

2018-08-29

# Visualising the Operation of Evolutionary Algorithms Optimising Water Distribution Network Design Problems

Walker, David

<http://hdl.handle.net/10026.1/11833>

---

10.29007/j62b

<https://easychair.org/publications/EPiC/Engineering>

EasyChair

---

*All content in PEARL is protected by copyright law. Author manuscripts are made available in accordance with publisher policies. Please cite only the published version using the details provided on the item record or document. In the absence of an open licence (e.g. Creative Commons), permissions for further reuse of content should be sought from the publisher or author.*

**Theme:** A. Technologies for Water Management and Monitoring

**Subtheme:** A.5. Optimization Techniques and their Application

## Visualising the Operation of Evolutionary Algorithms Optimising Water Distribution Network Design Problems

David J. Walker<sup>1,\*</sup>, Matthew J. Craven<sup>2</sup>

<sup>1</sup> Centre for Water Systems, University of Exeter, Exeter, EX4 4QF, UK.

<sup>2</sup> Centre for Mathematical Sciences, Plymouth University, Drake Circus, Plymouth, PL4 8AA, UK.

\* Corresponding author: D.J.Walker@exeter.ac.uk

### Abstract

Multi-objective evolutionary algorithms (MOEAs) are well known for their ability to optimise the water distribution network design problem. However, their complex nature often restricts their use to algorithm experts. A method is proposed for visualising algorithm performance that will enable an engineer to compare different optimisers and select the best optimisation approach. Results show that the convergence and preservation of diversity can be shown in a simple visualisation that does not rely on in-depth MOEA experience.

**Keywords:** evolutionary computation, optimisation, visualisation, water distribution network design.

### 1 Introduction

Optimisation problems abound in hydroinformatics, with water distribution network (WDN) design being a prominent example. An approach that has been widely used to optimise such problems is evolutionary computation. *Evolutionary algorithms* (EAs) are search algorithms that optimise complex problems based on the principles of natural selection observed in the real world. While methods for generating solutions using an EA are plentiful (see [1] for a recent review), visualisation methods associated with EAs have lagged behind. Some work exists on population visualisation (e.g., [2]), enabling the algorithm user to present a decision maker with a set of solutions from which they can select the final design for implementation, but there has been little work on visualising the optimisation processes used to generate solutions. Such visualisation is an important consideration, enabling the user to understand how effective their evolutionary search is; this paper demonstrates the use of such a visualisation method to visualise the performance of a suite of EAs used to optimise benchmark WDNs.

An EA is a complex system, typically highly sensitive to its parametrisation, and the level of understanding required for a practitioner to use them to their best advantage can be said to be responsible for a gap between academic advances and industrial adoption of methods. This work seeks to narrow that gap by proposing a visualisation method that can reveal how successful the parametrisation of an EA is at solving the problem at hand.

EAs comprise a set of *operators*, which mimic an aspect of natural evolution. The application of all of these operators is governed by parameters, controlling (for example) how much of a solution is mutated; a strong mutation changing most of a solution will encourage more exhaustive search of the solution space, but at the expense of the ability to exploit known good solutions. Parametrisations are problem specific, and thus it is important to have a principled approach for parametrising an algorithm for a new problem. This paper presents the advancement of a recent method [3] for visualising algorithm performance to the analysis of EAs optimising the WDN problem. The method has been applied to continuous test problems and to a basic instance of the WDN problem. Following more extensive analyses of that experiment, this paper presents a more thorough analysis of its use on a more complex WDN design problem - comprising a larger number of pipes and, thus, presenting the EA with a larger search space - which it optimises with EAs of populations of different sizes to compare the performance of the visualisations under different conditions. The paper is structured as follows. Following background material in Section 2, Section 3 outlines the visualisation method. Section 4 describes the experimental setup and Section 5 describes the visualisations resulting from the simulations. Concluding remarks are made in Section 6.

