

Article

Reconstructing Sewer Network Topology Using Graph Theory

Batoul Haydar ^{1,*}, Nanée Chahinian ¹ and Claude Pasquier ²¹ HSM, Univ Montpellier, CNRS, IRD, Montpellier, France; nanee.chahinian@ird.fr² Université Côte d'Azur, CNRS, I3S, Nice, France; claude.pasquier@univ-cotedazur.fr

* Correspondence: batoul.haydar@ird.fr

Abstract

To manage sewer networks, reliable data is needed, which is often challenging. This study proposes a novel methodology to reconstruct the sewer network topology using graph theory. Two core procedures—flow adjustment and edge addition—re-establish hydraulically consistent flow paths and restore connectivity in disconnected portions of the network by reversing and adding links. The proposed approach operates at the pipe level, repairing directional reachability. It leverages only the existing network topology to reconstruct connectivity, guided by the principle that every node must have a downstream path to an outlet. The methodology is first applied to reconstruct the sewer network of Montpellier Metropolis in the South of France. Then it is validated by deliberately removing and reversing edges and applying the algorithms to test the methodology's capability in recovering the correct topology. Both methods performed well individually, especially at lower percentages of reversal (1%) and removal (1%), with a correctness of 0.99 for flow adjustment and 0.8 for edge addition. Although the results were poorer when combining the methods and increasing data degradation—particularly at 10% reversal and 10% removal (correctness of 0.64)—the methodology continued to produce a functionally consistent and logically coherent network, highlighting its robustness given the absence of supporting attribute data.

Keywords: sewer networks; graph theory; topology; hydraulic conveyance; urban catchments

1. Introduction

As cities worldwide experience rapid growth, the demand for a robust infrastructure to support the evolving needs of urban residents becomes increasingly important. Sewer networks are one of the crucial components of the city's infrastructure that ensure the proper drainage of rainwater and the collection and transportation of wastewater to treatment facilities. These systems are designed to ensure public health and safety, environmental protection, and sustainable development [1]. To minimize life-cycle costs and disruptions to urban life and the environment, effective management and planning are essential, supported by modeling techniques that accurately assess and optimize network operations [2]. Robust modeling, in turn, depends on reliable data describing the network's spatial layout, geometry, and hydraulic characteristics [3]. Yet obtaining such detail is often difficult and impractical: field surveys are labor-intensive and time-consuming, older catchments may lack reliable records, and some information is restricted for confidentiality [4,5]. As a result, sewer network data are frequently limited in availability, applicability, and quality [6,7], which strongly constrains the accuracy and reliability of the resulting model simulations [8]. This limitation must be addressed through targeted data



Academic Editor: Stefano Alvisi

Received: 12 December 2025

Revised: 7 January 2026

Accepted: 12 January 2026

Published: 14 January 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

reconstruction and validation approaches. Common data issues in sewer networks can manifest in various forms, ranging from missing or inaccurate attribute values—such as pipe diameters or slopes—to topological errors related to incorrect or missing connections between network components.

Previous research has tackled the issue of data incompleteness and inaccuracy in sewer networks, proposing methods to reconstruct missing attributes and improve network connectivity. Hajibabaei et al. [9] provided an automated graph theory framework for reconstructing missing physical attributes (diameters, slopes) in urban drainage networks using network hierarchy and hydrodynamic modeling. Belghaddar et al. [10] also completed missing attribute values using machine learning models, leveraging the topological relationships inherent in sewer networks, while Díaz et al. [11] developed a graph neural network metamodel that efficiently predicts stormwater nodal depths from limited simulation data. Harvey and McBean [12] predicted the structural condition of uninspected pipes with random forests, addressing just the asset-condition gaps. Other studies inferred the stormwater topology from streets, Digital Elevation Models (DEMs), and land use via graph metrics, recovering network connectivity [13]. Chahinian et al. [14] proposed a method to reconstruct or generate the sewer (wastewater) network using manhole cover locations derived from surveys, aerial imagery, or databases as the nodes of the network. Et-Targuy et al. [15] introduced a graph-based representation that helps correct or reveal disconnected components, ensuring the network's topological consistency. Duque et al. [16] applied mathematical optimization frameworks, integrating elevation, street topology, and inflow data to select or design sewer layouts that are both cost-effective and hydraulically consistent.

Sewer datasets are often repaired by imputing or predicting attribute values, yet such efforts rarely address the core issue of global connectivity. Other works infer sub-surface layout from surface proxies (street networks, DEM-derived slopes, land use/land cover), but these simplifications can erase the network's rich topological structure; moreover, inferring pipe flow from surface slope is particularly unreliable in flat terrain or hydraulically regulated Sections [17,18]. Recent modeling evidence shows that network completeness—i.e., topological connectivity—materially conditions pluvial-flood predictions: pruning by where it alters the maximum flood extents unless primary collectors and inlet information are preserved [19]. Also, the documented fragility of inspection/attribute fields, where consistency and uniqueness issues are frequent [20], makes connectivity recovery from attributes such as diameter, slope, or invert elevations unreliable. In fact, it is estimated that 25% and 50% of network information could be discarded [20–23]. Despite occasional topological gaps, topology is more self-auditing: outlet locations (e.g., treatment plants, discharge structures, pumping stations) provide stable anchors, and structural rules (e.g., downstream reachability to an outlet) enable systematic detection of inconsistencies and targeted repair. Together, these considerations position a topology-first repair as both necessary and novel: correcting connectivity and reducing dependence on error-prone attributes. The present study advances a topology- and geometry-driven reconstruction that restores pipe-level connectivity, coherent reachability, and flow direction with minimal edge additions, producing well-connected networks without relying on attributes.

The paper is structured as follows: Section 2 describes the proposed methodology and the steps involved. The application of the methodology to real-world data is presented in Section 3 including evaluation and limitations, and the conclusions are summarized in Section 5.

2. Materials and Methods

A well-connected sewer network ensures the efficient conveyance of wastewater and stormwater towards designated outlets such as treatment plants or natural discharge points.

The nodes in sewer networks can be divided into 3 types: inlet, with only outgoing edges (representing the physical pipes of the network)—source node (can be inlets, manholes); outlet, with only incoming edges—sink node (can be treatment plants, discharge structure); and internal node, with at least one incoming and one outgoing edge (manholes, pumping stations, structures, apparatuses). To maintain the hydraulic gradient and ensure that the flow follows the intended downstream path, resolving any disconnections, closed loops, or dead ends is crucial.

Common topological problems that impede continuous flow in the network are categorized as: (i) Flow direction issues, which include topological structures that violate the required downstream flow, such as cycles and improper sink nodes. These defects interrupt the directed path to the final discharge point; (ii) Connectivity issues that primarily manifest as missing links or dead ends, resulting in isolated subnetworks (or “islands”) that lack a directed path to a functioning outlet.

To address these issues, our methodology sequentially applies strategies to either (i) adjust the flow direction (via edge reversal) and/or (ii) add missing connections. The core principle guiding this approach is to ensure that every node possesses a directed path to an outlet, thereby confirming uninterrupted flow across the entire network.

In this study, the sewer network is represented as a directed multigraph (MultiDi-Graph) [24]. Nodes correspond to network elements (e.g., manholes, treatment plants, pumping stations), while edges represent pipes or conduits. The multigraph structure allows multiple directed edges between the same pair of nodes, reflecting parallel or distinct pipes with different attributes (e.g., diameter, slope, flow condition). This representation enables the application of graph theory concepts and algorithms to study the network’s structure and behavior. Mathematically, the network can be expressed as:

$$G = (V, E)$$

where:

- V is the set of nodes, with each $v \in V$ representing a physical element of the sewer system,
- $E \subseteq V \times V \times K$ is the set of directed edges, where multiple edges between the same ordered pair of nodes (u, v) are distinguished by a key $k \in K$.

Thus, each edge can be denoted as:

$$e = (u, v, k) \quad \text{with} \quad u, v \in V, k \in K$$

and is associated with attributes such as pipe length, diameter, material, or flow type.

The workflow chart in Figure 1 illustrates the sequential steps of the proposed methodology. The process begins by validating outlet reachability for all nodes. If any node cannot reach an outlet, the methodology first applies the «Flow adjustment» algorithm to resolve flow direction issues. If non-reachable nodes persist after flow adjustment, the «Edge Addition» algorithm is then utilized to connect isolated subnetworks by adding missing links.

2.1. Flow Adjustment

This process begins with the decomposition of the network graph G into its weakly connected components (WCCs). The workflow for this method is presented in Figure 2. In graph theory, a WCC is a maximal subgraph in which a path exists between any two nodes when edge direction is ignored [25]. In the context of urban water networks, this represents a physically connected part of the network, though it does not guarantee a directed flow path to a single discharge point. Let $GRP = \{G'_1, \dots, G'_m\}$ be the collection of WCCs in

G. Each component G'_i , $1 \leq i \leq m$ is processed independently to resolve flow direction conflicts. The following procedure details the processing of a single component G'_k . Let $S(G'_k)$ denote the set of sink nodes (nodes with zero out-degree) within G'_k .

