

基于 QEM 以方差作为权值的 3D 网格模型简化

侯宝明, 刘雪娜, 鄂 旭

(渤海大学 信息科学与技术学院, 辽宁 锦州 121013)

摘要: 为了提高 3D 网格模型的简化速度和简化效果, 采用半边结构表示模型, 在拓扑重建过程中使用附加的索引表使拓扑重建时间复杂度由 $O(n \log n)$ 降低为 $O(n)$, 在模型简化时, 通过计算顶点相邻三角面片法向量的方差来确定顶点的重要度, 并将其作为权值加入到 Garland 的二次误差测度公式 QEM 中, 起到了很好的简化效果。与经典的 Garland 算法相比采用半边结构能直接获得邻接关系, 加快了模型简化速度。在处理边界时对边界情况进行详细分类, 既提高了简化速度又保留了模型外观。

关键词: 网格简化; 半边结构; 方差; QEM

中图分类号: TP391.72

文献标识码: A

文章编号: 1673-629X(2013)11-0050-04

doi: 10.3969/j.issn.1673-629X.2013.11.013

Simplification of 3D Mesh Model Based on QEM with Variance as Weights

HOU Bao-ming, LIU Xue-na, E Xu

(College of Information Science and Technology, Bohai University, Jinzhou 121013, China)

Abstract: For enhancing simplification speed and effect of 3D mesh model, adopt half-edge structure to represent the model, and reduce the time complexity of the topology reconstruction from $O(n \log n)$ to $O(n)$ by using an additional index table. By calculating the normal variance of triangular facets connected to a vertex to determine the importance of the vertex, and add it to Quadric Error Metrics (QEM) as weights, the effect is satisfied. Compared with the classical Garland algorithm the use of half-edge structure can directly gain the adjacency relation, and accelerate the speed of model simplification. In the processing of model boundary the detailed classifications of the boundary situation not only enhances the simplification speed but also retains the appearance of model.

Key words: mesh simplification; half-edge structure; variance; QEM

0 引言

近些年来用三角形网格表示实体变得日益流行, 在计算机图形学、几何造型及处理、游戏及影视制作、医学图像等领域内都有着广泛的使用。用三角形表示的实体也称为三角网格模型, 三角网格模型一般通过 3D 扫描技术生成。随着扫描技术的进步、硬件系统性能的提升, 或基于实际需要, 表示实体的三角片个数越来越大, 当然所表示的模型精度也越来越高。高精度模型对应的数据量必然很大, 对系统处理能力要求很高。现实中由于模型的使用领域不同, 需要同时兼顾处理速度、传输速度、渲染速度等, 降低模型的精度(分辨率)就显得非常必要。这就需要对原始的高分辨率模型进行简化, 通过交互式控制, 生成分辨率不同

的一系列网格模型。另外, 在某种约束条件下对模型进行简化也属于网格模型拓扑优化方面的内容。扫描生成的模型一般通过数据文件形式来存储, 一般的模型系统读取表示模型的数据文件, 重建模型, 然后对模型中的缺陷(拓扑缺陷、几何缺陷)进行检查、修复, 生成合格的二维流型网格以便进行处理。后续处理包括网格平滑、网格简化、网格参数化及网格重绘、网格变形等。

经过多年研究, 不同学者已经提出各具特色的多种网格简化方法, 对这些方法的分类方式不尽相同, 文献[1]对这一问题做了比较详尽的讨论。简化方法的设计需要考虑三个因素: 模型外观形状的保持、模型的简化速度以及简化算法的鲁棒性。Hoppe^[2]的基于能

收稿日期: 2013-02-22

修回日期: 2013-05-26

网络出版时间: 2013-08-28

基金项目: 辽宁省高等学校重点实验室项目(L2012397); 辽宁省教育一般研究项目(L2012396)

作者简介: 侯宝明(1976-), 男, 硕士, 讲师, 研究方向为几何造型。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20130827.1430.008.html>

