Outline

1 Introduction

2 Static Analysis: a Gentle Introduction

3 A General Framework in Transitional Style

4 A Technique for Scalability: Sparse Analysis

5 Specialized Frameworks

Scalability Challenge

Figure: Call graph of `less-382` (23,822 lines of code)

Sparse Analysis

- Exploit the semantic sparsity of the input program to analyze
- Spatial sparsity & temporal sparsity

Right part at right moment

Example Performance Gain by Sparse Analysis

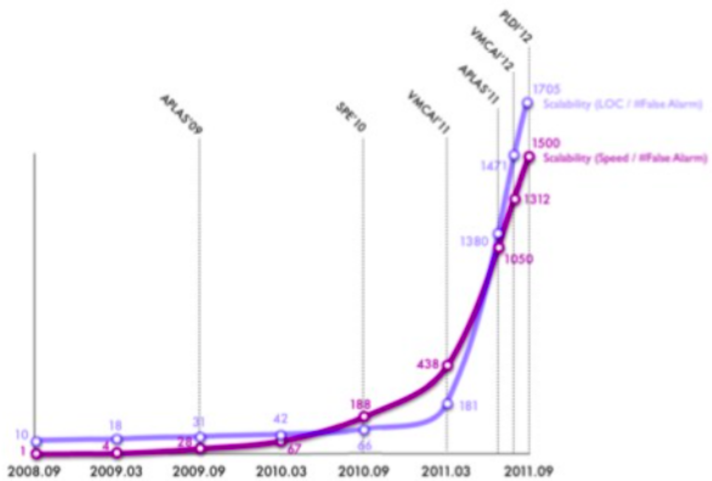
- Sparrow: a *sound*, global C analyzer for the memory safety property (no overrun, no null-pointer dereference, etc.)

  `http://github.com/ropas/sparrow`

- ~ 10 hours in analyzing million lines of C [PLDI'12, TOPLAS'14]

