THE LANDMARK HIERARCHY: A NEW HIERARCHY FOR ROUTING IN VERY LARGE NETWORKS

Paul F. Tsuchiya
tsuchiya@gateway.mitre.org

The MITRE Corporation

Abstract

Landmark Routing is a set of algorithms for routing in communications networks of arbitrary size. Landmark Routing is based on a new type of hierarchy, the Landmark Hierarchy. The Landmark Hierarchy exhibits path lengths and routing table sizes similar to those found in the traditional area or cluster hierarchy. The Landmark Hierarchy, however, is easier to dynamically configure using a distributed algorithm. It can therefore be used as the basis for algorithms that dynamically configure the hierarchy on the fly, thus allowing for very large, dynamic networks. This paper describes the Landmark Hierarchy, analyzes it, and compares it with the area hierarchy.

1. INTRODUCTION

This paper describes a new hierarchy that can be dynamically configured in communications networks. This hierarchy is the basis of a new set of routing algorithms, collectively called Landmark Routing, for routing in arbitrarily large networks. Landmark Routing has the following features:

1. It operates efficiently and automatically in networks of arbitrarily large size.
2. It responds to changing network conditions such as topology changes.
3. It provides full name-based addressing, thus accommodating any percentage of mobile nodes.
4. It provides automatic address assignment, thus easing network administration.
5. It accommodates administrative boundaries, providing control of routing paths, protection, and autonomy.

The linchpin of Landmark Routing is a new routing hierarchy called the Landmark Hierarchy. The purpose of any hierarchy is to optimize routing in very large networks, and the Landmark Hierarchy accomplishes this. The Landmark Hierarchy, however, can be more easily configured using a distributed algorithm (Tsuchiya, 1987b). As such, it responds automatically to changes in network topology. The only previously known hierarchy, the area hierarchy, is not easily configured using a distributed algorithm (Hagouel, 1983), (Shacham, 1985).

1.1. Contents

This paper presents a description of the Landmark Hierarchy. It also summarizes the results of an analysis of the Landmark Hierarchy, and compares those results with analogous work done for the area hierarchy. Finally, we briefly mention the other aspects of Landmark Routing—the underlying routing protocol, the name-to-address binding scheme, and the administrative boundaries. Three reports (Stine, Tsuchiya, 1987), (Tsuchiya, 1987a), (Tsuchiya, 1987b) contain a full description of the Landmark Hierarchy and the dynamic algorithms used with it.

In this paper, we use the word router to describe a switching element in a network, like a packet switch or a gateway. We use the word link to describe the communications medium that connects two routers. Finally, we use the word host to describe a source or sink of traffic in a communications network. A host does not participate in Landmark Routing, whereas all routers do.

2. HIERARCHY DESCRIPTION

In this section, we first describe the area hierarchy for background. We then describe the Landmark Hierarchy, followed by a performance comparison of both.

2.1. The Area Hierarchy

Figure 1 shows a 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 routers into areas, grouping areas into super-areas, and so on.

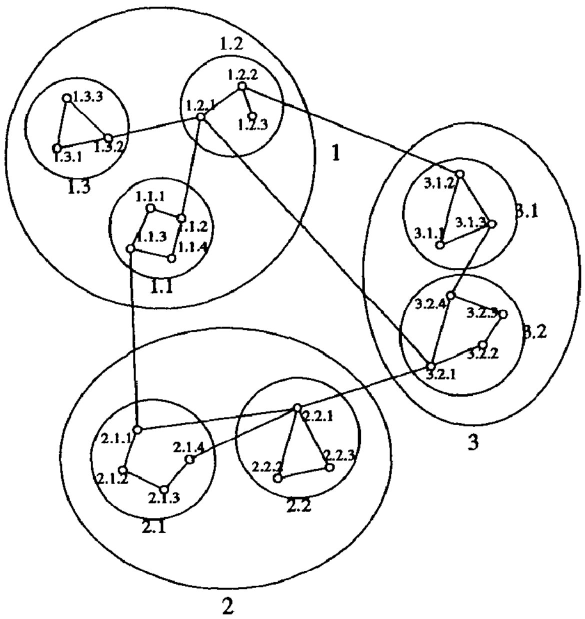

Figure 1: Area Hierarchy Example

For the sake of discussion, we will consider a single router in Figure 1 to be a Level 0 area, a group of routers 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 level 0 area. 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 level 0 area (i.e. 3.2.1).

