

전역 탐색 알고리즘을 이용한 유무선망의 최적화

오정근* · 변건식*

*동아대학교

Optimization of wire and wireless network using Global Search Algorithm

Jung-gun Oh* · Kun-sik Byon*

*Dong-A University

E-mail : ohjunggun@hanmail.net

요 약

이동 무선 통신 시스템을 설계할 때 기지국(BTS), 기지국 컨트롤러(BSC), 이동 교환국(MSC)의 위치는 매우 중요한 파라미터 중 하나이다. 무선 통신 시스템을 설계할 때 여러 가지 복잡한 변수들을 잘 조합하여 비용이 최소가 되도록 설계해야 한다. 이러한 문제를 해결하는데 필요한 알고리즘이 전역 최적화 알고리즘이며, 지금까지 전역 최적화 검색 기술로 Random Walk, Simulated Annealing, Tabu Search, Genetic Algorithm과 같은 조합 최적화 기술이 사용되어 왔다. 본 논문은 이동 통신 시스템의 기지국, 기지국 컨트롤러, 이동 교환국의 위치 최적화에 위의 4가지 알고리즘들을 적용하여 각 알고리즘의 결과를 비교 분석하며 알고리즘에 의한 최적화 과정을 보여준다.

ABSTRACT

In the design of mobile wireless communication system, the location of BTS(Base Transceiver Stations), BSC(Base Station Controllers), and MSC(Mobile Switching Center) is one of the most important parameters. Designing wireless communication system, the cost of equipment is need to be made low by combining various, complex parameters. We can solve this problem by combinatorial optimization algorithm, such as Simulated Annealing, Tabu Search, Genetic Algorithm, Random Walk Algorithm that have been extensively used for global optimization. This paper shows the four kind of algorithms which are applied to the location optimization of BTS, BSC, and MSC in designing mobile communication system and then we compare with these algorithms. And also we analyze the experimental results and shows the optimization process of these algorithms.

As a the channel of a CDMA system is shared among several users, the receivers face the problem of multiple-access interference (MAI). Also, the multipath scenario leads to intersymbol interference (ISI). Both components are undesired, but unlike the additive noise process, which is usually completely unpredictable, their space-time structure helps to estimate and remove them.

키워드

Optimization, wire and wireless network, Global Search Algorithm

1. 서 론

새로운 이동 무선 통신 네트워크 설계에 있어서 기지국(BTS), 기지국 컨트롤러(BSC), 이동 교환국(MSC)의 위치는 중요한 설정 중의 하나이다. 서비스 지역에서의 전파 확률 특성과 기지국이 위치할 잠재적

인 위치 목록이 장비의 가격(=기지국 숫자)을 최소화 하거나 서비스영역(=무선 커버리지)을 최대로 하는 방향으로 셀을 디자인하는데 사용된다. 이동 통신 네트워크 설계와 같이 여러 가지 변수가 복합적으로 이루어져 있는 경우에는, 일반적으로 조합 최적화 알고리즘으로 해결할 수 있다. 조합 최적화 문제에 자주 사용

되는 알고리즘에는 Random Walk (RW), Simulated Annealing (SA), Tabu Search (TS), 그리고 Genetic Algorithm (GA) 등이 있다.[4] 본 논문에서는 GA, SA, TS, RA를 이동 무선 통신 네트워크에 적용한 후, 그 결과를 비교 분석하였다.[2]

II. 여러 가지 알고리즘의 실행

2.1 문제 표현과 파라미터

본 논문에서는 실제의 부산광역시 서구(3Km × 3Km) 주변 서비스 지역의 무선 통신 네트워크를 설계하였다. 이 지역을 선택한 이유는 트래픽과 도시의 복잡도가 적절하게 분포되어 있어서 트래픽의 정도나 도시의 복잡도에 따른 기지국 위치 최적화의 성능을 잘 살펴볼 수 있고, 그에 따른 BSC나 MSC의 최적화 과정을 볼 수 있기 때문이다. 그림 1과 같이 최적화를 위해 이 지역을 한 격자가 60m인 50 × 50으로 하였다. 최적화 망을 설계하기 위한 조건으로 우선 BTS는 하나의 BSC에만 연결되어야 한다. 그리고 BSC에 연결되는 최대 BTS 개수(MAX_BTS)를 넘어서는 안 된다. 본 논문에서는 모든 BSC는 하나의 MSC에 연결되어 있다고 가정하였다.

