

# 联盟模式下云资源提供最优化算法

许利军<sup>1,2</sup>, 高国红<sup>3</sup>, 张小庆<sup>4</sup>

(1. 武汉理工大学 计算机学院, 武汉 430063; 2. 新乡学院 计算机与信息工程学院, 河南 新乡 453003;  
3. 河南科技学院 信息工程学院, 河南 新乡 453003; 4. 武汉轻工大学 数学与计算机学院, 武汉 430023)

**摘要:** 为了提高资源利用率和云资源提供者的收益, 提出一种云联盟模式下基于合作博弈的资源提供算法. 算法将资源联盟收益最优化形式化为线性规划问题, 通过合作博弈核求解了该收益最优化问题, 并证明了处于合作核中的资源提供策略同样是线性规划最优解. 进一步, 为了获得稳定的资源联盟结构, 设计了基于离散 Markov 链的动态联盟构造模型, 通过该模型可以得到收益最大化的最优联盟结构. 仿真实验结果表明, 对于具有博弈理性的资源提供者而言, 资源联盟可以有效提高个体资源提供者的收益, 同时, 由于结盟代价的存在, 稳定的联盟结构更加倾向于形成规模更小的联盟.

**关键词:** 云计算; 资源提供; 合作博弈; 联盟结构

**中图分类号:** TP393.1

**文献标志码:** A

云计算结合了网格计算、集群计算和效用计算的特点, 在以服务形式提供资源的同时, 更加具有可用性、可扩展性和灵活性<sup>[1]</sup>. 对于用户而言, 由于云计算资源访问的灵活性和需求易伸缩的特点, 可以极大地降低总体拥有成本. 对于云资源提供者 CRPs 而言, 云资源不仅可供内部用户使用, 而且还可以提供给外部公有云用户. 这样一来, 资源提供者不仅使其资源得到更有效地利用, 还可以从可用资源中获取更多的收益.

云供应商可以以服务的形式在数据中心提供各种类型的资源(如 CPU、存储和网络资源), 供应商之间也可以相互提供服务, 如: 三大云供应商之间, Amazon, Microsoft 和 Google 之间均可以得到对方提供的云服务, 服务形式为 IaaS, PaaS 和 SaaS, Google Doc(SaaS) 可以购买 Microsoft 的 Azure OS, 而两者均可以使用 Amazon Web Service 的基础设施. 本文将供应商之间的资源提供称为云内部需求, 而云计算的目标是向广大用户提供服务并按时收费, 一般内部需求量不足以使云供应商的资源得到充分利用, 因此, 云供应商还可以通过合作的形式形成一个更大的资源池向外部公有云用户提供资源<sup>[2]</sup>.

博弈理论是解决资源管理的经典经济学方法<sup>[3]</sup>. 相关研究中, 文献[4]提出一种确保 QoS 约束的云框架并建立为博弈模型, 利用 Nash 均衡得到了任务与可用资源之间的最优分配结果. 文献[5]提出一种云资源分配的协作博弈方案. 通过建立服务提供商之间的合作关系, 可以实现总体收益的最大化. 文献[6]提出资源分配负载均衡的博弈算法, 算法将云系统中的负载均衡问题构建为非合作博弈模型, 并证明了模型存在 Nash 均衡解. 文献[7]提出基于动态博弈的用户行为模型, 通过不完全信息多阶段博弈分析终端用户的类型, 博弈时将用户的当前行动和历史行动相结合, 并考虑了网络中存在的误报和漏报的情况, 以加强对云终端用户类型推断的准确性和全面性. 文献[8]提出一种基于效益博弈的云计算资源动态可协调分配算法, 算法采用时间矩阵和费用矩阵作为任务效益的衡量指标, 并构建为效益博弈模型, 通过动态协调机制求解博弈解. 综合现有研究成果可以看出, 现在成果多以非合作博弈对资源竞争分配进行建模<sup>[4,6-7]</sup>, 旨在以相互最优的形式得到个体的最大化收益, 但非合作博弈忽略了资源提供者间的资源联盟形式; 而合作博弈进行资源

收稿日期: 2016-09-29; 修回日期: 2016-12-17.

基金项目: 河南省科技攻关计划项目(152102210202); 河南省高等学校重点科研项目(15A520094); 新乡学院科技创新基金项目(15ZA04).

作者简介(通信作者): 许利军(1978-), 男, 河南清丰人, 新乡学院副教授, 武汉理工大学博士研究生, 研究方向为云计算和计算机网络, E-mail: xljwork@126.com.

分配求解的文献[5,8]旨在实现 Pareto 最优,但文献中只提出了理论模型,并没有实证分析.基于以上工作,本文提出一种联盟模式下的合作式博弈云资源提供算法,以合作核求解最大化收益,并在实验平台中对算法性能进行分析.

