

一种用于 workflow 引擎的任务预测与分配算法

钱 鹰 王寸涛 韦庆杰

(重庆邮电大学图形图像与多媒体实验室 重庆 400065)

摘 要 为了检测 workflow 系统在中短期内是否会出现任务超载的情况,首先利用灰色预测算法来预测不同类型任务在预测窗口内的变化趋势,并根据预测结果建立一种负载均衡偏差可控的整数线性规划模型,以求解最优的任务分配策略。当 workflow 系统在预测窗口内出现任务超载时,该方法可计算出在保证不同工作人员之间任务负载均衡和任务分配合理性的情况下所需增加的最少工作人员数量,并且在一定程度上提高了任务处理的时效性。

关键词 workflow 任务分配 负载均衡 负载预测

中图分类号 TP317.1 文献标识码 A DOI:10.3969/j.issn.1000-386x.2014.08.019

AN ALGORITHM FOR TASK PREDICTION AND ASSIGNMENT IN WORKFLOW ENGINE

Qian Ying Wang Cuntao Wei Qingjie

(Laboratory of Graphics Image and Multimedia, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract In order to detect whether the tasks in the workflow system will overload in short-medium term, we first make use of the grey prediction algorithm to predict the variation trend of different types of tasks within the prediction window, and establish an integer linear programming model supporting the controllable load-balancing deviation according to the prediction results so as to solve the problem of optimal task allocation strategy. When the tasks become overloaded for the staff in the prediction window, this method can calculate the least increased number of staff in the workflow system. At the same time, it can ensure the load balance among different staff and the rationality of task allocation. And to a certain extent, it can also improve the timeliness of task processing.

Keywords Workflow Task assignment Load balance Load prediction

0 引 言

在工作流引擎的设计中,任务的合理分配是一个很重要的功能模块,也是任务在业务流程中能否顺利流转的一个重要影响因素,而实施 workflow 管理的一个重要目的就是把组成业务过程的任务分配给系统中合适的参与者来完成^[1]。目前,针对 workflow 引擎中任务分配这一问题,国内外的一些科研人员已进行了较深入的研究,大致可以分为两类:第一类是在流程定义文件中指定每个节点上任务对应的角色或人员,这种方法实现简单,但是不能适用于业务流程复杂的系统,也不能对任务进行动态的分配;另一类就是在任务被执行时,综合考虑工作人员的能力、负载、任务类型,以及相关的回避策略等因素由 workflow 引擎进行动态的指派,这种方式实现了任务的动态分配^[2]。而在任务的动态分配中,任务分配的均衡性和合理性是一个不容忽视的问题,目前有很多专家学者致力于研究这一问题,在文献[3]和文献[4]中,郭希娟和刘怡等人分别考虑了任务的特点和人员的能力、负载、兴趣、经验值等因素来确定候选人集合,并从候选集中选择最合适的人员。陈君和张振明等人在文献[5]中增加了对候选人技能等级和历史任务执行成功率等因素的分析,并引入模糊集来处理模糊元素。刘少伟和孔令梅等人在文献[6]中对云环境下的任务动态分配策略进行了研究。

但是,上述文献及相关研究并没有考虑系统在中短期内是否仍能够实现任务的正常分配,一般情况下,可以根据任务处理的历史数据来预测未来任务数量的变化趋势。由于在基于 Petri 网^[7-9]的 workflow 模型中一般只在变迁节点内进行任务的分配与处理,而在 workflow 模型中一般会存在多个变迁节点,而且不同的变迁节点之间存在着复杂的分支和循环等结构,因此,在分布式系统中常用的服务器负载预测算法不能直接应用于 workflow 模型中的任务预测。本文结合了基于 Petri 网 workflow 模型的特点,利用灰色模型来预测任务数量的变化趋势,并建立整数线性规划模型来检测在中短期内 workflow 引擎是否能够保证任务的均衡分配,当 workflow 系统在预测窗口内出现任务超载时,本方法可以计算出所需增加的最少工作人员数量,并在 workflow 系统内产生报警信息。

