**Слайд 1. Титульник.**

Часто алгоритм функциионирования сложных систем представляют в виде диаграммы деятельности UML, отличающейся от классической блок-схемы наличием элементов для представления многопоточной обработки. При этом в многопоточных системах встают вопросы о достижимости состояний и наличии блокировок. Для ре­ шения этих вопросов часто диаграмму деятельности представляют в виде простой сети Петри, при этом моделируется лишь процесс выполнения без привязки к дан­ ным. В отличие от простых сетей Петри, в раскрашенных сетях немаловажную роль играет типизация данных, основанная на понятии множества цветов, которое анало­ гично типу в декларативных языках программирования. Таким образом, представив диаграмму в виде раскрашенной сети Петри можно провести моделирование работы с учетом типов входных данных, что максимально приблизит процесс моделирования к реальному функционированию процесса.

**Слайд 2. Цели и решаемые задачи.**

Целью работы является исследование, разработка и реализация метода представления диаграммы деятельности в виде раскрашенной сети Петри, позволяющего выявить блокировки и недостижимые состояния.

Решаемые задачи:

1. Провести обзор и классификацию существующих подходов к анализу диаграмм деятельности.
2. Разработать метод представления диаграммы деятельности в виде раскрашенной сети Петри.
3. Программно реализовать предложенный метод.
4. Исследовать факторы, влияющие на появление блокировок.

**Слайд 3. Классификация методов анализа диаграмм деятельности.**

Часто для анализа используется представление диаграмм в виде конечного автомата. Такой подход накладывает серьезные ограничения, т.к. с помощью автоматов невозможно представить блоки многопоточной обработки.

Исполняемый UML представляет собой модель системы с описанием ее действий (обычно в виде конечных автоматов), что позволяет по этой модели выстроить рабочий программный код.

Анализ, основанный на простой сети Петри позволяет выявить проблемы, связанные с самой структурой диаграммы, в то время, как раскрашенная сеть позволяет провести моделирование работы системы с привязкой к данным.

**Слайд 4. Функциональная модель системы.**

Функциональная модель системы представлена на слайде.

**Слайд 5. Представление UML диаграмм.**

Основным стандартом описания UML диаграмм является XMI. Для гибкости модели связи между вершинами не описываются напрямую. Вместо этого для каждой вершины приписывается список входных и выходных переходов. Каждый переход содержит информацию о исходящей и входящей вершине и логическое условие для срабатывания перехода.

**Слайд 6. Отображение диаграммы деятельности.**

Исходя из специфики диаграммы деятельности можно представить ее в виде ориентированного графа. Тогда для ее изображения будем использовать алгоритм полилинейного отображения планарного графа. Алгоритм состоит из трех этапов.

**Слайд 7. Построение ассоциированного орграфа.**

На первом этапе необходимо построить ассоциированный с исходным графом граф G\*. Онпредставляет собой граф связности граней исходного графа. Для его построения необходим получить список граней графа G. Будем понимать под гранью простой цикл графа, не содержащий в себе других циклов.

Топологическая сортировка представляет собой алгоритм упорядочивания вершин бесконтурного ориентированного графа согласно частичному порядку, заданному ребрами орграфа на множестве его вершин.

**Слайд 8. Мозаичное представление графа.**

Мозаичное представление графа – это такое его изображение, в котором каждая вершина, ребро и грань изображается с помощью плитки – прямоугольника, стороны которого параллельны осям координат.

На основе мазаичного представления легко конструируется полилинейное отображение графа. Для этого каждая вершина G может быть представлена некоторой точкой соответствующего вершинного отрезка, а каждая дуга (p, q) графа G ломанной, среднее звено которого образовано частью реберного отрезка, изображающего дугу.

На слайде слево приведено изображение исходного графа и ассоциированного с ним орграфа G\*. Справо представлено его полилинейное изображение.

**Слайд 9. Раскрашенные сети Петри.**

Раскрашенная сеть Петри представляет собой направленный граф с двумя типами вершин — позициями и переходами, при этом дуги не могут соединять вершины одного типа, т.е. граф является двудольным.

Множество позиции описывают состояния системы. Каждая позиция имеет определенное значение из множества цветов.

Переходы описывают условия изменения состоянии. Переход может сработать, если во всех входных позициях для данного перехода присутствует хотя бы одна фишка и выполнено логическое выражение, ограничивающее переход.

**Слайд 10. Этапы построения раскрашенной сети Петри.**

Этапы преобразования диаграммы деятельности в раскрашенную сеть Петри приведены на слайде.

**Слайд 11-13. Преобразование диаграммы деятельности в раскрашенную сеть Петри.**

Процесс преобразования диаграммы деятельности в раскрашенную сеть Петри состоит из четырех этапов:

На первом этапе для каждого элемента диаграммы выделяется список переменных и определяется структура каждой переменной.

На втором этапе вводятся начальные значения и определяется тип переменных, строиться предварительное множество раскрасок на основе списков переменных в каждой вершине.

На третьем этапе производится построение сети Петри по структуре диаграммы.

На четвертом этапе для каждой переменной определяется максимальная область ее видимости и с учетом этого формируется результирующая раскраска сети.

**Слайд 14.**

Основной задачей исследования сетей Петри является выявление недоступных состояний и блокировок. Выявим причины, по которым они могут возникнуть в диаграммах деятельности. В процессе исследования представим диаграмму деятельности в виде простой и раскрашенной сети Петри и сравним результаты.

Блокировки в простых сетях Петри могут быть вызваны лишь ситуацией, когда переходу принадлежит два и более места, но не все места имеют фишки.

Ситуация блокировки, изображенная на сладйе искусственная, т.к. по логике выход за пределы параллельных процессов невозможен.

Ситуация, изображенная на слайде возможна, и в результате в системе возникает блокировка, т.к. переход никогда не сработает. Для избежания такой ситуации используется введение дополнительной вершины, из которой будет единственная дуга в блок синхронизации.

Блокировка в раскрашенных сетях может быть вызвана не только отсутствием фишки в позиции, но и не срабатыванием спусковой функции из-за невыполнения логического условия. По этой причине, раскрашенная сеть Петри не всегда способна выявить некорректную структуру диаграммы, т.к. при конкретной начальной разметке, для которой построена сеть, не все переходы будут активны.

**Слайд 15. Выводы.**

1. Проведен оброз существующих методов анализа диаграмм деятельности и осущствленно их сравнение по набору критериев.
2. Разработан метод представления диаграммы деятельности в виде раскрашенной сети Петри.
3. Разработано программное обеспечение, реализующее предложенный метод.
4. Исследованы факторы, влияющие на точность и правильность представления диаграммы в виде раскрашенной сети Петри.