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 11 (8): 959-970, 2012
ISSN 1812-5638 / DOI: 10.3923/itj.2012.959.970

A Protocol of Reliable Multicast between CE and FE in ForCES Architecture Network Element

Chuanhuang Li, Kelei Jin, Weiming Wang and Dan Hu
College of Information and Electronic Engineering, Zhejiang Gongshang University,
Hangzhou 310018, China

Abstract: ForCES (Forwarding and Control Element Separation)-an open programmable network architecture, is one of the important direction in the research area of NGN (next-generation network). In an NE (network element) with this architecture, CE (Control Element) need send many types messages to FEs (Forwarding Element), such as forwarding tables in ForCES router. For improving the transmission efficiency, saving network bandwidth and reducing the delay, a transmission scheme between CE and FEs is needed in this architecture. By analyzing the requirement of communication in ForCES channel, a protocol of reliable multicast based on TCP/UDP TML (Transport Mapping Layer) is proposed to do the work in this study. The interaction process and detailed protocol packet formats are introduced, and the petri net validation indicates the protocol is safe and living. The performance evaluation for this protocol is also given. It shows the reliable multicast protocol has better performance than the TCP unicast, if CE need send a large number of same messages to many FEs in a ForCES NE.

Key words: ForCES, ForCES channel, reliable multicast, TCP/UDP TML

INTRODUCTION

In ForCES (Forwarding and Control Element Separation) architectural framework, a ForCES NE (Network Element) is composed by one primary CE (Control Element), some backup CEs and multiple FEs (Forwarding Element) (Khosravi and Anderson, 2003; Yang *et al*., 2004). CEs and FEs are one hop or multiple hops away from each other and they use ForCES protocol (Doria *et al*., 2010) to exchange information. CE is a logical entity that can do dynamic configuration and management to the resources of the FEs. It also implements control and signaling protocols. FE is a logical entity that provides per-packet processing and can be managed and controlled by CE. The ForCES interface between CE and FE includes two parts: the PL (Protocol Layer) and the TML (Transport Mapping Layer). Two different type messages exchanged between CE and FE are defined in ForCES PL. One is a control messages, such as association setup messages, LFB (Logical Function Block) (Halpern and Salim, 2010) configuration and query messages, event notification messages etc. The other is redirect messages, such as the routing protocol packets. These messages need to be transferred by TML. ForCES working group in IETF (The Internet Engineering Task Force) has defined a TML based SCTP (Stream Control Transmission Protocol) (Salim and Ogawa, 2010). In user's implementation, there can be more other TMLs, such as TCP/DCCP (Khosravi *et al*., 2006), TCP/UDP TML (Wang *et al*., 2007) etc.

CE sends all kinds of messages to FEs by unicast in SCTP-based TML (Salim and Ogawa, 2010). That is, if there are 100 FEs and they all want to receive the same message, CE need to send 100 copies. Apparently, this will not only increase the overhead of CE and the delay of FE, but also affect the performance of the whole network. For example, updating routing table is a common manipulation to a router. Generally, there are around 250,000 routing records in a core router (Schiller, 2006). If the core router is ForCES architecture with one CE and one hundred of FEs, then CE need send at least 250,000*100 configure messages. If there are more FEs, the number will be more large. The network bandwidth between CE and FEs will be wasted and the delay will be larger.

To reduce the ForCES message's transmission delay and increase the bandwidth utilization between CE and FEs, a reliable multicast protocol was proposed based on TCP/UDP TML. In this TML mode, all redirect messages are transferred by general UDP unicast, control messages are transferred by TCP unicast or reliable multicast. In above example, we can use the reliable multicast to transfer the configure messages of distributing routing table.

Corresponding Author: Chuanhuang Li, College of Information and Electronic Engineering, Zhejiang Gongshang University, Hangzhou 310018, China

ForCES Working Group has finished ForCES requirements (RFC 3654) (Khosravi and Anderson, 2003 and ForCES framework (RFC 3746) (Yang *et al*., 2004) in 2003 and 2004, respectively. Since then, they focused on the work of standards: ForCES protocol (Doria *et al*., 2010), FE model (Halpern and Salim, 2010), ForCES SCTP-based TML (Salim and Ogawa, 2010), ForCES MIB (Haas, 2010), LFB library definitions (Wang *et al*., 2011). At present, except LFB definitions, all became into RFC in 2010.

In recent years, more and more people have joined the research work of NE based on ForCES architecture. FlexiNET project (Haas and Suzuki, 2005) of IBM used ForCES architecture to design a distributed router. Through the module's dynamic addition and deletion in node, they can make the forwarding function loading and unloading. SUN's Neon project studied the architecture and implementation of programmable network device. They followed the basic idea of control plane and forwarding plane separation in its architecture. But they used the private protocol specification to provide an integrated network services in FE model (Schuba *et al*., 2005). DHCR project (Louati *et al*., 2004, 2008; Houidi *et al*., 2006) in Communication Institute of French developed ForCES router software architecture which was based on software component technology to achieve the dynamic deployment of network services and used CORBA middleware technology to support internal communication in the DHCR.

DROP project (Bolla *et al*., 2010) of Genoa University mainly focused on the CE implementation in the router. The research team of National Defense University of China studied the IPv6 router based on ForCES architecture. They developed a new generation ForCES-based router which are implemented based on the network processors and used self-development interface protocol (Zhao *et al*., 2005). Zhejiang Gongshang University, developed a ForCES router prototype system -ForTER (Wang *et al*., 2006), by using Intel IXP2851/2400 network processor development board and systems integration approach. They also focus on the research of flow control mechanism in ForCES router (Zhuge *et al*., 2011).

From the point of IP multicast being introduced at the mid of 1980s, multicast has been developed toward controllable and manageable direction. IETF Reliable multicast working group proposed three reliable multicast technologies: Ring-based, Cloud-based and Tree-based (Handley *et al*., 2000). Sally Floyd proposed SRM (Scalable Reliable Multicast) protocol (Floyd *et al*., 1997) which can ensure the reliability and also can solve the problem of network complexity in 1996. SRM has a good scalability and robustness but its delay feedback mechanism has greatly increased the delay of packets reparation. RMP (Reliable Multicast Protocol) (Garcia-Molina and Spauster, 1991) is ring-based reliable multicast protocol and it allows distributed processing and shared the load by all the nodes, but it has not solved the packets buffering problem which only a single node is responsible for. RMTP (Reliable Multicast Transport Protocol) (Lin and Paul, 1995) is a hierarchical tree-based reliable multicast protocol. It can release the load of sender and the network. Because the logic relationships among the root and the intermediate nodes of the tree are dependent on the sender, it is difficult to meet the demand for reliable communication of many sources to many destinations case. M. Hidell from Swedish Royal Institute of Technology defined Forz protocol to implement the router based on ForCES architecture (Fu and Hagsand, 2005; Hidell *et al*., 2005). It is worth to mention that, in the mean time, they used NORM (Negative-acknowledgment (NACK)-Oriented Reliable Multicast Protocol) (Adamson *et al*., 2004) to transfer Forz messages (such as forwarding table) which need be highly reliable and tested the performance of send messages to FEs (less than 16) by TCP, UDP, NORM, respectively (Hidell and Sjodin, 2006).

