확장된 블록 펄스 연산 행렬을 이용한 쌍일차계의 시변 파라미터 추정

論 文
50D - 8 - 4

Identification of Time-varying Parameters of Bilinear Systems via Extended Block Pulse Operational Matrices

金 泰 勳* · 金 珍 泰** · 李 勝*** · 李 明 珪§ · 安 斗 守§§
(Tai-Hoon Kim · Jin-Tae Kim · Seung Lee · Myoung-Kyu Lee · Doo-Soo Ahn)

Abstract - This paper considers the problem of identifying the time-varying parameters of Bilinear systems. The Parameters, in this paper, are identified by using the EBPOMs(Extended Block Pulse Operational Matrices) which can reduce the burden of operation and the volume of error caused by matrices multiplication.

Key Words : EBPOMs, bilinear systems, time-varying parameters

1. 서 론

연속 시간 모델 식별을 위하여 다양한 직교함수들(월쉬함수, 블록 펄스 함수 등)과 직교 다항식들(Chebyshev 다항식, Legendre 다항식, Laguerre 다항식 등)이 제안되었다[1-4]. 이러한 기법들의 공통적인 목적은 연속시간 시스템들의 미분방정식 모델에 나타나는 입출력 신호들의 시간 미분치에 대한 직접측정을 피하는 것이며, 이 신호들을 근사화된 급수로 전개하고 적분 연산 행렬을 적용하면 선형 대수방정식 집합들을 얻을 수 있고, 또한 그 미분 방정식들의 파라미터들을 추정할 수 있다. 이러한 기법들을 사용하여 연속시간 모델식별에 관한 만족스런 결과들을 얻을 수 있다.

시스템 공학(System Science)에 대한 직교함수들의 적용은 Corrington에 의해 제안되었다[5]. Corrington은 상미분 방정식을 풀기 위해 월쉬 연산 행렬들을 구성하였는데, 후에 Chen과 Hsiao[6-7]가 그 중 하나를 월쉬 연산 행렬로 채택한 것이 일반화되어 사용되었다. 그 후 만일 적분 연산 행렬이 월쉬 영역으로부터 블록 펄스 영역으로 전환된다면 계산상의 복잡함이 현저히 줄어든다는 것이 밝혀졌다[8-9]. 또한 다중 적분으로 인한 적분 연산 행렬의 거듭되는 곱으로 인하여 발생하는 오차를 줄이기 위하여 개선된 블록 펄스 적분 연산 행렬(GBPOMs : Generalized Block Pulse Operational Matrices)인 $\boldsymbol{P}_i$(단, $i = 0, 1, 2, \cdots$)가 도입되었고[10], 시변 요소를 포함하고 있을 경우 이것을 임의의 함수로 치환하여 연산을 간단히 할 수 있도록 하는 방법이 소개되었다[11]. 하지만 이 경우에 있어서, 치환된 함수도 또한 블록 펄스 급수 전개를 하여야 하므로 연산의 부담은 줄어들지 않게 된다.

본 논문에서는, 시변 요소를 포함하는 경우에 이 시변 요소의 특성을 아예 처음부터 적분 연산 행렬 안에 포함되도록 함으로써, 시변 요소들에 대하여 따로 연산을 수행하여야 하는 과정을 생략할 수 있는 확장된 블록 펄스 적분 연산 행렬(EBPOMs : Extended Block Pulse Operational Matrices)을 제안하고, 이를 이용하면 시간 요소를 포함하는 다중 적분 형식의 시스템을 해석할 때 연산이 간단해지고 정확성이 향상될 수 있음을 보였다. 또한 이를 비선형 시스템의 일종인 쌍일차계(BLS : Bilinear System)에 적용하여서, 시스템의 시변 파라미터들을 추정하였다.

2. 확장된 블록 펄스 적분 연산 행렬

다음과 같이 시변 요소를 갖는 형태로 주어진 적분을 고려해본다.

$$\int_0^t \cdots \int_0^t t^j f(t)\, dt \cdots dt \tag{2.1}$$

(단, $i, j = 0, 1, 2, \cdots$)

이러한 형태를 갖는 적분들의 블록 펄스 급수 전개는 선형 시변 시스템들의 문제를 풀 때 유용하다. 사실 이런 형태의 적분은 $r(t) = t^j$로 하여 일반적인 블록 펄스 적분 연산 행렬(CBPOMs : Common block pulse operational matrices)인 $\boldsymbol{P}$를 써서

* 正 會 員 : 프라임테크 선임연구원
** 正 會 員 : 成均館大 電氣電子및컴퓨터工學部 博士課程
*** 正 會 員 : 大林大 自動化 시스템 工學科 敎授 · 工博
§ 正 會 員 : 慶星大 電氣電子및컴퓨터工學部 敎授 · 工博
§§ 正 會 員 : 成均館大 電氣電子및컴퓨터工學部 敎授 · 工博
接受日字 : 2001年 4月 12日
最終完了 : 2001年 7月 14日

$$\int_0^t \cdots \int_0^t t^j f(t)\, dt \cdots dt \doteq \boldsymbol{E}^T \boldsymbol{D}_f \boldsymbol{D}_r \boldsymbol{P}^j \boldsymbol{\Psi}_{(m)}(t) \tag{i}$$

로 풀 수 있으며, 혹은 개선된 블록 펄스 적분 연산 행렬인 $\boldsymbol{P}_i$를 써서

$$\int_0^t \cdots \int_0^t t^j f(t)\, dt \cdots dt \doteq \boldsymbol{E}^T \boldsymbol{D}_f \boldsymbol{D}_r \boldsymbol{P}_i \boldsymbol{\Psi}_{(m)}(t) \tag{ii}$$

$$\left.\begin{array}{l} \text{단,}\quad \boldsymbol{E}=[1\ 1\ \cdots\ 1]^T,\quad \boldsymbol{f}^T=[f_1\ f_2\ \cdots\ f_m], \\ \boldsymbol{r}^T=[r_1\ r_2\ \cdots\ r_m],\quad \boldsymbol{D}_f=diag(f_1\ f_2\ \cdots\ f_m), \\ \boldsymbol{D}_r=diag(r_1\ r_2\ \cdots\ r_m) \end{array}\right)$$

로 풀 수 있다. 그러나 위 두 식에서는 함수들의 곱과 적분이 분리되어 블록 펄스 급수로 근사화된다. 이러한 분리된 계산으로 인하여 계산량이 많아지고 더 많은 계산상의 오차가 발생하게 된다. 하지만 확장된 블록 펄스 적분 연산 행렬을 사용하게 되면 이러한 단점을 피하여서 식 (2.1)을 한 단계만에 블록 펄스 급수로 전개할 수 있게 된다.

