Санкт-Петербургский государственный университет
Математико-механический факультет
Кафедра системного программирования

Разработка алгоритмов и программных средств управления проектами с использованием моделей и методов идемпотентной алгебры

Пузиков Александр Юрьевич,
группа 661

Научный руководитель: Кривулин Н.К., д.ф.-м.н., профессор

Введение

- Идемпотентная алгебра (ИА)
  - Молодая область
  - Новый взгляд на классические задачи

- Управление проектами (УП)
  - Классические методы и задачи

Постановка задачи

Цель работы: на основе существующих моделей и методов идемпотентной математики разработать вычислительные алгоритмы и программный прототип временного планирования проектов

Задачи:

1. Исследовать дисциплину управления проектами
2. Изучить различные модели и методы идемпотентной алгебры
3. На основе изученных методов разработать вычислительные алгоритмы временного планирования
4. Реализовать программный прототип, использующее методы для управления проектом.

Идемпотентная алгебра

X –множество, операции сложение $\bigoplus$ и умножение $\bigotimes$, нуль $\mathbb{0}$ и единица $\mathbb{1}$.

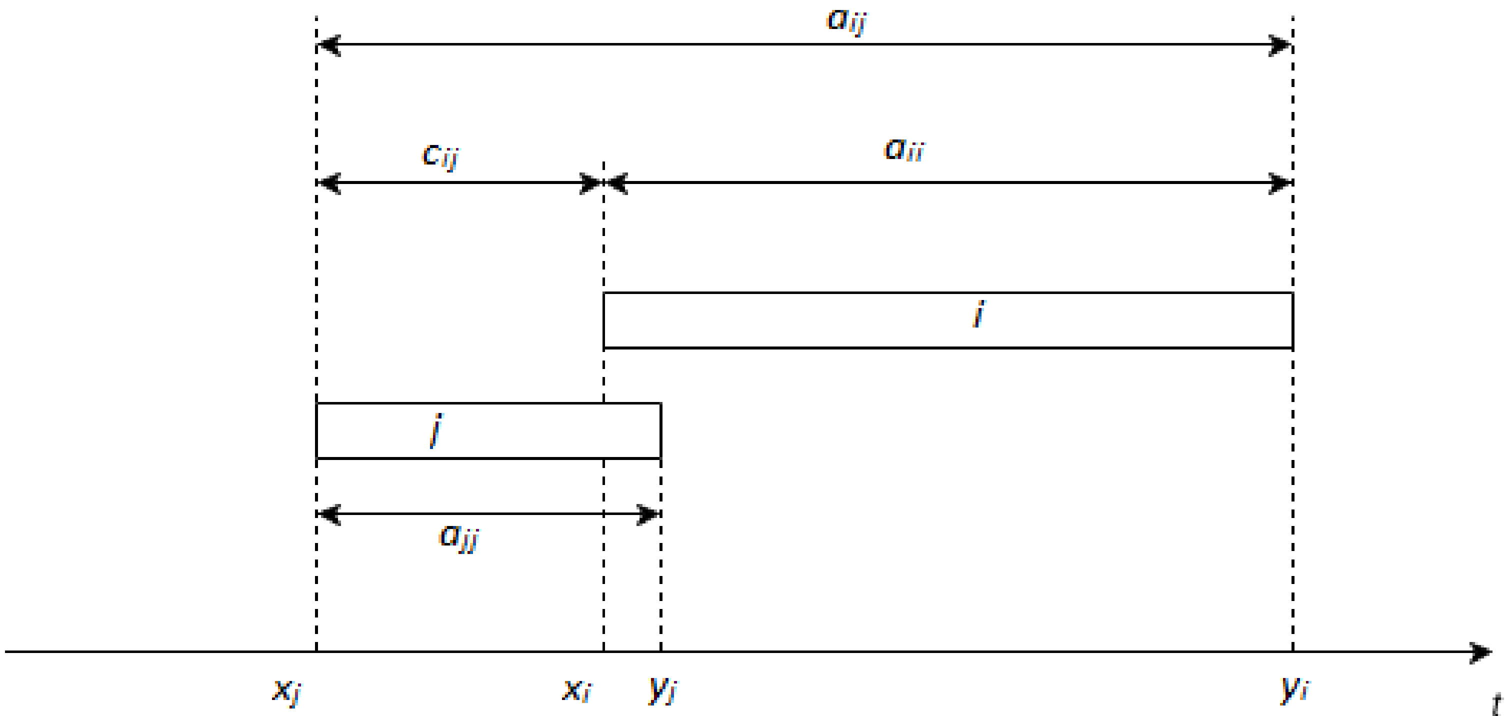
- Ассоциативность сложения и умножения
- $\boldsymbol{x} \bigoplus \boldsymbol{x} = \boldsymbol{x}$**,** (идемпотентность сложения)
- $\boldsymbol{x} \bigotimes \boldsymbol{x}^{-1} = \mathbb{1}$ (существование обратного для любого $\boldsymbol{x}$ кроме нуля)
- Дистрибутивность