THE PROTOCOL OF RELIABLE MULTICAST FOR FORCES TRANSFER CHANNEL

The architecture of reliable multicast: In ForCES router, CE is in charge of configuring the FEs and updating the table and data structure which needed by FEs. So, the reliable multicast needed is always sending packets from CE to FEs and the FEs just need to receive the packets. The reliable multicast architecture in ForCES router is showed in Fig. 1.

Reliable multicast control module (RMCM) is in-charge of setting up and deleting reliable multicast list. In the meantime, it also controls the packets' security level. That is, when ForCES messages have the requirement of security, RMCM will deliver them to Reliable Multicast Security Control Module (RMSCM) to perform functions as the authentication of source address of the packet, generation and distribution of the group secret key; otherwise, they will be processed by RMCM directly. RMSCM is in charge of the negotiation of multicast Security Association (SA). Reliable Multicast Module (RMM) is for sending the ForCES messages by reliable multicast.

When the upper application need use reliable multicast to send ForCES messages, the following process will be acted. (1): Launching RMCM, setting up reliable multicast list (FEs group which need receive the message),

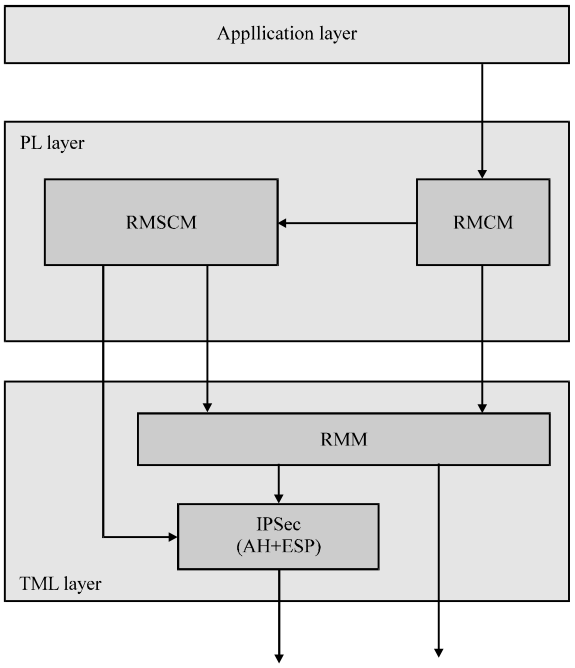


Fig. 1: Architecture of reliable multicast

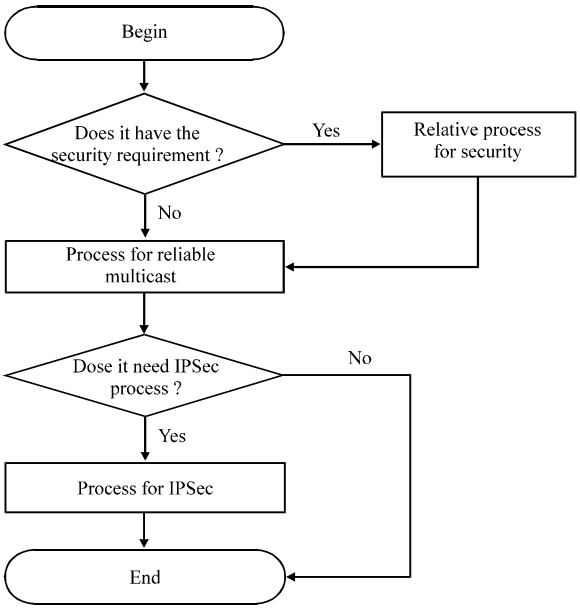


Fig. 2: Flowchart for reliable multicast

and checking the security requirements of reliable multicast. If the security is required, step (2) is wanted; otherwise, jump to step (4); (2): Acting the security processing. CE negotiates with FEs in the multicast list and Virtual Cluster Master (VCM) over group secret keys and the parameters of SA needed and then performs

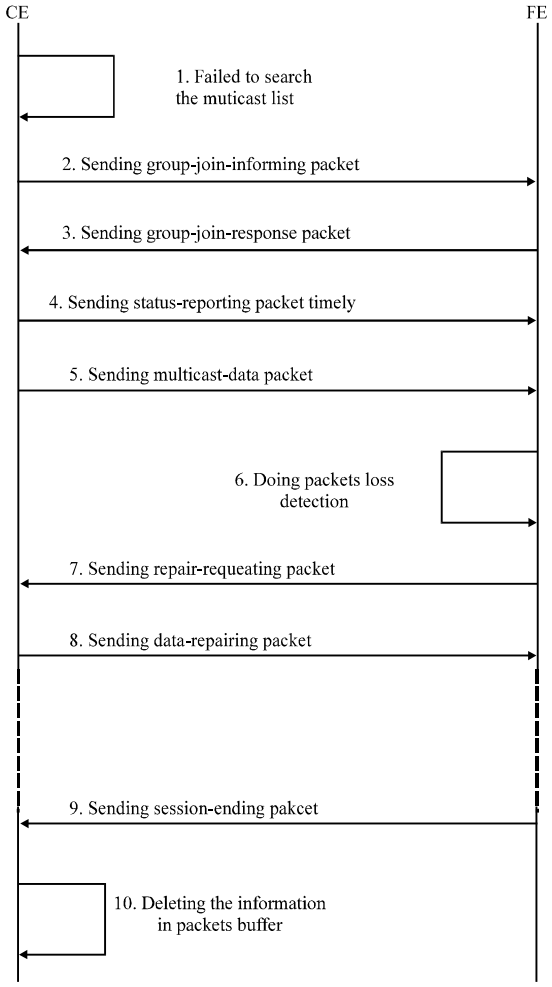


Fig. 3: Scenario of message exchange between CE and FE

authentication of source address of the packet; (3): Waiting for the ending of step (2); (4): Handling reliable processing through using RMM. (5) If the security is required, IPSec will be used. Flowchart is showed in Fig. 2.

We divide the interaction process with reliable multicast between CE and FEs into 3 phases, as showed in Fig. 3, multicast preparing phase (1, 2, 3), multicast sending phase (4, 5, 6, 7,8, 9), multicast ending phase (10).

