Software Verification

Monday, January 15th 2024, 3 hours

This assignment contains three independent parts: the first part deals with bounded model-checking, the second part is about cartesian abstraction, and the last part addresses abstract interpretation.

All documents are authorized during the examination
Answers can be written in French

1 Bounded Model-Checking (12pts)

This section will browse a few concepts that have been seen in the first part of the course (Bounded Model-Checking).

1.1 Principles

We start with some questions to check your understandings of the Bounded Model-Checking algorithms. We recall that given a transition system $\mathcal{T}$, the Bounded Model-Checking at depth n explores all executions of length at most n and checks if one of them reaches an undesirable state (such an execution is called *faulty*).

Question 1 *A Bounded Model-Checking search can return three answers: Counter-example found, No counter-example found (exhaustive), No counter-example found (non-exhaustive).*

For each possible answer, explain what the answer means about the presence of a faulty execution, its size, and the size of all executions of the model.

Question 2 *We have seen in the course two algorithms: Depth-First Search, and Global. Recall briefly how the executions are explored in the two algorithms and what guarantee they give on the faulty execution they return if they return one.*

1.2 Backward versus Forward

We consider the following program:

```
y = n+1;
x = -n;
while(y >= 0){
  y = y-1;
  assert(x+y == 0);
  x = x+1;
}
```

We suppose that n is a value that is fixed before that code, which can be any integer.

Question 3 *Give the control-flow automaton corresponding to that code. If you have skip edges of the form $q \xrightarrow{\text{skip}} q'$ such that q has no other outgoing edges, merge q and q' (in order to get a smaller automaton).*

You should get an automaton with 8 states and 8 transitions, with no skip edges.

Question 4 *Apply the bounded model-checking algorithm in forward to the previous system up to bound 7: draw the exploration tree displaying the formulæ added (only the added one not the complete formula) at each step in the Depth-First Search algorithm, highlight the unreachable nodes, and give the result of the algorithm.*

Is there a bound for which the answer would be different? Justify your answer.

Question 5 *Apply the bounded model-checking algorithm in backward to the previous system up to bound 7 (in the same style as the previous question).*

Is there a bound for which the answer would be different?

If the results are differents in the two directions, explain why.

Question 6 *Consider a control-flow automaton $\mathcal{A}$. If the forward bounded model-checking returns a counter-example at bound k, is it also the case for the backward Bounded-model checking? Why?*

Question 7 *Can the forward bounded model-checking returns Exhaustive while the backward one returns Non-Exhaustive? If so, give an example of such a system, if no, justify.*

1.3 Finite-valued systems

Question 8 *Consider a control-flow automaton $\mathcal{A}$ that can visit only finitely many configurations, but still has at least one infinite execution. Suppose furthermore that automaton cannot reach q_{bad}.*

Explain why the forward bounded model-checking returns Non-exhaustive on that system.

Question 9 *Propose a variant of forward depth-first search bounded model-checking that would ensure that the answer of that variant can be Exhaustive provided a large enough bound. What is the additional cost of such an approach?*

How does it change the complexity of the algorithm?

Question 10 *If we apply the previous approach to the backward case, what are the conditions over the sets of executions so that in the backward bounded model-checking the answer is Non-exhaustive, but is Exhaustive with the previous approach?*

2 Cartesian Abstraction (8pts)

Let X be the finite set of variables of a program. A predicate p is a formula over X, and a valuation v is a function $v \in \mathbb{Z}^X$. We write $v \models p$ if v is a model of p, and we denote by $[\![p]\!]$ the set of models of p. Let $p_1, \ldots, p_d$ be a sequence of predicates, and let $\mathbb{B} = \{0, 1\}$ be the set of boolean values. We associate with a valuation v the boolean vector $\eta(v) = (b_1, \ldots, b_d)$ in $\mathbb{B}^d$ defined as follows for every $i \in \{1, \ldots, d\}$:

$$b_i = \begin{cases} 1 & \text{if } v \models p_i \\ 0 & \text{otherwise} \end{cases}$$

