The Landmark Hierarchy: Description and Analysis

Paul F. Tsuchiya

June 1987

MTR-87W00152

SPONSOR:
Defense Communications Agency
CONTRACT NO.:
F19628-86-C-0001

The MITRE Corporation
Washington C3I Division
7525 Colshire Drive
McLean, Virginia 22102-3481

MITRE Department
and Project Approval: ______________________

ABSTRACT

Hierarchical routing structures are needed to reduce the amount of routing information stored and exchanged by switching nodes in large networks (data or voice). Only one hierarchical structure, the area hierarchy, has been available to network designers. This has resulted in a limited set of design alternatives. In particular, the area hierarchy is known to have some poor survivability characteristics. This paper introduces a new hierarchical structure, the Landmark Hierarchy. Analysis and simulation of the Landmark hierarchy in its static state show that is a viable alternative to the area hierarchy for large network routing. Further work is needed to determine the survivability characteristics of the Landmark Hierarchy in a dynamic network environment.

Suggested Keywords: routing, hierarchical networks, hierarchies, landmark routing, landmark hierarchy, random graphs data communications

ACKNOWLEDGMENTS

The author would like to extend his greatest appreciation to W. Worth Kirkman, who is co-inventor of the Landmark Hierarchy.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Concluded)

LIST OF FIGURES

Figure Number *Page*

LIST OF FIGURES (Concluded)

LIST OF TABLES

EXECUTIVE SUMMARY

INTRODUCTION

Because of the rapid growth of data networks, it is necessary to impose hierarchical routing structures on data networks in order to contain the vast amounts of routing information present in the networks. Until now, the only hierarchy known was the area hierarchy. It has seen extensive research over the past decade. The internet, for example, is a type of area hierarchy in the routing sense.

One problem with the area hierarchy is that it is difficult to manage dynamically. That is, it is difficult to adjust the area hierarchy to topological changes in a distributed, dynamic fashion. This difficulty subjects area hierarchy to failures such as the area partition. This problem is manageable on a limited basis, but not at the level which a DoD network might require.

To deal with this problem, we introduce a new hierarchy, the Landmark Hierarchy, which we believe is much easier to dynamically manage. This paper analyzes the Landmark Hierarchy in its static state—that is, we study its performance as a static element, but do not study how to dynamically manage it. In a companion paper (Tsuchiya, 1987), we consider qualities of the many issues of dynamically managing the Landmark Hierarchy and operating it in existing and future networks. Later, we will analyze and simulate these issues quantitatively.

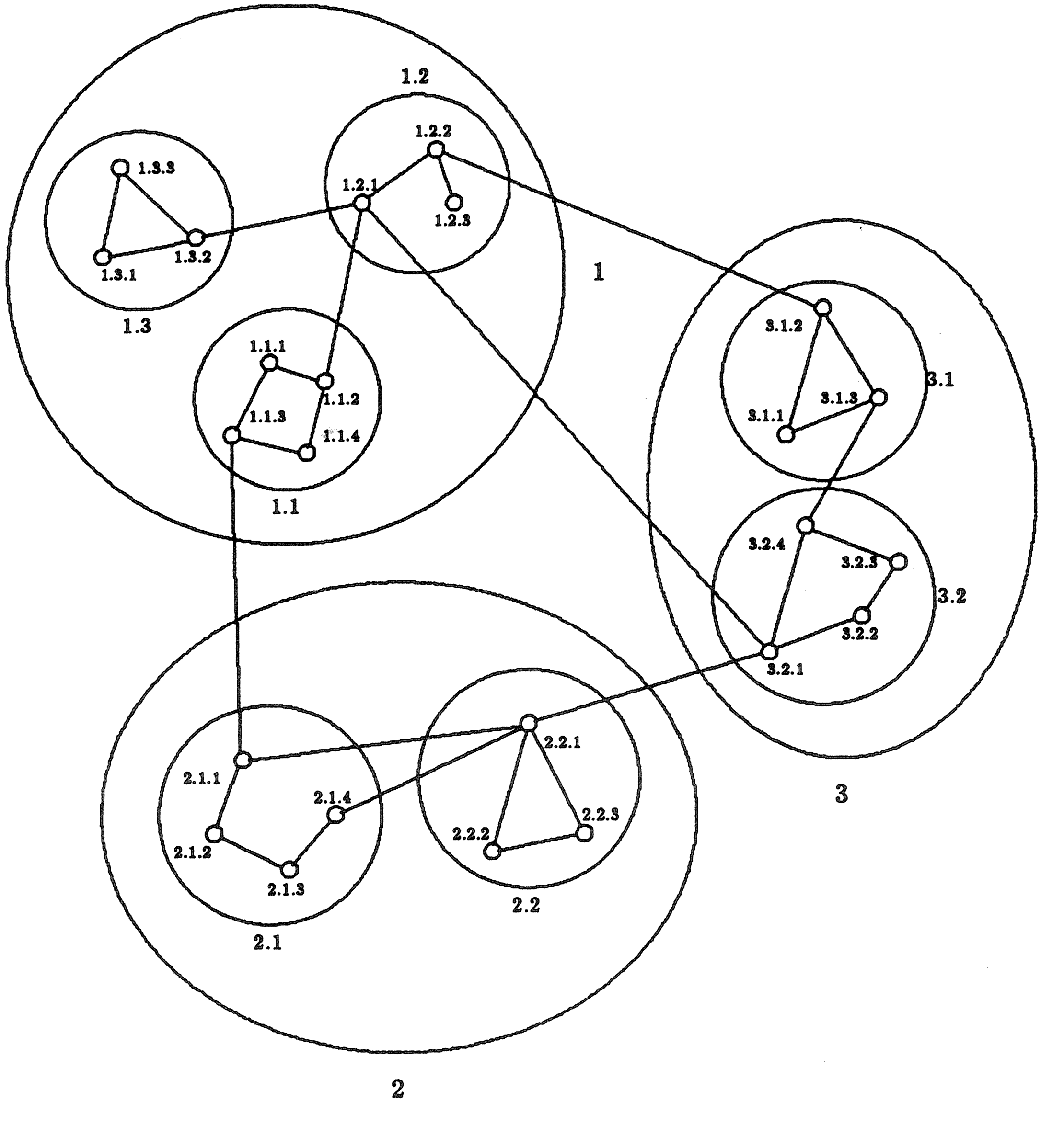
THE AREA HIERARCHY

Figure ES1 shows a computer network of arbitrary physical topology, that is, the topology does not have an obvious structure to it such as a hierarchy, ring, etc. An area hierarchy has been overlaid on the network of Figure ES1. This hierarchy is created by logically grouping nodes into areas, grouping areas into super-areas, and so on.

To route a message in an area hierarchy, a node examines the address of the destination node (the telephone number is a well-known example of such an address), determines which area the node is in, and routes the message to that area. Nodes in that area then further route the message to a sub-area, and so on until the message reaches its destination. This allows nodes outside of an area to view the area as a single entity. The result is that only one entry is required in that node's routing table to route to several nodes in another area. For instance, in Figure ES1, Node 2.1.1 views Nodes 2.2.1, 2.2.2, and 2.2.3 as a single entity, namely, 2.2 —a savings of 3 to 1 in memory overhead (for the table entries) and in link overhead (for the updates required to maintain that entry).

The penalty paid for this savings is increased path length. Using mathematical analysis, Kamoun and Kleinrock have shown that, using the area hierarchy, routing table sizes of $R = HN^{\frac{1}{H}}$, where R is the routing table size, and H is the number of hierarchical levels, can theoretically be achieved (Kamoun, Kleinrock, 1977). Simulation

Figure ES-1
Area Hierarchy Example

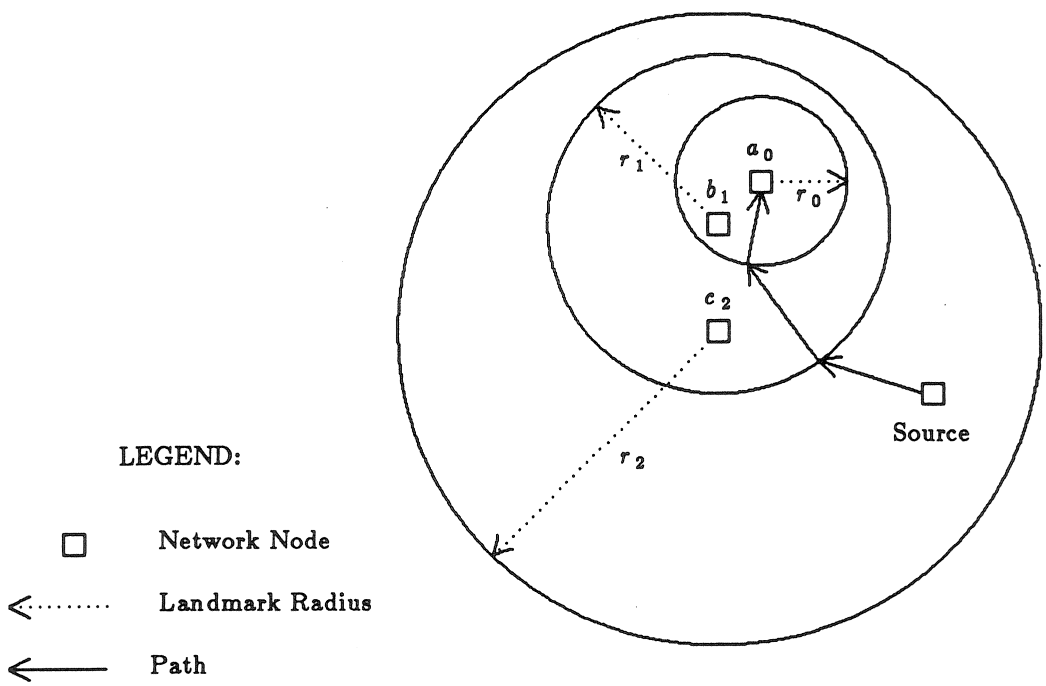


results on a 200-node network produce path lengths of 5% to 15% greater than shortest path, and routing table sizes of 40 or more nodes. This is nearly 3 times worse than Kamoun's theoretical best, and puts into question the practicality of Kamoun's results. Path lengths are even longer for networks with smaller diameters.

THE LANDMARK HIERARCHY

Whereas an area is a group of nodes *all of which have a routing table entry for each other*, a Landmark Vicinity is a group of nodes *all of which have a routing table entry for a single node*, namely, the Landmark. The Landmark, then, is at the center of a Landmark Vicinity, and every node r hops away from the Landmark has a routing table entry for that Landmark. A hierarchy of Landmarks is formed by having all nodes be Landmarks with small Landmark Vicinities, a portion of those nodes be Landmarks with larger Landmark Vicinities, a portion of those with still larger Landmark Vicinities, and so on until there are a few nodes network wide whose Landmark Vicinity covers the whole network. Whereas in the area hierarchy, a node is addressed by its membership in areas, a node in a Landmark Hierarchy is addresses by its proximity to Landmarks. Figure ES2 gives an example.

Figure ES-2
Landmark Hierarchy



Here, Node *a* is the lowest-level Landmark, whose vicinity is shown by the circle defined by radius r_0. Node *b* is the next level Landmark, and so on. Assume we wish to find a path from the node labeled Source to Node *a*. The Landmark Addresses of Node *a* is *c.b.a*. To find the path: Source will look in its routing tables and find an entry for *c* because Source is within the Landmark Vicinity of *c*. Source will not, however, find entries for either *b* or *a*, because Source is outside the Vicinity of those Landmarks. Source will choose a path towards *c*. The next node will make the same decision as Source, and the next, until the path reaches a node which is within the radius of *b*. When this node looks in its routing tables, it will find an entry for *b* as well as for *c*. Since *b* is finer resolution, the node will choose a path towards *b*. This continues until a node on the path is within the radius of *a*, at which time a path will be chosen directly to *a*. This path is shown as the solid arrow in Figure ES2.

There are two important things to note about this path. First, it is, in general, not the shortest possible path. The shortest path would be represented by a straight line directly from Source to *a*. This increase in path length is the penalty paid for the savings in network resources which the Landmark hierarchy provides.

The other thing to note is that often the path does not necessarily go through the Landmarks listed in a Landmark Address. This is an important reliability consideration in that a Landmark may be heavily congested or down, and yet a usable path may be found using that Landmark (or, more literally, using previous updates received from that Landmark).

LANDMARK HIERARCHY ANALYSIS AND SIMULATION

We analyze the Landmark Hierarchy in terms of three parameters: routing table sizes, path lengths, and path distribution. The size of routing tables tells us how much routing overhead the hierarchy produces. In general, this is much less than that seen with no hierarchy. Longer paths are the penalty we pay for using a hierarchy. Path distribution tells us whether the hierarchy skews traffic patterns, thus causing unfairness and undue stress in parts of the network.

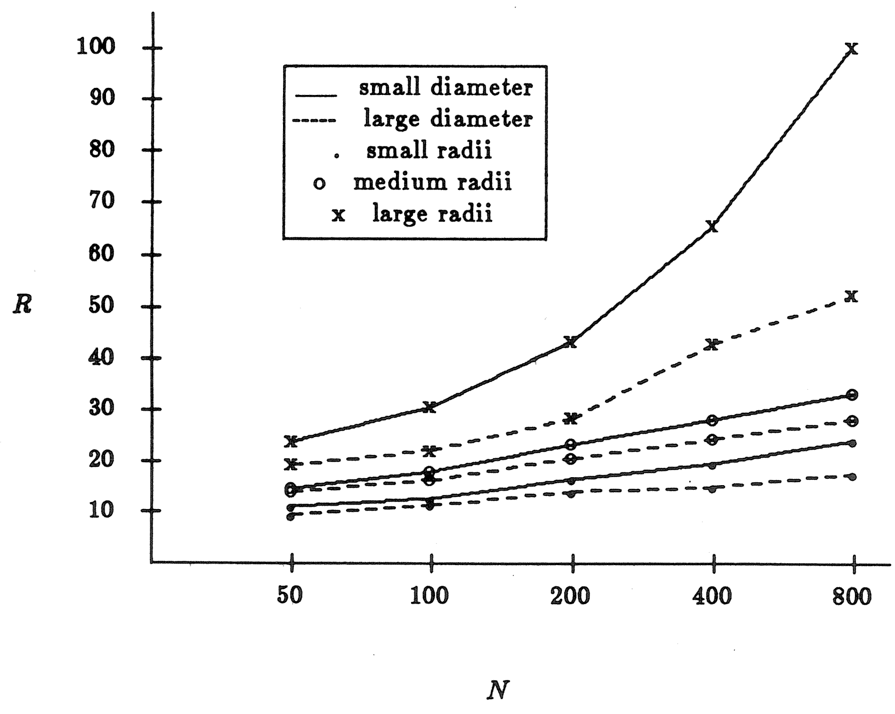
The major portion of our results comes from simulations of the Landmark Hierarchy. In all, we present results of over 1000 Landmark Hierarchy simulations, plus other simulations analyzing various network aspects. The simulations were run on nearly 50 different network types, using 23 different hierarchy descriptions and several variations on these. To support this work, we developed a new model for describing networks, the loop-span model, which allows us to generate quasi-random networks with control over the number of nodes, the node degree, and the diameter. Previous techniques only allow control over the first two parameters.

Our major results are as follows:

1. One can affect routing table sizes and path lengths through adjustments in the hierarchy parameters, namely, the density of Landmarks, and the distance Landmarks can be seen (Figures ES3 and ES4). ($\hat{P}$ is the ratio of the

Landmark Hierarchy path length over the shortest possible path.) This gives the designer considerable flexibility in optimizing the performance of the Landmark Hierarchy for any given network. We also believe that the dynamic Landmark Hierarchy management algorithms may be made to automatically adjust the Landmark Hierarchy parameters for optimal performance in response to changing network conditions.

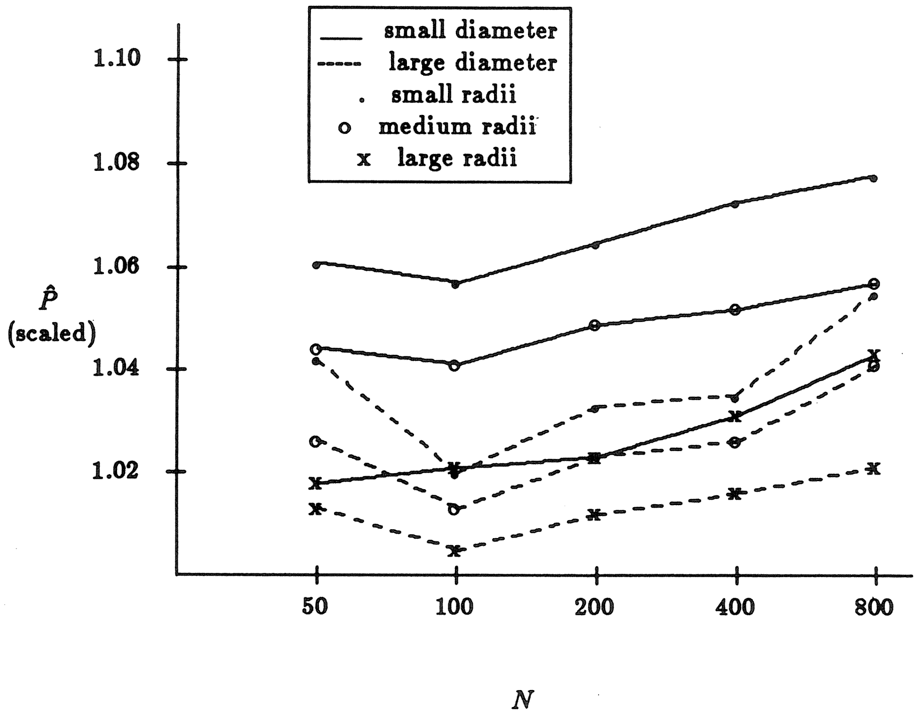
Figure ES-3
Routing Table Size for Realistic Networks



2. When adjusting the Landmark Hierarchy parameters, routing table sizes can only be made smaller at the expense of longer path lengths, and vice versa (Figures ES3 and ES4).

3. Routing table sizes and path lengths are strongly affected by network parameters, namely the number of nodes, the node degree, and the network diameter (Table 5). In particular, networks with very small diameters exhibit both larger routing tables and longer path lengths. In many cases, the

Figure ES-4
Path Lengths for Realistic Networks and Scaled Traffic Matrix



Landmark Hierarchy (and the area hierarchy, for that matter) may perform worse than non-hierarchical routing for these cases.

4. Random assignment of Landmarks performs nearly as well as a uniform assignment of Landmarks. This shows that the Landmark Hierarchy is extremely resilient to the placement of Landmarks—an important survivability consideration.

5. There is a large variance between the size of the largest and smallest routing tables in a network: the largest are about 6 times larger than the smallest.

6. Most routing table entries come from the lower hierarchy levels.

7. Path lengths are the same as shortest path at hierarchy level 0, and get longer at the higher levels. Overall path length increase is very dependent on the traffic matrix, and is significantly better when the majority of traffic is between nodes which are close to each other.

8. We were not able to determine satisfactorily what the effect on path distribution is. Our preliminary results show that the Landmark Hierarchy does not cause undue unfairness in path distribution. Several characteristics which we did not study, namely routing metrics (static and real-time) and the network topology, will tend to have a smoothing effect on path distribution.

9. The number of global Landmarks (those at the highest level of the hierarchy which all nodes can see) impacts routing table size and path length. In general, we were able to get significant improvement in these two parameters by increasing the number of global Landmarks from one to ten.

10. Our results show that, as a rule of thumb, routing table sizes of $R = 3\sqrt{N}$ are typical, where N is the number of nodes. These results are shown in Figures ES3, ES4, and ES5. This can be compared to the area hierarchy where $R = HN^{\frac{1}{H}}$. However, this figure for the area hierarchy is conditional on each area having an identical number of sub areas, a condition not achievable in practice. Further, this figure is not supported by simulation. In comparable simulations, the Landmark Hierarchy showed smaller routing table sizes than the area hierarchy.

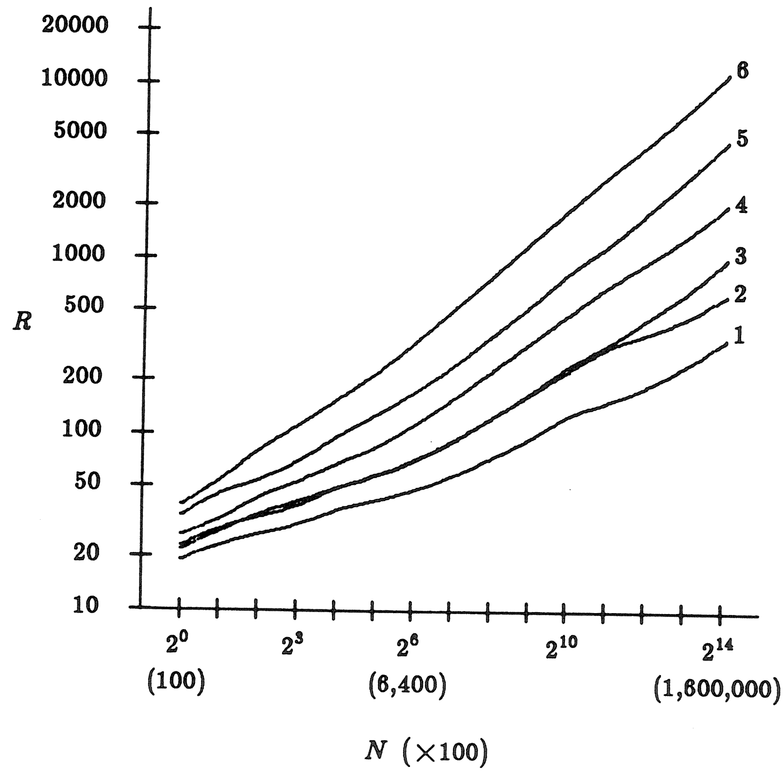
11. We were not able to obtain general figures for the increase in path length. However, we found that path lengths in the Landmark Hierarchy behave similarly to those in the area hierarchy. Our path length simulation results are shown in Figure ES4. Also, our simulation results show shorter path lengths than those shown in similar simulations for the area hierarchy.

12. For networks with very small diameters (close to the smallest possible for a given number of nodes and node degree) the Landmark Hierarchy performs very poorly, and cannot be recommended over the area hierarchy or no hierarchy. However, networks very small with diameters are not normally found in practice (the author knows of no such networks). It is worth noting that path lengths on the area hierarchy are also large for these small diameter networks.

When we compare the Landmark Hierarchy against the area hierarchy, we find that for networks with diameters of reasonable size, the two hierarchy exhibit similar routing table sizes. For networks with very small diameters, the area hierarchy performs much better, because its routing tables are not affected by diameter. For path lengths, we find that the Landmark Hierarchy and the area Hierarchy perform similarly for all diameters—both hierarchies perform poorly for very small diameters, and perform well for reasonable sized diameters.

We conclude that the Landmark Hierarchy is potentially a viable alternative to the area hierarchy, and that research on the Landmark Hierarchy, both in its static and dynamic states, should continue. We cannot recommend the Landmark Hierarchy in networks which do not expect to survive significant topology changes over a short period of time, such as land based commercial networks. However, for networks which are

Figure ES-5
Estimated Performance for Networks Larger Than 800 Nodes



1: Small radii, large diameter, $R = 4.4N^{.33}$
2: Small radii, small diameter, $R = 3.7N^{.38}$
3: Medium radii, large diameter, $R = 3.4N^{.42}$
4: Medium radii, small diameter, $R = 2.9N^{.48}$
5: Large radii, large diameter, $R = 2.9N^{.54}$
6: Large radii, small diameter, $R = 2.3N^{.62}$

expected to survive rapid change, such as any mobile packet radio network, and any DoD network, the Landmark Hierarchy is an attractive possibility. Ongoing research will further explore this possibility.

1.0 INTRODUCTION

1.1 Background

Switching nodes in computer networks execute routing functions to deliver data from source to destination. These routing functions involve 1) the storage of routing tables (memory overhead), 2) the exchange of routing updates with other nodes (link overhead), and 3) some set of algorithms to compute the routing table entries (CPU overhead). If the routing structure of the network is non-hierarchical, that is, every node maintains a routing entry for every other node in the network, then the overhead caused by the routing function is at best $O(N)$ (McQuillan, Richer, Rosen, 1980), N being the number of nodes in the network.

As networks grow to many thousands of nodes, the routing overhead can consume an unacceptable percentage of network resources. To overcome this problem, a hierarchical routing structure, the area hierarchy, may be used which theoretically can reduce the routing overhead to $HN^{\frac{1}{H}}$ (Kamoun, Kleinrock, 1977), where H is the number of hierarchy levels. The penalty paid for reduced overhead is increased path length, the amount of which depends on the diameter and node degree of the network (small diameter means longer path lengths) but which is typically small. The telephone network is a well-known example of a hierarchical routing structure.[1]

Until now, only one hierarchical routing structure, the area (or cluster) hierarchy, has been developed. This has resulted in a very limited set of design alternatives for large network routing schemes. During the past few years, considerable research has been devoted to the use of area hierarchies in computer networks (Kamoun, Kleinrock, 1977), (Kamoun, Kleinrock, 1979), (Kamoun, Kleinrock, 1980), (Sunshine, 1981), (Perlman, 1985), (Callon, Lauer, 1985), (Khanna, Seeger, 1986), (Sparta, 1986), (Saltzer, Reed, Clark, 1980), (Hagouel, 1983), (Shacham, 1985), (Zakon, 1987). This work has revealed several difficulties in the use of area hierarchies in networks where topology changes are expected (Perlman, 1985), (Sunshine, 1981), (Hagouel, 1983), (Shacham, 1985).

[1] The node degree of an individual node is the number of other nodes it is connected to. The average node degree of a network is the average of all of the individual node degrees. The diameter of a network is the length, here measured in hops, of the longest minimum-length path between any two nodes in the network.

In particular, problems arise in situations where a previously connected area is no longer connected, causing nodes to be unreachable by the routing algorithms even though there are physical paths to those nodes. This is an area partition, or simply, partition. The problem of managing partitions when they do not happen often has been solved (Perlman, 1985). This solution is adequate for most commercial networks, and is used in at least two. However, in military networks such as mobile packet-radio networks, or land-based stationary networks whose topologies may be stable during peace-time but not war-time, this solution is not adequate. A full solution to the problem has been worked on by Shacham. This work shows that dynamic management of an area hierarchy is very difficult. It has not yet been attempted in practice.

1.2 Motivation

This work is supported by the Defense Communications Agency (DCA), and specifically by the Defense Communications System Data Systems organization, which manages the Defense Data Network (DDN). There are two similar problems which have motivated this work.

The DDN operates several long-haul packet switching networks for use by Department of Defense (DoD) subscribers. These networks are growing, and are scheduled to merge in the future. The size of these networks is becoming such that the efficiency of the existing non-hierarchical routing (Shortest Path First (SPF)) is questioned (Sparta, 1986), (Khanna, 1986).

An even more severe problem is that of the growth of the Internet (The DDN plus connected networks, such as the NSFNET). This growth is greater than that of the DDN alone, and is now pushing the limits of the existing gateway routing protocols, the Gateway-to-Gateway Protocol (GGP), and the Exterior Gateway Protocol (EGP) protocol. A further problem here is that, unlike SPF, the gateway routing protocols are not very survivable. For instance, they do not respond to partitions.

1.3 Content

This paper introduces a new hierarchical structure, the Landmark hierarchy. This paper shows that routing overhead due to the Landmark Hierarchy is typically $3\sqrt{N}$, and can be considerably worse for networks with extremely small diameters. The increase in path length for the Landmark Hierarchy is similar to that seen in the area hierarchy. These results show that the Landmark hierarchy is a viable alternative to the area hierarchy for routing in large networks in the sense that it achieves a reduction in network overhead needed to accomplish routing.

However, we believe that the Landmark Hierarchy will be easier to manage dynamically than the area hierarchy. In this paper, we analyze the efficiency of the Landmark Hierarchy in its static state. In other words, we describe and measure the characteristics of the Landmark Hierarchy with regards to path lengths, path distributions, and routing table sizes.

We do not consider ways of dynamically managing the Landmark Hierarchy, although in Section 3.7 we give a brief description of why the Landmark Hierarchy will be simpler to manage than the area hierarchy. While this management function is important, and indeed is the whole motivation behind the invention of the Landmark Hierarchy, it is vital that we first understand everything we can about the Landmark Hierarchy in its steady state. It is of little use to show that the Landmark Routing algorithms perform well as a dynamic routing technique when we have not first shown that the Landmark Hierarchy performs well as a hierarchy. Further, what we learn in this paper may be applied to the development of distributed, dynamic algorithms. Dynamic management of the Landmark Hierarchy, then, will be the topic of subsequent papers.

1.4 Outline

This paper is for readers requiring a general but not detailed knowledge of the Landmark Hierarchy, for designers and implementors needing a detailed working knowledge of the Landmark Hierarchy, and for researchers wanting to extend, improve upon, or refute the work presented in this paper. Those in the first category should read Sections 1, 2, 3, and 8. For those between the casually interested and the strongly interested, Section 7 is recommended. For those still more interested, Section 4 should be read as well. Sections 5 and 6 are for designers and researchers of the Landmark Hierarchy.

Section 2 is an overview of the Area Hierarchy and its major concepts. Some limited performance characteristics are discussed in this section, and further discussed in Section 7.4. Section 3 is an overview of the Landmark Hierarchy and its major concepts. Also in Section 3 is a brief comparison of dynamic management of the area and Landmark Hierarchies. Section 4 provides a detailed description and an analysis of the Landmark Hierarchy. Section 5 describes the simulation techniques used to measure performance of the Landmark Hierarchy. Sections 4 and 5 provide the necessary background needed to fully understand Section 6. Section 6 gives the results and analysis of the simulations. These simulations are designed to isolate the various factors which affect performance of the Landmark Hierarchy. This section gives most of the technical details necessary for understanding the Landmark Hierarchy. Section 7 presents additional simulations which are designed more to reflect real networks than to isolate behavior. Section 7 also uses techniques derived from Section 4 to determine the performance of Landmark Hierarchies of any size. This is the major result of this paper. This Section also contains a comparison of the Landmark Hierarchy and the Area Hierarchy. Section 8 summarizes the major conceptual ideas of Sections 4, 5, and 6. It also gives conclusions about the utility of the Landmark Hierarchy and a discussion of the potential use of the Landmark Hierarchy in the DDN and the DoD Internet.

