

# 2022 GA Project 2. Max-Cut with Classical GA

22조 (서윤형, 서현규, 이재선)

## Abstract

본 보고서는 Max-Cut 문제를 순수 유전 알고리즘만을 이용해 최적의 값을 도출하고자 한 과정 및 결과에 대한 기술이다. 룰렛휠 방식 선택과 균등교차, 엘리티즘 대치를 Steady-State로 문제를 해결하였으며, 구현한 알고리즘으로 5개의 샘플 인스턴스에 대해서 각각 30번씩 실험을 수행한 결과 각 데이터셋의 최적해 값을 99, 358, 3287, 4713, 8972 까지 찾아낼 수 있었다. Introduction, Methodology, Result, Conclusion & Discussion 순으로 각각 문제에 대한 정의, GA구조, 실험 결과, 결론 및 새로운 현상에 대해 설명하고 있다.

## Introduction

Max-Cut 문제는 그래프 이론의 난제 중 하나로 polynomial time 안에 풀 수 없기 때문에 이를 해결하기 위해 여러가지 시도들이 있어왔고 그 중 하나가 유전알고리즘이다. Project 2에서는 지역최적화가 배제된 순수 유전알고리즘만을 사용하여 문제해결을 시도해야 하는데 최적의 해를 찾는 것이 쉽지 않기 때문에 이를 극복하기 위해 유전알고리즘의 다양한 연산자와 파라미터값들을 조정하는 방법을 사용하였다.

본 보고서에서는 이러한 방법에 대해 기술하고 샘플로 주어진 5개의 인스턴스에 대한 실험결과 및 세대 진행에 따른 해집합의 변화를 관찰한 결과를 포함하고 있다. 마지막으로 지역최적화가 배제된 상황에서 순수 유전알고리즘의 성능을 극대화하기 위해 어떻게 노력했는지 기술하였다.

## Methodology

프로젝트 수행을 위해 주어진 환경은 다음과 같다. 먼저 방향성이 없는 간선에 가중치가 부여된 단수 그래프  $G=(V, E)$ 가 주어진다.  $V$ 와  $E$ 는 각각 간선을 구성하는 정점(Vertex)과 간선(Edge)의 집합이다. 정점의 집합  $V$ 를 두개의 집합으로 분리하여 두 집합간에 존재하는 간선의 가중치합이 최대값이 되도록 하는 것이 목표이다. 제안되는 유전알고리즘의 성능을 확인하기 위해 5개의 샘플 인스턴스( $u50, u100, u500, w500, w297$ )를 각각 30회 이상 수행해야 하며 지역최적화 배제를 위해 선택 및 대치연산의 경우에만 해의 품질을 사용할 수 있으며 교차 및 변이의 경우 해의 품질을 이용할 수 없다. 프로그램 수행시간은 3분 이내이며 수행이 완료된 후 최종 해집합에서 가장 품질이 좋은 해가 프로그램 수행 결과로 도출된다.

### • GA 구조

본 프로젝트에서 사용된 유전알고리즘은 다음과 같은 구조로 되어 있다.

1. P개의 초기 해집합 생성
2. 제한시간(180초)동안 Steady-State GA 수행
  - a. 새로운 해 생성
    - i. P개의 해집합에서 2개의 부모해 선택(품질비례룰렛휠, 선택압 3.9)
    - ii. 선택된 2개의 부모해를 교차하여 새로운 해 생성(균등교차, 교차임계값 0.5)
    - iii. 생성된 새로운 해를 변이시킴 (1 bit flip)
  - b. 해집합에서 가장 품질이 나쁜 해를 새로운 해로 대치함 (엘리티즘 대치)
3. 해집합 중 최적해 도출

초기 해집합은 입력된 샘플 인스턴스와 같은 길이를 갖는 해를 랜덤함수를 사용하여 생성하였다. 해집합의 크기는 실험을 통해 적합한 값으로 설정하였는데, 해집합 크기가 작을 때는 해들의 표준편차가 큰 편이었고

큰 경우에는 표준편차가 작았기 때문에 프로그램 수행시간을 고려하여 해집합 크기를 선정하였다. 해집합의 크기는 해의 다양성 및 초기 탐색 범위와 연관이 있고 해의 수렴이나 정확도에 영향을 끼치기 때문에 주어진 상황에 따라 적당한 값을 선정하는 것이 필요하며 어느 상황이든 적용할 수 있는 최적의 해집합 크기를 찾는 것은 쉽지 않다[1][2].