A constraint on the definition of an area is: a path that 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 routers outside the area to view the area as a single entity. The result is that only one entry is required in that router's routing table to route to several routers in another area. For instance, in Figure 1, Router 2.1.1 views Routers 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, router 2.1.1 views all routers 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 router in Area 3 forward the message to Area 1. Having no knowledge about the internal topologies of Areas 3 and 1, however, 2.2.1 is forced to 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.

There are many possible variations on this basic structure. First, areas may overlap. This variation has been studied by Shacham (Shacham, 1985). Second, it is not necessary for every router to have routing entries for every area at every level. For instance, non-border routers may only have entries for border routers, which in turn have additional information about other areas. This has been studied by Hagouel (Hagouel, 1983). Other studies of the area hierarchy include (Kleinrock, Kamoun 1977), (Kleinrock, Kamoun, 1979), (Kleinrock, Kamoun, 1980), (Sunshine, 1981), (Westcott, Lauer, 1984), (Perlman, 1985), (Callon, Lauer, 1985), (Khanna, Seeger, 1986), (Sparta, 1986).

2.2. 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. Third, we describe how routers are addressed in a Landmark Hierarchy. Finally, we show how routing may take place with the Landmark Hierarchy.

2.2.1. The Landmark

The description of a Landmark is very simple. A Landmark is a router whose neighbor routers within a certain number of hops contain routing entries for that router. As an example, consider router 1 in the network of Figure 2. Routers 2 through 6 have routing entries for router 1 (as indicated by the arrowheads) and are therefore able to forward any packets addressed for router 1 to router 1. Routers 7 through 11 do not contain routing entries for router 1. Therefore, router 1 is a Landmark which can be "seen" by all routers within a distance of 2 hops. We refer to router 1 as a Landmark of radius 2. In general, a router for which all routers within r hops contain a routing entry is a Landmark of radius r. Note that Router 1 does not necessarily have routing entries for routers 2-6. (This is to be distinguished from an area in an area hierarchy, where *all* routers in an area have routing entries for *all others*.)

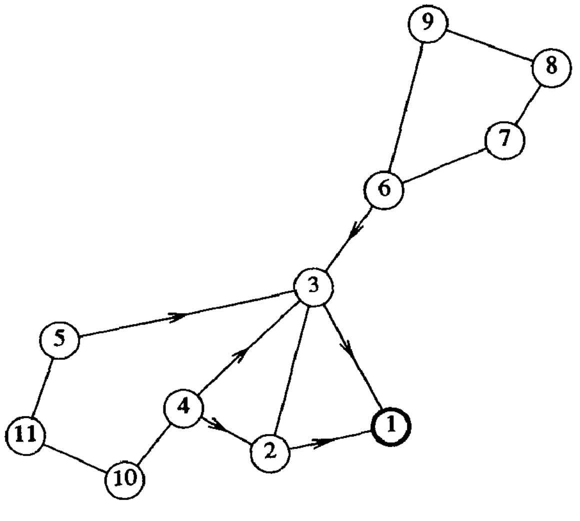

Figure 2: A Single Landmark

2.2.2. Hierarchy of Landmarks

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

Each $LM_i[id]$ has a corresponding radius $r_i[id]$. In the Landmark Hierarchy, every router 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 $r_1[id]$, 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 small set of routers are $LM_H[id]$'s each with an $r_H[id]$, with $r_H[id] \geq D$, D being the diameter of the network. Because the radius of these Landmarks is larger than the diameter of the network, all routers in the network can see these Landmarks. We call these global Landmarks.

Figure 3 illustrates the Landmark Hierarchy by showing a portion of a network. This is a two-dimensional representation, meaning that only routers drawn physically close to each other would share a link. For simplicity, only four of the routers are shown, and no links are shown. The dotted arrows and circle indicate the radius of the Landmarks; that is, the region within which routers contain routing entries for that Landmark. For instance, every router within the circle defined by $r_1[b]$ has an entry for, and can route to, $LM_1[b]$.

