

# Анализ UML диаграмм деятельности с помощью раскрашенных сетей Петри

Дипломная работа

Студент: Романов Алексей Сергеевич  
Руководитель: Рудаков Игорь Владимирович

# Цели и решаемые задачи

**Целью работы** является разработка метода представления диаграммы деятельности раскрашенной сетью Петри, позволяющего выявить блокировки и недостижимые состояния.

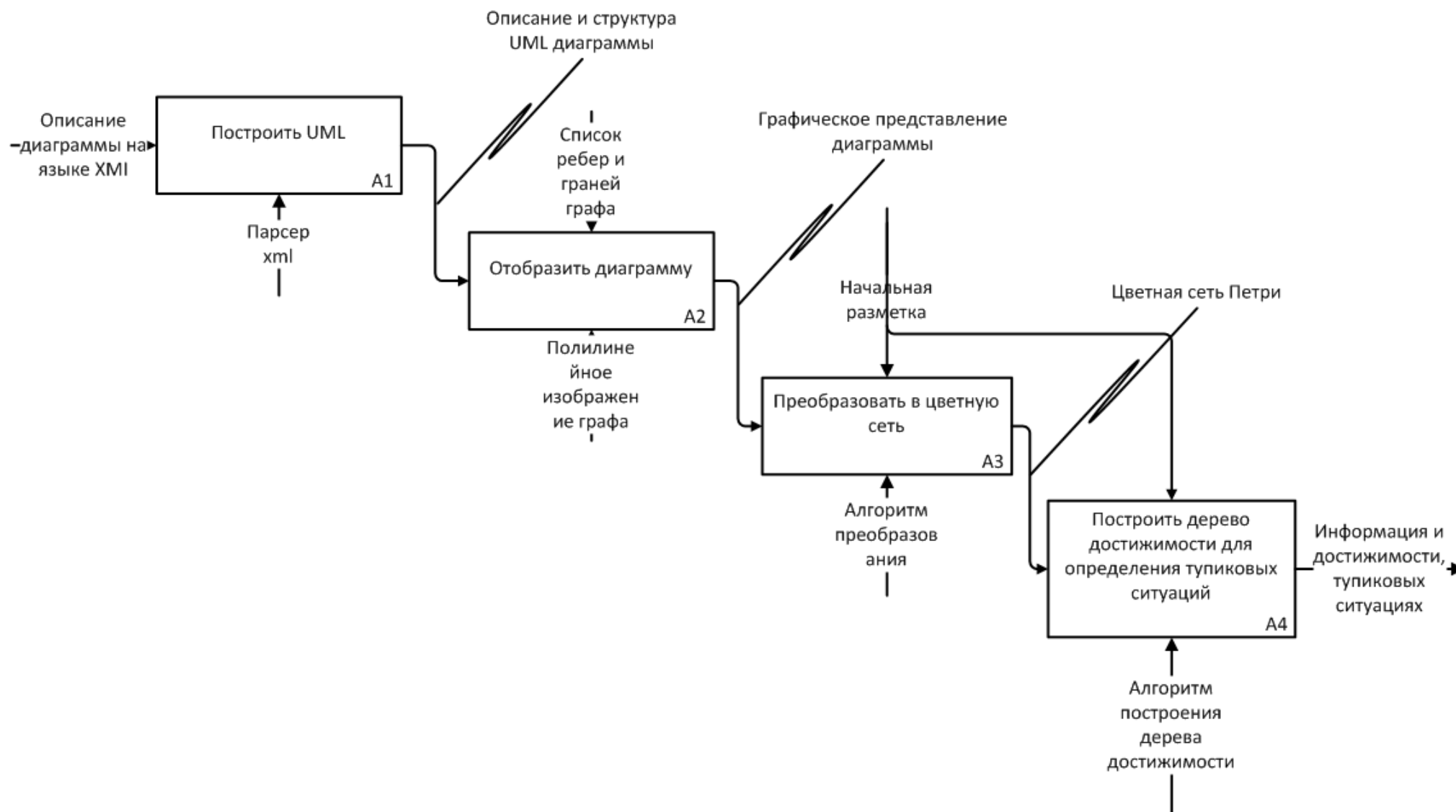
## Решаемые задачи

1. Классифицировать существующие методы анализа диаграмм деятельности.
2. Разработать метод представления диаграммы деятельности в виде раскрашенной сети Петри.
3. Программно реализовать метод представления диаграммы деятельности в виде раскрашенной сети Петри.
4. Исследовать факторы, влияющие на появление блокировок.
5. Исследовать корректность построения раскраски сети.

