

基于任务—资源分配图的虚拟企业伙伴优选算法^{*}

龙浩, 汪浩

(江西师范大学软件学院, 南昌 330029)

摘要: 针对时间成本均衡的虚拟企业伙伴选择问题, 建立了统一描述虚拟企业过程和资源的项目配置图, 并以任务—资源分配图作为调度模型, 采用基于相对费效比的启发式算法迭代求解。算法考虑了虚拟伙伴的制造和转运时间及费用, 在保证截止期约束的同时能有效降低总成本。实际算例和大量模拟实验证明了该方法能有效降低获取优化方案的运算时间。

关键词: 虚拟企业; 伙伴选择; 项目配置图; 任务—资源分配图; 相对费效比

中图分类号: TP391; TP301.6 **文献标志码:** A **文章编号:** 1001-3695(2013)12-3564-04

doi: 10.3969/j.issn.1001-3695.2013.12.011

Task-resource assignment graph based heuristic for partner selection in virtual enterprises

LONG Hao, WANG Hao

(School of Software, Jiangxi Normal University, Nanchang 330029, China)

Abstract: In order to optimize the partner selection with time-cost balance constraints in virtual enterprises, this paper used project deployment diagram (PDD) to describe the relation between available resources and the workflow process, used task-resource assignment graph (TRAG) to reflect a special scheduling scheme, which was a sub-structure of PDD and embody the mapping between tasks of workflow and the corresponding selected resources, and proposed a novel heuristics based on relative time-cost rate (RTCR) to solve the problem. This method considered cost and time in manufacturing and transferring stage, and proved its effectiveness in decreasing computing time by a practical example and experimental results.

Key words: virtual enterprise; partner selection; project deployment diagram; task-resource assignment graph; relative time-cost rate

虚拟企业是21世纪企业进行生产经营和市场竞争的主要模式, 由于具有成本低、快速响应市场机遇以及良好的适应能力等优势, 越来越受到业界的重视。如何从大量服务质量(如成本因素、生产时间、产品质量、合作信任度、风险因素等)各异的潜在合作者中选择最适当的企业进行产品生产, 是虚拟企业成功的关键。依据关注的不同服务质量因素, 虚拟企业的伙伴选择问题各不相同, 但其中的时间费用优化问题是最常见的, 它考虑在一个给定的截止期约束下, 如何最小化虚拟企业的总费用。Zeng等人^[1]证明这是一个NP-hard问题。很多学者^[2-6]使用遗传算法、蚁群算法、免疫算法、粒子群算法来解决这个问题, 能获得较好的性能, 但当问题规模较大时, 算法的时间开销很大。启发算法能够在合理的运行时间内获得问题的可接受解, 被广泛用于虚拟企业的伙伴选择。尚耀华等人^[7]以总费用最低为目标, 提出了基于图论的伙伴优选方法。文献^[8]考虑候选伙伴的资源和虚拟企业过程, 建立有向超图模型描述虚拟企业过程, 应用超图理论求解伙伴调度问题。Wu等人^[9]基于网络优化模型, 提出了一种考虑了制造和转运费用的伙伴优选算法, 但它们都没有考虑截止期约束。文献^[10]基于虚拟企业过程的网络图分解, 利用列生成技术给出虚拟企业伙伴选择的一种上下界求解方法, 该算法考虑了项目的拖期惩罚, 但没有考虑工件的转运时间和费用。

针对基于过程模型分析和图论思想的伙伴优选算法的不足, 本文基于图论和相对费效比^[11]思想, 提出一种新的启发式伙伴优选算法。该算法首先建立了统一描述虚拟企业过程模型和资源模型的项目配置图, 从项目配置图中任选一个任务—资源分配图^[12]作为初步解, 围绕截止期逐步调整合作伙伴, 不断将成本降低最快的伙伴替换进来, 把时间成本增长最快的伙伴替换出去。

