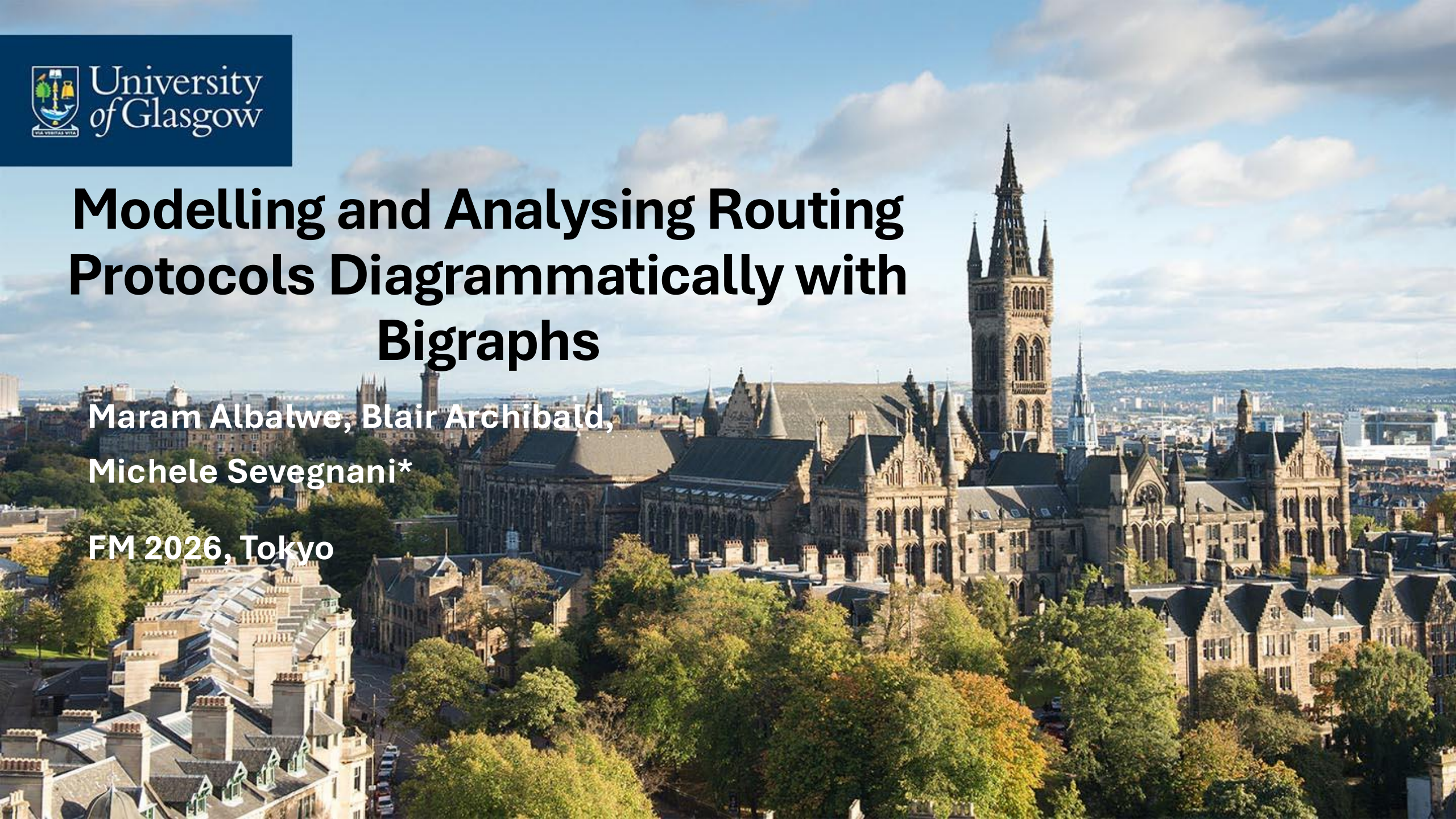


University of Glasgow

Modelling and Analysing Routing Protocols Diagrammatically with Bigraphs

Maram Albalwe, Blair Archibald,
Michele Sevegnani*

FM 2026, Tokyo

University of Glasgow

Protocol design practice

Simulation/emulation is standard practice

- Implementation fidelity is a strength
- Coverage is necessarily partial
- Exploring variants has engineering cost

Formal methods offer exhaustive reasoning

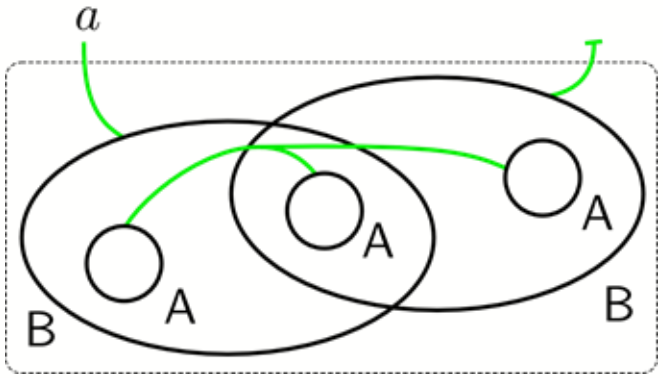
- But modelling notations can be hard to adopt
- Usability and scalability remain practical barriers

Question: can we keep rigour while improving modelling usability?

