

上海交通大学

SHANGHAI JIAO TONG UNIVERSITY

学士学位论文

BACHELOR'S THESIS



论文题目：基于 HEVC 运动估计模块的三维
电路划分技术研究

学生姓名：_____ 丁勇铭

学生学号：_____ 5122119046

专 业：_____ 微电子学

指导教师：_____ 何卫锋

学院(系)：_____ 电子信息与电气工程学院

基于 HEVC 运动估计模块的三维电路划分技术研究

摘要

HEVC 是最新一代的视频编码标准，相比于上一代的 H.264 的编码标准，压缩率提高了一倍以上。运动估计是 HEVC 编码器中重要的一步，它通过消除运动图像中时间域的相关性来达到压缩图像的目的。由于运动估计模块是 HEVC 编码器的硬件实现中最复杂的模块之一，为了提高性能、缩小面积，考虑使用三维集成电路技术对其进行硬件实现。因此需要对其进行电路划分，使得划分后的各层面积比较均匀，层与层之间的 TSV 数量最少，面积利用率提高。

本课题针对 HEVC 的运动估计模块，设计了面向三维集成的电路划分算法。包括基于 BFS 的初始划分算法和迭代改进的优化划分算法。并针对 HEVC 的运动估计模块，设计了三维集成电路划分算法评估平台，主要功能包括解析电路网表文件、解析 DC 综合面积报告、建立超图数据结构描述电路单元互连关系、对 hierarchy 结构的运动估计模块电路按照模块大小分布进行向下展开、多种方式生成初始划分、施加划分优化算法与划分结果评估。对 HEVC 中 IDCT 模块和 ME 模块使用设计的三维集成电路划分算法进行了划分，从割边数、面积利用率等方面评估了划分结果。IDCT 最适宜的划分层数为三层，与初始划分相比，割边数减少了 92.55%，面积利用率达到 97.38%，提高了 10.46%。对于 ME 模块，选择 $K=3$, $\alpha=0.5$ 作为 ME 模块的最优划分，与初始划分相比，割边数减少了 42.98%，面积利用率达到 95.51%，提高了 17.97%。

关键词：HEVC, 运动估计, 三维集成电路, 电路划分

3D IC PARTITIONING TECHNIQUE BASED ON HEVC

MOTION ESTIMATION MODULE

ABSTRACT

HEVC is the latest generation of video coding standards, compared to the last generation of H.264 coding standards, compression rate increased by more than doubled. Motion estimation is an important step in the HEVC encoder, which can achieve the purpose of compressing image by eliminating the correlation of time domain in the moving image. As the motion estimation module is one of the most complex modules in the hardware implementation of HEVC encoder, in order to improve the performance and reduce the area, the hardware implementation of 3D IC technology is considered. Therefore, it is necessary to make the circuit partitioning, so that the area of each layer is more uniform, and the number of TSV between layers is the least.

In this paper, a 3D IC partitioning algorithm is designed for the motion estimation module of HEVC. Including the initial partition algorithm based on BFS and the iterative optimization algorithm. And for HEVC motion estimation module, design a 3D IC partitioning algorithm evaluation platform, the main functions include analytic network file, comprehensive analysis of DC area report, the establishment of hypergraph data structure to describe unit circuit interconnection relations, the hierarchy structure of motion estimation circuit module according to the size of the module distribution was launched down, a variety of ways to generate an initial partition, applying partition algorithm and classification of the results of the assessment. The IDCT module and ME module using the design of three-dimensional integrated circuit partitioning algorithm is partitioned, from the number of cut edges, area utilization, and other aspects of the evaluation of the partition results. IDCT the most appropriate division of the number of layers for the three layer, compared with the initial division, cutting edge number decreased by 92.55%, the area utilization rate reached 97.38%, increased by 10.46%. For the ME module, the selection of K=3, alpha =0.5 as the optimal partition, compared with the initial division, the number of cut edges was reduced by 42.98%, the area utilization rate reached 95.51%, increased by 17.97%.

Key words: HEVC, motion estimation, 3D IC, IC partition

目 录

第一章 绪论-----	1
1.1 研究背景-----	1
1.1.1 三维集成电路简介-----	1
1.1.2 HEVC 视频编码标准-----	2
1.1.3 HEVC 的运动估计算法-----	2
1.2 研究意义与研究目标-----	3
1.3 研究内容-----	3
1.4 本章小结-----	4
第二章 三维集成电路划分算法研究现状-----	5
2.1 三维集成电路与划分-----	5
2.2 三维集成电路划分问题研究现状-----	7
2.3 本章小结-----	11
第三章 基于 HEVC 运动估计模块的三维集成电路划分算法设计-----	12
3.1 三维集成电路电路网表的数学模型-----	12
3.2 超图数据结构的建立-----	13
3.3 hierarchy 结构电路展开-----	14
3.4 初始划分的生成-----	15
3.5 移动增益模型-----	16
3.6 算法流程-----	16
3.7 三维集成电路划分评估平台-----	19
3.7.1 电路网表文件的解析-----	19
3.7.2 面积报告的解析-----	20
3.7.3 施加划分优化算法，评估划分结果-----	21
3.8 实验结果与分析-----	22
3.9 本章小结-----	23
第四章 结论-----	27
参考文献-----	28
谢辞-----	29

第一章 绪论

1.1 研究背景

1.1.1 三维集成电路简介

20世纪以来，人类社会逐渐进入信息时代，集成电路是信息时代的社会发展的基石，在我们的日常生活中无处不在。40年前，身为Intel公司的创始人之一的戈登·摩尔提出了在集成电路领域广为人知的摩尔定律：单位面积的集成电路上容纳的元器件数目，每隔一年半翻一番。随着集成电路的设计方法、制造工艺的一步步发展和进步，芯片的集成度不断提高，器件数量急剧增加。在芯片尺寸越来越小的趋势下，各种寄生效应引起的问题越来越严重，进一步提高芯片的集成度将会受到物理极限的约束。在传统的二维芯片上，由于模块之间的距离较远，模块间互连线的过长导致占用很多面积而且其互连线延迟已经成为提升芯片速度的主要瓶颈，严重阻碍了集成电路进一步提高集成度和性能。因此，三维集成电路技术成为了延续摩尔定律的新思路，并受到了业界的广泛关注。

三维集成电路是指拥有数层器件结构的集成电路，将多层芯片在垂直方向上进行堆叠，从而成倍地提高了单位面积上的器件数量，实现了提高集成度的目标。同时，在三维集成电路中，层与层之间使用穿透硅基质的硅通孔(Through Silicon Via, TSV)互连，使得原本在二维芯片上，距离较远的模块之间的长互连线改为垂直互连线，大大缩短互连线的长度，从而解决了长互连线带来的延迟问题，并且有可能实现并行信号的处理，有助于进一步提高集成电路的性能。与原先二维平面化的布局空间相比，三维集成电路增加了一个纵向的维度，从而在布局时不但可以控制各个模块的摆放位置，还可以选择不同的摆放层。所以，三维集成电路与传统二维电路相比的优点体现在更小的面积、更高的集成度、更短的线长、更高的工作频率和更低的功耗。

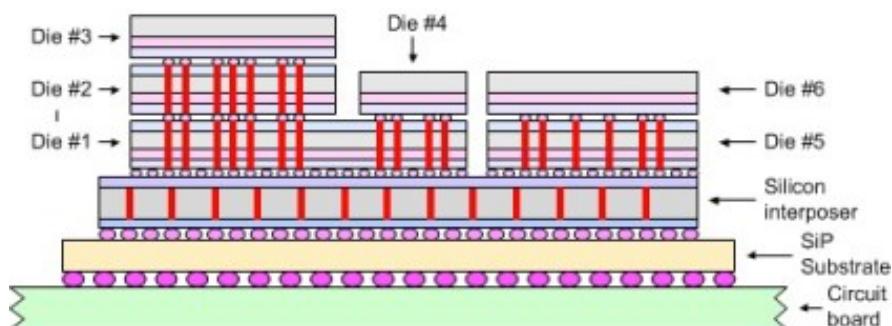


图 1-1 三维集成电路模型图

三维集成电路技术给芯片设计行业带来了新的挑战，如不同层的器件之间如何进行布局布线成为了芯片设计工程师的难题。因为三维集成电路中增加了一个纵向的维度，现有的EDA软件并不支持多层次间的布局布线，工程师们只能考虑使用现有的二维芯片的布局布线工具完成工作，所以需要提前将芯片的各个模块进行分层，在每层使用现有的工具进行布局布线操作。除了一般的互连线之外，三维芯片中的TSV是连接层与层之间的重要通道，包含信号、时钟、电源等信息，所以控制TSV的数量，安排TSV的摆放位置也是三维集成电路的设计中的一大难题。

1.1.2 HEVC 视频编码标准

随着互联网产业的发展和移动互联网的兴起，人们对在线视频的需求日益增加，数字视频在通信、娱乐、军事侦察、抗震救灾等方面的应用中扮演着关键性的角色。然而，我们在实际生活中接触到的视频，其实都是被压缩过的。因为未经压缩处理的原始视频文件，其数据量非常惊人，无法满足实际传播和存储的需求。因此，视频编码，也称视频压缩，是数字视频得以广泛应用的关键技术，其目的是尽可能去除原始视频文件中的冗余数据，减少表征视频的数据量。近年来，随着高分辨率显示器和录像设备的普及，超高清视频（分辨率达 $4K \times 2K$ 、 $8K \times 4K$ ）逐渐走入人们的视野，视频编码技术受到了巨大的挑战。此外，各式各样的移动视频应用层出不穷，视频广播、远程监测、医学成像、便携摄影等等都已走入人们的生活。视频应用的多样化和高清化趋势对视频压缩性能提出了更高的要求。

为此，2010 年 4 月，国际电信联盟电信标准化部门（International Telecommunication Union-Telecommunication Standardization Sector, ITU-T）的视频编码专家组（Video Coding Experts Group, VCEG）和国际标准化组织（International Organization for Standardization, ISO）/国际电工委员会（International Electrotechnical Commission, IEC）的动态图像专家组（Moving Picture Experts Group, MPEG）组建视频编码联合组（Joint Collaborative Team on Video Coding, JCT-VC），联手制定新一代视频编码标准——HEVC（High Efficiency Video Coding）^[11]。HEVC 作为最新一代的视频编码标准，其被设计以替代 H.264/AVC 编码标准，因此也称为 H.265 编码标准。与 H.264 编码标准相比，使用 HEVC 编码标准可以使视频在画面质量不变的情况下，压缩率提高一倍以上。这意味着用户可以在不增加流量消耗的情况下享受更高品质的视频内容，对视频网站运营方而言，带宽成本是运营成本的最大组成部分，采用 HEVC 编码标准可以节省一半以上的带宽消耗，可以给消费者带来更好的画质和用户体验。

1.1.3 HEVC 的运动估计算法

由于在大多数原始视频文件中，图像之间的相关性较强，相邻图像内容非常相似，其背景画面变化极小，数据冗余较大，因此不需要对每帧图像的全部信息都进行编码，只需要将当前图像中的运动物体的运动信息传给解码器。此时，解码器利用前一图像的内容以及当前图像的运动信息即可恢复当前图像，通过这种方式能有效地减少视频文件的大小^[13]。在 HEVC 的编码器中，运动估计（Motion Estimation, 简称 ME）/ 运动补偿（Motion Compensation, 简称 MC）模块便是要找到运动图像在先前帧中的运动位移，通过消除运动图像中时间域的相关性达到压缩图像的目的。为了更加灵活、高效地表现画面细节，处理运动画面，HEVC 编码标准相比前代 H.264/AVC 中做了许多改进，其中与运动估计算法关系比较密切的有如下几部分：

- (1) 编码单元。在 H.264/AVC 中，编码块的大小是固定的。而在 HEVC 中，编码块的大小是可变的。比如亮度编码块最大为 64×64 ，最小为 8×8 。一个亮度编码块和一个色度编码块和相关的句法元素共同组成一个编码单元。在 HEVC 中，一张图片可以被划分为数个互不重叠的树形编码单元。在每个树形编码单元内部，采用基于四叉树的循环分层结构，同一层的编码单元深度相同。一个树形编码单元可能包含一个或多个编码单元。这种更加灵活的编码单元表示方法，与 H.264/AVC 中固定的宏块大小（ 16×16 ）相比，在相对比较平缓的区域，使用较大的编码单元可以极大地提高编码效率，减少所用的比特数；在细节相对比较丰富的区域，使用较小的编码单元，从而更好地处理图片的细节，使得复杂运动图片的预测更加准确。
- (2) 预测单元。预测单元规定了编码单元的预测模式，包括帧内预测和帧间预测。一个编码单元可以划分成一个或者多个预测单元，在 HEVC 中，一个 $2N \times 2N$ 的编码单元划分成预测单元有 8 种可选方式，如图 1-2 所示。包括 4 种对称方式（ $2N \times 2N$, $2N \times N$, $N \times 2N$, $N \times N$ ），以及 H.264 中没有的四种非对称模式（左右两种 $2N \times 1.5N$, 上下两种 $1.5N \times 2N$ ）。新增加的非对称模式可以使得预测单元的划分选择更加多样化，提高了预测的精度。

