A Parameter-based Approach to Resource Discovery in Grid Computing Systems

Muthucumaru Maheswaran and Klaus Krauter

Advanced Networking Research Laboratory
Department of Computer Science
University of Manitoba
Winnipeg, MB R3T 2N2, Canada
{maheswar, krauter}@cs.umanitoba.ca

Abstract. A Grid system is essentially an infrastructure that allows location independent access to the resources and services that are provided by geographically distributed machines and networks. One of the fundamental operations needed to support location-independent computing is resource discovery. Generally, resource discovery schemes maintain and query a resource status database. Dissemination of the resource status information is one of the key operations required to keep the resource status databases consistent. This paper examines several approaches for resource status dissemination. A new concept called the Grid potential is introduced in this paper. This concept is used to control the extent of data dissemination in Grid systems.

1 Introduction

The deployment of faster networking infrastructures and the availability of powerful microprocessors have positioned network computing as a cost-effective alternative to the traditional computing approaches. The Grid is defined as a generalized, large-scale network computing system that is formed by aggregating the services provided several distributed resources [2, 6]. A Grid can potentially provide pervasive, dependable, consistent, and cost-effective access to the diverse services provided by the distributed resources and support problem solving environments that may be constructed using such resources.

One of the key motivations for constructing Grids is to provide application-level connectivity among the various machines so that resources and services supported by the individual systems can be shared in a Global fashion. To enable such sharing, it is necessary for the Grid architecture to support several services [2, 7] and resource discovery is one of them.

In a Grid system, the resource discovery service may operate in conjunction with the resource management service. When a client requests service, along with the request it presents a set of attributes that should be satisfied by a candidate resource. The resource discovery process may be responsible for generating a set of best possible candidates for the given set of attributes. The scheduling heuristics that are part of the resource management mechanism may allocate the best resource(s) from the set based on the some criterion. For example, the resource management may solicit bids from the potential candidates and select the resource with the highest bid

to serve the request. Along with other services, resource discovery is necessary to support resources going off-line and coming on-line. Further, the cascaded operation of resource discovery followed by resource allocation can be efficient in an heterogeneous dynamic system such as the Grid.

Generally, resource discovery services use "status" databases that are maintained by network-wide information services to fulfill the client requests. For scalable implementations, it is essential to organize the status databases in a distributed fashion. With a distributed organization for the status databases, the queries can be executed very efficiently but the updates to the databases may be costly. Most of the update costs are caused by the communication operations performed to disseminate status information across the Grid. This paper focuses on approaches for reducing the data dissemination overhead.

In this paper, we introduce a concept called the "Grid potential" that encapsulates the relative processing capabilities of the different machines and networks that constitute the Grid. We show how the Grid potential can be used to adaptively control the extent of data dissemination in a Grid.

Section 2 proposes the idea of Grid potential that is used to adaptively to control the data dissemination overhead. Section 3 discusses the data dissemination approaches for resource discovery operation in the Grid context. Some results from simulation studies that compare the different approaches to data dissemination for resource discovery are presented in this section. Section 4 examines the related work in the research literature.

2 Grid Potential

The Grid potential concept is similar to the time-to-live idea used in the Internet [5]. Informally, the Grid potential at a point in the Grid can be considered as the computing power that can be delivered to an application at that point on the Grid. The computing power that can be delivered to an application depends on the machines that are present in the vicinity and the networks that are used to interconnect them. Consequently, a high-performance machine when connected to the Grid will induce a large Grid potential. This potential, however, will decay as the launch point of the application moves away from the point at which the machine is connected to the Grid. The rate of potential decay depends on the network link capacities. The rest of this section presents a formal definition of the Grid potential idea.

A node in the Grid has several attributes that can be categorized as rate-based attributes and non rate-based attributes. Examples of rate-based attributes include CPU speed, FLOP rating, sustained memory access rate, and sustained disk access rate. A node in a Grid can be characterized by a vector where each element of the vector is an attribute-value pair.

