МЕТОД ПРОРИСОВКИ ГИПЕРРЕБЕР ИЕРАРХИЧЕСКОГО МНОГОСЛОЙНОГО ГИПЕРГРАФА

ПМ-1701

Гиперграф

Ребра в диаграммах потоков данных прорисовывают вертикальными или горизонтальными сегментами[1], а иерархический граф рассматривается как иерархический гиперграф, который отличается от обычного графа тем, что его ребра соединяют не две вершины различных слоев, а некоторое множество вершин (источников) верхнего слоя с некоторым множеством вершин (стоков) расположенных в нижних слоях.

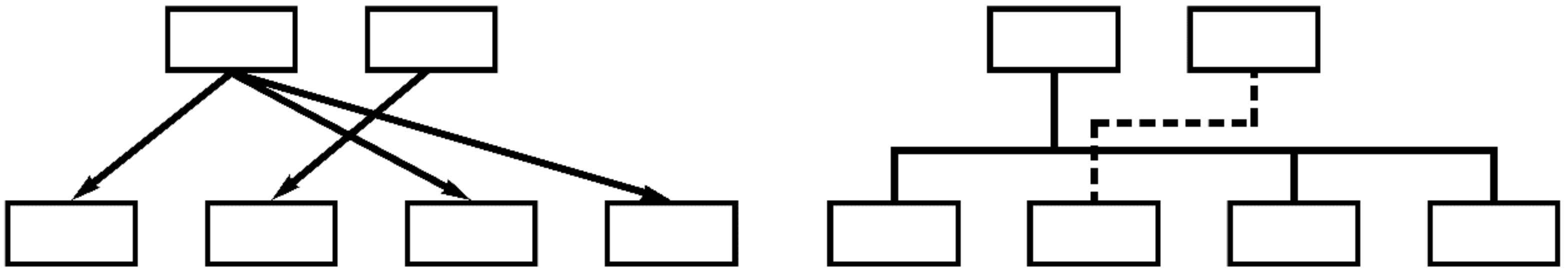

Рис. 1. Иерархический граф (слева) и соответствующий ему гиперграф с гиперребрами (справа).

[1] Schulze C. D. Optimizing Automatic Layout for Data Flow Diagrams // *Diploma Thesis*, Christian-Albrechts-Universität zu Kiel, 2011.

Гиперграф

В математических терминах, k-слойный гиперграф $H = (V, E_H, \lambda)$ – это граф, где каждой вершине $u \in V$ оператор λ ставит в соответствие натуральное число (номер слоя) $\lambda(u)$, $1 \leq \lambda(u) \leq k$, а E_H – множество гиперребер, при этом каждое гиперребро $e = (S, T)$ инцидентно множеству вершин-предков $S \subset V$ и множеству вершин-потомков $T \subset V$. Если S состоит только из одной вершины, то такое гиперребро называется «одноисточным»[2].

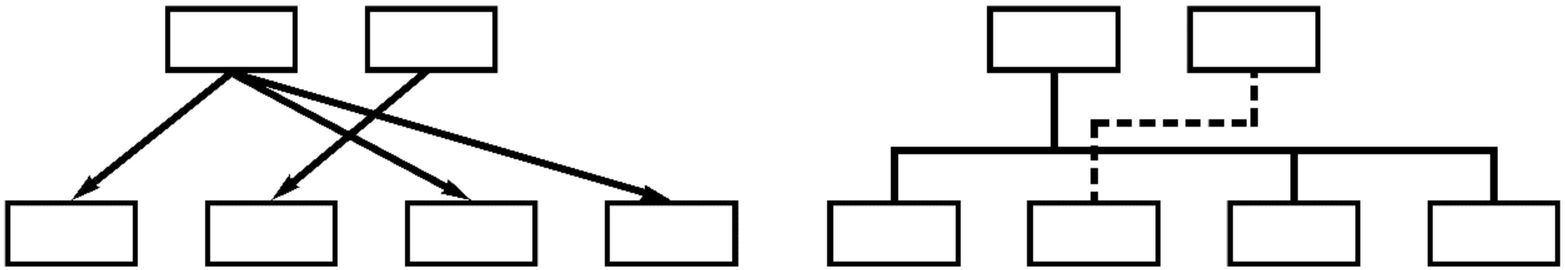

Рис. 1. Иерархический граф (слева) и соответствующий ему гиперграф с гиперребрами (справа).

[2] Sander G. Layout of directed hypergraphs with orthogonal hyperedges // *Graph Drawing*, vol. 2912 of the series Lecture Notes in Computer Science, 2003, pp. 381–386.

Требования к прорисовке

Ниже представлены требования к прорисовке гиперребер двухслойного гиперграфа, учитывающие особенности применения иерархических гиперграфов для визуализации финансовых потоков:

1. Вершины представлены в виде прямоугольников фиксированного размера шириной 2ρ и высотой ρ.
2. Каждое гиперребро инцидентно одной вершине-источнику и смежным вершинам нижнего слоя.
3. Каждое гиперребро состоит из трех сегментов:

- первый сегмент sV_1 представляет собой вертикальный отрезок, выходящий из вершины-источника;
- второй сегмент sH представляет собой горизонтальный отрезок, проходящий на одной ординате с ординатой нижнего конца сегмента sV_1 и имеющий с ним общую точку;
- третий сегмент sV_2 представляет собой набор вертикальных отрезков, начинающихся на сегменте sH и входящих в вершины-стоки.

Требования к прорисовке

4. Наложение двух гиперребер возможно только для сегментов sV_2. Во всех остальных случаях наложение горизонтальных и вертикальных сегментов недопустимо.