# Классификация методов анализа диаграмм деятельности



# Функциональная модель метода



# Представление UML диаграмм

UML := (XMI (XML) | OMG),  
где  
XMI — XML Metadata  
Interchange  
XML — eXtensible Markup  
Language  
OMG — Object Management  
Group

```
<activity_diagram>
  <states>
    <state id, name, type>
      <incoming transitions>
      <outgoing transition>
      <action>
    </state>
  </states>
  <transitions>
    <transition id>
      <source state>
      <target state>
      <guard>
    </transition>
  </transitions>
</activity_diagram>
```

# Отображение диаграммы деятельности

Представим диаграмму деятельности в виде ориентированного графа  $G = (V, E)$ , где

$V = (v_1 \dots v_k)$ ,  $v_i$  — вершина графа;

$E = (e_1 \dots e_n)$ ,  $e_i$  — переход между вершинами.

Алгоритм отображения состоит из трех этапов.

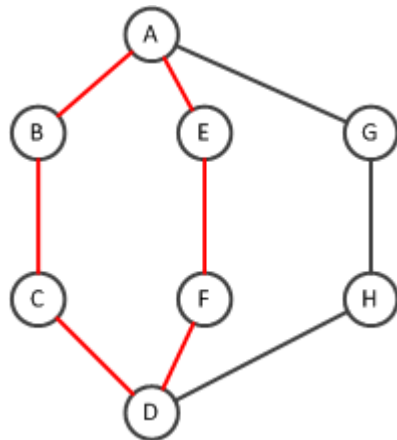
1. Построение ассоциированного орграфа.
2. Топологическая сортировка вершин исходного и ассоциированного графа.
3. Мозаичное и полилинейное представление.

# Построение ассоциированного орграфа

Ассоциированный с графом  $G$  граф  $G^* = (V^*, E^*)$ , где

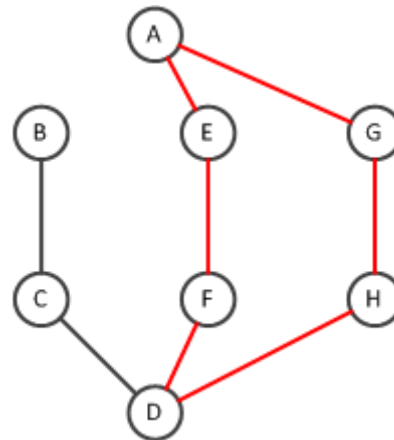
$V^* : v \in F$

$E^* : \forall e \in E : e^* = (f, g), f = left(e), g = right(e)$



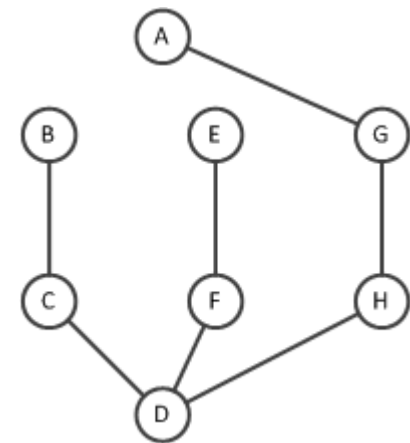
Цикл (A,B)

$\rightarrow (B, C) \rightarrow (C, D) \rightarrow$   
 $(D, F) \rightarrow (F, E) \rightarrow (E, A)$



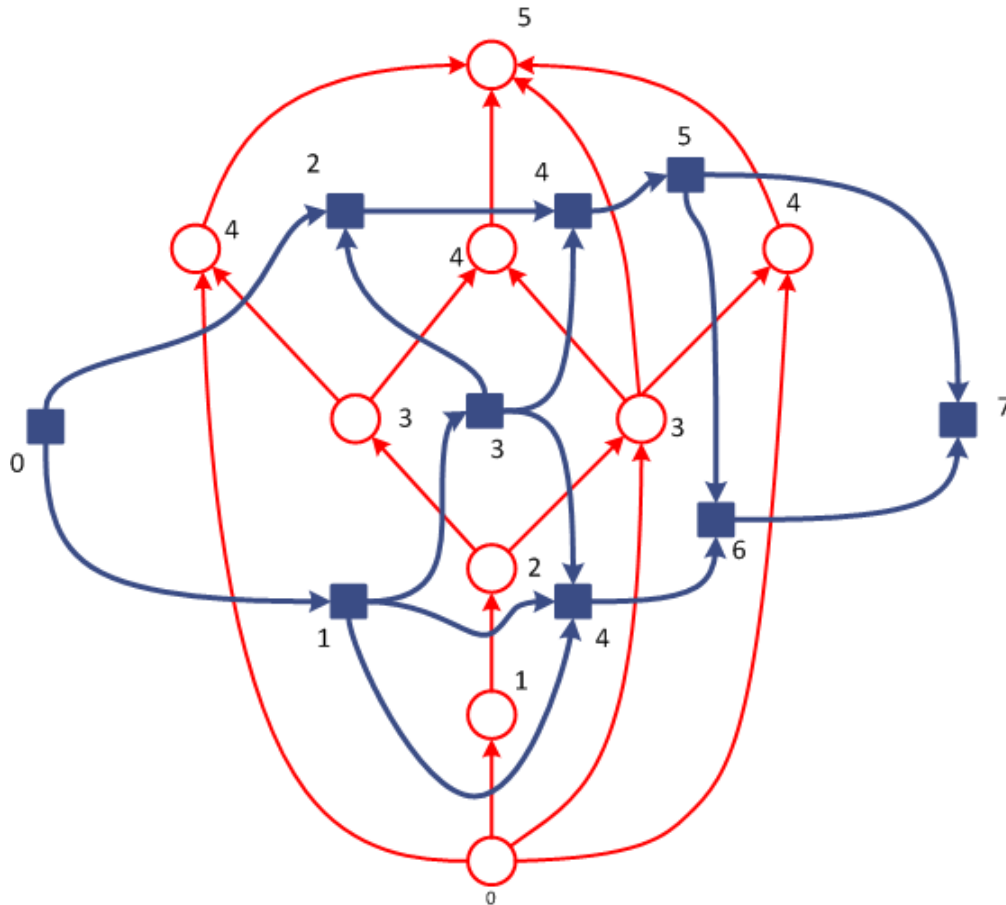
Цикл (A,E)

