A – прямоугольная матринца размером $m \times n$

$$A = \begin{pmatrix} 100010000000000 \\ 010001000001000 \\ 000000110010000 \\ 000000001000110 \\ 001000000100000 \\ 000100000000001 \end{pmatrix}$$

Матрица B — прямоугольная матрица размеро $m \times n$ где m – число элементов n – число выводов

$$B = \begin{pmatrix} 111100000000000 \\ 000011100000000 \\ 000000011100000 \\ 000000000011100 \\ 000000000000011 \end{pmatrix}$$

Представление коммутационной схемы в виде Графа элементных комплексов

Матрица Q — прямоугольная матрица размером $m_1 \times m$

$$Q = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}$$

$$Q = BA^T$$

Взвешенный граф схемы

R – матрица взвешенного графа

$$R = \begin{pmatrix} 0 & 2 & 1 & 1 & 1 \\ 2 & 0 & 1 & 2 & 0 \\ 1 & 1 & 0 & 2 & 1 \\ 1 & 2 & 2 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

$$\widetilde{R} = QQ^T$$

Матрица $\widetilde{R}$ всегда отличается от матрицы R только элементами главной диагонали

$$x_0 \to \{x_1, x_2, x_3, x_4\}$$

$$x_1 \to \{x_0, x_2 < x_3\}$$

$$x_2 \to \{x_0, x_1, x_2, x_4\}$$

$$x_3 \to \{x_0, x_1, x_2, x_4\}$$

$$x_4 \to \{x_0, x_2, x_3\}$$

Z – массив отображений W – массив весовых коэфициентов V – массив границ

Реализация на Matlab алгоритмов преобразования различных видов информации

Под компоновкой понимают процесс перезода от логико-функционального описания описания устройства к конструктивному, предполагающий распределение элементов по группам, узлам и т. п. В качестве критерия компоновки обычно выступают:

- число узлов
- число маодульных соединений
- минимальное число типов ячеек

Различают точные и приближенные методы компотовки. точные:

- Метод полного перебора

приближенные:

- Последовательные алгоритмы компановки
- Итерационные алгоритмы компановки

На пракитике используют приближенные методы — методы последовательного заполнения, итерационные методы, смешанные методы.

Последовательные методы применяются для создания базового варианта компоновки при определенных ограничениях на число элементов узла и число выводов узла. Общем для всех последовательных алгоритмов является то, что на каждом шаге выбирается эелемент, с максимальным или минимальным значением некоторого критерия.

Итерационные методы — последовательное улучшение первичного варианта компоновки.

Смешанные методы — содержат последовательную и итерационную часть.

Ограниечения при компоновке.

1. Ограничение на число выводов в узле
2. Ограничение на число межузловых соединений
3. Ограничение на задержки распространения сигнала
4. Ограничение на допустимый объем конструкции

$A(x_i)x_K$ – общее кол-во связей x_k с формируемым узлом, включащим элементы в скобках $\Pi(x_i)x_k$ – кол-во пассивынх связей рассматриваемых элемета с оставшимися элеметами, не вошедшими в узел

КОС $= \frac{\text{А}(x_i)x_k}{A(x_i)x_k+\Pi(x_i)x_k}$ – коэфициент относительной связности.

Пример:

$$A(x_1)x_2 = 15$$

$$A(x_1)x_3 = 10$$

$$A(x_1)x_4 = 5$$

$$\Pi(x_1)x_2 = 2+4+2+3 = 12$$

$$\Pi(x_1)x_2 = 2+1+4 = 7$$

$$\Pi(x_1)x_2 = 3+1+2 = 6$$

$$\text{КОС}(x_1)x_2 = \frac{15}{27} = 0.56$$

$$\text{КОС}(x_1)x_3 = \frac{10}{17} = 0.59$$

$$\text{КОС}(X_1)x_4 = \frac{5}{11} = 0.45$$

Вывод: x_1 в один узел с x_3

Последовательный метод компоновки по связанности.