- 해의 표현

본 프로젝트에서 해결해야 하는 문제는 방향성이 없는 간선에 가중치가 부여된 단순 그래프가 주어졌을 때 간선의 가중치합이 최대가 되도록 그래프를 분할하는 것이기 때문에 그래프를  $G = (V, E)$ 로 나타냈을 때 정점 집합  $V$ 가 해가 된다. 해( $V$ )를 분할한 두 개의 부분집합( $S_1, S_2$ )간 교집합은 존재하지 않으며 두 집합은 간선으로 연결되는 위치기반이다. 따라서 해는 이진값을 갖는 1차원 배열로 볼 수 있고 0을 갖는 정점 집합( $S_1$ )과 1을 갖는 정점 집합( $S_2$ )를 다음의 그림과 같이 표현할 수 있으며  $S_1$ 과  $S_2$ 는  $V$ 의 인덱스값을 갖게 하였는데. 문제에서 주어지는 입력파일이 간선으로 연결된 정점들의 인덱스값과 간선의 가중치값으로 구성되어 있기 때문에 정점들의 인덱스값을 알아야 가중치합을 구할 수 있기 때문이다.

	1	2	3	4	5	6	...	$n-2$	$n-1$	$r$
$V$	0	1	1	1	0	0	...	1	0	1
	1	2	3	...	$k$					
$S_1$	1	5	6	...	$n-1$					
	1	2	3	...	$n-k-1$	$n-k$				
$S_2$	2	3	4	...	$n-2$	$n$				

- 연산자

(선택연산) 선택연산은 교차 연산을 적용하기 위해 해집단에서 부모해를 선택하는 과정인데 전제되는 원칙은 우수한 유전형질이 선택되어 좋은 유전형질이 자식세대로 유전되어야 한다는 것이다. 우수한 정도를 판별하기 위해서 적합도를 계산할 필요가 있는데 해당 유전형질이 문제 해결에 얼마나 적합한지를 나타내는 값이다. 계산된 적합도를 기반으로 선택압을 조절하여 우수한 유전자와 우수하지 않은 유전자를 선택하는 비율을 조절한다. 본 프로젝트에서 사용된 선택연산자는 가장 널리 사용되는 품질비례 룰렛휠 방식을 사용하였으며 아래의 그림과 같은 수식을 사용하여 해의 적합도를 계산한다.

$$f_i = (R_i - R_w) + \frac{(R_b - R_w)}{(k - 1)}$$

$f_i$  : i 번째 해의 적합도

$R_i$  : i 번째 해의 Reward(= 가중치합)

$R_w$  : 해집합의 worst Reward(= lowest 가중치합)

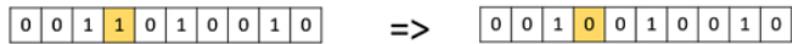
$R_b$  : 해집합의 best Reward(= highest 가중치합)

선택압은 결정할 때 고려할 사항이 있는데 적합도가 높은 해들을 우선할 경우 진화과정에서 설익은 해에 수렴할 가능성이 높기 때문에 적합도가 낮은 해들로 어느정도 선택함으로 유전적 다양성을 갖고 탐색 공간을 더 넓게 사용해 설익은 해에 수렴할 확률을 줄일 필요가 있다. 룰렛휠 방식의 경우 선택압은 일반적으로 3~4의 값을 선택하기 때문에 해당 범위내에서 가장 적합한 값을 실험적으로 결정하였다.

(교차연산) 교차연산은 문제 해결에 영향을 미치는 두 부모해의 유전적 특징을 조합하여 두 부모의 특징을 갖는 자식해를 만들어가는 과정인데 문제해결에 탁월한 유전자들이 모여있는 품질 좋은 부모해가 선택되었다고 가정했을 때 일정 교차의 경우 품질이 좋은 자식해가 생성될 가능성이 높으나 다점교차나 교란이 큰 다른 교차방식의 경우 공간 탐색을 보다 넓게 할 수 있다는 장점을 갖는다. 본 프로젝트에서는 순수

유전알고리즘을 활용해야 하기 때문에 가장 간단한 균등교차 방법을 사용하였다. 부모해의 크기만큼 난수를 발생시켜 임계확률 이상의 값을 갖는 경우에만 부(father)의 유전자 위치값을 취하고 아닌 경우에는 모(mother)의 유전자 값을 취하여 새로운 자식해를 생성하였다. 균등교차의 경우 교란의 정도가 큰 편이기 때문에 수렴시간이 오래 걸리는 단점이 있어 이에 대한 보완으로 대치연산에서 수렴이 빠른 Steady-State 대치연산을 사용하였다.

