

A complex network diagram with various nodes and connections. Nodes are represented by circles of different sizes and colors (black, blue, grey). Connections are thin grey lines. The background is light grey with faint circular patterns.

# ИСПОЛЬЗОВАНИЕ СВОБОДНОГО ПО ДЛЯ РАЗРАБОТКИ СРЕДСТВ МОДЕЛИРОВАНИЯ СЕТЕВЫХ МОДЕЛЕЙ СЛОЖНЫХ СИСТЕМ

---

Муравьев Н. Д., Кулагин В.П.

# GPTN

- Написан на Rust и C++
- Моделирование параллельных процессов
- Моделирование систем с распределенными ресурсами
- Анализ тупиковых ситуаций
- Синтез новых вычислительных структур

# ЗАДАЧИ РАЗРАБОТКИ

- Кроссплатформенность
- Графический интерфейс
- Набор методов для задания сетей Петри (графический, матричный)
- Визуализация графов
- Методы анализа (построение ДДР, анализ на основе СЛАУ)
- Синтез новых вычислительных структур

The background features a complex network of thin grey lines connecting various circular nodes. The nodes vary in size and color, including dark blue, light blue, and grey. Some nodes are highlighted with larger, semi-transparent circles of the same color. A prominent dark blue node is located at the top center, and a light blue node is at the bottom left. The overall aesthetic is clean and modern, typical of a digital or technological theme.

# ГРАФИЧЕСКИЙ ИНТЕРФЕЙС

# ГРАФИЧЕСКИЙ ИНТЕРФЕЙС

Платформозависимые



Кроссплатформенные



The background features a complex network visualization with various nodes and connections. Nodes are represented by circles of different sizes and colors, including dark blue, light blue, and grey. Some nodes are highlighted with larger, semi-transparent circles. The connections are thin grey lines forming a dense web. A prominent dark blue node is at the top center, and a light blue node is at the bottom left. A black horizontal bar is positioned across the middle of the image, containing the title text.

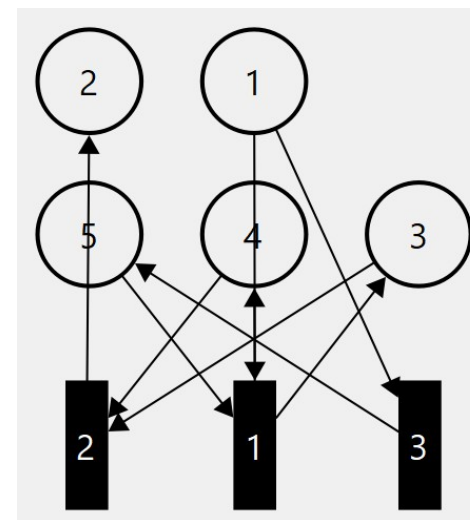
# ВИЗУАЛИЗАЦИЯ СЕТЕВЫХ МОДЕЛЕЙ

# ВИЗУАЛИЗАЦИЯ СЕТЕВЫХ МОДЕЛЕЙ

**Визуализация или отображение графов**, как ответвление теории графов, относящееся к топологии и геометрии — двумерное представление графа. В основном, это графическое представление укладки графа на плоскость, направленное, обычно, на удобное отображение некоторых свойств графа, или моделируемого объекта.

# ВИЗУАЛИЗАЦИЯ СЕТЕВЫХ МОДЕЛЕЙ

Input				Output			
	t1	t2	t3		t1	t2	t3
p1	1	0	1	p1	0	0	0
p2	0	0	0	p2	0	1	0
p3	0	1	0	p3	1	0	0
p4	0	1	0	p4	1	0	0
p5	1	0	0	p5	0	0	1

