文章编号:1671-4598(2017)08-0234-03 DOI:10.16526/j.cnki.11—4762/tp.2017.08.060 中图分类号:TP393 文献标识码:A

多核环境下内存数据库并发调度技术优化研究

游 琪
(广东科学技术职业学院,广东 珠海 519090)

摘要:对多核环境下内存数据进行并发调度,可以减少计算机宕机次数和数据切换时时间,提高数据并发调度精度,增加数据操作平稳性;当前的内存数据并发调度方法是利用PrebuiltTrigger对内存数据进行并发调度,在调度过程中,没有设定具体的内存数据调度目标,导致内存数据库中的数据因此错乱无序,存在数据并发调度精度低的问题;为此,提出一种基于Linux的多核环境下内存数据并发调度优化方法;该方法首先采用IACT算法对影响调度进行的数据和内存数据库中相似或重复数据进行清洗,然后以清洗的数据为基础,利用启发式算法对其进行数据特征选取,依据多属性决策理论对内存数据并发调度的最优路径属性权重集合进行计算,以其结果为依据,计算调度最优路径的偏差值,最后利用最小偏差值,建立调度最优路径线性规划模型,对每条调度路径的综合决策属性值进行排序,由此得到调度的最优路径,完成对多核环境下内存数据的并发调度;实验结果证明,所提方法可以对多核环境下内存数据进行高效率地并发调度,提高了数据调度精度,增加了内存数据的可循环利用性,为低开销的内存数据调度提供了支撑。

关键词:多核环境;内存数据库;并发调度;技术优化

Multicore Environment Memory Database Technology on the Concurrent Scheduling Optimization

You Qi
(Guangdong Polytechnic of Science and Technology, Zhuhai 519090, China)

Abstract: To multicore chips to concurrent scheduling, memory data can reduce the number of computer downtime and data when switching time, improve the accuracy of data concurrent scheduling, increase the smoothness of data operation. The current memory data concurrent scheduling method is to use PrebuiltTrigger to concurrent scheduling of data memory, in the process of scheduling, memory data scheduling goal is set, lead to memory data in the database disorder disorderly, therefore, has the problem of low data concurrent scheduling precision. For this, put forward a multi—core environment based on Linux memory data concurrent scheduling optimization method. This method firstly adopts IACT algorithm affect scheduling and memory of data in the database of similar or repeated data cleaning, and then on the basis of data cleaning, using heuristic algorithm for feature selection, data on the basis of the theory of multiple attribute decision making memory data concurrent scheduling of the optimal path through the calculation of the attribute weights are set based on the results, calculation of dispatching optimal path deviation, the use of the minimum deviation, scheduling the optimal path to the linear programming model is established, and the path to each of the scheduling of comprehensive decision attribute value to sort, the resulting scheduling optimal path, complete the memory data concurrent scheduling in multi—core environment. The experimental results show that the proposed method can memory multicore environment data efficiently concurrent scheduling, improved the precision of data scheduling, increased the memory data of recycled, provides low overhead of memory data scheduling with support.

Keywords: multicore environment; memory database; concurrent scheduling; technology optimization

0 引言

当前,随着无线网络和科学技术的不断发展,多核环境下的网络内存数据大范围的应用于企业发展、教育教学、政府办公、体育竞赛、医疗服务、电网电信、娱乐餐饮等多个领域[1],在各个领域的迅速发展中起不可缺失的作用。多核环境下内存数据的并发调度不仅可以增加内存数据库的使用率,而且还可以提高无线网络的整体性能和网络使用寿命。因此,多核环境下内存数据并发调度受到了人们的广泛关注和高度重视[2-3]。由于多核环境下内存数据具有灵活性、高效性、可调节性等特点,所以需要对其进行调度[4]。多数内存数据调度方法在进行内存数据调度时,无法对其进行高兼容性、高精度,低误差率地调度,导致多核环境下内存数据调度时,往往会出现数据丢失、数据调度路径不明确、调度用时较长等问题[5]。在该种情况下,如何提高内存数据的查准率与查全率,减少调度时所用时间和调度精度偏差成为了亟待解决的问题。而基于Linux的多核环境下内存数据并发调度优化方法,可以对内存数据进行全方位、立体化地并发调度,是解决上述问题的有效途径[6],受到了数据调度研究人员的高度重视和频繁研究,成为了多核环境下内存数据并发调度研究专家学者的必修课题,与此同时也出现很多良好的成果[7]。

