

# 계층셀 구조에서 이동 속도와 대역폭을 고려한 복합 호 처리 알고리즘

## (A Combined Call Control Algorithm based on Moving Speed and Bandwidth in Hierarchical Cellular Systems)

박 정 은 <sup>\*</sup>    채 기 준 <sup>\*\*</sup>

(Jeong Eun Park) (Kijoon Chae)

**요 약** 이동 통신 수요의 급격한 증가와 멀티미디어 서비스에 대한 요구에 의해 기존의 셀룰러 시스템은 용량 부족 및 QoS 처리에 있어서 한계에 부딪히게 되었다. 따라서 사용자들이 요구하는 QoS를 만족시켜줌과 동시에 최대한 많은 사용자를 수용하기 위해 계층셀 구조가 제안되었다. IMT-2000에서는 셀 구성 방안으로 계층셀 구조를 채택하였는데, 그에 따라 복잡한 중첩 구조를 간단하면서도 효율적으로 이용할 수 있는 호 처리 알고리즘의 필요성이 대두되었다. 본 연구에서는 기존에 제안된 복합 호 처리 방식에서 한 걸음 나아가 이동 속도와 대역폭을 고려한 호 처리 알고리즘을 제안하였다. 제안한 알고리즘에서는 버퍼와 가드 채널을 이용하여 호의 실패율을 낮추고 호의 이동 속도와 대역폭을 고려하여 기존의 방식보다 효율성을 높일 수 있도록 하였다. 또한 대역폭과 이동 속도를 고려하여 호들을 분배하여 처리함으로써 사용자의 QoS를 만족시켜주고 핸드오버율을 줄이고자 하였다. 제안한 알고리즘의 성능을 평가하기 위해서 본 논문에서는 계층 셀룰러 시스템을 구성하고 다양한 트래픽을 생성하여 시뮬레이션을 수행하였다. 블록킹률, 드로핑률, 채널 활용률, 계층간 핸드오버 수 등 다양한 파라미터를 토대로 기존의 방식과 제안한 방식을 비교해봄으로써 성능 향상을 입증하였다.

**키워드** : IMT-2000, 계층셀, 핸드오버, 호수락제어

**Abstract** Recently, the demand for mobile communications and multimedia services has rapidly increased so that conventional cellular system cannot fulfill the requirement of users (capacity and QoS) any more. Therefore, the hierarchical cellular system has been suggested in order to guarantee the QoS and to admit large population of users. IMT-2000 adopts the hierarchical cellular structure, which requires a call control algorithm capable of manipulating and utilizing the complicated structure of hierarchical cellular structure with handiness and efficiency.

In this thesis, as an improvement of conventional combined algorithm, a new call control algorithm considering the moving speed of terminal and bandwidth is suggested. This algorithm employs buffers and guard channels to reduce the failure rate. Also, this algorithm considers the moving speed of terminal and bandwidth to elevate the efficiency. Furthermore, calls are handled separately according to the moving speeds of terminal and bandwidths to improve the QoS and reduce the handover rate.

As an evaluation of the suggested algorithm, a model hierarchical cellular system is constructed and simulations are conducted with various types of traffic. As the result of the simulations, such indices as block rate, drop rate, channel utilization, and the number of inter layer handovers are examined to demonstrate the excellency of the suggested algorithm.

**Key words** : IMT-2000, Hierarchical Cell, Handover, Call Admission Control

<sup>\*</sup> 비 회 원 : 이화여자대학교 컴퓨터학과

aroa@samsung.co.kr

<sup>\*\*</sup> 종 식 회 원 : 이화여자대학교 컴퓨터학과 교수

kjchae@ewha.ac.kr

논문접수 : 2001년 10월 26일

심사완료 : 2002년 3월 15일

### 1. 서 론

IMT-2000의 상용화를 앞두고 최근 개인 휴대 통신에 대한 관심이 부쩍 증가하였다. IMT-2000은 언제나, 어디서나, 누구에게나 오디오, 데이터, 팩스, 동영상

등과 같은 멀티미디어 서비스를 제공한다는 취지로 곧 서비스를 실시할 예정이다. 하지만 기존의 매크로 셀 기반의 셀 구조에서는 용량의 한계로 인해 많은 수의 사용자와 대용량의 데이터를 처리할 수 없다. 또한 멀티미디어 등 다양한 서비스를 요구하는 사용자를 만족시켜 주기 어렵다. 따라서 이를 해결하기 위해 반경이 다른 여러 종류의 셀을 중첩하여 구성하는 계층셀 개념이 등장하였다.

계층셀 구조에서는 한 지역이 여러 계층셀의 범주에 들기 때문에 그 지역에서 발생하는 호는 여러 계층의 기지국에서 처리해 줄 수가 있다. 따라서 호의 실패율을 줄일 수 있고 용도에 따라 다른 셀을 이용하여 사용자가 요구하는 QoS를 만족시켜 줄 수 있다. 하지만 계층셀 구조는 복잡한 구조로 인해 복잡한 호 처리 알고리즘을 요구한다. 이는 시스템 복잡성을 증가시키게 되는데 이를 감소시키면서 효율적으로 호를 처리하기 위한 알고리즘이 필요하게 되었다.

계층셀 구조가 제안된 이후 지금까지 꾸준히 계층셀 호 처리 알고리즘이 제안되어 왔다. 이 알고리즘들은 호의 실패율을 줄이고, 사용자의 QoS를 만족시키며, 시스템을 최대한 이용하기 위한 방안들로 소개되었다. 한 계층의 채널이 고갈되면 중첩된 다른 계층의 셀로 호를 넘겨주는 기법과, 호의 블로킹률과 드로퍼율을 줄이기 위한 임시 저장 공간인 버퍼를 사용하는 기법, 핸드오버 호를 보호하기 위해 사용되는 가드 채널 기법 등이 그 예이다. 이렇게 제안된 다양한 기법들을 통합한 방법이 복합 호 처리 방식이다.

기존의 복합 호 처리 방식은 다양한 기법들을 함께 사용함으로써 성능 향상을 입증하였는데 여기서는 호들이 모두 일정한 대역폭을 가지고 호가 종료될 때까지 단말기의 대역폭이 변하지 않는다고 가정하여 성능을 측정하였다. 하지만 실제 셀룰러 시스템에서는 사용자마다 요구하는 대역폭이 다르며, 여러 가지 요인에 따라 대역폭이 변할 수 있다. 또한 기존의 방식에서는 호의 이동 속도와 대역폭을 고려하지 않고 있다. 하지만, 계층셀 구조를 보다 효과적으로 이용하기 위해서는 호의 이동 속도와 대역폭에 따라 다른 계층에서 처리해주는 알고리즘이 필요하다. 따라서 본 연구에서는 대역폭과 이동 속도의 다양성을 고려하여 별도로 처리해주는 새로운 복합 호 처리 알고리즘을 제안하였다. 또한 기존의 알고리즘에 비해 성능 향상을 가져옴을 입증하였다.

본 논문의 구성은 다음과 같다. 1장의 서론에 이어서 2장에서는 기존의 계층셀 구성 방안에 대해 연구된 내용들을 살펴보고, 3장과 4장에서는 기존의 다양한 계층

셀 호 처리 알고리즘에 대해 살펴본다. 5장에서는 제안한 알고리즘에 대해 소개하고, 6장에서는 수현한 시뮬레이션에 대해 설명하고 결과를 분석한다. 마지막으로 7장에서는 결론을 맺고 향후 연구 과제를 제시한다.

## 2. 계층셀 개요와 구성

앞서 설명한 바와 같이 계층셀 구성은 기존의 셀룰러 시스템의 과부하를 막고, 좀 더 효율적으로 호를 처리하기 위해 제안되었다. 계층셀은 논리적인 개념의 구조로서 같은 지역에 여러 계층의 셀이 중첩된(overlaid)구조로 되어있다. 계층셀을 실제 어떻게 구성하느냐는 사업자나 환경에 따라 조금씩 다르겠지만, IMT-2000에서는 피코 셀(picocell), 마이크로 셀(microcell), 매크로 셀(macrocell), 그리고 메가 셀(위성 셀, megacell)로 구성하는 것을 권고하고 있다. 각 셀의 타입별 파라미터는 표 1에서 정리하였다. 가장 작은 반경의 피코 셀은 작은 사무실 등을 위한 셀로 2Mbps까지의 대역폭을 제공할 수 있고, 마이크로 셀과 매크로 셀은 일반적으로 보행자나 차량 이동 사용자를 위한 셀이다. 가장 큰 반경의 매크로 셀은 위성을 이용한 셀로 위성셀이라고도 불린다. 본 논문에서는 피코 셀과 메가 셀을 제외한 마이크로 셀과 매크로 셀을 그 대상으로 하여 실험하였다. 그 이유는 가장 보편적으로 사용되는 셀이 마이크로 셀과 매크로 셀이기 때문이다.

표 1 셀 타입별 파라미터의 예

Cell type	Mega cell	Macro cell	Micro cell	Pico cell
Cell radius	100-500km	≤ 35km	≤ 1km	≤ 50m
Installation	LEO/HEO/GSO (전 세계적인 로빙)	Top of building/tower, etc.	Lamp-post /building wa	Inside building
Terminal speed		≤ 500 km/h	≤ 100 km/h	≤ 10km/h
Data rate		≤ 144 Kbps	≤ 384 Kbps	≤ 2 Mbps

계층셀을 어떻게 구성하는지는 여러 가지 환경 요소를 판단하여 결정해야 하는데, 이에 따라 셀룰러 시스템의 성능이나 용량이 좌우되기도 하므로 매우 중요하다고 할 수 있다. 기존의 계층셀 셀룰러 시스템은 그림 1과 그림 2와 같이 일정한 크기의 셀을 반복시켜 구성하였다. 일정 개수의 작은 셀이 모여 하나의 큰 상위 계층셀을 구성하는 형식이었다. 이렇게 구성할 경우 셀룰러 시스템의 구성이 매우 단순하며 간편해진다는 장점이 있지만, 사용자가 거의 없는 지역도 셀로 구성하여 기지국을 설치해야하는 낭비가 초래될 수 있다. 따라서 새로 고안된 방식이 그림 3과 같은 핫 스팟(hot-spot)을 고

려한 방식이다. 이 방식은 사용자들이 밀집한 지역만을 따로 셀로 구성하여 과부하를 막고, 보다 질 좋은 서비스를 제공해주고자 하는 방식이다. 본 논문에서도 핫 스팟을 고려한 방식으로 계층셀을 구성하였다.

