

마이크로 유전자 알고리즘을 이용한 그래프 분할에 관한 연구

황태웅^o, 한치근^{*}

^o*경희대학교 컴퓨터공학과 알고리즘 연구실

e-mail: algorithm@khu.ac.kr, cghan@khu.ac.kr

Micro Genetic Algorithm Methods for Graph Partition Problem

Tae-Woong Hwang^o, Chi-Geun Han^{*}

^o*Algorithms Lab, Dept. of Computer Engineering, KyungHee University

● 요약 ●

그래프 분할 문제는 각각의 가중치가 주어진 에지와 노드를 정해진 목적에 맞게 몇 개의 그룹으로 분할하는 문제이다. 이 문제는 휴리스틱 방법으로 해결되어져 왔으나, NP-hard 문제로 인한 지역 최적해에 빠지기 쉬운 단점을 갖는다. 유전자 알고리즘이 해결 방법으로 제시되고 있는 가운데 단순 유전자 알고리즘에서 초기의 모집단 메모리(population memory)를 이용하여 적은 크기의 모집단을 생성하고 외부메모리에 최적해들을 저장하고 있어 GA의 효율성을 높이며, 다수의 지역 최적해에 빠지지 않게 하며 수렴 속도를 향상시키는 마이크로 유전자 알고리즘을 적용한다. μ -GA를 통해 본 논문에서는 클러스터들의 가중치를 비교적 동일하게 하는 GPP를 해결하고자 한다.

키워드: 그래프 분할(graph partition), 마이크로 유전자 알고리즘(micro genetic algorithm)

1. 서론

그래프 분할 문제는 전산과학 분야의 대표적 어려운 문제인 NP-hard 문제군에 속한다[1]. 현재까지 그래프 이분할 문제에 유전자 알고리즘을 결합하여 푸는 방법과 기존의 다중 분할의 문제점을 개선하여 좀 더 효율적인 다중 분할 알고리즘을 개발하는 일들이 진행되었다. 그래프 분할 문제(graph partition problem)는 무방향그래프(undirected graph)의 각 에지(edge)별 가중치 및 노드(node)별 가중치가 주어져 있을 때, 정해진 목적에 가장 부합하도록 노드를 몇 개의 그룹으로 분할하는 문제이다. 분할된 노드의 그룹을 클러스터(cluster)라고 부르며 클러스터는 일정한 조건을 만족하도록 제약이 주어진다.

이러한 그래프 분할 문제(GPP)는 반도체 레이아웃과 네트워크 배치, 행렬 계산 시간 단축, 프로세서들의 스케줄링 및 클러스터링(clustering), VLSI 설계 등 다양한 문제에 응용될 수 있고, 조합최적화(combinatorial optimization) 분야의 대표적 문제 중 하나라는 이론적 중요성 때문에 많은 연구가 이루어져 왔다[3]. 특히 최근에는 통신네트워크의 설계와 관련한 응용 예도 많이 알려지고 있다 이러한 GPP는 대부분 다음과 같은 공통점을 갖고 있다.

첫째로 그래프 분할 후에 에지의 연결도는 변하지 않아야 한

다. 이것은 전체적인 연결의 변화가 생기면 안되는 것을 의미한다. 둘째로 분할된 그래프 사이의 에지의 수는 특정 개수를 넘을 수 없다. 여러 목적과 조건으로 인하여 그래프의 분할된 집단 사이의 에지 개수가 너무 많으면 관료나 혹은 공간적인 문제가 발생할 수 있기 때문이다. 셋째로 분할된 그래프를 연결하는 에지의 비용(cost)은 낮아야 한다. 이것은 중요한 에지들이 분할되는 구역에 들어가면 위험도가 증가하기 때문이다.

GPP의 유형으로는 clustering 문제, k -cut 문제, multiterminal cut 문제, multicut 문제, sparsest cut 문제, 네트워크 공격문제, 네트워크 단절문제 등이 있다.

