

Линейная алгебра как основа для языка запросов к графам

Семён Григорьев

Санкт-Петербургский Государственный Университет

09 декабря 2025

- Доцент кафедры системного программирования Санкт-Петербургского Государственного Университета
- Руководитель исследовательской группы
- Области интересов
 - ▶ **Высокопроизводительная линейная алгебра** для анализа графов
 - ★ **Обобщённая:** матрицы и вектора параметризованы типом элемента, операции над ними могут быть заданы пользователем
 - ★ **Разреженная:** специализированные структуры для хранения матриц и векторов, специализированные алгоритмы для их обработки
 - ★ В том числе, с использованием **графических ускорителей**
 - ▶ **Высокопроизводительный анализ графов**



- Email: s.v.grigoriev@mail.spbu.ru
- GitHub: [gsvgit](#)
- Google Scholar: [Semyon Grigorev](#)
- DBLP: [Semyon V. Grigorev](#)

- **Анализ больших графов:** графовые БД, анализ кода, поиск уязвимостей, анализ трафика, анализ транзакций, банковская аналитика, социальные сети. . .
 - ▶ Важна производительность
 - ▶ Разнообразные алгоритмы

- **Анализ больших графов:** графовые БД, анализ кода, поиск уязвимостей, анализ трафика, анализ транзакций, банковская аналитика, социальные сети. . .
 - ▶ Важна производительность
 - ▶ Разнообразные алгоритмы
- Путь к унифицированной параллельной обработке графов
 - ▶ Граф \iff **матрица** смежности
 - ▶ Линейная алгебра \iff **параллелизм** по данным
 - ▶ Метки на рёбрах \iff пользовательские типы и операции над ними

- **Анализ больших графов:** графовые БД, анализ кода, поиск уязвимостей, анализ трафика, анализ транзакций, банковская аналитика, социальные сети. . .
 - ▶ Важна производительность
 - ▶ Разнообразные алгоритмы
- Путь к унифицированной параллельной обработке графов
 - ▶ Граф \iff **матрица** смежности
 - ▶ Линейная алгебра \iff **параллелизм** по данным
 - ▶ Метки на рёбрах \iff пользовательские типы и операции над ними
- **Качественные базовые блоки** (как в BLAS и вычислительной математике)
 - ▶ Оптимизированные базовые функции: умножение матриц, поэлементные операции
 - ★ Параллельность
 - ★ Низкоуровневые оптимизации: работа с кэшами, . . .
 - ★ Другие особенности различных аппаратных платформ
 - ▶ Оптимизированные структуры данных для хранения примитивов
 - ▶ Переносимость (разные реализации предоставляют одинаковый интерфейс)

- API для создания алгоритмов анализа графов на основе линейной алгебры
 - Различные операции над матрицами и векторами (разреженными)
 - Параметризация алгебраическими структурами: полукольцами, моноидами и т.д.

¹<https://graphblas.org/GraphBLAS-Pointers/>

²<https://github.com/DrTimothyAldenDavis/GraphBLAS>

³<https://github.com/GraphBLAS/LAGraph>

⁴<https://graphblas.org/>

- API для создания алгоритмов анализа графов на основе линейной алгебры
 - Различные операции над матрицами и векторами (разреженными)
 - Параметризация алгебраическими структурами: полукольцами, моноидами и т.д.
- Позволяет выражать различные алгоритмы
 - Обход в ширину, поиск кратчайших путей, достижимость, ...
 - Подсчёт треугольников, PageRank, остовные деревья, кластеризация, ...
 - Навигационные запросы: **RPQ, CFPQ**, ...

¹<https://graphblas.org/GraphBLAS-Pointers/>

²<https://github.com/DrTimothyAldenDavis/GraphBLAS>

³<https://github.com/GraphBLAS/LAGraph>

⁴<https://graphblas.org/>

- API для создания алгоритмов анализа графов на основе линейной алгебры
 - Различные операции над матрицами и векторами (разреженными)
 - Параметризация алгебраическими структурами: полукольцами, моноидами и т.д.
- Позволяет выражать различные алгоритмы
 - Обход в ширину, поиск кратчайших путей, достижимость, ...
 - Подсчёт треугольников, PageRank, остовные деревья, кластеризация, ...
 - Навигационные запросы: **RPQ**, **CFPQ**, ...
- Подробнее
 - GraphBLAS Pointers¹
 - **SuiteSparse:GraphBLAS**² — эталон на чистом C
 - **LAGraph**³ — коллекция прикладных алгоритмов анализа графов

¹<https://graphblas.org/GraphBLAS-Pointers/>

²<https://github.com/DrTimothyAldenDavis/GraphBLAS>

³<https://github.com/GraphBLAS/LAGraph>

⁴<https://graphblas.org/>

Отношения и матрицы

$$R_1 = \{(a, y); (b, x); (b, z)\}$$

$$R_2 = \{(x, u); (x, w); (y, v); (y, w); (z, u)\}$$

$$|S_A| = n, |S_B| = k, |S_C| = m$$

$$\mathcal{N}_A : [0 \dots (n-1)] \rightarrow S_A, \text{ биекция}$$

$$\mathcal{N}_B : [0 \dots (k-1)] \rightarrow S_B, \text{ биекция}$$

$$\mathcal{N}_C : [0 \dots (m-1)] \rightarrow S_C, \text{ биекция}$$

$$R_1 \subseteq S_A \times S_B, R_2 \subseteq S_B \times S_C$$

$$R_3 = \{(x, z) \mid (x, y) \in R_1, (y, z) \in R_2\}$$

$$M_{n \times k}^{R_1} = 1 \iff (\mathcal{N}_A(i), \mathcal{N}_B(j)) \in R_1, \text{ иначе } M^{R_1} = 0$$

$$M_{k \times m}^{R_2} = 1 \iff (\mathcal{N}_B(i), \mathcal{N}_C(j)) \in R_2, \text{ иначе } M^{R_2} = 0$$

$$M_{n \times m}^{R_3} = M^{R_1} \times M^{R_2}$$

Отношения и матрицы

$$R_1 = \{(a, y); (b, x); (b, z)\}$$

$$R_2 = \{(x, u); (x, w); (y, v); (y, w); (z, u)\}$$

$$R_3 = \{(i, j) \mid (i, k) \in R_1, (k, j) \in R_2\}$$

$$|S_A| = n, |S_B| = k, |S_C| = m$$

$$\mathcal{N}_A : [0 \dots (n-1)] \rightarrow S_A, \text{ биекция}$$

$$\mathcal{N}_B : [0 \dots (k-1)] \rightarrow S_B, \text{ биекция}$$

$$\mathcal{N}_C : [0 \dots (m-1)] \rightarrow S_C, \text{ биекция}$$

$$R_1 \subseteq S_A \times S_B, R_2 \subseteq S_B \times S_C$$

$$R_3 = \{(x, z) \mid (x, y) \in R_1, (y, z) \in R_2\}$$

$$M_{n \times k}^{R_1} = 1 \iff (\mathcal{N}_A(i), \mathcal{N}_B(j)) \in R_1, \text{ иначе } M^{R_1} = 0$$

