

梯级水电站群长期发电优化调度多核并行 随机动态规划方法

王森, 程春田*, 武新宇, 李保健

大连理工大学水电与水信息研究所, 大连 116024

* E-mail: ctcheng@dlut.edu.cn

收稿日期: 2013-07-15; 接受日期: 2013-11-27

国家重点基础研究发展计划(“973”计划)(批准号: 2013CB035906)、国家自然科学基金(批准号: 51109024)和国家杰出青年科学基金(批准号: 51025934)资助项目

摘要 随机动态规划求解水电站群长期发电优化调度易产生“维数灾”问题, 导致计算耗时急剧增加, 求解效率降低. 如何缓解维数灾和提高计算效率, 一直是水库优化调度致力于研究的难点问题. 在随机动态规划的并行性分析基础上, 提出了基于 Fork/Join 并行框架的多核并行随机动态规划方法. 该方法将单个时段内所有变量组合状态下的计算任务作为父任务, 通过分治法递归分解为多个子任务, 并平均分配到不同的内核同时计算实现细粒度并行求解. 以澜沧江下游梯级水电站群为研究实例, 建立了 3 个变量离散数不同的调度方案, 并在多核环境下验证该方法的计算效率. 结果表明, 在 2 和 4 核环境下, 该方法的计算耗时与串行方法相比, 分别节省了约 50% 和 70%, 大幅度缩减计算耗时, 可充分利用多核资源; 同时, 计算任务的规模越大, 并行计算的耗时缩减幅度越大. 因此, 此方法为大规模水电系统优化调度提供了一种可行途径, 其并行原理可为其他应用所借鉴.

关键词

梯级水电站群
随机动态规划
并行
多核
优化调度

水电站群发电优化调度是系统工程研究领域较为复杂的问题之一. 调度过程可描述为: 根据水电站的入库径流过程, 利用最优化方法, 在满足水电站及系统相关约束条件的同时, 实现水电站群在整个调度期内的系统发电效益最大. 通过水电站群发电优化调度, 能够有效发挥水电站调蓄功能, 充分挖掘水资源蕴含的丰富水能使其转化为电能, 对电网安全稳定运行具有重要意义^[1].

水电站群优化调度问题具有大规模、非线性、高维数等特点^[2,3], 且电站之间存在复杂的水力和电力联系, 加剧了优化求解困难. 经过 70 多年的研究, 已

产生了很多好的优化方法. 郭生练等人^[2]和 Labadie^[3]对应用于水电站群优化调度的方法作了详细的评述与总结. 其中, 应用较为广泛的优化算法包括线性规划^[4]、非线性规划^[5]、动态规划^[6]以及智能算法^[7-9]等. 并且, 不同的调度问题对于方法的选择也存在一定差异, 选择的依据主要依赖于调度任务、数据完整性、优化目标以及约束条件等. 对于水电站群长期发电优化调度问题的研究, 按入库径流的处理方式主要分为确定性模型和随机模型. 确定性模型假定天然径流是已知和确定的; 随机模型考虑到天然径流不能准确预知, 把径流作为随机过程处理. 为了使数

引用格式: 王森, 程春田, 武新宇, 等. 梯级水电站群长期发电优化调度多核并行随机动态规划方法. 中国科学: 技术科学, 2014, 44: 209-218
Wang S, Cheng C T, Wu X Y, et al. Parallel stochastic dynamic programming for long-term generation operation of cascaded hydropower stations (in Chinese). Sci China Tech Sci (Chin Ver), 2014, 44: 209-218, doi: 10.1360/092013-800

学模型更接近调度实际, 需采用随机模型进行求解. 随机动态规划(Stochastic Dynamic Programming, SDP)在求解水电站群长期优化调度随机模型中应用较为广泛, 且国内外学者的相关研究成果较多^[10-12]. 动态规划方法最早由 Bellman^[13]提出, 是求解序贯决策过程最优化问题最常用的一种数学方法. 该方法把多阶段决策过程转化为一系列单阶段问题, 利用各阶段之间的关系, 通过计算单个阶段内所有离散变量的返回值, 以递推方式逐个求解. 变量离散数越多, 优化解的质量越高, 同时, 也意味着计算任务工作量越大, 计算耗时越多. 但是, 随着计算电站数目的增多, 易不可避免地造成“维数灾”问题^[2,3]. 对于随机模型, 由于考虑了径流的随机性, “维数灾”问题更加显著, 导致计算耗时急剧增加, 求解效率降低.

近年来, 并行技术已成为一种普遍认可的能够提高算法计算性能的有效手段, 并且, 在许多研究领域中已获得了成功的应用^[14-18]. 同时, 随着多核 CPU 的不断发展与更新, 为了避免处理器处于闲置状态, 串行方法的并行化已成为必然趋势. 考虑到 SDP 方法是通过对变量的离散实现优化解的迭代搜索, 算法本身具有天然的并行性. 因此, 在多核 CPU 环境下, 将并行技术与 SDP 方法相结合, 在保证计算结果与串行方法完全相同的前提下, 可以有效地减少计算耗时、提高求解效率.

