Incorporating filters in random search algorithms for the hourly operation of a multi-reservoir system

Duan Chen[1], Arturo S. Leon[2], Claudio Fuentes[3], Nathan L. Gibson[4],and Hui Qin[5]

Abstract: Optimization of short-term reservoir operation normally involves ramping constraints of outflows and water elevations at short time steps (e.g., hourly). Random search algorithms, such as Genetic Algorithms, have been widely used in optimization of reservoir operation. When applying random search algorithms to hourly reservoir operation, two important issues arise. The first one is the frequent violation of ramping constraints on the hourly reservoir outflows due to the random nature of the optimization algorithm. In other words, the optimization struggles to meet the ramping constraints when finding feasible solutions. The second issue is the zigzag fluctuation of the hourly decision variables as a result of the random search, which is unrealistic to implement in practice. In this study, the Savitzky-Golay smoothing filter (also known as least-squares filter) is incorporated periodically within the routine of the Non-dominated Sorting Genetic Algorithm (NSGA-II). The goal of this study is to smooth out the decision variables functions without deteriorating the performance of the optimization algorithm. The performance of the proposed approach is quantified through three indexes using a multi-reservoir system with 3360 decision variables as the test case. The results show that the use of the Savitzky-Golay filter not only

---

[1] Senior Engineer, Changjiang River Scientific Research Institute, Wuhan, China, 430010, (corresponding author), Email: chenduan@mail.crsri.cn

[2] Associate Professor, Department of Civil and Environmental Engineering, University of Houston, Houston, TX, US, 77204， Email: aleon3@Central.uh.edu

[3] Assistant Professor, Department of Statistics, Oregon State University, Corvallis, OR, USA, 97331, Email: fuentesc@science.oregonstate.edu

[4] Associate Professor, Department of Mathematics, Oregon State University, Corvallis, OR, USA, 97331, Email: gibsonn@oregonstate.edu

[5] Associate Professor, School of Hydropower and Information Engineering, Huazhong University of Science and Technology, Wuhan, China, 430074, Email: hqin@hust.edu.cn

provides a solution to the two aforementioned issues, but also significantly improves the performance of the NSGA-II for hourly reservoir operation. The optimal decisions obtained using the proposed approach display similar hourly variability to decisions of actual reservoir operation.

Keywords: Random search algorithm; zigzag operational scheme; Reservoir operation; Savitzky-Golay filter; Smoothing;

Introduction

Short-term reservoir operation usually involves short time steps (e.g., hourly) in an optimization horizon of several days or weeks. Ramping rates, which measure the changes on outflow and water surface elevation between the conservative time steps, are often considered in hourly reservoir operation due to navigational, environmental and recreational requirements (Edwards 2003; Niu and Insley 2013). The ramping rates are usually introduced in the optimization model as constraints that force them to lie between certain ranges. The inclusion of hourly ramping constraints can have a significant impact on reservoir operation (Veselka et al. 1995; Guisández et al. 2016) and correspondingly, on the performance of the optimization method.

Random Search Algorithms (RSA) refer to those algorithms that use some kind of random mechanism or probability (typically in the form of a pseudo-random number generator) in the optimization procedure. They are also known as stochastic optimization or global optimization methods (Zabinsky 2009). RSA include simulated annealing, tabu search, genetic algorithms, evolutionary programming, particle swarm optimization, and colony optimization, among others. None of these methods require the gradient of the problem to be optimized and hence, they can be used for functions that are not continuous or differentiable (Zabinsky 2015). Recently, various RSA have been widely applied to reservoir operation (Kumar and Reddy 2006; Afshar et al. 2007; Chen et al. 2016) due to their robustness, effectiveness, and global optimality properties. However, there are at least two issues that arise when using RSA for hourly reservoir operation, in which hourly ramping constraints are considered. The first issue is the recurrent violation of hourly ramping constraints due to the random generation of the initial population. RSA work by iteratively moving to better positions in the search space, which are sampled using some probability distribution (e.g., normal) defined around the current position. The random sampling may result in high fluctuations of the decision variables that are difficult to comply with the ramping constraints. The second issue is that the zigzag operational scheme resulting from high fluctuations in decision variables (Malekmohammadi et al. 2010) is often unrealistic to be implemented in practice.

Among the studies concerned with hourly reservoir operation with ramping constraints, the methods used for optimization mainly fall into the category of classical gradient-based methods, e.g., mix-integer linear programming (Needham et al. 2000; Chou and Wu 2015) or dynamic programming (Catalão et al. 2010; Wang and Zhang 2011). These methods do not iterate their candidate solutions by the mechanism of random distribution, and therefore the two issues mentioned above are not relevant in the classical gradient-based methods. However, other drawbacks such as the curse of dimensionality (Nandalal and Bogardi 2007) and not being appropriate to multi-objective optimization (Reddy and Kumar 2006) limit the classical methods for optimizing multi-objective and multi-reservoir systems. Recently, applications of the RSA to the optimization of multi-reservoir operation have shown promising results (Oliveira and Loucks 1997; Wardlaw and Sharif 1999; Labadie 2004; Reed et al. 2013; Chen et al. 2015) and have been receiving increasing attention. Most applications of the RSA on reservoir operation, however, focus on long-term planning and management with monthly time step or short-term optimization with a daily time step. The hourly ramping constraints are normally ignored for long time steps due to simplicity. Including hourly ramping constraints is essential for applying the RSA to the practice of reservoir operation. Furthermore, addressing the two aforementioned issues is critical for future applications of RSA to reservoir operation when using sub-hourly time steps, which are increasingly being considered in the optimization of power systems that combine wind generation and/or other renewable sources. These types of applications normally require sub-hourly time steps for their accurate representations (Wang and Liu 2011; Deane et al. 2014).

This study aims to address these issues by incorporating a filter function in the RSA. The goal is to smooth out the decision variables without deteriorating the performance of the optimization algorithm. Specifically, we consider the non-dominated sorting genetic algorithm, which is currently one of the most widely used random search methods. Malekmohammadi et al (2010) pointed out that high fluctuations of hourly outflows are a result from the Genetic Algorithm. In the study of Malekmohammadi et al (2010), the reservoir outflow itself is the objective for flood control and is incorporated with a coefficient of variation to minimize the hourly outflow variations. Our study, however, considers a much broader application in which the hourly ramping rates are expressed as constraints and the objectives of reservoir operation can be arbitrary. To test the performance of the proposed approach, a ten-reservoir system in the Columbia River, located in the Pacific Northwest of the United States, is used as a case study. For test case, we use three indexes to compare the performance of optimization experiments with and without filtering. The first index measures the ability of an optimization method to reduce constraint violation. The second index is the so-called hyper-volume index, which measures the convergence and diversity of the Pareto front, i.e., the final non-dominated solution. The third index measures the similarity (in variability) of model solutions to decisions of actual reservoir operation. This paper also investigates the influence of the frequency of filtering on the three aforementioned indexes.

