

정보기술응용연구  
제 3 권 제 3 호  
2001년 9월

## 유전적 프로그래밍을 이용한 응답면의 모델링 II: 최적의 다항식 생성

이 옥\* , 김남준\*\*

### 요 약

.....

본 논문에서는 유전적 프로그래밍(Genetic Programming)을 이용하여 최적의 다항식을 생성하는 기법을 제시하고자 한다. 다항식은 비선형성이 큰 응답면을 모델링해야 하며, 이를 위하여 GP 트리 생성시 2-3 차 오더의 Taylor Series를 사용하는 방법을 시도하였다. 아울러 생성되는 다항식의 크기를 제어하기 위해서 GP 트리가 표현할 수 있는 다항식의 최대 차수를 제한함과 동시에 하나의 주 트리와 보 트리로 구성되는 GAGPT(Group of Additive Genetic Programming Trees) 사용을 모색하였다. 마지막으로 두 개의 응용 예를 통하여 본 방법의 유용성을 검증하였다.

.....

---

\*) 대전대학교 전기공학과  
\*\*) 대전대학교 전기공학과

## 1. 서론

응답면(Response Surface)의 모델링에 가장 일반적으로 사용되는 것이 다항식(Polynomial)이다. 다항식은 그 함수 형태가 매우 간단하며, 미분식도 매우 쉽게 구할 수 있다는 장점이 있어 응답면의 근사화 뿐만 아니라 실시간 응용 등 여러 공학분야에 널리 사용된다. 차수(Order)가 낮은 2-3차 정도의 다항식을 사용하는 것이 가장 일반적인데, 이런 경우 요구되는 학습 데이터의 양은 작으나 비선형성을 표현하기가 어렵다는 문제점이 있다. 높은 차수의 다항식을 사용하면 다항식이 바람직하지 않은 진동 현상을 보일 가능성이 있고, 더욱이 다항식의 차수나 설계변수의 개수가 증가하면 다항식 항의 개수가 급격히 많아지게 된다. 이것은 바로 필요한 학습 데이터 양이 급격한 증가해야 한다는 것을 의미한다.

이와 같은 문제점을 해결하기 위해 다음과 같은 방안을 생각해 볼 수 있는데, 높은 차수의 다항식을 사용하되 기여도가 크지 않은 항을 판별 제거하여 전체 다항식 항의 개수를 줄임으로써, 비교적 적은 양의 학습 데이터를 사용함과 동시에 비선형성 표현의 향상을 도모하는 것이다[4,5]. 그러나 이와 같은 접근방식의 가장 큰 문제점은 다항식의 최대 차수를 미리 결정해 주어야 하고, 차수나 차원의 증가에 따른 고려해야 할 다항식 항의 개수가 매우 많아진다는 점이다.

본 논문에서는 유전적 프로그래밍(Genetic Programming, 이하 GP[3])을 이용하여 최소한의 데이터로 비선형성을 표현할 수 있는 임의의 차수를 갖는 다항식의 생성 방법을 다루고자 한다. 높은 차수의 다항식을 표현할 수 있는 GP 트리 생성 기법으로 Taylor 시리즈를 이용하는 방안을 모색하였다. 그런데 차수가 지나치게 높으면 너무 복잡한 다항식이 얻어지므로 본 논문에서는 가능한 한도 내에서 간단한 다항식을 얻기 위하여 다항식의 차수가 너무 커지게 되는 GP 트리의 생성을 제한하는 방법과 동시에 GAGPT(Group of Additive Genetic Programming Trees)[1,8]의 채용을 고려하였다. GAGPT를 사용하는 이유는 크기가 큰 GP 트리보다 크기가 작은 2개 정도의 GP 트리의 합이 보다 간단한 다항식을 얻을 수 있는 가능성이 높기 때문이다. 마지막으로 2개의 예제를 통하여 GP 기반의 최적 다항식 생성 방법의 유용성을 검증하였다.

## 2. 다항식 생성을 위한 유전적 프로그래밍

### 2.1 함수와 터미널 집합

다항식은 특별한 GP 트리의 한 형태에 불과하기 때문에 '+', '-', 그리고 '\*'만을

함수로 사용하면 GP 트리는 당연히 다항식이 된다. 그러나 이렇게 생성된 GP 트리가 높은 차수의 다항식이 되리라고 기대하기에는 무리가 따른다. 비선형성을 표현하기 위해서는 높은 차수를 갖는 다항식이 필요하므로, 본 연구에서는 앞에서 언급한 함수 이외에 다수의 수학 함수에 해당하는 Taylor 시리즈를 사용하여 GP 트리가 높은 차수의 다항식을 용이하게 표현할 수 있도록 유도하였다. Taylor 시리즈 또한 다항식의 형태이므로 여기서 고려할 사항은 시리즈의 몇 항까지 사용할 것인가 하는 것인데, 너무 많은 항을 사용하면 GP 트리가 표현하는 다항식이 지나치게 복잡해진다. 따라서 본 논문에서는 차수가 3차 이하가 되는 항까지만을 사용하였다. 다음은 정의된 함수와 터미널 집합이다.

$$F = \{+, -, *, g_i (i = 1, \dots, 18)\}$$

$$T = \{one, rand, x_i (i = 1, \dots, n)\}$$

$g_i$  에 대한 사항은 표-1에 나타나 있다. 일부  $g_i(x)$ 는  $x$ 의 범위에 따라서 Taylor 시리즈가 아닌 Laurent 시리즈가 되어야 하는 경우도 있으나, Laurent 시리즈는 다항식 형태가 아니므로 고려대상에서 제외 시켰다. 또한  $x$ 의 범위에 따라서 Taylor 시리즈는  $g_i(x)$ 에 근접할 수도 있고 전혀 다른 값을 줄 수도 있다.

