Кластеризация на графах и задача оптической коррекции

Е.Е. Егоров

Вводится система понятий и предлагается алгоритм обработки иерархических графов для решения задачи оптической коррекции в фотолитографическом процессе производства чипов. Алгоритм в некотором смысле аналогичен способу сжатия графической информации, и в частном случае, при $R = 0$, алгоритм доставляет некоторое решение задачи сжатия изображений.

В сфере производства полупроводниковых чипов методом литографии существует следующая задача, называемая Optical Proximity Correction (OPC): Задано черно-белое (точнее, прозрачно-непрозрачное) изображение И, представляющее геометрию некоторого слоя чипа. Подвергнувшись литографическому процессу, под действием оптических и прочих искажений, изображение И превращается в изображение И$'$.

Решить нужно обратную задачу, то есть при заданном И найти, какое И$_0$ надо нарисовать, чтобы в результате литографического процесса появился возможно более близкий к И образ.

Из физических свойств литографической установки следует, что с достаточной точностью можно считать имеющими место следующие свойства задачи OPC:

– Инвариантность относительно движений плоскости.

Это значит, что если заданный рисунок И получается движением Д из рисунка И$'$, то в качестве решения задачи OPC для И можно взять решение для И$'$ и подвергнуть его такому же сдвигу Д.

– Ограниченность пространственного влияния.

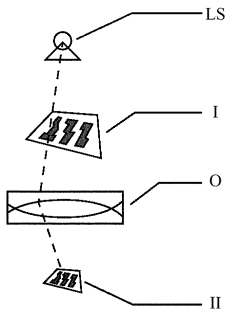

Рис. 1. LS — источник света, I — исходное изображение, O — оптическая система, II — преобразованное изображение.

Если два задания, И и И$'$, совпадают внутри области О на плоскости, то решения для $И_0$ и $И'_0$ тоже совпадают, только в несколько меньшей области $О_{-R}$. $О_{-R}$ есть подмножество О такое, что если x лежит в $О_{-R}$, то круг с центром в x и радиусом R лежит в О. Будем называть R радиусом влияния.

Дадим еще несколько определений. Рассмотрим двумерную плоскость $\mathbb{R}^2$ и ее подмножество $\mathbb{Z}^2$ точек с целочисленными координатами. Будем считать, что условия задач OPC, то есть слои чипов, а также и решения, задаются конечными наборами, возможно, пересекающихся между собой прямоугольников $\{\Pi_i\}$ с вершинами в $\mathbb{Z}^2$ и со сторонами, параллельными осям координат. Обозначим множество таких прямоугольников $\mathbf{\Pi}$.

Множество всех точек $\mathbb{R}^2$, принадлежащих хотя бы одному из прямоугольников $\{\Pi_i\}$, назовем *изображением* чипа $\{\Pi_i\}$. Наборы прямоугольников геометрически тождественны, если их изображения совпадают. Расстоянием между наборами прямоугольников назовем минимальное расстояние между изображениями наборов в метрике $d(x, y) = \max(|x_1 - x_2|, |y_1 - y_2|)$.

Движения $\mathbb{Z}^2$ образуются композициями параллельных переносов на целочисленные векторы, поворотов на углы кратные 90 градусам и отражениями от осей, параллельных осям координат. Обозначим множество таких движений $D(\mathbb{Z}^2)$.

Будем расматривать только прямоугольные области и в определении ограниченности влияния считаем O прямоугольником, а O_{-R} — уменьшенным со всех сторон на величину R прямоугольником.

Чтобы решить задачу с ограниченным влиянием, можно было бы разрезать всю площадь чипа на небольшие прямоугольники, для каждого прямоугольника взять окружение на расстоянии R, и решить все такие небольшие задачи. Однако на практике условия задач OPC состоят из порядка 10^8 прямоугольников, а решения часто имеют еще на порядок больше составляющих их прямоугольников, поэтому такой подход приводит к слишком большим временным затратам. Бороться с этой проблемой помогает то, что чипы имеют однородное строение, то есть состоят из повторяющихся частей. Описание чипа является иерархическим.

Пусть $\mathbf{C} = \{A_i\}$ — есть конечное множество, элементы которого будем называть ячейками. $\mathbf{C}$ будем называть *иерархической структурой.*

Каждая ячейка $A = \{e_i\}$ является также конечным множеством, элементами которого являются прямоугольники из $\mathbf{\Pi}$, а также элементы следующего вида: (T, φ), здесь T принадлежит $D(\mathbb{Z}^2)$ и есть преобразование координат, а φ есть функция из $\{1\}$ в $\mathbf{C}$, то есть указатель на какую-то ячейку из $\mathbf{C}$. Для краткости будем писать (T, A), если $\varphi : \{1\} \to A$. Такие пары будем называть вызовами ячеек и говорить что ячейка A вызывает ячейку B, если среди элементов B есть хотя бы один вызов A.