## 2 Background

Designing an effective WDN is a highly combinatoric problem. Hence, an exhaustive search of the search space of possible designs is infeasible. EAs are search algorithms commonly applied to such problems. The algorithms are based on the concepts of reproduction and survival of the fittest found in the natural world. The work presented herein is concerned with one of the best known varieties of EA, the *genetic algorithm* (GA). A GA optimises a problem by maintaining a pool of candidate solutions to the problem; initialised at random, over some number of iterations (called *generations*) the population aims to converge on a good approximation of the optimal solution, or set of solutions. One operator is the *crossover* operator, which generates a new solution from two (or more) parents, while another - the *mutation* operator - introduces variety into the population by making random changes to a child. Having analysed the quality of the child solutions generated using *objective* functions - mathematically defined notions of how suitable a candidate solution is - the *selection* operator is used to identify the strongest solutions and ensure that they persist as parents in the next generation.

Since the first optimisation of a WDN using a GA, the WDN design problem has been investigated in detail by the hydroinformatics community. Many problem formulations have been considered, including *single-objective* (considering a single fitness function) and *multi-objective* (considering a set of objective functions - either independently or using an aggregation function). Common problem objectives are the minimisation of network cost, minimisation of head loss, and the minimisation of water age - a measure of water quality. Research on visualisation within EC generally lags behind work on algorithm development, and within hydroinformatics this is also the case. While there exists some work, it is usually in the area of visualising populations of solutions to aid a decision maker (e.g., [4]). A simple method from the EC visualisation literature is the GAVEL tool, which visualises the way in which evolutionary operators are applied [5]. A method for visualising how individual solutions are evolved over time was applied to WDN design optimisation in [6], illustrating the pipes themselves that had better converged and did not require further mutation over those that did. This work presents a method for visualising populations of solutions to multiple EA runs, aiming to identify good (and bad) parametrisations of an algorithm.

## 3 Visualising Algorithm Performance

The visualisation consists of a circle, in which a small circle represents a solution during the execution of an EA. Figure 1 presents a schematic of the visualisation. The grey circle represents a single solution within one of the

populations of solutions at a specific point during EA execution. Information about the population is conveyed by the distance of the circle from the centre of the visualisation, its angle from the origin ('O' in the figure) and its colour. Two important aspects of algorithm operation are the convergence of the population to the true Pareto front, and the diversity within the population (which ensures that the search properly explores the space of candidate solutions). Convergence is shown using hypervolume [7], which computes the volume of objective space between the population and a predefined reference point. Specifically, the visualisation shows the difference between the hypervolume of one generation and the next; when the search begins, this will be large as the population searches through the space. As the algorithm converges, and exploits known strong solutions, they hypervolume will change less. The colour of a solution shows the mean crowding distance of solutions within the population. A low value on this measure conveys that solutions are close together, implying the search is not properly exploring the space. The radius of the circle from the centre of the visualisation shows an aspect of the optimiser; in this case, the type of mutation has been conveyed (each type of mutation, as discussed later, has its own ring).

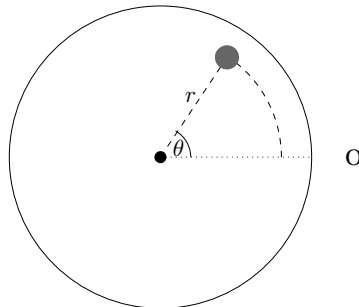


Figure 1. Illustrating the placement of a single solution in the visualisation. A candidate solution is shown by the filled grey circle. The distance  $r$  determines the distance of the solution from the centre of the visualisation, and the angle  $\theta$  governs its angle from the origin [3].

## 4 Experiments

This section describes the experimental setup used to generate data for visualisations. First the optimisation is defined, before the algorithms used to optimise it are introduced. To demonstrate the efficacy of the proposed methods, the New York Tunnels and Hanoi networks are optimised with EAs using both 10- and 100-solution populations. A similar initial experiment demonstrated the potential of this process wherein the execution of an EA with a 10-solution population was demonstrated [3]. Hydraulic information is provided by Epanet [8], and solutions are evaluated under two objective functions: network cost (1) and head deficit (2), both to be minimised.

