Leak Learning for Graph-based State Interpolation in Water Distribution Networks*

Luis Romero-Ben[1], Gabriela Cembrano[1] and Vicenç Puig[1,2]

***Abstract*— Graph-based state interpolation (GSI) is a state-of-the-art state reconstruction technique that operates over water distribution networks (WDN). This method retrieves the complete hydraulic state (nodal heads) of the network based on its topology and limited pressure measurements. To perform leak localization, GSI is coupled with a process that compares interpolated leak and leak-free states. This article presents a methodology to adapt GSI in order to learn from its off-line and on-line operation (i.e., gain knowledge about historical located leaks, as well as leaks appearing in the network) and using this information to improve localization in future leak events. The methodology is tested over a well-known case study (Modena), showing promising results in terms of localization performance.**

I. INTRODUCTION

One of the major challenges faced by water utilities is the occurrence of bursts and leakage in water distribution networks (WDN). These faults cause high economical, social, environmental and even sanitary costs, justifying the interest in leak detection and localization techniques, which reduce both the volume of water losses and their associated impacts. When focusing on the leak localization problem, existing literature shows three main categories within the steady-state software-based methods: model-based, mixed model-based/data-driven and data-driven.

Model-based techniques rely on a hydraulic model of the WDN, which must be calibrated with respect to network properties and nodal consumption, in order to perform simulations to retrieve the network's hydraulic state. Over the years, this hydraulic model has been used for different purposes, such as studying the pressure sensitivity to the leak effects [1] and solving inverse network hydraulic problems [2], among others. Model-based methods operate well under ideal conditions, but their performance can be degraded by modelling and calibration errors and the complexity of WDNs and their associated mathematical models.

The advancement of machine learning and data analysis algorithms has significantly expanded their application for addressing the challenge of leak localization, leading to the development of mixed model-based/data-driven methods. They use the hydraulic model exclusively for training-sample generation, considerably mitigating their dependence on this model. Techniques such as artificial neural networks [3] or deep learning [4] have been successfully used to solve the isolation problem. However, calibration and modelling requirements still persist within these approaches.

Lately, data-driven approaches have emerged as a promising solution to address the aforementioned drawbacks, considering their independence from a hydraulic model. In this category, interpolation-based approaches such as [5] and [8] can be highlighted. They provide a satisfactory performance indicating search areas for the leak, but the localization is degraded when reducing those areas or even operating at node-level precision.

The main contribution of this article consists of the extension of the state reconstruction method presented in [8], i.e., graph-based state interpolation (GSI), to learn from real-time leak data. Data-driven leak localization methods using GSI suffered from performance degradation in small search areas, due to the approximations included within the reconstruction method. The novel scheme presented in this article uses the information from incoming leaks to learn how to adapt the original GSI process to improve the localization performance, thus minimizing the degradation related to the approximations of GSI while maintaining the data-driven philosophy. Other recent graph-based learning techniques such as [6] leverage the hydraulic model during training, thereby falling within the mixed model-based/data-driven category. The performance of the new methodology is evaluated in a realistic benchmark (Modena), showing promising results.

II. METHODOLOGY

A. *Preliminaries*

GSI plays a central role in the methodology outlined in this article. This interpolation algorithm models the WDN as a connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with $\mathcal{V}$ representing the set of nodes (reservoirs and junctions) and $\mathcal{E}$ denoting the set of edges (pipes). The i-th node is expressed as $v_i \in \mathcal{V}$, while the k-th edge is referred to as $e_k = (v_i, v_j) \in \mathcal{E}$, which connects the source v_i with the sink v_j. Each node in the graph carries an attribute, which in this case is linked to the steady-state hydraulic state of the network. The hydraulic head (pressure plus elevation) at the nodes of the network is considered as a representative of this state, due to the pressure drops produced by leaks and the reduced cost and easier installation of pressure sensors. The basic idea of GSI is to locally approximate the non-linear equation relating

*This work was supported by the Spanish national project L-BEST (Ref. PID2020-115905RB-C21), by MCIN/ AEI /10.13039/501100011033.

[1]Luis Romero-Ben, Gabriela Cembrano and Vicenç Puig are with the Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Llorens i Artigas 4-6, 08028, Barcelona, Spain `luis.romero.ben,gabriela.cembrano}@upc.edu`

[2]Vicenç Puig is also with the Supervision, Safety and Automatic Control Research Center (CS2AC) of the Universitat Politècnica de Catalunya, Campus de Terrassa, Gaia Building, Rambla Sant Nebridi 22, Terrassa, 08222, Barcelona, Spain `vicenc.puig@upc.edu`

the hydraulic heads of neighbouring nodes, e.g., the Hazen-Williams formula [7], by a linear relation, namely:

$$\hat{h}_i = \frac{1}{d_i}\boldsymbol{w}_i\hat{\boldsymbol{h}}, \tag{1}$$

where $\hat{\boldsymbol{h}} \in \mathbb{R}^{|\mathcal{V}|}$ is the state vector that approximates the hydraulic heads in the network. The terms $\boldsymbol{w}_i$ and d_i come from a pair of matrices encoding the topology of the network, namely the weighted adjacency matrix $\boldsymbol{W} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$, which weighs the connection between neighbouring nodes, and the degree matrix $\boldsymbol{D} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ (with $d_i = \sum_{j=1}^{|\mathcal{V}|} w_{ij}$ denoting the *i-th* element of the diagonal of the diagonal matrix $\boldsymbol{D}$). Let us remark that GSI weighs more the relation of closer adjacent nodes, and hence the inverse of the pipe lengths is selected to represent the graph weights, i.e., $\boldsymbol{W}$.

