

Вопрос № 1

Автоматизация задач топологического синтеза при конструировании новых объектов

На этапе топологического синтеза решаются задачи создания обобщенной структуры конструируемого объекта. При этом решаются задачи компоновки элементов и трассировки связей между ними.

Компоновка - размещение элементов на плоскости или в объеме, обеспечивающих выполнение технологической машиной заданных функций.

Трассировка – соединение элементов заданными связями.

При решении задач топологического синтеза важнейшая роль отводится конструктору. При этом используются следующие приемы:

- 1) Метод проб и ошибок
- 2) Конструктивная преемственность
- 3) Метод трансформации и инверсии
- 4) Метод аналогии
- 5) Метод мозгового штурма

На этапе топологического синтеза две задачи (компоновки и трассировки) успешно решаются с помощью вычислительной техники. Задачи взаимосвязаны и решаются как правило с использованием различных алгоритмов.

Задачи компоновки и алгоритмы их решения

Это построение сложного объекта из элементов более низкого уровня.

Дано: набор элементов (количество и размеры), критерии сравнения компоновочных решений.

- А) Площадь (объем), занимаемый скомпонованной схемой. (мин)
- Б) Суммарная длина всех связей между элементами (мин)
- В) Экономические затраты (мин)

Ограничения:

- 1) Элементы не могут располагаться в одном месте пространства;
- 2) Связи не могут пересекаться;
- 3) Удобство обслуживания (около каждого элемента должна быть зона, свободная от других элементов);

- 4) Иногда необходимо предусмотреть минимально расстояние между связывающими линиями;
- 5) Элементы или связь не могут располагаться в некоторых, заранее определённых местах;
- 6) Заданная площадь (объём), на которой требуется скомпоновать схему.

Постановка задачи компоновки:

Пусть необходимо разместить n элементов на плоскости, аппроксимируя каждый элемент прямоугольником, в качестве базовой точки возьмём элементарную точку.

Пусть даны связи между элементами. Необходимо найти координаты базовых точек всех n элементов, при которых площадь, занимаемая всеми n элементами минимальна, при выполнении перечисленных выше ограничений. Таким образом, в данном случае, мы имеем плоскую задачу, где площадь сводится к минимуму.

Алгоритмы решения задач компоновки:

- 1) Метод полного перебора;
- 2) Последовательный алгоритм: размещение элемента осуществляется с учётом размещения предыдущих, на первом этапе – ранжирование элементов по важности, на втором осуществляется размещение одного наиболее важного и т. д.

Достоинство – быстрота.

Недостатки – не всегда оптимально решение.

- 3) Параллельно последовательный алгоритм:

Конструктором задано некоторое базовое размещение части элементов, после чего остальные элементы размещаются по последовательному алгоритму.

- 4) Итерационный – задаётся исходное размещение всех элементов, делается любым методом нелинейного программирования (градиентным, симплексным и т.д.), шаг путём изменения положения элементов. Если шаг удачен, то движение продолжается в данном направлении. Исходное размещение может быть случайным, а может быть решением задач компоновки последовательным или параллельно последовательным алгоритмом.

- 5) Эвристический алгоритм.

Задачи трассировки и алгоритмы их решения

Компоновочная схема – исходные данные, на ней уже размещены элементы. Плюс связи между элементами, критерий – суммарная длина всех связей.

Ограничения:

- 1) Трасса не может проходить через элемент
- 2) Трассы не могут пересекаться
- 3) Расстояние между трассами не менее заданного
- 4) Расстояние между трассой и элементом не менее заданного
- 5) Количество поворотов каждой трассы не более заданного
- 6) Длина трассы от i -го к $i+1$ -ому элементу не более заданной

Постановка задачи трассировки:

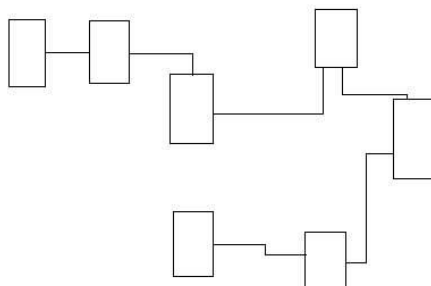
Необходимо соединить все скомпонованные элементы таким образом, чтобы суммарная длина всех связей была минимальной при выполнении ограничений.

