

多重掩模场景详细布线算法

武洁¹⁾, 吴皓莹¹⁾, 徐宁^{1)*}, 张亚东²⁾

¹⁾ (武汉理工大学信息工程学院 武汉 430070)

²⁾ (北京华大九天科技股份有限公司 北京 100102)

(xuning@whut.edu.cn)

摘要: 为了在多重掩模自动布线场景中取得缝线数目较少、总线长较小的布线效果, 提出一种基于虚拟格点 (virtual-vertex, V-V) 模型的满足多重掩模约束的详细布线算法。首先在均匀网格下构建可以支持同色及异色间距的 V-V 模型, V-V 的规模由工艺文件中定义的布线层的掩模数决定; 然后为每个多端线网生成考虑多重掩模约束的最短路径拓扑结构, 并将多端线网分解为多个双端线网; 最后基于提出的 V-V 模型结合搜索算法, 以最短路径拓扑结构作为指导, 对每个双端线网完成自动布线, 并通过代价函数严格控制缝线的引入, 保证了以最小缝线数完成线网的连接。在详细布线比赛的基准测试集及其衍生测例进行实验的结果表明, 与布线完成后再进行掩模分配的技术方案相比, 所提算法可以减少约 26% 的缝线数。

关键词: 多重掩模; 集成电路自动布线; V-V 模型

中图法分类号: TP391.41 **DOI:** 10.3724/SP.J.1089.2024.20074

Detailed Routing Algorithm for Multi-Mask Scenes

Wu Jie¹⁾, Wu Haoying¹⁾, Xu Ning^{1)*}, and Zhang Yadong²⁾

¹⁾ (School of Information Engineering, Wuhan University of Technology, Wuhan 430070)

²⁾ (Empyrean Technology Co., Ltd, Beijing 100102)

Abstract: A detailed routing algorithm is proposed to optimize the routing process in technology files with multiple masks. The algorithm is based on the virtual-vertex (V-V) model, which is designed to reduce the number of stitches and shorten the total length of the routing. The V-V model is constructed under a uniform grid and supports both uniform and non-uniform spacing. Its size is determined by the number of masks in the layer being routed. The algorithm generates a shortest path topology structure for each network with multiple terminals and splits the network into multiple two-terminal networks. The V-V model and search algorithm are then utilized to automatically route each two-terminal network with the shortest path topology as a guide. The introduction of stitches is strictly controlled through the cost function to ensure the minimum number of stitches required to connect the network. Experimental results show that the proposed algorithm can reduce the number of stitches by approximately 26% compared to the method of mask allocation after routing is completed. The algorithm satisfies the constraints of various masks and is proven to be effective in benchmark test sets and its derivative test cases of the detailed routing competition.

Key words: multiple mask; automatic routing of integrated circuits; V-V model

收稿日期: 2022-11-24; 修回日期: 2023-04-14. 基金项目: 科技创新 2030——“新一代人工智能”重大项目(2021ZD0114600). 武洁(1998—), 女, 硕士研究生, CCF 会员, 主要研究方向为电子设计自动化; 吴皓莹(1974—), 女, 博士, 副教授, 硕士生导师, CCF 会员, 主要研究方向为电子设计自动化、机器人触觉感知及模式识别、人机交互、人工智能; 徐宁(1968—), 男, 博士, 教授, 博士生导师, CCF 高级会员, 论文通信作者, 主要研究方向为电子设计自动化、计算机软件与系统; 张亚东(1987—), 男, 硕士, 高级工程师, 主要研究方向为集成电路物理设计自动化。

