

自动布局问题的进化计算算法^{*}

侯广坤, 李 明

(中山大学计算机科学系, 广东 广州 510275)

摘 要: 二维不规则形状物体的自动最优布局是属于 NP 完全的组合优化问题, 使用传统的方法很难得到满意解答. 文章针对该问题提出了一个基于进化计算的算法, 并以服装计算机辅助设计为例表明基于进化计算的算法卓有成效地应用到自动布局问题. 为了得到关于问题的自然和有效的编码, 将问题变换为一种与之等价的关于多边形的运动规划问题, 根据问题的特性设计了算法的 3 个重要算子. 在解的解码和评价过程中则综合地利用了已有的最优算法.

关键词: 进化计算; 组合优化问题; 布局; 多边形运动规划; 服装 CAD

中图分类号: TP311 **文献标识码:** A **文章编号:** 0529-6579 (2001) 02-0006-05

1 自动最优布局问题的描述

在运筹学和计算机科学里有一类被称为切割与填装 (cutting and packing) 的问题, 因为它们具有非常广泛的应用领域和理论上的巨大挑战性, 这些问题一直得到了广泛的重视和研究^[1,2]. 其中比较著名的问题包括一维装箱问题 (one dimensional bin packing): 将一组大小不同的物体装入一些容量相同的箱里, 在物体大小的总和不超过箱的容量的条件下, 使得使用箱的个数最少; 以及它的二维推广形式等.

已经证明一维装箱问题是一个 NP 完全问题^[2]. 其他问题的 NP 完全性可以通过将它们归约到一维装箱问题而得到证明. 尽管多年来这些问题一直受到广泛的重视和研究, 但直到目前为止, 绝大部分的问题还没有获得在实际应用中令人满意的解答. 这种情况充分反映了此类问题本身固有的困难性和复杂性.

二维不规则形状物体的自动最优布局问题是属于切割与填装类的问题: 将一组具有不规则形状的任意多边形物体装入一个宽度固定但长度无限的矩形中, 在这些物体可以根据实际要求进行一定旋转且互相不覆盖的条件下, 使得矩形使用最短的长度. 它涉及到了运筹学、计算几何学、组合优化理论以及人工智能等广泛的研究领域. 随着研究的不断深入, 人们越来越认识到这个问题本身固有的复杂性, 认为解决它需要高度的智

能. 本文的目的就是从人工智能中近年来日益受到重视的一个研究方向——计算智能——出发, 提出一种基于进化计算的算法, 为这一困难而又重要的问题给出一种可行的研究方法.

为方便问题的描述和解决, 采用 2 种坐标系: 全局坐标系和局部坐标系, 其中全局坐标系用来描述多边形的位置, 局部坐标系用来描述多边形的顶点坐标, 如图 1 所示.

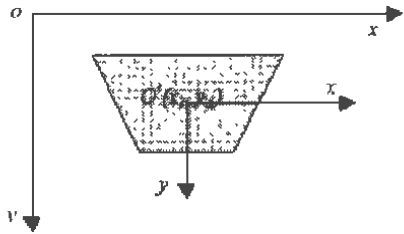


图 1 全局和局部坐标系
Fig 1 The global coordinate system and the local coordinate system

一个由 n 个任意形状的简单多边形 $P_i (i = 1, 2, \dots, n)$ 组成的集合 $M = \{P_1, P_2, \dots, P_n\}$, 矩形 R 的长度固定为 W , 长度 L 不定. P_i 为一个五元组,

$$P_i = (c_i, C_i, A_i, \theta_i, \sigma_i)$$

其中, $c_i = (x_{ci}, y_{ci})$ 是多边形 P_i 外接矩形中点(相对于全局坐标系), $C_i = \{(x_{i1}, y_{i1}), \dots, (x_{im_i}, y_{im_i})\}$ 是多边形 P_i 的顶点集合(相对于局部坐标

^{*} 基金项目: 国家自然科学基金资助项目 (60073056); 广东省自然科学基金资助项目 (001174)

收稿日期: 2000-06-09; 作者简介: 侯广坤 (1941—), 男, 教授.

系), n_i 为多边形 P_i 的顶点个数, $A_i > 0$, 为多边形 P_i 的面积, θ_i 为多边形 P_i 允许旋转的角度. 一般 $\theta_i \in \{0, \pi/2, \pi, 3\pi/2\}$, $\sigma_i \in \{0, 1\}$, 表示是否允许多边形 P_i 绕 x 轴作对称变换, $\sigma_i = 1$ 表示允许, $\sigma_i = 0$ 表示不允许.

问题 将 n 个多边形 $P_i (i = 1, 2, \dots, n)$ 互不相交地装入矩形 R , 使得矩形的长度 L 最短, 将这个最短长度记为 $L_{\min}$.

在工业生产中, 人们比较关心原材料的利用率, 因此本文给出下面定义:

定义 所谓一个布局的材料利用率 E 是指所有多边形的面积总和与使用的原材料面积之比, 即

$$E = \sum_{i=1}^n A_i \backslash (W \times L)$$

所以也可以将问题表述为: 求满足条件的最优布局, 使得材料的利用率 E 取最大值 $E_{\min}$. 一般地, 对于任意简单多边形的布局问题很难获得完全的材料利用率, 即对于大多数情况 $E_{\min}$.

可以证明^[1] 自动最优布局问题是一个 NP 完全问题.

自动最优布局问题与人工智能中机器人学的运动规划问题有一定的相似性, 根据这种相似性, 本文提出关于自动布局问题的一种等价变形.

自动最优布局问题可以看作是关于一组不规则多边形在一个宽度固定、长度不定的矩形容器中的一种运动规划问题. 即多边形在某个位置 (x, y) 处依次水平向左运动 (在运动开始前, 多边形可以根据实际问题的约束条件进行某种旋转和对称变换), 直到与其他多边形或矩形容器左边界刚好相碰为止 (如图 2 所示). 问题要求一种最优运动规划使得矩形容器的长度最短.

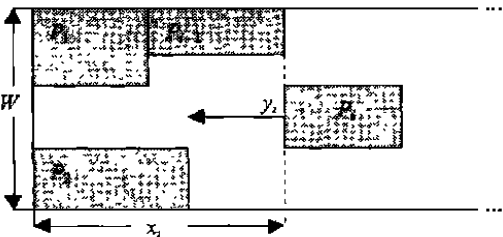


图 2 多边形的一种运动规划

Fig 2 A dynamic programming of polygons

显然自动布局问题有一个解当且仅当存在一

个这样的关于多边形的运动规划. 所以这两种问题的提法是等价的. 这样自动最优布局问题的一个可能解可以表示为:

$$S = ((O_1, x_1, y_1, \theta_1, \sigma_1), (O_2, x_2, y_2, \theta_2, \sigma_2), \dots, (O_n, x_n, y_n, \theta_n, \sigma_n)) \quad (1)$$

其中, $O_i (i = 1, 2, \dots, n)$ 为 $1, 2, \dots, n$ 的一个排列, 表示多边形 P_i 的运动次序; (x_i, y_i) 表示多边形 P_i 的出发位置, $x_i \geq 0, 0 < y_{\min} \leq y_i \leq y_{\max}$, $y_{\min}, y_{\max}$ 分别是 P_i 在垂直方向上最小和最大位置范围; $\theta_i, \sigma_i (i = 1, 2, \dots, n)$ 含义同上.

这个问题的提出具有广泛和真实的应用背景, 具有广泛应用和重要理论研究价值, 因而引起了广泛的关注, 许多学者已经对此进行了长时间的研究.

例如, 已经尝试过的方法主要有: 启发式方法 (heuristic methods)^[1, 3, 4], 模拟退火算法 (simulated annealing algorithm)^[5], 以及在服装 CAD 中的自动排料问题上^[6] 和新闻报业中的自动报纸版面布局问题上^[7] 使用基于学习的方法等. 但这些方法都存在着效率不高、算法复杂等不少问题.

2 自动布局的进化计算算法

自动布局的进化计算算法的问题的关键在于编码方式、评价方式和有关算子的设计.

2.1 自动布局进化计算算法的编码和解的评价

进化算法或演化计算是将自然界的生物演化过程作为一种关键元素来设计和实现基于计算机的自下而上的问题求解系统计算模型^[6, 8], 能够提供健壮的 (robust) 和通用的搜索机制, 具有智能性 (包括自组织、自适应和自学习性等) 和本质并行性显著特点, 解决自动布局问题具有一定的优越性. 本质并行性包括①是进化计算是内在并行的, 即进化算法本身非常适合大规模并行; ②是进化计算的内涵并行性.

