http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (1): 1-14, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.1.14

A Multipath Routing Protocol over Chord-based Internet Indirection Infrastructure (I3)

Sam Banani and Komwut Wipusitwarakun
Sirindhorn International Institute of Technology, Thammasat University,
131 Moo 5, Tiwanont Road, Bangkadi Muang, Pathum Thani 12000, Thailand

Abstract: One of architectural proposals for solving the "IP overloading problem" by splitting IP's "identifier" function out of "locator" function is the Internet Indirection Infrastructure (I3). In I3, a host indirectly communicates with the other by sending data packet, specified by destination's ID instead of the IP address. Although, the I3 architecture overcomes several limitations of the original IP, it suffers in higher performance overhead incurring from the rendezvous place's overlay layer, especially for real-time applications. This study proposed a solution to increase the I3 performance by utilizing more available resources in the Internet using multipath transmission approach. The proposed scheme uses the stacking and redirecting function of the I3 to intercept the data packets injected at the sender local server. Then, those data packets will be distributed through several possible paths toward the receiver local server where the data will be forwarded directly to the receiver. Both local servers will cooperate together to dynamically search for available paths via I3 servers according to the network condition on the Internet. Based on simulations, the results suggest that the proposed scheme can significantly improve the I3 performance even in environments with mobile hosts.

Key words: Internet indirection infrastructure, overlay network, next generation Internet, multipath routing, rendezvous-based communication

INTRODUCTION

The original Internet is designed to use its IP address as both "identifier" function and "locator" function in which identifier (id) explains "who" the host is and locator (loc) uniquely defines "where" the host is. This IP function overloading has been recently recognized as one of the key architectural problems which exposes several limitations including in seamless mobility, security, multi-homing, reduced address space and so on (Jian, 2006). The Internet Activity Board (IAB) workshop on routing and addressing (Meyer *et al*., 2007) has a consensus that decoupling the dual function of IP addresses should be explored and experimented for the long-term design of the Next Generation Internet (NGI).

To address such id/loc decoupling design principle, several architectural solutions, Moskowitz *et al*. (2006), Nordmark and Bagnulo (2009), Farinacci *et al*. (2012), O'Dell (1997), Pan *et al*. (2008) and Stoica *et al*. (2004) have been proposed in the literatures. The Internet indirection infrastructure (I3) (Stoica *et al*., 2004) is one attractive practical solution utilizing the application-layer overlaying technique to realize such id/loc decoupling on the top of the current public Internet. In the I3, every host which needs to be contacted by the others will put its contact information, called the "trigger" in this I3 proposal, in the rendezvous place. The "trigger" mainly consists of the host's identifiers (in any arbitrary format) accompanied with the host's current locator (IP address). The rendezvous place is implemented by an Internet-based overlay network of several I3 servers storing hosts' triggers in a distributed manner. Unlike the other overlay network for general content sharing P2P applications (Fuke and Yuhong, 2012; Androutsellis-Theotokis and Spinellis, 2004), those I3 server nodes will not be used only as a distributed database for content looking up service but it is also used to relay the application data as well. A sender host sends data packets to a receiver host via one of its reachable I3 server in the rendezvous place, by specifying the receiver's identifier instead of the receiver's locator (i.e., IP address). The data will be routed through series of I3 servers in the rendezvous place until it reaches the I3 server responsible for storing the receiver's trigger. Then, that I3 server will finally forward the data to the IP address correspondent to the specified identifier. Note that both triggers and data packets are distributed and routed through I3 servers in a scalable manner using Chord-based routing protocol (Stoica *et al*., 2001). Figure 1 shows the I3 infrastructure implemented by I3 overlay network on the top of the public Internet.

Corresponding Author: Sam Banani, Sirindhorn International Institute of Technology, Thammasat University, 131 Moo 5, Tiwanont Road, Bangkadi Muang, Pathum Thani 12000, Thailand

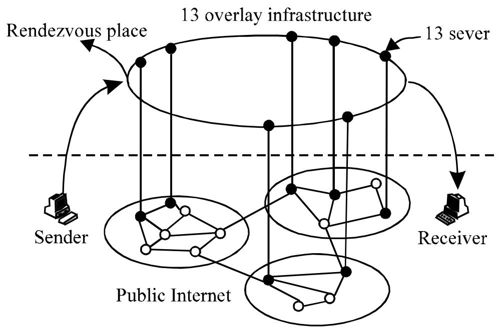


Fig. 1: I3 infrastructure implemented by I3 overlay network on the top of public Internet

By applying appropriate trigger format and dynamic trigger updating scheme, several aforementioned limitations of the original Internet architecture such as seamless mobility can be overcome in the I3. However, one key drawback of using rendezvous place's overlay layer in I3 is its higher performance overhead which leads to quite low performance in data transmission. Stoica *et al.* (2004), reported that the end-to-end delay may increase up to 6-10 times of the normal direct IP layer communication.

One possible solution to increase the I3 performance is by utilizing more transmission paths in the Internet. Since the I3 is implemented by overlaying techniques (Murase *et al*., 2006) which can bypass the default routing mechanism of the original Internet and there are several paths available between a pair of hosts in the Internet, the multipath transmission over I3 infrastructure (Banani and Wipusitwarakun, 2011) is possible.

In the Internet literature, there are several proposals addressing multipath routing approach. Banner and Orda (2007) and Sohn *et al.* (2006) uses multipath routing to avoid congestion while Gurtov and Polishchuk (2009) and Ishida and Yakoh (2008) use multipath to improve security and reliability in the Internet. However, there is no multipath routing proposal in the I3 environment.

This study aims to propose a multipath routing protocol in I3 network which allows a sender host to effectively find and utilize several available transmission paths toward the receiver host as fast as possible. In addition, the proposed protocol should still be compatible with the original I3 routing protocol and should achieve at least the same level of seamless mobility provided by the original I3.

I3 ARCHITECTURE

In I3, all communications take place through a rendezvous place implemented by a group of I3 servers as in Fig. 1. Those I3 servers function as a distributed system to maintain the I3 “trigger”, which is the mapping info from the hosts’ ID to the hosts address (i.e., the hosts’ IP+port number), in a scalable manner. Hosts which need to be contacted by the others will send their own triggers into the rendezvous place, via one of their reachable I3 server. The trigger will be routed through series of I3 servers until it reaches to a suitable I3 server where it will be stored at the moment. Triggers are dynamically distributed among I3 servers using P2P protocols, such as Chord (Stoica *et al*., 2001), Tapestry (Zhao *et al*., 2004), Pastry (Rowstron and Druschel, 2001), etc. When a sender host wants to send data to a receiver host, the I3 “data packet” comprising of the receiver’s ID and data is composed and is forwarded to one of its reachable I3 server in the rendezvous place. The I3 data packet will be routed, in the same manner as of I3 trigger, through series of I3 servers in the rendezvous place according to the implemented P2P protocol until it reaches the I3 server responsible for storing the receiver's trigger. Then, that I3 server will finally forward the data to the receiver’s address (i.e. IP+port number) correspondent to the specified ID in the I3 data packet. In I3, the sender host does not need to know the real address of the receiver. If the receiver changes point of attachment (address), the receiver only needs to update its trigger with the new address in the rendezvous place. The incoming I3 packets can still be forwarded to the new address automatically. Figure 2 shows data transmission model in I3 overlay network. In the figure, the receiver sends its trigger with its host ID “t_{id1}” and its address “R” into the rendezvous place, the trigger is routed among I3 servers based on the implemented P2P protocol and finally stored at the responsible I3 server “I”. When a sender wants to send data to the receiver, it will put the receiver’s ID in the I3 packet and send to one of reachable I3 server. The I3 server tries to match the packet’s ID with its own stored triggers. If it is applicable, the matched trigger will be used. Otherwise, it will be forwarded to the next I3 server until reaching to the server “I” where the ID in the I3 packet and the ID in the stored trigger is matched. Then, the address “R” of the matched record will be used to send the I3 packet directly to the receiver. Note that, the responsible I3 server “I” determination and forwarding decision are dependent to the selected P2P protocol. In additions, the real I3 packet will also contain the sender’s ID in order to let the receiver reply back correctly.

