Математические модели последовательных вычислений

mk.cs.msu.ru → Лекционные курсы
→ Математические модели последовательных вычислений

Блок 14

Стандартные схемы программ

Лектор:
Подымов Владислав Васильевич
E-mail:
valdus@yandex.ru

ВМК МГУ, 2022/2023, весенний семестр

Будем считать заданными:

1. Счётное множество переменных Var
2. Сигнатуру логики предикатов $\langle \mathrm{Const}, \mathrm{Func}, \mathrm{Pred} \rangle$, состоящую из
   - множества констант Const,
   - множества функциональных символов Func с сопоставленными им местностями (числами из $\mathbb{N}$)
     - (функциональный символ $\mathbf{f}$ местности n будет изображаться так: $\mathbf{f}^{(n)}$) и
   - множества предикатных символов Pred с сопоставленными им местностями
     - (предикатный символ P местности n будет изображаться так: $\mathrm{P}^{(n)}$)

Синтаксис термов логики предикатов (заданной сигнатуры) задаётся так:

$$t \ ::= \ \mathbf{c} \ | \ \mathrm{x} \ | \ \mathbf{f}^{(n)}(t_1, \ldots, t_n),$$

где $t, t_1, \ldots, t_n$ — термы, $\mathbf{c} \in \mathrm{Const}$, $\mathrm{x} \in \mathrm{Var}$ и $\mathbf{f}^{(n)} \in \mathrm{Func}$

Term — так будем обозначать множество всех термов

Атомарная формула (заданной сигнатуры) — это запись вида $\mathrm{P}^{(n)}(t_1, \ldots, t_n)$, где $\mathrm{P}^{(n)} \in \mathrm{Pred}$ и $t_1, \ldots, t_n$ — термы

Atom — так будем обозначать множество всех атомарных формул

Присваивание — это запись вида «`x := t`», где $x \in \mathrm{Var}$ и $t \in \mathrm{Term}$

Тест (или, по-другому, — условие) — это произвольный атом

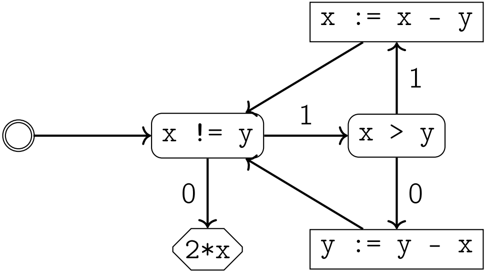
Стандартная схема программ отличается от схемы Ляпунова-Янова следующим:

1. В схеме может быть много выходов
2. Каждый выход помечен набором термов: $t_1, \dots, t_n$
   - (этот набор можно расценивать как результат, возвращаемый программой)
3. Каждый преобразователь помечен не операторным символом, а присваиванием
4. Каждый распознаватель помечен не логическим символом, а тестом

Пример

```
while (x != y)
  if (x > y)
    x = x - y;
  else
    y = y - x;
return 2*x
```

Этой программе соответствует такая стандартная схема:



Смысл основных элементов (примитивов) стандартной схемы задаётся интерпретацией логики предикатов, то есть системой $\mathcal{I} = \langle \mathrm{D}, \overline{\mathrm{Const}}, \overline{\mathrm{Func}}, \overline{\mathrm{Pred}} \rangle$, где:

- D — область интерпретации, или, по-другому, множество предметов
- $\overline{\mathrm{Const}} : \mathrm{Const} \to \mathrm{D}$ — оценка констант
  - $\overline{\mathbf{c}} = \overline{\mathrm{Const}}(\mathbf{c}) \in \mathrm{D}$
- $\overline{\mathrm{Func}} : \mathrm{Func} \to \bigcup_{n=1}^{\infty} (\mathrm{D}^n \to \mathrm{D})$ — оценка функциональных символов
  - $\overline{\mathbf{f}^{(n)}} = \overline{\mathrm{Func}}(\mathbf{f}^{(n)}) : \mathrm{D}^n \to \mathrm{D}$
- $\overline{\mathrm{Pred}} : \mathrm{Pred} \to \bigcup_{n=1}^{\infty} (\mathrm{D}^n \to \{0, 1\})$ — оценка предикатных символов
  - $\overline{\mathrm{P}^{(n)}} = \overline{\mathrm{Pred}}(\mathrm{P}^{(n)}) : \mathrm{D}^n \to \{0, 1\}$

Оценкой переменных (для заданной интерпретации $\mathcal{I}$) называется отображение вида $\xi : \mathrm{Var} \to \mathrm{D}$

Оценку переменных также будем называть состоянием данных схемы

$[x_1/d_1, \ldots, x_n/d_n]$ — так будем обозначать состояние данных ξ, такое что $\xi(x_1) = d_1, \ldots, \xi(x_n) = d_n$ и значения остальных переменных неважны

$\Xi_{\mathcal{I}}$ — так будем обозначать множество всех состояний данных для интерпретации $\mathcal{I}$

Значения термов t и атомарных формул A в интерпретации $\mathcal{I}$ на оценке ξ ($t[\xi]_{\mathcal{I}}$, $A[\xi]_{\mathcal{I}}$) задаются обычным естественным индуктивным образом:

- Если $x \in \mathrm{Var}$, то $x[\xi]_{\mathcal{I}} = \xi(x)$
- Если $\mathbf{c} \in \mathrm{Const}$, то $\mathbf{c}[\xi]_{\mathcal{I}} = \overline{\mathbf{c}}$
- Если $\mathbf{f}^{(n)} \in \mathrm{Func}$, то $\mathbf{f}(t_1, \ldots, t_n)[\xi]_{\mathcal{I}} = \overline{\mathbf{f}}(t_1[\xi]_{\mathcal{I}}, \ldots, t_n[\xi]_{\mathcal{I}})$
- Если $\mathrm{P}(t_1, \ldots, t_n) \in \mathrm{Atom}$, то $\mathrm{P}(t_1, \ldots, t_n)[\xi]_{\mathcal{I}} = \overline{\mathrm{P}}(t_1[\xi]_{\mathcal{I}}, \ldots, t_n[\xi]_{\mathcal{I}})$

Состояние вычисления стандартной схемы π в интерпретации $\mathcal{I}$ — это пара (v, ξ), где v — вершина схемы π и $\xi \in \Xi_{\mathcal{I}}$

Для состояния данных ξ, переменной x и предмета d записью $\xi[x \leftarrow d]$ обозначим состояние данных, получающееся из ξ присваиванием значения d в переменную x:

$$\xi[x \leftarrow d](y) = \begin{cases} d, & \text{если } y = x; \\ \xi(y) & \text{иначе} \end{cases}$$

Вычисление стандартной схемы π в интерпретации $\mathcal{I}$ на состоянии данных ξ ($comp(\pi, \mathcal{I}, \xi)$) — это (конечная или бесконечная) последовательность состояний вычисления

$$(v_1, \xi_1), (v_2, \xi_2), (v_3, \xi_3), \dots,$$

устроенная так:

1. $v_1 = \circledcirc$, $\xi_1 = \xi$, $v_1 \to v_2$, $\xi_2 = \xi$
2. Если $v_i = \boxed{\texttt{x := t}}$, то $v_i \to v_{i+1}$ и $\xi_{i+1} = \xi_i[x \leftarrow t[\xi_i]_{\mathcal{I}}]$
3. Если $v_i = \boxed{\mathrm{P}(t_1, \dots, t_n)}$, то $v_i \xrightarrow{\mathrm{P}(t_1,\dots,t_n)[\xi_i]_{\mathcal{I}}} v_{i+1}$
4. Если $v_i = \langle t_1, \dots, t_n \rangle$, то (v_i, ξ_i) — последний элемент последовательности

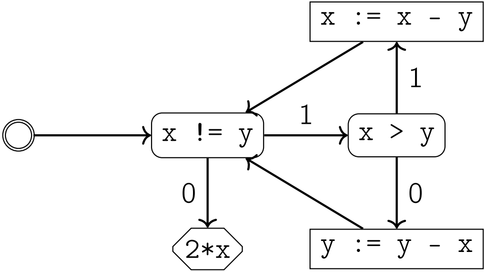
Результат $val(\pi, \mathcal{I}, \xi)$ вычисления стандартной схемы π в интерпретации $\mathcal{I}$ на состоянии данных ξ определяется так:

1. Если вычисление $comp(\pi, \mathcal{I}, \xi)$ конечно и оканчивается парой $(\boxed{t_1, \dots, t_n}, \xi')$, то $val(\pi, \mathcal{I}, \xi) = (t_1[\xi]_{\mathcal{I}}, \dots, t_n[\xi]_{\mathcal{I}})$

2. Если вычисление $comp(\pi, \mathcal{I}, \xi)$ бесконечно, то $val(\pi, \mathcal{I}, \xi) = \bot$, где $\bot$ не является набором термов

Семантика стандартной схемы π в заданной интерпретации $\mathcal{I}$ с предметной областью D описывается отображением $\overline{\pi}_{\mathcal{I}} : \Xi_{\mathcal{I}} \to (\mathrm{D}^* \cup \{\bot\})$, таким что $\overline{\pi}_{\mathcal{I}}(\xi) = val(\pi, \mathcal{I}, \xi)$

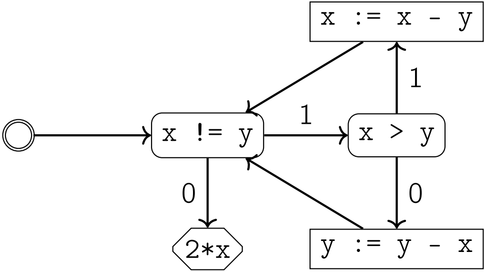
Пример вычисления стандартной схемы программ



Рассмотрим «естественную» арифметическую интерпретацию $\mathcal{I}_{ar}$:

- Предметы — это целые числа
- Символы `!=`, `>`, `-`, `*` и `2` оцениваются соответственно как отношения неравенства и сравнения (строго больше) чисел, операции разности и умножения чисел и константа 2

Пример вычисления стандартной схемы программ



Вычисление этой схемы в $\mathcal{I}_{ar}$ на оценке $[x/6, y/4]$ устроено так:

$(\circledcirc, [x/6, y/4])$, $(\boxed{\texttt{x != y}}, [x/6, y/4])$, $(\boxed{\texttt{x > y}}, [x/6, y/4])$,
$(\boxed{\texttt{x := x - y}}, [x/6, y/4])$, $(\boxed{\texttt{x != y}}, [x/2, y/4])$, $(\boxed{\texttt{x > y}}, [x/2, y/4])$,
$(\boxed{\texttt{y := y - x}}, [x/2, y/4])$, $(\boxed{\texttt{x != y}}, [x/2, y/2])$, $(\boxed{\texttt{2*x}}, [x/2, y/2])$

Результат этого вычисления — (4)