An Impact of Addressing Schemes on Routing Scalability

Huaiyuan Ma, Bjarne E. Helvik, and Otto J. Wittner

Abstract: **The inter-domain routing scalability issue is a major challenge facing the Internet. Recent wide deployments of multi-homing and traffic engineering urge for solutions to this issue. So far, tunnel-based proposals and compact routing schemes have been suggested. An implicit assumption in the routing community is that structured address labels are crucial for routing scalability. This paper first systematically examines the properties of identifiers and address labels and their functional differences. It develops a simple Internet routing model and shows that a binary relation T defined on the address label set A determines the cardinality of the compact label set L. Furthermore, it is shown that routing schemes based on flat address labels are not scalable. This implies that routing scalability and routing stability are inherently related and must be considered together when a routing scheme is evaluated. Furthermore, a metric is defined to measure the efficiency of the address label coding. Simulations show that given a 3000-autonomous system (AS) topology, the required length of address labels in compact routing schemes is only 9.12 bits while the required length is 10.64 bits for the Internet protocol (IP) upper bound case. Simulations also show that the α values of the compact routing and IP routing schemes are 0.80 and 0.95, respectively, for a 3000-AS topology. This indicates that a compact routing scheme with necessary routing stability is desirable. It is also seen that using provider allocated IP addresses in multihomed stub ASs does not significantly reduce the global routing size of an IP routing system.**

Index Terms: **Address label, compact routing, inter-domain routing, routing scalability, structured addresses.**

I. INTRODUCTION

As a global infrastructure, the Internet is crucial for the daily operation of our global village. The border gateway protocol (BGP) enables all autonomous systems (ASs) to interconnect with each other. Routing scalability is achieved through routing aggregation in Internet protocol version 4/version 6 (IPv4/v6). Routing aggregation requires IP addresses to be allocated hierarchically. That is, routing aggregation cannot occur unless a downstream Internet service provider's (ISP) entire IP address space is part of its immediate upstream ISP's IP address space. To achieve routing scalability, when a customer network changes its ISP, it must apply for and use provider-allocated (PA) IP addresses from its new immediate ISP's IP address space. This causes two major issues. Not only must all the customer network's equipments such as domain name servers (DNSes) be reconfigured, but also all programs embedding IP addresses from its old ISP's IP address space must be recompiled with new IP addresses from its new ISP's IP address space. This is cumbersome. Therefore, customer networks usually apply for its own provider-independent (PI) IP addresses from corresponding IP address administration authorities. In this way, the side effects of changing ISP can be minimized. This practice makes it impossible for the upstream providers to aggregate the PI IP prefixes from its customer networks, resulting in a routing scalability issue. In addition, a customer network may secure its connection to the Internet by multi-homing to two or more upstream providers. Load balancing is enforced over these links. An IP prefix with a shorter subnet mask is subdivided by a customer network into two or more IP prefixes with longer subnet masks that are announced towards the different upstream providers. This practice makes the routing scalability issue even worse and harder to solve.

It has been shown that the global BGP routing table growth is potentially exponential, i.e., faster than the growth of the number of ASs in the Internet [1]. Note that large BGP routing tables do not just cause major difficulties with respect to storage in BGP routers, frequent route updates, implied by a large set of global routing entries, is another cause of overload in BGP routers.

Currently, Internet research task force (IRTF) has a dedicated working group looking into proposals for handling routing scalability issues [2]–[5]. The proposed tunnel-based solutions relies on hierarchical IP routing aggregation.

Internet's inter-domain topology now resembles a small world topology [6]. No routing scheme relying on routing aggregation can work well on a small world graph [7]. The identifier/locator mapping dissemination introduces an additional severe challenge. It has been shown that such mechanism does not scale [7].

Compact routing schemes, which have routing scalability as their major design objective, have emerged as a promising approach for scalable routing. The key idea is to allow paths to be somewhat longer than the shortest paths, i.e., to stretch them, and thereby achieve smaller routing tables. These schemes show promising scaling performance, but have minor drawbacks in terms of the average routing stretch for stationary networks [8]–[15]. Unfortunately, current compact routing schemes are found to lack the ability to handle network dynamics resulting from network failures, maintenance, reconfigurations and so on [16].

So far, the approaches and proposals suggested have been based on structured address labels. Routing systems based on structured address labels can achieve routing scalability statically. However, when a node's structured address label cannot reflect its correct location in a topology, routing scalability is jeopardized. So, what about flat address labels? Does any scalable routing scheme exist that relies only on flat address labels? This paper seeks to answer this question by developing a simple Internet routing model and discussing thoroughly the address label's impact on routing scalability.

Manuscript received May 8, 2011.

H. Ma and B. E. Helvik are with the Centre for Quantifiable Quality of Service in Communication Systems, Norwegian University of Science and Technology, Norway, email: {huaiyuan, bjarne}@q2s.ntnu.no.

O. J. Wittner is with the IoU Department, UNINETT, 7465, Norway, email: otto.wittner@uninett.no.

1229-2370/11/$10.00 © 2011 KICS

Further, which routing scheme provides better address label coding efficiency? Which routing scheme has better routing scalability in terms of global routing table size? This paper attempts to answer these questions by developing metrics for address coding efficiency and presenting simulation results of how well selected routing schemes perform on different Internet topologies.

The rest of the paper is organized as follows. Section II reviews the state of the art in routing scalability. Section III introduces the concepts of identifier and address label, gives their formal definitions and discusses their properties. Section IV gives a formal definition of routing scalability. Section V presents a simple routing model and analyzes the impact of the address label structure on the routing scalability from a set theory point of view. Section VI compares the different routing schemes in terms of the address label coding efficiency and routing table size in a quantifiable way. Finally, some concluding remarks are given in Section VII.

II. RELATED WORK

A commonly recognized cause of routing scalability problems in the current Internet architecture is that the IP address has roles as both an identifier and a locator. An immediate and natural solution is to separate the dual roles of the IP address. This strategy has been adopted by a range of proposed future routing schemes. This paper does not attempt to investigate all of these schemes. However, many schemes have great similarities, and by examining a selection of schemes, each having features shared by a group of similar schemes, the authors believe that valuable knowledge may be gained. In the following a (not comprehensive) list of future routing schemes is presented. Note that only compact routing and traditional IP routing will be dealt with later in the paper.

Locator/identity (ID) separation protocol (LISP) [2] associates an ID with a set of locator addresses. The end-site addresses of hosts and routers in the edge network are seen as IDs from the backbone network, the associated locators indicate their locations. Packets are tunneled in the backbone network via an IP tunnel. The far endpoint of the IP tunnel is the corresponding locator of a host (only IP tunnels are applicable, i.e., multiple protocol label switch (MPLS) tunnels are not applicable in this scenario). The topology inter-connecting all locators should be a full mesh of IP tunnels. Managing the resulting large number of soft states of IP tunnels is a severe challenge. Furthermore, it is not obvious how the various types of network operators' investments and returns from the scheme will be aligned. Finally, a scalable ID-locator mapping service is required, which is far from obvious how to implement.

The generic stream encapsulation (GSE) scheme [4] differs from LISP in its IP address partitioning strategy. The address space is partitioned into three parts, where the lower N bytes are the end system designator (ESD), which serves as an ID; the middle M bytes represent the site topology partition (STP) for local routing; and the top (16-M-N) bytes are routing goop (RG) used for routing between providers. An RG is a locator representing the backbone-network entry point of an edge network. Hence, RG is only changed when an edge network changes its provider. GSE encounters the same challenges as the LISP scheme.

