Understanding the Influence of Network Topology and Network-layer Naming on the Scalability of Routing

Feng Zhao, Baosheng Wang, Xiaofeng Hu, Wei Peng, and Chunqing Wu
College of Computer, National University of Defense Technology, Changsha, P.R. China
{fengzhao, bswang, xfhu, wpeng, cqwu} @nudt.edu.cn

Abstract—The naming and routing play a very important role for network communications. To characterize the influence of network topology characteristics, network mobility features, network layer naming schemes on the routing scalability, we introduce a new concept called partial transparent graph. We show that a partial transparent graph can better characterize the topological properties of a layered computer network by introducing different kinds of nodes. And we present a routing analysis model to formalize topologies of layered computer networks, network-layer naming, and routing. Then we analyze the routing scalability of two special kinds of naming and routing schemes. We prove that in a static topology with a structured naming, the routing table size can be very small and does not depend on the network size. In a dynamic topology, in which a node can move randomly, the routing table size for name-independent routing with some constraints is linear with the network size.

Index Terms—network architecture, topology, mobility, network naming, name-independent, routing

I. Introduction

Internet has contributed greatly to the development of our society and has changed the way we work and live. Nonetheless, many parts of the current Internet architecture were developed many years ago. It is facing challenges from the network layer such as host mobility, multi-homing and routing scalability. There has been much recent research into network architectures, both evolutionary and 'clean slate'. The America NSF (National Science Foundation) announced the GENI (Global Environments for Network Innovation) program for developing an infrastructure to experiment and test futuristic networking ideas. The NSF effort was followed by the FIRE (Future Internet Research and Experimentation) program under the 7th Framework Program of the European Union, the AKARI (Architecture Design Project for New Generation Network) program in Japan and several other similar programs in other parts of the world. Recently, the NSF announced awards for four new projects ((NDN, Mobility First, NEBULA, and XIA), as part of the Future Internet Architecture (FIA) program, encouraging the network science research community to design and experiment with new comprehensive network architectures and networking concepts that can meet the challenges and opportunities of the 21st century.

Naming and routing are very important core elements, whether for evolutionary network architectures or clean slate network architectures. In the traditional Internet architecture, the use of IP addresses to name interfaces and to be keys for routing decisions is seen as the source of many ills, including the inability to properly incorporate mobility, multi-homing, and so on [1], [2]. Also the current Internet routing is marred with many problems. The biggest and most immediate concern is the routing scalability [3]. With the huge growth in network-able devices participating in the Internet, the routing nodes may be unable to cope with the growth in routing table sizes, number of update messages and churn due to dynamic nature of networks. In future Internet, what naming schemes and routing schemes should be taken to solve one or more of these problems? It has being a hot research topic.

There are basically two questions related with naming and routing that must be considered: What objects need to be named? What information should be used for routing nodes to make forwarding decisions?

Although novel naming approaches, such as naming data or content [4], [5], naming services [6], naming sessions, were proposed, we are in favor that the network layer naming (i.e. naming nodes) should still be necessary to effect communications. Naming network nodes is not a new idea. As early as 1982, Saltzer [7] outlined that a network should have node addresses, which are mapped to the point of attachment addresses. Mapping Saltzer's concepts to the Internet shows that node addresses are missing, and the point of attachment is named twice [8]. The developers of the CLNP (Connectionless Network Layer Protocol) and the OSI reference model [9] had understood the necessity of naming the node rather than the interface. Several proposals [10, 11] had been made to name "endpoints." Naming nodes is now widely acknowledged in the network research community [12]-[20].

Manuscript received July 5, 2013; revised October 29, 2013.
This work was supported by the National 863 program under Grant No.2011AA01A103，China NSF under Grant No.61103189, 61070199, 61100223, Hunan Provincial Natural Science Foundation of China under Grant No.11JJ7003, Program for Changjiang Scholars and Innovative Research Team in University under Grant No. IRT1012, and Program for Science and Technology Innovative Research Team in Higher Educational Institutions of Hunan Province ("network technology").
Corresponding author email: fengzhao@nudt.edu.cn.
doi:10.12720/jcm.8.10.672-680

In terms of the relationships between names and topologies, Network layer naming schemes can be categorized into two main kinds:

Topology-independent naming: The names of nodes are independent of network topologies (i.e. each node has a non-topological network-layer name, called an identifier or ID.)

Topology-dependent naming: The names of nodes are not independent of network topologies. They contain location information and are called locators. For example, when an IP address is used to naming a node, it is name-dependent.

In terms of the scope of names, Network layer naming schemes can be also categorized into two main kinds:

Globally unique naming: Each node has one globally unique network layer name.

Globally non-unique naming: A name may be assigned to two or more nodes. For example, after NAT is introduced, different hosts may have a same IP address.

In terms of the relationships between names and routing, Network layer routing schemes can be categorized into two main kinds:

Identifier based routing: The network layer does not contain location information in the packet header; instead, based directly on identifiers, routing nodes make forwarding decisions. In some documents, this kind of routing is called name-independent routing.

Locator based routing: The network layer contains location information in the packet header. Based on the location information, routing nodes make forwarding decisions. In some documents, this kind of routing is called name-dependent routing.

In terms of whether the names can be aggregated, Network layer routing schemes can be also categorized into two main kinds:

Aggregating routing: The routing tables of a node can be stored in a compact manner, by grouping the set of destination names that use the same output port into intervals of consecutive names. Actually, the routing schemes are based longest prefix match.

Non-Aggregating routing: The routing entries of a node can't be aggregated.

In terms of whether the keys used for routing table lookup can be changed, Network layer routing schemes can be also categorized into two main kinds:

Key-fixed routing: In a packet transmission, the keys used by the intermediate node for making forwarding decisions are fixed.

Key-changeable routing: In a packet transmission, different intermediate nodes may use different keys for making forwarding decisions. The key information in the packet header may be changed by intermediate nodes.

