

Курсовая работа

Улучшенный алгоритм оптимизации
совместного сопоставления с
образцом в компиляторе Простого
Рефала.

Савельев П.А.

Цель работы

Необходимо улучшить алгоритм оптимизации совместного сопоставления с образцом в компиляторе Простого Рефала.

Пример

```
F {  
  (E.State)(#X e.ValX) e.Items  
    = <F  
      (<Change e.State #X e.ValX>  
      e.Items  
      >;  
  
  (E.State)(#Y e.ValY) e.Items  
    = <F  
      (<Update e.State #X (e.ValY)>  
      e.Items  
      >;  
  
  (E.State)/* пусто */ = (e.State);  
}
```

Обобщение имеет вид: $(e.1) e.2 = \dots;$

Пример

```
F {  
  (E.State)(#X e.ValX) e.Items  
    = <F  
      (<Change e.State #X e.ValX>  
      e.Items  
      >;  
  
  (E.State)(#Y e.ValY) e.Items  
    = <F  
      (<Update e.State #X (e.ValY)>  
      e.Items  
      >;  
  
  (E.State)/* пусто */ = (e.State);  
}
```

Обобщение имеет вид: (e.1)(s.21 e.22) e.23 = ...;

Пример

```
F {  
  (E.State)(#X e.ValX) e.Items  
  =<F  
    (<Change e.State #X e.ValX>)  
    e.Items  
  >:  
  
  (E.State)(#Y e.ValY) e.Items  
  =<F  
    (<Change e.State #X e.ValY>)  
    e.Items  
  >:  
  
  (E.State)/* пусто */ = (e.State);  
}
```

Обобщение имеет вид: (e.1) = ...;

Улучшенный алгоритм

Дано:

- Набор предложений S
- Глобальное сложнейшее обобщение S
- Набор подстановок в $\Gamma CO(S)$ для каждого предложения

Обобщаем подстановки

Общая часть

: (e.1) e.2

Необобщенные подстановки

: e.1 \rightarrow e.State, e.2 \rightarrow (#X e.ValX) e.Items

: e.1 \rightarrow e.State, e.2 \rightarrow (#Y e.ValY) e.Items

: e.1 \rightarrow e.State, e.2 \rightarrow ε

Обобщенные подстановки

: (e.1)(e.2) , где e.2 $\neq \varepsilon$

: (e.1)(e.2) , где e.2 $\neq \varepsilon$

: (e.1)()

Находим количество предложений в группе

Обобщенные подстановки

: $(e.1)(e.2)$, где $e.2 \neq \varepsilon$

: $(e.1)(e.2)$, где $e.2 \neq \varepsilon$

: $(e.1)()$

Количество предложений в первой группе: 2

Находим разбиение на группы

Количество предложений в первой группе: 2

Предложения первой группы:

`: (E.State)(#X e.ValX) e.Items ()`

`: (E.State)(#Y e.ValY) e.Items ()`

Предложения второй группы:

`: (E.State) ()`

Находим ГСО для каждой группы

Общая часть всех предложений: $(e.1) e.2$

Общая часть первой группы: $(e.1)(s.21 e.22) e.23$

Общая часть второй группы: $(e.1)$

Предложения первой группы:

$: (E.State)(\#X e.ValX) e.Items ()$

$: (E.State)(\#Y e.ValY) e.Items ()$

Предложения второй группы:

$: (E.State) ()$

Формируем новую подстановку

Общая часть всех предложений: $(e.1) e.2$

Общая часть первой группы: $(e.1)(s.21 e.22) e.23$

Общая часть второй группы: $(e.1)$

Подстановки в первой группе:

$: e.2 \rightarrow (s.21 e.22) e.23$

$s.21 \rightarrow \#X, e.22 \rightarrow e.ValX, e.23 \rightarrow e.Items$

$: e.2 \rightarrow (s.21 e.22) e.23$

$s.21 \rightarrow \#Y, e.22 \rightarrow e.ValY, e.23 \rightarrow e.Items$

Подстановки во второй группе:

$e.2 \rightarrow \varepsilon$

Окончательный вид подстановок

Общая часть всех предложений: $(e.1) e.2$

Подстановки первой группы:

$: e.1 \rightarrow e.State, e.2 \rightarrow (s.21 e.22) e.23,$
 $s.21 \rightarrow \#X, e.22 \rightarrow e.ValX, e.23 \rightarrow e.Items$

$: e.1 \rightarrow e.State, e.2 \rightarrow (s.21 e.22) e.23$
 $s.21 \rightarrow \#Y, e.22 \rightarrow e.ValY, e.23 \rightarrow e.Items$

Подстановки второй группы:

$e.1 \rightarrow e.State, e.2 \rightarrow \varepsilon$

Требуется реализовать

- Переименование переменных
- Подстановку команд, сгенерированных для каждой группы
- Формирование и передачу вершины контекста для каждой группы

Вид функции до улучшения алгоритма

Функция:

```
Общие команды всех предложений
Защищенный блок {
    Команды первого предложения
}
```

...

```
Защищенный блок {
    Команды k-го предложения
}
```

...

Команды N-го предложения

Вид функции после улучшения алгоритма

Функция:

```
Общие команды всех предложений
Защищенный блок {
    Общие команды первой группы
    Защищенный блок {
        Команды первого предложения
    }
    ...
    Команды k-го предложения
}
Общие команды второй группы
Защищенный блок {
    Команды (k+1)-го предложения
}
...
Команды N-го предложения
```

Заключение

Реализована часть алгоритма, обобщающая подстановки, сформированные после вычисления глобального сложнейшего обобщения; реализовано разбиение исходных предложений на группы для последующего обобщения, а также дополнение подстановок подстановкой, содержащей общую левую часть для каждой из групп.

Дальнейшее развитие

- Изменить генерацию кода, чтобы объединять списки команд сопоставления выражения с обобщением для каждой группы с общим списком команд
- Добавить рекурсивное разбиение групп предложений
- Поработать над производительностью и чистотой кода