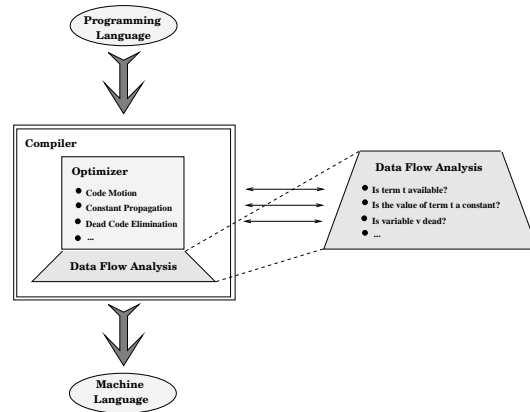
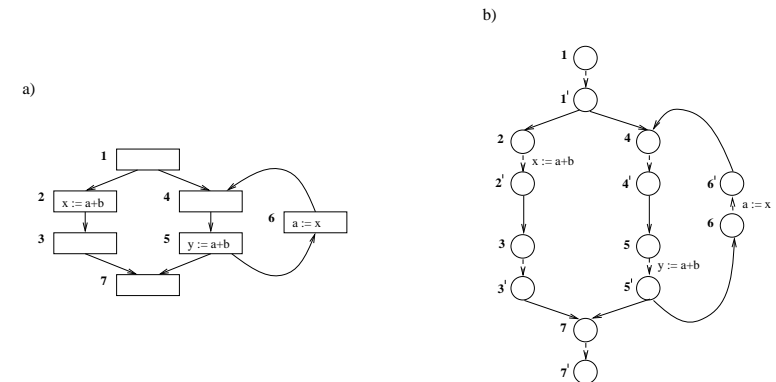


Erinnerung und Motivation (1)



Erinnerung und Motivation (2)

Knoten- vs. kantenbenannte Flussgraphen
(hier mit Einzelinstruktionenbenennung)



Erinnerung und Motivation (3)

Wir unterscheiden...

- Intraprozedurale
- Interprozedurale
- Parallele
- Konditionale
- ...

Datenflussanalyse.

Erinnerung und Motivation (4)

Ingredienzien *intraprozeduraler* Datenflussanalyse:

- (Lokale) *abstrakte Semantik*
 1. Ein *Datenflussanalyseverband* $\hat{\mathcal{C}} = (\mathcal{C}, \sqcap, \sqcup, \sqsubseteq, \perp, \top)$
 2. Ein *Datenflussanalysefunktional* $[[\]] : E \rightarrow (\mathcal{C} \rightarrow \mathcal{C})$
 3. Anfangszusicherung $c_s \in \mathcal{C}$
- *Globalisierungsstrategien*
 1. "Meet over all Paths"-Ansatz (*MOP*)
 2. Maximaler Fixpunktansatz (*MaxFP*)
- *Generischer Fixpunktalgorithmus*

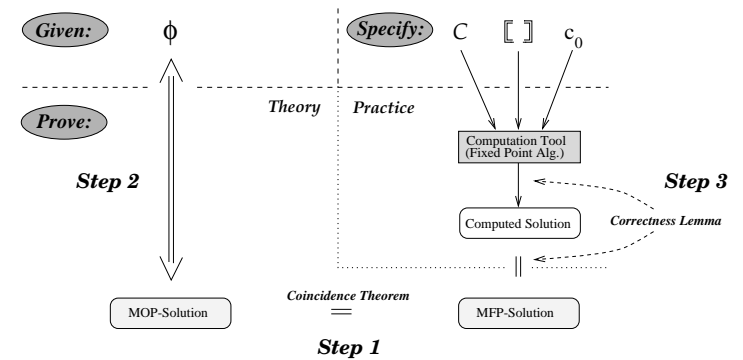
Erinnerung und Motivation (5)

Hauptresultate:

- Effektivitäts- (Terminierungs-) Theorem
- Sicherheits- (Korrektheits-) Theorem
- Koinzidenz- (Vollständigkeits-) Theorem

