2 Математические модели объектов дискретной оптимизации

Формализованное решение задач структурного анализа и синтеза невозможно без наличия математических моделей объектов проектирования.

Требования, предъявляемые к математической модели, определяются ее назначением. Так как модель является средством описания объекта проектирования, а само проектирование выполняется посредством ее преобразования и/или анализа, то возможность формальной постановки задачи зависит от степени формализации описания объекта и наличия математического аппарата, позволяющего выполнять преобразования модели.

2.1 Требования к математическим моделям

С точки зрения возможности и эффективности выполнения формальных преобразований к математической модели объекта предъявляются следующие *требования*:

- высокая степень *формализации* отображаемого объекта;
- наличие математического аппарата, позволяющего выполнять *формальные преобразования* модели;
- адекватность модели, т.е. *полнота* и *правильность* отображение в модели всей информации об объекте, которая является *существенной* для решения данного класса задач.

От *адекватности математических моделей* объекта и результата проектирования зависят точность и детерминированность формализованного решения задач структурного синтеза. *Правильность* отображения информации обеспечивается *корректностью правил перехода от объекта к модели*.

Требования к математическим моделям

- *Правила перехода* устанавливают соответствия между

компонентами объекта и элементами математической модели, а также соответствия их отношений.

Для схем ЭВМ это связано главным образом с решением вопроса о выборе способа представления электрической цепи.

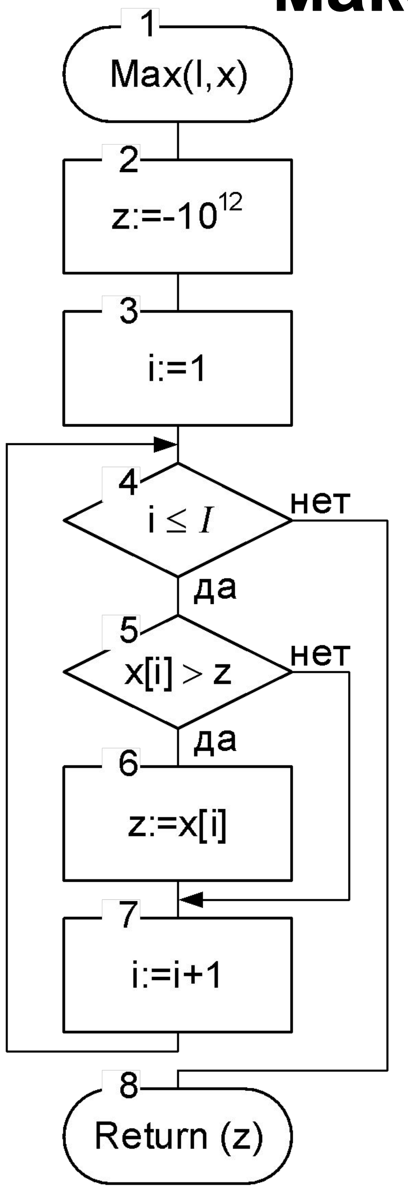
Результаты решения задач являются основными исходными данными для выпуска соответствующей технической документации. В связи с этим должна быть обеспечена однозначность перехода от модели к объекту.

- В наибольшей степени сформулированным выше основным требованиям удовлетворяет граф, являющийся содержательной моделью схемы

Математические модели объектов

- «В виде графов можно представлять блок-схемы программ (вершины – блоки, а дуги – разрешенные переходы от одного блока к другому), электрические цепи, географические карты и молекулы химических соединений, связи между людьми и группам людей» – Судоплатов С.В., Овчинникова Е. В. Элементы дискретной математики: Учебник, 2002.
- «Занимаясь статистической механикой Уленбек обозначал точками (вершинами) молекулы, а смежность вершин толковал как взаимодействие наибольшей близости (соседства) некоторого физического типа, например магнитное притяжение или отталкивание» – Харари Ф. Теория графов, 1973.
- «Учение о цепях Маркова… связано с ориентированными графами в том смысле, что события представляются вершинами, а ориентированное ребро (дуга), идущее из одной вершины в другую, указывает на то, что вероятность прямого перехода от одного события к другому положительна» – Там же.

Пример. Модель алгоритма поиска максимального элемента массива



Условные обозначения :

⊗ – терминальные блоки : t_1 - начало, t_6 - конец;

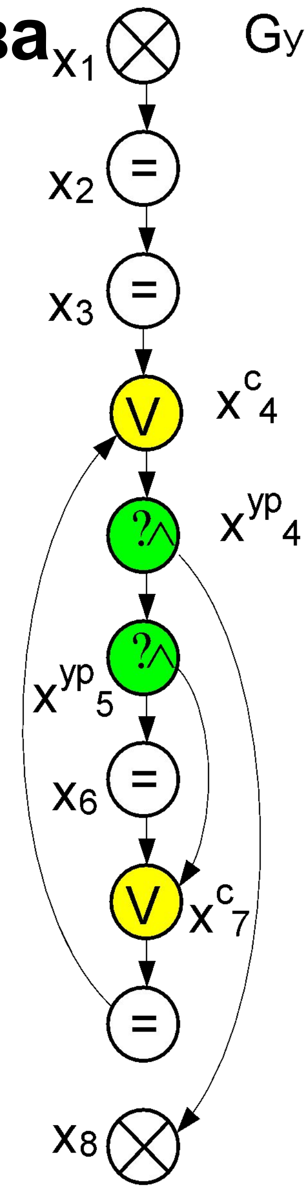
= – t_2 - обработка данных;

? – t_3 - вычисление условий;

Λ – t_4 - ветвление потоков управления ;

?Λ – $t_{3,4}$ - объединенный блок вычисления условия и ветвления;

V – t_5 - слияние потоков управления



2.2. Графы: ультра-, гипер- и обыкновенные

2.2.1. Общее определение графа.

1. *Граф* – множество вершин X, на элементах которого определены *двуместные отношения смежности* – $(x_j, x_i) \in F$, где $x_j, x_i \in X$.

   Тогда пара вершин, находящихся в отношении смежности, рассматривается как *ребро* $u_k = (x_j, x_i)$, $u_k \in U$.

2. *Граф* – два *непересекающиеся* множества: X – вершин и U –ребер, на элементах которых $x \in X$, $u \in U$ определён трёхместный предикат-инцидентор $P(X,U,X)$. Предикат-инцидентор $P(X,U,X)$ является конъюнкцией *двуместных предикатов-отношений инцидентности* $\Gamma_1(X, U)$ –«вершинам множества X инцидентны ребра множества U» и $\Gamma_2(U, X)$ – «ребрам множества U инцидентны вершины множества X»:

$$P(X, U, X) = \Gamma_1(X,U) \& \Gamma_2(U,X),$$

$$P(X,U,X)=«и»: \Gamma_1(x_i,u_j)=«и»\&\Gamma_2(u_j,x_k))=«и»/ x_j,x_k \in X, u_j \in U\}.$$

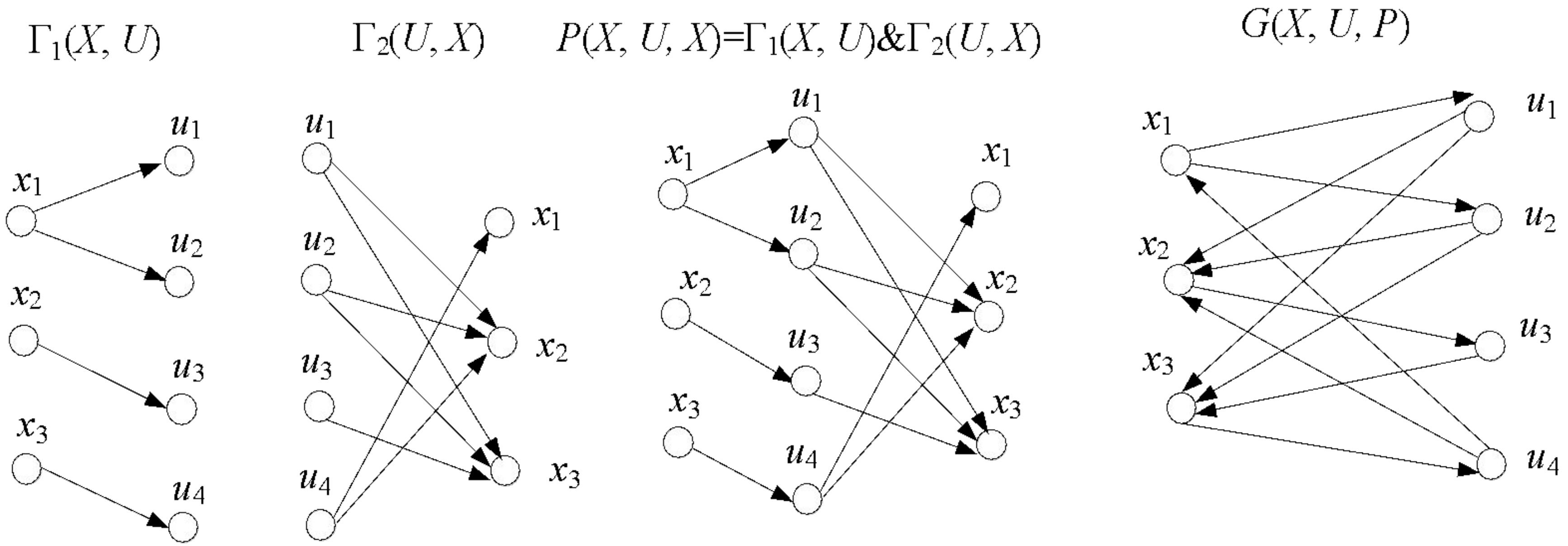
Общее определение графа

Положим, что при $X=\{x_1,x_2,x_3\}$ и $U=\{u_1,u_2,u_3,u_4\}$ предикаты $\Gamma_1(X,U)$ и $\Gamma_2(U,X)$ принимают значение «истина» на парах:

$\Gamma_1(X,U)$: (x_1,u_1), (x_1,u_2), (x_2,u_3), (x_3,u_4),

$\Gamma_2(U,X)$: (u_1,x_2), (u_1,x_3), (u_2,x_2), (u_2,x_3), (u_3,x_3), (u_4,x_1), (u_4,x_2).

Тогда геометрическая интерпретация предикатов $\Gamma_1(X,U)$, $\Gamma_2(U,X)$ и их конъюнкции будет иметь вид, показанный на рис. *а* и *б,* а граф – на рис. *в*.



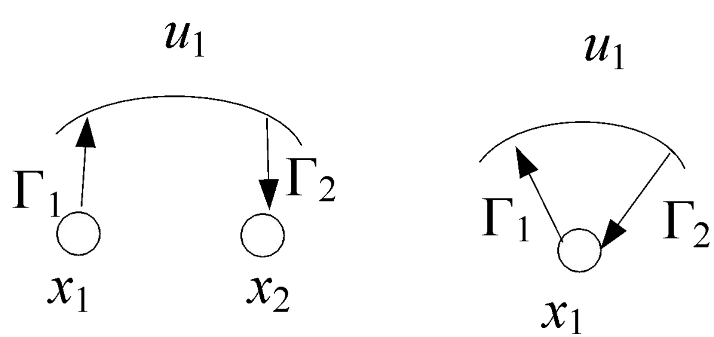
а *б* *в*

Общее определение графа

Предикаты $Г_1(X,U)$ и $Г_2(U,X)$ таковы, что для всех графов при $X \neq \varnothing$ и $U \neq \varnothing$:

$$Г_1(x_i,u_j)=«и»\&Г_2(u_j,x_k)=«и» \vee Г_1(x_i,u_j)=«и»\&Г_2(u_j,x_i) = «и»). \quad (1)$$

Данное условие устанавливает возможность существования в графе петель. Геометрическая интерпретация выражения (1) при $X=\{x_1,x_2\}$, $U=\{u_1\}$, $Г_1(x_1,u_1) = «и»$, и $Г_2(u_1,x_2) = «и»$, $Г_1(x_1,u_1)=«и»$, $Г_2(u_1,x_1) = «и»$ показана на рис. 1.



1

Виды графов

Данная трактовка графов допускает существование в них петель и кратных ребер. Вид графа – *обыкновенные неориентированный и ориентированный, гипер- и ультраграф* – определяется свойствами предикатов инцидентности $Г_1(X,U)$ и $Г_2(U,X)$, которые порождают свойства предикатов смежности и соответствующих им отношений.

Отношения смежности

На элементах множеств X и U определены также отношения смежности $F_1(X,X)$ и $F_2(U,U)$. Например, вершине x_i *смежна* вершина x_k, если существует ребро u_j, *инцидентное* x_i, такое, что x_k *инцидентно ему.* Аналогично ребру u_j *смежно ребро* u_l, если существует вершина x_i, *инцидентная* ребру u_j, такая, что u_l *инцидентна* этой вершине. Таким образом, понятие смежности *вторично* по отношению к понятию инцидентности.

Отношения смежности

В соответствии с определением понятия «смежность» предикат смежности $F_1(X, X)$ является композицией предикатов $\Gamma_1(X,U)$ и $\Gamma_2(U,X)$:

$$F_1(X, X) = \Gamma_1(X, U) \cdot \Gamma_2(U, X),$$

где $F_1(X, X) = \{F_1(x_i, x_t) = «и»: \exists u_j\ (\Gamma_1(x_i, u_j) = «и» \& \Gamma_2(u_j, x_t) = «и») \ /\ x_i, x_t \in X, u_j \in U\}$.

Здесь · – символ операции композиции.

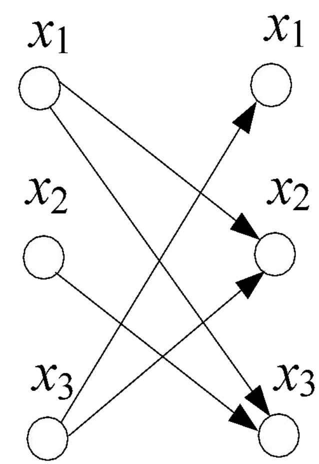
Предикат смежности $F_1(X, X)$, полученный композицией предикатов $\Gamma_1(X, U)$ и $\Gamma_2(U, X)$ изображён на рисунке *г*.

Аналогично предикат смежности рёбер графа является композицией предикатов $\Gamma_2(U, X)$ и $\Gamma_1(X, U)$:

$F_2(U, U) = \Gamma_2(U, X) \cdot \Gamma_1(X, U)$, где

$F_2(U, U) = \{F_2(u_j, u_k)=«и»: \exists x_i(\Gamma_2(u_j, x_i)=«и» \ \& \ \Gamma_1(x_i, u_k) = «и») \ /\ u_j, u_k \in U, x_i \in X\}$.

$$F_1(X, X)=\Gamma_1(X, U)\bullet\Gamma_2(U, X)$$



г

2.3 Предикаты-свойства

Определим *одноместные предикаты-свойства*, производные от предикатов $Г_1$, $Г_2$, F_1, F_2 (подстановка в двуместный предикат некоторого определенного элемента того или иного множества превращает его в одноместный):

- зафиксировав в $Г_1(X,U)$ некоторую вершину $x_i \in X$, получим предикат-свойство $Г_1x_i(U)$ – «вершине x_i инцидентны ребра множества U», истинность которого задается i-й строкой матрицы предиката $Г_1(X,U)$, а подставив некоторое ребро $u_j \in U$, получим предикат-свойство $Г_1u_j(X)$ – «ребро u_j инцидентно вершинам множества X». Истинность этого предиката определяет j-й столбец матрицы предиката $Г_1(X,U)$;

Предикаты-свойства

- зафиксировав в предикате $Г_2(U,X)$ ребро u_j, придем к предикату-свойству $Г_2\, u_j(X)$ – «ребру u_j инцидентны вершины множества X», истинность которого задает j-я строка матрицы предиката $Г_2(U,X)$, а подставив вершину x_i, получим предикат-свойство $Г_2 x_i(U)$ – «вершина x_i инцидентна ребрам множества U». Истинность данного предиката задает i-й столбец матрицы предиката-отношения $Г_2(X,U)$.

Предикаты-свойства

Характеристические множества рассмотренных предикатов-свойств будем обозначать через $\Gamma_1 x_i$, $\Gamma_1 u_j$, $\Gamma_2 u_j$, $\Gamma_2 x_i$, где:

- $\Gamma_1 x_i = U_{1i} = \{u_j \in U : \Gamma_1 x_i(u_j) =$ «и» $\}$, $U_{1i} \subseteq U$ – ребра, инцидентные вершине $x_i \in X$;
- $\Gamma_1 u_j = X_{1j} = \{x_i \in X : \Gamma_1 u_j(x_i) =$ «и» $\}$, $X_{1j} \subseteq X$ – вершины, которым инцидентно ребро $u_j \in U$;
- $\Gamma_2 u_j = X_{2j} = \{x_i \in X : \Gamma_2 u_j(x_i) =$ «и» $\}$, $X_{2j} \subseteq X$ – вершины, инцидентные ребру $u_j \in U$;
- $\Gamma_2 x_i = U_{2i} = \{u_j \in U : \Gamma_2 x_i(u_j) =$ «и» $\}$, $U_{2i} \subseteq U$ – ребра, которым инцидентна вершина $x_i \in X$.

Предикаты-свойства

Зафиксировав в $F_1(X,X)$ некоторую вершину $x_i \in X$, получим предикат-свойство $F_1x_i(X)$ – «вершине x_i смежны вершины множества X», истинность которого задается i-й строкой матрицы предиката $F_1(X,X)$, а подставив в $F_2(U,U)$ некоторое ребро $u_j \in U$, – предикат-свойство $F_2\,u_j(U)$ – «ребру u_j смежны ребра множества U». Истинность этого предиката определяет i-я строка матрицы предиката $F_2(U,U)$.

Характеристические множества рассмотренных предикатов-свойств будем обозначать через F_1x_i, F_2u_j, где:

- $F_1x_i = X_{1i} = \{x_j \in X : F_1x_i(x_j) = «и» \}$, $X_{1i} \subseteq X$ –вершины, смежные вершине $x_i \in X$;
- $F_2u_j = U_{2j} = \{ u_i \in U : F_2u_j(u_i) = «и» \}$, $U_{2j} \subseteq U$ –ребра, смежные ребру $u_j \in U$;

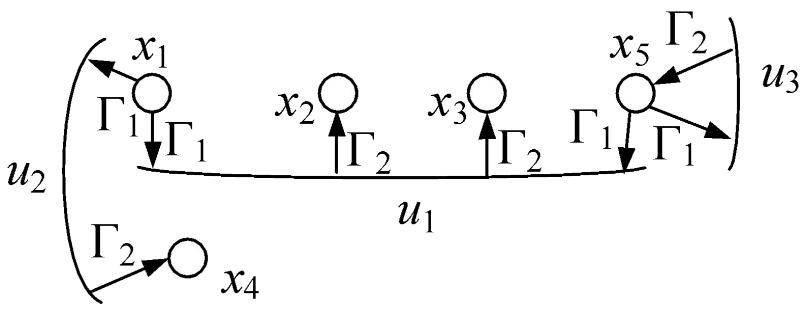
2.4 Ультраграфы

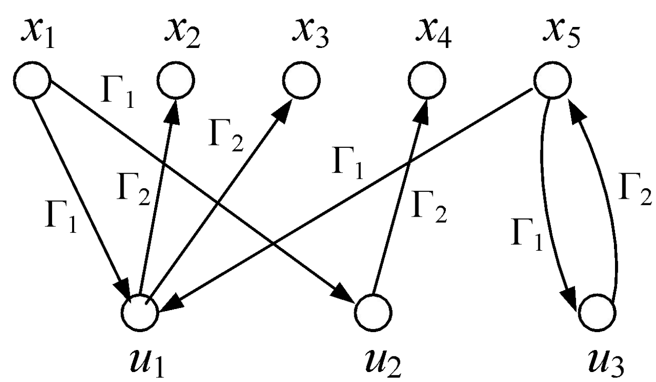
Определенный выше граф называется ультраграфом $H_U(X,U,\Gamma_1,\Gamma_2)$, если предикаты $\Gamma_1(X,U)$ и $\Gamma_2(U,X)$ обладают следующим свойством

$$\exists\ u_j \in U\ \ (|\Gamma_1 u_j| + |\Gamma_2 u_j|\ ) > 2,$$

(2)

т. е. в графе есть хотя бы одно ребро, суммарное количество вершин, которым оно инцидентно и которые инцидентны ему, больше двух.





Представление ультраграфа матрицами инцидентности

Полным и достаточно наглядным способом формального задания ультраграфа является его представление через две матрицы инцидентности A_1 и A_2, где A_1– матрица истинности предиката $Г_1(X,U)$ и A_2 – матрица истинности предиката $Г_2(U,X)$.

Матрица инцидентности A_1, задающая связь между вершинами и ребрами, – это прямоугольная матрица размером $n×m$, где $n = |X|$, $m = |U|$. Элементы этой матрицы определяются по правилу

$$a_{1\,i,j} = \begin{cases} 1 – \text{если } Г_1(x_i,u_j) = «и» \\ 0 – \text{если } Г_1(x_i,u_j) = «л» \end{cases},$$

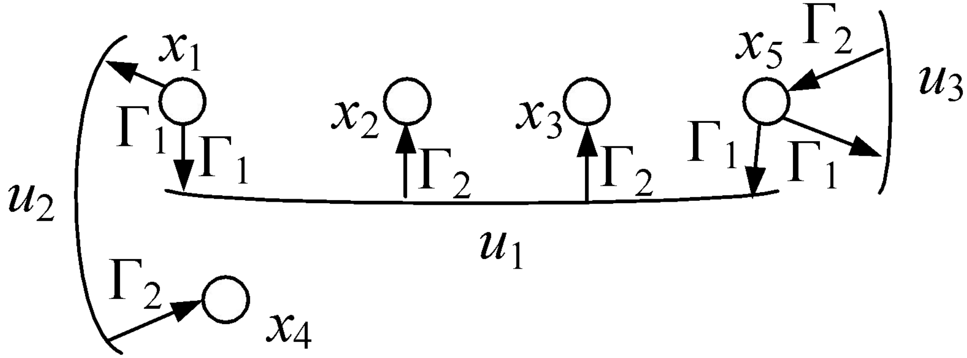
где $i = \overline{1, n}$; $j = \overline{1, m}$.

Представление ультраграфа матрицами инцидентности

Матрица инцидентности A_2 задает связь между ребрами и вершинами. Ее строки соответствуют ребрам, а столбцы – вершинам (размер матрицы $m \times n$). Элементы матрицы A_2 определяются по правилу

$$a_{2\,j,i} = \begin{cases} 1 \text{ – если } \Gamma_2(u_j, x_i) = \text{«и»}, \\ 0 \text{ – если } \Gamma_2(u_j, x_i) = \text{«л»}. \end{cases}$$

Представление ультраграфа матрицами инцидентности



$$A_1 = \begin{array}{c|ccc|} & u_1 & u_2 & u_3 \\ x_1 & 1 & 1 & 0 \\ x_2 & 0 & 0 & 0 \\ x_3 & 0 & 0 & 0 \\ x_4 & 0 & 0 & 0 \\ x_5 & 1 & 0 & 1 \end{array} , \qquad A_2 = \begin{array}{c|ccccc|} & x_1 & x_2 & x_3 & x_4 & x_5 \\ u_1 & 0 & 1 & 1 & 0 & 0 \\ u_2 & 0 & 0 & 0 & 1 & 0 \\ u_3 & 0 & 0 & 0 & 0 & 1 \end{array}$$

Аналитическое представление ультраграфа

Аналитически ультраграф полностью задается множествами X, U и образами этих множеств относительно предикатов $Г_1(X,U)$ и $Г_2(U,X)$.

Образом множества A относительно предиката Q(A,B) является множество

$$QA = \{Qa_i \,/\, a_i \in A\},$$

где $Qa_i = \{\, b_j \in B : Qa_i(b_j) = «и»\}$,– характеристическое подмножество предиката-свойства $Qa_i(B)$, т. е. образ элемента $a_i \in A$ относительно этого предиката.

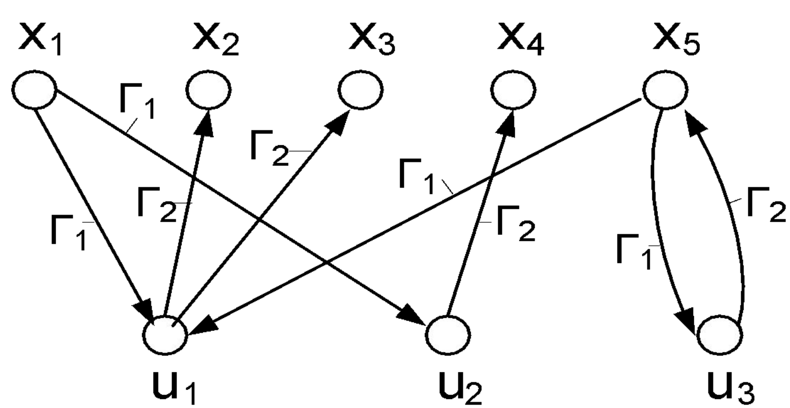
В ультраграфе $H_u(X, U, Г_1X, Г_2U)$:

$Г_1X=\{Г_1x_i/x_i \in X\}$, $Г_1x_i = U_{1i} \subseteq U$– *образ вершины* $x_i \in X$ (*инцидентные ей ребра*;

$Г_2U=\{Г_2u_j/u_j \in U\}$, $Г_2u_j = X_{2j} \subseteq X$– *образ ребра* $u_j \in U$, (*инцидентные ему вершины*).

Пример аналитического представления ультраграфа

Ультраграф данным способом будет задан, если заданы множества вершин *X*, ребер *U* и их образы.



$H_u(X, U, \Gamma_1 X, \Gamma_2 U)$: $X = \{x_1, x_2, x_3, x_4, x_5\}$, $U = \{u_1, u_2, u_3\}$,
$\Gamma_1 X = \{\Gamma_1 x_i \,/\, i = 1,5\}$, где: $\Gamma_1 x_1 = \{u_1, u_2\}$, $\Gamma_1 x_2 = \Gamma_1 x_3 = \Gamma_1 x_4 = \varnothing$, $\Gamma_1 x_5 = \{u_1, u_3\}$,
$\Gamma_2 U = \{\Gamma_2 u_j \,/\, j = 1,3\}$, где: $\Gamma_2 u_1 = \{x_2, x_3\}$, $\Gamma_2 u_2 = \{x_4\}$, $\Gamma_2 u_3 = \{x_5\}$.

Аналитическое представление ультрграфа образами и прообразами вершин и ребер

Рассмотренное представление ультраграфа, является полным, однако в ряде случаев затрудняет выполнение формальных преобразований и просмотр структуры ультраграфа. Например, для того чтобы определить, каким вершинам инцидентно ребро $u_j \in U$, необходимо проверить принадлежность этого ребра всем $\Gamma_1 x_i$ и сформировать подмножество вершин согласно выражению:

$$X_j = \{ x_i \in X : u_j \in \Gamma_1 x_i \}.$$

Аналогичное замечание справедливо и для определения множества ребер, которым инцидентна вершина $x_i \in X$.

Для задания таких множеств воспользуемся понятием «прообраз множества относительно предиката». По определению прообраз множества – это его образ относительно обратного предиката.

Аналитическое представление ультраграфа образами и прообразами вершин и ребер

Элементы обратных предикатов $\Gamma_2^{-1}(X,U)$ и $\Gamma_1^{-1}(U,X)$ определяются по правилам:

$\forall u_j \in U,\ \forall x_i \in X\ \ (\Gamma_2^{-1}(x_i, u_j) = «и» : \Gamma_2(u_j, x_i) = «и»)$ и

$\forall x_i \in X,\ \forall u_j \in U\ \ (\Gamma_1^{-1}(u_j, x_i) = «и» : \Gamma_1(x_i, u_j) = «и»)$

соответственно, т.е. таблицы истинности этих предикатов получаются транспонированием матриц истинности A_2 и A_1.