1 问题描述

虚拟企业项目可用二元组 $\langle \text{DAG}, R \rangle$ 表示。DAG (directed acyclic graph, 有向无环图) 描述过程模型, $\text{DAG} = \langle V, L \rangle$ 。其中: $V = \{v_i | 1 \leq i \leq n\}$ 是相互独立、有一定逻辑关系的子项目集合。 $L = \{l_{ij} | i \geq 1, j \leq n, i \neq j\}$ 是子项目相互关系的集合, $l_{ij} = (v_i, v_j)$ 表示子项目 v_i 与 v_j 执行上的时序关系, 即子项目 v_j 在 v_i 执行完成之后才能开始, 这时称 v_i 为 v_j 的紧前子项目, v_j 为 v_i 的紧后子项目。 $R = \langle E, D \rangle$ 描述资源模型, $E = \bigcup_i^n E_i = \bigcup_i^n \{e_{iu} | 1 \leq u \leq m_i\}$ 是候选企业集合, $m = |\bigcup_i E_i|$ 是虚拟企业伙伴总数, $||$ 是集合的势运算符。子项目 v_i 的候选企业集合为 E_i , m_i 是子项目 v_i 的候选企业数, $e_{iu} = (c_{iu}, t_{iu})$ 表示子项目 v_i 的候选企业 u 的 QoS 属性描述。本文主要关注加工成本 c_{ik} 与

收稿日期: 2013-01-28; 修回日期: 2013-03-13 基金项目: 国家自然科学基金资助项目(61262014)

作者简介: 龙浩(1970-), 男, 江西泰和人, 副教授, 博士, 主要研究方向为 workflow、服务计算(hhlong2010@hotmail.com); 汪浩(1962-), 男, 江西南昌人, 教授, 博士, 主要研究方向为服务计算、网络性能分析与优化。

时间 t_{ik} 。 $D = \{d_{iu,jv} | 1 \leq i, j \leq n, 1 \leq u \leq m_i, 1 \leq v \leq m_j, l_{ij} \in L\}$, $d_{iu,jv} = (c_{iu,jv}, t_{iu,jv})$ 表示子项目 v_i 在伙伴企业 u 、子项目 v_j 在伙伴企业 v 加工时工件在两个伙伴之间的转运成本和时间。本文假设每个虚拟企业伙伴加工能力没有限制,但任意时刻只能加工一个子项目,任意两个伙伴之间都有传输线路,传输能力没有限制,但任意时刻只能有一种工件转运。

伙伴优选的目标是为每个子项目选择一个候选伙伴,使项目在截止期内以最低成本完成。问题的形式化描述如下:

$$\min_{u,v} (\sum_i x_{iu} c_{iu} + \sum_j \sum_i x_{iu} x_{ju} c_{iu,jv}) \quad 1 \leq i, j \leq n, 1 \leq u \leq m_i, 1 \leq v \leq m_j \quad (1)$$

$$\text{s. t.} \quad \sum_{u=1}^{m_i} x_{iu} = 1 \quad 1 \leq i \leq n \quad (2)$$

$$f_{iu} + x_{iu} t_{iu} + x_{iu} x_{ju} t_{iu,jv} \leq f_{ju} \quad l_{ij} \in L; 1 \leq i, j \leq n; 1 \leq u \leq m_i; 1 \leq v \leq m_j \quad (3)$$

$$(f_{iu} - x_{iu} t_{iu} - x_{iu} x_{ju} t_{iu,jv} - f_{ju}) \times (f_{ju} - x_{ju} t_{ju} - x_{ju} x_{iu} t_{ju,iv} - f_{iu}) \geq 0$$

$$1 \leq u \leq m_i; 1 \leq i, j \leq n \quad (4)$$

$$f_{0u} = 0, f_{nu} = T \quad 1 \leq u \leq m \quad (5)$$

$$x_{iu} = \{0, 1\}, 0 \leq i \leq n, 1 \leq u \leq m_i \quad (6)$$

其中: x_{iu} 的取值表示子项目 v_i 是否由伙伴 u 加工; f_{iu} 表示子项目 v_i 在伙伴 u 的完工时间。式(1)是伙伴选择方案的成本,是优化问题的目标;式(2)表示各子项目只能在一个伙伴中加工;式(3)表示满足偏序约束的子项目加工及转运完成时间必须满足的条件;式(4)是共享资源约束,表示任何时刻,任一伙伴不能同时加工多个子项目;式(5)是子项目的完工时间,式(6)说明 x_{iu} 是布尔变量。

