

MATLAB으로 배워 보는 유전 알고리즘 (1)

김재은

(대구가톨릭대학교 기계자동차공학부)

1. 머리말

사람마다 서로 다른 의견이 있겠지만, 필자는 공학 문제의 최적화 방법을 크게 거시적 방법과 미시적 방법으로 구분하고 싶다. 거시적 방법에는 DOE(design of experiment)라 불리는 통계적/Taguchi식 실험 계획법 및 이에 따른 반응 표면법(RSM: response surface method) 등이 있다. 이는 최적화하고자 하는 시스템에 대한 정확한 수학적 모델이 없어도 해석 또는 실험의 경험적 방법에 의해 근사적인 최적화가 가능한 장점이 있다. 그러나, 대상 시스템에 대한 전문적인 지식이 필요 없는 black-box 접근 방법이므로 변수들간의 인과 관계 또는 관련 법칙의 발견 등이 곤란하다. 이에 반해 미시적 방법은 시스템 분석/해석 또는 민감도 분석을 위해 변수들간의 인과 관계를 표현하는 수학적 모델이 필요하고 따라서 최적화 대상 시스템에 대해 잘 알고 있어야 한다. 미시적 방법은 민감도 기반 기법과 확률적 탐색 기법 등으로 구분될 수 있으며 현재 다양한 기법 등이 개발되어 있다.

필자의 개인적인 경험으로는 여러 최적화 기법 중에서도 민감도 기반의 미시적 방법이 가장 효

율적이라고 생각한다. 그러나, 목적함수 및 제한 조건의 설계 변수에 대한 민감도 계산이라는 귀찮지만 매우 중요한 절차가 필요하다. 민감도 정보는 해석적 또는 유한차분법(FDM: finite difference method)을 통해 얻는 것이 가능하데, 해석적 방법은 경우에 따라 구할 수 없는 경우가 빈번하고 유한차분법은 설계 변수의 개수가 증가함에 따라 계산 시간이 증가하고 또한 선택된 설계 변수의 증분량에 따라 민감도 계산 결과가 부정확할 수 있다. 어떤 방법을 사용하든 민감도 기반의 최적화 방법은 민감도 계산상의 어려움뿐만 아니라 또한, 문제의 비선형성에 따라 전역 탐색 공간에서의 최적화(global optimum)가 아닌 국소 최적값(local optimum)을 계산할 가능성이 크다. 이에 비해 확률적 탐색의 대표적인 기법인 유전자 알고리즘(GA: genetic algorithm)은 최적화 결과를 얻기 위해 기다려야 하는 긴 시간을 제외한다면 이러한 문제들에서 어느 정도 벗어날 수 있게 해준다. 최적화 전공자 중에는 민감도 계산이라는 골칫거리에서 벗어날 수 있다는 이유만으로 유전자 알고리즘을 선호하기도 한다. 그러나, 보다 본질적으로 유전 알고리즘이 민감도 계산을 필요로 하지 않는다는 것은 목적 함수가 설

계 변수에 대해 불연속 또는 미분 불가능하거나 비선형성이 심한 경우에도 적용 가능하다는 것을 의미한다. 따라서, 유전 알고리즘은 설계 변수가 실수와 정수 값을 모두 갖는 mixed integer 문제에도 적용될 수 있다.

이번 강좌에서는 MATLAB의 Toolbox로 제공되는 유전 알고리즘의 기초 용어 및 기본 연산에 대한 다양한 선택 사항의 의미 등을 살펴보고, 간단한 예제에의 적용을 통해 MATLAB의 Toolbox 유전 알고리즘의 올바른 사용법을 설명하고자 한다.

