Genetic Algorithm HW #2 Max-cut

2015-21259 이현우

2016년 5월 17일

1 도입(Introduction)

1.1 지역 최적화 알고리즘

본 과제에서 확인하고자 하는 문제는 GA와 지역 최적화 알고리즘이 결합하였을 때 순수한 GA와 비교한 성능 변화이다.

문제에 적용한 지역 최적화 알고리즘은 다음과 같다.

Data: 이진 염색체 x_k
Result: 지역 최적화가 된 이진 염색체
$improved \leftarrow true$;
while $improved$ **do**
  $improved \leftarrow false$ **for** $i \leftarrow 1\,to\,n$ **do**
    if $\Delta(P[i]) > 0$ **then**
      $x_{P[i]} \leftarrow 1 - x_{P[i]}$;
      $improved \leftarrow true$;
    end
  end
end
최적해 출력;

Algorithm 1: 지역 최적화 알고리즘

위 지역 최적화 알고리즘은 임의의 이진 염색체 x_k에 대해서 생성된 순열에 따라 노드를 해당 노드가 속하지 않은 집합으로 옮겼을 때 적합도가 올라가는지 확인한 후 개선되면 개선된 염색체를 사용하는 것이다.

예를 들어, 염색체가 1101이고, 순열이 4213이라면 1101의 4번 노드를 다른 집합으로 보낸 1100으로 바꿔서 적합도를 계산하고 성능이 좋아지면, 1100을 사용하여 다음 순서에 대해 지역 최적화를 수행한다. 그러면, 1100의 2번 노드를 다른 집합으로 보낸 1000에 대해 적합도를 계산하고, 성능이 개선되면, 1000을, 개선되지 않으면 1100을 다음 순서에 대해 사용한다. 이런 식으로 순열이 끝날때까지 지역 최적화를 수행하면 지역 최적화가 된 이진 염색체가 나온다.

번호	선택	승자 비율	교차	변이율	교체	인구 수(N)	자손 수(K)	지역최적화 여부
1	토너먼트	0.6	reverse	0.1	하위 K개 대체	1000	1	O
2	토너먼트	0.6	reverse	0.1	하위 K개 대체	1000	1	X
3	토너먼트	0.6	reverse	0.1	하위 K개 대체	1000	800	O
4	토너먼트	0.6	reverse	0.1	하위 K개 대체	1000	800	X

표 1: 실험 번호별 연산자 및 파라미터 값

번호	평균값	최대값	최소값	표준편차
1	377.47	378	370	1.71
2	351.61	355	351	1.20
3	378.80	380	373	2.41
4	388.53	389	379	1.91

표 2: 100개 노드 295개 간선 그래프에서의 비교

위 지역 최적화 알고리즘을 적용한 GA 기본 뼈대는 다음과 같다.

Data: 그래프, 전체 염색체
Result: 최적의 염색체
기본 초기화(입력 그래프 초기화, 인구(population) 초기화, 기본 파라미터 초기화 등);
while 정지 조건*(stop condition)*에 부합하지 않는 동안 수행 **do**
  for 한 세대*(K)* 개수 만큼 **do**
    두 개의 부모해 선택(selection);
    교차 연산(crossover) 수행;
    변이 연산(mutation) 수행;
    지역 최적화(local_optimization) 수행;
  end
  대치 연산(replacement) 수행;
  인구(population) 정렬;
end
최적해 출력;

Algorithm 2: GA 알고리즘 수행 기본 뼈대

2 지역 최적화의 기본 실험 수행 결과

이 장에서는 지역 최적화 알고리즘을 GA 위에 적용하고, 이에 따른 결과를 통해 분석할 논제를 제시한다.

앞 장에서 논의한 지역 최적화 알고리즘의 효과를 살펴보기 위하여, 필자가 프로젝트 1을 수행하면서 주된 실험 대상 그래프로 사용하였던 정점 100개, 간선 495개의 그래프와 해당 그래프에서 가장 성능이 좋았던 연산자/파라미터 결합으로 우선 실험을 수행하였다. 그리고 동일한 연산자/파라미터 조합으로 정점 500개, 간선 4990개의 그래프 위에서 실험을 수행하였다. 모든 실험은 30회씩 동일한 시간 조건(170초) 하에서 수행되었다. 실험에 사용된 연산자와 파라미터 값은 표 1과 같으며 그래프별 실험 결과는 표 2, 3과 같다.

