A reduced complexity model with graph partitioning for rapid hydraulic assessment of sewer networks

Barnaby Dobson[1], Hannah Watson-Hill[1], Samer Muhandes[1,2], Morten Borup[3], Ana Mijic[1]

[1]Department of Civil and Environmental Engineering, Imperial College London, London, UK

[2]Project Centre Limited – Flood and Water Management Team, London, UK

[3]Krüger A/S, Veolia Water Technologies, 2860 Soborg, Denmark

Keywords: reduced complexity modelling, low-fidelity modelling, urban flooding, wastewater modelling, graph partitioning, spatio-temporal resolution

Highlights:

-Graph partitioning can reduce the complexity of sewer networks for faster simulations

-CityWat-SemiDistributed can provide a flexible implementation for hydraulic modelling of these reduced networks without calibration

-The combined approach produces accurate simulations at a fraction of the speed of existing high-fidelity models

Abstract

Existing tools for sewer network modelling are accurate but too slow for a range of modern applications such as optimisation or uncertainty analysis. Reduced complexity sewer network models have been developed as a response to this, however, current applications are slow to set up and still require high-fidelity models to be run for calibration. In this study, we compare and develop graph partitioning techniques to automatically group sections of sewer network into semi-distributed compartments. These compartments can then be simulated without calibration in the integrated modelling framework, CityWat-SemiDistributed (CWSD), which has been developed for application to sewer network modelling as part of this study. We find that combining graph partitioning with CWSD can produce accurate simulations 100-1,000x more quickly than existing high-fidelity modelling. We compare a range of graph partitioning techniques to enable users to specify the level of spatial aggregation of the partitioned network, also enabling them to preserve key locations for simulation. We test the impact of temporal resolution, finding that accurate simulations can be produced with timesteps up to one hour. Our experiments show a log-log relationship between temporal/spatial resolution and simulation time, which would enable a user to pre-specify the speed and accuracy needed for their application. We expect that the speed and flexibility of the approach presented in this work may facilitate a variety of novel applications of sewer network models ranging from continuous simulations for long-term planning to spatial optimisation of network design.

1 Introduction

Hydraulically based models are used to simulate how sewer networks of urban catchments will respond to precipitation events (Salvadore et al., 2015). These models enable planners to design interventions that might resolve current network issues, and to plan for changes in the urban catchment. However, the models also require the network and catchment to be represented at high spatial and temporal resolution, resulting in accurate simulations but lengthy simulation times. As regulations, such as the UK's newly introduced 'Drainage and Wastewater Management Planning', require more computationally expensive applications of sewer network models, such as optimisation, real-time control, or climate uncertainty analysis (Water UK, 2019), it is increasingly clear that alternative approaches are needed to complement traditional detailed sewer network modelling.

Surrogate modelling aims to reduce simulation model run times for computationally expensive applications (Razavi et al., 2012). The principles of surrogate modelling are to apply two broad families of approaches that improve speed, but still enable sufficiently accurate approximations of the model output of interest (e.g., performance metrics). The first is a response surface surrogate, which uses statistical/machine learning techniques to create a data-driven relationship between inputs and outputs, for example by complementing evaluations of a water distribution network model with a trained neural network to aid optimization (Andrade et al., 2016). Second is low-fidelity modelling, which creates a physically based representation at a reduced complexity, for example by aggregating in space as a lumped hydrological model would do (Beven, 2012). In this paper, we refer to low-fidelity modelling as 'reduced complexity' modelling to avoid the presumption that fidelity may either be just 'high' or 'low' (and nowhere in between) and to remove the implication that low-fidelity is equivalent to low-trustworthiness.

Following the modern planning demands placed on sewer network analysis, surrogate modelling concepts have recently been applied to sewer network modelling (Thrysøe et al., 2019; Li & Willems, 2020). These existing studies implement a hybrid approach that uses a data-driven analysis of high-fidelity model simulations to define the parameters of a reduced complexity model. We argue that the requirement for calibration by high-fidelity models limits the applicability of a surrogate approach to sewer network modelling. This is primarily because the intensive expertise, data and time requirements for high-fidelity sewer network model creation and simulation is counter to the exploratory purposes that surrogate models are suited for. Thus, we propose that a more useful and widely applicable approach for surrogate modelling of sewer networks would be to use an entirely physically based reduced complexity model.

Any physically based or hybrid approach for surrogate modelling of sewer networks will require a means to reduce complexity, as mentioned, typically achieved by spatial aggregation. Thrysøe et al. (2019) achieve this aggregation by manual delineation of sub-areas, referred to as 'compartments' of a sewer network. Although this allows a user to have some degree of control over the complexity of the surrogate model ultimately created, it is a labour intensive task and will produce different results depending on the person who is performing it. Instead, Li & Willems (2020) extract sub-graphs to identify each self-contained area of a network and lump at this scale. Although this approach is

automated and objective, it produces a model of a single, coarse, resolution. While not applied for surrogate modelling, Liu et al. (2018) demonstrate that graph partitioning algorithms can be used to automatically delineate water distribution networks into reduced graphs with a pre-specified complexity. We expect that these graph partitioning techniques might reasonably be used to aggregate sewer networks for modelling, rather than just for administrative purposes (as was done in Liu et al. (2018)).

Ochoa-Rodriguez et al. (2015) have shown that temporal resolution for precipitation can be equally important in simulating sewer networks as spatial resolution. Given these findings, we expect that reducing complexity via temporal resolution may also be an effective means to perform surrogate modelling of sewer networks. High-fidelity sewer network models are typically not well suited to low temporal resolution due to numerical instabilities that are introduced (Falter et al., 2013). However, surrogate modelling presents an opportunity to formulate simplified, but more accommodating, numerical schemes. Ultimately this will facilitate larger timesteps than would be possible with high-fidelity models, enabling investigating the interactions between spatial and temporal resolution and giving a reduced complexity model user more control of their model complexity to better suit their needs.

Thus, in this study we propose that, by combining graph partitioning with physically based reduced complexity modelling, we can create simplified sewer network models of a pre-specified speed and accuracy. Since none exist, we first present a novel physically based surrogate model for processing flows in an aggregated sewer system that only requires asset information without the need for calibration using high-fidelity modelling. We then compare a range of graph partitioning algorithms, including two novel approaches tailored to sewer networks, to enable spatial aggregation for surrogate modelling. Finally, we present experiments that investigate how spatial and temporal resolution of surrogate models interacts with both simulation speed and accuracy, information that is vital to know in advance of surrogate model applications such as optimisation and uncertainty analysis. Ultimately, we envisage such an approach to be of great value to those that require rapid evaluation of sewer network models, while also providing further insight into the recurring questions of spatial/temporal resolution in the context of environmental modelling.

2 Methods

To facilitate simulations with a speed and complexity that suits the needs of a given user, a modelling approach that can accommodate spatial and temporal aggregation is needed. The high-fidelity model against which simulations are compared, InfoWorks, is presented in Section 2.1.1. The novel, reduced complexity model that can make these accommodations, CityWat-SemiDistributed (CWSD), is presented in Section 2.1.2 and Appendix A. The graph partitioning algorithms that enable a user to specify the degree of spatial resolution in CWSD are presented in Section 2.2. We detail the evaluation of CWSD in Section 2.3. The case study catchment, Cranbrook, UK, is described in Section 3.1. The specific experiments that bring together CWSD and partitioning are detailed in Section 3.2. The workflow used in this study is summarised in Figure 1.

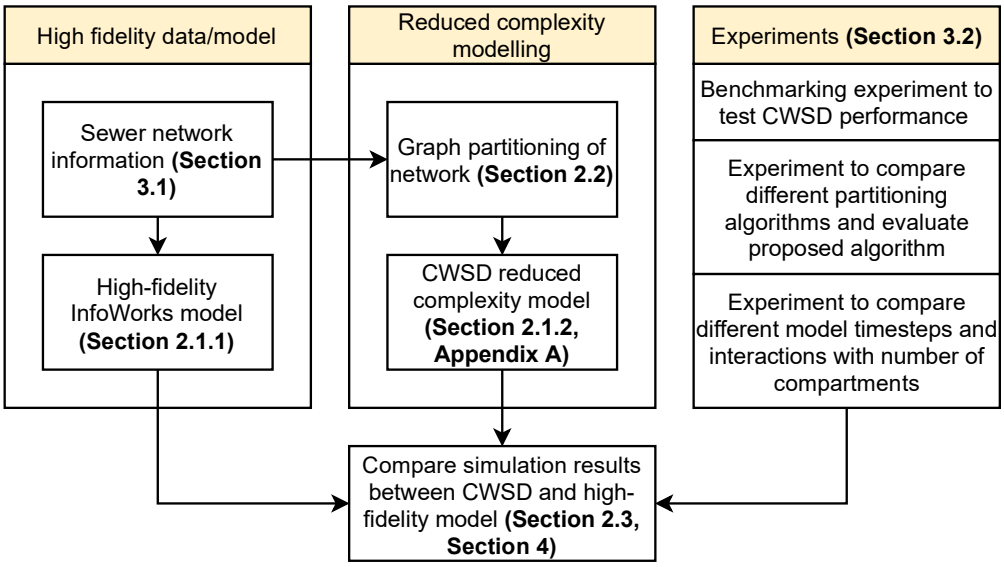


Figure 1: Illustration of the workflow presented in this study.

2.1 Sewer modelling at different levels of complexity

2.1.1 High-fidelity modelling

A range of high-fidelity models exist for sewer network simulations, for example, InfoWorks ICM (Innovyze, 2014), SWMM (Rossman, 2010), or MIKE URBAN (DHI, 2014). In this study we use InfoWorks ICM since it is widely used in the UK and in research applications (Ochoa-Rodriguez et al., 2015; Gong et al., 2018; Babovic & Mijic, 2019; Muhandes et al., 2021). In InfoWorks ICM, a conduit is represented as a defined length link in the network between two nodes. The boundary condition between the link and a node is either of the outfall or head loss type. The gradient of a conduit is defined by invert levels at each end of the link. The conveyance function is based on the Manning equation, that uses the hydraulic gradient resulted by the upstream and downstream head-difference when the network is surcharged or the conduit slope otherwise. InfoWorks ICM has its own 1D St Venant solvers to undertake hydraulic calculations of a drainage network for transient flow (Innovyze, 2014). For hydrology calculations, InfoWorks ICM uses SWMM5 hydrology and infiltration modules (Rossman, 2010).

