

# CS738: Advanced Compiler Optimizations

## Foundations of Data Flow Analysis

Amey Karkare

karkare@cse.iitk.ac.in

<http://www.cse.iitk.ac.in/~karkare/cs738>

Department of CSE, IIT Kanpur



## Agenda

- ▶ *Intraprocedural* Data Flow Analysis
  - ▶ We looked at 4 classic examples
  - ▶ Today: Mathematical foundations

## Taxonomy of Dataflow Problems

- ▶ Categorized along several dimensions
  - ▶ the information they are designed to provide
  - ▶ the direction of flow
  - ▶ confluence operator
- ▶ Four kinds of dataflow problems, distinguished by
  - ▶ the operator used for confluence or divergence
  - ▶ data flows backward or forward

## Taxonomy of Dataflow Problems

<b>Confluence</b> →	U	∩
<b>Direction</b> ↓		
<b>Forward</b>	R D	Av E
<b>Backward</b>	L V	V B E

## Why Data Flow Analysis Works?

- ▶ Suitable initial values and boundary conditions
- ▶ Suitable domain of values
  - ▶ Bounded, Finite
- ▶ Suitable meet operator
- ▶ Suitable flow functions
  - ▶ monotonic, closed under composition
- ▶ But what is **SUITABLE** ?

## Lattice Theory

## Partially Ordered Sets

- ▶ Posets
  - $S$ : a set
  - $\leq$ : a relation
  - $(S, \leq)$  is a **poset** if for  $x, y, z \in S$ 
    - ▶  $x \leq x$  (reflexive)
    - ▶  $x \leq y$  and  $y \leq x \Rightarrow x = y$  (antisymmetric)
    - ▶  $x \leq y$  and  $y \leq z \Rightarrow x \leq z$  (transitive)

## Chain

- ▶ Linear Ordering
- ▶ Poset where every pair of elements is comparable
- ▶  $x_1 \leq x_2 \leq \dots \leq x_k$  is a chain of length  $k$
- ▶ We are interested in chains of finite length

## Observation

- ▶ Any **finite nonempty subset** of a poset has **minimal** and **maximal** elements
- ▶ Any **finite nonempty chain** has **unique** minimum and maximum elements

## Semilattice

- ▶ Set  $S$  and meet  $\wedge$
- ▶  $x, y, z \in S$ 
  - ▶  $x \wedge x = x$  (idempotent)
  - ▶  $x \wedge y = y \wedge x$  (commutative)
  - ▶  $x \wedge (y \wedge z) = (x \wedge y) \wedge z$  (associative)
- ▶ Partial order for semilattice
  - ▶  $x \leq y$  if and only if  $x \wedge y = x$
  - ▶ Reflexive, antisymmetric, transitive

## Border Elements

- ▶ Top Element ( $\top$ )
  - ▶  $\forall x \in S, x \wedge \top = \top \wedge x = x$
- ▶ (Optional) Bottom Element ( $\perp$ )
  - ▶  $\forall x \in S, x \wedge \perp = \perp \wedge x = \perp$

## Familiar (Semi)Lattices

- ▶ Powerset for a set  $S$ ,  $2^S$
- ▶ Meet  $\wedge$  is  $\cap$
- ▶ Partial Order is  $\subseteq$
- ▶ Top element is  $S$
- ▶ Bottom element is  $\emptyset$

## Familiar (Semi)Lattices

- ▶ Powerset for a set  $S$ ,  $2^S$
- ▶ Meet  $\wedge$  is  $\cap$
- ▶ Partial Order is  $\supseteq$
- ▶ Top element is  $\emptyset$
- ▶ Bottom element is  $S$

## Greatest Lower Bound (glb)

- ▶  $x, y, z \in S$
- ▶ glb of  $x$  and  $y$  is an element  $g$  such that
  - ▶  $g \leq x$
  - ▶  $g \leq y$
  - ▶ if  $z \leq x$  and  $z \leq y$  then  $z \leq g$

## QQ

- ▶  $x, y \in S$
- ▶  $(S, \wedge)$  is a semilattice
- ▶ Prove that  $x \wedge y$  is glb of  $x$  and  $y$ .

