Script **generated by TTT**

Title: Seidl: Programmoptimierung (23.10.2013)

Date: Wed Oct 23 08:34:20 CEST 2013

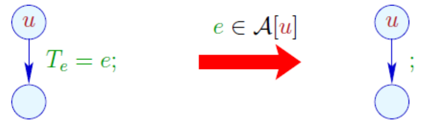
Duration: 85:44 min

Pages: 42

... analogously for $R = M[e];$ and $M[e_1] = e_2;$.

Transformation 1.2:

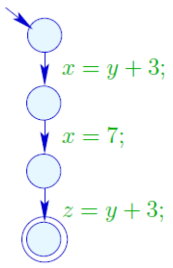
If e is available at program point u, then e need not be re-evaluated:



We replace the assignment with Nop :-)

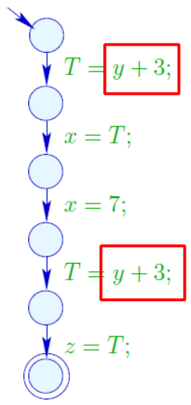
Example:

$$
\begin{array}{lcl}
x & = & \boxed{y+3;} \\
x & = & 7; \\
z & = & \boxed{y+3;}
\end{array}
$$



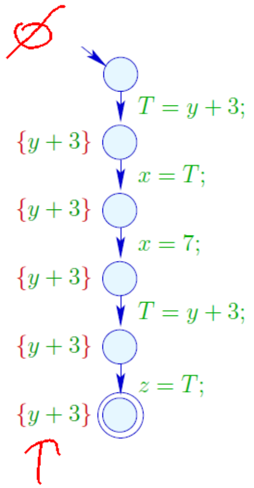
Example:

$$
\begin{array}{lcl}
x & = & y+3; \\
x & = & 7; \\
z & = & y+3;
\end{array}
$$



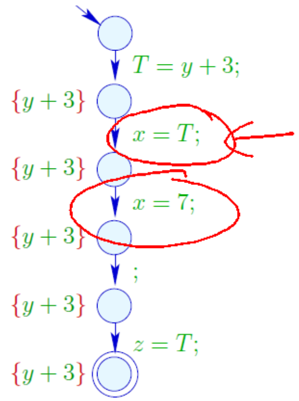
Example:

$$x = y + 3;$$
$$x = 7;$$
$$z = y + 3;$$



Example:

$$x = y + 3;$$
$$x = 7;$$
$$z = y + 3;$$



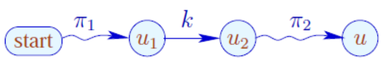
Correctness: (Idea)

Transformation 1.1 preserves the semantics and $\mathcal{A}[u]$ for all program points u :-)

Assume $\pi : start \to^* u$ is the path taken by a computation.

If $e \in \mathcal{A}[u]$, then also $e \in [\![\pi]\!]^\sharp \emptyset$.

Therefore, π can be decomposed into:



with the following properties:

- The expression e is evaluated at the edge k;
- The expression e is not removed from the set of available expressions at any edge in π_2, i.e., no variable of e receives a new value :-)

$$\Longrightarrow$$

The register T_e contains the value of e whenever u is reached :-))

Warning:

Transformation 1.1 is only meaningful for assignments $x = e;$ where:

→ $x \notin Vars(e)$;

→ $e \notin Vars$;

→ the evaluation of e is non-trivial :-}

Which leaves us with the following question ...

Question:

How do we compute $\mathcal{A}[u]$ for every program point u ??

Question:

How can we compute $\mathcal{A}[u]$ for every program point u ??

We collect all restrictions to the values of $\mathcal{A}[u]$ into a system of constraints:

$$\begin{array}{lll} \mathcal{A}[start] & \subseteq & \emptyset \\ \mathcal{A}[v] & \subseteq & [\![k]\!]^\sharp\,(\mathcal{A}[u]) \qquad k = (u, \_, v) \quad \text{edge} \end{array}$$

$$A[v] = \bigcap \{ [\![k]\!]^\sharp A[u] \mid k = (u, \_, v) \}$$

Question:

How can we compute $\mathcal{A}[u]$ for every program point u ??

