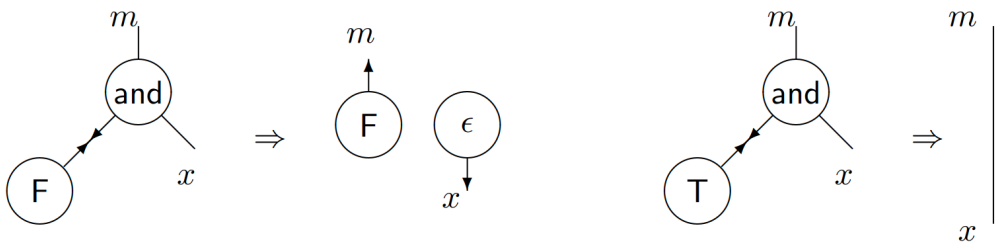
6CCS3COM Computational Models

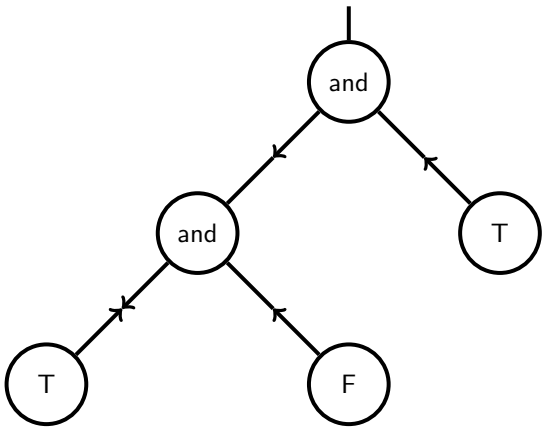
Tutorial 5

Josh Murphy

1. Give an interaction system to compute the Boolean function *and*.



2. Using your interaction system, draw the net representing the expression *(True and False) and True*. How many reductions are required to reduce the net to normal form?



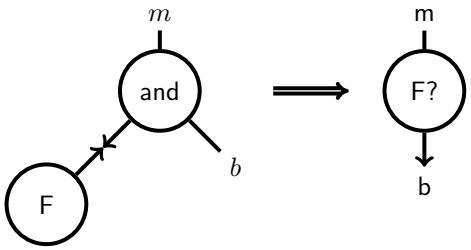
The net has an active pair (True, False) which reduces to False, and a new active pair (False, True) is created, which reduces to False, and True is erased.

3. Modify the interaction system to that the result is True if and only if both arguments have the same value (*i.e.* both True or both False).

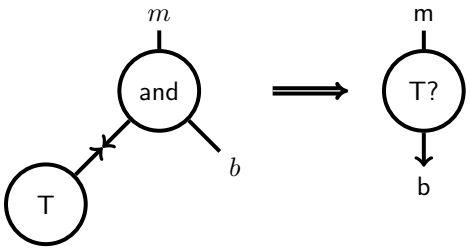
We need to change the first rule, erasing the agent F in the right-hand side and replacing the agent ϵ with an agent F? to check whether the second argument is F. For this, we need to add two rules for F?: it interacts with T giving F and with F giving T.

We also need to change the second rule, so that after testing the first argument, it checks the second. For this, in the right hand side of the second rule we introduce an agent T?, to check that the second argument is True. We need to add two rules: T? interacts with T giving T and with F giving F.

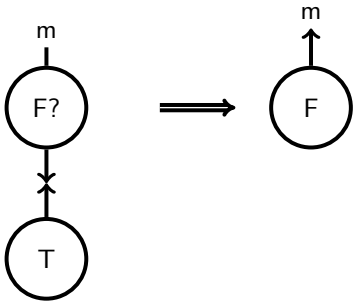
AND — RULE 1



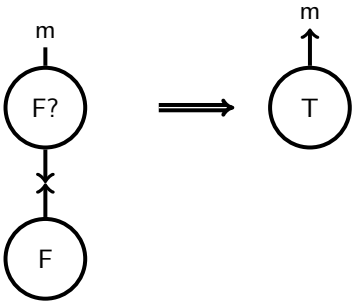
AND — RULE 2



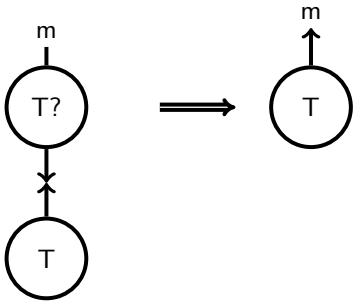
F? — RULE 1



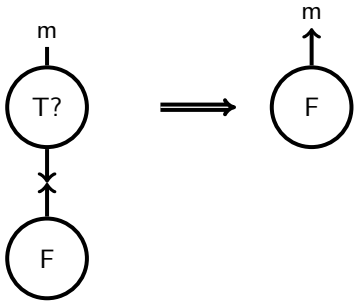
F? — RULE 2



T? — RULE 1



T? — RULE 2



4. Specify an interaction system that can generate infinite computations. Give an example net in the system that does not have a normal form (**i.e.** with an infinite sequence of interactions).

   An interaction system that has a rule where the right hand side of the rule is identical to the left hand side of the rule can generate nets with infinite computations. If an active pair "reduces" to itself then the net will always contain an active pair, and so no normal form will occur.

5. Define the function Parallel-and using the agent `amb`. Parallel-and is a binary Boolean operator returning the value False whenever one of the arguments is False and True when both are True.

   A parallel-and can be built, for instance, using an agent amb with both principal ports connected to the arguments, and both output ports connected to an and agent.