$$f_1 = \sum_{k=1}^K \left( 1.1 d_k^b \times l_k \right) \quad (1)$$

$$f_2 = \sum_{n=1}^P \left( \left( \hat{h}_n - h_n \right) > 0 \right) \quad (2)$$

We have that  $b$  is a network-specific constant ( $b = 1.24$  for New York Tunnels and  $b = 1.5$  for Hanoi). The length and diameter of pipe  $k$  (out of a total of  $K$  pipes in the WDN) are given by  $l_k$  and  $d_k$ , respectively. The target head for a node is given by  $\hat{h}_n$ , while the actual head is denoted by  $h_n$ . The problem is optimised using a GA, which uses a population of  $N$  solutions (where either  $N = 10$  or  $N = 100$ ). At each generation of the algorithm, members of the current parent population are combined using crossover to produce a population of

candidate child solutions. The child solutions are then mutated with the application of a simple heuristic. To show different algorithm behaviours in the visualisation, the algorithm is run with five different heuristics:

- **Shuffle:** a block of pipe diameters are selected at random and their order is randomly permuted.
- **Swap:** a pair of pipes are selected and their diameters exchanged.
- **Ruin and recreate:** the solution is replaced with a completely new chromosome, generated at random from the available pipe diameters.
- **Change pipe:** the diameter of a randomly selected pipe is changed to another available size at random.
- **Change by one size:** the diameter of a randomly selected pipe is increased or decreased to the next available size.

## 5 Results

Each parametrisation of the algorithm was executed for 5,000 generations. After that time, the estimated Pareto fronts for the New York Tunnel problem were as shown in Figure 2. Figure 3 presents three performance visualisations, showing different points during the execution. Each ring of the visualisation displays a mutation heuristic (1: change by one size; 2: change pipe; 3: ruin and recreate; 4: shuffle; and 5: swap). The left-hand plot shows generation 0. The scattering of solutions in the top section of the plot shows the solutions are randomly generated and have not converged. By the second plot (generation 20), the solutions are progressing into the bottom half of the plot, showing the population has begun to converge. By the final plot, the solutions of the better performing EAs are clustered around the origin. Three of the heuristics (ruin and recreate, shuffle and swap) have not converged; this behaviour is seen in Figure 3, wherein only the ‘change by one size’ and ‘change pipe’ heuristics have converged to the knee of the approximated Pareto front, the region of the front representing the best known trade-off between the two objectives. Colour between these three plots shows the reducing crowding distance (moving from red/green to blue), indicating diversity is reducing as the optimiser converges toward the Pareto front. Using such schematics allows the user to verify the optimiser is converging and that diversity is not reducing too quickly.

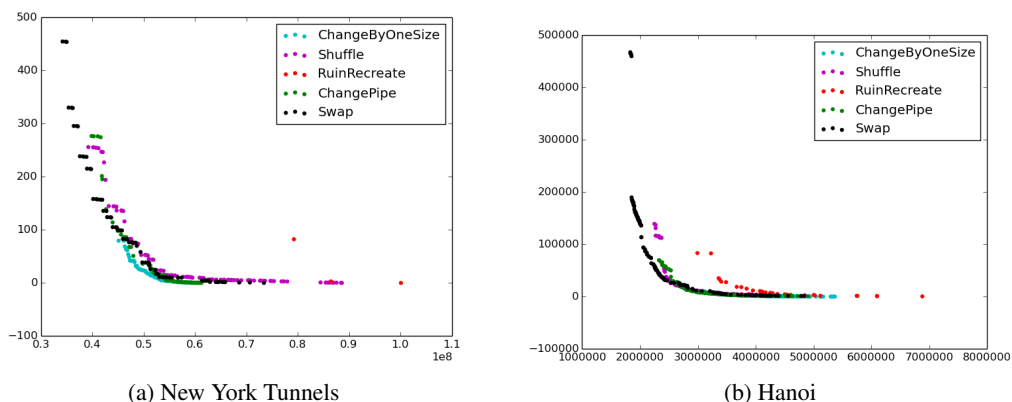


Figure 2. Estimated Pareto fronts for New York Tunnels produced by the optimisation algorithms. The left-hand version shows the final Pareto front approximation obtained over the 5,000 generations the algorithm executed for, while the right-hand panel shows the final search populations. Colour indicates the type of mutation used (cyan - change by one size; green - change pipe; red - ruin and recreate; magenta - shuffle; and black - swap).

