算法设计
——启发式算法

陈卫东

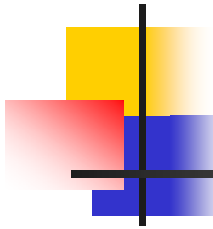
chenwd@scnu. edu. cn

华南师范大学计算机学院

2021-12

NP-hard问题的求解技术

- 如何对付**NP-hard**问题？
  - 许多问题一般无需求最优解，求近似最优解即可
  - NP-hard问题性质：集中研究某些典型NP-hard问题的算法
  - 人类社会的丰富经验可以借鉴
  - 大自然中的一些现象可以借鉴



NP-hard问题的求解技术

- 典型求解技术
  - 精确算法（**Exact Algorithms**）
  - 近似算法（**Approximate Algorithms**）
  - 启发式算法（**Heuristics**）

NP-hard问题的求解技术

- 启发式求解技术（**Heuristics**）

通用启发式算法（Meta-Heuristics）：

简单局部搜索、模拟退火法、禁忌搜索、遗传算法、……

面向问题的启发式算法（Problem-Specific Heuristics）：

贪心算法、拟物算法、 ……

- 快但不够精确。可求解较大规模问题实例，所得解常可满足于实际应用需要。
- 如何评价这类算法？——实验评估为主。

提 纲

- 基本概念
  - 组合优化问题、解、可行解、最优解、邻解关系、评价函数
  - 局部极小、跳坑策略、收敛准则
- 通用启发式算法
  - 简单局部搜索（爬山法）
  - 模拟退火法、禁忌搜索、遗传算法
- 面向具体问题的启发式算法
  - 贪心法及各种近似算法
  - 拟物方法
- 评价方法
  - 理论分析——时间复杂度、解的精确度（近似度）
  - 实验评估——运行时间、解的质量（通常使用Benchmarks）

基本概念

- 组合最优化问题

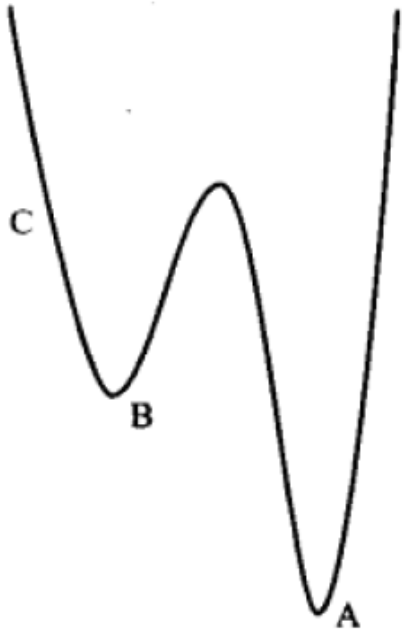
——目标函数、约束条件

——解、可行解、最优解

- 基于局部搜索的各类启发式算法的要素
  - 解的邻域结构（邻解关系）
  - 评价函数
  - 选择邻解的策略
  - 逃离局部最优解的策略（跳坑策略）
  - 终止准则

