基于负载均衡的空间线分组算法

张纯金[1]，魏海涛[2,3]，杜云艳[3]
(1. 山东科技大学 网络与信息中心，山东 青岛 266590；2. 山东科技大学 测绘科学与工程学院，山东 青岛 266590；
3. 中国科学院地理科学与资源研究所 资源与环境信息系统国家重点实验室，北京 100101)

摘　要：针对传统并行操作计算效率低的问题，提出以分组并行处理模式优化节点间的负载均衡。以表层浮标轨迹验证涡旋实验为例，给出面向不可分割空间线对象的快速分组方法，设计了两个分组调整算法。实验结果显示，算法可以使每个计算节点达到负载均衡。与串行计算的比较实验结果显示，算法具有较好的加速效果，且加速比随着计算节点个数的增加呈上升趋势。因此，基于负载均衡的空间线分组算法是对不可分割空间线的计算进行优化的有效途径。

关键词：负载均衡；线对象；并行处理；分组；浮标轨迹

中图分类号：TP301.6　　**文献标志码**：A　　**文章编号**：1672-3767(2014)06-0097-06

The Algorithm of Spatial Line Grouping Based on Load Balancing

Zhang Chunjin[1], Wei Haitao[2,3], Du Yunyan[3]
(1. Network Information Center, Shandong University of Science and Technology, Qingdao, Shandong 266590, China;
2. College of Geomatics, Shandong University of Science and Technology, Qingdao, Shandong 266590, China;
3. State Key Laboratory of Resources and Environmental Information System, Institute of Geographic Sciences and Natural Resources Research, CAS, Beijing 100101, China)

Abstract: To solve the problem of low computation efficiency in traditional parallel task operation, this paper proposed a grouping parallel processing mode for optimizing load balancing among computation nodes. The quick grouping algorithm for grouping the indivisible spatial line objects was applied to the vortex validation experiment, which is implemented by tracking surface buoy. Moreover, two group adaptation algorithms were given. The algorithm proposed in this paper balanced the load of each computation node and achieved perfect speedup. The speedup ratio increased with the numbers of computation nodes. The experimental results show that this spatial line grouping algorithm is an effective approach for optimizing the computation of indivisible spatial linear objects.

Key words: load balancing; the line object; parallel processing; grouping; buoy trajectory

在数据处理过程中，提高算法性能的重要途径是通过分组，即采用分而治之的问题处理模式，加快数据的处理速度。分组并行分为数据并行和任务并行两种。数据分组并行常用于处理模式较复杂的空间信息。目前，空间对象的分组算法多集中于点对象的分组，对于空间线的操作多集中于将线分割为线段进行分组[1-6]，或采用 MBR(minimum bounding rectangle)[7-8]、质心[4]化繁为简，转变为对点数据的分组。然而，在实际的应用中，空间线是不可分割的[9-11]，采用上述方法无法做到数据方面的负载均衡。本研究针对本身没有特殊拓扑关系，而组成空间线的各个点之间具有拓扑关系的线数据，提出空间线数据分组方法。

收稿日期：2014-10-14
基金项目：国家海洋公益性行业科研专项(201105033，201105017)
作者简介：张纯金(1977—)，男，山东郓城人，工程师，博士研究生，主要从事算法设计方面的研究. E-mail: zhangchjin@163.com
魏海涛(1979—)，女，山东滨州人，博士研究生，主要从事数据切分算法研究，本文通信作者.
E-mail: weiht@lreis.ac.cn