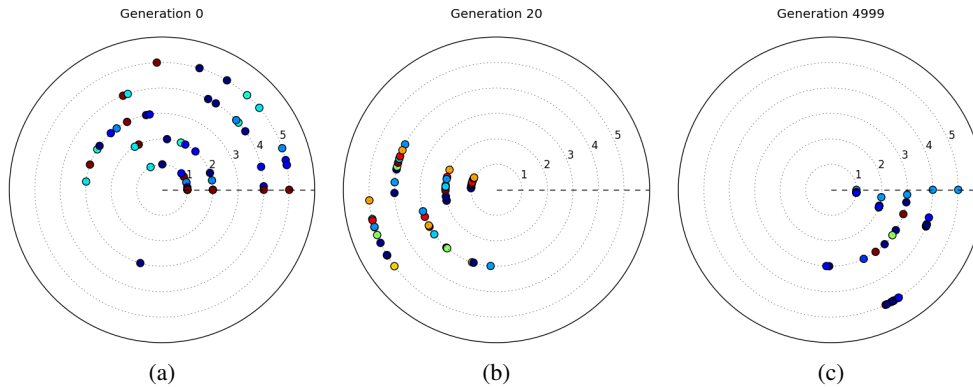


Figure 3. Visualisations of algorithm performance during the execution of the five EAs (using a population of 10 solutions) on the New York Tunnels problem. Colour indicates crowding distance - blue solutions have a low crowding distance, and red solutions a high distance.

Figure 4 shows similar results for the Hanoi problem. As above, this problem features a larger number of pipes for which sizes must be chosen from six available diameters. Here, similar behaviour is shown to that of Figure 3, in that the ‘change by one size’ and ‘change pipe’ heuristics have converged. In this example, the ‘shuffle’ heuristic has also resulted in convergence, while ‘ruin and recreate’ and ‘swap’ have still not converged.

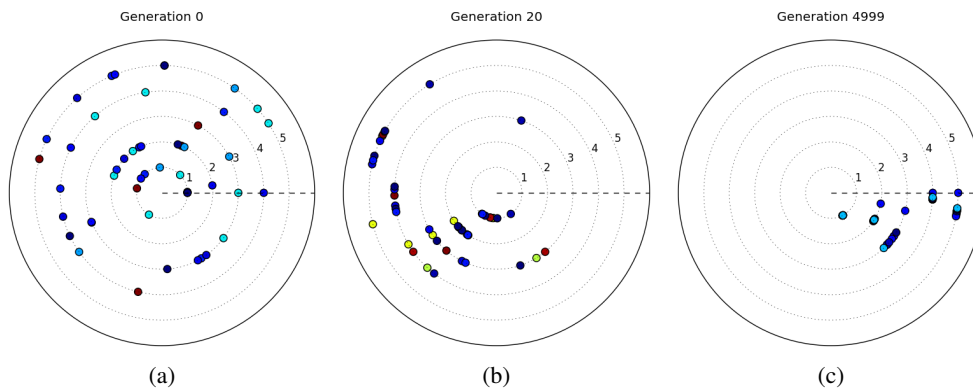


Figure 4. Visualisations of algorithm performance during the execution of the five EAs on the Hanoi problem. Colour indicates crowding distance - blue solutions have low crowding distance, and red solutions high distance.

Examples so far have focussed on EAs using small populations, to aid the description of the visualisation. Often, larger populations are necessary to achieve the full benefit of evolving solutions and increase diversity, so a demonstration of the method with such populations is also called for. Figure 5 shows estimated Pareto fronts for the same five heuristics on New York Tunnels (left) and Hanoi (right). Figure 6 shows corresponding performance visualisations for those EAs. As can be seen, the behaviour of the heuristics is generally consistent with that seen in the 10-solution population, with the ‘change by one size’ and ‘change pipe’ heuristics generally showing more consistently good performance than the other three heuristics. That said, due to the number of solutions in the population, it is now difficult to observe distinct solutions. To this end, the visualisation method was modified to improve this cluttered view of the populations. Rather than presenting the individual solutions, summary statistics on the convergence and diversity properties of the population are shown instead. A line shows the region between the maximum and minimum hypervolume shown by any individual within the population, and the solution with the

median hypervolume is marked by a circle. Colour still indicates diversity, this time showing the mean diversity score (in these experiments, mean crowding distance is shown).