为了提高随机动态规划的计算性能, 通过对随机动态规划的并行性分析, 提出了基于 Fork/Join 并行框架^[19]的多核并行随机动态规划(Parallel Stochastic Dynamic Programming, PSDP). 该方法将单个时段内所有变量组合状态下的计算任务作为父任务, 通过分治法递归分解为多个子任务, 并平均分配到不同的核心同时计算实现并行求解. 以澜沧江下游梯级水电站群为研究实例, 建立了 3 个变量离散数不同的调度方案, 并在多核环境下验证该方法的求解效率. 结果表明, 在多核环境下, 该方法能够充分利用多核资源, 计算加速比随着核数的增加而增加, 大幅度缩短计算耗时, 提高了求解效率, 避免了资源闲置.

本文结构分为 4 部分: 第 1 节描述了随机动态规划的数学模型, 包括余留效益函数递推方程、约束条件及计算收敛条件; 第 2 节分析了随机动态规划的并行性, 介绍了 Fork/Join 并行框架的原理, 并阐述了并行随机动态规划方法的求解步骤; 第 3 节以梯级水电站群长期发电优化调度问题为例验证了本文提出方

法; 第 4 节对全文进行了总结.

1 数学模型

对于 M 个计算电站, 周期含 T 个调度时段的梯级水电站群长期发电优化调度问题, 本文采用以下变量和符号定义.

t :	时段序号, $t=1, \dots, T$, 本文调度周期为 1 年, 即 $T=12$.
m :	电站序号, $m=1, \dots, M$.
y :	周期迭代次数, $y \geq 1$.
U :	单个时段内 M 维径流矢量的个数.
u :	径流矢量序号, $u=1, 2, \dots, U$.
V_t :	时段 t 的 M 维初始蓄水状态矢量.
Q_t^u :	时段 t 第 u 个 M 维径流矢量.
N_t :	时段 t 的 M 维决策出力矢量.
V_m^t :	水电站 m 时段 t 初始蓄水量.
Q_m^t :	水电站 m 时段 t 入库水量.
R_m^t :	水电站 m 时段 t 出库水量.
Qin_m^t :	水电站 m 时段 t 区间水量.
K_m :	水电站 m 的上游电站数.
k :	上游电站序号.
$R_{m,k}^t$:	水电站 m 时段 t 上游水库 k 的出库流量.
q_m^t :	水电站 m 时段 t 的发电流量.
d_m^t :	水电站 m 时段 t 的弃水流量.
$V_{m,\min}^t$:	水电站 m 时段 t 最小初始蓄水量.
$V_{m,\max}^t$:	水电站 m 时段 t 最大初始蓄水量.
$q_{m,\min}^t$:	水电站 m 时段 t 最小发电流量.
$q_{m,\max}^t$:	水电站 m 时段 t 最大发电流量.
$R_{m,\min}^t$:	水电站 m 时段 t 最小出库流量.
$R_{m,\max}^t$:	水电站 m 时段 t 最大出库流量.
$N_{m,\min}^t$:	水电站 m 时段 t 最小出力.
$N_{m,\max}^t$:	水电站 m 时段 t 最大出力.
$E_y(V_t)$:	迭代周期 y 时段 t 初始蓄水状态为 V_t 时, 从时段 t 到末时段的最大期望发电量.

$P(Q_t^u)$: 时段 t 入库流量 Q_t^u 的出现概率.
 $B_y(V_t, Q_t^u, N_t)$: 迭代周期 y 时段 t 初始蓄水状态为 V_t , 入库流量为 Q_t^u , 决策出力为 N_t 时的本时段发电量.
 $E_y(V_{t+1}(V_t, Q_t^u, N_t))$: 迭代周期 y 时段 t 初始蓄水状态为 V_t , 入库流量为 Q_t^u , 决策出力为 N_t 时, 计算得到的时段 t 末蓄水状态 V_{t+1} 所对应的余留期最大期望发电量.
 E^* : 周期最优期望效益.
 $N_y^*(V_t)$: 迭代周期 y 时段 t 初始蓄水状态为 V_t 的最优决策出力组合.
 $N^*(V_t)$: 计算收敛状态下时段 t 初始蓄水状态为 V_t 的最优决策出力组合.

1.1 余留效益函数递推方程

由于随机模型中考虑了入库径流的随机性, 因此, 随机模型的递推方程通过入库径流的出现概率对单阶段的最优值进行描述.

通常, 为了便于计算和分析, 入库径流可从时间和空间两方面进行描述. 在时间上, 相邻时段径流相互独立或存在相关关系; 在空间上, 同一时段各电站径流相互独立或存在相关关系. 本文只考虑空间相关关系, 每个时段的径流序列组合服从联合概率分布, 且相邻时段的入库径流过程相互独立. 余留效益函数逆序递推方程数学表达式如下:

$$E_y(V_t) = \max \left\{ \sum_{u=1}^U P(Q_t^u) [B(V_t, Q_t^u, N_t) + E_y(V_{t+1}(V_t, Q_t^u, N_t))] \right\}. \quad (1)$$

临界条件:

$$\begin{cases} E_y(V_{T+1}) = 0, & \text{如果 } y = 1, \\ E_y(V_{T+1}) = E_{y-1}(V_1), & \text{如果 } y > 1. \end{cases}$$

1.2 约束条件

1) 水量平衡约束

$$\begin{aligned} V_m^{t+1} &= V_m^t + Q_m^t - R_m^t, \\ Q_m^t &= Qin_m^t + \sum_{k=1}^{K_m} R_{m,k}^t, \\ R_m^t &= (q_m^t + d_m^t) \times \Delta_t \times 3600. \end{aligned} \quad (2)$$

