1 Specification

1.1 The Specification

MODULE *bridge*

EXTENDS *Naturals*, *Sequences*
CONSTANT *Cars*, *Capacity*

VARIABLE *bridge*

$$\mathit{Init} \triangleq \mathit{bridge} = \langle\rangle$$

$$\mathit{Type} \triangleq \mathit{bridge} \in \mathit{Seq}(\mathit{Cars})$$

$$\mathit{Safe} \triangleq \mathit{Len}(\mathit{bridge}) \leq \mathit{Capacity}$$

Test to see if a value is in a sequence

$$\mathit{IsOnBridge}(a) \triangleq \exists\, n \in \textsc{domain}\ \mathit{bridge} : \mathit{bridge}[n] = a$$

$$\begin{array}{l} \mathit{enter}(c) \triangleq \\ \qquad \land\ c \in \mathit{Cars} \\ \qquad \land\ \neg \mathit{IsOnBridge}(c) \\ \qquad \land\ \mathit{Len}(\mathit{bridge}) < \mathit{Capacity} \\ \qquad \land\ \mathit{bridge}' = \mathit{Append}(\mathit{bridge},\ c) \end{array}$$

$$\begin{array}{l} \mathit{exit} \triangleq \\ \quad \land\ \mathit{Len}(\mathit{bridge}) > 0 \\ \quad \land\ \mathit{bridge}' = \mathit{Tail}(\mathit{bridge}) \end{array}$$

$$\begin{array}{l} \mathit{Next} \triangleq \\ \quad \lor\ \exists\, c \in \mathit{Cars} : \mathit{enter}(c) \\ \quad \lor\ \mathit{exit} \end{array}$$

Modification History

Last modified Fri Feb 12 20:31:31 GMT 2021 by alunm

Created Thu Feb 11 10:48:41 GMT 2021 by alunm

2 The Model

2.1 Model Overview

The Behaviour specification is an *Initial predicate and next-state relation*

Initial predicate *Init*

Next-state relation *Next*

The Model values assigned to declared constants

The set of cars in the model $Cars \leftarrow 1..10$

The capacity of the bridge $Capacity \leftarrow 3$

2.2 Checks and verifications

Invariants Two invariants are checked

The Type Invariant *Type*

The Safety Invariant *Safe*

2.3 Results

A summary of the results

Statistics a summary of the actions and number of states found.

States found	1641
Distinct states	821

Action	Location	States Found	**Distinct states**
Init	Line 7	1	1
enter	Line 16	820	820
exit	Line 21	820	0

2.4 Discussion

2.4.1 Model description

The state of the system is modelled by a sequence of cars on the bridge.

The initial conditions are that the bridge is empty

The type invariant is that the state-variable *bridge* is a sequence of cars from the model set.

The safety invariant is that the length of the sequence is not greater than the bridge capacity.

The Next relation is that there is some car that can enter the bridge, or that a car exits the bridge.

A car can enter the bridge if it is not already on the bridge, and the bridge has not reached its capacity.

2.4.2 Interpretation of results

The specification verifies with the model, the type and safety invariants are kept. There isn't a set to see if the order of cars entering and leaving the bridge matches, we can infer this is the case from the definition of sequences.