Линейная алгебра как основа для языка запросов к графам

Семён Григорьев

Санкт-Петербургский Государственный Университет

09 декабря 2025

Семён Григорьев

- 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

Линейная алгебра и анализ графов

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

Линейная алгебра и анализ графов

- **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

Линейная алгебра и анализ графов

- **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)

GraphBLAS[4]

- 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/

GraphBLAS[4]

- 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/

GraphBLAS[4]

- 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 Pointers[1]
  - **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

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

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

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

$\mathcal{N}_\mathcal{A} : [0 \ldots (n-1)] \to S_A,$ биекция

$\mathcal{N}_\mathcal{B} : [0 \ldots (k-1)] \to S_B,$ биекция

$\mathcal{N}_\mathcal{C} : [0 \ldots (m-1)] \to S_C,$ биекция

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

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

$M^{R_1}_{n \times k} = 1 \iff (\mathcal{N}_\mathcal{A}(i), \mathcal{N}_\mathcal{B}(j)) \in R_1,$ иначе $M^{R_1} = 0$

$M^{R_2}_{k \times m} = 1 \iff (\mathcal{N}_\mathcal{B}(i), \mathcal{N}_\mathcal{C}(j)) \in R_2,$ иначе $M^{R_2} = 0$

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

Отношения и матрицы

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

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

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

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

$\mathcal{N}_{\mathcal{A}} : [0 \ldots (n-1)] \to S_A$, биекция

$\mathcal{N}_{\mathcal{B}} : [0 \ldots (k-1)] \to S_B$, биекция

$\mathcal{N}_{\mathcal{C}} : [0 \ldots (m-1)] \to S_C$, биекция

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

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

$M^{R_1}_{n \times k} = 1 \iff (\mathcal{N}_{\mathcal{A}}(i), \mathcal{N}_{\mathcal{B}}(j)) \in R_1$, иначе $M^{R_1} = 0$

$M^{R_2}_{k \times m} = 1 \iff (\mathcal{N}_{\mathcal{B}}(i), \mathcal{N}_{\mathcal{C}}(j)) \in R_2$, иначе $M^{R_2} = 0$

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

Отношения и матрицы

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

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

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

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

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

$\mathcal{N}_{\mathcal{A}} : [0 \dots (n-1)] \to S_A,$ биекция

$\mathcal{N}_{\mathcal{B}} : [0 \dots (k-1)] \to S_B,$ биекция

$\mathcal{N}_{C} : [0 \dots (m-1)] \to S_C,$ биекция

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

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

$M^{R_1}_{n \times k} = 1 \iff (\mathcal{N}_{\mathcal{A}}(i), \mathcal{N}_{\mathcal{B}}(j)) \in R_1,$ иначе $M^{R_1} = 0$

$M^{R_2}_{k \times m} = 1 \iff (\mathcal{N}_{\mathcal{B}}(i), \mathcal{N}_{C}(j)) \in R_2,$ иначе $M^{R_2} = 0$

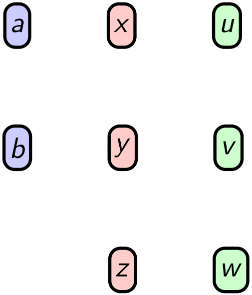
$M^{R_3}_{n \times m} = M^{R_1} \times M^{R_2}$

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

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

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

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



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

$\mathcal{N_A} : [0 \dots (n-1)] \to S_A,$ биекция

$\mathcal{N_B} : [0 \dots (k-1)] \to S_B,$ биекция

$\mathcal{N_C} : [0 \dots (m-1)] \to S_C,$ биекция

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

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

$M^{R_1}_{n \times k} = 1 \iff (\mathcal{N_A}(i), \mathcal{N_B}(j)) \in R_1,$ иначе $M^{R_1} = 0$

$M^{R_2}_{k \times m} = 1 \iff (\mathcal{N_B}(i), \mathcal{N_C}(j)) \in R_2,$ иначе $M^{R_2} = 0$

