

**블럭펄스 함수의 새로운 적분연산 행렬을 이용한
선형 시변계의 해석**

조영호* · 신승권* · 박정호* · 이한석** · 김재일*** · 안두수*
* 성균관대학교 전기전자및컴퓨터공학과 ** 한국원자력연구소 *** 동의공업대학 전자계산과

**Analysis of Time-Varying Linear System
Using the New Integral Operational Matrix via Block Pulse Functions**

Young-ho Cho* · Seung-kwon Shin* · Jung-ho Park* · Han-seok LEE** · Jae-il Kim*** · Doo-doo Ahn*
* Sungkyunkwan University Electrical & Computer Engineering
** Korea Atomic Energy Research Institute
*** Dongeui Institute of Technology Computer Science

Abstract - The operational properties of BPF (block-pulse functions) are much applied to the analysis of time-varying linear systems. The integral operational matrix of BPF converts the systems in the form of the differential equation into the algebraic problems. But the errors caused by using the integral operational matrix make it difficult that we exactly analyze time-varying linear systems. So, in this paper, to analyze time-varying linear systems we had used the recursive algorithm derived from the new integral operational matrix. And the usefulness of the proposed method is verified by the example.

1. 서 론

BPF는 구분연속값(piecewise-continuous constant)을 가지는 직교함수 집합으로 1969년 전기공학자인 Harmuth에 의해 처음 도입되었다. Chen과 Hsiao에 의해 처음 도입된 적분연산 행렬은 BPF 및 여러 직교함수들이 시스템의 해석, 동정, 제어기 설계 등의 분야에 널리 이용될 수 있는 계기를 제공하였고[1-2]. 또한 단일 적분연산 행렬이 월쉬영역으로부터 BPF영역으로 전환된다면 계산상의 복잡성을 현저히 줄일 수 있다는 것이 밝혀졌다. 1970년대 이후 BPF는 월쉬(Walsh) 또는 하알(Haar)함수 등의 직교함수보다 연산이 간단하기 때문에 선형 시변계와 비선형계 등의 제어문제 등에 널리 이용되어 왔다[3-4]. 1978년에 Rao 및 Strinivasan등은 시스템 해석에 BPF 변환을 이용하였고, 그 다음해에 시지연(time delay)에 대한 새로운 BPF의 연산행렬을 소개함으로써 직교함수의 응용을 더욱 확장시켰다[2].

본 연구에서는 선형 시변계를 해석하기 위한 새로운 적분연산 행렬을 이용하여 반복 연산 알고리즘(recursive algorithm)을 유도하였다. 라그랑지 보간법(Lagrange interpolation)의 근사함수에 의해 유도된 새로운 적분연산 행렬은 기존의 방법보다 오차를 줄일 수 있다[6]는 점을 이용하였다. 그리고 기존의 알고리즘과 제시한 방법을 비교하여 훨씬 유용하다는 것을 예를 통하여 밝혔다.

2절에서는 BPF 및 적분연산행렬을 소개하였고, 3절에서는 새로운 적분연산 행렬을 이용하여 반복 연산알고리즘을 유도하였다. 4절에서는 선형 시변시스템의 적용례를 보였고, 5절에서는 결론으로 구성하였다.

2. 블럭펄스 함수

BPF는 1과 0의 값만을 취하며 다음과 같이 구간 [0, t_f]에서 정의된다[1-3, 5].

$$\phi_i(t) = \begin{cases} 1, & i\Delta t \leq t < (i+1)\Delta t \\ 0, & \text{그외구간} \end{cases} \quad (2.1)$$

단, $i = 0, 1, 2, \dots, m-1$

식(2.1)은 구간 $t \in [0, t_f]$ 이 m개의 세부구간으로 나누어짐을 보인다. 그리고 i번째 BPF $\phi_i(t)$ 는 미소구간 $\Delta t = t_f/m$ 에서 단위 크기를 갖는 한 개의 구형파를 가진다.

