Aitiana L Mondragon

CS 4390

Dr. Moore

September 26, 2025

**Data Flow Analyses using the Worklist Algorithm**

1. Fill in the table below with the necessary information for each type of analysis:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Domain | Direction | Init | Merge | Transfer |
| Reaching Definitions | Set of defs | Forward | INIT(B) = θ for all blocks B | Union | f(outb ) = genb (inb – killb ) |
| Live Variables | Set of vars | Backward | INIT(B) = θ for all blocks | Union | f(outb ) = useb (outb – killb ) |
| Constant Propagation | Valuation of T | Forward | INIT(ENTRY) = T for all variables | Intersection | x = c 🡪 propagate c; otherwise keep current value. |
| Available Expressions | Set of expression | Forward | INIT(ENTRY) = θ, INIT(B) = U | Intersection | f(outb ) = e\_genb (inb – e\_killb ) |

1. Write a convincing argument that the worklist algorithm is guaranteed to converge to a solution, given a certain condition. Be sure to state that condition.

The worklist algorithm is guaranteed to converge to a solution based on two key ideas: finite height of the lattice (partial order with a unique lower bound) and monotonicity.

1. The lattice has a finite height: The domain of values has a finite number of distinct values and a finite number of steps from highest to lowest. As such, a value can only change a limited number of times before it reaches a stable and final state.
2. All transfer functions are monotonic: The transfer functions are monotonic. This correlates to the idea that they must preserve to partial order of the lattice. Particularly, a transfer function will never go to a less specific state.

Now, as the algorithm proceeds, nodes are repeatedly processes from the worklist. A node is added to the worklist only when the dataflow value for incoming edges change. Due to monotonicity of the transfer functions, the value for any given node can change in only one direction. Furthermore, for finite height of the lattice, this correlates to the idea that the changes must eventually stop. There’s a finite number of possible value changes for any node. Once no more values can change, the worklist ultimately becomes empty and the algorithm terminates. Therefore guaranteeing a point of termination within the algorithm with a guaranteed solution.