

Отчёт по лабораторной работе №10

Имитационное моделирование

Ганина Таисия Сергеевна, НФИбд-01-22

Содержание

| | | |
|----------|---------------------------------------|-----------|
| 1 | Цель работы | 5 |
| 2 | Задание | 6 |
| 3 | Теоретическое введение | 7 |
| 4 | Выполнение лабораторной работы | 8 |
| 4.1 | Упражнение | 12 |
| 5 | Выводы | 19 |
| | Список литературы | 20 |

Список иллюстраций

| | | |
|-----|---|----|
| 4.1 | Граф сети задачи об обедающих мудрецах | 9 |
| 4.2 | Задание деклараций задачи об обедающих мудрецах | 10 |
| 4.3 | Модель задачи об обедающих мудрецах | 11 |
| 4.4 | Запуск модели задачи об обедающих мудрецах | 12 |
| 4.5 | Пространство состояний для модели | 17 |
| 4.6 | Пространство состояний для модели | 17 |

Список таблиц

1 Цель работы

Реализовать модель задачи об обедающих мудрецах в CPN Tools.

2 Задание

- Реализовать модель задачи об обедающих мудрецах в CPN Tools;
- Вычислить пространство состояний, сформировать отчет о нем и построить граф.

3 Теоретическое введение

CPN Tools — специальное программное средство, предназначенное для моделирования иерархических временных раскрашенных сетей Петри. Такие сети эквивалентны машине Тьюринга и составляют универсальную алгоритмическую систему, позволяющую описать произвольный объект. CPN Tools позволяет визуализировать модель с помощью графа сети Петри и применить язык программирования CPN ML (Colored Petri Net Markup Language) для формализованного описания модели.

Назначение CPN Tools:

- разработка сложных объектов и моделирование процессов в различных прикладных областях, в том числе:
- моделирование производственных и бизнес-процессов;
- моделирование систем управления производственными системами и роботами;
- спецификация и верификация протоколов, оценка пропускной способности сетей и качества обслуживания, проектирование телекоммуникационных устройств и сетей.

Основные функции CPN Tools:

- создание (редактирование) моделей;
- анализ поведения моделей с помощью имитации динамики сети Петри;
- построение и анализ пространства состояний модели.

[1,2].

4 Выполнение лабораторной работы

Пять мудрецов сидят за круглым столом и могут пребывать в двух состояниях – думать и есть. Между соседями лежит одна палочка для еды. Для приёма пищи необходимы две палочки. Палочки – пересекающийся ресурс. Необходимо синхронизировать процесс еды так, чтобы мудрецы не умерли с голода.

Рисуем граф сети. Для этого с помощью контекстного меню создаём новую сеть, добавляем позиции, переходы и дуги (рис. 4.1).

Начальные данные:

- позиции: мудрец размышляет (philosopher thinks), мудрец ест (philosopher eats), палочки находятся на столе (sticks on the table)
- переходы: взять палочки (take sticks), положить палочки (put sticks)