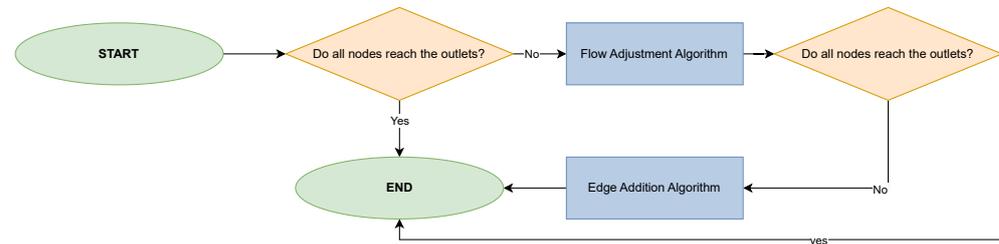


Figure 1. Workflow chart for the methodology for network construction.

Case 1: if $|S(G'_k)| = 1$, the flow is naturally directed toward a single node. This represents a topologically correct configuration and the component is left unchanged.

Case 2: if $|S(G'_k)| > 1$, physically connected nodes flow into two or more distinct sink nodes, indicating an abnormal flow configuration. The algorithm attempts to unify the flow by orienting all edges toward a single selected sink node denoted $S(G'_k)_t$. If one of the sink nodes represents a real outlet, this node is chosen. Otherwise, the algorithm selects the sink node that collects flow from the maximum number of predecessor nodes.

Case 3: if $|S(G'_k)| = 0$, then all nodes have at least one outgoing edge. These correspond to closed loops. Since there is no outlet in the WCC, the component is left unchanged.

Once $S(G'_k)_t$ is selected as the node with the maximum number of predecessors in G'_k , the sub-network that correctly flows into it is the induced subgraph on its predecessor set, denoted G''_k . The remaining sink nodes in the WCC, defined by the set difference $S(G'_k)_{\text{res}} = S(G'_k) \setminus \{S(G'_k)_t\}$, must now be connected to this valid sub-network G''_k .

For each remaining sink node $s \in S(G'_k)_{\text{res}}$, the shortest path P from s to any node in G''_k is computed on the *undirected* topology of G'_k . Let $L(P)$ be the topological length of this path (i.e., the number of edges along P). This path P is considered for flow reversal only if its length $L(P)$ is less than a user-defined threshold, e , which specifies the maximum permissible number of edge reversals. If $L(P) < e$, the edges along P are reversed. The sub-network G''_k is then recomputed to reflect the new, expanded directed flow path, and processing proceeds with the next sink node in $S(G'_k)_{\text{res}}$. If $L(P) \geq e$, the reversal is not performed, and the node s remains unresolved within G'_k . Processing for G'_k is complete once all nodes in $S(G'_k)_{\text{res}}$ have been assessed.

Similarly, for each component G'_k , corresponding to closed loops $C(G'_k)$, the shortest path P_c is computed from all nodes in $C(G'_k)$ to G''_k and the globally shortest path P is then selected. The edges along P are reversed if $L(P) < e$.

Examples of the flow adjustment algorithm are presented in Figure 3. The WCC in Figure 3a has 3 sink nodes, v_4 , v_7 and an *outlet*. The outlet is selected as the *TargetSinkNode* with the predecessor nodes v_1 , v_2 , v_6 , v_8 , and v_9 representing G''_k . Then, for the remaining sink nodes in the WCC, the shortest path to the G''_k is determined. For sink nodes v_4 and v_7 , the shortest paths each consist of a single edge— (v_2, v_4) and (v_8, v_7) , respectively. Thus, reversing the edges to (v_4, v_2) and (v_7, v_8) resolves the sink nodes and ensures flow continuity. In the second example in Figure 3b, WCC_1 has a single sink—the outlet—whereas WCC_2 has two sink nodes, v_3 and v_7 . v_3 is selected as the *TargetSinkNode*, because it has more predecessors; accordingly, its assigned predecessors are v_5 , v_4 , and v_6 representing G''_k . Although v_6 can reach both v_3 and v_7 , it is assigned exclusively to v_3 , and therefore v_7 has no predecessors. As the shortest path to G''_k is (v_6, v_7) , reversing this edge will allow the flow from v_7 to proceed to v_3 where an edge should be added to create a path to an *Outlet* to preserve hydraulic continuity.

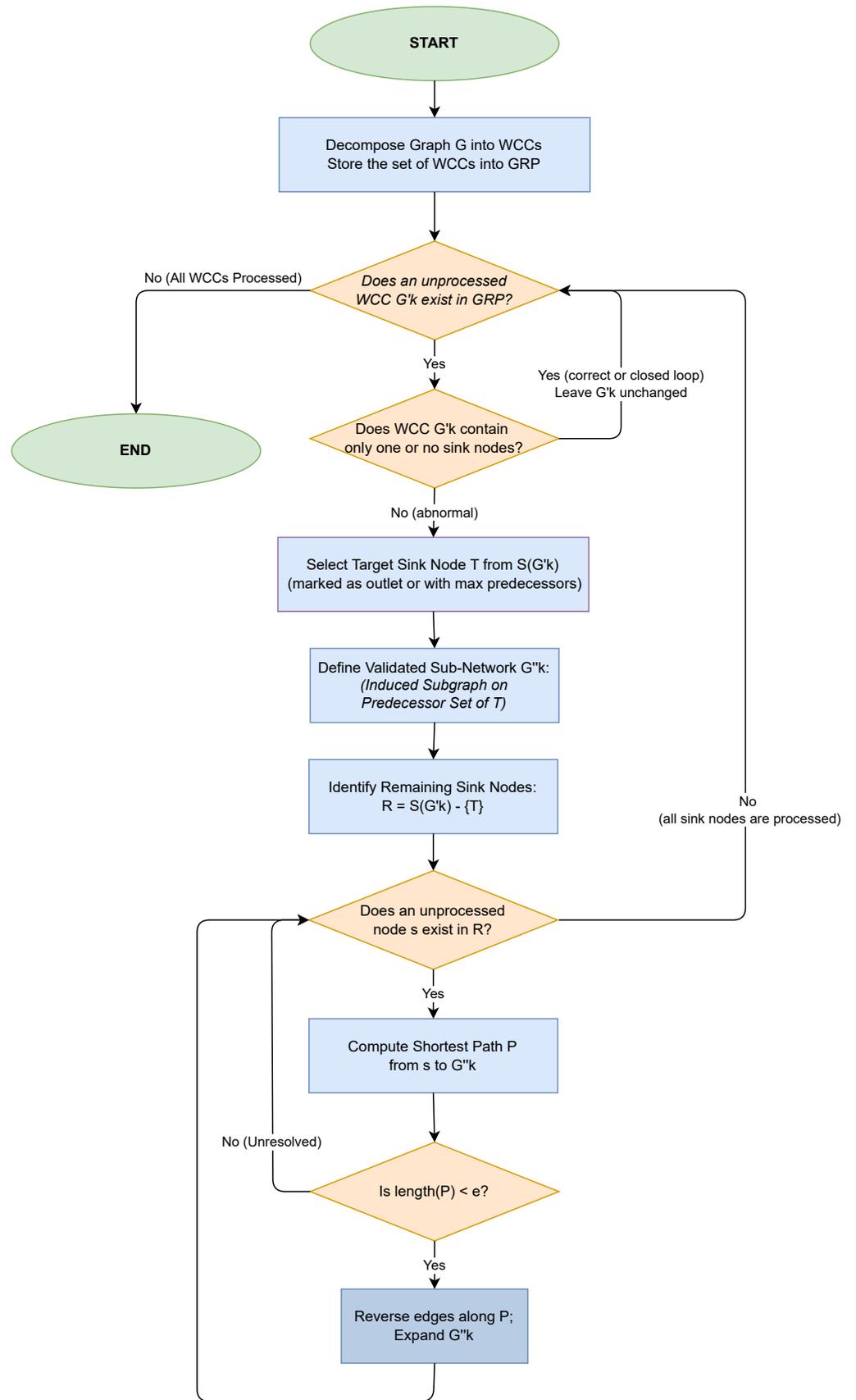


Figure 2. Workflow chart for the flow adjustment algorithm.