2. 유전 알고리즘의 기초

Darwin의 진화론에 의하면 자연의 각 개체들은 환경 적응의 유리한 정도에 따라 선택적으로 번성하게 된다. 이 과정에서 DNA 유전자(gene)로 구성된 염색체(chromosome) 들은 상호 교차(crossover)에 의한 부분 결합 및 돌연변이(mutation)를 통해 새로운 염색체로 변화된다. 이러한 생물 진화의 과정을 모방하여 문제 해결 방법으로 개발된 것이 바로 유전 알고리즘이다. 유전 알고리즘은 1960년대에 처음 제안되었으나, 교차 및 변이 연산자를 포함한 이론적 바탕은 1975년 John Holland의 저서 'Adaptation in Natural and Artificial Systems' 에서 마련되었으며, 그의 제자 David Goldberg에 의해 더욱 발전되었다.

2.1 용어의 정의 및 기본 연산

과학이나 공학에 있어서 다루고자 하는 분야의 용어에 대한 정의가 제대로 되어 있어야 본질 규명이 자연스럽다. 여기서는 MATLAB Toolbox에서 제공되는 유전 알고리즘을 기준으로 관련 용어를 설명하였다.

- 적합도 함수(fitness function): 진화론 자연 도

태설의 환경에 대한 적응 능력을 의미하는 것으로서 공학적으로는 최적화(최소화) 대상의 목적 함수를 의미한다. 원래의 의미에 맞게 MATLAB 내부에서는 최대화 함수로 변환된다.

- 개체(individual): 적합도가 평가되는 설계 변수의 집합으로서 염색체, 게놈(genome), D-string, genetic string 등으로도 불린다.

- 유전자(gene): 개체 (염색체)를 이루는 요소로서 설계 변수의 개념과 같다. V-string이라고도 한다.

- 집단(population): 개체가 모인 집합으로서 집단의 크기는 일반적으로 설계 변수의 개수 이상이어야 한다. 적합도가 높은 개체가 다음 세대에서 살아남을 확률이 크다.

- 세대(generation): 집단을 구성하는 개체들은 매번 적합도 평가를 통해 적합도가 높은 개체들로 구성된 집단으로 계속 진화한다. 일련의 연속적인 집단을 세대라고 하며 최적화의 iteration에 해당한다.

유전 알고리즘은 현재의 집단에서 다음 세대의 집단을 생성하기 위해 다음과 같은 3단계의 주요 연산 및 범칙을 사용한다.

- 선택(selection): 다음 세대의 자식 개체들(children)을 생성하기 위한 부모 세대(parents)를 선택함에 있어서 현재 집단에 대해 일정한 적합도 평가 방법을 이용한다.

- 교배 또는 교차(crossover): 선택된 부모 개체들을 일정한 교차 범칙에 의해 결합하여 부모의 일부 유전자를 갖는 새로운 개체들(crossover children)을 생성한다.

- 변이(mutation): 집단의 다양성을 갖도록 어떤 개체들에 대해서는 임의의 변화를 가하여 다음 세대의 자식 개체들(mutation children)을 생성한다.