Thus, the GSI process is defined by the following optimization problem[1]:

$$\min_{\hat{\boldsymbol{h}},\gamma} \quad \frac{1}{2}\big[\hat{\boldsymbol{h}}^T\boldsymbol{L}\boldsymbol{D}^{-2}\boldsymbol{L}\hat{\boldsymbol{h}} + \alpha\gamma^2\big], \tag{2a}$$

$$\text{s.t.} \quad \boldsymbol{B}\hat{\boldsymbol{h}} \leq \gamma \cdot \mathbf{1}^{|\mathcal{V}|\times 1}, \tag{2b}$$

$$\gamma > 0, \tag{2c}$$

$$\boldsymbol{S}\hat{\boldsymbol{h}} = \hat{\boldsymbol{h}}^s, \tag{2d}$$

where $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$ is the Laplacian of $\mathcal{G}$. The edge-node incidence matrix is $\boldsymbol{B} \in \mathbb{R}^{|\mathcal{E}|\times|\mathcal{V}|}$, which assigns a value of 1 to its entry b_{kj} if $e_k = (v_i, v_j) \in \mathcal{E}$, -1 if $e_k = (v_j, v_i) \in \mathcal{E}$ and 0 otherwise[2]. The optimization variable γ is a positive slack value that constrains the sign of the difference in heads between adjacent nodes. The measured heads through the n_s installed sensors are stored in $\hat{\boldsymbol{h}}^s$, whereas matrix $\boldsymbol{S} \in \mathbb{R}^{n_s\times|\mathcal{V}|}$ is defined so that $s_{gj} = 1$ only if the $g\text{-}th$ sensor is located in node v_j, and 0 otherwise.

In this way, the solution of (2) pursues two goals. The main aim is the harmonization of the nodal states. To this end, the difference between the state of each node, and the approximation denoted by (1) must be minimized, i.e.:

$$\sum_{i=1}^{|\mathcal{V}|}\Big[\hat{h}_i - \frac{1}{d_i}\boldsymbol{w}_i\hat{\boldsymbol{h}}\Big]^2 = \big(\hat{\boldsymbol{h}} - \boldsymbol{D}^{-1}\boldsymbol{W}\hat{\boldsymbol{h}}\big)^T\big(\hat{\boldsymbol{h}} - \boldsymbol{D}^{-1}\boldsymbol{W}\hat{\boldsymbol{h}}\big) = \hat{\boldsymbol{h}}^T\big(\boldsymbol{I}^{|\mathcal{V}|} - \boldsymbol{D}^{-1}\boldsymbol{W}\big)^T\big(\boldsymbol{I}^{|\mathcal{V}|} - \boldsymbol{D}^{-1}\boldsymbol{W}\big)\hat{\boldsymbol{h}} = \hat{\boldsymbol{h}}^T\big(\boldsymbol{D}^{-1}(\boldsymbol{D} - \boldsymbol{W})\big)^T\big(\boldsymbol{D}^{-1}(\boldsymbol{D} - \boldsymbol{W})\big)\hat{\boldsymbol{h}} = \hat{\boldsymbol{h}}^T\boldsymbol{L}\boldsymbol{D}^{-2}\boldsymbol{L}\hat{\boldsymbol{h}}, \tag{3}$$

where $\boldsymbol{I}^{|\mathcal{V}|}$ is the identity matrix of size $|\mathcal{V}|$.

Additionally, a directionality-related goal is pursued through the minimization of γ^2 (2a) and the constraints (2b, 2c), considering that the head of the source node of a pipe should be higher than the head of the sink in WDNs.

Ultimately, the known hydraulic heads are used to feed the optimization problem with actual hydraulic information through (2d).