GP에서 Taylor 시리즈를 사용한 것은 고차의 다항식을 얻기 위한 것이 주목적이기 때문에 시리즈의 값이 꼭  $g_i(x)$ 에 접근할 필요는 없다. 만일 트리의 진화 과정 중에 Taylor 시리즈가  $g_i(x)$ 에 근접할 필요가 있다면, GP의 특성 상  $x$ 의 값이 그에 상응하도록 트리가 진화할 것이다. 물론 진화 과정 중 Taylor 시리즈가 에 접근하도록 제어를 할 수도 있지만 GP 시스템의 구축이 복잡해지고, 이와 같은 제어가 없어도 원하는 결과를 얻을 수 있었기 때문에, 본 논문에서는 이 문제를 고려하지 않았다. 일반적으로 GP 트리  $g_i(x)$ 에서  $x$ 는 단순한 변수가 아니라 트리의 노드가 된다.  $[a_1](g_1[a_2]([a_3]g_1[a_4](g_1[a_5](g_1x_1))))$ 라는 트리를 예로 들면 ( $a_i$ 는 가중치로써 Hooke & Jeeves 기법을 사용하여 추정하였다) 다항식의 최고 차수가  $3 \times 3 \times 3 \times 3 = 81$  이나 되고 41 개의 항을 갖는 다항식이 된다. 종종 최고 차수가 컴퓨터의 정수 표현 한계를 넘는 경우도 발생한다. 이와 같이 지나치게 복잡한 다항식이 생성되는 것은 방지할 필요가 있는데, 이와 관련된 내용은 3절에 다루었다.

터미널 집합의 one은 값이 항상 1 인 상수를 뜻하고 rand 는 크기가 1 보다 작은 난수 (Random number)이며,  $x_i$ 는 설계변수이며, n은 설계변수의 개수를 의미한다.

[표-1] Taylor series defined in the function set of GP.

	Function	Taylor Series		Function	Taylor Series
$g_1$	$\sin(x)$	$x - \frac{1}{3!} x^3$	$g_2$	$\cos(x)$	$1 - \frac{1}{2!} x^2$
$g_3$	$\tan(x)$	$x + \frac{1}{3!} x^3$	$g_4$	$\log(1+x)$	$x - \frac{1}{2} x^2 + \frac{1}{3} x^3$
$g_5$	$\log\left(\frac{1+x}{1-x}\right)$	$2\left(x + \frac{1}{3} x^3\right)$	$g_6$	$\exp(x)$	$1 + \frac{1}{1!} x + \frac{1}{2!} x^2 + \frac{1}{3!} x^3$
$g_7$	$\sinh(x)$	$x + \frac{1}{3!} x^3$	$g_8$	$\cosh(x)$	$1 + \frac{1}{2!} x^2$
$g_9$	$\sqrt{1+x}$	$1 + \frac{1}{2} x - \frac{1}{8} x^2 + \frac{1}{16} x^3$	$g_{10}$	$(1+x)^{-1/2}$	$1 - \frac{1}{2} x + \frac{3}{8} x^2 - \frac{5}{16} x^3$
$g_{11}$	$(1+x)^{-1}$	$1 - x + x^2 - x^3$	$g_{12}$	$(1+x)^{-2}$	$1 - 2x + 3x^2 - 4x^3$
$g_{13}$	$\cos(x)^{-1}$	$\frac{\pi}{2} - \left(x + \frac{1}{2 \cdot 3} x^3\right)$	$g_{14}$	$\tan(x)^{-1}$	$x - \frac{1}{3} x^3$
$g_{15}$	$\cot(x)^{-1}$	$\frac{\pi}{2} - x + \frac{1}{3} x^3$	$g_{16}$	$\exp(\sin(x))$	$1 + x + \frac{1}{2!} x^2$
$g_{17}$	$\exp(\cos(x))$	$e^1 \left(1 - \frac{1}{2!} x^2\right)$	$g_{18}$	$\log(\cos(x))$	$-\frac{1}{2} x^2$

## 2.2 적합도 함수

2.1절에서 정의된 함수집합을 사용하여 GP 트리를 생성하면 비선형성이 큰 다항식이 생성되며, 또한 최소한의 학습 데이터만을 사용하기 때문에 Overfitting

현상이 발생한다. 따라서 본 논문에서는 전편에서 다루었던 DDBS(Directional Derivative-Based Smoothing)를 채용한 적합도 함수를 사용하였다. 이 때 모든 설계변수는 0 과 1 사이로 노말라이즈(Normalize)하였고, 근사화 하고자 하는 함수의 학습데이터는 0과 2 사이의 값이 되도록 스케일링 하였다.