量函数的边收缩算法简化效果很好,但是比较慢。Rossignac-Borrel^[3]的顶点聚类算法简化速度快,能适应任意三角片集合,但效果不好。Reinhard Klein 等^[4-5]的顶点删除算法能产生好的效果,但也是比较慢。Garland^[6]于 1997 年提出基于 QEM 的顶点对收缩(vertex pair contraction)算法在速度和效果之间找到了平衡,此后许多学者^[7-10]的算法思想都来自于 Garland 算法。

采用半边数据结构表示模型,在拓扑关系重建时采用文献[11]的快速算法在内存中构建出网格模型。设计一种把顶点相邻三角面片法向的方差作为权值,加入到 QEM 中形成新的误差测度公式,并采用顶点对收缩方法对网格进行简化,在边界处理时通过对边界情况的详细分类,既提高了速度又保留了模型外观。

1 网格模型的拓扑关系重建

文中采用半边数据结构描述模型的拓扑关系,与 Garland QSlim 算法的数据结构相比半边数据结构表示的拓扑信息更为完整,尤其是在进行邻接信息定位时无需遍历,只需简单的直接定位即可,效率更高。网格模型一般以数据文件的形式存储,文中采用 SMF 格式文件,在打开 SMF 文件之后首先建立一个动态的顶点结构数组,然后把文件中的顶点坐标数据依次读入到对应的顶点结构中。

从文件中读取三角面片时每建立一个三角面片节点都要检查对应的三个顶点是否已经存在,利用数组可以随机定位的特点,将读入的顶点编号作为顶点结构数组的下标直接定位到对应顶点的位置,查看是否为空。不为空说明该顶点节点已经存在,直接取出数据进行后续拓扑关系设置,为空说明该顶点节点尚未创建,创建新的顶点节点并把节点地址回送到顶点结构数组对应数组元素的节点地址域中^[11]。

2 Garland 的网格简化算法

2.1 Garland 算法的主要思想

Garland 使用迭代顶点对收缩的方法实现网格简化,其主要思想如下:

选定一个候选顶点对的集合,对于没有分离部分的模型而言,顶点对就是边(Edge);为每一个候选边指定收缩代价;按照收缩代价将所有的候选边放到堆上,拥有最小收缩代价的边放到堆顶。重复下面的步骤,直到目标模型满足要求:

(1) 从堆上移除具有最小收缩代价的边(V_i, V_j)。

(2) 收缩这个边,将对应的顶点(Vertex)、面(Face)和边(Edge)删除,见图 1。

(3) 更新所有与 V_i, V_j 相关联的边的收缩代价,然

后重新调整这些边在堆中的位置,以使堆顶放置的仍是收缩代价最小的边。

这个简化算法是一个贪心算法,每一步收缩产生的原模型的近似模型不一定是最优的,选取每一步收缩最优是一个 NP 难问题。

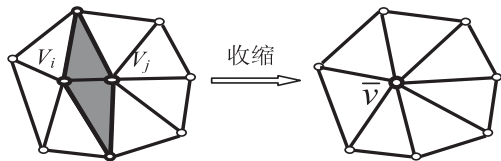


图 1 顶点对收缩示意图

大多数基于边收缩的算法采用上面的算法步骤,它们之间的差别主要是候选边收缩代价的计算方法不同。收缩代价描述了某一个边收缩后模型精度的损失程度,收缩代价越小说明收缩后的模型与原模型越接近。很显然,收缩代价的计算关系到边收缩的顺序问题,进而决定简化过程中模型外观形状的保持效果问题。

2.2 Garland 算法中的 QEM

2.2.1 QEM 的由来

一个点集表示平面的向量方程为 $\mathbf{n}^T \mathbf{v} + d = 0$, 其中 $\mathbf{n} = [abc]^T$, 单位向量, 且 $a^2 + b^2 + c^2 = 1$, d 是一个标量, 顶点 $\mathbf{v} = [xyz]^T$ 到该平面距离的平方可表示成:

$$D^2(\mathbf{v}) = (\mathbf{n}^T \mathbf{v} + d)^2 = (ax + by + cz + d)^2 \quad (1)$$

