Multi-objective optimisation of sewer maintenance scheduling

Sabrina Draude [ID]a,*, Ed Keedwellb, Zoran Kapelan IWAa,c and Rebecca Hiscockd

a College of Engineering, Mathematics and Physical Sciences, University of Exeter Engineering, Exeter EX4 4QF, UK
b College of Engineering, Mathematics and Physical Sciences, University of Exeter Computer Science, Exeter EX4 4QF, UK
c Department of Water Management, Delft University of Technology, Stevinweg 1, 2628CN Delft, The Netherlands
d Dwr Cymru Welsh Water, Fortran Road, St Mellons, Cardiff, CF3 0LT, UK
*Corresponding author. E-mail: sd406@exeter.ac.uk

[ID] SD, 0000-0003-0337-0659

ABSTRACT

Effective functioning of sewer systems is critical for the everyday life of people in the urban environment. This is achieved, among other things, by the means of regular, planned maintenance of these systems. A large water utility would normally have several maintenance teams (or crews) at their disposal who can perform related jobs at different locations in the company area and with different levels of priority. This paper presents a new methodology for the optimisation of related maintenance schedules resulting in clear prioritisation of the ordering of maintenance tasks for crews. The scheduling problem is formulated as a multi-objective optimisation problem with the following three objectives, namely the minimisation of the total maintenance cost, the minimisation of travel times of maintenance teams and the maximisation of the job's priority score, all over a pre-defined scheduling horizon. The optimisation problem is solved using the Nondominated Sorting Genetic Algorithm-II (NSGA-II) optimisation method. The results obtained from a real-life UK case study demonstrate that the new methodology can determine optimal, low-cost maintenance schedules in a computationally efficient manner when compared to the corresponding existing company schedules. Daily productivity of maintenance teams in terms of number of jobs completed improved by 26% when the methodology was applied to scheduling in the case study. Given this, the method has the potential to be applied within water utilities and the water utility Welsh Water (Dŵr Cymru Welsh Water (DCWW)) is currently implementing it into their systems.

Key words: multi-objective optimisation, planned maintenance scheduling, sewer system

HIGHLIGHTS

- A new methodology for optimised scheduling of maintenance activities in sewer systems is presented.
- The new methodology is capable of determining optimal, low-cost maintenance schedules that are superior to the existing ones.
- When compared to existing company schedules, the new methodology has increased the daily productivity of maintenance teams in terms of jobs completed daily per crew.

INTRODUCTION

Wastewater sewer networks are critical components of urban infrastructure. They consist of sanitary, storm or combined sewers and rising mains. Wastewater sewer networks collect wastewater from residential, industrial and commercial sources and convey it to wastewater treatment plants, which provide chemical and filtration treatment to the wastewater before it is discharged into the environment (Butler et al. 2018).

At present most water and sewerage companies (WaSCs) in developed nations carry out sewer maintenance throughout their networks to maintain sewer functionality. Sewer maintenance can be reactive, planned or proactive. Planned management involves preventing a known problem before it happens again, reactive management involves fixing something after it has become a problem and proactive maintenance involves preventing a problem before it ever occurs. Planned maintenance (PM) tasks available include defining issues through closed-circuit television (CCTV) and visual inspections, and repairing and addressing issues through jetting, flushing, desilting and tree root removals. The main motivation for PM is to avoid pollution and (local) flooding incidents that often follow sewer blockages. A blockage is an obstruction within a sewer not formed by a hydraulic overload, which causes a reportable problem (Water Services Regulation Authority (OFWAT) 2017).

Several investigations have been carried out in the literature with the aim of utilising the available historical incident and asset data owned by WaSCs to help predict future blockage incidents using data-driven techniques. For WaSCs there is a desire to gain blockage likelihood at the pipe level, to help with their maintenance scheduling. However, data are often limited to local area level, rather than global scale as seen by Savic et al. (2008). WaSC asset databases often also have poor linking of assets and no provision of pipe blockage risk. Bailey et al. (2016) used geographical aggregation to reduce the noise present in the data and to get a complete representation of the surrounding network for input to a blockage likelihood model.

Pipe prioritisation within sewerage systems suffers from various shortcomings. The first issue is that long-term impacts of decisions within the optimisation process are not usually considered. This is because long-term costs for maintenance decisions have many possible impacts each with high uncertainty. In addition, within the literature previous prioritisation models did not favour highly critical pipes, and instead optimised the average sewer network performance. In 2016, Altarabsheh (2016) investigated the breakage likelihood of pipes and developed a methodology that used a system of decision-support framework, which gives the ability for WaSCs to account for the impacts of feedback loops between the integrated models. However, their model is scenario-based and thus contains predetermined behaviours of its subsystem and therefore cannot model the dynamic behaviour of the subsystem and their interactions with one another. The non-interactive nature of the model may result in failure to capture important emerging behaviours of the wastewater system.

In the context of sewerage systems, little attention has been focused on combining planning, scheduling and executing maintenance operations. In 2009, Berardi et al. (2009) used multi-criteria optimisation on pipe failure risk, inspection costs and repair costs to identify the best Pareto-optimal inspection programme. The model works by ranking objects in the network based on how many times they were selected by the Multi-Objective Evolutionary Algorithm (MOEA). An archive of optimal inspection policies is created. The solutions produced by the study are highly dependent on the algorithm settings. As a result, the decision-maker cannot be sure that the final solution provided by the model is the most optimal. Similarly, Ward & Savić (2012) developed a new methodology for the optimisation of sewerage system rehabilitation activities. They used the following three criteria: maximal improvement of the structural condition of the system, minimisation of construction costs and minimisation of sewer failure risk. Their results show that the model strives towards optimal solutions, which would otherwise be overlooked. However, despite being a flexible and powerful tool, the optimality of the solutions derived from the model has not yet been globally guaranteed. Similarly, the work by Chen et al. (2017) which tackles the problem arising from planning maintenance of gully pots in Blackpool, UK cannot guarantee optimality. They present a risk-driven model that evaluates the performance of suggested schedules. They simulated the schedule by generating routes, which minimises the total travelling distance; however, they do not take into account the potential maintenance costs when evaluating job scheduling. They also fail to consider the potential costs of not prioritising critical maintenance jobs. Comparable work on optimising PM for sewerage systems is presented by Fontecha et al. (2016), which focuses on optimising the routing schedules and in turn minimising the costs. They use the methodology offered by Rodríguez et al. (2012) to incorporate the unforeseen nature of failures and adapted the combined maintenance and routing approach by López-Santana et al. (2016). The Rodríguez et al. (2012) methodology consisted of determining a plan of action for each set square within the case study. Their model attempts to achieve the best balance between proactive and reactive maintenance to minimise the overall cost of system operation. The Fontecha et al. (2016) methodology proved successful in reducing costs and is a benchmark example on how routing and scheduling optimisation problems (RASOPs) can be solved for WaSCs. However, the study remained limited in results due to a lack of data. Several studies have used multi-objective optimisation models within the water industry, for example, Turriago et al. (2014) applied a multi-objective optimisation model to a proactive maintenance plan, to prioritise maintenance on sewers with a high failure probability due to sediment-related blockages in a small city. The model was highly successful and correctly identified which areas of the city needed to be maintained, the maintenance scheme to apply, the specific asset to maintain and the schedule of maintenance. However, the model is rigid in its costing format, and does not account for potential costing changes through time. Outside of the water industry, De et al. (2019) developed a multi-objective mathematical model which addressed the routing and scheduling of maritime transport vessels. Despite the method being computationally slow, the MOEA provided diverse alternative solutions and helped to avoid short-sighted decision-making. MOEAs have previously been used for sewer network rehabilitation (Ward & Savić 2012; Ngamalieu-Nengoue et al. 2019) and pipe inspections (Berardi et al. 2009; Elmasry et al. 2019), but not for real-world maintenance scheduling as proposed here in this paper. This paper builds on the conference paper (Draude et al., 2021) by introducing the representation and methodology used to solve the routing and scheduling PM problem.

