Formal Methods

LaBRI

- Computational linguistics.
- Logic, graphs and languages.
- Modelling and Verification.

Computational linguistics (Retoré, Marlet)

Project SIGNES: Common project of LaBRI, INRIA Futur and Language department of Bordeaux III.

French sign language

- Accumulating and annotating the examples.
- Grammatical analysis of the language. Formalisation in generative grammars.

Generative grammars

- A tighter integration of morphology and syntax. Automatic computation of semantic representations from syntactic analyses.
- Generative grammars for language fragments.

Grail

- Tool for the development and prototyping of grammar fragments for categorical logics.
- Annotated dictionary of Dutch language.

Logics, graphs and languages

Objects of study

- Models of: computer systems, data structures, semantic representations.
  - Words, trees, graphs, hyper-graphs.
- Formalisms to talk about their properties.
  - First-order logic, monadic second-order logic, the mu-calculus
- Verification, synthesis, evaluation of queries.
  - Automata, equation systems, algebra, games.

LGL: cont (1)

Decomposition and structuring of graphs (Courcelle)

- Combinatorial structure: tree-width, clique-width,...
- Algebraic decompositions: vertex replacement, hyperedge replacement.
- Research directions:
  - Better understanding of the theory (Proof of Seese conjecture)
  - Application to query evaluation, routing, representation of maps.

External definitions (Senizergues, Walukiewicz)

- Rewriting systems.
- Graphs of configurations of machines (eg. pushdown automata).
- Extensions to higher-order stacks, recursive program schemes.
- Verification problems for such graphs.

LGL cont. (2)

Non-regular languages (Senizergues, Weil)

- Decidability of language equivalence of deterministic pushdown automata (DPDA). Extension to decidability of bisimilarity.
- Automata of iterated stacks.
- Computation in standard algebraic structures: monoids, groups.
- Traces: words with concurrency information.

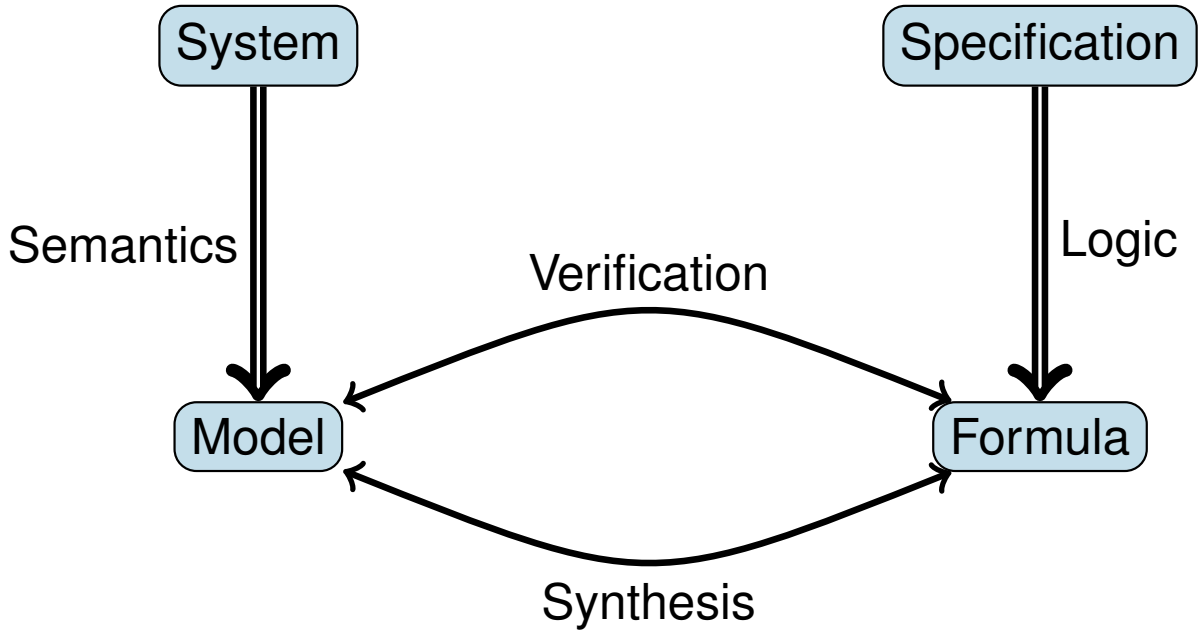
Visualization and Simulation of Distributed Algorithms (Mosbah)

- Simulation and visualization environment for distributed algorithms.
- On the crossroads of distributed algorithms, graph theory and rewriting systems.

