

Using Queuing Theory to Predict Organizational Metrics *

Bryan Horling[†]
Multi-Agent Systems Lab
University of Massachusetts,
Amherst, MA
bhorling@cs.umass.edu

Victor Lesser
Multi-Agent Systems Lab
University of Massachusetts,
Amherst, MA
lesser@cs.umass.edu

ABSTRACT

Most existing organizational design processes focus on either the qualitative or domain-independent features of candidate designs. This paper demonstrates the significance of domain-specific features through an examination of an organizationally-driven information retrieval network. The behavior of a distributed search process and the consequences of hierarchical control are described. A model capable of predicting these characteristics is created in the ODML framework using techniques from queuing theory. This model can be used to guide the search for an appropriate design.

Categories and Subject Descriptors

I.2.11 [Distributed Artificial Intelligence]: Intelligent Agents; I.6 [Simulation and Modeling]: Model Development; H.5.3 [Group and Organization Interfaces]: Organizational Design

General Terms

Design

Keywords

Organizations, Organizational Modeling, Queuing Theory, Hierarchical Systems, Quantitative Modeling

1. INTRODUCTION

Much of the existing work on organization design in multi-agent systems has focused on the qualitative aspects of those designs, or on a predefined set of numeric but largely domain-independent characteristics. In our recent work [2] we have developed a new

*This material is based upon work supported in part by the National Science Foundation Engineering Research Centers Program under NSF Award No. EEC-0313747. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

[†]Currently employed at Google, Inc.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

AAMAS'06 May 8–12 2006, Hakodate, Hokkaido, Japan.
Copyright 2006 ACM 1-59593-303-4/06/0005 ...\$5.00.

representation called ODML that incorporates arbitrary quantitative information into the organizational model. Such a model can be used to make detailed predictions of how candidate designs will perform in different circumstances, which can be used as part of a larger search process through the space of design alternatives. Although such models can be more difficult to construct, we believe the benefits this more refined view can provide warrant research.

In particular, we believe that the ability to make concrete, numeric predictions about a range of domain-specific organizational characteristics can improve the design process by capturing the complexities that exist in realistic systems. In contrast to solutions that use instrumented simulations to make similarly detailed predictions, such explicit models also offer rapid but precise design evaluation, facilitating search of the design space. In this paper we demonstrate this potential by describing how an organizationally-driven information retrieval system has been modeled in the ODML framework. The end product of the modeling process is a single, succinct construct that both describes the space of alternatives and predicts the performance of individual designs.

The information retrieval model presented here is inspired by work by Zhang et al. [3], which uses a hierarchical organization to structure an information retrieval system. This allows queries to quickly propagate to data sources, and results routed back to a single agent in the network. At the top level of each hierarchy is a mediator responsible for providing a description of the data available in the information sources present in the hierarchy below it. Information sources may be an individual database, or an aggregator that manages other sources. Multiple mediators may exist, which service user queries by first using the signatures of other mediators to discover data sources, then routing the query to the appropriate mediators, and finally collecting the resulting data.

This organizational design provides several advantages. The use of hierarchies introduces parallelism into query processing and distributes the communication load. Conversely, if the data sources are widely distributed it may require an extensive search process to obtain a good result. There also exists a tension between the width and height of hierarchical structures, as narrow structures can have high latency and wide structures create bottlenecks. Determining what particular design to use can be a difficult problem, owing to the complex and interdependent nature of the organization's characteristics, interactions with the environment, and specified system requirements. We show in this paper how an ODML model facilitates this selection process by capturing these interactions, and using them to make concrete predictions about runtime behavior.

2. MODELING RESPONSE TIME

Similar to models created with other organizational languages, this model includes notions of roles, a task environment and perfor-

mance constraints. However, other relevant phenomena that must be captured in this domain are significantly different, and drive the shape of the organization in different directions. These include statistically predicting the results of the source discovery process, determining how the information contents of a hierarchy affect its expected load and approximating the effects of increased signature uncertainty caused by summarization. Although each can be modeled as a particular, distinct characteristic of the system, in reality they interact in complex ways. The tensions that arise in the resulting model embody the tradeoffs and decisions that must be made when designing the organization. This paper will focus on these expected response time of the system; the reader can refer to [2] for a description other characteristics this model possesses.

The IR system's response time is the duration of time between a query and the final response, a characteristic that is clearly important from an evaluation standpoint. Several design features affect this value. For example, each communication event incurs some message transit latency. The query processing by the databases, and the aggregation performed by both the aggregator and mediator will take some variable amount of time. Because queries are reflected to all subordinates, the latter two entities must also wait for slowest of their information sources before they can themselves respond. Finally, because multiple queries can exist simultaneously in the network, additional delays at individual agents can be incurred when a query must wait for existing processing to complete.