1 工作流任务分配模型

工作流引擎是 workflow 管理系统的核心部件,而 workflow 引擎的一个重要工作就是通过其内部的任务管理器^[10]对系统中的

收稿日期:2012-12-21。国家自然科学基金项目(61171060);教育部留学回国人才科研启动基金项目(教外司留(2009)1509号)。钱鹰,教授,主研领域:医学图像处理,计算机仿真系统。王寸涛,硕士生。韦庆杰,高工。

任务进行分配,以保证流程实例能够顺利的流转。在基于 Petri 网的工作流模型中,库所用于描述系统的状态信息,变迁用于描述使系统状态发生改变的事件^[11],任务的分配和处理一般都发生在变迁节点中,图1为任务管理器与变迁节点关系示意图。

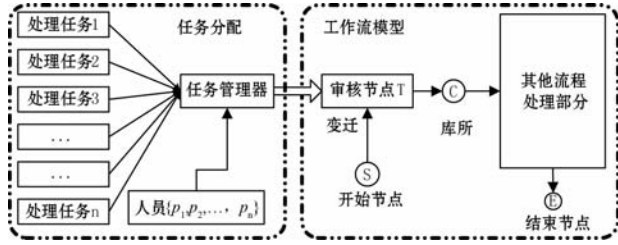


图1 任务管理器与变迁节点关系示意图

在图1中,假设一天内任务请求总数为 N ,变迁节点 T 处具有相关权限的任务处理人员集合为 $P = \{p_1, p_2, \dots, p_n\}$,每个人在处理不同类型的任务时所需要的时间是不相同的,任务类型集合可表示为 $W = \{w_1, w_2, \dots, w_k\}$ 。为简化模型,假设不同类型任务的发生时间是随机的,根据系统中任务处理的历史数据,可计算出人员 p_i 处理 w_j 类任务的平均处理时间为 $t_{i,j}$ 。

本文首先需要根据历史数据来预测未来不同类型任务的数量变化情况,并根据预测结果建立了一种整数线性规划模型来分析系统在预测窗口内是否能实现多类型任务的均衡分配。

假设第 w_i 类任务在第 q 天的数量为 $h_{i,q}$,那么,第 q 天需要处理的总任务量为:

$$N = \sum_{i=1}^k h_{i,q} \quad (1)$$

其中, $i=1,2,\dots,k$; q 为大于1的整数,当 $q=1$ 时表示第一天。

根据历史数据可以计算第 i 类任务中平均每个任务的处理时间,记为 \bar{t}_i 。那么完成未来第 k 天的任务需要的总工时为:

$$t_{all} = \sum_{i=1}^w h_{i,k} \times \bar{t}_i \quad (2)$$

任务管理器在分配第 k 天的任务时,既要保证不同人员的任务负载相对均衡,还要保证员工能够处理完每种类型的任务,在此基础上以完成全部任务所需时间总和最短为目标来计算最优的任务分配方案。若以 $d_{i,j}$ 表示人员 p_i 所分配的任务类型 w_j 的数量,以 O 表示系统所设置的人员每日工作时间长度,本文所建立的整数线性规划模型具体如下:

$$\begin{cases} \sum_{i=1}^n d_{i,j} \geq h_{i,q} & j = 1, 2, \dots, k \\ \sum_{j=1}^k d_{i,j} \times t_{i,j} \leq O & i = 1, 2, \dots, n \\ \max \left| \sum_{j=1}^k d_{i,j} \times t_{i,j} - \bar{t} \right| / \bar{t} \leq \beta & i = 1, 2, \dots, n \end{cases} \quad (3)$$

其中, \bar{t} 表示所有员工的日平均工作时间, β 表示不同员工之间的任务处理量相对偏差峰值,其取值区间为 $[0,1]$,具体的取值可由用户根据实际情况进行设定, β 越大表示任务分配得越不合理。

相应的目标函数为:

$$\min z = \sum_{i=1}^n \sum_{j=1}^k d_{i,j} \times t_{i,j} \quad (4)$$

若式(3)有解,则可得到每个人需要处理的每种类型任务的数量;否则,表示 β 设置的值过小,或者任务量已经超过了工作人员的最大负载能力。若 $\beta=1$ 时,式(3)仍然无解,此时需要增加任务处理人员的数量,以提高系统的整体任务吞吐量。

