

1-8-2019

## Integrated Generating Algorithm of Centroidal Capacity Constrained Power Diagram

Liping Zheng

*School of Computer and Information, Hefei University of Technology, Hefei 230009, China;*

Lu Chang

*School of Computer and Information, Hefei University of Technology, Hefei 230009, China;*

Ruiwen Cai

*School of Computer and Information, Hefei University of Technology, Hefei 230009, China;*

Zhiqiang Gui

*School of Computer and Information, Hefei University of Technology, Hefei 230009, China;*

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the [Artificial Intelligence and Robotics Commons](#), [Computer Engineering Commons](#), [Numerical Analysis and Scientific Computing Commons](#), [Operations Research, Systems Engineering and Industrial Engineering Commons](#), and the [Systems Science Commons](#)

---

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

---

# Integrated Generating Algorithm of Centroidal Capacity Constrained Power Diagram

## Abstract

**Abstract:** Power diagram is the weighted Voronoi diagram. Centroidal capacity constrained power diagram (CCCPD) can be obtained by imposing capacity constraint and centroid constraint to the ordinary power diagram. To increase the convergence speed of existing CCCPD algorithms which optimize the weight and location of site interlacedly, *an improved algorithm is proposed which optimizes all variables integratedly*. Research results exhibit that the performance of the proposed algorithm is 40% higher than the traditional method.

## Keywords

power diagram, capacity constrained, centroidal, fast computing

## Recommended Citation

Zheng Liping, Lu Chang, Cai Ruiwen, Gui Zhiqiang. Integrated Generating Algorithm of Centroidal Capacity Constrained Power Diagram[J]. Journal of System Simulation, 2018, 30(7): 2489-2496.

# 质心容量限制 Power 图一体化生成算法

郑利平, 路畅, 蔡瑞文, 桂志强

(合肥工业大学 计算机与信息学院, 安徽 合肥 230009)

**摘要:** Power 图是带权重的 Voronoi 图, 对 Power 图施加容量限制与质心限制, 即可得到基于质心的容量限制 Power 图(Centroidal Capacity Constrained Power Diagram, CCCPD)。为解决现有 CCCPD 生成算法交替优化权重和站点位置带来的相互干扰、收敛减慢问题, 提出对所有变量进行一体化优化的生成算法, 采用多目标优化策略, 直接优化质心限制和容量限制形成的混合能量。结果表明: 所提算法的生成性能比传统方法提高 40%。

**关键词:** Power 图; 容量限制; 质心; 快速生成

中图分类号: TP391.1

文献标识码: A

文章编号: 1004-731X (2018) 07-2489-08

DOI: 10.16182/j.issn1004731x.joss.201807008

## Integrated Generating Algorithm of Centroidal Capacity Constrained Power Diagram

Zheng Liping, Lu Chang, Cai Ruiwen, Gui Zhiqiang

(School of Computer and Information, Hefei University of Technology, Hefei 230009, China)

**Abstract:** Power diagram is the weighted Voronoi diagram. Centroidal capacity constrained power diagram (CCCPD) can be obtained by imposing capacity constraint and centroid constraint to the ordinary power diagram. To increase the convergence speed of existing CCCPD algorithms which optimize the weight and location of site interlacedly, *an improved algorithm is proposed which optimizes all variables integratedly*. Research results exhibit that the performance of the proposed algorithm is 40% higher than the traditional method.

**Keywords:** power diagram; capacity constrained; centroidal; fast computing

## 引言

在计算几何中, Voronoi 图是一种基本的几何结构, 在诸多领域有着广泛的用途。现有的研究对 Voronoi 图做了诸多推广, Power 图是一种带权重 Voronoi 图的推广。对 Power 图的每个区域施加容量约束, 即得到容量限制 Power 图(Capacity Constrained Power Diagram, 简称 CCPD)。进一步

对 CCPD 站点位置施加质心约束, 使每个站点刚好位于 Power 区域的质心, 即得到基于质心的容量限制 Power 图(Centroidal Capacity Constrained Power Diagram, 简称 CCCPD)。

Power 图继承 Voronoi 图优良几何特性的同时, 又具有精确限容等优点, 因而具有广泛用途。

在图形学领域, Power 图被用于采样和点画<sup>[1]</sup>, 蓝噪生成<sup>[2]</sup>, 计算机动画<sup>[3]</sup>, 流体仿真<sup>[4]</sup>等。Power 图作为一种有力的建模工具, 也应用到运筹学中, 通过对 P-中值问题进行建模<sup>[5]</sup>, 来解决选址优化问题<sup>[6]</sup>。同时, Power 图还可作为区域划分和聚类工具<sup>[7]</sup>, 应用到动物学领地研究<sup>[8]</sup>, 机器人和移动传感网络的覆盖路径规划<sup>[9]</sup>。