This paper has three appendices. Appendix A is a glossary of mathematical terms used in this paper. Appendix B contains a set of graphs which describe a particular set of network characteristics important to the analysis of the Landmark Hierarchy—namely, the average number of nodes within x hops of any given node. Appendix C contains a detailed description of the technique we used to generate networks. This technique, based on what we call the loop-span network model, contains improvements we have not seen anywhere else, and so deserves a separate discussion.

2.0 THE AREA HIERARCHY

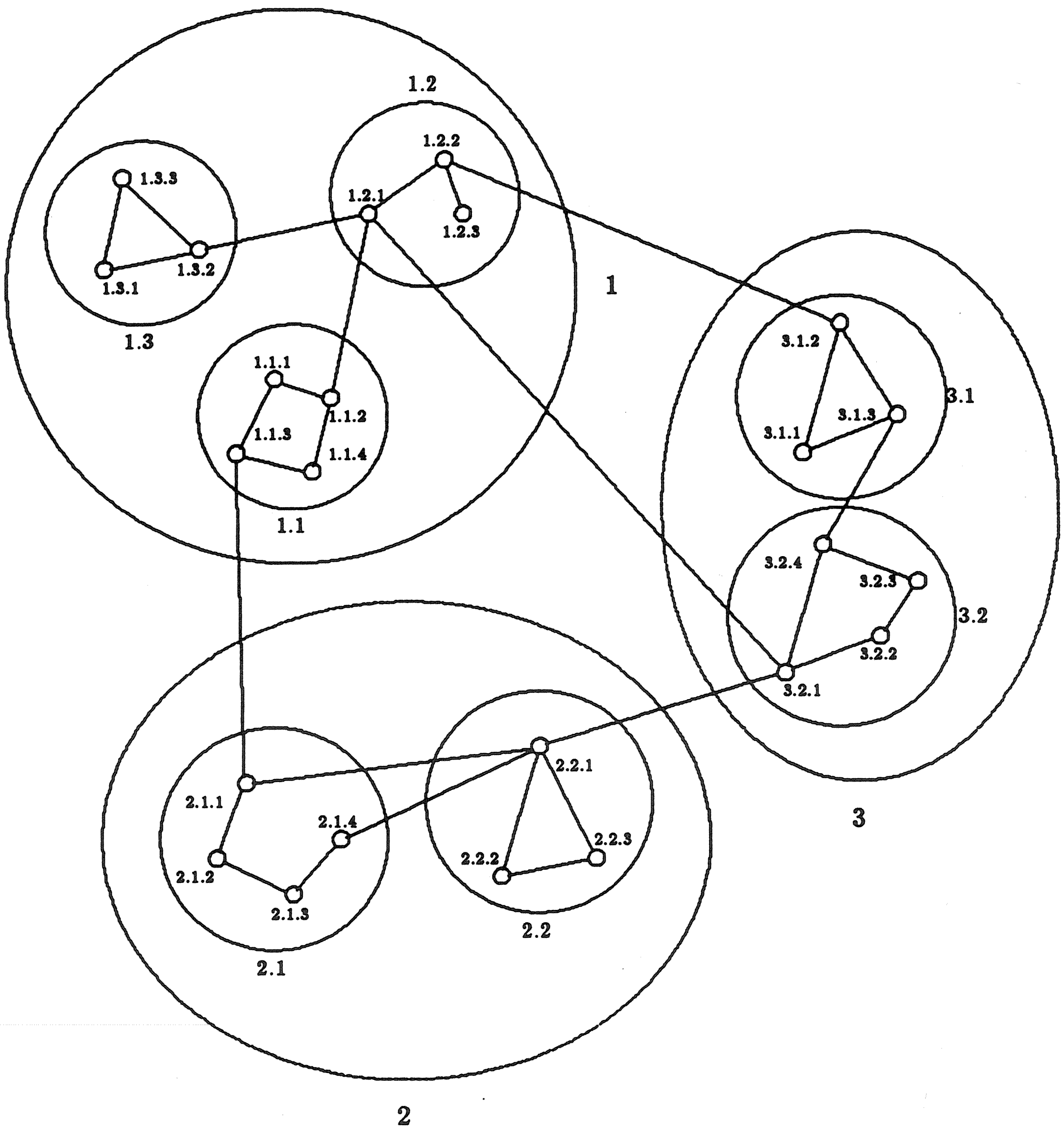
Figure 1 shows a computer network of arbitrary physical topology, that is, the topology does not have an obvious structure to it such as a hierarchy, ring, etc. An area hierarchy has been overlaid on the network of Figure 1. This hierarchy is created by logically grouping nodes into areas, grouping areas into super-areas, and so on.

For the sake of discussion, we will consider a single node in Figure 1 to be a Level 0 area, a group of nodes to be a Level 1 area, and a group of Level 1 areas to be a Level 2 area. The areas in Figure 1 have three-component addresses; the leftmost component refers to the Level 2 area, the middle component refers to the Level 1 area, and the rightmost component refers to the individual node. If an address is written as one digit only (i.e., area 3), then it refers to an entire Level 2 area. Likewise, 2 digit addresses refer to a Level 1 area (i.e. 3.2), and full 3 digit addresses refer to a single node (i.e. 3.2.1).

A constraint on the definition of an area is: a path which does not exit a Level k area must exist between every Level $k-1$ area in the Level k area. This way, once a message to a destination Level $k-1$ area enters the destination's Level k area, the message does not have to leave the Level k area to reach the destination. This allows nodes outside the area to view the area as a single entity. The result is that only one entry is required in that node's routing table to route to several nodes in another area. For instance, in Figure 1, Node 2.1.1 views Nodes 2.2.1, 2.2.2, and 2.2.3 as a single entity, namely, 2.2 — a savings of 3 to 1 in memory overhead (for the table entries) and in link overhead (for the updates required to maintain that entry). Moreover, node 2.1.1 views all nodes in area 1 as a single entity; a savings of 10 to 1.

The penalty paid for this savings is increased path length. For instance, consider a route from source 2.2.1 to destination 1.2.3. By examining the high-order component of the destination address (1.x.x), 2.2.1 determines that 1.2.3 is in a different Level 2 area. The choice available to 2.2.1 is to 1) route directly from its own Level 2 area into Area 1, or 2) to route first into Area 3 and let a node in Area 3 forward the message to Area 1. Having no knowledge about the internal topologies of Areas 3 and 1, 2.2.1 will forward the packet directly to Area 1 via Area 2.1. For this pathological case, the chosen route is nearly twice as long as the shortest possible path.

Figure 1
Area Hierarchy Example



Using mathematical analysis, Kamoun and Kleinrock have shown that, using the area hierarchy, routing overhead of no less than $HN^{\frac{1}{H}}$, where H is the number of hierarchy levels, and N is the number of nodes, can be achieved [Kamoun, Kleinrock, 1977]. Path lengths depend on the nature of the network, but are longer for networks with smaller diameters and larger node degrees. In experiments performed on randomly generated 200 node networks using a 2-level hierarchy, Hagouel found path lengths to be on the order of 15% longer than shortest path [Hagouel, 1983]. Using different routing and clustering techniques on randomly generated 200 node networks, Callon found path lengths to be no greater than 15% over shortest path, and on the average, about 5% over shortest path [Callon, Lauer, 1985]. These path increases are acceptable considering the savings in routing overhead.

There are many possible variations of an area hierarchy. Nodes may choose to keep selective information about the internal structures of other areas, or areas may be allowed to overlap [Shacham, 1985]. Use of these variations involve tradeoffs of their own, and must be considered on a case by case basis.

In this paper, we do not study the area hierarchy. We refer the reader to the literature. Except for a few isolated cases, we compare our results against those of shortest path. The reason for this is that, since there are so many possible variations on the area hierarchy, it would be too difficult to do a fair and thorough comparison of the Landmark Hierarchy to the area hierarchy.

3.0 THE LANDMARK HIERARCHY

In this section, we describe the Landmark hierarchy. We do this by first describing the Landmark itself. Then, we describe a hierarchical structure built from Landmarks. Thirdly, we describe how nodes are addressed in a Landmark hierarchy. Finally, we show how routing may take place with the Landmark hierarchy.

3.1 The Landmark

The description of a Landmark is very simple. A Landmark is a node whose neighbor nodes within a certain vicinity contain routing entries for that node. Determination of the vicinity based on hops; that is, the distance between any two nodes that share a link is one.[2]

As an example, consider Node 1 in the network of Figure 2. Nodes 2 through 6 have routing entries for Node 1 (as indicated by the arrowheads) and are therefore able to forward any packets addressed for Node 1 to Node 1. Nodes 7 through 11 do not contain routing entries for Node 1. Therefore, Node 1 is a Landmark which can be "seen" by all nodes within a distance of 2 hops. We refer to Node 1 as a Landmark of Radius 2. In general, a node for which all nodes within r hops contain a routing entry is a Landmark of radius r.[3]

In this case, the Landmark vicinity is expressed in "circular" terms—thus the radius. In general, the vicinity may not be circular (that is, may not be expressed in terms of a radius), but may have a more complex shape. As we will see in Section 4, a set of nodes outside the radius may also be included in the Landmark vicinity. For the sake of convenience, we usually refer to Landmarks as having a certain radius, and ignore the possibility of additional nodes unless otherwise stated.

3.2 The Landmark Hierarchy

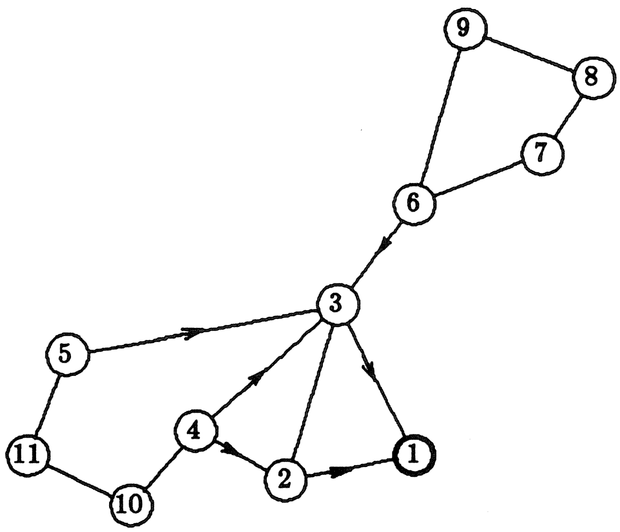
Next, let us consider a hierarchy built from Landmarks. The nomenclature LM_i

[2]The distance between two nodes is the length (here in hops) of the shortest path between those two nodes.

[3]Appendix A contains a glossary of the mathematical terms used in this paper.

[4]The notation here varies slightly. In general, we use id generically to mean any single Landmark. When it is necessary to distinguish between Landmarks in the same expression or sentence, we will use either id_x, id_y, or simply x, y. The latter form is usually in reference to a

Figure 2
A Single Landmark



refers to a Landmark of hierarchy level i, $i=0$ being the lowest level, and $i=H$ being the highest level. Throughout this paper, the term i is reserved to mean a hierarchy level. The nomenclature $LM_i[id]$ refers to a specific LM_i with label id, called the Landmark ID.[4] Each $LM_i[id]$ has a corresponding radius $r_i[id]$. In the Landmark hierarchy, every node in a network is a Landmark $LM_0[id]$ of some small radius $r_0[id]$. Some subset of $LM_0[id]$'s are $LM_1[id]$'s with radius $r_1[id]$, and with $r_1[id]$ almost always greater than $r_0[id]$, such that there is at least one $LM_1[id]$ within $r_0[id]$ hops of each $LM_0[id]$. Likewise, a subset of the $LM_1[id]$'s are $LM_2[id]$'s, with $r_2[id]$ almost always greater than $r_1[id]$, such that there is at least one $LM_2[id]$ within $r_1[id]$ hops of each $LM_1[id]$. These iterations continue until a few nodes are $LM_i[id]$'s each with an $r_i[id]$, with $r_i[id] \geq D$, D being the diameter of the network. The reason for this structure will become clear in Section 3.5.[5]

figure or specific example.

[5]The network diameter is the distance (here in hops) between the two nodes in the network furthest from each other.

Figure 3 illustrates the Landmark hierarchy by showing a portion of a network. This is a two-dimensional representation (meaning that only nodes drawn very close to each other will share a link). For simplicity, only four of the nodes are shown, and no links are shown. The dotted arrows and circle indicate the radius of the Landmarks; that is, the vicinity within which nodes contain routing entries for that Landmark. For instance, every node within the circle defined by $r_1[b]$ has an entry for, and can route to, $LM_1[b]$. Since nodes may be Landmarks at several levels, a node may have several Landmark IDs, one for each level. Again for simplicity, the nodes in Figure 3 are labeled only with the Landmark IDs which are pertinent to the examples herein.

In general, Landmark IDs only need to be locally unique, except at the highest level. It is beyond the scope of this paper to define what constitutes local uniqueness, or to discuss the issues surrounding the use of locally unique Landmark IDs. In this paper, we assume that all Landmark IDs are globally unique.

3.3 Routing Table

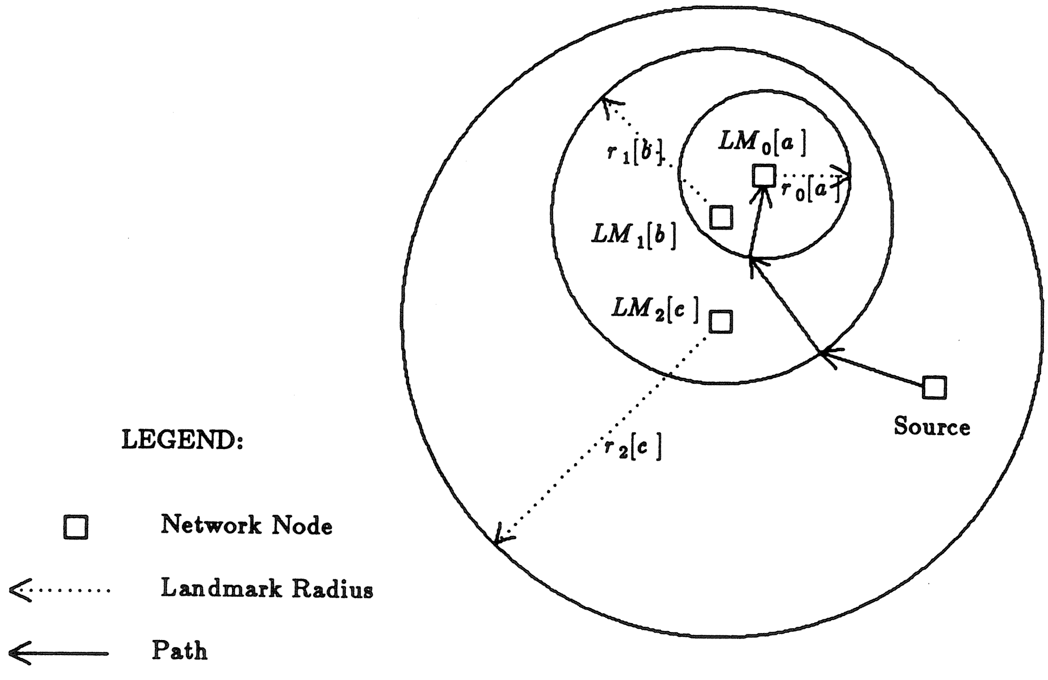
Each node in the network keeps a table of the next hop on the shortest path to each Landmark it has routing entries for. Each node will therefore have entries for every $LM_0[id]$ within a radius of $r_0[id]$, every $LM_1[id]$ within a radius of $r_1[id]$, and so on.

Since every node is an LM_0, and since every node has entries for every $LM_0[id]$ within a radius of $r_0[id]$, every node has full knowledge of all the network nodes within the immediate vicinity. Likewise, since a portion of all LM_0 are LM_1, every node will have knowledge of a portion of the network nodes further away. Similarly, each node will have knowledge of even fewer nodes further still, and so on. The result is that all nodes have full local information, and increasingly less information further away in all directions. This can be contrasted with the area hierarchy where a node on the border of an area may have full local information in the direction within the border, but virtually no local information in the direction across the border.

3.4 Addressing in a Landmark Hierarchy

In an area hierarchy, the address of a node is a reflection of the area(s) at each hierarchical level in which the node resides. The telephone number is a well-known example of this. In a Landmark hierarchy, the address of a node is a reflection of the

Figure 3
Landmark Hierarchy



Landmark(s) at each hierarchical level which the node is near. The Landmark Address, then, is a series of Landmark IDs: $LM_0[id_a].LM_1[id_b].\ .\ .\ .\ LM_z[id_z]$.

There are two constraints placed on Landmark Addresses. First, the Landmark represented by each address component must be within the radius of the Landmark represented by the next lower address component. For instance, the node labeled $LM_0[a]$ in Figure 3 may have the Landmark Address $LM_2[c].LM_1[b].LM_0[a]$. The address of the node labeled $LM_0[a]$ could be $LM_2[c].LM_1[e].LM_0[a]$ if and only if there existed a Landmark $LM_1[e]$ (not shown) which was within the the radius of the node labeled $LM_0[a]$. The reason for this constraint will become clear in Section 3.5. Since more than one Landmark may be within the radius of a lower level Landmark, nodes may have a multiplicity of unique addresses. Multiple addresses could be used to improve survivability and provide some traffic splitting (Tsuchiya, 1987).

3.5 Routing in a Landmark Hierarchy

Now we may consider how routing works in a Landmark Hierarchy. How can we find a path from the node labeled Source to the node labeled $LM_0[a]$ in Figure 3? The Landmark Addresses for the node labeled $LM_0[a]$ is $LM_2[c].LM_1[b].LM_0[a]$. To find the path Source will look in its routing tables and find an entry for $LM_2[c]$ because Source is within the radius of $LM_2[c]$. Source will not, however, find entries for either $LM_1[b]$ or $LM_0[a]$, because Source is outside the radius of those Landmarks. Source will choose a path towards $LM_2[c]$. The next node will make the same decision as Source, and the next, until the path reaches a node which is within the radius of $LM_1[b]$. When this node looks in its routing tables, it will find an entry for $LM_1[b]$ as well as for $LM_2[c]$. Since $LM_1[b]$ is finer resolution, the node will choose a path towards $LM_1[b]$. This continues until a node on the path is within the radius of $LM_0[a]$, at which time a path will be chosen directly to $LM_0[a]$. This path is shown as the solid arrow in Figure 3.

There are two important things to note about this path. First, it is, in general, not the shortest possible path. The shortest path would be represented in Figure 3 by a straight line directly from Source to $LM_0[a]$. This increase in path length is the penalty paid for the savings in network resources which the Landmark hierarchy provides. This will be analyzed in Section 4.

The other thing to note is that often the path does not necessarily go through the Landmarks listed in a Landmark Address. This happens more frequently if the Landmark vicinity for an LM_i goes well beyond an LM_{i+1}, in other words, if the vicinity overage is large. This is an important reliability consideration in that a Landmark may be heavily congested or down, and yet a usable path may be found using that Landmark (or, more literally, using previous updates received from that Landmark).

3.6 Landmark Hierarchy Example

To better illustrate the Landmark Hierarchy, Figure 4 shows the same network of Figure 1 with a Landmark Hierarchy rather than an Area Hierarchy. This network has 3 hierarchical levels. All nodes (small circles) are LM_0. LM_1 are denoted by a diamond, and

LM_2 by a large circle. The rightmost address component is the $LM_0[id]$, and is unique for each node in the network. The middle address component is the $LM_1[id]$ and indicates proximity to an LM_1, and the leftmost address component is an $LM_2[id]$, indication proximity to an LM_2. All $r_0 = 2$ hops, all $r_1 = 4$ hops, and all $r_2 = 8$ hops.

Figure 4
Landmark Routing Example

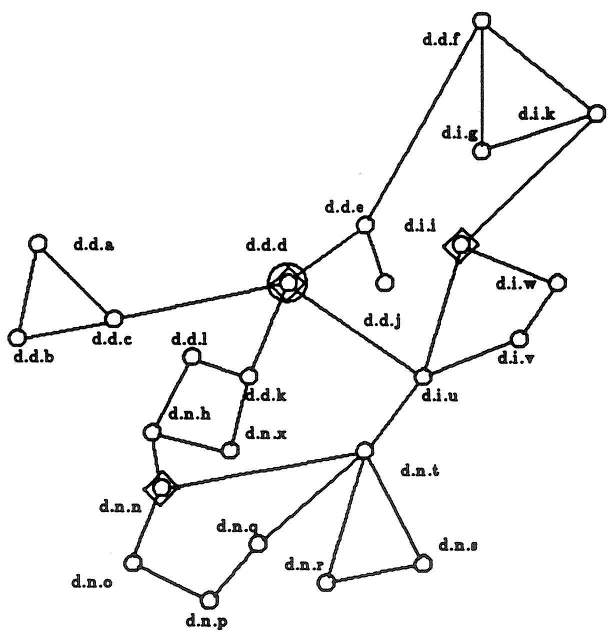


Table 1 shows the routing table for Node *g* in Figure 4. This length of this table has been optimized by including only one entry per node, even if that node is a Landmark at several different levels. Node *g* has less than one-fourth of the total network nodes in its routing table.

Let us consider a routing example where Node *g (d.i.g)* is routing a message to Node *t (d.n.t)*. Node *g* examines Node *t*'s Landmark Address—*d.n.t*—and does not find entries for either $LM_0[t]$ or $LM_1[n]$ in its routing table. Node *g* does, however, have an entry for

Table 1
Routing Table for Node *g* of Figure 4

Landmark	*Level*	*Next Hop*
$LM_2[d]$	*2*	*f*
$LM_1[i]$	*1*	*k*
$LM_0[e]$	*0*	*f*
$LM_0[k]$	*0*	*k*
$LM_0[f]$	*0*	*f*

$LM_2[d]$, and therefore forwards the message towards $LM_2[d]$ via Node *f*. Node *f* also does not have entries for $LM_0[t]$ or $LM_1[n]$, and therefore forwards the message towards $LM_2[d]$ via Node *e*. Node *e* does have an entry for $LM_1[n]$ (but not $LM_0[t]$), and forwards the message towards $LM_1[n]$ via Node *d*. Node *d* does have an entry for $LM_0[t]$, as does Node *u*, and the message is delivered. The resulting path, *g—f—e—d—u—t*, is 5 hops, 1 hop longer than the shortest path, *g—k—i—u—t*.

3.7 Dynamic Management of the Landmark Hierarchy

Although in this paper we do not study the problem of dynamic management of the Landmark Hierarchy, we say a few words about it here to give the reader a feel for why dynamic management of the Landmark Hierarchy should be easier than for the Area Hierarchy.

We first describe the process of building an area hierarchy from scratch. The first step is to group nodes into their lowest level areas. This is too difficult to do in a completely distributed fashion, and so "leaders" (nodes which will temporarily take charge of creating areas) are elected. The number of nodes elected will be approximately equal to the number of areas expected at that level. These election algorithms are well-understood and fairly straight forward (Garcia-Molina, 1982). A group of nodes are given an ordering, say by assigning numbers and numerically ordering them. Then, the node which is first in the ordering among the group of nodes running the election is elected.

The leaders then begin to form a group of nodes around them which will be in their area. They will do this according to some criteria: that there should be at least so many nodes in their area, and that there should be a certain richness of connectivity in their area. Sometimes these criteria may result in two leaders wanting the same node in their area, or in there being some nodes not wanted by any leader. These problems must be resolved. When these areas are formed, then the leaders again run an election, picking super-leaders, who may then take charge of forming the next level areas, and so on.

When an area becomes partitioned, this state must be recognized by the nodes in both partitions. Then a decision must be made whether to fold the partitions into other areas, to fold one of the partitions into other areas, or to create a new area. This decision will require the election of a new leader (in the partition without a leader), and will require that the leaders collect information about the partition and about other adjacent areas in order to make the decisions. As always, these decisions must be coordinated with the rest of the nodes in the partitions and with other leaders.

To create a Landmark Hierarchy from scratch, the Landmarks at Level 1 must first be elected. This is analogous to the election of leaders in the area hierarchy. However, once this election is completed, there is nothing left to to but start the next election for the next level—it is not necessary to form areas. By listening to broadcasts from the Landmarks, non-Landmark nodes are able to determine which Landmark they are closest to *independently* of what other nodes decide. Also by listening to broadcasts, nodes are able to determine if there are enough Landmarks and run another election if there are not, and Landmarks are able to determine what their radii should be. To illustrate how robust the Landmark Hierarchy is, we built Landmark Hierarchies in some simulations by simply randomly assigning Landmarks according to a probability scale. In a distributed environment, this is like a node arbitrarily deciding for itself whether it should be a Landmark at a certain level. These hierarchies performed very nearly as well as those created through an election algorithm!

The equivalent of a partition in the Landmark Hierarchy is when an LM_{i+1} falls outside the vicinity of an LM_i due to changes in topology. In this case, the LM_{i+1} can locally start another election algorithm to build back the hierarchy, or the LM_i can simply lengthen its radius.

3.7.1 Changing Addresses

Although we do not address this issue in this paper, we should mention that one of the inherent problems with flexibly and automatically building hierarchies (area or Landmark) is that addresses can change. There must therefore be a mechanism to bind the identification of network elements (nodes, hosts, whatever) to the addresses of those elements. Obviously, it does no good to have a self-adjusting hierarchy if the method of updating addresses floods the network with packets, thus negating any savings generated by the hierarchy.

In a related effort, we have found a suitable solution to this problem (Stine, Tsuchiya, 1987), (Tsuchiya, 1987).

4.0 ANALYSIS OF LANDMARK HIERARCHY

Now that we have described the fundamental ideas behind the Landmark Hierarchy, we can analyze the Landmark Hierarchy in detail. We are interested in studying three aspects of the Landmark Hierarchy:

1. routing tables size R,
2. path length P, and
3. fairness of path distribution F.

The routing table size gives an indication of the cost in network resources of the Landmark Hierarchy. The routing table size gives a direct indication of memory required by nodes. It indirectly gives an indication of 1) link usage, because we assume that the number of updates a node receives will be proportional to the size of its routing table, and 2) processor usage, because we assume that the amount of processing a node must do is also proportional to its routing table size. We expect the size of the routing table in the Landmark Hierarchy to be less than that of a non-hierarchical routing scheme. In other words, the Landmark Hierarchy results in a cost savings of network resources over the non-hierarchical routing scheme.

The path length also gives an indication of the cost in network resources of the Landmark Hierarchy. A path length longer than shortest path results in more links and nodes being utilized for a given amount of traffic. This causes increased memory (to buffer traffic), link (to carry traffic), and processor (to process traffic) usage. We expect path lengths in the Landmark Hierarchy to be longer than those in a non-hierarchical routing scheme, resulting in an increased cost of network resources. For many networks, we expect the cost decrease due to smaller routing table size to more than compensate the cost increase due to longer path length, resulting in a net improvement in overall network performance.

Also of interest is the distribution of paths chosen by Landmark routes. Paths should be evenly and fairly distributed so as not to cause unnecessary congestion in parts of the network. We explore several techniques to minimize uneven path distribution.

4.1 Describing the Landmark Hierarchy

In this section, we make a few observations about the Landmark Hierarchy and write down some additional nomenclature. This will aid us in describing how we specify the Landmark Hierarchy in the following section.

First, we know the following things from the description of the Landmark Hierarchy given in Section 3.

1. Every node is an LM_0. This means that $T_0 = N$, and all $d_0(id) = 0$. T_i is the total number of LM_i in the network. The parameter $d_i(id)$ is the distance in hops from Node id to the nearest LM_i. The parameter $d_i(id_x \rightarrow id_y)$ is the distance in hops from Node id_x to $LM_i[id_y]$. In this paper, we assume that the function $d_i(id_x \rightarrow id_y)$ is communicative. That is, $d_i(id_x \rightarrow id_y) = d_i(id_y \rightarrow id_x)$. This is possible because we assume full-duplex links with a metric of one in each direction. This assumption will not invalidate our results when we introduce non-communicative general metrics into the links because the structure of the hierarchy, even an operating network, is always determined by hops. General metrics are used to direct traffic flow once the hierarchy is established.

2. All $r_H[id] \geq D$. In other words, the radius of the highest level Landmarks must be at least equal to the diameter of the network. (Physically speaking, it cannot be greater than the diameter of the network. However, we can *specify* it to be greater than the diameter of the network.)

3. For an $LM_i[id_a]$ and an $LM_{i+1}[id_b]$ to be part of the same Landmark Address, the $LM_{i+1}[id_b]$ must be within $r_i[id_a]$ of the $LM_i[id_a]$, or,

   $$d_i(id_b \rightarrow id_a) \leq r_i[id_a]. \qquad 1$$

   In general, if a given node $LM_0[id_a]$ has a Landmark Address of

   $$LM_H[id_z], LM_{H-1}[id_y], \ldots\ldots, LM_0[id_a],$$

   then

   $$d_0(LM_1[id_b] \rightarrow id_a) \leq r_0[id_a], \; d_1(LM_2[id_c] \rightarrow LM_1[id_b]) \leq r_1[id_b], \; \ldots\ldots, \; d_H(LM_H[id_z] \rightarrow LM_{H-1}[id_y]) \leq r_{H-1}[id_y].$$

It follows, then, that we don't know the following:

1. The number of Landmarks at each level greater than 0, $(T_i, \; i = 1,2,3,\ldots,H)$.

2. The number of Landmark levels H.

3. The radii of the landmarks at levels below level H $(r_i[id],\ i = 0,1,2,.....,H-1)$,

4. The distance from nodes to Landmarks above level 0 $(d_i(id),\ i = 1,2,3,......,H)$.

5. The set of $LM_i[id]$ for $i = 1,2,3,....,H$.

We can, however, make the following general observations about these parameters.

