数据立方体计算方法研究综述*)

侯东风　陆昌辉　刘青宝　张维明
(国防科学技术大学信息系统与管理学院　长沙 410073)

摘　要　随着多维数据分析在各领域的广泛应用，基于数据立方体的计算方法受到大量研究者的关注。分析了影响数据立方体计算的各种因素，其中包括数据存储空间、查询处理效率和数据立方体的维护消耗，并且阐述了数据立方体的物化策略。分别从冰山立方体、紧凑数据立方体、高维数据立方体、近似计算、流式数据立方体等几个方面综述了国内外现有的计算方法，分析了各种方法的特点以及适用范围。

关键词　数据立方体，多维数据，联机分析处理，计算方法

Survey on Computation of Data Cubes

HOU Dong-feng　LU Chang-hui　LIU Qing-bao　ZHANG Wei-ming
(College of Information System and Management, National University of Defense Technology, Changsha 410073, China)

Abstract　With the wide application of multidimensional data analysis in various fields, data cube computation attracts more and more attentions of researchers. We analyzed the influencing factors of computation which include the size of storage space, the query processing efficiency, and the cost of maintenance, and discuss the strategy of data cube materialization. The existing approachs of data cube computation were reviewed from the aspects of iceberg cube, compressed cube, high dimension data cube, approximate computation, stream cube and so on, the property and suitable area of the approaches were discussed in detail.

Keywords　Data cube, Multidimensional data, OLAP, Computation approach

随着数据仓库技术和联机分析处理(OLAP)技术的发展，多维数据查询与分析已经广泛应用到商务、金融以及军事等多个领域的信息处理中，为各行业的决策分析提供了强大的支持。为了支持有效的多维数据分析，1996 年，Gray 等首次提出了数据立方体(data cube)[1]的概念，从此基于数据立方体的计算方法一直是数据仓库和 OLAP 领域研究者所关注的热点问题。

数据立方体是实现多维数据查询与分析的一种重要手段。从本质上，多维数据的属性分为维属性和度量属性。维属性是观察数据对象的角度，而度量属性则反映数据对象的特征。对于多维数据分析而言，本质上是沿着不同的维度进行数据获取的过程。在数据立方体中，不同维度组合构成了不同粒度的子立方体(cuboid)，不同维值的组合及其对应的度量值构成了相应的数据单元(cell)。对于不同的查询和分析，需要访问不同的子立方体或者数据单元。因此，数据立方体的构建和维护等计算方法成为了多维数据分析研究的关键问题。

通常，数据立方体中所包含的数据量是非常大的。假设存在 n 个维度，并且每个维度不存在概念分层，则所包含的子立方体数目为 2^n。如果每个维度存在复杂的概念层次，或者维度的基数很大，则保存完整的数据立方体则需要海量的存储空间。虽然从理论上讲，所有的聚集查询能够通过原始数据计算获得，但是面对大量的数据，进行聚集计算的代价是相当昂贵的，因此需要考虑一定的策略，用于缩短查询响应时间，提高数据立方体的更新和查询效率。预先计算聚集值是一种应用比较普遍的方法，即预先计算不同粒度的数据单元，在查询的时候可以直接获得，但是也增加了存储容量和数据立方体维护的复杂度。因此，数据立方体的计算是在存储空间、响应查询时间和数据更新维护消耗等几个主要因素之间寻求有效的折衷。

在现有的计算方法中，主要存在 3 种物化策略：

①不物化：构建立方体的过程中不进行任何预先计算，在响应查询的时候临时计算所需求的聚集结果。很明显，这种方法虽然能够缩减存储空间和减少维护的代价，但是对查询而言会导致很慢的响应速度，在大多数的应用中无法满足要求。

②完全物化：预先计算整个立方体。这种选择虽然能够为快速查询提供支持，但是需要大量的存储空间保存预先计算的聚集值，从而导致存储空间的爆炸式增长，同时增加数据立方体维护的复杂性。尽管如此，从计算方法的角度出发还是具有一定价值的，有助于设计和实现部分物化计算方法。多路数组聚集方法[2]就是典型的代表，其中的分块计算方法和多路聚集计算方法在数据立方体计算中具有重要的作用。一些研究者还提出了基于多路聚集计算的冰山立方体计算方法[3,4]。

③部分物化：即按照一定的规则选择数据立方体的一个子集进行预先计算。这种选择是存储空间和响应时间的一种折衷。因此，在大多数的计算方法中采用部分物化策略。典

*)国家自然科学基金(70771110)。**侯东风**　博士研究生，研究方向为流式数据管理与多维数据分析；**陆昌辉**　博士，讲师，主要研究方向为多维数据建模；**刘青宝**　博士，副教授，主要研究方向为数据仓库技术和数据挖掘；**张维明**　博士生导师，教授，主要研究方向为军事信息系统、信息综合处理与辅助决策。

型的方法包括冰山立方体计算方法[3-13]、紧凑数据立方体计算方法[14-27]、外壳片段计算方法[28-30,49]等等。

随着数据形式和需求的发展，在一些实际应用中，用户只关心趋势的变化，因此对数据查询的精确程度要求不高。针对这种情况，一些研究者提出了近似的数据立方体计算方法[31-42]。这样的方法能够减少大量的存储空间，提高响应速度。另外，随着流式数据处理技术的发展，流立方体计算方法[43-48]越来越受到领域研究者的关注。

