

# 수정된 GMDH 알고리즘을 이용한 비선형 동적 시스템의 모델링

## Modeling of Nonlinear Dynamic Systems Using a Modified GMDH Algorithm

홍연찬 · 임상수

Yeon Chan Hong and Sang Su Ohm

인천대학교 전자공학과

### 요 약

GMDH(Group Method of Data Handling)는 복잡한 비선형 시스템을 인식하는 데 유용한 데이터 분석 기법이다. 따라서 본 논문에서는 비선형 다이내믹 시스템을 모델링하기 위한 GMDH 알고리즘의 적용 방법을 제안한다. GMDH를 사용한 다이내믹 시스템의 인식은 일련의 입출력 데이터를 인가하여 필요한 계수들의 집합을 동적으로 산출함으로써 이루어진다. 또한, 본 논문에서는 데이터를 취사선택하는 기준을 순차적으로 감소시킴으로써 GMDH의 단점인 계산량의 과다를 방지하는 방법도 제안하였다.

### ABSTRACT

The GMDH(Group Method of Data Handling) is a useful data analysis technique for identification of nonlinear complex systems. Therefore, in this paper the application method of GMDH algorithm for modeling nonlinear dynamic systems is proposed. The identification of dynamic systems by using GMDH consists of applying a set of input/output data and computing the necessary coefficient set dynamically. Also, in this paper, by reducing sequentially the criterion which can adopt or reject the data, a method to prevent excessive computation that is a disadvantage of GMDH is proposed.

### 1. 서 론

시스템의 수학적 모델링은 시스템의 인식, 예측, 최적화, 제어를 위한 시스템 분석에 사용된다. 로보트, 비행기, 환경, 사회, 경영 시스템과 같은 복잡한 시스템의 모델링 문제는 그 시스템들의 특성이 비선형이고 차수가 크기 때문에 일반적으로 난해하다. 이러한 복잡한 시스템들에 대하여 시스템 인식 문제를 해결하기 위한 많은 연구들이 최근에 활발하게 진행되고 있으며, 그 중에서 Ivakhnenko[1,2]에 의해 제안된 GMDH에 의한 인식 방법이 잘 알려져 있다. GMDH 구조의 기본 개념은 발견적 자기구성(heuristic self-organization)에 기초를 두고, 그 결과로 생성되는 다항식 방정식은 Kolmogorov-Gabor 다항식[3]과 관련된다. GMDH의 구조는, 각 층(layer)에서 유용한 데이터를 자체적으로 추출하는 한계값을 갖는 다층의 알고리즘을 통하여 정확성을 증가시키기 위해 해로

운 정보를 제거하는 일종의 발견적 필터로 볼 수 있다[4].

GMDH 알고리즘의 기본 구조와 여러 가지 형태를 Farlow[5]가 잘 설명하고 있으며 GMDH 알고리즘의 개선 방법에 관한 연구들도 많은 연구자들에 의해 수행되었다. 그 중에서 Ivakhnenko[6]는 instrumental variable을 사용하여 모델링의 성능을 제고시켰고 Yoshimura[7]는 Akaike 정보기준[8]을 사용한 RGMDH(Revised GMDH)를 제안하였다. Hayashi[9]는 확률적 회귀 분석을 사용한 퍼지 GMDH를 제안하였다. 실제의 시스템에 GMDH를 응용한 사례도 많이 찾아 볼 수 있는데 그 중에서 Chao[10]가 생산 시스템을 GMDH로 모델링하였고 Chen[11]은 로보트에 GMDH를 응용하였다.

이상의 GMDH에 관한 연구들은 공통된 한 가지 단점을 갖고 있는데, 계산량의 과다가 바로 그것이다. 따라서 본 논문에서는 계산량이 과다하게 발생하는

\* 본 연구는 1996년도 인천대학교 연구비 지원에 의해 수행되었습.

것을 방지하기 위해서 데이터를 취사선택하는 기준을 순차적으로 감소시키는 방법을 제안한다. 또한, 일련의 입출력 데이터를 인가하여 필요한 계수들의 집합을 동적으로 산출함으로써, 비선형 다이내믹 시스템을 모델링하기 위한 GMDH 알고리즘의 적용 방법도 제안한다.

