(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 106897199 A
(43)申请公布日 2017.06.27

(21)申请号 201710027302.1

(22)申请日 2017.01.12

(71)申请人 河南理工大学
地址 454003 河南省焦作市高新区世纪大道2001号河南理工大学计算机学院

(72)发明人 张霄宏 赵文涛 智慧来 吴岩 曾艳阳 苗煜飞

(51)Int.Cl.
G06F 11/34(2006.01)

权利要求书2页 说明书6页

(54)发明名称

一种基于大数据处理框架共有特征的批作业执行时间预测方法

(57)摘要

本发明提出一种基于大数据处理框架共有特征的批作业执行时间预测方法，该方法可预测一批大数据作业的执行时间。根据预测结果，制定合理的调度策略，在生产性高优先级作业到来前，充分利用集群的计算资源在有限时间内执行尽可能多的作业，进一步提高集群的资源利用率和吞吐量。

CN 106897199 A

1.一种基于大数据处理框架共有特征的批作业执行时间预测方法，其特征在于：该方法包括以下步骤：

一、为复杂作业创建作业模型：在作业模型中指明被派生的作业类型及执行阶段，每个作业执行阶段的任务数量、输入数据信息；

二、分析历史数据，获取各类作业在多种准确度下各种类任务的执行时间：利用概率统计分析各类作业中每种任务在不同类型节点上的执行时间，获取每种任务在不同概率下的统计执行时间；

三、预测时间片的可用时间：如果时间片处于空闲状态，按1)表示可用时间；如果时间片处于忙状态，按照2)和3)预测可用时间；

1) $ts_{n,i}=-1$，$S_{n,i}$表示节点n上的第i个时间片；$ts_{n,i}$表示$S_{n,i}$的可用时间，即从$ts_{n,i}$时刻起，时间片可用；$ts_{n,i}$值为-1表示$S_{n,i}$当前处于空闲状态；

2)从上述步骤二的结果中查找与$S_{n,i}$正在执行的任务匹配的数据，计算$S_{n,i}$的可用时间$ts_{n,i}=ts_{f\to b}+T_{<x,y,ntype>}\times(I/I_{<x,y>})$，其中$ts_{f\to b}$表示$S_{n,i}$开始执行当前任务的时间，即当前时间片由空闲状态转变成忙状态的时间；$T_{<x,y,ntype>}$和$I_{<x,y>}$是步骤二的结果中与$S_{n,i}$正在执行的任务匹配的数据，$T_{<x,y,ntype>}$表示x类型的作业中y类型的任务在ntype类型节点上的执行时间，$I_{<x,y>}$表示与$T_{<x,y,ntype>}$对应的输入数据规模；I表示正占用$S_{n,i}$执行的任务的输入数据规模；

3)如果没有匹配的历史数据，则根据任务的执行进度预测任务的完成时间，即$ts_{n,i}=ts_{f\to b}+T_{<x,y,r>}/P_{<x,y,r>}$，其中$T_{<x,y,r>}$表示任务在$ts_{f\to b}$时刻开始后已执行的时间；$P_{<x,y,r>}$表示在$T_{<x,y,r>}$时间内任务的执行进度；

四、将每个时间片按照可用时间由小到大组织成队列：对于特定种类的任务只能在特定种类的时间片上执行的情况，需要首先对时间片分类，然后对每种类型的时间片分别建队列，每个队列中时间片按可用时间由小到大的顺序组织；

五、将批作业开始执行时间和结束执行时间分别记为T_{start}，T_{end}；T_{start}，T_{end}分别取最大长整型数和最小长整型数；

六、从等待队列中取出下一个待调度的作业，记为j，若判定该作业是简单作业，执行步骤七；若判定是复杂作业，执行步骤八；

七、预测简单作业的执行时间：

1)根据作业信息划分执行阶段，根据执行先后次序，由小到大对每个阶段进行编号；

2)结合历史信息和作业信息，计算在每个阶段派生的任务数量及任务的输入数据规模；

3)将j开始执行时间和结束执行时间分别记为$T_{j,start}$，$T_{j,end}$；$T_{j,start}$，$T_{j,end}$分别取最大长整型数和最小长整型数；