收稿日期: 2017-08-08 修回日期: 2018-02-02;  
基金项目: 国家重点研发计划(2016YFC0800100), 安徽省科技强警项目(1704d0802177);  
作者简介: 郑利平(1978-), 男, 湖北, 博士, 教授, 研究方向为可视化、群体和疏散仿真; 路畅(1991-), 男, 安徽, 硕士生, 研究方向为计算机图形学与信息可视化。

<http://www.china-simulation.com>

• 2489 •

现有工作对 Power 图生成算法进行了深入研究。Aurenhammer 等提出 Power 图的概念,对 Power 图的性质、计算方法和应用做了系统的总结<sup>[10-11]</sup>, Imai 等<sup>[12]</sup>针对平面点集 Power 图性质给出了证明。对于 Power 图的生成算法, Balzer 等通过反复试探站点权重来满足容量限制,并给出了面向有限空间<sup>[13]</sup>和连续空间<sup>[14]</sup>的容量限制 Power 图算法,在后续工作中通过集成 Lloyd 方法确保质心约束,从而生成基于质心的容量限制 Power 图<sup>[14]</sup>。以上方法采用试探法和逐点迭代,时间复杂度较高且收敛性较差。江婷<sup>[15]</sup>对 Balzer 算法进行改进,通过解析方法计算权值,来取代逐步迭代试探法,因而计算效率有较大提升,但仍然采用逐点迭代策略。Goes<sup>[2]</sup>引入拉格朗日方法来优化质心约束,并指出定点下的容量限制 Power 图是凹最大优化问题,从而提出使用牛顿法来优化权值,并结合 Mullen 提出的自适应步长梯度下降法来优化站点位置,两者交替迭代优化,从而计算生成基于质心的容量限制 Power 图。Goes 文中的距离定义为欧式距离平方,而 Power 图在实际应用中拥有多种不同应用的距离定义方式, Xin 等<sup>[16]</sup>将 Goes 算法拓展到一般距离定义,适用一般代价函数的普遍算法,提出一种超线性优化算法,将 L-BFGS 和 reduced-gradient 进行结合来计算 CCCPD,将其应用到位移插值、蓝噪采样和优化凸剖分等问题上。

现有 Power 图生成算法中,都将 Power 图容量限制和质心限制分割独立计算,或是分步优化,或是逐点优化,导致在优化过程中相互干扰,造成收敛速度缓慢。针对这种情况,在 Goes 算法<sup>[2]</sup>的基础上,提出一种针对所有变量一体优化的策略,以达到加速 Power 图生成的目的。

## 1 Power 图概述

### 1.1 Power 图

Power 图是 Voronoi 图的带权扩展。Voronoi 图是一种空间的划分方案,使每个 Voronoi 区域内

的点到该区域站点的距离最小。假设问题域为  $\mathcal{D} \in \mathbb{R}^2$ , 给定站点集  $X = \{x_i, i = 1, 2, \dots, n\}$ , 生成的 Voronoi 区域  $V = \{x_i, i = 1, 2, \dots, n\}$  其中站点  $x_i$  对应 Voronoi 区域  $v_i$ , 得到 Voronoi 划分:

$$v_i = \left\{ x \mid \|x - x_i\| \leq \|x - x_j\|, \right. \\ \left. \text{for } j = 1, \dots, n \text{ and } j \neq i \right\} \quad (1)$$

式中:  $\|\cdot\|$  定义为欧式距离。

如果为 Voronoi 图每个站点  $x_i$  赋予权重  $\omega_i$ , 重新定义距离, Voronoi 图拓展为 Power 图。本文采用欧式距离平方定义, 如下:

$$d(x, x_i) = \|x - x_i\|^2 - \omega_i \quad (2)$$

设有站点和权重的集合  $\{X, W\} = \{(x_i, \omega_i), i = 1, 2, \dots, n\}$ , 生成的 Power 区域定义为  $\mathbb{P} = \{\mathcal{P}_i^{\omega_i}, i = 1, 2, \dots, n\}$ , 对应的 Power 划分如下:

$$\mathcal{P}_i^{\omega_i} = \left\{ x \mid d(x, x_i) \leq d(x, x_j), \right. \\ \left. \text{for } j = 1, \dots, n \text{ and } j \neq i \right\} \quad (3)$$

### 1.2 基于质心的容量限制 Power 图

在 Power 图的基础上,分别施加容量约束和质心约束,即可得到基于质心的容量限制 Power 图 (CCCPD)。

设问题域  $\mathcal{D}$  是一个密度函数为  $\rho(x)$  的连续域,可以定义 Power 区域  $\mathcal{P}_i^{\omega_i}$  容量为:

$$m_i = \int_{x \in \mathcal{P}_i^{\omega_i}} \rho(x) dx \quad (4)$$

给定容量集  $C = \{c_i\}_{i=1}^n, c_i > 0$  且  $\sum_{i=1}^n c_i = \int_{\mathcal{D}} \rho(x) dx$ , 那么容量约束即通过调整站点权重集  $W$ , 使得任意一个站点  $x_i$  满足:

$$m_i = c_i \quad (5)$$

此时得到严格满足容量限制的 Power 图。

进一步,定义 Power 区域  $\mathcal{P}_i^{\omega_i}$  的质心为  $b_i$ , 在 CCPD 的基础上进一步优化站点的权重或位置,使所有 Power 区域的站点位于 Power 区域的质心,即:

$$x_i = b_i = \frac{\int_{x \in \mathcal{P}_i^{\omega_i}} x \rho(x) dx}{\int_{x \in \mathcal{P}_i^{\omega_i}} \rho(x) dx} \quad (6)$$