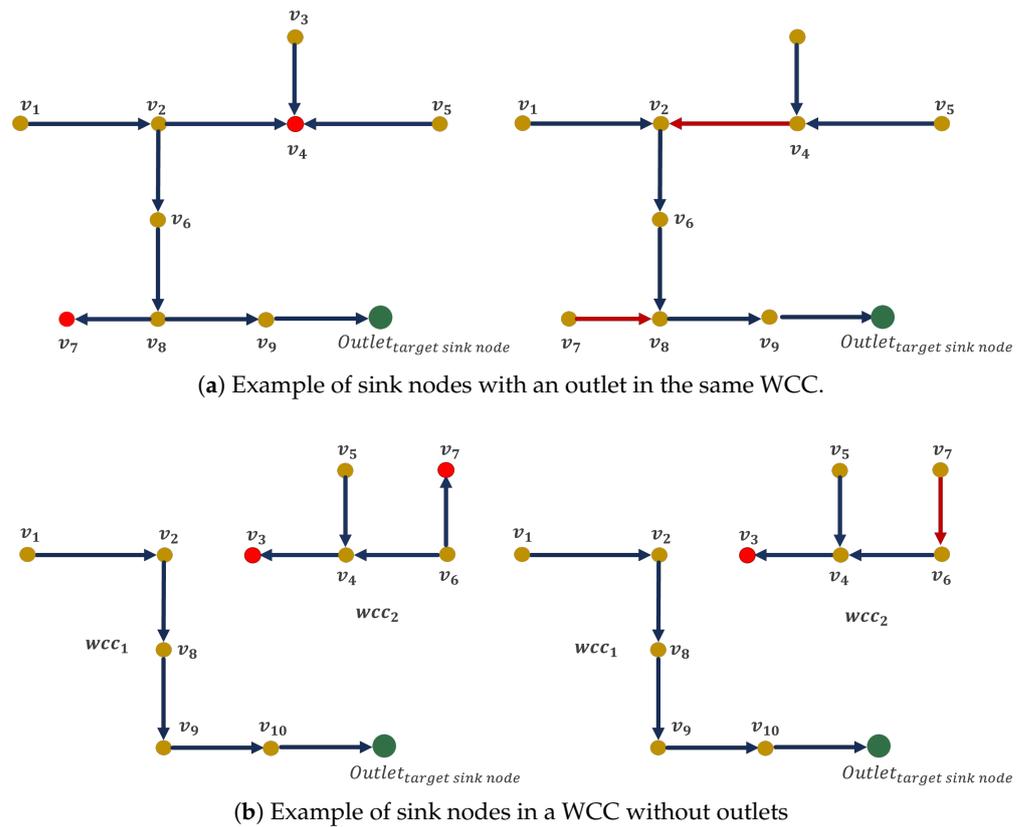


Figure 3. Examples of sink node correction. Red dots represent sink nodes, red arrows represent reversed edges.

2.2. Edge Addition

Disconnections in the network may arise from missing pipes, leaving some nodes unlinked to the rest of the system. Previous sections focused on reversing edges to restore flow continuity; when this is not possible, edges must instead be added.

The main challenge is defining edge-addition criteria in the absence of elevation and slope data. Distance should be part of the decision, but the nearest node is not necessarily the correct connection. In practice, sewer pipes are typically aligned with streets [13] to avoid crossing private property, limit disruption, and improve safety and aesthetics. Leveraging the road network as a fundamental spatial layer therefore provides a practical basis for inferring new connections.

The workflow chart for this method is presented in Figure 4, which outlines the steps taken to connect the remaining sink nodes to the outlets.

Edges are added from a sink node s to a candidate node c in the reachable node set (nodes that have a directed path to an outlet). When an edge is added (s, c), that sink node (and all its upstream predecessors) now inherits the candidate's reachability. In other words, they become part of the reachable set. Because each successful addition makes the reachable set grow, more sinks may now find nearby candidates that were not part of the reachable set before. The algorithm starts the search with a small cutoff which is only increased when no new edges are added (i.e., the reachable set did not grow). Starting with a small cutoff favors short, local, and likely correct connections. Gradually relaxing the cutoff avoids unnecessarily long links until all closer options have been exhausted.

The process begins by building the candidate set C from the reachable nodes and finding the set of sink nodes S . Then, for each s in S , the road accessibility is checked, i.e., by verifying that the perpendicular distance from s to the nearest road segment does not exceed the snap tolerance ϵ_{snap} . Usually ϵ_{snap} represents the road width and the

sidewalk. If $\text{dist}_\perp(\mathbf{s}_n, \mathcal{E}_{\text{road}}) \leq \varepsilon_{\text{snap}}$, road-based distance to candidates is computed. If $\text{dist}_\perp(\mathbf{s}_n, \mathcal{E}_{\text{road}}) > \varepsilon_{\text{snap}}$, Euclidean distance to candidates is computed.

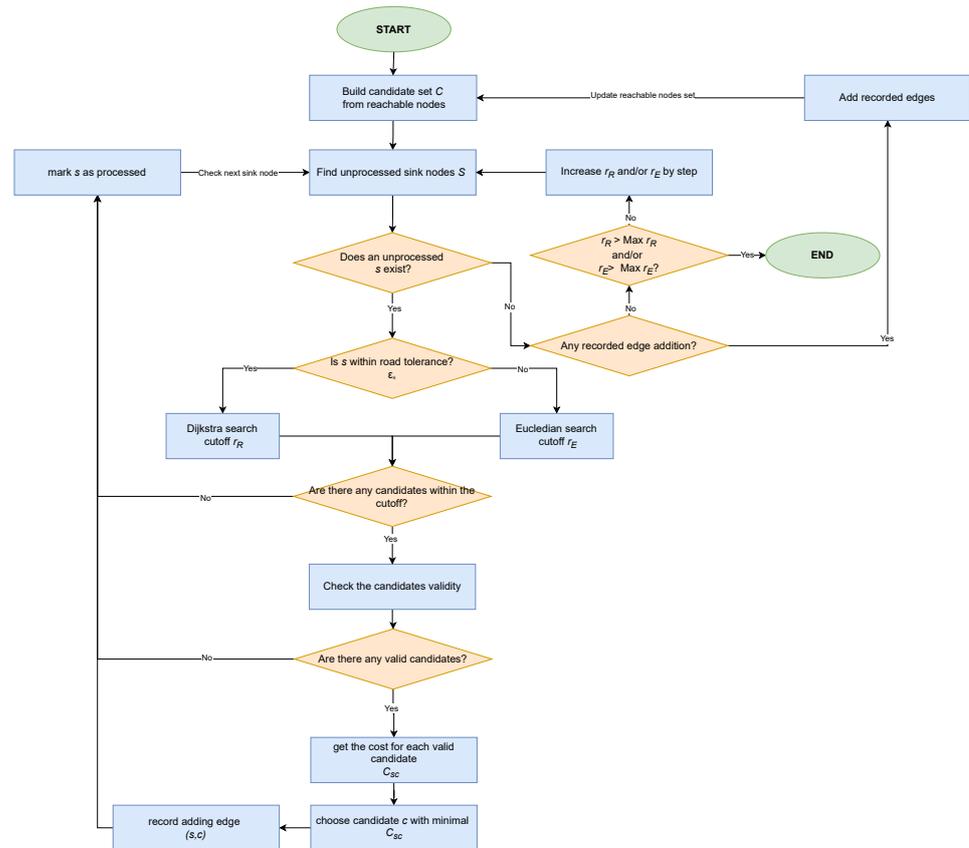


Figure 4. Workflow chart for edge addition algorithm.

The road-based distance is the shortest path-travel distance measured along the road network. To compute it, Dijkstra’s algorithm [26] is used, which is a graph search algorithm designed to find the shortest paths from a single source node to all other nodes in a weighted graph with non-negative edge. A Dijkstra single-source shortest-path search is executed on the road graph within the cutoff distance r_R to compute the shortest paths to all reachable road vertices. Candidates from the reachable set that are within radius r are then identified, snapped to the road edges if possible ($\text{dist}_\perp(\mathbf{s}_n, \mathcal{E}_{\text{road}}) \leq \varepsilon_{\text{snap}}$), and evaluated based on their road distance to the sink. If s cannot be snapped to the road, a fallback procedure is applied: candidate nodes are shortlisted based on their Euclidean distance within a cutoff radius r_E . If no candidates are found, whether s is on the road or not, the algorithm skips this node and checks another one. If s has candidates, their validity is checked against hydraulic and topological constraints.

The first check examines the circulation mode of the incoming edge of s and the outgoing edge of c . In a hydraulically consistent network, a gravity-fed pipe should not connect directly into a forced main unless s or c is of type *Pumping Station*. This requirement is universally enforced in hydraulic engineering design standards due to the need for a dedicated structure to manage the transition from low-pressure open-channel flow to high-pressure closed-conduit flow [27]. Thus, connecting s to c by directly connecting a gravity-fed pipe into a pressurized forced main without a proper hydraulic transition—such as a dedicated pump station, wet well, or a transition structure—is not hydraulically valid and not permissible within our methodology. Consequently, this candidate connection is deemed invalid.

Furthermore, a check is performed to ensure that adding the connection (s, c) does not create a closed loop or cycle within the network. In a gravity-driven system, this would violate the principle that flow must progress downstream toward an outlet; such cycles can trap or recirculate flow and break outlet reachability, making the reconstructed network hydraulically and topologically inconsistent. For pressurized/forced mains, loops can be intentionally designed, but they must be supported by a pump/valve; thus, c is considered invalid under gravity flow and only admissible when it is part of a forced flow. Similarly, if no valid candidates are found, it marks this node as processed and checks the next one.

If more than one valid candidate c exists, the one with the lowest cost is chosen. For each candidate edge connection (s, c) , the cost function combines length, angular consistency at both endpoints, and a node-degree penalty:

$$C_{sc} = w_L C_{L,sc} + w_\theta C_{\theta,sc} + w_{deg} C_{deg,sc} \tag{1}$$

with $w_L, w_\theta, w_{deg} \geq 0$ and typically $w_L + w_\theta + w_{deg} = 1$.