(변이연산) 변이는 부모해에서 찾을 수 없는 유전형질을 부여하기 위한 것으로 해집합의 다양성을 확보하고 최적해 탐색공간을 넓힐 수 있게 해준다. 본 프로젝트에서는 1-bit flip 변이연산을 사용하였는데, 아래의 그림과 같이 해의 인덱스 중 랜덤하게 하나를 선택하여 해당값의 이진값을 반전시켰다.



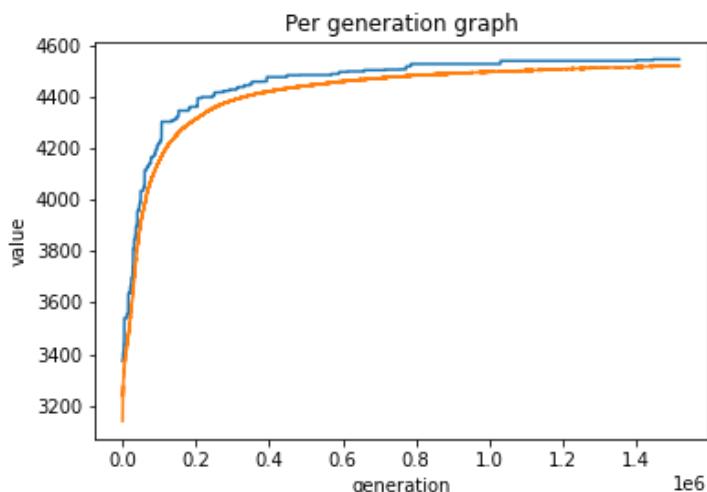
(대치연산) 대치는 기존의 해집합과 새로 생성된 자식해를 조합하여 새로운 해집합으로 재구성하는 과정이다. 본 프로젝트에서는 교란이 큰 균등교차방식을 사용하였기 때문에 이를 보상하기 위해 수렴이 빠른 Steady-State 방식을 사용하였는데 새로운 자식해가 생성될 때마다 기존의 해집합에서 가장 품질이 나쁜 해와 교체하는 방식이다.

## Results

구현한 GA에 대한 성능을 측정 및 분석하기 위하여 다음 두 가지 실험을 진행했다.

- 세대 진행에 따른 **population** 분석 (해들의 평균 품질, 최고 품질 등)

주어진 5개의 데이터셋 중 가장 크기가 크고, 그래프 구성이 복잡한 w500 데이터셋을 이용하여 GA를 수행하고, GA 내부 세대 변화에 따른 population을 분석했다. Population 분석 시 population 내부 최고 품질과 평균 품질을 측정했으며, 그 결과는 다음 그래프와 같다.



세대 변화에 따른 최고 품질, 평균 품질 그래프

주황색 그래프는 해들의 평균 품질을, 파란색 그래프는 해들의 최고 품질을 나타낸다. 위 그래프를 통하여 초기에 급격하게 해들의 품질이 증가하였지만, 곧 이내 최고 품질 그래프와 평균 품질 그래프가 거의 맞달게 되며 해의 품질이 수렴하는 것을 확인할 수 있었다.

- 샘플 인스턴스에 대한 실험결과

문제에서 주어진 5개의 샘플 인스턴스에 대해서 각각 30번씩 실험을 수행하였고 수행결과로 도출되는 최적해에 대해서 가중치합의 최고값, 평균값 그리고 표준편차를 계산하였으며 그 결과는 아래의 그림과 같다. 왼쪽은 trial & error 방식으로 파라미터를 맞춘 결과이고 오른쪽은 베이지안 최적화를 통해 파라미터를 맞춘 결과이다.