Алгоритмы решения:

- 1) Волновой

Основан на дискретном принципе оптимальности Беллмена (во-первых распространение числовой волны от входа к выходу, во-вторых – в обратном направлении).

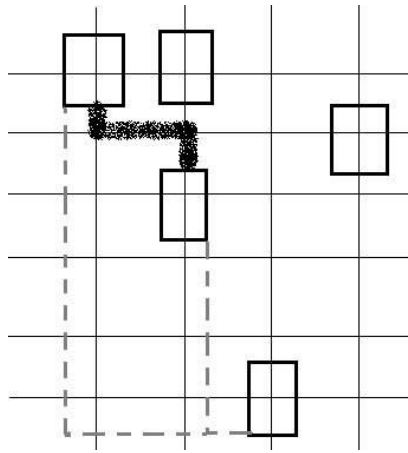
90% занимает расчет волны в прямом направлении, а 10% - в обратном.



- 2) Лучевой

Суть состоит в проложении трасс вдоль некоторых лучей.

Частный случай – канальный алгоритм, согласно которому трассы могут располагаться лишь в горизонтальном и вертикальном положении.



- 3) Итерационные алгоритмы – задается начальная или исходная трассировка
- 4) Эвристические

Вопрос № 2

Пакеты прикладных программ, используемые в САПР

Прикладное программное обеспечение представляют *пакеты прикладных программ (ППП)* для выполнения различных *проектных процедур*. Они разрабатываются на основе единого внутреннего представления графической и текстовой информации, единого входного языка, строятся *по модульному принципу* и ориентированы на использование непрограммистом-проектировщиком.

Различают несколько типов *ППП* в зависимости от состава пакета. *Пакеты прикладных программ простой* структуры характеризуются наличием только обрабатывающей части - набора функциональных программ (модулей), каждая из которых предназначена для выполнения только одной проектной процедуры. *Объединение* нужных модулей осуществляется средствами операционной системы ЭВМ.

Пакеты прикладных программ сложной структуры и *программные системы* появились в результате развития прикладного программного обеспечения. В первых из них имеется собственная управляющая часть - *монитор*, во вторых, кроме того, - *языковой процессор* с проблемно-ориентированным входным языком. Программные системы вместе с соответствующим лингвистическим и

информационным обеспечением называют программно-методическими комплексами САПР.

Управляющая часть программного обеспечения имеет иерархическую организацию, и в общем случае в ней можно выделить различные уровни: уровень операционных систем *вычислительной сети*, операционных систем отдельных ЭВМ, мониторных систем САПР и мониторов отдельных ППП.

AutoCad - универсальный графический пакет, предназначенный для любого специалиста, работающего с технической графикой. Фирма Autodesk, ориентируясь на самый широкий круг пользователей, заложила в пакет богатые возможности адаптации к любым предметным областям. Именно поэтому AutoCad завоевал широкую популярность и продолжает сохранять свои позиции на мировом рынке.

Кроме автоматизации собственно чертежно-графических работ, AutoCad с его расширениями (AutoShade, AutoFlix, 3D-STUDIO и др.) предоставляет следующие возможности:

1. графическое моделирование, т.е. использование компьютера в САПР в качестве мощного вычислительного средства, позволяющего без особых навыков программирования работать со сложными пространственными моделями;
2. создание и ведение информационной базы данных (архива) чертежей;
3. создание библиотеки стандартных элементов чертежей, относящихся к какой-то предметной области, с тем чтобы строить новые чертежи из уже созданных ранее элементов;
4. параметризация чертежей - построение деталей и чертежей с новыми размерами на основе один раз нарисованного чертежа (модели);
5. создание демонстрационных иллюстраций и мультфильмов.

