

$$\tau_{t_i,vm_j}(n+1) = \begin{cases} \rho^{*e^{-0.5}} \times \tau_{t_i,vm_j}(n) + \Delta\tau_{t_i,vm_j}, & \text{if } \tau_{t_i,vm_j}(n) > \sigma_{\max} \\ \rho^{*e^{0.5}} \times \tau_{t_i,vm_j}(n) + \Delta\tau_{t_i,vm_j}, & \text{if } \tau_{t_i,vm_j}(n) < \sigma_{\min} \end{cases} \quad (15)$$

如果在本次迭代中发现最优解,则发现最优解的蚂蚁在和其他蚂蚁一样更新完自己路径上的信息素后,还要再进行一次更新,所有蚂蚁更新公式如下:

$$\left. \begin{aligned} \tau_{t_i,vm_j}(n+1) &= \rho^* \times \tau_{t_i,vm_j}(n) + \Delta\tau_{t_i,vm_j} \\ \Delta\tau_{t_i,vm_j} &= \begin{cases} \sum_{k=1}^m \Delta\tau_{t_i,vm_j}^k + \chi \times \Delta\tau_{t_i,vm_j}^{bs}, & \text{if the Ant find the } bs \\ \sum_{k=1}^m \Delta\tau_{t_i,vm_j}^k, & \text{others} \end{cases} \\ \Delta\tau_{t_i,vm_j}^{bs} &= \begin{cases} 1/bscost, & \text{if } bs \text{ contains}(t_i,vm_j) \\ 0, & \text{others} \end{cases} \\ \Delta\tau_{t_i,vm_j}^k &= \frac{q}{\sum L_k} \end{aligned} \right\} \quad (16)$$

其中, m 为蚂蚁个数; n 为迭代次数; $\Delta\tau_{t_i,vm_j}^k$ 为蚂蚁 k 由任务节点 t_i 到虚拟机 vm_j 的路径上留下的信息素数量; ρ 为路径上信息素数量的蒸发系数; q 为信息素质量系数; χ 是一个随机数,在 $[0,1]$ 中进行选取; $\sum_{k=1}^m \Delta\tau_{t_i,vm_j}^k$ 是所有蚂蚁共同更新的信息素; bs 为当前最优解; $bscost$ 为当前最优解的费用值; $\sum L_k$ 为蚂蚁 k 的调度方案费用值。

2. 启发性信息的优化

为了增大效率高的虚拟机被选到的概率,修改启发性信息 η_{t_i,vm_j} 如下:

$$\left. \begin{aligned} \gamma &= \frac{(m_{vm_j} / P_{vm_j}) \times m}{\sum_{n=1}^m m_{vm_j} / P_{vm_j}} \\ \eta_{t_i,vm_j} &= \gamma / (et(t_i,vm_j) + st(t_i,vm_j)) \end{aligned} \right\} \quad (17)$$

其中, γ 代表 vm_j 在所有虚拟机中的效率值。

多科学动态调度策略为:若用户在提交的科学 workflow 未执行完成的情况下提交了新的科学 workflow,需给出新的调度参数,首先对新的科学 workflow 进行压缩,其次计算新的科学 workflow 每个任务节点的权重值;未执行完成的科学 workflow 剩下的节点层数从第 1 层开始计数,然后按层数从低到高将新的科学 workflow 每一层任务节点依据权重值递增排序的规则合并到还未执行完成的科学 workflow 所在的每一层任务队列中,生成新的调度序列,最后,利用优化的 ACO 算法进行调度.这样可以避免新的科学 workflow 因之前科学 workflow 剩余任务还未执行而得不到调度最终导致执行时间跨度变大的问题。

MSW-SDCOA 具体算法如算法 2 所示。

算法 2. MSW-SDCOA 算法.

