

基于 VPR 的 FPGA 布局算法的改进

杜星格, 袁小龙, 余 明

(北京交通大学 电子信息工程学院, 北京 100044)

摘 要: 通过在原模拟退火算法中加入回火过程对原算法进行优化, 同时, 减小内循环次数用于弥补回火导致的布局时间增量. 回火过程用于寻找“被遗漏”的最优解. 结果表明新算法利于跳出局部最优“陷阱”, 进一步搜索最优解, 具体体现在改进后的算法不但能够保证布局质量, 而且缩减了布局时间, 同时, 布线时间与电路关键路径延时得到不同程度的改善.

关键词: 现场可编程门阵列; 模拟退火算法; 通用布局布线工具; FPGA 布局算法; 回火过程; 退火表

中图分类号: TP312

文献标识码: A

文章编号: 1000-7180(2013)07-0064-04

Improvement of FPGA Placement Algorithm Based on VPR

DU Xing-ge, YUAN Xiao-long, YU Ming

(School of Electronic and Information Engineering, Beijing Jiaotong University, Beijing 100044, China)

Abstract: In this paper, adding the tempering process to the original algorithm and reducing the number of inner loop to compensate for layout delay increments due to tempering. This process is used to find better placement that haven been abandoned before. The results show that the new algorithm is conducive to jump out of the local optimum "trap", search for the optimal solution. In detail, the improved algorithm not only keep good placement results, but also largely reduce layout time, at the same time, routing time and critical path delay also have been reduced.

Key words: FPGA; simulated annealing algorithm; VPR; FPGA placement algorithm; tempering process; annealing table

1 引言

FPGA 的设计流程大致分为七步: 设计输入、功能仿真、逻辑综合、映射、布局布线、时序仿真和生成 bit 文件并下载到 FPGA^[1] 中. 本文主要研究 FPGA 布局算法, 布局算法能够确定实现电路功能所需要的逻辑单元在 FPGA 中的具体位置, 布局结果的好坏直接影响后续布线的长度、布线资源占用率以及关键路径延时等. FPGA 的布局问题属于 NP 完全问题, 模拟退火算法可以有效的解决此类问题.

VPR(versatile place and route)^[2], 国际上权威的通用布局布线工具, 并且开放源代码, 利于人们进行各种实验. VPR 的布局算法以经典模拟退火算法

为基础, 加入一些优化目标或约束条件, 最终形成适于解决 FPGA 布局的算法. VPR 工具支持 linux 和 windows 两种操作系统, 安装简便, 并且在执行布局布线的过程中能够打印出充足的关键信息, 如: FPGA 尺寸大小、代价函数的平均值, 布局交换次数, 布局、布线耗用时间关键路径延时和通道宽度等.

本文以 VTR 1.0 为实验平台, 在原有的模拟退火算法的基础上加入回火过程, 提出几种不同的回火方式, 通过对比实验结果得出结论.

2 VPR 中的模拟退火算法

模拟退火算法 (Simulated Annealing)^[3], 顾名思义该算法模仿逐渐冷却熔化金属以制造高质量金

收稿日期: 2012-10-06; 修回日期: 2012-11-03

基金项目: 国家自然科学基金(61076034); 中央高校基本科研业务费专项资金(2011JBZ002)

属材料的退火过程. 通过不断的降低温度来控制待布局各元素交换范围以及最终位置.

模拟退火算法具有内外两个循环过程, 内循环用于确定各温度层的布局交换次数, 外循环决定具体降温策略, 即退火过程. 退火表^[4]的优劣以及计算布局代价函数差值的速度是评价模拟退火算法好坏的重要指标. VPR中使用的退火方式即退火表的生成受当前温度下接受布局变动百分比的控制, 如果接受百分比比较大则降温相对较多, 反之降温较少. 在线长驱动布局中, 代价函数使用线网边界框的半周长之和. VPR中使用新型的代价函数^[5]: 考虑线性拥挤. 这是因为由于FPGA结构不同, 布线通道可能不完全相等, 这就需要将布线需求大的电路部分布局到通道宽度较宽的位置. 代价函数用Cost函数表示, 每次移动逻辑单元或IO导致的代价增量用 ΔCost 表示. 若 ΔCost 大于零, 则布局变坏, 反之布局得到优化.

VPR使用的寻找布局最优解的退火过程如图1所示.