КОМПАС 5 представляет собой современный программный продукт, функционирующий под управлением операционной системы Windows 95/98/NT.

Система имеет настраиваемый оконный интерфейс, соответствующий стандартам Windows, и управляется с помощью команд текстового меню, панелей кнопок, контекстно-зависимых (динамических) меню. Оформление экрана, состав кнопочных панелей и любые параметров системы могут быть настроены

непосредственно во время сеанса работы. Пользователь может формировать собственные кнопочные панели, в том числе подключая функции прикладных библиотек в качестве команд.

Поддерживается одновременная работа с несколькими документами, а также отображение каждого документа в нескольких окнах. Реализованы различные режимы резервного копирования загруженных документов.

Печать разработанных документов может выполняться на любых устройствах (принтерах или плоттерах), поддерживаемых Windows. Реалистичное изображение документов в режиме предварительного просмотра позволяет компоновать на поле вывода и распечатать одновременно несколько документов. Обеспечена гибкая настройка всех параметров печати. В состав системы входит утилита для разработки собственных драйверов перьевых устройств вывода(плоттеров).

КОМПАС 5 поддерживает технологию OLE, что позволяет вставить документы КОМПАС в любой документ, являющийся OLE-контейнером (например, в документ MS Word). Созданный таким образом OLE-объект в дальнейшем можно просматривать при помощи КОМПАС-Viewer или редактировать средствами КОМПАС. Если при вставке OLE-объекта сохранена связь с источником, то все вносимые в источник изменения будут отражаться в документе-контейнере.

Основными компонентами КОМПАС 5 являются КОМПАС-ГРАФИК –редактор конструкторской документации и КОМПАС-3D –система трехмерного твердотельного моделирования.

Графический редактор позволяет разрабатывать выпускать различные документы —эскизы, чертежи, схемы, плакаты и т.д. В системе предусмотрены два вида графических документов —чертежи и фрагменты. Чертеж обладает рамкой и основной надписью, в нем можно создавать до 255 видов (проекций, разрезов, сечений), имеющих разный масштаб изображения. На листе чертежа могут быть размещены спецификация, технические требования, знак неуказанной шероховатости. Фрагмент содержит изображение в натуральную величину без элементов оформления (рамки, технических требований и т.п.).

Любой вид чертежа или фрагмент может содержать до 255 слоев, каждый из которых можно делать текущим или недоступным для редактирования или невидимым.

Вопрос № 3

Составить алгоритм поиска экстремума функции двух переменных методом покоординатного спуска

$$F(x_1, x_2) = x_1x_2 + x_1^2x_2 + x_1x_2^2$$

Смысл метода в фиксировании одной из переменных и изменении другой, после нахождения первой оптимальной переменной, она фиксируется и начинает меняться другая, пока не находится оптимальное решение.

- 1) Выбираем $X_0(x_{10}, x_{20})$ -произвольно, $\epsilon=0.00001, \sigma=0.001$
- 2) Фиксируем $x_2=\text{const}$
- 3) $x_{1n+1}=x_{1n}+\sigma$
- 4) Если $f(x_{1n+1}, x_2) < f(x_{1n}, x_2)$, то шаг 3 иначе шаг 5
- 5) Фиксируем $x_1=\text{const}$
- 6) $x_{2n+1}=x_{2n}+\sigma$
- 7) Если $f(x_1, x_{2n+1}) < f(x_1, x_{2n})$, то шаг 6 иначе шаг 8
- 8) Если $((x_{1k+1} - x_{1k})^2 + (x_{2k+1} - x_{2k})^2) > \epsilon$ то шаг 2 иначе шаг 9
- 9) Вывод x_1^*, x_2^*