Отсюда *множество ребер U_{2i}, которым инцидентна вершина $x_i \in X$* – ее прообраз относительно предиката $\Gamma_2(U,X)$, – это характеристическое множество *i*-го вектора строки матрицы истинности предиката $\Gamma_2^{-1}(X,U)$ или *i*-го вектора-столбца матрицы A_2, т. е. предиката-свойства $\Gamma_2 x_i(U)$:

$$U_{2i} = \Gamma_2 x_i = \{u_j \in U : \Gamma_2 x_i(u_j) = «и»\}, \qquad U_{2i} \subseteq U .$$

Прообразом множества X относительно предиката $\Gamma_2(U,X)$ будет множество характеристических подмножеств предикатов-свойств $\Gamma_2 x_i(U)$:

$$\Gamma_2 X = \{\Gamma_2 x_i \,/\, x_i \in X\}.$$

Аналитическое представление ультраграфа образами и прообразами вершин и ребер

Аналогично множество вершин, которым инцидентно ребро $u_j \in U$ – его прообраз $Г_1 u_j$ относительно предиката $Г_1(X,U)$ –это характеристическое множество X_{1j} предиката-свойства $Г_1^{\,-1} u_j(X)$, соответствующего j-у вектору-строке матрицы истинности предиката $Г_1^{-1}(U,X)$, или предиката-свойства $Г_1 u_j(X)$, задаваемого j-м вектором-столбцом матрицы A_1 :

$$X_{1j} = Г_1 u_j = \{\, x_i \in X : Г_1 u_j(x_i) = «и» \,\}, \qquad X_{1j} \subseteq X.$$

Прообразом множества U относительно предиката $Г_1(X,U)$ является множество характеристических подмножеств предикатов-свойств $Г_1 u_j(X)$:

$$Г_1 U = \{Г_1 u_j \,/\, u_j \in U \}.$$

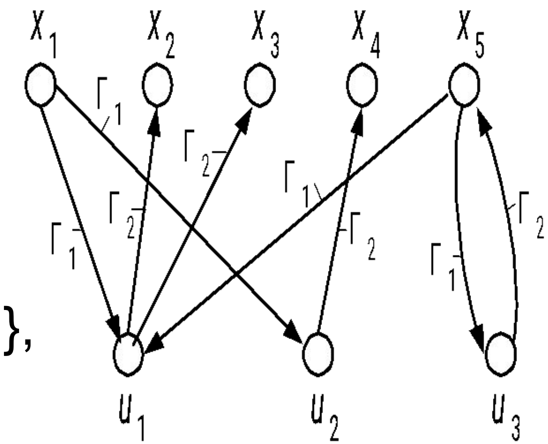
Аналитическое представление ультраграфа образами и прообразами вершин и ребер

Геометрическая интерпретация предикатов $\Gamma_1^{-1}(U,X)$ и $\Gamma_2^{-1}(X,U)$ показана на рисунке *б*.

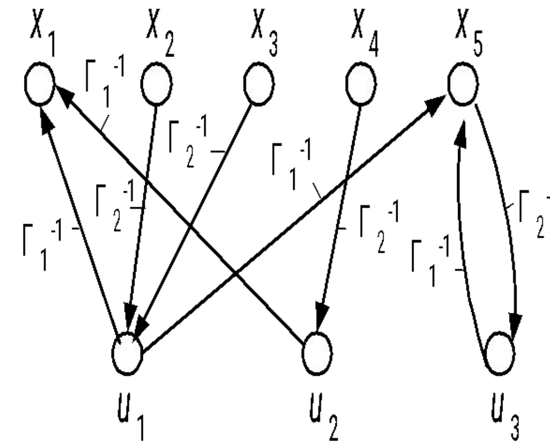
$\Gamma_2 X$: $\Gamma_2 x_1 = \varnothing$, $\Gamma_2 x_2 = \{u_1\}$, $\Gamma_2 x_3 = \{u_1\}$, $\Gamma_2 x_4 = \{u_2\}$, $\Gamma_2 x_5 = \{u_3\}$,

$\Gamma_1 U$: $\Gamma_1 u_1 = \{x_1, x_5\}$, $\Gamma_1 u_2 = \{x_1\}$,

$\Gamma_1 u_3 = \{x_5\}$.



à



á

Для данного способа представления ультраграф будем обозначать $H_u(X, U, \Gamma_1 X, \Gamma_2 X, \Gamma_1 U, \Gamma_2 U)$.

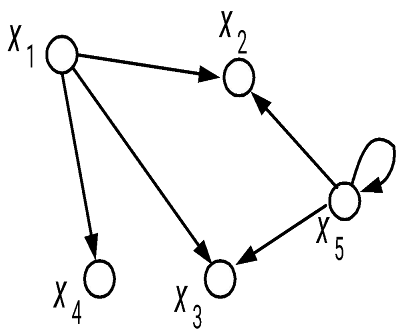
Представление ультраграфа матрицами смежности

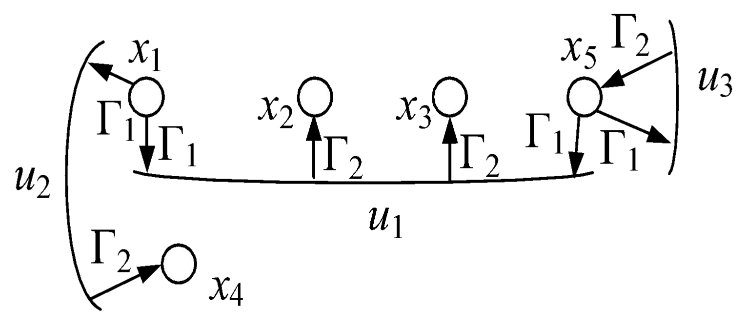
Предикат смежности вершин $F_1(X, X)$. Вершине x_i *смежна* вершина x_t, если существует ребро u_j, инцидентное вершине x_i, такое, что вершина x_t инцидентна ему. Отсюда элементы матрицы смежности R_1 определяются по правилу:

$$r_{1i,t} = \begin{cases} 1 - \text{если } \exists\ a_{1i,j} = 1\ \&\ a_{2j,t} = 1, \\ 0 - \text{в противном случае,} \end{cases}$$

где $i, t = 1, n$; $n = |X|$, $j = 1, m$; $m = |U|$.

$$R_1 = \begin{vmatrix} 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{vmatrix}$$





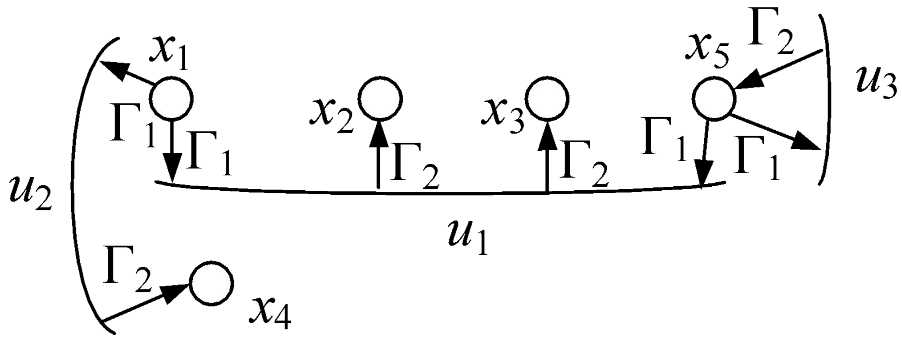
Представление ультраграфа матрицами смежности

Предикат смежности ребер $F_2(U, U)$. Ребру u_j *смежно ребро* u_k, если существует вершина x_i инцидентная ребру u_j, такое, что ребро u_k инцидентно ей. Отсюда элементы матрицы смежности R_2 определяются по правилу:

$$r_{2j,k} = \begin{cases} 1 \text{ – если } \exists\ a_{2j,i} = 1 \ \&\ a_{1i,k} = 1, \\ 0 \text{ – в противном случае,} \end{cases}$$

где $j,k = 1,m$; $m = |U|$, $i = 1,n$; $n = |X|$.

$$R_2 = \begin{vmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{vmatrix}.$$



Совокупность предикатов $F_1(X, X)$ и $F_2(U, U)$ задает ультраграф не ***полностью***

Образ и прообраз множества *X* относительно предиката смежности вершин $F_1(X, X)$

Образ и пробраз вершины x_i – это характеристические множества предикатов-свойств $F_1x_i(X)$, и $F_1^{-1}x_i(X)$:

- $F_1x_i = X_{1i} = \{x_j \in X : F_1x_i(x_j) = «и» \}$, $X_{1i} \subseteq X$ –вершины, смежные вершине $x_i \in X$;
- $F_1^{-1}x_i = X_{1i}^{-1} = \{ x_j \in X : F_1^{-1}x_i(x_j) = «и» \}$, $X_{1i}^{-1} \subseteq X$ –вершины, которым смежна вершина $x_i \in U$. *Они задаются i-ми вектором-строкой и вектором-столбцом матрицы R_1 соответственно:*

Для нашего ультраграфа

$F_1X = \{F_1x_i / i = 1,5\}$:

$F_1x_1 = \{x_2, x_3, x_4\}$,

$F_1x_2 = F_1x_3 = F_1x_4 = \varnothing$,

$F_1x_5 = \{x_2, x_3, x_5\}$,

$F_1^{-1}X = \{F_1^{-1}x_i / i = 1,5\}$:

$F_1^{-1}x_1 = \varnothing$,

$F_1^{-1}x_2 = \{x_1, x_5\}$,

$F_1^{-1}x_3 = \{x_1, x_5\}$.

$F_1^{-1}x_4 = \{x_1\}$,

$F_1^{-1}x_5 = \{x_5\}$.

Образ и прообраз множества U относительно предиката смежности ребер $F_2(U, U)$.

Образ и прообраз ребра u_i – это характеристические множества предикатов-свойств $F_2u_j(U)$ и $F_2^{-1}u_j(U)$:

- $F_2u_j = U_{2j} = \{ u_i \in U : F_2u_j (u_i) = «и» \}$, $U_{2j} \subseteq U$ –ребра, смежные ребру $u_j \in U$;
- $F_2^{-1}u_j = U_{2j}^{-1} = \{ u_i \in U : F_2^{-1}u_j (u_i) = «и» \}$, $U_{2j}^{-1} \subseteq U$ –ребра, которым смежно ребро $u_j \in U$; *Они задаются j-ми вектором-строкой и вектором-столбцом матрицы R_2 соответственно:*

Для рассматриваемого ультраграфа

$F_2U = \{F_2u_j / j = 1,3\}$:

$F_2u_1 = F_2u_2 = \varnothing$,

$F_2u_3 = \{u_1, u_3\}$.

$F_2^{-1}U = \{F_2^{-1}u_j / j = 1,3\}$:

$F_2^{-1}u_1 = \{u_3\}$,

$F_2^{-1}u_2 = \varnothing$, $F_2^{-1}u_3 = \{u_3\}$.

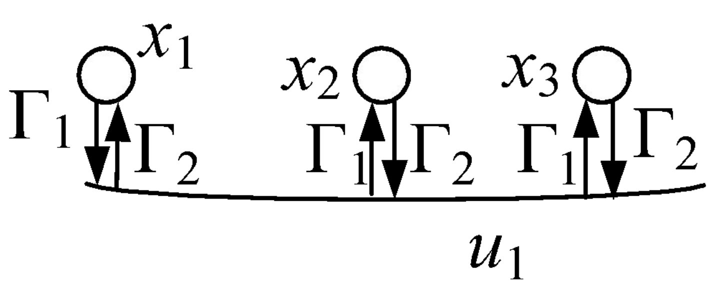
2.5 Гиперграфы

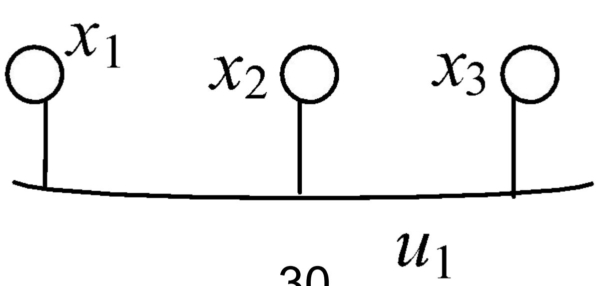
Данный вид графа получим в соответствии со сформулированным выше определением *при выполнении условий (1)* в том случае, когда

$\forall x_i, x_k \in X, \forall u_j \in U$ ((Г$_1$(x_i,u_j)= «и» & Г$_2$(u_j,x_i)= «и») (3, *а*)

и $\exists\ u_j \in U$ (|Г$_2 u_j$| >2. (3, *б*)

Условие (3) означает, что *предикат-отношение* Г$_2$(*U*, *X*) *является обратным к предикату-отношению* Г$_1$(*X*, *U*). Тогда гиперграф можно определить как два непересекающихся множества вершин *X* и ребер *U*, на элементах которых задана пара двуместных предикатов-отношений инцидентности Г$_1$(*X*,*U*) и Г$_2$(*U*,*X*) таких, что Г$_2$(*U*, *X*)= Г$_1^{-1}$(*X*, *U*).





Гиперграфы

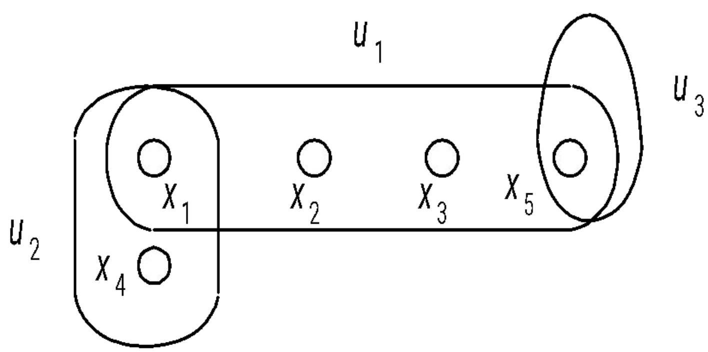
Вектор-строка таблицы истинности двуместного предиката-отношения $\Gamma_1(X,U)$ – матрицы инцидентности вершины-ребра A_1 *совпадает с вектором-столбцом* таблицы истинности предиката $\Gamma_2(U,X)$ – матрицы инцидентности ребра-вершины A_2. По определению предикаты эквивалентны, если их таблицы истинности совпадают. Отсюда следует, что:

- предикат-свойство $\Gamma_1 x_i(U)$ – «вершине x_i инцидентны ребра множества U» *эквивалентен* предикату-свойству $\Gamma_2 x_i(U)$ – «вершина x_i инцидентна ребрам множества U»,
- предикат-свойство $\Gamma_2 u_j(X)$ – «ребру u_j инцидентны вершины множества X» *эквивалентен* предикату-свойству $\Gamma_1 u_j(X)$ – «ребро u_j инцидентно вершинам множества X».

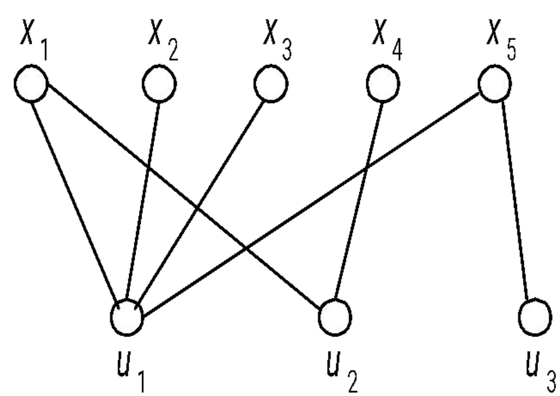
Гиперграфы

Отсюда, ***гиперграф будет полностью задан***, если заданы *множество вершин X, ребер U и один из предикатов* $\Gamma_1(X,U)$ *или* $\Gamma_2(U,X)$.

При геометрическом представлении гиперграфа в литературе вершины изображаются кружками, ребра – в виде контуров, а расположение вершины x_i внутри ребра u_j означает истинность $\Gamma_1(x_i,u_j)$, следовательно и $\Gamma_2(u_j,x_i)$.

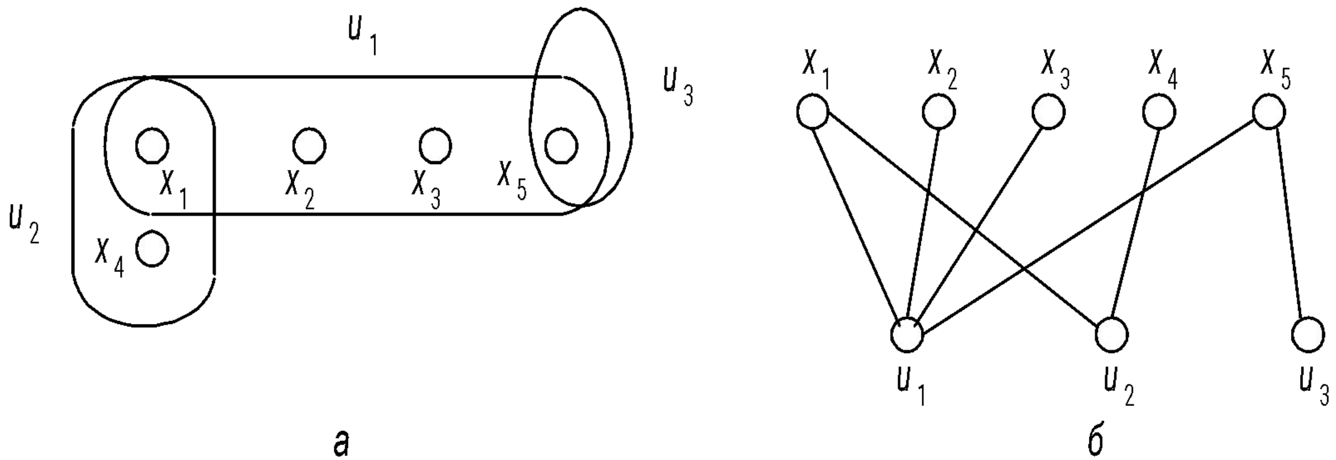


а



б

Представление гиперграфов



В качестве матрицы инцидентности A_H будем использовать матрицу истинности предиката $\Gamma_1(X,U)$.

$$A_H = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{array} \begin{array}{|ccc|} u_1 & u_2 & u_3 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{array} .$$

Аналитическое в форме $H(X,U,\Gamma_1X,\Gamma_2U)$:

$X=\{x_1,x_2,x_3,x_4,x_5\}$, $U=\{u_1,u_2,u_3\}$,

$\Gamma_1X=\{\Gamma_1x_i \,/\, i=1,5\}$, где: $\Gamma_1x_1=\{u_1,u_2\}$, $\Gamma_1x_2=\Gamma_1x_3=\{u_1\}$, $\Gamma_1x_4=\{u_2\}$, $\Gamma_1x_5=\{u_1,u_3\}$,
$\Gamma_2U=\{\Gamma_2u_j \,/\, j=1,3\}$, где: $\Gamma_2u_1=\{x_1,x_2,x_3,x_5\}$, $\Gamma_2u_2=\{x_1,x_4\}$, $\Gamma_2u_3=\{x_5\}$.

Представление гиперграфа матрицами смежности

Предикат смежности вершин $F_1(X, X)$. Элементы матрицы смежности R_1 гиперграфа по его матрице инцидентности определяются по правилу:

$$r_{1i,k} = \begin{cases} 1 \text{ – если } i \neq k \ \& \ \exists \ a_{i,j} = 1 \ \& a_{j,k} = 1, \\ 1 \text{ – если } i = k \ \& \ \exists \ a_{i,j} = 1 \ \& a_{j,k} = 1 \& f = 1, \\ 0 \text{ – в противном случае,} \end{cases}$$

где $i,k = 1,n$; $n = |X|$, $j = 1,m$; $m = |U|$, $f = \sum_{k=1,n} a_{j,k}$, $a_{i,j}$ и $a_{j,k}$ – элементы матрицы инцидентности A_H гиперграфа.

$$R_1 = \begin{vmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 \end{vmatrix}$$

Условие $i=k$ &$f=1$ означает, что при вершине x_i есть петля.

Представление гиперграфа матрицами смежности

Предикат смежности ребер $F_2(U, U)$. Элементы матрицы смежности R_2 гиперграфа по его матрице инцидентности определяются по правилу:

$$r_{2j,k} = \begin{cases} 1 \text{ – если } \exists\ a_{j,i} = 1\ \&a_{i,k} = 1, \\ 0 \text{ – в противном случае,} \end{cases}$$

где $j,k = 1,m$; $m=|U|$, $i = 1,n$; $n=|X|$, $a_{j,i}$ *и* $a_{i,k}$ – элементы матрицы инцидентности A_H гиперграфа.

$$R_2 = \begin{vmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{vmatrix}.$$

Образы вершин гиперграфа относительно предиката смежности F_1

Для каждой вершины $x_i \in X$ гиперграфа $H(X, U)$ *ее образ* $F_1 x_i$ относительно предиката смежности $F_1(X, X)$ (множество смежных ей вершин X_i) определяется характеристическим множеством предиката-свойства $F_1 x_i(X)$

$$X_i = F_1 x_i = \{x_k \in X : F_1 x_i (x_k) = «и»\}.$$

Истинность предиката-свойства $F_1 x_i(X)$ задается i-м вектором-строкой матрицы R_1.

Для рассматриваемого гиперграфа множество образов $F_1 X = \{ F_1 x_i / x_i \in X \}$ его вершин будет:

$F_1 X = \{F_1 x_i / i = 1,5\}$: $F_1 x_1 = \{x_2, x_3, x_4, x_5\}$, $F_1 x_2 = \{x_1, x_3, x_5\}$, $F_1 x_3 = \{x_1, x_2, x_5\}$, $F_1 x_4 = \{x_1\}$, $F_1 x_5 = \{x_1, x_2, x_3, x_5\}$.

Образы ребер гиперграфа относительно предиката смежности F_2

Образ F_2u_j ребра $u_j \in U$ гиперграфа $H(X, U)$ относительно предиката $F_2(U, U)$, т. е. множество U_j смежных ему ребер, определяется характеристическим множеством предиката-свойства $F_2u_j(U)$:

$$U_j = F_2u_j = \{ u_k \in U: F_2u_j(u_k) = «и» \}.$$

Истинность предиката $F_2u_j(U)$ задается j-м вектором-строкой матрицы R_2.

Для нашего гиперграфа :

$F_2U = \{F_2u_j \,/\, j = 1,3\}$: $F_2u_1 = \{u_2, u_3\}$, $F_2u_2 = \{u_1\}$, $F_2u_3 = \{u_1, u_3\}$.

Так же как для ультраграфа, совокупность матриц смежности, а также образов множеств вершин X и ребер U гиперграфа $H(X,U)$ относительно предикатов смежности вершин $F_1(X,X)$ и ребер $F_2(U,U)$ задает гиперграф не полностью.

2.6 Обыкновенные ориентированные графы

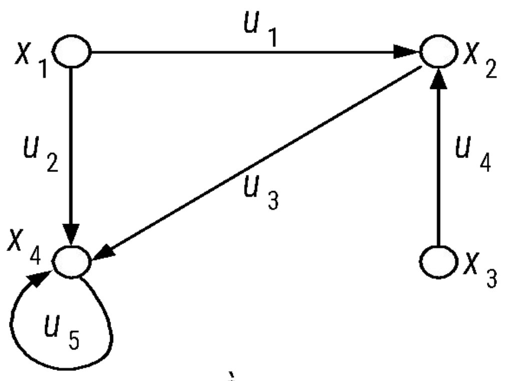
Этот вид графа получим в том случае, если предикаты $Г_1(X,U)$ и $Г_2(U,X)$ таковы, что

$$\forall\ u_j \in U\ \ (|Г_1 u_j| = |Г_2 u_j| = 1)\ , \qquad (5)$$

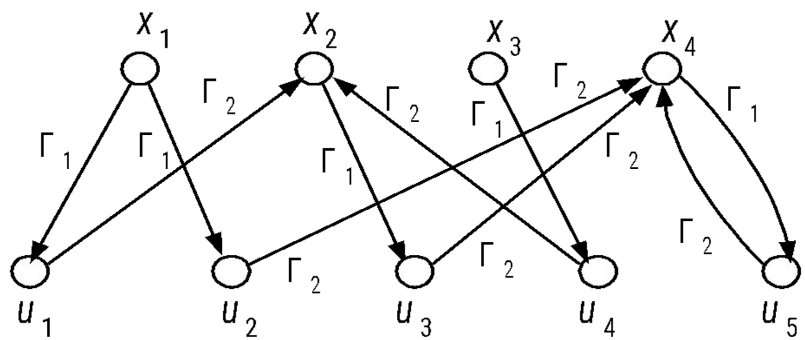
т. е. *в графе нет ребер, суммарное количество вершин, которым оно инцидентно и которые инцидентны ему, больше двух.* Данное условие допускает возможность существования в ориентированном графе петель. Из анализа (2) и (5) видно, что обыкновенный ориентированный граф является *частным случаем ультраграфа.*

Ребра ориентированного графа $G(X,U)$ обычно в литературе называют дугами и изображают стрелками, соединяющими соответствующие пары вершин. С учетом (1) примем такое же изображение, имея в виду, что дуга u_j идет из вершины x_i в вершину x_k, если $Г_1(x_i,u_j)$ = «и» & $Г_2(u_j,x_k)$ = «и», и соединяет вершину x_k с вершиной x_i, если $Г_1(x_k,u_j)$ = «и» & $Г_2(u_j,x_i)$ = «и».

Представление ориентированного графа



à



á

Матрицы инцидентности A_1 и A_2 этого графа определяются так же как и ультраграфа.

Образы и прообразы вершин и ребер относительно предикатов Γ_1 и Γ_2 соответственно:

$X=\{x_1, x_2, x_3, x_4\}$, $U=\{u_1, u_2, u_3, u_4, u_5\}$,

$\Gamma_1 X = \{\Gamma_1 x_i \,/\, i=1,4\}$, где: $\Gamma_1 x_1 = \{u_1, u_2\}$, $\Gamma_1 x_2 = \{u_3\}$, $\Gamma_1 x_3 = \{u_4\}$, $\Gamma_1 x_4 = \{u_5\}$,

$\Gamma_2 U = \{ \Gamma_2 u_j \,/\, j=1,5\}$, где: $\Gamma_2 u_1 = \Gamma_2 u_4 = \{x_2\}$, $\Gamma_2 u_2 = \{x_4\}$, $\Gamma_2 u_3 = \Gamma_2 u_5 = \{x_4\}$,

$\Gamma_2 X = \{\Gamma_2 x_i \,/\, i=1,4\}$, где: $\Gamma_2 x_1 = \Gamma_2 x_3 = \varnothing$, $\Gamma_2 x_2 = \{u_1, u_4\}$, $\Gamma_2 x_4 = \{u_2, u_3, u_5\}$,

$\Gamma_1 U = \{ \Gamma_1 u_j \,/ j=1,5\}$, где: $\Gamma_1 u_1 = \Gamma_1 u_2 = \{x_1\}$, $\Gamma_1 u_3 = \{x_2\}$, $\Gamma_1 u_4 = \{x_3\}$, $\Gamma_1 u_5 = \{x_4\}$.

Для данного способа представления ориентированный граф будем обозначать $G(X, U, \Gamma_1 X, \Gamma_2 X, \Gamma_1 U, \Gamma_2 U)$.

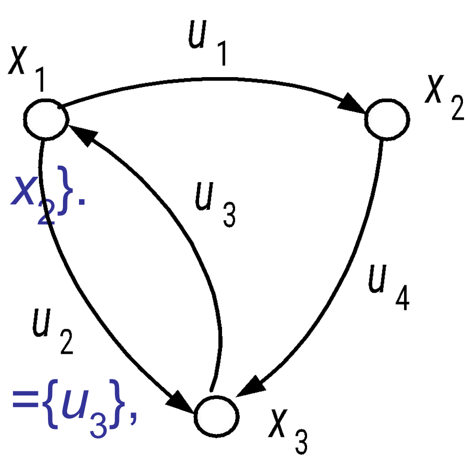
Смежность вершин и ребер ориентированного графа

Для ориентированного графа элементы матриц смежности вершин R_1 и ребер R_2, их образы и прообразы относительно предикатов смежности $F_1(X,X)$ и $F_2(U,U)$ определяются так же как и для ультраграфа.

