Journal of Environmental Management 197 (2017) 275–286

Contents lists available at ScienceDirect

Journal of Environmental Management

journal homepage: www.elsevier.com/locate/jenvman

Research article

Reservoir operation using a robust evolutionary optimization algorithm

Jafar Y. Al-Jawad, Tiku T. Tanyimboh*

Department of Civil and Environmental Engineering, University of Strathclyde Glasgow, 75 Montrose St, Glasgow G1 1XJ, UK

ARTICLE INFO

Article history:
Received 6 October 2016
Received in revised form
14 February 2017
Accepted 25 March 2017
Available online 7 April 2017

Keywords:
Evolutionary optimization algorithm
Reservoir operation policy
Multipurpose reservoir system
Reservoir drawdown limits
Self-adaptive recombination
Environmental water management

ABSTRACT

In this research, a significant improvement in reservoir operation was achieved using a state-of-the-art evolutionary algorithm named Borg MOEA. A real-world multipurpose dam was used to test the algorithm's performance, and the target of the reservoir operation policy was to fulfil downstream water demands in drought condition while maintaining a sustainable quantity of water in the reservoir for the next year. The reservoir's performance was improved by increasing the maximum reservoir storage by 14.83 million m^3. Furthermore, sustainable water storage in the reservoir was achieved for the next year, for the simulated low flow condition considered, while the total annual imbalance between the monthly reservoir releases and water demands was reduced by 64.7%. The algorithm converged quickly and reliably, and consistently good results were obtained. The methodology and results will be useful to decision makers and water managers for setting the policy to manage the reservoir efficiently and sustainably.

1. Introduction

Multipurpose reservoirs are widely used to serve multiple demands for domestic, industrial, irrigation, environment, hydropower production and flood control, to maximize the economic benefits. These types of systems are complex because of the nonlinear storage-inflow relationship, conflicting objectives, dynamic properties, nonlinear constraints, etc. (Haimes and Hall, 1977). In the field of water resources management, significant demands on water exploitation were observed in recent decades. This raises the challenge to manage and allocate water in a sustainable way, and reservoirs are essential for water resources management in a river basin (Horne et al., 2016; Jothiprakash and Shanthi, 2006).

Many methods for optimization were found to solve different types of problems such as linear programming, non-linear programming and dynamic programming, etc. (Horne et al., 2016). However, the classical optimization methods are generally not suitable for such complex problems for a number of reasons. For example, typically, they provide a single local optimum solution. Evolutionary algorithms on the other hand, use a population of solutions rather than one solution in every iteration (Deb, 2001). In recent decades, evolutionary optimization algorithms were widely used in different fields of engineering and science to solve real-world problems (Coello et al., 2007).

Regarding engineering applications, Formiga et al. (2003) used the Non-dominated Sorting Genetic Algorithm (NSGA II) to solve water distribution network problems. Régnier et al. (2005) applied NSGA II in electromechanical system design. In structural design, Tract (1997) used a genetic algorithm (GA) with Pareto ranking in truss design. Deb and Tiwari (2005) used NSGA II for design in the field of mechanical engineering. In the field of civil engineering, Feng et al. (1999) used a GA with Pareto ranking to optimize building construction planning.

To achieve effective operational management policies for water resources management problems, many researchers used different optimization approaches (Horne et al., 2016). Sharif and Wardlaw (2000) used a GA to maximize the hydropower production while allowing deficits to occur in irrigation supplies. Chenari et al. (2014) also used a GA to determine the releases from a reservoir. Furthermore, Tilmant et al. (2002) used fuzzy stochastic dynamic program to optimize the control rules for a multipurpose reservoir. Kim and Heo (2006) used MOGA (multi-objective genetic algorithm) to solve a multi-reservoir multi-objective problem. Wu and Zou (2012) applied MOGA to maximize both power generation and irrigation benefits. Scola et al. (2014) applied NSGA II to maximize

* Corresponding author.
E-mail addresses: jafar.al-jawad@strath.ac.uk (J.Y. Al-Jawad), tikutanyimboh@hotmail.co.uk (T.T. Tanyimboh).

http://dx.doi.org/10.1016/j.jenvman.2017.03.081
0301-4797/

power generation. Cancelliere et al. (2003) used a multi-objective optimization method to reduce the deficit in the releases for irrigation and improve municipal volumetric reliability.

Borg MOEA is a recent optimization algorithm that was introduced by Hadka and Reed (2013). In this research, Borg MOEA was used to solve a reservoir operation problem. These types of problems need a powerful algorithm to handle the complexity of the inflow-storage relationship. The Borg MOEA algorithm has six operators that compete to create offspring in each generation. The effectiveness of the algorithm is maintained throughout the optimization by deploying the most suitable combination of operators for crossover. In addition, Borg MOEA is able to detect stagnation and escape from local optima by reviving the search process.

The aim of the current study was to investigate the robustness and performance of the algorithm on a reservoir operation problem. A drought condition and an additional reservoir drawdown constraint were considered in order to test the algorithm's ability to find good solutions consistently in such critical conditions. In reservoir management, it is difficult to control the releases over the entire year in order to fulfil the downstream demands and to maintain the same or higher initial water storage in the reservoir for the next year in drought conditions. Hence, the influence of the extra drawdown constraint imposed was investigated.