위의 연산들을 이용한 MATLAB Toolbox 유전

기초강좌

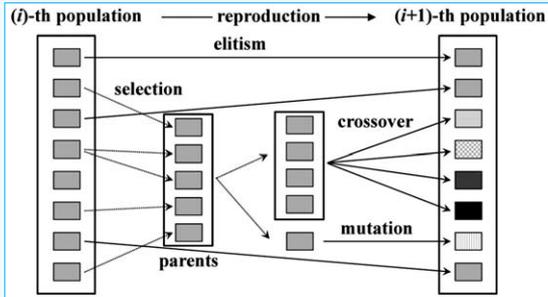


그림 1 MATLAB Toolbox로서 제공되는 유전 알고리즘 진행의 개략도

알고리즘의 개략도를 그림 1에 나타내었다. 그림 1에 나타낸 바와 같이 전 세대로부터 다음 세대의 집단을 생성(재생산: reproduction) 하기 위해 기본적으로 선택, 교배 및 변이 연산을 수행하게 되는데 이 외에도 유전 알고리즘의 성능을 개선하기 위해 엘리트 전략(elitism)을 사용하기도 한다. 즉, 전 세대에서 발견된 최적의 개체가 3가지 연산을 수행하면서 다음 세대에서 소멸할 수 있는데 이를 방지하기 위해 변경되지 않고 다음 세대로 그대로 생존시킬 개체의 비율을 미리 정할 수 있다. MATLAB Toolbox 유전 알고리즘에서는 이를 위해 엘리트 개체 수 (elite count: EliteCount.m)와 교배 개체 수 비율(crossover fraction)을 설정할 수 있다. 엘리트 개체 수는 당연히 집단의 크기와 같거나 작아야 하며 비정수 문제(non-integer problem)에서는 기본값이 2로 설정되어 있다. 정수 문제에서는 $0.05 \times \min(\max(10 \times \text{설계 변수의 개수}, 40), 100)$ 로 계산되어진다. 교배 개체 수 비율은 교배 대상이 되는 부모 개체 수의 비율로 0과 1사이의 값을 가지며 기본값은 0.8로 되어있다. 여기서 주의할 점은 교배 개체 비율의 계산 대상 개체 수는 집단의 크기에서 엘리트 개체 수를 제외한 수이다. 즉, 예를 들면 집단의 크기가 20으로 설정되어 있는 경우, 엘리트 개체 수가 2, 교배 개체 비율이 0.8로 설정되

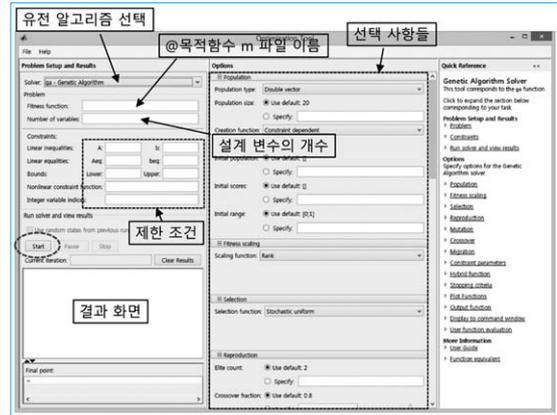


그림 2 MATLAB 유전 알고리즘 Toolbox의 사용자 인터페이스 구성

어 있다면 전 세대로부터 최적의 개체 2개가 그대로 생존하여 다음 세대의 구성원이 되며, $20 - 2 = 18$ 개 개체를 선택 연산에 의해 선택 후 이 중 $18 \times 0.8 \approx 14$ 개의 개체는 교배 연산에 의해, 나머지 4개의 개체는 변이 연산에 의해 다음 세대를 구성하게 된다.

2.2 MATLAB 유전 알고리즘 Toolbox의 구성

MATLAB 유전 알고리즘 Toolbox를 본격적으로 설명하고 사용하기에 앞서서 사용자 인터페이스 구성을 살펴보고자 한다. 이번 강좌에서는 독자가 MATLAB에 대해 어느 정도 익숙하다고 가정하고 유전 알고리즘의 사용법에 대해 설명할 것이다. MATLAB에서 유전 알고리즘 Toolbox를 이용하기 위해서는 명령어 창 및 m 파일에서 `ga(@fitnessfun, nvars, options)`의 형태로서 GA함수를 직접 이용해도 되지만, 여기서는 GUI를 기본으로 설명할 것이다. 명령어 창에서 `optimtool('ga')`를 입력하거나 간단히 `optimtool`을 입력하면 (이 경우 유전 알고리즘을 사용하기 위해 solver를 `ga - genetic algorithm`으로 선택) 그림 2와 같은 창을 볼 수 있다.