Different combinations of the above naming schemes and routing schemes can bring different naming and routing approaches, which imply multiple naming and routing directions. One possible direction is pure identifier based routing, such as ROFL [21], Disco [22]. Another possible direction is pure locator based, key-fixed routing, such as optimal IP routing. The third possible direction is network based ID-locator split which tries to keep the disaggregated IP addresses out of the global routing and the routing steps are divided into two levels: the edge routing based on identifier and the core routing based on global locators, such as LISP [23], SIX/ONE [24], APT [14], ILNP [25], ENCAPS [26], eFIT [27], and so on. The fourth direction is called host based ID-locator split which requires globally aggregately locators to be assigned to every host. The IDs are decoupled from locators in the end hosts' network stacks and the mapping between IDs and locators is done by a separate distributed system, such as HIP [28], Shim6 [29], MILSA[30], [31] , I3 [32], and so on. There are also several other possible directions, such as locator based, key-changeable routing, and so on. Many proposed network layer naming and routing schemes have their advantages and disadvantages. There is an on-going debate on deciding which way to go among these node naming and routing directions.

To find optimal network layer naming and routing schemes for future Internet, we think it is important to fully understand the essential relationships among network topology, network layer naming and routing. By far now, it is mainly through simulations or experimentations that the performance of many naming and routing schemes are evaluated. We think that giving insightful, theoretical analysis results for naming and routing is very important for future naming and routing design. Also it is a very challenging job.

In this paper we take an initial stab at this challenge. We try to establish a model for characterizing the relationships. In order to give the analytical results for all above naming and routing schemes, lots of research work need to be done. This paper mainly focuses on the analysis about two kinds of naming and routing schemes: *globally unique, topology-dependent naming, key-fixed, locator based, aggregating routing; globally unique, topology-independent naming , key-fixed, identifier based, aggregating routing.* The analysis for other kinds of naming and routing schemes is leaving for future work.

The present paper makes the following contributions:

1) We introduce a novel concept called partial transparent graph. We show that a partial transparent graph can better characterize the topological properties of a layered computer network by introducing different kinds of nodes. We think that the partial transparent graph is beneficial for analyzing layered computer networks.
2) We give formal definitions about partial transparent graph, network-layer naming and aggregating routing, and so on, and establish a basic model for formalizing a layered computer network and analyzing various naming and routing schemes.
3) Based on this model, we analyze the aforementioned two kind of naming and routing schemes. We prove that in a static topology with a topology-dependent naming, the number of local routing table entries can

be very small and does not depend on the network size.

4) And we prove that in a dynamic topology, in which a node can move randomly, the number of local routing table entries for key-fixed, aggregating routing on persistent short identifiers with some constraints is linear with the network size.

The work in this paper is still preliminary. We view this work as the beginning, not the end.

The present paper is organized as follows. We start by introducing the partial transparent graph concept in Section II. We then give a theoretical basic for modeling layered network and analyzing the relationship between topology, naming, and routing in Section III. In Section IV, we give a theoretical bound for the scalability of routing in a network whose topology is static. And in Section V, we give a theoretical bound for the scalability of routing in a network that each node can move randomly. We then discuss our results in Section VI and conclude this paper in Section VII.

II. Overview of Partial Transparent Graph

Conventional graphs are useful for the analysis of routing algorithms, network simulations and so on. However, because conventional graphs do not differentiate different kinds of nodes, they do not work well for the analysis of some kinds of naming schemes and routing schemes in computer networks. So as a complement to the current conventional graph concept, we present a novel concept, called partial transparent graph, to characterize the topological properties of a layered computer network.

There are two main reasons that we present this concept. First, the existing computer network is usually layered. Different nodes of different layers work together. Some nodes can be seen only in certain layers. These nodes are not visible to some other nodes. Different layer topologies may not be the same; however, they may not be independent. The upper layer topology is usually associated with the lower layer topology. Second, more than two nodes in the network layer may be connected by a broadcast link (e.g., Ethernet). Modeling the relationship between these nodes as a full interconnection topology may not be enough.

As shown in Fig. 1 (a), the physical topology consists of three routers, all of which are connected via a broadcast link, and with each router having only one interface. If the physical topology is modeled as a conventional graph (Graph 1) in Fig. 1 (b), then the degree of these nodes representing the routers will be 2. If the physical topology is modeled as Graph 2 in Fig. 1 (c), which introduces an auxiliary node corresponding to the broadcast link, then the degree of these nodes representing the routers will be 1. We think that Graph 2 characterizes the router features better than Graph 1. We call the nodes w0 in Graph 2 a transparent node, because it is invisible to other nodes.

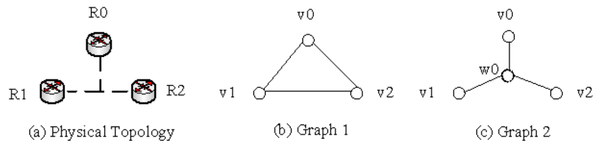


Figure 1. Topology modeling example 1

Aside from a single broadcast being modeled as a transparent node, two or more switches connected together can be modeled as a transparent node, as shown in Fig. 2. We refer to a graph that has one or more transparent nodes as a partial transparent graph.

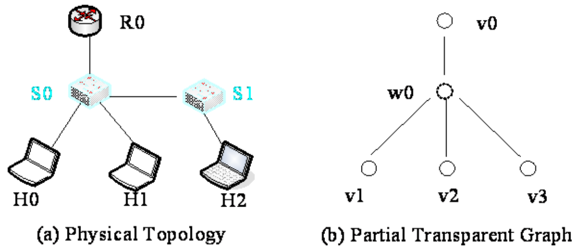


Figure 2. Topology modeling example 2

For the analysis of network layer naming and routing, a transparent node is actually an abstraction of one or more links or nodes of layer one or layer two.

