Non-Uniform Information Dissemination for Dynamic Grid Resource Discovery

Vishal Iyengar, Sameer Tilak, Michael J. Lewis and Nael B. Abu-Ghazaleh
Department of Computer Science
State University of New York at Binghamton
Binghamton NY 13902, USA

Abstract

Effective use of computational grids requires up-to-date information about widely-distributed resources within it – a challenging problem given the scale of the grid, and the continously changing state of the resources. We propose non-uniform information dissemination protocols to efficiently propagate information to distributed repositories, without requiring flooding or centralized approaches. Capitalizing on the observation that grid resources are of more interest to nearby users, we disseminate resource information with a frequency and resolution inversely proportional to the distance from the resource. Results indicate a significant reduction in the overhead compared to uniform dissemination to all repositories.[1]

1. Introduction

Wide area computational grids contain an abundance of resources that hold the potential of together solving computationally intensive problems faster than ever possible before. The diversity and heterogeneity of the component resources can help programmers match their applications to the resources that are best suited to run them. This suitability could be due to many factors, including (1) affinity of codes to certain processor types, (2) memory requirements, (3) processor speed requirements, and (4) proximity to necessary data sets and other suitable computational nodes (for other cooperating parts of the application). Importantly, this list can be affected by dynamic properties of the computational nodes. Current load averages affect CPU performance; the amount of available memory varies as processes start up, claim memory, free it, and terminate; data sets move, and processes migrate. To make effective placement decisions, and indeed to realize the potential of computational grids in solving larger problems more efficiently, grid schedulers—whose job is to map application objects to computational resources—must have access to up-to-date information about grid resources.

The basic approaches to collecting and discovering this information—the ones that may be appropriate for limited size single-site distributed systems—will not scale with the expected number of resources, applications, and resource discovery queries in grids. In particular, directly querying remote resources would require too many messages to remote locations, and does not address the problem of discovering the names of the resources to query. Alternatively, centralized information repositories that are proactively updated with grid resource information would contain too much information and would attract too much traffic, due both to the updating of this information and the queries against it. Caching and replication help, but do not necessarily yield scalable solutions.

This paper investigates a new approach for scalable grid resource discovery using replicated information repositories that are updated non-uniformly, each with the resources that are closest to them. The proposed ideas are based on approaches recently developed for sensor networks [15]. In particular, we use non-uniform dissemination (as opposed to full dissemination) of resource information to reduce the overhead of uniform information replication while maintaining accurate information at locations where it is most likely to be needed. We capitalize on the observation that grid resources tend to be of more interest to nearby users, because the overhead in starting the job and transferring data and results increases with the distance to the resources. The dissemination protocols work by propagating the resource state information more aggressively and in more detail to nearer information repositories than they do to farther ones. Thus, repositories have more accurate and more fresh information regarding nearby resources with less accurate and fresh information about distant resources.

We argue that this approach is suitable for grid environments because the number of resources and the relative uniformity of their distribution makes any given resource request satisfiable by resources residing in any one of sev-

1 This research is supported by NSF Career Award ACI-0133838 and DOE Grant DE-FG02-02ER25526.

eral locations within the grid. We propose augmenting dissemination with a query-side approach that pulls information from remote resources; this complementary approach would be triggered by a failure to satisfy the query after consulting actively disseminated information. Non-uniform information dissemination must balance the cost of propagating the information against its potential value. For grid resources that are truly unique or scarce, non-uniform dissemination has to propagate information widely; otherwise only nearby application schedulers would discover a scarce resource. However, since resource characteristics will often be common across enough distributed "sub-grids," and because we expect resources to be plentiful, a majority of the requests can be mapped to resources relatively close to the requester, thereby eliminating the requirement that all information be distributed completely across the entire grid.

The proposed protocols filter information at forwarding nodes, based on one or more of several criteria. This allows them to scale better by cutting down on the number of messages, while still allowing some information to get through to remote regions of the grid. Different protocols can be explored that filter data according to different criteria: for example, probabilistically, if data has not changed recently, if data is changing too rapidly, if data describes a common resource, or if the resource is too far from the source. Clearly, a wide range of protocols can be constructed using these ideas as a basis. We have chosen a representative subset of protocols, both probabilistic and intelligent, and investigated their performance by simulating them in large grid-like environments We varied the grid topologies and the model that defines how the underlying resource information changes over time. We measure the observed error in the monitored information at the remote nodes relative to the actual value of the information at its source. We couple the error graphs with a characterization of overhead—the amount of data that is propagated through the grid to achieve the reported error rates.

Our experiments demonstrate that a large saving in overhead in information dissemination is possible without losing much in accuracy. This is especially true if accuracy is measured as a function of resource information importance (which we define, for the sake of presenting the data, as the distance between the resource and the repository at which the accuracy is being measured). As a result, we believe that non-uniform dissemination holds the promise of improving the scalability of grid resource discovery. While both probabilistic and intelligent (change-aware or priority-based) protocols result in this general tradeoff, the intelligent approaches hold clear advantage over the probabilistic approaches. Thus, further research needs to focus on such protocols that intelligently and dynamically balance the cost of propagation against the value of the information.