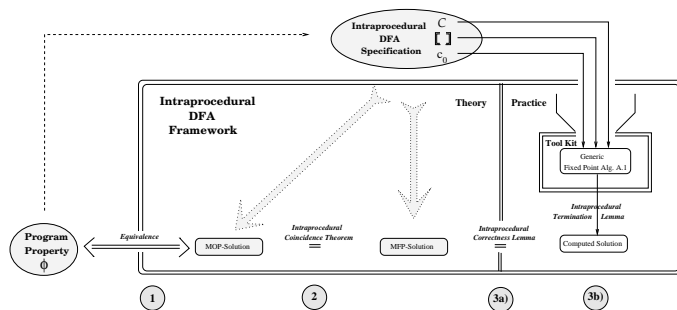
Intraprozedurale Datenflussanalyse (1)

...im Überblick:



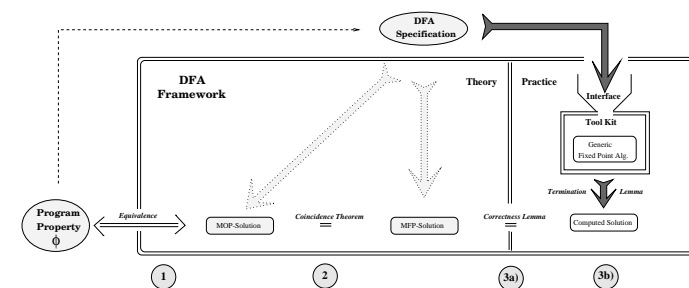
Intraprozedurale Datenflussanalyse (2)

...im Überblick:



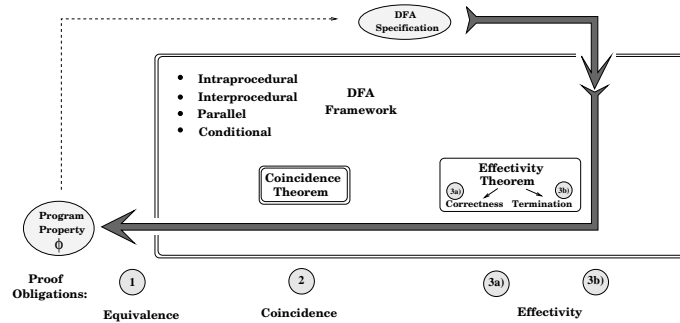
DFA-Frameworks / DFA-Toolkits (1)

...das allgemeine Muster:



DFA-Frameworks / DFA-Toolkits (2)

...das allgemeine Muster stärker abstrahiert:



Ausblick

...auf

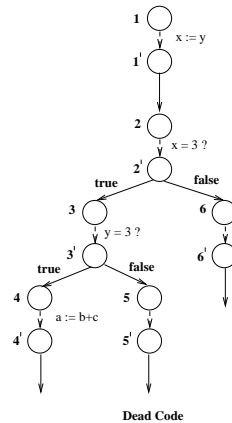
- konditionale,
- interprozedurale,
- parallele Datenflussanalyse

in...

- Rahmen- und
- Toolkit-Sicht.

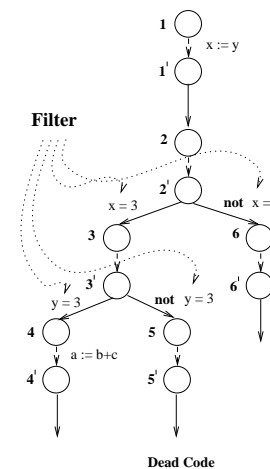
Konditionale Datenflussanalyse (1)

Ein Anwendungsbeispiel zur Motivation...