输入: $W=\{w_1,w_2,w_3,\dots,w_N\}$; //科学 workflow 列表

$VM=\{vm_1,vm_2,vm_3,\dots,vm_m\}$. //云环境下虚拟机列表

输出: $bestschedule$. //安全性和时间约束下的最优调度方案

1. Workflow compress
2. 通过任务调度序列生成算法生成队列 Q
3. 初始化信息素矩阵 τ ,初始化蚁群 Ants
4. **for** n to $maxgen$ **do** //蚁群开始进行迭代, $maxgen$ 是最大迭代数
5. **for** Ant in Ants **do**
6. Choose the VM for each task t in queue Q ;


```

7. Calculate the Ant's  $S(W)$  according to Eq.(4); //计算安全性
8. Calculate the Ant's makespan according to Eq.(8); //计算完成时间
9. Calculate the Ant's allc according to Eq.(9); //计算总费用值
10. if  $S(W) \geq scon \ \&\& \ makespan \leq deadline \ \&\& \ allc < bestschedule.allc$ 
11.   bestschedule=ant.schedule;
12. else
13.   continue;
14. end if
15. Update the  $\tau$ ;
16. end for
17. end for
18. if bestschedule==null
19.   return "please modify the scheduling parameters";
20. else
21.   return bestschedule;

```

算法的具体执行过程如下.

- (1) 对多科学 workflow 进行压缩(第 1 行).
- (2) 对压缩后的多科学 workflow 调用任务调度序列生成算法生成 Q (第 2 行).
- (3) 为 ACO 算法的运行做初始化工作(第 3 行).
- (4) 开始进行迭代,计算每一次迭代中每只蚂蚁调度方案的安全性、运行时间和费用,在安全性和时间约束下判断是否为最优调度方案:如果是,则保留最优方案,每一次迭代过后需要更新信息素矩阵(第 4 行~第 17 行).
- (5) 迭代完成后,如果最优调度方案为空,则说明未找到符合安全和时间约束下的调度方案,提醒用户修改调度参数;否则,返回安全和时间约束下的最优调度方案(第 18 行~第 21 行).

3 仿真实验

为了分析和评估本文提出的 MSW-SDCOA 算法的性能,将 MSW-SDCOA 算法与 HEFT-ACO(HEFT 算法加上传统 ACO 算法)、MW-DBS 算法^[27]和 CCRH 算法^[6]进行对比,本文在 CloudSim^[28]平台上进行仿真调度实验.为了能进行公平地对比,HEFT-ACO 和 MW-DBS 算法的安全性计算均采用本文所述的方法进行计算.实验基于以下环境:处理器为 Inter(R) Core(TM)i5-4590,3.30GHz;8GB 内存;Windows 7,64 位操作系统.

3.1 实验环境

本文不考虑数据中心问题,实验采用一个数据中心、10 个不同类型的虚拟机,假设每个虚拟机都拥有 3 种安全性服务,各个虚拟机参数信息见表 2.

Table 2 Virtual machine parameter information

表 2 虚拟机参数信息

编号	MIPS(Hz)	带宽(Mbps)	单价(\$/h)	编号	MIPS(Hz)	带宽(Mbps)	单价(\$/h)
1	531	1 200	1.59	6	890	1 334	2.67
2	1 550	785	4.65	7	1 223	1 350	3.67
3	985	980	2.95	8	1 079	1 446	3.24
4	605	558	1.81	9	1 237	980	3.71
5	1 345	899	4.03	10	786	1 043	2.36

实验所用的科学 workflow 为 Montage, LIGO 和 Epigenomics,通过采用 WorkflowGenerator 生成器^[29],生成任务数为 25,50,100 的 Montage 工作流、任务数为 30,50,100 的 LIGO 工作流和任务数为 24,46,100 的 Epigenomics

工作流.实验中,多科学工作流组合设定是用户同时提交的多个科学工作流.

对于 3 种安全性服务,实验使用 HMAC-MD5,HMAC-SHA-1 和 CBC-MAC-AES 这 3 种认证方式实现身份服务,使用 7 个哈希函数(MD4,MD5,RIPEMD,RIPEMD-128,SHA-1,RIPEMD-160 和 Tiger)实现完整性服务,使用 8 种加密算法(SEAL,RC4,Blowfish,Knufu/Khafre,RC5,Rijndael,DES 和 IDEA)实现机密性服务^[25].3 种安全服务的风险系数参数设定为 $\lambda^a=3.5,\lambda^b=2.5,\lambda^c=1.5$,参数设定参考文献[30,31].