——算法的基本思想

当势能地形有简单的漏斗结构时，容易找到最低点



多数地形比简单的漏斗复杂得多。例如，在这个“双漏斗”中有一个深的全局最小点和一个浅一些的局部最小点

在一般的能量地形中可能有非常大量的局部最小点，使得寻找全局最小点变得很困难，如这里所画的“带锯齿的漏斗”中那样

通用启发式算法（Meta-Heursitics）

- 简单局部搜索（爬山法）

——基本原理

——要素（实例：点覆盖、图聚类问题的简单局部搜索）

- 模拟退火法

——基本原理

——要素（实例：点覆盖、图聚类问题的模拟退火算法）

- 禁忌搜索

——基本基本原理

——要素（实例：点覆盖、图聚类问题的禁忌搜索算法）

- 遗传算法

——基本原理

——要素（实例：点覆盖、图聚类问题的遗传算法）

通用启发式算法（Meta-Heursitics）

- 模拟退火法**(SA, Simulated Annealing)**

模拟退火 SA(Simulated Annealing)是模拟物理世界中退火过程的随机性迭代寻优方法. SA 算法的一个明显特点就是利用了概率机制来控制"跳坑"的过程. 在搜索过程中,SA 算法不仅接受好解,而且以一定概率接受差解 . 在高温时,接受差解的概率比较大,随着温度的降低,接受差解的概率也随之下降. 当温度趋于 0 时,就不再接受任何差解,这就使得 SA 算法有更多逃离局部最优的"陷阱",避免了其它局部搜索算法存在的缺陷. 经过人们广泛使用,表明 SA 算法对于求解组合最优化问题是有效的.

```
procedure SA
begin
    初始化温度 T
    随机产生一个解 X,其评估值为 eval(X)
    repeat
      repeat
        在 X 邻域内随机选择一新解 X*
        if eval(X*) < eval(X) then
          X←X*
        else if random[0,1) < exp((eval(X) - eval(X*))/T) then
          X←X*
        end if
      until 在温度 T 下达到平衡状态
      T←α * T      //退火
    until 满足停机规则
end
```

通用启发式算法（Meta-Heursitics）

- 禁忌搜索(TS, Tabu Search或Taboo Search)

禁忌搜索的思想最早由Glover(1986)提出，它是对局部领域搜索的一种扩展，是一种全局逐步寻优算法，是对人类智力过程的一种模拟。TS算法通过引入一个灵活的存储结构和相应的禁忌准则来避免迂回搜索，并通过藐视准则来赦免一些被禁忌的优良状态，进而保证多样化的有效探索以最终实现全局优化。相对于**模拟退火**和**遗传算法**，TS是又一种搜索特点不同的 meta-heuristic算法。

通用启发式算法（Meta-Heursitics）

- 禁忌搜索(TS, Tabu Search或Taboo Search)

禁忌搜索是人工智能的一种体现，是局部邻域搜索的一种扩展。禁忌搜索最重要的思想是标记对应已搜索的局部最优解的一些对象，并在进一步的迭代搜索中尽量避开这些对象（而不是绝对禁止循环），从而保证对不同有效搜索途径的探索。

禁忌搜索关键概念

- 禁忌表（tabu list）
- 禁忌长度（tabu length）
- 候选解（candidate）
- 特赦准则（aspiration criterion）

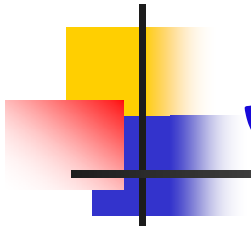
```
procedure TS
begin
    初始化禁忌表：H=∅
    选定一个初始解x
    repeat
        在x的邻域N(x)中选出不受禁忌的候选解集CN(x)
        在CN(x)中选一个评价值最佳的解x*，令x=x*
        更新禁忌表H
    until (满足停止规则)
    将搜索中最好的结果输出
end
```

J Comb Optim
DOI 10.1007/s10878-015-9909-3

Multi-start iterated tabu search for the minimum weight vertex cover problem

Taoqing Zhou[1] · Zhipeng Lü[1] · Yang Wang[1] · Junwen Ding[1] · Bo Peng[1]

Abstract The minimum weight vertex cover problem (MWVCP) is one of the most popular combinatorial optimization problems with various real-world applications. Given an undirected graph where each vertex is weighted, the MWVCP is to find a subset of the vertices which cover all edges of the graph and has a minimum total weight of these vertices. In this paper, we propose a multi-start iterated tabu search algorithm (MS-ITS) to tackle MWVCP. By incorporating an effective tabu search method, MS-ITS exhibits several distinguishing features, including a novel neighborhood construction procedure and a fast evaluation strategy. Extensive experiments on the set of public benchmark instances show that the proposed heuristic is very competitive with the state-of-the-art algorithms in the literature.