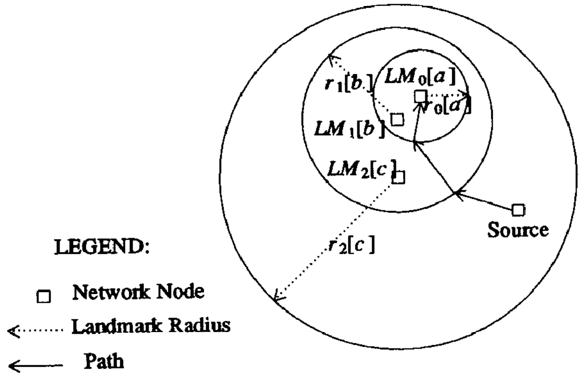

Figure 3: Landmark Hierarchy

2.2.3. Routing Table

Each router in the network keeps a table of the next hop on the shortest path to each Landmark for which it has routing entries. Each router 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 router is an LM_0, and since every router has entries for every $LM_0[id]$ within a radius of $r_0[id]$, every router has full knowledge of all the network routers within the immediate vicinity. Likewise, since a portion of all LM_0 are LM_1, every router will have knowledge of a portion of the network routers further away. Similarly, each router will have knowledge of even fewer routers further still, and so on. The end result is that all routers have full local information, and increasingly less information further away in all directions. This can be contrasted with the area hierarchy where a router 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.

2.2.4. Addressing in a Landmark Hierarchy

In an area hierarchy, the address of a router is a reflection of the area(s) at each hierarchical level in which the router resides. The telephone number is a well-known example of this. In a Landmark Hierarchy, the address of a router is a reflection of the Landmark(s) at each hierarchical level which the router is near. The Landmark Address, then, is a series of Landmark IDs: $LM_{i_H}[id_i].LM_{i-1}[id_{i-1}]. \ . \ . \ . \ .LM_0[id_0]$.

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 router labeled $LM_0[a]$ in Figure 3 may have the Landmark Address $LM_2[c].LM_1[b].LM_0[a]$. In this case, we call $LM_2[c]$ a parent of $LM_1[b]$, and we call $LM_1[b]$ a child of $LM_2[c]$. In this paper, the terms parent and child will always refer to two Landmarks, the lower of which is using the higher as part of its address. The address of the router 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 radius of the router labeled $LM_0[a]$. Since more than one Landmark may be within the radius of a lower level Landmark, routers may have a multiplicity of unique addresses. Multiple addresses could be used to improve survivability and provide some traffic splitting. (In the area hierarchy, overlapping areas may be employed to the same effect.)

2.2.5. Routing in a Landmark Hierarchy

Now we may consider how routing works in a Landmark Hierarchy. Assume we wish to find a path from the router labeled *Source* to the router labeled $LM_0[a]$ in Figure 3. The Landmark Addresses for the router labeled $LM_0[a]$ is $LM_2[c].LM_1[b].LM_0[a]$. The basic approach is the following: 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 router will make the same decision as Source, and the next, until the path reaches a router which is within the radius of $LM_1[b]$. When this router 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 router will choose a path towards $LM_1[b]$. This continues until a router 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 is more so if the radius for an LM_i goes well beyond an LM_{i+1}. 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).

2.2.6. Landmark Hierarchy Example

In order to better illustrate the Landmark Hierarchy, consider Figure 4. This network has 3 hierarchical levels. All routers (small circles) are LM_0. Diamonds denote LM_1 and large circles LM_2. The rightmost address component is the $LM_0[id]$, and is unique for each router 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 $LM_2[id]$, indicating proximity to an LM_2. For this example, all $r_0 = 2$ hops, all $r_1 = 4$ hops, and all $r_2 = 8$ hops. In the more general case, not all r_i are the same.

