

# 비선형 공정제어를 위한 매개변수 식별기법의 새로운 응용

## (A Novel Application of the Identification Technique to Control of Nonlinear Processes)

李志泰\*, 卞增男\*\*  
(Jietae Lee and Zeungnam Bien)

### 要 約

비선형 공정제어 문제에서 자주 요구되는 비선형 대수방정식들을 푸는 알고리즘을 제시하고자 한다. 먼저 방정식 변수들을 매개 변수화한 후 순환 식별 기법(recursive identification technique)을 적용하여 새로운 알고리즘을 구성한다. 결과적인 알고리즘의 형태와 그 특성은 Broyden의 Quasi-Newton법과 유사하지만, 그 유도과정과 순환식은 차이가 있으며, Broyden의 방법이 갖고 있는 거의 모든 장점을 갖고 있으면서 몇몇 풀기 어려운 함수에 대하여 더욱 빠르고 믿을 수 있음을 수치 비교로 알 수 있었다.

### Abstract

Algorithms for solving a set of nonlinear simultaneous equations, which is frequently required in problems of controlling nonlinear processes, are proposed. Here the equation variables are first parameterized and a recursive identification technique is utilized. The forms and characteristics of the resultant algorithms are very similar to the Broyden's quasi-Newton method, but their derivations and final recursion equations are different. Our methods possess almost all the merits of the Broyden's and numerical comparisons show our methods to be more efficient and reliable for some difficult problems.

### I. 序 論

순환 식별 기법(recursive identification technique)은 주로 동(動) 공정의 매개변수들을 찾아내는데 이용되고 있다. 이 매개변수들을 찾는 것은 그 단독으로 사용되기도 하지만 특히 self-tuning regulator 혹은

다른 적응제어의 중요한 부분으로 많이 이용되고 있다.

그러나 본 연구에서는 또 다른 응용으로써 여러 비선형 공정의 제어 문제에서 나타나는 비선형 대수 방정식들을 푸는데 이용할 수 있음을 보였다. 비선형 정(靜) 공정 혹은 유사 정 공정의 제어 문제는 바로 이런 비선형 대수 방정식을 해석하는 문제임을 알 수 있다. 예를 들어 살펴보면 그림 1에 보인 바와같이 공정의 제품의 주어진 양을 얻기 위해 필요한 원료의 입력량을 계산하여 조절하는 것과 같은 거시적인 안목으로 공정을 보면 공정의 입력과 출력은 정상상태의 비선형 함수관계를 가지며 이런 유사 정상상태로 운전되는 공정의 제어는 공정의 입·출력 함수관계의 역함수를 찾

\*正會員, 慶北大學校 工科大学 化學工學科  
(Dept. of Chem. Eng., Kyung Pook National Univ.)

\*\*正會員, 韓國科學技術院 電氣 및 電子工學科  
(Dept. of Elec. Eng., KAIST)

接受日字: 1983年 11月 15日

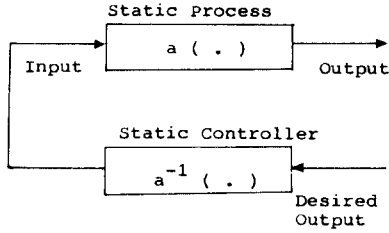


그림 1. 정 공정의 제어 모델  
Fig. 1. Control model of static process.

는 것으로써 비선형 대수 방정식을 해석하는 것이다. 이 외에도 비선형 동(動) 공정의 최적제어 문제에서 나타나는 경계치 문제(two point boundary value problem)를 Shooting법으로 푸는 것도 이에 속하는 것으로 본 연구 결과를 적용할 수 있다.<sup>6)</sup>