본 논문에서는 유무선 최적화를 위하여 기지국 최적화로서 BTS의 위치를 최적화하였고, 망 토폴로지 최적화로서 BSC의 위치를 최적화시켰다. 먼저 기지국의 최적화에 대하여 살펴보도록 하자. 무선 전파 모델을 이동 통신 환경에 적용하기 위하여 로그-노멀 페이딩 모델을 사용하였다. 이 모델에서 계산되는 각 기지국으로부터의 P_{loss} (무선 경로 손실)는 2500개의 모든 점에서 계산된다. 그리고 차단값 $P^*=100\text{dB}$ 와 비교하여 만약 $P_{loss} \geq P^*$ 이면 그 지역은 감쇠가 심해서 전파가 되지 않는다고 판단한다. 그리고 트래픽이나 도심지의 복잡도에 따라 그 정도가 심한 지역, 적당한 지역 그리고 약한 지역으로 나누었다. 이는 그림 1에서 진한 회색, 회색 그리고 흰색으로 표시하였다. 트래픽은 그 서비스 지역의 인구와 통화 발생 확률을 곱한 것이고, 도심의 복잡도는 서비스 지역에서의 전파 방해 정도(페이딩의 발생 정도)를 나타낸다. 기지국에서 수용 가능한 사용자의 수는 일정하기 때문에 트래픽이 높고 도심지의 복잡도가 증가할수록 페이딩은 증가하므로 커버리지는 줄어들게 된다. 그러므로 트래픽과 도심의 복잡도, 거리에 대한 감쇠를 고려하여 그 정도가 심한 지역은 커버리지 범위를 300m, 적당한 지역은 600m, 거의 없는 지역은 1.2Km로 정하였다. 이와 같이 50개의 모든 점에 대하여 적용하면 각각의

무선 커버리지를 구할 수 있게 된다. 마지막으로 기지국 최적화를 구현하기 위하여 이렇게 구해진 무선 커버리지를 이용하여 코스트 함수를 결정하였다.

기지국 위치를 최적화 한 다음 망 토폴로지에 최적화를 적용한다. 우선 MSC의 위치가 결정된다. 다음으로 기지국의 위치와 MSC의 위치를 고려하여 BSC의 위치를 최적화한다. BSC 개수는 장비의 비용에 의해 제한된다고 가정한다. 망 토폴로지 최적화의 코스트 함수는 링크의 길이에만 종속된다. 또한 BSC-BTS 링크와 BSC-MSC 링크에 대한 비용이 상이하므로 두 링크에 대한 가중치 α 를 설정하여 코스트 함수에 반영하였다.

2.2 코스트 함수

우선 기지국 최적화에 쓰인 코스트 함수는 일반적으로 도심지 환경에서 쓰이는 전파모델인 로그-노멀 모델을 사용하였으며 다음과 같다.

$$f = k \times \frac{N_{BTS}}{R^\beta} \quad (1)$$

여기서, N_{BTS} 는 선택된 기지국의 숫자, R 은 선택된 기지국에 의해서 구해진 무선 커버리지 퍼센트, β 는 기지국의 숫자를 최소로 하거나 무선 커버리지를 최대로 하는 무게 변수, K 는 크기 조정 상수이다. 본 논문에서는 $K = 10^8$, β 는 3~5를 사용하였다.[3]

다음으로 망 토폴로지 최적화에 사용한 코스트 함수는 다음과 같다.