面向具体问题的启发式算法

- 贪心法（包括拟人算法、各种近似算法）

——基本原理：人类处理类似问题的经验的抽象

——实例：圆Packing问题的贪心算法（拟人方法）

- 拟物方法（物理方法）

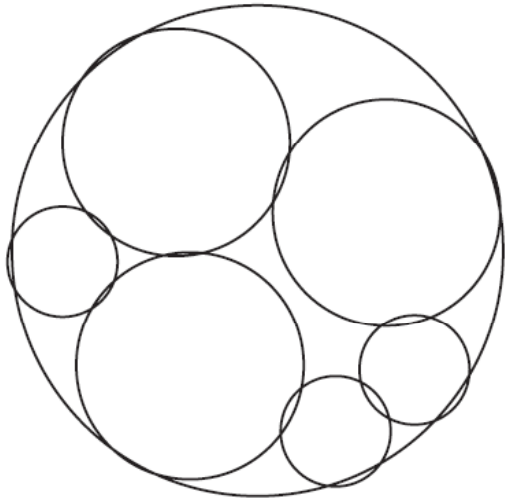
——基本原理：来自于对类似物理现象的模拟（算法运动图像美妙）

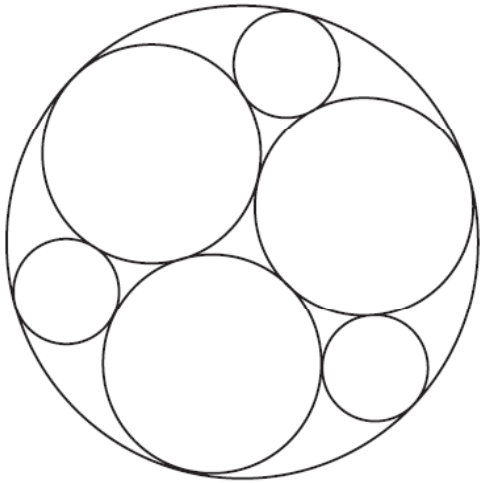
——实例：圆Packing问题的拟物算法的拟物算法

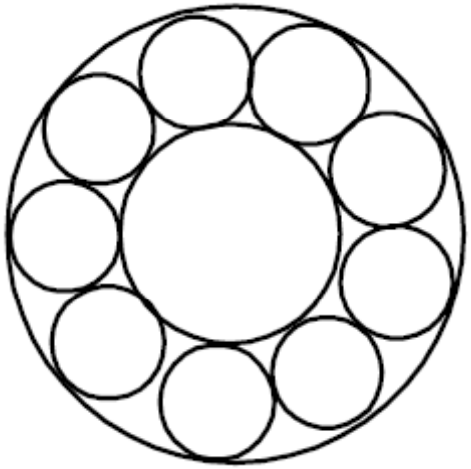
【例】 Circle Packing问题

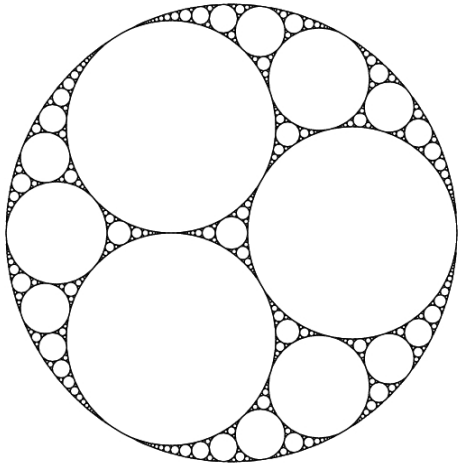
➢ 问题描述

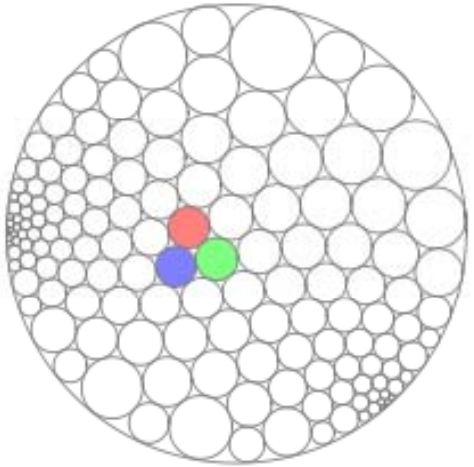
给定一个圆形的容器和若干圆饼, 问能否将这些圆饼互不重叠地放进容器中去.







