Programming Languages Theory Notes

Jae Tak Kim

Fall 2020

Contents

1 Introduction

Source: Isil Dillig – A Gentle Introduction to Program Analysis

1.1 Overview

1.1.1 What is a typical static analysis question?

Given source code of program P and desired property Q, does P exhibit Q in *all possible executions*?

1.1.2 Soundness vs Completeness?

In formal logic, an expression is sound if whenever the premise is true, the conclusion is also aways true. There's no guarantees about the conclusion if the premise is false. In the case of static analysis, the premise is that the program is unsafe and the conclusion is that our static analysis will say the program is unsafe. Thus, if the program is really unsafe, then the analysis will *always* tell us that the program is unsafe. However, it's possible that the analysis will take us it's unsafe even when the program is safe.

Our analysis is complete if our program analysis always tells us the truth. Thus, completeness is a stronger condition than soundness.

1.1.3 Describe the inherent limitations of static analysis.

The question of trying to see if the program has a property in every single execution is undecidable. A short proof of is uses the Rice Theorem which states that all nontrivial properties are undecidable. We can even reduce this to the halting problem.

Thus, our static analysis will be either unsound(we definitely don't want this since we don't want our analysis to say that our program is safe when it isn't), sound but incomplete($\exists$ false positives), or non-terminating(we don't want our analysis to run forever!). Clearly, the best option is the second one, so most analysis techniques will be sound but incomplete.

1.1.4 How do you design sound static analyses?

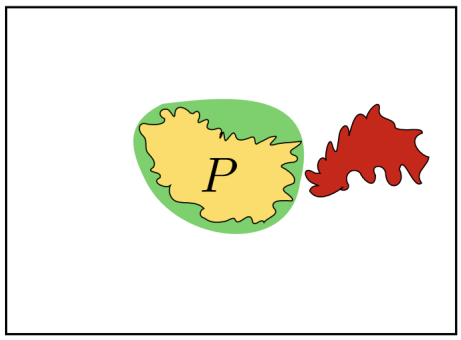
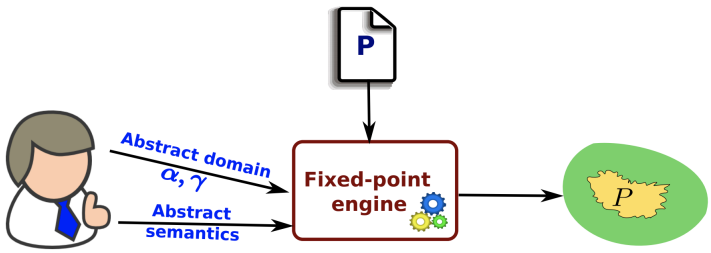


Figure 1: All possible states

We overapproximate our program behavior as little as we can. Overapproximating ensures that we never give an invalid answer. If P region is the actual behavior of our program, and the green bubble is the analysis, then any states outside of the green bubble will be correctly classified. States within the green bubble but outside of the region P will be false alarms (false positives). We overapproximate using abstractions so the goal of static analysis is to construct abstractions that are precise enough (few false alarms) and scale to real programs.

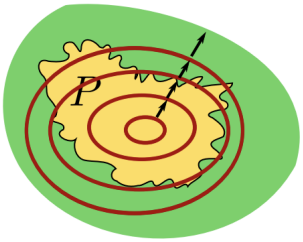
1.1.5 Abstraction interpretation

Framework for constructing sound-by-construction static analyses. Includes abstract domain, abstract function, concretization function, and abstract transformers/semantics.



1.1.6 What is the idea of least-fixed point in a fixed-point computation?

You start with an underapproximation and grow the approximation until it stops growing. This is the point where your analysis is as precise as it can be while still being sound.



2 Lattice Theory

2.0.1 Define Lattice and Fixed-points

2.0.2 Define a monotone function on a lattice + intuition

A function $f : L \to L$ is **monotone** if $\forall x, y \in L,\ x \sqsubseteq y \Rightarrow f(x) \sqsubseteq f(y)$. A monotonic function preserves partial ordering. Also can be thought of as "increasing the input also

increases output." Note that this is not the same thing as extensive. A function f is **extensive** if $\forall x \in L, x \sqsubseteq f(x)$.

We can think of a monotone function as a function where more precise input cannot lead to less precise output.

2.1 Equations and Fixed-Points

2.1.1 What is the fixed-point theorem and the general idea of the proof?

Theorem. *In a lattice L with finite height, every monotone function f has a unique least fixed-point given by*

$$fix(f) = \bigsqcup_{i \geq 0} f^i(\bot)$$

Proof Idea Because $\bot$ is the unique smallest element, we can get a chain of inclusions. Since the lattice is of finite height, it must stop somewhere, and this shows that for some i, $f^i(\bot) \sqsubseteq f^{i+1}(\bot)$, making this a fixed-point. We can make an argument with a least fixed-point x that because $\bot \sqsubseteq x$, $f^i(\bot) \sqsubseteq x$ as well, showing that this is indeed the least fixed-point. It is unique by the fact that the partial order relation is anti-symmetric. $\square$