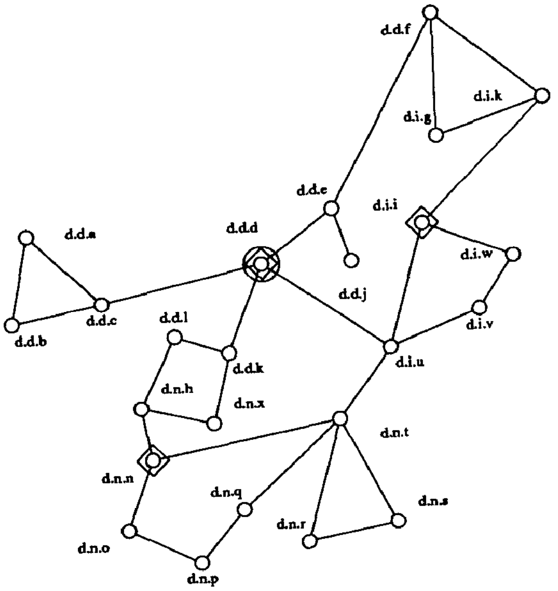

Figure 4: Landmark Routing Example

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

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

Table 1: Routing Table for Router *g* of Figure 4

Let's consider a routing example where Router *g* (*d.i.g*) is routing a message to Router *t* (*d.n.t*). Router *g* examines Router *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. Router *g* does, however, have an entry for $LM_2[d]$, and therefore forwards the message towards $LM_2[d]$ via Router *f*. Router *f* also does not have entries for $LM_0[t]$ or $LM_1[n]$, and therefore forwards the message towards $LM_2[d]$ via Router *e*. Router *e* does have an entry for $LM_1[n]$ (but not $LM_0[t]$), and forwards the message towards $LM_1[n]$ via Router *d*. Router *d* does have an entry for $LM_0[t]$, as does Router *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. DYNAMIC ALGORITHMS IN LANDMARK ROUTING

Due to space limitations, we do not discuss at length the dynamic algorithms used with the Landmark Hierarchy. For completeness, however, we briefly mention them here. They are described more fully elsewhere (Tsuchiya, 1987b).

Of course, an algorithm is required to assign Landmarks and determine their corresponding radii. We call this the hierarchy management algorithm. Simply stated, this algorithm builds a hierarchy of Landmarks from the bottom up. The hierarchy is built such that each Landmark has 3 or 4 children at steady state, with 1 and 7 being the minimum and maximum number of children respectively.

Each router is at least an LM_0 upon birth, chooses an r_0, and advertises itself that distance through the routing algorithm. As such, each LM_0 hears of peer LM_0's, and possibly higher level Landmarks. If it hears an LM_1, it can graft itself onto the hierarchy under that LM_1, and shrink its radius accordingly. If it doesn't, it can run an election with peers to build up the hierarchy.

The second algorithm needed in Landmark Routing is the routing algorithm, which is used to discover Landmarks as well as to establish paths to Landmarks It must be of the distance-vector type (also known as old-ARPANET, Bellman-Ford, or Tajibnapis' algorithm). Link-state type algorithms (also known as ARPANET, SPF, or Dijkstra's algorithm), will not work because they require a full topology map (or an abstraction thereof, such as is provided by the area hierarchy), which is not available using a Landmark Hierarchy.

Distance-vector routing algorithms have been extensively studied and are still being improved upon (Cegrell, 1975; Tajibnapis, 1977; McQuillan, Richer, Rosen, 1978; Jaffe, Moss, 1982; Hagouel, 1983; Garcia-Luna, 1987). The main reason for the proliferation of studies on distance-vector algorithms is the count-to-infinity problem, during which routing loops can occur. We have proposed a solution to this problem, called Alternate-path Distance-vector Routing (ADR), that exhibits extremely fast convergence because alternate paths are discovered before changes in topology occur (Tsuchiya, 1987b). The routing table complexity of ADR for any given router is $O(N'C)$, where N' is the number of destinations in the routing table, and C is the router degree (the number of neighbor routers).

Any distance-vector routing algorithm can be used in the Landmark hierarchy with only minor modification. Simply stated, distance-vector routing works as follows: Every router periodically or on an event driven basis informs its neighbors of its distance to one or more destinations. Upon receiving such information, a router adds to the reported distance its distance to its neighbors.

The router considers its distance to the destination to be the smallest of the received distances. The next hop to the destination is the one over which the shortest distance was received.

The modification needed for the Landmark hierarchy is that, when a router receives an update from a neighbor, it decrements an additional field that indicates how many hops away from the Landmark the update can travel. This field is set to the Landmark radius by the Landmark. If this field is decremented to zero, then the update is not passed on to any neighbors. This is the mechanism for preventing information about a Landmark from traveling beyond its radius.

The third algorithm required is one to bind permanent names or IDs to changing addresses. This is a particular problem in Landmark Routing because any router's address (and therefore any host's address) can change due to adjustments in the Landmark Hierarchy. We have proposed a solution called Assured Destination Binding (ADB) based on Mullender and Vitanyi's hash-binding (Mullander and Vitanyi, 1985). ADB is extremely efficient and robust (Tsuchiya and Stine, 1987; Tsuchiya, 1987b). Its memory requirements are only $O(k_1 \frac{H}{N} + k_2)$, where H is the number of hosts, N is the number of routers, and k_1 and k_2 are small constants (3 and 1 respectively are typical). However, it requires no broadcasts. Updates require on the average k_1 multihop (transits several routers) messages, and queries require on the average k_2 multihop messages.