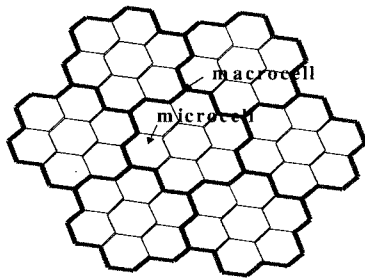


그림 1 마이크로 셀과 매크로 셀의 경계가 일치하는 경우

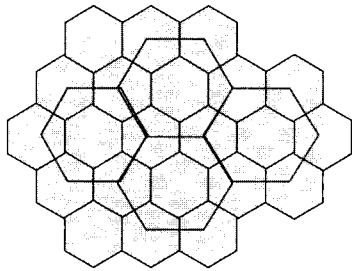


그림 2 마이크로 셀과 매크로 셀의 경계가 일치하지 않는 경우

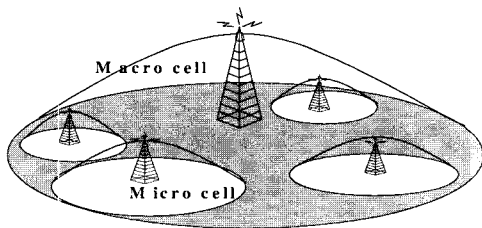


그림 3 핫 스팟(hot spot)을 이용한 계층셀 구성

### 3. 계층셀 호 처리 알고리즘

#### 3.1 개요

계층셀 구조는 단층셀 구조와 달리 반경을 달리하는 여러 계층의 셀이 중첩된 구조로 되어 있으므로 여러 가지 알고리즘이 복잡할 수밖에 없다. 따라서, 계층셀의 장점을 충분히 활용하기 위해서는 이에 적합한 호 처리 알고리즘이 필수적이다. 일반적인 호 처리 알고리즘과

계층셀에서의 호 처리 알고리즘의 큰 차이라면 계층셀 호 처리 알고리즘에서는 이동에 따른 셀간 핸드오버 뿐 아니라 계층간의 핸드오버도 처리해 주어야 한다는 것이다.

계층간의 핸드오버는 계층셀의 구조를 충분히 활용하기 위해서 수행하게 되는데 이동성이 적고 큰 대역폭을 요구하는 호는 작은 반경의 셀에서 처리해 주고, 이동성이 크고 작은 대역폭을 요구하는 호는 큰 반경의 셀에서 처리해 주도록 하는 것이다. 이렇게 함으로써 큰 대역폭을 요구하는 호에게는 대역폭을 만족시켜주고, 이동성이 큰 호의 핸드오버율을 줄여줌으로써 전체적인 시스템의 성능을 향상시킬 수 있다. 또한, 한 계층의 셀에서 호를 다 처리해 줄 수 없을 때 다른 계층으로 호를 넘겨주어 처리해 줌으로써 호의 블로킹(blocking)률과 드로핑(dropping)률을 줄일 수도 있다. 그림 4는 계층셀 호 처리 알고리즘을 간략히 나타내고 있는 그림이다.

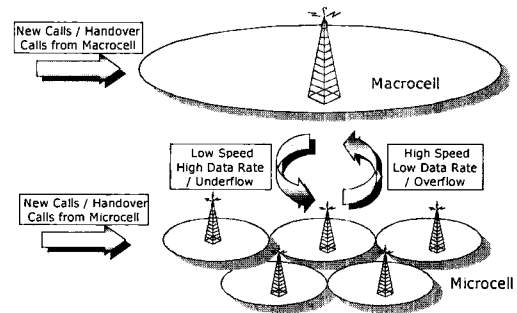


그림 4 계층셀 구조에서의 기본적인 호 처리 알고리즘

#### 3.2 계층셀 호 처리 알고리즘의 요구 사항

호 처리 알고리즘의 요구사항들은 계층셀 호 처리 알고리즘의 성능을 평가할 때도 적용될 수 있다. 아래에서 몇 가지 중요한 요구사항들을 나열해 보았다.

- **블로킹/드로핑률의 최소화** : 핸드오버나 호처리 알고리즘에 있어서 이 블로킹률과 드로핑률을 최소화하는 것은 가장 중요한 요구사항이다. 특히 드로핑률을 최소화하는 것이 중요한데, 그 이유는 통화중인 호가 갑자기 중단되었을 때 더 큰 불편을 느끼게 되기 때문이다. 따라서 드로핑률을 최소화하려는 노력은 꾸준히 진행되어 왔다.

- **핸드오버율의 최소화** : 핸드오버는 단말이 한 셀에서 다른 셀로 이동할 때 이동한 셀에서도 서비스를 끊어지지 않고 받고자 하는 과정인데, 이때에는 단말과 기지국간, 그리고 이전 기지국과 새 기지국간에 많은 신호

를 주고받는 복잡한 과정이 일어나게 된다. 따라서 단말이나 기지국의 입장에서 빈번한 핸드오버는 시그널링 트래픽을 증가시키게 되어 오버헤드를 초래한다. 또한 단말이나 기지국의 시스템에도 큰 부담을 줄 수 있다. 멀티미디어 서비스를 제공할 때도 잦은 핸드오버는 서비스의 질(QoS)을 보장하는 것을 어렵게 만든다.

• **채널 활용률의 최대화**: 채널 활용률은 한정된 수의 채널을 얼마나 효과적으로 여러 단말이 사용하는지를 의미한다. 셀룰러 시스템에서는 보통 한 지역을 논리적인 육각형의 셀로 나누어 인접하지 않은 셀들에서는 같은 주파수를 사용하더라도 간섭이 일어나지 않도록 하는 방법을 사용하여 채널의 활용률을 높이고 있다. 셀의 크기가 작으면 인접하는 셀들의 수가 적어지게 되어 채널 활용률은 높아지게 된다. 하지만 이에 반해 핸드오버율은 높아지게 되는데, 이 둘 간의 균형을 맞추는 것이 매우 중요하다. 또한 할당받은 채널을 사용하지 않은 채 낭비하게 되면 채널 활용률은 떨어지게 된다. 이는 채널 할당을 얼마나 효과적으로 하느냐에 달려있다.

• **단순성**: 복잡한 알고리즘은 복잡한 처리 과정을 요구한다. 호 처리 알고리즘도 마찬가지로 복잡한 알고리즘은 큰 시스템 부하를 초래하고 지연 시간도 길어지게 된다. 많은 핸드오버가 일어나는 시스템의 경우 복잡한 알고리즘에 의해 시스템에 장애가 발생할 수도 있고, 지연 시간이 길어지면 서비스를 받기 위해 기다려야 하는 시간까지 길어지므로 서비스의 질에 문제가 된다. 따라서 단순하면서도 효과적인 알고리즘일수록 효과적인 알고리즘이라 할 수 있다[1].

• **확장성**: 호 처리 알고리즘은 다양한 망 환경에서도 적용할 수 있어야 한다. 다양한 트래픽 상황이나 환경 상태에 대해서도 적용 가능해야 한다. 예를 들면, 시스템 로드가 극대화된 상황이나 보다 복잡한 요구사항에 대해서 확장 가능해야 한다[1]. 또한 PSTN, PDSN, Internet 등에서도 활용할 수 있어야 우수한 알고리즘이라고 할 수 있다. 또한 새로운 서비스에 대한 요구가 발생하였을 때에도 이를 수용하기 위한 처리가 쉬워야 한다. 특정 상황이나 환경에서만 적용 가능하다면 낮은 유연성으로 인해 적용 범위가 작아진다.

• **QoS 보장**: 사용자들이 요구하는 서비스의 질을 만족시켜 줄 수 있는 호 처리 알고리즘이어야 한다. 사용자나 서비스마다 요구하는 QoS는 다를 수 있다. 예를 들면, 지연없는 실시간 서비스를 요구하거나, 큰 대역폭을 요구하거나, 손실 없는 서비스를 요구할 수도 있다. 이럴 경우 기존 호의 QoS를 계속 만족시켜주면서 개개인이 요구하는 QoS를 최대한 만족시켜줄 수 있어야 한다[2].

### 3.3 호의 실패율을 줄이는 방법

#### 3.3.1 버퍼링 기법

전통적인 호 처리 방식에서는 채널 할당이 요청되었을 때 가용 채널이 없다면 호는 블로킹되거나 드로핑되어야 한다. 하지만, 호가 블로킹되거나 드로핑된 직후에 가용 채널이 발생할 경우도 종종 발생할 수 있다. 이 경우 호를 바로 블로킹이나 드로핑 시키지 않고 어느 정도 기다리게 함으로써 호의 실패율을 줄일 수 있다. 이 방법이 버퍼링 기법이다[3-6]. 이는 셀룰러 구조에서 셀들 간에 중첩되는 공간이 있기 때문에 가능하게 되는 것인데, 채널을 바로 할당받지 못했다고 해도 중첩된 구간에서는 어느 정도 이전의 신호를 받을 수 있기 때문이다.

버퍼를 어떻게 구성할지, 어떤 정책을 사용할지는 상황에 따라 매우 다양해질 수 있다. 버퍼를 선입선출(First In First Out) 방식으로 구현하게 되면 먼저 버퍼에 들어온 순서대로 차례로 채널을 할당받게 된다. 이는 간단하게 구현할 수 있으나 효율적이지 않은 방법이다. 좀 더 효율적으로 버퍼를 이용하기 위해서는 호의 우선 순위를 이용할 수 있다. 우선 순위는 신호의 세기로 할 수도 있고 이동성이나 QoS와 같은 호의 특성으로 할 수도 있다[7]. 선입선출 방식보다는 복잡하지만 효율적으로 채널을 할당할 수 있다.