To conclude, many similar problems have been studied on pipe maintenance but they do not consider the monetary constraints or impact assessments that dictate the company's priorities and actions and therefore a new methodology is required. There is currently no multi-objective optimisation-based scheduling and routing of maintenance in wastewater systems, which considers job priority, cost and travel time. Previous research for examining the potential for reducing blockage has primarily been concerned with interrogating historical sewer blockage records and scheduling proactive cleaning to the worst-performing parts of the network (Yin 2020).

Therefore, this work will introduce a novel, multi-objective optimisation-based approach for the problem of real-world PM scheduling for routing sewer maintenance. The methodology presented in this paper proposes to facilitate timely PM on high-priority jobs on the network to safeguard infrastructure and the environment by preventing blockages and their subsequent negative impacts such as flooding and pollution. The work presented explores the sensitivity of the multi-objective algorithmic parameters within one MOEA used to solve the RASOP optimisation problem for a WaSC. The methodology produced is validated by simulations as well as a real-world case study field test.

MATERIALS AND METHODS

Challenge within scheduling wastewater PM

The challenge is to determine how best to schedule the activities related to PM of a sewer network. A set of pre-defined maintenance jobs $J_y=(1, 2 ,… Y)$ that are geographically distributed (i.e. at different geographical locations) need to be completed over a scheduling horizon (equal to 1 month here). Jobs have different priorities and can be of different types (CCTV, flushing, jetting, visual inspections, root cutting, etc.) affecting the time required to complete the job and related costs.

A total of (M) maintenance teams (or crews) are available to complete the maintenance. Each crew has a homebase (defined by its postal code) from which the crew starts its daily travel and to which it has to return at the end of each working day.

The aim is to develop a methodology capable of identifying an optimal maintenance schedule that ensures all jobs are completed and in order of decreasing priority while minimising the costs and travel times. All wastewater companies try to reduce their costs, therefore cost minimisation is an objective. The methodology minimises the travel time objective to ensure that as many jobs are completed per day as possible while maximising the priority objective to ensure that higher priority jobs are completed as soon as possible to minimise the chance of a blockage within the wastewater pipe network.

The scheduling methodology is based on the following assumptions:

- The service time is different for each job category.
- The crews are homogeneous, i.e. they have the same work capacity for their working period. It means that each vehicle has an assumed time for their travelling time and service time.
- The crews start and finish at their individual homebase.
- The travel times are deterministic, there is an assumed constant travel speed for each crew.
- Travel cost per mile is constant.

The above assumptions reduce the likelihood of obtaining global optimum solutions, for example, one crew may be more productive than another but this is not considered in the costing model averages. However, adding in only these broad assumptions do amenities of concern increase the computational efficiency of the methodology and the ability of the MOEA to be run in real-time. Yet, the theory of evolutionary algorithms (EAs) states that we cannot guarantee to find the optimum solution using EAs. It is also hard to find the absolute optimum solution due to the uncertainties of predicting blockage likelihood and scheduling routing. This is due to uncertainties in model calibrations and prediction estimations. Nevertheless, the preceding assumptions do not cause a loss of solution feasibility, which is more important than global optimality for this problem.

Optimisation problem formulation

To address the above challenge, a three-objective optimisation problem is formulated and solved here.

Objectives

The optimisation objectives of the optimisation problem are as follows:

- Minimise the total scheduling cost C of maintenance activities.

- Minimise the crew travel time T during the maintenance day.
- Maximise the priority score P of the jobs completed.

The total scheduling cost C is calculated as follows:

$$C = \sum_{i=1}^{N} (C_{job}^{(i)} + C_{travel}^{(i)}) + C_{return} \tag{1}$$

where $C_{job}^{(i)}$ is the cost of a job (i), $C_{travel}^{(i)}$ is the cost of travelling from the previous location to the ith job, $C_{travel} = c_{mile} \times D$ where D is the distance travelled (miles) and c_{mile} is the unit cost of travel (equal to 1.20 £/mile in the case study here) and C_{return} is the cost of the crew returning to the homebase and equates to $c_{mile} \times D$ where D is the distance travelled (miles) and c_{mile} is the unit cost of travel.

The cost of completing the ith job is calculated as follows:

$$C_{job}^{(i)} = \sum_{j=1}^{N_a^{(i)}} (c_{activity}^{(j)} \times t_{activity}^{(j)}) + C_{material}^{(i)} + C_{miscellaneous}^{(i)} + C_{Overtime}^{(i)} \tag{2}$$

where $c_{activity}^{(j)}$ is the unit cost (£/hour) of the crew to perform jth maintenance activity within the ith job (the cost of activities depends on the cost of the type of job being done ($t_{activity}^{(j)}$)), $t_{job}^{(j)}$ is the amount of time it takes to complete jth activity within the ith job (h), $N_a^{(i)}$ is the number of individual (different type) activities in the ith job, $C_{material}^{(i)}$ is the cost of the materials used to complete the maintenance job, $C_{miscellaneous}^{(i)}$ is the extra cost of additional resources required to complete the ith job, $C_{Overtime}^{(i)}$ is the extra cost for working outside the normal crew working hours.

The total distance travelled by the kth crew $D^{(k)}$ is as follows:

$$D^{(k)} = \sum_{i=1}^{N^{(k)}} (D_{travel}^{(i,k)}) + D_{return}^{(k)} \tag{3}$$

where $D_{travel}^{(i,k)}$ is the distance travelled by the kth crew between the previous location and the location of their ith job, $D_{return}^{(k)}$ is the distance between the last job of the day and the crew homebase, $N^{(k)}$ is the total number of distances covered by the kth crew.

The total priority score for a maintenance schedule is calculated as:

$$P = \sum_{i=1}^{N} P_{job}^{(i)} \tag{4}$$

where $P_{job}^{(i)}$ is the priority score of the ith job, calculated as follows:

$$P_{job}^{(i)} = P_{model\_total}^{(i)} + P_{CSO}^{(i)} + P_{DG5}^{(i)} + P_{SEF}^{(i)} + P_{OSF}^{(i)} \tag{5}$$

where $P_{model\_total}^{(i)}$ is the priority score of the ith job as dictated by the priority model, $P_{CSO}^{(i)}$ is the additional priority score added if the job is classified as a CSO (combined sewer overflow), $P_{DG5}^{(i)}$ is the additional priority score if the job is classified as a DG5 event (properties at high risk of internal sewer flooding), $P_{SEF}^{(i)}$ is the additional priority score added if the job is classified as a serious external flooding (SEF) event (property at risk of SEF), $P_{OSF}^{(i)}$ is the additional priority score added if the job is at an OSF (onsite sewer facility).

The priority model has been created in geographic information system (GIS) and uses watercourse location data, blockage likelihood data estimated using the methodology by Bailey *et al.* (2016), historical repeat blockage incident data and population density data to calculate the priority of the jobs to be completed. The methodology by Bailey *et al.* 2016 has already been applied to the case study area of the paper and proven to be robust and accurate. If the postcode is in a highly populated region, close to a watercourse, has already had a blockage incident and is likely to have another then the postcode will have a

high-priority score. If the postcode is in a sparsely populated region, far from a watercourse, has never has a blockage incident and is unlikely to have one in the near future then it will have a low priority score. After the model has run it outputs a set of priority scores for each postcode within the case study area that the user is running the model.