2.1.2 CityWat-SemiDistributed (CWSD) reduced complexity modelling

CWSD is an open source integrated urban water modelling framework and software designed for easy setup, efficient computation times achieved by spatial aggregation (hence semi-distributed) and flexibility to accommodate a wide range of system types (Dobson, Jovanovic, et al., 2021). The Python source code used in this study is provided at (Dobson, Watson-Hill, et al., 2021). The modelling framework provides a node- and arc-based system representation, where nodes are various elements of the water system (e.g., treatment plants or aggregations of pipe network) and arcs are the means that flow information is passed between nodes (e.g., specific pipes or rivers). The semi-distributed nature of CWSD and its ability to flexibly change how specific subsystems are represented makes it a uniquely suitable tool to explore reduced complexity modelling in a sewer network context. In Figure 2 we demonstrate how a sewer network can be split into ‘compartments’ to

provide a reduced complexity semi-distributed representation. As in Thrysøe et al. (2019), the volume of water storage available in specific manholes and pipes can be aggregated to provide a compartment capacity. Thus, each CWSD node in this study is a compartment of semi-distributed sewer network, while arcs are the pipes between compartments as in the high-fidelity model.

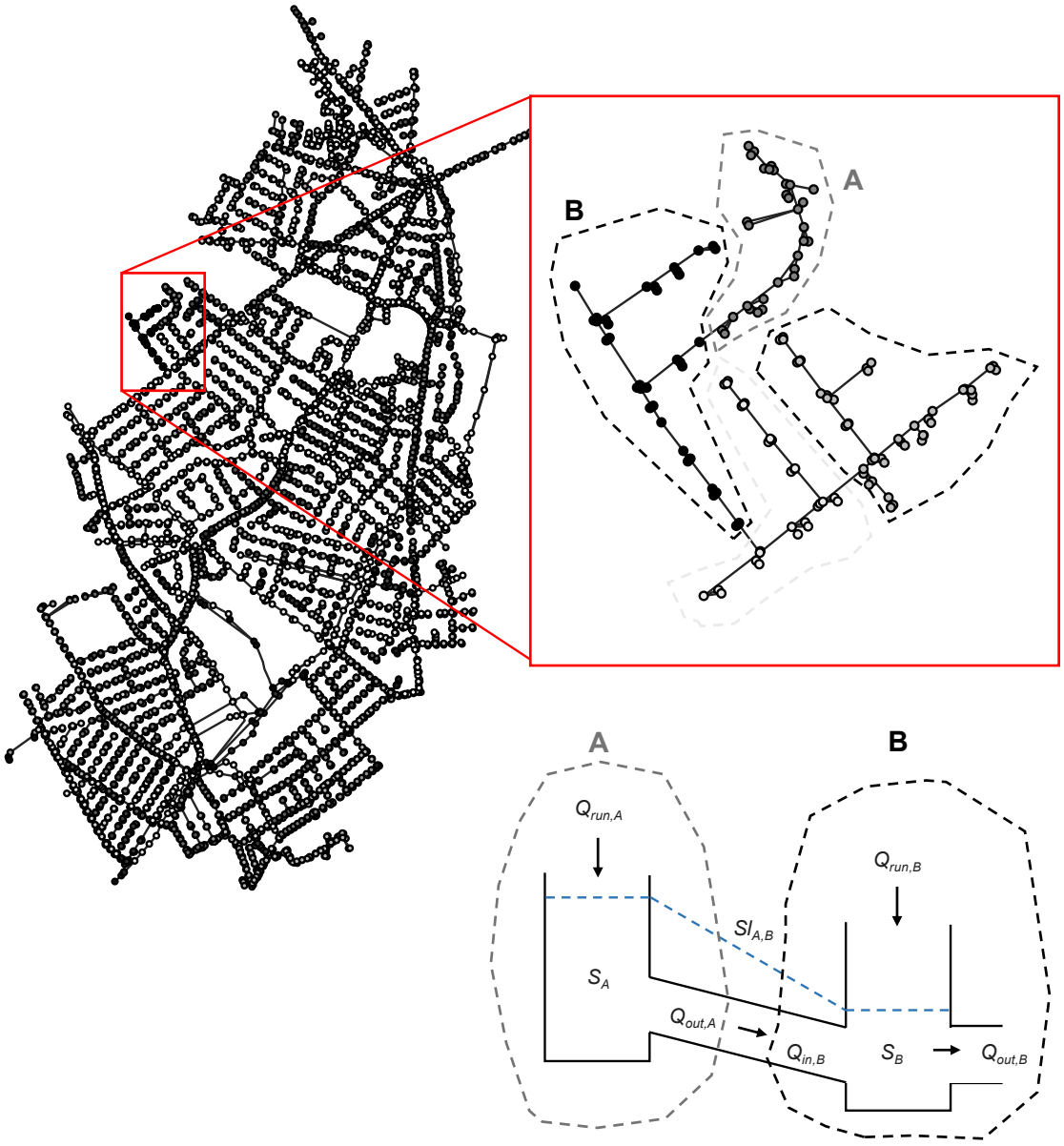


Figure 2: (Map) An example depicting how the Cranbrook sewer network (Babovic & Mijic, 2019), for our case study can be compartmentalised into a semi-distributed, reduced complexity, representation. (Schematic) Q indicates a flow (m3/s), S a storage (m3) and Sl a hydraulic gradient (m/m) of the two compartments, A and B.

In this study, flow information is passed entirely in the form of a push request, that is, a request to push water from an upstream to a downstream compartment. This form of information passing is a common integrated modelling technique in the context of urban wastewater modelling, for example as

in CityDrain3 (Burger et al., 2016). Thus, the simulation takes place by orchestrating a series of push requests between compartments, as demonstrated in Appendix A, Figure A1.

Previously, CWSD has been demonstrated only for extremely low-resolution urban water management modelling (i.e., London's sewer network was aggregated in semi-distributed areas of around 200 km^2). Thus, in this study, we have developed a range of functionalities to enable CWSD for the finer resolutions required for sewer network simulation, namely, pipe travel times, variable timestep size, simple hydraulics, and time-area runoff calculations. This more physically based representation, and the other aggregated compartment-scale parameters needed to implement them, are detailed in Appendix A.1. We summarise the key hydraulic assumptions below:

- Flow between compartments is governed by Manning's equation. Where elevation head is based on the elevation of sending/receiving manholes in a compartment and pressure head is determined by the current storage and total chamber area of manholes in a compartment. The area and hydraulic radius terms in Manning's equation are assumed constant at full-bore. This will result in a slight underestimation of travel times during less intense events, but we assess that the negative impact of this is negligible.
- Travel time within a compartment and between compartments is tracked but assumed to travel at the full-bore velocity of pipes. This will result in slight underestimation of travel times during less intense events.
- If push requests from previous timesteps arrive at a flooded compartment, they are rejected and immediately returned to the upstream compartment. This provides a representation of the backwater process but does not account for hydraulic head generated by flooded water.
- Surface runoff is represented using the time-area method as described in (Butler & Davies, 2004).

We also provide a worked spreadsheet example performing the compartment aggregation process and hydraulic calculations in Supplemental Material S1 an S2.

2.2 Sewer network aggregation

2.2.1 Application of graph partitioning

As anticipated in the introduction, to perform reduced complexity modelling that can be scaled to large sewer networks and improve the usability of these approaches, an automated method to delineate compartments, spatially aggregating them, is necessary. A key innovation in this study is the application of mathematical graph partitioning techniques to perform this task. The goal of graph partitioning is to allocate sets of nodes into sub-groups, resulting in a graph with a smaller number of edges but retaining the overall connectivity of the original network (Buluç et al., 2016). Because there is a wide range of different available techniques to perform graph partitioning, we compare a selection from different method families: spatial, spectral, and heuristic methods. It is also common to combine multiple algorithms to improve the efficiency of partitioning, called a 'multi-level' algorithm (Karypis & Kumar, 1997). We test a selection of these multi-level approaches. We propose two novel multi-level approaches that are tailored for use in sewer network modelling. Our two proposed methods aim to

harness the power of one of the most popular algorithms, Louvain (Blondel et al., 2008), but enable a user to specify the number of partitions that it produces to facilitate more widespread application for sewer network aggregation.

The partitioning methods used in this study, their family, the source for the algorithm implemented, and a summary are shown in Table 1. We include a shorthand in the 'Key' column that results are labelled with.

Method	Key	Family	Source (Python package/Reference)	Summary description
K-Means clustering	km	Spatial	scikit-learn (Pedregosa et al., 2011)	Allocate nodes to groups, iterate to minimise average distance between nodes and group centre
Agglomerative clustering	ac	Spatial	scikit-learn (Pedregosa et al., 2011)	Iteratively merge the two closest nodes until desired number of groups is met
Spectral clustering	sc	Spectral	scikit-learn (Shi & Malik, 2000; Pedregosa et al., 2011)	Allocate nodes to groups, iterate to minimise the eigenvalues of the graph Laplacian of groups
Fluid communities	fluid	Heuristic	networkx (Hagberg et al., 2008; Parés et al., 2018)	Iteratively propagate groups as fluids through the network, a node's membership is based on the density of fluids in that node
Louvain	louv	Heuristic	scikit-network (Blondel et al., 2008; Bonald et al., 2020)	Iteratively merge two nodes that will improve the modularity of the network until no more merging can improve the modularity
K-means with spectral bisection	km_sc	Multi-level	Based on (Yan et al., 2009)	Pre-aggregate the network using K-means, then bisect the largest groups using spectral until desired number of groups
Agglomerative with spectral bisection	ac_sc	Multi-level	Based on (Yan et al., 2009)	As previous, but using agglomerative in the first step
Louvain with spectral reduction	L_scn	Multi-level	Novel	Iterate steps in Louvain partitioning until there are more groups than desired, then use spectral on the already partitioned network to produce the desired number
Optimised Louvain parameters with spectral bisection	L_L_sc	Multi-level	Novel	Iterate Louvain partitioning with a range of tolerance parameters until there are more groups than desired, then bisect the largest groups using spectral until desired number of groups

Table 1: Summary of the partitioning methods used in this study.

2.2.2 Preservation of key sewer elements

A common requirement that we anticipate for sewer network aggregation techniques is the ability to preserve user defined key locations within the partitioned graph. In this study we do this to provide

common locations for comparison arcs across partitioning methods (Section 2.3) and to ensure hydraulic structures are represented (Section 3.2). The methods described in Table 1 do not make provisions for this. A simple technique to overcome this is presented below that is compatible with all methods in Table 1.

