Effective Bit-Width and Under-Approximation

Robert Brummayer and Armin Biere

Institute for Formal Models and Verification
Johannes Kepler University Linz, Austria

Eurocast 2009

Workshop: Applied Formal Verification

Las Palmas, Gran Canaria, Spain

February 20th, 2009

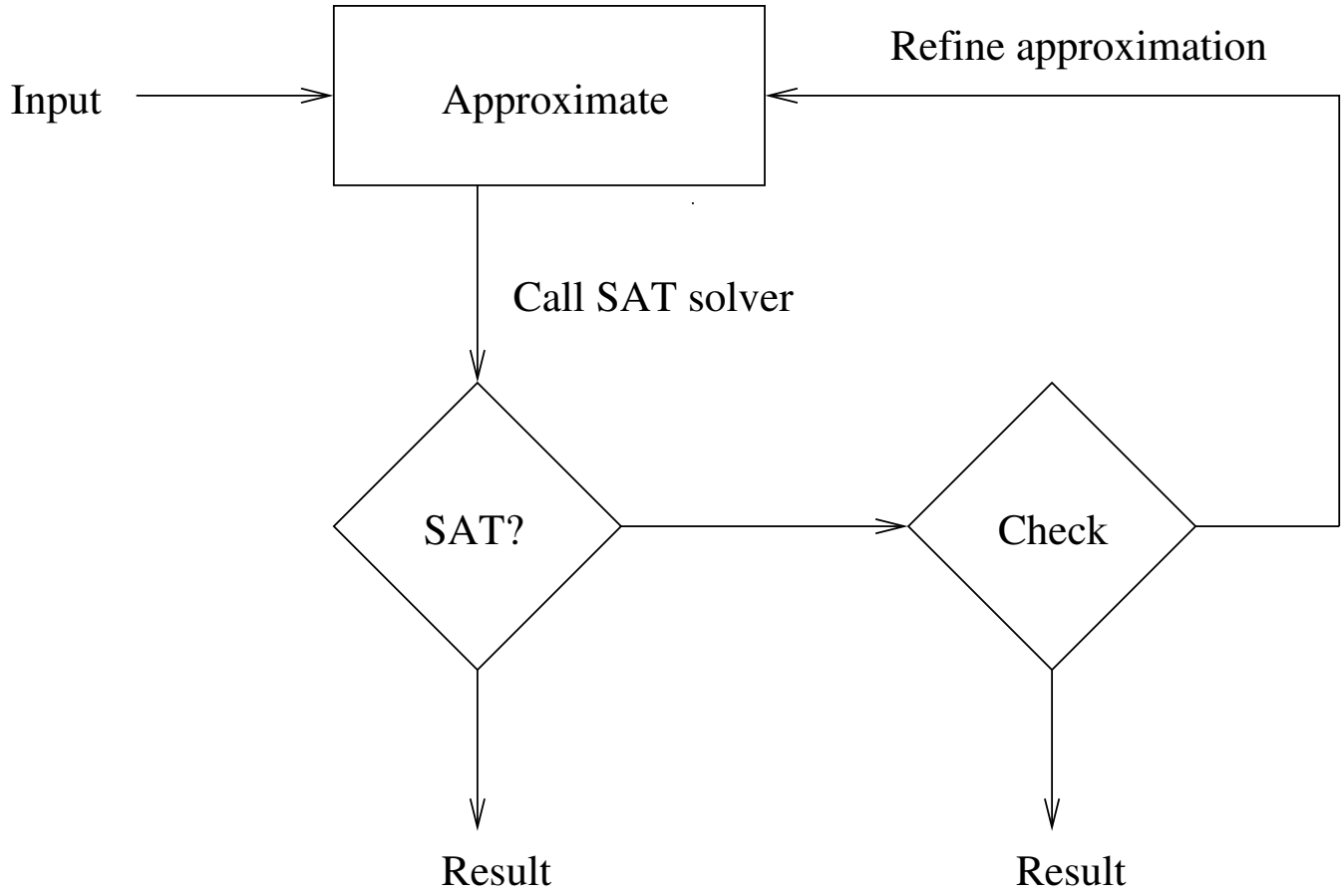
Context: Satisfiability Modulo Theories (SMT)

R. Brummayer, FMV, JKU Linz

- Generalization of the Boolean Satisfiability Problem (SAT)
- Satisfiability with respect to background theories
- Software and Hardware verification
- SMT Solvers
  - Z3, CVC3, STP, Barcelogic, Boolector, MathSAT, Spear, OpenSMT, ...
- Theories
  - Fragments of first-order logic, e.g. QF theory of bit-vectors
    - Recent work on formula approximations on bit-vectors [Bryant07]