The core router-integrated overlay (CRIO) scheme's objective is to trade an increase in the length of the forwarding path for a small global routing table. A new concept, virtual-prefix, is introduced. A virtual-prefix is a super-prefix that spans a large portion of the address space. A mapping is a relationship between a prefix and a tunnel endpoint. A router that advertises a given virtual prefix must hold the mappings for every prefix within the virtual prefix. This means that only a router that has the global routing table is eligible to advertise virtual prefixes. A router with no mapping for the prefix that includes the destination address of a packet will forward the packet toward the router announcing the corresponding virtual prefix. That router knows which tunnel endpoint it should forward the packet to. Simulation results show that CRIO can shrink the BGP forwarding information base (FIB) by nearly two orders of magnitude, the global FIB by one order of magnitude, and the virtual private network (VPN) FIB by ten to twenty times, all with very little increase in overall path length. It has been seen that forwarding path optimization under the constraint of the inter-domain business model, is an important issue. Since BGP has stringent rules on route aggregation, the virtual prefix concept can be considered as a release of route aggregation restrictions, and potential routing loops should be handled properly.

Compact routing schemes have emerged as a promising approach for scalable routing. An important metric with respect to compact routing is routing stretch, which is defined as the worst-case relative increase of the path-length compared to the shortest path. The first stretch-3 scheme (i.e., all paths were guaranteed to have path lengths less than or equal to 3 times that of the shortest path) was suggested by L. Cowen [13]. Its address structure comprises three parts: The landmark of a node; the identifier of the node; and the outgoing interface of the landmark through which the node can be reached. A packet is forwarded according to its identifier field if the identifier is found. Otherwise, it is forwarded toward its landmark. The scheme is generic and works for any topology. It has a fixed maximum stretch of 3 for any topology and a sub-linear routing table size with an upper limit $\tilde{O}(n^{\frac{2}{3}})$. Cowen's original scheme requires a centralized entity dealing with the nomination of landmarks, etc. However, a distributed compact routing algorithm based on Cowen's scheme was developed in [12].

Another compact routing scheme was suggested by M. Thorup and U. Zwick (TZ) [14]. The TZ scheme improved the Cowen's results, obtaining a routing table size upper bound $\tilde{O}(n^{\frac{1}{3}})$ while maintaining stretch-3. The scheme requires running a central landmark selection algorithm that requires the entire topology for operation. To the authors's knowledge, no distributed variant of the scheme has yet been developed.

When Cowen's scheme and the TZ scheme are applied to an Internet inter-domain level topology, some highly interconnected ASs are not meant to install routing entries towards all their immediate neighbours. In [10], by releasing this constraint, a smaller stretch is achieved while still keeping the routing scheme scalable.

To be practical, any routing scheme should handle network dynamics resulting from network element failures, network growth, maintenances, reconfigurations, etc. in a proper way. A scheme's capability in this respect can be measured in terms of, for instance, routing convergence speed and routing stability. Unfortunately, current compact routing schemes are found to lack the necessary routing stability when subjected to network dynamics [16].

The powernet compact routing scheme [9] makes use of the soft hierarchy existing in a small world topology, and achieves routing scalability by carrying the upstream path in the packet without the proven stretch constraint. A distributed algorithm that uses AS numbers as the locators (landmarks) for IP prefixes has been developed without the proven stretch limit [15].

III. IDENTIFIER AND ADDRESSES

An identifier is used to determine the sameness of something. In a network, to identify a node uniquely, an identifier $i \in I$ is only allowed to be assigned to at most one node. Here, $I = \{i_1, i_2, \cdots, i_m\}$ is the set of all nodes' identifiers in a network G. V is the set of all vertices of G, and $|V| = m$. An identifier's single and exclusive function is to identify a node. Any two different nodes must have different identifiers. This property can be expressed as follows

$$\forall\,\text{node}_k,\ \text{node}_j \in G,\ \text{where}\ k \neq j \implies i_k \neq i_j. \quad (1)$$

The only operation defined on I is the mathematical inequality operation $\neq$. Note that a node may have more than one identifier. This paper focuses on the impact address label variants have on routing scalability. An identifier may also, in addition to being an identifier, act as the address label of the node identified.

A. Address Labels

An address label is in general an indication of a location in a network, but it is tightly coupled with a routing system. Hence, to be useful, an address label must be routable, i.e., there must be a routing system with knowledge about these address labels and how to reach the locations indicated by them. A node in a network should have both an identifier and an address label. Depending on the routing scheme applied, a node's identifier and address label may be identical, overlapping, or different. For instance, a public network IP address is both an identifier and a locator in the global transit network (GTN), which comprises tier-1, tier-2, and tier-3 networks. However, a VPN IP address is invisible and is only seen as an identifier in the GTN. Then, again, it functions as an identifier and a locator in the corresponding VPN. The current practice shows that the method of constructing address labels of a routing scheme is crucial for its scalability. Therefore, the relationship between an identifier and its corresponding address label(s) should be discussed thoroughly. The extent of coupling between them depends on how the identifiers and address labels are constructed and allocated in a routing scheme.

Address labels can have different granularities, i.e., they can be dedicated or coarse. A dedicated address label can only indicate the location of a single node in a topology. Here, $A = \{a_1, a_2, \cdots, a_k\}$ is a set of address labels, in which each element a_i is the address label of a single node. Keep in mind that a node can have more than one address labels simultaneously, i.e., a node's identifier and address label have a $1 : n$ mapping. That is, a node i_j's address label set A_{i_j} s.t. $|A_{i_j}| = n$ and $n \geq 1$ consists of n corresponding address labels, where n may vary for each individual identifier depending on a routing scheme.

A coarse address label can denote the locations of multiple nodes. Typically, $L = \{l_1, l_2, \cdots, l_s\}$ is a set of coarse address labels, where each address label $l \in L$ can represent multiple related address labels from the set A. It is modeled by a mapping as follows

$$F_T : \mathcal{P}(\mathcal{A}) \to \mathcal{L} \quad (2)$$

where $\mathcal{P}(\mathcal{A})$ is a power set of A.

B. Flat and Structured Addresses

This section formulates the properties of flat and structured address labels in topological terms. There exist many addressing schemes. For example, an IP address with one prefix defines a two-level hierarchy. An IP address with more prefixes may define a multi-level hierarchy. There are also addresses using geographic coordinates. In this paper, multilevel hierarchical addressing schemes are addressed.

In general, $T(a_i, a_j)$ is a predicate on whether two address labels a_i and a_j satisfy a location relationship T in a topology. If $T(a_i, a_j) = \text{true}$, $\exists l_k \in L$, s.t. $F_T(\{a_i, a_j\}) = l_k$. In different routing schemes, T can be defined from different perspectives. On the other hand, multiple predicates can be defined on the same address label set A simultaneously. For example, for a cluster-based routing scheme, $T(a_i, a_j)$ can be defined as a predicate on whether labels a_i and a_j belong to the same cluster or not. For IPv4, $T(a_i, a_j)$ can be a predicate on whether IP addresses a_i and a_j share a common longest IP prefix or not. Both a cluster and an IP prefix are regarded as representing a location in a topology.