Coq: Proof assistant (Casteran)

- Formalisation of induction principles (ordinals)
- Verification of a Java-card.
- The first book on Coq: Y. Bertot and P. Casteran.

Modelling and Verification



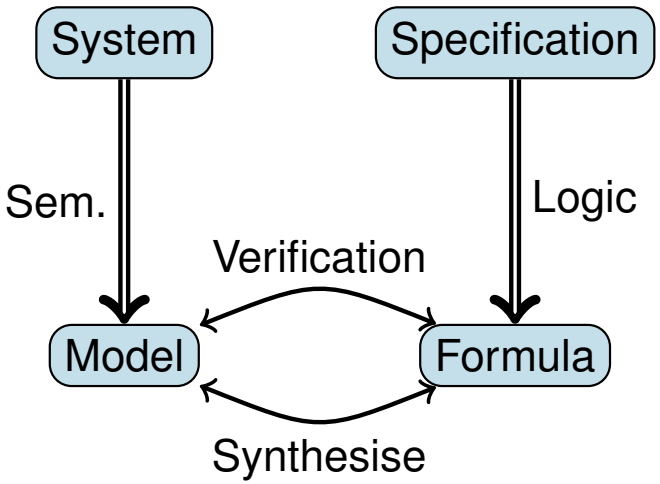
Verification (model checking) [Arnold, Clarke, Emerson, Sifakis]

Given a model M and formula α check if M satisfies α.

Synthesis

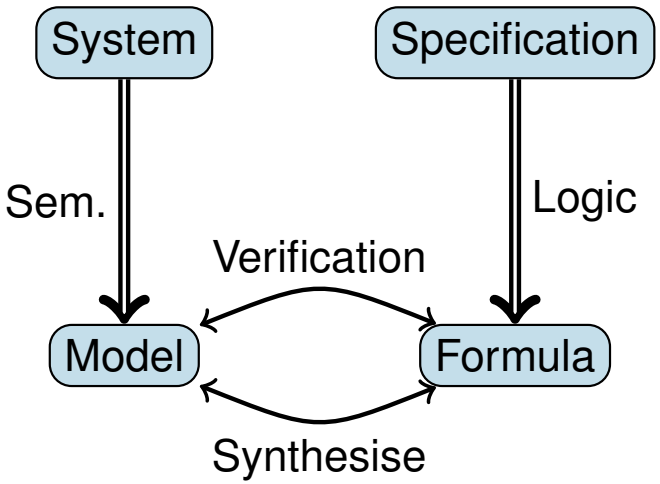
Given a formula α, find a model M that satisfies α.

MV: theory (Janin, Muscholl, Walukiewicz, Zeitun)



- Formalisms for verification.
- Game theory
- Verification of recursive programs.
- Verification of concurrent systems.
- Synthesis.

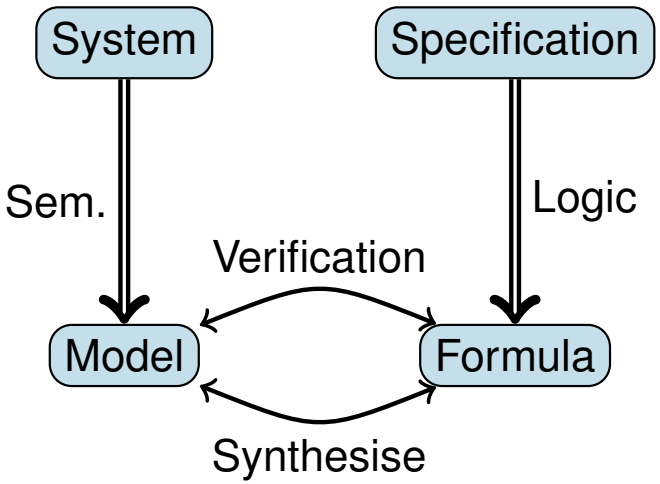
MV: THEORY (JANIN, MUSCHOLL, WALUKIEWICZ, ZEITUN)