也可表示成:

$$D^2(\mathbf{v}) = (\mathbf{v}^T \mathbf{n} + d)(\mathbf{n}^T \mathbf{v} + d) = (\mathbf{v}^T (\mathbf{n} \mathbf{n}^T) \mathbf{v} + 2(d \mathbf{n}^T) \mathbf{v} + d^2) \quad (2)$$

将 Q 定义成一个三元组: $\mathbf{Q} = (\mathbf{A}, \mathbf{b}, c)$, 并将其指定到顶点 \mathbf{v} 上, 与公式(2)相联系得到:

$$\mathbf{Q} = D^2(\mathbf{v}) = \mathbf{v}^T \mathbf{A} \mathbf{v} + 2\mathbf{b}^T \mathbf{v} + c \quad (3)$$

其中, \mathbf{A} 表示 $\mathbf{n} \mathbf{n}^T$, \mathbf{b} 表示 $d \mathbf{n}$, c 表示 d^2 。模型中的顶点 \mathbf{v} 一般与多个三角平面相邻, 所以 Q 应当写成和的形式, 进而可以定义一个二次的误差测度公式:

$$E_Q(\mathbf{v}) = \sum_i D_i^2(\mathbf{v}) = \sum_i Q_i(\mathbf{v}) = Q(\mathbf{v}) \quad (4)$$

若用 $\bar{\mathbf{v}}$ 表示收缩后的顶点, 那么 $Q(\bar{\mathbf{v}})$ 用来描述模型收缩前后的偏差程度, 由于 Q 是二次的, 所以称 $Q(\bar{\mathbf{v}})$ 为二次误差测度, 即 QEM (Quadric Error Metrics), 由公式(3)可知, $Q(\mathbf{v})$ 可以表示成更紧凑的形式:

$$Q(\mathbf{v}) = \bar{\mathbf{v}}^T \mathbf{Q} \bar{\mathbf{v}} = \bar{\mathbf{v}}^T \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{b}^T & c \end{bmatrix} \bar{\mathbf{v}} = \bar{\mathbf{v}}^T \begin{bmatrix} a^2 & ab & ac & ad \\ ab & b^2 & bc & bd \\ ac & bc & c^2 & cd \\ ad & bd & cd & d^2 \end{bmatrix} \bar{\mathbf{v}} \quad (5)$$

其中, $\bar{\mathbf{v}} = [\mathbf{v} \ 1]^T = [x \ y \ z \ 1]^T$, 是 \mathbf{v} 的齐次坐标。

由于 $Q = \sum_i Q_i$, 即 $Q_i(v) + Q_j(v) = (Q_i + Q_j)(v) = (A_i + A_j, b_i + b_j, c_i + c_j)(v)$, 说明计算顶点 v 到相邻三角平面距离的平方和只需将相关平面的 Q 值相加即可, 当要实现边收缩 $(V_i, V_j) \rightarrow \bar{v}$ 时只需计算 $Q(\bar{v}) = Q_i(\bar{v}) + Q_j(\bar{v})^{[12]}$ 。当顶点 v 位于相邻三角平面之上时计算出的 Q 值是 0, 在进行边收缩时, v 就变成了目标顶点 \bar{v} , 一般的, \bar{v} 并不在相邻三角平面上, 所以 $Q(\bar{v})$ 的值不为 0。

2.2.2 目标顶点 \bar{v} 的选择

找到一个 \bar{v} 使得 $Q(\bar{v})$ 最小, 即 (V_i, V_j) 收缩到 \bar{v} 之后对模型产生的误差是最小的。既然 Q 是二次的, 寻找它的最小值就是一个线性问题, Q 的斜率为 $2Av + 2b$, 让其等于 0, 从而得到 $\bar{v} = -A^{-1}b$, 当 A 的逆阵存在时, 最优位置 \bar{v} 是可求的, 当 A 的逆阵不存在时, Garland 采用折中的办法, 首先沿着直线段 (V_i, V_j) 寻找最优位置, 如果这个值不唯一, 那么直接选择 V_i 或 V_j 中的一个。通过计算矩阵逆的方法求 \bar{v} 比较费时, 但收缩效果非常好, 如果不通过计算直接替换效果就要差一些。

