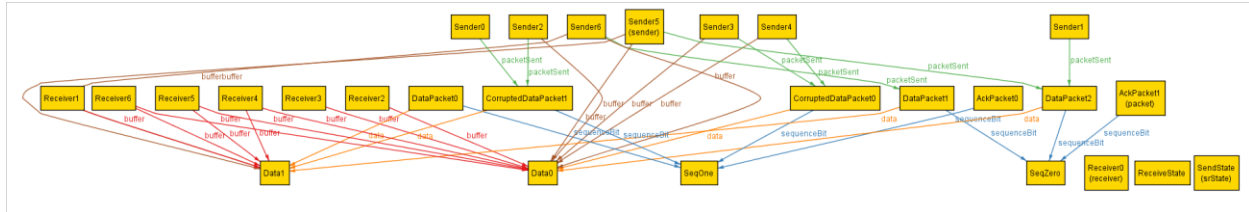


RDT2.2

We successfully modeled RDT2.2.

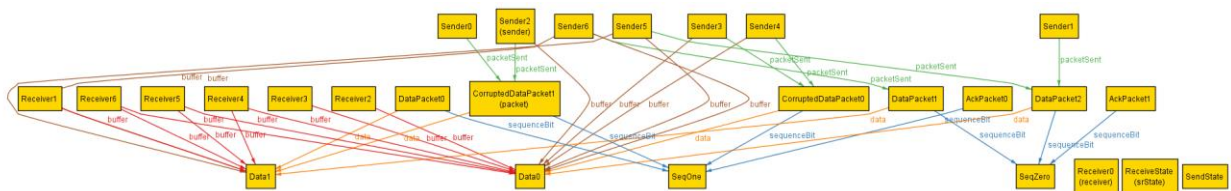
Property 1: There is at least one way for all data to be transferred.

State 0:



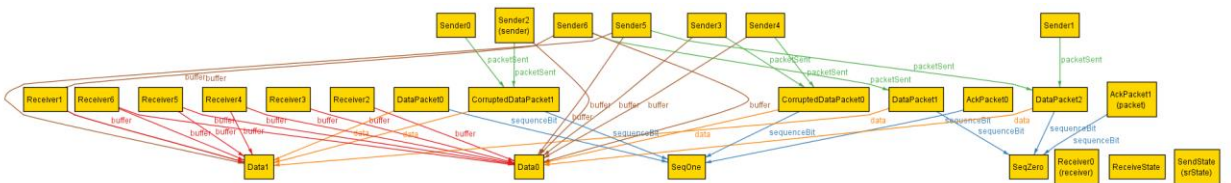
We are currently in the send state. The current packet is AckPacket which has SeqZero.

State 1:



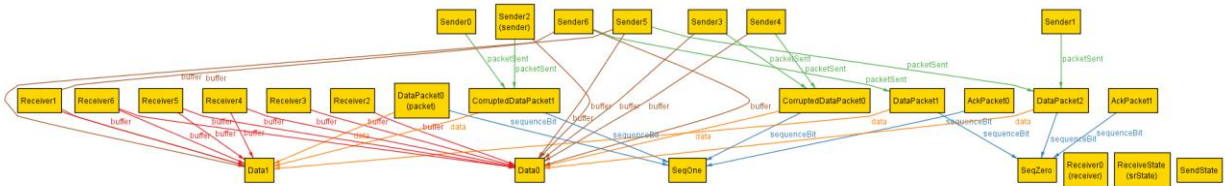
We are currently in the receive state. The current packet is CorruptedDataPacket which has SeqOne.

State 2:



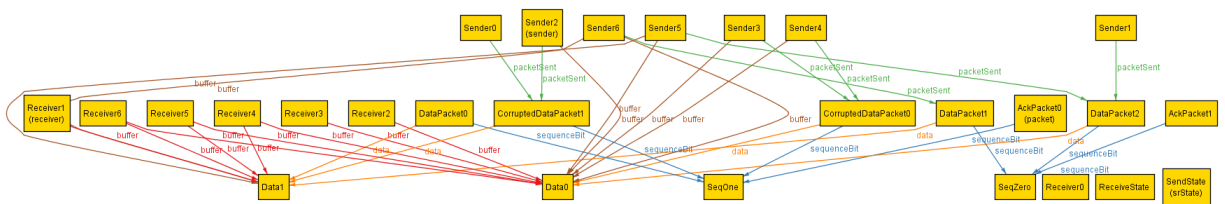
We are currently in the send state. The current packet is AckPacket1 which has SeqZero. This means that the data was not sent correctly. Therefore, the data is not held by the receiver.

State 3:



We are currently in the receive state. The current packet is DataPacket0 which has SeqOne.

State 4:

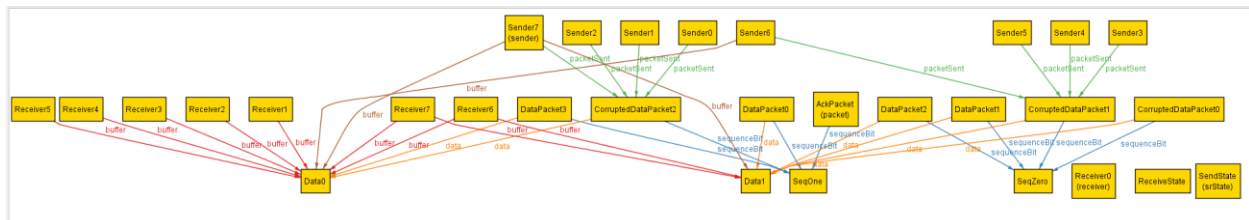


We are currently in the send state. The current packet is AckPacket0 which has SeqOne. Because the Ack is sending the correct sequence bit, the data was sent and received correctly; all data is now held in the receiver's buffer.

Property 2: It is not always possible for all data to be transferred

Alloy finds a counterexample, similar to the counterexample found for RDT2.1.

State 0:



We start in the send state. The current packet is AckPacket which has SeqOne.

State 1:

Executing "Check allDataCanBeTransferredWithErrorLimit for 10 but 2 Data"

Solver=sat4j Bitwidth=0 MaxSeq=0 SkolemDepth=1 Symmetry=20

8888 vars. 552 primary vars. 21317 clauses. 37ms.

No counterexample found. Assertion may be valid. 830ms.

When the constraint is added that there can be no more than one error for each data packet, Alloy finds no counterexample.