If all node identifiers also act as address labels, i.e., $I = A$ where $|I| = m$, and the identifiers are not allocated according to the topology, then no topology information is embedded into the address labels. Then, given two identifiers/address labels i_k and i_j, we cannot infer whether the nodes identified by i_k and i_j are neighbors in a topology or not. In the rest of this paper, such address labels are said to be flat or unstructured.

A predicate T_I on a flat address label set $A = I$ defines a relationship between any two nodes identified by flat address labels a_i and a_j in a topology. If there exists $T_I(a_i, a_j) = \text{true}, i \neq j$, then a_i and a_j are topology dependent. Put another way, they belong to the same group, which can use the same routing entry. In a routing scheme based on flat address labels, a_i and a_j are not allocated according to a topology and they are not correlated. That is, $T_I(a_i, a_j) = \text{false}$ unless $a_i = a_j \in A$.

As an example, in chord [22], a well known peer-to-peer (P2P) network architecture, routing scalability is achieved by allocating address labels following a ring topology. The address labels in the routing table are used to partition the whole address label space into multiple groups. Only one routing entry is needed for each group. According to the definition of T_I, it is clear that the address labels of chord are structured (not flat). Let us define $\mathcal{T}$ to be the set consisting of all "valid" predicates T defined on A. A "valid" predicate T is either a tautology or a satisfiable formula defined on the k-dimensional space

A^k for $k \geq 2$; otherwise, it is not included in $\mathcal{T}$. A routing scheme based on flat address labels is one with $I = A$ and $\mathcal{T}= \{T_I\}$. In this case, $\mathcal{T}$ is a singularity with a single predicate included. Unicast media access control (MAC) addresses are good examples of flat address labels. On the other hand, for any routing scheme, if $\mathcal{T} - \{\mathcal{T}_\mathcal{I}\} \neq \emptyset$, its address label is structured.

C. Routing Schemes

A routing scheme on a graph G can be characterized by $S = \{G, I, A, L, \mathcal{T}\}$. A routing scheme based on flat address labels is characterized by $S = \{G, I, I, L, \{T_I\}\}$, in this case, $I = A$ and $\mathcal{T}= \{T_I\}$.

Theoretically, any node i_k's address label a_{i_k} can be equivalently transformed as $l_{i1}l_{i2}\cdots l_{ir}i_k$, where $r \geq 0$. It can be seen that $l_{i1}l_{i2}\cdots l_{ir}i_k$, for $r \geq 0$, are encoded into a_{i_k}. For a routing scheme based on flat address labels, $a_{i_k} = i_k$. For an IPv4 address, if l_{i1} is an IP prefix with a mask length of 18 bits, l_{i2} would be a more specific IP prefix, where its first 18 bits will be the same as l_{i1} and the 19th bit will be the same as the 19th bit of the original IP address. If structured address labels are allocated according to the topology, a routing scheme can take advantage of the internal structures existing in the address labels to make the routing scalable. It is well known that routers in the Internet make forwarding decisions based on destinations. In this example, l_{i1} is an address label, and traffic going to node i_k can be forwarded toward l_{i1} first, and then toward l_{i2}, and so on, until it reaches the destination i_k. Hence, routers on the way to l_{i1} only need to install a routing entry for l_{i1}. Each address label $l_{i1}, l_{i2}, \cdots, l_{ik}, i_k$ needs only to be installed on relevant routers along the path to the destination node i_k.

In contrast, if a node i_k's address label is $a_{i_k} = i_k$, without the internal structure inside a_{i_k}, all routers have to install a routing entry for a_{i_k}. Typically, the number of hosts allowed in a network is enormous. In this case, it is commonly thought that such a routing scheme cannot be scalable. On the other hand, a routing scheme $S = \{G, I, I, L, \{T_I\}\}$ has a major merit: It does not need to perform network renumbering when the network topology changes. In general, network renumbering can affect session availability and performing it is cumbersome and should be avoided[16]. Therefore, a scalable routing scheme $S = \{G, I, I, L, \{T_I\}\}$ will be highly desirable.

In the following sections, we first give a formal definition of routing scalability. Then, we study the relationship between routing scalability and structured/structureless address labels.

IV. ROUTING SCALABILITY

In general, routing scalability should be interpreted as a routing scheme's ability to control the number of routes generated to accommodate growth and dynamics in a network topology. Before a quantifiable measure for routing scalability can be introduced, the length of an address label a_i, as a bitstring, need to be determined.

Consider a network with address label set $A = \{a_i\}$, where $\|a_i\|$ denotes the length of the address label a_i in bits. According to information theory, $\|a_i\| = \log(1/P(a_i))$. In this paper, by default, each address label a_i denotes the location of a single node, i.e., $P(a_i) \leq m^{-1}$ and all address labels in A have the same length. On the other hand, labels in L can indicate the locations of multiple nodes at the same time, therefore, their lengths are usually shorter.

We use the definition in [13] to measure routing scalability. If S_k is the routing table size of a router in a network, the routing scalability w.r.t a routing scheme is defined as $R(|A|)$ as follows

$$\exists S_i, \forall S_j, i \neq j, S_i \geq S_j \tag{3}$$

$$R(|A|) = S_i \leq k|A|^\alpha, k > 0, \alpha \in [0, 1] \tag{4}$$

where k is a constant, S_i is the maximum routing table size in the given network and α indicates the extent of a routing scheme's scalability. The smaller α is the more scalable the routing system is.

V. A SIMPLE INTERNET ROUTING MODEL

Recall that a routing scheme on a graph G can be characterized by $S = \{G, I, A, L, \mathcal{T}\}$, where $\mathcal{T}$ is a set consisting of all predicates defined on A. Here, we let $\mathcal{T} = \{T_1, T_2, \cdots, T_s\}$ be the set of relations on A. More precisely, we define a function on A to be

$$F : A \times A \rightarrow L \tag{5}$$

The function $F(a_i, a_j) = l$, where $a_i, a_j \in A$ and $l \in L$, states that two address labels $a_i, a_j \in A$ share a common compact address label $l \in L$. Based on the function F, a binary relation T on A can be defined as follows

$$T = \{(a_i, a_j)|F(a_i, a_j) = l, a_i, a_j \in A, l \in L\}. \tag{6}$$

It can be seen that all binary relations T on A are reflexive, symmetric, and transitive, i.e., they are equivalent relations on A. The binary relation T determines a partition $\{[a]_T|a \in A\}$, where $[a]_T$ is the address label a's equivalent class. The number of equivalent relations is equal to the number of partitions. Since $T(a_i, a_j) = l, \forall a_i, a_j \in [a]_T$ holds, and each equivalent class $[a]_T$ corresponds to a compact address label l. Therefore, the binary relation T determines the cardinality of the compact label set L.

As pointed out before, a flat address label space is one with $I = A$ and $\mathcal{T}= \{T_I\}$. If the address labels are structured, equivalently, $|\mathcal{T}| > 1$, and these features are used for achieving routing scalability. From another point of view, these features can also be regarded as constraints imposed on the address label space. When such constraints are not satisfied any longer, the routing scalability relying on them becomes invalid [1]. Therefore, a scalable routing scheme based on flat address labels is desirable. The question is, does such a routing scheme exist?

Theorem 1: A scalable routing scheme based on flat address labels $S = \{G, I, I, L, \{T_I\}\}$ with $A = I$ and $\mathcal{T}= \{T_I\}$ does not exist.