First, there is a direct relationship between r_i, the average LM_i radius, and d_i, the average distance from all nodes to their closest LM_i, where

$$r_i = \frac{\sum_{z=1}^{T_i} r_i[id_z]}{T_i} \qquad d_i = \frac{\sum_{z=1}^{N} d_i(id_z)}{N}.$$

(When we discuss these parameters in the context of our simulation experiments, where we take the average over many simulations, we use the same notation. In general, the context determines the set of elements over which the averaging applies. This notation is also sometimes used to refer to the radii of LM_i when all LM_i have the same radii. In this case the $[id]$ identifier is not needed.) Certainly, if LM_i have large r_i, nodes can be far away from them, and hence d_i can be large.

Second, H, the number of Landmark levels, is related to D and r_i. For instance, consider a network where $r_4 = 16$ If $D = 100$, then certainly H must be greater than 4. However, if $D = 16$, then H need not be greater than 4, because the radius $r_4 = 16$ would cover the entire network.

Third, there is a direct relationship between the size of the routing tables R_i, and the number of Landmarks T_i and the average Landmark radius r_i. Clearly, if there are either many LM_i in a network (large T_i), or if their r_i are large, then nodes will have more entries in their routing tables.

In particular, let $v(x)$ be a function which determines the number of nodes x hops or closer to a given node. Then, on the average, every LM_i will have $v(r_i)$ nodes within its Vicinity, and each of those nodes will have a routing entry for that LM_i. There will therefore be $T_i\, v(r_i)$ routing entries in the network, and on the average

$$R_i = \frac{T_i \, v(r_i)}{N} \qquad 2$$

routing entries in each node for each level i. While $v(r_i)$ is clearly dependent on r_i, a general solution for $v(r_i)$ is currently not known (Bollabas, 1985). (While the solution for $v(r_i)$ is known for regular structures such as a ring or lattice topology, analysis using these topologies was not useful as it obscured important characteristics of the Landmark Hierarchy which were discovered through simulation. Therefore, those analyses are not presented.)

Fourth, there is an inverse relationship between T_i, the number of LM_i in a network, and d_i. Consider that for each LM_i there are, on the average, $\frac{N}{T_i}$ nodes closer to that LM_i than any other LM_i. Then there will exist some $\hat{d}_i$ for which

$$v(\hat{d}_i) = \frac{N}{T_i}. \qquad 3$$

Clearly, $\hat{d}_i$ and d_i are related. If the average distance to an LM_i is small, then the average maximum distance from a node to an LM_i must also be small.

Finally, if we combine Equations 2 and 3, we get

$$R_i \approx \frac{v(r_i)}{v(d_i)}. \qquad 4$$

This shows that routing table sizes depend on the radius of the Landmarks, and the density of Landmarks in the network. In the following sections, we will consider several ways to specify the Landmark Hierarchy which will affect both r_i and d_i.

Clearly, from Equation 4 alone, we will not be able to determine the routing table sizes we might expect for some given network because we have no solution for $v(x)$. We must therefore experimentally determine R_i. Worse still, we are not able to analytically say *anything* about the path lengths P_i or the path distributions F_i. (This is not to mean that nothing analytic can be said. We were unable, however, to come up with anything meaningful.) Therefore, everything we learn about P_i and F_i will be through experimentation.

4.2 Specifying the Landmark Hierarchy

By determining for each node id in the network its $LM_i[id]$ and its corresponding Vicinity, we completely specify the Landmark Hierarchy. For the case where the Vicinity is circular, then the parameter $r_i[id]$ is adequate to define a Landmark's Vicinity. Unless otherwise stated, the use of the term $r_i[id]$ will imply a circular Landmark Vicinity. The question then becomes; how do we pick the $LM_i[id]$ and their corresponding $r_i[id]$?

There are several ways of picking the $LM_i[id]$ and their $r_i[id]$. It is important that we consider several ways because the different ways exhibit different characteristics with regards to routing table sizes R, path lengths P, and path distribution F. The method of specification also affects how the Landmark Hierarchy may be automatically generated in a distributed network.

In this paper, we consider two basic approaches to picking the $LM_i[id]$.

1. Place limits on the values of $r_i[id]$ and $d_i(id)$, then choose the LM_i within the constraints placed by those limits.

2. Randomly pick the LM_i based on a given T_i, thus determining $d_i(id)$, then determine the $r_i[id]$.

Both of these approaches need further explanation. However, it is clear that both of the approaches require knowledge or the relationship between $r_i[id]$ and $d_i(id)$. We therefore first consider this relationship in some detail before discussing the two approaches.

4.2.1 Relationship between Landmark Radii and Landmark Distances: Case 1

In this section, we establish the basic relationship between $r_i[id]$ and $d_i(id)$, and then develop that relationship by considering several constraints which may be placed on $r_i[id]$ and $d_i(id)$.

In Equation 1, an expression for $d_i(a \rightarrow b)$ for an $LM_{i+1}[a]$ with respect to an $LM_i[b]$ is given in terms of $r_i[b]$. We are, however, interested in expressing $d_i(a \rightarrow b)$ for any node a with respect to an $LM_i[b]$ in terms of $r_i[b]$. This is important because ultimately we wish to use the Landmark Hierarchy in an dynamic network environment where the hierarchy is generated in a distributed fashion on the fly. If we can express the Landmark Hierarchy in terms of what any node knows about *its own* environment, such as its distance from the various Landmarks, then we can allow nodes to make decisions concerning the hierarchy *by themselves*, thus reducing the amount of coordinated decision making needed in the network.

Figure 5 illustrates two extreme cases for values of d_i, in this case $d_2(a)$. In Figure 5, node $LM_0[a]$ requires the use of $LM_1[b]$, and either $LM_2[c]$ or $LM_2[d]$ for its Landmark Address. $LM_1[b]$ is some distance $d_1(a \rightarrow b) \leq r_0$ from $LM_0[a]$, in accordance with Equation 1. In the extreme cases, either 1) the LM_2 will be in the opposite direction from $LM_0[a]$ as $LM_1[b]$, as is $LM_2[c]$, or 2) the LM_2 will be in the same direction from $LM_0[a]$ as $LM_1[b]$, as is $LM_2[d]$. In the first case, to satisfy the inequality $d_1(c \rightarrow b) \leq r_1[b]$, we must have

$$d_2(a \rightarrow c) \leq r_1[b] - d_1(a \rightarrow b).$$

This can be written in terms of a maximum allowable value of $d_2(a \rightarrow c)$ as:

$$d_2^{\max}(a \rightarrow c) = r_1[b] - d_1(a \rightarrow b). \qquad 5$$

In the second case, we must have

$$d_2^{\max}(a \rightarrow c) = r_1[b] + d_1(a \rightarrow b). \qquad 6$$

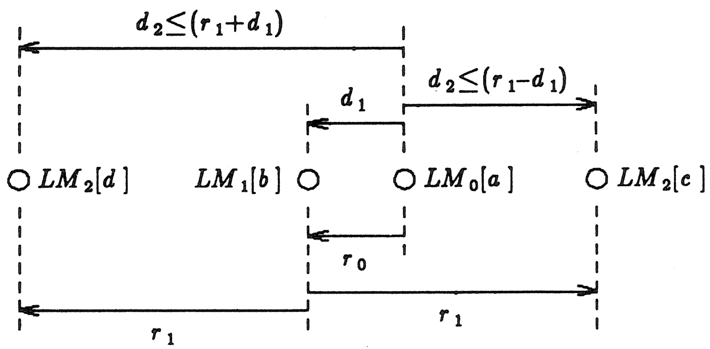
In the case where $d_1(a \rightarrow b) = r_0[a]$, 6 becomes $d_2^{\max}(a \rightarrow c) = r_1[b] + r_0[a]$. This can be generalized to give the absolute maximum distance a node whose address is $LM_i[id_z].LM_{i-1}[id_y]$..........$LM_0[id_a]$ can be from its LM_i:

$$d_i^{upper\ bound}(id_a \rightarrow id_z) = r_{i-1}[id_y] + r_{i-2}[id_x] + \ldots\ldots + r_0[id_a]. \qquad 7$$

This places an upper limit on the value of d_i. It is interesting to note that if $r_i[z] < r_{i-1}[y] + r_{i-2}[x] + \ldots\ldots + r_0[a]$, $(1 \leq i \leq H-1)$, then a node could be using an $LM_i[id_i]$ which it did not have in its routing table as part of its Landmark Address. In fact, it is entirely possible that a node may have 0 entries in its routing table for levels of i, $1 \leq i \leq H-1$.

The lower bound on $d_i(node \rightarrow LM_i[id])$ is the trivial case where the node is the LM_i. In this case, $d_i(node \rightarrow LM_i[node]) = 0$.

Figure 5
Relationship Between Landmark Radii and Landmark Distance: Case 2



In Equations 5 and 6, the directions of $LM_2[c]$ and $LM_2[d]$ with respect to $LM_1[b]$ and $LM_0[a]$ are known. We are interested in finding d^{max} for the more general case where this is not known. In this case, we must assume the smaller value of d^{max}, namely that given in Equation 5. For instance, consider Figure 6, which shows $LM_3[e]$ in the opposite direction from $LM_0[a]$ as $LM_2[c]$. The maximum allowable distance from $LM_0[a]$ to $LM_3[e]$ is $d_3^{max}(a \rightarrow e) = r_2[c] - d_2(a \rightarrow c)$. In general,

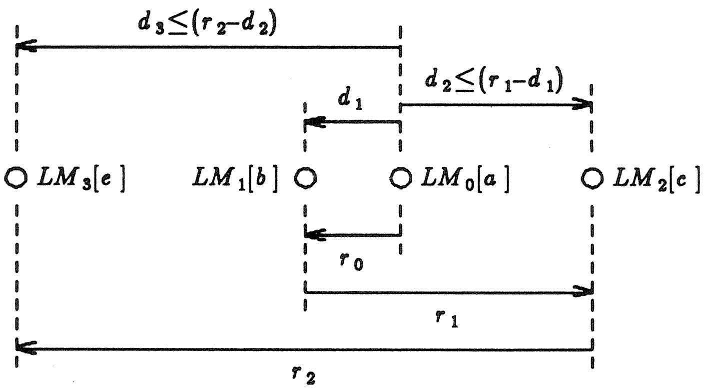
$$d_i^{max}(node \rightarrow LM_i[id_x]) = r_{i-1}[id_y] - d_{i-1}(node \rightarrow LM_{i-1}[id_y]) . \qquad 8$$

In Equation 8, the $d_i(id)$ are known. If they are not known, we must assume the maximum value for $d_i(id)$. Then, $d_1(a \rightarrow b)$ in Figure 6 becomes $r_0[a]$, and $d_2^{max}(a \rightarrow c) = r_1[b] - r_0[a]$. In general,

$$\begin{aligned} d_i^{max}(node \rightarrow LM_i[id_x]) &= r_{i-1}[id_y] - r_{i-2}[id_z] + \ldots\ldots + r_0[node] \quad (i \ odd) \\ d_i^{max}(node \rightarrow LM_i[id_x]) &= r_{i-1}[id_y] - r_{i-2}[id_z] + \ldots\ldots - r_0[node] \quad (i \ even) \end{aligned} \qquad 9$$

Now, we may consider in detail the approaches to specifying the Landmark Hierarchy.

Figure 6
Relationship between Landmark Radii and Landmark Distance: Case 2



4.2.2 Landmark Hierarchy Specification Approach 1—Limit Landmark Radii and Landmark Distances

In this approach, we place limits on the values of $r_i[id]$ and $d_i(id)$, then choose the LM_i within the constraints placed by those limits. There are several ways to place these limits.

4.2.2.1 Single Value for All Landmark Radii and Landmark Distances. The first way is to set both $r_i[id]$ and $d_i^{max}(id)$ to single values for all nodes in the network. In other words, $r_i[id_a] = r_i[id_b] = \ldots\ldots\ldots = r_i[id_z]$ and $d_i^{max}(id_a) = d_i^{max}(id_b) = \ldots\ldots\ldots = d_i^{max}(id_z)$. In terms of picking the $LM_i[id]$, this way is the simplest because the only information needed to pick an $LM_i[id]$ is $d_i(id)$. In other words, for a Node id to decide if it should be an $LM_i[id]$, it is only necessary for it to determine if its distance to its nearest LM_i is less than or equal to the chosen d_i^{max}. If $d_i(id) \leq d_i^{max}$, then Node id does not need to become an LM_i. If $d_i(id) > d_i^{max}$, then Node id or some other node within d_i^{max} hops must become an LM_i.

For determining the r_i and d_i^{max}, we may either 1) pick all of the r_i and derive the resulting d_i^{max}, or 2) we may pick all of the d_i^{max}, and derive the resulting r_i. For Case 1, we can generalize Equation 9 to

$$d_i^{\max} = r_{i-1} - r_{i-2} + \ldots\ldots + r_0 \qquad (i\ odd)$$
$$d_i^{\max} = r_{i-1} - r_{i-2} + \ldots\ldots - r_0 \qquad (i\ even) \qquad 10$$

For Case 2, we can generalize equation 8 to

$$d_i^{\max} = r_{i-1} - d_{i-1}^{\max} \ .$$

From this, we derive the expression for r_i as

$$r_i = d_{i+1}^{\max} + d_i^{\max} + \ldots\ldots\ldots\ldots + d_1 \ . \qquad 11$$

As an example, assume we pick the r_i as $r_0 = 2$, $r_1 = 4, \ldots\ldots\ldots, r_i = 2^{i+1}$. Then $d_1^{\max} = 2$ (remember that $d_0 \equiv 0$), $d_2^{\max} = 2$, $d_3^{\max} = 6$, $d_4^{\max} = 10$, and

$$d_i^{\max} = \begin{cases} \dfrac{2^{i+1} + 2}{3} & (i\ odd) \\ \dfrac{2^{i+1} - 2}{3} & (i\ even) \end{cases} .$$

Similarly, for $r_i = (i+1)^2$, $d_i^{\max} = \dfrac{i^2 + i}{2}$, and for $r_i = i+1$, $d_i^{\max} = \lfloor \dfrac{i}{2} \rfloor$.

4.2.2.2 Single Value for All Landmark Radii, but let the Landmark Distances Vary. The second way to place constraints on $r_i[id]$ and $d_i^{\max}(id)$ is to define single values for all r_i, but let $d_i^{\max}(id)$ vary depending on $d_{i-1}(id)$. In terms of picking the $LM_i[id]$, this method requires knowledge of both the $d_i(id)$ and the $d_{i-1}(id)$. In other words, for a Node id to decide if it should be an $LM_i[id]$, the node must know both its distance to the nearest LM_i and its distance to the nearest LM_{i-1}. For cases where $d_{i-1}(id)$ is small, $d_i(id)$ may be larger. This allows d_i to be larger, thus resulting in fewer LM_i (smaller T_i) and ultimately smaller routing tables (smaller R_i).

For this case, we generalize Equation 8 to

$$d_i^{\max}(node \rightarrow LM_i[id_x]) = r_{i-1} - d_{i-1}(node \rightarrow LM_{i-1}[id_y]) \ . \qquad 12$$

However, the last term here may vary from 0 to $d_{i-1}^{\max}(node \rightarrow LM_{i-1}[id_y])$. In general, any $d_i^{\max}$ may be as large as that given in Equation 10, or as small as $d_i^{\max} = r_{i-1}$.

4.2.2.3 Single Value for All Landmark Distances, but let the Landmark Radii Vary. The third way to place constraints on $r_i[id]$ and $d_i^{max}(id)$ is to vary $r_i[id]$, but fix d_i^{max} over all i. This is done by picking the LM_i as though both r_i and d_i^{max} were fixed, and then varying $r_i[id]$ according to $d_{i+1}(id)$ and either 1) some Landmark Overage constant $o_i[id]$ which determines how many hops beyond the nearest LM_{i+1} the $r_i[id]$ should extend, or 2) some Landmark Coverage constant $c_i[id]$ which determines how many LM_{i+1} the $r_i[id]$ should cover. In terms of picking the $LM_i[id]$, this method requires that each node has knowledge of $d_{i+1}(node \rightarrow c_i^{th}\ closest\ LM_{i+1}[id_s])$ so that the node may set its $r_i[id]$. Since typically $r_i[id]$ is reduced when it is modified, this method results in smaller R_i. In this approach, Equations 10 and 11 apply, with the addition of an equation to express the modified $r_i[id]$ as:

$$r_i[node] = d_{i+1}(node \rightarrow c_i^{th}\ closest\ \ LM_{i+1}[id_s]) + o_i \quad . \qquad 13$$

4.2.2.4 Vary Both the Landmark Radii and the Landmark Distances. The fourth way to place constraints on $r_i[id]$ and $d_i^{max}(id)$ is let both the $r_i[id]$ and the $d_i(id)$ vary. This is done by picking the $LM_i[id]$ as though there were a single value for r_i, but with varying $d_i(id)$ as in the second method described above, and then modifying the $r_i[id]$ as in the third method described above. In terms of picking the $LM_i[id]$, this method requires that each node has knowledge of both the $d_i(id)$ and the $d_{i-1}(id)$, plus knowledge of $d_{i+1}(node \rightarrow c_i^{th}\ closest\ LM_{i+1}[id_s])$ so that the node may set its $r_i[id]$. Here, equations 12 and 13 apply.

4.2.3 Landmark Hierarchy Specification Approach 2—Randomly Pick the Landmarks

In this approach, we randomly pick the LM_i based on a given T_i, thus determining $d_i(id)$, then determine the $r_i[id]$. In this approach, the $d_i(id)$ are automatically determined when the $LM_i[id]$ are picked. The $r_i[id]$ are then determined by Equation 13.

4.2.4 Non-Circular (Shaped) Landmark Vicinities

In addition to the above ways of picking the $LM_i[id]$ and their corresponding $r_i[id]$, it is possible to modify the Landmark Vicinities so that they are no longer circular. The original purpose for this was to improve the path distribution while still keeping the routing tables small. However, through simulation, we could not show that this was a problem. Nevertheless, we describe the potential problem below so that the motivation for this aspect of our simulations can be better understood. This potential problem requires more study.

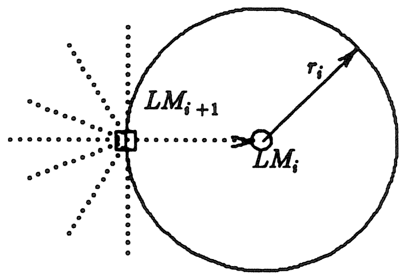
Figure 7 illustrates shaped Landmark Vicinities. In Figure 7a, the LM_{i+1} is at the edge of the Landmark Vicinity of LM_i. The paths from a large region of the network, shown by the dotted lines, all converge on the LM_{i+1} and subsequently follow the same path to the LM_i. This very poor distribution of paths penalizes the LM_{i+1} and all of the nodes and links between it and the LM_i. However, this represents efficient use of network resources overall because r_i is at a minimum, resulting in small routing tables.

In Figure 7b, the LM_{i+1} is closer to the LM_i (or, conversely, r_i is larger), resulting in a fair amount of Landmark Overage. In this case, the paths are more spread out, resulting in good (although still not as good as non-hierarchical routing) path distribution. The penalty, however, is paid in terms of the size of r_i, resulting in large routing tables.

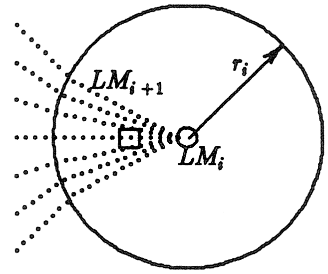
In Figure 7c, the LM_{i+1} is positioned the same as in Figure 7a, but the Landmark Vicinity is extended around the area where the LM_{i+1} is. This results in a fairly good path distribution without the inefficiencies seen in Figure 7b. This extension can be generated in a distributed fashion, and so is valid for use in a dynamic Landmark Routing environment.

When we define a Landmark Vicinity Extension, we define it in terms of hops from the Landmark around which the Extension extends, For instance, in Figure 7c, we say that the extension for the LM_i extends the $r_i[id]$ $e_i[id]$ hops around the LM_{i+1}.

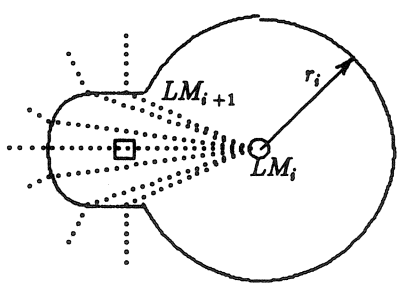
Figure 7
Landmark Vicinity Extension Example



7a: Poor path distribution, small routing tables



7b: Good path distribution, large routing tables



7c: Good path distribution and small routing tables using Landmark vicinity extension

5.0 LANDMARK HIERARCHY STATIC SIMULATION DESCRIPTION

To better understand the characteristics of the Landmark Hierarchy, we ran static simulations. By static, we mean that time was not an element in our simulations. This section describes the simulations.

5.1 Overview

Our simulations went as follows:

1. Generate a network to be simulated.
2. Create a Landmark Hierarchy for that network according to input parameters.
3. Measure routing table sizes, path lengths, and path distributions.
4. Analyze measurements.

Our approach to analyzing the Landmark Hierarchy was to do enough simulations so that any individual parameter could be isolated by averaging the results over many simulations where the parameter of interest was held constant, but other parameters varied.

The rest of this Section discusses each of the four parts of the simulations.

5.2 Generating Networks

Our simulations were run on 36 networks. The 36 networks come from all permutations of network parameters of 1) 200, 400, and 800 nodes; 2) average node degrees of 2.4, 4, and 6; and 3) and diameters of 30, 40, 50, and "very small"—the smallest diameter which we could generate of a network with a given number of nodes and node degree. Within these constraints, the networks were randomly generated.[6]

The nomenclature used throughout this paper to describe a network is *<number of nodes, node degree, diameter>*. For instance, a network with 400 nodes, a node degree of 2.4, and a diameter of 40 is 400,2.4,40. The networks with very small diameters are: 200,2.4,18; 200,4,8; 200,6,6; 400,2.4,22; 400,4,9; 400,6,7; 800,2.4,25; 800,4,11; and 800,6,7. The average diameter of the very small group is 12.56 hops.

We used an original technique for generating the networks while independently controlling the three parameters—number of nodes, node degree, and diameter—based on the loop-span network model. The loop-span network model is a derivation of the chordal ring (Arden, Lee, 1981; Doty, 1984). To our knowledge, this is the first published technique for automatically generating networks while independently changing these three parameters. Previous work allowed independent control of only the number of nodes and the average node degree (Callon, Hagouel, Sparta). Appendix B describes in detail the loop-span network model and how it can be used to independently control these parameters. We give a brief description here for completeness.

5.2.1 Automatically Generating Networks Using the Loop-span Network Model

A loop-span network consists of a Hamilton circuit, called the loop, and additional links bridging the Hamilton circuit, called spans. The loop, then, is a circular configuration of N nodes 1 through N, and N links, where Node 1 is connected to Node 2, Node 2 to Node 3, and so on until node N is connected back to Node 1. The loop alone has an average node degree of 2, and a diameter of $\frac{N}{2}\rfloor$, where $\rfloor$ is the floor function (for example, $2.5\rfloor = 2$). Now by adding an additional $L-N$ links (spans) to the loop, we shorten the diameter of the network. In general, we can make the diameter much shorter if we add long spans, and make the diameter a little shorter if we add short spans. The length of a span is a measure of the number of nodes in the loop that the span bridges. For instance, a span between Nodes 2 and 4 would be a span of length 2. The value of a span is based on the smallest number of nodes it bridges. Therefore, a span between Nodes 2 and 20 would be of length 18 if there were 36 or more nodes in the network, but would be a span of only 2 if there were, say, 20 nodes in the network.[7]

By specifying 1) the number of nodes in a network, 2) the number of links or, equivalently, the average node degree ($C = \frac{2L}{N}$, where C is the average node degree), and 3) the length of the maximum span in the network, one can control the diameter of the network while holding the number of nodes and links constant.

[7] A Hamilton circuit is a path through a network which visits every node exactly once, and returns to the starting node.

5.2.2 The Network Generation Algorithm

The algorithm for generating networks is simple. First, create the loop of N nodes and N links. Then, add $L-N$ spans, with the span lengths chosen uniformly among all possible span lengths from the minimum possible (2 hops) to the maximum specified span length s^{max}. The spans are added by randomly picking a node and checking to see if the node already has a span of that length, or has the maximum allowable number of links (12 in our simulations). If neither check is positive, the span is added. Otherwise, proceed is ascending order around the loop until a node is found with neither check positive.

5.3 Creating the Landmark Hierarchy

As described in Section 4, there are two approaches to specifying the Landmark Hierarchy, 1) assign the Landmarks according to values of r_i and d_i, and 2) assign the Landmarks randomly. Within Approach 1, there are four types,

a. Single value for all r_i and d_i^{max},

b. Single value for all r_i, but let $d_i^{max}(id)$ vary,

c. Single value for all d_i^{max}, but let $r_i\,[id]$ vary,

d. Vary both r_i and d_i^{max}.

Of the Landmark creation techniques which allow for a variable r_i, it is further possible to i) shape the Landmark vicinities and either, ii) specify the number of $LM_i\ i+1$ which are within r_i of an $LM_i\,(C_i)$, or iii) specify the overage of an LM_i vicinity over its nearest $LM_{i+1}\,(o_i)$.

In Approach 1, we attempted to assign Landmarks so that they were evenly distributed throughout the network, and so that there were as few Landmarks as possible within the specified constraints. The algorithm for assigning the Landmarks is given in Figure 8.

Notice in the third FOR statement, the process continues until there is only one LM sub i at the highest level. The reason for this is that it is very easy to truncate a distributed Landmark Hierarchy creation algorithm at this point, because there are no more Landmarks with which to run an election. However, we did experiment with

Figure 8
Algorithm for Assigning Landmarks in Simulations

FOR
    each node N, assign a Landmark priority number $pri(N)$ (either randomly or biased according to the node degree of each link).
FOR
    each node N, make the node an LM_0 and assign each node the appropriate r_0.
FOR
    each hierarchy level $i=1$ and up, while there is more than 1 LM_i at each level:

    FOR
        each node N until all nodes are either an LM_i or disqualified from becoming an LM_i:

        IF
            N is disqualified an LM_{i-1}
        OR
            N is an LM_{i-1},
        AND
            N is within $d_i^{max}(N)$ of an LM_i (i.e., $d_i(N) \leq d_i^{max}(N)$, where $d_i^{max}(N)$ is fixed for all N in approach 1a, and in approach 1b and 1c $d_i^{max}(N) = r_{i-1} - d_{i-1}(N)$, where in approach 1c r_{i-1} is fixed to some target value before being adjusted later on),
        THEN
            disqualify N from becoming an LM_i,
        ELSE IF
            $d_i(N) \geq d_i^{max}(N)$
        AND
            $pri(N)$ is greater than the Landmark priorities for all LM_{i-1} within $d_i^{max}(N)$ of N which are not disqualified from becoming and LM_i,
        THEN
            make N and LM_i.
    END FOR
END FOR

---

changing that to a higher number, thus causing multiple Landmarks at the highest level. Incidentally, this algorithm is very close to an algorithm that could be used in a distributed environment the create the Landmark Hierarchy.

In Approach 2, Landmarks are randomly assigned to each node for each hierarchical level until the specified number of Landmarks T_i for that level are assigned.

After the Landmarks are assigned, each Landmark $LM_i[id]$ is given an $r_i[id]$. In

Approaches 1a and 1b, the r_i are fixed for all hierarchy levels i. In Approaches 1c, 1d, and 2, $r_i[id] = d_{i-1}(id \rightarrow c_i^{th} closestLM_{i-1}) + o_i$.

5.4 Landmark Hierarchy Measurements

After creating the Landmark Hierarchy, several measurements are made on the hierarchy. First, the number of hierarchical levels H, the number of Landmarks at each level T_i, and the total number of Landmarks T are counted.

Second, the average routing table sizes for each level R_i, the average total routing table sizes R, and the maximum and minimum routing table size R_i^{max}, R^{max}, R_i^{min}, and R^{min} for the network (one simulation) are calculated. (For multiple simulation experiments, the average of these maximums and minimums are calculated.) Nodes will not contain routing table entries for levels lower than the i^{th} level of an LM_i. For example, if a node is within the vicinity of an LM_4 (which by definition is also an LM_0, LM_1, LM_2, and an LM_3), then it will only add to the statistics on R_4, not R_0 through R_3. In a real Landmark Routing implementation, a node may (or may not) store all instances of a single Landmark separately. However, it will still only require one message over a link to update all of the separate entries. Therefore, our method of counting routing table entries accurately reflects link overhead.

Third, the average distance d_i from all nodes to their closest LM_i, and the average r_i and $v(r_i)$ for all LM_i, are calculated for all i.

Fourth, several statistics on path lengths are collected. The average path length between all node pairs for both the shortest path P^{sh} and the path used in the Landmark Hierarchy P^{lm} are calculated. These statistics are also kept for each hierarchy level, based on the hierarchy level that was used for the routing decision on the first hop from the source node. Also, the ratios $\hat{P} = \frac{P^{lm}}{P^{sh}}$ and $\hat{P}_i$ are calculated. Finally, the longest path for each level is recorded.

Last, statistics on the distribution of paths F_i are calculated. This is done by counting the number of paths which traverse each node, and then averaging these values for the LM_i at each level i. These statistics are presented two ways. First, all nodes are ranked from that with the least traffic to that with the most traffic. Then, for each level

i, the average percentile ranking F_i^{per} for all of the LM_i at that level is calculated. This is done because the loading varies so much from node to node that actual values are less useful than a ranking. Second, for each level i, the ratio of the average number of paths for each LM_i over the average number of paths for all nodes F_i^{rat} is calculated.

Table 2 summarizes the statistics collected in the simulations.