Approximation Refinement with SAT Solver

R. Brummayer, FMV, JKU Linz



Over-Approximation vs. Under-Approximation (1/2)

R. Brummayer, FMV, JKU Linz

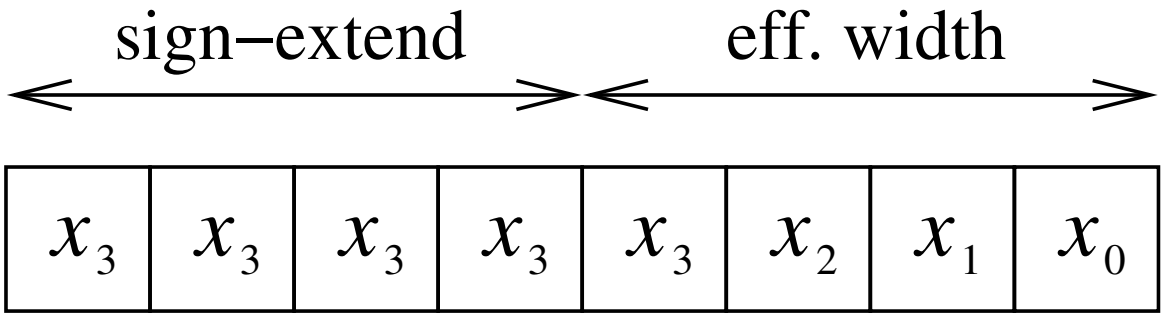
- We consider quantifier-free SMT-formula ϕ
- Over-approximation $\bar{\phi}$
  - For example replace function applications by fresh variables
    - Abstraction does not enforce function congruence
  - Formula approximation $\bar{\phi}$ may have more models than ϕ
  - If $\bar{\phi}$ is unsatisfiable then also ϕ is unsatisfiable
  - If $\bar{\phi}$ is satisfiable then check model σ
    - If σ is spurious then refine abstraction
    - If σ is consistent then ϕ is satisfied by σ and thus satisfiable

Over-Approximation vs. Under-Approximation (2/2)

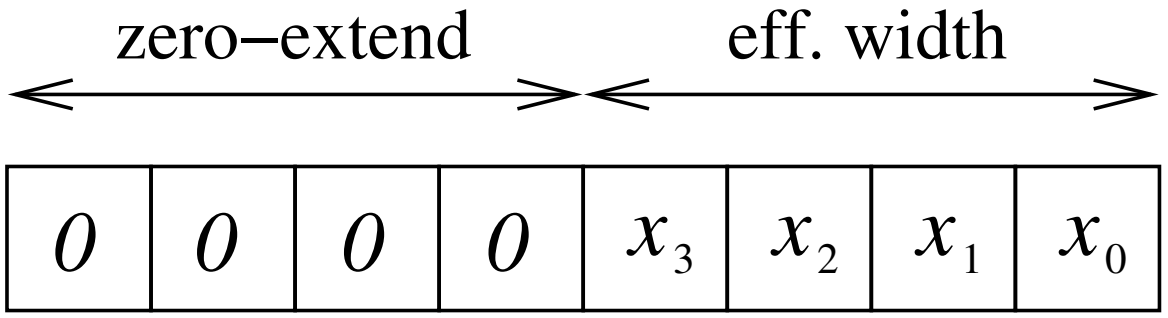
R. Brummayer, FMV, JKU Linz

- Under-approximation $\underline{\phi}$
  - For example restrict domain of variables
    - Additional restrictions may speed up decision procedure
  - Formula approximation $\underline{\phi}$ may have fewer models than ϕ
  - If $\underline{\phi}$ is satisfied by σ then ϕ is also satisfied by σ and thus satisfiable
  - If $\underline{\phi}$ is unsatisfiable then ease restrictions
    - Analyze unsat proof to find out which restrictions to ease