먼저 식(2.1)에서, $f(t)$가 블록 펄스 함수 $\psi_k(t)$로 표시되는($f(t)=\psi_k(t)$) 특별한 경우를 생각해 본다.

$$\int_0^t \cdots \int_0^t t^j \psi_k(t)\, dt \cdots dt \tag{2.2}$$

단, $k = 1, 2, 3, \cdots, m$

블럭 펄스 함수는 지연을 갖는 단위 계단함수로 표현할 수 있으므로 다음처럼 나타낼 수 있다.

$$\begin{aligned} t^j\psi_k(t) &= t^j u(t-(k-1)h) - t^j u(t-kh) \\ &= \{(t-(k-1)h)+(k-1)h\}^j u(t-(k-1)h) \\ &\quad -((t-kh)+kh)^j u(t-kh) \end{aligned} \tag{2.3}$$

단, $h=\dfrac{T}{m}$

앞의 식(2.3)에 이항정리(binomial theorem)를 적용하면 식(2.4)에서와 같이

$$\begin{aligned} t^j\psi_k(t) &= \sum_{q=0}^{j}\binom{j}{q}((k-1)h)^{j-q} \\ &\quad \times (t-(k-1)h)^q u(t-(k-1)h) \\ &\quad -\sum_{q=0}^{j}\binom{j}{q}(kh)^{j-q}(t-kh)^q u(t-kh) \end{aligned} \tag{2.4}$$

이 되고, 다시 이 식에 라플라스 변환을 행하면

$$\begin{aligned} L\{t^j\psi_k(t)\} &= \sum_{q=0}^{j}\binom{j}{q}((k-1)h)^{j-q} \\ &\quad \times \frac{q!}{s^{q+1}}\exp\{-(k-1)hs\} \\ &\quad -\sum_{q=0}^{j}\binom{j}{q}(kh)^{j-q}\frac{q!}{s^{q+1}}\exp\{-khs\} \end{aligned} \tag{2.5}$$

과 같이 된다. 위 식은 라플라스 변환의 적분 특성에 의하여

$$\begin{aligned} &L\left\{\int_0^t \cdots \int_0^t t^j \psi_k(t)\, dt \cdots dt\right\} \\ &= \sum_{q=0}^{j}\binom{j}{q}((k-1)h)^{j-q}\frac{!}{s^{j+q+1}}\exp\{-(k-1)hs\} \\ &\quad -\sum_{q=0}^{j}\binom{j}{q}(kh)^{j-q}\frac{!}{s^{j+q+1}}\exp\{-khs\} \end{aligned} \tag{2.6}$$

와 같이 표현될 수 있다. 식(2.6)에 다시 라플라스 역변환을 행하면

$$\begin{aligned} &\int_0^t \cdots \int_0^t t^j \psi_k(t)\, dt \cdots dt \\ &= \sum_{q=0}^{j}\binom{j}{q}((k-1)h)^{j-q} \\ &\qquad \frac{q!}{(j+q)!}(t-(k-1)h)^{j+q}u(t-(k-1)h) \\ &\quad -\sum_{q=0}^{j}\binom{j}{q}(kh)^{j-q}\frac{q!}{(j+q)!}(t-kh)^{j+q}u(t-kh) \\ &= [c_{i,j,k,1}\ \ c_{i,j,k,2}\ \ \cdots\ \ c_{i,j,k,m}]^T\Psi_{(m)}(t) \end{aligned} \tag{2.7}$$

으로 나타낼 수 있다. 이때, 행렬의 원소 $c_{i,j,k,l}(l=1,2,\cdots,m)$은 다음 식으로부터 계산할 수 있다.

$$\begin{aligned} c_{i,j,k,l} &= \frac{1}{h}\int_0^t\cdots\int_0^t t^j\psi_k(t)\,dt\cdots dt \quad ((l-1)h\le t\le lh) \\ &= \begin{cases} 0, & l<k \\ h^{i+j}\sum_{q=0}^{j}\binom{j}{q}\frac{q!}{(i+q+1)!}(k-1)^{j-q}, & l=k \\ h^{i+j}\sum_{q=0}^{j}\binom{j}{q}\frac{q!}{(i+q+1)!}\{(k-1)^{j-q}[(l-k+1)^{i+q+1} \\ \quad -(l-k)^{i+q+1}]-k^{j-q}[(l-k)^{i+q+1}-(l-k-1)^{i+q+1}]\}, & l>k \end{cases} \end{aligned} \tag{2.8}$$

식 (2.7)을 이용하여, m개의 블록 펄스 함수 적분의 블록 펄스 급수를 다음처럼 하나의 행렬 $\boldsymbol{P}_{i,j}$로 표현할 수 있다.