Let L_{sc} be the road or Euclidean length of the candidate edge (s, c) and L_{max} the maximum length allowed for the candidate edge. The length cost is defined as

$$C_{L,sc} = \frac{L_{sc}}{L_{max}} \tag{2}$$

so that shorter connections are systematically favored.

The angular criterion is designed to (i) avoid U-turns and acute deflections at the sink node and (ii) preserve a smooth continuation of the flow at the candidate node. To this end, two checks are done for a candidate edge (s, c) .

Let $\mathcal{I}(s)$ denote the set of nodes k such that (k, s) is an existing incoming edge to the sink s and $\mathcal{O}(c)$ the set of nodes ℓ such that (c, ℓ) is an existing outgoing edge from the candidate node c .

(1) Angles at the sink node

For each incoming edge (k, s) with $k \in \mathcal{I}(s)$, the angle θ_{ksc} is computed at node s between the existing edge (k, s) and the candidate edge (s, c) :

$$\theta_{ksc} = \angle(\vec{sk}, \vec{sc}) \tag{3}$$

where θ_{ksc} is expressed in degrees and defined such that 180° corresponds to a straight continuation of the flow, 90° to an orthogonal deviation, and small angles correspond to U-turn or acute configurations.

(2) Angles at the candidate node

For each outgoing edge (c, ℓ) with $\ell \in \mathcal{O}(c)$, the angle $\theta_{sc\ell}$ is computed at node c between the candidate edge (s, c) and the existing edge (c, ℓ) :

$$\theta_{sc\ell} = \angle(\vec{cs}, \vec{c\ell}) \tag{4}$$

again interpreted such that angles close to 180° represent a smooth continuation and acute angles represent sharp deflections.

(3) Local angular penalty

Each angle θ is mapped to a dimensionless penalty $C_\theta(\theta) \in [0, 1]$ to favor 180° alignments, apply a smooth linear ramp for intermediate turning angles, and strongly penalize acute and U-turn configurations:

$$C_{\theta}(\theta) = \begin{cases} 1, & \text{if } \theta < 45^{\circ}, \\ \frac{135^{\circ} - \theta}{90^{\circ}}, & \text{if } 45^{\circ} \leq \theta < 135^{\circ}, \\ 0, & \text{if } 135^{\circ} \leq \theta \leq 180^{\circ}, \end{cases} \quad \theta = \min(\theta_{\text{raw}}, 360^{\circ} - \theta_{\text{raw}}). \quad (5)$$

where the directed rotational angle $\theta_{\text{raw}} \in [0^{\circ}, 360^{\circ})$ and the hydraulic interior angle $\theta \in [0^{\circ}, 180^{\circ}]$, obtained as $\theta = \min(\theta_{\text{raw}}, 360^{\circ} - \theta_{\text{raw}})$.

The function gives no penalty to angles approaching 180° , applies a linear ramp through intermediate deviations near 90° , and assigns the maximum cost to acute and U-turn geometries close to 0° .

(4) Aggregate angular cost for the candidate edge

The angular cost at the sink node i is obtained as the worst (maximum) penalty over all its incoming edges:

$$C_{\theta}^{\text{sink}}(s, c) = \begin{cases} \max_{k \in \mathcal{I}(s)} C_{\theta}(\theta_{ksc}), & \text{if } |\mathcal{I}(s)| > 0, \\ 0, & \text{if } |\mathcal{I}(s)| = 0, \end{cases} \quad (6)$$

where $\mathcal{I}(s)$ is the set of nodes k such that (k, s) is an existing incoming edge.

Similarly, the angular cost at the candidate node c is defined as the maximum penalty over all its outgoing edges:

$$C_{\theta}^{\text{cand}}(s, c) = \begin{cases} \max_{\ell \in \mathcal{O}(c)} C_{\theta}(\theta_{sc\ell}), & \text{if } |\mathcal{O}(c)| > 0, \\ 0, & \text{if } |\mathcal{O}(c)| = 0, \end{cases} \quad (7)$$

where $\mathcal{O}(c)$ is the set of nodes ℓ such that (c, ℓ) is an existing outgoing edge.

The total angular cost associated with the candidate edge (s, c) is then taken as the worst violation over both endpoints:

$$C_{\theta,sc} = \max(C_{\theta}^{\text{sink}}(s, c), C_{\theta}^{\text{cand}}(s, c)), \quad (8)$$

so that a single strongly unfavorable angle, either with the incoming flow at the sink or with the outgoing directions at the candidate, is sufficient to assign a high angular cost to the candidate connection.

For the node-degree penalty, let $d_c = \text{deg}(c)$ be the degree of c in the current partial network (before adding (s, c)). Nodes with $d_c \leq 1$ are favored and connections to already crowded junctions are penalized. Given a tolerated upper bound d_{max} , the node-degree cost is defined as:

$$C_{\text{deg},sc} = \begin{cases} 0, & d_c \leq 1, \\ \frac{d_c - 1}{d_{\text{max}} - 1}, & 1 < d_c < d_{\text{max}}, \\ 1, & d_c \geq d_{\text{max}}. \end{cases} \quad (9)$$

In combination with (1), this term discourages attaching new connections to highly connected nodes, favoring connections that are both geometrically and topologically plausible.

Out of the valid candidates, the candidate with the minimal cost is chosen, and the edge between the sink and its candidate is recorded for addition. s is marked as processed to avoid checking it again, and the next s is processed.

When all sink nodes have been processed, the algorithm adds the edges recorded between s and c , finds the new reachable nodes set, rebuilds the candidate set, and then processes the remaining sink nodes for which no edge has been added. However, if no edges have been recorded, meaning no valid candidates are found for any sink node and the reachable nodes set remains the same, the cutoff search, whether for Dijkstra or Euclidean, is increased by a *step*, and the algorithm proceeds to reconnect s following the same steps described before. The algorithm proceeds until either S is empty or no additional edges have been added, and the cutoff distances have reached the maximum allowed. In this case, the algorithm stops.

On the other hand, closed loops, which are subsets of nodes that are strongly connected but have no path to any outlet, are treated as pseudo-sinks and evaluated by the algorithm cost function (all nodes in the loop are considered); however, only the node–candidate pair with the minimum total cost is retained, so a single edge is added to open the loop from the reachable set.

2.3. Evaluation Criteria

For validation purposes, a controlled test is conducted over a real-world sewer network in which a subset of edges is deliberately removed and reversed. The algorithms are then applied to this modified graph, and the resulting structure is compared with the original network to evaluate the method’s capability to recover the correct topology. The performance of the methodology is evaluated using the criteria of completeness, correctness and quality:

$$\text{Completeness} = \frac{TP}{TP + FN} \quad (10)$$

$$\text{Correctness} = \frac{TP}{TP + FP} \quad (11)$$

$$\text{Quality} = \frac{TP}{TP + FN + FP} \quad (12)$$

where TP represents True Positives (i.e., the number of correctly reversed or added pipes), FN False Negatives (number of pipes that should have been reversed or added, but the algorithm missed them), FP False Positives (number of pipes that are added or reversed by the algorithm but should not have been).

Completeness represents the percentage of the recovered share of the reference network. Correctness represents the percentage of correctly added/reversed pipes. Quality measures overall agreement, penalizing both omissions and incorrect additions/reversals. It takes into account completeness and correctness with the optimum value 1 for the three criteria.

The algorithm presented in this work is tested using a PC, 12th Gen Intel(R) Core(TM) i7-12800H (2.40 GHz) with 32.0 GB installed RAM. The code is available in Python (Version 3.10.8) on the GitHub repository <https://github.com/batoulhaydar/Reconstructing-Sewer-Network-Topology> (accessed on 11 January 2026).

3. Results

As a case study, the methodology is applied to the Montpellier Metropolis sewer graph to construct a well-connected network (Section 3.1).

3.1. Construction of Montpellier Metropolis Sewer Network

The Open Data of Montpellier Méditerranée Métropole provides information on the sanitation networks of the Montpellier Metropolis [28] (Figure 5). The data is updated

frequently and available in CSV, JSON, and shapefile formats. The wastewater network open data consists of seven files in ESRI shapefile format: one describing the pipes and their attributes (including diameter, material, circulation mode, and elevations) and six others representing different nodes (manholes, treatment plants, pumping stations, structures, and appurtenances) and their attributes. To build the graph, the RDF graph developed previously in [29] for the Montpellier sewer network (accessed in July 2023) is translated into a NetworkX graph [30], where subjects represent both nodes and pipes, and predicates define either connectivity or attribute relationships among them. A MultiDiGraph of the sewer network is constructed where the directed edges represent the pipes and their flow direction and the nodes represent the different structures and apparatus connecting the pipes.