## Semi(?) -Lattice

- ▶ We can define symmetric concepts
  - ▶  $\geq$  order
  - ▶ Join operation ( $\vee$ )
  - ▶ Least upper bound (lub)

## Lattice

- ▶  $(S, \wedge, \vee)$  is a lattice  
iff for each **non-empty finite** subset  $Y$  of  $S$   
both  $\bigwedge Y$  and  $\bigvee Y$  are in  $S$ .
- ▶  $(S, \wedge, \vee)$  is a complete lattice  
iff for each subset  $Y$  of  $S$   
both  $\bigwedge Y$  and  $\bigvee Y$  are in  $S$ .

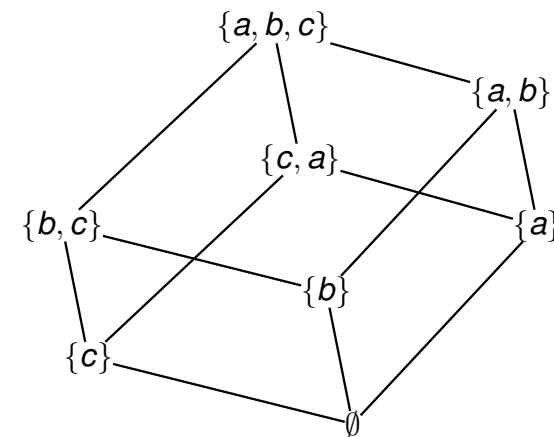
## Lattice

- ▶ Complete lattice  $(S, \wedge, \vee)$ 
  - ▶ For every pair of elements  $x$  and  $y$ , both  $x \wedge y$  and  $x \vee y$  should be in  $S$
  - ▶ Example : Powerset lattice
- ▶ We will talk about **meet** semi-lattices only
  - ▶ except for some proofs

## Lattice Diagram

- ▶ Graphical view of posets
- ▶ Elements = the nodes in the graph
- ▶ If  $x < y$  then  $x$  is depicted lower than  $y$  in the diagram
- ▶ An edge between  $x$  and  $y$  ( $x$  lower than  $y$ ) implies  $x < y$  and no other element  $z$  exists s.t.  $x < z < y$  (i.e. transitivity is excluded)

## Lattice Diagram

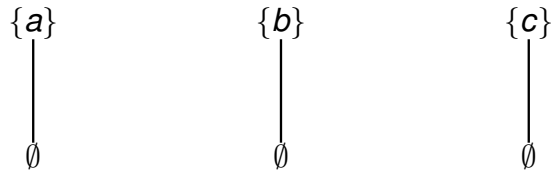


Lattice Diagram for  $(\{a, b, c\}, \cap)$

$x \wedge y$  = the highest  $z$  for which there are paths downward from both  $x$  and  $y$ .

## What if there is a large number of elements?

- ▶ Combine simple lattices to build a complex one
- ▶ Superset lattices for singletons