The boolean abstraction $\alpha_{\text{bool}} : (\mathcal{P}(\mathbb{Z}^X), \subseteq) \to (\mathcal{P}(\mathbb{B}^d), \subseteq)$ and the boolean concretization $\gamma_{\text{bool}} : (\mathcal{P}(\mathbb{B}^d), \subseteq) \to (\mathcal{P}(\mathbb{Z}^X), \subseteq)$ are defined for every set $V \subseteq \mathbb{Z}^X$ and $B \subseteq \mathbb{B}^d$ as follows:

$$\alpha_{\text{bool}}(V) = \{\eta(v) \mid v \in V\}$$
$$\gamma_{\text{bool}}(B) = \bigcup_{b \in B} [\![\phi_b]\!]$$

where ϕ_b is the following predicate where $b = (b_1, \ldots, b_d)$:

$$\phi_b = \bigwedge_{i=1}^{d} \begin{cases} p_i & \text{if } b_i = 1 \\ \neg p_i & \text{if } b_i = 0 \end{cases}$$

Question 11 *Prove that* $(\alpha_{\text{bool}}, \gamma_{\text{bool}})$ *forms a Galois connection.*

For each $i \in \{1, \ldots, d\}$, we introduce the projection function $\Pi_i : \mathcal{P}(\mathbb{B}^d) \to \mathcal{P}(\{0, 1\})$ defined by $\Pi_i(B) = \{b_i \mid (b_1, \ldots, b_d) \in B\}$ for every $B \subseteq \mathbb{B}^d$.

The cartesian abstraction $\alpha_{\text{cart}} : (\mathcal{P}(\mathbb{B}^d), \subseteq) \to (\mathcal{P}(\mathbb{B}^d), \subseteq)$ is defined as follows for every set $B \subseteq \mathbb{B}^d$:

$$\alpha_{\text{cart}}(B) = \Pi_1(B) \times \cdots \times \Pi_d(B)$$

Question 12 *Provide the set* $\alpha_{\text{cart}}(\{(0, 0), (1, 1)\})$.

Question 13 *Provide the concretization function* $\gamma_{\text{cart}} : (\mathcal{P}(\mathbb{B}^d), \subseteq) \to (\mathcal{P}(\mathbb{B}^d), \subseteq)$ *such that* $(\alpha_{\text{cart}}, \gamma_{\text{cart}})$ *forms a Galois connection.*

We introduce the functions $\alpha_{\text{bc}} : (\mathcal{P}(\mathbb{Z}^X), \subseteq) \to (\mathcal{P}(\mathbb{B}^d), \subseteq)$ and $\gamma_{\text{bc}} : (\mathcal{P}(\mathbb{B}^d), \subseteq) \to (\mathcal{P}(\mathbb{Z}^X), \subseteq)$ defined by $\alpha_{\text{bc}} = \alpha_{\text{cart}} \circ \alpha_{\text{bool}}$ and $\gamma_{\text{bc}} = \gamma_{\text{bool}} \circ \gamma_{\text{cart}}$.

Question 14 *Prove that* $(\alpha_{\text{cart}}, \gamma_{\text{cart}})$ *forms a Galois connection.*

Let *op* be a program operation over X. We introduce the set $X' = \{x' \mid x \in X\}$ as a disjoint copy of X. We denote by ψ_{op} a formula over $X \cup X'$ such that $\rho \in \mathbb{Z}^{X \cup X'}$ is a model of ψ_{op} if, and only if, $v \xrightarrow{op} v'$ where $v, v' \in \mathbb{Z}^X$ are the valuation defined by $v(x) = \rho(x)$ and $v'(x) = \rho(x')$ for every $x \in X$.

Question 15 *Let $b, b' \in \mathbb{B}^d$. Provide an easily computable (in polynomial time) formula that is satisfiable if, and only if, $b' \in \alpha_{\text{bc}} \circ \text{post}_{op} \circ \gamma_{\text{bc}}(\{b\})$.*

The boolean under-approximation $F(V)$ of a set $V \subseteq \mathbb{Z}^X$ is the set $F(V) = \{b \in \mathbb{B}^d \mid \gamma_{\text{bool}}(\{b\}) \subseteq V\}$.