本文将主要针对数据立方体的各种计算方法进行分析和研究。第1部分介绍冰山立方体的计算方法，第2部分介绍紧凑立方体的计算方法，第3部分描述高维数据立方体的解决方案，第4和第5部分分别研究数据立方体的近似计算方法和流立方体计算方法，最后进行分析和总结。

1 冰山立方体计算方法

在冰山立方体的物化计算中，仅聚集和物化高于某个最小阈值的子立方体，这是一种部分物化的解决方法。这种计算方法的研究动机是数据立方体的空间多数被低度量值的数据单元所占据，而这些数据单元往往是分析者很少关心的内容。这种方法的优点是能够减少物化数据单元所占用的存储空间。另外，通过特定冰山条件能够实现数据分析的聚焦。冰山立方体的计算方法研究近几年产生了很多研究成果[3-13]。

Beyer提出了BUC算法[5]，用于计算冰山立方体。BUC算法采用了自顶向下的计算方法，即首先计算整个数据立方体的度量值，然后沿着每个维度进行递归搜索，同时检查冰山条件，对不满足条件的分枝进行剪枝操作。如果一个单元不满足冰山条件，则其后代不满足冰山条件。在BUC算法中使用了线性排序和快速排序提高数据划分的效率。文献[13]中结合数据仓库的压缩技术提出了两个基于BUC思想的冰山立方体计算方法。BUC算法中采用了分治策略(divide-and-conquer)，这种策略的优点是能够分担划分开销，减少不必要的计算消耗，但是BUC的性能容易受到维的次序和不平衡数据的影响，而且不能利用父子关系进行聚集，需要多次扫描数据集。

Star-Cubing方法[6,7]结合了多路数组聚集方法和BUC算法中的剪枝策略，利用星型树的数据结构进行存储，其中核心的部分是引入共享维的概念。如果共享维的聚集值不满足冰山条件，则共享维向下的所有单元都不满足冰山条件，可以根据这一条件进行剪枝。首先构建基本的星型树，按照深度优先搜索遍历每个子树，并且根据冰山条件进行剪枝操作，直至产生最终的星型树。这种方法的优点是提高了搜索的效率，但是同样对维的次序是敏感的。

MMCubing方法[9]是一种基于分解格空间的冰山立方体计算方法。这种方法基于一种观察，即在原始数据集中存在稀疏、密集的子集。由此，可以根据所包含的数据中维度元素的密集程度将整个格空间划分为一个稠密子空间和三个稀疏子空间。在算法中首先计算数值的频率并且进行排序和划分，确定每个维度的主要元素(Major)和次要元素(Minor)，完成格空间的分解。在稠密子空间中利用多路聚集方法进行计算，在稀疏子空间中利用递归调用方法进行检验。这种方法的优点是能够适应数据的分布，因而继承了不同方法的优点。但是分解格空间的过程相对比较复杂。

以上几种思路中均存在一个假设，即冰山条件是反单调的。若某个单元不满足条件，则其子孙单元也不满足条件。但是在有些情况下，并不是所有的冰山条件都是反单调的，例如度量值为平均值。H-Cubing方法[8]利用H树结构和度量值的转化用于计算具有复杂度量的冰山立方体。在这种方法中，并不直接计算单元的平均值，而转化成计算该单元所包含的基本单元Top-k平均值，并且利用这样的计算结果作为剪枝的判断条件，这样能够保证剪枝操作是安全的。文献[10]进一步扩展这种思想，提出了划分和近似策略，通过划分子空间将非单调的冰山条件转化为弱的或者强的单调条件进行计算。文献[3,11]中采用了界定分组聚集值的方法进行剪枝操作，能够支持分布和代数聚集函数。

C-Cubing方法[12]是基于数据立方体的封闭性度量(closedness)的计算方法，在这个方法中，结合了MMCubing和Star-Cubing计算方法，并且根据定义的封闭性度量，选择封闭的单元进行物化，提高了冰山立方体的计算效率，其中的消耗主要集中在数据单元的封闭性检查上。这是一种比较新颖的研究思路。

冰山立方体不仅能够在存储空间和处理时间上提高多维数据分析的效率，而且能够实现分析聚焦，在数据立方体的部分物化计算中起到重要的作用。尽管如此，冰山立方体的计算在一些应用中却存在一定的局限性，例如在动态性较强的流式数据处理中。因为流式数据是连续更新的，随着时间的推移，有新的数据加入到立方体中，同时有一些陈旧的数据被丢弃，从而由于部分数据的丢弃造成了无法准确计算度量值是否满足冰山条件。

2 保持语义的紧凑数据立方体计算方法

另外一种提高立方体计算效率的思路是通过共享元组来压缩数据立方体的存储，即紧凑数据立方体计算方法，这类方法的一个重要特点是能够保持数据立方体的钻取语义。各种方法在压缩的方式和表现形式上表现出不同的特征，其中包括浓缩立方体(Condensed cube)、侏儒立方体(Dwarf cube)、商立方体(Quotient cube)及其后继的QC-Tree等，这些都是近年来出现的一系列新型的数据立方体的存储结构。