ID/address stacking and redirection function: One of interesting feature in I3 is its ID/Address (ID/ADS) stacking and redirection function of I3 trigger and I3 data packet. Figure 3 shows a general format of I3 trigger and I3 packet.

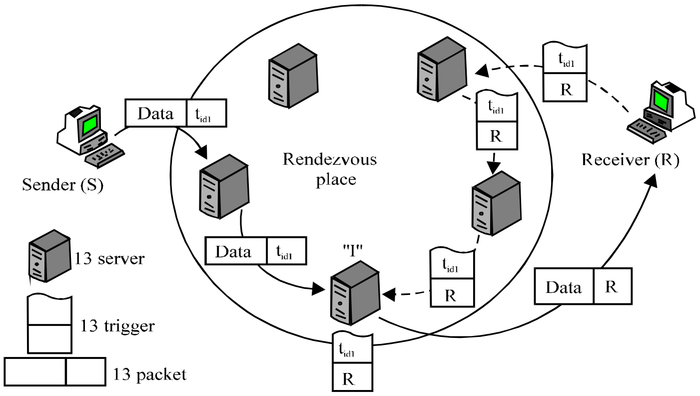


Fig. 2: I3 data transmission model. The ID and the address of the receiver are t_{id1} and R, respectively

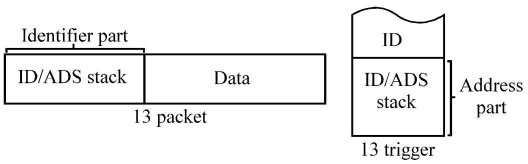


Fig. 3: I3 packet and trigger general format

In I3, it is allowed to replace the "address part" in I3 trigger and the "identifier part" in I3 packet with a stack of identifiers (ID/ADS) whose type is either I3 host's ID or host's address (IP+port number). At any I3 server, the most outer identifier will be considered. If it is of address type, the I3 packet will be sent directly to the host by the address. Otherwise, it will be matched with the ID of triggers stored in the server. If the matched trigger is found, the stack of identifiers of the trigger will be used to replace the considered identifier, i.e., "the most outer" in the I3 data packet, then the new most outer identifier will be considered again. However, if the matched trigger is not found, the I3 packet will be forwarded to the "next" I3 server based on the implemented P2P protocol. Note that, if the mentioned "next" is the I3 server itself according to the P2P protocol, the current most-outer identifier will be popped out of the I3 packet. If there is identifier remaining in the I3 packet, it will be processed. Otherwise, the I3 packet will be dropped. Figure 4 shows an example of I3 packet processing with identifier stacking function.

The proposed multipath transmission utilizes the stacking function to redirect I3 packets through series of intermediate I3 servers as desired.

Chord based trigger/data packet routing process: To reach the responsible I3 server, both I3 trigger and data packet are routed through the rendezvous place based on the specified trigger's identifier. The determination for the responsible I3 server at a specific moment is controlled by the Chord (Stoica *et al*., 2001) protocol which provides following two basic functions:

- **Common identifier (cid) mapping:** Each trigger's identifier and each I3 server will be mapped to the common space of m-bit identifier, called common identifier circle, as in Eq. 1 and 2, respectively. The responsible I3 server of a trigger with common identifier, cid_t, will be the first server whose common identifier, cid_s, is equal to or clockwise follows the cid_t in the identifier circle. This is called the successor of cid_t, denoted as successor (cid_t). As an example shown in Fig. 5a, there are three I3 servers at the moment. Each I3 server will be responsible for cid_t which is in range of (preceding server's cid_s, its cid_s]. Thus, the responsible (successor) server of $t_{id}^{(1)}$ with $cid_t^{(1)}$ is server "S2".

$$cid_t = hash\_function(t_{id}) \quad (1)$$

$$cid_s = hash\_function\ (server\ specific\ info:\ IP+port) \quad (2)$$

- **Trigger/data I3 packet routing:** Each I3 server will maintain routing information which is used to accelerate forwarding process of I3 packets toward the packet's responsible I3 server. Such info comprises (i) range of its responsible cid and (ii) m-pointers to m successors of the specific value of cid on the identifier circle, called m-entries finger table. The ith entry (finger) in the table at the certain I3 server with common identifier, cid_s, contains the address of I3 server that succeeds cid_s by at least 2^{i-1} on the common identifier circle, i.e., successor (cid_t+2^{i-1}) as shown in Fig. 5b. If the arrival trigger/data's cid matches to server's range of

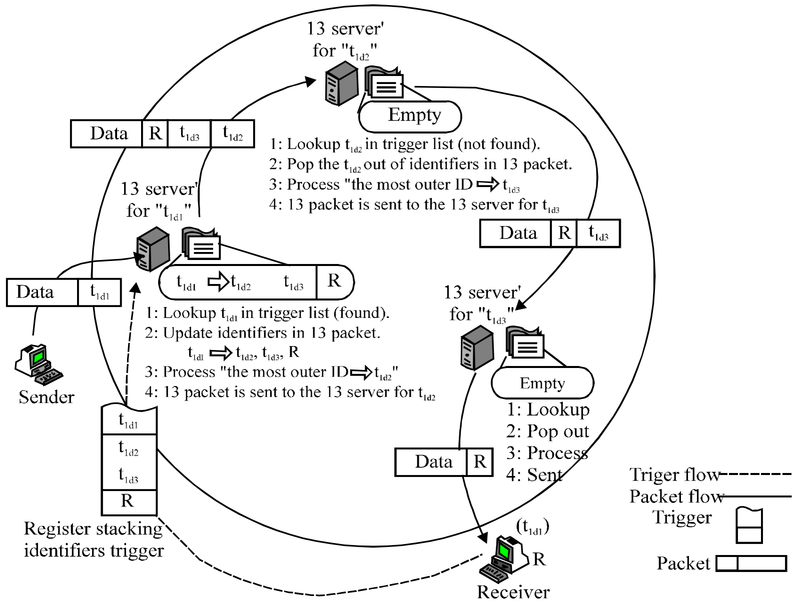


Fig. 4: I3 packet processing with identifier stacking function

responsible cid (i), the server will accept the trigger/data. Otherwise the entries in finger table (ii) will be considered in order to find the next-hop I3 server, starting from the last (mth) entry to the first (1st) entry. For any specific ith entry, the address value in the "responsible server" field will be used to forward the I3 packet if the arrival trigger/data's cid is greater or equal to the "pointed cid" field of the considered finger. Figure 5c shows an example of I3 overlay which uses 4-bits identifier and consists of five I3 servers, “S1”-“S5” with correctly configured routing information at the moment. A host sends its trigger with $t_{id}^{(1)}$ into I3 overlay via the reachable I3 server “S1”. At “S1”, the calculated cid of $t_{id}^{(1)}$ which is $cid_t^{(1)} = 7$ is compared with the server's responsible cid range (15, 2]. Since it does not matched, entries in the finger table are considered. In this example the 3rd finger is applicable to the $cid_t^{(1)}$.