随着超大规模集成电路的发展不断向先进工艺节点推进, 电路的集成度越来越高, 芯片的特征尺寸越来越小, 需要更高分辨率的光刻技术才能完成芯片的制造. 极紫外(extreme ultra-violet, EUV)光刻等技术以其高分辨的优势被视作下一代光刻技术, 但其仍有尚未克服的技术难点而无法大规模生产. 由于 EUV 光刻技术的延迟, 多重掩模光刻被认为是工艺节点不断进步的解决方案, 因此多重掩模技术通过将版图上的图形分配到不同掩模版分步进行光刻, 以增加间距大小并提高分辨率, 得到特征密度更高的芯片. 为适应多重掩模场景, 要求 EDA 工具在版图设计时得到能够用于多重掩模光刻的布线结果, 但由于这一要求是在近些年集成电路发展到先进工艺节点才被提出的, 因此目前 EDA 工具大多不支持该场景. 近年来, 多重掩模场景下的布局布线问题被广泛关注并研究, 文献[1-2]对多重掩模感知的布局问题进行了研究, 文献[3-6]研究了多重掩模场景下的详细布线及版图分解问题, 文献[7]还考虑了掩模分配的均匀性问题, 并对其进行了优化.

多重掩模技术发展至今, 从芯片设计的不同完成步骤, 可以将现有工作归为 2 类: (1) 布线完成后再进行掩模分配, 此类技术方案是在布线完成之后对版图上的所有图形进行掩模分配. 值得注意的是, 由于在布线过程中没有考虑多重掩模的约束, 因此通常会导致版图图形集成度较高且版图图形分布较为复杂, 进而使得布线完成后进行掩模分配时出现无解的情况. 针对这一情况, 只能将同一个图形进行拆分再分配到不同掩模, 这将使光刻难度急剧增加, 如果拆分图形还是无法完成全部图形的掩模分配, 则需要更改版图设计, 将严重拖延设计周期. (2) 在版图设计过程中考虑掩模的分配, 将多重掩模约束转化为详细布线时需要满足的设计规则, 进行考虑多重掩模约束的详细布线, 布线完成的同时实现掩模的分配. 综合以上分析, 本文将采用在布线过程中考虑掩模分配的技术方案, 文献[5]也采用了此种技术方案并提出了一种适用于双重及三重掩模场景的格点模型, 但该模型描述较为复杂, 总格点及边的种类较多, 因此本文提出了一种新的格点模型, 提高了在实际应用中的可行性. 当前工作中被广泛应用的布线算法有迷宫、A*和线搜索等, 综合考虑算法的搜索速度及本文基于均匀网格实现的因素, 拟采用 A*算法完成搜索布线.

针对需要满足多重掩模约束的自动布线场景,

本文提出了一种基于 V-V 模型的多重掩模场景下的详细布线算法, 并与布线后考虑掩模分配的开源版图分解器的效果进行比较, 在保证布通率的前提下, 实现有效地减少缝线数的目的.

1 问题描述

1.1 相关术语

多重掩模技术(multiple patterning technology, MPT)是一种光刻技术, 用于对版图上已经分配好掩模的图形依次进行光刻, 因此这一技术要求在版图设计阶段对版图图形完成掩模的分配. 在 EDA 工具中, 将版图图形分配给不同掩模版的过程称为着色, 且不同的掩模版可以用不同的颜色进行表示.

间距用于描述布线时图形与图形之间需要保持的最小距离. 在 MPT 模式下, 间距可分为同色间距(same mask spacing, SMS)与异色间距(different mask spacing, DMS). 具体而言, SMS 是指分配在相同掩模的图形间的最小距离, DMS 是指分配在不同掩模的图形间的最小距离.

冲突用于描述布线时图形与图形之间违反设计规则检查(design rule check, DRC)的情况.

版图上同一个图形设计时由于规则冲突而导致需分配到不止一个掩模版, 即这个图形着不止一种颜色, 在不同颜色交界处称为缝线. 一种典型的 MPT 着色缝线示意图如图 1 所示^[8].