Мы потребуем, чтобы для $\mathbf{C}$ не существовало цепочек ячеек вида $AB_1B_2 \dots B_iA$, где каждая последующая ячейка вызывает предыдущую. Говорим тогда, что $\mathbf{C}$ корректна. В корректной иерархической структуре обязательно есть ячейки, которые не вызываются никакими другими ячейками. Будем называть такие ячейки корневыми, или чипами, а некорневые — подчипами. Будем считать далее, что в $\mathbf{C}$ имеется только одна корневая ячейка $C(\mathbf{C})$.

Графически иерархическую структуру можно изобразить в виде нагруженного дерева с корнем $D(\mathbf{C}) = (V, v_0, \varphi, R)$, определяемого следующим образом:
$\varphi : V \to \{A_i\}$, то есть функция нагрузки φ сопоставляет вершинам дерева ячейки иерархии $\mathbf{C}$,

$\varphi(v_0) = C(\mathbf{C})$,
(v_1, v_2) принадлежит $R \Leftarrow$ ячейка $\varphi(v_1)$ вызывает ячейку $\varphi(v_2)$.

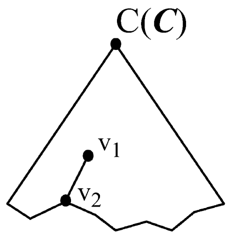

Рис. 2.

В дальнейшем нам понадобится использовать несколько операций над иерархическими структурами. Пусть в списке элементов некоторой ячейки A имеется вызов (T, B) ячейки B, а B есть $\{e_1 e_2 \dots e_n\}$. Заменим этот вызов на набор $T * e_1 T * e_2 \dots T * e_n$, где $T * e$ есть действие преобразования T на элемент e. $T * \Pi$ есть преобразованный прямоугольник, $T * (T', C)$ есть $(T * T', C)$. $T * T'$ есть произведение преобразований из $D(\mathbb{Z}^2)$. Такая замена называется *операцией взрывания вызова.*

Если иерархическая структура $\mathbf{C}$ задана корректно, то для любой ячейки A в результате конечного числа операций взрывания мы получим набор, состоящий только из прямоугольников. Такой набор называется *максимальным набором ячейки.* Очевидно, что максимальный набор корневой ячейки единственный, в смысле, один и тот же независимо от порядка проведения последовательности взрываний. Это и есть чип в обычном смысле, не иерархический.

Изображением ячейки назовем изображение, заданное максимальным набором этой ячейки. Изображением чипа, заданного иерархической структурой, назовем изображение, заданное максимальным набором корневой ячейки этой иерархической структуры.

Рассмотрим теперь преобразования иерархической структуры чипа, сохраняющие изображение чипа. Это операции взрывания и, обратно, выделения ячеек, а также, операция подстановки вызовов геометрически тождественных ячеек.

Пусть в иерархической структуре **C** ячейка A состоит из элементов $e_1e_2e_3 \dots e_n$. Выберем некоторое подмножество $\{f_j\}$ в множестве $\{e_i\}$. Добавим в **C** новую ячейку $B = \{f_j\}$, а в A заменим все елементы f на один вызов (E, B). Здесь E — тождественное преобразование координат. Это есть наша операция выделения ячейки, то есть, образования новой ячейки. Если окажется, что изображение ячейки B с некоторым преобразованием T переводится в изображение ячейки B', то B можно не добавлять в **C**, а вместо вызова (E, B) поставить в A вызов (T^{-1}, B'). Это отождествление и подстановка.

Утверждение 1. *Применение преобразований взрывания и выделения/отождествления ячеек к иерархической структуре не меняет изображения ячеек.*

Любой список элементов, получившийся из списка элементов ячейки A с помощью операций взрывания и выделения ячеек, назовем базовым списком для ячейки A. Множество ячеек, встречающихся в вызовах базового списка, назовем базовым множеством ячеек для ячейки A.

Главным понятием в данной работе является понятие R-покрывающей иерархии.

