带参缓存一致性协议正确性证明系统paraVerifier设计

段凯强

# 语言能力设计

作为paraVerifier的输入语言，必须能够处理缓存一致性协议的所有情形（最基本的情形不要求带参，因为带参问题可以通过其他方式解决）。在此基础上，要求从方便建模的角度加入更多的语言特性，如条件判断、函数、过程、局部变量等。

## 缓存一致性协议的描述

### 基本情形

缓存一致性协议相当于是描述了一个状态机，即其描述的系统可以用一系列的状态（变量集）和迁移（规则）表示。

* 状态：由一些全局变量构成，对所有全局变量的并行赋值构成一个状态；
* 对变量的赋值：将一个数值表达式的值赋给某变量；
* 并行赋值：对一系列的全局变量进行的赋值是并行的，没有先后关系；
* 迁移：当前状态满足迁移条件时，按照指定的方式对指定的全局变量并行赋值而进入下一个状态；
* 迁移条件：根据某些全局变量得到的满足性判断，可以用条件公式表示；
* 条件公式：一系列布尔表达式的复合；
* 布尔表达式：是布尔值，或者对数值表达式的判等运算；
* 数值表达式：对全局变量的数值运算；
* 初始状态集；
* 可达状态集：如果从一个状态到任意一个初始状态之间有路，则这个状态就在可达状态集中。

而对这样的系统，需要验证的性质包括安全性和活性（可选）。

* 安全性：在每个可达到的状态都满足的条件公式，用不变式表示；
* 活性：在某个达到的状态必须满足的条件公式，用断言表示。

### 带参情形

带参能力决定了语言的泛化能力。支持的带参能力多，则使用该语言建模所需要的人的工作量就越少。

* 全局变量带参：某些全局变量可以参数化，以模拟分布式协议；
* 规则带参：规则可以参数化，以描述规则集（注意带参规则很可能需要引入局部变量）；
* 性质带参：模型验证的性质可以参数化，以描述性质集；

## 语言表达能力递进描述

下面递进描述语言的表达能力，每一级都在低一级的基础上对语言的表达能力进行扩展，经过一定的算法处理，可以将满足高级特性的语言转化为满足低一级特性的语言。

### 基本能力（A层）

语言定名为Paramecium。

* 类型定义
* 常量
* 全局变量定义
* 基本类型，一般为枚举以及整形
* 数值表达式，计算结果为数值，可以是数值计算或者条件表达式
* 布尔表达式，布尔值或者对数值的判等
* 条件公式，包括与、或、非、蕴含
* 赋值语句，包括由赋值语句组成的并行赋值语句
* 规则：由前置条件Guard和后置赋值Statements表示

### 带参能力（B层）

语言定名为Loach。

* 全局变量带参
* 规则带参
* 性质带参
* 条件公式带参
* 赋值语句带参

### 过程、函数支持（C层）

TODO：局部变量、forall/exists、for语句、分支语句

语言定名为Frog。

* 过程：可以理解为带参赋值；过程调用
* 函数：有临时变量定义和返回值的过程；函数调用

### 其他高级特征支持（D层）

语言定名为Gecko。

* 复杂类型定义，即Record
* 扩充循环：while
* 扩充分支：switch
* 别名：alias
* 断言
* 异常

### 高级语法支持（E层）

语言定名为Eagle。

类高级语言的语法，而不是使用元语言的语法。

## 各表达能力层次间的翻译

### B层到A层

* 局部变量：翻译为表达式
* 带参全局变量：建立一系列新变量，根据带参变量名称和参数取值范围命名
* 带参规则：实例化为多个规则
* 带参性质：实例化为多个性质
* 条件公式扩展：映射为多个条件公式
* 赋值扩展for：映射为并行的一系列赋值；要在分支翻译之前先进行
* 分支形式1/2：分裂为两个规则

### C层到B层

* 过程：实例化后替换过程调用
* 函数：过程部分实例化后置于函数调用前方，然后用返回值替换函数调用

### D层到C层

* 复杂类型定义：具有复杂类型的变量可以表示为一系列具有基本类型的变量的集合，从而转化为变量定义
* 扩充循环while：暂无法支持
* 扩充循环switch：多个（嵌套的）if语句
* 别名：翻译为实际的名称
* 断言：暂无法支持
* 异常：暂无法支持

### E层到D层

属于程序语言设计范畴，待定