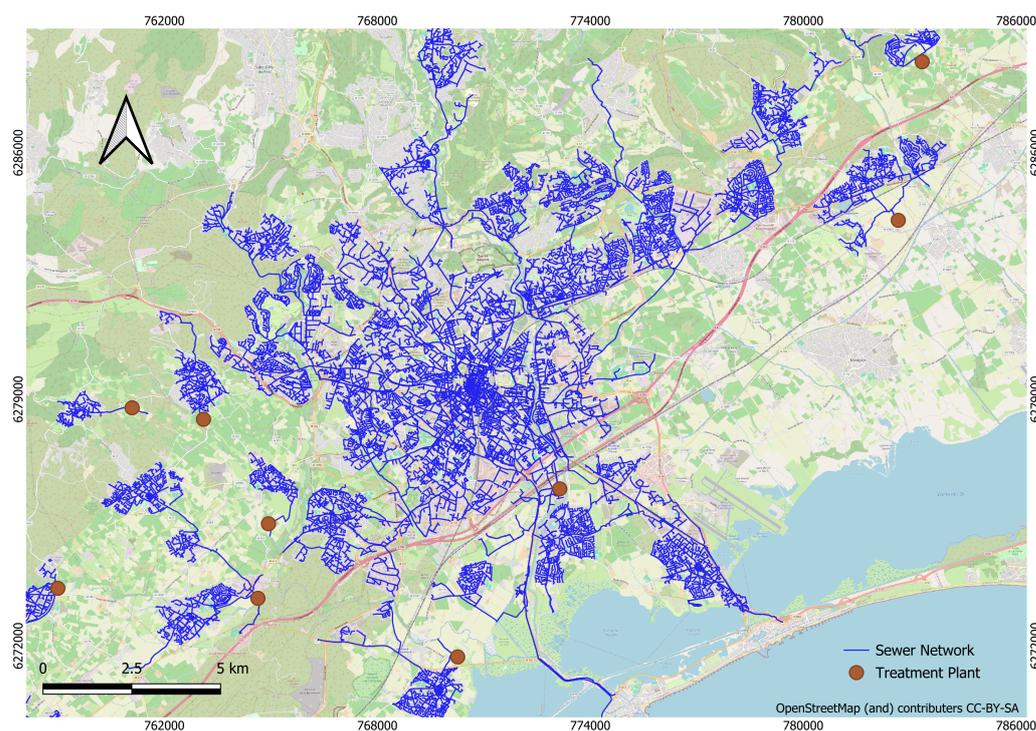


Figure 5. Sewer network of Montpellier Metropolis on an Open StreetMap background. Blue lines represent the network and brown dots represent the treatment plants.

A network preprocessing and cleaning step is carried out before applying the methodology. It includes verifying topological consistency, removing redundant elements, resolving missing nodes, and addressing isolated nodes to obtain a coherent and connected sewer graph. In case a pipe is missing one of its end nodes, a node is added to the edge and noted as ‘Added_initial_node_for_<pipeId>’ or ‘Added_terminal_node_for_<pipeId>’, ensuring that no repetition of nodes occurs between two consecutive pipes. Information such as material, diameter, and circulation mode on the pipes and elevations, coordinates and geometry on the nodes are added as attributes to the edges and the nodes, respectively. In some cases, the node is not missing but inaccurately digitized. The Euclidean distance is checked between the added node and the closest isolated node. If the distance is less than the diameter value of the edge, then the added node is replaced with the isolated node. In a graph, the edge connects two nodes, which is not necessarily the case in sewer networks that may include intermediate nodes. Depending on the type of the node, if it is an apparatus (e.g., valve), the intermediate node/s are represented as attributes of the edge, and if it is a structure (e.g., manhole), the edge is split into two, where the intermediate node then becomes an extremity for both resulting edges. As for the attributes, the length and geometry are recalculated for the new edges, while other attributes, such as material

and diameter, remain identical for both. At the end of the preprocessing step, and after adding and replacing nodes and removing isolated and redundant nodes and pipes, the sewer graph consists of 50,781 nodes and 50,889 edges, representing more than 1600 km of network length.

The first step of the methodology involves checking whether all the nodes reach the outlet. The sewer graph comprises 26 outlets, including 13 treatment plants and 16 discharge structures. Among these, three treatment plants discharge their treated effluent into a downstream discharge structure. Around 38% ($N = 19,465$) of the nodes, are non-reachable, so the flow adjustment algorithm is first applied. Parameter e —the number of edge reversals required to establish a valid path from the sink and to reach the outlet, respectively—is set to 5. The results of applying this algorithm are summarized in Table 1. Reversing only 135 edges resolved more than 60% of the unconnected sink nodes and restored outlet reachability for more than 28% of previously non-reachable nodes.

Then the edge addition algorithm is applied to add the missing connections using the road network of Montpellier Metropolis [31]. The road graph is built from the shapefiles, where edges represent the roads and the nodes are road vertices. Road attributes are added to the edges.

The snapping tolerance $\varepsilon_{\text{snap}}$ is defined for each road segment e as a function of its highway class h_e , width w_e , lane count L_e , sidewalk configuration, and a small positional margin m . Let s_e denote the effective sidewalk allowance (if any), $\bar{w}(h_e)$ the default lane width for motorized classes, and $\bar{W}(h_e)$ the default corridor width inferred from the highway class. Then

$$\varepsilon_{\text{snap}} = \begin{cases} w_e + s_e + m, & \text{if } w_e \text{ is known,} \\ L_e \bar{w}(h_e) + s_e + m, & \text{if } w_e \text{ is unknown and } L_e \text{ is known,} \\ \bar{W}(h_e) + s_e + m, & \text{if } w_e, L_e \text{ are unknown and } h_e \text{ is known,} \\ 10 \text{ m,} & \text{if } h_e \text{ is unknown.} \end{cases}$$

In practice, $\varepsilon_{\text{snap}}$ is thus derived from the highway class (e.g., residential, primary, cycleway, footway), the number of lanes and corridor width, and the presence of sidewalks; when none of these are available, a conservative default tolerance of 10 m is adopted.

The parameters for the edge addition algorithm chosen for this graph are presented in Table 2. We set the initial search thresholds for candidate connections to $r_R = r_E = 20$ m, representing a first pass that favors the nearest already-reachable nodes; both the Euclidean and Dijkstra searches are then allowed to expand up to a maximum cutoff of $r_{\text{max}} = 160$ m with a 10 m step. This upper bound is consistent with the Montpellier sewer technical guidance, which specifies a maximum spacing of approximately 80 m between adjacent manholes [32]; since our procedure adds only edges between existing nodes (no new nodes are created), a 160 m cap effectively corresponds to at most two missing pipe segments between two manholes or one pipe between other node types. Given that the number of residual sink nodes is small, we assume in practice that a single new pipe (one edge) will usually suffice to restore outlet reachability for each sink. If a connection would require a span exceeding 160 m, this would typically indicate the need to insert an intermediate node, which is outside the scope of the present method. As for the cost function, the parameters are in Table 3. $L_{\text{max}} = 160$ m is the same as r_{max} and $d_{\text{max}} = 4$ since studies show that real sewer networks are predominantly tree-like with most nodes of degree 2–3 [33,34]. Penalizing degree > 4 therefore filters out connections that contradict typical flow distribution and network topology. As for w_L , w_θ and w_{deg} weight parameters, they are set to 0.4, 0.4 and 0.2 respectively giving comparable priority to keeping connections

short and geometrically consistent, while using the degree term as a softer preference rather than a hard constraint.

Table 1. Results of applying the flow adjustment and edge addition algorithm to Montpellier sewer network.

Metric	Flow Adjustment Count	Edge Addition Count
Initial sink nodes	152	60
Reversed or added edges	135	46
Resolved sink nodes	92	46
Resolved reachable nodes	5477	13,869

Table 2. Parameters for the edge addition method.

Parameter	Value (m)
ϵ_{snap}	10
r_E	20
r_R	20
$step$	10
r_{max}	160

Table 3. Parameters for the cost function in the edge addition method.

Parameter	Value
w_L	0.4
w_θ	0.4
w_{deg}	0.2
d_{max}	4
L_{max}	160

The results of the algorithm are presented in Table 1 where 46 edges are added to the sink nodes, allowing 99% of the remaining nodes to reach the outlets. For the other 14 sink nodes, representing less than 1% of the remaining non-reachable nodes (119 nodes), no candidates were found either because the sink node lies far from the main network, or the candidates found are not valid based on the defined criteria.

Unresolved Cases

This section describes the cases where no candidates were found for the 14 remaining sink nodes. The first case, representing more than 50 % of the unresolved cases, is related to the maximum cutoff allowed. In Figure 6, the eight-node disconnected subgraph illustrates a case in which two additional pipes are insufficient to restore outlet reachability; introducing the necessity of node connections.

In one case, when the sink node is snapped to the road, the subsequent search is restricted to the road network where no valid candidate is found, thereby excluding candidate nodes located off-road.

For the remaining cases, representing 35 % of the cases, validity checks described before—U-turn, cycle, and circulation constraints—prevented the connections of the sink nodes.



Figure 6. Example of a small subgraph beyond the maximum cutoff where blue lines represent the network and red dots represent the manholes on an OpenStreetMap background.

4. Discussion

The constructed network is used to evaluate the proposed methodology by deliberately reversing and removing edges to assess whether the method could reconstruct the original topology.