4)按照阶段编号由低到高的顺序，选择一个阶段；

5)从选定阶段中选择一个任务，对于选中任务，执行下列预测操作：

a)从选中任务对应的时间片队列中摘取队首时间片，记为$S_{n,i}$，即该时间片是第n个节点上的第i个；

b)如果$ts_{n,i}$的值为-1，则将系统的当前时间赋值给$ts_{n,i}$；

c)如果时间片满足$T_{j,start} > ts_{n,i}$,则$T_{j,start} = ts_{n,i}$;

d)从历史信息中找到与该任务匹配的执行时间$T_{<x,y,ntype>}$,即x类型作业中y类型的任务在ntype类型节点上的执行时间;

e)重新计算时间片的可用时间,即$ts_{n,i} = ts_{n,i} + T_{<x,y,ntype>} \times (I / I_{<x,y>})$,此处I为当前任务的输入数据规模;

f)如果$S_{n,i}$满足$ts_{n,i} > T_{j,end}$,则$T_{j,end} = ts_{n,i}$;

g)按照可用时间由小到大的顺序将该时间片重新插入对应的时间片队列;

6)重复执行步骤5),直到预测完选定阶段中所有任务的执行时间;

7)重复执行步骤4)-5),直到所有阶段中所有任务的执行时间都预测结束;

8)将$T_{j,start}$和$T_{j,end}$作为作业j的开始时间和结束时间返回;

八、预测复杂作业的执行时间:

1)将作业看作简单作业,根据步骤七预测作业的执行时间;

2)从作业模型中找到与该作业对应的模型,根据模型创建由该作业触发的所有作业并插入等待队列末尾;

九、如果$T_{start} > T_{j,start}$,则$T_{start} = T_{j,start}$;如果$T_{j,end} > T_{end}$,则$T_{end} = T_{j,end}$;

十、重复步骤六至步骤九,直到等待队列中最后一个作业的执行时间预测结束为止。记$T_{end}-T_{star}$为执行完等待队列中现有作业所需的时间。

一种基于大数据处理框架共有特征的批作业执行时间预测方法

技术领域

[0001] 本发明涉及一种作业执行时间预测方法，具体地，涉及一种基于大数据处理框架共有特征的批作业执行时间预测方法，属于大数据技术领域。

背景技术

[0002] 随着计算机技术和互联网技术的迅速发展，数据呈爆炸式疯狂增长。为了应对海量数据处理压力，先后出现了MapReduce、Dryad、Spark等多种大数据处理框架。这些框架的基本原理都是首先将海量数据划分成小块，然后分布到不同的节点，并行处理。在实际应用中，首先需要将框架部署到大规模集群上，依托集群提供的计算和存储资源进行大数据处理。有学者的研究表明，集群中的负载可以分成两类：非周期性作业和周期性作业。非周期性作业通常是一些实验负载，规模大小不一，运行数秒数分钟都有可能。而周期性作业往往是规模较大的生产负载，与公司的核心业务密切相关，一旦提交，必须立即执行。但是，在资源竞争激烈的大数据环境，保证及时执行高优先级作业极具挑战。

[0003] 通常，为了保证及时执行生产作业，由人工估算等待队列中已有作业的执行时间，并根据估算结果在生产作业到来之前拒绝接收新作业，以便给已有作业预留足够多的执行时间。然而，由于人工估算误差较大，集群往往提前处理处理完这些作业并进入空转状态。由于生产作业与核心业务密切相关，现有的策略必须保证生产作业到来时等待队列为空，如此生产作业一旦提交就可立即执行。为做到这一点，现有策略宁可让集群进入空转状态，也不愿尽可能多执行一些作业，从而降低了资源利用率和系统吞吐量。如果能获得等待队列中所有作业较准确的执行时间，集群便可合理的安排作业调度，在生产作业到达前处理尽可能多的作业，从而提高资源的利用率和系统的吞吐量。