## 1 模型定义

为了便于对本文的模型进行形式化描述,本节给出以下模型所涉及的相关符号及对应说明:

$N$  表示云资源供应商集合,表示为  $N = \{1, 2, \dots, N\}$ ,  $R_i$  表示云供应商  $i$  的资源集合,  $R_i = \{\text{CPU}, \text{Mem}, \text{Str}, \text{Net}\}$ ,  $i \in N$ , 分别表示 CPU、内存、磁盘存储和网络带宽资源,  $R_{i,r}$  表示资源提供者  $i$  提供的  $r$  类型资源的资源量,  $r \in R_i$ ,  $T$  表示用户应用集合,  $t \in T$ , 假设每个虚拟机 VM 执行一个应用实例  $t$ ,  $V_{i,r}$  表示应用  $t$  请求的  $r$  类型资源的资源量,以 VM 数量表示,  $x_t$  表示预约阶段(承诺提供)为应用  $t$  提供的资源量,  $y_t$  表示即付即用阶段为应用  $t$  提供的资源量(预约使用和即付即用是目前用户使用云资源的主要两种方式,本文考虑的场景为用户先按预约使用资源,若预约资源量不能满足用户需求,则进一步利用即付即用使用资源),  $d_{i,t}$  表示提供者  $i$  对于应用  $t$  的内部需求,令内部需求是随机的,  $d_{i,t} \in D_{i,t}$ ,  $D_{i,t} \in \{D_{i,t,1}, D_{i,t,2}, \dots, D_{i,t,k}, \dots, D_{i,t,|D_{i,t}|}\}$ ,  $d_{i,t}$  取值为  $D_{i,t,k}$  的概率为  $p(D_{i,t,k})$ ,  $\theta_{i,r}$ ,  $\theta_{i,o}$  表示预约阶段和即付即用阶段中对应用  $t$  提供的单位 VM 的价格,一般地,  $\theta_{i,r} \leq \theta_{i,o}$ , 这样可以有效降低用户使用资源的代价  $\theta_{i,d}$  表示供小于求时,未满足需求的 VM 价格.

## 2 资源提供最优化目标

### 2.1 最优化模型

云计算环境中,云资源提供者在满足 CRPs(Cloud Resource Providers)内部需求和向外部公有云用户以两阶段提供资源的同时,是以最大化收益为目标的,基于以上的相关定义,资源提供者  $i$  的最优化目标可形式化为:

$$\begin{aligned} \max_{x_t, y_t, z_t} &= \sum_{t \in T} (\theta_{i,r} x_t) + \sum_{t \in T} (\theta_{i,o} y_t - \theta_{i,d} z_t), \\ \text{s. t.} & \sum_{t \in T} (x_t + y_t - z_t) V_{i,r} \leq R_{i,r} - \sum_{t \in T} d_{i,t} V_{i,r}, \forall r \in R_i, d_{i,t} \in D_{i,t}. \end{aligned} \quad (1)$$

由于内部需求  $d_{i,t}$  的不确定性,该最优化问题可视为一个带有随机变量的线性规划问题<sup>[9]</sup>. 目标是通过向公有云用户提供 VM 资源并减去未满足需求代价情况下的最大化收益. 约束条件确保了提供的 VM 资源在给定的内部需求条件下不超过总的可用资源,决策变量  $x_t$  和  $y_t$  分别表示在预约阶段和即付即用阶段提供的 VM 数,决策变量  $z_t$  表示未满足需求的 VM 量.

### 2.2 合作博弈模型

资源池的建立方式为云供应商进行合作建立联盟  $N$ . 对于联盟而言,需求决定两个阶段中资源提供的决策变量. 为了求解决策变量,提出一种可转移效用合作博弈框架进行求解. 该合作博弈中,博弈者为联盟  $N$  中的所有云供应商,博弈策略定义为联盟向公有云用户提供 VM 的策略,即决策变量,收益定义为从资源池中获得的收益. 基于合作博弈的相关定义,将该合作博弈框架定义为  $(N, v(S))$ ,  $S \subseteq N$ ,  $v(S)$  表示合作博弈的特征函数,  $v(S)$  给出子联盟  $S$  所能获得的总效用,也称为联盟  $S$  的价值. 将以上最优化问题以合作博弈的形式表示为:

$$\begin{aligned} v(S) &= \max_{x_t, y_t, z_t} \left( \sum_{t \in T} (\theta_{i,r} x_t) \right) + \sum_{t \in T} (\theta_{i,o} y_t - \theta_{i,d} z_t), \\ \text{s. t.} & \sum_{t \in T} (x_t + y_t - z_t) V_{i,r} \leq b_r(S), r \in \bigcup_{i \in S} R_i, \\ & x_t \geq 0, y_t \geq 0, z_t \geq 0, \end{aligned} \quad (2)$$

其中,  $b_r(S)$  表示子联盟  $S$  向公有云用户提供的  $r$  类型资源的可用资源量,  $x_t$  和  $y_t$  分别表示在两个阶段中子联盟  $S$  组成的资源池提供的 VM 资源量.