B. Generation of target states

GSI was originally conceived as the initial stage of an integrated fully data-driven leak localization scheme. In [8], a geometric-based comparative method, referred to as leak candidate selection method or LCSM, is proposed to serve as the secondary localization stage. Specifically, this process requires two input vectors to operate: the interpolated hydraulic state recovered from the measurements of the leak scenario under analysis, i.e., $\hat{\boldsymbol{h}}^{leak}$; and the interpolated state obtained from measurements from a nominal (leak-free) scenario with similar boundary conditions [1], i.e., $\hat{\boldsymbol{h}}^{nom}$. Each pair of analogue (in the same position) entries of the vectors are used as representatives of the x-y coordinates of a cloud of 2-D points. The best-fitting line to this cloud is computed, and the network nodes related to the furthest points to the line compose the set of candidates. A dynamic thresholding is used to limit the set size, considering the standard deviation of the point-to-line distance of the candidates. This process takes into account all the head values and their interconnections in the candidate set decision. Nevertheless, the relation between the leak and leak-free state of each node, that is, the pressure residual $\hat{r}_i = \hat{h}_i^{leak} - \hat{h}_i^{nom}$, plays a major role in the final decision.

Upon closer examination of the operational flow within the GSI-LCSM framework, a notable feature is the unidirectional flow of information from GSI to LCSM. Note that the current absence of a feedback mechanism neglects the potential of harnessing LCSM information to enhance the GSI performance from a isolation point of view. An advancement in GSI could be achieved by utilizing localization data to produce solutions in which the maximum pressure drop is close to the node where the leak occurs. Note that this cannot be normally imposed to GSI during real-time execution of the localization process, because the leak location is not available information but the objective of the localization process. However, once the leak is localized and fixed, the corresponding hydraulic measurements, which were used by GSI to retrieve the complete hydraulic state. can be associated to the actual leak location. Moreover, this association can be directly carried out if a historical dataset of leaks (containing hydraulic measurements and leak that caused them) is available.

Thus, the methodology proposed in this article includes a process that estimates the desired solution of GSI, in terms of maximum pressure drop location, when given the hydraulic measurements caused by a specific leak. Then, we can feed a learning process, trained to transform the solutions of standard GSI into the solutions of this new process. This would guarantee the maximum pressure drop resulting from the comparison between leak and leak-free scenarios to be located at the leak node. This maximum pressure drop would

[1] Note that the notation $x^{n\times m}$ denotes a matrix of size $n \times m$ with all its elements having a value of x.

[2] GSI adopts a structural approach to construct an approximated incidence matrix, considering for each pipe the most used direction when computing all the shortest-path between the water inlets and all the junctions. This is required due to the lack of flow data in water utilities, which would indicate the actual water directionality. See [9] for a detailed description of the algorithm to obtain the approximation of $\boldsymbol{B}$

be represented by the minimum value in the residual vector $\hat{\boldsymbol{r}}$. Thus, the objective of this process is to impose the following:

$$\boldsymbol{C}\hat{\boldsymbol{r}} \leq \hat{\boldsymbol{r}}, \tag{4}$$

where $\boldsymbol{C} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ is a sparse matrix that encodes the leak location as $\boldsymbol{C} = [\boldsymbol{0}^{|\mathcal{V}|\times(f-1)} \;\; \boldsymbol{1}^{|\mathcal{V}|\times 1} \;\; \boldsymbol{0}^{|\mathcal{V}|\times(|\mathcal{V}|-f)}]$, and f indicates the index of the leaking node within the node set $|\mathcal{V}|$, i.e., the leaky node is v_f.

Considering the presented constraint and the definition of the residual vector, the expression in (4) can be manipulated as follows:

$$\begin{aligned} \boldsymbol{C}\hat{\boldsymbol{r}} &\leq \hat{\boldsymbol{r}} \\ \boldsymbol{C}(\hat{\boldsymbol{h}}^{leak} - \hat{\boldsymbol{h}}^{nom}) &\leq (\hat{\boldsymbol{h}}^{leak} - \hat{\boldsymbol{h}}^{nom}) \\ \boldsymbol{C}\hat{\boldsymbol{h}}^{leak} - \hat{\boldsymbol{h}}^{leak} &\leq \boldsymbol{C}\hat{\boldsymbol{h}}^{nom} - \hat{\boldsymbol{h}}^{nom} \\ (\boldsymbol{C} - \boldsymbol{I}^{|\mathcal{V}|})\hat{\boldsymbol{h}}^{leak} &\leq \hat{\boldsymbol{y}}^{nom} \end{aligned} \tag{5}$$

where $\hat{\boldsymbol{y}}^{nom} = (\boldsymbol{C} - \boldsymbol{I}^{|\mathcal{V}|})\hat{\boldsymbol{h}}^{nom}$. Note that $\hat{\boldsymbol{h}}^{nom}$ is known, as the nominal interpolated vector can be retrieved from the leak-free measurements. Thus, $\hat{\boldsymbol{y}}^{nom}$ is also known.