$$\int_0^t\cdots\int_0^t t^j\psi_k(t)\,dt\cdots dt \doteq \boldsymbol{P}_{i,j}\Psi_{(m)}(t) \tag{2.9}$$

$$\boldsymbol{P}_{i,j}=\frac{j!\,h^{i+j}}{(i+j+1)!}\begin{pmatrix} p_{i,j,1,1} & p_{i,j,1,2} & p_{i,j,1,3} & \cdots & p_{i,j,1,m} \\ 0 & p_{i,j,2,2} & p_{i,j,2,3} & \cdots & p_{i,j,2,m} \\ 0 & 0 & p_{i,j,3,3} & \cdots & p_{i,j,3,m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & p_{i,j,m,m} \end{pmatrix} \tag{2.10}$$

$$p_{i,j,k,l} = \begin{cases} 0\ , & l<k \\ \sum_{q=0}^{j} \frac{(i+j+1)!}{(j-q)!(i+q+1)!}(k-1)^{j-q}\ , & l=k \\ \sum_{q=0}^{j} \frac{(i+j+1)!}{(j-q)!(i+q+1)!}\{(k-1)^{j-q}[(l-k+1)^{i+q+1} \\ \quad -(l-k)^{i+q+1}]-k^{j-q}[(l-k)^{j+q+1}-(l-k-1)^{j+q+1}]\}\ , & l>k \end{cases}$$

식(2.10)의 행렬 $P_{i,j}$는 피적분항에 t^j항을 가진 함수의 i번 적분과 관련된 블록 펄스 적분 연산 행렬, 즉 t^j와 관련된 i번째 확장된 적분 연산 행렬로 정의된다. 행렬 $P_{i,j}$로부터 개선된 블록 펄스 적분 연산 행렬[10]로 표현되는 P_i는 $P_i = P_{i,0}$(단, i=1,2,…)가 되어 $P_{i,j}$의 특수한 경우라는 것을 알 수 있다. 또한, 일반적인 블록 펄스 적분 연산 행렬은 $P = P_{1,0}$가 됨을 알 수 있다.

지금까지의 결과로부터 식(2.1)의 블록 펄스 급수 전개는 다음 식처럼 한 단계로 얻어질 수 있음을 알 수 있다.

$$\int_0^t \cdots \int_0^t t^j f(t)\,dt \cdots dt \doteq f^T P_{i,j} \Psi_{(m)}(t) \qquad \text{(iii)}$$

앞의 식 (i), (ii) 그리고 (iii)으로부터 확장된 블록 펄스 적분 연산 행렬을 사용했을 때의 장점을 다음처럼 요약할 수 있다.

[장점 1] 시변 요소인 t를 따로 계산하지 않아도 되므로 연산량이 감소하게 된다(식 (i), (ii)와 식 (iii)의 비교로 알 수 있다).

[장점 2] 다중 적분을 행할 때 연산 행렬의 곱을 피할 수 있게 되므로 오차의 누적을 줄일 수 있다.

위에서 기술한 것을 다음의 예제를 통해 확인할 수 있다.

<예제> 구간이 t∈[0,5)이고 전개 항수가 m=10일 때, 다음의 식(2.11)로 주어진 적분

$$g(t) = \int_0^t \int_0^t t^2 e^{-t} dt dt \qquad (2.11)$$

의 블록 펄스 급수를 일반적인 블록 펄스 적분 연산 행렬과 확장된 블록 펄스 적분 연산 행렬로 각각 구해본다.

(1)일반적인 블록 펄스 적분 연산 행렬을 이용한 방법

먼저 $f(t) = \exp(-t)$, $r(t) = t^2$로 놓고, 각각의 블록 펄스 급수벡터인 f와 r을 구한다. 그 다음 f와 r을 이용하여 행렬 D_f와 D_r을 구하여 곱하고, 행렬 P를 제곱하여 다음 연산을 행한다.

$$\begin{aligned} g(t) &\doteq E^T D_f D_r P^2 \Psi_{(m)}(t) \\ &\doteq [0.0063\ \ 0.0441\ \ 0.1563\ \ 0.3826\ \ 0.7437\ \ 1.2408 \\ &\quad 1.8620\ \ 2.5895\ \ 3.4029\ \ 4.2837\ ]\ \Psi_{(m)}(t) \end{aligned} \qquad (2.12)$$

(2)확장된 블록 펄스 적분 연산 행렬을 이용한 방법

이 방법을 쓰면 앞에서 행하였던 연산 과정들이 다음처럼 한 단계로 간단해 질 수 있다.

$$\begin{aligned} g(t) &\doteq f^T P_{2,2}\ \Psi_{(m)}(t) \\ &\doteq [0.0008\ \ 0.0207\ \ 0.1084\ \ 0.3096\ \ 0.6466\ \ 1.1210 \\ &\quad 1.7205\ \ 2.4267\ \ 3.2192\ \ 4.0792\ ]\ \Psi_{(m)}(t) \end{aligned} \qquad (2.13)$$

이때, 주어진 함수의 실제 적분값은 다음의 식

$$\begin{aligned} g(t) &\doteq -6+2t+(6+4t+t^2)e^{-t} \\ &\doteq [0.0019\ \ 0.0253\ \ 0.1131\ \ 0.3078\ \ 0.6312\ \ 1.0853 \\ &\quad 1.6591\ \ 2.3349\ \ 3.0938\ \ 3.9176\ ]\ \Psi_{(m)}(t) \end{aligned} \qquad (2.14)$$

과 같이 되고, 각 경우의 결과를 비교해 보면 $P_{i,j}$를 사용함으로써 연산 과정이 줄어들고 정확성이 향상됨을 알 수 있다. 다음 그림 2.1은 각각의 경우의 결과를 도시하여 비교한 것이다.

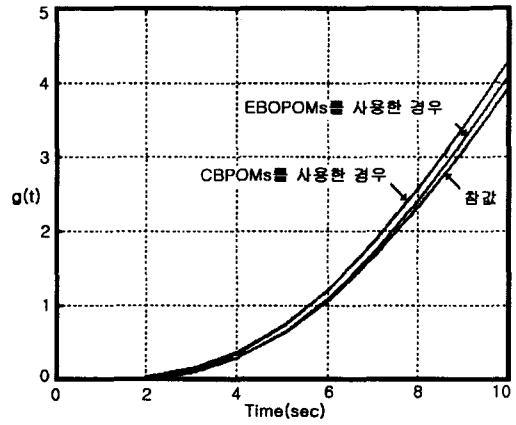


그림 2.1 예제의 결과

Fig 2.1 The result of example

3. 쌍일차계의 시변 파라미터 추정

3.1 쌍일차계

쌍일차계는 가장 간단한 형태의 비선형계이다. 비선형계의 특성은 상태들만에 대해서, 또는 제어 입력만에 대해서는 선형성을 갖게되지만, 상태와 제어 입력에 대해 동시에 선형성을 갖지는 못한다는 것이다. 다시 말해서 상태와 제어 입력의 곱을 포함하고 있다는 것이고, 그럼으로 인하여 중첩의 원리가적용될 수 없게 된다는 것이다.

