무선 ATM 네트워크에서 빠르고 매끄러운 핸드오프 제어

고 재 영*

요 약

제안한 분산 앵커(Anchor) CX 탐색 알고리즘은 무선 ATM 네트워크에서 빠르고 매끄러운 핸드오프를 지원해 주는데 중요한 역할을 한다. 본 방법은 그룹핑 한 네트워크 내에서 각 그룹은 Anchor 스위치가 그룹의 연결을 관리하고, Anchor과 Anchor 사이에는 정보교환을 위하여 작은 대역폭의 PVC를 할당한다. 제안한 알고리즘은 빠르게 CX를 찾을 수 있고, 전체 네트워크 관리가 용이하며, 시스템 오버헤드나 전달지연시간이 줄어 매끄러운 핸드오프를 지원 할 수 있다.

Fast and Seamless Handoff Control in Wireless ATM Networks

Jae Young Koh*

ABSTRACT

We propose a distributed anchor Crossover Switches (CX) searching algorithm to play an important role in ensuring fast and seamless handoff control in wireless ATM networks. Within networks that are grouped together, connection management is done for each group by anchor switches, and Permanent Virtual Circuit (PVC) with a narrow bandwidth is assigned between anchors for exchange of information. The proposed algorithm enables quick searching of a targeted CX, makes management of the overall network easier, and reduces system overhead or propagation delay time, thus providing fast and seamless handoff.

Key words : Wireless Network Management, Handoff Control

* 국가보안기술연구소 책임연구원

1. Introduction

The development of both Medium Access Control (MAC) protocols and effective and seamless handoff approaches is currently of critical concern to the emerging wireless ATM networks. In efforts to provide an effective handoff for wired and wireless networks, various approaches have been presented and a lot of research is underway. The handoff control algorithm in wireless/mobile networks must meet the general requirements for the existing cellular networks as well as ATM's unique requirements. Since wireless ATMs, unlike the existing cellular networks, focus on providing data application services, they should not only prevent the cells from being lost and duplicated but also ensure sequences during handoff.

The method for virtual connection during handoff is divided into path extension and path rerouting. Additionally, it is broken down into backward/forward handoff, depending on whether either the existing base station or a new one calls for handoff [5].

In order to ensure an optimal path during path rerouting, a CX needs to be dynamically selected for each connection to a mobile terminal [4]. This dynamic process of selecting crossover switches has to be initiated by an interrupt of the originating side that has called for connection. Until an optimal CX is searched, an algorithm for selecting crossover switches should be implemented for every switch on the path for virtual connection. Therefore, all switches within the network should support this CX selection algorithm. The remoter the crossover switches selected by the algorithm become from the existing base station, the more the cells must be transferred between the existing base station and the crossover switches. This means longer delay time in switching avirtual path connection or much cell loss during handoff. CX searching is mainly intended to analogize the minimum hop count path to the CX for VC, which is configured by BS_{new} and currently in use. The selected CX will affect the shorter resultant path as well as circuit reuse.

Among currently available CX search algorithms, the Prior Path Knowledge CX search algorithm adopts a centralized approach to network connection management where complete topology information is required during routing. The Distributed Hunt CX search algorithm adopts a distributed approach to network connection management where Link-State or Distance-Vector routing is used [1].

The Prior Path Knowledge CX search algorithm has the advantage of reducing overall system overhead where a connection server performs overall network connection management. However, it causes longer propagation delays in the process of updating the connection information for each node.

The Distributed Hunt CX search algorithm that uses a distributed approach in network connection management has the advantage of causing no propagation delay where each node has connection management databases. However, it increases costs to maintain connection management databases for each node as well as overall system overhead resulting from the fact that connection configuration is made by each node.

For the Prior Path CX search algorithm that uses a centralized approach to network connection management, the role of a connection server is important. BS_{new} calls on the connection server to search for every possible convergence node, finding the minimum hop path among every possible crossover switch from BS_{new} to the convergence node. If multiple CXs are searched, a node closest to CLS_{old} should be selected as CX. Once a targeted CX is selected, BS_{new} requests the connection server to create a partial path for the CX.

The Distributed Hunt CX search algorithm is used to find the minimum hop by allowing each node to voluntarily search for every possible convergence without any connection server. By maintaining the Local Connectivity Table (LCT), each node continues to update the information about connection setup and release for itself and its neighboring nodes. BS_{new} identifies the LCT to make sure whether CLS_{new} is a CX or not. If CLS_{new} isn't a CX, it broadcasts CX search packets via CLS_{new} to all switches within the network before waiting for responses. After a limited period of time awaiting responses, BS_{new} performs a mapping of every possible CX, selecting among them a node with a minimum hop as CX. If multiple nodes are selected that have the same min. hop count, a random node is selected as CX. Finally, if a CX is configured and BS_{new} receives a control packet to get ACK, a partial path is created.

