Correctness of Parallel Programs: Testing, Debugging, Specification, and Verification

Jacob Burnim, Tayfun Elmas, Nicholas Jalbert, Pallavi Joshi, Chang-Seo Park, Koushik Sen, Christos Stergiou

David Gay, Costin Iancu, Madan Musuvathi, Mayur Naik, George Necula, Cristiano Pereira, Gilles Pokam, Shaz Qadeer

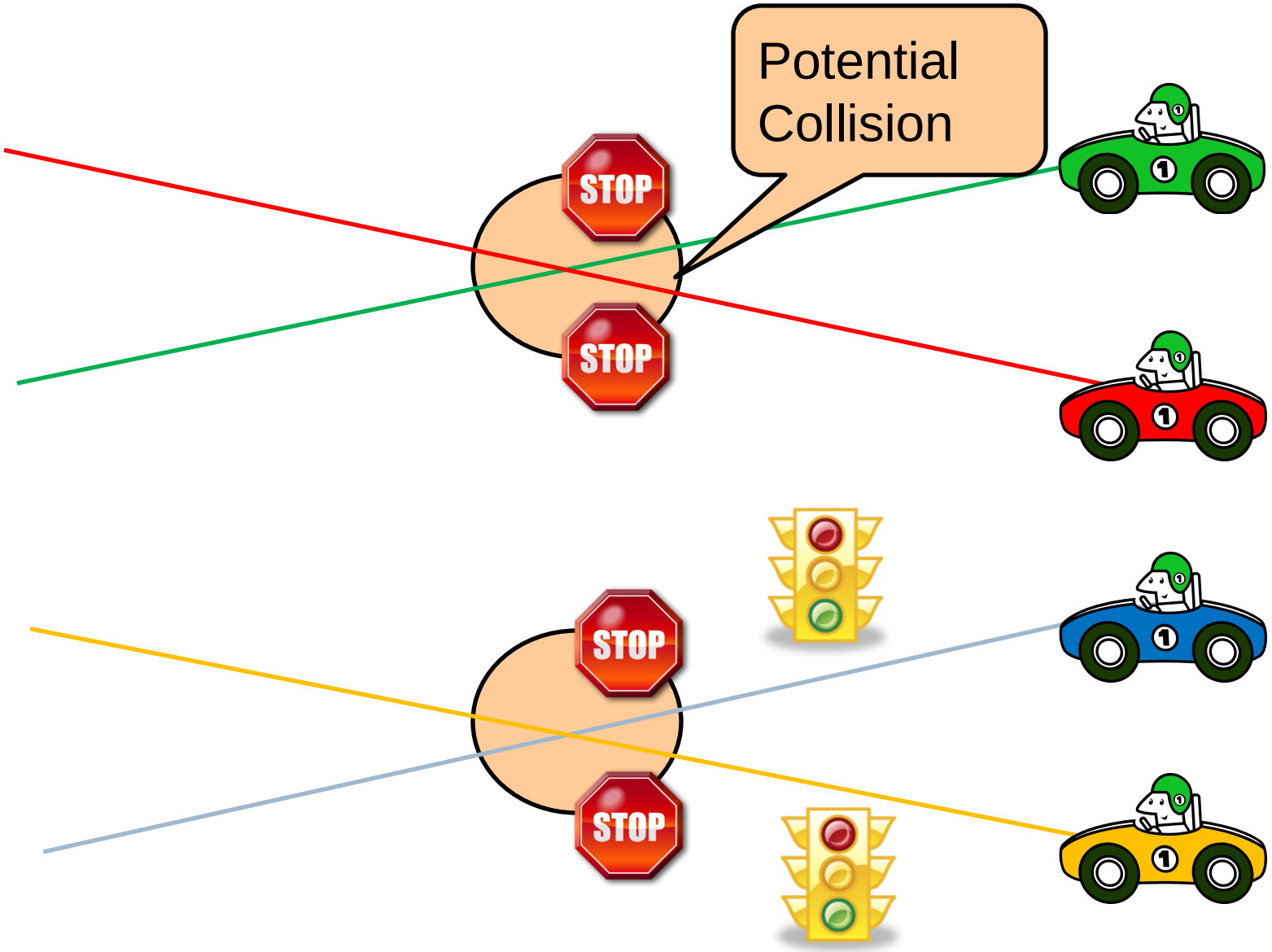
Where We Started

- Difficult to write correct parallel software.
  - **Key:** Interference between parallel threads
  - **New bugs**: races, atomicity, deadlocks, …
  - Complicates reasoning, testing, debugging
- Great need for tools to automatically find and reproduce parallelism errors.
  - **Our Technique: Active Testing**