Table 2
Statistics Collected in Simulations

Statistic	Nomenclature
Number of Hierarchy Levels	H
Number of Landmarks	T_i
Average Routing Table Sizes	R, R_i
Maximum Routing Table Sizes	R^{max}, R_i^{max}
Minimum Routing Table Sizes	R^{min}, R_i^{min}
Average Distance to Landmark	d_i
Average Landmark Radii	r_i
Average Nodes Covered by Landmark	$v(r_i)$
Average Shortest Path Lengths	P^{sh}, P_i^{sh}
Average Landmark Path Lengths	P^{lm}, P_i^{lm}
Increase in Path Length	$\hat{P} = \frac{P^{lm}}{P^{sh}}$, $\hat{P}_i$
Maximum Landmark Path Lengths	$P_i^{sh\ max}$, $P_i^{lm\ max}$
Paths Through Landmarks (percentile)	F_i^{per}
Paths Through Landmarks (above average)	F_i^{rat}

5.5 Landmark Hierarchy Parameter Values

In this section, we give the parameters used for most of the Landmark Hierarchy simulations. These parameters are summarized in Table 3. Other isolated simulations with different parameters were done to test specific variations, but these were not included in the main body of experiments.

Table 3 is divided into several tables, each listing the parameters which apply to given sets of approaches. The parameters were chosen for learning as much as possible about the Landmark Hierarchy, while staying within the constraints of practical choices.

Table 3
Parameters Used in Simulations

Approach 1a: Fix r_i and d_i^{max} Approach 1c: Vary r_i, Fix d_i^{max}						
Parameter	$i=0$	$i=1$	$i=2$	$i=3$	$i>3$	Nomenclature
r_i (initial)	2	4	8	16	2^{i+1}	h1
d_i^{max}	0	1	2	4	2^{i-1}	

Approach 1b: Fix r_i, Vary d_i^{max} Approach 1d: Vary r_i, Vary d_i^{max}						
Parameter	$i=0$	$i=1$	$i=2$	$i=3$	$i>3$	Nomenclature
r_i (initial)	2	4	8	16	2^{i+1}	h2
d_i^{max}	0	≤ 1	≤ 4	≤ 8	$\leq 2^i$	
r_i (initial)	1	3	5	8	2^i	h3
d_i^{max}	0	≤ 1	≤ 2	≤ 3	$\leq 2^{i-1}$	

Approach 1c: Vary r_i, Fix d_i^{max} Approach 1d: Vary r_i, Vary d_i^{max} Approach 2: Randomly Pick LM_i						
Parameter	$i=0$	$i=1$	$i=2$	$i=3$	$i>3$	Nomenclature
o_i	0	0	0	0	0	o1
o_i	*	0	1	2	2^{i-2}	o2
o_i	*	0	0	1	2^{i-3}	o3
c_i	1	1	1	1	1	c1
c_i	2	2	2	2	2	c2
c_i	3	3	3	3	3	c3
e_i	$r_0/1$	$r_1/1$	$r_2/1$	$r_3/1$	$r_i/1$	e1
e_i	$r_0/2$	$r_1/2$	$r_2/2$	$r_3/2$	$r_i/2$	e2
e_i	$r_0/3$	$r_1/3$	$r_2/3$	$r_3/3$	$r_i/3$	e3

* Use fixed r_i (usually 2) instead of o_i

Approach 2: Randomly Pick LM_i						
Parameter	$i=0$	$i=1$	$i=2$	$i=3$	$i>3$	Nomenclature
T_i	N	$N/2$	$N/4$	$N/8$	$N/2^i$	t2
T_i	N	$N/3$	$N/9$	$N/27$	$N/3^i$	t3

The nomenclature listed in Table 3 allows us to label each simulation experiment by the parameters used. For instance, if we simulate a 400 node network with node degree 4 and diameter 40 using Approach 1c with hierarchy parameters (r_i and d_i^{max}) h2, overage (o_i) o2, and a vicinity extension (e_i) of e2, then we label that experiment 400,c4,40:1c,h2,o2,e2. In what follows, we explain our choices for each of the parameters.

5.5.1 Landmark Radii and Distance to Landmark

For all hierarchies in Approach 1, the Landmark radii r_i and distances d_i^{max} are chosen as powers of 2. The idea behind this choice is to prevent there from being too many landmark levels in the hierarchy for networks with large diameters—a condition which would occur if the values of r_i and d_i^{max} increased linearly with i. As it turns out, we did experiment with linear increases for a few isolated simulations, but did not use these for the main body of experiments.

For Approach 1a, we chose r_i to be 4 times greater than d_i^{max}. This provides a healthy amount of overage (see Equation 10 and subsequent example). We chose $r_0 = 2$ because it allows for more complete local network information. This information may be useful in dealing with local problems, such as routing around broken links. It is interesting to note that this was the hierarchy that we originally thought would be appropriate for the Landmark hierarchy in all situations, and initially was the only hierarchy simulated.

For Approach 1c, the same parameters were used, but r_i was modified after the Landmarks were assigned. For Approach 1b, the same r_i is used, but now d_i^{max} is allowed to be as high as one-half r_i. For Approach 1d, an additional level was added below $r_i = 8$, and r_0 was reduced from 2 to 1. This was done because it was discovered that the routing table sizes are effected greatly by entries at those lower levels.

5.5.2 Adjusting the Landmark Vicinity

This capability is represented by the third table in Table 3. The parameters o_i and c_i are used to change the size of r_i after the Landmarks are assigned. The parameter e_i is used to shape the Landmark Vicinity.

For o1, there is no overage—the edges of the Landmark Vicinities for an LM_i fall right at the LM_{i+1}. For o2 and o3, the overage increases in powers of 2, but at higher levels than r_i. This makes the overage a percentage of r_i, which is also incremented in powers of 2.

The Coverages c1, c2, and c3 cover 1, 2, or 3 LM_i respectively at all levels. From a dynamic network operation point of view, more coverage adds robustness to the hierarchy. It means that a node may lose its association with one LM_i, but will still others which it may use in its address. The multiple LM_i may also be used simultaneously to disperse paths, either for distribution of loading, or for survivability.

Notice that we do not allow for the specification of both $o_i > 1$ and $c_i > 1$ (o1 is the default when c2 and c3 are used, and c1 is the default when o2 and o3 are used). There is no reason in practice why both couldn't be independently varied. We didn't do it because varying both simultaneously would obscure the effect that either was having.

The Vicinity Extensions e_i are also proportional to r_i, with e1 being the largest extension and e3 being the smallest.

5.5.3 Adjusting the Number of Landmarks

In Approach 2, we pick the number of Landmarks T_i at each level. We specified geometric progressions for T_i for the same reason we do for r_i and d_i. For t1, each increasing level i has half as many LM_i as the last level. For t2, each increasing level i has one third as many LM_i as the last level.

6.0 RESULTS OF SIMULATIONS

We initially did 756 simulations; 21 different hierarchy parameter combinations on 36 networks each. These simulations make up the main body of our experiments. In our analysis, we consider each simulation only once, rather than take the average of several identically configured simulations. For each experiment we take the average of multiple simulations where one parameter (the one of interest) is held constant. Normally, we do not include the networks with very small diameters in the set of averaged simulations. This is because, since their diameters are different for different numbers of nodes and different node degrees, we often cannot use them without skewing the results. Therefore, we only use the simulations with the very small diameters when 1) comparing them against other experiments where the diameter is held constant, or 2) when considering them individually.

This main body of simulations provides the basis for testing the impact of differences in networks or differences in hierarchies. In addition, we run several other simulations when needed. These additional simulations are done on a limited basis, and therefore are not included in the averages done with experiments from the main body of simulations. The parameters for these simulations are given where the simulations are discussed.

6.1 Variation Between Individual Simulations

To determine the variability between individual simulations, we ran one simulation 20 times, each time using a different random number generator seed digit to obtain different results. For this test, we chose the simulation 200,2.4,vs:2,t3,o1,e3. This simulation is likely to exhibit as much variability as any simulation because 1) it uses the smallest network, and so any variation between individual nodes will be greater in proportion to the total number of nodes, 2) it uses the smallest number of links, 3) it uses a random process of choosing Landmarks, and 4) of the random approaches, it has the smaller number of Landmarks.

We calculated the mean, median, and standard deviation for both the routing table sizes R and the path length increases $\frac{P^{lm}}{P^{sh}}$ for all twenty simulation runs. For the routing table sizes, we got a mean of $\mu_R = 26.29$, a median of $m_R = 25.72$, and a standard deviation of $\sigma_R = 2.04$. For the path length increases, we got a mean of $\mu_{\hat{P}} = 1.11$, a median of $m_{\hat{P}} = 1.11$, and a standard deviation of $\sigma\hat{P} = .015$. These results tell us two things. First, the mean is a useful value for characterizing the results of multiple simulations. Second, for a single simulation, we have 95 percent confidence that the result will be within about 15 percent (two standard deviations) of the routing table size, and within 3 percent of the path length increase. For the experiments which take the average multiple simulations, our results will be even more accurate, depending on the number of simulations. The smooth trends in the data in our large body of simulations also supports the statistical significance of our simulations.

6.2 Presentation of Experiment Results

Most of our results are presented in tabular rather than graphical form. This is done for three reasons. First, we usually present enough data, or the differences between different datum are small enough that a graph only muddles up the presentation. Second, the tables allow for much more data to be presented, even data which we may not discuss. This allows the reader to draw his or her own conclusions. Third, it is usually adequate to simply observe general trends or specific values in the data. Tables are adequate for this.

When it is necessary to show data graphically, to observe shapes of curves for instance, then we do present the data that way.

6.3 Broad Comparison of Hierarchy Types

Table 4 lists the experiments and gives the results for a broad comparison of hierarchy types. Each column in Table 4 gives the average of the hierarchy type over 27 networks (36 networks minus the 9 networks with very small diameters). The *'s in the network type positions of the experiment descriptions imply an averaging over all such parameters. Two of the parameters have a ranking associated with them; R and $\hat{P} = \frac{P^{lm}}{P^{sh}}$. The columns in Table 4 are in order of the R ranking. These results give us a comparison of the various Landmark Hierarchy approaches. The following sections analyze Table 4.

Table 4
Hierarchy Comparison Simulations and Results

	H	R				$\hat{P}$		F^{per}
			R	R^{min}	R^{max}			
,,*:1c,h1,o1	7.67	*1*	14.50	6.33	25.00	*19*	1.16	0.54
,,*:1d,h2,o1	7.00	*2*	14.82	6.37	25.30	*18*	1.15	0.55
,,*:1d,h2,o1,e3	6.96	*3*	17.87	8.44	30.96	*17*	1.14	0.48
,,*:1d,h2,o1,e2	6.93	*4*	18.37	9.19	31.26	*16*	1.14	0.46
,,*:2,t2,o1,e3	9.00	*5*	18.65	8.37	34.70	*21*	1.17	0.57
,,*:2,t2,o1,e2	9.00	*6*	19.03	8.37	35.81	*20*	1.17	0.57
,,*:2,t3,o1,e3	6.00	*7*	20.21	9.00	38.07	*15*	1.14	0.59
,,*:2,t3,o1,e2	6.00	*8*	21.16	9.44	39.85	*14*	1.14	0.59
,,*:1d,h2,o1,e1	7.00	*9*	22.02	10.63	36.30	*12*	1.11	0.44
,,*:1d,h3,o3,e3	7.93	*10*	22.12	10.93	36.96	*11*	1.11	0.48
,,*:1d,h2,o2	7.00	*11*	22.90	11.26	36.70	*10*	1.11	0.48
,,*:2,t2,c2	9.00	*12*	22.98	11.26	39.59	*8*	1.10	0.58
,,*:1d,h3,o1,e1	7.96	*13*	23.26	10.70	37.70	*7*	1.10	0.41
,,*:2,t2,o1,e1	9.00	*14*	24.40	10.96	42.00	*13*	1.13	0.54
,,*:1d,h2,o3,e2	6.96	*15*	24.81	12.07	42.00	*9*	1.11	0.47
,,*:2,t3,c2	6.00	*16*	26.40	12.56	44.93	*5*	1.09	0.56
,,*:2,t3,o1,e1	6.00	*17*	27.14	12.52	46.59	*6*	1.10	0.54
,,*:2,t2,c3	9.00	*18*	30.84	15.81	50.30	*4*	1.08	0.55
,,*:2,t3,c3	6.00	*19*	35.50	18.93	58.15	*3*	1.06	0.53
,,*:1b,h2	7.00	*20*	36.38	19.52	54.07	*2*	1.05	0.44
,,*:1a,h1	7.67	*21*	46.88	29.00	64.44	*1*	1.03	0.41

6.3.1 Strong Inverse Relationship Between Routing Table Size and Path Lengths

The first thing we notice is the strong inverse relationship between the average routing table sizes R and the increase in landmark path length over shortest path length $\hat{P} = \frac{P^{lm}}{P^{sh}}$. This relationship is remarkably consistent. The product $R(\hat{P}-1)$ for all experiments in Table 4 yields a standard deviation of 0.42 on a mean of 2.51. We take this strong consistency (small standard deviation) to mean that these hierarchies perform approximately the same (unless of course one strongly prefers small tables to small paths, or vice versa). We see From Table 5 that this product will change with changes in the network parameters (number of nodes, node degree, and diameter). Also, the product is not stable for very small $\hat{P}$ or very small R (notice that $R(\hat{P}-1) = 1.41$ for :1a,h1). However, for reasonable values of R and $\hat{P}$, the product is surprisingly consistent.

The results for R and $\hat{P}$ in Table 4 do not imply that there are no advantages of one Landmark hierarchy over another. The best hierarchy to use depends on the circumstances in a network. For instance, if a routing protocol is used that requires very few updates, and network nodes have sufficient memory, then one of the hierarchies with a large R and small $\hat{P}$ may be the most efficient. The opposite also holds true. It is good, however, that the specification of the Landmark Hierarchy allows for quite a bit of latitude for adjusting its performance to fit a certain environment. In our experiments R varied over an order of magnitude of 3, and $\hat{P}$ varied over an order of magnitude of 5. This variation could be further increased with larger R smaller $\hat{P}$, because one can make Landmark Vicinities arbitrarily large.

We see from Hagouel's study of the area hierarchy that this sort of adjustment may be made for the area hierarchy as well, namely by adjusting the average number of sub-areas in an area.

6.3.2 Large Variance in Routing Table Sizes

In Table 4, the averages of the maximum and minimum routing table sizes (R^{max} and R^{min} respectively) are shown. These were calculated by taking the maximum and minimum size routing table in each network, and then averaging all of those for each hierarchy type.

The average of $\frac{R^{max}}{R^{min}}$ for all entries in Table 4 is 3.53 (with a standard deviation of 0.58). So, depending on where in a network a node is, it may expect to have routing table sizes ranging over a magnitude of around 5.3 (the mean plus three standard deviations). This would lead to increased routing traffic on links in some areas of a network, a condition which network designers must be aware of, and which will require more study.

If we compare $\frac{R^{max}}{R^{min}}$ for Approaches 1 and 2, we see that Approach 2 has an average range of 3.78, while Approach 1 an average range of 3.31. This shows a small increased variability in the random approach, which is expected. If we compare $\frac{R^{max}}{R^{min}}$ for the first 10 columns in Table 4 (small R) with the last 10 (large R), we get 3.87 and 3.23 respectively. This shows a small increased variability with small table sizes.

6.3.3 Path Distribution

The values for path distribution (F^{per}) do not have any strong trends. Recall that this value shows how the traffic passed by a Landmark node compares with that passed by a non-Landmark node. The mean for F^{per} over all experiments is 0.51 (51st percentile), with a standard deviation of 0.06. This shows that the traffic carried by Landmarks is virtually the same as that carried by non-Landmarks. It also shows that path distribution is fairly stable between different hierarchy types, and is not a strong design consideration when picking hierarchy types. We can, however, find two trends in F^{per} from Table 4.

First, the mean for the experiments using Approach 2 (random assignment of LM_i) is 0.56, while that for Approach 1 is 0.47. Both of these values are slightly less than one standard deviation away from the overall mean. Random assignment performs slightly worse than what might be called uniform assignment. It is interesting to note, however, that random assignment performs about the same as uniform assignment with regards to routing table size and path lengths.

Second, the mean for the first 10 columns is 0.53, and 0.50 for the last 10 columns—almost no difference. This lack of difference is good, although it is a surprise to us. It means that the small amounts of overage of a Landmark Vicinity does not appear to have the bad effect we expected.

It is interesting to note that in the experiment listed in the last column in Table 4, the Landmarks carry a disproportionately small percentage of the traffic. In fact, this value, 0.41, is further from the mean (nearly two standard deviations) than any other F^{per} in Table 4. This is a little distressing in that, if these Landmarks are getting a smaller than average percentage of traffic, other nodes are getting a larger than average percentage of traffic. The value here, however, is not low enough to warrant focusing on this potential problem any more at the moment.

We are also interested in seeing whether the Vicinity Extensions have any effect on path distribution. For hierarchy types *,*,*:2,t2,o1,e3 and *,*,*:2,t3,o1,e3, we have an average F^{per} of 0.58. For hierarchy types *,*,*:2,t2,o1,e1 and *,*,*:2,t3,o1,e1, we have an average F^{per} of 0.54. Contrary to our expectations, the larger Extension performed worse than the smaller extension, although the difference is not great. We conclude then that from this analysis we cannot find any advantage to Vicinity Extensions.

One possible explanation for the lack of difference is as follows. It seems likely that there are two opposing influences on path distribution. On the one hand, when a packet is being routed towards an LM_{i+1}, it will have the effect of putting more traffic through the LM_{i+1} than what would be normal. On the other hand, when the packet reaches the point where it routes toward an LM_i, it could be that this causes the packet to be diverted *away* from the LM_{i+1}, causing less traffic to pass through the LM_{i+1} than what would be normal. The second effect would then cancel out the first effect. If an LM_{i+1} is close to the edge of the LM_i Vicinity, then more traffic would initially be routed towards it. However, this also means that the LM_{i+1} is further away from the LM_i, so that when traffic is finally diverted towards the LM_i, it is diverted away from the LM_{i+1} more sharply than it would if the two LM's were close together.

More study is required to verify this explanation. Fortunately, none of the path distributions are bad, and so are not of great concern. We note also that path distributions will change, and in particular even out, when metrics (other than 1) are given to the links—especially when those metrics are traffic dependent.

6.4 Broad Comparison of Network Types

For this comparison, we ran 10 experiments, each experiment isolating one network parameter—number of nodes, node degree, or diameter. Table 5 lists the experiments and gives the results. As with Table 4, each parameter has been averaged over all of the simulations indicated by the experiment (for instance, all simulations with 200 nodes). Note again that the networks with "very small" diameter were not included in the averages derived for the experiments which isolate number of nodes and node degree For these, only the networks with diameters of 30, 40, and 50 were used. Table 5 does not have a ranking like Table 4. Instead, the experiments are considered in groups of like parameter type; for instance, the three experiments where the number of nodes changes, or the five experiments where the diameter changes. (The double bars in Table 5 indicate this grouping.)

Table 5
Network Comparison Simulations and Results

	H	R			$\hat{p}$	F^{per}
		R	R^{min}	R^{max}		
200,*,*:*	6.92	17.84	10.93	26.67	1.10	0.50
400,*,*:*	7.38	23.60	11.71	39.73	1.11	0.51
800,*,*:*	7.86	31.46	13.31	55.13	1.14	0.53
,2.4,:*	7.37	23.30	10.45	44.30	1.12	0.51
,4,:*	7.39	23.96	11.85	39.12	1.11	0.52
,6,:*	7.39	25.64	13.66	38.11	1.12	0.51
,,vs:*	6.41	43.47	16.16	78.43	1.28	0.62
,,30:*	7.22	26.71	12.42	45.61	1.13	0.53
,,40:*	7.43	23.56	11.43	39.20	1.11	0.50
,,50:*	7.51	22.63	12.11	36.72	1.10	0.51

6.4.1 Number of Hierarchy Levels

The average number of hierarchy levels H is increasing with number of nodes. This increase is very nearly linear, showing approximately an additional half of a hierarchical level for every doubling of the number of nodes. This is at first glance surprising, because we stated in Section 4.1 that H should only be dependent on network diameter and r_i. However, this trend is entirely due to the Approach 2 hierarchies. Since Approach 2 assigns a certain number of Landmarks T_i at each level until there is only one LM_H at the highest level, the number of nodes will directly impact the number of levels. When we run the experiments using only Approach 1 type simulations, we get $H_{200} = 7.29$, $H_{400} = 7.27$, and $H_{800} = 7.27$—virtually no difference.

We see also that there no change in number of hierarchy levels due to changing node degree.

With regards to the group of experiments isolating diameter, we see a clear increase in the number of hierarchical levels H with increasing diameter. This is the expected result, based on the discussion in Section 4.1.

6.4.2 Routing Table Sizes

We see that all three groups of experiments in Table 5 have an effect on routing table size, although the effect from changing node degree is less dramatic, and the effect from changing diameter is more dramatic. Routing table sizes increase with increases in both the number of nodes and the node degree, and decrease with an increase in network diameter. Recalling Equation 4, routing table size is dependent on the ratio $\frac{v(r_i)}{v(d_i)}$. It makes sense that the function $v(x)$ is dependent on the number of nodes, the node degree, and the diameter. If the node degree and the diameter is held constant, then obviously the number of nodes within x hops ($v(x)$) will increase. Likewise, for a given number of nodes and a given node degree, if the diameter increases, the $v(x)$ will decrease.

It is apparent, however, that either 1) the function $v(x)$ is nonlinear with x, or 2) r_i and d_i^{max} are affected by the three network parameters tested. As it turns out, both are true, although the nonlinearity of $v(x)$ is the major contributor to changing routing table sizes. In Section 6.3, we explore the function $v(x)$ in more detail.

We see from Table 5 that more nodes increase the routing table size. This increase in R is not linear with the doubling of network size. If it were, we would have some indication that $R = O(\log N)$. Unfortunately, the increase in R is greater than linear. On the good side, the increase in R is certainly not the same as that of the number of nodes. Also, these experiments are for non-changing diameter. In real networks, diameter tends to increase roughly with the log of the number of nodes [McQ80]. Since larger diameters means smaller routing table sizes, we expect to see some improvement in the increase of R with increasing N over that shown in Table 5. In section 7.3, we determine the general relationship between R and N.

We also notice that the ratio $\frac{R^{\min}}{R^{\max}}$ is increasing with node size. The indication, then, is that there is greater variability in routing table sizes as the number of nodes grows. In addition, we see that the average maximum routing table size is increasing faster than the average routing table size. For 200,*,*:*, $\frac{R^{\max}}{R} = 1.49$, for 400,*,*:*, $\frac{R^{\max}}{R} = 1.68$, and for 800,*,*:*, $\frac{R^{\max}}{R} = 1.75$. This increase, however, appears to be leveling off, and so should not be a significant concern for increasing numbers of nodes.

It is interesting to note the surprising result that, for increasing node degree, R and $R^{\min}$ increase slightly, but $R^{\max}$ decreases. We believe that this is because the number of nodes connected to a node will have a larger impact on $v(x)$ in networks with small node degrees than in networks with large node degrees. If a node in a network with node degree 3 has 4 nodes connected to it, it will see a dramatic difference in $v(x)$ as compared to a node with only 2 other nodes connected to it, especially for small x. However, if a node in a network with node degree 6 has 7 nodes connected to it, it will not see that much difference in $v(x)$ as compared to a node with 5 nodes connected to it. The reason for this will be more clear after reading the analysis of $v(x)$ in Section 6.3.

6.4.3 Path Lengths

For path lengths ($\hat{P} = \frac{P^{lm}}{P^{sh}}$), we see trends for number of nodes and diameter, but not node degree. The trends are in the same direction as they are for routing table sizes—namely, larger path lengths for increasing number of nodes, and decreasing path lengths for increasing diameter. This is different from what we saw in Table 4. There, routing table sizes consistently decrease with increasing path lengths. In Table 5, routing table sizes and path lengths vary together, resulting in undisputable better performance in networks with small number of nodes, small node degrees, and large diameters. One would of course expect better performance with a small number of nodes—with the Landmark hierarchy as well as the area hierarchy and non-hierarchical routing. In Sections 6.7 and 6.10, we study path lengths in more detail.

6.5 Analysis of Function v(x)

It is clear from the preceding discussions that the behavior of the function $v(x)$ (the average number of nodes within x hops of a node) is very important to routing table size. In this section, we consider $v(x)$ in detail. To do this, we compared three groups of three networks each. In the first group, all have network diameters of 36 and node degrees of 4, but the number of nodes are 200, 400, and 800. In the second group, all have 400 nodes and node degrees of 4, but diameters of 9, 19 and 36. In the last group, all have 400 nodes and diameters of 22, but node degrees of 2.4, 4, and 6. With these three groups, we can isolate either the number of nodes, the node degree, or the diameter. For each network, we calculated $v(x)$ for all x, $0 \leq x \leq D$. In Figures B1, B2, and B3, we plot $v(x)$ for each of the three groups. These Figures are in Appendix C.

Each of these plots have been normalized so that the shape of the curves can be directly compared. First consider Table B1, where the effect of the number of nodes on $v(x)$ can be seen. Here we see that the shapes of the curves are very similar, although there is a slight noticeable difference. The curves are nearly linear, although as it turns out, that is due to the diameter and node degree chosen. We notice also that the magnitude of $v(x)$ is proportional to the number of nodes in the network. (This is not visible because the graphs are normalized. The maximum value for each curve is stated below the graph.) We conclude, then, that the number of nodes in a network has a large effect on the magnitude of $v(x)$, but only a small effect on its shape.

In Figure B2 and B3, however, we see different curve shapes, some of which have a strong non-linearity. It is this non-linearity that will cause different values of $\frac{v(r_i)}{v(d_i)}$ for different values of r_i and d_i.

Figures B4, B5, and B6 give plots of $v(x) - v(x-1)$. These graphs show the slope of the curves in Figures B1, B2, and B3 respectively. They are also normalized. Here we see that, for the lines that appear largely linear in Figures B1, B2, and B3, there is a steep rise in slope at the beginning before a constant slope is reached. From these curves, we see that close to a node, there is a rapid increase in the additional number of nodes picked up for each additional hop away from the node. We call this fan-out. However, within a few hops, a steady state is reached, whereby roughly the same additional number of nodes is picked up with each additional hop away from the node. This continues until the diameter of the network is nearly reached, at which time almost all of the nodes in the network are picked up, and very few additional nodes are picked up with each remaining additional hop.

For the case where there is little linearity, such as can be seen by the line for node degree of 2.4 in Figures B3 and B6, we see a less marked fan-out, and steady state is never reached. At a point where about half of the nodes in the network are picked up (at about half of the network diameter), we see a decrease in the number of nodes picked up. The rate of this decrease is similar to that of the increase—the curve is fairly symmetrical.

The reason for this fan out is as follows. Consider a network topology where a node is connected to C other nodes (C is the node degree). Each of those C nodes are in turn connected to $C-1$ additional nodes, each of those nodes are in turn connected to $C-1$ additional nodes, and so on. At each additional hop, we see an exponential increase in the number of nodes picked up, namely

$$C \times (C-1)^{x-1} \qquad 14$$

where x is the number of hops. It is this exponential fan-out which causes the fan-out we see in Figures B1 through B6.

In real networks, however, the fan-out sooner or later becomes saturated. This is because each picked-up node doesn't necessarily connect to $C-1$ new nodes. In particular, some of the links to "new" nodes will actually connect back to nodes already picked up. It is the node degree, and the percentage of new nodes, that actually determine the degree of fan-out. A large percentage of new nodes will result in many nodes being picked up, resulting in a small network diameter (relative to the number of nodes), and a curvincy $v(x)$ line. For a given node degree and number of nodes in the network, a small percentage of new nodes will result in not many new nodes being picked up, and therefore a large network diameter, and a mostly linear $v(x)$ curve.

Because Figures B1 through B6 are normalized, they don't give much of an idea of magnitude. Figures B7, B8, and B9 are non-normalized plots of $v(x)$. Because of the large difference in scale between the various curves, these are only plotted for the first five hops. This allows a more detailed view of the initial fan-out.

6.6 Analysis of Routing Table Size by Hierarchical Levels

Now we would like to consider routing table sizes in more detail. For this analysis, we have two tables, Table 6 and Table 7. Both tables show routing table sizes for all hierarchical levels. Table 6 shows this over averages for all hierarchy types, similar to Table 5. Table 7 shows this for three specific hierarchy types and three specific networks. The three hierarchy types represent small table sizes (:1c,h1,o1), large table sizes (:1a,h1) and something in between (:1d,h2,o2). The three network types also represent small table sizes (800,2.4,153:), large table sizes (800,6,7:), and something in between (800,4,36:), for 800 node networks.

The first thing we notice about Tables 6 and 7 is that most of the routing table entries come from the first few hierarchical levels. From Table 6, except for the node degree 2.4 experiment, nearly half of the routing table entries come from the first two hierarchical levels. From Table 6, we see some evidence that networks with a large fan-out (small diameter, large node degree) have a larger percentage of routing table entries in the lower few hierarchical levels. For instance, the Node Degree 6 experiment has 55%

Table 6
Routing Table Sizes by Network Type

	R_i										R
	0	1	2	3	4	5	6	7	8	9	
200,*,*:*	5.37	3.09	2.71	2.47	1.85	1.38	1.21	1.00	1.00		17.84
400,*,*:*	6.53	4.25	3.56	3.60	2.50	1.47	1.18	1.42	1.00		23.60
800,*,*:*	8.14	5.75	4.73	5.17	3.36	1.98	1.22	1.27	1.62	1.00	31.46
,2.4,:*	4.33	3.12	3.83	4.87	3.60	1.94	1.22	1.21	1.28	1.00	23.30
,4,:*	6.72	4.73	3.84	3.38	2.11	1.50	1.18	1.26	1.35	1.00	23.96
,6,:*	8.99	5.23	3.33	2.99	2.01	1.44	1.21	1.20	1.29	1.00	25.64
,,vs:*	10.01	11.16	8.65	8.32	3.10	1.98	1.52	1.42	1.33	1.00	43.47
,,30:*	7.42	4.92	4.10	4.35	2.90	1.57	1.18	1.29	1.33	1.00	26.71
,,40:*	6.27	4.28	3.62	3.69	2.41	1.62	1.18	1.21	1.34	1.00	23.56
,,50:*	6.36	3.88	3.28	3.20	2.40	1.70	1.26	1.19	1.26	1.00	22.63

of its routing table entries come from the first two levels, while Node Degree 2.4 has only 32%. This can be seen to a lesser degree in the experiments isolating diameter. Here, the very small diameter experiment has 49% coming from the first two levels, while diameter 50 has 45%.