The final expression in (5) is added as a constraint to the GSI formulation posed in (2), leading to the following adapted optimization problem, whose associated operation would be referred to as MPD/GSI (maximum pressure drop):

$$\min_{\hat{\boldsymbol{h}}} \quad \frac{1}{2}\big[\hat{\boldsymbol{h}}^T \boldsymbol{L}\boldsymbol{D}^{-2}\boldsymbol{L}\hat{\boldsymbol{h}} + \alpha\gamma^2\big], \tag{6a}$$

$$\text{s.t.} \quad \boldsymbol{B}\hat{\boldsymbol{h}} \leq \boldsymbol{1}_n \cdot \gamma, \tag{6b}$$

$$(\boldsymbol{C} - \boldsymbol{I}^{|\mathcal{V}|})\hat{\boldsymbol{h}} \leq \hat{\boldsymbol{y}}^{nom}, \tag{6c}$$

$$\gamma > 0, \tag{6d}$$

$$\boldsymbol{S}\hat{\boldsymbol{h}} = \hat{\boldsymbol{h}}^s. \tag{6e}$$

Remark 1. *This problem would only be solved for the leaky case, as the nominal case is solved through standard GSI. Thus, the terms $\hat{\boldsymbol{h}}^{leak}$ in* (5) *and $\hat{\boldsymbol{h}}$ in* (6c) *are equivalent.*♦

An important issue must be considered about MPD/GSI and its associated optimization problem. GSI-LCSM operates assuming that the maximum pressure drop occurs in the leak node. This holds in an ideal situation, but in practice there are sources of uncertainty that can alter this condition. Specifically, the residuals can be affected by the differences in boundary conditions between leak and leak-free scenarios (e.g., nodal consumption). Thus, actual measurements from the network, at specific time instants, might indicate larger residuals at areas where these discrepancies between leak and leak-free scenarios are occurring, instead of the leak area. This can be almost completely avoided by considering a time window of measurements and averaging over it.

Regarding (6), it is essential to avoid the introduced problem if the leak appeared in one of the nodes with a sensor. The solution may be not feasible if the maximum drop we are trying to impose is located in a sensorized nodes, due to a conflict between constraints (6c) and (6e). By inspection of (5), it can be appreciated that the constraint does not tolerate another sensor to have a lower (negative) residual than the sensorized leaky node. Thus, two possible solutions can be considered to avoid this problem:

1) Filter the gathered measurements to remove harmful time instants when the leak case occurred in a sensorized node.
2) Associate the corresponding time instants to a leak occurring in the nearest neighbouring node.

C. Knowledge integration

The exploitation of MPD/GSI lets us derive the required target interpolated vectors that GSI would ideally generate to optimize the leak localization performance. Then, the first layer that GSI needs to acquire knowledge about both past leaks, if historical data exist, and any future leak, has been presented in the previous section. The second layer should be constituted by a learning-based scheme, capable of using off-line and on-line information of leaks to improve the leak localization. Endless options can be proposed to play the role of the learning algorithm. Let us remark that GSI has already been successfully combined with learning stages in the past, e.g., Dictionary Learning [9].

Besides, note that at least two ways of applying the gathered leak knowledge can be devised:

1) On the one hand, a post-processing algorithm could be trained to convert already generated GSI states into MPD/GSI states. In this way, GSI would not be altered in the on-line application, but its output would be provided to the trained scheme, which would give a solution as similar as possible to MPD/GSI.
2) On the other hand, the leak knowledge can be used to alter one of the basis of GSI, that is the graph associated to the network. In this way, the properties of this graph and its associated structural matrices would be modified, so that GSI (applied over this new graph) would yield MPD/GSI solutions.

In this article, a first solution is proposed in the direction of the first option, although some insights about the second option are provided in the Conclusions section. The GSI-to-MPD/GSI process consists of a simple linear operation over the input GSI samples. If a target vector produced by MPD/GSI is referred to as $\hat{\boldsymbol{h}}^*$, then:

$$\hat{\boldsymbol{h}}^* \approx \boldsymbol{\Omega}\hat{\boldsymbol{h}}^{leak} + \boldsymbol{\beta}, \tag{7}$$

where $\boldsymbol{\Omega} \in \mathbb{R}^{|\mathcal{V}|\times|\mathcal{V}|}$ and $\boldsymbol{\beta} \in \mathbb{R}^{|\mathcal{V}|}$ are a diagonal matrix and a bias vector respectively.

These matrix-vector duple is obtained through an optimization process. In order to introduce the formulation of this problem, let us define the characteristics of the training dataset:

- First, we consider that leak information is only available for a subset $\mathcal{F} \subset \mathcal{V}$ of nodes. However, hydraulic data from non-labelled cases is also useful, because while we seek that the trained matrices transform known leak GSI vectors into MPD/GSI vectors, these trained matrices should minimize the degradation of GSI vectors from unknown leaks. Thus, we can define a target dataset $\hat{\boldsymbol{H}}^{targ}$, whose entries correspond to $\hat{\boldsymbol{h}}^*$ for labelled leak scenarios, and $\hat{\boldsymbol{h}}$ (i.e., from standard GSI) for the rest (non-labelled leak scenarios, nominal scenarios).

- Besides, we may have available samples associated with different time instants, obtained with the network presenting different boundary conditions. Note that the selection and availability of the elements in this set depend on several factors: sampling time of the sensors, the existence of a noise reduction pre-process that operates over a time window, etc. In this case, the set of time instants is represented as $\mathcal{T}$, and we denote the set of train samples representing a specific time instant as $\hat{\boldsymbol{H}}^{targ}(t)$.