3 文中简化算法描述

3.1 加权误差测度公式

基本的二次误差测度中的 Q 是一个和值, 是将所有三角平面的 Q 值求和, 这些三角平面具有相同的权值, 这在实际简化过程中并不合适。比如: 当一些三角面片非常大而另一些非常小时采用相同权值来计算 Q , 收缩将产生比较糟糕的结果。为此 Garland 将三角形面积作为权值加入到误差测度公式中, 形成的误差测度公式为:

$$Q(v) = \sum_i w_i Q_i(v) = (\sum_i w_i Q_i)(v) \quad (6)$$

对保持模型外形有一定效果, 但三角形面积并不能反映模型局部的变化程度, 这使得简化后的模型表面迅速变得比较平坦。为了解决这个问题, 国内不少学者提出自己的解决方案, 陈伟海^[7]等通过把顶点的相对曲率值作为权值加入到误差测度公式中。杜晓晖^[8]等通过对三角形进行分类并分别计算它们的重要度并加入到公式中都取得了不错的效果。

3.2 法向量的方差作为权值

文中通过计算顶点相邻三角面片法向的变化程度来确定该顶点的重要度, 并且把顶点的重要程度作为权值放入误差测度公式中。首先计算顶点 v 所有相邻的三角面片法向量的平均值 \bar{n} (每个三角面片的法向量在网格模型的拓扑重建过程中已计算过, 当然一个边收缩之后受影响的三角面片的法向需要重新计算), 然后根据公式(7)计算法向量的方差。

$$S = \frac{1}{N} [(n_1^2 + n_2^2 + \dots + n_N^2) - N \bar{n}^2] \quad (7)$$

公式(7)是方差的简化公式, N 表示与顶点 v 相邻三角面片的个数。由于在半边模型结构中通过顶点可以直接找到相邻三角面片, 无需查找遍历, 所以方差 S 的计算并不复杂。

文中的误差测度公式为:

$$Q(v) = (\sum_i S_i Q_i)(v) \quad (8)$$

当顶点 v 相邻平面的法向量变化较小, 计算的方差值也较小, 法向量变化较大计算的方差值也较大, 所以很好地反映了模型表面的变化程度。与 Garland 算法相比, 新的误差测度公式强化了模型表面变化剧烈程度对误差测度值的影响, 在三角面片法向量变化较大的区域将会产生较大的误差测度值。由于边收缩过程是按误差测度值由小到大的顺序来进行的, 所以区域变化比较剧烈的部分将延后进行简化, 也就是说模型轮廓清晰的部分被延后简化, 因而起到了很好的简化效果。相比于文献[7-8]中的误差测度, 文中的公式更为简洁, 复杂度更低。

3.3 算法描述

3.3.1 网格简化预处理

(1) 从数据文件中读取 3D 模型的数据, 采用第一节的方法在内存中建立模型的拓扑结构。

(2) 在模型拓扑重建时每输入一个三角面片就计算三角面片所在平面方程的系数 a, b, c, d 。并根据系数计算 Q 值, 依次加入到对应三个顶点的数据结构中。

(3) 根据 3.2 节的描述为每个顶点计算方差。

(4) 遍历模型中所有的边:

a. 计算 $Q = Q_i + Q_j$ 。

b. 然后根据 2.2.2 节的描述计算收缩顶点 \bar{v} 的位置。

c. 以边的 $Q(\bar{v})$ 值为依据, 将它放到堆上, 并调整堆使堆顶的边的 Q 值最小。

3.3.2 进行边收缩操作

重复取出堆顶的边进行收缩操作直到简化目标已完成(以图 1 中的 V_i, V_j 边为例):