From Table 7, we see a similar trend, but with more variation. All of the experiments with Diameter 153 have contributions to the routing table coming fairly evenly from all hierarchical levels, although again more so from the lower levels. From the experiments with Diameter 7 we see the contributions coming more from one or two levels, but not always the lowest level. In 800,6,7:1c,h1,o1, Level 2 is by far the largest contributor to the routing table. For 800,6,7:1d,h2,o2, it is Level 0 (although Levels 1 and 2 still contribute quite heavily), while for 800,6,7:1a,h1, it is Levels 1 and 2. This heavy contribution comes from the positioning of r_i and d_i on the steep part of the $v(x)$ curve, so that $v(r_i)$ is much greater than its corresponding $v(d_i)$.

Table 7
Routing Table Sizes for Nine Individual Simulations

	R_i										R
	0	1	2	3	4	5	6	7	8	9	
:1c,h1,o1											
800,2.4,153:	2.21	0.56	1.85	1.46	1.61	1.33	0.82	0.90	0.99	1.00	12.72
800,4,36:	3.61	2.82	4.71	2.68	1.35	0.61	0.98	1.00			17.75
800,6,7:	5.11	5.73	16.83	8.00	1.00						36.66
:1d,h2,o2											
800,2.4,153:	4.12	1.71	2.32	2.31	2.02	1.43	1.15	1.00	1.00		17.08
800,4,36:	11.98	6.99	4.25	3.10	1.82	1.00	1.00				30.13
800,6,7:	26.08	19.69	16.86	3.00	1.00						66.63
:1a,h1											
800,2.4,153:	4.45	1.36	6.65	5.61	5.93	4.76	3.61	2.00	1.00	1.00	36.36
800,4,36:	12.23	12.83	19.64	9.84	4.89	2.00	1.00	1.00			63.43
800,6,7:	26.15	85.17	78.00	9.00	1.00						199.32

6.7 Analysis of Path Length by Hierarchical Levels

Tables 8 and 9 show paths lengths. They are given for the same experiments as shown in Tables 6 and 7 respectively. Tables 8 and 9 give the increase in path length for Landmark paths over shortest path for all hierarchy levels.

The first thing one notices from Tables 8 and 9 is that path lengths are shorter (compared to shortest path) at the lower levels than at the higher levels. This of course is what is expected. At the lower levels, routing is based on a finer granularity of information. At level 0, for instance, routing is based entirely on the destination node. Therefore, the paths chosen are the same as shortest path.

It appears that many of the path lengths become suddenly much worse at some hierarchical level. For instance, for *,*,vs:* in Table 8, Level 2 shows a sudden jump from 1.05 to 1.15, compared with a smaller increase of 1.05 to 1.07 for *,*,30:*. From Table 9, we see this is true for almost all of the entries. For instance, for

Table 8
Path Lengths by Network Type

	$\frac{P_i^{lm}}{P_i^{sh}}$										$\frac{P^{lm}}{P^{sh}}$
	0	1	2	3	4	5	6	7	8	9	
200,*,*:*	1.00	1.03	1.03	1.03	1.05	1.11	1.19	1.25			1.10
400,*,*:*	1.00	1.05	1.06	1.06	1.07	1.13	1.15	1.17	1.26		1.11
800,*,*:*	1.00	1.06	1.09	1.09	1.09	1.14	1.20	1.15	1.18	1.30	1.14
,2.4,:*	1.00	1.02	1.04	1.06	1.08	1.12	1.18	1.16	1.22	1.29	1.12
,4,:*	1.00	1.06	1.07	1.06	1.06	1.12	1.18	1.18	1.22	1.32	1.11
,6,:*	1.00	1.07	1.07	1.06	1.06	1.13	1.19	1.21	1.20	1.29	1.12
,,vs:*	1.00	1.05	1.15	1.27	1.35	1.38	1.42	1.49	1.57	1.62	1.28
,,30:*	1.00	1.05	1.07	1.08	1.10	1.18	1.20	1.18	1.24	1.29	1.13
,,40:*	1.00	1.05	1.06	1.06	1.06	1.12	1.18	1.17	1.20	1.25	1.11
,,50:*	1.00	1.04	1.05	1.05	1.05	1.08	1.17	1.19	1.19	1.37	1.10

800,2.4,153:1c,h1,o1, path lengths are fairly small until Level 8, when suddenly they jump from 1.05 to 1.27. We are unfortunately unable to correlate this behavior with anything from Table 7 showing routing table sizes. In fact, from visual inspection of all of the simulation data, we were not able to find any pattern between these jumps in path length and any other measured parameters.

When we study the parameters r_i and d_i in Section 6.10, we see that the ratio $\frac{r_i}{d_i}$ varies with changes in path length. In that section, we revisit the question of path length behavior.

It is important to point out that the aggregate path length for all hierarchical levels in Tables 8 and 9 assumes that there is an equal traffic distribution between all nodes. In networks where there is much more traffic between nodes which are closer to each other, the aggregate values for path length will of course improve, because they will be weighted more heavily by the path length figures for lower hierarchy levels.

Table 9
Path Length by for Nine Individual Simulations

	$\frac{P_i^{lm}}{P_i^{sh}}$ 0	1	2	3	4	5	6	7	8	9	$\frac{P^{lm}}{P^{sh}}$
:1c,h1,o1											
800,2.4,153:	1.00	1.02	1.05	1.05	1.02	1.01	1.00	1.05	1.27	1.38	1.11
800,4,36:	1.00	1.09	1.20	1.14	1.10	1.15	1.30	1.57			1.21
800,6,7:	1.00	1.06	1.28	1.60	1.76						1.51
:1d,h2,o2											
800,2.4,153:	1.00	1.01	1.03	1.03	1.01	1.00	1.03	1.28	‡		1.09
800,4,36:	1.00	1.05	1.14	1.09	1.05	1.18	‡				1.14
800,6,7:	1.00	1.04	1.34	1.55	‡						1.36
:1a,h1											
800,2.4,153:	1.00	1.01	1.02	1.01	1.00	1.00	1.00	1.04	‡	‡	1.01
800,4,36:	1.00	1.07	1.07	1.05	1.03	1.06	‡	‡			1.04
800,6,7:	1.00	1.08	1.22	‡	‡						1.13

‡ This level never used for routing

6.8 Analysis of the Number of Landmarks by Network Type and Hierarchy Type

In this section, we are interested in considering how many Landmarks are at each level T_i and how many nodes each Landmark covers: $\frac{N}{T_i}$. Recall from Equation 2 that $R_i = \frac{T_i\, v(r_i)}{N}$, so large $\frac{N}{T_i}$ results in small R_i.

Tables 10 and 11 show T_i and $\frac{N}{T_i}$ for the different hierarchy types and the different network parameters respectively. In both cases, the hierarchy types with Vicinity Extensions are not considered. This is because this data will be used in comparison with r_i later in this section. But the value r_i is obscured by the Vicinity Extension, and so Vicinity Extensions are not included in these analysis. Some of the data in Tables 10 and 11 is not valid for comparison because some of the individual simulations do not have data points at higher levels of the hierarchy, causing a shift in the overall value of N. For Table 10, the hierarchy types are listed in the same order as that given in Table 4—from small routing table sizes to large routing table sizes.

Table 10
Number of Landmarks by Hierarchy Type

	T_i (N/T_i)									
	0	1	2	3	4	5	6	7	8	9
,,*:1c,h1,o1	466.67 1.00	159.56 3.00	90.22 5.77	33.22 17.41	12.48 45.14	4.19 126.29	1.70 309.88	1.00 ‡		
,,*:1d,h2,o1	466.67 1.00	172.74 2.79	53.81 10.20	20.89 25.72	6.89 72.51	2.41 219.51	1.13 ‡	1.00 ‡		
,,*:1d,h2,o2	466.67 1.00	172.74 2.79	53.81 10.20	20.89 25.72	6.89 72.51	2.41 219.51	1.13 ‡	1.00 ‡		
,,*:2,t2,c2	466.67 1.00	233.33 2.00	116.67 4.00	58.33 8.00	29.00 16.22	14.33 32.89	7.00 66.67	3.33 155.56	2.00 ‡	1.00 ‡
,,*:2,t3,c2	466.67 1.00	155.00 3.02	51.33 9.09	16.67 28.24	5.00 96.30	2.00 ‡	1.00 ‡			
,,*:2,t2,c3	466.67 1.00	233.33 2.00	116.67 4.00	58.33 8.00	29.00 16.22	14.33 32.89	7.00 66.67	3.33 155.56	2.00 ‡	1.00 ‡
,,*:2,t3,c3	466.67 1.00	155.00 3.02	51.33 9.09	16.67 28.24	5.00 96.30	2.00 ‡	1.00 ‡			
,,*:1b,h2	466.67 1.00	172.74 2.79	53.81 10.20	20.89 25.72	6.89 72.51	2.41 219.51	1.13 ‡	1.00 ‡		
,,*:1a,h1	466.67 1.00	159.56 3.00	90.22 5.77	33.22 17.41	12.48 45.14	4.19 126.29	1.70 309.88	1.00 ‡		

‡ Data not valid because of changing N

Both Tables 10 and 11 have what initially appear to be surprises. In Table 10, we see that there is no correlation between T_i and R_i. If there were, then we would see small values of T_i and large values of at the top of Table 10. Instead high and low values are sprinkled throughout.

Table 11
Number of Landmarks by Network Type

	T_i (N/T_i)									
	0	1	2	3	4	5	6	7	8	9
200,*,s:*‡	200.00	75.01	34.09	15.21	6.74	3.79	1.98	1.00		
	1.00	2.78	6.79	16.53	45.22	72.70	125.15	200.00		
400,*,*:*‡	400.00	155.17	65.57	27.23	11.20	4.53	2.62	2.20	1.00	
	1.00	2.66	7.54	19.94	56.88	182.88	266.67	240.00	400.00	
800,*,*:*‡	800.00	307.81	126.31	50.59	19.94	8.16	3.73	3.73	3.00	1.00
	1.00	2.69	8.44	25.01	75.52	234.35	553.85	436.36	266.67	800.00
,2.4,:*‡	466.67	192.41	95.93	40.53	15.26	5.95	2.86	2.27	2.00	1.00
	1.00	2.48	5.50	14.63	45.92	‡‡	‡‡	‡‡	‡‡	‡‡
,4,:*‡	466.67	181.49	70.42	27.48	11.49	5.53	2.89	2.27	2.00	1.00
	1.00	2.67	7.59	21.46	63.59	‡‡	‡‡	‡‡	‡‡	‡‡
,6,:*‡	466.67	164.10	59.62	25.02	11.12	5.41	2.86	2.27	2.00	1.00
	1.00	2.98	9.69	25.39	68.12	‡‡	‡‡	‡‡	‡‡	‡‡
,,vs:*‡	466.67	179.28	72.04	27.57	10.89	6.79	4.60	3.33	2.00	1.00
	1.00	2.68	10.51	55.72	‡‡	‡‡	‡‡	‡‡	‡‡	‡‡
,,30:*‡	466.67	178.73	73.68	29.99	11.98	5.00	2.80	3.33	2.00	1.00
	1.00	2.75	8.15	22.72	64.85	‡‡	‡‡	‡‡	‡‡	‡‡
,,40:*‡	466.67	181.80	77.33	31.53	12.42	5.77	2.83	2.17	2.00	1.00
	1.00	2.63	7.15	19.63	60.68	‡‡	‡‡	‡‡	‡‡	‡‡
,,50:*‡	466.67	177.47	74.95	31.52	13.48	6.12	2.99	1.93	2.00	1.00
	1.00	2.76	7.47	19.13	52.09	‡‡	‡‡	‡‡	‡‡	‡‡

‡ Vicinity Extensions not included
‡‡ Data not valid because of changing N

In Table 11, we see trends in all three parameter groups, but all three trends are in the opposite direction of what is expected. In each case, we see small $(\frac{N}{T_i})$ where we have small R_i: in networks with fewer nodes, smaller node degrees, and larger diameters.

On reflection, however, we see that this is not a surprise at all. Recall from Equation 3 that $v(\hat{d}_i) = \frac{N}{T_i}$, where $\hat{d}_i \approx d_i$. Then the numerator and the denominator of $R_i \approx \frac{v(r_i)}{v(d_i)}$ are the same function. If R_i is large for some given hierarchy it not because $v(r_i)$ is large and $v(d_i)$ is small, but because $v(r_i)$ gets large faster than $v(d_i)$.

Notice that in Table 10, several pairs of columns are identical. This is because the Landmarks were assigned the same way for these hierarchies. The difference between them is in how r_i is modified after the Landmarks are assigned.

The sharp-eyed reader may also notice a couple of apparent inconsistencies in Table 11. In the experiments *,*,vs:* and *,*,30:*, level 3, we see that the values of T_i are similar, but the values of $(\frac{N}{T_i})$ are very different. Also notice that for experiment 400,*,*:*, level 7, and experiment 800,*,*:*, levels 7 and 8, the value of $(\frac{N}{T_i})$ decreases where it otherwise has been increasing. In both cases, the reason for this is that a large number of the values of T_i are 1, which causes a very high $\frac{N}{T_i}$, twice as high as the case where $T_i = 2$, for instance. For experiments 400,*,*:* and 800,*,*:*, the high $\frac{N}{T_i}$ (that is, $T_i = 1$) occurred at the previous level.

6.9 Analysis of the Landmark Radii and Distance Between Landmarks by Network Type and Hierarchy Type

In this section we analyze r_i, d_i, and $\frac{r_i}{d_i}$ for different hierarchy types and different network types. Tables 12 and 13 show this data. As with Table 10, the hierarchies in 12 are listed in the same order as in Table 4—small routing tables to large routing tables. Also as with Tables 10 and 11, 12 and 13 only consider those hierarchy types with no Vicinity Extensions.

Table 12 gives us favorable data. Notice that small and large values of r_i and d_i appear throughout Table 12 for any given hierarchy level. However, the values $\frac{r_i}{d_i}$ consistently move from small to large from the top of Table 12 to the bottom. In other words, for identical networks, as the routing table R_i grows, so the the value $\frac{r_i}{d_i}$. From Equation 4, we see that this is exactly as we would expect.

Table 12
Landmark Radii and Landmark Distances by Hierarchy Type

	r_i^{av} d_i^{av} $(r_i / d_i)^{av}$									
	0	1	2	3	4	5	6	7	8	9
,,*:1c,h1,o1	1.00	1.90	3.00	5.25	10.09	20.27	35.56	45.00		
	0.00	0.64	1.20	2.26	4.02	7.67	13.99	22.15		
	∞	2.96	2.59	2.35	2.53	2.69	2.57	2.03		
,,*:1d,h2,o1	1.00	2.14	4.06	7.15	14.70	30.80	40.96	50.00		
	0.00	0.57	1.43	2.61	5.09	10.77	18.83	25.24		
	∞	3.79	2.89	2.78	2.92	2.89	2.20	1.98		
,,*:1d,h2,o2	2.00	2.14	5.06	9.15	18.70	35.93	41.25	50.00		
	0.00	0.57	1.57	2.85	5.38	10.89	18.83	25.24		
	∞	3.79	3.27	3.24	3.52	3.35	2.22	1.98		
,,*:2,t2,c2	1.66	2.47	3.51	5.74	9.20	12.51	24.28	33.51	40.00	40.00
	0.00	0.54	0.98	1.69	2.78	4.30	6.89	12.35	15.80	18.64
	∞	4.55	3.57	3.31	3.30	2.88	3.47	3.00	2.70	2.14
,,*:2,t3,c2	2.01	3.83	7.83	16.98	33.47	40.00	40.00			
	0.00	0.85	2.10	4.22	10.17	15.18	19.42			
	∞	4.46	3.64	3.93	3.49	2.95	2.05			
,,*:2,t2,c3	2.07	3.07	4.71	7.25	11.93	19.40	29.30	35.59	40.00	40.00
	0.00	0.59	1.19	2.06	3.08	5.18	8.29	13.09	15.55	18.66
	∞	5.17	3.94	3.46	3.84	3.74	3.81	3.02	2.79	2.18
,,*:2,t3,c3	2.53	4.74	10.33	25.98	35.53	40.00	40.00			
	0.00	0.91	2.13	4.39	10.38	16.52	18.65			
	∞	5.20	4.86	5.86	3.85	2.63	2.18			
,,*:1b,h2	2.00	4.00	8.00	16.00	31.33	40.00	41.25	50.00		
	0.00	0.63	1.79	3.19	5.77	11.02	18.83	25.24		
	∞	6.37	4.53	5.06	5.50	3.72	2.22	1.98		
,,*:1a,h1	2.00	4.00	8.00	16.00	31.63	40.00	40.00	45.00		
	0.00	0.66	1.28	2.43	4.19	7.77	14.10	22.15		
	∞	6.09	6.50	6.68	7.65	5.32	2.87	2.03		

Notice that this smooth increase in $\frac{r_i}{d_i}$ begins to break up a little at levels 4 and higher. This is because in this range, we begin to have fewer data points, and not all of the networks are represented fully, causing a skew in some of these values. Also, in this

Table 13
Landmark Radii and Landmark Distances by Network Type

	r_i^{av} d_i^{av} $(r_i / d_i)^{av}$									
	0	1	2	3	4	5	6	7	8	9
200,*,*:*	1.86	3.50	7.30	15.18	24.42	30.05	40.30	43.33		
	0.00	0.67	1.52	3.05	6.76	8.45	14.28	21.51		
	∞	5.14	4.69	4.72	4.24	3.70	3.02	2.03		
400,*,*:*	1.80	3.04	5.75	11.88	22.98	31.21	34.96	42.00	40.00	
	0.00	0.66	1.52	2.87	5.44	10.73	14.04	15.37	19.41	
	∞	4.61	3.81	4.02	4.22	3.23	2.81	3.05	2.06	
800,*,*:*	1.76	2.88	5.11	9.43	18.13	29.51	34.23	33.81	40.00	40.00
	0.00	0.66	1.51	2.65	4.75	8.95	14.98	14.42	11.94	18.65
	∞	4.37	3.43	3.49	3.74	3.32	2.46	2.69	3.43	2.16
,2.4,:*	2.06	3.48	6.41	12.48	21.66	30.21	35.91	39.11	40.00	40.00
	0.00	0.68	1.53	3.18	6.17	9.92	14.45	17.36	16.31	17.68
	∞	5.10	4.33	3.86	3.61	3.13	2.60	2.39	2.58	2.29
,4,:*	1.74	3.00	6.03	11.92	21.96	29.69	36.02	40.22	40.00	40.00
	0.00	0.65	1.53	2.78	5.29	9.11	14.22	17.14	14.78	18.63
	∞	4.60	3.88	4.06	4.28	3.48	2.80	2.66	2.95	2.16
,6,:*	1.63	2.94	5.73	12.10	21.90	30.90	36.70	39.93	40.00	40.00
	0.00	0.66	1.49	2.61	5.48	9.32	14.77	17.37	15.93	19.64
	∞	4.43	3.73	4.31	4.31	3.57	2.79	2.63	2.70	2.03
,,vs:*	1.74	2.84	4.73	7.03	9.70	12.76	11.94	10.22	13.50	14.33
	0.00	0.66	1.60	2.85	4.41	6.05	5.79	4.73	6.35	6.64
	∞	4.34	3.05	2.49	2.15	2.07	2.14	2.06	2.04	2.07
,,30:*	1.77	2.95	5.58	10.64	18.90	24.50	27.03	26.17	30.00	30.00
	0.00	0.66	1.49	2.72	5.02	8.78	11.58	9.96	12.43	13.71
	∞	4.43	3.78	3.79	3.83	2.94	2.55	2.85	2.57	2.22
,,40:*	1.84	3.25	6.09	11.98	22.44	31.05	35.82	37.14	40.00	40.00
	0.00	0.66	1.54	2.89	5.81	9.43	14.90	16.63	16.77	18.91
	∞	4.84	3.95	3.99	4.06	3.44	2.64	2.42	2.52	2.12
,,50:*	1.81	3.23	6.49	13.88	24.20	35.25	44.58	47.28	50.00	50.00
	0.00	0.66	1.52	2.95	6.12	10.14	16.58	20.75	17.82	23.33
	∞	4.85	4.20	4.44	4.32	3.80	2.99	2.55	3.15	2.15

range, values of r_i are being limited by the network diameter. This ceiling occurs at different levels for different networks, again causing a skew in the values. However, because most routing table entries come from the first few levels, we are not so concerned with the behavior of $\frac{r_i}{d_i}$ at the higher levels.

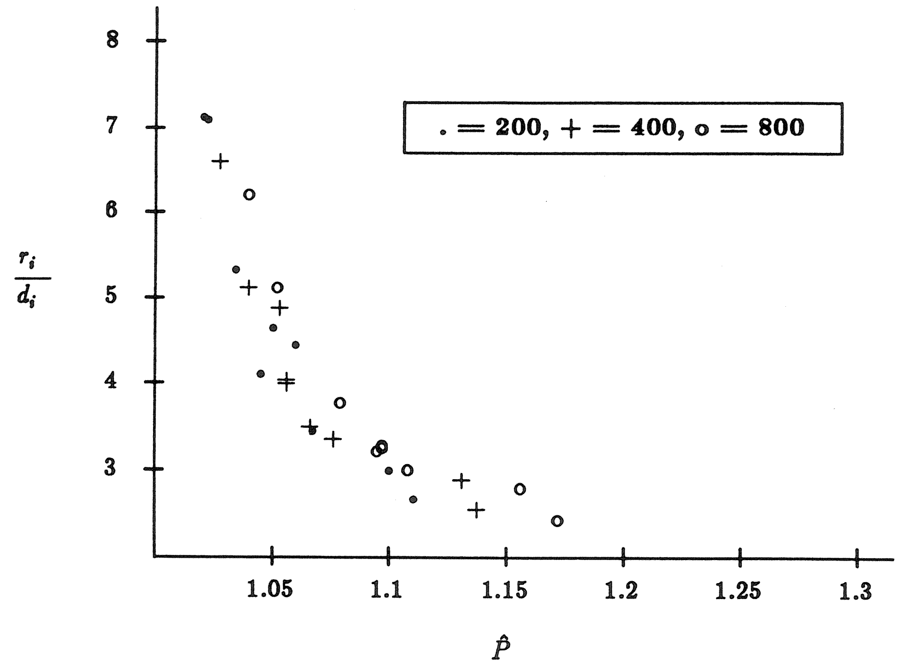
Table 13 shows some interesting data. Here we see that, in all three parameter groups, the ratio $\frac{r_i}{d_i}$ gets smaller with larger table sizes for any given hierarchy level. This is true up through level 4. At first glance, this seems to contradict with Table 12, where $\frac{r_i}{d_i}$ increased with increased R_i. Recall, however, that the routing table size is due to $\frac{v(r_i)}{v(d_i)}$, not $\frac{r_i}{d_i}$. In Table 12, the hierarchy types are being compared using the same networks, so the function $v(x)$ is the same for each comparison. In this case, $\frac{r_i}{d_i}$ must change to bring about a change in R_i. However, in Table 13, the function $v(x)$ is different for the different network types, and $\frac{v(r_i)}{v(d_i)}$ dominates over $\frac{r_i}{d_i}$.

If we compare Tables 12 and 13 with Tables 4 and 5, we see that the ratio $\frac{r_i}{d_i}$ follows the inverse of path length. In other words, large $\frac{r_i}{d_i}$ means small path lengths. This is the only parameter we have seen which consistently varies as $\hat{P}$ varies. As such, we are interested in considering this relationship in more detail.

Figures 9, 10, and 11 show graphically the relationship between $(\frac{r_i}{d_i})$ and $\hat{P}$ for different numbers of nodes, different diameters, and different node degrees respectively. Each graph is scaled the same, so they are visually comparable.

Right away we notice the asymptotic shape of the curves, with the y-asymptote being 1 for shortest path. It is not as clear where the x-asymptote should be. The smallest $\frac{r_i}{d_i}$ we achieved was right at 2. Values of $\frac{r_i}{d_i}$ much less than this would

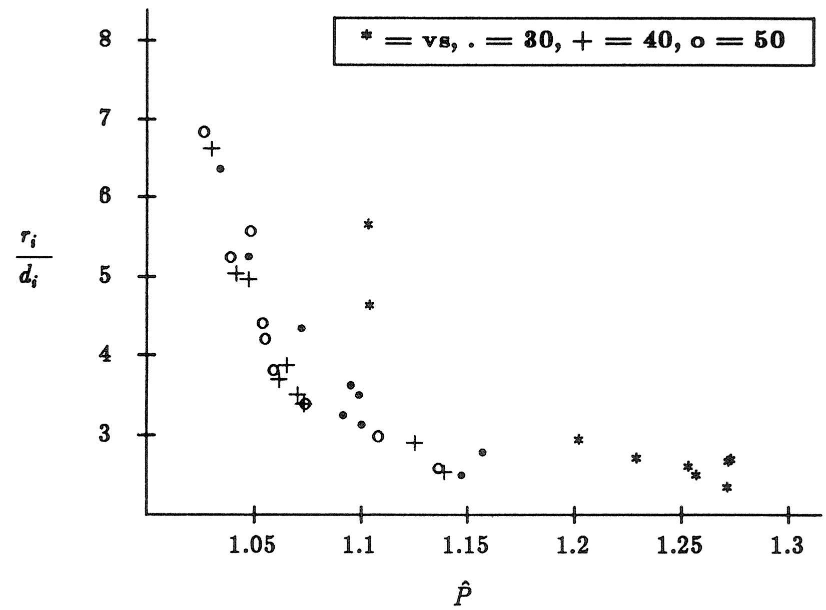
Figure 9
Path Length by Number of Nodes



constitute a broken hierarchy, because the level i Landmark Vicinities would not reach the level $i+1$ Landmarks (assuming that d_{i+1} is roughly twice d_i). In this case, $\hat{P} = \infty$, because nodes become unreachable.

In Figures 9 and 11, we notice that all of the data points fall nearly on the same curve. There is a slight shift right (longer path lengths) from smaller to larger numbers of nodes, and from smaller to larger node degree, but this shift is only on the order of .02 or .03. In Figure 10, we see similar behavior for diameters of 30, 40, and 50, but the very small diameters show a clear departure from the others towards longer path lengths.

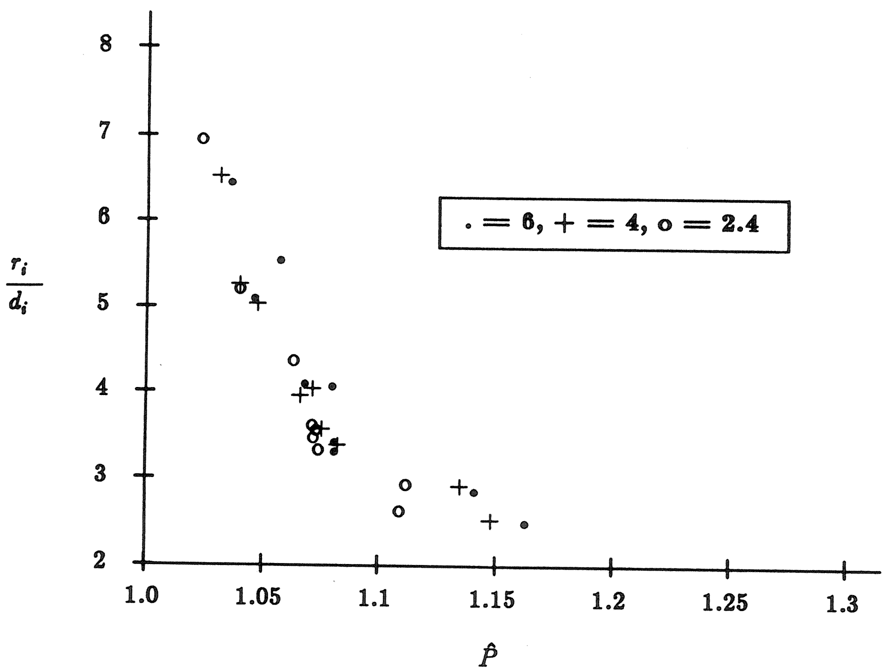
Figure 10
Path Length by Diameter



Since this represents our worst performance with regards to path lengths, we wish to consider it more closely.

It is interesting to note that the small diameter networks show worse *relative* $(\frac{P^{lm}}{P^{sh}})$ behavior compared to shortest path, but very small diameters show better performance when *absolute* $(P^{lm} - P^{sh})$ behavior compared to shortest path is considered. For instance, values of $(P^{lm} - P^{sh})$ for $\frac{r_i}{d_i}$ around 2.5 are roughly 1.7 for very small diameters (average diameter 12.56), 2 for diameter 30, 3 for diameter 40, and 3.5 for diameter 50.