$$f = \sum_{i=1}^{N_{BTS}} l_{BTS, \rightarrow BSC(i)} + \alpha \sum_{j=1}^{N_{BSC}} l_{BSC, \rightarrow MSC} \quad (2)$$

여기서, $l_{BTS, \rightarrow BSC(i)}$ 는 i 번째 BTS와 BSC사이의 직접 링크 길이이며 BTS가 각 BSC에 할당된다. 그리고 $l_{BSC, \rightarrow MSC}$ 는 j 번째 BSC와 MSC사이의 직접링크 길이이며 α 는 위에서 설명한 것과 같이 BTS와 BSC사이의 링크에 대한 BSC와 MSC사이의 링크 길이의 가중치이다.

2.3 근방정의

근방정의란 하나의 점에서 다른 점으로 탐색하는 방법을 묘사한 것이다. 본 논문에서 쓰인 기지국 최적화의 근방정의는 다음과 같다.

- 근방정의 A : 랜덤하게 현재 스트링의 한 비트를 반전한다.
- 근방정의 B : 돌연변이 확률로 현재 스트링의 한 비트를 반전한다.
- 근방정의 C : 새로운 스트링을 완전 랜덤하게 생성한다.

망 토폴로지 최적화에 쓰인 근방정의는 다음과 같다.

- **BTS Move A** : BTS의 할당을 바꾸지 않고 MSC의 위치를 이동한다.
- **BTS Move B** : 두개의 BSC를 선택하여 BTS의 할당을 다른 BSC에 넘겨준다.
- **BTS Move C** : 두개의 BSC에서 각각 하나씩의 BTS를 추출하고 서로 할당을 바꾼다.

2.4 알고리즘의 설명 및 특징

2.4.1 Random Walk

이 알고리즘은 지역 탐색 알고리즘 중에서 가장 간단한 알고리즘이다. 각 반복마다 만들어진 탐색점은, 현재 시점에서 **코스트 함수**가 작거나 같으면 조건 없이 받아들이고, 그렇지 않고 더 높은 코스트 값을 가지게 되면 허용 확률 p 에 따라 조건적으로 받아들인다. 그러므로 이 알고리즘의 범위는 순수한 greedy ($p=0$) 탐색에서부터 완전한 Random 탐색 ($p=1$)까지 가능하다. p 의 값에 따라 알고리즘의 성능은 달라지며 여기서 p 가 너무 크면 알고리즘의 효력이 없어지고 p 가 너무 작으면 국소 최적에 빠질 위험이 있다.

2.4.2 Simulated Annealing

물체를 가열하였다가 냉각시키는 과정을 알고리즘에 응용한 것으로 탐색점이 현재의 코스트보다 낮은 코스트를 가지면 허용한다. 그러나 현재의 코스트 보다 높다면 허용 확률의 기준으로서 Metropolis criterion을 적용하여 선택한다. 여기서 적용된 허용확률은 온도 T 에 비례하고 이 온도 T 는 알고리즘이 진행됨에 따라서 점차 감소된다. Metropolis criterion은 다음과 같다

$$P_{x \rightarrow x'} = \min(1, \exp[-\frac{f(x') - f(x)}{T}]) \quad (3)$$

그리고 다른 중요한 변수에는 초기온도 T 와 쿨링율 α 가 있다. 여기서 초기 온도가 높으면 uphill 확률을 높일 수 있다. 그리고 쿨링율이 높으면 알고리즘은 느리게 진행하고 쿨링율이 낮으면 매우 빠르게 진행되므로 이것은 greedy 탐색과 비슷해진다.

2.4.3 Tabu Search

Tabu Search는 메모리의 개념을 알고리즘에 도입시킨 것으로 타부 목록에 이전의 값을 일정기간 저장하여 값을 비교하면서 최적화시키는 알고리즘이다. 여기서 중요한 변수로 타부목록 크기 V 와 보유기간 K 이다. 여기서 V 는 각 탐색마다 얼마나 많은 후보들을 고려할 것인가를 결정한다. 그리고 K 는 목록에 얼마나 오래 동안 보관할지를 나타내는 변수이다.