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

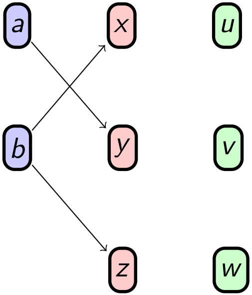
Отношения и матрицы

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

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

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

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



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

$\mathcal{N}_{\mathcal{A}} : [0 \ldots (n-1)] \to S_A,$ биекция

$\mathcal{N}_{\mathcal{B}} : [0 \ldots (k-1)] \to S_B,$ биекция

$\mathcal{N}_{\mathcal{C}} : [0 \ldots (m-1)] \to S_C,$ биекция

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

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

$M^{R_1}_{n \times k} = 1 \iff (\mathcal{N}_{\mathcal{A}}(i), \mathcal{N}_{\mathcal{B}}(j)) \in R_1,$ иначе $M^{R_1} = 0$

$M^{R_2}_{k \times m} = 1 \iff (\mathcal{N}_{\mathcal{B}}(i), \mathcal{N}_{\mathcal{C}}(j)) \in R_2,$ иначе $M^{R_2} = 0$

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

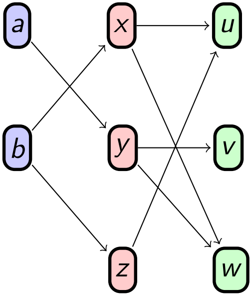
Отношения и матрицы

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

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

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

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



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

$\mathcal{N}_{\mathcal{A}} : [0 \ldots (n-1)] \to S_A,$ биекция

$\mathcal{N}_{\mathcal{B}} : [0 \ldots (k-1)] \to S_B,$ биекция

$\mathcal{N}_{\mathcal{C}} : [0 \ldots (m-1)] \to S_C,$ биекция

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

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

$M^{R_1}_{n \times k} = 1 \iff (\mathcal{N}_{\mathcal{A}}(i), \mathcal{N}_{\mathcal{B}}(j)) \in R_1,$ иначе $M^{R_1} = 0$

$M^{R_2}_{k \times m} = 1 \iff (\mathcal{N}_{\mathcal{B}}(i), \mathcal{N}_{\mathcal{C}}(j)) \in R_2,$ иначе $M^{R_2} = 0$

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

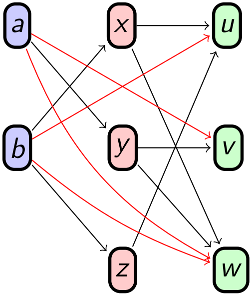
Отношения и матрицы

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

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

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

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



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

$\mathcal{N}_{\mathcal{A}} : [0 \ldots (n-1)] \to S_A,$ биекция

$\mathcal{N}_{\mathcal{B}} : [0 \ldots (k-1)] \to S_B,$ биекция

$\mathcal{N}_{C} : [0 \ldots (m-1)] \to S_C,$ биекция

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

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

$M^{R_1}_{n \times k} = 1 \iff (\mathcal{N}_{\mathcal{A}}(i), \mathcal{N}_{\mathcal{B}}(j)) \in R_1$, иначе $M^{R_1} = 0$

$M^{R_2}_{k \times m} = 1 \iff (\mathcal{N}_{\mathcal{B}}(i), \mathcal{N}_{C}(j)) \in R_2$, иначе $M^{R_2} = 0$

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

Умножение матриц и «полукольца»

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

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

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

Умножение матриц и «полукольца»

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

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

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

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

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

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

Умножение матриц и «полукольца»

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

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

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

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

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

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

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

$\oplus : T_3 \times T_3 \to T_3$

$\mathbb{0} : T_3$

Умножение матриц и «полукольца»

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

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

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

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

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

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

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

$\oplus : T_3 \times T_3 \longrightarrow T_3$

$\mathbb{0} : T_3$

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

Умножение матриц и «полукольца»

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

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

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

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

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

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

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

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

$$\oplus : T_3 \times T_3 \to T_3$$

$$\mathbb{0} : T_3$$

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

Умножение матриц и «полукольца»

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

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

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

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

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

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

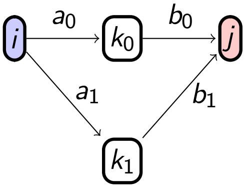
$\oplus : T_3 \times T_3 \rightarrow T_3$