由于对原问题直接进行编码是很困难的. 但是在把自动最优布局问题变换为等价的关于多边形的运动规划问题后, 寻找一个最优的布局就等价于寻找一种达到这个最优布局的运动规划. 每一个这样的候选解都可以用 (1) 式的形式来表示. 因此, (1) 式是关于这个问题的进化算法解的编码方案.

自动最优布局问题解的解码与评价过程实际上是进行一次多边形运动规划的模拟. 当完成了解的解码并得到对应的布局后, 就可以立即获得

解的适应度值。在每一个待解码的候选解中包含了每一个多边形开始向左运动的次序、位置、旋转角度及是否作对称变换等所需信息, 所以只需要先将每一多边形进行所需的旋转和对称变换后再按顺序从指定位置向左运动直到与其他多边形或矩形容器的左边界刚好相碰。对多边形的旋转和对称变换均可以在不超过多边形顶点个数量级的时间内完成。而对于一个多边形向另一多边形作直线运动直到刚好与它碰撞为止的问题则是一个很复杂的问题, 一般称之为多边形的靠接问题。多边形的靠接问题在许多实际领域有重要的应用, 而且一直就是计算机图形学和机器人学等领域研究的基本问题。

2.2 选择算子

选择算子采用基于适应值比例的选择操作, 即一个候选解 S_i 被选择为父代个体的概率为

$$p_i = f_i \backslash \sum_{i=1}^N f_i$$

其中, N 为群体大小, f 为适应度值函数。这可以通过模拟转轮盘式选择来实现。首先计算出个体的相对适应值 p_i , 然后根据选择概率 $\{p_i, i=1, 2, \dots, N\}$ 把一个圆盘分成 N 份, 其中第 i 个扇形的中心角为 $2\pi p_i$ 。在进行选择时, 假想转动一下圆盘, 若某个参照点落入第 i 个扇形内, 则选择个体 i 。小扇区的面积越大, 参照点落入其中的概率也越大, 即个体适应值越大, 它被选择到的概率也越大。此外, 在每次选择操作后我们总是保留当前最好解, 并将当前群体的最好解与保留的最好解进行比较以获得到目前为止的最好解。这主要是因为已经从理论上证明了在选择算子后保留当前最好解的进化算法能以概率收敛到问题的最优解。

2.3 重组算子

重组算子的主要作用就是希望将上一代个体中比较好的部分解 (即基因块) 组合在一起, 以生成新的和更好的解。本文采用一般的单点重组算子: 首先两个父代个体根据一定的概率 p_c 发生重组, 然后随机地在这两个父代个体上选择一个点, 再交换这两个串对应的子串 (图 3)。

注意到, 在所采用的编码策略下, 经过这样的重组操作后可能会产生一些不合法的解, 即在一个候选解中可能会有两个多边形的顺序是一样的。本文采用了一种启发式方法来解决这个问题: 当出现这种情况时, 规定面积较大的多边形

的顺序较前。

2.4 变异算子

对同一个候选解的不同部分采用了 2 种不同的变异算子。①一般的变异算子: 对候选解中的除多边形位置 y_k 外的参数均采用一般的随机变异策略, 即在参数的合法取值范围内随机地选取一个值进行替换; ②自适应变异算子: 对多边形位置的参数 y_k 采用经过改进的自适应变异算子。在传统的演化算法中, 算子的作用与演化代数是没有直接关系的。从而, 当算法演化到一定代数以后, 由于缺乏局部搜索, 传统的算子很难获得收益。基于上述原因, Michalewicz^[4] 首先将变异算子的结果与演化代数联系起来, 使得在演化的初期, 变异的范围相对较大, 而随着演化的推进, 变异的范围越来越小, 起着一种对演化系统的微调作用。潘正君等^[9] 则进一步考虑了解的质量与算法局部搜索能力的关系, 认为更合理的情形应该是使适应值大的个体在较小范围内搜索, 而使适应值小的个体在较大范围内搜索。其具体描述如下:

设 $S_i = ((O_1, x_1, y_1, \theta_1, \sigma_1), (O_2, x_2, y_2, \theta_2, \sigma_2), \dots, (O_n, x_n, y_n, \theta_n, \sigma_n))$ 为一个父解, 分量 y_k 被选为进行变异, 其定义区间是 $[y_{kmin}, y_{kmax}]$, 则变异后的解为 $S_i' = ((O_1, x_1, y_1, \theta_1, \sigma_1), (O_2, x_2, y_2, \theta_2, \sigma_2), \dots, (O_n, x_n, y_n', \theta_n, \sigma_n))$ 。其中,

$$y_k' = \begin{cases} y_k + \Delta(T, y_{kmax} - y_k), \text{rnd}(2) = 0 \\ y_k - \Delta(T, y_y - y_{kmin}), \text{rnd}(2) = 1 \end{cases}$$

