Remote DLNA Communication System Based on NTMobile

Kohei SHIMIZU, Hidekazu SUZUKI
and Akira WATANABE
Graduate School of Science and Technology
Meijo University
Aichi, Japan 468-8502

Katsuhiro NAITO
Graduate School of Engineering,
Mie University
Mie, Japan, 514-8507

***Abstract*—With the spread of information home appliances certified by Digital Living Network Alliance (DLNA), it has become quite easy to share digital contents within a home network. In the meantime, the mobile Internet has been drawing much attention along with the popularization of high-performance mobile devices and the development of high-speed wireless communication technologies. Under such circumstances, there is a growing desire to obtain digital contents stored in a home network from a remote network in a simplified manner. In this paper, we propose a DLNA-based, remote communication method that enables the sharing of contents between a home network and a foreign network, by expanding functions of NTMobile (Network Traversal with Mobility), which is our original technology that can realize mobility and solve the NAT traversal simultaneously. By implementing our proposed method in a prototype system, we confirmed that a node can transmit media contents over a NAT and keep on browsing the contents even when the node moving to another network during communication.**

I. Introduction

In recent years, use of information home appliances certified by Digital Living Network Alliance (DLNA) [1] have been rapidly spreading. These devices can realize the sharing of contents kept in a home network (HNW), without any special settings, regardless of difference in manufactures and product types. Hence, there is also a growing desire for the sharing of contents stored in a DLNA device in the HNW with a network away from home, in a similarly simplified manner. Meanwhile, with popularization of high-performance mobile devices such as smartphones and the development of high-speed wireless communication technologies such as LTE and WiMAX, demand for the mobile Internet has been also increasing. And as a natural extension of such a desire and demand, there is now a desire of obtaining contents stored in HNW from remote locations even while moving.

However, because DLNA-certified devices are using Simple Service Discovery Protocol (SSDP) [2] in order to search for the most appropriate devices for communication, it is not possible to make the searching via a Network Address Translation (NAT) router. Even if you could find the device, you cannot receive a response from the device, as DLNA permits access from the same network only. If a user wants to make a connection from outside the home to a DLNA device at home, a different access method is required because of the NAT traversal problem.

Although there have been various technologies proposed that realize DLNA-based remote communications by solving the above-mentioned problems [3], [4], none of them supports mobility functions simultaneously. Therefore, if the IP address of a player node is changed, once established connection is broken and the user cannot keep on browsing the contents.

In this paper, we propose a new type remote DLNA communication system that solves the above-mentioned problem by expanding functions of our original NTMobile (Network Traversal with Mobility) [5]–[7] that realizes connectivity and mobility at the same time in IPv4 and IPv6 networks. In our proposed method, we make Digital Media Player (DMP) that is a node to reproduce the contents equipped with NTMobile with expanding functions. We also install in the HNW a device called "DLNA Agent" based on the expanded NTMobile. This device realizes a remote DLNA communication by relaying the communication between DMP outside the HNW and a Digital Media Server (DMS), which is the node holding the contents.

II. NTMobile System

As our proposed method is realized by extending certain function of NTMobile, which is our mobility technology, we first show the outline of the NTMobile in this Chapter.

A. Outline

NTMobile is a technology to realize mobility and solves the NAT Traversal problem in IPv4 networks simultaneously, by using virtual IP addresses and UDP tunnels. Fig. 1 shows the system configuration of NTMobile. NTMobile comprises NTMobile-compatible nodes (NTM nodes), Direction Coordinator (DC), and Relay Server (RS). DC manages the address information of NTM nodes and gives instructions to create UDP tunnels. RS relays packets between a pair of NTM nodes behind NATs, or relays packet between a NTM node and a general node which does not have NTMobile functions.