$\mathbb{0} : T_3$

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

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

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



Умножение матриц и «полукольца»

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

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

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

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

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

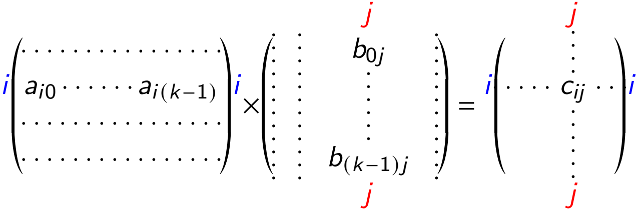
$\otimes : T_1 \times T_2 \to T_3$

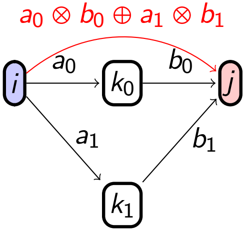
$\oplus : T_3 \times T_3 \to T_3$

$\mathbb{0} : T_3$

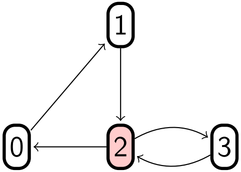
$$C[i,j] = \bigoplus_{k=0}^{n-1} A[i,k] \otimes B[k,j]$$

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

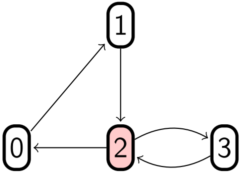




Обход в ширину

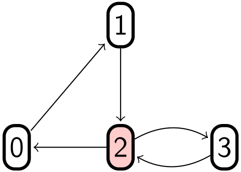


Обход в ширину



$$\underbrace{\begin{pmatrix} 0 & 0 & 1 & 0 \end{pmatrix}}_{\text{Текущий фронт}} \underset{\text{Полукольцо}}{\times} \underbrace{\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}}_{\text{Матрица смежности}} = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 1 \end{pmatrix}}_{\text{Новый фронт}}$$

Обход в ширину



Текущий фронт

Матрица смежности

Новый фронт

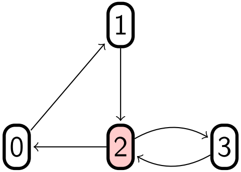
$$\begin{pmatrix}0 & 0 & 1 & 0\end{pmatrix} \times \begin{pmatrix}0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 1 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{pmatrix} = \begin{pmatrix}1 & 0 & 0 & 1\end{pmatrix}$$