Methodology

Non-dominated sorting genetic algorithm

The non-dominated sorting genetic algorithm, known as NSGA-II (Deb et al. 2002), is a widely used random search method for multi-objective problem (MOP) and has received increasing attention for study of reservoir operation (Prasad and Park 2004; Atiquzzaman et al. 2006; Yandamuri et al. 2006; Sindhya et al. 2011; Chen et al. 2013). The NSGA-II is a member of the Genetic Algorithm (GA) family and follows the primary principles of the classical GA. First, a set of candidate solutions (population) is generated randomly (first generation) that is essentially white noise. By using the selection operator, some candidate solutions in the population are selected. A so-called binary tournament is implemented and the chosen candidate solutions are compared in pairs based on the performances on the constraints and the objectives. For two feasible solutions (all the constraints are satisfied), the one that is better than the other according to the definition of dominance of the multi-objective is declared the winner. If one is feasible and another is not, the feasible one is better. If both solutions are infeasible, the one with smaller overall constraints violation wins the tournament. The winners of the tournament reproduce children (next generation) by using recombination and mutation operators. A child can be viewed as a random generation around a parent by some type of distribution. The evolution process continues until a stopping criterion is met. One of the most common stopping criteria is the number of generations. This criterion is problem-dependent, but generally, a large number of generations is used for ensuring solution convergence.

Savitzky-Golay smoothing filter

Filter functions are commonly used for time series data to smooth out short-term fluctuations and focus on longer-term trends and patterns. One of the simplest types of filters is the finite impulse response filter (FIR), which produces an output that is essentially a weighted average of the inputs or original data. The process can be described by the following equation (Giron-Sierra 2017):

$$S(t) = \sum_{n=-n_L}^{n_R} c_n G(t+n) \quad (1)$$

where *S(t)* is the output at time *t*. G(*) is the input data at time *; the index *n* indicates the number of the input data for generating one output data and ranges from n_L, the number of points to the left of the data point *t*, up to n_R, the number of points to the right of data point *t*. Finally, c_n represents the weighting factors that are used to emphasize the importance of the data at some specific time step. If we assume that $n_L = n_R$ and $c_n = 1/(n_L + n_R + 1)$, the smoothing process becomes the so-called moving average function (MAV).

The MAV is one of the standard averaging FIR filters, which tends to filter out a significant portion of the signal's high-frequency content along with the noise. This means that some information, such as the amplitude, may be reduced. In order to preserve the pertinent high-frequency components of the signal, the Savitzky-Golay smoothing filter (Savitzky and Golay 1964), also known as digital smoothing polynomial filters or least-squares smoothing filters, was developed. Unlike the constant weights used in the MAV, the Savitzky-Golay filter approximates the underlying time-series data by a polynomial. Specifically, for each point *G(t)*, a polynomial is fit, using least-squares, to all $n_L + n_R + 1$ points in the moving window, and then S(t) is set to be the value of that polynomial at position *t*. The Savitzky-Golay filter is essentially an optimization problem which minimizes the least-squares error of the polynomial fitted to frames of noisy data (Schafer 2011). The problem can be written in the following:

$$Minmize\ \sum_{n=-n_L}^{n_R} \left(\sum_{k=0}^{N} a_k\, n_k - x(n)\right)^2 \quad (2)$$

Where N is the order of the fitted polynomial. a_k is the coefficient for the k_{th} order of the polynomial and are determined in the process of finding the smallest least-squares error. Akaike information criterion (AIC; Akaike 1973) is used to determine the order of N and the window length i.e. n_L+ n_R+1. The model with N=2 i.e., quadratic model and window length of 5 has the smallest normalized AIC value (0.793) among the candidate models (N range from 1 to 5 and window length range from 2 to 10) and therefore, are selected in the study.

The Savitzky-Golay filter is typically used to "smooth out" a noisy signal whose frequency span (without noise) is large. For this reason, in this type of application, the Savitzky-Golay smoothing filter performs much better than the MAV and preserves more information from the original data (Vivó-Truyols and Schoenmakers 2006). The main purpose of adding a filter to random search algorithms is to smooth out the high fluctuation between two consecutive time steps. But at the same time, the amplitude of the decision variable is preserved, since this information may be helpful for finding the global optimal. To illustrate the advantage of the Savitzky-Golay filter with respect to the MAV filter, consider a time series data comprised of 120 hourly reservoir outflows. The reservoir outflows can be thought as a set of candidate decisions on how much water are being released. The reservoir outflow may be changed for every hour depending on the reservoir inflow and the power demand etc. However, the decision makers often prefer smooth change in the practice. First, the data was randomly generated by the NSGA-II algorithm without a filter. We then apply the Savitzky-Golay filter with a second-degree polynomial and an MAV filter, each with a moving window of 5, to the data. The comparison (Figure 1) shows that the Savitzky-Golay filter preserves much of the amplitude of outflows while as the MAV filter largely reduces the amplitude of outflows.

Incorporating the Savitzky-Golay filter to NSGA-II

To start the optimization, the NSGA-II randomly generates multiple sets of candidate decisions as the first generation. Each set of candidate decisions contains a certain number of decision variables.

Conventionally, each set of decision variables in the first generation is assigned a value that is randomly generated in the range of an upper bound and a lower bound, i.e., the so-called box constraint. Due to this generating mechanism, one decision variable may be assigned two very different consecutive values, which may result in a large zigzag fluctuation as shown in Figure 1.

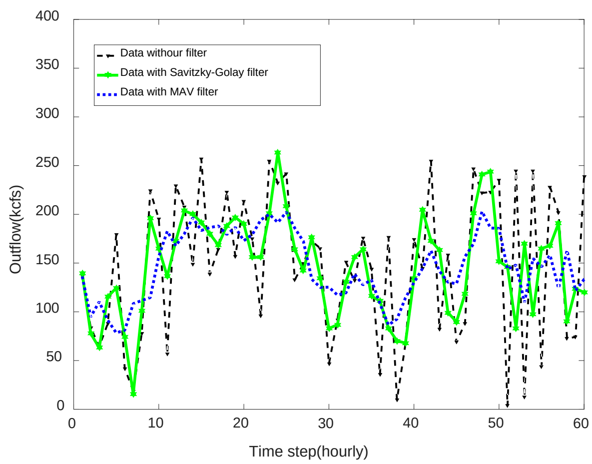


Figure 1. Comparison of the Savitzky-Golay and MAV filter on the data that are randomly generated by the NSGA-II(without filter)

In the present study, the Savitzky-Golay smoothing filter is incorporated in the routine of the NSGA-II. First, multiple sets of candidate decisions are randomly generated. Then, the Savitzky-Golay filter is applied on each set of candidate decisions, where the original generation is reconstructed by the smoothed out data. Then, the optimization process is continued as usual. The main steps of the optimization process are selection, recombination and mutation, where the decision variables can be replaced in the latter two steps. The fluctuation in the decision variables may be reintroduced at these two steps at later stages of the optimization. To maintain the smoothness of the decision variables, the Savitzky-Golay filter is applied periodically in the optimization process. However, the filtered candidate decisions may deteriorate the quality of the solutions. Hence, the frequency of the filtering is a parameter that can be evaluated for its trade-off on optimization performance. The procedure of incorporating the Savitzky-Golay filter into the

NSGA-II is shown in Figure 2. The incorporation of the filter into the NSGA-II involves only a few steps and its implementation is straightforward. The computational cost of adding the filter is small since the least-square process in the Savitzky-Golay filter involves only a linear matrix inversion and can be solved in advance (Press 2007). The frequency of applying the Savitzky-Golay filter is the only parameter that needs to be specified.