Proof: For a routing scheme based on flat address labels, the only function on A is F_I.

$$F_I(a_i, a_j) = \begin{cases} a_i, \ a_i = a_j, \\ \perp, \ a_i \neq a_j. \end{cases} \tag{7}$$

Here, $F_I(a_i, a_j) = a_i$, where $a_i = a_j \in A$, states that an address label a_i is only correlated with itself from the location point of view. On the other hand, $F_I(a_i, a_j) = \perp$, where

$a_i \neq a_j \in A$, indicates that the relation between a_i and a_j is undefined. Put another way, no common address label can be found for a_i and a_j. The only binary relation on A is $T_I = \{(a_i, a_i) | F(a_i, a_i) = a_i = l, a_i \in A, l \in L\}$. In this case, an address label $a \in A$'s equivalent class is $[a]_{T_I} = \{a\}$. Here, T_I defines the finest partition on A. Therefore, $A = L$. A routing scheme based on flat address labels is not scalable. □

Accepting Theorem 1, the rest of this paper investigates the scalability properties of compact routing and IP routing schemes with flat label routing kept as a reference.

Theoretically, a node i_d's address label a_{i_d} can be decomposed to $l_{i1}l_{i2}\cdots l_{ik}i_d$. Given a network G with an identifier set I, corresponding address label set A and compact address label set L, it is possible to express the theoretical address coding limit of an address scheme in bits by applying Shannon entropy. For a routing scheme $S = \{G, I, A, L, \mathcal{T}\}$, the coding length is given by

$$S_R = \sum_{l \in L \cup I} P(l) \log \frac{1}{P(l)} \tag{8}$$

where $P(l)$ is the number of occurrences of an address label l relative to the total number of occurrences of address labels, i.e.,

$$P(l) = \frac{\sum\limits_{a \in A} \mathcal{I}(l, a)}{\sum\limits_{l \in L \cup I} \sum\limits_{a \in A} \mathcal{I}(l, a)} \tag{9}$$

where $\mathcal{I}\ (l, a)$ is an indicator function given by

$$\mathcal{I}(l, a) = \begin{cases} 1, & l \in a \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

Thus, $P(l)$ is essentially the relative frequency with which l appears in A. Then, according to the Shannon entropy, (8) gives the average address label length of a routing scheme in bits.

VI. SIMULATION STUDY

This section presents simulation studies investigating the scalability of three well known routing schemes: Flat label routing, IP routing, and compact routing. Based on the discussion in subsection III-B, a routing scheme has two kinds of scalability: (1) Address label scalability and (2) routing table scalability. Address label scalability relates to how the length of address labels, expressed in bits by (8), grows with respect to the growth in network size. Routing table scalability relates to how the required number of entries in the FIB of a node grows with respect to growth in network size, and also to how frequently the entries are updated. It is well known that routing table scalability has a significant impact not only on the memory size required to store FIBs, but also on routing convergence speeds when the system is subjected to network dynamics. The Internet routing is policy-based and is governed by the business relationship between any pair of immediate nighboring ASes. Therefore, keeping business models in mind is equally important when analyzing routing table scalability. Achieving both kinds of scalability simultaneously for a routing scheme is desirable. Development of such schemes will be pursued in future work by the authors.

A. Scenario

It is an objective of our simulation study that the simulated topology should have the same properties as the real Internet AS topology. Faloutsos *et al.* found that the Internet AS topology approximately followed a power law distribution, where the frequency of nodes with degree d is proportional to d raised to the power of a constant O, i.e., $f(d) \propto d^O$, [6]. However, in [18] it is pointed out that this power law distribution holds only if 1.5% to 2% of the highest degree nodes are removed from the topology. To perform simulations on a topology with these properties, two Internet-like inter-domain topologies with 3000 ASs and 2500 ASs have been generated based on the Dimitropoulos method [17], [19]–[21]. It has been verified that the generated topologies have the desirable frequency distribution of nodes with a degree larger than d, i.e., $\bar{F}(d) = \sum_{i=d}^{\infty} f(d)$. Furthermore, a fundamental difference between intra and inter-domain network topologies is that an intra-domain network is governed by an economical autonomous entity, while the inter-domain network topology is governed by a business model. The business model is based on three business relationships between adjacent nodes in the topology: Provider-customer, customer-provider and peer-peer. This model plays a crucial role in the current IP network.

B. Simulation Approach

When the above mentioned business model is embedded in a topology model, realistic BGP behavior may be simulated in some detail: a) IP prefixes are allocated hierarchically and routes are aggregated when possible; and b) a simple stub AS with a single provider only requires a default route, and a multihomed stub AS receives all route announcements from different points of presence (PoPs) in order to load balance the outgoing traffic. These behaviours are consistent with current commercial BGP practices. The Internet inter-domain routing is policy-based, and route selection, announcement, and withdrawal policies are configured based on the business relationship between two neighboring AS networks. For route announcement, "valley-free" routing is enforced in our simulation. That is, an AS only announces the routes received from its providers toward its customers (never toward its other peers or providers). An AS can announce the routes received from one of its customers toward its other customers, peers, and providers. However, an AS is not allowed to announce the routes received from one of its peers to its other peers. An AS prefers a route received from its customers over those received from its peers over those received from its providers. Enforcing a valley-free routing policy is crucial for guaranteeing routing convergence in an IP address based network. This also holds for a flat label routing model.

Next, we consider IP prefix allocation for stub ASs. If a stub AS has a single provider, it is allocated a provider-allocated IP prefix by its provider. If a stub AS is multihomed, we consider two cases: (1) It is allocated a provider-allocated IP prefix by one of its providers (with respect to the routing table size, this yields a lower bound) and (2) it is allocated a provider-independent IP prefix (this yields an upper bound of the routing table size). We are interested in comparing both the upper bound

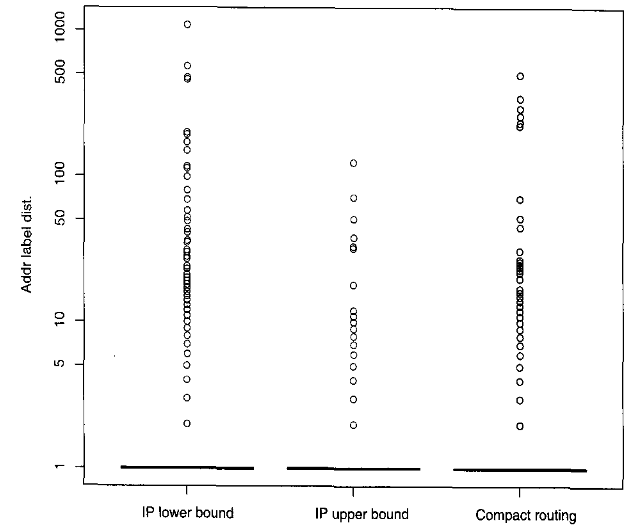


Fig. 1. Address label distribution (number of covered address label for 3000 ASs).