[0004] 通过分析现有大数据处理过程的特征，发现作业的执行过程都被划分成多个不同的阶段，每个阶段包含若干任务并执行特定的处理操作，且只有在前一阶段所有任务执行结束后才能进入后一阶段。阶段之间存在一定的数据依赖关系，即前一阶段的输出数据是后一阶段的输入数据。只要作业类型相同，划分出的阶段和阶段中任务执行的操作也相同，差别只在每个任务要处理的数据规模。因此，在数据规模一定的前提下，如果能获取每个阶段中任务的执行时间、集群可用的资源信息等因素，便可预测单个作业的完成时间，进而预测等待队列中所有作业的执行时间。

发明内容

[0005] 为了解决现有技术中存在的种种问题，本发明提出了一种基于大数据处理框架共有特征的批作业执行时间预测方法。该方法包括以下步骤：

[0006] 一、为复杂作业创建作业模型：在作业模型中指明被派生的作业类型及执行阶段，每个作业执行阶段的任务数量、输入数据信息；

[0007] 二、分析历史数据，获取各类作业在多种准确度下各种类任务的执行时间：利用概

率统计分析各类作业中每种任务在不同类型节点上的执行时间，获取每种任务在不同概率下的统计执行时间；

[0008] 三、预测时间片的可用时间：如果时间片处于空闲状态，按1)表示可用时间；如果时间片处于忙状态，按照2)和3)预测可用时间；

[0009] 1) $ts_{n,i}=-1$，$S_{n,i}$表示节点n上的第i个时间片；$ts_{n,i}$表示$S_{n,i}$的可用时间，即从$ts_{n,i}$时刻起，时间片可用；$ts_{n,i}$值为-1表示$S_{n,i}$当前处于空闲状态；

[0010] 2)从上述步骤二的结果中查找与$S_{n,i}$正在执行的任务匹配的数据，计算$S_{n,i}$的可用时间$ts_{n,i}=ts_{f_{\to}b}+T_{<x,y,ntype>}\times(I/I_{<x,y>})$，其中$ts_{f_{\to}b}$表示$S_{n,i}$开始执行当前任务的时间，即当前时间片由空闲状态转变成忙状态的时间；$T_{<x,y,ntype>}$和$I_{<x,y>}$是步骤二的结果中与$S_{n,i}$正在执行的任务匹配的数据，$T_{<x,y,ntype>}$表示x类型的作业中y类型的任务在ntype类型节点上的执行时间，$I_{<x,y>}$表示与$T_{<x,y,ntype>}$对应的输入数据规模；I表示正占用$S_{n,i}$执行的任务的输入数据规模；

[0011] 3)如果没有匹配的历史数据，则根据任务的执行进度预测任务的完成时间，即$ts_{n,i}=ts_{f_{\to}b}+T_{<x,y,r>}/P_{<x,y,r>}$，其中$T_{<x,y,r>}$表示任务在$ts_{f_{\to}b}$时刻开始后已执行的时间；$P_{<x,y,r>}$表示在$T_{<x,y,r>}$时间内任务的执行进度；

[0012] 四、将每个时间片按照可用时间由小到大组织成队列：对于特定种类的任务只能在特定种类的时间片上执行的情况，需要首先对时间片分类，然后对每种类型的时间片分别建队列，每个队列中时间片按可用时间由小到大的顺序组织；

[0013] 五、将批作业开始执行时间和结束执行时间分别记为T_{start}，T_{end}；T_{start}，T_{end}分别取最大长整型数和最小长整型数；