即得到基于质心的容量限制 Power 图 (CCCPD)。

## 2 CCCPD 一体化生成算法

Balzer<sup>[1]</sup>和 Goes<sup>[2]</sup>所提算法均采用交替优化权重和位置的优化策略,这种优化策略分割了权重和位置的优化过程,破坏整体收敛性。本文在 Goes 算法的基础上,尝试给出一种同时优化权重和位置的 CCCPD 生成算法,以保证整体收敛性。

### 2.1 优化函数

给定密度函数为  $\rho(x)$  的问题域和一组随机分布的站点  $X = \{x_i, i = 1, 2, \dots, n\}$ , 以及站点的预设容量为  $C = \{c_i, i = 1, 2, \dots, n\}$ 。为了生成质心容量限制 Power 图, 首先定义 Power 图区域的容量约束如式(7), 同时定义质心约束, 从优化的角度定义能量函数  $E_z$  如式(8), 当  $E_z$  到达最小值时, 可得到一个满足质心限制的 Power 图剖分。

$$m_i - c_i = 0, \text{ for } i = 1, 2, \dots, n \quad (7)$$

$$E_z = \sum_{i=1}^n \int_{P_i^{\text{opt}}} \rho(x) \|x - x_i\|^2 dx \quad (8)$$

将式(8)视为目标函数, 式(7)作为等式约束条件, CCCPD 优化问题转变为  $n$  个等式约束条件的函数极小化问题。引入拉格朗日乘子  $\Lambda = \{\lambda_i\}_{i=1}^n$ , 最终的优化过程即通过优化联合能量  $E$ , 能量函数  $E$  为:

$$\min E = \sum_{i=1}^n \lambda_i (m_i - c_i) + E_z \quad (9)$$

为了简化计算, 将式(9)转化为式(10)。

$$\min F(W, X) = \sum_{i=1}^n \omega_i (m_i - c_i) + E_z \quad (10)$$

文献[2]证明式(10)和式(9)具有相同的极值点, 这意味着可以通过最优化方法求解  $F(W, X)$  最小值, 生成 CCCPD。

### 2.2 优化策略

#### 2.2.1 现有优化方法分析

Goes 在文献[2]中使用牛顿法调整权重, 自适应步长最速下降法调整站点位置, 并在文中给出证明式(10)的海森矩阵恰好与其负拉普拉斯变换具有相同的形式, 即  $\nabla_w^2 F(W, X) = -\Delta^{w, \rho}$ , 这为牛顿

法的快速计算提供了基础。该算法为超线性收敛, 时间效率和精度都有极大的提高。然而 Goes 方法的初衷是为了生成优质的蓝噪点, 面向的是相同容量限制的特殊背景。另外 Goes 算法不关注 Power 图的各向同性, 在求解过程中只需寻找局部极值点, 以生成更加自然的蓝噪点分布。

Goes 在算法每次牛顿法优化权重之前, 都将权重向量置为 0, 因此每一轮迭代都丢失了上一轮优化的权重信息, 当  $\|\nabla_w F(W, X)\|$  逐渐变小时, 权重置 0 对时间效率造成极大的浪费。Goes 同样观察到此现象, 采用预处理方式加快收敛, 以节省计算成本, 即使用最速下降法之前, 采用 Lloyd 迭代法对 Power 图进行预处理(通常需要 5 次迭代), 直到质心精度低于阈值。

#### 2.2.2 整体优化策略

本文面向欧氏距离平方定义的 Power 图, 相比 L-BFGS 方法, Goes 所提的牛顿法具有较快的收敛速度, 因此本文仍采用牛顿法。

Xin 等<sup>[16]</sup>指出, 式(10)的优化既不是单纯最大化问题, 也不是单纯最小化问题。因而  $F(W, X)$  不是一个严格凸函数或者凹函数, 可能包含多个极值点, 直接同时优化  $W$ 、 $X$  可能导致算法在局部极值点收敛。由此, 我们先采用同 Goes 类似方式对  $F(W, X)$  进行预处理, 达到极值点附近时, 再进行一体化优化。