Multicast preparing phase is responsible for the preparation job. In traditional IP multicast, the host can join in a new group by using IGMP (Internet Group Management Protocol). In ForCES router, CE is in charge of setting up, updating and configuration the messages in FEs, so it is also responsible for informing FEs to join in a new multicast group by using IGMP.

In multicast sending phase, beside the normal multicast packets sending, FEs need to check whether it has received all the packets (package loss detection) and

CE need to retransfer the packets which have be lost. This phase is the most complex stage.

The multicast ending phase is responsible for reclaiming the system resources which have been used in one multicast transfer process.

The following is the scenario of messages exchanged in all 3 phases.

- CE checks whether the multicast list is existed in RMCM. If it exists, multicast sending phase will be started at once. Otherwise, an empty list will be created and CE itself will join in a new special multicast group by IGMP
- In multicast preparing phase, CE sends a notice message (group-join-informing packet) to FEs those CE wants them to receive the ForCES multicast messages. This message is for informing FEs to join in the special group. It contains multicast group ID and group address information
- When one FE received this notice message, it will join in the group by IGMP. Then, send a response to tell CE that it has joined in the group (group-join-response packet)
- Entering the multicast sending phase. CE timely sends the status-reporting packet to all FEs in the group. This packet contains the information of maximal sequence number of the packets CE has sent
- CE sends the data packets (multicast-data packet) by using multicast. These packets are stored in CE's packets buffer temporarily
- After FE has received the status-reporting packet, FE will do packet loss detection. If there are no packets lost, the data packets will be sent to the upper layer to get further process
- If there is some packet lost, FE will send repair-requesting packet to CE
- CE searches the packets buffer to get the packet needed to be repaired and then it will send the Data-repairing packet to the corresponding FE
- FEs in the group will all send a session-ending packet to CE to tell it they have received all the data packets
- When the multicast sending phase is over, CE will delete the messages stored in the packets buffer temporally. Moreover, CE will maintain the multicast list for reuse next time

Stochastic petri net model validation: From the flowchart of ForCES reliable multicast interaction showed in Fig 3, we can get his SPN(Stochastic Petri Net) model which is showed in Fig. 4.

The explanation of the SPN's position and transition is showed in Table 1.

Assuming that the transition rate is as follows:

$$\lambda_0 = \lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = \lambda_5 = \lambda_6 = \lambda_{16} = 0.01$$
$$\lambda_7 = \lambda_8 = \lambda_9 = \lambda_{10} = 1.0,\ \lambda_{11} = \lambda_{12} = \lambda_{13} = \lambda_{14} = 0.1,\ \lambda_{15} = 0.9$$

We use the PIPE (Platform Independent Petri Net editor, a simulation software tool for stochastic petri net) to simulate the stochastic petri net model of the reliable multicast transmission and we can get the set of tangible states as Table 2 showed and reachability graph as Fig. 5, showed.

Remark:

- From Table 2, we can know that the number of the token in every position is no more than one, so the SPN model of the reliable multicast transmission is bounded and safe

Table 1: The explanation of the SPN's position and transition

The definition of position	The definition of transition
P0: CE is in the multicast preparing stage;	T0: CE searches the multicast list in the protocol layer;
P1: CE create a empty multicast list in the protocol layer;	T1: CE adds a new multicast group through using the IGMP protocol;
P2: CE utilize the existing multicast list;	T2: CE searches the multicast list in the protocol layer;
P3: Group Join Information(GJI) message is in the TML;	T3: CE sends the GJI to the FEs through the control protocol message channel
P4: FEs are in the multicast preparing stage;	T4: FEs accept the GJI;
P5: FEs join in multicast group;	T5: FEs reply the GJR;
P6: CE prepare to receive Group Join Response(GJR) message;	T6: CE accepts the GJR;
P7: GJR is in the TML;	T7: CE sends the MD;
P8: FEs are in the multicast accepting stage;	T8: FEs accept the MD;
P9: CE is in the multicast sending stage;	T9: CE sends the SR timed;
P10: Multicast Data(MD) is in the TML;	T10: FEs accept the SR;
P11: FEs prepares to accept the Status Reporting(SR) message;	T11: CE accepts the RR;
P12: CE encapsulates to accept the SR;	T12: FEs send the RR;
P13: SR is in the TML;	T13: CE sends the DR;
P14: FEs detects the losing data packets;	T14: FEs accept the DR;
P15: CE prepares to accept the Repair Requesting(RR) message;	T15: FEs don't detect the losing data-packets;
P16: Data Repairing(DR) packet is in the TML;	T16: FEs accept the SE from the upper layer module;
P17: FEs prepares to accept the DR;	
P18: CE searches the repair data in the BMU;	
P19: DR is in the TML;	
P20: FEs submit the Session Ending(SE) packet to the upper layer module;	

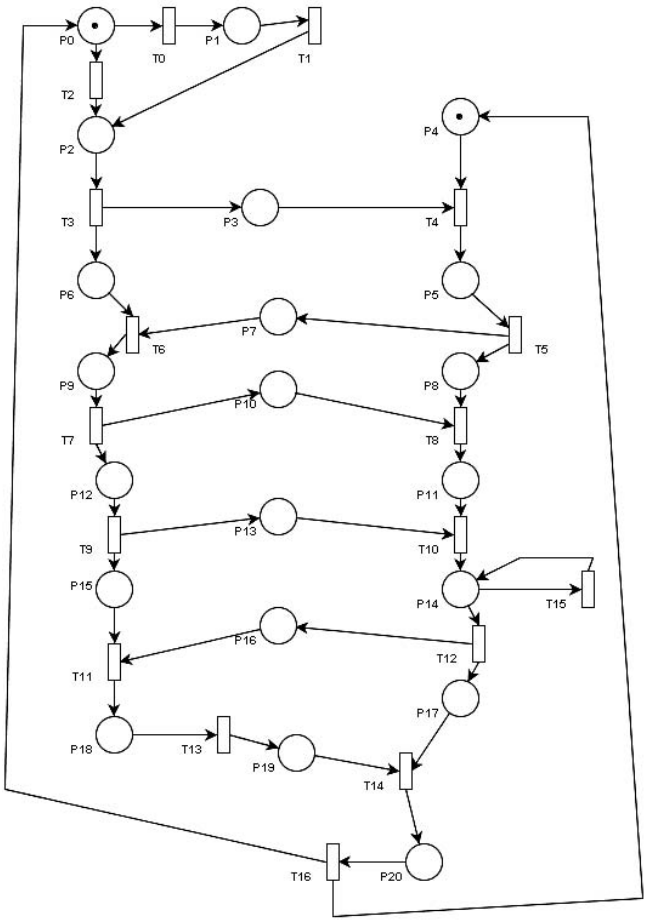


Fig. 4: SPN model of the reliable multicast transmission

Table 2: Set of tangible states

	Location																				
Mark	P0	P1	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P2	P20	P3	P4	P5	P6	P7	P8	P9
M0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
M1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0
M2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
M3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0
M4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
M5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0
M6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
M7	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
M8	0	0	1	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0
M9	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
M10	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
M11	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
M12	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0
M13	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
M14	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0
M15	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