[0014] 六、从等待队列中取出下一个待调度的作业，记为j，若判定该作业是简单作业，执行步骤七；若判定是复杂作业，执行步骤八；

[0015] 七、预测简单作业的执行时间：

[0016] 1)根据作业信息划分执行阶段，根据执行先后次序，由小到大对每个阶段进行编号；

[0017] 2)结合历史信息和作业信息，计算在每个阶段派生的任务数量及任务的输入数据规模；

[0018] 3)将j开始执行时间和结束执行时间分别记为$T_{j,start}$，$T_{j,end}$；$T_{j,start}$，$T_{j,end}$分别取最大长整型数和最小长整型数；

[0019] 4)按照阶段编号由低到高的顺序，选择一个阶段；

[0020] 5)从选定阶段中选择一个任务，对于选中任务，执行下列预测操作：

[0021] a)从选中任务对应的时间片队列中摘取队首时间片，记为$S_{n,i}$，即该时间片是第n个节点上的第i个；

[0022] b)如果$ts_{n,i}$的值为-1，则将系统的当前时间赋值给$ts_{n,i}$；

[0023] c)如果时间片满足$T_{j,start}>ts_{n,i}$，则$T_{j,start}=ts_{n,i}$；

[0024] d)从历史信息中找到与该任务匹配的执行时间$T_{<x,y,ntype>}$，即x类型作业中y类型的任务在ntype类型节点上的执行时间；

[0025] e)重新计算时间片的可用时间，即$ts_{n,i}=ts_{n,i}+T_{<x,y,ntype>}\times(I/I_{<x,y>})$，此处I为当前任务的输入数据规模；

[0026] f)如果$S_{n,i}$满足$ts_{n,i}>T_{j,end}$，则$T_{j,end}=ts_{n,i}$；

[0027] g)按照可用时间由小到大的顺序将该时间片重新插入对应的时间片队列；

[0028] 6)重复执行步骤5)，直到预测完选定阶段中所有任务的执行时间；

[0029] 7)重复执行步骤4)-5)，直到所有阶段中所有任务的执行时间都预测结束；

[0030] 8)将$T_{j,start}$和$T_{j,end}$作为作业j的开始时间和结束时间返回；

[0031] 八、预测复杂作业的执行时间：

[0032] 1)将作业看作简单作业，根据步骤七预测作业的执行时间；

[0033] 2)从作业模型中找到与该作业对应的模型，根据模型创建由该作业派生的所有作业并插入等待队列末尾；

[0034] 九、如果$T_{start}>T_{j,start}$，则$T_{start}=T_{j,start}$；如果$T_{j,end}>T_{end}$，则$T_{end}=T_{j,end}$；

[0035] 十、重复步骤六至步骤九，直到等待队列中最后一个作业的执行时间预测结束为止。记$T_{end}-T_{star}$为执行完等待队列中现有作业所需的时间。

[0036] 本发明可预测一批大数据作业的执行时间。根据预测结果，制定合理的调度策略，在高优先级作业到来前，充分利用集群的计算资源在有限时间内执行尽可能多的作业，进一步提高集群的资源利用率和吞吐量。

具体实施方式

[0037] 一、为复杂作业创建作业模型。作业模型指明由该作业派生的作业及其执行方式。模型中要明确指明被派生的作业的执行阶段，每个阶段的任务数量、输入数据信息。

[0038] 二、分析历史数据，获取各类作业中各种类任务在多种准确度下的执行时间。

[0039] 在分析过程中，利用概率统计的方法分析各类作业中各种任务在不同类型节点上的执行时间，获取其在不同概率下的统计执行时间。本发明用概率表示准确度。如果要求预测准确度为80％，则选择概率为0.8的分析结果作为预测参数。

[0040] 步骤三：预测时间片的可用时间。如果时间片处于空闲状态，按1)所示方法表示其可用时间。如果时间片处于忙状态，则按照2)和3)预测其可用时间。

[0041] 1)$ts_{n,i}=-1$，$S_{n,i}$表示节点n上的第i个时间片；$ts_{n,i}$表示$S_{n,i}$的可用时间，即从$ts_{n,i}$时刻起，时间片是可用的。$ts_{n,i}$值为-1表示$S_{n,i}$当前处于空闲状态。

