



UNIVERSIDADE DO MINHO

DEPARTAMENTO DE INFORMÁTICA

MESTRADO EM ENGENHARIA INFORMÁTICA

MÉTODOS FORMAIS DE PROGRAMAÇÃO

Programação Cíber-Física
Trabalho Prático I

Ariana Lousada (PG47034) Ana Ribeiro (PG47841)

16 de maio de 2022

1 Parte I

1.1 Design

Após uma análise do problema, a equipa de trabalho notou inicialmente que se teriam como entidades 3 semáforos distintos e um sensor. Contudo, após uma observação mais detalhada, observou-se que ambos os semáforos localizados na estrada principal terão sempre o mesmo comportamento, apresentando sempre estados iguais. Com isto, decidiu-se criar três autómatos diferentes: **S1**, que representa os semáforos da estrada principal; **S2**, que representa os semáforos da estrada secundária e o **Sensor** que representa o sensor da estrada secundária.

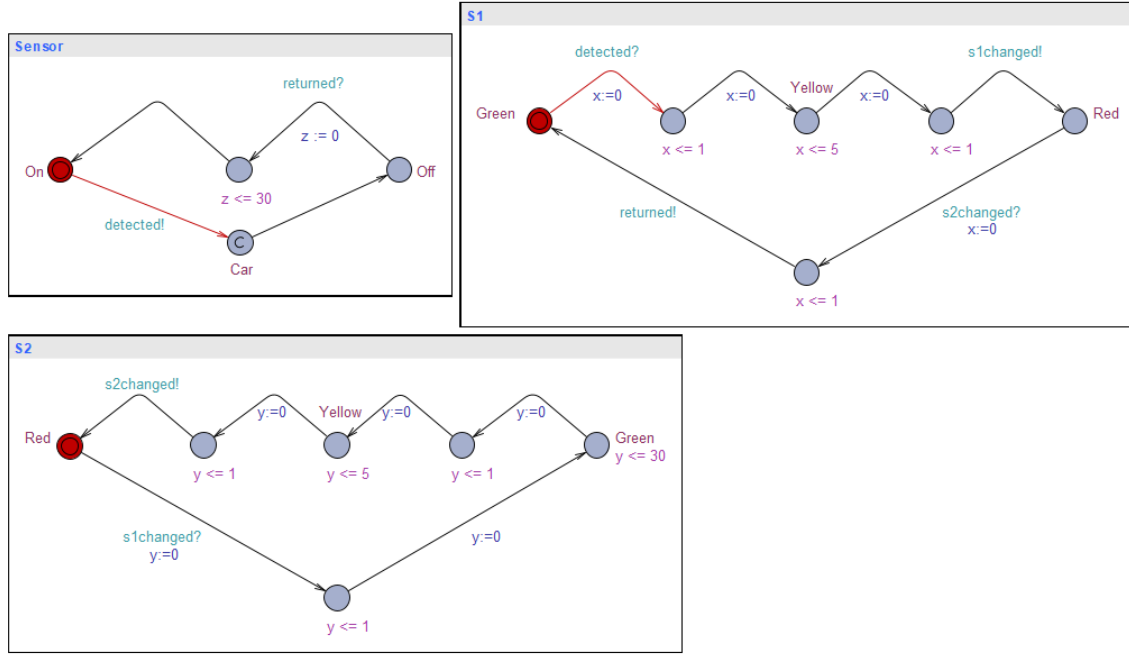


Figura 1: Sistema com as três entidades, Parte 1

O sensor contém dois estados principais: **On**, quando se encontra à espera da deteção de veículos e **Off** quando o veículo já ultrapassou o semáforo. O estado intermédio do **Sensor** (o estado **Car**) representa o momento de deteção entre a mudança entre os estados principais e foi decidido pela equipa de trabalho inseri-lo como um estado *committed*, de modo a que seja priorizado e que seja executado em primeiro lugar.

Quando um veículo é detetado, este desencadeia o processo de mudança de sinal vermelho para sinal verde do **S2**, mudando simultaneamente o **S1** para vermelho.