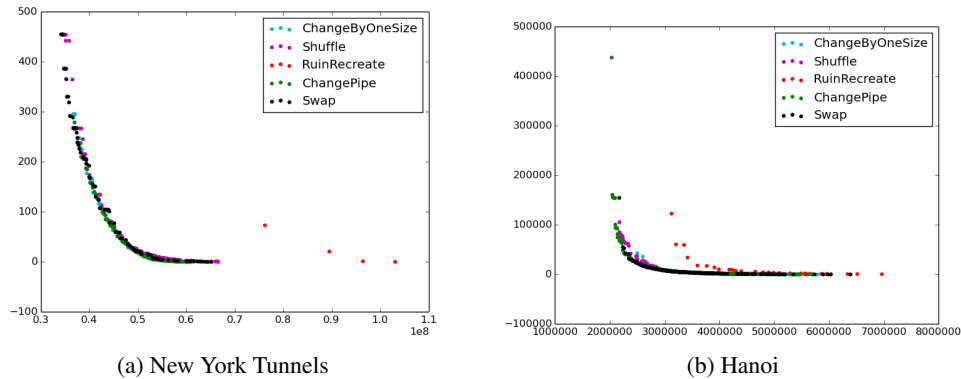


Figure 5. Final Pareto fronts for New York Tunnels (left) and Hanoi (right), produced using EAs with a population of 100 solutions. As can be seen, the coverage of the Pareto front is better in both cases.

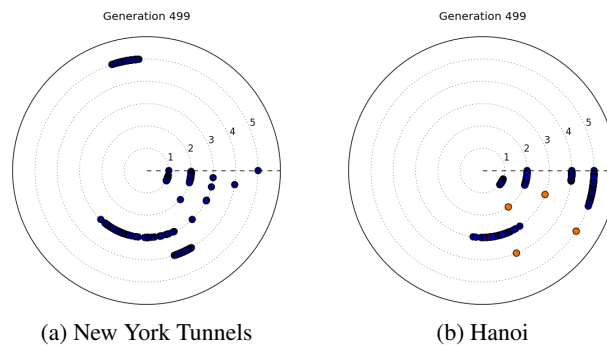


Figure 6. A visualisation of algorithm performance for the final generation of (a) the New York Tunnels and (b) Hanoi optimisations. Individual solution clarity is reduced due to the increased number of solutions. (Note: the EA executed for fewer generations to take account of the increased number of solutions in the population.)

Figure 7 shows the summary statistic version of the Figure 3 visualisations. Rather than a distinct point marking each individual solution, the range of convergence values (hypervolume scores) is shown with an arc, and can be interpreted in the same way as a box and whisker plot would be. Though offering a simpler view, in this small-population example there is not much extra information to be gleaned in terms of convergence over that available in Figure 3. Interestingly, those solutions with a high crowding distance in Figure 3 appear to be outliers, as the mean crowding distance is uniformly low. This indicates that the EA in use is perhaps not searching the space fully, and is unsurprising since the EA in use does not make use of diversity-preserving mechanisms.