[0042] 2)从上述步骤二的结果中查找与$S_{n,i}$正在执行的任务匹配的数据，计算$S_{n,i}$的可用时间$ts_{n,i}=ts_{f\to b}+T_{<x,y,ntype>}\times(I/I_{<x,y>})$，其中$ts_{f\to b}$表示$S_{n,i}$开始执行当前任务的时间，即当前时间片由空闲状态转变成忙状态的时间；$T_{<x,y,ntype>}$和$I_{<x,y>}$是步骤二的结果中与$S_{n,i}$正在执行的任务匹配的数据，$T_{<x,y,ntype>}$表示x类型的作业中y类型的任务在ntype类型节点上的执行时间，$I_{<x,y>}$表示与$T_{<x,y,ntype>}$对应的输入数据规模；I表示正占用$S_{n,i}$执行的任务的输入数据规模。

[0043] 3)如果没有匹配的历史数据，则根据任务的执行进度预测任务的完成时间(时间片空闲时间)$ts_{n,i}=ts_{f\to b}+T_{<x,y,r>}/P_{<x,y,r>}$，其中，$T_{<x,y,r>}$表示任务在$ts_{f\to b}$时刻开始后已执行的时

间;$P_{<x,y,r>}$表示在$T_{<x,y,r>}$时间内任务的执行进度。

[0044] 步骤四:将每个时间片按照可用时间由小到大的顺序组织成队列。

[0045] 具体地,对于特定种类的任务只能在特定种类的时间片上执行的情况,需要首先对时间片进行分类,然后对每种类型的时间片分别建队列,每个队列的时间片按可用时间由小到大的顺序组织。以基于MapReduce的大数据处理为例,时间片可以分成用于执行map任务的时间片和用于执行reduce任务的时间片两种,分别排成两个队列。

[0046] 步骤五:将批作业开始执行时间和结束执行时间分别记为T_{start},T_{end};T_{start},T_{end}分别取最大长整型数和最小长整型数。

[0047] 步骤六:从等待队列中取出下一个待调度的作业,记为j。若判定该作业是简单作业,执行步骤七;若判定是复杂作业,执行步骤八。

[0048] 具体地,根据历史信息判断当前作业是简单作业还是复杂作业。作业分成简单作业和复杂作业。简单作业在一次执行结束后,产生最终输出结果。复杂作业在一次执行结束后,产生中间结果,并派生出新的作业。新作业以中间结果为输入,继续执行。派生的新作业可能是简单作业,也可能是复杂作业。无论是哪种作业,都通过模拟真实集群中作业的调度执行过程来预测其执行时间。在具体预测时,简单作业的预测如步骤七,复杂作业的预测如步骤八。

[0049] 步骤七:预测简单作业的执行时间。

[0050] 1)根据j的作业信息划分其执行阶段,根据执行先后次序,由小到大对每个阶段进行编号。最先执行的阶段,编号最小。最后执行的阶段,编号最大。

[0051] 2)结合历史信息和作业信息,计算在每个阶段派生的任务数量及任务的输入数据规模。

[0052] 3)将j的开始执行时间和结束执行时间分别记为$T_{j,start}$,$T_{j,end}$;$T_{j,start}$,$T_{j,end}$分别取最大长整型数和最小长整型数。

[0053] 4)按照阶段编号由低到高的顺序,选择一个阶段。

[0054] 5)从选定阶段中选择一个任务,对于选中任务,执行下列预测操作:

[0055] a)从该任务对应的时间片队列中摘取队首时间片,记其编号为$S_{n,i}$,即该时间片是第n个节点上的第i个Slot。

[0056] b)如果$ts_{n,i}$的值为-1,则将系统的当前时间赋值给$ts_{n,i}$。

[0057] c)如果时间片满足$T_{j,start} > ts_{n,i}$,则$T_{j,start} = ts_{n,i}$。

[0058] d)根据作业信息、时间片的节点信息和所能接受的预测准确度,从历史信息中找到与该任务匹配的执行时间$T_{<x,y,ntype>}$以及与其对应的输入数据规模$I_{<x,y>}$。