The rate at which an individual mediator will receive user queries is its *arrival_rate*. Responses will be returned back at this same rate, on average. More concretely, the existing model assumes that queries have a Poisson arrival distribution and mean rate *arrival_rate*. After the query makes its way down through any aggregators, the leaf databases will receive the query, also at rate *arrival_rate*. Each database has a *service_rate*, defining how quickly it can process queries. At any given time there may be previously received queries already being processed or waiting at the database. We assume FIFO processing, so the amount of time any new query must wait will depend in part on these existing queries.

Queuing theory can be used to analyze how long the wait will be, by modeling a database as a *M/M/1* queue with Poisson task arrival rate and service rate. The pdf $f_M(x, \lambda)$ and cdf $F_M(x, \lambda)$ of the *M/M/1* queue's waiting time distribution are given below, where $x \geq 0$ and $\lambda = \text{service_rate} - \text{arrival_rate}$.

$$\begin{aligned} f_M(x, \lambda) &= \lambda e^{-\lambda x} \\ F_M(x, \lambda) &= 1 - e^{-\lambda x}, \end{aligned}$$

The mediator waits for responses from all its information sources before progressing, so its expected service time will be dependent on the maximum service time of those below it. This value can be found by computing the maximum order statistic (the expected maximum value) of those source service times. The model makes a simplifying assumption that the source service time distributions are independent, but does not assume they are identical. The model generates the pdf $f_{(n)}$ and cdf $F_{(n)}$ distributions of this inid max order statistic using the following functions[1]:

$$f_{(n)}(x) = \left[\prod_{i=1}^n F_i(x) \right] \sum_{i=1}^n \left(\frac{f_i(x)}{F_i(x)} \right) \quad (1)$$

$$F_{(n)}(x) = \prod_{i=1}^n F_i(x), \quad (2)$$

where f_i and F_i represent the pdf and cdf of the i th sample, respectively (i.e., the service time distribution of the i th source).

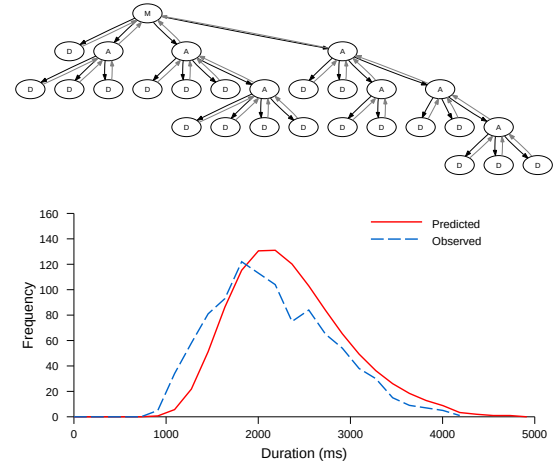


Figure 1: The predicted and observed response time distributions in organizations with twenty-eight agents. Node M is a mediator, A are aggregators, and D are databases.

The mediator itself is not simply a pass-through, but must process and aggregate the resulting data as well, introducing additional latency. Thus, the mediator can also be viewed and modeled as a *M/M/1* queue. The service rate of the mediator depends on the number of responses it receives, which depends on the number of information sources below it. The mediator's pdf and cdf can also be produced using Equations 1 and 1, with $\text{service_rate} = \text{response_service_rate} / \text{num_sources}$, and a Poisson rate of $\lambda = \text{arrival_rate} - \text{service_rate}$.

The model determines the total service time by combining these two activities, modeled as the sum of the times exhibited by these two random variables. The total service time pdf f_C and cdf F_C can then be determined by finding the convolution of the corresponding distribution functions, which has the general form:

$$f_C(x) = \sum_{i=0}^{d.\text{range}/d.\text{step}} f_s(i) f_l(x - i) d.\text{step} \quad (3)$$

$$F_C(x) = \sum_{i=0}^{d.\text{range}/d.\text{step}} f_s(i) F_l(x - i) d.\text{step} \quad (4)$$

For the mediator, f_s would be the aggregate information source pdf given in Equation 1, while f_l and F_l would be the pdf and cdf of the waiting time for the local *M/M/1* queuing process.

Note that Equations 1-4 are recursive, in that they rely upon both the pdf and cdf distributions of the sources below the mediator. The equations make no assumptions about the form of those distributions, so they can be used both when the information source is a single database or an arbitrarily complex aggregator hierarchy. This same assumption also allows Equations 1 and 2 to be used to compute the pdf and cdf distributions for the aggregator itself. The definition terminates in the exponential response distribution exhibited by the databases. By incorporating the cumulative overhead incurred by the message transit times of the query and result propagation process the model can now compute the expected *service_time* of the mediator, allowing prediction of the *response_time* distribution of the organization as a whole.