$X=\{x_1, x_2, x_3\}$, $U=\{u_1, u_2, u_3, u_4\}$,

$F_1X = \{F_1x_i \,/ i=1,3\}$: $F_1x_1 = \{x_2, x_3\}$, $F_1x_2=\{x_3\}$, $F_1x_3 = \{x_1\}$,

$F_1^{-1}X = \{F_1^{-1}x_i \,/ i=1,3\}$: $F_1^{-1}x_1=\{x_3\}$, $F_1^{-1}x_2 =\{x_1\}$, $F_1^{-1}x_3 =\{x_1, x_2\}$.



$F_2U = \{F_2u_j \,/ j=1,4\}$: $F_2u_1 = \{u_4\}$, $F_2u_2 = \{u_3\}$, $F_2u_3 = \{u_1,u_2\}$, $F_2u_4 =\{u_3\}$,

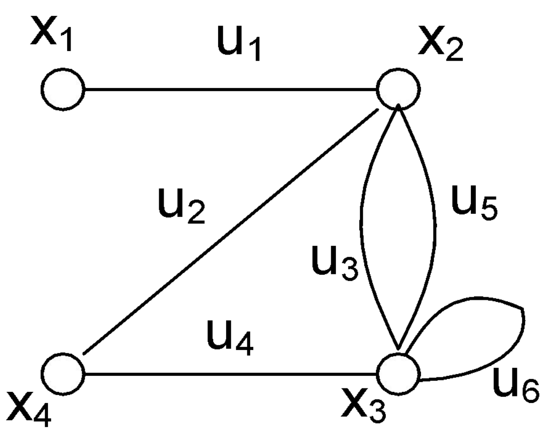
$F_2^{-1}U = \{F_2^{-1}u_j \,/ j=1,4\}$: $F_2^{-1}u_1= F_2^{-1}u_2 =\{u_3\}$, $F_2^{-1}u_3 =\{u_2, u_4\}$, $F_2^{-1}u_4 =\{u_1\}$.

1.2.6. Обыкновенные неориентированные графы

Неориентированный граф можно определить как два непересекающихся множества вершин X и ребер U, на элементах которых задана пара двуместных предикатов-отношений инцидентности $Г_1(X,U)$ и $Г_2(U,X)$, *удовлетворяющих условиям* (1), *(*3, *a*) *и* $\forall\ u_j \in U\ (|Г_1 u_j| = |Г_2 u_j| = 1)$. Таким образом ребра обыкновенного неориентированного графа соединяют вершины попарно, а предикат $Г_2(U,X)$ на основании (3, *a*) является обратным к предикату $Г_1(X,U)$. *Из сказанного выше следует, что неориентированный граф является частным случаем гиперграфа.*

Обыкновенный неориентированный граф будет задан, если заданы множества X, U и один из этих предикатов. Ребра неориентированного графа изображают линиями, соединяющими вершины.

Представление неориентированного графа



В качестве матрицы инцидентности A_H обычно используют матрицу истинности предиката $\Gamma_1(X,U)$.

$$A_H = \begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \end{vmatrix}$$

Образы вершин и ребер относительно предикатов Γ_1 и Γ_2:

$X=\{x_1, x_2, x_3, x_4\}$, $U=\{u_1, u_2, u_3, u_4, u_5, u_6\}$,

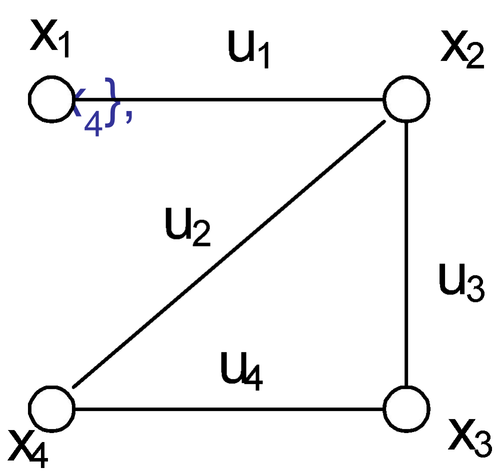
$\Gamma_1 X = \{\Gamma_1 x_i / i=1,4\}$, где: $\Gamma_1 x_1 = \{u_1\}$, $\Gamma_1 x_2 = \{u_1,u_2,u_3,u_5\}$,
$\Gamma_1 x_3 = \{u_1,u_4,u_5,u_6\}$, $\Gamma_1 x_4 = \{u_2,u_4\}$,

$\Gamma_2 U = \{\Gamma_2 u_j / j=1,6\}$, где: $\Gamma_2 u_1 = \{x_1, x_2\}$, $\Gamma_2 u_2 = \{x_2, x_4\}$, $\Gamma_2 u_3 = \{x_2, x_3\}$,
$\Gamma_2 u_4 = \{x_3, x_4\}$, $\Gamma_2 u_5 = \{x_2, x_3\}$, $\Gamma_2 u_6 = \{x_3\}$.

Аналитическое представление неориентированного графа, $G(X, U, \Gamma_1 X, \Gamma_2 U)$, так же как и матричное, является полным.

Смежность вершин и ребер неориентированного графа

Для неориентированного графа элементы матриц смежности вершин R_1 и ребер R_2 и их образы относительно предикатов смежности $F_1(X,X)$ и $F_2(U,U)$ определяются так же как и для гиперграфа.



$X=\{x_1, x_2, x_3, x_4\}$, $U=\{u_1, u_2, u_3, u_4\}$,

$F_1X = \{F_1x_i \,/ i =1,4\}$: $F_1x_1 = \{x_2\}$,

$F_1x_2=\{x_1, x_3, x_4\}$, $F_1x_3=\{x_2, x_4\}$,

$F_1x_4 = \{x_2, x_3\}$,

$F_2U = \{F_2u_j \,/\, j =1,4\}$: $F_2u_1 = \{u_2, u_3\}$,

$F_2u_2 = \{u_1,u_3, u_4\}$,

$F_2u_3 = \{u_1,u_2, u_4\}$,

$F_2u_4 =\{u_2, u_3\}$,

2.8 Характеристики вершин и ребер графов

Характеристики вершин ультра- и ориентированного графа :

- $\rho^+(x_i)$ – *полустепень исхода,* т. е. количество ребер, инцидентных вершине $x_i \in X$: $\rho^+(x_i) = |\ \Gamma_1 x_i\ |$;
- $\rho^-(x_i)$ – *полустепень захода*, т. е. количество ребер, которым инцидентна вершина $x_i \in X$: $\rho^-(x_i) = |\ \Gamma_2 x_i\ |$;
- $s^+(x_i)$ – количество вершин, смежных вершине x_i : $s^+(x_i) = |\ F_1 x_i\ |$;
- $s^-(x_i)$ – количество вершин, которым смежна вершина x_i : $s^-(x_i) = |\ F_1^{-1} x_i\ |$.
- $e(x_i) = |\{\ u_j \in U : \Gamma_1 u_j = \Gamma_2 u_j = x_i\ \}|$ – количество петель при вершине x_i.

У ориентированного графа без кратных ребер $s^+(x_i) = \rho^+(x_i)$ и $s^-(x_i) = \rho^-(x_i)$.

Характеристики вершин и ребер графов

Характеристики ребер ультра- и ориентированного графа :

- $A^{+}(u_j)$ – количество вершин, инцидентных ребру $u_j \in U$: $A^{+}(u_j)= | \Gamma_2 u_j |$;
- $A^{-}(u_j)$ – количество вершин, которым инцидентно ребро $u_j \in U$: $A^{-}(u_j)=| \Gamma_1 u_j |$;
- $s^{+}(u_j)$ – количество ребер, смежных ребру u_i :
  $s^{+}(u_j) =| F_2 u_j |$;
- $s^{-}(u_j)$– количество ребер, которым смежно ребро u_i :
  $s^{-}(u_j) =| F_2^{-1} u_j |$;

Характеристики вершин и ребер графов

Характеристики вершин гипер- и неориентированного графа :

- $\rho(x_i)$ – *локальная степень вершины,* т. е. количество ребер*,* инцидентных вершине $x_i \in X$. $\rho(x_i) = |\, \Gamma_1 x_i \,|$ *;*
- $s(x_i)$ – количество вершин, смежных вершине $x_i \in X$: $s(x_i) = |F_1 x_i\,|$;
- $e(x_i)$ – количество петель при вершине : $e(x_i) = |\{u_j \in \Gamma_2 x_i : |\, \Gamma_2 u_j \,| = 1\}|$.

Характеристики ребер гипер- и неориентированного графа:

- $A(u_j)$– количество вершин множества X, инцидентных ребру u_j : $\rho(u_j) = |\, \Gamma_2\, u_j \,|$ (только для гиперграфа);
- $s(u_j)$ – количество ребер, смежных ребру $u_j \in U$: $s(u_j) = |F_2 u_j\,|$.

2.9 Некоторые особенные графы

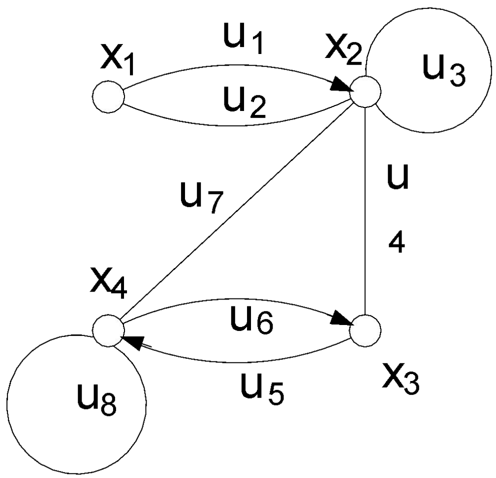
Конечный, пустой и смешанный графы.

Граф $G(X,U)$ называется *конечным*, если число его вершин $|X|=n$ конечно.

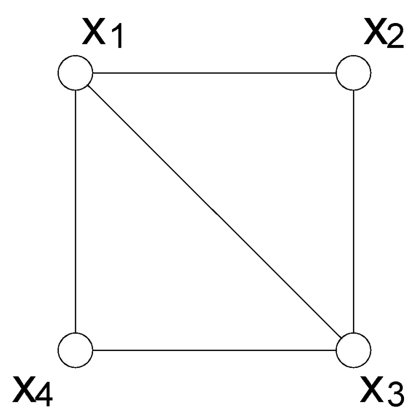
Граф $G(X,U)$, у которого $X = \varnothing$ и $U = \varnothing$, т. е. не содержащий ни вершин, ни ребер, называется *пустым* и обозначается $G_{\varnothing}$.

Граф, на вершинах и ребрах которого заданы две пары двуместных предикатов-отношений инцидентности таких, что для первой пары справедливы (1) и (5), а для второй – (1), (3) и (5), называется *смешанным*.

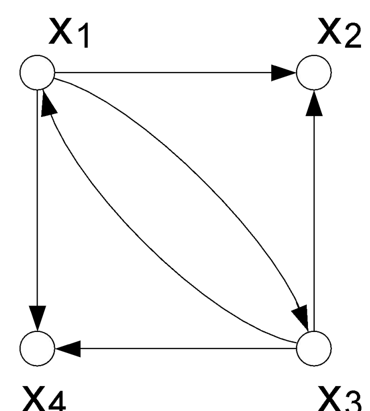
Смешанный



Неориентированный



Ориентированный



Полный и однородный неориентированный графы

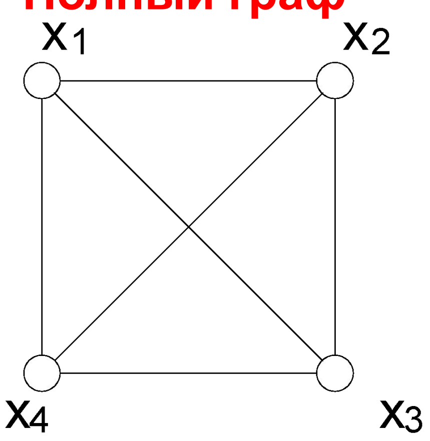
Количество ребер неориентированного графа определяется через локальные степени вершин как:

$$m = \frac{1}{2} \sum_{i=1}^{n} \rho(x_i), \text{ где } |X| = n.$$

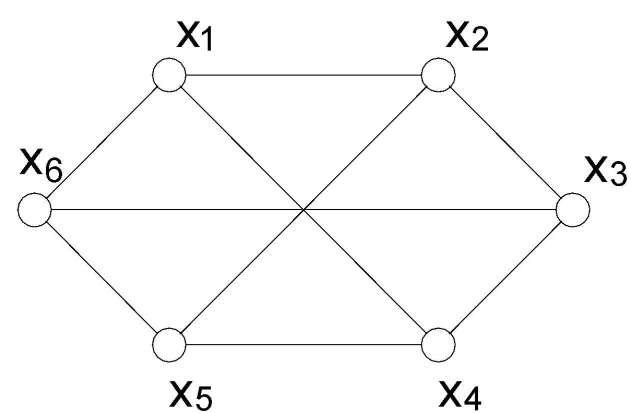
Неориентированный граф называется *полным*, если каждая из его вершин соединена ребрами с остальными, т. е. $(\forall x_i \in X)\ (F_1 x_i = X \setminus x_i)$. У полного графа $\rho(x_i) = n-1$, откуда число его ребер $m = n(n-1)/2$.

Граф называется *однородным* степени K, если $(\forall x_i \in X) \rho(x_i) = K$.

Полный граф



Однородный граф K=3



Полный ориентированный граф

Количество ребер ориентированного графа

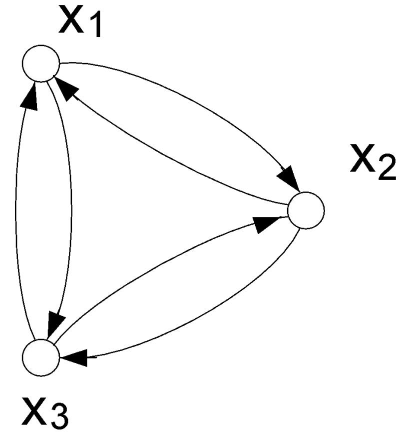
$$r = \sum_{i=1}^{n} \rho^{+}(x_i) \text{ или } r = \sum_{i=1}^{n} \rho^{-}(x_i).$$

Ориентированный граф называется *полным*, если:

$(\forall x_i, x_j \in X)\,(\exists u_k, u_t \in U)(\Gamma_1 u_k = \Gamma_2 u_t = \{x_i\} \,\&\, \Gamma_2 u_k = \Gamma_1 u_t = \{x_j\} \,\&\, i \neq j \,\&\, k \neq t)$, т. е. у каждой пары вершин существует по две дуги, по одной в каждом направлении.

Число ребер полного ориентированного графа – $m = n(n-1)$.

Полный ориентированный граф



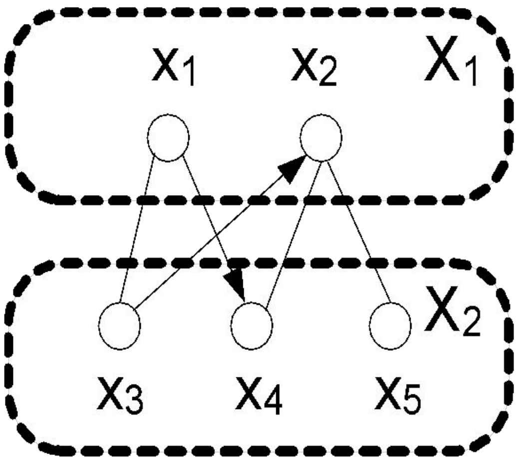
Двудольный граф (граф Кенига)

Граф называется *двудольным* или *графом Кенига*, если его множество вершин X распадается на два непересекающихся подмножества X_1 и X_2, таких, что существуют отношения смежности между элементами этих множеств и не существует таких отношений между элементами каждого множества, т. е.:

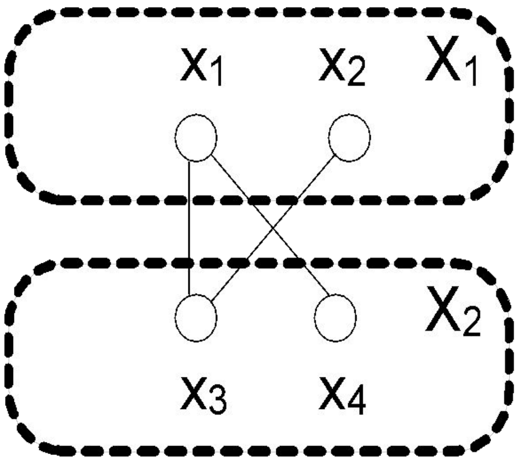
$(\forall x_i, x_j \in X_1)\, x_j \notin F_1 x_i$, $(\forall x_k, x_t \in X_2)\, x_t \notin F_1 x_k$ - для неориентированных и

$(\forall x_i, x_j \in X_1)\, x_j \notin F_1 x_i \,\&\, x_j \notin F_1^{-1} x_i$, $(\forall x_k, x_t \in X_2)\, x_t \notin F_1 x_k \,\&\, x_t \notin F_1^{-1} x_k$ – для ориентированных.

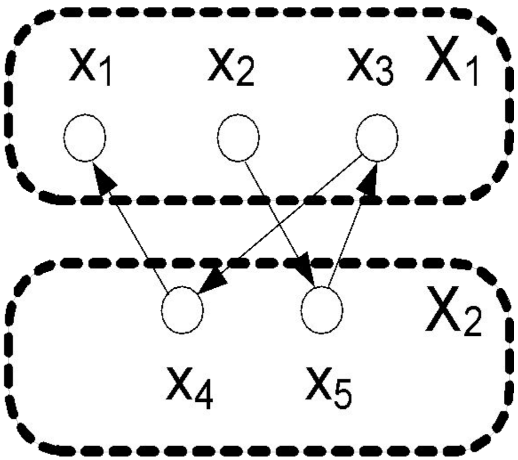
Смешанный



Неориентированный



Ориентированный

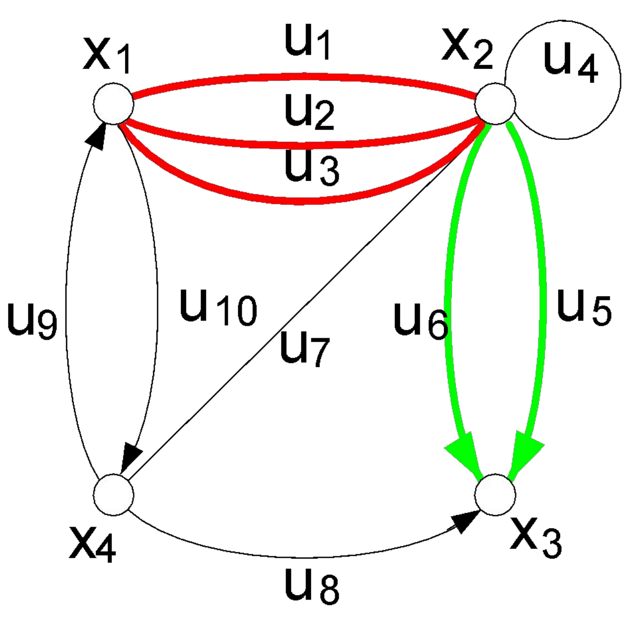


Мультиграф

Граф, у которого хотя бы для *двух ребер* $u_j, u_l \in U$ справедливо

$$\Gamma_2 u_j = \Gamma_2 u_l \,\&\, \Gamma_1 u_j = \Gamma_1 u_l,$$

называется *мультиграфом*, а максимальное количество кратных ребер – *мультичислом*.



Мультичисло:

неориентированных ребер $r = 3$,

ориентированных – $r = 2$

Маршрут, цепь, цикл

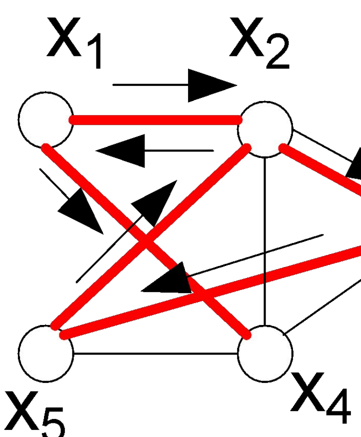
Последовательность смежных ребер неориентированного графа без петель и кратных ребер называется *маршрутом*. В общем случае ребра в маршруте могут встречаться неоднократно.

Если все ребра маршрута различны, он называется *цепью*.

Замкнутая цепь называется *циклом*.

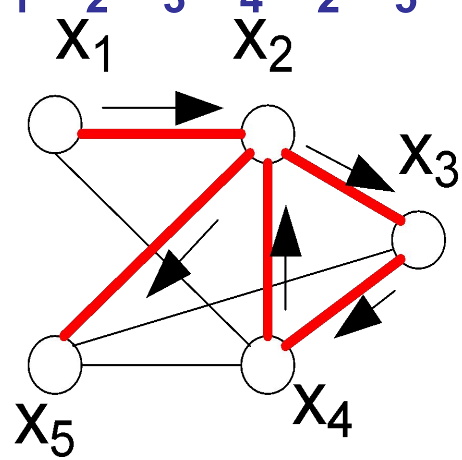
Маршрут

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



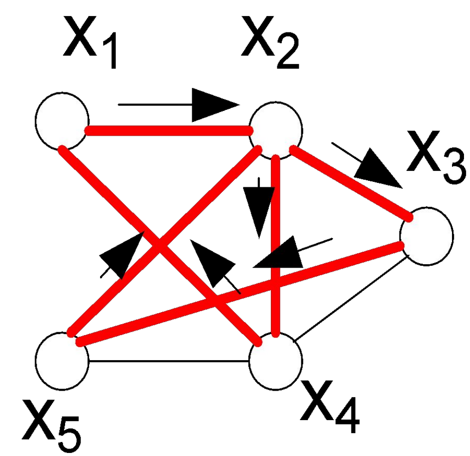
Цепь

x_1,x_2,x_3,x_4,x_2,x_5



Цикл

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



Цепи и циклы будут *простыми*, если они не содержат повторяющихся вершин, например, x_1,x_2,x_3,x_4,x_5 – простая цепь,

а x_1,x_2,x_3,x_4,x_1 – простой цикл.

Эйлеров и гамильтонов циклы

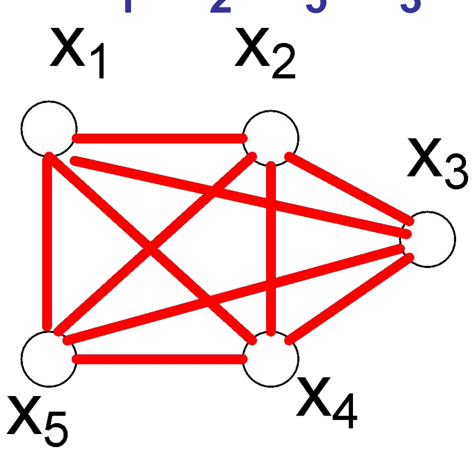
Цикл, включающий все ребра графа, называется *эйлеровым*. Связный граф содержит эйлеров цикл, если локальные степени всех его вершин – четные, т. е.

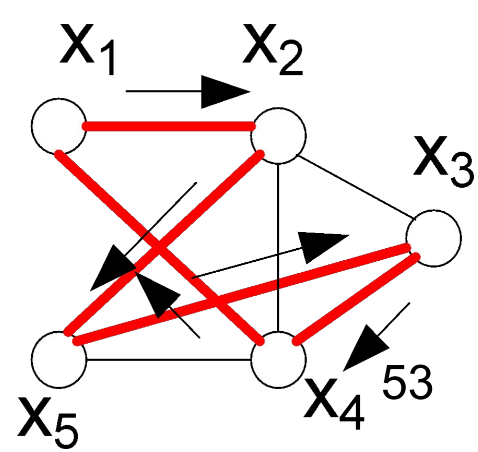
$$(\forall x_i \in X)\ \sum \rho\,(x_i) \equiv 0 \ . \\ \textbf{\textit{mod}}\ \mathbf{2}$$

Простой цикл, проходящий через все вершины графа, называется *гамильтоновым*. Это понятие используется при определении планарности графа. Граф G имеет гамильтонов цикл, если сумма локальных степеней любой пары вершин больше или равна числу его вершин, т. е.

$$\forall x_i,\ x_j \in X\ (\rho(x_i) + \rho(x_j)) \geq n,\ i \neq j,\ |X| = n,$$

например, $\boldsymbol{x_1, x_2, x_5, x_3, x_4, x_1}$.





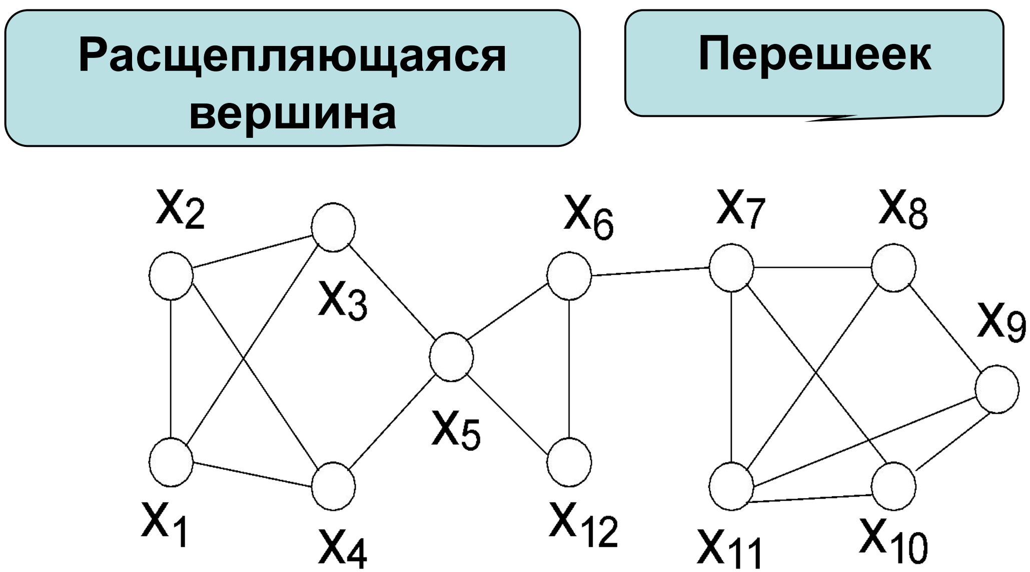
Связность графа

Две вершины x_i, $x_j \in X$ называются *связанными*, если в графе $G(X,U)$ существует маршрут $S = S(x_i,x_j)$.

Граф G – *связный*, если любые две его вершины связаны, т. е. $(\forall x_i, x_j \in X)\ \exists\ S(x_i,x_j)$.

Ребро, удаление которого приводит к разбиению графа на две *компоненты связности*, называется *перешейком*.

Вершина называется *расщепляющейся*, если в ней можно граф разделить на две или более компоненты связности путем ее дублирования.

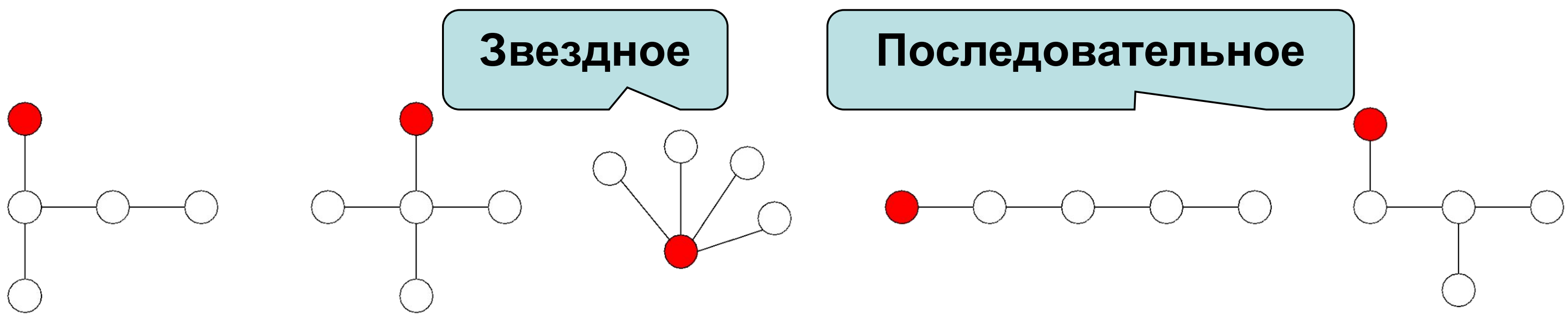


Деревья

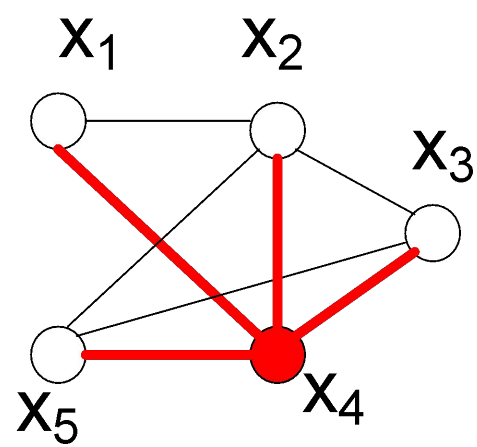
Связный граф, не имеющий циклов, называется *деревом*.

