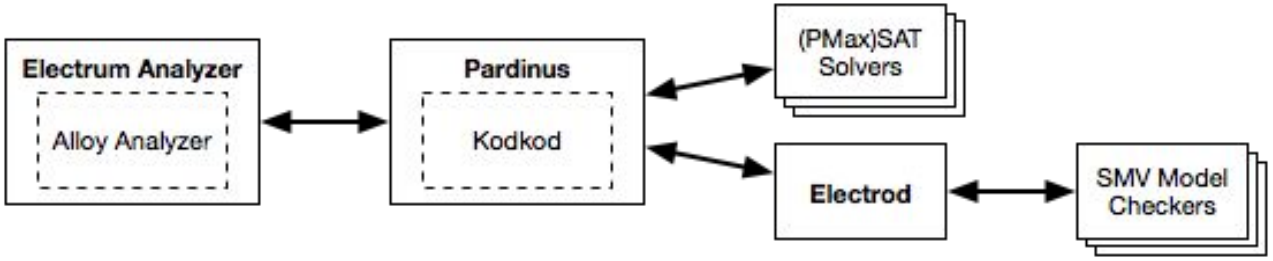
On Extending Kodkod to Support Temporal Features and Scenario Exploration

Nuno Macedo and Alcino Cunha et al
HASLab, INESC TEC & Universidade do Minho

Context

- Our group uses Alloy in research, teaching and consultancy
- We have explored extensions to problem expressibility, scenario exploration and solving procedures
  - target-oriented model finding
  - decomposed parallel solving strategy
  - symbolic relation bounds
  - dynamic relations and linear temporal formulas
- Required adapting or extending Kodkod, unified into a single release - *Pardinus*
- Testbed for functionalities, used under Electrum



Kodkod Model Finding

- Problem definition
  - universe of atoms

    ```
    {a,b}
    ```
  - relations declared with upper- and lower-bounds (tuple sets)

    ```
    r : {} {a,b}        s : {} {a,b}
    ```
  - first-order relational formulas
- Solving
  - SAT solvers
  - incremental solving for solution iteration
  - symmetry breaking
- Scenario exploration
  - generate solution to problem
  - new problem discarding previous solution

Target-Oriented Model Finding

- Problem definition
  - relations may have **targets** assigned (tuple sets between lower- and upper-bounds)

    `r : {} {a} {a,b}` `s : {} {b} {a,b}`
  - improved expressibility (search for optimal solution)
- Solving
  - PMaxSAT solvers
  - Nicely fits Kodkod's architecture, but solvers still unpredictable
  - how to perform symmetry breaking?
- Scenario exploration
  - generate minimal/maximal solutions to problem
  - solution with minimal/maximal changes from the previous solution

Decomposed Model Finding

- Problem definition
  - set of **partition** variables (define *configurations*)

```
r : {} {a,b}
```

  - manual or automatic criteria
- Solving
  - staged, generate configurations, then try to extend to full solutions in parallel

```
r : {a} {a}          s : {} {a,b}
r : {b} {b}          s : {} {a,b}
                ...
```

  - large performance gains for certain classes of problems
  - symmetry breaking preserved
- Scenario exploration
  - focus on alternative configurations
  - challenging since configurations solved in parallel

Model Finding with Symbolic Bounds

- Problem definition
  - bounds are **symbolic**, relational expressions over relations + tuple sets

    `r : {} {a,b}` `s : {} r`
  - cleaner bounds, but no added expressibility
- Solving
  - bounds are resolved into tuple sets prior to plain SAT solving
  - establish dependencies between relations, used in decomposition criterion
  - resolution of symbolic bounds results in smaller search spaces when decomposed

    `r : {a} {a}` `s : {} {a}`
- Scenario exploration
  - NA

Temporal Model Finding

- Problem definition
  - relations declared as **static** our **dynamic** with upper- and lower bounds **traces**

    `r : {} {a,b}` `s : [{},...,{}] [{},...,{a,b}]`
  - first-order relational **LTL** formulas
  - search within a range of **trace lengths**
- Solving
  - **bounded**: problem expanded into plain Kodkod with state idiom
  - **unbounded**: translation into SMV (through Electrod)
  - can we break symmetries specific to traces?
- Scenario exploration
  - solution with minimal trace length
  - solution with minimal/maximal states
  - solution fixed with a known prefix
  - solution with same/different static configuration
  - ...