$\rightarrow (E, F) \rightarrow (F, D) \rightarrow$   
 $(D, H) \rightarrow (H, G) \rightarrow (G, A)$

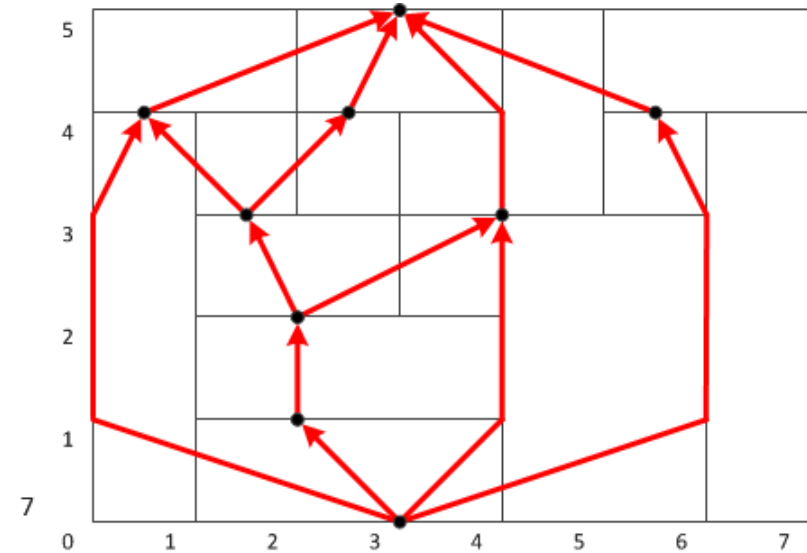


Циклов нет

# Мозаичное представление графа



Граф  $G$  (красный) и  
ассоциированный оргграф  $G^*$   
(синий).



Полилинейное представление  
графа  $G$ .