또한 BPF들은 구간 $t \in [0, t_f]$ 에서 다음과 같은 비결합 특성과 직교특성을 가진다.

$$\phi_i(t) \phi_j(t) = \begin{cases} \phi_i(t), & i=j \\ 0, & i \neq j \end{cases} \quad (2.2)$$

$$\int_0^{t_f} \phi_i(t) \phi_j(t) dt = \begin{cases} \Delta t, & i=j \\ 0, & i \neq j \end{cases} \quad (2.3)$$

단, $i, j = 0, 1, 2, \dots, m-1$

구간 $t \in [0, t_f]$ 에서 적분 가능한 임의의 함수 $f(t)$ 는 m개의 BPF로 식(2.4)와 같이 유한급수 전개할 수 있다[1-2, 5].

$$f(t) \cong \sum_{i=0}^{m-1} f_i \phi_i(t) \quad (2.4)$$

$$f_i = \frac{1}{\Delta t} \int_0^{t_f} f(t) \phi_i(t) dt = \frac{1}{\Delta t} \int_{i\Delta t}^{(i+1)\Delta t} f(t) dt \quad (2.5)$$

식(2.5)에서 i번째 BPF의 계수벡터 f_i 는 i번째 미소구간에서 원함수 $f(t)$ 의 적분 평균값이다. 이는 i번째 BPF의 계수벡터인 f_i 는 $f(t)$ 의 곡선 중 미소구간 $t \in [i\Delta t, (i+1)\Delta t]$ 에서의 면적과 관계가 있다. 이와 같은 관점에서 함수 $f(t)$ 를 쉽게 BPF 급수로 전개할 수 있다.

적분 가능한 임의의 함수를 BPF로 유한급수 전개할 수 있는 것과 같이 BPF의 적분 역시 같은 적분 연산행렬을 이용하여 유한 급수 전개할 수 있고[1-3] 적분 연산행렬은 다음과 같다.

$$P = \frac{\Delta t}{2} \begin{bmatrix} 1 & 2 & 2 & \dots & 2 \\ 0 & 1 & 2 & \dots & 2 \\ 0 & 0 & 1 & \dots & 2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \end{bmatrix} \quad (2.6)$$

단, $\Delta t = t_f/m$

그리고 새로운 적분 연산행렬은 다음과 같다[6].

$$\overline{P} = \frac{\Delta t}{2} \begin{bmatrix} 1 & \frac{7}{3} & \frac{23}{12} & 2 & 2 & \cdots & 2 & 2 \\ 0 & \frac{3}{4} & \frac{7}{3} & \frac{23}{12} & 2 & \cdots & 2 & 2 \\ 0 & 0 & \frac{3}{4} & \frac{7}{3} & \frac{23}{12} & \cdots & 2 & 2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & \frac{3}{4} & \frac{7}{3} \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & \frac{3}{4} \end{bmatrix} \quad (2.7)$$

단, $\Delta t = t_f/m$

이러한 적분 연산행렬은 BPF의 적분을 대수적인 계산으로 바꾸어 주어 대단히 편리하다.

3. 선형 시변시스템의 해석

앞절에서 언급한 BPF와 적분 연산행렬을 이용하여 다음과 같은 n 차 선형 시변시스템의 해석하자.

$$\frac{d}{dt} \mathbf{x}(t) = \mathbf{A}(t) \mathbf{x}(t) + \mathbf{B}(t) \mathbf{u}(t) \quad (3.1a)$$

$$\mathbf{x}(t_0) = \mathbf{x}_0 \quad (3.1b)$$

단, $\mathbf{x}(t)$ 와 $\mathbf{u}(t)$ 는 각각 n 차 상태벡터와 r 차 입력벡터

$\mathbf{A}(t)$ 와 $\mathbf{B}(t)$ 는 각각 $n \times n$, $n \times r$ 행렬

먼저 식(3.1a)의 양변을 적분하면 다음과 같다.

$$\mathbf{x}(t) - \mathbf{x}(t_0) = \int_{t_0}^t \mathbf{A}(t) \mathbf{x}(t) dt + \int_{t_0}^t \mathbf{B}(t) \mathbf{u}(t) dt \quad (3.2)$$

시스템의 시변행렬 $\mathbf{A}(t)$ 와 상태변수 $\mathbf{x}(t)$ 를 BPF로 유한급수 전개하면 다음과 같이 표현된다.

$$\mathbf{A}(t) \cong \sum_{i=0}^{m-1} \mathbf{A}_i \phi_i(t) \quad (3.3)$$

$$\mathbf{x}(t) \cong \sum_{i=0}^{m-1} \mathbf{X}_i \phi_i(t) \quad (3.4)$$

여기서 BPF의 비결합 특성을 이용하면

$$\begin{aligned} \mathbf{A}(t) \mathbf{x}(t) &\cong \left[\sum_{i=0}^{m-1} \mathbf{A}_i \phi_i(t) \right] \left[\sum_{j=0}^{m-1} \mathbf{X}_j \phi_j(t) \right] \\ &= \sum_{i=0}^{m-1} \mathbf{A}_i \mathbf{X}_i \phi_i(t) \end{aligned} \quad (3.5)$$