III. Routing Analysis Model

A. Partial Transparent Graph: Related Terms, Definitions, and Notations

A partial transparent graph is a 2 layers graph, which contains two kinds of nodes, which belong to set V and set U respectively. For the analysis of network layer naming and routing, the upper layer nodes (nodes in set V) represent the nodes of the network layer(layer 3), such as routers or hosts, the lower layer nodes (nodes in set U) represent the nodes or links of the physical layer or data link layer (layer 1 or layer 2), such as switches, hubs or broadcast links. When modeling the topology of a layer computer network, we hope that as few auxiliary nodes as possible are introduced, while the properties are kept. In consideration that two or more nodes of the physical layer or data link layer connected directly can be abstracted as a transparent node, so there is a constraint with the nodes in set U : any two nodes in set U can't be connected directly in a partial transparent graph.

Definition 1: Let V , U , and E be finite sets, $V \neq \Phi$, $V \cap U = \Phi$. Let ψ be a function from E to $V \cup U$. $G = <V, U, E, \psi>$ is called a **partial transparent graph** if ψ satisfies

$$\psi\text{:}\ E \to \{\{v_1, v_2\} \mid (v_1 \in V \cup U) \wedge (v_2 \in V \cup U) \wedge (v_1 \in U \to v_2 \notin U) \wedge (v_2 \in U \to v_1 \notin U)\}$$

Definition 2: Let $G = <V, U, E, \Psi>$ be a partial transparent graph. The elements of V are called **opaque nodes**, the elements of U are called **transparent nodes**, the number of elements of V is called the **opaque order** of graph G, and the number of elements of U is called the **transparent order** of graph G.

The set of edges associated with node v is denoted by $C_\psi(v)$. If $\psi(e)=\{v_1,v_2\}$, then we denote $v_2=\psi(e,v_1)$.

Definition 3: Let $G=<V,U,E,\psi>$ be a partial transparent graph. $v_1,v_2 \in V$, v_1 is **opaque adjacent** with v_2 if one of the following conditions is met:

i. There is an edge $e\in E$, such that $\psi(e)=\{v_1,v_2\}$;

ii. There is n node $w\in U$ and two edges $e_1,e_2\in E$, such that $\psi(e_1)=\{v_1,w\}$ and $\psi(e_2)=\{w,v_2\}$.

The set of nodes that are **opaque adjacent** with v v is denoted by $A_{o\psi}(v)$. The size of $A_{o\psi}(v)$ is called the **opaque degree of** node v.

A partial transparent graph is essentially a graph, so the other concepts or definitions, such as **degree, ends, neighbors, adjacent, directed, undirected, weighted, path, connectivity,** and **trees,** remain the same as those in the traditional graph theory.

B. Graph Naming- and Routing-Related Terms, Definitions, and Notations

In the TCP/IP naming and routing scheme, a node is usually identified with one or more 32-bit or 128-bit binary numbers and two or more routes can be aggregated based on the relationship of their related binary numbers used for destination. We think that in future networks, the network layer naming schemes still be based on binary numbers and routes can be aggregated for scalability. So we introduced the following notations for modeling the network layer naming and routing, both for traditional networks and future networks.

B : set of 1-bit binary numbers, $B=\{0,1\}$;

k : a natural number;

B^k : set of k -bit binary numbers, $B^k=\{<b_1,b_2,\ldots,b_k>|\, b_i\in B, 1\le i\le k\}$;

0^k : k -bit binary number 0;

$\vartheta(B^k)$: power set of B^k;

I_k : the bijective function from B^k to the set of decimal numbers $\{0,1,\ldots,2^k-1\}$;

$I_k(<a_{k-1},a_{k-2},\ldots,a_0>)=\sum_{i=0}^{k-1}(a_i*2^i)$;

I_k^{-1} : if $I_k(x)=y$, then $x=I_k^{-1}(y)$;

M^k : set of k -bit binary numbers used for masks;

$M^k=\{<b_1,b_2,...,b_k>|(<b_1,b_2,...,b_k>\in B^k), and$
$\forall_{i,j}((1\le i\le j\le k)\wedge(b_j=1)\rightarrow(b_i=1))\}$

N_k : set of all sub-network identifiers, $N_k=B^k\times\{0,1,\ldots,k\}$;

Definition 4: Let $|V|\le 2^k$. ϕ is a full function from V to $\vartheta(B^k)$; that is, $\phi: V\rightarrow\vartheta(B^k)$. ϕ is a **naming** on V if ϕ satisfies the following conditions:

i. $\forall_v((v\in V)\rightarrow\phi(v)\ne\Phi)$;

ii. $\forall_{v_1,v_2}((v_1\in V\wedge v_2\in V\wedge v_1\ne v_2)\rightarrow\phi(v_1)\cap\phi(v_2)=\Phi)$.

In the traditional IP naming scheme or future network naming schemes, one node may have one or more names. So in this naming model, any node has one or more names, but a name can belong to only a node.

We denote the collection of all node names by $\Gamma=\bigcup_{v\in V}\phi(v)$. We denote $v=\delta^{-1}(x)$ if $x\in\phi(v)$.

The traditional IP routing scheme is based on the longest prefix match method. So in the following we model the network layer *key-fixed* routing schemes.

Definition 5: Let $G=<V,U,E,\psi>$ be a partial transparent graph. Let ϕ be a **naming** on V, and $\Gamma=\bigcup_{v\in V}\phi(v)$, $R\subseteq V\times\Gamma\times\Gamma$. R is a **basic routing** on G if R satisfies the following conditions:

i. $\forall v\forall x(v\in V\wedge x\in\Gamma\wedge x\notin\phi(v)\rightarrow$
$\exists y(y\in\Gamma\wedge A_{o\psi}(v,\delta^{-1}(y))\wedge<v,x,y>\in R))$

ii. Given two names, x and y, if $\delta^{-1}(x)\ne\delta^{-1}(y)$, then there exist names $x_1=x,x_2,...,x_n=y$, such that $\forall i((1\le i\le n-1)\rightarrow<\delta^{-1}(x_i),y,x_{i+1}>\in R$.