해집합크기 400, 선택압크기 3.9, 교차임계값 0.5						해집합크기 378, 선택압크기 3.4, 교차임계값 0.249					
최적값	99	358	3,314	4,743	9,340	최적값	99	358	3,314	4,743	9,340
데이터	u50	u100	u500	w500	w297	데이터	u50	u100	u500	w500	w297
1	99	355	3,265	4,665	8,972	1	99	352	3,242	4,699	8,840
2	99	358	3,248	4,608	8,972	2	99	349	3,246	4,606	8,080
3	99	354	3,287	4,627	8,428	3	98	348	3,270	4,578	8,492
4	98	358	3,251	4,651	8,764	4	99	349	3,235	4,645	8,588
5	99	354	3,245	4,568	8,528	5	99	358	3,279	4,639	8,912
6	99	349	3,262	4,639	8,492	6	99	358	3,241	4,615	7,868
7	99	358	3,241	4,575	8,492	7	99	357	3,230	4,609	8,212
8	99	353	3,218	4,611	8,420	8	99	357	3,240	4,609	8,372
9	96	355	3,244	4,578	8,420	9	99	358	3,251	4,633	8,532
10	99	358	3,230	4,671	8,584	10	99	352	3,259	4,634	8,428
11	99	358	3,237	4,633	8,472	11	99	358	3,266	4,646	8,632
12	99	346	3,252	4,621	8,792	12	99	353	3,255	4,569	8,476
13	99	345	3,253	4,656	8,320	13	99	351	3,279	4,615	8,476
14	99	358	3,255	4,631	8,232	14	99	353	3,220	4,628	8,620
15	98	358	3,273	4,617	8,704	15	99	344	3,266	4,575	8,628
16	99	347	3,251	4,620	8,396	16	99	347	3,279	4,590	8,368
17	99	352	3,231	4,596	8,424	17	99	355	3,271	4,635	8,572
18	99	347	3,230	4,673	8,044	18	99	358	3,261	4,645	8,088
19	99	352	3,268	4,581	8,384	19	98	357	3,228	4,644	8,584
20	99	355	3,276	4,591	8,620	20	96	353	3,249	4,628	7,988
21	99	354	3,230	4,637	8,672	21	99	358	3,269	4,633	8,184
22	96	349	3,264	4,626	8,524	22	99	358	3,260	4,602	8,348
23	99	353	3,257	4,713	8,632	23	99	352	3,240	4,656	8,480
24	98	358	3,249	4,667	8,668	24	99	354	3,267	4,611	8,240
25	99	348	3,260	4,680	8,368	25	99	358	3,278	4,607	8,444
26	99	353	3,258	4,657	8,324	26	98	348	3,240	4,639	8,440
27	99	354	3,244	4,594	8,876	27	99	349	3,218	4,603	8,924
28	99	354	3,253	4,622	8,396	28	98	347	3,269	4,619	8,828
29	99	352	3,272	4,590	8,680	29	96	349	3,272	4,634	8,428
30	99	356	3,261	4,656	8,364	30	99	353	3,222	4,648	8,512
최고값	99	358	3,287	4,713	8,972	최고값	99	358	3,279	4,699	8,924
평균값	98.70	353.37	3,252.17	4,628.47	8,532.13	평균값	98.67	353.10	3,253.40	4,623.13	8,452.80
표준편차	0.79	4.01	15.68	35.69	213.49	표준편차	0.80	4.26	18.95	26.89	256.57

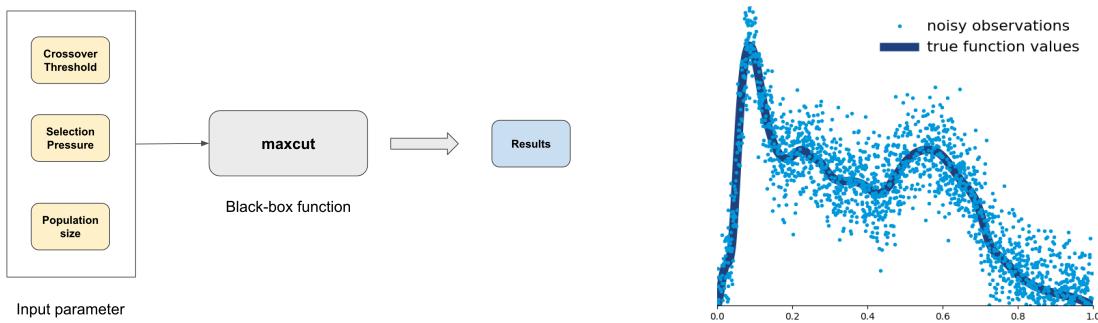
두 결과가 거의 유사하지만, trial & error 방식은 최적화된 파라미터를 찾는데 오랜 시간과 실험자의 많은 노력이 들은 반면, 베이지안 방식은 파라미터 값을 찾는데 비교적 빠른 시간과 매우 적은 노력이 들었던 장점이 있었다.