분명히 선형계는 쌍일차계의 특별한 경우이고, 이것은 다음의 가장 간단한 예로써 쉽게 알 수 있다. 시불변 쌍일차계는 다음과 같은 형태를 갖는데,

$$\frac{dx(t)}{dt} = Ax(t) + Bu(t) + \sum_{k=1}^{m} N_k u_k(t) x(t) \qquad (3.1)$$

(단, $x \in R^m$, $N_k \in R^{m \times n}$)

여기서 마지막 항을 제거하면 선형계가 됨을 알 수 있다. 식(3.1)의 마지막 항은 때때로 간단히 $N(u,x)$나 $Nu \cdot x$와 같이 표시되기도 하고, 이에 해당하는 쌍일차계는 그림 3.1에 보였으며, 출력은 $y(t) = Cx(t)$로 나타낸다.

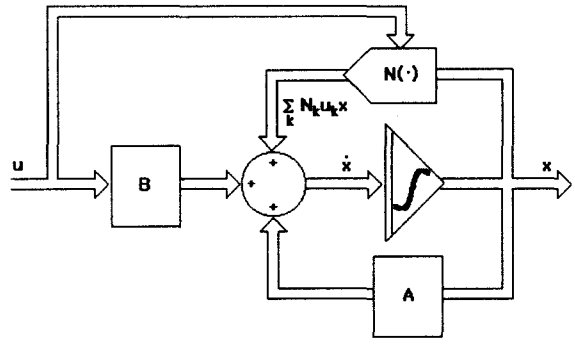


그림 3.1 쌍일차계의 상태도
Fig. 3.1 The state diagram of bilinear system

실제의 시스템들 속에서(예를 들어, 핵 융합로 등) 쌍일차계는 고도의 비선형계속에 서로 결합되어 있다. 그러한 시스템의 결합은 선형이 될 수도 있고 비선형이 될 수도 있다.

상태와 입력이 곱해진 구조로 되어 있는 비선형계에 속하는 쌍일차계는 선형계의 특성과 비선형계의 특성을 동시에 가지고 있다. 실제계는 거의가 비선형계인데, 이것은 취급이 어렵기 때문에 선형계에 가까운 쌍일차계를 이용하여 비선형계를 다루고 있다. 비선형계로부터 쌍일차계를 가지는 상태 방정식을 유도할 수 있는데, 일반적으로 다음 식(3.2)와 같이 표시되는 비선형계

$$\dot{x} = f(x(t), u(t), t) \tag{3.2}$$

(단, $x \in R^n$, $u \in R^m$, $t \in (t_0, \infty)$)

에 Tayler 급수를 적용시키면 식(3.3)이 되고,

$$\triangle \dot{x} = \frac{\partial f}{\partial x} \mid \triangle x + \frac{\partial f}{\partial x} \mid \triangle u + \frac{\partial f}{\partial x \partial u} \mid \triangle x \triangle u + H.O.T. \tag{3.3}$$

여기서 $H.O.T.$을 무시하면 쌍일차계를 유도할 수 있다.

3.2 쌍일차계의 시변 파라미터 추정

쌍일차계는 가장 간단한 형태의 비선형 시스템이고, 다음의 식 (3.4)와 같은 형식을 갖는다. 이 때, 주어진 시스템이 안정하고 모든 상태의 측정이 가능하다고 가정하고 시변 파라미터를 추정한다.

$$\dot{x}(t) = A(t)\,x(t) + B(t)\,u(t) + \sum_{k=1}^{q} N_k(t)\,u_k(t)\,x(t)$$

$$\left(\text{단, } x \in R^n,\ u \in R^q,\ N_k(k=1,2,\ldots,q) \in R^{n\times n},\ A \in R^{n\times n},\ B \in R^{n\times q} \right) \tag{3.4}$$

행렬 $A(t)$, $B(t)$ 그리고 $N_k(t)$등은 시변 요소를 포함하고 있으며, 다음과 같이 표현될 수 있다.

$$A(t) = A_0 + A_1 t + A_2 t^2 + \ldots + A_\sigma t^\sigma \tag{3.5-1}$$

$$B(t) = B_0 + B_1 t + B_2 t^2 + \ldots + B_s t^s \tag{3.5-2}$$

$$N_1(t) = N_{10} + N_{11} t + N_{12} t^2 + \ldots + N_{1l_1} t^{l_1} \tag{3.5-3}$$

$$\vdots$$

$$N_q(t) = N_{q0} + N_{q1} t + N_{q2} t^2 + \ldots + N_{ql_q} t^{l_q} \tag{3.5-4}$$

식 (3.4)의 i번째 행을 쓰면

$$\begin{aligned}\dot{x}_i(t) = {} & a_i{}^0 x(t) + a_i{}^1 t x(t) + \ldots + a_i{}^\sigma t^\sigma x(t) \\ & + b_i{}^0 u(t) + b_i{}^1 t u(t) + \ldots + b_i{}^s t^s u(t) \\ & + n_{1i}{}^0 u_1(t)\,x(t) + n_{1i}{}^1 t u_1(t)\,x(t) + \ldots \\ & \qquad + n_{1i}{}^{l_1} t^{l_1} u_1(t)\,x(t) \\ & + \ldots \\ & + n_{qi}{}^0 u_q(t)\,x(t) + n_{qi}{}^1 t u_q(t)\,x(t) + \ldots \\ & \qquad + n_{qi}{}^{l_q} t^{l_q} u_q(t)\,x(t)\end{aligned} \tag{3.6}$$