1. Remove the sewer elements that are to be preserved from the network graph. If an arc is to be preserved, then remove both the up- and downstream nodes.
2. Sub-graphs are created from the new network graph, resulting in multiple graphs that are separated by the removed nodes.
3. Partitioning algorithms are run separately on each sub-graph. If the number of desired groups is provided (as is the case for all methods in Table 1 except for Louvain), then these are distributed in proportion to the number of nodes in each sub-graph. If sub-graphs contain a small number of nodes, they will still be assigned a minimum of one group, which may cause the resulting number of groups to be greater than the desired number.

2.3 Model evaluation

Since no observation data exists for our case study network, we compare CWSD model simulations against the high-fidelity InfoWorks results, as is common practice in sewer network modelling (Thrysøe et al., 2019). The two types of timeseries data we compare are pipe flows and total compartment storage. The arcs between compartments are conceptually identical to specific pipes in the InfoWorks model and so a direct comparison can be made. Because specific sewer elements (pipes/nodes) are simulated in the InfoWorks model, rather than compartments, we aggregate InfoWorks in-node, in-pipe and flooded volume simulation data to compartment scale to create a timeseries of total compartment storage (compartment storage hereafter), which can then be compared against CWSD simulations. We examine compartment storage rather than only compartment flooded volume because, conceptually, flooding in a compartment must exceed capacity at every compartment sewer element to flood rather than an individual one as is the case in reality. We later provide supporting results (Supplemental Material S3) to show that maximum compartment storage is closely related to maximum flooded volume, thus justifying that it is still information useful to decision makers who want to understand flooding impacts. We use the Nash-Sutcliffe Efficiency (NSE) performance metric to evaluate both flow and storage simulations.

Comparing different partitioning algorithms with different numbers of desired groups is not trivial to do because there are no common sewer elements for comparison and not all sewer elements are equally important to simulate. We provide a comparison from a user-perspective; selecting arcs that must be preserved during partitioning (see Section 2.2.2) or compartments that contain specific nodes and comparing only these locations. To account for the importance of different sewer elements, we randomly select a sewer element from each twentieth percentile of flow/storage to preserve. This will ensure that we can determine how different algorithms and numbers of desired groups reproduce behaviour in both low and high flow/storage sewer elements. We reproduce all experiments twenty times, randomly selecting different sewer elements to preserve for comparison and to account for variability in how difficult specific different sewer elements are to model. In the main text we provide

results for the experiments that compare compartment storage or flow in arcs in the largest twentieth percentile, adjudging these to be of most interest to decision makers, and provide the remainder in Supplemental Material S4.

3 Experimental setup

3.1 Case study catchment

To test the proposed CWSD simulation model and network aggregation techniques, we use the Cranbrook stormwater drainage network in London, UK. The network depicted in Figure 2 drains 8.5 km^2 with over 100 km of pipes, before discharging into the Roding River at two outfall locations. It is ideal for this experiment because it is large enough that manually delineating compartments would be undesirable from a user perspective, but small enough that InfoWorks simulations can be run in high resolution for month-long precipitation timeseries in a feasible amount of time (one or two days) for the purposes of model comparison. The network also contains a variety of complex hydraulic features (lakes, weirs, pumps, orifices, headwalls, and culverted watercourses that drain into the stormwater network) that will test the ability of CWSD to represent more varied systems. We run the experiments (Section 3.2) for two particularly wet month-long storm timeseries, January 2017 and August 2018. These storms were produced from a radar composite, described in (Dobson, Jovanovic, et al., 2021). The Cranbrook stormwater catchment, and its InfoWorks modelled representation, has been presented and validated in a range of studies (Ochoa-Rodriguez et al., 2015; Babovic & Mijic, 2019; Muhandes et al., 2021).

3.2 Experiments

Because the application of the CWSD simulation modelling framework is novel in a sewer network modelling context we first perform a benchmarking experiment to ensure that it can suitably be applied to test the performance of different partitioning algorithms. To create a benchmark semi-distributed representation of the network we use the Louvain algorithm, Table 1. We select this algorithm over a manual delineation, as was used in Thrysøe et al. (2019), to ensure the representation is objective. We select Louvain over other available algorithms because it is one of the simplest to apply, its ubiquity makes it commonplace in benchmarking experiments, and, unlike the other methods, it produces a fixed number of groups so cannot be easily manipulated by tuning parameters. We used a one-minute timestep for CWSD in the benchmarking experiment, in line with the InfoWorks timestep.

Second, we compare the various partitioning algorithms. Since all algorithms in Table 1, except for Louvain, require the user to specify the number of groups produced we test a variety of different specified number of compartments ranging from one to five hundred. As anticipated in Section 2.3, we randomly preserve sewer elements to ensure like-for-like comparison of the algorithms. To ensure the results are not biased by the random choice of sewer elements to preserve, we repeat the experiment twenty times. We used a one-minute timestep for CWSD in the benchmarking experiment, in line with the InfoWorks timestep.

Finally, to understand how complexity in space and time interact, we run experiments that compare how performance changes depending on the desired number of compartments and CWSD timestep size. Our end-goal in this experiment is for a user to be able to specify their desired model accuracy and be able to use the most efficient reduced complexity network aggregation to achieve that accuracy. As well as varying the number of compartments, as in the second experiment, we also vary the timestep size between thirty seconds and one day. We use the partitioning algorithm that is adjudged to perform best from the second experiment.

Because previous surrogate sewer network experiments (e.g., Thrysøe et al. (2019)) have highlighted the importance of ensuring hydraulic structures are not grouped into compartments, we preserve lakes, weirs and pumps in the partitioning, as described in Section 2.2.2.

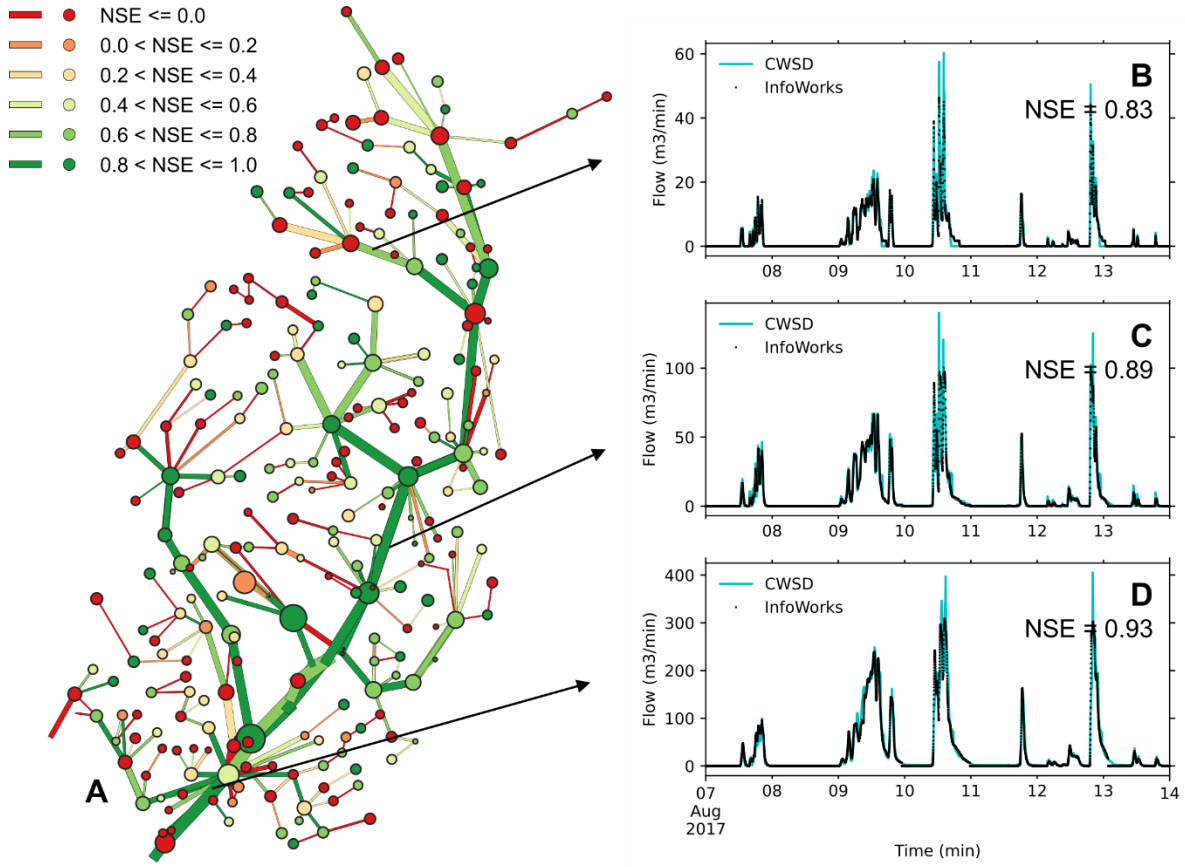
To determine whether the modelling approach can be used to simulate individual sewer nodes (i.e., specific manholes) we also include a supplemental selection of results (S4) where these have been preserved in the same manner as hydraulic structures.

Finally, to avoid timestep to timestep variations in the flow without any hydrological significance obscuring the result comparisons, a five-minute rolling average window has been applied along flows in arcs before plotting or calculating performance measures.

4 Results

4.1 Application of graph partitioning and CWSD to sewer network modelling

As described in Section 3.2, the first experiment is to verify that the proposed CWSD simulation framework can produce simulation results comparable to the high-fidelity InfoWorks simulations using the Louvain partitioning algorithm for benchmarking purposes. In Figure 3A, we depict all compartments and arcs coloured by their Nash-Sutcliffe Efficiency (NSE) value and sized in proportion to their mean flow or compartment storage. It highlights that larger pipes are accurately simulated, while smaller pipes are more variable. In Figure 3B-D, we also provide a close-up of key pipe flows through the network on a single day in the month-long simulation.



298

299 Figure 3: (A) Aggregated network for the benchmark Louvain partitioning algorithm where
300 compartments/arcs are coloured by their NSE value and sized proportionately to their storage
301 or flow (B-D) three sub-sections of flow timeseries along a selection of increasingly large arcs.