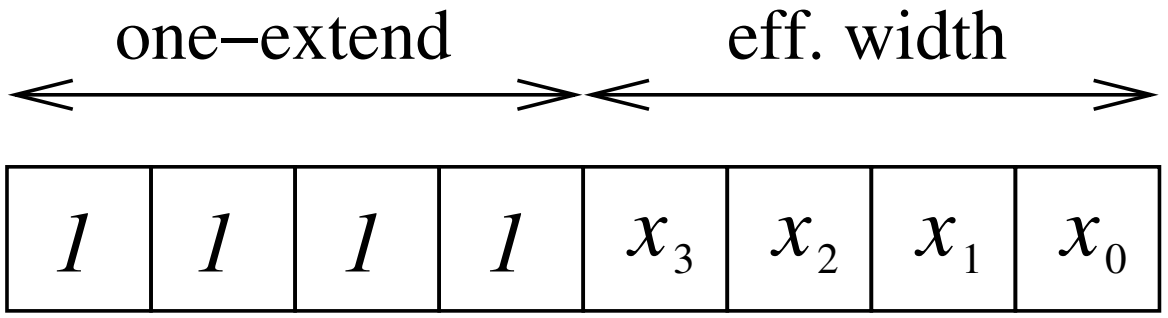
- Sign-Extension of BV variables
  - Let the n least significant bits be variable
  - We call n the effective bit-width
  - Sign-extend the n^{th} bit
  - Appropriate in software verification with two's complement semantics



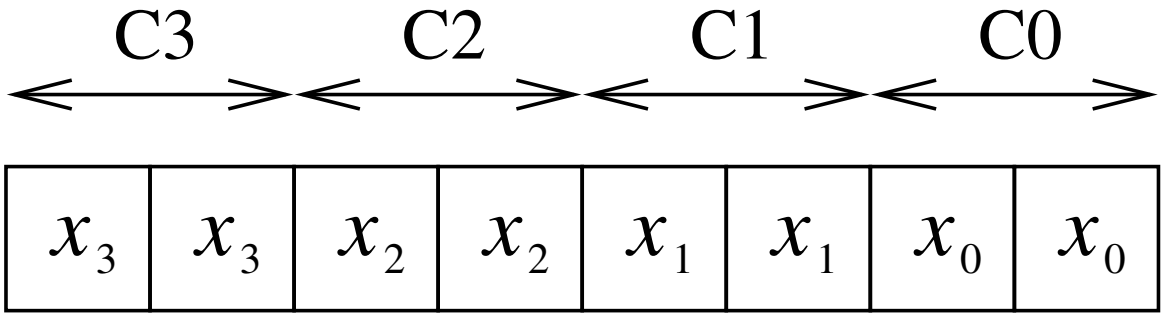
- Zero-Extension (Unsigned-Extension) of BV variables
  - Let the n least significant bits be variable
  - Set remaining bits to zero
  - Appropriate in verification with unsigned semantics
    - May speed up decision procedure more than sign-extension



- One-Extension of BV variables
  - Let the n least significant bits be variable
  - Set remaining bits to one
  - Beneficial in context of negative numbers
    - May speed up decision procedure more than sign-extension



- Equivalence class splitting of BV variables
  - Split bit-vector into n equivalence classes
  - Refinement variants
    - Either refine overall class splitting
    - Or find out which classes to refine



Under-Approximation Encoding on CNF layer

R. Brummayer, FMV, JKU Linz

- Assume x is a bit-vector of bit-width 8 and eff. bit-width 4.
- We perform under-approximation by sign-extension encoding
  - x_0 to x_3 are not constrained by under-approximation
  - We restrict the remaining bits to be equal to x_3
- We introduce fresh boolean under-approximation variable e
- For i = 4 to 7 add clauses: $(x_3 \vee \bar{x_i} \vee \bar{e}) \wedge (\bar{x_3} \vee x_i \vee \bar{e})$
- Finally, let SAT solver assume e

Global vs. Local Refinement

R. Brummayer, FMV, JKU Linz

- Global refinement
  - Manage one under-approximation variable for all bit-vectors globally
  - If $\underline{\phi}$ is unsat then refine all under-approximations
  - Pro: only one fresh boolean variable, Contra: imprecise refinement
- Local refinement
  - Manage under-approximation variable for each bit-vector individually
  - If $\underline{\phi}$ is unsat then refine only under-approximations that are involved
  - Pro: precise refinement, Contra: more fresh variables