과 같이 될 수 있고, a_i, b_i와 n_{qi}는 각각 행렬 A, B, 그리고 N_q 의 i 번째 행벡터와 같다. 위 식은 다시 다음의 식 (3.7)과 같이

$$\begin{aligned}\dot{x}_i(t) = {} & a_i{}^0 x(t) + a_i{}^1 t x(t) + \ldots + a_i{}^\sigma t^\sigma x(t) \\ & + b_i{}^0 u(t) + b_i{}^1 t u(t) + \ldots + b_i{}^s t^s u(t) \\ & + n_{1i}{}^0 \begin{bmatrix} u_1(t)\,x_1(t) \\ u_1(t)\,x_2(t) \\ \vdots \\ u_1(t)\,x_n(t) \end{bmatrix} + \ldots + n_{1i}{}^{l_1} t^{l_1} \begin{bmatrix} u_1(t)\,x_1(t) \\ u_1(t)\,x_2(t) \\ \vdots \\ u_1(t)\,x_n(t) \end{bmatrix} \\ & + \ldots \\ & + n_{qi}{}^0 \begin{bmatrix} u_q(t)\,x_1(t) \\ u_q(t)\,x_2(t) \\ \vdots \\ u_q(t)\,x_n(t) \end{bmatrix} + \ldots + n_{qi}{}^{l_q} t^{l_q} \begin{bmatrix} u_q(t)\,x_1(t) \\ u_q(t)\,x_2(t) \\ \vdots \\ u_q(t)\,x_n(t) \end{bmatrix}\end{aligned} \tag{3.7}$$

로 표현하여 t=0에서 t=T 까지 적분한 후 블록 펄스 급수로 전개한다.

$x(t)$ 와 $u(t)$를 블록 펄스 급수 전개한 것을 각각

$$x(t) = \Gamma^T \Psi_{(m)}(t) \tag{3.8-1}$$

$$u(t) = \Delta^T \Psi_{(m)}(t) \tag{3.8-2}$$

로 놓으면 $u_i(t)x_j(t)$는

$$u_i(t)x_j(t) = \Delta_i^T D_{\gamma_j} \Psi_{(m)}(t) \tag{3.9}$$

(단, γ_j는 Γ의 j번째 행)

로 쓸 수 있다. 그러므로 적분의 결과는 다음과 같게 된다.

$$\begin{aligned}
&\Psi_{(m)}^T(t)[\gamma_i - X_{0,i}] \\
&= \Psi_{(m)}^T(t) P_{1,0}^T \Gamma (a_i^0)^T + \dots + \Psi_{(m)}^T(t) P_{1,\sigma}^T \Gamma (a_i^\sigma)^T \\
&+ \Psi_{(m)}^T(t) P_{1,0}^T \Delta (b_i^0)^T + \dots + \Psi_{(m)}^T(t) P_{1,s}^T \Delta (b_i^s)^T \\
&+ \Psi_{(m)}^T(t)[P_{1,0}^T D_{\gamma_1}^T \Delta_1 \vdots \dots \vdots P_{1,0}^T D_{\gamma_n}^T \Delta_1](n_{1i}^0)^T + \\
&\dots + \Psi_{(m)}^T(t)[P_{1,l_1}^T D_{\gamma_1}^T \Delta_1 \vdots \dots \vdots P_{1,l_1}^T D_{\gamma_n}^T \Delta_1](n_{1i}^{l_1})^T \\
&+ \dots \\
&+ \Psi_{(m)}^T(t)[P_{1,0}^T D_{\gamma_1}^T \Delta_q \vdots \dots \vdots P_{1,0}^T D_{\gamma_n}^T \Delta_q](n_{qi}^0)^T + \\
&\dots + \Psi_{(m)}^T(t)[P_{1,l_q}^T D_{\gamma_1}^T \Delta_q \vdots \dots \vdots P_{1,l_q}^T D_{\gamma_n}^T \Delta_q](n_{qi}^{l_q})^T
\end{aligned} \tag{3.10}$$

단, $X_{0,i}$는 $x_i(t)$의 초기치의 블럭 펄스 급수 벡터,

Δ_q는 Δ의 q번째 행

이것을 간략하게 다시 쓰면

$$\begin{aligned}
&\Psi_{(m)}^T(t)[\gamma_i - X_{0,i}] \\
&= \Psi_{(m)}^T(t) P_{1,0}^T \Gamma (a_i^0)^T + \dots + \Psi_{(m)}^T(t) P_{1,\sigma}^T \Gamma (a_i^\sigma)^T \\
&+ \Psi_{(m)}^T(t) P_{1,0}^T \Delta (b_i^0)^T + \dots + \Psi_{(m)}^T(t) P_{1,s}^T \Delta (b_i^s)^T \\
&+ \Psi_{(m)}^T(t) F_{l_1,0,1} (n_{1i}^0)^T + \dots + \Psi_{(m)}^T(t) F_{l_1,l_1,1} (n_{1i}^{l_1})^T \\
&+ \dots \\
&+ \Psi_{(m)}^T(t) F_{l_q,0,q} (n_{qi}^0)^T + \dots + \Psi_{(m)}^T(t) F_{l_q,l_q,q} (n_{qi}^{l_q})^T
\end{aligned} \tag{3.11}$$

(단, $F_{l_q,l_q,q} = [P_{1,l_q}^T D_{\gamma_1}^T \Delta_q \vdots \dots \vdots P_{1,l_q}^T D_{\gamma_n}^T \Delta_q]$)

과 같이 될 수 있다. 식 (3.11)을 행렬 형태로 표현하면 다음 식 (3.12)가 된다.

$$\begin{aligned}
&\Psi_{(m)}^T(t)[\gamma_i - X_{0,i}] \\
&= \Psi_{(m)}^T(t)[P_{1,0}^T \Gamma \vdots \dots \vdots P_{1,\sigma}^T \Gamma \vdots P_{1,0}^T \Delta \vdots \dots \vdots P_{1,s}^T \Delta \vdots \\
&\quad F_{l_1,0,1} \vdots \dots \vdots F_{l_1,l_1,1} \vdots F_{l_q,0,q} \vdots \dots \vdots F_{l_q,l_q,q}] \\
&\times [a_i^0 \vdots \dots \vdots a_i^\sigma \vdots b_i^0 \vdots \dots \vdots b_i^s \vdots \\
&\quad n_{1i}^0 \vdots \dots \vdots n_{1i}^{l_1} \vdots n_{qi}^0 \vdots \dots \vdots n_{qi}^{l_q}]^T
\end{aligned} \tag{3.12}$$