浓缩立方体计算方法[14]引入了基本单一元组(Basic Single Tuple，BST)的概念，该方法是一种无损的数据立方体压缩存储策略。其特点是由单一元组产生的所有聚集单元具有相同的聚集值，因此可以将同一基本单一元组计算生成的多个聚集单元压缩成为一个单元表示。根据这个特点，该文献提出了BST浓缩立方体和最小BST浓缩立方体，并且证明了不同的最小BST浓缩立方体之间是相互等价的。这样的数据立方体结构对于稀疏数据立方体能够起到压缩作用，但是对于稠密的数据立方体存储效率不高。文献[15]进一步研究了关于浓缩数据立方体的索引和增量更新的问题，提出了一种CuboidTree索引结构。文献[16]进一步分析了浓缩立方体中的前缀冗余，提出了一种前缀立方体(PrefixCube)用于浓缩立方体的数据存储，提高了存储效率。文献[17]分析了计算最小浓缩立方体算法的复杂性，引入了纯BST和隐BST的概念，提出了一种快速计算最小浓缩数据立方体的SQCube计算方法，提高了浓缩立方体的计算效率。

侏儒立方体[18]在浓缩立方体基础上进行了发展，该方法利用一个有向无环图结构来存储数据，在构建的过程中可以同时大量减少前缀冗余(大量存在于稠密的数据立方体中)和后缀冗余(大量存在于稀疏的数据立方体中)。前缀扩张

(prefix expansion)和后缀聚合(suffix coalescing)是计算中重要的步骤。在一些共享相同后缀的元组之间,其对应的聚集值是相同的,从而引入了后缀冗余,因此可以考虑利用同一个节点表示。这种结构不仅适用于数据立方体的稠密部分,在稀疏部分的压缩能力也强于浓缩立方体。文献[19]证明了侏儒立方体占用的空间随着维度的增加呈多项式复杂度增长,还提出了一种估计侏儒立方体大小的有效方法。文献[20]中分析了构建侏儒立方体的算法,并且指出依然存在一定的后缀冗余,针对这些问题提出了一种 PID 计算方法,通过自底向上的划分计算方法构建侏儒立方体,并且提出压缩部分 ALL 单元的计算方法。

商立方体[21]按弱一致性划分(Weak Congruence Partition)将数据立方体中的数据单元划分为若干个互不相交的等价类,一个等价类包含多个等价的数据单元,利用同一个存储单元表示。在计算过程中采用了深度优先搜索的策略。事实上,可以采用各种不同等价类的划分标准和构造方法,因此可以将商立方体看成是此类结构的一个抽象。QC-Tree 是基于商立方体扩展的特例[25],它是使用树结构存储的、使用覆盖划分(Cover Partition)的商立方体。QC-tree 使用的划分标准为覆盖划分,只使用一个上界节点表示整个划分类,而且 QC-tree 的一个等价类内的节点是等价的,所以在上卷、下钻操作时可以直接互相转换。它在很大程度上压缩了数据,同时减少了数据对聚集数据的更新次数。这样,就极大限度地减少了数据的存储容量和数据维护的工作量。封闭立方体[22]中采用了立方体截线(cube transversals)和立方体闭包(cube closure)的概念,具有商立方体的表达能力,但是所占用的存储空间更小。此外,李盛恩等在商立方体的基础上对封闭立方体进行了深入研究[26],在压缩数据存储空间的同时,综合考虑了数据立方体的构建效率和查询响应效率。此外,商立方体的增量式更新算法也得到了广泛关注。基于 Galois 格的计算方法[23]是一种新颖的计算方法,覆盖划分本质上构成了 Galois 格,因此采用了传统的增量式格构造的思想,提高了计算效率。在上面的方法中,大多集中在分布式的聚集计算上,但在整体性的聚集计算上缺乏方法支持。文献[24]提出了基于商立方体的整体性聚集计算方法,例如中值的计算。

向隆刚等则从另外一个角度研究了这一问题,提出了下钻立方的概念[27]。在该方法中,首先构造冗余的下钻立方,然后通过广度优先遍历搜索消除冗余下钻信息,以更为直观的方式实现了数据立方体的构造。

上述的几种紧凑立方体计算方法不仅能够大幅度减少数据立方体的空间需求,而且能够保持数据立方体的钻取语义,有效解决了数据立方体计算中较高的存储和计算代价带来的问题。但是在处理快速响应和更新方面需要进一步深入研究。

3 高维数据立方体计算方法

在高维的数据立方体计算中,核心的问题是维度数量的增加对数据存储以及查询处理的影响。尽管上述的各种数据立方体计算方法能够在一定程度上压缩数据的存储空间,但是在高维情况下,仍然需要预先计算大量的数据单元,同时增加了数据立方体的构建和维护复杂性。因此,高维数据立方体的计算面临很大的挑战性。