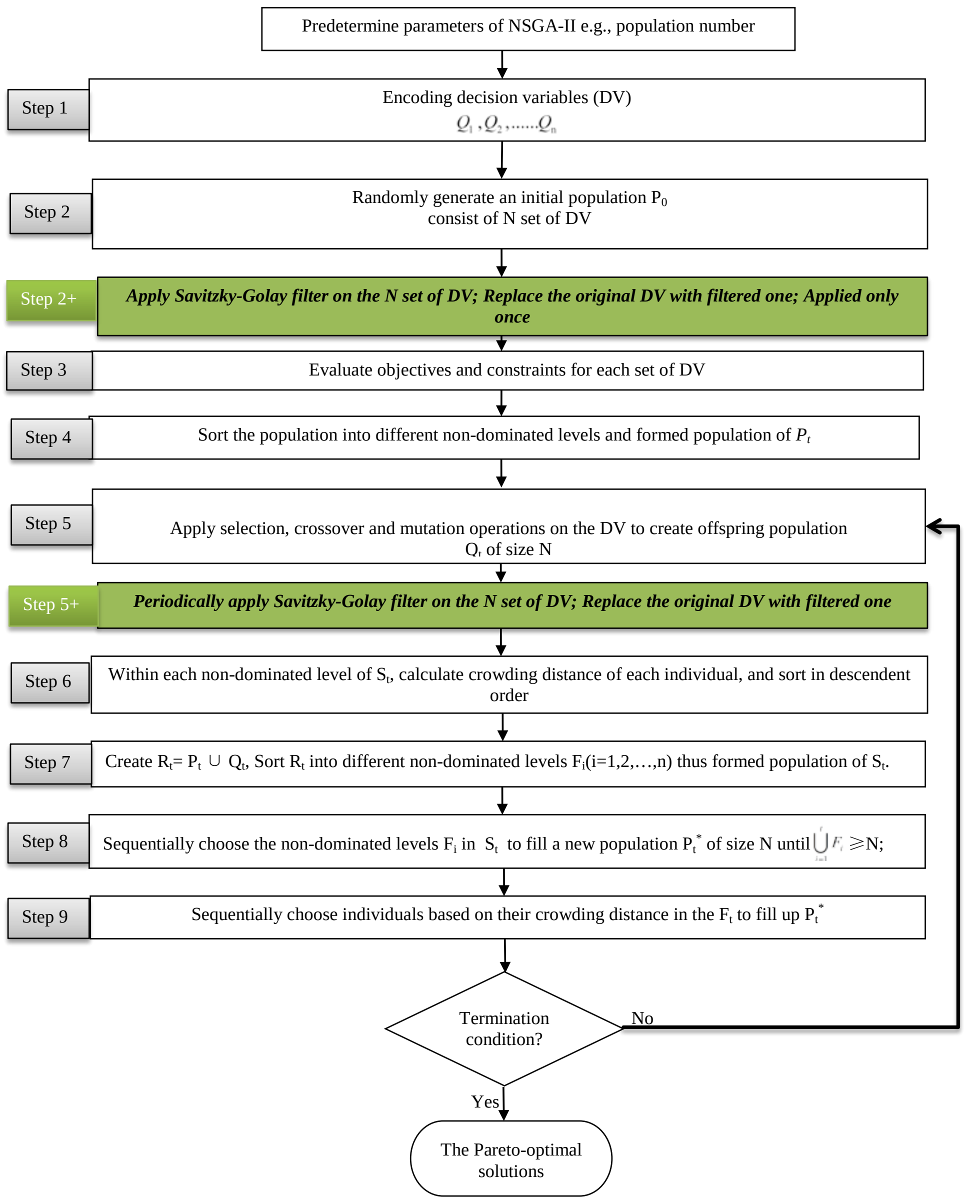


Figure 2. Incorporating the Savitzky-Golay filter into the NSGA-II (in italic and bold)

Indexes of performance evaluation

V-index

During the optimization process, the candidate solutions often violate the constraints, especially at early stage of the optimization. It is common that most or even all candidate solutions in the first generation are infeasible because of the random generation. The binary tournament in the process of the NSGA-II compares the infeasible solutions and selects the one with less violation of the constraints to reproduce children solutions. The process is expected to evolve the solutions with progressively less violation until feasible solutions are found. However, a feasible solution may be achieved only after many generations for cases in which the constraints are difficult to satisfy, i.e., a highly constrained problem within a complex search space. Therefore, the ability for reducing constraint violation is an important aspect for the optimization model. To compare the optimization performance in finding feasible solutions, this study propose a so-called V-index. The V-index is formulated in the following:

$$V_{\text{index}} = \frac{Con_{\text{intial}}}{G_{\text{f}}} \tag{3}$$

where G_f is the number of generations required to find the feasible solution and Con_{intial} is the average constraint violation of the initial population (the 1st generation). Violation of each constraint is indicated by a positive number, and its magnitude is proportional to the extent of the violation. A negative number or zero indicates no violation. The V_{index} can be viewed as a rate of reduction of constraints violation. The greater is the value of the V_{index}, the better is the performance of the optimization in finding feasible solutions. Note that this index can be near zero if no feasible solutions are found even when a large number of generations are used.

H-index

One of the most important evaluations of performance in the multi-objective optimization problem is the global optimality, commonly determined by two main aspects: convergence and diversity of the Pareto front (Deb et al. 2002). In this context, the hyper-volume index (H-index) is found to be a good metric for evaluating the performance of multi-objective optimization (Zitzler et al. 2000; Reed et al. 2013) due to its ability to combine convergence and diversity metrics into a single index. The H-index is defined as

$$H_{index} = \int_{(0,0)}^{(1,1)} \alpha_A(Z) dz \quad (4)$$

where *A* is an objective vector set, *Z* is the hyper-cube $(0,1)^n$ of the normalized objectives (n=2 in our test case). The $\alpha_A(Z)$ is a generalization of the multivariate cumulative distribution function$Fx(z) = P(X \leq z)$, also called attainment function (Fonseca et al. 2001). The $\alpha_A(Z)$ is equal to 1 if *A* is a weakly dominated solution set in *Z*. Basically, the H-index measures the volume of the objective space covered by a set of non-dominated solutions by calculating the volume of the objective space enclosed by the attainment function and the axes. Higher values of the hyper-volume index suggest better quality of the solutions in terms of convergence and diversity. In general, a true Pareto front or best-known Pareto approximation set (i.e., reference set) is ideal or preferred for performance evaluation. However, the hyper-volume index can be used to compare two intermediate solution sets (Knowles and Corne 2002).