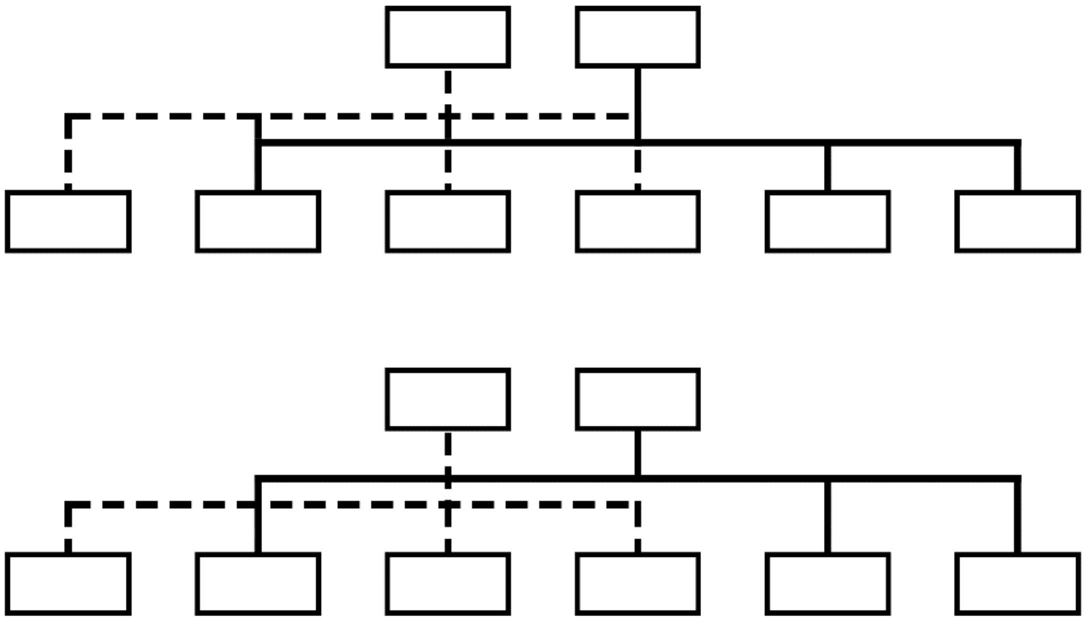

Рис. 2. Недопустимое наложение первого вертикального сегмента «сплошного» гиперребра и второго вертикального сегмента «пунктирного» гиперребра (сверху); корректная укладка гиперграфа, полученная изменением порядка горизонтальных сегментов гиперребер (снизу).

Требования к прорисовке

5. Если два гиперребра имеют общую вершину-сток, и при этом для одного гиперребра она является крайней левой среди всех инцидентных этому ребру вершин-стоков, а для второго гиперребра – крайне правой, то горизонтальные сегменты гиперребер могут иметь одинаковую ординату.

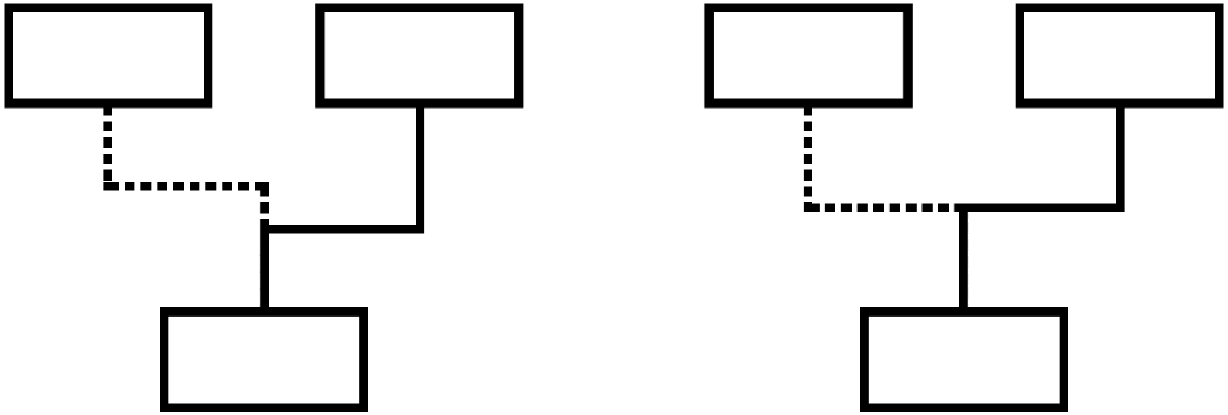

Рис.3. Гиперграф без пересечения горизонтальных сегментов (слева) и с пересечением горизонтальных сегментов (справа).

Требования к прорисовке

6. Сегмент sV_1 каждого гиперребра выходит строго из середины вершины-источника[3].

7. Все вертикальные отрезки сегмента sV_2 каждого гиперребра входят строго в середину каждой инцидентной ему вершины-стока.

[3] Sander G. Layout of directed hypergraphs with orthogonal hyperedges // *Graph Drawing*, vol. 2912 of the series Lecture Notes in Computer Science, 2003, pp. 381–386.

Требования к прорисовке

Если гиперграф содержит Q гиперребер, то горизонтальный сегмент каждого гиперребра может быть расположен на одной из t горизонталей, пронумерованных от 1 до Q сверху вниз.

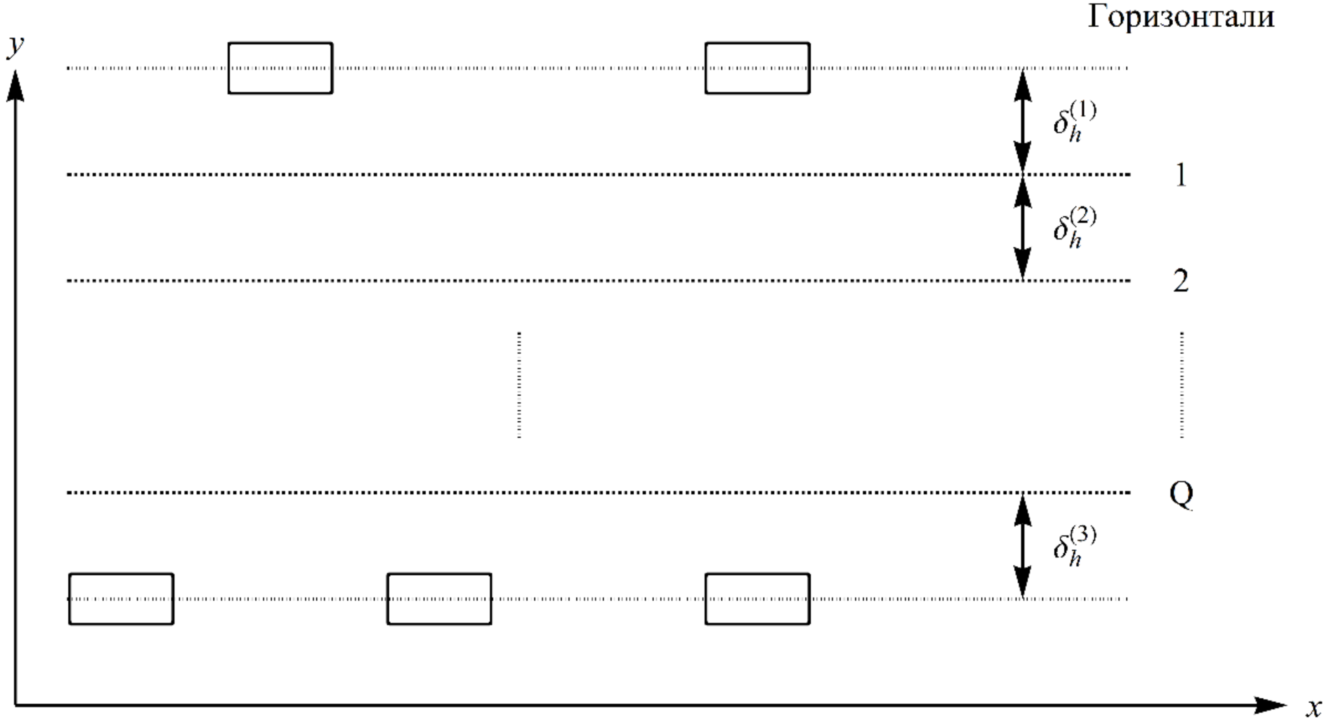

Рис. 4. Шаблон прорисовки гиперребер для двухслойного гиперграфа.