2) 蓄水量约束

$$V_{m,\min}^t \leq V_m^t \leq V_{m,\max}^t. \quad (3)$$

3) 发电流量约束

$$q_{m,\min}^t \leq q_m^t \leq q_{m,\max}^t. \quad (4)$$

4) 出库约束

$$R_{m,\min}^t \leq R_m^t \leq R_{m,\max}^t. \quad (5)$$

5) 出力约束

$$N_{m,\min}^t \leq N_m^t \leq N_{m,\max}^t. \quad (6)$$

1.3 收敛条件

根据 Su 和 Deininger^[20]证明, 对(1)式无限逆序递推, 相邻周期相同时段相同蓄水状态的余留期效益之差趋于稳定值, 即周期最优期望效益, 见(7)式. 同时, 周期内各时段各蓄水状态的最优决策出力趋于稳定值, 见(8)式.

$$\lim_{y \rightarrow \infty} [E_y(V_t) - E_{y-1}(V_t)] = E^*, \quad (7)$$

$$\lim_{y \rightarrow \infty} N_y^*(V_t) = N^*(V_t). \quad (8)$$

本文通过离散决策出力搜索各蓄水状态下的最优决策出力, 因此, 根据(8)式, 当相邻周期任意时段相同蓄水状态下的最优决策出力均相同时, 则计算达到收敛, 见(9)式.

$$N_y^*(V_t) = N_{y-1}^*(V_t). \quad (9)$$

2 并行随机动态规划

2.1 SDP 并行性分析

SDP 方法求解梯级水电站群长期发电优化调度问题获得的最终成果是各时段各蓄水状态组合下的最优决策出力组合. 由于考虑了入库径流的随机性, 需对电站蓄水状态、入库径流以及决策出力进行离散, 计算各组合状态下的梯级水电站群期望发电量, 并逐时段递推寻优, 直至相邻调度周期同时段同蓄水状态组合下的最优决策出力组合均相同.

为了更好地反映 SDP 方法计算任务的密集型特点, 以单个水库的余留效益函数递推最优决策出力过程为例, 见图 1. 其中, 蓄水状态离散个数 L , 入库流量离散个数 G , 决策出力离散个数 C . 以图 1 时段 t 初始蓄水状态 V_t^i 为例, 对于每个离散的决策出力 (如图 1 实线表示 N_t^i 、虚线表示 N_t^C), 在不同的离散

入库径流下, 均需返回 G 个不同的结果, 即需完成 G 次计算任务, 才可求得初始蓄水状态 V_t^i 决策出力 N_t^i (或 N_t^c) 的全时期 (时段 t 到时段 T) 期望发电量. 因此, 根据余留效益函数递推公式(1), 求得初始蓄水状态 V_t^i 的最优决策出力需完成 $G \times C$ 次计算任务, 则对于单个时段内所有蓄水状态的最优决策出力, 需完成 $L \times G \times C$ 次计算任务. 因此, 求解计算水电站数目为 M 的水电系统, 获得单个时段内所有蓄水状态的最优决策出力需计算 $(L \times G \times C)^M$ 次. 若蓄水状态离散个数 20, 入库流量离散个数 10, 决策出力离散个数 10, 求解 1 个水电站单个时段内所有蓄水状态的最优决策出力则需完成 2×10^3 次计算任务, 2 个水电

站则为 4×10^6 次, 计算任务呈指数迅速增长, 突显“维数灾”问题.

多核并行是指两个或者两个以上不同的任务可同一时刻在不同的 CPU 内核上同时执行, 以缩短计算时间, 提高求解效率. 因此, 并行计算必须满足 2 个条件. 1) 计算机配置具有多个 CPU 内核; 2) 不同的计算任务之间必须相互独立. 并行计算的工作流程见图 2.

目前, 不仅对于服务器或者工作站, 即使对于个人电脑, 多核 CPU 配置已非常普遍, 为并行计算的广泛应用提供了强有力的硬件支持; 同时, 根据对 SDP 方法计算任务的分析, 单个时段内每次计算任务是蓄水状态、入库径流、决策出力三者状态组合下

