**Inlining control blocks**

Inlining is an optimization that replaces a function [call site](https://en.wikipedia.org/wiki/Call_site) with the body of the called function. We are specifically doing this for control blocks.

Control block inlining is replacing the apply method with its called function apply method and adding supporting tables, actions from the called control block to the callee block respectively.

Why we inline control Blocks?

Calling/Using the methods of other controlBlock is limited by hardware. So, inlining helps in making each controlBlock independent of one another.

Overview :

The headers (and other metadata) in p4 parsers can be manipulated and transformed within control blocks.

A control block is declared with a name, parameters, optional type parameters, and a sequence of declarations of constants, variables, actions, tables, and other instantiations.

The body of a control block resembles a traditional imperative program. Within the body of a control block, match-action units can be invoked to perform data transformations. Match-action units are represented in P4 by constructs called tables. A table can be invoked by calling its apply method.

P4 also allows controls to invoke the services of other controls, similar to subroutines. To invoke the services of another control, we can directly call the apply method of other control block.

The following example shows a control invocation:

control Callee(inout IPv4 ipv4) {

     table t3 { ... }

table t4 { ... }

     apply {

                t3.apply();

     }

}

control Caller(inout Headers h) {

      table t1 { ... }

table t2 { ... }

      apply {

                 Callee.apply(h.ipv4);

      }

}

table t1

table t1

table t3

table t2

Table t4

t3.apply()

Equivalent result control

table t3

table t4

t3.apply()

Callee Control Block

table t1

Callee.apply()

table t2

Caller Control Block

Implementation:

1. Collect all the control blocks in a p4 program.
2. Inline each control block one after other(Caller ControlBlock). Inlining a control block means processing tables, actions, apply methods. Declare a list of actions, list of tables, list of applyStatements for a control block.
   1. Add all the local table declarations to the list of tables and all the actions to the list of actions.
   2. Then processing apply method is processing all the statements in that method.
      1. If the statement is not a call to other control block, then add the statement to the applyStatements list.
      2. If the statement is a call to other control block’s apply method(Callee controlBlock), then get the Callee control block from the collected controlBlocks list in 1st step and process it similarly i.e repeat the steps 2 until no call to other control is made.
      3. Accumulate all the Callee list of tables, actions, statements to the Caller controlBlocks tables, actions and statements.
   3. Form the control block context with the list of accumulated actions, methods, applyStatements.
   4. Replace the old controlBlock context with the new context in the p4 program.
3. Finally after inlining all the controlBlocks in a p4 program, all the controlBlocks become independent of each other.

Example: p4 program

control Callee(inout IPv4 ipv4) {

     table t3 { ... }

table t4 { ... }

     apply {

                t3.apply();

     }

}

control Caller(inout Headers h) {

      table t1 { ... }

table t2 { ... }

table t3 { … }

table t4 { … }

      apply {

t3.apply();

      }

}