- Each transition in the reachability graph is fired once at least and no transition is never fired, so for the reachability set R(M0), there is one transition path from the root mark M0 to M to each mark M. We can get the result that the SPN model of the reliable multicast transmission is living and deadlock could not occur

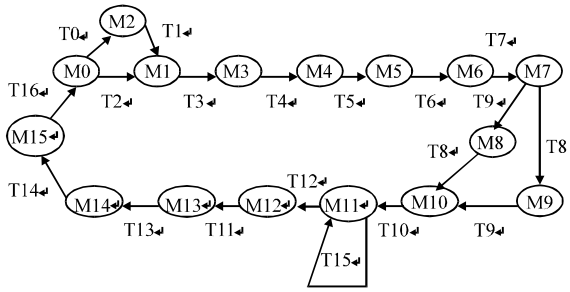


Fig. 5: Reachability graph

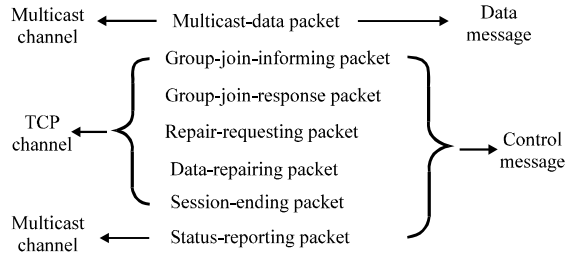


Fig. 6: Messages transferred in reliable multicast

- For each M'ªR (M0) in Fig. 5, it is true that M0ªR(M'), so it means that the SPN model of the reliable multicast transmission is invertible

In the SPN model, the average rate of the transition is influenced by the number of the FEs. We assume there are 200, 500 and 1000 FEs respectively in the SPN model of the reliable multicast transmission, so the average rates of the transition of each case are different but the average rates of transition T0, T2, T5, T6, T11, T12, T15 and T16 are same in the three cases. Here we adopt the average rates as the empirical value. We use rate for short of the average rate of the transition and No. for short of the number of the FEs. The rates in these three cases are showed in Table 3.

By using the PIPE to simulate the SPN model in the three cases, we can calculate the empirical average delay of the SPN model and that is the empirical average delay of the reliable multicast transmission procedure. Compared with the empirical average delay in the unicast transmission, we can get Table 4. We use Delay for short of the empirical average delay.

As showed in Table 4, the average delay increases linearly with the increasing of FEs number when using the unicast transmission. By contrast, the average delay increases slowly when we use reliable multicast transmission. Therefore, through theoretical validation by using petri net model, we can know the reliable multicast can improve the performance of the ForCES NE.

Table 3: Rates in three cases

	Rate								
No.	T1	T3	T4	T7	T8	T9	T10	T13	T14
200	0.5	0.5	0.5	2	1.8	2	2	0.2	0.2
500	0.2	0.2	0.2	5	4.5	5	5	0.5	0.5
1000	0.1	0.1	0.1	10	9	10	10	1	1

Table 4: Delay in two transmissions mode

	Delay	
FE Number	Unicast transmission	Reliable multicast transmission
200 FEs	100	46.964
500 FEs	250	48.4863
1000 FEs	500	61

Table 5: Defined packets type

Packet type	Value
Group-join-informing	0x01
packet	0x02
Group-join-response	0x03
packet	0x04
Multicast-data packet	0x05
Repair-requesting packet	0x06
Data-repairing packet	0x07
Status-reporting packet Session-ending packet	

Protocol packets format: All types of packets transferred in the reliable multicast process include: MD (Multicast-data) packet, GJI (Group-join-informing) packet, GJR (Group-join-response) packet, RR (Repair-requesting) packet, DR (Data-repairing) packet, SE (Session-ending) packet, SR (Status-reporting) packet. As showed in Fig. 6, in order to ensure the reliability, the control messages, except SE packet, are transferred by using TCP channel. SE packet is required to send to all FEs and need not be reliable, so it is transferred by multicast channel.

These types' messages have a common header, as showed in Fig. 7 which includes the fields of version, Message Type, Length. Version is the current reliable multicast version number. Message Type indicates the seven message types introduced just now. The defined values are showed in Table 5. The value of length field is the total message length including common header length.

Group-join-information packet (GJI packet): Group-join-informing packet is for CE to notice FEs to join a multicast group. As showed in Fig. 8, only MulticastID (multicast group Identification) is needed in this kind of packet. CE knows the information of all associated FEs, including IP address and it will inform the FEs whom CE wants them to receive the multicast packets to join the special group by using TCP's reliable connection.

Group-join-response packet (GJR packet): FEs will response to CE whether they has joined the special group, when CE informed them to join. Fig. 9 shows the contents of GJR packet. It includes MulticastID (Multicast Group Identification) and ValidInfor (Validation Information). MulticastID is the group ID that CE has

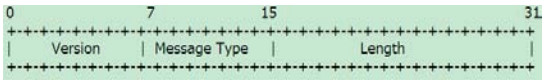


Fig. 7: Common header

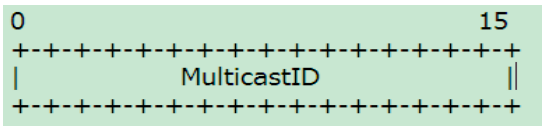


Fig. 8: Group-join-informing packet (GJI packet)

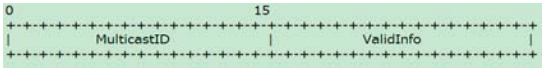


Fig. 9: Group-join-response packet (GJR packet)

informed to FEs to join. ValidInfor shows that whether FE has joined the current group. The values of ValidInfor are defined as follows:

- 0x00-to indicate the FE failed to join the multicast group. CE need to decide what should be done next, such as resending a Group-join-informing message, or giving up the multicast sending
- 0x01-to indicate the FE succeed to join the multicast group

Multicast-data packet (MD packet): Multicast-data packets are the real messages that CE wants to send to some FEs by multicast. They are sent to FEs by using IP multicast. The transmission of IP multicast is UDP. Owing UDP is unreliable and out-of-order, all the data should be stored in cash orderly at the first step.