1. На каждом шаге выбирается один из нераспределенных элементов и включается в очередной узел. Узел считается завершенным если число эелеметов в узле равно заданному количеству или если включение любого из элементов приводит к образованию такого числа внешних связей, которое превышает допустимое количество. Элементам включаемым в узел является тот, который имеет наибольшее число связей с распределенными в узел элементами. три функции необходимые для определения:

   - L_1 – число цепей связывающих рассматриваемый элемент x_i с элементами схемы не вошедшеми в узлы (за исключением элемента x_0). Эта функция нужна только на 1й итерации.
   - L_2 – число внешних соединений формируемого узла.
   - L_3 – число связей с распределенными в узел элементами.

   Причем: $L_1 \to max; L_2 \to min; L_3 \to max$

$$\begin{pmatrix}
1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1\\
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0\\
1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1\\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0
\end{pmatrix}$$

Провести компановку элементов при условии: В каждом узле не более 3-х элементов. Каждый узел должен иметь не более 5 выводов.

Проводим задачу компановки:

1. Таблица:

r	J_r	L_1	i_r	J'_r	L_2	L_3	i_r^2	J''_r	L'_2	L'_3	i''_r
	x_1	3									
	x_2	3		x_2							
	x_3	3		x_3							
	x_4	3		x_4							
1	x_5	3	x_1	x_5	4	4					
	x_6	3		x_6							
	x_7	3		x_7							
	x_8	3		x_8							
	x_9	3		x_9							

Основой итерационных алгоритмов является использование процесса обмена местами элементов или группы элементов принадлежащих различным узлам, с целью минимизации некоторого притерия.

Итерационный алгоритм компоновки: $\Delta F(x_i, x_j) = -(D_{x_i} + D_{x_j} - 2r_{ij})$, Где r_{ij} – элемент матрицы R $D_{x_i} = A(x_i) - B(x_j)$, $A(x_i)$ – число внешних соединений x_i, $B(x_i)$ – число внутренних соединений x_i

Пример: В начальном варианте компоновке вся схема разбита на 2 узла.

1й содержит: $\begin{matrix} x_1\\ x_2\\ x_3\\ x_4 \end{matrix}$, 2й содержит $\begin{matrix} x_5\\ x_6\\ x_7\\ x_8 \end{matrix}$

$$R = \begin{pmatrix}
0 & 1 & 0 & 0 & 1 & 3 & 0 & 0\\
1 & 0 & 2 & 0 & 0 & 0 & 2 & 0\\
0 & 2 & 0 & 1 & 4 & 0 & 0 & 0\\
0 & 0 & 4 & 0 & 0 & 5 & 0 & 3\\
1 & 0 & 4 & 0 & 0 & 2 & 0 & 0\\
3 & 0 & 0 & 5 & 2 & 0 & 0 & 2\\
3 & 0 & 0 & 5 & 2 & 0 & 0 & 2\\
0 & 2 & 0 & 0 & 0 & 0 & 0 & 1\\
0 & 0 & 0 & 3 & 0 & 2 & 1 & 0
\end{pmatrix}$$

$$D_{x_1} = 1 + 3 - 1 = 3$$
$$D_{x_2} = -1$$
$$d_{x_3} = 1$$
$$D_{x_4} = 7$$
$$D_{x_5} = 3$$
$$D_{x_6} = 4$$
$$D_{x_7} = 1$$
$$D_{x_8} = 0$$

Далее найдем все возможные пересетановки и выделим среди них целесообразные

$$\Delta F_{15} = 2r_{15} - D_{x_1} - D_{x_5} = 2 * 1 - 3 - 3 = -4$$

Далее по формуле

$$\Delta F_{16} = -1$$
$$\Delta F_{17} = -4$$
$$\Delta F_{18} = -3$$
$$\Delta F_{25} = -2$$
$$\Delta F_{26} = -3$$
$$\Delta F_{27} = 4$$

