

Лабораторная работа № 10

Задача об обедающих мудрецах

Дворкина Ева Владимировна

Содержание

| | | |
|----------|--|-----------|
| 1 | Цель работы | 4 |
| 2 | Задание | 5 |
| 3 | Теоретическое введение | 6 |
| 4 | Выполнение лабораторной работы | 7 |
| 4.1 | Задача об обедающих мудрецах | 7 |
| 4.2 | Упражнение | 10 |
| 5 | Выводы | 15 |
| | Список литературы | 16 |

Список иллюстраций

| | | |
|-----|--|----|
| 4.1 | Граф сети задачи об обедающих мудрецах | 8 |
| 4.2 | Задание деклараций задачи об обедающих мудрецах | 9 |
| 4.3 | Модель задачи об обедающих мудрецах | 9 |
| 4.4 | Запуск модели задачи об обедающих мудрецах | 10 |
| 4.5 | Пространство состояний для модели задачи об обедающих мудрецах | 10 |

1 Цель работы

Цель данной лабораторной 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 [1].

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.

4 Выполнение лабораторной работы

4.1 Задача об обедающих мудрецах

Постановка задачи

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 [2].

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)

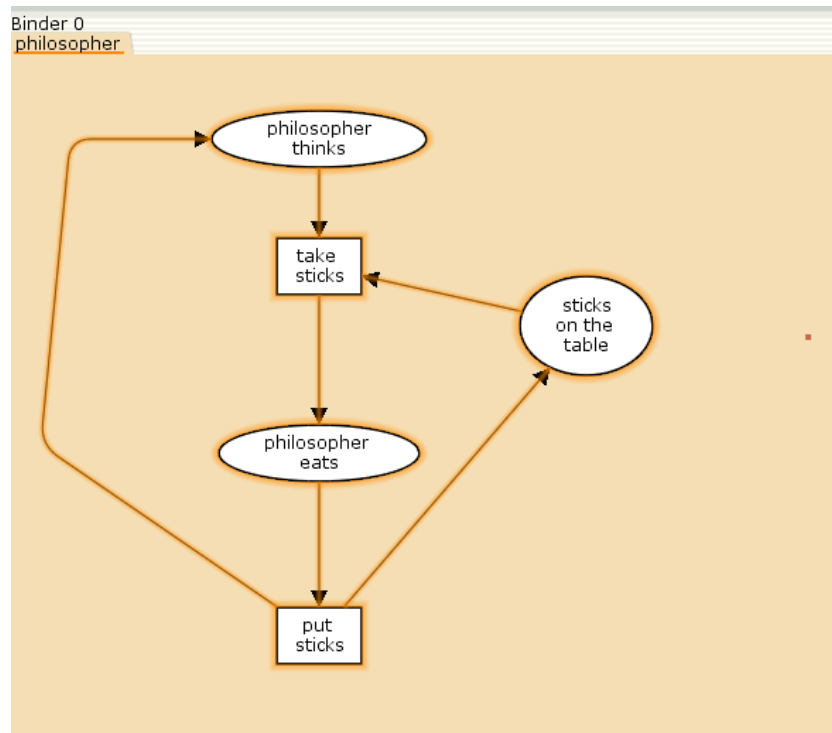