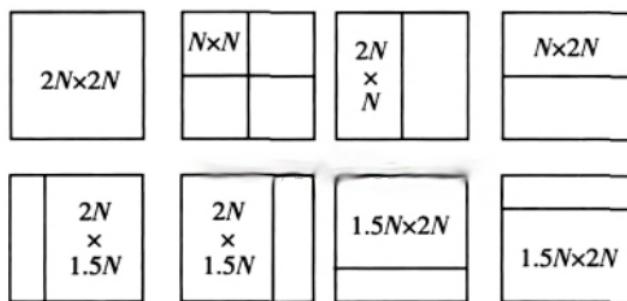


图 1-2 预测单元划分模式示意图

- (3) Merge 技术是 HEVC 在运动向量的预测方面提出的一个新技术。Merge 技术会为当前的预测单元建立一个运动向量候选列表，包含 5 个候选运动向量和对应的参考图像，通过遍历这 5 个候选运动向量，计算率失真代价，选取率失真代价最小的一个运动向量作为该 Merge 模式的最优运动向量。编码器只需传输最优运动向量在候选列表中的索引即可，大大节省了运动信息的编码比特数^[12]。
- (4) 高级运动向量预测 (AMVP) 技术也是 HEVC 新提出的运动向量预测技术。AMVP 技术利用空间域和时间域上运动向量的相关性，为当前预测单元建立了预测运动向量列表。编码器从中选择出最优的预测运动向量，然后对运动向量进行差分编码。解码器通过建立相同的列表，仅需要运动向量残差 (MVD) 与预测运动向量在该列表中的索引即可计算出当前预测单元的运动向量，大大节省了运动信息的编码比特数。

1.2 研究意义与研究目标

由于运动估计算法在整个 HEVC 编码过程中占据了百分之 60 以上的运行时间，运动估计算法在 HEVC 编码器的硬件实现中是最重要、最复杂的模块之一，为了提高 HEVC 编码器硬件实现的性能、减少时延、降低功耗、缩小面积，考虑使用三维集成电路技术对 HEVC 编码器的运动估计算法进行硬件实现。由于三维集成电路与传统二维电路相比增加了一个纵向的维度，在对电路进行布局布线之前，需要确定每个器件的摆放层，因此需要对原始的二维电路进行划分，根据划分结果得到每层的器件信息之后，才能进行后续的布局布线工作。

因为划分的结果对整体电路的实现有很大影响，不同划分之间的连线数，即割边的数量决定了层与层之间 TSV 的数量，过多的割边数使得 TSV 所占面积过大，从而影响电路的整体性能。而且进行三维集成电路物理实现时的总面积要按照最大一层的面积进行计算，所以划分的结果最好使得各层器件的面积比较均匀以减少电路的总面积。

研究目标为针对给定的 HEVC 运动估计算法结构，开发面向三维集成的电路划分算法，以划分间之间的互连线最小为主要优化目标，同时使得最大层面积较小，各层面积比较均匀，提高三维集成电路实现时的面积利用率。

1.3 研究内容

本课题针对 HEVC 的运动估计算法，在满足面积平衡的约束条件下，以最小化割边数、提高三维集成电路的面积利用率为优化目标，进行面向三维集成的电路划分算法的设计与实现。具体的研究内容包括：

- (1) 了解和熟悉 HEVC 编码标准，HEVC 编码标准在运动估计算法方面相比 H.264/AVC 标准所做的改进，HEVC 编码器运动估计算法的具体流程和实现方式以及运动估计算法的电路结构。
- (2) 对针对三维集成电路划分问题的算法研究现状进行调研，按照划分策略对不同算法进行分类，包括迭代改进算法、线性规划算法、模拟退火算法和多级划分算法等三维集成电路划分算法。并分析各种算法之间性能、效果和适用范围的不同。

- (3) 针对 HEVC 的运动估计模块，设计了面向三维集成的电路划分算法。包括使用超图数学模型抽象电路网表、建立模块的移动增益模型、基于 BFS 的初始划分算法和迭代改进的优化划分算法。
- (4) 针对 HEVC 的运动估计模块，设计了三维集成电路划分算法评估平台，主要功能包括解析电路网表文件、解析 DC 综合面积报告、建立超图数据结构描述电路单元互连关系、对 hierarchy 结构的运动估计模块电路按照模块大小分布进行向下展开、多种方式生成初始划分、施加划分优化算法与划分结果评估。
- (5) 对 HEVC 编码器中 IDCT 模块和 ME 模块，在搭建好的三维集成电路划分评估平台中，使用设计的三维集成电路划分算法进行了划分，从割边数、面积利用率等方面评估了划分结果。

1.4 本章小结

本章介绍了本课题的研究背景、研究意义、研究目标和研究内容。背景部分包括三维集成电路技术和 HEVC 编码技术的背景，以及 HEVC 编码在运动估计算法上进行的改进。运动估计算法在整个 HEVC 编码过程中占据了百分之 60 以上的运行时间，运动估计模块在 HEVC 编码器的硬件实现中是最重要、最复杂的模块之一，为了提高 HEVC 编码器硬件实现的性能、减少时延、降低功耗、缩小面积，考虑使用三维集成电路技术对 HEVC 编码器的运动估计模块进行硬件实现。所以研究目标为针对给定的 HEVC 运动估计电路结构，开发面向三维集成的电路划分算法。研究内容概括得介绍了本课题所做的各方面工作，将由以下的篇幅具体描述。

第二章 三维集成电路划分算法研究现状

2.1 三维集成电路与划分

随着集成电路工艺的不断发展，对集成电路芯片的集成度、功耗、时延性能等要求也越來越高。传统的二维集成电路芯片，在不断地提高晶体管数量的同时，芯片的面积也不可避免地逐渐增大。面积的增大使得芯片上的互连线长度增加，从而导致芯片的性能降低，功耗加大。三维集成电路（3D IC）技术是一种电路设计的新思路，结合三维空间的概念，将传统的二维集成电路 die 在垂直方向上堆叠起来。使用这种方式，可以有效地减少电路的投影面积，芯片的集成度将成倍提高。而且因为每一层电路面积相比二维集成电路的面积大大缩小，芯片上的互连线长度也有所缩短，从而减少了互连线的延迟，降低了功耗。在三维集成电路中，层与层之间使用 TSV 进行互连，然而因为 TSV 本身有一定的面积占用，如果 TSV 的数量太多，会使得三维集成电路的面积增大、降低面积利用率、限制了芯片集成度的提升，而且对后续的布局布线工作也有一定的限制，导致三维集成技术带来的好处大打折扣。

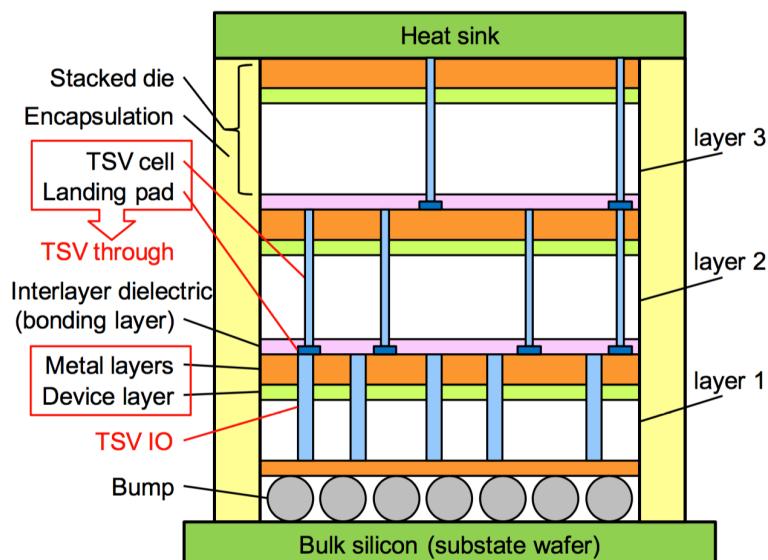


图 2-1 使用 TSV 进行垂直连接的三维集成电路

三维集成电路划分在三维集成电路实现过程中是非常重要的一步，它主要是指在三维集成电路的设计过程中，将整个系统划分成 N ($N \geq 2$) 个较小的系统，将各个元器件分配至不同的集合中。在划分的过程中，一般需要将逻辑关系比较紧密的电路模块划分在同一部分，而将之间连线比较稀疏的模块划分在不同部分。这样可以使得不同划分之间的连线数减少，如图 2-2 所示，反应在三维集成电路的物理实现中即层与层之间的 TSV 数量较少，有助于减少三维集成电路的面积，并使得划分后的电路获得更好的延迟性能。

虽然在二维集成电路中，电路划分的问题也广泛存在。但是由于三维集成电路的所有互连线和外部信号的连接都通过同一层，三维集成电路的划分问题与传统的二维集成电路划分问题有显著不同。从二维集成电路划分算法得到的结果往往不能保证划分质量的好坏，因为一条割边它所连接的两个电路层中间有可能隔着其他电路划分层，为了实现这条割边可能需要多个 TSV，如图 2-3 所示。因此，传统的二维集成电路中最小割为优化目标的划分算法不能准确地解决三维集成电路中最少化 TSV 数量的划分问题。

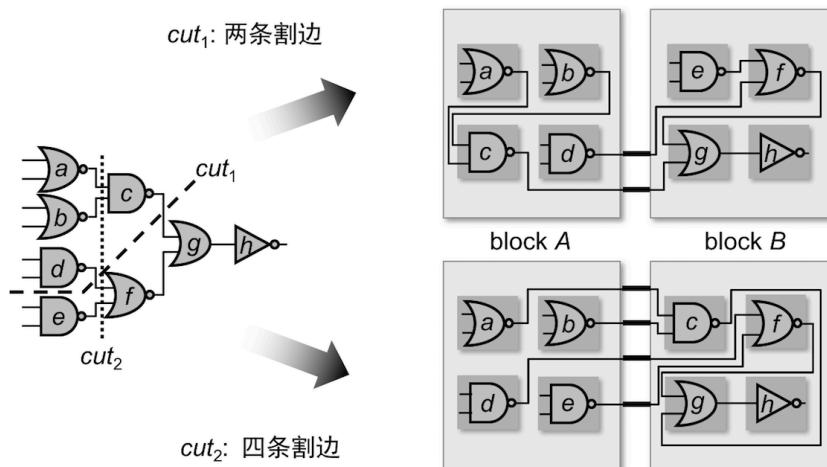


图 2-2 一个电路示例，两种划分方式

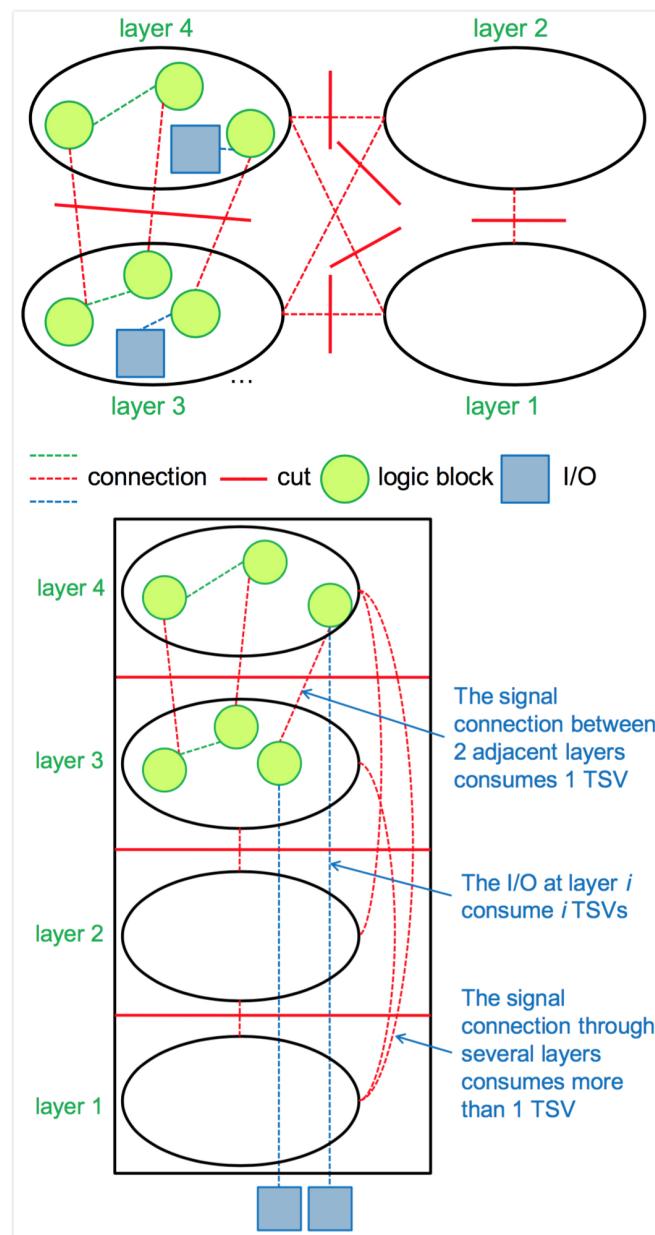


图 2-3 二维电路与三维电路划分时的区别

2.2 三维集成电路划分问题研究现状

三维集成电路划分问题是 NP 困难问题，也就是说，随着问题规模的线性增加，寻找最优解耗费的时间比其他多项式复杂度的算法问题所需要的时间增长快得多，因此很难在有限的时间内得到最优解。尽管如此，国内外的学者通过不断地研究实验，提出了许多性能不错的划分算法，可以在有限的时间内得到一个可行解。在针对三维集成电路划分问题上，依据划分策略，大致可以分为迭代改进算法、线性规划算法、模拟退火算法和多级划分算法。