## 2. 수정된 GMDH 알고리즘

일반적으로 차분 방정식(difference equation)에 의한 비선형 다이내믹 시스템의 상태 방정식은 다음과 같다.

$$\begin{aligned} x(k+1) &= \Phi[x(k), u(k)] \\ y(k) &= \Psi[x(k)] \end{aligned} \quad (1)$$

이때,  $x(k)$ ,  $y(k)$ ,  $u(k)$ 는 각각 상태, 출력, 입력을,  $\Phi[\cdot]$ 와  $\Psi[\cdot]$ 는 비선형 함수를 나타낸다. 이와 같은 비선형 차분 방정식을 재구성하면 다음과 같은 식으로 나타낼 수 있다[12].

$$\begin{aligned} y(k+1) &= f[y(k), y(k-1), \dots, y(k-n+1), \\ &u(k), \dots, u(k-m+1)] \end{aligned} \quad (2)$$

이때  $f[\cdot]$ 는 비선형 함수이다.

GMDH는 입력 노드(input node)들의 집합으로 이루어져 있으며, 이 입력 노드들로부터 출력을 얻기 위하여 중간층들을 거치게 된다. 각 층에서의 노드들은 바로 전단계 층의 노드들의 출력들로부터 두개씩을 취하여 출력을 만들어낸다. GMDH의 구성도를 그림 1에 나타내었다. 그림 1에서  $z_{s,i}$ 는 s번째 층과 i번째 노드의 출력을 나타내며 m은 입력의 갯수이다. 두번째 층은 i개의 노드들로 구성되어 있는데,  $i < \frac{m(m-1)}{2}$ 이고

$j < \frac{i(i-1)}{2}$ 이다. 이 부등식들이 성립하는 이유는 뒤에서 언급할 데이터 선택 기준에 의해 영향이 적은 데

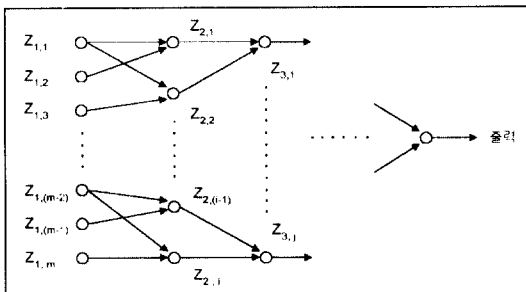


그림 1. GMDH의 구성도.

이터를 버리기 때문이다. 최종적으로 하나의 출력을 얻게 되는 과정도 데이터 선택 기준의 최소값에 의해 결정된다. 모든 층의 각각의 노드들의 출력은 GMDH 알고리즘의 일반적인 표현에 따라 다음 식과 같이 2차 방정식의 형태를 취하게 된다.

$$\begin{aligned} z_{s,i} &= a_{s,i} z_{(s-1),u}^2 + b_{s,i} z_{(s-1),v}^2 + c_{s,i} z_{(s-1),u} + d_{s,i} z_{(s-1),v} \\ &+ e_{s,i} z_{(s-1),u} z_{(s-1),v} + f_{s,i} \end{aligned} \quad (3)$$

이때  $a_{s,i}, b_{s,i}, c_{s,i}, d_{s,i}, e_{s,i}, f_{s,i}$ 는 노드의 연결강도(connection weight)이고, 아래첨자 u와 v는 임의의 u번째와 v번째 노드를 가리킨다.  $z_{(s-1),u}^2$ 와  $z_{(s-1),v}^2$ 는 벡터로서 각 요소들을 제공한다든 것을 의미하고,  $f_{s,i}$ 는 각 요소에  $f_{s,i}$ 를 더한다는 것을 의미한다.

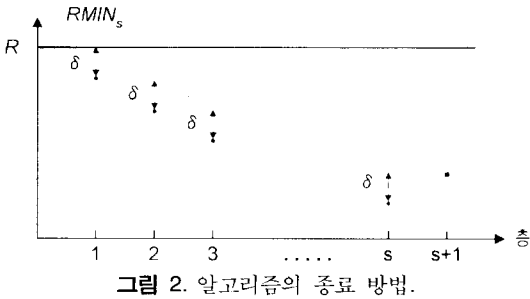
다시 말해서, GMDH의 원리는 예를 들어 3번째 층으로 들어오는 i개의 입력들의 조합의 수( $c_2^i$ )에 의해서 다음 층의 노드 수가 결정되고 난 후, 이 노드들의 출력이 (3)식에 의해 각각 계산되며, 계산된 출력의 적합성 여부를 성능 기준에 의하여 판별한다는 것이다. 이처럼 각각의 층에서의 적합치 못한 노드들을 판별해내기 위해서는, 다음과 같이 원하는 최종 출력과 실제 노드 출력들 사이의 에러를 계산하는 것이다.

