基于粒子群优化算法的集成电路无网格布线

黄训诚, 庄奕琪, 耿阿囡
(西安电子科技大学 微电子学院,陕西 西安 710071)

摘要: 提出了一种改进的粒子群优化算法,并将其应用于集成电路布线,建立了相应的优化模型。对于给定的版图布线平面,该算法结合无网格算法的思路,首先由障碍图形和各个线网的端点生成一个包含最短路径的无网格访问点阵,然后根据粒子群算法的思路建立初始粒子位置矩阵,并利用其全局寻优功能找到当前布线路径上的最短路径.

关键词: 粒子群优化算法;无网格布线;版图布局优化;Prufer 数
中图分类号:TP301 **文献标识码**:A **文章编号**:1001-2400(2007)01-0034-04

Gridless net routing of the integrated circuit with the particle swarm optimization algorithm

HUANG Xun-cheng, *ZHUANG Yi-qi*, *GENG A-nan*
(School of Microelectronics, Xidian Univ., Xi′an 710071, China)

Abstract: A particle swarm optimization algorithm is presented for the layout of IC design. Particle swarm optimization based on swarm intelligence is a new evolutionary computational tool and is successfully applied in function optimization, neural network design, classification, pattern recognition, signal processing, robot technology and so on. A modified algorithm is presented and applied to the layout of IC design. For a given layout plane, first of all, this algorithm generates the corresponding grid group by barriers and nets'ports with the thought of gridless net routing, establishes the initialization fuzzy matrix, then utilizes the global optimization character to find out the best layout route only if it exits. The results of model simulation indicate that the PSO algorithm is feasible and efficient in IC layout design.

Key Words: particle swarm optimization algorithm; gridless net routing; layout optimization; Prufer number

集成电路布线[1]是一个 NP-完全问题,对于 NP-完全问题,人们已经提出了许多智能解决方法[2~4]. 由于旅行商问题(TSP)可以代表一类组合优化问题,对其演化变形可以解决很多同类的问题,版图布局布线问题就是其典型的推广. 对于基本的版图布线来说,一般可分为均匀网格布线、非均匀网格布线以及无网格布线,相比较而言,无网格算法更适合于大面积的布线平面. 笔者尝试将粒子群优化运用于集成电路的布线算法中,并结合无网格布线的思路探讨线网布线优化.

1 粒子群优化算法

粒子群算法(PSO)是美国学者 Eberhart R C 和 Kennedy J 于 1995 年提出的一种优化算法. 在 PSO 算法[5~7]中,用粒子的位置表示待优化问题的解,每个粒子性能的优劣程度取决于待优化问题目标函数确定的适应值,每个粒子由一个速度矢量决定飞行方向和速率大小. 即在一个 D 维的目标搜索空间中,设有 m 个粒子组成一个群体,则第 i 个粒子在 D 维搜索空间中的位置 $\boldsymbol{x}_i$ 可以表示为 $\boldsymbol{x}_i=(x_{i1},x_{i2},\cdots,x_{iD})$, $i=1,2,\cdots,m$, 第 i 个粒子的飞翔速度也是一个 D 维相量 $\boldsymbol{v}_i=(v_{i1},v_{i2},\cdots,v_{iD})$, $i=1,2,\cdots,m$.

收稿日期:2006-06-06
基金项目:国家自然科学基金资助(60376023)
作者简介:黄训诚(1958-),男,西安电子科技大学博士研究生.

开始执行PSO算法时,首先随机初始化m个粒子的位置和速度,然后通过迭代寻找最优解.在每一次迭代中,粒子通过跟踪两个极值来更新自己的速度和位置:一个极值是粒子本身迄今搜索到的最优解,称为个体极值,表示为 $\boldsymbol{p}_i=(p_{i1},p_{i2},\cdots,p_{iD}),i=1,2,\cdots,m$;另一个极值是整个粒子群到目前为止找到的最优解,称为全局极值,表示为 $\boldsymbol{p}_g=(p_{g1},p_{g2},\cdots,p_{gD})$.