버퍼의 크기에 따라서도 성능의 차이는 다양하게 나타난다. 버퍼는 호들을 임의로 지연시켜 강제 종료율을 낮추고자 하는 방법이다. 따라서 버퍼의 크기가 커지면 많은 호를 지연시킴으로써 호의 강제 종료율을 줄어줄 수 있지만, 지연 시간은 증가하게 된다. 따라서 적절한 수준의 버퍼 크기를 유지하여야 좋은 성능을 나타낼 수 있다.

버퍼의 정책은 버퍼링 기법에서 매우 중요한 부분이다. 여러 개의 버퍼를 사용하고 있을 때 정책 설정은 더욱 어려워진다. 여러 개의 버퍼간의 우선 순위를 어떻게 설정할 것이며 특정 버퍼가 채널 할당을 계속 받지 못하는 상황 등을 고려해야 한다. 보통 핸드오버 호의 우선 순위를 높게 주어 핸드오버 호에게 우선적으로 채널을 할당해주지만 이럴 경우 시작 호가 계속 채널을 할당받지 못할 수가 있다. 이를 대비하기 위해 핸드오버 호가 어느 정도 채널을 할당받으면 시작 호의 우선 순위를 높여주는 등의 방법을 사용하게 된다.

#### 3.3.2 가드 채널(Guard Channel)

핸드오버 호는 시작 호보다 우선적으로 처리되어야 한다. 그 이유는 전화가 걸리지 않는 것이 통화가 중간에 끊어지는 것보다 낫기 때문이다. 핸드오버 호를 보호하고 드로핑률을 줄이는 방안의 하나로 가드 채널을 이용한 방법을 들 수 있다. 가드 채널은 일정 양의 채널을

특정 호를 위하여 예약해 놓는 방식으로, 대개 핸드오버 호를 위하여 가드 채널을 사용한다[3][8-11]. 가드 채널을 이용한 방식에서 시작 호가 발생하게 되면 가드 채널을 제외한 채널을 할당받을 수 있다. 그러다가 가드 채널을 제외한 가용 채널이 고갈되면 시작 호는 블로킹된다. 하지만 핸드오버 호는 가드 채널을 이용하여 서비스를 지속할 수 있다.

가드 채널을 이용한 방법에서도 중요한 고려 사항은 가드 채널의 크기를 어느 수준으로 하느냐이다. 가드 채널의 크기를 늘리면 특정 호는 실패율이 줄어들게 되지만 예약된 채널은 다른 호가 사용할 수 없으므로 다른 호의 실패율은 도리어 늘어나는 결과를 초래할 수 있다. 따라서 전체 시스템의 효율성을 따져서 어느 수준으로 하는 것이 효율적인지를 판단하여 가드 채널의 크기를 결정하여야 한다. 몇몇 방식에서는 트래픽의 상황을 파악하여 가드 채널의 크기를 다이내믹(dynamic)하게 결정하는 방법을 사용하기도 한다. 효율적이긴 하지만 많은 프로세싱을 요구한다는 단점이 있다[10].

#### 4. 기존의 복합 호 처리 알고리즘

##### 4.1 개요

Kuen-Rong Lo, Cooper Chang 그리고 Chung-Ju Chang의 논문에서는 가드 채널과 버퍼를 함께 사용한 방법을 제시하고 있다[3][8]. 보통의 경우 가드 채널과 버퍼를 함께 사용함으로써 효율성을 높이고 있는데, 이 방법에서도 둘을 함께 사용하고 있다. 이 방법에서는 핸드오버 호만을 위한 버퍼뿐만 아니라 시작 호를 위한 버퍼도 사용하여 핸드오버 호와 시작 호를 모두 보호하고자 한다. 또한 마이크로 셀에서 가용 채널이 없을 때 매크로 셀에서 처리할 수 있도록 호를 넘겨주는 오버플로우와, 매크로 셀의 가용 채널이 없을 때 마이크로 셀로 호를 넘겨주는 언더플로우를 모두 고려하고 있다. 버퍼링 기법에서 중요한 항목인 대기 호 제거(renegeing)도 고려하고 있는데, 이는 버퍼에서 대기중인 핸드오버 호가 핸드오버 지역을 넘어서서 이미 드로핑 되었을 경우나, 시작 호가 사용자의 중도 포기로 인해 의미가 없어질 경우에 호를 버퍼에서 제거하는 것을 말한다.

그림 5는 이 방식의 구조를 좀 더 자세히 설명하고 있다. 왼쪽의 셀들은 한 매크로 셀에 포함된 N개의 마이크로 셀들을 나타낸다. 각 마이크로 셀에는 세 개의 버퍼를 두었는데, 하나는 매크로 셀의 포화로 인해 마이크로 셀로 넘어온 언더플로우 핸드오버 호를 위한 버퍼이고, 다른 하나는 인접 마이크로 셀로부터 넘어온 핸드오버 호를 위한 버퍼이다. 마지막 버퍼는 마이크로 셀에

서 발생한 시작 호를 위한 버퍼이다. 역방향(reversible) 핸드오버 호는 매크로 셀의 버퍼와 마이크로 셀의 버퍼 간의 균형을 위해 제안된 방식으로, 마이크로 셀의 가용 채널이 생겼을 때 매크로 셀의 버퍼와 마이크로 셀의 버퍼의 사용 정도를 비교하여 마이크로 셀의 버퍼 사용량이 작다면 마이크로 셀로 호를 넘겨주는 방식이다. 매크로 셀도 마이크로 셀과 유사한 구조이며 유사한 역할을 하는 버퍼들이 존재함을 볼 수 있다. 단, 매크로 셀에서는 마이크로 셀의 가용 채널이 없을 때 매크로 셀에서 시작 호를 처리할 수 있도록 시작 호를 위한 오버플로우 버퍼를 두고 있다.

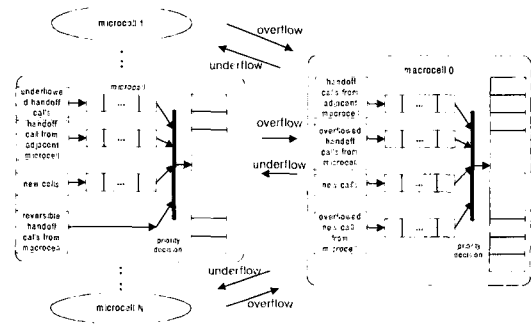


그림 5 시작 호, 핸드오버 호, 오버플로우/언더플로우 호에 버퍼를 할당해 준 경우

Lo와 Chang이 제안한 방식은 기존에 제안된 다양한 계층셀 핸드오버를 종합한 방식이다. 이 방식에서는 모든 호는 같은 크기의 대역폭을 할당받고, 한번 생성된 호의 대역폭은 항상 일정하다는 가정 하에 실험하여 좋은 성능을 보임을 확인하였다. 하지만 실제 IMT-2000 시스템에서는 각 사용자마다 요구하는 서비스가 다르므로 인해 요구하는 대역폭이 다를 것이다. 즉, 동영상 등의 서비스를 받고자 하는 사용자와 단순히 음성 통화만을 하는 사용자가 요구하는 대역폭은 큰 차이를 보이게 될 것이다. 그리고 통화를 끊지 않고서도 서비스 전환을 하고자 할 때 요구 대역폭이 변화하여야 할 것이다.

이 방식에서 고려하지 못한 또 하나의 사항은 매크로 셀에서 처리할 수 있는 최대 대역폭이다. 매크로 셀과 같이 큰 반경을 갖는 셀의 경우에는 그 크기로 인해 제공할 수 있는 대역폭에 제한을 받게 된다[12]. 따라서 ITU-R에서는 계층셀 별로 제공할 수 있는 대역폭이 다름에 따라 계층별 용도를 달리 설정해놓고 있다. 예를 들어 피코 셀의 경우는 2Mbps, 마이크로 셀의 경우는 384Kbps, 매크로 셀의 경우는 144Kbps로 설정해놓고

있는 것이 그것이다. 하지만 Lo, Chang의 논문에서는 이에 대한 고려를 하지 않고 있다. 따라서 매크로 셀에서 처리할 수 없는 대역폭을 요구할 경우 호를 강제 종료시키거나 대역폭을 강제로 낮춰 할당해 줄 수밖에 없다. 이동 속도의 면에 있어서도 속도가 빠른 호를 마이크로 셀에서 처리해 줄 경우 많은 핸드오버가 일어나 오버헤드가 커지게 되는데, 이에 대한 고려도 하고있지 않다.

본 논문에서는 이 문제점들을 해결하기 위해 보완된 알고리즘을 제안하였는데, 5.2.2에서 자세히 설명하도록 한다.

4.2 기존의 방법의 알고리즘

4.2.1 매크로 단독 지역에서의 호 처리

그림 6은 매크로 셀 단독 지역에서 호가 발생했을 경우의 플로우 차트이다. 여기서의 호는 매크로 단독 지역에서 발생한 시작 호와 매크로 단독 지역으로 이동해 온 핸드오버 호를 포함한다. 매크로 셀 단독 지역에서 호는 매크로 셀에서 처리될 수밖에 없기 때문에 계층간 핸드오버는 발생하지 않는다. 단, 매크로 지역에서 처리할 수 있는 최대 대역폭(MACRO\_MAX\_BW)이 있음을 고려해서, 매크로 단독 지역에서 생성되는 호 중 MACRO\_MAX\_BW를 넘는 대역폭을 요구하는 호의 경우 요구하는 대역폭보다 낮춰 할당하도록 하였다. 이 경우 원하는 대역폭을 100% 만족시켜주지 못하기 때문에 서비스의 질은 저하될 수밖에 없다. 매크로 셀에서 처리 가능한 대역폭의 경우에는 남아있는 대역폭을 검사하여 처리 가능하다면 처리해주고 남아있는 대역폭이 요구 대역폭보다 작다면 버퍼에 들어가게 된다. 이때 버

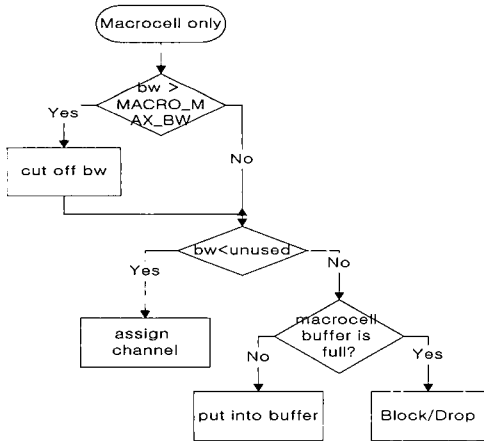


그림 6 비교 모델의 매크로 단독 지역 호 처리

퍼도 가득 차 있다면 그 호는 블로킹되거나 드로핑된다. 시작 호와 핸드오버 호의 차이는 핸드오버 호의 경우 가용 대역폭을 검사할 때 가드 채널도 이용하여 대역폭을 할당받을 수 있다. 반면 시작 호의 경우는 가드 채널을 제외한 대역폭에서만 할당받을 수 있다. 또한 핸드오버 호와 시작 호는 그림 5와 같이 사용하는 버퍼가 다르고 그에 따라 우선순위도 달라진다.