**定理 1** 子联盟  $S$  向公有云用户提供的可用资源  $b_r(S)$  满足可加性.

**证明** 由于

$$b_r(S) = \sum_{i \in S} R_{i,r} - \sum_{i \in S} \sum_{t \in T} d_{i,t} V_{t,r} \quad (3)$$

$\sum_{i \in S} R_{i,r}$  表示子联盟  $S$  的资源池中的  $r$  类型资源的总可用量,  $\sum_{i \in S} d_{i,t}$  表示总体内部需求. 由等式右边的求知项可知, 联盟中的可用资源是可加的.

**定理 2** (2) 式定义的资源 and 收益共享的随机线性规划问题是一个线性规划问题.

**证明** (2) 式可等价于

$$\begin{aligned} v(S) = \max_{x_t, y_{t,k}, z_{t,k}} & \left( \sum_{t \in T} (\theta_{t,r} x_t) \right) + \sum_{t \in T} \sum_{i \in S} \left( \sum_{s=1}^{|D_{i,t}|} p(D_{i,t,k}) (\theta_{t,o} y_{t,k} - \theta_{t,d} z_{t,k}) \right), \\ \text{s. t.} & \sum_{t \in T} (x_t + y_{t,k} - z_{t,k}) V_{t,r} \leq b_{r,k}(S), r \in \bigcup_{i \in S} R_i, \\ & x_t \geq 0, y_{t,k} \geq 0, z_{t,k} \geq 0, \end{aligned} \quad (4)$$

其中,  $b_{r,k}(S)$  表示给定内部需求  $k$  时, 子联盟  $S$  向公有云用户提供的  $r$  类型资源量, 可定义为

$$b_{r,k}(S) = \sum_{i \in S} R_{i,r} - \sum_{i \in S} \sum_{t \in T} D_{i,t,k} V_{t,r}, \quad (5)$$

决策变量  $y_{t,k}$  和  $z_{t,k}$  在实际的内部需求量  $D_{i,t,k}$  获得后可以得到.

由于内部需求概率是给定的, 相当于一个常数, 因此, 对于子联盟  $S$ , 以上线性规划问题能以标准矩阵形式进行定义:

$$\begin{cases} v(S) = \max_{\mathbf{x}} \mathbf{x}^T \mathbf{p}, \\ \text{s. t. } \mathbf{x}^T \mathbf{V} \leq \mathbf{b}, \mathbf{x} \geq 0, \end{cases} \quad (6)$$

其中,  $\mathbf{x} = [\dots, x_t, \dots, y_{i,t,k}, \dots, z_{i,t,k}, \dots]^T$  表示决策变量向量,  $\mathbf{p} = [\dots, \theta_{t,r}, \dots, p(D_{i,t,k})\theta_{t,o}, \dots, \dots, p(D_{i,t,k})\theta_{t,d}, \dots]^T$  表示目标函数价格向量,  $\mathbf{V}$  是  $V_{t,r}$  的向量,  $\mathbf{b} = [\dots, b_{r,k}(S), \dots]^T$  表示对于公有云的可用资源向量. 那么, 该线性规划是可行且有界的 (即最优解  $x^*$  存在), 即对于所有的子联盟  $S$ ,  $S \subseteq N \setminus \{\emptyset\}$ :

1) 以上不等式无等式约束; 2) 矩阵  $\mathbf{V}$  是非负的 (即资源需求  $V_{t,r}$  非负).

因此, 该随机线性规划问题是线性规划问题.

### 3 资源最优化分配博弈的解

随机线性规划的解即为联盟  $N$  的收益分配的解. 在这种情况下, 考虑合作博弈的核 (Core).

**定义 1** 合作博弈的核

$$C = \{u = R^{|N|} \mid \sum_{i \in N} u_i = v(N), \sum_{i \in S} u_i \geq v(S), \forall S \subseteq N\}, \quad (7)$$

其中,  $u = [u_1, \dots, u_i, \dots, u_{|N|}]^T$  表示收益分配向量.  $\sum_{i \in N} u_i = v(N)$  表明分配  $u$  是可行的,  $\sum_{i \in S} u_i \geq v(S)$  表明没有比分配方式  $u$  做得更好的子联盟.

核的两个特征使其适合于进行收益分配, 效率和联盟理性. 令  $u_i$  表示提供者  $i$  的收益分配,  $\sum_{i \in N} u_i = v(N)$  表示收益分配是有效的, 直观含义即所有提供者的收益分配必须等于联盟的价值. 联盟理性则表明所有资源提供者的收益分配之和必须等于或大于所有子联盟的价值 (即  $\sum_{i \in S} u_i \geq v(S), S \in N$ ). 换言之, 处于合作核中的分配方式的提供者收益优于不合作方式.