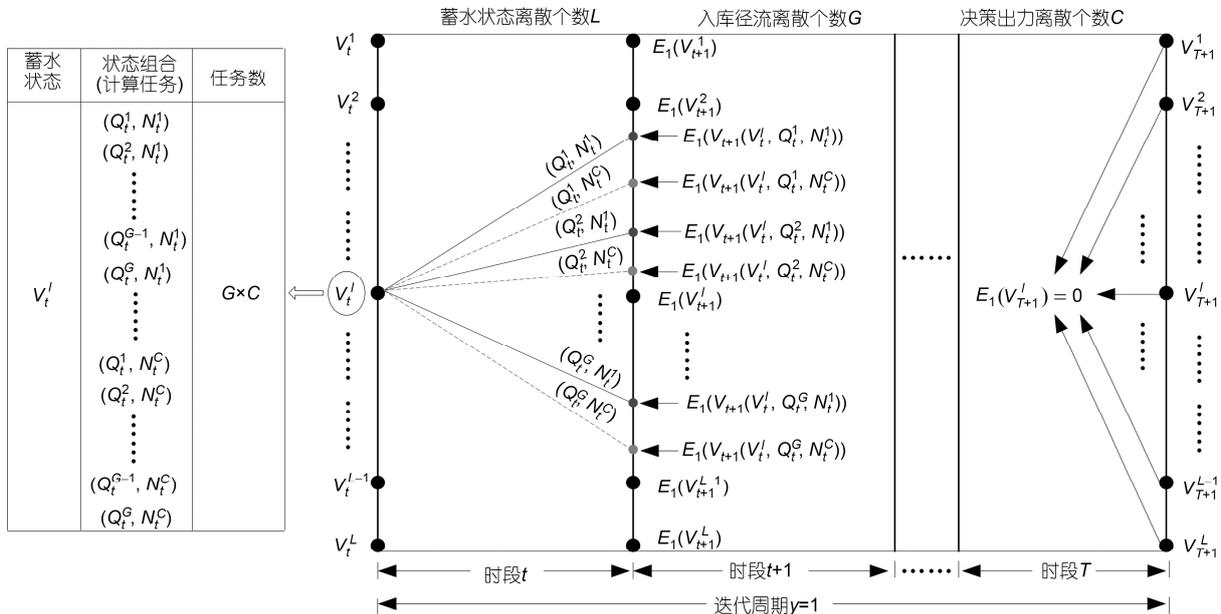


图 1 余留效益函数递推最优决策出力过程示意图

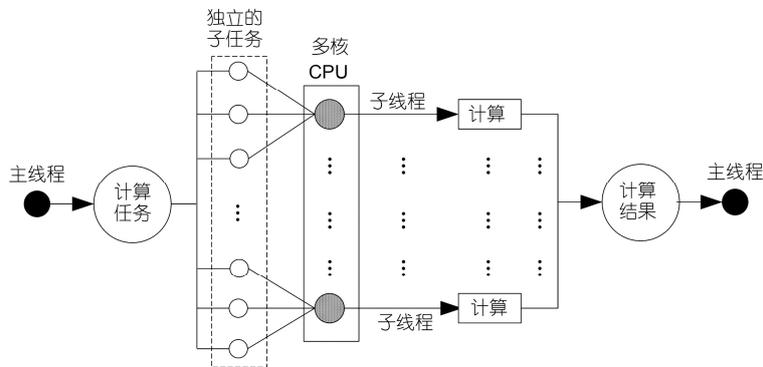


图 2 并行计算的工作流程图

的期望发电量, 显然, 各次计算任务之间相互独立. 因此, 在递推时段 t 各水位组合状态下的最优出力组合时, 可运用并行技术使计算任务同一时刻在不同 CPU 上同时执行, 以缩短计算时间, 提高求解效率.

2.2 Fork/Join 并行框架

Fork/Join 是一种基于多核并行且适合计算密集型任务的并行框架. 该框架的核心方法采用“分治法”实现, 即将父任务递归分解成若干个子任务, 再在并行环境中处理各个子任务, 最后将子任务的结果组合, 返回最终结果. “分治法”递归分解计算任务的过程见图 3. 由于“分治模式”实现了父任务的层层递归分解, 因此, Fork/Join 实现了更细粒度的并行化.

Fork/Join 通过一个阈值控制子任务的数目, 当任务的计算规模小于或等于阈值时, 则停止任务分解. 阈值的大小对并行效率具有较大的影响. 若阈值过小, 则子任务数目较多, 任务管理消耗加大; 若阈值过大, 则子任务数目较少, 无法充分利用多核资源. 因此, 为了避免资源闲置, 合适的阈值可按(10)式置:

$$\beta = \left\lceil \frac{f}{w} \right\rceil, \quad (10)$$

其中符号 $\lceil \cdot \rceil$ 表示取上整数; β 表示阈值; f 表示任务的计算规模; w 表示 CPU 逻辑线程数, 一般, CPU 逻辑线程数与内核数相同, 若 CPU 处理器支持“超线程”技术(即利用特殊的硬件指令, 让单个内核具有 2 个逻辑线程计算), 则 CPU 逻辑线程数等于内核数的 2 倍. 目前, “超线程”技术在 CPU 配置中比较普遍, 但是, “超线程”技术只能在一定意义上提高 CPU 的计算性能, 并不能真正达到内核的并行计算效果.

此外, 在 Fork/Join 开始运行时, 默认创建与 CPU 逻辑线程数目相同的工作线程, 并采用线程池进行管理, 避免了反复创建线程和关闭线程带来的

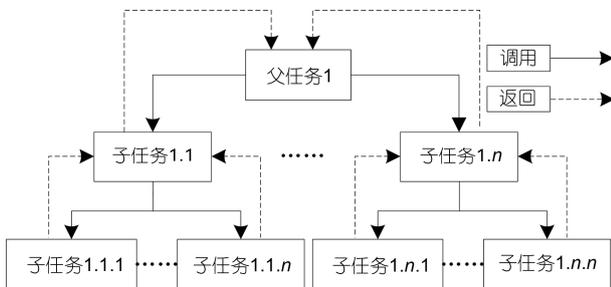


图 3 分治法递归分解过程示意图

巨大开销, 节省了系统资源. 并且, Fork/Join 采用了“工作窃取”技术. 当线程的任务队列为空, 它将尝试从另一个线程的双端队列尾部窃取另一个任务, 减少了工作队列争用, 使工作线程尽量一直处于工作状态, 提高了工作线程的利用率.

