Parallel Combination of Abstract Interpretation and Model-Based Automatic Analysis of Software

Patrick COUSOT
École Normale Supérieure
DMI, 45, rue d'Ulm
75230 Paris cedex 05
France
cousot@dmi.ens.fr
http://www.ens.fr/~cousot

Radhia COUSOT
CNRS & École Polytechnique
LIX
91440 Palaiseau cedex
France
rcousot@lix.polytechnique.fr
http://lix.polytechnique.fr/~radhia

AAS'97, Paris, January 14, 1997

Combining Model-Checking and Abstract Interpretation Why?

- Model-checking:
  - Finite state space
  - Sound and complete property verification
- Abstract Interpretation:
  - Infinite state space
  - Sound but uncomplete property determination

Combining Model-Checking and Abstract Interpretation How?

1. *Abstract symbolic methods:*
   - Use symbolic representations of properties (BDDs, convex polyhedra, ...)
   - One can make approximations (e.g. widenings)

     $\Rightarrow$ Approximate properties of an exact model
2. *Model abstraction:*
   - The finite model is an abstraction of the system

     $\Rightarrow$ Exact properties of an approximate model

In this paper ...

3. *Parallel combination of model-checking and abstract interpretation:*
   - Model-checking:
     - Exact symbolic representation of properties
     - The model is an exact representation of the system

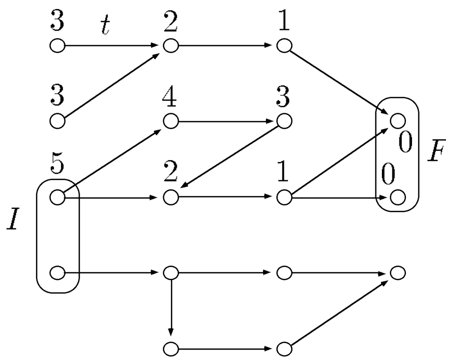
       $\Rightarrow$ Exact properties of exact model
   - Abstract interpretation:
     - Preliminary/parallel analysis of the model by abstract interpretation

       $\Rightarrow$ Limit the state search space

$\Longrightarrow$ Exact properties of an exact sub-model

Example: Maximum Delay Problem [1]

Find the maximum delay to reach a final state starting from some initial state:

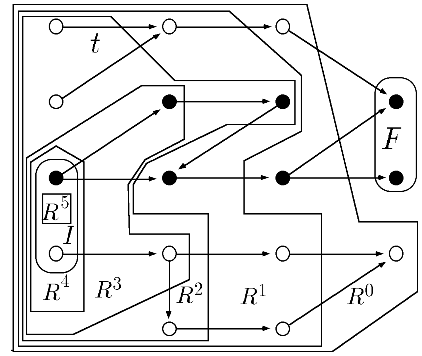


Maximum Delay Algorithm "maximum1" [2]

procedure maximum1 $(I,\ F)$;
$R' := S$;
$n := 0$;
$R := (S - F)$;
while $(R \neq R' \wedge R \cap I \neq \emptyset)$ **do**
  $R' := R$;
  $n := n + 1$;
  $R := \mathsf{pre}[t]\ R' \cap (S - F)$;
od;
return if $(R' = R)$ **then** ∞ **else** n;

[1] Halbwachs, N. Delays analysis in synchronous programs. CAV '93, LNCS 697, 1993, pp. 333–346.
[2] Campos, S., Clarke, E., Marrero, W., and Minea, M. Verus: A tool for quantitative analysis of finite-state real-time systems. Proc. ACM SIGPLAN 1995 Workshop on Languages, Compilers & Tools for Real-Time Systems, La Jolla, Calif., jun 21–22, 1995, pp. 75–83.

Execution trace of the "maximum1" algorithm



It is useless to explore the states which are not:

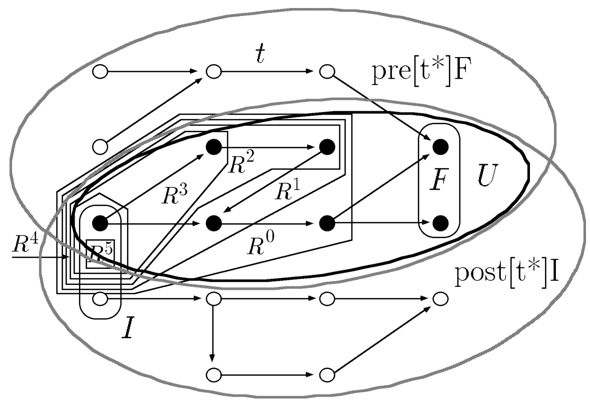
- descendants of the initial states;
- ascendants of the initial states.