– число положительное поэтому нецелесообразная перестановка

$$\Delta F_{28} = 1$$
$$\Delta F_{35} = 4$$
$$\Delta F_{36} = -5$$
$$\Delta F_{37} = -2$$
$$\Delta F_{38} = -1$$
$$\Delta F_{45} = -10$$
$$\Delta F_{46} = -1$$
$$\Delta F_{47} = -8$$
$$\Delta F_{48} = -1$$

Наиболее целесообразной перестановкой является перестановка F_{45} – выбраем ее и пересчитываем все заного до тех пор пока не будет отрицаетльных перестановок

МСВД — минимальная суммарная взвешенная длина соединений

Алгоритмы размещения элементов:

- Непрерывнодискретные
  - Градиентные алгоритмы
  - Алгоритмы, построенные на динамических моделях
- Дискретные
  - Метод ветвей и грациц
  - Последовательные алгоритмы
  - Параллельные алгоритмы
  - Последовательно-параллельные алгоритмы
  - Матричные схемы
  - Итерационные алгоритмы

Пусть электрическая схема описывается матрицей R, размером $n \times n$ и существует коммутационное поле с фиксированным размером $m \times n$. Если $m > n$ то вводим $m-n$ фиктивных эелементов не имеющих связей с остальными элементами связей. Коммутационное поле описывается Матрице D, которая представляет собой матрицу расстояний мужду i j позициями

$$d = |x_i - x_j| + |y_i - y_j|$$

$$D = \begin{pmatrix} 0 & 1 & 2 & 1 & 2 & 3 & 2 & 3 & 4 & 3 & 4 & 5 \\ 1 & 0 & 1 & 2 & 1 & 2 & 3 & 2 & 3 & 4 & 3 & 4 \\ 2 & 1 & 0 & 3 & 2 & 1 & 4 & 3 & 2 & 5 & 4 & 3 \\ 1 & 2 & 3 & 0 & 1 & 2 & 1 & 2 & 3 & 2 & 3 & 4 \\ 2 & 1 & 2 & 1 & 0 & 1 & 2 & 1 & 2 & 3 & 2 & 3 \\ 3 & 2 & 1 & 2 & 1 & 0 & 3 & 2 & 1 & 4 & 3 & 2 \\ 2 & 3 & 4 & 1 & 2 & 3 & 0 & 1 & 2 & 1 & 2 & 3 \\ 3 & 2 & 3 & 2 & 1 & 2 & 1 & 0 & 1 & 2 & 1 & 2 \\ 4 & 3 & 2 & 3 & 2 & 1 & 2 & 1 & 0 & 3 & 2 & 1 \\ 3 & 4 & 5 & 2 & 3 & 4 & 1 & 2 & 3 & 0 & 1 & 2 \\ 4 & 3 & 4 & 3 & 2 & 3 & 2 & 1 & 2 & 1 & 0 & 1 \\ 5 & 4 & 3 & 4 & 3 & 2 & 3 & 2 & 1 & 2 & 1 & 0 \end{pmatrix}$$

$$\text{МСВД} = \frac{1}{2} \sum_{i=1 j \neq s}^{n} \sum_{i=1 j \neq s}^{n} r_{ij} d_{p(i)p(j)} + \sum_{j=1}^{n} r_{is} d_{p(i)p(s)}$$

$p(i)$ – Позиция i-го элемента. s – инедкс жеско закреплённого элемента

Основные этапы алгоритма:

1. В первый ряд позиций назначается первый элемент x_i из числа незафиксированных
2. Определяются значения функции МСВД в каждой из рассматриваемых позиций. Выбирается тот элемент значение функции которого минимально

3. Выбирается следующий элемнт и устанавливается в любую из позиций 2го ряда и т.д.