2.3 PSDP 算法设计

时间复杂度是指在计算机科学与工程领域完成一个算法所需要的时间, 是衡量一个算法优劣的重要标志. 根据 2.1 节 SDP 的并行性分析可知, 可将推求单个时段内所有蓄水状态组合下的最优决策出力组合需完成的计算任务作为父任务, 计算规模为 $(L \times G \times C)^M$, 则时间复杂度为 $O((L \times G \times C)^M)$. 因此, 通过 Fork/Join 并行框架将单个时段内的父任务递归分解为多个子任务, 平均分配到不同的内核上同时进行计算, 可实现 SDP 的并行计算. 若 CPU 内核数为 p , 逻辑线程数为 w , 则每个子任务的理想计算规模为 $(L \times G \times C)^M/w$, 但并行算法的理想时间复杂度只与内核数有关, 为 $O((L \times G \times C)^M/p)$. 因此, 通过并行处理, SDP 的计算效率理论上可随着核数的增加而增加. PSDP 详细计算步骤如下所述.

Step 1. 数据准备. 读取计算相关的静态数据, 包括电站基础属性、特征曲线以及约束条件等, 储存在主线程中, 用于各内核共同调用.

Step 2. 参数初始化. 蓄水状态离散个数 L , 入库流量离散个数 G , 决策出力离散个数 C ; 初始迭代周期 $y=1$, 初始计算时段 $t=12$.

Step 3. 创建线程池, 默认生成的线程池的工作线程数目与 CPU 逻辑线程数目相同.

Step 4. 采用余留效益函数逆序递推迭代周期 y 时段 t 不同蓄水状态组合下的最优出力组合, 并且把当前时段内不同变量组合状态下的计算任务作为父任务, 通过 Fork/Join 实现并行计算. 并行阶段计算步骤如下所示.

1) 根据参与计算的电站数目以及各个变量的离散数, 确定推求单个时段内各蓄水状态组合下的最优出力组合所需要的计算次数.

2) 确定父任务递归分解方式以及阈值大小. 为了便于父任务的递归分解, 可采用二分法递归分解, 即每次父任务的递归分解产生两个计算规模相同的子任务; 根据(10)式计算阈值大小, 当子任务的计算规模小于或等于阈值时, 停止任务的递归分解, 分配

各个子任务在不同的线程上同时进行计算。

3) 各子线程获取分配的子任务. 为了确保各线程之间的计算不会出现数据相关, 需保证各线程之间的计算数据相互独立. 计算数据主要包括静态数据与动态数据. 静态数据不会造成子线程之间数据相关, 直接从主线程中调用; 动态数据包括电站的计算水位、出力以及各状态组合的期望发电量结果集等等, 则在各个线程中单独定义。

4) 计算各状态组合的期望发电量. 为了满足水电系统保证出力的保证率要求, 引入罚函数求解各状态组合的期望发电量, 对不满足水电系统保证出力的决策出力组合给予一定的惩罚。

5) 各子线程的返回值储存在各自的结果集中, 当工作线程完成所有子任务的计算, 关闭工作线程。

6) 将各子线程的结果集进行合并, 并行计算结束, 返回主线程。

Step 5. 根据主线程获取的合并结果集, 求得当前时段不同水位组合状态下的最优出力组合以及最大期望发电量。

Step 6. 判断周期是否全部计算. 若当前时段 $t \neq 1$, 则 $t=t-1$, 并返回 Step 4; 若 $t=1$, 则进入下一步。

Step 7. 判断是否满足收敛条件. 比较当前迭代周期 y 与前一迭代周期 $y-1$ 任意时段相同蓄水状态下的最优决策出力是否均相同. 若不满足要求, 则 $y=y+1$, $t=T$, 返回 Step 4; 若满足要求, 则计算收敛, 输出计算结果。

PSDP 方法计算流程见图 4。

3 应用实例

3.1 工程背景

为了验证算法的可行性和有效性, 本文以澜沧江下游梯级水电站群为研究实例, 对 PSDP 算法的计算性能进行测试. 澜沧江流域位于云南省境内, 水能资源丰富, 开发条件优越, 是我国十三大水电基地之一, 其干流总装机容量约为 25750 W, 年发电量约为 1167.45 亿 kWh. 其中, 澜沧江下游重要电站主要包括: 小湾、漫湾、大朝山、糯扎渡和景洪. 小湾和糯扎渡为巨型多年调节水电站, 小湾调节库容 99.0 亿 m^3 , 糯扎渡调节库容 113.3 亿 m^3 , 对澜沧江下游梯级水电站群补偿调度发挥重要作用. 水电站群拓扑结构见图 5, 各电站特征参数信息见表 1。

3.2 计算条件

1) 变量离散

采用 1953~2008 年的径流资料统计各电站各月份入库流量组合的联合分布, 且相邻时段径流看作相互独立, 各电站入库流量离散 G 份, 根据入库流量组合的联合分布得到各离散入库流量组合的概率; 各电站水位从正常高水位(汛期为汛限水位)到死水位离散 L 份; 各电站决策出力按装机容量离散 C 份. 为了测试 PSDP 的并行效率以及变量离散数对优化结

