TUDelft

Delft University of Technology

A Greedy Scheduling of Post-Disaster Response and Restoration using Pressure-Driven Models and Graph Segment Analysis

Deuerlein, Jochen; Gilbert, Denis ; Abraham, Edo; Piller, Olivier

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

A Greedy Scheduling of Post-Disaster Response and Restoration using Pressure-Driven Models and Graph Segment Analysis

Jochen Deuerlein[1], Denis Gilbert[2], Edo Abraham[*3], and Olivier Piller[4]

[1]3S Consult GmbH, Albtalstrasse 13, 76137 Karlsruhe, Germany
[2,4]Irstea, UR ETBX, Water Department, Bordeaux regional centre, Cestas F-33612, France
[3]Department of Water Management, Faculty of Civil Engineering and Geosciences, Delft Univ. of Technology, Stevinweg 1, 2628 CN, Delft, Netherlands

[*] *e.abraham@tudelft.nl*

ABSTRACT

In this manuscript, we consider the problem of optimally scheduling the restoration of a water distribution network with multiple failures after a disaster. The decisions made are the sequence of burst/broken pipes and leaks that need to be replaced or repaired, respectively, subject to constraints on workforce availability and physical hydraulic conditions. In order to sufficiently capture the objectives of the utility (e.g. service levels, resilience loss and customer minutes lost without service), which are all pressure dependent, pressure driven leakage and demand models (PDM) are employed.

The restoration decisions are modelled as time-dependent closure and opening of links and are simulated using a PDM in EPANET, propagating decisions as pressure-driven hydraulic constraints and computing a posteriori their impact on the multiple desired restoration objectives, some of which have trade-offs. The combination of discrete decisions with time-dependent couplings, and the presence of objectives that are conditional functions of demands met and time indices make it difficult to pose this optimal task scheduling problem as a standard numerically tractable mixed-integer programming problem. To make the problem tractable for the given large-scale water distribution system, we propose greedy heuristics that use a hierarchical decomposition of the decision space using structural properties of the network graph and hydraulics. Firstly, PDM simulations are used to sort the breaks and leaks from the biggest losses to the smallest, or determine visibility of the damages. In addition to solving the multi-criteria scheduling problem, we also use engineering principles to derive meta-heuristic that can prioritise water loss reductions. The greedy algorithm is employed to iteratively schedule isolation and repairs by first stabilizing the system with the isolation of the biggest breaks; an alternative approach considers all objectives 'equally'. With these we explore the trade-offs in response between water loss and resilience indicators.

To enable the scheduling, we use graph decomposition techniques to identify the valves that need to be closed to isolate a hydraulic segment (i.e. set of links sharing same closing valves) for replacement; this gives us a map (or look up table) that will be used in the scheduling. The map also includes information about the number of nodes isolated, unsatisfied demand when isolating each segment and the total pipe length of the segments. We also analyse the system flows, network pressures and how the depletion of tanks affects service levels. Using these, we make recommendations for improving the capacity of the system, including the improvement of pumping stations, installation of control valves and some pipe re-enforcement. The same greedy task scheduling algorithm is then used under these alternative network improvements, to show a much better response in all criteria.

Keywords: Disaster Response Scheduling, Graph Decomposition, Greedy Algorithm, Pressure Driven Modelling, Resilience

1   Introduction

Water distribution Networks (WDNs) are an essential part of the urban infrastructure and, as such, should have the capacity to maintain sufficient functionality under failure/disaster scenarios. Thus, water utilities also aim to ensure a quick response to effectively restore service levels when a natural or man-made disaster significantly damages the infrastructure. To manage disaster operations well, utilities need to consider all the four stages of *mitigation, preparedness, response and recovery* [1]. In this manuscript, we examine only the last two time-critical operations of response and recovery after disaster. In addition, by examining multiple disaster scenarios as in this exercise, it is also possible to improve mitigation (*e.g.* assessing and improving storage and pumping capacity, the ability of control valves to maintain pressure levels, and the number and locations of isolation valves in order to improve the restoration performance maximally) and preparedness (*e.g.* the availability and ability of human capital to respond).

This manuscript presents an approach for scheduling three crews in the disaster response and restoration of the WDN shown in Figure 1a, which is assumed to be damaged by an earthquake. This work is part of the the Battle of Post-Disaster Response and Restoration (BPDRR), the eighth battle competition of the Water Distribution Systems Analysis (WDSA) conferences to be presented at a special session of the joint WDSA/CCWI 2018 conference in Kingston, Canada [2].

Since the problem description is detailed in [2], we only briefly describe the network and problem objectives and constraints. Then, the methodology used for scheduling is explained, which consists of simulation based judgement using network analysis and visualization tools to simplify the decision space, graph analysis to generate link failure to segment maps (*i.e.* the set of links sharing same isolation valves) and a greedy-algorithm to iteratively update schedules. The results are then presented and discussed using the performance indicators, followed by a summary that includes a discussion of recommendations based on this exercise and future work to extend this work.

1.1   Some Information on the Modification of Simulation Tools

In this battle [2], the use of EPANET models is mandatory for assessing the performance of different actions through simulations. Within the given EPANET model, leaks and breaks are implemented using a combination of Check valves (CVs) and emitters. For leaks, when repairs are complete, the leak can be discarded by closing the pipes labelled "SXXX" and "HXXXX", which are connected to the emitter. For the breaks, these "SXXX" and "HXXXX" pipes are CVs to make sure there is no flow past the break. Since *time-based Controls* of Check valves is not allowed within EPANET, the isolation or removal after replacement of breaks is simulated by introducing small length 'lossless' pipes at the end of the CVs - see Figure 1b.