2.3 Rastrigin함수의 최소화 문제 적용 예

여기서는 간단한 문제에 대해 MATLAB 유전 알고리즘을 적용해 보자. 사용자 매뉴얼에서 설명하고 있는 Rastrigin함수 최소화를 구하는 문제인데 이 함수는 새롭게 제안/개선된 유전 알고리즘의 성능을 평가할 때 자주 사용되는 함수로서 그림 3에서 보는 바와 같이 매우 많은 국소 최소점을 갖고 있으며, 아래와 같이 정의 된다.

$$f(x, y) = 20 + x^2 + y^2 - 10(\cos 2\pi x + \cos 2\pi y) \quad (1)$$

MATLAB에서 기본적으로 제공하는 rastriginsfcn.m 파일은 식 (1)의 경우 2개 변수를

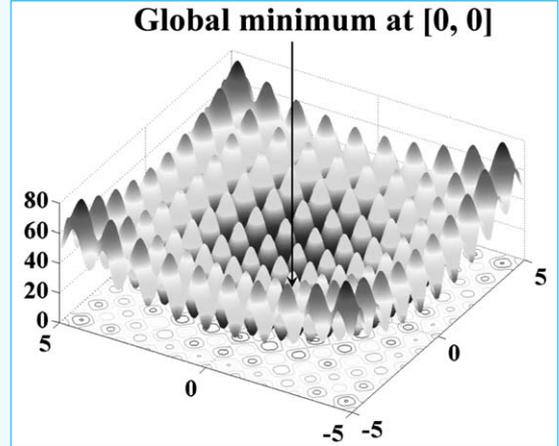


그림 3 Rastrigin 함수와 전역 최적점 $(x, y) = (0, 0)$

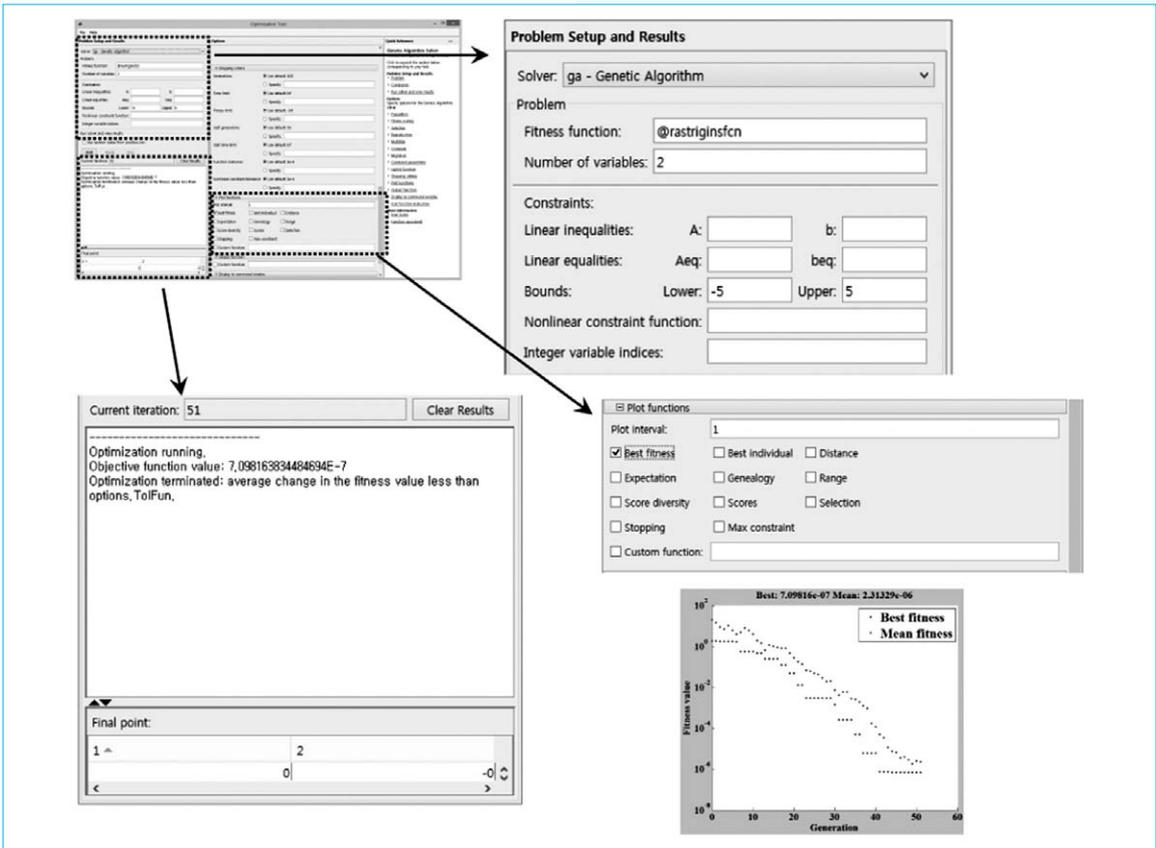


그림 4 유전 알고리즘을 이용한 Rastrigin 함수의 최소화 문제

기초강좌

입력으로 하여 함수 값을 출력하도록 작성되어 있다. 또한, 위 함수의 최소값을 여기에서는 $-5 \leq x \leq 5$ 및 $-5 \leq y \leq 5$ 영역에서 구할 것이며 설계 변수가 2개 이므로 그림 4와 같이 입력한다. 선택 사항의 'Plot functions'의 'Best fitness'에 표시를 하고 'Start'를 누르면 좌측 하단에 현재 iteration가 표시되며 결과 창에는 최종 목적 함수 값과 그에 대응하는 설계 변수의 값이 나타난다. 나머지 선택 사항은 기본으로 하였다. 집단의 크기(population size)는 20으로 하였다.