4.2.2 중첩 지역에서의 호 처리

중첩된 지역에서 호가 발생할 경우의 플로우 차트는 그림 7과 같다. 중첩된 지역에서 호가 발생한 경우는 마이크로 셀과 매크로 셀 모두에서 처리가 가능하다. 하지만 MACRO\_MAX\_BW 보다 큰 대역폭을 요구하는 경우는 마이크로 셀에 가용 채널이 존재할 경우 원하는 대역폭을 제공해줄 수 있지만, 매크로 셀에서 처리될 경우는 대역폭을 낮춰서 처리해주어야 한다. 그렇지 않으면 블로킹될 수밖에 없다. 여기서는 호의 블로킹/드로핑을 최소화하기 위해 대역폭을 낮춰 처리해주는 방식으로 처리하였다. Lo와 Chang은 호의 분산을 위해 중첩 지역에서 발생한 호의 경우 먼저 마이크로 셀에서 처리해줄도록 하였다.

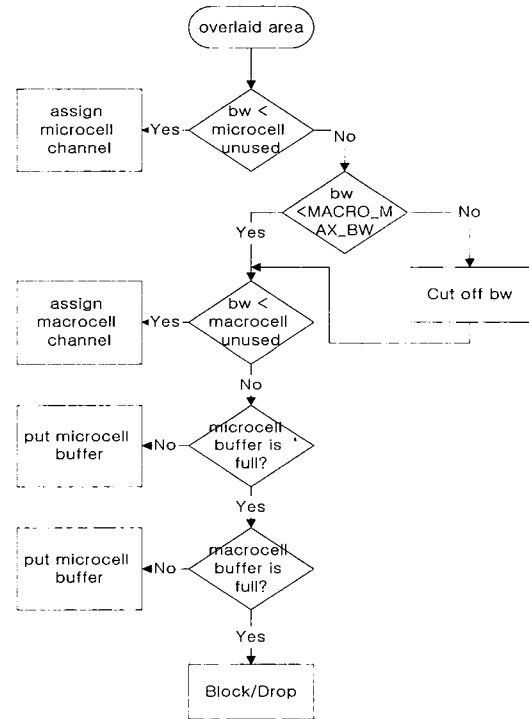


그림 7 비교 모델의 중첩 지역에서의 호 처리

중첩된 지역에서 시작 호가 발생하거나 중첩된 지역으로 호가 핸드오버되어 넘어올 경우, 마이크로 셀의 대역폭을 검사하여 가용 대역폭이 존재한다면 할당하고 존재하지 않다면 매크로 셀로 호를 넘긴다. 이 때 시작 호의 경우는 예약된 대역폭을 제외한 대역폭 중에서 할당받을 수 있지만, 핸드오버 호의 경우는 예약된 대역폭을 사용하여 채널을 할당받을 수 있다. 매크로 셀로 넘어온 호는 매크로 셀 처리 가능 대역폭이라면 가용 대역폭을 검사하게 되고, 매크로 셀에서 처리 불가능한 대역폭이라면 대역폭을 낮춰 할당하게 된다. 채널을 바로 할당받지 못한 호들은 버퍼로 들어가서 채널을 할당받기를 기다리게 된다. 이 경우와 같이 호의 컨트롤이 마이크로 셀과 매크로 셀을 이동할 때를 계층간 핸드오버라 한다. 실제 성능을 평가할 때는 블로킹, 드로핑률 뿐만 아니라 계층간 핸드오버율도 고려하였다. 그 이유는 계층간 핸드오버 시에도 기지국과 단말간의 많은 상호작용이 필요하고 그로 인해 많은 오버헤드가 발생되기 때문이다. 따라서 최소의 계층간 핸드오버로 최대의 성능을 낼 수 있다면 좋은 알고리즘이라 할 수 있을 것이다.

5. 제안한 복합 호 처리 알고리즘

5.1 개요

Lo와 Chang의 방식은 다양한 기법들을 활용해 좋은 성능을 보였지만 호의 대역폭과 이동 속도를 고려하지 않는 단점이 있었다. 본 연구에서는 오버플로우 기법과 언더플로우 기법, 가드 채널과 버퍼를 사용한 Lo와 Chang의 방식에서 좀 더 나아가 매크로 셀에서 처리 가능한 대역폭과, 속도를 고려한 알고리즘을 제안하였다. 제안한 알고리즘에서는 대역폭과 이동 속도를 고려하여 적합한 계층에서 처리해줌으로써 사용자가 요구하는 QoS를 최대한 만족시켜주고, 핸드오버율을 낮추고자 하였다.

제안한 알고리즘에서 사용한 버퍼와 가드 채널 구성은 Lo와 Chang이 사용한 구성과 같다. 시작 호를 위한 버퍼, 오버플로우된 시작 호를 위한 버퍼, 핸드오버 호를 위한 버퍼, 오버플로우/언더플로우된 핸드오버 호를 위한 버퍼를 따로 두었다.

중첩된 지역에서 요구하는 대역폭이 큰 호는 마이크로 셀에서 처리되도록 하였고 작은 대역폭을 요구하는 호는 매크로 셀에서 먼저 처리되도록 하였다. 큰 대역폭을 요구하는 호를 마이크로 셀로 먼저 보내는 이유는 매크로 셀에서 대역폭이 낮춰지는 현상을 최소화하여 최대한 QoS를 만족시켜 주려는 것이다. 작은 대역폭을 요구하는 호는 되도록 매크로 셀의 채널을 할당받게 하

여 마이크로 셀의 채널을 비워둠으로써 우선 순위가 높은 호가 요구하는 QoS를 만족시켜 주도록 한다. 또한 빠른 속도로 이동하는 호는 매크로 셀에서 먼저 처리하도록 하였다. 이는 많은 핸드오버가 발생하게 되어 잦은 핸드오버로 인해 네트워크 부하도 심해지는 것을 막기 위해서이다.

제안하고자 하는 알고리즘에서는 매크로 셀 단독 지역에서는 계층간 핸드오버가 불가능하므로 기존의 방식과 같은 방식으로 처리해주었다. 하지만 중첩 지역에서는 큰 대역폭을 요구하는 호와 작은 대역폭을 요구하는 호, 빠른 이동 속도의 호와 느린 이동 속도의 호로 구분하여 처리한다. 이 알고리즘의 자세한 사항은 5.2절에서 설명하였다.

5.2 제안된 방법의 알고리즘

5.2.1 매크로 셀 단독 지역에서의 호 처리

매크로 셀 단독 지역에서 발생한 시작 호의 경우 루틴(routine)은 비교 모델과 같다. 매크로 지역에서 처리할 수 있는 대역폭을 요구하는 호가 생성되었을 경우는 가용 대역폭을 검사하여 할당 할 수 있으면 할당해준다. 또 가용 대역폭이 부족한 경우에는 버퍼에 임시 저장하여 최대한 실패율을 줄일 수 있도록 한다. 만약 매크로 지역에서 처리할 수 없는 대역폭이 생성되었을 경우에는 대역폭을 낮춰서 할당할 수 있게 한다.