번호	평균값	최대값	최소값	표준편차	비고
1	3234.20	3252	3219	9.19	-
2	2811.10	2830	2797	9.58	-
3	3244.53	3258	3231	6.44	시간 내 수행 못함
4	2870.10	2890	2850	10.42	-

표 3: 500개 노드 4990개 간선 그래프에서의 비교

번호	수행 시간(초)	평균값	최대값	최소값	표준편차	인구 수	자손 수
1	11	356.97	358	355	0.98	1000	1
2	23	357.27	358	355	0.77	1000	2
3	34	357.50	358	356	0.62	1000	3
4	55	357.70	358	356	0.53	1000	5
5	112	357.97	358	357	0.18	1000	10

표 4: 100개 노드 495개 간선 그래프에서 단순 지역 최적화 멀티스타트 정도에 따른 비교

표 2와 표 3을 통해서 알 수 있듯이, 표 2의 3번과 4번 실험을 제외하고는 기본적으로 지역 최적화를 적용하면 품질이 상승하는 것을 확인할 수 있다. 또한 표 2와 표 3의 1번과 3번을 통해서 확인할 수 있듯이, 지역 최적화를 적용할 때 해의 개수는 큰 차이가 없다. 이 결과를 통해 다음의 문제를 다루고자 한다.

- 최적화를 적용하는 해의 개수에 따라 변화가 있는가? 그 이유는 무엇인가?
- 지역 최적화가 기본적으로 최대값을 올려준다면, 그 이유는 무엇인가?
- 지역 최적화를 적용했을 때 품질이 되레 떨어진 경우(표 2의 3과 4), 그 이유는 무엇인가? 단순히 지역최적화의 속도 때문인가? 그렇다면 동일한 수의 지역최적화를 수행하면 품질이 올라간다고 볼 수 있는가?
- 지역 최적화는 수렴 시간을 빠르게 하는가?
- GA를 적용하지 않으면서 지역 최적화를 수행하면 어떻게 되는가?

3 지역 최적화 분석

이 장에서는 앞 장에서 제기한 문제를 해결하면서 지역 최적화 자체와 GA와의 관계에 대해 고찰을 하고자 한다.

3.1 단순 지역 최적화의 결과

이 장에서는 지역 최적화 자체의 성능을 확인하기 위한 실험과 이의 결과에 대해 논한다. 여기서 "단순 지역 최적화"란 매회 임의의 난수를 생성한 뒤, 지역 최적화를 적용하는 것을 뜻한다. 즉, 인구 수가 1000이고, 자손 수가 800이면, 매 루프마다 임의의 800개의 난수를 생성하고, 각각에 대해 지역 최적화를 수행하면서 품질이 좋지 않은 인구를 대체해 나가는 것이다. 실험은 100개 노드 495개 간선 그래프에서 수행하였으며, 해의 개수는 1개, 2개, 3개, 5개, 10개로, 지역 최적화는 1000회 적용하였다. 그 결과는 표 4에 정리되어 있다.

해 개수	순수 GA		단순 지역 최적화		하이브리드 GA	
	평균값	표준편차	평균값	표준편차	평균값	표준편차
1	300.90	5.99	356.97	0.98	358.70	3.03
2	308.53	3.56	357.27	0.77	361.00	4.14
3	313.70	4.57	357.50	0.62	361.17	2.61
5	320.00	3.82	357.70	0.53	363.90	3.69
10	328.97	3.54	357.97	0.18	368.07	3.13

표 5: 100개 노드 495개 간선 그래프에서 GA 여부에 따른 지역 최적화 적용 결과 비교

결과를 통해 알 수 있듯이, 해의 개수가 1개, 2개, 3개, 5개, 10개로 해의 개수가 증가할 수록 평균은 다소 증가하고, 분산은 감소하는 것으로 확인되었다. 해의 개수가 증가할수록 평균이 증가하는 결과가 나타난 이유는 최적화를 적용하는 난수의 개수가 증가하였기 때문이다. 이로 인해 전반적인 품질이 상승한 것으로 보인다. 이는 해의 개수에 따른 결과이기 보다는 수행된 양과 시간이 길었기 때문이다.

지역 최적화는 결과를 통해서 확인할 수 있듯이, 초기에 최적화를 통해 어느 정도의 값에 도달하면 더 이상 품질을 상승 시키는 것이 어렵다. 이는 좋은 형질을 유지하여 개선해 나가는 GA와 달리 지역 최적화는 계속 임의의 난수를 취급하기 때문에, 좋은 품질의 해를 얻을 가능성은 좋은 품질을 만드는 난수를 만들 가능성과 동일하며, 이는 세대를 거듭한다고 해서 이 가능성이 증가하지는 않는다.