H. S. Ye, M. C. Chi 和 S. H. Huang 在 2010 年提出了使用 Fiduccia-Mattheyses (FM) 算法的概念去解决三维集成电路划分问题^[1]，该算法在处理一个拥有 85013 个模块的测试电路时，在只产生了 142 个 TSV 的情况下将其分成 5 层，同时额外的面积开销只有 0.1%。该算法流程如图 2-4 所示，包含了初始划分和迭代改进两部分。FM 算法是一个典型的通过不断地迭代改进来得到最佳划分的算法，在 1982 年由 C. M. Fiduccia 和 R. M. Mattheyses 一起提出^[2]。FM 算法在给定了初始划分的情况下，计算每个顶点的移动增益，考虑每个模块的面积，模块能否需要考虑是否满足面积平衡约束。FM 算法每轮选择拥有最大移动增益的顶点进行移动，并将其锁定，不断迭代，直到割边数没有减少，算法结束，图 2-5 和图 2-6 展示了 FM 算法在一个轮次中的运行示例。2012 年，Chang H L, Lai H C 和 Hsueh T Y 在考虑了供电 TSV 的情况下提出了一个三维集成电路划分算法^[3]，在 2011 年台湾的 IC/CAD 比赛上，该算法的划分结果相比另外三支队伍分别有 351%，7% 和 39% 的 TSV 数量减少，同时面积利用率也较高。该算法将电路中的 TSV 分为传输信号的 TSV 和用来供电的 TSV，因为层数较高的电路供电需要在底层引入额外的 TSV，所以该算法通过将耗能较低的电路安排在较高的层数来减少供电 TSV 的数量，从而优化了划分结果。该算法的迭代改进过程与 FM 算法类似，流程如图 2-7 所示。

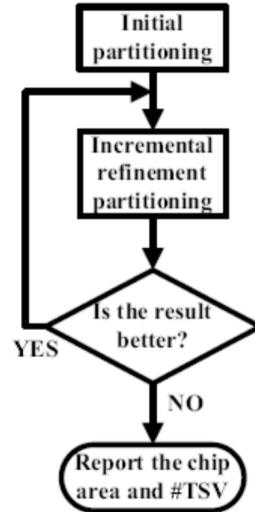


图 2-4 H. S. Ye 等提出的三维电路划分算法流程图

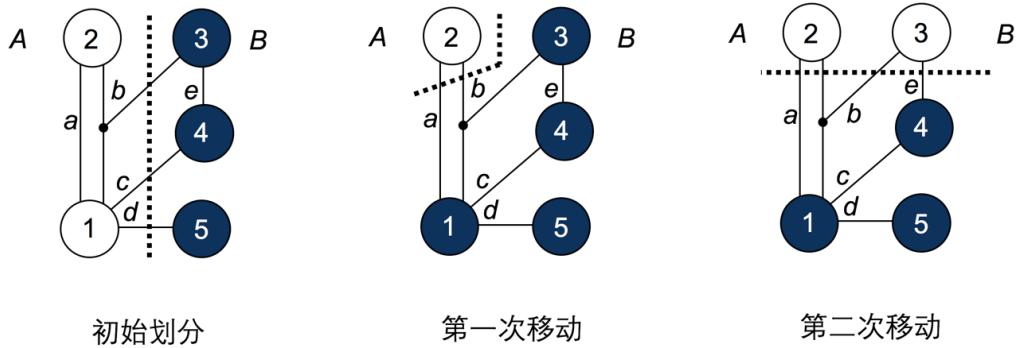


图 2-5 FM 算法示例图 1

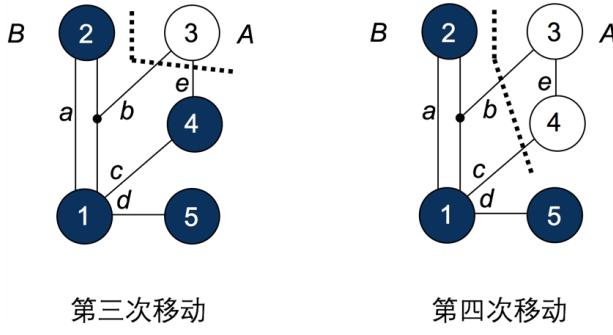


图2-6 FM算法示例图2

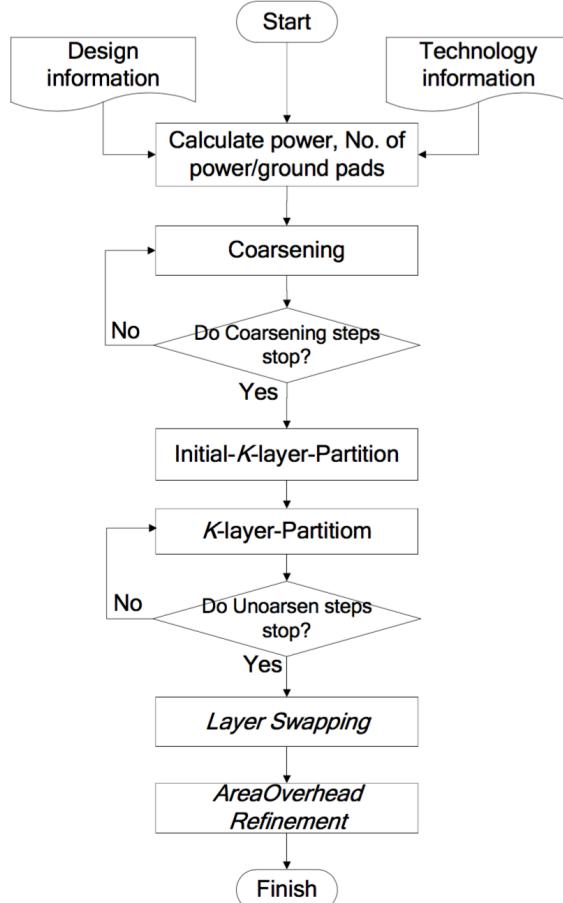


图2-7 Chang H L等提出的考虑供电TSV的划分算法流程图

HR Jiang 在 2009 年第一次提出使用整数线性规划 (integer linear programs, ILPs) 的方法对三维集成电路进行划分^[4], 目标函数即 TSV 的数量表示为:

$$\sum_{ei \in E} \sum_{p=1 \dots k} y_{i,p} \quad \text{式 (2-1)}$$

$y_{i,p}$ 是一个 0 或 1 的值, 其值的计算公式如下:

$$\begin{aligned}
 y_{i,p} = & [(x_{i1,1} \text{OR } x_{i1,2} \text{OR } \dots \text{OR } x_{i1,p-1}) \text{OR } \\
 & (x_{i2,1} \text{OR } x_{i2,2} \text{OR } \dots \text{OR } x_{i2,p-1}) \text{OR } \dots \text{OR } \\
 & (x_{iq,1} \text{OR } x_{iq,2} \text{OR } \dots \text{OR } x_{iq,p-1})] \text{AND } \\
 & [(x_{i1,p} \text{OR } x_{i1,p+1} \text{OR } \dots \text{OR } x_{i1,l}) \text{OR } \\
 & (x_{i2,p} \text{OR } x_{i2,p+1} \text{OR } \dots \text{OR } x_{i2,l}) \text{OR } \dots \text{OR } \\
 & (x_{iq,p} \text{OR } x_{iq,p+1} \text{OR } \dots \text{OR } x_{iq,l})], 1 \leq p \leq l
 \end{aligned} \quad \text{式 (2-2)}$$

图 2-8 列出了 ILP 公式中各个变量代表的物理意义。

Category	Variable	Description
General	l	The number of layers l in a 3D IC.
		The external world is viewed as layer 0.
	p	Cut p is the cut between layers $p-1$ and p .
	G	Input design is a hypergraph $G = (V, E)$
	v_i	Vertex $v_i \in V$ is a logic block or an I/O terminal.
	e_i	Hyperedge $e_i = \{v_{i1}, v_{i2}, \dots, v_{iq}\} \in E$ is a q -pin net.
Layer assignment	$x_{i,j}$	$x_{i,j}$ is a 0-1 integer variable associated with vertex v_i . $x_{i,j} = 1$ if v_i is assigned to layer j ; otherwise, $x_{i,j} = 0$.
		For an I/O terminal v_i , $x_{i,0} = 1$, while for a logic block v_j , $x_{j,0} = 0$.
TSV	$y_{i,p}$	$y_{i,p}$ is a 0-1 integer variable associated with net e_i . $y_{i,p} = 1$ if net e_i introduces a TSV to cut p , between layers $p-1$ and p , $1 \leq p \leq l$; otherwise, $y_{i,p} = 0$.
Area	s_i	s_i is the area of vertex v_i .
		$s_i = 0$ if v_i is an I/O terminal.
	t	t is the area of a single TSV.
	A	$A = \sum_{v_i \in V} s_i$, total area of all vertices.
	α	α is the perfectly balanced area for each layer, $\alpha = A/l$.
	ϵ	ϵ is the α -bounded ratio of the deviation from the perfectly balanced area, $0 < \epsilon < 1$.
		The α -bounded area for each layer is subject to the range $[(1-\epsilon)\alpha, (1+\epsilon)\alpha]$.

图2-8 IPL方程中各变量的物理意义

该基于ILP的三维集成电路划分算法比基于迭代改进FM算法的hMetis电路划分工具,划分结果的割边数有5%左右的提升,缺点是当划分的模块数量较多时,ILP的执行时间会很长。

Tsung-Wan Mei在2011年提出了ILP方程在解决三维集成电路划分问题时两种加速方法^[5]。第一种方法iterative ILP,即反复地运行ILP,每一次运行将电路切割为面积不平均的上下部分,确定切割出的上半部分的摆放层数I。然后在剩余的电路上继续运行ILP,切割出的上半部分安排在层数I-1。不断迭代,直到整个三维集成电路的器件摆放完毕,该算法的伪代码如图2-9所示。第二个方法是提前对电路做聚类处理,将连接关系比较紧密的逻辑模块合为一体。从而减少了ILP需要处理的变量数,达到加速运算的目的。

```

ITERATIVE-ILP( $G, l$ )
1.    $m=0$ ; //#TSVs
2.   while  $l>0$ 
3.        $m=m+UNBALANCED-2LAYER-ILP(G, l);$ 
4.        $G'=G-\{v_i|x_{i,2}=1\};$ 
5.        $l=l-1;$ 

```

图2-9 iterative ILP的伪代码

H H Yeh, S H Huang, K H Li 将 ILP 方程运用到以最小化温度升高程度为优化目标的三维集成电路的划分中^[6]。因为电路层中电介质的热传导性很弱,在三维集成电路中,叠在一起的电路层会导致温度的大幅度上升。该算法使用紧凑电阻网络热模型 (compact resistive network thermal model) 来计算电路中温度上升总量,建立了以最小化温度为目标的 ILP 方程,在没有引入额外的 TSV 的情况下,相比其他算法减少了 13.6% 的温度增量。

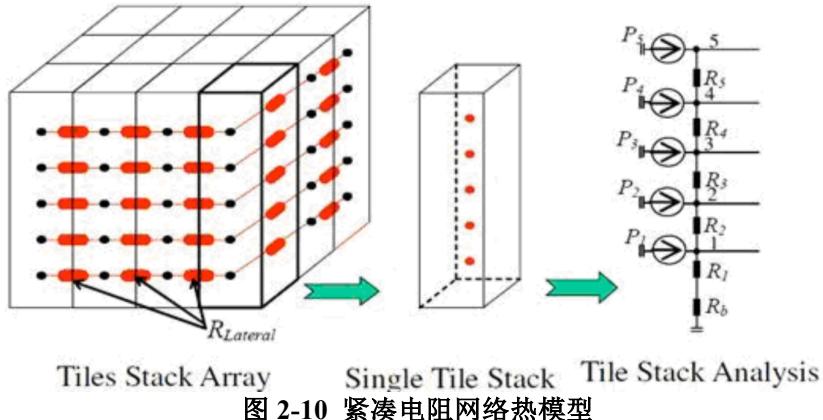


图 2-10 紧凑电阻网络热模型

W.L.Hung 等人于 2006 年第一次将模拟退火算法应用到了三维集成电路的划分中^[7]，模拟退火算法基于固体退火的原理，是一种基于概率的算法。从一个随机生成的初始解出发，根据 Metropolis 接受准则，决定是否接受新的状态，伴随着算法进程中不断递减的控制参数（模拟的是固体退火时的温度变量），最终求得全局最优解。该算法在解决三维集成电路划分问题时，利用了芯片面积和线长信息指导了电路划分优化的方向，但是耗时较长而且没有考虑 TSV 引起的额外开销。2011 年，文献[8]在考虑了 TSV 数目的情况下对模拟退火算法在三维电路划分时的应用进行了改进，并且使用简单的层面积计算代替了复杂的模块面积之和的计算，减少了算法耗时，该算法的流程如图 2-11 所示。