Требования к прорисовке

Пересечением двух гиперребер назовем наличие общей точки горизонтального сегмента первого гиперребра с одним из вертикальных сегментов второго. Число пересечений гиперребер, которое, очевидно, зависит от взаимного расположения их горизонтальных сегментов, будет равно количеству таких общих точек.

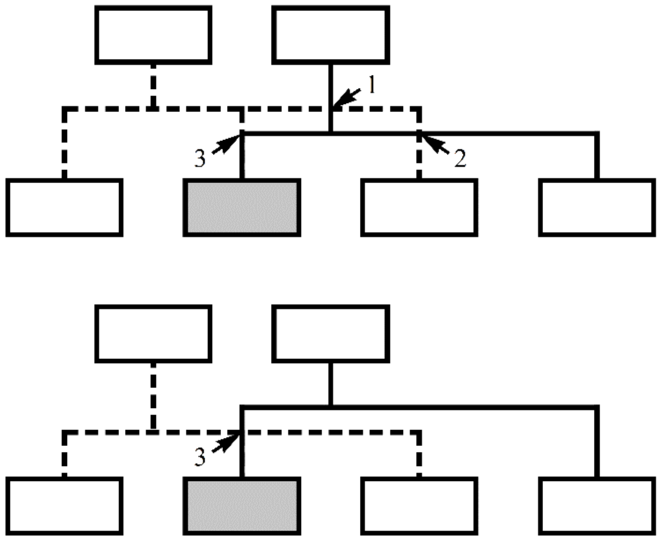

Рис. 5. Пересечения гиперребер различных типов.

Требования к прорисовке

Алгоритмы зависят от требований к прорисовке:

- Eichelberger H. Aesthetics and Automatic Layout of UML Class Diagrams *PHD Thesis*, Bayerischen Julius-Maximilians-Universität Würzburg, 2005.
- Eschbach T., Gunther W., Becker B. Orthogonal hypergraph drawing for improved visibility // *Journal of Graph Algorithms and Applications*, vol. 10, no. 2, 2006, pp. 141–157.
- Sander G. Layout of directed hypergraphs with orthogonal hyperedges // *Graph Drawing*, vol. 2912 of the series Lecture Notes in Computer Science, 2003, pp. 381–386.
- Sander G. A fast heuristic for hierarchical Manhattan layout // *Graph Drawing*, vol. 1027 of the series Lecture Notes in Computer Science, 2005, pp. 447–458.
- Schulze C. D. Optimizing Automatic Layout for Data Flow Diagrams // *Diploma Thesis*, Christian-Albrechts-Universität zu Kiel, 2011.
- Schulze C. D., Spönemann M., R. von Hanxleden. Drawing Layered Graphs with Port Constraints // *Journal of Visual Languages & Computing*, 2013, pp. 1–34.
- Siebenhaller M. Orthogonal Graph Drawing with Constraints: Algorithms and Applications // *PHD Thesis*, Eberhard-Karls-Universitat, 2009.

Требования к прорисовке

При выполнении прорисовки гиперребер будем руководствоваться следующими метриками эстетичности:

- число пересечений гиперребер ($M1$);
- номер самой нижней занятой горизонтали ($M2$);
- сбалансированное (центрированное) положение горизонтальных сегментов гиперребер ($M3$);
- отклонение относительного удлинения (отношение ширины рисунка к его высоте) укладки от общепринятых значений ($M4$).

Набор алгоритмов для прорисовки гиперребер

Дан двухслойный гиперграф $H_2 = (V_1, V_2, E_{H_2})$, где $V_1 = \{u_1, \dots, u_{N_1}\}$ и $V_2 = \{v_1, \dots, v_{N_2}\}$ – множества вершин верхнего и нижнего слоя соответственно, а E_{H_2} – множество одноисточных гиперребер. Для каждой вершины $v \in V_1 \cup V_2$ двухслойного гиперграфа известны координаты $(x(v), y(v))$.

Алгоритмы минимизации числа пересечений гиперребер

Алгоритмы для сокращения числа горизонталей

Алгоритм сбалансированное расположения горизонтальных сегментов гиперребер

Алгоритм минимизации отклонения относительного удлинения укладки от общепринятых значений

Алгоритмы минимизации числа пересечений гиперребер

Обобщенная постановка задачи минимизации пересечений гиперребер

Для удовлетворения всех требований, описанных в предыдущем разделе, был разработан алгоритм, включающий в себя следующие этапы:

1. предварительная обработка графа; на этом этапе производится подсчет пересечений второго типа для всех пар гиперребер с учетом взаимного расположения;

2. решение оптимизационной задачи в точной постановке; целью этапа является минимизация пересечений гиперребер и отсутствие недопустимых наложений. В результате должна быть получена укладка графа, где каждому гиперребру сопоставляется уникальная горизонталь;

3. постобработка укладки графа.

Алгоритмы минимизации числа пересечений гиперребер

Обобщенная постановка задачи минимизации пересечений гиперребер

Дан двухслойный гиперграф $H_2 = (V_1, V_2, E_{H_2})$, где $V_1 = \{u_1, \dots, u_{N_1}\}$ и $V_2 = \{v_1, \dots, v_{N_2}\}$ – множества вершин верхнего и нижнего слоя соответственно, а E_{H_2} – множество одноисточных гиперребер. Для каждой вершины $v \in V_1 \cup V_2$ двухслойного гиперграфа известны координаты $(x(v), y(v))$.

Каждое гиперребро $e_n = (u_\eta, T_\eta) \in E_{H_2}$ характеризуется списком $R_n = \{r_{n,1}, \dots, r_{n,|V_2|}\}$, элементы которого определены следующим образом:

$$r_{n,j} = \begin{cases} 1, & \text{если } v_j \in T_\eta \\ 0, & \text{если } x(v_j) \notin int \\ -1, & \text{если } v_j \notin T_\eta \text{ и } x(v_j) \in int \end{cases}, \quad j = 1, \dots, |V_2|,$$

где $int = \left[\min\limits_{v \in T_\eta}\{x(v), x(u_\eta)\}, \max\limits_{v \in T_\eta}\{x(v), x(u_\eta)\}\right]$ – это интервал на оси абсцисс, соответствующий горизонтальному сегменту гиперребра e_n.