Начальная вершина дерева называется *корнем*, выходящие из него ребра – *ветвями*. Дерево, имеющее n вершин, содержит $m=n-1$ ребер.

На одних и тех же n вершинах можно построить $t_n = n^{n-2}$ различных деревьев.



Дерево называется *покрывающим* граф или *остовным*, если оно содержит все его вершины.



Части графа

Граф $G_i(X_i,U_i)$ называется *частью графа* $G(X,U)$, если он находится в *отношении включения* к нему $G_i \subseteq G$, т. е. $X_i \subseteq X$ и $U_i \subseteq U$.

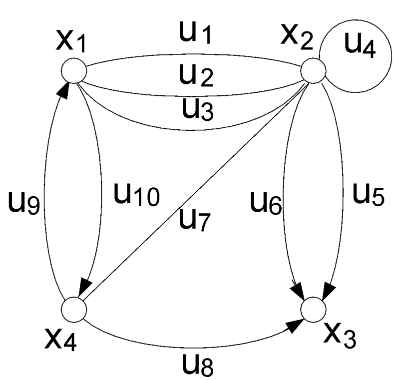
Часть графа $G_i(X_i, U_i)$ называется *куском*, если $X_i \subset X$, $U_i \subseteq U$, причем в U_i входят все ребра, инцидентные X_i.

Множество ребер куска таково, что $U_i = U_{i,i} \cup U_{i,j}$, где $U_{i,i}$ – множество ребер, оба конца которых инцидентны X_i, а $U_{i,j}$ – множество ребер, один конец которых инцидентен X_i, а второй – $X_j = X \setminus X_i$.

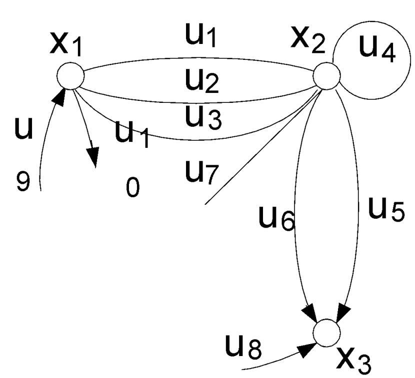
Часть графа $G_i(X_i, U_i)$ называется *подграфом*, если $X_i \subset X$, $U_i \subset U$, т. е. подграф образуется удалением из графа некоторых вершин и всех инцидентных им ребер.

Часть графа $G_i(X_i, U_i)$ называется *суграфом*, если $X_i = X$, а $U_i \subset U$, т. е. суграф получается удалением из графа части ребер.

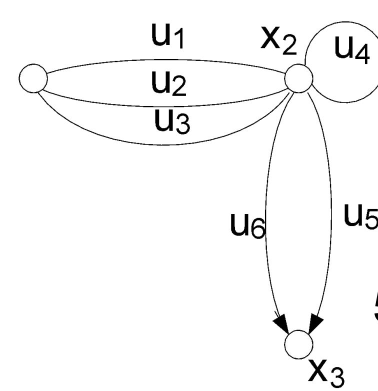
Граф



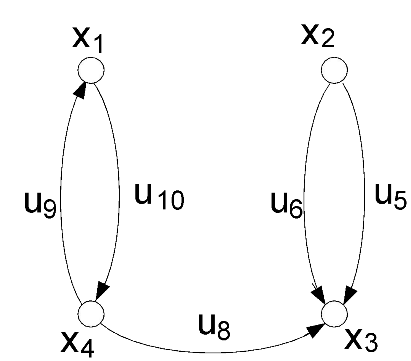
Кусок графа



Подграф



Суграф



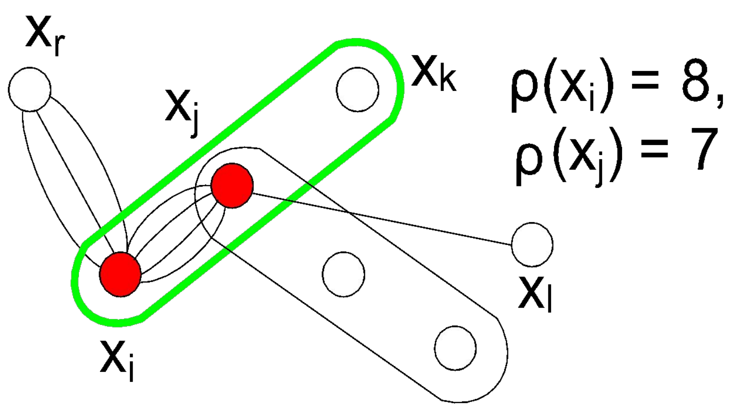
Минимальные массивы

Множество X_i вершин куска $H_i(X_i,U_i)$ гиперграфа $H(X,U)$ называется *минимальным массивом*, если удаление из него любой вершины приводит к увеличению количества внешних ребер,

т. е. для любого куска $H_i'(X_i',U_i')$, в котором $X_i' \subset X_i$, справедливо

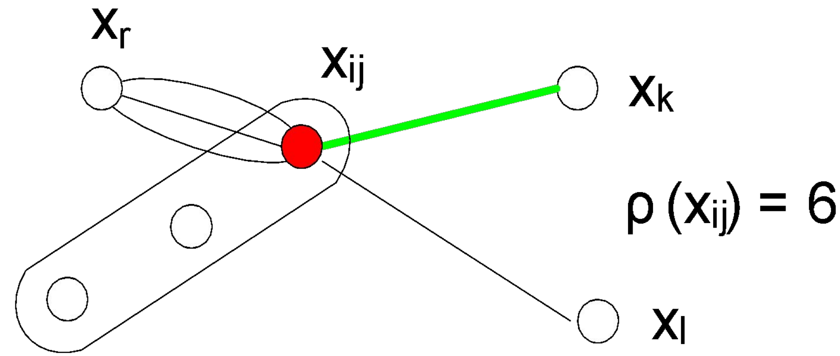
$$\rho(X_i') > \rho(X_i),$$

где $\rho(X_i')$ и $\rho(X_i)$ – количество внешних ребер кусков H_i' и H_i.



Минимальный массив $\{x_i,x_j\}$

Свертка минимального массива



2.10 Представление структур сложных систем графами

Для перехода от объектов задач структурного синтеза к их математическим моделям в виде различного рода графов необходимо:

- сформулировать правила, по которым компоненты объекта будут поставлены в соответствие элементам графа;
- установить вид этих соответствий (взаимно однозначные, однозначные, многозначные) и свойства отношений, определенных на элементах графа (симметричность, рефлексивность, бинарность);

Представление структур сложных систем графами

- задать способ отображения свойств и характеристик компонент объекта в характеристики графа и его элементов.

Все это определяется, исходя из отношений, существующих между компонентами объекта, а также свойств объекта и характеристик его компонент, которые являются необходимыми и достаточными для решения задачи.

2.10.1 Представление схемы ультраграфом

Ультраграф *является универсальной (обобщенной) моделью*, так как позволяет в общем случае отобразить всю информацию, необходимую *для решения широкого класса задач.*.

Модель схемы в виде ультраграфа необходима в тех случаях, когда:

- существенной является информация о принадлежности подсистем соединениям с указанием является подсистема источником сигнала для данной цепи или приемником из нее;
- количество подсистем проектируемого объекта, являющихся источниками/приемниками информации, более одного, т.е. в схеме есть цепи, соединяющие более двух подсистем.

К числу таких задач можно отнести, например, задачи идентификации и покрытия, временного анализа топологической реализации схемы и др.

Представление схемы ультраграфом

Для этих задач *адекватность математической модели объекту следует рассматривать в смысле полноты и правильности отображения той информации, которая характеризует ее функционирование*. В частности для задач покрытия и идентификации – тех свойств схемы, которые определяют логику ее работы.

Тогда в математической модели системы необходимо отобразить следующую информацию о ней:

- имена и функции подсистем, в том числе и монтажной логики;
- имена цепей и, возможно, передаваемых по ним сигналов;
- связанность подсистем как некоторой цепью, так и в системе в целом;
- принадлежность подсистем цепям с точностью до вывода с учетом направления распространения сигнала;
- допустимые значения времени распространения сигналов от подсистем -источников к подсистемам-приемникам.

Представление схемы ультраграфом

При разработке математической модели системы в общем случае будем рассматривать множества подсистем (компонентов) структуры сложной системы П, их типов ТП, контактов К, множество соединений С и имен сигналов *V*. В модели необходимо отобразить подключена ли связь к входу или выходу компонента. Свойства, которые определяют является ли компонент источником сигнала в цепь или наоборот, задаются высказываниями:

– « к выходам подсистем П подключены соединения С» и

– « соединения С подключены к входам подсистем П».

Обозначим эти высказывания как $Пр_1$(П, С) и $Пр_2$(С, П) соответственно.

Представление схемы ультраграфом

Адекватность ультраграфа как структурной модели в указанных выше условиях обеспечивается следующими правилами перехода:

– множеству подсистем структуры П и множеству соединений С поставим во взаимно однозначное соответствие множества вершин ультраграфа X и ребер U;

– типы подсистем отобразим, задав однозначное (сюръективное), возможно взаимно однозначное, соответствие множеств X и ТЭ;

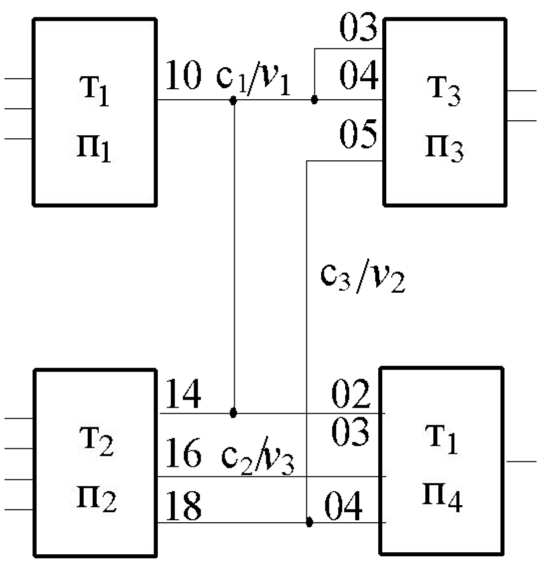
– имена сигналов, передаваемых по соединениям С, отобразим, задав взаимно однозначное, возможно однозначное, соответствие множеств U и V;

– свойства $Пр_1$(П, С) и $Пр_2$(С, П) формально зададим предикатами $Г_1(X, U)$ и $Г_2(U, X)$ соответственно.

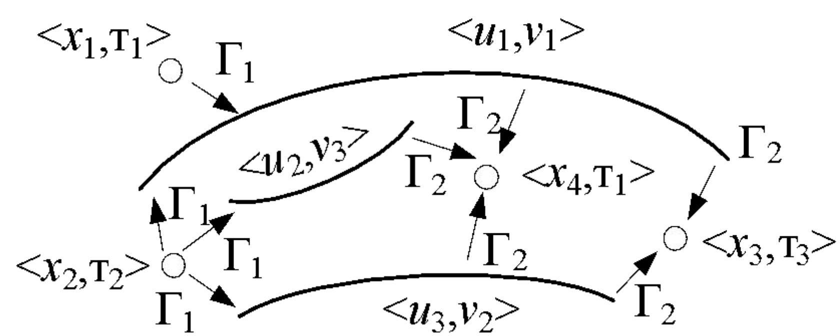
Представление схемы ультраграфом

Формальная запись правил перехода от структуры системы к ее модели в виде ультраграфа $H_U\ (\langle X, T\rangle, \langle U, V\rangle, \Gamma_1, \Gamma_2)$ *имеет вид:*

П ↔ X, С ↔ U, X→ ТЭ (X↔ ТЭ), U ↔ V (U→ V), S_1*(П, С) ~* Γ_1*(X, U),* S_2*(С, П) ~* Γ_2*(U, X).*



а



б

Представление схемы ультраграфом

Информации о номерах выводов подсистемы и времени распространения сигнала от подсистемы-источника до каждого из подсистем-приемников в данной цепи может быть задана присваиванием весов вершинам и/или ребрам ультраграфа.

Множества образов и прообразов рёбер показанного ультраграфа будут:

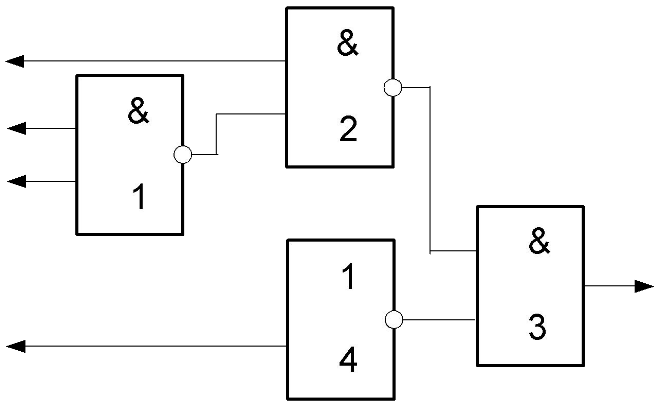
$<Г_2U, К_2>$: $<Г_2u_1, К_2u_1> = \{<x_3, \{03,04\}>, <x_4, 02>\}$, $<Г_2u_2, К_2u_2> = \{<x_4, 03>\}$, $<Г_2u_3, К_2u_3> = \{<x_3, 05>, <x_4, 04>\}$;

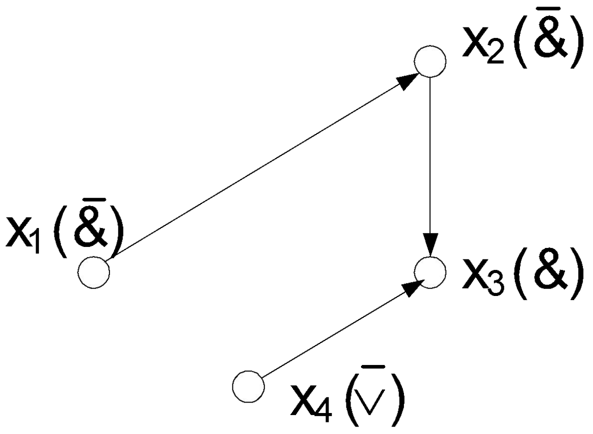
$<Г_1U, К_1>$: $<Г_1u_1, К_1u_1> = \{<x_1, 10>, <x_2, 14>\}$, $<Г_1u_2, К_1u_2> = \{<x_2, 16>\}$, $<Г_1u_3, К_1u_3> = \{<x_2, 18>\}$.

Такой ультраграф будем обозначать $H_U(<X,ТЭ>,< U, V >, Г_1X, Г_2X, <Г_1U, К_1>, <Г_2U, К_2>)$.

2.10.2 Представление схемы ориентированным графом

Обыкновенный ориентированный граф является частным случаем ультраграфа - в нем нет ребер, суммарное количество вершин, которым оно инцидентно и которые инцидентны ему, больше двух. Следовательно, эта модель адекватно отображает схему для *задач структурного анализа и синтеза*, если все цепи схемы соединяют элементы только попарно. *Правила перехода от объекта (схемы соединения элементов) к ориентированному графу такие же, что и для ультраграфа.*





2.10.4 Представление структуры объекта гиперграфом и неориентированным графом

В соответствии с характерными особенностями задач декомпозиции/композиции, размещения, трассировки и др. *математическая модель* системы *должна*:

- задавать принадлежность подсистем соединениям;
- позволять точно оценивать число соединений между подсистемами и частями системы;
- не диктовать порядок соединения подсистем, т. е. отражать фактор неизвестности соединения в пределах одной цепи;
- нести информацию о метрических параметрах и топологических свойствах подсистем и, возможно, соединений.

При этом:

- характер принадлежности связи (вход или выход) не существенен;

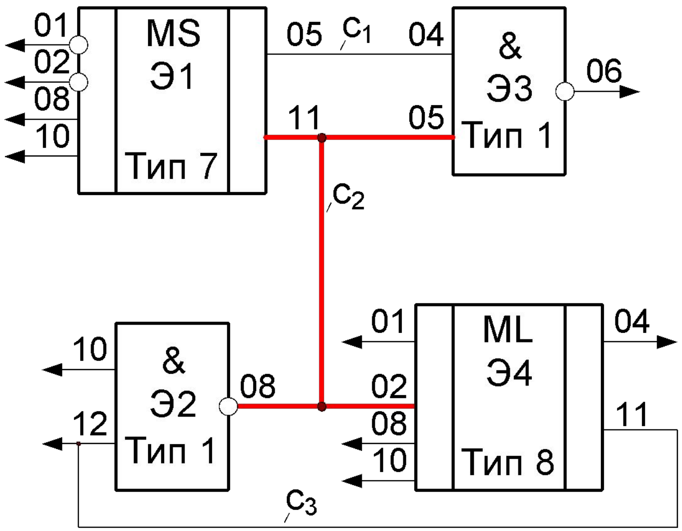
Адекватной моделью системы, если в ней есть цепи, соединяющие более двух подсистем является *гиперграф*.

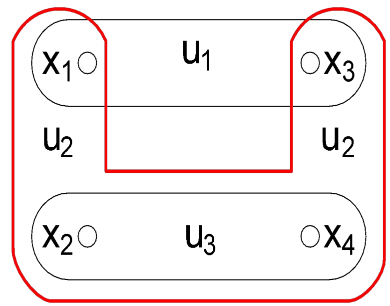
Представление структуры объекта гиперграфом и неориентированным графом

Для решения указанных задач в математической модели системы должна быть отображена следующая информация:

- *имена подсистем, их связанность* с точностью до вывода;
- *принадлежность подсистем цепям*, которые определяются своими именами и, возможно, характеризуются типами;
- *метрические параметры подсистем* (их размеры и размеры полей контактов);
- *координаты подсистем и полей контактов* (после решения задачи размещения);
- *топологические свойства подсистем*, обусловливающие ограничения на построение соединений (порядок расположения выводов, допустимость прохода соединений между ними и под *подсистемой*);
- *возможные варианты топологической реализации* или ориентации *подсистем* и сведения об инвариантности выводов.

Представление схемы гиперграфом





$X = \{x_1, x_2, x_3, x_4\}$,

$U = \{u_1, u_2, u_3\}$,

$Г_1x_1 = Г_1x_3 = U_1 = U_3 = \{u_1, u_2\}$,

$Г_1x_2 = Г_1x_4 = U_2 = U_4 = \{u_2, u_3\}$,

$Г_2u_1 = X_1 = \{x_1, x_3\}$,

$Г_2 u_2 = X_2 = \{x_1, x_2, x_3, x_4\}$,

$Г_2 u_3 = X_3 = \{x_2, x_4\}$.

При переходе от схемы к гиперграфу:

- множеству элементов *Э* поставим во взаимнооднозначное соответствие множество вершин *X* : *Э* ↔ *X*,
- множеству цепей *C* – множество ребер *U*: *C* ↔ *U*,

где $n = |Э|$ и $m = |C|$ – количество элементов и цепей схемы.

Принадлежность цепей элементам и наоборот задается предикатами инцидентности $Г_1(X,U)$ и $Г_2(U,X)$ соответственно:

$Пр_1(Э,С) \sim Г_1(X,U)$, $Пр_2(С,Э) \sim Г_2(U, X)$.

Отношение связанности элементов цепью c_j задается предикатом-свойством $Г_2u_j(X)$. Это - множество вершин $X_j = Г_2u_j$.

Представление схемы гиперграфом с точностью до выводом элементов

Типы элементов, а также имена или *типы цепей* в гиперграфе можно отобразить, задав однозначное соответствие множества X в множества типов элементов ТЭ:

$$X \rightarrow ТЭ$$

и взаимно однозначное отображение множества U в множество типов цепей V:

$$U \leftrightarrow V.$$

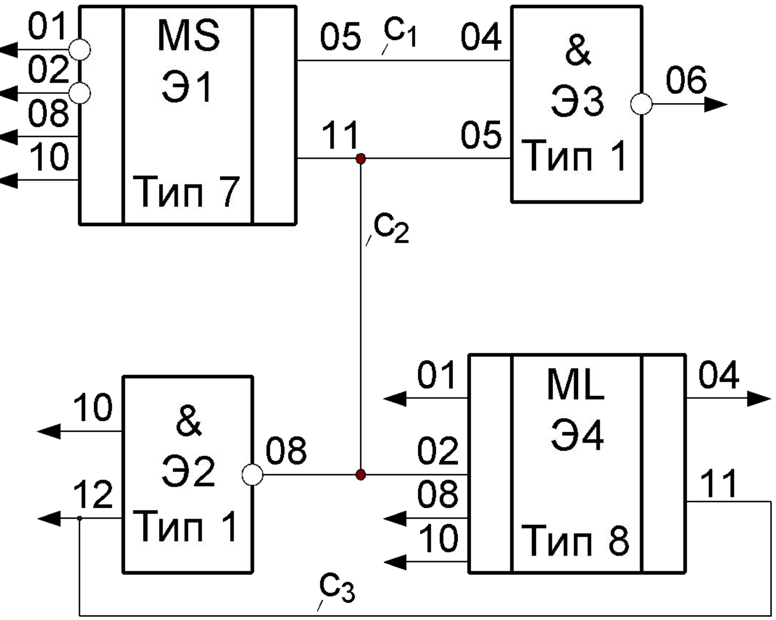
Принадлежность элемента $э_i \in Э$ цепи $c_j \in C$ с точностью до вывода может быть определена следующим образом:

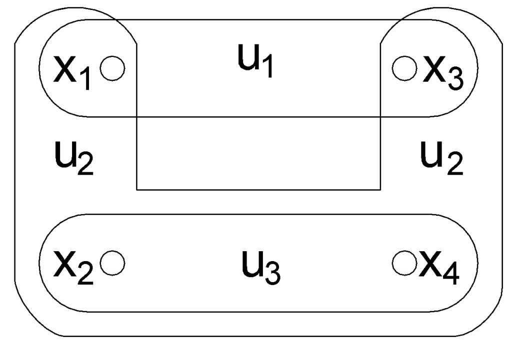
- для отображения $\Gamma_2 U$ каждой вершине $x_i \in \Gamma_2 u_j$ ставится в многозначное соответствие множество контактов элемента $э_i \leftrightarrow x_i$, принадлежащее цепи $c_j \leftrightarrow u_j$.
- для отображения $\Gamma_1 X$ каждому ребру $u_j \in \Gamma_1 x_i$ ставится в многозначное соответствие множество контактов элемента $э_i \leftrightarrow x_i$ входящих в цепь $c_j \leftrightarrow u_j$.

В результате применения указанных правил получаем гиперграф в форме $H(<X,ТЭ>,<U,V>, <\Gamma_1 X, К>, <\Gamma_2 U, К>)$.

Представление схемы взвешенным гиперграфом

Гиперграф в форме $H(<X,ТЭ>, U, Г_1X, <Г_2U, К>)$.





$X = \{< x_1,07 >, < x_2,01 >, < x_3,01 >, < x_4,08 >\}$,

$Г_2 u_1 = X_1 = \{x_1, x_3\}$, $K_1 = \{5,4\}$,

$Г_2 u_2 = X_2 = \{x_1, x_2, x_3, x_4\}$, $K_2 = \{11,8,5,2\}$,

$Г_2 u_3 = X_3 = \{x_2, x_4\}$, $K_3 = \{12,11\}$

или : $Г_2 u_1 = X_{1T} = \{< x_1,05 >, < x_3,04 >\}$,

$Г_2 u_2 = X_{2T} = \{< x_1,11 >, < x_2,08 >, < x_3,05 >, < x_4,02 >\}$,

$Г_2 u_3 = X_{3T} = \{< x_2,12 >, < x_4,11 > \}$.

Определение связности элементов по гиперграфу

Для того, чтобы определить, соединены ли элементы $э_i$ и $э_k$ с цепью c_j, достаточно проверить условие $x_i, x_k \in X_j$. Так как один элемент схемы может принадлежать разным цепям, то в общем случае

$$X_t \cap X_j \neq \varnothing, \text{ где } X_t = \Gamma u_t,\ X_j = \Gamma u_j,\ t, j \in M = \overline{1,m}.$$

Решающее правило определения множества $C_{1,2}$ и количества $s_{1,2}$ цепей, соединяющих две подсхемы, множества элементов которых $Э_1 \subset Э$ и $Э_2 \subset Э$ соответственно($Э_1 \cap Э_2 = \varnothing$) будет:

$$U_{1,2} = \Gamma_1(X_1) \cap \Gamma_1(X_2),\ C_{1,2} \leftrightarrow U_{1,2} \text{ и } s_{1,2} = |\,U_{1,2}\,|, \text{ где } X_1 \leftrightarrow Э_1 \text{ и } X_2 \leftrightarrow Э_2.$$

Следовательно, *по гиперграфу можно точно оценить количество электрических соединений между частями или элементами схемы.*

Так как *множество $X_j = \Gamma_2 u_j$ неупорядоченно*, то последовательность записи в нем вершин гиперграфа не диктует порядок соединения соответствующих им элементов схемы - гиперграф отражает *фактор неизвестности* порядка соединения элементов цепью.

Представление схемы неориентированным графом

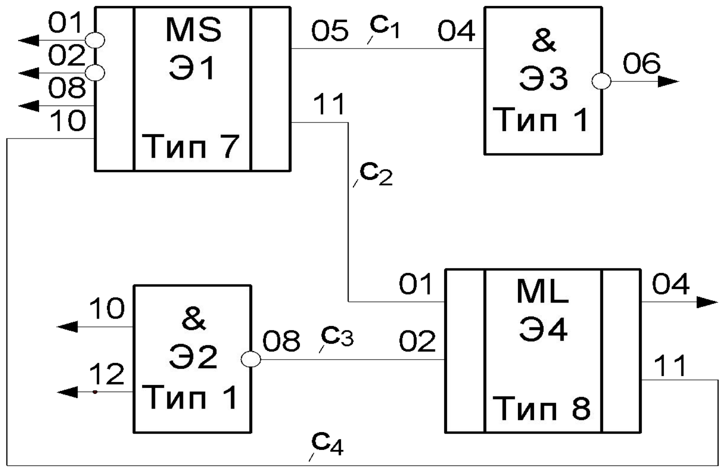
Такая модель, как правило, используется для задач коммутации. В ней необходимо отобразить:

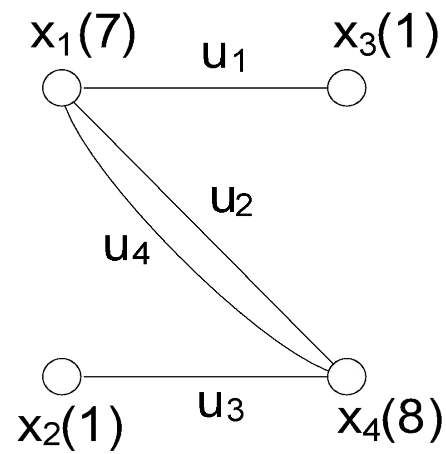
- элементы объекта, их характеристики и координаты;
- возможно, функциональное назначение элементов объекта;
- связи между элементами и их характеристики.

Ранее было указано, что обыкновенный неориентированный граф является частным случаем гиперграфа – в нем нет ребер с суммарным количеством инцидентных вершин большим двух. Следовательно, эта модель может быть корректной только для объектов, элементы которых связаны попарно и для решения задачи несущественно является элемент источником или приемником.

В этом случае она позволяет правильно оценить количество связей между элементами и частями объекта.