$$R = \begin{pmatrix} 0 & 1 & 5 & 10 & 2 & 4 & 1 & 1 & 3 \\ 1 & 0 & 4 & 7 & 1 & 5 & 1 & 1 & 1 \\ 5 & 4 & 0 & 3 & 0 & 1 & 0 & 0 & 1 \\ 10 & 7 & 3 & 0 & 1 & 3 & 0 & 0 & 1 \\ 2 & 1 & 0 & 1 & 0 & 2 & 0 & 1 & 1 \\ 4 & 5 & 1 & 3 & 2 & 0 & 2 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 2 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 3 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \end{pmatrix}$$

$$F_j^i$$

– МСВД при размещении i-го элемента в j-ю позицию

$$F_1^1 = r_{17} d_{p(1)p(7)} = r_{17} d_{12} = 1$$

$$F_3^1 = r_{17} d_{p(1)p(7)} = r_{17} d_{32} = 1$$

Теперь второй элемент:

$$F_4^2 = r_{27} d_{p(2)p(7)} + \frac{1}{2} r_{21} d_{p(2)p(1)} = r_{27} d_{42} + \frac{1}{2} r_{21} d_{41} = 5.5$$

$$F_4^2 = r_{27} d_{p(2)p(7)} + \frac{1}{2} r_{21} d_{p(2)p(1)} = r_{27} d_{52} + \frac{1}{2} r_{21} d_{51} = 3$$

$$F_4^2 = r_{27} d_{p(2)p(7)} + \frac{1}{2} r_{21} d_{p(2)p(1)} = r_{27} d_{62} + \frac{1}{2} r_{21} d_{61} = 5.5$$

Мы получили 3 значения, надо выбрать наименьшее значение среди полученных, т е второй вариант (3) Третий элемент:

$$F_7^3 = r_{37} d_{p(3)p(7)} + \frac{1}{2} r_{31} d_{p(3)p(1)} + \frac{1}{2} r_{32} d_{p(3)p(2)} = 9$$

$$F_8^3 = 9.5$$

$$F_9^3 = 14$$

Учитывая полученные значения, наиболее вакантным вариантом является 1 (9) Далее пятый элемент:

$$F_4^5 = r_{57} d_{p(5)p(7)} + \frac{1}{2} r_{52} d_{p(5)p(2)} + \frac{1}{2} r_{53} d_{p(5)p(3)} + \frac{1}{2} r_{54} d_{p(5)p(4)} = 4$$

$$f_6^5 =$$

Конструктивные алгоритмы размещения.

Особенностью этих алгоритмов является то, что используется максимум связности, который не зависит от положения элементов. Очередным размещаемым элементом будет тот, который имеет максимальную характеристику связанности. Размещение происходит на таком расстоянии чтобы сумма расстояний между этим элементом и оставшимися была минимальна.

Матричные схемы выбора размещения. Основа для выбора элемента и позиции на k-ом шаге является специальная квадратная матрица M размером $n-k+1$ в которой элемент представляет "цену"назначения элемента x_i в позицию p_j. Предполагается что k элементов уже размещены. Выбор элемента и позиции основан на принципе минимального риска. Матрица M в каждой строке и каждом столбце находятся 2 минимальных элемента. Затем определяется максимум разности этих минимальных элементов. Определяется индекс элемента с максимумом разности и индекс позиции, которой элемент матрицы M минимален. Оправданием использования данной тактики служит минимизация потерь от назначения элементов, которые приводят к большим приращениям.

$$m = \begin{pmatrix} 7 & 3 & 2 & 8 & 17 & 10 \\ 5 & 9 & 8 & 7 & 3 & 6 \\ 1 & 0 & 0 & 9 & 14 & 4 \\ 0 & 3 & 9 & 10 & 1 & 3 \\ 5 & 6 & 6 & 7 & 1 & 8 \\ 1 & 1 & 2 & 2 & 13 & 14 \end{pmatrix}$$

Далее находим наименьший и второй элемент по степени меньшинства в столбце, вычитая из одного другое, записываем в row-major вектор:

$$m = \begin{pmatrix} 1 & 1 & 2 & 5 & 0 & 1 \end{pmatrix}$$

Т к наибольшая разница находится в столбце x_4, то элементом для перемещения выбираем x_4, и ставим его в место с наименьшим значением в этом столбце.

