Contents

1 Objectives of this lecture  2

2 A simple 2-way intersection  2

3 The Context Machine  2

4 The SimpleTwoWay Machine  2

5 The Invariant  3
  5.1 Animation  3
  5.2 Strengthening the invariant  3
  5.3 Other Invariants  4
  5.4 Further Animation  4
  5.5 Strengthening the Precondition  5
  5.6 The SimpleTwoWay machine  5

6 Sequencing  6
  6.1 The precondition of ToRed  6
  6.2 The precondition of ToGreen  6
  6.3 Equivalent predicates  7
  6.4 The precondition of ToAmber  7

7 The TwoWay Machine  8
1 Objectives of this lecture

- To explore the specification of some simple traffic light controllers.
- To explore the use of a state invariant to ensure safety.
- To explore the use of preconditions that ensure an operation will not violate the state invariant, when the precondition is satisfied.
- To use the animator to illustrate these explorations.

2 A simple 2-way intersection

Consider traffic lights at the intersection of two roads, one running North-South and the other East-West. There are four sets of lights, each capable of showing Red, Green and Amber, placed at North, East, South and West positions.

The North and South lights are always identical, as are the East and West lights.

There are no right-turn lights.

Lights should change in the sequence:

Red → Green → Amber → Red → ...

We wish to specify a traffic light controller that ensures safety and the correct sequencing.

3 The Context Machine

We will introduce a context machine containing the enumerated sets DIRECTION and LIGHT.

We will also specify a constant (function) OTHER_DIR that maps each direction into the other direction.

Context machines are used commonly and frequently in the B Method (B) to define sets and constants.

MACHINE TwoWay_Ctx
SETS
   DIRECTION = { NorthSouth , EastWest } ;
   LIGHT = { Red , Green , Amber }
CONSTANTS OTHER_DIR
PROPERTIES
   OTHER_DIR ∈ DIRECTION → DIRECTION ∧
   OTHER_DIR = { NorthSouth ↦ EastWest , EastWest ↦ NorthSouth }
END

4 The SimpleTwoWay Machine

1. We will write a basic machine with no non-trivial state invariant and no non-trivial preconditions.
2. The machine will see the context machine and have a state of one variable, \( \text{lights} \), which is a total function from \( \text{DIRECTION} \) to \( \text{LIGHT} \).

3. There is one operation: \( \text{ChangeLight}(\text{dir}, \text{colour}) \) that changes the light to \( \text{colour} \) in the direction \( \text{dir} \).

\[
\begin{align*}
\text{MACHINE} & \text{ SimpleTwoWay0} \\
\text{SEES} & \text{ TwoWay_Ctx} \\
\text{VARIABLES} & \text{ lights} \\
\text{INVARIANT} & \text{ lights } \in \text{DIRECTION } \rightarrow \text{LIGHT} \\
\text{INITIALISATION} & \text{ lights } := \{ \text{NorthSouth } \mapsto \text{Red} , \text{EastWest } \mapsto \text{Red} \}
\end{align*}
\]

\[
\begin{align*}
\text{OPERATIONS} & \\
\text{ChangeLight} & (\text{dir}, \text{colour}) \equiv \\
\text{pre} & \text{ dir } \in \text{DIRECTION } \land \text{ colour } \in \text{LIGHT} \\
\text{then} & \text{ lights } (\text{dir}) := \text{colour} \\
\text{end} \\
\text{END}
\end{align*}
\]

5 The Invariant

5.1 Animation

1. Use the animator to explore the behaviour. When animating, choose to display the invariant, normally turned off.

2. It is, of course, trivial to establish that the controller is unsafe.

3. Try to formulate an invariant that will ensure safety.

4. Whenever the state is unsafe, the invariant must be \( \text{false} \).

\[
\neg(\text{safe}) \Rightarrow \neg(\text{invariant}) \tag{1}
\]

5. Conversely, whenever the invariant is \( \text{true} \), the state should be safe.

\[
\text{invariant} \Rightarrow \text{safe} \tag{2}
\]

6. Of course, \( \text{1 and 2 are equivalent; one is the contrapositive of the other: } P \Rightarrow Q \equiv \neg Q \Rightarrow \neg P \)

7. If we have the \( \text{weakest} \) invariant, then whenever the state is safe, the invariant will be \( \text{true} \).

8. Find adequately strong preconditions.

5.2 Strengthening the invariant

Clearly, the light in both directions cannot be either \text{Green} or \text{Amber}. 
This leads to the invariant:

\[ \neg (\text{lights}(\text{NorthSouth}) \in \{\text{Green, Amber}\} \land \\
\text{lights}(\text{EastWest}) \in \{\text{Green, Amber}\}) \]
\[ \equiv \\
(\text{lights}(\text{NorthSouth}) \in \{\text{Green, Amber}\} \Rightarrow \\
\neg (\text{lights}(\text{EastWest}) \in \{\text{Green, Amber}\})) \]
\[ \equiv \\
(\text{lights}(\text{NorthSouth}) \in \{\text{Green, Amber}\} \Rightarrow \\
\text{lights}(\text{EastWest}) = \text{Red}) \]