Question 16 *Show that $F(V)$ is the greatest set B of $(\mathcal{P}(\mathbb{B}^d), \subseteq)$ satisfying $\gamma_{\text{bool}}(B) \subseteq V$.*

Question 17 *Let $b \in \mathbb{B}^d$, and p be a predicate. Provide an easily computable (in polynomial time) formula that is satisfiable if, and only if, $b \in F([\![p]\!])$.*

3 Abstract Interpretation (10pts)

This section deals with abstract interpretation, and is divided into two independent subsections. The first subsection applies range analysis on an example. The second subsection studies and applies multi-valued constant propagation analysis, an extension of constant propagation analysis to finite sets of values.

3.1 Range analysis

We perform range analysis — this analysis was presented in the course — on the control-flow automaton depicted in Figure 1. This control-flow automaton has two variables `x` and `y`, both ranging over integers. The initial location is q_{in} and the bad location is q_{bad}. Recall that range analysis uses the *abstract domain of intervals*. Like in the course, an analysis will

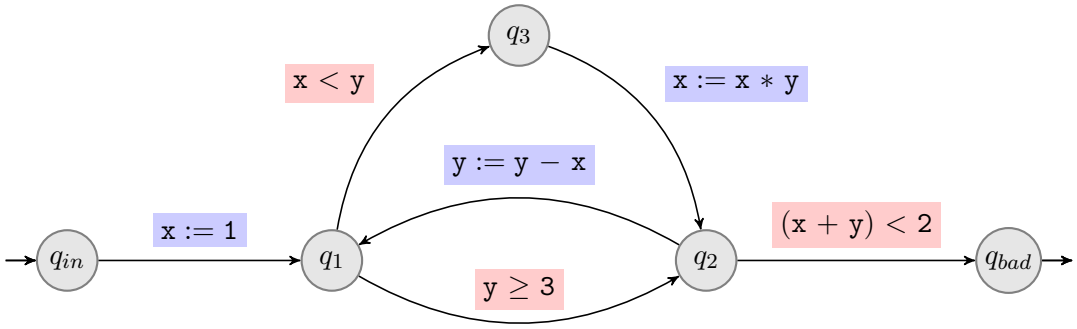

Figure 1: Control-flow automaton of Questions 18 and 19.

be called *successful* when the abstract value obtained for q_{bad} has an empty concretization. Round-robin iteration shall use the following order on locations: $q_{in}, q_1, q_2, q_3, q_{bad}$.

Question 18 *Apply the round-robin algorithm with widening. Do* not *use narrowing. Is the analysis successful?*

Question 19 *Starting from the result of the previous question, perform a decreasing iteration with narrowing. Is the analysis successful?*

3.2 Multi-valued constant propagation analysis

3.2.1 Design of the analysis

Let us introduce the abstract domain $(A, \preceq)$ that will be used for the remainder of this section. We write $\mathcal{P}(\mathbb{Z})$ the set of all subsets of $\mathbb{Z}$, and we let $\mathcal{P}_f(\mathbb{Z})$ denote the set of all *finite* subsets of $\mathbb{Z}$. The set A of abstract elements is defined by

$$A \quad = \quad \mathcal{P}_f(\mathbb{Z}) \ \cup \ \{\star\}$$

where $\star$ is a particular element, with $\star \notin \mathcal{P}_f(\mathbb{Z})$, that will be used to represent infinite subsets of $\mathbb{Z}$. We define the binary relation $\preceq$ on A as follows:

$$\preceq \quad = \quad (A \times \{\star\}) \ \cup \ \{(a, b) \in \mathcal{P}_f(\mathbb{Z}) \times \mathcal{P}_f(\mathbb{Z}) \mid a \subseteq b\}$$

It is readily seen that $\preceq$ is a partial order on A.

Question 20 *Prove that the partially-ordered set $(A, \preceq)$ is a complete lattice with greatest lower bound $\wedge$ and least upper bound $\vee$ verifying*

$$\wedge X \ = \ \begin{cases} \star & \text{if } X \subseteq \{\star\} \\ \cap(X \setminus \{\star\}) & \text{otherwise} \end{cases} \qquad \vee X \ = \ \begin{cases} \cup X & \text{if } X \subseteq \mathcal{P}_f(\mathbb{Z}) \text{ and } \cup X \text{ is finite} \\ \star & \text{otherwise} \end{cases}$$

for every subset $X \subseteq A$.