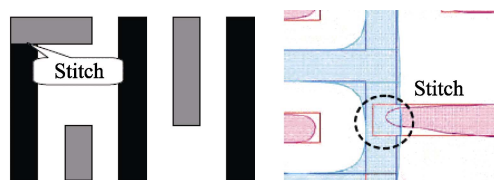


图 1 MPT 着色缝线示意图^[8]

1.2 多重掩模详细布线问题描述

MPT 详细布线的问题可描述为在详细布线过程中成功地完成对版图原有图形及布线生成图形的掩模分配, 将掩模分配转化为一种物理约束, 则 MPT 详细布线问题是考虑多重掩模约束的详细布线. 具体而言, 被分配在同一掩模上的图形, 在详细布线过程中将被着相同的颜色, 这些图形之间需要保持 SMS; 而被分配在不同掩模上的图形, 在详细布线时将被着不同的颜色, 它们之间需要保持 DMS. 因此掩模分配的过程在详细布线过程中表现为一种设计规则, 即多重掩模场景的间距

约束,满足该间距约束进行详细布线得到的布线结果,即可完成掩模分配,从而可以用于多重掩模光刻。

多重掩模场景下的详细布线有2个优化目标。

(1) 总线长最小。在详细布线过程中通常要求节约布线资源,实现总线长最小。由于MPT模式下的详细布线增加了约束,这些新增的约束将会带来绕线的情况,因此相较于基于普通模式的详细布线,MPT详细布线会增加线长。在满足多重掩模约束的前提下,如何使布线结果的总线长不会显著增大,是MPT详细布线问题的一个优化问题。

(2) 缝线数。缝线数是用于衡量多重掩模场景下详细布线效果的重要指标。如图1所示,因为缝线处很容易产生重叠误差和线端效应,带有缝线的图形会增加光刻难度,严重影响光刻效果。因此,多重掩模布线场景中要尽量减少缝线的产生,通过采用让其打孔、绕线及拆线重布等多种不同策略,从而严格控制缝线的数量。

2 本文算法

2.1 概述

本文首先提出了一种V-V模型,然后为多端线网构建能够指导其以最短路径布线的拓扑结构,对每个线网基于本文提出的格点模型探索如何在多重掩模场景下实现详细布线的性能提升,图2所示为本文的算法框图。

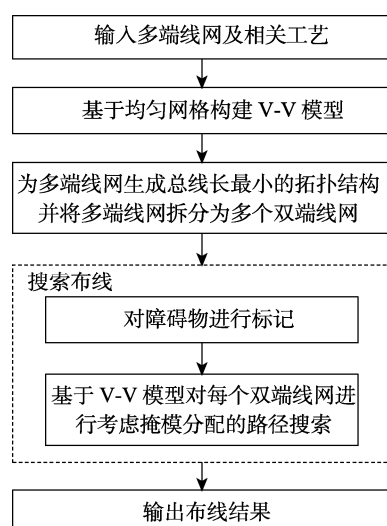


图2 本文算法框图

2.2 V-V 模型

布线算法通常分为有网格布线和无网格布线。

本文算法是基于均匀网格实现的,因此本文提出的V-V模型,也是在均匀网格基础上构建的。

在普通模式下,即不考虑掩模分配的详细布线场景中,一个可布线的二维金属层平面上,通过 x 和 y 坐标即可唯一确定这一平面上的一个格点。而在多重掩模场景中,V-V模型构建后,在一个金属层中,仅通过 x 和 y 坐标无法得到唯一的一个格点。首先读取工艺中定义金属层的掩模数,V-V模型根据掩模数确定构建V-V的规模,当金属层构建V-V模型后,一对 (x,y) 坐标指向的位置将会有多个V-V,通过再增加一重属性掩模,掩模的值大于等于1且小于等于掩模数,从而可以实现唯一确定一个V-V,其索引表示如图3所示。