The priority model methodology undergoes several steps in ArcMap before the priority scores are outputted into the MS access database. Each job for the month is assigned a priority score using the methodology ($P_{model\_total}^{(i)}$). The methodology is generic and can be applied to any case study area:

- First, the population density per postcode, the watercourse location data, the repeat incidents per postcode and the blockage likelihood per postcode (priority data) are added to ArcMap.
- A shapefile to cover the user case study region is created.
- The priority data for each case study are then clipped to their respective shapefiles.
- The model creates hotspots based on historical incidents of the postcodes, using 200 m as the radius search size.
- The model creates a population density raster dataset.
- The model creates a distance raster from the rivers in the case study area, using 200 m as the furthest distance for a blockage to be able to pollute a river.
- The model creates blended hotspots of all four variables, with a priority score assigned to every hotspot zone by using a weighting system. The model then assigns a weighting to the variables. 60% was assigned to the point dataset containing repeated incidents, population, and watercourse proximity data and 40% was assigned to the blockage likelihood model results.
- The assigned priority scores were then created into data points across the case study area.
- x, y coordinates were added to the priority point data.
- Priority point data were joined with all the postcodes in the case study areas.
- Export the attribute table of priority point data per postcode into the preliminary database (P_{model_total}).

The total travel time is calculated as follows:

$$T = \sum_{i=1}^{N} (T_{travel}^{(i)}) + T_{return} \tag{6}$$

where $T_{travel}^{(i)}$ is the distance travelled between the previous location to ith job, T is the travel time for the crew returning to the homebase. The known distance travelled between each previous location to the ith job is provided by the distance model. The distance model creates a matrix of distances between all jobs available for each month and each crew start and end location (otherwise known as the crew's homebase) using the library 'openrouteservice' (ORS) (Schmitz *et al.* 2008). ORS was used because it is free to access, has no data caps and quickly creates a matrix using a large dataset. The distance model assumes an average speed of travel of 45 mph.

Constraints

The constraints of the problem are to:

- Set a constraint violation (*CV_Ti*) if the schedule overruns the total allowed scheduling time for the day. The constraint violation incurs a cost fine, which is set relative to the violation. The *CV_Ti* constraint is applied if the total crew hours scheduled exceeds the total scheduling time available for the crew plus 30 min of overtime for each day (in case of traffic or jobs overrunning the expected time). The following constraint violation will be activated:

$$C = C + \sum_{i=1}^{N} + = (total\_day\_duration - extra\_time) \times 2 \tag{7}$$

where *extra_time* is the total scheduling time available for the crew plus 30 min of overtime for each day in case of traffic or jobs overrunning the expected time. Where *total_day_duration* is the total time of working and travelling completed by the crew for the day. Where C stands for the total scheduling cost outlined in Equation (1).

- Set a constraint violation (*CV_Tr*) if the crew travel time exceeds more than 20% of their total daily schedule. Previous guidelines published by DCWW suggest that crews should be completing maintenance for at least 70% of their total

shift time. The condition of constraint violation CV_Tr is that if the total travel duration of the crew per day is greater than a fifth of the total of the crew's scheduled working day, then add a constraint violation.
- Set a constraint violation (*CV_D*) if any of the jobs are duplicated in the total schedule. The condition of constraint violation *CV_D* is that if the individual job numbers for the final schedule are duplicated anywhere within the schedule, then add a constraint violation. The constraint violation checks whether the length of the job numbers is the same after the duplicate job numbers are removed by using the set function in python.

Decision variables

The problem is represented to the algorithm as a permutation of *n* jobs in *t* timeslots using an indirect representation that ensures that the proscribed work can be completed within a given period. Each position in the chromosome represents a potential timeslot into which a job can be inserted. Each job has an associated duration and during initialisation jobs are added constructively to a crew's day until no further jobs can be inserted, ensuring adherence to the overtime constraint of initial solutions. Jobs that would exceed the day's work are added to the following day's schedule. Any jobs that have durations that exceed the final day of the schedule are considered not to have been completed and do not count towards the objective values for that solution. The mutation and crossover operations will result in a perturbed schedule which will also have the above rules applied, ensuring that workers are not overworked, but raising the prospect that previously completed jobs may now no longer be completed by the end of the week and will fall from the end of the schedule.

The chromosomes for this problem are demonstrated as follows, where each chromosome is encoded as an *nt* (number of jobs in this case) element array:

$$\text{Chromosome}_t = [n1, n2, n3, \ldots, nt]$$

The decision variables (*nt*) for the problem represent the allocation of jobs for the specified scheduling time period. For example, in a month there may be 170 jobs with 26 h or slots to schedule the jobs in. So, there are 4,420 different slots in which jobs could be placed.

Optimisation method

The above optimisation problem is solved by using the soft computing method: MOEA. MOEAs are computationally efficient in comparison with other soft computing methods such as various stochastic algorithms, however, for MOEAs the dominant factor of the probability rules of the operator affects the global search solution (Shende & Chau 2019). Unlike deterministic models, soft computing methods are not limited to a particular case study. Soft computing methods can also overcome difficulties in missing sampling data using averages whereas statistical methods are restricted while using assets with insufficient previously recorded information (Amiri-Ardakani & Najafzadeh 2021). The Strength-Pareto Evolutionary Algorithm-II (SPEA-II) and Nondominated Sorting Genetic Algorithm-II (NSGA-II) have both been found effective within the literature at solving similar problems to this (Hasani & Hosseini 2020). Salazar *et al.* (2016), studied more than 100,000 separate optimisation problems, using all MOEAs to solve their problems. The study found that the NSGA-II was one of the most scalable genetic algorithms and a top performer in terms of its hypervolume. However, the study found that NSGA-II generally required larger initial search populations than some of the other MOEAs. The study found that SPEA-II, despite having high zones of performance across the experiments, did not have clear trends in its results. Instead, SPEA-II showed isolated parametric bubbles of good performance. Other MOEAs which have been less frequently used on the RASOP have also been used for comparison for their ability to solve this problem, namely IBEA, MOEAD, GDE3 and OMOPSO (Petelin *et al.* 2021). Other possible optimisation methods include tabu search, which is a metaheuristic search method that employs mathematical optimisation methods for local search. Unlike MOEA methods, tabu search methods only work well with small instances. If there are over 50 tasks to be scheduled the solution takes a long time to reach (Mathlouthi *et al.* 2021). Other optimisation techniques explored to solve the problem include rollout algorithms, ant colony optimisations and particle swam methods, but these methods all performed worse than NSGA-II on highly complex scheduling problems (Goodson *et al.* 2016). Ultimately, the NSGA-II, MOEA method was selected as it demonstrated the best search performance detailed in the case study below and the literature reviewed on RASOPs.

To provide best performance on the problem, the parameters of NSGA-II were optimised, including the initial population size, the selection, crossover and mutation operator types and related rates and the total number of generations to be used for

optimisation. The supplementary material in Supplementary Material Section 1.1 provides the details of different options considered. The adopted operators and parameter values that are used in the case study are summarised in Table 1. The alternative tested approaches by the author are also outlined in the table. Sensitivity analysis using the hypervolume indicator was used to determine the best approaches for the different MOEA parameters used in the case study and shown in Table 1.

Once the MOEA run is completed a Pareto-optimal front of maintenance schedules is generated. As a final step, the multi-criteria decision analysis (MCDA) type method is used to select a single schedule to implement. This is done by using a simple linear weighted sum MCDA method (Marler & Arora 2010). This method is used because it is consistent and easy to implement. The simple linear weighted sum method has also produced similar results to the harder to interpret MCDA methods in other academic studies such as the single synthesising criterion approach and other outranking methods (El Amine *et al.* 2014). This method is used to determine the best solution for the user from the final population produced by the optimisation phase. Further details on the simple linear weighted sum MCDA method can be found in Marler & Arora (2010). The robustness of this method was also confirmed via additional analyses (see supplementary material section 1.2 for details).