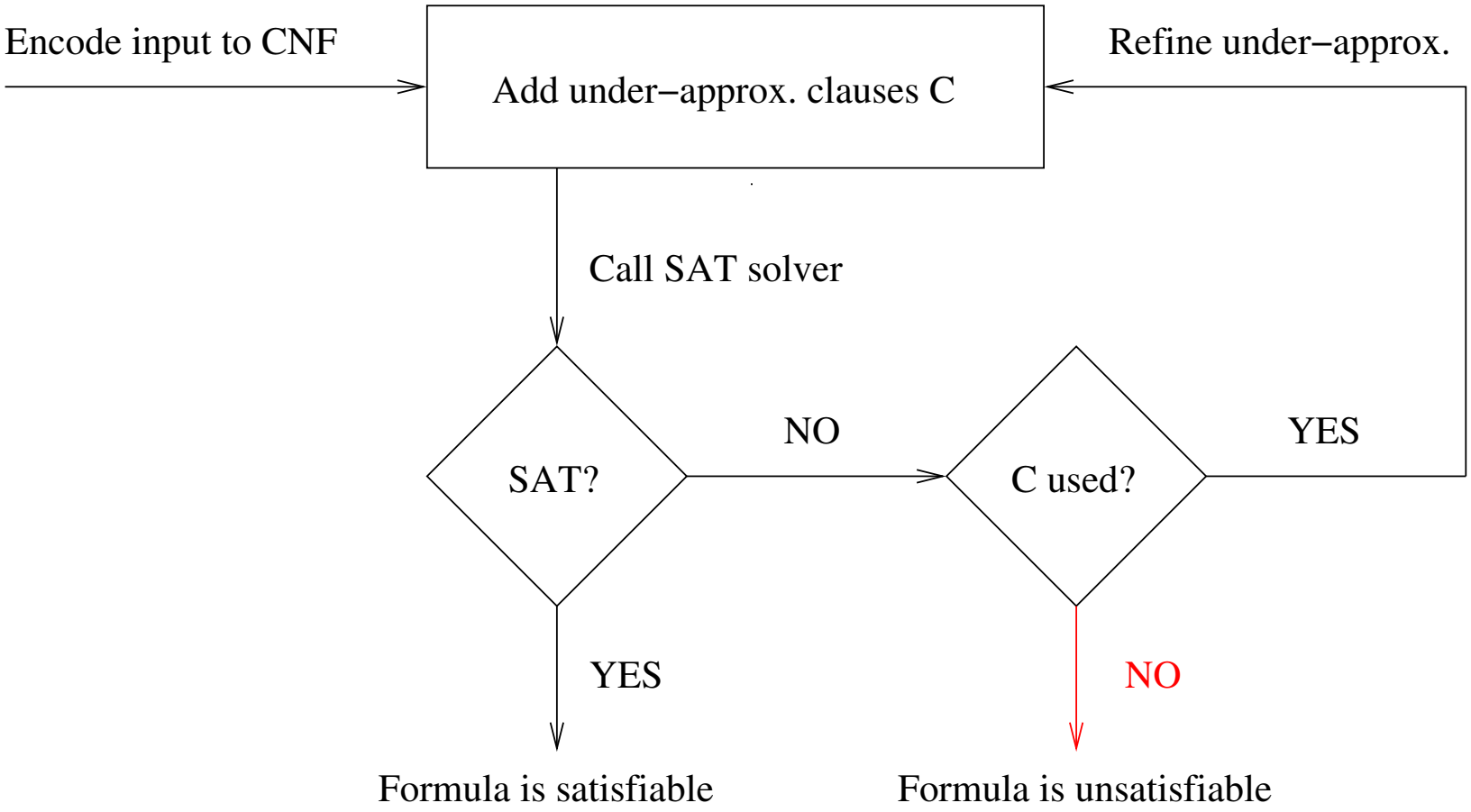
Early Unsat Termination (1/2)

R. Brummayer, FMV, JKU Linz

- Under-approximation $\underline{\phi}$
  - For example restrict domain of variables
    - Additional restrictions may speed up decision procedure
  - Formula approximation $\underline{\phi}$ may have fewer models than ϕ
  - If $\underline{\phi}$ is satisfied by σ then ϕ is also satisfied by σ and thus satisfiable
  - If $\underline{\phi}$ is unsatisfiable then ease restrictions
    - Analyze unsat proof to find out which restrictions to ease
      - Which under-approx. variables have been used by SAT solver?
    - If no under-approx. constraint has been used, terminate with unsat

Early Unsat Termination (2/2)