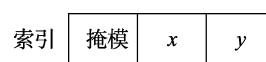


图3 基于V-V模型的格点索引

图4和图5分别为掩模数为2和3的V-V示意图及其索引表示,即双重掩模和三重掩模场景的V-V模型。

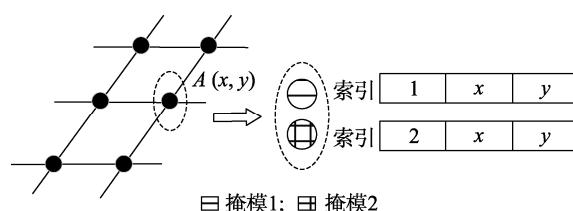


图4 双重掩模场景V-V模型

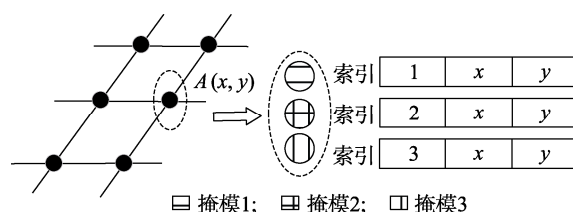


图5 三重掩模场景V-V模型

当掩模数为2时,表示当前金属层有2个掩模可选;当掩模数为3时,则表示当前金属层有3个掩模可选。以图5所示的三重掩模场景V-V模型为例,在网格中任取一点 A 分析其V-V模型,在普通网格中通过坐标 (x,y) 可以唯一确定 A 点,但构建V-V模型后,坐标 (x,y) 会指向3个V-V,只有增加掩模的值,才能唯一确定该位置的V-V,掩模的值代表该格点被分配在哪个掩模上,这样就可以实现在搜索时每搜到一个点同时完成其掩模分配,

即在详细布线过程中考虑掩模分配。

现针对三重掩模场景 V-V 模型对算法时间复杂度的影响进行分析, 由于本文拟采用 A*搜索算法完成搜索布线, 因此此处分析相较于普通网格, V-V 模型对 A*算法性能的影响。A*算法的时间复杂度本身受其代价估计函数的影响, 在最差情况下, A*退化为迪杰斯特拉算法, 其在普通网格下的时间复杂度为 $E \lg V$, 其中, E 为格点的邻接边数, V 为总格点数; 构建 V-V 模型后, 总格点数变为 $3V$, 每个 V-V 的邻接边数也变为原来的 3 倍, 因此为 $3E$, 故 A*算法在 V-V 模型下运行的最差情况时间复杂度为 $3E \lg(3V)$, 实际布线的时间将在实验结果部分进行列举和分析。

上述的 V-V 模型具有描述简洁与灵活性较强的特点, 其灵活性表现在: 首先, 不同金属层之间模型的建立是相互独立的, 在版图设计时, 对于不需要考虑掩模分配的布线层可以不构建 V-V 模型, 直接使用普通网格即可, 不同金属层可以分别定义掩模数, 互不影响; 其次, V-V 模型可以支持任意掩模数, 随着集成电路集成度的提升, 此模型有望适用于四重甚至更多重掩模场景。下文将以三重掩模场景为例, 详细介绍基于 V-V 模型的详细布线算法设计与实现。

2.3 获取最短路径拓扑结构

通常待布线的线网中会存在多端线网, 对于每个多端线网, 通过将 FLUTE 算法^[9]与多重掩模场景进行深度结合, 增加符合多重掩模约束的斯坦纳点, 为每个多端线网生成直线斯坦纳最小树, 即具有最小总线长的拓扑结构。

图 6 所示为一个 4 端线网生成的斯坦纳树, 其中, 黑色节点为线网引脚, 白色节点为添加的斯坦纳点, 这些节点共同构成了此线网的最小线长拓扑结构。在多重掩模场景中, 斯坦纳点的确定需要考虑多重掩模约束, 即检测斯坦纳点是否满足 SMS 及 DMS。