Each NTM node has a virtual IP address that does not change even if it switches its networks. Application in the NTM node establishes a connection with another node using their virtual IP addresses. As the connection is not affected by the change in the real IP address of NTM nodes, communication is not interrupted even if the node switches its networks. In the meantime, a UDP tunnel is created between

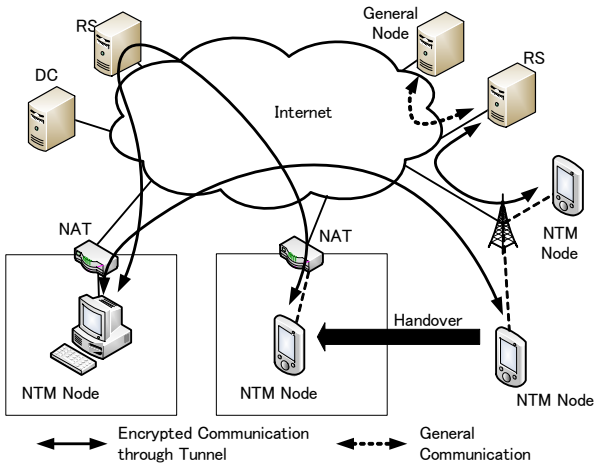

Fig. 1. System configuration of NTMobile.

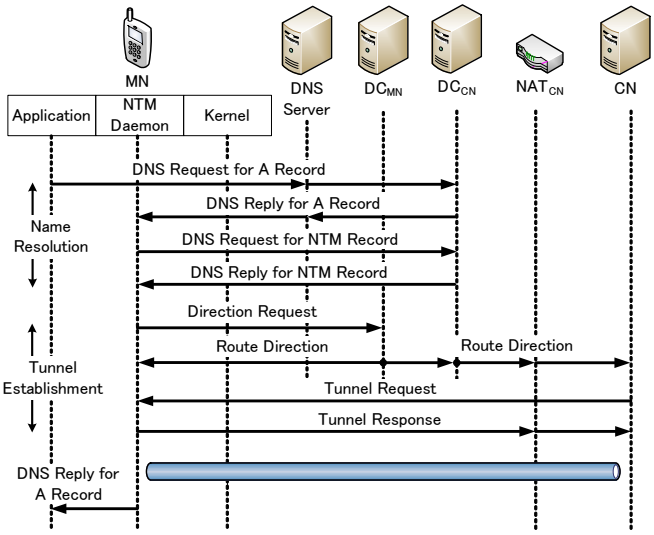

Fig. 2. Tunnel creation process between global to private.

a pair of NTM nodes in order to transmit packet through the real network.

An UDP session for the purpose of exchanging control messages is always maintained between NTM node and its DC. At the beginning of a communication, NTM node sends to the DC a request to create a tunnel with the correspondent node. DC, upon receipt of this request, issues instructions to both nodes to create a tunnel. As instructions are sent to NTM nodes through the DC that manages the corresponding NTM node, instructions can reach the NTM nodes behind the NAT by using the UDP session. In this way, it is possible to start communication from outside the NATs.

In NTMobile, a daemon program (NTM daemon) and the kernel module (NTM kernel module) are implemented in each NTM node. NTM daemon is userland program which exchanges messages for the purpose of creating a tunnel. NTM kernel module performs a function of hooking a particular packet, and the encapsulate/decapsulate of communication packets.

B. Tunnel creation procedures

Fig.2 shows the tunnel creation procedures of NTMobile. In Fig. 2, MN begins to communicate with CN behind NAT. Here, which MN and CN are both NTM nodes. MN sends a DNS A query with CN's FQDN to a DNS Server, and receives a DNS reply. At this timing, MN temporarily stores the reply, and MN sends a NTM query to DC_{CN} in order to get CN's NTM record. NTM record contains address information, such as real IP address (RIP) and virtual IP address (VIP).

After MN gets CN's NTM record, MN sends a Direction Request to DC_{MN}. This message contains both MN's NTM record and CN's NTM record.

DC_{MN} receives a Direction Request from MN, and determines the optimal tunnel route from the address information of MN and CN. After determining the tunnel route, DC_{MN} sends instruction to MN and CN by a Route Direction. Route Direction is message to NTM Node to instruct tunnel creation. In Fig. 2, because CN exist behind the NAT_{CN}, Route Direction is sent to CN via DC_{CN}.

MN and CN, upon receipt of the Route Direction, exchange the Tunnel Request/Response for creation of a tunnel. In the case of Fig. 2, CN behind NAT_{CN} first sends the Tunnel Request to MN, and MN, upon receipt of Request, sends a Tunnel Response to CN. In this way, a tunnel is created between MN and CN.

