reachability analysis for continuous one counter automata

Lars Van Roy
dept. of Mathematics and Computer Science
University of Antwerp
lars.vanroy@student.uantwerpen.be

May 8, 2021

Current status

Current status

- Continuous one counter automata

Current status

- Continuous one counter automata
  - No parameters

Current status

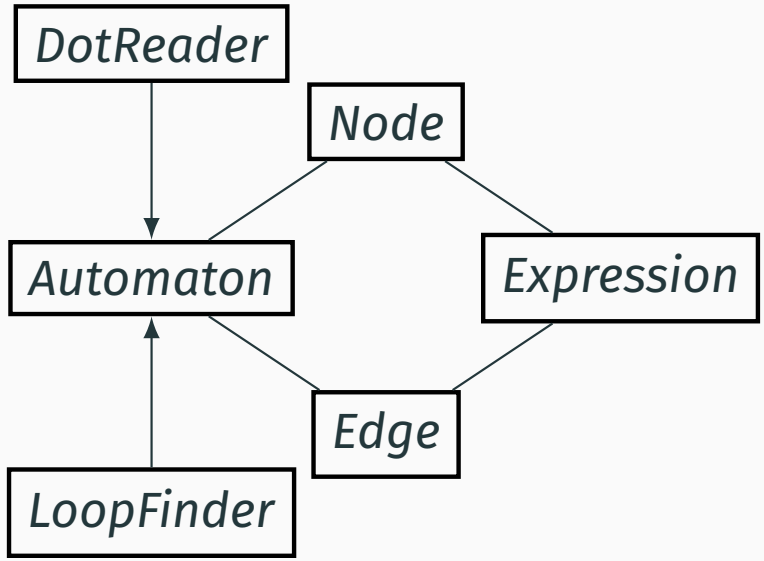
- Continuous one counter automata
  - No parameters
  - Operations: +, -

Current status

- Continuous one counter automata
  - No parameters
  - Operations: +, -
  - Conditions: <=, >=, =