Параллельно-последовательные алгоритмы размещения.

Пусть задано множество размещаемых элементов матрицы R и множество позиций коммутационного поля матрицы D.

1. Для каждого элемента определяется суммарное число всех его соединений с остальными соединениями.

2. Для каждой позиции матрицы D находят сумму всех элементов $d_i = \Sigma d_{ij}$

3. Упорядочиваем элементы r_i по возрастанию.

4. Упорядочиваем позиции d_i по убыванию.

5. Соответствующая пара упорядоченных элементов и позиций задает размещение элементов.

$$R = \begin{pmatrix} 0 & 2 & 2 & 3 & 9 & 4 \\ 2 & 0 & 1 & 10 & 5 & 6 \\ 2 & 1 & 0 & 3 & 4 & 7 \\ 3 & 10 & 3 & 0 & 7 & 8 \\ 9 & 5 & 4 & 7 & 0 & 11 \\ 4 & 6 & 7 & 8 & 11 & 0 \end{pmatrix}$$

$$D = \begin{pmatrix} 0 & 1 & 2 & 3 & 4 & 5 \\ 1 & 0 & 1 & 2 & 3 & 4 \\ 2 & 1 & 0 & 1 & 2 & 3 \\ 3 & 2 & 1 & 0 & 1 & 2 \\ 4 & 3 & 2 & 1 & 0 & 1 \\ 5 & 4 & 3 & 2 & 1 & 0 \end{pmatrix}$$

$$r = \begin{pmatrix} 20 & 24 & 17 & 31 & 36 & 36 \end{pmatrix}$$

$$d = \begin{pmatrix} 15 & 11 & 9 & 9 & 11 & 15 \end{pmatrix}$$

Располагаем элементы по возрастанию r:

$$x_3, x_1, x_2, x_4, x_5, x_6$$

Располагаем позиции по убыванию d

$$p_1, p_6, p_2, p_5, p_3, p_4$$

Отсюда видим зависимость расположения элементов.

Итерационные алгоритмы размещеия.

На каждом этапе рачитываем некоторую функцию, смотрим ее приращение от попарной перестановки элементов. Если приращение положительно, то перестановка имеет смысл.

Градиентные методы размещения.

$$\text{МСВД} = F(x_i)$$

$$x_{new} = X_{o|d} - \mu gradF$$

μ – Параметр скорости градиентного спуска.

1. Поиск оптимального решения в непрерывных координатах
2. Округление полученных координат в существующию координатную сетку

$$\Sigma(x_i - A_i)^2 + (y_i - B_i)^2 \to min$$

К достоинствам градиентных методов относятся сравниетльно небольшие затраты машинного времени на поиск экстремума целевой функции. А так же наличие стандартных программ для решения данного класса задач. Недостатками являются: возможность получения лишь локального экстремума. Низкая эффективновность при пологом экстремуме. Большая неравномерность распределения элементов

Алгоритмы использующие динамические модели.

1. см. 1й этап градиентные методы

2. сила притяжения пропорциональна числу электрических связей Сила отталкивания обратно пропорциональна расстоянию между материальными точками. Чтобы устранить возникновение незатухающих колебаний, вводят силы сопротивления среды, пропорциональной скорости движения материальных точек.

3. См. 2й этап градиентных методов

Задача трассировки.
Алгоритмический метод трассировки соединений.
Критерий трассировки

1. Минимум суммарной длины соединений.

2. Минимум числа пересечений проводников.

3. Равномерность распределения проводников на плате.

4. Минимальная протяженность параллельных участков соседних проводников.