와 같고, 시변행렬 $\mathbf{B}(t)$ 와 입력 $\mathbf{u}(t)$ 를 BPF로 각각 유한급수 전개한 후 비결합 특성을 이용하면,

$$\begin{aligned} \mathbf{B}(t) \mathbf{u}(t) &\cong \left[\sum_{i=0}^{m-1} \mathbf{B}_i \phi_i(t) \right] \left[\sum_{j=0}^{m-1} \mathbf{U}_j \phi_j(t) \right] \\ &= \sum_{i=0}^{m-1} \mathbf{B}_i \mathbf{U}_i \phi_i(t) \end{aligned} \quad (3.6)$$

와 같다. 이제 식(3.3)~(3.6)의 관계를 식(3.2)에 도입하면 다음과 같다.

$$\begin{aligned} \sum_{i=0}^{m-1} \mathbf{X}_i \phi_i(t) - \sum_{i=0}^{m-1} \mathbf{x}_{0i} \phi_i(t) \\ = \sum_{i=0}^{m-1} \mathbf{A}_i \mathbf{X}_i \int_0^t \phi_i(\tau) d\tau + \sum_{i=0}^{m-1} \mathbf{B}_i \mathbf{U}_i \int_0^t \phi_i(\tau) d\tau \end{aligned} \quad (3.7)$$

단, \mathbf{x}_{0i} 는 초기값으로 $\mathbf{x}(0)$ 의 BPF 계수벡터

여기서 기존의 반복 연산 알고리즘은 식(3.7)에서 식(2.6)의 BPF 적분 연산식을 도입하여 나타낸 것으로 다

음과 같다[7].

$$\mathbf{X}_0 = \left(\mathbf{I}_n - \frac{1}{2} \Delta t \mathbf{A}_0 \right)^{-1} \left[\mathbf{x}(0) + \frac{1}{2} \Delta t \mathbf{B}_0 \mathbf{U}_0 \right] \quad (3.8a)$$

$$\begin{aligned} \mathbf{X}_j = \left(\mathbf{I}_n - \frac{1}{2} \Delta t \mathbf{A}_j \right)^{-1} \left[\left(\mathbf{I}_n + \frac{1}{2} \Delta t \mathbf{A}_{j-1} \right) \mathbf{X}_{j-1} \right. \\ \left. + \frac{1}{2} \Delta t \left(\mathbf{B}_{j-1} \mathbf{U}_{j-1} + \mathbf{B}_j \mathbf{U}_j \right) \right] \end{aligned} \quad (3.8b)$$

단, $j = 1, 2, \dots, m-1$

또한 제시하고자하는 알고리즘은 식(3.7)에서 식(2.7)의 BPF 적분 연산식을 도입하여 다음과 같이 표현할 수 있다.

$$\begin{aligned} \sum_{i=0}^{m-1} \mathbf{X}_i \phi_i(t) - \sum_{i=0}^{m-1} \mathbf{x}_{0i} \phi_i(t) \\ = \sum_{i=0}^{m-1} \left[\left(\mathbf{A}_i \mathbf{X}_i + \mathbf{B}_i \mathbf{U}_i \right) \int_0^t \phi_i(\tau) d\tau \right] \\ = \sum_{i=0}^{m-1} \left[\left(\mathbf{A}_i \mathbf{X}_i + \mathbf{B}_i \mathbf{U}_i \right) \left(\frac{1}{8} \Delta t \phi_i(t) \delta(t) \right) \right. \\ \left. + \frac{3}{8} \Delta t \phi_i(t) + \frac{7}{6} \Delta t \phi_{i+1}(t) + \frac{23}{24} \Delta t \phi_{i+2}(t) \right. \\ \left. + \Delta t \sum_{j=i+3}^{m-1} \phi_j(t) \right] \end{aligned} \quad (3.9)$$

$\mathbf{x}(t)$ 의 BPF 계수벡터 \mathbf{X}_i 는 위 식으로부터 BPF 반복 연산 알고리즘을 다음과 같이 유도할 수 있다.

