Optimal Cost is Monotonic over Preconditions

Dennis Sprokholt
March 2022

1 Introduction

Often, we can leverage preconditions in program optimization. After all, if a program need only be correct for a smaller set of inputs, then a larger set of programs meets that requirement. Given such a set of programs, we are interested in the optimal program; For instance, the program with minimal execution time.

I realized that the cost of an optimal program for some specification is monotonic over the considered precondition (i.e., set of inputs). This has some interesting implications for superoptimizers, as weakening the precondition does not forfeit optimality; Though, it may violate correctness, but that is often easier to determine.

For simplicity, we consider the case where execution cost is independent from program input. This independence holds for linear instruction sequences. (The theorem should also hold when cost is dependent on the input – under some additional assumptions – which increases proof complexity)

2 Definitions

• \( S \) – The program specification.
• \( \text{exec}(p, x) \) – exec executes program (or specification) \( p \) on input \( x \), producing an output.
• \( \text{cost}(p) \) – The cost of executing program \( p \) on any input.
• \( \equiv_X \) – Extensonal equivalence over domain \( X \) defined as (forall \( p \) and \( q \)):
  \[
  p \equiv_X q \triangleq \forall x \in X. [\text{exec}(p, x) \equiv \text{exec}(q, x)]
  \]
• \( \text{optcost}(S, X) \) – The cost of some program \( p \), provided that both:
  - \( p \) satisfies \( S \) (over \( X \)): \( p \equiv_X S \)
  - \( p \) has minimal cost (over \( X \)): \( \neg \exists q. [\text{cost}(q) < \text{cost}(p) \land q \equiv_X S] \)

The domain \( X \) over which \( p \) is correct (w.r.t. \( S \)) represents the precondition. We demonstrate:

\textbf{Theorem 1} (Optimal Cost is Monotonic over Preconditions).

\( X \subseteq Y \implies \text{optcost}(S, X) \leq \text{optcost}(S, Y) \) (for any \( X, Y, S \))

If we strengthen the precondition, the optimal cost may only decrease. Conversely, if we weaken the precondition, the optimal cost may only increase.
3 Proof

We prove Theorem 1. For some specification $S$ and preconditions $X$ and $Y$,

(A) given that $X \subseteq Y$,
we show $\text{optcost}(S, X) \leq \text{optcost}(S, Y)$.

First, we define a lemma.

**Lemma 1.** If $p \equiv_Y q$ and $X \subseteq Y$ then $p \equiv_X q$. (for any $X, Y, p, q$)

This follows trivially from the definition of $\equiv_X$ (and $\subseteq$).

Following the definition of $\text{optcost}$, we know:

- There exists a program $p$ where $\text{cost}(p) = \text{optcost}(S, X)$, which satisfies:
  
  (B) $p$ satisfies $S$ (over $X$): $p \equiv_X S$
  (C) $p$ has minimal cost (over $X$): $\neg \exists z. [\text{cost}(z) < \text{cost}(p) \land z \equiv_X S]$

- There exists a program $q$ where $\text{cost}(q) = \text{optcost}(S, Y)$, which satisfies:
  
  (D) $q$ satisfies $S$ (over $Y$): $q \equiv_Y S$
  (E) $q$ has minimal cost (over $Y$): $\neg \exists z. [\text{cost}(z) < \text{cost}(q) \land z \equiv_Y S]$

We proceed with a proof by contradiction.

Assume $\text{cost}(p) > \text{cost}(q)$.

$q \equiv_Y S$ by (D), then by Lemma 1 with (A), we know $q \equiv_X S$. Thus we have:

$$\text{cost}(q) < \text{cost}(p) \land q \equiv_X S$$

However, by (C), $q$ cannot exist – as $p$ has minimal cost (over $X$). Hence, we reached a contradiction.

Thus $\text{cost}(p) \leq \text{cost}(q)$. Following their definitions, $\text{optcost}(S, X) \leq \text{optcost}(S, Y)$.

\qed