一种思路是仅预先计算涉及少数维度的子立方体,就形成整个数据立方体的一个外壳。当涉及到其他维度的时候,则需要临时计算聚集结果。但是仅计算数据立方体的外壳,对于高维度的数据来说,需要聚集计算的数据单元数量仍然是相当大的。因此,相关研究者提出仅计算其片段[28]的方法,基于主要的观察是在 OLAP 过程中,只涉及到少数的几个维度。外壳片段的计算方法的主要思想是:给定高维数据集,将维划分为互不相交的维片段,并且将每个片段转换成为倒排索引,然后构造外壳片段立方体。这样可以利用预先计算的片段,动态组装和计算所需的子立方体单元。这种方法的优点是减少了计算数据立方体所需的数据空间,适用于高维数据的处理,同时能够快速响应涉及到少量维度的查询。但是在外壳片段立方体的过程中,维度的划分准则缺乏一定的依据,通常需要领域专家的参与,而且建立倒排索引的过程也是非常复杂的。刘运涛等在外壳片段计算方法的基础上,引入了分片计算的思想,提出了 CBFrag-Cubing 方法[29],该方法主要针对基数较小的高维数据集。同时利用位图索引结构提高了数据立方体的存储和计算效率,但是在维度基数较高的情况下性能会下降。

胡孔法等研究了分段共享数据立方体技术[30,49],将高维立方体划分成若干个低维立方体 mini-Cube,划分的方法同外壳片段计算方法相同,然后利用并行处理技术将每个不同的 mini-Cube 分别聚集计算,并利用维层次编码提高数据的检索效率。此外还研究了 mini-Cube 的增量式更新方法。

高维的数据立方体计算方法依然是研究的难点问题,特别是在动态环境下的高维数据处理。维度增长带来的复杂性成为了最大的影响因素,其中包括对数据的存储空间、创建和维护时间、查询响应时间等方面的影响。

4 近似计算方法

在前面所列举的方法中,存在的一个共同特点是数据立方体的精确表示。然而,在一些应用中,却面临着无法完全存储和精确查询的困难,例如在数据量非常大或者数据变化非常快的情况下。因此关于数据立方体的近似计算方法逐渐成为了一个研究热点。其主要思想是将数据信息压缩存储到一种有效的概要数据结构中,然后根据存储的概要信息对查询结果进行近似的估计,这是在计算效率和精确查询之间的一种折衷。常见的计算方法有基于直方图的计算方法、基于小波的计算方法以及其他的一些方法。

基于直方图的计算方法的主要思想是:将多维数据取值域按照数据分布划分为互不连接的区间,每个区间称为桶(bucket),在每个桶中保存区间内数据的聚集信息,用于响应近似的聚集查询。Poosala 等提出了基于 MHist 直方图的近似计算方法[31],采用了根据维度值和度量值进行划分的方法。在每一次划分中,选择边缘分布最符合约束条件的维度进行划分,从而保证了摘要信息的精确性。Gunopulos 等提出了 GenHist 直方图用于计算多维数据的近似查询[32],在这种方法中最大的特点是直方图中不同的桶之间允许相互重叠。这是一种基于数据密度的计算方法,能够准确反映数据在整个数据空间中的分布。但是在计算的过程中输入参数的确定是比较困难的,其中包括数据网格的划分参数、直方图桶的数量以及每次迭代网格划分参数的衰减因子。Furfaro 等提出了一种基于网格层次二元直方图(GHBH,Grid Hierarchical Binary Histograms)的计算方法[33],这种方法同样是以原始数据的数据分布为基础,在选取划分的维度和划分的位

置都遵从一定的划分准则。贪婪算法的采用使得算法能够有效利用存储空间。同时在划分中采用了基于网格的划分策略，为物理实现提供了极大的便利。

小波是一种强大的数学计算方法，Vitter 在文献[34]中提出了利用小波分解近似计算数据立方体的方法。其主要思想是：①利用原始数据立方体计算部分求和立方体(Partial Sum Data Cube)；②对部分求和立方体进行小波分解，得到一组 N 个小波系数；③根据一定的准则选取 m 个有代表性的小波系数进行保存，而将其他的小波系数置为 0，其中 $m \ll N$。根据保存的系数可以恢复近似的数据立方体信息，能够为近似的范围查询提供支持。在进一步的研究中将这一思想扩展到高维的稀疏数据立方体的处理[35]。Matias 将小波技术和直方图方法相结合，提出了基于小波的直方图计算方法[36]，其核心思想是利用 Haar 小波系数的紧凑子集作为数据的近似分布。文献[37]则对基于小波的计算方法进行了分析，提出了在原始数据立方体上进行小波分解，利用小波分解固有的多分辨特征，提出了渐进式的范围查询估计方法。

Quasi-Cube 是另外一种基于数据分布的估计方法[38]。在这种方法中，利用 loglinear 模型描述数据立方体中稠密的部分，并且根据一定的阈值保存一部分异常点。根据所保存的模型参数以及异常点能够支持一定误差范围内的近似查询，压缩了数据立方体占用的存储空间，而基于划分的建模方法能够提供更为精确的查询结果。同时提出了高效的数据动态更新方法。