粒子在迭代中根据下列规则更新自己的速度和位置:

$$v_{id}(t+1)=\omega v_{id}(t)+c_1r_1[p_{id}(t)-x_{id}(t)]+c_2r_2[p_{gd}(t)-x_{gd}(t)]\quad, \tag{1}$$

$$x_{id}(t+1)=x_{id}(t)+\alpha v_{id}(t+1)\quad, \tag{2}$$

式中$i=1,2,\cdots,m$;$d=1,2,\cdots,D$;ω是惯性因子[8,9],取大值可使算法具有较强的全局搜索能力,取小值则算法倾向于局部搜索;c_1和c_2为学习因子;r_1和r_2是介于[0,1]间的随机数.α称为约束因子[8,9],目的是控制速度的权重.

2 无网格粒子群优化算法布线

无网格PSO算法布线的基本思想是:首先结合无网格算法的思想由布线平面、障碍图形以及线网的端点确定出一个无网格访问点阵(在访问点阵中一定包含布线的最短路径),然后随机选择两个线网端点,按照粒子群优化算法的基本思想迭代搜索其间的最优路径,循环遍历所有线网端点,进行全局最优路径搜索.

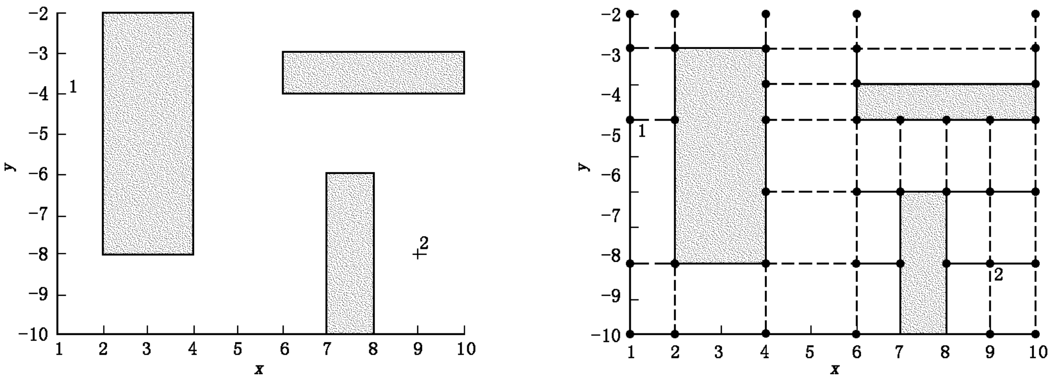
2.1 数学模型

选取总的布线路径长度为目标函数,建立布线优化模型$L=\sum\limits_{i=1}^{n-1}l_i$.

约束条件:①必须保证网络的连通性,即保证所有访问点均与网络连通;②必须满足网络辐射性约束.

2.2 无网格访问点阵的生成

对一个含有障碍图形的布线平面,可以构造出一个不包含障碍图形的布线平面无网格访问点阵 total_dot_group(这里假定障碍图形的边缘是可以走线的).因为点阵中的点是可以互相连通的,所以必定能通过其中有限个点的连接而形成一条从起始点到目标点的最短路径[1].无网格访问点阵 total_dot_group 的生成可以采用线搜索的方式来形成,即从每个障碍图形的顶点以及线网的端点分别向四周产生不可以穿透障碍、布线平面边界以及线网端点的扫描线.如此所得到的扫描线段的两个端点以及相交扫描线段的相交点便构成了所需要的总体访问点阵.



(a) 布线平面图 (b) 总体访问点阵图

图1 无网格算法访问点阵图

图1中,填充区域为障碍图形,虚线为所产生的扫描线,实心点为所生成的访问点,其中图1(a)和图1(b)中标号为1、2的点为线网端点.

2.3 Prufer数编码树

受到粒子群优化算法本身的限制,一般的编码方式难以与粒子的位置和速度相结合.笔者采用Prufer数对树进行编码和解码,先通过模糊矩阵得到一个Prufer数,再对Prufer数进行解码,得到一组线路序列,并由此计算相应的适应度值.Prufer数的编码和解码过程见文献[10].

辐射性的约束条件无法用数学表达式来表示,笔者引入图论中的最小生成树问题[10]. 图论计算中的一个经典定理是 Cayley 定理,即在一个包含 n 个端点的完全图中有 n^{n-2} 个不同的树. Prufer 证明可以仅用 $n-2$ 个数字的排列来惟一地表达一棵树,其中每一个数字都是 1 和 n 之间的整数. 这个排列通常称为 Prufer 数.