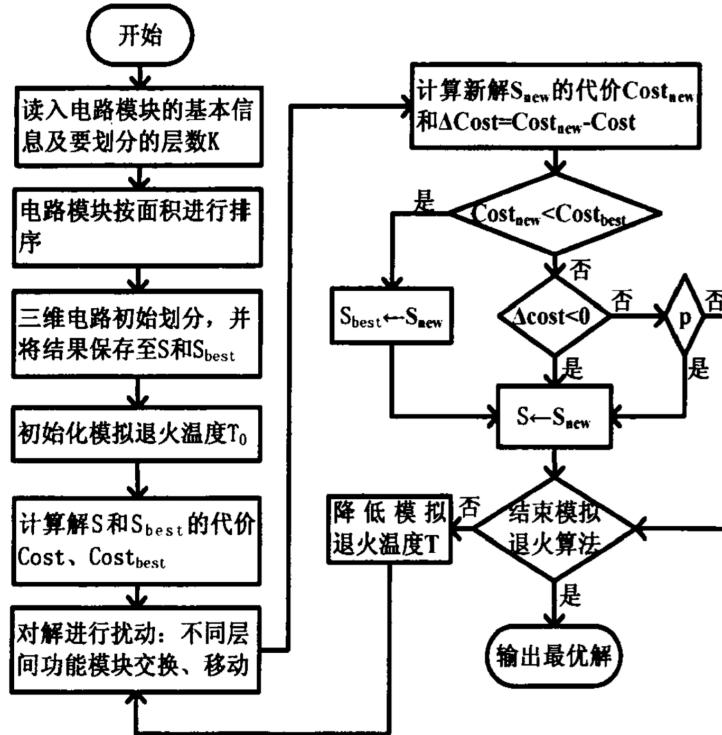


图 2-10 基于模拟退火算法的三维电路划分算法示意图

对于超大规模的三维集成电路划分问题，多级划分算法的应用十分广泛[9][10]。多级划分算法为其他三维电路划分算法如迭代改进算法、线性规划算法提供一个多级框架。以文献[9]为例，该文献便是使用了一个类 FM 算法嵌入一个多级划分算法框架。该框架主要包含以下几个步骤，如图 2-11 所示：首先，将原始展平的电路网表进行粗糙化，即将连接紧密的节点进行结群，视为一个节点。其次，在结群的维度进行初始划分，并使用类 FM 算法进行迭代优化。然后，将结群逐渐展开还原为原始的电路网表，并使用类 FM 算法将结群展开后的电路单元进行移动，在逐步求精的过程中不断优化割边的总代价。最后，所有的结群都已被展开，进行反粗糙化操作，将划分结果还原为对原始电路网表的划分。多级划分算法可



以较好地解决超大规模三维集成电路的划分问题，在并行环境下使用时，可以划分百万数量级电路单元。同时，由于每次在小规模网表上的类 FM 算法多次调用，解决了类 FM 算法在处理数量较多的节点划分问题时会在得到较高割代价或运行数次便结束的问题，进一步提升了已有划分算法的解的质量。

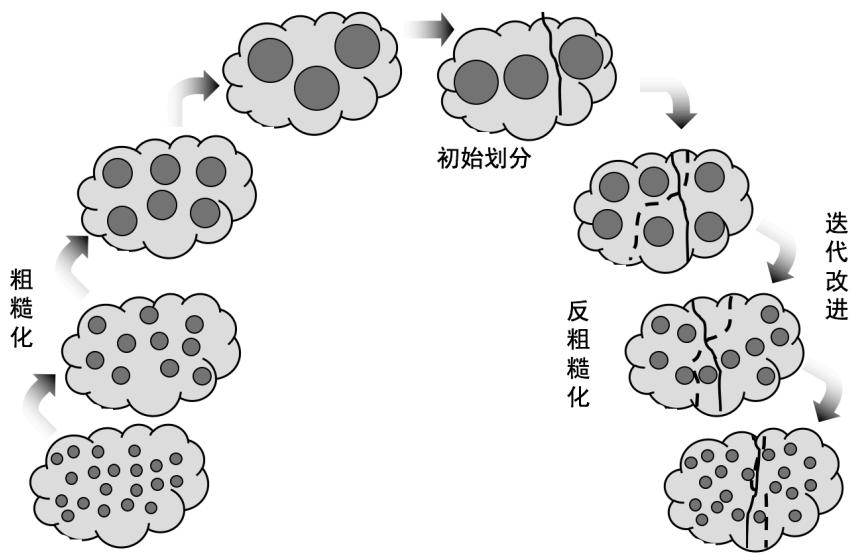


图 2-11 多级划分框架示意图

2.3 本章小结

本章首先对三维集成电路划分问题进行了描述，阐述了三维集成电路划分在三维集成电路技术中的地位，并解释了传统的二维集成电路划分与三维电路划分的不同之处。然后对国内外目前的电路划分算法进行调研，依据划分模块的策略，大致有迭代改进算法、线性规划算法、模拟退火算法和多级划分算法。在每类算法中各选择了一或两个典型算法进行了概要的描述，分析了其算法特点和发展历程。

第三章 基于 HEVC 运动估计模块的三维集成电路划分算法 设计

3.1 三维集成电路电路网表的数学模型

为了对三维集成电路进行划分，需要用适当的数学模型来描述电路网表，比较常用的有赋权图和超图数学模型。

在赋权图模型中，图中的每个顶点代表电路中的一个模块或单元，顶点的权值代表该模块或单元的面积，用以计算面积约束。图中的边代表模块之间的连线，边的权值代表这条边两端的模块在电路中实际的连线数。电路划分，即将赋权图中的顶点分为 k 个子集，同时每个子集的顶点权值之和不超过一个阈值，代表这个划分满足面积约束。对于图中的每条边，边的两端连接的顶点如若不在同一个子集中，则这条边便为割边，其割边代价为这条边的权值。划分的割边总代价便为赋权图中每个割边的权值之和。

超图 (Hypergraph) 与赋权图相比是一个更广义的图的数学模型，其顶点的定义与赋权图一致，但是超图的边，称为超边 (Hyperedge)，是一条可以连接任意多个顶点的边。在图论中，边的度是指这条边连接了多少个顶点，赋权图中的边全是度为 2 的边，而超图中的边的度全都不小于 2。

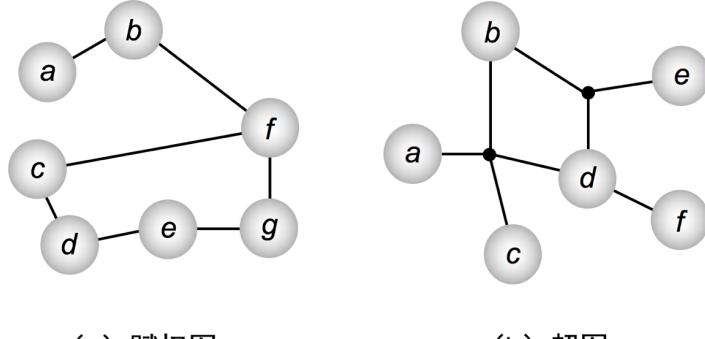


图 3-1 一个赋权图和超图示例

使用超图的数学模型可以更准确地描述电路网表，因为电路中的各个模块往往不止只有一条线连接，而每条线也往往不止连接两个模块，使用赋权图计算出的割边总代价一般会偏大，因为不同的割边可能重复计算了真实电路中只会出现一次的割线。

在本课题中，我使用超图数学模型来描述电路网表， K 路电路划分问题的数学描述如下，给定一个超图 $G = (V, E)$ ，顶点总数 $|V|=n$ ，将顶点集合 V 划分成 k 个子集，每个子集记为 $V_1, V_2, V_3, \dots, V_k$ ，要求每个顶点只能出现在一个子集中，即对于任意的 i, j 有：

$$V_i \cap V_j = \emptyset, i \neq j \quad \text{式 (3-1)}$$

对每个节点 $v \in V$ ， $\text{area}(v)$ 表示这个节点对应的电路模块的面积，对于每条边 $e \in E$ ， $w(e)$ 表示这条边对应的互连线的优先级或权重。对于任意一个子集 V_i 其内部节点的 area 函数权值之和不能超过规定的面积上下限 area_{min} 、 area_{max} ，数学公式表示为：

$$\text{area}_{min} < \sum_{v \in V_i} \text{area}(v) < \text{area}_{max} \quad \text{式 (3-2)}$$

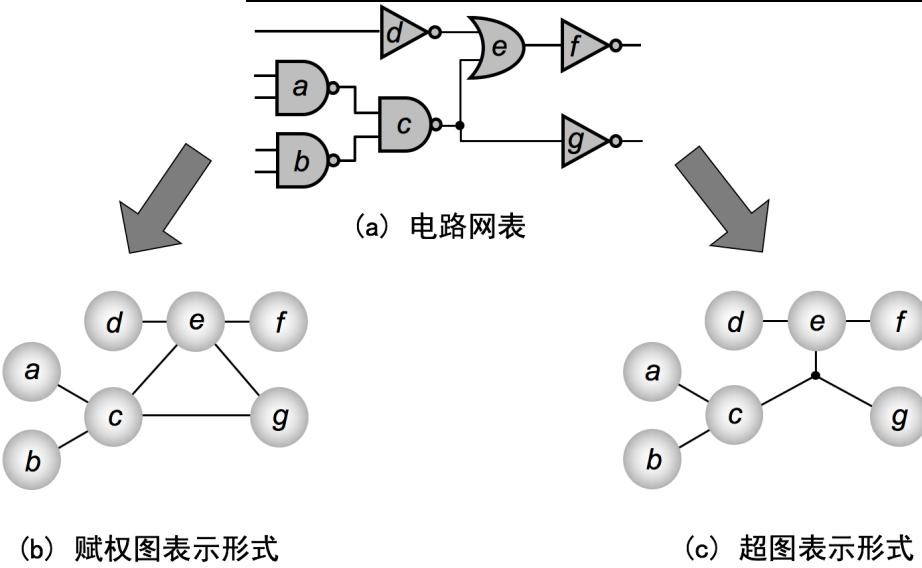


图 3-2 赋权图和超图描述电路网表时的区别

对于一条边 $e \in E$, 如果它连接的顶点 $v_1, v_2, v_3, \dots, v_k$ 不都在同一个子集中, 则称这条边 e 为一个割边, 其产生的割代价为 $w(e)$ 。如果使用 Ψ 表示割边的集合, 即割集, 则最小割代价的优化目标可以表示为:

$$\sum_{e \in E} w(e) \quad \text{式 (3-3)}$$

因此三维集成电路划分问题便转化为了在满足划分要求 (式 3-1) 和面积平衡约束 (式 3-2) 的条件下, 最小化割代价 (式 3-3) 的超图分割问题。

3.2 超图数据结构的建立

在章节 3.1 中我们介绍了超图和赋权图两种数学模型, 因为超图比赋权图更适合用来抽象实际电路的互连关系, 所以我决定使用建立超图的数据结构来描述电路网表。

在章节 3.7.1 中, 我搭建的三维集成电路划分平台会对门级网表.v 文件进行解析, 生成抽象语法树(Abstract Syntax Tree, AST)供我们建立描述超图的数据结构。为了准确地描述一个超图 $G = (V, E)$, 并且便于划分算法实现时的使用, 我定义了以下几个数据结构:

module_list: 一个由 module 名称组成的列表, 表示当前所有需要划分的 module 集合, 即超图中的顶点集合 V 。

wire_list: 一个由 wire 名称组成的列表, 表示当前电路中所有 net 的集合, 即超图中的超边集合 E 。

module_wire_dict: 一个描述 module 到 wire 映射关系的字典, 即 AST 中每个 module 后声明的 port 的集合, 对应于超图中为一个可以通过顶点索引与之相连的超边的字典。

wire_module_dict: 一个描述 module 到 wire 映射关系的字典, 是 module_wire_dict 的逆向字典, 对应于超图中为一个通过超边索引与之相连的顶点的字典。

design_list: 一个由 design 名组成的列表, 表示电路中所有 design 的集合。

module_design_dict: 一个描述 module 到 design 映射关系的字典, 因为面积信息是由 design 索引, 所以通过这个字典可以得到每个 module 的面积。

wire_weight_dict: 一个描述 wire 到该连线位宽的字典, 用以表示该连线在电路中实际的连线数。

将 AST 转化为超图数据结构的函数, 接收 AST 文件名和要解析的 top 模块的名称为参

数, 由以下几个步骤进行:

- (1) 打开 AST 文件, 找到与参数中 top 模块名称相同的 moduleDef 属性元素, 逐行解析 top 模块下的内容。
- (2) 如果是 net 属性的元素, 将其加入 wire_list, 计算出位宽 weight, 将这对 net、weight 以 key-value 的形式插入 wire_weight_dict。
- (3) 如果是 module 属性的元素, 将其加入 module_list, 读取它的 design 名称, 加入 design_list, 并将这对 module、design 插入字典 module_design_dict。
- (4) 如果是 port 属性的元素, 继续读取在同一个 module 下所有的 port 属性元素, 将其合为一个列表, 然后将这对 module、port 列表插入字典 module_wire_dict。
- (5) top 模块的内容处理完毕后, 根据 module_wire_dict 的值生成逆字典 wire_module_dict。
- (6) 将 module_wire_dict, wire_module_dict, module_list, wire_list, design_list, wire_weight_dict, module_design_dict 的值返回。

3.3 hierarchy 结构电路展开

在将实际电路进行 k 路划分时, 由于一些模块的面积较大会制约可以划分的层数 k。比如 HEVC 的运动估计模块, 其模块总面积为 $1263419\mu\text{m}^2$, 而最大的单个模块面积可达 $673293\mu\text{m}^2$, 超过了总面积的一半。此时如果直接将该电路进行二分是没有意义的, 因为在初始划分时就不可能将划分间的面积达到均衡。不过因为 HEVC 的运动估计模块具有 hierarchy 的电路结构, 可以将面积过大的模块向下展开为面积较小的模块, 从而使其也可进行三维集成电路 k 路划分。