(1) 找到共享的两个三角面片和欲删除的三个边。

(2) 对于每个三角面片, 将它的 Q 值信息从三个顶点中删除。

(3) 用目标顶点 \bar{v} 的坐标值设置 V_i , 将以 V_j 为端点的边都合并到 V_i 上, 并对半边进行合并。

(4) 从边链表中删除三个相关的边, 从顶点链表中删除顶点 V_j , 从三角面片链表中删除两个三角面片。

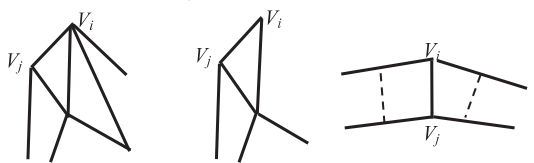
(5)对所有与收缩边相关联的三角面片做如下处理:重新计算平面法向量、 Q 值,并把新的 Q 值附加到三个顶点上,同时找出所有受收缩影响的边形成一个“受影响边链表”。

(6)对所有受影响的顶点计算新的方差。
(7)重新计算所有相关边的 Q 值。
(8)堆中“受影响边链表”中的边的 Q 值已发生变化,重新调整堆。

3.3.3 网格边界处理

如果对模型的网格边界上的边采用普通边的收缩算法,算法就会失败。必须对边界进行单独处理,模型的边界情况比较复杂,具体可分成两大类:

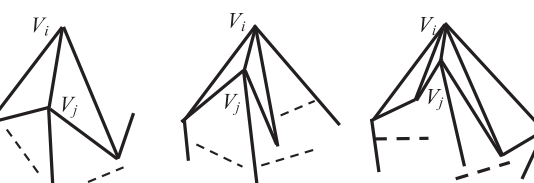
第一类, V_i 和 V_j 都是边界上的点,见图 2。



a) 普通边界边 b) 边界边在悬面上 c) 穿透边

图 2 第一类边界情况示意图

第二类, V_i 和 V_j 只有一个边界上的点。经过简单处理始终让 V_i 是网格边界上的点, V_j 是网格内部的点,见图 3。



a) 双边界 b) 单边界 c) 普通情况

图 3 第二类边界情况示意图

在半边数据结构中,一个边(Edge)应当对应两个半边(Half-Edge),如果某一条 Edge 只对应一个半边,则可以判定该边是一个边界边。具体地根据 V_i 和 V_j 的位置不同又分成图 2 和图 3 的几种情况,在进行边收缩操作时所执行的代码也都各不相同,限于篇幅这里不一一说明。

在实际简化算法执行过程中,为了尽量保持模型的边界外形,一般都将边界边赋予较高的 Q 值使其滞后收缩。

4 实验验证

在一台普通的 Pentium 4(2.80 GHz CPU,512 M 内存,Windows XP 系统)的 PC 上对文中算法和 Garland QSlim 算法进行了实现,测试了几个 SMF 文件格式的实例。表 1 中列出了 Garland QSlim 算法和文中算法的拓扑重建时间。

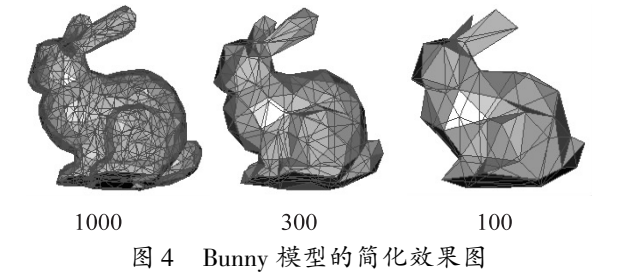
从表 1 可以看出,与 Garland QSlim 算法相比文中

算法的拓扑重建速度稍快一些,说明相对于 QSlim 算法的数据结构而言虽然半边结构有些复杂,但由于索引表的使用大幅加快了重建过程。

表 1 模型拓扑重建耗时(second)

模型名称	三角面片数	Garland 算法	文中算法
Cow	5 804	0.045	0.051
阀体	23 470	0.238	0.237
轮毂	50 940	0.513	0.511
Bunny	69 451	0.696	0.687
Horse	96 966	0.962	0.951
dragon	108 588	1.001	0.987