2   Scheduling Methodology

2.1   Segment Graphs and Schedule Maps

As this BPDRR problem demonstrates, valves play a critical role in the capacity of a utility to respond to failures in affective manner. As such, keeping track of the location, condition and status of valves
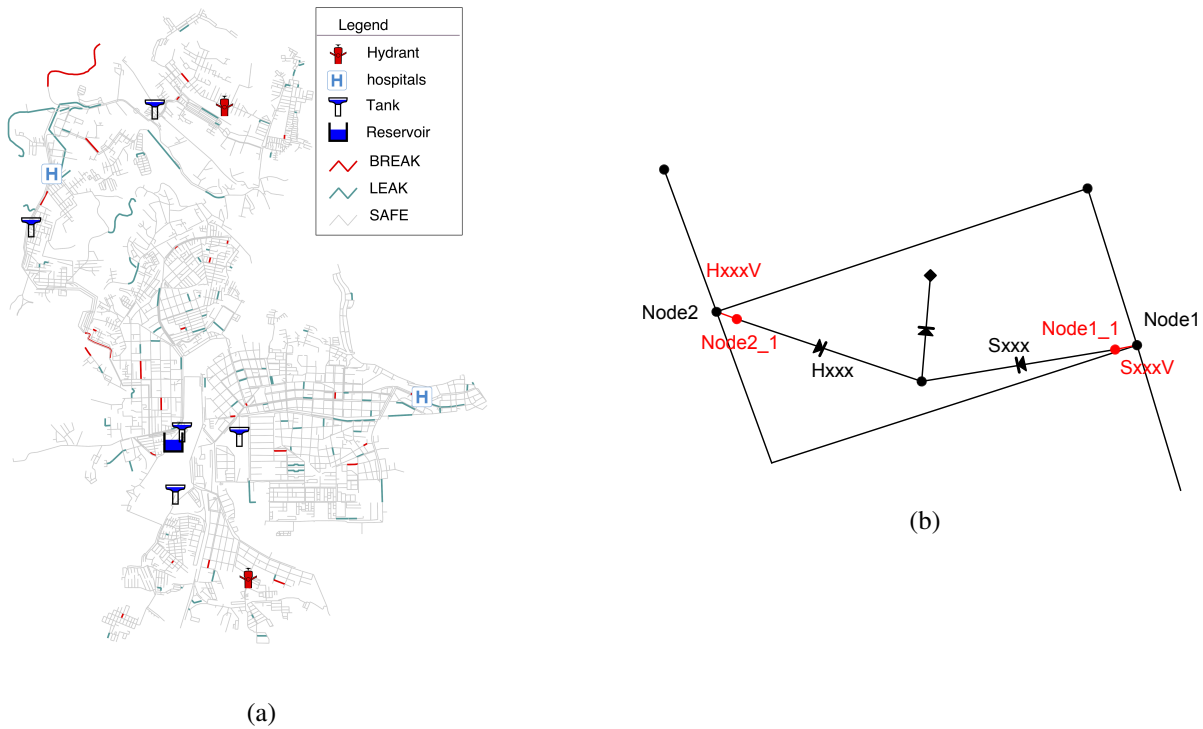
(a)

(b)

Figure 1: (a) A graph of the WDN with leaky and broken pipes, critical consumption nodes and storages distinctly depicted: Scenario 2 from the given five failure scenarios. (b) Model modification to simulate removal of replaced breaks in the EPANET model.

is an essential part of asset management and operations for water utilities [3].

As shown in the example network in Figure 2, isolating a broken pipe or link is not equivalent to simply closing it or removing it from a hydraulic model [4]. Assume, for example that pipes p_1 and p_2 in Figure 2a fail. Since p_1 has valves at each end, it can be isolated by closing these two valves. However, p_2 has only a single valve at one end and is connected to links with 1 or no isolation valves at the other end. In fact, most pipes in this network have either 1 or no isolation valve at their end. Therefore, isolating p_2 requires isolating the connected links and nodes in red labelled "2" in Figure 2(b), by closing off 7 valves. The connected set of links that share the same set of valves for isolation are called hydraulic segments [4, 5]; Figure 2(b) shows all the such segments with different colours for adjacent segments.

Using graph theoretical analysis, we also identify the valves that need to be closed to isolate a hydraulic segment for replacement; this gives us a map (or look up table as shown in Table 1) that will be used in the scheduling. Figure 2(b) shows that the extent of the line-outage as a result of a break in p_2 is also much bigger, including three nodes compared to no nodes for p_1. The segment map can also include this information about the number of nodes isolated, their criticality and unsatisfied demand when isolating each hydraulic segment to isolate one or more pipe failures in the segment. The connectivity among the segments is modelled by the so-called segment graph that consists of one node for each segment and links that represent the valves connecting the segments, as depicted in Figure 2(c).

The segment graph calculation is based on the decomposition of the network graph [5]. In the first step, the subdivision graph of the original network graph is calculated. Normally, the information

about the locations of isolation valves are stored as properties of the pipes; as given in the description code of the pipes in this exercise [2], there is no extra link associated with the valves in the hydraulic model. In contrast, the subdivision graph has a link for each isolation valve. Since the increased number of links and nodes would heavily impact the performance of hydraulic network simulations, the two topologies are maintained separately.

The identification of segments is done by basic connectivity analysis of the subdivision graph where the links of isolation valves are not considered. As a consequence, the subdivision graph is subdivided into maximal connected components, the hydraulic segments of the water distribution network. Please note that there is no general 1:n relationship between segments and links of the original graph [5]. For the creation of the subdivision graph the original pipes maybe split into different subsections depending on the location of the isolation valves. However, for this battle it is assumed that the valves are all located at the ends of pipes [2]. Therefore, the original links belong to exclusively one segment. This greatly simplifies the mapping between links and segments. To compute the segment graphs the software SIR 3S$^{\circledR}$ (www.3sconsult.de) has been used, which includes a suite of network topology tools including subdivision and segment graph calculation. The map of hydraulic segments was generated, with IDs of pipes and nodes in each segment, and the valves needed to isolate each segment, which were later used in the scheduling, see Table 1 for an example of segment map data used in scheduling.

Table 1: Segments Map membership for (broken) links for Scenario 2.