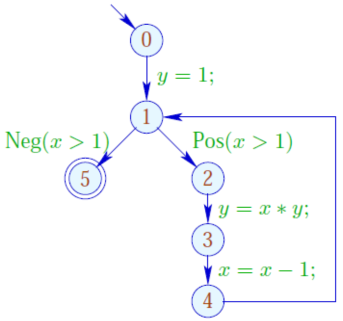
We collect all restrictions to the values of $\mathcal{A}[u]$ into a system of constraints:

$$\begin{array}{lll} \mathcal{A}[start] & \subseteq & \emptyset \\ \mathcal{A}[v] & \subseteq & [\![k]\!]^\sharp\,(\mathcal{A}[u]) \qquad k = (u, \_, v) \quad \text{edge} \end{array}$$

Wanted:

- a maximally large solution (??)
- an algorithm which computes this :-)

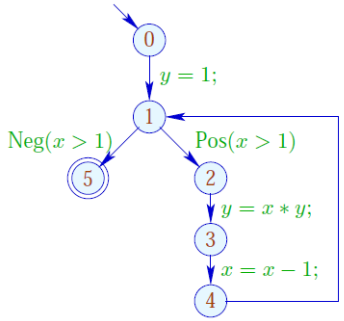
Example:



Wanted:

- a maximally large solution (??)
- an algorithm which computes this :-)

Example:

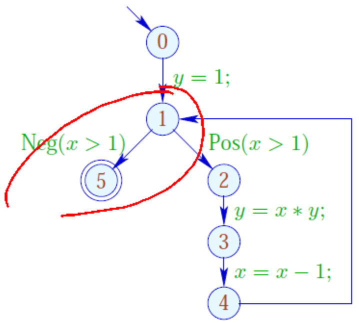


$$\mathcal{A}[0] \subseteq \emptyset$$

Wanted:

- a maximally large solution (??)
- an algorithm which computes this :-)

Example:

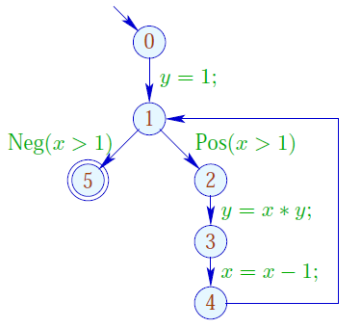


$$\begin{array}{lcl}
\mathcal{A}[0] & \subseteq & \emptyset \\
\mathcal{A}[1] & \subseteq & (\mathcal{A}[0] \cup \{1\}) \backslash Expr_y \\
\mathcal{A}[1] & \subseteq & \mathcal{A}[4] \\
\mathcal{A}[2] & \subseteq & \mathcal{A}[1] \cup \{x > 1\} \\
\mathcal{A}[3] & \subseteq & (\mathcal{A}[2] \cup \{x * y\}) \backslash Expr_y \\
\mathcal{A}[4] & \subseteq & (\mathcal{A}[3] \cup \{x - 1\}) \backslash Expr_x \\
\mathcal{A}[5] & \subseteq & \mathcal{A}[1] \cup \{x > 1\}
\end{array}$$

Wanted:

- a maximally large solution (??)
- an algorithm which computes this :-)

Example:



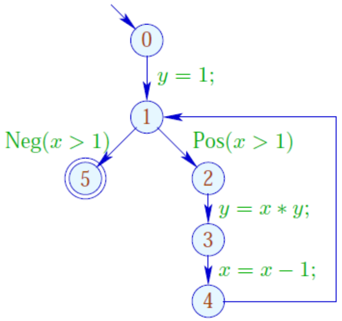
Solution:

$$\begin{array}{lcl}
\mathcal{A}[0] & = & \emptyset \\
\mathcal{A}[1] & = & \{1\} \\
\mathcal{A}[2] & = & \{1, x > 1\} \\
\mathcal{A}[3] & = & \{1, x > 1\} \\
\mathcal{A}[4] & = & \{1\} \\
\mathcal{A}[5] & = & \{1, x > 1\}
\end{array}$$