Todas as restrições de *delays* na mudança das luzes são controladas por *clocks* (um para cada autómato).

1.2 Fórmulas para teste do sistema

De modo a verificar a validade do sistema construído foram testadas as seguintes fórmulas CTL:

1. O semáforo da estrada secundária pode ficar verde:

$$E\Diamond (S2.Green) \quad (1)$$

2. O semáforo da estrada principal pode ficar vermelho:

$$E\Diamond (S1.Red) \quad (2)$$

3. O sistema nunca entra em estado de *deadlock*:

$$A\Box (\neg \text{deadlock}) \quad (3)$$

4. Os semáforos da estrada principal e da estrada secundária não podem estar a Verde em simultâneo:

$$A\Box (S1.Green \Rightarrow \neg S2.Green) \quad (4)$$

5. Se existem carros à espera (na estrada secundária) então eventualmente o semáforo desta vai mudar para Verde:

$$Sensor.Car \rightsquigarrow S2.Green \quad (5)$$

6. Em qualquer um dos semáforos, se estiverem a Amarelo então eventualmente irão mudar para Vermelho:

$$S1.Yellow \rightsquigarrow S1.Red \quad ; \quad S2.Yellow \rightsquigarrow S2.Red \quad (6)$$

7. Os semáforos nunca estão ambos Vermelhos em simultâneo.:

$$A\Box (S1.Red \Rightarrow \neg S2.Red) \quad (7)$$

De notar que em contexto real isto acontece durante cerca de 1 unidade de tempo. Contudo no sistema essa cor vermelha de um dos semáforos é representada por um estado intermédio de mudança de cor. O objetivo desta expressão é testar se os semáforos nunca ficam ambos na cor Vermelha sem ser em momentos de transição de estados.

Para verificar a validade do sistema construído, testou-se então a validade das fórmulas CTL anteriores no *Verifier* do UPPAAL:



Figura 2: Teste da validade das fórmulas CTL na ferramenta UPPAAL

2 Parte II

2.1 Design

Para a adaptação do modelo construído na primeira parte do projeto apenas se inseriu um novo sensor para o semáforo da estrada principal, visto que o controlo de transições de ambos os semáforos e sensores desta irá ser igual. Com isto então tem-se como entidades o semáforo da rua principal - S1, o semáforo da rua secundária - S2, o sensor da estrada principal - Sensor1 e o sensor da rua secundária - Sensor2.