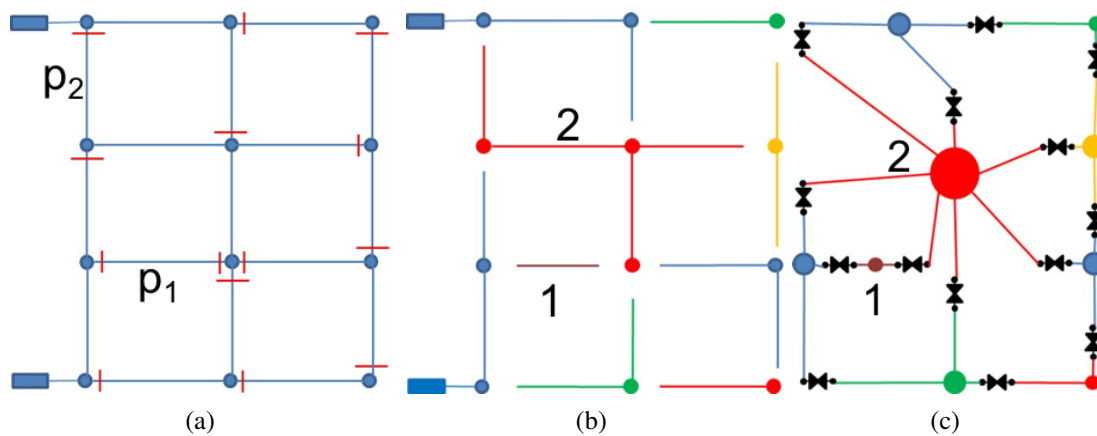| $PipeID$	# valves to close	Pipes Closed for Isolation	Initial Loss (L/s)	Diameter (mm)
3414	4	3280,3359,3402,3443	17.1	200
1951	3	1618,1948,1951	16.52	200
4988	2	4988	13.58	200
5251	2	5251	14.38	200
3404	4	3404,3344,3399,3401	9.35	200
6005	3	6005,6026,6027	8.91	150
⋮	⋮	⋮	⋮	⋮

(a)   (b)   (c)

Figure 2: (a) Network Graph with valves (b) hydraulic Segments and (c) segment graph

2.2   A Greedy Scheduling Heuristic

The aim of this exercise is to find the 'best way' to respond to the disaster and to restore functionality. However, there are multiple objectives as set out in the battle problem [2] for assessing the response. For example, the best way to minimise water loss (i.e. minimise objective C_6) is the simple heuristic of sorting all leaks and breaks by the amount of loss, isolating all in order of decreasing loss levels, and then optimising the supply restoration process (with respect to other objectives) only after all breaks and leaks are isolated and repaired, respectively. As can be seen from Table 1, the biggest losses are in the larger diameter pipes, which have the longest replacement time. For example, it takes around 20 team hours to isolate all the breaks for Scenario 2 but it takes more than 30 hours to replace only the first four biggest breaks. However, not replacing these larger pipes sooner may also mean having many more nodes without supply (i.e. larger values for C_1, C_4 and C_5, and indirectly for C_3), with non-trivial trade-offs.

In order to 'solve' this multi-objective problem and explore trade-offs among the multiple criteria, we can consider a weighted sum of the objectives,

$$f := \sum_{i=1:6} w_i C_i, \tag{1}$$

where w_i are weights that capture the relative importance of each objective C_i.

In scheduling a response by minimizing (1), the decisions are modeled as time-dependent closure and opening of links and are simulated using a PDM in EPANET, propagating decisions as pressure-driven hydraulic constraints and computing their impact on (1). The formulation of this scheduling problem contains discrete variables for scheduling decisions, continuous variables for pressures and flows, and nonlinear complementarity constraints (i.e. task dependencies, and changing connectivities depending on tasks) on crew allocation involving mixed variables with different time requirements at differing locations, and the multiple objectives in the restoration decisions (*e.g.* $C_{01} - C_{05}$) that are conditional non-smooth functions of demands met and time indices. All these, together, make it difficult to pose the optimal scheduling problem as a numerically tractable mixed-integer programming problem. Therefore, we consider simulation based heuristics as an alternative to explore the search space.

To compute a schedule that 'approximately minimizes' (1), we employ a greedy algorithm - a heuristic that seeks a solution to the scheduling problem by making a sequence of local optimal choices that aim to approximate the global solution. Although such a heuristic cannot guarantee closeness to global optima, greedy algorithms and multi-stage heuristics have been shown to be effective for many combinatorial problems [6], including optimisation problems for water distribution networks [7, 8, 9]. If we consider a large time horizon with multiple schedules for each team (*e.g.* time long enough for 95% functionality to be restored, or even much shorter time with only a handful of tasks for each team), the solution space and the number of extended time PDM simulations needed to explore the space grows combinatorially and becomes computationally infeasible. Our simulation based heuristic is greedy in the sense that it makes decisions based on the incremental contribution of a single task at a time for each available team.

At the start of the algorithm all jobs for leaky and burst pipes are classified according to their types (isolation, repair or replacement) and their duration. As tasks are accomplished, the remaining tasks are also updated; isolated breaks are updated with replacement tasks. At every 15 minute interval,

which is the smallest allowed unit of task period in the exercise [2], the algorithm checks if one of the teams has just finished its current job and is therefore free to start a new one. To solve the scheduling problem the jobs are distributed among the three teams according to their availability; see Algorithm 1. Depending on the choice of strategy (*i.e.* the weights chosen for each criteria in (1) or whether we first isolate all losses before optimising other criteria), the most significant jobs are identified.

Since the multiple criteria are of different units and magnitude, it is not trivial how weights w_i should be chosen or what the appropriate trade-offs between the different criteria are for the utility. For the greedy algorithm, we consider minimising the weighted relative increase in objective functions for each criteria; *i.e.* we schedule the job that results in the least relative increase among all criteria weighted also by how long the job takes:

$$\tilde{f}(t^k; J_l) := \sum_{i \in \{1, 3-6\}} \frac{\alpha_i}{f_i(t^k) J_l} \left[f_i(t^{k+J_l}) - f_i(t^k) \right] = \sum_{i \in \{1, 3-6\}} \alpha_i \frac{f_i(t^{k+J_l}) - f_i(t^k)}{f_i(t^k) J_l} \qquad (2)$$

where α_i are optional weights for each criterion, J_l is the number of 15 minute time intervals needed to complete the lth job from all available tasks remaining in the TasksToDoSet at the time t^k, and $f_i(t^k)$ are C_i computed up to time t^k and are non-decreasing with time (see also Algorithm 1). Note that, if more teams are available at time t^k, the first team decision considers simulations with just one intervention and chooses a job. The decision for the next team takes into account the first schedule already chosen to start at the same instant t^k, and so on. Although this may be less optimal than considering all combination of tasks for the available teams at t^k, it is not an important issue in practice. After the first few iterations, it is rare that more than one team is available at the same instant t^k.

The greedy algorithm we implement considers all objectives except for C_2, which requires long time horizons to compute and is not (necessarily) defined over the decision horizon of the next scheduled event at each iteration. This criteria is, however, computed a posteriori for completed schedules to assess relative impact of the different approaches on C_2.

Considering the number of PDM simulations needed to compute all task schedules, Algorithm 1 scales well with the number of damages. At each time instant, the algorithm runs simulations only if there is a team available. If so, an extended time PDM simulation is carried out for each remaining task at the time. As the tasks are accomplished with time, the number of simulations needed at each instant decreases. Therefore, if there are a maximum of N tasks to accomplish over the whole response time, the number of PDM simulations needed can be bounded by $\frac{N(N+1)-1}{2}$, *i.e.* by roughly $N^2/2$ for a large N. This bound is derived with a simple partial sum of the number of tasks available as a series, with the number of tasks decreasing by one each time a team is available. As the PDM simulations are the bottleneck, time complexity for the algorithm scales similarly. Since all PDM simulations of different tasks are independently computed, they can be parallelized (*i.e.* assuming we have sufficient parallel resources; see line 8 of Algorithm 1). By parallelizing all tasks' simulations the time complexity can be reduced significantly, the bound becoming proportional to $N/2$. Note also that each extended time PDM computation is performed only over the task horizon.

Algorithm 1 A greedy task scheduling algorithm for disaster response scheduling

---

Input: Network data and failure scenarios, simulation period (t_0, t_f) (response time is in minutes)

Output: Schedules and predicted performance for all criteria

    Initialisation: Hydraulic segment analysis to derive isolation maps; initial losses and criteria values accrued until response starts

 1: Schedules $\leftarrow \{\}$

 2: TasksInProcessSet $\leftarrow \{\}$

 3: $k = 0, \quad t^k = t_0$

 4: TasksToDoSet $\leftarrow$ all tasks available at the start of response

 5: AvailableTeamsSet $\leftarrow$ all teams available at t^0

 6: **while** $t^k < t_f$ & TasksToDoSet is not empty **do**

 7:   **if** AvailableTeamsSet is not empty at time t^k **then**

 8:     Compute $\tilde{f}(\cdot)$ for each task in TasksToDoSet

 9:     **for** Each Team in AvailableTeamsSet(t^k) **do**

10:       Assign task that has the lowest $\tilde{f}(\cdot)$

11:       TasksToDoSet $\leftarrow$ Update by removing assigned task at t^k, update isolated breaks for replacement tasks

12:       AvailableTeamsSet $\leftarrow$ Update with assigned Team and its next time of availability

13:       Schedules $\leftarrow$ Add new tasks assigned, Team ID, and start & end times

14:     **end for**

15:   **end if**

16:   $t^k \leftarrow t^k + 15$

17: **end while**

18: Compute Criteria over 7 days

19: **return** Schedules, All performance criteria

---

3   Results and Discussion

Although many alternatives can be considered by using Algorithm 1, we consider two metaheuristics here for the sake of brevity. The first approach sets all α_i in Equation (2) to 1 over the whole scheduling horizon, *i.e.* all five objectives are equally weighted, which we call "Free Optimisation" from here on as no extra preferential constraints are imposed in Algorithm 1. As an alternative, a more sustainability focused approach prioritises water loss reduction by first dealing with isolation of leaky breaks before considering all of the criteria together as in the first case; we call this an "Isolate First" approach. The isolate first approach considers first only C_6 in Algorithm 1 (*i.e.* Equation (2) has all α_i set to zero except for $\alpha_6 = 1$), where the greedy impact ranking is done by assessing current water losses when decisions are scheduled. During the first 48 hours, the heuristic also computes the water loss levels at each 15 minutes interval by non-isolated bursts or non-repaired leaks to check if they have become visible at the time the decision is made. At each instant, isolation of visible breaks and then repairs of visible leaks are considered; the latter is done only if there are no breaks to isolate at the time instant. We consider the repair of leaks in the "Isolate first" approach because we discovered from simulations that they take much less time than replacing breaks and also the battle description does not allow isolation of leaks. Once isolation of all breaks and repairs of all leaks are finished,

the heuristic switches back to optimising replacements using the same objective evaluation as the free optimisation.

The performance of the schedules derived by these two metaheuristics are summarised for all the disaster scenarios in Table 2. For all the five scenarios, the isolate first approach results in significant savings in total volume of water lost over the 7 days; the free optimisation solutions result in 25%-40% more water loss across the different scenarios. However, this comes at the cost of worse performance with respect to other criteria.

We visualise this trade-off between water loss minimisation and other criteria in Figure 3, where the ratio of the criteria $c_i(FreeOpt)/C_i(IsolateFirst)$ for all $i \in \{1, \ldots, 6\}$ are plotted for three scenarios. From the figure, we see that either some or all other criteria become worse depending on the scenario. In Figure 4, we also show the functionality loss (*i.e.* loss of water supply rate) and leakage levels at each instant of the scheduling horizon. The total area under the leakage curve is total volume lost (C_6), and the area under the functionality loss curve represents resilience loss (C_3). In all scenarios, the isolate first approach reduces the leakage level to zero more quickly (by 1-2 days), resulting in a lower value for C_6. This is because isolating breaks and repairing leaks takes much less time compared to replacing breaks. By scheduling pipe replacement to last, losses are stopped sooner.

However, waiting for isolating and repair tasks before replacing (significant) breaks also means that it could take longer to restore supply to many customers without supply (higher values for criteria C_4 and C_3), including critical customers (higher values for criteria C_1), or more customers being without water longer (higher values for C_5). In all scenarios, C_2 seems to be the least impacted when we prioritise loss reduction. One reason can be the fact that it takes many days to restore supply to 95% level because it requires replacements and low leakage levels to achieve anyway; whether we prioritise water loss reduction first or not, loss rates are close to zero by the time 95% supply is restored. To some unknown extent, C_2 may be least affected because of the fact that we do not explicitly optimise it and so the two approaches perform similarly with respect to this criterion.

Of course, these trade-offs are not trivial and depend on the burst scenarios, *i.e.* where the damages are and their severity. For example, in Scenario 2 and Scenario 5, by prioritising water loss reduction, no hospitals and fire-fighting nodes are impacted at all - C_1 is the same for both metaheuristics. This is because the breaks do not impact these nodes in the first place. However, in Scenario 1, the break on pipe 437 (a 500 mm mains pipe that transports most of the water to the northern part of the network through the larger pumping station) plays a vital role in functionality loss, including in the supply of the hospital and fire hydrant in the north; please see also Figure 1a. This pipe break causes over 102 L/s loss at the start and is the first to be isolated by the "isolate first" metaheuristic. However, until this pipe is replaced, neither the critical customers in the north of the network are supplied, nor does the functionality go above 95% (compare also Figure 4a and Figure 4d). After isolation of all breaks, and repairs of leaks, this pipe is the first to be replaced as its restoration suddenly increases functionality at around 113 hours after the start of the response. Since the improvement in PDM conditions by isolating losses will have no impact on topologically disconnected parts of the network, we recommend that this should be analysed before prioritising water loss reduction.

Table 2: The performance criteria for response schedules by the greedy algorithm; 'Isolate First' refers to the case where loss minimisation is first prioritised and then other criteria are optimised via replacements when all losses are isolated, and 'Free Opt' refers to the case where Algorithm 1 optimises objective (2) freely with equally weighted criteria over the whole response horizon.

	Scenario 1		Scenario 2		Scenario 3		Scenario 4		Scenario 5	
Criteria	Free Opt	Isolate First	Free Opt	Isolate First	Free Opt	Isolate First	Free Opt	Isolate First	Free Opt	Isolate First
C_01 (minutes)	1620	5670	0	0	90	60	720	3045	0	0
C_02 (minutes)	3705	6780	3645	3645	5085	3705	5085	5205	3645	4545
C_03 (minutes *%)	1434.5	5619.7	616.2	667.2	842.7	773.7	1152.3	2138.0	849.57	1307.6
C_04 (minutes)	149.9	797.8	10.3	64.5	14.0	24.3	63.9	231.2	65.3	109.6
C_05 (#)	613	741	11	71	9	17	304	332	54	119
C_06 (x1000 L)	66,702	47,952	51,988	39,425	73,346	63,137	64,579	46,099	54,342	40,547

(a)   (b)   (c)

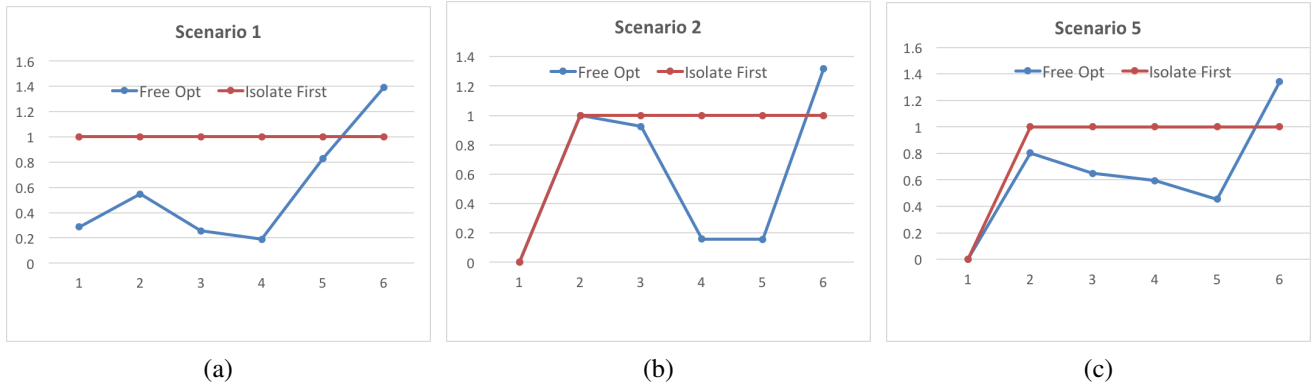*Figure 3: A plot of the relative performances of the 'Isolate First' vs 'Free Opt' approaches (i.e. $c_i(FreeOpt)/C_i(IsolateFirst)$) applied to example scenarios 1, 2 and 5. Absolute values computed for each criteria are in Table 2.*

4   Summary and Recommendations

In this work, we have shown that post disaster response can be scheduled using a combination of graph decomposition tools (for hydraulic segment analysis) and a greedy algorithm. In addition to its computational efficiency (i.e. time complexity of the Algorithm scaling with $N/2$, where N is the number of restoration tasks) and its ease of implementation, we have been able to show the performance of two metaheuristics using this greedy approach. For example, we have shown that prioritising water loss reduction may delay the restoration of supply, affecting all other criteria negatively. Based on this, we recommend using both free and isolate first approaches to understand these trade-offs before making a decision. Perhaps, in a city with dire water shortages, it may even be necessary to choose loss reduction over resilience loss.

(a) Scenario 1, FreeOpt   (b) Scenario 2, FreeOpt   (c) Scenario 5, FreeOpt

(d) Scenario 1, Isolate First   (e) Scenario 2, Isolate First   (f) Scenario 5, Isolate First
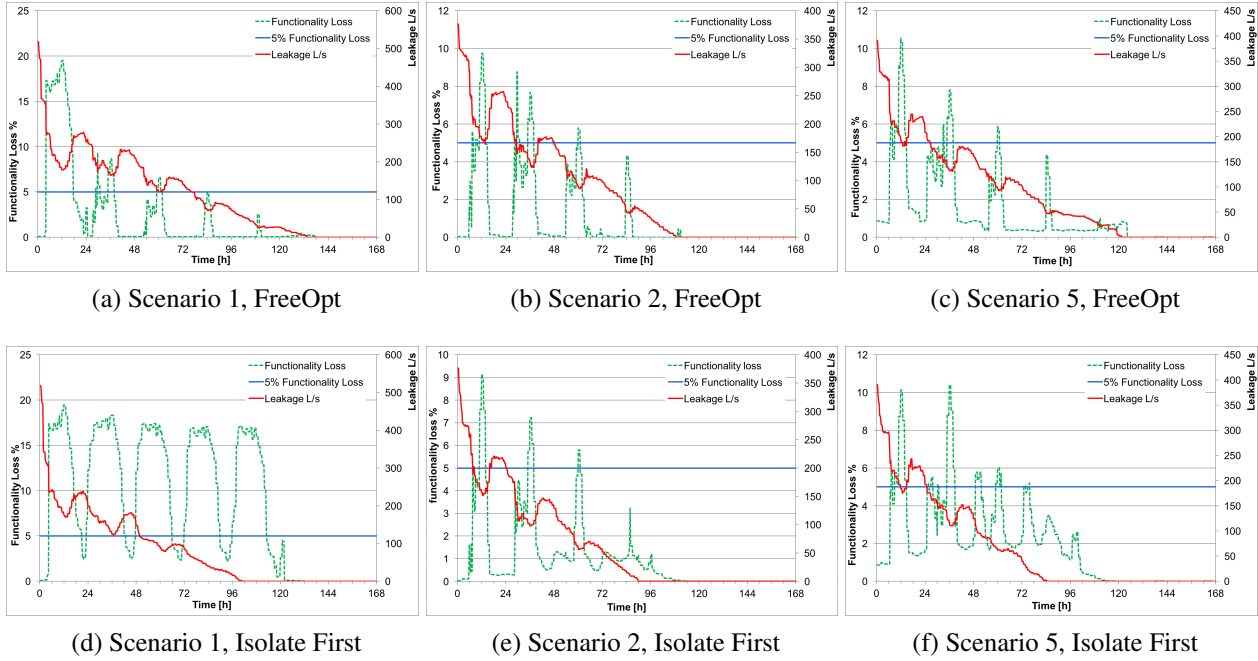
Figure 4: The evolution of the different objectives with time, functionality (water supply rate) and leakage levels shown with 95% functionality line, for the Free Optimisation and Isolate First heuristic. See Table 2 for accumulated final values C_i for all criteria.

Network Modifications

Although we have provided a methodology for optimising the response scheduling, even in the free optimisation based heuristic where most criteria have better performance, the response may not be satisfactory. For example, from Table 2, it takes some 3645 minutes (i.e. 2.5 days) to restore service level to 95% for the two most likely scenarios (i.e Scenarios 2 and 5, which together account for $\approx 76\%$ likelihood of damages [2]). In Scenario 1, the big damage to a mains pipe resulted in the north part without supply for longer, affecting all customers including hospitals and fire-fighting. Therefore, we did simulation based analysis of the original undamaged system and the different schedules to ascertain how resilience of the network can be improved through a combination of network reinforcements and response scheduled of such an updated system.

In Figure 5 we show the dynamics of the tank levels for the original undamaged network (diurnal) and under disaster response over seven days for scenario 2. For all scenario disaster responses, it was readily apparent that the tanks were completely drained in the first few hours, stayed depleted for more than 8 hours at a time in the first three days, and never go back to their original levels until service level is fully restored - compare the performance in Figures 4b and 4e, respectively, with the corresponding tank levels in Figures 5b and 5c, respectively. Therefore, increasing tank volumes may improve performance. For the same network with all tank volumes doubled, Figures 6a shows the evolution of losses and functionality in disaster response and in Figure 6b we show the corresponding tank levels over the seven days. Compared to the original network (see tank levels in Figure 5b), this improves response performance *only immediately after the disaster*, where none of the tanks are completely depleted for some eight hours. As a result, criteria C_3, C_4 and C_5, improve marginally. However, improving supply in the first day by having more network pressure also increases total
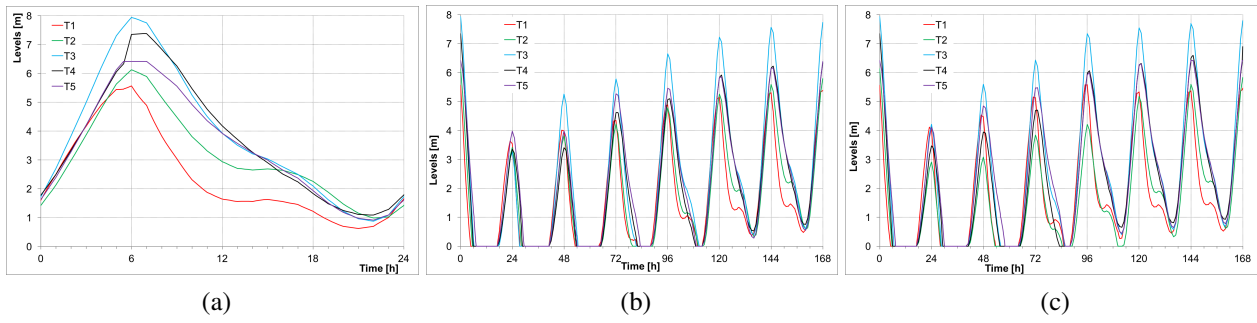
Figure 5: The profile of tank levels (a) the diurnally repeated tank levels for the original undamaged network (b) for Scenario 2 with intervention via the greedy algorithm (Free Opt in Table 2) (c) for Scenario 2 with intervention via the greedy algorithm (Isolate First Opt in Table 2)

volume loss (C_6) marginally; please also see column (a) of Table 3. This tank volume change can be considered as a static measure, and its utility may not be as important in the dynamic response we consider in the BPDRR [2]. Therefore, we also consider alternatives to improve pumping and pressure control devices.