For the Backward Tracking CX search algorithm [2, 3], BS_{old} sends a back-track search packet to the predecessor (firstly CLS_{old}). The predecessor makes inquires of BS_{new} about the routing table to find the next hop node. As a result of making inquires of the connection server, if both the predecessor and the next hop node are included in the CLS_{old} and CLS_{dest}, the back-track packet is forwarded to the next predecessor. Otherwise, the node is selected as CX.

2. Proposed CX Searching Algorithm

All nodes in an ATM network are grouped together by a specific number of nodes. The anchor switch refers to the node with the largest degree among nodes in each group. That's because the node with the largest degree in each group has higher probability of being selected as CX. The network model is designed in such a way that an anchor switch in each group is interconnected with anchors in other groups. To ensure information exchange between anchors, resources with a narrow bandwidth are assigned using a Permanent Virtual Circuit (PVC), which allows you to search for CXs quickly.

Anchor switches continue to update new information while maintaining the Group Connectivity Table (GCT) that includes their present connection IDs, connection IDs for all nodes in the group, and the information about the adjacent groups connected with this group via PVCs. Each node does not maintain the LCT. The anchor in each group continues to monitor and update the connection status of every node in each group and of every anchor node that is connected by PVC. The process of searching for CXs in the algorithm proposed here is as follows :

Step 1 : The anchor in each group makes inquiries about the GCT. If the position towhich the anchor has moved is searched for the switch just adjacent to the present Base Station $(BS)_{old}$ within the group, the anchor directs path extension. Otherwise, BS_{new} performs the next step to reset paths.

Step 2 : If BS_{new}'s cluster is the anchor of a group, BS_{new} sends a search packet to an adjacent anchor reset by a PVC to request it to reset the VC. If BS_{new}'s cluster isn't the anchor of a group, BS_{new} sends a search packet to the anchor of the group and the anchor sends a search packet to an adjacent anchor reset by a PVC.

Step 3 : The anchor of each group knows the connection information about all nodes of the group. Upon receipt of a search packet, the anchor of a group makes inquiries about the GCT of the sending anchor. If a corresponding node exists, the anchor stops sending a packet while sending an ACK signal to BS_{new}. It repeats this process to allow all nodes of the already set VC to perform this algorithm.

Step 4 : In comparison with the hop counts sent by multiple anchors for a limited period of time, the node with a minimum hop count is selected as CX.

Step 5 : If multiple CXs are searched, a random CX is selected.

Step 6 : If a CX is selected, a VC from the CX to BS_{new} is set, and a VC from BS_{old} to the CX is released.

The following describes the proposed Distributed Anchor CX Discovery algorithm :

Distributed Anchor CX Discovery Algorithm

1. G = (V, E) represents the ATM backbone network and H = (V, E) represents the anchor switch within an ATM network (i.e.$\subset$G).
2. 'O'refers to the currently available path : $CLS_{old} \rightarrow CLS_{dest}$, and 'A' refers to the path connected by an anchor.
3. 'O' is 'G's subgraph, while 'A' is 'O's subgraph. $V(A) = \{O_1$ or $A_1 = CLS_{old}, A_2, \cdots, O_y$, or $A_z = CLS_{dest}\}$, where $z < y$.

Begin

4. Obtain the node information for the discovered path : $CLS_{old} \rightarrow CLS_{dest}$.
If $((CLS_{new} \rightarrow \in A_i) \cap (CLS_{old} \in A_i))$,
then CX = CLS_{old} /*Anchor A_i leading to path extension*/ goto done :

else Begin

5. Respective $A_i \in V(A)$, where $\{i = 1, 2, ,\cdots, z\}$,
6. Compute the minimum hop count for $CLS_{new} \rightarrow A_i$.
7. This path is referred to as 'M_i'.
If $(M_i \neq 0)$,
Begin /*Anchor A_i leading to path rerouting */
Compute A_x.
Where M_x = Shortest $\{M_i\}$.

Accordingly, CX = A_x. /*Anchor becomes a CX.*/
or
If there exist many paths : $CLS_{new} \rightarrow$ CX
Begin where $(| M_x | = | M_y | = \cdots = | M_z |)$
If Shortest$\{M_i\} = \{M_x, M_y, \cdots, M_z\}$,
Select a random CX. (i. e. CX = O_y)
End
End
else
Begin
G is partitioned (i.e.CLS_{new} cannot have access to CX.)
End
End
done :
End

Using the notion of anchor, a PVC with a narrow bandwidth is set for exchange of information between anchors, which allows for faster and more efficient CX searching, lowers system overhead and additional costs caused by the Distributed Hunt CX searching algorithm, and significantly reduces propagation delay time between connection servers and nodes caused by the Prior Path Knowledge CX searching algorithm.