对于优化的 ACO 算法,实验中定义参数 $\rho=0.65,\alpha=1.1,\beta=4.0,q=700,\varphi=3$,蚁群数量为 60,迭代次数为 200,参数设定参考文献[30,31].

3.2 实验结果对比

图 3 显示了在截止时间相同的情况下,对比在不同安全性约束下,将 MSW-SDCOA 算法的调度结果与 HEFT-ACO 算法、MW-DBS 算法、CCRH 算法的调度结果进行对比.实验中分别采用了(LIGO-30,Montage-25,Epigenomics-24),(LIGO-50,Montage-50,Epigenomics-46)以及(LIGO-100,Montage-100,Epigenomics-100)这 3 种多科学工作流实例,MSW-SDCOA 算法、HEFT-ACO 算法、MW-DBS 算法和 CCRH 算法在多种不同的多科学工作流实例进行比较,保证实验的准确性.实验采取的截止时间为 $(t_{\min}+t_{\max})/2$, t_{\min} 和 t_{\max} 分别代表多科学工作流在虚拟机配置条件不变且没有安全性约束下的最小完成时间和最大完成时间,不同组合的多科学工作流截止时间不同.从图 3 可以看出,对于不同组合的多科学工作流,随着安全性约束的提高,4 种调度算法的费用也在增大.这是因为随着安全级别的提升,任务节点在虚拟机上运行的时间会逐渐增加.