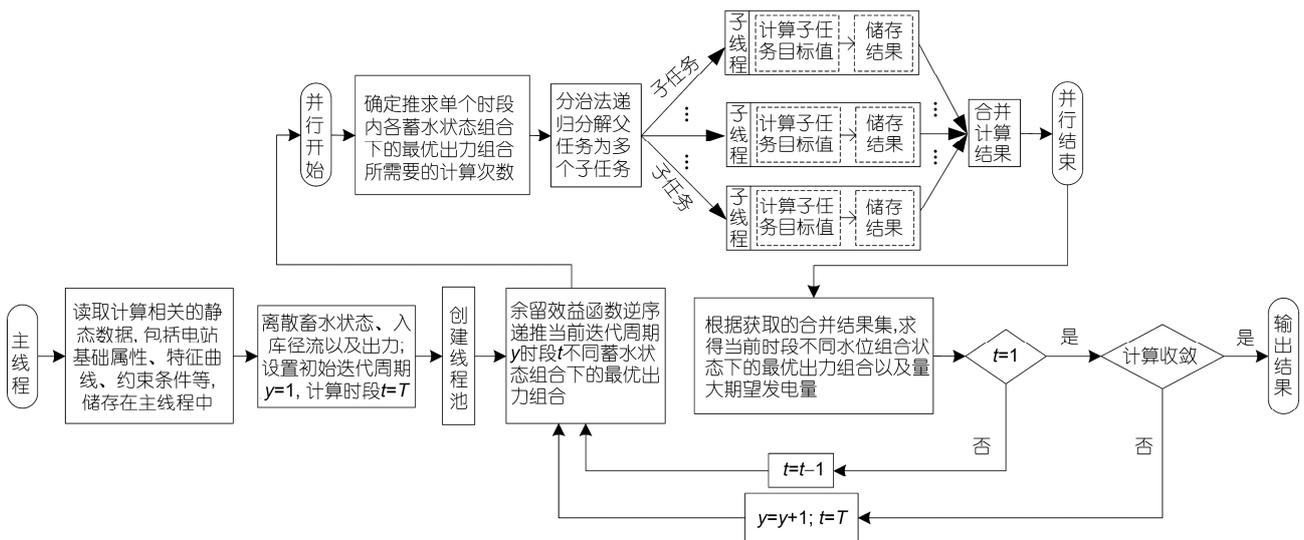


图 4 PSDP 算法计算流程图

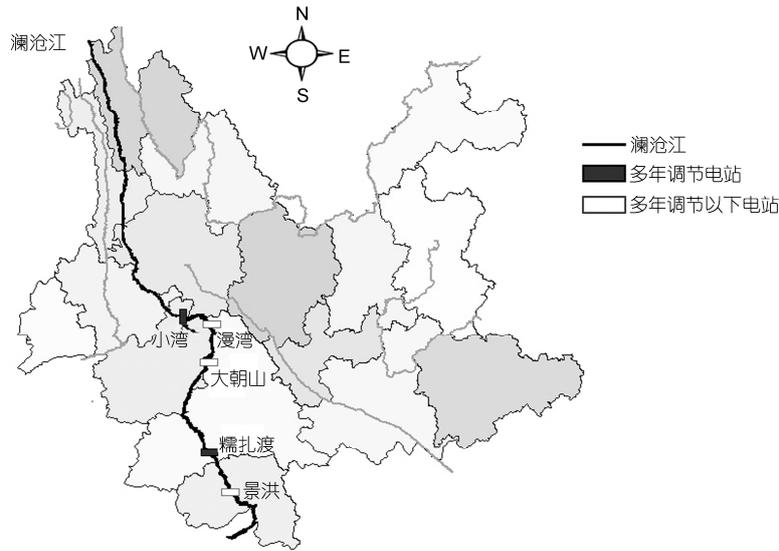


图5 澜沧江下游梯级水电站群拓扑结构图

表1 电站特性参数信息

电站	调节性能	正常高水位 (m)	死水位 (m)	装机容量 (MW)	调节库容 (亿 m ³)
小湾	多年调节	1240.00	1166.00	4200	99.0
漫湾	季调节	994.00	982.00	1550	2.58
大朝山	年调节	899.00	882.00	1350	3.7
糯扎渡	多年调节	812.00	765.00	5850	113.3
景洪	季调节	602.00	591.00	1750	3.1

果的影响, 建立 3 个含不同变量离散数的调度方案, 即设定不同的 G, L, C 进行仿真计算. 各调度方案的变量离散数设置见表 2.

2) 模型简化

从表 1 中可以看到, 小湾和糯扎渡两座巨型水电站的调节库容远大于其他 3 座水电站. 因此, 为了简化计算, 在 PSDP 求解该梯级水电站群长期发电优化调度问题时, 仅对小湾和糯扎渡电站进行变量离散参与优化计算, 其他电站保持高水头发电, 通过梯级水电站群补偿调度满足系统发电保证率要求(保证出力为 6500 MW).

3) 并行性能指标

评价并行计算性能的两个重要指标分别是加速比 S_p 和效率 E_p , 数学表达式如下.

表2 各调度方案变量离散数设置

	G	L	C
方案 1	6	20	10
方案 2	6	20	15
方案 3	6	30	15

$$S_p = T_1 / T_p, \tag{11}$$

$$E_p = S_p / p, \tag{12}$$

其中 T_1 为单核环境下的运行时间; T_p 为 p 个内核环境下的运行时间.

4) 编程语言及 CPU 配置

本文计算方法采用 java 编程语言实现. 采用的主机类型为 Dell Precision WorkStation T1600, CPU 类型为 Intel(R) Xeon(R) E3-1245@3.30 GHz, 4 核 8 线程 (“超线程”).

3.3 结果分析

为了分析不同内核数对 PSDP 计算效率的影响, 在 1 核(串行)、2 核、4 核环境下对不同调度方案进行测试, 计算结果见表 3. 其中多年平均发电量以及梯级保证出力的保证率根据 PSDP 优化结果对 1953~2008 年模拟调度计算得到.

从表 3 中看到, 3 个调度方案优化结果的梯级保证出力的保证率均大于 95%, 满足梯级保证出力的保证

率要求,但是,随着变量离散数的增加,梯级保证出力的保证率影响并不大;多年平均发电量随着变量离散数的增加而增加,但是,增加的幅度并不明显,方案1多年平均发电量663.33亿kWh,方案2为663.96亿kWh,方案3为664.22亿kWh,方案3仅比方案1增加了0.89亿kWh.从串行计算时间角度分析,方案1计算耗时8768s,方案2计算耗时20208s,方案3计算耗时59200s,方案3计算耗时较长,是方案2的2.9倍,是方案1的6.8倍,变量离散数的增加对计算耗时的影响非常显著.

另外,在表3中,通过并行技术与SDP相结合,计算耗时显著下降.与串行方法相比,2核环境下,方案1减少了4317s,方案2减少了10002s,方案3减少了29453s;4核环境下,方案1减少了6312s,方案2减少了14862s,方案3减少了43745s.因此,在并行环境下,随着计算任务的规模逐渐加大,计算耗时减少幅度越大,并行计算的优势越明显.

图6(a)和(b)分别表示3个调度方案在不同环境下的计算加速比和效率.图6(a)中,各方案在2核环境下加速比均达到1.95以上,4核环境下均达到3.50以上,获得了较好的加速比,且加速比随着内核数的增加而

增加,充分利用了内核资源,避免了资源闲置.同时,在相同的并行环境下,计算任务的规模越大,计算加速比越大,且越接近于理想加速比.例如,在4核环境下,理想加速比为4,方案1计算加速比为3.57,方案2计算加速比为3.78,方案3计算加速比达到3.83.但是,对于同一计算任务,随着内核数的增加,计算加速比与理想加速比的差距逐渐加大,计算效率(见图6(b))逐渐下降.例如,方案3在2核环境下加速比1.99,与理想加速比之差为0.01;在4核环境下加速比3.83,与理想加速比之差为0.17.造成这一现象的主要原因有3点.1)PSDP算法在计算单个时段所有蓄水状态组合的最优出力组合时,采用并行计算,其他步骤为串行计算.2)随着内核数的增加,逻辑线程数增加,线程池对工作线程的管理(子任务划分、启闭等)消耗增加.3)由于并行计算须在各子线程中单独定义动态数据,计算机内存占用量增大,在一定程度上影响了CPU计算性能.

综上所述,将并行技术与SDP相结合,减少了计算耗时,提高了算法的求解效率;同时,随着多核CPU的不断更新,PSDP方法的加速比可能进一步提高.

表3 多核环境下各调度方案串并行结果对比

调度方案	多年平均发电量(亿kWh)	梯级保证出力保证率	内核数	计算时间(s)		加速比			理想与并行之差	效率	
				串行	并行	串行	并行	理想		串行	并行
方案1	663.33	96.8%	1	8768	-	1.00	-	-	-	100.0%	-
			2	8768	4451	1.00	1.97	2	0.03	50.0%	98.5%
			4	8768	2456	1.00	3.57	4	0.43	25.0%	89.3%
方案2	663.96	97.0%	1	20208	-	1.00	-	-	-	100.0%	-
			2	20208	10206	1.00	1.98	2	0.02	50.0%	99.0%
			4	20208	5346	1.00	3.78	4	0.22	25.0%	94.5%
方案3	664.22	96.4%	1	59202	-	1.00	-	-	-	100.0%	-
			2	59202	29749	1.00	1.99	2	0.01	50.0%	99.5%
			4	59202	15457	1.00	3.83	4	0.17	25.0%	95.8%

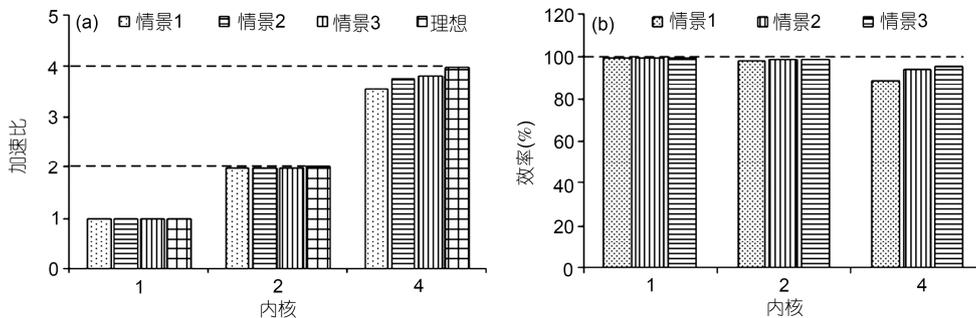


图6 多核环境下各调度方案计算加速比(a)及效率(b)对比

4 结论

随着我国西南地区特大流域梯级水电站群大批建成, 梯级水电站群精细化调度需求将更为强烈, 急需构建更加高效、科学的优化调度模型. 本文以澜沧江下游梯级水电站群为研究实例, 构建了 Fork/Join 并行框架下的梯级水电站群长期发电优化调度的多核并行随机动态规划方法, 在多核环境下实现了随机动态规划的优化调度并行计算, 并分析了该方法的计算效率. 结果表明以下 4 点.

1) 计算任务的规模越大, 计算耗时减少幅度越大, 并行计算的优势越显著, 这对于我国特大流域梯级水电站群调度非常重要, 因为这些流域干流梯级

十多级电站较为常见, 且有很多调节性能较好的水电站.