As showed in Fig. 10, the Multicast-data packet is composed of packet header and packet body. Packet header shows the information of current multicast data packet. It contains three parts: MulticastID (multicast group identification), ApplicationID (application identifier), SeqNumber (sequence number). ApplicationID is for different application. For example, in the application of dispatching routing table to all FEs, the packets are all with the same ApplicationID. SeqNumber is the packet's sending order number in the same application and group. These contents are convenient for FE to do packet loss detection and retransmission.

Repair-requesting packet (RR packet): T he pack loss detection will be implemented after FE received multicast-data packet from CE. If there are some packets lost, FE will send repair-requesting message to

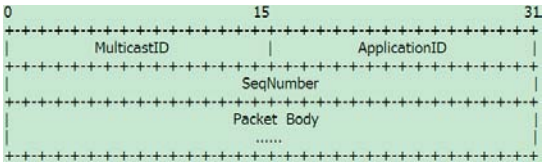


Fig. 10: Multicast-data packet (MD packet)

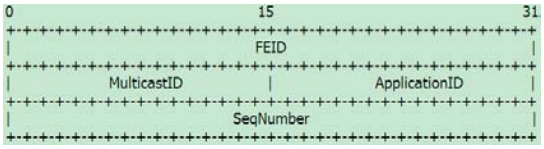


Fig. 11: Repair-requesting packet (RR packet)

CE. Fig. 11 shows the contents of RR packet. This packet hasn't data payload and it only includes some simple control information: FEID, MulticastID, ApplicationID, SeqNumber. These contents are used to identify which packet has been lost.

Data-repairing packet (DR packet): After received the repair-requesting packet from FEs, CE will find out the lost packet from buffer and encapsulate them into data-repairing packets to send to FEs. Data-repairing packet is a datagram that CE retransmits to specific FEs, so the message format is the same as multicast-data packet.

FE will send session-ending packet to report to CE that it has received all multicast data packets in the past interval time. As showed in Fig. 12, the packet contents are: FEID, MulticastID, ApplicationID, MaxSeqNumber. MaxSeqNumber indicates the packets with smaller sequence number in current application and group are all received. CE can use this information to maintain its packets buffer.

Status-reporting packet (SR packet): CE will send status-reporting packets timely by using multicast. These packets include the information of MulticastID, ApplicationID and the current maxim sequence number: MaxSeqNumber. The contents are showed in Fig. 13.

Implementation in ForCES architecture NE: The jobs of CE and FE in multicast process are decided by ForCES framework. So, the CE's reliable multicast modules is different from FE's. In this study, we introduce the reliable multicast modules in CE and FE, respectively.

Reliable multicast in CE TML: Reliable multicast sending function is implemented mainly by RMM in CE TML. As

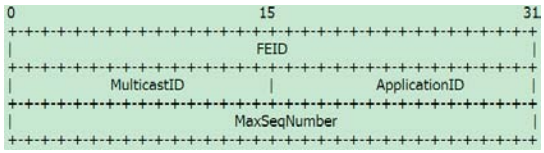


Fig. 12: Session-ending packet (SE packet) SE packet

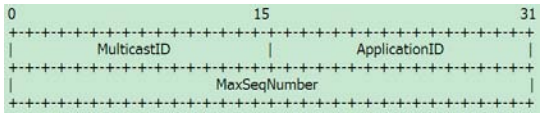


Fig. 13: Status-reporting packet

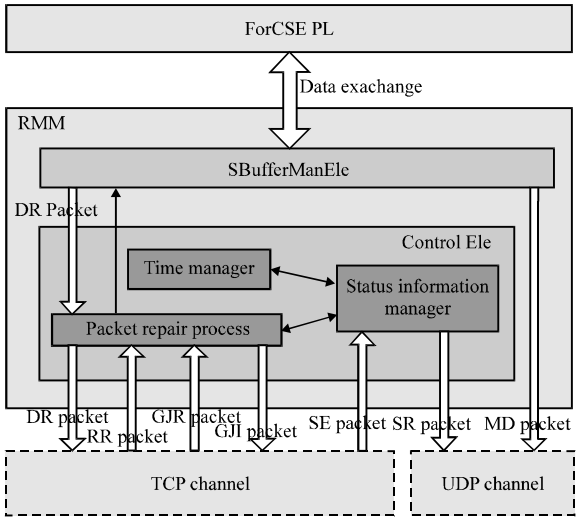


Fig. 14: Reliable multicast module in CE

showed in Fig. 14, it includes SBufferManEle (storage buffer management element) and ControlEle (control element).

SbufferManEle is for the storage of packets which transferred by using multicast. It is used to prevent packet loss because of the high rate of coming data from up layer. So, the data from upper layer should be stored in buffer and be sent to FEs one by one. When FE found some packet was lost, they will send repair-requesting packet to CE and CE will find the packet which is needed repaired in SBufferManEle and send it to FE.

The internal architecture of SBufferManEle is showed in Fig. 15. There are many buffers in SBufferManEle to be used to store different data and to send to different multicast group. For example, buffer 1 can store the data which will be sent to multicast group 1(ID 1); buffer 2 can store the data which will be sent to multicast group 2(ID 2). So CE can send messages to several multicast groups in parallel. In the mean time, there are many pages

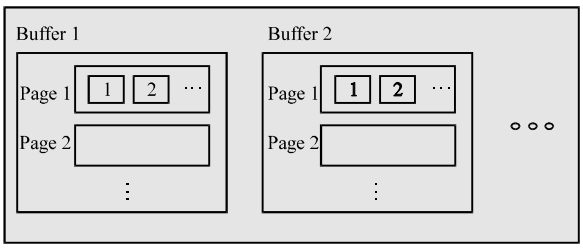


Fig. 15: Internal architecture of SbufferManEle

in one buffer. Different page is for different application. For example, the packets for dispatching routing table to all FEs are all in one page. The data in one page is stored in order. They are labeled by sequence numbers.

ControlEle is the core element of RMM in CE TML. which is in charge of sending RP packet and sending SR packet timely. It is composed by RRSM (repair-retransmission sub-module), SIMSM (status information management sub-module) and TMSM (timer management sub-module). RRSM will look for the data in SBufferManEle which is required from FEs when RRSM received a repair-requesting message. Then the data will be generated a repaired data which is sent to FEs by TCP. The timer is started by SIMSM which is in charge of reading the maxim sequence number from SBufferManEle in CE. Then the maxim sequence number will be packed in a status-reporting packet. At last, the status-reporting packet will be sent to all FEs in the current group.