$$\theta_{MSE} = \theta_{MSE} \theta_s (\theta_{MSE} - 1/2 \theta_s)^2 \quad (1)*$$

여기서

$$\begin{aligned} \theta_{MSE} &= 1/m \sum_{i=1}^m [f_{GP}(\overline{X}_i) - Y_i]^2 \theta_s \\ &= 1/(mpq) \sum_{i=1}^m \left[ \sum_{j=1}^p \left\{ \sum_{k=1}^q (\nabla f_{GP}(\overline{h}_{i,j}^k) \cdot \overline{D}_{i,j} - s_{i,j})^2 \right\} \right] \end{aligned}$$

\* 자세한 설명은 전편을 참조할 것.

### 3. 다항식의 복잡도 제어

#### 3.1 다항식의 최대 차수 제한

다항식의 복잡도를 측정하는 가장 정확한 방법은 항의 개수를 판별하는 것이다. 그러나 이것을 구현하려면 대단히 많은 시간과 노력이 소요되며, Mathematica와 연동 한다고 해도 복잡한 다항식을 전개하는데 Mathematica가 많은 시간을 소비함으로써, 그 효용성이 크지 않다고 할 수 있다. 본 논문에서는 그 대안으로 GP 트리로부터 다항식의 최대 차수를 판정하여 사용하는 방법을 강구하였다. 최대 차수 산정은 비교적 그 구현이 간단한데 스택(Stack)을 사용한 알고리즘을 기술하면 다음과 같다.

- 1) Clear Stack1 and Stack2. Locate the top most node of the tree. Travel all nodes of the GP tree from the top most node to the last node(leaf). At each node visited during the travel, Push a corresponding function or terminal into Stack1.
- 2) If Stack1 is empty, then n=Pop Stack2, and n is a maximum order of the GP tree, so return n.
- 3) t=Pop Stack1. If is a terminal Go to 4), Else Go to 5).
- 4) If t is 'rand' or 'one', Push 0 into Stack2, Else if t is xi, Push 1 into

- Stack2, and Go to 2).
- 5) If t is a gi, then n=a maximum order of gi, m=Pop Stack2. n=nxm, Push n into Stack2, and Go to 2), Else Go to 6).
- 6) m1=Pop Stack2, m2=Pop Stack2. If t is '+' or '-', then, n=max(m1,m2), Push n into Stack2, and Go to 2), Else if t is '\*', then n=m1+m2, Push n into Stack2, and Go to 2).

초기 개체군(Population) 생성 시, 생성된 GP 트리에 대하여 최대 차수를 판정하고 주어진 허용 차수보다 크면 그 트리를 제거하는 방법으로 모든 트리가 허용 차수보다 작은 다항식이 되도록 개체군을 만들 수 있다. 그런데 유전적 연산자를 사용하여 새로운 트리를 만들 경우 위에서 언급한 방법을 사용하면 문제가 발생하는데, 특히 교배 연산 (Crossover) 적용 시 무한 루프에 빠질 수 있다. 교배 연산은 두 개의 트리를 선정하고, 이 두 트리의 임의의 일부분을 서로 교환하여 새로운 두 개의 트리를 생성시키는데, 이 때 생성 가능한 모든 트리가 허용 최대 차수를 초과하는 상황이 종종 발생한다. 따라서 본 논문에서는 유전적 연산자를 적용하여 새로운 트리들을 생성할 때는 최대 허용 차수에 대한 구속 조건을 사용하지 않고, 생성된 새로운 트리가 허용 차수를 초과하면 트리의 적합도 값에 매우 큰 값을 부여하여 다음 진화과정에서 제외시키는 방법을 사용하였다.

### 3.2 GAGPT 체용

다항식의 복잡도를 제어할 수 있는 또 다른 방법은 GP 트리의 크기가 허용 범위를 넘지 않도록 구속하는 것인데, 보통 최대 허용 노드 수를 작은 값으로 사용하면 된다. 그런데, 이런 경우 발생하는 문제점은 GP 트리가 학습 데이터를 잘 Fitting 하지 못하거나 비선형성을 잘 표현하지 못한다는 것이다. 따라서 하나의 트리를 사용하는 것이 아니라 2개 이상의 크기가 작은 다중 트리를 사용하는 방안을 고려할 수 있다.

GAGPT[1,8]는 하나의 주 트리(Primary Tree)  $f_{GP}^p$  와 1개 이상의 보 트리(Auxiliary Tree)  $f_{GP}^{a,i}$  로 구성된다. 본 논문에서는 하나의 보 트리만을 사용하였다. 초기 진화 과정에서는  $f_{GP}^p$ 만 생성되고, 진화 과정을 통하여  $f_{GP}^p$ 가 찾아지면 그 다음엔 최적의  $f_{GP}^{a,i}$ 를 찾는 진화 과정이 시작된다. 이때 사용되는 적합도 함수의 값은  $f_{GP}^p$ 의 적합도 계산에 필요한 출력 값에  $f_{GP}^{a,i}$ 의 대응되는 출력 값을 합산하여 계산하게 된다. 즉 (2)와 (3)식을 사용하여 (1)식의 적합도 함수의 값을