Wanted:

- a maximally large solution (??)
- an algorithm which computes this :-)

Example:



Solution:

$$\begin{array}{lcl} \mathcal{A}[0] & = & \emptyset \\ \mathcal{A}[1] & = & \{1\} \\ \mathcal{A}[2] & = & \{1, x > 1\} \\ \mathcal{A}[3] & = & \{1, x > 1\} \\ \mathcal{A}[4] & = & \{1\} \\ \mathcal{A}[5] & = & \{1, x > 1\} \end{array}$$

Observation:

- The possible values for $\mathcal{A}[u]$ form a complete lattice:

$$\mathbb{D} = 2^{Expr} \quad \text{with} \quad B_1 \sqsubseteq B_2 \quad \text{iff} \quad B_1 \supseteq B_2$$

Observation:

- The possible values for $\mathcal{A}[u]$ form a complete lattice:

$$\mathbb{D} = 2^{Expr} \quad \text{with} \quad B_1 \sqsubseteq B_2 \quad \text{iff} \quad B_1 \supseteq B_2$$

- The functions $[\![k]\!]^\sharp : \mathbb{D} \to \mathbb{D}$ are monotonic, i.e.,

$$[\![k]\!]^\sharp(B_1) \sqsubseteq [\![k]\!]^\sharp(B_2) \quad \text{whenever} \quad B_1 \sqsubseteq B_2$$

Observation:

- The possible values for $\mathcal{A}[u]$ form a complete lattice:

$$\mathbb{D} = 2^{Expr} \quad \text{with} \quad B_1 \sqsubseteq B_2 \quad \text{iff} \quad B_1 \supseteq B_2$$

- The functions $[\![k]\!]^\sharp : \mathbb{D} \to \mathbb{D}$ are monotonic, i.e.,

$$[\![k]\!]^\sharp(B_1) \sqsubseteq [\![k]\!]^\sharp(B_2) \quad \text{whenever} \quad B_1 \sqsubseteq B_2$$

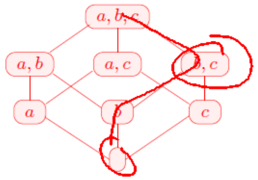
Background 2: Complete Lattices

A set $\mathbb{D}$ together with a relation $\sqsubseteq \subseteq \mathbb{D} \times \mathbb{D}$ is a partial order if for all $a, b, c \in \mathbb{D}$,

$a \sqsubseteq a$	*reflexivity*
$a \sqsubseteq b \wedge b \sqsubseteq a \implies a = b$	*anti−symmetry*
$a \sqsubseteq b \wedge b \sqsubseteq c \implies a \sqsubseteq c$	*transitivity*

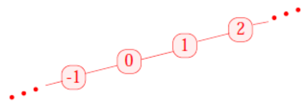
Examples:

1. $\mathbb{D} = 2^{\{a,b,c\}}$ with the relation "$\subseteq$" :

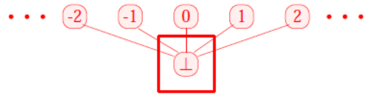


2. $\mathbb{Z}$ with the relation "$=$" :

3. $\mathbb{Z}$ with the relation "$\leq$" :



4. $\mathbb{Z}_\perp = \mathbb{Z} \cup \{\perp\}$ with the ordering:



$d \in \mathbb{D}$ is called upper bound for $X \subseteq \mathbb{D}$ if

$$x \sqsubseteq d \qquad \text{for all } x \in X$$

$d \in \mathbb{D}$ is called upper bound for $X \subseteq \mathbb{D}$ if

$$x \sqsubseteq d \qquad \text{for all } x \in X$$

d is called least upper bound (lub) if

1. d is an upper bound and
2. $d \sqsubseteq y$ for every upper bound y of X.