Полукольцо

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

Посещённые вершины

Обход в ширину



Текущий фронт

Матрица смежности

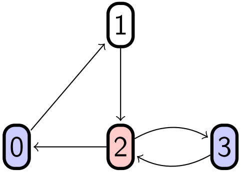
Новый фронт

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

Полукольцо

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

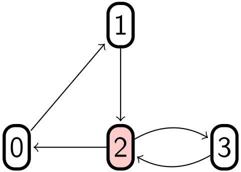
Посещённые вершины



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

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

Обход в ширину



Текущий фронт

Матрица смежности

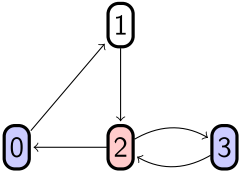
Новый фронт

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

Полукольцо

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

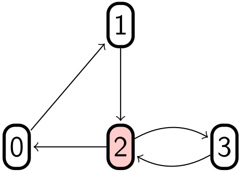
Посещённые вершины



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

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

Обход в ширину



Текущий фронт

Матрица смежности

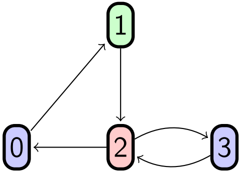
Новый фронт

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

Полукольцо

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

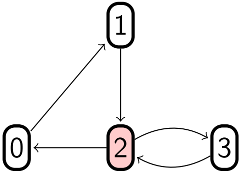
Посещённые вершины



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

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

Обход в ширину



Текущий фронт

Матрица смежности

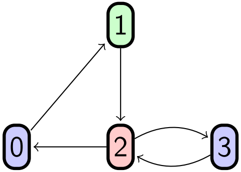
Новый фронт

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

Полукольцо

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

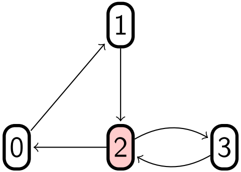
Посещённые вершины



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

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

Обход в ширину



Текущий фронт

Матрица смежности

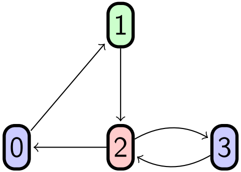
Новый фронт

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

Полукольцо

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

Посещённые вершины

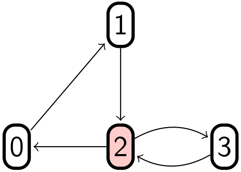


$$\begin{pmatrix} 0 & 1 & 0 & 0 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & \cancel{1} & 0 \end{pmatrix}$$

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

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

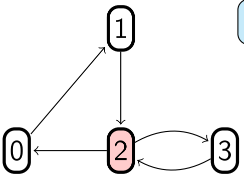
Обход в ширину с построением дерева обхода



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

Стартовая — предок

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

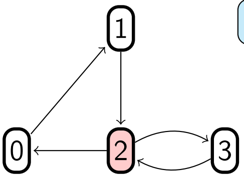
$$\begin{pmatrix} \_ & \_ & -1 & \_ \end{pmatrix}$$

«Нейтральный элемент»

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



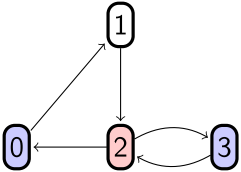
У стартовой нет предка

Стартовая — предок

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

$$\begin{pmatrix} \_ & \_ & -1 & \_ \end{pmatrix}$$

«Нейтральный элемент»



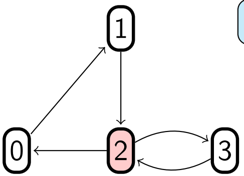
$$\begin{pmatrix} 2 & \_ & \_ & 2 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} \_ & 0 & 3 & \_ \end{pmatrix}$$

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

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



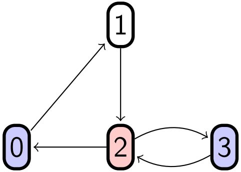
У стартовой нет предка

Стартовая — предок

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

$$\begin{pmatrix} \_ & \_ & -1 & \_ \end{pmatrix}$$

«Нейтральный элемент»



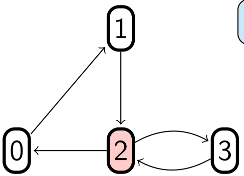
$$\begin{pmatrix} 2 & \_ & \_ & 2 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} \_ & 0 & \cancel{3} & \_ \end{pmatrix}$$

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

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



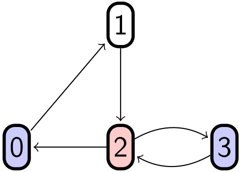
У стартовой нет предка

Стартовая — предок

«Нейтральный элемент»

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

$$\begin{pmatrix} \_ & \_ & -1 & \_ \end{pmatrix}$$



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

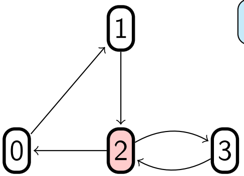
$$\begin{pmatrix} 2 & \_ & -1 & 2 \end{pmatrix}$$

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



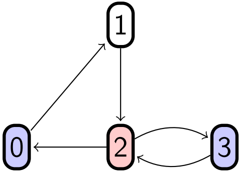
У стартовой нет предка

Стартовая — предок

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

$$(\_ \quad \_ \quad -1 \quad \_)$$

«Нейтральный элемент»



$$(2 \quad \_ \quad \_ \quad 2) \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = (\_ \quad 0 \quad \cancel{3} \quad \_)$$

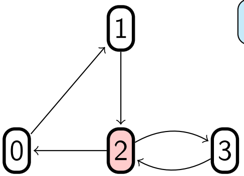
$$(2 \quad \_ \quad -1 \quad 2)$$

result: $(2 \quad 0 \quad -1 \quad 2)$

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



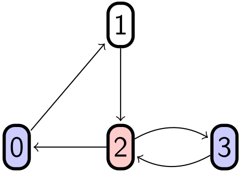
У стартовой нет предка

Стартовая — предок

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

$$\begin{pmatrix} \_ & \_ & -1 & \_ \end{pmatrix}$$

«Нейтральный элемент»