CASE STUDY

Description

The scheduling methodology presented in this paper is applied to a case study in Swansea, South Wales. This research project has been formed in partnership with DCWW that provided data and maintenance teams for the Swansea catchment over the time of the research. The case study is shown in Figure 1, which represents the catchment area of Swansea. The routing and scheduling methodology was applied to the Swansea catchment in November 2020 using the data detailed above and maintenance teams provided by DCWW to analyse and demonstrate the effectiveness of the methodology. In 2020, the methodology was run for two PM crews in Swansea, with 105 jobs to be scheduled. The methodology was run once for every 3 working days. Every job was completed by the 11th working day for crew 1 and the 13th working day for crew 2. The methodology is compared against the 2019 human-derived methodology used by DCWW. The present general company practice (used in the 2019 results) is firstly for the network team of engineers and specialist technicians to create a list of PM jobs which need to be completed over specified periods of time to reduce blockage likelihood, based on historical feedback from reactive crews from reactive blockage events. The PM manager gets given a set of PM jobs every month, which they assign equally to the PM crews for each area. The jobs given to the crews do not come with scheduled routes, so the scheduler assigns the best route usually by grouping four-digit postcodes. This can lead to poor planning, and often incomplete jobs at the end of the month.

The performance of the MOEA applied to the methodology depends highly on the operators and parameters used in the optimisation algorithm and the specific criterion for success in the MCDA. This includes the probability used for crossover, the probability used for mutation, the population size, and the number of iterations. These parameters can be adjusted after

Table 1 | Selected and tested MOEA parameters

| Component	Best approach	Alternative tested approaches	Methodology used to test the different approaches
MOEA	NSGA-II	SPEA-II, IBEA, MOEAD, GD3, OMOPSO	Hypervolume scores after 50,000, 60,000 and 40,000 iterations using parallelisation computing
Selector	Tournament selector	Steady state, elitism selector	
Crossover	Simulated binary crossover	Cycle crossover, partially mapped crossover	
Mutation	Flip bit, with probability	shrink	
Population size	125	50, 100, 150, 175	
Number of generations	50,000	40,000, 60,000	
Mutation rate	0.3	0.1, 0.2, 0.4, 0.5	
Crossover rate	0.5	0.4, 0.6, 0.7	
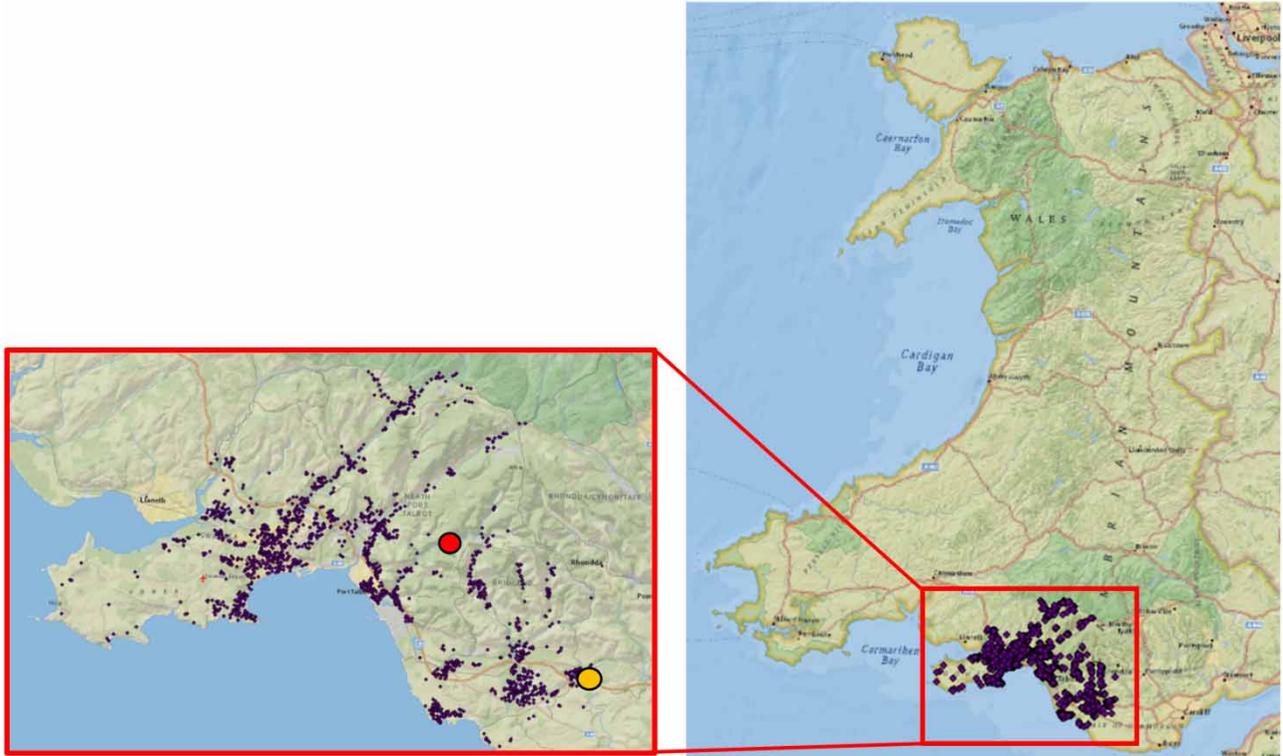
Figure 1 | Swansea, Wales case study region. The purple dots indicate maintenance jobs to be completed in November 2020. The red dot represents the location of the homebase of crew 1 and the orange dot represents the homebase of crew 2. Please refer to the online version of this paper to see this figure in colour: http://dx.doi.org/10.2166/hydro.2022.149.

assessing the algorithm's performance after a few trial runs on the historical input data. The methods used for MCDA on the solutions and the decision of which MOEA to use for this problem must also be assessed. All the experimentations and tests for this case study have been run using Python 3.8 on a windows 10 64-bit machine, with 8GB of RAM. For this case study several sensitivity analyses and various performance indicators on historical data were used to decide the best way to formulate the methodology for the live Swansea case study. First, the method was run for the Swansea catchment using historical data from September 2020, using NSGA-II for 80,000 iterations for the 3-day scenario, to see at which point the data converges. For the scheduling scenario the MOEA was run 10 times. The results of the long hypervolume run displayed that the MOEA had fully converged by 50,000 iterations for the 3-day scenario. Second, to decipher which MOEA to use for the problem, several well-known MOEAs, proven by the literature to solve similar optimisation problems, were run on this problem. The MOEAs tested were SPEA-II, NSGA-II, IBEA, MOEAD, GDE3 and OMOPSO. All of the MOEAs were run on the problem for 50,000 iterations, 10 times a day for the 1-day, 3-day and 5-day thresholds and were assessed using the hypervolume performance indicator. Three thresholds have been used to see if different days produce conflicting results. The results showed that NSGA-II works slightly better (1–5% better according to the hypervolume results) than the other MOEAs on the problem for all the scenarios except for 5 days/5,000 iterations where the result is 0.5% worse than the result from SPEA-II. NSGA-II is therefore used in this case study because for all but one of the test scenarios it performs best for this problem. The results are unusual as NSGA-II is often criticised within the literature for its multi-objective optimisation performance with three or more objectives (Elarbi *et al.* 2017). For the MCDA criterion weightings for this case study a literature survey on suggested MCDA frameworks based on the work by Okworia *et al.* (2021) and an evaluation team of wastewater managers at DCWW analysed the needs, goals and limitations of water companies, to decide on the evaluation criteria. The evaluation team decided that priority score was the criteria they wished to prioritise the most. It was therefore decided that the following criterion weightings would be the most appropriate for the company's needs, minimise cost: 0.2, minimise distance travelled: 0.3, maximise priority score: 0.5. A sensitivity analysis was trialled to see whether these weightings would satisfy the company's needs, or whether a better weighting score would be more appropriate. The sensitivity analysis used for the MCDA displays the changes in the rankings for the alternative solutions based on the modifications

of the criteria weights. Ten different scenarios were executed, each of which presented different criteria weight distributions. Scenario 1 (S1) includes the initial weights as provided by the expert opinion of the water company representatives. The decision was aided by using the visualisation of the Pareto-front. The other nine scenarios have modified criteria weights compared with S1. The scenarios are from a 3-day run in Swansea from September 2020, the schedules are formed using NSGA-II with a population of 125, with a mutation rate of 0.3 and a crossover rate of 0.5. After analysis of the test results, the following criteria results (minimise cost: 0.1, minimise distance travelled: 0.3 and maximise priority score: 0.6) boasted the best results with the company monetary results available and are used in the MCDA for the case study.