1 基于负载均衡的空间线分组算法原理

数据分组算法是面向应用解决高性能问题的重要手段，在 Argo 浮标轨迹数据提取涡旋的验证应用中，由于数据量比较大(点数据个数为 3 091 256)，算法的计算效率较低。此外，经实验分析，针对 Argo 的每条轨迹是不可分割的最小计算单元，时间的消耗除了和轨迹的点个数有关，最重要的是判断该轨迹成环及对环拟合成椭圆的算法。因此，在任务调整时，除了考虑每个漂流浮标轨迹点的个数，还要兼顾该浮标出现环的个数。基于以上特点，本研究应用式(1)为浮标轨迹设置一个体现其计算量的值：

$$C=W_1\times P+W_2\times L, \tag{1}$$

$$P_{\min}=\frac{D_{\min}}{S}。 \tag{2}$$

其中：一个浮标轨迹上，C 代表计算量，P 代表点的个数，L 代表可能出现的环的个数，W_1，W_2 分别代表点数和环数在整个数据处理中的权重，由实验可得，取值分别为 0.25 和 0.75(同等数据量下，环状数据和不成环状数据处理所用的时间比值)；$P_{\min}$ 代表可能成环的最小浮标点个数，$D_{\min}$ 代表涡旋的最小时间周期，S 代表浮标最小的采样周期。

以每个浮标为单位，做 DBSCAN(density-based spatial clustering of applications with noise)[12-13]计算，其中扫描半径为涡旋的最大半径，最小点个数为 $P_{\min}$，在查询核心点时，若该点为核心点，则其扫描半径内的点不再参与下一个核心点和边缘点的查找，并记录该核心点在扫描半径内包含的点个数 N，L 为 DBSCAN 核心点的个数乘以每个核心点内可能包含的环个数($N/P_{\min}$)。

针对以上空间线对象数据量较大和不可分割的特性，本分组算法包括两步：首先，对空间线数据进行粗粒度的分组，把临近的线对象放到一组里，组成一个不可分割的新对象；然后，基于粗粒度分割产生的对象集合，根据负载均衡原则进行再分组和组间对象调整，使各组间对象的个数大体相同。下面详细介绍这两步的实现细节。

1.1 基于 List 的空间线数据分组方法

对大量浮标进行长期数据采集决定了浮标空间数据海量的特性。以每个线对象为单位做负载均衡的调整时，I/O 开销是不可忽视的。为了解决运行效率瓶颈，算法的第一步实现负载均衡调整，即应用时间片轮转法对线对象进行粗略分组，将分组后的每一组作为一个对象，在后续的调整算法中把此对象组看作不可分割的最小元素。

基于负载均衡的算法调整步骤如图 1 所示。以每个线对象的 ID 为主索引创建 List 列表，并根据线对象中的点个数升序排列，采用时间片轮转法进行线对象的初次分组，(1)(2)(3)(4)……依次分配，尽量达到每个集合的数据量均衡。组内线对象的个数应适当，若个数太少，后续调整算法中涉及到的组个数就会很多，从而引起后续调整算法时间消耗过大；同理，太多会无法达到较好的负载均衡。本文集合内部线对象的个数为 $\left\{G_n \mid G_n\leqslant\frac{L_n}{2^{16}}\text{的最大偶数}\right\}$，其中：$G_n$ 为组内对象的个数，L_n 为线对象的总个数，2^{16} 代表数组的上限(与操作系统的位数有关)，G_n 值取偶数为了尽可能保证数据量的负载均衡。

图 1 数据分组示意图

Fig. 1 The schematic of data grouping

1.2 基于负载均衡的线对象集合调整算法

基于 List 的线数据分组后，以每个集合为对象，通过顺序索引记录线对象上的点和组内参与处理的点的总数(为了提高效率，可采用多线程实现加速)，通过分组算法中的初次调整和再次调整达到数据量的负载均衡。该算法的核心是根据计算节点的个数 n，将矢量目标集(顶点数量为 s)均衡划分为 n 份，各个计算节点能得到大约 s/n 个目标矢量集的目标顶点。

