

Towards Better Interactive Formal Specifications

Will Schultz

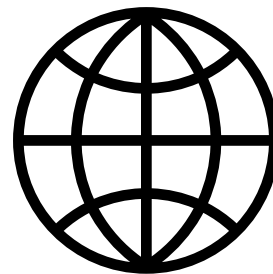
TLA+ Conference 2024

Goals: Better Interactive TLA+



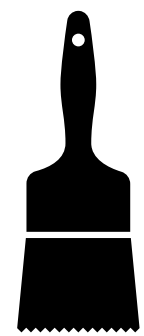
Exploring

Interactively explore behaviors of a specification.



Sharing

Easily share and link to counterexample traces.



Visualization

Dynamic viewing and animation of traces.

Our Work

Native, interactive TLA+ explorer for the browser.

Our Work

Native, interactive TLA+ explorer for the browser.

- Javascript interpreter for TLA+.



- Web interface for exploration of specs and models.



Try it!

github.com/will62794/tla-web

Javascript TLA+ Interpreter

- Our TLA+ interpreter (TLAJS) built in plain Javascript
- Builds on tree-sitter TLA+ parser by Andrew Helwer (github.com/tlaplus-community/tree-sitter-tlaplus)
- ~4000 lines of Javascript. On and off development over last 1-2 years.



Javascript TLA+ Interpreter

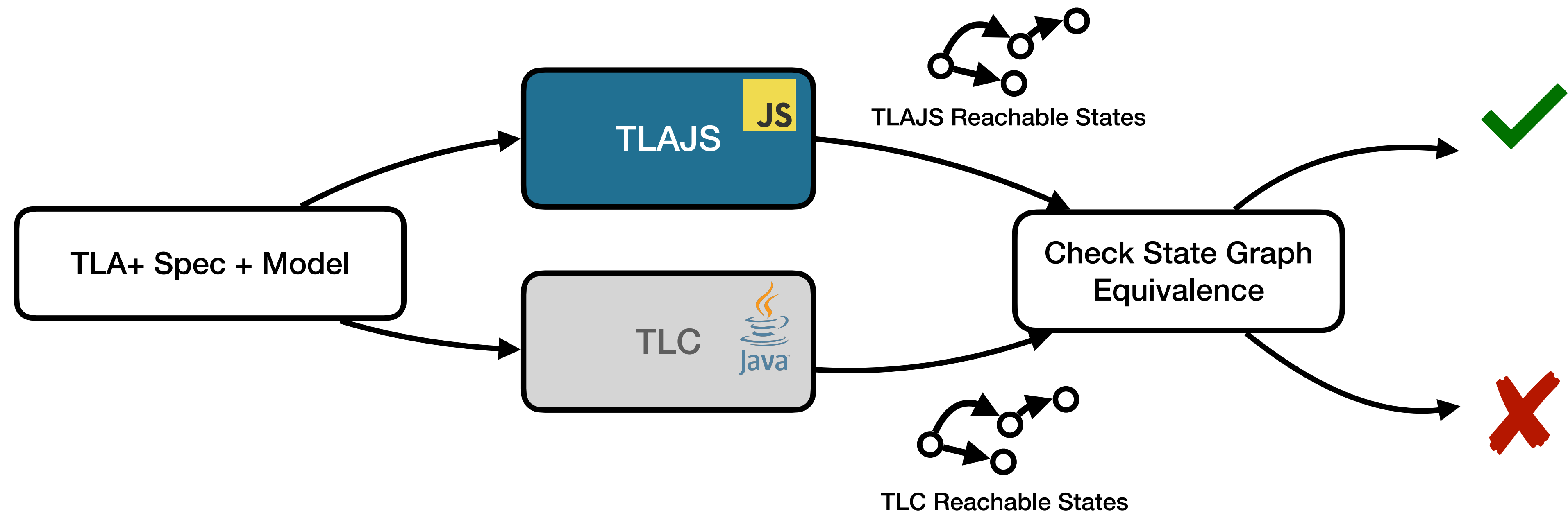
Language Feature Support

- Close to language feature parity with TLC
- Preliminary but incomplete implementation of module imports. Monolithic specs assumed as input currently
- Most standard module operators built in

