Master Thesis : Ongoing

Start Date : Nov 15th, 2015
End Date : May 15th, 2016
Title : Job Scheduling for Adaptive Applications in Future HPC Systems
Supervisor : Prof. Dr. Michael Gerndt
Advisor : Isaias Alberto Comprés Urena

Description:

This master thesis deals with the problem of Job Scheduling for Adaptive Applications in Future HPC Systems. Adaptive applications are parallel applications that can exhibit resource dynamism by adapting to a changing allocation of resources at runtime. To improve fault tolerance, load imbalance, energy efficiency and resource utilization in emerging exascale systems, adaptive programming paradigms such as Invasive MPI is being developed as a part of the ongoing INVASIC project. Invasive MPI will allow programmers to implement adaptive MPI applications. In order to support such kind of applications in the workload of HPC systems, Existing resource management and scheduling software needs to be enhanced as they currently support only static applications which do not change their resource set at runtime.

In this work, we explore the idea of separating the concerns of batch and runtime scheduling into two different software layers / components in contrast to the existing systems where both are merged together. A negotiation protocol will be implemented as a means of communication between the two. The motivation behind the negotiation protocol is the conflicting set of objectives between batch scheduler(user perspective: faster response time, fairness for jobs etc.) and run time scheduler(system perspective: maximize utilization, throughput, energy efficiency etc.). The role of the batch scheduler is to forward jobs to a runtime scheduler which is managing the resources in the partition(s) and supports runtime scheduling of adaptive applications. Runtime scheduler will make intelligent expand / shrink decisions by observing scalability behavior of the running applications and use it to predict future resource requirements. The proposed protocol will also allow us to integrate such adaptive resource management systems into existing batch systems and thereby allowing for an easy migration. It also helps us to realize a dynamic and flexible scheduling strategy by balancing the conflicting objectives at the two layers.

The objective of the thesis is to develop an early prototype that can simulate the negotiation between the two layers. The protocol involves the runtime scheduler sending a resource offer to the batch scheduler which then maps the jobs from its queue onto this offer and replies back with this mapping. The dispatching of batch jobs is only event based upon receiving an offer in contrast to the existing systems that have a periodic batch scheduling cycle. The resource offer specifies the availability of nodes at a given point of time and the batch scheduler can either accept or reject the offer depending on its decision making strategy. Similarly, The runtime scheduler can also either accept or reject the mapping it receives. If it accepts, then it will launch the jobs that have been mapped and can do this at a finer granularity level of cores as compared to nodes done by the batch scheduler. It will also provide future time slots for those that cannot be launched. If it rejects then it will try to transform the state of the system by using expand / shrink strategies on the running jobs and send back a new resource offer to the batch scheduler. Negotiations therefore continue until a configurable number of attempts is reached when both parties unconditionally agree to each other.

The prototype will be implemented using SLURM which is an open source workload manager designed for linux clusters that has a plugin-based architecture, is highly scalable and widely used.