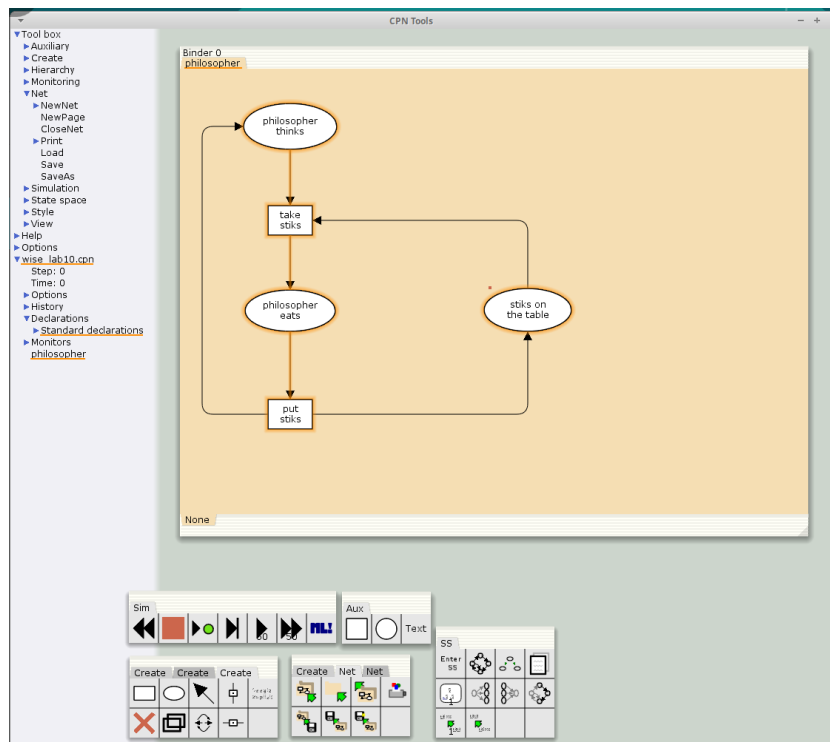


Рис. 4.1: Граф сети задачи об обедающих мудрецах

В меню задаём новые декларации модели (рис. 4.2): типы фишек, начальные значения позиций, выражения для дуг:

- n — число мудрецов и палочек ($n = 5$);
- p — фишки, обозначающие мудрецов, имеют перечисляемый тип PH от 1 до n ;
- s — фишки, обозначающие палочки, имеют перечисляемый тип ST от 1 до n ;
- функция $\text{ChangeS}(p)$ ставит в соответствие мудрецам палочки (возвращает номера палочек, используемых мудрецами); по условию задачи мудрецы сидят по кругу и мудрец $p(i)$ может взять i и $i + 1$ палочки, поэтому функция $\text{ChangeS}(p)$ определяется следующим образом:

```
fun ChangeS (ph(i))=
1`st(i)++st(if = n then 1 else i+1)
```

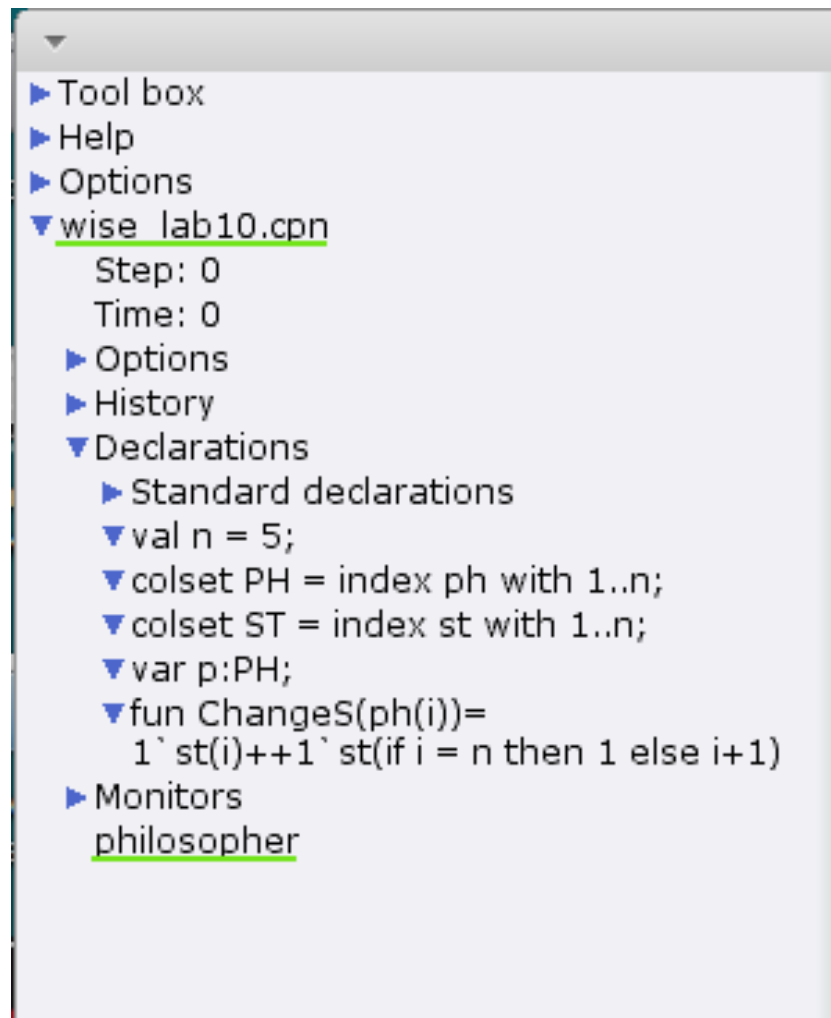


Рис. 4.2: Задание деклараций задачи об обедающих мудрецах

В результате получаем работающую модель (рис. 4.3).