Алгоритмы минимизации числа пересечений гиперребер

Обобщенная постановка задачи минимизации пересечений гиперребер

Списки R_n и R_m позволяют определить число пересечений второго типа c_{e_n,e_m} и c_{e_m,e_n} между гиперребрами $e_n = (u_\eta, T_\eta)$ и $e_m = (u_\mu, T_\mu)$ в зависимости от взаимного расположения их горизонтальных сегментов. Для каждой пары гиперребер $e_n = (u_\eta, T_\eta)$ и $e_m = (u_\mu, T_\mu)$ составляется матрица

$$M_{n,m} = \begin{pmatrix} r_{n,1} & \cdots & r_{n,|V_2|} \\ r_{m,1} & \cdots & r_{m,|V_2|} \end{pmatrix}.$$

Для расчета величины c_{e_m,e_n} (горизонтальный сегмент гиперребра вершины-источника e_m выше горизонтального сегмента гиперребра вершины-источника e_n), очевидно, необходимо в той же матрице $M_{n,m}$ подсчитать количество столбцов вида $\begin{pmatrix} -1 \\ 1 \end{pmatrix}$.

Алгоритмы минимизации числа пересечений гиперребер

Обобщенная постановка задачи минимизации пересечений гиперребер

Если избежать недопустимых наложений за счет взаимного расположения горизонтальных сегментов гиперребер не удается, то тогда сдвиг одной или нескольких вершин гиперграфа по горизонтали приведет к нужному результату. Сдвиг вершины $v \in V_1 \cup V_2$ осуществляется на определенное целое число «единичных сдвигов» длины Δ, при этом если число отрицательное, то сдвиг выполняется влево, а если положительное – то вправо. Суммарное количество единичных сдвигов всех вершин также необходимо минимизировать.

Алгоритмы минимизации числа пересечений гиперребер

Обобщенная постановка задачи минимизации пересечений гиперребер

Введем в рассмотрение 8 видов целочисленных переменных Z', Z'', AZ', AZ'', HH, $CT1$, $CT2$ и BV.

Переменные Z_i' и Z_j'' характеризуют количество единичных сдвигов вершин верхнего $u_i \in V_1$ и нижнего слоя $v_j \in V_2$ соответственно, а их знаки определяют направления сдвигов. Тогда новые (вследствие проведенных сдвигов) абсциссы вершин $u_i \in V_1$ и $v_j \in V_2$ будут равны

$$x'(u_i) = x(u_i) + \Delta \times Z_i',$$

$$x'(v_j) = x(v_j) + \Delta \times Z_j''.$$

Переменные AZ_i' и AZ_j'' – это количество единичных сдвигов вершин u_i и v_j, соответственно, т.е. $AZ_i' = |Z_i'|$, а $AZ_j'' = |Z_j''|$.

Алгоритмы минимизации числа пересечений гиперребер

Обобщенная постановка задачи минимизации пересечений гиперребер

Введем в рассмотрение 8 видов целочисленных переменных Z', Z'', AZ', AZ'', HH, $CT1$, $CT2$ и BV.

Для каждой пары гиперребер (e_n, e_m), где $e_n = \left(u_\eta, T_\eta\right)$, $e_m = (u_\mu, T_\mu)$ и $x\left(u_\eta\right) < x\left(u_\mu\right)$ переменные HH, $CT1$ и $CT2$ определены следующим образом:

- $HH_{n,m} = \begin{cases} 0, & \text{если горизонтальный сегмент } e_m \text{ выше, чем } e_n, \\ 1, & \text{в противном случае.} \end{cases}$
- $CT1_{n,m} = \begin{cases} 0, & \text{если нет пересечения первого типа,} \\ 1, & \text{в противном случае.} \end{cases}$
- $CT2_{n,m} = \begin{cases} c_{e_n,e_m}, & \text{если горизонтальный сегмент } e_n \text{ выше, чем } e_m, \\ c_{e_m,e_n}, & \text{в противном случае.} \end{cases}$

Алгоритмы минимизации числа пересечений гиперребер

Обобщенная постановка задачи минимизации пересечений гиперребер

Введем в рассмотрение 8 видов целочисленных переменных Z', Z'', AZ', AZ'', HH, $CT1$, $CT2$ и BV.

Переменные $BV_{\eta,j}$ – это булевы фиктивные переменные, которые вводятся для каждого гиперребера $e_n = (u_\eta, T_\eta)$ и множества вершин $\{v_j \mid v_j \in V_2 \setminus T_\eta\}$, т.е. вершин нижнего слоя, не смежных вершине u_η.

Алгоритмы минимизации числа пересечений гиперребер

Обобщенная постановка задачи минимизации пересечений гиперребер

Выбор целевой функции задачи целочисленного программирования обусловлен целями:

- минимизация пересечений первого и второго типа;
- минимизация сдвигов вершин.

Обозначенным целям удовлетворяет следующий вид целевой функции:

$$L = w_z \sum_{i=1}^{|\mathrm{V}_1|} AZ'_i + w_z \sum_{j=1}^{|\mathrm{V}_2|} AZ''_j + \sum_{n=1}^{|E_{H_2}|-1} \sum_{m=n+1}^{|E_{H_2}|} (w_{c_1} \times CT1_{n,m} + w_{c_2} \times CT2_{n,m}) \to \min \tag{1}$$

Алгоритмы минимизации числа пересечений гиперребер

Обобщенная постановка задачи минимизации пересечений гиперребер

Система ограничений задачи целочисленного программирования имеет вид:

$$
\begin{gathered}
CT1_{n,m} \geq \frac{x\left(u_{\eta}\right) + \Delta\, Z'_{\eta} - \left(x\left(v_j\right) + \Delta\, Z''_j\left(v_j\right)\right)}{K} - K\, HH_{n,m}, \\
e_n = \left(u_{\eta}, T_{\eta}\right) \in E_{H_2},\ \ e_m = \left(u_{\mu}, T_{\mu}\right) \in E_{H_2}, \\
x\left(u_{\eta}\right) < x\left(u_{\mu}\right),\ \ v_j \in T_{\mu},\ \ x\left(v_j\right) = \min_{v \in T_{\mu}} x(v)
\end{gathered}
\tag{2}
$$

$$
\begin{gathered}
CT1_{n,m} \geq \frac{x\left(v_j\right) + \Delta\, Z''_j\left(v_j\right)\ - (x\left(u_{\mu}\right) + \Delta\, Z'_{\mu})}{K} - K\left(1 - HH_{n,m}\right), \\
e_n = \left(u_{\eta}, T_{\eta}\right) \in E_{H_2},\ \ e_m = \left(u_{\mu}, T_{\mu}\right) \in E_{H_2}, \\
x\left(u_{\eta}\right) < x\left(u_{\mu}\right),\ \ v_j \in T_{\eta},\ \ x\left(v_j\right) = \max_{v \in T_{\eta}} x(v)
\end{gathered}
\tag{3}
$$

$$
\begin{gathered}
CT2_{n,m} \geq c_{e_n,e_m} - K\left(1 - HH_{n,m}\right), \\
e_n = \left(u_{\eta}, T_{\eta}\right) \in E_{H_2},\ \ e_m = \left(u_{\mu}, T_{\mu}\right) \in E_{H_2},\ \ x\left(u_{\eta}\right) < x\left(u_{\mu}\right)
\end{gathered}
\tag{4}
$$