The remainder of the paper is organized as follows. Section 2 defines the resource discovery problem that the research in this paper addresses. Section 3 then describes related work. Section 4 presents our non-uniform dissemination protocols and makes the case for their application to proactive grid resource information dissemination. To investigate the usefulness of the protocols, we simulate them on various grid topologies, with several different models for how the underlying data changes over time; Section 5 describes these aspects of our experiments, along with the simulation and computing testbed environments. Section 6 presents the results of the simulations. Finally, Section 7 summarizes the paper's contributions and conclusions, and describes several directions for future exploration.

2. Grid Resource Discovery

The grid resource discovery problem can be defined as the problem of matching a query for resources, described in terms of required characteristics, to a set of resources that meet the expressed requirements. The problem is complicated by the fact that some resource information (e.g., CPU load or available storage) changes dynamically. The problem may be viewed as one of efficient access to widely-distributed real-time data streams representing the state of grid resources. Similar problems have been studied (e.g., web-content delivery acceleration and others; further details are provided in Section 3). Our problem differs in important ways in terms of the type of information being accessed and the nature of the queries that are generated.

A primary design decision is whether to pull the data from the sources in response to direct queries, or to push the data proactively towards potential information consumers. Pushing data may result in high overhead unless the data is of interest to many users. Furthermore, between updates, the value of the data may become stale. Pulling data increases the query delay and overhead, but results in fresh data.

Caching may be used to optimize the pull model. However, given the dynamic nature of grid resources, it is unclear whether caching will be beneficial; data may quickly become stale. Similarly, replication may be used for the proactive push model such that the information is pushed to multiple distributed repositories. The repositories can hold mutually exclusive (a resource state is maintained in exactly one repository), fully replicated (each repository contains full resource information), or partially replicated (some information replication exists) data. Replicating the data makes query execution faster; each query can be answered by a resource repository close to it. However, the cost of replication may be excessive – the data has to be disseminated to all replicas. Whereas caches are populated by prior queries, information repositories are populated proactively either with a specified period or based on the rate of change of the resource value.

The approach we propose in this paper fits within the distributed information repository category, with partial non-uniform replication and update. Like full replication, the approach aims to have the required resource information available locally to query originators. However, instead of tolerating the overhead of full dissemination of resource information as is required by full replication, we disseminate the information non-uniformly. More precisely, as a resource update is forwarded from the resource to neighboring repositories, they each forward this information selectively based on some criteria (this process is repeated recursively). As a result, the freshness and resolution of resource information at a given repository is a function of the value of the resource and the distance between the resource and the repository—nearby repositories will have fresh high resolution information, while distant ones receive less frequent updates. Since most queries would prefer nearby resources, this approach provides many of the benefits of full replication at a small fraction of the overhead. We use a pull-based safety method to allow a wider resource search if local repository information is not sufficient.

3. Related Work

The resource discovery problem comprises several necessary components. Iamnitchi and Foster [7] define four axes of the solution space. In their taxonomy, the *membership protocol* determines how nodes are added to the grid and start being discovered, *overlay construction* determines direct collaborator pairs among members, *preprocessing* describes steps that are executed prior to information requests being issued, and *request processing* maps specific requests to resource sets that can satisfy them. Relative to Iamnitchi and Foster's taxonomy, our work is best viewed as preprocessing that is intended to make query processing more efficient. Our information dissemination protocols can be viewed as orthogonal to the overlay topology of forwarding nodes. However, the overlay topology could have a significant influence on the effectiveness of the protocols, so we investigate several different topologies. The final deployed solution must make the forwarding probabilities sensitive to the overlay topology.

Recently Butt et. al. proposed "flocking" [5] of Condor pools, combined with the Pastry [13] peer-to-peer overlay for scalable discovery of resources [1]. Pastry is locality-aware, so Condor ends up mapping resource requests to nearby Condor pools that can service them, keeping application deployment time down. The aspect of this comprehensive resource discovery and job scheduling scheme that is most related to our work is the advertisement of available resources between Condor pools. A Condor pool sends resource availability information directly to all other pools in it's Pastry routing table, and this information is forwarded with a time-to-live (TTL) field that determines its reach to other more distant pools, which are not present in the local Pastry routing table. This approach achieves non-uniformity in a different way than the randomized protocols we discuss in Section 4; however, Condor shares our goal of deploying information and servicing requests locally.

In our approach, the information propagation is controlled by the intermediate nodes (rather than the information source per Condor). The value of a resource may not be uniformly a function of distance; our protocols let forwarding nodes determine if the resource is important enough to forward. This also allows these nodes to shape the dissemination pattern of the information adaptively according to the observed query patterns from their region of the grid. Furthermore, in the Condor approach, the algorithm for selecting the pools to forward information to (after those specified in the Pastry routing table) is left unspecified. We believe that our study of different non-uniform dissemination protocols could potentially be used to help optimize how these pools are selected, because we characterize the error and overhead associated with different decisions.

