

# Comparative Study on Single-objective Optimization Algorithms for Reservoir Operation\*

Xinyuan Liu, Yonghui Zhu, Xiaohu Guo

Changjiang River Scientific Research Institute of Changjiang Water Resources Commission, Wuhan  
Email: wishesliu@126.com

Received: Sep. 2<sup>nd</sup>, 2013; revised: Oct. 10<sup>th</sup>, 2013; accepted: Oct. 24<sup>th</sup>, 2013

**Abstract:** Selection of optimal algorithms is one of the most complex problems for reservoir operation. Progressive optimality algorithm (POA), genetic algorithm (GA) and differential evolution algorithm (DE) were selected in this paper, and the performance of these algorithms were compared from the aspects of the number of decision variables, selection of arithmetic operators, determination of parameter values, constraint handling, etc. Results show that modern intelligent algorithms were applicable to reservoir operation optimization with big differences in performance for different intelligent algorithms. So it is necessary to select appropriate operators and parameters when using modern intelligent algorithms in reservoir operation. GA and DE with proper operators and parameters may have an advantage over POA in performance for reservoir operation problems with less decision variables, but POA is still superior to GA and DE for complex reservoir operation problems with large number of decision variables. This study helps to select proper optimization algorithms and parameter values for reservoir operation.

**Keywords:** Reservoir Operation; POA; GA; DE; Constraint Handling

## 水库单目标优化调度技术比较研究\*

刘心愿, 朱勇辉, 郭小虎

长江水利委员会长江科学院, 武汉  
Email: wishesliu@126.com

收稿日期: 2013年9月2日; 修回日期: 2013年10月10日; 录用日期: 2013年10月24日

**摘要:** 优化算法的选择是水库优化调度中的一个难点。本文选取了传统优化算法 POA、智能优化算法 GA 和 DE 等, 从变量规模、算子选择、参数选定、约束处理方法等方面对算法在水库优化调度中的性能进行了深入的比较研究。结果表明: 智能优化算法在水库优化调度中具有一定的适用性, 不同算法的搜索效率差异较大, 需要根据问题和算法特点对算子和参数进行精心设计和选择。对于变量较少的小规模水库优化调度问题, 通过选取合适的算子和参数, 智能优化算法 GA 和 DE 从耗时等方面会优于传统优化技术 POA; 但对于变量较多的复杂水库优化调度问题, POA 在耗时和解的质量等方面仍然具有显著的优势。研究成果可为水库优化调度优化算法的选择提供参考。

**关键词:** 水库调度; 逐次优化算法; 遗传算法; 差分进化算法; 约束处理

\*基金项目: 中央级公益性科研院所基本科研业务费项目(CKSF 2013003/HL), 十二五国家科技支撑项目(2012BAB04B03)。  
作者简介: 刘心愿(1981-), 男, 河南周口人, 博士, 工程师, 主要从事水库调度、河道演变及治理等研究工作。

## 1. 引言

水库优化调度是多约束多阶段决策的最优控制问题, 特别是对于多目标的综合利用水库, 以及大规模的水库群联合优化调度, 更增加了优化求解的复杂度。随着优化技术的发展以及水库调度研究的不断深入, 先后有各类优化算法不断地被引入到水库的优化调度中, 包括一些经典优化算法、最优控制理论方法(如大系统方法)以及现代智能算法等。关于这些算法的性能也进行了不少研究, 由于多采用一些标准测试函数, 虽然在一定程度上反映了一些实际问题的基本特征, 但无法体现实际工程应用问题计算的复杂性和多样性。作为现代智能算法之一的遗传算法(GA), 刘攀等<sup>[1,2]</sup>认为虽然理论上存在收敛到最优解的可能性, 但对于大规模的复杂水库优化调度问题, GA 不一定能够在有限的时间内寻求到满意的解。对于水库优化调度问题, 本文重点选择现代智能算法 GA 和 DE 等, 从参数、算子选择以及变量规模、约束处理技术等方面, 对其在水库优化调度中的性能进行深入的探讨, 并与经典优化算法 POA 的应用效果进行详细的分析和比较, 为水库优化调度中算法和参数的选择提供参考, 以减少算法、算子和参数选取的盲目性。