$$M_{k \times m}^{R_2} = 1 \iff (\mathcal{N}_B(i), \mathcal{N}_C(j)) \in R_2, \text{ иначе } M^{R_2} = 0$$

$$M_{n \times m}^{R_3} = M^{R_1} \times M^{R_2}$$

Отношения и матрицы

$$R_1 = \{(a, y); (b, x); (b, z)\}$$

$$R_2 = \{(x, u); (x, v); (y, v); (y, w); (z, u)\}$$

$$R_3 = \{(i, j) \mid (i, k) \in R_1, (k, j) \in R_2\}$$

$$\begin{matrix} & \overset{x}{\color{red}} & \overset{y}{\color{red}} & \overset{z}{\color{red}} \\ \overset{a}{\color{blue}} & \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} & \times & \begin{matrix} & \overset{u}{\color{green}} & \overset{v}{\color{green}} & \overset{w}{\color{green}} \\ \overset{x}{\color{red}} & \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{pmatrix} & \overset{y}{\color{red}} & \end{matrix} = \begin{matrix} & \overset{u}{\color{green}} & \overset{v}{\color{green}} & \overset{w}{\color{green}} \\ \overset{a}{\color{blue}} & \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix} & \overset{b}{\color{blue}} & \end{matrix} \end{matrix}$$

$$|S_A| = n, |S_B| = k, |S_C| = m$$

$$\mathcal{N}_A : [0 \dots (n-1)] \rightarrow S_A, \text{ биекция}$$

$$\mathcal{N}_B : [0 \dots (k-1)] \rightarrow S_B, \text{ биекция}$$

$$\mathcal{N}_C : [0 \dots (m-1)] \rightarrow S_C, \text{ биекция}$$

$$R_1 \subseteq S_A \times S_B, R_2 \subseteq S_B \times S_C$$

$$R_3 = \{(x, z) \mid (x, y) \in R_1, (y, z) \in R_2\}$$

$$M_{n \times k}^{R_1} = 1 \iff (\mathcal{N}_A(i), \mathcal{N}_B(j)) \in R_1, \text{ иначе } M^{R_1} = 0$$

$$M_{k \times m}^{R_2} = 1 \iff (\mathcal{N}_B(i), \mathcal{N}_C(j)) \in R_2, \text{ иначе } M^{R_2} = 0$$

$$M_{n \times m}^{R_3} = M^{R_1} \times M^{R_2}$$

Отношения и матрицы

$$R_1 = \{(a, y); (b, x); (b, z)\}$$

$$R_2 = \{(x, u); (x, w); (y, v); (y, w); (z, u)\}$$

$$R_3 = \{(i, j) \mid (i, k) \in R_1, (k, j) \in R_2\}$$

$$\begin{matrix} x & y & z \\ a & \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \end{matrix} \times \begin{matrix} u & v & w \\ x & \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{pmatrix} \end{matrix} = \begin{matrix} u & v & w \\ a & \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix} \end{matrix}$$

a

x

u

b

y

v

z

w

$$|S_A| = n, |S_B| = k, |S_C| = m$$

$$\mathcal{N}_A : [0 \dots (n-1)] \rightarrow S_A, \text{ биекция}$$

$$\mathcal{N}_B : [0 \dots (k-1)] \rightarrow S_B, \text{ биекция}$$

$$\mathcal{N}_C : [0 \dots (m-1)] \rightarrow S_C, \text{ биекция}$$

$$R_1 \subseteq S_A \times S_B, R_2 \subseteq S_B \times S_C$$

$$R_3 = \{(x, z) \mid (x, y) \in R_1, (y, z) \in R_2\}$$

$$M_{n \times k}^{R_1} = 1 \iff (\mathcal{N}_A(i), \mathcal{N}_B(j)) \in R_1, \text{ иначе } M^{R_1} = 0$$

$$M_{k \times m}^{R_2} = 1 \iff (\mathcal{N}_B(i), \mathcal{N}_C(j)) \in R_2, \text{ иначе } M^{R_2} = 0$$

$$M_{n \times m}^{R_3} = M^{R_1} \times M^{R_2}$$

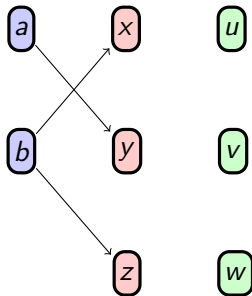
Отношения и матрицы

$$R_1 = \{(a, y); (b, x); (b, z)\}$$

$$R_2 = \{(x, u); (x, w); (y, v); (y, w); (z, u)\}$$

$$R_3 = \{(i, j) \mid (i, k) \in R_1, (k, j) \in R_2\}$$

$$\begin{matrix} x & y & z \\ a & \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \end{matrix} \times \begin{matrix} u & v & w \\ x & \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{pmatrix} \end{matrix} = \begin{matrix} u & v & w \\ a & \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix} \end{matrix}$$



$$|S_A| = n, |S_B| = k, |S_C| = m$$

$$\mathcal{N}_A : [0 \dots (n-1)] \rightarrow S_A, \text{ биекция}$$

$$\mathcal{N}_B : [0 \dots (k-1)] \rightarrow S_B, \text{ биекция}$$

$$\mathcal{N}_C : [0 \dots (m-1)] \rightarrow S_C, \text{ биекция}$$

$$R_1 \subseteq S_A \times S_B, R_2 \subseteq S_B \times S_C$$

$$R_3 = \{(x, z) \mid (x, y) \in R_1, (y, z) \in R_2\}$$

$$M_{n \times k}^{R_1} = 1 \iff (\mathcal{N}_A(i), \mathcal{N}_B(j)) \in R_1, \text{ иначе } M^{R_1} = 0$$

$$M_{k \times m}^{R_2} = 1 \iff (\mathcal{N}_B(i), \mathcal{N}_C(j)) \in R_2, \text{ иначе } M^{R_2} = 0$$