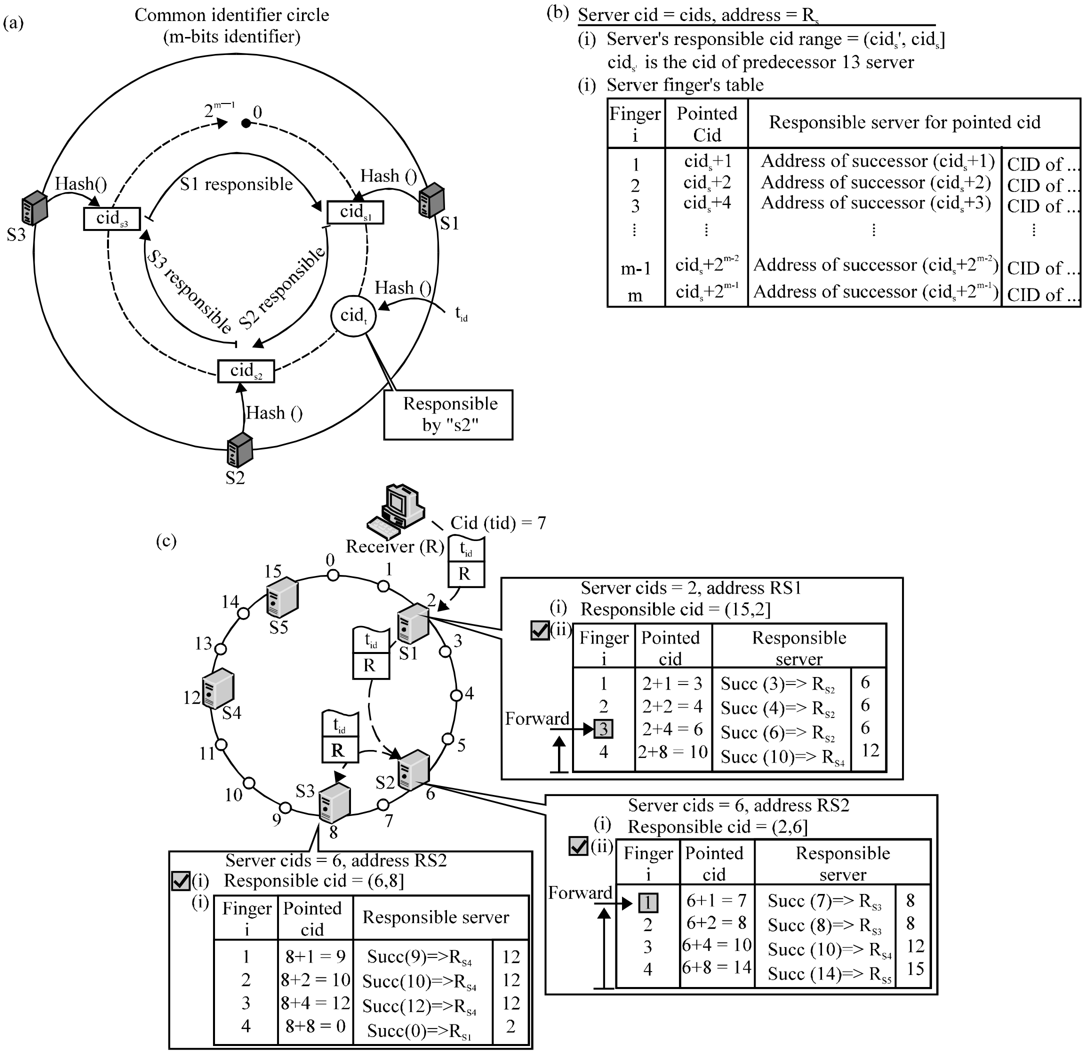
Thus, the address R_{S2} is used to forward the trigger to the next I3 server which is “S2”. At “S2”, the same routing process is repeated. The 1st finger with R_{S3} is applicable to the $cid_t^{(1)}$. Finally, the trigger is accepted at the server “S3” since the trigger $cid_t^{(1)}$ matched to the “S3”'s responsible cid range (6, 8].

Note that the control protocol in Chord (Stoica *et al*., 2001) will dynamically update I3 servers' finger tables and reassign (if appropriate) stored triggers to other (more suitable) I3 server when an I3 server joins or leaves the rendezvous overlay network.

THE PROPOSED MULTIPATH ROUTING PROTOCOL

Since the I3’s rendezvous place is natively implemented by overlay layer on the top of public Internet, I3 data packets are forwarded according to I3’s ID through rendezvous place, bypassing the default routing mechanism of underlying IP. By employing the proposed transmission model, it is possible to increase application performance and reliability by utilizing more network resources, i.e., several available paths through the rendezvous place, for sending I3 data packets.

Multipath routing model: Figure 6 shows the proposed multipath routing model. Both sender host and receiver host access to the I3 via their reachable I3 server, which are Sender-side Local Server (SLS) and Receiver-side Local Server (RLS), respectively. As shown in Fig. 6a, the receiver publishes its own ID into the rendezvous place by sending the permanent trigger (T_p) containing the receiver's ID, current network address and packet distribution preference, i.e., t_{id1}, R and dsp, to the RLS. At the RLS, the necessary RLS's multipath extension info (MEI), such as cid_{RLS} and R_{RLS} will be added into the T_p before the T_p will be forwarded according to the implemented P2P protocol, e.g., Chord protocol, until reaching at the Main Server (MS) responsible for the t_{id1} where the T_p with multipath-extension info is maintained. When the sender wants to send data to



(b) Server cid = cids, address = R_s

(i) Server's responsible cid range = $(cid_s', cid_s]$

$cid_{s'}$ is the cid of predecessor 13 server

(i) Server finger's table

Finger i	Pointed Cid	Responsible server for pointed cid	
1	cid_s+1	Address of successor (cid_s+1)	CID of ...
2	cid_s+2	Address of successor (cid_s+2)	CID of ...
3	cid_s+4	Address of successor (cid_s+3)	CID of ...
⋮	⋮	⋮	⋮
m-1	cid_s+2^{m-2}	Address of successor (cid_s+2^{m-2})	CID of ...
m	cid_s+2^{m-1}	Address of successor (cid_s+2^{m-1})	CID of ...