为求解合作博弈的核, 将线性规划问题的解以对偶形式进行求解, 其对偶问题可表示为:

$$\begin{aligned} \min_{e_r^k} & \sum_{t \in T} \sum_{i \in N} \left( \sum_{s=1}^{|D_{i,t}|} \sum_{r \in R} e_{r,k} b_{r,k}(N) \right), \\ \text{s. t.} & \sum_{i \in N} \sum_{s=1}^{|D_{i,t}|} \sum_{r \in R} e_{r,k} V_{t,r} \geq \theta_{t,r}, \\ & \sum_{i \in N} \sum_{s=1}^{|D_{i,t}|} \sum_{r \in R} e_{r,k} V_{t,r} \geq p(D_{i,t,k}) \theta_{t,o}, \\ & \sum_{i \in N} \sum_{s=1}^{|D_{i,t}|} \sum_{r \in R} e_{r,k} V_{t,r} \geq p(D_{i,t,k}) \theta_{t,d}, e_{r,k} \geq 0, \end{aligned} \quad (8)$$

其中,  $R = \cup_{i \in S} R_i$ ,  $e_{r,k}$  表示对偶变量, 表示给定内部需求场景为  $k$  时  $r$  类型资源约束的影子价格, 则提供者  $i$  从联盟  $N$  中获得的收益分配可表示为:

$$u_i^*(N) = \sum_{r \in T} \sum_{i \in N} \left( \sum_{s=1}^{|D_{i,r}|} \sum_{r \in k} e_{r,k}^* (R_{i,r} - \sum_{i \in T} D_{i,t,k} V_{t,r}) \right), \quad (9)$$

其中,  $e_{r,k}^*$  表示对偶问题的最优解, 令  $u^* = [u_1^*(N), \dots, u_i^*(N), \dots, u_{|M|}^*(N)]^T$ , 则  $u^* \in C$ . 即: 收益分配的最优解处于合作博弈的核中. 对于收益分配的解  $u_i^*(N)$ , 资源分配决策变量  $x_i^*$  和  $y_i^*$  可以通过(2)式得到.

同时, 由于建立资源联盟时资源提供者需要承担建立资源池时的额外网络带宽开销, 假设该代价是固定的, 表示为  $C_{i,o}$ , 则联盟中资源提供者  $i$  向公有云用户提供资源的收益可表示为:

$$\phi_i(N) = u_i(N) - C_{i,o}. \quad (10)$$

由于结盟成本的存在, 对于云提供者而言, 若最终收益为负, 那么建立联盟是不可行的.

## 4 联盟形成机制

### 4.1 联盟形成规则

由上节分析可知, 由于结盟成本的存在, 组成联盟并不一定会给所有资源提供者带来更高的个人收益, 某些提供者可能会脱离联盟组织, 以下通过联盟博弈来分析云提供商之间的理性联盟形成过程.

联盟博弈中, 所有云提供者定义为理性的博弈者,  $I$  表示所有云提供商的集合, 也称为大联盟 (Grand Coalition), 一个博弈者联盟表示为  $N \subset I$ , 博弈者的策略定义为是否同意组成联盟, 博弈者的收益为  $\phi_i(\cdot)$ , 由(10)式定义. 联盟结构表示  $I$  中所有博弈者可能组成联盟的集合形式, 定义为:

$$w = \{N_1, \dots, N_l, \dots, N_L\}, \quad (11)$$

其中, 对于  $l \neq l', N_l \cap N_{l'} = \emptyset$ , 且  $\cup_{l=1}^L N_l = I$ , 即子联盟之间没有交集.  $L$  表示联盟结构  $w$  中联盟的总数, 即  $L = |w|$ . 联盟博弈的解即是求解稳定的联盟组成结构  $w$ . 此时, 博弈者对于加入或脱离任一联盟的决策取决于从联盟中获得的收益. 任意博弈者的结盟决策均包含以下两种情况:

1) 联盟分割. 对于给定的初始联盟  $N_i$ , 若满足(12)式,  $N_i$  中的成员可以选择分割形成多个新的子联盟  $N_i^+$ , 分割后的新联盟集表示为  $M^+$ :

$$\phi_i(N_i^+) > \phi_i(N_i), \forall i \in N_i, \quad (12)$$

其中,  $N_i = \cup_{l' \in M^+} N_{l'}^+$ . 直观解释为: 若所有博弈者  $i \in N_i$  的收益比在原始联盟  $N_i$  中多, 博弈者会分割为多个联盟  $N_{l'}^+$ ,  $l' \in M^+$ .

2) 联盟合并. 当满足(13)式的条件时, 联盟集合  $M^{++}$  中的多个联盟  $N_i$  可选择组成一个新的联盟  $N_i^{++}$ :

$$\phi_i(N_i^{++}) > \phi_i(N_i), \forall i \in N_i, \quad (13)$$

其中,  $N_i^{++} = \cup_{l' \in M^{++}} N_{l'}$ . 直观解释为: 若所有候选联盟集合中的所有博弈者通过合并可以获得更高的收益, 这个联盟集合可以组成一个新的联盟.

