1. INTRODUCTION

The dynamic behavior of large modern parallel simulation codes can lead to imbalances in computational load among processors. In this thesis, I address how to evaluate load imbalance at runtime and make its correction affordable.

2. QUANTIFYING THE EFFECTIVENESS OF LOAD BALANCE ALGORITHMS

(a) Statistical load metrics do not shed light on how to correct the imbalance.
(b) To evaluate and correct the imbalance, we need to attribute load to the migratable tasks in the application.

Load/Process

<table>
<thead>
<tr>
<th>Process</th>
<th>idle</th>
<th>load</th>
<th>all</th>
<th>skew</th>
</tr>
</thead>
<tbody>
<tr>
<td>P0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>P1</td>
<td>25</td>
<td>25</td>
<td>25</td>
<td>0</td>
</tr>
<tr>
<td>P2</td>
<td>50</td>
<td>50</td>
<td>50</td>
<td>0</td>
</tr>
<tr>
<td>P3</td>
<td>75</td>
<td>75</td>
<td>75</td>
<td>0</td>
</tr>
<tr>
<td>P4</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>0</td>
</tr>
</tbody>
</table>

Need a model of:
- Weighted migratable tasks
- Relationships between tasks

3. EFFICIENT LOAD BALANCE ALGORITHM FOR N-BODY SIMULATIONS WITH NON-UNIFORM DENSITY

Tasks of Highly Variable Sizes are Difficult to Balance

<table>
<thead>
<tr>
<th>Task Size</th>
<th>Load Balancer Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Less Uniform</td>
<td>0.60%</td>
</tr>
<tr>
<td>More Uniform</td>
<td>0.80%</td>
</tr>
</tbody>
</table>

Our Method Uses Aggressive Adaptive Sampling to Define Uniform Tasks in Highly Non-Uniform Density

- Centroid
- Sampled Interactions Define Voronoi Cells
- Multi-Layer Voronoi Cells

4. LAZY LOAD BALANCING: OFFLOADING LOAD BALANCE COMPUTATION TO NON-APP. RESOURCES

Balance Asynchronously to the App.
- Run load balancer on a separate set of nodes in parallel to application computation
- Evaluate and decide how to correct imbalance without pausing the application
- Overlap LB method with application computation
- Decouple LB method partition size from app. partition size
- Effects of delayed decision?
- Impact of application drift?
- Impact of system scale?

Application and LB Algo scale differently — decouple resources
- LB algorithm performance varies with resources (i.e., graph partitioner)
- Communication overhead depends on resources reserved for LB algorithm

Run the load balance algorithm in a separate partition, asynchronously to the application

5. CONTRIBUTIONS

- Lazy load balancing framework that decouples and offloads the load balance computation to make it affordable at scale
- A model for selecting the right load balance algorithm for the job
- An accurate and fast method to balance N-body applications with highly non-uniform density