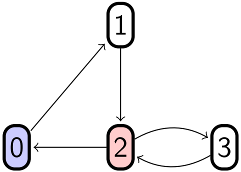
$$\begin{pmatrix} 2 & \_ & \_ & 2 \end{pmatrix} \times \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \begin{pmatrix} \_ & 0 & \cancel{3} & \_ \end{pmatrix}$$

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

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

Выбор входящих и исходящих (инцидентных) рёбер

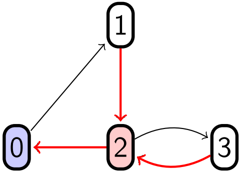
Выбор входящих рёбер — умножение на диагональную матрицу справа



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

Выбор входящих и исходящих (инцидентных) рёбер

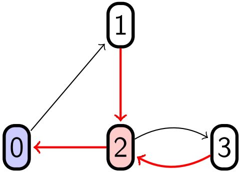
Выбор входящих рёбер — умножение на диагональную матрицу справа



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

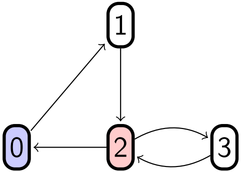
Выбор входящих и исходящих (инцидентных) рёбер

Выбор входящих рёбер — умножение на диагональную матрицу справа



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

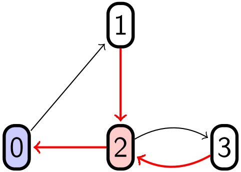
Выбор исходящих рёбер — умножение на диагональную матрицу слева



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

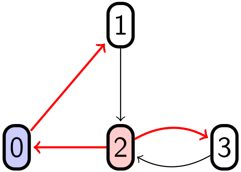
Выбор входящих и исходящих (инцидентных) рёбер

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



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

Пример[5]

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

[5] Код и описание: https://github.com/SparseLinearAlgebra/PageRankBenchmark

Пример[5]

- 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

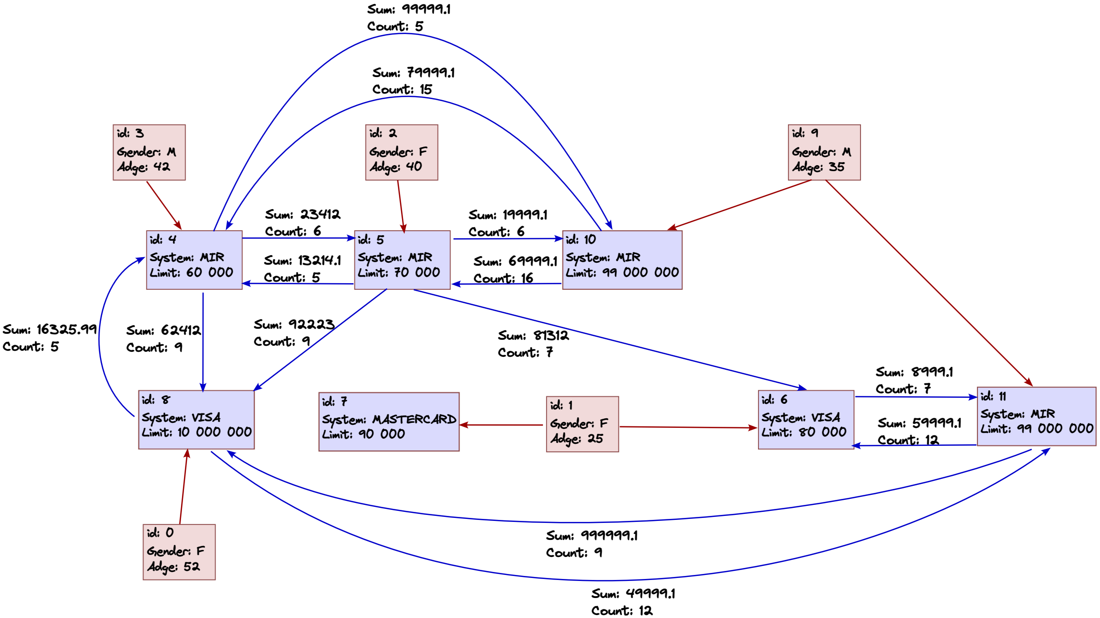
[5] Код и описание: `https://github.com/SparseLinearAlgebra/PageRankBenchmark`