这里, $\text{rnd}(2)$ 表示将随机均匀产生的正整数模 2 所得的结果, T 为变异温度, 而函数 $\Delta(T, y)$ 的表达式可取为

$$\Delta(T, y) = y \times (1 - r^\lambda)$$

这里, r 是 $[0, 1]$ 上的一个随机数, λ 起着调整局部搜索区域的作用, 其取值一般为 2 到 5。变异温度 T 定义为

$$T = 1 - f_i / f_{\max}$$

其中, $f_{\max}$ 是所解问题的最大适应度值。因为对于很多问题 $f_{\max}$ 是难以确定的, 所以一般采用当前群体中最大适应度值近似地作为 $f_{\max}$ 。这样定义的变异算子将保护较好的解, 使搜索在其较小的领域内进行, 而对适应度不好的解搜索的领域较大。这样使得变异能根据解的质量自适应地调整搜索区域, 从而能明显提高搜索的能力。

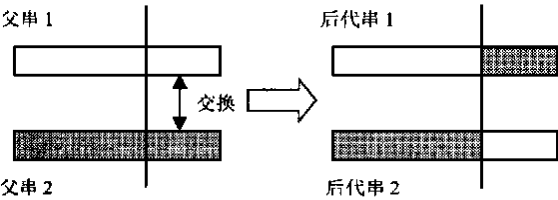


图 3 单点重组算子
Fig 3 The single point reconstructor

3 算法的应用

作者已经在 Pentium II 266 上实现了本文提出的基于演化计算的自动最优布局算法。程序采用 Visual C++ 5.0 开发，在 Windows NT 4.0 平台上运行。实验中采用的数据均来自服装生产中得到的真实数据。

3.1 参数设置

演化算法在实际应用中表现出的性能与如何选取演化算法的各种参数（包括群体大小 N 、重组概率 p_c 和变异概率 p_m 等）有很大的关系，但到目前为止，还没有理论上的研究结果可以用于指导我们如何选取演化算法中最优参数。一般在演化算法的实际应用中采用的都是从已有的实验中获得的经验数据。本文的实验采用了现在一般认为比较合适的参数^[9]，选取群体大小 $N=50$ ，重组概率 $p_c=0.7$ ，变异概率 $p_m=0.06$ 。

3.2 实验结果

选取一部分从生产实际中获得的衣片，共有 21 片。本文实现的算法在上述参数设置下，进化了 3 000 代，最后获得的结果如图 4 所示。

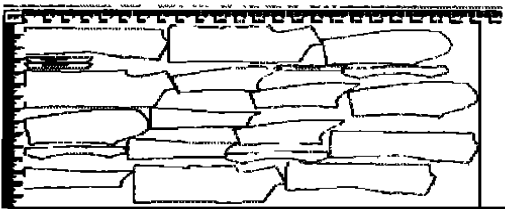


图 4 算法输出的一种可能布局
Fig 4 A layout which may be output by the algorithm

图 5 是算法在进化过程中每一代最优个体的变化曲线。从图 5 中可以看出，我们对于最优布局问题所设计的算法在性能上有两个变化趋势：①随着进化的不断向前推进，群体中最好个体的利用率得到了不断地和稳定地提高；②而且这种提高相对来说是比较快的。我们注意到在前

1000 代演化中最好个体的利用率提高得很快，之后大约每经过 500 代进化，群体中最好个体的利用率就约提高 45%。这种性能稳定提高的趋势和速度在实际应用中是很可观的，这充分说明进化计算非常适合于自动最优布局问题。可以预计在性能更高的计算环境（如并行环境）下，进化算法的群体并行搜索特性将得到更充分的发挥并在较短的时间内获得更好的结果。

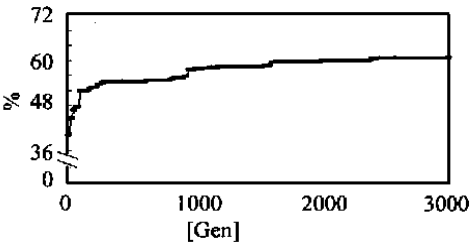


图 5 算法的性能曲线
Fig 5 The performance of the algorithm

参考文献：

[1] LI Z. Compaction algorithms for non-convex polygons and their applications[D]. Division of Applied Sciences, Harvard University, 1994.