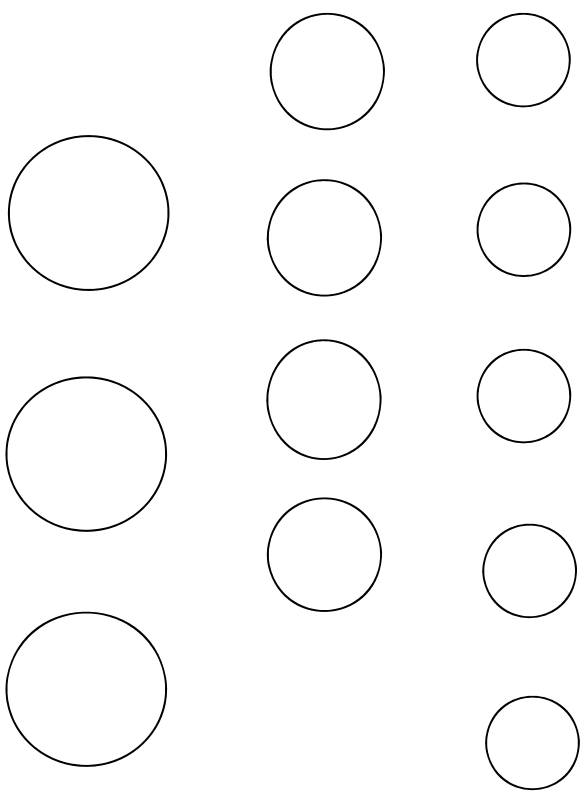


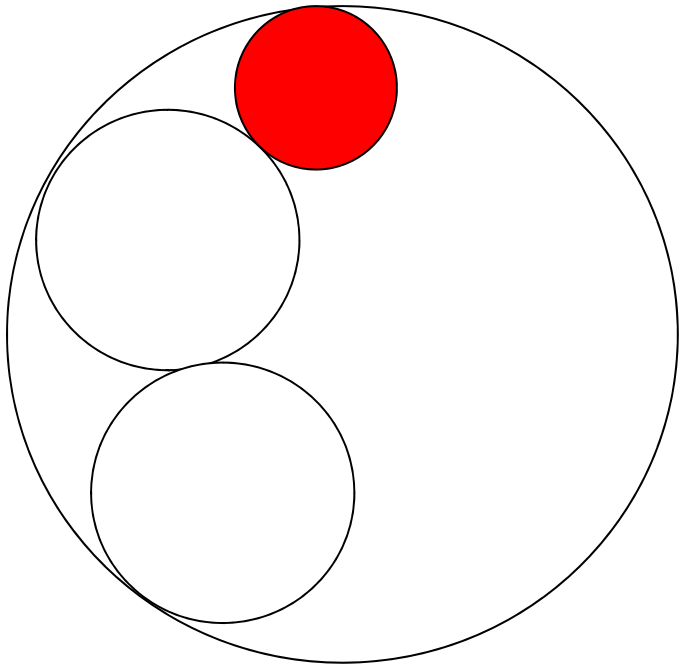


➢ 拟人算法

来自于对人类处理类似问题的经验的抽象：

- 大的优先放
- 靠边放使得剩余空间最大...