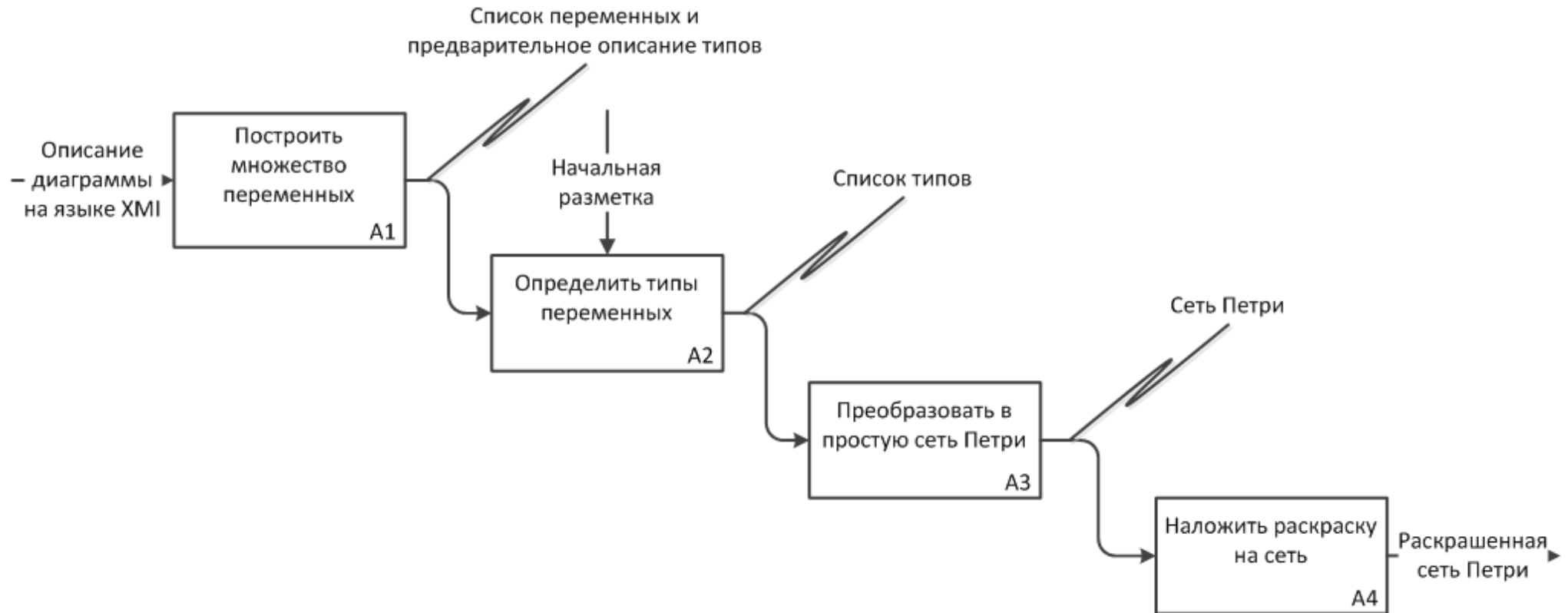


# Раскрашенные сети Петри

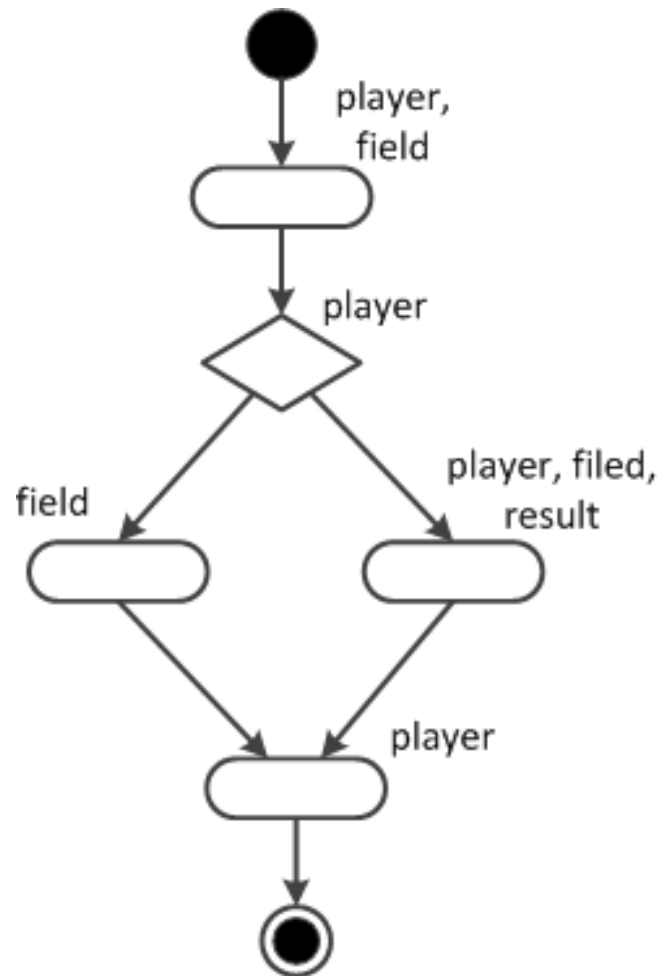
Раскрашенная сеть Петри  $CPN = (\Sigma, P, T, A, N, C, G, E, I)$ , где

- $\Sigma$  — конечное непустое множество цветов;
- $P$  — конечное множество позиций;
- $T$  — конечное множество состояний;
- $A$  — конечное множество дуг, таких что  $P \cap T = P \cap A = T \cap A = \emptyset$ ;
- $N$  — позиционная функция, отображающая  $A$  в  $P \times T \cup T \times P$ ;
- $C$  — функция раскраски, отображающая  $P$  в  $\Sigma$ ;
- $G$  — защита переходов, отображающая  $T$  в выражение вида  $\forall t \in T : [Type(G(t)) = boolType(Var(G(t))) \subseteq \Sigma]$ ;
- $E$  — выражения на дугах, отображающая  $A$  в выражение вида  $\forall a \in A : [Type(E(a)) = C(p(a))_{ms}Type(Var(E(a))) \subseteq \Sigma]$ , где  $p(a)$  — позиция  $N(a)$ ;
- $I$  — функция инициализации, отображающая  $P$  в выражение вида  $\forall p \in P : [Type(I(p)) = C(p)_{ms}]$ .

