

Лабораторная работа №10

Имитационное моделирование

Серёгина Ирина Андреевна

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
5	Выводы	14

Список иллюстраций

4.1	Граф сети	8
4.2	Декларации модели	9
4.3	Работающая модель задачи об обедающих мудрецах	10
4.4	Граф пространства состояний	13

Список таблиц

1 Цель работы

Реализовать модель задачи об обедающих мудрецах с помощью CPNtools.

2 Задание

1. Построить модель задачи об обедающих мудрецах.
2. Выполнить упражнение, построить граф пространства состояний и сформировать отчёт о нём.

3 Теоретическое введение

Задача об обедающих мудрецах — классическая задача о блокировках и синхронизации процессов.

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.

4 Выполнение лабораторной работы

Рисую граф сети, для этого с помощью контекстного меню создаю новую сеть, добавляю 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)

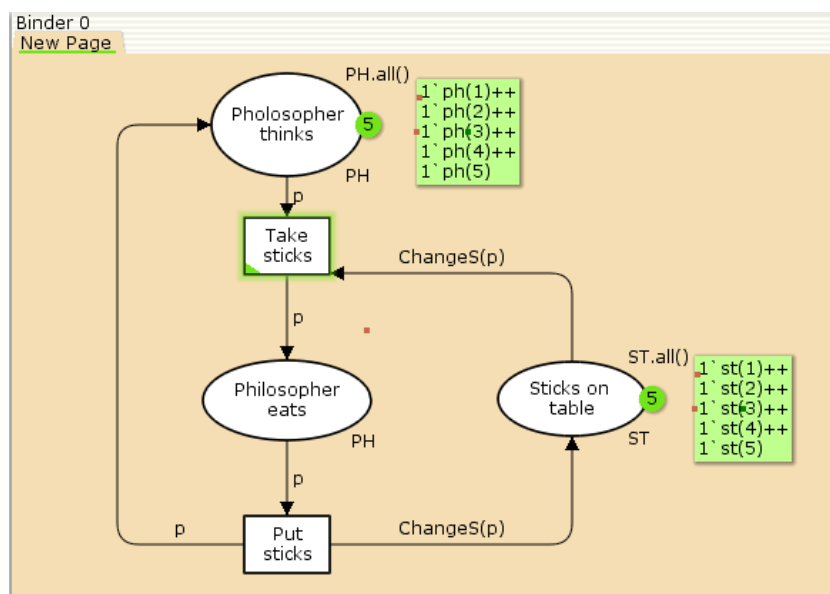
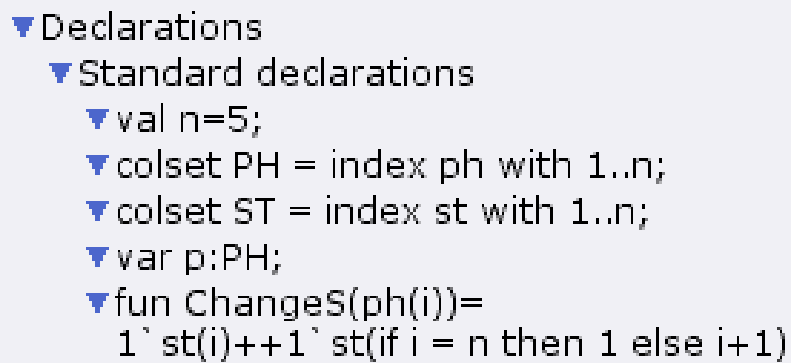


Рис. 4.1: Граф сети

В меню задаю 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)
```



```
▼ Declarations  
  ▼ Standard declarations  
    ▼ val n=5;  
    ▼ colset PH = index ph with 1..n;  
    ▼ colset ST = index st with 1..n;  
    ▼ var p:PH;  
    ▼ fun ChangeS(ph(i))=  
      1`st(i)++1`st(if i = n then 1 else i+1)
```

Рис. 4.2: Декларации модели

В результате получаю работающую модель, после запуска которой наблюдаю, что одновременно палочками могут воспользоваться только два из пяти мудрецов (рис. 4.3).

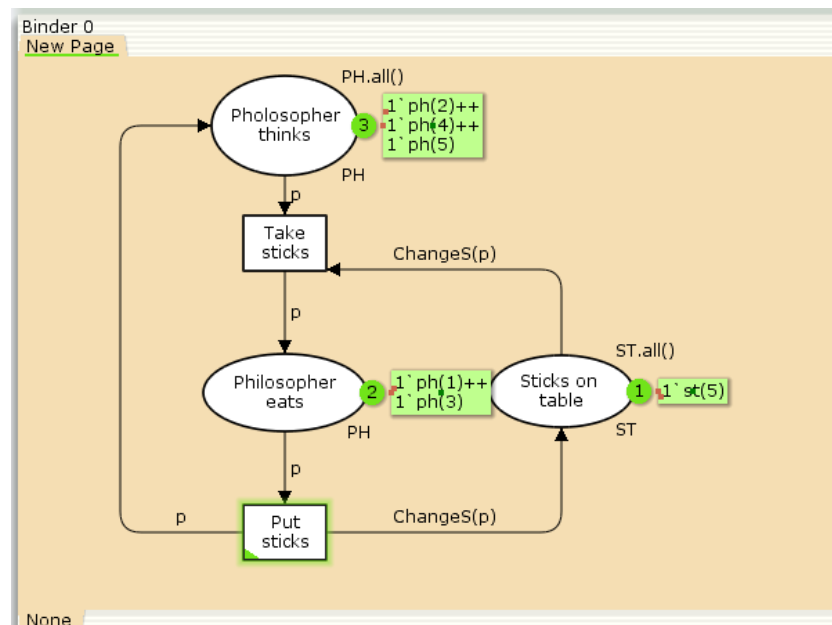


Рис. 4.3: Работающая модель задачи об обедающих мудрецах

После этого формирую отчёт о пространстве состояний из которого можно сделать выводы о том, что мы имеем 11 состояний и 30 переходов между ними, думающих мудрецов может быть максимум - 5, минимум - 3, обедающих мудрецов максимум - 2, минимум - 0, палочек на столе максимум - 5, минимум - 1. Также указано, что бесконечно часто происходят события положить и взять палочку.

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

Строю граф пространства состояний (рис. 4.4).

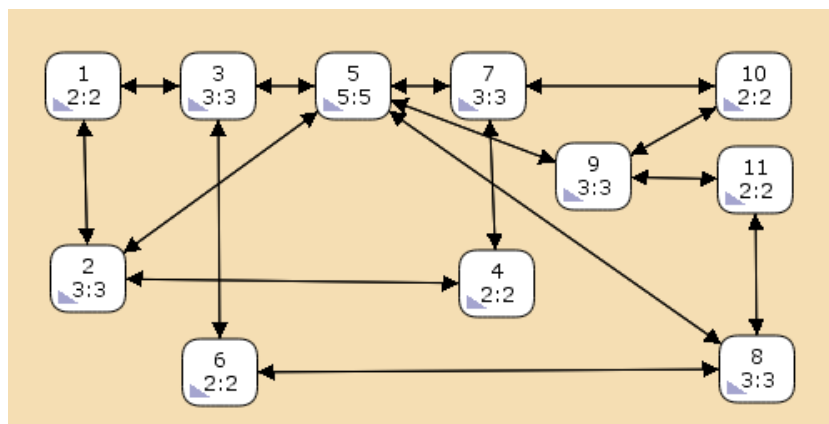


Рис. 4.4: Граф пространства состояний

5 Выводы

Я реализовала модель задачи об обедающих мудрецах с помощью CPNtools.