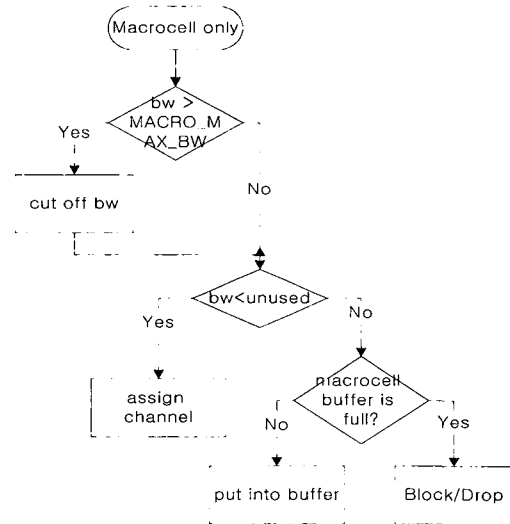


그림 8 제안 알고리즘의 매크로 셀 단독 지역에서의 호 처리

5.2.2 중첩 지역에서의 호 처리

중첩된 지역에서 호가 발생한 경우의 플로우 차트는

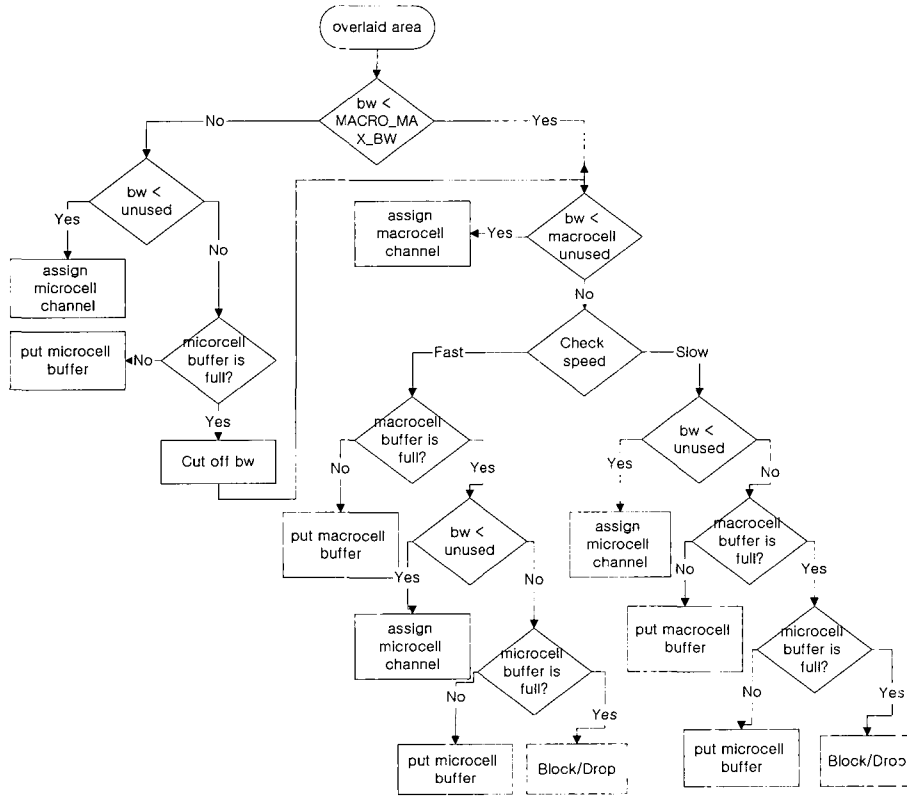


그림 9 제안 알고리즘의 중첩 지역에서의 호 처리

그림 9와 같다. 제한한 알고리즘에서는 비교 알고리즘과는 달리 매크로 지역 처리 가능 대역폭을 미리 검사하여 큰 대역폭을 요구하는 경우에는 마이크로 셀을 먼저 검사하게 한다. 이는 매크로 셀에서 처리 불가능한 큰 대역폭을 요구하는 호들을 마이크로 셀에서 먼저 처리하게 해줌으로써, 큰 대역폭을 요구하는 호의 실패율을 줄이려는 의도이다. 큰 대역폭을 요구하는 호들은 먼저 마이크로 셀의 가용 대역폭을 검사하여 할당 가능하다면 할당해주고 부족할 경우 마이크로 셀의 버퍼를 검사한다. 만약 버퍼도 가득 차 있다면 대역폭을 낮춰 매크로 셀의 채널을 받으려 시도하게 된다. 매크로 셀에서는 먼저 매크로 셀의 가용 대역폭을 검사하여 할당 가능하다면 할당해준다. 할당을 받지 못한 경우라면 호의 이동 속도를 검사하게 된다. 이유는 빠른 호를 매크로 셀에 할당하여 핸드오버율을 줄이기 위한 것이다. 느린 속도로 이동하는 호라면 마이크로 셀로 언더플로우되어 가용 대역폭을 검사하게 되고, 빠른 속도로 이동하는 호라면 매크로 셀의 버퍼를 먼저 검사하여 매크로 셀의 채

널을 할당받으려는 시도를 하게 된다. 이 방식의 장점은 호의 대역폭을 미리 파악하여 큰 대역폭을 요구하는 호의 QoS를 최대한 보호해주며, 이동 속도로 나누어 별도로 처리해줌으로써 핸드오버율을 최소화하려는 것이다.

또한 대역폭이 낮춰진 호라도 다시 마이크로 셀의 가용 대역폭을 검사할 수 있게 하여 호의 실패율을 줄이고 좀 더 효율적으로 처리하게끔 하였다.

## 6. 시뮬레이션 및 결과 분석

### 6.1 시뮬레이션 환경 및 파라미터

본 절에서는 계층셀 셀룰러 시스템을 모델링함에 있어서 설정해 준 몇 가지 파라미터와 환경 변수, 가정 사항을 제시한다. 먼저 계층셀 구조는 2장에서 설명한 바와 같이 핫 스팟을 이용한 구성 방법을 쓰되 마이크로 셀과 매크로 셀의 2계층의 구조를 사용하였다. 매크로 셀은 총 10개로 설정하였고 매크로 셀과 중첩된 마이크로 셀은 각 매크로 셀마다 9개로 가정하였다. 매크로 셀의 반경은 5km로, 마이크로 셀의 반경은 500m로 설정



하였으며, 호가 이동하게 되는 경우는 매크로 셀에서 매크로 셀, 매크로 셀에서 마이크로 셀, 마이크로 셀에서 매크로 셀, 마이크로 셀에서 마이크로 셀의 네 가지 경우가 모두 발생하도록 하였다. 매크로 셀 기지국의 대역폭은 30M, 마이크로 셀 기지국의 대역폭은 5M를 갖는다. 그리고 이동 단말의 속도는 빠른 단말의 경우 평균 60km/h ~ 120km/h 사이에, 느린 단말의 경우 평균 이동 속도는 0.5m/s ~ 1.2m/s 사이에서 균일한 분포를 따르는 것으로 가정하였다. 단말의 이동 방향은 각 이웃 셀로의 방향, 즉 6개의 방향으로 이동한다고 가정하였고, 한 셀에서 벗어나게 되면 다음 셀로 들어가는 것으로 처리하였다. 호의 대역폭도 큰 대역폭을 요구하는 경우는 평균 145kbps ~ 384kbps, 작은 대역폭을 요구하는 경우는 평균 30kbps ~ 144kbps의 대역폭을 요구하도록 하였다. 호는 셀 내에서 고르게 발생하고, 각 호의 발생률은 평균  $\lambda$ 의 포아송(Poisson) 분포를 따르며, 호의 지속 시간은 평균  $1/\mu$ 인 지수(exponential) 분포를 따른다고 가정하였다. 호의 대역폭이 변하는 것을 모델링하기 위해 호가 셀의 경계를 넘어갈 경우 기지국과 QoS 제협상을 통해 새 대역폭을 할당받도록 하였다. 본 시뮬레이션은 자바로 구현하였고 Sun Sparc에서 실행하였다.

표 2 실제 시뮬레이션에 사용된 파라미터

파라미터	값	설명
originate rate	0.1 ~ 1.0	시작 호의 발생률
Highbw:Lowbw	1:9 ~ 8:2	높은 대역폭 요구 호: 낮은 대역폭 요구 호
Fast:Slow	1:9 ~ 9:1	빠른 이동 속도 호: 느린 이동 속도 호
guard channel	10% ~ 70%	가드 채널의 비율
buffer size	1 ~ 10	버퍼의 크기
buffer duration	5 ~ 25	버퍼에서 대기할 수 있는 시간
call duration	100 ~ 300	호의 통화 시간

표 3 시뮬레이션을 통해 측정하고자 하는 항목

항목	설명
Blocking rate	시작 호의 블로킹률
Dropping rate	핸드오버 호의 드로핑률
Bandwidth utilization	대역폭 이용률
Inter-layer handover rate	계층간 핸드오버율
Lambda_h(inter-cell handover rate)	핸드오버율
Bandwidth cut off rate	대역폭 강제 낮춤률
Buffer utilization	버퍼 이용률

실제 시뮬레이션을 수행할 때 사용한 파라미터는 표 2에서 나타내었다. 여러 가지 상황에서 성능을 측정하기 위해 실제 이동 통신 시스템에서 변화할 수 있는 파라미터를 결정한 것이다. 버퍼와 가드 채널의 크기는 임의로 설정할 수 있는 변수이지만 다른 파라미터는 실제로 조절할 수 없는 파라미터들이다. 또한 시뮬레이션을 통해 측정하고 비교 분석해 본 항목은 표 3과 같다.

6.2 시뮬레이션 결과

제안한 알고리즘의 성능을 평가하기 위하여 셀의 블로킹/드로핑률, 기지국에서의 요구 QoS 보장, 셀간 핸드오버율, 계층간 핸드오버율, 대역폭/버퍼 이용률을 분석해보았다. 또한 여러 가지 상황에서의 성능을 평가하기 위하여 호 생성률을 0.1에서 1.0까지 변화시켜보았을 때의 결과와 큰 대역폭을 요구하는 호의 비율을 10%에서 80%까지 변화시켜 보았을 때, 그리고 빠른 속도로 이동하는 호의 비율을 90%까지 변화시켜 보았을 때의 결과를 그래프를 통해 비교해보았다. 또한 버퍼의 크기를 1에서 10까지 변화시켜 보고, 가드 채널의 비율을 10%에서 70%까지 변화시켜 보면서 결과를 분석해보았다. 그래프에서 'LC'로 표기된 것은 Lo와 Chang의 알고리즘의 실험 결과를 나타낸 것이고, '제안'으로 표기된 것은 제안한 알고리즘의 실험 결과를 나타낸 것이다.

6.2.1 블로킹/드로핑률

Blocking probability

$$= \frac{\sum_{i=1}^{MN_0} (N_{block} + N_{renegingBt}) + \sum_{j=1}^{mN_0} (N_{block} + N_{renegingBt})}{\sum_{i=1}^{MN_0} N_{origin} + \sum_{j=1}^{mN_0} N_{origin}}$$

Dropping probability

$$= \frac{\sum_{i=1}^{MN_0} (N_{drop} + N_{renegingD}) + \sum_{j=1}^{mN_0} (N_{drop} + N_{renegingD})}{\sum_{i=1}^{MN_0} N_{handover} + \sum_{j=1}^{mN_0} N_{handover}}$$

$mN_0$  = Number of microcell,  
 $MN_0$  = Number of macrocell

$N_{block}$  = Number of BlockingCalls,  
 $N_{origin}$  = Number of OriginatedCalls

$N_{drop}$  = Number of DroppingCalls,  
 $N_{handover}$  = Number of HandoverRequestCalls

$N_{renegingBt}$  = Nmber of RenegingBlock,  
 $N_{renegingD}$  = Nmber of RenegingDrop

위 식과 같이 블로킹률은 발생한 시작 호 중에서 블로킹된 호의 수와 시작 호를 위한 버퍼에서 삭제된 경우의 호의 수를 더한 값의 비를 구한 값이다. 드로핑률도 발생한 핸드오버 호 중에서 드로핑된 호의 수와 핸

드오버 버퍼에서 삭제된 호의 수를 더한 값의 비율이다. 호의 블로킹률과 드로핑률은 호 처리 알고리즘을 평가할 때 매우 중요한 항목이다. 통화가 잘 연결되고 끊어질 확률이 낮아야 좋은 호 처리 알고리즘이라 할 수 있다. 하지만 통화가 처음부터 걸리지 않는 것이 중간에 끊어지는 것보다는 위험이 적으므로 대부분의 호 처리 알고리즘에서는 핸드오버 호를 보호하고자 가드 채널을 사용하고 있다.