The methodology was run every 3 days in the case study and relayed to the maintenance teams the night before their shifts, starting from November 1, 2020. Crew shifts vary because blockages occur at all times. The case study is comprised of 105 job locations which are shown in Figure 1. As can be seen from Figure 1, the maintenance locations are distributed spatially throughout the catchment. The homebase for crew 1 is in Neath, at the centre of the catchment, and crew 2 is located in Bridgend, at the southeast of the catchment. Selected pertinent statistics and assumptions of the crew schedule are outlined in Table 2.

DISCUSSION

All 105 PM jobs were completed during the November 2020 trial, plus additional reactive jobs. The schedules all involved a 1-h daily lunch break and a 30-min vehicle and equipment check in the morning. Three of the scheduling days were cut short due to the maintenance teams having to respond to emergency reactive maintenance work. Table 3 displays the breakdown of the jobs completed for the crews in November 2020 and a comparison with the schedule completed by crew 1 in November 2019.

The results from Table 3 show a 26% increase in jobs completed per crew in November 2020 (6.92 jobs per day per crew) compared with the jobs completed in November 2019 (5.50 jobs per day per crew). However, there are several uncertainties with the data comparisons. The details of the hours worked by crew 1 in 2019 are unavailable. It is therefore unclear on how many hours crew 1 worked per day to produce the averaged 5.5 job completion per day. On day 7, crew 1 managed to complete 13 jobs, but it is conceivable that the crew was covering a double shift. Yet it can be concluded that using the scheduling methodology on average 7.18 jobs can be completed per 8.5-h shift as shown in Table 3.

The optimisation performance of the evolutionary approach produced for the first three days of scheduling in November 2020 for crew 1 is shown in Figure 2. The figure shows the differences in the number and optimality of nondominated solutions after 5,000 EA iterations and for 50,000 iterations for the first 3-day simulated run in November 2020.

Figure 2 shows a clear progression towards the utopia point (minimal distance, minimal cost and maximised summed job priority score) by the evolutionary algorithm. The single EA run results in an average of 12 jobs completed over the 3-day period, whereas the 50,000 run results in an average of 15 jobs completed over the 3-day period. The 2019 human-derived schedule is included as a reference (labelled as 'November Result 2019'). The figure clearly shows how the computationally derived solution outperforms the human scheduling for all the objectives. The human-derived schedule has a priority score of 47, the distance travelled for the crew is 116 miles and the cost is at £1608, whereas the average results for the EA was a priority score of 224, distance travelled for the crew was 184 miles and a cost of £1367. Furthermore, the figure demonstrates the range of solutions on the pareto-front provided by the algorithm that are available for the water company to select from if they do not wish to use the MCDA described below. The range of solutions allow the company to fine-tune the schedule to meet priority, cost or routing distance goals for the business. It also allows for the trade-off between these objectives to be visualised and therefore to provide an understanding of the relationship between them.

Table 2 | Crew scheduling assumptions

Number of jobs	105
Number of crews	2
Crew shift hours	8–9.5
Crew daily preparation time	30 min
Crew daily break times	1 h
Source of routing maps	OS open maps

Table 3 | Job completion rate for the 2020 schedules for crew 1 and crew 2 and the November 2019 schedule for crew 1

Day	Crew 1 (2020)	Crew 2 (2020)	Crew 1 (2019)
1	6	7	5
2	6	6	3
3	6	6	4
4	9	5	2
5	8	7	2
6	7	6	11
7	7	7	13
8	8	6	9
9	8	6	1
10	7	8	6
11	7	8	2
12		8	8
13		7	
Total jobs completed	79	87	66
Average jobs completed per day	7.18	6.7	5.5
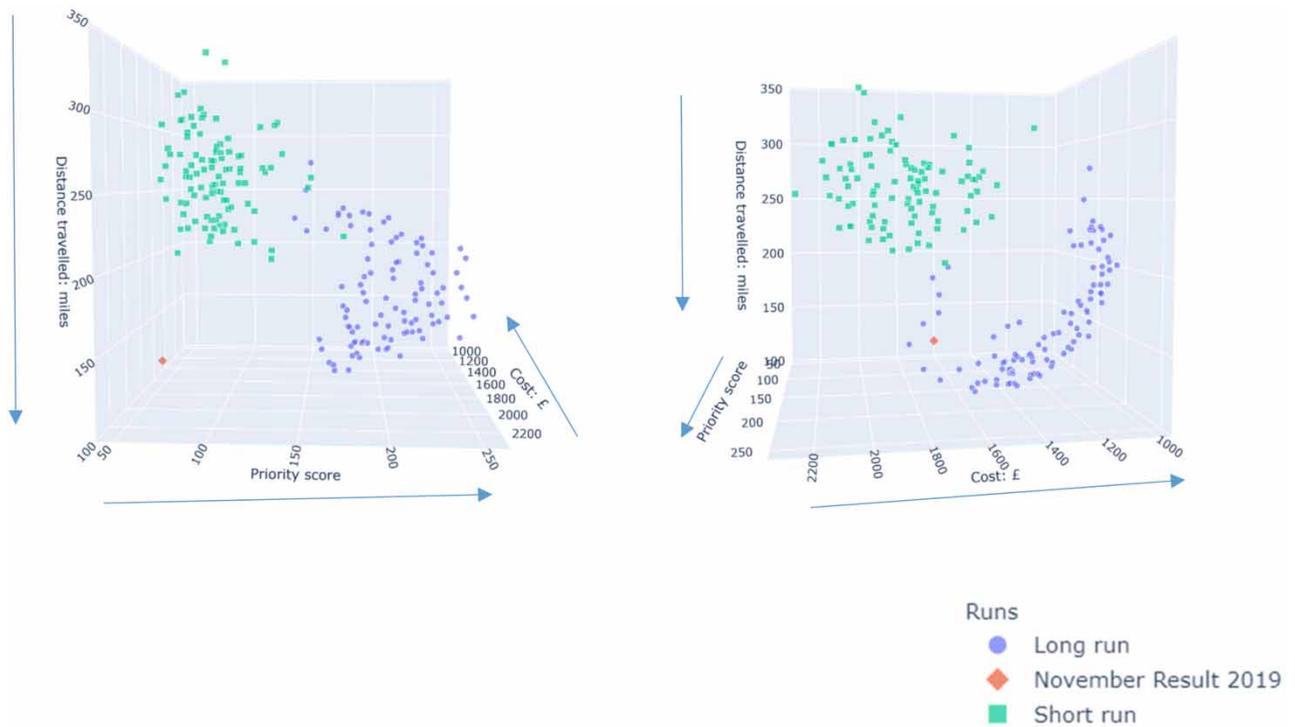
Figure 2 | Pareto-front differences between 5,000 iterations (short run) and 50,000 iterations (long run) and the November 2019 schedule. The preferred direction of objective performance is indicated by arrows on each axis label.