## 2. 水库调度优化算法

### 2.1. 逐次优化算法(POA)

POA 是 Howson 和 Sancho<sup>[3]</sup>提出的一种求解多阶段决策问题的优化计算方法, 通过将复杂的序列决策问题化为一系列的二阶段极值问题, 使原问题得到简化。对于每一个二阶段优化子问题, 可以采用非线性优化方法进行直接寻优, 进而得到较为精确的解。目前研究中, 多采用对求解域离散的方式<sup>[4,5]</sup>, 在一定程度上限制了搜索的精度和全局寻优能力, 本文采用单纯形法进行两阶段决策优化搜索。单纯形法最早由 Splendy 等<sup>[6]</sup>提出, Nelder 和 Mead<sup>[7]</sup>针对该法不能加速搜索, 以及在曲谷中或曲脊上进行搜索所遇到的困难, 对搜索方法做了若干改善。改进后的方法允许改变单纯形的形状, 应用  $E^n$  中  $n+1$  个顶点的可变多面体把具有  $n$  个独立变量的函数极小化。每一个顶点可由一个矢量  $\mathbf{x}$  确定, 在  $E^n$  中产生的  $f(\mathbf{x})$  最高值的顶点, 通过其余各顶点的形心连成射线, 用更好的点逐

次代替  $f(\mathbf{x})$  具有最优值的点, 就能找到目标函数的改进值, 一直到  $f(\mathbf{x})$  的极小值被找到为止。

### 2.2. 遗传算法(GA)

GA 是一类借鉴生物界的适者生存优胜劣汰遗传进化规律演化而来的随机化搜索方法。由 Holland<sup>[8]</sup>于 1975 年提出, 其主要特点是直接对结构对象进行操作, 不存在求导和函数连续性的限定; 具有内在的隐并行性和更好的全局寻优能力; 采用概率化的寻优方法, 能自动获取和指导优化的搜索空间, 自适应地调整搜索方向, 不需要确定的规则。标准遗传算法(SGA)采用二进制编码方式、简单算术交叉算子和均匀变异算子, 在实际工程问题中都有很大的局限性。通过对简单算术交叉算子进行改进, 目前已出现了多种实数编码交叉方法, 如混和交叉(BLX)<sup>[9]</sup>、模拟二进制交叉(SBX)<sup>[10]</sup>、单峰正态分布交叉(UNDX)<sup>[11]</sup>、单形交叉(SPX)<sup>[12]</sup>等, 其中 BLX 思路简单、实现方便、效率较高。混合交叉<sup>[9]</sup>实质上是在父代个体创建的超三角形内随机地创建后代。实数编码遗传算法中常用的变异算子有: 均匀变异算子、边界变异算子、非均匀变异算子、高斯变异算子等。其中非均匀变异算子<sup>[13]</sup>比较简单高效, 其搜索区域随着进化代数的增加有逐渐缩小的趋势。变异算子在遗传算法中是辅助算子, 具有一定的局部搜索能力; 交叉算子为主要算子, 具有较强的全局搜索能力而作为。

本文主要采用简单高效的混合交叉算子和非均匀变异算子的遗传算法, 并对其性能进行分析。混合交叉算子采用 BLX- $\alpha$  形式,  $\alpha$  为区间(0, 1)上的常数; 非均匀变异算子包含一个参数  $\beta$ , 它决定了随机数扰动对进化代数的依赖程度, 一般取 2~5 之间的整数。 $\alpha$  和  $\beta$  都需要预先根据经验进行选取。