Then, the optimization problem can be posed as follows:

$$\min_{\boldsymbol{\Omega},\boldsymbol{\beta}} \sum_{k=1}^{n_{train}} \sum_{t\in\mathcal{T}} \left[\hat{\boldsymbol{\delta}}(t)^T\hat{\boldsymbol{\delta}}(t)\right] + \tau\left(\|\boldsymbol{\Omega}-\boldsymbol{I}^{|\mathcal{V}|}\|_F + \boldsymbol{\beta}^T\boldsymbol{\beta}\right), \tag{8}$$

where $\hat{\boldsymbol{\delta}}(t) = \hat{\boldsymbol{H}}_k^{targ}(t) - (\boldsymbol{\Omega}\hat{\boldsymbol{H}}_k^{GSI}(t) + \boldsymbol{\beta})$, with $\hat{\boldsymbol{H}}_k^{targ}(t)$ being the k-th entry (column) of the dataset $\hat{\boldsymbol{H}}^{targ}(t)$ at time instant $t \in \mathcal{T}$ and $\hat{\boldsymbol{H}}_k^{GSI}(t)$ being the result of applying GSI to the same measured hydraulic information used to generate $\hat{\boldsymbol{H}}_k^{targ}(t)$. Note that the complete training dataset has a length of n_{train} samples.

Thus, the first term of of (8) seeks the minimization of the difference between the target states from the dataset and the states produced by (7). The second term pursues a solution that is as similar as possible to $\boldsymbol{\Omega} = \boldsymbol{I}^{|\mathcal{V}|}$ and $\boldsymbol{\beta} = \boldsymbol{0}^{|\mathcal{V}|\times 1}$, because this would imply the minimum possible degradation to the actual performance of GSI, protecting the solutions for non-labelled samples. The weight τ is settled to a low value, considering that this objective is also tackled by including non-labelled data in $\hat{\boldsymbol{H}}^{targ}(t)$.

D. General overview

The previous stages complement GSI, leading to a learning methodology that is able to gain knowledge from the existing and past leaks affecting the network. This approach is completed with a leak/leak-free comparison step, leading to a new leak localization methodology, henceforth referred to as Leak Learning GSI-LCSM (LL-GSI-LCSM). In this scheme, we can distinguish between the training and application processes. The operational flow of the training stage is represented by Algorithm 1.

Algorithm 1 Training — LL-GSI-LCSM

```
Require: $\hat{\boldsymbol{H}}^{s,leak}, \hat{\boldsymbol{H}}^{s,nom}, \mathcal{G} = (\mathcal{V}, \mathcal{E})$
  Compute $\hat{\boldsymbol{H}}^{GSI,nom}$ from $\hat{\boldsymbol{H}}^{s,nom}$ and $\mathcal{G}$ solving (2)
  Compute $\hat{\boldsymbol{H}}^{GSI,leak}$ from $\hat{\boldsymbol{H}}^{s,leak}$ and $\mathcal{G}$ solving (2)
  Compose $\hat{\boldsymbol{H}}^{GSI}$ from $\hat{\boldsymbol{H}}^{GSI,leak}$ and $\hat{\boldsymbol{H}}^{GSI,nom}$
  Divide $\hat{\boldsymbol{H}}^{GSI,leak}$ into $\hat{\boldsymbol{H}}^{GSI,\mathcal{F}}$ and $\hat{\boldsymbol{H}}^{GSI,\mathcal{F}^C}$
  Compute $\hat{\boldsymbol{H}}^*$ from $\hat{\boldsymbol{H}}^{s,\mathcal{F}}$, $\hat{\boldsymbol{H}}^{GSI,nom}$ and $\mathcal{G}$ solving (6)
  Compose $\hat{\boldsymbol{H}}^{targ}$ from $\hat{\boldsymbol{H}}^*$, $\hat{\boldsymbol{H}}^{GSI,\mathcal{F}^C}$ and $\hat{\boldsymbol{H}}^{GSI,nom}$
  Compute $\boldsymbol{\Omega}$ and $\boldsymbol{\beta}$ from $\hat{\boldsymbol{H}}^{targ}$ and $\hat{\boldsymbol{H}}^{GSI}$ solving (8)
  return $\boldsymbol{\Omega}, \boldsymbol{\beta}$
```

In the presented algorithm, $\hat{\boldsymbol{H}}^{s,nom}$ and $\hat{\boldsymbol{H}}^{s,leak}$ denote the measurements datasets for the nominal scenario and all the recorded leak events (labelled and non-labelled)[3]. Additionally, $\mathcal{F}^C$ represents the complement set of $\mathcal{F}$, i.e., $\hat{\boldsymbol{H}}^{GSI,\mathcal{F}}$ stores the entries for the labelled leaks, and $\hat{\boldsymbol{H}}^{GSI,\mathcal{F}^C}$ stores the rest of leak cases. Finally, note that the training process is performed considering several time instants, so that $\boldsymbol{\Omega}$ and $\boldsymbol{\beta}$ are not over-fitted for specific network boundary conditions.

