Sayers, William ORCID: 0000-0003-1677-4409 (2021) Artificial Intelligence and Optimisation Techniques for Risk Reduction in Civil Engineering. In: Handbook of Research on Digital Transformation, Industry Use Cases, and the Impact of Disruptive Technologies. Advances in E-Business Research Book Series . IGI Global, Hershey PA, pp. 55-72. ISBN 9781799877127

Official URL: https://www.igi-global.com/chapter/artificial-intelligence-and-optimisation-techniques-for-risk-reduction-in-civil-engineering/288642
DOI: http://dx.doi.org/10.4018/978-1-7998-7712-7.ch004
EPrint URI: https://eprints.glos.ac.uk/id/eprint/10495

Chapter 4
Artificial Intelligence and Optimisation Techniques for Risk Reduction in Civil Engineering

William K. P. Sayers
University of Gloucestershire, UK

ABSTRACT

Artificial intelligence techniques are at the centre of a major shift in business today. They have a very broad array of applications within businesses, including that of optimisation for risk reduction in civil engineering projects. This is an active area of research, which has started to see real-world applications over the last few decades. It is still hindered by the extreme complexity of civil engineering problems and the computing power necessary to tackle these, but the economic and other benefits of these emerging technologies are too important to ignore. With that in mind, this chapter reviews the current state of research and real-world practice of optimisation techniques and artificial intelligence in risk reduction in this field. It also examines related promising techniques and their future potential.

INTRODUCTION

Civil engineering comprises the design, construction, operation, and maintenance of buildings and infrastructures including a variety of works such as residence, bridges, and roads (Zavala et al., 2014). Since the second World War, with the rapid advances made in computational methods, optimisation techniques based on mathematical programming have been increasingly deployed in the field of civil engineering (Topping, 1983). Optimisation refers to acquiring the best outcome under specific conditions (Rajput & Datta, 2019), and optimization problems are evident in many disciplines, including operations research, computing, engineering and economics. Optimisation techniques consist of a powerful set of tools that can be deployed to help the effective management of a company's resources, and can be seen as an artificial intelligence (AI) tool. Kolter and Procaccia (2017) noted that "one of the most significant

DOI: 10.4018/978-1-7998-7712-7.ch004

trends in AI in the past 15 years has been the integration of optimization methods throughout the field" (p.5). More specifically, in the field of civil engineering, optimisation can be executed in each step of a project life cycle, from design and construction to operation and maintenance, but can also be applied more generally to risk estimation and reduction (Dede et al., 2019; Mei & Wang, 2021).

Optimisation algorithms can be used to identify solutions to correctly formulated optimisation problems. This requires the formulation of an equation, or objective function, which calculates a measure of performance. Variables in the problem being optimised can then be represented as combinations of parameters for this function. For example, a problem such as drainage network optimisation can be represented as an optimisation problem by the use of an equation for calculating the cost of making changes to an existing drainage network. The parameters for this equation involve a mobilization cost (M) representing an initial cost of making change. Additionally, a combination of a cost for each pipe altered (I), the length of pipe requiring alteration (L) and the cost of purchasing pipes of a particular cross-section (c). Finally, storage tank alteration costs (S), area of the storage tank (a) and a base cost (b) as shown in equation 1 (Sayers, 2015).

$$Cost = M + \left(\sum_{i=0}^{n} I \times L_i \times c_i\right) + \left(\sum_{j=0}^{n} S \times a_j + b\right) + \left(\sum_{k=0}^{n} o_k\right) \tag{1}$$

Importantly for most algorithms to function well, this objective function must return values which are differentiable and represent the problem space well. This means that identifying a suitable objective function is in many cases one of the most complex parts of formulating a problem as an optimisation problem. Making this more challenging, is the fact that in most optimisation algorithms this objective function will be called a very high number of times, as different solutions are tested through the optimisation process. This is because of an effect known as "combinatorial explosion", where the number of parameters has an polynomial effect upon the size of the search space (dramatically increasing it in response to a small increase in parameters, or parameter values). Because of this, if an objective function takes even a few seconds to complete, this will result in extremely time-consuming algorithm runs when trying to be sure to undertake a reasonable exploration of the search space. Unfortunately, the majority of civil engineering optimisation problems are extremely complex and sometimes multi-objective optimisation problems, meaning they are generally computationally intractable, as exhaustive, or even mostly exhaustive, searches. Because of this, optimisation algorithms for these applications are generally non-deterministic, and are based around the use of heuristics. These heuristics allow them to be applied to problems where an exhaustive search for solutions with a computer is impractical, and still identify a reasonable solution. If this problem is even more pressing, then meta-heuristics and artificial intelligence techniques can be used to further alleviate this, such as machine learning regressors trained to approximate the above equation without the need to complete the processing associated with a full evaluation of this equation.

The objectives of this chapter are to first present a review of academic literature on optimisation techniques, including an outline of optimisation techniques and how they work and may be applied to engineering related problems. This is followed by an assessment of some examples of risk reduction and risk assessment optimisation in civil engineering, and a discussion of emerging issues. Finally, the concluding section offers some reflection on research directions, prospects for the field, and the practicalities of applications.

OPTIMISATION TECHNIQUES AND META-HEURISTICS

Optimisation algorithms are very varied, due to the extremely varied nature of the problems they are applied too. Small differences in algorithms can mean that they are more suitable for the nuances of different fields or particular specialisms, and therefore many different optimisation algorithms are simultaneously viable in their own right, when applied to particular problems (Fan et al., 2020; Weerasuriya et al., 2021). Because of this, there are several sub-categories of the broader category of optimisation algorithm, including single-objective and multiple objective algorithms, examples of which are discussed below.

Meta-Heuristic Artificial Intelligence Techniques

Optimisation and machine learning are closely intertwined throughout their history. Many of the machine learning training algorithms commonly in use today are gradient descent optimisers at their core (Baldi, 1995; Choromanska et al., 2015). Machine learning techniques are also used within optimisation algorithms (di Pierro et al., 2009; Jourdan et al., 2005a, 2005b, 2004; Sayers, 2015) often as either approximators of complex objective functions or as filtering mechanisms to identify promising candidates prior to the full objective function being run to evaluate the candidates.

Single-Objective Optimisation

One sub-group of optimisation algorithms is those specialised for single-objective optimisation such as linear programming (Schrijver, 1998), integer programming (Schrijver, 1998), non-linear programming (Bertsekas, 1999), gradient descent based algorithms (Baldi, 1995; Burges et al., 2005), evolutionary strategies (Beyer & Schwefel, 2002; Rechenberg, 1965, 1973; Schwefel, 1965, 1970, 1975, 1977). These algorithms specialise in finding a single desirable item or state, amongst a superset of items or states that meet any constraints.

Single objective algorithms can also be split into two groups, deterministic and non-deterministic. Deterministic encompasses algorithms such as gradient descent (Baldi, 1995; Burges et al., 2005), A* search (Liu & Gong, 2011), or TABU search (Gendreau & Potvin, 2005; Glover & Laguna, 1997; Glover & Taillard, 1993; Soriano & Gendreau, 1996). Deterministic algorithms are simple, efficient and predictable methods for solving fairly simple problems such as pathfinding. However, as the complexity of the problem to be solved increases, deterministic algorithms become less suitable due to the computational power that would generally be required to solve a complex problem well in a deterministic manner. Non-deterministic algorithms encompass algorithms such as genetic algorithms (Goldberg, 1989; Holland, 1962, 1975; Jong, 1975), simulated annealing (Kirkpatrick et al., 1983), and ant-colony optimisation (Dorigo, 1992; Dorigo & Blum, 2005; Stũtzle & Dorigo, 2002). These non-deterministic algorithms are inefficient for use on simple problems, but become more suitable as the complexity of the problem increases, due to their use of heuristic techniques, which can allow them to reduce the area of the search space that needs to be explored to identify a good solution. Civil engineering and flood risk problems generally fall into the category of non-linear, NP-hard problems which deterministic algorithms generally may not solve well due to lacking this heuristic driven ability and being overwhelmed by the sheer volume of the search space. For this reason, this chapter will review some of the common non-deterministic algorithms, and leave aside consideration of deterministic algorithms.

Genetic Algorithms

Genetic algorithms are a form of non-deterministic algorithm, which were inspired by Darwin's theory of evolution, in particular the idea that in a competitive environment, organisms with useful traits will supplant or exist alongside preexisting organisms. Genetic algorithms use a population-based approach and are generally highly suitable for non-linear non-convex, multi-model and discrete problems with which deterministic algorithms may struggle to converge to a reasonable solution (Nicklow et al., 2010). A genetic algorithm can be broken into four stages, generation, selection, crossover and mutation, and also requires a well-formulated objective (or fitness) function which will identify how good a particular population member is at solving the problem in question (Holland, 1962, 1975; Jong, 1975). First an initial population of random problem solutions is generated and scored via this function. Then selection takes place, and members of the population are chosen to be "parents" of new "offspring". Various methods are used for selection such as stochastic universal sampling (Ghimire et al., 2013), or tournament selection (Miller & Shaw, 1996; Nicklow et al., 2010; Sayers, 2015). These selected "parent" individuals are then combined via a "crossover" technique (Deb & Agrawal, 1994; Gwiazda, 2006). They are then modified with some small random chance for the "mutation" stage and different techniques and constraints can be placed on this as appropriate (Nicklow et al., 2010; Sayers, 2015). These "child" individuals then replace the "parent" population, and the process starts again at selection. Genetic algorithms were first developed in the 1960s, and gained in traction through the 1970s, particularly with the publication of Holland's (1975) "Adaptation in Natural and Artificial Systems" and Kenneth De Jong's (Jong, 1975) "An analysis of the behaviour of a class of genetic adaptive systems." Because of their population based nature, they are inherently suitable for parallelisation, and can be robust in terms of avoiding local optima within the search space.

Ant-Colony Optimisation

Ant-colony optimisation is a more recent algorithm than the other single-objective optimisation algorithms discussed in this chapter, being developed in the early 1990s as an optimisation algorithm for combinatorial optimisation (Dorigo, 1992; Dorigo et al., 1991; Dorigo & Stũtzle, 2002). Ant-colony optimisation is inspired by the methods used by colonies of ants to locate food in the wild. Ant workers will be sent out to randomly search for food, leaving pheromone trails behind them. Where food is found, they will then follow their own trail back to the nest, reinforcing that trail as they go. Other workers coming across a pheromone trail are more likely to follow it, and reinforce it themselves, the stronger it is. If multiple paths exist to the food, then the shorter paths will over time be more reinforced, leading to a gradual preference for the shortest route. This is a very effective self-organising approach which ant-colony optimisation seeks to emulate (Dorigo & Stützle, 2010). It does this by emulating several artificial ants who plot a path through a problem space represented as a plot of nodes (solutions to the problem) and edges (paths between solutions), over a number of iterations. The ants are precluded from revisiting nodes and new steps of their path are calculated at each iteration by a stochastic approach using a "pheromone" hyper parameter associated with the edges that the "ants" can read and modify. Similarly to genetic algorithms, many variants of ant-colony optimisation exist (Dorigo & Stützle, 2010; Mohan & Baskaran, 2012). Ant-colony optimisation is a more complex algorithm than simulated annealing, and can often be more complex than genetic algorithm based approaches. It is very effective as an optimisation approach however, and shares the strength of genetic algorithms of being potentially parallelisable and robust to local optima, as well as having a very good capability for online execution, and a capability to effectively cope with live modifications to the problem being optimised.

Simulated Annealing

The simulated annealing algorithm (Kirkpatrick et al., 1983) is inspired by the metalworking process of the same name. Annealing in metal-working is the process of heating and cooling metal to achieve desired properties of the material. Simulated annealing involves having a "temperature" hyper-parameter, tracked by the algorithm, which starts at a high value and gradually reduces as the algorithm proceeds. Normally a halting condition of a minimum temperature is set as another hyper-parameter. Initially, a random solution to the problem in question is generated, and a score obtained from the objective function which represents the "energy" of that state. At each cycle of the algorithm, a new state is generated from the old, which is then also evaluated. If the new state has a lower energy than the current state, it replaces the current. If the new state has a higher energy than the current state, the current state may be replaced with a particular probability linked to the current temperature and difference in energy states. The chance of inferior solutions replacing the main solution is reduced by this mechanism as the temperature lowers (Smith & Savić, 2006). A proof exists that with a sufficiently drawn out cooling schedule, the simulated annealing algorithm will always converge to the best possible solution (Geman & Geman, 1984). However, in order for simulated annealing to be useful as a non-deterministic optimisation algorithm, a much faster cooling schedule is necessary to finish the algorithm within a reasonable time frame. Simulated annealing is a very effective and simple algorithm that performs well across a very broad range of cases.

Multiple-Objective Optimisation

In addition to single-objective algorithms, multiple objective algorithms exist as a separate category. The development of these algorithms is challenging, necessitating a capacity to incorporate multiple objective functions. A way to weigh relative objective fitness between objectives is necessary, and although initial approaches often used a weights and sums approach (Schaffer, 1984), this is more recently generally done via a Pareto front based approach to optimisation (Coello Coello, 1999, 2005). Rather than generating one solution, these algorithms therefore usually generate multiple solutions which are all equivalent in fitness, but which vary in terms of how fit they are in each specific objective. These multiple solutions may be distinguished between manually, although decision support tools exist which can help to make the decision by way of, for example, sliders or comparisons (Kapelan et al., 2005). This set of solutions is known as the "Pareto-set", "non-inferior" or "non-dominated" set, which contains only Pareto optimal solutions. A solution is considered Pareto optimal if it is not possible to improve its fitness for any individual objective, without decreasing its fitness for another objective (Coello Coello, 1999, 2005; Deb et al., 2002). A number of multiple objective optimisation algorithms exist and due to the inherently population based approach of genetics inspired techniques, and the maturity of genetic algorithms, many are genetic algorithm inspired. Here three such algorithms are reviewed: The Non-Dominated Sorting Genetic Algorithm (NSGA) II/III, which is one of the most commonly used multiple-objective algorithm techniques; the Population-based Ant-Colony Optimisation (P-ACO), which is an ant-colony optimisation based approach to multiple objective optimisation; and Multiple-Objective Simulated Annealing techniques.

This section will review NSGA-II, which is one of the most commonly used multiple objective genetic algorithm techniques, P-ACO, which is an ant-colony optimisation based approach to multiple objective optimisation, and multiple-objective simulated annealing techniques.

Non-Dominated Sorting Genetic Algorithm II/III

The non-dominated sorting genetic algorithm version two (NSGA-II) is a multi-objective genetic algorithm which works to create an approximate Pareto-front using a population-based approach where any number of objectives are evaluated and optimised against in an iterative manner (Deb et al., 2002; Fan et al., 2020). A subsequent version of the algorithm, NSGA-III has been published, but is a relatively minor modification of the NSGA-II algorithm and has not been so widely applied as of yet (Deb & Jain, 2013). The main modification is in terms of the selection criteria and the calculation of crowding distance which is accomplished with the use of reference points, rather than the distance method outlined below. NSGA-II functions in a manner conceptually similar to a genetic algorithm, with additional complexities in order to deal with Pareto optimality and optimising an estimated non-dominated set against a series of Pareto sets.

Initially, a population of solutions is generated randomly. This population is sorted based on non-domination (Deb et al., 2002). This is accomplished by calculating two values per solution, *np* the number of solutions which dominate the current solution, and *Sp* the set of solutions dominated by this current solution. The first non-dominated set is thus all solutions with a *np* of zero. Every solution dominated by this set is then checked, and its *np* reduced by one. If any of those solutions reach a *np* of zero through this, they become part of the second non-dominated set. This process is continued until all solutions have been ranked into non-dominated sets. Each solution is then considered to be as fit as its non-dominated rank. Standard genetic algorithm operators of selection, crossover and mutation can then be used to create a "child" population. This "child" population is then combined with the "parent" population to create a population of size *2n* and ranked into non-dominated sets as previously described. To construct a population of size *n* from this, as many full non-dominated sets as will fit are combined into a population. Then remaining spaces are filled with members of the remaining set or sets, using a "crowded comparison" operator to distinguish them. This crowded comparison operator prefers solutions first by non-dominated set, and then by average euclidean distance to other solutions within the same front (preferring higher average distances). By using this approach, diversity is encouraged in the population in terms of parameter values and premature convergence is discouraged. The algorithm can then continue having fully generated a new population of size *n*, utilising the crowded comparison operator throughout for selection.

As can be seen, NSGA-II/III are complex algorithms compared to single-objective algorithms, but they are extremely effective, demonstrating very strong results against many optimisation problems both practical and theoretical (Behzadian et al., 2009; Deb et al., 2000, 2002; Fan et al., 2020; Jourdan et al., 2004; Sayers, 2015; Sayers et al., 2019; Woodward, 2012; Woodward et al., 2013, 2014).

Population-Based Ant-Colony Optimisation

Ant-colony optimisation, as described previously is a successful and effective single-objective optimisation algorithm. Because of this, and because of its strengths in online optimisation and parallelisation, significant research has been undertaken in applying ant-colony optimisation to multiple-objective optimisation problems. One of the primary techniques is the Population-based Ant-Colony Optimisation (P-ACO) technique (Doerner et al., 2003). The P-ACO algorithm consists of two stages, construction and evaluation. The algorithm also makes use of a "pheromone matrix" and a "solution archive". The solution archive is initialised as empty, and at each iteration of the algorithm, a solution is added and the pheromone matrix is updated in light of this. After a maximum number of solutions is reached, new solutions entering the archive replace existing solutions by a strategy specified by a pre-set hyper-parameter. The most common strategies are age-based (replacing the oldest), quality based (replacing the least fit), or elitist (ensuring that the best solution found at any point persists) (Fan et al., 2020).

P-ACO has not been used as widely as NSGA-II as of yet, but does show promise as a robust multi-objective optimisation algorithm that could have very specific strengths making it particularly applicable to certain problems (Fan et al., 2020; Jing et al., 2018).

Multi-Objective Simulated Annealing

Simulated annealing as a single-objective optimisation algorithm is, relatively speaking, simple and efficient, as well as offering very robust performance. It is therefore highly attractive to apply the same search methodology to multiple-objective problem sets (Amine, 2019). Most multi-objective simulated annealing approaches have built upon work by Serafini (1994). Building on this has led to suggestions for different probabilistic acceptance heuristics based upon acceptance of non-dominated solutions, including accepting any non-dominated solution, or acceptance of any non-dominating or non-comparable solutions, driving further diversity within the search (Amine, 2019). A combination of these two approaches has also been suggested. An approach distinct from this, known as multi-objective simulated annealing (MOSA) utilises simulated annealing as a search method for exploring the solution space of the given optimisation problem, and builds an estimated Pareto front by archiving non-dominated solutions as the algorithm progresses (Ulungu et al., 1999). Another approach is Dominance Based Simulated Annealing which uses the relative dominance of a solution as the energy value. This approach has been successfully applied to real world problems and proven to be effective (Smith et al., 2008; Smith & Savić, 2006).

Much like multi-objective ant-colony optimisation, multi-objective simulated annealing has not seen the broad range of applications that multi-objective genetic algorithm based approaches have due to its lacking their maturity. But it demonstrates promise in the applications so far, suggesting that further work and applications of these techniques could be fruitful.

Meta-Heuristic Optimisation

As can be inferred from the brief synopses of these algorithms, even single-objective optimisation can require many thousands of executions of its objective function, and when you increase the complexity of the algorithm by making it multi-objective, this is amplified even further. Civil engineering related problems often lend themselves to multiple-objective optimisations, thus requiring these more complex algorithms (Behzadian et al., 2009; Boelee & Kellagher, 2015; Sayers, 2015; Sayers et al., 2019; Woodward et al., 2014). Additionally, many civil engineering risk related problems are best represented (for the purposes of an objective function) through complex modelling of some kind, which may even require the execution of simulations of physical infrastructure (Cesses & Kellagher, 2009; Kellagher et al., 2008; Kellagher & Cesses, 2009). As a result of these factors, optimisation, particularly multiple-objective optimisation, of civil engineering problems can be a completely intractable computing demand. This is as a result of combinatorial explosion of the potential number of executions of the objective function. If an algorithm were to undertake ten thousand iterations of a problem with a population of one hundred individuals in our algorithm (fairly conservative numbers), that is (depending on algorithm) potentially a million plus executions of our objective function. If this objective function takes ten minutes to execute, our algorithm will take somewhere around nineteen years to execute fully. These are very rough numbers and quite conservative figures to illustrate the point that with complex objective functions like civil engineering models, even complex and very modern algorithms can struggle due to the sheer computational load.

This challenge is one that meta-heuristic optimisation seeks to solve by incorporating additional heuristics within the already heuristic optimisation algorithms (Behzadian et al., 2009; Jourdan et al., 2005a, 2005b; Sayers et al., 2019; Wojtusiak & Michalski, 2006). This has the effect of further reducing the search space which must be explored, or the number of objective function executions that are necessary. This approach was first suggested by Blanning (1975) in his paper "The Construction and Implementation of Metamodels." These additional heuristics are often machine learning models which attempt to learn some representation of the objective function, that can then be incorporated in order to either estimate scores at points in the algorithm, saving an objective function run, or to filter potential solutions in some manner, minimising the number of solutions that must be evaluated at each iteration.

For example, the learnable evolution model approach (LEM) uses machine learning models which learn why some solutions are superior to others within the population (Michalski, 2000; Wojtusiak & Michalski, 2006). These models can then be utilised to distinguish potentially good solutions, from potentially poor solutions, generate new population members which are potentially good, and incorporate these into the algorithm. Due to this mechanism, LEM has the appealing benefit that it can make "leaps of intuition" where if a promising area of search space is identified, it can be very quickly explored. Learnable evolution models for multiple objectives (LEMMO) (di Pierro et al., 2009; Jourdan et al., 2005a, 2005b, 2004) build on this, applying a similar technique to multiple objective models using the C4.5 algorithm (often used for decision-tree generation and data mining) as a rule induction algorithm in this case (Quinlan, 1993). Learning evolution models for multiple objectives with artificial neural networks (LEMMO-ANN) have also been developedþ, which in place of the C4.5 algorithm uses a feed forward artificial neural network as its distinguishing model (Fan et al., 2020; Sayers, 2015; Sayers et al., 2014, 2019).

An alternative approach is demonstrated in Behzadian's (2009) paper in which an artificial neural network is utilised to estimate fitness scores via regression rather than simply classifying solutions, and a caching mechanism is also incorporated to prevent re-evaluation of existing solutions. Both of these approaches are effective and show considerable promise in their application to challenging optimisation problems.

Whilst these meta-heuristic models do allow for approximation of some elements of optimisation algorithms internal metrics, thus dramatically speeding up the process of the algorithm's completion, they do come with their own challenges. The field of heuristics and particularly deep-learning and artificial neural networks has been progressing at an incredible pace over the last decade but these models do themselves have fairly high computational demands for training, and huge demands for data for that training to be effective. This often necessitates the use of graphics-processing units for their fast matrix multiplication capabilities, or even bespoke system-on-chip (SOC) designs of processing unit (Gordienko et al., 2021; LeCun et al., 2015). The time taken to train and utilise these algorithms can still be a net gain over neglecting their use, but it is important to note that they are not a free replacement due to these computational demands.

FINDINGS: OPTIMISATION OF RISK REDUCTION IN CIVIL ENGINEERING

Optimisation techniques for risk reduction in civil engineering projects is an active field of research, which often features in wider related fields of research such as risk analysis, optimisation and civil engineering. Civil engineering is a risk-averse profession and operation, for very good reasons, and it can therefore take time for new technologies and approaches to gain traction, as it is necessary that they are robustly evaluated and, as much as possible, proven effective. Nevertheless, there have been many significant research initiatives in recent years in academia and private industry (Boelee & Kellagher, 2015; Cesses & Kellagher, 2009; Kellagher et al., 2008; Kellagher & Cesses, 2009; Sayers et al., 2019). It is often incorporated into multi-objective optimisation where computing capabilities allow, as the consideration of risk as an isolated objective has limited value unless the model takes into account enough constraints. For example, in a flood risk scenario, building a city-sized subterranean storage tank under a given city will result in very low flood risk, but with a prohibitive cost even given the extensive funding available for many civil engineering projects. Constraints on a model will often remove large areas of the search space from exploration, and so there is a potential for improved solutions with multi-objective approaches where these are feasible. Additionally, historical methods of estimating risk do not always lend themselves well to a purely computational approach, and therefore methods of evaluating risk have been, and are, being developed, such as with the system-based analysis and management (SAM) approach to urban flood risks, which developed an expected annual damage (EAD) based analysis approach (Cesses & Kellagher, 2009; Kellagher et al., 2008; Kellagher & Cesses, 2009). This methodology involved several newly developed tools such as SAM-Risk 1 & 2, SAM-UMC (Wills, 2013), and a rapid flood-spreading model (RFSM) (Lhomme et al., 2008). Combining these tools with a series of design-storm or time-series based rainfall data allows for drainage system simulations to be run which produce the excess head of water present at each manhole in the drainage system. RFSM is then used to spread this excess water on the surrounding terrain, giving flood depth values, that can then be combined with other data, and other model runs, to give an estimate of expected annual monetary damage for that particular storm.

More recent examples include Sharafat et al.'s (2021) work on applying generic bow-tie risk analysis to tunnel-boring machine projects taking place in challenging conditions. Currently tunnel-boring machines are rapid and efficient methods for excavating but are considered very high risk when being used with adverse ground conditions. Determining exact geological conditions before commencement of tunneling projects is almost impossible, and so improved risk-analysis will allow more informed decisions to be made. Bow-tie risk analysis is a powerful tool used commonly in high-risk industries, involving the creation of bow-tie analysis diagrams which make analysis of given risks clear and easily interpretable in the context of potential threats, mitigations, and consequences. Currently this work lends itself more to human evaluation but could feasibly be adapted to computational interpretation. Another recent example is Yin et al.'s (2021) paper on the quantitative risk analysis of offshore well blowout utilising a bayesian network based approach with identification of principal risk factors and main causes for blowouts. A Bayesian approach here offers the possibility of taking better account of the multivariate problem of geological condition, methods, technologies, and other aspects, to better quantify risk. As a final example, Zeinalnezhad et al.'s (2021) presentation of a hybrid risk analysis model for wind farms, called the hybrid interpretive structural modelling, coloured petri-nets method (hybrid ISM-CPP). This approach is currently linked to questionnaires given to survey targeting experts, but it is possible to see how machine learning models over time could be used to make this a more automated risk-analysis approach.

DISCUSSION

The above review of optimisation techniques and worked examples suggest some issues and points worthy of further discussion.

Firstly, although a human element will probably exist within risk analysis and risk reduction for some time, it is likely that *computer-driven and optimisation approaches will see increasing adoption*, because of the support they can give to humans in comparing and contrasting solutions, and identifying novel solutions that may be unintuitive. One of the strengths of optimisation approaches is this ability to identify solutions which may be unintuitive until tested, and thus may not have been otherwise considered. In addition, access to hitherto unaffordable computing power is now available through cloud technologies, even from standard computing devices, due to the ubiquity of internet connectivity which is progressing through technologies such as 5G (Rana et al., 2021) and satellite mega-constellations. These technology advances offer internet access at high speeds in locations where it may not have existed previously (del Portillo et al., 2021; LC, 2020), which could allow computationally expensive work such as objective function evaluation or artificial intelligence online-learning to take place within the cloud.

Secondly, as more human-centric approaches start to be replaced by computer-driven and optimisation approaches, questions will arise around the *ethics of these approaches*. It is imperative that these technologies are used with safety foremost in mind, and with considered expert evaluation of their suggestions rather than with blind trust. Should a careless approach result in a loss of life or some other tragedy, besides the obvious and natural desires to avoid such outcomes, trust in these technologies may not recover. That said, it is worth noting that where the techniques discussed in this chapter are proven effective, ethical considerations should ultimately lead to their adoption. Ethics is a vital consideration in the usage of artificial intelligence techniques also and a burgeoning ethical artificial intelligence research field has been growing in recent years. Therefore, it seems likely that for a long while before any replacement of experts in fields where optimisation is possible, the skillset associated with these kind of specialist roles may instead shift to incorporate knowledge of optimisation techniques, computational risk assessment, ethical considerations, and related areas. Additionally, more cross-disciplinary work with experts such as data scientists and machine learning engineers may be necessary.

Thirdly, as risk assessment becomes a task which is more automated, supported by human expertise, *the capacity for thorough risk assessments related to civil engineering and other projects should increase.* A reduction in the time taken for projects due to additional throughput should allow for the consideration of projects and opportunities that prior to this may not have been considered.. On a similar note, some projects which previously may not have been considered feasible may become feasible through a better understanding of the risks associated, and through novel unintuitive solutions to remove risk from the projects, generated by optimisation algorithms. These advances therefore could lead to a safe increase in the number of viable civil engineering projects.

Fourthly and finally, as the field progresses, *the need for experts in machine learning, data science, civil engineering, and engineering in general to work in an interdisciplinary nature is likely to grow exponentially*. This is a trend likely to be reflected across a broad range of industries, as optimisation and machine learning lend themselves to extremely diverse applications.The skillsets required of experts in their field are likely to shift to represent this increasingly interdisciplinary aspect of the field. These fields are too complex and technical for expert skills across them all to be commonly present in one individual.

CONCLUSION

This chapter has explored a range of optimisation techniques and their application to risk reduction, particularly in relation to civil engineering projects. It has also touched upon areas such as data analytics, deep learning, and risk analysis methods. The scope of this coverage has meant that this can be no more than an overview, but with reference across a broad range of relevant literature, which it is hoped will serve well as an introduction to the topic.

The application of meta-heuristic methods is still in its infancy as a research field, and the possibilities are immense. Machine learning and data-science have seen very significant advances in recent years, and although the computational needs for training deep models are extreme, cloud resources are making these much more accessible (Dong et al., 2021; Kou et al., 2021; Krizhevsky et al., 2012; LeCun et al., 2015; Pinto et al., 2021). In addition, the requirements for inference are considerably more modest, meaning pre-trained models can be used on accessible and affordable devices.

These emerging capabilities in data science and machine learning should transfer well into the optimisation and meta-heuristics research field. Meta-heuristics and artificial intelligence techniques, which are far more powerful, offer promising prospects for multi-objective optimisation, which is capable of dealing with even more complex problems than currently, as well as dealing with current optimisation problems more effectively. As well as these advances in the machine learning area, which link well into meta-heuristic approaches, there is also scope for research into the combination of meta-heuristics with different optimisation algorithm approaches, leading to an overall more effective algorithm. In terms of algorithms research, the success of ensemble methods within the machine learning area (Abuassba et al., 2021; Shiue et al., 2021) indicates the potential of combining algorithms and models, with their diverse strengths and weaknesses, in optimisation applications (Han et al., 2020; Tóth et al., 2020; Ye et al., 2021).

In combination with many other fields, civil engineering is likely to become more interdisciplinary in nature over time, and demand increasingly technical skillsets from its practitioners. These demands are likely to come with great opportunities within the field, for those who can facilitate them.

REFERENCES

Abuassba, A. O., Dezheng, Z., Ali, H., Zhang, F., & Ali, K. (2021). *Classification with ensembles and case study on functional magnetic resonance imaging*. Digital Communications and Networks. doi:10.1016/j.dcan.2021.03.004

Amine, K. (2019). Multiobjective Simulated Annealing: Principles and Algorithm Variants. *Advances in Operations Research*, *2019*, 1–13. Advance online publication. doi:10.1155/2019/8134674

Baldi, P. (1995). Gradient Descent Learning Algorithm Overview: A General Dynamical Systems Perspective. *IEEE Transactions on Neural Networks*, *6*(1), 182–195. doi:10.1109/72.363438 PMID:18263297

Behzadian, K., Kapelan, Z., Savić, D. A., Ardeshir, A., Savic, D., & Ardeshir, A. (2009). Stochastic sampling design using a multi-objective genetic algorithm and adaptive neural networks. *Environmental Modelling & Software*, *24*(4), 530–541. doi:10.1016/j.envsoft.2008.09.013

Bertsekas, D. P. (1999). Nonlinear Programming (2nd ed.). Athena Scientific.

Beyer, H.-G., & Schwefel, H.-P. (2002). Evolution strategies - A comprehensive introduction. *Natural Computing, 1*(1), 3–52.

Blanning, R. W. (1975). The Construction and Implementation of Metamodels. *Simulation*, *24*(6), 177–184. doi:10.1177/003754977502400606

Boelee, L., & Kellagher, R. (2015). ADAPT: A Drainage Analysis Planning Tool. In *TRUST: Transitions to the Urban Water Services of Tomorrow*. HR Wallingford. http://files/656/ADAPT-ADrainageAnalysisPlanningTool.pdf

Burges, C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., & Hullender, G. (2005). *Learning to rank using gradient descent*. ACM. doi:10.1145/1102351.1102363

Cesses, Y., & Kellagher, R. (2009). *Optimisation options and Objectives Model Simplifications*. HR Wallingford.

Choromanska, A., Henaff, M., Mathieu, M., Ben Arous, G., & LeCun, Y. (2015). The loss surfaces of multilayer networks. *Journal of Machine Learning Research*, *38*, 192–204. https://arxiv.org/abs/1412.0233

Coello Coello, C. A. (1999). A Comprehensive Survey of Evolutionary Multiobjective Optimization Techniques. *Knowledge and Information Systems*, *1*(3), 269–308. doi:10.1007/BF03325101

Coello Coello, C. A. (2005). Twenty Years of Evolutionary Multiobjective Optimization: A Historical Overview of the Field. *IEEE Computational Intelligence Magazine*, *1*(1), 28–36. doi:10.1109/MCI.2006.1597059

De Jong, K. (1975). *An Analysis of the Behaviour of a Class of Genetic Adaptive Systems*. University of Michigan.

Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. Lecture Notes in Computer Science, 1917, 849–858. doi:10.1007/3-540-45356-3_83

Deb, K., & Jain, H. (2013). An Evolutionary Many-Objective Optimization Algorithm Using Reference-point Based Non-dominated Sorting Approach, Part I. *IEEE Explore*, *18*(c), 1–1. doi:10.1109/TEVC.2013.2281534

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, *6*(2), 182–197. doi:10.1109/4235.996017

Dede, T., Kripka, M., Togan, V., Yepes, V., & Rao, R. V. (2019). Usage of Optimization Techniques in Civil Engineering During the Last Two Decades. *Current Trends in Civil & Structural Engineering*, *2*(1). Advance online publication. doi:10.33552/CTCSE.2019.02.000529

del Portillo, I., Eiskowitz, S., Crawley, E. F., & Cameron, B. G. (2021). Connecting the other half: Exploring options for the 50% of the population unconnected to the internet. *Telecommunications Policy*, *45*(3), 102092. doi:10.1016/j.telpol.2020.102092 di Pierro, F., Khu, S. T., Savić, D., & Berardi, L. (2009). Efficient multi-objective optimal design of water distribution networks on a budget of simulations using hybrid algorithms. *Environmental Modelling & Software*, *24*(2), 202–213. doi:10.1016/j.envsoft.2008.06.008

Doerner, K. J., Gutjahr, W., Hartl, R., Strauss, C., & Stummer, C. (2003). Ant Colony Optimization in Multiobjective Portfolio Selection. *Proceedings of the 4th Metaheuristics International Conference*, 243-248.

Dong, S., Wang, P., & Abbas, K. (2021). A survey on deep learning and its applications. *Computer Science Review*, *40*, 100379. doi:10.1016/j.cosrev.2021.100379

Dorigo, M. (1992). *Optimization*. Learning and Natural Algorithms.

Dorigo, M., & Blum, C. (2005). Ant colony optimization theory: A survey. *Theoretical Computer Science*, *344*(2–3), 243–278. doi:10.1016/j.tcs.2005.05.020

Dorigo, M., Maniezzo, V., Colorni, A., & Dorigo, M. (1991). *Positive Feedback as a Search Strategy.* Technical Report 91-016. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.52.6342

Dorigo, M., & Stũtzle, T. (2002). A Short Convergence Proof for a class of ACO Algorithms. *IEEE Transactions on Evolutionary Computation, 6*, 358–365. http://files/737/01027747.pdf

Dorigo, M., & Stützle, T. (2010). Ant Colony Optimization: Overview and Recent Advances. In Handbook of Metaheuristics (Vol. 146, Issue May, pp. 227–263). doi:10.1007/978-3-319-91086-4_10

Fan, X., Sayers, W., Zhang, S., Han, Z., Ren, L., & Chizari, H. (2020) . *Review and Classification of Bio-inspired Algorithms and Their Applications.*, *17*, 611–631.

Geman, S., & Geman, D. (1984). Stochastic Relaxation, Gibbs Distributions and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *6*(6), 721–741. doi:10.1109/TPAMI.1984.4767596 PMID:22499653

Gendreau, M., & Potvin, J. Y. (2005). Tabu Search. In E. K. Burke & G. Kendall (Eds.), *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques* (pp. 165–186). Springer. doi:10.1007/0-387-28356-0_6

Ghimire, B., Chen, A. S., Guidolin, M., Keedwell, E. C., Djordjević, S., & Savić, D. A. (2013). Formulation of a fast 2D urban pluvial flood model using a cellular automata approach. *Journal of Hydroinformatics*, *15*(3), 676–686. doi:10.2166/hydro.2012.245

Glover, F., & Laguna, M. (1997). *Tabu Search*. Kluwer Academic Publishers. doi:10.1007/978-1-4615-6089-0

Glover, F., Taillard, E., & Taillard, E. (1993). A user's guide to tabu search. *Annals of Operations Research*, *41*(1), 1–28. doi:10.1007/BF02078647

Goldberg, D. E. (1989). Genetic algorithms in search, optimization, and machine learning. *Choice Reviews Online, 27*(2). doi:10.5860/CHOICE.27-0936

Gordienko, Y., Kochura, Y., Taran, V., Gordienko, N., Rokovyi, O., Alienin, O., & Stirenko, S. (2021). "Last mile" optimization of edge computing ecosystem with deep learning models and specialized tensor processing architectures. In Advances in Computers 122, 303-341. Academic Press Inc. doi:10.1016/bs.adcom.2020.10.003

Han, H., Zhang, Z., Cui, X., & Meng, Q. (2020). Ensemble learning with member optimization for fault diagnosis of a building energy system. *Energy and Building*, *226*, 110351. doi:10.1016/j.enbuild.2020.110351

Holland, J. H. (1962). Outline for a Logical Theory of Adaptive Systems. *Journal of the Association for Computing Machinery*, *9*(3), 297–314. doi:10.1145/321127.321128

Holland, J. H. (1975). Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence. In *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. MIT Press.

Jing, L., Zhuo-Qun, Z., Li-li, Z., & Kang-jie, S. (2018). Multi-Objective Ant Colony Optimization Algorithm Based on Discrete Variables. *IOP Conference Series. Earth and Environmental Science*, *189*(4), 42031. doi:10.1088/1755-1315/189/4/042031

Jourdan, L., Corne, D., Savić, D., & Walters, G. (2004). Hybridising Rule Induction and Multi-Objective Evolutionary Search for Optimising Water Distribution Systems. *Fourth International Conference on Hybrid Intelligent Systems*, 434–439. 10.1109/ICHIS.2004.58

Jourdan, L., Corne, D., Savić, D., & Walters, G. (2005a). Preliminary Investigation of the `Learnable Evolution Model' for Faster/Better Multiobjective Water Systems Design. *Evolutionary Multi-Criterion Optimization*, 841–855. http://files/630/PreliminaryInvestigationoftheLearnableEvoluti.pdf

Jourdan, L., Corne, D. W., Savić, D., & Walters, G. (2005b). LEMMO: Hybridising rule induction and NSGA II for multi-objective water systems design. *Proceedings of the 8th International Conference on Computing and Control for the Water Industry, CCWI 2005: Water Management for the 21st Century, 2*, 57.

Kapelan, Z., Savić, D., & Walters, G. (2005). Optimal Sampling Design Methodologies for Water Distribution Model Calibration. *Journal of Hydraulic Engineering*, *131*(3), 190–200. doi:10.1061/(ASCE)0733-9429(2005)131:3(190)

Kellagher, R., & Cesses, Y. (2009). *SAM -System-based analysis and management of urban flood risks*. HR Wallingford.

Kellagher, R., Cesses, Y., & Counsell, C. J. (2008). *Management of urban flood risks-the development of a risk-based method.* Academic Press.

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, *220*(4598), 671–680. doi:10.1126cience.220.4598.671 PMID:17813860

Kolter, J. Z., & Procaccia, A. (2017). *15-780 – Graduate Artificial Intelligence: Optimization*. Carnegie Mellon University. Retrieved July 20, 2021, from https://www.cs.cmu.edu/~arielpro/15780s17/slides/optimization.pdf

Kou, L., Wang, X., Guo, X., Zhu, J., & Zhang, H. (2021). Deep learning based inverse model for building fire source location and intensity estimation. *Fire Safety Journal*, *121*, 103310. doi:10.1016/j.firesaf.2021.103310

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In F. Pereira, C. J. C. Burges, L. Bottou, & K. Q. Weinberger (Eds.), Advances in Neural Information Processing Systems (Vol. 25, pp. 1097–1105). Curran Associates, Inc. http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf

LcLC. (2020, September 12). Global internet via satellite takes shape. *New Scientist*, *247*(3299), 21. doi:10.1016/S0262-4079(20)31576-1

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. doi:10.1038/nature14539 PMID:26017442

Lhomme, J., Sayers, P., Gouldby, B., Samuels, P., Wills, M., & Mulet-Marti, J. (2008). *Recent Development and Application of a Rapid Flood Spreading Method*. HR Wallingford. doi:10.1201/9780203883020.ch2

Liu, X., & Gong, D. (2011). A comparative study of A-star algorithms for search and rescue in perfect maze. *2011 International Conference on Electric Information and Control Engineering, ICEICE 2011 - Proceedings*, 24–27. 10.1109/ICEICE.2011.5777723

Mei, L., & Wang, Q. (2021). Structural optimization in civil engineering: A literature review. In Buildings 11(2), 1-28. MDPI AG. doi:10.3390/buildings11020066

Michalski, R. S. (2000). Learnable evolution model: Evolutionary processes guided by machine learning. *Machine Learning*, *38*(1), 9–40. doi:10.1023/A:1007677805582

Miller, B. L., & Shaw, M. J. (1996). *Genetic Algorithms with Dynamic Niche Sharing for Multimodal Function Optimization*. Academic Press.

Mohan, B. C., & Baskaran, R. (2012). A survey: Ant Colony Optimization based recent research and implementation on several engineering domain. *Expert Systems with Applications*, *39*(4), 4618–4627. doi:10.1016/j.eswa.2011.09.076

Nicklow, J., Asce, F., Reed, P., & Asce, M. (2010). State of the art for genetic algorithms and beyond in water resources planning and management. *Journal of Water Resources Planning and Management*, *136*(4), 412–432. doi:10.1061/(ASCE)WR.1943-5452.0000053

Pinto, G., Piscitelli, M. S., Vázquez-Canteli, J. R., Nagy, Z., & Capozzoli, A. (2021). Coordinated energy management for a cluster of buildings through deep reinforcement learning. *Energy*, *229*, 120725. doi:10.1016/j.energy.2021.120725

Quinlan, J. R. (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc.

Rajput, S. P. S., & Datta, S. (2019). A review on optimization techniques used in civil engineering material and structure design. *Materials Today: Proceedings*, *26*, 1482–1491. doi:10.1016/j.matpr.2020.02.305

Rana, A., Taneja, A., & Saluja, N. (2021). Materials Today : Proceedings Accelerating IoT applications new wave with 5G : A review. *Materials Today: Proceedings*. Advance online publication. doi:10.1016/j.matpr.2021.03.292

Rechenberg, I. (1965). Cybernetic solution path of an experimental problem. In Royal Aircraft Establishment Translation No. 1122, B. F. Toms, Trans. Ministry of Aviation, Royal Aircraft Establishment.

Rechenberg, I. (1973). *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Frommann-Holzboog.

Sayers, W. (2015). *Artificial intelligence techniques for flood risk management in urban environments*. https://ore.exeter.ac.uk/repository/handle/10871/21030

Sayers, W., Savic, D., & Kapelan, Z. (2019). Performance of LEMMO with artificial neural networks for water systems optimisation. *Urban Water Journal*, *16*(1), 21–32. doi:10.1080/1573062X.2019.1611886

Sayers, W., Savić, D., Kapelan, Z., & Kellagher, R. (2014). Artificial intelligence techniques for flood risk management in urban environments. *Procedia Engineering*, *70*(February), 1505–1512. doi:10.1016/j.proeng.2014.02.165

Schaffer, J. D. (1984). Some experiments in machine learning using vector evaluated genetic algorithms. In *Other Information Thesis Ph D.* https://www.osti.gov/energycitations/product.biblio.jsp?osti_id=5673304

Schrijver, A. (1998). *Theory of Linear and Integer Programming*. Wiley. https://books.google.co.uk/books?id=zEzW5mhppB8C

Schwefel, H.-P. (1965). *Kybernetische Evolution als Strategie der experimentellen Forschung in der Strömungstechnik.* Unpublished Master's thesis, Technical University of Berlin. https://www.researchgate.net/publication/216301423_Kybernetische_Evolution_als_Strategie_der_Exprimentellen_Forschung_in_der_Stromungstechnik

Schwefel, H.-P. (1970). *Two-phase nozzle and hollow core jet experiments.* Academic Press.

Schwefel, H.-P. (1975). *Evolutionsstrategie und numerische Optimierung* (Unpublished PhD Thesis). Technische Universität Berlin.

Schwefel, H.-P. (1977). *Numerische Optimierung von Computer-Modellen mittels der Evolutionsstrategie: mit einer vergleichenden Einführung in die Hill-Climbing-und.* http://scholar.google.com/scholar?q=schwefel&btnG=&hl=en&as_sdt=0%2C39#0

Serafini, P. (1994). *Simulated Annealing for Multi Objective Optimization Problems.* In G. H. Tzeng, H. F. Wang, U. P. Wen, & P. L. Yu (Eds.), *Multiple Criteria Decision Making* (pp. 283–292). Springer New York.

Sharafat, A., Latif, K., & Seo, J. (2021). Risk analysis of TBM tunneling projects based on generic bow-tie risk analysis approach in difficult ground conditions. *Tunnelling and Underground Space Technology, 111*(July), 103860. doi:10.1016/j.tust.2021.103860

Shiue, Y. R., You, G. R., Su, C. T., & Chen, H. (2021). Balancing accuracy and diversity in ensemble learning using a two-phase artificial bee colony approach. *Applied Soft Computing*, *105*, 107212. doi:10.1016/j.asoc.2021.107212

Smith, K. I., Everson, R. M., Fieldsend, J. E., Murphy, C., & Misra, R. (2008). Dominance-Based Multiobjective Simulated Annealing. *IEEE Transactions on Evolutionary Computation*, *12*(3), 323–342. doi:10.1109/TEVC.2007.904345

Smith, K. I., & Savić, D. (2006). *A Study of Simulated Annealing Techniques for Multi-Objective Optimisation*. University of Exeter.

Soriano, P., & Gendreau, M. (1996). Diversification Strategies in Tabu Search Algorithms for the Maximum Clique Problems. *Annals of Operations Research*, *63*(2), 189–207. doi:10.1007/BF02125454

Stũtzle, T., & Dorigo, M. (2002). A short convergence proof for a class of ant colony optimization algorithms. *IEEE Transactions on Evolutionary Computation*, *6*(4), 358–365. doi:10.1109/TEVC.2002.802444

Topping, B. H. V. (1983). Shape Optimization of Skeletal Structures: A Review. *Journal of Structural Engineering*, *109*(8), 1933–1951. doi:10.1061/(ASCE)0733-9445(1983)109:8(1933)

Tóth, J., Tomán, H., & Hajdu, A. (2020). Efficient sampling-based energy function evaluation for ensemble optimization using simulated annealing. *Pattern Recognition*, *107*, 107510. doi:10.1016/j.patcog.2020.107510

Ulungu, E. L., Teghem, J., Fortemps, P., & Tuyttens, D. (1999). MOSA Method: A Tool for Solving Multiobjective Combinatorial Decision Problems. *Journal of Multi-Criteria Decision Analysis*, *8*(4), 221–236. doi:10.1002/(SICI)1099-1360(199907)8:4<221::AID-MCDA247>3.0.CO;2-O

Weerasuriya, A. U., Zhang, X., Wang, J., Lu, B., Tse, K. T., & Liu, C. (2021). Performance evaluation of population-based metaheuristic algorithms and decision-making for multi-objective optimization of building design. *Building and Environment*, *198*(April), 107855. doi:10.1016/j.buildenv.2021.107855

Wills, M. (2013). *SAM-UMC 1.1.5*. HR Wallingford.

Wojtusiak, J., & Michalski, R. S. (2006). The LEM3 implementation of learnable evolution model and its testing on complex function optimization problems. *GECCO 2006 - Genetic and Evolutionary Computation Conference, 2*, 1281–1288. doi:10.1145/1143997.1144197

Woodward, M. (2012). The use of real options and multi-objective optimisation in flood risk management. In *CEMPS: Vol. PhD*. https://ore.exeter.ac.uk/repository/handle/10036/3714

Woodward, M., Gouldby, B., Kapelan, Z., & Hames, D. (2014). Multiobjective optimization for improved management of flood risk. *Journal of Water Resources Planning and Management*, *140*(2), 201–215. doi:10.1061/(ASCE)WR.1943-5452.0000295

Woodward, M., Kapelan, Z., & Gouldby, B. (2013). Adaptive Flood Risk management under Climate Change Uncertainty using Real Options and Optimisation. *Risk Analysis, 1*(34), 75–92. http://files/505/HRPP593_Adaptive_flood_risk_management_under_climate_change_uncertainty_.pdf

Ye, Y., Wang, Z., & Zhang, X. (2021). Sequential ensemble optimization based on general surrogate model prediction variance and its application on engine acceleration schedule design. *Chinese Journal of Aeronautics*, *34*(8), 16–33. Advance online publication. doi:10.1016/j.cja.2021.03.010

Yin, B., Li, B., Liu, G., Wang, Z., & Sun, B. (2021). Quantitative risk analysis of offshore well blowout using bayesian network. *Safety Science, 135*(July), 105080. doi:10.1016/j.ssci.2020.105080

Zavala, G. R., Nebro, A. J., Luna, F., & Coello Coello, C. A. (2014). A survey of multi-objective metaheuristics applied to structural optimization. In Structural and Multidisciplinary Optimization 49(4), 537-558. Springer Verlag. doi:10.100700158-013-0996-4

Zeinalnezhad, M., Gholamzadeh, A., & Kleme, J. (2021). *A hybrid risk analysis model for wind farms using Coloured Petri Nets and interpretive structural modelling.* doi:10.1016/j.energy.2021.120696

KEY TERMS AND DEFINITIONS

Evaluation Function: Equivalent to objective function, fitness function.

Fitness Function: Equivalent to objective function, evaluation function.

Heuristic: A process by which an approximate answer to a solution can be derived.

Hyper Parameter: A value which can be modified, resulting in some change to the function of a computer model/process which is manipulating parameters relating to an objective function.

Meta-Heuristic: A heuristic process which is an addition to a process which is already in itself heuristic.

Non-Dominated Front/Set: Equivalent to the Pareto front/set, non-inferior front/set.

Non-Inferior Front/Set: Equivalent to the Pareto front/set, non-dominated front/set.

Objective Function: A function which when supplied a number of parameters relating to an optimisation problem, will return a value identifying the suitability of those parameters as a solution to the problem. Equivalent to fitness function, evaluation function.

Optimisation Algorithm: An algorithm which can be applied to a properly structured optimisation problem, identifying one or more solutions which meet defined criteria of suitability.

Optimisation Problem: A problem consisting of a number of parameters which can be manipulated to result in changes to the objective functions result, and one or more objectives which are quantifiable and differentiable.

Parameter: A value which can be modified, resulting in some change to an objective functions value.

Pareto Front/Set: The Pareto front/set is a set of all solutions in a multiple-objective search space, for which there is no way to improve performance in any objective, without simultaneously decreasing performance in one or more other objectives. Equivalent to non-dominated front/set, non-inferior front/set.