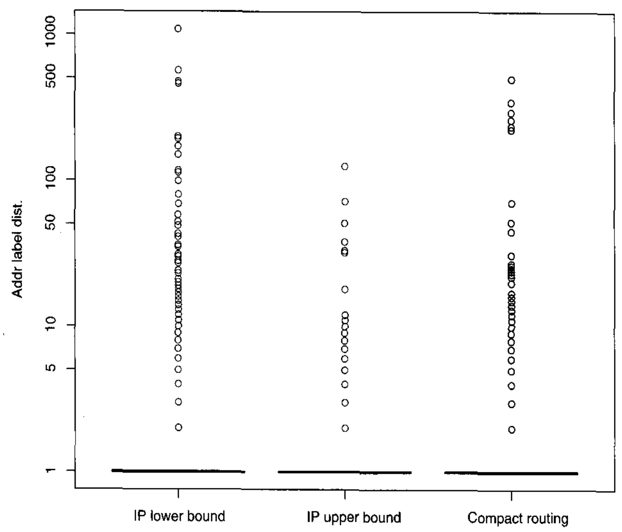


Fig. 2. Address label distribution (number of covered address labels for 2500 ASs).

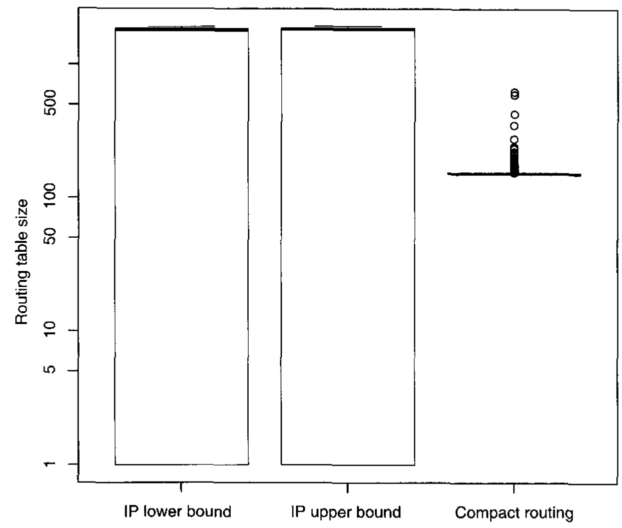


Fig. 3. Routing table size (number of routing entries for 3000 ASs).

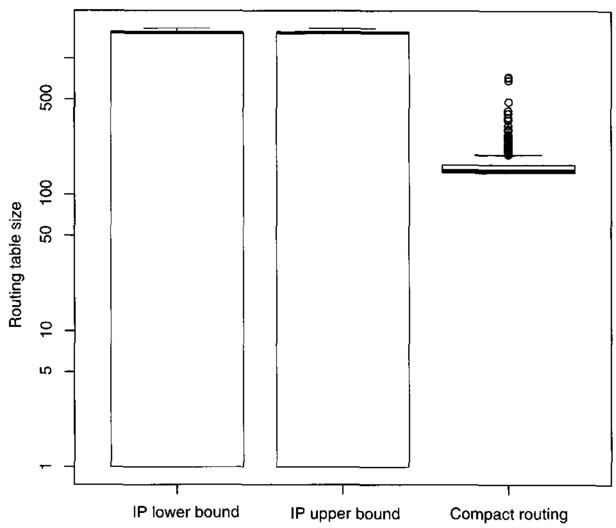


Fig. 4. Routing table size (number of routing entries for 2500 ASs).

Table 1. The address label coding theoretical limits of different routing schemes.

Routing scheme	3000 ASs	2500 ASs
IP routing (upper bound)	10.64 bits	10.36 bits
IP routing (lower bound)	8.60 bits	8.39 bits
Compact routing	9.12 bits	7.98 bits
Flat label routing	11.55 bits	11.29 bits

and the lower bound cases. From the above network topologies with embedded business relations, numerical values for (8) can be obtained.

Simulators were written from scratch in C and run on a Linux super cluster with 8 G bytes of memory. In this way, we could achieve the greatest simulation flexibility and perform simulations on relatively large Internet topologies. We believe that simulation of interdomain networks with 3000 ASs can reasonably reflect the behaviour of the real inter-domain topology. Two topologies with different size, i.e., 3000-ASs and 2500-ASs, are used to investigate the change in properties due to changes in network size for different routing schemes.

C. Some Results

Recall that $P(l)$ is the distribution of address label l. The address label distributions for IP routing and compact routing for 3000 ASs and 2500 ASs are shown in Figs. 1 and 2, respectively. By examining the experimental data, it can be found that for the given specific 2500-AS network topology, only 12%, 5%, and 2% of nodes cover more than one address label in the lower bound IP routing scheme, upper bound IP routing scheme, and compact routing scheme, respectively. For the same network topology, the maximum numbers of address labels covered by a single address label are 852, 130, and 1083 in the lower bound IP routing scheme, upper bound IP routing scheme, and compact routing scheme, respectively. For the 3000-AS topology, 12%, 5%, and 4% of nodes cover more than one address label in the lower bound IP routing scheme, upper bound IP routing scheme, and compact routing scheme, respectively. For the

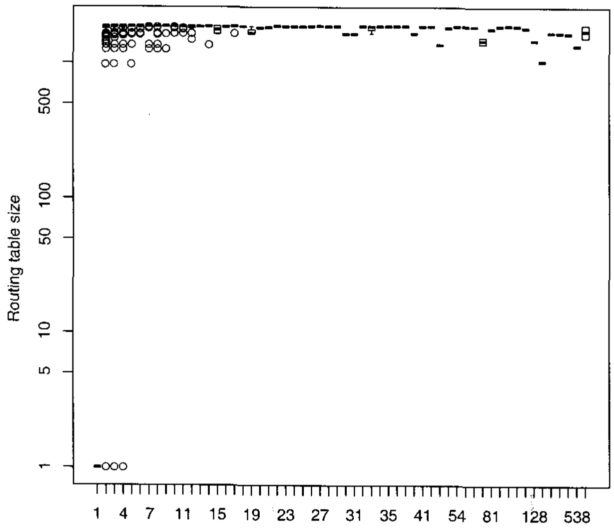


Fig. 5. IP lower bound routing table size (number of routing entries) vs. node degree in 3000-ASs topology.

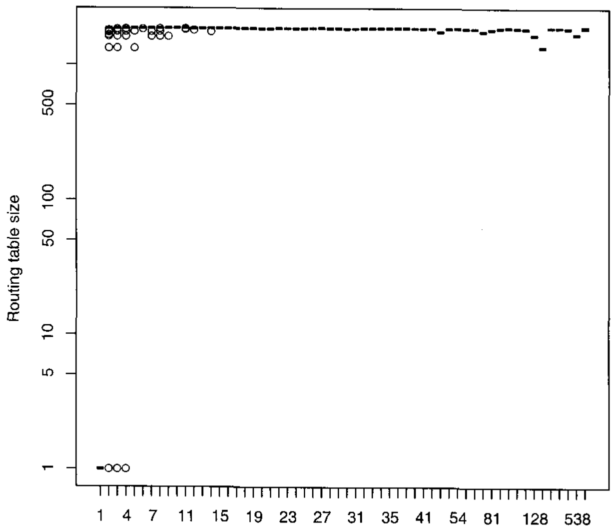


Fig. 6. IP upper bound routing table sizes (number of routing entries) vs. node degree in 3000-ASs topology.