Maximum Delay Algorithm "maximum2" (with state search space restriction)

procedure maximum2 $(I,\ F)$;
$R' := S$;
$n := 0$;
$R := \boxed{(U_0 - F)}$;
while $(R \neq R' \wedge R \cap I \neq \emptyset)$ **do**
  $R' := R$;
  $n := n + 1$;
  $R := \mathsf{pre}[t]\ R' \cap \boxed{(U_n - F)}$;
od;
return if $(R' = R)$ **then** ∞ **else** n;

where: $\forall n \geq 0 : U_n \supseteq U \stackrel{\text{def}}{=} \mathsf{post}[t^\star]\ I \cap \mathsf{pre}[t^\star]\ F$

Execution trace of the "maximum2" algorithm



- Any upper-approximations $U_0, U_1, \ldots, U_n, \ldots$ of U can be used;
- In the worst case $U_n = S$ (all states), hence "maximum2" = "maximum1".

Analysis of the model by abstract interpretation

- We can compute:

$$U_0 \supseteq U_1 \supseteq \ldots \supseteq U_n \supseteq U \stackrel{\text{def}}{=} \text{post}[t^\star]\, I \cap \text{pre}[t^\star]\, F$$

  by abstract interpretation;
- The abstract interpretation can be done in parallel with the model-checking (at almost no supplementary cost);
- The abstract interpretation results are used on the fly for U_n as they become available to restrict the state search space;
- Several restriction operators have been proposed for symbolic model checking (with BDDs & convex polyhedra [3]).

[3] Halbwachs, N. and Raymond, P. On the use of approximations in symbolic model checking. Tech. rep. SPECTRE L21 (jan 1996), VERIMAG laboratory, Grenoble, France.

Upper approximation D of $\text{post}[t^\star]\, I = \text{lfp}^{\subseteq}\, \lambda X \cdot I \cup \text{post}[t]\, X$ by abstract interpretation [4]

1. Consider an abstract domain $\langle L, \sqsubseteq \rangle$ approximating sets of states $\langle \wp(S), \subseteq \rangle$;
2. define a correspondence:

$$\langle \wp(S), \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle L, \sqsubseteq \rangle$$

   which is a Galois connection:

$$\forall P \in \wp(S) : \forall Q \in L : \alpha(P) \sqsubseteq Q \Longleftrightarrow P \subseteq \gamma(Q) \; .$$

   The abstract value $\alpha(P)$ is the approximation of $P \subseteq S$: $P \subseteq \gamma(\alpha(P))$.

3. Define an abstract post-image transformer $\mathcal{F} \in L \overset{\text{m}}{\longmapsto} L$:

$$\forall Q \in L : \alpha \circ (\lambda X \cdot I \cup \text{post}[t]\, X) \circ \gamma(Q) \sqsubseteq \mathcal{F}(Q)$$

4. Define a *widening operator* $\nabla \in L \times L \mapsto L$:
   - it is an upper approximation [5],
   - it enforces finite convergence of $\mathcal{F}$-upward iterates [6];
5. The *upward forward iteration sequence with widening*:
   - $\hat{\mathcal{F}}^0 \stackrel{\text{def}}{=} \alpha(\emptyset)$,
   - $\hat{\mathcal{F}}^{i+1} \stackrel{\text{def}}{=} \hat{\mathcal{F}}^i$ if $\mathcal{F}(\hat{\mathcal{F}}^i) \sqsubseteq \hat{\mathcal{F}}^i$
   - $\hat{\mathcal{F}}^{i+1} \stackrel{\text{def}}{=} \hat{\mathcal{F}}^i \nabla \mathcal{F}(\hat{\mathcal{F}}^i)$ otherwise

   is ultimately stationary;

   its limit $\hat{\mathcal{F}}$ is a sound upper approximation of $\text{post}[t^\star]\, I$ in that:

$$\text{post}[t^\star]\, I \subseteq \gamma(\text{lfp}^{\sqsubseteq}\, \mathcal{F}) \subseteq \gamma(\hat{\mathcal{F}}) \; .$$

[4] Cousot, P. and Cousot, R. Abstract interpretation: a unified lattice model for static analysis of programs by construction or approximation of fixpoints. 4th POPL, Los Angeles, 1977, pp. 238–252.

