Grid-Enabled Hydropad: 
a Scientific Application for Benchmarking 
GridRPC-Based Programming Systems

Michele Guidolin*  
Alexey Lastovetsky†

Heterogeneous Computing Laboratory  
School of Computer Science and Informatics  
University College Dublin (UCD)

* michele.guidolin@ucd.ie  
† alexey.lastovetsky@ucd.ie
Introduction

Grid Computing is widely used for scientific applications

- GridRPC is a standard API promoted by Open Grid Forum
- Several Grid middleware are GridRPC compliant

What are the potential performance of GridRPC based systems?
Grid Computing is widely used for scientific applications

* GridRPC is a standard API promoted by Open Grid Forum
* Several Grid middleware are GridRPC compliant

What are the potential performance of GridRPC based systems?

Different GridRPC systems are tested with artificially engineered applications

 TASKS WITH HIGH COMPUTATION
 TASKS WITH EXECUTION OF MANY PARALLEL TASKS
 TASKS WITH MINIMAL DATA COMMUNICATION BETWEEN TASKS

Results that they have an equally good performance
Grid Computing is widely used for scientific applications

* GridRPC is a standard API promoted by Open Grid Forum
* Several Grid middleware are GridRPC compliant

What are the potential performance of GridRPC based systems?

Different GridRPC systems are tested with artificially engineered applications

- tasks with high computation
- execution of many parallel tasks
- minimal data communication between tasks

Results that they have an equally good performance

We need a real-life, non Grid friendly, application as a benchmark!

- To show the eventual limits and benefits of a GridRPC middleware
- To show that Grid computing can be used to speed-up a wide range of applications
Hydropad

✓ An astrophysical application that simulates the clustering of galaxies from the big bang till present.

It is based on the Lambda-C spins model which assumes that the universe is composed of two matters:

✓ Baryonic matter (all bright objects)
✓ Dark matter (gravitational mass)

Hydropad is not an ideal application to get top performance in a Grid environment

✓ relatively low complexity of its tasks (log-linear at maximum)
✓ large amount of input and output data moved between tasks.
It consists of four parts

**Initialisation**
- **Gravitational** (FFT)
- **Dark Matter** (N-Body)
- **Baryonic Matter** (PPM)

**Characteristics**

- $C_{bm} \gg C_{dm}$
- $D_{bm} \ll D_{dm}$
We compare the three versions of Hydropad:

1. Local sequential version  
   (Comparison base)
2. GridSolve version
3. SmartGridSolve version  
   (Our own extensions of GridRPC model)
We compare the three versions of Hydropad:

1. Local sequential version  
   (Comparison base)
2. GridSolve version
3. SmartGridSolve version  
   (Our own extensions of GridRPC model)

We analyse how Grid-enabled versions of Hydropad can improve the performance

* Computation
* Communication
* Memory footprint
GridSolve vs. SmartGridSolve

GridSolve:

- Maps tasks individually
- Uses client-server/ star network data communication
GridSolve vs. SmartGridSolve

GridSolve:
- Maps tasks individually
- Uses client-server/ star network data communication

SmartGridSolve: a new middleware that extends the execution model of GridRPC to overcome its limitations
- Maps a group of tasks
- Uses fully connected network data communication
The hardware configuration:

- Client machine with 256MB of memory and 248MFlops
- Two heterogeneous servers with 1GB of memory 498 and 531 MFlops respectively
- The server-to-server link speed is 1GB/s
- Two client-to-server link setups: 100MB/s and 1GB/s

The initial parameters and data sizes for the problems:

<table>
<thead>
<tr>
<th>Problem ID</th>
<th>N_p</th>
<th>N_g</th>
<th>Data Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>120^3</td>
<td>60^3</td>
<td>73MB</td>
</tr>
<tr>
<td>P2</td>
<td>140^3</td>
<td>80^3</td>
<td>142MB</td>
</tr>
<tr>
<td>P3</td>
<td>160^3</td>
<td>80^3</td>
<td>176MB</td>
</tr>
<tr>
<td>P4</td>
<td>140^3</td>
<td>100^3</td>
<td>242MB</td>
</tr>
<tr>
<td>P5</td>
<td>160^3</td>
<td>100^3</td>
<td>270MB</td>
</tr>
<tr>
<td>P6</td>
<td>180^3</td>
<td>100^3</td>
<td>313MB</td>
</tr>
<tr>
<td>P7</td>
<td>200^3</td>
<td>100^3</td>
<td>340MB</td>
</tr>
<tr>
<td>P8</td>
<td>220^3</td>
<td>120^3</td>
<td>552MB</td>
</tr>
<tr>
<td>P9</td>
<td>240^3</td>
<td>120^3</td>
<td>624MB</td>
</tr>
</tbody>
</table>
## GridSolve Results Table