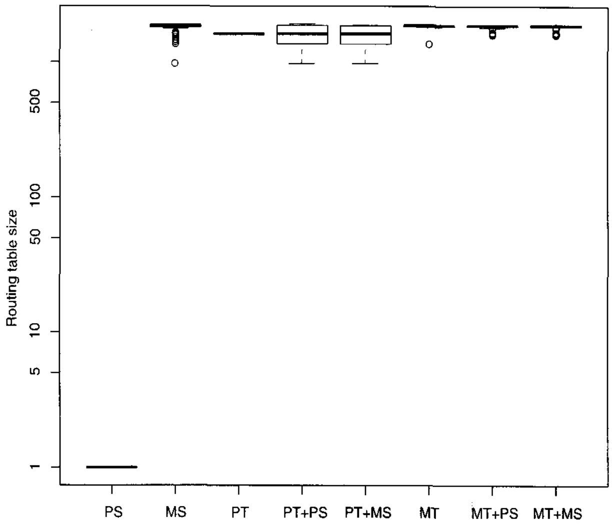


Fig. 7. IP lower bound routing table sizes (number of routing entries) vs. node role in 3000-ASs topology.

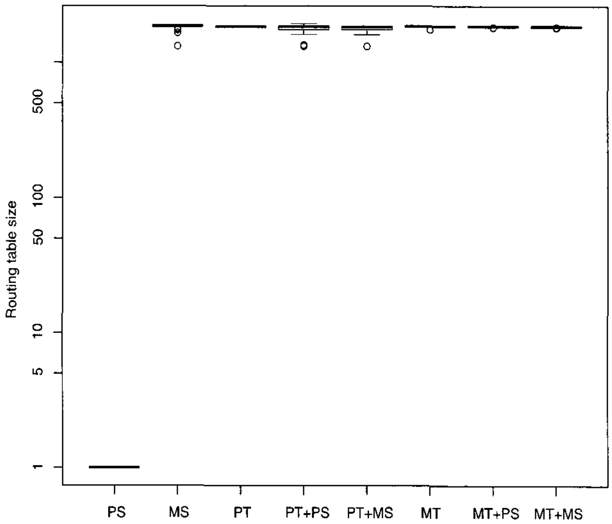


Fig. 8. IP upper bound routing table sizes (number of routing entries) vs. node role in 3000-ASs topology.

same network topology, the maximum numbers of address labels covered by a single address label are 1090, 125, and 500 in the lower bound IP routing scheme, upper bound IP routing scheme, and compact routing scheme, respectively.

Applying (8) to these address label distributions, Table 1 shows that for both the 2500-AS topology and the 3000-AS topology, the number of bits required by compact routing is less than those for the other routing schemes. It is shown that compact routing is better than IP routing, as well as, more surprisingly, flat label routing, from the address label coding efficiency point of view. The address label scheme of compact routing is the most efficient.

With the same 3000-AS and 2500-AS network topologies, the boxplots of the routing table sizes for the different routing schemes are presented in Figs. 3 and 4. It can be seen that the routing table size distribution of compact routing scheme has a smaller variance than those of the other schemes. On the other hand, approximately 33% of the nodes have just one routing entry in their IP routing tables (both lower bound and upper bounds); the rest of the nodes in the upper bound IP routing scheme have a routing table size range of [1323, 1948]. On the other hand, the routing table size ranges from 972 to 1918 in the IP routing scheme in the lower bound case. It is shown that route aggregation in the lower bound IP routing scheme is more efficient than in the upper bound IP routing schemes. In the compact routing scheme, 94.5% of nodes have routing table sizes varying from 150 to 160. The other 5.5% of nodes have different routing table sizes varying from 160 to 628.

Figs. 5 and 6 show the IP routing table sizes of nodes with different degrees in the 3000-AS topology. By comparing the figures, it can be seen that in the IP routing lower bound case, the routing table size distributions of nodes with lower degrees have larger variances than those for the IP routing upper bound case. This is because, these ASs with lower degrees are generally stub ASs or transit ASs with stub ASs attached, in the IP routing lower bound case. In this case, each stub AS always

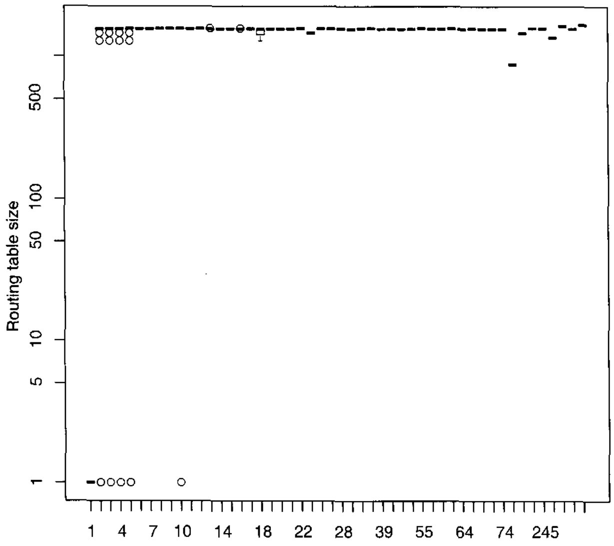


Fig. 9. IP lower bound routing table sizes (number of routing entries) vs. node role in 2500-ASs topology.

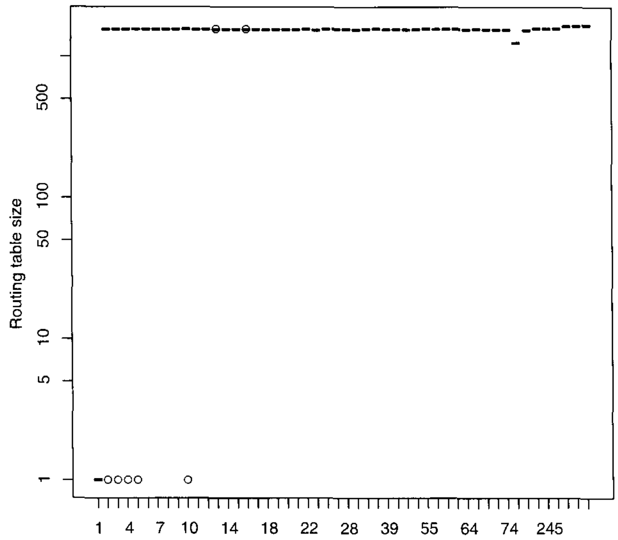


Fig. 10. IP upper bound routing table sizes (number of routing entries) vs. node role in 2500-ASs topology.

use provider allocated addresses. Hence, more IP prefixes are aggregated and these aggregates are not announced or leaked into the rest of the Internet. However, in the IP routing upper bound case, only pure stub (normally these ASs have lower degrees) or transit AS can use provider allocated addresses. Other ASs, for instance multihomed stub ASs, have to use provider independent IP addresses. These provider independent IP prefixes cannot be aggregated and they have to be announced and installed everywhere, which leads to more uniform routing table sizes.

To make the plots easier to read, abbreviations have been introduced to denote the different roles of ASs. Here, P stands for Pure, M for Multi-homed, T for Transit, and S for Stub. Combinations are used, e.g., "MS ASs" denotes multihomed stub ASs, "PT+PS ASs" represents pure transit ASs with their downstream pure stub ASs attached, and a PT+MS AS is a pure transit AS that has multihomed stub ASs attached to it.