R. Brummayer, FMV, JKU Linz



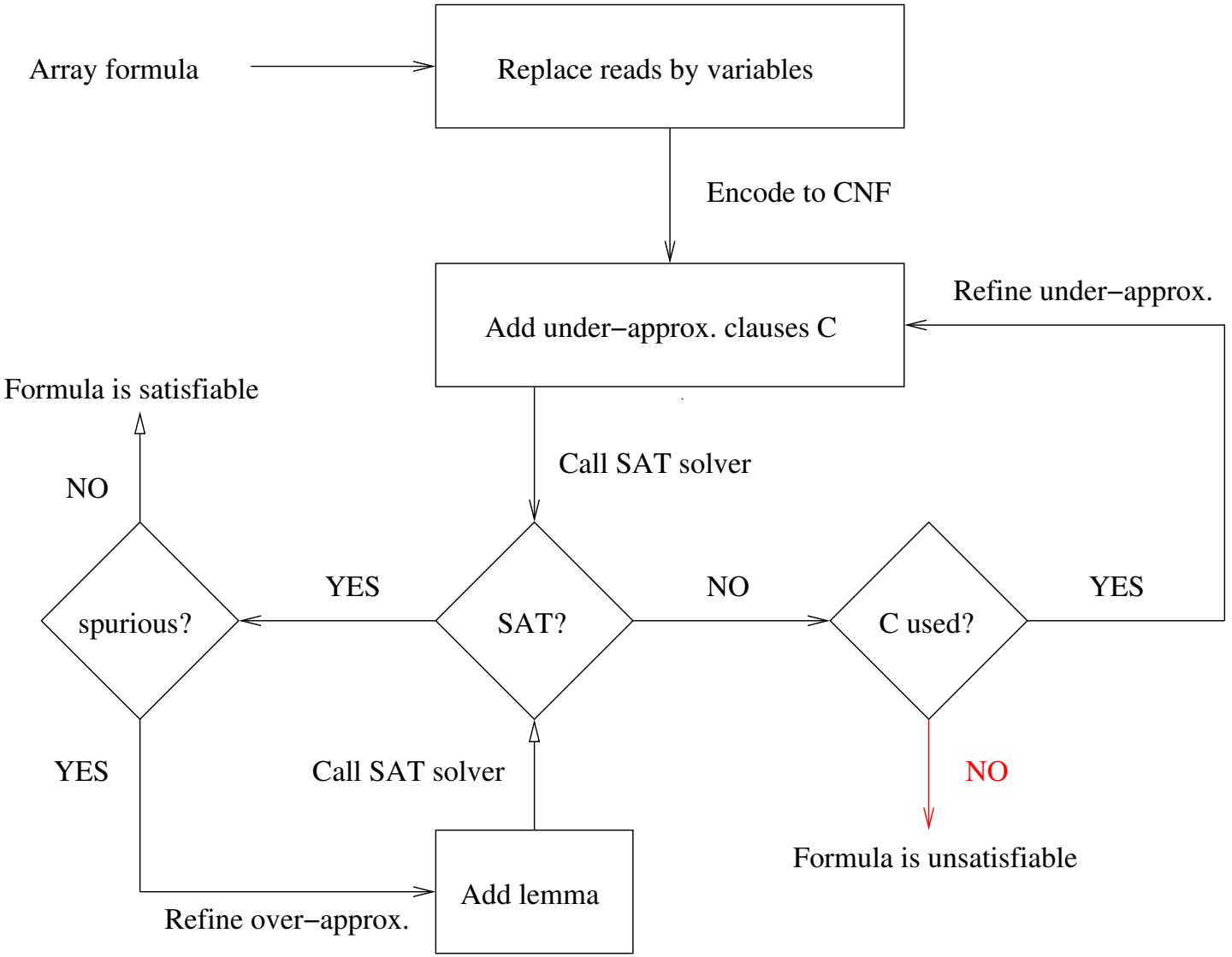
Over-approximating Array Formulas [BrummayerBiere08,Ganesh07]

R. Brummayer, FMV, JKU Linz

- Replace reads in ϕ by fresh abstraction variables to obtain $\bar{\phi}$
- Let SAT solver "guess" solution
  - If SAT solver cannot find a solution, ϕ is unsat
- Explicitly check if model is consistent with theory
- If check succeeds, ϕ is sat
- If check fails
  - Add lemma to refine formula
  - Let SAT solver "guess" a new solution

Combining Over-Approximation and Under-Approximation

R. Brummayer, FMV, JKU Linz



Conclusion

R. Brummayer, FMV, JKU Linz

- Under-approximation
  - Encoding techniques for bit-vectors on CNF layer
  - Refinement strategies: Global vs. Local
  - Early unsat termination
- Over-approximation
  - Handle complex array formulas
- Combination of under-approximation and over-approximation
- Implemented in our SMT solver Boolector