그림 4에 나타난 결과를 통해 51번의 iteration (generation)만에 $(x,y) = (0,0)$ 에서 최소값 약 $7.09E-7$ 이 됨을 알 수 있다. 유전 알고리즘은 확률론적 탐색 기법이므로 매번 최적화 수행 시 마다 서로 다른 결과를 가져올 수 있음을 유의하여야 한다. 또한, 기존의 민감도 기반 최적화 방법에서는 각 iteration당 계산된 민감도 및 이를 이용한 특정 알고리즘에 의해 단 1개의 설계점이 최적점을 찾아가는 것과는 달리 유전 알고리즘에서는 각 세대에서 난수 발생기(random number generator)를 기반으로 선택된 다수의 설계점이 존재하고 이 중에서 최적의 설계값을 갖는 설계점이 최적해에 근접하게 된다. 이는 그림 4의 iteration 대비 Best/Mean fitness 결과 그림을 통해 확인할 수 있다. 각 세대의 평균 적합도 및 세대 중의 가장 적은 적합도 값을 통해 최적화 과정이 수렴됨을 알 수 있다.

3. 유전 알고리즘의 선택 사항 설명

유전 알고리즘을 이용하여 최적의 해를 얻기 위해서는 MATLAB 유전 알고리즘 Toolbox에서 제공되는 여러 가지 선택 사항을 매번 다르게 선택하여 여러 번 수행해 보는 것이 바람직하다. 따라서, 선택 사항의 의미를 정확히 알고 있어야

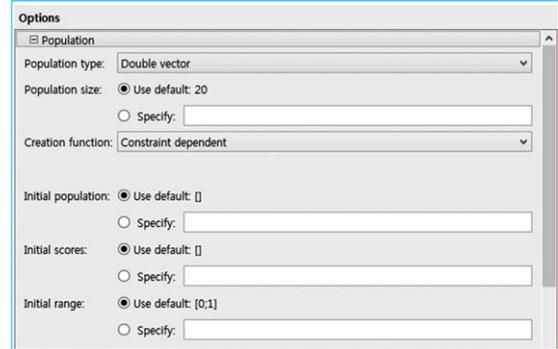


그림 5 초기 집단의 생성에 관한 선택 사항

주어진 문제의 최적화를 효율적으로 수행할 수 있을 것이다. 앞에서는 비교적 간단한 Rastrigin 함수의 최소화 문제에 대해 모든 선택 사항을 주어진 대로 두고 유전 알고리즘을 적용하였지만, 문제에 따라서는 다른 선택 사항을 고려해야만 원하는 최적화 결과를 얻는 경우가 더욱 많을 것이다.

