

高分辨率全球非结构网格生成实践*

刘 壮, 黄小猛

(清华大学 地球系统科学系, 北京 100084)

摘要: 球面质心 Voronoi 网格(SCVT)是一种高质量的非结构网格,被国内外知名地球系统模式所采用。国际上,已有开源项目 Message Passing Interface SCVT(MPI-SCVT)实现了并行的 SCVT 生成。然而,尽管 MPI-SCVT 可以很好地适用于一般分辨率网格的生成,当所需网格的分辨率非常高时,MPI-SCVT 的使用会出现多个问题,包括初始网格点质量差、大数据 MPI 通信、外部库调用以及 I/O 问题。对于这些问题,提出了一整套解决方案。实践表明,应用此方案可以顺利完成全球 1.9 km 分辨率网格的生成,且到达相同精度时所需迭代步数显著减少,节约了计算时间。

关键词: 非结构网格生成;质心 Voronoi 网格;高分辨率;迭代优化

中图分类号: TP311

文献标识码: A

DOI: 10.16157/j.issn.0258-7998.222602

中文引用格式: 刘壮,黄小猛. 高分辨率全球非结构网格生成实践[J]. 电子技术应用, 2022, 48(6): 103-106, 111.

英文引用格式: Liu Zhuang, Huang Xiaomeng. Generating high-resolution global unstructured meshes[J]. Application of Electronic Technique, 2022, 48(6): 103-106, 111.

Generating high-resolution global unstructured meshes

Liu Zhuang, Huang Xiaomeng

(Department of Earth System Science, Tsinghua University, Beijing 100084, China)

Abstract: The Spherical Centroid Voronoi tessellations(SCVT) are a kind of high-quality unstructured meshes, which are adopted by well-known earth system models at home and abroad. Internationally, the open source project Message Passing Interface SCVT (MPI-SCVT) has realized the parallel SCVT generation. However, although MPI-SCVT works well for the generation of general-resolution meshes, the use of MPI-SCVT presents several problems when the required mesh is of very high resolution, including the poor quality of initial mesh points, large data MPI communication problems, external library calling problems and I/O problems. For these problems, a set of solutions are proposed. Practice shows that the application of these solutions can successfully complete the generation of the global 1.9 km resolution grid, and the number of iterations is significantly reduced when reaching the same accuracy, which saves the calculation time.

Key words: generation of unstructured meshes; centroid Voronoi tessellations; high resolution; iteration optimization

0 引言

近年来,随着计算机计算能力的提升与数值方法的发展,非结构网格在地球系统模式中得到了越来越多的应用^[1-2]。不同于结构网格,非结构网格没有极点附近网格收缩而出现奇异的问题;同时,非结构网格可以更灵活地处理各种边界条件、实现局部网格加密等。

在各种非结构网格中,球面质心 Voronoi 网格^[3](Spherical Centroid Voronoi Tessellations, SCVT)因具备良好的性质而得到广泛应用。生成 SCVT 一般采用的 Lloyd 算法^[3]是一个不动点迭代算法,其计算量与生成的网格点数成正比。为应对快速生成高分辨率 SCVT 的挑战, Jacobsen 等^[4]提出了基于区域分解和球极投影的并行算法,并在 GitHub 开源了其代码(MPI-SCVT; [https://github.com/dou-](https://github.com/douglasjacobsen/MPI-SCVT)

[glasjacobsen/MPI-SCVT](https://github.com/douglasjacobsen/MPI-SCVT))。数值实验显示了 MPI-SCVT 相对于串行算法的加速效果。

尽管 MPI-SCVT 可以很好地适用于一般分辨率下 SCVT 的生成,当所需网格的分辨率非常高(例如 2 km 以下)时, MPI-SCVT 的使用会出现一系列问题。首先, MPI-SCVT 内置的几种网格初始化方法生成的网格点质量较差,与最终的收敛结果距离较远,这使得迭代步数过多,程序整体运行时间长。其次,高分辨率导致迭代过程中通信的数据量大, MPI-SCVT 使用的 boost MPI(<http://www.boost.org>) 单次通信数据量超过一定大小(约 2 GB)时会出错。再者, MPI-SCVT 调用外部库 Triangle^[5]实现平面 Delaunay 三角网格^[4]的构建,当单次输入网格点数超过约 42 600 000 时, Triangle 无法正常运行。最后,高分辨率导致输出变量较大,使用外部库 NetCDF-CXX 无法完成所需 NetCDF 文件的输出。