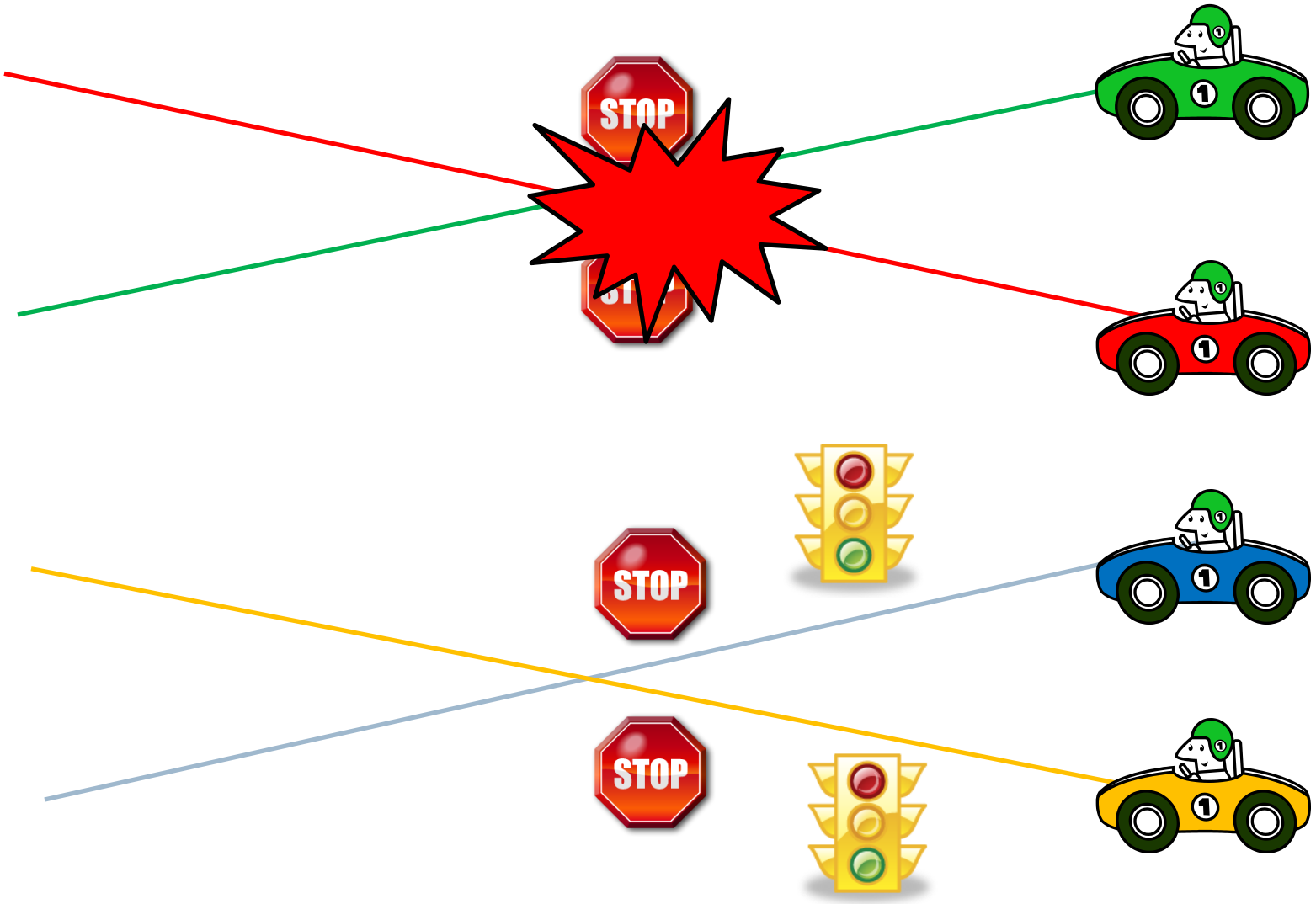
Active Testing: Key Ideas

- **Existing Techniques:**
  - Detect and predict **possible bugs** (static and dynamic program analysis)
  - Bias/control the thread schedule (stress testing, noise making, model checking, etc.)
- **Active Testing**: Use both.
  - **Phase I**: Dynamic or static program analysis to predict **possible** bugs
  - **Phase 2**: Biased random execution to find and reproduce real executions exhibiting bugs

Active Testing I: Prediction



Active Testing II: Directed Testing



Active Testing Advantages

- **Key:** Practical, easy-to-use testing tools, with sophisticated analysis internally.
  - Tested Java, C programs up to 500 kLoC
  - Finds many bugs quickly
  - Finds rare bugs with high probability
  - Creates an actual run showing a bug

Active Testing Successes

- Found many parallel bugs in apps, libraries.
  - Data races [PLDI ‘08]
  - Atomicity and typestate errors, and deadlocks [FSE‘08, ASE‘08, ICSE’09 (**best paper**), PLDI‘09]
  - Memory consistency errors [ISSTA ‘11]
  - Simplifying error traces [FSE ’10]
- **CalFuzzer** for Java, **THRILLE** for C
  - Bug benchmark suite [HotPar ‘11]
  - Commercial tool **ZVM** from Parallocity

Demo: Active Testing for UPC

- **Goal:** Make active testing to work for UPC and other PGAS languages.
  - **Challenge**: Efficient for 1000's distributed cores
- **UPC-THRILLE:** Active testing for UPC
  - Effective race detection, reasonable overhead
  - Now official part of Berkeley UPC
  - Winter '11 retreat, SC '11, PPoPP '13, ICS '13
  - Chang-Seo Park will now demo
  - Testimonial from Costin Iancu at end of talk