위 식은 다시 간단한 형태로 표현될 수 있다.

$$\Psi_{(m)}^T(t)[\gamma_i - X_{0,i}] = \Psi_{(m)}^T(t) H \theta_i \tag{3.13}$$

$$\begin{cases} \text{단, } H = [P_{1,0}^T \Gamma \vdots \dots \vdots P_{1,\sigma}^T \Gamma \vdots P_{1,0}^T \Delta \vdots \dots \vdots P_{1,s}^T \Delta \vdots \\ \qquad F_{l_1,0,1} \vdots \dots \vdots F_{l_1,l_1,1} \vdots F_{l_q,0,q} \vdots \dots \vdots F_{l_q,l_q,q}], \end{cases}$$

$$\begin{cases} \text{단, } \theta_i = [a_i^0 \vdots \dots \vdots a_i^\sigma \vdots b_i^0 \vdots \dots \vdots b_i^s \vdots \\ \qquad n_{1i}^0 \vdots \dots \vdots n_{1i}^{l_1} \vdots n_{qi}^0 \vdots \dots \vdots n_{qi}^{l_q}]^T \end{cases}$$

여기서 $H \in R^{m \times \alpha}$, $\theta_i \in R^{\alpha \times 1}$ 이 되므로, $m \geq \alpha$의 관계로 만족하도록 블록 펄스 함수의 전개 항수 m을 결정한다면 행렬 $P_{i,j}$의 모든 열은 상삼각 행렬 특성을 유지하여 항상 선형 독립이 되기 때문에 θ_i는 최소자승법에 의하여 다음과 같이 구할 수 있다.[12]

$$\theta_i = (H^T H)^{-1} H^T [\gamma_i - X_{0,i}] \tag{3.14}$$

4. 시뮬레이션

다음과 같은 식으로 주어지고 초기값을 '영'으로 하는 쌍일차계를 가정한다.

$$\dot{x}(t) = B(t)u(t) + N_1(t)u(t)x(t) \tag{4.1}$$

단, $x \in R^{2\times 1}$, $u \in R^1$, $B \in R^{2\times 1}$, $N_1 \in R^{2\times 2}$

주어진 식의 계수 행렬들이 각각 다음과 같다고 하면

$$\begin{aligned}
B(t) &= \begin{bmatrix} b_{11} + b_{12}t + b_{13}t^2 \\ b_2 t \end{bmatrix} = \begin{bmatrix} b_{11} \\ 0 \end{bmatrix} + \begin{bmatrix} b_{12} \\ b_2 \end{bmatrix} t + \begin{bmatrix} b_{13} \\ 0 \end{bmatrix} t^2 \\
&= B_0 + B_1 t + B_2 t^2
\end{aligned} \tag{4.2-1}$$

$$N_1(t) = \begin{bmatrix} 0 & n_{112}t^2 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & n_{112} \\ 0 & 0 \end{bmatrix} t^2 = N_{12} t^2 \tag{4.2-2}$$

앞의 식 (3.6)으로부터

$$\dot{x}_1 = b_{11}u(t) + b_{12}tu(t) + b_{13}t^2u(t) + n_{11}^2 t^2 \begin{bmatrix} u(t)x_1(t) \\ u(t)x_2(t) \end{bmatrix} \tag{4.3-1}$$

$$\dot{x}_2 = b_2 tu(t) \tag{4.3-2}$$

이 됨을 알 수 있다. 앞 식들을 블록 펄스 함수로 전개하면 식 (3.11)로부터 다음의 식

$$\boldsymbol{\Psi}_{(m)}{}^{T}(t)\ \boldsymbol{r}_1$$
$$= \boldsymbol{\Psi}_{(m)}{}^{T}(t)\ \boldsymbol{P}_{1.0}{}^{T}\Delta(b_{11}) + \boldsymbol{\Psi}_{(m)}{}^{T}(t)\ \boldsymbol{P}_{1.1}{}^{T}\Delta(b_{12})$$
$$+ \boldsymbol{\Psi}_{(m)}{}^{T}(t)\ \boldsymbol{P}_{1.2}{}^{T}\Delta(b_{13}) + \boldsymbol{\Psi}_{(m)}{}^{T}(t)\ \boldsymbol{P}_{1.2}{}^{T}\boldsymbol{D}_{r_2}\Delta(n_{112}) \quad (4.4\text{-}1)$$

$$\boldsymbol{\Psi}_{(m)}{}^{T}(t)\ \boldsymbol{r}_2 = \boldsymbol{\Psi}_{(m)}{}^{T}(t)\ \boldsymbol{P}_{1.1}{}^{T}\Delta(b_2) \quad (4.4\text{-}2)$$

이 되고, 이것을 행렬 형태로 표시하면 다음과 같다.

$$\boldsymbol{\Psi}_{(m)}{}^{T}(t)\ \boldsymbol{r}_1 = \boldsymbol{\Psi}_{(m)}{}^{T}(t)\ \boldsymbol{H}_1\ \boldsymbol{\theta}_1 \quad (4.5\text{-}1)$$
$$\boldsymbol{\Psi}_{(m)}{}^{T}(t)\ \boldsymbol{r}_2 = \boldsymbol{\Psi}_{(m)}{}^{T}(t)\ \boldsymbol{H}_2\ \boldsymbol{\theta}_2 \quad (4.5\text{-}2)$$

여기서 $\boldsymbol{H}_1$, $\boldsymbol{H}_2$, $\boldsymbol{\theta}_1$, $\boldsymbol{\theta}_2$은 다음을 만족한다.