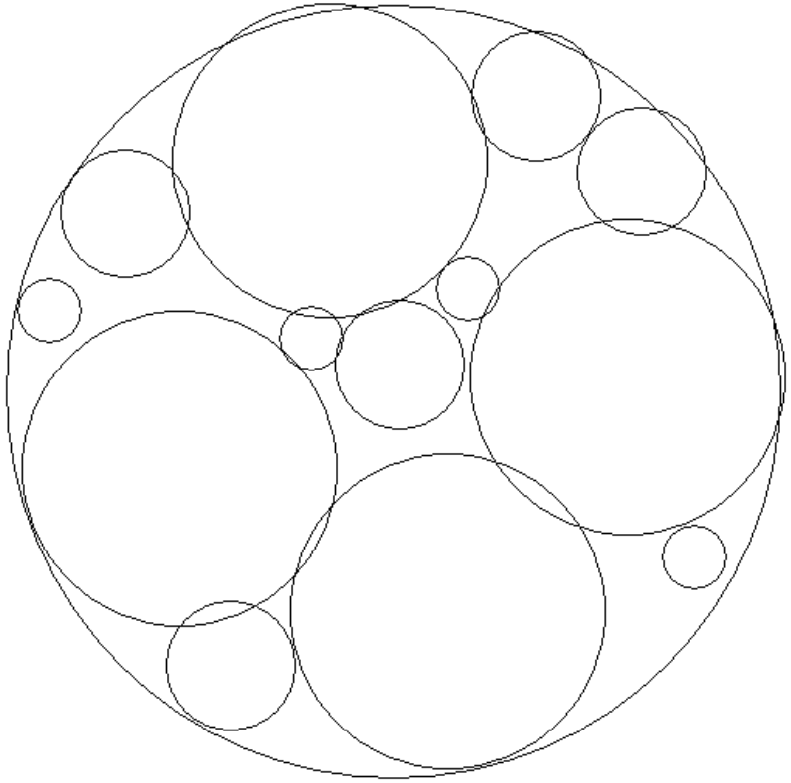


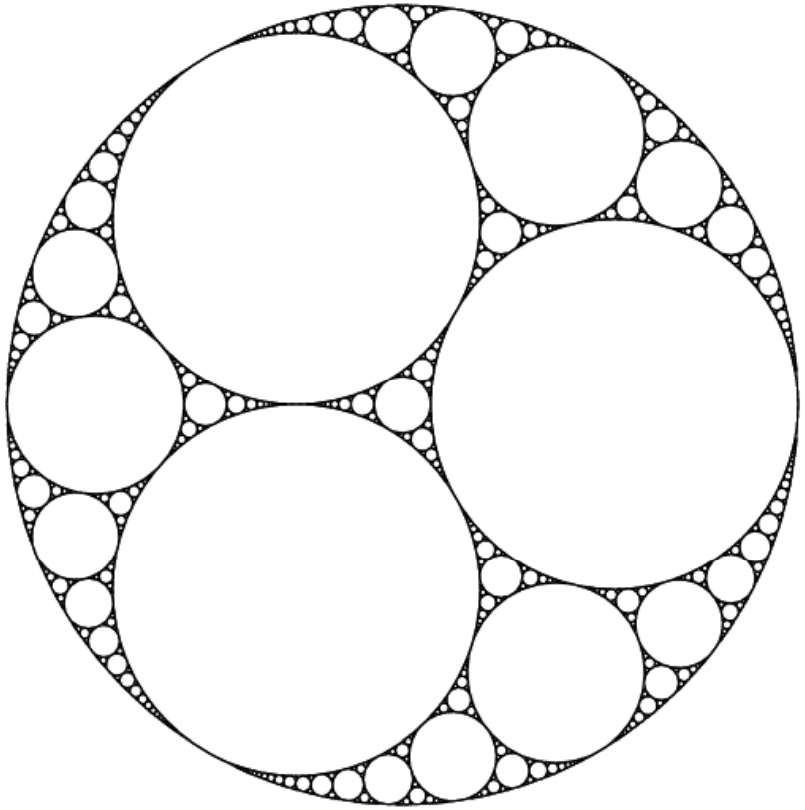


➢ 拟物算法

拟物方法原理

来自于对类似物理现象的模拟，算法运动图像美妙，是“上帝”的杰作。





华中科技大学教授黄文奇在国际上首次提出求解NP难度问题的拟物方法.

(参见 百度百科：黄文奇)

黄文奇教授 (1938-2013)

[1] 黄文奇,金人超. 求解SAT问题的拟物拟人算法—Solar. 中国科学(E辑), 1997, 27(2): 179-186

[2] 黄文奇,许如初. 近世计算理论导引—NP难度问题的背景、前景以及其求解算法研究, 科学出版社, 2004

中国科学 : 信息科学　2011 年　第 41 卷　第 6 期 : 686–693
www.scichina.com　info.scichina.com
《中国科学》杂志社
SCIENCE CHINA PRESS
论　文

求解等圆 Packing 问题的拟物型全局优化算法

黄文奇, 叶涛*

华中科技大学计算机科学与技术学院, 武汉 430074
* 通信作者. E-mail: yeetao@gmail.com

收稿日期: 2010–02–02; 接受日期: 2010–08–20
国家自然科学基金 (批准号: 60773194) 和国家重点基础研究发展计划 (批准号:2004CB318000) 资助项目

摘要　等圆 Packing 问题是一个著名的几何难题, 也是全局优化领域的一个天然明白客观公正的算法试金石. 文中为等圆 Packing 问题提出了一个拟物型的全局优化算法. 在算法中, N 个圆饼在弹性挤压力的作用下平缓地运动, 到达某个局部最优格局; 适当的时期, 又在高强度的引力和斥力的作用下剧烈地运动, 跳出局部最优格局的陷阱, 到达前景可能更好的地方. 使用 N ($N = 1,2,\ldots,150$) 等圆最紧布局的国际记录对算法进行了测试. 对这 150 个算例中的 37 个算例, 此算法找到了比之前此国际最优记录更优的布局方案; 对于剩下的 113 个算例, 都找到了优度与当前国际记录持平的布局方案.