Пример представления схемы неориентированным графом





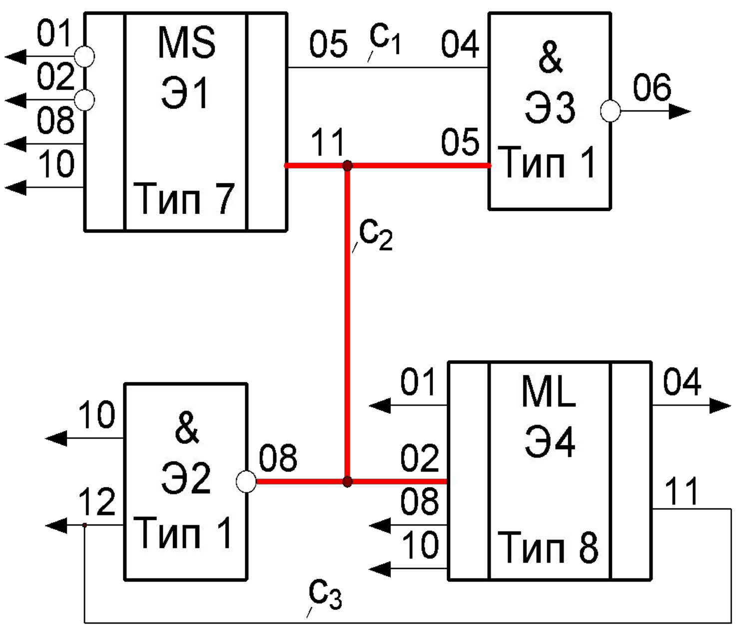
Правила перехода от объекта (схемы соединения элементов) к неориентированному графу такие же, что и для гиперграфа.

Если функциональное назначение связей и/или их характеристики различны, необходимо функции и/или характеристики отразить в графе в виде весов ребер, так как отношение связанности на графе обладает свойством транзитивности. Тогда, чтобы обеспечить получение компоненты связности объекта, соответствующей определенной функции, отношение транзитивности в графе необходимо строить по ребрам, имеющим одинаковый вес.

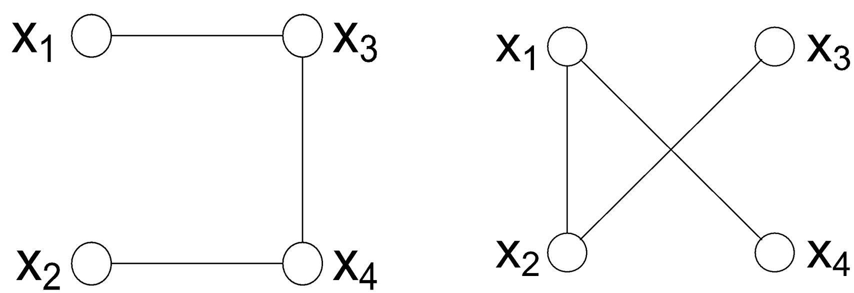
Представление цепей деревом и полным графом

Если соединение связывает *более двух элементов*, то в качестве модели ***цепи*** для размещения и трассировки можно использовать *остовное* дерево. Однако, конкретное дерево отображает один из вариантов порядка соединения выводов элементов.

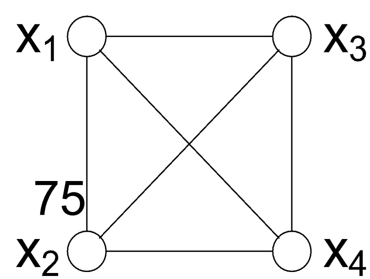
В том случае, когда целесообразно иметь обобщенную модель нескольких возможных вариантов порядка соединения выводов цепью, она может быть получена объединением соответствующих графов, причем для всех возможных вариантов граф будет *полным*.



Варианты представления цепи c_2



Полный граф цепи c_2



2.10.5 Математические модели монтажного пространства

Математические модели монтажного пространства используются для задач размещения и трассировки.

Сущность этих задач – определение положения, которое будут занимать соединения и подсистемы (элементы) в монтажной области объекта.

В математической модели монтажного пространства с учетом метрических параметров, характеристик и топологических свойств объекта, его компонентов и связей между ними, должны быть *формальным образом заданы возможные позиции реализации фрагментов соединений или компонентов объекта*.

Нередко монтажное пространство объектов, в том числе конструктивных модулей средств ЭВТ имеет прямоугольную форму и регулярное расположение позиций. Это в наибольшей степени удовлетворяет требованию конструктивно-технологической унификации.

Позиции установки подсистем (элементов) предыдущего ранга фиксированы и имеют, как правило, постоянный шаг.

При разработке топологии ИС и БИС и проектировании субблока на разногабаритных элементах нельзя заранее зафиксировать позиции для размещения элементов. Монтажное пространство в этом случае является нерегулярным.

Математические модели монтажного пространства

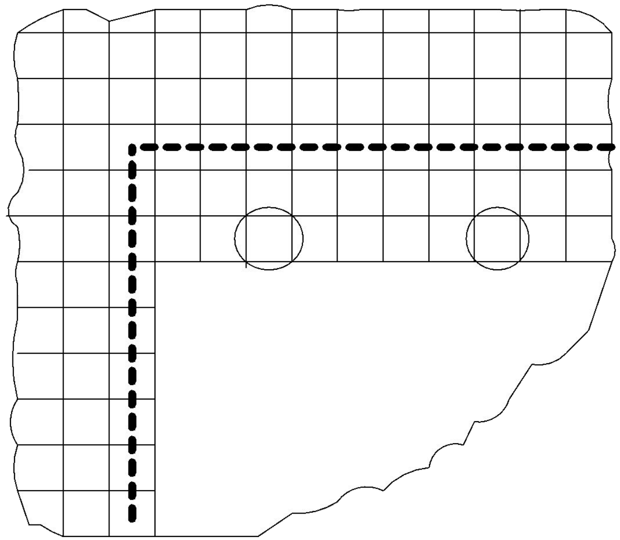
В качестве математической модели монтажного пространства используется неориентированный граф решетки G_r.

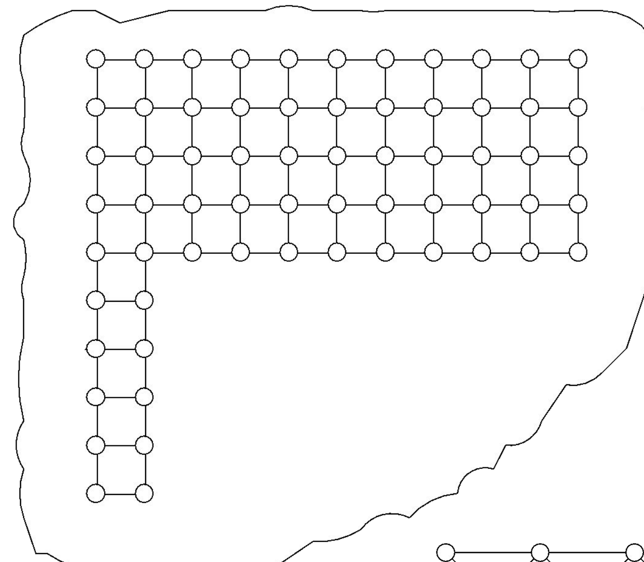
Каждую плоскость монтажного пространства *для задачи трассировки* разбивают на элементарные площадки, стороны которых равны шагу проложения проводника по соответствующему направлению (для печатного монтажа элементарная площадка – квадрат).

Каждой элементарной площадке ставят в соответствие вершину графа решетки.

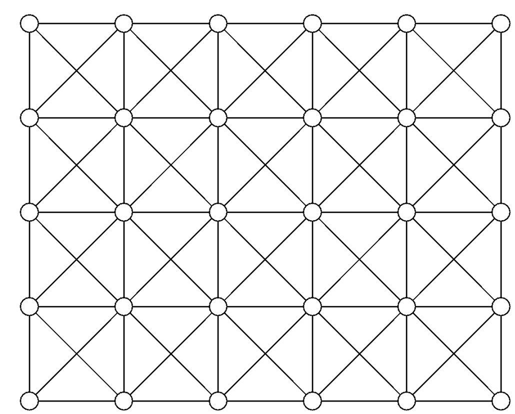
Две вершины соединены ребром, если между соответствующими элементарными площадками может быть проведено соединение с учетом метрических параметров и топологических свойств типовых конструкций, реализуемых в данном монтажном пространстве.

Модель монтажной плоскости фрагмента верхнего слоя печатной платы с ортогональным монтажом при запрещении проведения проводников под микросхемами



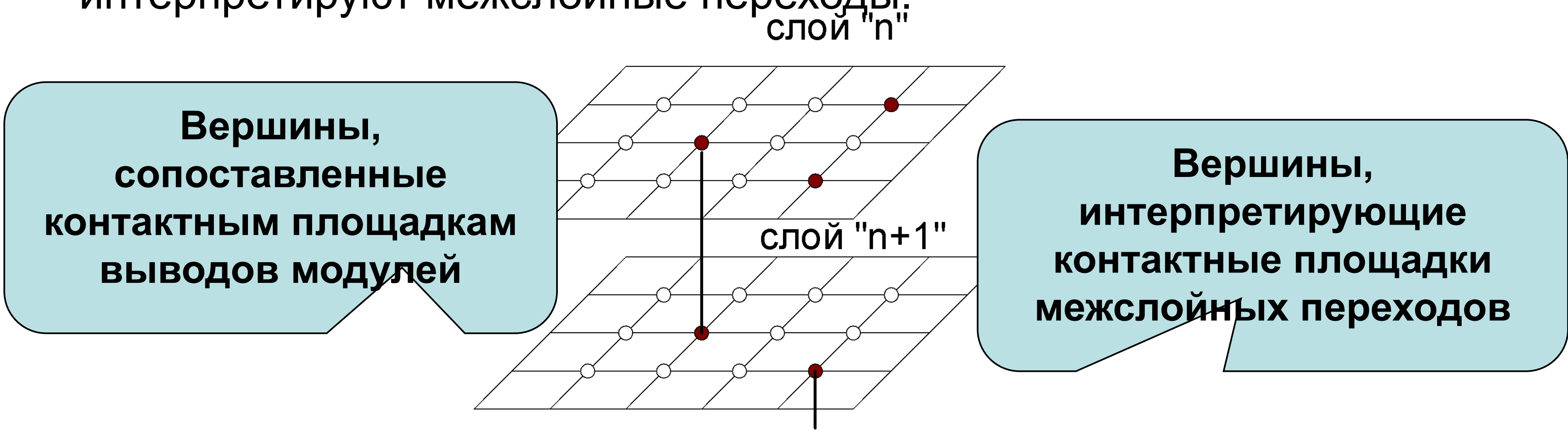


Если проводники разрешается проводить под углом 45°, каждой вершине может быть инцидентно восемь ребер.



Математические модели монтажного пространства (3)

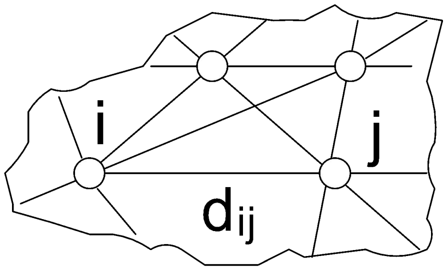
В модели многослойной печатной платы вертикальные ребра интерпретируют межслойные переходы.



В случае выполнения соединений монтажными проводами в любом направлении вершины графа решетки сопоставляют выводам конструктивного элемента (микросхемы, разъема, соединитель-ной платы и т. п.). Варианты различных соединений представляются *полным графом*.

В конкретной реализации соединений необходимо учитывать ограничения на число проводников, подводимых к одному контакту.

Фрагмент полного семивершинного графа



Математические модели монтажного пространства (4)

Расстояние между i-м и *j*-м узлами графа решетки в общем случае будет

$$d_{i,j} = (|S_i - S_j|^k + |t_i - t_j|^k)^h; \qquad i,j = \overline{1,m};$$

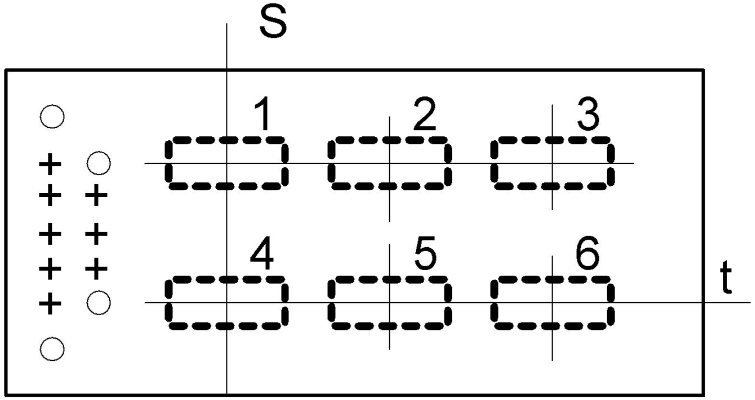
$$k = (1;\ 2);\ h = (0{,}5;\ 1),$$

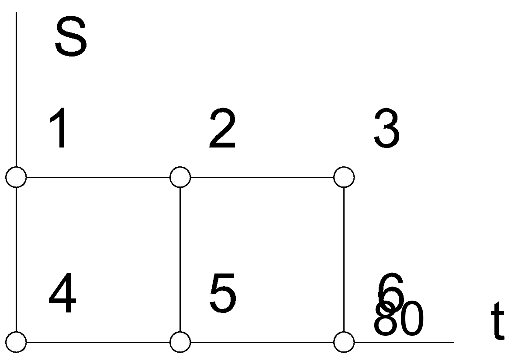
где m — число узлов графа решетки.

При ортогональной трассировке $k = h = 1$, выражение принимает вид

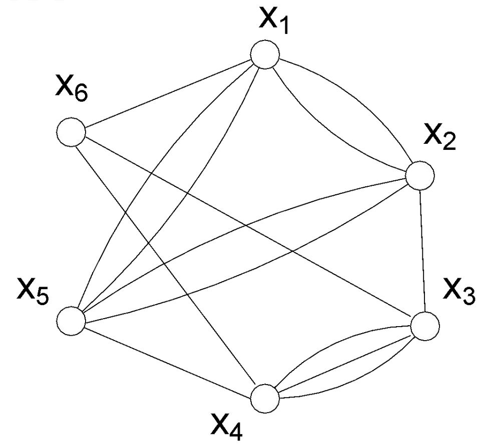
$$d_{i,j} = |S_i - S_j| + |t_i - t_j| .$$

Для регулярного монтажного пространства в качестве модели ***поля размещения*** может быть использован граф решетки, вершины которого сопоставлены установочным позициям типовых конструкций предыдущего уровня.





Приближенный подсчет суммарной длины соединений между модулями



Пусть моделью схемы соединений является неориентированный мультиграф *G* с взвешенной матрицей смежности M_F:

$$M_F = \begin{bmatrix} 0&2&0&0&2&1 \\ 2&0&1&0&2&0 \\ 0&1&0&3&0&1 \\ 0&0&3&0&1&1 \\ 2&2&0&1&0&0 \\ 1&0&1&1&0&0 \end{bmatrix}.$$

Для графа *G*, отображенного в решетку G_r, строится матрица расстояний D_r:

$$D_r = \begin{bmatrix} 0&1&2&1&2&3 \\ 1&0&1&2&1&2 \\ 2&1&0&3&2&1 \\ 1&2&3&0&1&2 \\ 2&1&2&1&0&1 \\ 3&2&1&2&1&0 \end{bmatrix}.$$

Матрица геометрии D_y получается поэлементным умножением матриц M_F и D_r:

$$D_y = D_r \times M_F = \begin{bmatrix} 0&2&0&0&4&3 \\ 2&0&1&0&2&0 \\ 0&1&0&9&0&1 \\ 0&0&9&0&1&2 \\ 4&2&0&1&0&0 \\ 3&0&1&2&0&0 \end{bmatrix}.$$

Откуда суммарная длина ребер $L(G) = 25$.

2.11 Математические модели структур данных

Отображение данных в память ЭВМ требует реализации *содержательных связей между их отдельными частями*, т. е. организации упорядоченного представления информации в виде *структур данных*.

Структуры данных для представления графов должны позволять выполнение формальных преобразований над ними и обеспечивать высокую эффективность этих преобразований и экономное использование памяти.

Представление графа в памяти ЭВМ должно отображать в виде структур данных отношения смежности и инцидентности, заданные на множествах X и U, а также, возможно, дополнительные отношения, позволяющие снизить вычислительную сложность выполнения некоторой или совокупности операций.

Различные структуры данных имеют свои достоинства и недостатки, которые поразному проявляются в алгоритмах решения конкретных задач. *От используемого способа организации множеств и мультимножеств, задающих граф, и связей между ними в значительной степени зависит вычислительная сложность алгоритмов и память, требуемая для хранения данных.*

Базовыми структурами памяти для хранения множеств являются *вектор* и *линейный односвязный список*.

Требования к моделям структур данных

Эти модели должны:

– обеспечивать реализацию операций над данными соответствующими операциями над моделью;

– учитывать *структурные особенности различных способов организации данных*, существенные с точки зрения оценки вычислительной сложности выполняемых операций, а также емкостной сложности структуры данных;

– позволять выполнять формальные преобразования, обеспечивающие синтез новых структур из имеющихся;

– позволять автоматически оценивать вычислительную и емкостную сложность базовой, производных и комбинированных конструкций.

Требования к моделям структур данных

Таким образом в модели необходимо отобразить следующую информацию о структуре данных:

– адреса элементов памяти;

– содержимое элементов памяти, которое может быть как данными об объекте проектирования, так и адресами-указателями элементов рассматриваемой или другой структуры;

– достижимость элементов памяти извне и от других;

– наличие данного (адреса-указателя) в элементе памяти;

– вычислительную сложность операций над данными, обеспечиваемую этой структурой;

– объем, необходимый для хранения содержимого элемента памяти.

Выбор или разработка структур данных

Полный анализ применимости различных структур данных должен включать оценку вычислительной и емкостной сложности алгоритма.

Точная оценка сложности реализации алгоритма для каждого представления графа очень трудоемка, поэтому для выбора структур данных целесообразно использовать оценки трудоемкости выполнения *операций на графе, часто используемых в рассматриваемом алгоритме*.

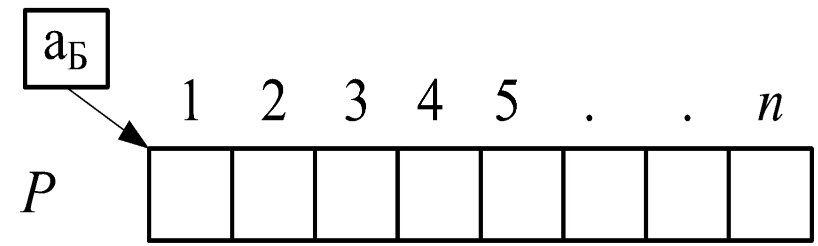
Комбинированные одноуровневые структуры данных позволяют сочетать достоинства и избежать недостатков базовых и производных структур.

Выбор базовых или разработка комбинированных структур данных должны основываться:

- на оценках вычислительной сложности учитываемых операций над структурами данных;
- количестве повторений этих операций, что, как правило, определяется размерностью представляемых множеств.

Вектор

Вектором называют однородный линейный массив фиксированной длины. Это – одномерная последовательность элементов данных одного типа и размера, которая отображается в ***последовательную память.***



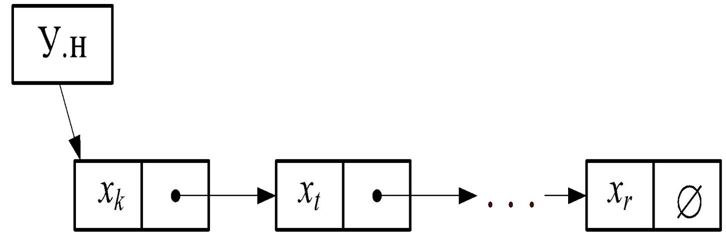
Векторное представление данных имеет следующие преимущества:

- *непосредственный доступ к любому элементу* массива, быстрая и эффективная обработка всех данных или выделенных подмножеств благодаря индексной адресации;
- *минимальная потребность в отслеживании различных имен групп данных* за счет представления этих групп в виде одного массива;
- *эффективное использование памяти ЭВМ*, так как для организации векторной структуры не требуется дополнительной памяти.

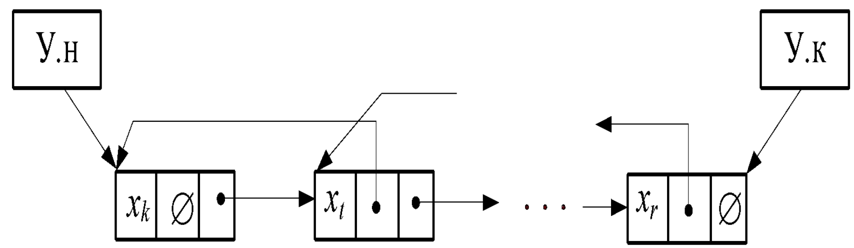
Недостатком векторного представления является *довольно высокая трудоемкость выполнения операций удаления/добавления* элементов векторов (если нельзя заменять последним и добавлять в конец) или некоторой их последовательности,.

Односвязный и двусвязный списки

Линейный односвязный список – это линейный массив перемен-ного размера, каждый элемент которого представляет собой пару, состоящую из информаци-онной части и указателя на следующий.



Двусвязный список. В двусвязном списке существует указатель предыдущего *i*–1-го компонента и указатель конца списка, что обеспечивает дополнительный порядок обработки – от конца списка к его началу. Это устраняет необходимость операций поиска последнего и предыдущего элемента.



Односвязный и двусвязный списки

Для списковых структур характерны следующие преимущества:

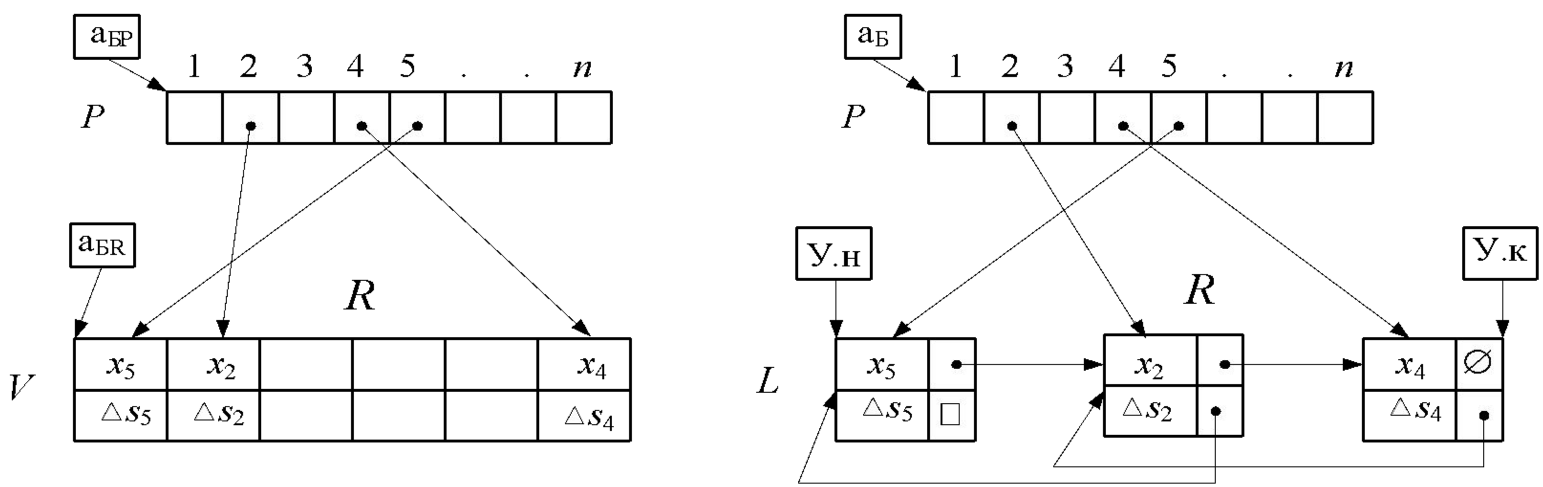
- низкая трудоемкость выполнения операций удаления/добавления данных;
- простота последовательного доступа к элементам данных, содержательные связи между которыми реализованы указателями списка.

Однако списковые структуры не обеспечивают прямого доступа к данным и эффективного использования памяти, так как для указателей требуется дополнительный объем памяти, который может превышать память, хранящую данные.

Комбинированные одноуровневые структуры данных

Такие структуры позволяют эффективно выполнять операции прямого доступа к элементам множества, являющегося подмножеством универсума, и операции удаления/ добавления его элементов, сохраняя порядок их следования .

Прямой доступ обеспечивается вектором *P*, размер которого равен количеству элементов универсума, а i-м элементом является указатель на элемент списка, в котором хранятся x_i и ΔS_i.



а *б*

Модель вектора

Основными компонентами вектора данных как непрерывной последовательности элементов памяти, являются множество адресов элементов – $A_Э$, адрес базы – $a_Б$ и множество значений элементов – $З_Э$. Непрерывное размещение элементов предполагает, что к любому из них можно обратиться непосредственно, определив его смещение относительно адреса базы вектора по номеру и длине элемента. Для элементов вектора определены понятия первый, последний, следующий и предыдущий, т.е. на них задано отношение порядка, при этом адрес каждого элемента может быть получен из адреса любого предыдущего или последующего.

Модель вектора

Возможность непосредственного доступа к элементам памяти определяет свойство достижимости адресов множества $A_Э$ из адреса $a_Б$ – Д($a_Б$, $A_Э$). Такой непосредственный доступ описывается *отношением Ra – «ключ – любой элемент записи множества»*, в котором ключом является адрес базы $a_Б$, а множество координат K – множеством адресов $A_Э$.

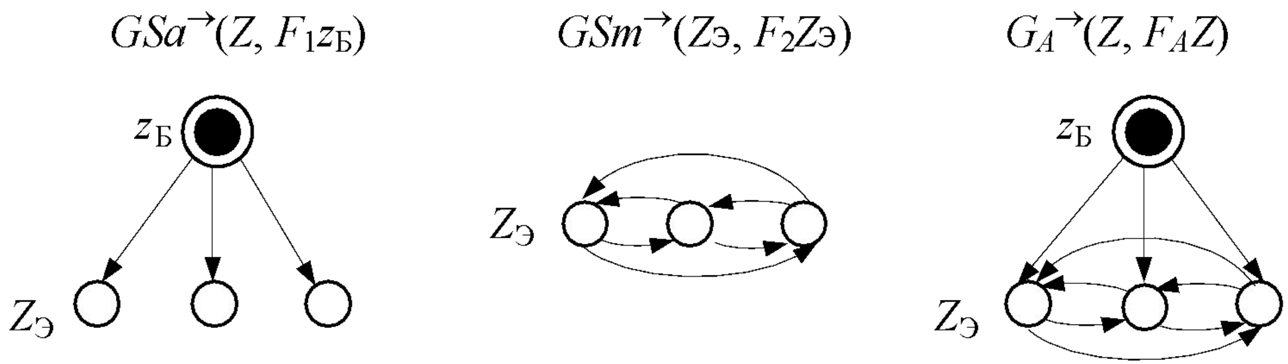
Модель вектора элементов памяти $GSa^{\rightarrow}(Z, Fz_Б)$, отражающую их достижимость от базы, получим по следующим правилам:

$$A \leftrightarrow Z, \text{ где } A = \{a_Б, A_Э\}, \text{ Д}(a_Б, A_Э) \sim F1(z_Б, Z).$$

Здесь $Z = \{z_Б, Z_Э\}$, $Z_Э \leftrightarrow A_Э$, $z_Б \leftrightarrow a_Б$, а $F_1(z_Б, Z)$ – предикат смежности, такой, что $F_1 z_Б = Z_Э$ & $F_1^{-1} z_Б = \varnothing$, $\forall\ z_i \in Z_Э$ $(F_1 z_i = \varnothing)$.

