

셀룰라 이동 컴퓨팅 환경에서 유전 알고리즘을 이용한 채널차용 기법

(A Channel Borrowing Scheme using Genetic Algorithm in Cellular Mobile Computing Environment)

이 성 훈 [†] 이 동 우 ^{**} 이 상 구 ^{***}
 (Seong Hoon Lee) (Dong Woo Lee) (Sang Gu Lee)

요 약 셀룰라 이동 컴퓨팅 환경에서 정적 채널 할당 방법은 각 셀에 고정된 개수의 채널들이 존재하여 해당 셀에 들어오는 요청들에 대한 서비스를 제공한다. 이 방법은 매우 간단한 반면에 과다 셀 문제(Hot Cell problem)를 제어할 수 없는 단점을 보인다. 이같은 문제를 해결하기 위해서는 “채널 차용(channel borrowing)” 개념이 필요하다. 본 논문에서는 유전 알고리즘(genetic algorithm)을 이용한 부하균등(채널 차용, 채널 재할당) 접근법을 제안한다. 본 논문에서 유전 알고리즘을 사용하는 목적은 ① 유희한 채널을 포함하는 셀을 조기에 발견하고 ② 셀을 발견하기 위한 조사횟수를 줄이며 ③ 최적에 근접한 유희 채널을 발견하여 해당 셀로 이전하기 위함이다. 제안된 알고리즘의 효율성을 보이기 위해 다양한 시뮬레이션을 하였다.

키워드 : 셀룰라 이동 컴퓨팅, 채널할당 기법, 채널차용 기법, 유전 알고리즘

Abstract In the static channel assignment scheme for cellular mobile computing environment, each cell has a fixed number of channels and supports a service for a user's request entering to the cell. This scheme has an advantage of simplicity. However, this scheme has a disadvantage that can't control for hot cell problem. Therefore, to solve this problem, the "channel borrowing" concept is needed. In this paper, we propose a load balancing(channel borrowing, channel reassignment) approach using genetic algorithm. The purposes of using genetic algorithm in this paper are ① to find early a cell including an available channel and ② to decrease a number of probings and ③ to migrate to the cell after searching an available channel near upon optimality. To represent effectiveness of the proposed algorithm, we simulated various experiments.

Key word : Cellular mobile computing, Channel assignment method, Channel borrowing method, Genetic algorithm

1. 서 론

현재 셀룰라 통신의 도입으로 급속하게 무선 통신망이 확대되고 있으며 개인 통신 서비스(PCS : Personal

Communication Service) 사용자 및 규모 면에서 매우 빠른 증가추세를 보이고 있다. 이는 기존의 단순한 음성 신호의 전달뿐만 아니라 디지털 통신등 새로운 기능을 요구하고 있다. 또한 무선통신의 다양성으로 인하여 그 응용분야는 컴퓨터관련 분야 및 실세계에 적용될 수 있는 모든 문제로 확대되고 있다. 현재 휴대용 컴퓨터가 일반화되고 이와 관련된 소프트웨어 기술 및 무선 네트워킹 기술의 급속한 발전은 이동 컴퓨팅 환경을 실현 가능하게 하고 있다[1, 2, 3].

일반적으로 기술되고 있는 셀룰라 이동 컴퓨팅 시스템의 구조는 다음 그림 1과 같으며 셀이라 불리는 지역적 영역의 집단으로 구성되어 있다.

· 이 논문은 우송대학교 학술연구비를 부분적으로 지원 받았음.

[†] 비 회 원 : 천안대학교 정보통신학부 교수
 shlee@mail.chonan.ac.kr

^{**} 정 회 원 : 우송대학교 컴퓨터전자정보공학부 교수
 dwlee@allison.woosong.ac.kr

^{***} 종신회원 : 한남대학교 컴퓨터공학과 교수
 sglee@mail.hannam.ac.kr

논문접수 : 2001년 6월 4일
 심사완료 : 2002년 1월 23일

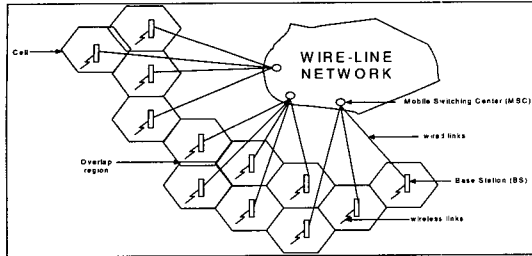


그림 1 셀룰라 이동 컴퓨팅 시스템 구조

위의 그림에서와 같이 이동 컴퓨팅 모델은 두 가지 종류 - 고정 호스트(fixed host) 및 이동 호스트(mobile host) - 로 구성된 컴퓨터들의 모임으로 구성된다. 고정 호스트는 기지국(base station) 및 MSC(mobile switching center)라 불리는 호스트 컴퓨터이다. 각 이동 호스트들은 하나의 기지국과 채널을 통하여 무선(wireless)으로 통신하게 되며 하나의 기지국이 담당하는 통신지역을 셀(cell)이라 한다. 따라서 하나의 셀내에는 여러 개의 이동 호스트들이 존재하며 이들은 무선으로 해당 기지국과 통신을 하게 된다. 또한 하나의 MSC는 여러 개의 기지국과 유선(wired)으로 통신하게 된다. 이같은 셀룰라 이동 컴퓨팅 환경에서 주파수 채널(frequency channel)의 재사용(reuse) 정도가 아주 중요한 문제로 대두되고 있다. 이는 매우 적은 주파수 채널을 이용하여 이동 호스트들을 지원해야 하기 때문이다. 따라서 채널의 재사용 정도를 극대화하기 위해 채널 할당(channel assignment) 방법에 관한 연구들이 진행되어 왔다[4, 5, 6, 7, 8, 9, 10, 11].

지금까지 셀룰라 이동 컴퓨팅 환경에서 연구되어 왔던 채널 할당 기법들은 크게 3가지 유형 - 정적(static), 동적(dynamic), 유연적(flexible) - 으로 연구되고 있으며 이들 각각은 다음과 같다.