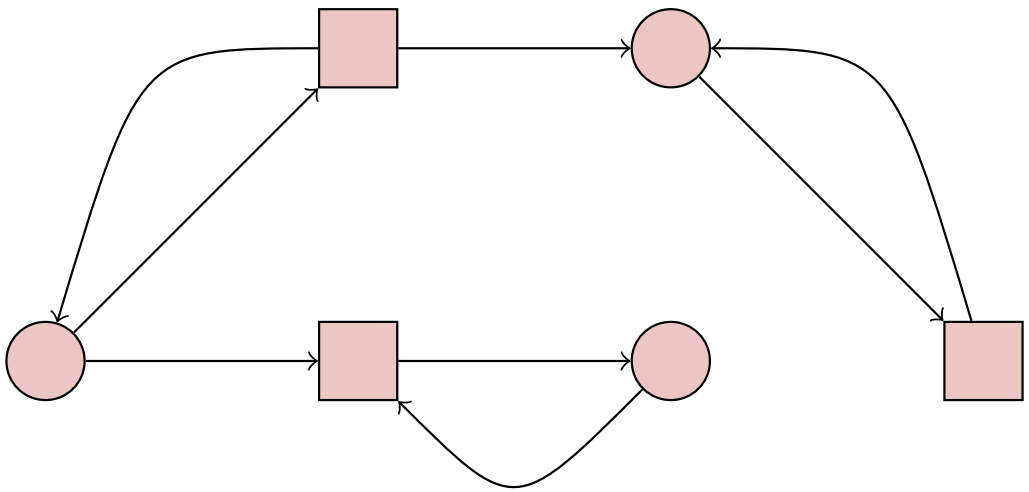
- Formalisms for verification.
  - Expressive power.
  - Algorithmic properties (model checking, satisfiability)
- Game theory
- Verification of recursive programs.
- Verification of concurrent systems.
- Synthesis.

- First-order logic,
- monadic second order logic,
- modal logics, the mu-calculus,
- temporal logics, LTL, CTL

MV: THEORY (JANIN, MUSCHOLL, WALUKIEWICZ, ZEITUN)

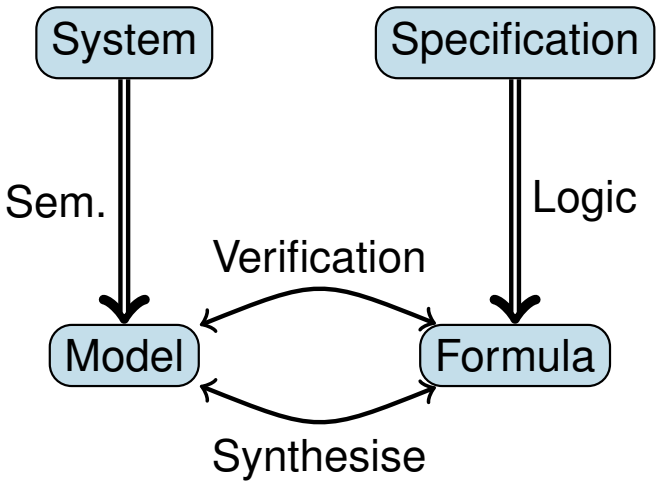


- Formalisms for verification.
- Game theory
  - Solving games.
  - Quality of strategies.
- Verification of recursive programs.
- Verification of concurrent systems.
- Synthesis.



Infinite duration two player games. Winning conditions of type: Büchi, Muller,. . .

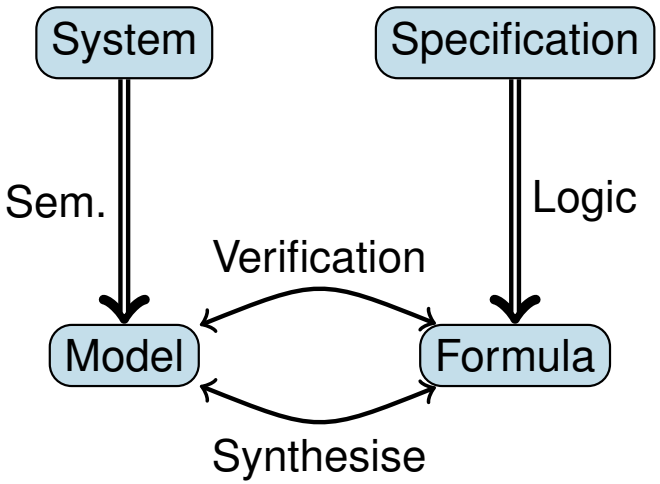
MV: THEORY (JANIN, MUSCHOLL, WALUKIEWICZ, ZEITUN)



- Formalisms for verification.
- Game theory
- Verification of recursive programs.
  - Theory of stack automata and recursive schemes
  - Program semantics.
- Verification of concurrent systems.
- Synthesis.