3.2 GA 여부와 지역 최적화 여부에 따른 결과

이 장에서는 GA 여부와 지역 최적화 여부에 따른 멀티스타트 알고리즘의 비교 분석을 수행한다. 이는 GA와 지역 최적화 알고리즘 사이의 관계 및 결과를 확인하기 위함이다. 여기서 GA가 없는 지역 최적화란 앞 장에서 실행한 단순 지역 최적화를 의미하며, GA가 있는 지역 최적화는 2장 기본 실험에서 수행하였던 Hybrid GA를 의미한다. 또한 지역 최적화가 없는 GA는 순수 GA에 해당한다. 이 비교 실험은 100개 노드 495개 간선 그래프에서 수행하였으며, 동등한 비교를 위해 지역 최적화는 1000회 적용하였다. GA를 적용한 것과 비교한 결과는 표 5에 정리되어 있다.

표를 보면 알 수 있듯이, 동일한 루프 횟수에서 품질의 결과는 순수 GA, 단순 지역 최적화, 하이브리드 GA 순이었다. 특이할 점은, 순수 GA와 하이브리드 GA는 해의 개수(자손의 수)가 늘어날 수록 품질이 증가하는 반면에, 단순 지역 최적화는 무관하다는 점이다. 이 결과는 좋은 형질(스키마)의 보존 여부에 따른 차이에서 비롯된다. GA의 성질은 좋은 형질을 보존시키면서 발전시키기 때문에, 시간이 흐르고, 자손 수가 많을 수록 대체로 더 좋은 품질을 보이지만, 단순 지역 최적화는 임의의 난수의 확률에 따르므로, 무관한 것이다.

3.1장의 결과에 비추어 볼 때, 시간이 더 흐른다고 해서 단순 지역 최적화의 최대 품질이 반드시 올라간다는 보장이 없기 때문에, 더 많은 시간을 수행하면, 순수 GA가 단순 지역 최적화를 넘어서는 순간이 확인될 것으로 보이며, 하이브리드 GA는 더 높은 품질을 보일 것으로 예측된다.

순수 GA와 하이브리드 GA의 차이는 지역 최적화 적용에 따른 차이로, 지역 최적화가 인구의 전반 품질을 빠르게 높임으로써 얻어진 결과이다.

번호	지역 최적화 여부	평균값	최대값	최소값	표준편차	인구 수	자손 수
1	O	372.90	378	368	2.24	1000	1
2	O	373.13	379	368	2.53	1000	2
3	O	372.97	377	368	2.20	1000	3
4	O	373.10	378	368	2.71	1000	5
5	O	373.60	380	369	2.91	1000	10
6	O	376.92	383	370	3.08	1000	100
7	O	372.97	380	367	2.68	1000	800

표 6: 100개 노드 495개 간선 그래프에서 멀티스타트 정도에 따른 비교

번호	지역 최적화 여부	평균값	최대값	최소값	표준편차	인구 수	자손 수
1	O	3234.20	3252	3219	9.19	1000	1
2	O	3234.70	3270	3216	9.93	1000	2
3	O	3234.50	3260	3216	11.01	1000	3
4	O	3234.03	3266	3219	10.05	1000	5
5	O	3230.50	3247	3215	7.61	1000	10
6	O	3236.43	3255	3219	9.80	1000	100
7	O	3244.53	3258	3231	6.44	1000	800

표 7: 500개 노드 4990개 간선 그래프에서 멀티스타트 정도에 따른 비교

3.3 동일한 시간 하에서 멀티 스타트 혼합 GA에 따른 지역 최적화의 수행 결과

이 하위 장에서는 임의의 여러 해(자손의 개수)를 생성하고 GA 루프 내에서 지역 최적화를 각각에 대해 수행할 때 나타나는 결과에 대한 분석을 진행한다. 우선 임의의 해의 개수의 변화에 따른 품질 변화에 대해서 100개 노드, 500개 노드에 대해 실험을 수행하였다. 수행된 연산자나 파라미터는 프로젝트 1에서 가장 최적의 해를 만들었던 조합과 동일하다. 즉, 인구(N)는 1000개 이고, 선택 연산은 토너먼트(승률 0.6)를 사용하며, 교차 연산은 리버스 교차 (절반을 자른 후, 50%의 확률로 각각을 뒤집고 결합), 0.1 비율의 변이 연산, 대치 연산은 하위 K개(자손의 개수) 대치이다. 정지 조건은 전체 인구 중 해의 품질이 동일한 염색체가 70% 이상이거나 수행 시작 후에 170초가 넘은 경우이다.