$$M_{n \times m}^{R_3} = M^{R_1} \times M^{R_2}$$

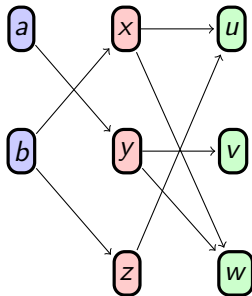
Отношения и матрицы

$$R_1 = \{(a, y); (b, x); (b, z)\}$$

$$R_2 = \{(x, u); (x, w); (y, v); (y, w); (z, u)\}$$

$$R_3 = \{(i, j) \mid (i, k) \in R_1, (k, j) \in R_2\}$$

$$\begin{matrix} & \begin{matrix} x & y & z \end{matrix} \\ \begin{matrix} a \\ b \end{matrix} & \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \end{matrix} \times \begin{matrix} & \begin{matrix} u & v & w \end{matrix} \\ \begin{matrix} x \\ y \\ z \end{matrix} & \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{pmatrix} \end{matrix} = \begin{matrix} & \begin{matrix} u & v & w \end{matrix} \\ \begin{matrix} a \\ b \end{matrix} & \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix} \end{matrix}$$



$$|S_A| = n, |S_B| = k, |S_C| = m$$

$$\mathcal{N}_A : [0 \dots (n-1)] \rightarrow S_A, \text{ биекция}$$

$$\mathcal{N}_B : [0 \dots (k-1)] \rightarrow S_B, \text{ биекция}$$

$$\mathcal{N}_C : [0 \dots (m-1)] \rightarrow S_C, \text{ биекция}$$

$$R_1 \subseteq S_A \times S_B, R_2 \subseteq S_B \times S_C$$

$$R_3 = \{(x, z) \mid (x, y) \in R_1, (y, z) \in R_2\}$$

$$M_{n \times k}^{R_1} = 1 \iff (\mathcal{N}_A(i), \mathcal{N}_B(j)) \in R_1, \text{ иначе } M^{R_1} = 0$$

$$M_{k \times m}^{R_2} = 1 \iff (\mathcal{N}_B(i), \mathcal{N}_C(j)) \in R_2, \text{ иначе } M^{R_2} = 0$$

$$M_{n \times m}^{R_3} = M^{R_1} \times M^{R_2}$$

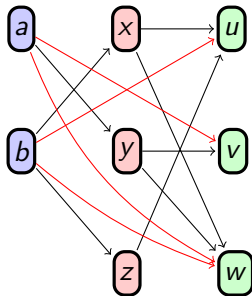
Отношения и матрицы

$$R_1 = \{(a, y); (b, x); (b, z)\}$$

$$R_2 = \{(x, u); (x, w); (y, v); (y, w); (z, u)\}$$

$$R_3 = \{(i, j) \mid (i, k) \in R_1, (k, j) \in R_2\}$$

$$\begin{matrix} & \begin{matrix} x & y & z \end{matrix} \\ \begin{matrix} a \\ b \end{matrix} & \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \end{matrix} \times \begin{matrix} & \begin{matrix} u & v & w \end{matrix} \\ \begin{matrix} x \\ y \\ z \end{matrix} & \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{pmatrix} \end{matrix} = \begin{matrix} & \begin{matrix} u & v & w \end{matrix} \\ \begin{matrix} a \\ b \end{matrix} & \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix} \end{matrix}$$



$$|S_A| = n, |S_B| = k, |S_C| = m$$

$$\mathcal{N}_A : [0 \dots (n-1)] \rightarrow S_A, \text{ биекция}$$

$$\mathcal{N}_B : [0 \dots (k-1)] \rightarrow S_B, \text{ биекция}$$

$$\mathcal{N}_C : [0 \dots (m-1)] \rightarrow S_C, \text{ биекция}$$

$$R_1 \subseteq S_A \times S_B, R_2 \subseteq S_B \times S_C$$

$$R_3 = \{(x, z) \mid (x, y) \in R_1, (y, z) \in R_2\}$$

$$M_{n \times k}^{R_1} = 1 \iff (\mathcal{N}_A(i), \mathcal{N}_B(j)) \in R_1, \text{ иначе } M^{R_1} = 0$$

$$M_{k \times m}^{R_2} = 1 \iff (\mathcal{N}_B(i), \mathcal{N}_C(j)) \in R_2, \text{ иначе } M^{R_2} = 0$$

$$M_{n \times m}^{R_3} = M^{R_1} \times M^{R_2}$$

Умножение матриц и «полукольца»

$$C_{n \times n} = A_{n \times n} \times B_{n \times n}$$

$$C[i, j] = \sum_{k=0}^{n-1} A[i, k] * B[k, j]$$

$$\begin{matrix} i & \begin{pmatrix} \dots & \dots & \dots & \dots & \dots \\ a_{i0} & \dots & \dots & \dots & a_{i(k-1)} \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \end{pmatrix} & i \end{matrix} \times \begin{matrix} \begin{pmatrix} \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \end{pmatrix} & \begin{matrix} j \\ b_{0j} \\ \vdots \\ b_{(k-1)j} \\ j \end{matrix} \end{matrix} = \begin{matrix} \begin{pmatrix} \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \end{pmatrix} & \begin{matrix} j \\ c_{ij} \\ \vdots \\ c_{ij} \\ j \end{matrix} \end{matrix}$$

Умножение матриц и «полукольца»

$$C_{n \times n} = A_{n \times n} \times B_{n \times n}$$

$$C[i, j] = \sum_{k=0}^{n-1} A[i, k] * B[k, j]$$

$$A : \text{Matrix}\langle T_1 \rangle$$

$$B : \text{Matrix}\langle T_2 \rangle$$

$$C : \text{Matrix}\langle T_3 \rangle$$

$$\begin{matrix} i & \left(\begin{array}{cccc} \dots & \dots & \dots & \dots \\ a_{i0} & \dots & \dots & a_{i(k-1)} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \end{array} \right) & i & \times & \begin{pmatrix} \vdots & \vdots & \overset{j}{b_{0j}} & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & b_{(k-1)j} & \vdots \\ \vdots & \vdots & \underset{j}{\vdots} & \vdots \end{pmatrix} & = & \begin{matrix} & \overset{j}{\vdots} \\ i & \left(\dots & c_{ij} & \dots \right) & i \\ & \underset{j}{\vdots} \end{matrix} \end{matrix}$$