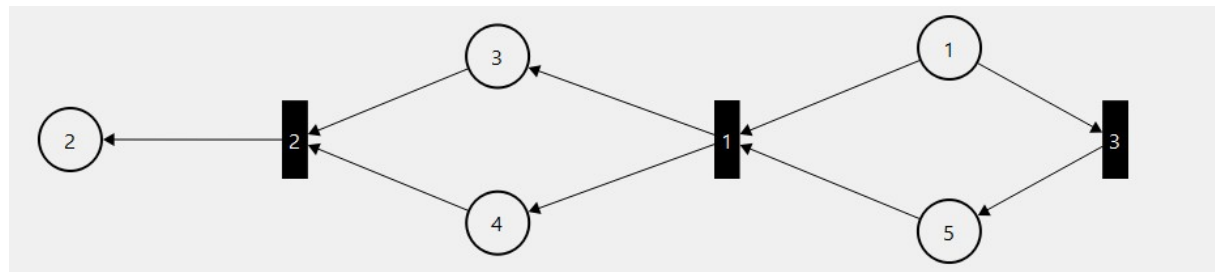
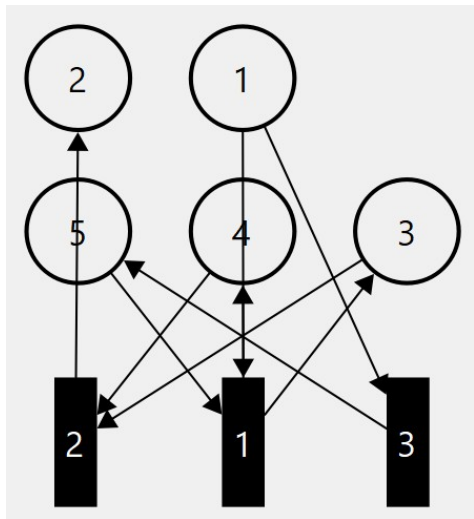




# ВИЗУАЛИЗАЦИЯ СЕТЕВЫХ МОДЕЛЕЙ (GRAPHVIZ)

- Dot – иерархическая визуализация, предназначенная для ориентированных графов
- Neato
- FDP (Force Directed Placement) – визуализация в основном неориентированных графов
- SFDP (Scalable Force Directed Placement)

# ВИЗУАЛИЗАЦИЯ СЕТЕВЫХ МОДЕЛЕЙ (GRAPHVIZ)



The background features a complex network diagram with various nodes and connections. Nodes are represented by circles of different sizes and colors, including dark blue, light blue, and grey. Some nodes are highlighted with larger, semi-transparent circles. The connections are thin grey lines forming a dense web. The overall aesthetic is clean and technical.

# АНАЛИЗ СЕТЕВЫХ МОДЕЛЕЙ НА ОСНОВЕ РЕШЕНИЯ СЛАУ

# АНАЛИЗ СЕТЕВЫХ МОДЕЛЕЙ НА ОСНОВЕ РЕШЕНИЯ СЛАУ

- Последовательная (содержит все переходы в ДДР)
- Инвариантная (сумма меток во всех позициях одинаковая)
- Живая и ограниченная (последовательная и инвариантна)
- Тупиковая (не последовательная)