표 4와 표 5에서 보듯이, 임의의 해의 개수는 품질 변화에 큰 영향을 끼치지 않는 것을 확인할 수 있다. 한편으로는 해의 개수가 많아질수록, 당연히도 지역 최적화에 소모되는 시간이 급증하여, 주어진 예산 시간 내에서 수행이 어렵다는 것을 확인할 수 있었다. 이 결과로부터 두 가지에 대해서 생각해 보아야 한다. 하나는 해의 개수와 품질 변화가 상관관계가 없는 이유이고, 또 하나는 앞선 분석과 반대로, 지역 최적화 시간을 감소 시키면서 해의 개수를 활용할 수 있는 방안에 대한 고려이다.

임의의 해의 개수와 품질의 상관관계가 거의 없다는 것을 문제 공간 탐색 측면에서 고려해 보자. 초기에 필자는 실험을 수행하기 전 해의 개수와 지역 최적화 사이에는 연관관계가 없을 것이라고 조심스럽게 예측하였다. 이유인즉, 교차와 변이를 거친 후의 자손 각각의 지역 최적화하는 것은 동일한 봉오리(지역 해)를 모두가 빠르게 올라가는 것과 유사할 뿐이고, 결국 도착하는 봉오리는 동일할 것이기 때문에, 단순 중복으로 예측되었다. 다만, 해의 개수가 작은 범위에서 1개인가 2개인가, 3개인가는 유의미한 차이를 남길수도 있을 것이라고 예측하였다. 이유는, 봉오리(지역해)로 빠르게 오르도록 하는 지역 최적화 알고리즘은 설익은 수렴할 가능성이 언제나 존재하기 때문에 임의의 해의 개수는 이를 견제할 역할을 할수도 있겠다고 생각하였다. 수가 많으면 중복 계산도 많아지기 때문에 성능이 떨어질 것이기 때문에 작은 해의 개수 내에서 변화를 중심으로 관찰하였다. 그리하여 1개, 2개, 3개,

번호	평균값	평균시간(초)	최대시간(초)	최소시간(초)	표준편차
1	377.47	101.00	148	97	12.97
2	351.60	123.30	167	109	23.26
3	378.80	99.50	162	31	18.63
4	388.53	147.67	165	121	5.82

표 8: 100개 노드 295개 간선 그래프에서의 시간 비교

번호	평균값	평균시간(초)	최대시간(초)	최소시간(초)	표준편차
1	3247.27	118.07	159	41	35.87
2	2829.23	166.30	169	124	7.99
3	3253.73	1012.93	1039	993	17.81
4	2886.07	154.03	169	123	19.06

표 9: 500개 노드 4990개 간선 그래프에서의 시간 비교

5개를 선택하였으나, 결과는 위 표 4와 표 5에서 확인이 되듯이, 유의미한 차이를 남기지 못하였다. 이는 1개, 2개, 3개, 5개에서 차이를 만들기에는 부모의 선택 영역이 넓기 때문이라고 사료된다.

3.4 지역 최적화에 따른 최대값 수렴 시간 분석

이번 장에서는 지역 최적화 알고리즘이 얼마나 해들을 봉오리 (지역해)로 빠르게 올리는지 확인해보고자 한다. 즉, 지역 최적화를 적용하지 않았을 때 대비 지역 최적화를 적용하면 각 실험에서의 최대값을 얼마나 빨리 찾는지 확인하고자 함이다. 이를 통해서 지역 최적화에 의한 탐색 시간을 확인 및 다른 공간 탐색 가능성을 확인하고자 한다.

표 8과 표 9에서 확인할 수 있듯이, 지역 최적화를 수행하면, 해당 실험에서의 최대값을 얻기까지 수행 시간이 단축되는 것을 확인할 수 있다. 이는 매 임의의 주어진 해에 대해 지역 최적화를 수행하여, 주어진 해와 순열을 활용하여 만들 수 있는 가장 좋은 해를 찾기 때문이다. 다만, 이런식으로 만들어진 해는 어느 정도 품질이 보장된 구성이므로, 교차 연산이나 변이 연산의 결과 품질이 낮아질 가능성이 높다.