4.1. Evaluation of Proposed Methodology

To evaluate the proposed methodology, the constructed sewer network of Montpellier is used, where the remaining non-reachable nodes described in Section “Unresolved Cases” are removed to provide a well-connected graph as a reference with 50,662 nodes and 50,829 edges. Flow adjustment and edge addition algorithms are first tested individually and then applied in combination. The first part of the evaluation examines the flow adjustment algorithm using varying edge-reversal percentages. At each reversal percentage, random reversal realizations ($n = 4$) are generated; the algorithm is applied to each, and performance is summarized by the metrics described in Section 2.3, obtained by comparison with the reference graph. The parameter e of the flow adjustment algorithm representing the maximum allowed number of edge reversals is chosen to be 5 for these tests.

The results in Table 4 show that even with only 1% of edges being misoriented, more than half of the nodes don’t have a path to the outlet and reaching 99% of the nodes when only 20% of the edges are reversed. Also, they show that the algorithm is significantly able to enhance node reachability even at higher reversal rates where only 14 % of nodes are left as non-reachable with 20% reversal.

Table 5 shows the results of the mean error of applying the algorithm under different percentages of edge reversals. A higher density of sink nodes makes correct edge reorientation more challenging, resulting in a slight decrease in completeness, correctness, and quality as the percentage of reversed edges increases. With more flow-direction errors, the task becomes less identifiable: the search space of valid orientations expands. Moreover, errors propagate—initial mis-reversals alter outlet paths and upstream/downstream assignments, misleading subsequent decisions and producing additional errors. Even so,

performance remains solid: with 20% of edges reversed, the completeness, correctness, and quality measures exceed 0.9. Some edges are left unreversed by design because the procedure prioritizes resolving sink nodes; if an edge does not induce a sink, it is not processed. Others are untouched when no outlet can be reached within the allowed path-length e , sometimes as an effect of earlier orientations. As e increases, the mean error tends to rise. A gradual schedule for e can mitigate this: under the 20% reversal scenario, increasing e from 2 to 5 reduced the error values (Completeness = 0.94, correctness = 0.99, quality = 0.93). The trade-off is computational cost: the incremental schedule (starting at $e = 2$ and increasing by 1 when no further reversals occur, up to $e = 5$) increased the runtime from 7 min (fixed $e = 5$) to 59 min.

Table 4. Results of applying the flow adjustment on the nodes' reachability under different percentages of edge reversal.

Non-Reachable Nodes	1% (N = 508)	5% (N = 2541)	10% (N = 5082)	20% (N = 10,165)
Reversed network	58%	92%	98%	99%
Repaired network	2.1%	4.2%	6.12%	14%

Table 5. Mean quality measures for the flow adjustment algorithm under different percentages of edge reversal.

Metric	1% (N = 508)	5% (N = 2541)	10% (N = 5082)	20% (N = 10,165)
Completeness	0.98	0.98	0.95	0.92
Correctness	0.99	0.99	0.982	0.98
Quality	0.98	0.97	0.93	0.91

Similarly, to test the edge addition algorithm, different percentages of edges were removed from the reference graph, followed by the application of the algorithm. For each removal percentage, random removal realizations ($n = 4$) are generated; the algorithm is applied to each, and performance is summarized by the same quality metrics described before. The networks corresponding to varying percentages of removal 1%, 5% and 10% are shown in Figure 7. The edge addition experiments are limited to a 10% removal threshold, as removing a higher proportion of edges would lead to excessive fragmentation of the network and unrealistic testing conditions. Beyond this level, the remaining topology no longer provides sufficient structural continuity for the algorithm to infer meaningful connections, and the reconstruction process becomes dominated by random or ambiguous candidate choices rather than topology-driven inference.

The results of applying the algorithm for the different versions of the graph with different percentages of missing pipes are shown in Table 6. To show the effect of adding the road network as a spatial layer that guides the addition process, two methods are evaluated under the same cost function: (i) the edge addition method proposed in this paper and (ii) a Euclidean-only method which relies solely on straight-line distances. Regarding the edge addition method, the overall quality metrics consistently declined with increasing percentage of edge removal. In terms of completeness, the reconstruction task becomes progressively more challenging as the proportion of missing edges increases. The observed decrease from 0.73 to 0.42 as the percentage of missing edges increases from 1% to 10% reflects the compounding difficulty of recovering true network connectivity under higher levels of data loss. As the network becomes more fragmented, many nodes that were previously part of well-connected components become isolated or part of small disconnected components with no path to outlets. This results in a decline in the number of potential candidate connections. The lack of reliable neighbors or reference paths causes

the algorithm to misinterpret connectivity, often skipping valid links or proposing incorrect ones that fail to match the original topology. As a result, the ratio of correctly reconstructed edges (true positives) decreases relative to the total number of missing edges, leading to a sharp decline in completeness.



Figure 7. Section of Montpellier sewer network with different percentages of random edges removed. Blue lines represent the network after removal and the red lines represent the removed edges for each percentage.

A similar behavior is observed in correctness, which decreases from 0.80 to 0.60 between 1% and 10% of missing edges. This pattern indicates that as more edges are removed, the algorithm not only misses a growing share of true connections but also introduces a higher ratio of incorrect additions. However, the fact that correctness values remain consistently higher than completeness values suggests that, despite increased uncertainty, most of the edges added by the algorithm are still plausible within the existing topological and spatial constraints. In other words, the algorithm remains conservative in its selection—favoring fewer but more reliable additions rather than maximizing recovery at the cost of introducing excessive errors.

The combined effect of these two metrics is reflected in quality, which drops from 0.62 to 0.36, capturing both the loss of true positives and the introduction of false connections. As the network becomes increasingly fragmented, the algorithm's balance between recovery and precision deteriorates.

The results also show that the Euclidean-only method tends to add more edges than the edge addition method, where the completeness ratio is higher, reflecting the effect of spatial constraints imposed by the road geometry. When a node lies on a road segment, the search for its candidate connections is restricted to the road to prevent creating unrealistic crossings through buildings. This constraint narrows the search space and often limits the number of valid candidates, leading to fewer added edges overall. Nevertheless, the higher correctness indicates that incorporating road information improves the spatial plausibility of the reconstructed connections.

However, the improvement remains modest relative to the additional data requirements and computational cost introduced by integrating the road network. For instance, at 10 % removal, the Euclidean-only method was completed in approximately 2 min, whereas incorporating the road network increased the computation time to about 8 min. Given this small performance gap and the increase in runtime, the Euclidean-only approach may be preferred when computational efficiency or data availability is a concern.

Table 6. Mean quality measures for 1%, 5% and 10% removal of edges using the developed method and Euclidean method.

Metric	Removal Percentage					
	1%		5%		10%	
	Edge Addition	Euclidean Only	Edge Addition	Euclidean Only	Edge Addition	Euclidean Only
Completeness	0.73	0.74	0.6	0.64	0.42	0.52
Correctness	0.8	0.76	0.72	0.66	0.6	0.55
Quality	0.62	0.59	0.48	0.48	0.36	0.33

When combining both tests, two disjoint edge sets from the reference graph are randomly selected: one to be reversed and another to be removed. Then the flow adjustment algorithm is applied to correct directional errors along with the algorithm described in Appendix A that defines potential sink nodes for reversal, followed by the edge addition algorithm to reconstruct the missing links. As in the individual tests, different percentages of edges are reversed and removed, and the corresponding results of the quality metrics are reported in Tables 7–9 for the random realizations ($n = 4$).

When both types of errors—reversals and removals—were present simultaneously, the performance of each algorithm declined sharply even when evaluated on its own. At 10% reversal and 10% removal, the completeness and quality of the addition method dropped from 0.42 to 0.21 and 0.36 to 0.16, respectively, while the flow-adjustment method fell from 0.95 to 0.70 in completeness and 0.93 to 0.53 in quality. This degradation confirms that the coexistence of direction and topology errors substantially increases reconstruction complexity where both reversed and missing edges disrupt the reachability to outlets required for reliable orientation correction and needed to find candidates for addition. The flow adjustment method consistently outperforms the edge-addition method, reflecting its relative robustness and greater reliability. However, since the edge addition algorithm operates on the output of the flow adjustment step, any false positives or false negatives generated before directly influence its subsequent performance. Incorrectly oriented or unresolved edges alter the network topology and the sink nodes present in the addition step, thereby affecting candidate selection and increasing the likelihood of reconstruction errors.

Although the edge addition step alone exhibits relatively low performance, the combined results at 10 % removal and 10 % reversal remain satisfactory when considering the limited information available. The workflow relies solely on topological relationships, without incorporating additional spatial or hydraulic attributes. Under these conditions, achieving a global completeness of 0.45, correctness of 0.64, and quality of 0.36 demonstrates a reasonable recovery capacity given the degree of network degradation. The results

suggest that even with modest reconstruction accuracy, the integrated procedure can restore a large portion of the original connectivity purely from structural cues.

Table 7. Mean quality measures for 1% reversal of edges and 1% removal of the edges.

Metric	Flow Adjustment Method	Edge Addition Method	Combined
Completeness	0.87	0.7	0.95
Correctness	0.95	0.7	0.96
Quality	0.83	0.53	0.92

Table 8. Mean quality measures for 5% reversal of edges and 5% removal of the edges.