### 2.3. 差分进化算法(DE)

DE 也是一种基于群体智能理论的优化算法, 与 GA 类似, 主要有变异和杂交等算子, 但 DE 是先变异后杂交, 且变异机制也有很大的区别。其基本原理是将种群中任意两个个体的差分向量加权后根据一定规则加到第三个个体上来获得新个体, 如果新个体目标函数值比种群中预先确定的一个个体的目标函数值小, 则用新个体替代, 否则原个体保存。DE 具

体实现细节见文献[14]。DE有多种变化形式,为表示方便,统一采用DE/x/y/z的形式来进行描述。 $x$ 表示变异操作中被变异的个体矢量的选择方式, $x$ 可为“RAND(表示从种群中随机选择一个个体)”或“BEST(表示选择当前种群中适应值最优的个体)”; $y$ 表示变异操作所用差分矢量的数目; $z$ 表示交叉方式,可以为“BIN”(表示采用伯努利试验方式来进行交叉操作)或者“EXP”(表示交叉操作概率分布满足指数分布)。即有DE/RAND/1/BIN、DE/BEST/1/BIN、DE/RAND\_to\_BEST/1/BIN、DE/BEST/2/BIN、DE/RAND Z/2/BIN、DE/RAND/1/EXP、DE/BEST/1/EXP、DE/RAND\_to\_BEST/1/EXP、DE/BEST/2/EXP、DE/RAND/2/EXP等多种组合形式的差分进化算法。

差分进化算法的主要控制参数有两个:变异因子 $F$ 和交叉概率 $CR$ 。 $F$ 为一个实常数,控制差分向量的缩放程度,进而控制着种群多样性和收敛性, $F \in [0, 2]$ ;  $CR$ 可控制个体参数的各维对交叉的参与程度,以及全局与局部搜索能力的平衡, $CR \in [0, 1]$ 。

### 3. 水库优化调度模型及约束处理

#### 3.1. 水库优化调度模型

实际水库调度问题比较复杂,需要从防洪、发电和弃水等方面进行考虑,为便于比较各算法性能,这里仅选取水库的防洪调度目标。对于一场洪水过程,采取最大防洪安全保证准则,即在满足大坝(或库区)防洪安全及下游防洪控制断面安全泄量等限制要求的条件下,尽可能多下泄,使留出的防洪库容最大,以备调蓄后续可能发生的大洪水。这里以平均出库流量最大为目标函数,即

$$\max Q = \frac{1}{T} \sum_{t=1}^T Q_t \quad (1)$$

式中, $T$ 为计算的时段个数, $Q_t$ 为洪水调度期内第 $t$ 个时段的总出库流量。主要约束条件如下:

##### 1) 水量平衡

$$V_{t+1} = V_t + (I_t - O_t) \cdot \Delta t \quad (2)$$

##### 2) 库容限制

$$V_t \leq V_t \leq V_u \quad (3)$$

##### 3) 机组出力限制

$$P_l \leq P_t \leq P_u \quad (4)$$

#### 4) 泄洪流量约束

泄洪流量需要考虑水库在某一水位 $Z$ 的最大泄流能力 $O_{\max}^Z$ 约束,也需要兼顾下游的防洪安全泄量 $O_{\max}^{\text{down}}$ 要求,取两者最小值作为泄洪设施的最大泄洪流量约束,即

$$O_t \leq \min(O_{\max}^Z, O_{\max}^{\text{down}}) \quad (5)$$

以上各式中, $t=1, 2, \dots, T$ ;  $\Delta t$ 为计算时段长, $I_t$ 和 $O_t$ 分别为入库流量和出库流量, $V_l$ 和 $V_u$ 分别水库蓄水量下限和上限, $V_t$ 为第 $t$ 个时段的蓄水量, $P_l$ 和 $P_u$ 分别机组出力的下限和上限, $P_t$ 为第 $t$ 个时段的机组出力,其他变量含义同上。