* 基金项目: 国家重点研发计划(2017YFC1502203)

本文针对以上问题,提出一套整体解决方案。首先,通过采用正二十面体二分加密的方法,提高初始网格质量,减少收敛所需的迭代步数。其次,通过数据分段、多次通信,实现迭代过程中大数据的传输。再者,通过增加分片处理区域的数量,减少每次调用 Triangle 时输入的网格点数,顺利完成 Delaunay 三角网格的构建。最后,通过更换 Paralell NetCDF 库(后文简称 PnetCDF),将大变量数据分块,多次调用输出接口进行输出,消除了原来单变量不能超过 4 GB 大小的限制。

1 SCVT 生成方法

这里给出 SCVT 的基本定义及生成算法,其详细介绍可以参考文献[3-6]。

1.1 网格定义

首先介绍 Voronoi 单元。记 $d(x, y)$ 为球面 Ω 上两点 x 和 y 的测地线距离,给定球面上的一组点 $\{z_i\}_{i=1}^k$, 那么 $V_i = \{x \in \Omega: d(x, z_i) < d(x, z_j) \text{ for } j=1, \dots, k, j \neq i\}$ 称为点 z_i 对应的 Voronoi 单元。

下面介绍 Delaunay 三角剖分。同样,记 $\{z_i\}_{i=1}^k, k \geq 3$ 为球面不在同一个大圆上的一组点。这组点的三角剖分的定义是满足如下条件的一组三角形:

- (1) 三角形的顶点都是 $\{z_i\}_{i=1}^k$ 中的点;
- (2) 三角形内部都不包含其他顶点;
- (3) 三角形两两不相交;

(4) 所有三角形的并集组成点集 $\{z_i\}_{i=1}^k$ 的凸包(凸包指包含 $\{z_i\}_{i=1}^k$ 的最小凸集,凸集即满足集合内任意两点的测地线连线都包含在此集合内的集合)。

Delaunay 三角剖分是满足所有三角形的外接圆内部都不包含其他顶点的三角剖分。

下面给出 Voronoi 单元与 Delaunay 三角形的关系,其详细证明可以参见文献[6]。Voronoi 单元与 Delaunay 三角形互为对偶:Voronoi 单元的边是 Delaunay 三角形边的中垂线,Voronoi 单元的顶点是 Delaunay 三角形的外心。因此,给定 Delaunay 三角剖分,即可构建出 Voronoi 网格剖分。图 1 给出了 Delaunay 三角剖分及其对应的 Voronoi 剖分示意图。

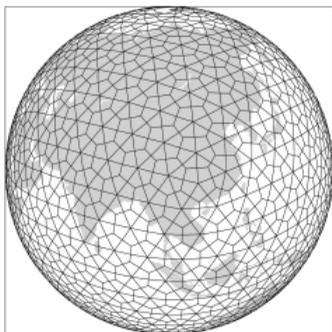


图 1 Delaunay 三角剖分和 Voronoi 剖分

最后,给出 SCVT 网格定义。给定 Ω 上的密度函数 $\rho(x)$,Voronoi 单元 V_i 的质心定义为:

$$z_i^* = \text{Proj}_{\Omega} \left[\frac{\int_{V_i} x \rho(x) dx}{\int_{V_i} \rho(x) dx} \right]$$

式中 Proj_{Ω} 表示通过乘以一个常数因子将点投影到球面 Ω 上。如果 Ω 为单位球面,那么:

$$\begin{cases} \text{Proj}_{\Omega}(x_1, x_2, x_3) = \frac{1}{r}(x_1, x_2, x_3) \\ r = x_1^2 + x_2^2 + x_3^2 \end{cases}$$