After completing the tunnel creation, MN rewrites CN's real IP address "RIP_{CN}" (described in the DNS A Reply) to virtual IP address "VIP_{CN}", and passed over to upper software. Through this process, application starts communication with MN's and CN's virtual IP addresses and the communication is transmitted through UDP tunnel

III. Our Proposed Method

We propose a DLNA-based, remote communication system that enables communications between a DLNA-certified device in the HNW and the node on the Internet, by applying the NTMobile technology.

A. Configuration of our proposed method

Fig. 3 shows the system configuration of our proposed method. In this system, NTMobile function is implemented in the node that acts as DMP, and a special NTM node called "DLNA Agent" or "DA" equipped with expanded functions of address translation and packet forwarding capability is installed in HNW. DA plays the role of RS in NTMobile. In other words, this node has a function to relaying functions between DMP equipped with the NTMobile function and a general DMS, and DMP creates a tunnel to connect with DA in order to communicate with DMS in HNW. DA has also the role of searching for the most appropriate DMS in HNW at the request of DMP.

AS the assumption, DMP is supposed to know FQDN of DA.

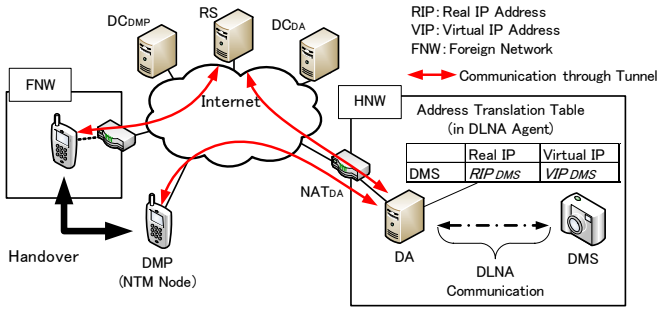

Fig. 3. System configuration of proposed method.

B. Extension of the NTMobile functions

We expand the function of NTMobile to be implemented in DMP and DA in order to realize our proposed method. Extension function of each terminal is as follows.

1) DMP:

- Addition of a trigger function to create a tunnel.
  As a trigger to create a tunnel, we add a function to dispatch on M-SEARCH message. With this addition, DMP is able to create a tunnel to get connection with DA in the DLNA sequence.

2) DA:

- Function of pooling virtual IP addresses.
  DA has multiple virtual IP addresses to be connected with DMSs in HNW. DA assigns a virtual IP address (VIP_{DMS}) to the real IP address (RIP_{DMS}) of each DMS.
- Functions of address translation and packet transmission
  DLNA packet contains the real IP address of DLNA device in the payload. Therefore, DA first converts the real IP address to the associated virtual IP address and then, DA forwards the packet from DMS to DMP.

In addition, as a common function to both DMP and DA, M-SEARCH Request, which is a message requesting for device searching within HNW, is defined.

C. Communication sequence

1) Device searching (discovery): Fig. 4 shows the sequence of device discovery. When DLNA Application if DMP is activated, M-SEARCH message is sent. DMP hooks this message by using NTM kernel module and sends a NTM query to DC_{DA}. DMP, upon receipt of NTM record from DC_{DA}, creates a tunnel to get connected with DA existing in HNW. DMP sends an M-SEARCH Request to DA through this tunnel. DA, upon receipt of the M-SEARCH Request from DMP, searches DMS in HNW on behalf of DMP. If DMS exists in HNW, a 200 OK message is sent back, and DA can find DMS. DA assigns a virtual IP address (VIP_{DMS}) to the real IP address (RIP_{DMS}) of DMS, which is described in the received 200 OK message.

Thereafter, DA changes the destination of the 200 OK message to VIP_{DMS} and the source address to a virtual IP address of DA (VIP_{DA}). Furthermore, DA replaces RIP_{DMS}

