Modeling a Grid-Based Problem Solving Environment for Mobile Devices

Stan Kurkovsky, Bhagyavati, Arris Ray, Mei Yang
Department of Computer Science, Columbus State University
{Kurkovsky_Stan, Bhagyavati, Ray_Arris, Yang_Mei}@colstate.edu

Abstract

The paradigm of grid computing has been successfully applied in the domain of computationally-intensive applications supporting scientific research using server-class computers interconnected by wired network. The users of the existing and growing number of wireless mobile devices often demand more computational power than their devices can currently provide. We propose using the computational grid paradigm to build a problem-solving environment for wireless mobile devices.

1. Introduction

A computational grid can be viewed as a transparent aggregation of many computing devices on a network that enables sharing of distributed resources, typically considered in the context of sharing processor power of many computers interconnected by a wired network [2]. Most of the current grid implementations focus on high-performance computing mostly supporting applications of scientific research and they usually employ homogeneous collections of resource-rich, server-class computers. However, wireless mobile devices characterized by limited resources (slower processors, small amount of memory and short battery life) and by their high mobility are gaining popularity and their numbers are growing rapidly. Given the wireless nature of their network connectivity, mobile devices inherit many drawbacks of their transmission medium. Interference in wireless communications can be caused by atmospheric disturbances and environmental factors such as terrain and vegetation. The other major cause of the network instability is the device mobility. Any wireless application involving simultaneous cooperation with other computing devices must be robust enough to shield the user from possible instability of the network by making network access transparent to the end user.

In addition to limited processing power, energy constraints imposed on mobile devices by small batteries present another motivation to move the computational load elsewhere [3]. By redistributing the computational load on demand, a grid-based problem solving environment would enable the users of mobile devices to solve far more complex and resource-demanding problems that they would be able to solve using their individual devices on a stand-alone basis.

In this paper we present a prototype of a grid-based problem solving environment for wireless mobile devices with limited processing power [1]. Its primary purpose is to allow mobile devices with limited resources to solve problems that they would not be able to solve individually by redistributing the computational load onto other computing devices.

2. Grid-based environment architecture

Previously, we proposed an architecture that embraces mobile devices within a wireless cell to share their computational power by joining a framework modeled after a computational grid [1]. A mobile device may be capable of solving a task on its own if it has enough processing power, available memory and battery charge. Otherwise if the task is too big and/or resource-intensive, the mobile device may delegate the task to other available and willing mobile devices available within the grid-based environment. The environment infrastructure facilitates the distribution of the resource-intensive tasks across the community of other grid-enabled mobile devices and enables an effective communication and coordination among them.

Each mobile device willing to join the grid-based environment runs a client application implementing a light weight communication protocol for information exchange with the grid's *Brokering Service*. Any mobile device running this client application is called a *Subordinate* and it can be requested to participate in solving a part of a computationally-intensive task for another mobile device. Any Subordinate mobile device may become an *Initiator* of a distributed task if its user starts a large and/or computationally intensive task. In this case, Initiator must make this task easily distributable across the grid-based environment and to submit this distributed task to the Brokering Service. The Brokering Service and other components of the infrastructure coordinate the distributed execution of the task and facilitate all communication among the devices participating in the grid. The *Keep-Alive Server* handles all aspects of communication between the mobile devices and the rest of the grid-based environment infrastructure. There is a clear separation between the functionality of the Keep-Alive Server and the Brokering Service based on the nature of their responsibilities. The Keep-Alive Server is only concerned with the communication between mobile devices and environment infrastructure. The Brokering Service has all information about available mobile devices. It knows about the task distribution, but not about the task semantics, which are encapsulated within the Initiator. Subordinates are mere workhorses whose processing power is used to solve pieces of a large distributed task.

3. Prototyping the grid-based environment

The main objectives of this prototype are to prove the feasibility of our grid-based architecture, to model all interactions between mobile devices and the environment infrastructure, to prove the concept of task distribution and assembly, and to conduct a number of simulation experiments governed by several control parameters that determine the behavior of the grid.