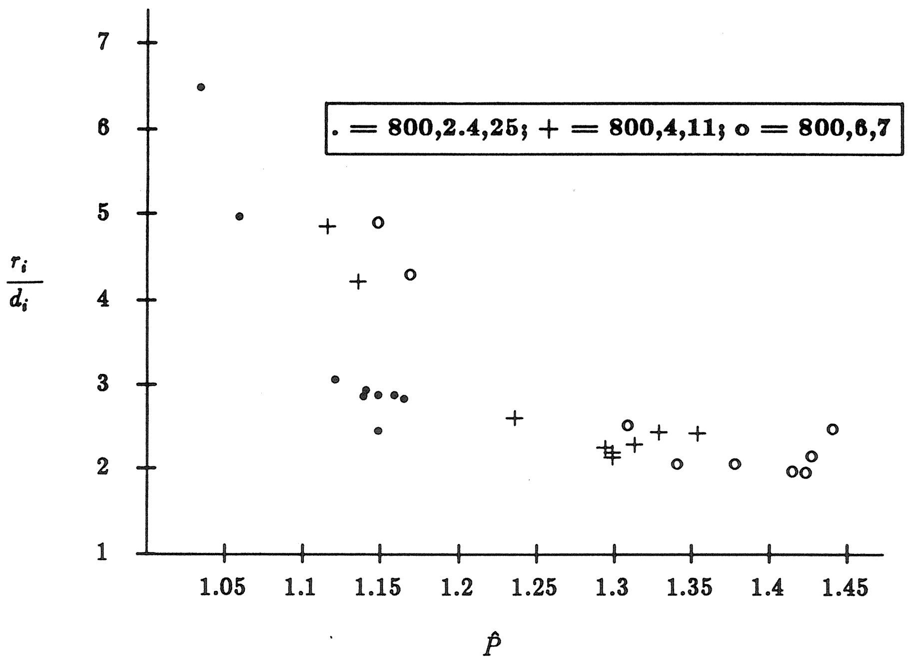
Figure 11
Path Length by Average Node Degree



In other words, networks with small diameters show worse path lengths only because they are being compared against shorter path lengths.

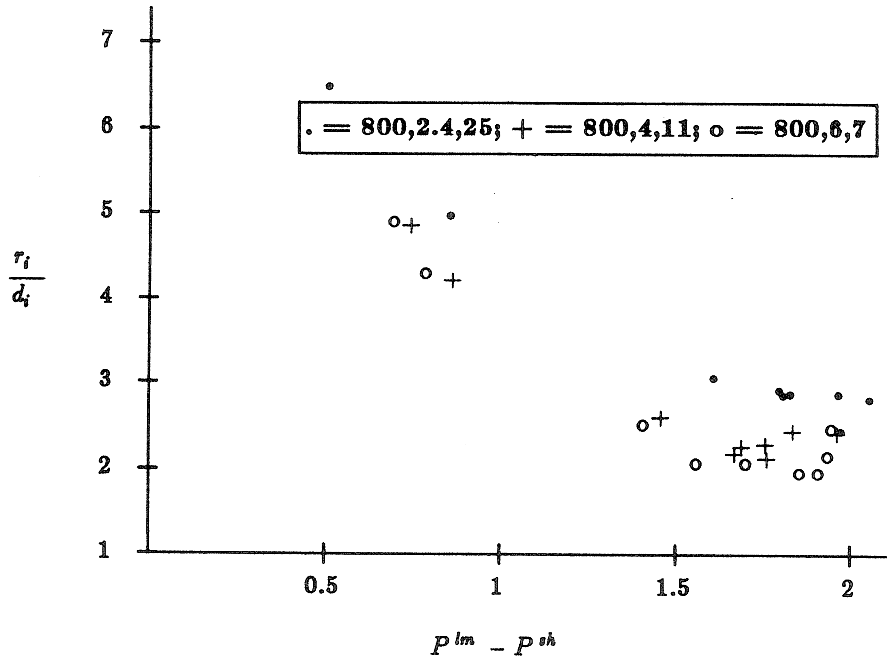
Figures 12 and 13 illustrate the difference between relative and absolute comparison of path lengths, plus show another interesting aspect of path lengths. Figures 12 and 13 show path lengths for three networks of 800 nodes each whose diameters are the smallest possible (for our network generation technique) for the given node degrees. In other words, we used the largest possible maximum span length in generating each of the networks. Figure 12 plots $\frac{r_i}{d_i}$ against $\hat{P} = \frac{P^{lm}}{P^{sh}}$, and Figure 13 plots $\frac{r_i}{d_i}$ against $P^{lm} - P^{sh}$. Notice that Figure 12 is on a different scale than Figures 9, 10, and 11.

Figure 12
Path Lengths by Smallest Achievable Diameter (Relative Comparison)



In Figure 12, we notice again that path lengths are worse for the smaller diameter network. In Figure 13, however, we see that the absolute comparison of path lengths are similar for the different diameters. This is different from what was stated previously, where absolute path lengths got *better* as relative path lengths got worse. If we adjusted the data points in Figure 13 to make up for the different node degrees, we would push

Figure 13
Path Lengths by Smallest Achievable Diameter (Absolute Comparison)



the data points for node degree 2.4 to the right, and those for node degree 6 to the left, causing slightly more difference than what is actually seen in Figure 13. This is a moot point, however, because any change in node degree for these very small diameters would change the diameter.

In Figure 13, we see a situation where shrinking the diameter can only be accomplished by increasing the node degree. While shrinking the diameter would tend to push the data points in Figure 13 to the left, increasing the node degree would push them to the right. The net effect is little or no movement at all. In other words, at least at the low diameter end where the diameter or the node degree can be lowered, but not both, the absolute difference between P^{lm} and P^{sh} remains roughly constant. For 200 node networks and $\frac{r_i}{d_i}$ =2, $P^{lm} - P^{sh}$ =1.4. This compares closely with 1.7 for 800 node networks, and shows that this lower wall is not strongly impacted by the number of nodes. This lower wall says that as the number of nodes N in a network gets larger, and the diameter gets corresponding larger by log(N), we do not expect to see path lengths get pathologically longer.

6.10 Fine-tuning the Hierarchy

If one compares Table 7 with Table 9, one sees that path lengths get larger as table sizes get smaller for increasing hierarchy levels. It seems feasible that one may be able to "tune" the hierarchy to behave optimally for certain networks. Two methods suggest themselves. The first is to increase the number of hierarchy levels so that there are more routing table entries at the higher levels, and we hope shorter paths at the higher levels. The second is to increase the number of Landmarks T_H at the highest hierarchy level H, thus decreasing the number of hierarchy levels overall, and hopefully decreasing path lengths. The first method resulted in no improvement, while the second did show improvement. These modifications are presented below.

6.10.1 Increasing the Number of Hierarchical Levels

What we hope to do, then, is change the specification of the hierarchy so that it has more levels, especially for the networks with small diameters and large node degrees. One would only want to do this, of course, if there was a significant amount of traffic between nodes far apart in the network, or if performance between nodes far apart was to be maximized.

Table 14 shows the results of attempting to fine-tune the hierarchy for network 800,6,vs:. This table gives the hierarchy parameters r_i and d_i^{max}, the routing table sizes R_i and R, and the path lengths P_i and P. It shows two new hierarchy types, :1d,h4,o1 and :1d,h5,o1, plus a previously seen one, 1d,h2,o1, shown here for comparison. These new hierarchy types increase r_i linearly with i rather than geometrically, as the previous hierarchies did. This results in more hierarchy levels, but a small $\frac{r_i}{d_i}$ at each level. We only use this modification on networks with small diameters, because there would be too many hierarchy levels if it were used on large diameter networks.

Table 14
Attempt to Fine-tune the Hierarchy

	800,6,vs:													
	0	1	2	3	4	5	6	7	8	9	10	11	12	all
:1d,h2,o1														
r_i (initial)	2	4	8	16	32									
d_i^{max}	0	1	4	8	16									
R_i	5.09	19.69	8.95	2.83	1.00									37.56
P_i	1.00	1.05	1.31	1.68	1.67									1.51
:1d,h4,o1														
r_i (initial)	1	2	3	4	5	6	7	8	9	10	11	12	13	
d_i^{max}	0	1	2	3	4	5	6	7	8	9	10	11	12	
R_i	3.57	0.98	4.78	2.70	4.74	4.35	4.56	4.45	3.29	2.68	1.00	1.00	1.00	39.11
P_i	1.00	1.01	1.04	1.15	1.18	1.30	1.35	1.52	1.67	1.73	1.76	1.74	1.65	1.57
:1d,h5,o1														
r_i (initial)	1	2	4	6	8	10	12	14						
d_i^{max}	0	1	2	4	6	8	10	12						
R_i	3.93	1.67	13.65	4.05	10.10	2.55	1.00	1.00						37.95
P_i	1.00	1.01	1.08	1.31	1.51	1.72	1.72	1.59						1.56

In both of the new hierarchy types, we reduced the vicinity sizes, resulting in more hierarchy levels. In both cases, we did achieve more hierarchy levels, and a smoothing out of the number of routing table entries at each level. This is especially true for hierarchy :1d,h4,o1. However, in both cases, the overall hierarchy does not perform as well as the original hierarchy :1d,h2,o1—in either routing table size or path length. Given the closeness of the values in Table 14, and the variation in individual experiments, we can conclude that performance is the same for the three hierarchy types tried.

We see from this experiment and previous ones that the performance of the hierarchy is very resilient to variations in the hierarchy. In Table 4, for instance, the random assignment of Landmarks did not significantly change performance. Table 14 shows this resilience even more dramatically. In experiment 800,6,vs:1d,h2,o1, there are 5 hierarchical levels; in experiment 800,6,vs:1d,h5,o1 there are 13—a difference in order of magnitude of nearly 3 times. Yet, the routing table size and path lengths are virtually identical.

The advantage of this is that one can afford a lot of error in building a Landmark Hierarchy dynamically in a network. Changes in the hierarchy will not greatly impact the overall performance of the hierarchy. In fact, it seems entirely feasible that nodes in a network could dynamically adjust the routing table sizes and path lengths by adding and deleting Landmarks, and by adjusting the vicinity sizes.

We also notice that in experiments 800,6,vs:1d,h2,o1 and 800,6,vs:1d,h5,o1, we see sudden jumps in the path lengths, between levels 1 and 2 in the former case, and between levels 2 and 3 in the latter. In experiment 800,6,vs:1d,h4,o1, we do see such a jump. Path lengths increase in a smoother fashion. Even so, path lengths between the three are essentially the same. This implies that it may not be of any benefit to try to design the hierarchy to avoid these sudden jumps.

6.10.2 Increasing the Number of Landmarks at the Highest Level

If one looks at Figure 8, the algorithm for assigning Landmarks in our simulations, one notices that the algorithm stops when there is only one Landmark at the highest level. We chose this number because we know it will be easy to end a distributed algorithm with one Landmark at the highest level, because this single Landmark would recognize that it was the only Landmark, and would not run any more elections. If, however, we allow for multiple Landmarks T_H at the highest level H, we will in effect reduce the number of levels, because the hierarchy building algorithm will end earlier.

In Table 15, we show the results of experiments modifying the Landmark assignment algorithm so that it ends when there are 1) five or less Landmarks at a level, and 2) ten or less Landmarks at a level. We ran this experiment on all network types, but only three hierarchy types, :1c,h1,o1, :2,t2,c2, and :1a,h1. These three hierarchy types represent the smallest routing table sizes (and longest path lengths), the largest routing table sizes (and shortest path lengths), and something in between. It also represents both Approaches 1 and 2. Since the hierarchy types represent the full range of hierarchy types, we can use just these three in our averages for the different network types and still compare these results to previous results where all 21 hierarchy types are averaged. In Table 15, we additionally give the results from Tables 4 and 5 for easy comparison.

Table 15
Comparison of Number of Level H Landmarks

	$T_H = 1$‡‡		$T_H \leq 5$		$T_H \leq 10$	
	R	$\hat{P}$	R	$\hat{P}$	R	$\hat{P}$
,,*:1c,h1,o1	14.50	1.16	15.45	1.10	17.79	1.07
,,*:2,t2,c2	22.98	1.10	22.98	1.10	24.41	1.07
,,*:1a,h1	46.88	1.03	46.85	1.03	47.12	1.03
200,*,*:*‡	17.84	1.10	19.19	1.06	20.98	1.03
400,*,*:*‡	23.60	1.11	26.83	1.07	27.80	1.05
800,*,*:*‡	31.46	1.14	39.26	1.10	40.54	1.08
,2.4,:*‡	23.30	1.12	31.23	1.09	32.36	1.07
,4,:*‡	23.30	1.11	26.96	1.08	28.62	1.05
,6,:*‡	25.64	1.12	27.09	1.07	28.34	1.05
,,vs:*‡	43.47	1.28	59.02	1.26	59.53	1.26
,,30:*‡	26.71	1.13	32.11	1.10	33.43	1.08
,,40:*‡	23.56	1.11	27.68	1.07	28.91	1.05
,,50:*‡	22.63	1.10	25.50	1.06	26.98	1.04

‡ Averaged over hierarchies :1c,h1,o1, :2,t2,c2, and :1a,h1 only
‡‡ Averaged over all hierarchy types (data from Tables 4 and 5)

Looking at the first three columns, we see that additional T_H results in substantial improvements for experiments :1c,h1,o1 and :2,t2,c2, and no substantial difference for experiment :1a,h1. In experiment :1c,h1,o1, $T_H \leq 10$ shows a decrease in path length $\hat{P}$ over $T_H = 1$ of more than 50%, while increasing routing table size R by only 23%. For experiment :2,t2,c2, we have a decrease in $\hat{P}$ of 30% with an increase in R of only 6%.

Clearly, increasing T_H has a tendency to decrease path lengths at the expense of routing table sizes. This tendency is stronger with hierarchies which exhibit large path lengths. However, in general, the decrease in path length is greater that the increase in routing table size, resulting in an overall improvement. As before, the real improvement depends on the traffic pattern and frequency of routing updates seen in a particular network.

While in general we see improvements in the experiments for various network types, these improvements are not as good for those networks for which we are most concerned—namely, networks with a large number of nodes, and small diameters. For instance, for the 200 node networks, increase in R for $T_H \leq 10$ over $T_H = 1$ is 18% while decrease in $\hat{P}$ is 70%. However, for the 800 node networks, increase in R is 29%, while the decrease in $\hat{P}$ is only 42%. Worse still, for the very small diameter experiments, performance with $T_H \leq 10$ is worse than that for $T_H = 1$. Here, increase in R is 36%, while decrease in $\hat{P}$ is only 7%.

Overall, adjusting T_H seems to be a powerful tool for tuning and improving the performance of the Landmark Hierarchy. However, it must be used with care, for it may hurt performance in some cases. We regret not having more time to study this modification. We believe that a good understanding of this parameter (T_H) according to network type and hierarchy type could yield excellent performance improvements in most situations. This is clearly a good topic for further study.

7.0 OVERALL PERFORMANCE OF THE LANDMARK HIERARCHY

In Section 6, we analyze the Landmark Hierarchy by isolating certain network and hierarchy parameters and studying them independently. In real networks, networks parameters are not isolated—each parameter impacts the others. For instance, it is commonly accepted that network diameters grow roughly as the log of the number of nodes (McQuillan, Richer, Rosen, 1980; Kamoun, Kleinrock, 1977). The exact growth depends on the network in question. Land-based packet radio networks, where connectivity tends to be to nearby neighbors, will exhibit large diameters, whereas global land-based and satellite-based networks will exhibit smaller diameters. Further, node degrees will vary from situation to situation.

Secondly, our simulations were by necessity for relatively small networks—800 nodes. Global data networks will (and in some cases, now do) have many thousands of nodes. We would like to be able to realistically predict the performance we might see for networks of these sizes.

To do this, we have created a final set of experiments. These experiments have networks with 50, 100, 200, 400, and 800 nodes, and diameters which grow as the log of the number of nodes. With this progression of network sizes, we generate data for r_i and d_i, as well as for R and $\hat{P}$. Using the resulting r_i and d_i, and a function which estimates $v(x)$, we then predict routing table sizes R for any size network.

7.1 Experiment Description

As stated, in this experiment, we study networks with 50, 100, 200, 400, and 800 nodes. We wish to obtain results for networks with relatively large diameters and relatively small diameters, in order to satisfy the largest possible set of requirements. The two groups of five networks studied are 1) 50,5,8:, 100,5,11:, 200,5,14:, 400,5,17:, and 800,5,20:, and 2) 50,4,10:, 100,4,20:, 200,4,30:, 400,4,40:, and 800,4,50:. (See Section 5.2 for an explanation of this nomenclature.) In both groups, we have diameters progressing proportionately to $\log_2 N$. For the small diameter group, $D = 2.07\log_2 N$, and for large diameter group, $D = 5.19\log_2 N$.

We have chosen hierarchies :1c,h1,o1, :2,t2,c2, and :1a,h1. These three hierarchies give a small table size, a medium table size, and a large table size (and conversely, large path length increases, medium path length increases, and small path length increases) respectively.

We have chosen different values of T_H, also proportional to $\log_2 N$; $T_H^{50} = 2$, $T_H^{100} = 4$, $T_H^{200} = 6$, $T_H^{400} = 8$, and $T_H^{800} = 10$.

Finally, we have scaled the results for $\hat{P}$ to reflect a more realistic traffic matrix. We assume that traffic between node pairs decreases with distance; in this case, traffic is cut in half for each increase in hierarchy level. This obviously improves our overall average path lengths. Nodes far apart, however, will still see a disproportionate increase in path lengths. Again, this also holds true for the area hierarchy.

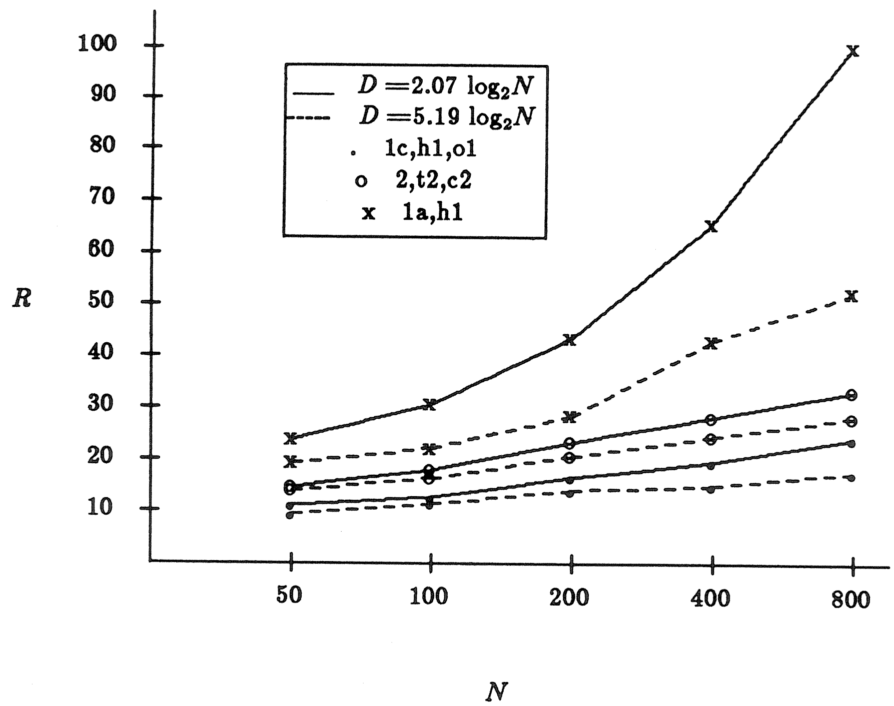
We believe that these diameters and traffic matrix are realistic. For instance, consider the DoD Internet. A perfectly realistic path from one host to another may cover say 3 hops to get to a MILNET gateway, 5 hops to a MILNET-ARPANET mail bridge, another 5 hops to an ARPANET gateway, 3 more hops to a mail-translating host, 2 hops to a TYMNET gateway, 5 hops through TYMNET, another 3 hops to some destination. Here we have a 23 hop path (about 8 or 9 of which were IP gateways) which never even left the USA. With a little ingenuity, I believe one could find paths in existence today which consisted of 2 or 3 times that many hops. Obviously, paths like this are used very rarely, and our traffic matrix assumes this. One might argue that the above stated path is only that long because of administrative constraints placed on current routing, and that in fact a much shorter *physical* path almost certainly exists. However, the forwarding of routing updates in a Landmark Hierarchy would reflect administrative boundaries. In essence, when we talk about diameters of this length, we are talking about logical diameters due to administrative restrictions placed on paths, not pure physical diameters.

We ran each of the 30 simulation 5 times, in order to get an accurate average for each one.

7.2 Experiment Results

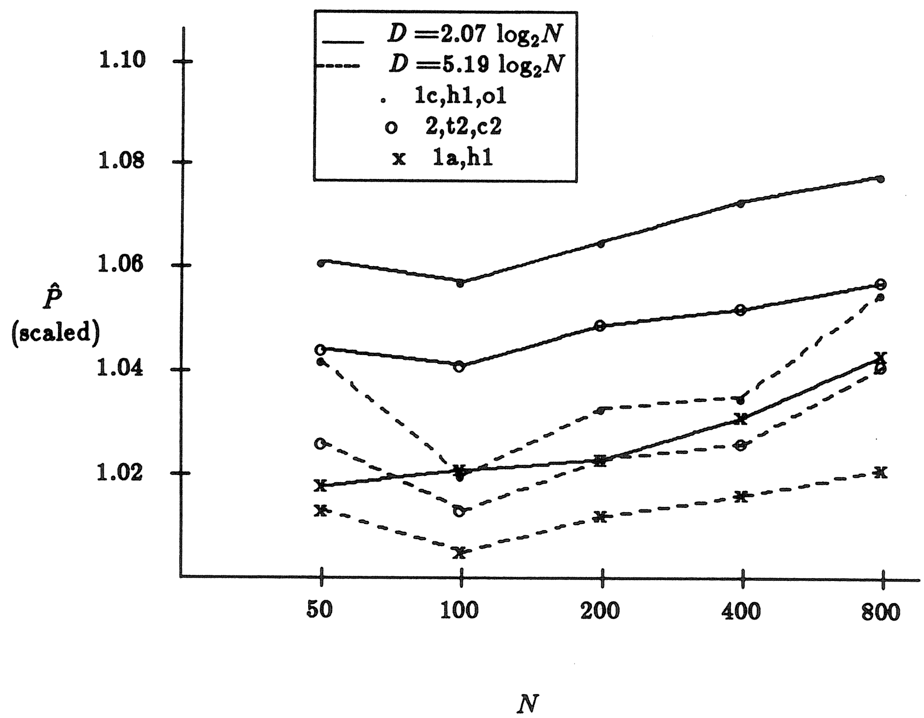
Figure 14 shows the results for routing table size R, and Figure 15 shows the results for path length $\hat{P}_{scaled}$. Each of these Figures are plotted with the x-axis (number of nodes) on a log scale.

Figure 14
Routing Table Size for Realistic Networks



Clearly, the :1a,h1 experiments perform the worst. Both are increasing faster than $\log_2 N$. The :1c,h1,o1 and :2,t2,c2 experiments, on the other hand, are increasing much slower, and for this plot, are virtually linear. These results are better than we saw in the earlier experiments when diameter did vary with an increase in the number of nodes.

Figure 15
Path Lengths for Realistic Networks and Scaled Traffic Matrix



The data for path lengths does not have smooth curves like the data for routing table sizes. Again, the 50 node network experiments deviate the most. The small diameter experiments :1c,h1,o1 and :2,t2,c2 show a stepping pattern—high at 50 nodes, low at 100, high again at 200, less high at 400, and more high at 800 nodes. This behavior is also seen in the values of $\frac{r_i}{d_i}$ for these two experiments, except in reverse—small $\frac{r_i}{d_i}$ for large $\hat{P}_{scaled}$. Both r_i and d_i, in addition to their ratio, have this behavior.

This oscillation can be seen to a much lesser extent in Figure 16. We have no explanation for this behavior.

7.3 Estimated Performance

Before we can estimate the performance of the Landmark Hierarchy, we do to estimate the function $v(x)$.

7.3.1 Estimating the Function v(x)

As discussed in Section 6.5, the function $v(x)$ initially increases exponentially with x, and then eventually reaches saturation, at which point it increases linearly (see Figures B1 through B9). When x is close to the diameter of the network, then $v(x)$ flattens out and increases almost not at all. The initial exponential fan-out is given in Equation 14 as:

$$C \times (C-1)^{x-1} \qquad 15$$

where C is the node degree.

We do not know the rate at which $v(x)$ levels off when x reaches the network diameter. However, we do not care too much, because there is very little contribution to R from the upper hierarchy levels; that is, from large x. Since the curves in Figures B4 through B6 ($\hat{v}(x) = v(x) - v(x-1)$) appear to be approximately symmetrical, we will assume that $\hat{v}(x)$ tapers off exponentially at the same rate as the initial fan-out, that is, $\hat{v}(D-x) = v(x)$ ($v(0) = 1$).

Having made this assumption, we are now able to fully describe $v(x)$, where $v(x) = \sum_{x=1}^{N} \hat{v}(x)$. Given N, D, and C, an algorithm for determining $\hat{v}(x)$, $1 \leq x \leq D$ is as follows.

1. Let $v(0) = v(D) = 1$.
2. Let $j = 1$.
3. If $C \times (C-1)^{j-1} \geq \frac{(N - (v(j-1) \times 2))}{D} - (i \times 2) + 1$, then let all $\hat{v}(x) = \frac{(N - (v(j-1) \times 2))}{D - (j \times 2) + 1}$, with $(j \leq x \leq D-j)$. Otherwise, increment j,

and repeat step 3. If $j > D/2$, then the chosen diameter D is too small for the chosen node degree C.

Note that it is possible to create networks with diameters much smaller than above algorithm can handle. However, for the class of networks we are considering—those with a quasi-random connectivity—this algorithm is appropriate.

Using the χ^2 (chi-square) calculation, we compared the $\hat{v}(x)$ generated by our algorithm with those measured from our simulated networks. For most cases, we got $\chi^2 \leq 2$ with 9 or more degrees of freedom, indicating a very good fit. On networks with very small diameters (those where $\hat{v}(x)$ never non-hierarchicaltened out), we did not do as well. However, all of our generated $\hat{v}(x)$ had fan-outs which either peaked sooner or peaked higher than those seen in the simulated networks, and so will give us worse results than might expected from simulations, which is acceptable.

7.3.2 Estimating the Landmark Radii and Landmark Distances

Of course, to predict the performance of R, we need to determine the expected of values of r_i and d_i. We have found that we can engineer the Landmark Hierarchy to give us varying values of $r_i - d_i$ ranging upwards from approximately 3. We have also found that we have significant control over the progression of r_i and d_i with increasing i (i.e., in powers of 2, in powers of three, proportional with i, and so on). Therefore, we can simply pick values of r_i and $\frac{r_i}{d_i}$, and calculate $R_i = \frac{v(r_i)}{v(d_i)}$ from our generated $v(x)$. This value for R_i will result in routing table sizes greater than what one would realistically expect. This is because d_i is less than the $\hat{d}_i$ given in Equation 3. However, the estimated results should be proportional to the simulated results.

Looking at the data for r_i and d_i from the experiments in this Section 6.9, we find that r_0 roughly in powers of 2 with increasing i. We also find that for the :1a,h1 experiment, $\frac{r_i}{d_i}$ is roughly 7; 4 for the :2,t2,c2 experiment, and 3 or slightly less for the :1c,h1,o1 experiment.

7.3.3 Estimation Experiment Results—Routing Table Sizes

Using values of 7, 4, and 3 for all $\frac{r_i}{d_i}$, and with $r_0 = 1.5$, we calculated R for networks ranging from 100 nodes to 1,600,000 nodes, in powers of 2. We did this for the two diameters studied in this Section, namely $D = 2.07 \log_2 N$ and $D = 5.19 \log_2 N$.

Figure 16 compares the :2,t2,c2 experiments for the simulated experiment and for the estimated experiment. We see that our estimated values are within an order of magnitude of the simulation results. As mentioned, we expect our estimated results to be somewhat larger than, but proportional to, our simulated results. Since our estimated results are greater than our simulated results, we feel safe in using them for estimating networks with more than 800 nodes.

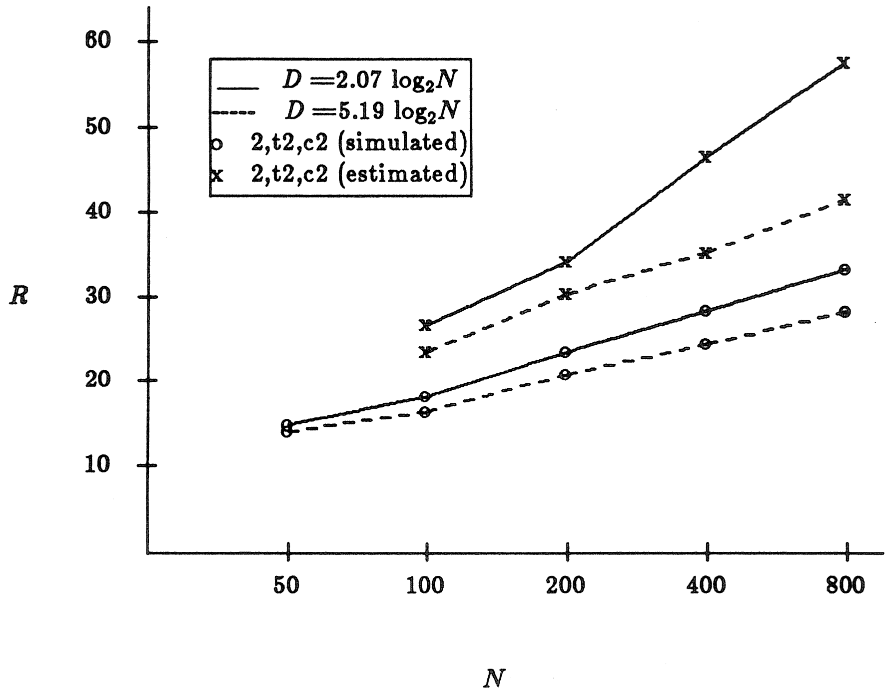
Figure 17 shows the results of our estimates for networks larger than 800 nodes. This is plotted on a log-log scale. We see that R grows as kN^m, where k and m are constants. For the small diameter experiments, m ranges from .33 (cube root) to .54 (roughly square root). For the large diameter experiments, m ranges from .38 to .62. Both ranges encompass .5, the square root. In fact, for the :2,t2,c2 experiment, which gives us our intermediate results (lines 3 and 4 in Figure 17), we have $m = .42$ and $m = .48$, both slightly better than but close to the square root.