斯坦纳点的坐标通过 FLUTE 算法求出。斯坦纳点的坐标具有二维特性, 对于有多个可布线层

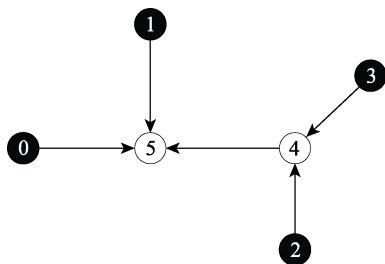


图 6 4 端线网拓扑结构

的版图设计, 还需要增加层信息才能实现唯一确定斯坦纳点。对于每个斯坦纳点, 在多重掩模场景下, 需要考虑多个因素来确定其最终的位置。如图 7 所示, 从可布线层的起始层依次遍历, 首先检查在当前层的斯坦纳点坐标处是否违反 DRC, 如该坐标是否在当前层的障碍物范围内等; 然后再遍历该斯坦纳点位置处的 V-V 模型掩模值, 检查是否有可用的掩模。如果当前布线层在该斯坦纳点位置处没有违反 DRC 并且该位置处至少有一种掩模可以用, 则当前布线层可以确定为该斯坦纳点所在的层, 否则, 斯坦纳点不能选当前层, 换另一可布线层继续检查上述条件, 直到找到一个满足条件的布线层, 确定为该斯坦纳点所在层。

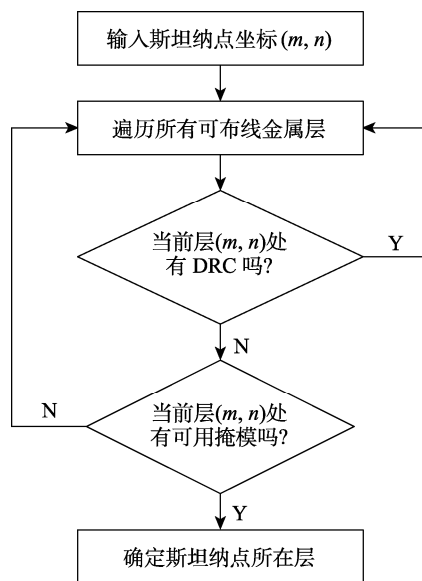


图 7 确定斯坦纳点所在层流程图

每个斯坦纳点都要经过上述判断过程确定其所在布线层, 所有斯坦纳点的层信息均确定之后, 才得到完整的拓扑结构。基于此拓扑结构, 将多端线网拆分为多个双端线网, 如图 6 中带有方向的箭头所示, 每个箭头指示一个双端线网, 从双端线网的起始点指向目标点。将该 4 端线网拆分为 5 个双端线网, 然后对每个双端线网使用基于 V-V 模型的搜索布线算法进行布线, 从而实现多端线网所有引脚的连接。

2.4 基于 V-V 模型搜索布线

本节算法是针对每个双端线网, 基于 V-V 模型并且与 A*算法结合实现的搜索布线。算法步骤如下。

输入: 点集 $(S, T) = \{(S_1, T_1), (S_2, T_2), (S_3, T_3), \dots\}$,

每个双端线网对应一个点对 (S_n, T_n) , 其中, S_n 表

示双端线网的起始点, T_n 表示双端线网的目标点.

输出: 点链 $P = \{p_1, p_2, p_3, \dots\}$, 每组点链代表一个双端线网搜索得到的路径.

Step1. 取出一个双端线网, 获取对于当前线网的障碍物图形数据, 依次对每个障碍物范围进行外扩, 标记其同色及异色冲突区域.

Step2. 对每个双端线网以 S 为起点 T 为目标点搜索布线:

Step2.1. 每搜到一个格点, 依据标记结果, 获取其可用的 V-V;

Step2.2. 依次遍历每个 V-V, 依据缝线最少原则选择一个 V-V;

Step2.3. 到达 T 点后结束搜索, 输出从 S 到 T 的一组点链.

2.4.1 标记

对于当前待布线的双端线网而言是障碍物的图形, 包括其他线网及不可布线区域, 可列为标记的对象. 标记方法是首先获取障碍物图形的边界数据, 然后外扩 SMS 得到同色冲突区域, 外扩 DMS 得到异色冲突区域, DMS 的值小于 SMS. 如图 8 所示, 对掩模 1 的图形进行标记.

