진화 프로그래밍을 이용한 안정지수 결정

Finding Stability Indices Using Evolutionary Programming

신진욱, 김인택, 강환일
명지대학교 전기정보제어공학부
0335) 330-6475
Email: jinugi@wh.myongji.ac.kr

요 약

진화 프로그래밍은 유전자 알고리즘과 함께 진화 연산 분야에 속하며 넓은 탐색공간에 존재하는 해를 찾는데 유용한 방법으로 알려져 있다. 본 논문에서는 이 두 가지 방법을 비교하기 위해서 Manabe 표준형을 기준으로 사용자의 요구사항에 맞는 스텝응답을 만족하는 계수, 즉 안정지수를 이 두 가지 방법을 적용하였다.

ABSTRACT

Both evolutionary programming (EP) and genetic algorithm (GA) are classified as evolutionary computation. They have been successful in finding a solution in a wide search space. In this paper, our objectives are to find the coefficients of characteristic equation, in terms of the stability indices using EP and GA, and to make a comparison of both methods.

I. 서론

이미 잘 알려진 바와 같이 최적화 문제를 해결하기 위한 하나의 방법으로 진화 연산 (Evolutionary Computation)을 들 수 있다. 진화연산 (Evolutionary Computation) 에는 진화 전략 (Evolution Strategy), 유전자 알고리즘 (Genetic algorithm), 그리고 진화 프로그래밍 (Evolutionary Programming) 등을 포함한다 [1]. 이들의 공통적인 동작원리는 진화되는 개체집단으로부터 더 나은 해가 발생하여 궁극적으로는 주어진 문제에 적합한 해를 찾고자 한다. 본 논문에서는 이들 중 유전자 알고리즘과 진화 프로그래밍을 특정 문제에 적용하여 그 차이를 관찰하고자 한다.

두 방법의 비교를 위해, 적용대상이 되는 문제는 Manabe 표준형의 계수, 즉 안정지수를 구하는 것이다[2]. 시스템을 분석하거나 설계할 때 스텝 응답의 상승시간 (rise time)과 정정시간 (settling time)은 매우 중요한 사양이다. 일반적인 시스템에서는 적은 값의 상승시간과 정정시간이 요구된다.

기존의 Manabe 표준형은 일반화된 스텝 응답을 제안했다. 즉 $\tau=a1/a0$ 가 고정되어 있을 때 안정지수γ_i (i=1,2,3,$\cdots$,n-1)를 특정한 값 ($\gamma_1=2.5$, $\gamma_2=2$, $\gamma_3=2$,.. $\gamma_{n-1}=2$)으로 선택하여 안정된 시스템을 설계 한다는 것이다. 그러

나 본 논문에서는 사용자의 요구에 맞는 설계 제약이 주어진 경우 Manabe 표준형에서 제한한 상승시간과 정정시간보다 더 빠른 계수를 찾아야 할 필요성이 발생한다.

논문[3]에서는 이와 같은 작업을 위하여 유전자 알고리즘을 사용하여 상승시간과 정정시간을 만족하는 안정지수 γ를 구하였다.

II. 진화 프로그래밍의 적용

1. 유전자 알고리즘과 진화프로그래밍의 차이점

유전자 알고리즘과 진화 프로그래밍은 최적화문제를 해결하기 위한 진화 연산의 대표적인 알고리즘으로 알려져 왔다.

두 알고리즘의 공통점은 개체집단이 세대를 거듭할수록 더 우량한 개체가 출현한다는 적자 생존의 가정에 근거를 준 연산 방법이다. 그러나 구체적인 방법을 살펴보면 아래와 같은 몇 가지의 차이점을 발견할 수 있다.

첫째로 개체를 표현하는 방법이 다르다. 진화프로그램은 해(Solution)를 표현하는데 벡터들을 사용하지만 고전적인 유전자 알고리즘은 이진 벡터를 사용한다. 이진벡터를 사용하여 개체를 표현 할 경우 개체를 이진 벡터로 변환하는 단계가 더 추가되기 때문에 계산 속도가 더 둔화 될 수 있다.