S-index

In reservoir operation practice, smooth changes of decision variables, e.g. outflows, are preferred rather than large zigzag fluctuations. To compare the applicability of model solutions, the historical outflows are used as a benchmark. We propose an index that measures the similarity of the model solution to the benchmark in terms of shape smoothness. It is pointed out that we do not want to match the model solution exactly with the historical solution since there will be differences due to the optimization. Instead, we prefer a similar smoothness or linearity of the two sets, e.g., greater similarity (in shape) instead of smaller distances between the sets. Therefore, the *Lp-norm*, which measures the distance between two time series data, is not appropriate for this purpose. Instead, the Dynamic Time Warping (Berndt and Clifford 1994; Müller 2007) algorithm is used. The Dynamic Time Warping (DTW) is an algorithm that measures the similarity between two temporal sequences that may vary in time. The DTW has been successfully applied in fields of data mining and information retrieval due to its advantage for recognizing the "local shape" of the time series data (Petitjean et al. 2014). The DTW applies a local distance measure to compare the partial shape of two underlying data sets. A small distance indicates that the two set are similar in shape. For our study, we prefer model solutions with smaller DTW. On the other hand, the same or fewer turning points (from decrease to increase and vice versa) in the model solution are also desirable. Combining these two conditions, we define the S-index as

$$S_{\text{index}} = \frac{1}{\log(DTW_{\text{d}}) * \frac{TP_{\text{m}}}{TP_{\text{h}}}} \tag{5}$$

where DTW_d are the DTW distances from the model solution to the historical decisions. The DTW itself is an optimization problem and a program is used in the study (https://cn.mathworks.com/matlabcentral/fileexchange/43156-dynamic-time-warping--dtw-.) to calculate the DTW distance. The TP is also calculated by a program made by the authors in which a turning point is detected whenever the sign of the difference between two consecutive points are changed. TP_m and TP_h are the turning points in the model solution and historical decisions, respectively. The log function is used to reduce the magnitude of DTW_d , so that $Log(DTW_d\ )$ can have the same order of magnitude as $TP_{\text{m}}/TP_{\text{h}}$. According to Equation 4, a smaller DTW distance and fewer turning points result in a higher S_{index}. The larger the index is, the more applicable the model solution will be.

Case Study

The test case is a reservoir system on the Columbia River in the United States, which comprises 10 reservoirs. A sketch of the ten-reservoir system is shown in Figure 3.

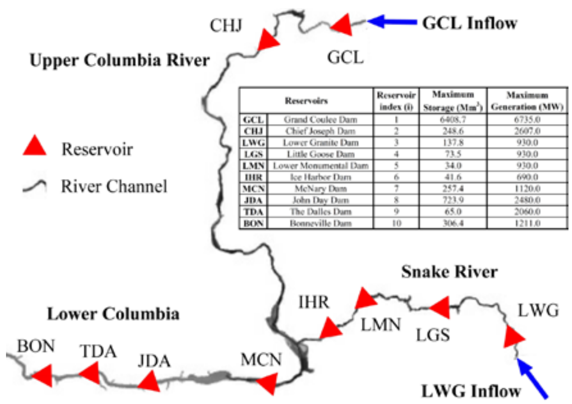


Figure 3. Sketch of the ten-reservoir system in the Columbia River (reprinted from Chen et al. 2017)

The reservoir system serves multiple purposes, e.g. power generation, ecological and environmental objectives (Schwanenberg et al. 2014, Chen et al. 2014). The optimization period is set to two weeks, beginning on August 25th and ending on September 7th. The reservoir system shifts some of the objectives during this two-week period based on seasonal consideration for fish migration and survival (Chen et al. 2014). It should be noted that the choice of the two-week period would not affect the performance of the proposed filter method as this method is designed for general use on short-term reservoir operation. The decision variables are the outflows at each reservoir for each hour during the optimization horizon, resulting in 3360 decision variables in total. The decisions normally are made through a joint team which including many stakeholders such as US Army Corps and Bonneville Power Administration. The reservoir system is coordinated under the decision-making team.

Objectives

Minimizing Power deficit to the demand

An important objective of the reservoir system is to meet power demand in the region. A deficit occurs when the generated power is less than the demand. Though the deficit can be compensated from buying power from an electricity market, it is desirable to minimize the power deficit during the operational horizon. This objective is expressed as

$$Minimize \sum_{t=1}^{T_h} (\min(0, \sum_{i=1}^{Nr} (PG_t^i) - PD_t)) \tag{6}$$

where PG is hydropower generated in the system (MWh), PD is power demand in the region (MWh). The variable t denotes time in hours and T_h is the optimization period (3360 hours). The index i represents reservoirs in the system, and N_r is the total number of reservoirs. The function *min(0, *)* expresses that the deficit is equal to 0 if the total power generated is greater than or equal to the power demand at time t.

Maximizing power generation for heavy load hours

It is desirable to generate more power during heavy load hours (certain hours in a day) for selling power to the electricity market at a higher price, which would increase the revenue. This objective is expressed as

$$Maximize \sum_{T_d=1}^{14} (\sum_{hr=6}^{22} (\max(0, \sum_{i=1}^{Nr} PG_{hr}^i - PD_{hr})) \tag{7}$$

where hr means heavy load hours (HLH) for a day (typically from 06:00 to 22:00). The quantity T_d corresponds to the optimization period in days (14 in our case). The function *max(0, *)* expresses that there is no excess power if the total power generated is smaller than or equal to the power demand at heavy load hours.

The two aforementioned objectives are generally conflicting when trying to move power generation from one period to another. One extreme case is to generate power only during HLH, which may lead to a large deficit on the demand in light load hours (LLH). Another extreme case is to meet the demand at all times (zero deficit) while generating excess power during HLH. However, the latter case is only possible if enough water is available. In the optimization model, the two objectives are normalized using a dimensionless index between zero and one. Other purposes of reservoir operation such as flood control, special operation Forebay(SOF) and seasonal requirements for fish migration and survival(fish flow) are expressed as constraints, and are described below.

Constraints

Reservoir forebay elevation constraints

The reservoir elevation constraints are expressed as

$$H_{r\min,i} \leq H_{r,i}^{t} \leq H_{r\max,i} \qquad (8)$$

where H_r is forebay elevation or reservoir water surface elevation; H_{rmin} and H_{rmax} are allowed minimum and maximum forebay elevations, respectively.

Fish flow constraints

To assist juvenile salmon and steelhead species in surface passage past the dams, most of the reservoirs in the system are required to spill a certain amount of flow through non-turbine structures such as sluices or gates. These flow requirements are expressed as either a fixed flow rate or a percentage of the total outflow of a reservoir (NOAA Fisheries 2014), these requirements are expressed as

$$Q_{s,i}^{t} = Q_{sr,i} \quad \text{(for } i = 5,7,8,9\text{)} \qquad (9)$$

$$Q_{s,i}^{t} = \frac{q_{s,i}}{100} Q_{out,i}^{t} \text{ (for } i = 3,4,6,10) \quad (10)$$

where Q_s is the spill flow, Q_{sr} is the fixed fish flow requirement, q_s is the flow rate and Q_{out} is the total outflow from the reservoir. According to the "Biological Opinion" issued by the National Oceanic and Atmospheric Administration (NOAA), the Grand Coulee (i=1) and Chief Joseph (i=2) reservoirs are not required to satisfy any fish flow requirement. Furthermore, the flow constraints are only required for the first week of the chosen period, namely from August 25th to August 31st.

