1. **Goal: Compositional Reasoning for Simulink**

   - **Compositional Static Analysis:** detect inconsistencies, compute preconditions, eliminate internal variables, check substitutability (when can a block replace another?), etc., at compile-time, without flattening!

   - **Compositional Dynamic Analysis:** generate top-level contracts and check consistency.

   - **Formal Verifier:** verify contracts using model checking.

2. **The RCRS Toolset (Refinement Calculus of Reactive Systems)**

   - Challenge: How to represent graphical diagrams in a textual notation with formal semantics?
   - **Block diagrams** represented as **atomic monotonning predicates/property transformers (MPTs).**
   - Some examples:
     - **Stateless basic block:**
     - **Non-atomic basic block:**
     - **Continuous-time stateful basic block:**
     - **Feedback composition:**
     - **Feedbackless translation strategy:**

3. **The Algebra of Hierarchical Block Diagrams (HBDs)**

   - **Basic blocks** represented as atomic monotonic predicate/property transformers (MPTs). Some examples:
     - **Statistical blocks**:
     - **Discrete-time statistical block**:
     - **Continuous-time statistical block**:
     - **Feedback composition**:

4. **Translating HBDs to Algebraic Terms**

   - **Challenge:** Given a graphical diagram, many possible translations:

5. **Formal Analyzer: Expansion and Simplification**

   - **Simplify HBDs:**
   - **Expand HBDs:**
   - **Simplify**:

6. **RCRS: A Contract-Based Framework with Refinement**

   - **Outline:**
   - **Horizontal contracts:**
   - **Vertical contracts:**
   - **Refinement:**

7. **Case Study: A Fuel Control System (FCS)**

   - **System Description:**
   - **Model Description:**
   - **Verification:**

8. **Publications**

   - **S. Tripakis.** "Compositional Semantics and Analysis of Hierarchical Block Diagrams." FPS 2016.