The Grid potential is based on the computing power or operating rate of a node. Therefore, to characterize a node for deriving the Grid potential only rate-based attributes are considered. Let $X = \langle x_0 = a_0, x_1 = a_1, \ldots \quad x_{N-1} = a_{N-1} \rangle$, where x_i

is a rate-based attribute of the system and a_i its value at a given time. Let F be a set of functions $\{f_0, f_1, \ldots, f_{k-1}\}$, where f_i operates on the set X to return a scalar value $l_i = f_i(x_0, x_1, \ldots x_{N-1})$. Depending on the system, different functions may be defined for it. The functions essentially form weighted sums of the attributes that can be interpreted as different types of potentials. For example, the function $l_c = f_c(x_0, x_1, \ldots x_{N-1})$ may be interpreted as the compute potential of the system and another function $l_s = f_s(x_0, x_1, \ldots x_{N-1})$ may be interpreted as the secondary storage potential. While the compute potential f_c may be based on attributes that relate to the processing rate of the node the storage potential f_s may be based on attributes that relate to the performance of the storage subsystem. Further, we could have functions that compute application specific potentials that could be useful if the Grid is used exclusively for particular sets of applications.

While the above functions characterize the different Grid potentials of a node in terms of its operating rates, they are not sufficient to measure the different potentials. Therefore, a suite of corresponding "benchmarking" programs are introduced to measure the different potentials.

Let Γ_i be a suite of benchmark programs meant to measure the potential that corresponds to function f_i. In the benchmark suite $\Gamma_i = \{t^i_0, \ldots t^i_{N-1}\}$, t^i_j is a program specifically designed to evaluate attribute x_j of the node. Designing such programs is feasible because only rate-based attributes are considered for computing the potentials of a node. For example, one of the benchmarking programs might be measuring the rate at which arithmetic operations are being executed.

Definition 1: *Node component potential* (p^C_j) with respect to attribute x_j is defined as the number of operations performed by the node in one second as measured by the benchmarking program t^i_j.

The performance of a node with respect to an application depends on the rate at which the basic operations required by the application can be performed by the node, i.e., the ultimate node performance depends on a weighted average of the individual node component potentials.

Definition 2: *Weighted node potential* (p^W) is defined as a weighted average of the node component potentials $\{p^C_0, p^C_1, \ldots p^C_{N-1}\}$, i.e.,

$$p^W = a_0 p^C_0 + a_1 p^C_1 + \ldots + a_{N-1} p^C_{N-1}$$

The node potential as expressed by the above equation can be considered as a function of the weighting factors and the node component potentials. The weighting factors determine the relative impotance of the different component potentials. In

addition to varying the weighting factors, the component potentials may be varied under certain situations.

We define the potential induced by a machine i at the point of its attachment to the Grid as the *local induced Grid potential* and is defined as $p_i^L = m\, p^W$ where $0 \le m \le 1$. When the machine is exclusively used for Grid computations, $m = 1$ and $0 \le m < 1$ otherwise.

Definition 3: *Grid potential* (p^G) is defined as the maximum of local induced Grid potentials. Suppose M machines are attached to a given node j, then the Grid potential at that node is given by

$$p^G = \max_{i \in [0..M]} \left\{ p_j^L(i) \right\}.$$

The Grid potential induced at the point of attachment (node) drops off as we move away from the node along the Grid. This potential drop is dependent on the network characteristics. The Grid potential induced by a machine at a node other than its point of attachment to the Grid is defined as the *remote induced Grid potential*. Consider a machine that is attached to the Grid at node *i*. Let p_{ij}^R denote the remote induced Grid potential of this machine at node *j*. The remote induced Grid potential p_{ij}^R can be considered as the effective processing power of the machine at node *j*.

3 Data Dissemination for Resource Discovery

3.1 Overview

Maintaining the consistency of the distributed status databases involves disseminating the status information. Based on the extent of message propagation, we can classify the data dissemination schemes into three groups.

Universal awareness: This class of data dissemination algorithms distributes the status information such that a node can learn about every other node in the Grid. For large network sizes, the approaches in this group cause significant amount of communication due to large number of message transfers.