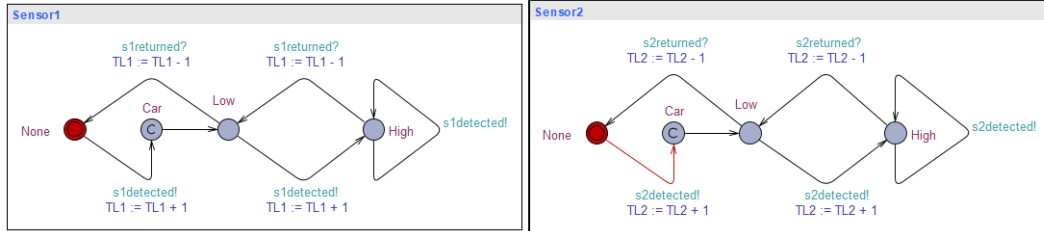


Figura 3: Sensores do sistema - Parte 2

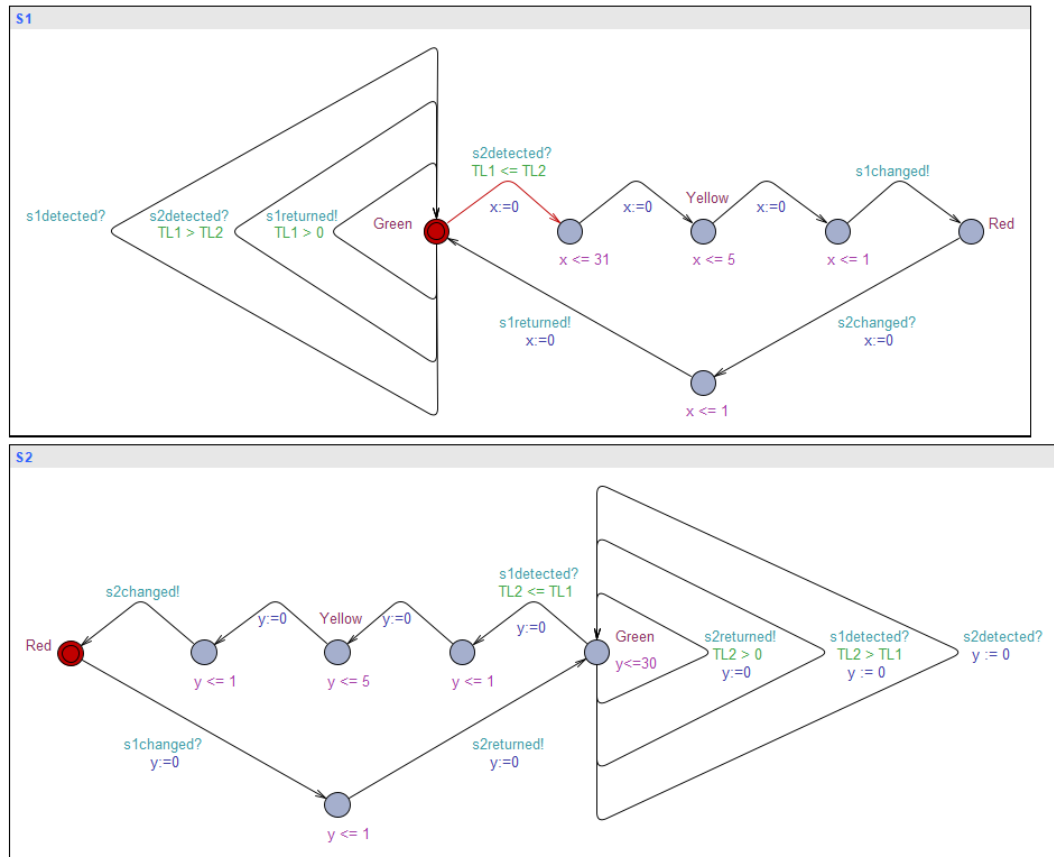


Figura 4: Semáforos do sistema - Parte 2

A maior modificação no modelo aplicou-se nos sensores do sistema, de modo a possuírem três estados distintos que representam vários volumes de trânsito: **None**, **Low** ou **High**. De modo ser possível testar o nível de trânsito em ambas as estradas foram declaradas duas variáveis globais adicionais: **TL1** - *traffic level S1* e **TL2** - *traffic level S2*. Estas variáveis são apenas controladas pelos sensores e possuem um dos seguintes valores:

- 0 - simboliza o nível de trânsito **None**;

- 1 - simboliza o nível de trânsito **Low**;
- 2 - simboliza o nível de trânsito **High**.

O objetivo da inserção destas variáveis é ajudar a manter os semáforos no estado necessário durante mais do que os 30 segundos definidos na parte anterior do projeto: é necessário que o semáforo da rua que possui mais trânsito de momento se mantenha a Verde enquanto o nível de trânsito da rua deste não descer em comparação com o trânsito da outra rua.

Com a inserção de novos estados para os sensores, foi necessário estabelecer uma ordem de mudança semelhante à aplicada nos semáforos na transição de cor Verde para Vermelho.

O nível de trânsito de cada estrada só aumenta caso o semáforo desta esteja a Vermelho e só diminui caso esteja a Verde.

Em termos de inicialização escolheu-se manter o estado inicial do semáforo da rua principal a Verde, uma vez que intuitivamente esta irá ter sempre um volume de trânsito mais elevado.

2.2 Fórmulas para teste do sistema

Para além das fórmulas definidas na primeira parte do projeto, foram inseridas novas formas CTL para teste do sistema:

8. Se o sensor da estrada atual detetar trânsito de nível **High**, então o semáforo da rua em questão terá de estar a Verde.

$$A \Diamond (Sensor2.High \text{ and } Sensor1.None \Rightarrow S2.Green) \quad (8)$$

$$A \Diamond (Sensor1.High \text{ and } Sensor2.None \Rightarrow S1.Green) \quad (9)$$

9. Os semáforos só devem mudar de cor quando houver uma alteração de um dos níveis de trânsito.
10. Os semáforos não deverão ficar no mesmo estado por tempo indefinido quando se observa um nível alto(High) de trânsito em ambas as ruas.