At sending stage, CE will store the data in cache which will be sent to FEs. Then the data will be sent to FEs, respectively. The timer in SIMSM will be started when the timer receives the first message from SBufferManEle. Then the stage will be turned into status information flow: at first, checking the ending notice from RRSM, if the ending notice is exist, the status information flow is over; if the ending notice is not exist, checking the maxim sequence number, then the maxim sequence number will be wrote in a message which will be sent by redirection as SR packet and starting timer. Status information flow is showed in Fig. 16.

Reliable multicast in FE TML: The same as in the CE TML, there is RMM in FE TML. As showed in Fig. 17, it includes PRSSM (packets receiving and sending sub-module) and PLDSM (packets loss detection sub-module).

PRSSM is responsible for receiving and sending all kinds of packets. The TCP channel, we called control channel, UDP, redirect channel, respectively. Multicast-data packet and status-reporting packet are transferred through redirect channel. These packets will be passed to PLDSM to further process. The other packets are all

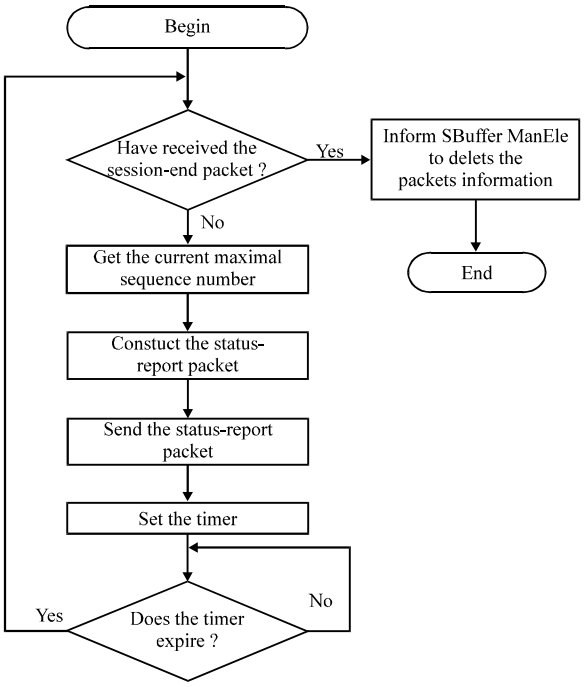


Fig. 16: Working flowchart in status information manager

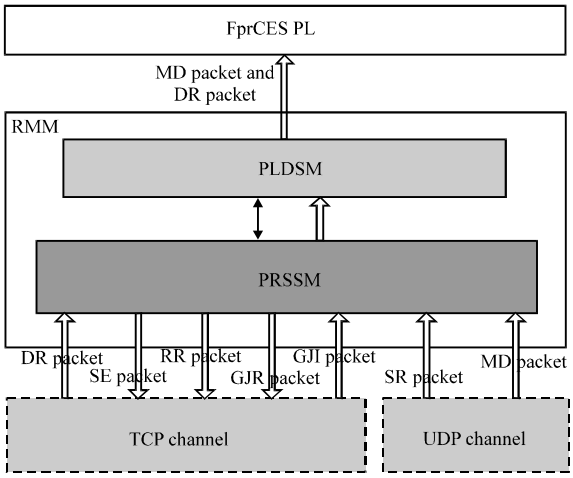


Fig. 17: Reliable multicast module in FE

transferred through control channel. PLDSM will inform PRSSM to generate the repair-requesting packet, if it found some packet has been lost. PRSSM will also generate session-ending packet, if PLDSM tells it the multicast sending for a special application from CE is end. When PRSSM received a data-repairing packet, it will update the FEDataNumTable (which record the packet sequence number FEs received) table and then deliver this packet to ForCES PL to further process.

Packets loss detection is the key method to ensure the reliable transfer for multicast packets. We can use the timer and sequence number policy to realize this reliability. When PLDSM received the message form PRSSM, it will act the following processing. (1) Checking the type of redirection message. If the message is status-reporting packet, we directly enter into 3). If the message is multicast-data packet, the FEDataNumTable should be checked to find out whether the packet is already received. If it already exists, the packet received will be dropped; otherwise, we enter into stage 2). (2) Updating the FEDataNumTable, then the packet will be delivered to ForCES PL to process. (3) Obtaining the maxim sequence number from the status-reporting packet and recording it in the Session Table 4) Checking the FEDataNumTable by the maxim sequence number to judge whether there is packet loss. If some packets are lost, the repair-requesting packet will be generated in PLDSM and be passed to the PRSSM sub-module to send; otherwise, we turn into 5). For example, if the records in FEDataNumTable are {1, 3, 5, 7} and the maxim sequence number is {8}, we think the packets with sequence number {2, 4, 6, 8} are lost. 5) Checking the sequence number in SessionTable, if there are the same sequence numbers in a SessionTable, we think the multicast sending for a special application from CE is over. And if packets loss detection is also finished, PLDSM will tell PRSSM to send a session-ending packet to CE. The flow chart for packets loss detection is showed in Fig. 18.

PERFORMANCE EVALUATION

From the previous sections, we can know the reliable multicast can work in ForCES element. In this section, we will test and compare the delay of interaction between CE and FEs by using TCP unicast and reliable multicast in a real environment. Several groups' data will be tested and the test time is conducted in PM 20:30~22:00, in which there is the same stable network environment. In normal stable network, the delay is related to the number of FEs and the number of packets CE sent. The two influencing factors will be tested, respectively.

Delay evaluation with altering FEs number: The test is conducted in a good network environment and the packet number sent by CE is 10485, 119 Bytes per packet. Then we observing the relationship between number of FEs and delay through alter the number of FEs. The result is shown in Fig. 19.

As showed in Fig. 19, when the number of FE is one, the total delay by using the reliable multicast is larger than that by using TCP unicast. That is because reliable