Question 21 *Provide the least element of A and the greatest element of A.*

We consider the abstraction function $\alpha : \mathcal{P}(\mathbb{Z}) \to A$ and the concretization function $\gamma : A \to \mathcal{P}(\mathbb{Z})$ defined as follows:

$$\alpha(c) \ = \ \begin{cases} c & \text{if } c \text{ is finite} \\ \star & \text{otherwise} \end{cases} \qquad \gamma(a) \ = \ \begin{cases} a & \text{if } a \neq \star \\ \mathbb{Z} & \text{otherwise} \end{cases}$$

Question 22 *Prove that $(\mathcal{P}(\mathbb{Z}), \subseteq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (A, \preceq)$ is a Galois connection.*

Recall that a partially-ordered set satisfies the *ascending* (resp. *descending*) *chain condition* if it does not contain any strictly increasing (resp. decreasing) infinite sequence of elements.

Question 23 *Does $(A, \preceq)$ satisfy the ascending chain condition? Does $(A, \preceq)$ satisfy the descending chain condition? Justify your answers.*

We introduce the functions $\nabla : (A \times A) \to A$ and $\Delta : (A \times A) \to A$ defined by:

$$a \nabla b \ = \ \begin{cases} a \cup b & \text{if } a \in \mathcal{P}_f(\mathbb{Z}) \text{ and } b \in \mathcal{P}_f(\mathbb{Z}) \text{ and } |a \cup b| \leq 3 \\ \star & \text{otherwise} \end{cases} \qquad a \Delta b \ = \ b$$

In the above definition, $|X|$ denotes the cardinal of a set X.

Question 24 *Prove that the function ∇ is a widening operator for $(A, \preceq)$.*

Question 25 *Prove that the function Δ is a narrowing operator for $(A, \preceq)$.*

By *multi-valued constant propagation analysis*, we mean the abstract interpretation based on the Galois connection $(\mathcal{P}(\mathbb{Z}), \subseteq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (A, \preceq)$ defined above, and using the widening and narrowing operators ∇ and Δ defined above.

3.2.2 Application of the analysis

To conclude this subsection, we apply multi-valued constant propagation analysis to the control-flow automaton depicted in Figure 2. This control-flow automaton has a single variable `x`, which ranges over integers. The initial location is q_{in} and the bad location is q_{bad}. Like in the course, an analysis will be called *successful* when the abstract value

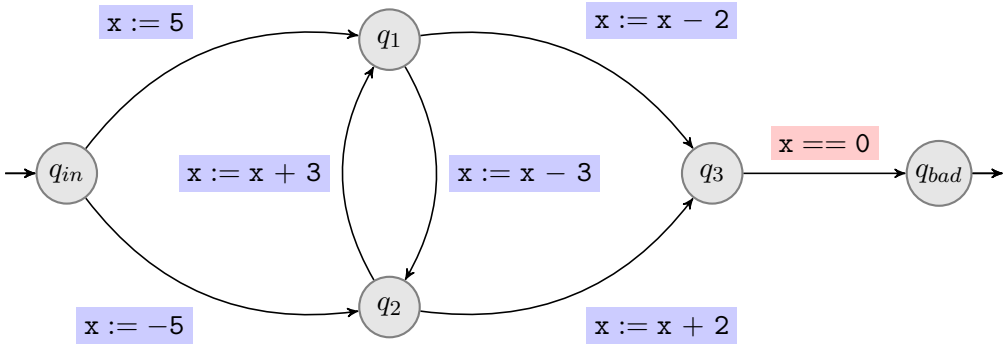

Figure 2: Control-flow automaton of Questions 26 and 27.

obtained for q_{bad} has an empty concretization. Round-robin iteration shall use the following order on locations: $q_{in}, q_1, q_2, q_3, q_{bad}$.

Warning. In the questions below, we use the Galois connection $(\mathcal{P}(\mathbb{Z}), \subseteq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (A, \preceq)$ and the widening and narrowing operators ∇ and Δ defined in § 3.2.1.

Question 26 *Apply the round-robin algorithm with widening. Do* not *use narrowing. Is the analysis successful?*

Question 27 *Starting from the result of the previous question, perform a decreasing iteration with narrowing. Is the analysis successful?*