The simulation prototype is developed to determine the result of introducing mobility and network instability into the wireless grid-based problem solving environment for mobile devices. A high-level view of the prototype architecture consists of the following three components: 1) client application installed on participating mobile devices, 2) Keep-Alive Server, and 3) Brokering Service.

A light-weight client application is installed on all mobile devices enabling them to access and communicate across the grid-based environment. Subordinates may request to join or leave the environment, initiate a task, run assigned partial tasks or return results of completed partial tasks. Initiators may create a distributed task, abort an initiated task or retrieve results for the completed partial tasks.

The Keep-Alive Server is a TCP/IP service that is used for transparent handling of all communication between the grid-based environment and the client applications running on mobile devices. Changes to the state of the grid are reported to the Keep-Alive Server from network devices and from the Brokering Service.

The Brokering Service is the central part of the grid-based environment. It is implemented as a suite of XML Web Services that exposes methods used to collect and store detailed data, which describes the characteristics of the mobile devices comprising the environment population. It also contains information about the current state of the tasks initiated and distributed across the grid-based environment.

If a mobile device registered with the environment initiates a new distributed task, the source code and relevant parameters are submitted to the Brokering Service, which dynamically compiles the code into an executable segment. Following that, the Brokering Service assigns the newly submitted task to available Subordinates. The Keep-Alive Server is then notified that a task is ready to be distributed and transmits the task to the appropriate Subordinates specified by the Brokering Service.

4. Issues of transparency and dependability

Transparent access to grid-based aggregation of computing power grants mobile devices with limited resources the means to obtain results of computationally-intensive tasks by harnessing the collective resources of the grid. Because it shields the user from the system's complexity, the proposed grid-based model provides transparent access to shared resources. The grid-based problem solving model attempts to encapsulate the complexity of the underlying architecture from clients of the resource-sharing services.

Dependability can be defined as the trustworthiness of a computing system. While the wireless grid can be used for computationally-intensive tasks, it must also be a dependable system in itself or part of the underlying dependable network.

5. Preliminary results and future work

The prototype of the grid-based problem solving environment is designed to allow for the modification of three control variables: initial grid population size, device mobility and frequency of task initiation. The initial grid population size specifies the number of mobile devices in the environment at the beginning of a simulation run. Device mobility indicates the probability per unit of time of the event that a new mobile device joins or a current device leaves the environment. The frequency of task initiation indicates the probability per unit of time of the event that a device initiates a new distributed task (that is, a Subordinate becomes an Initiator). The variable used to measure the efficiency of the grid implementation is the total time taken for a task to be initiated, distributed, solved by the Subordinates and returned to the Initiator.

Our plans for future work include conducting a wide variety of simulation experiments using various scenarios determined by different combinations of the control variables. Each scenario is determined by fixed values of two out of the three control variables, while the third variable is subject to changes in its value. We are particularly interested in the effects of network instability and high device mobility on the overall performance of the grid. Specifically, we plan to focus our experiments to determine the varying impact of network instability on the grids performance in situations with different device populations.

6. References

[1] Bhagyavati and S. Kurkovsky. "Wireless Grid Enables Ubiquitous Computing", in *Proceedings of The 16th International Conference on Parallel and Distributed Computing Systems (PDCS-2003)*, Reno, NV, Aug. 2003.

[2] I. Foster et al. "The Anatomy of The Grid: Enabling Scalable Virtual Organizations", in F. Berman, G.C. Fox, A.J.G. Hey (Eds.), *Grid Computing*, Wiley, 2003.

[3] Z. Li, C. Wang and R. Xu. "Computation Offloading to Save Energy on Handheld Devices: A Partition Scheme", in *Proceedings of International Conference on Compilers, Architectures and Synthesis for Embedded Systems (CASES'01)*, Atlanta, GA, Nov. 2001.