一般的, 我们认为当电路中最大模块的面积超过了划分后每层的平均面积时, 当前层数的划分便无法进行。对于具有 hierarchy 结构的电路来说, 可以将面积过大的模块向下展开, 直到其子模块的面积均小于划分后每层的平均面积, 如此便可实现多层划分。具体实现如下: 对所有的模块的面积求和, 得到当前电路模块总面积 $\text{area}(V)$ 。对于给定的 k 路划分, 定义一个面积阈值 $\text{area}(K)$, 其中:

$$\text{area}(K) = \text{area}(V)/k \quad \text{式 (3-4)}$$

电路中有些模块如 memory 不能进行向下展开, 他们的存在也会制约划分层数 k。比如 HEVC 的运动估计模块中, inst_sr_mem_cluster 模块的面积为 $415132\mu\text{m}^2$, 将近运动估计模块总面积的三分之一, 而且是一个不能向下展开的 memory 模块。该模块导致 HEVC 的运动估计模块最多进行 $k=3$ 的三维集成电路划分。因为这种面积较大而且无法向下展开的 memory 模块的存在, 定义另一个面积阈值 $\text{area}(\text{Mem})$, 其值为这些无法展开的模块面积最大值。

如果 $\text{area}(K) < \text{area}(\text{Mem})$, 说明因为不能展开模块的限制, k 层的划分无法实现。

反之, 说明该电路可以通过进行展开来满足 k 层划分的要求。创建一个待展开模块的任务队列, 将 top 中的模块按面积大小排序后遍历, 将面积大于 $\text{area}(K)$ 的模块添加到队尾。

遍历待展开模块的任务队列, 使用 3.2 中描述的构建超图数据结构的函数, 以该模块的名称作为 top 模块的名称传入, 返回值即为将该模块展开后的数据信息。将该模块展开后的数据结构与 top 模块中的各数据结构一一合并, 遍历该模块展开后的子模块, 将面积仍然大于 $\text{area}(K)$ 的模块添加至待展开模块队列的队尾, 将该模块从待展开队列和划分模块列表中删除。

3.4 初始划分的生成

初始划分是三维集成电路划分中的重要一步，初始划分生成的各层电路面积是否均匀、割边数量的多少对后续运行的迭代改进算法影响很大，往往优秀的初始划分可以产生更好的划分结果。在针对 HEVC 的运动估计模块的三维集成电路划分算法中，我设计与实现了三种初始划分的生成方式：由用户指定初始划分、随机生成初始划分、种子增广法产生初始划分。

由用户指定初始划分：用户可以根据电路模块之间的逻辑关系、功能联系的紧密程度自己进行初始划分，再使用划分算法进行迭代改进。使用这种初始划分方式，用户需要让自定义的每层电路都满足面积平衡约束，然后将每层的 top 模块名称作为参数传入即可。电路划分评估平台会根据每层的 top 模块名称解析.v 电路网表文件，并生成每一层的模块列表（module_list）供进一步优化划分使用。在 HEVC 的运动估计模块中，一个生成两层初始划分的思路如下：将 SRAM 和数据流控制器放在一层，将由 router 单元、absolute 单元组成的 PE 阵列放在另一层，这样划分使得功能相近的模块放在一起，使得逻辑关系紧密的模块之间的互连线缩短。

随机生成初始划分：当用户没有指定初始划分时，可以使用随机算法生成初始划分。首先，通过面积平衡约束计算出每层划分的面积上下限，为每层划分生成一个空的列表。然后，从上至下选择一层划分，每次从未添加至初始划分的模块中随机抽取一个，如果当前划分的面积加上抽取的模块面积没有超过面积上限，那么将其加入该层划分。如果当前划分的面积已经超过了每层的平均面积，则该层划分生成完毕，选择下一层划分，重复随机抽取模块进行添加。最后，在耗尽了所有的模块后生成了最后一层划分，初始划分结束。随机划分速度较快，但因为没有考虑模块间的连接情况，最后生成的初始划分质量可能较差。

种子增广法：种子增广法是一种构造类算法，由一个用户指定的“种子”模块开始，围绕该模块进行广度优先遍历（BFS），不断将与之连接关系紧密的模块加入当前的划分中，直到该划分超过超过了面积平衡约束的要求。然后，为下一个划分选择一个新的“种子”模块，直至生成所有层的初始划分。种子增广法考虑到了模块间的连接关系，通过将互连关系紧密的模块放入同一个划分中，可以有效得减少划分间的割边数，经过后续优化划分算法的迭代改进后，往往能取得比随机算法更好的效果，也是我默认的初始划分的生成方式。其具体流程如下：

- (1) 选择一个节点，作为“种子”节点，记为 V_{seed} 。
- (2) 从 V_{seed} 开始，广度优先遍历超图，按照遍历顺序将与 V_{seed} 相连的节点加入队列。
- (3) 不断将队列中的节点按顺序加入 V_{seed} 所在的划分中。
- (4) 如果当前划分超过了每层面积上限，停止添加节点。
- (5) 在剩余的节点中，重新选择一个“种子”节点。
- (6) 重复 2-4 步，以新种子为起始，生成一层新的划分。
- (7) 重复以上过程，直至节点耗尽，初始划分生成完毕。

3.5 移动增益模型

在第二章中，我们介绍过经典的电路划分算法——FM 算法。然而 FM 算法在计算节点的移动增益时，只考虑了两个划分集合之间的割边情况。一般的，在扩展到 k 路划分时，需要在所有可能的划分集合对之间使用 KM 算法进行二分的迭代改进，直到整体的割边数没有改善。但是在三维集成电路的 k(k 大于 2)层划分中，这样的划分算法不再适用。因为割边反映在三维集成电路中便是 TSV，两个划分集合之间的割边可能会影响别的划分集合。

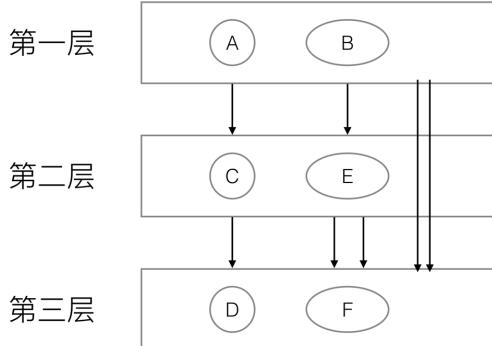


图 3-3 三维集成电路划分算法割边示例图

如图 3-3 所示,这是一个将三层电路划分的示意图,在第一层与第二层之间有两条割边,第二层与第三层之间有三条割边,第一层与第三层之间有两条割边。如果按照 FM 算法中的割边模型,此时共有 $2+3+2=7$ 条割边。然而在三维集成电路中,第一层与第三层之间的割边需要经过第二层,所以每条割边需要两个 TSV 才能实现,即实际等效有 $2+3+2*2=9$ 条割边,同时还有可能出现第一、三层的割边与第二、三层的割边重复的情况,层数越多情况越复杂。所以为了更准确地描述三维集成电路中 TSV 与割边的抽象关系,需要重新建立三维电路划分的移动增益模型。

为了描述上文中提到的位于不同层之间的割边对应的 TSV 数量,我们从上而下为每层划分编号为 1、2、3、...、 k ,即每个划分层用 L_1 、 L_2 、...、 L_k 表示。对每条边 E_i 定义一个代价 $Cost(E_i)$,其值为 E_i 出现的划分层中最大一层的编号减去最小一层的编号,反映在实际电路中,即为了实现割边 E_i 需要的 TSV 的个数。对于一个节点/模块 V_i ,与之相连的边的集合记为 $Adj(V_i)$,当其放在划分层 L_j 时,定义其代价为与之相连的边的代价之和。公式表示为:

$$Cost(V_i, L_j) = \sum_{k=1}^{num(Adj(V_i))} Cost(E_k) \quad \text{式 (3-5)}$$

式 3-5 中, $num(Adj(V_i))$ 代表 V_i 相连的边的数量。

因为 V_i 置于不同的划分层时代价不同,所以 V_i 的移动增益定义为其在初始划分层时的代价减去其在代价最小的划分层时的代价,公式表示如下:

$$Gain(V_i) = Initial\ cost(V_i) - min_{1 \leq j \leq K} Cost(V_i, L_j) \quad \text{式 (3-6)}$$

3.6 算法流程

对于一个由超图模型 $G = (V, E)$ 描述的电路, V 是节点/模块的集合, E 是超边/连线的集合,每条超边连接着两个或两个以上的节点。我们要将超图 G 划分为 K 层,自上而下每层分别用 L_1 、 L_2 、...、 L_k 表示,划分结果需要满足式 3-7,算法运行流程如下:

$$\begin{aligned} \bigcup_{i=1}^K L_i &= V \\ L_i \cap L_j &= \emptyset, \forall i \neq j \end{aligned} \quad \text{式 (3-7)}$$

- (1) 计算 K 路划分的面积平衡准则,因为划分层数的增多,经典 FM 算法中使用最大模块的面积作为面积平衡的浮动约束会使面积上下限相差过大,导致最后划分结果的各层面积相差很大。在 K 路划分中,引入一个面积平衡容忍因子 α ,代表着



每层与平均面积 $area(L)_{avg}$ 偏离比例的容忍范围。所以面积平衡约束的上下限 $area(L)_{max}, area(L)_{min}$ 为：

$$area(L)_{max} = area(L)_{avg} * \left(1 + \frac{\alpha}{100}\right) \quad \text{式 (3-8)}$$

$$area(L)_{min} = area(L)_{avg} * \left(1 - \frac{\alpha}{100}\right)$$

每个划分层应当满足的面积平衡约束条件为：

$$area(L)_{min} \leq area(L_i) \leq area(L)_{max}, 1 \leq i \leq K \quad \text{式 (3-9)}$$

- (2) 使用 3.5 描述的节点移动增益模型，计算所有节点的初始移动增益。如果一个模块的移动增益为 0，说明该模块当前所处的划分层已经是它最好的位置了，我们在这个轮次中将其锁定。
- (3) 将自由节点按照移动增益的大小进行排序，找到不违反面积平衡约束的增益最大的节点，将其移动到它最好的划分层，将其在这个轮次中锁定，并重新计算所有与它相连的自由节点的移动增益。如果有多个节点都满足面积约束条件而且增益相同，那么连接更多的超边的节点具有更高的优先级。原因是当该节点移动后，与它连接的节点数较多，即需要更新移动增益的节点较多，有可能会产生更多的正增益的节点，使得整个划分的质量更好。
- (4) 不断地重复上述步骤，直到所有的节点都被锁定，求出最大正增益移动序列，如果在本轮次中拥有正的总增益，那么将划分结果作为初始划分开始下一轮迭代。否则，结束算法。

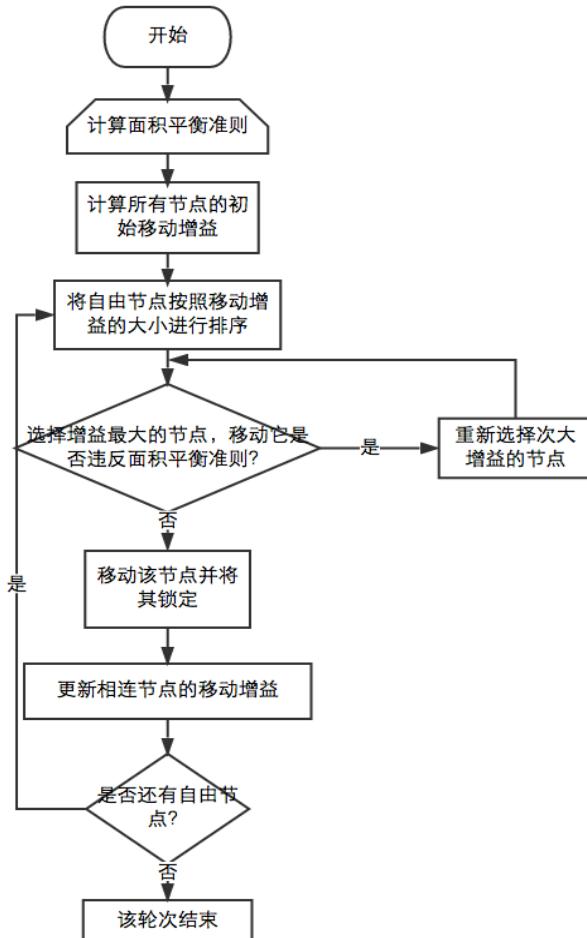


图 3-4 三维集成电路划分算法流程图

下面对该算法在一个三维集成电路的三层划分问题示例进行说明：

如图 3-4 (a) 所示，该三层三维集成电路被抽象为一个由节点 1、2、3、4、5、6 和超边 a、b、c、d、e 组成的超图模型。使用式 4-2 计算每个节点的割边代价：对于节点 1，它只与超边 a 相连，所以节点 1 的割边代价即为超边 a 的代价。当节点 1 分别处于第一、二、三层时，其割边代价计算如下：

$$\begin{aligned} Cost(V_1, L_3) &= 3 - 2 = 1 \\ Cost(V_1, L_2) &= 3 - 2 = 1 \\ Cost(V_1, L_1) &= 3 - 1 = 2 \end{aligned} \quad \text{式 (3-10)}$$