$$\boldsymbol{H}_1 = \left[\ \boldsymbol{P}_{1.0}{}^{T}\Delta \vdots\ \boldsymbol{P}_{1.1}{}^{T}\Delta \vdots\ \boldsymbol{P}_{1.2}{}^{T}\Delta \vdots\ \boldsymbol{P}_{1.2}{}^{T}\boldsymbol{D}_{r_2}\Delta\ \right]$$
$$\boldsymbol{H}_1 = \boldsymbol{P}_{1.1}{}^{T}\Delta$$
$$\therefore\ \boldsymbol{H}_1 \in \boldsymbol{R}^{m\times 4},\quad \boldsymbol{H}_2 \in \boldsymbol{R}^{m\times 1}$$
$$\boldsymbol{\theta}_1{}^{T} = [\,b_{11} : b_{12} : b_{13} : n_{112}\,],\quad \boldsymbol{\theta}_2{}^{T} = b_2$$
$$\therefore\ \boldsymbol{\theta}_1 \in \boldsymbol{R}^4,\quad \boldsymbol{\theta}_2 \in \boldsymbol{R}^1 \quad (4.5\text{-}3)$$

위 식에서, 전개 항수(m개)가 미지 파라미터의 수(4개, 1개)보다 많으면 식(3.14)에서와 같이 파라미터를 결정할 수 있다.

$$\boldsymbol{\theta}_1 = (\boldsymbol{H}_1{}^{T}\boldsymbol{H}_1)^{-1}\boldsymbol{H}_1{}^{T}\boldsymbol{r}_1 \quad (4.6\text{-}1)$$
$$\boldsymbol{\theta}_2 = (\boldsymbol{H}_2{}^{T}\boldsymbol{H}_2)^{-1}\boldsymbol{H}_2{}^{T}\boldsymbol{r}_2 \quad (4.6\text{-}2)$$

다음의 표 1과 그림들은 구간 [0,1)에서 16항 전개하였을 때와 32항 전개하였을 때의 결과이다.

표 1. 파라미터들의 추정값

Table 1. Identified value of parameters

전개항수 m	$\hat{b}_{11}$	$\hat{b}_{12}$	$\hat{b}_{13}$	$\hat{n}_{112}$	$\hat{b}_2$
16	0.9149	2.1742	2.8057	4.6640	1.9027
32	0.9744	2.0150	2.8569	5.0210	1.9497
참 값	1	2	3	5	2

5. 결 론

블록 펄스 함수를 이용하여 미분방적식으로 모델링된 시스템의 파라미터를 추정하고자 할 때, 블록 펄스 함수 적분

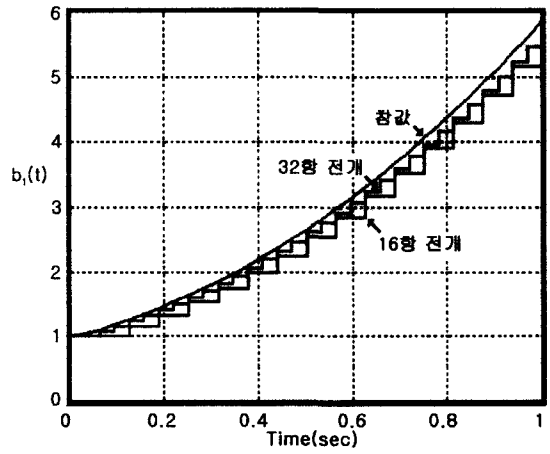


그림 4.1 파라미터 $b_1(t) = 1 + 2t + 3t^2$ 값의 추정

Fig. 4.1 Identifying of

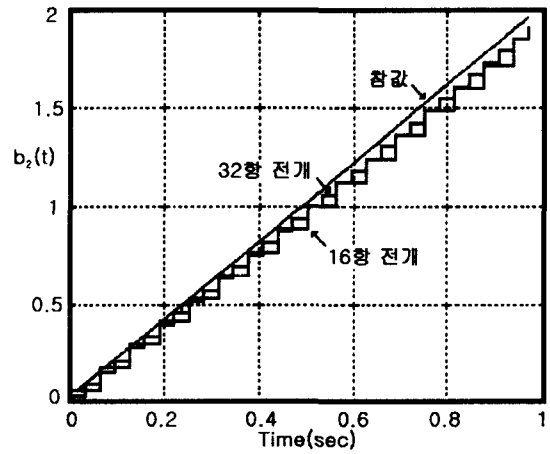


그림 4.2 파라미터 $b_2(t) = 2t$ 값의 추정

Fig. 4.2 Identifying of $b_2(t) = 2t$

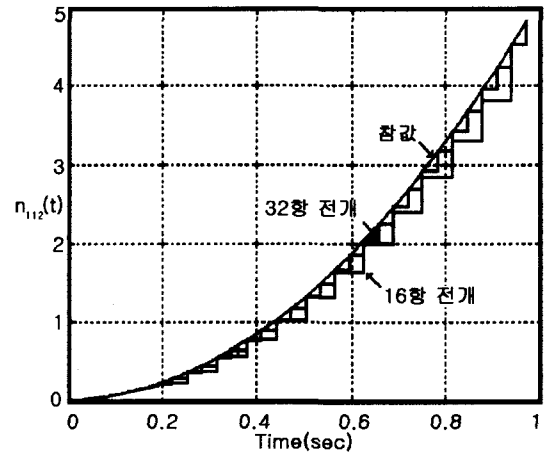


그림 4.3 파라미터 $n_{112}(t) = 5t^2$ 값의 추정

Fig. 4.3 Identifying of $n_{112}(t) = 5t^2$

연산행렬(일반적인 블록 펄스 적분 연산 행렬)을 적분 수만큼 곱하는 방법이 쓰여왔다. 하지만 블록 펄스 함수적분 연산행렬에는 오차가 포함되어 있기 때문에 여러번 곱을 행하다 보면 오차의 누적이 점차 심해지게 된다. 또한 블록 펄스 함수 적분연산행렬은 시변요소를 갖는 형태에 대한 고려하지 않은 것이므로 결과적으로 연산 단계의 증가를 초래하게 된다. 확장된 블록 펄스적분연산행렬은 유도과정이 블록 펄스 함수 적분연산행렬보다 약간 복잡한 단점이 있기는 하지만, 반복연산을 통해 구해진 결과를 적용함에 있어서는 여러 가지 장점을 갖게된다.

