

EE482c Project Proposal

Improving Unstructured Mesh Application Performance on Stream Architectures

Nuwan Jayasena, Yangjin Oh, Hsiao Heng Lee,
Anand Ramalingam, Francois Labonte

May 9, 2002

1 Problem Definition

Unstructured mesh application (where each node has a different number of neighbors) do not map easily to stream architectures because:

1. Irregularity in the number of elements required to update each element
2. Elements require same nodes which need to be fetched from the memory to the SRF many times
3. Elements are duplicated in SRF
4. Some updates can be applied to each node (like gravity between 2 planets), the simple way is to compute the interaction twice simplifying the update.

2 Plan

We plan to address these issues by finding proper stream algorithms and possible architecture improvements to the stream model. Our implementation plan is the following:

1. Create a data-set that exceeds the SRF size and a with a core computation representative of an unstructured mesh application.
2. Write a C application that will be the base for a correct computation.

3. Write an implementation on Imagine as it is using conditional streams to handle the irregularity of neighbors, this will enable comparison with architecture improvements
4. Integrate a cache between the SRF and the memory in isim, the imagine cycle accurate simulator to act as a memory bandwidth multiplier and improve the performance of our imagine implementation.
5. To tackle the duplication of elements in the SRF, we will look into indexing into the SRF, by dividing its banks. Some issues will arise like conflicting bank indexing, which we will try to handle by requiring a certain latency between an indexing and the return of the values.