Finally, there is a means to accommodate administrative concerns. In particular, one is able to define administrative boundaries, and 1) keep ones own traffic within ones boundaries, 2) prevent third-party traffic from crossing ones boundaries if desired, 3) allow ones own routing metrics, and 4) allow correct internal operation in the face of external failures.

Simply put, border routers are hand-configured before start-up to indicate which neighbors are across administrative boundaries. The hierarchy maintenance, routing, and binding algorithms then configure internally before any external configuration occurs. This way, a complete Landmark Hierarchy exists within the administrative boundaries. This internal Landmark Hierarchy is a sub-tree of the global Landmark Hierarchy.

4. STATIC PERFORMANCE

In this section, we compare the static performance of the Landmark Hierarchy and the area hierarchy. We are interested in two parameters, the average routing table size R, and the increase in path length over shortest path, as expressed by the ratio $\hat{P} = \frac{P_{hier}}{P_{sh}}$.

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

The path length also gives a rough indication of the cost in network resources of paths that result from routing in the routing hierarchy. A path length longer than shortest path results in more links and routers being utilized for a given amount of traffic. This causes increased link (to carry traffic), and processor (to process traffic) usage. We expect path lengths in the routing 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 that the cost decrease from smaller routing table size will more than compensate the cost increase from longer path length, resulting in a net improvement in overall network performance.

All of the results given in this paper for the area hierarchy come from Kleinrock (Kleinrock, Kamoun, 1977), and Hagouel (Hagouel, 1983). Most of the results for the Landmark Hierarchy come from Tsuchiya (Tsuchiya, 1987a). The derivations are not given here.

Kleinrock found the average number of routing table entries in the area hierarchy to be at best

$$R_{area} = HN^{\frac{1}{H}}, \qquad 1$$

where H is the number of hierarchical levels, and N is the number of routers in the network. This assumes that every level i area is composed of the same number of level $i-1$ areas for every level of the hierarchy. The analogous result for the Landmark Hierarchy is approximately

$$R_{lm} = 4N^{\frac{1}{3}} \qquad 2$$

Both of these results assume the smallest possible routing tables, and are not achievable practically speaking. They are given here for comparison only. Nevertheless, we see that the area hierarchy, under ideal conditions, can achieve smaller routing table sizes than the Landmark Hierarchy if a fairly large number of levels are used in the area hierarchy. A realistic estimate of routing table sizes for the Landmark Hierarchy is

$$R_{lm} = 3N^{\frac{1}{2}}. \qquad 3$$

Also of interest, however, are simulation results. In his thesis, Hagouel studied several aspects of the area hierarchy. Among these, he studied heuristics for forming areas in an area hierarchy, and he studied alternate routing schemes for routing in an area hierarchy. To support his conclusions, Hagouel simulated twenty 200-router, quasi-random networks. Selected results of these simulations are shown as the "o" data points in Figure 5, and are taken from his Figure 5.1. The labels C1 and C2 refer to his clustering techniques V3.2 and Aggl respectively. The labels R1 and R2 refer to his routing techniques BNEC and CIEC respectively. The various data points connected by each line refer to the maximum number of routers per area used in generating the clusters. The leftmost data points have fewer routers, and the rightmost data points have more routers. The hierarchy simulated was a two-level hierarchy. We do not discuss these techniques in this paper.

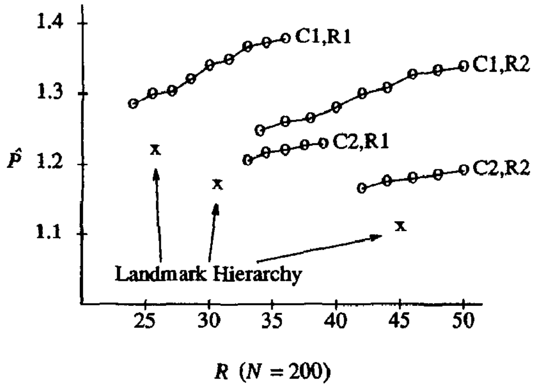

Figure 5: Comparison of Routing Table Size and Path Length for Landmark and Area Hierarchies

The "x" data points are from simulations of the Landmark Hierarchy designed to be directly comparable with that of Hagouel's in that the networks used were also 200 router, quasi-random networks. Note that for similar values of either $\hat{P}$ or R, the Landmark Hierarchy performs better than the best area hierarchy results, although not by enough to get too terribly excited about.