Умножение матриц и «полукольца»

$$C_{n \times n} = A_{n \times n} \times B_{n \times n}$$

$$C[i, j] = \sum_{k=0}^{n-1} A[i, k] * B[k, j]$$

$$A : \text{Matrix}\langle T_1 \rangle$$

$$B : \text{Matrix}\langle T_2 \rangle$$

$$C : \text{Matrix}\langle T_3 \rangle$$

$$\otimes : T_1 \times T_2 \rightarrow T_3$$

$$\oplus : T_3 \times T_3 \rightarrow T_3$$

$$\mathbb{0} : T_3$$

$$\begin{matrix} & & j \\ & & b_{0j} \\ & & \vdots \\ i & \begin{pmatrix} \dots & \dots & \dots \\ a_{i0} & \dots & a_{i(k-1)} \\ \dots & \dots & \dots \\ \dots & \dots & \dots \end{pmatrix} & \times & \begin{pmatrix} \vdots \\ \vdots \\ \vdots \\ b_{(k-1)j} \\ \vdots \end{pmatrix} & = & \begin{pmatrix} \vdots \\ \vdots \\ \vdots \\ c_{ij} \\ \vdots \end{pmatrix} \\ & & j \end{matrix}$$

Умножение матриц и «полукольца»

$$C_{n \times n} = A_{n \times n} \times B_{n \times n}$$

$$C[i, j] = \sum_{k=0}^{n-1} A[i, k] * B[k, j]$$

$$A : \text{Matrix}\langle T_1 \rangle$$

$$B : \text{Matrix}\langle T_2 \rangle$$

$$C : \text{Matrix}\langle T_3 \rangle$$

$$\otimes : T_1 \times T_2 \rightarrow T_3$$

$$\oplus : T_3 \times T_3 \rightarrow T_3$$

$$\mathbb{0} : T_3$$

$$C[i, j] = \bigoplus_{k=0}^{n-1} A[i, k] \otimes B[k, j]$$

$$\begin{matrix} & & & j \\ & & & \vdots \\ & & & b_{0j} \\ & & & \vdots \\ & & & b_{(k-1)j} \\ & & & j \\ i & \left(\begin{array}{cccc} \dots & \dots & \dots & \dots \\ a_{i0} & \dots & \dots & a_{i(k-1)} \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \end{array} \right) & \times & \left(\begin{array}{c} \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \end{array} \right) & = & \left(\begin{array}{c} \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \end{array} \right) \\ & & & j \end{matrix}$$

Умножение матриц и «полукольца»

$$C_{n \times n} = A_{n \times n} \times B_{n \times n}$$

$$C[i, j] = \sum_{k=0}^{n-1} A[i, k] * B[k, j]$$

$$A : \text{Matrix}\langle T_1 \rangle$$

$$B : \text{Matrix}\langle T_2 \rangle$$

$$C : \text{Matrix}\langle T_3 \rangle$$

$$\otimes : T_1 \times T_2 \rightarrow T_3$$

$$\oplus : T_3 \times T_3 \rightarrow T_3$$

$$\mathbb{0} : T_3$$

$$C[i, j] = \bigoplus_{k=0}^{n-1} A[i, k] \otimes B[k, j]$$

$$\otimes : (\text{int} \times \text{int} \times T_1) \times \underbrace{(\text{int} \times \text{int} \times T_2)}_{\text{Координаты обрабатываемого элемента}} \rightarrow T_3$$

$$\begin{matrix} & & j \\ i & \left(\begin{array}{cccc} \dots & \dots & \dots & \dots \\ a_{i0} & \dots & a_{i(k-1)} & \dots \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \end{array} \right) & \times & \begin{pmatrix} \vdots & \vdots & \overset{j}{b_{0j}} & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \underset{j}{b_{(k-1)j}} & \vdots \end{pmatrix} & = & \begin{matrix} j \\ i \left(\begin{array}{c} \vdots \\ \dots \\ c_{ij} \\ \dots \\ \vdots \end{array} \right) i \\ j \end{matrix} \end{matrix}$$

Умножение матриц и «полукольца»

$$C_{n \times n} = A_{n \times n} \times B_{n \times n}$$

$$C[i, j] = \sum_{k=0}^{n-1} A[i, k] * B[k, j]$$

$$A : \text{Matrix}\langle T_1 \rangle$$

$$B : \text{Matrix}\langle T_2 \rangle$$

$$C : \text{Matrix}\langle T_3 \rangle$$

$$\otimes : T_1 \times T_2 \rightarrow T_3$$

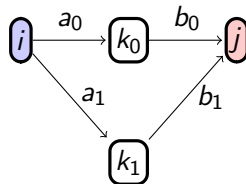
$$\oplus : T_3 \times T_3 \rightarrow T_3$$

$$\mathbb{0} : T_3$$

$$C[i, j] = \bigoplus_{k=0}^{n-1} A[i, k] \otimes B[k, j]$$

$$\otimes : (\text{int} \times \text{int} \times T_1) \times (\underbrace{\text{int} \times \text{int} \times T_2}_{\text{Координаты обрабатываемого элемента}}) \rightarrow T_3$$

$$\begin{matrix} i & & j \\ \left(\begin{array}{cccc} \dots & \dots & \dots & \dots \\ a_{i0} & \dots & a_{i(k-1)} & \dots \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \end{array} \right) & \times & \begin{matrix} j \\ \left(\begin{array}{c} \vdots \\ b_{0j} \\ \vdots \\ b_{(k-1)j} \\ \vdots \end{array} \right) \\ j \end{matrix} & = & \begin{matrix} j \\ \left(\begin{array}{c} \vdots \\ \dots \\ c_{ij} \\ \dots \\ \vdots \end{array} \right) \\ j \end{matrix} \end{matrix}$$



Умножение матриц и «полукольца»

$$C_{n \times n} = A_{n \times n} \times B_{n \times n}$$

$$C[i, j] = \sum_{k=0}^{n-1} A[i, k] * B[k, j]$$

$$A : \text{Matrix}\langle T_1 \rangle$$

$$B : \text{Matrix}\langle T_2 \rangle$$

$$C : \text{Matrix}\langle T_3 \rangle$$

$$\otimes : T_1 \times T_2 \rightarrow T_3$$

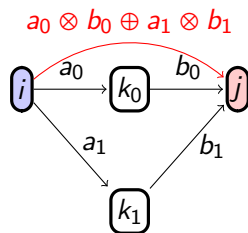
$$\oplus : T_3 \times T_3 \rightarrow T_3$$

$$0 : T_3$$

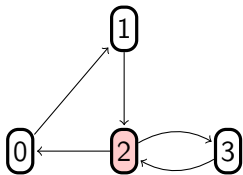
$$C[i, j] = \bigoplus_{k=0}^{n-1} A[i, k] \otimes B[k, j]$$

$$\otimes : (\text{int} \times \text{int} \times T_1) \times (\underbrace{\text{int} \times \text{int} \times T_2}_{\text{Координаты обрабатываемого элемента}}) \rightarrow T_3$$