$$r_{s,i} = |y(k) - (z_{s,i})| \quad (4)$$

여기서  $(z_{s,i})$ 은 s번째 층과 i번째 노드의 출력인  $z_{s,i}$  벡터의 마지막 요소를 의미한다. 이 에러의 크기를 구하여 큰 에러를 가진 노드는 없애고 에러가 작은 노드들만을 가지고 앞서 설명한 과정들을 원하는 출력을 얻을 때까지 반복하게 되는 것이다. 그리고 이 알고리즘은 일반적으로 사용되고 있는 신경망 구조와는 달리 중간층의 수와 노드들의 수가 미리 정해지는 것이 아니다.

일반적인 GMDH 알고리즘에서는 데이터를 취사선택하는 기준으로서 미리 정한 일정한 값 R을 에러  $r_{s,i}$ 와 비교하여  $r_{s,i}$ 가 R보다 작으면 그 노드의 출력을 취하고  $r_{s,i}$ 가 R보다 크면 그 노드의 출력은 버린다. 그러나 이러한 방식을 사용하면, 층이 증가함에 따라서 노드의 출력이 실제의 출력에 점점 가까워지기 때문에 잔존하는 노드의 수가 점점 증가한다. 따라서 알고리즘의 계산량이 증가하는 단점이 발생한다. 그러므로 본 논문에서는 기준 R을 순차적으로 감소시키므로써 계산량의 과다를 방지하는 방법을 제안한다.

우선 알고리즘의 종료 방법을 설명하면 각 층에서의 에러의 최소값  $RMIN_s (s=1, 2, \dots)$ 를 구하여 이  $RMIN_s$ 가 최소가 되는 층에서 알고리즘을 종료한다. 이 과정을 그림 2에 나타내었다. 그림 2에서 점으로



표시된 것이  $RMIN_s$ 이다. 그러므로  $s$ 층에서 최소의 에러를 갖는 노드의 출력이 실제의 출력의 추정치가 된다.

그림 2에서 알 수 있듯이 일반적인 GMDH 알고리즘에서는 에러가  $R$ 보다 작은 노드들이 모두 선택되며 층이 증가함에 따라서  $RMIN_s$ 와  $R$  사이의 간격이 커지므로 노드의 수가 크게 증가한다. 따라서 본 논문에서는 노드의 수가 크게 증가하는 것을 막기 위해 그림 2에 표시했듯이 다음과 같은 기준을 선정한다.

$$R_s = RMIN_s + \delta \quad (5)$$

여기서  $\delta$ 는 미리 정한 일정한 값이다. 선택 기준을 (5)식과 같이 정함으로써 노드의 수가 크게 증가하는 것을 막을 뿐만 아니라 데이터를  $RMIN_s$ 에 대하여 균등하게 선택할 수 있다는 것을 알 수 있다.

### 3. 비선형 다이내믹 시스템의 모델링

(2)식으로 나타낸 비선형 다이내믹 시스템을 모델링하기 위한 GMDH 알고리즘을 다음과 같이 3 단계로 나누어 설명할 수 있다.

[단계 1] 변수벡터의 구성

(3)식에 의하여  $k$  순간에서  $s$ 층의  $t$ 번째 노드의 출력  $z_{s,t}(k)$ 를 만들어 내며 다음식과 같이 간단하게 표현할 수 있다.

$$z_{s,t}(k) = X_{s,t} \theta_s \quad (6)$$

이때,

$$X_{s,t} = [z_{(s-1),u}^2(k); z_{(s-1),v}^2(k); z_{(s-1),u}(k); z_{(s-1),v}(k); z_{(s-1),u}(k)z_{(s-1),v}(k); 1]$$