基于数组 $A\{n_1,n_2,\cdots,n_i\}$，n_i 代表经过粗粒度分组后第 i 组线对象集合中点的个数，进行初次调整和再

次调整。

1)初次调整

初次调整的主要思想为:从升序数组的大元素开始取,用小元素值做补充,以每组中元素值的和为依据,进行组合,得到初次的调整结果。

调整流程如图 2 所示:对存放在图 1 中的各个集合(集合 1、集合 2…)内各个数组求和,数组内对象按着和值升序排列;临界值 S_c 代表理想状态下分组后每组数据的元素值(每组点数据的个数);如果 A 中存在一个元素 X 的值大于 S_c,由于对象不可分割的特点,X 被分离出来单独构成一组,并不参加再次调整算法的执行。其他元素的分组算法的代码如下:

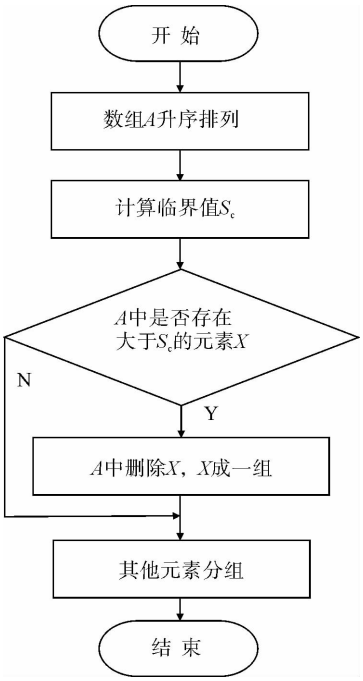


图 2 初次调整流程图

Fig. 2 The flow chart of the first adjustment

```
①  for i from 0 to 数组 A 的长度−1 step 1
     g<−1;   // 记录找的第几组数据
   p1<−S_c−数组 A 中的最大值 M;
   if p1>0 M 从数组 A 中删除,加入到数组 A_g 中;
     for j from 0 to 数组 A 的长度−1 step 1
         if a[j]<p1 且 a[j+1]>p1 then
           p2<−p1−a[j];p1<−p2;a[j]从数组 A 中
删除,加入到数组 A_g 中
              else
                   if g=n−1 then A_{g+1}<−A;
                     else g<−g+1;return ①;
                   end if
               end if
           end for
       end if
   end for
```

2)再次调整

为了最大限度地达到数据负载均衡,将初次调整的结果进行再次调整,计算过程如下:

- @将数组 A_1,A_2,…分别进行由小到大的排序;
- 计算每个数组的各个元素之和;
- 选出和值最大数组和最小数组;
- 两个和值相减值为 t;
- 最大数组的各个元素的值减最小数组的值得各个元素 s;
- if $0<s<t$ then 两数组中这两个元素互换;跳转到@;
- end if

3)线对象集合调整实例

假设数组 A 为{41,16,72,19,38,39,17,48,73,31,3};

初次分组的结果为$\underbrace{\{3,16,48,73\}}_{140}$,$\underbrace{\{19,41,72\}}_{132}$,$\underbrace{\{17,31,38,39\}}_{125}$;

再次调整后的分组结果为$\underbrace{\{16,31,38,48\}}_{133}$,$\underbrace{\{3,17,39,73\}}_{132}$,$\underbrace{\{19,41,72\}}_{132}$。

分组后三个数据集合的点个数分别为 133,132,132,本分组算法能实现数据的负载均衡。

2 实验与分析

本研究通过表层浮标轨迹涡旋[11]的实验来验证以上算法的有效性。表层浮标轨迹中，参与运算的点的数量是影响计算时间的重要因素，但每个浮标的轨迹是一个整体，不能分割，因而，通过以上提出的基于负载均衡的空间线分组算法提高运算效率。

2.1 实验取值

实验采用的数据是太平洋区域的表层漂流浮标数据，时间跨度为 1979-02-14—2010-06-30，表层浮标位置点共有 9 397 340 个，共有 4 970 条浮标轨迹。为了进行对比实验，随机抽取了 4 个数据集，数据详细信息如表 1 所示。

表 1 数据概况

Tab. 1 Summary of data sets

数据集	点个数	轨迹个数
DS-1	8 997 341	4 970
DS-2	7 988 159	3 864
DS-3	4 514 573	3 249
DS-4	3 015 756	2 840