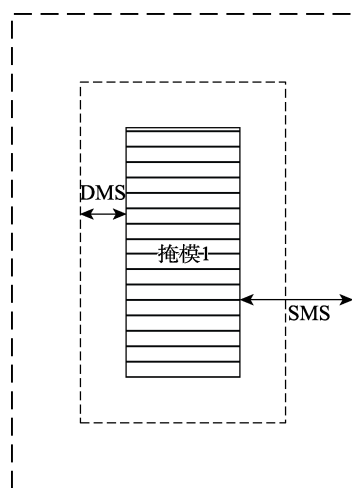


图8 标记障碍物示意图

图8中实线图形为障碍物, 虚线范围为冲突区域, 其中粗虚线矩形的范围为同色冲突区域, 位于该区域内的格点的 V-V 模型中与障碍物同色的 V-V 在搜索过程中均不可用, 即该区域内所有掩模 1 的 V-V 不可用; 细虚线矩形范围为异色冲突区域, 位于该区域范围内的格点的 V-V 模型中与障碍物异色的 V-V 在搜索过程中将不可用, 即窄虚线矩形区域内掩模 2 和掩模 3 的 V-V 不可用. 依据上述对 V-V 模型标记的结果搜索得到的路径可满足多重掩模场景中 SMS 及 DMS 的约束.

2.4.2 搜索布线

三重掩模场景的局部布线效果如图9所示, 图中包含3条线网的布线路径及掩模分配情况, 数字相同的引脚属于同一线网. 横线填充、网格填充、竖线填充分别代表掩模 1、掩模 2、掩模 3, 以其中的线网 2 为例详细介绍该路径的搜索过程.

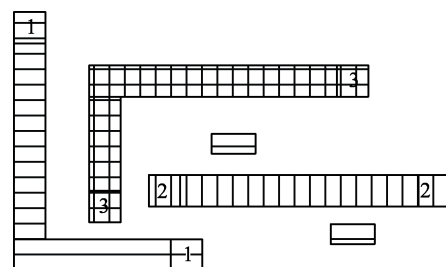


图9 三重掩模布线效果

将图9中线网2的布线环境还原, 如图10所示, 虚线区域是障碍物标记的同色冲突区域, 障碍物的掩模为掩模1, S 与 T 是线网的两端, 起始点与目标点的掩模已确定为掩模3. 路径中各点的掩模确定过程如图11所示.

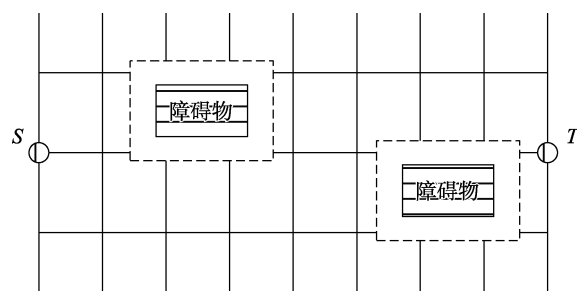


图10 双端线网示例

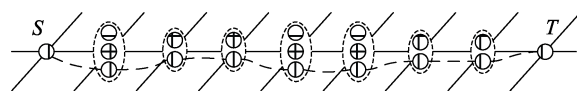


图11 路径搜索示意图

图11中将组成路径的格点进行标号, 由 S 向 1 搜索时, 由于位置 1 不在任何掩模的冲突区域内, 因此位置 1 有 3 种 V-V 可选, 根据最小缝线数原则, 位置 1 会选择与 S 相同的掩模, 即位置 1 会选择掩模 3 的 V-V; 由 1 向 2 搜索时, 由于位置 2 位于掩模 1 障碍物的冲突区域内, 相应掩模的 V-V 不可用, 即掩模 1 的 V-V 不可用, 因此位置 2 只有 2 种 V-V 可选; 类似地, 根据缝线最小原则, 位置 2 将