The application of the algorithm, presented in Algorithm 2, is simple and easy to implement.

Algorithm 2 Application — LL-GSI-LCSM

```
Require: $\hat{\boldsymbol{h}}^{s,leak}(t), \hat{\boldsymbol{h}}^{s,nom}(t), \mathcal{G} = (\mathcal{V}, \mathcal{E})$
  Compute $\hat{\boldsymbol{h}}^{GSI,nom}(t)$ from $\hat{\boldsymbol{h}}^{s,nom}(t)$ and $\mathcal{G}$ solving (2)
  Compute $\hat{\boldsymbol{h}}^{GSI,leak}(t)$ from $\hat{\boldsymbol{h}}^{s,leak}(t)$ and $\mathcal{G}$ solving (2)
  Compute $\hat{\boldsymbol{h}}^{LL\text{-}GSI}(t)$ from $\hat{\boldsymbol{h}}^{GSI,leak}(t)$, $\boldsymbol{\Omega}$ and $\boldsymbol{\beta}$
  Obtain $\mathcal{C}$ from $\hat{\boldsymbol{h}}^{LL\text{-}GSI}(t)$ and $\hat{\boldsymbol{h}}^{GSI,nom}(t)$ using LCSM*
  return $\mathcal{C}$
```

During the on-line application, the leak localization algorithm can be applied to the incoming measurements at each time instant, i.e., $\hat{\boldsymbol{h}}^{s,leak}(t)$, considering the availability of an entry in the historical nominal dataset with similar boundary conditions, or the measurements of the previous instants to the leak detection, i.e., $\hat{\boldsymbol{h}}^{s,nom}(t)$. The candidate selection process, summarized in Section II-B, retrieves the set of node candidates to be the leak location, i.e., $\mathcal{C}$. This LCSM* process is adapted from the original by adding the normalized residuals to the distance-based metric computed in LCSM, due to the importance that LL-GSI-LCSM gives to the leak/leak-free residuals, considering that they mostly constitute the learning objective of MPD/GSI.

III. Case Study

In order to evaluate the performance of the methodology, the benchmark of the network of Modena (Italy) has been selected [10]. The benchmark was originally designed to correspond to a problem of realistic dimensionality, considering its physical size (268 nodes, 4 water inlets or reservoirs and 317 pipes, with a total pipe length around 72 km) and the nodal demands, which add up to around 400 l/s in total. The network topology is schematically represented in Fig. 1

The evaluation of the leak localization scheme requires the selection of a set of nodes to represent the installed sensors throughout the network. In this case, we have exploited a recent fully data-driven sensor placement methodology, which uses genetic algorithms to derive the sensors set that minimizes a topological-based metric, related to the sensor-to-node distance [11]. The cited research shows that the data-driven placement result was competitive in comparison to a sensitivity-based one, and thus we can derive a complete

[3] Note that detection algorithms can be applied over historical measurements data to classify the dataset entries into nominal and non-labelled leaks. This would not be required during the labelled leak scenarios.

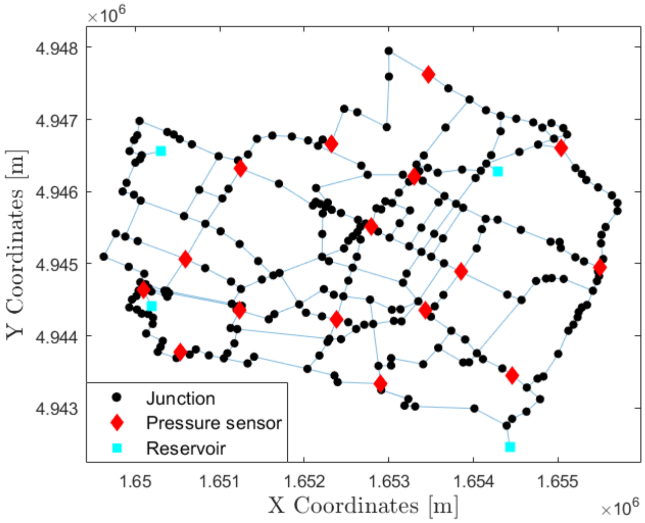

Fig. 1. Graph associated to the Modena WDN.

sensor placement + leak localization methodology which would not require a hydraulic model.