302 In Figure 4A, we provide the same map as in Figure 3A, but with simulated compartment storage
303 shown in Figures 4B-D. We see generally that compartment storage performance is lower than arc
304 performance, although still high in many large storage compartments.

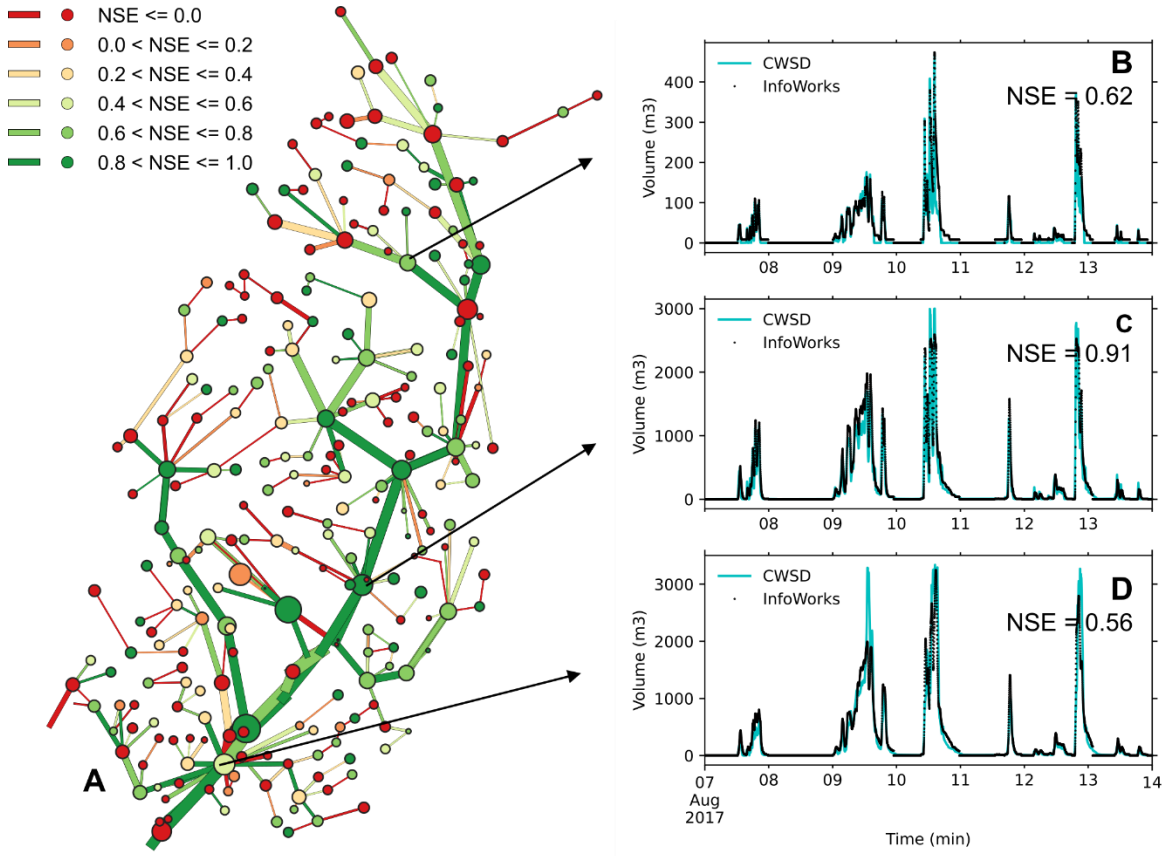


Figure 4: (A) Aggregated network for the benchmark Louvain partitioning algorithm where compartments/arcs are coloured by their NSE value and sized proportionately to their storage or flow (B-D) three timeseries of compartment storage for a selection of compartments.

Figures 3A and 4A highlight that performance appears to be influenced by compartment or pipe size. Thus, in Figure 5 we compare maximum pipe flow and maximum compartment storage against NSE performance. We see that, for pipes of low flows (<1 m3/min, shaded red) and compartments of low storages (<100 m3, shaded red), reasonable performance (NSE>0.5) is rare. For average pipes (between 1 and 10 m3/min, shaded blue) and compartments (between 100 and 1000 m3, shaded blue), reasonable performance is common but poor performance (NSE<0) is also present. For large pipes (>15 m3/min, shaded green) and compartments (>1000 m3, shaded green) reasonable performance is common and poor performance is rare.

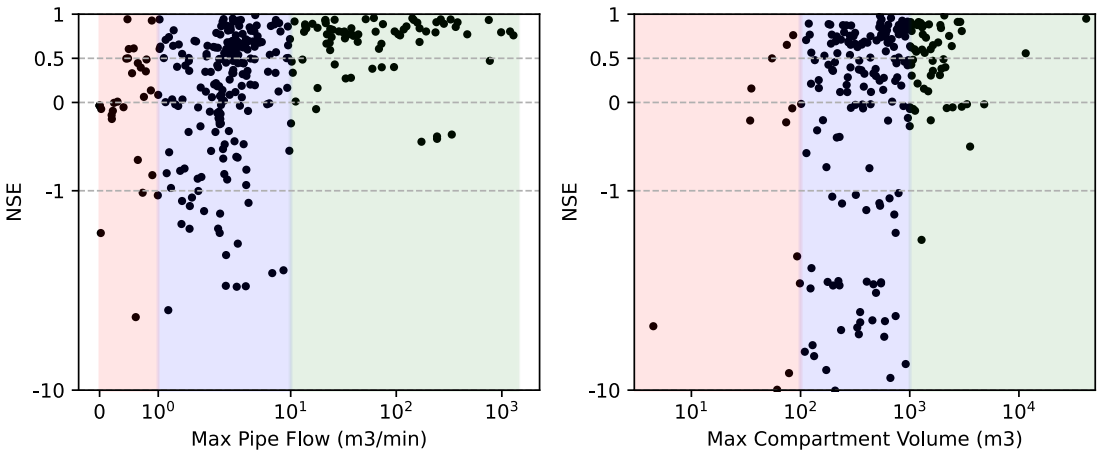


Figure 5: (Left) NSE of flow in pipes for the benchmark Louvain partitioning algorithm simulations (y-axis) against the maximum flow in that pipe for a given simulation (x-axis). (Right) same as (Left) but for NSE of compartment storage. To improve readability, the scale of both axes is linear in the range [-1,1] and log elsewhere.

As anticipated in Section 2.3, compartment storage represents the total volume of water in nodes, pipes and flooded in a compartment, which is not conceptually identical to solely flooded volume that decision makers are most likely to be interested in. Thus, in Supplemental Material S3, Figure S2 we compare maximum compartment storage as simulated by CWSD against maximum flooded volume as simulated by InfoWorks. We see close agreement between the two, justifying the continued investigation of compartment storage in this study.

4.2 Graph partitioning for reduced complexity models

In this section, we compare the ability of different partitioning techniques in delineating compartments to provide accurate simulations. As anticipated in Section 2.3, we randomly compare compartments that contain a specific node and flows in preserved arcs in the network to ensure common points of comparison can be made across methods. In Figure 6, we show the NSE values in preserved arcs from the top twentieth percentile pipes (i.e., the pipes with greatest flow). Each panel demonstrates a different partitioning algorithm. Each point is the NSE performance (y-axis) of a preserved arc for a given number of specified compartments (x-axis). The x-value of each point is randomly moved within +/-5% of its actual value to better distinguish the density of points, i.e., random jitter. Of the compared partitioning methods, Louvain (louv) and the two novel methods, Louvain with spectral reduction (L_scn) and optimised Louvain with spectral bisection (L_L_sc), produce reasonable (>0.5NSE) median performance metrics (dashed blue lines) when the total number of compartments is greater than 100. All partitioning methods produce inferior (<0.5NSE) median performance when the total number of compartments is less than 100. We see that some preserved arcs have high NSE values (e.g., the arc highlighted in cyan points) across all simulations, and some have low NSE (e.g., the arc highlighted in red points). The network surrounding the arc highlighted in both cyan and red is shown in Supplemental Material S5, Figure S21. It shows that the cyan arc is a shallow gradient pipe at the

end of a linear stretch of network, while the red arc is a steep gradient pipe among a complex tangle of network.