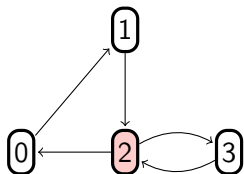
$$\begin{matrix} & & j \\ i & \left(\begin{array}{cccc} \dots & \dots & \dots & \dots \\ a_{i0} & \dots & a_{i(k-1)} & \dots \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \end{array} \right) & \times & \begin{pmatrix} \vdots & \vdots & b_{0j} & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & b_{(k-1)j} & \vdots \\ \vdots & \vdots & \vdots & \vdots \end{pmatrix} & = & \begin{matrix} j \\ i \left(\begin{array}{c} \vdots \\ \dots c_{ij} \dots \\ \vdots \end{array} \right) i \\ j \end{matrix} \end{matrix}$$



Обход в ширину



Обход в ширину



Текущий фронт

Матрица смежности

Новый фронт

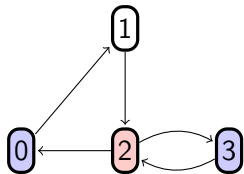
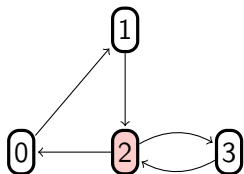
Полукольцо

Посещённые вершины

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

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

Обход в ширину



Текущий фронт

Матрица смежности

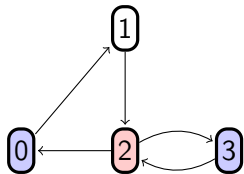
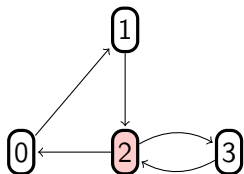
Новый фронт

Полукольцо

Посещённые вершины

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

Обход в ширину



Текущий фронт

Матрица смежности

Новый фронт

Полукольцо

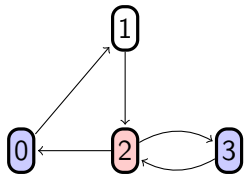
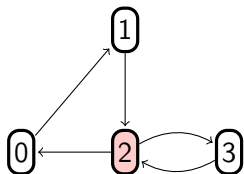
$$\begin{pmatrix} 0 & 0 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 1 \end{pmatrix}$$

Посещённые вершины

$$\begin{pmatrix} 1 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & \cancel{1} & 0 \end{pmatrix}$$

Graph diagram showing the state of the breadth-first search after the second iteration. The current front (Текущий фронт) is {2}, which is highlighted in pink. The new front (Новый фронт) is {1, 3}, which are highlighted in pink. The adjacency matrix (Матрица смежности) is shown as a 4x4 matrix with rows [0, 1, 0, 0], [0, 0, 1, 0], [1, 0, 0, 1], and [0, 0, 1, 0]. The current front is represented by the vector (1, 0, 0, 1).

Обход в ширину



Текущий фронт

Матрица смежности

Новый фронт

Полукольцо

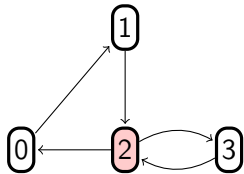
$$\begin{pmatrix} 0 & 0 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 1 \end{pmatrix}$$

Посещённые вершины

$$\begin{pmatrix} 1 & 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & \cancel{1} & 0 \end{pmatrix}$$

result: $\begin{pmatrix} 1 & 1 & 1 & 1 \end{pmatrix}$

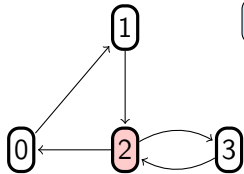
Обход в ширину с построением дерева обхода



Обход в ширину с построением дерева обхода

$p \otimes (i, j, _) = i$: получаем предка

$i_1 \oplus i_2 = i_1$: из нескольких возможных предков возьмём первого



У стартовой нет предка

(_ _ -1 _)

«Нейтральный
элемент»

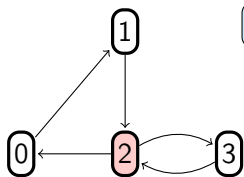
$$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 2 & - & - & 2 \\ - & - & -1 & - \end{pmatrix}$$

Стартовая — предок

Обход в ширину с построением дерева обхода

$p \otimes (i, j, _) = i$: получаем предка

$i_1 \oplus i_2 = i_1$: из нескольких возможных предков возьмём первого

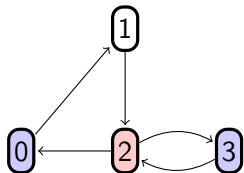


У стартовой нет предка

$$\begin{pmatrix} _ & _ & -1 & _ \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 2 & _ & _ & 2 \end{pmatrix}$$

Стартовая — предок

«Нейтральный элемент»

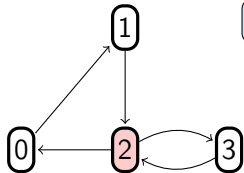


$$\begin{pmatrix} 2 & _ & _ & 2 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} _ & 0 & 3 & _ \\ 2 & 0 & -1 & 2 \end{pmatrix}$$

Обход в ширину с построением дерева обхода

$p \otimes (i, j, _) = i$: получаем предка

$i_1 \oplus i_2 = i_1$: из нескольких возможных предков возьмём первого

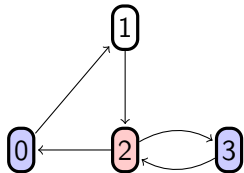


У стартовой нет предка

$$\begin{pmatrix} _ & _ & -1 & _ \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 2 & _ & _ & 2 \end{pmatrix}$$

Стартовая — предок

«Нейтральный элемент»

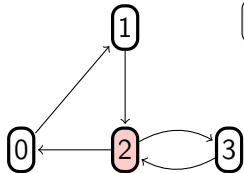


$$\begin{pmatrix} 2 & _ & _ & 2 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} _ & 0 & -1 & _ \end{pmatrix}$$

Обход в ширину с построением дерева обхода

$p \otimes (i, j, _) = i$: получаем предка

$i_1 \oplus i_2 = i_1$: из нескольких возможных предков возьмём первого

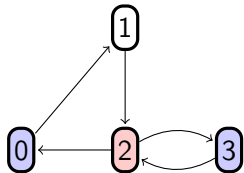


У стартовой нет предка

$$\begin{pmatrix} _ & _ & -1 & _ \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 2 & _ & _ & 2 \end{pmatrix}$$

Стартовая — предок

«Нейтральный элемент»