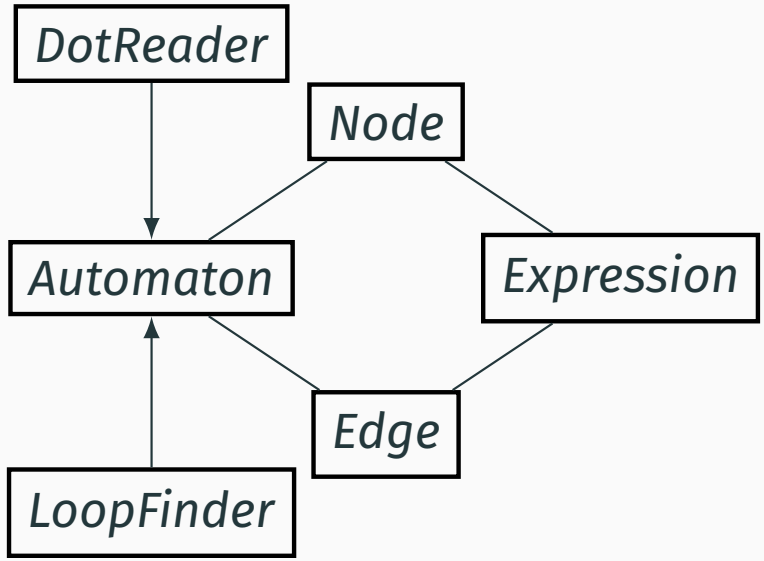
Automaton

- Convert dot to Automaton



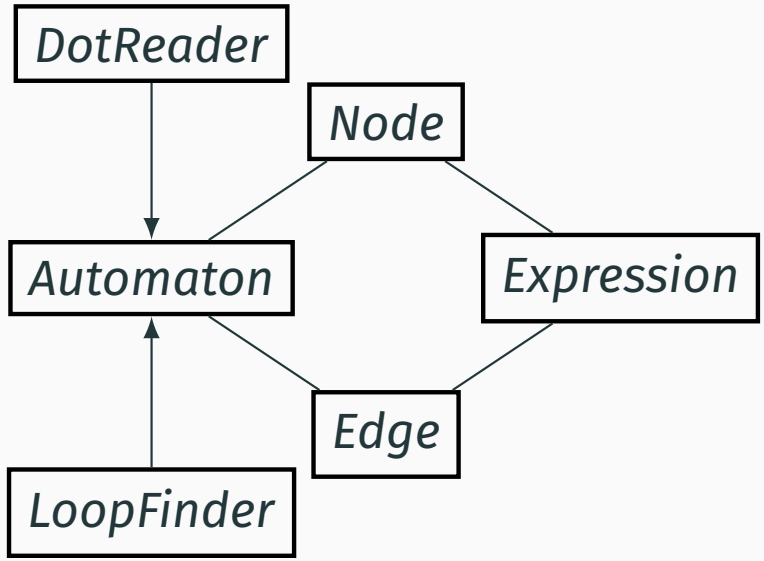
Automaton

- Convert dot to Automaton
- Ensure conformance
  - Initial node
  - Operations in edge
  - Conditions in node

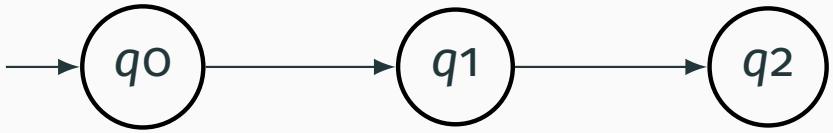


Automaton

- Convert dot to Automaton
- Ensure conformance
  - Initial node
  - Operations in edge
  - Conditions in node
- Detect all loops
  - Breadth first search

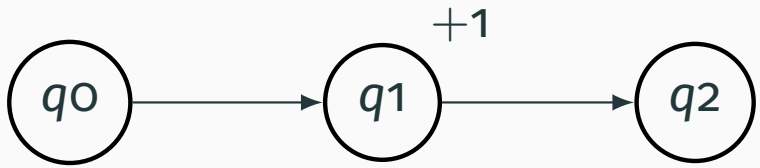


Initial node



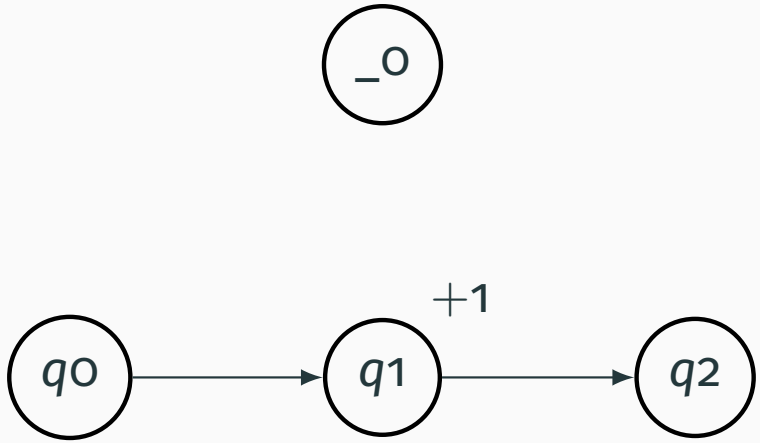
Operation in node

- Given a node with an operation label



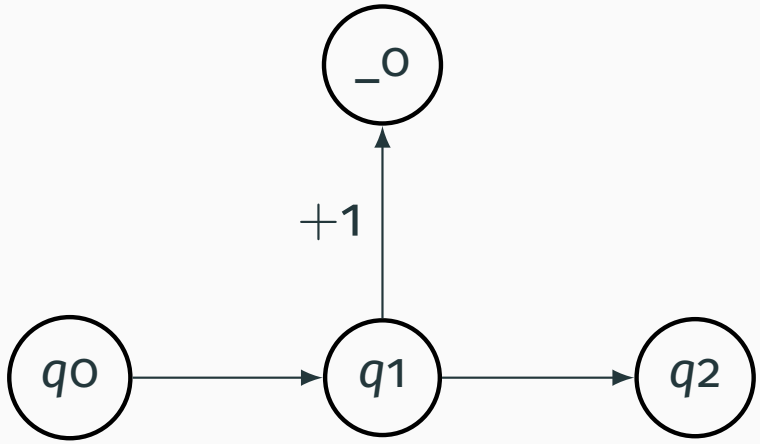
Operation in node

- Given a node with an operation label
- Insert a new node



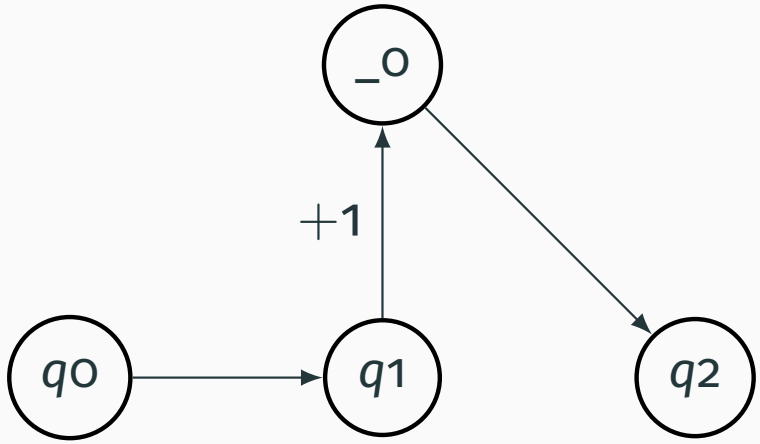
Operation in node

- Given a node with an operation label
- Insert a new node
- Create an edge to this node
- Add the operation to the edge



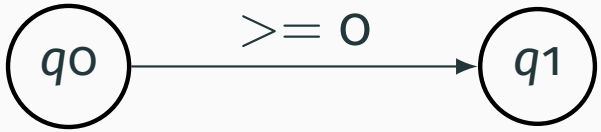
Operation in node

- Given a node with an operation label
- Insert a new node
- Create an edge to this node
- Add the operation to the edge
- Reconnect all pre-existing edges



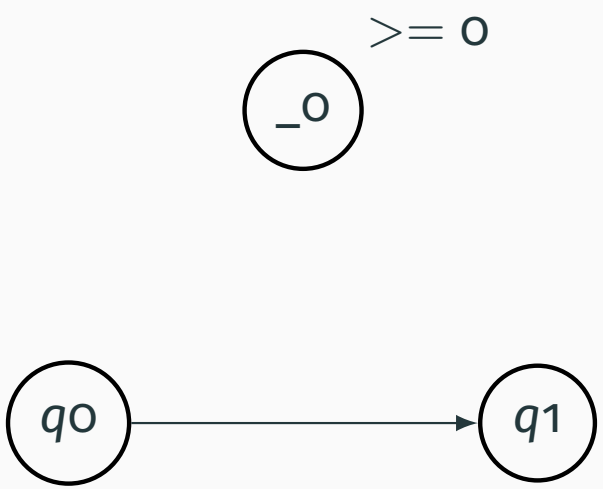
Condition in edge

- Given an edge with a conditional label



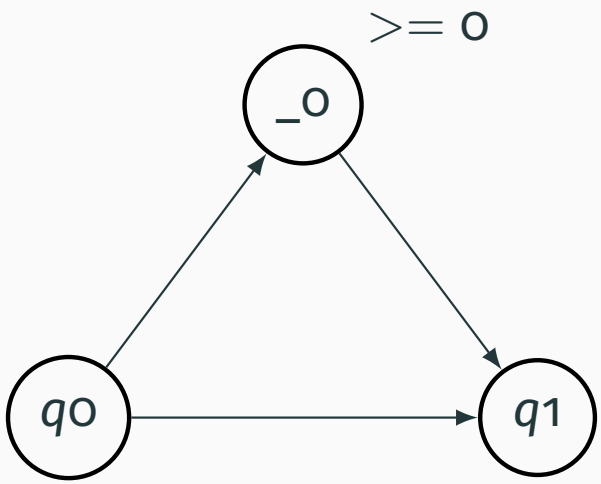
Condition in edge

- Given an edge with a conditional label
- Insert a new node
- Add the condition to the node



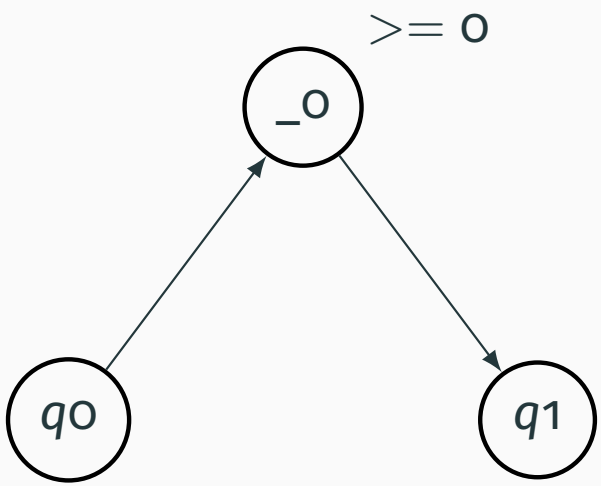
Condition in edge

- Given an edge with a conditional label
- Insert a new node
- Add the condition to the node
- Connect the two nodes via the new node