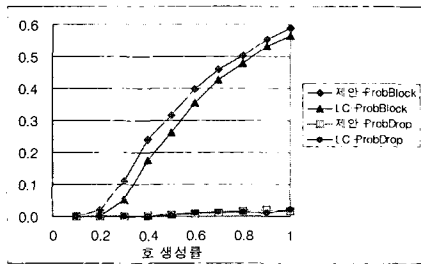


그림 10 호 생성률 변화시 블로킹률과 드로핑률

호의 생성률을 0.1에서 1.0까지 변화시켜 보았을 때 블로킹률과 드로핑률의 변화를 그림 10에서 보여주고 있다. Lo와 Chang의 알고리즘과 제안한 알고리즘을 비교해볼 때 블로킹률은 제안한 알고리즘이 4~5% 정도 높게 나타났다. 그 이유는 Lo와 Chang의 방식에서 대역폭을 무조건 낮춰서 처리해준 것과는 달리 제안한 알고리즘에서는 요구하는 대역폭을 최대한 만족시켜 주고자 하였기 때문이다. 대역폭을 최대한 만족시켜 주게 되면 그 호들이 차지하는 대역폭이 커지게 되고, 그로 인해 다른 호들이 할당받을 대역폭이 줄어들기 때문이다. 따라서 기지국이 일정한 대역폭을 가지고 있다고 가정했을 때 이는 불가피하게 발생한 것이다.

반면에 드로핑률을 살펴보면 두 알고리즘에서 모두 매우 낮은 수치로 차이를 거의 보이지 않고 있다. 시작 호보다는 핸드오버 호가 우선 순위가 높은 것을 감안해볼 때 어느 정도의 블로킹률의 증가는 기지국에서의 QoS 보장을 위해 감수할 수밖에 없다.

그림 11에서처럼 높은 대역폭을 요구하는 호가 많았을 경우에는 가용 대역폭의 감소로 인해 블로킹률과 드로핑률이 모두 증가하였다.

그림 12에서는 빠른 호가 증가할수록 블로킹률과 드로핑률의 결과가 역전됨을 볼 수 있다. 호의 블로킹률은 점점 감소하였고, 드로핑률은 점점 증가하였다. 이는 호의 이동성이 클수록 빨리 대역폭을 해제하기 때문에 시작 호가 할당받을 대역폭은 증가하여 블로킹률은 낮아

지고, 반대로 핸드오버율은 증가하였기 때문에 드로핑률은 증가한 것이다.

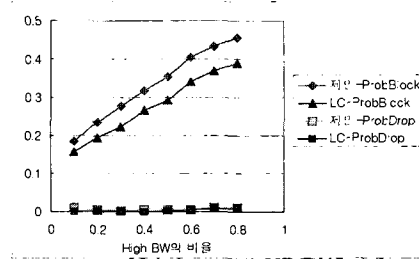


그림 11 큰 대역폭 요구 호 비율 변화시 블로킹률과 드로핑률

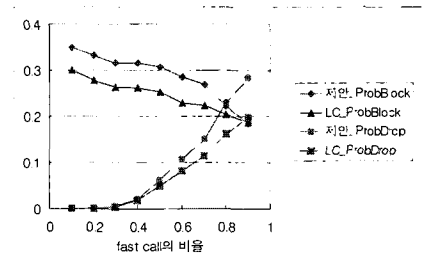


그림 12 빠른 호 비율 변화시 블로킹률과 드로핑률

6.2.2 기지국에서의 요구 QoS 보장(대역폭 낮춤률)

$$Cut-off\ rate = \frac{\sum_{i=1}^{MNo} N_{cutoff} + \sum_{i=1}^{MNo} N_{cutoff}}{\sum_{i=1}^{MNo} N_{highbuNew} + \sum_{i=1}^{MNo} N_{highbuHo} + \sum_{i=1}^{MNo} N_{highbuNew} + \sum_{i=1}^{MNo} N_{highbuHo}}$$

$mNo$  = Number of microcell,  
 $MNo$  = Number of macrocell

$N_{cutoff}$  = Number of Cut-offCalls,  
 $N_{highbuNew}$  = Number of HighbuNewCalls

$N_{highbuHo}$  = Number of highbuHandoverCalls

이동 통신 사업자의 입장에서는 등급에 따라 사용자를 구분하여 다른 QoS를 제공해야 한다. 또한 사용자의 입장에서도 등급에 따라 다른 QoS를 요구할 수 있어야 한다. 즉, 사용자의 QoS를 보장해주는 것은 매우 중요한 사항이라 할 수 있다. 이를 위해서 본 연구에서는 호가 요구하는 대역폭에 따라 달리 처리해주는 방식을 제안하였다. 시뮬레이션 시에는 높은 대역폭을 요구하는 그룹에 높은 등급을 주어 원하는 대역폭을 저하시키지 않고 최대한 만족시켜주고자 하였다.

시뮬레이션에서 QoS를 평가하기 위해 측정한 값은

원하는 위 식에서와 같이 큰 대역폭을 요구한 호 중에서 대역폭을 다 할당받지 못하고 낮춰서 할당받은 호의 비율을 측정하였다. 그 결과가 아래 그림 13, 그림 14, 그림 15이다. 그림 13은 호의 생성률을 변화시켰을 때의 대역폭 낮춤률을 나타낸 그래프이다. 초기의 생성률이 아주 작을 때는 Lo, Chang의 알고리즘과 큰 차이를 보이지 않으나 생성률이 높아질수록 격차가 커짐을 볼 수 있다. 이는 원하는 QoS를 받지 못한 호가 크게 증가했음을 보여주는 것이다.

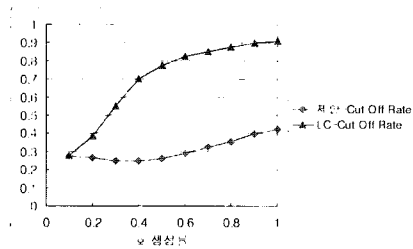


그림 13 호 생성률 변화시 대역폭 낮춤률

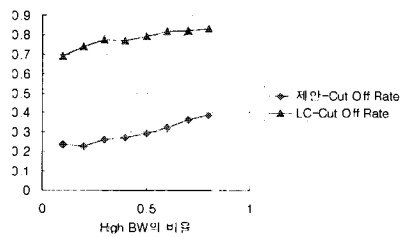


그림 14 큰 대역폭 요구 호의 비율 변화시 대역폭 낮춤률

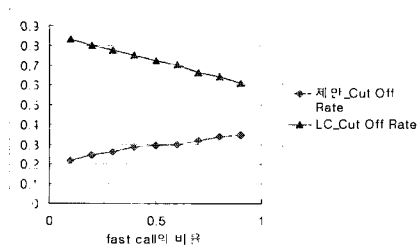


그림 15 빠른 호의 비율 변화시 대역폭 낮춤률

그림 14와 그림 15는 큰 대역폭을 요구하는 호가 증가했을 경우와 빠른 속도로 이동하는 호가 증가했을 때의 대역폭 낮춤률을 나타낸다. 큰 대역폭을 요구하는 호가 증가했을 때 낮춤률이 커짐은 당연하다고 할 수 있

다. 따라서 그림 14와 같이 점점 증가하고 있다. 하지만, 제안한 알고리즘이 크게 낮춤을 확인한 수 있다. 그림 15에서는 제안한 알고리즘의 경우 빠른 속도로 이동하는 호가 증가할 경우 핸드오버가 증가함에 따라 대역폭 낮춤률이 증가했기 때문에 수치가 점점 증가하였다. 하지만 Lo와 Chang의 알고리즘의 경우 빠른 속도로 이동하는 호가 증가할수록 대역폭 낮춤률은 감소하고 있는데, 그 이유는 대역폭 낮춤 수의 증가에 비해서 핸드오버 수가 더 급격히 증가함에 따라 대역폭 낮춤률이 낮아지게 된 것이다. 하지만 이 경우 역시 제안한 알고리즘의 대역폭 낮춤률이 Lo와 Chang의 알고리즘에 비해 월등히 낮음을 알 수 있다.

본 논문에서는 시뮬레이션 시 핸드오버가 일어날 때마다 새 기지국에서는 원래 요구하던 대역폭을 제공해 주기 위해 다시 요구 QoS를 높이도록 시뮬레이션 하였다. 즉, 이동한 셀의 기지국에서는 이전 셀에서 호가 대역폭이 낮춰졌다 하더라도 이동했을 때 다시 큰 대역폭을 할당해주도록 하였다.

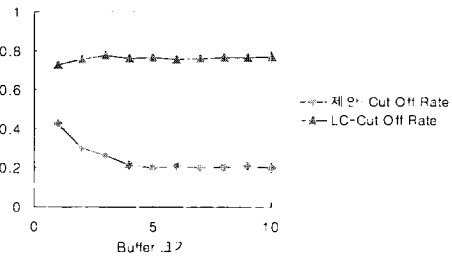


그림 16 버퍼 크기 변화시 대역폭 낮춤률

그림 16에서는 버퍼의 크기를 늘렸을 때 대역폭 낮춤률을 나타내었다. 버퍼가 커질수록 제안한 알고리즘은 대역폭을 낮추는 확률이 크게 줄어들었지만 Lo와 Chang의 알고리즘에서는 별로 변화가 없다. 제안한 알고리즘을 사용할 때 버퍼를 늘려주었을 경우 더 좋은 성능을 나타낼 수 있음을 예상할 수 있다.

### 6.2.3 셀간 핸드오버율

계층셀 구성은 빠른 속도로 이동하는 호를 변경이 큰 셀에서 처리해주어 핸드오버율을 줄이고자 하는 것이 목적이다. 하지만 Lo와 Chang의 알고리즘에서는 이동 속도에 대한 고려가 없었다. 핸드오버는 기지국과 기지국, 기지국과 단말 간의 많은 트래픽의 교환과 처리를 필요로 하는 오버헤드가 큰 작업이므로 최소화하는 것이 중요하다.