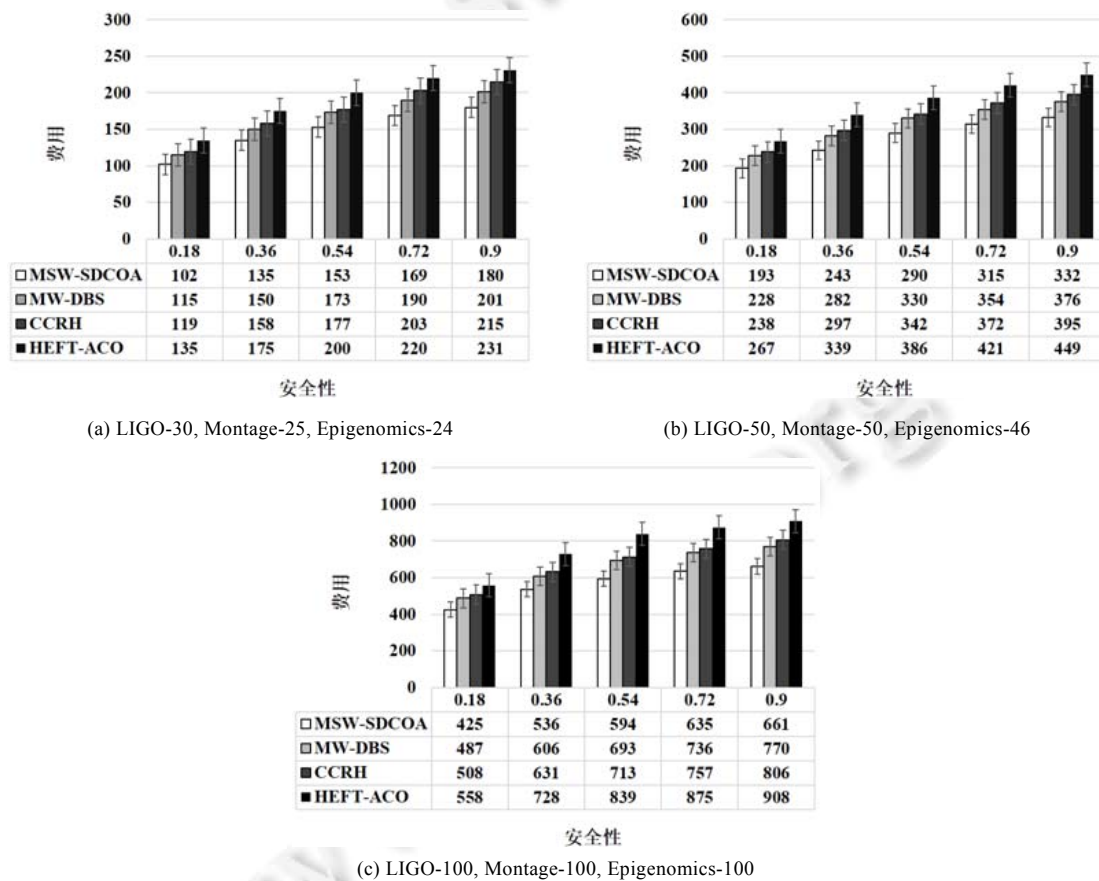


Fig.3 Scheduling results of different multi-scientific workflows under different security constraints

图 3 不同安全性约束下,不同多科学工作流组合调度结果

图 4 显示了在安全性相同的情况下,对比在不同截止时间下,将 MSW-SDCOA 算法的调度结果与 HEFT-ACO 算法、MW-DBS 算法、CCRH 算法的调度结果进行对比.实验中分别采用了(LIGO-30, Montage-25, Epigenomics-24),(LIGO-50, Montage-50, Epigenomics-46)以及(LIGO-100, Montage-100, Epigenomics-100)这 3 种多科学 workflow 实例.实验中,安全性值设为 0.7,选取最小完成时间和最大完成时间区间中的 5 个数值作为用户截止时间,即 $deadline = t_{min} + \theta \times (t_{max} - t_{min})$,其中, $\theta \in \{0.15, 0.3, 0.45, 0.6, 0.75\}$.从图 4 可以看出,随着用户截止时间的提高,4 种调度算法的费用在降低.这是因为随着截止时间的提高,符合安全性和时间约束的调度方案逐渐增多.