[0059] e)重新计算时间片的可用时间,即$ts_{n,i} = ts_{n,i} + T_{<x,y,ntype>} \times (I / I_{<x,y>})$。

[0060] f)如果$S_{n,i}$满足条件$ts_{n,i} > T_{j,end}$,$T_{j,end} = ts_{n,i}$。

[0061] g)按照可用时间由小到大的顺序将该时间片重新插入对应的时间片队列。

[0062] 6)重复执行步骤5),直到预测完选定阶段中所有任务的执行时间。

[0063] 7)重复执行步骤4)-5),直到所有阶段的所有任务的执行时间都预测结束。

[0064] 8)将$T_{j,start}$和$T_{j,end}$作为作业j的开始时间和结束时间返回。

[0065] 步骤八:预测复杂作业的执行时间。

[0066] 1)将作业看作简单作业，并根据步骤七预测作业的执行时间。

[0067] 2)从模型库中找到与该作业对应的模型，根据模型创建由该作业派生的所有作业并插入等待队列末尾。

[0068] 步骤九：如果$T_{start}>T_{j,start}$，$T_{start}=T_{j,start}$；如果$T_{j,end}>T_{end}$，$T_{end}=T_{j,end}$。

[0069] 步骤十：重复步骤六至步骤九，直到等待队列中最后一个作业的执行时间预测结束为止。记$T_{end}-T_{star}$为执行完等待队列中所有作业所需的时间。

[0070] 大数据环境中的作业可以分成简单作业和复杂作业。简单作业在一次执行结束后，产生最终输出结果。复杂作业在一次执行结束后，派生出新的作业，新作业经调度才可执行。在预测包括复杂作业在内的批作业执行时间时，由复杂作业派生的新作业的执行时间也应考虑在内。要预测派生的新作业的执行时间，需要掌握这类作业的阶段信息、任务信息等。在本发明中，通过为复杂作业建模的方式表达由其派生的新作业信息。

[0071] 通常，一个作业在同一个数据中心同类型节点上中无论执行多少次，每次执行时间都大体相同。在给定作业类型和数据规模的前提下，通过分析历史数据，获取该类作业不同阶段任务的执行时间，并将其作为预测同类型其它作业中任务执行时间的一个参数。任一阶段都同时存在多个任务并行执行，由于资源竞争等因素，这些任务的执行时间并不相同。本发明采用概率统计的方法对历史数据进行分析，并呈现多个统计结果，由用户自行决定采用哪个结果作为预测参数。

[0072] 任务执行时间与计算复杂程度、输入数据规模、计算节点的可用资源等相关，在分析结果中应该体现这些因素的影响，具体地，类型为“job-A”的作业包括两类任务，类型分别为“task-A”和“task-B”。在输入数据规模为256MB的前提下，“task-A”类型的任务在类型为“node-A”节点上在10000毫秒内执行结束的概率是100％，在9500毫秒内执行结束的概率是90％。“task-A”类型的任务在类型为“node-B”节点上在15000毫秒内执行结束的概率是100％，在9700毫秒内执行结束的概率是90％。在输入数据规模为234MB的前提下，“task-B”类型的任务在类型为“node-A”节点上在8000毫秒内执行结束的概率是100％，在7700毫秒内执行结束的概率是90％。“task-A”类型的任务在类型为“node-B”节点上在8100毫秒内执行结束的概率是100％，在8000毫秒内执行结束的概率是90％。

[0073] 在开始预测执行时间之前，必须先预测集群中所有时间片的可用时间。因为集群只在有空闲时间片的情况下，才会调度执行作业。因此，只有在获得所有时间片的可用时间后，才能预测作业的执行时间。当一个时间片正在执行任务时处于忙状态，反之处于空闲状态。忙时间片在任务执行完成后，转变成空闲状态。预测时间片的可用时间，实际上就是预测占用当前时间片的任务的完成时间。本发明提供了两种方法预测任务的完成时间，分别是基于历史数据的方法和基于执行进度的方法。

