Multidimensional Dataflow in Ptolemy

Chris Shaver  Dai Bui  Stavros Tripakis
University of California, Berkeley

Motivation
- Dataflow models processing arrays of data.
- ArrayOL formalism for multidimensional dataflow.
- Coarse and fine grain parallelization.
- SpearDE, developed by Thales.

Goals
- Develop the Pthales domain for multidimensional dataflow.
- Use the Ptolemy platform as a basis for research in multidimensional dataflow models in the context of heterogenous system semantics, code generation, scheduling techniques, and the semantics of time.
- Implement the semantics of SpearDE and ArrayOL in Pthales.
- Formulate semantics for static and dynamic models.
- Integrate the Pthales domain with other models of computation.
- Develop code generation and hardware mapping.
- Experiment with temporal semantics in Pthales.

Dynamic Pthales
- Embedding Pthales semantics into Process Networks.
- Expanding specification language to include expressions as parameters.
- Automatic generation of parameters from other specifications.
- Code generation for multicore platforms.
- Pipelined scheduling techniques.
- Hierarchical characterization of dynamic semantics:
  - How can different forms of dynamic multidimensional semantics be interrelated or included in one another?
  - What is the relationship between scenario-based and modal model approaches to dynamic semantics?

Future Work

Pthales Port Semantics

At each port, Pthales actors produce and consume patterns of data tiled within arrays. Ports in the model are given a set of array data parameters of the following form:

\[ \text{parameter} = \{a = x_a, b = x_b, \ldots\} \]

where \((a, b, \ldots)\) are names and \((x_a, x_b, \ldots)\) are associated data.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Form (x)</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>base (h)</td>
<td>(h) - base coordinate</td>
<td></td>
</tr>
<tr>
<td>pattern (p_i, h_i)</td>
<td>(p_i) - pattern dimension, (h_i) - pattern step size</td>
<td></td>
</tr>
<tr>
<td>tiling (l_i)</td>
<td>(l_i) - tiling step size</td>
<td></td>
</tr>
<tr>
<td>size (s_i)</td>
<td>(s_i) - array dimension</td>
<td></td>
</tr>
<tr>
<td>repetitions (r_i)</td>
<td>(r_i) - pattern repetitions</td>
<td></td>
</tr>
</tbody>
</table>

[if \(p_i\) is substituted for \(p_i, h_i\), \(h_i\) defaults to 1]

Dynamic Pthales

Modal Pthales
- Static Pthales extended with modal models.
- Parameter Dynamic Pthales allowing \(\vec{s}\) to vary with data and \(\vec{r}\) to be recomputed.

Parameter-Dynamic Pthales
- Parameters such as data sizes, tiling, pattern, repetitions, are dynamically changed.
- Parameter changes are propagated throughout models using header information included in data packets.

Computing Parameters

A parameter set \((\vec{b}, \vec{p}, \vec{h}, \vec{t}, \vec{s}, \vec{r})\) is overdetermined with the constraint that the repetitions must fill the array as much as possible. \(\vec{s}\) can be calculated from \(\vec{r}\):

\[ \vec{s} \subseteq \vec{b} + \{\vec{t}, \vec{r} - \vec{t}\} + \{\vec{p}, \vec{b} - \vec{t}\} + \vec{t} \]

where \(\{\vec{a}, \vec{b}\}\) denotes point-wise multiplication. If \(\vec{s}\) is given, \(\vec{r}\) can be calculated:

\[ \vec{r} \subseteq \{\vec{s} - \vec{b} - \vec{t} - \{\vec{p}, \vec{b} - \vec{t}\} - \vec{t}, \vec{r}' - \vec{t}\} \]

the greatest \(\vec{r}'\) satisfying this condition being:

\[ \vec{r}' = \{\vec{s} - \vec{b} - \vec{t} - \{\vec{p}, \vec{b} - \vec{t}\} - \vec{t}, \vec{r}' - \vec{t}\} \]