$d \in \mathbb{D}$ is called upper bound for $X \subseteq \mathbb{D}$ if

$$x \sqsubseteq d \qquad \text{for all } x \in X$$

d is called least upper bound (lub) if

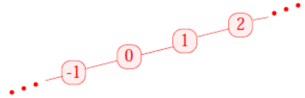
1. d is an upper bound and
2. $d \sqsubseteq y$ for every upper bound y of X.

Caveat:

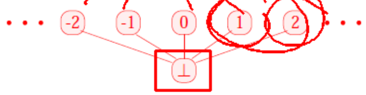
- $\{0, 2, 4, \ldots\} \subseteq \mathbb{Z}$ has no upper bound!
- $\{0, 2, 4\} \subseteq \mathbb{Z}$ has the upper bounds $4, 5, 6, \ldots$

2. $\mathbb{Z}$ with the relation "$=$" :

3. $\mathbb{Z}$ with the relation "$\leq$" :



4. $\mathbb{Z}_\perp = \mathbb{Z} \cup \{\perp\}$ with the ordering:



A complete lattice (cl) $\mathbb{D}$ is a partial ordering where every subset $X \subseteq \mathbb{D}$ has a least upper bound $\bigsqcup X \in \mathbb{D}$.

Note:

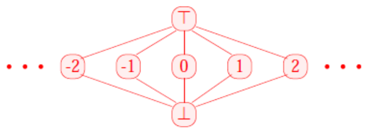
Every complete lattice has

→ a least element $\perp = \bigsqcup \emptyset \in \mathbb{D}$;

→ a greatest element $\top = \bigsqcup \mathbb{D} \in \mathbb{D}$.

Examples:

1. $\mathbb{D} = 2^{\{a,b,c\}}$ is a cl :-)
2. $\mathbb{D} = \mathbb{Z}$ with "$=$" is not.
3. $\mathbb{D} = \mathbb{Z}$ with "$\leq$" is neither.
4. $\mathbb{D} = \mathbb{Z}_\perp$ is also not :-(
5. With an extra element $\top$, we obtain the flat lattice $\mathbb{Z}_\perp^\top = \mathbb{Z} \cup \{\perp, \top\}$:



We have:

Theorem:

If $\mathbb{D}$ is a complete lattice, then every subset $X \subseteq \mathbb{D}$ has a greatest lower bound $\bigsqcap X$.

We have:

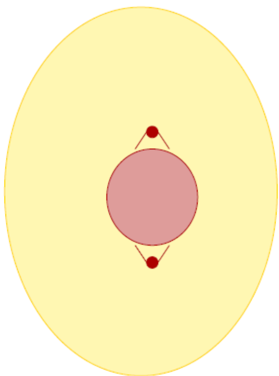
Theorem:

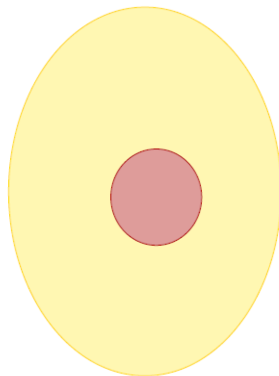
If $\mathbb{D}$ is a complete lattice, then every subset $X \subseteq \mathbb{D}$ has a greatest lower bound $\bigsqcap X$.

Proof:

Construct $U = \{u \in \mathbb{D} \mid \forall x \in X : u \sqsubseteq x\}$.

// the set of all lower bounds of X :-)





We have:

Theorem:

If $\mathbb{D}$ is a complete lattice, then every subset $X \subseteq \mathbb{D}$ has a greatest lower bound $\bigsqcap X$.

Proof:

Construct $U = \{u \in \mathbb{D} \mid \forall x \in X : \; u \sqsubseteq x\}$.

// the set of all lower bounds of X :-)

We have:

Theorem:

If $\mathbb{D}$ is a complete lattice, then every subset $X \subseteq \mathbb{D}$ has a greatest lower bound $\bigsqcap X$.

Proof:

Construct $U = \{u \in \mathbb{D} \mid \forall x \in X : \; u \sqsubseteq x\}$.