Пример[5]

- 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

[5] Код и описание: https://github.com/SparseLinearAlgebra/PageRankBenchmark


Sum: 99999.1
Count: 5
Sum: 79999.1
Count: 15
id: 3
Gender: M
Adge: 42
id: 2
Gender: F
Adge: 40
id: 9
Gender: M
Adge: 35
Sum: 23412
Count: 6
Sum: 19999.1
Count: 6
id: 4
System: MIR
Limit: 60 000
id: 5
System: MIR
Limit: 70 000
id: 10
System: MIR
Limit: 99 000 000
Sum: 13214.1
Count: 5
Sum: 69999.1
Count: 16
Sum: 16325.99
Count: 5
Sum: 62412
Count: 9
Sum: 92223
Count: 9
Sum: 81312
Count: 7
Sum: 8999.1
Count: 7
id: 8
System: VISA
Limit: 10 000 000
id: 7
System: MASTERCARD
Limit: 90 000
id: 1
Gender: F
Adge: 25
id: 6
System: VISA
Limit: 80 000
id: 11
System: MIR
Limit: 99 000 000
Sum: 59999.1
Count: 12
id: 0
Gender: F
Adge: 52
Sum: 999999.1
Count: 9
Sum: 49999.1
Count: 12

Представление графа

- 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 $\texttt{Own\_Edges}_{12\times12}$ задаёт отношение «Владеет»

Представление графа

- Пользовательские типы (структуры) для описания пользователя, карты, информации о переводах: `User`, `Card`, `Trans`
- Булева матрица $\texttt{Own\_Edges}_{12\times 12}$ задаёт отношение «Владеет»
- Матрица $\texttt{Trans\_Edges}_{12\times 12}$ типа `Matrix<Trans>` задаёт отношение «Перевод»

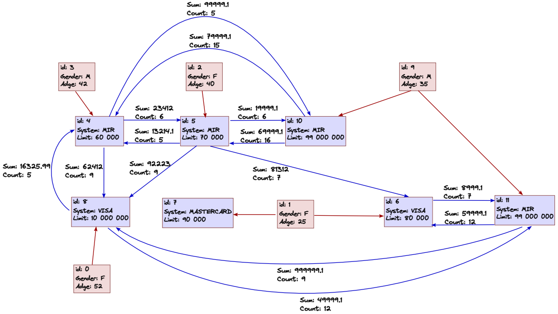
Представление графа

- Пользовательские типы (структуры) для описания пользователя, карты, информации о переводах: `User`, `Card`, `Trans`
- Булева матрица $\texttt{Own\_Edges}_{12\times 12}$ задаёт отношение «Владеет»
- Матрица $\texttt{Trans\_Edges}_{12\times 12}$ типа `Matrix<Trans>` задаёт отношение «Перевод»
- Вектор пользователей $\texttt{Users}_{12}$ хранит данные о пользователях

Представление графа

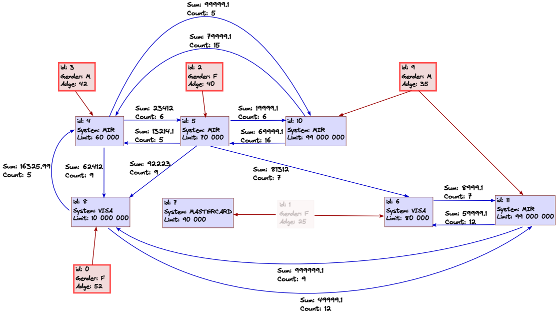
- Пользовательские типы (структуры) для описания пользователя, карты, информации о переводах: `User`, `Card`, `Trans`
- Булева матрица $\texttt{Own\_Edges}_{12\times 12}$ задаёт отношение «Владеет»
- Матрица $\texttt{Trans\_Edges}_{12\times 12}$ типа `Matrix<Trans>` задаёт отношение «Перевод»
- Вектор пользователей $\texttt{Users}_{12}$ хранит данные о пользователях
- Вектор карт $\texttt{Cards}_{12}$ хранит данные о картах

Построение подграфа



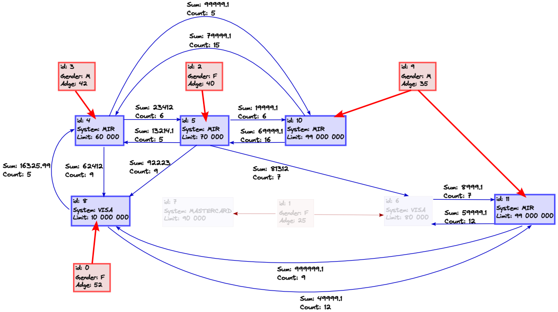
Построение подграфа

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



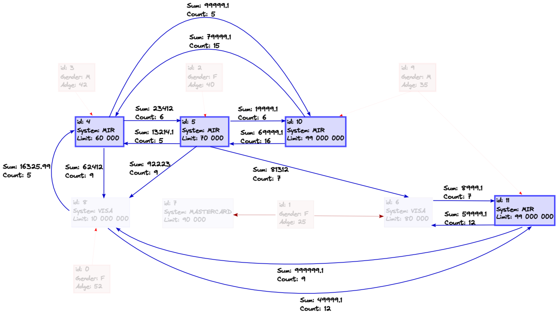
Построение подграфа

- Выбираем пользователей старше 30 лет
  - `s_users = 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



Построение подграфа

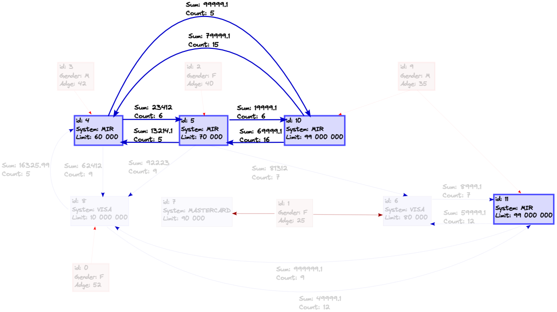
- Выбираем пользователей старше 30 лет
  - `s_users = Choose(filter, Users)`
  - `filter` — пользовательский предикат
- Выбираем карты пользователей
  - Один шаг обхода в ширину
  - `s_cards = s_users * Owns_Edges`
  - `s_cards` — не карты, а их индексы



- 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: $SoftMax(\overrightarrow{z})_i = \frac{e^{f(z_i)}}{\sum\limits_{k=1}^{K} e^{f(z_k)}}$

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: $SoftMax(\overrightarrow{z})_i = \frac{e^{f(z_i)}}{\sum\limits_{k=1}^{K} e^{f(z_k)}}$
- 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: $SoftMax(\overrightarrow{z})_i = \frac{e^{f(z_i)}}{\sum\limits_{k=1}^{K} e^{f(z_k)}}$
- Знаменатель: `W = ReduceRows(+, Map(f,cards_subgraph))`
- Элементы `W` на нужных местах: `D = Mask(cards_subgraph,` $M_W \times M_1$)

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

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: $SoftMax(\overrightarrow{z})_i = \frac{e^{f(z_i)}}{\sum\limits_{k=1}^{K} e^{f(z_k)}}$
- Знаменатель: `W = ReduceRows(+, Map(f,cards_subgraph))`
- Элементы `W` на нужных местах: `D = Mask(cards_subgraph,` $M_W \times M_1$`)`

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

- Матрица весов $\mathcal{W}$ — поэлементное деление матриц: `Map(f,cards_subgraph)/D`

PageRank

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

PageRank

- 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

PageRank

- 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: $\mathcal{W}$
  - Вектор (столбец) весов вершин: v
  - Итерируем $v = \mathcal{W} \times v$ пока изменения v значимы

Линейная алгебра и языки запросов

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

Линейная алгебра и языки запросов

- 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 и языки запросов

- 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