2.4.4 Genetic Algorithm

유전자 알고리즘은 매개변수의 최적화 문제를 자연의 진화원리에 배경을 둔 알고리즘으로서 변수의 표현을 유전자

형태로 바꾼 다음 교차나 돌연변이에 의해서 알고리즘을 진행하는 알고리즘이다. 중요한 변수는 집단의 크기 p 이며, 집단의 크기에 따라 다음 세대의 부모와 자식의 수가 결정되기 때문이다.[1]

III. 시뮬레이션 및 결과

3.1 기지국 최적화

초기 기지국 위치 50개 중 랜덤하게 25개의 기지국을 선택하여 시뮬레이션을 시작하였으며 각 알고리즘당 코스트 평가 횟수는 동일하게 설정하였다. 그리고 모든 알고리즘은 동일한 코스트에서 출발하여 공정성을 높이도록 하였다.

최적화 알고리즘을 적용하기 전에는 25개의 기지국으로 95.72%를 커버할 수 있었고, 코스트는

$$\text{Coverage percent} = 2393/2500 = 95.72\%$$

$$\text{Cost} = 10^8 \times \frac{19.4}{95.72^{4.5}} = 23.62$$

이다. 그러나 하나 이상의 기지국에 의해 오버랩되는 지역이 많고 트래픽이 많은 지역에도 서비스가 되지 않는 지역이 많음을 확인하였다. 이는 25개보다 더 적은 기지국을 가지고도 이와 비슷한 서비스 면적을 얻을 수 있고, 트래픽이 많은 지역을 우선으로 하여 더 적절하게 최적화시킬 수 있음을 알 수 있다.

최적화를 수행한 결과는 그림 1과 같다. 코스트는 구해보면 다음과 같다.

$$\text{Coverage percent} = 2325/2500 \times 100 = 93\%$$

$$\text{Cost} = 10^8 \times \frac{5.8}{93^{4.5}} = 8.0399$$

최적화 결과 기지국 수는 13으로, 커버리지 퍼센트는 93%로 줄어들었고, Cost는 28.0399로 줄어들었다.

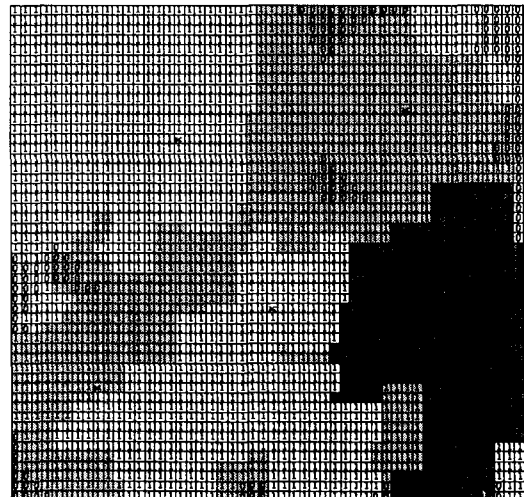


그림 1. 기지국 최적화 결과값

3.2 망 토폴로지 최적화

알고리즘의 코스트 평가 횟수는 100,000이고 다른 조건은 기지국 최적화 과정과 동일하다. BTS의 숫자는 100, MSC=1, MOVE A, B, C의 확률은 모두 0.3, 가중치 $\alpha=2$, MAX_BTS=20로 설정하였다. 최적화 전 초기값에서의 코스트 값은 411.284이고 BSC의 위치가 MSC와는 많이 떨어져 있고 할당되어 있는 BTS도 거리에는 상관없이 할당되어 있었다. 따라서 각 링크간의 거리 합이 상당히 커지며, 이로 인해 시스템의 설치 총 비용이 많이 들게 될 것이다. 최적화 결과는 그림 2와 같고 코스트 값은 169.13으로 작아졌다. 각 링크간의 거리 합은 최소가 되는 BSC의 위치를 설정한 것으로 볼 수 있다.