A. Generation of the hydraulic datasets

To comprehensively evaluate the method's performance under realistic conditions, several factors have been considered to configure the EPANET simulations that produce the evaluation hydraulic data:

i. A batch of simulations have been carried out to generate leak data for all the possible leak scenarios, i.e., varying the leak location among all the network junctions.
ii. Considering the average water inflow to the network ($\sim$400 l/s), a leak size of 2.5 l/s has been selected, as it only represents a 0.63% of that inflow. Each leak was simulated by configuring the emitter coefficient of the corresponding node in EPANET.
iii. The benchmark's demand patterns are adjusted for a timestep of 1 hour, and hence simulations of 24 hours have been performed.
iv. These demand patterns have been altered with a random noise of 1% in comparison to the noise-free reference, introducing uncertainty in the consumption of the users.
v. Extra random uncertainty of 1% have been considered in the pipe diameters and roughness coefficients, considering the higher difficulty of obtaining exact values (in comparison to pipe lengths).
vi. Finally, the pressure sensors are considered to provide a precision of ±1 cm, mitigating the impact of leaks that occurred below that precision in the sensor readings.

Similar settings have been used to assess methods within the state-of-the-art, such as [8] and [9].

IV. Results

Once the evaluation dataset was generated, the performance of the leak localization method could be tested. To this end, the dataset was divided into training and testing:

- For training, 10 hours out of 24 were selected, sampling over different times of the day to learn the leak effects while considering the variability of the demand patterns.
- For testing, a time instant that was not included among the training ones was selected, so that the trained method faces a data entry that has different boundary conditions from those used during learning.

Furthermore, three different scenarios are considered regarding the amount of available labelled leaks: 10, 70 and 200, which represent a 3.7%, 26.1% and 74.6% of the potential leaks. This selection allows us to explore scenarios with low, medium and high density of labelled leaks, enabling the analysis of their impact in the localization performance. These labelled leaks were placed using the sensor placement method in [11], ensuring that the leaks are scattered throughout the WDN.

The localization results are presented using two types of metrics:

- Accuracy-based: this metric evaluates the performance in terms of classification accuracy. Standard GSI produces search areas because of the limited performance at node-level. Thus, the accuracy has been measured in this article by considering seven levels of successful-localization area, starting at node-level and ranging from 1 to 6-degree-neighbours[4]. Note that the accuracy result for k-degree-neighbours[5] (henceforth referred to as k-D-N) is computed as the proportion of leaks scenarios in which the best candidate from the candidates set (computed by the localization method) is included inside the k-neighbourhood of the leak. The term ”best candidate” stands for the node in the candidates set that is given the highest probability of being the leak location by the localization stage.
- Distance-based: they measure the performance in terms of distance from the candidates set to the actual leak location. In this case, four metrics are presented:
  1) *Best*: Euclidean distance from the best candidate to the leak location.
  2) *Min*: Euclidean distance from the closest candidate (within a set of the 5 best candidates) to the leak.
  3) *Mean*: mean Euclidean distance from the set containing the 5 best candidates to the leak.
  4) *Max*: Euclidean distance from the furthest candidate (within a set of the 5 best candidates) to the leak.

The accuracy-based results are presented in Table I, whereas the distance-based results are displayed in Table II. In these tables, the first row displays the standard GSI results (and hence it uses zero labeled leaks), whereas the rest are obtained through LL-GSI. The displayed results show how the learning of a batch of labelled leaks helps in both increasing the accuracy and reducing the candidate-

[4] In Modena, the average 1 to 6-degree-neighbour areas represent respectively a 1.25%, 2.55%, 4.36%, 6.74%, 9.75% and 13.32% of $\mathcal{V}$.

[5] The set $\mathcal{N}(k,i)$ of k-degree-neighbours of a node v_i is defined as $\mathcal{N}(k,i) = \{j \mid \psi_{ij} > 1\}$, where $\boldsymbol{\Psi} = \boldsymbol{I}^{|\mathcal{V}|} + \sum_{n=1}^{k} \prod_{m=1}^{n} A$, and A is the combinatorial adjacency matrix of the graph.

to-leak distance, confirming the promising performance of the new methodology, and justifying the implementation of the additional layers plugged into GSI.

TABLE I

ACCURACY-BASED LEAK LOCALIZATION PERFORMANCE.

Labeled leaks	**Degree-of-neighbours**						
	0	1	2	3	4	5	6
0*	5.6	14.55	27.99	38.06	49.63	57.46	65.57
10	5.6	16.04	28.73	39.93	51.87	59.7	67.16
70	5.6	16.42	29.1	40.3	51.87	59.7	67.16
200	5.97	17.16	30.6	39.93	51.49	59.33	67.16

TABLE II

DISTANCE-BASED LEAK LOCALIZATION PERFORMANCE.