Sparrow The Early Bird

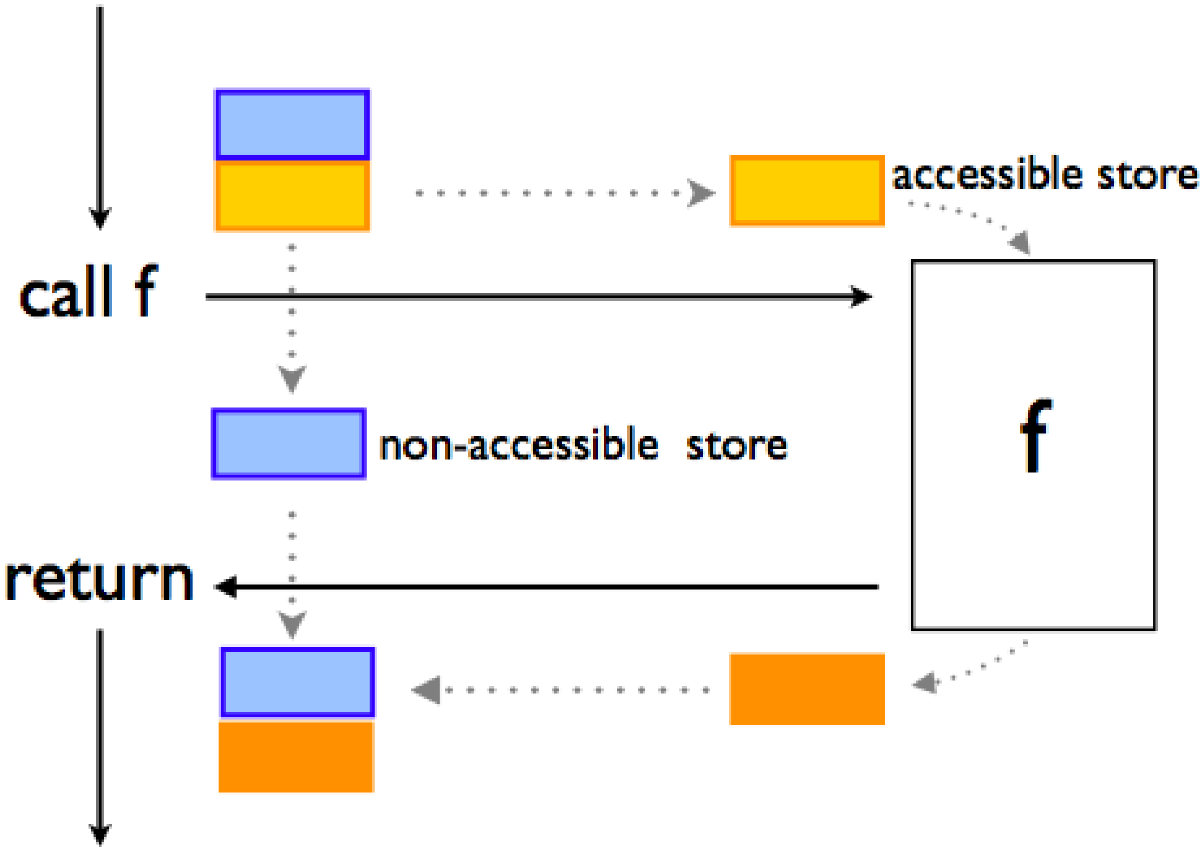
sound-&-global version



- < 1.4M in 10hrs with intervals
- < 0.14M in 20hrs with octagons

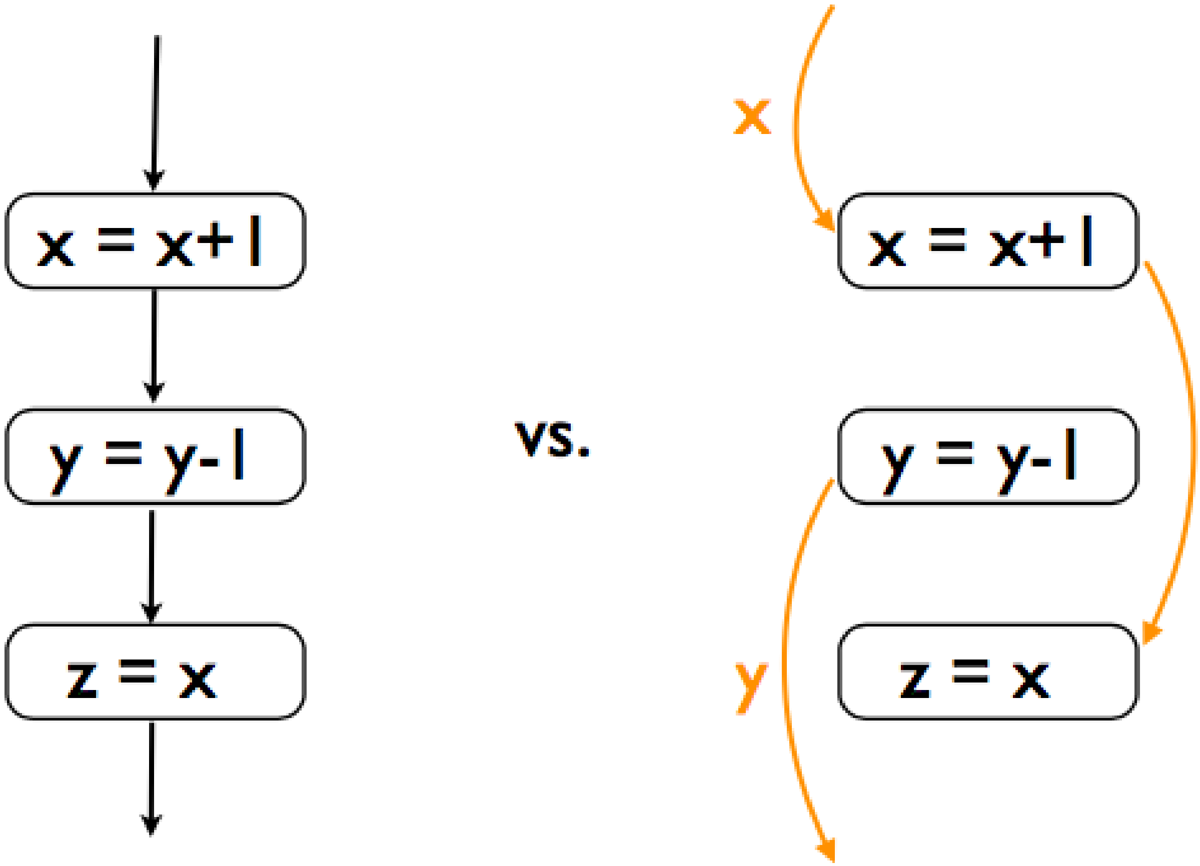
Spatial Sparcity

Each program portion accesses only a small part of the memory.