Since the anchor determines path extension or path rerouting, it can reduce problems arising during path rerouting by leading to path extension between the just adjacent clusters. In addition, the anchor can resolve one of the drawbacks occurring during path extension, so-called the routing Loop problem, through control of each node and faster alternative routing.

In term of PVC assignment, the proposed scheme based on the PVC assigned only between anchor nodes can reduce a waste of bandwidth characterized by the VCT scheme as well as eliminate an additional procedure scheme for setting a PVC, which is required by the in the SMRC scheme when moving to another region.

3. Performance Analysis

A simulation is conducted to analyze the performance of the proposed algorithm. Firstly, a network model is designed to evaluate the proposed algorithm. Using this model, the proposed algorithm is evaluated in comparison with the existing algorithms in terms of convergence characteristics, resultant path characteristics, and circuit reuse. The cluster concept has been used in the network-level handoff, while the model has been designed using a higher-level network concept.

In order to ensure quick CX searching, a network model that adopts the concept 'Anchor' was used in grouped networks. An anchor was defined as the node with the largest degree among nodes in each group. Using PVCs, some bandwidth was assigned between anchors and their adjacent anchors. The anchor in a group was linked to every node of its group with whatever paths it may choose. One hundred numbers of the node (ATM switch) were created for use in the performance test. One hundred numbers of the node were used to compare convergence characteristics. In comparing resultant path characteristics and reuse rate, a network model was designed for performance analysis with the number of the node assigned 30, 50, and 100, respectively. The network model used for simulation has set one group by 5 to 6 nodes, with the average degree set by 6 to 7 anchor nodes and 3 to 4 normal nodes. Assuming that an inter-cluster handoff occurred, mobile terminal's handoff regions were taken into account to ensure path rerouting, not path extension. In order to find the minimum hop, the well-known Dijkstra's Algorithm was used, which can be simply expressed as follows:

Where the path from the vertex 'V0' to the intermediate vertex 'u' is decided as distance [u], and a path from 'u' to 'w' is to be selected, if distance [u], it can be expressed as follows : distance [w] Where distance [u] is the minimum distance value selected from the yet-to-be searched vertex 'V0'). The same weights were assigned to links to ensure fairness. Differentiated simulations were made to the node groups that were not connected. The connectivity between nodes is determined by the following probability :

e : Link degree,

d (x, y) : Euclidean distance between the nodes 'x' and 'y',

α : ($\alpha > 0$), Parameter that controls the number of connection for remote nodes,

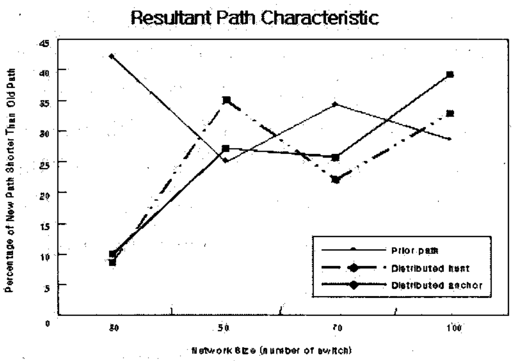
β : ($\beta \leq 1$), Parameter that controls the edge number of each node,

n : Number of two nodes,
L : Largest distance between the two nodes.

In order to the performance of the proposed algorithm, a comparative analysis was performed according to the following items :

Convergence characteristics : Focus was put on how to quickly search for a target CX. A factor that needs consideration is the number of implementation required when searching for atarget CX. Ultimately, this characteristics can be broken down toreduced handoff latencies by way of performing new path routing in search of a CX with the minimum hop count and the smaller number of implementation. Where the number of the node is 100, the following are the convergence characteristics obtained from a comparative analysis of respective algorithms.

(Fig. 1) shows a comparison of the algorithm proposing resultant path characteristics, Prior Path Knowledge CX search algorithm, and Distributed Hunt CX search algorithm.