不难看出,当 $\rho(x) = 1$ 时, z_i^* 为单元 V_i 的几何中心。

当所有网格点满足 $z_i = z_i^*$ 时,即单元 V_i 的生成点 z_i 与其质心 z_i^* 重合时,此网格称为 SCVT。

1.2 Lloyd 算法及并行化

生成 SCVT 一般采用 Lloyd 算法,如算法 1^[3]所述。

算法 1 生成单位球面 SCVT 的 Lloyd 算法

输入:密度函数 $\rho(x)$,网格点数 k 。

输出:网格点 $\{z_i\}_{i=1}^k$ 。

- (1) 通过某种方法生成初始网格点 $\{z_i\}_{i=1}^k$;
- (2) 构建 $\{z_i\}_{i=1}^k$ 对应的 Voronoi 单元 $\{V_i\}_{i=1}^k$;
- (3) 在 Voronoi 单元 $\{V_i\}_{i=1}^k$ 上积分,计算其以 $\rho(x)$ 为密度函数的质心 $\{z_i\}_{i=1}^k$,将计算之后的质心赋值给 $\{z_i\}_{i=1}^k$;

(4) 收敛判断:如果第(3)步迭代前后 $\{z_i\}_{i=1}^k$ 变化较小,达到收敛条件,则算法停止,否则返回第(2)步。

Jacobsen 等^[4]提出了基于区域分解和球极投影的并行算法,称为 MPI-SCVT,如算法 2 所述。

算法 2 MPI-SCVT 并行算法

输入:密度函数 $\rho(x)$,网格点数 k ,MPI 进程数 N 。

输出:网格点 $\{z_i\}_{i=1}^k$ 。

(1) 在球面均匀洒 N 个点,分别以这 N 个点为圆心、以圆心到其邻居点(与其同属于一个 Delaunay 三角形的点)的距离的最大值为半径做圆,得到 N 个球面区域,将其分配给 N 个进程。

(2) 通过某种方法生成初始网格点 $\{z_i\}_{i=1}^k$,并将其发送给所有进程。

(3) 各进程扫描落在本区域内的网格点,之后以本区域中心的球心对称点为极点、以此区域中心处的切平面为投影平面,将本区域内的网格点投影到切平面上(如图 2 所示)。这样,区域内每个球面网格点即可对应于一个平面网格点。然后调用 Triangle 生成平面 Delaunay 三角网格。

(4) 各进程根据平面 Delaunay 三角网格生成球面 Delaunay 三角网格,利用球面 Delaunay 三角网格计算 Voronoi

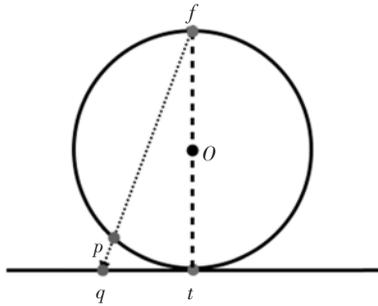


图2 球极投影示意图(t 为区域中心, f 为其球心对称点, p 为区域内一点, q 为 p 的球极投影)

单元积分。计算方法是利用三角形外心与三边中点的连线将其分为三个子部分,之后计算各子部分的积分,将其累加到对应顶点的 Voronoi 单元积分,如图3所示。为避免重复计算,每个进程只计算最靠近中心点为本进程负责区域中心点的顶点的 Voronoi 单元积分,如果某个顶点到两个区域中心距离相等,那么选择进程号小的进程计算。

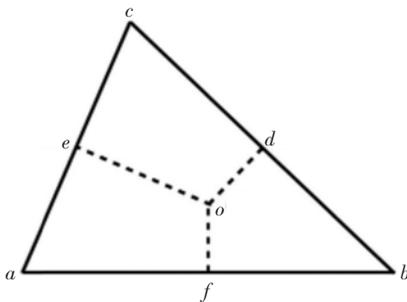


图3 Voronoi 单元积分计算方法

(5)各进程用其计算的 Voronoi 单元积分值 $\{z_i^*\}_{i=1}^k$ 替换 $\{z_i\}_{i=1}^k$,并将其更新后的网格点发送给其他进程,计算网格点的变化量。