2.4 Prufer 数与访问点的关系

论域 X 和 Y 都是有限集合,表示成 $X=\{X_1,X_2,\cdots,X_m\}$,$Y=\{Y_1,Y_2,\cdots,Y_n\}$.

X 和 Y 的模糊关系可以用模糊矩阵表示为 $\boldsymbol{R}=(r_{ij})_{m\times n}=\begin{bmatrix} r_{11} & \cdots & r_{1n} \\ \vdots & & \vdots \\ r_{m1} & \cdots & r_{mn}\end{bmatrix}$,其中 $r_{ij}\in[0,1]$,它表示论域 X 中第 i 个元素 X_i 与论域 Y 中第 j 个元素 Y_j 对于关系 R 的隶属程度. $m\times n$ 阶模糊矩阵表示 Prufer 数与各个访问点之间的关系,其中 n 为访问点数目,$m=n-2$, 为 Prufer 数的位数.

2.5 初 始 化

粒子位置的初始化 $\boldsymbol{P}_0=\begin{bmatrix} P_{11} & \cdots & P_{1n} \\ \vdots & & \vdots \\ P_{n1} & \cdots & P_{nn}\end{bmatrix}$,矩阵中的元素按照如下的条件随机产生:$\sum\limits_{j=1}^{n} P_{ij}=1$,$i=1$,$2,\cdots,n$,$P_{ij}\in(0,1)$. 速度初始化 $\boldsymbol{V}_0=\begin{bmatrix} V_{11} & \cdots & V_{1n} \\ \vdots & & \vdots \\ V_{n1} & \cdots & V_{nn}\end{bmatrix}$,矩阵中的元素也是随机产生的,并满足 $\sum\limits_{j=1}^{n} V_{ij}=0$,$i=1,2,\cdots,n$.

2.6 非模糊化

对模糊矩阵进行解码得到可行解的过程称为非模糊化(解模糊). 粒子群优化算法的位置矩阵每一行代表 Prufer 数的每一位数,位于第 i 行第 j 列的数字,就代表 Prufer 数中第 i 位数等于数字 j 的概率,矩阵每一行的和为 1,也就是代表总概率为 1. 因此可以用最大数法来完成非模糊化的过程. 在这种方法中,每次选择每一行中数字也就是概率最大的那一列,并把它的列号记录下来.

对粒子进行非模糊化后可得到一组数,即为 Prufer 数中的各位数字. 对 Prufer 数进行解码,可得到一组线路序列,对其计算相应的目标函数值,利用粒子群算法的寻优特性进行比较、迭代,最终可收敛于最优解,得到最优的访问点序列.

3 算法描述

算法的软件编程采用 Matlab 实现,以下是基于模糊粒子群优化算法的无网格布线步骤:

(1) 初始化粒子群,使每个粒子得到一个随机的初始位置和一个随机的初始速度.

(2) 对粒子位置进行非模糊化,得到 Prufer 数,对 Prufer 数进行解码,得到访问点序列,计算相应的目标函数值.

(3) 寻找个体极值和全局极值.

(4) 对每一个粒子计算粒子速度 V 和粒子位置 X.

(5) 对粒子的新位置进行非模糊化,得到新的 Prufer 数,对 Prufer 数进行解码,得到访问点序列,计算相应的目标函数值.

(6) 比较当前适应度值是否是历史最优适应度值,选取新的最优适应度值.

(7) 选取当前粒子群中最佳粒子. 若当前最佳粒子的适应度值优于历史最佳粒子的适应度值,则更新最优适应度值.

(8) 重复执行步骤(4)~(7),直至满足终止条件或达到最大迭代次数.

4 实例仿真和结果

由于迄今为止没有标准的两端线网和多端线网布线测试模型,所以笔者自行设计一些布线模型来验证算法.

图 2 是一个两端线网的布线模型,图 3 是一个包含 14 个端点的布线模型. 通过仿真测试,均取得均匀网格以及非均匀网格粒子群优化算法布线更优的仿真结果. 采用粒子群优化算法的无网格与均匀网格的布线结果比较见表 1.