Server cids = 2, address RS1

(i) Responsible cid = (15,2]

☑(ii)

Finger i	Pointed cid	Responsible server	
1	2+1 = 3	Succ (3)=> R_{S2}	6
2	2+2 = 4	Succ (4)=> R_{S2}	6
3	2+4 = 6	Succ (6)=> R_{S2}	6
4	2+8 = 10	Succ (10)=> R_{S4}	12

Server cids = 6, address RS2

(i) Responsible cid = (2,6]

☑(ii)

Finger i	Pointed cid	Responsible server	
1	6+1 = 7	Succ (7)=> R_{S3}	8
2	6+2 = 8	Succ (8)=> R_{S3}	8
3	6+4 = 10	Succ (10)=> R_{S4}	12
4	6+8 = 14	Succ (14)=> R_{S5}	15

Server cids = 6, address RS2

☑(i) Responsible cid = (6,8]

(i)

Finger i	Pointed cid	Responsible server	
1	8+1 = 9	Succ(9)=>R_{S4}	12
2	8+2 = 10	Succ(10)=>R_{S4}	12
3	8+4 = 12	Succ(12)=>R_{S4}	12
4	8+8 = 0	Succ(0)=>R_{S1}	2

Fig. 5(a-c): Chord based trigger/data routing process, (a) Common identifier mapping, (b) Trigger/data routing information and (c) Example of routing trigger in chord protocol

the receiver, I3 packets with receiver's ID will be injected into the rendezvous place via the SLS. The SLS will add its multipath-extension info including cid_{SLS} and R_{SLS} into the first I3 packet before forwarding the I3 packet through series of I3 servers in the same manner as the trigger forwarding until arriving at the MS. The MS will perform the following 3 processes, as shown in Fig. 6b:

- Directly forward the I3 packet to the receiver using the receiver's address (R) in the T_p
- Directly send the temporary distribution trigger (T_{td}) to the SLS using the SLS's address (R_{SLS}) in the I3 packet
- Directly send the temporary collecting trigger (T_{tc}) to the RLS using the RLS's address (R_{RLS}) in the T_p

The T_{td} contains necessary information (R, cid_{RLS}, R_{RLS} and dsp) to let SLS locate the RLS and be able to find the available paths toward the RLS or the ultimate receiver while the T_{tc} contains information (R) to let RLS forward any arrival I3 packets directly to the receiver.

At the SLS, the arrival T_{td} will signal the SLS to intercept the successive I3 packets with the specific ID (t_{id1}) and distribute them through available paths toward the receiver at the moment, instead of forwarding via the MS. The SLS will dynamically locate suitable available paths for relaying the successive I3 packets toward the RLS.

As shown in Fig. 6c, the first and second paths can be determined by information carried in the T_{td} while the other additional paths needs additional I3 server

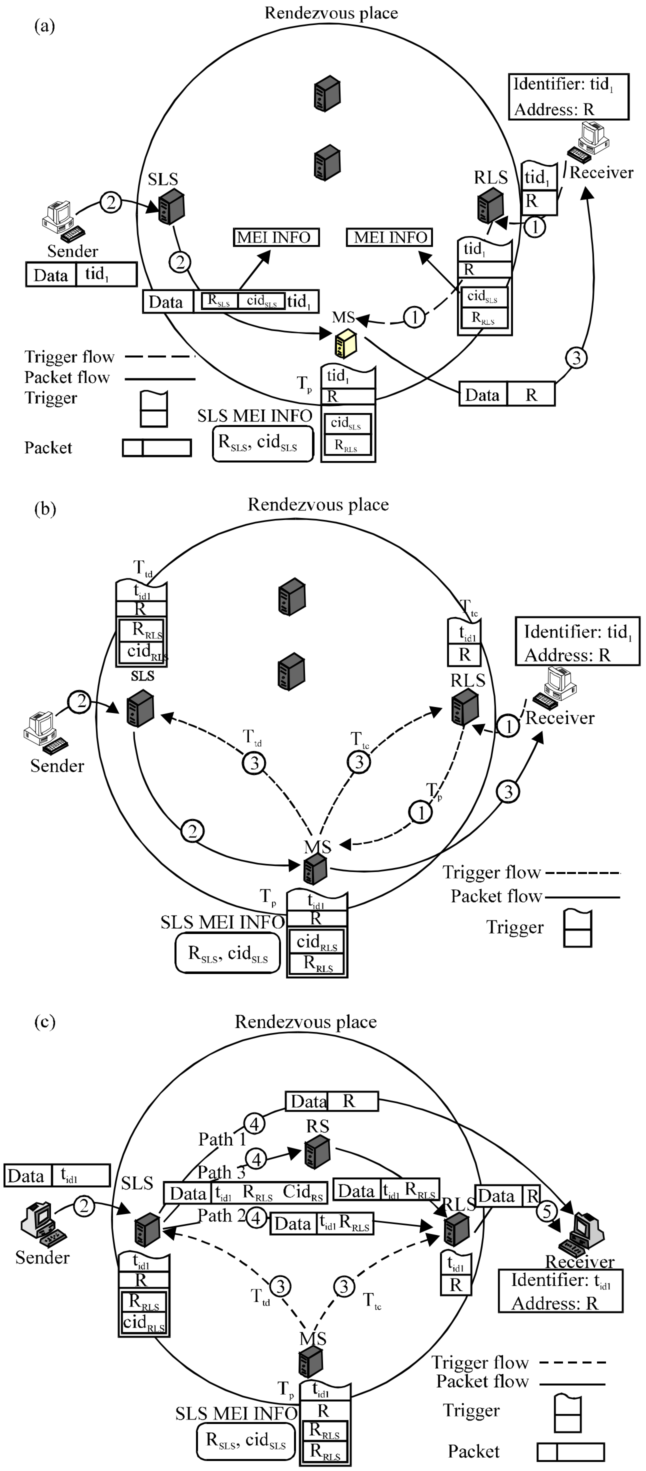


Fig. 6(a-c): Multipath routing model, (a) MEI is automatically added in the receiver's trigger and the sender's data packet by the RLS and SLS, respectively, (b) The temporary distribution and collecting triggers are sent to the SLS and the RLS when the first data packet reaches to the MS and (c) The successive data packets are intercepted by the SLS and are redirected through several alternative paths toward the RLS, then finally the receiver

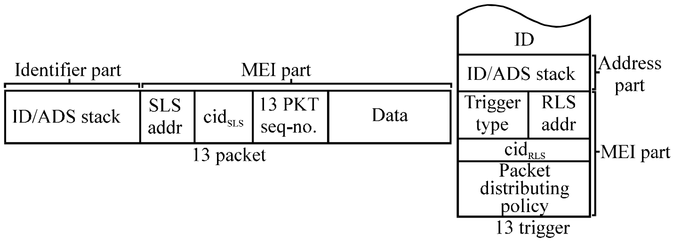


Fig. 7: I3 packet and trigger with multi-path extension info

(Relaying Server: RS) as the relaying point and can be determined by the RS server selection algorithm. It is possible that the SLS may distribute I3 packets over several available paths at the same time if the forwarding preference in the T_{td} specifies. When those I3 packets arrive at the RLS, they will be forwarded directly to the receiver address (R) in the T_{tc}.

Triggers and I3 packets with multipath extension info: Multipath Extension Info (MEI) is added into triggers and I3 data packets in order to facilitate the multipath routing scheme. Figure 7 shows the I3 packet and trigger with multipath extension information. The MEI is added as the last part in the identifier stack of both I3 packet and trigger as necessary. In I3 packet, the MEI is used to carry necessary information such as SLS's address, SLS's cid, I3 packets' sequence number etc. In trigger, the MEI is used to carry information such as trigger type, RLS's address, RLS's cid, packet distributing preference (dsp) and receiver's address (optional). The addresses of SLS and RLS are used by the MS to update the T_{td} and T_{tc} while the cid of SLS and RLS are used in the process of systematic RS selection described. The I3 packets' sequence number is used to detect the duplication of I3 packets arriving at the RLS if employing x% redundant distribution.

For the packet distributing preference, it can be specified by the receiver when publishing the T_p. The distributing preference consists of the following 4 parameters.