먼저 정적 채널 할당 방법[4, 6, 9, 10]에서는 셀룰라 구조의 기본 단위인 각 셀에 고정된 개수의 채널들이 존재하여 해당 셀에 들어오는 요청들에 대한 서비스를 제공하게 된다. 또한 각 셀내에 포함된 채널들은 상호 채널 간섭(co-channel interface)이 허용될 수 있는 정도의 지역내에 있는 다른 셀에서 사용될 수 있다. 이 방법은 매우 간단하다는 장점을 지니고 있다. 또한 대체적으로 동적 채널 할당 기법보다 과부하 트래픽 환경하에서 더 좋은 성능을 보인다는 점이다[10]. 반면에 만일 해당 셀에서의 서비스 요청 개수가 할당된 채널의 개수보다 많다면 초과된 요청들은 대기 상태(block state)에 있게 되는 문제점이 발생한다. 또한 각 셀에 할당된 채널

의 개수가 변하지 않고 고정되어 있기 때문에 과다 셀 문제(hot spot problem)[11]를 제어할 수 없는 단점을 보인다. 따라서 이동 컴퓨팅의 근본 목적인 시간 및 장소에 구애없이 없이 들어온 요청들을 처리할 수 없는 근본 문제점을 보인다.

동적 채널 할당 방법[5, 7, 8, 9]은 정적 방법의 문제점을 개선하고자 하는 것으로 각 셀에는 미리 설정되어 있는 채널들이 없다는 점이 정적 방법과의 상대적인 차이점이다. 따라서 모든 채널들을 전역적 장소(global pool)에 모아 놓고 요청이 들어오면 해당 셀에 채널을 할당하게 된다. 이 방법에서는 채널 할당 비용 함수(channel assignment cost function)를 사용한다. 함수를 사용하는 목적은 요청이 들어온 셀에 할당될 채널을 결정하기 위한 것으로 전역적 장소에 있는 모든 채널들에 이 함수를 적용하여 가장 작은 함수 값을 갖는 채널을 셀에 할당하게 된다. 따라서 위에서 기술된 정적 방법에서의 문제점을 개선할 수 있는 잠재력을 갖는다.

마지막으로 유연적 할당 방법은 정적 방법 및 동적 방법을 혼합한 방법으로서 각 셀에는 고정된 개수의 채널이 존재하며 또한 할당된 채널보다 서비스 요청이 많으면 전역적 장소로부터 채널들을 할당받을 수 있는 방법이다.

위에서 기술된 채널 할당 방법들은 "채널 차용(channel borrowing)" 개념이 포함되지 않은 방법으로 시간과 장소에 따라 특정 지역에서의 서비스 요청이 동적으로 변하는 이동 컴퓨팅 환경에 능동적으로 대처하기가 어렵다. 따라서 채널 이전(channel migration)을 포함하는 할당 방법 즉, 부하균등(load balancing)을 포함하는 채널 할당 방법에 대한 연구가 이루어지고 있다.

[12]의 방법에서는 이웃 셀로부터의 채널 차용을 포함하지 않는 방법으로 대신 가입자(서비스 요청자)가 핸드오프(hand-off) 과정에서 현재의 셀에서 다른 셀로 이동하게 된다. 핸드오프란 연산 수행중인 이동 호스트가 다른 셀로 넘어가는 것을 의미하며 이동중인 이동 호스트는 새로운 셀의 기지국 및 MSC와 구조적으로 연산을 계속 수행하게 된다. [11]의 방법은 채널 차용 개념을 포함한다. 한 셀내에 포함된 채널들의 집합이 모두 이용될 때 채널 차용 메커니즘을 사용한다. 이때 채널은 오직 이웃 셀로부터 차용된다. 한 셀내의 채널 집합은 7개의 그룹으로 구성되어 있다. 한 그룹은 해당 셀의 사용자들에 대한 서비스를 전담하며 그 외의 6개 그룹내 채널들은 이웃 셀들로부터의 채널 요청을 위해 제공된다. 만일 하나의 채널 그룹내 채널 개수가 모두 사용된다면 하나의 채널을 사용하는 가입자는 다른 그

를 내 유휴 채널에 스위치될 수 있다. [13]의 방법 역시 채널 차용 개념을 포함한다. 유용한 채널들의 집합이 소모되기 전에 해당 셀에 의해 선택적 채널 차용(selective borrowing of channels)을 이용하여 저부하 셀(cold cell)로부터 과부하 셀(hot cell)로 채널들을 이전하여 작업을 수행하게 된다. 이전받은 채널들은 사용자의 우선 순위에 따라 할당된다.

[12]의 연구에서는 가입자가 셀과 셀 사이를 이동함으로써 채널 차용 개념을 모델화한 것이라 할 수 있다. 하지만 셀과 셀간을 이동하기 위한 많은 핸드오프가 발생하며 또한 상호채널 간섭이 증가하는 문제점을 갖는다. [11]에서는 이웃 셀내에 존재하는 채널들 중에서 유휴 상태인 채널을 차용하는 것으로 hot spot 문제에서는 심각한 성능 저하를 보인다. [13]에서는 사용자들의 우선 순위를 고려하기 때문에 많은 연산이 필요하며 또한 부하균등 수행시 최적의 대여자(lender)를 계산하기 때문에 상대적으로 많은 연산이 필요하게 된다.

본 논문에서는 간단하면서도 과부하 트래픽 환경에서 우수한 특성을 보이는 정적 채널할당 방법을 기반으로 하여 과다 셀 문제를 제어하기 위해 유전 알고리즘을 기반으로 하는 부하균등(채널 차용) 접근법을 제안한다. 본 논문에서 유전 알고리즘을 이용하는 목적은 ① 유휴한 채널을 조기에 발견하고 ② 조사 횟수를 줄이며 ③ 최적에 근접한 유휴 채널을 발견하여 해당 셀로 이전하기 위함이다.