选与 1 相同掩模的 V-V 位置 3, 6 和 7, 由于它们位于掩模 1 的冲突区域内, 相应掩模的 V-V 不可用, 因此这些点只有 2 种 V-V 可选, 而位置 4, 5 不在任何掩模的冲突区域内, 因此有 3 种 V-V 可选. 到达 T 点后结束搜索, 搜索结果如图 11 中的虚线所示, 每个点的掩模都被唯一确定, 得到从 S 到 T 的一组点链, 形成该线网的路径, 即为图 9 中线网 2 的布线结果.

3 实验及结果分析

本文算法采用 C++ 语言编写, 在 Linux 环境下运行, 通过对测试用例进行测试, 测例由比赛用例及工业单元测试用例组成. 为进一步研究 V-V 模型对布线算法性能的影响, 将 A* 算法在普通网格与 V-V 模型下的布线时间进行对比; 另外, 将采用本文算法得到布线结果同开源版图分解器进行性能对比, 此处没有与同样是在布线过程中考虑掩模分配的文献[5]做比较, 是因为该文献的算法与测例均不是开源的, 而选择开源的版图分解器进行对比, 则能够使用相同的测例测试不同算法的效果. 由于版图分解器采用的是布线后进行掩模分配的技术方案, 因此其输入是布线完成的版图数据, 因此需要先用普通模式进行布线, 即布线过程中不考虑多重掩模约束, 得到的布线结果用于版图分解器进行掩模分配.

测试用例的相关参数如表 1 所示, 其中展示了 5 个测试用例的芯片尺寸、布线层数及线网数量.

表 1 测试用例参数

测试用例	芯片尺寸/mm ²	布线层数	线网数量
test1	0.010×0.008	2	81
test2	0.090×0.090	2	892
test3	0.195×0.195	9	20 471
test4	0.148×0.146	7	12 640
test5	0.073×0.089	5	3 110

表 2 所示为 A* 算法分别在普通网格与 V-V 模型下的布线时间, 可以看出, 相比普通网格, A* 算法在 V-V 模型下平均增加了 8.74% 的布线时间. 用时增加的原因有 2 个方面: (1) V-V 模型构建后, 导致格点规模的增大造成布线用时增加; (2) V-V 模型下的布线需要满足多重掩模约束, 求解的问题复杂度更高, 导致用时增加. 尽管在 V-V 模型下运行的算法布线时间会增加, 但在布线的同时完成了掩模分配, 并且当线网规模增大后, 依然可以在

表 2 布线时间统计

测试用例	布线用时/s		用时增加/%
	普通网格	V-V 模型	
test1	4.73	4.90	3.59
test2	32.56	34.94	7.31
test3	1 824.64	2 091.40	14.62
test4	675.03	744.96	10.36
test5	123.90	133.56	7.80
平均	532.17	601.95	8.74

较合理的时间内完成布线.

表 3 所示为普通布线模式与 MPT 布线模式的布线结果总线长, 图 12 为缝线数对比图, 包含 5 个测例的结果, 黑色为版图分解器的实验结果, 白色为本文算法的实验结果. 其中, 版图分解器缝线数是先用普通布线模式, 基于该布线结果使用版图分解器进行掩模分配依次得到每个测例的缝线数; 而本文算法缝线数是通过基于 V-V 模型的 MPT 布线模式所得的缝线数. 通过性能结果可以看出, 尽管使用本文算法得到的最终版图设计结果会平均增加 2.6% 的线长, 但随着线网规模逐步增大, 线网数越多的布线场景复杂度将急剧增加, 导致测例的缝线数增多, 与版图分解器的布线后再进行掩模分配的技术方案相比, 使用本文基于 V-V 模型在详细布线过程中考虑掩模分配的算法可以降低约 26.09% 的缝线数.