### 4.2 稳定联盟结构

为了获得稳定的联盟结构, 本文提出一种基于离散 Markov 链结构的动态联盟形成模型. 该 Markov 链的状态空间定义为

$$\Omega = \{(w_k), k = \{1, 2, \dots, B_{|I|}\}\}, \quad (14)$$

其中,  $w_k$  表示一种包含所有博弈者的联盟结构,  $B_{|I|}$  表示贝尔数,

$$B_n = \sum_{m=0}^{n-1} \binom{n-1}{m} B_m, \forall n \geq 1, B_0 = 1. \quad (15)$$

令  $T$  表示 Markov 链的状态转移概率矩阵,  $T_{w,w'} \in T$  表示其元素, 其含义为由所有博弈者组成联盟结构 (状态) 从  $w$  转移为  $w'$  的概率.  $C_{w,w'}$  表示对是否从姿态  $w$  转移为  $w'$  的做出联盟形成决策的候选博弈者集合, 则状态转移概率

$$T_{w,w'} = \begin{cases} \alpha^{|C_{w,w'}|} (1 - \alpha)^{|I| - |C_{w,w'}|} \prod_{i \in C_{w,w'}} \beta_i(w' | w), & w \succ w', \\ 0, & \text{其他,} \end{cases} \quad (16)$$

其中,  $w \succ w'$  表示可行性条件,即,如果在给定所有博弈者的决策的情况下,若联盟结构  $w'$  是可达的,则条件  $w \succ w'$  成立,否则,  $w \succ w'$  不成立.  $\alpha$  表示单个博弈者做出决策的概率,  $\beta_i(w' | w)$  表示博弈者  $i$  (资源提供者) 的最优回应,即,  $\beta_i(w' | w)$  表明博弈  $i$  决定从联盟结构  $w$  转移至  $w'$  的概率

$$\beta_i(w' | w) = \begin{cases} \beta', \phi_i(N_i^{(i)} \subset w') > \phi_i(N_i^{(i)} \subset w) \\ \varepsilon, \text{其他.} \end{cases} \quad (17)$$

其中,  $\beta'$  表示一个常数  $0 < \beta' < 1$ ,  $\varepsilon$  表示博弈者做出非理性决策的概率,为极小值. 此时,状态转移概率  $T_{w,w'}$  由所有做出转移决策和未做出转移决策的博弈者的转移概率的乘积决定. 由于博弈者做出非理性决策的非零性 ( $\varepsilon > 0$ ),由 Markov 链的不可归约和非周期性的特点可知,Markov 链的稳定概率可通过下式得到

$$\begin{cases} \pi^T T = \pi^T, \\ \pi^T \mathbf{1} = 1, \end{cases} \quad (18)$$

其中,  $\pi = [\pi_{w_1}, \pi_{w_2}, \dots, \pi_{w_k}, \dots, \pi_{w_{B_{1,l}}}]$  表示稳定概率向量,  $\pi_w$  表示联盟结构处于状态  $w$  时的稳定概率. 同时,博弈者  $i$  的平均个人收益

$$\bar{\phi}_i = \sum_{k=1}^{B_{1,l}} \pi_{w_k} \left( \sum_{N_l \in w_k} \phi_i(N_l) \right). \quad (19)$$

若非理性决策概率接近 0,即  $\varepsilon \rightarrow 0$ ,则在 (14) 式定义的 Markov 链及 (16) 式定义的状态转移概率中存在状态  $w$  的一个遍历集合  $E \in \Omega$ . 若对于  $w \in E, w' \notin E$ ,有  $T_{w,w'} = 0$ ,则遍历集合  $E$  是存在的. 换言之,单个遍历集可视为 Markov 链的吸收态,即,如果  $T_{w,w'} = 1$ ,则  $w \in \Omega$  吸收.

一旦所有博弈者到达遍历集中的状态 (联盟结构),该状态将保持不变. 因此,吸收态可视为稳定的联盟结构  $w^*$ . 在稳定的联盟结构中,由于此时所有博弈者的个体收益均达到最大化,没有任何博弈者会有意愿改变当前的决策.

云提供者能以最大化总体收益而不是个体收益形成联盟,形成联盟的结果是带来最优的联盟结构,其解表示为以下最优化问题:

$$w^* = \arg \max_w \sum_{N \in w} \sum_{i \in N} \phi_i(N). \quad (20)$$

使用求解稳定联盟结构的方法可以求得该最优联盟结构,不同之处在于 (17) 式可修改为:

$$\beta_i(w' | w) = \begin{cases} \beta', \sum_{N'_l \in w'} \sum_{i \in N'_l} \phi_i(N'_l) > \sum_{N_l \in w} \sum_{i \in N_l} \phi_i(N_l), \\ \varepsilon, \text{其他,} \end{cases} \quad (21)$$

此时吸收态即为定义为最优联盟结构  $w^*$ .