但是,为了降低企业的人力资源成本,需要在保证完成全部任务的前提下,使得增加的人员数量最少,假设新添的人员没有相关的任务处理经验,在处理不同类型的任务时,平均处理时间是相同的,均为 t_{new} 。

那么,式(3)、式(4)变为式(5)、式(6),如下所示:

$$\begin{cases} \sum_{i=1}^{n+num} d_{i,j} \geq h_{i,k} & j = 1, 2, \dots, k \\ \sum_{j=1}^k d_{i,j} \times t_{i,j} \leq O & i = 1, 2, \dots, n+num \\ \max \left| \sum_{j=1}^k d_{i,j} \times t_{i,j} - \bar{t} \right| / \bar{t} \leq \beta & i = 1, 2, \dots, n+num \\ t_{i,j} = t_{new} & i = n+1, n+2, \dots, n+num \end{cases} \quad (5)$$

$$\min z = \sum_{i=1}^n \sum_{j=1}^k d_{i,j} \times t_{i,j} + num \times t_{new} \quad (6)$$

在式(5)有解的情况下,计算此时的 num 值,即为需要新增的最少员工数量。

2 任务数量的预测

目前,用于数据预测与分析的算法很多,比较常用的算法包括神经网络算法、多元线性回归、灰色预测算法等,由于在数据较少时神经网络存在网络训练外延的问题,而多元线性回归等预测方法的预测精度不太够高^[12],因此,本文选取灰色 $GM(1,1)$ ^[13]模型来预测任务数量的变化趋势。

2.1 数据预处理

首先,获取2005年至2012年期间某网购工作流系统中3类任务的日处理数据,每类任务包括2922组数据,其中任务1和任务2的数量呈逐渐增长的趋势,任务3的数量呈逐渐下降的趋势。其工作流模型如图2所示,图3为2012年期间该模型中节点 T 处不同类型任务的数量变化情况。

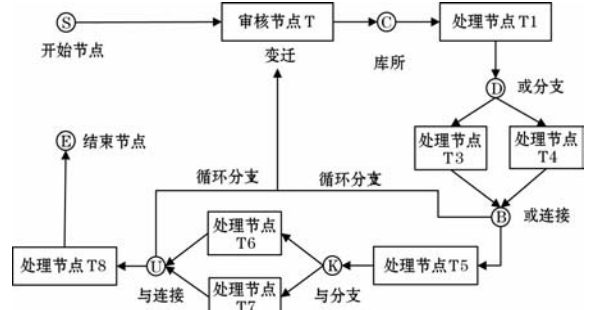


图2 具有循环和分支结构的业务流程示意图

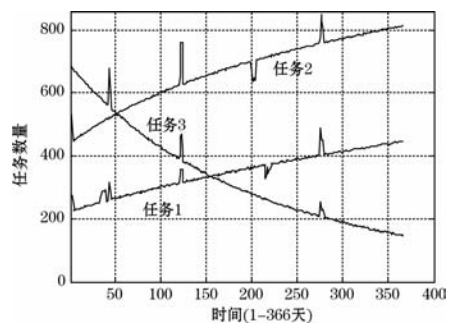


图3 2012年内系统所处理的不同类型任务的数量

从图3可以看出,不同类型的任务数量在一些特殊的时间点上存在数据突增和突减的现象,这些特殊数据一般是商家在

一些节假日促销所造成的,为了保证数据预测的准确性,需要根据历年节假日、展销会等特殊时间点的任务数量来计算预测窗口内的特殊数据,然后对近期的任务处理数据序列进行特殊数据过滤后再预测其他日常数据,二者的整合即为最终的预测结果。其中,特殊数据过滤算法对应的伪代码如下所示:

```

输入: 原始任务量数据序列
输出: 过滤掉特殊数据后的任务量数据序列
Begin: 读取原始任务量数据序列 data,并设定过滤窗口大小 windows 和
过滤阈值 Threshold;
将 data 序列的第一个值追加到 data 的起始位置,重复 windows 次;
for(i 从 windows + 1 开始,到 length(data),步长为 1)
{
    计算 data 序列中第 i-windows 到第 i-1 个值的中值,保存为 datam;
    if(abs(data(i) - datam)/datam >= Threshold/datam)
    {
        if(data 序列是上升的趋势)
            把 data 序列中第 i-windows 个到 i-1 个值中的最大值赋给 data(i);
        else
            把 data 序列中第 i-windows 个到 i-1 个值中的最小值赋给 data(i);
    }
}
return data 序列中的第 windows + 1 个到最后一个值;
End
    
```

