Санкт-Петербургский Политехнический университет
Петра Великого
Институт прикладной математики и механики
Высшая школа прикладной математики и вычислительной физики

КУРСОВАЯ РАБОТА
Разработка контроллера светофоров и его верификация

по дисциплине
"Верификация распределённых алгоритмов и протоколов"

Выполнил:
студент гр.3640102/00201 **Ланcков.Н.В.**
Преподаватель:
к.т.н, доцент ВШПИ ИКНТ **Шошмина И.В**

Санкт-Петербург
2020

Содержание

1 Список иллюстраций

2 Список таблиц

3 Постановка задачи

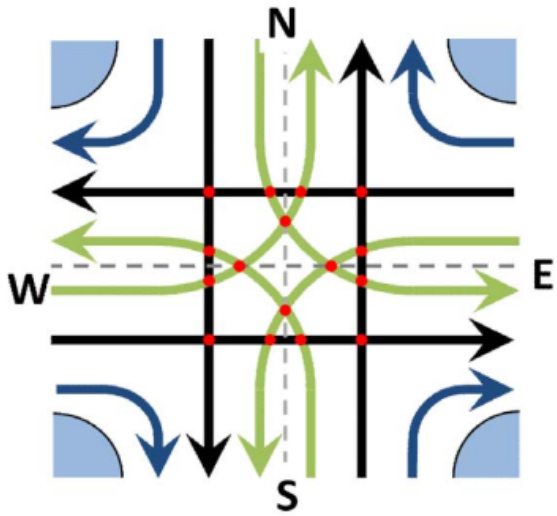
Дан перекрёсток с четырьмя двухсторонними направлениями движения. В каждом направлении имеются три полосы. Точная схема перекрёстка задаётся пересечениями направлений движения, указанными в таблице 1

Вариант: 12, 13, 15

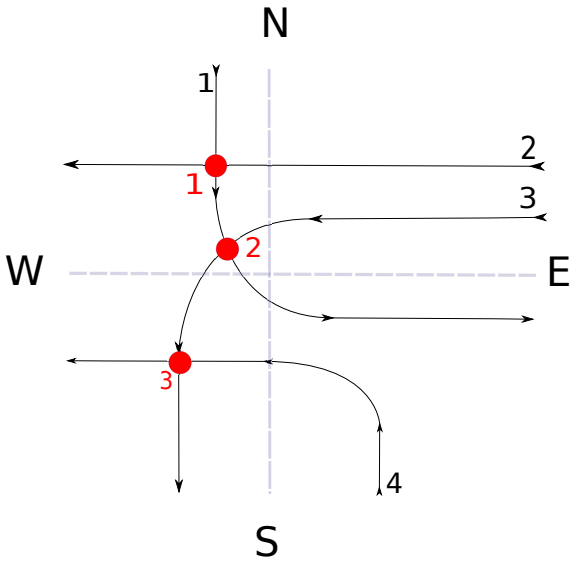
Вариант	Пересечение	Вариант	Пересечение
1	WN, NS	9	SN, WE
2	WN, NE	10	SN, EW
3	WN, SW	11	SN, ES
4	WN, EW	**12**	**NE, EW**
5	NS, SW	**13**	**NE, ES**
6	NS, WE	14	SW, WE
7	NS, EW	**15**	**SW, ES**
8	SN, NE	16	WE, ES

Таблица 1: Варианты пересечений

Для наглядности также привожу схематическое изображение полос движения и пересечений для своего варианта 1.



(a) Схема всех направлений движения



(b) Схема направлений и пересечерний для текущего варианта

Рис. 1: Перекрёсток

Каждое направление движения регулируется своим светофором. Если машин нет - светофор горит красным светом. Если машины есть - светофор загорается зелёным (как только появится такая возможность) и пропускает все машины, после чего снова загорается красным. Требуется разработать модель контроллера светофоров, которая бы удовлетворяла следующим свойствам, заданным в виде ltl формул.

Также в модели требуется отразить поведение внешней среды.