关键词　等圆 Packing 问题　全局优化　拟物方法　启发式算法

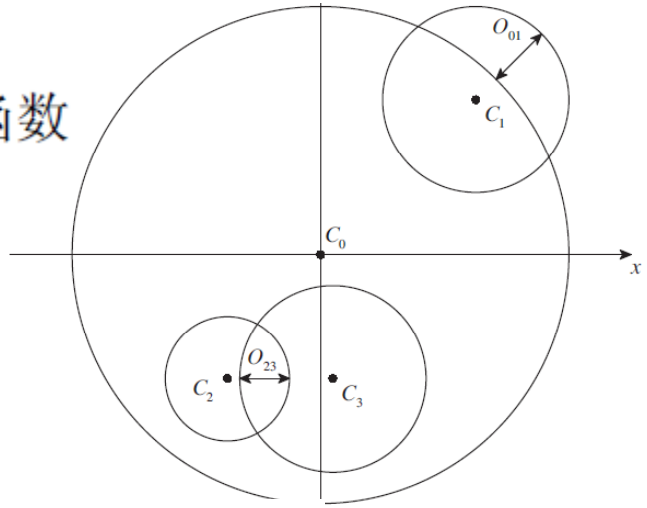
拟物方法描述

拟物算法是通过模拟大自然物质运动规律而得到的连续优化方法, 其作用是从圆形 Packing 问题 (等圆或不等圆) 的任一初始格局收敛至对应的局部最优格局　　拟物算法的主要思想为　　将大圆盘想象成位置固定的弹性圆形容器, 将所有待放置圆饼想像成活动的有弹性的圆形物体. 在初始格局下, 将所有物体强行塞入大容器. 若存在相互嵌入, 根据弹性力学, 相互嵌入的物体之间 (或物体与大容器之间) 存在相互作用的弹性排斥力. 在弹性排斥力的作用下, 存有嵌入的物体将不断地运动以降低嵌入深度, 直至达到局部最优格局为止. 模拟该运动过程, 可实现拟物算法.

拟物算法的本质是梯度下降法, 但其思想来源更天然, 实现方法更简洁, 是求解圆形 Packing 问题的理想的连续优化方法.

为了度量格局 X 的优劣程度, 定义其势能函数

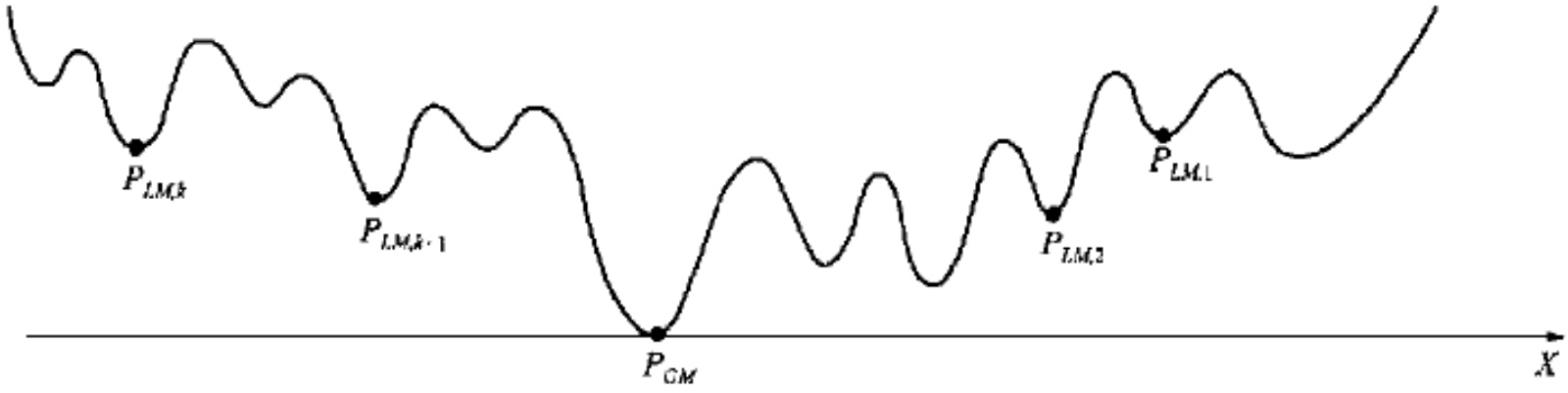
$$U(X)=\sum_{i=0}^{N-1}\sum_{j=i+1}^{N}O_{ij}^2,$$