2) 随着 CPU 内核数的增加, 计算加速比逐渐增大, 计算耗时更小, 说明在现有多核计算机普及情况下, 不需要增加硬件投资, 多核技术并行化是提高系统计算效率的可行途径.

3) 在相同的多核 CPU 环境下, 计算任务的规模越大, 计算加速比与理想加速比之差越小, 这说明, 充分利用多核技术对于大规模优化调度问题更为显著.

4) 在不同的多核 CPU 环境下, 计算任务的计算加速比与理想加速比的差距随着核数的增加逐渐加大, 计算效率逐渐下降.

参考文献

- 程春田, 申建建, 武新宇, 等. 大规模复杂水电优化调度系统的实用化求解策略及方法. 水利学报, 2012, 43: 785-795
- 郭生练, 陈炯宏, 刘攀, 等. 水库群联合优化调度研究进展与展望. 水科学进展, 2010, 21: 496-503
- Labadie J W. Optimal operation of multireservoir systems: State-of-the-art review. J Water Res Pl, 2004, 130: 93-111
- Azamathulla H M, Wu F C, Ghani A A, et al. Comparison between genetic algorithm and linear programming approach for real time operation. J Hydro-Environ Res, 2008, 2: 172-181
- Barros M T L, Tsai F T C, Yang S L, et al. Optimization of large-scale hydropower system operations. J Water Res Pl, 2003, 129: 178-188
- Zhao T T G, Cai X, Lei X, et al. Improved dynamic programming for reservoir operation optimization with a concave objective function. J Water Res Pl, 2012, 138: 590-596
- Moeini R, Afshar M H. Extension of the constrained ant colony optimization algorithms for the optimal operation of multi-reservoir systems. J Hydroinform, 2013, 15: 155-173
- Zhang R, Zhou J Z, Ouyang S, et al. Optimal operation of multi-reservoir system by multi-elite guide particle swarm optimization. Int J Elec Power, 2013, 48: 58-68
- Darlane A, Momtahan S. Optimization of multireservoir systems operation using modified direct search genetic algorithm. J Water Res Pl, 2009, 135: 141-148
- Santiago C, Jesus M L, Andres R. Stochastic dual dynamic programming applied to nonconvex hydrothermal models. Eur J Oper Res, 2012, 218: 687-697
- Peng Y, Wang G L, Tang G L, et al. Study on reservoir operation optimization of Ertan hydropower station considering GFS forecasted precipitation. Sci China Tech Sci, 2011, 54: 76-82
- Sangamreddi C, Umamahesh V N. Comparison of stochastic and fuzzy dynamic programming models for the operation of a multipurpose reservoir. Water Environ J, 2011, 25: 547-554
- Bellman R E. Dynamic Programming. Princeton: Princeton University Press, 1957
- Zheng G N, Yang G W, Qian W. Flutter analyses of complete aircraft based on hybrid grids and parallel computing. Sci China Tech Sci, 2013, 56: 398-404
- Tang Y H, Zhang B D, Wu J J, et al. Parallel architecture and optimization for discrete-event simulation of spike neural networks. Sci China Tech Sci, 2013, 56: 509-517
- Jiang Q Y, Jiang H. OpenMP-based parallel transient stability simulation for large-scale power systems. Sci China Tech Sci, 2012, 55: 2837-2846
- 杨华勇, 宋月超, 徐兵. 复杂工况下柱塞泵流量脉动并行仿真与试验研究. 中国科学: 技术科学, 2012, 42: 1463-1471
- 王则力, 樊建人, 罗坤. 采用内嵌边界方法模拟流体中颗粒运动的并行算法. 中国科学 E 辑: 技术科学, 2009, 39: 642-647
- Lea D. A Java fork/join framework. In: Proceedings of the ACM 2000 conference on Java Grande. San Francisco, 2000. 36-43
- Su S Y, Deininger R A. Generalization of White's method of successive approximation to periodic Markovian decision processes. Oper Res, 1972, 20: 318-326

Parallel stochastic dynamic programming for long-term generation operation of cascaded hydropower stations

WANG Sen, CHENG ChunTian, WU XinYu & LI BaoJian

Institute of Hydropower and Hydroinformatics, Dalian University of Technology, Dalian 116024, China

Stochastic Dynamic Programming (SDP) for long-term generation operation of cascaded hydropower stations will bring about the curse of dimensionality, resulting in the rapid increase of computational time and the decrease of computational efficiency. Therefore, alleviate the dimensionality problem and improve the computational efficiency are always difficult issues for long-term generation operation of cascaded hydropower stations. On the basis of the parallelism analysis for SDP, a parallel stochastic dynamic programming (PSDP) based on Fork/Join parallel framework was proposed. In this method, all computational tasks for the returns from all discrete combinations in one stage were taken as parent task, which was decomposed into several subtasks by divide-and-conquer method. After this, the decomposed subtasks were solved in different cores respectively for achieving fine-grain parallel computation. The proposed approach was implemented to long-term generation operation of cascaded hydropower stations located on Lancangjiang River, and 3 different schemes with different discrete number of variables were established for testing the computational efficiency in multi-core environment. The result shows that the computational time, compared with serial computation, decreased respectively about 50% in 2-core environment and 70% in 4-core environment, making full use of multi-core resources. In addition, the larger computational scale can reduce more computational time in multi-core environment. Hence, the proposed approach is effective for operation of large-scale hydropower system, and can provide guidance for other applications.

cascaded hydropower stations, stochastic dynamic programming, multi-core, parallel, optimization

doi: 10.1360/092013-800