### 3.2. 约束条件处理

复杂约束的处理是各类算法实际应用过程中的难点,直接影响着算法优化性能的好坏,约束条件处理方法主要有舍弃策略、修复策略、惩罚函数法和多目标法等。水库优化调度一般都是复杂的高维、多约束问题,惩罚函数法和修复策略是相对较为可行的约束处理方法。

#### 1) 惩罚函数法

水库调度问题中存在不同类型的约束,有蓄水量约束、出力限制约束、流量约束等,约束的量纲和数量级都是不一样的,本文采用静态的分层罚因子法<sup>[15]</sup>,该方法将约束违反的程度分为多个等级,不同等级采用不同的罚因子,约束违反量越大,违反等级越高,对应的罚因子就越大。分级处理机制相对单一惩罚因子方法更为合理,但也更为复杂,需要一定的先验信息来确定各等级的罚因子值。

#### 2) 修复策略

修复策略是对于不可行解进行修正。水库调度问题是一个多阶段决策问题,当前阶段决策直接影响着后面阶段的决策,所以当前阶段的解直接影响到后面阶段解的可行性。由于来水的随机性,可能会出现后面阶段解无法修复的情况,即无法同时满足所有约束,如水位已接近正常高水位时,来水量仍然很大,按下安全泄量进行控制,库水位会超过正常高水位,而如果按水库正常高水位进行控制,又会导致出库流量超过下游防洪安全泄量要求,这样需要对前面

阶段的解进行重新调整和修复, 使得前后阶段的解都能满足所有约束要求, 因此修复过程比较复杂, 需要前后反复试算。

### 4. 水库优化调度算法比较研究

以三峡水库为研究对象, 选取资料是宜昌站 1954 年典型 100 年一遇设计洪水, 以 6 h 为计算时段, 30 天共 120 个时段; 为便于分析不同变量规模对算法性能的影响, 分别采用了不同变量规模  $n(n = 10、20、30、40、\dots、120)$  来进行优化, 表示对该洪水过程的前  $n$  个连续时段的资料进行优化调度。为便于统一比较, 不同变量个数的 GA 和 DE 种群数量都设定为 500; 迭代终止条件为: 收敛到最优值迭代终止, 或收敛不到最优值但连续运行 1000 代最优值变化量小于 1, 或超过最大代数 50,000。GA 和 DE 的运行步数即是进化的代数, POA 的运行步数指循环优化的次数, 对所有变量依次优化完毕一次算一步。对于不同的变量个数、参数组合和约束处理方法, 分别运行 10 次, 分别统计其最优值、运行步数和运行时间的平均值等。

#### 4.1. 不同交叉率和变异率对 GA 算法性能的影响分析

选取混合交叉算子( $\alpha = 0.7$ )和非均匀变异算子( $\beta = 2$ ), 分别对不同的交叉率  $P_c$  和变异率  $P_m$  组合来进行比较。图 1~3 分别为变量个数为 10、30 和 50 时不同交叉率和变异率组合的最优解和运行步数分布情况, 图中都是蓝色越深的区域越优。从图中可以看出, 变异率对解的质量和运行步数影响较大, 特别是变量较多时(大于 30), 变异率越大, 解的质量越差, 运行步数也就越多耗时越长, 所以变异率不超过 0.1 为宜, 当变量规模更大时, 变异率宜更小; 不同的交叉率对解的质量影响不大, 但对程序运行步数影响明显, 一般变量数为 30 左右时, 交叉率取 0.6~0.8 为佳, 当变量数为 50 左右, 交叉率宜低于 0.6; 可以看出, 变量规模越大, 交叉率宜取越小的值。

#### 4.2. 不同变异因子和交叉概率对 DE 算法性能的影响分析

选取 DE/BEST/1/EXP 模式, 对于不同变量规模和参数组合, 算法收敛时的运行代数分布图如图 4 所

示, 蓝色越深的区域越优。从图上可以看出, 当优化计算变量规模比较小时,  $F$  取较小值和  $CR$  取较大值(但小于 1)时, 算法的搜索性能最好; 随着变量规模的扩大, 为发挥算法的最佳搜索性能,  $F$  取值应该逐渐加大,  $CR$  取值应该逐渐减小, 如变量数不超过 30 时,