Рис. 4.1: Граф сети задачи об обедающих мудрецах

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)

```

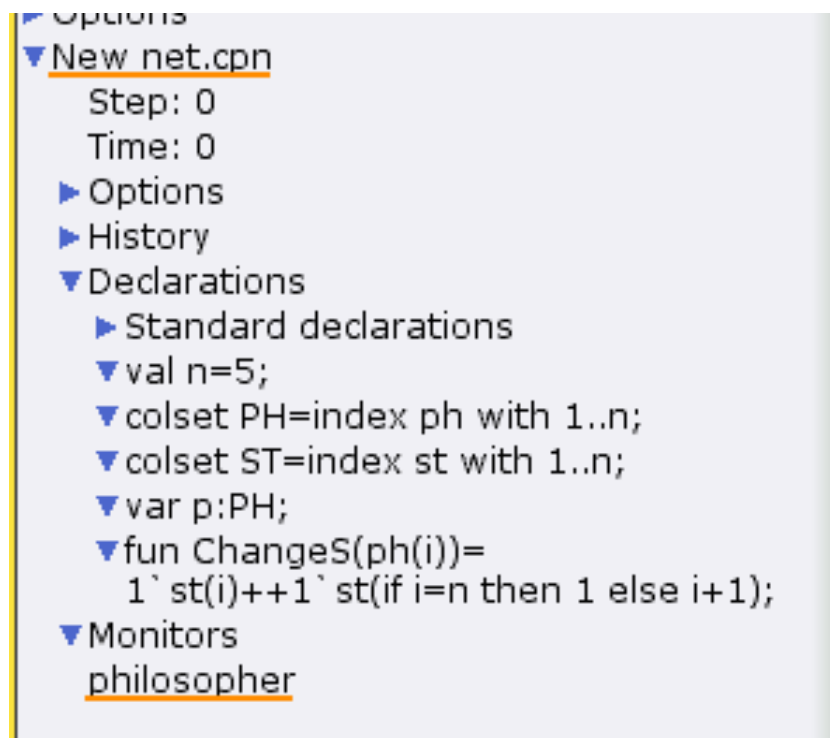



Рис. 4.2: Задание деклараций задачи об обедающих мудрецах

В результате получаем работающую модель (рис. 4.3).

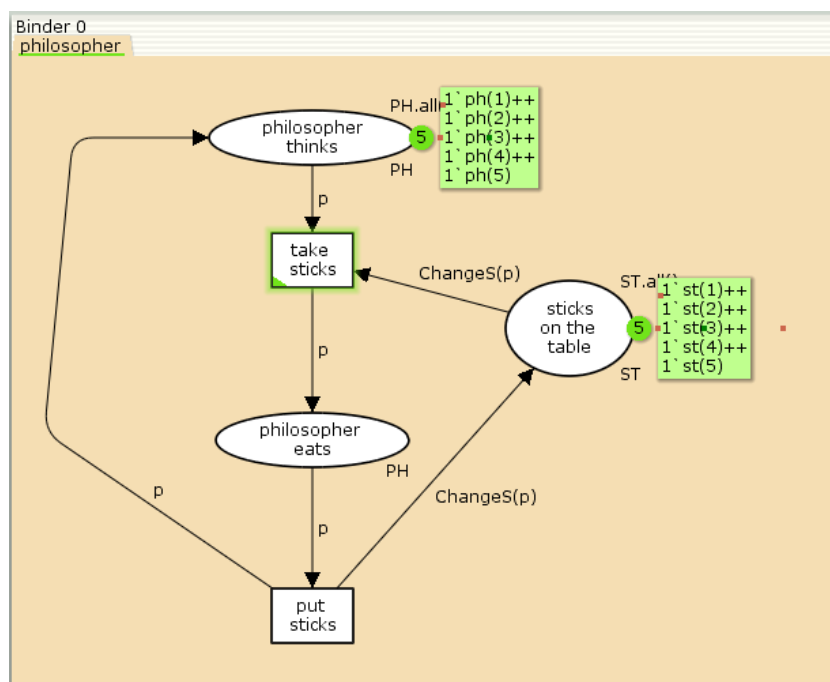


Рис. 4.3: Модель задачи об обедающих мудрецах

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

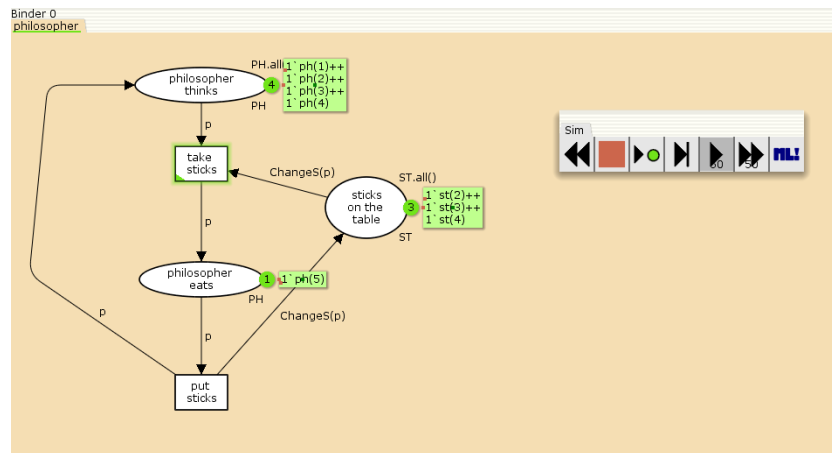


Рис. 4.4: Запуск модели задачи об обедающих мудрецах

4.2 Упражнение

Сформируем граф пространства состояний, всего их 11 (рис. 4.5):

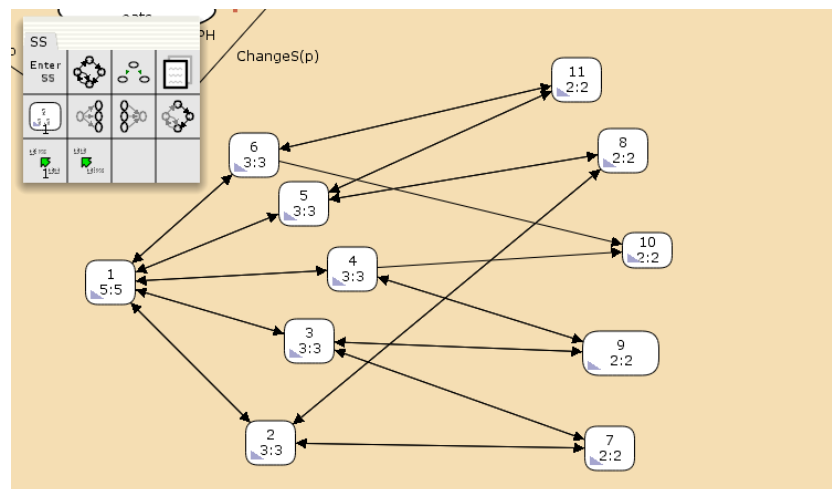


Рис. 4.5: Пространство состояний для модели задачи об обедающих мудрецах

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 отчёт 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 увидеть:

- есть 11 состояний и 30 переходов между ними.
- Затем указаны границы значений для каждого элемента: думающие мудрецы (максимум - 5, минимум - 3), мудрецы едят (максимум - 2, минимум - 0), палочки на столе (максимум - 5, минимум - 1, минимальное значение 2, так как в конце симуляции остаются пирожки).
- Также указаны границы в виде мультимножеств.
- Маркировка home для всех состояний, так как в любую позицию мы можем попасть из любой другой маркировки.
- Маркировка dead равная None, так как нет состояний, из которых переходов быть не может.
- В конце указано, что бесконечно часто происходят события положить и взять палочку.

CPN Tools state space report for:

/home/openmodelica/philosopher.cpn

Report generated: Mon Mar 3 04:21:10 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'sticks_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'sticks_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'sticks_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_sticks 1

Impartial

philosopher'take_stiicks 1

Impartial

5 Выводы

При выполнении данной лабораторной работы я реализовала модель задачи об обедающих мудрецах в CPN Tools.

Список литературы

1. Королькова А.В., Кулябов Д.С. Сети Петри. Моделирование в CPN Tools [Электронный ресурс].
2. Королькова А.В., Кулябов Д.С. Лабораторная работа 10. Задача об обедающих мудрецах [Электронный ресурс].