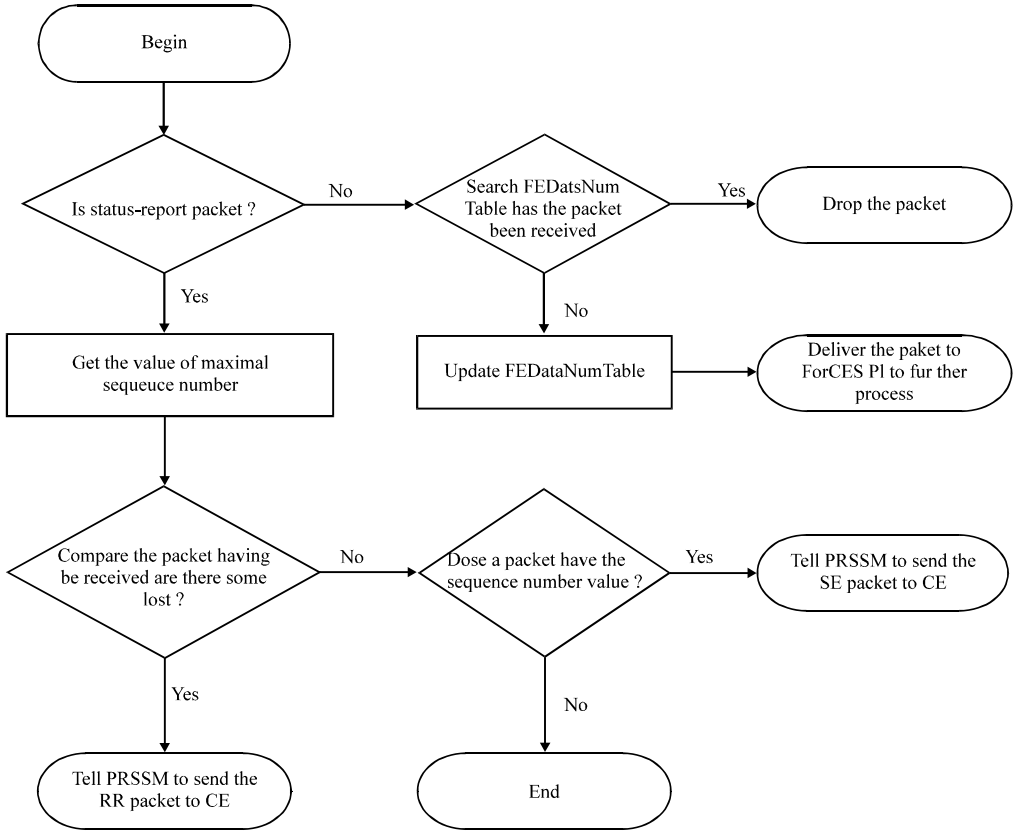


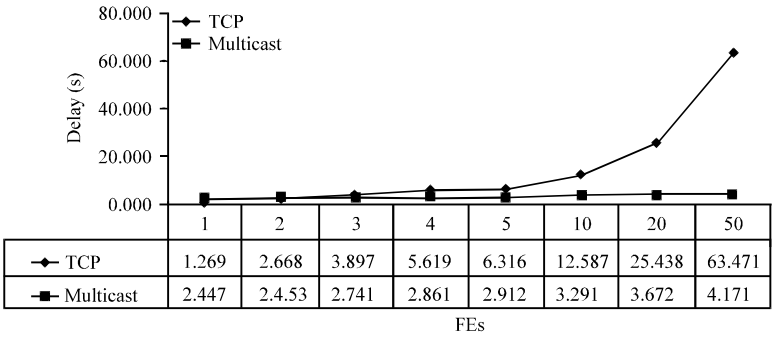
Fig. 18: Flowchart of packet loss detection module

multicast's preparatory stage needs interaction time. When the number is two, the total delay of TCP unicast is linear growing, however, reliable multicast's delay don't change obviously and the reliable multicast's delay is shorter than that of TCP unicast. When the number is greater than or equal to three, reliable multicast's delay is shorter, obviously. According to Fig. 19, the total delay of TCP unicast is linear growing with the increasing of FE's number; however, the delay of reliable multicast is stable. In the stable network environment and the unchangeable packet's number condition, we can conjecture that TCP's delay will be very long when the FE's number is greater than 100; however, reliable multicast's delay is shorter than TCP's. Assuming the FEs' number is 100, TCP's delay is 126.9 sec, however, reliable multicast's delay is 4.171 sec. To sum up, when the environment is a good network and the number of packet is unchangeable and FEs' number is big, the performance of reliable multicast is greater than TCP's.

Delay evaluation with altering packets number: In this section, the test environment is a stable network too, and the FE number is fixed. 5 FEs and 119 Bytes in one packet are under testing. The relationship between the number of packets and delay is shown in Fig. 20.

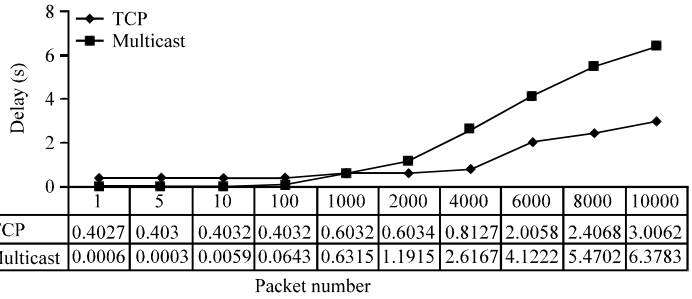
According to Fig. 20, the delay by using TCP unicast is linear growing with the FEs increasing. However, the delay by using reliable multicast is different from that of TCP unicast. When the number of packets is less than 1000, the delay of reliable multicast is stable (about 0.4 sec). Generally, the delay of reliable multicast is longer than that of TCP unicast in this case. That is because preparatory stage spends more time. However, if the number of packets is greater than 1000, reliable multicast's delay is increasing slowly, but the delay of TCP unicast is liner growth. When the number of packets is huge, the influence of preparatory stage to multicast's delay is small. But for that case by using TCP unicast, the whole packets that CE need to send to FE is huge. For example, if there are 2000 packets information need to be sent in CE, there are only 2000 packets need to be sent by reliable multicast, however, 2000*5 = 10000 packets need to be sent by TCP unicast.

In a severe network environment, the advantage of reliable multicast is not apparent. That is because the number of packets lost is huge, if the packets is



	1	2	3	4	5	10	20	50
TCP	1.269	2.668	3.897	5.619	6.316	12.587	25.438	63.471
Multicast	2.447	2.4.53	2.741	2.861	2.912	3.291	3.672	4.171

Fig. 19: The relationship between FEs number and delay



	1	5	10	100	1000	2000	4000	6000	8000	10000
TCP	0.4027	0.403	0.4032	0.4032	0.6032	0.6034	0.8127	2.0058	2.4068	3.0062
Multicast	0.0006	0.0003	0.0059	0.0643	0.6315	1.1915	2.6167	4.1222	5.4702	6.3783

Fig. 20: The relationship between packet number and delay

transferred by UDP in this severe environment and reliable multicast will send more repair-requesting packets and data-repairing packets. These will affect the multicast performance directly. If most of the UDP packets are lost, the multicast's delay will more higher than that of by using TCP unicast. In consideration of this case, we use a switching mechanism to solve. When the system detects the network environment being severe, the transmission mode will switch to TCP unicast mode.