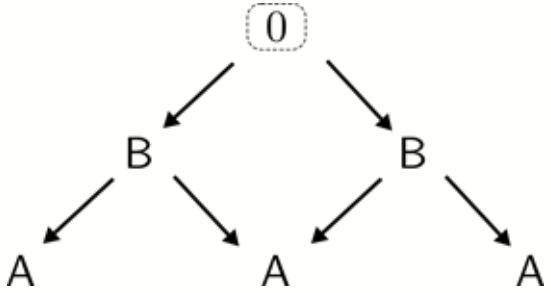
University of Glasgow

Bigraphs: structure

1. Bigraph: superimposition of a **place graph** and a **link graph**
2. Place graph (**with sharing**): **DAG** - topological space - no distances - containment relation
3. Link graph: **Hypergraph** - relationships between sets of entities (e.g. communication capabilities)



Bigraph



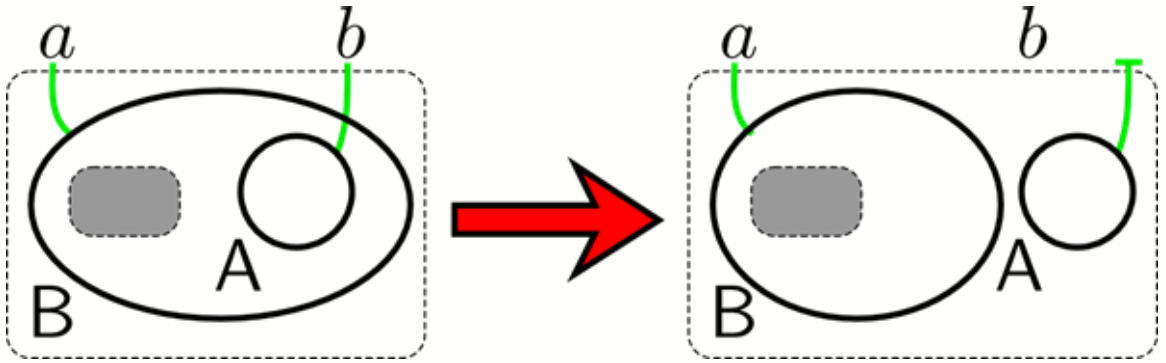
Place graph

Link graph

University of Glasgow

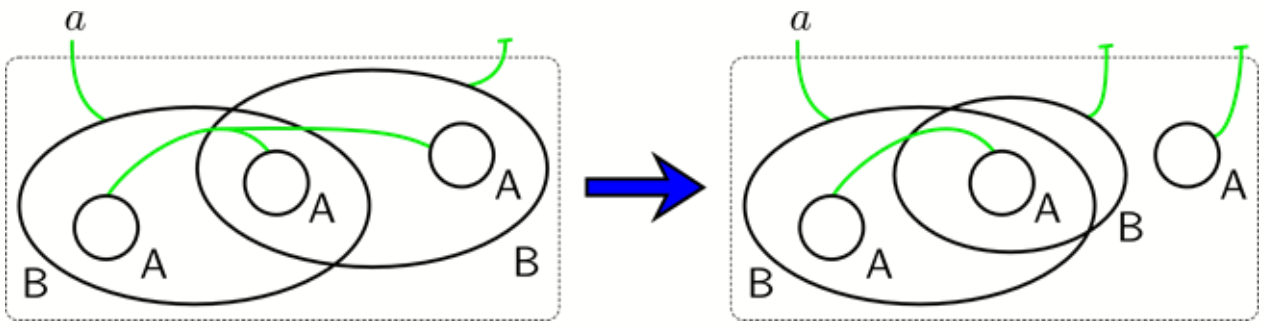
Bigraphs: dynamics

- Specified by a set of reaction rules
  1. Identify occurrences of the lhs in the bigraph
  2. Substitute each occurrence with the rhs (rewriting)
- Global behaviour emerges from all possible rule applications

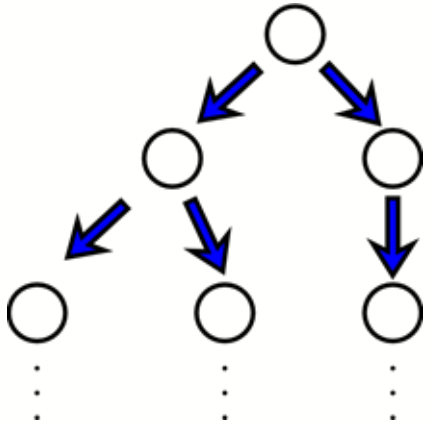


Bigraphical Reactive Systems

- An **initial** bigraph and a **set** of reaction rules
- By performing all the rewriting steps we find all the reachable configurations of the system
- The resulting transition system can be labelled and model checked



Transition



Transition system

University of Glasgow

Approach overview

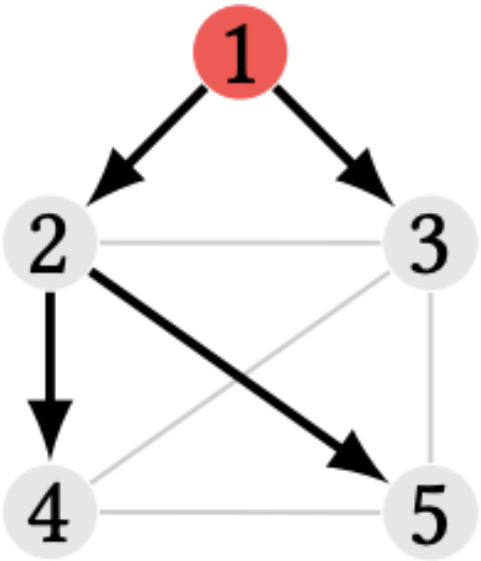
1. Encode network topologies as bigraphs
2. Define behaviour via reaction rules
3. Generate the labelled transition system (for a given topology)
4. Validate model by verifying properties with a model checker
5. Compare our model against a popular simulation-based approach