Maheswaren et. al. [11] also share the approach of associating higher value with nearby information. They introduce the notion of "grid potential", which weights a grid resource's capability with its distance from the application "launch point". The authors propose and study the tradeoffs between three different protocols. The *universal* protocol attempts to disseminate information uniformly, the *neighborhood* protocol limits the scope of dissemination to nearby nodes, and the *distinctive awareness* protocol is intended for unique grid resources. The idea of having different protocols for different types of resources is similar to our Prioritized Dissemination Protocol (PDP), described in Section 4.2. The authors use simple tests to measure message complexity (overhead) and dissemination efficiency (error), as we do, but despite calling their method a "parameter-based approach", they only explore a single point in the space; the focus of our protocols and the experiments in this paper is on configurability and characterizing tradeoffs between multiple different algorithms.

Other grid resource discovery systems, such as MDS-2 [16] and Legion's collection objects [2], recognize the need to implement scalable resource discovery systems, but do not specify (nor study the properties of) the specific overlay structures into which nodes should be organized. Thus, these approaches provide mechanism and base protocols for storing information and building scalable distributed collections of servers, but do not propose specific organization strategies, overlay topologies, or dissemination protocols.

This paper uses the concept of non-uniform information dissemination recently proposed by Tilak et. al. [15] and applies it in the context of resource discovery in grid environments. Non-uniform information dissemination has

been studied in networking, especially in the ad hoc community. Flooding has historically been used as a base model for resource discovery in networks [6]. Gossiping protocols (e.g., [10]) have been proposed as a more efficient approach than flooding for information dissemination. Kempe et. al. [8] presented theoretical results for gossiping protocols with resource location as a motivating problem and delay as the primary consideration. However, they do not consider application level performance criteria such as accuracy. Ad hoc network routing protocols such as DREAM [14] and Fisheye [12], update the routing tables based on the distance between two nodes, and their mobility. However, their work is limited to adjusting routing tables and does not apply to the actual data that is exchanged between two nodes.

4. Non-Uniform Information Dissemination

This section describes the protocols we have developed for non-uniform information dissemination. We assume that the grid is organized into an overlay topology connecting sites to one another. Each site contains a resource repository; as information is propagated through the overlay topology, a repository that receives the information records it and decides whether to forward it, depending on the specific protocol criteria.

Probabilistic forwarding protocols, described in Section 4.1, are adapted directly from their sensor networks counterparts [15]. They treat all resources the same, and forward information about them probabilistically. Hybrid protocols, described in Sections 4.2 and 4.3, are based on the probabilistic protocols and consider the source and/or value of the information being disseminated when making forwarding decisions. We have designed the hybrid protocols specifically for the grid information dissemination problem; these are new protocols with no counterpart in sensor networks. All of the protocols attempt to filter information to reduce message overhead, so that local repositories contain more accurate and fresh information than distant repositories. The protocols differ in how much data is filtered, and the criteria used to filter, resulting in different tradeoffs between error and performance.

4.1. Probabilistic Protocols

Sensor networks and grids share the property that it is more important to have accurate and fresh information about nearby resources. This is the key property that non-uniform information dissemination exploits; therefore, we expect this class of protocols to be useful in grid environments, as it is in sensor networks.

In probabilistic protocols, information repository ir_k forwards information generated by resource r with probability $p_{r,k}$. Intuitively, when an information repository receives a message, it generates a random number and uses it to decide whether or not to forward the message. Probabilistic protocols can be either *biased* or *unbiased* depending on how the probabilities are set. Unbiased protocols do not consider the source of the information in deciding whether to forward it ($p_{r,k} = p_{s,k}$ for all pairs (r, s)). Conversely, in biased protocols, forwarding probabilities decrease as the forwarding node gets further away from the information source. Note that the implementation of the biased protocols need not maintain separate forwarding probabilities for all pairs of repositories and resources. Instead, a time to live (TTL) field can be set at the resource and decremented on each hop; repositories can use the TTL field to determine the forwarding probability depending on how many hops the information has traveled.

Biased and unbiased probabilistic protocols have several advantages. First, they are simple. The protocols require only that each information repository make a decision, based solely on the forwarding probability (and for the biased protocol, on the number of hops that the information has traveled), as to whether or not to forward each incoming packet. Furthermore, no state information about other nodes in the system, previous values of information, or resource existence, must be maintained. The protocols are also easily configurable. Increasing or decreasing the forwarding probability determines how widely the information is disseminated; in the unbiased protocol, the fact that probabilities are multiplied to determine whether information traverses a given number of hops reduces the likelihood that it will, and for the biased protocol, the dissemination aggressiveness is determined more directly by how the number of hops and the base probability affects forwarding decisions. The result is a protocol that can be easily configured by simple parameters to cover the tradeoff between error an overhead, as illustrated by representative values reported in Section 6. Finally, the protocols can be used without having to assign meaning to the information they disseminate and forward. This makes it easy to add new resource information without changing protocols, and enables the same overlay topology and information repositories to be used for a flexible and extensible set of information types.