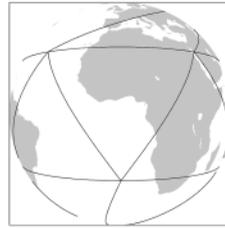
(6)收敛判断:如果第(5)步计算得到的 $\{z_i^*\}_{i=1}^k$ 与 $\{z_i\}_{i=1}^k$ 差别较小,达到收敛条件,则算法停止;否则,返回第(3)步。

2 高分辨率网格生成解决方案

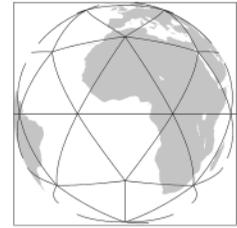
2.1 初始点集生成

如算法1和算法2所述,Lloyd算法是一个不动点迭代算法,因此其收敛速度与初始点集密切相关。MPI-SCVT 内置了三种网格点初始化方法:Monte Carlo法、Generalized Spiral法和Fibonacci法。然而,几种方法生成的初始网格质量并不好,与最终收敛结果距离较远。

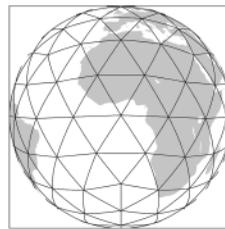
在此,采用球面正二十面体二分加密的方法,如图4所示。首先给定球面正二十面体网格点,形成球面 Delaunay 三角网格(G0网格),然后连接各三角形三边中点,将原来一个三角形一分为四,得到右上网格(G1网格)。接着将右上网格的每个三角形用同样的方法一分为四,得到左下网格(G2网格),依此进行下去,即可逐步得到高分



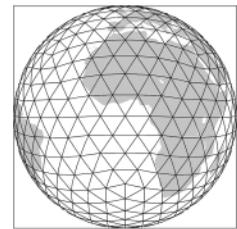
(a)球面正二十面体网格(G0网格)



(b)正二十面体二分加密网格(G1网格)



(c)右上网格二分加密网格(G2网格)



(d)左下网格二分加密网格(G3网格)

图4 球面正二十面体二分加密方法

辨率网格。

表1中展示了使用几种初始化方法后达到相同收敛精度所需的迭代步数。此处判定收敛的变量为相邻两次迭代网格点变化量的 l_1 范数的平均值,收敛参数分别为 1×10^{-8} 、 1×10^{-9} 和 1×10^{-10} 。可以看到,对于不同分辨率的网格,二十面体二分加密法(Icosahedron)所需的迭代步数都远少于其他几种方法。值得注意的是,随着收敛准则变严格(即收敛参数变小),其他几种初始化方法不能收敛的情况变得更多。当网格分辨率较高时,只有二十面体二分加密法可以收敛。此测试结果表明使用二十面体二分加密的初始化方法可以获得更好的初始网格。

表1 相邻两次迭代格点变化量 l_1 范数达到收敛条件所需的迭代步数

Convergence Criterion	Grid	Monte Carlo	Generalized Spiral	Fibonacci	Icosahedron
1×10^{-8}	G4	11 672	7 580	10 496	228
	G5	55 680	22 411	fail	780
	G6	42 091	45 753	48 933	2 581
	G7	fail	59 833	35 104	8 166
1×10^{-9}	G4	14 546	10 571	23 244	284
	G5	67 239	34 150	fail	1 004
	G6	84 798	99 914	fail	3 479
	G7	fail	fail	fail	11 757
1×10^{-10}	G4	17 485	13 588	26 497	340
	G5	78 888	45 910	fail	1 229
	G6	fail	fail	fail	4 377
	G7	fail	fail	fail	15 349