It is interesting to note, however, that for certain clustering parameters, the performance of the area hierarchy can degrade. In Figure 5, we see performance degradation due to the increased maximum number of routers per area. The reason for this can be described in the following example. Assume a network of 1000 routers, and 3 hierarchy levels. If there are 10 elements at each level (10 routers per area, 10 areas per super-area, and 10 super-areas), then each router will have $10+10+10=30$ routing table entries. If, however, there are 40 routers per area, 5 areas per super-area, and 5 super-areas, then each router will have $40+5+5=50$ routing table entries. In addition, the number of hierarchical levels in an area hierarchy can affect performance, as seen in Equation 1.

The analogous situation for the Landmark Hierarchy exists to a much smaller degree than for the area hierarchy. For instance, in three simulations on the same network, hierarchies with 5, 8, and 13 levels had values of R and $\hat{P}$ within 5% of each other. In addition, the performance of the Landmark Hierarchy is very insensitive to the choice of Landmarks. Simulations where Landmarks were chosen at random performed as well as those where a uniform placement of Landmarks was attempted. Because of the insensitivity of Landmark Hierarchy performance to placement of Landmarks and number of hierarchy levels, it is easy to manage the Landmark Hierarchy in a dynamic, distributed fashion.

The obvious question, then, is what does affect the performance of the Landmark Hierarchy? Clearly, in Figure 5 we were able to adjust R and $\hat{P}$ over a wide range of values. How are these adjustments accomplished?

To answer, we must first discuss the parameters that describe the Landmark Hierarchy. We have already discussed the Landmark radius r_i—the distance that a level i Landmark can be seen. Here we introduce another parameter d_i, which is the distance from a router to the nearest level i Landmark. (When written without the subscript, we mean the average over all i. We make the following observations:

1. As r grows larger, the routing table sizes R will also grow larger, and the path lengths P_{lm} will grow smaller. R will grow larger because, as r is increased, more routers will see Landmarks, and will thus have more routing table entries. P_{lm} will grow smaller because paths will key in on lower level Landmarks sooner, thus creating better paths.
2. For a given network, if there are more Landmarks at a given level i, then each router will on the average be closer to a level i Landmark, and d_i will on the average be smaller.
3. Finally, consider that if d_i is smaller, r_{i-1} can correspondingly be smaller. This results from the requirement that the radius of a level $i-1$ Landmark must only be large enough to cover the nearest level i Landmark. If the level i Landmarks is nearer, the radius can be correspondingly smaller.

It is the placement of Landmarks that determines d. Once d is set, then the radius determines the performance of the Landmark Hierarchy. In other words, it is not the absolute values of d and r that determine Landmark Hierarchy performance, but their ratio. This can be seen in Figure 6. Here we see the results of 9 groups of simulations. Each group was performed on 36 networks ranging from 200 to 800 routers, router degrees from 2.4 to 6, and diameters from 7 to 50 hops. The difference between each group was the value of r/d. These ratios range from 2.6 to 6.4.