Several characteristic solutions have been identified from Figure 2 and placed individually into Figure 3. These solutions include the highest costing solution, the medium costing solution, the lowest costing solution, the best possible solution according to the MCDA and the historical schedule from November 2019. The solution metrics are shown in Table 4.
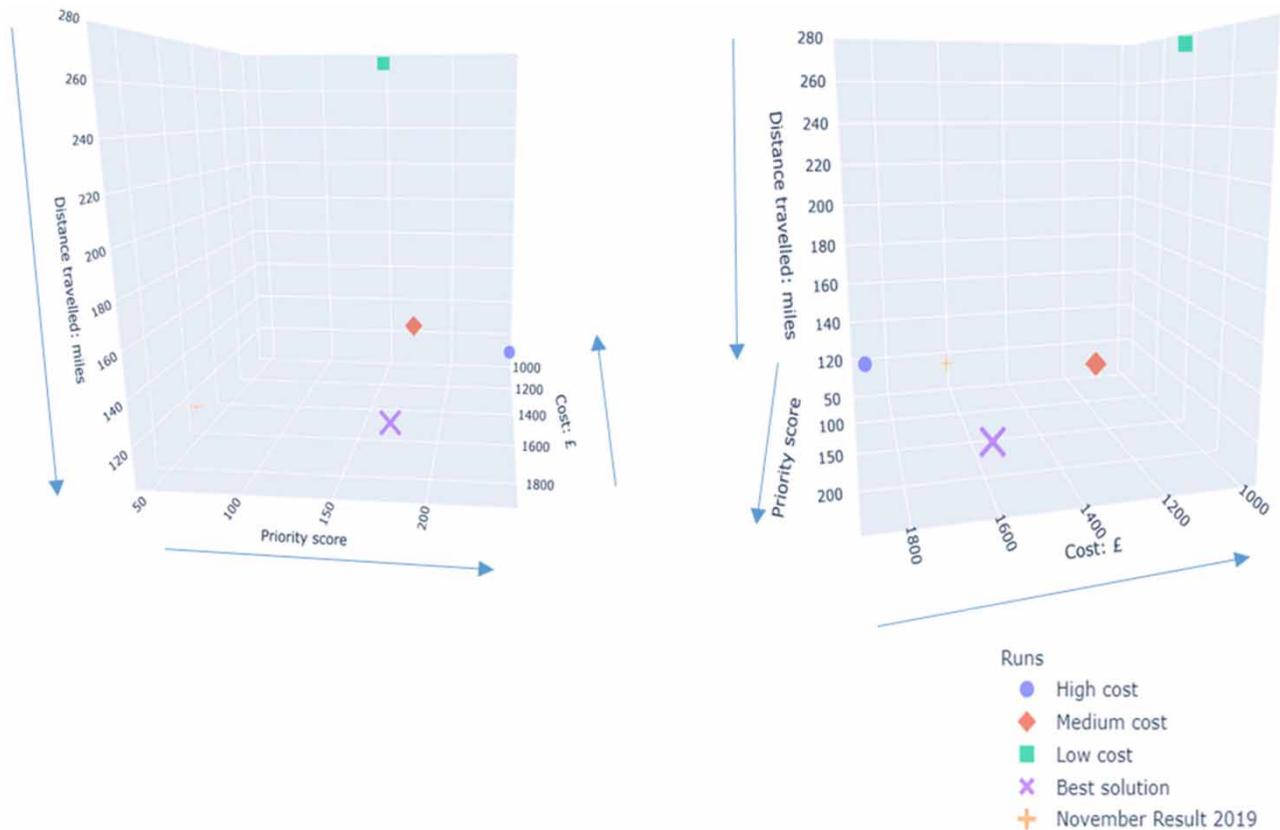
Figure 3 | Characteristic solutions shown on a Pareto-front. Low-, medium- and high-cost solutions are displayed. The best possible solution decided by the MCDA and the historical November 2019 results are also displayed.

Table 4 | Solution specifics from the low-, medium- and high-cost solutions

Solution	Cost (£)	Priority score	Distance travelled (miles)	Jobs completed
Low cost	988.67	148	274.87	14
Medium cost	1,276.46	177	140.26	15
High cost	1,880.77	242	162.78	17

The lower costing solution is 62% cheaper than the most expensive option, however the priority score produces a 48% lower result. In terms of completion rate, the high-cost solution completes two extra jobs per day on average compared with the medium solution and has the quickest completion time.

Using the MCDA results in a selected solution which compromises on the three objectives, to give the best solution for the user. From these results, the MCDA produces a solution with a cost of £1553.61, with a priority score of 169, a distance travelled of 107.1 miles with 17 jobs completed, which is better than the low-cost solutions for all the criterions. The result is 20% more expensive than the medium cost solution and the priority score is 5% worse but the distance travelled is 27% better and the job completion rate is higher with 2 days more completed per day. The result is 19% cheaper than the most expensive solution, but has the same completion rate, with 44% less distance travelled per day. However, the priority score is 36% lower. To compare the historical result produces a solution with a cost of £1608, with a priority score of 47, a distance travelled of 115.8 miles and 13 jobs completed. This is 3% worse in terms of cost, and 8% worse in terms of distance travelled and 113% worse in priority score compared with the MCDA result. The MCDA result is used to analyse the results in the following sections.
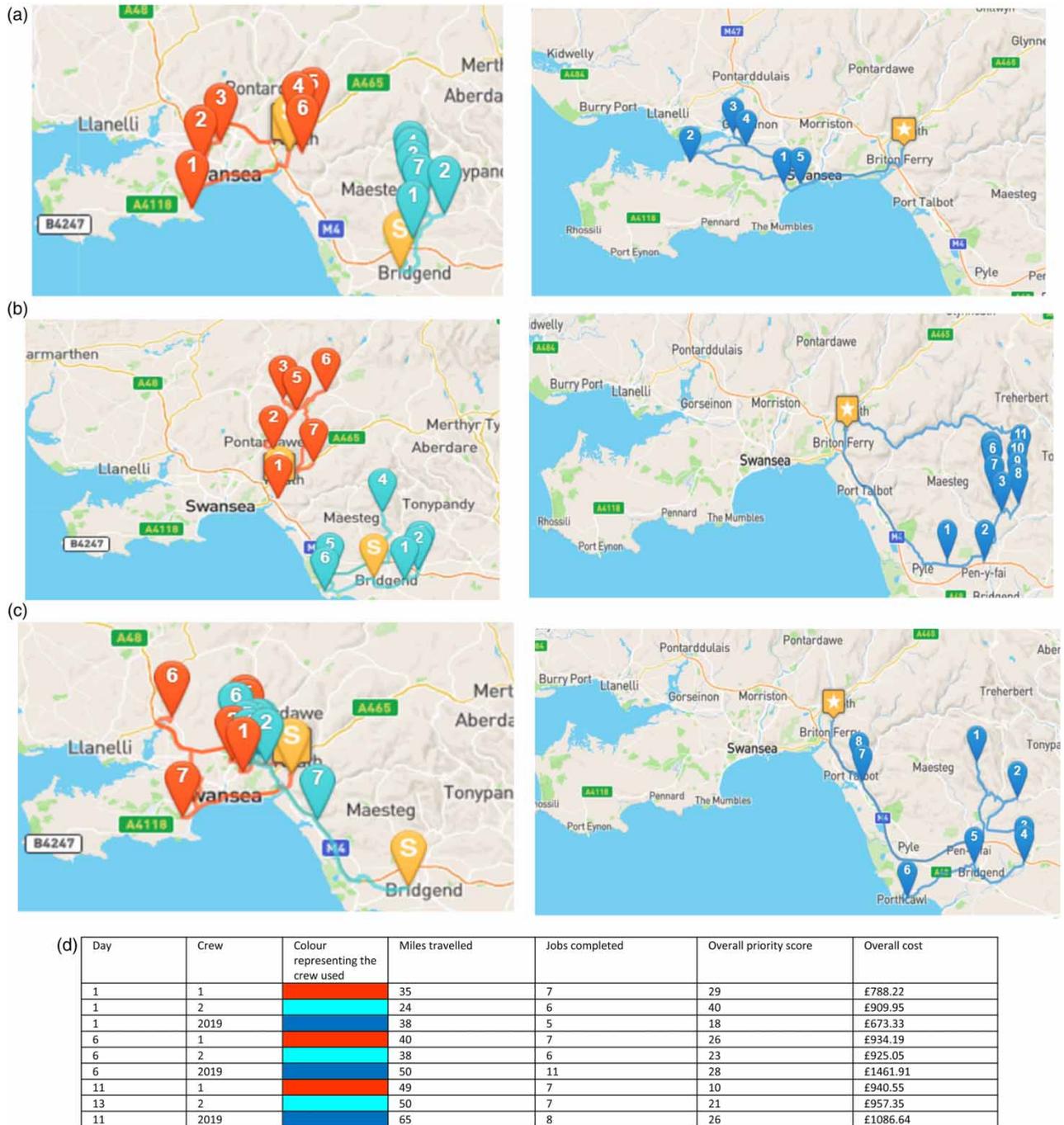
Figure 4 | (a) Day 1 of the schedule, (b) day 6 of the schedule, (c) final day of the schedules and (d) figure key, outlining the intricate details of days 1, 6 and the final day of scheduling for crews 1 and 2 in 2020 and crew 1 in 2019. The yellow icons represent the respective crews' homebase. Please refer to the online version of this paper to see this figure in colour: http://dx.doi.org/10.2166/hydro.2022.149.