지금까지 이런 비선형 대수 방정식들을 푸는 방법으로는 고전적인 Newton-Raphson법과 Quasi-Newton법이 많이 쓰이고 있지만 Newton-Raphson법의 경우에는 수렴속도가 아주 빠른 반면에 Jacobian 행렬과 이의 역행렬을 매번 계산하여야 한다. 경계치 문제와 같이 Jacobian 행렬을 구하기가 힘든 문제에는 적합하지 못하다. Quasi-Newton 법은 수렴속도가 어느 정도 빠르면서 Jacobian 행렬을 매번 구할 필요는 없기 때문에 여러 문제에서 이용되고 있다. 그러나 이방법은 비선형성이 심하거나 초기값의 추정이 나쁠 경우 잘 동작하지 않는 것이 보고되고 있다. 본 연구는 비선형 공정의 제어 문제들중 특히 최적제어에서 나타나는 경계치 문제와 같이 Jacobian 행렬을 구하기가 힘든 비선형 대수 방정식들을 푸는 방법으로 Quasi-Newton법이 갖고 있는 특성을 갖고 있으면서 기존의 Quasi-Newton법으로 풀기 어려운 몇몇 문제에 대하여 더욱 빠르고 확실한 방법을 제시하고자 한다.

여러 비선형 대수 방정식들을 푸는 방법들을 검사하는 전형적인 문제들을 풀어 기존의 Quasi-Newton 법과 새로이 제시하는 방법들과의 비교를 하였다. 결론적으로 본 방법들이 몇몇 풀기 어려운 문제에 대하여 수렴은 더욱 빠르고 확실함을 수치 비교로 알 수 있었다.

II. 本 論

비선형 정상상태 혹은 유사 정상상태로 운전되는 공정의 제어 문제는 바로 비선형 대수 방정식의 근을 찾는 문제로 볼 수 있고, 이런 공정의 최적화 문제에서도 주어진 목적함수의 Gradient함수의 근을 찾는 문제로 볼 수 있다. 더불어 동 공정의 최적제어 문제에

서도 이를 푸는 여러 방법들중 하나로써 경계치 문제로 변환하여 Shooting법으로 해결할 경우 역시 비선형 대수 방정식들을 푸는 문제와 같아진다. 예를 들면 제약조건이 없는 초기값과 양끝 시간,  $[t_0, t_f]$  들이 주어진 최적제어 문제를 고려한다. 잘 알려진 절차로써 Hamiltonian을 구성하여 필요 조건을 구하면 State 변수의 시간 미분에 관한 방정식  $z(t)$ , Costate 변수의 시간 미분에 관한 방정식  $\lambda(t)$ , 그리고 Hamiltonian의 제어변수에 대한 부분 미분식을 영으로 하는 식으로 된다. 이들을 정리하여 제어변수를 소거하면 다음과 같은 State  $z(t)$ 와 Costate  $\lambda(t)$ 에 관한 식으로 된다.

$$z(t) = a(z(t), \lambda(t), t), z(t_0) = z_0$$

$$\lambda(t) = b(z(t), \lambda(t), t), \lambda(t_f) = c(x(t_f), t_f) \quad (1)$$

식(1)은 경계조건이  $t_0$ 와  $t_f$ 에서 각각 주어지는 경계치 문제로써 몇몇 해석 방법이 제시되어 있다. 그중 Shooting법으로 푸는 방법은  $\lambda(t_0)$ 를 추정하여 식(1)을  $t_0$ 에서  $t_f$ 까지 적분하여  $t_f$ 에서의 경계조건이 맞는 값  $\hat{\lambda}_0$ 를 찾는 것이다. 즉  $\hat{\lambda}(t_f)$ 와  $\hat{x}(t_f)$ 를  $\lambda(t_0) = \hat{\lambda}_0$ 로 하였을 경우의 Costate와 State 변수의  $t_f$ 에서의 값으로 두면,

$$d(\hat{\lambda}_0) \triangleq \hat{\lambda}(t_f) - c(\hat{x}(t_f), t_f) = 0 \quad (2)$$

의 비선형 대수방정식을 푸는 문제로 된다.<sup>8)</sup>

본 연구에서는 다음 식으로 대표되는 비선형 대수방정식들을 푸는 문제를 다루기로 한다.

$$f_i(x) = 0, \quad i=1, 2, \dots, n$$

이들을 벡터 형태로 나타내면,

$$f(x) = 0 \quad (3)$$

으로 표시할 수 있다.  $f$ 와  $x$ 는 각각 벡터 함수와 변수를 나타낸다.