2 虚拟企业伙伴选择算法

2.1 相关定义

为了直观反映不同子项目的虚拟伙伴加工能力、转运性能、可能存在的共享资源冲突,首先将虚拟企业项目过程模型

和资源模型转换为项目配置图。

定义 1 给定图 $DAG = \langle V, L \rangle$, 子项目 v_i 的深度 $BD(i)$ 定义为 v_i 到出口节点 v_n 的最长路径长度(路径长度指边的个数)。

$$BD(i) = \begin{cases} 1 & i = 1 \\ \max_{j \in \text{succ}(i)} \{BD(j) + 1\} & \text{otherwise} \end{cases} \quad (7)$$

定义 2 虚拟企业项目 $\langle DAG, R \rangle$ 的项目配置图 $G = \langle V', L', E', D' \rangle$ 是一个层数为 $BD(n)$ 的多层网络图,第 k 层的网络节点数 $n_k = \{ \cup_j E_j | BD(j) = k \}$; 如果前后相邻层有相同节点,它们之间的工件传输时间和成本均为 0; 如果 $d_{iu,jv} \in D(l_{ij} \in L, DB(i) = k, DB(j) = k + 1)$, 则 k 层 u 节点与 $k + 1$ 层的 v 节点之间存在网络连接,子项目 i, j 在 u, v 节点上的加工时间和成本分别为 c_{iu}, c_{ju} 和 t_{iu}, t_{ju} , 转运成本和时间分别为 $c_{iu,jv}, t_{iu,jv}$ 。

项目配置图能够统一描述虚拟企业项目和子项目的候选企业伙伴加工性能、工件在候选企业伙伴之间的转运性能。

定义 3 任务—资源分配图(task-resource assignment graph, TRAG)是项目配置图上满足 DAG 约束和资源约束的一个子图,每一个任务—资源分配图都是虚拟企业伙伴选择问题的一个解。如果每个任务都选择加工时间最短的虚拟伙伴,转运时间假设为 0,则该方案的关键路径为最小关键路径(minimum critical path, MCP),其路径长度称为最小完工时间(fastest time)。

定义 4 设 u_1, u_2 分别是第 j 层上的 j_1 子项目可选的两个网络节点,任务—资源分配图 TRAG 中 j_1 子项目的紧前子项目 i_1 、紧后子项目 k_1 分别由第 i 层网络节点 i_p 、第 k 层的 k_q 承担,第 i 层网络节点 i_p 与第 j 层上的 u_1, u_2 均有直接网络连接,第 j 层上的 u_1, u_2 与第 k 层的 k_q 也均有网络连接。将 j_1 子项目从 u_1 节点迁移到 u_2 节点时任务—资源分配图变为 TRAG', 则相对效费比 $r(j_1, (u_1, u_2))$ 是任务资源分配图变化后总体成本变化与完工时间变化之比,即