### (a) Local

<table>
<thead>
<tr>
<th>P.ID</th>
<th>Time Step</th>
<th>Paging</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>14.09s</td>
<td>No</td>
</tr>
<tr>
<td>P2</td>
<td>29.95s</td>
<td>No</td>
</tr>
<tr>
<td>P3</td>
<td>35.29s</td>
<td>No</td>
</tr>
<tr>
<td>P4</td>
<td>55.13s</td>
<td>Light</td>
</tr>
<tr>
<td>P5</td>
<td>61.63s</td>
<td>Light</td>
</tr>
<tr>
<td>P6</td>
<td>83.66s</td>
<td>Yes</td>
</tr>
<tr>
<td>P7</td>
<td>128.55s</td>
<td>Yes</td>
</tr>
<tr>
<td>P8</td>
<td>227.89s</td>
<td>Heavy</td>
</tr>
<tr>
<td>P9</td>
<td>280.07s</td>
<td>Heavy</td>
</tr>
</tbody>
</table>

### (b) GridSolve C - 1GB/s

<table>
<thead>
<tr>
<th>Time Step</th>
<th>Paging</th>
<th>(S_p) v Loc</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.20s</td>
<td>No</td>
<td>1.96</td>
</tr>
<tr>
<td>15.51s</td>
<td>No</td>
<td>1.93</td>
</tr>
<tr>
<td>16.48s</td>
<td>No</td>
<td>2.14</td>
</tr>
<tr>
<td>29.11s</td>
<td>No</td>
<td>2.14</td>
</tr>
<tr>
<td>29.07s</td>
<td>No</td>
<td>2.12</td>
</tr>
<tr>
<td>36.74s</td>
<td>Light</td>
<td>2.28</td>
</tr>
<tr>
<td>48.06s</td>
<td>Yes</td>
<td>2.67</td>
</tr>
<tr>
<td>77.91s</td>
<td>Heavy</td>
<td>2.92</td>
</tr>
<tr>
<td>91.75s</td>
<td>Heavy</td>
<td>3.06</td>
</tr>
</tbody>
</table>

### (c) GridSolve C - 100MB/s

<table>
<thead>
<tr>
<th>Time Step</th>
<th>Paging</th>
<th>(S_p) v Loc</th>
</tr>
</thead>
<tbody>
<tr>
<td>18.01s</td>
<td>No</td>
<td>0.78</td>
</tr>
<tr>
<td>35.02s</td>
<td>No</td>
<td>0.86</td>
</tr>
<tr>
<td>43.09s</td>
<td>No</td>
<td>0.82</td>
</tr>
<tr>
<td>55.66s</td>
<td>No</td>
<td>0.97</td>
</tr>
<tr>
<td>58.17s</td>
<td>No</td>
<td>1.06</td>
</tr>
<tr>
<td>72.50s</td>
<td>Light</td>
<td>1.15</td>
</tr>
<tr>
<td>80.05s</td>
<td>Yes</td>
<td>1.61</td>
</tr>
<tr>
<td>133.47s</td>
<td>Heavy</td>
<td>1.71</td>
</tr>
<tr>
<td>155.36s</td>
<td>Heavy</td>
<td>1.81</td>
</tr>
</tbody>
</table>

### Sources of speed-up:

- ✔️ Parallel tasks
- ✔️ Powerful servers
- ✔️ Less paging on the client machine
- ✗ Save the temporary data allocation on remote tasks
- ✗ Paging only during communication, **not computation**
Sources of speed-up:

- Better mapping (group of tasks)
- Minor paging on the client machine
- The direct server-to-server communication minimises the data on the client
- Paging only at beginning and at end of the group of tasks
- Better communication model
'SmartGridSolve Results Figure

- GridSolve C-100MB/s
- GridSolve C-1GB/s
- SmartGridSolve C-100MB/s
- SmartGridSolve C-1GB/s

GridSolve Paging

SmartGridSolve Paging

evolve time step (s)
data size (MB)
Conclusion

Grid-enabled Hydropad is a freely available application: hcl.ucd.ie

✔ It represents a good benchmark for GridRPC-based systems
  ❌ it exemplifies typical real-life scientific applications
  ❌ it push the limits of the GridRPC model
  ❌ highlights the strengths and exposes the weaknesses of a particular middleware implementation

The experiments show:

* GridSolve version of Hydropad
  - slower for smaller problem sizes
  - faster for larger problem sizes

* SmartGridSolve version of Hydropad
  - achieves better performance than the other two versions:
    * better mapping
    * better communication model
    * less paging