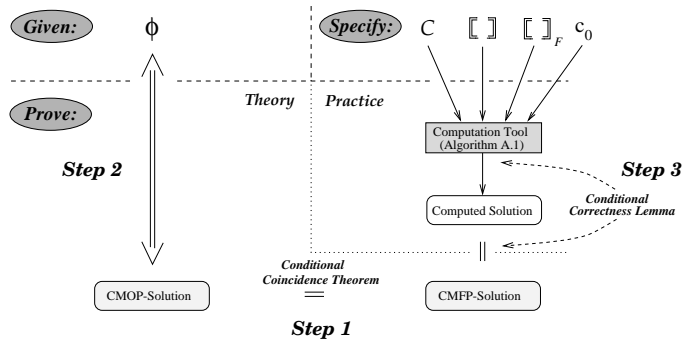
Konditionale Datenflussanalyse (2)

Technische Umsetzung: Mittels *Filterfunktionen*...



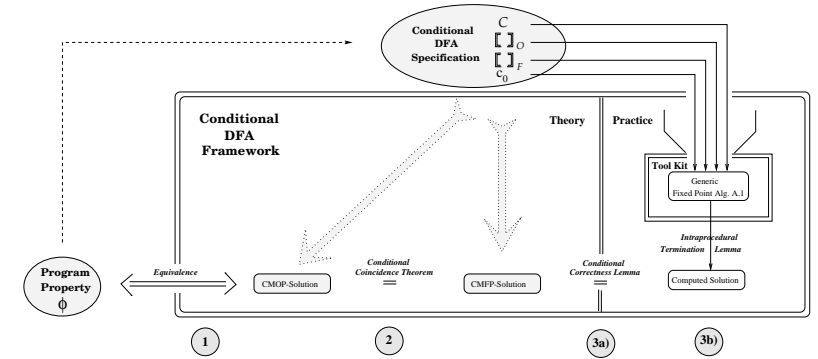
Konditionale Datenflussanalyse (3)

Der *konditionale* DFA-Rahmen im Überblick:



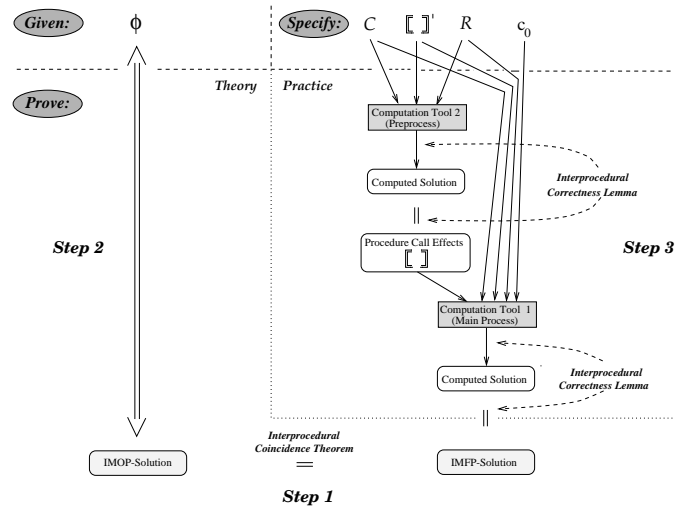
Konditionale Datenflussanalyse (4)

Der *konditionale* DFA-Rahmen in der Toolkit-Sicht:



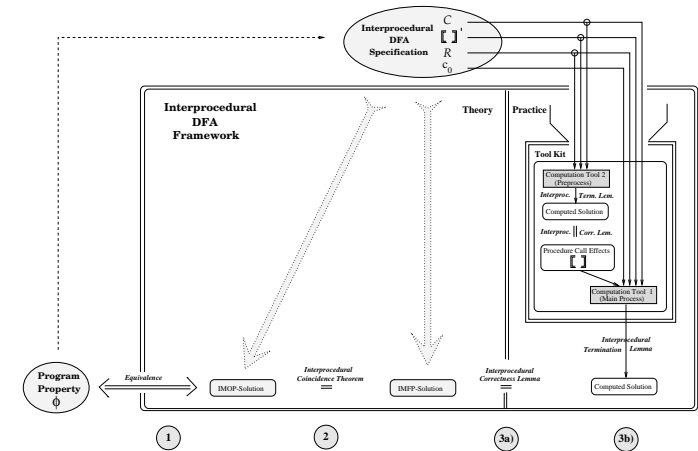
Interprozedurale Datenflussanalyse (1)

Der *interprozedurale* DFA-Rahmen im Überblick:



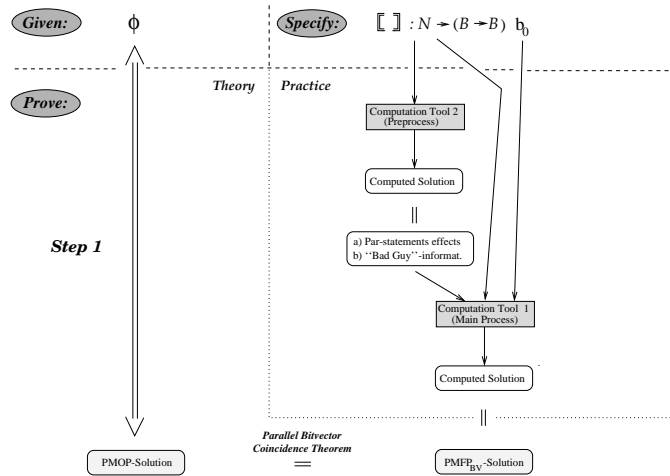
Interprozedurale Datenflussanalyse (2)

Der *interprozedurale* DFA-Rahmen in der Toolkit-Sicht:



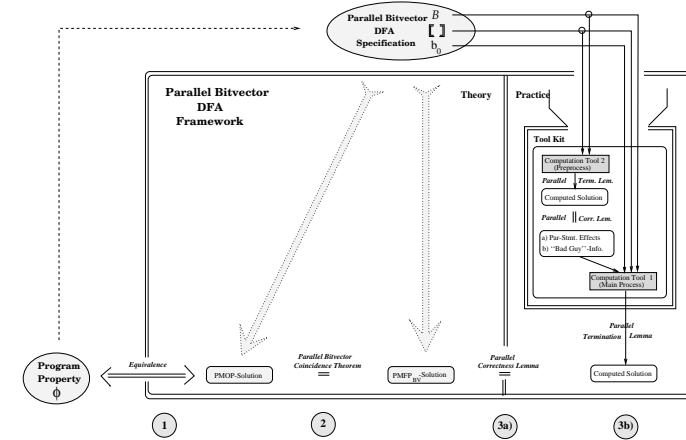
Parallele Datenflussanalyse (1)

Der *parallele* DFA-Rahmen im Überblick:



Parallele Datenflussanalyse (2)

Der *parallele* DFA-Rahmen in der Toolkit-Sicht:



In der Folge

...zweierlei:

- Beispiele konkreter Datenflussanalyse (-probleme)
- Basisblöcke vs. Einzelinstruktionen: Vor- und Nachteile

Basisblöcke (1)

...und die ihnen gemeinlich aus ihrer Verwendung zugeschriebenen Vorteile ("Folk Knowledge"):

- *Performanz*: "...weil weniger Knoten in die teure iterative Fixpunktberechnung involviert sind".
- *Kompaktheit*: "...weil größere Programme in den Hauptspeicher passen".

Basisblöcke (2)

...und die in der Folge aus ihrer Verwendung behaupteten Nachteile:

- *Höhere konzeptuelle Komplexität:* ... Basisblöcke führen zu einer unerwünschten *Hierarchisierung*, die sowohl theoretische Überlegungen wie Implementierungen erschwert.
- *Notwendigkeit von Prä- und Postprozessen:* ... oft erforderlich, um die hierarchieinduzierten Zusatzprobleme zu behandeln (z.B. bei *dead code elimination*, *constant propagation*, ...); oder "trickbehaftete" Formulierungen nötig macht, um sie umzugehen (z.B. bei *partial redundancy elimination*).
- *Eingeschränkte Allgemeinheit:* ... bestimmte praktisch relevante Analysen und Optimierungen sind nur schwer oder gar nicht auf Basisblockebene auszudrücken (z.B. *faint variable analysis and elimination*).

MOP -Ansatz (Einzelinstruktionen)

... für kantenbenannte Einzelinstruktionsgraphen:

Die *MOP-Lösung*:

$$\forall c_s \in \mathcal{C} \forall n \in N. MOP_{(\llbracket \cdot \rrbracket_l, c_s)}(n) =_{df} \bigcap \{ \llbracket p \rrbracket_l(c_s) \mid p \in \mathbf{P}_G[s, n] \}$$

MaxFP -Ansatz (Einzelinstruktionen)

... für kantenbenannte Einzelinstruktionsgraphen:

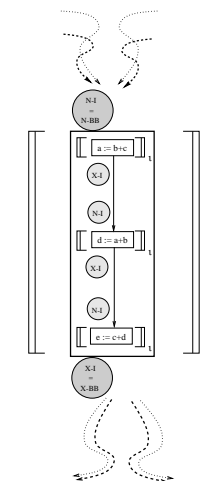
Die *MaxFP-Lösung*:

$$\forall c_s \in \mathcal{C} \forall n \in N. MaxFP_{(\llbracket \cdot \rrbracket_l, c_s)}(n) =_{df} info_{c_s}(n)$$

wobei $info_{c_s}$ die größte Lösung des folgenden Gleichungssystems bezeichnet:

$$info(n) = \begin{cases} c_s & \text{falls } n = s \\ \bigcap \{ \llbracket (m, n) \rrbracket_l(info(m)) \mid m \in pred_G(n) \} & \text{sonst} \end{cases}$$

Hierarchisierung durch Basisblöcke



MOP -Ansatz (Basisblöcke) (1)

... für knotenbenannte Basisblockgraphen:

The MOP-Solution: (Basisblockebene)

$\forall c_s \in \mathcal{C} \forall \mathbf{n} \in \mathbf{N}. MOP_{(\llbracket \cdot \rrbracket_{\beta, c_s})}(\mathbf{n}) =_{df} (N-MOP_{(\llbracket \cdot \rrbracket_{\beta, c_s})}(\mathbf{n}), X-MOP_{(\llbracket \cdot \rrbracket_{\beta, c_s})}(\mathbf{n}))$
mit

$$N-MOP_{(\llbracket \cdot \rrbracket_{\beta, c_s})}(\mathbf{n}) =_{df} \bigcap \{ \llbracket p \rrbracket_{\beta}(c_s) \mid p \in \mathbf{P}_G[s, \mathbf{n}] \}$$

$$X-MOP_{(\llbracket \cdot \rrbracket_{\beta, c_s})}(\mathbf{n}) =_{df} \bigcap \{ \llbracket p \rrbracket_{\beta}(c_s) \mid p \in \mathbf{P}_G[s, \mathbf{n}] \}$$

MOP -Ansatz (Basisblöcke) (2)

... für knotenbenannte Basisblockgraphen:

The MOP-Solution: (Instruktionsebene)

$\forall c_s \in \mathcal{C} \forall n \in \mathbf{N}. MOP_{(\llbracket \cdot \rrbracket_{l, c_s})}(n) =_{df} (N-MOP_{(\llbracket \cdot \rrbracket_{l, c_s})}(n), X-MOP_{(\llbracket \cdot \rrbracket_{l, c_s})}(n))$

mit

$$N-MOP_{(\llbracket \cdot \rrbracket_{l, c_s})}(n) =_{df} \begin{cases} N-MOP_{(\llbracket \cdot \rrbracket_{l, c_s})}(\text{block}(n)) & \text{falls } n = \text{start}(\text{block}(n)) \\ \llbracket p \rrbracket_l(N-MOP_{(\llbracket \cdot \rrbracket_{l, c_s})}(\text{block}(n))) & \text{sonst (} p \text{ Präfixpfad} \\ & \text{von } \text{start}(\text{block}(n)) \\ & \text{bis (ausschl.) } n \end{cases}$$
$$X-MOP_{(\llbracket \cdot \rrbracket_{l, c_s})}(n) =_{df} \llbracket p \rrbracket_l(N-MOP_{(\llbracket \cdot \rrbracket_{l, c_s})}(\text{block}(n))) \quad (p \text{ Präfix} \\ \text{von } \text{start}(\text{block}(n)) \text{ bis (einschl.) } n)$$

MaxFP -Ansatz (Basisblöcke) (1)

... für knotenbenannte Basisblockgraphen:

Die MaxFP-Lösung: (Basisblockebene)

$\forall c_s \in \mathcal{C} \forall \mathbf{n} \in \mathbf{N}. MaxFP_{(\llbracket \cdot \rrbracket_{\beta, c_s})}(\mathbf{n}) =_{df} (N-MFP_{(\llbracket \cdot \rrbracket_{\beta, c_s})}(\mathbf{n}), X-MFP_{(\llbracket \cdot \rrbracket_{\beta, c_s})}(\mathbf{n}))$

mit

$$N-MFP_{(\llbracket \cdot \rrbracket_{\beta, c_s})}(\mathbf{n}) =_{df} \text{pre}_{c_s}^{\beta}(\mathbf{n}) \quad \text{und} \quad X-MFP_{(\llbracket \cdot \rrbracket_{\beta, c_s})}(\mathbf{n}) =_{df} \text{post}_{c_s}^{\beta}(\mathbf{n})$$

wobei $\text{pre}_{c_s}^{\beta}$ und $\text{post}_{c_s}^{\beta}$ die größten Lösungen folgenden Gleichungssystems bezeichnen:

$$\text{pre}(\mathbf{n}) = \begin{cases} c_s & \text{falls } \mathbf{n} = s \\ \bigcap \{ \text{post}(\mathbf{m}) \mid \mathbf{m} \in \text{pred}_G(\mathbf{n}) \} & \text{sonst} \end{cases}$$
$$\text{post}(\mathbf{n}) = \llbracket \mathbf{n} \rrbracket_{\beta}(\text{pre}(\mathbf{n}))$$

MaxFP -Ansatz (Basisblöcke) (2)

Die MaxFP-Lösung: (Instruktionsebene)

$\forall c_s \in \mathcal{C} \forall n \in \mathbf{N}. MaxFP_{(\llbracket \cdot \rrbracket_{l, c_s})}(n) =_{df} (N-MFP_{(\llbracket \cdot \rrbracket_{l, c_s})}(n), X-MFP_{(\llbracket \cdot \rrbracket_{l, c_s})}(n))$

mit

$$N-MFP_{(\llbracket \cdot \rrbracket_{l, c_s})}(n) =_{df} \text{pre}_{c_s}^l(n) \quad \text{und} \quad X-MFP_{(\llbracket \cdot \rrbracket_{l, c_s})}(n) =_{df} \text{post}_{c_s}^l(n)$$

wobei $\text{pre}_{c_s}^l$ und $\text{post}_{c_s}^l$ die größten Lösungen des folgenden Gleichungssystems bezeichnen:

$$\text{pre}(n) = \begin{cases} \text{pre}_{c_s}^{\beta}(\text{block}(n)) & \text{falls } n = \text{start}(\text{block}(n)) \\ \text{post}(m) & \text{sonst (} m \text{ ist eindeutig bestimmter} \\ & \text{Vorgänger von } n \text{ in } \text{block}(n) \end{cases}$$
$$\text{post}(n) = \llbracket n \rrbracket_l(\text{pre}(n))$$

Bsp.: Verfügbarkeit von Ausdrücken (1)

Verfügbarkeit für knotenbenannte BB-Graphen:

Phase I: Die Basisblockebene

Lokale Prädikate: (assoziiert mit BB-Knoten)

- $\text{BB-XCOMP}_\beta(t)$: β contains an instruction ι computing t , and neither ι nor any instruction of β following ι modifies an operand of t .
- $\text{BB-TRANSP}_\beta(t)$: β contains no instruction modifying an operand of t .

Bsp.: Verfügbarkeit von Ausdrücken (2)

Das Gleichungssystem von Phase I:

$$\text{BB-N-AVAIL}_\beta = \begin{cases} \text{false} & \text{if } \beta = s \\ \prod_{\tilde{\beta} \in \text{pred}(\beta)} \text{BB-X-AVAIL}_{\tilde{\beta}} & \text{otherwise} \end{cases}$$

$$\text{BB-X-AVAIL}_\beta = \text{BB-N-AVAIL}_\beta \cdot \text{BB-TRANSP}_\beta + \text{BB-XCOMP}_\beta$$

Bsp.: Verfügbarkeit von Ausdrücken (3)

Phase II: Die Instruktionsebene

Lokale Prädikate: (assoziiert mit EI-Knoten)

- $\text{COMP}_\iota(t)$: ι computes t .
- $\text{TRANSP}_\iota(t)$: ι does not modify an operand of t .
- BB-N-AVAIL^* , BB-X-AVAIL^* : greatest solution of the equation system of Phase I.

Das Gleichungssystem von Phase II:

$$\text{N-AVAIL}_\iota = \begin{cases} \text{BB-N-AVAIL}_{\text{block}(\iota)}^* & \text{if } \iota = \text{start}(\text{block}(\iota)) \\ \text{X-AVAIL}_{\text{pred}(\iota)} & \text{otherwise (note that } |\text{pred}(\iota)| = 1) \end{cases}$$

$$\text{X-AVAIL}_\iota = \begin{cases} \text{BB-X-AVAIL}_{\text{block}(\iota)}^* & \text{if } \iota = \text{end}(\text{block}(\iota)) \\ (\text{N-AVAIL}_\iota + \text{COMP}_\iota) \cdot \text{TRANSP}_\iota & \text{otherwise} \end{cases}$$

Bsp.: Verfügbarkeit von Ausdrücken (4)

Verfügbarkeit für knotenbenannte EI-Graphen:

Lokale Prädikate: (assoziiert mit Knoten)

- $\text{COMP}_\iota(t)$: ι computes t .
- $\text{TRANSP}_\iota(t)$: ι does not modify an operand of t .

Das Gleichungssystem:

$$\text{N-AVAIL}_\iota = \begin{cases} \text{false} & \text{if } \iota = s \\ \prod_{\tilde{\iota} \in \text{pred}(\iota)} \text{X-AVAIL}_{\tilde{\iota}} & \text{otherwise} \end{cases}$$

$$\text{X-AVAIL}_\iota = (\text{N-AVAIL}_\iota + \text{COMP}_\iota) \cdot \text{TRANSP}_\iota$$

Bsp.: Verfügbarkeit von Ausdrücken (5)

Verfügbarkeit für kantenbenannte EI-Graphen:

Lokale Prädikate: (assoziiert mit EI-Kanten)

- $\mathbf{COMP}_\varepsilon(t)$: instruction ι of edge ε computes t .
- $\mathbf{TRANSP}_\varepsilon(t)$: instruction ι of edge ε does not modify an operand of t .

The Equation System:

$$\mathbf{Avail}_n = \begin{cases} false & \text{if } n = s \\ \prod_{m \in \text{pred}(n)} (\mathbf{Avail}_m + \mathbf{COMP}_{(m,n)}) \cdot \mathbf{TRANSP}_{(m,n)} & \text{otherwise} \end{cases}$$

Vorschau auf die weiteren Vorlesungstermine...

Achtung: Abweichende Zeit am 23.01.2007!

- Di, 23.01.2007, Vorlesung von 16:00 Uhr bis 17:00 Uhr, Bibliothek E185/1
- Di, 30.01.2007, Vorlesung von 17:45 Uhr bis 19:15 Uhr, Bibliothek E185/1