The summary statistics plots shown in Figure 8 illustrate algorithm performance for an EA optimising New York Tunnels with a 100-solution population. Unlike a comparison between Figures 3 and 6, comparing these plots with those in Figure 7 shows that there is no extra complexity introduced into the visualisation by increasing the population size. As has been seen before, ‘change by one size’ and ‘change pipe’ are the winning heuristics, on this occasion by a considerable margin. The ‘swap’ heuristic in particular has converged very poorly on this run. It is important to note that the algorithm allows “backward moves”, that is, solutions can be perturbed to a position poorer than their starting point; such a solution would not be placed into the archive, but might persist in the population, as is the case here. The worst solution in the ‘swap’ heuristic ring is worse by the end of the



optimisation than it is at the beginning. Figure 9 shows algorithm performance for the 100-solution population EA optimising the Hanoi network. Again, convergence is clearly visible for ‘change by one pipe’ and ‘change pipe’, with the usual heuristics lagging behind. As with the previous two examples, diversity in the population is low. As can be seen from Figure 5, this has not prevented the solutions from spreading out along the Pareto front.

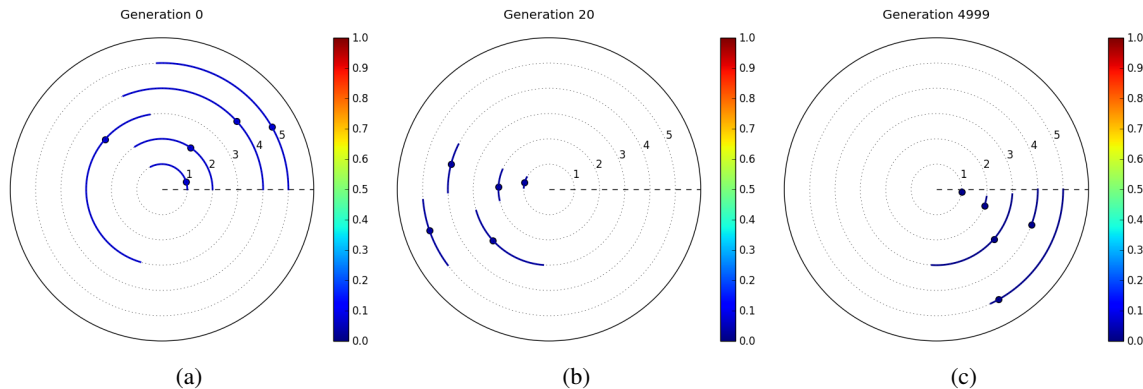


Figure 7. Visualisations of algorithm performance for New York Tunnels, for the 10-solution population EA (corresponding to the results shown in Figure 3), showing summary population statistics.

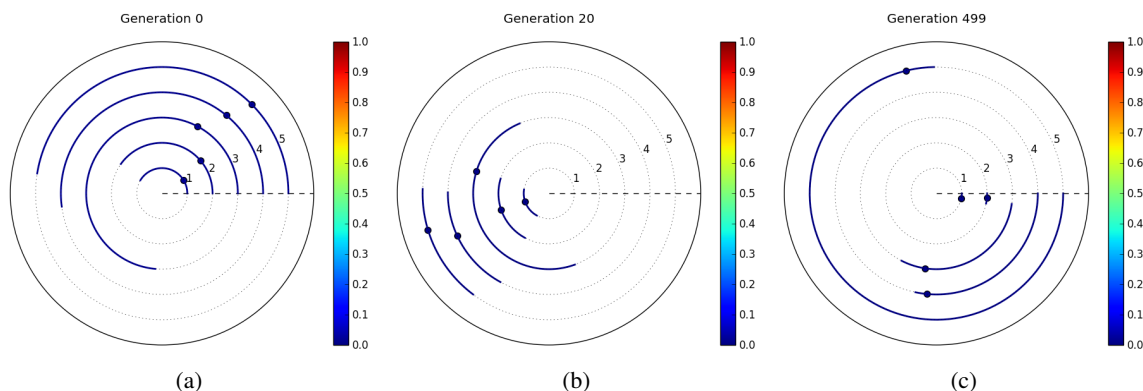


Figure 8. Visualisations of algorithm performance for New York Tunnels, for the 100-solution population EA showing summary population statistics rather than individual solutions.