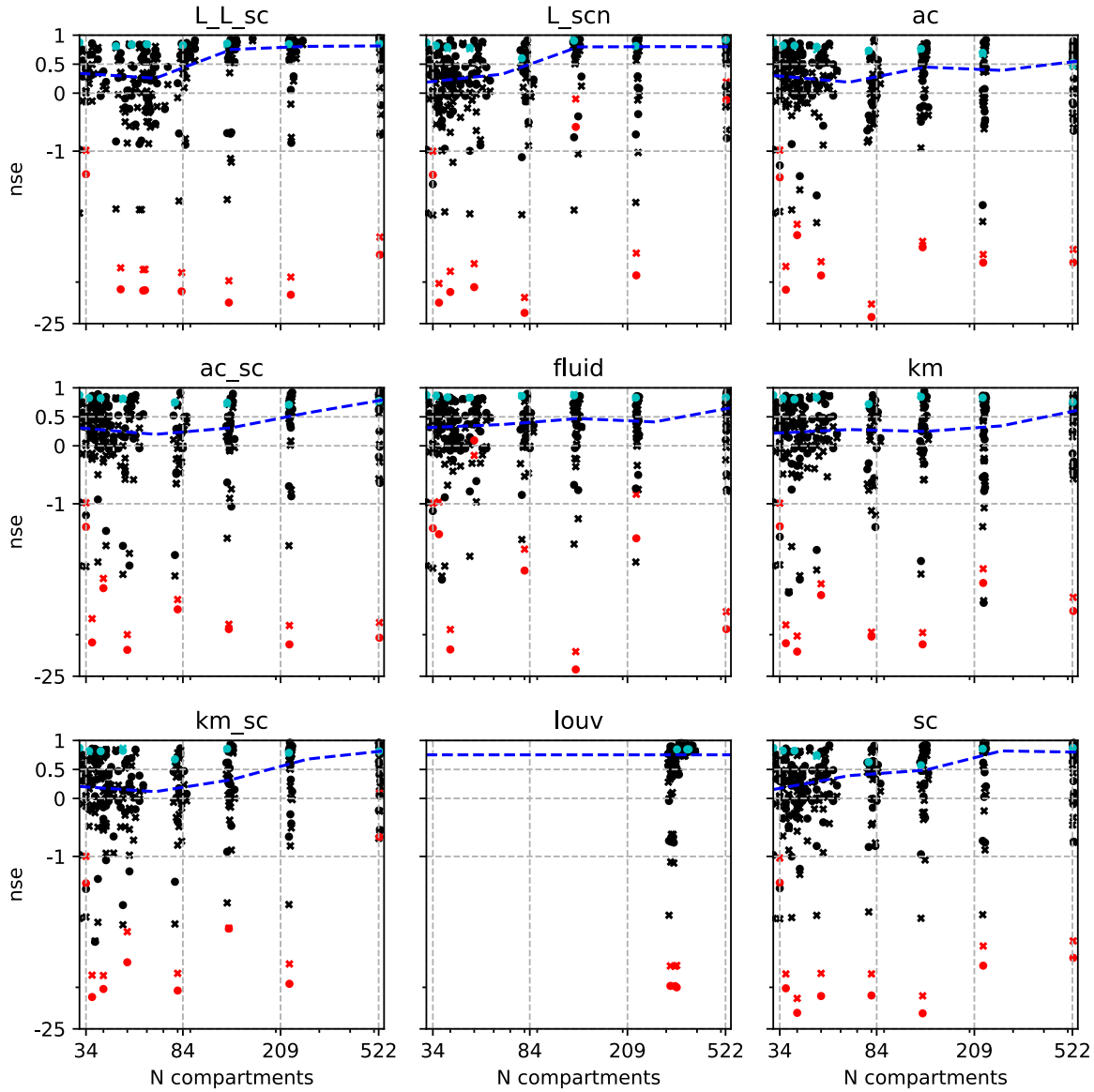


Figure 6: Simulation performance along preserved arcs (points). Interactions between different clustering methods (panels) and different number of compartments (x-axis) is shown. Two month-long precipitation timeseries have been used (cross and circles). The median performance of preserved arcs is plotted (blue dashed lines). The worst performing (red points) and best performing (cyan points) preserved arcs are highlighted. Random jitter in the x-axis has been added to better distinguish between points. The y-axis has been limited to [--25, 1] to improve readability, the scale is linear in the range [-1,1] and log elsewhere.

In Figure 7, we show NSE values of compartment storage for compartments that contain a randomly selected sewer network node from the top twentieth percentile nodes (i.e., nodes with the greatest storage). Of the compared partitioning methods, Louvain (louv) and the two novel methods, Louvain

with spectral reduction (L_scn) and optimised Louvain with spectral bisection (L_L_sc), produce reasonable (>0.5NSE) median performance metrics (dashed blue lines) when the total number of compartments is less than 200. All partitioning methods produce inferior (<0.5NSE) results when the total number of compartments is greater than 400. Although NSE performance for compartment storage is, on average, lower than for arc flows, it is below 0 less frequently than for arcs. We see in general some compartment storage timeseries are easier/harder to predict across all methods (red points are generally lower while cyan points are generally higher), however this is not as ubiquitous as with arcs (i.e., we also see some high NSE values for red points and some low NSE values for cyan points). In Supplemental Material 5, Figure S22, we see that the preserved red node is near the culverted watercourses in the catchment, while the area around the preserved cyan node is paved.

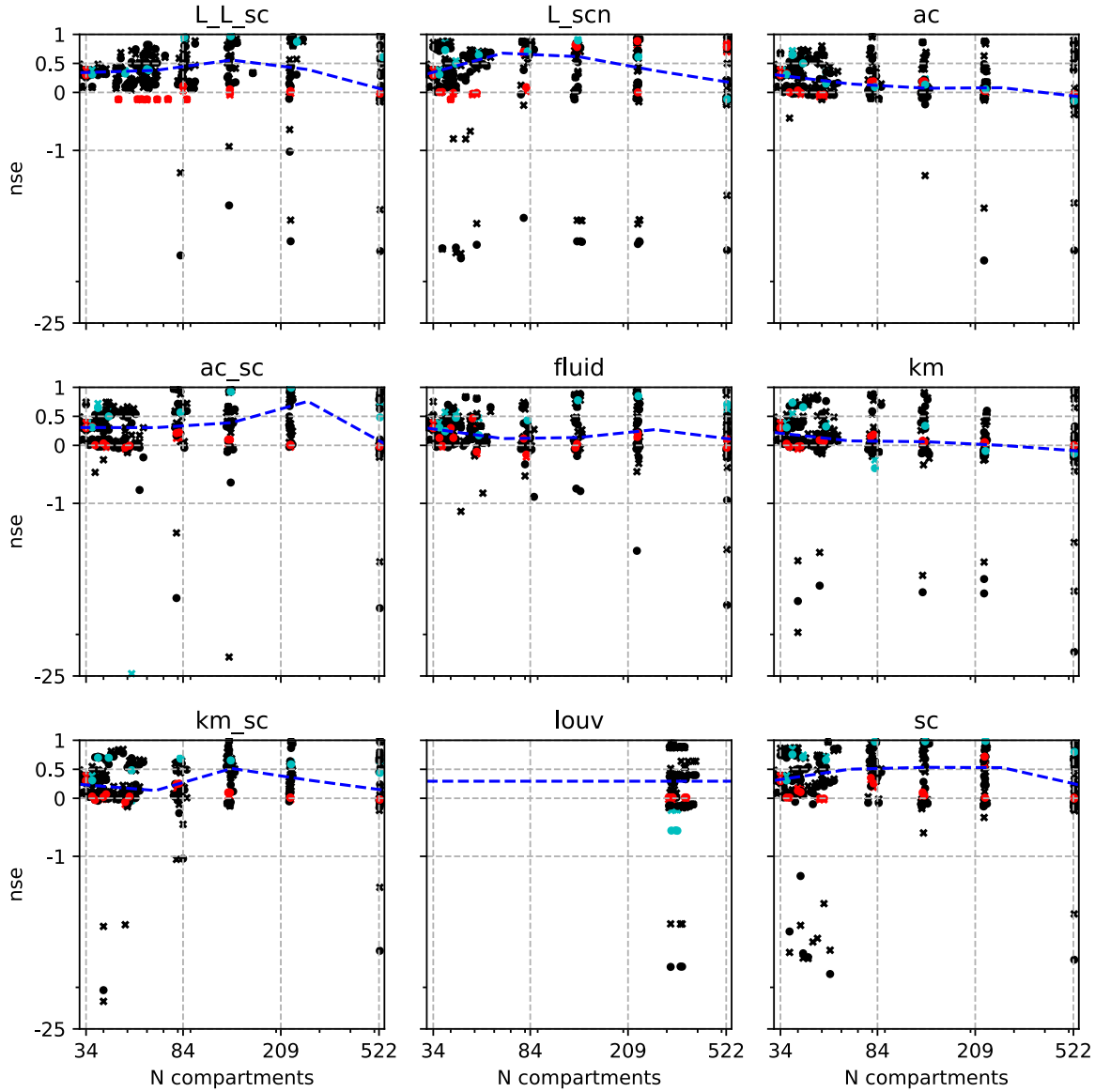


Figure 7: Same as Figure 6, but for compartment storage rather than flow in arcs.

4.3 Interactions between number of compartments and timestep size

Following the results that the proposed 'optimised Louvain with spectral bisection' partitioning algorithm provided the best results over a range of different number of compartments, we use it to investigate interactions with timestep size. These interactions are displayed in Figure 8 for NSE values in the random sample of preserved arcs from the top twentieth percentile pipes (i.e., the pipes with greatest flow). We see clearly that performance degrades as both the total number of compartments decreases and as the timestep increases. It is noteworthy that reasonable performance (>0.5 NSE) can be attained with a timestep as large as one hour. There is a marked difference in performance between total number of compartments less than and greater than 100. In the supplementary material S4 we demonstrate that performance degrades as pipe size decreases and there is less evidence of clear interactions between timestep and total number of compartments. In supplemental material S4, Figure S19, we also demonstrate that this figure is not sensitive to whether individual sewer elements are preserved for comparison or not.

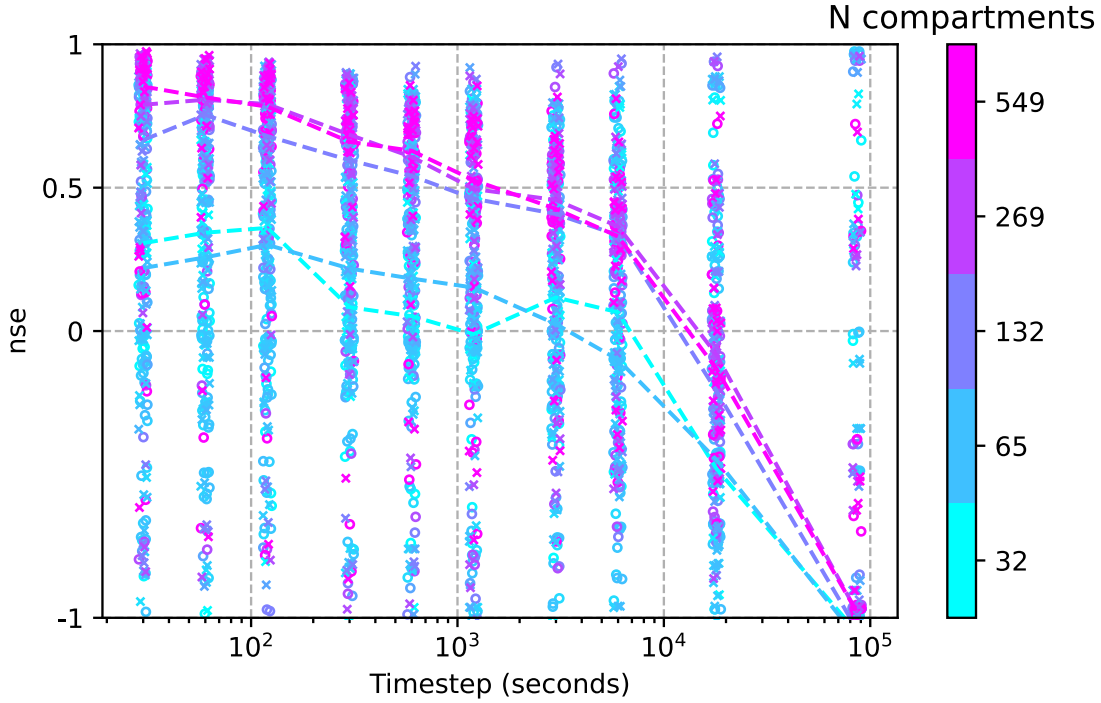


Figure 8: Simulation performance along preserved arcs (points). Interactions between total number of clusters (colour) and model timestep (x-axis) is shown. Two month-long precipitation timeseries have been used (cross and circles). The median performance of preserved arcs is plotted with dashed lines. Random jitter in the x-axis has been added to better distinguish between points. The y-axis has been limited to [-1, 1] to improve readability.