본 논문에서는 클러스터들의 가중치를 비교적 동일하게 하는 GPP를 해결하고자 한다. 이에 소수의 모집단으로 진화시켜 단순 유전자 알고리즘보다 수렴 속도 효율이 좋은 마이크로 유전자 알고리즘(micro genetic algorithm)을 적용하여 다수의 지역 최적해에 빠지지 않게 하며 수렴 속도를 향상시킬 수 있는 방법을 제안한다.

본 논문의 구성은 2장에서는 GPP에 대해서 정의를 하고, 3장에서 기존 연구 방법 및 일반적인 유전자 알고리즘에 대해 설명한다. 4장에서 GPP를 위해 제안된 μ -GA에 대해서 설명하고, 5장에서는 결론 및 추후 연구 방향에 대해서 논의한다.

II. 관련 연구

1. 문제 정의

본 절에서는 일반적인 그래프의 분할 문제를 수학적으로 정의 하기로 한다[3].

무향 그래프 $G=(V,E)$ (V :노드들의 집합, E :에지들의 집합)는 n 개의 노드를 갖고 있으며, 각 노드에는 가중치 w_i ($0 < w_i \leq p, i = 1, \dots, n, p > 0$)를 갖고 있다. 노드 i, j 간의 $cost$ 는 두 노드들의 가중치의 합을 의미한다.

$$c_{ij} = w_i + w_j, \quad i, j = 1, \dots, n$$

k 를 자연수라고 할 때, k -way partition 문제는 다음과 같이 정의된다.

$$\bigcup_{i=1}^k V_i = V, \quad V_i \cap V_j = \emptyset, \quad i, j = 1, \dots, k, i \neq j$$

일 때,

$$\sum_{a \in V_i, b \notin V_i} c_{ab} \leq p, \quad i = 1, \dots, k$$

를 만족하는 $V_i (i = 1, \dots, k)$ 로 분할하는데, 이때

$$f = \sum_{i=1}^k \sum_{a \in V_i, b \notin V_i} c_{ab}$$

를 최소화하는 문제로 정리된다. 이것을 그림으로 표현하면 그림 1과 같다

그림 1은 그래프가 3부분으로 나누어지는 것을 표현한다. 선의 두께는 에지의 비용을 표현한다. 나누어진 그래프를 연결하는 에지의 비용은 나누어진 그래프 안에 포함된 에지의 비용보다 낮다. 즉, 그래프를 연결하는 에지의 비용을 최소로 하는 문제이다[7].

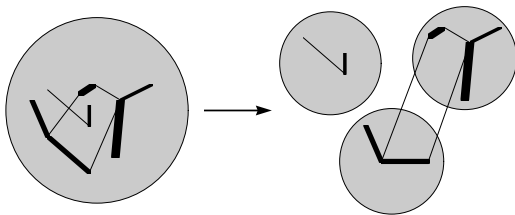


그림 1. 문제 정의
Fig. 1. Problem definition

2. 기존 연구

그래프 분할 문제를 해결하기 위해 기존의 여러 방법들이 사용되어져 왔다. Jones and Beltramo[5]가 제안한 greedy algorithm에서의 PMX 연산자가 있으며, 또 다른 하나는 ST(strict transmission) 연산자가 있다.

유전자 알고리즘(genetic algorithm)은 자연의 진화와 적자생존의 원리를 이용해 만든 알고리즘이다. 자연에서는 같은 종류 사이에도 우성의 특성과 열성의 특성을 가진 개체가 있고 적자생존의 원리에 따라 진화하고 발전한다. 먼저 초기 집단을 구성한다. 이 집단은 진화원리를 모방한 유전 연산자에 의해 점진적으로 개선되어 진다. 집단에서의 구성원인 각 개체의 우수성은 적합도(fitness) 함수에 의해 계량화 된다. 유전자 알고리즘(GA)에서는 적합도가 큰 개체들을 선택하고 이들 개체들 간에 유전정보가 교환되어 다음 세대의 집단이 형성된다.