- **Multi-path flag:** To indicate whether the I3 packets should employ the multi-path extension or not
- **Multi-path number:** To specify the maximum number of available path used in the process
- **Path minimum delay:** To specify the minimum acceptable delay of the available path
- **Redundancy percentage:** To specify the redundant transmission percentage. For example, 0% means that an I3 packet will be sent over one available path at a moment in the round-robin fashion while 100% means that an I3 packet will be copies and sent over all available paths at a moment.

RS selection algorithm: As described earlier, the SLS determines available alternative paths toward the ultimate receiver. The first (one hop: directly to the receiver) and second (two hops: to the RLS then the receiver) paths are the shortcuts, determined by the MEI info exchanged using the proposed multipath routing protocol. The third type of available path is the diverse paths which are needed to relay through the suitable I3 server (RS), then to the RLS. This paper proposes the following two algorithms to select the RS servers:

- **Random RS selection (RRS):** The algorithm chooses the RS server by randomly selecting the I3 server ID which is not the SLS and the RLS. This approach is very simple and fast but may lead to the paths with high number of hops from the SLS to the RLS. Note that, according to the normal ID-based I3 routing process, several hops may be required to send an I3 packet from an I3 server (I_A) to the other I3 server (I_B) if the I_B's address is not maintained at the I_A. However, it depends on the employed trigger distribution P2P protocol which will let the I_A find the way to the I_B in distributed manner. For an example, the Chord protocol uses the finger table to find the way and route the I3 packet to the specified server's ID with guarantee for the worst number of hop at O(log (N)) where N is number of I3 servers in the system
- **Chord-based RS selection (CRS):** In this method, the algorithm chooses the suitable RS in order to minimize the number of hops, or other performance metrics, from the SLS to the RLS as fast as possible by utilizing more specific information from the employed Chord trigger/packet distribution/routing protocol. Details about the Chord-based routing process using finger table can be found in (Stoica *et al*., 2004)

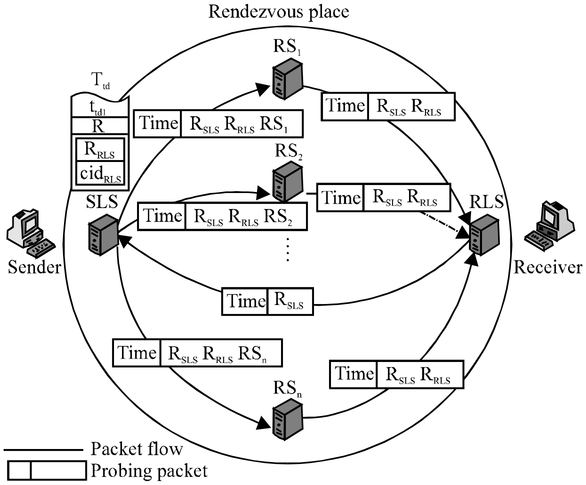


Fig. 8: SLS sends performance probing messages to RLS via RSs. Then, RLS sends back the performance feedback packet back to SLS

At the SLS, such RS candidates can be found by selecting any I3 servers which are pointed by entries in the SLS's finger table since those RS candidates require only one hop from SLS to RS. Then, probing messages will be sent through those RS candidates toward RLS as shown in Fig. 8. The RLS will collect probing messages and send back performance feedback of those paths back to the SLS. At the SLS, those feedbacks will be used to determine the best RS which is suitable to the application requirement.

Mobility support: The proposed multipath routing protocol can react correctly when the receiver and the sender change their point of network attachment (moves). Figure 9 shows the mobility support in cases of the receiver, 9a and the sender, 9b, movement as follows:

- **Receiver mobility:** As in Fig. 9a, when the receiver changes point of attachment to the Internet, it will be assigned new IP address and probably access to the new RLS. The updated permanent trigger T_p with the same ID, t_{id1} but updated other information such as R_2, cid_{RLS2} and R_{RLS2} will be republished and routed to the MS. Upon the updated trigger arrival, the MS will regenerate the new T_{td} and the T_{tc} for the SLS and the new RLS. At the SLS, it will search for new alternative paths toward the receiver based on the updated T_{td}. Then, those paths will be used to send the I3 packets, instead of the out-of-date paths
- **Sender mobility:** As in Fig. 9b shows, when the sender changes point of attachment to the Internet and it needs to access to the new SLS. In this case, the sending I3 packet will pass through the SLS without the T_{td}. The SLS will add its own MEI and forward that first packet accordance with the original I3 routing mechanism to the MS. At the MS, it will redirect T_{td} to the new SLS, instead of the old one. Now, the new SLS can use the arrival T_{td} to intercept the I3 packets and send them via alternative paths from itself to the RLS

PERFORMANCE EVALUATION

OPNET simulator has been used to evaluate the proposed multipath transmission method, comparing to existing systems. In simulations, I3 server and host objects have been created. Hosts inject traffic into I3 network via their nearest I3 server (local I3 server). The topology between I3 servers is logically full mesh, utilizing underlying IP network. The processing delay of I3 servers in each simulation run are uniformly assigned from the range of $[P_{min}, P_{max}]$ msec with the assumption that each node employs FIFO queue with infinite queue length. The average virtual link (across underlying IP paths) delay between I3 servers are uniformly distributed in the range of $[L_{min}, L_{max}]$ msec. The Chord is selected as the P2P protocol for distributing triggers and routing I3 packets inside I3 overlay network (rendezvous place). The traffic flows with constant packet rate, 50 pps (packet per sec), whose packet-size is exponentially distributed with mean 400 bytes has been generated between pairs of hosts with Poisson arrival rate, λ. The flow duration is on average 300 sec exponentially distributed. The following traffic transmission methods have been compared:

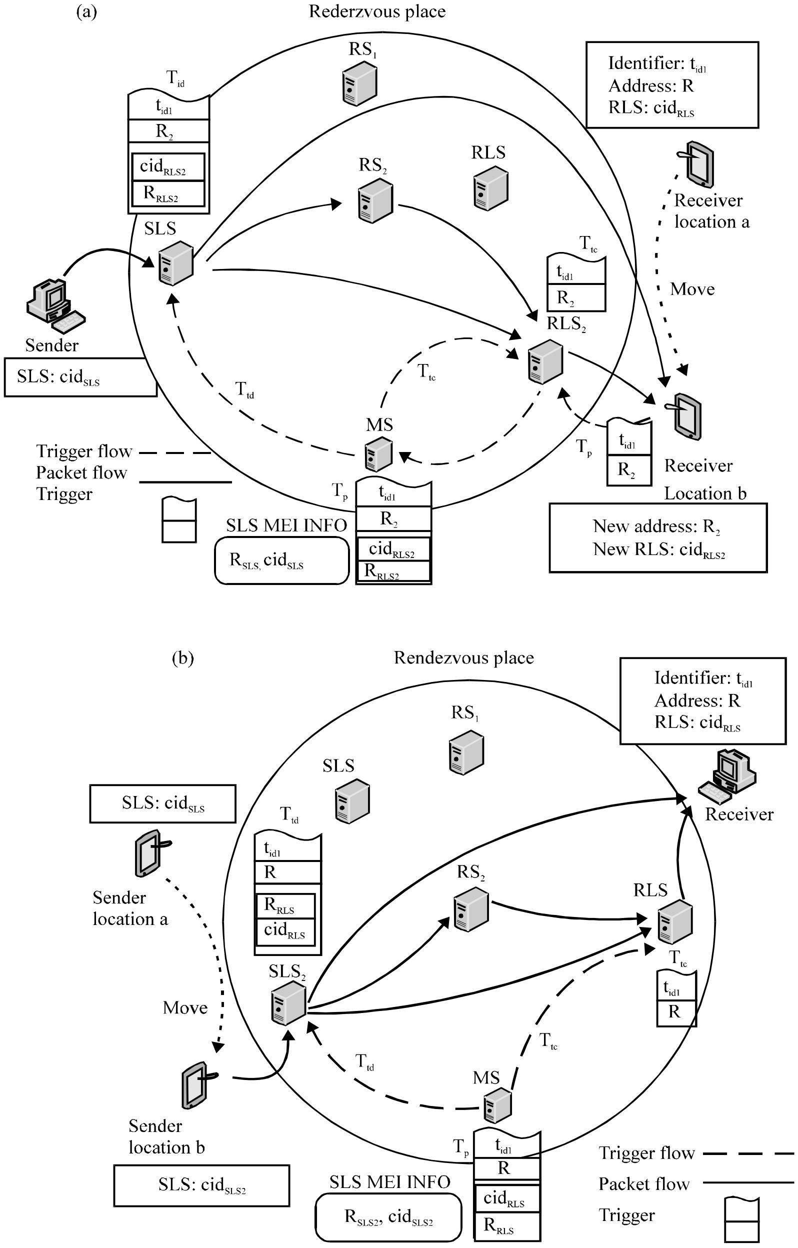


Fig. 9(a-b): Mobility support in the proposed scheme, (a) Case of receiver mobility and (b) Case of sender mobility

- Original Chord-based I3 routing protocol (Stoica *et al*., 2004)
- The proposed multipath I3 routing protocol (I3-MP_ss,dd)

In the performance analysis:

- Destination update permanent trigger T_p every 30 sec
- The maximum numbers of alternative paths are 5. T_{tc} and T_{td} are updated every 15 sec
- "ss" is relay server selection which can be either CRS or RRS
- "dd" is packet duplication ratio which is means:
- "0" means packets are distributed in round-robin fashion among multiple alternative paths
- "100" means each packet duplicated and distributed over alternative paths

Figure 10 shows the average end-to-end delay versus the offered load of I3 networks with the [5,86] msec average overlay link delay, the processing delay is on average 24 msec and 40% overlay link background traffic fluctuation factor.

The offered load is calculated as the ratio of the generated I3 packet rate over the capacity to process the I3 packet/trigger in the network, i.e.,:

$$(\lambda * n * r)\sum_{i=1}^{n}\left(\frac{1}{P_i}\right)$$

where, P_i is the processing delay of each I3 server in the network with n I3 server. Note that each plot is an average value from the results of 100 simulation runs with 95% confidence interval.

At low load, all versions of I3-MP can outperform the original I3 with better delay improvement as 37.5, 65 and 82.5% from I3-MP_RRS,0, I3-MP_CRS,0 and I3-MP_CRS,100, respectively. This is because the proposed I3-MP can find and use shorter overlay paths compared to the original one. In the proposed I3-MP, the I3 packet will be forwarded directly between the locally access I3 servers while the original I3 protocol needs to relay through the I3 server which holds the trigger of the desired destination. Additionally, since several alternative paths can be utilized in the proposed I3-MP, the I3 packets always passes through the (nearly) best paths at the moment even the background traffic fluctuates. The best improvement can be seen in the "I3-MP_CRS,100" which can find several high quality alternative path (compared to the RRS) and uses all paths for transmitting all packets.

The shorter overlay path is confirmed in the Fig. 11 which plots the effective end-to-end hop number (EH) of all I3 routing protocol in the same simulation runs while changing the offered load. The EH is defined in equation 3 as the ratio of the end-to-end delay and the time needed to process an I3 packet on one hop transmission which are the sum of I3 server processing delay and the link delay:

$$EH=\bar{D}/(\bar{L}+\bar{P}) \qquad (3)$$

where, $\bar{D}$, $\bar{L}$ and $\bar{P}$ are average end-to-end delay of I3 protocols, average link delay and average I3 server processing delay, respectively. At low load, the EH of the proposed "I3-MP_CRS,100" is as low as 1 hop while the original I3 is at about 5 hops. Approximately, about 4.63, 1.49 and 1.04 EH gain can be achieved in the proposed I3-MP protocol due to (a) the proposed common shortcut routing protocol (b) the CRS relay selection and (c) the I3 packet duplication policy, respectively.

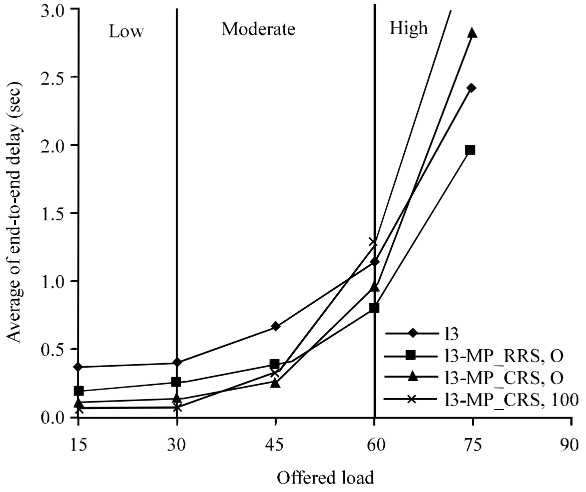


Fig. 10: Offered load versus average of end-to-end delay

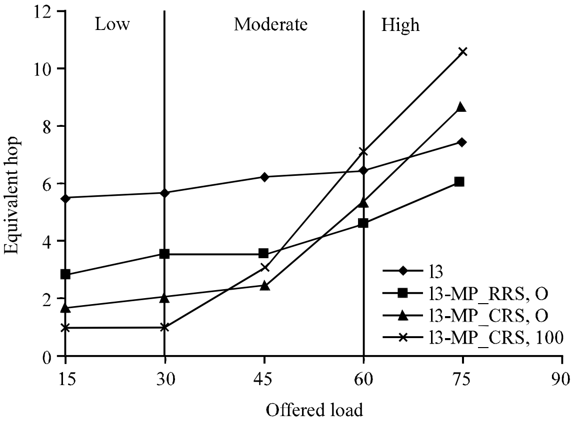


Fig. 11: Offered load versus equivalent hop

At moderate load, the proposed I3-MP still outperforms the original I3 in general. However, the performance improvement begins decreasing when the offered load increases. At the load level 45, the I3-MP_CRS,0 becomes the best while, at the higher load level 60, the I3-MP_RRS shifts to the best one. Such trend can be seen in both figure 10 and its correspondent EH value, i.e., Fig. 11.

The reason is that the I3-MP generates more control traffic compared with the original I3 in order to find the better shortcut paths, especially in the case of CRS which uses path probing messages. Moreover, traffic become doubles in case of 100% packet duplication ratio (MP_CRS,100). Those traffic highly impacts to the queuing delay of I3 servers when the load becomes higher enough. Consequently, longer time is needed to find the shortcut path and even to route each I3 packet. Thus, both end-to-end delay and the EH of the proposed I3-MP converses to that of the original I3 protocol since less I3 packets take the better shortcut path.

At high load, the performance of the I3-MP_CRS rapidly degrades since the I3-MP_CRS generates too high probing messages. Finally, only the I3-MP_RRS can outperform the original I3. However, when the load approaches 100%, the end-to-end delay of all routing protocols increases sharply to infinity due to exponential increase in I3 servers' queuing delay.