注:“fail”表示在100 000步迭代内没有达到收敛条件。

2.2 解决大变量问题

2.2.1 MPI 通信问题

MPI-SCVT 初始化及迭代过程中需要使用 MPI

broadcast 进行数据广播,当数据大小超过一定量(约 2 GB)时会出错。因此,对 broadcast 代码进行修改,设置传输数据数量上限。对于未超过上限的数据像原来一样进行一次通信,对于超过上限的数据,将数据分段,进行多次通信,即可解决此问题。

具体地,MPI-SCVT 使用 broadcast 广播的大变量是 points,其类型为 vector<pnt>,pnt 为自定义的 class 类型,单个实例所占用内存空间为 48 B。因此,设置传输数据数量上限参数 num_pts_limit_bcast 为 40 000 000。在通信前,主进程将全部网格点数 num_pts 广播给其他进程。之后在传输数据前进行判断,如果 num_pts<=num_pts_limit_bcast,那么直接进行一次通信;否则,将数据分段,进行多次通信。此处需要注意的是,对于非源进程,boost MPI 的 broadcast 函数对于未分配内存的 vector,可以根据源进程 vector 大小自动为其分配内存并使用接收到的数据赋值;对于已分配内存的 vecotor,将自动覆盖 vector 里面的内容。另一方面,对于多次通信的情况,因为每次通信一部分数据,所以需要指定 points 向量的起始位置,此时非源进程的 points 必须是已经分配过内存的。因此,在进行通信前为未分配内存的 points 分配内存,即可保持代码统一且简洁。

2.2.2 Triangle 调用问题

Triangle 是生成平面 Delaunay 三角网格的程序,当输入的网格点数超过约 42 600 000 时无法正常运行。为此,将球面分成更多的区域(即用更多进程运行),减少每个区域包含的网格点数量,即可解决此问题。

2.2.3 NetCDF 格式问题输出问题

大变量 NetCDF 格式文件输出存在同样的问题。MPI-SCVT 使用 NetCDF-CXX 进行输出,除了最后一个变量,其余变量大小均不能超过 4 GB。为支持更多大变量的输出,将 NetCDF-CXX 库替换为 PnetCDF 库,即可消除变量大小的限制。此处需要注意的是,因为 PnetCDF 单次输出数据大小限制在 2 GB 以内,所以对于超过 2 GB 大小的变量,需要将其分块,进行多次输出。

在进行分块输出时,对于高维数据,要确定在哪个维度方向进行划分。一般需要在与网格点数(或者三角形个数)相关的维度进行。如果此维度不是变化最慢的维度,那么需要分配一个临时 buffer 数组将每次要输出的数据拷贝到相应的位置上。这样做的原因是,PnetCDF 每次输出的数据必须是一段连续内存的数据,如果不进行拷贝,那么无法在对应文件的位置写入正确的数据。

现以二维变量分块输出为例进行说明。如图 5 所示,假设 num_pts_limit_bcast=10,要输出的二维数据大小为 2×10,那么需要将大小为 10 的维度划分,分两次进行输出,即每次输出 2×5 的数据块。图中给出了两次调用 PnetCDF 输出接口时输入的 start 和 count 数组。如果变量存储顺序符合 C 语言习惯,即一行一行顺序存储,那

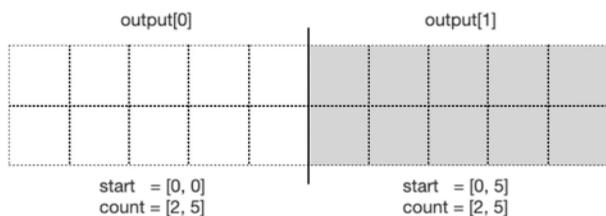


图 5 PnetCDF 二维数据分块输出示意图

么两次输出时,分别需要将数组的白色部分和灰色部分拷贝到一个连续数组,将此连续数组作为输入参数;如果变量符合 Fortran 语言习惯,即一列一列顺序存储,则不需要拷贝到临时数组,只需分别将两次输出的变量起始地址传递给 PnetCDF 输出接口。