图 4 为过滤掉特殊数据后的 2012 年期间任务数量历史数据序列。

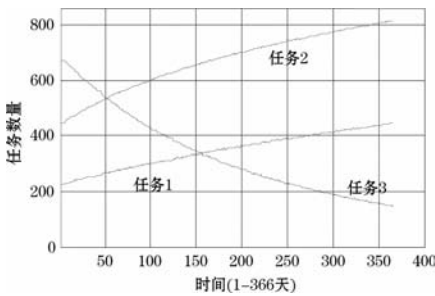


图 4 过滤掉特殊数据后的任务数量序列

从图 4 可以看出,过滤掉特殊后的数据序列比原始数据序列更加平滑。

2.2 任务量预测

在灰色预测模型中,假设模型的输入数据序列为 $X^0 = (X^0(1), X^0(2), \dots, X^0(n))$,那么一次累加生成和一次均值生成分别为:

$$X^1 = (X^1(1), X^1(2), \dots, X^1(n)) \quad (7)$$

$$Z^1 = (Z^1(1), Z^1(2), \dots, Z^1(n)) \quad (8)$$

其中:

$$X^1(k) = \sum_{i=1}^k X^0(i)$$

$$Z^1(k) = [X^1(k) + X^1(k-1)]/2 \quad k = 2, 3, \dots, n$$

那么,预测模型可以简写为^[13]:

$$X^0(k) + a \times Z^1(k) = b \quad (9)$$

其中, a, b 为预测模型的参数。

上述方程的解可以表示为:

$$X_1^{(1)}(k+1) = [X^0(1) - \frac{b}{a}]e^{-ak} + \frac{b}{a} \quad k = 0, 1, 2, \dots, n \quad (10)$$

然后,通过累减操作可以还原为相应变量的原序列预测值:

$$\hat{x}_1^{(0)}(k+1) = \hat{x}_1^{(1)}(k+1) - \hat{x}_1^{(1)}(k) \quad (11)$$

将系统内不同类型的任务历史数据数列作为 X^0 ,即可得到具体的预测模型。在利用 2012 年第 60 ~ 366 天的任务量数据预测 2013 年第 1 ~ 50 天的任务量数据时,可得到 2012 年第 60 ~ 366 天的任务量实际值与预测值之间的对比情况,如图 5 所示。

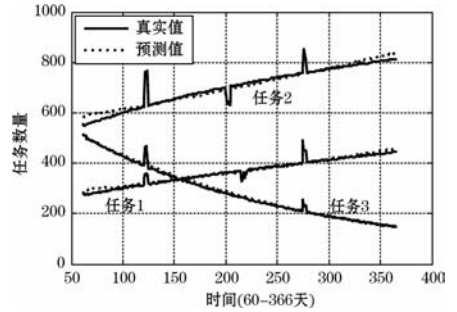


图 5 2012 年不同类型任务量的真实值与预测值对比

2.3 预测模型精度检验

在得到预测结果后,本文通过分析预测结果与真实值的相对误差,以及进行后差检验的方式来分析预测模型的准确度,其中相对误差为:

$$\Delta_k = \frac{|x_1^{(0)}(k) - \hat{x}_1^{(0)}(k)|}{x_1^{(0)}(k)} \times 100\% \quad (12)$$

式中, $\hat{x}_1^{(0)}$ 和 $x_1^{(0)}(k)$ 分别为序列的预测值和实际值。

后差检验一般通过参数 P 和参数 C 来表示^[14],其计算式分别如式(13)和式(14)所示:

$$P = p \{ |\varepsilon(k) - \bar{\varepsilon}| < 0.6745S_1 \} \quad (13)$$

$$C = \frac{S_2}{S_1} \quad (14)$$

其中 $S_1^2 = \frac{1}{n} \sum_{k=1}^n (x^{(0)}(k) - \bar{x}^{(0)})^2$, $S_2^2 = \frac{1}{n} \sum_{k=1}^n (\varepsilon(k) - \bar{\varepsilon})^2$ 分别为原始数据序列 $x^{(0)}(k)$ 及残差数列 $\varepsilon(k)$ 的方差; $\bar{x}^{(0)} = \frac{1}{n} \sum_{k=1}^n x^{(0)}(k)$, $\bar{\varepsilon} = \frac{1}{n} \sum_{k=1}^n \varepsilon(k)$ 分别为原始数据序列 $x^{(0)}(k)$ 及残差数列 $\varepsilon(k)$ 的均值。而且,参数 P 越大越好,参数 C 越小越好。