4.2. Prioritized Dissemination Protocol (PDP)

This class of protocols forwards resource information differently depending on the nature of the resource. Computational grids are inherently heterogeneous in several fundamental ways, including their components' underlying hardware, systems software, and the communication protocols and networks that connect them.

Resources themselves might have reason to influence how aggressively the information about them is dissemi-

nated. Commercializing the services available in a grid [9] will lead to accounting for resource usage. Widespread dissemination of fresh information about a particular resource could lead to that resource being selected more frequently by queries, which could in turn lead to higher utilization and more revenue generated.

Furthermore, the grid system software might decide to advertise some resources more aggressively than others. This might be useful, for example, if queries and utilization are not uniformly distributed throughout the grid, leading to some resources becoming underutilized. The grid could sense this and propagate information about those resources to more distant information repositories, in hopes of attracting more usage and better distributing load.

These examples motivate the need for mechanism that enables resources to be treated differently from one another, in terms of their dissemination policies. The Priority Dissemination Protocol (PDP) operates as follows. The intermediate information repositories, unlike in the probabilistic protocols, do not implement a fixed forwarding policy; instead, they provide the mechanism for different resources to realize different individual policies. This allows resources, or the systems-level software that might generate and propagate information about them, to influence how aggressively information is disseminated. Essentially, resources are categorized into distinct priority classes, similar to Quality of Service (QoS) networks, and a different forwarding policy can be applied to each class. Full dissemination is clearly not an option for *all* resources, but may be an effective technique when applied to only the very small percentage of machines that are most scarce or most powerful. The dissemination technique does not have to scale with the number of nodes in the grid, only with the number of nodes in the resource class.

The range of protocols that is realizable using this approach is characterized by the number of priority classes and the forwarding policy for each. It is also possible to dynamically vary the priority depending on the behavior of the system (e.g., based on resource usage, query pattern, or query success rate). We investigate a representative subset of choices across these two dimensions in our experiments.

4.3. Change-Sensitive Protocol (CSP)

The volatility of disseminated information also affects how aggressively it should be forwarded. Resource state such as CPU load, and available memory and storage, can change frequently during times of heavy usage. During these times, disseminated information can quickly become inaccurate, and therefore it is unlikely that widespread dissemination will be beneficial. Likewise, when resource state changes slowly over time, propagating updates frequently is unlikely to improve the number of queries that might match the resources, since the values—and therefore the set of queries that would match them—would be only slightly different than if the information was not propagated.[2]

The Change Sensitive Protocol (CSP) protocol filters information that is changing too rapidly or too slowly, and forwards moderate changes in values more aggressively. The forwarding probability depends on the difference between the information that might be forwarded, and the most recent corresponding value that was forwarded.

4.4. Protocol Characteristics

Iamnitchi and Foster [7] list four requirements of resource discovery approaches. These requirements are (1) lack of central or global control, (2) attribute-based (rather than name-based) search, (3) support for intermittent resource participation, and (4) scalability. The protocols in this paper satisfy all four properties. First, they are peer-to-peer based; they neither contain nor require any centralized data structures or entities, once the overlay topology of forwarding nodes is built[3]. Second, the approach is inherently attribute-based, as the very information that we disseminate from resources contains the attributes that describe the current state of the resource, and determine what can be queried against.

The requirement of intermittent resource participation can be supported by associated time to live (TTL) information or by using an invalidation based approach. The scalability requirement is investigated using the performance study of Sections 5 and 6, and is the focus of this paper. Scalability is not a black or white issue; as reported below, our protocols scale to different degrees depending on the overlay topology, the model that describes how the resource state changes, and the parameters of the protocol itself.

5. Description of Experiments

To study the protocols, we varied two factors: the overlay topology that defines the direct connections between information repositories, and the model that defines how the disseminated information changes at the resource. We select a handful of points in a large space of possibilities, and describe them below.

---

2 This observation assumes that the query does not consider the time of the last update, and assume more recent information is more accurate, which is one logical policy; thus, query assumptions and policies should be chosen to match the characteristics of the preprocessing dissemination protocols.

3 Building the overlay topology need not be centralized. Further, it is also possible that existing overlays, used for global naming, scheduling (or any other service that requires a scalable middleware infrastructure), could be reused for information dissemination. This would allow multiple services to benefit from the same infrastructure, while sharing overlay generation and maintenance costs.

5.1. Overlay Topologies

The topologies below are generated by the GT-ITM topology generation tool [17], whose output is a series of graph edges that we converted into DML (Domain Modeling Language) schema for use with SSFNet. We tested *random*, *Waxman*, *locality-based*, and *hierarchical* topologies.

The random topologies, with average node degree of about four, simply provide a basis for comparison with the other more representative models.

The Waxman model [18] bases the probability of an edge between any two nodes in the graph on their Euclidean distance. It is an exponential model whose two input parameters, called alpha and beta, control the total number of edges in the graph and the ratio of long to short edges. We used an alpha value of 0.2 and a beta value of 0.1 to make the number of edges in the graphs comparable to those from the other models, and to ensure that most of the edges in the topology were short, since we expect short edges to be common in real grid overlays.