[5] $\forall x, y \in L : x \sqsubseteq x \nabla y$ and $\forall x, y \in L : y \sqsubseteq x \nabla y$.

[6] for all increasing chains $x^0 \sqsubseteq x^1 \sqsubseteq \ldots \sqsubseteq x^i \sqsubseteq \ldots$ the increasing chain defined by $y^0 = x^0, \ldots, y^{i+1} = y^i \nabla x^{i+1}, \ldots$ is not strictly increasing.

6. Define a *narrowing operator* $\triangle \in L \times L \mapsto L$ such that:
   - it is an upper approximation [7],
   - it enforces finite convergence of $\mathcal{F}$-downward iterates [8]
7. the *downward forward iteration sequence with narrowing*:
   - $\check{\mathcal{F}}^0 \stackrel{\text{def}}{=} \hat{\mathcal{F}}$,
   - $\check{\mathcal{F}}^{i+1} \stackrel{\text{def}}{=} \check{\mathcal{F}}^i$ if $\mathcal{F}(\check{\mathcal{F}}^i) = \check{\mathcal{F}}^i$
   - $\check{\mathcal{F}}^{i+1} \stackrel{\text{def}}{=} \check{\mathcal{F}}^i \triangle \mathcal{F}(\check{\mathcal{F}}^i)$ otherwise

   is ultimately stationary;

   its limit $\check{\mathcal{F}}$ is a better sound upper approximation post$[t^\star]\, I$ in that:

$$\text{post}[t^\star]\, I \subseteq \gamma(\text{lfp}^{\sqsubseteq}\, \mathcal{F}) \subseteq \gamma(\check{\mathcal{F}}) \subseteq \gamma(\hat{\mathcal{F}})\ .$$

Abstract interpretation design

- The design of:
  - the abstract algebra $\langle L,\ \sqsubseteq,\ \bot,\ \top,\ \sqcup,\ \sqcap,\ \nabla,\ \triangle,\ f_1,\ \ldots,\ f_n\rangle$
  - the transformer $\mathcal{F}$ (usually composed out of the primitives f_1, $\ldots$, f_n)

  are problem dependent;
- Natural choices in the model-checking context are:
  - BDDs (discrete systems),
  - Convex polyhedra (hybrid systems);

  for which widening operators have been defined [9, 10].

[7] $\forall x, y \in L : x \sqsubseteq y \Longrightarrow x \sqsubseteq x \triangle y \sqsubseteq y$.

[8] For all decreasing chains $x^0 \sqsupseteq x^1 \sqsupseteq \ldots$ the decreasing chain defined by $y^0 = x^0, \ldots, y^{i+1} = y^i \triangle x^{i+1}, \ldots$ is not strictly decreasing.

[9] Mauborgne, L. Abstract interpretation using TDGs. In SAS '94, 20–22 sep 1994, LNCS 864, pp. 363–379.

[10] Cousot, P. and Halbwachs, N. Automatic discovery of linear restraints among variables of a program. In 5th POPL, Tucson, 1978, pp. 84–97.

Upper approximation A of pre$[t^\star]\, F$ = lfp$^{\subseteq}\, \lambda X \cdot F \cup$ pre$[t]\, X$ by abstract interpretation [11]

Use the same abstract algebra $\langle L,\ \sqsubseteq,\ \bot,\ \top,\ \sqcup,\ \sqcap,\ \nabla,\ \triangle,\ f_1,\ \ldots,\ f_n\rangle$:

8. Define an abstract pre-image transformer $\mathcal{F} \in L \stackrel{\text{m}}{\longmapsto} L$:

$$\forall Q \in L : \alpha \circ (\lambda X \cdot F \cup \text{pre}[t]\, X) \circ \gamma(Q) \sqsubseteq \mathcal{B}(Q)$$