본 논문의 구성은 다음과 같다. 2장에서는 본 논문에서 제안하고자 하는 내용, 즉 유전 알고리즘을 기반으로 하는 채널 재할당(부하균등) 알고리즘을 기술한다. 3장에서는 제안된 방법의 효율성을 보이기 위한 다양한 시뮬레이션 및 결과를 기술한다. 마지막으로 4장에서 결론 및 향후 연구방향에 대해 기술한다.

2. 유전 알고리즘 기반의 부하균등 알고리즘

본 장에서는 이동 컴퓨팅 시스템에서 채널의 부하균등을 위한 유전 알고리즘 접근법을 다룬다. 따라서 유전 알고리즘의 원리를 간략히 기술한 후 필요한 적합도 함수의 정의, 이동 시스템내 각 셀들에 대한 유전적 코딩 방법 및 각 셀의 부하 정도를 측정하기 위한 척도를 정의하고 유전 알고리즘을 기반으로 하는 부하균등 알고리즘들을 기술한다.

2.1 유전 알고리즘

유전 알고리즘은 하나의 탐색 알고리즘으로서 탐색공간에 대해 어떠한 이전지식(priori knowledge)도 이용

하지 않는다. 이러한 특성으로 인해 알고리즘이 간단하고 견고(robustness)하며 일반적이기 때문에 많은 응용 문제들에 이용되고 있다.

기본적인 유전 알고리즘은 선택(selection), 교배(crossover), 돌연변이(mutation)라는 연산자들을 포함한다. 이들 연산자는 많은 임의성을 바탕으로 하고 있으며 이러한 임의성이 유전 알고리즘의 견고성 및 일반성(generality)을 유지하게 하는 주 요인이 되고 있다.

유전 알고리즘은 생물학적 유전학(genetics) 및 자연선택(natural selection)을 모델화하여 최적화를 수행하는 방법[14, 15, 16, 17, 18, 19]으로서 수많은 세대를 거쳐 개체(individual, chromosome, string)들로 구성된 하나의 집단(population)이 갖는 특성을 유지하고 조정하여 최적화 과정을 수행한다. 이러한 과정을 통하여 연속적인 집단들이 바람직한 특성들을 포함하는 개체들을 생성하도록 한다. 따라서 궁극적으로 유전 알고리즘은 여러 세대를 거치면서 연속적으로 "집단개선(population improvement)"을 형성하게 된다.

본 논문에서는 유전 알고리즘을 이용하여 부하균등을 위한 셀을 결정하며 이때 결정된 셀은 유전 알고리즘의 특성에 따라 여러 세대를 거치면서 이루어지는 집단개선을 통해 결정된다.

2.2 부하 척도(load measure)

셀룰라 이동 컴퓨팅 시스템에서 부하균등은 과부하 셀에 대한 서비스 요청을 만족시키기 위하여 저부하 셀로부터 채널을 빌려오는 것을 의미한다. 따라서 하나의 셀이 서비스 요청을 받았을 때 저부하 셀인지 과부하 셀인지를 결정할 수 있는 판별식이 필요하다. 본 논문에서의 판별식은 다음과 같은 식 (1)을 정의하여 사용한다.

$$DegreeofColdness_i = \frac{No - Available - Channels}{No - Total - Channels} \quad (1)$$

($0 \leq i \leq n-1$, n 은 셀 개수)

위의 식 (1)에서 사용된 변수 $DegreeofColdness_i$ 는 각 셀의 부하척도를 의미하며 이를 기준으로 해당 셀이 저부하인지 과부하인지를 결정한다. 또한 $No-Total-Channels$ 는 각 셀에 포함된 전체 채널의 개수를 의미하며 $No-Available-Channels$ 는 각 셀에 포함된 가용할 수 있는 채널들의 개수를 의미한다.

이동 컴퓨팅 시스템에서 채널들에 대한 부하균등을 위해 과부하 셀로부터 저부하 셀로 채널을 이전하기 위한 이전 정책(migration policy)은 임계값 정책(threshold policy)을 이용한다. 임계값 정책은 위의 식 (1)에서 기술된 부하척도 즉, $DegreeofColdness_i$ 를 기반으로 하여 시행된다. 이같은 임계값 정책은 해당 셀에

서비스 요청이 도착할 때 이루어진다. 본 논문에서는 해당 셀이 저부하인지 과부하인지를 판별하기 위해 2개의 임계값 - 상한 임계값(T_{up}), 하한 임계값(T_{low}) - 을 사용한다.

2개의 임계값을 이용하여 각 셀의 부하 상태를 결정하는 기준은 다음과 같다.

▶ 과부하 셀(Hot cell) : 만일 새로 들어온 서비스 요청으로 인해 해당 셀의 *DegreeofColdness_i*가 상한 임계값을 초과하는 경우

▶ 저부하 셀(cold cell) : 해당 셀의 *DegreeofColdness_i*가 하한 임계값 미만인 경우

본 논문에서는 이동 컴퓨팅 시스템의 부하균등을 위해 모든 셀에서의 부하상태를 3 단계로 나누어 표현하며 부하 상태는 각 셀의 *DegreeofColdness_i*로 결정된다. 본 논문에서 사용되는 3단계 부하 표현 방법은 다음 표 1과 같다.

표 1 3단계 부하 표현

부하상태	의미	기준(criteria)
cold	저부하	$DegreeofColdness_i < T_{low}$
middle	정상부하	$T_{low} \leq DegreeofColdness_i \leq T_{up}$
hot	과부하	$DegreeofColdness_i > T_{up}$

만일 해당 셀이 과부하인 경우 해당 셀의 기지국(base station)은 저부하 셀로부터 채널을 이전받기 위해 부하균등 알고리즘을 수행한다.