계산한다.

$$\theta_{MSE} = 1/m \sum_{i=1}^m [f_{GP}^p(\bar{X}_i) + f_{GP}^{a,i} - Y_i]^2 \quad (2)$$

$$\theta_s = 1/(mpq) \sum_{i=1}^m \left[ \sum_{j=1}^p \left\{ \sum_{k=1}^q ((\nabla f_{GP}^p(\bar{h}_{i,k}^k) + \nabla f_{GP}^{a,i}(\bar{h}_{i,j}^k)) \cdot \bar{D}_{i,j} - s_{i,j})^2 \right\} \right] \quad (3)$$

이런 방식으로 두 개의 트리를 사용하면 학습오류를 크게 줄일 수 있고 비선형성이 큰 함수도 용이하게 근사가 가능하다. 또한 크기가 큰 하나의 GP 트리 보다는 크기가 작은 두 개의 GP 트리의 합에서 보다 간단한 다항식을 얻을 확률이 높다.

#### 4. GP 트리로부터 다항식의 생성

GP 트리는 오퍼레이터(Operator)가 오퍼랜드(Operand)의 앞에 위치한 Prefix 연산자 형태의 수식이라고 볼 수 있다. 통상적으로 사용되는 다항식은 오퍼레이터가 두 오퍼랜드 가운데 위치하는 Infix 연산자 형태의 수식이다. 결국 일반적인 형태의 다항식을 얻는 과정은 Prefix 형태의 수식인 GP 트리에서 Infix 형태로 전환하는 것인데, 전환하는 알고리즘은 이미 잘 알려져 있으므로 본 논문에서는 이에 대한 기술은 생략하였다. 얻어진 Infix 형태의 다항식은 최종적으로 Mathematica를 이용하여 보다 간결히 정리할 수 있다.

#### 5. 응용 예

[표-2]는 본 논문의 응용 예에 공동적으로 사용된 GP의 주요 파라미터이다.

[표-2]의 내용을 살펴보면 최적의 주 트리와 보 트리를 생성하기 위해서 사용된 허용 세대 수(Allowable number of generation)가 각각 10, 5이다. 주 트리와 보 트리의 조합을 생각하지 않고, 하나의 GP 트리 생성만을 고려한다면, 이와 같이 충분하지 못한 세대 수로는 만족할만한 최적의 GP 트리를 얻기 어렵다. 그러나 GAGPT에서는 만족할 만한 주 트리가 얻어지지 못하더라도 10세대 이후에 보 트리가 첨가되고, 이에 따라 급격한 학습오류의 감소 현상을 보이게 된다. 따라서 주 트리와 보 트리의 합인 GAGPT는 허용 세대 수가 작더라도 원하는 근사화 성능을 발휘할 수 있게 되며, 보 트리가 계속 첨가됨에 따라서 학습오류

는 계속해서 감소하게 되는 경향을 갖는다. 또한 GP에서는 유전적 알고리즘과는 달리 세대 수를 증가 시키는 것보다는 큰 개체군을 사용하는 것이 최적의 GP 트리를 생성하는데 보다 용이하다고 할 수 있다.

[표-2] Parameters used in the GAGPT algorithm with DDBS.

Allowable maximum generation for $f_{GP}^p$	10
Allowable maximum generation for $f_{GP}^{a,i}$	5
Selection method	Tournament with 30 trees
Reproduction probability	0.15
Crossover probability	0.7
Mutation probability	0.15
p in (3)	2
q in (3)	40

### 5.1 Goldstein-Price Function

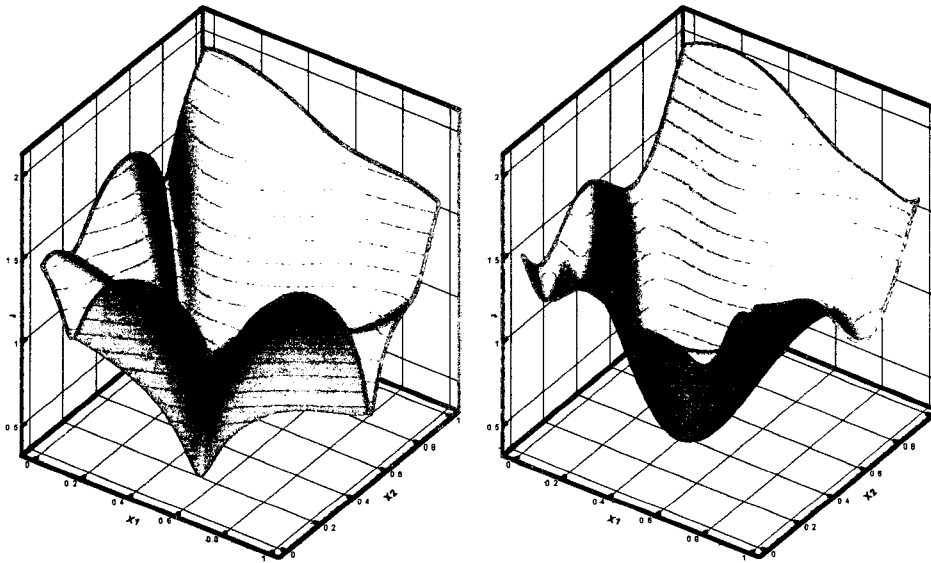
$$\begin{aligned}
 f(x_1, x_2) = & (1 + (x_1 + x_2 + 1)^2 \cdot (19 - 14x_1 + 3x_1^2 - 14x_2 + 14x_2^2 + 6x_1x_2 + 3x_2^2)) \\
 & \cdot (30 + (2x_1 - 3x_2)^2 \cdot (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)) \\
 & - 2 \leq x_i \leq 2 \quad i = 1, 2 \quad (4)
 \end{aligned}$$