식(3)을 반복적으로 푸는 방법들중 한 유형은

$$x^{(k+1)} = x^{(k)} - H^{(k)} f^{(k)} \quad (4)$$

으로 대표된다. 여기서 윗 첨자 (k)는 k번째 반복된 것을 의미하고, 특히  $f^{(k)}$ 는  $f(x^{(k)})$ 를 의미한다. 식(4)에서  $H^{(k)}$ 로  $x^{(k)}$ 에서의 Jacobian 행렬의 역행렬을 취하면 Newton-Raphson법이 되고, 이 역행렬의 근사치를 취하면 Quasi-Newton법이 된다.

Broyden은  $H^{(k)}$ 를 Jacobian행렬의 역행렬의 근사치로써 다음 식으로부터 얻었다.<sup>11)</sup>

$$H^{(k+1)} = H^{(k)} + \frac{(s^{(k)} - H^{(k)} y^{(k)}) s^{(k)T} H^{(k)}}{s^{(k)T} H^{(k)} y^{(k)}} \quad (5)$$

여기서  $s^{(k)} = x^{(k+1)} - x^{(k)}$ 이고,  $y^{(k)} = f^{(k+1)} - f^{(k)}$ 이다. Broyden의 Quasi-Newton법은 많은 연구가 있어 왔고 여러 문제에서 성공적으로 이용되고 있다.<sup>11)12)</sup>

본 연구에서는 Broyden의 방법을 개량하려는 시도

로써 순환 식별 개념을 도입하여 새로운 두 가지 방법을 제시하기로 한다.

먼저 식(3)을 Taylor 전개를 이용하여 매개변수화 한다.

$$f^{(k+1)} = f^{(k)} + A^{(k)}(x^{(k+1)} - x^{(k)}) + \gamma(x^{(k+1)} - x^{(k)}) \quad (6)$$

여기서  $A^{(k)} = \left( \frac{\partial f_i}{\partial x_j} \right)_{x^{(k)}}$  이고,  $\gamma$ 은 벡터함수로 다음의 조건을 만족한다.

$$\lim_{\|x\| \rightarrow 0} \|\gamma(x)\| / \|x\| = 0 \quad (7)$$

$\|\cdot\|$ 은 norm을 나타낸다.

식(6)에서  $\gamma(x^{(k+1)} - x^{(k)})$  항을 무시하고 나면 다음 두 가지의 모델과 결과적으로 유도되는 방법들을 얻을 수 있다.

방법 1 : 식(6)에서  $A^{(k)}$ 만 모르는 매개변수로 보고 지수함수적으로 가중된 최소자승법의 순환 식별 기법을 이용하여 구한다(부록 1 참조). 즉 secant 조건으로 알려진,

$$y^{(k)} = B^{(k+1)} s^{(k)} \quad (8)$$

으로부터  $A^{(k)}$ 의 근사치  $B^{(k+1)}$ 을 구한다.

$$B^{(k+1)} = B^{(k)} + \frac{(y^{(k)} - B^{(k)} s^{(k)}) s^{(k)\top} P^{(k)}}{s^{(k)\top} P^{(k)} s^{(k)}} \quad (9)$$

실제로 필요한  $B^{(k+1)}$ 를  $H^{(k)}$ 로 표시하여,  $H^{(k)}$ 에 관한 식으로 matrix inversion lemma를 이용하여 나타내면

$$H^{(k+1)} = H^{(k)} + \frac{(s^{(k)} - H^{(k)} y^{(k)}) s^{(k)\top} P^{(k)} H^{(k)}}{s^{(k)\top} P^{(k)} H^{(k)} y^{(k)}} \quad (10)$$

$$P^{(k+1)} = \frac{1}{\alpha} \left( P^{(k)} - \frac{P^{(k)} s^{(k)} s^{(k)\top} P^{(k)}}{\alpha + s^{(k)\top} P^{(k)} s^{(k)}} \right) \quad (11)$$

여기서  $\alpha$ 는 forgetting 인수로 알려진 상수로 0과 1사이의 값을 취한다. 식(11)에서  $P^{(k)}$ 가 positive-definite 대칭행렬이면  $P^{(k+1)}$ 도 역시 positive-definite 대칭행렬로 된다. 실제 적용에서는  $s^{(k)\top} P^{(k)} s^{(k)} \leq 0$ 이면 그 점에서 처음으로 돌아가 다시 시작한다.