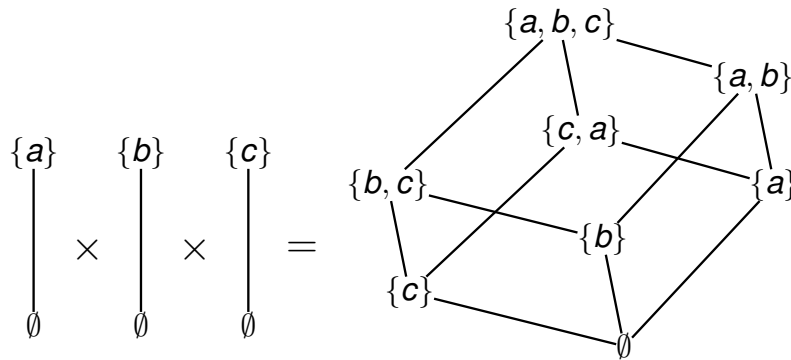


- ▶ Combine to form superset lattice for multi-element sets

## Product Lattice

- ▶  $(S, \wedge)$  is product lattice of  $(S_1, \wedge_1)$  and  $(S_2, \wedge_2)$  when  
 $S = S_1 \times S_2$  (domain)  
For  $(a_1, a_2)$  and  $(b_1, b_2) \in S$   
 $(a_1, a_2) \wedge (b_1, b_2) = (a_1 \wedge_1 b_1, a_2 \wedge_2 b_2)$   
 $(a_1, a_2) \leq (b_1, b_2)$  iff  $a_1 \leq_1 b_1$  and  $a_2 \leq_2 b_2$   
 $\leq$  relation follows from  $\wedge$
- ▶ Product of lattices is associative
- ▶ Can be generalized to product of  $N > 2$  lattices
- ▶  $(S_1, \wedge_1), (S_2, \wedge_2), \dots$  are called component lattices

## Product Lattice: Example



## Height of a Semilattice

- ▶ Length of a chain  $x_1 \leq x_2 \leq \dots \leq x_k$  is  $k$
- ▶ Let  $K = \max$  over lengths of all the chains in a semilattice
- ▶ Height of the semilattice =  $K - 1$

## Data Flow Analysis Framework

- ▶  $(D, S, \wedge, F)$
- ▶  $D$ : direction – Forward or Backward
- ▶  $(S, \wedge)$ : Semilattice – Domain and meet
- ▶  $F$ : family of transfer functions of type  $S \rightarrow S$  (see next slide)

## Transfer Functions

- ▶  $F$ : family of functions  $S \rightarrow S$ . Must Include
  - ▶ functions suitable for the boundary conditions (constant transfer functions for *Entry* and *Exit* nodes)
  - ▶ Identity function  $I$ :

$$I(x) = x \quad \forall x \in S$$

- ▶ Closed under composition:

$$f, g \in F, \quad f \circ g \Rightarrow h \in F$$

## Monotonic Functions

- ▶  $(S, \leq)$ : a poset
- ▶  $f : S \rightarrow S$  is monotonic iff
$$\forall x, y \in S \quad x \leq y \Rightarrow f(x) \leq f(y)$$
- ▶ Composition preserves monotonicity
  - ▶ If  $f$  and  $g$  are monotonic,  $h = f \circ g$ , then  $h$  is also monotonic

## Monotone Frameworks

- ▶  $(D, S, \wedge, F)$  is monotone if the family  $F$  consists of monotonic functions only

$$f \in F, \quad \forall x, y \in S \quad x \leq y \Rightarrow f(x) \leq f(y)$$

- ▶ Equivalently

$$f \in F, \quad \forall x, y \in S \quad f(x \wedge y) \leq f(x) \wedge f(y)$$

- ▶ Proof? : QQ in class

## Knaster-Tarski Fixed Point Theorem

- ▶ Let  $f$  be a monotonic function on a complete lattice  $(S, \wedge, \vee)$ . Define
  - ▶  $\text{red}(f) = \{v \mid v \in S, f(v) \leq v\}$ , pre fix-points
  - ▶  $\text{ext}(f) = \{v \mid v \in S, f(v) \geq v\}$ , post fix-points
  - ▶  $\text{fix}(f) = \{v \mid v \in S, f(v) = v\}$ , fix-points

Then,

- ▶  $\bigwedge \text{red}(f) \in \text{fix}(f)$ . Further,  $\bigwedge \text{red}(f) = \bigwedge \text{fix}(f)$
- ▶  $\bigvee \text{ext}(f) \in \text{fix}(f)$ . Further,  $\bigvee \text{ext}(f) = \bigvee \text{fix}(f)$
- ▶  $\text{fix}(f)$  is a complete lattice

## Application of Fixed Point Theorem

- ▶  $f : S \rightarrow S$  is a **monotonic** function
- ▶  $(S, \wedge)$  is a **finite height** semilattice
- ▶  $\top$  is the top element of  $(S, \wedge)$
- ▶ Notation:  $f^0(x) = x, f^{i+1}(x) = f(f^i(x)), \forall i \geq 0$
- ▶ The greatest fixed point of  $f$  is

$$f^k(\top), \text{ where } f^{k+1}(\top) = f^k(\top)$$

## Fixed Point Algorithm

```
// monotonic function f on a meet semilattice
x :=  $\top$ ;
while (x  $\neq$  f(x)) x := f(x);
return x;
```