Модель вектора

Модели достижимости адресов вектора: достижимость адресов из адреса базы (*а*), достижимость адреса из любого другого *б*) и их объединение (*в*)



а *б* *в*

Модель вектора

Достижимость элемента памяти от любого другого Д($A_Э$, $A_Э$) реализуется отношением *Rm* «текущий – все предыдущие и следующие». Моделью вектора элементов памяти в данном аспекте будет полный ориентированный граф $GSm^{\rightarrow}(Z_Э, FZ_Э)$, в котором $Z_Э$ – то же, что и выше, Д($A_Э$, $A_Э$) ~ $F_2(Z_Э, Z_Э)$ и $\forall\ z_i \in Z_Э$ ($F_2 z_i = Z_Э \setminus z_i$). Граф $GSm^{\rightarrow}(Z_Э, F_2 Z_Э)$ изображён на рис. *б*. Следовательно, модель достижимости адресов векторной структуры памяти, показанная на рис. *в*, будет определяться как

$$G_A{}^{\rightarrow}(Z_Э, F_A Z), = GSa^{\rightarrow}(Z, F_1 z_Б) \cup GSm^{\rightarrow}(Z_Э, F_2 Z_Э),$$

где $F_A Z = \{F_1 z_Б, F_2 Z_Э\}$.

Модель вектора

Наличие значения $зн_i$ *в элементе памяти с адресом* a_i задает отношение достижимости Д($A_Э$, $З_Э$). Это отношение является вырожденным случаем отношения D «координата элемента в записи множества B соответствует координате ассоциированного с ним элемента в записи множества C». В данном аспекте его можно определить как «координате элемента в записи множества B соответствует элемент записи множества C». Модель вектора элементов памяти, отражающая тот факт, что значение элемента $з_{Эi}$ находится в элементе памяти с адресом a_i, получим по следующим правилам:

$$A_Э \leftrightarrow Z_Э,\ З_Э \leftrightarrow Y \text{ и } Д(A_Э,\ З_Э) \sim F_3(Z_Э,\ Y).$$

Для предиката $F_3(Z_Э, Y)$ справедливо:

$$\forall z_i \in Z_Э\ (F_3 z_i = y_i), \forall y_i \in Y(F_3 y_j = \varnothing) \text{ и } y_i = F_3 z_i \ \&\ y_i = F_3 z_k \rightarrow z_i = z_k.$$

Модель вектора

Тогда моделью достижимости значений данных будет граф $G_З^{\rightarrow}(\{Z_Э, Y\}, F_3 Z_Э)$, в котором $Z_Э \cap Y = \varnothing$ (см. рис *а* на следующем слайде). Этот граф представляет собой двудольный ориентированный граф, состоящий из *n* компонент связности:

$$G_З^{\rightarrow} = \{g_{зi}^{\rightarrow} (\{z_i, y_i\}, F_3 z_i) \;/\; i = 1, n \}, \text{ где } n = |Z_Э|.$$

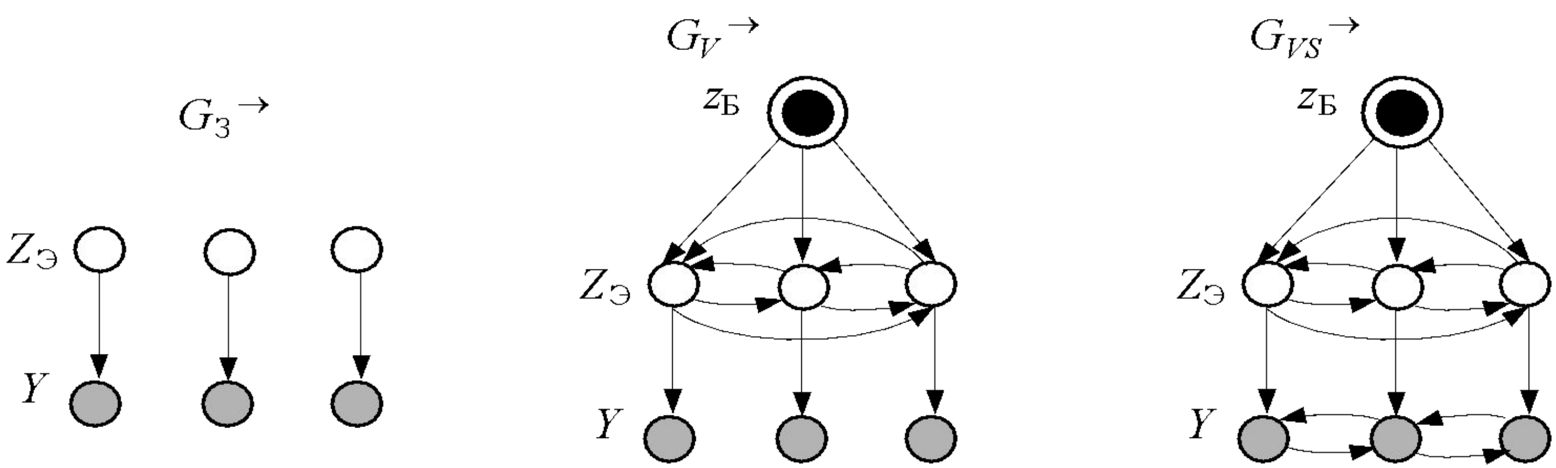
Окончательно моделью вектора данных, отображающую как достижимость элементов памяти, так и принадлежность значения данного этому элементу, будет граф

$$G_V^{\rightarrow} (\{Z, Y\}, FZ) = G_A^{\rightarrow} (Z, F_A Z) \cup G_З^{\rightarrow} (\{Z_э, Y\}, F_3 Z_э),$$

в котором $FZ_Э = \{F_1 z_Б, F_2 Z_Э, F_3 Z_Э \}$. Эта модель показана на рис. *б* следующего слайда.

Модель вектора

Модели достижимости значений данных (*а*), вектора данных (*б*) и вектора упорядоченных данных (*в*)



а *б* *в*

Модель двусвязного списка

Переход от списковой структуры к его модели в виде ориентированного графа проиллюстрируем на примере двусвязного списка. В моделях указателей начала и конца списка необходимо отобразить их адреса $a_{у.н}$ и $a_{у.к}$, адреса первого a_1 и последнего a_m элементов списка и достижимости Д $(a_{у.н}, a_1)$ и Д$(a_{у.к}, a_m)$. Это отношения *Rf* «*ключ – первый элемент*» и *Rl* «*ключ – последний элемент*» записи множества. В списке ключам доступа к элементам записи множества *kf* и *kl* соответствуют адреса указателей начала и конца списка $a_{у.н}$ и $a_{у.к}$.

Модель списка получим по следующим правилам. Поставим во взаимно однозначное соответствие адресам указателей начала и конца списка $A_у = \{a_{у.н}, a_{у.к}\}$ вершины множества $Z_у = \{z_{у.н}, z_{у.к}\}$:

$$a_{у.н} \leftrightarrow z_{у.н},\ a_{у.к} \leftrightarrow z_{у.к},$$

Модель двусвязного списка

адресам элементов списка $A_Э$ – вершины множества $Z_{ЭD}$, значениям элементов данных $З_Э$ – вершины множества Y_D:

$$A_Э \leftrightarrow Z_{ЭD},\ З_Э \leftrightarrow Y_D.$$

Отношения достижимости первого и последнего элементов списка сопоставим предикатам смежности, определённым на вершинах $\{z_{у.н}, z_1\}$ и $\{z_{у.к}, z_m\}$:

$$Д(a_{у.н}, a_1) \sim F(z_{у.н}, z_1) \text{ и } Д(a_{у.к}, a_m\}) \sim F(z_{у.к}, z_m).$$

Тогда моделями указателей начала и конца списка будут соответственно следующие ориентированные графы (см. рис. *а* на следующем слайде):

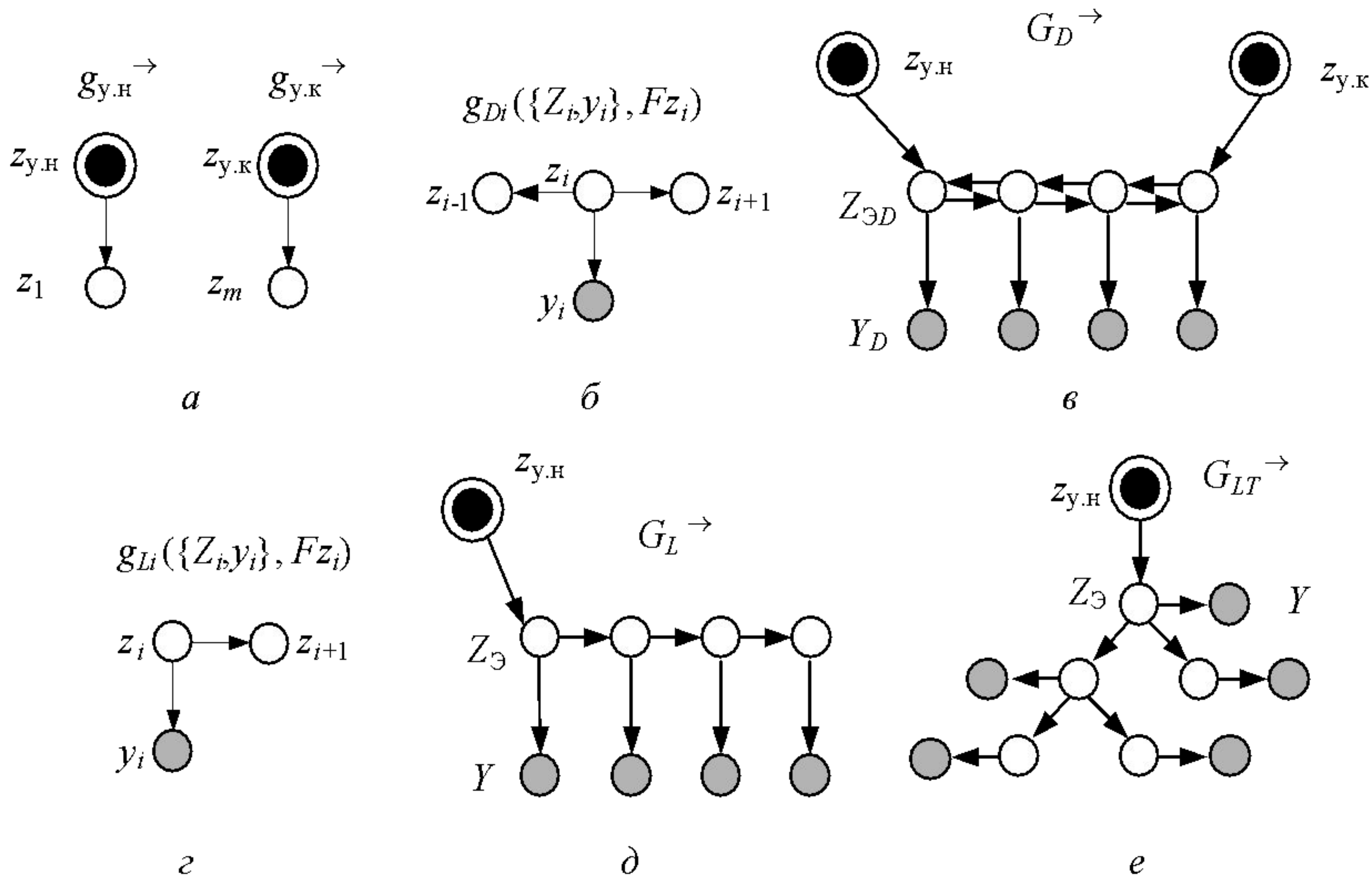
$$g_{у.н}{}^{\rightarrow}(\{z_{у.н}, z_1\}, F\{z_{у.н}, z_1\}),$$

где $Fz_{у.н} = \{z_1\}$, $Fz_1 = \varnothing$, $z_1 \in Z_{ЭD}$ или $g_{у.н}{}^{\rightarrow}(\{z_{у.н}, z_1\}, Fz_{у.н})$ и

$$g_{у.к}{}^{\rightarrow}(\{z_{у.к}, z_m\}, F\{z_{у.к}, z_m\}),$$

где $Fz_{у.к} = \{z_m\}$, $Fz_m = \varnothing$, $z_m \in Z_{ЭD}$ или $g_{у.к}{}^{\rightarrow}(\{z_{у.к}, z_1\}, Fz_{у.к})$.

Модель двусвязного списка



Модель двусвязного списка

В моделях элементов списка необходимо отобразить их адреса $a_i \in A_Э$, рассмотренные выше кортежи и отношение достижимости из текущего элемента следующего и предыдущего (это отношения $R^{\rightarrow}$ и $R^{\leftarrow}$). Правила перехода к моделям элементов списка будут иметь вид:

$a_i \leftrightarrow z_i \in Z_{ЭD}$, $<з_{Э1}, \varnothing, a_2> \leftrightarrow <y_1, \varnothing, z_2>$, $<з_{Эi}, a_{i\text{-}1}, a_{i+1}> \leftrightarrow <y_i, z_{i\text{-}1}, z_{i+1}>$ и $<з_{Эm}, a_{m\text{-}1}, \varnothing> \leftrightarrow <y_m, z_{m\text{-}1}, \varnothing>$ для первого, i-го ($1 < i < m$) и последнего элементов соответственно.

С учётом рассмотренного выше отношения достижимости значения элемента моделью i-го элемента списка будет следующий ориентированный граф(см. рис. *б предыдущего слайда*):

$$g_{Di}^{\rightarrow}(\{Z_i, y_i\}, Fz_i),$$

Модель двусвязного списка

Этот граф является объединением графов $g_i^{\rightarrow}(Z_i, Fz_i)$, $g_i^{\leftarrow}(Z_i, Fz_i)$ и графа $g_{3i}^{\rightarrow}(\{z_i, y_i\}, Fz_i)$, определённого при получении модели вектора данных.

Моделями первого и последнего элементов списка будут соответственно следующие ориентированные графы:

$$g_1^{\rightarrow}(\{Z_1, y_1\}, Fz_1),$$

где $Z_1 = \{z_1, z_2\}$, $Fz_1 = <y_1, \varnothing, z_2>$ ($Fy_1 = Fz_2 = \varnothing$) и

$$g_m^{\rightarrow}(\{Z_m, y_m\}, Fz_m),$$

где $Z_m = \{z_{m\text{-}1}, z_m\}$, $Fz_m = <_{ym}, z_{m\text{-}1}, \varnothing>$ ($Fy_m = Fz_{m\text{-}1} = \varnothing$).

Модель двусвязного списка (см. рис. *в*) получим объединением моделей его элементов и указателей его начала и конца:

$G_D^{\rightarrow}(\{Z_у, Z_{ЭD}, Y_D\}, F\{Z_у, Z_{ЭD}\}) = g_{у.н}^{\rightarrow} \cup g_{у.к}^{\rightarrow} \cup \{g_{Di}^{\rightarrow} \cup \ / \ i=1,m\}$,

где для двусвязного списка $Z_у = \{z_{у.н}, z_{у.к}\}$, $Z_{ЭD} \leftrightarrow A_Э$, $Y_D \leftrightarrow З_Э$.

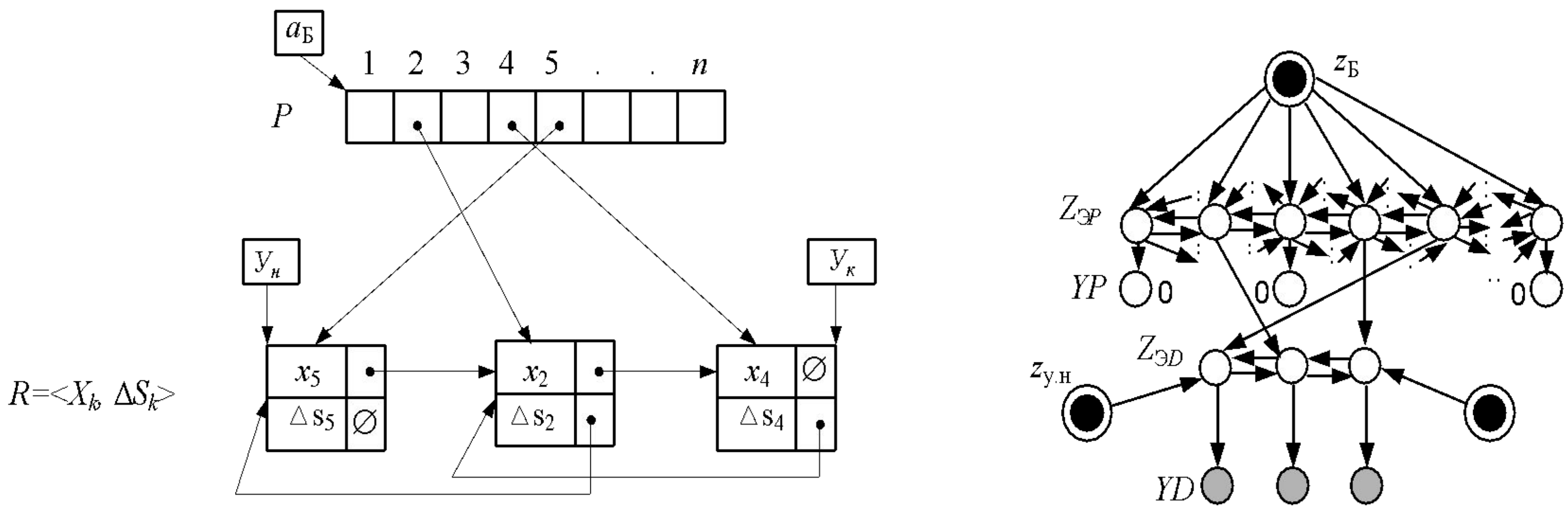
Модель комбинированной одноуровневой структуры

Рассматриваемая структура предназначена для обеспечения вычислительной сложности равной 1 операции добавления/удаления и поиска элемента по номеру при сохранении порядка следования элементов подмножества R. Каждый элемент подмножества R – пара, состоящая из вершины графа $x_j \in X_к \subset X$ и значения локального критерия целесообразности ее выбора Δs_j : $R = \{r_j \,/\, j = 1, |X_к|\}$, $r_j = \langle x_j, \Delta s_j \rangle$. Множество R хранится в двусвязном списке L, обращение к элементам которого осуществляется через вектор указателей (адресов прямого доступа) P. Элемент этого вектора содержит адрес элемента двусвязного списка, если $x_i \in X$ равен $x_j \in X_к$. В противном случае элемент вектора равен нулю.

Модель комбинированной одноуровневой структуры

Моделью этой комбинированной структуры является граф – объединение моделей вектора прямого доступа *P* и двусвязного списка:

$$G_C^{\rightarrow}(\{z_Б, Z_{ЭP},\{z_{у.н}, z_{у.к}\}, Z_{ЭD}, YP, YD\}, FZ_P, FZ_D).$$



а *б*

Модель комбинированной одноуровневой структуры

В этом графе: $z_Б \leftrightarrow a_Б$, $Z_{ЭР} \leftrightarrow A_{ЭР}$, $Z_{ЭР} = \{z_i \,/\, i = 1, n\}$, $A_{ЭР}$ – адреса элементов вектора, $n = |X|$; $z_{у.н} \leftrightarrow a_{у.н}$, $z_{у.к} \leftrightarrow a_{у.к}$, $Z_{ЭD} \leftrightarrow A_{ЭD}$, $Z_{ЭD}= \{zd_j \,/\, j = 1, m\}$, $A_{ЭD}$ – адреса элементов списка, $m = |X_к|$. Для реализации прямого доступа к элементам списка L использовано отношение P «координата элемента в записи множества B соответствует координате ассоциированного с ним элемента в записи множества C».

Элементы остальных множеств определяются по правилам:

$YP = \{yp_i \,/\, i = 1, n\}$, $yp_i = zd_j$, если $x_i = x_j \in r_j$, $zd_j \in Z_{ЭD}$, $x_i \in X$, $x_j \in X_k$ и yp_i =0 в противном случае;

$YD = \{yd_j \,/\, j = 1, m\}$, $yd_j = <x_j, \Delta s_j>$, $x_j \in X_k$, $\Delta s_j \in \Delta S_k$;

$Z_P = \{z_Б, Z_{ЭР}\}$, $F_1 z_Б = Z_{ЭР}$ & $F_1^{-1} z_Б = \varnothing$ и $\forall z_i \in Z_{ЭР}$ $(F_2 zi = Z_{ЭР} \setminus z_i$, $F_3 z_i$ = $=yp_i)$;

Модель комбинированной одноуровневой структуры

$Z_D = \{\{z_{у.н}, z_{у.к}\}, Z_{ЭD}\}$, $Fz_{у.н} = zd_1$, $Fz_{у.к} = zd_m$, $Fzd_1 = < yd_1, \varnothing, zd_2>$, $Fzd_j = <yd_j , zd_{j-1}, zd_{j+1}>$ для $1< j < m$, $Fzd_m =<yd_m , zd_{m-1}, \varnothing>$.

Добавление вектора прямого доступа к двусвязному списку не только позволяет снизить вычислительную сложность операции поиска элемента по номеру до $q_i = 1$ вместо $Q = n$, но и не влияет на сложность выполнения остальных операций доступа.

Рассмотренная структура приводит к увеличению емкостной сложности представления данных, так как вектор адресов прямого доступа должен иметь размер соответствующего универсума (в нашем примере множества X).

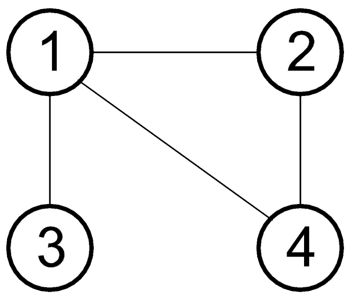
Двухуровневые структуры данных

Представление графов множествами вершин, рёбер и их образов (прообразов) требуют организации двухуровневых структур, так как образы и прообразы вершин (рёбер) – это множества подмножеств, каждое из которых соответствует элементу множества X (U). Таким образом, $\Gamma_1 X$, $\Gamma_2 U$, $F_1 X$, $F_2 U$ и т. д. – должны быть связанными с X и U множествами одноуровневых структур с учетом их иерархической подчиненности (в графе множества вершин X и ребер U – первичны, а образы и прообразы вершин и ребер – вторичны). Связь между элементами множеств вершин (рёбер) и их образами (прообразами) относительно предикатов инцидентности и/или смежности реализуется биективным отношением (предикатом) $R1$ в трактовке «координатам элементов множества в указанной структуре соответствуют базы (указатели) структур данных, хранящих их образы».

Двухуровневые структуры данных

Ниже рассмотрены структуры для представления графа в форме $G(X,F_1X)$.