정리 1 : 식(3)의 함수  $f$ 가 조건들 즉  $f$ 가 open set,  $O$ 에서 연속적으로 미분 가능하고,  $O$ 내에  $f(x^*) = 0$

이고  $A^* = \left( \frac{\partial f_i}{\partial x_j} \right)_{x^*}$ 가 nonsingular 한  $x^*$ 가 존재하고,  $O$ 내의 모든  $x^{(k)}$ 에 대하여  $\|A^{(k)} - A^*\| \leq \mu \|x^{(k)} - x^*\|$ 를 만족하는 상수  $\mu$ 가 존재한다는 조건들을 만족하면 식(4), (10) 그리고 (11)의 방법 1은 locally superlinearly 수렴한다.

증명은 부록(2)에 기술하였다.

방법 2 : 식(6)에서  $x^{(k+1)}$ 을 근  $x^*$ 로 치환하면  $f^{(k+1)} = 0$ 가 되고, 이로부터 직접  $x^*$ 와  $A^{(k)}$ 의 근사치를 동시에 구한다.  $A^{(k)}$ 가 nonsingular 하다고 가정하면  $x^{(k)} \approx x^* + A^{(k)-1} f^{(k)}$  (12)

으로 되고, 다음의 관계식을 만들 수 있다.

$$x^{(k)} = W^{(k+1)} u^{(k)} \quad (13)$$

여기서  $W^{(k+1)}$ 은  $(x^* : A^{(k)-1})$ 의 근사치를 나타내고  $u^{(k)}$ 는  $\begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$ 를 나타낸다. 식(13)에 앞의 방법 1에서

이용한 순환 식별 기법을 적용하여  $W^{(k)}$ 를 구한다.  $W^{(k)}$ 는 우리가 필요로 하는  $x^*$ 의 근사치를 포함하고 있다.

$$W^{(k+1)} = W^{(k)} + \frac{(x^{(k)} - W^{(k)} u^{(k)}) u^{(k)\top} R^{(k)}}{u^{(k)\top} R^{(k)} u^{(k)}} \quad (14)$$

$$R^{(k+1)} = \frac{1}{\alpha} \left( R^{(k)} - \frac{R^{(k)} u^{(k)} u^{(k)\top} R^{(k)}}{\alpha + u^{(k)\top} R^{(k)} u^{(k)}} \right) \quad (15)$$

$$x^{(k+1)} = \text{the first column of } W^{(k+1)} \quad (16)$$

식(14), (15) 그리고 (16)을  $x^*$ 와  $A^{(k)-1}$ 의 근사치부분으로 각각 가르면  $x^{(k+1)}$ 은 다음과 같이 된다.

$$x^{(k+1)} = x^{(k)} - \eta^{(k)} H^{(k)} f^{(k)} \quad (17)$$

여기서

$$\eta^{(k)} = \text{the first element of } \frac{R^{(k)} u^{(k)}}{u^{(k)\top} R^{(k)} u^{(k)}} \quad (18)$$

이고,  $A^{(k)-1}$ 의 근사치를  $H^{(k)}$ 로 나타내었다. 그리고  $\|f^{(k)}\|$ 가 0으로 감에 따라  $\eta^{(k)}$ 는 1에 수렴한다. 이들의 특성들과 식(17)은 본 방법 2가 Quasi-Newton 법과 같은 유형에 속하는 것됨을 알 수 있다.

### III. 수치적 계산 예

본 방법들의 응용성을 조사하기 위해 여섯가지의 잘 알려진 검사용 함수들을 풀었다. 널리 사용되는 Broyden의 Quasi-Newton법과 수렴속도 비교를 하였다. 함수들은 다음과 같다.<sup>[5]</sup> 여기서 아래 첨자는 각 벡터의 성분을 나타낸다.