Shanmugasundaram 等提出了利用混合概率密度模型描述数据立方体的方法[39]，在该方法中采用了分组聚类的方法，能够根据存储空间大小的变化调整聚类的结果，对于聚类得到的每个簇利用多元 Gaussian 分布密度函数表示。这种建模方法的优点是能够在连续维度上进行范围查询而无需离散化数据，但是混合模型的建立是一个计算量比较大的过程。

Cuzzocrea 提出了利用多项式近似方法估计数据立方体中范围查询的方法[40]。在该方法中分析了在近似数据立方体计算中的主要需求，采用了最小平方近似计算方法(LSA，Least Square Approximation)计算多维数据立方体的概要信息，同前面的几种方法类似，是基于数据分布函数的一种估计策略。该作者在文献[41]中分析了近似查询的两个主要问题：维度增长对计算的影响以及数据立方体的近似程度，并且提出了利用 KLT 变换进行维度约简的计算方法。针对数据查询近似程度的问题研究，Cuzzocrea 提出了基于 TP-Tree[42] 的计算方法，在该方法中将给定的数据立方体数据划分为异常值和正常值，分别进行组织和存储。对于异常点采用四分树索引结构进行组织，而正常数据则通过均一采样保存在 TP-Tree 中，对于不稳定的数据分布适应能力较强，同时基于异常点的管理提高了数据查询的准确性。另外一方面，TP-Tree 的重要特点是能够根据查询的负载调整数据的划分。

在上述的几种数据立方体近似计算方法中，由于不需要存储大量的聚集数据和原始数据，从而节省了大量的数据存储空间，同时提高了数据的查询处理速度，但是在数据精确性上存在一定的损失。有些方法[37,42]则提出了渐进式的查询结果估计方法，但是在计算的过程中需要进一步检索原始数据，会增加计算的复杂程度。因此，近似计算方法的研究重点依然集中在数据的存储效率、查询和更新效率和查询结果的精确性等几个方面。

5 流式数据立方体的计算方法

随着各种信息技术的发展，数据的存在形式趋于多样化，特别是在一些实时监控系统、通信网络、金融、科学工程实验以及传感器网络等动态环境产生的流式数据，为数据的查询与处理提出了新的要求。流立方体则是针对流式数据的多维分析提出来的解决方法，与传统的静态数据相比较具有截然不同的特点。但是在数据立方体的计算上存在同样的问题，聚集计算整个数据立方体也是不切合实际的，在处理过程中应该充分考虑数据的存储空间、查询与维护的效率等因素，而且具有更高的要求。随着流式数据处理技术的逐步完善和多维数据分析的发展，数据库领域的研究学者在流立方体的研究方面取得了一些进展。

Chen Yixin 等在文献[43]中提出了一种回归立方体(Regression Cubes)，用于计算流式数据的多维回归分析。其中涉及到三个主要部分：倾斜时间框架、关键层次表示和基于异常的计算和钻取。倾斜时间框架的一个主要特点是最近的时间粒度是精细的，而远一点的时间粒度是粗糙的，从而极大缩减了内存需求。关键层次由最小兴趣层和观察层次组成。在最小兴趣层和观察层之间依然存在大量的子立方体需要物化，在该文献中提出物化异常单元(exception cells)的方法。还提出了一种关键路径的物化方法。其中，在每个数据立方中，采用的是最小平方误差(Least Square Error)拟合的线性回归方法计算参数，并且以一种简洁的表示方式存储计算所需的参数，大大减少了数据存储的压力。文献[46]则提出了步进式的回归方法(Step-by-step Regression)和分割策略进行计算，将整个回归值域划分为若干个片段，对于每个片段分别进行计算，从时间和空间上提高了效率。

Chen Yixin 和 Han Jiawei 等在文献[44,45]中进一步深化了流式立方体的概念，重点分析了流式数据多维分析的一些需求，依然采用了上述的倾斜时间框架、关键层次和关键路径物化方法，设计了一种 H 树结构，实现了流立方体的初始化计算、增量式更新以及响应在线的查询。

Hershberger 等提出了一种新颖的多维流式数据概要计算方法[48]，采用了一种适应性空间划分(Adaptive Spatial Partitioning)的策略，对数据空间进行层次分解，并且利用 ASP 树结构保存概要信息，主要用于解决流式数据中的热点跟踪、范围查询计数、频繁元素发现以及分位点计算等问题。

Orlando 则提出了移动对象的轨迹计算数据立方体[47]。事实上，在移动对象的轨迹跟踪中，主要涉及到对象的位置以及时间维度，随着对象的移动，形成了一定的流式轨迹数据。在计算的过程中，利用 FM 草图计算轨迹的概要信息。

上述的研究为流式数据的多维分析提供了有效的解决方案，但是目前还处在初步研究阶段，还存在一些需要进一步研究的问题。例如立方体中关键层次以及关键路径还需要预先确定；在建立之后不能根据新的情况进行调整。此外，还需要考虑数据立方体的存储以及增量更新的效率等问题。

结束语 数据仓库和 OLAP 技术的发展迅速并且在很多领域得到了广泛应用，数据立方体计算作为核心问题取得了大量的研究成果。由于在数据存储空间、查询响应时间和数据立方体更新维护耗费等几个方面条件的约束下，使得数据立方体的计算面临很多挑战。

