Fault Tolerant stencil computation on cloud based GPU spot instances

Abstract:

Stencil computation is a class of widely used computation methods that iteratively sweep through all elements in a grid, updating them using values of neighbors. It forms the core part of many finite-difference partial differential equation (PDE) solvers, and thus crucial for applications based on PDEs, such as fluid dynamics, heat diffusion and image processing.

Existing system

High-resolution simulations can easily exceed the memory capacity of any single computation node. Donath et al. show that a simulation of free surface can scale from hundreds of gigabytes to terabytes of memory usage. For such applications, a distributed solution using multiple nodes is necessary. One of the most cost effective ways of performing distributed processing today is via the cloud. The other issue with a cloud implementation is communication delay. Since the neighboring data are required for the computation of an element in the stencil grid, data must be exchanged between the computation nodes. This makes communication delay critical for the performance of any distributed stencil computation.

Disadvantages:

1. Communication delay is presented in these applications
2. Application performance is slow.

Proposed system:

In this paper, we propose and implemented a fault tolerant stencil framework for cloud-based GPU clusters using spot instances that takes advantage of pipelining to hide communication overhead. With pipelining, the dependency between stencil computation and data movement are

decoupled, and hence execution proceeds in parallel regardless of dependencies. The only issue affecting performance is the balance between stages of the pipeline - a pipeline is only as fast as its slowest component. We investigated the relative as well as scaling relationships between pipeline stages, then modeled and tuned the pipeline to achieve best possible balance.

Advantages:

1. Communication overhead has been avoided.
2. Application performance has been improved by avoiding the communication delay.

SYSTEM REQUIREMENTS

H/W System Configuration:-

- Processor - Pentium –III
- RAM - 256 MB (min)
- Hard Disk - 20 GB
- Key Board - Standard Windows Keyboard
- Mouse - Two or Three Button Mouse
- Monitor - SVGA

S/W System Configuration:-

- Operating System : Windows95/98/2000/XP
- Application Server : Tomcat5.0/6.X
- Front End : HTML, Jsp
- Scripts : JavaScript.
- Server side Script : Java Server Pages.
- Database : MySQL 5.0
- Database Connectivity : JDBC