GA은 최소화 및 최대화문제의 휴리스틱 해를 얻는데 효과적인 방법을 제공한다.

```

Procedure GA;
Begin
  initialize population P(0);
  evaluation P(0);
  t=1;
  repeat
    select P(t) from P(t-1);
    crossover P(t);
    mutate P(t);
    evaluate P(t);
  until (termination condition);
End.
    
```

그림 2. 단순 유전자 알고리즘
Fig. 2. Simple genetic algorithm

그림 2와 같이 GA에서 임의의 해들로 초기 모집단을 구성하고 교배나 돌연변이와 같은 유전 연산을 수행한다. 유전 연산은 해들을 교배시켜 더 나은 해를 탐색하게 하고, 돌연변이를 통하여 탐색 공간이 지역적인 공간에 머무르지 않도록 해준다. 이를 통해 새로운 해를 얻고, 기존의 해와 새롭게 얻은 해들 중 우수한 해들로 다시 모집단을 구성하게 된다. 따라서 초기 모집단으로부터 반복적인 수행을 통해 좋은 해로 수렴하도록 유도하는 것이 GA의 기본적인 목적이다. 이러한 과정은 GA이 최적 해와 근사한 해들을 얻을 수 있도록 해준다.

III. 본론

1. 제안된 마이크로 유전자 알고리즘 적용

1.1 마이크로 유전자 알고리즘

GA은 많은 개체를 운용할수록 최적값을 안정적으로 구할 수 있다는 장점이 있지만 개체 수만큼의 연산 시간이 증가하게 된다. 이러한 단점을 보완하고 개체 수 증가에 따른 부담을 해결하기 위하여 Krishnakumar[6]는 매우 작은 집단을 운용하는 마이크로 유전자 알고리즘(micro-genetic algorithm)을 제안하였다. 또한 적은 디자인 변수를 갖고 있는 문제에 높은 효율성을 보여주며 단순 유전자 알고리즘에 비해 더욱 단순한 유전 연산을 사용하므로 유전자 알고리즘에서와 같은 까다로운 매개변수(돌연변이 확률) 설정

정을 피할 수 있으며 적은 수의 개체를 사용하므로 개체 수 증가에 따른 해석시간을 줄일 수 있다. μ -GA에서 채용하고 있는 엘리트 보존전략은 집단 중에서 가장 적응도가 높은 개체를 그대로 다음 세대로 남기는 방법이다. a를 N세대까지 가장 좋은 개체라고 가정한다. 만일, 새로운 세대 N+1세대에서 a가 존재하지 않으면 N+1세대에서 가장 낮은 적응도를 가진 염색체와 교체하는 전략이다. 이 방법은 가장 좋은 해가 교차나 돌연변이에 의해서 파괴되지 않음으로 좋은 유전자의 확률적 손실을 방지해 줄 수 있다.

1.2 μ -GA 적용

1.2.1 μ -GA

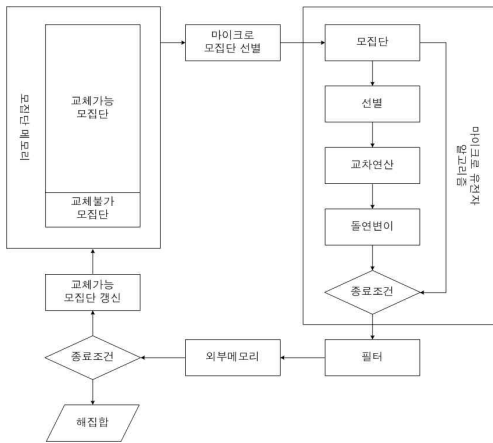


그림 3. 마이크로 유전자 알고리즘 흐름도
Fig. 3. Micro genetic algorithm flowchart

μ -GA은 초기의 모집단 메모리(population memory)를 이용하여 적은 크기의 모집단을 생성하고 외부메모리에 최적해들을 저장하고 있어 GA의 효율성을 높이며, 다수의 지역 최적해에 빠지지 않게 하며 수렴 속도를 향상 시킬 수 있도록 설계되어야 한다[8]. μ -GA의 흐름도는 그림 3과 같다.