本文针对这些问题详细回顾了国内外在数据立方体计算方面的研究成果，详细分析了数据立方体计算中主要影响因

素，综述了冰山立方体计算方法、紧凑数据立方体计算方法、高维数据立方体计算方法、近似数据立方体计算方法以及流式数据立方体计算方法，详细分析了各种方法的主要优缺点和适用范围。

在进一步的工作中，数据立方体的计算方法依然是数据仓库与OLAP研究的热点问题。特别是数据立方体的近似计算和流式数据立方体的计算问题，需要更加深入的研究和发展。

参 考 文 献

[1] Gray J, Bosworth A, Layman A, et al. Data cube: A relational aggregation operator generalizing group-by, cross-tab, and sub-totals// Proc. of the 12th IEEE Intl. Conf. on Data Engineering. Vienna, 1996: 152-159

[2] Zhao Y, Deshpande P M, Naughton J F. An array-based algorithm for simultaneous multidi-mensional aggregates // Proc. of the ACM SIGMOD Intl. Conf. on Management of Data. Tucson, AZ, 1997: 159-170

[3] Chou L, Zhang X. Computing complex iceberg cubes by multiway aggregation and bounding// Proc. of the 6th Intl. Conf. on Data Warehousing and Knowledge Discovery. Zaragoza, Spain, 2004

[4] Chou L, Zhang X. Multiway iceberg cubing on trees. Technical Report. School of CS IT, RMIT University, 2005

[5] Beyer K, Ramakrishnan R. Bottom-upcomputation of sparse and iceberg cubes// Proc. of the ACM SIGMOD Intl. Conf. on Management of Data. Philadelphia, PA, 1999: 359-370

[6] Xin Dong, et al. Star-Cubing: Computing Iceberg Cubes by Top-Down and Bottom-Up Integration// Proc. of the 29th Intl. Conf. on Very Large Data Bases. Berlin, Germany, 2003

[7] Xin Dong, Han Jiawei, et al. Computing Iceberg Cubes by Top-Down and Bottom-Up Integration: The StarCubing Approach. IEEE Transactions on Knowledge and Data Engineering, 2007, 19(1): 111-126

[8] Han Jiawei, et al. Efficient Computation of Iceberg Cubes with Complex Measures// Proc. of the ACM SIGMOD Intl. Conf. on Management of Data. Santa Barbara, California, USA, 2001

[9] Shao Zheng, et al. MM-Cubing: Computing Iceberg Cubes by Factorizing the Lattice Space// Proc. of the 16th Conf. on Statistical and Scientific Database Management. Santorini Island Greece, 2004

[10] Wang Ke, Jiang Yuelong, Yu J Xu, et al. Divide-and-Approximate: A Novel Constraint Push Strategy for Iceberg Cube Mining. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(3): 354-368

[11] Zhang Xiuzhen, Chou P L, Dong Guozhu. Efficient Computation of Iceberg Cubes by Bounding Aggregate Functions. IEEE Transactions on Knowledge and Data Engineering, 2007, 19(7): 903-918

[12] Xin Dong, Shao Zheng, Han Jiawei, et al. C-Cubing: Efficient Computation of Closed Cubes by Aggregation-Based Checking // Proc. of the 22nd Intl Conf. on Data Engineering. Atlanta, Georgia, USA, 2006

[13] 骆吉洲，李建中，赵锴. 大型压缩数据仓库上的 Iceberg Cube 算法. 软件学报，2006，17(8)：1743-1752

[14] Wang W, et al. Condensed Cube: An Effiective Approach to Reducing Data Cube Size// Proc. of the 18th IEEE Intl Conf. on Data Engineering. San Jose, California, USA, 2002

[15] Feng Jianlin, Si Hongjie, Feng Yucai. Indexing and Incremental Updating Condensed Data Cube// Proc. of the 15th Intl Conf. on Scientific and Statistical Database Management. Cambridge, MA, USA, 2003

[16] Feng Jianlin, Fang Qiong, DingHulin. PrefixCube: Prefix-sharing Condensed Data Cube // Proc. of ACM 7th Intl Workshop on Datawarehouse and OLAP. Washington, DC, USA, 2004

[17] 王琢，鲍玉斌. 一种快速生成最小浓缩数据立方的算法. 小型微型计算机系统，2005，26(12)：2212-2215

[18] Sismanis Y, Deligiannakis A, Roussopoulos N, et al. Dwarf: shrinking the petacube// Proc. of the ACM SIGMOD Intl Conf. on Management of Data, Madison, Wisconsin, USA, 2002: 464-475

[19] Sismanis Y, Roussopoulos N. The Dwarf Data Cube Eliminates the High Dimensionality Curse. Technical Report. University of Maryland, Available online http://hdl. handle. net/1903/1333

[20] Xiang Longgang, Feng Yucai, Gui Hao. Construction and compression of Dwarf. Journal of Zhejiang University SCIENCE, 2005, 6A(6): 519-527

[21] Lakshmanan L, Pei J, Han J. Quotient cube: How to summarize the semantics of a data cube // Proc. of the 28th Intl Conf. on Very Large Data Bases. Hong Kong, China, 2002: 778-789