在算法的预处理部分, 采用牛顿法和 Lloyd 方法交替优化权重和位置, 算法步骤如下: (1) 固定站点集  $X$ , 通过牛顿法找到满足容量限制的权重集, 更新 Power 图; (2) 使用 Lloyd 迭代方法, 移动所有站点到对应 Power 区域的质心, 更新 Power 图; (3) 若位置精度未达到预设精度  $t_x$ , 重复(1)(2), 否则结束。参照 Goes 算法, 本文预处理模块设置  $t_x$  为  $0.2\sqrt{nm^3}$ 。实验中观察, 通常经过 3 次迭代就能达到预设精度, Power 区域基本均匀。

而经过预处理之后, Power 区域容量基本均匀, 后期可以采用同时优化所有变量的策略。最终实验表明, 相比 Goes 算法, 本文所提的一体化优

化策略具有更好的收敛性和适应性。

## 2.3 CCCPD 生成算法

本文通过改进 Goes 算法实现质心容量限制 Power 图快速生成。Goes 所提出的 CCCPD 生成算法如算法 1 所示。本文所提算法在算法 1 基础上进行优化改进，过程如下：(1) 首先引用算法 1 的预处理模块进行预处理；(2) 其次通过牛顿法优化权重，直到生成严格满足容量限制的 Power 图；(3) 最后采用同时优化所有变量的策略，一体化优化 Power 图的权重和质心，即为 CCCPD 一体化生成算法，如算法 2 所示。

算法 1: Goes 算法

输入: 站点集  $X = \{x_i, i = 1, 2, \dots, n\}$ ,

预设容量集  $C = \{c_i, i = 1, 2, \dots, n\}$ ,

密度  $\rho(x)$ ,

权值精度  $t_w$ , 位置精度  $t_x$

输出: CCCPD

////预处理模块

1. REPEAT
2. Capacity-Constraints()(14-17)
3. //Lloyd 优化质心  $X \leftarrow (b_1, b_2, \dots, b_n)^t$
4. UNTIL  $\|\nabla_x F\| \leq 0.2\sqrt{nm^3}$
- ////交替优化
5. REPEAT
6. REPEAT
7.  $W \leftarrow 0$
8. Capacity-Constraints()(14-17)
9. 更新 Power 图
10. UNTIL  $\|\nabla_w F\| \leq t_w$
11. Centroid-Constraints()(18-21)
12. 更新 Power 图
13. UNTIL  $\|\nabla_x F\| \leq t_x$
- ////牛顿法优化 W
14. 子程序 1 Capacity-Constraints()
15. 牛顿法计算权重方向向量  $\delta$
16. 计算适应 Armijo 条件的步长  $\alpha$

17.  $W \leftarrow W - \alpha\delta$

////自适应时间步长法优化 X

18. 子程序 2 Centroid-Constraints()

19.  $d = \nabla_x F$

20. 计算满足 Armijo 条件的步长  $\beta$

21.  $X \leftarrow X - \beta d$

算法 2: 一体优化 CCCPD 算法

输入: 站点集  $X = \{x_i, i = 1, 2, \dots, n\}$ ,

预设容量集  $C = \{c_i, i = 1, 2, \dots, n\}$ ,

密度  $\rho(x)$ ,

权值精度  $t_w$ , 位置精度  $t_x$

输出: CCCPD

1. 预处理(算法 1(1-4))
2. 牛顿法优化权重(算法 1(6-10))
- ////一体化优化
3. REPEAT
4. 牛顿法计算权重方向向量  $\delta$
5. 计算适应 Armijo 条件的步长  $\alpha$
6. 最速下降法计算位置方向向量  $d = \nabla_x F$
7. 计算满足 Armijo 条件的步长  $\beta$
8.  $W \leftarrow W - \alpha\delta$  &&  $X \leftarrow X - \beta d$
9. 更新 Power 图
10. UNTIL  $\|\nabla_x F\| \leq t_x$  &&  $\|\nabla_w F\| \leq t_w$

## 3 实验结果分析

本文实验环境是操作系统 Ubuntu 14.04 64 位, CPU 为 Intel (R) Core (TM) i7-5500U 4 核 2.40 GHz, 内存为 8.00GB。编译环境是 Qt C++ 4.8.6, 采用 CGAL-4.5.2 (计算几何算法库)、Boost-1.58.0、SuiteSparse 4.5.3 (稀疏矩阵运算库)作为计算核心, 使用 OpenGL 渲染结果。

### 3.1 算法适应性分析

分析本文算法的适应性, 给出本算法在不同站点数量、不同容量限制以及各种复杂密度场下生成 CCCPD 实验结果。实验结果如图 1~2 所示, 图中站点的不同颜色对应不同的权重。

图 1 在均匀密度场下实验结果, 分别选取 40 和 400 个随机站点进行分析。

图 2 为复杂密度场实验结果, 站点数量为 40, 其中(a)、(b)为坐标原点在区域中心的密度分布, (c)、(d)密度函数为, 坐标原点在区域左下角, 密度场渲染到 Power 图的背景中。不同容量限制的 Power 区域使用不同颜色区别。