$$\theta_s = [a_{s,t}; b_{s,t}; c_{s,t}; d_{s,t}; e_{s,t}; f_{s,t}]^T$$

여기서  $X_{s,t}$ 은 차수가  $p \times 6$ 인 데이터 행렬로서 두 벡터 입력  $z_{(s-1),u}(k)$ 와  $z_{(s-1),v}(k)$ 를 합성한 것이다.  $(k+1)$ 인 순간에 첫번째 층의 입력과 출력은 그림 3과 같다. GMDH로 비선형 다이내믹 시스템을 모델링 할 때 인가되는 입력 벡터는  $(m+n)$ 개이다. 각 벡터의 요소는  $p$ 개이며 임의로 정해진다.  $p$ 의 크기에 따라서 인가되는 데이터의 양이 결정된다. (6)식은  $\theta_s$ 와 함께 2차 방정식을 만들어낸다. 이때  $z_{s,t}(k)$ 를 얻기 위해서는, 먼저 (6)식을 풀어 계수 벡터인 연결강도  $\theta_s$ 를 얻어야 한다. 각 노드의 출력  $z_{s,t}(k)$ 가 최종적으로 시스템의 출력  $y(k)$ 가 되도록 하는 것이 이 알고리즘의 목적이므로 그 일반식은 다음과 같다.

$$\theta_s = (X_{s,t}^T X_{s,t})^{-1} X_{s,t}^T y(k) \quad \text{if } p \geq 6 \quad (7)$$

$$X_{s,t}^T (X_{s,t} X_{s,t}^T)^{-1} y(k) \quad \text{if } p < 6$$

[단계 2] 영향이 적은 데이터의 제거

앞서 설명했듯이, 단계 1에서 구해진 모든 노드들의 출력들이 다음층으로 전달되는 것은 아니다. 즉 적자 생존의 원칙을 적용해야 한다. 일반적인 GMDH에서는 출력과 새로 생성된 데이터를 훈련 부분과 검사 부분으로 나누어야 한다. 그래서 출력과 생성된 데이터의 훈련부분 과정에서 구해진 평균제곱에러에 의해 필요없는 노드들을 제거한다. 이를 수행하는 방법들에는 여러 가지가 제시되고 있으나 본 논문에서는 비선형 시스템을 동적으로 모델링하고자 하므로 그림 3의 모든 데이터를 훈련 부분에서 사용하고 (4)식과 같이 데이터 벡터의 마지막 요소를 검사 부분으로 사용하였다.

$r_{s,t}$ 값의 증가순으로  $s$ 층의 모든  $z_{s,t}(k)$ 를 다시 정렬

| 출 령        | 입 령        |                  |              |            |                  |              |
|------------|------------|------------------|--------------|------------|------------------|--------------|
| $y(k-p+2)$ | $u(k-p+1)$ | $u(k-p) \dots$   | $u(k-m-p+2)$ | $y(k-p+1)$ | $y(k-p) \dots$   | $y(k-n-p+2)$ |
| $y(k-p+2)$ | $u(k-p+2)$ | $u(k-p+1) \dots$ | $u(k-m-p+3)$ | $y(k-p+2)$ | $y(k-p+1) \dots$ | $y(k-n-p+3)$ |
| $\vdots$   | $\vdots$   | $\vdots$         | $\vdots$     | $\vdots$   | $\vdots$         | $\vdots$     |
| $y(k)$     | $u(k-1)$   | $u(k-2) \dots$   | $u(k-m)$     | $y(k-1)$   | $y(k-2) \dots$   | $y(k-n)$     |
| $y(k+1)$   | $u(k)$     | $u(k-1) \dots$   | $u(k-m+1)$   | $y(k)$     | $y(k-1) \dots$   | $y(k-n+1)$   |

그림 3. GMDH의 입력과 출력.

시킨다. 그리고 (5) 식으로 구한 기준값  $R_s$ 보다 큰 값을 갖는 노드들을 제거한다. 그리고 제거되지 않은 변수  $z_{s,i}(k)$ 들을  $(s+1)$ 층의 입력으로 사용한다.

[단계 3] 최적성의 시험

이 단계에서는 훈련을 계속할지를 판별하기 위하여, 단계 2에서 계산된  $r_s$ 들의 최소값  $RMIN_s$ 를 전 단계의  $RMIN_{s-1}$ 과 비교하는 것이다. 그림 2와 같이  $RMIN_s < RMIN_{s-1}$ 이면 단계 1과 2를 반복하고,  $RMIN_s > RMIN_{s-1}$ 이면  $RMIN_{s-1}$ 에 해당하는  $z_{(s-1),i}(k)$ 를 출력의 추정치  $\hat{y}(k)$ 로 보고 모든 훈련을 끝내게 되는 것이다. 그러면 이  $z_{(s-1),i}(k)$ 는 Ivakhnenko 다항식[5]이 된다.