$$
\begin{gathered}
CT2_{n,m} \geq c_{e_m,e_n} - K\, HH_{n,m}, \\
e_n = \left(u_{\eta}, T_{\eta}\right) \in E_{H_2},\ \ e_m = \left(u_{\mu}, T_{\mu}\right) \in E_{H_2},\ \ x\left(u_{\eta}\right) < x\left(u_{\mu}\right)
\end{gathered}
\tag{5}
$$

$$
\begin{gathered}
x(v_j) + \Delta\, Z_j''(v_j) \;\; \leq x(u_\eta) + \Delta\, Z_\eta' + K\; BV_{\eta,j} + K\; HH_{n,m} - \varepsilon, \\
e_n = (u_\eta, T_\eta) \in E_{H_2}, \;\; e_m = (u_\mu, T_\mu) \in E_{H_2}, \\
x(u_\eta) < x(u_\mu), \;\; v_j \in T_\mu
\end{gathered} \tag{6}
$$

$$
\begin{gathered}
x(v_j) + \Delta\, Z_j''(v_j) \;\; \leq x(u_\mu) + \Delta\, Z_\mu' + K\; BV_{\mu,j} + K(1 - \; HH_{n,m}) - \varepsilon, \\
e_n = (u_\eta, T_\eta) \in E_{H_2}, \;\; e_m = (u_\mu, T_\mu) \in E_{H_2}, \\
x(u_\eta) < x(u_\mu), \;\; v_j \in T_\eta
\end{gathered} \tag{7}
$$

$$
\begin{gathered}
x(u_\eta) + \Delta\, Z_\eta' \leq x(v_j) + \Delta\, Z_j''(v_j) \; + K(1 - \; BV_{\eta,j}) + K\; HH_{n,m} - \varepsilon, \\
e_n = (u_\eta, T_\eta) \in E_{H_2}, \;\; e_m = (u_\mu, T_\mu) \in E_{H_2}, \\
x(u_\eta) < x(u_\mu), \;\; v_j \in T_\mu
\end{gathered} \tag{8}
$$

$$
\begin{gathered}
x(u_\mu) + \Delta\, Z_\mu' \leq x(v_j) + \Delta\, Z_j''(v_j) \; + K(1 - \; BV_{\mu,j}) + K\; (1 - HH_{n,m}) - \varepsilon, \\
e_n = (u_\eta, T_\eta) \in E_{H_2}, \;\; e_m = (u_\mu, T_\mu) \in E_{H_2}, \\
x(u_\eta) < x(u_\mu), \;\; v_j \in T_\eta
\end{gathered} \tag{9}
$$

$$AZ_i' \geq Z_i', AZ_i' \geq -Z_i', \;\; u_i \in V_1 \tag{10}$$

$$AZ_j'' \geq Z_j'', AZ_j'' \geq -Z_j'', \;\; v_j \in V_2 \tag{11}$$

$$x(u_{i+1}) + \Delta\, Z_{i+1}' - (x(u_i) + \Delta\, Z_i') \geq \delta_w, \;\; i = 1, \dots, |V_1| - 1 \tag{12}$$

$$x\left(v_{j+1}\right) + \Delta\, Z_{j+1}'' - \left(x\left(v_j\right) + \Delta\, Z_j''\right) \geq \delta_w, \;\; j = 1, \dots, |V_2| - 1 \tag{13}$$

$$\begin{gathered} 0 \leq HH_{l,m} - HH_{l,n} + HH_{n,m} \leq 1, \\ e_l = (u_\lambda, T_\lambda) \in E_{H_2}, \;\; e_m = \left(u_\mu, T_\mu\right) \in E_{H_2}, \;\; e_n = \left(u_\eta, T_\eta\right) \in E_{H_2}, \\ x(u_\lambda) < x\left(u_\mu\right) \; < x\left(u_\eta\right) \end{gathered} \tag{14}$$

Алгоритмы минимизации числа пересечений гиперребер

Обобщенная постановка задачи минимизации пересечений гиперребер

На рис. 6 представлена укладка двухслойного гиперграфа, полученная в результате решения оптимизационной задачи (1) – (14). Были выбраны следующие значения параметров прорисовки:

$$\delta_h^{(1)} = \delta_h^{(2)} = \delta_h^{(3)} = \rho, \;\; \delta_w = 3\rho.$$

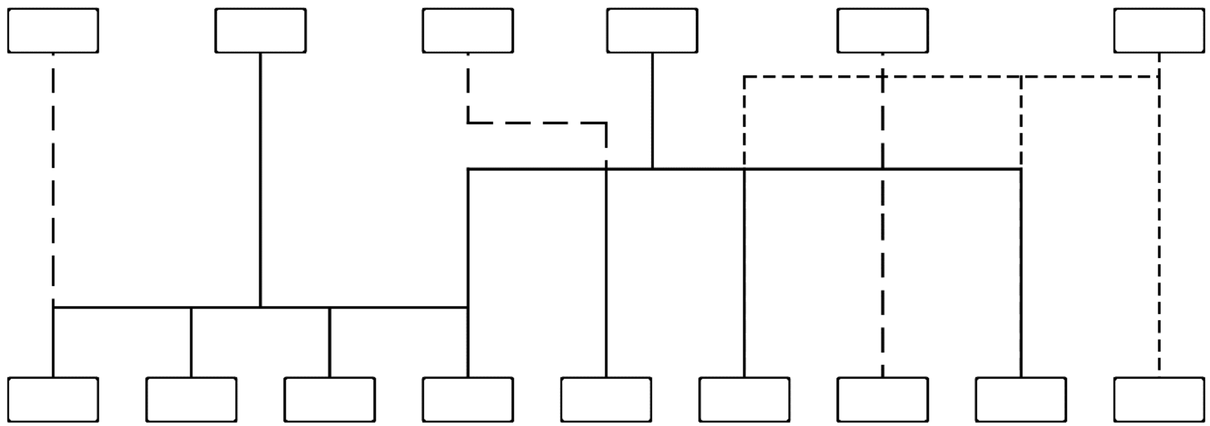

Рис. 6. Укладка двухслойного гиперграфа.

Алгоритмы минимизации числа пересечений гиперребер

Формулирование задачи минимизации пересечений гиперребер двухслойного подграфа как задачи устранения циклов в направленном графе

Одним из подходов к решению задачи минимизации числа пересечений гиперребер является ее формулировка в виде задачи устранения циклов в некотором взвешенном направленном графе. Каждая вершина этого графа соответствует горизонтальному сегменту гиперребра, и каждая пара вершин соединена двумя противонаправленными взвешенными дугами. Направление дуги показывает, какая вершина (т.е. соответствующий ей горизонтальный сегмент) расположена выше, а ее вес определяется взвешенной суммой числа пересечений и недопустимых наложений гиперребер, соответствующих вершинам.