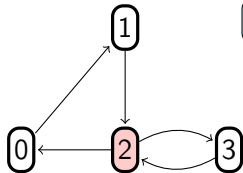
$$\begin{pmatrix} 2 & _ & _ & 2 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} _ & 0 & -1 & _ \end{pmatrix}$$

result: $\begin{pmatrix} 2 & 0 & -1 & 2 \end{pmatrix}$

Обход в ширину с построением дерева обхода

$p \otimes (i, j, _) = i$: получаем предка

$i_1 \oplus i_2 = i_1$: из нескольких возможных предков возьмём первого

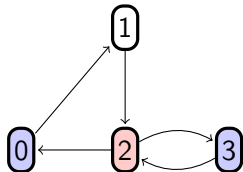


У стартовой нет предка

$$\left(\begin{array}{cccc} _ & _ & -1 & _ \end{array} \right) \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \left(\begin{array}{cccc} 2 & _ & _ & 2 \end{array} \right)$$

Стартовая — предок

«Нейтральный элемент»



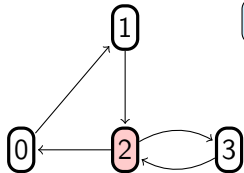
$$\left(\begin{array}{cccc} 2 & _ & _ & 2 \end{array} \right) \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \left(\begin{array}{cccc} _ & 0 & \cancel{3} & _ \end{array} \right)$$

result: $\left(\begin{array}{cccc} 2 & 0 & -1 & 2 \end{array} \right)$

Обход в ширину с построением дерева обхода

$p \otimes (i, j, _) = i$: получаем предка

$i_1 \oplus i_2 = i_1$: из нескольких возможных предков возьмём первого



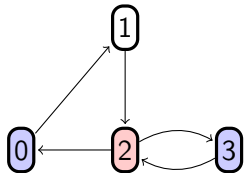
У стартовой нет предка

(_ _ -1 _)

«Нейтральный элемент»

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

Стартовая — предок

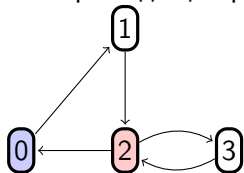


$$\begin{pmatrix} 2 & - & - & 2 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} - & 0 & - & - \\ 2 & 0 & -1 & 2 \end{pmatrix}$$

result: (2 0 -1 2)

Выбор входящих и исходящих (инцидентных) рёбер

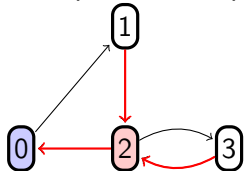
Выбор входящих рёбер — умножение на диагональную матрицу справа



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

Выбор входящих и исходящих (инцидентных) рёбер

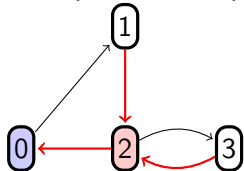
Выбор входящих рёбер — умножение на диагональную матрицу справа



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

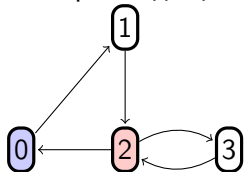
Выбор входящих и исходящих (инцидентных) рёбер

Выбор входящих рёбер — умножение на диагональную матрицу справа



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

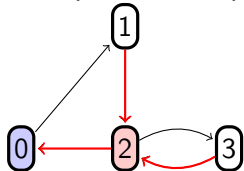
Выбор исходящих рёбер — умножение на диагональную матрицу слева



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

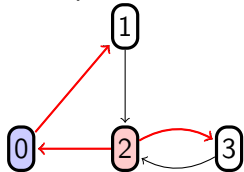
Выбор входящих и исходящих (инцидентных) рёбер

Выбор входящих рёбер — умножение на диагональную матрицу справа



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

Выбор исходящих рёбер — умножение на диагональную матрицу слева



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

Пример⁵

- Граф с двумя типами вершин: пользователи и карты
 - ▶ «Пользователь»: пол, возраст
 - ▶ «Карта»: тип (МИП, VISA, MASTERCARD), лимит средств

⁵Код и описание: <https://github.com/SparseLinearAlgebra/PageRankBenchmark>

Пример⁵

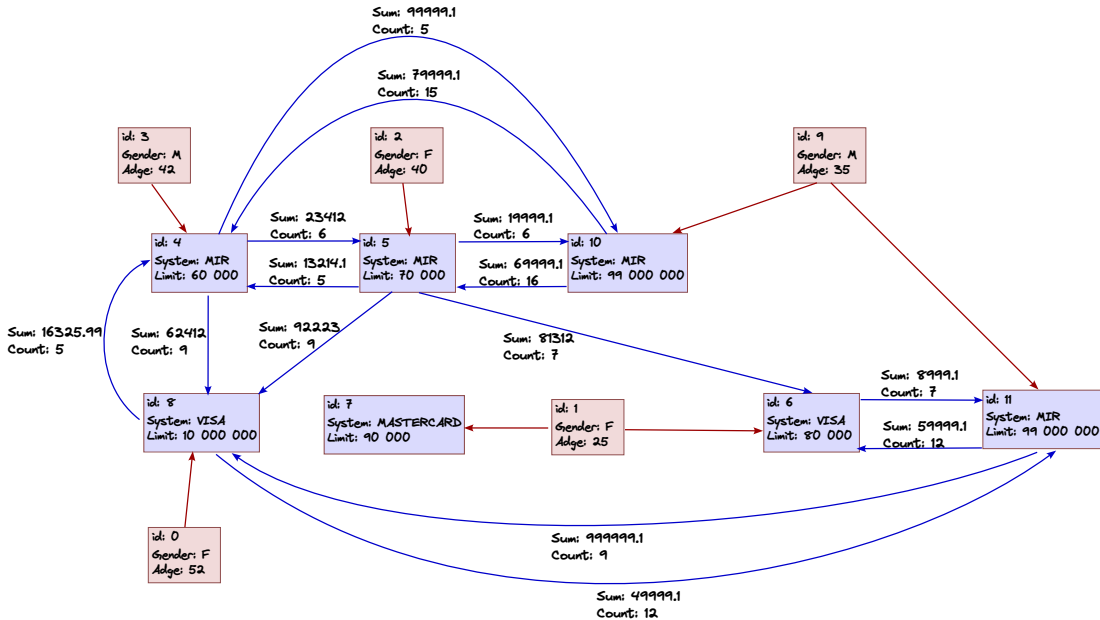
- Граф с двумя типами вершин: пользователи и карты
 - ▶ «Пользователь»: пол, возраст
 - ▶ «Карта»: тип (МИП, VISA, MASTERCARD), лимит средств
- Два типа ориентированных рёбер
 - ▶ «Перевод»: соединяет две карты (откуда и куда перевод)
 - ★ Метка: общая сумма и «количество транзакций»
 - ▶ «Владеет»: соединяет пользователя и карту (от владельца карты к карте)
 - ★ Не имеет меток