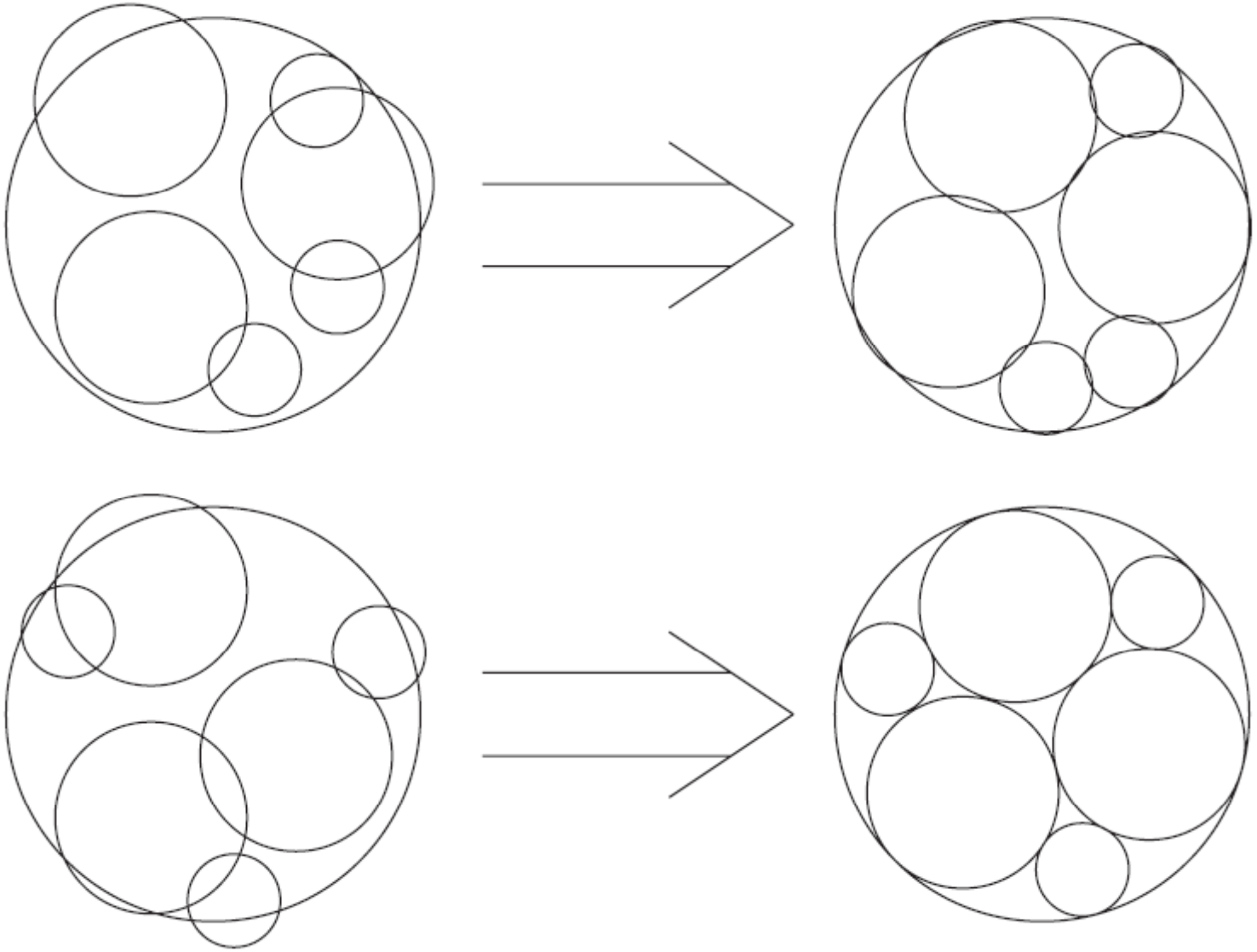


其中, O_{0i} 是第 i 个圆饼 C_i 与大圆盘 C_0 之间的嵌入深度, O_{ij} 是 C_i 与 C_j 之间的嵌入深度.