[0074] 在有的大数据框架中，作业中不同种类的任务需要占用不同种类的时间片执行。为应对这种情况，本发明对时间片进行分类管理，为每类时间片建立专门的队列，并按可用时间从小到大的顺序将时间片放入相应的队列。所有时间片都按照类型和可用时间组织好后，就可以开始预测作业的执行时间。

[0075] 本发明采用模拟真实集群调度执行作业的方式预测执行时间。根据集群中采用的调度策略，从等待队列中选择一个作业。模拟选定作业的任务调度过程建立时间片和任务之间的对应关系，从历史数据中找到与选定作业及时间片匹配的历史数据按照发明内容中

步骤五至步骤九所示方法预测作业的执行时间。

[0076] 具体地，以类型为job-A的作业j为例说明预测执行时间的过程，要求准确度为90％。假设j包含两个阶段，第一个阶段包含3个类型为task-A的任务，分别记为$task_{j,0}$、$task_{j,1}$和$task_{j,2}$，对应的输入数据的规模分别为：201MB，176MB和256MB。第二阶段包括一个类型为task-B的任务，记为$task_{j,3}$，其输入数据规模为192MB。通过模拟调度器的调度策略建立任务和时间片之间的映射关系：$task_{j,0}$->$S_{1,4}$，$task_{j,1}$->$S_{11,1}$，$task_{j,2}$->$S_{5,6}$和$task_{j,3}$->$S_{4,2}$，且$ts_{1,4} < ts_{11,1} < ts_{5,6} < ts_{4,2}$。$S_{1,4}$和$S_{11,1}$对应的节点类型为node-A，$S_{5,6}$和$S_{4,2}$对应的节点类型为node-B。记$T_{j,start}$，$T_{j,end}$分别为j开始执行的时间和结束执行的时间，二者分别取最大长整型数和最小长整型数。可按如下步骤预测j的执行时间：

[0077] (1)查找历史数据分析结果，找到与类型job-A匹配的信息。

[0078] (2)对于每个任务，根据预测准确度、任务类型、时间片所在的节点类型找到对应的任务完成时间和任务的输入数据规模。以$task_{j,0}$为例，要选取的完成时间为9500，这个时间对应的输入数据规模为256MB。

[0079] (3)检查与选定任务对应的时间片的可用时间，如果其值为-1，则将系统的当前时间赋值给它。以$task_{j,0}$为例，要检查$S_{1,4}$的可用时间，即检查$ts_{1,4}$的值。若$ts_{1,4}$=-1，则将当前系统时间赋值给它。

[0080] (4)如果$S_{1,4}$满足条件$ts_{1,4} < T_{j,start}$，$T_{j,start} = ts_{1,4}$。

[0081] (5)按照发明内容步骤七->5)->e)中给出的方法重新计算与任务匹配的时间片的可用时间，即$ts_{1,4} = ts_{1,4} + 9500 \times 201/256$。

[0082] (6)如果$S_{1,4}$满足条件$ts_{1,4} > T_{j,end}$，$T_{j,end} = ts_{n,i}$。

[0083] (7)按照可用时间从小到大的顺序，重新把$S_{1,4}$插入对应的时间片队列。

[0084] (8)重复(2)-(7)，直到没有任务可选时结束。

[0085] (9)记作业的执行时间为$T_{j,end} - T_{j,start}$。

[0086] 如果作业j是简单作业，执行时间的预测到此结束。如果是复杂作业，还需要根据发明内容中步骤一的分析结果，生成所有由作业j派生出的作业，并将其插入等待队列。

[0087] 以同样的方法预测等待队列中剩余作业的执行时间。通过比较各个作业的开始时间和结束时间，可以确定这批作业的执行时间：所有作业中最大的结束时间和所有作业中最小的开始时间之差即为这批作业所需的执行时间。