

# 指称语义和小步语义的等价性

语义虽然有多种，但是描述的事情可以是一样的。这里展示对指称语义和小步语义等价性的证明思路。

## 定理

我们要证明的是这样一件事情：

$$(st_1, st_2) \in \text{ceval}(c) \iff (c, st_1) \rightarrow^* (\text{skip}, st_2)$$

其中  $\text{ceval}(c)$  表示程序  $c$  的指称。

## 从左到右

先证明  $(st_1, st_2) \in \text{ceval}(c) \implies (c, st_1) \rightarrow^* (\text{skip}, st_2)$ 。

直接对  $c$  做结构归纳。重点在于先对  $\text{aexp}$ ,  $\text{bexp}$  先做类似的归纳，即最终两者可以求出某个具体的值来。这是为了辅助赋值和  $\text{if}$ 、 $\text{while}$  语句的证明。

然后就容易了，对于原子语句可以直接证明，对于复合语句我们由归纳假设可以知道每一个部分的结果，然后把结果利用多步的传递性质套进去即可。

## 从右到左

再证明  $(c, st_1) \rightarrow^* (\text{skip}, st_2) \implies (st_1, st_2) \in \text{ceval}(c)$ 。

还是对  $c$  做结构归纳。关键的在于串联形式  $c_1; c_2$  的证明。

按照  $c_1; c_2$  的指称定义，我们要证明的是

$$\exists st, (c_1; c_2, st_1) \rightarrow^* (\text{skip}; c_2, st) \rightarrow (c_2, st) \rightarrow^* (\text{skip}, st_2)$$

如果我们能够直接说明  $\exists st, (c_1, st_1) \rightarrow^* (\text{skip}, st)$  那显然是再方便不过。但是**由于有不可停止的程序存在，这条性质并不成立**。

为此，对多步关系做 1-n 归纳，即从最后面开始向前接。在归纳的时候就可以通过反演确定具体是哪一种情况。

对于其他的也是类似，用 1-n 归纳证明复合语句的多步关系一定蕴含子语句的多步关系，最后把这些都拼起来即可。

## 再一次从右到左

上面的证明我们本来想要用一个看起来非常美好的性质： $\exists st, (c_1, st_1) \rightarrow^* (\text{skip}, st)$ 。但是用不得。

实际上，如果把这一性质减弱一点，就可以得到另一个比较类似但仍然很有用的性质。这条新的可证明的性质描述的是**小步语义对终止状态的保持 (Preservation)**，即

$$(c, st_1) \rightarrow (c', st'_1) \rightarrow^* (\text{skip}, st_2) \implies (st'_1, st_2) \in \text{ceval}(c) \implies (st_1, st_2) \in \text{ceval}(c)$$

这就绕过了停机性，也就是先说明结果一定存在然后向前推。

对这个的证明需要对小步关系进行归纳。证明完这个之后就只需要对  $(c, st_1) \rightarrow^* (\text{skip}, st_2)$  做 1-n 归纳即可。