CONCLUSION

We mainly proposed a protocol of reliable multicast used between CE and FEs in ForCES architecture NE. This protocol is suited for the case that CE need send a large number of same messages to many different FEs reliably. The interaction process of the protocol was presented. From the petri net validation, we saw the protocol was safe and living and it would not be deadlock in working. The six packet formats used in the protocol was detailed. A simple implementation example in ForCES architecture NE was also showed in this paper, including the structure in CE TML and FE TML. We also described the performance comparing result between the reliable multicast protocol and the TCP unicast, It showed the protocol had better performance than the TCP unicast in a stable network environment.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for helpful comments. This study is supported by the National High Technology Development 973 Program of China (No. 2012CB315902), National Natural Science Foundation of China (No. 60903214, 60970126), Zhejiang Provincial NSF of China (No.Y1090452, Y1100871, 1120KZ411060G1, Y1111117), Department of Education of Zhejiang Province Project (No. Y201018208) and Zhejiang Sci and Tech Project (No. 2011C21049).

REFERENCES

Adamson, B., C. Bormann, M. Handley and J. Macker, 2004. Negative-acknowledgment (NACK)-oriented reliable multicast (NORM) protocol. Network Working Group, IETF RFC 3940. http://www.ietf.org/rfc/rfc3940.txt

Bolla, R., R. Bruschi, L. Carbone and G. Lamanna, 2010. A cooperative middleware for distributed SW routers. Proceedings of the International Symposium on Performance Evaluation of Computer and Telecommunication Systems, July 11-14, 2010, Ottawa, Canada, pp: 250-257.

Doria, A., J.H. Salim, R. Haas, H. Khosravi and W.M. Wang *et al*., 2010. Forwarding and control element separation (ForCES) protocol specification. RFC 5810. http://tools.ietf.org/html//rfc5810

Floyd, S., V. Jacobson, C.G. Liu, S. McCanne and L.X. Zhang, 1997. A reliable multicast framework for light-weight sessions and application level framing. Trans. IEEE/ACM Network., 5: 784-803.

Fu, J. and O. Hagsand, 2005. Designing and evaluating network processor applications. Proceedings of the Workshop on High Performance Switching and Routing, May 12-14, 2005, Hong Kong, pp: 142-146.

Garcia-Molina, H. and A.M. Spauster, 1991. Ordered and reliable multicast communication. ACM Trans. Comput. Syst., 9: 242-271.

Haas, H. and T. Suzuki, 2005. Architecture of the Flexinet ForCES-based control point. Proceedings of the 63rd IETF Meeting, August 5, 2005, Paris.

Haas, R., 2010. Forwarding and control element separation (ForCES). MIB, RFC 5813. http://www.faqs.org/rfcs/rfc5813.html

Halpern, J. and J.H. Salim, 2010. Forwarding and control element separation (ForCES) forwarding element model. RFC 5812. http://tools.ietf.org/search/rfc5812

Handley, M., S. Floyd, B. Whetten, R. Kermode, L. Vicisano and M. Luby, 2000. The reliable multicast design space for bulk data transfer. Network Working Group, http://www.ietf.org/rfc/rfc2887.txt

Hidell, M., P. Sjodin and O. Hagsand, 2005. Control and forwarding plane interaction in distributed routers. Proceedings of the 4th International IFIP-TC6 Networking Conference, May 2-6, 2005, Waterloo, Canada, pp: 1339-1342.

Hidell, M. and P. Sjodin, 2006. Performance of NACK-oriented reliable multicast in distributed routers. Proceedings of the Workshop on High Performance Switching and Routing, June 7-9, 2006, Poznan, Poland.

Houidi, I., W. Louati and D. Zeghlache, 2006. An extensible software router data-path for dynamic low-level service deployment. Proceedings of the IEEE Workshop on High Performance Switching and Routing, June 7-9, 2006, Poznan, Poland, pp: 161-166.

Khosravi, H. and T. Anderson, 2003. Requirements for separation of IP control and forwarding. Network Working Group, http://www.ietf.org/rfc/rfc3654.txt

Khosravi, H., S. Chawla, F. Ansari and J. Maloy, 2006. TCP/IP based TML (transport mapping layer) for ForCES protocol. Working Group: ForCES, http://tools.ietf.org/html/draft-ietf-forces-tcptml-04.

Lin, J.C. and S. Paul, 1995. RMTP: A reliable multicast transport protocol. Proceedings of the IEEE INFOCOM'95, April 2-6, 1995, Boston, pp: 1414-1424.

Louati, W., B. Jouaber and D. Zeghlache, 2004. Configurable software-based edge router architecture. Proceedings of the 4th Workshop on Applications and Services in Wireless Networks, August 9-11, 2004, Boston, USA.

Louati, W., I. Houidi, M. Kharrat, D. Zeghlache and H.M. Khosravi, 2008. Dynamic service deployment in a distributed heterogeneous cluster based router (DHCR). Cluster Comput., 11: 355-372.

Salim, J.H. and K. Ogawa, 2010. SCTP-based Transport Mapping Layer (TML) for the forwarding and control element separation (ForCES) protocol. IETF RFC5811. http://tools.ietf.org/html/rfc5811

Schiller, J., 2006. IPv6 potential routing table size. Proceedings of the 65th Internet Engineering Task Force (IETF) Meeting, March 19-24, 2006, Dallas, TX., USA.

Schuba, C., J. Goldschmidt, K. Kalajan and M.F. Speer, 2005. Integrated network service processing using programmable network devices. SMLI TR-2005-138. http://labs.oracle.com/techrep/2005/smli_tr-2005-138.pdf

Wang, W., L. Dong and B. Zhuge, 2006. ForTER: An open programmable router based on forwarding and control element separation. Proc. DCABES, 2: 1069-1077.

Wang, W.M., L.G. Dong and B. Zhuge, 2007. TCP and UDP based ForCES protocol TML over IP networks. ForCES Working Group, http://tools.ietf.org/html/draft-wang-forces-iptml-02

Wang, W., E. Haleplidis, K. Ogawa, C. Li and J. Halpern, 2011. ForCES Logical Function Block (LFB) library. IETF Draft. http://tools.ietf.org/html/draft-ietf-forces-lfb-lib-06

Yang, L., R. Dantu, T. Anderson and R. Gopal, 2004. Forwarding and control element separation (ForCES) framework. RFC 3746. Network Working Group, The Internet Society. http://www.elook.org/computing/rfc/rfc3746.html

Zhao, F., J. Su and X. Cheng, 2005. OpenRouter: A TCP-based lightweight protocol for control plane and forwarding plane communication. Proceedings of the International Conference on Computer Networks and Mobile Computing, August 2-4, 2005, Zhangjiajie, Hunan.

Zhuge, B., C. Yu, K.P. Liu and W.M. Wang, 2011. Research on internal flow control mechanism of for CES routers. Inform. Technol. J., 10: 626-638.