Neighborhood awareness: The dissemination algorithms in this group propagate status information such that a node learns about the other nodes that are less than a fixed distance away from it. Although the approaches in this class limit the dissemination overhead and is scalable to very large network sizes, other components of the resource discovery mechanism should be able to handle the incomplete information in the status databases that are associated with the different nodes.

Distinctive awareness: Because the Grid is a highly heterogeneous system, various nodes on the Grid have different attributes. The nodes with distinct attributes are more significant. The extent of a node's status information propagation is controlled by the

significance of the node. If all nodes are homogeneous, an algorithm in this group reduces to an algorithm in the neighborhood awareness group. In a highly heterogeneous Grid, an algorithm in this group should deliver a resource discovery efficiency close to a universal awareness type algorithm while having a communication complexity closer to the neighborhood awareness algorithm. One way of implementing distinctive awareness is to use the Grid potential idea presented in the previous section.

3.2 Data Dissemination Algorithms

Figure 1 presents the pseudo-code for the dissemination algorithm that executes on each node. This particular algorithm uses the swamping approach for dissemination. Once a message comes into the node it is validated. The validation process implements the different types of dissemination: universal awareness, neighborhood awareness, and distinctive awareness. In universal awareness, the validation process permits all incoming messages. In the neighborhood awareness, it checks the distance from the source to the current node and discards the message if it exceeds the predefined limit.

```
while (true) {
    // process incoming message
    receive messsage (X) {
        // validate the incoming message: this may depend on the local policy
        // if universal awareness this function is always true
        // if neighborhood awareness returns true only
        // if the distance to source is less than m
        // if distinctive awareness returns true only if the local Grid potential
        // is less than or equal to remote induced local Grid potential
        if (validate(X)) {
            // update the data structures that keep awareness information in the node
            process(X)
        }
        // if there are no incoming message then break out the loop to send messages
    } or timeout (n)

    if (currentTime > lastSentTime + n) {
        lastSentTime = currentTime
        // send to logical neighbors
        get the list of neighboring nodes Y
        foreach node in Y
            send status update message
    }
}
```

Figure 1: Pseudo-code for flooding based data dissemination.

The distinctive awareness is implemented by the validation routine discarding the message if the remote induced Grid potential at the local node is less than the Grid potential at the node. It should be noted that the Grid potential at the local node is the maximum of all local induced potentials. Therefore, the messages arriving from remote nodes that induce less remote potential at the local node than its own potential will be discarded. This creates a "masking problem" for nodes "behind" powerful nodes in a network. For example, if a network of nodes is connected to the rest of the network via a powerful node (as explained in earlier sections, we model the Grid as a connected graph with nodes representing machines), the powerful node will drop all incoming data dissemination messages. Thus, the powerful node will block the dissemination of the status information of the "interior nodes." This masking problem is there when a flooding-based algorithm is used for data dissemination. A swamping-based algorithm that increases the neighborhood set as it discovers new nodes will be able to overcome this problem.

To reduce the high message overhead of the swamping approach, it is possible to use a random node-based approach such as the Name-dropper algorithm [3]. Using the random node-based approach instead of the flooding approach avoids the masking problem. Consider the example situation where a powerful node connects a network of less powerful nodes to the rest of the network. As part of their update messages each node will advertise their immediate neighbors to the other nodes. Therefore, the nodes behind the powerful node will be reachable.

3.3 Experimental Evaluation of the Algorithms

To evaluate the performance of the various data dissemination schemes we devised the following simulation study. In this simulation study a computational Grid is modeled by a random graph with the nodes denoting the machines. The data dissemination scheme is responsible for updating the status database that is maintained at each node. Depending on the scheme that is under consideration, we might have a complete database at each node or an incomplete database at each node. We define *data dissemination efficiency* to be 100% if the particular data dissemination algorithm creates local database that is same as an ideal global database. Higher the value the above parameter is the more accurately the local database captures the actual global status picture.