3.1 초기 집단의 생성('Population')

유전 알고리즘을 적용하기 위한 첫 단계로서 초기 집단의 생성에 관한 선택 사항은 그림 5에 나타내었다.

- 설계 변수의 형태(population type)

'Double vector', 'Bit string', 'Custom' 등을 선택할 수 있으나 대부분의 공학 문제는 설계 변수가 실수(일부 정수 포함 가능)이므로 'Double vector'를 선택한다. 이번 강좌에서는 설계 변수의 형태가 'Double vector'인 경우에 한정하여 설명하기로 한다.

- 집단의 크기(population size)

한 세대의 집단을 이루는 개체의 개수를 의미하는 것으로서 앞서 설명한 바와 같이 적어도 설계 변수의 개수보다는 커야 한다. 기본적으로 20이 설정되어 있으며 설계 변수가 많지 않으면 그

대로 두어도 된다. 크기가 커지면 보다 넓은 설계 공간을 탐색할 수 있다는 장점이 있지만 계산 시간 및 수렴 속도를 고려하여 설정한다.

· 초기 집단의 생성 함수(creation function)

설계 변수의 형태가 'Double vector' 인 경우 'Constraint dependent' 가 기본으로 되어 있다. 그대로 두고 해도 되나 다른 선택 사항에 대해서는 다음과 같다.

- 균일 분포 생성 ('Uniform' : gacreation-uniform.m): 균일 확률 분포를 이용하여 초기 집단을 생성한다. 선형 부등식, 등식 제한 조건이 없거나, 설계 변수 제한 조건만 있는 경우에 기본적으로 적용된다.

- 유용 집단 생성 ('Feasible population' : gacreationlinearfeasible.m): 선형 부등식, 등식 제한 조건이나 설계 변수 제한 조건이 있는 경우의 초기 집단 생성 방법으로서 집단의 개체들이 선형 제한 조건의 경계에 있으면서도 적절히 분포된 초기 집단을 생성한다. 선형 제한 조건이 있는 경우에 기본적으로 적용된다.

· 'Initial population' 및 'Initial scores' 는 그대로 두어도 된다.

· 초기 집단의 분포 구간(initial range)

기본적으로는 [0;1]로 되어 있으며 이는 초기 집단이 분포되는 구간을 의미한다. 만약, 대략의 최소값의 위치를 안다면 초기 집단의 분포 구간이 이를 포함하도록 설정하면 좋을 것이다. 이는 초기 집단의 생성 분포 구간을 의미할 뿐 이후의

세대에서는 설계 제한 조건의 'Bounds' 설정에 따라 이를 포함하지 않을 수 있다. 그러나, 변이 연산 및 적합도 조정 등과 함께 집단의 다양성을 확보하여 다양한 설계 공간을 탐색할 수 있도록 하는 수단이므로 너무 좁지 않게 설정하는 것이 좋다.

4. 맺음말

이번 강좌에서는 MATLAB Toolbox에서 제공하는 유전 알고리즘을 효과적으로 잘 이용하기 위해 유전 알고리즘 관련 기초 용어 및 Toolbox구성을 설명하였다. 또한, 초기 집단의 생성에 관한 선택 사항에 대해서도 설명하였다. 다만, 지면이 부족한 관계로 유전 알고리즘의 핵심이라 할 수 있는 적합도 조정, 3가지 기본 연산 및 이들의 다양한 선택 사항에 대해서는 다음 호에서 기술하고자 한다. 간단한 구조 최적화 설계 문제에 대해서 MATLAB Toolbox 유전 알고리즘을 적용해 보는 기회도 가질 것이다. [KSNVE](#)

참고문헌

- (1) MathWorks, Global Optimization Toolbox-User's Guide, R2012a.
- (2) 진강규, 2004, 유전알고리즘과 그 응용, 교우사.
- (3) 문병로, 2001, 유전알고리즘, 다성출판사.