University of Glasgow

RPL: Routing Protocol for Low-power and Lossy Networks

- RPL builds a DAG that defines an optimal routing path from the root to each reachable node
- There are four control messages

1. DIS (DAG Information Solicitation): requests information about a nearby DAG
2. DIO (DAG Information Object): transmits information, e.g. rank, to other nodes
3. DAO (Destination Advertisement Object): used by nodes joining the DAG to update other nodes with new routing information
4. DAO-ACK (Destination Advertisement Object Acknowledgement): sent by the root to a joining node



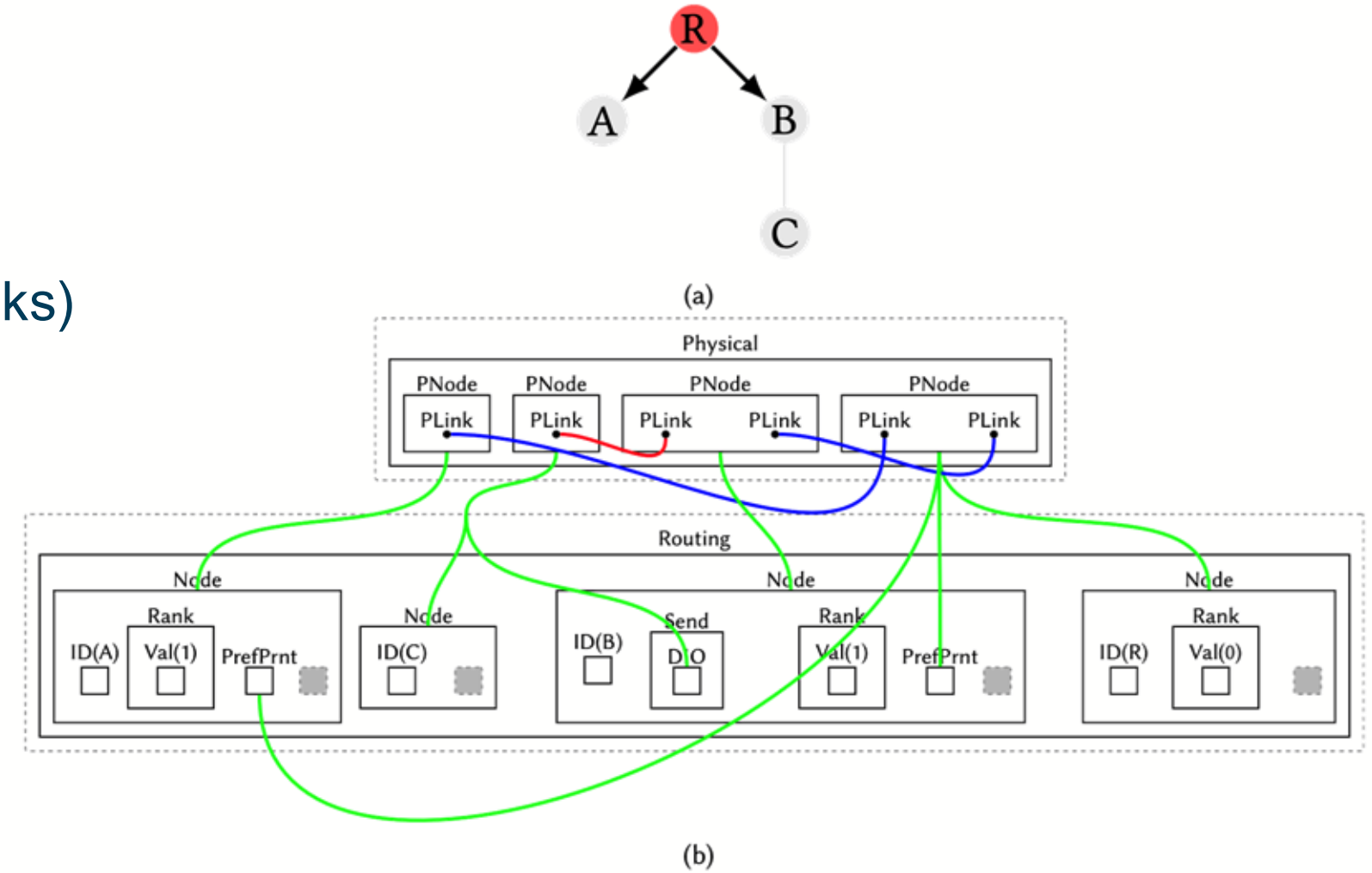
Modelling structure: two perspectives

Physical perspective

- PNode
- PLink (for radio links)

Routing perspective

- Node
- Rank
- Messages
- ….



