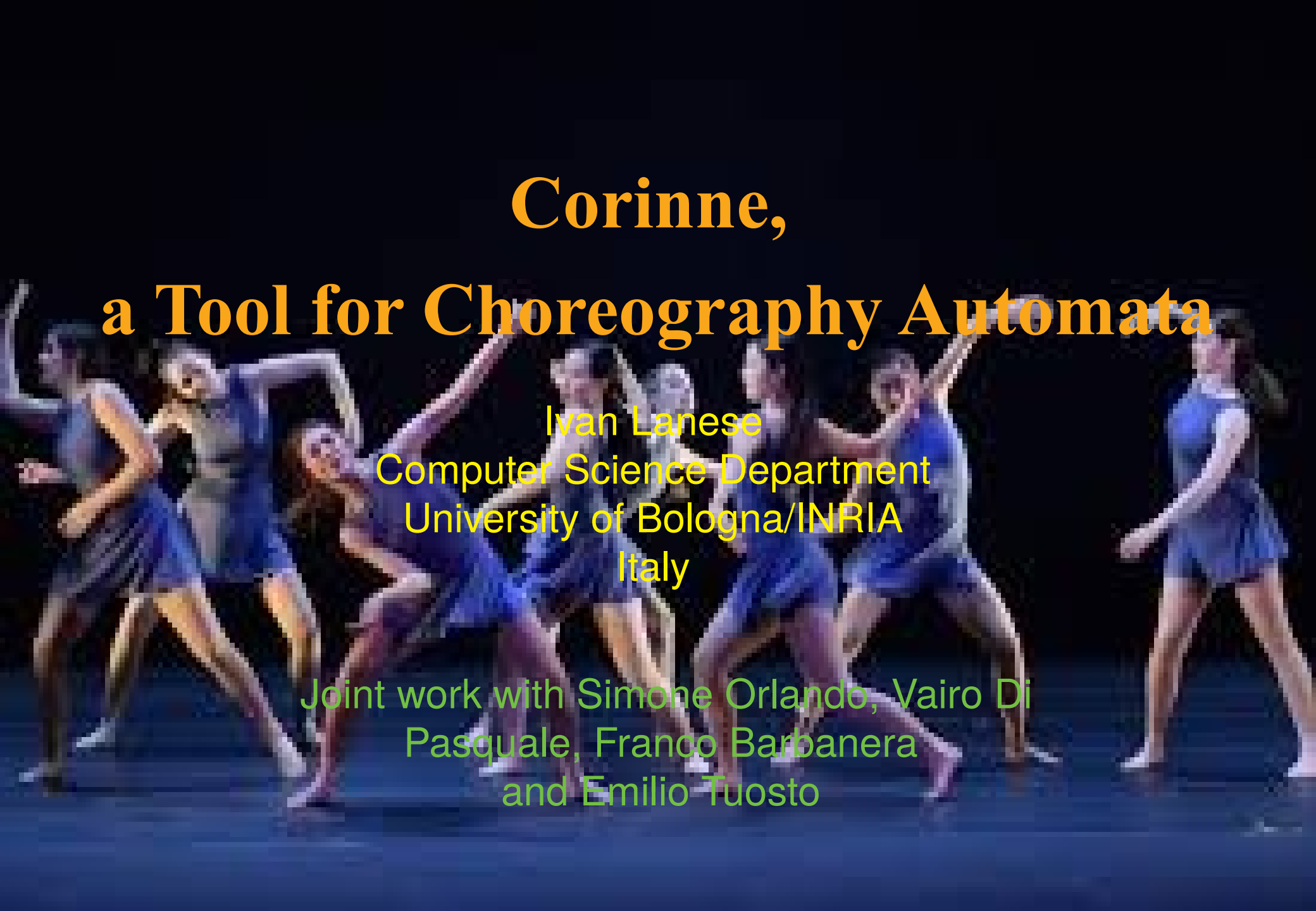


Corinne, a Tool for Choreography Automata

Ivan Lanese
Computer Science Department
University of Bologna/INRIA
Italy

Joint work with Simone Orlando, Vairo Di Pasquale, Franco Barbanera and Emilio Tuosto