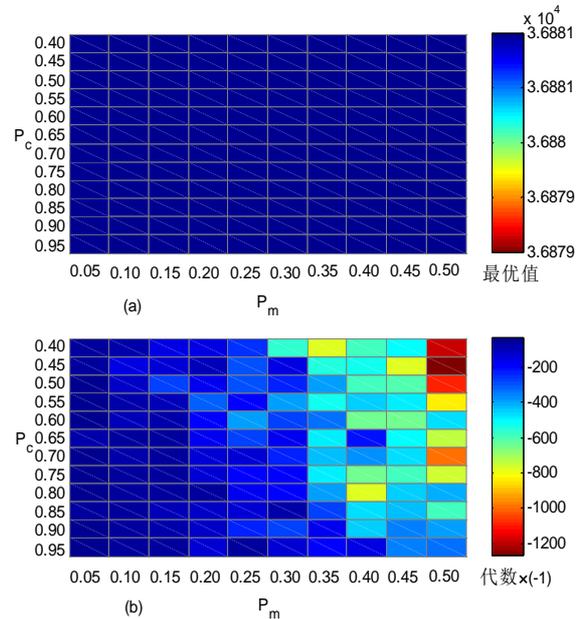


Figure 1. GA's performance with different crossover and mutation rates ( $n = 10$ )  
 图 1.  $n = 10$  时不同交叉率变异率组合的 GA 算法优化性能

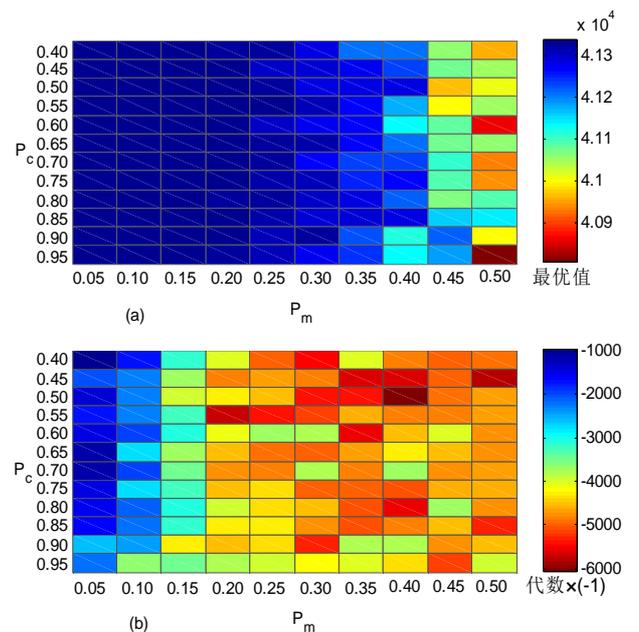


Figure 2. GA's performance with different crossover and mutation rates ( $n = 30$ )  
 图 2.  $n = 30$  时不同交叉率变异率组合的 GA 算法优化性能

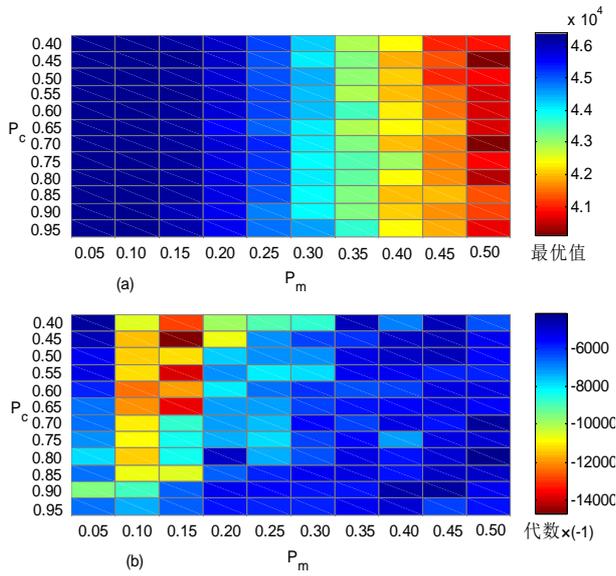


Figure 3. GA's performance with different crossover and mutation rates ( $n = 50$ )