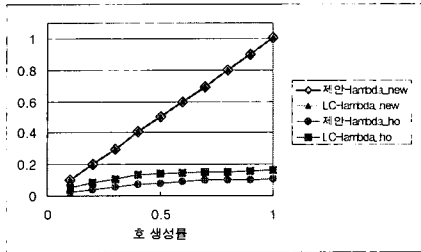


그림 17 호 생성률 변화시 시작 호와 핸드오버 호 발생률 (Calls/Sec/Cell)

그림 17은 호의 생성률이 증가함에 따른 시작 호의 발생률과 핸드오버 호의 발생률을 나타낸 그래프이다. 호의 발생률은 람다(lambda)를 사용하여 계산하였다. 즉, 시간당 각 셀에서 발생한 호의 수를 나타낸 것이다. 생성률은 시작 호의 발생률과 비례하기 때문에 그림에서 보는 것처럼 등비례하여 증가하였다. 핸드오버율은 드로핑이 일어나기도 하고, 호가 종료되기도 하기 때문에 시작 호와는 달리 일정한 비율로 증가하지는 않았다. 하지만 제안한 방식의 핸드오버 발생률이 Lo, Chang의 방식에서의 발생률보다 약 6~7% 정도 적음을 볼 수 있다. 이는 전체적으로 핸드오버율을 낮춰 전체적인 시스템 부하를 줄일 수 있게 된다.

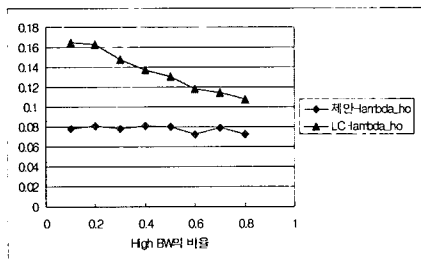


그림 18 큰 대역폭 요구 호 비율 변화시 핸드오버 발생률 (Calls/Sec/Cell)

그림 18에서는 큰 대역폭을 요구하는 호가 많아질수록 핸드오버 발생률이 줄어들었는데, 큰 대역폭을 요구하는 호가 많을수록 시작 호의 블로킹률이 증가하여 전체적으로 핸드오버 호가 감소했다고 추론해볼 수 있다.

빠른 속도로 이동하는 호가 많아질 경우 핸드오버율을 그림 19에서 나타내었다. 빠르게 이동하는 호가 많아질수록 셀간 핸드오버가 많이 일어나게 됨은 당연하다. 그림 19에서 보면 빠른 호가 차지하는 비율이 적은 경우에는 제안한 알고리즘과 Lo, Chang의 알고리즘이 큰

차이를 보이지 않고 있다. 하지만 빠른 속도로 이동하는 사용자가 증가할 경우는 점점 차이가 벌어짐을 볼 수 있다.

그림 20은 가드 채널의 비율을 높였을 때의 핸드오버 발생률의 변화를 나타낸 그래프이다. 가드 채널의 비율이 높아지면 시작 호가 사용할 수 있는 대역폭이 작아진다. 따라서 블로킹률이 높아지게 되고 블로킹률이 높아짐에 따라 핸드오버 호의 수도 적어지게 된 것이다.

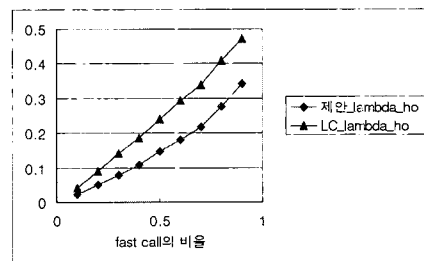


그림 19 빠른 호 비율 변화시 핸드오버 발생률 (Calls/Sec/Cell)

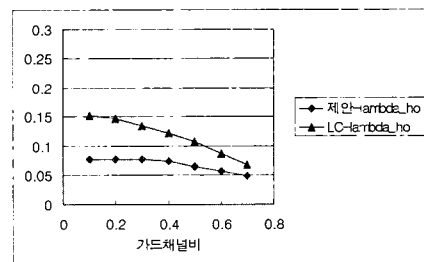


그림 20 가드 채널 비 변화시 핸드오버 발생률 (Calls/Sec/Cell)

6.2.4 계층간 핸드오버율

계층간 핸드오버도 셀간 핸드오버와 같이 복잡한 처리과정을 거치는 작업이다. 따라서 계층간 핸드오버율 역시 낮을수록 효과적인 알고리즘이라 할 수 있다. 계층간 핸드오버율 역시 셀간 핸드오버율과 같이 람다를 사용하여 성능을 비교하였다. 그림 21과 그림 23에서는 제안한 알고리즘이 계층간 핸드오버율이 약 8~9% 작음을 볼 수 있다. 하지만 그림 22에서 볼 수 있듯이 호들이 요구하는 대역폭의 정도에 따라서 상당히 큰 차이를 보이고 있다. 제안한 알고리즘에서는 큰 대역폭을 요구하는 호에게 먼저 마이크로 셀의 대역폭을 할당해주도록 하고 있다. 하지만 Lo, Chang의 알고리즘에서는 무조건 마이크로 셀에서 모든 호를 처리해준다. 큰 대역폭을 요

구하는 호의 비율이 적을 경우에 제안한 알고리즘에서는 매크로 셀로 보내는 호가 많다. 따라서 매크로 셀에 호가 집중되기 때문에 매크로 셀에서 마이크로 셀로 이동하는 계층간 핸드오버가 많이 일어나게 된 것이다. 하지만 큰 대역폭을 요구하는 호가 증가할수록 제안한 알고리즘에서는 호들을 나누어 처리하게 함으로써 매크로 셀과 마이크로 셀간의 트래픽 균형을 맞추게 된 것이다. 계층간 핸드오버율을 최소화하려고 할 때에는 큰 대역폭을 요구하는 호의 비율이 30%보다 클 경우 제안한 알고리즘이 훨씬 우수하다.

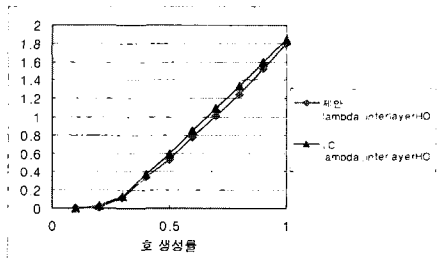


그림 21 호 생성률 변화시 계층간 핸드오버율 (Calls/Sec/Cell)

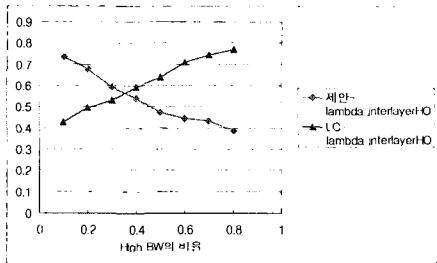


그림 22 큰 대역폭 요구 호 비율 변화시 계층간 핸드오버율(Calls/Sec/Cell)

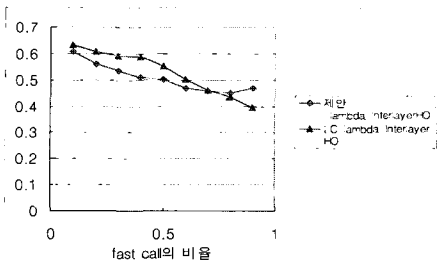


그림 23 빠른 호 비율 변화시 계층간 핸드오버율 (Calls/Sec/Cell)

그림 23에서는 빠른 호의 비율이 70%를 넘은 경우에 제안한 알고리즘의 계층간 핸드오버율이 증가함을 볼 수 있다. 그 이유도 위와 같이 설명할 수 있는데, 제안한 알고리즘에서는 빠른 호를 매크로 셀에 할당해주기 때문에 빠른 호의 비율이 증가할수록 매크로 셀에 호가 집중되어 계층간 핸드오버가 많이 발생한 것이다.

6.2.5 대역폭/버퍼 이용률

$$Channel\ Utilization = \frac{\sum_{i=1}^{mNo} N_{usingC} + \sum_{i=1}^{mNo} N_{usingC}}{\sum_{i=1}^{mNo} N_{totalC} + \sum_{i=1}^{mNo} N_{totalC}}$$

$$Buffer\ Utilization = \frac{\sum_{i=1}^{mNo} N_{usingB} + \sum_{i=1}^{mNo} N_{usingB}}{\sum_{i=1}^{mNo} N_{totalB} + \sum_{i=1}^{mNo} N_{totalB}}$$

$mNo$  = Number of microcell  
 $MNo$  = Number of macrocell

$N_{totalC}$  = Number of TotalChannels  
 $N_{usingC}$  = Number of UsingChannels

$N_{totalB}$  = Number of TotalBuffers  
 $N_{usingB}$  = Number of UsingBuffers

대역폭과 버퍼를 얼마나 효율적으로 이용하는지를 평가하는 것은 알고리즘의 효율성을 평가하기 위한 목적도 있겠지만, 적정 수준의 기지국 대역폭과 버퍼 크기를 정할 때에도 유용하게 사용할 수 있는 정보가 될 것이다. 대역폭과 버퍼의 이용률은 전체 할당된 대역폭과 버퍼 중에서 사용하는 대역폭과 버퍼의 비율을 계산한 것이다.

그림 24는 호 생성률에 따라 변화된 채널과 버퍼 이용률을 보여주는 그래프이다. 대역폭의 이용률은 Lo, Chang의 알고리즘과 제안한 알고리즘이 거의 비슷한 수치를 나타낸다. 단, 버퍼 이용률은 제안한 알고리즘이 2~8% 정도 높게 나타나 우수함을 보이고 있다. 그 이유는 제안한 알고리즘이 좀 더 많이 버퍼를 검사하여 버퍼를 할당받기 때문이라 분석된다. 제안한 알고리즘에서는 대역폭이 낮춰진 경우라도 다시 마이크로 셀의 가용 대역폭과 버퍼를 검사하기 때문에 이용률을 높여 주고 있다.