表 3 布线结果总线长统计

测试用例	总线长/ μm		线长增加/%
	普通布线模式	MPT 布线模式	
test1	1 235	1 255	1.6
test2	10 828	11 072	2.2
test3	189 019	195 875	3.5
test4	103 924	107 360	3.2
test5	34 195	35 108	2.6
平均	67 840	70 134	2.6

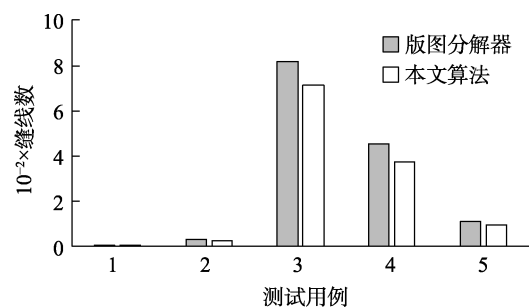


图 12 5 个测例的缝线数对比

4 结 语

本文提出了一种适用于多重掩模场景的详细布线算法, 基于均匀网格构建 V-V 模型, 基于该模型完成详细布线及掩模分配. V-V 模型格点总数规模小且形成的边的种类少, 没有模型内的边, 形式简洁, 因此可灵活与多种布线算法结合解决考虑多重掩模约束的详细布线问题. 另外本文列举了三重掩模应用场景, V-V 模型理论上还可以支持四重, 甚至更多重掩模的分配. 为获得总线长更优的布线结果, 节省布线资源, 对于多端线网, 本文采用直线斯坦纳最小树生成最短路径的拓扑结构. 尽管实际路径搜索过程中会存在障碍物等因素, 导致最终形成的路径与拓扑结构相比无法完全一致, 但最短路径的拓扑结构可对多端线网的布线起到指导作用, 得到兼顾总线长及缝线数的布线结果.

最后, 对多端线网进行拆分再布线, 每个双端线网的路径搜索是依次进行的, 如果考虑进一步采用多源搜索, 可实现有效地降低运行时间的目标.

参考文献(References):

- [1] Chien H A, Chen Y H, Han S Y, *et al.* On refining row-based detailed placement for triple patterning lithography[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2015, 34(5): 778-793
- [2] Lin Y B, Yu B, Xu B Y, *et al.* Triple patterning aware detailed placement toward zero cross-row middle-of-line conflict[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2017, 36(7): 1140-1152
- [3] Li W, Ma Y Z, Sun Q, *et al.* OpenMPL: an open-source layout decomposer[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2020, 40(11): 2331-2344
- [4] Zhong W, Hu S X, Ma Y Z, *et al.* Deep learning-driven simultaneous layout decomposition and mask optimization[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2022, 41(3): 709-722
- [5] Ma Q, Zhang H B, Wong M D F. Triple patterning aware routing and its comparison with double patterning aware routing in 14nm technology[C] //Proceedings of 49th ACM/EDAC/IEEE Design Automation Conference. Los Alamitos: IEEE Computer Society Press, 2012: 591-596
- [6] Liu Z Q, Liu C W, Young E F Y. An effective triple patterning aware grid-based detailed routing approach[C] //Proceedings of the Design, Automation & Test in Europe Conference & Exhibition. Los Alamitos: IEEE Computer Society Press, 2015: 1641-1646
- [7] Chen K J, Fang S Y. Printability enhancement with color balancing for multiple patterning lithography[J]. IEEE Transactions on Emerging Topics in Computing, 2019, 7(2): 244-252
- [8] Cho M, Ban Y C, Pan D Z. Double patterning technology friendly detailed routing[C] //Proceedings of the IEEE/ACM International Conference on Computer-Aided Design. Los Alamitos: IEEE Computer Society Press, 2008: 506-511
- [9] Chu C, Wong Y C. FLUTE: fast lookup table based rectilinear Steiner minimal tree algorithm for VLSI design[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2008, 27(1): 70-83