文献[8]提出了一种基于MySQL的多核环境下内存数据并发调度方法。该方法为了保证数据并发调度的精度,利用MySQL并发控制机保障内存数据调度的可行性,然后采用LnnoDB引擎激活内存数据并发调度器,最后依据并发调度器完成对内存数据的并发调度。该方法在对内存数据的并发调度实现中较为简单,但是存在调度时间较长的问题。文献[9]

收稿日期:2017-04-12; **修回日期**:2017-04-27。
基金项目:广东省高职教育一类品牌专业资助(2016gzpp007)。
作者简介:游 琪(1981-),女,江西九江人,硕士研究生,讲师,主要从事计算机应用、数据库方向的研究。

提出了一种基于 SDN 的多核环境下内存数据并发调度方法。该方法首先通过最优调度路径监测模块，对调度的最优路径进行实时监测，然后利用 SDN 技术根据调度缓存数据所占的比率和调度路径流量对内存数据库传输的数据进行调度，由此完成对多核环境下内存数据的并发调度。该方法虽然用时较短，但是存在内存数据并发调度精度低的问题。文献［10］提出了一种基于网络编码的多核环境下内存数据并发调度方法。该方法先利用网络编码将内存数据库中的数据节点进行排序，然后采用网络编码中的 P2P 流媒体推拉，与数据并发调度器进行连接，并将其转向数据库调度节点，最后依据 customr2 完成对多核环境下内存数据的并发调度。该方法虽然调度效率较高，但是在进行并发调度时数据库数据丢包率较大。

针对上述产生的问题，提出一种基于 Linux 的多核环境下内存数据并发调度优化方法。该方法首先利用 IACT 算法对影响并发调度的内存数据，和内存数据库中的相似或重复的数据进行清洗，然后以清洗的数据为基础，采用启发式算法进行数据特征选取，最后依据多属性决策理论确定数据调度的最优路径，完成基于 Linux 的多核环境下内存数据并发调度。

1　基于 Linux 的内存数据并发调度优化方法

1.1　内存数据的清洗与特征选取

为了保障多核环境下内存数据的并发调度速度更快，采用 IACT 算法对内存数据进行清洗，清洗过程中首先对影响并发调度的数据进行清洗，然后清洗内存数据库中的相似或重复的数据，由此完成对内存数据库中数据的清洗。

假设从内存数据库属性中清洗出一部分对数据的并发调度有影响的数据属性，达到改善数据库并发调度质量的目的。则清洗公式为：

$$SGF(z,x,c) = A(c/x) - A\{c/x \cup z\} \tag{1}$$

其中：$SGF(z,x,c)$ 代表清洗值，A 代表数据库清洗参数，z 代表数据库中某一数据属性值，x 代表数据库属性集，c 代表数据库清洗决策属性值。在数据清洗过程中，若 $SGF(z,x,c)$ 值越大，则数据属性 z 对数据库清洗决策属性 c 就越重要。相反，则数据属性 z 对数据库清洗决策属性 c 就越就越不重要。此时设定阈值 f，当 $SGF(z,x,c) \geqslant f$ 时，保留数据属性 z，当 $SGF(z,x,c) < f$ 时，对数据属性 z 进行清洗。当前数据调度方法中不含有数据清洗的过程，此步骤是对当前数据调度方法中此模块的优化。

以上述说明为依据，对内存数据库中的相似或重复数据进行清洗，清洗过程中，两个或几个数据是否重复或相似，需要通过数据属性匹配决定，利用 Smith－Waterman 算法对内存数据库中数据的属性相似度进行匹配，匹配公式为：

$$T_i = \sum_{i=1}^{z} SGF(z,x,c) \tag{2}$$

其中：T 代表内存数据库中数据的属性相似度匹配值，i 代表内存数据库中数据属性个数。

由此设定控制内存数据库中数据相似率和重复率的阈值，该阈值计算公式为：

$$W = \frac{SGF(z,x,c) \times \delta}{T_i} \tag{3}$$

其中：W 代表控制内存数据库中数据相似率和重复率的阈值，δ 代表内存数据清洗阈值。实验证明，该阈值范围在 0.64～0.65 时，对内存数据库中的相似数据和重复数据清洗的效率最高。