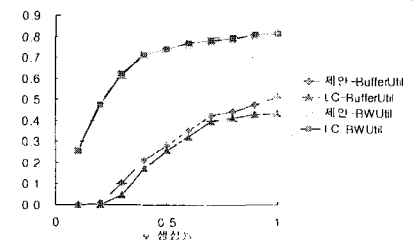


그림 24 호 생성률 변화시 대역폭과 버퍼 이용률

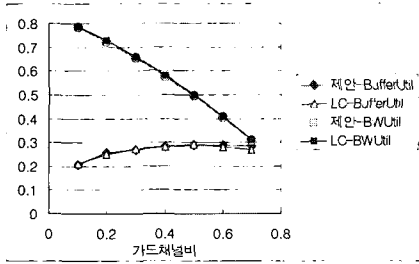


그림 25 가드 채널 변화시 대역폭과 버퍼 이용률

그림 25는 가드 채널을 크게 설정했을 때 대역폭과 버퍼의 이용률을 나타내었다. 가드 채널의 비율을 높게 하면 많은 시작 호들이 가드 채널을 제외한 적은 대역폭만을 사용하기 위해 경쟁할 수밖에 없고, 적은 수의 핸드오버 호는 늘어난 가드 채널을 충분히 사용하지 않기 때문에 대역폭 이용률은 급속히 감소한다. 반면 대역폭을 할당받지 못한 시작 호들은 버퍼에 들어가게 되므로 버퍼 이용률은 줄어들지 않았다.

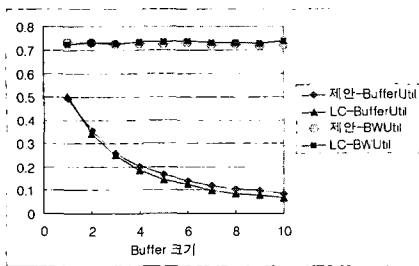


그림 26 버퍼 크기 변화시 대역폭과 버퍼 이용률

그림 26에서는 버퍼의 크기가 증가할수록 버퍼 이용률이 급속히 감소함을 볼 수 있다. 이는 버퍼의 크기를 크게 늘린 경우 사용하지 않는 버퍼도 증가함을 알 수 있다. 적정 수준의 버퍼 크기를 정하는 것은 시스템의 효율성 측면에서나 호의 지연 시간 측면에서나 중요한 사항이다.

7. 결론

이동 통신 수요에 증가에 따라 기존에 유선 망에서 받던 서비스를 무선 망에서 받고자 하는 욕구가 대두되었다. 또한 꿈의 통신이라 불리는 IMT-2000의 상용화로 인해 높은 대역폭을 요구하는 다수의 사용자들 수용해야 하는 문제에 봉착하게 되었다. 따라서 기존의 시스템으로는 한계에 이를 수밖에 없다는 판단 하에 새로운

제안된 것이 계층셀 구조이다. 계층셀 구조는 반경이 다른 여러 계층의 셀을 중첩하여 보다 효과적으로 호를 처리해줄 수 있는 장점이 있으나 호 처리 알고리즘이 복잡하고 시스템의 부하를 증가시킨다는 단점이 있다. 따라서 최적의 호 처리 알고리즘을 찾는 것은 매우 중요한 문제이다.

본 연구에서는 기존에 Lo와 Chang에 의해 제안된 복합 기법에서 좀 더 나아가 이동 속도와 대역폭을 고려한 보완된 알고리즘을 제안하였다. 실제로 호의 대역폭은 다양하고 상황이나 사용자의 요구에 따라 수시로 변할 수 있고, 호의 실패율을 줄이는 것 못지않게 사용자의 요구 QoS를 보장해주는 것이 중요하므로, QoS의 보장을 위해 Lo와 Chang의 방식에 덧붙여 대역폭에 따라 구분하여 처리해주는 모듈을 추가하였다. 또한 이동 속도에 따라 다른 계층에서 처리하는 모듈을 추가하여 빠른 속도로 이동하는 호를 반경이 큰 호에서 처리해줌으로써 빈번한 핸드오버율을 줄이고자 하였다.

제한한 알고리즘의 성능을 평가하기 위해 먼저 매크로 셀과 마이크로 셀의 2계층으로 이루어진 계층 셀룰러 시스템을 구성하였다. 그리고 이동체의 속도와 대역폭, 이동 방향, 발생률 등을 모델링하여 시뮬레이션을 수행하였다. 시뮬레이션 결과 전체적으로 블로킹률은 약간씩 증가하였지만 드로핑률은 거의 차이를 보이지 않았고, 높은 대역폭을 요구하는 호의 요구 대역폭을 만족시켜줌으로써 QoS를 보장해 줄 수 있었다. 또한 핸드오버율도 감소하여 핸드오버로 인한 오버헤드를 적게 일으킴을 확인할 수 있었다.

이러한 결과들을 종합해 본 결과 제안한 알고리즘은 치명적인 드로핑률은 감소시키지 않고 사용자들이 요구하는 QoS를 만족시켜줌과 동시에 핸드오버율도 감소시키는 효율적인 알고리즘임을 확인할 수 있었다. 계층셀 구조는 단층셀 구조와는 달리 한 계층에서 처리하지 못하는 호는 다른 계층으로 넘어가게 되기 때문에 서로 영향을 주게 된다. 예를들어 한 계층에서 블로킹률이나 드로핑률이 높은 경우 다른 계층으로 핸드오버할 호도 적어지게 되므로, 다른 계층에서의 블로킹률과 드로핑률을 오히려 증가시키는 효과를 낳기도 한다. 따라서, 각 계층의 셀에 호를 적절히 분배하여 처리하는 것이 무엇보다 중요하다고 할 수 있겠다.

본 연구에서는 여러 기법을 활용한 Lo와 Chang의 방식에서 보완된 호 처리 알고리즘을 제안하여 성능 향상을 가져옴을 입증하였다. 이 알고리즘은 앞으로 상용화 될 IMT-2000 시스템에서도 시스템 활용률을 높이고 사용자의 QoS를 만족시켜주기 위한 알고리즘으로 활용될

수 있을 것이다.

본 연구에서는 기지국이 단말의 이동 속도와 대역폭을 모두 알고있다고 가정하고 시뮬레이션 하였다. 하지만 실제 시스템에서 기지국이 단말의 이동 속도와 대역폭의 변화를 계속적으로 모니터링하면서 적절히 대응하기에는 상당히 오버헤드가 클 것으로 예상된다. 따라서 단말의 이전 이동 속도나 대역폭을 정보로 활용하여 이동 속도나 대역폭을 예측하는 방안이 연구되어야 할 것이고 기지국과 단말의 갖은 QoS 협상 시 오버헤드를 줄이기 위한 방안이 연구되어야 할 것이다.

이와 더불어 CBR, ABR, rt-VBR, nrt-VBR 등 다양한 멀티미디어 트래픽을 전송할 때 적합한 호 처리 알고리즘에 대한 연구와, 빠른 속도로 이동하며 높은 대역폭을 요구하는 호의 경우 매크로 셀과 마이크로 셀에서 처리할 때의 성능을 비교해보는 것도 하나의 이슈가 될 수 있겠다.

참 고 문 헌

[1] Nikos Dimitriou, Rahim Tafazolli, "Quality of Service for Multimedia CDMA," IEEE Communications magazine, vol. 38, no. 7, Jul. 2000.

[2] 한국전자통신연구원, "무선 ATM 기술 개론", 진한도서, 1998.

[3] Cooper Chang, Chung-Ju Chang, Kuen Rong Lo, "Analysis of a Hierarchical Cellular System with Reneging and Dropping for Waiting New and Handoff Calls," IEEE Transactions on Vehicular Technology, vol. 48, No. 4, 1999.

[4] Dong Myung Lee, Dong Cheul Son, "Performance Estimation of Prioritized Queueing Handoff Scheme in CDMA Cellular System," IEEE VTC'98, vol. 3, pp.1049~1053, 1998.

[5] Romando Fantacci, "Performance Evaluation of Prioritized Handoff Scheme in Mobile Cellular Networks," IEEE Transaction on Vehicular Technology, vol. 49, no. 2, Mar. 2000.

[6] J. G. Markoulidakis, J. E. Dermizakis, G. L. Lyberopoulos, M. E. Theologou, "Optimal system capacity in handover prioritized schemes in cellular mobile telecommunication systems," Computer Communications, vol. 23, pp462~475, 2000.

[7] 한정안, 박상준, 조재준, 김병기. "IMT-2000망에서 이동체의 속도와 트래픽 특성을 고려한 핸드오버 기법", 정보과학회논문지(A), 제26권 제12호, pp.1504~1515, 1999.

[8] Kuen-Rong Lo, Chung Ju Chang, Cooper Chang, C. Bernard Shung, "A combined channel assignment mechanism for hierarchical cellular systems,"

Computer Communications, vol. 21, no. 13, 1998.

[9] I. Katzela, M. Naghshineh, "Channel Assignment Schemes for Cellular Mobile Telecommunication Systems: A Comprehensive Survey," IEEE Personal Communications, Jun. 1996.

[10] Mohamed M. Abdallah, Khaled M. Al-Sayed, Mahomoud T. El Hadidi, "Effect of User Mobility on the QoS Parameters for the Guard Channel Policy," IEEE WCNC, vol. 3, pp.1503~1507, 1999.

[11] Yan Jiang, Vijay K. Bhargava, "Mobility Oriented Guard Channel Assignment For Personal Communicatio Systems," IEEE ICPWC'97, pp. 15~18, 1997.

[12] A. M. Viterbi and A. J. Viterbi, "Erlang Capacity of a Power Controlled CDMA System," IEEE JSAC, vol. 11, no. 6, Aug. 1993.



박 정 은

1999년 이화여자대학교 컴퓨터학과 졸업(학사). 2001년 이화여자대학교 컴퓨터학과(공학석사). 2001년 ~ 현재 삼성전자 통신연구소 연구기획그룹 연구원. 관심분야는 IMT-2000, 무선인터넷



채 기 준

1982년 연세대학교 수학과 졸업(학사). 1984년 미국 Syracuse University 전자계산학과(석사). 1990년 미국 North Carolina State University 컴퓨터공학과(공학박사). 1990년 ~ 1992년 미군 해군사관학교 전자계산학과 조교수. 1992년 ~ 현재 이화여자대학교 컴퓨터학과 교수. 관심분야는 네트워크 보안, 액티브 네트워크 보안 및 관리, 인터넷/무선 통신망/고속통신망 프로토콜 설계 및 성능분석