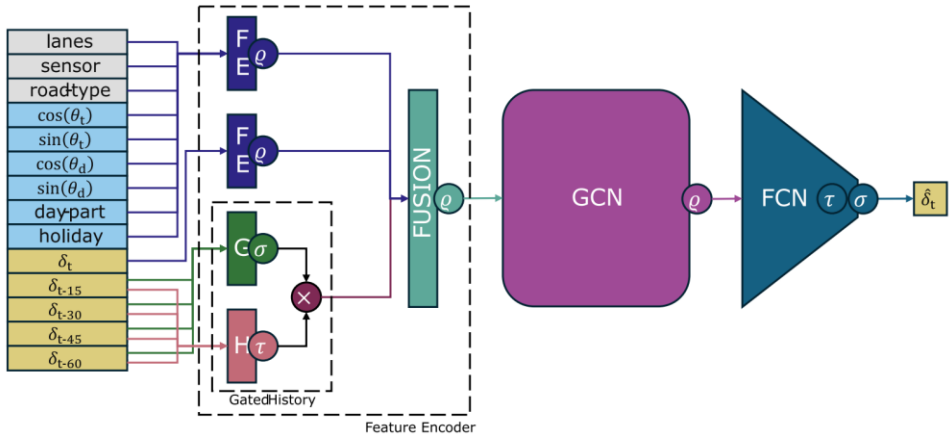
- s_o is selected from the reference sensor set S_R since it represents sensors located in the most central positions of the road graph **which are affected by a larger number of other sensors**, unlike those placed at the border of the city road graph.
- the composed loss **assigns more relevance to the error on s_o** (which requires an actual prediction), while preventing the network from diverging from the provided inputs of the other sensors
- sensors lack direct connections and are instead interconnected through a network of road segments. Being capable of correctly predicting $\hat{\delta}_t^i$ in the sensor positions **requires the neural network to be able to properly propagate the traffic** on the whole road graph.

ARCHITECTURE

Feature encoding layers are used to learn proper embedding of input features

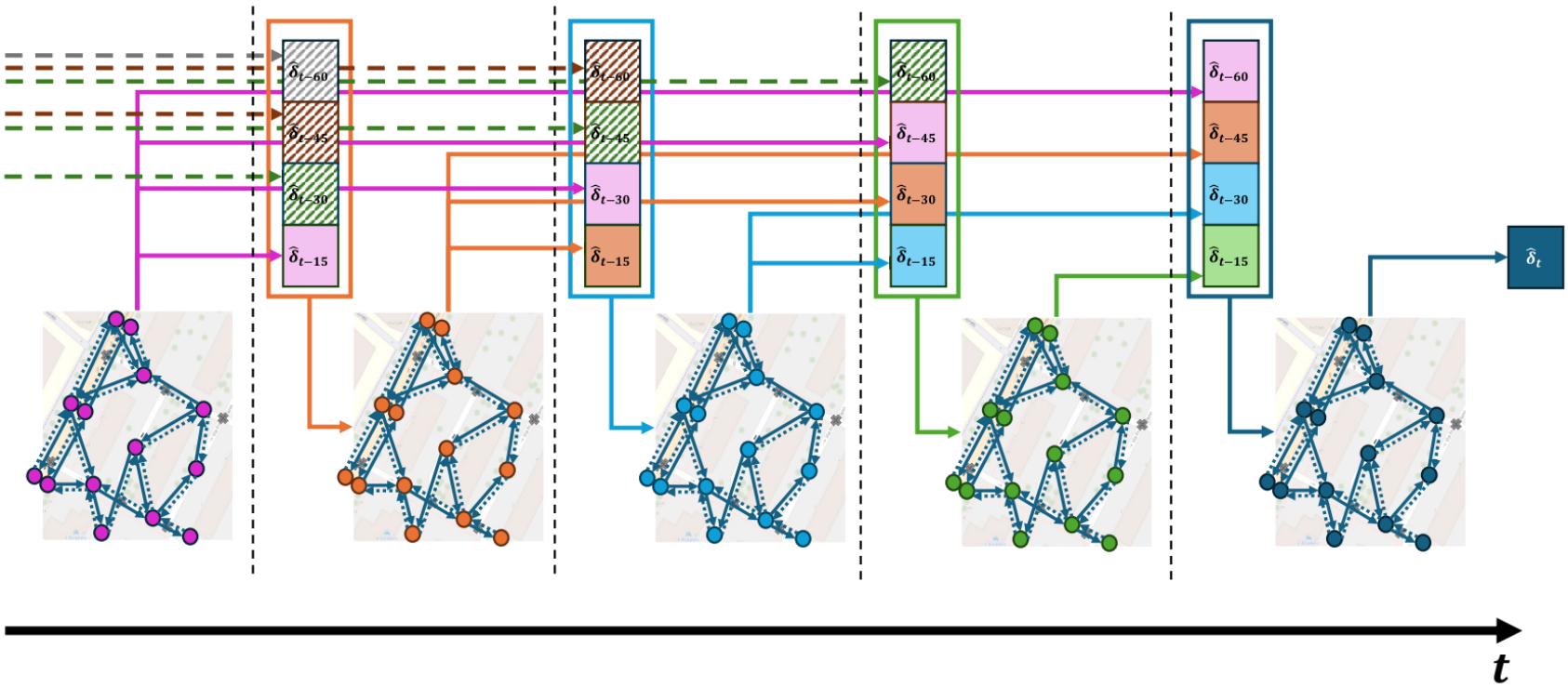
The G and H layers are used to learn the importance of the past density estimates:

- H layer performs an encoding of the past densities,
- G layer, which output is point-wise multiplied with the output of H, is used to weigh the contribution of the encoded past densities.



G-H layers avoid the net to rely too heavily on previous densities

CONTINUOUS ESTIMATION



RESULTS

The GNN based solution achieved error **comparable with the physic-based approach**, with some additional advantages:

- GNN can deal with any kind of road graph representation, it does not require domain discretization
- GNN can work on different cities without retraining or finetuning (no need to re-estimate the TDM)

RESULTS – RELEVANCE OF VIRTUAL EDGES

Without virtual edges



(g) 08:00



(h) 11:00



(i) 14:00



(j) 17:00



(k) 20:00



(l) 22:00

RESULTS – RELEVANCE OF VIRTUAL EDGES

With virtual edges



(a) 08:00



(b) 11:00



(c) 14:00



(d) 17:00



(e) 20:00



(f) 22:00

RESULTS – WHAT IF

Reconstruction in case of road closure



(a)



(b)