# Этапы построения раскрашенной сети Петри



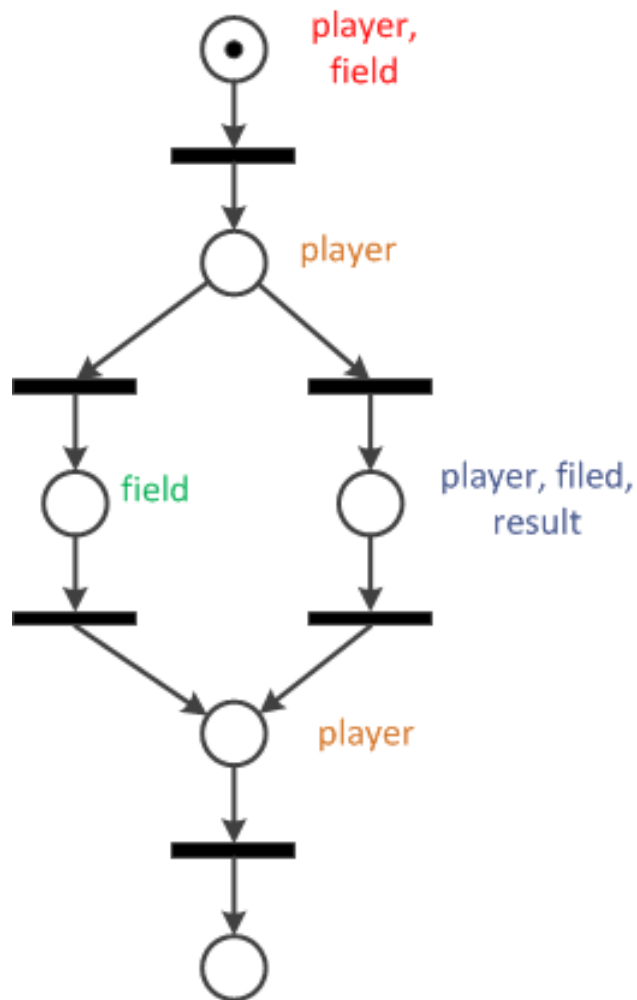
# Преобразование диаграммы деятельности в раскрашенную сеть Петри (1)



1. Разбор выражений.

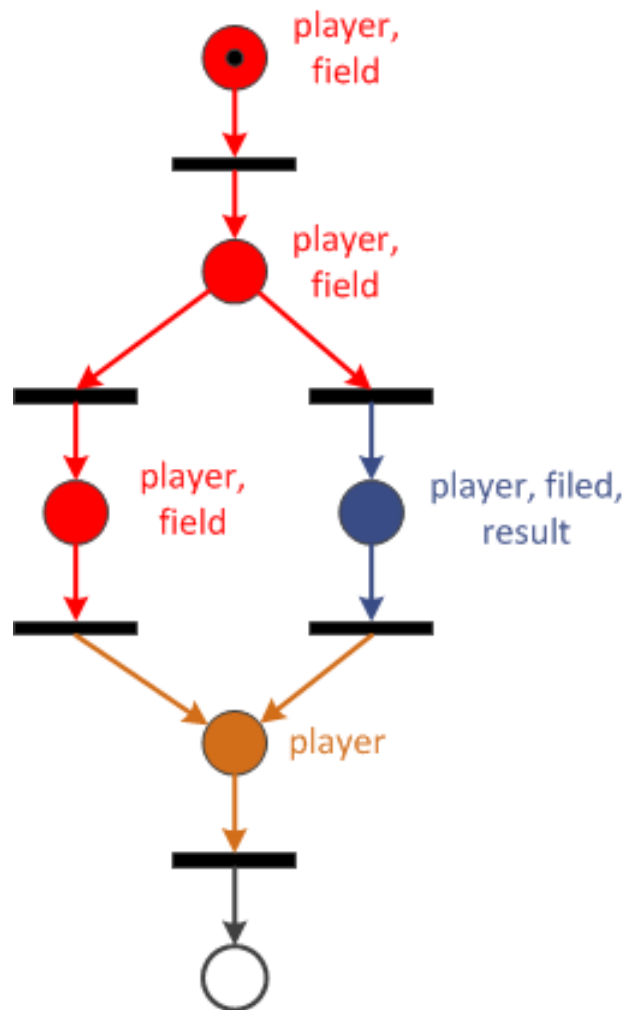
2. Выделение списка переменных для каждой вершины.

# Преобразование диаграммы деятельности в раскрашенную сеть Петри (2)



1. Преобразование диаграммы деятельности в простую сеть Петри.
2. Формирование множества ТИПОВ переменных.
3. Предварительное определение множества раскрасок.