The proposed protocol achieved better performance than the original I3 especially when networks become more sparse and have higher fluctuation in background traffic. This is shown in Fig. 12 and 13 which plot the Performance Gain (PG) of the proposed "I3-MP_CRS,100" when the link delay deviation factor and the background traffic fluctuation factor 40% vary.

The Performance Gain (PG) is defined in equation 4 as how faster the proposed I3-MP can send I3 packet on average, compared with the original I3 protocol. The link delay deviation factor represents how much each overlay link average delay differs from the whole network's average overlay link delay. The higher it is the networks become more sparse and bigger. While the background traffic fluctuation factor represents how much each overlay link delay at a moment differs from the link average delay. The higher it is, the traffic of underlying IP networks become more fluctuated:

$$PG=(\overline{D}_{I3}-\overline{D}_{I3\text{-}MP\_CRS,100})/\overline{D}_{I3} \quad (4)$$

where, $\overline{D}_x$ is average end-to-end delay of I3 packets using the x protocol. The network load in this simulation runs is set at moderate level about 34%. Based on the results, the PG is increased when the value of link delay deviation factor and background traffic fluctuation factor increase. In bigger and more sparse networks, there is higher possibility that the original I3 will take longer paths while the proposed protocol still uses the shorter shortcut path. In networks with highly fluctuated traffic, the proposed protocol utilizes several paths which are decreased the possibility of using the path with high delay at the moment. The results suggest that the proposed protocol is more applicable to the big sparse public Internet with high fluctuated traffic than the original I3.

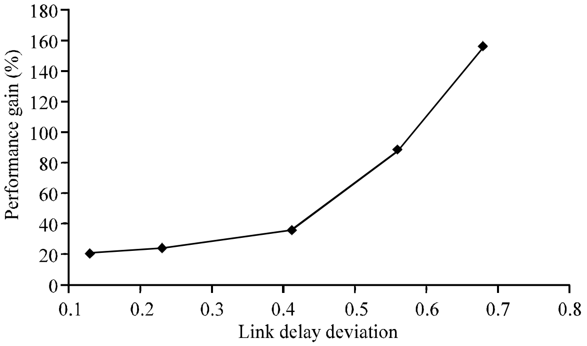


Fig. 12: Link delay deviation versus performance gain percentage

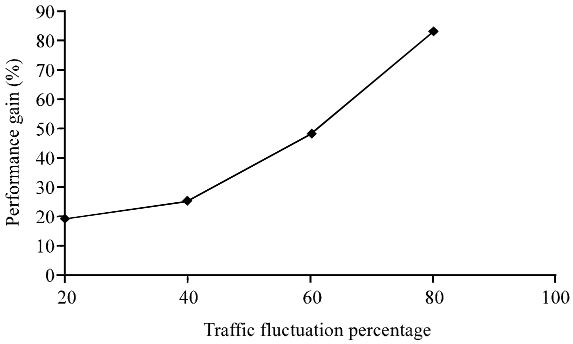


Fig. 13: Traffic fluctuation percentage versus performance gain percentage

The next simulation shows the behaviors of I3 protocol when hosts (send/receive data) changes their network attachment point. This scenario represents mobile users who can move and changing their ISP while sending/receiving data. As a result, the SLS or RLS may be changed during the communication session.

In this simulation, the network average link delay is set at [5, 86] msec and the processing delay is on average 24 msec. It is assumed that all I3 servers' locations are fixed. The Performance Gain (PG) is plotted versus the stationary index (SI) in Fig. 14. The SI is defined as the ratio of the average network attachment point changing interval for mobile hosts and the average communication flow duration. Lower SI value represents more frequent move of the hosts. In general, the proposed I3-MP performs much better than the original I3 at high SI level. The PG decreases and approaches to zero when SI decreases. Finally, the PG crosses the x-axis and becomes negative at very low SI. In situation of host movements, the proposed protocol is designed to fall back to the original I3 mechanism when the distribution or collection trigger is not found at the SLS or RLS. In the meanwhile, the proposed protocol will start looking for new shortcut paths between new SLS or RLS and will utilize them afterward. Thus, the minimum performance on that phase should be theoretically as that of the original I3. However,

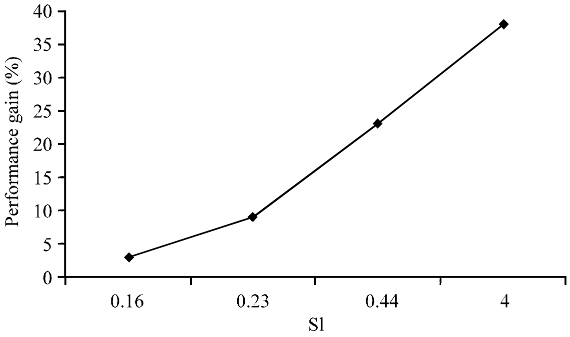


Fig. 14: Sender movement (stationary index "SI") versus performance gain percentage

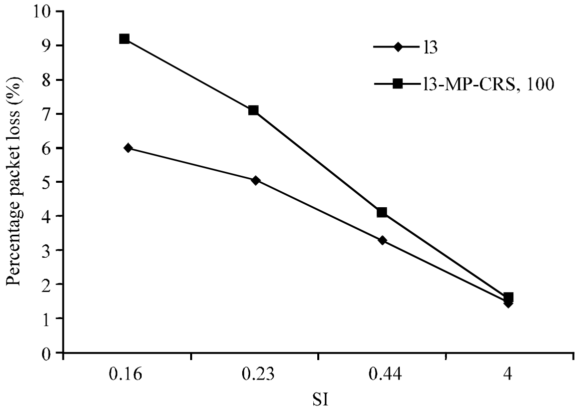


Fig. 15: Receiver movement (stationary index "SI") versus packet loss percentage

if the movement is more often (compared with each communication session), the ratio of I3 packets which takes performance gain from the proposed protocol become less. Especially at very low SI value, there are a high number of out-of-date I3 control packets which are generated from the proposed protocol.

Another performance indicator which is only applicable to the case of receiver movement is the packet loss rate. As shown in Fig. 15 the packet loss rate of both original I3 and proposed I3-MP are not different much and increases when receiver moves more often (lower SI). The packet loss may occur during the new location updating period which is divided into the following 2 phases; i.e., (P1) new location trigger is sending from the receiver to the MS I3 server and (P2) the distribution trigger is sending from the MS to the SLS. The original I3 experiences only P1 while the proposed I3-MP experiences both. However, the P1 is quite high relative to the P2 since the new location trigger needs to traverse through several I3 servers before reaching the trigger's MS while the MS knows the SLS's IP address and can send the distribution trigger directly to the SLS.

DISCUSSION

Although, the simulation results shown in the last section suggests that the proposed multipath routing protocol works as desired and can significantly increase the performance of the original I3 network, there are some issues needed to be justified.

The first issue is whether the proposed multipath extension is applicable to any I3 network employing the P2P protocol other than the Chord such as CAN, Pastry and Tapestry. In this paper, the proposed multipath is initially designed for I3 networks running Chord P2P protocol for its trigger/data packet distribution. However, the key functionality in the proposed method uses the common features (ID/address stacking and packet redirecting functions) of the I3 system which are independent to the employing P2P protocol. There will be only the part of RS selection option, i.e., CRS, which may be needed to be modified to utilize each protocol's proprietary routing information, instead of the finger table information of the Chord, in order to accelerate the RS selection process. By the way, as also suggested by the original I3 paper, more exploration of I3 network with multipath extension using various P2P protocols might be an interesting further study topic.