In the simulations, we "estimate" the above parameter by scheduling a stream of jobs onto the Grid using an ideal global database and local database. We use the same scheduling algorithm in both situations and the differences in the decisions taken gives a measure of the difference between the two databases. In addition to the above parameter, we also report another performance measure that is the *schedule deviation.* This parameter is, however, more dependent on the scheduling algorithm than the above parameter, i.e., it is dependent on how far the decisions taken by the scheduling algorithm is dependent on the completeness of the status information.

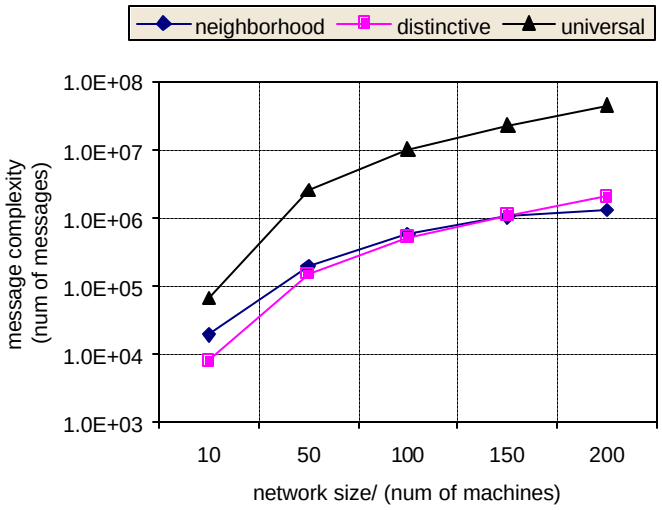

Figure 2: Variation of message complexity with network size.

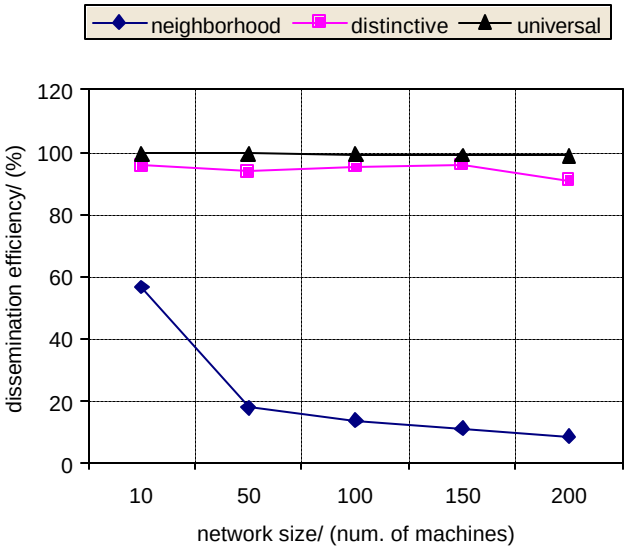

Figure 3: Variation of dissemination efficiency with network size.

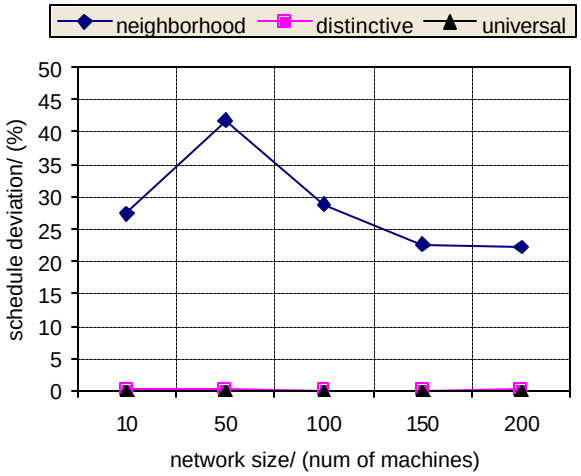

Figure 4: Variation of the schedule deviation with network size.

Figure 2 shows the variation of the message complexity with network size for the different data dissemination schemes. Figure 3 shows the variation of the efficiency of data dissemination with network size and Figure 4 shows the variation of the schedule deviation with network size.