결국, 본 논문의 GMDH 알고리즘과 전형적인 GMDH 알고리즘의 차이점은, 단계 2에서 전형적인 GMDH 알고리즘은 일정한  $R_s$ 를 기준값으로 설정하지만 본 논문의 GMDH 알고리즘은 (5) 식으로부터  $R_s$ 를 구하므로 기준값이 점차 감소하게 되어 계산량의 과다를 방지할 수 있고 비교적 큰 에러를 갖는 노드들을 제거할 수 있다는 것이다. 또한, 다이내믹 시스템을 모델링하기 위하여 그림 3과 같이 입출력 데이터를 윈도우(window)화하는 적용 방법을 제안하였다.

4. 시뮬레이션 결과

수정된 GMDH 알고리즘의 성능을 분석하기 위한 컴퓨터 시뮬레이션을 위해 다음과 같은 비선형 시스템을 설정하였다.

$$y(k+1) = \frac{y(k)}{1+y^2(k)} + u^3(k) \tag{8}$$

그림 3에서 입출력 데이터의 갯수를 나타내는 m과 n을 각각 3과 4로 선정했을 때의 모델링결과를 그림 4에 나타내었다. 이때 (5)식의  $\delta$ 를 0.001로 선택하였

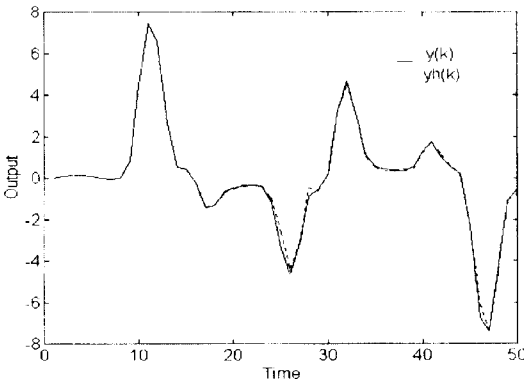


그림 4. m=3, n=4  $\delta=0.001$  일 때의 모델링 결과.

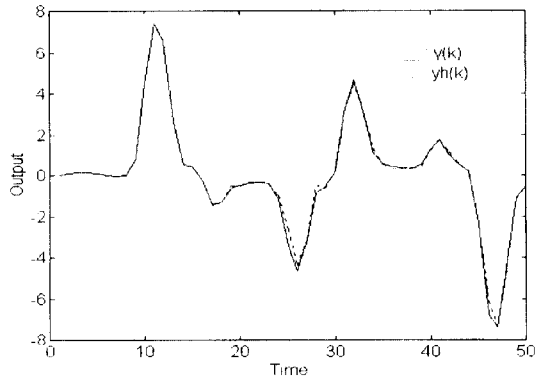


그림 5. m=3, n=4  $\delta=0.1$  일 때의 모델링 결과.

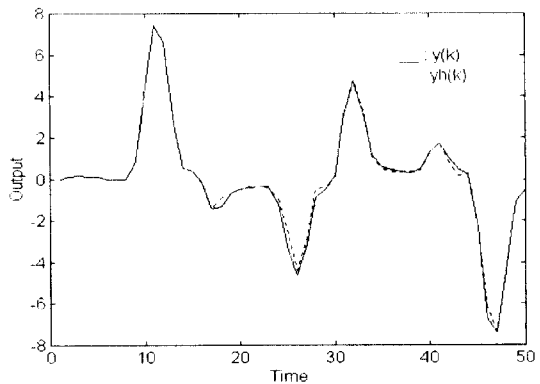


그림 6. m=3, n=4  $\delta=0.1$  일 때의 모델링 결과.

다. 본 논문에서 제안한 수정된 GMDH 알고리즘에 의해 모델링 출력  $\hat{y}(k)$  (그림에서는 yh(k))가 실제의 출력  $y(k)$ 를 잘 추종하고 있다는 것을 알 수 있다. 그림 5와 6은  $\delta$ 의 값을 증가시켜서 제거되는 노드를 줄임으로써 모델링 특성을 향상시킬 수 있는지의 여부를 확인하기 위하여  $\delta$ 를 각각 0.01과 0.1로 선정하였다. 이 세가지 경우에 대하여 계산량과 평균 에러를 표 1에 정리하였다. 여기서 계산량은 제거되지 않은 노드의 총갯수이며, 대부분의 연산이 노드에서 이루어지므로 계산량을 노드의 갯수로 하였다. 표 1에서 보는 바와 같이  $\delta$ 의 값을 0.01로 증가시켰을 때 계산량이 증가한 반면에 평균 에러는 0.0013 만큼 감소하