In order to understand the routes of the schedules, images of the routes were created and displayed in Figure 4. Figure 4 details the routes for crew 1 in 2020, crew 2 in 2020 and crew 1 in 2019 for their first, sixth and final days of scheduling. These particular days were chosen to show the variation in schedules from the start of the scheduling, the middle of scheduling and the end of scheduling period. By the end of the schedule, the priority score for crew 1 in 2019 is 26 and the crew has travelled 65 miles. On the other hand, crew 1 in 2020 has travelled 49 miles and crew 2 in 2020 has travelled 50 miles to complete jobs with a lower priority score and at lower cost.

It can be inferred from Figure 4(a) and 4(b) that at the start of the month higher priority jobs are completed and the routes are quicker. This clearly demonstrates the efficiency gained from the optimisation-based methodology, because by the end of the month high-priority jobs have been completed. For the first 6 days of scheduling the routes were on average more efficient than in 2019, with on average 7.2 miles travelled less each day for the crews. However, by the end of the month, the crews are both travelling between 49 and 50 miles on the last day to complete the final few jobs. One reason for this could be that crew 2 completed all their jobs on the east of the catchment near their homebase and had to assist in the completion of jobs in west Swansea. This added considerable travel time to their schedules. The Priority scores on the last day of scheduling were recorded as 10 and 21, at the start of the schedule the priority scores are 29 and 40. This shows that the methodology is correctly targeting the higher priority jobs at the start of the month. Figure 4(c), on the other hand, shows that the crew in 2019 were travelling 38 miles at the start of the month, to complete jobs with a priority score of only 18. However, the total day cost is considerably cheaper than both of the 2020 first days of scheduling.

By the middle of the schedule (day 6 of scheduling) the crew in 2019 is completing 11 jobs in one day, however the priority score is only marginally higher than the priority score of crew 1 in 2020. The overall cost is also considerably higher for the 2019 schedule, due to the 50 miles of travel the crew completes on this day. To compare, crews 1 and 2 have only travelled 40 and 38 miles on day 6, respectively. By the end of the schedule, the priority score for crew 1 in 2019 is 26 and the crew is travelling 65 miles. In contrast, the priority score for crew 1 in 2020 was 10 and the crew was travelling 49 miles.

To understand the statistical significance of the distance travelled results from the trials in comparison to the 2019 result, a Wilcoxon signed rank test has been undertaken to test the hypothesis 'The daily travel distances derived by the thesis for crew 1 in 2020 are significantly different to the daily travel distances derived by the human schedule from 2019 for crew 1'. The Wilcoxon test has been chosen because the data to compare in this instance are non-parametric and related, and the Wilcoxon test is a non-parametric statistical hypothesis test which can be used to compare two related samples. The result obtained from the statistical test indicated statistical significance, with the p-value below the 5% threshold at 0.04. It is therefore suggested that the two different scheduling techniques perform differently in regard to minimising the distance travelled for their crews, with the optimisation-based methodology resulting in less travel per crew per day.

Priority score criterion results

To further emphasise the results, the following bar chart of average priority score per job per day is shown in Figure 5. Here, we can see that for the 2020 schedules the average job priority score decreases with time. Yet, the 2019 schedule has the second highest average priority score per day on the final day of scheduling.

To understand the statistical significance of the priority score results from the trials in comparison to the 2019 result, a Wilcoxon signed rank test has been undertaken to test the hypothesis 'The daily priority scores derived by the thesis for
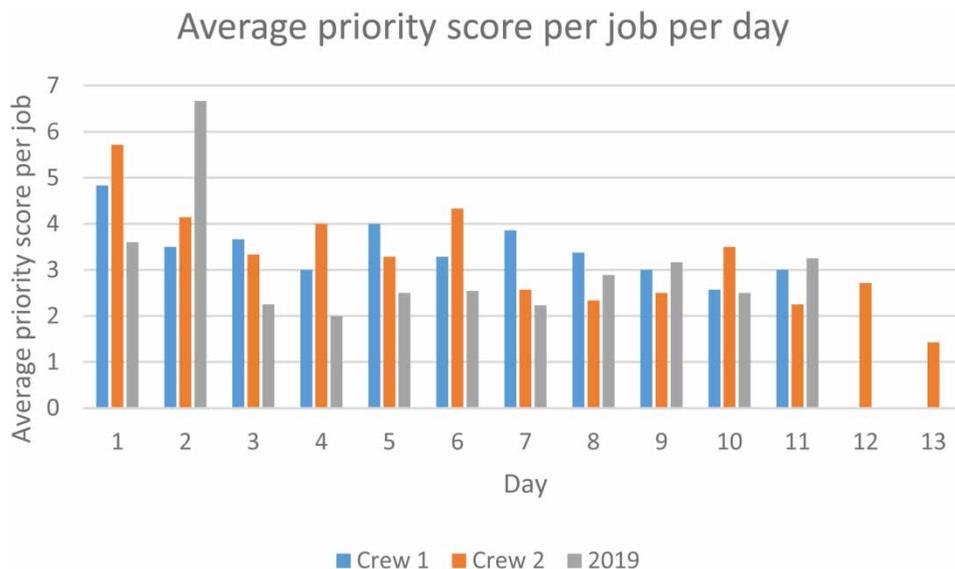

Figure 5 | Average daily priority scores per job for crews 1 and 2 in November 2020 and crew 1 in November 2019.

crew 1 in 2020 are significantly different to the daily priority scores derived by the human schedule from 2019 for crew 1'. The result obtained from the statistical test indicated statistical significance, with the p-value below the 5% threshold at 0.03. It is therefore suggested that the two different scheduling techniques perform differently in regard to maximising the priority score of jobs completed by the crew, with the optimisation-based methodology resulting in a higher total priority score of jobs completed per day.

CONCLUSIONS

This paper presents the novel methodology for optimal scheduling of PM activities in a sewer network covering a larger geographical area. The scheduling problem is formulated as a multi-objective optimisation problem with the following three objectives, namely the minimisation of total maintenance cost, the minimisation of travel times of maintenance teams and the maximisation of the job's priority score, all over a pre-defined scheduling horizon. The optimisation problem is solved using the NSGA-II optimisation method. Unlike other methods which focus on travel time reduction only, the methodology presented here also takes into account the priority of different maintenance jobs while scheduling.

The results obtained from a real-life UK case study led to the following conclusions:

- The proposed scheduling methodology works well resulting in multiple benefits. When compared to exiting, real-life company maintenance schedules, it reduces both related costs and travel and ineffective time/overtime, resulting in improved productivity of maintenance crews (26% improvement in terms of jobs completed in November 2020 vs. November 2019 in the case study shown here). Furthermore, the methodology reduces travel distance, resulting in savings in fuel and carbon emissions. It also increases the priority score of PM jobs at the start of the scheduling period. This leads to high-priority jobs completed on time, and therefore avoided blockages and their negative impacts for customers and the environment.
- The methodology presented is generic, i.e. transferable to scheduling maintenance in other sewer networks and water utilities in general, potentially other infrastructure systems and assets as well. This, of course, would require adapting various maintenance options considered and the corresponding cost and other models to the specific circumstances of these other systems.