Листинг 1: Формулы линейной темпоральной логики, отражающие требования к системе

```
#define crash_1 (traffic_lights_color[0] == GREEN && traffic_lights_color[1] =
#define crash_2 (traffic_lights_color[0] == GREEN && traffic_lights_color[2] =
#define crash_3 (traffic_lights_color[2] == GREEN && traffic_lights_color[3] =

#define car_sense_0 (len(car_sensor[0]) > 0)
#define car_sense_1 (len(car_sensor[1]) > 0)
#define car_sense_2 (len(car_sensor[2]) > 0)
#define car_sense_3 (len(car_sensor[3]) > 0)

#define tl_green_0 (traffic_lights_color[0] == GREEN)
#define tl_green_1 (traffic_lights_color[1] == GREEN)
#define tl_green_2 (traffic_lights_color[2] == GREEN)
#define tl_green_3 (traffic_lights_color[3] == GREEN)

/* LTL formulae descriptions */

/* Safety */
[] (!crash_1)
[] (!crash_2)
[] (!crash_3)

/* Liveliness */
 [] (car_sense_0 -> <> tl_green_0)
 [] (car_sense_1 -> <> tl_green_1)
 [] (car_sense_2 -> <> tl_green_2)
 [] (car_sense_3 -> <> tl_green_3)

/* Fairness */
[] <> !(tl_green_0 && car_sense_0)
[] <> !(tl_green_1 && car_sense_1)
[] <> !(tl_green_2 && car_sense_2)
[] <> !(tl_green_3 && car_sense_3)
```

4 Построение модели

4.1 Внешняя среда

Внешняя среда моделируется при помощи процессов, которые генерируют потоки машин по каждому из доступных направлений. Каждый такой процесс посылает в бесконечном цикле сообщение о прибывших машинах в канал ёмкостью 1, и затем блокируется до тех пор, пока это сообщение не будет обработано контроллером светофора, обслуживающего соответствующее направление движения.

4.2 Дорожные пересечения

Дорожные пересечения моделируются процессами, которые блокируются и разблокируются процессами светофоров.

4.3 Светофоры

Светофоры также моделируются соответствующими процессами. Суть работы светофора - при поступлении сообщения о наличии машин:

1. Заблокировать соответствующие пересечения
2. Пропустить поток машин
3. Разблокировать "захваченные"пересечения в обратном порядке

4.4 Взаимодействие процессов

Таким образом, модель состоит из процессов, моделирующих поведение светофоров, дорожных пересечений, а также из процессов, генерирующих потоки машин. Процесс, генерирующий поток машин для определённого направления, оставляет сообщение в канале $car\_sense[i]$ ёмкостью 1. Контроллер светофора, обслуживающий тоже направление, читает сообщение из этого канала, и затем начинает последовательно захватывать ресурсы (пересечения). Это делается посредством отправки сообщений процессам, обслуживающим соответствующие пересечения по каналам блокировки. Процесс, управляющий пересечением, читает сообщение о блокировке, отправляет по рандеву каналу сообщение о подтверждении блокировки, и блокируется, ожидая сообщения об освобождении от блокирующего контроллера светофора. Когда контроллер светофора заблокировал все пересечения, через которые проходит регулируемый маршрут, он читает сообщение из канала о том, что машины ожидают проезда, меняет свой цвет на зелёный, и переключается обратно на красный. После этого отправляются сообщения о разблокировке пересечений в обратном порядке. Каналы, передающие пересечениям сообщения о блокировке, выглядят следующим образом:

Листинг 2: Каналы сообщений, обеспечивающие взаимодействие процессов контроллеров светофоров и процессов дорожных пересечений

```
23 chan lock[N_OF_INTERSECTIONS] = [N_OF_TRAFFIC_LIGHTS] of { mtype, byte }
```

```
24 chan accept[N_OF_TRAFFIC_LIGHTS] = [0] of { mtype }
25 chan release[N_OF_INTERSECTIONS] = [0] of { mtype }
```

Таким образом обеспечивается последовательная обработка светофорами приходящие потоки машин по всем направлениям, а рандеву каналы позволяют осуществлять синхронное взаимодействие между процессами контроллеров светофоров и дорожных пересечений.

5 Верификация алгоритма средствами spin

При верификации, за один "такт"верификатор spin делает только 1 шаг в одном из активных процессов, тем самым переводя систему в следующее состояние. Чтобы spin мог корректно верифицировать построенную модель, будем проводить процедуру верификации при условии "слабой справедливости" [?]. Эта опция spin позволяет гарантировать, что каждый доступный к исполнению процесс рано или поздно будет выбран верификатором spin для исполнения. Также при верификации были увеличены такие параметры, как доступный при верификации объём оперативной памяти, а также число допустимых состояний, которое может быть достигнуто при верификации.

6 Анализ результатов работы

В результате работы были изучены особенности проектирования систем параллельных вычислений, [?] построена корректно работающая модель контроллера светофоров, а также получены практические навыки работы с пакетом spin [?], [?], [?].