Investigating Syntax-Guided Synthesis of Distributed Systems

Isaac Chan[1], Federico Mora[2], Lauren Pick[2], Sanjit A. Seshia[2]

[1]City College of San Francisco, [2]University of California, Berkeley

Abstract

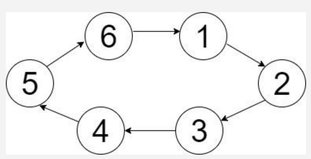
Distributed systems have broad applications, e.g., cloud services. Modeling these systems is critical for testing, but writing the models manually is often difficult and error-prone. Program synthesis, the automated creation of a program given user specifications, has the potential to produce reliable and accurate models. We investigate the synthesis of distributed systems models given the traces of existing ones. Previous work used syntax-guided synthesis, a program synthesis technique, to learn models from traces, but relies on additional instrumentation that is effort-intensive or impossible for certain systems [1,2]. Our goal is to synthesize P language programs based on traces that do not contain additional instrumentation [3].

Research Question

Is syntax-guided synthesis from traces a promising technique for distributed systems?

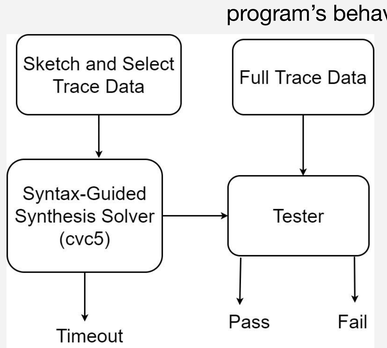
Methods

We modeled a classical distributed system in the P language



A model of the ring-leader election protocol

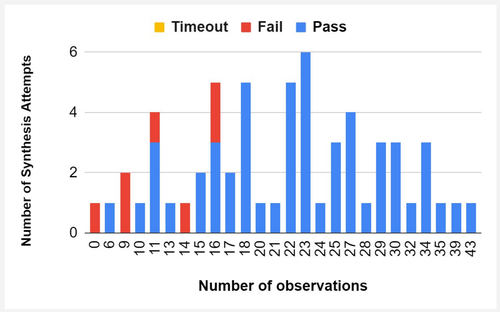
- One observation is a node's combined reception of and response to a communication
- Sketches provide a blueprint for the synthesizer
- Made up of observations, traces contain a full log of a program's behavior



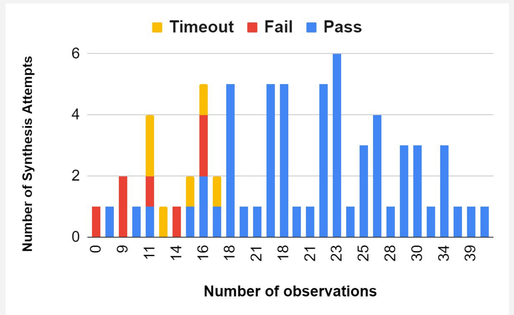
Our experimental setup

Results

We ran three sketches (easy, medium, hard) on 60 sets of trace data using CVC5, time limit of 10 minutes

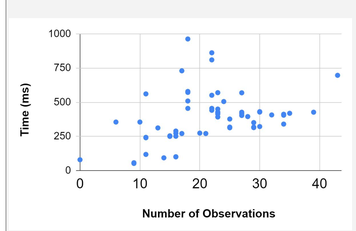


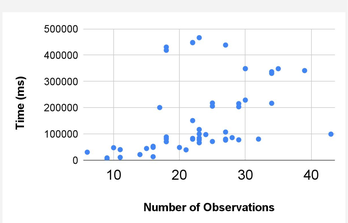
Number of synthesis attempts vs. number of observations per set of trace data (easy sketch)



Number of synthesis attempts vs. number of observations per set of trace data (medium sketch)

- Most synthesis results passed tests on easy and medium difficulty sketches, meaning synthesis succeeded for the data set
- The lowest numbers of observations appear to be associated with lower chances of success
- The most difficult sketch caused timeouts on all runs





Synthesis time vs. number of observations, easy sketch (left) and medium sketch (right)

More observations is associated with longer synthesis time

Conclusion

- Syntax-guided synthesis is generally successful in creating models of our target system
- Successful synthesis may require a certain number of observations, but more observations past this may be detrimental to synthesis time

Future Work

- Running with a longer timeout
- Improving the efficiency of synthesis through identifying the optimal characteristics and number of traces
- Applying the same synthesis process to other distributed systems

References

[1] N. Y. Jeppu, T. Melham, D. Kroening and J. O'Leary, "Learning Concise Models from Long Execution Traces," 2020 57th ACM/IEEE Design Automation Conference (DAC), 2020, pp. 1-6, doi: 10.1109/DAC18072.2020.9218613.

[2] R. Alur et al., "Syntax-guided synthesis," 2013 Formal Methods in Computer-Aided Design, 2013, pp. 1-8, doi: 10.1109/FMCAD.2013.6679385.

[3] A. Desai, V. Gupta, E. Jackson, S. Qadeer, S. Rajamani, and D. Zufferey, "P," *ACM SIGPLAN Notices*, vol. 48, no. 6, pp. 321–332, Jun. 2013, doi: 10.1145/2499370.2462184.

Acknowledgments

This work was funded by the Hopper-Dean Foundation and the Transfer-to-Excellence Summer Research Program at the University of California, Berkeley. I would like to give a very special thanks to my mentors, Dr. Lauren Pick and Federico Mora, and my PI, Prof. Sanjit Seshia, for their guidance throughout this journey. Thank you also to TTE REU staff, Nicole, Tony, Naz, and the entire TTE cohort for their support and encouragement.

Contact Information

Isaac Chan
Email: ifchan751@gmail.com