所以当节点 1 位于第二层或者第三层时，割边代价最小为 1。节点 1 的移动增益为：

$$Gain(V_1) = 1 - \min(1, 1, 2) = 0 \quad \text{式 (3-11)}$$

对于节点 2，与之相连的超边有 a、b、c、e，所以其割边代价为这四条超边的割边代价之和，计算如下：

$$\begin{aligned} Cost(V_2, L_3) &= (3 - 2) + (3 - 1) + (3 - 2) + \\ &\quad (3 - 1) = 6 \\ Cost(V_2, L_2) &= (3 - 2) + (2 - 1) + (2 - 2) + \\ &\quad (2 - 1) = 3 \\ Cost(V_2, L_1) &= (3 - 2) + (2 - 1) + (2 - 1) + \\ &\quad (1 - 1) = 3 \end{aligned} \quad \text{式 (3-12)}$$

所以当节点 2 处于第一、二层时有最小割代价 3，节点 2 的移动增益为：

$$Gain(V_2) = 6 - \min(3, 3, 6) = 3 \quad \text{式 (3-13)}$$

同理，对于节点 3、4、5、6，计算出他们的移动增益和拥有最小割代价的层数 OP_{layer} 如下：

$$\begin{aligned} Gain(V_3) &= 2 - \min(1, 2, 4) = 1 \\ OP_{layer}(V_3) &= 3 \\ Gain(V_4) &= 0 \\ OP_{layer}(V_4) &= 2, 1 \\ Gain(V_5) &= 2 \\ OP_{layer}(V_5) &= 2, 3 \\ Gain(V_6) &= 2 \\ OP_{layer}(V_6) &= 3 \end{aligned} \quad \text{式 (3-14)}$$

计算完移动增益后我们选择拥有最大增益 3 的节点 2，检查它是否满足面积平衡约束。我们假设节点的面积依次为 3、2、4、1、3、5，所以每层的面积依次为 8、5、5。取面积平衡因子 α 为 0.5，所以最大允许的层面积为 9，最小的划分层面积为 3。对于节点 2，移动至第二层时，第三层面积变成 3，第二层面积变成 7，满足面积平衡约束。假设将节点 2 移动至第一层，第一层面积变成 10，超过了层面积的上限，所以节点 2 只能移动至第二层。移动后的划分情况如图 3-4 (b) 所示，电路的总割边数由 7 减少至 4。将节点 2 锁定，更新与节点 2 相连的节点的增益，然后再找出移动增益最大的节点进行移动。不断重复上述过程，直到划分结果没有改善，算法结束。

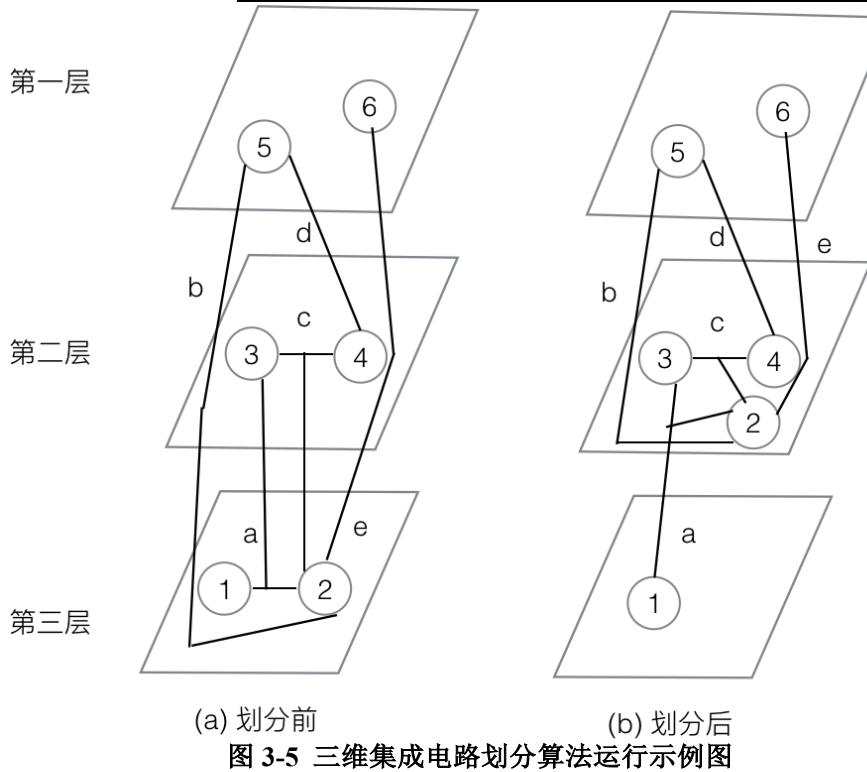


图 3-5 三维集成电路划分算法运行示例图

3.7 三维集成电路划分评估平台

为了针对 HEVC 运动估计模块的硬件实现，评估三维集成电路划分算法的划分质量与效果，我搭建了一个三维集成电路划分评估平台。主要包括以下几个步骤：

- (1) 解析硬件实现的电路网表文件
 - (2) 解析 DC 综合产生的模块面积报告
 - (3) 建立超图数据结构描述电路单元互连关系
 - (4) 对 hierarchy 结构的电路按需进行模块展开
 - (5) 生成初始划分, 供划分优化算法使用
 - (6) 施加划分优化算法, 评估划分结果

其中，步骤 3、4、5 已分别在章节 3.2、章节 3.3、章节 3.4 进行了详细描述，这里便不再重复。本节中将主要介绍步骤 1、2、6 的功能和实现。

3.7.1 电路网表文件的解析

该平台接收一个由 DC 综合产生的.v 文件，将其转化为抽象语法树(Abstract Syntax Tree, AST)供后续处理。DC 生成的.v 文件为门级网表文件，与 RTL 级的.v 文件相比代码没有复杂的逻辑关系。每个设计(Design)中主要包括 design、net、port、cell 等元素，针对这些元素，在解析过程中构造了 moduleDef、net、width、module、port 等类分别进行处理，然后生成 AST 文件输出。下面以一段删减过的 idct 设计文件来对 AST 的生成进行详细说明。

```
module idct ( clk, rstn, mode, start, din, dout, dout_mode );
    input [15:0] din;
    output [15:0] dout;
    wire [255:0] idet in col;
```

```
s2p_1 inst_s2p1 (.clk(clk), .rstn(rstn), .start(start), .mode(mode), .din(
    din), .dout(idct_in_col), .s2p_ready(s2p_ready_col), .mode_out(
    s2p1_mode));

endmodule
```

首先，我们读到了 idct design 的声明，将其作为 moduleDef 输出，对于 design 声明里描述的接口我们忽略，因为只有 design 在作为 cell 调用时才有 net 的互连关系的说明。

然后，是三个 net 的声明：din、dout、idct_in_col，对于 input、output、wire 的连线声明，我都将其归入 net 类中，对于描述线位宽的[15:0]等数字区间，使用 width 类对区间的上下限分别进行输出。

最后，是一个 design 为 s2p_1 的 cell 调用，我使用 module 类对其进行处理，按照 cell 名、design 名的顺序进行输出，对于调用的接口的声明，使用 port 类对括号内的 net 名进行输出，即真实的连线名而不是 design 声明时的端口名称。

对于该示例设计文件，输出的 AST 文件代码如下，供以后建立超图数据结构时使用。

```
moduleDef idct
net din
width 15
width 0
net dout
width 15
width 0
net idct_in_col
width 255
width 0
module inst_s2p1 s2p_1
port clk
port rstn
port start
port mode
port din
port idct_in_col
port s2p_ready_col
port s2p1_mode
```

3.7.2 面积报告的解析

为了建立面积平衡约束，不仅需要得到电路单元间的互连关系，还需要知道每个模块的面积，所以需要从 DC 综合生成的面积报告中得到每个 design 的面积，一个面积报告的示例如下。

I_tier1	590119.8538	46.7	1761.1200	1044.0000	49620.5664	tier1
I_tier1/inst_cb_block_ctrl	370.8000	0.0	86.0400	284.760	0.0000	cb_block_ctrl

I_tier1/inst_cb_ext_ctrl 365.7600 0.0 77.4000 288.3600 0.0000 cb_ext_ctrl

在面积报告中，每一行都是一个 cell 的面积信息，其中第一个元素是指 hierarchy 结构下的 cell 名称，第二个元素是指该 cell 的绝对面积，最后一个元素是该 cell 的 design 名称。因为同一个 design 会由不同的 cell 调用，所以我建立了一个 design 与面积的字典，这样便可以索引每一个 module 的面积。

在后续的工作中，我发现有些基本单元如 NOR、AND 门的面积并没有在 DC 的面积报告中展示，对于这些基本单元，我通过在库的.lef 文件中找到它们长和宽的尺寸信息，然后计算出相应的面积将其加入 design 与面积的字典中。

3.7.3 施加划分优化算法，评估划分结果

初始划分生成完毕后，统计每层的模块数量、面积和划分的割边总代价。通过模块面积的分布情况，计算出可以划分的层数 k，施加划分优化算法，统计算法结束后的每层模块数量、面积和割边总代价。划分结果主要从以下几个方面进行评估：

- (1) 割边总数。割边数是最重要的评估指标，因为在三维集成电路的划分中，层与层之间的互连线使用 TSV 实现。每个 TSV，加上垫脚和隔离区，占据了相当可观的电路面积。较多的 TSV 数量，不仅会大大增加每层电路的面积，还给后续的布局布线工作带来了困难。
- (2) 划分后的最大层面积。划分后的最大层面积，不仅反映了划分的各层之间面积均匀程度，还决定了整个电路使用 3D IC 技术实现后的总面积。因为三维集成电路的总面积为最大层面积与层数的乘积，所以最大层面积越小，划分各层之间的面积越均匀，总面积越小，面积利用率越高。面积利用率定义为所有模块总面积与三维集成电路实现时的总面积之比，用 max_layer_area 表示最大层面积、chip_area 表示所有模块的总面积、K 代表划分的层数，则面积利用率 radio 公式如下：

$$radio = \frac{chip_area}{K * max_layer_area} \quad \text{式 (3-11)}$$

- (3) 划分后的各层面积分布情况。因为在划分时统计的只是模块面积之和，没有考虑到 TSV 面积的影响。在三维集成电路的实现中，每条割边会在其连接的两层之中的上面那层引入 TSV 面积。所以，划分各层之间的面积最好按照从上到下依次递增，这样可以修正一部分 TSV 面积对最大层面积的影响，在三维集成电路的物理实现时会得到更好的面积利用率。

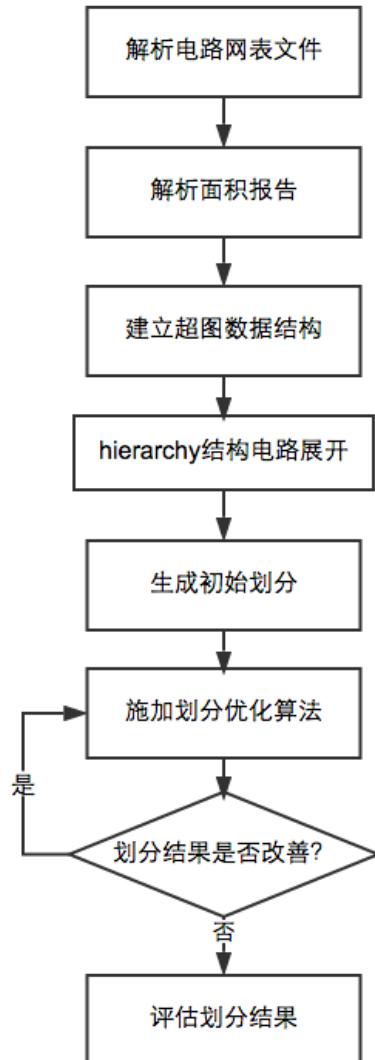


图 3-6 三维集成电路划分平台流程图

3.8 实验结果与分析

我使用 Python 语言完成了三维集成电路划分算法评估平台的搭建和面向三维集成电路 K 路划分的 FM 算法的实现。系统环境为 OS X El Capitan, 版本 10.11.5, Intel Core I7 2.2GHz CPU, 16GB RAM。测试电路为 HEVC 编码器的 IDCT 和 ME 模块，两个模块的电路规模如表 4-1 所示：

表 3-1 IDCT、ME 模块的电路规模

	TOP 中的模块数	展开一层的模块数
IDCT	75	755
ME	1141	18847

对于 IDCT 电路，其最大不可展开的模块面积大约为总面积的 1/6，所以我对 IDCT 电路在面积平衡容忍因子 α 分别为 0.3, 0.4, 0.5 的条件下对其进行 K=2、3、4、5、6 层划分。其中 K=2、3、4 时，不需要向下展开，IDCT 的总模块数为 75；划分为 5、6 层时，需要对面积过大的模块展开一层，此时 IDCT 的总模块数为 755。划分结果如图 3-5、图 3-6 和表 3-1 至表 3-5 所示。