3 结论

本文介绍了生成高分辨率 SCVT 的解决方案,重点包括初始点集生成方法以及应对大变量问题的几种方法。通过采用正二十面体二分加密的方法,显著减少了收敛所需的迭代步数;将大变量分块处理,解决了 MPI 通信、Triangle 调用以及 NetCDF 格式文件输出的问题。通过采用这一整套方案,可顺利生成全球 1.9 km 分辨率网格。

最后需要指出的是,本文介绍的正二十面体二分加密初始化方法应用领域为生成全球准均匀网格,即密度函数 $\rho(x) \equiv 1$ 的情况。对于变分辨率网格($\rho(x) \neq 1$),此方法生成的网格与最终迭代收敛的网格之间的差距并不会比其他几种方法小,因此不再有优势。对于 $\rho(x) \neq 1$ 时,如何生成相应的初始网格,使其与迭代收敛网格差距尽可能小,是下一步的研究方向。

参考文献

- [1] SKAMAROCK W C, KLEMP J B, DUDA M G, et al. A multiscale nonhydrostatic atmospheric model using centroidal Voronoi tessellations and C-Grid staggering[J]. Monthly Weather Review, 2012, 140(9): 3090-3105.
- [2] ZHANG Y, LI J, YU R C, et al. A layer-averaged nonhydrostatic dynamical framework on an unstructured mesh for global and regional atmospheric modeling: Model description, baseline evaluation and sensitivity exploration[J]. Journal of Advances in Modeling Earth Systems, 2019, 11: 1685-1714.
- [3] DU Q, GUNZBURGER M D, JU L L. Constrained centroidal voronoi tessellations for surfaces[J]. SIAM Journal on Scientific Computing, 2003, 24(5): 1488-1506.
- [4] JACOBSEN D W, GUNZBURGER M, RINGLER T, et al. Parallel algorithms for planar and spherical Delaunay construction with an application to centroidal Voronoi tessellations[J]. Geoscientific Model Development, 2013, 6: 1353-1365.
- [5] SHEWCHUK J R. Triangle: Engineering a 2D quality mesh generator and delaunay triangulator[M]// Applied Computa-

(下转第 111 页)

- [9] 王豪, 纪少波, 刘振革, 等. 高速行驶车辆位置信息快速检测系统设计[J]. 测控技术, 2019, 38(8): 84-88.
- [10] 沈健. 一种基于 PIC18F45 芯片车辆检测系统设计[J]. 电子技术, 2018, 47(10): 76-77, 69.

(收稿日期: 2021-09-18)

作者简介:

周亚(1990-), 男, 硕士, 工程师, 主要研究方向: 硬件电

路设计。

戴伟(1990-), 通信作者, 男, 硕士, 工程师, 主要研究方向: 物联网应用, E-mail: daiwei@cetc-iot.com。

张鑫(1994-), 男, 硕士, 工程师, 主要研究方向: 硬件电路设计。



扫码下载电子文档

(上接第 102 页)

- 派的内核系统启动[J]. 现代信息科技, 2018, 2(12): 29-34.
- [2] 任礼虎. 一种基于龙芯 2K1000 的 COM-E 功能主板产品设计方法[J]. 电子技术, 2018, 47(4): 46-47.
- [3] 齐刚. 龙芯拟发布多款芯片——国产唯一支持多路互联的“中国芯”诞生[J]. 计算机与网络, 2017, 43(9): 24-25.
- [4] 钱家乐. 计算机操作系统的应用与发展分析[J]. 无线互联科技, 2015(19): 95-96.
- [5] 俞冠中, 王巍, 刘玉升. 基于状态机的 AI/AO 模块 HART 通信软件实现方法[J]. 测控技术, 2020, 39(11): 106-112.
- [6] 王巍, 俞冠中, 靳子洋. 基于 Cortex-M 的嵌入式设备固件更新方法研究[J]. 单片机与嵌入式系统应用, 2021, 21(6): 69-73.
- [7] 徐海亚, 赵增基, 朱波, 等. 基于中标麒麟的 POWERLINK 节点实时性解决方法[J]. 火力与指挥控制, 2014, 39(5): 164-167.
- [8] 刘剑, 仲宇, 王琦. 嵌入式 Linux 实时性改造技术综述[J]. 航天控制, 2018, 36(2): 93-97.
- [9] 赵旭, 夏靖波, 王哲. Linux 内核进程调度的研究与改进[J]. 测控技术, 2009, 28(Z1): 65-67.
- [10] 王朴, 周晴. 基于龙芯 1E 的嵌入式 Linux 实时性的优化与可靠性设计[J]. 微电子学与计算机, 2019, 36(11): 17-21.
- [11] 王娜, 李彦峰, 孙菲艳, 等. Linux 中断线程化分析及中断延时测试[J]. 智能计算机与应用, 2018, 8(6): 20-23, 27.