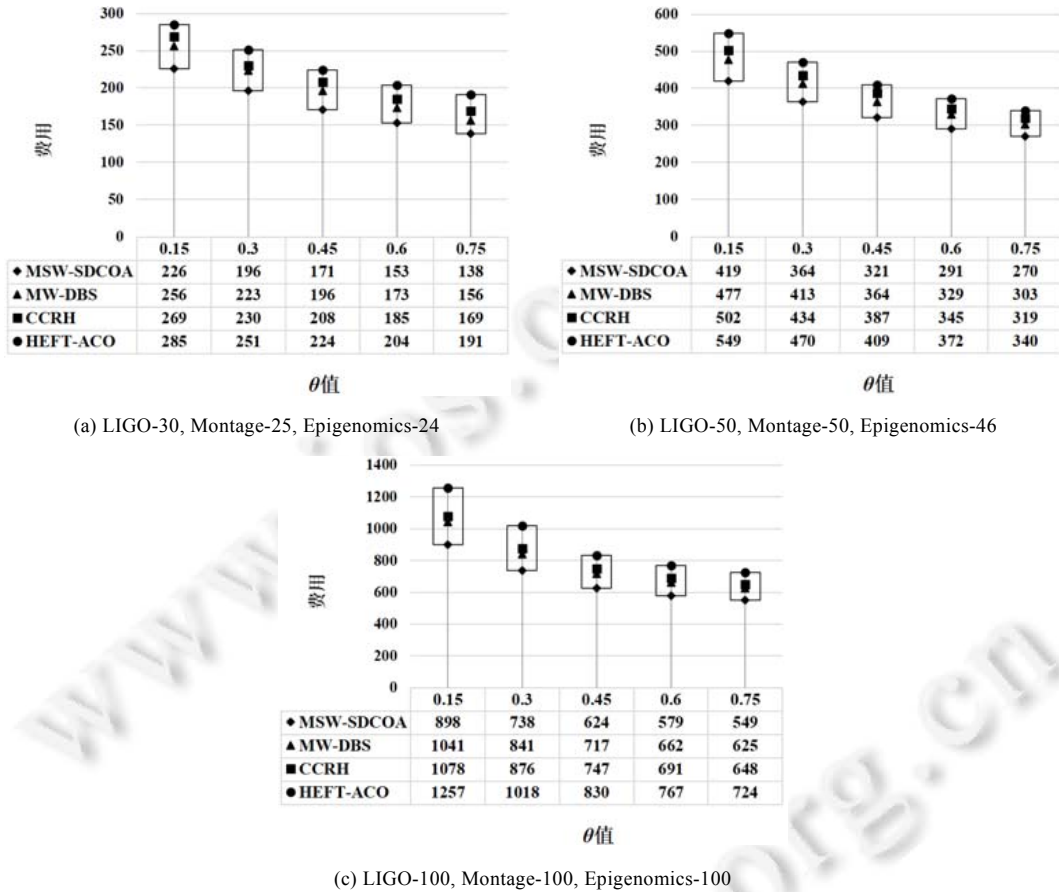


Fig.4 Scheduling results of different multi-scientific workflows under different deadline constraints

图 4 不同时间约束下,不同多科学 workflow 组合调度结果

在费用优化方面,从图 3 和图 4 可以看出,MSW-SDCOA 算法所生成的调度方案费用值最低.这是由于 MSW-SDCOA 算法中的任务调度序列生成算法考虑了全局多目标优化,从而生成更加合理的调度序列,并且在信息素的优化中,对于信息素数量的蒸发系数的改进、增加信息素的上下阈值以及发现最优调度方案时的处理策略,避免了 ACO 算法出现过早收敛的情况,以至于增强了 ACO 算法的全局搜索能力,最终能够得到更好的调度结果.

图 5 显示了在安全性和截止时间约束相同的情况下,对比在同时到达的不同多科学 workflow 个数下的完成时间大小.从图 5 中可以看出,随着多科学 workflow 个数的增加,各算法运行的时间也增加,但是本文方法的耗时间低于 HEFT-ACO 算法、MW-DBA 算法和 CCRH 算法.这是由于在初始阶段进行了 workflow 压缩,使得任务数减少,并且避免了部分任务之间的数据传输,在寻找最优调度方案中,对启发式信息的优化,使效率高的虚拟机被

选到的概率增大,从而减少多科学 workflows 调度时间.

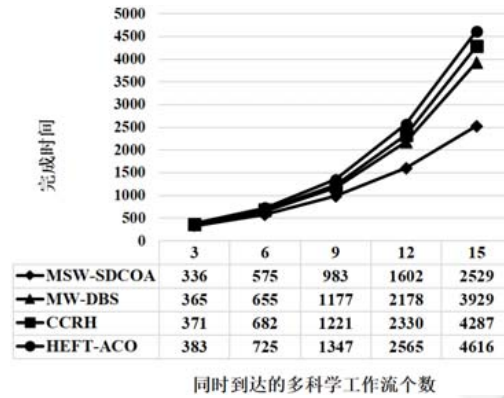


Fig.5 Scheduling time of the number of different scientific workflows

图 5 不同多科学 workflows 个数下调度完成时间

图 6 显示了 4 种算法对于图 3 和图 4 中 6 个实验的云资源平均利用率,从图中可以看出,本文算法对于云资源的利用率高于 HEFT-ACO 算法、MW-DBA 算法和 CCRH 算法,这体现出在相同的云资源条件下,MSW-SDCOA 算法能够有更好的收益.