图3.  $n = 50$  时不同交叉率变异率组合的 GA 算法优化性能

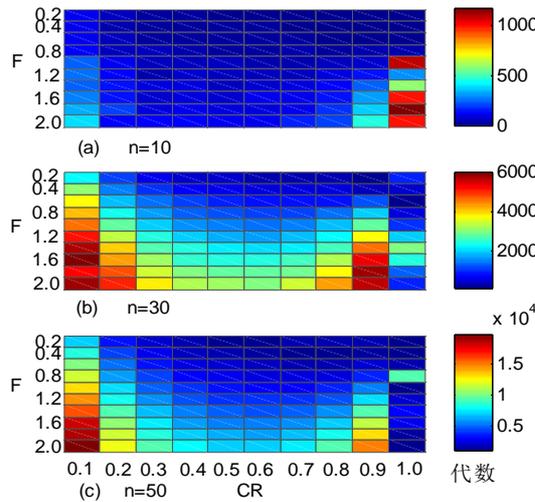


Figure 4. DE's performance with different parameter pairs  
图4. 不同变量规模、 $F$  和  $CR$  组合的 DE 算法优化性能

$F$  和  $CR$  的最佳取值分别为 0.2 和 0.9；变量数为 50 左右时， $F$  和  $CR$  的最佳取值分别为 0.4 和 0.8 等。需要注意的是， $CR = 1.0$  时，程序近似纯随机搜索、易破坏优良模式，搜索结果不稳定。

### 4.3. 不同变量规模和约束处理方法对算法性能的影响分析

分别采用 POA、GA 和 DE 通过约束惩罚方式对模型不同数量的优化变量进行优化。由于优化变量最长达 120 个，为便于比较，GA 交叉概率和变异概率

分别取值为 0.70 和 0.05；DE 采用 DE/BEST/1/EXP 模式，变异因子和交叉概率分别取值 0.4 和 0.8。对于不同的变量个数、算法和约束处理方法，分别运行 10 次，分别统计其最优值、运行步数和运行时间的平均值，结果如表 1 所示。

从表 1 中数据可以看到，结合单纯形法的 POA 传统优化算法在运行效率上远远高于 GA 和 DE，特别是随着变量个数的增加，DE 和 GA 优化效率下降明显，耗时增多和解的质量下降均较明显，而 POA 算法性能受变量规模增加的影响较小，能收敛到全局最优解，且计算的耗时增加不多。与 GA 相比，DE 收敛速度快，耗时较少，对于文中优化问题变量个数超过 70 时，难以搜索到可行解，而 GA 方法能够收敛到可行解，但变量规模大于 30 时搜索效率就开始明显降低，耗时较长且收敛速度慢。

对比表 1 中数据还可以看出，对于 GA 算法，当变量数较少时，对于不可行解采用修复策略较有效，耗时少且解的质量高；当变量数较多时，采用修复策略就较耗时，且难以显著提高解的质量。对于 DE 算法，无论优化变量规模大小与否，采用修复策略都显著减少了计算耗时，并明显提高了解的质量。当优化变量较少时(如不超过 20 个)，基于修复策略的 GA 和 DE 算法性能优越于 POA 算法，但是变量规模逐渐增大时，GA 和 DE 算法耗时都迅速增加，解的质量也会下降，尤其以 GA 算法性能下降最快，而结合单纯形法的 POA 性能比较稳定，无论是解的质量还是优化耗时都明显优于 GA 和 DE。