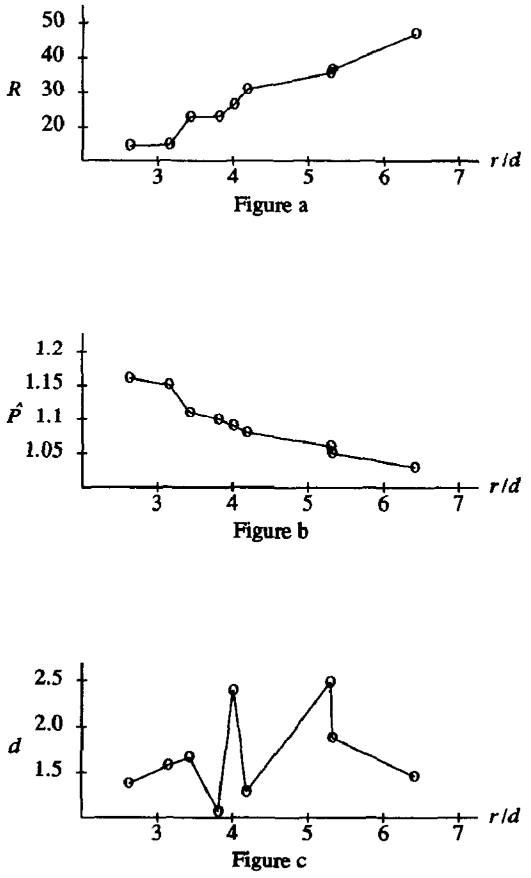

Figure a

Figure b

Figure c

Figure 6: Effect of r/d on Landmark Hierarchy Performance

In Figure 6a, we see the strong relationship between R and r/d. In Figure 6b, we see the strong inverse relationship between $\hat{P}$ and r/d. In Figure 6c, we see that the ratio r/d has only a small relationship with the value of d. This shows that it is the ratio r/d that dominates Landmark Hierarchy performance, not actual values of r or d.

This is important because, in a dynamic environment, the radius of the Landmark is easy to change. The decision to change it comes from the Landmark itself, and the mechanism for doing so is to change a field in the Landmark Update that dictates how many hops outward the Landmark Update travels.

Finally, we should mention that the Landmark Hierarchy does not perform equally well across all types of networks. In particular, it performs worse for networks with very small diameters compared to the number of routers. However, the range of network diameters for which the Landmark Hierarchy performs poorly are smaller than those seen in real networks, even those such as the ARPANET, which are designed to have small diameters.

5. SUMMARY

A new hierarchy for routing in communications networks of arbitrary size has been described. This hierarchy, called the Landmark Hierarchy, exhibits path lengths and routing table sizes similar to those found in the traditional area hierarchy.

The Landmark Hierarchy, however, is easier to configure than the area hierarchy. In particular, the performance of the Landmark Hierarchy (path length and routing table size) is relatively insensitive to the placement of Landmarks, whereas area hierarchy performance is sensitive to the grouping of areas. In addition, it is easier to configure manipulate Landmarks (single routers) than it is areas or clusters (groups of routers). As a result, the Landmark Hierarchy is easier to dynamically configure, and can therefore be made more robust in dynamically changing networks. At the same time, the Landmark Hierarchy is more sensitive to the failure of single routers, and therefore is more complex to operate in fairly stable environments.

We briefly mention the techniques required for dynamic routing in the Landmark Hierarchy. In particular, we mention the hierarchy management algorithm, the routing algorithm, and name-to-address binding algorithm, and administrative boundaries.

We conclude that the Landmark Hierarchy is a promising alternative routing hierarchy, especially for large networks with rapidly changing topologies.

REFERENCES

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

Cegrell, T. (June 1975), "A Routing Procedure for the Tidas Message-Switching Network," IEEE Trans. on Communications, Vol. COM-23, No. 6, pp. 575-585.

Garcia-Luna-Aceves, J. J. (1987), "A New Minimum-Hop Routing Algorithm," Proceedings IEEE Infocom '87, pp. 170-180.

Hagouel, J. (1983), "Issues in Routing for Large and Dynamic Networks," PhD. Thesis, Columbia University.

Jaffe, J. M. and Moss, F. H. (July 1982), "A Responsive Distributed Routing Algorithm for Computer Networks," IEEE Trans. on Communications, COM-30, No. 7, pp. 1758-1762.

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

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

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

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

McQuillan, J. M., Richer, I., Rosen, E. C. (April 1978) "ARPANET Routing Algorithm Improvements First Semiannual Technical Report," Bolt Beranek and Newman Inc., Report No. 3803.

Mullander, S. J. and Vitanyi, P. M. B. (August 1985), "Distributed Matchmaking for Processes in Computer Networks," Proceedings Fourth Conference on Principles of Distributed Computing, Minaki, Ontario.

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

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: Sparta Incorporated.

Stine, R. H. Jr. and Tsuchiya, P. F. (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.

Tajibnapis, W. D. (July 1977), "A Correctness Proof of a Topology Information Maintenance Protocol for Distributed Computer Networks," Communications of the ACM, Vol. 20, No. 7, pp.477-485.

Tsuchiya, P. F. (June 1987a), "The Landmark Hierarchy: Description and Analysis," MTR-87W00152, McLean, VA: The MITRE Corporation.

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

Westcott J. and Lauer, G. (1984), "Hierarchical Routing for Very Large Networks," Cambridge, MA: Bolt Beranek and Newman Incorporated.