3.5 주어진 시간에서 품질을 높이기 위한 방안

표 2의 기본 실험에서 실험 3과 실험 4의 결과는 다소 예상과 다르다. 지역 최적화를 사용한 실험 3(하이브리드 GA)보다 지역 최적화를 사용하지 않은 실험 4(순수 GA)가 더 좋은 품질을 만들어낸 것이다. 그러나 이 이유는 표 5와 주어진 시간 동안 거친 세대 수를 통해서 이해할 수 있다. 우선 표 5를 통해서 알 수 있는 사실은 동일한 수의 루프(세대)를 거쳤을 때, 하이브리드 GA의 품질이 순수 GA보다 좋았다. 이로부터 동일한 수의 지역 최적화일 때는 하이브리드 GA가 더 좋은 품질을 낸다는 것을 확인할 수 있다. 표 2의 기본 실험 결과는 주어진 시간 내의 세대 수가 차이 나기 때문에 발생한 것(순수 GA의 경우 170초 동안 4876.13번 수행하였으며, 하이브리드 GA의 경우, 30.37번 수행하였다.)이며, 이는 지역 최적화 알고리즘의 수행으로 인한 딜레이에서 연유한다. 즉, 지역 최적화 알고리즘을 사용하면서 충분한 시간 혹은 동일한 세대 수만큼 수행되었다면 더 품질이 좋을 것으로 예상된다.

그렇다면 어떻게 하면 지역 최적화를 용이하게 활용할 수 있을 것인가? 우선, 주어진 인구(해집합)에 대해 바로 GA를 수행하기 보다는 우선적으로 지역 최적화를 사용하여 전체 품질을 높인 뒤,

하이브리드 GA를 활용한다면 유리할 것으로 사료 된다. 이 과정에서 고려해야 할 사항은 지역 최적화 알고리즘 수행에 따른 오버헤드이며, 당연히 지역 최적화 알고리즘의 수행 오버헤드(수행 시간)를 줄이고, 중복 계산을 줄인다면 큰 성능 개선이 있을 것으로 사료 된다.

4 결론 및 마무리

본 과제를 통해서 Maxcut 문제를 해결하기 위해 지역 최적화 알고리즘을 GA에 적용하였다. 이상의 실험을 통해서 지역 최적화와 관련하여 알게되는 사실을 다음과 같다.

1. 단순 지역 특성화의 특성 및 한계

   지역 최적화는 임의의 난수에서 시작하여 이를 통해 얻을 수 있는 최적의 형태를 얻기 때문에 시간이 흐를 수록 전체 인구의 품질을 높이지만, 임의의 해의 개수와는 무관하다. 또한 좋은 형질(스키마)를 보존하는 방식이 아니기 때문에 시간이 흐른다고 해서 최적해를 얻을 가능성이 올라가지는 않으며, 언제나 그 확률은 동일하다.

2. 지역 최적화의 GA로의 적용

   지역 최적화는 위에서 언급하였듯, 빠르게 품질을 올리지만, 어느 값 이상으로 계속 올리기에는 한계가 있다. 이는 좋은 형질(스키마)을 보존하지 않기 때문이다. 그러나 전반적인 품질을 빠르게 올리기 때문에, 인구가 정제되지 않은 상태에서 순수 GA를 적용하는 것보다 지역 최적화와 결합하여 GA를 적용하면 빠르게 좋은 품질을 얻을 수 있다.

3. 하이브리드 GA에서의 지역 최적화의 역할

   하이브리드 GA에서 지역 최적화는 기본적으로 전체 인구의 품질을 재빠르게 상승 시켜주는 역할을 수행한다. 그리고, 주어진 해와 순열을 통해 유사한 혹은 인접한 염색체에 대해 검사함으로써 좋은 품질의 해를 찾는 시간을 단축시킨다.

4. 주어진 시간에서 품질을 높이기 위한 고려 사항

   기본 실험에서 하이브리드 GA보다 순수 GA가 높게 나타난 까닭은 지역 최적화 알고리즘의 오버헤드에 의해 세대 수가 극히 작았기 때문으로 확인하였다. 결국 주어진 시간 내에 지역 최적화가 유리하게 작용하기 위해서는 가급적 지역 최적화 알고리즘의 수행 시간을 줄이고 중복 계산을 줄일 수 있어야 할 것이다. 한편으로는 지역 최적화의 수행 시간이 긴 경우 지역 최적화를 통해 전체 품질을 빠르게 올린 뒤, 상대적으로 수행 시간이 빠른 순수 GA를 사용하는 것도 한 선택 사항이 될 것이다.