Defines *interaction semantics* and uses them to prove that a 13-pass compiler is correct and link-safe.

Correctness means that the target program shows the same trace of behaviors as the source. Observable behaviors are halting, external calls, memory writes, and memory frees.

Linking is defined as a function taking a set of interaction semantics to one large interaction semantics (with a heterogenous call stack). Link safety means that (A) linking a whole program, then compiling it and (B) compiling a set of programs, then linking them give the same result.

The interaction semantics are the crucial part of this work, as both correctness & link-safety are stated in terms of them. An interaction semantics is a 5-state operational model of a program. The states are: `initial`, `running`, `halted`, `at_external`, and `after_external`.

Strengths

- They did it! Go Princeton!
- I think it's a strength that there's nothing really complicated in the proofs & proof strategies. They did the straightfoward, meticulous thing; in my opinion justifying the "informal" reasoning compiler writers would previously do.

Weaknesses

- What problems does this solve? Section 4 lists two requirements for external calls: do not rely on memory that will be removed by the compiler or memory introduced by the compiler. Is this year-long, thousand-line effort just for that? If so, what did we learn to make future proofs easier (besides to keep a very detailed memory model).
- They removed optimizations! The optimizations are the main selling point of CompCert over other (non-Coq) high-assurance compilers. These compositional techniques *need* to scale to the full CompCert compiler.
- (Not a flaw of the paper, but) Section 6 says "we have not yet applied much proof automation at all, so we believe there is room for improvement" [1]. The original CompCert paper made a similar claim. Still, I have yet to see *any* large Coq project where proof were within 2x lines of code. Where are the proof automation success stories?

References

[1] Gordon Stewart, Lennart Beringer, Santiago Cuellar, and Andrew W. Appel. Compositional compcert. In *POPL*, 2015.