Алгоритмы минимизации числа пересечений гиперребер

Алгоритм постобработки для устранения недопустимых наложений

Задача устранения недопустимых наложений может быть сформулирована в виде задачи целочисленного программирования, как частный случай постановки, представленной в статье, без учета переменных и ограничений, связанных с минимизацией числа пересечений гиперребер. Недопустимые наложения должны быть устранены за счет сдвигов вершин нижнего и/или верхнего слоя на определенное целое число «единичных сдвигов» длины Δ.

Введем в рассмотрение 5 видов целочисленных переменных Z', Z'', AZ', AZ'' и BV.

Переменные Z_i' и Z_j'' характеризуют количество единичных сдвигов вершин верхнего $u_i \in V_1$ и нижнего слоя $v_j \in V_2$ соответственно, а их знаки определяют направления сдвигов. Тогда новые (вследствие проведенных сдвигов) абсциссы вершин $u_i \in V_1$ и $v_j \in V_2$ будут равны

$$x'(u_i) = x(u_i) + \Delta \times Z_i',$$
$$x'(v_j) = x(v_j) + \Delta \times Z_j''.$$

Переменные AZ_i' и AZ_j'' – определяют число единичных сдвигов вершин u_i и v_j, соответственно, т.е. $AZ_i' = |Z_i'|$, а $AZ_j'' = |Z_j''|$.

Переменные $BV_{\mu,j}$ – это булевы фиктивные переменные, которые вводятся для каждого гиперребра $e_m = (u_\mu, T_\mu)$ и множества вершин $v_j \in V_2^m$, где V_2^m – объединение вершин нижнего слоя, инцидентных гиперребрам, чьи горизонтальные сегменты находятся выше горизонтального сегмента гиперребра e_m.

Алгоритмы минимизации числа пересечений гиперребер

Алгоритм постобработки для устранения недопустимых наложений

Целевая функция задачи целочисленного программирования минимизирует общее число сдвигов вершин:

$$L(Z', Z'', AZ', AZ'', BV) = \sum_{i=1}^{|V_1|} AZ'_i + \sum_{j=1}^{|V_2|} AZ''_j \rightarrow \min \qquad (15)$$

Система ограничений задачи целочисленного программирования имеет вид:

$$x(v_j) + \Delta\, Z''_j(v_j) \le x(u_\mu) + \Delta\, Z'_\mu + K\, BV_{\mu,j} - \varepsilon, \quad \forall\, e_m = (u_\mu, T_\mu),\ \ v_j \in V_2^m \qquad (16)$$

$$x(u_\mu) + \Delta\, Z'_\mu \le x(v_j) + \Delta\, Z''_j(v_j) + K(1 - BV_{\mu,j}) - \varepsilon, \quad \forall\, e_m = (u_\mu, T_\mu),\ \ v_j \in V_2^m \qquad (17)$$

$$AZ'_i \ge Z'_i,\ \ AZ'_i \ge -Z'_i,\quad \forall\, u_i \in V_1 \qquad (18)$$

$$AZ''_j \ge Z''_j,\ \ AZ''_j \ge -Z''_j,\quad \forall\, v_j \in V_2 \qquad (19)$$

$$x(u_{i+1}) + \Delta\, Z'_{i+1} - (x(u_i) + \Delta\, Z'_i) \ge \delta_w,\ \ i = 1, \dots, |V_1| - 1 \qquad (20)$$

$$x(v_{j+1}) + \Delta\, Z''_{j+1} - \left(x(v_j) + \Delta\, Z''_j\right) \ge \delta_w,\ \ j = 1, \dots, |V_2| - 1 \qquad (21)$$

Алгоритмы для сокращения числа горизонталей

Решение задачи о минимизации числа пересечений дает возможность минимизировать метрику $M1$, в результате сопоставления горизонтальному сегменту каждого гиперребра e уникальной горизонтали $h(e)$. Оптимизация метрики $M2$ (номер самой нижней занятой горизонтали) может быть выполнена за счет расположения на одной горизонтали нескольких горизонтальных сегментов, т.е. за счет сокращения числа горизонталей, при этом, однако, число пересечений гиперребер не должно увеличиться.

Алгоритмы для сокращения числа горизонталей

Последовательный подъем горизонтальных сегментов гиперребер

Простейшим вариантом алгоритма для сокращения числа горизонталей является *последовательный подъем* горизонтальных сегментов гиперребер (уменьшение номера соответствующей ему горизонтали), начиная с верхнего. При этом подъем горизонтального сегмента $sH(e_m)$ на каждую более высокую горизонталь осуществляется, только если его проекция на ось абсцисс не пересекается ни с одной из проекций горизонтальных сегментов $sH\left(e_{n_1}\right), \ldots, sH\left(e_{n_2}\right)$, уже расположенных на этой горизонтали:

$$pr_x\, sH(e_m) \cap pr_x\, sH(e_n) = \emptyset \quad \forall n \in \{n_1, \ldots, n_2\},$$

где $pr_x\, sH(e)$ – проекция горизонтального сегмента гиперребра e на ось абсцисс. Все горизонтали, на которых, в результате выполнения такого алгоритма, не осталось горизонтальных участков гиперребер, исключаются, и, соответственно, уменьшается номер самой низкой используемой горизонтали.

Алгоритмы для сокращения числа горизонталей

Обобщенный алгоритм подъема горизонтальных сегментов гиперребер

Более сложный, *обобщенный*, алгоритм также последовательно поднимает горизонтальные сегменты гиперребер, не рассматривая, однако, некоторые варианты их наложения на одной горизонтали как препятствие для подъема. На рис. 7 показаны все возможные варианты допустимого (не препятствующего подъему) расположения горизонтальных сегментов гиперребер на одной горизонтали в случае их наложения.

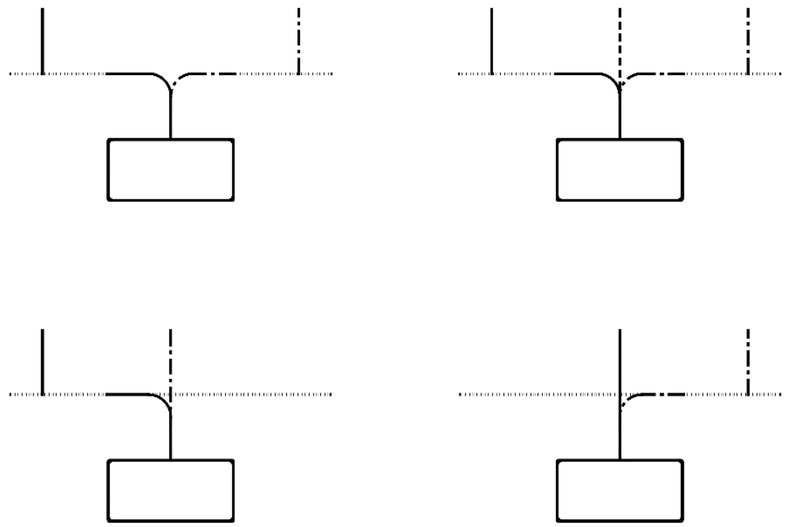

Рис 7. Возможные варианты допустимого расположения горизонтальных сегментов гиперребер на одной горизонтали в случае их наложения.