This analysis and a look at the flows for both the original and the network with doubled tank volumes revealed that the central pumping station (at the outlet of the reservoir "R1"), could not sustain sufficient pressure to sufficiently fill the tanks, whether the volumes of the tanks were doubled or not. Therefore, in addition to increasing tank capacity, we considered the following (by no means exhaustive) list of improvements to the network:

(a) **Double tank volumes to improve initial response**: discussed above.

(b) **Increase the capacity of the central pumping station (Pump ID 6071)**: this was introduced to improve the pressure levels feeding into the areas south and east of reservoir "R1", which have the lowest pressure levels after six hours of the disaster. For example, we considered replacing the pumping station with a Variable Speed Drive (VSD) pumps, so that pressure at the downstream of the reservoir can be maintained at 155m. This value was derived after experimenting with a few parameter levels, and will mean an extra pressure of 54m introduced by the VSD. In the EPANET model, this was performed by closing the original Pump and setting the reservoir elevation to 155m.

(c) **Multiple actions**: this was introduced to also improve the pressure levels feeding into the northern part of the network, which suffered substantially in Scenario 1, where a mains pipe passing the northern pumping station (Pumps 6068, 6069, 6070) was broken. The improvements involved

    i) Increase the capacity of the central pumping station (Pump ID 6071) as in (b), but with the outlet pressure fixed at 150m rather than 155m.

    ii) **Exchange the end nodes of pumps 6068 and 6070**: connect Pump 6068 to Pipe 781, and Pump 6070 to Pipe 606. Then, increase the diameter of "Pipe 781" to 500m to increase supply to north east part.

    Pipe 436 (350mm) and Pipe 437 (500m) together are the main supply to the north part. Since Pipe 437 supplies most of the water, of that most passing through Pump 6068,

reconfiguring the outlet of the pumps can increase supply pressure. Pump 6070 is fed by a 350mm pipe, and should then be connected to the 350mm diameter pipe (pipe 606).

iii) **Replace throttle valves with pressure reducing valves (PRVs)**: Replace valve 6074 by PRV with fixed head setting 40m, and replace valve 6073 by PRV set to 60m.

The throttle control valves were introducing big losses and reducing flow to the south and eastern parts of the network. By replacing them with PRVs, higher pressure can be maintained and flow increased. The PRVs would automatically open when pressure levels fall downstream, allowing more flow downstream.

The performance of the response schedules for the improved network and Scenario 2 of the disaster are shown in Table 3 and Figure 6. The improvements in control devices and pumping have the most impact on performance.

Improvement (b) results in a C_2 value of 0, as supply levels stay above 95% from the start. The loss of functionality is also smaller at all times, resulting in lower value for C_3, C_4 and C_5. From Figure 6d, we note that tank "T1" always stays around full as the VSD keeps the pressure at the tank inlet high. Since, tank T1 serves no purpose in this new configuration, it can be removed/decommissioned to save money. Of course, we may also keep it to increase storage capacity in case of a failure at the central pumping station.

Improvement (c) also results in a C_2 value of 0 with above 95% supply and smaller functionality loss at all times, resulting in the lowest values for C_3, C_4 and C_5 compared to all other configurations in Table 3. In this case,tank T1 is more dynamic than case (b) and supplies parts of the south and eastern parts of network together with tanks T2 and T3, the later tanks becoming less dynamic compared to previous configurations. This is also because the PRVs on the main pipes supplying these tanks regulate pressure levels. The other main improvement is the provision of supply to the northern parts, which is the main improvement in the criteria values compared to the network modification in (b).

Table 3: The performance criteria for response schedules by the greedy algorithm 'Free Opt' when improvements have been applied to the network as described in (a), (b) and (c) above.

	Network modification			
Criteria	Original	(a)	(b)	(c)
C_01 (minutes)	0	0	0	0
C_02 (minutes)	3645	3645	0	0
C_03 (minutes *%)	616.19	370.88	121.95	114.58
C_04 (minutes)	10,25	7,74	11,20	10,70
C_05 (#)	11	7	6	4
C_06 (x1000 L)	51,988	57,248	60,794	63,477