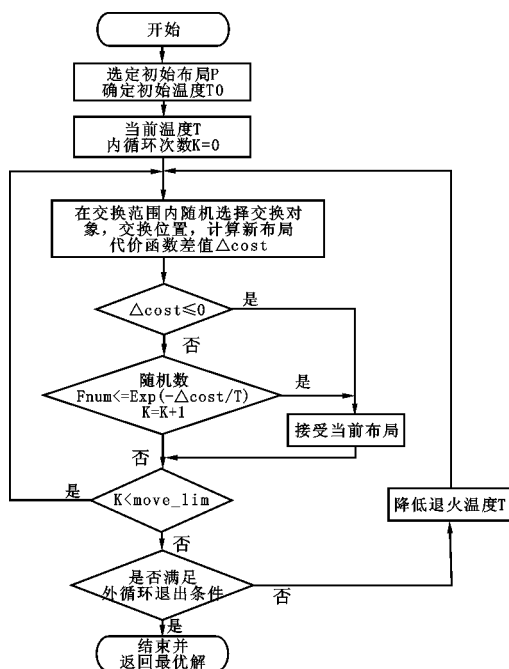


图1 VPR布局过程

第一步: 初始化

① 确定初始温度: 布局准备阶段确定FPGA尺寸即逻辑单元数(CLB)和IO个数, 并随机生成初始布局位置, 选择 20σ 作为初始温度 T_0 , 其中 σ 是对初始布局进行一组随机移动成本函数的标准偏差;

② 初始化内循环次数 $K=0$, K 值记录当前内循环执行次数;

③ 终止条件: $T < \epsilon \frac{\text{Cost}(P_k)}{N_{\text{nets}}}$, 其中, ϵ 设为 0.

005, $\text{Cost}(P_k)$ 为当前布局的代价函数, N_{nets} 为电路中线网的总数;

④ 每一个温度下的交换次数 $\text{move_lim} = \text{inner_num} * N_{\text{blocks}}^{1.333}$.

第二步: 在交换范围(与当前温度密切相关)内随机选择CLB或IO进行交换, 交换后的布局为 P_k , 计算评价函数Cost的增量 $\Delta\text{Cost} = \text{Cost}(P_k) - \text{Cost}(P_{k-1})$ 的值;

第三步: 若 $\Delta\text{Cost} \leq 0$, 则接受本次交换; 反之, 根据Metropolis准则以 $e^{-\Delta\text{Cost}/T}$ 的概率接受此次交换. 此时, 随机生成 $[0, 1]$ 内的随机数若该数不大于 $e^{-\Delta\text{Cost}/T}$ 接受本次交换, 即本次交换使布局变差但在可允许范围内, 反之, 交换失败. K 值加1, 用于记录布局交换已经进行的次数, 依此控制内循环是否介绍;

第四步: 判断 K 值是否大于内循环执行次数 move_lim , 如果小于它, 跳转到第二步; 反之, 跳出内循环判断是否满足终止条件, 若满足取当前解为最优解, 否则, 跳转到第五步, 进行退火降温;

第五步: 退火: $T = \lambda T_{\text{old}}$, 其中, λ 的值根据 α 的变化而变化, α 为当前温度下接受交换的百分比, 如表1所示.

表1 退火方式

α	λ
$\alpha > 0.96$	0.5
$0.8 < \alpha \leq 0.96$	0.9
$0.15 < \alpha \leq 0.8$	0.95
$\alpha \leq 0.15$	0.8

跳转到第二步, 在新温度下进行交换.

3 退火一回火算法

由上述模拟退火算法可知, 随着退火的进行, 接收概率仍按照Metropolis准则以概率 $e^{-\Delta\text{Cost}/T}$ 判断与否接受本次交换. 此时经过很多次退火降温过程, 温度 T 已经变得很小, 故接收概率很低, 跳出局部最优解较为困难. 加入回火^[6]过程, 即将温度适当提高, 但回火后的温度应低于前一次回火温度值, 否则不仅会浪费时间, 布局结果也将难以达到最优. 回火后将在当前最优解的基础上重复寻找最优解的迭代过程, 重复退火一回火过程有利于跳出局部最优解“陷阱”. 但是, 如果单纯的加入回火过程而不做其他任何修改, 无疑会导致布局耗时大幅增加. 此时, 本文选择降低各个温度层下的布局交换次数即 move_

lim, 而布局交换次数受 inner_num 控制, 故通过降低 inner_num 抵消回火带来的布局时间增量。

本文使用两种不同的回火方式:

第一种方法, 回火阶段贯穿布局的整个过程, 即每一次退火完成后都伴随回火过程;

第二种方法, 高温阶段并不加入回火过程, 直到退火过程进行到一定阶段再加入回火过程。

本文共进行了五组实验, 分别如下:

①原 VPR 无回火;

②全程回火 (0.3): 使用第一种方法, inner_num 取原值的 0.3 倍;

③全程回火 (0.1): 使用第一种方法, inner_num 取原值的 0.1 倍;

④中途回火^[7] (0.3): 使用第二种方法, inner_num 取原值的 0.3 倍;

⑤中途回火 (0.1): 使用第二种方法, inner_num 取原值的 0.1 倍。

以上实验均使用 MCNC 基准电路中最大的 20 个电路, 详见下文。回火过程是温度升高的过程, 由于回火后的温度不得超过前一次回火温度, 结合表 1 的退火方式可知回火倍数不得超过 1.0526, 实验中选择 1.025。单纯退火过程进行 80 次以后加入回火过程。

4 实验结果分析

本实验以 VTR 1.0 (VPR 的最新版本) 为实验平台, 实验所用计算机处理器为 pentium2.8GHz, 内存 1.0GB, windows XP 操作系统。

交换次数与布局时间成正相关关系, 如图 2 所示。全程回火 (0.3) 较原算法基本没有变化, 甚至稍有些变差; 其余三种实验方法在交换次数上均有所改善, 节约了布局时间。较为大型的电路交换次数降低较为明显, 仅就交换次数改善程度而言, 中途回火 (0.1) 最优, 全程回火 (0.1) 次之, 中途回火 (0.3) 第三, 全程回火 (0.3) 最差, 据统计中途回火 (0.1) 的平均交换次数仅占原来的 17.7%。该结果是可以预见的, 关键取决于节约布局时间是否是以牺牲布局效果为代价的, 因此, 需要观察其对布线带来的影响, 此时以布线时间、关键路径延时和布线通道宽度为衡量标准。

将布线时间进行比对, 如图 3 所示。中途回火 (0.1) 在 ex1010 处由原来的 400.172s 变为 637.72s, 说明布局结果远不如原算法。从整体上看, 加入回火过程对前 12 个相对规模较小的电路影响很小, 这说明回火过程在保证布局质量的前提下提高了布局速率。对

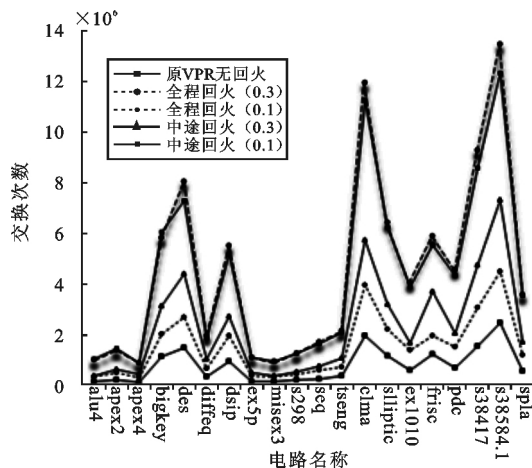


图 2 布局交换次数对比

于之后的 8 个相对较大的电路影响则不尽相同, 不排除个别电路布线时间增大的情况, 但整体结果可以接受。计算布线时间的平均值后发现全程回火 (0.3) 提高 2.3%, 全程回火 (0.1) 提高 16.8%, 中途回火 (0.3) 提高 3.8%, 中途回火 (0.1) 提高 4%。

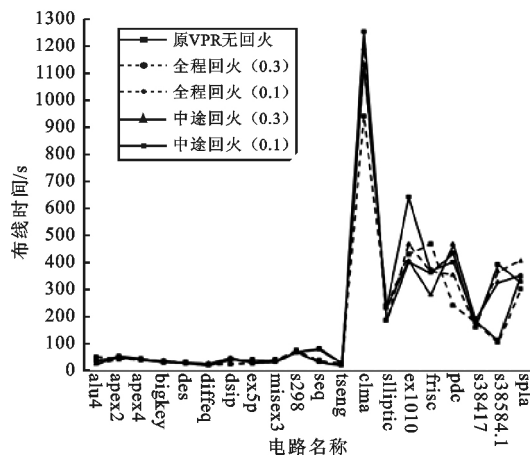


图 3 布线时间对比

以原算法平均值为基准, 分别计算其余四种回火方式下关键路径延时和通道宽度均值的相对值。结果表明, 四种方法布线速率均有提高; 通道宽度^[8]除全程回火 (0.1), 其余均稍有恶化。由于表中数据为平均值, 该表仅代表总体趋势, 针对不同电路有不同的效果。通道宽度恶化值仅为原来的千分之四或千分之八, 故可以认为总体通道宽度无明显影响。如表 2 所示。

表 2 关键路径延时、通道宽度均值统计

平均值	原 VPR 无回火	全程回火 (0.3)	全程回火 (0.1)	中途回火 (0.3)	中途回火 (0.1)
关键路径延时	1	0.977	0.981	0.976	0.979
通道宽度	1	1.004	0.998	1.004	1.008

综上所述,全程回火(0.1)优于其它三种方法. 实际应用中,需要综合考虑找到各因素之间的折中点. 图4以alu4电路的布局布线结果.