- 함수 1 (Rosenbrock function) :

$$f_1(x) = 10(x_2 - x_1^2), \quad f_2(x) = 1 - x_1 \\ f(x) = 0 \text{ at } x^* = (1, 1)^T$$

- 함수 2 (Freudenstein and roth function) :

$$f_1(x) = -13 + x_1 + ((5 - x_2)x_2 - 2)x_2 \\ f_2(x) = -29 + x_1 + ((1 + x_2)x_2 - 14)x_2 \\ f(x) = 0 \text{ at } x^* = (5, 4)^T$$

- 함수 3 (Powell's badly scaled function) :

$$f_1(x) = 10000 x_1 x_2 - 1 \\ f_2(x) = \exp(-x_1) + \exp(-x_2) - 1.0001 \\ f(x) = 0 \text{ at } x^* = (1.098 \dots \times 10^{-5}, 9.106 \dots)^T$$

- 함수 4 (Powell's singular function) :

$$f_1(x) = x_1 + 10x_2, \quad f_2(x) = \sqrt{5}(x_3 - x_4), \\ f_3(x) = (x_2 - 2x_3)^2, \quad f_4(x) = \sqrt{10}(x_1 - x_4)^2 \\ f(x) = 0 \text{ at } x^* = (0, 0, 0, 0)^T$$

- 함수 5 (Broyden's tridiagonal function) :

$$f_1(x) = (3 + c_1 x_1)x_1 - 2x_2 + c_2$$

$$f_i(x) = -x_{i-1} + (3 + c_1 x_i) x_i + 2x_{i+1} + c_2, \quad i=2, 3, \dots, (n-1)$$

$$f_n(x) = -x_{n-1} + (3 + c_1 x_n) x_n + c_2$$

$$c_1 = -0.5, \quad c_2 = 1$$

• 함수 6 (Wood function) :

$$f_1(x) = -400(x_2 - x_1^2)x_1 - 2(1 - x_1)$$

$$f_2(x) = 200(x_2 - x_1^2) + 20(x_2 + x_4 - 2) + (x_2 - x_4)/5$$

$$f_3(x) = -360(x_4 - x_3^2)x_3 - 2(1 - x_3)$$

$$f_4(x) = 180(x_4 - x_3^2) + 20(x_2 + x_4 - 2) - (x_2 - x_4)/5$$

$$f(x) = 0 \text{ at } x^* = (1, 1, 1, 1)^T$$

각 방법들에 필요한 초기값들은 유한차분법으로 구하였다.<sup>[1]</sup> 방법 1의 경우는

$$P^{(0)} = \frac{1}{\beta^2} I$$

$$H^{(0)} = \beta Y^{-1}$$

여기서

$$Y = (y^{(01)} : y^{(02)} : \dots : y^{(0n)})$$

$$y^{(0i)} = f(x_1^{(0)}, x_2^{(0)}, \dots, x_i^{(0)} + \beta, \dots, x_n^{(0)}) - f(x^{(0)}),$$

$$i=1, 2, \dots, n$$

방법 2의 경우는

$$R^{(0)} = U^{(0)-T} U^{(0)}$$

$$W^{(0)} = X^{(0)} U^{(0)-1}$$

여기서

$$U^{(0)} = (u^{(01)} : u^{(02)} : \dots : u^{(0n)})$$

$$X^{(0)} = (x^{(01)} : x^{(02)} : \dots : x^{(0n)})$$

$$x^{(0i)} = (x_1^{(0)}, x_2^{(0)}, \dots, x_i^{(0)} + \beta, \dots, x_n^{(0)})^T$$

$$u^{(0i)} = \begin{pmatrix} 1 \\ f(x^{(0i)}) \end{pmatrix}, \quad i=1, 2, \dots, n$$

$$u^{(0)} = \begin{pmatrix} 1 \\ f(x^{(0)}) \end{pmatrix}$$

들이다. 차분량  $\beta$ 는 함수 2를 풀때만 0.1로 하고, 그 외는 0.01로 하였다. 함수 2의 경우 0.01로 하면 각 방법들이 헤어지기 어려운 지역으로 들어가 버린다.