From the above results, it can be observed that the message complexity of the neighborhood and distinctive approaches are about the same and much less than the universal approach. This is expected because in the universal approach, each node sends a message to every other node in the network.

4 Related Work

Because resource discovery is a fundamental operation in distributed computer systems it has been examined in a variety of distributed systems including: mobile computing, wireless sensor networks [4], high throughput computing [9], naming systems [1].

Several data dissemination algorithms based on the universal awareness scheme are examined in [3]. Their paper presents a new algorithm called the Name-Dropper that is proved to have a better communication complexity when compared with three other algorithms based on flooding, swamping, and random pointer jumping, respectively. Our study is different from [3] because we examine the trade-offs between various data dissemination approaches.

Matchmaking [9] is a distributed resource management mechanism developed as part of the Condor [8] project for Grid systems. The matchmaking is based on the idea that resources providing services and clients requesting service advertise their characteristics and requirements using classified advertisements (classads). A matchmaker service that may be either centralized or distributed matches the client requests to the appropriate resources. The matchmaking framework includes several components of a resource discovery mechanism.

The classad specification defines the syntax and semantic rules for specifying the evaluating the attributes associated with the characteristics and requirements. The advertising protocol lays down the rules for disseminating the advertisements. Our study differs from their work because we examine techniques for performing efficient data dissemination to support resource discovery. It may be possible to use the classad language as the specification language in the implementation of our scheme.

5 Conclusions

In this paper, we examine various strategies for data dissemination. We introduce a new class of data dissemination strategies called the distinctive awareness. This class of strategies can result in algorithms that have improved resource discovery efficiency with reduced communication overhead. We use a new concept called the Grid potential for implementing this class of algorithms. The Grid potential quantifies the relative processing powers of the different machines in a Grid.

We performed simulation studies to examine the performance trade-offs of the different data dissemination schemes. Several aspects of the Grid potential concept needs further investigation. One of them is to use application based measurement strategies for the Grid potential instead of using special benchmarks as proposed in this paper. Another one would be construct theoretical performance models for data dissemination algorithms that belong to the distinctive awareness category.

In summary, this paper introduces a new class of data dissemination for resource discovery in distributed computing systems and in particular for resource discovery in Grid systems. A novel idea called the Grid potential is also presented.

References

[1] W. Adjie-Winoto, E. Schwartz, H. Balakrishnan, and J. Lilley, "The design and implementation of an intentional naming system," *Operating System Review*, Vol. 34, No. 5, Dec. 1999, pp. 186-201.

[2] I. Foster and C. Kesselman, *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann, San Fransisco, CA, 1999.

[3] M. Harchol-Balter, T. Leighton, and D. Lewin, "Resource discovery in distributed networks," *ACM Symposium on Principles of Distributed Computing*, May 1999, pp. 229-237.

[4] W. R. Heinzelman, J. Kulik, and H. Balakrishnan, “Adaptive protocols for information dissemination in Wireless sensor networks,” *ACM Mobicomm*, 1999, pp. 174-185.

[5] C. Huitema, *Routing in the Internet*, *Second Edition*, Prentice-Hall, Upper Saddle River, NJ, 2000.

[6] W. E. Johnston, D. Gannon, and B. Nitzberg, “Information Power Grid Implementation Plan: Research, Development, and Testbeds for High Performance, Widely Distributed, Collaborative, Computing and Information Systems Supporting Science and Engineering,” NASA Ames Research Center, `http://www.nas.nasa.gov/IPG`, 1999.

[7] K. Krauter and M. Maheswaran, *Architecture for a Grid Operating System*, Technical Report TR-CS-00-12, Department of Computer Science, University of Manitoba, May 2000.

[8] M. J. Litzkow, M. Livny, and M. W. Mutka, “Condor - A hunter of idle workstations,” *8th International Conference on Distributed Computing Systems*, 1988, pp. 104-111.

[9] R. Raman, M. Livny, and M. Solomon, “Matchmaking: Distributed resource management for high throughput computing,” *7th IEEE International Symposium on High Performance Distributed Computing*, 1998, pp. 28-31.