综合可以看出，GA 和 DE 智能算法有一定的适用性，在水库调度问题中，当变量规模不大时，GA 和 DE 等智能算法有一定的优越性，但当变量规模比较大时，传统算法 POA 在解的质量和耗时方面仍然具有优势。

## 5. 结论

通过多种方法的比较分析，可以得出以下两条主要结论：

- 1) 传统优化算法，尤其是结合其他非线性优化技术的 POA 算法，收敛速度快，并且随着问题优化规模的增加，耗时增加不多，所以是一种非常好的优化技术，特别适合于水库调度的复杂问题优化。

**Table 1. Comparison of several optimization methods in performance**  
**表 1. 不同算法性能比较表**

优化方法	变量个数	10	20	30	40	50	70	80	100	110	120
理论最优值		<b>36,880</b>	<b>37,940</b>	<b>41,340</b>	<b>44,505</b>	<b>46,404</b>	<b>48,574</b>	<b>49252.5</b>	<b>50,202</b>	<b>50,547</b>	<b>50,835</b>
	均值	<b>36,880</b>	<b>37,940</b>	<b>41,340</b>	<b>44,505</b>	<b>46,404</b>	<b>48,574</b>	<b>49252.5</b>	<b>50,202</b>	<b>50,547</b>	<b>50,835</b>
POA	步数	1	1	1	1	2	2	2	2	2	2
	时间(s)	0.03	0.06	0.14	0.17	0.28	0.41	0.50	0.47	0.62	0.73
GA (罚函数法)	均值	<b>36,880</b>	<b>37,940</b>	<b>41,340</b>	<b>44,503</b>	<b>46,401</b>	<b>48,572</b>	<b>49,249</b>	<b>50,192</b>	<b>50,532</b>	<b>50,810</b>
	步数	31	162	1425	3838	6570	13,857	16029	22,875	26,034	28,711
DE (罚函数法)	时间(s)	0.18	1.69	21.70	83.76	197.65	654.45	900.44	1700.30	2169.77	2610.02
	均值	<b>36,880</b>	<b>37,940</b>	<b>41,340</b>	<b>44,505</b>	<b>46,404</b>	<b>48,574</b>	-	-	-	-
DE (罚函数法)	步数	9	39	629	890	1144	1690	-	-	-	-
	时间(s)	0.05	0.36	8.80	18.23	31.83	76.06	-	-	-	-
GA (修复策略)	均值	<b>36,880</b>	<b>37,940</b>	<b>41,340</b>	<b>44,505</b>	<b>46,397</b>	<b>48,565</b>	<b>49,224</b>	<b>50,144</b>	<b>50,471</b>	<b>50,572</b>
	步数	1	1	263	1935	3878	7984	8443	15,345	14,579	11,877
DE (修复策略)	时间(s)	0.01	0.03	4.24	41.22	109.11	337.27	435.06	1715.83	2755.61	2641.10
	均值	<b>36,880</b>	<b>37,940</b>	<b>41,340</b>	<b>44,505</b>	<b>46,404</b>	<b>48,574</b>	<b>49252.5</b>	<b>50,202</b>	<b>50,547</b>	<b>50,713</b>
DE (修复策略)	步数	1	1	17	52	99	158	205	158	153	1169
	时间(s)	0.01	0.01	0.29	1.09	2.58	5.92	8.68	8.82	9.75	74.95

2) 智能优化算法, 都是基于群体的优化技术, 不同的算法搜索效率差异很大, 难点在于算子和参数组合的选择, 需要根据问题和算法的特点进行精心设计和选择。对于小规模优化计算, 通过选取合适的算子和参数, 智能优化算法 DE 从耗时等方面会优于传统优化技术; 但对于大规模的复杂优化问题, 传统优化算法 POA 在耗时和解的质量等方面仍然具有显著的优势。