In Figure 9, we display the NSE values for compartment storage in the random sample of compartments selected from the top twentieth percentile nodes (i.e., the nodes with the greatest storage). Performance degrades quite dramatically once the timestep exceeds one hour, following a similar trend to Figure 8, albeit with a steeper drop-off in performance. In contrast to Figure 8, the

interaction with total number of compartments seems to prefer around one hundred compartments, with performance degrading as the number deviates from this. We see performance peak at 132 total number of compartments, degrading as the total number increases or decreases. This appears to be due to some aggregation effect provided by compartmentalising, since it is not seen in the supporting experiment where individual sewer elements are preserved, supplemental material 4, Figure S20. In supplementary material S4 we see that node performance degrades as node size decreases; however, this degradation is to a significantly lesser extent than for the arcs.

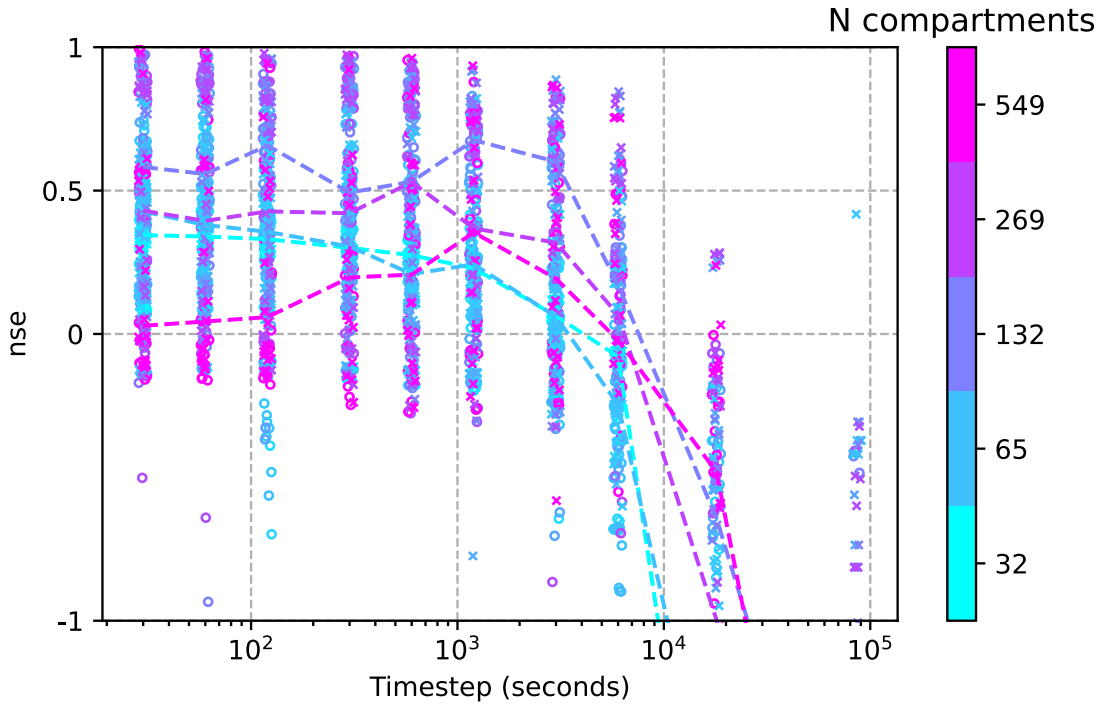


Figure 9: Same as Figure 8, but for compartment storage rather than flow in arcs.

4.4 Computational speed of CWSD

A key goal of this work iss the ability to specify simulation speed, and not just accuracy, as the previous results focus on. Thus, in Figure 10 we show the simulation times from Figures 8 and 9, highlighting how they change with both the total number of compartments and the timestep. We see a deterministic log-log relationship between the three variables. Even the lengthiest simulations, 10^4 seconds, are an order of magnitude quicker than InfoWorks (circled cross in Figure 10), which takes around one day to run for a month-long continuous simulation. The simulation time of the InfoWorks model appears to be in line with deterministic log-log relationship that the CWSD simulation times follow.

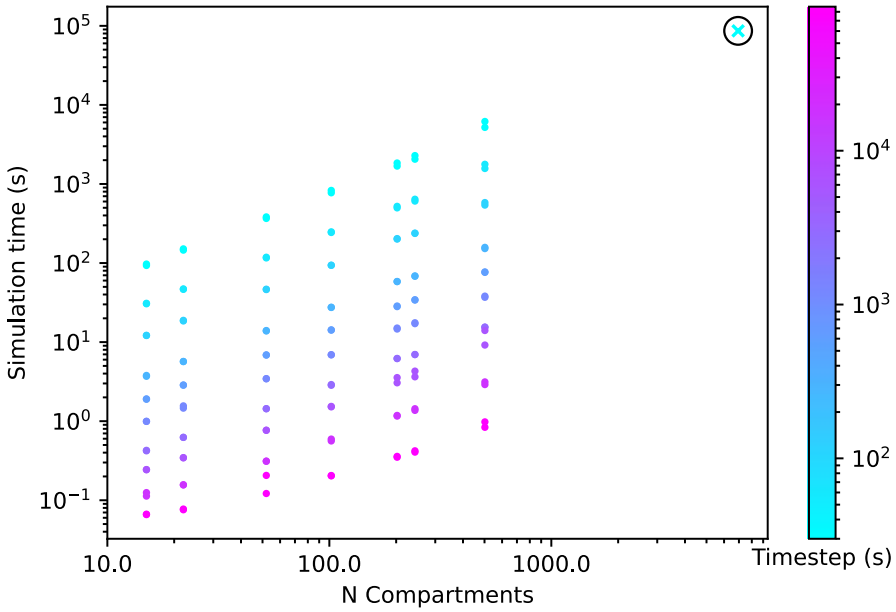


Figure 10: (Filled circles) Interaction between number of compartments, timestep size and total simulation time for the simulations shown in Figures 8 and 9. (Cross circled in black) Total number of compartments and simulation time for the high-fidelity InfoWorks model.

We also comment here on the speed of pre-processing steps:

- By far the lengthiest step was cleaning and checking the raw network data. The high-fidelity model was the only record of this data. Exporting this data revealed obvious errors in the high-fidelity network model (which had to be fixed) and ambiguous notations (which had to be understood). This process required close coordination with an InfoWorks expert.
- Partitioning can take up to one minute, although typically it takes far less. The partitioning speed depends on the total number of clusters specified and the choice of algorithm, with the Louvain algorithm being the quickest (taking 0.01 seconds).
- Aggregating the network information into a compartmentalised model takes around two minutes. It is not sensitive to number of compartments or timestep size.

5 Discussion

The first aim of this study is to develop a modelling approach to automatically create a reduced complexity simulation model for hydraulic simulations that is suitable for rapid assessment of sewer networks without the need for lengthy calibration by a high-fidelity model. In Figures 3 and 4 we show that graph partitioning can automatically reduce the spatial complexity of a sewer network and that the improvements to CityWat-SemiDistributed made in this study facilitate reasonable simulation results in comparison to a high-fidelity model. We expect that further improvements can be made to the approach. Figures 6 and S21 show that a steep arc in a particularly complex area of network is consistently challenging. Figures 7 and S22 show that a compartment that contains culverted watercourses is also difficult to simulate. These findings imply that improvements to the simplified

physical representation in CityWat-SemiDistributed, presented in Section 2.1.2 and Appendix A, may be needed in areas with more complex hydraulic behaviours, which will be addressed in future work.

The second aim of this study is to test a range of graph partitioning algorithms for suitability in reduced complexity sewer network modelling. Figures 6 and 7 show that results are highly dependent on the algorithm selected and that the novel proposed algorithms in this study perform best. Making broad comments about families of partitioning algorithms is difficult; approaches within the heuristic and multi-level families both contain reasonable and poor algorithms. However, exclusively spatial approaches consistently perform poorly, indicating that a graph partitioning approach must, at a minimum, consider the connectivity of the network. We demonstrate that a variety of reasonably performing algorithms can be used to pre-specify nodes and arcs in a network to preserve as an isolated element. Although this is useful in our study for preserving hydraulic structures and ensuring that common arcs can be compared across methods, we also expect that this would be useful to those who have key monitoring points or whose goal is to optimally locate monitoring points to maximise ‘value-for-information’. Combined with the dramatic speed gains shown in Figure 10, we believe the approach presented in this study is ideally suited to a monitoring placement optimisation application.

The final aim of this study is to investigate the impact of spatial and temporal resolution on model performance. We first discuss spatial resolution. Figure 5 shows that the model performs best for larger arcs (pipes) and compartments (aggregated areas of sewer network). This indicates, quite intuitively, that sewer elements whose behaviour is dependent on a larger area, or a larger collection of other sewer elements, are those whose behaviour is best captured by reduced complexity modelling. The seemingly unintuitive result seen in Figure 9, that compartment storage performance degrades as the total number of compartments increases from 100 to 500, is also explained by this finding. We can assume that the compartment storage performance improves between 30 and 100 compartments because, as shown in Figure 8, arc performance increases dramatically between 30 and 100 compartments, levelling off after 100. We can verify this by observing Figure S20, an experiment in which individual manholes were preserved and compared for node storage rather than entire compartments, where the effect is not seen. This implies that reduced complexity modelling requires getting the right balance between capturing spatial heterogeneity (i.e., getting good performance of arcs) and benefitting from the smoothing provided by aggregation (i.e., simulating components that aggregate behaviour over larger areas).

From a temporal resolution perspective, Figures 8 and 9 demonstrate that the presented approach can return reasonable performance in arcs and at compartments across a range of timestep sizes, between 30 seconds and one hour. Salvadore et al., (2015) show that the range of resolutions (both temporal and spatial) covered by this study are not unprecedented for any individual application. However, we cannot identify any sewer network simulation approach that can seamlessly navigate the entire range in a self-contained framework as we do in this study. We believe this flexibility will be of great utility to users because one may start with a desired simulation time in mind and identify their options for spatial/temporal resolution accordingly, as we show in Figure 10. We expect that these

fast and flexible simulation speeds may facilitate new applications for sewer network simulation models under continuous simulations that span far longer periods than can currently be simulated.