2.3 표현(representation, or coding) 방법

셀룰라 이동 컴퓨팅 시스템내 각 셀은 유전 연산자가 적용될 자신의 집단(population)을 갖는다. 집단은 여러 개의 개체(individual)들로 구성되며 개체는 개체 표현 방법에 따라 다양하게 표현될 수 있다. 지금까지 이용되고 있는 개체 표현 방법으로는 크게 3개 - 이진(binary) 표현, 문자(character) 및 실수(real) 표현, 트리(tree) 표현 - 로 분류된다. 본 논문에서는 가장 일반적으로 이용되고 있는 이진 표현 방법을 사용한다. 이진 표현 방법에 의해 집단내 개체들은 이진 코드 벡터로 표현될 수 있으며 이는 $\langle c_1, c_2, \dots, c_n \rangle$ 으로 표현될 수 있다. 이진 코드 벡터의 내용은 궁극적으로 이전 요청 메시지가 전송 될 셀들의 집합을 의미한다. 만일 이전 요청 메시지가 특정 셀에 전송되면 c_i 는 '1'로 설정되고 그렇지 않으면 c_i 는 '0'으로 설정된다.

위의 내용을 이동 컴퓨팅 시스템내 셀들이 15개일 때의 예를 들면 다음 그림 2와 같다.

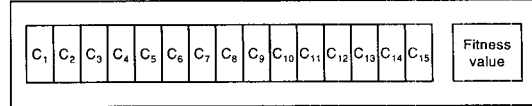


그림 2 유전 표현

2.4 적합도 함수(fitness function)

유전 알고리즘에서 적합도 함수는 한 집단에 포함된 개체들을 평가하기 위한 요소로서 각 응용 문제에 종속적인 특성을 포함한다. 이동 컴퓨팅 시스템의 특성으로 과부하인 셀에서 사용자 서비스 요청이 들어올 때 먼저 고려해야 하는 사항은 가장 인접한 저부하 셀(cold cell)이 존재하는지 여부이다. 동시에 고려되어야 하는 요소로는 셀의 부하 정도(degree)로서 부하 정도가 낮으면 낮을수록 유휴 상태에 있는 채널들이 많이 존재할 수 있기 때문이다. 따라서 다음과 같은 식 (2)로 적합도 함수를 정의할 수 있다.

$$f_i = \frac{1}{\alpha \times NoMSG + \beta \times Distance + \gamma \times DegreeofColdness} \quad (2)$$

($0 \leq i \leq m-1$, m 은 스트링 개수)

위의 식에서 사용된 매개변수들의 의미는 다음 표 2와 같다.

표 2 매개변수 의미

매개변수	의미
NoMSG	▶ 전달할 메시지 개수 - 메시지의 개수가 적을수록 효율적
Distance	▶ 현재 셀로부터 저부하 셀까지의 거리 - 거리가 짧을수록 효율적
DegreeofColdness	▶ 셀의 부하 정도 - 셀의 부하가 적을수록 채널을 빌려 줄 확률 증가

식 (2)에서 사용된 NoMSG는 전송될 요청 메시지들의 처리 시간으로 전송될 요청 메시지의 개수와 각 메시지의 처리시간의 곱으로 정의할 수 있으며 다음과 같은 식 (3)으로 나타낼 수 있다. 본 논문에서의 처리시간은 모든 셀에서 메시지를 처리하는 단위시간(time unit)은 동일하다고 가정한다. n 은 이동 시스템에 포함된 전체 셀의 개수를 의미한다.

$$NoMSG = \sum_{k \in X} (ReMN_k \times timeunit) \quad (3)$$

$$x = \{i | v_i = 1 \text{ for } 0 \leq i \leq n-1\}$$

식 (3)에서 사용된 $ReMN_k$ 는 k 번째 유전 인자의 값이 '1'이기 때문에 대응되는 셀로 요청 메시지를 보낸다

는 의미를 갖는다.

또한 식 (2)에서 사용된 *Distance*는 과부하 셀로부터 선정된 개체내 '1'로 설정된 각 비트에 대응되는 저부하 대상 셀들까지의 메시지 전송 시간의 합을 의미한다. 따라서 다음과 같은 식 (4)로 정의될 수 있다. *n*은 이동 시스템에 포함된 전체 셀의 개수를 의미한다.

$$Distance = \sum_{k \in x} EMTT_k \quad (4)$$

$$x = \{i | v_i = 1 \text{ for } 0 \leq i \leq n-1\}$$

식 (4)에서 사용된 *EMTT_k*는 과부하 셀로부터 *k*번째 셀 까지의 메시지 전송 시간을 의미한다.

마지막으로 식 (2)에서 사용된 *DegreeofColdness*는 선정된 개체에서 '1'로 설정된 비트에 대응하는 셀들의 전체 부하량을 의미하며 따라서 부하 상태가 저 부하인 개체가 선정되어 과부하 셀로 채널을 이전할 수 있는 저부하 셀이 용이하게 발견될 수 있다. 수식으로 표현하면 다음 식 (5)와 같다. *n*은 이동 시스템에 포함된 전체 셀의 개수를 의미한다.

$$DegreeofColdness = \sum_{k \in x} (DegreeofColdness_k) \quad (5)$$

$$x = \{i | v_i = 1 \text{ for } 0 \leq i \leq n-1\}$$

궁극적으로 위에서 정의된 적합도 함수에 의해 최대 값을 갖는 집단내 개체가 선정되며 선정된 개체의 내용에 의해 '1'로 설정된 비트에 대응되는 해당 셀로 채널을 요청하기 위한 요청 메시지들을 보내게 된다.

정의된 식 (2)는 다목적 함수(multi-objective function)로 각 매개 변수의 영향이 어느 한쪽으로 편중되어 반영되어서는 안되며 전체적으로 영향을 균등하게 주어야 한다. 따라서 3개의 매개 변수들에 대한 가중치를 사용하기 위해 α, β, γ 를 이용한다. 가중치와 관련된 내용은 다음 장에서의 실험 부분에서 자세한 내용을 기술한다.