5. Минимум числа изгиба проводников.

6. Минимум числа слоев.

Исходными данными для задачи трассировки является список цепей. Параметры конструкции элементов. Данные по размещению элементов.

Виды монтажных соединений

1. цепь

2. звезда

3. ортогональные соединения

4. Соединения типа вывод- проводник

5. Соединения типа проводник-проводник

Трассировка проводных соединений: Алгоритм прима

Алгоритм Прима позволяет организовывать просмотр ребер графа. Который связывает вершины строящегося поддерева. С новыми еще неприсоеденными вершинами

1. На первом шаге - любая произвольная вершина соединяется с ближайшей соседней. На каждом последующем шаге присоединяют очередное ребро минимально возможной длины, связывающею новую, еще неприсоединенную вершину. С одной из вершин поддерева

Для реализации этого алгоритма первоначально составляется матрица длин D. Просматриваются жлементы 1й строкии и находятся минимальные из них. Пусть таким эелментом оказался эелментом оказался элемент j-го столбца, тогда весь первый и j-й исключают, а соединения проводят между точками 1 и j. Просматривают 1ю и j-ю строки матрицы с оставшимися элементами. Снова находят минимальный. Индекс столбца будет соответствовать индексу точки в графе и т. д.

Пример:

$$D = \begin{pmatrix} 0 & 10 & 4 & 10 & 8 & 16 & 7 & 15 & 21 \\ 10 & 0 & 8 & 6 & 10 & 6 & 15 & 13 & 11 \\ 4 & 8 & 0 & 6 & 4 & 12 & 7 & 11 & 17 \\ 10 & 6 & 7 & 0 & 4 & 6 & 9 & 7 & 11 \\ 8 & 10 & 4 & 4 & 0 & 8 & 5 & 7 & 13 \\ 16 & 6 & 12 & 6 & 8 & 0 & 11 & 9 & 5 \\ 7 & 15 & 7 & 9 & 5 & 11 & 0 & 8 & 14 \\ 15 & 13 & 11 & 7 & 7 & 9 & 8 & 0 & 6 \\ 21 & 11 & 17 & 11 & 13 & 5 & 14 & 6 & 0 \end{pmatrix}$$

Трассировка печатных соединений.

Распределение соединений по слоям.

Расслоение до трассировки.

Простейшим способ расслояния до трассировки является выделение преимущественных направлений группирования соединений. Например: Все горизонтальные отрезки в один слой, а вертикальные в другой.

Для многослойных печатных плат в качестве направлений по слоям могут быть выбраны оси симметрии, секторов некоторого круга.

Метод расслоения с использованием охватывающих прямоугольников. При ортогональной прокладке трасс строится минимальный охватывающих прямоугольник трасс. Число прямоугольников соответствует максимально возможному числу слоев. Комплексы цепей называются перекрывающимися если их пересечение не пустое множество.

Из анализа пересечения прямоугольников строят граф пересечений

Алгоритм раскраски графа. Исходной информацией для раскраски графа является матрица смежности вершин R_s в которой 0 елементы главной диагонали заменены на единицы.

Вначале просматриывается первая строка матрицы R_s до первого нуля стоящего в Q-м столбце проводится логическое сложение первой и q-й строк. Если результат сложения состоит из одних единиц, то первое множество несоединенных вершин найдено. Если реазультат содержит нули то просматривают его до первого нуля стоязем в r-том столбце. Процесс продолжается до тех пор пока результат не будет состоять из одних единиц.

$$R_s = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 070 & 0 & 0 & 0 & \\ 1 & 1 & 0 & 1 & 0 & 070 & 1 & 0 & \\ 0 & 1 & 070 & 1 & 1 & 0 & 0 & 1 & \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 \\ 070 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & \end{pmatrix}$$

Алгоритм (ли)

Все ячейки монтажного поля подразделяют на занятые и свободные. Занятыми считаются ячейки в которых уже расположены проводники или находятся монтажные выводы элементов а так же ячейки соответствующие границе платы и запрещенным для прокладывания проводников участка.