6 Conclusions

Modern applications of sewer network modelling, such as optimisation and climate uncertainty analysis, require far faster simulation speed than can be provided by the existing suite of high-fidelity models that are commonly used by water companies. In this study, we demonstrate the combination of graph partitioning for spatial aggregation and the CityWat-SemiDistributed modelling framework for hydraulic simulations. This calibration free approach can produce accurate simulations, particularly for larger pipes in the sewer network, at a fraction (<<1%) of the computational time of a high-fidelity model. Spatial and temporal resolution in this approach is entirely flexible and can be associated with a deterministic simulation time. We believe that the techniques presented in this study will open a wide range of novel applications for sewer modelling that have previously been prohibited by slow simulations and inflexible modelling approaches.

7 Acknowledgements

The research reported in this paper was taken as part of the CAMELLIA project (Community Water Management for a Liveable London), funded by the Natural Environment Research Council (NERC) under grant NE/S003495/1. The software used in this study is available at (Dobson, Watson-Hill, et al., 2021). This can be considered for transparency purposes only, since the authors are currently creating a more user-accessible software implementation of CityWat-SemiDistributed for a wide range of integrated modelling purposes. The data used in this study is either provided at (ibid.) or has instructions for acquisition at (ibid.). The raw network data cannot be shared due to privacy reasons; however, we provide a version of the data with randomised locations and pipe/node parameters at (ibid.). The authors would like to thank Thames Water for providing the urban drainage model of Cranbrook. We are grateful to Yuting Chen for providing the high-resolution storm precipitation data used in this study.

Appendix A – CityWat-SemiDistributed (CWSD) sewer modelling

In this appendix we first describe the fundamentals of integrated modelling in CWSD and then how the CWSD modelling framework has been applied for sewer network modelling. We then describe the physical processes that have been added and describe how they have been represented for our semi-distributed methodology. The term ‘sewer junction’ is used to represent all the manholes, breaks, lakes, and other node elements that make up the sewer network.

A.1 Integrated modelling with CWSD

As described in Section 2.1.2, CWSD is a node- and arc-based integrated modelling framework and software. Nodes represent physical system components or some semi-distributed aggregation of the system in question (i.e., a compartment), while arcs represent flows between nodes. Nodes and arcs are categorised into a variety of types, each type providing generic functionality for the kind of system

component that it represents. The node and arc types used in this study and their functionality is described below:

- Arcs track both incoming and outgoing water between two nodes. The arc capacity is determined using hydraulic gradient (Section A.2) and the travel time is tracked before water can enter a downstream node (Section A.3). The travel time along an arc is fixed at the velocity of a pipe's full-bore flow capacity.
- Sewer nodes represent either individual sewer junctions if they have been preserved for comparison purposes (as described in Section 2.3) or to maintain hydraulic structures (Section 3.2), or they represent an aggregation of sewer junctions and pipes within a compartment.
    - These nodes have a storage property that tracks how full they are. Their capacity is the total available capacity of all sewer junctions and pipes in the compartment that the node represents.
    - The key functionality of these nodes is to receive and discharge water. Received water is tracked to represent travel time across the compartment (Section A.3) before being added to the node's 'active storage', which is available for discharge downstream. A node's active storage is first discharged to downstream nodes in proportion to available capacity along outgoing links. After this, if the active storage still exceeds node capacity, then the excess is discharged to the adjoining land node as flooded water. A sewer node's total storage (both active and travelling within compartment) and its adjoining land node's flooded volume make up the 'total compartment storage' variable used for head calculations (Section A.2), and to assess model performance at nodes in Section 4.
- Land nodes represent the sub-catchments that drain into an adjoining sewer node.
    - These nodes have a storage property that tracks the flooded volume from spilled sewer nodes or water that cannot be received by sewer nodes.
    - Their key functionality is to read precipitation data and apply the time-area method to determine runoff (Section A.3).
- Outfall nodes represent the two outfalls of the catchment. They track received water for mass balance checks.

For a given timestep, first land nodes read precipitation and send runoff to sewer nodes. Secondly, starting from the most upstream sewer nodes and working downstream, sewer nodes discharge water. Finally, at the end of every timestep, mass balance checking is performed, both for each node individually and for the network as a whole. We summarise this process in Figure A1.

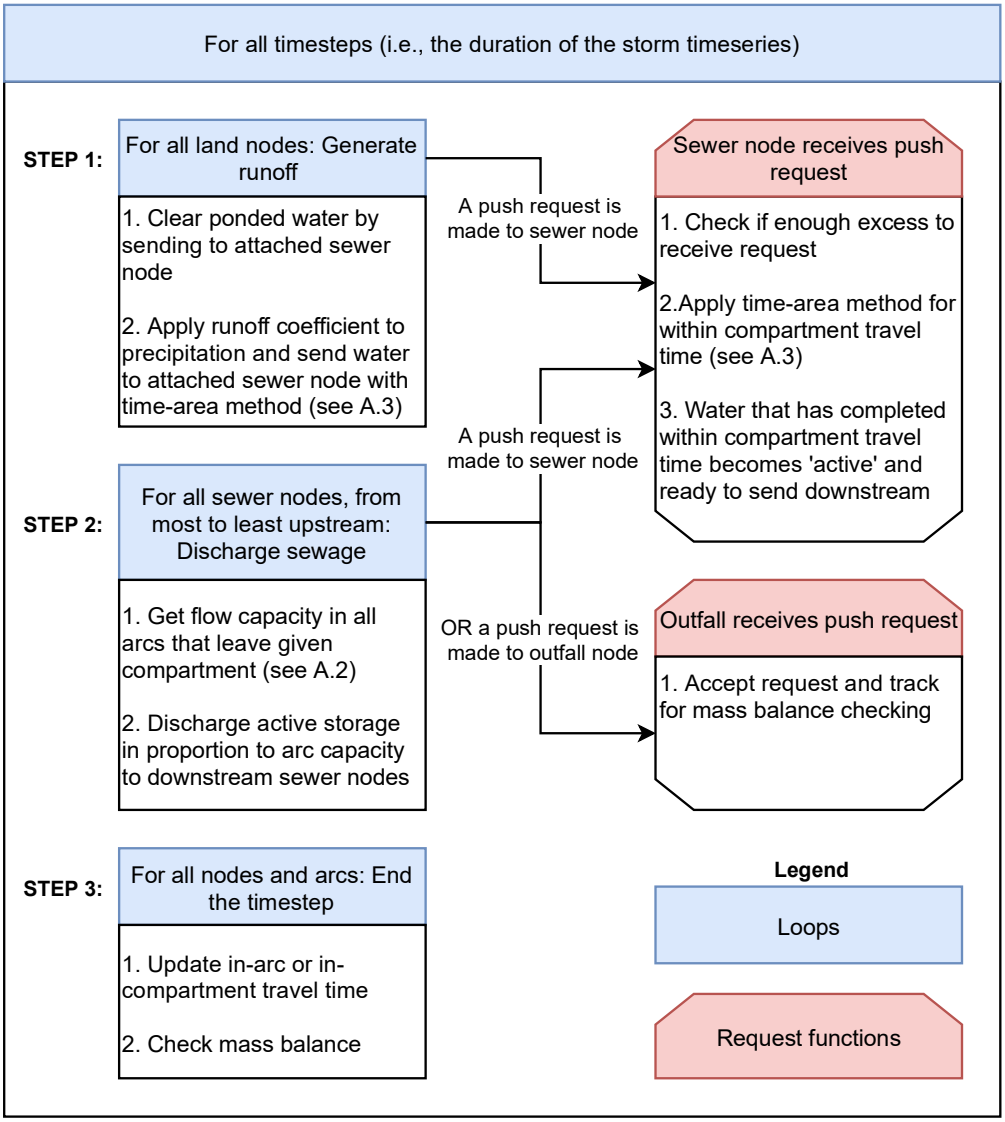


Figure A1: Schematic highlighting the orchestration of nodes during a timestep of simulation in this study's application of CWSD.

A.2 Hydraulics for between compartment pipe flow

Implementation of pipe hydraulics in this study was reasonably simple because the pipes between compartments are conceptually identical to specific pipes from the high-fidelity model. Thus, the parameters of those pipes (roughness, length, slope, cross-sectional area) remain the same, regardless of how a compartment is aggregated. These parameters enable Manning's equation to be evaluated to derive flow, which determines how much water may travel between two compartments. Manning's equation is given by equation 1,

$$Q_{i,j} = \frac{A_{p,i,j} R_{i,j}^{\frac{2}{3}} Sl_{i,j}^{\frac{1}{2}}}{n_{i,j}} \quad (1)$$

, where Q is the flow, A_p is pipe cross-sectional area, R is the hydraulic radius, Sl is the slope of the hydraulic gradient, n is the Manning's roughness coefficient along the pipe i,j flowing between compartment i and compartment j. To simplify calculations, A_p and R of pipe i,j are assumed constant

at its full-bore flow, thus implying flows are larger than in practice when pipes are not full. *Sl* is calculated using equation 2,

$$Sl_{i,j} = \frac{H_i - H_j}{L_{i,j}} \quad (2)$$

, where *H* is water head at a compartment and *L* is the pipe's length. While *L* is simply the length of the pipe *i,j* in the high-fidelity model, *H* is an aggregated value that represents water head at a compartment scale. It is described in equation 3,

$$H_i = \begin{cases} z_i, & V_i \leq V_{p,i} \\ z_i + \frac{V_i - V_{p,i}}{A_{c,i}}, & V_i > V_{p,i} \end{cases} \quad (3)$$

, where *z* is the floor elevation of compartment *i*, *V* is the volume of water currently in *i*, V_p is the total volume capacity of pipes in the compartment, and A_c is the total chamber area of sewer junctions in a compartment. Compartments have two possible values for *z*, an incoming and outgoing value. The incoming *z* is the average elevation of sewer junctions that receive water from pipes entering the compartment, while the outgoing *z* is the average elevation of sewer junctions from which pipes leave the compartment. As highlighted in equation 3, volume of water in a compartment that is less than the total volume of pipes in a compartment is assumed not to generate additional head.