- [12] 袁辉建, 陈曾汉, 晏来成. 基于 ARM9 的嵌入式 Linux 测控系统实时性增强研究[J]. 测控技术, 2007(1): 61-63.
- [13] 刘宇帅, 苏宇, 王金波, 等. 航天嵌入式 Linux 实时性能优化研究[J]. 航天控制, 2018, 36(3): 57-62, 78.
- [14] 张晓龙, 郭锐锋, 陶耀东, 等. Linux 实时抢占补丁研究及实时性能测试[J]. 计算机工程, 2014, 40(10): 304-307.
- [15] 王帅华, 杨东升, 王允森, 等. 基于 ARM 的 Linux 实时抢占补丁的研究与实现[J]. 组合机床与自动化加工技术, 2015(9): 1-4.
- [16] 张帆, 求伟, 韩大鹏. Linux 在嵌入式实时系统的研究与改进[J]. 制造业自动化, 2011, 33(3): 87-89.

(收稿日期: 2021-12-11)

作者简介:

俞冠中(1989-), 男, 硕士, 工程师, 主要研究方向: 智能通信网络与信息处理、计算机网络与系统安全、云计算与物联网安全、操作系统及虚拟化技术、分布式计算机控制技术。

韦雄(1989-), 男, 硕士, 工程师, 主要研究方向: 智能通信网络与信息处理、计算机网络与系统安全、云计算与物联网安全、操作系统及虚拟化技术、分布式计算机控制技术。

史旭明(1972-), 通信作者, 男, 博士, 教授级高级工程师, 主要研究方向: 智能通信网络与信息处理、计算机网络与系统安全、云计算与物联网安全、操作系统及虚拟化技术、分布式计算机控制技术, E-mail: shixuming@snpas.com.cn。



扫码下载电子文档

(上接第 106 页)

- tional Geometry: Towards Geometric Engineering. Berlin: Springer, 1996: 741-778.
- [6] RENKA R. Algorithm 772: STRIPACK: Delaunay triangulation and Voronoi diagram on the surface of a sphere[J]. ACM Transactions on Mathematical Software, 1997, 23: 416-434.

(收稿日期: 2022-01-28)

作者简介:

刘壮(1988-), 男, 博士, 工程师, 主要研究方向: 高性能计算。

黄小猛(1980-), 通信作者, 男, 博士, 副教授, 主要研究方向: 海洋模式、并行计算与大数据, E-mail: hxm@tsinghua.edu.cn。



扫码下载电子文档

版权声明

经作者授权，本论文版权和信息网络传播权归属于《电子技术应用》杂志，凡未经本刊书面同意任何机构、组织和个人不得擅自复印、汇编、翻译和进行信息网络传播。未经本刊书面同意，禁止一切互联网论文资源平台非法上传、收录本论文。

截至目前，本论文已经授权被中国期刊全文数据库（CNKI）、万方数据知识服务平台、中文科技期刊数据库（维普网）、DOAJ、美国《乌利希期刊指南》、JST 日本科技技术振兴机构数据库等数据库全文收录。

对于违反上述禁止行为并违法使用本论文的机构、组织和个人，本刊将采取一切必要法律行动来维护正当权益。

特此声明！

《电子技术应用》编辑部

中国电子信息产业集团有限公司第六研究所