Labeled leaks	**Distance-to-leak (m)**			
	Best	*Min*	*Mean*	*Max*
0*	1081	745	1073	1426
10	1039	751	1062	1367
70	1035	747	1062	1357
200	1033	746	1064	1356

Additionally, regarding the number of labelled leaks, several conclusions can be drawn:

- In the accuracy-based metric, two interesting effects occur when increasing the labelled leaks. First, the comparison between 10 and 70 leaks show an improvement in the 1/2/3-D-N cases, while the rest remain the same. This is caused by non-successfully localized leaks in the 10-learned-leak case being successfully classified in the 70-learned-leak case, within those levels of degree-of-neighbour areas. Then, the comparison between the cases of 70 and 200 learned leaks cases show another improvement of the previous D-N areas, including now the node-level case, as well as a decrease in the 4/5-D-N. This is produced by new non-localized leaks starting to be correctly isolated, as well as leaks that were only correctly localized at 4/5-D-N level upgrading to a 0/1/2 or 3 D-N level.
- In the case of the distance-based metrics, the *Best* and *Max* cases continuously decrease, meaning that the most likely leak location suggested by LL-GSI is closer to the leak, and implying a reduction of outliers within the 5 best candidates, which would increase the *Max* metric. Note that the *Min* metric was deteriorated by LL-GSI with respect to GSI. This is not desirable, but it is an effect of LL-GSI effectively grouping the best candidates, which is a consequence of learning with the MPD/GSI targets (this is also the cause for the improvement in *Max*). Nevertheless, the addition of more labelled leaks helped reduce this degradation.

V. CONCLUSIONS

This article presents a leak localization methodology that adapts GSI, a state-of-the-art data-driven method, to learn from historical and future leak data. The proposed strategy computes target states from GSI obtained ones, which locate the maximum pressure drop at the known leaky node. Then, a learning scheme can be trained using the generated targets to convert GSI samples to MPD/GSI samples, which lead to a better localization when compared to the nominal states.

The methodology was evaluated using the Modena benchmark. To this end, realistic conditions were imposed to generate the evaluation data. The results showed how LL-GSI improves GSI in terms of both localization accuracy and candidate-to-leak distance. This confirms the promising performance of the methodology.

Several improvements can be performed to the methodology in the future. First, MPD/GSI can still be enhanced to produce state vectors which are not only better for localization purposes, but also closer to the actual head distribution in the network. Moreover, the learning stage can be extensively improved, considering that this paper only presents a simple approach to show the benefits of learning labelled leaks, but a wide variety of learning schemes can be plugged into this stage. Additionally, future work lines will follow the second way of applying the gained knowledge, introduced in Section II-C, by means of a matrix or set of matrices that minimally alter the graph weighted adjacency matrix to make GSI produce MPD/GSI solutions.

REFERENCES

[1] R. Pérez, V. Puig, J. Pascual, J. Quevedo, E. Landeros and A. Peralta, Leakage isolation using pressure sensitivity analysis in water distribution networks: Application to the Barcelona case study, IFAC Proceed. Vol., vol. 43, no. 8, 578-584, 2011.

[2] D.B. Steffelbauer, M. Günther, D. Fuchs-Hanusch, Leakage Localization with Differential Evolution: A Closer Look on Distance Metrics, Procedia Eng., vol. 186, pp. 444-451, 2017.

[3] M. Capelo, B. Brentan, L. Monteiro and D. Covas, Near–real time burst location and sizing in water distribution systems using artificial neural networks, Water, vol. 13, no. 13, pp. 1841, 2021.

[4] J., Li, W., Zheng and C., Lu, An accurate leakage localization method for water supply network based on deep learning network, Water Resour. Manag., vol. 36, pp. 2309–2325, 2022.

[5] A. Soldevila, J. Blesa, T. N. Jensen, S. Tornil-Sin, R. M. Fernandez-Canti and V. Puig, Leak localization method for water-distribution networks using a data-driven model and Dempster–Shafer reasoning, IEEE Trans. Control Syst. Tech., vol. 29, no. 3, pp. 937–948, 2020.

[6] G. Ö., Gardarsson, F., Boem and L. Toni, Graph-Based Learning for Leak Detection and Localisation in Water Distribution Networks, IFAC-PapersOnLine, vol. 55, no. 6, 661-666, 2022.

[7] C. P. Liou, Limitations and proper use of the Hazen-Williams equation, J. Hyd. Eng., vol. 124, no. 9, pp. 951-954, 1998.

[8] L. Romero-Ben, D. Alves, J. Blesa, G. Cembrano, V. Puig and E. Duviella, Leak Localization in Water Distribution Networks Using Data-Driven and Model-Based Approaches, J. Water Resour. Plan. Manag., vol. 148, no. 5, 04022016, 2022.

[9] P. Irofti, L. Romero-Ben, F. Stoican and V. Puig, Learning Dictionaries from Physical-Based Interpolation for Water Network Leak Localization, IEEE Trans. Control Syst. Techn., 2023.

[10] C. Bragalli, C. D'Ambrosio, J. Lee, A. Lodi and P. Toth, On the optimal design of water distribution networks: a practical MINLP approach, Optim. Eng., vol. 13, 219–246, 2012.

[11] L. Romero-Ben, G. Cembrano, V. Puig and J. Blesa, Model-free sensor placement for water distribution networks using genetic algorithms and clustering, IFAC-PapersOnLine, vol. 55, no. 33, pp. 54–59, 2022.