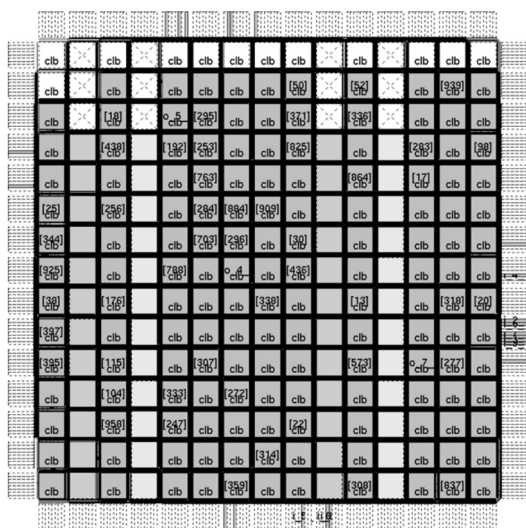


图4 alu4 布线结果

5 结束语

本文通过加入回火算法对原有VPR布局中的模拟退火算法进行改进,实验数据表明,加入回火过程有利于跳出原来的局部最优解找到更加优化的新解. 改进的布局算法一方面保证布局质量,即方便后续布线过程,具体表现在布线耗费时间上,同时关键路径延时有所下降;另一方面,加入回火算法后很大程度上降低了布局用时,有利于VPR更好更快的完成FPGA布局布线过程. 模拟退火算法改进方面的文献资料有很多,大多通过改变扰动模型、产生新解方式、接受概率与退火结束温度对布局算法进行改进. 但结果多以牺牲布线时间、通道宽度与关键路径延时为代价来换取布局时间的减少. 而本文的优势在于方法简单,效果明显,以保证布局质量为前提,大大节省布局时间. 但本文的改进方法并未考虑CLB内部连线^[9]、面积和功耗^[10]问题,实际应用中可以根据需要选择合适的回火参数,以及回火过程加入方式.

参考文献:

- [1] Deming Chen, Jason Cong, Peichen Pan. FPGA design automation: a survey [J]. Electronic Design Automation, 2006, 1(3):195-198.
- [2] Betz V. VPR user's manual [EB/OL]. (2012-01-19). [2012-09-15]. <http://www.eecg.toronto.edu/~jayar/>.
- [3] Kirkpatrick S, Gelattand C, Vecchi M. Optimization by simulated annealing [J]. Science, 1983, 220(4598):671-680.
- [4] LAM J, Delosme J. Performance of a new annealing schedule[C]//DAC. Anaheim CA, USA, 1988.
- [5] Ebeling C, McMurichie L, Hauch S A. et al. Place and route tools for the Triptych FPGA[C]// IEEE Trans. on VLSI. USA: Princeton, New Jersey, 1995.
- [6] 张相芳, 张洪梅, 田蔚风. 基于模拟回火退火的DT-MR图像平滑和估计[J]. 上海交通大学学报, 2007, 41(4):654-658.
- [7] 析火林. 基于VPR的FPGA布局算法研究与改进[D]. 武汉: 武汉理工大学, 2009.
- [8] 谈琨. FPGA互连结构与布局布线算法研究[D]. 上海: 复旦大学, 2008.
- [9] 邓庆绪, 韩瑶方, 金曦. 基于VPR算法的FPGA布局问题的研究[J]. 计算机工程与科学, 2008, 30(A1): 236-239.
- [10] Cong J, Ding Y. Flow map: an optimal technology mapping algorithm for delay optimization in lookup-table based FPGA design [J]. IEEE Transactions on Computer-Aided Design & Systems, 1994, 13(1):1-12.

作者简介:



杜星格 女, (1987-), 硕士. 研究方向为数字集成电路设计与分析.

袁小龙 男, (1973-), 副教授. 研究方向为数字集成电路设计与分析、EDA工具开发和应用.

余明 男, (1993-), 硕士研究生. 研究方向为数字集成电路设计与分析.