# Преобразование диаграммы деятельности в раскрашенную сеть Петри (3)

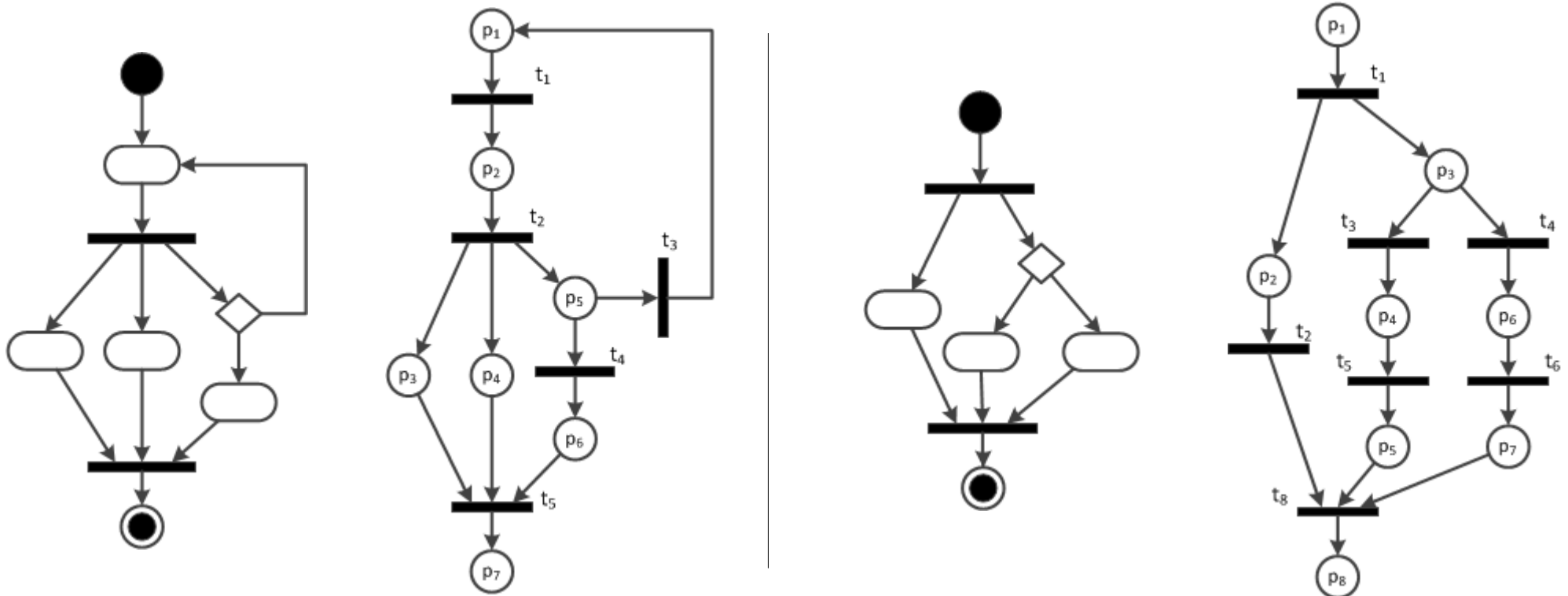


1. Определение максимальной области видимости переменных.
2. Формирование результирующей раскраски.

# Исследование причин возникновения блокировок

Причины возникновения блокировок:

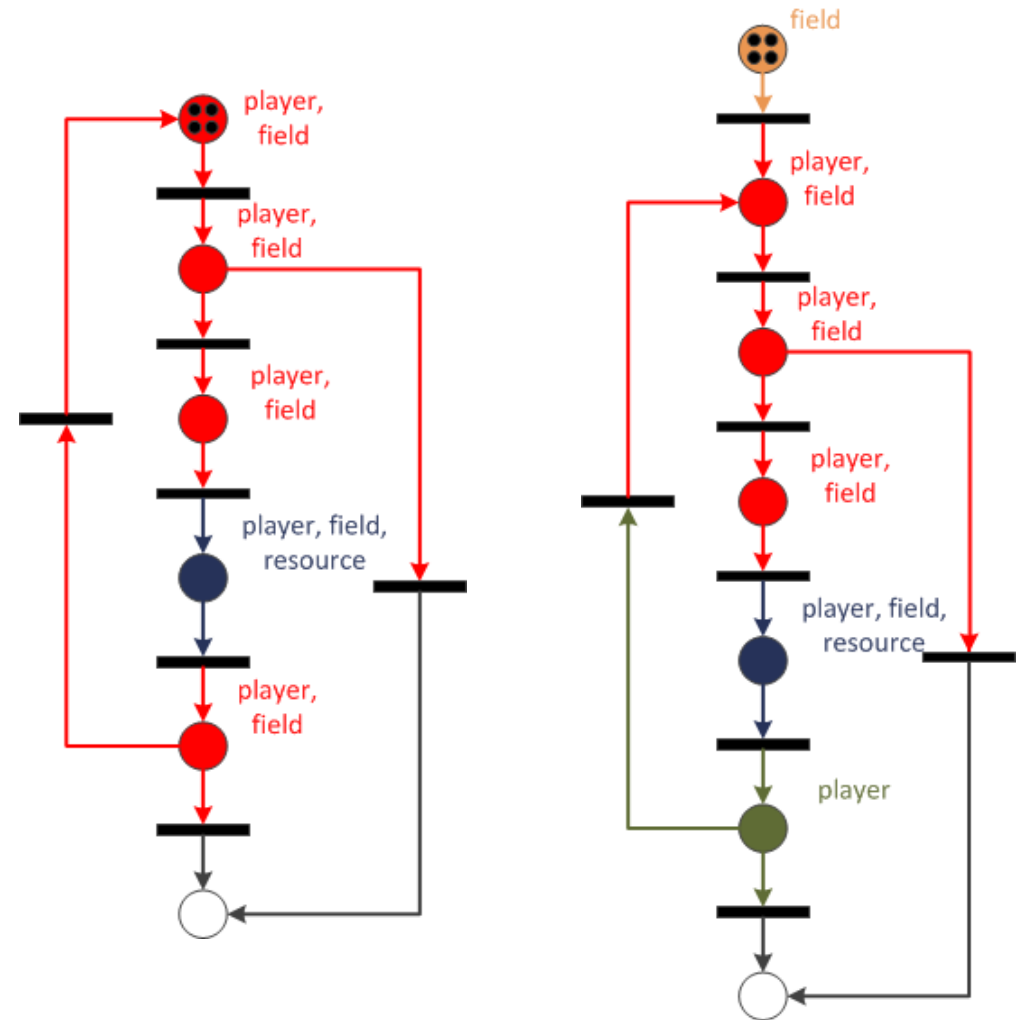
- неверная структура диаграммы;
- невозможность перехода из-за невыполнения логического условия спусковой функции.



