文章编号：1008－8857(2022)01－0046－07　**DOI**: 10.13259/j.cnki.eri.2022.01.007

基于改进萤火虫算法的分布式能源供应链配置研究

潘冯超，刘勤明，叶春明，刘文溢
（上海理工大学 管理学院，上海 200093）

摘　要：针对分布式能源供应链的配置问题，提出了改进人工萤火虫算法，结合云计算技术解决该配置问题。首先，以人工萤火虫算法的决策域半径为切入点，改进人工萤火虫算法的决策域半径，有效地解决了传统人工萤火虫算法寻优不稳定、算法精度低、后期收敛速度较慢的缺点；其次，全面采集系统信息，考虑各云处理中心各服务器的负载情况，建立基于改进人工萤火虫算法的分布式能源供应链配置需求侧均衡模型，以达到云计算环境下能源供应链中的配置均衡目标；最后，仿真分析表明，改进人工萤火虫算法可以更快、更稳定、更均衡地处理系统中的任务，优化分布式能源供应链配置。

关键词：人工萤火虫算法；云计算；分布式能源；供应链；决策域
中图分类号：TP391　　**文献标志码**：A

Research on distributed energy supply chain configuration based on improved glowworm swarm optimization algorithm

PAN Fengchao，LIU Qinming，YE Chunming，LIU Wenyi
(Business School, University of Shanghai for Science and Technology, Shanghai 200093, China)

Abstract: Aiming at the configuration problem of distributed energy supply chain, an improved glowworm swarm optimization algorithm was proposed, which was combined with cloud computing technology to solve the configuration problem. Firstly, the radius of decision domain of glowworm swarm optimization algorithm was taken as the cut-in point to improve its radius of this algorithm, which could effectively solve the shortcomings of traditional glowworm swarm optimization algorithm such as unstable optimization, low accuracy, and slow convergence rate in

收稿日期：2019－05－12
基金项目：国家自然科学基金项目(71840003、71471116、71632008)；教育部人文社会科学研究青年基金项目(15YJCZH096)
第一作者：潘冯超(1994—)，男，硕士研究生。研究方向：分布式能源供应链。
E-mail：14280250@qq.com
通信作者：刘勤明(1984—)，男，副教授。研究方向：维护调度、人工智能等。
E-mail：qmliu@usst.edu.cn

the later stage. Secondly, the system information was collected comprehensively. And the load of servers in cloud processing centers was taken into accounts. The demand-side balanced model of distributed energy supply chain configuration was established based on the improved glowworm swarm optimization algorithm to achieve the goals of balanced allocation in the energy supply chain under cloud computing environment. Finally, the simulation results showed that the improved glowworm swarm optimization algorithm could deal with the tasks of distributed energy supply chain faster, more steadily, and in equilibrium, and thus optimize its allocation.

Keywords: glowworm swarm optimization algorithm; cloud computing; distributed energy; supply chain; decision domain

随着中国能源结构的调整，分布式能源的比重不断增加，各种新兴能源的兴起，我国对于分布式能源供应链系统越来越重视。目前我国已进入实质开发实施阶段的分布式能源主要为风能、光能等。在我国不断加快的智能电网建设中，有效解决分布式能源供应链配置问题具有十分重要意义。

分布式能源可以缓解环境与能源需求、经济发展之间的矛盾，国内许多专家为此做了许多研究。Wu 等[1]通过一个基于代理的模拟，评估市场调节、能源消耗和能源容量在分布式能源系统低碳转型中的作用。董明[2]针对规模庞大、复杂且随机的能源网络提出分布式能源供应链网络的建模框架。邢海军等[3]对主动配电网模型的多阶段性规划和不确定性规划进行了分析，总结了主动配电网和传统配电网技术特点的差异，但系统还不够完善，后期还有很大的改进空间。李燕等[4]在解决主动配电网扩展规划问题中运用了不确定网络理论，并对二阶锥规划问题进行求解，实验结果证明了所建模型的有效性，但其模型难以应对大规模优化问题。随权等[5]提出了基于两阶段鲁棒风险偏好模型的主动配电网经济调度策略。Kandil 等[6]提出了多种分布式能源存储系统的最优资源配置的组合模型公式，分析了马尔可夫链蒙特卡罗仿真模型。田冬冬等[7]分析和研究了基于云计算智能电网的能源监控管理的效益和风险。马俊[8]利用云计算模型架构记录分析了云计算环境下资源能源的利用率情况。林君豪等[9]、马瑞等[10]利用蝙蝠算法、鲁棒优化理论、遗传算法等对分布式发电系统双层优化模型进行了求解。李军祥等[11]分析了云计算和分布式联络中心的发展趋势并在分布式联络中心的部分服务中运用了云计算技术。李瑞婷等[12]运用了改进的萤火虫算法求解 TFT−LCD 单元的装配调度问题，分析了不同学习音质和遗忘率对所求目标函数的影响。单好民[13]将云计算环境下资源调度模型中的萤火虫算法对应云计算资源节点，在个体初始化中引入遗传算法优化初始解，模型仿真结果表明该模型提高了资源调度性能，但模型相对复杂，计算所需时间较长，无法应对需快速反应的突发状况。目前对于分布式能源供应链及主动配电网系统资源配置问题的研究多集中在管理规划、完善调度模型、多目标调度等方面，很少有从分布式能源供应链需求侧配置角度对用户用电量与供电量均衡模型进行的研究。

综上分析，本文从提高资源利用效率、节能和经济等方面出发，将人工萤火虫优化算法应用于分布式能源供应链需求侧配置中，以决策半径为切入点，改进人工萤火虫算法决策半径，相对全面地研究平衡指标，建立基于改进人工萤火虫算法的分布式能源供应链配置需求侧均衡模型，在云计算环境下进行仿真实验，以期提高能源供应链配置效率。

1　问题描述

分布式能源供应链利用先进的计算机技术、智能网络监控、远程控制技术实现智能管理，不需要人工进行值守。未来能源社会的服务体系将实现专业化的管理，保证了每个能源系统的稳定

运行。为了满足这一需求，本文提出了基于改进人工萤火虫算法的分布式能源供应链配置模型，并在云计算环境下运行，解决了云计算基础下的分布式能源供应链配置均衡问题。该方法克服了传统人工萤火虫算法后期精度不高、收敛速度慢的缺点，全面考虑能源节点的资源分配，建立能源资源分配均衡模型，实现分布式能源供应链的均衡配置。通过模型验证，该方法能有效提高能源利用率，并在短时间内达到较优的平衡。

2 数学模型

2.1 模型构架设计

根据分布式能源供应链配置模型各部分设计，该模型构架主要由分布式能源节点、能源运处理中心、用户端三部分组成。模型结构如图 1 所示。假设能源云处理中心有 M 个物理机，每个物理机中有 N 个虚拟机，虚拟机使用 MapReduce 模型[14–15]将用户能源需求的总任务分成 X 个互相独立的子任务，并读取每个服务器的信息。每个子任务由一个或多个用户能源需求组成，系统将子任务分配到合理的服务器并行处理。

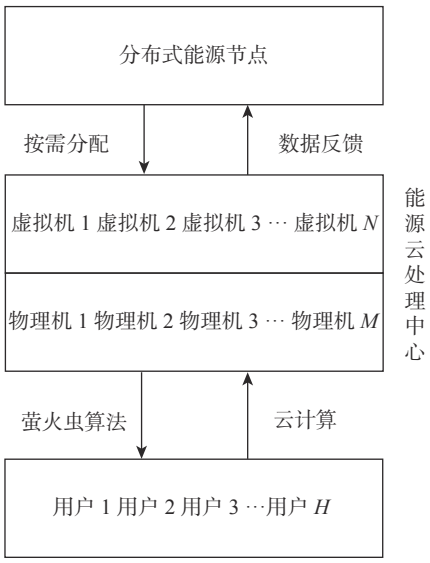

图 1 模型结构
Fig. 1 Model structure

本模型旨在全面搜集能源系统中各个用户的需求信息，把部分子任务分配给负载小的服务器，减轻繁忙时段任务重的服务器的负担，提高能源云处理中心系统的处理效率，进而提高能源利用率。本模型服务器的基本参数主要有可用磁盘空间 D、可用内存 Q、处理器利用率 C 和网络带宽 W 等。D、Q、C、W 的数值越大，则服务器的处理速度越高，负载能力越强，反之则越低越弱。因此，建立的负载值方程为

$$\theta = \frac{\sigma_1 \overline{C_j}}{C_{ij}} + \frac{\sigma_2 \overline{W_j}}{W_{ij}} + \frac{\sigma_3 \overline{Q_j}}{Q_{ij}} + \frac{\sigma_4 \overline{D_j}}{D_{ij}} \tag{1}$$

式中：θ为负载值；C_{ij}为服务器 j 处理子任务 i 时的 CPU 利用率；$\overline{C_j}$为服务器 j 的平均 CPU 利用率；W_{ij}为服务器 j 处理任务 i 时的网络带宽；$\overline{W_j}$为平均带宽；Q_{ij}为服务器 j 处理子任务 i 时的剩余内存；$\overline{Q_j}$为平均剩余内存；D_{ij}为服务器 j 处理子任务 i 时的剩余磁盘空间；$\overline{D_j}$为平均剩余磁盘空间；σ_1、σ_2、σ_3、σ_4为一组权重系数，且$\sigma_1+\sigma_2+\sigma_3+\sigma_4=1$，为了使系统达到更好的优化效果，$\sigma_1$、$\sigma_2$、$\sigma_3$、$\sigma_4$可以适当调节。

当服务器没有处理任务时，其负载值为 0。因此负载值模型为

$$L_{ij} = \exp(-\Delta t)(L_{1j} + \frac{\theta t_{1j}}{\overline{t_j}}) \tag{2}$$

式中：L_{ij}为任务 i 分配到服务器 j 时服务器的负载值；L_{1j}为服务器 j 在上一个任务处理后的负载值；t_{1j}为上一个任务的处理时间；$\overline{t_j}$为服务器 j 处理任务的平均完成时间；$\exp(-\Delta t)$为时间间隔因子，当 $\Delta t \to 0$，$\exp(-\Delta t) \to 0$，因此$L_{ij} \to 0$。

因此，可以得到一个物理机集群服务器的负载值 K，即

$$K = \sum_{j=1}^{N} \sum_{i=1}^{X} L_{ij} \tag{3}$$

所有物理机的总负载值K_s为

$$K_s = \sum_{m=1}^{M} K_m \tag{4}$$

式中，K_m 为第 m 个物理机的负载值。

所有物理机服务器的平均负载值为$\overline{K}$。根据以上条件可算出服务器负载值方差。方差小说明服务器的处理能力相差不大，可将方差作为适应度函数，公式为

$$F = \sum_{m=1}^{M} (K_m - \overline{K})^2 \tag{5}$$

式中，F 为方差。

2.2　改进的萤火虫算法

自然界中萤火虫的求偶方式一般是寻找比自己更亮的萤火虫并向其移动，以此为启发，Krishnanad 和 Ghose 总结形成了基本的人工萤火虫算法（glowworm swarm optimization，GSO）。萤火虫寻找比自己更亮的萤火虫，然后数量越来越多的萤火虫聚集在一起，最终找到局部的最优解。萤火虫的迭代有利于快速寻找全局最优值，同时也有助于局部寻优[16]。GSO 主要包含 4 个阶段：萤火虫的初始化位置、荧光素值更新、萤火虫位置更新、策半径更新。

设可行域中有 n 只萤火虫，每只萤火虫的荧光素值为l_0，荧光素值的更新率为γ，初始化步长为 s，邻域阈值为n_{t}，荧光素消失率为 p，动态决策域更新率为β，萤火虫感知域为r_{s}，萤火虫移动时动态决策域为r_0，萤火虫的迭代次数为 e。荧光素更新公式为

$$l_i(1+t)=(1-p)l_i(t-1)+\gamma J(x_i(t)) \tag{6}$$

式中：$l_i(t)$为萤火虫 i 在 t 时刻荧光素值；$x_i(t)$为 t 代萤火虫的位置，$J(x_i(t))$为相对应的函数适应度值。

每只萤火虫都有对应的决策域范围，它们将利用自己的决策范围寻找其他萤火虫。决策域范围是动态改变的，决策域范围可因该范围内萤火虫数量少而增大，也可因萤火虫数量多而减小。更新动态决策域公式为

$$N_i(t)=\{j:\|x_j(t)-x_i(t)\|<r_d^i;l_i(t)<l_j(t)\} \tag{7}$$

$$r_{\mathrm{d}}^i(t)=\min\left\{r,\max\left\{0,r_{\mathrm{d}}^i(t)+\beta(n_{\mathrm{t}}-|N_i(t)|)\right\}\right\} \tag{8}$$

式中：$r_{\mathrm{d}}^i(t)$为萤火虫 i 在 t 时刻的动态决策域；r 为径向传感器范围；$N_i(t)$为邻域内萤火虫的数目。

萤火虫移动自己的位置后，更新萤火虫 i 的位置为

$$x_i(t+1)=x_i(t)+s_{\mathrm{t}}\frac{\left[x_j(t)-x_i(t)\right]}{\left\|x_j(t)-x_i(t)\right\|} \tag{9}$$

式中：s_{t}为移动步长；‖‖为欧式距离。

传统人工萤火虫算法因存在寻优不稳定、算法精度低、后期收敛速度较慢等缺点。当萤火虫向比它更亮的萤火虫移动时，其决策域范围是固定的。但如果决策范围过大，所需时间就较长，不利于局部寻优；如果决策范围小，则会出现局部最优的情况，最终导致精度不够高。为克服这些缺点，本文对式(8)进行了改进，得到

$$r_{\mathrm{d}}^i(t+1)=\exp\left(\frac{l_j-l_i}{e}\right)\min\left\{0,r_{\mathrm{d}}^i(t)+\beta(n_t-|N_i(t)|)\right\} \tag{10}$$

式中，e 为迭代次数。

在迭代前期，为寻找全局最优，决策域范围可以设定得大一些；在迭代后期，为增强后期的局部搜索能力，决策域范围可慢慢缩小，以防止其他萤火虫的干扰。为了使算法结果最优并克服传统人工萤火虫算法的缺点，在式(8)中加入$\exp\left(\frac{l_j-l_i}{e}\right)$因子。随着迭代次数的增多，决策域范围慢慢减小。由此可知，改进后的算法既提高了算法的精度，又能准确判别各服务器的负载情况，从而了解系统任务分配及处理情况。

3　模型求解

模型求解步骤为：

(1)设定萤火虫的初始参数，确定初始位置；

(2)根据式(6)确定每只萤火虫的荧光素值，同时满足条件：$F=\sum_{m=1}^{M}(K_m-\overline{K})^2$的结果越小，$J(x_i(t+1))$的值越大；

(3)萤火虫在决策域半径内寻找比自己更亮的萤火虫组成邻域集$N_i(t)$，按照$P_{ij}(t)=\left[l_j(t)-l_i(t)\right]/\sum_{k\in N_i(t)}[l_k(t)-l_i(t)]$决定移动方向。

(4)萤火虫开始移动，按照式(9)改变自己的位置。

(5)萤火虫通过式(10)更新自己的决策域半径。

(6)在萤火虫完成一次迭代后就判断是否达到最大迭代次数，若未达到，则继续步骤(2)，若达到最大迭代次数，则输出解。

改进的人工萤火虫算法利用了萤火虫向比自己更亮的萤火虫移动的特性。为处理能力较强的服务器设定了较高的荧光素值，引导萤火虫向处理能力较强的服务器移动，使得各服务器都能处理合理数量的任务，优化了系统配置，从而提高了系统效率。

4 仿真分析

本仿真模拟构架如图 2 所示。分布式能源各节点对接的是云处理中心的各服务器，每个用户的需求量对应的是每个子任务，仿真实验在云处理中心进行。使用 Matlab 软件进行算法运算。参考文献[17]并结合本研究相关配置，获得各参数的取值：$p = 0.5$、$\beta = 0.07$、$\gamma = 0.5$、$s_t = 0.04$，$e = 200$。虚拟机为 Xen，其配置 RAM 为 512 MB、CPU 频率为 4 000 MHz、带宽为 1 000 Mbps。

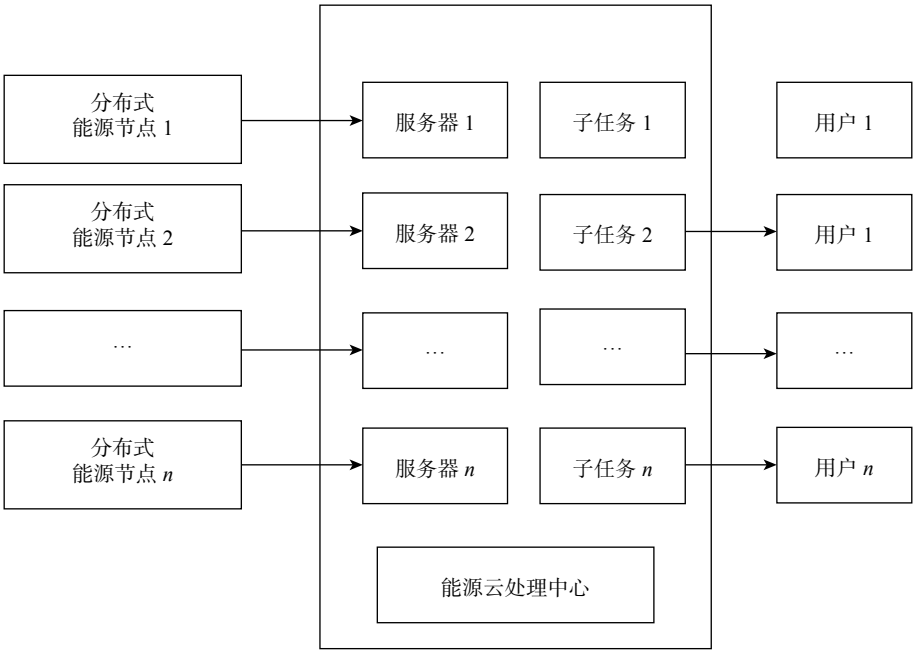

图 2 仿真模拟构架

Fig. 2 Simulation architecture

本研究使用 Linux 系统在 cloudsim 云平台[18]进行仿真实验，设资源分配服务器为 R1，在 R1 上分别运行改进人工萤火虫算法和传统人工萤火虫算法 10 次，得到结果如图 3 所示。两种算法计算得到的负载值虽然都在 0.5 附近波动，但改进人工萤火虫算法得到的负载值波动增幅较小，比较稳定，精度也更高；而传统人工萤火虫算法计算得到的负载值波动幅度较大，不稳定，精度也较低，会影响服务器的运行效率。

将 2 000 个任务分配到 5 个资源分配服务器 R1、R2、R3、R4、R5，其中设置资源分配服务器的处理能力大小依次为 R5、R3、R2、R1、R4。分别运行改进人工萤火虫算法、传统人工萤火虫算法后得到的的任务分配情况如图 4 所示。由分析可知，改进人工萤火虫算法较为全面地考虑了负载指标，因此能够较为均衡地分配任务，能将子任务准确地分配给处理能力强的服务器。在传统萤火虫算法运行过程中出现了任务分

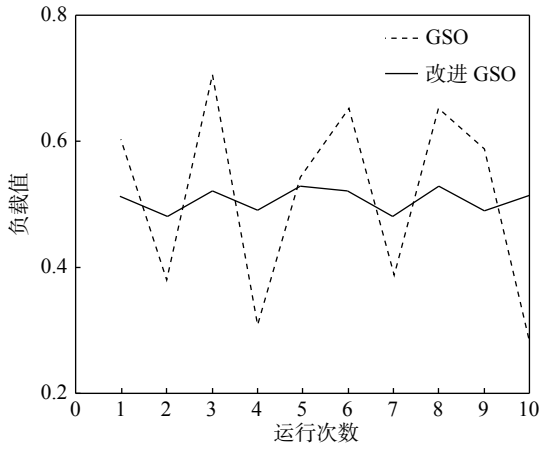

图 3 不同算法运行负载值对比

Fig. 3 Comparison of load values from different algorithms

配不均匀的情况，未给处理能力强的服务器上分配任务，而分配给处理能力低的服务器过多的任务，导致系统运行效率低下。

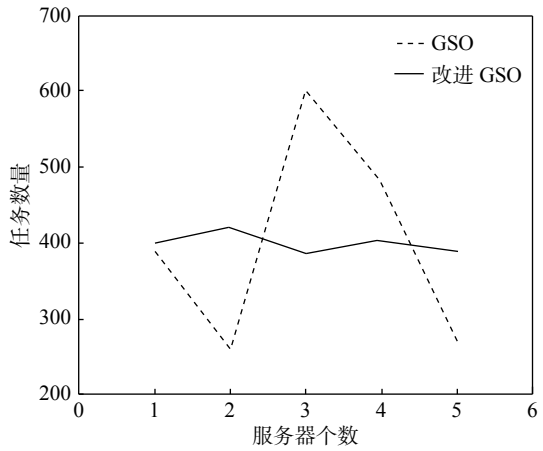

图 4　不同算法服务器任务分配情况

Fig. 4　Task allocation of the servers for different algorithms

将 2 000 个任务分配给 20 个资源分配服务器，分别运行传统人工萤火虫算法和改进人工萤火虫算法。在不同任务数量下，两种算法的完成时间对比如表 1 所示。由此可知，两种算法在处理任务数量较少时完成时间基本一致。但随着任务数量的增加，改进人工萤火虫算法的优势突显，完成时间远短于传统人工萤火虫算法。

表 1　任务完成时间对比

Tab. 1　Comparison of task completion time

任务数量	任务完成时间/(10^2ms)	
	改进GSO	GSO
50	0.420	0.423
100	0.820	0.811
500	4.122	4.466
1 000	7.422	8.658
1 500	11.619	15.814
2 000	17.288	23.921

综合以上实验仿真结果发现，改进人工萤火虫算法在资源分配任务处理中可以有效克服传统人工萤火虫算法的缺点，提高系统寻优的稳定性并缩短任务处理时间，在云计算环境下寻找全局最优的配置方案，使得各服务器都处理相应数量的任务，提高了服务器的利用率，从而提高了能源利用率。因此，本算法能够较好地应用于分布式能源供应链配置的处理。

5　结　论

在云计算环境下，对于大量任务处理及配置问题，为提高系统运行效率、减少任务等待时间，需要一个有良好优化算法的模型。本研究提出的改进人工萤火虫算法在云计算环境下，利用萤火虫会寻找比自身亮的萤火虫并向其移动的特性，移动完毕后更新自己的亮度、位置等信息，以寻找全局最优的配置。与改进前的传统人工萤火虫算法相比，改进后的人工萤火虫算法具有寻优速度快、精度高、稳定强等优点，能够快速得出最优的配置方案，求出系统模型的最优解。该算法应用于分布式能源供应链问题时，提高了能源利用率，使其达到一个高效率的状态，解决了分布式能源的配置问题。分布式能源供应链配置需求侧与供给侧的均衡分配，让整个系统的需求与供给相匹配，减少了资源的浪费和资源过剩的情况，使系统达到节能环保的状态。在萤火虫向其他萤火虫移动的过程中是否有其他因素影响，在萤火虫数量较多时是否会影响萤火虫的移动方向，这些外在影响因素都有待进一步研究，进而优化人工萤火虫算法。本文仅考虑了需求侧的平衡及智能优化，对于供给侧的优化分配，以及后期供给侧与需求侧之间的动态平衡的双层优化配置模型还有待进一步研究。

参考文献：

[1] WU X F, XU Y C, LOU Y T, et al. Low carbon transition in a distributed energy system regulated by localized energy markets[J]. Energy Policy, 2018, 122: 474 – 485.

[2] 董明. 分布式能源供应链网络建模解析集成框架 [J]. 上海交通大学学报, 2018, 42(11): 1772 – 1775.

[3] 邢海军, 程浩忠, 张沈习, 等. 主动配电网规划研究综述 [J]. 电网技术, 2015, 39(10): 2705 – 2711.

[4] 李燕, 胡志坚, 仉梦林, 等. 基于不确定网络理论计及

经济性与可靠性的主动配电网规划 [J]. 电力系统自动化, 2019, 43(16): 68 - 77.
[5] 随权, 林湘宁, 童宁, 等. 基于改进两阶段鲁棒优化的主动配电网经济调度 [J]. 中国电机工程学报, 2020, 40(7): 2166 - 2180.
[6] KANDIL S M, FARAG H E Z, SHAABAN M F, et al. A combined resource allocation framework for PEVs charging stations, renewable energy resources and distributed energy storage systems[J]. Energy, 2017, 143: 961 - 972.
[7] 田冬冬, 潘鹏. 基于云计算智能电网的能源监控管理分析及发展研究 [J]. 山东工业技术, 2018(24): 208.
[8] 马俊. 云计算环境的能源效率分析研究 [J]. 信息技术, 2017(2): 111 - 113, 117.
[9] 林君豪, 张焰, 陈思, 等. 考虑可控负荷影响的主动配电系统分布式电源优化配置 [J]. 电力自动化设备, 2016, 36(9): 46 - 53, 73.
[10] 马瑞, 金艳, 刘鸣春. 基于机会约束规划的主动配电网分布式风光双层优化配置 [J]. 电工技术学报, 2016, 31(3): 145 - 154.
[11] 李军祥, 戴韬, 叶春明. 云计算下的分布式联络中心服务设计 [J]. 上海理工大学学报, 2014, 36(4): 338 - 344.
[12] 李瑞婷, 叶春明, 吴思思. 运用改进的萤火虫算法求解 TFT-LCD 单元装配调度问题 [J]. 上海理工大学学报, 2018, 40(6): 525 - 533.
[13] 单好民. 云计算下的基于萤火虫-遗传算法的资源调度 [J]. 计算机系统应用, 2016, 25(5): 187 - 191.
[14] 杨朝辉, 康磊. Hadoop 平台中的 MapReduce 模型及优化 [J]. 信息技术与信息化, 2017(12): 82 - 85.
[15] 曹建, 李峥, 杨璞, 等. 云计算环境下基于 MapReduce 的并行化排列熵算法 [J]. 电力信息与通信技术, 2019, 17(1): 1 - 6.
[16] TANG S H, LIU X M. An improved adaptive step glowworm swarm optimization algorithm[J]. CAAI Transactions on Intelligent Systems, 2015, 10(3): 470 - 475.
[17] 杨爱华. 云计算下的蚁群优化算法资源调度研究 [J]. 通化师范学院学报: 自然科学, 2019, 40(4): 1 - 5.
[18] JAMMAL M, HAWILO H, KANSO A, et al. Generic input template for cloud simulators: a case study of CloudSim[J]. Software: Practice and Experience, 2019, 49(5): 720 - 747.

信息

“深海一号”累计产气超 10 亿 m^3

2 月 13 日，从中国海洋石油集团有限公司获悉，自 2021 年 6 月 25 日正式投产以来，我国首个自营 1 500 m 超深水大气田“深海一号”累计生产天然气超 10 亿 m^3。

“深海一号”大气田是我国迄今为止自主发现的平均水深最深、勘探开发难度最大的海上超深水气田，气田量身定制了全球首座十万 t 级深水半潜式生产储油平台——“深海一号”能源站，投产后每年可向粤港琼等地稳定供气 30 亿 m^3，能满足粤港澳大湾区 1/4 的民生用气需求。

“深海一号”能源站生产工艺处理系统的建设规模和流程的复杂程度在世界同类型海上油气处理设施中居于前列，生产区管线长度超过 6 万 m，温度跨度从−25℃ 到 200℃。

宝钢湛江钢铁开建国内首套百万 t 级氢基竖炉

2 月 15 日，宝钢湛江钢铁零碳示范工厂百万 t 级氢基竖炉工程正式开工。该项目是国内首套百万 t 级氢基竖炉，也是首套集成氢气和焦炉煤气进行工业化生产的直接还原生产线。项目预计 2023 年底建成，投产后每年可减少二氧化碳排放 50 万 t 以上，也是后续国内自主集成并研发全氢冶炼技术的创新平台，对推动钢铁行业绿色低碳转型、推进钢铁新生态绿色化、促进区域清洁能源优化和经济发展具有重要意义。

（王 波）