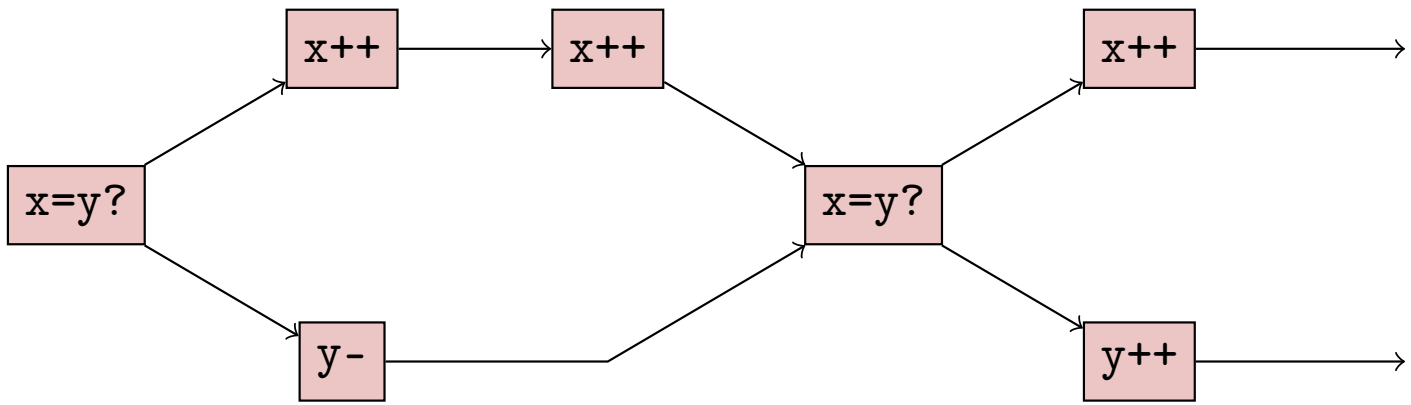
```
Fib(n+2) := Fib(n)+Fib(n+1);


map(f,list)=cons(
                 f(head(list)),
                 map(f,tail(list))
                );
```

MV: THEORY (JANIN, MUSCHOLL, WALUKIEWICZ, ZEITUN)

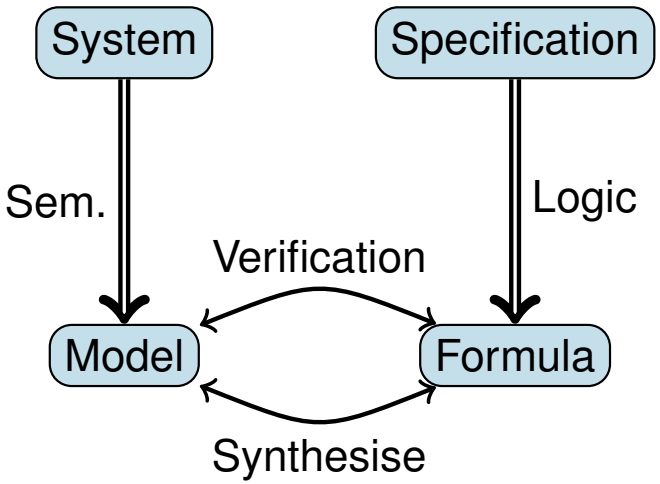


- Formalisms for verification.
- Game theory
- Verification of recursive programs.
- Verification of concurrent systems.
  - Trace theory, logics for traces.
  - Unfolding
- Synthesis.

$$P_1 || P_2 || \ldots || P_n$$



MV: THEORY (JANIN, MUSCHOLL, WALUKIEWICZ, ZEITUN)



- Formalisms for verification.
- Game theory
- Verification of recursive programs.
- Verification of concurrent systems.
- Synthesis.
  - Flexible and simple frameworks.
  - Algorithmic analysis.

$$P_1||?_1||\ldots||?_k \vDash \alpha$$

MV: Modelling

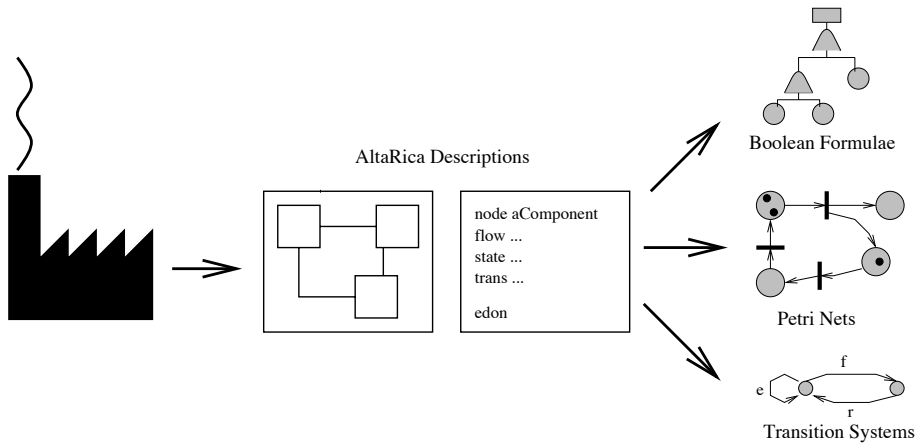
Verification of systems

- Need for formal and flexible description language.
- Often limited to decidable formalisms (finite automata).
- Need for the industrial strength tools.