随后对多个模型使用文中的简化算法进行了简化处理,其中 Bunny 的简化效果如图 4 所示。



Bunny(原始模型的三角面片数是 69 451)最终被简化成只含有 100 个三角面片时仍能保持基本的外观形状。

5 结束语

从实验结果看出与 Garland 的 QEM 算法相比,半边结构的采用使得在模型简化时可以直接确定邻接关系而无需查找链表,所以速度更快;采用三角面片法向量的方差作为顶点重要度形成新的加权 QEM,并使用它作为误差测度来简化模型,取得了良好的效果。

参考文献:

[1] 何辉光,田 捷,张晓鹏,等. 网格模型化简综述[J]. 软件学报,2002,13(12):2215-2224.

[2] Hoppe H. Progressive meshes[C]//Proc of SIGGRAPH'96. [s. l.]:[s. n.],1996:99-108.

[3] Rossignac J,Borrel P. Multi-resolution 3D approximations for rendering complex scenes[M]//Geometric modeling in computer graphics. [s. l.]:Springer-Verlag,1993:455-465.

[4] Klein R,Liebich G,Straßer W. Mesh reduction with error control[C]//Proceedings of IEEE visualization. [s. l.]:ACM Press,1996:311-318.

[5] Ciampalini A,Cignoni P,Montani C,et al. Multi-resolution decimation based on global error[J]. Visual computer,1997,13(5):228-246.

[6] Garland M,Heckbert P S. Surface simplification using quadric error metrics[J]. Computer graphics,1997,31(3):209-216.

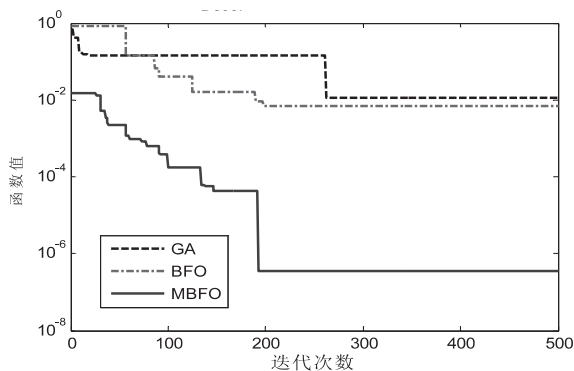


图 3 De Jong 函数三种算法比较图

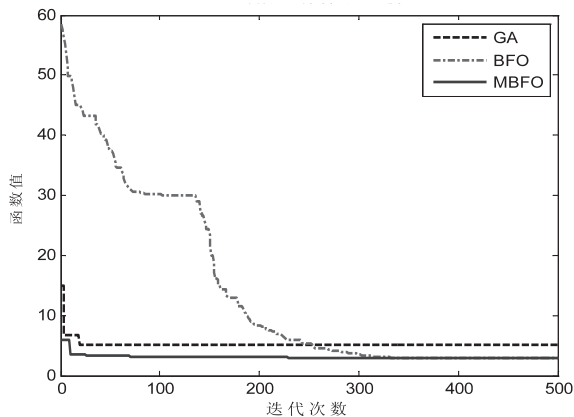


图 4 GP 函数三种算法比较图

复杂函数,比普通遗传算法和普通菌群算法都要好,收敛速度、算法性能方面都有明显的优势,运用于无显著峰谷的非线性优化问题的求解上表现出了优越性。

4 结束语

文中对原始菌群寻优算法进行改进,针对多峰值以及超多极值环境,提出一种新型的多峰值菌群算法,即将算法分为前期和后期两个阶段,前期操作和菌群优化算法相同,使得菌群较快聚集在多个峰值周围,后期阶段将已经分割区域的菌群,在各自所在的小单峰值区域内迭代执行菌群优化算法,独立寻优。通过基准函数的测试实验表明,该算法在收敛性和寻找全局最优解方面有非常好的效果。但是多峰值算法的各阶段参数设置及对收敛性能影响尚未明确,还需要进一步研究和探讨该算法能否应用在 0-1 背包问题上等一系列问题。

(上接第 53 页)

[7] 陈伟海,徐鲤鸿,刘敬猛,等. 网格简化中基于特征矩阵的二次误差测度算法[J]. 北京航空航天大学学报,2009,35(5):572-575.