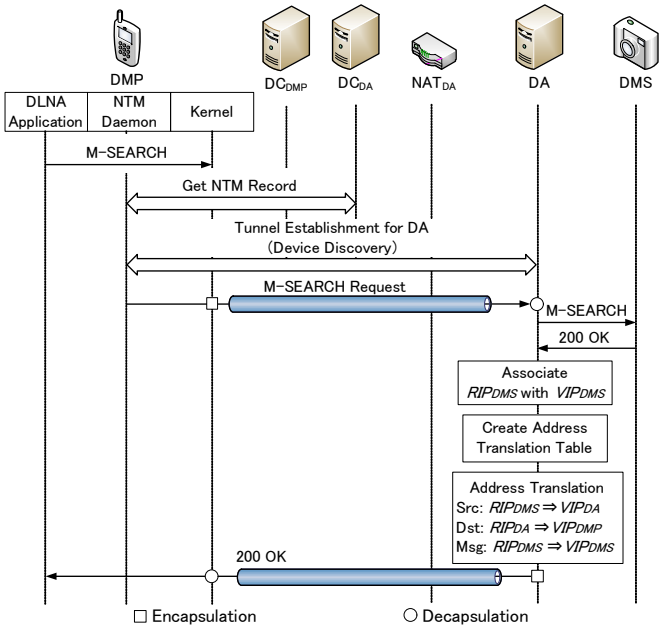

Fig. 4. Sequence of device discovery.

in the payload with VIP_{DMS}, and forwards the packet to DMP through UDP tunnel. In this way, DMP recognizes DMS virtually, and sends thereafter all DLNA-related packets to VIP_{DMS}. DMP also memorizes that the device managing VIP_{DMS} is DA.

2) Acquisition of device information: Fig. 5 shows the acquisition sequence of device information. DMP sends a HTTP GET request to VIP_{DMS}, which was acquired through device searching. In NTMobile, a UDP tunnel is created for each communication partner recognized by the virtual IP address. As the tunnel used to send M-SEARCH Request is a tunnel created to get connection with DA (VIP_{DA}), DMP is supposed to create another tunnel to get connection with DMS (VIP_{DMS}) to send packets to DMS, in accordance with the ordinary NTMobile mechanism. However, as VIP_{DMS} is managed by DA, the actual partner with which a tunnel is created is DA.

DMP sends a packet addressed to DMS, to DA, by using the tunnel for data transfer. On this occasion, the destination of the packet sent by DMP is VIP_{DMS}, the IP address of the DMS in the payload also is VIP_{DMS}. Thus, DA, after decapsulating the packet from DMP, changes VIP_{DMS} to corresponding RIP_{DMS}. In addition, DA relays this packet to DMS after converting the source address of the packet from VIP_{DMP} to its own real IP address (RIP_{DA}). Through the above process, DMS recognizes that the communication partner is the DA in the same network, and DMP can get access to DMS via DA.

In the case of a response from the DMS, DA follows a reverse process of the above. Namely, DA changes the source address of the packet from RIP_{DMS} to VIP_{DMS} and the destination from RIP_{DA} to VIP_{DMP}.

As DMP and DA are communicating using NTMobile, even

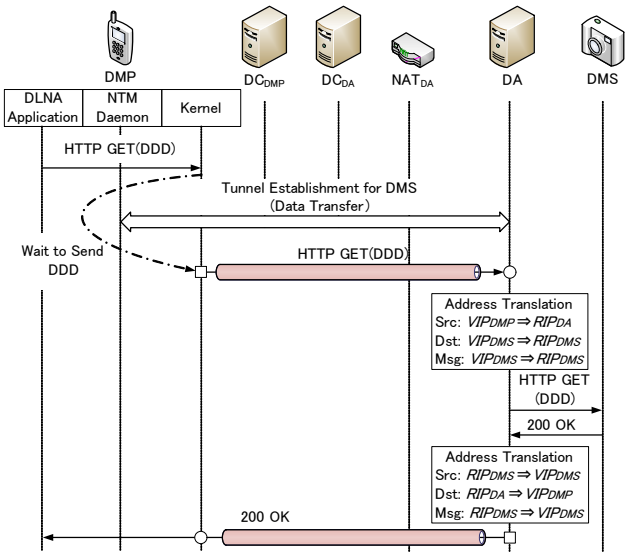

Fig. 5. Acquisition sequence of device information.