We denote $R(v)=\{<v,x,y>|<v,x,y>\in R\}$, which is called a **basic routing table** of node v.

Definition 6:

Let $p_1=<x_1,y_1>\in N_k$, $p_2=<x_2,y_2>\in N_k$, $x_1=<a_1,a_2,...,a_k>$, $x_2=<b_1,b_2,...,b_k>$. p_1 is called the **parent prefix** of p_2, and p_2 is called the **child prefix** of p_1 if the following conditions are satisfied:

i. $y_1<y_2$;

ii. $\forall i((1\le i\le y_1)\rightarrow a_i=b_i$.

We denote this relation by $P(p_1,p_2)$.

Obviously, this relation is transitive. $<0^k,0>$ is the parent prefix of any other element in N_k.

Definition 7: Let $G=<V,U,E,\psi>$ be a partial transparent graph. Let ϕ be a **naming** on V, and $\Gamma=\bigcup_{v\in V}\phi(v)$, $R'\subseteq V\times N_k\times\Gamma$. R' is a **aggregating routing** on G if R' satisfies the following conditions:

i. $\forall v\forall x((v\in V\wedge x\in\Gamma\wedge x\notin\phi(v)\rightarrow$
$\exists y\exists z(y\in N_k\wedge P(y,<x,k>)\wedge z\in$
$\Gamma\wedge A_{o\psi}(v,\delta^{-1}(z))\wedge<v,y,z>\in R'))$

ii. Given two names, x and y, if $\delta^{-1}(x) \neq \delta^{-1}(y)$, then there exist names $x_1 = x, x_2, ..., x_n = y$, such that $\forall i \exists z((1 \leq i \leq n-1) \rightarrow (P(z, <y, k>) \wedge <\delta^{-1}(x_i), z, x_{i+1}> \in R'))$.

We denote $R'(v) = \{<v, x, y, e> | <v, x, y, e> \in R'\}$, which is called an aggregating routing table of node v.

IV. Relationship among Static Topology, Naming, and Routing

In this section, we explore what best extent the scalability of network layer routing can be achieved to. We try to get the answer by explore a special kind of naming and routing schemes: *globally unique, topology-dependent naming, key-fixed, locator based, aggregating routing.*

We show that for any kind of static transparent connectivity graph, there exists a naming and a network routing, such that the routing table size of a node is very small, which is less than or equal to the maximum opaque degree among all nodes. We will give a theorem to characterize the features and prove it.

A. The Theorem and the Proof

Theorem 1: Let $G = <V, U, E, \psi>$ be a partial transparent graph. There exists a **naming** ϕ on V and a **aggregating routing** R, such that $\underset{v \in V}{Max} |R(v)| \leq \underset{v \in V}{Max}(|A_{o\psi}(v)|)$.

If we can find the ϕ and R, such that R satisfies the degree constraint, then this theorem can be proved. The complex proving process is in Appendix A. And an illustrational example is in Appendix B.

B. The Meaning of Theorem 1

Given a static network topology, if we adopt the hierarchical naming and the naming following the network topology properties, then the nodes in the network need not run any routing protocol, and each node only needs to maintain a little information for routing. Theorem 1 shows that network topology layering and topology-dependant naming are very beneficial to routing scalability. Therefore, for some special or specific types of networks, we can design specific naming scheme and proprietary protocols to achieve very good routing scalability and small routing overhead.

V. Relationship among Dynamic Topology, Naming, and Routing

In the last section, we show that very good routing scalability can be achieved when the topology of a network is static. However, future networks will be composed of a great deal of mobile devices. The topology-independent naming schemes are preferable to topology-dependent naming schemes for mobility, multi-homing and so on. So in this section, we explore the routing scalability in mobile networks. We try to get the answer by explore another special kind of naming and routing schemes: *globally unique, topology-independent naming, key-fixed, identifier based, aggregating routing.*

We show that the network storage overhead for routing cannot be sub-linear under some constraints, even if we do not require the shortest path routing. We will give another theorem to characterize the features and prove it.

A. The Theorem and the Proof

Theorem 2: Given a set of n_o network nodes and a number $\overline{d}$ ($\overline{d} > 2$), there exists a naming on these nodes and a network formed by these nodes, in which each node degree is not greater than $\overline{d}$, such that for any routing, the routing table size of this network is $\Omega(n_o)$.

Proof. Let $d = \overline{d} - 1$. We first consider two kinds of special cases: binary tree and d-ary-tree ($d = 2^w, w > 1$). We then consider a more general case.

Case 1: Binary tree ($d = 2$ **)**

First, we give a simple example to illustrate our ideas used to prove this theorem. In a special binary tree with 21 opaque nodes, as shown in Fig. 3 (cycles represent opaque nodes, and squares represent transparent nodes), each edge is labeled; the left branch is labeled 0, and the right branch is labeled 1. The partial identifier of a leaf node is formed by composing the labels from this leaf to the root together. By appending a bit after these identifiers, we can extend the 4-bits partial identifiers to 5-bits partial identifiers, and assign each internal opaque node a different identifier. We then get a naming. Considering that two routes can be aggregated only when they have the same output interface and the same next hop, these two route destinations have a common prefix and other conditions to guarantee correct routing. Thus, in this naming, the root node cannot aggregate any two routes to leaf nodes except for introducing a default route; the route number cannot be less than 8.