[22] Casali A, Cicchetti R, Lakhal L. Extracting Semantics from Data Cubes Using Cube Transversals and Closures // Proc. of the ACM SIGMOD Intl Conf. on Management of Data. Washington, DC, USA, 2003

[23] Li Cui-Ping, Tung Kum-Hoe, Wang Shan. Incremental maintenance of quotient cube based on galois lattice. Journal of Computer Science & Technology, 2004, 19(3): 302-308

[24] Li Cuiping, Cong Gao, Tung K H, et al. Incremental Maintenance of Quotient Cube for Median // Proc. of the 10th ACM SIGKDD Intl Conf. on Knowledge Discovery and Data Mining. Seattle, Washington, USA, 2004

[25] Lakshmanan L, Pei J, Zhao Y. Qc-trees: An efficient summary structure for semantic OLAP// Proc. of the ACM SIGMOD Intl. Conf. on Management of Data. San Diego, California, USA, 2003: 64-75

[26] 李盛恩，王珊. 封闭数据立方体技术研究. 软件学报，2004，15(8)：1165-1171

[27] 向隆刚，龚健雅. 一种高度浓缩和语义保持的数据立方. 计算机研究与发展，2007，44(5)：837-844

[28] Li Xiaolei, Han Jiawei, Gonzalez H. High-Dimensional OLAP: A Minimal Cubing Approach// Proc. of the 30th Intl Conf. on Very Large Data Bases. Toronto, Canada, 2004: 528-539

[29] 刘运涛，鲍玉斌，等. CBFrag-Cubing：一种基于压缩位图的高维数据立方创建算法. 计算机科学，2005，32(11)：91-93

[30] 胡孔法，陈岐，等. 数据仓库系统中高维联机分析处理聚集数据存储技术研究. 计算机集成制造系统，2006，12(7)：1095-1101

[31] Poosala V, Ganti V. Fast Approximate Answers to Aggregate Queries on a Data Cube// Proc. of the 11th Intl Conf. on Statistical and Scientific Database Management. 1999

[32] Gunopulos D, Kollios G, Tsotras V J. Approximating Multi-dimensional Aggregate Range Queries over Real Attributes // Proc. of the ACM SIGMOD Intl. Conf. on Management of Data. Dallas, Texas, United States, 2000

[33] Furfaro F, Mazzeo G M, Sacca D, et al. Hierarchical Binary Histograms for Summarizing Multi-dimensional Data// Proc. of the 2005 ACM Symposium on Applied Computing. Santa Fe, New Mexico, 2005

（下转第 32 页）

Boston: IEEE Press, 2005: 147-158
[8] 贺鹏,李建东,陈彦辉,等. 基于 Delaunay 三角剖分的 Ad Hoc 网络路由算法. 软件学报, 2006: 49-54
[9] Gabriel K R, Sokal R R. A new statistical approach to geographic variation analysis. Systematic Zoology, 1969: 259-278
[10] Toussaint G. The relative neighborhood graph of a finite planar set. Pattern Recognition, 1980: 261-268
[11] Seada K, Helmy A, Govindan R. On the effect of localization errors on geographic face routing in sensor networks // Proceedings of the Third International Symposium on Information Processing in Sensor Networks (IPSN). ACM Press, 2004: 71-80
[12] Kim Y J, Govindan R, Karp B, et al. On the pitfalls of geographic face routing // Proceedings of DIAL-M-POMC. ACM Press, 2005: 34-43
[13] Kim Y J, Govindan R, Karp B, et al. Geographic routing made practical // Proceedings of NSDI 2005. CA: ACM Press, 2005: 217-230
[14] Kim Y J, Govindan R, Karp B, et al. Lazy Cross-Link Removal for Geographic Routing // Proceedings of the 4th International Conference on Embedded Networked Sensor Systems. Boulder: ACM Press, 2006: 112-124
[15] Kuhn F, Wattenhofer R, Zollinger A. Asymptotically optimal geometric mobile ad-hoc routing // Proceedings of the of the 6th International Workshop on Discrete Agorithms and Methods for Mobile Computing and Communications (Dial-M). ACM Press, September 2002: 24-33
[16] Kuhn F, Wattenhofer R, Zollinger A. Worst-Case Optimal and Average-case Efficient Geometric Ad-Hoc Routing // Proceedings of the 4th ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc). 2003
[17] Fang Q, Gao J, Guibas L, et al. GLIDER: Gradient landmark-based distributed routing for sensor networks // Proc. of the 24th Conference of the IEEE Communication Society (INFOCOM). volume 1, March 2005: 339-350
[18] Fang Qing, Gao Jie, Guibas L J. Landmark-based Information Storage and Retrieval in Sensor Networks // Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom). 2006
[19] Nguyen A, Milosavljevic N, Fang Qing, et al. Guibas. Landmark Selection and Greedy Landmark_descent Routing for sensor Networks // Proceedings of the 26th Conference of the IEEE Communication Society (INFOCOM). 2007
[20] Funke S, Milosavljevi'c N. Guaranteed-delivery Geographic Routing Under Uncertain Node Locations // Proceedings of the 26th Conference of the IEEE Communication Society (INFOCOM). 2007
[21] Bruck J, Gao J, Jiang A. MAP: Medial axis based geometric routing in sensor networks // Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCOM). 2006: 88-102
[22] Fonseca R, Ratnasamy S, Culler D, et al. Beacon vector routing: Scalable point-to-point in wireless sensornets. IRB-TR-04-012. Berkeley: Intel Research, 2004: 1-14
[23] Newsome J, Song D. GEM: graph EMbedding for routing and data-centric storage in sensor networks without geographic information // Proceedings of the 1st International Conference on Embedded Networked Sensor Systems. Los Angeles, California, USA, November 2003
[24] Zhang Fenghui, Li Hao, Jiang Anxiao A, et al. Face Tracing-based Geographic Routing in Nonplanar Wireless Networks // Proceedings of the 26th Conference of the IEEE Communication Society (INFOCOM). 2007
[25] Fang Q, Gao J, Guibas L. Locating and Bypassing Routing Holes in Sensor Networks // Proceedings of the 23th Conference of the IEEE Communication Society (INFOCOM). 2004
[26] Leong Ben, Liskov B, Morris R. Geographic Routing without Planarization // Proceedings of the 3rd Symposium on Network Systems Design and Implementation (NSDI 2006). San Jose, CA, May 2006
[27] Arad N, Shavitt Y. Minimizing Recovery State in Geographic Ad-Hoc Routing // Proceedings of the 7th ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc). 2006