The Cranbrook network used in this study has nine weirs. Among these, there are a variety of weir types and so we do not reproduce each equation separately here, however the same weir equation and parameters as in the high-fidelity modelling is used if an arc between compartments is also a weir. The head calculation for these weir equations is performed the same as in equation (3).

A worked example that demonstrates these equations for a simplified network is presented in Supplemental Material S1 and S2.

A.3 Travel time with Flow Time Information Packaging (FTIP)

Implementing travel time in network models is not trivial in purpose-built software, and this difficulty is exacerbated in an integrated modelling framework where nodes and arcs are discrete from each other, but computational efficiency aims to be preserved. Thus, we introduce a parsimonious approach that is designed to be quick to simulate, easy to understand, flexible to apply and as physically based as possible. We term this technique as 'Flow Time Information Packaging' (or FTIP). Using FTIP, every time water is sent somewhere by the model, it is sent in a package (implemented as a dictionary data structure) containing both the amount of water and an integer tracking remaining travel time. Every time a simulation completes a timestep the travel time integer is decremented by one until it reaches zero, at which point the arc attempts to send the FTIP package to the downstream compartment. If a downstream node returns that water could not be received, then that rejected water is assumed to have backed up and returns to the upstream node. Although this accounts for backwater, it prevents flooded sewer nodes from receiving more water. In practice flooded sewer nodes can still receive water if the hydraulic gradient permits, however we considered that calculating the head generated by flooded water above a sewer node would overly increase the complexity of the

model, running counter to its ultimate purpose. This will cause underestimates of downstream node flood volumes and overestimates of upstream node flood volumes; however, we consider the effect to be negligible and will not impact estimates of flood occurrence.

Implementing a typical pipe with FTIP requires translating some physical concepts into more discrete terms that can be simulated efficiently. We describe these below:

- Since FTIP tracks volume and not flow, for pipe capacity calculations FTIP packages must be translated into flow terms. For example, a pipe with a capacity of 10 m^3/s can accommodate two FTIP packages with a volume of 5 m^3 and travel time of 1s. It also means that pipe capacity is only applied to FTIP packages that are sent in each timestep. Thus, if a pipe has a pre-existing FTIP package from a previous timestep, that will not count towards the capacity calculation of the current timestep. This ensures that flow is accurate in steady state conditions but may not always be timestep accurate.
- An FTIP pipe may also back up based on a calculation performed on the pipe's outflow, given by equation 4,

$$V_{b,n} = \begin{cases} \sum_{m=0}^{n} V_{FTIP,m,i,j} - Q_{i,j}, & \sum_{m=0}^{n} V_{FTIP,m,i,j} > Q_{i,j} \\ 0, & \sum_{m=0}^{n} V_{FTIP,m,i,j} \leq Q_{i,j} \end{cases} \quad (4)$$

, where V_b is the volume of backed up water from FTIP package *n* that is about to be sent, $V_{FTIP,m,i,j}$ is the volume of a given FTIP package with a travel time of zero in pipe *i,j* and *m* is either an FTIP package that has already been sent in that timestep or the package *n*, and $Q_{i,j}$ is the flow capacity of pipe *i,j* according to equation 1 with units of volume per timestep.

Travel time in compartments is also tracked. A different method is used depending on whether the flow is arriving from an upstream compartment or for land runoff:

- Flow arriving from upstream compartments is assumed to take the average full-bore travel time between all the sewer junctions in the given compartment that receive flow from upstream compartments and all the sewer junctions that send flow to downstream compartments.
- Flow from runoff is modelled using the time-area method (Butler & Davies, 2004). In pre-processing, when compartment-scale parameters are calculated, a time-area diagram is calculated for each compartment. This is normalised with total drainable surface area of a compartment to associate percentages of rainfall with travel times. The time-area method also assumes full-bore flow travel time calculations.

8 References

Andrade, M. A., Choi, C. Y., Lansey, K., & Jung, D. (2016). Enhanced Artificial Neural Networks Estimating Water Quality Constraints for the Optimal Water Distribution Systems Design. *Journal of Water Resources Planning and Management*, *142*(9), 04016024. https://doi.org/10.1061/(ASCE)WR.1943-5452.0000663

Babovic, F., & Mijic, A. (2019). The development of adaptation pathways for the long-term planning of urban drainage systems. *Journal of Flood Risk Management*, *12*(March 2018), 1–12. https://doi.org/10.1111/jfr3.12538

Beven, K. (2012). *Rainfall-Runoff Modelling: The Primer*. https://doi.org/10.1002/9781119951001

Blondel, V. D., Guillaume, J. L., Lambiotte, R., & Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, *2008*(10), 1–12. https://doi.org/10.1088/1742-5468/2008/10/P10008

Bonald, T., de Lara, N., Lutz, Q., & Charpentier, B. (2020). Scikit-network: Graph Analysis in Python. *Journal of Machine Learning Research*, *21*(185), 1–6. Retrieved from http://jmlr.org/papers/v21/20-412.html

Buluç, A., Meyerhenke, H., Safro, I., Sanders, P., & Schulz, C. (2016). Recent advances in graph partitioning. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9220 LNCS*, 117–158. https://doi.org/10.1007/978-3-319-49487-6_4

Burger, G., Bach, P. M., Urich, C., Leonhardt, G., Kleidorfer, M., & Rauch, W. (2016). Designing and implementing a multi-core capable integrated urban drainage modelling Toolkit:Lessons from CityDrain3. *Advances in Engineering Software*, *100*, 277–289. https://doi.org/10.1016/j.advengsoft.2016.08.004

Butler, D., & Davies, J. w. (2004). Urban Drainage 2nd Edition. In *Urban drainage*.

DHI. (2014). *MOUSE Pipe Flow Reference Manual*. https://doi.org/10.1007/s13398-014-0173-7.2

Dobson, B., Jovanovic, T., Chen, Y., Paschalis, A., Butler, A., & Mijic, A. (2021). Integrated modelling to support analysis of COVID-19 impacts on London's water system and in-river water quality. *Frontiers in Water*, *3*(April), 26. https://doi.org/10.3389/frwa.2021.641462

Dobson, B., Watson-Hill, H., Muhandes, S., & Mijic, A. (2021). *CityWat-SemiDistributed Partition*. https://doi.org/10.5281/zenodo.5095282

Falter, D., Vorogushyn, S., Lhomme, J., Apel, H., Gouldby, B., & Merz, B. (2013). Hydraulic model evaluation for large-scale flood risk assessments. *Hydrological Processes*, *27*(9), 1331–1340. https://doi.org/10.1002/hyp.9553

Gong, Y., Li, X., Zhai, D., Yin, D., Song, R., Li, J., … Yuan, D. (2018). Influence of Rainfall, Model Parameters and Routing Methods on Stormwater Modelling. *Water Resources Management*, *32*(2), 735–750. https://doi.org/10.1007/s11269-017-1836-x

Hagberg, A. A., Schult, D. A., & Swart, P. J. (2008). Exploring Network Structure, Dynamics, and Function using NetworkX. In G. Varoquaux, T. Vaught, & J. Millman (Eds.), *Proceedings of the 7th Python in Science Conference* (pp. 11–15). Pasadena, CA USA.

Innovyze. (2014). *InfoWorks ICM manual*. Portland, USA.

Karypis, G., & Kumar, V. (1997). *METIS: A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices*. Minneapolis, USA.

Li, X., & Willems, P. (2020). A hybrid model for fast and probabilistic urban pluvial flood prediction. *Water Resources Research*, 1–26. https://doi.org/10.1029/2019wr025128

Liu, H., Zhao, M., Zhang, C., & Fu, G. (2018). Comparing topological partitioning methods for district metered areas in the water distribution network. *Water (Switzerland)*, *10*(4). https://doi.org/10.3390/w10040368

Muhandes, S., Dobson, B., & Mijic, A. (2021). A method for adjusting design storm peakedness to reduce bias in hydraulic simulations. *Proceedings of the Institution of Civil Engineers - Water Management*, 1–13. https://doi.org/10.1680/jwama.20.00092

Ochoa-Rodriguez, S., Wang, L. P., Gires, A., Pina, R. D., Reinoso-Rondinel, R., Bruni, G., … Ten Veldhuis, M. C. (2015). Impact of spatial and temporal resolution of rainfall inputs on urban

673 hydrodynamic modelling outputs: A multi-catchment investigation. *Journal of Hydrology*, *531*,
674 389–407. https://doi.org/10.1016/j.jhydrol.2015.05.035

675 Parés, F., Gasulla, D. G., Vilalta, A., Moreno, J., Ayguadé, E., Labarta, J., … Suzumura, T. (2018).
676 Fluid communities: A competitive, scalable and diverse community detection algorithm. *Studies*
677 *in Computational Intelligence*, *689*, 229–240. https://doi.org/10.1007/978-3-319-72150-7_19

678 Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E.
679 (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, *12*,
680 2825–2830.

681 Razavi, S., Tolson, B. A., & Burn, D. H. (2012). Review of surrogate modeling in water resources.
682 *Water Resources Research*, *48*(7). https://doi.org/10.1029/2011WR011527

683 Rossman, L. A. (2010). *Storm water management model user's manual, version 5.0*. Cincinnati.

684 Salvadore, E., Bronders, J., & Batelaan, O. (2015). Hydrological modelling of urbanized catchments:
685 A review and future directions. *Journal of Hydrology*, *529*(P1), 62–81.
686 https://doi.org/10.1016/j.jhydrol.2015.06.028

687 Shi, J., & Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern*
688 *Analysis and Machine Intelligence*, *22*(8), 888–905. https://doi.org/10.1109/34.868688

689 Thrysøe, C., Arnbjerg-Nielsen, K., & Borup, M. (2019). Identifying fit-for-purpose lumped surrogate
690 models for large urban drainage systems using GLUE. *Journal of Hydrology*, *568*(July 2018),
691 517–533. https://doi.org/10.1016/j.jhydrol.2018.11.005

692 Water UK. (2019). *A framework for the production of Drainage and Wastewater Management Plans*.
693 Retrieved from https://www.water.org.uk/wp-
694 content/uploads/2020/01/Water_UK_DWMP_Framework_Report_Main_September-2019.pdf

695 Yan, D., Huang, L., & Jordan, M. I. (2009). Fast approximate spectral clustering. *Proceedings of the*
696 *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 907–915.
697 https://doi.org/10.1145/1557019.1557118

698