$\alpha$ 는 작을수록 수렴이 빨라지나 수치적으로 불안정해진다.  $\alpha$ 로 0.01을 취하였다. 제한한 방법들에서  $P^{(k)}$ 와  $R^{(k)}$ 는 positive-definite 대칭행렬들로서,  $s^{(k)T} P^{(k)} s^{(k)} > 0$  혹은  $u^{(k)T} R^{(k)} u^{(k)} > 0$ 들을 만족해야 한다. 수치적인 문제로 이 성질을 잃었을 경우에는 처음으로 돌아가 그 점에서 새로운 초기값들을 구해 새로 시작한다.

각 방법들의 성능을 비교하기 위해 각 방법이 주어진 함수의 norm의 제곱 즉  $\|f^{(k)}\|^2$ 을  $10^{-15}$ 까지 줄이는데 필요한 반복 횟수, 함수 계산 횟수, 그리고 마지막으로 구한 함수의 norm의 제곱을 표 1에 나타내었다.

함수 1과 5의 경우 비교적 풀기가 쉬운데 세 가지

표 1. 여섯 잘 알려진 검사 함수에 대한 성능비교표

Table 1. Performances for six well-known test functions

+ ;not converge within 100 iterations

\* ;erraneous termination

\*\* ;terminate to local optimum

( , , );number of iterations, number of function evaluations, final  $\|f\|^2$  with HP3000/III Fortran IV (16 bit, double precision).

번호	함수	Broyden의 방법	방법 1	방법 2
	$x^{*T}$			
1	(-1.2, 1)	(3, 6, 3.0×10 <sup>-20</sup> )	(3, 6, 1.1×10 <sup>-20</sup> )	(4, 7, 8.8×10 <sup>-20</sup> )
	(-12, 10)	(3, 6, 0)	(3, 6, 0)	(4, 7, 3.8×10 <sup>-20</sup> )
2	(-0.5, 2)	>100*	(16, 19, 9.1×10 <sup>-10</sup> )	(16, 19, 6.8×10 <sup>-10</sup> )
	(-0.6, 2.1)	(60, 63, 1.7×10 <sup>-17</sup> )	(13, 16, 4.7×10 <sup>-17</sup> )	(15, 18, 7.8×10 <sup>-17</sup> )
3	(0, 1)	(29, 32, 3.3×10 <sup>-22</sup> )	(20, 23, 7.3×10 <sup>-22</sup> )	(20, 23, 2.6×10 <sup>-22</sup> )
	(1, 1)	Overflow*	Overflow	(43, 49, 4.3×10 <sup>-17</sup> )
4	(3, -1, 0, 1)	(66, 71, 1.7×10 <sup>-10</sup> )	(41, 58, 6.6×10 <sup>-10</sup> )	(34, 44, 6.9×10 <sup>-10</sup> )
	(-3, 1, 0, -1)	>100	(43, 60, 8.5×10 <sup>-10</sup> )	(34, 44, 8.4×10 <sup>-10</sup> )
5	n=5(-1, ..., -1)	(7, 13, 3.7×10 <sup>-10</sup> )	(8, 14, 1.9×10 <sup>-10</sup> )	(8, 14, 1.4×10 <sup>-10</sup> )
	n=10(-1, ..., -1)	(7, 18, 4.9×10 <sup>-10</sup> )	(8, 19, 4.4×10 <sup>-10</sup> )	(8, 19, 6.8×10 <sup>-10</sup> )
	n=20(-1, ..., -1)	(7, 28, 5.4×10 <sup>-10</sup> )	(8, 29, 5.8×10 <sup>-10</sup> )	(8, 29, 1.3×10 <sup>-10</sup> )
6	(3, 1, 3, 1)	>100	(48, 53, 7.0×10 <sup>-10</sup> )	(27, 32, 3.5×10 <sup>-10</sup> )
	(3, -1, 3, -1)	(93, 98, 2.3×10 <sup>-10</sup> )	(79, 92, 3.7×10 <sup>-10</sup> )	(28, 33, 1.5×10 <sup>-10</sup> )
	(-3, -1, -3, -1)**	>100	(58, 71, 9.6×10 <sup>-10</sup> )	(59, 74, 2.5×10 <sup>-10</sup> )

방법들 모두 좋은 비슷한 결과를 주고 있다. 반면에 나머지 함수들에 대하여는 풀기가 까다로운데 제시한 방법들이 Broyden의 방법에 비해 훨씬 좋은 결과를 주는 것을 볼 수 있다. 즉 초기값 추정에 덜 민감하고 반복 횟수도 적다. 본 연구에서 제시한 두 방법들은 서로 비슷함을 알 수 있다.