Алгоритмы для сокращения числа горизонталей

Алгоритм сбалансированное расположения горизонтальных сегментов гиперребер

Алгоритмы, оптимизирующие метрику $M2$, уменьшают число горизонталей, на которых расположены горизонтальные сегменты его гиперребер, до величины $Q \leq |E_{H_2}|$.

Минимизация метрики $M3$ осуществляется алгоритмом, который определяет сбалансированное (усредненное) положение горизонтальных сегментов гиперребер. В соответствии с ним, для каждого гиперребра e (последовательно, начиная с того, чей горизонтальный сегмент занимает горизонталь с наибольшим номером и слева направо, если горизонтальные сегменты гиперребер расположены на одной горизонтали) вычисляется самое нижнее допустимое положение его горизонтального сегмента $h_{low}(e)$, при котором не увеличивается число пересечений гиперребер. В результате, определяется интервал допустимых номеров горизонталей, и горизонтальный сегмент помещается в «среднее» положение

$$h(e) = \frac{h_{high}(e) + h_{low}(e)}{2}.$$

Алгоритмы для сокращения числа горизонталей

Алгоритм минимизации отклонения относительного удлинения укладки от общепринятых значений

Минимизация метрики $M4$ выполняется за счет решения оптимизационной задачи для двухслойного гиперграфа, в результате чего относительное удлинение (отношение высоты к ширине) его укладки приближается к некоторому элементу ar из заранее определенного списка AR общепринятых значений[4].

Целевая функция задачи линейного программирования минимизирует величину dev, обозначающую отклонение относительного удлинения укладки двухслойного гиперграфа от ar, одного из элементов списка AR:

$$F_{ar}(\delta_h^{(1)}, \delta_h^{(2)}, \delta_h^{(3)}, dev\,) = dev \to \min \qquad (22)$$

Система ограничений задачи целочисленного программирования имеет вид:

$$dev \geq \frac{\delta_h^{(1)} + (Q-1) \times \delta_h^{(2)} + \delta_h^{(3)}}{ar \times width} - 1 \qquad (23)$$

$$dev \geq -\frac{\delta_h^{(1)} + (Q-1) \times \delta_h^{(2)} + \delta_h^{(3)}}{ar \times width} + 1 \qquad (24)$$

$$\delta_{h,min}^{(i)} \leq \delta_h^{(i)} \leq \delta_{h,max}^{(i)}\,, \quad i = 1, 2, 3 \qquad (25)$$

---

[4] Taylor M., Rodgers P. Applying Graphical Design Techniques to Graph Visualisation // *Proceedings of the Ninth International Conference on Information Visualisation*, 2005, pp. 651 – 656.

Числовые расчеты

Были выбраны следующие значения параметров прорисовки:

$$\delta_{h,min}^{(1)} = \rho,\ \delta_{h,max}^{(1)} = 2.5\rho,$$

$$\delta_{h,min}^{(2)} = \rho,\ \delta_{h,max}^{(2)} = 5\rho,$$

$$\delta_{h,min}^{(3)} = \rho,\ \delta_{h,max}^{(3)} = 2.5\rho,$$

$$\Delta = 0.5\rho,\ \delta_{w} = 3\rho,$$

$$AR = \left\{1, \frac{3}{4}, \frac{8}{11}, \frac{2}{1+\sqrt{5}}, \frac{1}{1.85}, \frac{1}{1+\sqrt{2}}, \frac{1}{2.76}, \frac{1}{4}\right\}.$$

Числовые расчеты

- ТП – обобщенная точная постановка задачи минимизации пересечений гиперребер;
- ТА – точная постановка для решения задачи устранения циклов во взвешенном направленном графе, с последующим применением алгоритма постобработки для устранения недопустимых наложений;
- ЭА – жадный алгоритм для решения задачи устранения циклов во взвешенном направленном графе с последующим применением алгоритма постобработки для устранения недопустимых наложений;
- ПЭ – алгоритм последовательного подъема горизонтальных сегментов гиперребер с последующим балансированием их расположения;
- РЭ – обобщенный алгоритм подъема горизонтальных сегментов гиперребер с последующим балансированием их расположения.

Для примера применения алгоритмов при прорисовке гиперребер ниже представлены (рис. 3):

- укладка исходного графа относительного небольшого размера. При этом были решены задачи: разбиения множества вершин графа на подмножества (слои), определения последовательности вершин на слоях с целью минимизации пересечения ребер, поиска координат вершин;
- результат решения задачи в обобщенной постановке задачи минимизации пересечений гиперребер соответствующего гиперграфа;
- результат применения обобщенного алгоритма подъема горизонтальных сегментов гиперребер с последующим балансированием их расположения.