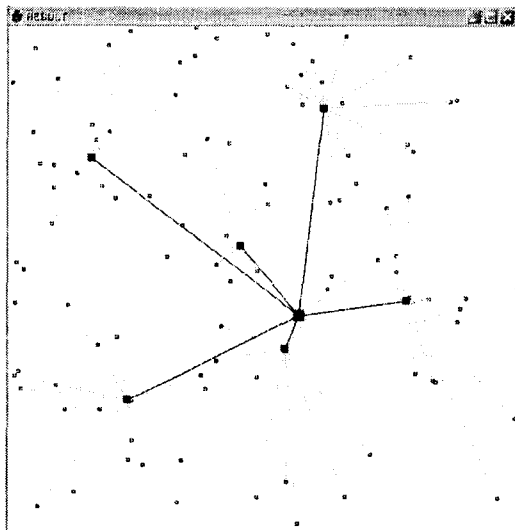


그림 2. 망 토폴로지 최적화 결과

IV. 결 론

4.1 기지국 최적화에서의 실행 결과비교

그림 3은 기지국 최적화와 망 토폴로지 최적화의 각 알고리즘 실행 결과이다. 그림에서 알 수 있듯이 기지국 최적화 결과는 TS와 RW(0.03)가 성능이 뛰어났으며 그 중에서도 TS가 제일 최적값까지 접근한 것을 볼 수 있었다.

4.2 망 최적화의 실행 결과 비교

그림 3에서 볼 수 있듯이 망 토폴로지 최적화에서는 GA와 TS가 성능이 뛰어났고 나머지 알고리즘들은 최적화근처에는 접근하였지만 성능이 좋지 않았을 알 수 있었다. 그리고 사용된 각각의 알고리즘 모두는 연산의 복잡성 때문에 연산량이 많아져 최적화 시간이 많이 소요되었다.

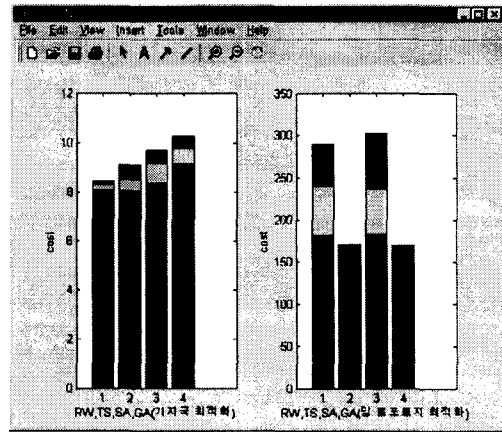


그림 3. 알고리즘 실행 결과

4.3 최종 솔루션과 향후과제

기지국 위치 최적화에서는 TS와 RA가 최적의 성능을 보였지만 망 토폴로지 최적화에서는 GA와 TS가 최적의 성능을 보였다. 일반적으로 TS와 RA는 국소 지역이나 연산량이 작은 경우에는 좋은 성능을 발휘 하지만 전역 탐색이나 연산량이 많은 경우에는 좋은 성능을 발휘하지 못한다. 따라서 향후에는 이러한 문제점을 극복하기 위하여 최적화 근방까지는 GA로 최적화를 시킨 다음 그 이후에는 TS나 RA를 이용하는 하이브리드 알고리즘을 적용할 생각이다.

참고문헌

- [1] 조성배, GA의 기초, 공학응용 및 인공생명 유전자 알고리즘, 1966.
- [2] Theodore S. Rappaport, Wireless Communications, Prentice Hall, 1996.
- [3] P. Calegari., "Genetic approach to radio network optimization for mobile systems," IEEE 47th Vehicular Technology Congerence vol. 2, pp. 755-9, 1997.
- [4] E. Aarts and J.K. Lenstra, Local search in Combinatorial Optimization, wiley, 1997.