由实验结果表明, 在不同站点数量、复杂密度场函数和容量限制情况下, 本文算法均能很好生成满足相应条件的 CCCPD。

### 3.2 算法收敛性分析

本文算法优化过程, 即优化能量函数过程, 通

过寻找能量函数  $F(W, X)$  极小值点  $(W^*, X^*)$ , 生成 CCCPD。在最优化求解过程中,  $(W, X)$  逐渐逼近  $(W^*, X^*)$ ,  $F(W^*, X^*)$  的偏导数趋向 0, 因此可以通过函数的梯度表示算法收敛性。分析对比本文算法与 Goes 算法的收敛性, 此处定义容量误差、质心误差为能量函数梯度向量的模长  $\|\nabla_w F\|$ 、 $\|\nabla_x F\|$ , 误差比定义为本文算法误差与 Goes 算法误差的比值。

选取图 2(c)所示密度场, 随机选取 20 k 初始站点, 记录预处理后每一次优化迭代后的误差值如表 1 所示。

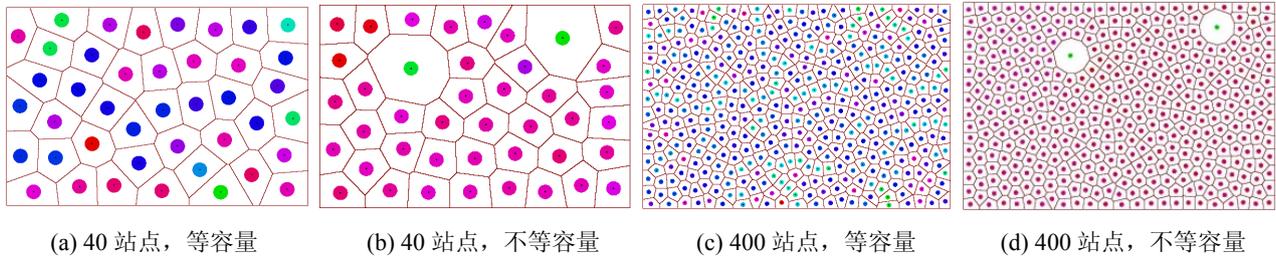


图 1 常密度场下均匀与非均匀容量限制下不同站点生成 CCCPD 对比

Fig. 1 CCCPD comparison at different sites under uniform and unbalanced capacity constraints in a normal density field

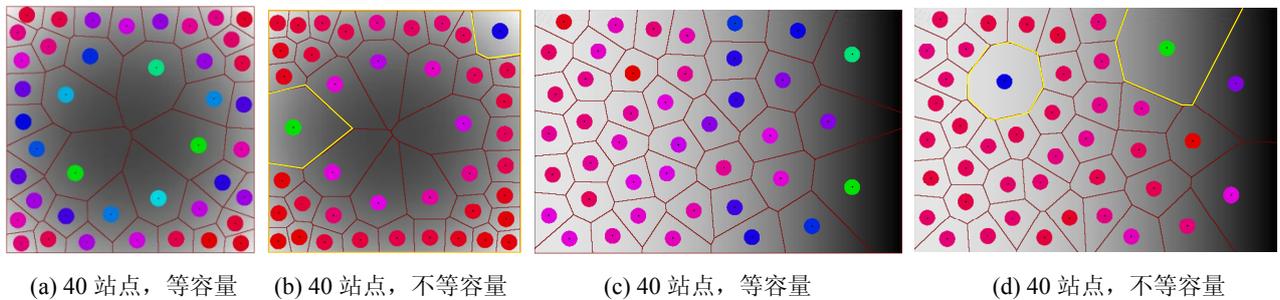


图 2 CCCPD 复杂密度场适应性

Fig. 2 CCCPD complex density field adaptability

表 1 20k 站点优化过程实例

Tab. 1 20k site optimization process example

序号	Goes 方法		本文方法		误差对比%	
	容量误差	质心误差	容量误差	质心误差	容量误差	质心误差
预处理结束	1.0e-04	2.2e-06	1.0e-04	2.2e-06	100.0	100.0
第 1 次优化	8.5e-05	1.4e-06	1.0e-06	1.2e-06	1.2	85.7
第 2 次优化	6.2e-05	1.1e-06	3.9e-07	1.0e-06	0.6	90.9
第 3 次优化	2.8e-05	8.9e-07	2.1e-07	8.4e-07	0.8	94.4
第 4 次优化	2.2e-05	7.4e-07	1.1e-07	7.3e-07	0.5	98.6
第 5 次优化	1.9e-05	6.7e-07	7.7e-08	6.4e-07	0.4	95.5

通过对比分析可知, Goes 算法使用交替优化权重和位置的策略时, 两者相互干扰(特别是容量精度), 破坏了算法的整体收敛性, 而本文算法较好的保持了容量误差的快速收敛, 从图 3 给出的  $\|\nabla_w F\|$  变化曲线可看出。需要指出的是, 由于 Goes 算法每轮迭代的最后一步是采用最速下降法优化质心位置, 因此每轮优化结束后的质心精度较为精确。即便如此, 从表 1 统计的数据可以看出在一体优化策略下, 本文算法的质心精度仍然具有优势。