为了求解最优的稳定联盟结构,设计 Algorithm 1 对资源提供的合作博弈过程进行详细描述.

Algorithm 1: Resource provision algorithm in cooperative game

- 1) Input: Resource capacity, users' requirement, prices of two stage, internal demand, cost
- 2) Output: Optimal revenue solution of resource provision
- 3) Initialize parameters of cloud resources
- 4) Initialize parameters of cloud users' resource requirements and internal demand
- 5) For all available cloud resources
- 6) Identify their capability and estimate the load of requirements
- 7) End For
- 8) For each cloud user
- 9) Build revenue objective function and bidding strategy
- 10) Compute revenue and resource allocation sharing
- 11) Save current allocation sharing and resource price
- 12) End For

- 13) Build cloud users coalition to construct a resource pool in local population
- 14) For a resource coalition
- 15) Initialize the number of iterations
- 16) Send coalition bidding to resource
- 17) Calculate allocation sharing and revenue
- 18) Revenue division among the coalition's members
- 19) While revenue is bigger than individual revenue without cooperation Do
- 20) Keep the user coalition
- 21) End while
- 22) While the revenue from a resource pool of coalition is bigger than the coalition cost Do
- 23) Quit the coalition and keep the individual revenue
- 24) End While
- 25) Update iteration

- 26) End For
- 27) Find the core of cooperative game in revenue sharing

28) Return optimal revenue solution of resource provision

## 5 仿真实验

### 5.1 实验环境与参数配置

选取仿真器 CloudSim<sup>[10]</sup>作为算法验证的平台. 为了实现不同资源提供商的资源合作分配, 本文扩展了 CloudSim 内核中的 CloudletScheduler 类、Datacenter 类以及 DatacenterBroker 类, 并重载了 DatacenterBroker 类中的 bindCloudletToVm() 方法. 仿真初始化时, 资源提供商实体均会通过 CIS 进行注册, 当用户请求到达时, CIS 根据用户的应用需求, 从列表中选择合适的服务提供商. 3 个类的数据流如图 1 所示. 实验参数配置如表 1 所示.

表 1 实验参数配置

参数	设置
资源提供者	{CRP1, CRP2, CRP3}
资源能力	$R_{i,c} = 40, R_{i,m} = 80 \text{ GB}, R_{i,s} = 2 \text{ TB}, R_{i,n} = 5 \text{ GB/s}$
资源需求( $t=1$ )	{1 个 CPU, 300 GB 存储, 50 MB/s 带宽}
资源需求( $t=2$ )	{1 个 CPU, 600 GB 存储, 100 MB/s 带宽}
内部需求与概率	$D_{i,j} \in \{2, 10, 15\}, \{(0.8, 0.1, 0.1), (0.3, 0.3, 0.4), (0.3, 0.2, 0.5)\}$
预约阶段	$\theta_{1,r} = 10, \theta_{2,r} = 15$
即付即用阶段	$\theta_{1,o} = 8, \theta_{2,o} = 13$
cpu、内存、硬盘、带宽价格	{3, 2, 1, 1}
代价	$\theta_{1,d} = \theta_{2,d} = 20, C_{i,c} = 10$

### 5.2 实验结果

实验 1 首先考虑两个资源提供者 CRP1 和 CRP2 在合作与不合作时的收益情况. 图 2 所示横坐标表示 CRP1 的可用带宽资源的变化. 可以看出, 随着带宽资源的增加, 无论是否与 CRP2 合作, CRP1 的收益将增加, 这是由于更多的 VM 可以提供给公有云用户. 但是, 达到确定点时, 其收益趋近于常数, 因为此时存在资源瓶颈(此时为 CPU 资源). 同时, 在带宽小于  $4.4 \text{ GB} \cdot \text{s}^{-1}$  时, 合作时 CRP1 的收益高于不合作时. 由于 CRP1 能够联合 CRP2 的可用资源向外提供服务, 合作会获得更多的个人收益. 然而, 若 CRP1 的可用带宽大于  $4.4 \text{ Gbps}$ , CRP1 可以从自身的可用资源中获得更多的个人收益, 它没有必要与 CRP2 共享资源. 因此, 每个合作的提供者收益均低于不合作的收益.

若不合作, CRP1 的可变可用资源对于 CRP2 没有影响. 若合作, CRP1 的可用带宽资源将影响 CRP2 的收益. 随着可用带宽资源的增加, 由于 CRP1 从服务提供中获得了更多的收益, CRP2 的收益将降低. 但合作时个人收益仍大于不合作时.