## 6 Conclusion

This work has demonstrated a visualisation tool for understanding the behaviour of MOEAs tasked with optimising the WDN design problem. Results are shown that indicate the promise of this method; exposing aspects of the algorithms operation visually enables a water engineer to use the algorithm without in-depth knowledge of MOEAs, and this work is a first step towards such a framework. The initial version of the visualisation tool, showing distinct solutions, is able to convey information about the algorithms operation; however, the visualisation becomes cluttered when population size increases. Rather than visualising individual solutions that cannot be seen within the visualisation, summary statistics are shown to offer similar information in a less cluttered fashion. Further ongoing experiments will demonstrate a wider range of algorithm performance measures that will cause the algorithms to behave in different ways. Specific attention is being given to highlighting multi-objective aspects of the EAs, such as highlighting non-dominated population members. Current work is also examining methods for incorporating



decision space into the visualisation, rather than just considering objective space as is the case herein. A promising avenue of investigation in this direction is examining the evaluation of decision space diversity.

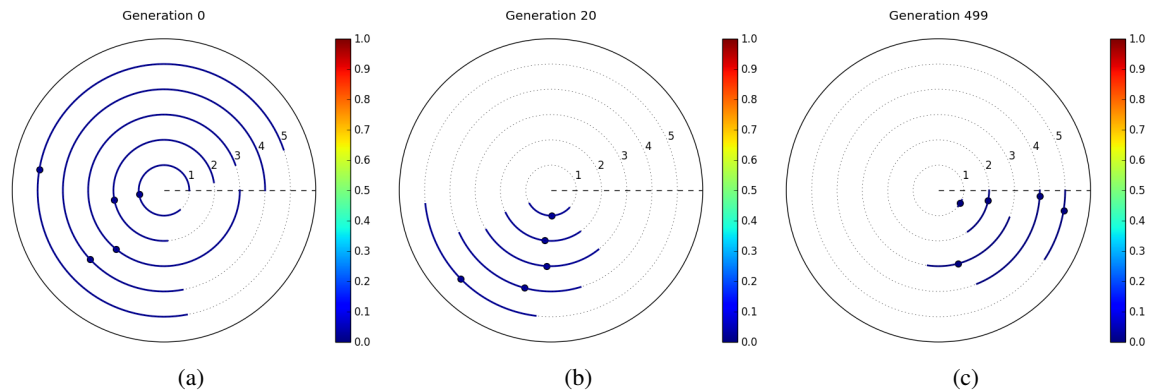


Figure 9. Visualisations of algorithm performance for Hanoi, for the 100-solution population EA, showing population statistics.

## 7 Acknowledgements

David Walker was supported by EPSRC grant EP/P009441/.

## References

- [1] Wang Q, Guidolin M, Savić D, Kapelan Z (2014) Two-Objective Design of Benchmark Problems of a Water Distribution System via MOEAs: Towards the Best Known Approximation of the True Pareto Front. *J Water Resources Planning and Management* 141.
- [2] Fu G, Kapelan Z, Kasprzyk JR, Reed P Optimal Design of Water Distribution Systems Using Many-objective Visual Analytics. *J Water Resources Planning and Management* 139: 624-633.
- [3] Walker DJ, Craven MJ (2018) Toward the online visualisation of algorithm performance for parameter selection. In: *International Conference on the Applications of Evolutionary Computation*. Springer, pp. 547–560.
- [4] Walker DJ, Everson RM, Fieldsend JE (2013) Visualising mutually non-dominating solution sets in many-objective optimization. *IEEE Transactions on Evolutionary Computation* 17.
- [5] Hart E, Ross P (2001) GAVEL - a new tool for genetic algorithm visualization. *IEEE Transactions on Evolutionary Computation* 5: 335-348.
- [6] Keedwell E, Johns M, Savić D (2015) Spatial and temporal visualisation of evolutionary algorithm decisions in water distribution network optimisation. In: *Proceedings of Visualisation in Genetic and Evolutionary Computation (VizGEC 2015) held at GECCO 2015*. *GECCO Companion '15*, pp. 941–948.
- [7] Fleischer M (2003) The Measure of Pareto Optima. Applications to Multi-objective Metaheuristics. In: *Evolutionary Multi-Criterion Optimization*. Second International Conference, EMO 2003. Springer, pp. 519–533.
- [8] Rossman LA (2000) EPANET 2 Users Manual.