利用相似数据和重复数据清洗效率的指标来衡量相似数据和重复数据的清洗效率，该指标包括数据召回率和数据误识别率。其中，将数据召回率定义为：内存数据库中重复数据和相似数据中被清洗的记录，占内存数据库中真正含有重复数据和相似数据记录的百分比，以下叙述将内存数据中重复数据和相似数据统称为重复数据。则召回率表达式为：

$$\text{召回率} = \frac{\text{被清洗的重复数据记录}}{\text{所有重复数据记录}} \times 100\% \tag{4}$$

将数据误识别率定义为内存数据库中被错误清洗的数据占所有重复数据记录的百分比，表达式为：

$$\text{误识别率} = \frac{\text{误被清洗的数据记录}}{\text{所有重复数据记录}} \times 100\% \tag{5}$$

假设，数据召回率大于数据误识别率，则说明对内存数据库中的相似数据和重复数据清洗的足够彻底，假设，数据召回率小于数据误识别率，则说明对内存数据库中的相似数据和重复数据清洗的不够彻底，此时需要重复数据清理参数 α 对数据误识别率进行控制，其表达公式为：

$$Q = \alpha \times \left(\frac{\text{误被清洗的数据}}{\text{所有重复数据}} \times 100 \right) \tag{6}$$

其中：Q 代表重复数据清理参数 α 对数据误识别率控制值，综上所述，完成对内存数据的清洗。将清洗过的内存数据利用启发式算法进行数据特征选取，具体过程如下所示。

假设，G 代表内存数据特征集，其中是存放数据特征子集的，U 和 O 分别代表内存数据库中数据特征条件属性集和数据特征决策属性集，V 代表内存数据库中即将进行特征选取的数据集，B 代表内存数据库中数据特征不相容记录集，ξ 代表数据特征不相容记录阈值。对以上所给数据进行以下操作。

1）对内存数据库中数据特征不相容记录集 B 进行计算，将记录集中大于等于 ξ 的记录加至 $POS_U(O)$ 中，$POS_U(O)$ 代表内存数据库中数据特征决策属性集对数据特征条件属性集的影响值；

$$B = V - POS_U(O) \tag{7}$$

2）若 $POS_G(O) = POS_U(O)$，则 G 中的数据特征就是内存数据库中将要选取的数据特征，数据特征选取参数 λ 可以控制 $POS_G(O)$ 和 $POS_U(O)$ 的相等，由此给出关于 λ 的公式：

$$M = \lambda \times \sum_{i}^{O} POS_G(U) \tag{8}$$

其中：M 代表使上述等式成立的决策值，由 λ 的控制，使 $POS_G(O) = POS_U(O)$，G 中的数据特征就是将要选取的数据特征，完成对内存数据的特征选取，当前的调度方法对数据特征进行选取时，没有设定决策值，导致数据特征选取不明确，此步骤是对当前数据调度中数据特征选取进行了优化。

1.2　内存数据并发调度最优路径

利用多属性决策理论对内存数据的并发调度最佳路径进行选择，过程中首先对内存数据并发调度的最优路径属性权重信息集合进行计算，以计算结果为依据，计算数据调度最优路径的偏差值，然后利用最小偏差值，建立数据调度最优路径线性规划模型，最后对每条数据调度路径的综合决策属性值进行排序，得到内存数据调度的最佳路径，完成多核环境下内存数据的并发调度。

假设内存数据并发调度的路径有 j 条，将 2.1 中数据特征

相似或相同的数据进行合并，本文不对特征相似数据的合并进行研究。为了使不同的特征数据类在进行数据并发调度时，并发调度路径都能达到最优，令 ϕ 代表内存数据并发调度的最优路径属性权重信息集合，则：

$$\phi = \{\ln e_i^k - (t_i)\} - \ln e \tag{9}$$

其中：k 代表内存数据库中特征相似或相同的数据进行合并的个数，t 代表数据调度最佳路径的系数，e 代表内存数据并发调度中一个参数，实验证明，此参数取值在 0.04～0.05 间，数据调度误差率最小。由内存数据并发调度最优路径属性权重信息集合 ϕ 为依据，计算每个相同特征数据类的并发调度最优路径偏差值：

$$\min R = \ln e - \sum_{i=1}^{k} \phi \tag{10}$$

其中：R 代表相同特征数据类的并发调度路径偏差值，将内存数据并发调度最优路径偏差值降到最小，建立并发调度最优路径线性规划模型：

$$H = \min\left\{ \sum_{k}^{l} \sum_{j}^{i} [\ln e] - \ln k \right\} \tag{11}$$

其中：H 代表内存数据调度最优路径线性规划模型，l 代表内存数据库并发调度最优路径偏差值降到最小的计算次数，利用系数法将此模型简化为非线性最优路径规划模型，且该模型有最优解，也就是内存数据并发调度的最优路径。假设其最优解的属性权重值为（s_1，s_2，…，s_n），根据内存数据库中数据本身的属性权重和相似特征数据类属性权重，对于内存数据并发调度路径的认识值，利用平均算子建立每条内存数据并发调度路径的属性值。

$$\eta_i = \sum_{j=1}^{n} \sum_{k=1}^{n} (s_1, s_2, \cdots s_i) \tag{12}$$

其中：η 代表利用平均算子建立每条数据调度路径的属性值，n 代表数据并发调度最优路径控制阈值，将权重值（s_1，s_2，…，s_n）代入至上式进行迭代计算，可获得每条数据调度路径的综合决策属性值，对这些综合决策属性值进行排序便可以得到内存数据并发调度的最优路径。综 2.1 和 2.2 所述，完成了多核环境下内存数据的并发调度。