Having said the above, this research work, like any other, is not without limitations. The most important limitations are as follows:

- The scheduling methodology focuses on generating the operational schedule for PM and interruptions in scheduled work due to emergency reactive maintenance are not considered.
- Although every effort has been made to ensure high data quality and all inconsistencies are removed or addressed, the extent of error that could remain within the datasets is uncertain. The errors presented could relate to inconsistencies that were not removed during the data cleaning process, as well as rainfall and sewer level recording errors.
- The runtime comparisons between the algorithms are only relative to the computational power of the user and so the study runtimes do not report the actual runtimes. The runtimes could be improved by more efficient coding.

This research was undertaken primarily because the field of automating maintenance scheduling is still being developed, and there are a range of topics that could be considered for additional research projects. This is further reinforced by the rapid advances in genetic algorithms. Based on the findings and limitations of this research, the recommendations for future work are as follows:

- The scheduling methodology could benefit from considering emergency reactive jobs related to unplanned maintenance as well. Given the nature of these jobs, perhaps a robust scheduling methodology for PM under uncertainty could be developed.
- The scheduling methodology could benefit from further testing and validation on additional wastewater systems with even larger and different datasets. Doing so would also enable a better demonstration of the routing and scheduling methodology proposed here. Not only would this improve the confidence in the method but it would allow for better tuning of the internal parameters, enabling better guidelines to be set for selecting key decision thresholds and improve its capabilities. Similarly, the method should be continually compared to developing alternative machine learning techniques.

DATA AVAILABILITY STATEMENT

Data cannot be made publicly available; readers should contact the corresponding author for details.

REFERENCES

Altarabsheh, A. G. 2016 *Managing Urban Wastewater System Using Complex Adaptive System Approach*. Doctoral Dissertation, Purdue University.

Amiri-Ardakani, Y. & Najafzadeh, M. 2021 *Pipe Break Rate Assessment While Considering Physical and Operational Factors: A Methodology Based on Global Positioning System and Data Driven Techniques*.

Bailey, J., Harris, E., Keedwell, E., Djordjevic, S. & Kapelan, Z. 2016 Developing decision tree models to create a predictive blockage likelihood model for real-world wastewater networks. *Procedia Engineering* **154**, 1209–1216.

Berardi, L., Giustolisi, O., Savic, D. A. & Kapelan, Z. 2009 An effective multi-objective approach to prioritisation of sewer pipe inspection. *Water Science and Technology* **60** (4), 841–850.

Butler, D., Digman, C., Makropoulos, C. & Davies, J. W. 2018 *Urban Drainage*. E & FN Spon, New York.

Chen, Y., Cowling, P., Polack, F., Remde, S. & Mourdjis, P. 2017 Dynamic optimisation of preventative and corrective maintenance schedules for a large scale urban drainage system. *European Journal of Operational Research* **257** (2), 494–510.

De, A., Wang, J. & Tiwari, M. K. 2019 Hybridizing basic variable neighborhood search with particle swarm optimization for solving sustainable ship routing and bunker management problem. *IEEE Transactions on Intelligent Transportation Systems* **21** (3), 986–997.

Draude, S., Keedwell, E., Kapelan, Z. & Hiscock, R. 2021 Wastewater systems planned maintenance scheduling using multi-objective optimisation. In: *Proceedings of the Genetic and Evolutionary Computation Conference Companion*. pp. 309–310.

El Amine, M., Pailhes, J. & Perry, N. 2014 Critical review of multi-criteria decision aid methods in conceptual design phases: application to the development of a solar collector structure. *Procedia CIRP* **21**, 497–502.

Elarbi, M., Bechikh, S., Gupta, A., Said, L. B. & Ong, Y. S. 2017 A new decomposition-based NSGA-II for many-objective optimization. *IEEE Transactions on Systems, man, and Cybernetics: Systems* **48** (7), 1191–1210.

Elmasry, M., Zayed, T. & Hawari, A. 2019 Multi-objective optimization model for inspection scheduling of sewer pipelines. *Journal of Construction Engineering and Management* **145** (2), 04018129.

Fontecha, J. E., Akhavan-Tabatabaei, R., Duque, D., Medaglia, A. L., Torres, M. N. & Rodríguez, J. P. 2016 On the preventive management of sediment-related sewer blockages: a combined maintenance and routing optimization approach. *Water Science and Technology* **74** (2), 302–308.

Goodson, J. C., Thomas, B. W. & Ohlmann, J. W. 2016 Restocking-based rollout policies for the vehicle routing problem with stochastic demand and duration limits. *Transportation Science* **50** (2), 591–607.

Hasani, A. & Hosseini, S. M. H. 2020 A bi-objective flexible flow shop scheduling problem with machine-dependent processing stages: trade-off between production costs and energy consumption. *Applied Mathematics and Computation* **386**, 125533.

López-Santana, E., Akhavan-Tabatabaei, R., Dieulle, L., Labadie, N. & Medaglia, A. L. 2016 On the combined maintenance and routing optimization problem. *Reliability Engineering & System Safety* **145**, 199–214.

Marler, R. T. & Arora, J. S. 2010 The weighted sum method for multi-objective optimization: new insights. *Structural and Multidisciplinary Optimization* **41** (6), 853–862.

Mathlouthi, I., Gendreau, M. & Potvin, J. Y. 2021 A metaheuristic based on Tabu search for solving a technician routing and scheduling problem. *Computers & Operations Research* **125**, 105079.

Ngamalieu-Nengoue, U. A., Martínez-Solano, F. J., Iglesias-Rey, P. L. & Mora-Meliá, D. 2019 Multi-objective optimization for urban drainage or sewer networks rehabilitation through pipes substitution and storage tanks installation. *Water* **11** (5), 935.

Ofwat 2017. Available from: https://www.ofwat.gov.uk/publication/sewer-blockages/ (accessed August 2021).

Okworia, E., Pericault, Y., Ugarelli, R., Viklander, M. & Hedström, A. 2021 A data-driven asset management in urban water pipe networks: a proposed conceptual framework. *Journal of Hydroinformatics* **23** (5), 1014–1029.

Petelin, G., Antoniou, M. & Papa, G. 2021 Multi-objective approaches to ground station scheduling for optimization of communication with satellites. *Optimization and Engineering* **2021**, 1–38.

Rodríguez, J. P., McIntyre, N., Díaz-Granados, M. & Maksimović, Č. 2012 A database and model to support proactive management of sediment-related sewer blockages. *Water Research* **46** (15), 4571–4586.

Salazar, J. Z., Reed, P. M., Herman, J. D., Giuliani, M. & Castelletti, A. 2016 A diagnostic assessment of evolutionary algorithms for multi-objective surface water reservoir control. *Advances in Water Resources* **92**, 172–185.

Savic, D. A., Djordjevic, S., Cashman, A. & Saul, A. 2008 A whole-life cost approach to sewerage and potable water system management. In: *Dangerous Pollutants (Xenobiotics) in Urban Water Cycle* (P. Hlavinek, O. Bonacci, J. Marsalek & I. Mahrikova, eds.). Springer, Dordrecht, pp. 3–12.

Schmitz, S., Zipf, A. & Neis, P. 2008 New applications based on collaborative geodata – the case of routing. In: *Proceedings of XXVIII INCA International Congress on Collaborative Mapping and Space Technology*.

Shende, S. & Chau, K. 2019 Design of water distribution systems using an intelligent simple benchmarking algorithm with respect to cost optimization and computational efficiency. *Water Supply* **19** (7), 1892–1898.

Torres Turriago, J. D., Rodriguez, J. P. & Palacio, J. D. 2014 *An Optimization Model For Prioritizing Sewerage Maintenance Scheduling*.

Ward, B. & Savić, D. A. 2012 A multi-objective optimisation model for sewer rehabilitation considering critical risk of failure. *Water Science and Technology* **66** (11), 2410–2417.

Yin, X. 2020 *A Machine Learning-Based Framework for Preventive Maintenance of Sewer Pipe Systems*.