표 1.  $\delta$ 의 변화에 대한 계산량과 평균에러 (m=3, n=4)

| $\delta$ | 계산량    | 평균에러   |
|----------|--------|--------|
| 0.001    | 1,546  | 0.0836 |
| 0.01     | 5,049  | 0.0823 |
| 0.1      | 34,611 | 0.0961 |

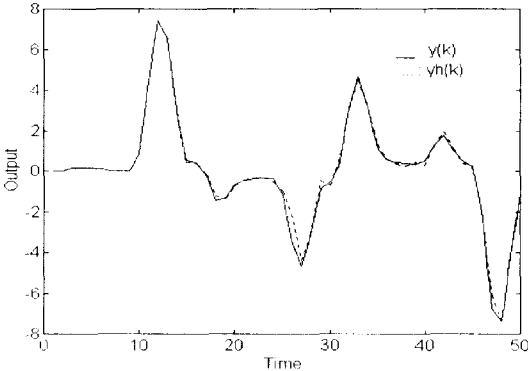


그림 7.  $m=2, n=3, \delta=0.01$  일 때의 모델링 결과.

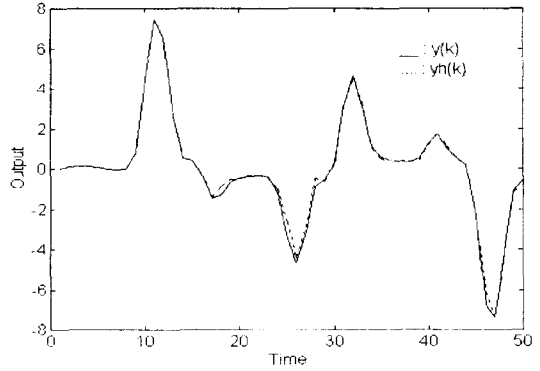


그림 9.  $R_s=0.1$  인 전형적인 GMDH의 모델링 결과. ( $m=3, n=4$ )

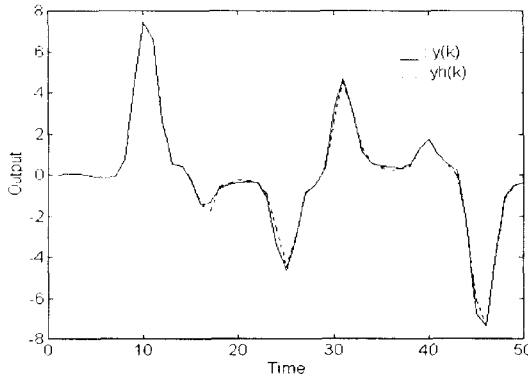


그림 8.  $m=4, n=5, \delta=0.01$  일 때의 모델링 결과.

었다.  $\delta=0.1$ 일 때는 도리어 평균 에러도 증가하고 계산량도 상당히 증가한다. 그 이유는 영향이 적은 입력과 노드가 포함되어 모델링 출력을 생성하기 때문이다. 따라서  $\delta$ 의 값을 선정할 때 계산량과 평균 에러를 고려하여 너무 크지 않은 값으로 선정해야 한다.

그림 7과 8은  $\delta=0.01$ 일 때  $m=2, n=3$ 인 경우와  $m=4, n=5$ 인 경우의 모델링 결과이다. 이 두 경우와 그림 5의 경우에 대하여 계산량과 평균 에러를 표 2에 정리하였다.  $m=2, n=3$ 인 경우에는 계산량이 적지만 평균 에러가 크고,  $m=4, n=5$ 인 경우에는 계산량도 많고 평균 에러도 줄지 않는다. 따라서 과거의 입력력 데이터들을 과다하게 포함시킬 필요가 없다는 것을 알 수 있다.

그림 9는  $m=3, n=4$ 일 때 (5)식의  $R_s$ 를 0.1로 고정시

킨 전형적인 GMDH 알고리즘에 의한 모델링 결과이다. 이때 계산량은 34,437이고 평균 에러는 0.1026이므로 본 논문에서 제안한 수정된 GMDH의 결과가 전형적인 GMDH의 결과보다 계산량도 적고 평균 에러도 작은 우수성을 나타내고 있다. 따라서 원하는 최종 출력과 노드 출력 간의 에러가 큰 노드들을 제거하지 않으면 도리어 모델링 출력의 에러가 더 커질 수도 있다는 것을 알 수 있다.