## Conclusion & Discussion

- Bayesian optimization

이번 과제를 수행하면서 연산자를 어떻게 구현할지 (roulette wheel selection vs tournament selection) 도 어려웠지만 가장 어려웠던 부분은 각 연산자의 파라미터 (선택압 등)을 어떻게 최적화시킬지에 대한 부분이었다. 구현한 GA 에는 교차 임계치, 선택압, population 크기 등의 파라미터가 존재했는데, 각 파라미터 별로 10가지 경우의 수가 있다고 가정한다면 총 경우의 수는 1000 가지로써 모든 경우를 탐색하기란 매우 어려운 문제였다.

GA 를 구현한 이유는 *maxcut* 이라고 하는 일종의 black box function 의 output 을 최대로 하기 위함인데, 그렇다면 black box function 의 parameter (crossover threshold, selection pressure, population size) 의 최적화된 값을 찾을 수 있는 bayesian optimization 을 사용한다면 최적화된 파라미터를 찾는데 도움이 될 것 같다는 생각이 들었다.



### *Bayesian optimization* 도입 배경

이러한 발상에 착안하여 *Bayesian optimization* 을 maxcut 에 도입하기 위해, *maxcut C* 프로그램에 대한 *bayesian optimization* 을 수행하는 스크립트를 실행시켜 최적화된 GA 파라미터 들을 얻을 수 있었다.

- **Population Init**

해집합을 랜덤하게 초기화하는 데에 **iteration**을 줄이기 위해 또 다른 방식으로 구현을 하여 실험을 진행하였다. 5개의 데이터 셋 중 특히 키메라 데이터셋에서 두드러지는 효과를 보여줬는데, 초기해 집합내 0으로만 혹은 1이 적게 섞여 이루어진 스키마가 존재할 확률이 좀 더 높기 때문에 키메라 데이터셋에 최적화된 해를 찾아내기가 더 수월했던 것으로 판단된다. 최종 코드에는 반영하지 않았지만 키메라 데이터셋에서 30번의 시도에 최소 8864 점수를 획득하였다. 초기해 집합 설정방식 또한 문제를 푸는데에 중요한 요소 중 하나라고 사료된다.

- **Conversion**

유전 알고리즘에서 주어진 180초를 모두 활용하는 것이 최적의 해를 찾기 위한 조건이라고 가정하고, 수렴을 늦추기 위한 방법으로 토너먼트 셀렉션, **generation** 교체 방식에 일정세대 이후 돌연변이 비율의 변화와 세대 교체 비율에 대한 파라미터 조정을 시도했다. 큰 성과가 나오지 않았는데, 다음 과제에 지역최적화와 함께 적용시켜 보면 해가 더 좋아질 것이라 예상한다.

- **Conclusion**

여러가지 방식의 유전 알고리즘을 사용하고 파라미터를 조정시키면서 모든 상황에 가장 좋은 각각의 대치, 선택 연산 알고리즘이 있는 것이 아니라, 특정 선택 연산을 선택하면 그것에 상호적으로 다른 알고리즘을 선택하여야 좋은 결과를 낼 수 있었다. 마찬가지로 파라미터 또한 최적의 선택압이 존재하는 것이 아니고 해집합 크기, 교차 임계값 등이 연관되어 작용하였다.

이번 과제를 통하여 순수 **GA**만으로는 중간에 수렴이 되는 경우 가장 좋은 해를 얻기가 힘들다는 것을 깨달을 수 있었다. 지역최적화를 통해 다음 과제는 최적의 해를 도출할 수 있도록 기대해본다.

#### 팁 기여도

- 이재선 ( $\frac{1}{3}$ ), 서윤형 ( $\frac{1}{3}$ ), 서현규 ( $\frac{1}{3}$ )

#### References

- [1] Gotshall, S. and Rylander, B., "Optimal Population Size and the Genetic Algorithm", Proc On Genetic And Evolutionary Computation Conference, 2000.
- [2] Luke, S., Balan, G.C. and Panait, L, "Population Implosion In Genetic Programming", Department of Computer Science, George Mason University. <http://www.cs.gmu.edu/~eclab>