其中,2012 年第 60 ~ 366 天的任务量实际值与预测值之间的误差分析如表 1 所示。

表 1 不同输入数据序列下的预测结果及精度分析表

任务类型	变迁节点	预测窗口	P	C	最大相对误差	平均相对误差
任务 1	T	50 天	1	0.09	0.072	0.015
任务 2	T	50 天	1	0.107	0.074	0.016
任务 3	T	50 天	1	0.282	0.042	0.016

一般情况下,当 $P > 0.95$ 和 $C < 0.35$ 时即可判定模型的预测精度为好^[15],从表 1 可以看出,本模型的预测精度良好。

3 模型的应用及分析

为了进一步分析系统预测窗口(2013 年第 1 ~ 50 天)内工作人员的负载是否存在超载的情况,可以选择预测窗口内任务总量最大的时间点进行研究,即满足 $t_{all} = \max(\sum_{i=1}^w h_{i,k} \times \bar{t}_i)$ 的

时间点。

根据系统中的任务处理记录,计算三种不同类型任务的平均处理时间,即 $t_1 = 48$ s, $t_2 = 44$ s, $t_3 = 57$ s。那么,图2中节点T在预测窗口内处理不同类型任务所需时间之和的变化趋势如图6所示。

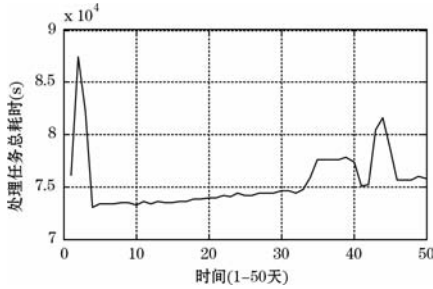


图6 节点T处理任务所需时间之和变化趋势

从图6可以看出,节点T处不同类型任务总耗时预测值的变化趋势也存在跳变,并在第2天时达到最大,此时为24.27小时(87382 s),其中任务一、任务二和任务三数量的预测值分别为560、1024、146。

在图2中,假设节点T处的各类任务必须在当天处理完成,共有三个工作人员,每个工作人员每天的限定工作时长为8小时,其处理不同类型的任务时所需时间如表2所示。

表2 图5中假设节点T处每个人处理不同类型任务平均所需时间表 单位:s

任务类型\人员	p_1	p_2	p_3
w_1	56	39	46
w_2	53	45	71
w_3	40	49	55

根据式(3)、式(4),可以得到式(15):

$$\left\{ \begin{array}{l} \sum_{i=1}^3 d_{i,1} \geq 560 \quad \sum_{i=1}^3 d_{i,2} \geq 1024 \quad \sum_{i=1}^3 d_{i,3} \geq 146 \\ \sum_{j=1}^3 d_{i,j} \times t_{i,j} \leq 8 \times 60 \times 60 \quad i = 1, 2, 3 \\ \max \left| \sum_{j=1}^3 d_{i,j} \times t_{i,j} - \bar{t} \right| / \bar{t} \leq \beta \quad i = 1, 2, 3 \\ \text{目标: } \min z = \sum_{i=1}^n \sum_{j=1}^{k=3} d_{i,j} \times t_{i,j} \end{array} \right. \quad (15)$$

令 $\beta = 0.15$,经过计算可知上式存在满足条件的最优解,具体分配情况如表3所示,此时处理全部任务的总耗时为22.43小时(80749 s),此时即保证了不同工作人员之间的任务负载均衡和任务分配的合理性,也提高了任务处理的时效性。