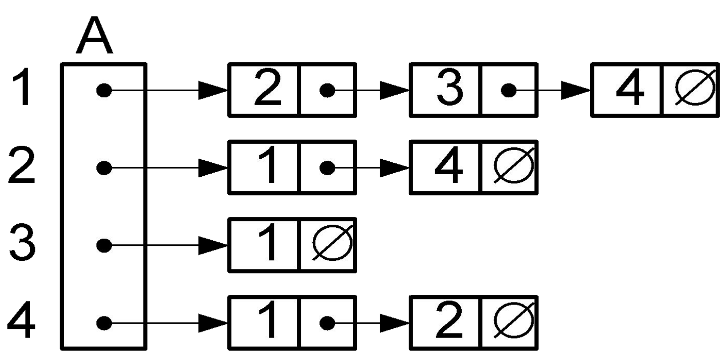
Неориентированный граф



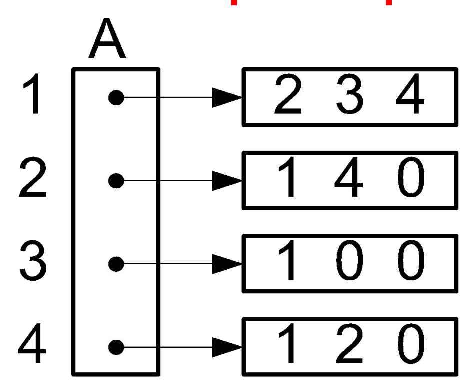
Матрица смежности

	1	2	3	4
1	0	1	1	1
2	1	0	0	1
3	1	0	0	0
4	1	1	0	0

Вектор списков

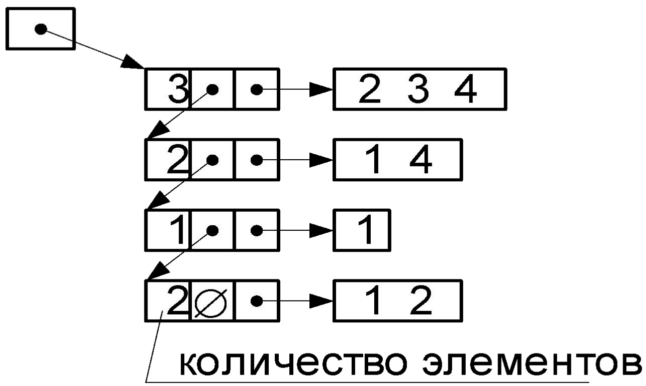


Вектор векторов

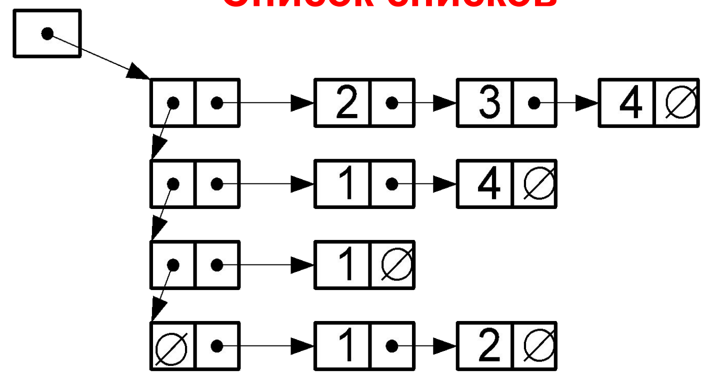


Двухуровневые структуры данных

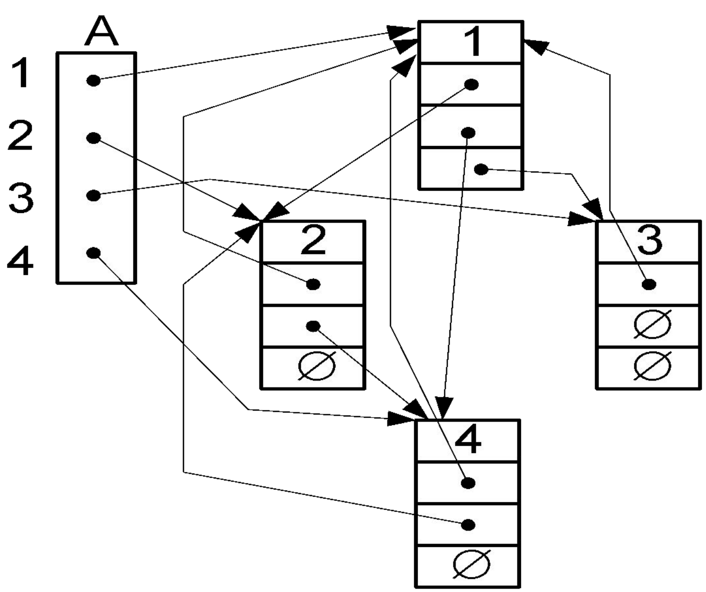
Список векторов



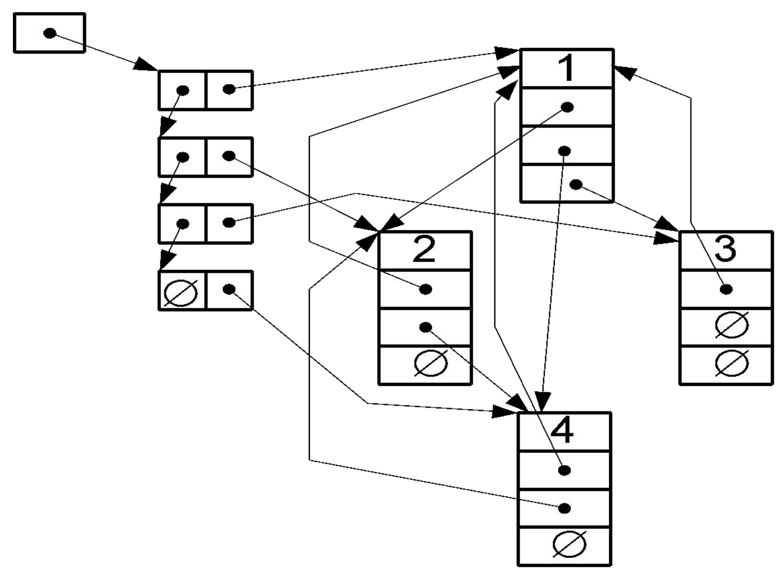
Список списков



Вектор *n*-связных списков

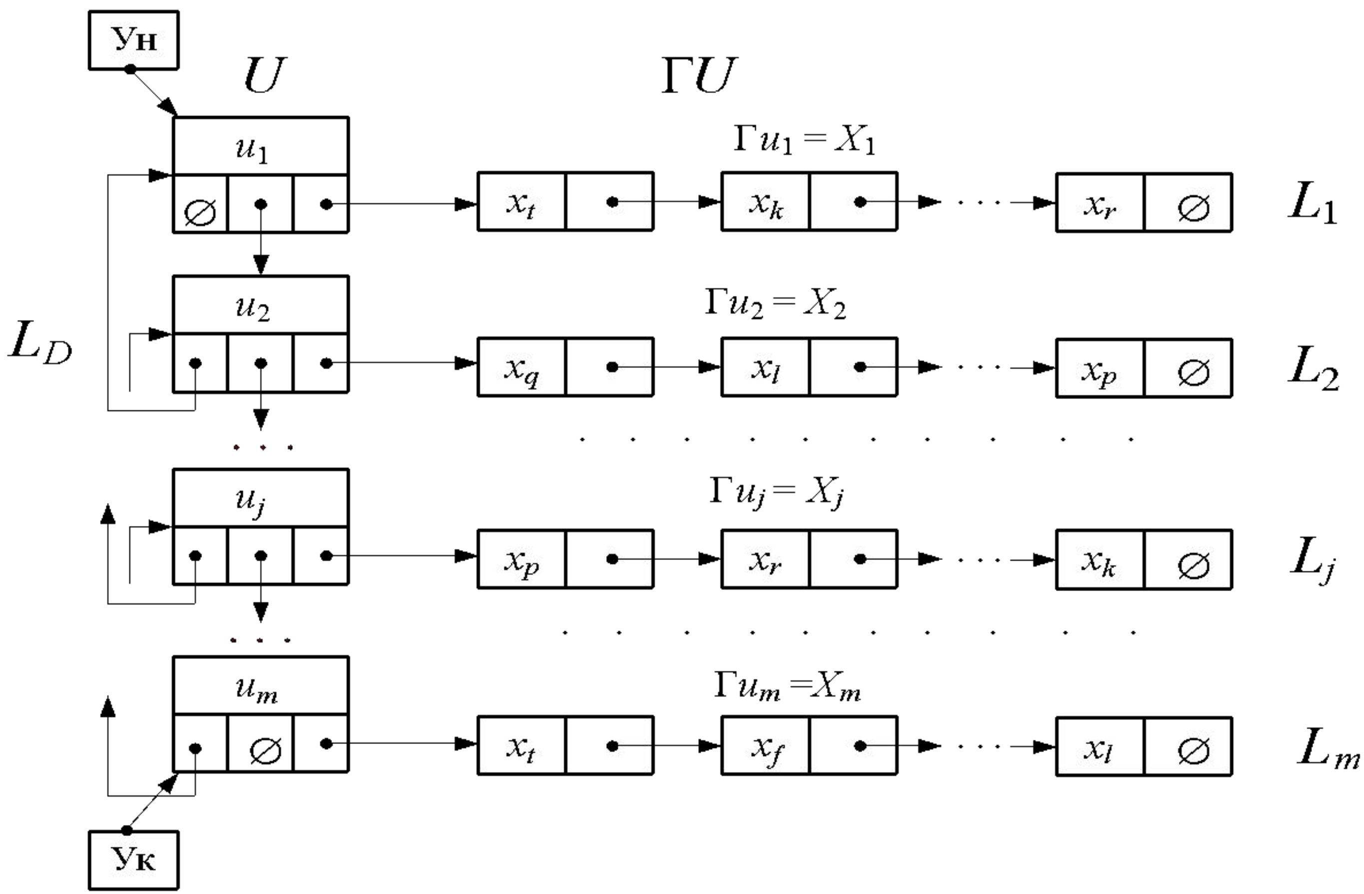


Список *n*-связных списков



Модель двухуровневой структуры данных список-списков

Модель двухуровневой структуры данных рассмотрим для части аналитического представления гиперграфа *H*(*X*, *U*)

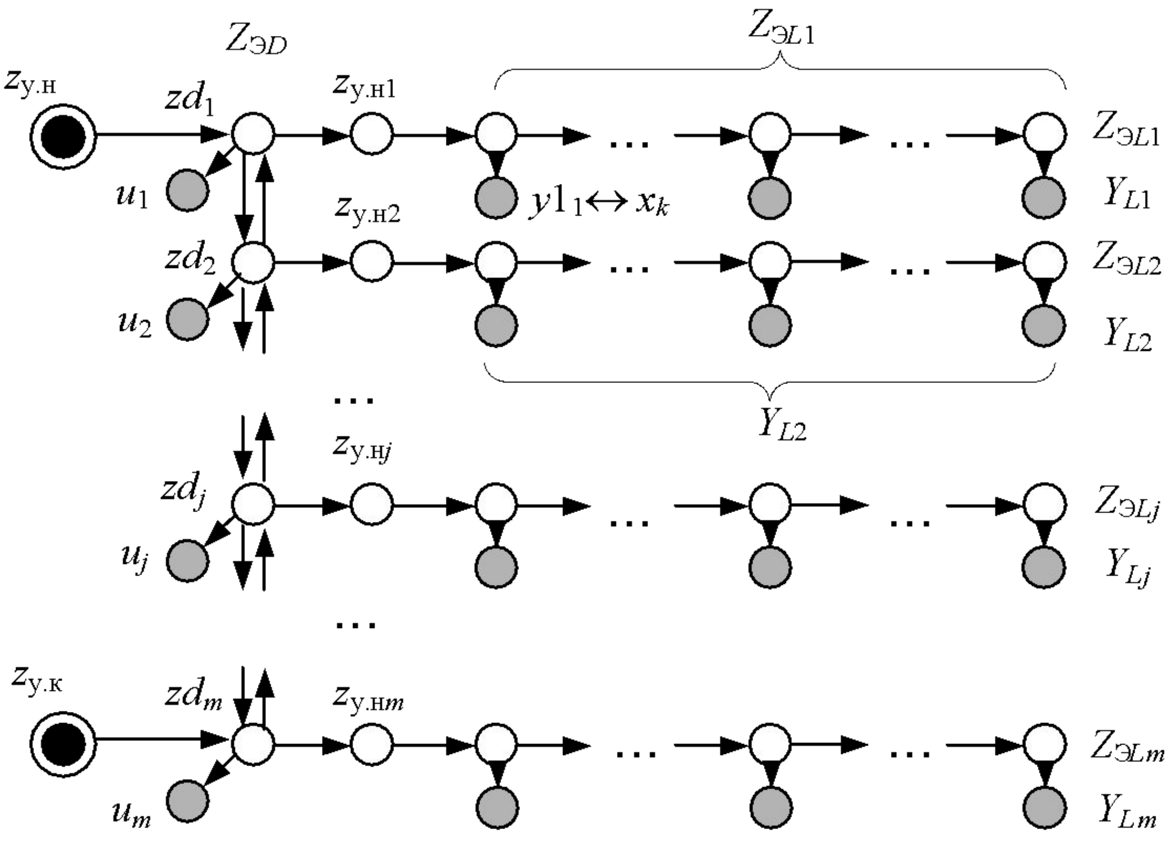


Модель двухуровневой структуры данных список-списков

В данной структуре хранятся два вида данных: множество ребер U и множество образов этих ребер $\Gamma U = \{\Gamma u_j \,/\, j = 1, m\}$, $m = |U|$. Первые организованы в двухсвязный список L_D, а вторые собраны в m односвязных списков L_j, хранящих образ каждого элемента $u_j \in U$. Для реализации связи между ребром u_j и его образом Γu_j в каждом элементе двусвязного списка L_D предусмотрено дополнительное поле, в котором записан указатель на начало списка L_j.

Модель двухуровневой структуры данных список-списков

Моделью этой двухуровневой структуры является граф $G_S(\{Z_у, Z_{ЭD}, ZУ, Z_{ЭL}, Y_D, Y_L\}, FZ_D, FZ_L)$.



Модель двухуровневой структуры данных список-списков

Правила перехода от двусвязного списка к его модели те же, что и выше. Моделью каждого односвязного списка L_j является граф $G_{Lj}^{\rightarrow}(\{z_{у.н j}, Z\text{Э}_{Lj}, Y_{Lj}\}, FZ_{Lj})$, в котором:

– $z_{у.н j} \leftrightarrow a_{у.н j}$, где $a_{у.н j}$ адрес указателя начала списка;

– $Z_{\text{Э}Lj} \leftrightarrow A_{\text{Э}Lj}$, где $A\text{Э}_{Lj} = \{aj_i \,/\, i = 1, |X_j|\}$ адреса элементов списка, $Z_{\text{Э}Lj} = \{zj_i \,/\, i = 1, |X_j|\}$, $zj_i \leftrightarrow aj_i$, $|X_j| = \Gamma u_j$;

– $Y_{Lj} \leftrightarrow \Gamma u_j$, $Y_{Lj} = \{yj_i \,/\, i = 1, |X_j|\}$, $yj_i \leftrightarrow x_l$, $x_l \in X_j$;

–$Z_{Lj} = \{z_{у.н j}, Z_{\text{Э}Lj}\}$;

– $FZ_{Lj} = \{Fz_{у.н j}, FZ_{\text{Э}Lj}\}$, $Fz_{у.н j} =\{zj_1\}$, $FZ_{\text{Э}Lj} = \{Fzj_i \,/\, i = 1, |X_j|\}$, $Fzj_i = <yj_i, zj_{i+1}>$.

Организация двухуровневых структур требует дополнительной информации в представлении множества. В двусвязном списке *LD* - это дополнительное информационное поле в его элементе.

Пример двухуровневой комбинированной структуры данных

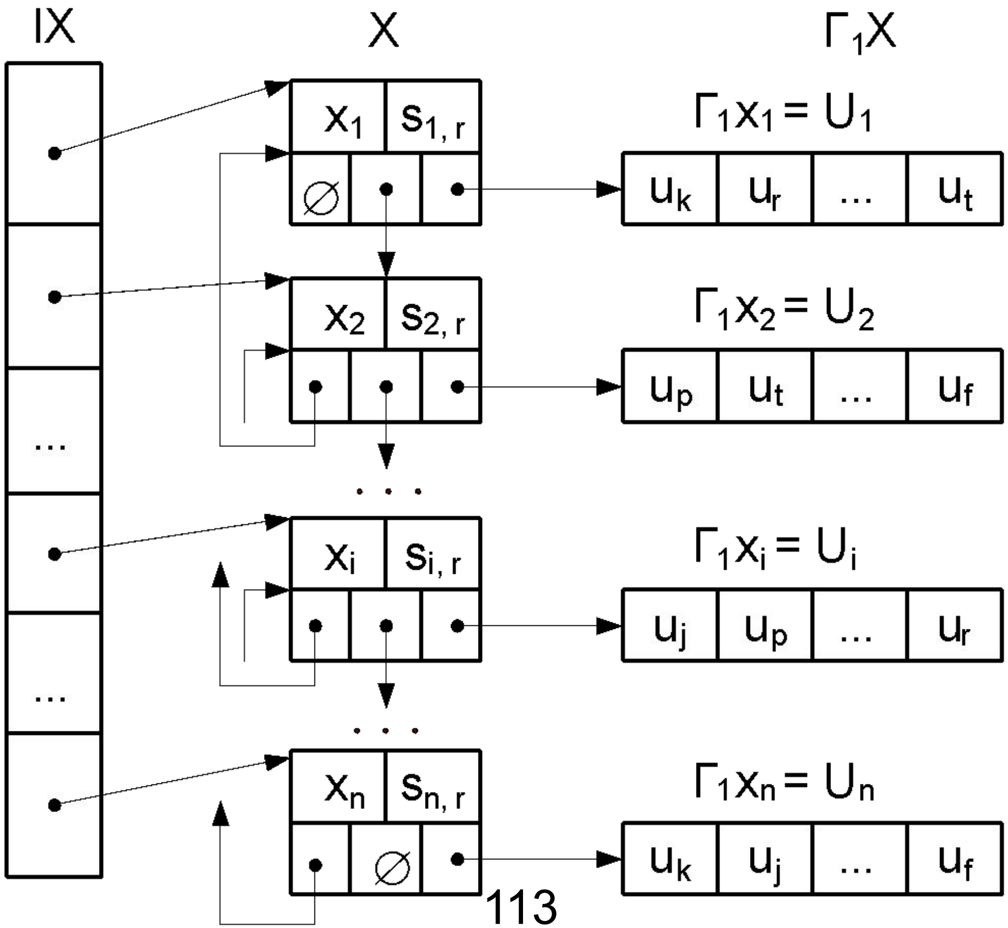
Необходимо разработать структуры данных для представления гиперграфа в форме $H(X,U,\Gamma_1 X,\Gamma_2 U)$. Основные операции алгоритма – добавление/удаление вершин и удаление ребер только из множества U.

Поскольку операции удаления/добавления подразумевают предварительный поиск с вычислительной сложностью O(n) необходимо организовать прямой доступ к вершинам и ребрам графа.

Вершины взвешены значениями некоторого локального критерия.

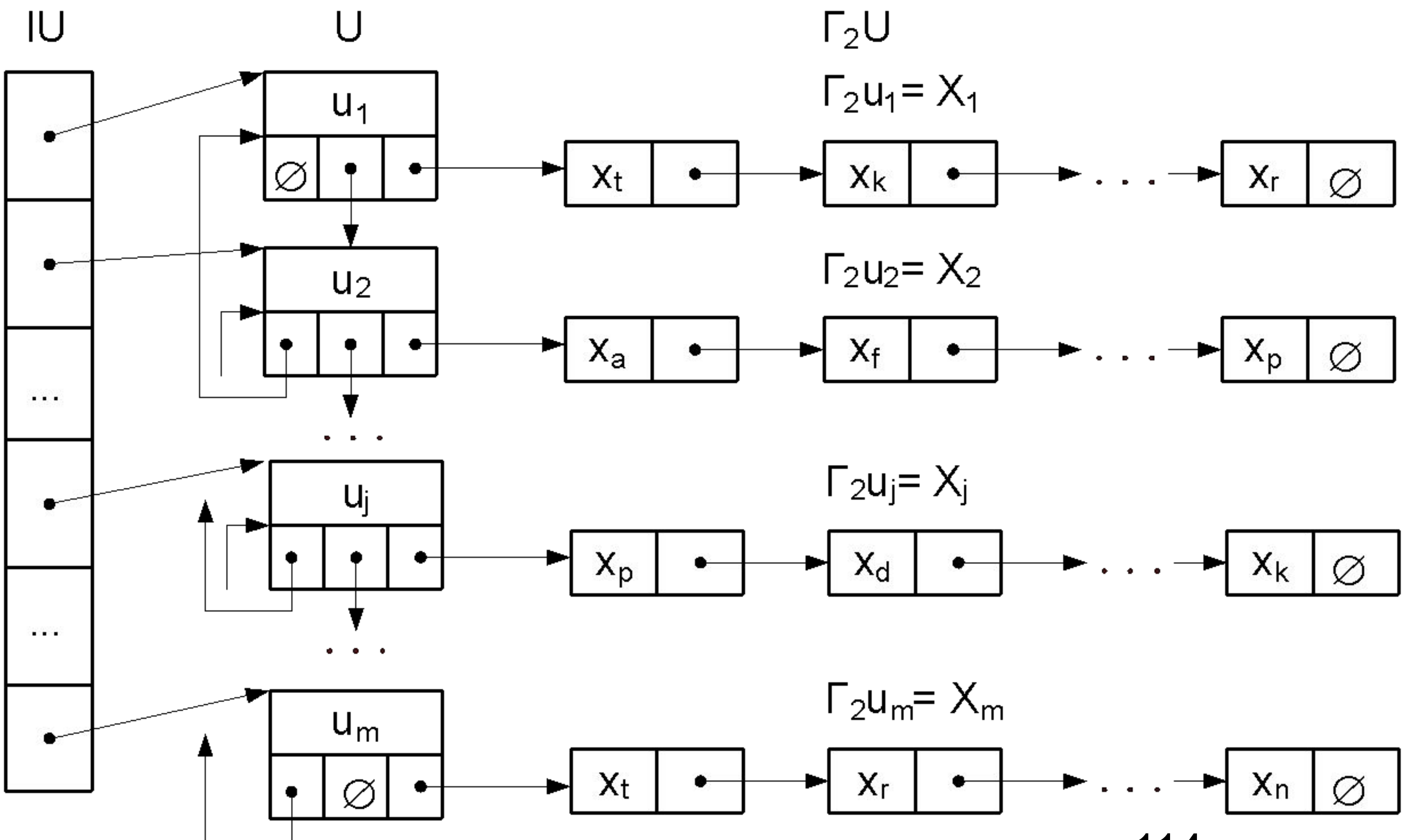
Таким образом для представления X и U следует использовать комбинацию вектора указателей с двусвязным списком.

Добавление/удаление вершины x_i влечет добавление/удаление всего множества $\Gamma_1 x_i$, следовательно $\Gamma_1 X$ – множество векторов.



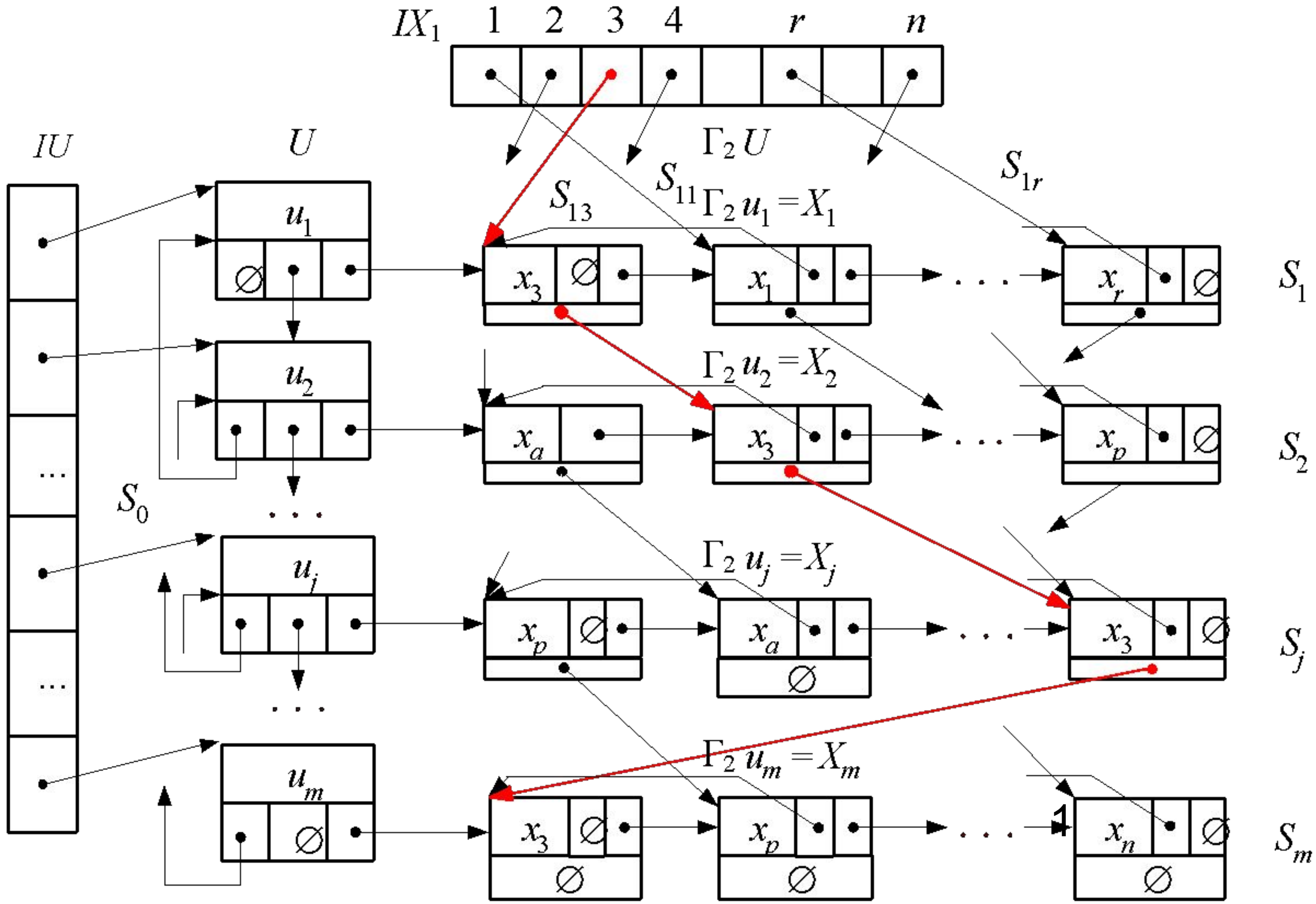
Пример двухуровневой комбинированной структуры данных (2)

При удалении вершины из множества X ее необходимо удалять из всех $Г_2 u_j$, в которых она присутствует. Таким образом $Г_2 u_j$ в простейшем случае – списковая структура.

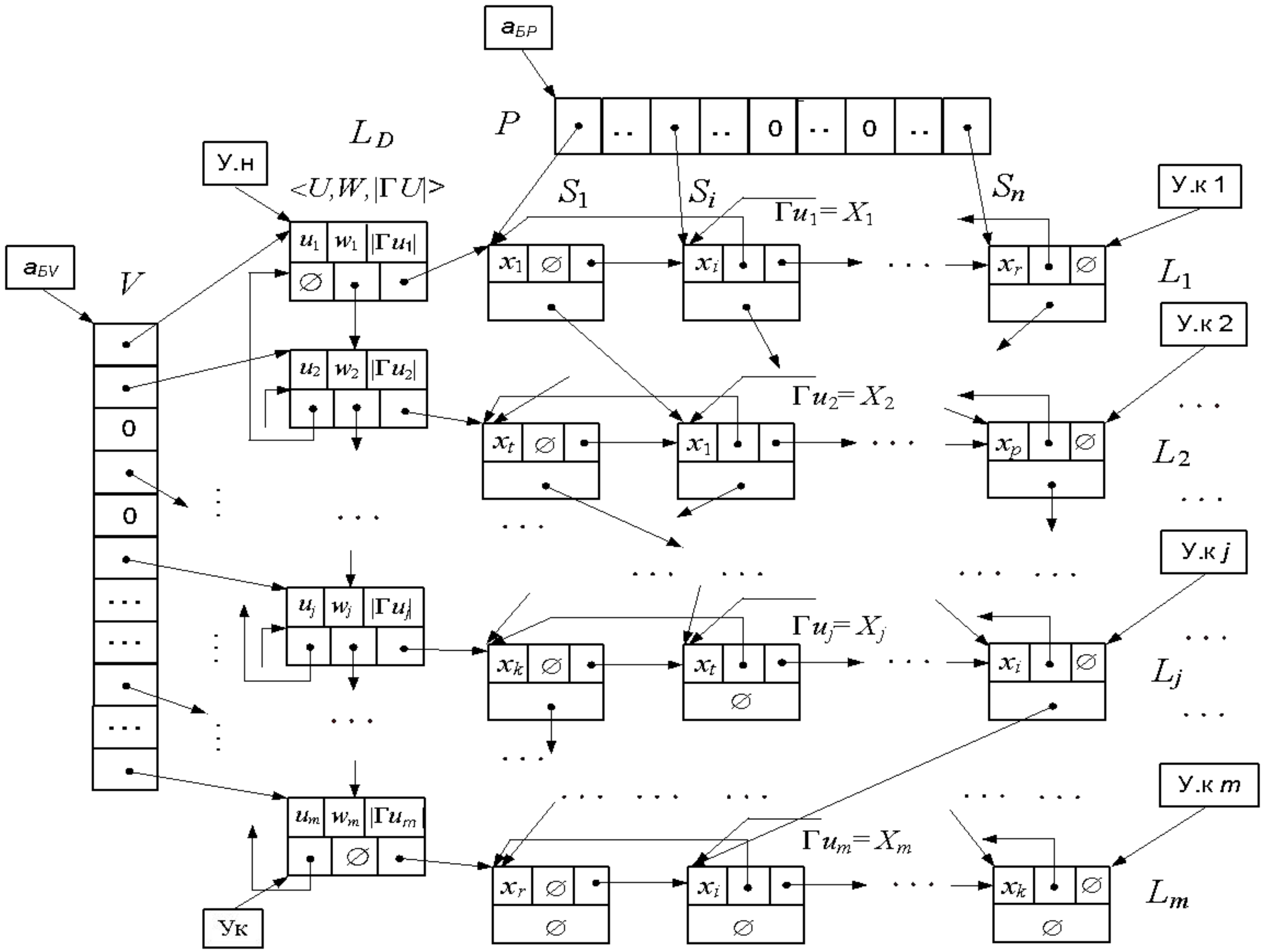


Пример двухуровневой комбинированной структуры данных (3)

В разработанной выше структуре поиск x_i в $\Gamma_2 U$ требует максимум *nm* операций сравнения. Для организации прямого доступа к x_i во всем $\Gamma_2 U$ достаточно использовать вектор указателей IX_1 и соответствующие указатели в списковых структурах $\Gamma_2 U$.

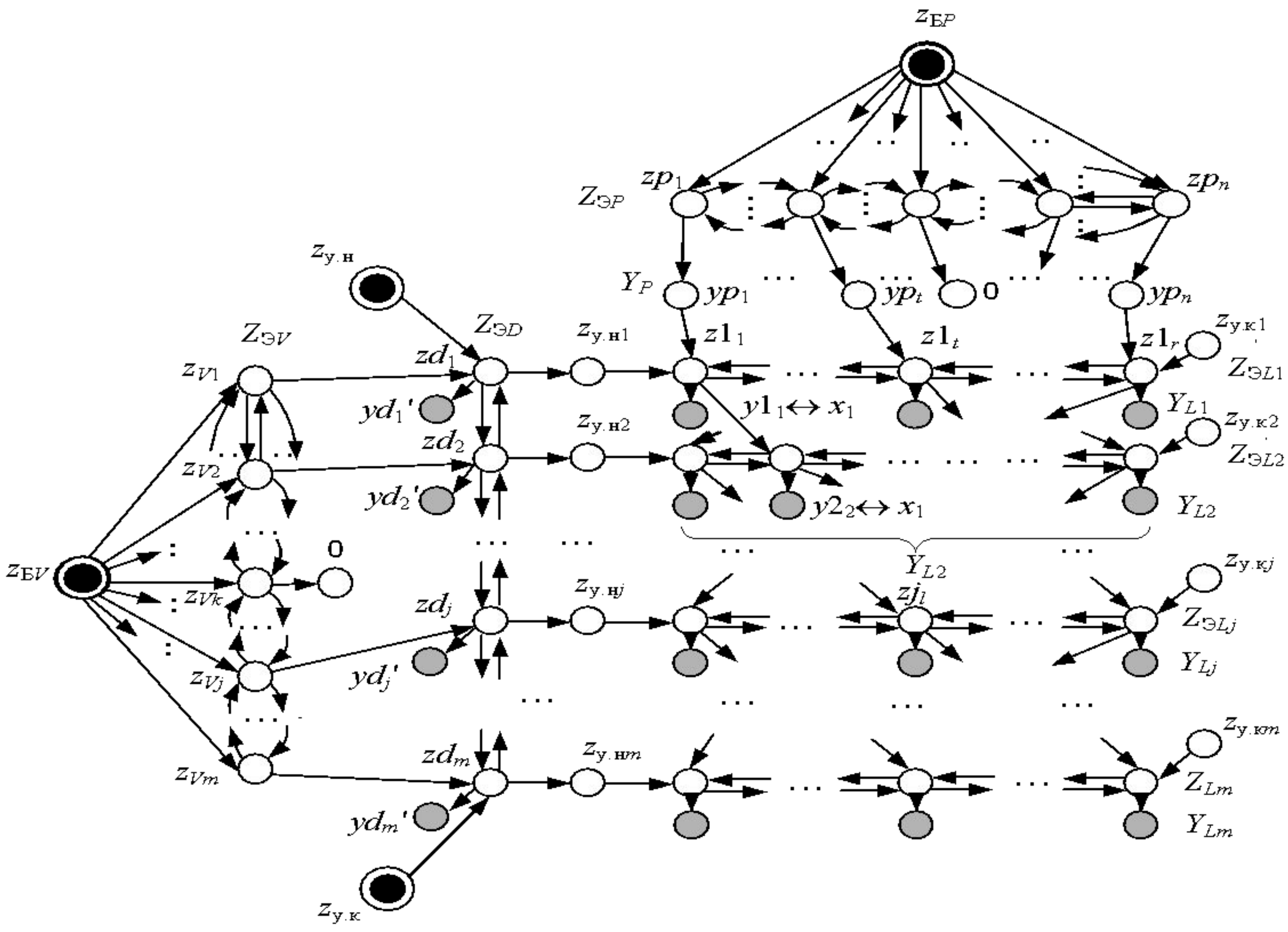


Трехсвязный список с векторами прямого доступа



Модель двухуровневой комбинированной структуры

На рисунке представлена модель двухуровневой структуры – комбинация трехсвязного списка с векторами прямого доступа



$yd_j' \leftrightarrow < u_j,\ w_j,\ |\Gamma u_j| >,\ yp_t = z_{у.н}S_t$

2.12 Математическая модель алгоритма

Для автоматизации анализа вычислительной и емкостной сложности, а также выполнения оптимизирующих преобразований и проверки правильности трансляции алгоритма, написанного на некотором формальном языке высокого уровня, *необходимо иметь его математическую модель.*

Со структурной и процедурной точек зрения алгоритм – совокупность операций, выполняемых в заданной или вычисляемой последовательности и обрабатывающих набор структурированных данных. *Таким образом компонентами структуры алгоритма являются операции преобразования данных, операции управления и связи между ними.*

Для задач *анализа* алгоритмов, в том числе *потокового,* и *формального выполнения эквивалентных преобразований* необходимо знать порядок выполнения операторов, т. е. существенным является отношение преемник-предшественник и значение предиката условия, по которому происходит переход.