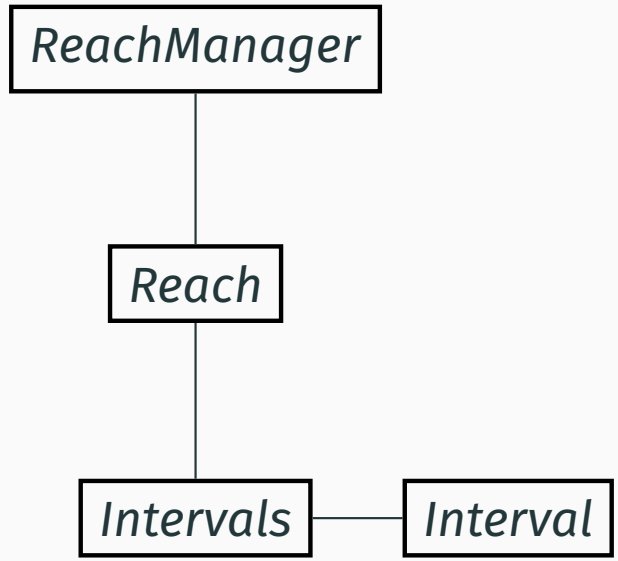
Condition in edge

- Given an edge with a conditional label
- Insert a new node
- Add the condition to the node
- Connect the two nodes via the new node
- Remove the old edge



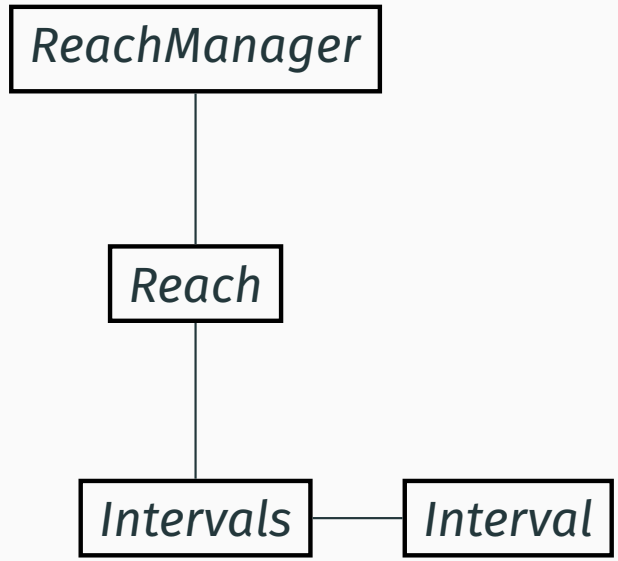
Reachability

- Initialise Reach per node



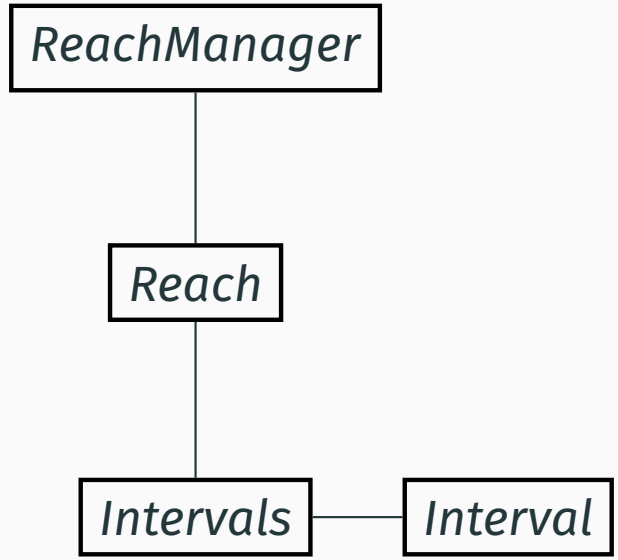
Reachability

- Initialise Reach per node
- Update Reach instances
  - Do one update step
  - Rescale Reach
  - Check loop acceleration
  - Check end condition



Reachability

- Initialise Reach per node
- Update Reach instances
  - Do one update step
  - Rescale Reach
  - Check loop acceleration
  - Check end condition
- Reachability
  - Node has a Reach that is not empty



Remaining work

- Add support for parametric counter automata

Remaining work

- Add support for parametric counter automata
- Apply to a (or more) bigger use case(s)
- Document the results