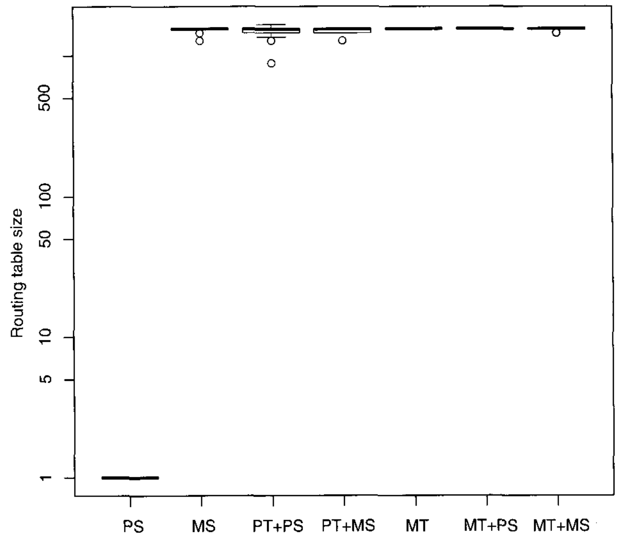


Fig. 11. IP lower bound routing table sizes (number of routing entries) vs. node role in 2500-ASs topology.

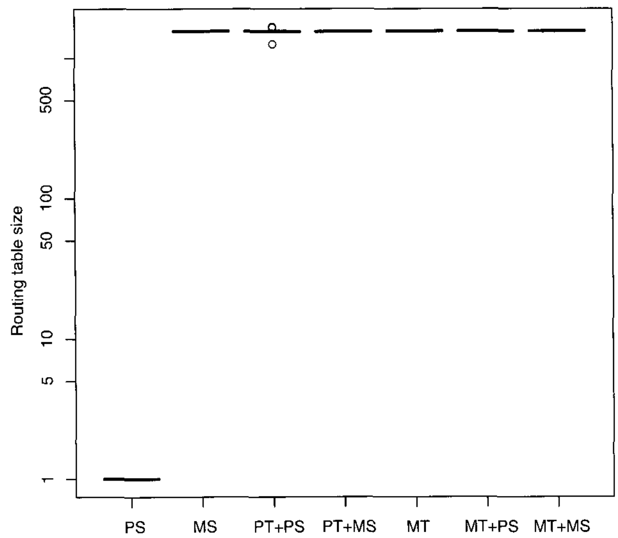


Fig. 12. IP upper bound routing table sizes (number of routing entries) vs. node role in 2500-ASs topology.

Table 2. Average routing table size.

Routing scheme	3000 ASs	2500 ASs
IP (upper bound, pure stub excluded)	1861	1460
IP (lower bound, pure stub excluded)	1445	1273
IP (upper bound, pure stub included)	975	834
IP (lower bound, pure stub included)	960	833
Compact routing	392	153
Flat label routing	3000	2500

By observing Figs. 7 and 8, it can be seen that the routing table sizes of MS ASs, PT+PS ASs, and PT+MS ASs change significantly from the IP routing lower bound case to the upper bound case. This is due to different effects of IP routing aggregation on different ASs with different roles. A pure transit ASs can suppress the announcements of IP prefixes that belong to its IP address space. Depending on how pure transit ASs partition their IP address space, their routing table sizes can vary dramatically, since those partitioned IP prefixes must be installed into their routing tables as well. In the lower bound IP routing case,

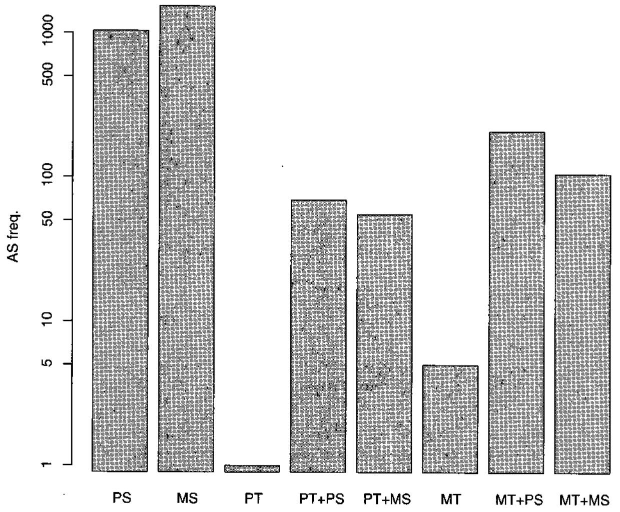


Fig. 13. Role statistics (number of AS in each categories) of 3000 ASs.

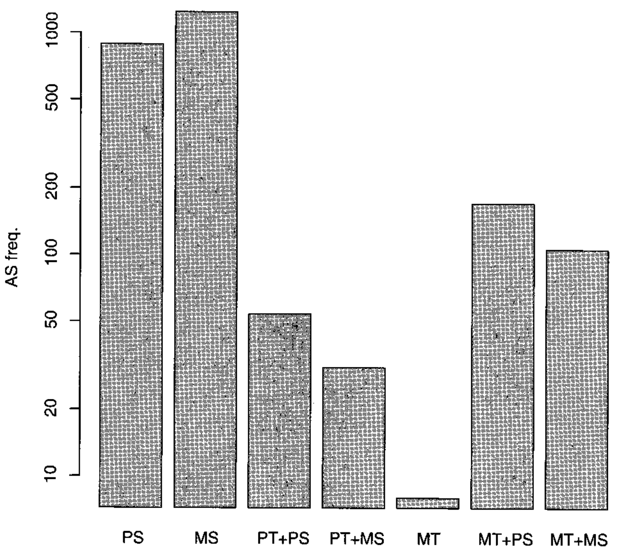


Fig. 14. Role statistics (number of AS in each categories) of 2500 ASs.

multihomed stub ASs use provider allocated IP addresses, which may also cause routing aggregation to some extent. That is, the provider from which the multihomed AS receives its IP prefix will aggregate its IP prefixes. However, this is not the case for the upper bound IP routing case. In this case, the multihomed stub ASs always use PI IP addresses which cannot be aggregated by their providers. Thus, the PI IP prefix leaks into the routing system of the whole Internet. Since multihomed stub ASs have to install the complete IP routing table in order to perform traffic engineering, the changes in their routing table sizes reflect both the effects discussed above. By checking the role distributions of different ASs in Fig. 13, it can be seen that the multihomed stub ASs or pure transit ASs with pure stub ASs or multihomed stub ASs attached make up approximately 50% of all ASs. The effect of these ASs on the routing system is more dramatic than those of the other ASs. By comparing the IP routing lower bound and upper bound cases in Figs. 7 and 8, it can be seen that whether multihomed ASs use provider allocated IP addresses has no significant impact on the overall performance of the IP routing systems. This is because, although multihomed stub ASs use PA

Table 3. Values of α in (4) for different routing schemes.

Routing scheme	3000 ASs	2500 ASs
IP routing (upper bound)	0.95	0.93
IP routing (lower bound)	0.94	0.93
Compact routing	0.80	0.82
Flat label routing	1.00	1.00

IP addresses, the routing aggregation only happens on the ASs from which the multihomed stub ASs receive their IP prefixes. The IP prefix used by the multihomed stub AS is still announced to another connected transit AS and further to the rest of the Internet. In current IP routing practice, every AS, except for pure stub ASs, still installs an IP prefix announced by a multihomed stub AS. This fact can be confirmed in Table. 2.