.

Математическая модель алгоритма

Элементарный базис логической структуры алгоритма составляют операторы:

- начала и конца работы алгоритма;
- ввода/вывода (передачи) данных;
- обработки данных, изменяющие их значения;
- вычисления условий;
- ветвления потоков управления;
- слияния потоков управления.

Таким образом, множество типов операторов $V = \{v_i \ / \ i = 1, 8\}$, где v_1 – начало, v_2 – ввод данных, v_3 – обработка данных, v_4 – слияние потоков управления, v_5 – вычисление условий, v_6– ветвление потоков управления, v_7– вывод данных, v_8 – конец.

Математическая модель алгоритма

Помимо операций и их связей, компонентами алгоритма, отражающими процедурный подход, являются входные и получаемые данные, а также связи данные-операции и наоборот.

Оценка вычислительной и емкостной сложности требует следующей информации:

- вид операций, их вычислительная сложность в функции от базисных и количество повторений этих операций;
- характеристики входа задачи, промежуточных и окончательных данных, типы структур, в которые эти данные организованы, вычислительную сложность базовых операций над используемыми структурами данных.

Для оценки количества операций и вычислительной сложности алгоритма необходимы также сведения о связях между операциями и вероятностях передач управления.

Математическая модель алгоритма

В математической модели должны быть отражены следующие компоненты алгоритма, связи между ними, характеристики компонентов и свойства связей:

1) операторы ввода/вывода данных;

2) операторы преобразования данных, их типы, реализуемые ими функции и вычислительная сложность их выполнения;

3) операторы вычисления условий, предикаты, реализующие проверку условий, и вычислительная сложность этой проверки;

4) управляющие связи между операторами с учетом отношений следования, условий и вероятности их реализации, причем последние две характеристики относятся только к альтернативным передачам управления;

5) исходные, промежуточные и окончательные данные, их вид, размерности и, возможно, структуры;

Математическая модель алгоритма

6) связи данные – операторы и наоборот, их типы, определяющие вычислительную сложность чтения-записи данных для различных структур.

7) связи данные – данные, позволяющие решать вопрос об их независимости.

Для отображения операторов и отношений между ними используется *управляющий граф* $G_y^{\rightarrow}(X, F_1X)$ с двумя выделенными вершинами x_1 ($x_н$) и x_n ($x_к$) - начало и конец.

Управляющий граф называется *правильным*, если все его вершины принадлежат хотя бы одному пути из $x_н$ в $x_к$.

Алгоритмы, построенные в одном операторном базисе с одинаковой структурой, выполняющие обработку данных по различным функциональным зависимостям, ход работы которых определяется результатами проверки разных наборов условий, составляют некоторый класс.

Математическая модель алгоритма

Моделью алгоритмов данного класса является управляющий граф, в котором не отражены конкретные наборы данных, функциональные зависимости и предикаты, реализующие проверку условий. Эта модель несет информацию необходимую для изучения свойств всего класса алгоритмов.

Кроме структуры алгоритма класс определяет базис B, в который входят четыре непересекающихся множества $B = \{D_B, \Phi_B, P_B, O_B\}$, где D_B – символы переменных; Φ_B – символы функций; P_B – символы предикатов, реализующих проверку условий; O_B – символы операторов.

В нашем случае $O_B = V$, а операторы обработки данных $O' \subset O$ и вычисления условий $O'' \subset O$, где O – множество операторов алгоритма, должны быть помечены функциональными и предикатными символами из Φ_B и P_B соответственно.

Математическая модель алгоритма

События передачи управления порождают отношения достижимости оператор – оператор, обозначим его $Д_1(O, O)$. Связи между операторами и данными, определяющими является ли данное выходным или входным, задаются отношениями $Д_2(O, D)$ и $Д_3(D, O)$ соответственно. Поскольку эти события и связи не являются «самостоятельными» элементами, граф алгоритма целесообразно в большинстве случаев задавать множеством вершин и предикатом их смежности.

Основываясь на выполненном анализе, переход от алгоритма к управляющему графу осуществляется по следующим правилам:

1. Множеству операторов алгоритма O поставим во взаимно однозначное соответствие множество вершин графа X:

$$O \leftrightarrow X.$$

Математическая модель алгоритма

2. Тип и вычислительную сложность оператора отобразим, задав однозначное (возможно, взаимно однозначное) отношения R_1 и R_2 из множества X во множества V и T:

$$X\,R_1\,V \text{ и } X\,R_2\,T,$$

где $v \in V$ и $t \in T$, V и T – множество типов операторов и значений их вычислительной сложности.

3. Символы множеств Φ_B и P_B базиса отобразим в модели, задав однозначные (возможно взаимно однозначные) отношения R_3 и R_4 из множеств $X' \leftrightarrow O'$ вершин обработки данных и $X'' \leftrightarrow O''$ вершин вычисления условий во множества Φ_B и P_B соответственно:

$$X'\,R3\Phi B \text{ и } X''\,R4\;PB.$$

4. Множеству передач управления алгоритма C поставим во взаимно однозначное соответствие множество рёбер графа U:

$$C \leftrightarrow U.$$

Математическая модель алгоритма

5. Отношения $П_1(O, C)$ – «управление передаётся от оператора» и $П_2(C, O)$ – «управление передаётся к оператору» будут соответствовать двуместным предикатам инцидентности:

$$П_1(O, C) \sim Г_1(X, U) \text{ и } П_2(C, O) \sim Г_2(U, X).$$

6. Отношение достижимости операторов $Д_1(O, O)$ будет соответствовать двуместному предикату смежности $F_1(X, X)$ множества вершин:

$$Д_1(O, O) \sim F_1(X, X).$$

7. Вершинам, принадлежащим образам F_1x_i вершин разветвления потока управления, присвоим вес из множества $L = \{true, false\}$, определяющий связь со значением предиката, вычисленным в вершине проверки условия, и вес из множества E – вероятностей переходов. Для чего, зададим однозначные отношения R_5 и R_6 из подмножеств F_1x_i во множества L и E:

$$F_1x_i\ R_5\ L,\ F_1x_i\ R_6\ E.$$

Математическая модель алгоритма

8. Вершинам прообразов $F_1^{-1}x_i$ вершин разветвления потока управления, присвоим аналогичные веса:

$$x_i\ R_7\ L,\ x_i\ R_8\ E.$$

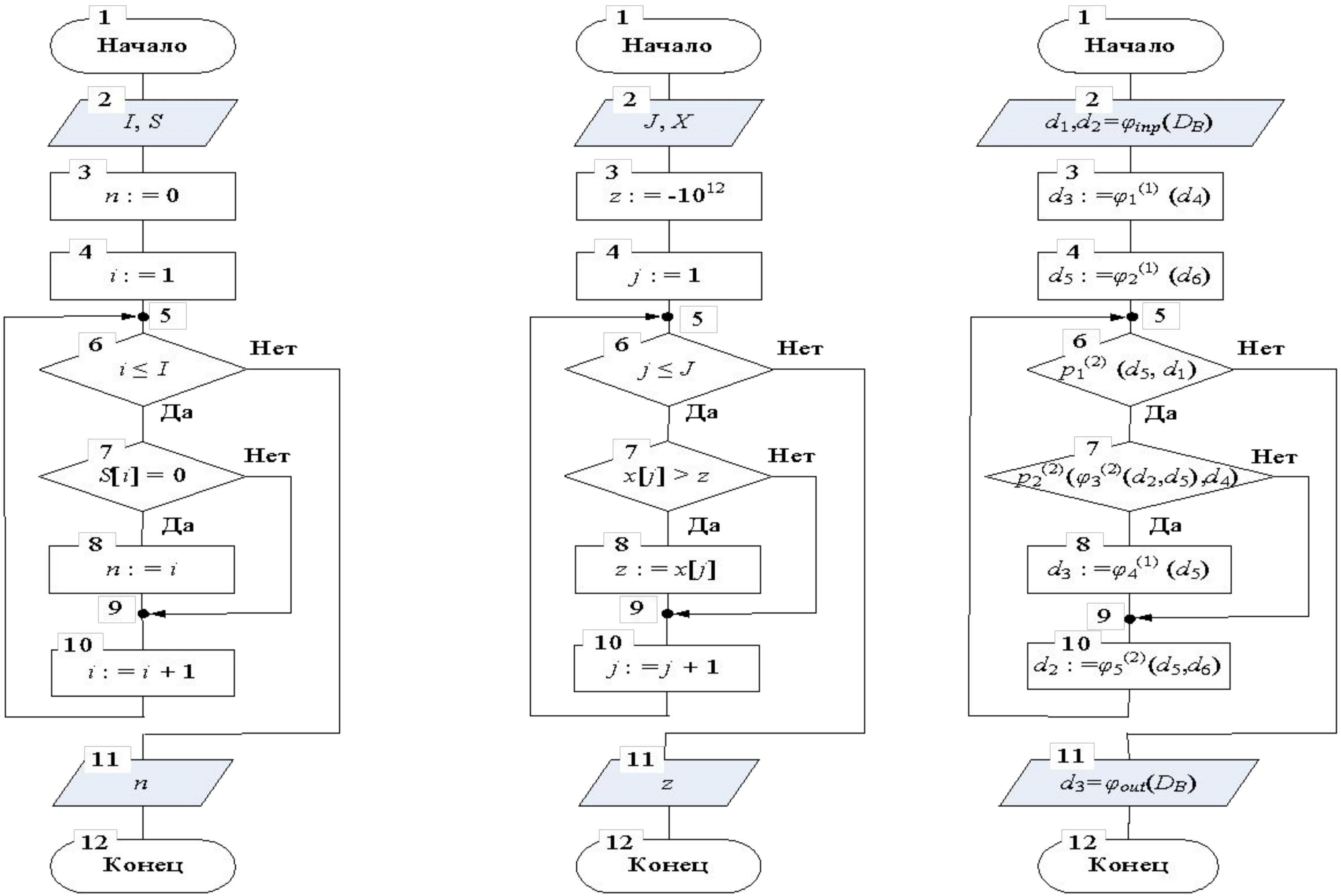
Полученный ориентированный управляющий граф с взвешенными вершинами в множестве X и в подмножествах, представляющих их образы и прообразы, задает структуру класса алгоритмов с учетом порядка выполнения операторов, их типов или вычислительной сложности и вероятности осуществления передач управления. Как правило его достаточно задавать в форме Gу($<X, V, T, (ФB, PB)>, \{<F_1X, L, E>, < F_1^{-1}X, L, E>\}$).

Математическая модель алгоритма

На рисунке следующего слайда показаны схемы алгоритмов программ: *A*1 – определения номера последнего равного нулю элемента массива *S* (рис. *а*), *A*2 – нахождения максимального элемента массива *X* (рис. *б*) и схема класса алгоритмов, к которому они принадлежат (рис. *в*).

Математическая модель алгоритма

Схемы алгоритмов



а *б* *в*

Математическая модель алгоритма

На рис. *в:*

$D_B = \{d_i \,/ i = 1,6\}$, где d_i – символы данных;

$\Phi_B = \{\varphi_{inp}, \varphi_1^{(1)}, \varphi_2^{(1)}, \varphi_3^{(2)}, \varphi_4^{(1)}, \varphi_5^{(2)}, \varphi_{oupt}\}$;

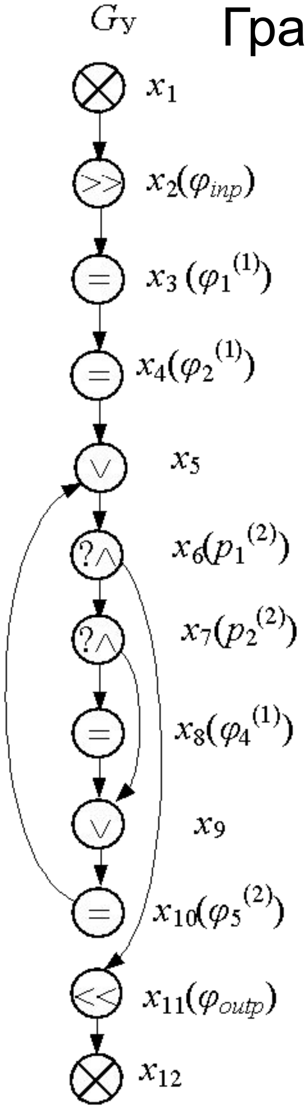
где $\varphi_1^{(1)}, \varphi_2^{(1)}, \varphi_4^{(1)}$ – символы одноместных функций присваивания значения аргумента, $\varphi_3^{(2)}$ – символы двуместных функций сложения аргументов, $\varphi_5^{(2)}$ – символ двуместной функции определения адреса элемента массива;

$PB = \{p_1^{(2)}, p_2^{(2)}\}$,

где $p_1^{(2)}, p_2^{(2)}$ – символы двуместных предикатов, реализующих проверку условий (верхний индекс обозначает местность функционального или предикатного символа).

Математическая модель алгоритма

Граф $G_У$ алгоритмов данного класса



$\otimes$ –терминальные блоки: v_1 - начало, v_8 - конец;

(>>) –v_2 - ввод данных;

(=) – v_3 - обработка данных;

($\vee$) – v_4 - слияние потоков управления;

(?$\wedge$) – $v_{5,6}$ - объединённый блок вычисления условий и ветвления;

(?) – v_5 - вычисление условий;

($\wedge$) – v_6 - ветвление потоков управления;

(<<) –v_7- вывод данных.

Математическая модель алгоритма

Вторая компонента модели алгоритма должна отображать отношения данные – операторы, разделяя для каждого оператора данные на входные и выходные, и позволять определять функциями каких входных данных является результат выполнения оператора (в том числе и управления). Таким образом, отношение данное – оператор или оператор – данное является бинарным.

В этой модели нет необходимости задавать связи между операторами, типы операторов (вычислительную сложность их выполнения), так как они уже отображены в графе $G_{\text{У}}$.

Связи данные – данные, задающие их зависимость, также являются бинарными и могут быть определены по связям данные – оператор, в этом случае данные являются входными, и связям оператор – данное, которое является результатом функции, реализуемой в этом операторе.

Математическая модель алгоритма

Тогда вторая модель алгоритма будет представлять собой двудольный граф, множество вершин которого $Z = X \cup Y$, причем $X \leftrightarrow O$, $Y \leftrightarrow D$, где O – множество операторов алгоритма, D – множество данных, принадлежащих области определения. Для класса алгоритмов множество данных D – это множество символов переменных базиса D_B. Элементы множества D_B абстрагированы от таких свойств и характеристик как вид данных (скаляр или множество), их размерность и структура.

Таким образом, в качестве второй компоненты модели алгоритма (модели «операторы – данные») будем использовать взвешенный двудольный ориентированный граф.

Математическая модель алгоритма

Переход к этому графу выполняется по правилам:

1. Множеству операторов алгоритма O поставим во взаимно однозначное соответствие подмножество вершин $X \subset Z$, а множеству данных D (символов переменных базиса D_B) – подмножество вершин $Y \subset Z$:

$$O \leftrightarrow X,\ D \leftrightarrow Y, \text{ причем } X \cap Y = \varnothing,\ X \cup Y = Z.$$

Здесь во множество данных D включены и промежуточные результаты работы алгоритма.

2. Отношения $Д_2(O, D)$ и $Д_3(D, O)$ будут соответствовать двуместным предикатам смежности $F_2(X, Y)$ и $F_3(Y, X)$:

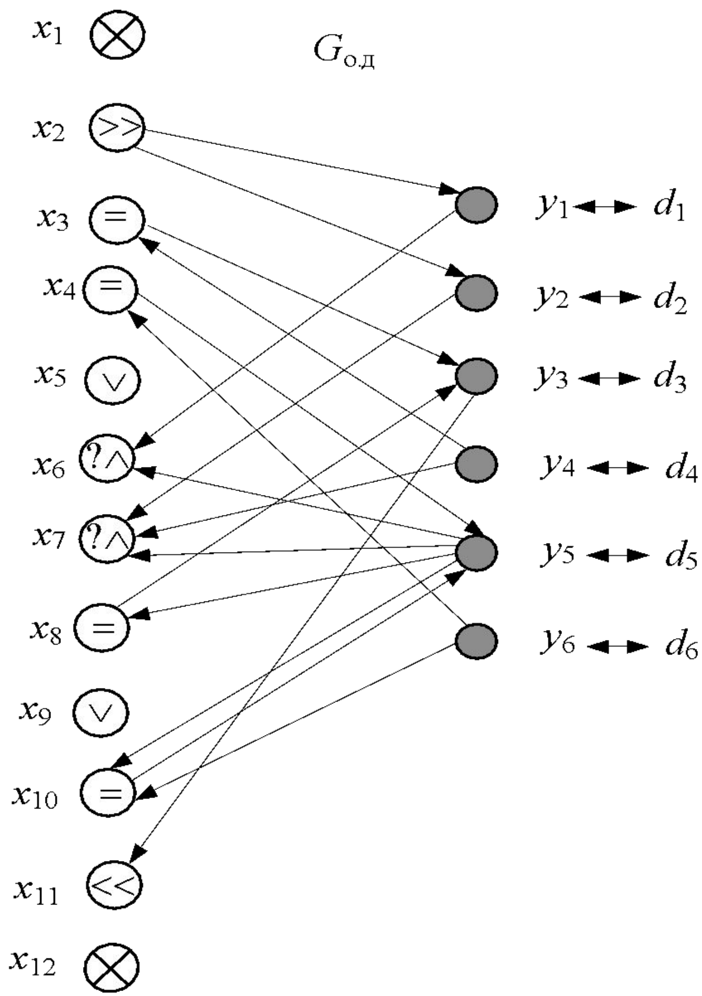
$$Д_2(O, D) \sim F_2(X, Y) \text{ и } Д_3(D, O) \sim F_3(Y, X).$$

В качестве второй компоненты модели класса алгоритмов, отобра жающей данные, операторы и отношения между ними, получаем ориентированный двудольный граф:

$$G\text{о.д}(\{X, Y\}, F_2X, F_2^{-1}X, F_3Y, F_3^{-1}Y).$$

Математическая модель алгоритма

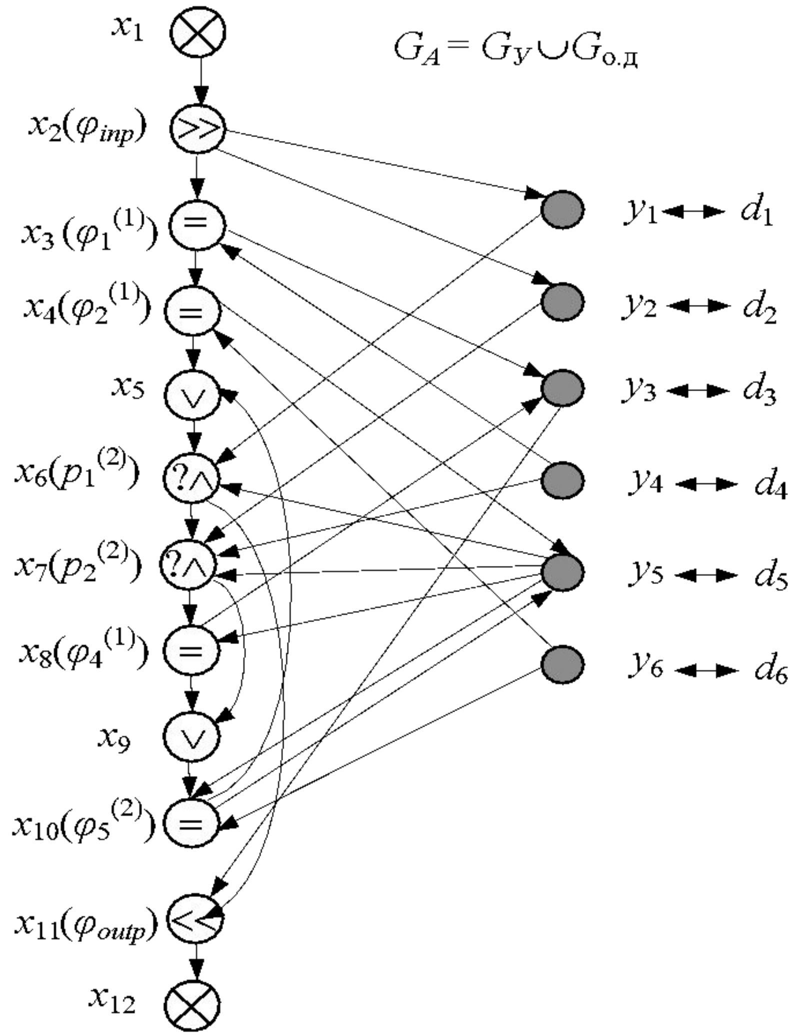
Модель класса алгоритмов, отображающая отношения между данными и операторами



Математическая модель алгоритма

Интегральной моделью класса алгоритмов будет граф:

$G_A(\{X, Y\}, F_1X, F_1^{-1}X , F_2X, F_2^{-1}X, F_3Y, F_3^{-1}Y) = G_У \cup G_{о.\,д}$.



Математическая модель алгоритма

В модели конкретного алгоритма должны быть отражены реализуемые функции и предикаты, а также вид данных области его определения. Для получения модели конкретного алгоритма необходимо выполнить интерпретацию I_1 модели класса алгоритмов в область определения данного алгоритма. Для алгоритма $A1$ получим:

$I_1(D_B) : d_1 = I, d_2 = S, d_3 = n, d_4 = 0, d_5 = i, d_6 = 1;$

$I_1(\Phi_B) :$

$I_1(\varphi_{inp}(D_B)) = I, S;$

$I_1(\varphi_1{}^{(1)}\ (D_B)) = \varphi_1{}^{(1)}\ (d_4) = 0;$

$I_1(\varphi_2{}^{(1)}\ (D_B)) = \varphi_2{}^{(1)}\ (d_6\ ) = 1;$

$I_1(\varphi_3{}^{(2)}\ (D_B)) = \varphi_3{}^{(2)}\ (d_2, d_5) = S[i];$

$I_1(\varphi_4{}^{(1)}\ (D_B)) = \varphi_4{}^{(1)}\ (d_5) = i;$

$I_1(\varphi_5{}^{(2)}\ (D_B)) = \varphi 5^{(2)}\ (d_5, d_6) = i + 1;$

Математическая модель алгоритма

$I_1(P_B)$:

$I_1(p_1^{(2)}\ (D_B)) = p_1^{(2)}\ (d_5, d_1)$ = «i меньше или равно I»;

$I_1(p_2^{(2)}\ (\Phi_B, D_B)) = p_2^{(2)}\ (\ \varphi_3^{(2)}\ (d_2, d_5), d_4)$ = «$S[i]$ равно 0»;

$I_1(\varphi_{outp}(D_B)) = n$;

$I_1(O_B) = V$.

В модели «операторы - данные» конкретного алгоритма должны быть отражены такие свойства и характеристики данных области определения как вид, размерность и структура памяти для множеств, вычислительные сложности чтения и записи данных для определенных структур.

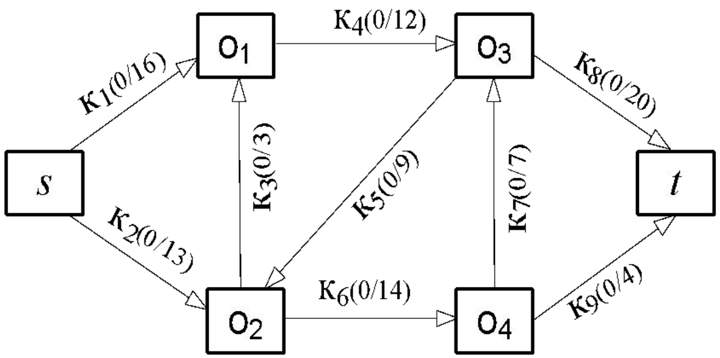
2.13 Модель сети

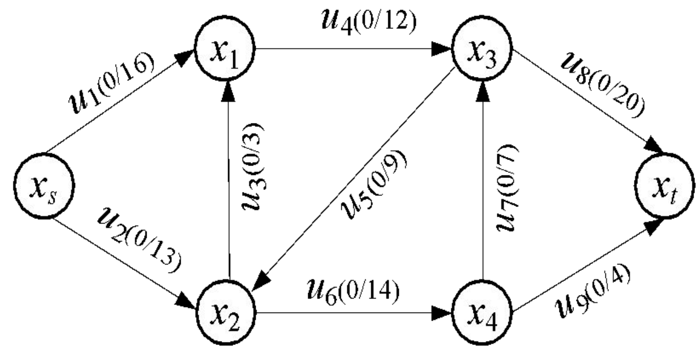
Неформально сеть можно определить следующим образом: имеется система, состоящая из некоторого множества объектов, связанных каналами передачи сообщений. Один из этих объектов является только источником сообщений (*s*), другой – только приемником (*t*). Остальные объекты выполняют прием и дальнейшую передачу сообщений без потерь. Каналы передачи сообщений имеют ограниченную пропускную способность. Необходимо определить максимальное количество сообщений по сети.

На следующем слайде (рис.*а*) показано исходное состояние некоторой сети. Около каналов в круглых скобках через косую черту указаны количество сообщений при данном состоянии системы и пропускная способность канала.

Модель сети

Моделью сети является ориентированный граф $G^{\rightarrow}(X, \langle U, f, c \rangle)$





$X \leftrightarrow$ О, где О = {s, о$_1$, о$_2$, . . . , о$_{n\text{-}2}$, t} – объекты системы, и $X = \{x_s, x_1, x_2, \ldots, x_{n\text{-}2}, x_t\}$;
К $\leftrightarrow U$, где К – каналы передачи сообщений, $U = \{u_1, u_2, \ldots, u_m\}$;
функция f: $U \rightarrow R$ – *поток* в сети, значение которой показывает количество сообщений по каналам в данном состоянии системы;
c – функция, задающая пропускную способность каналов.

Таким образом $f(u_j)$ – поток, передаваемый по ребру u_j, а $c(u_j)$ – пропускная способность этого ребра.