9. First use an *upward backward iteration sequence with widening* finitely converging to $\hat{\mathcal{B}}$;
10. Improve by a *downward iteration sequence with narrowing* finitely converging to $\check{\mathcal{B}}$ such that:

$$\text{pre}[t^\star]\, F = \text{lfp}^{\subseteq}\, \lambda X \cdot F \cup \text{pre}[t]\, X \subseteq \gamma(\text{lfp}^{\sqsubseteq}\, \mathcal{B}) \subseteq \gamma(\check{\mathcal{B}}) \subseteq \gamma(\hat{\mathcal{B}})$$

Sequence of upper approximations $U_0,\ U_1,\ \ldots,\ U_n,\ \ldots$ of U = post$[t^\star]\, I \cap$ pre$[t^\star]\, F$ by abstract interpretation [12, 13]

- $U_0 = S$, all states;
- U_1 is the γ-concretization of the limit of the upward forward iteration sequence with widening for $\mathcal{F}$;
- U_2 is the γ-concretization of the limit of the corresponding downward forward iteration sequence with narrowing for $\mathcal{F}$ starting from U_0;
- $\cdots$

[11] Cousot, P. and Cousot, R. Systematic design of program analysis frameworks. In 6th POPL, San Antonio, 1979, pp. 269–282.

[12] Cousot, P. Méthodes itératives de construction et d'approximation de points fixes d'opérateurs monotones sur un treillis, analyse sémantique de programmes. Ph. D. thesis, Université scientifique et médicale de Grenoble, 1978.

[13] Cousot, P. and Cousot, R. Abstract interpretation and application to logic programs. J. Logic Prog. 13, 2–3, 103–179. (The editor of JLP has mistakenly published the unreadable galley proof. For a correct version of this paper, see http://www.ens.fr/~cousot.)

- $\cdots$
- U^{4n+3} is the γ-concretization of the limit of the upward backward iteration sequence with widening for $\lambda X \cdot (U^{4n+2} \sqcap \mathcal{B}(X))$;
- U^{4n+4} is the γ-concretization of the limit of the corresponding downward backward iteration sequence with narrowing for $\lambda X \cdot (U^{4n+2} \sqcap \mathcal{B}(X))$ starting from U^{4n+3};
- U^{4n+5} is the γ-concretization of the limit of the upward forward iteration sequence with widening for $\lambda X \cdot (U^{4n+4} \sqcap \mathcal{F}(X))$;
- U^{4n+6} is the γ-concretization of the limit of the corresponding downward forward iteration sequence with narrowing for $\lambda X \cdot (U^{4n+4} \sqcap \mathcal{F}(X))$ starting from U^{4n+5};
- $\cdots$

Correctness

- The sequence U_0, U_1, U_2, $\ldots$, U^{4n+3}, U^{4n+4}, U^{4n+5}, U^{4n+6}, $\ldots$ is a descending chain;

$\Rightarrow$ The restriction is more and more precise as the model-checking goes on;

- All elements U_k is the sequence are sound:

$$U_k \subseteq \mathsf{post}[t^\star]\, I \cap \mathsf{pre}[t^\star]\, F$$

- Stop the abstract interpretation computation with a narrowing or when the parallel model-checking terminates;

Problematic termination

- The abstract interpretation always terminate;
- The abstract interpretation is approximate so the state-space restriction may not be finite;

$\Rightarrow$ The parallel combination of abstract interpretation and model-checking is incomplete since it may not terminate;

- In case of nontermination the information gathered by abstract interpretation is reusable for verification by:
  - abstract symbolic methods,
  - model abstraction;

  which are also incomplete but guarantee termination.

Conclusion

- We have proposed a method for the parallel combination of model-analysis by abstract interpretation and verification by model-checking where the verification:
  - makes no approximation on states and transitions,
  - explores an (hopefully finite) subgraph;
- Semi-algorithm since there is no guarantee that the explored subgraph will be finite:
  - classical model-checking would have failed anyway,
  - case by case experimentation is needed;
- The method should be used <u>before</u> resorting to model-checking of a more abstract model (the information gathered about the exact model being reusable).