[2] GAREY M R, JOHNSON D S. Computers and intractability: A guide to the theory of NP-completeness[M]. New York: W. H. Freeman and Company, 1979.

[3] DANIELS K, MILENKOVIC V J. Multiple translational containment I . An approximate algorithm[J]. Algorithmica, 1997, 19: 148—182.

[4] MILENKOVIC V J. Multiple translational containment II . Exact algorithms[J]. Algorithmica, 1997, 19: 183—218.

[5] 康立山, 谢云, 尤矢勇, 等. 非数值并行算法(第一册)——模拟退火算法[M]. 北京: 科学出版社, 1994.

[6] SPEARS W M, De JONG K A, BACK T, et al. An overview of evolutionary computation[A]. In: Proceedings of the 1993 European conference on machine learning[D], 1993. 11—23.

[7] GOLDBERG D E. Genetic algorithms in search, optimization, and machine learning[M]. Addison Wesley, Reading, MA, 1989.

[8] 刘弘, 曾广周, 林宗楷. 具有类比学习机制的优化排料系统[J]. 计算机辅助设计与图形学学报, 1997, 9(5): 21—27.

[9] 潘正君, 康立山, 陈毓屏. 演化算法[M]. 北京: 清华大学出版社, 1998.

(下转第 26 页)

法, 若采用丢弃分组的方法, 则会导致更多分组的重传, 将加剧网络拥塞. 对数据业务只能用减少源节点发送分组的速率来降低网络的负载, 减轻网络拥塞.

3.3 中断连接

如果网络拥塞继续严重存在, 就采取中断一些优先级较低的连接, 以确保优先级高的连接继续使用网络, 并使拥塞尽快消除, 恢复网络的正常运行.

拥塞控制的实质是网络带宽控制的一个重要方面, 避免和解决拥塞现象与网络控制层、链路控制层和业务控制层都有关系. 我们正是以 3 个

控制层的协调操作来进行共同控制, 所以能得到较好的效果.

参考文献:

- [1] 汤子瀛, 汤小丹. 计算机网络技术及其应用[M]. 成都: 电子科技大学出版社, 1999. 62
- [2] TANENBAUM A S. Computer network[M]. Third ed. 北京: 清华大学出版社, 1999. 286.
- [3] 赵键. 主动网络技术在网络拥塞中的控制策略[A]. 见: '99 青岛—香港国际计算机会议论文集[D]. 1999. 446—451.
- [4] 赵慧玲. ATM、帧中继、IP 技术与应用[M]. 北京: 电子工业出版社, 1999. 105.

The Research of the FRPISCN Network Congestion Control

LUO Yuan-ming, LI Rui-lin, LIU Mei

(Information Science and Technology College, Zhongshan University, Guangzhou 510275, China)

Abstract: Network congestion control is a very important problem of network designing and management. This paper discusses the characters of congestion control and methods to congestion on the private integrated service communication network (FRPISCN) which is based on frame relay network.

Keywords congestion control; flow control; package discarded; forward explicit congestion notification (FECN); time delay; backward explicit congestion notification (BECN)

(上接第 9 页)

A Two Dimensional Object Layout Algorithm Based on Evolutionary Computation

HOU Guang-kun, LI Ming

(Department of Computer Science, Zhongshan University, Guangzhou 510275, China)

Abstract: The two dimensional object layout problem, i. e. layout two dimensional polygonal pieces of some material inside a rectangular sheet of stock material of fixed width and minimum length, has been proved to be an NP complete combinatorial optimization problem, and it is hard to obtain a satisfying solution by traditional methods. In this paper, an Evolutionary Computation (EC) based algorithm about this problem is proposed and implemented, and also applied to an apparel CAD system, showing that it is very good at solving layout problem. In order to encode the problem's candidate solutions, the problem is transformed into its equivalent form about polygon motion planning. The algorithm adopts current optimal algorithms about polygon motion planning in decoding and evaluating candidate solutions.

Keywords: evolutionary computation; combinatorial optimization problem; layout; polygon motion planning; apparel CAD