$$\mathbf{X}_0 = \left(\mathbf{I}_n - \frac{1}{2} \Delta t \mathbf{A}_0 \right)^{-1} \left[\mathbf{x}(0) + \frac{1}{2} \Delta t \mathbf{B}_0 \mathbf{U}_0 \right] \quad (3.10a)$$

$$\begin{aligned} \mathbf{X}_1 = \left(\mathbf{I}_n - \frac{3}{8} \Delta t \mathbf{A}_1 \right)^{-1} \left[\mathbf{x}(0) + \frac{7}{6} \Delta t \mathbf{A}_0 \mathbf{X}_0 \right. \\ \left. + \frac{1}{24} \Delta t \left(28 \mathbf{B}_0 \mathbf{U}_0 + 9 \mathbf{B}_1 \mathbf{U}_1 \right) \right] \end{aligned} \quad (3.10b)$$

$$\begin{aligned} \mathbf{X}_2 = \left(\mathbf{I}_n - \frac{3}{8} \Delta t \mathbf{A}_2 \right)^{-1} \left[\mathbf{x}(0) + \frac{1}{24} \Delta t \left(23 \mathbf{A}_0 \mathbf{X}_0 \right) \right. \\ \left. + 28 \mathbf{A}_1 \mathbf{X}_1 + \frac{1}{24} \Delta t \left(23 \mathbf{B}_0 \mathbf{U}_0 + 28 \mathbf{B}_1 \mathbf{U}_1 \right) \right. \\ \left. + 9 \mathbf{B}_2 \mathbf{U}_2 \right] \end{aligned} \quad (3.10c)$$

$$\begin{aligned} \mathbf{X}_j = \left(\mathbf{I}_n - \frac{3}{8} \Delta t \mathbf{A}_j \right)^{-1} \left[\frac{1}{24} \Delta t \left(\mathbf{A}_{j-3} \mathbf{X}_{j-3} \right) \right. \\ \left. - 5 \mathbf{A}_{j-2} \mathbf{X}_{j-2} + \left(\mathbf{I}_n + \frac{19}{24} \Delta t \mathbf{A}_{j-1} \right) \mathbf{X}_{j-1} \right. \\ \left. + \frac{1}{24} \Delta t \left(\mathbf{B}_{j-3} \mathbf{U}_{j-3} - 5 \mathbf{B}_{j-2} \mathbf{U}_{j-2} \right) \right. \\ \left. + 19 \mathbf{B}_{j-1} \mathbf{U}_{j-1} + 9 \mathbf{B}_j \mathbf{U}_j \right] \end{aligned} \quad (3.10d)$$

단, $j = 3, 4, \dots, m-1$

4. 적용 예

새로운 알고리즘의 유용성을 판단하기 위해 다음과 같이 표현되는 선형 시변계를 생각해 보자.

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ t & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}, \quad \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad (4.1)$$

식(4.1)의 실제값은 다음과 같다.

$$x_1(t) = 1, \quad x_2(t) = \frac{1}{2} t^2 \quad (4.2)$$

그리고 제안된 방법을 이용하여 아래와 같이 계수벡터 \mathbf{X} 를 구한다[7]. 단, 구간 $[0, 1)$, 전개항수 8

· 식(4.1)의 양변을 적분의 형태로 바꾸어 준다.

- 적분 형태를 식(3.7)과 같이 BPF 급수 전개한다.
- 식(3.10)의 적분 연산알고리즘을 적용하여 계수값을 구한다.

표 1은 계수벡터 X_2 의 값을 나타내었다. 여기서 식(2.5)에 의해 실제 계수벡터 값을 구했고 식(3.8)에 의해 기존의 적분 연산행렬로 계수벡터 값을 구했다.

표 1. 계수벡터 X_2 의 값

Table 1. Values of parameter vector X_2

| X_{2i} | 실제 계수 | 기존의 적분 연산행렬을 이용한 계수 | 제안된 적분 연산행렬을 이용한 계수 |
|----------|---------|---------------------|---------------------|
| X_{20} | 1.00260 | 1.00390 | 1.00390 |
| X_{21} | 1.01822 | 1.01953 | 1.01790 |
| X_{22} | 1.04947 | 1.05078 | 1.04947 |
| X_{23} | 1.09635 | 1.09765 | 1.09635 |
| X_{24} | 1.15885 | 1.16015 | 1.15885 |
| X_{25} | 1.23697 | 1.23828 | 1.23697 |
| X_{26} | 1.33072 | 1.33203 | 1.33072 |
| X_{27} | 1.44010 | 1.44140 | 1.44010 |

표 1에서 보는바와 같이 기존의 적분 연산행렬을 이용한 경우 계수값들이 약 0.0013의 오차가 계속 발생하였다. 반면에 제안된 방법은 처음에 0.00130 그리고 두 번째 0.00032의 오차가 생겼고 그후에는 실제 계수값으로 수렴했다. 또 두 알고리즘 사이의 오차를 비교하기 위해

$$E_{error} = \sqrt{\sum_{i=1}^m (x_i - \bar{X}_i)^2} \quad (4.3)$$

여기서, x_i : 실제 계수.