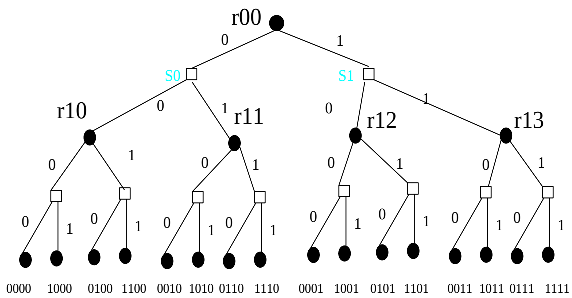


Figure 3. A binary tree naming example

Assume the tree height of this kind of special binary tree with n_o opaque nodes is h. The nodes of layer h are leaf nodes. Each node of layer h-2 cannot aggregate the routes to its sub-tree nodes. Similarly, each node of layer h-4 cannot aggregate the routes to leaf nodes. Ultimately, the root nodes cannot aggregate the routes to leaf nodes because of the difference in identifiers. That is, the root node needs to maintain a route for each leaf node if a default route is not used.

Let $k = h/2$. Then for a binary tree,

$n_o = 1 + 2^2 + 2^4 + ... + 2^{2k} = (1-(2^2)^{k+1})/(1-2^2)$.

Therefore, the number of leaf nodes $2^h = 2^{2k} = (3n_o + 1)/4$.

After introducing a default route, the number of routes to leaf nodes can be reduced by 50% at the most. Thus,

$|R| \geq (3n_o + 1)/8$.

Case 2: A d-ary-tree ($d = 2^w, w > 1$)

First, we label each branch of a node from left to right with a unique w-bit binary number. We then get a partial identifier of a leaf node by composing the labels from this leaf to the root together. Similar to that in a binary tree, the root nodes cannot aggregate the routes to leaf nodes because of the difference in identifiers.

Similarly, the number of leaf nodes is d^{2k}.

$d^{2k} = ((d^2 - 1)\ n_o + 1)/d^2 = (1 - 1/d^2)n_o + 1/d^2$.

After introducing a default route, the number of routes to leaf nodes can be reduced by d^{2k-1} at the most. Thus,

$|R| \geq d^{2k} - d^{2k-1} = (1-1/d)(1-1/d^2)n_o + (1-1/d)(1/d^2)$.

In the two special cases mentioned above, a relationship between the number of opaque nodes and the degree constraint exists. That is,

$n_o = (d^{2k+2}\text{-}1)/(d^2 - 1)$.

We will consider a general case in the following scenarios.

Case 3: General case

For a given number n_o and a degree constraint number $\overline{d}$ ($\overline{d}$ > 2), we can find a number w, such that $2^{w+1} > d \geq 2^w$.

Let $d' = 2^w$. We can find a number k, such that

$(d'^{2k+2}\text{-}1)/(d'^2 - 1) \leq n_o < (d'^{2(k+1)+2}\text{-}1)/(d'^{2(k+1)+2}\text{-}1)$.

Let $n_o' = (d'^{2k+2}\text{-}1)/(d'^2 - 1)$. According to Case 2, we can then construct a d-ary-tree with n_o' opaque nodes. In this sub-tree, the size of routing table R' of the root node is

$|R'| \geq d'^{2k} - d'^{2k-1} = (1\text{-}1/d')d'^{2k}$.

Given that $n_o < (d'^{2(k+1)+2}\text{-}1)/(d'^2 - 1)$, $d'^{2k} > (n_o(d'^2 - 1) + 1)/d'^4$.

Thus, $|R'| \geq (1\text{-}1/d')d'^{2k} > (1\text{-}1/d')(n_o(d'^2 - 1) + 1)/d'^4)$.

Obviously, we can extend this d-ary-tree to a tree with n_o opaque nodes and the same root node, which follows the degree constraint. The size of routing table R of the root node is

$|R| \geq |R'| > (1\text{-}1/d')(n_o(d'^2 - 1) + 1)/d'^4)$.

Hence, the routing table size of this network is Ω (n_o) for any routing.

This completes the proof of Theorem 2.

B. *The Meaning of Theorem 2*

The theorem shows that in a network whose nodes can move randomly, if we require that the nodes identifiers are independent of network topologies, and the length of identifiers is set as short as possible, the network storage overhead for routing cannot be sub-linear even if we do not require the shortest path routing. In other words, if we want to take advantage of name-independence and low packets header overhead, appropriate storage overhead is necessary.

VI. Discussion

Our work is closely related to past research work on the analysis of routing in distributed networks. Kleinrock and Kamoun [33] showed that the routing table length (the number of entries) could be reduced to $e\ln(n)$ with optimal hierarchical clustering structure. Our work shows that the routing table length can be reduced no more than the maximum node degree of a network. A lower bound of $\Omega(n)$ state is proven necessary to implement any local routing function of any stretch factor s<3 and stretch < k requires $\Omega(n^{2/(k-1)})$ [34]-[37]. Moreover, shortest path routing algorithms require $\Omega(n)$ local memory for a network with n destinations [37]. Our work indicates that $\Omega(n)$ local memory is needed even without the stretch constraint for a dynamic network with some naming constraint.

VII. Conclusion and Future Work

Simply modeling the topology of a layered computer network as a common graph loses some important information about physical topology. Therefore, we introduce a new concept called partial transparent graph. We think this kind of graph should be useful for analyzing networks.

For analyzing the relationship between topology, naming, and routing, we present a routing analysis model to formalize topologies of layered computer networks, network-layer naming, and routing. We prove that in a static topology with a structured naming, the routing table size can be very small and does not depend on the network size; in a dynamic topology in which a node can move randomly and with some naming constraints, the routing table size for routing on persistent short identifiers is linear with the network size.

Our current work is still preliminary. We only give the theoretical results for two special cases in this paper. But we think our routing analysis model provides a basic for analyzing other naming and routing schemes. And we hope some other researchers would like to focus on this topic, and work together to give the theoretical results for other kinds of naming and routing schemes, which can cast a light for future naming and routing design.