University of Glasgow

Modelling RPL dynamics

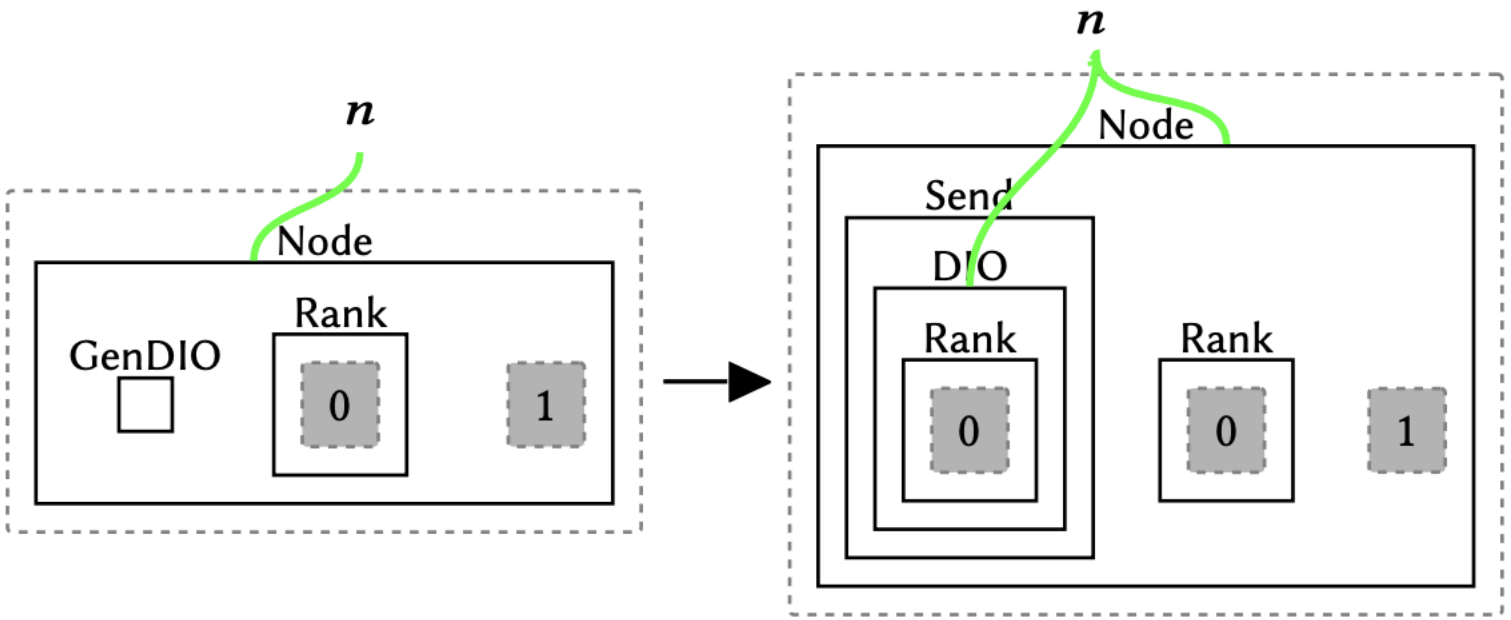
- 16 reaction rules encoding the four main steps of the RPL DAG constructions
- Rules can only rely on local context to encode the behaviour of RPL
- Complex operations like rank comparison and multicast are encoded by multiple rules
- We also assign priorities to control the order in which rules are applied