$$r(j_1, (u_1, u_2)) = \begin{cases} +\infty & c_{j_1, u_1} + \sum_{i_1} c_{i_1 i_p, j_1 u_1} + \sum_{k_1} c_{j_1 u_1, k_1 k_q} > c_{j_1, u_2} + \sum_{i_1} c_{i_1 i_p, j_1 u_2} + \sum_{k_1} c_{j_1 u_2, k_1 k_q} \text{ 且 } CP(\text{TRAG})_{j_1 \rightarrow u_1} = CP(\text{TRAG}')_{j_1 \rightarrow u_2} \\ -\infty & c_{j_1, u_1} + \sum_{i_1} c_{i_1 i_p, j_1 u_1} + \sum_{k_1} c_{j_1 u_1, k_1 k_q} < c_{j_1, u_2} + \sum_{i_1} c_{i_1 i_p, j_1 u_2} + \sum_{k_1} c_{j_1 u_2, k_1 k_q} \text{ 且 } CP(\text{TRAG})_{j_1 \rightarrow u_1} = CP(\text{TRAG}')_{j_1 \rightarrow u_2} \\ 0 & c_{j_1, u_1} + \sum_{i_1} c_{i_1 i_p, j_1 u_1} + \sum_{k_1} c_{j_1 u_1, k_1 k_q} = c_{j_1, u_2} + \sum_{i_1} c_{i_1 i_p, j_1 u_2} + \sum_{k_1} c_{j_1 u_2, k_1 k_q} \\ \frac{c_{j_1, u_1} + \sum_{i_1} c_{i_1 i_p, j_1 u_1} + \sum_{k_1} c_{j_1 u_1, k_1 k_q} - (c_{j_1, u_2} + \sum_{i_1} c_{i_1 i_p, j_1 u_2} + \sum_{k_1} c_{j_1 u_2, k_1 k_q})}{CP(\text{TRAG}')_{j_1 \rightarrow u_2} - CP(\text{TRAG})_{j_1 \rightarrow u_1}} & \text{otherwise} \end{cases} \quad (8)$$

其中: $CP(\text{TRAG})_{j \rightarrow u}$ 表示子任务 j 分配到伙伴 u 时 TRAG 的关键路径长度。

相对效费比 $r(j_1, (u_1, u_2))$ 体现了整个项目在其他子项目调度不变时,子项目 j_1 从第 j 层上的 u_1 网络节点变换到 u_2 节点时调度方案的成本时间变化。 $r(j_1, (u_1, u_2)) < 0$, 加工节点的变化导致的总体时间变化和总体成本变化是相同的,选择其中时间减少的候选伙伴,可以同步降低总体成本; $r(j_1, (u_1, u_2)) = -\infty$ 时,候选伙伴的变换只能使总体成本增加而关键路径长度不变; $r(j_1, (u_1, u_2)) > 0$, 加工节点的变化导致的关键路径时间变化和总体成本变化是相异的,选择使关键路径长度增加的候选伙伴,可以降低成本,选择成本增加的候选伙伴,可以减少时间,且相对效费比值越大,效果越显著; $r(j_1, (u_1, u_2)) = \infty$ 时,候选伙伴的变换会导致总体成本减少而总体时间不变。

虚拟企业伙伴选择,就是围绕给定的截止期,在项目配置图上不断构建最优任务—资源分配图的过程。在任何时候,都

应当无条件选择相对效费比为负值且关键路径长度减少或相对效费比为 ∞ 的虚拟伙伴的子项目进行调整;当虚拟项目完工时间低于截止期时,选择且有最大正值相对效费比且关键路径长度增加的子项目进行调整;当虚拟项目完工时间超过截止期时,选择最大正值相对效费比且关键路径长度减少的子项目进行调整。

2.2 算法描述

算法 1 DAG 到项目配置图的转换算法

输入: DAG、矩阵 E 和 D 。

输出: 项目配置图。

a) 按照式(7)计算 DAG 中各个子项目的深度。

b) 根据 $BD(n)$ 确定项目配置图的层数。

c) 根据矩阵 E 确定项目配置图各层的虚拟企业伙伴。

d) 为相邻前后层的相同节点添加网络连线。

e) 按照矩阵 D , 如果 $d_{iu,jv} \in D(l_{ij} \in L, DB(i) = k, DB(j) = k + 1)$, 则 k 层 u 节点与 $k + 1$ 层的 v 节点之间建立网络连接。

f) 输出项目配置图。

算法 2 基于图和相对效费比的伙伴选择算法

输入:项目配置图 G 、用户给定的截止期 T 。

输出:虚拟伙伴选择方案。

a) 计算项目配置图 G 的相对效费比。

b) $k_0 = 0$; 随机生成一个任务—资源分配图, 作为初步的伙伴选择方案, 计算任务—资源分配图的关键路径时间 T_s 。

c) 查找任务—资源分配图上选择相对效费比为负值且关键路径长度减少或相对效费比为 ∞ 的虚拟企业的子项目, 调整调度方案, 计算任务—资源分配图的关键路径时间 T_s 。

d) 如果 $T_s > T$, 调整任务—资源分配图中具有最大正值相对效费比且关键路径长度减少的子项目伙伴; 如果 $T_s \leq T$, 调整任务—资源分配图中具有最大正值相对效费比且关键路径长度增加的子项目伙伴。如果调度中最大正值相对效费比序列内出现循环子序列, 转步骤 e); 否则转步骤 c)。

e) 选择完工时间低于且最接近截止期的任务—资源分配图作为最优的伙伴选择方案并输出。

性质 1 相对效费比调度方法是收敛的。

证明 不失一般性, 假设虚拟企业关键路径上某子项目不同调度方案的时间成本如图 1 所示。

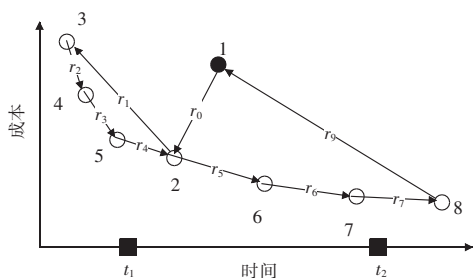


图 1 虚拟企业某子项目 j 不同调度方案的时间成本

假设调度方案中其他子项目不存在相对效费比为负值且关键路径长度减少或相对效费比为 ∞ 的虚拟伙伴, 它们的所有正值相对效费比均小于 r_1, r_2, r_3, r_4 , 子项目 j 的加工及工件传输时间限制为 t_1 , 伙伴 1 是子项目 j 的当前选择, 此时方案完工时间超过截止期, 且调度方案下相对效费比 $r(j, (1, 2)), r(j, (1, 5)), r(j, (1, 4))$ 小于 0 且 $r(j, (1, 2))$ 最小, $r(j, (1, 6)), r(j, (1, 7)), r(j, (1, 8))$ 大于 0。显然选择伙伴 2 可以同步降低成本和时间, 此时方案完工时间超过截止期。继续选择相对效费比为正且最大(r_1)的伙伴 3, 方案完工时间低于截止期; 继续选择具有最大正值相对效费比(r_2)的伙伴 4, 方案完工时间依然低于截止期; 选择具有最大正值相对效费比(r_3)的伙伴 5, 方案完工时间依然低于截止期; 选择具有最大正值相对效费比(r_4)的伙伴 2, 此时方案完工时间超过截止期, 选择相对效费比为正且最大(r_3)的伙伴 3。可以看出, 调度中最大正值相对效费比序列内出现了循环子序列(r_1, r_2, r_3, r_4), 完工时间低于且最接近截止期的伙伴 5 为最终方案中该子项目的伙伴选择。

同理, 如果该子项目的伙伴加工时间为 t_2 , 其他子项目不存在相对效费比为负值且关键路径长度减少或相对效费比为 ∞ 的虚拟伙伴, 它们的所有正值相对效费比小于 r_6, r_7, r_8, r_9 。调度中最大正值相对效费比序列内将出现正值相对效费比循环子序列(r_6, r_7, r_8, r_9), 完工时间低于且最接近截止期的伙伴 7 为最终该子项目的伙伴选择。

2.3 算例

某虚拟企业项目由六个子项目组成, 它们的时序关系如图 2 所示。各子项目分别由六个候选企业中的 3、2、3、3、3、2 个候选企业竞标。这些候选企业的加工性能和转运性能分别用表 1 和式(9)描述。用户给定的截止期为 18。

1	2	3	4	5	6
(10, 5)	-	-	(8, 4)	-	(9, 4)
-	(4, 2)	(8, 2)	-	-	-
-	-	(18, 4)	-	(4, 2)	(5, 2)
(25, 5)	-	(20, 6)	-	-	(33, 6)
(11, 3)	-	-	-	(12, 3)	(9, 3)
-	(10, 3)	(13, 3)	-	-	-

表 1 虚拟企业项目的 D 矩阵

AB 1 2	(3, 1)	AC 1 3	(3, 1)	BE 2 1	(1, 1)	CE 3 1	(3, 1)	DF 1 2	(3, 2)
1 3	(3, 2)	1 5	(2, 2)	2 5	(1, 2)	3 5	(2, 1)	3 2	(2, 2)
4 2	(4, 1.5)	1 6	(2, 1.5)	2 6	(2, 2)	3 6	(2, 2)	6 2	(2, 1)
4 3	(2, 2)	4 3	(3, 2)	3 1	(2, 1)	5 1	(2, 1)	1 3	(1, 1)
6 2	(2, 1)	4 5	(1, 1)	3 5	(2, 1.5)	5 5	(0, 0)	3 3	(0, 0)
6 3	(4, 3)	4 6	(3, 2)	3 6	(2, 1)	5 6	(2, 1)	6 3	(3, 2)
BD 2 1	(2, 2)	6 3	(3, 1)			6 1	(2, 2)	EF 1 2	(1, 1)
2 3	(2, 1)	6 5	(3, 2)			6 5	(2, 1)	1 3	(3, 2)
2 6	(3, 2)	6 6	(0, 0)			6 6	(0, 0)	5 2	(3, 1.5)
3 1	(1, 1)							5 3	(3, 1)
3 3	(0, 0)							6 2	(3, 2)
3 6	(2, 1)							6 3	(2, 1)

假设虚拟企业的初始任务—资源分配图如图 3 所示, 则总花费与时间分别为 76 和 20, 子项目 A 的相对效费比 $r(A, (1, 4)) = -4, r(A, (1, 6)) = -\infty$ 。

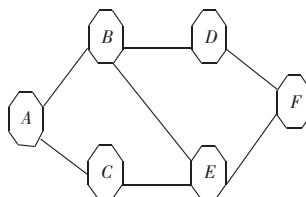


图 2 虚拟企业项目的 DAG 图

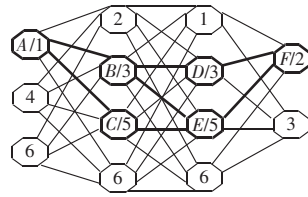


图 3 虚拟企业的项目配置图与初始任务—资源分配图

由于该子项目存在时间减少且负值相对效费比的候选伙伴 4, 将子项目 A 的伙伴选择为 4。任务—资源分配图见图 4(a), 总花费与时间分别为 72 和 19。

子项目 C 的相对效费比 $r(C, (5, 3)) = -6$, 但伙伴 3 的时间增加; $r(C, (5, 6)) = -2.5$, 但伙伴 6 的时间增加, 所以子项目 C 的伙伴先不作调整; 子项目 B 的相对效费比 $r(B, (3, 2)) = 2$, 不存在负值相对效费比的候选伙伴, 子项目 B 的伙伴先不作调整; 子项目 D 的相对效费比 $r(D, (3, 1)) = r(D, (3, 6)) = -\infty$, 因为成本增加, 所以子项目 D 的伙伴也先不作调整; 子项目 E 的相对效费比 $r(E, (5, 1)) = r(E, (5, 6)) = -\infty$, 所以子项目 E 的伙伴也先不作调整; 子项目 F 的相对效费比 $r(F, (2, 3)) = 0.5$, 不存在负值相对效费比且关键路径长度减少或无穷大正值的候选伙伴, 子项目 F 的伙伴也先不作调整。

由于方案完工时间超过截止期, 选择相对效费比为正值, 且时间减少、成本增加的子项目 B 调整, 将子项目 B 的伙伴选为 3, 总花费与时间分别为 73 和 17。调度结果见图 4(b)。

由于方案完工时间低于截止期, 选择相对效费比正值最大, 且时间增加、成本减少的子项目 F , 将子项目 F 的伙伴选为 3, 总花费与时间分别为 72 和 17.5。调度结果见图 4(c)。

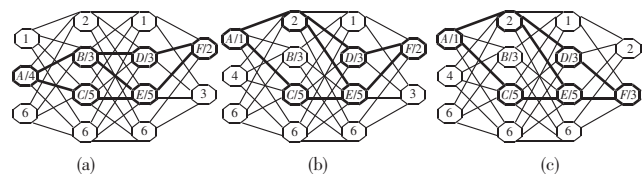


图 4 调度方案的调整过程

方案中不存在相对效费比为负且时间减少的子项目, 也不存在相对效费比正值并保证关键路径长度低于 18 的子项目, 调整结束。

3 实验结果与分析

为了评估算法的性能,对不同 DAG 的虚拟企业应用进行模拟测试,并与分支定界法的性能和效率进行比较。实验仿真环境为 PIV 2.6 GHz, 512 MB, 操作系统为 Windows 2000。虚拟企业过程模型采用与文献[11]类似的 DAG 自动生成器,节点数由用户输入。虚拟伙伴数采用固定值 5。子项目 v_i 在候选企业 u 的加工成本 c_{ik} 与时间 t_{ik} 分别是种子为 20 和 10 的随机数;子项目 v_i 在伙伴企业 u 、子项目 v_j 在伙伴企业 v 加工时工件在两个伙伴之间的转运成本和时间 $c_{iu,jv}$ 和 $t_{iu,jv}$ 分别是种子为 10 和 5 的随机数。给定的完工时间取 $2 \times \text{MCP}$ 。每种节点数的 DAG(分别为 10、15、20、25、30)各运行 10 次,取其平均的成本耗费和运算时间进行比较。

图 5 描述了不同节点下两种算法最终调度方案的成本耗费和运算时间。其中圆点线是本文算法结果,从低到高依次为 10、15、20、25、30 五种 DAG 节点时的最优成本耗费和运算时间;方框线是分支定界法的结果,从低到高依次为 10、15、20、25、30 五种 DAG 节点时的最优成本耗费和运算时间。

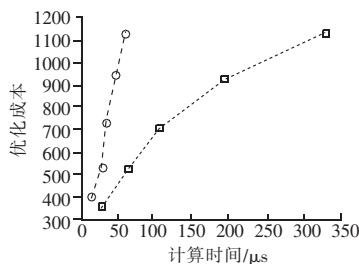


图 5 两种算法获得最终方案的运算时间与成本耗费

从图中可以看到两种算法的比较效果:

a) 随着 DAG 节点数的增加,本文算法的计算时间呈线性递增,而分支定界法的计算时间呈指数递增。这是因为本文算法在完工时间超过截至期时,总是选择成本下降最快的虚拟伙伴加入;在完工时间低于截至期时,总是选择加工时间增长最慢的虚拟伙伴加入,并将对应任务的原伙伴替换出去。

b) 对于同样节点数的 DAG 而言,分支定界法得到的最终成本均优于本文算法,但随节点数的增加,两者之间的差距越来越小。这是因为在不同的 DAG 下,本文算法最终方案的完工时间与给定的截至期有一定的时间碎片,无法以成本进行弥

补;随着 DAG 节点数的增加,这个空闲的时间碎片变得很小,成本弥补的效果也越来越低。

4 结束语

存在工件转运时间和费用的虚拟企业伙伴选择是典型的 NP-hard 问题。本文将虚拟企业伙伴优选问题建模为项目配置图,将任务—资源分配图作为求解方案,给出了基于相对费效比的启发式算法。算法在伙伴优选过程中围绕给定的截止期逐步调整任务—资源分配图,实现优化求解。通过实例验证了模型及其求解算法的有效性。

参考文献:

- [1] ZENG Zhi-bin, LI Yan, ZHU Wen-xing. Partner selection with a due date constraint in virtual enterprises [J]. *Applied Mathematics and Computation*, 2006, 175(2): 1353-1365.
- [2] 贾瑞玉, 潘雯雯, 刘范范. 粗糙集与遗传算法的虚拟企业伙伴选择 [J]. *哈尔滨工程大学学报*, 2012, 33(6): 730-734.
- [3] 张以文, 倪志伟, 宋捷, 等. 云计算环境下动态虚拟企业伙伴选择模型 [J]. *计算机科学*, 2011, 38(7): 212-215.
- [4] 王正成, 潘晓弘, 潘旭伟. 基于蚁群算法的网络化制造资源服务链构建 [J]. *计算机集成制造系统*, 2010, 16(1): 174-181.
- [5] 苏兆品, 蒋建国, 夏娜, 等. 一种基于免疫的敏捷虚拟企业伙伴选择算法 [J]. *中国机械工程*, 2008, 19(8): 925-928.
- [6] NIU S H, ONG S K, NEE A Y C. An enhanced ant colony optimiser for multi-attribute partner selection in virtual enterprises [J]. *International Journal of Production Research*, 2012, 50(8): 2286-2303.
- [7] 尚耀华, 万威武. 基于图论的虚拟企业制造伙伴选择优化算法 [J]. *系统工程学报*, 2006, 21(4): 375-380.
- [8] 孙雪冬, 李中华, 刘晓峰, 等. 支持个性化优化的业务过程建模 [J]. *计算机集成制造系统*, 2013, 19(1): 137-145.
- [9] WU Nai-qi, MAO Ning, QIAN Yan-ming. An approach to partner selection in agile manufacturing [J]. *Journal of Intelligent Manufacturing*, 1999, 10(6): 519-529.
- [10] LP W H, YUNG K L, WANG Ding-wei. A branch and bound algorithm for sub-contractor selection in agile manufacturing environment [J]. *International Journal Production Economics*, 2004, 87(2): 195-205.
- [11] 龙浩, 邱瑞华, 梁毅. 基于相对费效比的网格 workflow 调度算法 [J]. *计算机集成制造系统*, 2010, 16(3): 589-597.
- [12] 陈廷伟, 张斌, 郝宪文. 基于任务—资源分配图优化选取的网格依赖任务调度 [J]. *计算机研究与发展*, 2007, 44(10): 1741-1750.

(上接第 3556 页)

- [8] PANDEY G, ATLURI G, STEINBACH M, et al. An association analysis approach to biclustering [C]//Proc of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2009: 677-686.
- [9] ZHANG Ya, ZHA Hong-yuan, CHU C H. A time-series biclustering algorithm for revealing co-regulated genes [C]//Proc of IEEE International Conference on Information and Technology: Coding and Computing. 2005: 32-37.
- [10] WANG Guo-ren, YIN Lin-jun, ZHAO Yu-hai, et al. Efficiently mining time-delayed gene expression patterns [J]. *IEEE Trans on IEEE Systems, Man, and Cybernetics, Part B: Cybernetics*, 2010, 40(2): 400-411.
- [11] 王森, 尚学群, 谢华博, 等. 行常量差异共表达基因模式挖掘算法研究 [C]//第 29 届中国数据库学术会议论文集 (B 辑). 2012: 228-234.
- [12] CHO R, CAMPBELL M, WINZELER E, et al. A genome-wide tran-

scriptional analysis of the mitotic cell cycle [J]. *Molecular Cell*, 1998, 2(1): 65-74.

- [13] OBA S, SATO M A, TAKEMASA I, et al. A Bayesian missing value estimation method for gene expression profile data [J]. *Bioinformatics*, 2003, 19(16): 2088-2096.
- [14] ODIBAT O, REDDY C, GIROUX C. Differential biclustering for gene expression analysis [C]//Proc of the 1st ACM International Conference on Bioinformatics and Computational Biology. New York: ACM Press, 2010: 275-284.
- [15] IHMELS J, BERGMANN S, BARKAI N. Defining transcription modules using large-scale gene expression data [J]. *Bioinformatics*, 2004, 20(13): 1993-2003.
- [16] HARRIS M, CLARK J, IRELAND A, et al. The gene ontology database and informatics resource [J]. *Nucleic Acids Research*, 2004, 32(database issue): 258-261.
- [17] 杨蜜静, 尚学群, 许涛, 等. 面向时序基因表达数据的双聚类算法 [J]. *计算机应用研究*, 2013, 30(8): 2308-2314.