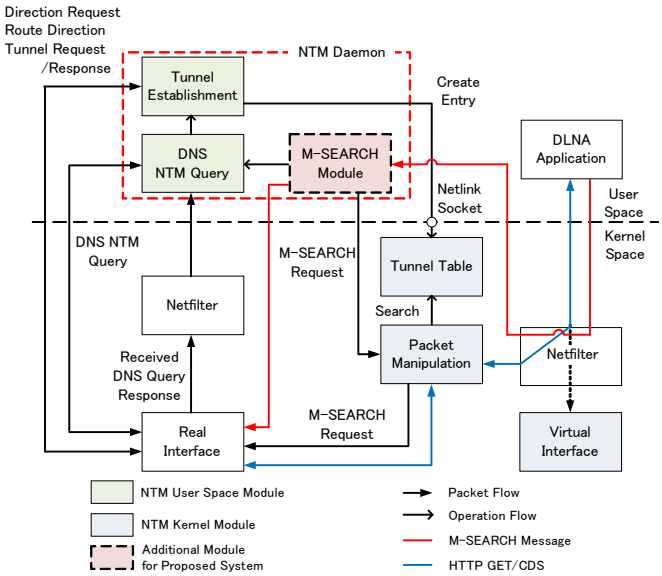

Fig. 6. Module configuration of DMP.

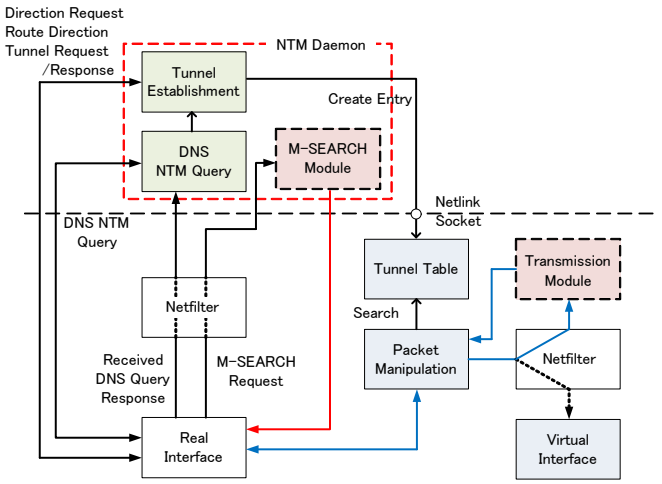

Fig. 7. Module configuration of DA

if IP address changes due to the reason that DMP switches networks during communication, a tunnel is recreated, and connection is maintained. Accordingly, DMP can freely move while browsing contents without interruption of communication.

IV. Implementation and Evaluation

We implemented our proposed method in a prototype system using Linux PC and verified its operation. In this section, we show the outline of our implementation and the result of the operation verification.

A. Implementation

We explain the implementation of DMP and DA as below. We omits explanation about DC and RS as functions of these are not expanded.

1) DMP: Fig. 6 shows the module configuration of DMP. DMP's NTM kernel module hooks M-SEARCH messages by using Netfilter and passes the message to NTM daemon.

NTM daemon, upon receipt of the M-SEARCH message from NTM kernel module, adds a module to create a tunnel get connection with DA. This module is also equipped with a function to send M-SEARCH Request message to DA.

2) DA: Fig. 7 shows the module configuration of DA. DA's NTM kernel module hooks M-SEARCH Request message from DMP by using Netfilter. This M-SEARCH Request message is passed onto NTM daemon. NTM daemon, after receiving the M-SEARCH Request message, sends the M-SEARCH message to HNW. NTM kernel module is also equipped with a function to translate the address of the packet between DMP and DMS and forward it.

B. Evaluation

Fig. 8 shows the network configuration of the trial system. DMP exists in a foreign network away from home and performs DLNA communication with DMS within HNW. Here, it is assumed that a private IP address is to be assigned, because we are going to apply our proposed method to Android smartphone, and in that case, DMP is connected to the network by 3G while you away from home. We assumed a situation where you come back home while doing remote DLNA communication and switched connection to Wi-Fi in HNW. Each DC and RS is operating as virtual machine within VMware ESXi 4.1. In our experiment of this time, in order to verify NTMobile's operation and clarify the overhead inherent to NTMobile, we created our experiment system in the form of a local network so that we can neglect the influence of communication delays of the real network as much as possible. round trip times (RTT) among device are shown in Fig. 8.

Based on the result of operation verification, we confirmed that DMP recognizes the DMS within HNW and that we are able to browse the contents held in DMS.