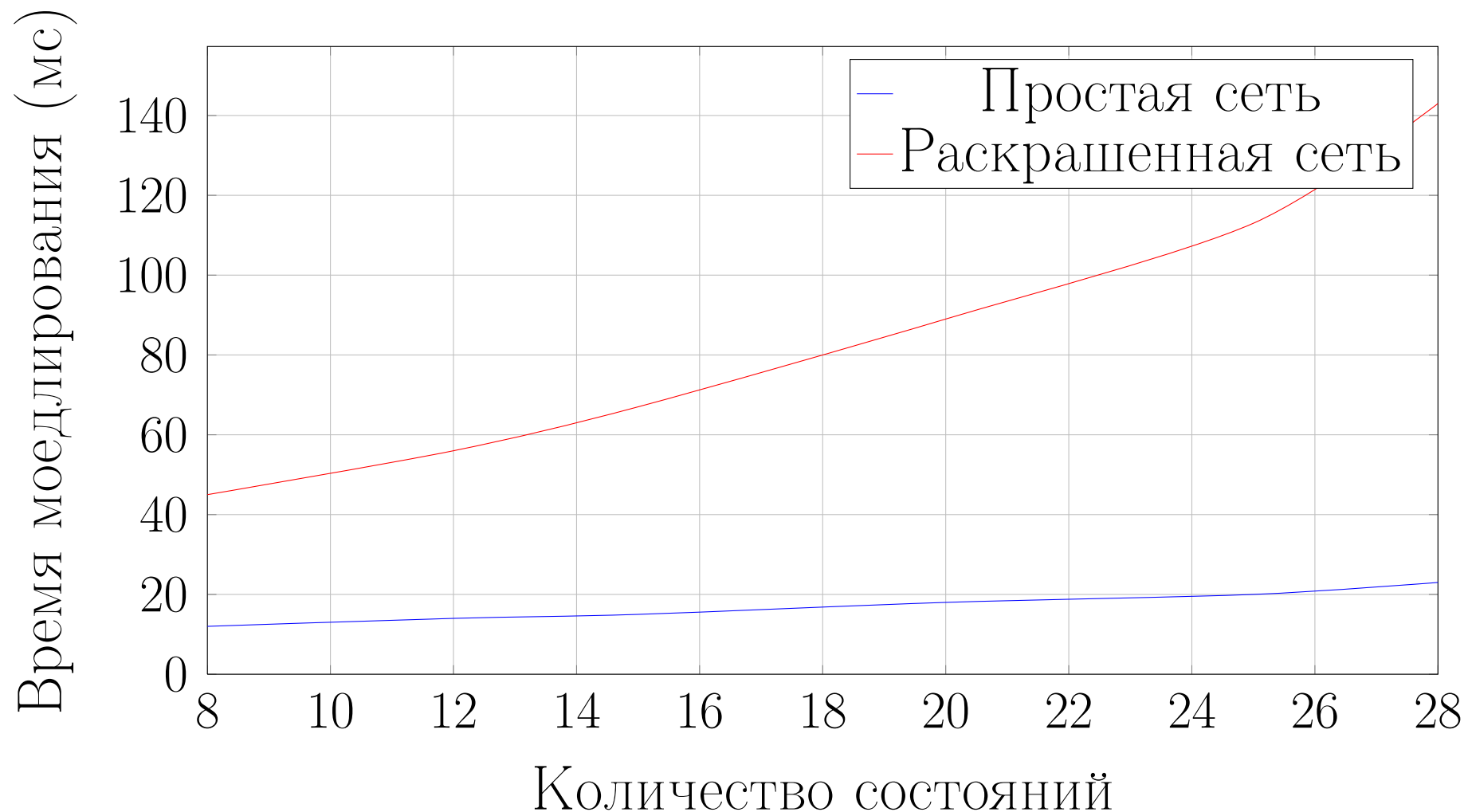
# Исследование корректности построения раскраски сети

Два подхода к построению раскраски:

1. определять раскраску всех элементов цикла как кортеж переменных наибольшей мощности;
2. определять раскраску не учитывая обратную дугу цикла.