⁵Код и описание: <https://github.com/SparseLinearAlgebra/PageRankBenchmark>

Пример⁵

- Граф с двумя типами вершин: пользователи и карты
 - ▶ «Пользователь»: пол, возраст
 - ▶ «Карта»: тип (МИР, VISA, MASTERCARD), лимит средств
 - Два типа ориентированных рёбер
 - ▶ «Перевод»: соединяет две карты (откуда и куда перевод)
 - ★ Метка: общая сумма и «количество транзакций»
 - ▶ «Владеет»: соединяет пользователя и карту (от владельца карты к карте)
 - ★ Не имеет меток
- 1 Выбрать хотим все карты системы «МИР», которыми владеют люди старше заданного возраста
 - 2 Посчитать PageRank на подграфе, заданном переводами между отобранными картами

⁵Код и описание: <https://github.com/SparseLinearAlgebra/PageRankBenchmark>



- Пользовательские типы (структуры) для описания пользователя, карты, информации о переводах: `User`, `Card`, `Trans`

Представление графа

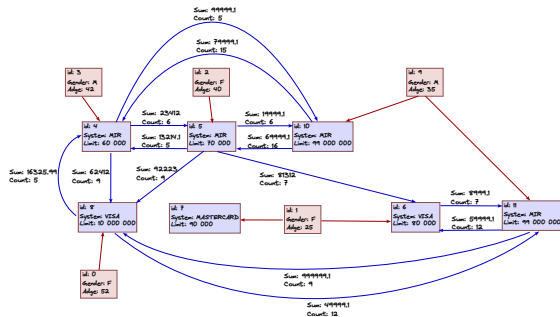
- Пользовательские типы (структуры) для описания пользователя, карты, информации о переводах: User, Card, Trans
- Булева матрица $\text{Own_Edges}_{12 \times 12}$ задаёт отношение «Владеет»

- Пользовательские типы (структуры) для описания пользователя, карты, информации о переводах: User, Card, Trans
- Булева матрица $\text{Own_Edges}_{12 \times 12}$ задаёт отношение «Владеет»
- Матрица $\text{Trans_Edges}_{12 \times 12}$ типа $\text{Matrix}\langle \text{Trans} \rangle$ задаёт отношение «Перевод»

- Пользовательские типы (структуры) для описания пользователя, карты, информации о переводах: User, Card, Trans
- Булева матрица $\text{Own_Edges}_{12 \times 12}$ задаёт отношение «Владеет»
- Матрица $\text{Trans_Edges}_{12 \times 12}$ типа $\text{Matrix}\langle \text{Trans} \rangle$ задаёт отношение «Перевод»
- Вектор пользователей Users_{12} хранит данные о пользователях

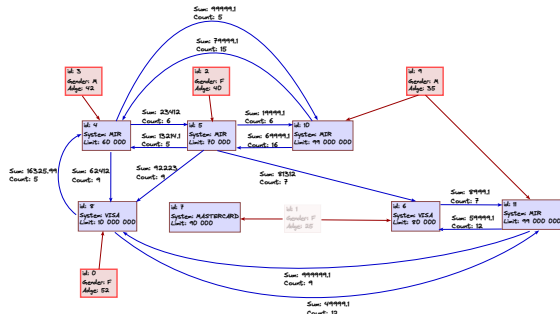
- Пользовательские типы (структуры) для описания пользователя, карты, информации о переводах: User, Card, Trans
- Булева матрица $\text{Own_Edges}_{12 \times 12}$ задаёт отношение «Владеет»
- Матрица $\text{Trans_Edges}_{12 \times 12}$ типа $\text{Matrix}\langle \text{Trans} \rangle$ задаёт отношение «Перевод»
- Вектор пользователей Users_{12} хранит данные о пользователях
- Вектор карт Cards_{12} хранит данные о картах

Построение подграфа



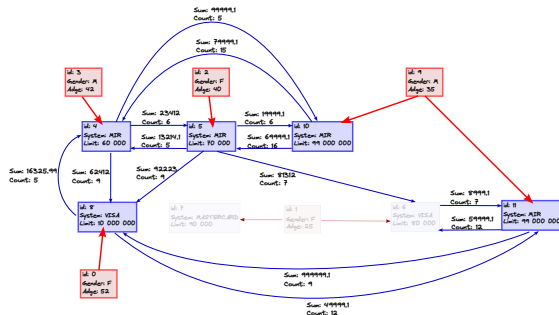
Построение подграфа

- Выбираем пользователей старше 30 лет
 - `s_users = Choose(filter, Users)`
 - `filter` — пользовательский предикат



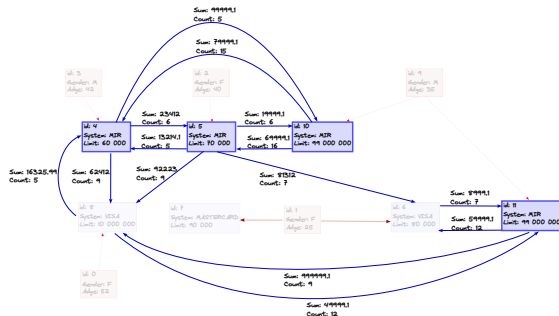
Построение подграфа

- Выбираем пользователей старше 30 лет
 - $s_users = \text{Choose}(\text{filter}, \text{Users})$
 - filter — пользовательский предикат
- Выбираем карты пользователей
 - Один шаг обхода в ширину
 - $s_cards = s_users * \text{Owns_Edges}$
 - s_cards — не карты, а их индексы



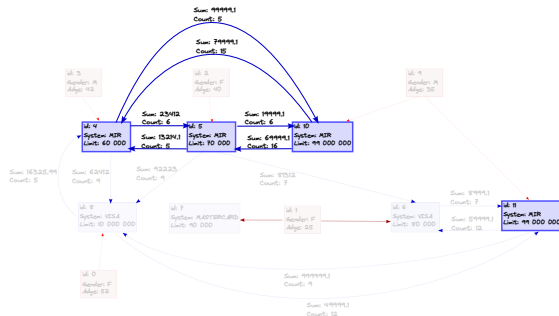
Построение подграфа

- Выбираем пользователей старше 30 лет
 - `s_users = Choose(filter, Users)`
 - `filter` — пользовательский предикат
- Выбираем карты пользователей
 - Один шаг обхода в ширину
 - `s_cards = s_users * Owns_Edges`
 - `s_cards` — не карты, а их индексы
- Выбираем карты «МИР»
 - `s_cards` используем как маску
 - `s_cards' = Mask(s_cards, Choose(filter', Cards))`