2.5 유전 알고리즘에 의한 부하균등(채널 재할당) 알고리즘

이동 시스템내 해당 셀이 과부하 상태이면 다른 셀로부터 채널을 이전받기 위해 유전 알고리즘을 기반으로 하는 부하균등 알고리즘을 수행한다. 부하균등 알고리즘을 수행하는 시점은 해당 셀에 서비스 요청이 들어올 때 수행된다. 해당 셀에 서비스 요청이 들어오면 자신의 부하 상태를 판별하기 위해 부하 정도를 측정한다. 측정된 결과가 과부하이면 다른 셀로부터 유휴 채널을 이전받기 위해 유전 알고리즘 기반의 부하균등 알고리즘을 수행한다.

본 논문에서의 부하균등 알고리즘은 5개 모듈 - 초기

모듈(Initialization), 부하측정(Check_load), 개체 평가(Individual_evaluation), 유전 연산 (Genetic_operation), 메시지 평가 (Message-evaluation) - 로 구성된다.

또한 유전 연산 모듈은 3개의 부 모듈 - 돌연변이 (mutation), 선택(selection), 교배(crossover) - 로 구성되어 있으며 알고리즘으로 표현하면 다음 표 3과 같다.

표 3 부하균등을 위한 유전 알고리즘

```

Algorithm : Genetic-Algorithm_for_Load_Balancing
{
    Initialization()
    while( Check_load() )
    {
        Individual_evaluation();
        Genetic_operation();
        Message_evaluation();
    }
}

Procedure Genetic_operation()
{
    Mutation();
    Selection();
    Crossover();
}
    
```

Initialization()에서는 한 집단을 형성하는 모든 개체들이 중복성 없이 임의로 형성되게 된다.

Check_load()에서는 해당 셀에 하나의 서비스 요청이 들어올 때 해당 셀의 부하정도를 측정하여 저부하인지 과부하인지를 결정하는 기능을 수행한다. 결정 기준으로는 *DegreeofColdness*을 이용한다. 만일 부하 상태가 과부하(hot cell)이면 채널들에 대한 부하균등을 위해 다음의 수행을 계속하게 된다. 이를 알고리즘화하면 다음 표 4와 같다.

표 4 부하측정 모듈

```

Check_load()
if (service request arrives at cell Ci)
{
    if (DegreeofColdnessi of Ci > Tup)
        /* Tup : 상한 임계값 */
        let Ci be a hot cell;
    else wait until another service requests;
}
    
```

Individual_evaluation()에서는 해당 집단에 포함된 개체들을 위에서 정의된 식 (2)를 이용하여 평가한다.

다음으로 Genetic_operation()에서는 해당 집단에 포함된 각 개체들에 대해 다음과 같이 적용되며 적용된 후에 새로운 개체들로 구성된 하나의 집단이 형성된다.

▣ Mutation(돌연변이)

돌연변이 연산은 각 개체들에 대해 적용된다. 먼저 첫 번째 개체가 선정되어 하나의 복사 본(copy version)을 만든다. 이어서 복사 본에 대해 돌연변이 연산을 수행하며 복사 본에 대해 적합도 함수를 이용하여 적합도(fitness value)를 측정한다. 다음으로 원래 개체와 돌연변이 연산이 적용된 후의 개체에 대한 적합도를 비교하여 복사 본에 해당하는 개체의 적합도가 높으면 원래 개체를 복사 본 개체로 대체시킨다. 다음으로 두 번째 개체에 대해 위에서 기술된 과정 즉, 복사 본 형성, 돌연변이 연산 적용, 적합도 함수에 의한 평가, 비교 과정을 수행한다. 해당 집단에 포함된 모든 개체들에 대해 이같은 과정을 반복 수행하게 된다.

따라서 궁극적으로 돌연변이 연산을 적용한 후의 각 개체들은 원래의 개체들보다 우수한 특성을 유지하는 개체들로 구성되게 되는 효과를 갖는다. 위의 과정을 알고리즘화하면 다음 표 5와 같다.

표 5 돌연변이 연산 알고리즘

```

Sub-module : Mutation()
for (i=1; i<= total-individual-number; i++)
/* total-individual-number : 전체 개체의 개수 */
{
select individual[i];
generate copy version of the selected individual[i];
apply mutation operator to the individual[i];
evaluate the mutated new individual;
if (fitness of new individual > fitness of original
individual)
original individual ← new individual;
}
    
```

▣ Selection(선택)

선택 연산은 위의 돌연변이 연산 적용 후 새로이 형성된 개체들에 적용된다. 이같은 선택 연산은 자연의 적자 생존의 원리를 모델화한 것으로 우수한 해를 갖는 개체는 살아남고 그렇지 않은 개체는 도태된다. 본 논문에서는 선택 방법으로 일반적으로 많이 이용되고 있는 "wheel of fortune" 방법을 이용한다.

이 방법에서는 하나의 개체가 다음 세대에 선택될 수 있는 기대회수를 나타내는 "기대치 값(expected value)"을 이용한다. 산출된 기대치 값은 해당 집단에 포함된 모든 개체의 평균 적합도로 각 개체의 적합도를 나누어

서 나온 결과를 의미한다. 이를 수식화하면 다음 식 (6)과 같다.

$$No-of_Selection = ROUND\left(\frac{fitness\ value}{average\ fitness\ value}\right) \quad (6)$$

▣ Crossover(교배)

교배 연산은 새로이 형성된 개체들에 적용된다. 이 연산에서는 우수한 부모의 유전 인자를 포함하는 자식들이 만들어지는 과정으로 선택과정에서 선정된 개체들 중에서 임의로 짝을 선정하여 임의의 위치에 있는 유전 인자를 상호 교환하게 된다. 본 논문에서는 일반적인 "one-point crossover" 방법을 이용한다[15][18][19].