Числовые расчеты

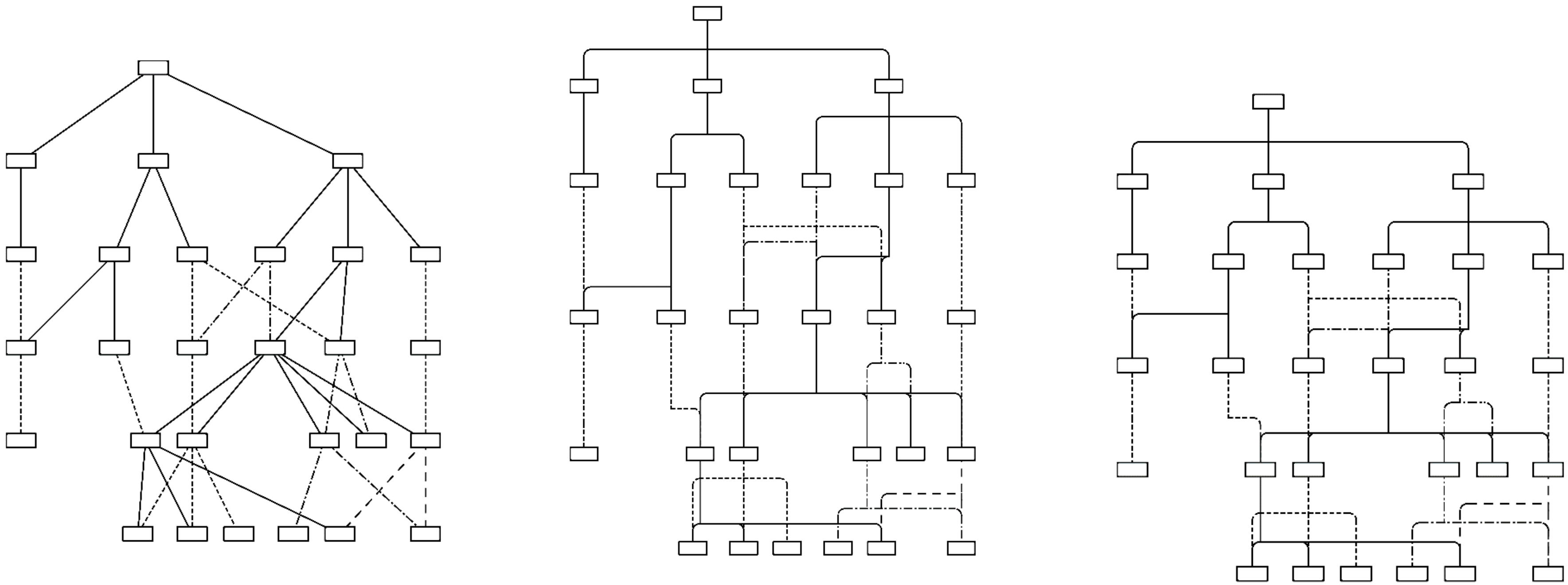

Рис 8. Укладка исходного графа; применение алгоритма ТП для соответствующего гиперграфа; применение алгоритма РЭ.

Числовые расчеты

Значение $M2$ равно сумме номеров самых нижних горизонталей каждого двухслойного подграфа, а величина $M3$ вычисляется по формуле:

$$M3 = \frac{1}{|E_H|} \sum_{e \in E_H} |2h(e) - h'_{high}(e) - h'_{low}(e)|,$$

где $h'_{high}(e)$ ($h'_{low}(e)$) – номер самой высокой (низкой) горизонтали, для которой проекция $pr_x\ sH(e)$ с выколотыми крайними точками не имеет пересечений с проекциями на ось абсцисс горизонтальных сегментов гиперребер данной горизонтали. Расчет $M4$ выполняется следующим образом:

$$M4 = \sum_{i=1}^{k-1} \min_{ar \in AR} \left| \frac{height_i}{width_i} - ar \right| / ar\ ,$$

где $height_i$ и $width_i$ – высота и ширина укладки i-го по порядку двухслойного подграфа исходного k-слойного гиперграфа.

Алгоритм	Значение метрики	ТП	ТА	ЭА
Начальный этап	$M1$	38	38	40
	$M2$	36	36	36
	$M3$	4.8	4.4	3.6
	$M4$	0.38	0.38	0.38
	τ, сек	2.2	5.8	2.7
ПЭ	$M1$	38	38	40
	$M2$	31	30	30
	$M3$	3.9	3.4	3
	$M4$	0.38	0.38	0.38
	τ, сек	0.005	0.006	0.006
РЭ	$M1$	38	38	40
	$M2$	27	26	27
	$M3$	3.5	3.1	2.9
	$M4$	0.03	0.03	0.03
	τ, сек	0.02	0.02	0.02

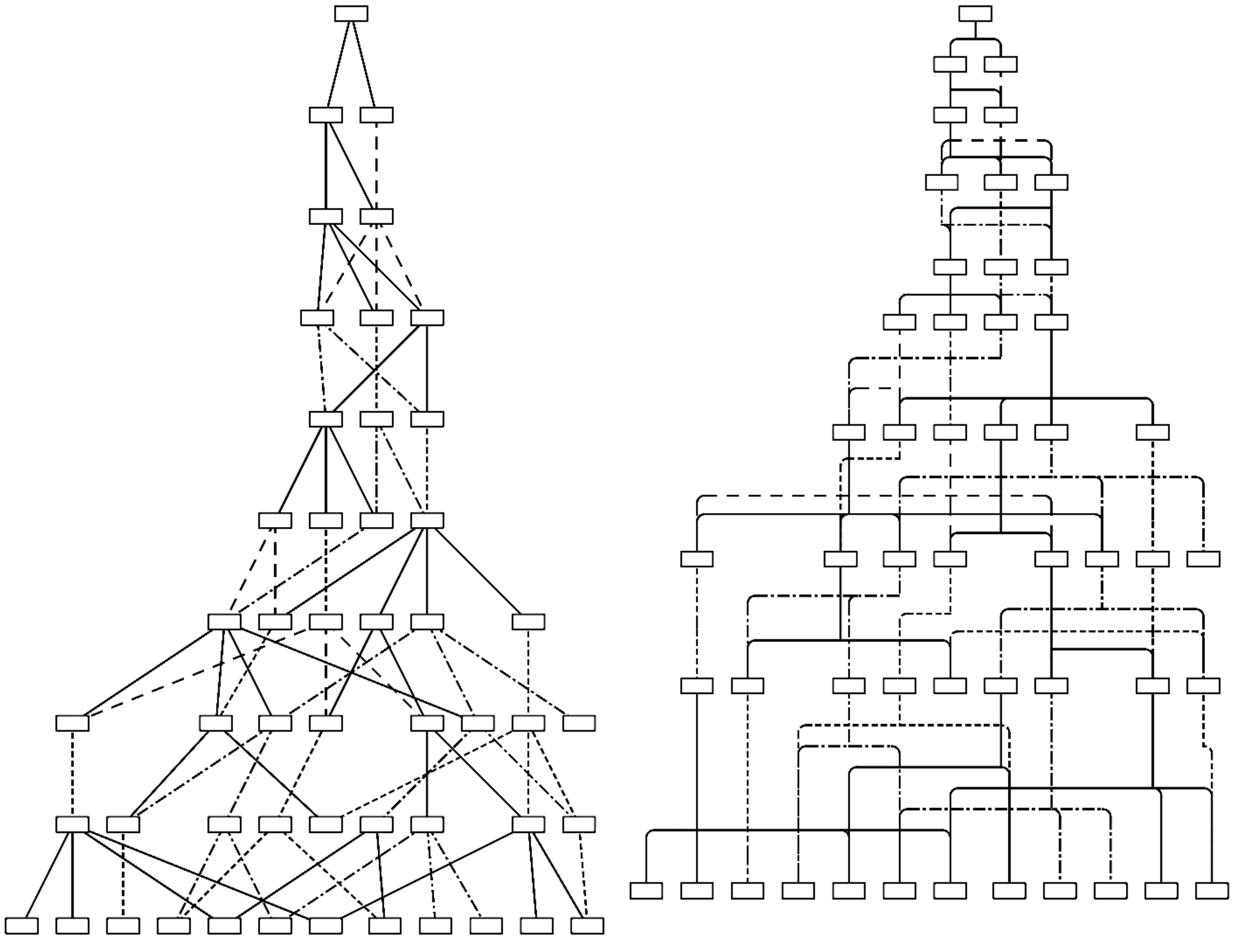

Рис 9. Укладка иерархического многослойного графа (слева), укладка соответствующего гиперграфа (справа).