Map of the talk

- Choreography automata
- Properties of choreography automata
- Composition of choreography automata
- Conclusion

Demo

Map of the talk

- Choreography automata
- Properties of choreography automata
- Composition of choreography automata
- Conclusion

Choreographic models

- Choreographic models (e.g., BPMN choreographies, multiparty session types, ...) describe the global behavior of a communicating system
- Useful to:
  - Understand the overall behavior
  - Ensure by construction or check behavioral properties such as deadlock freedom
- Equipped with a projection operation to derive the behavior required by each role

Choreography automata

- A choreographic model based on finite state automata
- Automata where edges are labeled by interactions

  $A \rightarrow B{:}m$

  Participant A sends a message m to participant B and B receives it
- Project to communicating finite state machines, one per participant

Systems of CFSMs

- A CFSM is a finite state automaton whose transitions are labelled with communication actions:
  - AB!m: A sends a message m to B
  - AB?m: B receives message m from A
- A system is composed by one CFSM per participant
- Synchronous semantics: A and B can move iff A can perform AB!m and B can perform AB?m (for some m)

Map of the talk

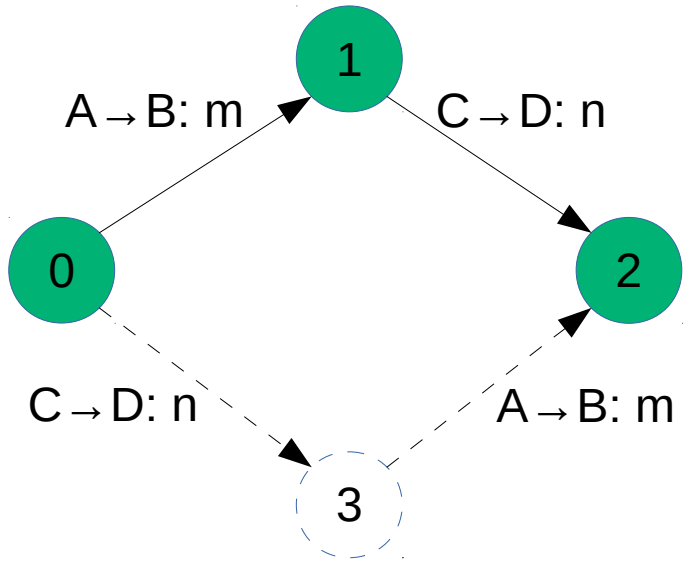
- Choreography automata
- Properties of choreography automata
- Composition of choreography automata
- Conclusion

Properties of systems of CFSMs

- We would like systems of CFSMs
  - To follow the behavior specified by the c-automaton
  - To enjoy properties such as deadlock freedom and liveness
- This can be ensured by checking two properties on the starting c-automaton [Barbanera, Lanese, Tuosto: COORDINATION 2020]
  - Well-sequencedness
  - Well-branchedness

Well-sequencedness

- If two transitions have disjoint sets of participants, then they form a commuting diamond



Well-branchedness

- If there is a choice then
  - there is a participant making the choice
  - the other participants either behave in the same way or are made aware of the choice outcome
- Formalization quite complex, intuition is enough for the purpose of this talk

