Software Support for Regular & Irregular Application in Parallel Computing
Da Li, Michela Becchi
University of Missouri - Columbia

Abstract
Today's applications can be divided into two basic categories: regular and irregular applications. The parallel computing community has acquired a deep understanding of the characteristics of regular applications and has proposed numerous abstractions, technologies and solutions to facilitate their parallelization. However, the research on irregular applications is still at an initial phase and the availability of programming models and tools targeting these applications is still limited.

My research focuses on programming models and system software support for regular and irregular algorithms on parallel architecture. More specifically, it can be summarized in the following aspects:

a) Parallelization of regular and irregular applications on GPU
b) Programming model and runtime design for regular and irregular applications

Equation-based Programming Model & Runtime for Matrix-based Regular Applications

Applications
Needleman-Wunsch

\[ M(i, j) = \max \left\{ \begin{array}{ll}
M(i-1, j-1) + S(x, y) \\
M(i-1, j) + G \\
M(i, j-1) + G
\end{array} \right. \]

Parallelization Pattern

Programming Model and Runtime

Integral Histogram

\[ H(x, y, b) = \sum_{x} \sum_{y} Q(i, j, b) \]

Adaptive Runtime for Graph Algorithms

Motivation
Application

• BFS (breadth-first search)
• SSSP (single source shortest path)

Dataset (real-world graphs)

<table>
<thead>
<tr>
<th>Network</th>
<th>#Nodes</th>
<th>#Edges</th>
<th>Node Outdegree</th>
<th>Size of working set</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO-road</td>
<td>835,400</td>
<td>5,661,800</td>
<td>4.3</td>
<td>553</td>
</tr>
<tr>
<td>Citeseer</td>
<td>3,317,053</td>
<td>1,438,843</td>
<td>4.3</td>
<td>2,180</td>
</tr>
<tr>
<td>Fil</td>
<td>3,317,053</td>
<td>1,438,843</td>
<td>4.3</td>
<td>2,180</td>
</tr>
<tr>
<td>Amazon</td>
<td>10,985,912</td>
<td>1,438,843</td>
<td>4.3</td>
<td>2,180</td>
</tr>
<tr>
<td>DBLP</td>
<td>4,5,6,7,8,9,10,11,12,13</td>
<td>64,100</td>
<td>200,000</td>
<td>16.0</td>
</tr>
</tbody>
</table>

Decision Space

Graph API

• Mapping decisions based on static and dynamic information
• Adaptive mapping at runtime
• Overhead's reduction by decreasing sampling rate
• Dynamic parallelism: Multiple mapping decisions for each execution

Implementation Space

Contact us: dix7t@umail.missouri.edu