1.2.2 유전자 구성

본 논문에서 제안하는 유전자는 그래프(G)에서 분할된 클러스터의 수를 3개로 구성하였다.

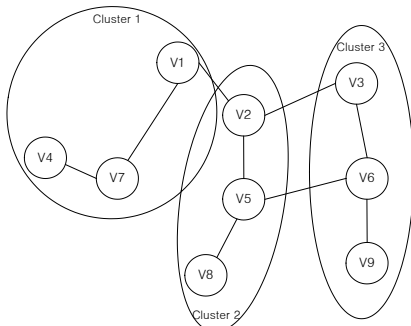


그림 4. 임의의 그래프 G
Fig. 4. Random of Graph, G

그림 5는 그림 4의 임의의 그래프 G를 유전자로 나타낸 것이다. (V1, V4, V7) 노드의 경우 클러스터 1에 포함되므로 1으로 나타나고, (V2, V5, V8) 노드의 경우는 클러스터 2에, (V3, V6, V9) 노드의 경우는 클러스터 3에 각각 나타난다.

V1	V2	V3	V4	V5	V6	V7	V8	V9
1	2	3	1	2	3	1	2	3

그림 5. 유전자 구성
Fig. 5. Genetic organization

1.2.3 선택 연산

선택 연산은 이전 세대에서 우성의 성질을 갖는 개체들을 선택하여 그 특성들을 다음 세대로 전달하는 역할을 한다. 교차 연산과 돌연변이가 연산이 선택 연산에 의해 선택된 스트링으로 새로운 개체들을 조합하기 때문에 선택 연산은 중요하다고 할 수 있다.

그리고 선택 연산은 우성과 열성간의 선택 기준이 되는 Fitness 함수에 의해 이루어진다. 일반적으로 선택 연산을 위해서는 각 개체의 선택 연산에 의해 선택될 가능성을 나타내는 확률값(f_k)을 계산해야 한다. f_k 가 k번째 스트링의 fitness 값을 가리키고, 한 세대 내의 전체 스트링의 수를 L이라고 하면, f_k 는 다음과 같이 정의된다. 이 때 f_k 가 크면 우성이 강한 것을 의미한다[4]. 그리고 이 값은 각 개체의 우열을 나타내는 척도가 된다. 이러한 값들로 다음 세대를 형성하는데 필요한 스트링을 선택하는 방법으로 가장 보편적인 Roulette Wheel 방법을 사용한다.

제한한 GPP를 위한 Fitness Function f_k 는 클러스터에 대한 총 차이의 합

$$\sum_{j=1}^k \sum_{i=1}^k |C_i - C_j| \quad (\text{단, } i \neq j)$$

에 대한 역수를 사용하며 다음과 같다.

$$f_k = \frac{1}{\sum_{j=1}^k \sum_{i=1}^k |C_i - C_j|} \quad (\text{단, } i \neq j)$$

1.2.4 교차 연산 및 돌연변이 연산

본 논문에서는 교차 연산자와 돌연변이 연산자로 이점 교차 연산자와 단일점 돌연변이 연산자를 이용한다.

이점 교차 연산자는 두 개의 부모 유전자에서 임의의 두 점을 선택하여 두 점 사이의 염색체를 교환하는 것으로 그림 6과 같다.

단일점 돌연변이 연산자는 한 개의 부모 유전자에서 임의의 한 점을 선택하여, 그 염색체가 k라면 k-1로 변환하는 방법이며 그림 7과 같다.