图 3 是采用 Argo 轨迹提取涡旋的算法[14]从表层浮标轨迹提取涡旋所消耗的时间对比图，可以看出，当数据量变大时(DS-4 到 DS-1)，其轨迹条数也不断增多，处理这些数据所耗费的时间却呈现出类似二次曲线的上升规律。当数据量巨大时，将花费大量的处理时间，因此在计算之前必须进行有效的分组。

2.2 算法优越性分析

1)分组后的计算量均衡性分析

从表层浮标数据产生的轨迹数据中随机抽取 4 个数据集，通过分组，衡量每组计算量，结果如图 4 所示。图 4(a)中，针对任意一个数据集(DS-1～DS-4)，分组后每个小组包含的计算量基本相同，数据量曲线呈现近似直线状态。图 4(b)为数据集 DS-1 分组结果的放大图，图 4(c)是数据集 DS-4 分组结果的放大图，由两图可知，分组后每组数据的实际计算量之间存在差异，为了进一步验证不同数据集间差异的大小是否影响到整体任务的均衡效果，从 DS-1 到 DS-4 的切分实验中随机抽取 30 个样本，做分组后的数据量与理想等分的数据量之间差异性的 t 检验，$t_{0.05/2.5}=1.857>t=0.4$ 即 $P>0.05$，按双侧检验的水准 $a=0.05$，表示两者的差别无统计意义，分组后的样本与期望的总体平均数是无差异的，因此，在计算量较大的情况下(若计算量不大，就失去了分组的必要性)，各组内对象的个数差异可以忽略不计。由此可得，本算法达到了数据负载均衡的要求。

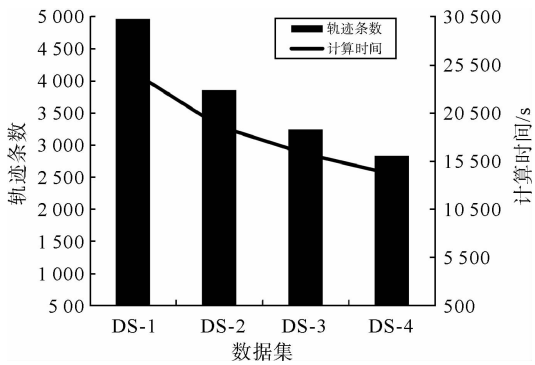
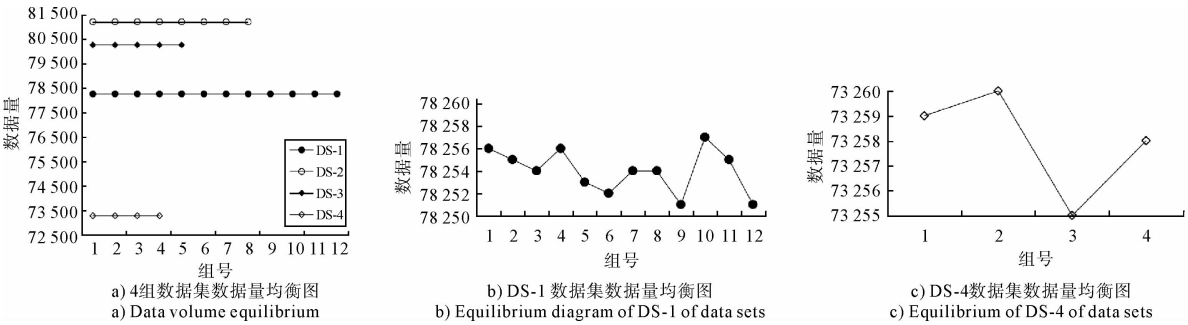