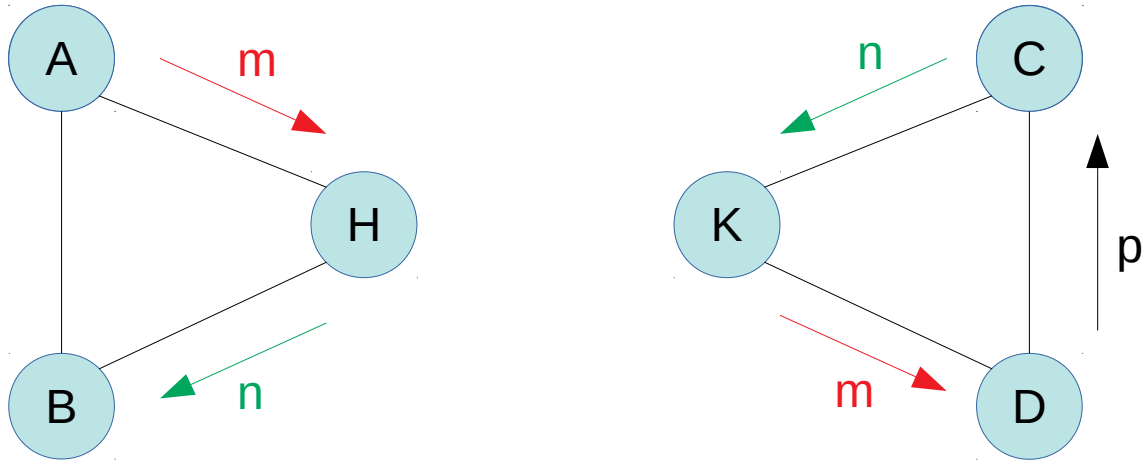
Map of the talk

- Choreography automata
- Properties of choreography automata
- Composition of choreography automata
- Conclusion

Demo

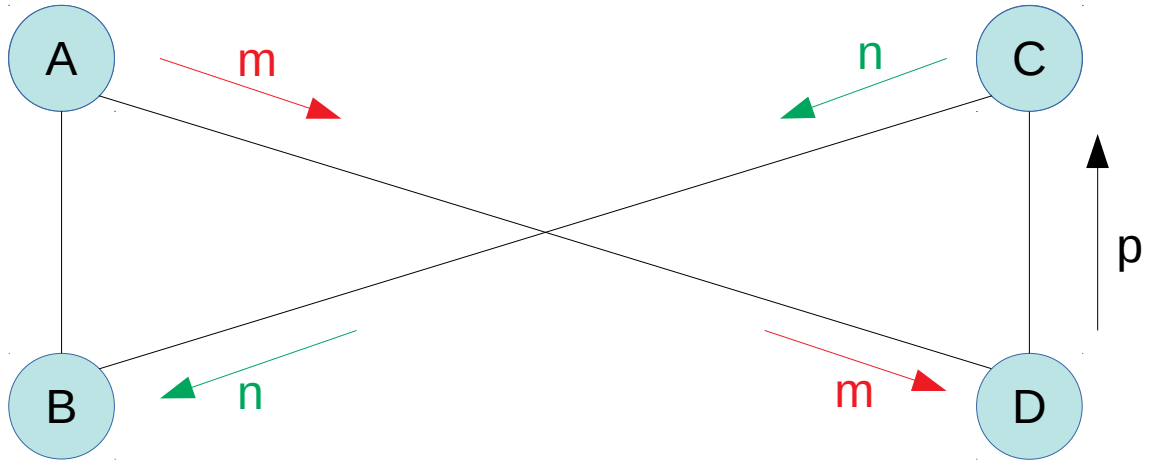
Opening systems of CFSMs

- C-automata are close: communications target other participants of the same system
- Composition idea from [Barbanera, de'Liguoro, Hennicker: JLAMP 2019]
- Open systems by selecting a participant as interface towards another system



Composing systems of CFSMs

- Systems can now be composed via dropping the interfaces and connecting the systems directly
- Any two participants be chosen as interfaces, provided they are compatible



Map of the talk

- Choreography automata
- Properties of choreography automata
- Composition of choreography automata
- Conclusion

Summary

- Corinne allows one to work on c-automata, in particular:
  - Project them
  - Check well-sequencedness and well-branchedness
  - Compose them
- Can import choreographies from other tools such as chorgram
- Allowed us to find a couple of minor bugs in the examples in our papers
- Available online at https://github.com/lanese/corinne-3
- > 2K lines of python3
- Based on tkinter for graphical interface, antlr4 for parsing and graphviz for drawing automata

Future work

- Extend the tool to support other operations
  - Other forms of composition
  - Checking properties for asynchronous semantics
  - Compute the semantics
- Refine some conditions [requires theoretical study]
  - Weaken the conditions for well-branchedness
  - Improve the complexity of the check of well-branchedness

End of talk

Thanks!

Questions?