We can safely say, then, that for what we consider networks with reasonable diameters, one can achieve routing table sizes of the square root of the the number of nodes times some constant 3 or greater. In other words, $R = 3\sqrt{N}$.

7.3.4 Estimating Path Lengths

Unfortunately, we do not have a method of predicting path lengths for networks larger than those we can simulate. On one hand, most of the lines in Figure 15 do not exhibit upward exponential trends. Two of the lines, large diameter :1a,h1, and small diameter 2,t2,c2, appear to almost have a decreasing slope. The three lines which exhibit the stepping behavior, large diameter :1c,h1,o1 and :2,t2,c2, and small diameter 1c,h1,o1, are nearly linear. We fit lines at the three points equidistant between the data peaks at 100, 200, 400, and 800 nodes for each of the three experiments. In all cases, there was less that 3% error between the lines at the equidistant points. On the other hand, the lines

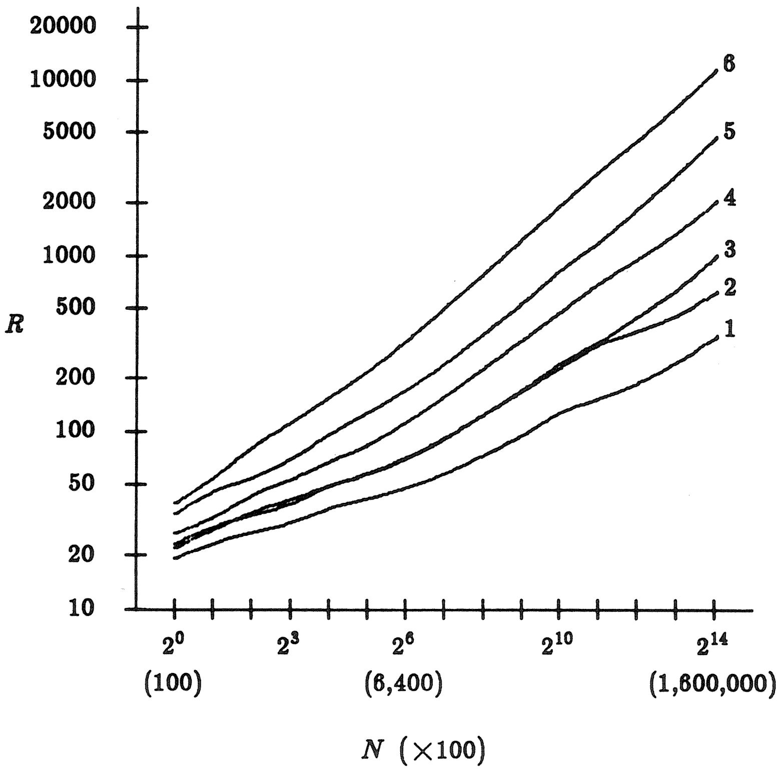
Figure 16
Comparison of Simulated and Estimated Routing Table Size



may appear linear in Figure 15, but exponentially increase for networks much larger than 800 nodes.

We would like to make three points with regards to path lengths. First, path lengths in the Landmark Hierarchy behave similarly to path lengths in the area hierarchy. Both show poorer performance at higher levels of the hierarchy, both perform worse for small diameter networks, and both perform worse with smaller table sizes. This is explored in Sections 6.7 and 6.10. However, we can assume that path lengths in the Landmark Hierarchy are roughly the same as those in the area hierarchy.

Figure 17
Estimated Performance for Networks Larger Than 800 Nodes



1: $r_i / d_i = 3$, $D = 5.19 \log_2 N$, $R = 4.4N^{.33}$
2: $r_i / d_i = 3$, $D = 2.07 \log_2 N$, $R = 3.7N^{.38}$
3: $r_i / d_i = 4$, $D = 5.19 \log_2 N$, $R = 3.4N^{.42}$
4: $r_i / d_i = 4$, $D = 2.07 \log_2 N$, $R = 2.9N^{.48}$
5: $r_i / d_i = 7$, $D = 5.19 \log_2 N$, $R = 2.9N^{.54}$
6: $r_i / d_i = 7$, $D = 2.07 \log_2 N$, $R = 2.3N^{.62}$

Second, overall increase in path length depends on the traffic matrix. If nearly all traffic in the network is between nearby neighbors, then path lengths will be better.

Finally, paths depend somewhat on the physical topology chosen. For instance, if a network exhibits the properties of a topological hierarchy, such as an internet does with gateways and subnetworks, then by making gateways higher level Landmarks, it may be possible to improve path lengths for more distant neighbors.

Clearly, this is an area for further study.

7.4 Comparison with the Area Hierarchy

We are interested in comparing our simulation results with some obtained for the area hierarchy. We are aware of two large-network simulations of area hierarchies, Callon's and Hagouel's. Unfortunately, Callon didn't calculate routing table sizes, and so we may not fully compare our results with his. His results do, however, show some interesting trends in the area hierarchy. We will discuss this after we present Hagouel's results.

7.4.1 Comparison of Routing Table Size and Path Length

Hagouel's simulations of the area hierarchy (Hagouel, 1983), results in show $R = 46$ and $\hat{P} = 1.18$. These results are typical for one of his better performing schemes. The data for this calculation comes from Hagouel's Figure 5.1. It is based on 200 node networks, with average node degrees of 2.8. Hagouel does not tell us his diameters, but based on his network generation scheme, which would allow any node to be connected to any other node with equal probability, his diameters would be about the same for the loop-span model with longest possible maximum span—about 12 hops.

As a comparison, we generated a 200 node network with average node degree 2.8 and diameter 12, and calculated R and $\hat{P}$ over all hierarchy types. We obtained $R = 25.75$, and $\hat{P} = 1.22$. For an individual simulation with $R = 45.08$, we had $\hat{P} = 1.10$. For an individual simulation with $\hat{P} = 1.17$, we had $R = 30.63$. Clearly, at least for a 200 node network, we see better performance with the Landmark Hierarchy.

7.4.2 Comparison of Effect of Diameter and Node Degree

Kamoun's study of the area hierarchy (Kamoun, Kleinrock, 1977) is highly theoretical and idealized. No simulation is done. His results show that routing table sizes of $R = HN^{\frac{1}{H}}$ are possible (his Equation 6, where his notation is $\bar{l}$ for routing table size, and m for number of hierarchy levels). The conditions necessary for this area hierarchy equation are that all areas be composed of identical numbers of sub-areas. The fact that Hagouel's simulations give results of approximately 3 times worse than theoretical best brings the practical validity of the theoritical results under suspicion. (For Landmark Routing, we get $R = kN^m$, where m is on the order of 1/2 and k is on the order of 3.)

It is not known how easy it is to achieve this condition. It appears, however, that it may be easier to achieve this condition in networks with larger node degrees and smaller diameters. Hagouel shows that the number of nodes in an area goes up as the number of border nodes (nodes on the edge of the area) goes down. The number of border nodes present will go up as node degree goes up and diameter goes down (compare a fully connected network with a loop network). Therefore, we will see small areas, and hence a larger number of hierarchy levels and smaller routing tables for networks with smaller diameters. This is the opposite of what we see for the Landmark Hierarchy.

Kamoun also gives an equation for an upper bound on the increase in path length for the area hierarchy with the above condition. Among other things, his results show that relative path lengths get worse as 1) the connectivity goes up (and likewise as diameter goes down), and 2) as the number of hierarchical levels goes up. These are the same result we have observed for the Landmark Hierarchy.

These findings on path length are confirmed by Callon. In his simulations of 200 node networks, he varied the node degree by 4, 6, and 10 (since he is studying packet radio networks, his node degrees are higher), and got diameters of roughly 32, 18, and 12 respectively. (We am estimating on these diameters. Callon gave average shortest path length, not diameter. However, our simulations show that diameters are consistently nearly twice that of the average shortest path length.) For his pure hierarchical scheme (Figure 2-4c in his paper), he got $\hat{P} = 1.02$, $\hat{P} = 1.07$ and $\hat{P} = 1.11$ respectively. Clearly, path lengths are worse for smaller diameters. In fact, even Callon's absolute path length increases ($P^{area} - P^{sh}$) are getting worse for smaller diameters, which is worse that what we saw for the Landmark Hierarchy.

What we see here, then, is that for the Landmark hierarchy, smaller diameters mean larger table sizes and longer path lengths, whereas for the area hierarchy, smaller diameters mean smaller table sizes and longer path lengths. On one hand, this seems to imply better performance from the area hierarchy for small diameter networks. On the other hand, nobody has simulated the area hierarchy for very small diameter networks, so we are not sure what path lengths would be. Both Hagouel's and Callon's simulations were for networks which were out of the very small diameter range. Therefore, we have no hard numbers to verify this statement.

Another very important consideration is that, for a hierarchical topology the area hierarchy performs very well. For a perfect topological hierarchy, the area hierarchy will provide the appropriate reduction in routing table size, but no penalty will be paid in path length. Of course, hierarchical topologies are less survivable than general topologies.

We have yet to determine the performance of the Landmark hierarchy on a topology with hierarchical characteristics, such as an internet. Since most networks exhibit some characteristics of a topological hierarchy, this is an important area for further research. However, we believe that by forcing nodes higher up in the topological hierarchy to be higher-level Landmarks, we can get efficient path lengths using a Landmark Hierarchy.

In addition, the area hierarchy is based on the same structure as administrative hierarchies (ANSI X3S3.3, 1987), which are common in data networking. In this sense, area hierarchies are convenient for use in data networks, and are expected to perform well. Again, the use of the Landmark Hierarchy in administrative hierarchies needs to be studied. However, we believe that by 1) putting administrative hierarchy as well as Landmark Hierarchy information in addresses, and 2) by allowing nodes on the edges of administrative areas to filter out certain Landmark Hierarchy updates, we may achieve good performance in administrative hierarchies.

8.0 CONCLUSION

This closing section gives a summary of the Landmark Hierarchy, which can replace Sections 4 through 7 for readers not requiring analytical details. The summary is followed by conclusions emphasizing applicability to the DDN and DoD Internet. The section ends with a discussion of future work.

8.1 Summary

Due to the rapid growth of data networks, it is necessary to impose hierarchical routing structures on data networks in order to contain the vast amounts of routing information present in the networks. Until now, the only hierarchy known was the area hierarchy. It has seen extensive research over the past decade. The internet, for example, is a type of area hierarchy in the routing sense.

One of the problems with the area hierarchy is that it is very difficult to manage dynamically. That is, it is difficult to adjust the area hierarchy to topological changes in a distributed, dynamic fashion, thereby subjective, the area hierarchy to such failures as the area partition. This problem is manageable on a limited basis, but not at the level which a DoD network might require.

To deal with this problem, we introduce a new hierarchy, the Landmark Hierarchy, which we believe is much easier to dynamically manage. This paper presents an analysis of the Landmark Hierarchy in its static state—that is, we study its performance as a static element, but do not study how to dynamically manage it. That will be the topic of forthcoming work.

We analyze the Landmark Hierarchy in terms of three parameters: routing table sizes, path lengths, and path distribution. The size of routing tables tells us how much routing overhead the hierarchy produces. In general, this is much less than that seen with no hierarchy. Longer paths are the penalty we pay for using a hierarchy. Path distribution tells us whether the hierarchy skews traffic patterns, thus causing unfairness and undue stress in parts of the network. Our analysis of routing table sizes is quite thorough. Unfortunately, we were able to analytically say very little about path lengths and path distribution.

The major portion of our results comes from simulations of the Landmark Hierarchy. In all, we present results of over 1000 Landmark Hierarchy simulations, plus other simulations analyzing various network aspects. The simulations were run on nearly 50 different network types, using 23 different hierarchy descriptions and several variations on these. To support this work, we developed a new model for describing networks, the loop-span model, which allows us to generate quasi-random networks with control over the number of nodes, the node degree, and the diameter. Previous techniques only allow control over the first two parameters.

Our major results are as follows:

1. One can affect routing table sizes and path lengths through adjustments in the hierarchy parameters, namely, the density of Landmarks, and the distance Landmarks can be seen (Figures 14 and 15). This gives the designer considerable flexibility in optimizing the performance of the Landmark Hierarchy for any given network. We also believe that the dynamic Landmark Hierarchy management algorithms may be made to automatically adjust the Landmark Hierarchy parameters for optimal performance in response to changing network conditions.

2. When adjusting the Landmark Hierarchy parameters, routing table sizes can only be made smaller at the expense of longer path lengths, and vice versa (Figures 14 and 15).

3. Routing table sizes and path lengths are strongly affected by network parameters, namely the number of nodes, the node degree, and the network diameter (Table 5). In particular, networks with very small diameters exhibit both larger routing tables and longer path lengths. In many cases, the Landmark Hierarchy (and the area hierarchy, for that matter) may perform worse than non-hierarchical routing for these cases.

4. Random assignment of Landmarks performs nearly as well as a uniform assignment of Landmarks. This shows that the Landmark Hierarchy is extremely resilient to the placement of Landmarks—an important survivability consideration.

5. There is a large variance between the size of the largest and smallest routing tables in a network—the largest are about 6 times larger than the smallest.

6. Most routing table entries come from the lower hierarchy levels.

7. Path lengths are the same as shortest path at hierarchy level 0, and get longer at the higher levels. Overall path length increase depends heavily on the traffic matrix, and is significantly better when most traffic is between nodes which are close to each other.

8. We were not able to determine what the effect on path distribution is. Our preliminary results show that the Landmark Hierarchy does not cause undue unfairness in path distribution. Several characteristics which we did not study, namely routing metrics (static and real-time) and the network topology, will tend to have a smoothing effect on path distribution.

9. The number of global Landmarks (those at the highest level of the hierarchy which all nodes can see) impacts routing table size and path length. In general, we were able to get significant improvement in these two parameters by increasing the number of global Landmarks from one to ten.

10. Our results show that, as a rule of thumb, routing table sizes of $R = 3\sqrt{N}$ are typical, where N is the number of nodes. These results are shown in Figures 14, 16, and 17. This can be compared to the area hierarchy where $R = HN^{\frac{1}{H}}$. However, this figure for the area hierarchy is conditional on each area having an identical number of subareas, a condition not achievable in practice. Further, this figure is not supported by simulation. In comparable simulations, the Landmark Hierarchy showed smaller routing table sizes than the area hierarchy.

11. We were not able to obtain general figures for the increase in path length. However, we found that path lengths in the Landmark Hierarchy behave similarly to those in the area hierarchy. Our path length simulation results are shown in Figure 15. Also, our simulation results show better path lengths that those shown in similar simulations for the area hierarchy.

12. For networks with very small diameters (close to the smallest possible for a given number of nodes and node degree) the Landmark Hierarchy performs very poorly, and cannot be recommended over the area hierarchy or no hierarchy. However, networks with diameters are not normally found in practice (the author knows of no such networks). It is worth noting that path lengths on the area hierarchy are also poor for these small diameter networks.

8.2 Conclusions

The major conclusion from this paper is that, in its static state, the Landmark Hierarchy is a viable alternative to the area hierarchy. In its static state, the area hierarchy performs better, because its routing table sizes are not subject to change due to the network parameters as the Landmark Hierarchy is. The results from this paper alone, however, are not sufficient proof that one should use a Landmark Hierarchy. For a network where survivability is not a major concern, and where the topology is normally static, we recommend use of the area hierarchy.

To determine whether the Landmark Hierarchy should be used in networks where either survivability is a major concern, or where the topology changes quickly and often, further work concerning the dynamic management of the Landmark Hierarchy is required. However, from the results of this study, we may recommend that work on the Landmark Hierarchy continue.

With regards to the DDN, the use of the Landmark Hierarchy may result in overall savings in routing overhead as compared with non-hierarchical routing, depending on the network situation. Recent DDN studies of the area hierarchy (Khanna, Seeger, 1986; Sparta, 1986), assume a DDN of 1000 nodes (Packet Switched Nodes, or PSNs, in this case). From Sparta, we see that, given the DDN routing discipline, Shortest Path First (SPF), this amounts to 6133 bits/second, or 11% of a 56,000 bit/second link. Using the data from Figures 14 and 15 for the small diameter experiment :1a,h1, and extending the curves to 1000 nodes, we get $R = 130$ and $\hat{P} = 1.05$. Assuming 80% loading of links, we get 2240 bits/second from the 5% increase in path length, and roughly 790 bits/second from the updates due to the 130 routing table entries. Now, assuming a flooding discipline for propagating address changes, and assuming that at times of stress, we see 10 address changes per second (which should be worst case), we get an additional 2000 bits/second, for a total of 5030 bps. This is only a roughly 20% savings over flat SPF, hardly worth the effort. If we assume 1.54 Mbps links, then flat SPF is by far the better scheme. If we assume that at times of stress, many DDN links will be limited to 9600 bps, then we see only 3174 bps (roughly 1/2 of flat SPF) from the Landmark Hierarchy, 2000 bps of which is due to the address changes. If we assume a more efficient address updating scheme (which we believe is possible (Stine, Tsuchiya, 1987)), and low bandwidth links, then the Landmark Hierarchy begins to look more attractive. However, we must first prove that the dynamic management of the Landmark Hierarchy is workable.

The Landmark Hierarchy looks much more attractive for the DoD Internet. Here we see several thousands of nodes, including both PSNs and Internet Protocol (IP) gateways, and 56,000 bps and sometimes slower links. Clearly, a non-hierarchical routing scheme is out of the question. Using Curve 4 from Figure 17 (larger diameter experiment) and assuming say 12,000 nodes and updates every minute, then we get a routing table size about 150 nodes. We will assume path lengths of 10% over shortest path. (We believe that this assumption is realistic, based on our simulations and on optimalities which may be achieved by taking advantage of the Internet topology.) Now we see roughly 460 bps from the routing updates, and 4480 bps from the increased traffic. Assuming a good address binding scheme which produces a negligible amount of traffic, we get roughly 5000 bps, or about 11% of the 80% loaded 56000 bps link. This is acceptable performance.

Of course all of these figures assume certain diameters of the DDN and the DoD Internet. A map of the ARPANET with 39 PSNs showed a diameter of 12 hops. This is larger than the diameter we used in our DDN calculations. We don't know the diameter of the Internet (where the Internet is defined as that collection of connected hosts and gateways which use the DoD Internet Protocol). Based on the example path in Section 7, we believe that diameters larger than that shown in our large diameter experiment are and will be typical.

8.3 Future Work

We have just scratched the surface of all the work that needs to be done on the Landmark Hierarchy. First, there is the study of the algorithms needed to accomplish both routing and dynamic management of the Landmark Hierarchy. This is the most important work to do now. It is of a different nature than the work done in this paper—dynamic rather than static. In a companion paper (Tsuchiya, 1987), we discuss the architectures and issues associated with the dynamics of the Landmark Hierarchy—what we globally call Landmark Routing.

However, there is also much work of a static nature still to be done at some point in time. This work has been mentioned throughout this paper, and is summarized here.

1. First, we believe that more analysis and simulation on the Landmark Hierarchy is necessary. This is crucial for determining path lengths. Ultimately, we hope for a situation where one may take a set of equations or algorithms, put in certain network parameters such as number of nodes, diameter, node degree, link bandwidths, a traffic matrix, and get out performance numbers.

2. We believe more work could be done on improving the performance of the Landmark Hierarchy. In Section 6.10.2 we show that substantial improvement is possible by adjusting the number of nodes at the highest hierarchy level. However, we didn't have time to explore this improvement at length. Other improvements may also be possible.

3. We need to consider different types of networks. This includes networks with different types (delay and bandwidth) of links; networks with a regular topology, especially a hierarchical topology; and networks with extensive broadcast media, such as satellite and packet radio networks.

4. We need to do survivability studies on the Landmark Hierarchy. In particular, we need to determine how easy it is to partition the Landmark Hierarchy for given Hierarchy Parameters.

5. We need to consider the structure and performance of the Landmark Hierarchy in networks with administrative boundaries.

6. We need to consider the use of locally but not globally unique Landmark IDs. For various reasons not mentioned here, locally unique Landmark IDs are a necessity in a dynamic network (all networks are dynamic).

APPENDIX A

Glossary of Mathematical Terms

N	The total number of nodes in a network.
D	The diameter of a network (the maximum distance between any two nodes).
C	The average node degree of a network, where the node degree of a single node is the number of nodes it is connected to.
L	The number of links in a network.
H	The highest Landmark Level (0 is the lowest).
i	The Landmark Hierarchy level.
LM_i	A Landmark of level i.
$LM_i\,[id\,]$	An LM_i with ID $= id$.
$V_i\,[id\,]$	The set of nodes which lie within the Landmark Vicinity of an $LM_i\,[id\,]$.
$r_i\,[id\,]$	The radius of a Landmark Vicinity for a particular $LM_i\,[id\,]$.
r_i	The average radius of a group of LM_i. When this appears in the text, the set of $r_i\,[id\,]$ which are being averaged is understood by context. If a single network or simulation is the context, r_i is averaged over all Landmarks $LM_i\,[id\,]$ in the network. When an experiment (a group of simulations) is the context, r_i is averaged over all Landmarks for all simulations in the experiment.
$d_i\,(id_x \rightarrow id_y\,)$	The distance from a particular node with ID id_x to either, 1) $LM_i\,[id_y\,]$ if id_y is specified, or to the closest LM_i if it is not.
d_i	The average distance from each node to its closest LM_i.
$d_{i\ \mathrm{max}}$	The maximum distance a node may be from an LM_i.
$v\,(x\,)$	The number of nodes within x hops of a node.
T_i	The total number of LM_i in a network.

R The average total number of entries in node's routing tables ($R[id]$ for a specific node).

R_i The average total number of LM_i in a node's routing table ($R = \sum_{i=0}^{H} R_i$).

P^{sh} The length of the shortest path between two nodes.

P_i^{lm} The average length of a path between two nodes when a level i Landmark was the highest level used for routing.

$\hat{P}_i$ The average increase in path length over shortest path for all P_i^{lm} (P hat sub i~ =~ {sum from {node~ pairs} {P sub i sup lm (node~ pair)} over {P sup sh (node~ pair)}} over node~ pairs .

$\hat{P}$ The average $\hat{P}_i$ for all levels i.

F_i^{per} An indication of the number of paths which transit Landmarks F_i at a given level i. F_i^{per} is the average of the percentile rankings (ranging from 0 to 1) of each LM_i compared to all nodes. This value is used because F_i varies so much from node to node that an absolute value is less meaningful that a ranking.

F_i^{rat} Another indication of the number of paths which transit Landmarks F_i. This is the ratio of F for the LM_i over the average F over all Landmarks.

$e_i[id]$ The number of hops a Landmark Vicinity Extension for an $LM_i[id]$ extends around all LM_{i+1} within $r_i[id]$ hops of $LM_i[id]$.

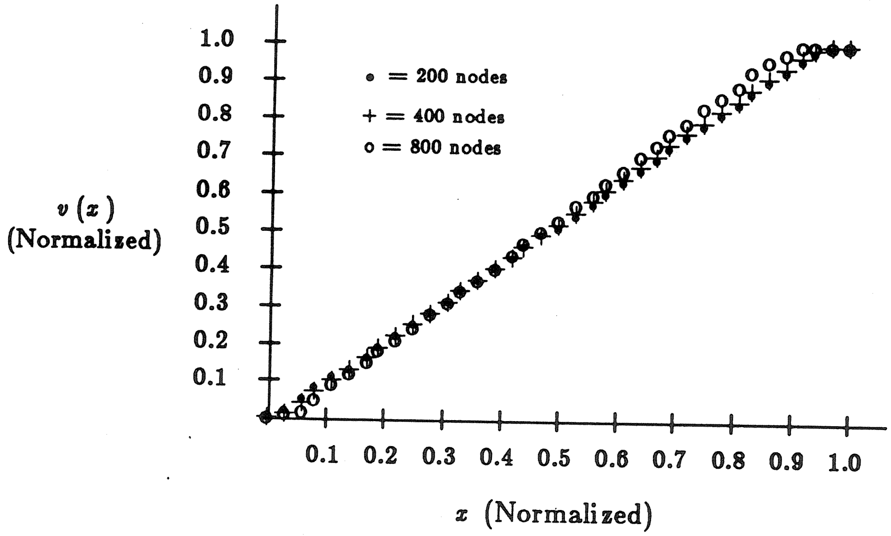
o_i The distance in hops that a Landmark Vicinity will extend past the nearest LM_{i+1}.

c_i The number of LM_{i+1} that a Landmark Vicinity should encompass.

APPENDIX B

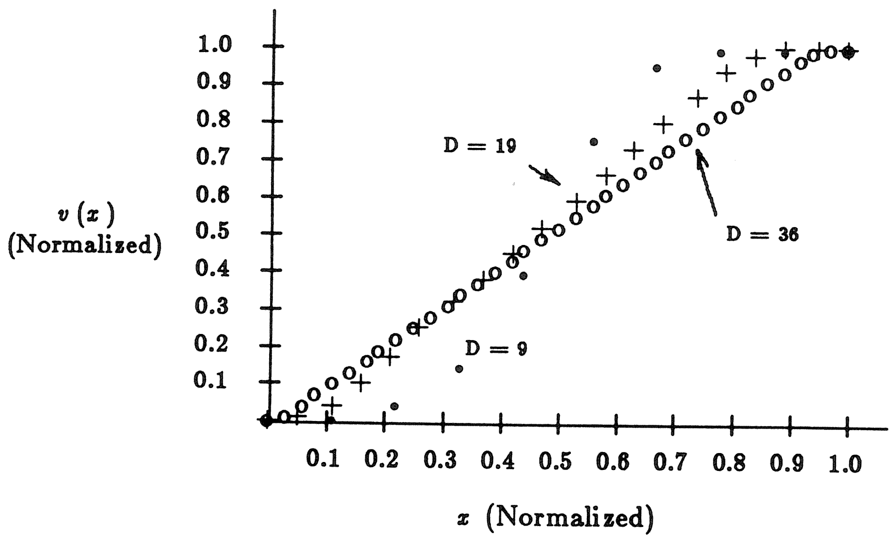
Landmark Vicinity Size vs. Landmark Radii

Figure B-1
Effect of Number of Nodes on Landmark Vicinity Size: Graph 1



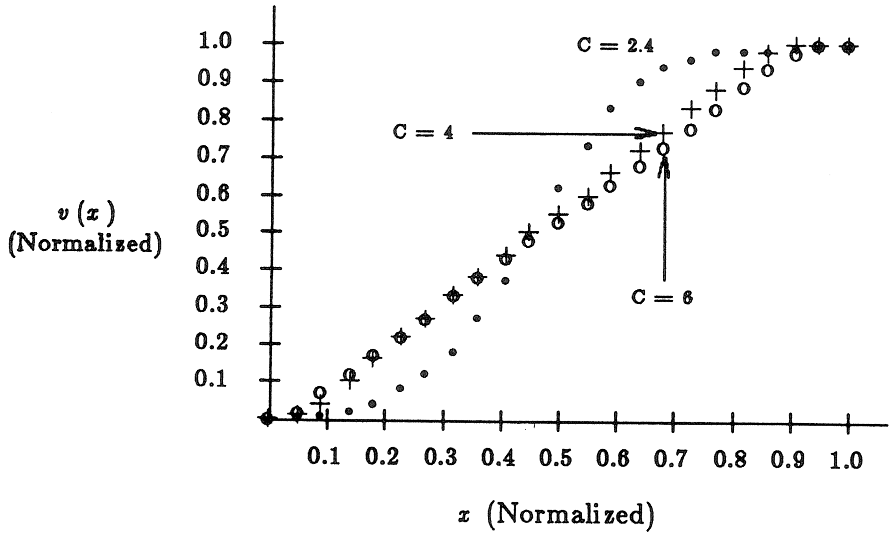
$D = 36$
$C = 4$
$x^{\max} = 36$
$v(x)^{\max}$: .=200 +=400 o=800

Figure B-2
Effect of Network Diameter on Landmark Vicinity Size: Graph 1



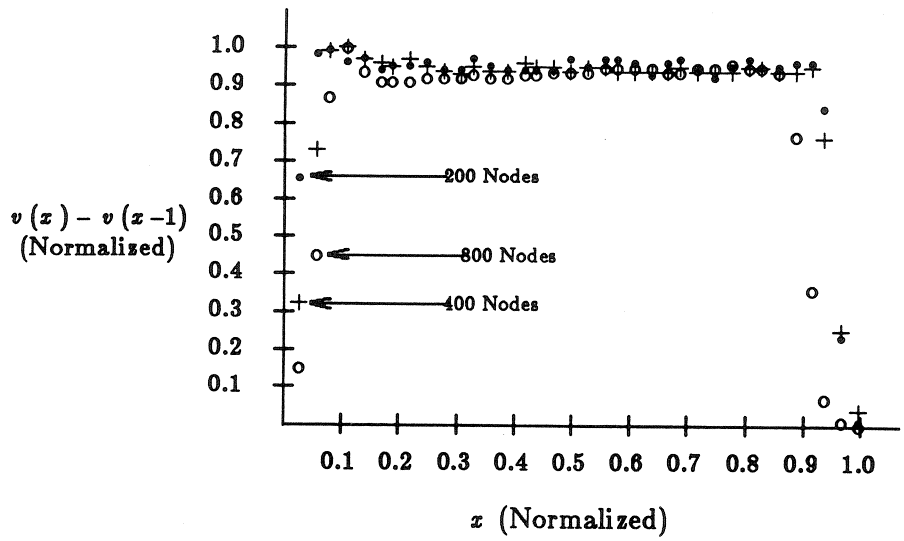
$N = 400$
$C = 4$
x^{max}: .=9 +=19 o=36
$v(x)^{max} = 400$