图 3 计算效率与数据量的变化趋势

Fig. 3 Change trend of computational efficiency and data volume



a) 4组数据集数据量均衡图 a) Data volume equilibrium

b) DS-1 数据集数据量均衡图 b) Equilibrium diagram of DS-1 of data sets

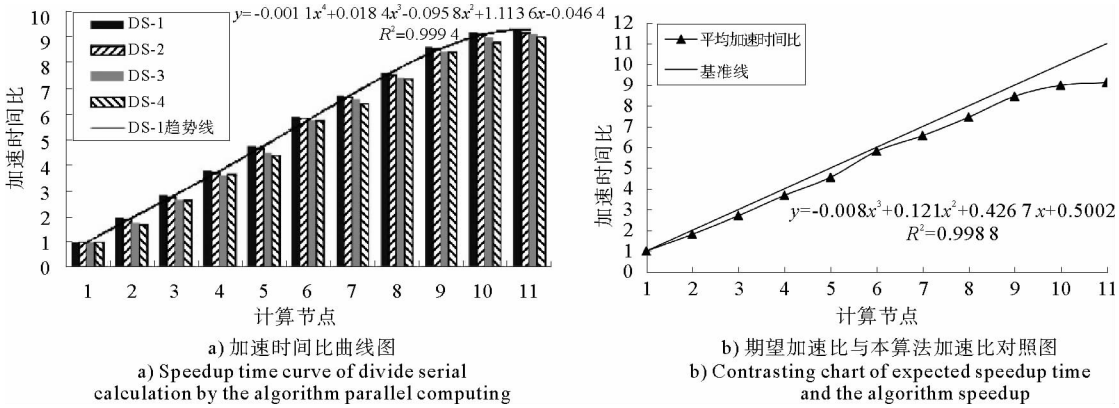
c) DS-4数据集数据量均衡图 c) Equilibrium of DS-4 of data sets

图 4 计算量均衡图

Fig. 4 Comparison of data groups

2)分组算法的高效性分析

利用表 1 的数据集，采用相同的计算节点，对本算法与串行算法所需时间计算加速比，并取结果的平均值，不同计算节点的加速比如图 5 所示。



a) 加速时间比曲线图
a) Speedup time curve of divide serial calculation by the algorithm parallel computing

b) 期望加速比与本算法加速比对照图
b) Contrasting chart of expected speedup time and the algorithm speedup

图 5　加速时间比曲线

Fig. 5　Speedup time curve

图 5(a)列出了 4 个数据集在不同计算节点参与计算时，串行计算时间与通过分组后的数据处理时间相比得到的加速时间比。在 5 个计算节点参与计算时，DS-1 到 DS-4，对应的加速比分别为 4.72，4.64，4.51，4.34，由此可得，计算节点一定的情况下，随着数据量的增加，加速时间比不断增大。因此，大数据量的情况下，该算法具有较好的加速时间比。当数据量一定的条件下，随着计算节点的增加，加速时间比不断增大，例如 DS-1 数据集，但增大的趋势并不是线性的，即随着计算节点的增大，增长率变小，其原因是当数据分组个数变大时，数据均衡的效果会变差(同等粒度分元素调整，分组个数越多，差异越大)，造成加速时间比下降。

图 5(b)显示了 4 组数据在同等计算节点的前提下，所得加速时间比的平均值。其中，基准线代表不同的计算节点理想状态下的加速时间比(单处理器上最优串行化算法计算时间与使用 n 个处理器并行计算时间比值)，例如，计算节点为 6 时，理想状况加速比为 6，实际加速比为 5.81，基本上得到了接近线性的加速效果，证明表层浮标轨迹验证涡旋的实验中，利用数据分组后得到了较好的加速效果。

3　结论

针对不可分割的空间线对象大数据处理效率低的问题，提出了一种基于数据均衡的分组方法。通过粗略调整，将线对象进行分组，然后将不可分割的分组对象作两次调整，使分配到每个小组内的数据量达到均衡，为每个计算节点并行处理数据提供了条件。实验表明，基于负载均衡的分治法进行分组是对不可分割空间线的计算进行优化的有效途径，此外，该算法也适用于不可分割的点数据集合的处理。

参考文献：

[1]Jonk A，Smeulders A W. An axiomatic approach to clustering line-segments[C]//The 3rd International Conference on Document Analysis and Recognition. Montreal，Quebec，Aug. 14-16，1995：386-389.