**参考文献 (References)**

[1] 刘攀, 郭生练, 李玮, 易松松. 遗传算法在水库调度中的应用综述[J]. 水利水电科技进展, 2006, 26(4): 78-83.  
LIU Pan, GUO Shenglian, LI Wei and YI Songsong. A review of application of genetic algorithm to reservoir operation. Advances in Science and Technology of Water Resources, 2006, 26(4): 78-83. (in Chinese)

[2] 刘攀, 郭生练, 雒征, 等. 求解水库优化调度问题的动态规划 - 遗传算法[J]. 武汉大学学报(工学版), 2007, 40(5): 1-6.  
LIU Pan, GUO Shenglian, LUO Zheng, et al. Optimization of reservoir operation by using dynamic programming & genetic algorithm. Engineering Journal of Wuhan University, 2007, 40(5): 1-6. (in Chinese)

[3] HOWSON, H.R., SANCHO, N.G.F. A new algorithm for the solution of multi-state dynamic programming problems. Mathe-

tical Programming, 1975, 8(1): 104-116.

[4] 周佳, 马光文, 张志刚. 基于改进 POA 算法的雅砻江梯级水电站群中长期优化调度研究[J]. 水力发电学报, 2010, 29(3): 18-22.  
ZHOU Jia, MA Guangwen and ZHANG Zhigang. Study on the mid-long term optimal dispatching of cascaded hydropower stations on Yalong River based on POA modified adaptive algorithm. Journal of Hydroelectric Engineering, 2010, 29(3): 18-22. (in Chinese)

[5] 宗航, 周建中, 张勇传. POA 改进算法在梯级电站优化调度中的研究和应用[J]. 计算机工程, 2003, 29(17): 105-109.  
ZONG Hang, ZHOU Jianzhong and ZHANG Yongchuan. Research and application for cascaded hydroelectric optimized scheduling based on modified adaptive POA. Computer Engineering, 2003, 29(17): 105-109. (in Chinese)

[6] SPLENDY, W., HEXT, G.R. and HIMSWORTH, F.R. Sequential application of simplex design in optimization and evolutionary design. Technometrics, 1962, 4: 441-461.

[7] NELDER, J.A., MEAD, R. A simplex method for function minimization. Computer Journal, 1965, 7: 308-313.

[8] HOLLAND, J. Adaptation in natural and artificial systems. Ann Arbor: The University of Michigan Press, 1975.

[9] TAKAHASHI, M., KITA, H. A crossover operator using independent component analysis for real-coded genetic algorithm. Proceedings of the 2001 Congress on Evolutionary Computation, 2001: 643-649.

[10] DEB, K., AGRAWAL, R.B. Simulated binary crossover for continuous search space. Complex Systems, 1995, 9(6): 115-148.

[11] ONO, I., KITA, H. and KOBAYASHI, S. A robust real-coded genetic algorithm using unimodal normal distribution crossover augmented by uniform crossover: effects of self-adaptation of crossover probabilities. Proceedings of the Genetic and Evolu-

- tionary Computation Conference, San Mateo: Morgan Kaufmann Publishers, 1999: 496-503.
- [12] TSUTSUI, S., YAMAMURA, M. and HIGUCHI, T. Multiparent recombination with simplex crossover in real coded genetic algorithms. Proceedings of the Genetic and Evolutionary Computation Conference, San Mateo: Morgan Kaufmann Publishers, 1999: 657-664.
- [13] MICHALEWICZ, Z. Genetic Algorithms + Data Structures = Evolution Programs. New York: Springer-Verlag, 1996.
- [14] PRICE, K.V., STORN, R. Differential evolution—A simple evolution strategy for fast optimization. *Dr. Dobb's Journal*, 1997, 22: 18-24.
- [15] HOMAIFAR, A., LAI, S. and QI, X. Constrained optimization via genetic algorithms. *Simulation*, 1994, 62(4): 242-254.