Temporal Sparcity

After the def of a memory, its use is far.

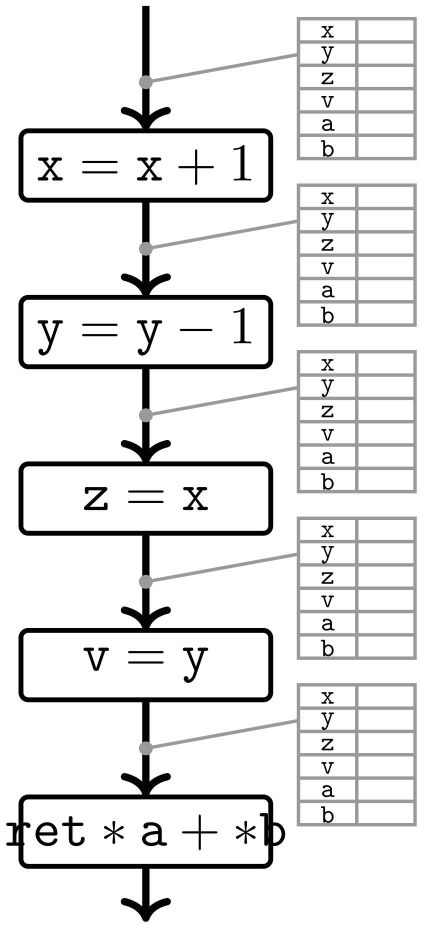


Example (Code fragment)

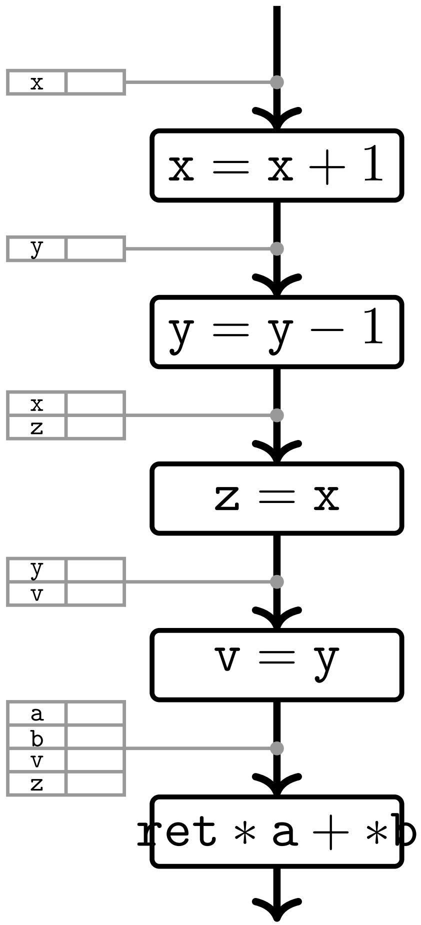
```
x = x + 1;
y = y - 1;
z = x;
v = y;
ret *a + *b
```

Assume that `a` points to `v` and `b` to `z`.