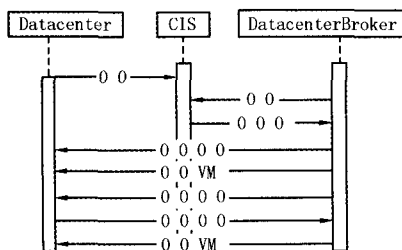


图 1 主要仿真类数据流

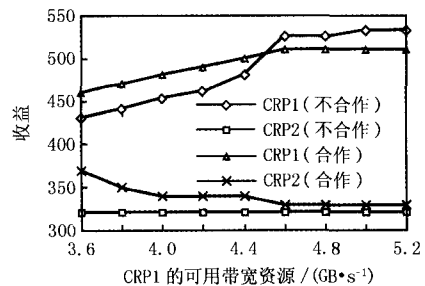


图 2 CRP1 和 CRP2 的收益变化

实验 2 研究了内部需求对资源提供者收益的影响, 如图 3 所示. 随着内部需求概率的增加, CRP1 向公

有云用户提供更多 VM 资源的概率将增大,这将带来更多收益. 结果是,无论合作与否,其个人收益都将增加. 同时,若不合作,该概率对 CRP2 的收益没有影响. 然而,合作时,CRP2 能够获得更多收益,这与 CRP1 的内部需求变小的概率增大相关. 同时,可以看到,在概率[0.26,0.53]之间,CRP1 和 CRP2 通过合作都可以增加收益. 在这个范围内,两个资源提供者可以形成稳定的联盟结构.

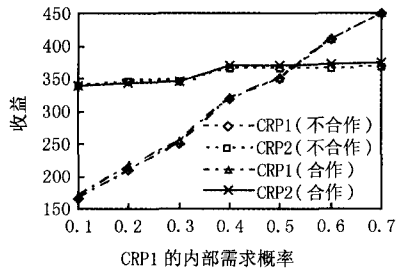


图 3 给定内部需求时 CRP1 和 CRP2 的收益变化

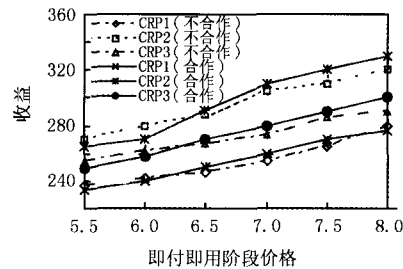


图 4 价格变化对收益的影响

实验 3 考察了 3 个资源提供者的收益随着第二阶段资源价格的变化,如图 4 所示. 合作表示为  $\{CRP1, CRP2, CRP3\}$ , 不合作表示为  $\{\{CRP1\}, \{CRP2\}, \{CRP3\}\}$ . 可以看出,随着价格的升高,资源提供者的收益均会增加. 另外,当价格较低时,对于资源提供者而言没有必要合作,由于结盟成本可能高于获取的联盟收益. 同时,当价格较高时,从联盟中获取的收益高于结盟成本,结盟是一个更好的选择.

实验 4 观察了 3 个资源提供者 CRPs 在不同的联盟结构中的收益分割情况. 本文重点考察:如果通过合作形成联盟可以提高个体收益,理性的资源提供者将选择形成联盟,同时考察联盟结构的稳定性问题. 本节实验改变结盟代价以观察不同的联盟结构中个体收益的分配情况. 如表 2 所示结盟代价为 0 时的收益分配.

表 2 代价为 0 时不同联盟结构下收益分割

CRPs 联盟结构	CRP1	CRP2	CRP3
$w = \{\{CRP1\}, \{CRP2\}, \{CRP3\}\}$	524	320	296
$w^* = \{\{CRP1, CRP2, CRP3\}\}$	547	337	319
$w^* = \{\{CRP1, CRP2\}, \{CRP3\}\}$	527	344	296
$w^* = \{\{CRP1, CRP3\}, \{CRP2\}\}$	527	320	326
$w = \{\{CRP2, CRP3\}, \{CRP1\}\}$	524	331	309

基于本文提出的算法,当结盟代价为 0 时共产生了 3 组稳定的联盟结构: $w^* = \{\{CRP1, CRP2\}, \{CRP3\}\}$ ,  $w^* = \{\{CRP1, CRP3\}, \{CRP2\}\}$  和  $w^* = \{\{CRP1, CRP2, CRP3\}\}$ . 对于联盟结构  $w^* = \{\{CRP1, CRP2\}, \{CRP3\}\}$ , 由于 CRP1 和 CRP2 通过建立联盟  $\{CRP1, CRP2\}$  获得了更高的个体收益(即此时的个体收益大于  $\{CRP1\}, \{CRP2\}$  结构中的个体收益), 因此其联盟结构是稳定的. 同样,在  $w^* = \{\{CRP1, CRP3\}, \{CRP2\}\}$  中, CRP1 和 CRP3 通过建立联盟获得了更高的个体收益.  $w^* = \{\{CRP1, CRP2, CRP3\}\}$  通过形成大联盟的形式使得所有资源提供均获得更高的个体收益. 而对于  $w = \{\{CRP2, CRP3\}, \{CRP1\}\}$ , 由于 CRP1 可以通过加入联盟  $\{CRP2, CRP3\}$  得到更高的收益,其联盟结构是不稳定的.