Например, $R_{max;+}$ имеет нуль $\mathbb{0} = -\infty$ и единицу $\mathbb{1} = \mathbf{0}$.

Прямоугольные матрицы m x n над X: $X^{m\text{x}n}$.

Сопряженно-транспонированная матрица: $\boldsymbol{A}^{-} \in \boldsymbol{X}^{m\text{x}n}$ с элементами $\boldsymbol{a}^{-}_{ij} = \boldsymbol{a}^{-1}_{ji}$ если $\boldsymbol{a}_{ji} \neq \mathbb{0}$, и $\boldsymbol{a}^{-}_{ij} = \mathbb{0}$ иначе.

Проекты и операции



Проект состоит из *n* операций.

Связи между операциями можно представить в виде детерминированной сетевой модели.

Для фиксации последовательности операций вводится дополнительная матрица инцидентности *B.*

Критерии оптимальности

1. Минимизация максимального отклонения времени завершения операций от директивных сроков.
2. Минимизация общего времени выполнения (makespan).
3. Минимизация максимального времени цикла операций (flow time). (Demeulemeester, Herrolen)
4. Минимизация (just-in-time планирование) и максимизация разброса времени завершения операций(V. T’kindt и J.-C. Billaut)

Временные ограничения проекта

Ограничение	Классический вид	Идемпотентный аналог
Старт-финиш	$\max\limits_{1 \le j \le n}(x_j + a_{ij}) = y_i$	$Ax = y$
Старт-старт	$\max\limits_{1 \le j \le n}(x_j + c_{ij}) \le x_i$	$Cx \le x$
Самые ранние сроки начала	$x_i \ge g_i$	$g \le x$
Самые поздние сроки завершения	$h_i \ge y_i$	$h \ge y$

A – матрица значений a_{ij}

C – матрица значений c_{ij}

g – вектор границ времени завершения операций

h – вектор границ самого раннего времени начала операций

Задачи в терминах идемпотентной алгебры

1. Минимизация общего времени выполнения (makespan)

$$\max_{1\le i\le n} y_i + \max_{1\le i\le n}(-x_i) \rightarrow min$$

$$\mathbb{1}^T A x x^- \mathbb{1} \rightarrow min$$

$$\Delta = \mathbb{1}^T A \mathbb{1}$$

$$x = \left(I\ (\mathbb{1}^T A \mathbb{1})^{-1} \mathbb{1}\ \ \mathbb{1}^T A\right) u, u > 0,\ \ y = Ax,$$

где I – единичная матрица, $\mathbb{1} = \{0, \dots, 0\}^T$.

2. Минимизация (just-in-time планирование) разброса времени завершения операций.

$$\max_{1\le i\le n}(y_i) + \max_{1\le i\le n}(-y_i) \rightarrow min$$

$$\mathbb{1}^T y y^- \mathbb{1} \rightarrow min$$

$$\Delta = (AC^*(\mathbb{1}AC^*)^-)^-\mathbb{1}$$

$$x = C^*(\mathbb{1}AC^*)^-, y = AC^*(\mathbb{1}AC^*)^-.$$

Задачи в терминах идемпотентной алгебры

3. Максимизация разброса времени завершения операций.