Metric	Flow Adjustment Method	Edge Addition Method	Combined
Completeness	0.82	0.4	0.55
Correctness	0.88	0.56	0.75
Quality	0.74	0.3	0.43

Table 9. Mean quality measures for 10% reversal of edges and 10% removal of the edges.

Metric	Flow Adjustment Method	Edge Addition Method	Combined
Completeness	0.7	0.21	0.45
Correctness	0.8	0.4	0.64
Quality	0.6	0.16	0.36

Nevertheless, the proposed methodology does not aim to generate or optimize a complete sewer layout as in previous work [13,14,16]. Instead, it operates under substantially more limited information, addressing the far more constrained problem of reconstructing pipe-to-pipe connectivity and correcting flow direction in large, partially unreliable networks. The evaluation follows the same principle: it is performed strictly at the pipe level, where an inferred edge is deemed correct only if both its endpoints and its directionality exactly match the reference network. This contrasts with existing approaches, which typically assess performance using coverage-based metrics [13], cost-driven criteria [16], or length-overlap measures [14].

The current implementation builds the network graph from an existing RDF graph, but the methodology is representation-independent and could just as well use a graph constructed directly from GIS data (e.g., shapefiles). In practice, repairing connectivity via ontology reasoning alone is too impractical, so the reconstruction is performed with explicit graph-theoretic algorithms on the extracted network.

4.2. Limitations

While the proposed methodology is effective in addressing the connectivity of the network in the absence of reliable elevation values, it has inherent limitations. For adding the connections, the method assumes pipes generally follow roads, so when a sink node is near a road, candidate connections are limited to other road nodes to avoid unrealistic crossings through buildings and private spaces. This is a practical but imperfect simplification: in reality, some pipes run through parks or off-road areas, so true off-road connections may be missed. Future work could relax this limitation by incorporating building footprints and land features into the search. A similar approach was tested by [35] to design urban drainage systems.

Another underlying assumption is that missing connections concern only pipes, which may not always hold true—missing or misclassified nodes (e.g., outlets or nodes with undefined types) can also occur. Such information can be validated by resorting to other data sources. For instance, in Montpellier, it was verified that only thirteen treatment

plants exist within the city. However, this does not apply to discharge structures. A comparison between the GIS dataset and OpenStreetMap revealed at least two nodes without an assigned type, which, upon verification, were confirmed to correspond to discharge structures.

Overall, while topology alone cannot fully reproduce the physical behavior of the system, it provides a strong structural basis for reconstruction under incomplete information. Addressing these cases would require incorporating additional contextual data, such as land and hydrological features, to better infer or validate their function.

Moreover, it is important to acknowledge that the existing network layout itself does not necessarily represent the most optimal configuration from a hydraulic or geometric standpoint. In many cases, the connections observed today reflect the best feasible choices at the time of construction, shaped by the design standards, site constraints, and urban context prevailing when the pipes were originally laid. As networks evolve, subsequent extensions or partial rehabilitation often preserve these historical patterns, even when they no longer align with idealized design principles. Consequently, deviations between the reconstructed and actual network may not reflect solely algorithmic limitations but also the practical compromises and temporal variability inherent in real infrastructure development. Hence, human intervention might still be necessary to implement solutions in these instances. Nevertheless, the methodology provides a robust and scalable solution for recovering pipe-level connectivity in large real-world sewer networks and addresses a practical gap in current practice.

5. Conclusions

This study presents a methodology for the construction and correction of sewer networks in the absence of reliable attribute data. The proposed framework comprises two main algorithms: a flow adjustment algorithm, which resolves sink nodes and restores correct flow directions, and an edge addition algorithm, which reconnects disconnected nodes to the rest of the network. The underlying principle guiding the approach is that all nodes should have a valid path to an outlet, defined as either a treatment plant or a discharge structure. The methodology relies solely on the topological structure of the network but can be further enriched with hydraulic, geometric, or spatial attributes when available. It was applied to the Montpellier Metropolis sewer network and evaluated by systematically removing and reversing varying percentages of edges to assess performance. Both algorithms behaved much better at the individual level, especially flow adjustment, where even after 20% of edges were reversed, the algorithm still achieved correctness equal to 0.98 (Table 5). As for edge addition, correctness decreased to 0.6 after removing 10% of the edges (Table 6). However, the results were poorer when combining the two algorithms and with increasing data degradation—particularly at 10 % removal and 10 % reversal—where the correctness decreased to 0.8 and 0.4 for flow adjustment and edge addition algorithms respectively (Table 9). Still the reconstructed network demonstrated reasonable connectivity and logical coherence given the complete absence of auxiliary information.

The methodology is fully adaptable and can be applied to any sewer network by specifying the outlets and selecting suitable parameter values for each case. It operates on existing systems, adding missing connections and correcting flow-direction errors, and it can be enriched with additional attributes—such as elevation, slope, or capacity—whenever these are available. The output files are in generic format, i.e., they can be used for any modeling software or code. Ensuring a coherent and topologically correct network is essential, as downstream analyses in hydraulics, flooding and asset management all depend on having a structurally consistent network.

Author Contributions: B.H.: Conceptualization, Methodology, Writing—Original Draft, Writing—Review & Editing. N.C.: Conceptualization, Methodology, Writing—Review & Editing, Validation, Supervision. C.P.: Conceptualization, Methodology, Writing—Review & Editing, Validation, Supervision. All authors have read and agreed to the published version of the manuscript.

Funding: This research has received support from the European Union’s Horizon research and innovation program under the MSCA-SE (Marie Skłodowska-Curie Actions Staff Exchange) grant agreement 101086252; Call: HORIZON-MSCA-2021-SE-01; Project title: STARWARS (STormwAteR and WastewAteR networkS heterogeneous data AI-driven management). This research has also received support from the ANR CROQUIS (Collecte, représentation, complétion, fusion et interrogation de données de réseaux d’eau urbains hétérogènes et incertaines) project, grant ANR-21-CE23-0004 of the French research funding agency—Agence Nationale de la Recherche (ANR).

Data Availability Statement: The data that support the findings of this study were derived from resources available in the public domain: <https://data.montpellier3m.fr/dataset/reseaux-dassainissement-de-montpellier-mediterranee-metropole> (accessed on 1 July 2023).

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Appendix A. Sink Node on Road Intersection

The flow-adjustment algorithm initially assumes that when a WCC contains a unique sink node and no further reversals are possible, the only remaining action is to add an outgoing edge from this sink. In practice, however, this assumption can be incorrect. A sink at the end of a component does not necessarily indicate the true location of the missing connection. The actual break in connectivity may lie elsewhere in the component, and the apparent sink may simply be the result of an incorrect flow direction. To distinguish genuine missing connections from misidentified sinks, we develop a sink node on road intersection algorithm that uses both pipe geometry and the structure of the underlying road network.

This algorithm evaluates the spatial position of a sink node relative to nearby road intersections and examines whether the geometry of the incoming edges is compatible with the addition of a new outgoing connection at that location. Only for sink nodes located near a road junction, the method inspects all road branches meeting at that intersection and compares them with the set of incoming road directions associated with the pipes connected to the node. If a road branch exists for which no corresponding pipe is present, the algorithm interprets this as a plausible missing connection and proposes adding a new edge along that road. If all available road branches are already accounted for by incoming pipes, the sink is unlikely to be the correct location for an added connection. In such cases, the sink may be the result of an incorrectly oriented pipe upstream, and the algorithm instead marks one of the incoming edges as a candidate for reversal. As in Figure A1, where in Figure A1a, an outgoing edge can be added to sink node A unlike Figure A1b, where adding an outgoing edge might cross building as no road exists.

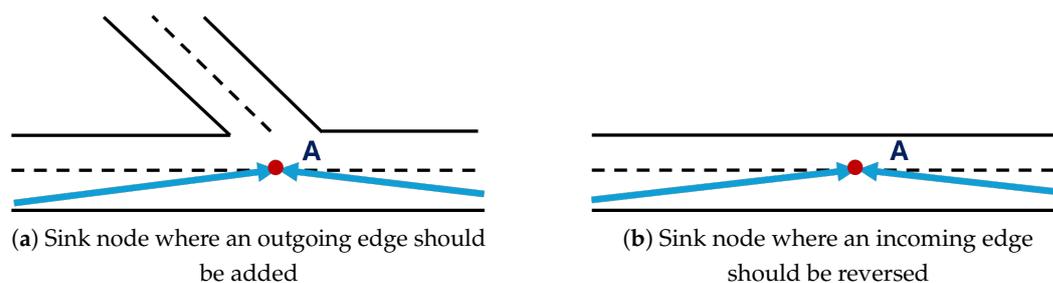


Figure A1. An example of the cases checked by the sink node intersection algorithm where red point A corresponds to the sink node, the blue segments with arrows illustrate the direction of flow along the incoming edges, and the black dotted line represents the road alignment.