[8] 杜晓晖,尹宝才,孔德慧. 基于加权二次误差测度的边折叠简化算法[J]. 北京工业大学学报,2007,33(7):731-736.

[9] 陈忆群,曹瑾音,林淑金. 一种均衡代价的网格简化算法[J]. 计算机工程与应用,2011,47(15):75-79.

参考文献:

[1] Passino K M. Biomimicry of bacterial foraging for distributed optimization and control[J]. IEEE control systems magazine, 2002,22(3):52-67.

[2] Muller S, Marchetto J, Airaghi S, et al. Optimization based on bacterial chemotaxis[J]. IEEE trans on evolutionary computation,2002,6(1):16-29.

[3] 李威武,王 慧,邹志君,等. 基于细菌群体趋药性的函数优化方法[J]. 电路与系统学报,2005,10(1):58-63.

[4] 刘文霞,刘晓茹,张建华,等. 基于微分进化和混沌迁移的细菌群体趋药性算法[J]. 控制理论与应用,2009,26(4):353-357.

[5] 储 颖,邵子博,糜 华,等. 细菌觅食算法在图像压缩中的应用[J]. 深圳大学学报(理工版),2008,25(2):153-157.

[6] Chen Hanning, Zhu Yunlong, Hu Kunyuan. Cooperative bacterial foraging algorithm for global optimization[C]//2009 中国控制与决策会议论文集. [s. l.]:[s. n.], 2009:3896-3897.

[7] 李 明,杨成梧. 细菌菌落优化算法[J]. 控制理论与应用,2011,28(2):223-228.

[8] 黄伟锋,林卫星,范怀科,等. 细菌觅食优化的智能 PID 控制[J]. 计算机工程与应用,2011,47(21):82-85.

[9] 杨俊安,庄镇泉,史 亮. 多宇宙并行量子遗传算法[J]. 电子学报,2004,32(6):923-928.

[10] Kim H D, Cho H J. Adaptive tuning of PID controller for multi-variable system using bacterial foraging based optimization [C]//Proceedings of 3rd international Atlantic Web intelligence conference on advances. New York: IEEE, 2005:231-235.

[11] Chen H C. Bacterial foraging based optimization design of fuzzy PID controllers[C]//Proceedings of 4th international conference on intelligent computing. New York: IEEE, 2008:841-849.

[12] Golipudi S V R S, Pattnaik S S, Bajpai O P, et al. Bacterial foraging optimization technique to calculate resonant frequency of rectangular microstrip antenna[J]. International journal of RF and microwave computer-aided engineering, 2008,18(4):383-388.

[13] 张代远. 一类新型改进的广义蚁群优化算法[J]. 计算机技术与发展,2012,22(6):39-44.

[10] 李现民,李桂清,张小玲,等. 基于子分规则的边折叠简化方法[J]. 计算机辅助设计与图形学学报,2002,14(1):8-13.

[11] 侯宝明,崔红霞,刘雪娜. 三维网格模型的快速拓扑重建算法[J]. 计算机应用,2010,30(11):3002-3004.

[12] Garland M. Quadric-based polygonal surface simplification [D]. Carnegie: Carnegie Mellon University, 1999.

基于QEM以方差作为权值的3 D网格模型简化

作者：[侯宝明](#)，[刘雪娜](#)，[鄂旭](#)，[HOU Bao-ming](#)，[LIU Xue-na](#)，[E Xu](#)

作者单位：[渤海大学 信息科学与技术学院, 辽宁 锦州, 121013](#)

刊名：[计算机技术与发展](#)

英文刊名：[Computer Technology and Development](#)

ISTIC

年，卷(期)：2013(11)

本文链接：http://d.wanfangdata.com.cn/Periodical_wjfz201311014.aspx