The Locality Model [18] uses a discrete approach to proximity: it uses two probability parameters, one for local edges and one for non-local ones. Locality is defined using a Euclidean distance computed as a function of the radius of the graph. We used probabilities of 0.05 for local edges and 0.03 for non-local ones; these probabilities were chosen to provide similar average degree to the other models.

Finally, we used a four-layer hierarchical topology to test the performance of our protocols. The idea of non-uniform information dissemination is orthogonal to the overlay topology or the information dissemination mechanism used to push information to all nodes in the overlay. Although we use flooding as the base broadcast mechanism, the hierarchical overlay topology allows us to investigate how a more optimized dissemination backbone would behave.

5.2. Resource Information Variation Models

The characteristics by which the information varies at the resource affects the performance of the protocols. If information rarely changes, or if changes are short-lived, aggressive and complete propagation is wasteful. We explored four different Resource Information Variation Models (RIVMs): (1) *Monotonic Step*; (2) *Gaussian*; (3) *Uniform*; and (4) *Poisson* distributions.

In the Monotonic Step RIVM, resource information values start from zero and increase by one every 100 milliseconds, for the entire period of the simulation. The model was designed as a worst case scenario, because error increases as a function of the staleness of the information, because the information is changing continuously and monotonically. The Gaussian distribution is used as an approximation of CPU load. We set the mean to be 0.5, and the standard deviation to be 0.25 to represent typical CPU loads. We also made sure that the values generated were always positive. A variation of the Uniform distribution was used to model hard drive usage. We assume that disk usage increases steadily, with occasional dips when files are deleted. We used a Uniform distribution to generate and accumulate random numbers—the sum is intended to represent the number of hard drive bytes used. We started at zero, and for each generated random number, we added the value to the running total with probability 0.75, and subtracted it with probability 0.25. The Poisson distribution was considered to simulate rare conditions, like a network link failure. While these are rare occurrences, they indicate events that need to be noticed quickly. The mean for the Poisson distribution was set to 10.

5.3. The Simulation Environment (SSFNet)

The experiments were performed using the Scalable Simulation Framework Network (SSFNet) [4, 3] environment. Each simulation was run for 120 simulation seconds; longer simulations (240 and 300 seconds) showed little difference in behavior.

6. Results

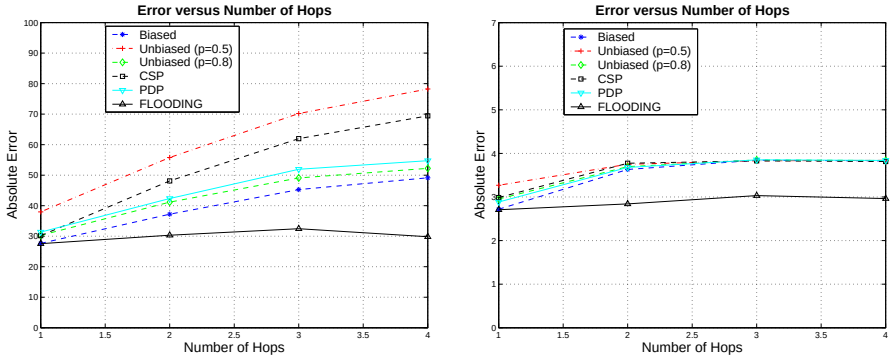
The aim of the simulations was to study the effect of various parameters such as the network topology and the resource information variation model (RIVM) for our dissemination protocols. Clearly, there exists a trade off between communication overhead and error. As the overhead increases (more information is disseminated), the overall error should decrease (assuming that network does not become congested).

To calculate accuracy, we find the difference between a repository's local view of another repository's data and the actual value of that repository's data. A *view* is essentially the latest data that one repository has about another. This view is then weighted with the distance (hop-count) between them. Let $V(R_{i,j})$ denote repository R_i's view of repository R_j's data, and let n be the total number of repositories within the overlay-network. Similar to [15], we use weighted error as one performance metric. The weighted error e_i for a given repository R_i is given as:

$$e_i = \frac{1}{n} \Sigma_{j=1}^{n} |(V(R_{i,j}) - V(R_{j,j})| * w_{ij} \quad (1)$$

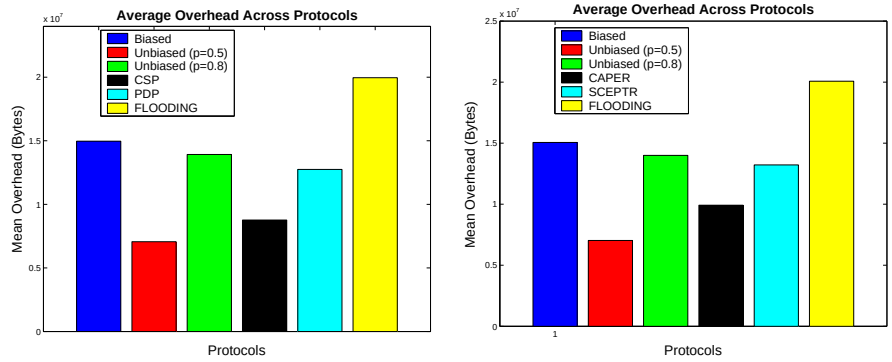
$$w_{ij} = \frac{1}{hop(R_i, R_j)} \quad (2)$$

where $hop(R_i, R_j)$ is the hop-count (application-level) between repository R_i and R_j.