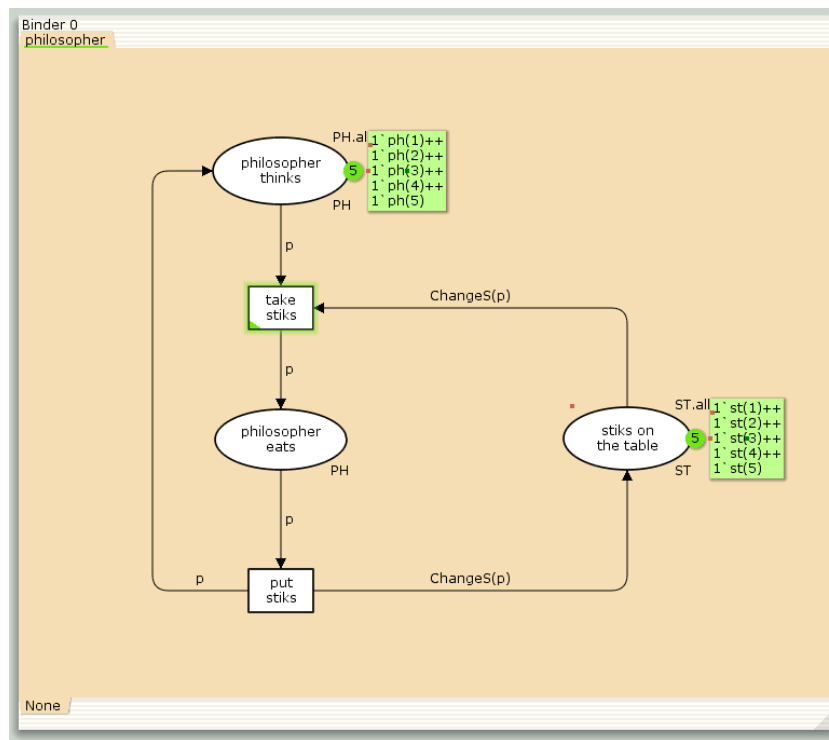


Рис. 4.3: Модель задачи об обедающих мудрецах

После запуска модели наблюдаем, что одновременно палочками могут воспользоваться только два из пяти мудрецов (рис. 4.4).

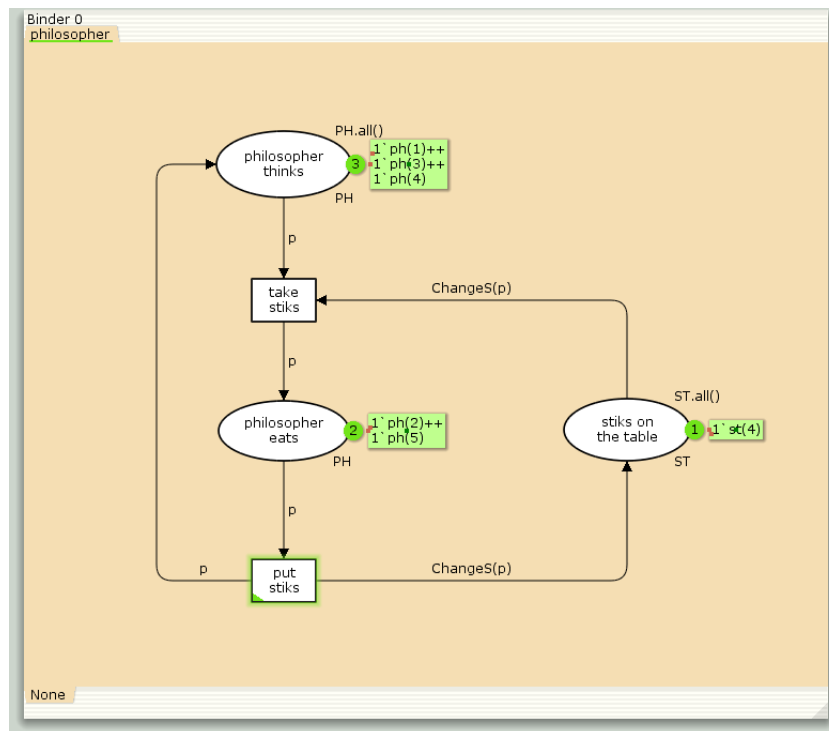


Рис. 4.4: Запуск модели задачи об обедающих мудрецах

4.1 Упражнение

Вычислим пространство состояний. Прежде, чем пространство состояний может быть вычислено и проанализировано, необходимо сформировать код пространства состояний. Этот код создается, когда используется инструмент Войти в пространство состояний. Вход в пространство состояний занимает некоторое время. Затем, если ожидается, что пространство состояний будет небольшим, можно просто применить инструмент Вычислить пространство состояний к листу, содержащему страницу сети. Сформируем отчет о пространстве состояний и проанализируем его. Чтобы сохранить отчет, необходимо применить инструмент Сохранить отчет о пространстве состояний к листу, содержащему страницу сети и ввести имя файла отчета.

Из отчета можем узнать, что:

- есть 11 состояний и 30 переходов между ними;
- Границы значений:
 - Одновременно едят от 0 до 2 мудрецов (максимум при использовании 4 палочек из 5);
 - Размышляют от 3 до 5 мудрецов;
 - На столе остаётся от 1 до 5 палочек;
- указаны границы в виде мультимножеств;
- маркировка home для всех состояний, что означает возможность возврата в любое состояние;
- маркировка dead равна None;
- события «взять/положить палочки» происходят бесконечно часто с равной вероятностью (impartial);
- максимальное число одновременно используемых палочек — 4 (при двух едящих мудрецах);

Эта модель показывает, как пять мудрецов могут делить пять палочек так, чтобы никто не остался голодным и система работала без остановок. Одновременно едят только два человека — остальные размышляют. Палочки распределяются справедливо, а действия происходят циклично: взял палочки - поел - положил палочки - подумал - снова взял палочки.

Отчёт:

CPN Tools state space report for:

/home/openmodelica/wise_lab10.cpn

Report generated: Sat Mar 29 15:17:55 2025

Statistics

State Space

Nodes: 11
Arcs: 30
Secs: 0
Status: Full

Scc Graph

Nodes: 1
Arcs: 0
Secs: 0

Boundedness Properties

Best Integer Bounds

| | Upper | Lower |
|----------------------------------|-------|-------|
| philosopher'philosopher_eats 1 | 2 | 0 |
| philosopher'philosopher_thinks 1 | 5 | 3 |
| philosopher'stiks_on_the_table 1 | 5 | 1 |

Best Upper Multi-set Bounds

philosopher'philosopher_eats 1
1`ph(1)++
1`ph(2)++
1`ph(3)++

```

1`ph(4)++
1`ph(5)
    philosopher'philosopher_thinks 1
        1`ph(1)++

1`ph(2)++
1`ph(3)++
1`ph(4)++
1`ph(5)
    philosopher'stiks_on_the_table 1
        1`st(1)++

1`st(2)++
1`st(3)++
1`st(4)++
1`st(5)

```

Best Lower Multi-set Bounds

```

    philosopher'philosopher_eats 1
        empty
    philosopher'philosopher_thinks 1
        empty
    philosopher'stiks_on_the_table 1
        empty

```

Home Properties

Home Markings

All

Liveness Properties

Dead Markings

None

Dead Transition Instances

None

Live Transition Instances

All

Fairness Properties

philosopher'put_stiks 1

Impartial

philosopher'take_stiks 1

Impartial

Построим граф пространства состояний (рис. 4.5, 4.6):