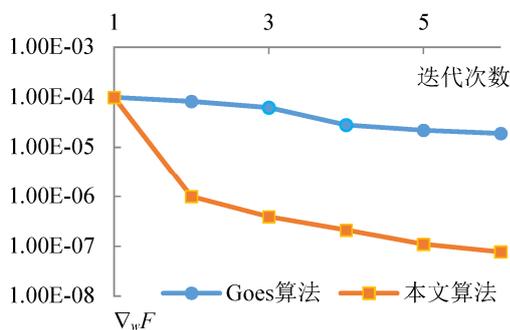


图 3 交替优化和一体优化收敛性对比

Fig. 3 Convergence comparison between interlaced and integrated optimization

### 3.3 算法性能分析

本节通过两组实验分析算法的时间性能和收敛性。第 1 组实验设定问题域为常密度场, 1 000 个种子站点随机分布, 采用 Goes 算法和本文算法分别进行 16 次优化迭代, 算法优化结果、性能和收敛性如图 4(a)~(e)所示; 第 2 组实验设定密度函数为, 40 个种子站点, 优化迭代 32 次, 算法优化结果、性能和收敛性如图 4(f)~(j)所示。图 4(a)、(f)为 Goes 算法结果, (b)、(g)为本文算法结果, (c)、(h)对比两种算法的能量函数, (d)、(i)对比两种算法的权重梯度收敛性, (e)、(j)对比两种算法的位置梯度收敛性。(c)、(d)、(e)、(h)、(i)、(j)中第 1 个数值为预处理结束后的结果, 第 2 个点为牛顿法优化权重后的结果, 此前 2 种算法完全一致, 所以前两次数值相互重叠, 随后本文算法采取一体化优化策略, 可以明显观察到算法效率的提升。

通过实验我们观察到:

(1) Goes 算法生成的 Power 区域显示出各向异

性分布, 本文算法在相同迭代次数下, 得到的 Power 区域形状更加均匀和紧凑, 显示出各向同性的分布。

(2) Goes 算法在每次权重优化前将权重重置零, 实验结果表明, 这种策略有助于 Power 图各向异性分布, 但代价是牺牲容量限制的精度, 从图 4(d)、(i)可以观察到, Goes 算法的权重梯度上下波动, 且不能稳定收敛于较高的精度。本文采取一体化迭代策略, 解决了权重梯度不能稳定收敛的缺陷, 权重梯度的收敛曲线更加陡峭, 收敛性更好。

(3) 文献[16]中证明, 对于能量函数  $F(W, X)$ , 固定权重  $W$  优化位置  $X$ , 是凹最小化问题, 固定位置  $X$  优化权重  $W$  则是凸最大化问题, 这种特性导致能量函数优化难度大。在实验中我们发现这些局部极值点对牛顿法求解造成干扰, 当问题域的密度场较复杂时更加明显, 如图 4(i)所示, 本文算法的权重梯度在收敛过程中出现了波动现象, 在 7~8 步的迭代中陷入了局部最优, 但随着质心位置的变化, 最终跳出了局部极值点, 继续收敛于最值点。

(4) 图 4(e)、(j)实验对比两种算法的质心收敛性, 可以观察到相对于权重收敛性取得的数量级提升, 本文算法的质心收敛性未有明显的提升。这是由于 Goes 算法每次迭代的最后一步为优化质心, 质心精度较高, 位置梯度的收敛性也优于权重梯度, 但是在此情况下本文算法仍有部分提升。

第 3 组实验进一步扩大站点规模, 假定问题域密度场为二次函数, 分别设定变量权重  $w$ 、质心位置  $x$  收敛精度分别为  $0.1 m$ ,  $0.1\sqrt{nm^3}$ , 算法时间如表 2 所示。表 2 第 1 列为实验序号, 第 2、3 列给出实验的站点个数和预设收敛条件的权重梯度, 第 4 列为算法预处理时间, 第 5、6 列分别为使用 Goes 算法策略和本文算法策略的耗时, 第 7、8 列为算法加速比, 第 7 列为总体时间, 第 8 列为不含预处理所用时间。可以看出本文算法在任意站点数量下, 时间性能均有稳定的提升。同时, 我们从另一个角度对比性能, 即固定算法的生成时间, 反过来比较算法精度和收敛性, 实验数据如表 3 所示。通过实验表明, 在此情况下, 本文算法的容量精度具有几个数量级的提升。