APPENDIX A PROOF OF THEOREM 1

Theorem 1: Let $G=<V,U,E,\psi>$ be a partial transparent graph. There exists a **naming** ϕ on V and a **aggregating routing** R, such that $\underset{v\in V}{Max}|R(v)|\le \underset{v\in V}{Max}(|A_{o\psi}(v)|)$.

Proof: If we can find the ϕ and R, such that R satisfies the degree constraint, then this theorem can be proved. The proving process includes the followings steps.

Step I: Given that G is connected, we can certainly **find one spanning tree** $T'=<V,U,E',\psi'>$. We introduce a notation $S_o(v)$ to represent all nodes of the sub-tree whose root is node v in tree T'.

Step II: **Construct a set of node names** according to the following sub-steps (which we call the prefix join methods):

Sub-step 1: **Initialization and selection of the root.** A node with the maximum opaque degree among all nodes in T' is selected as the root node, which is denoted by v_r'. Let partial identifiers set $N'=\Phi$, $T''=T'$, $\varphi_0(e)=<\varepsilon,0>$ for each $e\in E'$, $\varphi_0(v)=<\varepsilon,0>$ for each $v\in V\cup U$.

Sub-step 2: **Determination of partial auxiliary identifiers for adjacent edges and nodes**. Assume the root node of T'' is v_r'. If $|C_{\psi'}(v_r')|\ne 0$, then the following process is followed. Assume the set of edges associated with v_r' is $C_{\psi'}(v_r')=\{e_0,e_1,...,e_{m-1}\}$, if $m=1$, then let $k_0=1$; otherwise, let $k_0=\lceil\log_2(m)\rceil$. Let $\varphi_0(e_i)=<I_{k_0}^{-1}(i),k_0>$, $0\le i<m$, $N'=N'\cup\{<e_i,\varphi_0(e_i)>|0\le i<m\}$. For each edge e_i, let $u=\psi'(e_i,v_r')$. If $u\in V$, let $\varphi_0(u)=<I_{k_0}^{-1}(0),1>$; if $u\in U$, then let $A_{\psi'}(u)\text{-}\{v_r'\}=\{v_0,v_1,...,v_{m'-1}\}$, and if $m'=1$, let $k_0'=1$; otherwise, let $k_0'=\lceil\log_2(m')\rceil$. Let $\varphi_0(v_i)=<I_{k_0}^{-1}(i),k_0'>$, $0\le i<m'$.

Sub-step 3: **Determination of partial auxiliary identifier**s for other edges and nodes. For each node v in $A_{o\psi'}(v_r')$, assume T''_v is a sub-tree rooted at v in T'. Let $T''=T''_v$, go to sub-step 2.

Sub-step 4: **Determination of the node identifier length.** Let $x\in B^{k_0}\times\{k_0\}$, $x=<x_0,k_0>$, $x_0=<b_0,b_1,...,b_{k_0\text{-}1}>$, $x'\in B^{k_0'}\times\{k_0'\}$, $x'=<x_0',k_0'>$, $x_0'=<b_0',b_1',...,b_{k_0'\text{-}1}'>$. Let $x''=<b_0,b_1,...,b_{k_0\text{-}1},b_0',b_1',...,b_{k_0'\text{-}1}'>$, and a function J to represent the joint of x and x', $J(x,x')=<x'',k_0+k_0'>$, then $J(x_0,x_1,...,x_{n-1})=J(x_0,J(x_1,...,x_{n-1}))$. In particular, if $x'=<\varepsilon,0>$, then $J(x,x')=x$.

For each opaque node v in T', assume the simple path from the root v_r' to the node v is $v_r'e_0'v_1'e_1',...,v_{m-1}'e_{m-1}'v$. The partial auxiliary identifiers prefix is denoted by $\alpha(v)$, $\alpha(v)=J(\varphi_0(e_0'),\varphi_0(v_1'),\varphi_0(e_1'),\ ...,\varphi_0(e_{m-1}'),\varphi_0(v))$.

Assume $\alpha(v)=<p,k>$, let $l'(v)=k$, and let the node identifiers length be $k_0=\underset{v\in V}{Max}(l'(v))$.

Sub-step 5: **Determination of node identifiers.** Denote the set of <node v, the identifier prefix that has been assigned to v> by Ω. First let $\Omega=\Phi$. We can determine the node identifiers with the following recursive process, which is denoted by Naming (v), as shown in Fig. 4. This process is called initially with the root node as the input parameters.

/* Recursive Process Naming (v)*/
Naming (v)
{
  If there is not a prefix p, such that $<v,p>\in\Omega$,
  then
  {
    If v is a leaf node,
    Then let
    $\gamma(v)=J(\alpha(v),<I_{k_0-\alpha'(v)}^{-1}(1),k_0-\alpha'(v)>)$,
    $\Omega=\Omega\cup\{<v,\gamma(v)>\}$.
    Else
    {
      If all nodes in $S_o(v)$ have been named,
      Then
      find a minimum binary number x_0 that has the prefix $\alpha(v)$, such that
      $<x_0,k_0>\notin\{x|<v,x>\in\Omega\}$. Let
      $\Omega=\Omega\cup\{<v,<x_0,k_0>>\}$.
      Else
      for each child node v', called Naming (v').
    }
  }
}

Figure 4. Process for determining node identifiers

Eventually, we can get a naming ϕ_0 on nodes V:

$\phi_0(v)=\{x|<v,<x,k_0>>\in\Omega\}$.

Step III: **Construct a set of routing entries**.

Let $\Gamma_0 = \bigcup_{v\in V} \phi_0(v)$. We can construct a routing entry set R_0' ($R_0' \subseteq V \times N_{k_0} \times \Gamma_0$) by the following:

Let $x = << b_1, b_2, ..., b_k >, k >$, $k < k_0$. The k_0 -expansion is denoted by $\theta_{k_0}(x) = << b_1, b_2, ..., b_k, 0, ..., 0 >, k >$, $\theta_{k_0}(x) \in N_{k_0}$.