SOF constraints

For the same purpose of assisting fish migration, the forebay elevations of reservoirs in the system are required to be kept within specific ranges, i.e., the SOF. The SOF requirements are expressed as follows

$$SOF_{lower,i} \leq H_{r,i}^{t} \leq SOF_{upper,i} \quad (11)$$

where H_r is forebay elevation, and SOF_{lower} and SOF_{upper} are lower and upper boundary for the SOF requirement, respectively. This flow constraint is also only required for the first week during the two-week period.

Turbine flow constraints

The turbine flow constraints are expressed as follows

$$Q_{tb\_\min,i} \leq Q_{tb,i}^{t} \leq Q_{tb\_\max,i} \quad (12)$$

where Q_{tb} is turbine flow, Q_{tb_min} and Q_{tb_max} are allowed minimum and maximum turbine flows, respectively.

Ramping limits for outflow

The ramping limits for the outflow are expressed as follows

$$\left| Q_{out,i}^{t} - Q_{out,i}^{t+1} \right| \leq Q_{out\_ramp\_allow,i} \quad (13)$$

where Q_{out} is outflow from the reservoir, $Q_{out\_ramp\_allow}$ is allowed ramping rate for the outflow between any two consecutive time steps.

Ramping limits for forebay elevation

The ramping limits for the forebay elevation are expressed as follows

$$H_{r,i}^{t} - H_{r,i}^{t+1} \leq H_{ramp\_down,i} \ (if \ H_{r,i}^{t} - H_{r,i}^{t+1} > 0 \ ) \tag{14}$$

$$H_{r,i}^{t+1} - H_{r,i}^{t} \leq H_{ramp\_up,i} \ (if \ H_{r,i}^{t} - H_{r,i}^{t+1} < 0 \ ) \tag{15}$$

where H_{ramp_up} is the allowed ramping rate when the reservoir water level is increasing and H_{ramp_down} is the allowed ramping rate when the reservoir water level is decreasing.

Ramping limits for tail water elevation

The ramping limits for tail water elevation are expressed as follows

$$TW_{r,i}^{t} - TW_{r,i}^{t+1} \leq TW_{ramp\_down,i} \ (if \ TW_{r,i}^{t} - TW_{r,i}^{t+1} > 0 \ ) \tag{16}$$

where TW_{ramp_down} is the allowed ramping rate for tailwater, which is only applied when tailwater elevation is decreasing.

Output constraints

The output constraints are

$$N_{d\_\min,i} \leq N_{d,i}^{t} \leq N_{d\_\max,i} \tag{17}$$

where N_d is power output, N_{d_min} is minimum output requirement, and N_{d_max} is maximum output capacity.

Constraints on end-of-optimization forebay elevation

The Forebay elevations of the ten reservoirs at the end of optimization are expected to stay within certain elevations in order to fulfill their future obligations. These targets are often determined by middle-term or long-term optimization models (Lund 1996), which are not part of this study.

In the present test case, historical forebay elevations are used as the target elevations at the end of the optimization. These constraints are expressed as:

$$H_{r,i}^{end} \geq H_{tar,i} \tag{18}$$

where $H_{r,i}^{end}$ is forebay elevation at the end of optimization; H_{tar} is the target forebay elevation at the end-of-optimization.

Reservoir System Modelling

The reservoir storages at each time step are modeled through the following equation (i.e., continuity equation) as to conserve the mass

$$V_i^{t+1} - V_i^t = ((Q_{in,i}^t + Q_{in,i}^{t+1})/2 - (Q_{out,i}^t + Q_{out,i}^{t+1})/2) \cdot \Delta t \tag{19}$$

where V is reservoir storage; Q_{in} and Q_{out} are inflow to and outflow from reservoirs, respectively; Δt is time step. The inflow is input to the model and the outflows are the decision variables. The evaporation or seepage is important for reservoir operation model set-up, particularly for long-term planning model or for the arid or semi-arid research area (Celeste and Billib 2010). Due to the short time frame in our study, water losses such as evaporation and seepage are not considered in the model.

The forebay elevations are obtained from the established forebay-storage relation by the given storages. The tail water for each dam is determined using a regression equation as a function of the dam outflow and the forebay elevation of the downstream reservoir. The turbine flow is modeled by relating the outflow with the fish flow requirement through the following procedures

$$Q_{tb}^t = \begin{cases} Q_{tb\_\min} & if \;\; Q_{tb\_\min} \leq Q_{out,i}^t < Q_{sr,i} + Q_{tb\_\min} \\ Q_{out,i}^t - Q_{sr,i} & if \;\; Q_{sr,i} + Q_{tb\_\min} \leq Q_{out,i}^t < Q_{sr,i} + Q_{tb\_\max} \\ Q_{tb\_\max} & if \;\; Q_{sr,i} + Q_{tb\_\max} \leq Q_{out,i}^t \\ Q_{out,i}^t & else \end{cases} \tag{20}$$

where Q_{tb} is turbine flow, Q_{tb_min} and Q_{tb_max} are allowed minimum and maximum turbine flows, respectively.

The power generation is computed based on the turbine flow and the water head (a function of forebay elevation and tailwater elevation) with project-aggregated coefficients

$$N_{d,i}^{t} = K_i(H_{r,i}^{t} - TW_i^{t}) \times Q_{tb}^{t} \quad (21)$$

where N_d is power output, TW is the tailwater elevation. K is the coefficient to express the overall efficiency of each turbine, which is aggregated as one value for each project (reservoir). In general, however, this value depends on water head and flow released in the turbines (i.e., a function of water head and flow). Schwanenberg et al (2014) validated the Big-10 Reservoir system by comparing the historical power generation from 2008-2012 with the simulated results from equation (20). The overall bias of the simulated project-aggregated power generation is in the range of −0.7 and 1.7MW and is, therefore, negligible when compared to the average generation of the individual projects. Therefore, the efficiency of the turbine as aggregated at the plant level is appropriate within the current modeling context. Note, however, this simplification may not be sufficient for unit commitment (UC) or other scheduling problems (Hidalgo et al., 2014), which are not being considered here, as the efficiency of turbines is sensitive to the performance of individual turbines. For the UC problem, a nonlinear function (normally high degree polynomial) of the generating discharge and the water head is often used to calculate power for each unit (Finardi et al. 2006).

The flow propagation within the reservoir-river network is modeled using Muskingum-Cunge routing method with calibrated coefficients. Most of the propagation times in the river between two reservoirs are 1-3 hours except the river reach between CHJ reservoir and MCN reservoir with an average propagation time of 21 hours.

Results

For each optimization run, the population and generation were set to 50 and 5000, respectively. Fifteen different experiments were tested in this case study. Because of the random nature of Genetic Algorithms, optimization results may have some differences for different runs, like other random-based search algorithms. For each experiment, a 30 random-seed replicate runs are used and the average values are reported, as in Fu et al. 2011. The typical parameters of the NSGA-II i.e., crossover rate were set as default values as recommended by Deb et al. 2002.

The first experiment (Ex0) did not use a filter while as the remaining fourteen (Ex1 to Ex14) used a different number of times that the filter is applied. The number of filtering times ranged from 1 to 40 with an increment of 3. The number of filtering times was evenly distributed among the total number of generations (i.e., 5000). The optimization model (written in Matlab) was executed on a desktop with Intel E3-1240/3.40GHZ/Dual Cores/24GB RAM. The CPU time for a typical experiment (population = 50, number of generations = 5000) was approximately 25 minutes. The time difference between the experiments with different filtering times is small since the filter is simply a few function evaluations during the model run. For an instance, the experiment with 1 times filter runs 1478s averagely and the experiment with 40 times filter runs 1483s averagely, resulting in 0.3% time difference.

For each run, the three aforementioned indexes were computed using Equations (2) to (4). To facilitate the comparison of results, all indexes were normalized to the range 0-1, where 0 and 1 correspond to the worst and best performance, respectively. The three indexes for all experiments are shown in Figure 4. To investigate the violation of constraints as a function of the generations for various filtering times, these are plotted in Figure 5. To assess the variation of the S-index, the Pareto fronts of various experiments are presented in Figure 6. To illustrate the best model solution in comparison to the historical operation, the solution of Ex7 and the historical hourly outflows are shown in Figure 7.

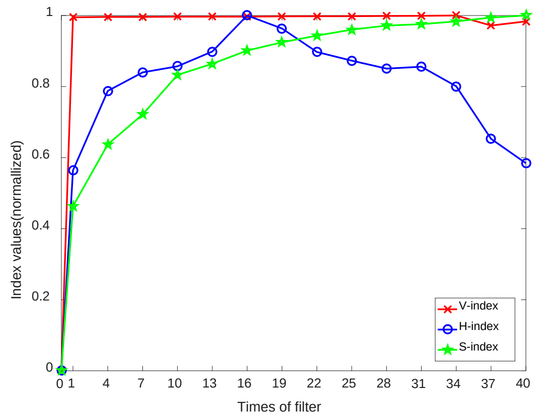


Figure 4. Index values for the experiments with different number of filtering times

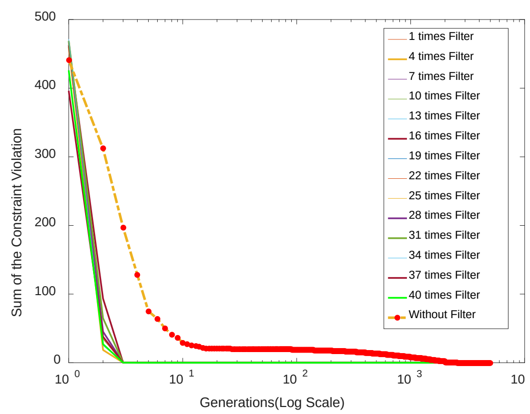


Figure 5. Violation of constraints versus generations for various filtering times

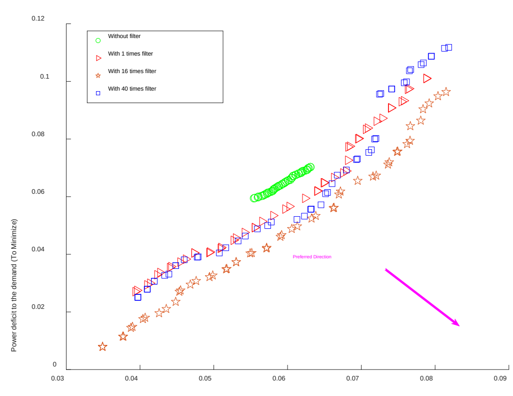
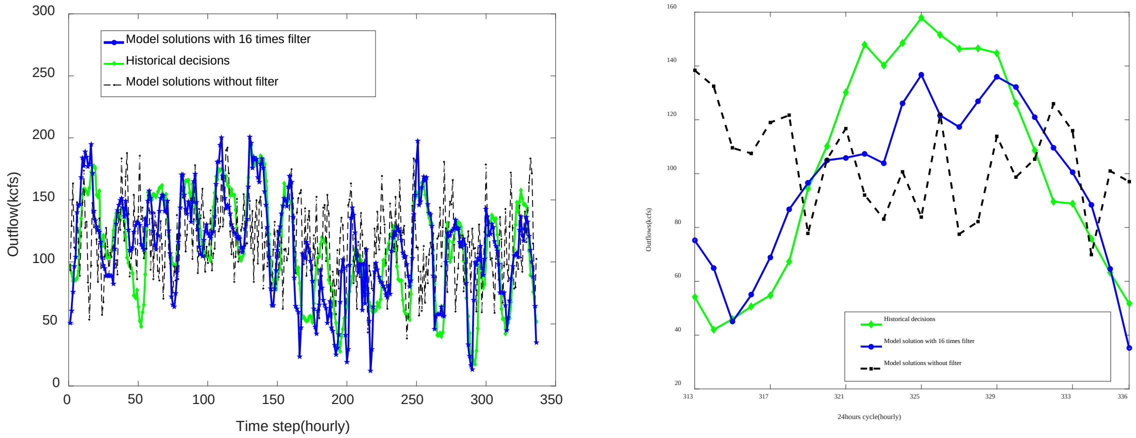


Figure 6. Pareto front for various filtering times



(a) **(b)**

Figure 7. Grand Coulee reservoir outflows for various scenarios of 336 hours (a) and that of a typical 24 hours cycle (b)

In Figure 6 we can observe that all three indexes for Ex0 (no filtering) are zero, meaning that this experiment has the worst performance compared to those that use a filter. For the experiments with filter (Ex1 to Ex14), the V-index for all experiments achieved similar values. However, the H-index and S-index varied significantly with the number of filtering times. The S-index increased monotonically with the number of filtering times. The H-index increased monotonically with the number of filtering times until NF = 16 (NF is the number of filtering times) and then decreased monotonically with the number of filtering times.

Figure 7 shows that the experiment with no filter (Ex0) required more than 2000 generations to reduce the violation of constraints to zero. Contrastingly, the experiments with filter (Ex1 to Ex14) reduced the violation of constraints to zero in as few as 3 or 4 generations. Figure 8 compares Pareto fronts of Ex0 (without filtering), Ex1 (1 time filtering), Ex7 (16 times filtering) and Ex14 (40 times filtering). Since this case study is a Max-Min optimization problem, the best solutions would be located at the bottom right corner of the objective space. However, a spread Pareto front is preferred for extending the range of optimal solutions (Deb et al., 2002). Notice that the solution of Ex0 is inferior to all other solutions in the figure. Figure 6 also shows that the solution of Ex7 (16 times filtering) has the best overall performance in terms of solution convergence and diversity. It is noted that Ex7 has also the best H-index as shown in Figure 4. Furthermore, as shown in Figure 9, the solution of Ex7 has a better agreement with the historical hourly outflows, in terms of frequency and amplitude, than the solution without filtering.

Discussion

The incorporation of a filter greatly improves the performance of NSGA-II in finding feasible solutions. For the traditional NSGA-II (with no filter), the initial population is randomly generated within a box constraint. Normally the decision variables, i.e., reservoir outflows can range from 0 to a large value such as 300 kcfs in our case study. Due to the random generation of the decision variables, the ramping constraints, which are the limits of the changes of two consecutive decision variables, can be frequently violated. Using a large number of generations may reduce the violation of constraints. However, this leads to a high computational cost. Incorporating a filter helps to smooth out the variability of the decision variables and therefore, to satisfy the ramping constraints much more efficiently. In addition, the number of generations needed to find feasible solutions (3 or 4 generations) is much smaller than those required when not using a filter (more than 2000 generations) as can be observed in Figure 7. This explains why the V-index, which measures the performance of finding feasible solutions, is much higher for the experiments with filtering than those without filtering (Figure 6).

The quick finding of the feasible solutions also contributes to a better Pareto front. The H-index, which measures the overall quality of the Pareto front, are higher for the experiments with filter compared to that without filter (Figure 6). At the same time, the Pareto front obtained from these experiments shows better convergence and diversity than the experiment without filter

(Figure 8). This is because of the so-called elitist preserving mechanism in the NSGA-II. Similar to other RSA, the NSGA-II maintains the best genes at each generation by assigning a higher probability to them for reproduction. For the experiments without a filter, most or even all candidate solutions may be unfeasible for many generations. In the latter case, the genes with the least violations are maintained and may dominate the population, which may lead to little or no improvement of solutions. This so-called premature convergence (Hrstka and Kučerová 2004, Chen et al. 2009) is caused by lack of diversity of the candidate solutions. This premature convergence is less critical for the experiments with the filter because feasible solutions are obtained after only a few generations.

Operational schemes obtained by an optimization model with the filter are more similar to the historical operation than those without a filter, as observed in Figure 9. This means that solutions obtained by the model with the filter are more reasonable to be implemented in practice. Thus, as expected, the S-index, which measures the similarity of model solutions to the historical operation, is higher for the models with filter compared to those without filter (Figure 6).

For the experiments with a different number of filtering times, the three indexes show different patterns (Figure 6). The V-index is almost the same for all experiments with filter, indicating that the V-index is not sensitive to the number of filtering times. This also indicates that the first filter reduces most of the zigzag fluctuation in the decision variables. Successive filtering is less effective in reducing fluctuation, since the data has already been smoothed out the first time the filter was applied. It is worth mentioning that applying filters an excessive number of times may decrease the quality of the Pareto front solutions i.e., lower H-index in Figures 6 and 8. This is because a filter removes some information from the original data (e.g., amplitude), which may help in finding optimal solutions. On the other hand, the S-index is monotonically increased with the number of filtering times, which indicates that the operational scheme obtained by a model with more filtering resembles better the historical operation. However, it should be noted that, the approach used in the study for determining the frequency of filtering (i.e., number of filtering times) is essentially a sensitivity analysis on single parameter and the result provide a somewhat ad-hoc solution. Different number of filtering times are expected for other cases and consequently, make itself a problem-dependent parameter. Since the filtering is involved in the process of optimization, the effect of filtering may interact with other parameters of the NSGA-II such as the population size and the number of generations. The difference (in terms of the zigzag behavior) between random generated solutions and the preferred (final) solutions can also affect the number of filtering times. Quantifying those interactive relations requires more cases studies and a global sensitivity analysis, which can be explored in future studies.

Since the optimization is multi-objective, each experiment results in a Pareto front that contains multiple points. Each point of the Pareto front is associated with a solution in the objective space and an operational scheme in the search space. Since each point on the Pareto front is indifferent in the context of multi-objective, selection of a point from the Pareto front merely depends on the preference of the decision maker. A neutral preference, representing a balanced attitude of the decision maker (towards the two objectives) is considered in the study. However, there are quite a few techniques which can help DM to select a "good" choice if some information is given such as the attitude towards risk (Emmerich and Deutz 2006; Blasco et al. 2008).

Conclusions

The two issues of the NSGA-II for hourly reservoir operation, i.e., a frequent violation of ramping constraints and unrealistic zigzag operational scheme, are addressed by incorporating a Savitzky-Golay smoothing filter in the NSGA-II optimization routine. The incorporation of this filtering technique significantly increases the ability of the optimization model in finding feasible solutions and overcoming the difficulty in satisfying the hourly ramping constraints. The V-index, which measures performance in finding feasible solutions, is much higher for the model with a filter than without a filter. The incorporation of a filter also smooths out the decision variables and the resulting operational scheme is not zigzag between consecutive time steps. The S-index, which measures the similarity of the model solution to the historical solution, is higher for the model with a filter than that without a filter. This means that the operational scheme obtained by the model with filter is similar to the historical operation. Hence, solutions obtained by the model with the filter are reasonable to be implemented in practice and greatly improve the performance of the NSGA-II. Furthermore, the H-index, which measures the overall quality of the Pareto front, is increased when the filter is incorporated.

Although the NSGA-II was the algorithm of choice in this study, the flexibility of the Savitzky-Golay filter would allow it use with other random search algorithms. Future work include the incorporation of wind generation into the power supply. The power generated from wind farms normally require sub-hourly time steps for their accurate representations, which may prompt the system operator to seek an even shorter time step solution from reservoir operations.

Acknowledgements

We Thank the Editor and two anonymous referees for their insightful and constructive comments. This work was supported by the Bonneville Power Administration projects TIP258 and TIP342. The authors would also thank the support from the National Natural Science Foundation of China (51479188, 91647114) and the Fundamental Research Funds for the Central Universities (CKSF2016009/SL).

References

Afshar, A., Haddad, O. B., Mariño, M. A., & Adams, B. J. (2007). Honey-bee mating optimization (HBMO) algorithm for optimal reservoir operation. Journal of the Franklin Institute, 344(5), 452-462.

Atiquzzaman, M., Liong, S. Y., & Yu, X. (2006).Alternative decision making in water distribution network with NSGA-II. Journal of water resources planning and management, 132(2), 122-126.

Berndt, D. J., & Clifford, J. (1994, July). Using Dynamic Time Warping to Find Patterns in Time Series. In KDD workshop (Vol. 10, No. 16, pp. 359-370).

Catalão, J. P. S., Pousinho, H. M. I., & Mendes, V. M. F. (2010). Scheduling of head-dependent cascaded reservoirs considering discharge ramping constraints and start/stop of units. International Journal of Electrical Power & Energy Systems, 32(8), 904-910.

Celeste, A. B., & Billib, M. (2010). The role of spill and evaporation in reservoir optimization models. Water resources management, 24(4), 617-628.

Chen, D., Leon, A. S., & Hosseini, P. (2014). Optimizing short-term operation of a multireservoir system during transition of objectives and constraints. In World Environmental and Water Resources Congress (pp. 1093-1105).

Chen, D., Li, R., Chen, Q., & Cai, D. (2015). Deriving optimal daily reservoir operation scheme with consideration of downstream ecological hydrograph through a time-nested approach. Water Resources Management, 29(9), 3371-3386.

Chen, D., Leon, A. S., Gibson, N. L., & Hosseini, P. (2016). Dimension reduction of decision variables for multireservoir operation: A spectral optimization model. Water Resources Research. 52 (1) pp. 36-51.

Chen, D., Leon, A. S., Hosseini, P., Gibson, N. L., & Fuentes, C. (2017). Application of Cluster Analysis for Finding Operational Patterns of Multireservoir System during Transition Period. Journal of Water Resources Planning and Management, 04017028.