## 6. 결 론

복잡한 비선형 시스템의 입출력 데이터만을 가지고 그 시스템을 모델링하는 방법의 하나인 GMDH 알고리즘에서 데이터의 취사선택 기준을 순차적으로 감소시키는 방법을 제안하였고, 다이내믹 시스템의 모델링을 위하여 입출력 데이터를 인가하는 방법을 제안하였으며, 이 방법들을 컴퓨터 시뮬레이션을 통하여 입증하였다.

컴퓨터 시뮬레이션 결과에 의하면 전형적인 GMDH보다 본 논문의 수정된 GMDH가 더 우수한 특성을 갖는다는 것을 알 수 있었다. 입출력 데이터를 적게 인가했을 때는 계산량이 적지만 평균 에러가 커진다는 것을 알 수 있었고, 입출력 데이터를 과다하게 인가했을 때는 계산량도 많아지고 평균 에러도 도리어 커질 수 있다는 것을 알 수 있었다. 따라서 적절한 양의 입출력 데이터를 인가해야 한다.

표 2.  $m, n$ 의 변화에 대한 계산량과 평균에러 ( $\delta=0.01$ )

| $m$ | $n$ | 계산량    | 평균에러   |
|-----|-----|--------|--------|
| 2   | 3   | 2,691  | 0.1205 |
| 3   | 4   | 5,049  | 0.0823 |
| 4   | 5   | 35,426 | 0.0957 |

## 참고문헌

- [1] A. G. Ivakhnenko, "Polynomial theory of complex systems," *IEEE Trans. Syst. Man Cybern, SMC-1*, pp. 364-378, 1971.
- [2] A. G. Ivakhnenko, "Heuristic self-organization in

- problems of engineering cybernetics," *Automatica*, **6**, pp. 207-219, 1970.
- [3] R. A. Roy and J. Sherman, "A learning technique for Volterra series representation," *IEEE Trans. Automat. Contr.*, **AC-12**, pp. 761-764, 1967.
- [4] T. Nishikawa and S. Shimizu, "The characteristics of a biased estimator applied to the adaptive GMDH," *Mathl. Comput. Modelling*, **17**(1), pp. 37-48, 1993.
- [5] S. J. Farlow, *Self-Organizing Methods in Modelling: GMDH Type Algorithms*. New York: Marcel Decker Inc., 1984.
- [6] A. G. Ivakhnenko and A. A. Zholnarskiy, "Estimating the coefficients of polynomials in parametric GMDH algorithms by the improved instrumental variables method," *Soviet Automatic Control*, **25**(3), pp. 25-32, 1992.
- [7] T. Yoshimura, R. Kiyozumi, K. Nishino and T. Soeda, "Application of revised GMDH algorithms to the prediction of air pollutant concentrations," *Proc. 23rd IFAC World Congr.*, **XXIII**, pp. 13-17, Kyoto, Japan, Aug. 1981.
- [8] H. Akaike, "A new look at the statistical model identification," *IEEE Trans. Automat. Contr.* **AC-19**, pp. 716-723, 1974.
- [9] I. Hayashi and H. Tanaka, "The fuzzy GMDH algorithm by possibility models and its application," *Fuzzy Sets and Systems*, **36**, pp. 245-258, 1990.
- [10] P. Y. Chao, P. M. Ferreira and C. R. Liu, "Applications of GMDH-type modeling in manufacturing," *Journal of Manufacturing Systems*, **7**(3), pp. 241-253, 1988.
- [11] C. L. P. Chen and A. D. McAulay, "Robot Kinematics learning computations using polynomial neural networks," *Proc. 1991 IEEE Int. Conf. on Robotics and Automation*, pp. 2638-2643, Sacramento, California, Apr. 1991.
- [12] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Networks*, **1**(1), pp. 4-27, Mar. 1990.

---

**홍연찬 (Yeon Chan Hong) 정회원**  
제 7 권 제 3 호 참조

---



**엄상수 (Sang Su Ohm) 학생회원**  
1975년 1월 25일생  
1997년 2월 : 청주대학교 전자공학과 졸업  
1997년 ~ 현재 : 인천대학교 대학원 전자공학과 석사과정  
관심분야 : GMDH, 신경망, 문자인식

---