First, for any node $v \in V$, let $R_0'(v) = \Phi$.

Second, in the tree $T' = < V, U, E', \psi' >$, for each node v_s in $S_o(v)$, assume $y \in \phi_0(v_s)$. A route is added into the routing table of node v ; that is, $R_0'(v) = R_0'(v) \cup \{< v, \theta_{k_0}(\alpha(v_s)), y >\}$. If node v has a parent node v_p , and if v_p is an opaque node, then let $v_{op} = v_p$; otherwise, let v_{op} be the parent node of v_p . Assume $y' \in \phi_0(v_{op})$. A default route is added into the routing table of node v ; that is, $R_0'(v) = R_0'(v) \cup \{< v, p_d^{k_0}, y' >\}$.

Third, let $R_0' = \bigcup_{v\in V} R_0'(v)$.

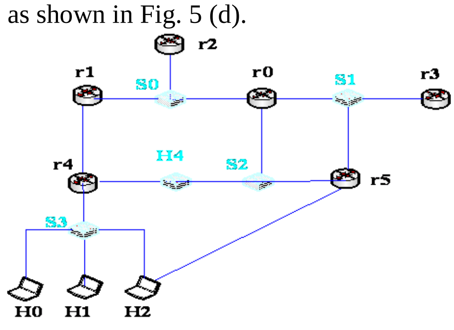
According to the construction process of R_0', R_0' is obviously a **aggregating routing** on G . The number of adjacent opaque nodes of a node in the tree $T' = < V, U, E', \psi' >$ is not greater than that in the original graph G , so $\underset{v\in V}{Max} | R_0'(v) | \le \underset{v\in V}{Max}(A_{o\psi}(v))$. Thus, we find a naming ϕ_0 and a network routing R_0' , such that $\underset{v\in V}{Max} | R_0'(v) | \le \underset{v\in V}{Max} |A_{o\psi}(v)|$.

This completes the proof of Theorem 1.

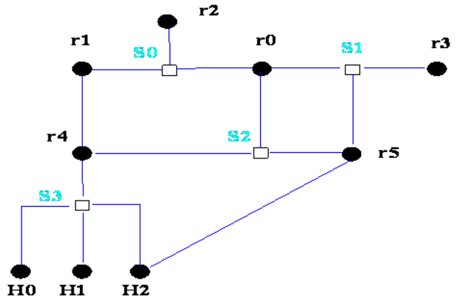
APPENDIX B AN ILLUSTRATIONAL EXAMPLE

To illustrate the above construction process of naming and routing in the proving process, we give a simple example here.

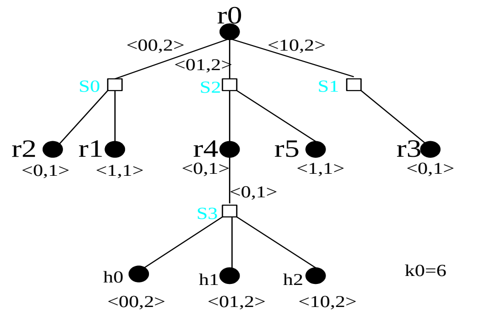
Given a network topology shown in Fig. 5 (a), which can be modeled as a partial transparent graph as shown in Fig. 5 (b), we can construct a transparent tree rooted at node r0 , as shown in Fig. 4 (c). Starting from the root, by following the sub-steps 2 and 3, we can get the partial auxiliary identifiers gradually from top to bottom, as shown in Fig. 5 (c). We can then determine that the identifiers' length value is 6 by following sub-step 4. Following sub-step 5, we can get the identifiers of all nodes, as shown in Fig. 5 (d).



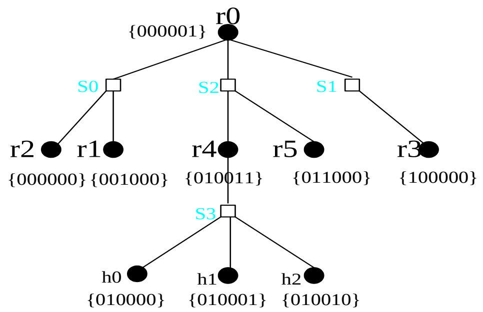
(a) Physical Topology



(b) Partial Transparent Graph



(c) Identifiers parts



(d) Identifiers

Figure 5. The illustration of naming and routing in a example topology

According to these identifiers and the transparent tree, we can construct a routing table for each node by following step III. For example, the routing table for node r0 is $R_0'(r0) = \{< r0, < 000,3 >, 000000 >$, $< r0, < 001,3 >, 001000 >$, $< r0, < 010,3 >, 010011 >$, $< r0, < 011,3 >, 011000 >$, $< r0, < 100,3 >, 100000 >\}$. Similarly, we can get the routing table of other nodes.

REFERENCES

[1] D. E. Comer, *Internetworking with TCP/IP Vol 1: Principles, Protocols, and Architecture*, 3rd ed. Prentice Hall, 1995.

[2] General requirements for ID/locator separation in NGN, ITU-T Standard Y.2015-2007.

[3] Report from the IAB Workshop on Routing and Addressing, IETF RFC 4984-2007.

[4] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proc. 5th International Conference on Emerging Networking Experiments and Technologies*, New York, 2009, pp. 1-12.

[5] U. Saif and J. M. Paluska, "Service-oriented network sockets," in *Proc. ACM MobiSys*, San Francisco, 2003.

[6] T. Koponen, M. Chawla, B. chun, *et al,* "A data-oriented (and beyond) network architecture," *ACM SIGCOMM Computer Communication Review*, vol. 37, pp. 181-192, October 2007.

