

Лабораторная работа 10

Задача об обедающих мудрецах

Лихтенштейн Алина Алексеевна

Содержание

1	Введение	4
2	Выполнение лабораторной работы	5
	2.1 Упражнение	8
3	Выводы	13

Список иллюстраций

2.1	Граф сети задачи об обедающих мудрецах	6
2.2	Задание деклараций задачи об обедающих мудрецах	7
2.3	Модель задачи об обедающих мудрецах	7
2.4	Запуск модели задачи об обедающих мудрецах	8
2.5	Граф пространства состояний	12

1 Введение

Цель работы

Реализовать модель задачи об обедающих мудрецах в CPN Tools.

Задание

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

2 Выполнение лабораторной работы

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.

Нарисуем граф сети. Для этого с помощью контекстного меню создадим новую сеть, добавим позиции, переходы и дуги (рис. 2.1).

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)

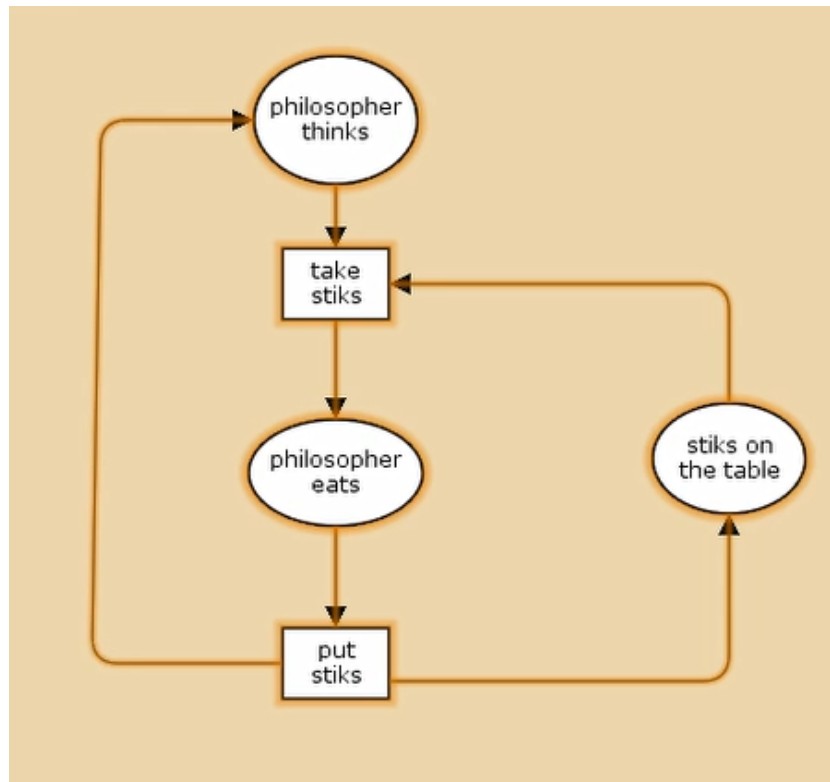


Рис. 2.1: Граф сети задачи об обедающих мудрецах

В меню зададим новые декларации модели (рис. 2.2): 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)

```

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

```

```

▼ petry philosopher.cpn
  Step: 0
  Time: 0
  ▶ Options
  ▶ History
  ▼ 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)
    ▶ Standard declarations
  ▶ Monitors
    philosopher

```

Рис. 2.2: Задание деклараций задачи об обедающих мудрецах

В результате получим работающую модель (рис. 2.3).

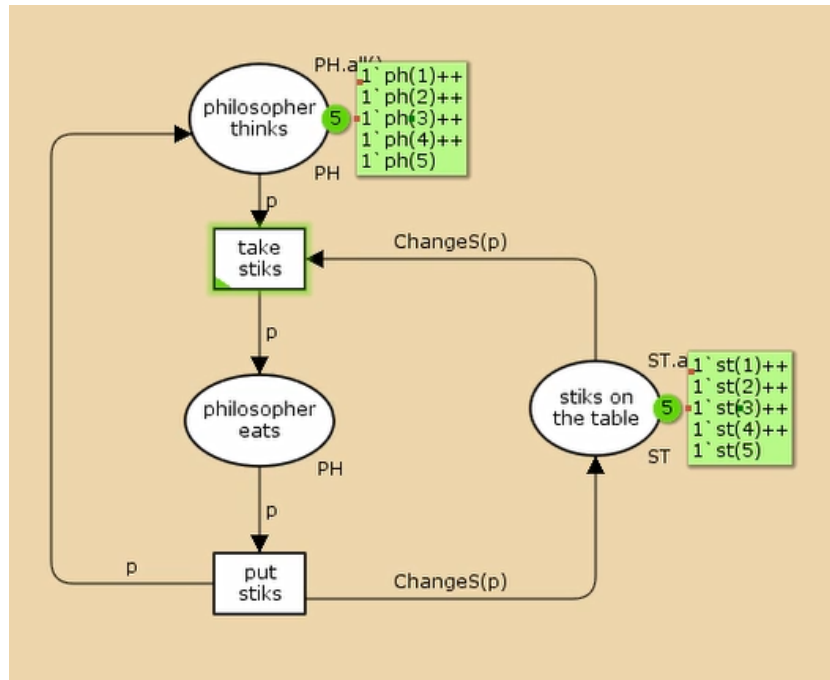


Рис. 2.3: Модель задачи об обедающих мудрецах

Заметим, что одновременно палочками могут воспользоваться только два из пяти мудрецов (рис. 2.4).