Verification of programs

- Undecidability $\rightarrow$ abstraction and/or/ semi-algorithms.
- Need for tools for validating and comparing different approaches.

AltaRica (Griffault)



Design principles

- Language adapted to modelling (hierarchical, all types of communication).
- Clear mathematical semantics (finite automata).

Partners

- ONERA, IML.
- Dassault, Thales, Total, Schneider, Renault, IPSN, CEA, CFI.

AltaRica cont.

Applications to safety

- Fault trees.
- Minimal cuts.
- Stochastic Petri Nets.
- Scenarios.

Industrial implementations

- AltaRica DataFlow.
- OCAS (Dassault).
- Combava (ARboost).
- SIMFIA V2 (EADS).

A community

- Common formalism. A large set of tools
- WWW-site: `http://altarica.labri.fr`
- Yearly workshops.

Infinite state systems

Example:

- Programs manipulating integers.

```
int double(int x)
{
  int y=0;
  while (x>=0){
    x=x-1;
    y=y+2;
  }
  return y;
}
```

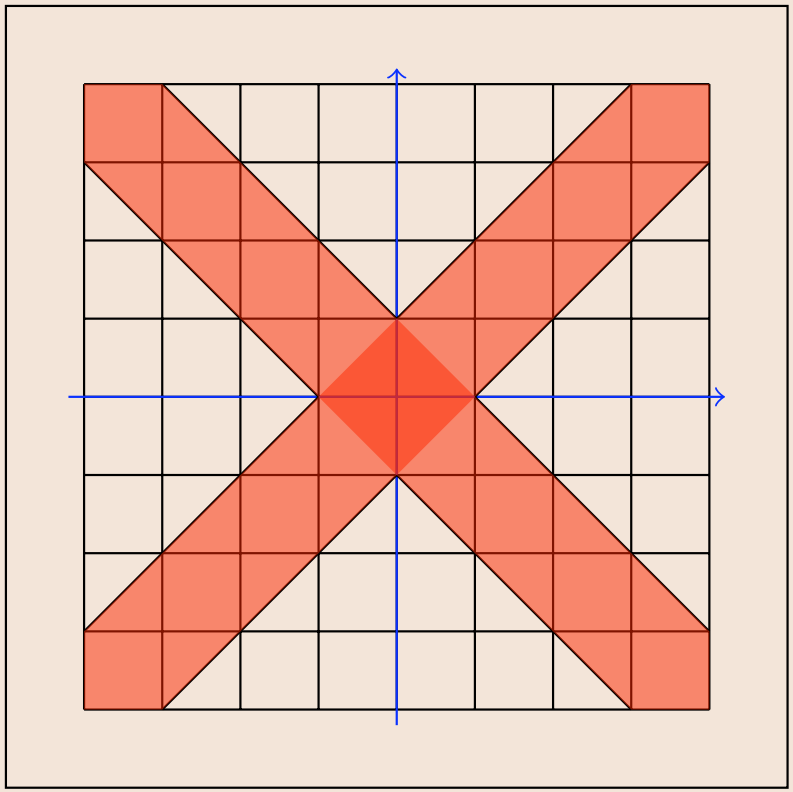
$(y' = 2x \ \wedge \ x \geq 0) \ \vee \ (y' = 0 \ \wedge x \leq 0)$

- Parametrized systems (number of clients).

Tools for arithmetic with addition

Goal: manipulate efficiently sets of vector of integers.

Example: $-1 \leq x + y \leq 1 \ \vee \ -1 \leq x - y \leq 1$



Tools: Sataf (2005), Prestaf (2005), Genepi (2006), Batof (2006).

Integration with AltaRica (http://altarica.labri.fr)

Summary

Research areas

- Computational linguistics.
- Logic Languages and Automata.
- Modelling and Verification.

Some distinctions

- Gödel Prize 2002 :
  G. Sénizergues
- Citation Laureate 2004 :
  B. Courcelle

Cooperation

- India, Vietnam, Belgium, Poland, Germany, Israel, Portugal, Italy, Hungary, Spain, USA.
- European projects.
- National projects.
- IRISA (Rennes), LIAFA (Paris VII), LSV (Cachan), ONERA (Toulouse).