Построение подграфа

- Выбираем пользователей старше 30 лет
 - ▶ `s_users = Choose(filter, Users)`
 - ▶ `filter` — пользовательский предикат
- Выбираем карты пользователей
 - ▶ Один шаг обхода в ширину
 - ▶ `s_cards = s_users * Owns_Edges`
 - ▶ `s_cards` — не карты, а их индексы



- Выбираем карты «МИР»
 - ▶ `s_cards` используем как маску
 - ▶ `s_cards' = Mask(s_cards, Choose(filter', Cards))`
- Выбираем переводы между отобранными картами
 - ▶ Рёбра, идущие **только** между выбранными картами
 - ▶ `cards_subgraph = Diag(s_cards') * Trans_Edges * Diag(s_cards')`

Получение матрицы весов по данным о переводах

- Преобразовываем структуры (метки рёбер) в веса: $f(w) = \frac{w.Sum}{w.Count * 1000}$

Получение матрицы весов по данным о переводах

- Преобразовываем структуры (метки рёбер) в веса: $f(w) = \frac{w.Sum}{w.Count * 1000}$
- Сумма весов исходящих рёбер равна 1: $SoftMax(\vec{z})_i = \frac{e^{f(z_i)}}{\sum_{k=1}^K e^{f(z_k)}}$

Получение матрицы весов по данным о переводах

- Преобразовываем структуры (метки рёбер) в веса: $f(w) = \frac{w.Sum}{w.Count * 1000}$
- Сумма весов исходящих рёбер равна 1: $SoftMax(\vec{z})_i = \frac{e^{f(z_i)}}{\sum_{k=1}^K e^{f(z_k)}}$
- Знаменатель: $W = ReduceRows(+, Map(f, cards_subgraph))$

Получение матрицы весов по данным о переводах

- Преобразовываем структуры (метки рёбер) в веса: $f(w) = \frac{w.Sum}{w.Count * 1000}$
- Сумма весов исходящих рёбер равна 1: $SoftMax(\vec{z})_i = \frac{e^{f(z_i)}}{\sum_{k=1}^K e^{f(z_k)}}$
- Знаменатель: $W = ReduceRows(+, Map(f, cards_subgraph))$
- Элементы W на нужных местах: $D = Mask(cards_subgraph, M_W \times M_1)$

$$Mask \left(\begin{pmatrix} 0 & c_0 & 0 \\ c_2 & 0 & c_1 \\ c_3 & 0 & 0 \end{pmatrix}, \overset{W}{\begin{pmatrix} w_0 & 0 & 0 \\ w_1 & 0 & 0 \\ w_2 & 0 & 0 \end{pmatrix}} \times \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \right) = \begin{pmatrix} 0 & w_0 & 0 \\ w_1 & 0 & w_1 \\ w_2 & 0 & 0 \end{pmatrix}$$

Получение матрицы весов по данным о переводах

- Преобразовываем структуры (метки рёбер) в веса: $f(w) = \frac{w.Sum}{w.Count * 1000}$
- Сумма весов исходящих рёбер равна 1: $SoftMax(\vec{z})_i = \frac{e^{f(z_i)}}{\sum_{k=1}^K e^{f(z_k)}}$
- Знаменатель: $W = ReduceRows(+, Map(f, cards_subgraph))$
- Элементы W на нужных местах: $D = Mask(cards_subgraph, M_W \times M_1)$

$$Mask \left(\begin{pmatrix} 0 & c_0 & 0 \\ c_2 & 0 & c_1 \\ c_3 & 0 & 0 \end{pmatrix}, \begin{pmatrix} w_0 & 0 & 0 \\ w_1 & 0 & 0 \\ w_2 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \right) = \begin{pmatrix} 0 & w_0 & 0 \\ w_1 & 0 & w_1 \\ w_2 & 0 & 0 \end{pmatrix}$$

- Матрица весов W — поэлементное деление матриц: $Map(f, cards_subgraph) / D$

- Что: ранжирование вершин по их «важности»
 - ▶ Оригинальная идея: вероятность попасть в вершину при случайном блуждании по графу

- Что: ранжирование вершин по их «важности»
 - Оригинальная идея: вероятность попасть в вершину при случайном блуждании по графу
- Зачем (например): выделение «подозрительных» вершин в графе транзакций

- Что: ранжирование вершин по их «важности»
 - Оригинальная идея: вероятность попасть в вершину при случайном блуждании по графу
- Зачем (например): выделение «подозрительных» вершин в графе транзакций
- Как: по определению
 - Матрица весов рёбер: W
 - Вектор (столбец) весов вершин: v
 - Итерируем $v = W \times v$ пока изменения v значимы

- FalkorDB и Cypher

- ▶ **FalkorDB** — графовая БД на основе линейной алгебры (SuiteSparse:GraphBLAS)
- ▶ Поддерживает подмножество Cypher
- ▶ Транслятор подмножества Cypher в операции линейной алгебры

- FalkorDB и Cypher

- ▶ **FalkorDB** — графовая БД на основе линейной алгебры (SuiteSparse:GraphBLAS)
- ▶ Поддерживает подмножество Cypher
- ▶ Транслятор подмножества Cypher в операции линейной алгебры

- Разное про SQL

- ▶ **TenSQL: An SQL Database Built on GraphBLAS**
- ▶ **TCUDB: Accelerating Database with Tensor Processors**
- ▶ ...

- FalkorDB и Cypher
 - ▶ FalkorDB — графовая БД на основе линейной алгебры (SuiteSparse:GraphBLAS)
 - ▶ Поддерживает подмножество Cypher
 - ▶ Транслятор подмножества Cypher в операции линейной алгебры
- Разное про SQL
 - ▶ TenSQL: An SQL Database Built on GraphBLAS
 - ▶ TCUDB: Accelerating Database with Tensor Processors
 - ▶ ...
- Линейная алгебра как основа для анализа данных
 - ▶ Towards a linear algebra semantics for SQL
 - ▶ Fast Join Project Query Evaluation using Matrix Multiplication
 - ▶ Fast Matrix Multiplication for Query Processing
 - ▶ A linear algebra approach to OLAP
 - ▶ Expressive Power of Linear Algebra Query Languages
 - ▶ On the expressiveness of Lara: A proposal for unifying linear and relational algebra