- I will only show two rules

University of Glasgow

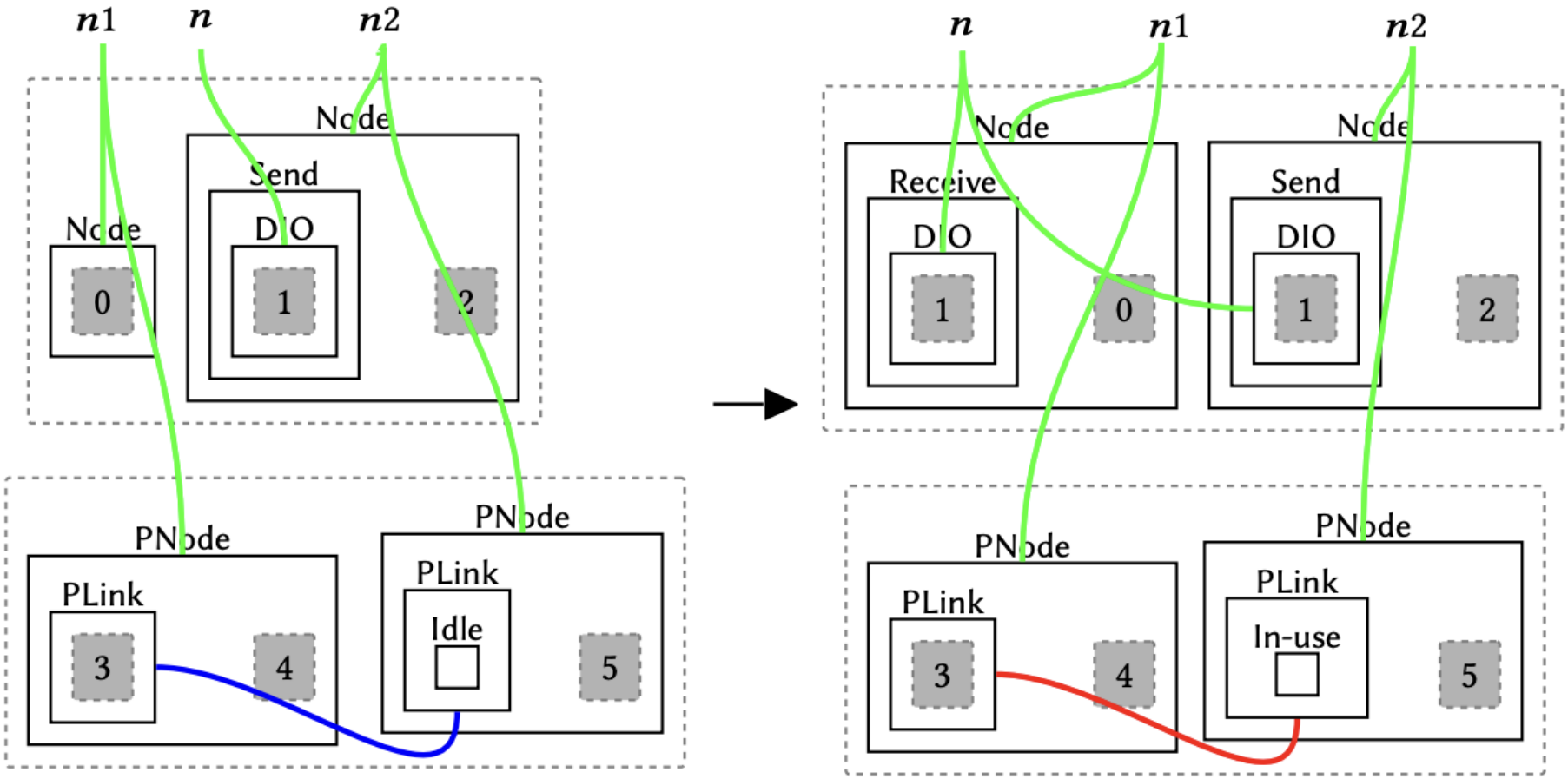
Modelling RPL dynamics: generate DIOs

We use an instantiation map to duplicate the content of Rank in the rhs



University of Glasgow

Modelling RPL dynamics: multicast DIOs

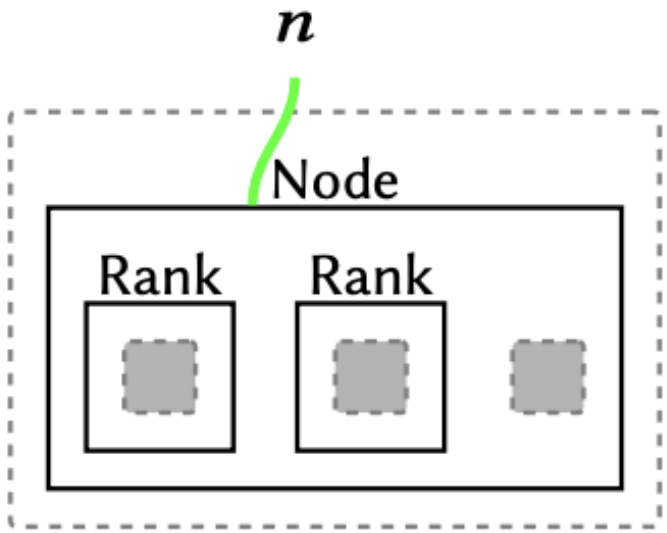