Chen, Q., Chen, D., Li, R., Ma, J., & Blanckaert, K. (2013). Adapting the operation of two cascaded reservoirs for ecological flow requirement of a de-watered river channel due to diversion-type hydropower stations. Ecological modelling, 252, 266-272.

Chou, F. N. F., & Wu, C. W. (2015). Stage-wise optimizing operating rules for flood control in a multi-purpose reservoir. Journal of Hydrology, 521, 245-260.

Deane, J. P., Drayton, G., & Gallachóir, B. Ó. (2014). The impact of sub-hourly modelling in power systems with significant levels of renewable generation. Applied Energy, 113, 152-158.

Deb K, Pratap A, Agarwal S, and Meyarivan T(2002) A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 6 (2): 182-197

Edwards, B.K., 2003. The Economics of Hydroelectric Power. Edward Elgar Publishing.

Fonseca, V. G., Fonseca, C. M., & Hall, A. O. (2001). Inferential performance assessment of stochastic optimisers and the attainment function. In International Conference on Evolutionary Multi-Criterion Optimization (pp. 213-225). Springer Berlin Heidelberg.

Fu, G., Kapelan, Z., & Reed, P. (2011).Reducing the complexity of multiobjective water distribution system optimization through global sensitivity analysis.Journal of Water Resources Planning and Management, 138(3), 196-207.

Finardi, E. C., & da Silva, E. L. (2006). Solving the hydro unit commitment problem via dual decomposition and sequential quadratic programming. IEEE transactions on Power Systems, 21(2), 835-844.

Giron-Sierra, J. M. (2017). Digital Filters. In Digital Signal Processing with Matlab Examples, Volume 1 (pp. 239-310). Springer Singapore.

Guisández, I., Pérez-Díaz, J. I., & Wilhelmi, J. R. (2016). Influence of the Maximum Flow Ramping Rates on the Water Value. Energy Procedia, 87, 100-107.

Hidalgo, I. G., Correia, P. B., Arnold, F. J., Estrócio, J. P. F., de Barros, R. S., Fernandes, J. P., & Yeh, W. W. G. (2014). Hybrid model for short-term scheduling of hydropower systems. Journal of Water Resources Planning and Management, 141(3), 04014062.

Knowles, J., &Corne, D. (2002). On metrics for comparing nondominated sets. In Evolutionary Computation, 2002.CEC'02.Proceedings of the 2002 Congress on (Vol. 1, pp. 711-716).IEEE.

Kumar, D. N., & Reddy, M. J. (2006). Ant colony optimization for multi-purpose reservoir operation. Water Resources Management, 20(6), 879-898.

Labadie, J. W. (2004). Optimal operation of multireservoir systems: state-of-the-art review. Journal of water resources planning and management, 130(2), 93-111.

Malekmohammadi, B., Zahraie, B., & Kerachian, R. (2010). A real-time operation optimization model for flood management in river-reservoir systems. Natural hazards, 53(3), 459-482.

Müller, M. (2007). Dynamic time warping. Information retrieval for music and motion, 69-84.

Nandalal, K. D. W., & Bogardi, J. J. (2007). Dynamic programming based operation of reservoirs: applicability and limits. Cambridge university press.

Needham, J. T., Watkins Jr, D. W., Lund, J. R., & Nanda, S. K. (2000). Linear programming for flood control in the Iowa and Des Moines rivers. Journal of Water Resources Planning and Management, 126(3), 118-127.

Niu, S., & Insley, M. (2013). On the economics of ramping rate restrictions at hydro power plants: Balancing profitability and environmental costs. Energy Economics, 39, 39-52.

Oliveira, R., & Loucks, D. P. (1997). Operating rules for multireservoir systems. Water resources research, 33(4), 839-852.

Petitjean, F., Forestier, G., Webb, G. I., Nicholson, A. E., Chen, Y., & Keogh, E. (2014, December). Dynamic time warping averaging of time series allows faster and more accurate classification. In Data Mining (ICDM), 2014 IEEE International Conference on (pp. 470-479). IEEE.

Prasad, T. D., & Park, N. S. (2004).Multiobjective genetic algorithms for design of water distribution networks. Journal of Water Resources Planning and Management, 130(1), 73-82.

Press, W. H. (2007). Numerical recipes 3rd edition: The art of scientific computing. Cambridge university press.

Press, William H., and Saul A. Teukolsky. "Savitzky‐Golay Smoothing Filters." Computers in Physics 4, no. 6 (1990): 669-672.

Reddy, M. J., & Kumar, D. N. (2006). Optimal reservoir operation using multi-objective evolutionary algorithm. Water Resources Management, 20(6), 861-878.

Reed PM, Hadka D, Herman J, Kasprzyk J, Kollat J(2013) Evolutionary Multiobjective Optimization in Water Resources: The Past, Present, and Future. Adv Water Resour 51:438-456

Savitzky, A., & Golay, M. J. (1964). Smoothing and differentiation of data by simplified least squares procedures. Analytical chemistry, 36(8), 1627-1639.

Schafer, R. W. (2011). What is a Savitzky-Golay filter?[lecture notes]. Signal Processing Magazine, IEEE, 28(4), 111-117.

Schwanenberg, D., Xu, M., Ochterbeck, T., Allen, C., &Karimanzira, D. (2014).Short-term management of hydropower assets of the Federal Columbia River power system. Journal of Applied Water Engineering and Research, 2(1), 25-32.

Sindhya, K., Deb, K., &Miettinen, K. (2011).Improving convergence of evolutionary multi-objective optimization with local search: a concurrent-hybrid algorithm. Natural Computing, 10(4), 1407-1430.

Veselka, T. D., Hamilton, S., & McCoy, J. (1995). Optimizing hourly hydro operations at the Salt Lake City area integrated projects (No. CONF-950414--). Argonne National Laboratory (ANL), Argonne, IL.

Vivó-Truyols, G., & Schoenmakers, P. J. (2006). Automatic selection of optimal Savitzky-Golay smoothing. Analytical chemistry, 78(13), 4598-4608.

Wang, J., & Liu, S. (2011). Quarter-hourly operation of hydropower reservoirs with pumped storage plants. Journal of Water Resources Planning and Management, 138(1), 13-23.

Wang, J., & Zhang, Y. (2011). Short-term optimal operation of hydropower reservoirs with unit commitment and navigation. Journal of Water Resources Planning and Management, 138(1), 3-12.

Wardlaw, R., & Sharif, M. (1999). Evaluation of genetic algorithms for optimal reservoir system operation. Journal of water resources planning and management, 125(1), 25-33.

Yandamuri, S. R., Srinivasan, K., &MurtyBhallamudi, S. (2006). Multiobjective optimal waste load allocation models for rivers using nondominated sorting genetic algorithm-II. Journal of water resources planning and management, 132(3), 133-143.

Zabinsky, Z. B. (2009). Random search algorithms. Wiley Encyclopedia of Operations Research and Management Science.

Zabinsky, Z. B. (2015). Stochastic Adaptive Search Methods: Theory and Implementation. In Handbook of Simulation Optimization (pp. 293-318). Springer New York.

Zitzler, E., Deb, K., & Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. Evolutionary computation, 8(2), 173-195.