表3 $\beta = 0.15$ 时节点T处每个人的任务分配数量情况

任务类型\人员	p_1	p_2	p_3
w_1	0	53	507
w_2	430	594	0
w_3	146	0	0

当系统中设定 $\beta = 0.1$ 时,经计算式(3)无解,若设定 $t_{new} = 60$ s,经过计算可以得知需要在节点T处需至少新增1位工作人员才能保证在预测窗口内不出现任务超载的情况。

4 结语

为了检测在未来中短期内, workflow 管理系统是否能实现对

任务的均衡分配,本文首先利用灰色GM(1,1)模型来预测未来中短期内任务数量的变化趋势,并结合整数线性规划的方法来建立一种负载均衡偏差可控的任务分配模型,以分析在预测窗口内系统的任务是否能得到合理的分配。

参考文献

- [1] 肖正进,何钦铭,陈奇. 模糊环境中工作流任务分配的多级模型[J]. 计算机研究与发展,2007,44(2):302-309.
- [2] 杨天怡,董红林,黄勤,等. 应用角色和任务访问控制的工作流动态授权模型[J]. 计算机应用研究,2010,27(4):1511-1513.
- [3] 郭希娟,李墨华. 基于多准则的动态任务分配算法[J]. 计算机应用,2008,28(10):2507-2509.
- [4] 刘怡,张戡. 基于负载均衡和经验值的工作流任务分配策略[J]. 计算机工程,2009,35(21):57-59.
- [5] 陈君,张振明,田锡天,等. 基于模糊集与TOPSIS的工作流任务分配方法研究[J]. 计算机应用研究,2011,28(8):2883-2885.
- [6] 刘少伟,孔令梅,任开军,等. 云环境下优化科学工作流执行性能的两阶段数据放置与任务调度策略[J]. 计算机学报,2011,34(11):2021-2130.
- [7] V C Gerogiannis. Comparative study and categorization of high level Petri nets [J]. Journal of System and Software,1998,43(3):133-160.
- [8] Claude Girault, Rudiger Valk. Petri Nets for Systems Engineering-A Guide to Modeling, Verification, and Application [M]. Springer-Verlag Berlin Heidelberg,2003:8-33.
- [9] van der Aalst W M P, ter Hofstede A H M, et al. Workflow Patterns [J]. Distributed and Parallel Databases,2003,14(1):5-51.
- [10] Workflow Management Coaliton. The Workflow Reference Model [R]. WFMCTC00-1003. Hampshire; Workflow Management Coalition,1995.
- [11] 黄国言,杨永涛,郝晓冰. 基于Petri网的计算机支持的协同设计仿真技术研究[J]. 燕山大学学报,2011,35(6):549-555.
- [12] 李国成,吴涛,徐沈. 灰色人工神经网络人口总量预测模型及应用[J]. 计算机工程与应用,2009,45(16):215-218.
- [13] 刘思峰,党耀国,方志耕,等. 灰色系统理论及其应用[M]. 北京:科学出版社,2004:163-183.
- [14] 李红霞,赵新华,迟海燕,等. 基于改进BP神经网络模型的地面沉降预测及分析[J]. 天津大学学报,2009,42(1):60-64.
- [15] 邓聚龙. 灰色控制系统[M]. 2版. 武汉:华中理工大学出版社,1997.

(上接第27页)

- [5] ScottMcFarling. Combining Branch Predictors[R]. WRL Technical Note TN-86, June 1993.
- [6] Chih-Chieh Lee, I Cheng K Chen, Trevor N Mudge. The Bi-Mode Branch Predictor [C]//Proceedings of the 30 Annual International Symposium on Microarchitecture 1997.
- [7] Eden A N, Mudge T N. The YAGS Branch Prediction Scheme [C]// Proceeding of the 31st Annual ACM/IEEE International Symposium on Microarchitecture, November 1998.
- [8] Shen Zheng, He Hu, Yang Xu, et al. Architecture Design of a Variable Length Instruction Set VLIW DSP [J]. Tsinghua Science and Technology, 2009, 14(5):561-569.
- [9] 张延军,何虎,周志雄,等. RFCC-VLIW:一种适用于超长指令字处理器的寄存器堆结构[J]. 清华大学学报:自然科学版,2008(10):1651-1654.