Figs. 9–12 and 14 illustrate the properties of different routing schemes for a 2500-AS topology. It is obvious that the above discussions about the properties of different routing schemes on the 3000-AS topology is also valid for the 2500-AS topology.

The different α values in Table 3 are calculated based on (4). It is again confirmed that the compact routing scheme is more scalable than IP solutions.

VII. CONCLUSIONS

This paper investigated fundamental issues of IP routing architectures, i.e., the properties of identifiers and address labels and their impacts on routing scalability in different routing schemes. In order to discuss these issues, a simple Internet routing model was developed and it was shown that a binary relation T defined on the address label set A determines the cardinality of the compact label set L. A basic conclusion is that there is no scalable routing scheme based on flat address labels. Equivalently, to be scalable, any routing scheme must be based on structured address labels. This implies that routing scalability and routing stability are inherently related and must be considered together when a routing scheme is evaluated.

A metric was defined to measure the address label coding efficiencies of different routing schemes. Simulations showed that the address label coding efficiency in the compact routing scheme is 16% higher than in IP upper bound case. In addition, the IP routing scheme has worse performance than the compact routing scheme in terms of routing table sizes, as well. Another discovery is that making multihomed stub AS use PA IP addresses does not reduce the global routing table sizes of the IP routing system significantly.

Further research efforts should focus on compact routing schemes with the necessary routing stability.

REFERENCES

[1] D. Meyer, L. Zhang, and K. Fall, Report from the IAB workshop on routing and addressing, Internet Engineering Task Force, 2007.

[2] D. Farinacci, V. Fuller, D. Oran, D. Meyer, and S. Brin, "Locator/ID separation protocol", IETF Trust, Oct. 10, 2008.

[3] C. Vogt, "Six/one router: A scalable and backwards compatible solution for provider-independent addressing," in *Proc. MobiArch*, New York, NY, USA, 2008, pp. 13–18.

[4] L. Zhang, "An overview of multihoming and open issues in gse," *IETF Journal*, Dec. 11, 2006.

[5] X. Zhang, P. Francis, J. Wang, and K. Yoshida, "Scaling ip routing with the core router-integrated overlay," in *Proc. IEEE ICNP*, Washington, DC, USA, 2006, pp. 147–156.

[6] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the internet topology," in *Proc. SIGCOMM*, New York, NY, USA, 1999, pp. 251–262.

[7] D. Krioukov, K. Claffy, K. Fall, and A. Brady, "On compact routing for the internet," in *Proc. ACM SIGCOMM Comput. Commun, Rev.*, vol. 37, no. 3, 2007, pp. 41–52.

[8] A. Brady and L. Cowen, "Compact routing on power law graphs with additive stretch," in *Proc. ALENEX*, 2006.

[9] I. Norros, "Powernet: Compact routing on internet-like random networks," in *Proc. NGI*, July 2009, pp. 1–8.

[10] H. Ma, B. E. Helvik, and O. J. Wittner, "A compact routing scheme with lower stretch," in *Proc. NTMS*, Dec. 2009, pp. 1–5.

[11] I. Abraham, C. Gavoille, D. Malkhi, N. Nisan, and M. Thorup, "Compact name-independent routing with minimum stretch," in *Proc. SPAA*, New York, NY, USA, 2004, pp. 20–24.

[12] H. Ma, O. J Wittner, and B. E Helvik, "A stochastic clustering algorithm for swarm compact routing," in *Proc. NGI*, Aveiro, Portugal, July 2009.

[13] L. J. Cowen. "Compact routing with minimum stretch," in *Proc. SODA*, Philadelphia, PA, USA, 1999, pp. 255–260.

[14] M. Thorup and U. Zwick, *Compact Routing Schemes*. ACM Press, pp. 1–10, 2001.

[15] R. Winter. "Two steps towards practical compact routing," in *Proc. ITC*, Paris, France, Sept. 2009, pp. 1–8.

[16] H. Ma, B. E. Helvik, and O. J. Wittner, "The stability of compact routing in dynamic inter-domain networks," in *Proc. CTRQ*, June 2010, pp. 1–5.

[17] X. Dimitropoulos and G. Riley, "Modeling autonomous-system relationships," in *Proc. PADS*, Washington, DC, USA, 2006, pp. 143–149.

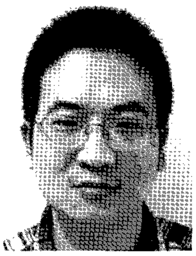
[18] C. Jin, Q. Chen, and S. Jamin, "Inet: Internet topology generator," University of Michigan at Ann Arbor, Tech. Rep. CSE–TR–433–00, 2000.

[19] X. Dimitropoulos, D. Krioukov, A. Vahdat, and G. Riley, "Graph annotations in modeling complex network topologies," *ACM Trans. Model, Comput. Simul. (TOMACS)*, vol. 19, no. 4, 2009.

[20] X. Dimitropoulos, D. Krioukov, M. Fomenkov, B. Huffaker, Y. Hyun, K. claffy, and G. Riley, "AS relationships: Inference and validation," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 1, pp. 29–40, 2007.

[21] X. Dimitropoulos, D. Krioukov, B. Huffaker, K. Claffy, and G. Riley. "Inferring AS relationships: Dead end or lively beginning?" in *Proc. 4th Workshop on Efficient and Experimental Algorithms*, Santorini,Greece, Jan. 2005, pp. 113–125.

[22] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan "Chord: A scalable peer-to-peer lookup service for Internet applications," in *Proc. ACM SIGCOMM*, San Diego, CA, USA, Aug. 2001.



Huaiyuan Ma has been a Ph.D. student in Q2S Research Centre, NTNU, Norway since Oct. 2007. His major research interests are telecommunication theory and technologies, especially IP technology. From Oct. 2005 to Aug. 2007, he worked as an IP expert in Huawei Technologies Co., Ltd. He received his Siv.ing. degree (MSc in technology) from Chalmers University of Technology, Gothenburg, Sweden in 2005. He held positions as software engineer, test engineer and quality control engineer in the Hebei Semiconductor Research Institute, China from 1995 until 2003.

Bjarne E. Helvik received his Siv.ing. degree (M.Sc. in technology) from the Norwegian Institute of Technology (NTH), Trondheim, Norway in 1975, and was awarded the degree Dr. Techn. from the NTH in 1982. He is currently Professor at the Norwegian University of Science and Technology (NTNU), the Department of Telematics, and is one of the founders of and principal academic at the Norwegian Centre of Excellence (CoE) for Quantifiable Quality of Service in Communication systems (Q2S). Since 2009, he has been Vice Dean with responsibility for research at the Faculty of Information Technology, Mathematics and Electrical Engineering at NTNU. His research interests includes QoS, dependability evaluation, fault-tolerant computing systems and survivable networks as well as related communication system architectural issues. He has published widely in these areas. He is active in international collaborations and on a number of scientific technical program committees.

Otto J. Wittner received his B.Eng. in Information Systems Engineering from University of Surrey, UK, in 1993 and his Ph.D. in Network Management from the Norwegian University of Science and Technology in 2003. Between and during studies he held positions as system programmer (ICL and Alcatel) and research fellow (NTNU). In 2003, he joined the Q2S centre of excellence at the same university as a postdoc. Currently, he holds a research position in the Innovation and Development department of UNINETT, the Norwegian research network operator. His main research interests are dependable stochastic routing systems, traffic analysis, and multimedia transport and management.