当结盟代价为 10 时,个体收益情况如表 3 所示. 此时,仅有一个稳定联盟  $w^* = \{\{CRP1, CRP2, CRP3\}\}$ . 可以看出,所有资源提供者均无法通过形成更小的联盟结构而得到更高的个体收益. 通常情况下,随着联盟代价的增加,稳定的联盟结构将具有更小的联盟规模,可以解释为:对于 CRPs 而言,以合作形式向公有云用户提供资源将会付出更高昂的代价.

## 6 结束语

本文提出了一种合作式云资源提供算法. 首先,将资源提供者的收益最优化问题建立为一个线性规划问题,进而通过合作博弈的核求解了这个线性规划问题,得到了资源提供者的资源提供策略. 结果表明,相互协

作组建资源联盟可以提高资源提供者的收益,但若结盟代价过高,资源提供者仍会以个人为单位提供资源.进一步的研究将集中在分析联盟的稳定性以及如何构建稳定的资源提供者联盟等问题上.

表3 代价为10时不同联盟结构下收益分割

CRPs 联盟结构	CRP1	CRP2	CRP3
$w = \{\{CRP1\}, \{CRP2\}, \{CRP3\}\}$	524	320	296
$w^* = \{\{CRP1, CRP2, CRP3\}\}$	537	327	309
$w = \{\{CRP1, CRP2\}, \{CRP3\}\}$	517	334	296
$w = \{\{CRP1, CRP3\}, \{CRP2\}\}$	517	320	316
$w = \{\{CRP2, CRP3\}, \{CRP1\}\}$	524	321	299

### 参 考 文 献

- [1] Cusumano M. Cloud computing and SaaS as new computing platform[J]. Communication of the ACM, 2011, 53(4): 27-29.
- [2] CHUN L, LA L. Optimal resource provisioning for cloud computing environment[J]. Journal of Supercomputing, 2012, 62(2): 989-1022.
- [3] 泰迪里斯. 博弈论导论[M]. 北京:中国人民大学出版社, 2015: 34-38.
- [4] Guiyi W, Vasilakos A V, Naixue X. Scheduling Parallel Cloud Computing Services: An Evolutional Game[C]//in 2009 1st International Conference on Information Science and Engineering. Washington D C: IEEE Computer Society, 2009: 376-379.
- [5] 朱匆, 刘元君, 彭自然, 等. 移动云计算中基于协作式博弈模型的资源分配方案[J]. 计算机应用研究, 2014, 31(3): 912-916.
- [6] Duan R, Prodan R, Li X. Multi-Objective Game Theoretic Scheduling of Bag-of-Tasks Workflows on Hybrid Clouds[J]. IEEE Trans on Cloud Computing, 2014, 2(1): 29-42.
- [7] 陈亚睿, 田立勤, 杨扬. 云计算环境下基于动态博弈论的用户行为模型与分析[J]. 电子学报, 2011, 9(8): 1818-1823.
- [8] 李卫平, 武海燕, 杨杰. 基于效益博弈的云计算资源动态可协调分配策略研究[J]. 计算机工程与科学, 2016, 38(1): 57-61.
- [9] 束金龙, 闻人凯. 线性规划理论与模型应用[M]. 北京: 科学出版社, 2016: 57-78.
- [10] Rodrigo C, Rajiv R, Anton B, et al. CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms[J]. Software: practice and experience, 2011, 41(1): 23-50.

## Cloud Resource Provision Optimization Algorithm in Federation Mode

Xu Lijun<sup>1,2</sup>, Gao Guohong<sup>1</sup>, Zhang Xiaoqing<sup>3</sup>

(1. School of Computer, Wuhan University of Technology, Wuhan 430063, China; 2. School of Computer and Information Engineering, Xixiang University, Xixiang 453003, China; 3. School of Information Engineering, Henan Institute of Science & Technology, Xixiang 453003, China; 4. School of Mathematics and Computer Science, Wuhan Polytechnic University, Wuhan 430023, China)

**Abstract:** In order to improve the resource utilization and the resource providers' revenue, a resource provision algorithm based on cooperative game in cloud coalition mode is proposed. The revenue optimization for the resource coalition is formalized as a linear programming problem in our algorithm and the optimization problem is solved by the core in cooperative game. Then, we prove that the resource provision strategy in the core of cooperative game is also the optimal solution of linear programming. Meanwhile, for obtaining the stable resource coalition structure, a dynamic coalition formation model based on discrete Markov chain is designed, in which the optimal coalition structure maximizing resource provider's profit can be obtained. Simulation experimental results show that, for the resource providers with game rationality, the forming resource coalition can improve the resource providers' profit. And, due to the coalitional formation cost, the stable coalition structure prefers to selecting a smaller size of coalition.

**Keywords:** cloud computing; resource provision; cooperative game; coalitional structure