(a) Tank volumes doubled

(b) Increased capacity for central pumping station

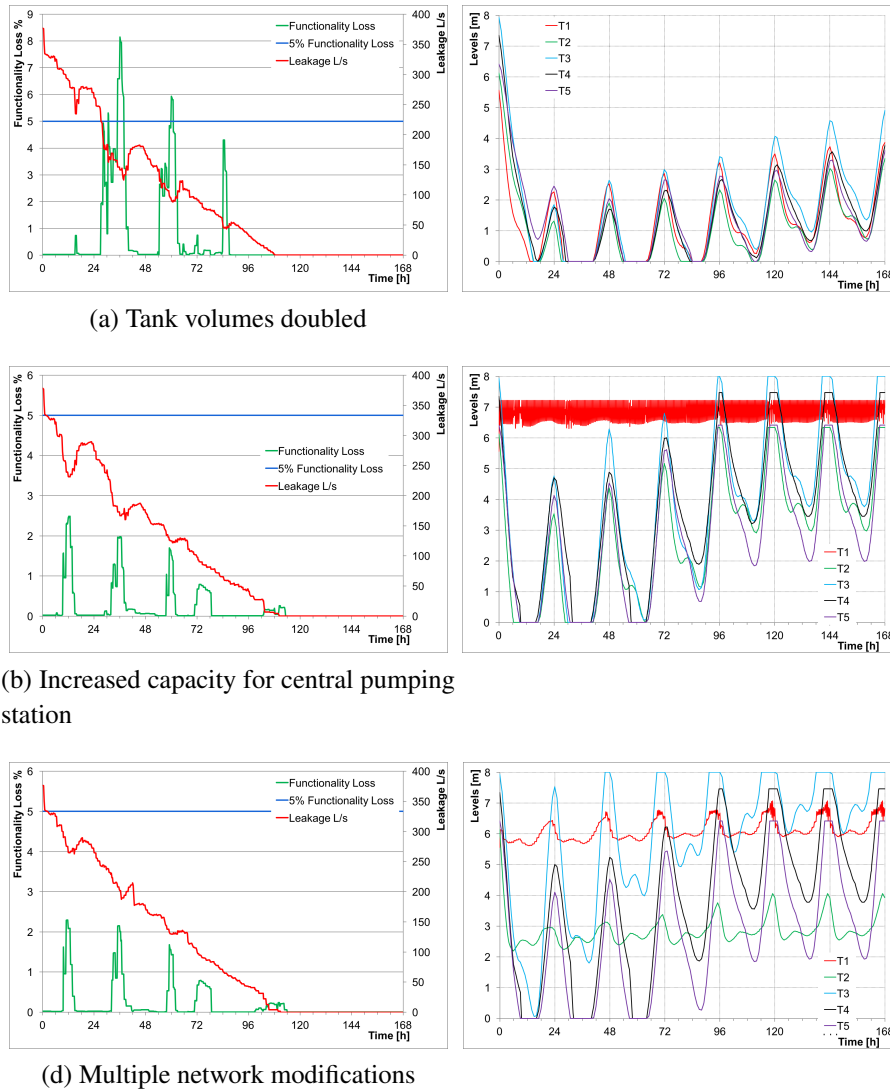
(d) Multiple network modifications

Figure 6: Restoration performance for improved networks described in (a), (b) and (c) above in.

Future Work

Although the proposed algorithms have been useful, with insightful schedules and trade-off analyses, they are suboptimal methods by definition of being computationally greedy. Future studies can also include a more thorough exploration of the decision and objective space through the following:

- Longer decision time horizons, where a dynamic programming approach can be adopted. In our "greedy algorithm" implementation, the decision taken at each time instant depends on decisions in the past. Therefore, the lack of backtracking means that our solutions may potentially be far from optimal.

- A multi-objective approach where the interaction between different objectives can be analysed in more depth.

- PDM computations were terminated after a number of simulations (40 or 100 here). We noted that EPANET returned values were sensitive to whether one used a 32 bit or 64 bit implemen-

tation. Although the errors were small, they accrued over time to make noticeable differences in the objective evaluations; especially when the values were small.

- Some important pipes do not have multiple or nay valves, while some segments have too many valves and took too long to isolate. Future work can explore where new valves should be installed in order to improve the restoration performance maximally (*e.g.* average demand shortfalls, reaction time, resilience loss etc.) under multiple disaster scenarios.

- The proposed modifications were mainly based on engineering judgement and network graph simplification and visualization tools. This is certainly not optimal with reference to the VSD and PRV settings. Optimisation can be employed to determine PRV locations and settings, and VSD pump settings with respect to resilience indices [10, 11].

References

[1] N. Altay and W. G. Green III, "OR/MS research in disaster operations management," *European journal of operational research*, vol. 175, no. 1, pp. 475–493, 2006.

[2] D. Paez, Y. Fillion, and M. Hulley, "Battle of post-disaster response and restauration (BP-DRR): problem description and rules," 2018, published online at http://www.queensu.ca/wdsa-ccwi2018/problem-description-and-files, version: 2018-02-12.

[3] M. Blokker and I. Pieterse-Quirijns, "Asset management of valves," *Water Asset Management International*, vol. 7, no. 4, pp. 08–12, 2011.

[4] T. M. Walski, "Water distribution valve topology for reliability analysis," *Reliability engineering & system safety*, vol. 42, no. 1, pp. 21–27, 1993.

[5] J. W. Deuerlein, "Decomposition model of a general water supply network graph," *Journal of Hydraulic Engineering*, vol. 134, no. 6, pp. 822–832, 2008.

[6] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to algorithms," *Growth*, vol. 5, no. A8, p. B1.

[7] R. Pinzinger, J. Deuerlein, A. Wolters, and A. Simpson, "Alternative approaches for solving the sensor placement problem in large networks," in *World Environmental and Water Resources Congress 2011: Bearing Knowledge for Sustainability*, 2011, pp. 314–323.

[8] J. Uber, R. Janke, R. Murray, and P. Meyer, "Greedy heuristic methods for locating water quality sensors in distribution systems," in *Critical Transitions in Water and Environmental Resources Management*, 2004, pp. 1–9.

[9] D. Gilbert, E. Abraham, I. Montalvo, and O. Piller, "Iterative multistage method for a large water network sectorization into dmas under multiple design objectives," *Journal of Water Resources Planning and Management*, vol. 143, no. 11, p. 04017067, 2017.

[10] F. Pecci, E. Abraham, and I. Stoianov, "Scalable pareto set generation for multiobjective co-design problems in water distribution networks: a continuous relaxation approach," *Structural and Multidisciplinary Optimization*, vol. 55, no. 3, pp. 857–869, 2017.

[11] R. Menke, E. Abraham, and I. Stoianov, "Modeling variable speed pumps for optimal pump scheduling," in *World Environmental and Water Resources Congress 2016*, 2016, pp. 199–209.