(a) Monotonic RIVM. **(b)** Poisson RIVM.

Figure 1. Waxman Topology (100 Nodes) Mean absolute error versus number of hops



(a) Monotonic RIVM. **(b)** Poisson RIVM.

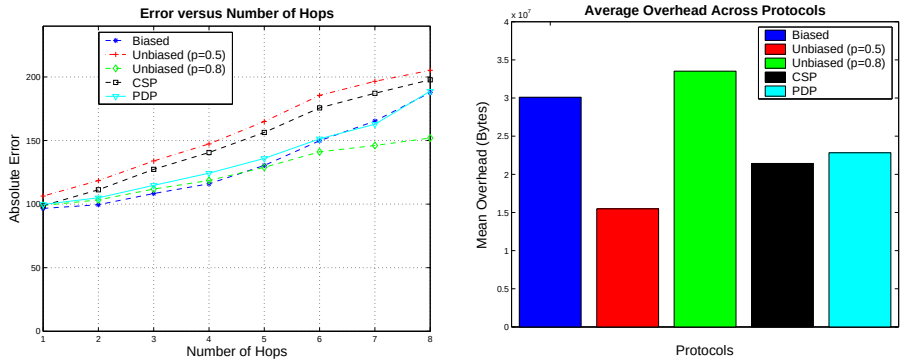
Figure 2. Waxman (100 Nodes) Overhead versus hop number

The first equation shows that for a given repository we calculate weighted average error with respect to all other repositories. To simulate the non-uniform information granularity criteria, the weight varies inversely with the number of hops between a pair of repositories. The term "absolute error" means that the weight is always set to one; that is, error at a repository is not considered relative to its distance from the resource. We also consider mean overhead as another performance metric.[4]

Figures 1, 2, and 3 show the absolute error and overhead for all the protocols using the monotonic and Poisson RIVMs. Absolute error (where error is independent of hop count) is plotted against mean overhead incurred (in bytes). For both models, flooding has low error compared to other protocols (at the cost of high overhead). However, the relative performance of the proposed protocols depends on the RIVM. For the Monotonic RIVM, the biased, the unbiased (p=0.8), and PDP achieve comparable performance, while CSP is better than the unbiased (p=0.5) protocol. Note that CSP has slightly higher overhead than the unbiased (p=0.5) protocol, while PDP has low overhead compared to the unbiased (p=0.8) and biased protocols.

Figure 4 presents the comparative overhead (mean overhead incurred, in bytes) for all RIVMs and topologies. The results show that overhead is a function of both the topology and the RIVM. Furthermore, only the CSP and PDP protocols are sensitive to the type of RIVM used. Thus, in the cases where the domain specific knowledge (such as how and how much the data varies) is not available, simple probabilistic protocols can be used since their performance does

4 All the values reported are averages of 10 simulation runs.



(a) Uniform RIVM.

(b) Uniform RIVM

Figure 3. Random Topology (150 Nodes), Uniform RIVM

not depend heavily on data variations. However, when domain specific knowledge is available, then specialized protocols such as CSP and PDP are promising alternatives.

Figure 5 plots the weighted error, calculated using equations 1 and 2, against the mean overhead incurred (in bytes), for different topologies and RIVMs. Figure 5(a) contains results for the hierarchical topology model and all protocols, including flooding. Clearly, flooding has very low error as compared to the other protocols, at the cost of higher overhead. Figure 5's other graphs are for different topologies with 150 Nodes.[5] The unbiased protocols with (p = 0.5) and (p = 0.8) mark two ends of the error and overhead trade off. More specifically, overhead when the forwarding probability is 0.5 is very low at the cost of high error. Aggressive forwarding (p = 0.8) results in low error at the cost of is very similar to that of the biased protocol both in terms of error and overhead. The overhead for CSP and PDP is in between the basic randomized protocols. PDP and CSP are attractive because they exhibit lower overhead compared to the randomized protocols, while maintaining comparable error.

6.1. Prototype implementation in JAVA

In addition to the SSFNet study, we also built Java-based prototype implementations of the protocols. The multi-threaded implementation uses RMI for all communication. We tested the implementation using various configurations on a cluster of 16 dual-processor machines. The results from these tests followed similar trends to our SSFNet simulations. Although we have not included those results in this paper, they resemble those presented here and are available for reference.