// the set of all lower bounds of X :-)

Set: $g := \bigsqcup U$

Claim: $g = \bigsqcap X$

(1) g is a lower bound of X :

Assume $x \in X$. Then:

$u \sqsubseteq x$ for all $u \in U$

$\Longrightarrow$ x is an upper bound of U

$\Longrightarrow$ $g \sqsubseteq x$:-)

We have:

Theorem:

If $\mathbb{D}$ is a complete lattice, then every subset $X \subseteq \mathbb{D}$ has a greatest lower bound $\bigsqcap X$.

Proof:

Construct $U = \{u \in \mathbb{D} \mid \forall x \in X : \; u \sqsubseteq x\}$.

// the set of all lower bounds of X :-)

Set: $g := \bigsqcup U$

Claim: $g = \bigsqcap X$

(1) g is a lower bound of X :

Assume $x \in X$. Then:

$u \sqsubseteq x$ for all $u \in U$

$\Longrightarrow$ x is an upper bound of U

$\Longrightarrow$ $g \sqsubseteq x$:-)

(1) g is a lower bound of X :

Assume $x \in X$. Then:

$u \sqsubseteq x$ for all $u \in U$

$\Longrightarrow$ x is an upper bound of U

$\Longrightarrow$ $g \sqsubseteq x$:-)

(2) g is the greatest lower bound of X :

Assume u is a lower bound of X. Then:

$u \in U$

$\Longrightarrow$ $u \sqsubseteq g$:-))

We are looking for solutions for systems of constraints of the form:

$$x_i \sqsupseteq f_i(x_1, \ldots, x_n) \qquad (*)$$

We are looking for solutions for systems of constraints of the form:

$$x_i \sqsupseteq f_i(x_1, \ldots, x_n) \qquad (*)$$

where:

x_i	unknown	here:	$\mathcal{A}[u]$
$\mathbb{D}$	values	here:	2^{Expr}
$\sqsubseteq \; \subseteq \; \mathbb{D} \times \mathbb{D}$	ordering relation	here:	$\supseteq$
$f_i \colon \mathbb{D}^n \to \mathbb{D}$	constraint	here:	...

We are looking for solutions for systems of constraints of the form:

$$x_i \sqsupseteq f_i(x_1, \ldots, x_n) \qquad (*)$$

where:

x_i	unknown	here:	$\mathcal{A}[u]$
$\mathbb{D}$	values	here:	2^{Expr}
$\sqsubseteq \; \subseteq \; \mathbb{D} \times \mathbb{D}$	ordering relation	here:	$\supseteq$
$f_i \colon \mathbb{D}^n \to \mathbb{D}$	constraint	here:	...

Constraint for $\mathcal{A}[v]$ $(v \neq start)$:

$$\mathcal{A}[v] \subseteq \bigcap\{[\![k]\!]^\sharp (\mathcal{A}[u]) \mid k = (u, \_, v) \text{ edge}\}$$

We are looking for solutions for systems of constraints of the form:

$$x_i \sqsupseteq f_i(x_1, \ldots, x_n) \qquad (*)$$

where:

x_i	unknown	here:	$\mathcal{A}[u]$
$\mathbb{D}$	values	here:	2^{Expr}
$\sqsubseteq \; \subseteq \; \mathbb{D} \times \mathbb{D}$	ordering relation	here:	$\supseteq$
$f_i \colon \mathbb{D}^n \to \mathbb{D}$	constraint	here:	...

Constraint for $\mathcal{A}[v]$ $(v \neq start)$:

$$\mathcal{A}[v] \subseteq \bigcap\{[\![k]\!]^\sharp (\mathcal{A}[u]) \mid k = (u, \_, v) \text{ edge}\}$$

Because:

$$x \sqsupseteq d_1 \wedge \ldots \wedge x \sqsupseteq d_k \quad \text{iff} \quad x \sqsupseteq \bigsqcup\{d_1, \ldots, d_k\} \qquad :-)$$

A mapping $f : \mathbb{D}_1 \to \mathbb{D}_2$ is called monotonic, if $f(a) \sqsubseteq f(b)$ for all $a \sqsubseteq b$.