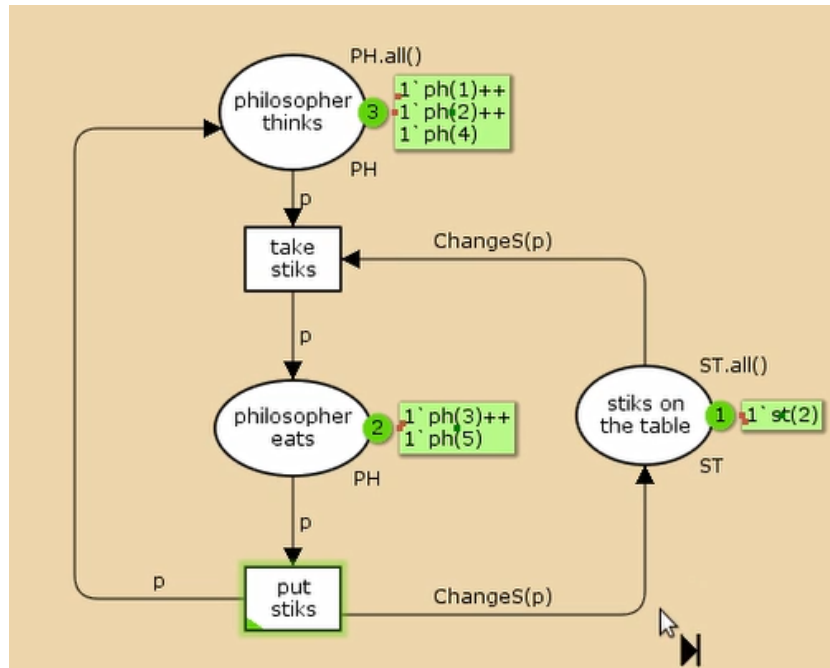


Рис. 2.4: Запуск модели задачи об обедающих мудрецах

2.1 Упражнение

Вычислим пространство состояний. Сформируем отчёт о пространстве состояний и проанализируем его.

Из отчета можем узнать, что:

- есть 11 состояний и 30 переходов между ними;
- 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);
- указаны границы в виде мультимножеств;
- маркировка home для всех состояний;

- маркировка dead равна None;
- указано, что бесконечно часто происходят события положить и взять палочку.

CPN Tools state space report for:

/home/openmodelica/philosopher.cpn

Report generated: Sat May 25 00:45:34 2024

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's_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_sticks 1
 Impartial

Построим граф пространства состояний (рис. 2.5).

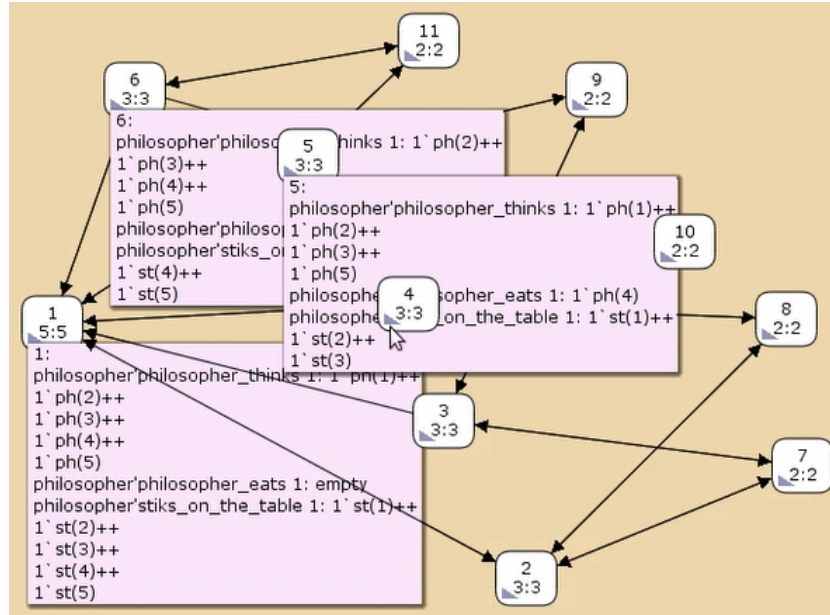


Рис. 2.5: Граф пространства состояний

3 Выводы

В процессе выполнения данной лабораторной работы была реализована модель задачи об обедающих мудрецах в CPN Tools.