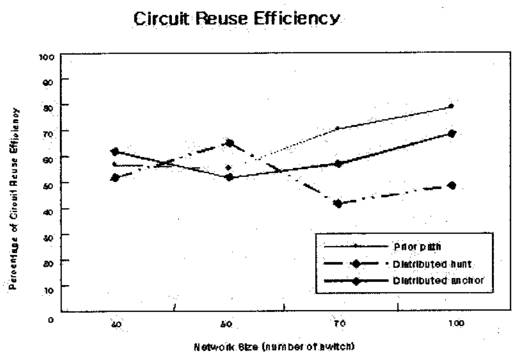


(Fig. 1) Comparison of resultant path characteristics

According to the figure, the existing Distributed Hunt CX Search Algorithm shows the increased probability of a new path being shorter than a previous one as the size of networks increases. On the contrary, the Prior Path Knowledge CX Search Algorithm shows the decreased probability of a new path being shorter than a previous one as the size of networks increases.

This is due to the method for network connection management. For the Prior Path Knowledge CX Search Algorithm, a connection server performs connection management of the overall network, leading to efficient network management in small-scale networks. However, it achieves lower performance due to propagation delays as the size of networks becomes larger.

The algorithm proposed here continues to achieve improved resultant path characteristics as the size of networks increases, showing better performance than other existing algorithms starting from 80 numbers of nodes. This shows that the proposed algorithm achieves better point-to-point delay characteristics or overall link utilization with the increased size of a network.



(Fig. 2) Comparison of Circuit Reuse Efficiency

(Fig. 2) shows a comparison of the algorithm proposing circuit reuse efficiency, Prior Path

Knowledge CX Search Algorithm, and Distributed Hunt CX Search Algorithm. Circuit reuse efficiency is a function of how frequently the previous paths are used. According to this figure, as the size of a network increases, the Prior Path Knowledge CX Search Algorithm shows an increase in circuit reuse efficiency, while the Distributed Hunt CX Search Algorithm shows a reduction in circuit reuse efficiency.

The lower circuit reuse efficiency of the Distributed Hunt CX Search Algorithm is attributable to the search algorithm implemented by every node, which lowers circuit reuse efficiency with the increased size of a network because there are a lot of remotely positioned nodes from the current VC.

The proposed algorithm shows a result similar to the Prior Path Knowledge CX Search Algorithm's by demonstrating improved circuit reuse efficiency with the increased size of a network, ensuringdata integrity against the possibility of data being lost or sequence maintenance.

4. Conclusion

In this paper, the Prior Path Knowledge CX Search Algorithm and the Distributed Hunt CX Search Algorithm are described as representative algorithms. The centralized approach to network connection management poses problems in that one connection server performs the connection status of the overall network, thus leading to much longer propagation delay time. Unlike this, the distributed approach to network connection management causes overall network system overhead because every node implements a search algorithm.

In order to improve these problems, the Distributed Anchor CX search algorithm uses the concept 'Grouping' to define an anchor of a group where the anchor performs connection management of the group. In addition, PVCs are connected between anchors, allowing more reduced waste of bandwidth and easier prediction of mobility than the existing PVC assignment schemes. The use of this algorithm is expected to enable faster CX searching, easier management of the entire network, and reduced system overhead or propagation delay time, thus providing faster and seamless handoff.

References

[1] Chai-Keong Toh, "Performance Evaluation of Cross Switch Discovery Algorithms for Wireless ATM LANs", IEEE INFOCOM '02, 2002.

[2] Oliver T. W. YU, Victor C. M. Leung, "Connection Architecture and protocols to support Efficient handoffs over an ATM/B-ISDN personal communication network", Mobil Network And Application 1, Vol. 7, No. 2, 2004.

[3] Anthony S. Acampora, Mahmoud Naghshineh, "An Architecture and Methodology for Mobile-Executed Handoff in Cellular ATM Networks", IEEE Trans. on selected area in Communication. Vol. 17, No. 8. Oct. 2004.

[4] Chai-Keong Toh, "The Design & Implementation of a Hybrid Handover Protocol for MultiMedia Wireless LANs", ACM First International Conference on Mobile Computing & Networking, 1995.

[5] Bora A. Akyol, Donald C.COX, “Rerouting for Handoff in a Wireless ATM network”, IEEE, Personal Comm. Magazine, 1996.

[6] Duk Kyung Kim, Dan Keun Sung, “Handoff/Resource Managements Based on PVCs and SVCs in Broadband Personal Communication Networks”, IEEE. 2001.

Jae Young Koh

1984년 B.S. (Electronics) Chonbuk National University

1992년 M.S. (Electronics) Chonbuk National University

1998년 Ph.D. (Electronics & Telecom) Chonbuk National Univ.

1984년~2000년 Senior Researcher, Team Leader, ADD

2000년~2001년 Principal Member of Eng. Staff, Team Leader, NSRI(National Security Research Institute)

2002년~2005년 Principal Member of Eng. Staff, Director, NSRI

2004년~2005년 Secretary General, NISA(National Information Security Alliance)

2006년~Now Principal Member of Eng. Staff, NSRI