Goldstein-Price function은 하나의 전역점 최적점 (Global optimum point)과 두 개의 지역적 최적점 (Local optimum point)을 갖도록 인위적으로 만들어진 함수인데, 최적화 알고리즘의 성능을 테스트하기 위해서 가장 일반적으로 사용되는 함수라고 할 수 있다. 그런데, 함수 값이 0에서 1E10 까지 변하기 때문에 적은 학습 데이터를 사용하여 정확히 근사화한다는 것은 대단히 어려운 일이라고 할 수 있다. 즉 근사화 시 1E10 근처의 학습 데이터의 영향이 지대하기 때문에 실제로 근사적 최적화에서 중요한 0 근처의 학습 데이터 기여도가 매우 미미하게 되는 문제점이 있다. 따라서 본 논문에서는 (5)식과 같이 로그 스케일을 취하였고, 그림-1.a 에 그 결과가 가시화 되어 있다. 그러나 이 경우도 매우 심한 비선형성을 보이기 때문에 정확한 근사화를 기대하기 어렵지만 근사화 성능을 측정하는데 사용되는 Benchmarking problem으로 활용할 수 있다.



$$f_{\log}(x_1, x_2) = \log(1 + f(x_1, x_2)) \quad (5)$$

(5)식을 근사하기 위해서 사용한 개체군의 크기는 5000 이며, GP 트리의 최대 허용 노드 수는 20 이다. 최대 허용 차수는 27 이며, 학습데이터는 5×5 Grid 타입으로 생성하였고, 테스트 데이터는 200×200 개이다. DDBS를 채용한 GAGPT의  $\theta_{MSE}$  는 1.4237E-2 이고 테스트 오류는 2.100E-2 이다. 그림-1.b는 최적의 GAGPT 결과를 가시화한 것이다. 만일 최고 차수의 값을 증가시키면 보다 정확한 모델링이 가능하지만, 이것은 곧 다항식이 복잡해진다는 것을 의미한다. 그림-2는 GAGPT로부터 얻은 다항식을 Mathematica로 간결히 정리한 다항식이다. Goldstein-price function은 비선형성이 크기 때문에 최대 허용 차수가 27 로 비교적 크고, 매우 큰 개체군을 사용했음에도 불구하고 일반 수학 함수를 사용한 전편의 GP 결과와 비교해 볼 때 테스트 오류가 2 배정도 크게 나타난다.



a. The original function.                      b. GAGPT with DDBS

[그림-1] The results of the GAGPT that adopt DDBS.

일반적으로 응답면 모델링에서는 (5)식을 근사하기 위해서 2 차 다항식을 사용하게 되고, 다항식의 계수는 OLS(Ordinary least squared) 방법[4,5]을 사용한다. 이런 경우 Fitting 된 다항식은 너무 Smooth 하기 때문에 원래의 함수 특성

을 제대로 표현하지 못한다는 것은 주지의 사실이다. 또한 다항식의 차수를 학습 데이터의 개수인 25 근처로 정하고, OLS를 사용하여 다항식의 계수를 산정한 후 그 결과를 가시화해 보면 대부분의 경우 판독이 불가능할 정도로 Overfitting 현상이 발생하게 됨을 관찰할 수 있다. 본 논문은 주로 GP 를 이용하여 최적의 다항식 생성 기법을 다룬 것이기 때문에, 응답면 모델링 사용될 수는 있는 근사화 기법인 Krigging, 인공신경망, 또는 Lazy learning(Locally weighted regression) 등에 대한 비교 연구는 수행하지 않았다. 그러나 응답면 모델링 관점에서 이와 같은 비교 연구는 흥미 있는 추후의 연구과제라고 할 수 있다.