\bar{X}_i : 적분연산에 의해 얻어진 계수

의 식을 이용하였는데, 기존의 값은 0.010417의 오차가 발생하였고, 제안된 값은 0.001628을 얻었다. 즉, 제안된 적분 연산행렬을 이용한 경우에 오차값이 기존의 방법에 비해 8배정도 적었다.

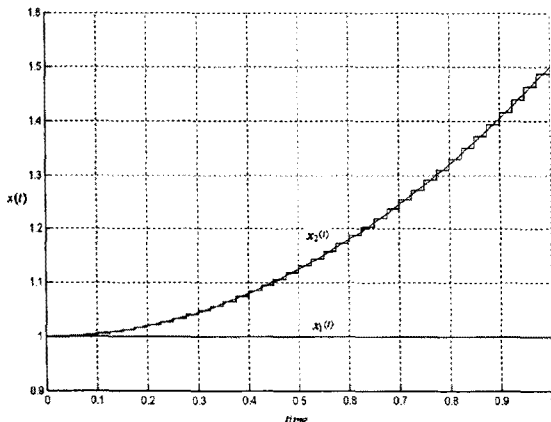


그림 1 상태 궤적
Fig. 1 state trajectory

그림 1은 상태 $x_1(t)$ 과 $x_2(t)$ 의 시간에 따른 변화를 나타낸 것이다. 그림에서 계단형 $x_2(t)$ 는 40항 전개시 제안된 방법을 그린 것이고, 실선 $x_2(t)$ 는 실제값을 그린 것이다. 상태 $x_1(t)$ 은 상수값이므로 실제값과 제안한 방법에 의한 값이 같다.

5. 결 론

BPF 함수가 시스템의 해석 및 제어기의 설계에 많이 이용되는 이유는 간단한 형태와 편리한 연산 때문이다. 더욱이 적분연산 행렬의 경우에는 미·적분의 연산을 대수적으로 바꾸어 주기 때문에 더 편리하다 할 수 있다. 본 논문에서는 새로운 적분연산 행렬을 이용하여 반복 연산알고리즘을 제시하였다. 그리고 이 알고리즘을 기존의 알고리즘과 비교하였다. 앞절의 결과를 보면 제안한 방법이 기존의 방법보다 개선된 값을 얻었음을 알 수 있다. 특히 기존의 방법이 실제값으로 수렴하지 못하고 일정한 오차값을 유지하는 반면에 제안한 방법은 적분대상이 1차인 경우에 전개항수에 관계없이 3번째 계수값부터 실제 계수값에 수렴함을 볼 수 있었다. 그러므로 제안된 알고리즘을 이용한다면 향후 최적제어 문제와 비선형계의 해석에도 문제를 좀 더 쉽게 해결할 수 있으리라 사료된다.

(참 고 문 헌)

- [1] K. B. Datta and M. Mohan, Orthogonal Functions in Systems and Control, Word Scientific Publishing Co., 1995.
- [2] Z. H. Jian, W. Schoufelberger, Block Pulse Functions and Their Applications in Control System, Springer-verlag, 1992.
- [3] 이한석, 조영호, 이명규, 안두수, "블럭펄스 함수에 의한 비선형계의 2계층 최적제어", 대한전기학회 논문지, 47권 4호, pp. 494-502, 1998
- [4] A. Deb, G. Sarkar and S. K. Sen, "Linear pulse width modulated block pulse functions and their application to linear SISO feedback control system identification", IEE Proc. Control Theory Appl., Vol. 142, No. 1, pp. 44-50 January, 1995
- [5] W. Shienyu, "Convergence of block pulse series approximation solution for optimal control problem", Int. J. Systems Sci. Vol. 21, No. 7, pp. 1355-1368, 1990.
- [6] 조영호, 신승권, 이한석, 안두수, "보간법을 이용한 블럭펄스 함수에 대한 새로운 적분 연산행렬의 유도", 대한전기학회 논문지, 48권 6호 게재 예정, 1999
- [7] N. S. Hsu and B. Cheng, "Analysis and optimal control of time-varying linear systems via block pulse functions", Int. J. Control, Vol. 33, No. 6, 1107-1122, 1981