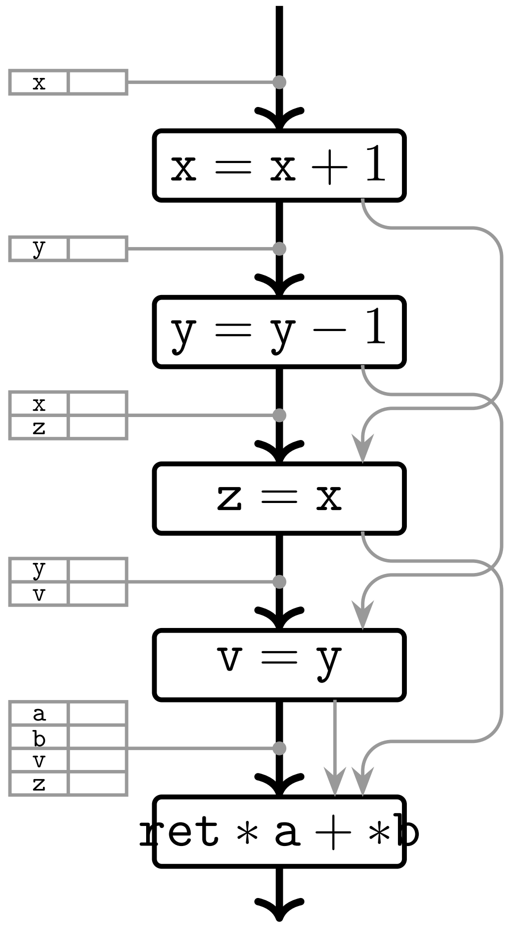
Spatial and Temporal Sparsity of the Example Code



(a) Without exploiting the sparsities



(b) Spatial sparsity



(c) Spatial & temporal sparsity

Exploiting Spatial Sparsity: Need $Access^\sharp(l)$

"abstract garbage collecition", "frame rule"

$$F^\sharp : (\mathbb{L} \to \mathbb{M}^\sharp) \to (\mathbb{L} \to \mathbb{M}^\sharp)$$

becomes

$$F^\sharp_{sparse} : (\mathbb{L} \to \mathbb{M}^\sharp_{sparse}) \to (\mathbb{L} \to \mathbb{M}^\sharp_{sparse})$$

where

$$\mathbb{M}^\sharp_{sparse} = \{M^\sharp \in \mathbb{M}^\sharp \mid dom(M^\sharp) = Access^\sharp(l), l \in \mathbb{L}\} \cup \{\perp\}.$$

Exploiting Temporal Sparsity: Need Def-Use Chain

Need the def-use chain information as follows.

- we streamline the abstract one-step relation

$$(l, M^\sharp) \hookrightarrow^\sharp (l', M'^\sharp) \quad \text{for } l' \in \texttt{next}^\sharp(l, M^\sharp).$$

  so that the link $\hookrightarrow^\sharp$ should follow the **def-use chain**:
  - from (def) a label where a location is defined
  - to (use) a label where the defined location is read

Precision Preserving Sparse Analysis Framework

Goal

$$F^\sharp : D^\sharp \to D^\sharp \quad \xrightarrow{\text{sparsify}} \quad F^\sharp_{sparse} : D^\sharp \to D^\sharp$$

$$\mathsf{lfp} F^\sharp \quad \stackrel{\text{still}}{=} \quad \mathsf{lfp} F^\sharp_{sparse}$$

Precision Preserving Sparse Analysis: for Spatial Sparsity (1/3)

Need to safely estimate

$$Access^\sharp(l).$$

Use yet another sound static analysis, a futher abstraction:

$$(\mathbb{L} \to \mathbb{M}^\sharp, \sqsubseteq) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (\mathbb{M}^\sharp, \sqsubseteq_M)$$

(a “flow-insensitive” version of the “flow-sensitive” analysis design)

Precision Preserving Sparse Analysis: for Temporal Sparsity (2/3)

