1. Software Validation and Verification

- **Software Testing** [Myers 1979]
  - Widely used (Verification and Qualification)
  - derived from the specification, model-based testing, ...
  - code coverage metrics
  - Still Program testing can be used to show the presence of bugs, but never to show their absence! (E. Dijkstra 1972)

- **Computer-assisted Proof**
  - Property of interest is seen as a Theorem (logically valid) using Predicates calculus
  - Powerful. Complete formal proof of the 4 colors Theorem (1976)
  - Tools: Coq, SMV ...
  - Issue: Decidability, termination, hardly automated and not scalable

- **Model Checking** [Clarke, Emerson, and Sifakis 1981]
  - Derive a formal model from the real program (temporal logic automaton) ...
  - Prove the needed properties on the model
  - Tools: BLAST, SPIN, …
  - Issue: State explosion problem

- **Abstract Interpretation** [Cousot and Cousot 1975]
  - Semantic formalized as a fixed-point of a monotonic operator in a partially ordered structure.
  - Fully automated.
  - Industrial tools exists: Polyspace Verifier (MathWorks), Astrows (ENS), Fluctuate (CEA), ast (AISNSI) ...
  - Issue: find the suitable abstract domain for the properties of interest.

2. Concrete Semantics

- **Second order Filter**

```
begin
En = [-1,1]; i
Sm1 = 0; Sm2 = 0; i = 0;
while (i<1000) do
Sm1 = h(Sm2, Sm1); Sm2 = Sm1; Sm1 = Sm2; Sm2 = i = i+1;
done;
end
```

3. Abstract Interpretation Based Static Analysis

- **Equations System**

```
X₀ = \{ V \rightarrow 1 | \exists j \in \mathbb{N} \cup \{0\}, R \}
X₁ = En \leftarrow [-1, 1], [X₀]
X₂ = [S_{n-1} \leftarrow 0, [X₁]]
X₃ = [S_{n-1} \leftarrow 0, S_{n-2}, [X₂]]
X₄ = [S_{n-2} \leftarrow S_{n-1}, [X₃]]
X₅ = [S_{n-2} \leftarrow S_{n-1}, S_{n-2}, [X₃]]
X₆ = [S_{n-1} \leftarrow 0, [X₃]]
X₇ = En \leftarrow [-1, 1], [X₀]
```

4. Affine Forms Abstract Domain [Graichen and Patrel: 06, 08]

```
\[ x = a₀ + \sum_{i=1}^{n} a_i x_i \]
```

- **Shared** noise symbols express implicit dependencies between variables

5. Contributions

- **ATV (Cristian ST) Case Study** [DASIA 09]
- **Use of optimization techniques (Semidefinite Programming) in abstract transfer functions.**
- **Development of an abstract domain, Tagfor!s (Licence GPL), based on affine forms as a new domain of APROPOS Library ([http://apron.cri.ensmp.fr](http://apron.cri.ensmp.fr))** [CAV 09]

**Ongoing work**: Design of an abstract domain with support of constraints over noise symbols in order to improve the abstraction of the text transfer functions.