$$\operatorname{grad} U = \left(\frac{\partial U}{\partial x_1}, \frac{\partial U}{\partial y_1}, \ldots, \frac{\partial U}{\partial x_n}, \frac{\partial U}{\partial y_n} \right).$$

$$\left(x_1^{(1)}, y_1^{(1)}, \ldots, x_n^{(1)}, y_n^{(1)} \right) = \left(x_1^{(0)}, y_1^{(0)}, \ldots, x_n^{(0)}, y_n^{(0)} \right) - h_0 \operatorname{grad} U.$$





采用拟物算法进行连续优化

算法性能评估

Table 1 comparison between the new R_0 and the previously best-known R_0 on 37 improved instances

N	Previous R_0	New R_0	Improvement	N	Previous R_0	New R_0	Improvement
101	11.147199663	11.146933575	0.000266088	128	12.502313068	12.502310071	0.000002997
104	11.317667871	11.317658566	0.000009305	129	12.557855010	12.553717819	0.004137191
105	11.362659648	11.362659613	0.000000035	130	12.602416147	12.602318937	0.000097210
106	11.422447188	11.421834366	0.000612822	131	12.652663183	12.649620461	0.003042722
107	11.472056038	11.472051837	0.000004201	133	12.735331299	12.735273089	0.000058210
108	11.524032176	11.524016134	0.000016042	134	12.772961279	12.771446240	0.001515039
110	11.616921533	11.616861550	0.000059983	135	12.814485271	12.814254772	0.000230499
111	11.662844995	11.662811184	0.000033811	136	12.866888606	12.865759551	0.001129055
112	11.706430293	11.705274526	0.001155767	137	12.915752358	12.914725417	0.001026941
113	11.752226059	11.747528122	0.004697937	138	12.965351253	12.962702608	0.002648645
114	11.798800385	11.795173364	0.003627021	139	13.010791419	13.008987241	0.001804178
116	11.896935573	11.896704500	0.000231073	140	13.061904634	13.061097215	0.000807419
118	11.985568667	11.985551046	0.000017621	141	13.108677936	13.107255295	0.001422641
119	12.042990054	12.042334444	0.000655610	142	13.147405345	13.146411626	0.000993719
120	12.085387934	12.085212460	0.000175474	143	13.200055830	13.197400825	0.002655005
124	12.321728055	12.321708315	0.000019740	147	13.360648155	13.357112495	0.003535660
125	12.368767243	12.368225321	0.000541922	148	13.388017468	13.386939355	0.001078113
126	12.418944183	12.417463956	0.001480227	150	13.461184036	13.460806371	0.000377665
127	12.465924146	12.461549515	0.004374631				

【例】最大割问题的爬山法

- 最大割问题

略.

- 爬山算法

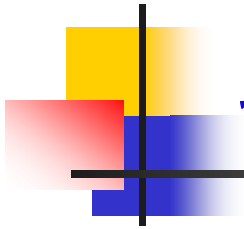
1. 初始化：点集V的两部$V_1=V$，$V_2=\varnothing$，及当前割Cut=∅。
2. 重复下列操作直到Cut不能改进为止：将某点从一部移动到另一部来改进Cut。

- 性能分析

下证该算法是2-近似算法。当算法终止时，点集V的两部V_1和V_2中任意点v在部内相邻点数$dn(v)$不超过部间相邻点数$di(v)$，即$di(v)\geq dn(v)$。
注意，$d(v)=di(v)+dn(v)$且$\sum d(v)=2m$（m是边数）。因此，

$$\sum di(v) \geq m。$$

又$O(I)\leq m$，$A(I)=|\text{Cut}|=(\sum di(v))/2 \geq m/2$。即，$O(I)/A(I) \leq 2$。



评价方法

- 理论分析

——适用范围：贪心算法、简单搜索算法等

——分析内容：时间复杂度、解的精度（近似度）

——实例：最大割问题的爬山法

- 实验评估

——适用范围：各种启发式算法

——评估内容：运行时间、解的质量

——评估方法：常在Benchmarks上与相关State-of-the-Art算法进行比较

——实例：加权点覆盖问题的禁忌搜索算法