The algorithm is not a standalone requirement of the workflow; it is an optional diagnostic that can be invoked with the flow adjustment algorithm. Its purpose is to flag sinks that lie near road intersections where an additional outgoing link would be topologically plausible but absent in the current graph. This check is conservative (it does not force changes), lightweight to run, and useful as a final sanity pass. However, it can be omitted without affecting the core methodology and should not be used as the sole criterion for editing flow directions.

References

1. BS EN 752:2017; Drain and Sewer Systems Outside Buildings—Sewer System Management. Technical Report; BSI: London, UK, 2017.
2. Reyes-Silva, J.D.; Novoa, D.; Helm, B.; Krebs, P. An Evaluation Framework for Urban Pluvial Flooding Based on Open-Access Data. *Water* **2023**, *15*, 46. [CrossRef]
3. Montalvo, C.; Reyes-Silva, J.; Sañudo, E.; Cea, L.; Puertas, J. Urban pluvial flood modelling in the absence of sewer drainage network data: A physics-based approach. *J. Hydrol.* **2024**, *634*, 131043. [CrossRef]
4. Dai, Y.; Chen, L.; Shen, Z. A cellular automata (CA)-based method to improve the SWMM performance with scarce drainage data and its spatial scale effect. *J. Hydrol.* **2020**, *581*, 124402. [CrossRef]
5. He, L.; Nan, J.; Ye, X.; Chen, L.; Ji, S.; Chen, Z.; Xiao, Q. A graph neural network using physical attributes to improve the system-wide nodal water-level prediction in sparsely monitored urban drainage systems. *J. Hydrol.* **2025**, *663*, 134306. [CrossRef]
6. Laakso, T.; Kokkonen, T.; Mellin, I.; Vahala, R. Sewer life span prediction: Comparison of methods and assessment of the sample impact on the results. *Water* **2019**, *11*, 2657. [CrossRef]
7. Shrestha, A.; Mascaro, G.; Garcia, M. Effects of stormwater infrastructure data completeness and model resolution on urban flood modeling. *J. Hydrol.* **2022**, *607*, 127498. [CrossRef]
8. Qi, X.; Khu, S.T.; Yu, P.; Liu, Y.; Cai, T.-y.; Wang, M. Assessing the impact of incomplete stormwater network data on uncertainty in simulation results. *J. Hydrol.* **2025**, *661*, 133788. [CrossRef]
9. Hajibabaei, M.; Hesarkazzazi, S.; Sitzenfrei, R. Filling data gaps in urban drainage networks: An automated graph theory framework for data collection and reconstruction. *Water Res.* **2025**, *287*, 124272. [CrossRef]
10. Belghaddar, Y.; Chahinian, N.; Seriai, A.; Begdouri, A.; Abdou, R.; Delenne, C. Graph Convolutional Networks: Application to Database Completion of Wastewater Networks. *Water* **2021**, *13*, 1681. [CrossRef]
11. Díaz, A.G.; Kapelan, Z.; Langeveld, J.; Taormina, R. Transferable and data efficient metamodeling of storm water system nodal depths using auto-regressive graph neural networks. *Water Res.* **2024**, *266*, 122396. [CrossRef]
12. Harvey, R.R.; McBean, E.A. Predicting the structural condition of individual sanitary sewer pipes with random forests. *Can. J. Civ. Eng.* **2014**, *41*, 294–303. [CrossRef]
13. Chegini, T.; Li, H. An algorithm for deriving the topology of belowground urban stormwater networks. *Hydrol. Earth Syst. Sci.* **2022**, *26*, 4279–4300. [CrossRef]
14. Chahinian, N.; Delenne, C.; Commandré, B.; Derras, M.; Deruelle, L.; Bailly, J. Automatic mapping of urban wastewater networks based on manhole cover locations. *Comput. Environ. Urban Syst.* **2019**, *78*, 101370. [CrossRef]
15. Et-Targuy, O.; Delenne, C.; Benferhat, S.; Ma, T.; Do, T.; Begdouri, A. From GIS to Graphical Representation for Maintaining Connectivity of Wastewater Network Elements. *SN Appl. Sci.* **2024**, *6*, 851. [CrossRef]
16. Duque, N.; Duque, D.; Aguilar, A.; Saldarriaga, J. Sewer Network Layout Selection and Hydraulic Design Using a Mathematical Optimization Framework. *Water* **2020**, *12*, 3337. [CrossRef]

17. Dunton, A.; Gardoni, P. Generating network representations of small-scale infrastructure using generally available data. *Comput.-Aided Civ. Infrastruct. Eng.* **2024**, *39*, 1143–1158. [[CrossRef](#)]
18. Jeffers, S.; Montalto, F. Modeling urban sewers with artificial fractal geometries. *J. Water Manag. Model.* **2018**. [[CrossRef](#)]
19. Montalvo, C.; Tamagnone, P.; Sañudo, E.; Cea, L.; Puertas, J.; Schumann, G. Sewer Network Data Completeness: Implications for Urban Pluvial Flood Modelling. *J. Flood Risk Manag.* **2025**, *18*, e70107. [[CrossRef](#)]
20. Khaleghian, H.; Shan, Y. Developing a Data Quality Evaluation Framework for Sewer Inspection Data. *Water* **2023**, *15*, 2043. [[CrossRef](#)]
21. Caradot, N.; Rouault, P.; Clemens, F.; Cherqui, F. Evaluation of uncertainties in sewer condition assessment. *Struct. Infrastruct. Eng.* **2018**, *14*, 264–273. [[CrossRef](#)]
22. Dirksen, J.; Clemens, F.; Korving, H.; Cherqui, F.; Le Gauffre, P.; Ertl, T.; Plihal, H.; Müller, K.; Snaterse, C. The consistency of visual sewer inspection data. *Struct. Infrastruct. Eng.* **2013**, *9*, 214–228. [[CrossRef](#)]
23. Salman, B.; Salem, O. Risk Assessment of Wastewater Collection Lines Using Failure Models and Criticality Ratings. *J. Pipeline Syst. Eng. Pract.* **2012**, *3*, 68–76. [[CrossRef](#)]
24. Bang-Jensen, J.; Gutin, G. *Digraphs: Theory, Algorithms and Applications*, 2nd ed.; Springer: London, UK, 2009. [[CrossRef](#)]
25. Wieten, R.; Bex, F.; Prakken, H.; Renooij, S. Information graphs and their use for Bayesian network graph construction. *Int. J. Approx. Reason.* **2021**, *136*, 249–280. [[CrossRef](#)]
26. Dijkstra, E.W. A note on two problems in connexion with graphs. *Numer. Math.* **1959**, *1*, 269–271. [[CrossRef](#)]
27. Water Environment Federation; American Society of Civil Engineers. *Design of Wastewater and Stormwater Pumping Stations*, 3rd ed.; Manual of Practice No. FD-4; Water Environment Federation: Alexandria, VA, USA, 2012.
28. Montpellier Méditerranée Métropole. Réseaux d’assainissement de Montpellier Méditerranée Métropole. Available online: <https://data.montpellier3m.fr/dataset/reseaux-dassainissement-de-montpellier-mediterranee-metropole> (accessed on 11 January 2026).
29. Haydar, B.; Chahinian, N.; Pasquier, C. From Standards to an Ontology-Based Data Access system for Sewer Networks. In Proceedings of the 10ème Journées Doctorales en Hydrologie Urbaine, Nantes, France, 2–4 October 2024; p. 5.
30. Hagberg, A.; Swart, P.J.; Schult, D.A. *Exploring Network Structure, Dynamics, and Function Using NetworkX*; Los Alamos National Laboratory (LANL): Los Alamos, NM, USA, 2007; p. 12. [[CrossRef](#)]
31. Montpellier Méditerranée Métropole. Filaire des Voies de Montpellier Méditerranée Métropole. Available online: <https://data.montpellier3m.fr/dataset/filaire-des-voies-de-montpellier-mediterranee-metropole> (accessed on 11 January 2026).
32. Régie des eaux de Montpellier Méditerranée Métropole. *Guide Technique des Réseaux d’Assainissement*; Technical Report; Montpellier Méditerranée Métropole: Montpellier, France, 2023. Available online: https://regiedeseaux.montpellier3m.fr/medias/pdf/Guide_technique_travaux_ouvrages_assainissement.pdf (accessed on 11 January 2026).
33. Reyes-Silva, J.D.; Helm, B.; Krebs, P. Meshness of sewer networks and its implications for flooding occurrence. *Water Sci. Technol.* **2020**, *81*, 40–51. [[CrossRef](#)] [[PubMed](#)]
34. Baboun, J.; Beaudry, I.S.; Castro, L.M.; Gutierrez, F.; Jara, A.; Rubio, B.; Verschae, J. Identifying outbreaks in sewer networks: An adaptive sampling scheme under network’s uncertainty. *Proc. Natl. Acad. Sci. USA* **2024**, *121*, e2316616121. [[CrossRef](#)]
35. Zhong, Q.; Situ, Z.; Zhou, Q.; Xiao, J.; Xu, X.; Feng, W.; Jiang, S.; Su, J. Automatic topology and capacity generation framework for urban drainage systems with deep learning-based land use segmentation and hydrological characterization. *J. Hydrol.* **2024**, *641*, 131766. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.