Javascript TLA+ Interpreter

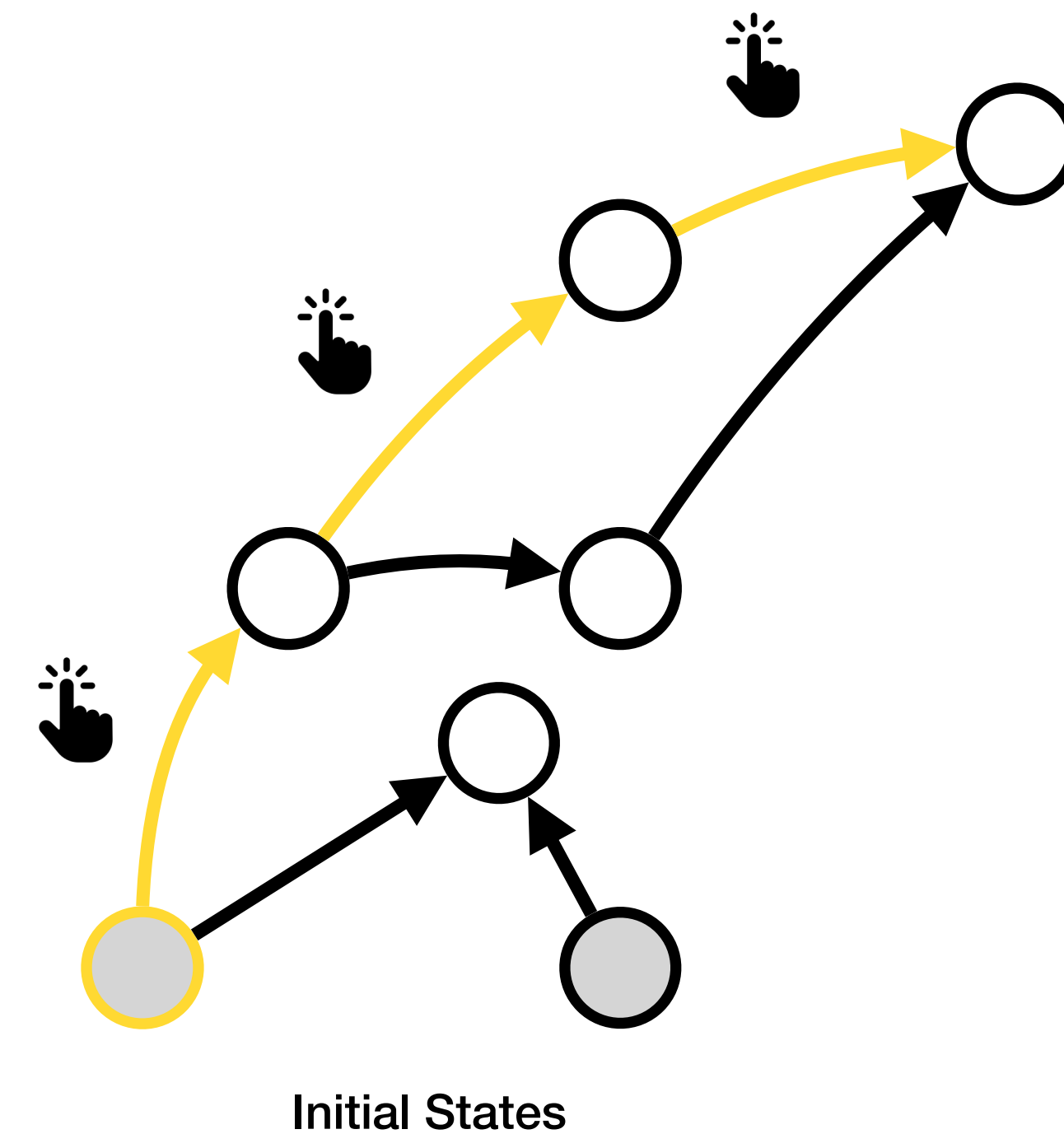
Testing

- Correctness testing done via conformance with TLC.
- For a given spec, compare TLAJS generated stategraph with TLC state graph.
- 72 specs currently tested (~2500 LOC of TLA+)



Interactive Web Interface

- Load spec from URL, dynamically instantiate CONSTANT values
- Explore traces by manually selected enabled transitions
- Trace exploration features:
 - Trace expressions
 - Hide/show variables
 - Copy link to trace
 - State visualization



Demo

Future Work

- Automatic PlusCal label detection
- Explode trace view on process/node set
- Model checking simple properties/invariants in interface
- More complete module support

Try it out!

github.com/will62794/tla-web