University of Glasgow

Model verification

We validate our model by checking:

- All nodes eventually join the DAG
- Nodes join with the optimal rank
- DAG is cycle-free



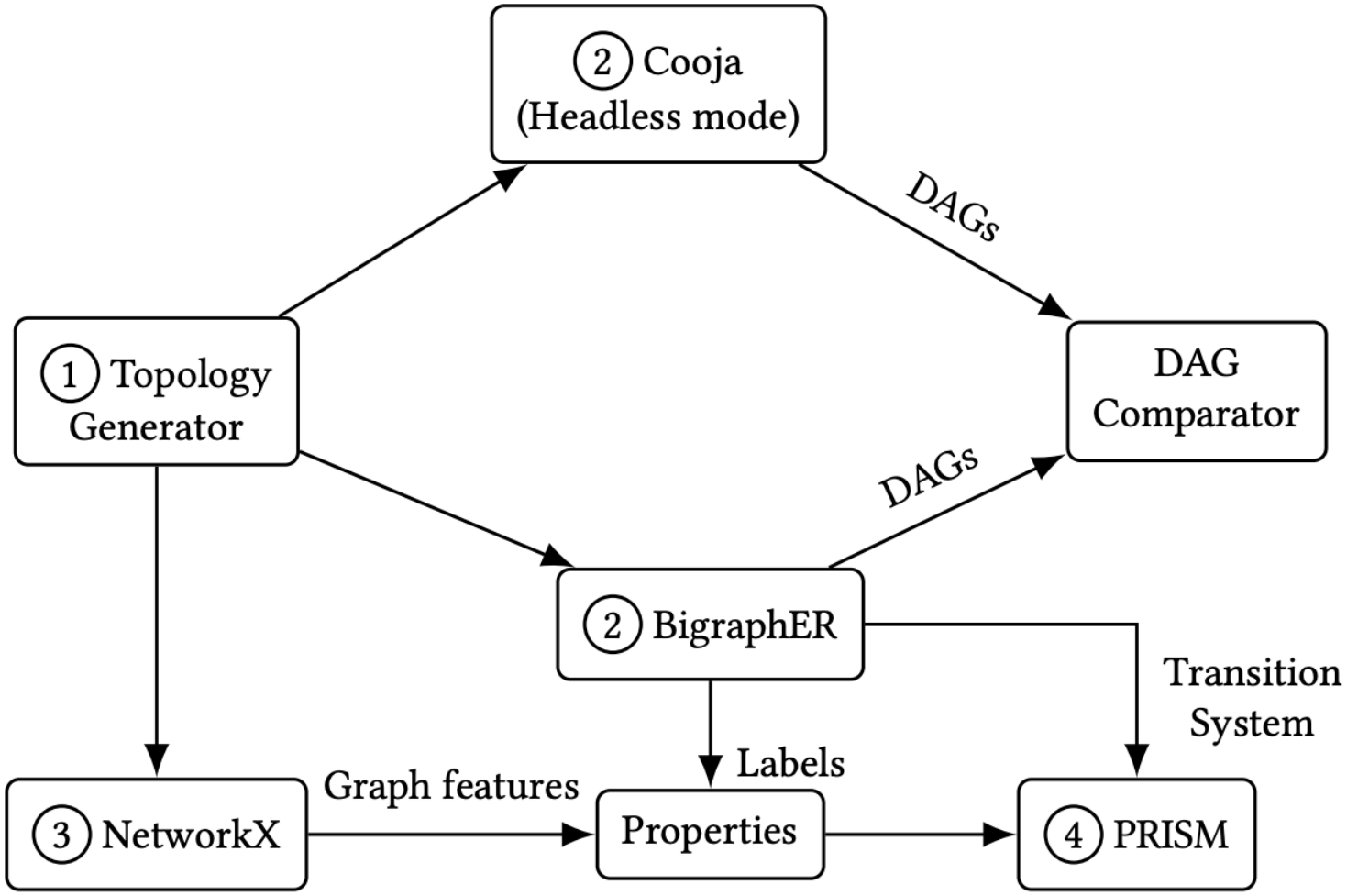
We use BigraphER to generate the labelled transition systems and PRISM to check the properties of a network topology