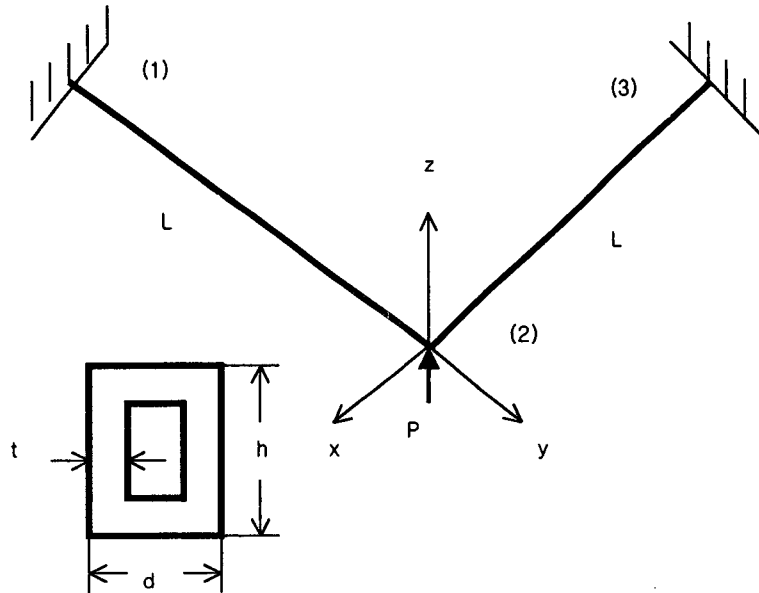
$$\begin{aligned}
 &0.0015625 (-2.95955 - 0.0465424 \\
 &\quad (1. - 0.125 (1. + x_1^2) (2. + x_1^2)) \\
 &\quad (-0.628053 + x_2) (3.22451 - 0.153915 x_2 + x_2^2) \\
 &\quad (6.24746 - 0.0522899 x_2 + x_2^2) \\
 &\quad (0.508096 + 0.206205 x_2 + x_2^2) \\
 &\quad (5.36725 + 0.628053 x_2 + x_2^2)) + \\
 &0.35 (-0.890001 x_1 + 0.319664 x_1^3 + 0.104501 x_1^5 + \\
 &\quad 0.00915453 x_1^7 + 0.00026732 x_1^9 + \\
 &\quad 0.825 (0.288693 x_1^2 - 0.486842 x_1^4 - \\
 &\quad 0.760335 (-13.9757 + x_2) (0.229703 + x_2) \\
 &\quad (0.632784 - 1.58471 x_2 + x_2^2) \\
 &\quad (1.00348 + 0.165435 x_2 + x_2^2)) - \\
 &0.388032 \text{ Power}[0.274435 x_1^2 - 0.462799 x_1^4 - \\
 &\quad 0.722784 (-13.9757 + x_2) (0.229703 + x_2) \\
 &\quad (0.632784 - 1.58471 x_2 + x_2^2) \\
 &\quad (1.00348 + 0.165435 x_2 + x_2^2), 3])
 \end{aligned}$$

[그림-2] The polynomial for the Goldstein-price function. In the figure the symbol 'Power[x, n]', represents  $x^n$ .

## 5.2 Two - Member Frame

[그림-3]은 Two-member frame을 나타낸 것으로 응력(Stress)에 관련된 2 개의 구속 조건은 (6)식과 같다[2,7]. Two-member Frame 문제는 응답면 기법을 채용한 근사 최적화 기법의 성능을 계측하기 위해서 자주 사용되는 문제 중에 하나이다. 두 개의 부재로 구성된 구조물으로써 (6)식의 조건을 만족시키는 동시에

부재의 단면적이 최소가 되도록 두께(t), 폭(d), 높이(h)를 구하는 것이 그 주된 내용이라고 할 수 있다. 본 논문은 응답면의 모델링에 초점을 두고 있기 때문에 (6)식을 적절히 모델링 할 수 있는 다항식을 GP를 사용하여 구하고자 한다.



[그림-3] Two-member frame.

$$f_i(x_1, x_2, x_3) = \{(\sigma_i + 3\tau^2)/\sigma_a^2(\leq 1.0)\}_{i=1,2} \quad (6)$$

여기서

$$\sigma_i = M_i h / (2I) \quad (i = 1, 2), \quad \tau = T / (2At) \quad M_i = 2EK(-3U_1 + U_2L) / L^2$$

$$M_2 = 2EK(-3U_1 + 2U_2L) / L^2, \quad T = -GJU_3 / L$$

$$I = (1/12)[dh^3 - (d-2t)(h-2t)(h-2t)^3], \quad J = 2t[(d-t)^2(h-2t)^2] / (d+h-2t)$$

$$A = (d-t)(h-t)$$

$$\frac{EI}{L^3} \begin{bmatrix} 24 & -6L & 6L \\ -6L & 4L^2 + \frac{GI}{EI}L^2 & 0 \\ 6L & 0 & 4L^2 + \frac{GI}{EI}L^2 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} = \begin{bmatrix} P \\ 0 \\ 0 \end{bmatrix}$$

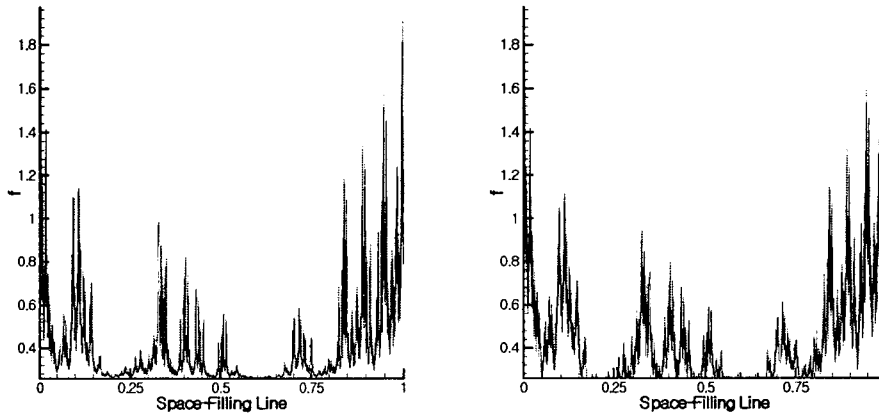
$$2.5\text{in} \leq d \leq 10\text{in}, \quad 2.5\text{in} \leq h \leq 10\text{in} \quad 0.1\text{in} \leq t \leq 1.0\text{in},$$