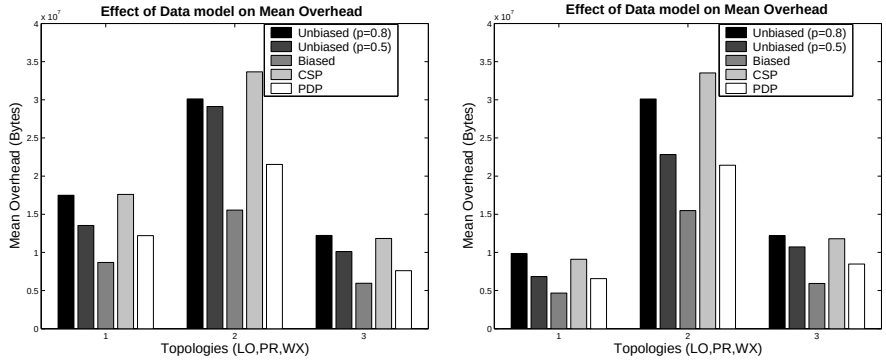
5 We could not simulate flooding for these topologies, due to computational resource constraints.

7. Summary and Future Work

This paper describes the grid resource discovery problem, and outlines the space of solutions that can be used to address it. We suggest the use of non-uniform dissemination protocols to propagate resource information more accurately and more frequently to nearby information repositories (as opposed to uniformly across all repositories). We introduce two new protocols for dynamic information dissemination; the Change Sensitive Protocol (CSP) filters information from being disseminated if it changes too quickly or too slowly, and the Prioritized Dissemination Protocol (PDP) allows resources to be separated into priority classes, with different forwarding policies implemented for each. The protocols are simple, require little or no state information, and can be customized to realize different accuracy/overhead tradeoffs. Our SSFNet simulation of several representative instances of the protocols reveals the characteristics of the tradeoff between message overhead and observed error for different overlay topologies and data change models.

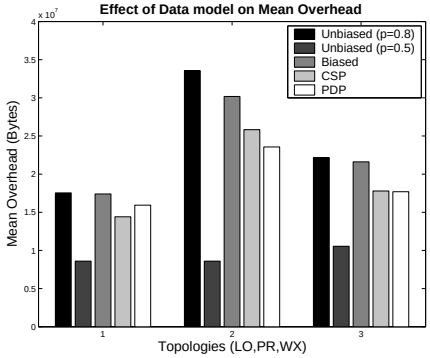
Our protocols represent a first step toward scalable grid resource discovery. This is a rich problem domain with many open problems. More detailed evaluation would include a wider range of topologies and better information dissemination overlay backbones. The criteria for propagaing information non-uniformly can be studied further and improved.[6] Promising approaches include, for example, allowing query patterns or a resource's similarity to other resources or resource utilization to determine forwarding probabilities. Alternatively, query success rate can be used

6 Developing such criteria can benefit from resource and query traces from a realistic large-scale grid environment. Unfortunately, such information is not currently available.

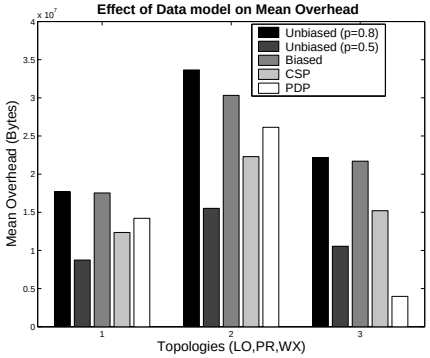


(a) Monotonic RIVM

(b) Uniform RIVM.



(c) Gaussian RIVM



(d) Poisson RIVM.

Figure 4. Comparative Overhead

to influence forwarding probabilities using closed loop control.

Another promising idea is resource aggregation. This paper explores reducing the frequency of resource updates using filtering at intermediate nodes. An alternative approach to reducing dissemination overhead is to reduce the resolution of the advertised resources using aggregation. We envision a hierarchical organization of information repositories. Each repository would have detailed resource information about neighboring resources.

We believe that learning the RIVM from the data itself might enable self-adaptive protocols that dynamically adjust their parameters (such as forwarding probability) as a function of the observed resource state behavior. We believe that such a cross-layered architecture, where data semantics drive protocol behavior, can enable self-adaptive and fault-tolerant large-scale grid middleware.

References

[1] A. R. Butt, R. Zhang, and Y. C. Hu. A Self-Organizing Flock of Condors. *SC '03*, November 15-21, 2003, Phoenix, AZ.