- Let
$$D^\sharp : \mathbb{L} \to \wp(\mathbb{X}) \text{ and } U^\sharp : \mathbb{L} \to \wp(\mathbb{X})$$
be the def and use sets from the original analysis.
- Need to safely estimate $D^\sharp$ and $U^\sharp$.
- Use yet another sound static analysis to compute
$$D^\sharp_{pre} \text{ and } U^\sharp_{pre}$$
such that
  - $\forall l \in \mathbb{L} : D^\sharp_{pre}(l) \supseteq D^\sharp(l) \quad \text{and} \quad U^\sharp_{pre}(l) \supseteq U^\sharp(l).$
  - $\forall l \in \mathbb{L} : U^\sharp_{pre}(l) \supseteq D^\sharp_{pre}(l) \setminus D^\sharp(l).$

Precision Preserving Sparse Analysis: for Temporal Sparsity (3/3)

Let $D^\sharp_{pre}$ and $U^\sharp_{pre}$ be, respectively, safe def and use sets from a pre-analysis as defined before.

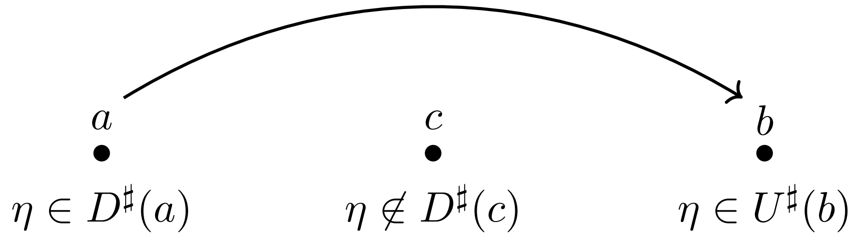
Definition (Precision preserving def-use chain)

Label a to label b is a def-use chain for an abstract location η whenever $\eta \in D^\sharp_{pre}(a)$, $\eta \in U^\sharp_{pre}(b)$, and η may not be re-defined inbetween the two labels.

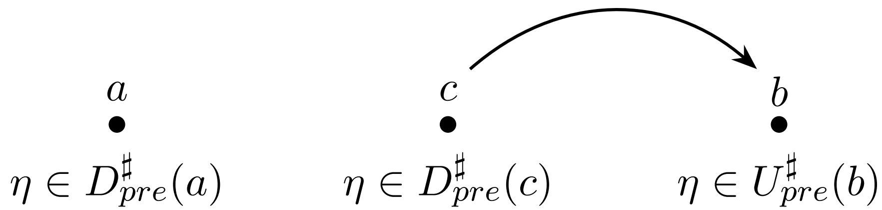
Precision preservation

Then, the resulting sparse analysis version has the same precision as the original non-sparse analysis.

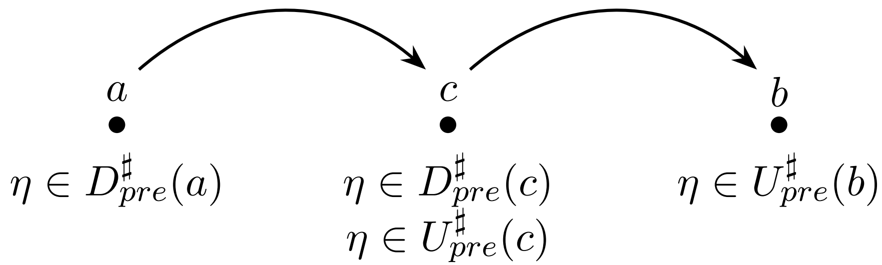
Need for the Second Condition for $D^\sharp_{pre}$ and $U^\sharp_{pre}$



(d) Original analysis def-use edge for η



(e) Missing def-use edge (a to b) for η because of over-approximate $D^\sharp_{pre}(c)$



(f) Recovered def-use edge (a to b via c) for η by safe $U^\sharp_{pre}(c)$