$$\mathbf{A}[\mathbf{G}\neg\texttt{multijoin}]$$

Experimental comparison

We compare our approach against simulation

- Cooja is a popular emulator for IEEE 802.15.4 networks
- 100 randomly generated network topologies with between 7 and 9 nodes



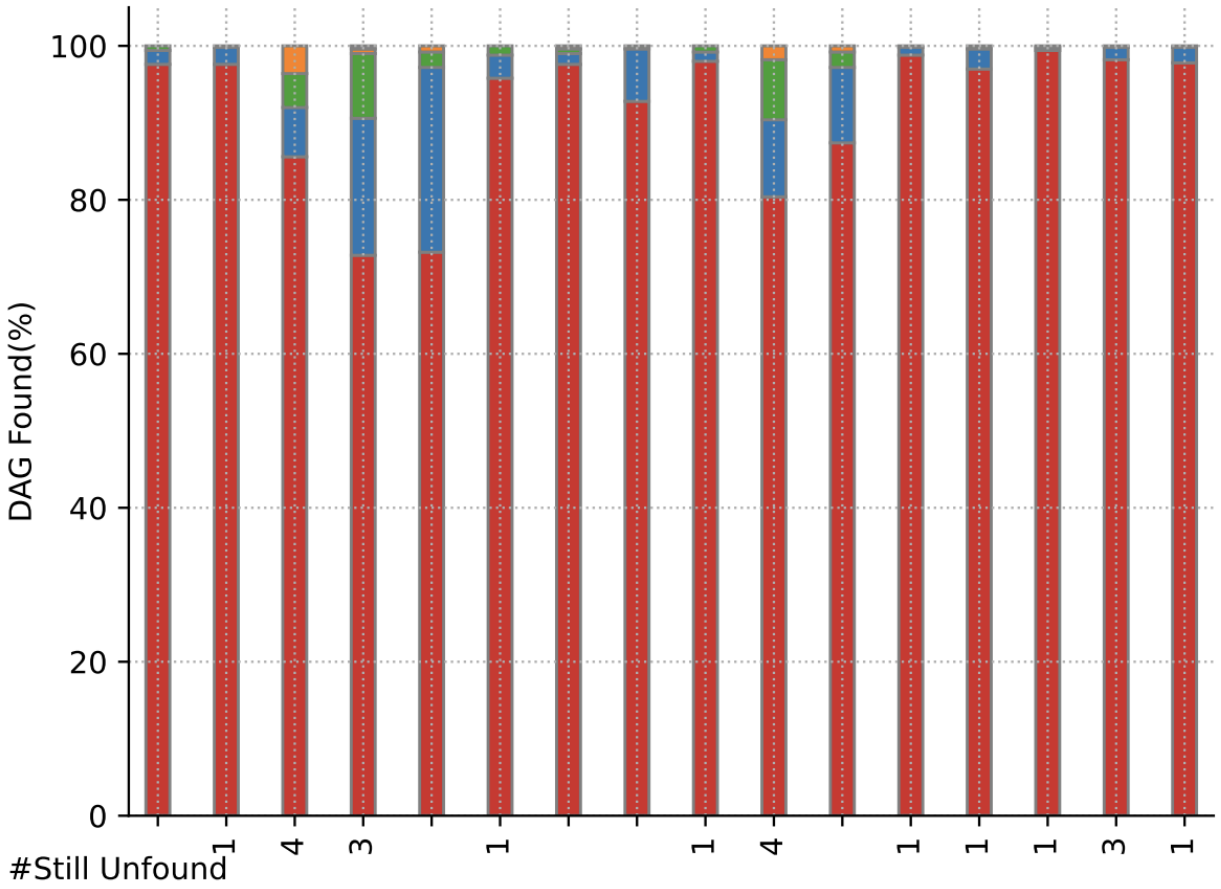
University of Glasgow

Results: coverage is the main difference

The bigraph model identifies all the valid DAGs for a topology

Cooja produces one DAG per run

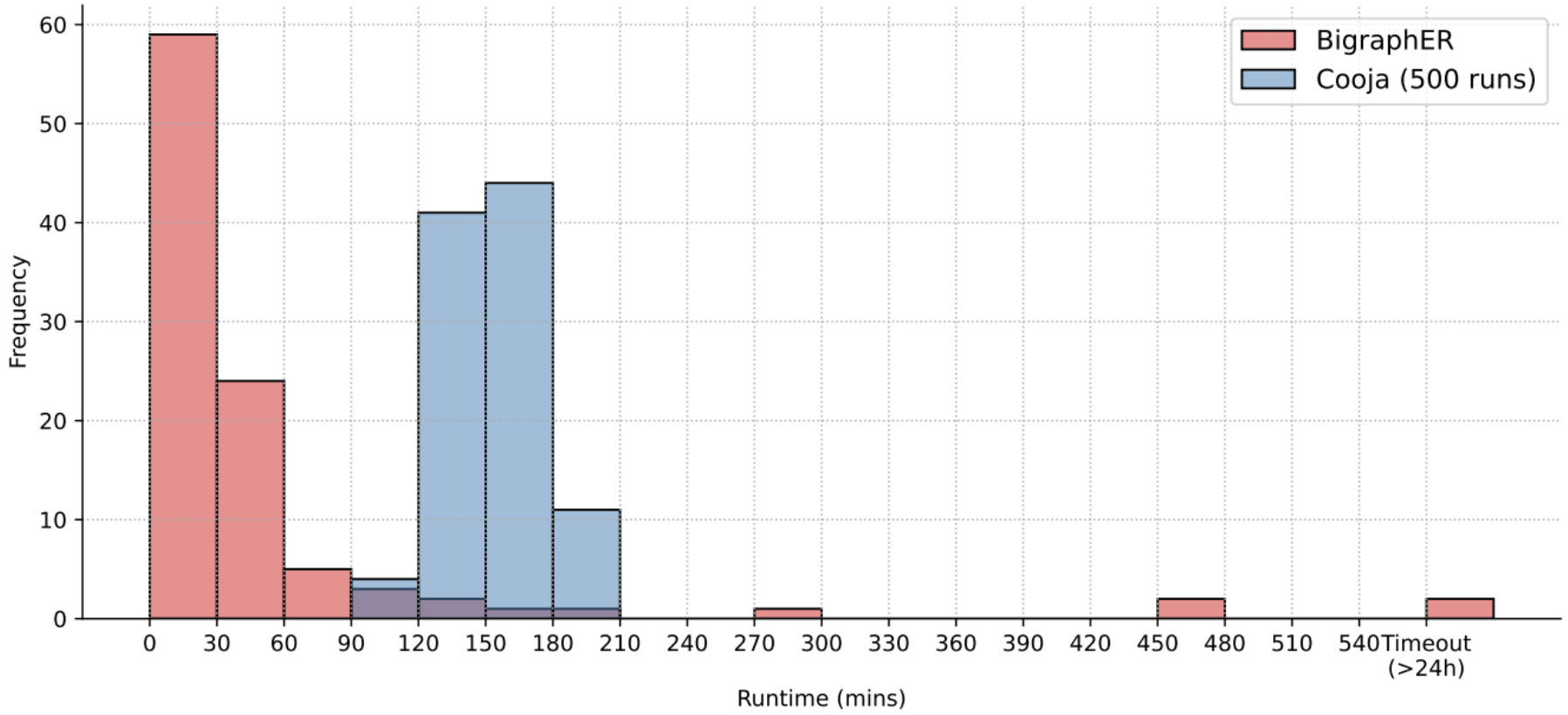
- Several runs (500) needed to find multiple DAGs but still many misses
- Simulation outcomes depend on timing and heuristics