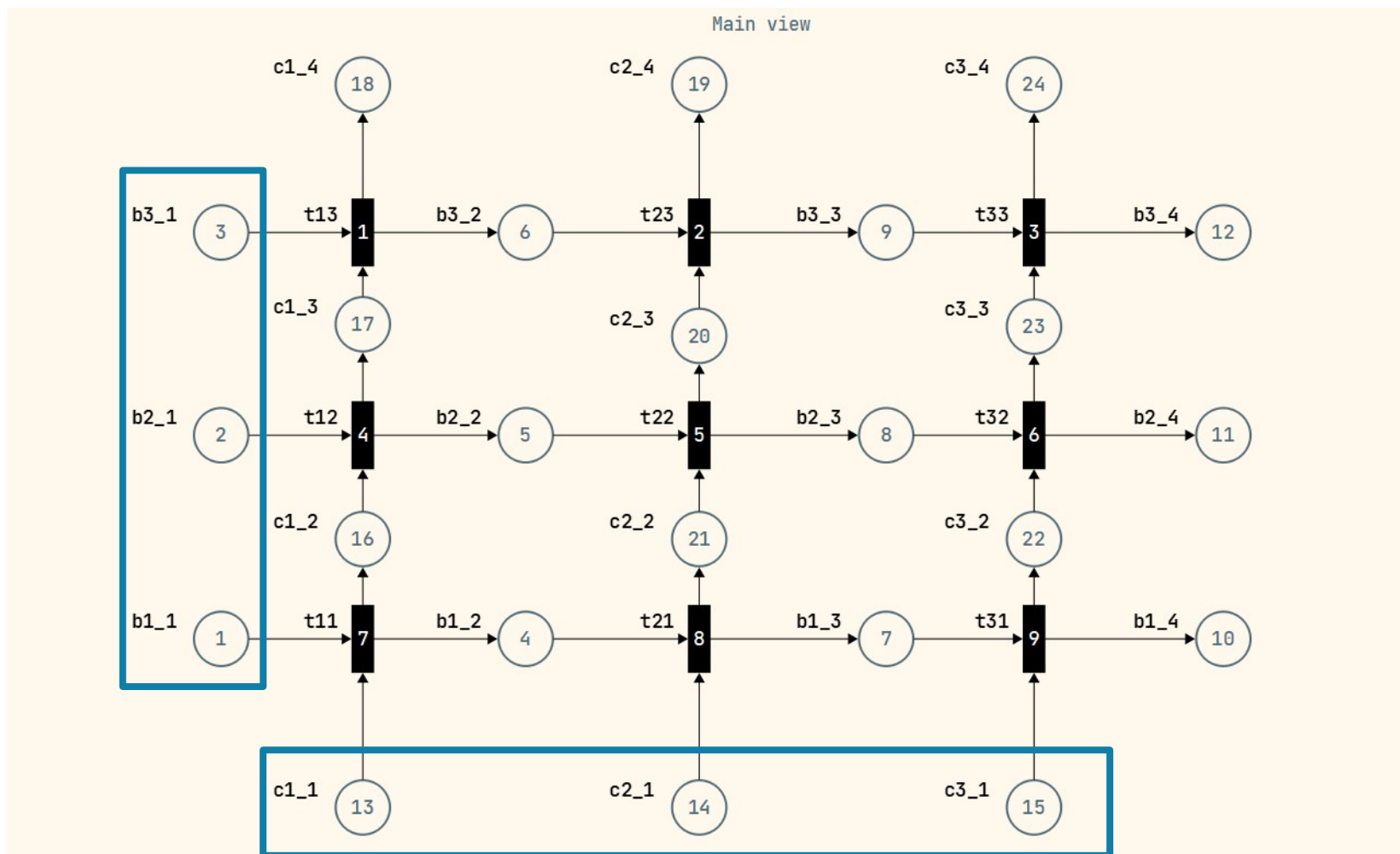
# OPENBLAS

- Написан на C
- Реализует базовые подпрограммы линейной алгебры
- Оптимизирован под разные архитектуры и виды процессоров
- Позволяет работать с матрицами

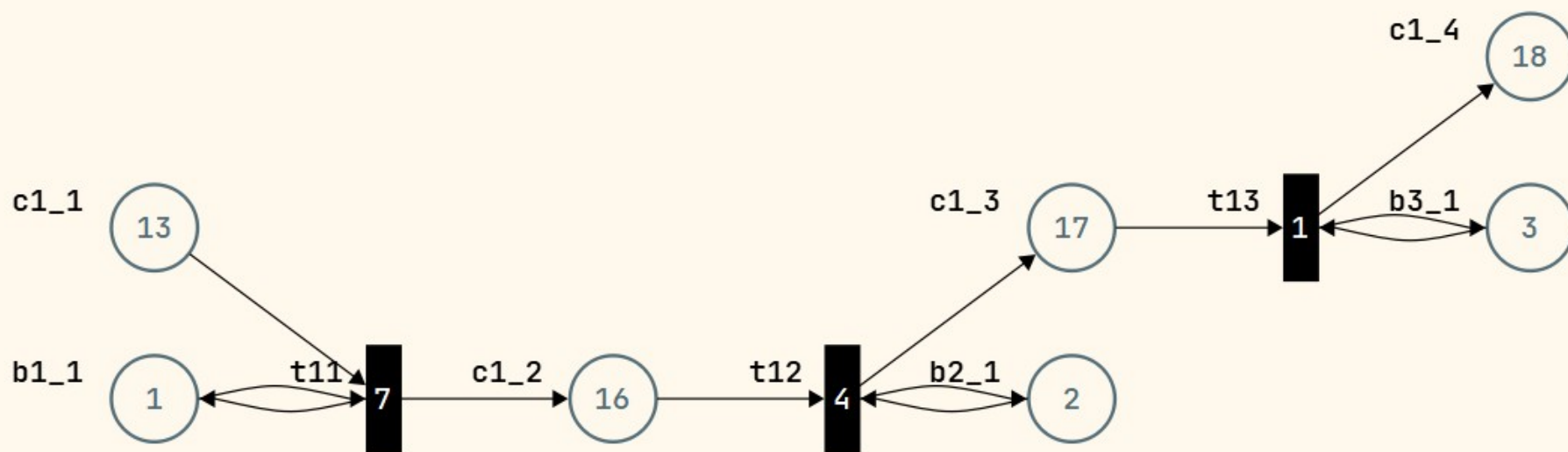
The background features a complex network of thin grey lines connecting various nodes. The nodes are represented by circles of different sizes and colors, including dark blue, light blue, and grey. Some nodes are highlighted with larger, semi-transparent circles of the same color. A prominent dark blue node is located at the top center, and a light blue node is at the bottom left. The overall aesthetic is clean and technical, suggesting a focus on data or network structures.