(上接第 5 页)

[34] Vitter J S, Wang M, Iyer B. Data Cube Approximation and Histograms via Wavelets // Proc. of the 7th ACM Intl Conf. on Information and Knowledge Management. Bethesda MD USA, 1998
[35] Vitter J S, Wang M. Approximate Computation of Multidimensional Aggregates of Sparse Data Using Wavelets // Proc. of the ACM SIGMOD Intl Conf. on Management of Data. Philadelphia, Pennsylvania, US, 1999
[36] Matias Y, Vitter J S, Wang M. Dynamic Maintenance of Wavelet-based Histograms // Proc. of the 26th Intl Conf. on Very Large Data Bases. Cairo, Egypt, 2000
[37] Wu Yi-Leh, Agrawal D, Abbadi A E. Using Wavelet Decomposition to Support Progressive and Approximate Range-Sum Queries over Data Cubes // Proc. of 2000 ACM Intl Conf. on Information and Knowledge Management. McLean, VA, USA, 2000
[38] Barbara D, Wu Xintao. Loglinear-based quasi cubes. Journal of Intelligent Information Systems, 2001, 13(3): 255-276
[39] Shanmugasundaram J, Fayyad U, Bradley P S. Compressed Data Cubes for OLA PAggregate Query Approximation on Continuous Dimensions // Proc. of the 5th ACM SIGKDD Intl Conf. on Knowledge Discovery and Data Mining. San Diego, CA, USA, 1999
[40] Cuzzocrea A. Improving range-sum query evaluation on data cubes via polynomial approximation. Data & Knowledge Engineering, 2006, 56(2): 85-121
[41] Cuzzocrca A. Overcoming Limitations of Approximate Query Answering in OLAP // Proc. of the 9th Intl Database Engineering & Application Symposium. 2005
[42] Cuzzocrea A, Wang Wei. Approximate range-sum query answering on data cubes with probabilistic guarantees. Journal of Intelligent Information Systems, 2007, 28(2): 161-197
[43] Chen Yixin, Dong Guozhu, Han Jiawei, et al. Multi-dimensional regression analysis of time-series data streams // Proc. of the 28th Intl Conf. on Very Large Data Bases. Hong Kong, China, 2002: 323-334
[44] Chen Yixin, Dong Guozhu, Han Jiawei, et al. Online Analytical Processing Stream Data: Is It Feasible? // ACM SIGMOD Workshop on Research Issue in Data Mining and Knowledge Discovery. Madison, Wisconsin, USA, 2002
[45] Han Jiawei, et al. Stream Cube: An Architecture for Multi-dimensional Analysis of Data Streams. Distributed and Parallel Databases, 2005, 18(2): 173-197
[46] Liu Chao, Zhang Ming, Zheng Minrui, et al. Step-by-Step Regression: A More Efficient Alternative for Polynomial Multiple Linear Regression in Stream Cube // Proc. of the 7th Pacific-Asia Conf. on Knowledge Discovery and Data Mining. Seoul, Korea, 2003
[47] Orlando S, Orsini R, Raffaetà A, et al. Spatio-Temporal Aggregations in Trajectory Data Warehouses // Proc. of the 9th Intl Conf. on Data Warehousing and Knowledge Discovery. Regensburg, Germany, 2007
[48] Hershberger J, Shrivastava N, Suri S, et al. Adaptive Spatial Partitioning for Multidimen- sional Data Streams. Algorithmica, 2006, 46(1): 97-117
[49] Hu Kong-fa, Ling Chen, Jie Shen, et al. Computing High Dimensional Mola Pwith Parallel Shell Mini-cubes // Proc. of the 2nd Intl Conf. on Fuzzy Systems and Knowledge Discovery. Changsha, China, 2005