Figure B-3
Effect of Node Degree on of Landmark Vicinity Size: Graph 1



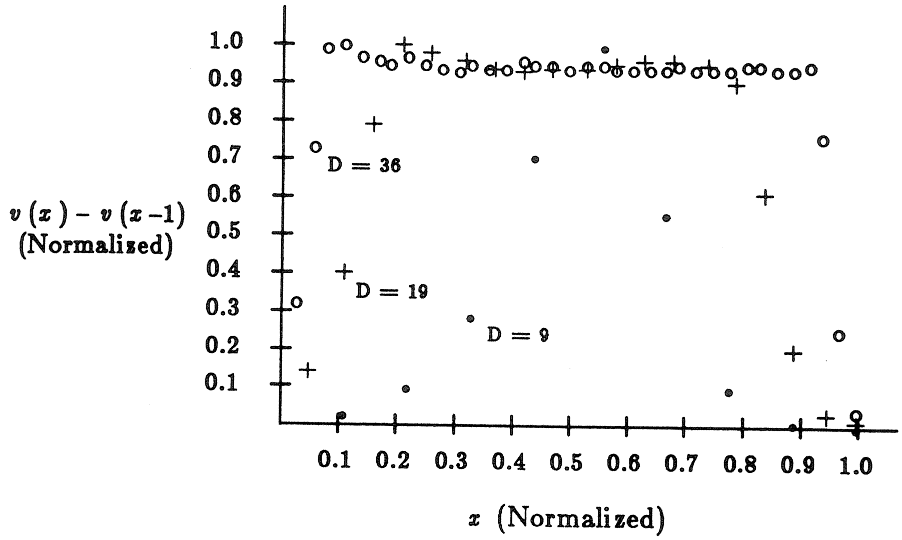
$N = 400$
$D = 22$
$x^{max} = 22$
$v(x)^{max} = 400$

Figure B-4
Effect of Number of Nodes on Landmark Vicinity Size: Graph 2



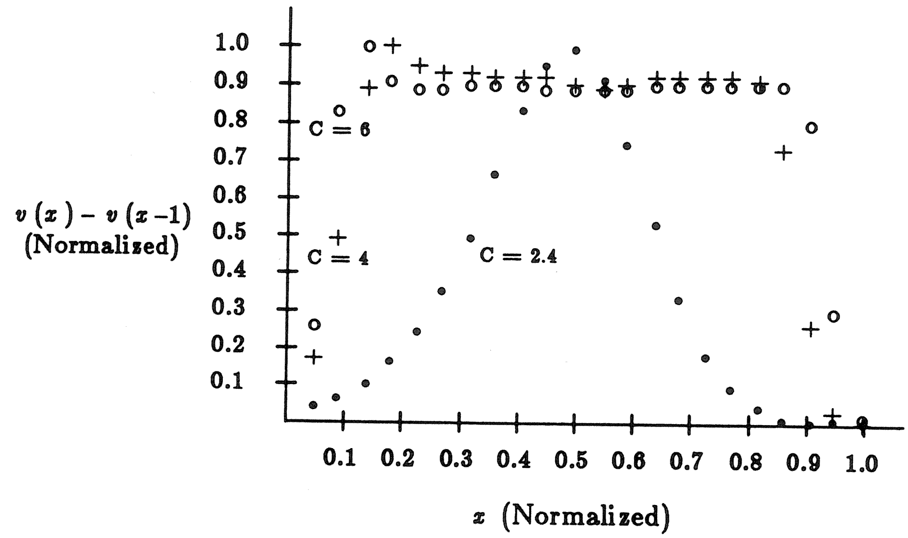
$D = 36$
$C = 4$
$x^{max} = 36$
$v(x)^{max}$: .=6.10 +=12.64 o=27.59

Figure B-5
Effect of Network Diameter on Landmark Vicinity Size: Graph 2



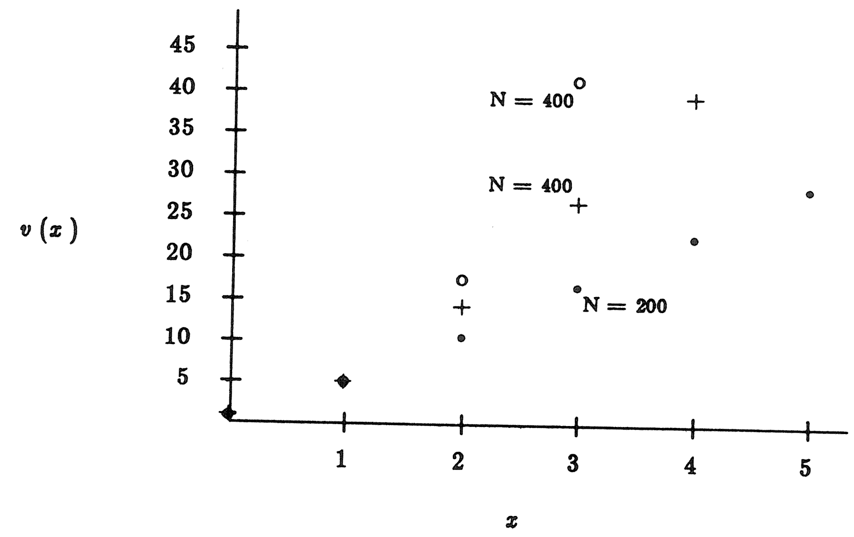
$N = 400$
$C = 4$
x^{max}: .=9 +=19 o=36
$v(x) - v(x-1)^{max}$: .=142.66 +=29.35 o=12.64

Figure B-6
Effect of Node Degree on Landmark Vicinity Size: Graph 2



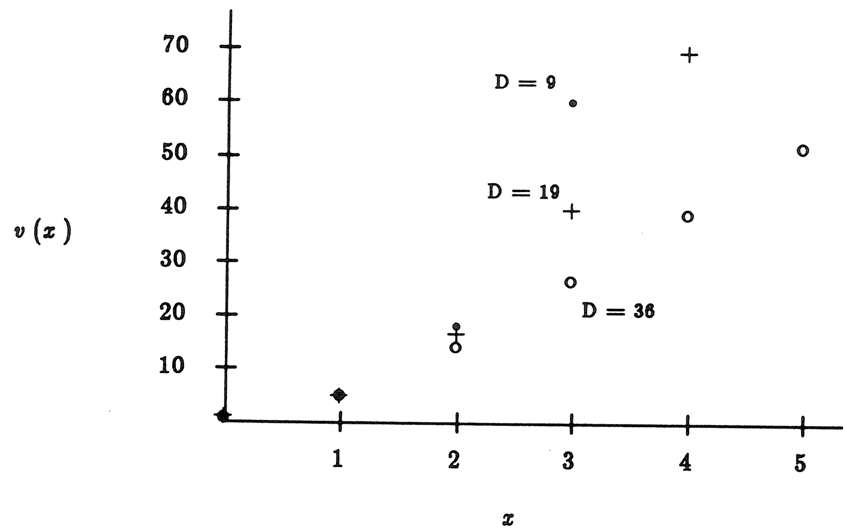
$N = 400$
$D = 22$
$x^{max} = 22$
$v(x) - v(x-1)^{max}$: .=50.21 +=24.32 o=22.73

Figure B-7
Effect of Number of Nodes on Landmark Vicinity Size: Graph 3



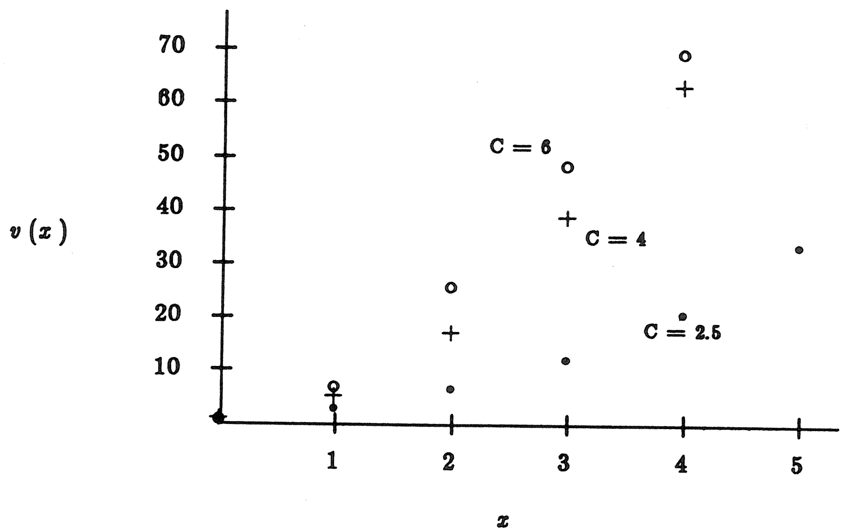
$D = 36$
$C = 4$

Figure B-8
Effect of Network Diameter on Landmark Vicinity Size: Graph 3



$N = 400$
$C = 4$

Figure B-9
Effect of Node Degree on Landmark Vicinity Size: Graph 3



$N = 400$
$D = 22$

APPENDIX C

Loop-Span Method of Generating Networks

C.1.0 INTRODUCTION

This Appendix describes the technique we used to generate networks for simulation. This technique automatically generates connected (meaning there is a path from every node to every other node) general-topology (meaning there is no regular structure to the topology) networks. Our method accepts as input the number of nodes desired N, the number of links L or equivalently the average node degree C, and either the desired diameter D, or a parameter we call maximum span length s^{max}, which affects the diameter.

C.1.1 Existing Techniques for Generating Connected, General-topology Networks

Most network generation schemes come from the fields of Random Graphs [1] and Percolation Theory (Pike, 1974). These schemes are based on two models of random graphs. In the first model, a number of nodes N (up to infinity), and the probability that any two nodes share a link L are specified. In the second model, nodes are placed in some m-dimensional space with a certain probability, and any two nodes that have a certain spatial relationship (i.e., they are within a certain distance of each other, or their positions are aligned a certain way) are said to share a link. Neither of these schemes ensures connectivity; instead, they evaluate the connectivity of the networks generated.

We found three examples of network generation schemes using the first model with the constraint that the resulting network be connected (Callon, Lauer 1985; Sparta 1986; Hagouel, 1983). In all of these cases, the number of nodes and links could be controlled, but not the diameter. Callon and Sparta ensured connectivity by selectively adding links after the network was generated if it was not connected. Hagouel generated the network giving all nodes had at least one link, thus decreasing the probability of a disconnected network. If the resulting network was still disconnected, he threw it out.

It appears that one could use the second model to control the diameter of a network by changing its dimensionality m. Here, a larger dimensionality would result in a smaller diameter. (This is hinted at in Seager and makes sense intuitively.) We did not pursue this method of controlling diameter because 1) working in m-dimensional space is complex, and 2) it requires checks for connectivity, which increases both the computational expense of the network generation scheme and the time it takes to generate the connected network.

Toueg and Steiglitz present a network generation technique where the object is to create a network with a certain number of nodes and links and the smallest diameter possible. Their approach is to take a sample connected network, and to try changing pairs of links, each time checking for a smaller diameter. We believe a similar approach could be used to generate networks of any given diameter—that is, changing pairs of links, but each time checking for a diameter closer to the desired one. However, this approach is also computationally expensive.

C.2.0 THE LOOP-SPAN TECHNIQUE FOR GENERATING RANDOM GRAPHS

This section first describes the loop-span model, and then describes our network generation scheme and presents the results of our scheme in generating networks.

C.2.1 The Loop-Span Network Model

A loop-span network consists of a Hamilton circuit of N nodes and links, and $L-N$ additional links (called spans). This model is derived from the chordal ring network model (Arden, Lee, 1981; Doty, 1984). The difference is that the chordal ring network model deterministically places the spans in strategic places to achieve the smallest possible network diameter.

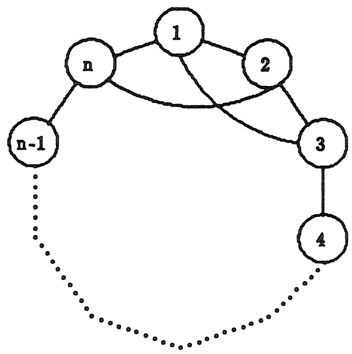
Figure C1 shows examples of a network drawn as a loop-span. The value of a span s_L is the number of loop links which the span bridges. For example, the spans shown in Figure C1a are of length $s_L = 2$, and those shown in Figure C1b are of length $s_L = \frac{N}{2}\rfloor$, where $\rfloor$ is the integer floor function. The range of values for s_L is $2 \leq s_L \leq \frac{N}{2}\rfloor$. A span may also be expressed in normalized fashion s_N, where $s_n = \frac{2s_L}{N}$. Here, the range of values is $0 < s_n \leq 1$. The longest span in a loop-span network is denoted by s_L^{max} (or equivalently, s_n^{max}).

We further specify a loop-span network by stating that the spans must be uniformly distributed among the set of possible spans ranging from $2 \leq s_L \leq s_L^{max}$. Finally, we state that spans may be randomly placed in a loop with the constraint that only one link may join a pair of nodes. A loop-span network, then, is specified with three parameters: the number of nodes N, the number of links L, and the maximum span s_n^{max}.

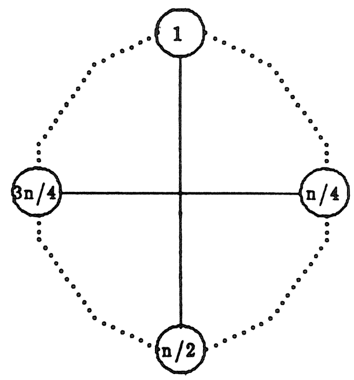
In the following paragraphs, we discuss several aspects of the loop-span network model.

First, we observe that the length of the spans in the loop-span model impacts the network diameter. To illustrate this, consider a loop-span network with N nodes and $L = N+2$ links. If the two remaining spans are added as shown in Figure C1, the diameter is $D = \frac{N-1}{2}\rfloor$. However, if the two spans are added as shown in Figure C2, the diameter is much smaller, approximately $D \simeq \frac{N}{4}$. Second, we argue that, from a

Figure C-1
Example Networks Showing Varying Diameters



1a: Large Diameter



1b: Small Diameter

practical standpoint, a loop-span model is a valid representation of most real, large networks. For any real network to be perfectly modeled using the loop-span model, it must of course contain a Hamilton circuit, which only a small percentage of real networks have [4]. However, for the real networks which we drew using the loop-span model, we found that we could come close to finding a Hamilton circuit. Usually we couldn't find one because either 1) there were instances of the topology shown in Figure C2a where both nodes couldn't be picked up by the Hamilton circuit, or 2) there were instances of the topology shown in Figure C2b, where a stub node couldn't be picked up. However, the subtopology of Figure C2a will not affect the diameter of a network. The subtopology of Figure C2b will not usually affect the diameter of the network, because the network diameter is realized only for a small percentage of node pairs, and the stub node will with high probability not be one of those nodes.

There are, however, three types of networks that the loop-span model does not represent well.

1. Non-connected networks.

2. Networks that do not form a loop, but instead have a long, skinny topology.

3. Networks with a large percentage of Non-Hamilton subtopologies.

The first type does not apply here since non-connected networks have infinite diameters. The second type could be handled with an open-loop-span model, where one of the links on the loop was missing. However, the spans in that model would need to be small, since a long span could close the loop, thus approximating the loop-span model. In this appendix, we do not consider the open-loop-span model, because most large real

Figure C-2
Two Non-Hamilton Subtopologies

2a: Double Loop Subtopology 2b: Stub Subtopology

networks exhibit a rich enough connectivity that an open-loop is not needed. The third type could be easily handled by adding numerous instances of those subtopologies to an otherwise completed loop-span network. This appendix, however, does not explore the third type. These types may interest someone who wishes to study the loop-span model in more detail.

C.2.1.1 Generating the Spans

Given the maximum span length $s_L^{\max}$, we calculate the average number of spans of any given length is $g^{av} = \frac{L - N}{s_L^{\max} - 1}$. Here, $L - N$ is the total number of spans, and $s_L^{\max}-1$ is the number of different span lengths from 2 to $s_L^{\max}$. Since g^{av} will usually not be an integer, and since s_L must be an integer, we use the following method for determining the number of spans for each span length.

Let the number of spans of length s_L be denoted as g_{s_L}. Let $g_2 = g^{av} \rfloor$; then add the remainder of the floor function for g_2, denoted g_2^{rem}, to g^{av}, and let $g_3 = (g^{av} + g_2^{rem})\rfloor$, $g_4 = (g^{av} + g_3^{rem})\rfloor$, and so on. As an example, if $N=200$, $L=400$, and $s_L^{\max}=29$; then $g^{av}=7.143$, $g_2 = 7.143\rfloor = 7$, $g_3 = (7.143+0.143)\rfloor = 7.286\rfloor = 7$, $g_4 = (7.143+0.286)\rfloor = 7.429\rfloor = 7$, and so on.

C.2.1.2 Generating the Network

When specifying a network to be generated, we give the number of nodes, either the average node degree or the number of links, and either the desired diameter or $s^{\max}$.

When the desired diameter is specified, values of $s^{\max}$ must be chosen by the software and networks generated until a network with the desired diameter results. Our algorithm for doing this was to first pick the largest and the smallest possible $s^{\max}$ ($s_n^{\max} = 1$ and $s_n^{\max} = 0$, which defaults to $s_L^{\max} = 2$). From these two choices, two diameters are generated. A third $s_n^{\max}$ is then chosen based on an interpolation of the two diameters and the desired diameter. For instance, if the resulting diameters were 10 and 30, and the desired diameter was 15 (1/4 of the way between 10 and 30), we would then try $s_n^{\max} = .25$. This third try would result in a new diameter. Depending on whether this new diameter was higher or lower than the desired diameter, it would replace the old higher or lower chosen diameter, representing a better guess. Then another interpolation would take place, and so on until the desired diameter resulted.

The algorithm for generating a random network given N, L or C, and s^{max} is simple. First create the loop. Then, for each span, randomly pick a node and place the span at that node. If the node already has a span of that length, incrementally pick successive nodes on the loop until one is found without a span of that length, and put the span there. One may place other constraints on the placement of the span if desired. For instance, it may be desirable to restrict the total number of links any node may have. Since the loop ensures connectivity, there is no need to check connectivity and patch the network, as is required with other network generation schemes.

C.2.2 Results of the Experiments

In our experiments, we generated a total of 525 networks. We generated 5 networks each of all possible permutations of networks with 1) N=200, N=400, and N=800 nodes; 2) node degrees of C=2.4, C=3, C=4, C=5, and C=6; and 3) maximum spans of s_n^{max}=0.02, s_n^{max}=0.03, s_n^{max}=0.05, s_n^{max}=0.1, s_n^{max}=0.2, s_n^{max}=0.5, and s_n^{max}=1. For each network we calculated the network diameter. Then, for each group of five networks, we calculated the mean μ_D and standard deviation σ_D of the diameters, where

$$\mu_D = \frac{\sum_{tries} D}{tries} \qquad \sigma_D = \sqrt{\frac{\sum_{tries} (\mu_D - D)^2}{tries}}.$$

This group of 5 is one experiment.

We could isolate any parameter by taking the mean of μ_D and σ_D over all the experiments that held that parameter. For instance, if we wished to isolate the effect of s_n^{max}=.02, we take the means μ_μ and μ_σ for all experiments where s_n^{max}=.02. Except where specified, all results are for the random span distribution.

Table 1 gives the results of interest, isolating the node degree C, the number of nodes N, and the maximum span length s_n^{max}.

We see immediately that the diameter is affected by both the maximum span length and by the node degrees, as previously explained. In our experiments, network diameter was changed by more than a factor of 7 by adjusting s^{max}. In the extreme, diameter can be varied from 1 to $\frac{N}{2}\rfloor$ by varying C (from 2 to N), but for the realistic cases that we tried (C=2.4 is a very thinly connected network, while C=6 is a very richly connected network), the diameter varied only by approximately a factor of 3 by varying C. We have shown here that s^{max} is a strong factor in determining network diameter. This is the main finding of this appendix.

The last two entries of Table C1 represent extreme cases of diameter, where both s^{max} and C are, or they are both high. This gives an idea of the full range of diameters which can result from changing both of these parameters.

Table C-1
Loop-Span Experiment Results

parameter(s)	μ_μ	μ_σ	μ_μ / μ_σ (×100)
s_n^{max}=.02	86.8	1.40	1.61%
s_n^{max}=.03	67.57	1.80	2.66%
s_n^{max}=.05	47.68	1.59	3.33%
s_n^{max}=.10	28.61	1.65	5.77%
s_n^{max}=.20	17.77	0.95	5.35%
s_n^{max}=.50	12.35	0.64	5.18%
s_n^{max}=1.0	11.28	0.50	4.43%
C=2.4	72.10	3.05	4.23%
C=3	43.43	1.47	3.38%
C=4	30.06	0.76	2.53%
C=5	25.21	0.44	1.75%
C=6	23.23	0.38	1.64%
N=200	30.84	0.85	2.76%
N=400	39.71	1.24	3.12%
N=800	46.05	1.57	3.41%
s_n^{max}=.02, c=2.4	148.53	2.17	1.46%
s_n^{max}=1.0, c=6	6.07	0.13	2.14%

We note that from Table C1, the number of nodes appears to have less of an impact on network diameter. This is somewhat misleading, because since both C and s_n^{max} are effected by N, we are not really showing the total effect of varying N. For a given number of links L, and a given s_L^{max} where s^{max} here is expressed in terms of hops, changing N will indeed affect diameter.

It is interesting to consider the standard deviations shown in Table C1. First, we note that they are small. This means that we can fairly well predict which diameter will result from a given s^{max}; consequently, not many network generation trials are be required to produce an exact diameter. On the average, 13.1 tries were required to produce the desired diameter, with a standard deviation of 93 (quite high). Second, we see that $\frac{\mu_\mu}{\mu_\sigma}$ varies 1) as s^{max} and 2) as the inverse of C. The quotient $\frac{\mu_\mu}{\mu_\sigma}$ gives a measure of standard deviation in proportion to the network diameter. In other words, the more spans there are, and the shorter those spans are, the less variation there is in diameter from network to network. The standard deviation decreasing as the number of spans makes sense in that the more links there are, the less impact each will have (thus resulting in a smoothing effect on the diameter). The standard deviation increasing as the length of the spans increases makes sense in that longer spans have more impact on network diameter than short spans.

C.3.0 FUTURE WORK

This appendix presents several intriguing avenues for future work. First, several variations on the loop-span model presented in this appendix are possible. One link of the loop may be removed, resulting in the open-loop-span previously mentioned. The distribution of spans may be something other than uniform, including the case where the spans are all the same length. Also, span lengths could be randomly chosen rather than deterministically chosen. Second, as a new way to model networks, the loop-span model offers possible advances in all the many areas of study where networks are used to model problems: Random Graphs, Graph Theory, Percolation Theory, and Operations Research (to name a few). To explore these possibilities, a more rigorous, mathematical study of the loop-span model is called for.

C.4.0 SUMMARY

This appendix considers approaches to the problem of generating connected general-topology networks with control over the network diameter. We briefly review existing techniques, and then present an original technique, based on the loop-span model (a derivative of the chordal ring), which allows for more control over network diameter while reducing the computational expense of generating the network. We present a detailed description of the loop-span network model, and show (through network generation experiments) how the loop-span model may be used to control network diameter.

REFERENCES

American National Standards Institute ANSI X3S3.3, 3.3/86-215 (April 1987), *Draft Routing Architecture.*

Arden, B. and Lee, H. (April 1981), *Analysis of Cordal Ring Network*, IEEE Transactions on Computers, Vol C-30, pp. 291-295.

Bollabas, B. (1985), *Random Graphs*, Academic Press.

Callon, R. and Lauer, G. (June 1985), *Hierarchical Routing for Packet Radio Networks,* Report No. 5945, SRNTN No. 31, Cambridge, MA: BBN Laboratories Incorporated.

Doty, K. W. (May 1984), *New Designs for Dense Processor Interconnection Networks,* IEEE Transactions on Computers, Vol. C-33, No. 5.

Garcia-Molina, H. (January 1982), *Elections in a Distributed Computing System,* Vo. C-31, No. 1, pp. 48-49, IEEE Transactions on Computers.

Hagouel, J. (1983), *Issues in Routing for Large and Dynamic Networks,* RC 9942 (#44055), Yorktown Heights, NY: IBM Thomas J. Watson Research Center.

Harary, F. (October 1972), *Graph Theory,* Addison-Wesley Publishing Company, Third printing.

Kamoun, F. and Kleinrock, L. (1977), "Hierarchical Routing for Large Networks: Performance Evaluation and Optimization," *Computer Networks*, Vol. 1, pp. 155-174.

Kamoun, F. and Kleinrock, L. (November 1979), "Stochastic Performance Evaluation of Hierarchical Routing for Large Networks," *Computer Networks*, Vol. 3, No. 5, pp. 387-353.

Kamoun, F. and Kleinrock, L. (1980), "Optimal Clustering Structures for Hierarchical Topological Design of Large Computer Networks," *Computer Networks*, Vol. 10, No. 3, pp. 221-248.

Khanna, A. and Seeger, J. (January 1986), *Large Network Routing Study Design Document*, Report No. 6119, Cambridge, MA: BBN Communications Corporation.

McQuillan, I. Richer, I. and Rosen, E.C. (May 1980), *The New Routing Algorithm for the ARPANET,* Vol. COM-28, No. 5, IEEE Transactions on Communications.

Perlman, R. (1985), "Hierarchical Networks and the Subnetwork Partition Problem," *Computer Networks and ISDN Systems*, Vol. 9, pp. 297-303.

Pike, G.E. and Seager, C.H. (August 1974), *Percolation and Conductivity: A Computer*

Study, Physical Review B, Vol. 10, No. 4, pp. 1421-1434.

Saltzer, J., Reed, D., and Clark, D. (August 1980), "Source Routing for Campus-wide Internet Transport," *Proceedings of the IFIP WG 6.4 Workshop on Local Networks.*

Shacham, N. (November 1985), *Hierarchical Routing in Large, Dynamic Ground Radio Networks,* Menlo Park, CA: SRI International.

Sparta Incorporated, (April 1986), *Design and Analysis for Area Routing in Large Networks,* McLean, VA.

Stine, B. and Tsuchiya, P. (March 1987), *Assured Destination Binding: A Technique For Dynamic Address Binding*, MTR-87W00050, McLean, VA: The MITRE Corporation.

Sunshine, C. (April 1981), "Addressing Problems in Multi-network Systems," *Internet Engineering Note (IEN) 178.*

Toueg, S. and Stieglitz, K. (July 1979), *The Design of Small-Diameter Networks by Local Search,* IEEE Transactions on Computers, Vol. c-28, No. 7.

Tsuchiya, P. (September 1987), *Landmark Routing: Architecture, Algorithms, and Issues*, MTR-87W00174, McLean, VA: The MITRE Corporation.

Zakon, S. (October 1985), "An Architecture for Routing in the ISO Connectionless Internet," *Computer Communication Review*, Vol. 15, No. 5, pp. 10-39.

DISTRIBUTION LIST

MITRE Washington

A-10 G. MacDonald
A. Tachmindji

D-14 J. Babcock
E. Brady
J. Quilty

W-30 G. Benke
R. Britton
R. Grayson
L. Keane
R. Shirey
L. Wentz
D. Woodall
M. Zobrak

W-31 S. Bhanji
H. Duffield
D. Gomberg
R. Hansen
W. Lazear
T. Louden
S. McLeod-Reisig
M. Meltzer
A. Messeh
R. Miller
P. Tsuchiya (30)
A. Whitaker
R. Wilmer
D. Wood (2)
Associate Tech. Staff
Technical Staff

W-34 T. Minton
E. Schmidt
A. Schoka

W-35 C. Bowen
D. Jurenko
W. Kinzinger
W. Stewart

W-36 E. Boyle
B. Brooks
G. Lipsey
D. Zugby

W-37 R. Haller
R. Pesci

W-75 W. Blankertz

MITRE Washington Library

Record Resources (2)

MITRE Bedford

A-10 C. Zraket

D-110 G. Koehr
I. Richer

D-111 J. Woodward

D-114 H. Bayard

D-118 S. Ames

MITRE Bedford Library

External

SPONSORS

DCA/DCSO

Mr. W. Harding, Code B101
COL T. Herrick, Code B600
Mr. E. Schonborn, Code B601
Mr. L. Tabacchi, Code B602
LtCol G. Mundy, Code B602A
Mr. W. Grindle, Code B610
Capt M. St. John, Code B612
LTC A. Maughan, Code B620

DISTRIBUTION LIST (Concluded)

LtCol C. Holland, Code B630
LTC D. Schreiner, Code B640
Mr. J. Milton, Code B650

OSD/C^3I

Mr. M. Corrigan

DARPA

MAJ M. Pullen

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION (U)	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY N/A	3. DISTRIBUTION / AVAILABILITY OF REPORT
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE N/A	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) MTR-87W00152	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION The MITRE Corporation — 6b. OFFICE SYMBOL (If applicable) W31	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS (City, State, and ZIP Code) 7525 Colshire Drive McLean, VA 22102	7b. ADDRESS (City, State, and ZIP Code)
8a. NAME OF FUNDING / SPONSORING ORGANIZATION DCA — 8b. OFFICE SYMBOL (If applicable) B600	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code) 8th and South Courthouse Road Arlington, VA 22204-2199	10. SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO. — ; PROJECT NO. 359A; TASK NO. — ; WORK UNIT ACCESSION NO. —

11. TITLE (Include Security Classification)
The Landmark Hierarchy: Description and Analysis

12. PERSONAL AUTHOR(S)
Paul Tsuchiya

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Final	FROM ______ TO ______	1987-06-30	115

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Hierarchical routing structures are needed to reduce the amount of routing information stored and exchanged by switching nodes in large networks (data or voice), Only one hierarchical structure, the area hierarchy, has been available to network designers. This has resulted in a limited set of design alternatives. In particular, the area hierarchy is known to have some poor survivability characteristics. This paper introduces a new hierarchical structure, the Landmark Hierarchy. Analysis and simulation of the Landmark hierarchy in its static state show that is a viable alternative to the area hierarchy for large network routing. Further work is needed to determine the survivability characteristics of the Landmark Hierarchy in a dynamic network environment.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION
22a. NAME OF RESPONSIBLE INDIVIDUAL LTC A. Maughan, Code B620	22b. TELEPHONE (Include Area Code) (703) 285-5101 — 22c. OFFICE SYMBOL B620

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED