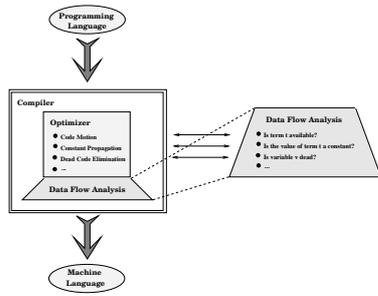
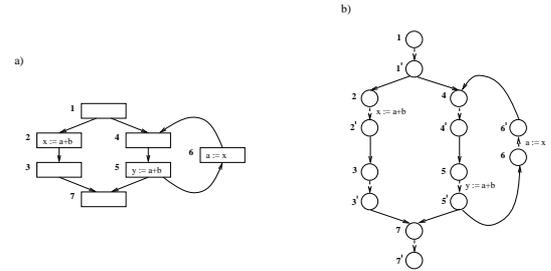


## Erinnerung und Motivation (1)



## Erinnerung und Motivation (2)

Knoten- vs. kantenbenannte Flussgraphen  
(hier mit Einzelinstruktionsbenennung)



## Erinnerung und Motivation (3)

Wir unterscheiden...

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

Datenflussanalyse.

## Erinnerung und Motivation (4)

Ingredienzien *intraprozeduraler* Datenflussanalyse:

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

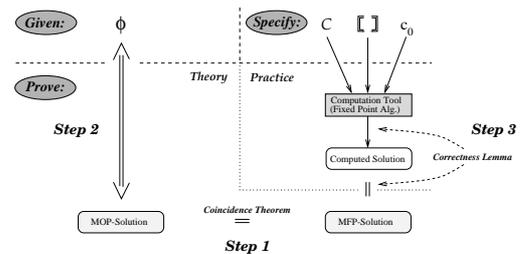
## Erinnerung und Motivation (5)

Hauptresultate:

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

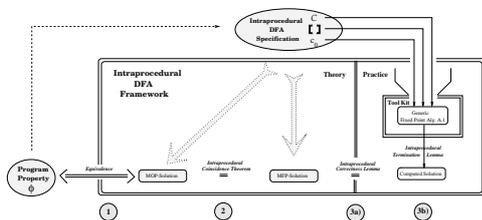
## Intraprozedurale Datenflussanalyse (1)

...im Überblick:



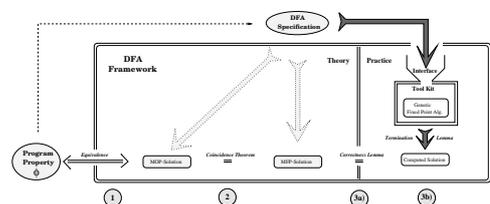
## Intraprozedurale Datenflussanalyse (2)

...im Überblick:



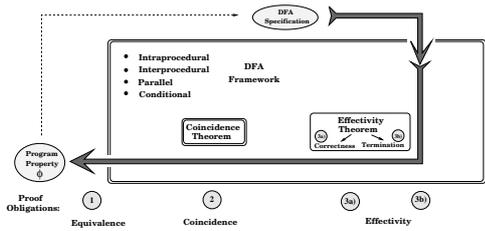
## DFA-Frameworks / DFA-Toolkits (1)

...das allgemeine Muster:



## DFA-Frameworks / DFA-Toolkits (2)

...das allgemeine Muster stärker abstrahiert:



## Ausblick

...auf

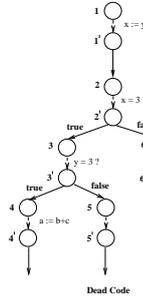
- konditionale,
- interprozedurale,
- parallele Datenflussanalyse

in...

- Rahmen- und
- Toolkit-Sicht.

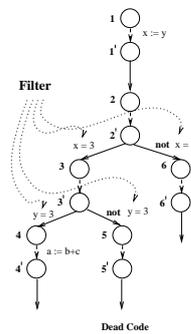
## Konditionale Datenflussanalyse (1)

Ein Anwendungsbeispiel zur Motivation...



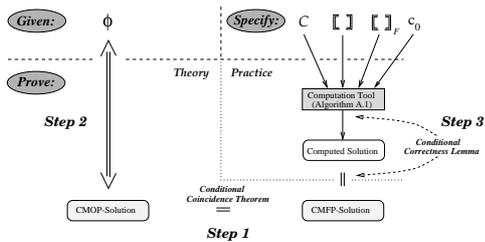
## Konditionale Datenflussanalyse (2)

Technische Umsetzung: Mittels Filterfunktionen...



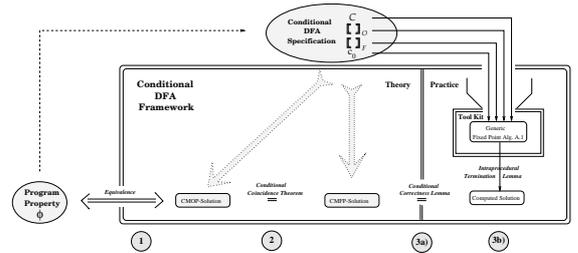
## Konditionale Datenflussanalyse (3)

Der konditionale DFA-Rahmen im Überblick:



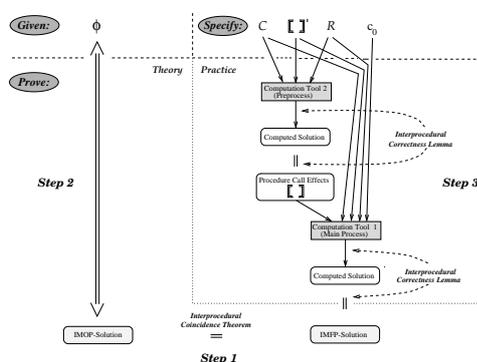
## Konditionale Datenflussanalyse (4)

Der konditionale DFA-Rahmen in der Toolkit-Sicht:



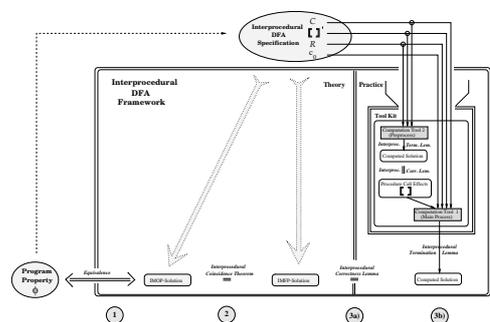
## Interprozedurale Datenflussanalyse (1)

Der interprozedurale DFA-Rahmen im Überblick:



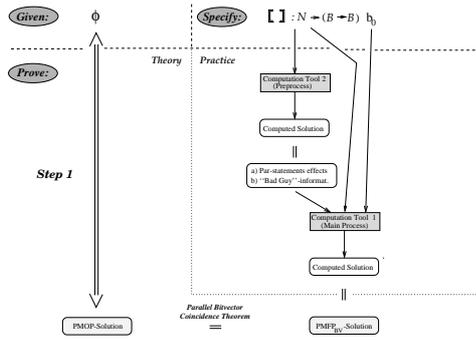
## Interprozedurale Datenflussanalyse (2)

Der interprozedurale DFA-Rahmen in der Toolkit-Sicht:



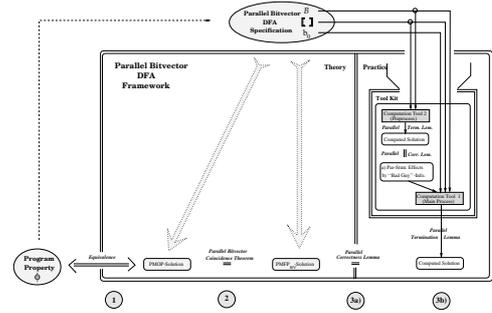
## Parallele Datenflussanalyse (1)

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:**

$$\forall c_s \in \mathcal{C} \forall n \in N. MOP_{(\llbracket \cdot \rrbracket_{l,c_s})}(n) =_{df} \bigcap \{ \llbracket p \rrbracket_{l,c_s} \mid p \in P_G[s, n] \}$$

## MaxFP -Ansatz (Einzelinstruktionen)

... für kantenbenannte Einzelinstruktionsgraphen:

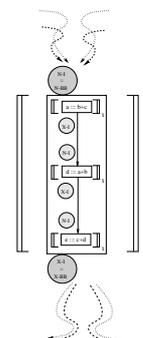
Die **MaxFP-Lösung:**

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

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

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

## Hierarchisierung durch Basisblöcke



## MOP -Ansatz (Basisblöcke) (1)

... für knotenbenannte Basisblockgraphen:

### The MOP-Solution: (Basisblockebene)

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

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

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

## MOP -Ansatz (Basisblöcke) (2)

... für knotenbenannte Basisblockgraphen:

### The MOP-Solution: (Instruktionsebene)

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

mit

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

$$X-MOP_{(\llbracket \cdot \rrbracket_{\iota, c_s})}(n) =_{df} \llbracket p \rrbracket_{\iota}(N-MOP_{(\llbracket \cdot \rrbracket_{\iota, c_s})}(\text{block}(n))) \quad (p \text{ Präfix von start(block(n)) bis (einschl.) n})$$

## MaxFP -Ansatz (Basisblöcke) (1)

... für knotenbenannte Basisblockgraphen:

### Die MaxFP-Lösung: (Basisblockebene)

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

mit

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

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

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

$$\text{post}(n) = \llbracket n \rrbracket_{\beta}(\text{pre}(n))$$

## MaxFP -Ansatz (Basisblöcke) (2)

### Die MaxFP-Lösung: (Instruktionsebene)

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

mit

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

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

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

$$\text{post}(n) = \llbracket n \rrbracket_{\iota}(\text{pre}(n))$$

## Bsp.: Verfügbarkeit von Ausdrücken (1)

### Verfügbarkeit für knotenbenannte BB-Graphen:

#### Phase I: Die Basisblockebene

Lokale Prädikate: (assoziiert mit BB-Knoten)

- $BB-XCOMP_{\beta}(t)$ :  $\beta$  contains an instruction  $\iota$  computing  $t$ , and neither  $\iota$  nor any instruction of  $\beta$  following  $\iota$  modifies an operand of  $t$ .
- $BB-TRANSP_{\beta}(t)$ :  $\beta$  contains no instruction modifying an operand of  $t$ .

## Bsp.: Verfügbarkeit von Ausdrücken (2)

Das Gleichungssystem von Phase I:

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

$$BB-X-AVAIL_{\beta} = BB-N-AVAIL_{\beta} \cdot BB-TRANSP_{\beta} + BB-XCOMP_{\beta}$$

## Bsp.: Verfügbarkeit von Ausdrücken (3)

#### Phase II: Die Instruktionsebene

Lokale Prädikate: (assoziiert mit EI-Knoten)

- $COMP_{\iota}(t)$ :  $\iota$  computes  $t$ .
- $TRANSP_{\iota}(t)$ :  $\iota$  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:

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

$$X-AVAIL_{\iota} = \begin{cases} BB-X-AVAIL_{\text{block}(\iota)} & \text{if } \iota = \text{end}(\text{block}(\iota)) \\ (N-AVAIL_{\iota} + COMP_{\iota}) \cdot TRANSP_{\iota} & \text{otherwise} \end{cases}$$

## Bsp.: Verfügbarkeit von Ausdrücken (4)

### Verfügbarkeit für knotenbenannte EI-Graphen:

Lokale Prädikate: (assoziiert mit Knoten)

- $COMP_{\iota}(t)$ :  $\iota$  computes  $t$ .
- $TRANSP_{\iota}(t)$ :  $\iota$  does not modify an operand of  $t$ .

Das Gleichungssystem:

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

$$X-AVAIL_{\iota} = (N-AVAIL_{\iota} + COMP_{\iota}) \cdot TRANSP_{\iota}$$

---

## Bsp.: Verfügbarkeit von Ausdrücken (5)

### Verfügbarkeit für kantenbenannte EI-Graphen:

Lokale Prädikate: (assoziiert mit EI-Kanten)

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

The Equation System:

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

---

## 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