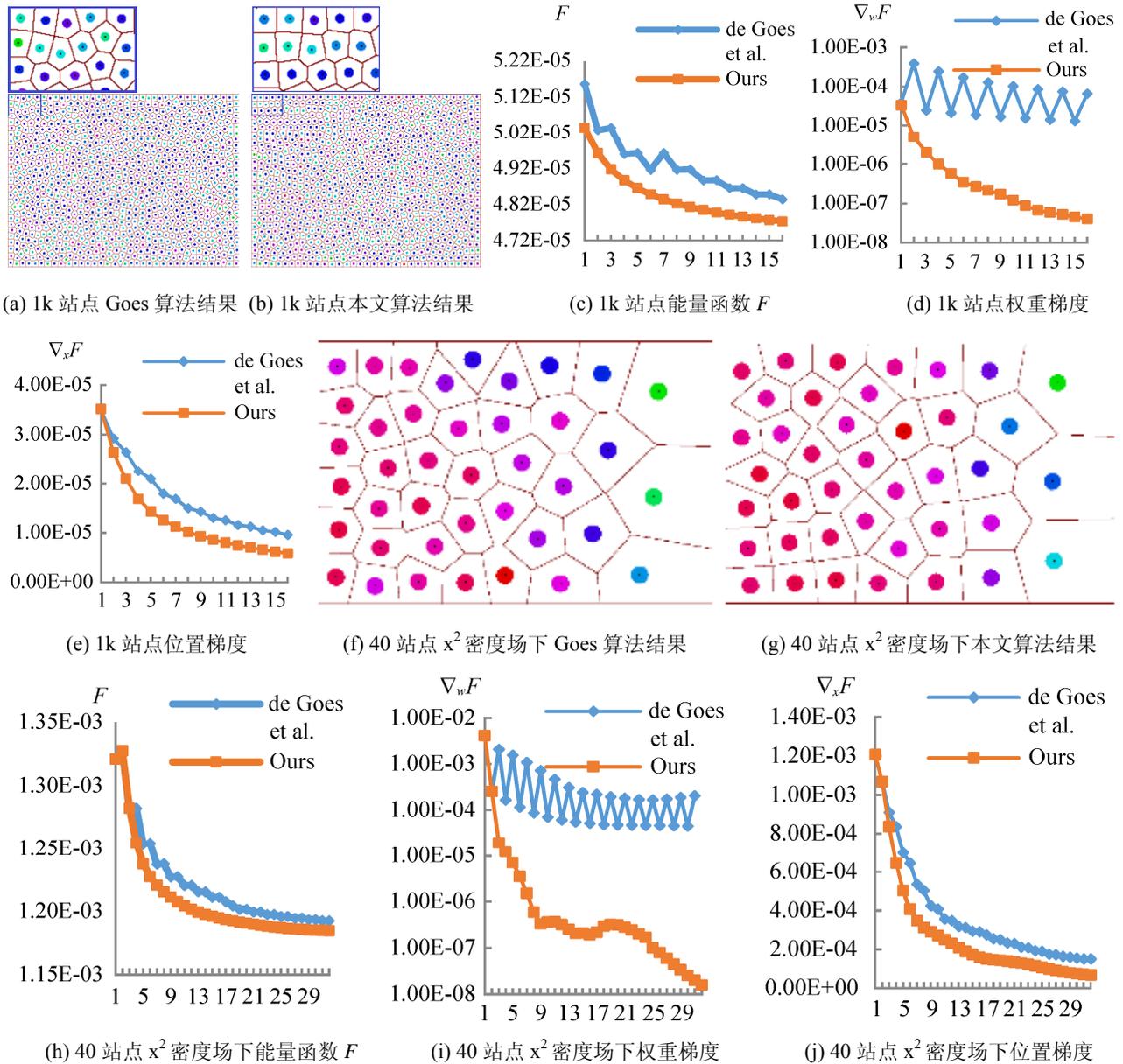


图 4 算法性能与收敛性对比

Fig. 4 Comparison of algorithm performance and convergence.

表 2 相同精度下两种方法生成 CCCPD 时间性能

Tab. 2 Time performance of CCCPD generated by two methods with the same accuracy

序号	站点个数	预设容量精度	预处理/s	交替优化/s	本文方法/s	加速比例	
						总体	优化
1	10	4.4e-03	0.23	0.61	0.36	1.70	2.90
2	100	4.4 e-04	0.44	1.10	0.61	1.80	3.90
3	1 k	4.4e-05	1.00	13.00	1.60	8.10	20.00
4	10 k	4.4e-06	10.60	130.10	15.60	8.30	23.90
5	20 k	2.2e-06	13.70	352.70	22.70	15.5	27.70

表3 相同时间下2种方法生成 CCCPD 精度对比  
Tab. 3 Accuracy performance comparison of CCCPD generated by two methods at the same time

序号	站点个数	时间/s	交替优化		本文方法	
			容量误差	质心误差	容量误差	质心误差
1	10	1	1.0e-4	1.5e-04	4.9 e-08	1.0e-04
2	100	1	4.1e-05	9.7e-05	5.8e-07	9.3e-05
3	1 k	2	2.4e-05	2.6e-05	5.7e-07	1.4e-05
4	10 k	20	7.4e-05	2.2e-06	9.8e-07	1.4e-06
5	20 k	30	2.8e-05	8.9e-07	2.1e-07	8.4e-07