둘째로 선택과정에 있다. 진화 프로그래밍은 단일 세대에서, μ 개의 부모 세대들을 돌연변이를 (또는 재조합을 함께) 사용하여 λ 개의 자손 세대들을 생성한다. λ 개의 자손 세대들과 μ 개의 부모 세대들로 임시 개체집단을 생성하고 개체집단으로부터 가장 높은 적합도(fitness)를 갖는 개체를 선택한다. 반면에 유전자알고리즘은 단일 세대에서 Population Size 크기의 개체집단으로부터 적합도가 좋은 개체들을 선택한다. 각 세대에서 적합도가 좋은 개체들을 선택하여 비교 후 마지막으로 가장 좋은 적합도를 가진 개체들을 선택한다.

2. 진화프로그래밍의 연산자

진화 프로그래밍은 하나의 개체집단에서 돌연변이 연산자를 사용하여 새로운 자손세대를 생성한다. 생성된 자손세대와 부모세대의 적합도를 비교하여 적합도가 좋은 세대를 선택하여 새로운 개체집단을 생성하는 것이다. 이를 위해 주로 돌연변이 연산자를 사용하며 혹은 재조합 연산자를 사용하는 경우도 있다. 이들 연산자는 아래와 같다.

- 돌연변이 연산자 (*M*)

(*M* : Variable→Mutated Variable)

돌연변이 된 변수 (Mutated Variable) = 변수 (Variable) ± N(0, σ) 분포를 가진 랜덤 변수

- 재조합 연산자 (*R*)

(*R* : (parent1, parent2) → offspring)

offspring = parent 1 + α (parent 2 - parent 1)

3. 문제의 정의

n 차 시스템의 특성 방정식은 다음과 같다.

$$f(s) = a_n s^n + a_{n-1} s^{n-1} + \Lambda + a_1 s^1 + a_0$$

위의 식에서 안정지수의 정의는 아래와 같다.

$$r_i = \frac{a_i^2}{a_{i-1} \times a_{i+1}} \qquad (i = 1,2,3,\Lambda\ n-1)$$

또한 역으로 γ_i 으로부터 특성방정식의 계수 a_i 는 아래의 식에 의해서 구할 수 있다.

(단 $\tau = a_1/a_0$ 이며 $i = 1,2,..n$ 이다.)

$$a_i = \frac{a_0 \tau^i}{\gamma_{i-1} \times \gamma^2{}_{i-2} \cdots \gamma_2{}^{i-2} \times \gamma_1{}^{i-1}}$$

4. 진화 프로그래밍의 적용

본 논문에서는 n=6 인 Manabe 표준형의 계수를 구하기 위해 아래와 같은 순서로 진화 프로그래밍을 적용하였다.

[단계 1] 초기화

n=6 일 때 5 개의 개체집단을 하나의 세대로 설정하고 각 개체집단은 $\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5$ 의 값을 가진 실수 벡터이다. 이 때 각각의 γ_i (i =1,2,3,4,5)는 구간 [1.4656, 10]의 값을 가진다.

[단계 2] 연산자 적용

개체집단으로부터 돌연변이 및 재조합 연산자를 적용한다.

[단계 3] Squared Error 에 의한 순위 매김

각 개체집단의 r1,r2,r3,r4,r5 으로부터 특성방정식을 구하고 스텝 응답 도형에서 상승시간(T_r)과 정정시간(T_s)을 구한다. 또 목표로 하는 상승시간(T_{ro})과 정정시간(T_{so}) 간의 차이의 자승에러 (Squared error)를 구한다.

Squared error= $(T_{ro} - T_r)^2+(T_{so} - T_s)^2$

이를 토대로 자승에러가 적은 순으로 각 개체를 나열한다.

[단계 4] 위의 자승에러 또는 세대수가 주어진 값이 만족하면 시행을 중단하고, 그렇지 않은면 단계 2 로 간다.

III. 시뮬레이션

Manabe 표준형의 상승시간과 정정시간보다 빠른 상승시간과 정정 시간을 정하여 계수를 구하였다. 두 알고리즘의 성능 비교를 위하여 유전자 알고리즘에서는 크기가 50 인 개체집단을 100 세대에 거쳐 관찰하였으며, 진화 프로그래밍은 5 개의 개체집단을 1000 개의 세대동안 살펴보았다.

목표는 아래와 같이 설정하였다.

상승시간 : T_{ro} = 0.9180

정정시간 : T_{so} =2.0300

1. 유전자 알고리즘의 실험 결과

유전자 알고리즘은 적용한 결과는 아래 표 1 과 같다.