University of Glasgow

Results: running times

Comparable performance for the analysed scale



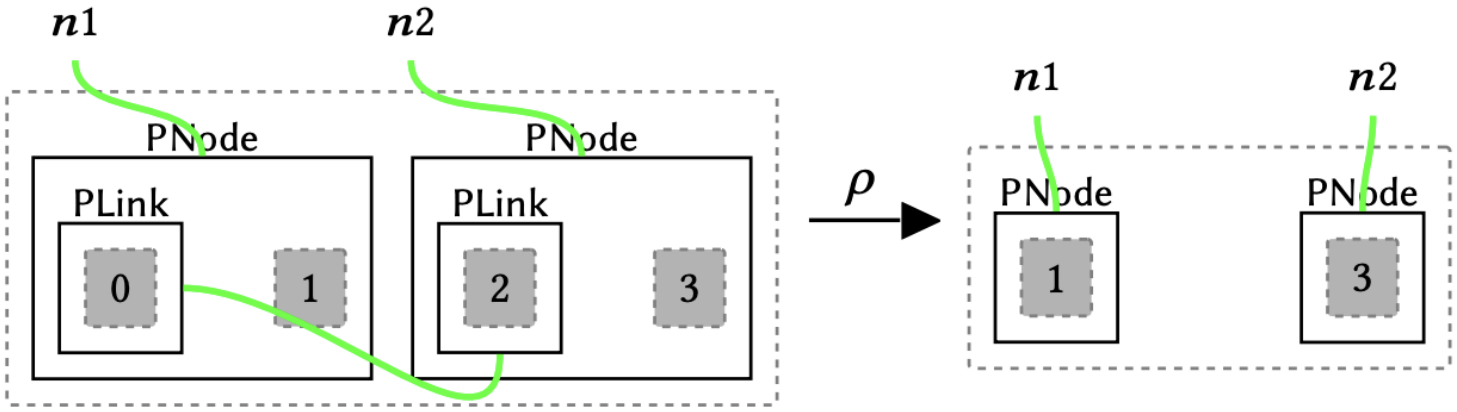
Extensibility

Low modelling overhead for design-space exploration

- Protocol variations as rule changes (add or modify reaction rules)
- Useful to explore behaviour before re-implementation

Examples:

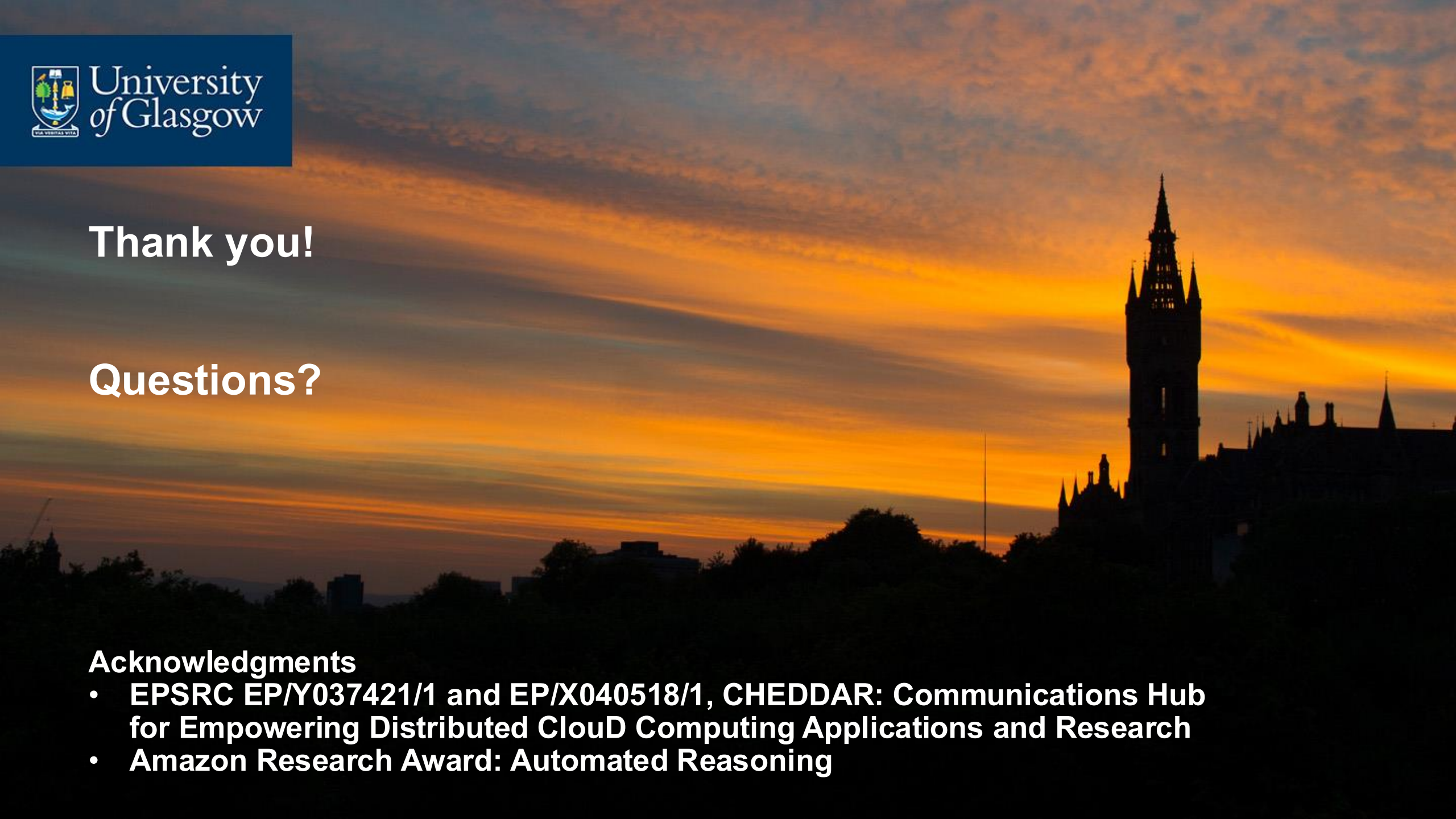
- Adversarial behaviour
- Link instability



University of Glasgow

Conclusion

- Bigraphs give a diagrammatic but formal specification
- The model is executable via BigraphER
- Verification is performed using standard model checking tools
- Exhaustive analysis complements simulation
- Rule-based models support rapid iteration



Thank you!

Questions?

Acknowledgments

- EPSRC EP/Y037421/1 and EP/X040518/1, CHEDDAR: Communications Hub for Empowering Distributed ClouD Computing Applications and Research
- Amazon Research Award: Automated Reasoning