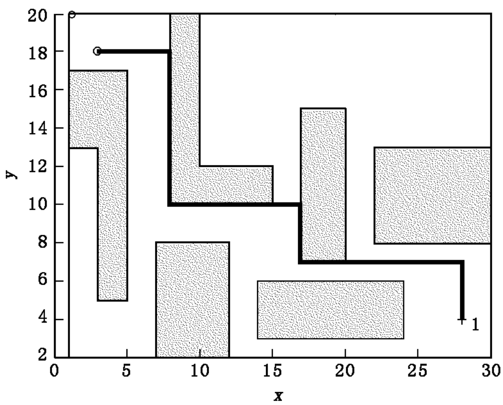


图 2　两端线网布线模型

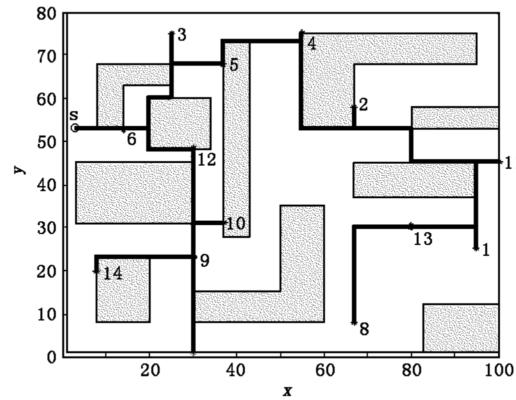


图 3　多端线网布线模型

表 1　采用粒子群优化算法的无网格与均匀网格布线结果比较

布线实例	平均布线长度(10 次)		布通率		平均单次布线时间	
	均匀网格/cm	无网格/cm	均匀网格	无网格	均匀网格/s	无网格/s
两端线网(图 2)	38.4	37	100%	100%	58.609 4	5.719 0
多端线网(图 3)		251		100%		30.861 3

图 2 和 3 中填充区域为障碍图形,粗实线为布好的连线. 在仿真测试中,通过调整粒子群算法中各个参数的设置,对线网的布线长度以及算法的迭代次数进行优化. 优化后的粒子群优化算法参数如下:粒子数 $P_{\mathrm{num}}=200$,最大迭代次数 600 次,最大惯性权重 $\omega_{\max}=1.2$,最小惯性权重 $\omega_{\min}=0.1$,学习系数 $c_1=c_2=2$.

5　与其他算法的比较

粒子群算法和遗传算法均属于演化算法,是对某种自然现象的模拟,两种算法在计算机技术上有着类似的过程:(1)随机初始化群体;(2)对每个个体进行适应性评价;(3)根据某种规则对群体进行演化;(4)迭代循环,直至满足某种停机准则.

两种算法的区别在于演化规则不同,粒子群优化算法模拟群体模型中的信息共享机制;遗传算法模拟物种优胜劣汰的进化机制. 粒子群优化算法没有遗传算法中的交叉、变异操作,每个粒子根据自己的“记忆”和来自同伴的“信息”来调整自己的速度.

蚁群算法具有较强的鲁棒性,蚁群算法模型稍加改进,就可以应用于其他问题;具有本质并行性,易于并行实现;易于和其他方法结合,以改善算法的性能等优点. 蚁群算法应用于解决集成电路的布线问题时,会出现进化速度较慢,特别是随着计算规模的扩大,计算时间会较长,正反馈原理强化了性能较好的解,却容易出现停滞现象,存在着易于陷于局部最小值的缺陷.

与遗传算法相比,粒子群算法在实际计算过程中,以较小的群体规模可以更快地收敛到最优解,这是由于在搜索过程中,把信息传给所有粒子,使得粒子的运动紧跟当前的最优解,加快了搜索的速度,随机参数的设置使得粒子的搜索更加细致,同时,惯性系数的设置可以调整粒子的搜索能力,防止迭代陷入局部最优解而发生早熟现象.

表 2　采用粒子群优化算法的无网格与蚁群算法的无网格布线结果比较

布线实例	平均布线长度(10 次)		布通率		平均单次布线时间	
	蚁群算法/cm	粒子群算法/cm	蚁群算法	粒子群算法	蚁群算法/s	粒子群算法/s
两端线网(图 2)	37(37)	37	100%	100%	7.758	5.719 0
多端线网(图 3)	256.3(251)	251	100%	100%	42.672	30.861 3

采用粒子群优化算法的无网格布线结果与采用蚁群算法的无网格布线结果比较见表 2. (下转第 86 页)