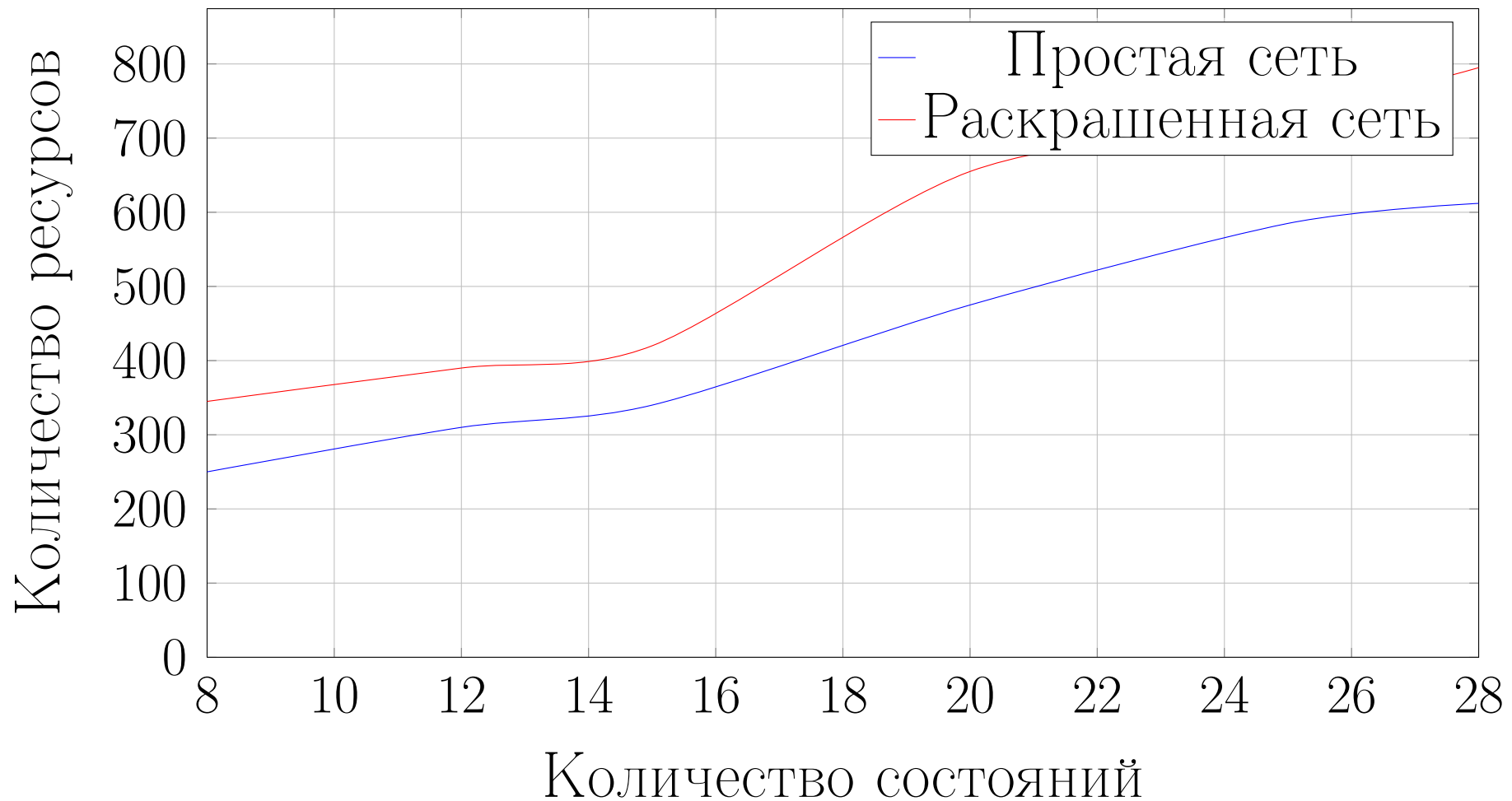


# Сравнение скорости работы простой и раскрашенной сети Петри





# Сравнение количества затраченных ресурсов для простой и раскрашенной сети Петри



# Выводы

1. Проведена классификация существующих методов анализа диаграмм деятельности.
2. Разработан метод представления диаграммы деятельности в виде раскрашенной сети Петри.
3. Разработано программное обеспечение, реализующее предложенный метод.
4. Исследованы факторы, влияющие на появление блокировок.
5. Исследована корректность построения раскраски сети.