## 4 结论

本文基于 Goes 所提算法, 给出一种所有变量一体优化的 CCCPD 生成算法, 解决现有 CCCPD 生成算法交替优化权重和站点位置带来的相互干扰、收敛缓慢问题。同现有方法相比, 本文所提算法具有很好的计算加速比, 同时具有高精度容量限制等特点。下一步工作包括探索兼容不同距离定义的 Power 图生成算法和利用 GPU 等方法进一步加速 Power 图的生成。

## 参考文献:

- [1] Balzer M, Schlömer T, Deussen O. Capacity-Constrained Point Distributions: A Variant of Lloyd's Method[J]. ACM Transactions on Graphics (S0730-0301), 2009, 28(3): 86.
- [2] De Goes F, Breeden K, Ostromoukhov V, et al. Blue noise through optimal transport [J]. ACM Transactions on Graphics (S0730-0301), 2012, 31(6): 567-571.
- [3] 郑利平, 程亚军, 周乘龙, 等. 异构群体队形光滑变换控制方法[J]. 计算机辅助设计与图形学学报, 2015, 27(10): 1963-1970.  
Zheng Li-Ping, Cheng Ya-Jun, Zhou Cheng-Long, et al. Research on Smooth Formation Control of Heterogeneous Crowds[J]. Journal of Computer-Aided Design & Computer Graphics, 2015, 27(10): 1963-1970.
- [4] Mridul Aanjaneya, Ming Gao, Haixiang Liu, et al. Power diagrams and sparse paged grids for high resolution adaptive liquids[J]. ACM Transactions on Graphics(S0730-0301), 2017, 36(4): 140.
- [5] 郑利平, 江婷, 周乘龙, 等. 基于 Power 图求解容量限制 P-中值问题[J]. 计算机应用, 2015, 35(6): 1623-1627.  
Zheng Li-Ping, Jiang Ting, Zhou Cheng-Long, et al. Solving approach of capacity constrained P-median problem based on Power diagram [J]. Journal of Computer Applications, 2015, 35(6): 1623-1627.
- [6] 郑利平, 刘玉飞, 江婷, 等. 稠密需求下城市应急中心布局方法[J]. 计算机辅助设计与图形学学报, 2014, 26(6): 948-955.  
Zheng Li-Ping, Liu Yu-fei, Jiang Ting, et al. A Layout Approach of City Emergency Centers with Dense Demand [J]. Journal of Computer-Aided Design & Computer Graphics, 2014, 26(6): 948-955.
- [7] Borgwardt S. On Soft Power Diagrams [J]. Journal of Mathematical Modelling and Algorithms in Operations Research (S2214-2487), 2015, 14(2): 173-196.
- [8] G Barlow. Hexagonal territories [J]. Animal Behav (S0003-3472), 1974(22): 876-878.
- [9] 郑利平, 程亚军, 路畅, 等. 质心 Power 图下覆盖路径规划算法[J]. 系统仿真学报, 2017, 29(5): 1120-1124.  
Zheng Liping, Cheng Yajun, Lu Chang, et al. Coverage Path Planning Algorithm Based on Centroidal Power Diagram [J]. Journal of System Simulation, 2017, 29(5): 1120-1124.
- [10] Aurenhammer F. Voronoi diagram-A survey of a fundamental geometry data structure [J]. ACM Computing Surveys(S0360-0300), 1991, 23(3): 345-405.
- [11] Aurenhammer F. Power diagrams: properties, algorithms and applications [J]. SIAM Journal on Computing (S0097-5397), 1987, 16(1): 78-96.
- [12] H Imai, M Iri, K Murota. Voronoi diagram in the Laguerre geometry and its application[J]. SIAM Journal on Computing(S0097-5397), 1985, 14(1): 93-105.
- [13] Balzer Michael, Daniel Heck. Capacity-constrained Voronoi diagrams in finite spaces[C]//Proceedings of the 5th Annual International Symposium on Voronoi Diagrams in Science and Engineering, Voronoi's Impact on Modern Science, 2008, 4(2): 44-56.
- [14] Balzer M. Capacity-constrained Voronoi diagrams in continuous spaces[C]. ISVD, 2009: 79-88.
- [15] 江婷. 容量限制 Power 图的快速生成及其应用[D]. 合肥: 合肥工业大学, 2015.  
Jiang Ting. A Rapid Generating Method of Capacity Constrained Power Diagram and Its Applications [D]. Hefei: Hefei University of Technology, 2015.
- [16] Xin S Q, Vy B, Chen Z, et al. Centroidal power diagrams with capacity constraints: computation, applications, and extension [J]. ACM Transactions on Graphics (S0730-0301), 2016, 35(6): 1-12.