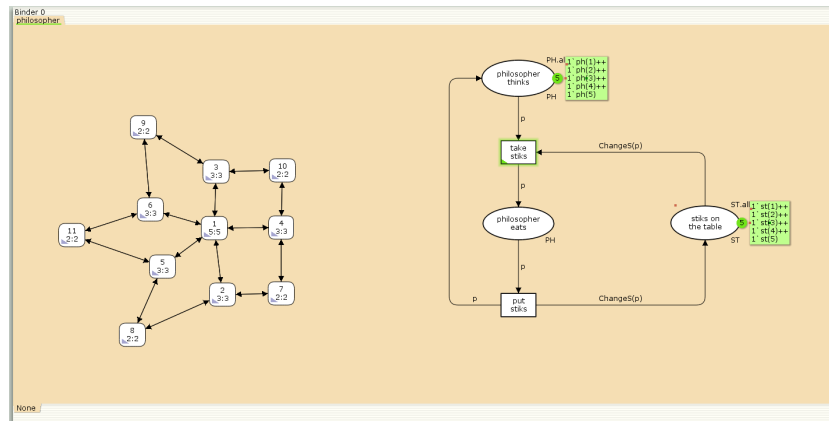


Рис. 4.5: Пространство состояний для модели

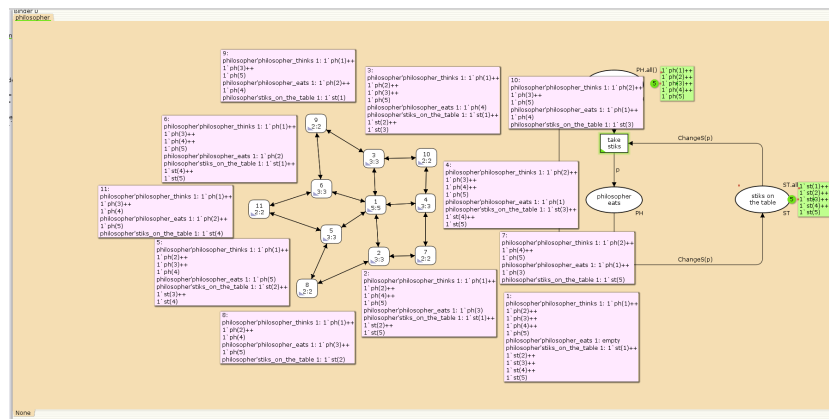


Рис. 4.6: Пространство состояний для модели

На представленном изображении показан граф пространства состояний для задачи об обедающих мудрецах.

Общая структура пространства состояний: - Количество состояний: 11 узлов (пронумерованы от 1 до 11); - У нас всего 15 стрелок, но так как они двунаправленные, получается в итоге 30 переходов. Они представляют собой переходы между состояниями, вызванные срабатыванием переходов take_stiks и put_stiks.

- **Состояние 1 (5:5):** Начальное состояние, в котором все мудрецы размышляют, а все палочки находятся на столе.

– philosopher_thinks: 1`ph(1)++1`ph(2)++1`ph(3)++1`ph(4)++1`ph(5)

- philosopher_eats: пуст
- stiks_on_the_table: 1`st(1)++1`st(2)++1`st(3)++1`st(4)++1`st(5)

- **Состояние 2 (3:3):** Мудрец 3 ест, остальные думают, три палочки лежат на столе.

- philosopher_thinks: 1`ph(1)++1`ph(2)++1`ph(4)++1`ph(5)
- philosopher_eats: 1`ph(3)
- stiks_on_the_table: 1`st(1)++1`st(4)++1`st(5)

- **Состояние 6 (3:3):** Мудрец 2 ест, остальные думают, три палочки лежат на столе.

- philosopher_thinks: 1`ph(1)++1`ph(3)++1`ph(4)++1`ph(5)
- philosopher_eats: 1`ph(2)
- stiks_on_the_table: 1`st(1)++1`st(4)++1`st(5)

<...>

- **Состояние 7 (3:3):** Два мудреца едят, три думают, одна палочка на столе.

- philosopher_thinks: 1`ph(1)++1`ph(4)++1`ph(5)
- philosopher_eats: 1`ph(2)++1`ph(3)
- stiks_on_the_table: 1`st(1)

И так далее. Состояния 2-6 описывают моменты, когда есть один мудрец, состояния 7-11 перебирают “комбинации мудрецов”, когда едят одновременно два мудреца.

5 Выводы

В ходе 10 лабораторной работы была реализована модель задачи об обедающих мудрецах в CPN Tools, вычислено пространство состояний, сформирован отчет о нем и построен граф.

Список литературы

1. Цветные сети Петри и язык распределенного программирования UPL: их сравнение и перевод, Аркадий Валентинович Климов [Электронный ресурс]. URL: https://psta.psiras.ru/read/psta2023_4_91-122.pdf.
2. CPN Tools, Michael Westergaard, August 2010, Eindhoven, Netherlands [Электронный ресурс]. URL: <https://westergaard.eu/wp-content/uploads/2010/09/CPN-Tools.pdf>.