Дадим формальное определение этой иерархической структуры, подходящей для решения задачи с ограниченным влиянием. Пусть в иерархии **C** ячейка A вызывает ячейку B. Проведем максимально возможную серию операций взрывания элементов описания ячейки A, не трогая вызовы ячейки B. Получим список элементов, в котором, кроме прямоугольников, имеются только вызовы ячейки B, а именно, пусть это будут $(T_1, B), (T_2, B) \dots (T_k, B)$. Пусть теперь $\{\Pi_i\}$ — максимальный набор ячейки A. *Проекцией* A на B назовем набор прямоугольников $\{T_j^{-1}\Pi_i\}$.

Иерархическая структура **C** называется *R-покрывающей*, если каждой её ячейке A сопоставлен прямоугольник W_a так, что выполнены следующие два условия:

1. Если в каждой ячейке A структуры **C** рассмотреть ее максимальный набор $\{\Pi_i(A)\}$ и «отрезать лишнее», то есть заменить $\{\Pi_i(A)\}$ на набор $\{\Pi_i(A) \times W_a\}$, то изображение, задаваемое полученной новой структурой, совпадает с изображением исходной структуры **C**.

2. Для каждой ячейки A изображение проекции главной ячейки C на A совпадает с изображением ячейки A внутри прямоугольника W_a.

Опишем алгоритм построения R-покрывающей иерархии.

Возьмем корневую ячейку C и с помощью последовательности операций взрывания и выделения/отождествления ячеек построим некоторый базовый список $S = \{f_i\}$, пусть $f_i = (T_i, A_{j(i)})$. Теперь для каждого элемента f_i рассмотрим множество элементов $N(f_i, R)$, состоящее из тех элементов из списка S, расстояние от которых до f_i меньше или равно R. Образуем новую ячейку $A'_i = \{\{f_i\} \cup N(f_i, R)\}$.

Рассмотрим теперь список $S' = \{f'_i\}$, где $f'_i = (T_i, A'_i)$. Этот список получен заменой вызовов ячеек $A_{j(i)}$ на вызовы соответствующих ячеек A'_i, преобразования T_i остаются те же. И наконец образуем новую иерархию $\mathbf{C}'$ заменой корневой ячейки C на $C' = \{S'\}$. В этом списке проводим операции отождествления ячеек. Успех решения задачи связан именно с количеством отождествленных ячеек.

Нетрудно видеть, что справедливо следующее

Утверждение 2. *Иерархия $\mathbf{C}'$ является R-покрывающей иерархией.*

Утверждение 3. *Если для задачи OPC построена R-покрывающая иерархия, то ее решение можно получить, решив соответствующие задачи OPC для базовых ячеек.*

Действительно, для каждой базовой ячейки A R-покрывающей иерархии решим задачу OPC, то есть построим набор прямоугольников $\{P_i\}$, соответствующий решению задачи OPC. А теперь «отрежем лишнее». Вместо набора прямоугольников $\{P_i\}$ запишем набор пересечений $\{P'_i = P_i \times W_a\}$. Полученная таким образом иерархическая структура будет решением задачи OPC для всего чипа.

На практике такой способ решения задачи OPC является приемлемым по временным затратам. Оказывается, что отождествлений ячеек при проведении процедуры построения R-покрывающей иерархиеи бывает достаточно много, что дает способ решения задачи OPC, являющийся гораздо более эффективным, чем описанный вначале способ разрезания.

Можно многими способами выбрать базовый список для построения R-покрывающей иерархии. Количество базовых списков равно количеству поддеревьев в дереве иерархии $D(\mathbf{C}')$ с тем же корнем.

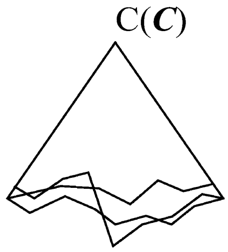

Рис. 3.

Чем меньше суммарная сложность базового набора ячеек, построенного по данному базовому списку, тем меньше времени потребуется на их решение. Для выбора базового набора ячеек воспользуемся следующим соображением. Пусть в базовом наборе вызывается всего K различных ячеек и пусть на расстоянии R от любого одного вызова другие вызовы могут располагаться N различными способами. Тогда, если базовый список достаточно длинный, то можно ожидать, что при применении описанного алгоритма постороения R-покрывающей иерархии нам встретятся все возможные в таких условиях конфигурации и мы получим в результате N^K базовых ячеек.

Отсюда видно, что в процессе построения базового списка при выборе очередного взрывания имеет смысл минимизировать в первую очередь число K — количество ячеек базового набора.

В частном случае, при $R = 0$, алгоритм постороения R-покрывающей иерархии превращается в алгоритм сжатия изображения, не теряющий информацию.