An Introduction to the Model Checker and Software Verification Tool:

Spin

Amshah Mushtaq and Emily Burda
Undergraduate Extra Credit Presentation
COMP 335 001

Contents

Background

- **S**imple **P**romela **In**terpreter
- Developed in 1980 at Bell Labs
- Gerard J. Holzmann and others
- Written in C coding language
- Open-source software verification tool
- Used for efficient formal verification of multi-threaded software applications

Rationale and Purpose

- Rigorous and automated verification of the correctness of concurrent software models
- Supports multi-core computers and embedded C code
- Scalability and large problem sizes
- Many successful major projects and examples
  - Control algorithms for flood control barriers
  - Algorithms for space missions, such as Mars Science Laboratory and the Mars Exploration Rovers
  - Logic verification of call processing software for a commercial data and phone switch

Example

```
// a small example spin model
 // Peterson's solution to the mutual exclusion problem (1981)

bool turn, flag[2];          // the shared variables, booleans
byte ncrit;                  // nr of procs in critical section

active [2] proctype user()  // two processes
{
    assert(_pid == 0 || _pid == 1);
again:
    flag[_pid] = 1;
    turn = _pid;
    (flag[1 - _pid] == 0 || turn == 1 - _pid);

    ncrit++;
    assert(ncrit == 1);      // critical section
    ncrit--;

    flag[_pid] = 0;
    goto again
}
// analysis:
// $ spin -run peterson.pml
```

Description and Classification

- Traces logical design errors in distributed systems, such as operating systems, concurrent algorithms, and nuclear power plants
  - Different from TLA+ because Spin generates C sources for problem-specific model checkers
- Provides notation for expressing correctness requirements
- Checks logical consistency of a specification (design choices and correctness requirements)
  - Deadlocks, race conditions, incompleteness, and assumptions regarding process speeds

Description and Classification

- Formal models are built using Promela (**Pro**cess **Me**ta **La**nguage)
  - Promela is Spin's input language
- Promela models can be derived from concurrent C code
  - Models are then verified with Spin
  - Model extractor modex is used
- 4 modes of Spin
  - Simulator, exhaustive verifier, proof approximation system, and driver for swarm verification

Evaluation!

- Good **readability** and **writability**
  - Overall simplicity, not too many features, and easy to learn and read
    - Graphical user interface extension that makes SPIN more user friendly
    - Verification model has fewer statements than the simulation model to keep the state space as small as possible when specifying the essential behavior of the model

➔ Good **readability** and **writability** cont.

- Provides direct support for the use of embedded C code as part of model specifications
- Provides direct support for the use of multi-core computers for model checking runs while supporting safety and liveness verifications

◆ There is also support for abstraction

- Simulator and verifier are separate so that you can simulate models with low levels of abstractions instead of constructing a model first from a high level and working down

➔ Good **reliability, scalability,** and **cost**

- Exception handling, compilation, maintenance
  - The simulator is helpful at uncovering obvious bugs and analyzing counter-examples
  - Can perform random simulations that provide a good view of the operation of the system during the initial design phase
  - Can perform interactive simulations to explore suspected problem areas of design
  - Can perform guided simulations to uncover the cause of an error that created a counter-example during a verification run

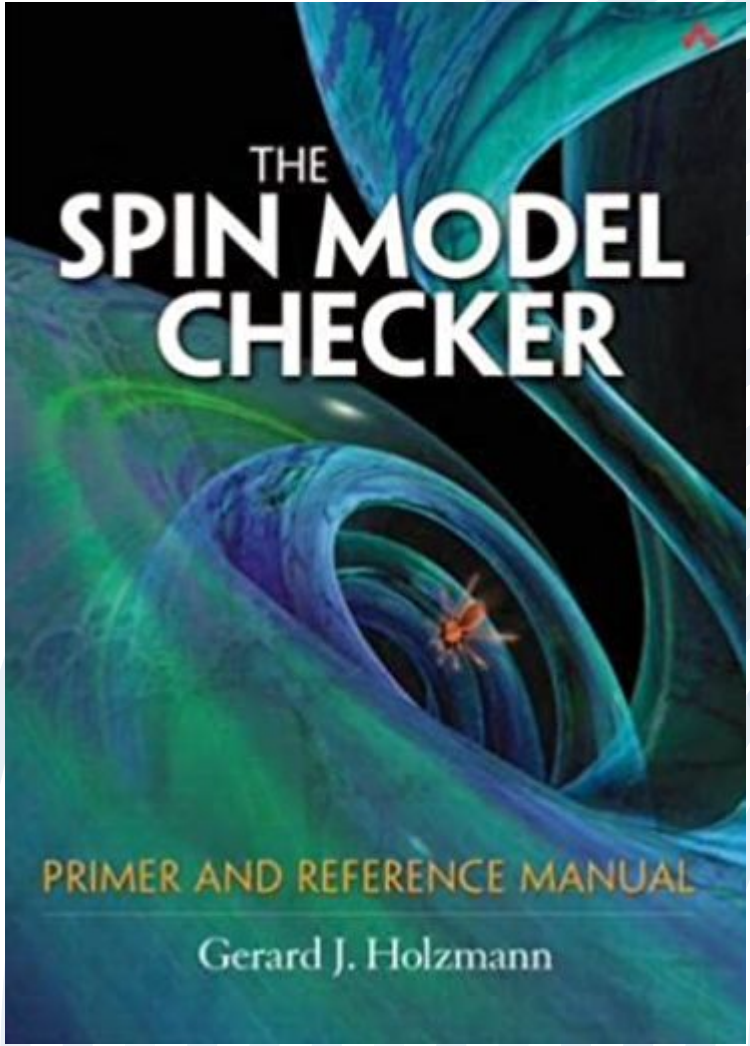
- Traces logical design errors and checks the logical consistency of a specification
- Does not need to pre construct a global state graph before verifying system properties
- Designed to scale well and handles very large problem sizes well

Community, Ecosystem, and Coolness

- Many books and resources
  - The SPIN MODEL CHECKER: Primer and Reference Manual by Gerard J. Holzmann
  - Online tutorials on Spin website
- Community
  - Forum: spinroot.com/fluxbb/
  - Annual Symposia April 26 to 27, 2023 in Paris
- Involvement with NASA, Toyota, and more

Conclusion

Spin is a popular tool used internationally that rigorously verifies the correctness of multi-threaded software applications



Thank You!

Any questions or comments?

https://spinroot.com/spin/whatispin.htm