A representative prediction is shown in Figure 1. The scenario measures the response time performance of a different IR organi-

Organization	Query Rate									
	0.1	0.25	0.5	1	2	3	4	5	6	7
1. 	725	723	719	710	672	544	0	0	0	0
2. 	715	714	711	704	681	621	0	0	0	0
3. 	716	715	712	707	691	664	598	0	0	0
4. 	715	714	712	706	691	664	598	0	0	0
5. 	713	712	709	702	679	621	0	0	0	0
6. 	710	709	706	701	687	662	599	0	0	0
7. 	711	710	708	704	693	677	651	597	465	0
8. 	712	711	709	705	695	680	657	612	482	0
9. 	705	704	702	699	690	677	657	619	517	0
10. 	710	709	707	702	689	665	604	0	0	0
11. 	721	719	717	711	695	665	581	0	0	0
12. 	711	710	707	703	690	666	607	0	0	0
13. 	720	718	715	709	687	629	0	0	0	0
14. 	710	708	706	701	687	662	600	0	0	0
15. 	710	709	707	703	691	674	646	587	439	0
16. 	703	702	701	697	687	673	650	605	471	0
17. 	705	704	702	698	688	674	651	605	472	0
18. 	709	707	704	698	678	641	536	0	0	0

Figure 2: The utility predicted for the range of possible six-database organizations when the query rate is varied. Hollow circles are mediators and aggregators, with solid databases at the leaves. Optimal values for each rate are shown in bold.

zational design, by submitting 1000 queries to it in a Poisson fashion. The tested design is depicted above, along with the predicted and empirical response time data below. The solid line represents the distribution of response times predicted by the ODML model, while the dashed line indicates the observed frequency of response times in the simulation.

As can be seen in the performance graphs, the ODML model does a good job of predicting the response time distribution of the different organizational designs. Additional response time trials were performed for organizations with three [1M,2D], five [1M,4D], 10 [1M,2A,7D], 14 [1M,3A,10D], and 15 agents [1M,6A,8D], with similar results. The coefficient of determination R^2 was greater than 0.8 for all tested scenarios.

3. DETERMINING DESIGN UTILITY

There are many characteristics that impact design choice, including agent load, response quality and resource consumption. These are explicitly represented in the complete ODML model, and are used to derive two aggregate metrics, *response_time* and *response_recall*, which are combined to produce a single *utility* field that can then be used to compare and rank candidate designs during a search process. A complete description of how response recall is estimated in the model can be found in [2].

Figure 2 shows how utility is affected by the expected user query rate. This figure shows the full range of possible organizations that are possible in a six database environment with a maximum height of three and a minimum of two subordinates per node. The design parameters are such that response recall is identical among the alternatives, so utility is determined by the response time. When the query rate exceeds the service rate at one or more points, resulting in an infinite queue, the organization receives zero utility.

The single-level, single-mediator organization number 1 is predicted to be optimal when the query rate is 0.5 or less (i.e., less than one query every other second). This is intuitive, because the slow query rate avoids queuing delays, causing the response time to be dominated by the height of the organization. Note that other short hierarchies such as numbers 11 and 13 also have high utility.

As the query rate increases, first organization 11, then number 8 and finally number 9 become optimal, as the highly-connected mediator in organization 1 becomes a bottleneck. In contrast to most of the competing designs, organizations 11, 8 and 9 are all balanced (as is number 1). In an unbalanced organization, the segment with greater load tends to dominate the response time because the final result must wait for the slowest responder. These three designs avoid this by evenly spreading the load among participants.

The differences between the three revolve around where the final aggregation takes place, and how distributed the database search is. For example, 8 and 9 employ an additional mediator to aggregate the final results, while 11 does not. In 11 and 18, one of the mediators serves this role, which will slightly unbalance the load in the organization for any given query. Therefore, although the lack of another aggregating level saves on communication, the transient load imbalance ultimately leads to an increase in queuing delays.

These tests show one dimension of design tradeoffs that can be made in this environment. Organization 1 is fast but quite centralized. To avoid the increasing chance of bottlenecks, organizations 11, 8 and 9 trades off response time. Other organizations may trade off response quality, or other characteristics. Which design is most appropriate depends on the operational context. These needs may be incorporated into the model's definition of *utility* so that design alternatives are correctly ranked.

This range of viable alternatives, each possessing a set of interacting features, motivates the need for organizational models capable of representing the space of alternatives and predicting their quantitative character. More importantly, this work demonstrates how significant domain-specific characteristics can be affected by choices made in an organizational design. If such characteristics are not accounted for in the design process, or if they are approximated to the degree that important interactions are lost, then the ramifications of decisions that affect those characteristics may also be lost. We believe this will affect the quality of any resulting design, as those interactions are real and must be considered. This fact supports the detailed view of organizational behavior that is expressed in this paper, and the general ODML modeling framework we have designed to capture such details.

4. REFERENCES

- [1] H. A. David. *Order Statistics*, 2nd Ed. Wiley, 1981.
- [2] B. Horling. *Quantitative Organizational Modeling and Design for Multi-Agent Systems*. PhD thesis, University of Massachusetts at Amherst, February 2006.
- [3] H. Zhang and V. Lesser. A Dynamically Formed Hierarchical Agent Organization for a Distributed Content Sharing System. In *Proceedings of the International Conference on Intelligent Agent Technology (IAT 2004)*, pages 169–175, Beijing, September 2004. IEEE Computer Society.