각 개체들에 교배 연산을 수행한 후 적합도 함수를 이용하여 각 개체들을 평가하며 해당 집단에서 가장 높은 개체를 선정하여 선정된 개체의 내용 즉, 비트가 '1'로 설정된 셀로 채널을 요청하기 위하여 집단에 대응하는 기지국은 요청 메시지를 보내게 된다.

Message_evaluation()는 하나의 셀이 다른 셀로부터 메시지를 받을 때 수행된다. 만일 하나의 셀이 다른 셀로부터 메시지를 받으면 부하 척도인 DegreeofColdness을 기준으로 하여 자신의 부하를 측정하여 승인(accept) 메시지 혹은 거절(reject) 메시지를 되 보낸다.

예를 들어 서로 다른 대역폭을 갖는 9개의 셀들로 구성된 이동 시스템에서 셀 C₀에 새로운 서비스 요청이 들어왔다고 가정한 후의 수행 과정은 다음과 같다. 해당 셀 C₀는 거부하 셀을 발견하기 위해 유전 알고리즘을 수행한다. 그러면 적합도에 비례하는 확률로 가장 높은 적합도를 갖는 개체가 선정된다. 선정된 개체가 I₂라 하고 I₂

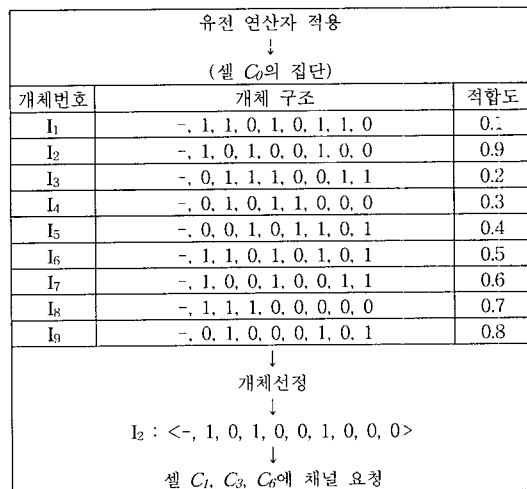


그림 3 알고리즘 처리과정

의 내용이 $\langle -, 1, 0, 1, 0, 0, 1, 0, 0, 0 \rangle$ 라 가정하자. 그러면 C_0 에 해당하는 기지국은 C_1, C_3, C_6 에 해당하는 기지국으로 여분의 채널을 이전받기 위해 이전요청 메시지를 보내게 된다. 그러면 이들 3개의 기지국은 자신의 부하를 $DegreeofColdness_i$ 을 이용하여 측정한다. 이 중에서 C_1 이 저부하라 가정하면 C_1 에 해당하는 기지국은 C_0 에 해당하는 기지국에 승인 메시지를 보내게 된다. 그러면 C_0 기지국은 C_1 기지국으로부터 하나의 채널을 이전받아 해당 서비스 요청을 처리하게 된다. 이같은 내용을 그림으로 간략히 표현하면 다음 그림 3과 같다.

3. 실험 결과

본 장에서는 제안된 알고리즘에 대한 효율성을 보이기 위해 다양한 시뮬레이션(simulation) 및 결과에 대해 기술한다. 본 논문의 시뮬레이션은 SUN 워크스테이션에서 C 언어를 사용하여 수행되었으며 시뮬레이션을 위해 가정된 사항은 다음과 같다.

첫째 전체 셀룰라 이동 시스템의 전체 부하는 90 퍼센트(percent)라 가정한다. 둘째 이동 시스템에 포함된 셀들의 개수는 60개로 설정하였고 각 셀에 포함된 채널의 개수는 20개라 가정하였다.

유전 알고리즘과 관련된 가정으로는 첫째 교배 확률(P_c)과 돌연변이 확률(P_m)은 0.7과 0.3으로 설정하였다. 또한 하나의 집단에 포함된 개체의 개수는 50으로 설정하였다. 이같은 이유는 일반적으로 많은 응용 문제에 적용하였을 때 평균적으로 가장 좋은 결과를 보이기 때문이다[14, 16, 19]. 본 논문에서의 돌연변이 확률은 일반적으로 다른 많은 응용들에 널리 사용되는 0.1보다 높은 값을 설정하였다. 이같은 이유는 본 논문 내용은 돌연변이 연산을 강조한 특성을 갖고 있기 때문이다.

제 2 장에서 언급된 적합도 함수에서의 식 (2)는 다목적 함수로 함수에서 이용된 가중치 α, β, γ 의 값은 다음과 같다. 설정된 각 매개변수의 값을 간략히 나타내면 다음 표 6과 같다.

표 6 매개변수 내용

셀의 개수	60
각 셀에서의 채널의 개수	20
교배 확률(P_c)	0.7
돌연변이 확률(P_m)	0.3
개체(individual)의 개수	50
α ($NoMSG$ 의 가중치)	0.75
β ($Distance$ 의 가중치)	0.15
γ ($DegreeofColdness$ 의 가중치)	0.1

[실험 1] 본 실험은 처리될 서비스의 개수가 5000개라 가정했을 때 하나의 채널을 대역할 수 있는 적절한 저부하 셀을 탐색하는데 필요한 탐색시간을 알아보기 위한 실험으로 결과는 아래 그림 4와 같다.

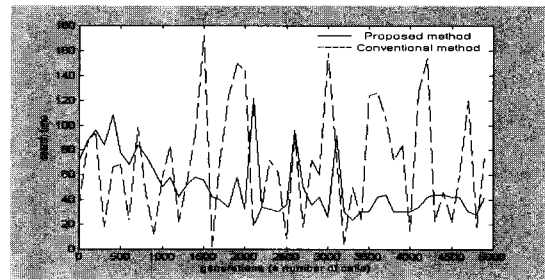


그림 4 저부하 셀 탐색 시간 비교

위 실험 결과로부터 알 수 있는 내용은 먼저 제안된 알고리즘의 성능이 기존의 방법[11, 13]보다 개선되었다는 것을 알 수 있다. 위의 그림에서 보여주는 것처럼 기존 방법보다 향상된 결과는 유전 알고리즘의 특성이 반영된 결과라 할 수 있다. 유전 알고리즘의 특성은 현재 세대(current generation)를 구성하는 개체들은 이전 세대(previous generation)의 개체들보다 좀 더 우수한 특성을 보인다는 점이다. 따라서 약간의 편차는 있지만 전체적으로 초기에는 높은 탐색 시간을 보이지만 세대가 지나면서 약간씩 낮아지는 현상을 위의 그림에서 볼 수 있다.