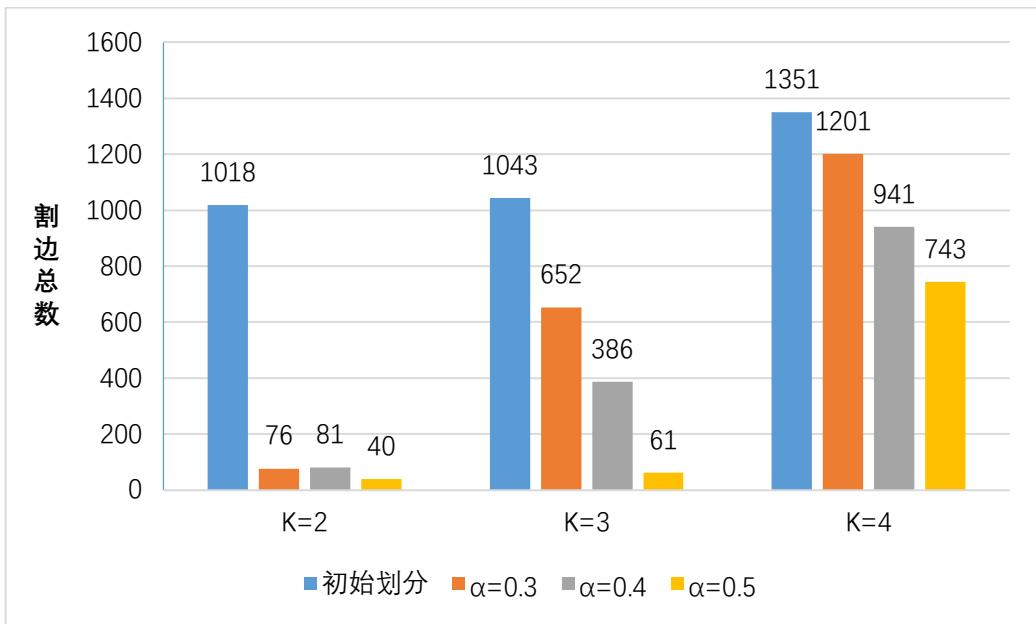


图 3-7 IDCT 模块划分结果示意图 1

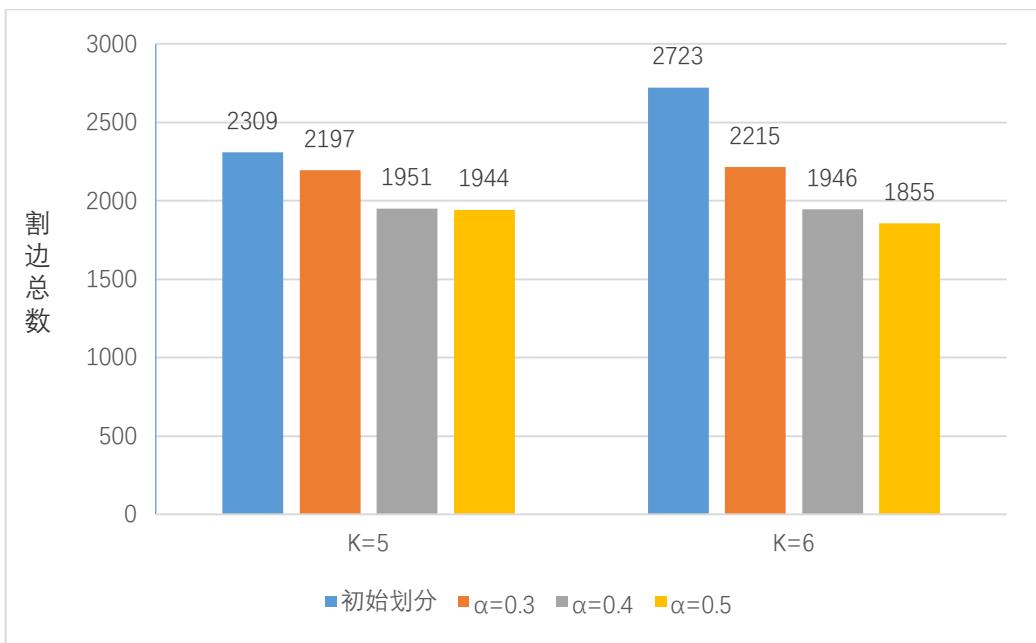


图 3-8 IDCT 模块划分结果示意图 2

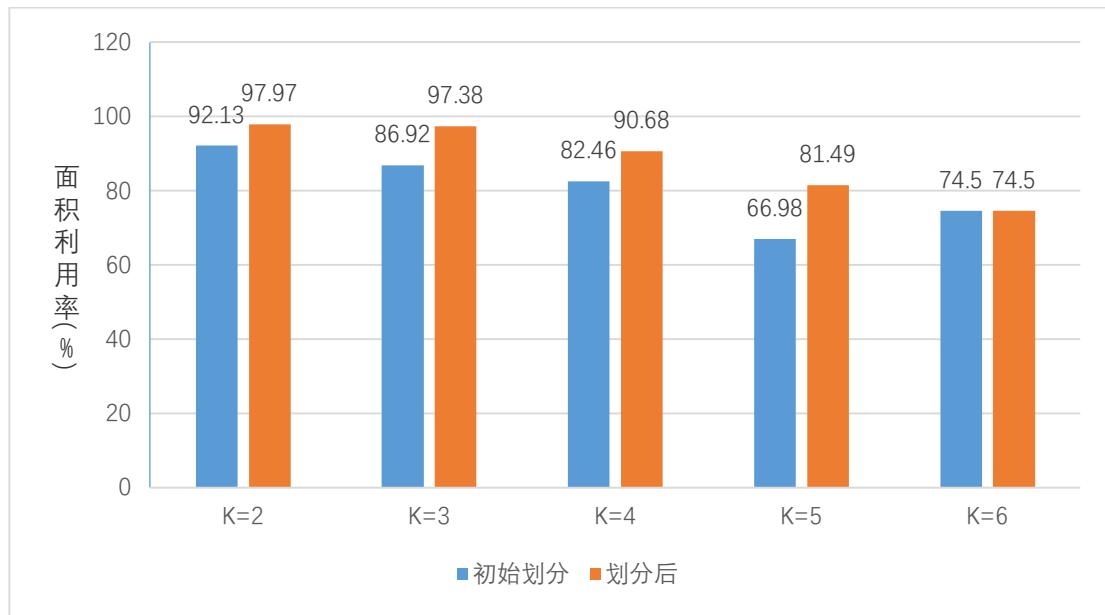


图 3-9 IDCT 模块划分结果示意图 3

对于 IDCT 电路，在划分成两层时有最少的割边数 40，但是两划分层之间的面积相差较大。在划分成三层时，有较少的割边总数 61，三层之间的面积比较均匀。在划分为四层以上时，由于划分层数增多，电路展开后模块数量增加了十倍，此时的割边总数大大增加。所以，综合看来，IDCT 最适宜的划分层数为三层，此时的总割边数为 61，每层的面积分别为 385385、368598、371881，单位为 μm^2 ，面积利用率为 97.38%。与初始划分相比，割边数减少了 92.55%，面积利用率达到了 97.38%，提高了 10.46%。

对于 HEVC 的运动估计模块，在进行 hierarchy 结构展开后，最大模块的面积为 $415132\mu\text{m}^2$ ，将近运动估计模块总面积的三分之一，而且是一个不能向下展开的 memory 模块。该模块导致 HEVC 的运动估计模块最多进行 $k=3$ 的三维集成电路划分。所以我对 ME 电路在面积平衡容忍因子 α 分别为 0.3, 0.4, 0.5 的条件下对其进行 $K=2, 3$ 层划分。划分结果如图 3-8、图 3-9 和表 3-2、表 3-3 所示。

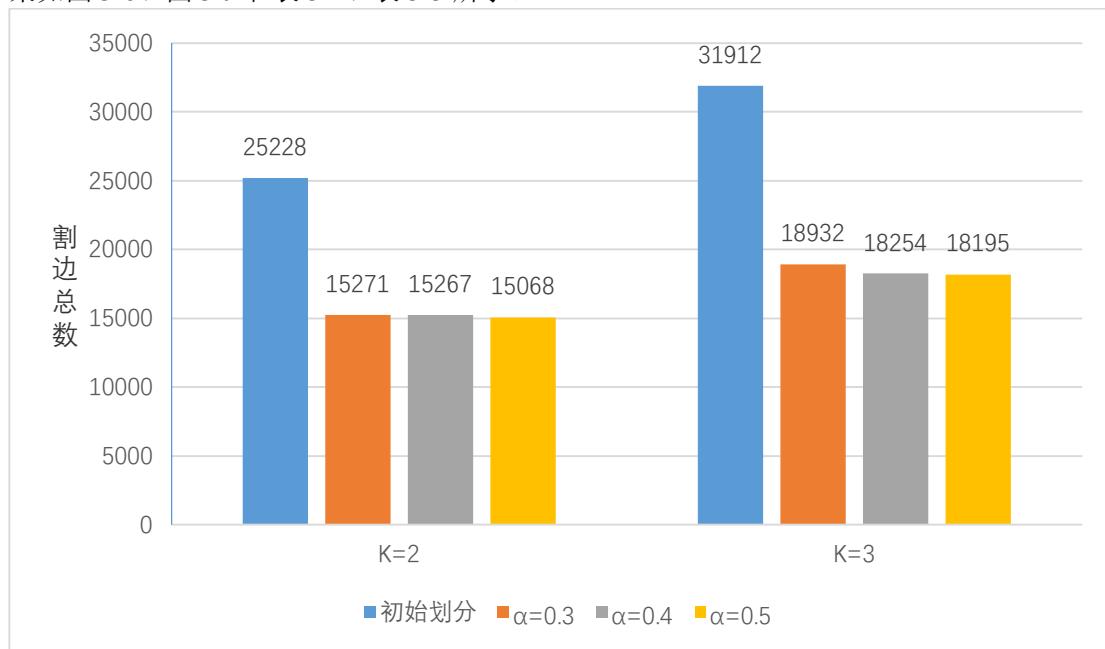


图 3-10 ME 模块划分结果示意图 1

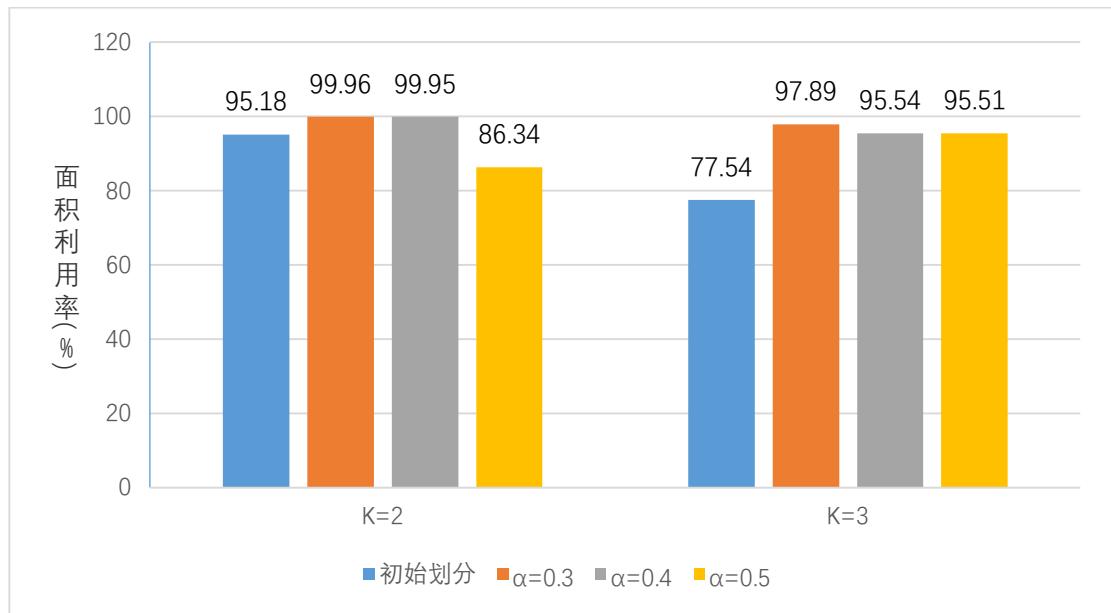


图 3-11 ME 模块划分结果示意图 2

表 3-2 ME 模块两层划分结果统计表

	割边总数	第一层面积 (μm^2)	第二层面积 (μm^2)	面积利用率
初始划分	25228	673100	608171	95.18%
$\alpha=0.3$	15271	640890	640381	99.96%
$\alpha=0.4$	15267	640940	640331	99.95%
$\alpha=0.5$	15068	742002	539269	86.34%

表 3-3 ME 模块三层划分结果统计表

	割边总数	第一层面积 (μm^2)	第二层面积 (μm^2)	第三层面积 (μm^2)	面积利用率
初始划分	31912	427166	303291	550814	77.54%
$\alpha=0.3$	18932	427495	417481	436295	97.89%
$\alpha=0.4$	18254	422685	411554	447032	95.54%
$\alpha=0.5$	18195	422204	411915	447152	95.51%

对于 ME 模块，进行两层划分时， $\alpha=0.3$ 、 0.4 时都有比较好的划分效果和面积利用率。进行三层划分时， $\alpha=0.5$ 时划分效果最好。考虑到 ME 模块的电路单元比较多，在进行布局布线后，三层划分的面积会更有优势。所以选择 $K=3$ ， $\alpha=0.5$ 作为 ME 模块的最优划分，此时的割边数为 18195，三个划分层的面积为 422204 、 411915 、 447152 ，单位为 μm^2 。与初始划分相比，此时割边数减少了 42.98%，面积利用率达到 95.51%，提高了 17.97%。