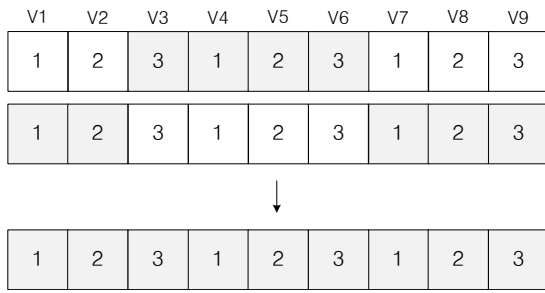


그림 6. 이점 교차 연산자
Fig. 6. Two point cross operator

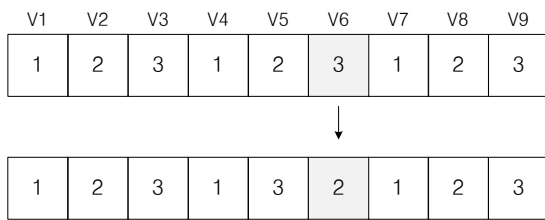


그림 7. 단일점 돌연변이 연산자
Fig. 7. One point mutation operator

IV. 결론

반도체 레이아웃과 네트워크 배치, 행렬 계산 시간 단축, 프로세서들의 스케줄링 및 클러스터링(clustering), VLSI 설계 등 다양한 문제에 응용되는 그래프 분할 문제(graph partition problem)는 무방향그래프(undirected graph)의 각 에지(edge)별 가중치 및 노드(node)별 가중치가 주어질 때, 정해진 목적에 가장 부합하도록 노드를 몇 개의 그룹으로 분할하는 문제이다. 분할된 노드의 그룹을 클러스터(cluster)라고 부르며 클러스터는 일정한 조건을 만족하도록 제약이 주어진다. 이 문제는 휴리스틱 방법으로 해결되어져 왔으나, NP-hard 문제로 인한 지역 최적해에 빠지기 쉬운 단점을 갖는다.

본 논문에서는 초기의 모집단 메모리(population memory)를 이용하여 적은 크기의 모집단을 생성하고 외부메모리에 최적해들을 저장하고 있어 GA의 효율성을 높이며, 다수의 지역 최적해에 빠지지 않게 하며 수렴 속도를 향상시키는 마이크로 유전자 알고리즘을 적용하였다.

앞으로 시뮬레이션을 통한 평가함수 적합도를 확인하고, 유전자 알고리즘에서의 스트링의 우열 판명은 다음 세대 구성에 중요한 역할을 하기 때문에 선택연산에 있어서 보다 나은 평가 방법의 연구가 필요하다.

참고문헌

- [1] G. Chartand and O. R. Oleccermann, "Applied and Algorithmic Graph Theory", McGraw-Hill, 1993.
- [2] Jonathan L. Herlocker, Lee R. Gordon, Jhon Riedl, "GroupLens : Applying Collaborative Filtering to Usenet News", Communication of ACM. Vol.40 No.3, March 1997. pp.87-97.
- [3] B.W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs", BSTJ, pp. 291-307, 1969.
- [4] D.E Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesly Publishing, 1989.
- [5] D.R.Jones and M.A.Beltramo, "Solving partitioning problems with genetic algorithms", (1991)
- [6] K. Krishnakumar, "Micro-genetic algorithms for stationary and non-stationary function optimization." SPIE Proceedings: Intelligent Control and Adaptive Systems, Vol. 1196, pp.289-296, 1989.
- [7] 성연중, 한치근, "탐색 공간 Smoothing을 이용한 그래프 분할에 관한 연구", 경희대학교 석사학위논문, 1997.
- [8] 전성화, 한치근, "멀티캐스트 라우팅을 위한 다목적 마이크로 유전자 알고리즘", 경희대학교 석사학위논문, 2006.