5.3 Other Invariants

There are other invariants that adequately express safety for a two-way intersection:

\[ \text{lights}(\text{NorthSouth}) = \text{Red} \lor \text{lights}(\text{EastWest}) = \text{Red} \]

\[ \text{Red} \in \text{ran}(\text{lights}) \]

But these conditions do not generalise to intersections with more than two ways. Indeed the expression of the invariant that best generalises is

\[ \forall \text{dir}. (\text{dir} \in \text{DIRECTION} \land \text{lights}(\text{dir}) \in \{\text{Green, Amber}\} \Rightarrow \\
\text{lights}(\text{OTHER_DIR}(\text{dir})) = \text{Red}) \]

5.4 Further Animation

Further Animation

Try animating with the invariant on the preceding slide, using the animation script on the following slide.
5.5 Strengthening the Precondition

We now have states and arguments of the operation ChangeLight that lead to a state that violates the state invariant.

This is not satisfactory!

The preconditions need to be strengthened so that the post-state violates the invariant only if the precondition is false.

This illustrates how preconditions and invariants collaborate in achieving safety.

Notice that preconditions ensure safety by imposing a proof obligation to be discharged in any context in which the operation is used.

A possible precondition is

\[
\text{colour} \in \{\text{Green, Amber}\} \Rightarrow \text{lights(OOTHER\_DIR(dir)) = Red}
\]

5.6 The SimpleTwoWay machine

In this model of SimpleTwoWay we put the following alternative formulations of the invariant into the assertions:

\[\forall \text{dir}. (\text{dir} \in \text{DIRECTION} \wedge \text{lights(dir)} \in \{\text{Green, Amber}\}) \Rightarrow \text{lights(OOTHER\_DIR(dir)) = Red}\]

\[\text{lights(EastWest)} \in \{\text{Green, Amber}\} \Rightarrow \text{lights(NorthSouth) = Red}\]

Assertions contain properties that are implied by the invariant, and hence discharging the proof obligations for the assertions proves that each conjunct in the assertions is implied by the invariant.
6 Sequencing

The SimpleTwoWay machine ensures safety, but does not enforce any sequencing.

To achieve the desired sequencing we will introduce a new machine: TwoWay, that INCLUDES SimpleTwoWay and has three operations ToRed(dir), ToGreen(dir), and ToAmber(dir) for changing the lights.

The notion is that the body of each operation will use the operation ChangeLight(dir, colour) to change the colour.

The precondition of each operation will constrain the sequencing, and also must ensure that the precondition of ChangeLight is satisfied.

6.1 The precondition of ToRed

Since the precondition of ChangeLight is always satisfied for the colour Red, we need to only be concerned with sequencing.

Hence the precondition is

\[ \text{lights}(\text{dir}) = \text{Amber} \]

6.2 The precondition of ToGreen

When setting a light to Green the precondition of ChangeLight requires

\[ \text{lights}(\text{OTHER_DIR}(\text{dir})) = \text{Red} \]
Sequencing requires

\[ \text{lights}(\text{dir}) = \text{Red} \]

Thus the precondition is the conjunction

\[ \text{lights}(\text{dir}) = \text{Red} \land \text{lights}(\text{OTHER_DIR}(\text{dir})) = \text{Red} \]

### 6.3 Equivalent predicates

This may be expressed by a number of equivalent predicates, when combined with the sequencing predicate \( \text{lights}(\text{dir}) = \text{Red} \):

1. \( \text{dir} = \text{NorthSouth} \Rightarrow \text{lights}(\text{EastWest}) = \text{Red} \land \text{dir} = \text{EastWest} \Rightarrow \text{lights}(\text{NorthSouth}) = \text{Red} \)

2. \( \forall \text{dir}. (\text{dir} \in \text{DIRECTION} \Rightarrow \text{lights}(\text{dir}) = \text{Red}) \)

3. \( \text{lights}[\text{DIRECTION}] = \{\text{Red}\} \)

4. \( \text{ran}(\text{lights}) = \{\text{Red}\} \)

It is worth noting that alternatives 2, 3 and 4 if generalized to more directions, are too strong.

### 6.4 The precondition of ToAmber

In order to satisfy the precondition of \textit{ChangeLight}, the light in the other direction must be showing \textit{Red}.

But, since the sequencing condition is \( \text{lights}(\text{dir}) = \text{Green} \),

the state invariant implies that

\[ \text{lights}(\text{OTHER_DIR}(\text{dir})) = \text{Red} \]

So, only the sequencing condition is required

\[ \text{lights}(\text{dir}) = \text{Green} \]

\[ ^1 \text{This uses relational image } r[s] = \text{ran}(s < r) \]
7 The TwoWay Machine

MACHINE TwoWay
SEES TwoWay_Ctx
INCLUDES SimpleTwoWay

OPERATIONS

ToRed ( dir ) ≜
pre dir ∈ DIRECTION ∧ lights ( dir ) = Amber
then ChangeLight ( dir , Red )
end ;
ToGreen ( dir ) ≜
pre dir ∈ DIRECTION ∧ lights ( dir ) = Red ∧
lights ( OTHER_DIR ( dir ) ) = Red
then ChangeLight ( dir , Green )
end ;
ToAmber ( dir ) ≜
pre dir ∈ DIRECTION ∧ lights ( dir ) = Green
then ChangeLight ( dir , Amber )
end
END