3.9 本章小结

本章首先描述了三维集成电路的两种数学模型赋权图和超图，介绍了选用超图抽象电路网表的原因，并建立超图的数据结构描述电路网表。其次，针对 HEVC ME 的部分模块面积过大问题，提出了 hierarchy 结构电路的展开方案，并介绍了三种生成 ME 电路的初始划分的方法。再其次，建立了划分算法的移动增益模型，并详细描述了整个算法的具体流程，包括计算面积平衡准则、计算初始移动增益、移动节点进行迭代改进等步骤。然后，描述了三维集成电路划分评估平台的搭建过程，具体功能包括解析电路网表文件、解析 DC 综合面积报告、建立超图数据结构描述电路单元互连关系、对 hierarchy 结构的运动估计模块电路按照模块大小分布进行向下展开、多种方式生成初始划分、施加划分优化算法与划分结果评估。

最后，对 HEVC 中 IDCT 模块和 ME 模块使用改进后的算法进行了划分，对实验结果进行了展示和分析。IDCT 最适宜的划分层数为三层，此时的总割边数为 61，每层的面积分别为 385385、368598、371881，单位为 μm^2 ，面积利用率为 97.38%。与初始划分相比，割边数减少了 92.55%，面积利用率达到了 97.38%，提高了 10.46%。对于 ME 模块，选择 $K=3$, $\alpha=0.5$ 作为 ME 模块的最优划分，此时的割边数为 18195，三个划分层的面积为 422204、411915、447152，单位为 μm^2 。与初始划分相比，此时割边数减少了 42.98%，面积利用率达到了 95.51%，提高了 17.97%。

第四章 结论

本文针对 HEVC 编码器运动估计模块的硬件实现，就三维集成电路的划分问题进行了研究。在满足面积平衡的约束条件下，以最小化割边数、提高三维集成电路的面积利用率为优化目标，进行面向三维集成的电路划分算法的设计与实现。

HEVC 是最新一代的视频编码标准，与前代 H.264/AVC 编码标准相比，在运动估计算法方面对编码单元、预测单元进行了改进，并增加了 merge 技术和高级运动向量预测技术。运动估计算法在整个 HEVC 编码过程中占据了百分之 60 以上的运行时间，运动估计模块在 HEVC 编码器的硬件实现中是最重要、最复杂的模块之一，为了提高 HEVC 编码器硬件实现的性能、减少时延、降低功耗、缩小面积，考虑使用三维集成电路技术对 HEVC 编码器的运动估计模块进行硬件实现。

三维集成电路技术是一种电路设计的新思路，结合三维空间的概念，将传统的二维集成电路 die 在垂直方向上堆叠起来。使用这种方式，可以有效地减少电路的投影面积，芯片的集成度将成倍提高。而且因为每一层电路面积相比二维集成电路的面积大大缩小，芯片上的互连线长度也有所缩短，从而减少了互连线的延迟，降低了功耗。

本课题针对 HEVC 的运动估计模块，设计了面向三维集成的电路划分算法。包括基于 BFS 的初始划分算法和迭代改进的优化划分算法。并针对 HEVC 的运动估计模块，设计了三维集成电路划分算法评估平台，主要功能包括解析电路网表文件、解析 DC 综合面积报告、建立超图数据结构描述电路单元互连关系、对 hierarchy 结构的运动估计模块电路按照模块大小分布进行向下展开、多种方式生成初始划分、施加划分优化算法与划分结果评估。对 HEVC 中 IDCT 模块和 ME 模块使用设计的三维集成电路划分算法进行了划分，从割边数、面积利用率等方面评估了划分结果。IDCT 最适宜的划分层数为三层，此时的总割边数为 61，每层的面积分别为 385385、368598、371881，单位为 μm^2 ，面积利用率为 97.38%。与初始划分相比，割边数减少了 92.55%，面积利用率达到 97.38%，提高了 10.46%。对于 ME 模块，选择 $K=3$, $\alpha=0.5$ 作为 ME 模块的最优划分，此时的割边数为 18195，三个划分层的面积为 422204、411915、447152，单位为 μm^2 。与初始划分相比，此时割边数减少了 42.98%，面积利用率达到 95.51%，提高了 17.97%。

参考文献

- [1] Ye H S, Chi M C, Huang S H. A design partitioning algorithm for 3D ICs[C]. International Symposium on Computer Communication Control and Automation. 2010:229-232.
- [2] Fiduccia C M and Mattheyses R M. A Linear-Time Heuristic for Improving Network Partitions. 19th ACM/IEEE Design Automation Conference. 1982. 175-181.
- [3] Chang H L, Lai H C, Hsueh T Y, et al. A 3D IC Designs Partitioning Algorithm With Power Consideration[C]. Quality Electronic Design (ISQED), 2012 13th International Symposium on. 2012:137 - 142.
- [4] Jiang, I.H.-R. Generic integer linear programming formulation for 3D IC partitioning[C]. SOC Conference, 2009. SOCC 2009. IEEE International. 2009:321-324.
- [5] 梅宗菀. 適用於三維積體電路之線性規劃[J]. 2010:1-33.
- [6] Yeh H H, Huang S H, Li K H. 3D IC design partitioning for temperature rise minimization[C]. Microsystems, Packaging, Assembly and Circuits Technology Conference (IMPACT), 2011 6th International. IEEE, 2011:447-450.
- [7] Hung W L, Link G M, Xie Y, et al. Interconnect and thermal-aware floor planning for 3D microprocessors[C]. International Symposium on Quality Electronic Design. IEEE Computer Society, 2006:6 pp. - 104.
- [8] 邹毅文. 三维芯片 TSV 敏感的电路划分和温度敏感的布图规划研究[D]. 合肥工业大学, 2012.
- [9] Yu C H, Yin L C, Chi M C. A multilevel multilayer partitioning algorithm for 3D ICs.[J]. Quality Electronic Design International Symposium on, 2010:483-487.
- [10] Hsueh T Y, Yang H H, Wu W C, et al. A layer prediction method for minimum cost 3D ICs.[J]. 2011:359-363.
- [11] 蔡成能. HEVC 框架下基于 3D--SSIM 的主观质量优化编码研究[D]. 西南交通大学, 2015.
- [12] 董朵, 端木春江. 基于 Hough 变换的高效视频编码标准帧内预测模式选择快速算法[J]. 计算机应用, 2014(12):3560-3564.
- [13] 安然, 王浩全, 张秀林,等. 下一代视频编码标准 H.265 的核心技术研究[J]. 计算机技术与发展, 2014(4):210-213.

谢辞

在上海交通大学的四年本科生时光是我一生中永远难忘的一段经历，母校为我们提供了丰厚的教师资源、完善的实验设施、丰富的学术活动。在老师和同学们的帮助下，四年里我的知识和能力得到了长足的进步。

感谢我的毕业设计论文的指导老师何卫锋老师。何老师从开题撰写任务书、中期组织检查答辩、研究指导实验方向，为本论文的完成付出了巨大的努力。何老师对学生的亲切关怀、言传身教，不仅带给了我智慧的启蒙，还教会了我很多做人做事的道理。在此，谨向敬爱的何老师表示衷心的感谢。

感谢实验室的胡沥学长，胡沥作为一名研二的学长，在本课题中为我提供了多种实验测试样例，与学长的交流中我学到了很多的专业知识。感谢同组的顾家威同学，每次与他讨论问题总有新的收获。

另外特别感谢我的家人，正是他们无私的爱与关怀让我可以在学校里安心完成学业。

3D IC PARTITIONING TECHNIQUE BASED ON HEVC MOTION ESTIMATION MODULE

In this paper, the hardware implementation of motion estimation module of HEVC encoder is studied. In order to minimize the number of edges and improve the area utilization ratio of the three-dimensional integrated circuits, the design and implementation of a 3D IC partitioning algorithm is realized.

Since twentieth Century, human society has entered the information age, the integrated circuit is the cornerstone of the social development of the information age, in our daily life everywhere. 40 years ago, Intel co-founder Gordon Moore proposed Moore's Law, which are well known in the field of integrated circuit: per unit area of the integrated circuit containing the number of components, once a year half double. With the development and progress of the design method and manufacturing process of the integrated circuit, the integration of the chip is increasing and the number of the devices is increasing dramatically. Under the trend of smaller and smaller chip size, the problems caused by various parasitic effects become more and more serious. In the traditional two-dimensional chips, due to the distance between modules and module interconnect long lead to occupy a lot of area and the interconnect delay has become the main bottleneck to higher chip speed, a serious impediment to the integrated circuit further enhance the degree of integration and performance. Therefore, the three-dimensional integrated circuit technology has become a continuation of Moore's law of new ideas, and has been widely concerned about the industry.

3D IC is to have several layers of device structure of integrated circuits, multilayer chip in vertical direction are stacked, to multiply improve the unit area of the number of devices, to achieve the goal of improving the degree of integration. At the same time, in the three-dimensional integrated circuit, layer and layer between using TSV (through silicon via) interconnect, making originally in the 2D chip. The long distance between the module interconnects changed into vertical interconnects, greatly shorten the interconnect length, thus solving the interconnect delay problems, and it is possible to realize the parallel signal processing, is helpful to further improve the performance of the integrated circuit. Compared with the original 2D planar space layout, three-dimensional integrated circuits increased a vertical dimension, resulting in the layout can not only control the placement of each module, you can also select different placing layers. So, compared to the 3D IC with traditional 2D IC, the utility model has the advantages of reflected in the smaller area, higher integration degree, shorter wire length, higher frequency and lower power consumption.

With the rise of the Internet industry and the development of mobile Internet, the increasing demand for online video, digital video communication, entertainment, military reconnaissance, disaster relief and other aspects in the application plays a key role. However, in real life, we have access to the video, in fact, are compressed. Because the original video file without compression processing, the amount of data is very alarming, unable to meet the needs of the actual transmission and storage. Therefore, video coding, also known as video compression, is the key technology of digital video to be widely used, the purpose is to remove the redundant data in the original video

file, reduce the amount of data to represent the video. In recent years, with the popularity of high resolution display and video equipment, ultra high definition video (2K * 4K, 4K * 8K) has gradually come into people's vision, video coding technology has been a huge challenge. In addition, a wide range of mobile video applications, video broadcasting, remote monitoring, medical imaging, portable photography and so on have been into people's lives. The diversification of video applications and the trend of HD video compression put forward higher requirements for video compression.

To this end, in April 2010, the International Telecommunication Union Telecommunication Standardization Sector (International Telecommunication Union-Telecommunication Standardization Sector, ITU-T) video encoding expert group (Video Coding Experts Group, VCEG) and the international organization for Standardization (International Organization for Standardization ISO, the International Electrotechnical Commission (International) / Electrotechnical Commission, IEC) of the MPEG (Moving Picture Experts Group, MPEG) set up a video encoding combined group (Joint Collaborative Team on Video Coding, JCT-VC), jointly developed a new generation of video encoding standard HEVC (High Efficiency Video Coding). HEVC as the latest generation of video coding standard, which is designed to replace the H.264/AVC coding standard, therefore, also known as the H.265 coding standard. Compared with the H.264 coding standard, the video quality of the video can be improved by more than one times by using the HEVC coding standard. This means that users can enjoy more high-quality video content without increasing traffic consumption, to video website operators and bandwidth costs are the largest component of operating costs, the HEVC coding standard can save more than half of the bandwidth consumption can bring better quality and better user experience to consumers.

Because motion estimation algorithm in the whole process of HEVC coding occupy percent more than 60 percent of the running time, motion estimation module in HEVC encoder hardware implementation is the most important and one of the most complex modules, in order to improve HEVC encoder hardware performance, reduce the delay, reduce power consumption, reducing the area, consider using three-dimensional integrated circuit technology of HEVC encoder motion estimation module for hardware implementation. Due to the three-dimensional integrated circuit and the circuit of the traditional two-dimensional compared to an increase of a vertical dimension, before the circuit layout wiring and need to determine each device placed layer. Therefore, it is necessary to divide the original 2D circuit, according to the classification results obtained by each layer of the device information, in order to follow the layout of the wiring work. Because the classification results of the whole circuit implementation has great influence, is divided between the number of connections, namely edge cut the number determines the number of TSV between layer and layer, more than the number of cut edges in the TSV occupied area is too large, thus affecting the overall performance of the circuit. And the total area of the three-dimensional integrated circuit physical implementation should be calculated according to the area of the largest layer, so the results of the division is the best to make the area of each layer of devices to reduce the total area of the circuit.

For given HEVC motion estimation circuit structure as the research target, developed for the three-dimensional integrated circuit partitioning algorithm, divided between the interconnect minimum as the main target of optimization and the maximum floor area is small, each area is relatively uniform, improve the three-dimensional integrated circuit implementation area utilization.

In this paper, a three-dimensional integrated circuit partitioning algorithm is designed for the

motion estimation module of HEVC. Including the initial partition algorithm based on BFS and the iterative optimization algorithm. And for HEVC motion estimation module, design the three-dimensional integrated circuit partitioning algorithm evaluation platform, the main functions include analytic network table file, comprehensive analysis of DC area report, the establishment of hypergraph data structure to describe unit circuit interconnection relations, the hierarchy structure of motion estimation circuit module according to the size of the module distribution was launched down, a variety of ways to generate an initial partition, applying partition algorithm and classification of the results of the assessment. The IDCT module and ME module using the design of three-dimensional integrated circuit partitioning algorithm is divided, from the number of cut edges, area utilization, and other aspects of the evaluation of the results of the division of the HEVC. IDCT the most appropriate division of the number of layers for the three layer, compared with the initial division, cutting edge number decreased by 92.55%, the area utilization rate reached 97.38%, increased by 10.46%. For the ME module, the selection of $K=3$, $\alpha=0.5$ as the optimal partition, compared with the initial division, the number of cut edges was reduced by 42.98%, the area utilization rate reached 95.51%, increased by 17.97%.