На множестве свободных ячеек коммутационного поля моделируют волну влияния одной ячейки на другую. Первая ячейка в которой зарождается волна называется источник. Ячейка в которую должна прийти волна называется приемником. Чтобы иметь возможность следить за прохождением фронта волны его фрагментам на каждом присваивают определенные веса

$$P_{\text{к}} = P_{k-1} + \Delta P$$

P_k – вес k-го фронта волны P_{k-1} – Вес $k-1$-го фронта волны ΔP – определяется критерием оптимизации

Процесс распространения волны продолжает до тех пор, пока ее фронт не достигнет приемника или не возможно достигнуть приемника, в последнем случае проведение трассы не представляется возможным. Если волна достигла приемника, то осуществляют проведение пути, двигаясь от приемника к источнику, в направлении монотонного убывания весов.

Пример. Чтобы исключить неопределенность при проведении пути вводят понятие путевых координат задающих предпочтительность поведения пути, каждое направление кодируют двоичным чилослом. Пердпочтительным будет то направление, которое имеет меньший код.

Вес незанятой ячейки k-го фронта считают равным весу соседнему $k-1$ фронта плюс число соседних ячеек, через которые проходят ранее построенные проводники. Занятыми являются ячейке конструктивных элементов, имеются изгибы или пересечения ранее построенных проводников, а так же ячейки в которых направления проводников совпадают с путевой координатой строящегося пути. Вес ячейки k-го фронта равен весу ячейки соседнего $k-1$-го фронта, если в этой ячейке нет ранее построенного проводника и на единцу больше в противном случае.

Проведение пути с минимальным числом изгибов.

Вес незанятой ячейки k-го фронта считается равной весу ячейке $k-1$-го фронта если путевая координата в этой ячейке не меняется

Параллельная оптимизация пути по нескольким параметрам.

$P_k = P_{k-1} + 1$ Если в данной и соседних ячейках нет ранее построенных проводников и путевая координата не меняется свое направление. $P_k = P_{k-1} + 2$ Если путевая координа меняет направление, но в соседних ячейках нет ранее построенных проводников. $P_k = P_{k-1} + 3$ Если в данной ячейке данная путевая координата не меняет направления, но в соседней ячейке есть проводники. $P_k = P_{k-1} + 5$ Если в данной ячейке происходит пересечение с ранее построенным проводником.

Путь где изменение ΔP_k минимально является наиболее предпочтительным.

Модификации волнового алгоритма.

Метод кодирования весов по модулю 3.

Метод кодирования весов по Акерсу (1 - 1 - 2 - 2)

Метод магистральной трассировки.

Построение пути производится путем исследования пространства магистралей (линий) а не отдельных ячеек.

Если магистрали первого уровня не пересекаются, то строят магистрали второго уровня. Они перпендикулярны магистралям первого уровня и проводятся через точки в узлах основной сетки.

При переходе от магистрали 1-го уровня к магистрали 2-го уровня добавляется по крайней мере 1 поворот. Алгоритм минимизирует количество изгибов, но не длину. Рекомендуется использовать для двусторонних ПП. Минимизирует количество межслойных переходов.

Распределение внешних выводов в конструктивных узлах

Рассмотрим один из методов учитывающий плоскостные ограничения при распределении внешних узлов.

Для каждой внешней цепи V составим список выводов (кандидатов) для назначения. Упорядочим внешние цепи таким образом, чтобы приоритет отдавался цепи имеющей наименьшее число выводов в списке.

Осуществляем трассировку этой цепи выбрав в цписке тот вывод, который приводит к наименьшему увеличению общей длины соединений. Удаляем этот вывод всех остальных списков, вновь сортируем цепи и продолжаем процесс до распредееления всех выводов.

Пример:

Решение задач.