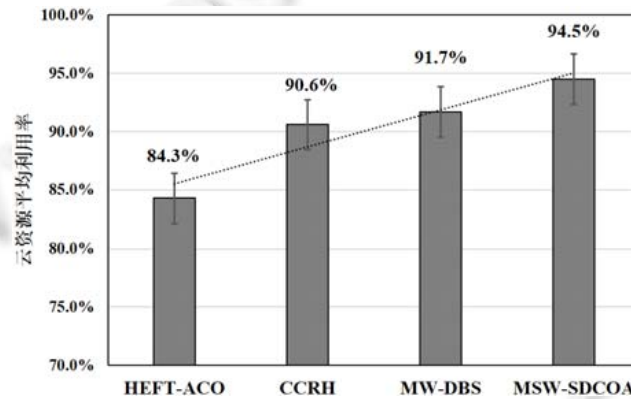


Fig.6 Average utilization of cloud resources

图 6 云资源平均利用率

4 结束语

针对现有多科学 workflows 费用优化模型未考虑安全调度问题,本文设计了带有安全性和时间约束多科学 workflows 费用优化模型,并提出了针对该模型的费用优化算法 MSW-SDCOA. MSW-SDCOA 利用数据依赖关系压缩科学 workflows,减少了费用开销,并通过改进 HEFT 算法形成调度序列以及优化 ACO 中信息素更新策略和启发式信息,进一步改善费用优化效果.实验中使用了 3 种不同领域下的真实科学 workflows,与 HEFT-ACO 和 MW-DBA 算法进行比较,证明了 MSW-SDCOA 算法的有效性.在未来的研究工作中,将考虑多个用户同时提交多个科学 workflows 的情况以及资源负载均衡对科学 workflows 调度的影响.

References:

- [1] Deelman E, Singh G, Livny M, Berriman B. The cost of doing science on the cloud: The montage example. In: Proc. of the 2008 ACM/IEEE Conf. on Supercomputing. Piscataway: IEEE, 2008. 1-12.

- [2] Kannas CC, Kalvari I, Lambrinidis G, Neophytou CM, Savva CG, Kirmizoglou I, Antoniou Z, Achilleos KG, Scherf D, Pitta CA. LiSIs: An online scientific workflow system for virtual screening. *Combinatorial Chemistry & High Throughput Screening*, 2015, 18(3):281–295.
- [3] Bennett JC, Bhagatwala A, Chen JH, Seshadhri C, Pinar A, Salloum M. Trigger detection for adaptive scientific workflows using percentile sampling. *SIAM Journal on Scientific Computing*, 2016,38(5):S240–S263.
- [4] Pradal C, Artzet S, Chopard J, Dupuis D, Fournier C, Mielewicz M, Nègre V, Neveu P, Parigot D, Valduriez P. InfraPhenoGrid: A scientific workflow infrastructure for plant phenomics on the grid. *Future Generation Computer Systems*, 2017,67:341–353.
- [5] Zhao Y, Li Y, Raicu I, Lu S, Lin C, Zhang Y, Tian W, Xue R. A service framework for scientific workflow management in the cloud. *IEEE Trans. on Services Computing*, 2015,8(6):930–944.
- [6] Wang Y, Jia C, Xu Y. Multiple dags dynamic workflow scheduling based on the primary backup algorithm in cloud computing system. In: *Proc. of the Ninth Int'l Conf. on Broadband and Wireless Computing, Communication and Applications*. Piscataway: IEEE, 2014. 177–182.
- [7] Tischer A, Auton M. Scheduling strategies for mapping application workflows onto the grid. In: *Proc. of the 14th IEEE Int'l Symp. on High Performance Distributed Computing*. Piscataway: IEEE, 2005. 125–134.
- [8] Zhao H, Sakellariou R. Scheduling multiple DAGs onto heterogeneous systems. In: *Proc. of the 2006 Parallel and Distributed Processing Symp.* Washington: IEEE, 2006. 1–14.
- [9] Rodriguez MA, Buyy R. Deadline based resource provisioning and scheduling algorithm for scientific workflows on clouds. *IEEE Trans. on Cloud Computing*, 2014,2(2):222–235.
- [10] Weins K. Cloud computing trends: 2017 state of the cloud survey. 2017. <https://www.rightscale.com/blog/cloud-industry-insights/cloud-computing-trends-2017-state-cloud-survey>
- [11] Yurcik W, Koenig GA, Meng X, Greenesid J. Cluster security as a unique problem with emergent properties: Issues and techniques. In: *Proc. of the 5th Lei Int'l Conf. on Linux Clusters: The Hpc Revolution*. 2004. 18–20.
- [12] Arabnejad H, Barbosa JG, Prodan R. Low-Time complexity budget-deadline constrained workflow scheduling on heterogeneous resources. *Future Generation Computer Systems*, 2016,55:29–40.
- [13] Lin B, Guo W, Chen G, Xiong N, Li R. Cost-Driven scheduling for deadline-constrained workflow on multi-clouds. In: *Proc. of the 2015 IEEE Int'l Parallel and Distributed Processing Symp. Workshop*. Piscataway: IEEE, 2015. 1191–1198.
- [14] Meena J, Kumar M, Vardhan M. Cost effective genetic algorithm for workflow scheduling in cloud under deadline constraint. *IEEE Access*, 2016,4:5065–5082.
- [15] Singh L, Singh S. Deadline and cost based ant colony optimization algorithm for scheduling workflow applications in hybrid cloud. *Journal of Scientific & Engineering Research*, 2014,5(10):1417–1420.
- [16] Verma A, Kaushal S. A hybrid multi-objective particle swarm optimization for scientific workflow scheduling. *Parallel Computing*, 2017,62:1–19.
- [17] Zissis D, Lekkas D. Addressing cloud computing security issues. *Future Generation Computer Systems*, 2012,28(3):583–592.
- [18] Huang Q, Yang Y, Shen M. Secure and efficient data collaboration with hierarchical attribute-based encryption in cloud computing. *Future Generation Computer Systems*, 2017,72:239–249.
- [19] Sookhak M, Yu FR, Khan MK, Xiang Y, Buyya R. Attribute-Based data access control in mobile cloud computing: Taxonomy and open issues. *Future Generation Computer Systems*, 2017,72:273–287.
- [20] Ali M, Khan SU, Vasilakos AV. Security in cloud computing: Opportunities and challenges. *Information Sciences*, 2015,305:357–383.
- [21] Sharif S, Taheri J, Zomaya AY, Nepal S. Mphc: Preserving privacy for workflow execution in hybrid clouds. In: *Proc. of the 2013 Int'l Conf. on Parallel and Distributed Computing, Applications and Technologies*. Piscataway: IEEE, 2013. 272–280.
- [22] Zeng L, Veeravalli B, Li X. SABA: A security-aware and budget-aware workflow scheduling strategy in clouds. *Journal of Parallel and Distributed Computing*, 2015,75:141–151.
- [23] Bala A, Chana I. Intelligent failure prediction models for scientific workflows. *Expert Systems with Applications*, 2015,42(3):980–989.