또한 기존의 방법에서는 임의의 접근 방법으로서 요청 메시지를 보낼 셀들을 결정하기 때문에 이의 특성을 반영하여 많은 편차를 보이는 것을 알 수 있다.

[실험 2] 본 실험에서는 특정 셀(C_2)에서의 가장 우수한 개체의 적합도의 수렴 정도(convergency)를 알아보기 위한 실험으로 결과는 다음 그림 5와 같다.

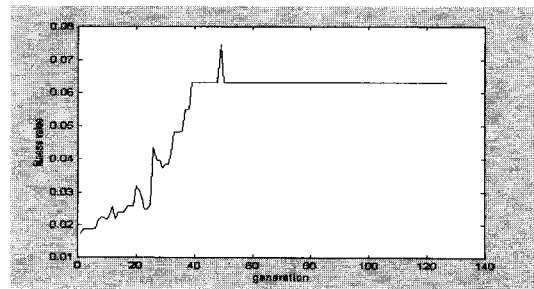


그림 5 셀 C_2 에서의 수렴성

본 실험을 통하여 알 수 있는 점은 5000개의 서비스 요청 중 130여 개가 셀 C_2 에 적용되었다는 점을 알 수 있다. 그러면서 약 100여 세대를 거치면서 전체적으로 수렴하고 있음을 알 수 있다. 또한 세대를 지나면서 위의 그림에서 편차를 보이는 것은 각 세대에서 가장 우수한 개체를 대상으로 하기 때문이다.

[실험 2-1] 본 실험에서는 셀 C_7 에서의 가장 우수한 개체의 적합도의 수렴 정도(convergency)를 알아보기 위한 실험으로 결과는 다음 그림 6과 같다.

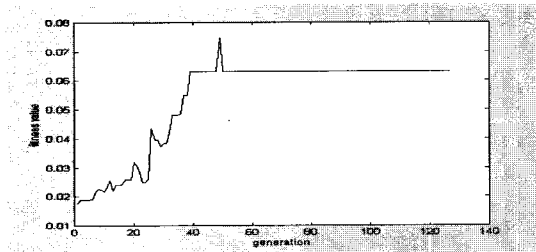


그림 6 셀 C_7 에서의 수렴성

본 실험은 셀 C_7 에 대한 가장 우수한 개체의 수렴성을 나타낸다. 결과는 위의 그림 5와 거의 유사한 결과를 보임으로서 본 유전 알고리즘을 기반으로 하는 부하균등 알고리즘의 적용 가능성을 보여주고 있다.

[실험 3] 본 실험은 셀 C_2 에 포함된 모든 개체들의 평균 적합도의 수렴 정도를 알아보기 위한 실험으로 결과는 다음 그림 7과 같다.

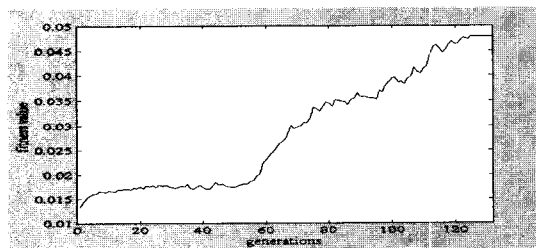


그림 7 셀 C_2 에서 모든 개체들의 평균 적합도

위의 실험 결과에서 알 수 있는 내용은 [실험 2]에서 보다 편차가 그리 심하지 않다는 점이다. 이는 한 집단에 포함된 모든 개체들의 평균 적합도에 기인 한 결과라 할 수 있다.

[실험 3-1] 본 실험은 셀 C_7 에 적용된 서비스 요청에 대한 모든 개체들의 평균 적합도의 수렴 정도를 알아보

기 위한 실험으로 결과는 다음 그림 8과 같다.

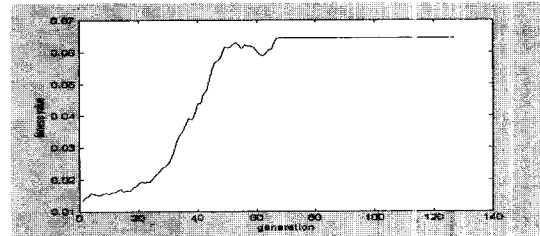


그림 8 셀 C_7 에서의 모든 개체들의 평균 적합도

본 실험 결과는 셀 C_7 에 적용된 서비스 요청에 대한 모든 개체들의 평균 적합도를 보여준 결과로서 위의 그림 7에서와 마찬가지로 수렴성을 보이고 있다.

[실험 4] 본 실험에서는 채널차용 횟수를 알아보기 위한 실험으로서 실험 결과는 다음 표 7과 같다. 실험 결과에서는 제안된 방법이 기존의 임의접근 방법보다 채널차용을 위한 기회를 증대시킬 수 있음을 보이고 있다.

표 7 채널 차용 회수 결과

서비스 개수	1500	3000	5000
접근 방법			
임의 접근 방법	1297	2582	4372
제안된 방법	1425	2925	4925

본 연구의 실험 결과를 종합해서 기술하면 먼저 실험 1에서 제안된 알고리즘이 저 부하 셀을 발견할 때 짧은 탐색시간을 나타내고 있으며 짧은 탐색시간을 보이면서도 실험 4에 나타난 것처럼 채널 차용 횟수에 있어서 많은 기회를 포함한다는 것을 보인다.