2.3 Problemas de implementação

Após a elaboração de testes das condições inseridas na ferramenta UPPAAL verificou-se que uma delas falha:

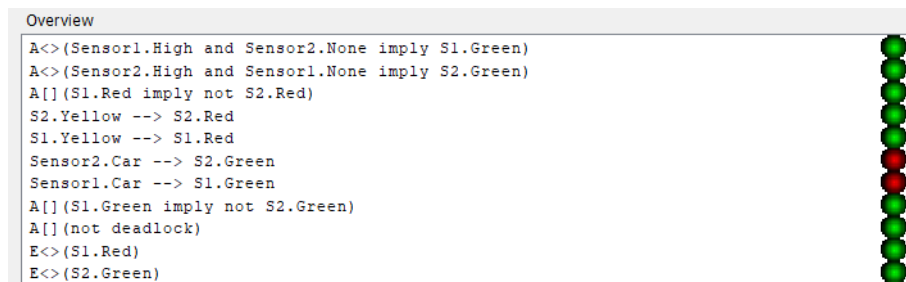


Figura 5: Teste da validade das fórmulas CTL na ferramenta UPPAAL

Através de várias observações ao sistema, reparou-se que sempre que os sensores detetem pelo menos um carro, estes mudam os respetivos sinais eventualmente, apesar do contra-exemplo apresentado pela ferramenta UPPAAL:

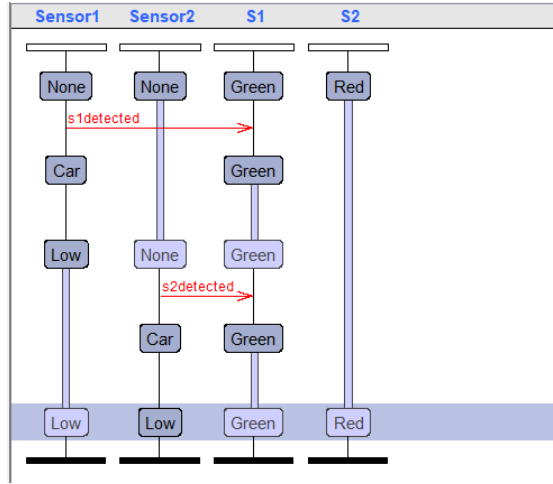


Figura 6: Contra-exemplo apresentado pela ferramenta UPPAAL

Neste contra-exemplo é apresentado um cenário no qual o **Sensor2** deteta um veículo e o correspondente semáforo não muda para Verde.

Contudo, se o estado for avançando, independentemente da transição escolhida, acaba-se por chegar a um cenário no qual o S2 muda para verde:

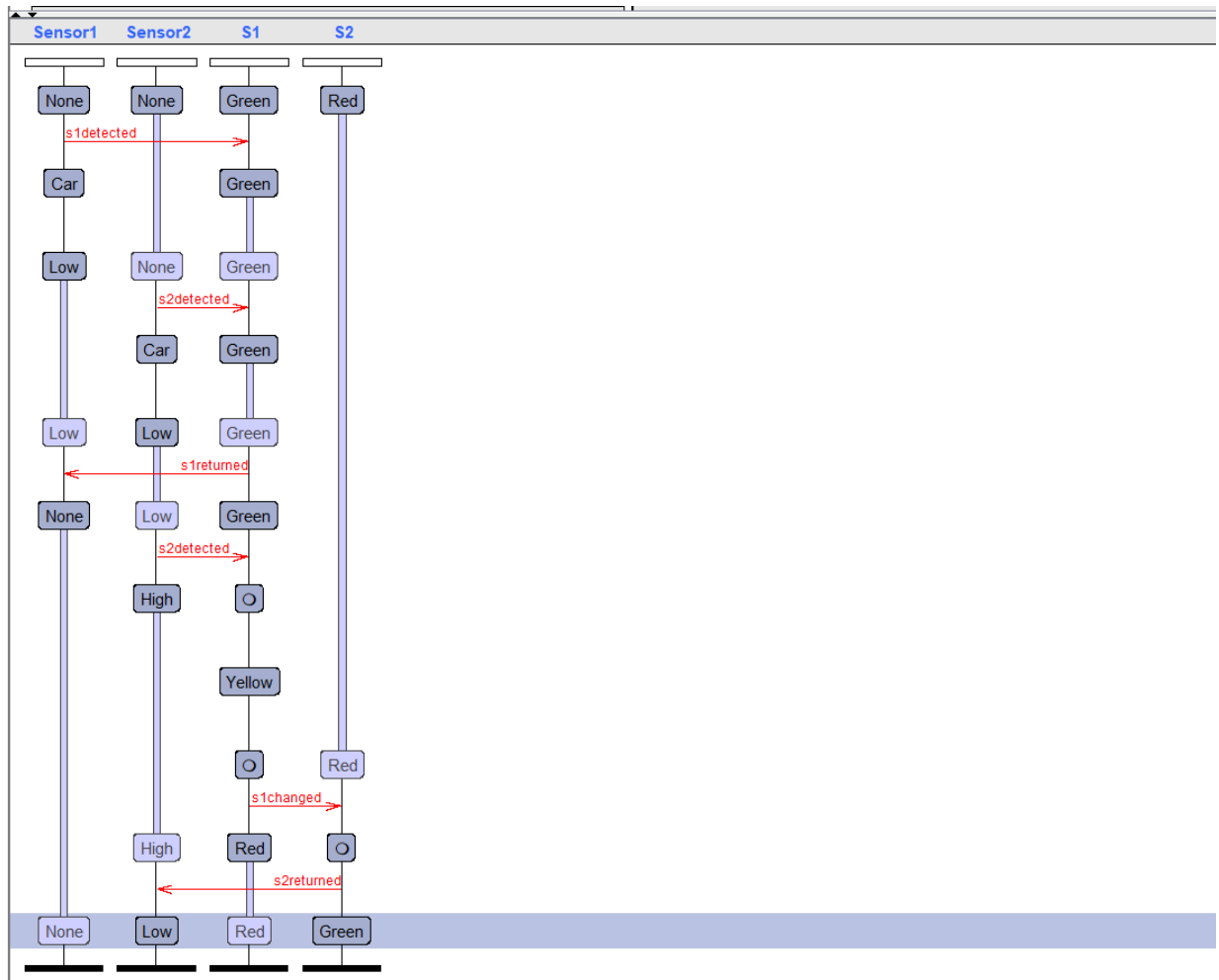


Figura 7: Continuação do contra-exemplo apresentado anteriormente

Com isto em conta, a equipa de trabalho viu-se incapaz de corrigir estas duas expressões de modo a

serem verificadas pela ferramenta.

A equipa de trabalho também se deparou com dificuldades nomeadamente na transformação das condições 2 e 3 definida anteriormente para linguagem CTL.

3 Conclusões

Após a construção de dois modelos, pode-se concluir que o segundo será o mais apropriado para uso em contexto real, uma vez que tem em conta as imprevisibilidades dos níveis do trânsito e gere os seus componentes de acordo com essa informação.

O bom desenvolvimento do segundo modelo pode em contexto real ter impactos positivos no ambiente e na vida diária sobretudo nas grandes cidades, como a redução de emissão de poluentes para a atmosfera e uma melhoria no trânsito.

Para o desenvolvimento e planeamento de sistemas críticos é necessária a análise de vários cenários que possam ocorrer quando o sistema é aplicado num ambiente real. Estes cenários devem ser verificados exhaustivamente recorrendo a ferramentas como o UPPAAL e a definições de condições em CTL. Isto auxilia um desenvolvimento correto de sistemas para redução de custos e eliminar o risco de perdas em termos de vidas humanas.