$\sigma_a=40,000\text{psi}$ ,  $E=3.0E7\text{psi}$   $G=1.154E7\text{psi}$ ,  $P=-10,000\text{lbs}$   $x_1=d$ ,  $x_2=h$ ,  $x_3=t$

$U_1$ ,  $U_2$  그리고  $U_3$ 는 각각 노드 (2)의 수직 이동(Displacement), 부재(3)-(2)에 대한 회전, 그리고 부재(1)-(2)에 대한 회전을 뜻한다. 설계변수가 3 개인, 3 차원 문제이며, 두 개의 구속조건 식은 (7)식과 같이 로그 스케일을 취하였다.

$$f_{\log}^i = \log(2 + f_i(x_1, x_2, x_3)) \quad i=1, 2 \quad (7)$$

두 개의 구속조건 식은 형태가 유사하므로  $f_{\log}^1$ 만 DDBS GAGPT로 근사화를 시도하였다. 학습 데이터는 Grid 타입의  $3 \times 3 \times 3=27$  개이며, 테스트 데이터는  $34 \times 34 \times 34=39,304$  개이다. 개체군의 크기는 1500개이며, 허용 최대 차수는 15, 트리의 최대 허용 노드 수는 15이다.



a. The original function.

b. GAGPT with DDBS.

[그림-4] The results of DDBS GP plotted by using the space-filling curve. The f-axis represents the value of  $f_{\log}^1$ .

그림-4는 DDBS GAGPT를 가시화한 것인데, 3 차원의 테스트 데이터를 Sierpinski's Curve[6]를 사용하여 [0,1] 사이의 실수 값으로 일대일 매핑하고 이에 대응하는 GAGPT 값을 도시화한 것이다. 약간의 지역적인 차이는 있으나 전체 그래프의 윤곽은  $f_{\log}^1$ 의 특징을 잘 표현하고 있다. 학습 오류( $\theta_{MSE}$ )는  $6.798E-4$ , 이고 테스트 오류는  $5.585E-3$ 이다. 그림-5는 GAGPT로부터 얻은 다항

식을 나타낸 것이다. 학습과 테스트 오류의 값을 Goldstein-Price Function의 경우와 비교해보면, 상대적으로 그 값이 매우 작은 것을 알 수 있는데, (6)식의 비선형성이 (5) 보다 적어, 비록 3 차원 문제라고 하더라도 그 결과가 비교적 좋게 나타났다고 판단된다. 근사화 관점에서는 학습오류와 테스트 오류의 값이 서로 비슷하게 나타나는 것이 바람직하나, 본 예에서는 학습오류가 테스트 오류보다 작는데, 이것은 생성된 최적의 다항식이 약간의 Overfitting 현상을 보이기 때문이라고 판단된다.

$$\begin{aligned}
 & -0.400911 (-0.0723626 + 5.64456 x_1 - 12.208 x_1^2 + 17.494 x_1^3 - \\
 & 16.6488 x_1^4 + 8.92624 x_1^5 - 2.132 x_1^6 - 2.74039 x_2 + \\
 & 2.4156 x_2^3) (0.0416099 x_1 - \\
 & 0.0304064 (-0.303027 + x_3) (11.2033 - 2.13101 x_3 + x_3^2) - \\
 & 0.423647 \text{Power}[0.0304891 x_1 - \\
 & 0.0222799 (-0.303027 + x_3) (11.2033 - 2.13101 x_3 + x_3^2) \\
 & , 3]) + 0.290356 (0.686969 x_1 + \\
 & 0.300848 (3.39476 + 0.170886 x_1^2 - 1.24009 x_2 - \\
 & 0.393074 x_2^2 + 0.685583 x_2^3 + \\
 & x_1 (-0.904688 - 0.7989 x_3) - 1.50442 x_3 + \\
 & 0.933726 x_3^2 - 0.00109983 \\
 & \text{Power}[-23.2 + 8.76443 x_1 - 1.65551 x_1^2 + 12.0137 x_2 + \\
 & 3.80802 x_2^2 - 6.64179 x_2^3 + 14.5745 x_3 + \\
 & 7.73958 x_1 x_3 - 9.04575 x_3^2, 3]) + \\
 & 0.0106549 \text{Power}[23.2 - 8.76443 x_1 + 1.65551 x_1^2 - \\
 & 12.0137 x_2 - 3.80802 x_2^2 + 6.64179 x_2^3 - \\
 & 14.5745 x_3 - 7.73958 x_1 x_3 + 9.04575 x_3^2, 2]))
 \end{aligned}$$

[그림-5] Polynomial obtained from the GAGPT for  $f_{\log}^1$ .