[7] J. Saltzer, "On the naming and binding of network destinations," *Local Computer Networks*, pp. 311–317, April 1982.

[8] J. Day, *Patterns in Network Architecture: A Return to Fundamentals,* 1st ed. Prentice Hall, 2008.

[9] Information Processing Systems—Open System Interconnection— Basic Reference Model, ISO/IEC 7498-1994.

[10] A Proposed Enhancement to the Internet Architecture, IETF internet-draft-1995.

[11] TCP & UDP with Network-independent Endpoints (TUNE), IETF internet-draft-1992.

[12] B. Ahlgren, J. Arkko, L. Eggert, and J. Rajahalme, "A node identity internetworking architecture," in *Proc. IEEE INFOCOM Global Internet Symp,* Barcelona, 2006, pp. 1−6.

[13] R. Atkinson, S. Bhatti, and S. Hailes, "Evolving the internet architecture through naming," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 8, pp. 1319–1325, October 2010.

[14] D. Jen, M. Meisel, H. Yan, D. Massey, L. Wang, B. Zhang, and L. Zhang, "Towards a new internet routing architecture: Arguments for separating edges from transit core," in *Proc. ACM HotNets*, Calgary, Canada, 2008.

[15] D. Clark, R. Braden, A. Falk, and V. Pingali, "FARA: Reorganizing the addressing architecture," *ACM Computer Communications Review*, vol. 33, no.4, pp. 313-321, October 2003.

[16] H. Balakrishnan, K. Lakshminarayanan, S. Ratnasamy, S. Shenker,I. Stoica, and M. Walfish, "A layered naming architecture for the internet," in *Proc. Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications,* New York, 2004, pp. 343–352.

[17] S. Jain, Y. Y. Chen, Z. L. Zhang, and S. Jain, "VIRO: A scalable, robust and namespace independent virtual id routing for future networks," in *Proc. INFOCOM*, Shanghai, 2011, pp. 2381-2389.

[18] J. Hou, Y. Liu, and Z. Gong, "Silms: A scalable and secure identifierto-locator mapping service system design for future internet," in *Proc. International Workshop on Computer Science and Engineering*, Qingdao, China, 2009, pp. 54–58.

[19] J. Hou, Y. P. Liu, and Z. H. Gong, "A framework for identifier-based routing for future internet," in *Proc. Eighth IEEE International Conference on Dependable, Autonomic and Secure Computing*, Chengdu, China, 2009.

[20] R. Tu, J. Su, Z. Meng, and F. Zhao, "UCEN: User centric enterprise network," in *Proc. CACT*, 2008.

[21] M. Caesar, T. Condie, J. Kannan, K. Lakshminarayanan, I. Stoica, and S. Shenker, "ROFL: Routing on flat labels," in *Proc. SIGCOMM,* Pisa, 2006, pp. 363–374.

[22] A. Singla, P. B. Godfrey, K. Fall, G. Iannaccone, and S. Ratnasamy, "Scalable routing on flat names," in *Proc. ACM CoNEXT* , Philadelphia, 2010.

[23] Locator/ID Separation Protocol (LISP), IETF Internet-Draft-2013.

[24] C. Vogt, "Six/one router: A scalable and backwards compatible solution for provider-independent addressing," in *Proc. 3rd International Workshop on Mobility in the Evolving Internet Architecture*, New York, 2008, pp. 13–18.

[25] R. Atkinson, S. Bhatti, and S. Hailes, "ILNP: Mobility, multi-homing, localised addressing and security through naming," *Telecommunication System*s, vol. 42, pp. 273–291, December 2009.

[26] R. Hinden, New scheme for Internet routing and addressing (ENCAPS) for IPNG, IETF RFC 1955-1996.

[27] A proposal for scalable Internet routing and addressing, IETF Internet Draft- 2007.

[28] Host Identity Protocol (HIP) Architecture, IETF RFC 4423-2006.

[29] Shim6: Level 3 Multihoming Shim Protocol for IPv6, IETF RFC 5533-2009.

[30] J. Pan, S. Paul, R. Jain, *et al*, "MILSA: A mobility and multi-homing supporting identifier locator split architecture for next generation internet," in *Proc. IEEE GLOBECOM*, New Orleans, 2008.

[31] J. Pan, S. Paul, R. Jain, *et al*, "Enhanced MILSA architecture for naming, addressing, routing and security issues in the next generation internet," in *Proc. IEEE ICC*, Dresden, 2009.

[32] I. Stoica, D. Adkins, *et al*, "Internet indirection infrastructure," in *Proc. ACM SIGCOMM*, Pittsburgh, 2002.

[33] L. Kleinrock and F. Kamoun. "Hierarchical routing for large networks: Performance evaluation and optimization," *Computer Networks*, vol. 1, pp. 155-174, January 1977.

[34] C. Gavoille and M. Gengle, "Space-efficiency for routing schemes of stretch factor three," *J. Parallel Distrib. Comput.* vol. 61, pp. 679-687, 2001.

[35] M. Thorup and U. Zwick, "Compact routing schemes," in *Proc. SPAA*, 2001, pp. 1-10.

[36] I. Abraham, C. Gavoille, D. Malkhi, N. Nisan, and M. Thorup, "Compact name-independent routing with minimum stretch," *ACM Transactions on Algorithms*, vol. 4, no.3, pp. 1-12, 2008.

[37] P. Fraigniaud and C. Gavoille, "Memory requirement for universal routing schemes in *Proc. PODC*, Ottawa, 1995, pp. 223-230.

Feng Zhao was born in 1980 in the province of Yunnan, China. He received the B.S., M.S. and Ph.D. degrees in computer science and technology from National University of Defense Technology in 2000, 2002 and 2007, respectively. He is currently an assistant professor at the College of Computer, National University of Defense Technology. His research interests include computer network, future Internet architecture, and network science and so on.