2. Overview of the optimization approach

Hadka and Reed (2013) introduced Borg MOEA for many-objective optimization problems. Some of the features in Borg MOEA include (a) diversity preservation; (b) measurement of search progress and stagnation; (c) restart to move away from local optima; (d) multiple recombination operators that compete to produce offspring; and (e) use of a dominance archive. The algorithm uses six operators in the recombination process to improve the search progress and a dominance archive to store all the non-dominated solutions.

To preserve diversity, the objective space is divided into hyper-boxes whose dimensions are all equal to ε, as in Fig. 1. Thus the ε-box index vector is used to find the dominant solutions instead of the objective function values. The algorithm calculates this index by dividing the objective function value by ε, and then sets the result as the succeeding integer value. If two or more solutions are in the same ε-box, the dominant solution is the one which is nearest to the lower-left corner of the ε-box, in the case of a minimization problem.

For stagnation measurement, ε-progress was introduced, which measures the improvement while searching for new solutions. If the algorithm finds new solutions in a new unoccupied ε-box, it means that there is progress and the algorithm is allowed to continue. This can be observed more clearly in Fig. 1. On the other hand, if there is no improvement based on ε-progress for a certain number of evaluations, a revival process is triggered, to escape from any local optima. The details of the restart procedure are available in Hadka and Reed (2013). The algorithm maintains the population size as a certain ratio of the archive size during the optimization process. This feature was adopted from ε-NSGA II (Kollat and Reed, 2006) and is called the injection rate.

The algorithm employs multiple recombination operators to produce offspring. In fact, Borg MOEA provides a framework in which the selection of the recombination operators adjusts depending on the dynamic properties of the objective and solution spaces of the optimization problem, including the make-up and diversity of the candidate solutions, and the landscape of the objectives. The recombination operators in Borg MOEA are:

(a) simulated binary crossover (SBX) (Deb and Agrawal, 1994);

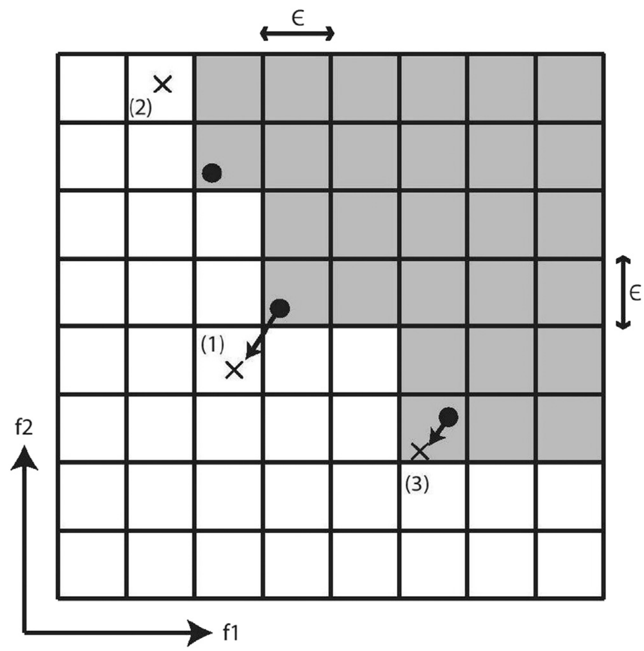

Fig. 1. Graphical representation of the ε-progress concept in a minimization problem with two objectives. Solutions (1) and (2) are new solutions in unoccupied boxes and thus represent improvements. Solution (3) is not considered as an improvement because it resides in a previously occupied box. The shaded boxes were previously occupied while the unshaded boxes were not previously occupied (Hadka and Reed, 2013).

(b) differential evolution (DE) (Storn and Price, 1997);
(c) parent-centric crossover (PCX) (Deb et al., 2002a,b);
(d) unimodal normal distribution crossover (UNDX) Kita et al. (2000);
(e) simplex crossover (SPX) (Tsutsui et al., 1999); and
(f) uniform mutation (UM) (Michalewicz et al., 1994).

Also, the polynomial mutation (PM) (Deb and Agrawal, 1994) is applied to the offspring produced by all the operators except for UM.

The probability of choosing a particular recombination operator to produce offspring depends on its ability to contribute non-dominated solutions in the dominance archive, compared to the other operators; hence the operator selection probabilities are proportional to their effectiveness and respective contributions.

The values of the decision variables in the offspring generated lie within the upper and lower bounds of the decision variables. The algorithm has many coefficients and parameters as summarised in Table 1 (Hadka and Reed, 2013) in which L represents the number of

Table 1
Default values of the parameters used in Borg MOEA.

Parameter	Value	Parameter	Value
Initial population size	100	SPX parents	10
Tournament selection size	2	SPX offspring	2
Epsilon, ε	0.01	SPX epsilon	2.0
SBX rate	1.0	UNDX parents	10
SBX distribution index	15.0	UNDX offspring	2
DE crossover rate	1.0	UNDX σ_ξ	0.5
DE step size	3.0	UNDX σ_η	$0.35/\sqrt{L}$
PCX parents	10	UM rate	$1/L$
PCX offspring	2	PM rate	$1/L$
PCX $\boldsymbol{\sigma_\eta}$	0.1	PM distribution index	20
PCX $\boldsymbol{\sigma_\zeta}$	0.1		

ε is the dimension of hyper-boxes in objective space; L is the number of decision variables; and the various σ symbols are variance-related parameters.