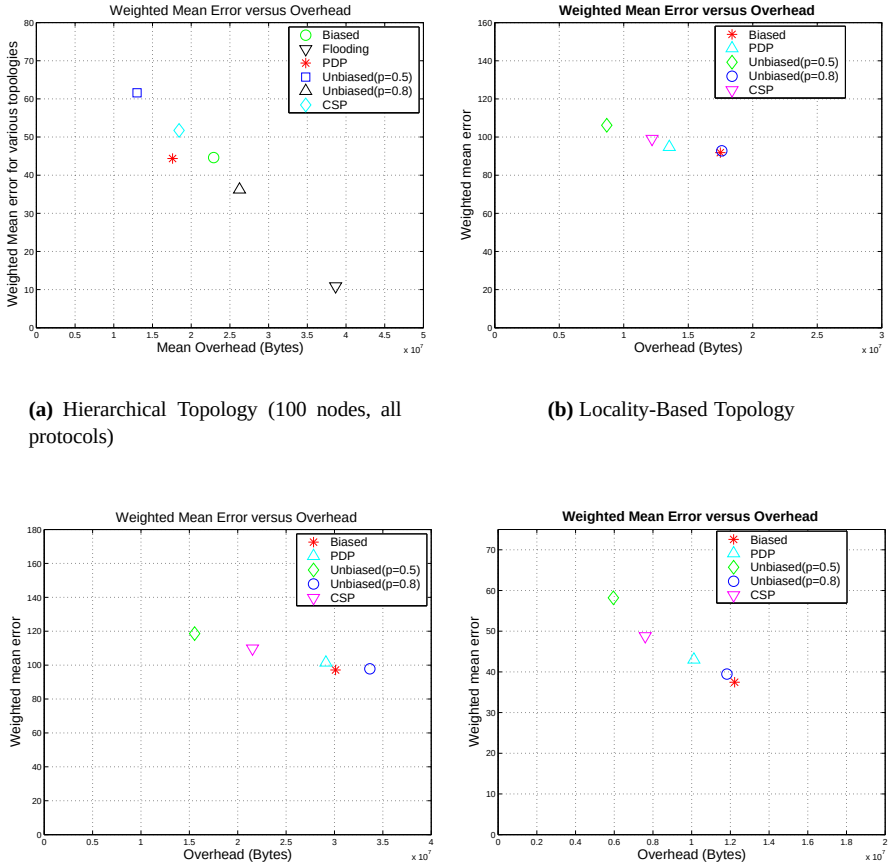
[2] S. J. Chapin, D. Katramatos, J. Karpovich, and A. Grimshaw. The legion resource management system. *Proceedings of the 5th Workshop on Job Scheduling Strategies for Parallel Processing (JSSPP '99)*, April 1999.

[3] J. Cowie, H. Liu, J. Liu, D. Nicol, and A. Ogielski. Towards realistic million-node internet simulations. In *Proceedings of the 1999 International Conference on Parallel and Distributed Processing Techniques and Applications*, June 1999.

[4] J. H. Cowie, D. M. Nicol, and A. T. Ogielski. Modeling the global Internet. *Computing in Science and Engineering*, 1(1):42–50, January/February 1999.

[5] D. Epema, M. Livny, R. Dantzig, X. Evers, and J. Pruyne. A Wordwide Flock of Condors: Load Sharing Among Workstation Clusters. *Journal of Future Generations of Computer Systems*, 12, 1996.

[6] M. Harcol-Balter, P. Leighton, and D. Lewin. Resource discovery in distributed networks. In *Proc. of ACM PODS 1999*,



(a) Hierarchical Topology (100 nodes, all protocols)

(b) Locality-Based Topology

(c) Pure Random Topology

(d) Waxman Topology.

Figure 5. Weighted Error versus Overhead for Monotonic RIVM

pages 229–237, 1999.

[7] A. Iamnitchi and I. Foster. A Peer-to-Peer Approach to Resource Location in Grid Environments. In J. W. et. al., editor, *Grid Resource Management*. Kluwer Publishing, 2003.

[8] D. Kempe, J. Kleinberg, and A. Demers. Spatial gossip and resource location protocols. In *Annual ACM Symposium on Theory of Computing (STOC)*, 2001.

[9] C. Kenyon and G. Cheliotis. Architecture requirements for commercializing grid resources. In *Proceedings of the 11 th IEEE International Symposium on High Performance Distributed Computing HPDC-11 20002 (HPDC'02)*, page 215. IEEE Computer Society, 2002.

[10] L. Li, J. Halpern, and Z. Haas. Gossip-based ad hoc routing. In *IEEE Infocom*, 2002.

[11] M. Maheswaran and K. Krauter. A parameter-based approach to resource discovery in grid computing system. In *GRID*, pages 181–190, 2000.

[12] G. Pei and M. Gerla. Fisheye state routing in mobile ad hoc networks. In *Proceedings of ICC'2000*, pages D71–D78, 2000.

[13] A. Rowstron and P. Druschel. Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems. *Lecture Notes in Computer Science*, 2218:329–350, 2001.

[14] B. S., I. Chlamtac, V. R. Syrotiuk, and B. A. Woodward. A distance routing effect algorithm for mobility (dream). In *4th Annual ACM/IEEE Intl Conf. on Mobile Computing and Networking*, 1998.

[15] S. Tilak, A. Murphy, and W. Heinzelman. Non-uniform information dissemination for sensor networks. In *11th IEEE International Conference on Network Protocols (ICNP'03)*, 2003.

[16] G. P. Website. Mds-2.1 alpha release.

[17] E. Zegura and K. Calvert. GT Internetwork Topology Models (GT-ITM). `http://www.cc.gatech.edu/projects/gtitm/`.

[18] E. W. Zegura, K. L. Calvert, and M. J. Donahoo. A quantitative comparison of graph-based models for Internet topology. *IEEE/ACM Transactions on Networking*, 5(6):770–783, 1997.