The second issue is about mobility support provided by the I3 system. Even though the proposed multipath extension is designed to support for host mobility, it is unavoidable to still have some packet loss when a host moves. Generally, it takes approximately $4D_{ip}$ (Stoica *et al.*, 2004), for the updated trigger to reach the MS while the D_{ip} is the average end-to-end delay between a pair of hosts in the underlying IP network. During the trigger updating interval, some data packets of the ongoing connection may get lost. There are two cases. The first case is when the receiver changes IP but still accesses to the old RLS. In this case, only data packet using the path from SLS directly to receiver (old IP) will get lost since the unchanged RLS will update the new receiver IP in its own database when the updated trigger passing through it. In the second case when both receiver IP and RLS get changed, all data packets which are sent before the updated trigger reaches to the SLS will get lost. However, such packet loss during host movement can be seen in the other solutions (Vixie *et al.*, 1997; Perkins, 2002) providing similar mobility support functionality as well. For an example, the dynDNS (Vixie *et al.*, 1997) extends the normal DNS system to provide mobility support in the application layer by letting a moving host update its changing IP associated to its domain name automatically. The key difference between the proposed I3 protocol and the dynDNS in providing mobility support might be that the I3 can associate any arbitrary format of user defined name to IP while the dynDNS needs to use a fixed hierarchical domain name standard. Compared to dynDNS which normally updates new IP every 5-120 sec, the packet loss generated during IP updating interval which is about 1.2 sec (for D_{ip}= 300 msec) should be reasonable lower. By the way, a detail comparative study between systems providing mobility support in different protocol layer such as I3, dynDNS and Mobile IP might be another interesting research topic.

CONCLUSION

This study proposed framework of multipath transmission protocol over Chord-based I3 overlay network. The proposed multipath routing protocol uses ID/address stacking and packet redirecting function to intercept I3 packets at the Senders local server, then sends packets over multiple possible paths toward the destination. Some possible paths can be determined by the protocol itself while some other paths can be found by utilizing information from the trigger/packet routing Chord protocol. Based on simulation results, the proposed method functions appropriately and could achieve up-to 82.5% better end-to-end delay performance, compared with the original I3 especially when the network is big and sparse and the offered load in the network is moderated (around 34%). The proposed protocol is also applicable to I3 networks with mobile users where it performs better than the original I3 protocol. However, in high mobility environment its performance converses to that of original protocol. Note that the proposed multipath protocol is designed to be added as an option in the original I3 infrastructure for users which are like to use multipath transmission based on their application.

ACKNOWLEDGMENTS

Parts of this research are supported by the grant from Higher Education Commission, Ministry of Education, Thailand. The OPNET simulator software is supported by Telecommunication Research and Industrial Development Institute (TRIDI), Thailand.

REFERENCES

Androutsellis-Theotokis, S. and D. Spinellis, 2004. A survey of peer-to-peer content distribution technologies. ACM Comput. Surv., 36: 335-371.

Banani, S. and K. Wipusitwarakun, 2011. A multipath traffic transmission over I3 network infrastructure. Proceedings of the International Conference on Information and Communication Technology for Embedded System sec (IC-ICTES 2011), January 27-29, 2011, Pattaya, Thailand.

Banner, R. and A. Orda, 2007. Multipath routing algorithms for congestion minimization. IEEE/ACM Trans. Networking, 15: 413-424.

Farinacci, D., V. Fuller, D. Meyer and D. Lewis, 2012. Locator/ID Separation Protocol (LISP) Draft-ietf-lisp-22. February 12, http://tools.ietf.org/html/draft-ietf-lisp-22

Fuke, S. and Z. Yuhong, 2012. A hybrid P2P-CDN based on locality-awareness and interest-awareness. Inform. Technol. J. (In Press).

Gurtov, A. and T. Polishchuk, 2009. Secure multipath transport for legacy Internet applications. Proceedings of the 6th International Conference on Broadband Communications, Networks and Systems, September 14-16, 2009, Madrid, Spain, pp: 1-8.

Ishida, T. and T. Yakoh, 2008. Fault tolerant multipath routing with overlap-aware path selection and dynamic packet distribution on overlay network for real-time streaming applications. Proceedings of the IEEE International Workshop on Factory Communication Systems, May 21-23, 2008, Dresden, Germany, pp: 287-294.

Jian, R., 2006. Internet 3.0: Ten problems with current Internet architecture and solution for the next generation. Proceedings of the Military Communications Conferences, October 23-25, 2006, USA., pp: 1-9.

Meyer, D., L. Zhang and K. Fall, 2007. RFC 4984 report from the IAB workshop on routing and addressing. RFC 4984, September 2007.

Moskowitz, R., P. Nikander and P. Jokela, 2006. Host Identity Protocol (HIP) architecture. RFC4423, May 2006, http://www.ietf.org/rfc/rfc4423.txt.

Murase, M., H. Shimonishi and M. Murata, 2006. Overlay network technologies for QoS control. IEICE Traans. Commun., E89-B: 2280-2291.

Nordmark, E. and M. Bagnulo, 2009. Shim6: Level 3 multihoming shim protocol for IPv6. RFC 5533, June 2009, http://tools.ietf.org/html/rfc5533.

O'Dell, M., 1997. GSE: An alternate addressing architecture for IPv6 (draft-ietf-ipngwg-gseaddr-00. February 24, http://tools.ietf.org/html/draft-ietf-ipngwg-gseaddr-00

Pan, J., S. Paul, R. Jain and M. Bowman, 2008. MILSA: A mobility and multihoming supporting identifier locator split architecture for naming in the next generation Internet. Proceedings of IEEE Global Communications Conference (Globecom), November 30-4 December, 2008, New Orleans, LA, USA.

Perkins, C.E., 2002. Mobile IP. IEEE Commun. Mag., 40: 66-82.

Rowstron, A. and P. Druschel, 2001. Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems. Proceedings of the IFIP/ACM International Conference on Distributed System sec Plat-form sec (Middleware 2001), November 2001, Heidelberg Germany, pp: 287-293.

Sohn, S., B.L. Mark and J.T. Brassil, 2006. Congestion-triggered multipath routing based on shortest path information. Proceedings of the 15th International Conference on Computer Communications and Networks, October 9-11, 2006, Arlington, VA., USA., pp: 191-196.

Stoica, I., R. Morris, D. Karger, M.F. Kaashoek and H. Balakrishnan, 2001. Chord: A scalable peer-to-peer lookup service for Internet applications. Comput. Commun. Rev., 31: 149-160.

Stoica, I., D. Adkins, S. Zhuang, S. Shenker and S. Surana, 2004. Internet indirection infrastructure. IEEE/ACM Trans. Networking, 12: 205-218.

Vixie, P., S. Thom secon, Y. Rekhter and J. Bound, 1997. Dynamic updates in the Domain Name System (DNS UPDATE). IETF RFC 2136. http://art.tools.ietf.org/pdf/rfc2136.pdf

Zhao, B.Y., L. Huang, J. Stribling, S.C. Rhea, A.D. Joseph and J.D. Kubiatowicz, 2004. Tapestry: A resilient global-scale overlay for service deployment. IEEE J. Selected Areas Commun., 22: 41-53.