[2]Jang J H，Hong K S. Fast line segment grouping method for finding globally more favorable line segments[J]. Pattern Recognition，2002，35(10)：2235-2247.

[3]Kelly A R，Hancock E R. Grouping-line segments using eigenclustering[C]//The 11th British Machine Vision Conference. Bristol，Sep. 11-14，2000：58659-5.

[4]Thomas J C R. A new clustering algorithm based on k-means using a line segment as prototype[C]//Progress in Pattern Recognition，Image Analysis，Computer Vision，and Applications. Berlin Heidelberg：Springer，2011：638-645.

[5]柳盛，吉根林，李文俊. 一种基于连接度的空间线对象聚类算法[J]. 计算机科学，2011，38(8)：179-181.
Liu Sheng，Ji Genlin，Li Wenjun. Spatial lines clustering algorithm based on connectivity[J]. Computer Science，2011，38(8)：179-181.

[6]Abugov D. Oracle spatial partitioning：Best practices：An Oracle white paper[M]. Redwood：Oracle CorporationWorld，2004：1-16.

[7]周芹，钟耳顺，黄耀欢. 基于分区技术的静态 R 树索引并行计算技术[J]. 计算机工程，2009，35(2)：68-69.
Zhou Qin，Zhong Ershun，Huang Yaohuan. Parallel computation technique for static R_tree index based on partition techno-

logy[J]. Computer Engineering,2009,35(2):68-69.

[8]Guttman A. R-trees: A dynamic index structure for spatial searching[C]//The International Conference on Management of Data. New York:ACM Press,1984:47-57.

[9]HamiltonP. Eddy statistics from Lagrangian drifters and hydrography for the northern Gulf of Mexico slope[J]. Journal of Geophysical Research:Oceans (1978-2012),2007,112:C09002.

[10]Beron-Vera F,Olascoaga M,Goni G. Oceanic mesoscale eddies as revealed by Lagrangian coherent structures[J]. Geophysical Research Letters,2008,35(12):L12603.

[11]Dong C M,Liu Y,Lumpkin R,et al. A scheme to identify loops from trajectories of oceanic surface drifters:An application in the Kuroshio extension region[J]. Journal of Atmospheric and Oceanic Technology,2011,28(9):1167-1176.

[12]Guilherme A,Gabriel R,Daniel M,et al. G-DBSCAN:A GPU accelerated algorithm for density-based clustering[J]. Procedia Computer Science,2013,18:369-378.

[13]Lee J G,Han J,Whang K Y. Trajectory clustering:A partition-and-group framework[C]//The 2007 ACM SIGMOD International Conference on Management of Data. New York:ACM Press,2007:593-604.

[14]Nencioli F,Dong C,Dickey T,et al. A vector geometry-based eddy detection algorithm and its application to a high-resolution numerical model product and high-frequency radar surface velocities in the Southern California Bight[J]. Journal of Atmospheric and Oceanic Technology,2010,27(3):564-579.

(责任编辑:吕文红)

“矿山物联网技术”研究专栏征稿

征稿范围:

◇矿山物联网架构
◇矿用传感器技术及网络
◇井下人员定位技术
◇矿山 GIS
◇矿山监测监控技术
◇矿山通信网络
◇矿山数据仓库
◇矿山应急指挥系统
◇数字矿山理论与技术
◇矿山虚拟现实技术

欢迎相关领域专家学者和工程技术人员踊跃投稿,来稿请注明“矿山物联网技术”研究专栏。稿件通过专家评审后优先发表。

投稿平台:http://xuebao.sdust.edu.cn/index_z.asp

电子邮箱:xbgjcl@126.com

联系电话:0532-86057826

山东科技大学学报(自然科学版) 编辑部