$$\max_{1\le i\le n}(y_i) + \max_{1\le i\le n}(-y_i) \to max \Leftrightarrow \mathbb{1}^T y y^{-} \mathbb{1} \to max$$

$$Ax = y;$$

$$Cx \le x.$$

4. Минимизация максимального времени цикла операций (flowtime)

$$\max_{1\le i\le n}(y_i - x_i) \to min \Leftrightarrow x^{-}\ y \to min$$

$$Ax = y;\ x \le g;$$

$$Cx \le x;\ h \ge y\quad .$$

5. Минимизация максимального отклонения времени завершения операций от директивных сроков.

b – вектор срокова завершения для каждой задачи *i*

Ax = b – запись ограничений на директивные сроки

$$\max(\max_{1\le i\le n}(a_{ij}+x_i - b_i)\ ,\ \max_{1\le i\le n}(b_i - (a_{ij}+x_i))) \to min$$

$$b^{-}Ax \oplus (Ax)^{-}b \to min$$

Известные подходы

Классические решения:

1. Project Evaluation and Review Technique (Pert)

На основе метода критического пути (Critical path method) и диаграмм Ганта для детерминированных сетей.

1. Определение необходимые и реальные сроки выполнения проектов.
2. Обнаружение критических операций в проекте.

Используется в Microsoft Project, Visio, Gantt Project etc.

2. Graphical Evaluation and Review Technique (Gert)

Для стохастический сетей:

1. Детерминированные и вероятностные типы событий.
2. Позволяет определяет длительность и вероятность реализации последовательности событий.

Моделирование Монте-Карло.

Известные подходы

1. Применяются методы целочисленного, линейного и нелинейного программирования.

2. Вычислительные процедуры способны искать только частные решения или устанавливать, что решений нет.

3. Различные алгоритмические решения, как правило, всякий раз требуют разработки новых программных средств.

4. Распараллеливание итеративных вычислительных схем обычно затруднено.

Используемые в разработке библиотеки

Java SE 1.7 с использованием:

1. **JgraphX.**
2. **SwiftGanttChart**
3. **Tropical**

Пример (flow-time).

5 операций в проекте.

A – матрица ограничений старт-финиш

C – матрица ограничений старт-старт

g и h – векторы ограничений.

$$A = \begin{pmatrix} 4 & 0 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 6 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}. \quad B = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 8 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 10 & 12 & 0 \end{pmatrix} \quad C = AB = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 12 & 0 & 0 & 0 & 0 \\ 0 & 15 & 0 & 0 & 0 \\ 0 & 15 & 0 & 0 & 0 \\ 0 & 0 & 15 & 18 & 0 \end{pmatrix}.$$

$$g = \begin{pmatrix} 0 \\ 10 \\ 19 \\ 19 \\ 29 \end{pmatrix}, \quad h = \begin{pmatrix} 14 \\ 20 \\ 25 \\ 28 \\ 33 \end{pmatrix}. \qquad x = \begin{pmatrix} 0.0 \\ 10.0 \\ 20.0 \\ 20.0 \\ 32.0 \end{pmatrix}, \quad y = \begin{pmatrix} 4.0 \\ 15.0 \\ 25.0 \\ 26.0 \\ 33.0 \end{pmatrix}.$$

Примем значение 0.0 за дату 05 мая 21:52:00 2014 года.

Пример (flow-time).

Временные ограничения на:

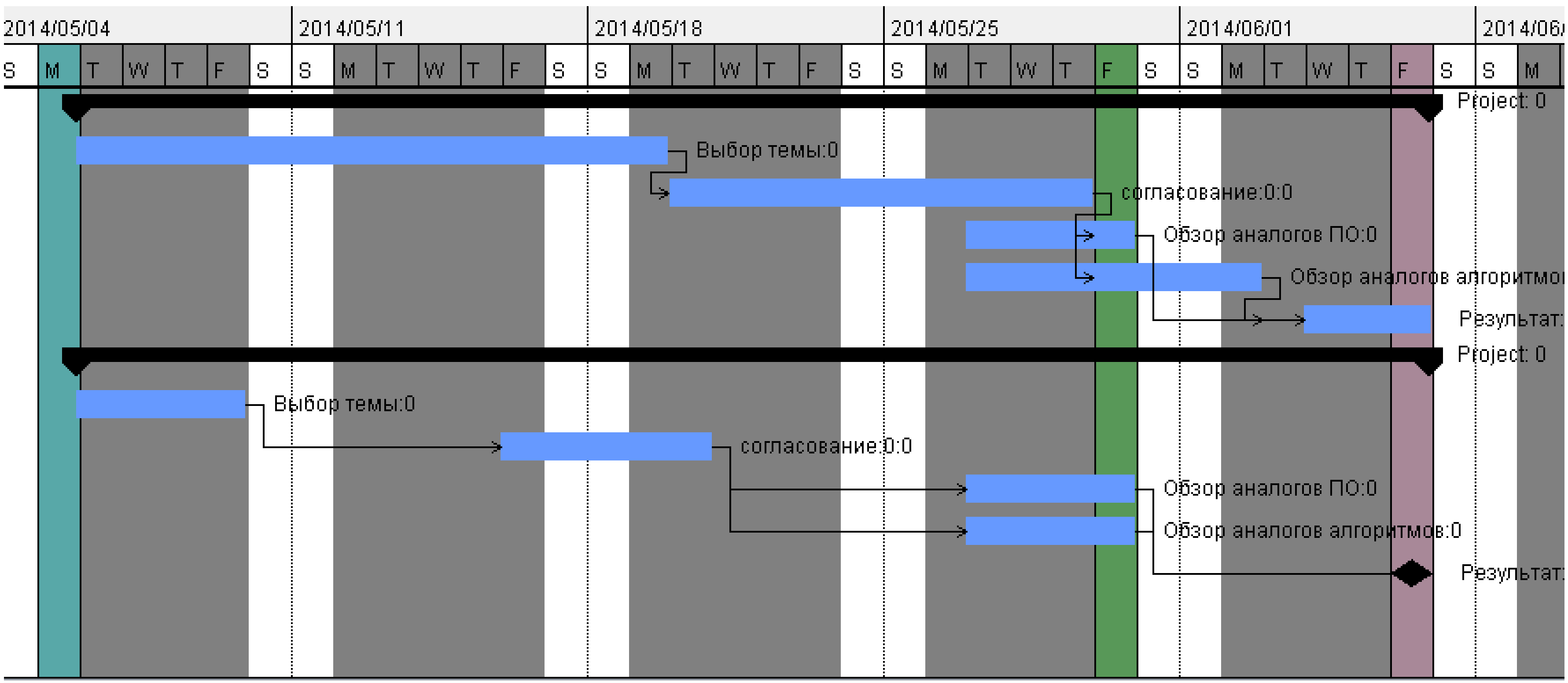
Start - finish

	Выбор те...	согласов...	Обзор ан...	Обзор ан...	Результа...
Выбор те...	4.0	-Infinity	-Infinity	-Infinity	-Infinity
согласов...	-Infinity	5.0	-Infinity	-Infinity	-Infinity
Обзор ан...	-Infinity	-Infinity	5.0	-Infinity	-Infinity
Обзор ан...	-Infinity	-Infinity	-Infinity	6.0	-Infinity
Результа...	-Infinity	-Infinity	-Infinity	-Infinity	1.0

Временные ограничения на:

Incidence

	Выбор те...	согласов...	Обзор ан...	Обзор ан...	Результа...
Выбор те...	-Infinity	-Infinity	-Infinity	-Infinity	-Infinity
согласов...	4.0	-Infinity	-Infinity	-Infinity	-Infinity
Обзор ан...	-Infinity	5.0	-Infinity	-Infinity	-Infinity
Обзор ан...	-Infinity	5.0	-Infinity	-Infinity	-Infinity
Результа...	-Infinity	-Infinity	5.0	6.0	-Infinity



Результаты

1. Исследованы методы управления проектами, в том числе алгоритмы (PERT, GERT, CPM)
2. Изучены различные модели и методы идемпотентной алгебры и способы их применения в задачах планирования проектов
3. На основе изученных методов разработаны вычислительные алгоритмы временного планирования
4. На платформе Java SE 1.7 создан программный прототип использующий разработанные методы для управления проектом.