# СИНТЕЗ НОВЫХ ВЫЧИСЛИТЕЛЬНЫХ СТРУКТУР

# СИНТЕЗ НОВЫХ ВЫЧИСЛИТЕЛЬНЫХ СТРУКТУР



# СИНТЕЗ НОВЫХ ВЫЧИСЛИТЕЛЬНЫХ СТРУКТУР





# СИНТЕЗ НОВЫХ ВЫЧИСЛИТЕЛЬНЫХ СТРУКТУР

- Декомпозиция на линейно-циклические и линейные фрагменты
- Построение примитивной системы
- Поиск тензора преобразования из примитивной системы в линейно-базовую
- **Вычисление программ синтеза**
- Задание метрических шкал
- Отображение синтезированных структур на графике

# ВЫЧИСЛЕНИЕ ПРОГРАММ СИНТЕЗА (ПОЧЕМУ ТАК СЛОЖНО?)

- Поиск всех возможных сочетаний комбинаций вершин без пересечения
- Сложность алгоритма
- Более 30 минут при 12 вершинах
- Потребление памяти ~
- Более 4 GB памяти при 28 вершинах

# ВЫЧИСЛЕНИЕ ПРОГРАММ СИНТЕЗА (РЕШЕНИЕ)

Увеличение производительности и уменьшение потребления памяти

- **Постраничное\*** вычисление программ синтеза (увеличение производительности и уменьшение потребления памяти)
- **Фильтрация** программ синтеза с помощью экспертных систем
- **Распараллеливание** алгоритма поиска программ синтеза
- 
-

# ВЫЧИСЛЕНИЕ ПРОГРАММ СИНТЕЗА (КАК?)

- CPU (многопоточность)
- GPU (SIMD)
- Многомашинный комплекс

# ВЫЧИСЛЕНИЕ ПРОГРАММ СИНТЕЗА (ЧЕМ?)

- CUDA (только NVIDIA)
- Metal (только под MacOS)
- Vulkan API/OpenGL (ближе к графике)
- OpenCL

# OPENCL

- Написан на C
- Является полностью открытым стандартом
- Позволяет вычислять на различных графических и параллельных процессорах



# ДРУГИЕ СТОРОННИЕ ЗАВИСИМОСТИ

# ДРУГИЕ СТОРОННИЕ ЗАВИСИМОСТИ

- **Qt Advanced Docking System** – полнофункциональная система стыковки окон программы написанной на Qt
- **IndexMap** (Rust) – hash map с индексами
- **lib C** – bindings к системным библиотекам
- **ndarray-linalg** – обертка над OpenBLAS



# ПОЛЕЗНЫЕ ССЫЛКИ

- GraphViz - <https://www.graphviz.org/>
- QT - <https://www.qt.io/>
- OpenBLAS - <https://github.com/xianyi/OpenBLAS>
- LinAlg - <https://github.com/rust-ndarray/ndarray-linalg>
- GPTN - <https://github.com/tuplecats/GPTN>
- Qt Advanced Docking System - <https://github.com/githubuser0xFFFF/Qt-Advanced-Docking-System>
-