- [24] Topcuoglu H, Hariri S, Wu M. Performance-Effective and low-complexity task scheduling for heterogeneous computing. *IEEE Trans. on Parallel and Distributed Systems*, 2002,13(3):260–274.
- [25] Xie T, Qin X. Scheduling security-critical real-time applications on clusters. *IEEE Trans. on Computers*, 2006,55(7):864–879.
- [26] Tang XY, Li K, Zeng Z, Veeravalli B. A novel security-driven scheduling algorithm for precedence-constrained tasks in heterogeneous distributed systems. *IEEE Trans. on Computers*, 2011,60(7):1017–1029.
- [27] Arabnejad H, Barbosa JG. Multi-Workflow QoS-constrained scheduling for utility computing. In: *Proc. of the 2015 IEEE 18th Int'l Conf. on Computational Science and Engineering*. Piscataway: IEEE, 2015. 137–144.
- [28] Calheiros RN, Ranjan R, Beloglazov A, Rose CAFD, Buyya R. CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software: Practice and Experience*, 2011,41(1):23–50.
- [29] Bharathi S, Chervenak A, Deelman E, Mehta G, Su MH, Vahi K. Characterization of scientific workflows. In: *Proc. of the 3rd Workshop on Workflows in Support of Large-Scale Science*. Piscataway: IEEE, 2008. 1–10.
- [30] Vinay VP, Sridharn R. Taguchi method for parameter design in ACO algorithm for distribution-allocation in a two-stage supply chain. *Int'l Journal of Advanced Manufacturing Technology*, 2013,64(9-12):1333–1343.
- [31] Siemiński A. Ant colony optimization parameter evaluation. *Multimedia and Internet Systems: Theory and Practice*, 2013,183: 143–153.



袁友伟(1966—),男,湖北潜江人,博士,教授,CCF 专业会员,主要研究领域为云计算, workflow 调度.



俞东进(1969—),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为软件工程理论和方法,业务过程管理,行业大数据.



鲍泽前(1994—),男,硕士生,主要研究领域为 workflow 调度,大数据分析.



李万清(1979—),男,博士,副教授,主要研究领域为大数据分析.