4. 결론

본 논문에서는 셀룰라 이동 컴퓨팅 시스템에서의 성능을 개선하기 위해 유전 알고리즘을 기반으로 하는 채널에 대한 부하균등 알고리즘을 제안하였다. 유전 알고리즘에 의한 부하균등을 위해 셀에 대한 부하적도를 정의하고 부하 표현 방법을 기술하였으며 적합도 함수를 정의하였다. 마지막으로 유전 알고리즘을 이용한 부하균등 알고리즘을 기술하고 이에 대한 성능 평가를 위해 다양한 시뮬레이션을 수행하였다. 시뮬레이션 결과에서는 서론에서 기술한 유전 알고리즘을 사용하는 목적에 부합하는 결론을 얻었으며 제안된 유전 알고리즘을 이용한 부하균등 알고리즘이 셀룰라 이동 컴퓨팅 환경에

적용될 수 있음을 보였다. 향후 연구과제로는 본 논문에서 정의된 적합도 함수에서 사용된 가중치 값이 성능에 많이 의존하는 경향이 있다. 향후 연구로서는 가중치 값이 성능에 의존하지 않는 연구가 필요하다.

참 고 문 헌

[1] D. Duchamp, "Issues in wireless mobile computing," *Proceeding of 3rd IEEE Workshop on Workstation Operation Systems*, pp. 1-7, Apr 1992.

[2] G. H. Forman and J. Zahorjan, "The Challenges of mobile computing," *IEEE Computers*, Vol. 27, No. 6, pp. 38-47, Apr 1994.

[3] T. Imielinski and B. R. Badrinath, "Wireless mobile computing: challenges in data management," *CACM*, pp. 18-28, 1994.

[4] V. H. Macdonald, "Advanced mobile phone service: The cellular concept," *Bell Syst. Tech. J.*, Vol. 58, pp. 15-41, Jan 1979.

[5] D. C. Cox and D. O. Reudink, "Increasing channel occupancy in large scale mobile radio systems: dynamic channel reassignment," *IEEE Trans. Veh. Technol.*, Vol. COM-26, pp. 218-222, Nov 1973.

[6] S. M. Elnoubi, R. Singh, S. C. Gupta, "A new frequency channel assignment in high capacity mobile communication systems," *IEEE Trans. Veh. Technol.*, VT-31, No. 3, Aug 1982.

[7] K. N. Sivarajan, R.J. McEliece, Ketchum, "Dynamic channel assignment in cellular radio," *IEEE Veh. Tech. Conf.*, pp. 631-637, Nov 1990.

[8] E. Del Re, R. Fantacci, G. Giambene, "Handover and Dynamic channel allocation techniques in mobile cellular networks," *IEEE Trans. Technol.*, Vol. VT-44, No. 2, May 1995.

[9] M. Zhang and T. S. Yum, "Comparisons of channel assignment strategies in cellular mobile telephone systems," *IEEE Trans. Veh. Technol.*, Vol. 38, Nov 1989.

[10] Z. Xu and P. B. Mirchandani, "Virtuality fixed channel assignment for cellular radio-telephone systems: a model and evaluation," *IEEE International Conference on Communications*, Chicago, 1992.

[11] H. Jiang and S. S. Rappaport, "CBWL: A new channel assignment and sharing method for cellular communication systems," *IEEE Trans. Veh. Technol.*, Vol. 43, No. 2, May 1994.

[12] J. Karlsson and B. Eklundh, "A Cellular Mobile Telephone System with Load Sharing - An Enhancement of Directed Retry," *IEEE Trans. Communications*, Vol. 37, No. 5, May 1989.

[13] S. K. Das, S. K. Sen, R. Jayaram, "A Dynamic Load Balancing Strategy for Channel Assignment Using Selective Borrowing in Cellular Mobile

Environment," *IEEE/ACM Conference on Mobile Computing and Networking*, New York, Nov 1996.

[14] Z. Michalewicz, *Genetic Algorithms+Data Structures=Evolution Programs*, Springer-Verlag, 1992.

[15] J. R. Filho and P. C. Treleaven, "Genetic-Algorithm Programming Environments," *IEEE Computer*, pp. 28-43, June 1994.

[16] J. Grefenstette, "Optimization of Control Parameters for Genetic Algorithms," *IEEE Trans. System, Man and Cybernetics*, Vol. SMC-16, No. 1, Jan 1986.

[17] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, 1989.

[18] M. Srinivas and L. M. Patnaik, "Genetic Algorithms : A Survey," *IEEE Computer*, pp. 17-26, June 1994.

[19] M. Srinivas and L. M. Patnaik, "Adaptive Probabilities of Crossover and Mutation in Genetic Algorithms," *IEEE Trans. Systems, Man and Cybernetics*, Vol. 24, No. 5, pp. 656-667, April 1994.

[20] S. Tekinay and B. Jabbari, "Handover and channel assignment in mobile cellular network," *IEEE Communication Magazine*, Nov 1991.



이 성 훈
한남대학교 컴퓨터공학과 졸업. 고려대학교 컴퓨터학과 졸업(석사). 고려대학교 컴퓨터학과 졸업(박사). 1998년 ~ 현재 천안대학교 정보통신학부 교수. 관심 분야는 유전알고리즘, Bioinformatics, 분산 시스템



이 동 우
1981년 고려대학교 전자공학과(공학사). 1984년 고려대학교 전자공학과(공학석사). 1996년 고려대학교 컴퓨터학과(박사 수료). 1995년 ~ 현재 우송대학교 컴퓨터전자정보공학부 조교수. 관심분야는 데이터 베이스, 시스템 통합, 분산 시스템



이 상 구
서울대학교 공과대학 전자공학과 졸업. 한국과학기술원 전산학과 졸업(석사). 와세다대학 전기전자컴퓨터공학과 졸업(박사). 1983~현재 한남대학교 컴퓨터공학과 교수. 관심분야는 퍼지이론, 컴퓨터 구조, 병렬처리