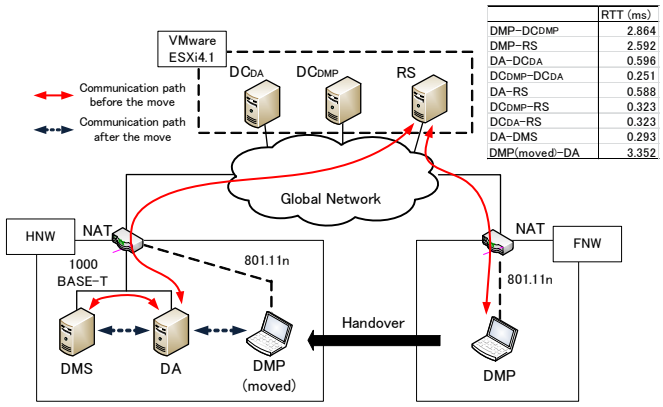

Fig. 8. Virtual network configuration for validation.

TABLE I
PROCESSING TIME OF OUR PROPOSED METHOD.

	Time (ms)
Tunnel establishment for device discovery	21.96
Device discovery	6.86
Tunnel establishment for data transfer	10.12
Acquisition of device information	8.36

We measured processing overhead of our proposed method and communication interruption time during movement in the above-mentioned verification. Table I shows the time measured for each step. It took 21.96 ms to create a tunnel to send a request to search for the most appropriate device, and 6.86 ms for the DMP to acquire the search result. It took 10.12 ms to create a tunnel to transfer data to DMS, and 8.36 ms thereafter to the point of acquiring device information. Through the above-mentioned process, application of DMP recognizes DMS, and the name of DMS is displayed on the screen of DMP.

If you take into account the employment of the system in a real network, you need to add to the above results the real round trip time between nodes, but the time is as small as several tens to several hundreds of milliseconds. Thus, we can say that the overhead by our proposed method is not so much as to give any influence on the browsing of the contents.

Table II shows the time required for access switching and tunnel recreation at the time when DMP moves from a foreign network to the home network to the home network. It took 6.91 seconds to get connected to the access point, and 0.08 seconds to recreation for data transfer to DMS. In the operation verification of this time, as DMP was switched by Wi-Fi, reproduction of the content was interrupted temporarily, but it was resumed soon after the completion of the switching. In the meantime, the time required for tunnel recreation is quite short compared with the network switching time. Thus, we think it necessary for DMP to realize a seamless handover by efficiently utilizing multiple communication interfaces such as a combination of 3G and Wi-Fi.

TABLE II
PROCESSING TIME IN THE CASE OF DEVICE MOVING

	Time (sec)
Connection to the Access Point	6.91
Tunnel recreation	0.08

V. CONCLUSION

In this paper, we proposed a DLNA-based, remote communication system that enables access from a DMP to a general DMS in the HNW, by expanding functions of our NTMobile that realizes mobility and solves the NAT traversal problem simultaneously. Based on the result of implementing our proposed method in a prototype system for operation verification, we confirmed that our proposed method cause practically no problem, as the overhead (delay in operational time) associated with the system is within a negligible range. It was demonstrated that our proposed method is practically usable as the delay in the operation time turned out to be within a negligible range. As the next step, we will further verify the operability of the system and performance evaluation by implementing our proposed method to Android smartphone.

REFERENCES

[1] Consumer Home. http://www.dlna.org/.
[2] Goland, Y. Y., Cai, T., Leach, P., GU, Y. and Albright, S.: Simple Service Discovery Protocol/1.0 Operating without an Arbiter <draft-cai-ssdp-v1-03.txt> , Internet draft, IETF (1999).
[3] Motegi, S., et al.: *Proposal on Wide Area DLNA Communication System*. Proc. of IEEE CCNC2008, pp. 233–237 (2008).
[4] Haruyama, T., Mizuno, S., Kawashima, M. and Mizuno, O.: Dial-to-Connect VPN System for Remote DLNA Communication, *Proc. of IEEE CCNC2008*, pp. 1224–1225 (2008).
[5] Doi, T., et al.: *Studies on RS in NTMobile*. DICOMO 2012, Vol. 2012, No. 1, pp. 1162–1168 (2012).
[6] Kamienoo, K., et al.: *Implementation and Evaluation of NTMobile that realize IP Mobility in IPv4 and IPv6 Networks*. DICOMO 2012, Vol. 2012, No. 1, pp. 1169–1179 (2012).
[7] Nishio, T., et al.: *Implementation od seamless IPv4/IPv6 address management for NTMobile*. DICOMO 2012, Vol. 2012, No. 1, pp. 1180–1186 (2012).