표 1. 유전자 알고리즘의 적용 결과

	상승시간	정정시간	자승에러
1 차	0.9550	2.4010	0.1390
2 차	1.0340	2.9810	0.9179
3 차	0.9730	2.5170	0.2402
4 차	0.9770	2.5350	0.2585
5 차	0.9900	2.7290	0.4938

2. 진화 프로그래밍의 실험 결과

진화 프로그래밍의 적용을 위해서 표 2 와 같이 4 가지 방법으로 선택하였다.

표 2. 진화 프로그램의 적용 방법들

	Case 1	Case 2	Case3	Case 4
New_P1	P1	P1	P1	P1
New_P2	(P1⊥P2)′	P1′	P1′	P1′
New_P3	(P2⊥P3)′	P1 ″	P1″	P2 ″
New_P4	(P3⊥P4)′	P2 ′	P2′	(P2⊥P3)
New_P5	(P4⊥P5)′	P3 ′	P5′	(P3⊥P4)

(재조합: ⊥, 돌연변이: ′, 새로운 돌연변이: ″)

다음 쪽의 표 3 부터 표 6 까지는 위의 표 2 에 의한 방법을 적용한 결과이다. 표 1 과 표 3~6 에 나타난 결과를 살펴보면, 진화 프로그래밍을 사용한 방법이 유전자 알고리즘을 사용한 방법에 비해 우수하다는 점을 확인할 수 있다. 이를 통계적으로 입증하기 위하여 SAS 를 수행시켜 Scheffe 에 의한 다중 비교 (Multiple Comparison)를 수행한 결과 아래와 같은 2 가지의 결론을 얻었다[4].

결과 1: 유전자 알고리즘의 결과와 진화 프로그래밍의 결과는 차이가 있다.

결과 2: 진화 프로그래밍들 (case 1, case 2, case 3, case 4) 의 결과는 차이가 없다.

표 3. Case 1 에 의한 결과

	상승시간	정정시간	자승에러
1 차	1.0500	2.0820	0.0201
2 차	1.0440	2.0740	0.0178
3 차	1.0050	2.0530	0.0081
4 차	1.0650	2.0780	0.0239
5 차	1.0090	2.0090	0.0087

표 4. Case 2 에 의한 결과

	상승시간	정정시간	자승에러
1 차	1.0430	2.0850	0.0186
2 차	1.0030	2.0370	0.0073
3 차	0.9900	2.0050	0.0058
4 차	0.9450	2.0270	0.0007
5 차	1.0730	2.0600	0.0249

표 5. Case 3 에 의한 결과

	상승시간	정정시간	자승에러
1 차	0.9810	2.0420	0.0041
2 차	1.0690	2.0540	0.0234
3 차	0.9700	2.1030	0.0080
4 차	0.9800	2.0000	0.0047
5 차	1.0210	2.0200	0.0107

표 6. Case 4 에 의한 결과

	상승시간	정정시간	자승에러
1 차	1.0050	2.0740	0.0095
2 차	0.9720	2.0620	0.0039
3 차	0.9760	1.9920	0.0048
4 차	0.9780	2.0140	0.0039
5 차	0.9870	2.0230	0.0048

IV. 결과

유전자 알고리즘과 진화 프로그래밍을 사용하여 Manabe 표준형의 안정지수를 찾아보았다. 이를 위해 유전자 알고리즘과 진화 프로그래밍을 사용하였는데 이 두 가지 방법의 실험을 통해 진화 프로그래밍의 사용이 더 나은 결과를 가져왔다. 이러한 결론은 개체의 수와 세대의 수가 일정하다는 조건에서 구한 것이며, 특별히 이 논문에 사용된 문제에 한하여 얻은 것이다. 따라서 이 결론을 일반화하는 것은 무리일 수 있다. 그러나 유전자 알고리즘의 사용은 이진 벡터를 해로 사용하는 관계로 소요 시간 측면으로도 진화 프로그래밍과 비교할 때 매우 비효율적으로 나타났다.

참고 문헌

[1] J. Zurada, R. Marks II, and C. Robinson, *Computational Intelligence: Imitating Life,* IEEE Press, 1994.

[2] S.Manabe. "Coefficient Diagram method,"*14th IFAC Symposium on Automatic Control in Aerospace,*pp.199-210,August,1998.

[3] Hwan Il Kang. "A Study on the Generalization of the Manabe Standard Forms with the Genetic Algorithm," *「한국자동제어학회」*, E-116,1999.

[4] 김동재 교수(카톨릭 대학교)가 수행.

본 논문은 한국과학재단 1999-2-302-013-3 의 지원을 받았음.