2 仿真实验结果与分析

为了证明基于 Linux 的多核环境下内存数据并发调度方法的整体有效性，需要进行一次仿真实验。在 MATLAB 的环境下搭建内存数据并发调度实验仿真平台。实验数据取自于网络数据调度研究所的 10 台计算机，利用本文所提方法对 10 台计算机中的内存数据进行并发调度，由此观察基于 Linux 的多核环境下内存数据并发调度方法的整体性能。表 1 为文献［8］所提方法、文献［9］所提方法和文献［10］所提方法与本文所提方法，在数据量（万个）相同时，内存数据库中的数据清洗时间（s）的对比。

表 1 不同方法下数据清洗时间对比

内存数据调度方法	数据清洗所用时间/s
MySQL	30
SDN	28
网络编码	20
Linux	8

分析表 1 可知，文献［8］、文献［9］和文献［10］与本文所提方法在数据清洗所占时间上差距很大，在相同的数据下，本文所提方法数据清洗时间明显低于文献所提方法，这种主要是因为本文方法在进行数据清洗时，为了使内存数据清洗时间尽量减少，采用 IACT 算法对内存数据进行清洗，清洗过程中首先对影响并发调度的数据进行清洗，然后对内存数据库中的相似或重复的数据进行清洗，所以从根本上大大减少了数据清洗时间，证明了本文所提方法的有效性。表 2 是利用本文方法对内存数据进行特征选取时，数据量（万个）与数据特征选取时间（s）的关系描述。

表 2 数据量与数据特征选取时间关系

内存数据量/万个	数据特征选择时间/s
500	3
1000	5
1500	7
2000	8
2500	9

通过表 2 可知，数据特征选择时间虽然随着内存数据量的增加也在不断增加，但是数据量的提高并没有对数据特征选择时间产生很大影响，在数据量为 1500 万个时数据特征选择时间开始按照 1 s 的速度增加，说明了利用启发式算法进行数据特征选取为数据并发调度节省了时间，证明了本文所提方法的整体有效性，可实践性和兼容性较强。为了说明本文所提方法的整体性能和可行性，δ 代表数据清洗阈值，观察该阈值取值范围对数据清洗效率（%）的影响，图 1 为数据清洗阈值对数据清洗效率（%）影响的描述。

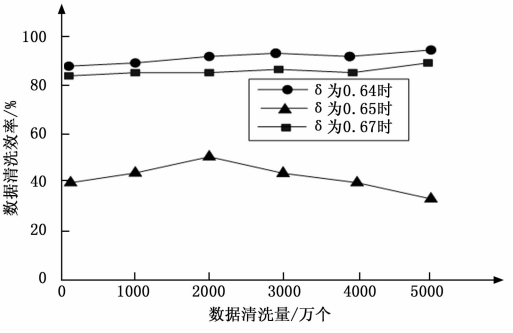

图 1 数据清洗阈值对数据清洗效率的影响

由图 1 可知，数据清洗阈值 δ 的取值范围对数据清洗效率的高低有很大影响，数据清洗阈值在 0.64－0.65 时，数据清洗效率相对较高，相比之下，当数据清洗参数为 0.67 时，虽然数据清洗量在 2000 万个之前数据清洗效率曲线呈上升趋势，但数据清洗效率曲线后半段呈下滑趋势，明显低于阈值在 0.64－0.65 时的数据清洗效率，进一步证明了本文所提方法的良好有效性。图 2 是内存数据并发调度参数 e 取值范围对数据调度误差率（%）的影响。

分析图 2 可知，数据调度误差率的曲线呈不断波动的趋势，但当内存数据并发调度参数 e 取值范围在 0.04～0.05 间时，数据调度误差率明显处于最低，因为在数据调度的最优路径选择时，对内存数据并发调度的最优路径属性权重信息集合进行了计算，而在计算中，调度参数 e 对调度的误差率有很大

（下转第 240 页）