

Schedule Code Generator for SubSystem

Given an affine system with subsystems/useEquations, this page shows how to specify the targetmapping for the system and generate the code.

Matrix Multiplication with subsystem

The following code is the alpha program for matrix matrix multiplication with dot-product subsystem.

```

affine matrix_product_SubSyst {N,K,M | N>0 && K>0 && M > 0} // Product
between a N*K matrix and a K*M matrix
input
  float A {i,k | 0<=i<N && 0<=k<K};
  float B {k,j | 0<=k<K && 0<=j<M};
output
  float C {i,j | 0<=i<N && 0<=j<M};
let
  use {iP,jP|0<=iP<N && 0<=jP<M} dot_product[K]
((pi,pj,k->pi,k)@A,(pi,pj,k->k,pj)@B) returns (C);
.

affine dot_product {N| N>0} // Product between 2 vector of size N
input
  float vect1 {i | 0<=i<N };
  float vect2 {i | 0<=i<N };
output
  float Res;
local
  float temp {i | 0<=i<N};
let
  temp[i] = case
    {i|i==0} : vect1[0] * vect2[0];
    {i |0<i<N} : temp[i-1] + vect1[i]*vect2[i];
  esac;
  Res[] = temp[N-1];
.

```

The program contains two systems. The dot_product system takes two vectors as inputs and computes the dot product of these two vectors. The matrix_product_SubSyst computes matrix $C=A*B$, the (ip,jp) th element for the answer matrix C is computed by calling the dot product subsystem, and the (ip) th row of A, and (jp) th column of B is passed as input to the subsystem call.

TargetMapping for the Matrix Multiplication Example

The schedule code generator treats every subsystem call (an instance of the subsystem) as an function call in C. In order to ensure the correctness of the code,

From: <https://www.cs.colostate.edu/AlphaZ/wiki/> - **AlphaZ**

Permanent link: https://www.cs.colostate.edu/AlphaZ/wiki/doku.php?id=schedule_code_generator_for_code_with_subsystem&rev=1404763678

Last update: **2014/07/07 14:07**