쌍일차계는 알려진 비선형계중에서 가장 간단한 형태이며, 선형계와의 유사성 때문에 많은 관심을 끌어왔다. 쌍일차계의 파라미터를 추정하기 위해서는 제어입력과 상태의 곱으로 이루어진 항의 처리가 문제로 남아있었는데 이것은 직교함수의 비결합 특성을 통해 해결할 수 있다는 것이 밝혀졌다. 하지만 쌍일차계의 파라미터가 시변요소를 포함하고있을 경우의 추정은 복잡한 연사과정에 의존할 수밖에 없었다.

본 논문은 확장된 블록 펄스 적분 연산 행렬을 제안하고, 비선형계의 일종인 쌍일차계의 시변요소를 갖는 파라미터를 추정하기 위해서 의 특성을 적용하였고, 그결과 연산과정이 간단해짐을 수식을 통해 보이고 정확성이 향상된다는 것을 그림을 통해 보였다. 또한, 파라미터의 추정과정에서 블록 펄스 함수의 비결합 특성을 간접적으로 보였다.

참 고 문 헌

[1] K. R. Palisamy, "System Identification via Block Pulse Functions", Int. J. Systems Sci. Vol. 10, pp. 1493, 1981.

[2] G. P. Rao, "System Identification via Walsh Functions", Pro. IEE., Vol. 122, pp. 1160, 1975

[3] P. R. Clement, "Laguerre Functions in System Analysis and Parameter Identification", J. Fran. Inst., Vol. 133, pp. 85, 1982

[4] Paraskevopoulds, :Legendre Series Approach to Identification and Analysis of Linear Systems", IEEE. Trans, Auto. Cont., Vol. 30, pp. 585, 1985

[5] N. S. Corrington, "solution of Differential and Integral Equations with Functions", IEEE Trans. Circuit Theory, Vol. CT-20, No. 5, pp. 470, 1973

[6] C. F. Chen and C. H. Hsiao, "Design of Piecewise Constant Gains for Optimal Control via Walsh Functions", IEEE Trans. Autom. Control, Vol. AC-20, pp. 596, 1975

[7] C. F. Chen and C. H. Hsiao, "A State Space Approach to Walsh Series Solution of Linear Systems", Int. J. Systems Sci., Vol. 6, pp. 833, 1975

[8] C. F. Chen, Y. T. Tsay and T. T. Wu, "Walsh Operational Matrices for Fractional Calculus and Their Application to Distributed Systems", J. Fanklin Inst., Vol. 303, pp. 267, 1977

[9] C. Hwang and T. Y. Guo, "Identification of Lumped Linear Time-varying Systems via Block Pulse Functions", Int. J. Systems Sci., Vol. 15, pp. 361, 1984

[10] C. H. Wang, "Generalized Block Pulse Operational Matrices and Their Applications to Operational Calculus", Int. J. Control, Vol. 36, No. 1, pp. 67 1982

[11] Z. H. Jiang and W. Schaufelberger, "Block-pulse Functional and Their Applications in Control Systems", Springer-Verlag, 1992

[12] G. Strang, "Linear Algebra and Its Applications", 2nd ed., Academic Press, pp. 103, 1980

[13] N. S. Hsu, "Identification of Non-linear Distributed Systems via Block-pulse Functions", Int. J. Control, Vol. 36, No. 2, pp. 281, 1982

[14] M. S. P. Sinha, V. S. Rajamani and A. K. Sinha, "Identification of Non-linear Distributed Systems Using Walsh Functions", Int. J. Control, Vol. 32, No. 4, pp. 669, 1980

저 자 소 개

김 태 훈 (金 泰 勳)

1971년 2월 1일 생. 1995년 성균관대 공대 전기공학과 졸업. 1997년 동 대학원 전기전자 및 컴퓨터 공학과 졸업(석사). 1999년~현재 동 대학원 전기전자 및 컴퓨터 공학과 박사과정

Tel : 0331-290-7167
E-mail : taihoon@netsgo.com

김 진 태 (金 珍 泰)

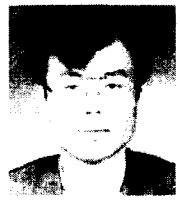
1972년 1월 1일 생. 1997년 경성대 공대 전기공학과 졸업. 1999년 성균관대 대학원 전기전자 및 컴퓨터 공학과 졸업(석사). 1999년~현재 동 대학원 전기전자 및 컴퓨터공학과 박사과정

Tel : 0331-290-7167
E-mail : oshkjt@ece.skku.ac.kr

이 승 (李 勝)

1963년 5월 12일 생. 1988년 성균관대 공대 전기공학과 졸업. 1990년 성균관대 대학원 전기공학과 졸업(석사). 2000년 성균관대 대학원 전기공학과 졸업(공박). 1993~현재 대림대 자동화시스템공학과 공학부 부교수

Tel : 031-467-4890, 4853
E-mail : slee@daelim.ac.kr



이 명 규 (李 明 珪)

1960년 12월 15일생. 1982년 성균관대 공대 전기공학과 졸업. 1984년 동 대학원 전기공학과 졸업(석사). 1990년 동 대학원 전기공학과 졸업(공박). 1991~현재 경성대 전기전자 및 컴퓨터공학부 교수

Tel : 051-620-4772
E-mail : mklee@star.kyungsung.ac.kr

안 두 수 (安 斗 守)

1937년 6월 9일 생. 1960년 서울대 공대 전기공학과 졸업. 1964년 서울대 대학원 전기공학과 졸업(석사). 1982년 중앙대 대학원 전기공학과 졸업(공박). 1971~현재 성균관대 전기전자 및 컴퓨터공학부 교수

Tel : 0331-290-7101
E-mail : dsahn@ece.skku.ac.kr