

2 Апрель 2010

Инсерционное моделирование 1

Лекция 7

Стратегии

Системы переписывающих правил

<rewriting system> ::= rs(<list of variables separated by ",">)

(<list of rules separated by "," >)

<rule> ::= <simple rule> | <conditional rule>

<simple rule> ::=

<algebraic expression> = <algebraic expression>

<conditional rule> ::= <condition> -> <simple rule>

<variable> ::= <identifier>

Базовые стратегии переписывания

applr: рекурсивная стратегия

applis: итеративная стратегия

2-algebra of logic

cnf: cnf_dn, cnf_up, стратегия can_ord
dnf: dnf_dn, dnf_up, стратегия can_ord
eliminate, стратегия ntb
is_id, I1, стратегия ntb2

strat.ap

Все стратегии
Переписывающая машина

Доказательство корректности алгебраических программ

```
ntb(x,eliminate);  
can_ord(x,cnf_dn,cnf_up);  
can_ord(x,dnf_dn,dnf_up);  
is_id x
```

Индукция

Фундированное частично-упорядоченное множество

Каждое подмножество имеет минимальный элемент

Вполне-упорядоченное: линейно-упорядоченное фундированное

Принцип индукции

$$\forall x (\forall (y < x) P (y) \rightarrow P (x)) \rightarrow \forall x P (x)$$

Доказательство от противного

Рассмотрим $\{x \mid \neg P (x)\}$ выберем минимальный элемент

...

Корректность программы

$\text{ntb}(x, \text{eliminate})$

$\text{is_prop } x \Rightarrow [\text{ntb}(x, \text{eliminate})] \text{ is_nnf } x$

$w(x) = (m1, m2, m3)$

$m1 = \text{количество} \Leftrightarrow$

$m2 = \text{количество} \rightarrow$

$m3 = \text{максимальная}$
 высота отрицания

```
ntb:=proc(t,R)loc(s,i)(
```

```
  a:=w(t);
```

```
  appls(t,R);
```

```
  assertion:(w(t)<a)&...;
```

```
  forall(s=arg(t,i),
```

```
    ntb(s,R)
```

```
  );
```

```
  assertion: forall(s=(arg t,i) is_nnf s);
```

```
  t:=can(t)
```

```
);
```

Forall x (is_prop x=u & w(x)=a =>

[appls(x,R)]

w(x)<a & (is_conj x | / is_disj x | / is_atom x)

is_prop x, x \Leftrightarrow u)

Максимальная высота отрицания

$$h(\sim(x|/y)) = \max(h(\sim(x)), h(\sim(y)))$$

$$h(\sim(x\&y)) = \max(h(\sim(x)), h(\sim(y)))$$

$$h(\sim(x|<=>y)) = \max(h(\sim(x))+1, h(\sim(y))+1)$$

$$h(\sim(x->y)) = \max(h(x)+1, h(\sim(y))+1)$$

$$h(\sim(\sim(x))) = h(x)$$

$$h(x) = 0, x \text{ -- литерал}$$

```
can_ord:=proc(t,R1,R2)loc(s,i)(  
  t:=can(t);  
  appls(t,R1);  
  forall(s=arg(t,i),  
    can_ord(s,R1,R2)  
  );  
  can_up(t,R2)  
);
```

```
can_up:=proc(t,R)loc(s,i)(  
  appls(t,R);  
  while(yes,  
    forall(s=arg(t,i),  
      can_up(s,R)  
    );  
    appls(t,R)  
  );  
  t:=mrg(can(t))  
);
```