## 5. 결론

본 논문에서는 최소한의 학습 데이터가 주어진 조건에서 GP를 이용하여 비선형 응답면을 모델링 할 수 있는 최적의 다항식을 생성하는 방법을 제시하였다. 다항식을 표현한 GP 트리가 비선형성을 갖도록 함수 집합에 다양한 수학 함수에 대한 3차 이하의 차수를 갖는 Taylor 시리즈를 사용하였다. 이런 방식으로 생성된 다항식은 비선형성을 잘 표현할 수 있으나 다항식의 항이 지나치게 많아지는 경

향이 있어, 비교적 간단한 다항식의 생성을 위하여 최대 차수를 제한하는 방법과 함께 GAGPT를 채용하였다. 2개의 응용 예를 살펴보면 대체적으로 만족할 만한 결과를 주는 것으로 판단할 수 있으나 일반 수학 함수를 사용하여 생성된 GP 트리 보다는 성능이 약간 떨어질 것으로 판단된다. 물론 다항식의 허용 최대 차수를 크게 하거나 더 높은 차수의 Taylor 시리즈를 사용하면, 일반 수학 함수를 사용하여 생성된 GP 트리와 동등한 성능을 보이게 된다.

GP의 입장에서 볼 때, GP 트리를 다항식으로만 생성하게 되면 연산 오류가 없고, 비교적 Overfitting의 경향이 줄어든다는 장점이 있다.

비교적 크기가 작고, 일반적인 GP 트리과 성능이 대등한 다항식을 생성하는 방법 중에 하나는  $g_i(x)$ 를 표현한 낮은 차수의 Taylor 시리즈가  $g_i(x)$ 를 보다 근접할 수 있도록 하는 것인데, 이것은  $x$  값의 크기에 의존한다. 일반적으로 이 값이 작을수록 Taylor 시리즈는  $x$ 에 근접하게 된다. 현재, 이와 관련하여 진화 과정 중에 트리가 이런 거동을 보이도록 제어할 수 있는 전략에 대해 연구를 수행하고 있다.

## 참고문헌

- [1] 연윤석, 가중 선형 연상기억을 채용한 유전적 프로그래밍과 그 공학적 응용, 한국 CAD/CAM 학회 논문집, Vol.3, No.1, pp.57-67, 1998.
- [2] Arora, J.S., *Introduction to Optimum Design*, McGraw-Hill, New York, 1989.
- [3] Koza, J.R., *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, The MIT Press, 1992.
- [4] Myers, R.H. and Montgomery, D.C., *Response Surface Methodology: Process and product Optimization Using Designed Experiments*, John Wiley & Sons, Inc., 1995.
- [5] Ott, R.L., *An introduction to Statistical Methods and Data Analysis*, Wadsworth Inc., 1993.
- [6] Perez-Cortes, J.C., Vidal, E. The Extended General Spacefilling Curves Heuristic, *Int. Conf. on Pattern Recognition ICPR-98*, Brisbane, Australia, 1998.
- [7] Simpson, T.W., Allen, J.K., and Mistree, F., Spatial Correlation and Metamodels for Global Approximation in Structural Design Optimization, *Proc. Of DETC98*, ASME, 1998.
- [8] Yeun, Y.S., Suh, J.C., and Yang, Y.S., "Function Approximation by Superimposing Genetic Programming Trees: with Applications to Engineering Problems", *Information Sciences*, Vol.122, Issue 2-4, 2000.

## **Response Surface Modeling by Genetic Programming II: Search for Optimal Polynomials**

Wook Rhee, Nam-Joon Kim

### **Abstract**

This paper deals with the problem of generating optimal polynomials using Genetic Programming(GP). The polynomial should approximate nonlinear response surfaces. Also, there should be a consideration regarding the size of the polynomial. It is not desirable if the polynomial is too large. To build small or medium size of polynomials that enable to model nonlinear response surfaces, we use the low order Taylor series in the function set of GP, and put the constrain on generating GP tree during the evolving process in order to prevent GP trees from becoming too large size of polynomials. Also, GAGPT(Group of Additive Genetic Programming Trees) is adopted to help achieving such purpose. Two examples are given to demonstrate our method.

**Key Words** : polynomial, response surface, genetic programming, group of additive genetic programming

◆저자소개◆

이 옥(Wook, Rhee )



승실대학교 대학원 전기공학과 공학 박사학위를 취득하였으며, 1981년 3월에서 1992년 2월간 전주공업대학 전기과 부교수로 역임하였고 현재 대진대학교 전기공학과 교수로 재직중이다. 관심분야는 유전적 알고리즘/프로그램 이용한 전력설비 모델링 및 고장진단이다.

E-mail : rheewook@road.daejin.ac.kr

Tel : 031-539-1911 (mobile: 016-366-0594)

김남준(Nam Jun, Kim)



한양대학교 대학원 전기공학과 공학박사학위를 취득하였으며, 현재 대진대학교 전기공학과 교수로 재직중이다. 관심분야는 유전적 알고리즘/프로그램을 이용한 파라미터 추정 및 시스템 동정이다.

E-mail : njkim@road.daejin.ac.kr

Tel : 031-539-1912 (mobile: 017-539-1912)