#### IV. 結 論

비선형 공정제어 문제에서 자주 대두되는 비선형 대수 방정식들을 푸는 새로운 방법들을 제시하였다.

Quasi-Newton법과 같은 유형에 속하는 것들로서, 기존의 Broyden의 방법과는 그 유도과정과 결과식에서 차이가 있으며, Broyden의 방법에 비해 몇몇 풀기 어려운 문제에 대하여 더욱 수렴속도가 빠르며 확실함을 수치 비교로부터 알 수 있었다.

#### 附 錄

1) 순환 식별 기법: 여러 방법들이 제시되어 있지만 본문에서 이용한 지수함수적으로 가중된 최소자승법을 간략히 기술한다.<sup>[2]</sup>

본 방법은  $(q^{(k)}, v^{(k)})$ 의 자료 쌍들로부터  $q = \theta^T v$ 의 관계를 지을 수 있는 최적의  $\theta$ 를 찾는 것으로 다음의 목적함수를 최소화시키는  $\theta$ 를 구한다.

$$\min_{\theta^{(k+1)}} \left\{ \sum_{i=0}^k \alpha^{k-i} (q^{(i)} - \theta^{(k+1)T} v^{(i)})^2 \right\} \quad (19)$$

문제 (19)를 순환식을 유도하여 푼다.

$$\theta^{(k+1)} = \theta^{(k)} + \frac{D^{(k)} v^{(k)} (q^{(k)} - \theta^{(k)T} v^{(k)})}{\alpha + v^{(k)T} D^{(k)} v^{(k)}} \quad (20)$$

$$D^{(k+1)} = \frac{1}{\alpha} \left( D^{(k)} - \frac{D^{(k)} v^{(k)} v^{(k)T} D^{(k)}}{\alpha + v^{(k)T} D^{(k)} v^{(k)}} \right) \quad (21)$$

여기서

$$0 < \alpha \leq 1$$

이다.  $\alpha$ 는 옛 자료들을 잊어버리는 비율로써 작을수록 수렴속도를 증가시키지만 수치적인 문제를 일으킨다. 본 연구에서는 식(20)에서  $\alpha$ 를 제거하여 이용하였다. 이 변형으로 기존의 Quasi-Newton법에서 개발된 정리들을 이용할 수 있었다.

2) 정리 1의 증명

$E^{(k)} \triangleq B^{(k)} - A^*$ 로 두고,  $P^{(k) \frac{1}{2}}$ 은 positive-definite 대칭행렬  $P^{(k)}$ 의 자승근을 표시하는 것으로 한다. Matrix inversion lemma를 이용하면 식(11)은

$$P^{(k+1)^{-1}} = \alpha P^{(k)^{-1}} + s^{(k)} s^{(k)T}$$

으로 된다. 그리고 식(9), (11) 그리고 norm 정의로부터

$$\begin{aligned} & \| P^{(k+1) \frac{1}{2}} E^{(k+1)} P^{(k+1)^{-\frac{1}{2}} } \|^2 = \text{trace} ( P^{(k+1)^{-1}} E^{(k+1)T} \\ & P^{(k+1)} E^{(k+1)} ) = \text{trace} \left\{ \left( \alpha P^{(k)^{-1}} + s^{(k)} s^{(k)T} \right) E^{(k+1)T} \right. \\ & \left. \left( \frac{1}{\alpha} P^{(k)} - \frac{P^{(k)} s^{(k)} s^{(k)T} P^{(k)}}{\alpha (\alpha + s^{(k)T} P^{(k)} s^{(k)})} \right) E^{(k+1)} \right\} \\ & \leq \| P^{(k) \frac{1}{2}} E^{(k+1)} P^{(k)^{-\frac{1}{2}} } \|^2 + \frac{1}{\alpha} \| P^{(k) \frac{1}{2}} E^{(k+1) s^{(k)}} \|^2 \\ & \| P^{(k) \frac{1}{2}} E^{(k+1)} P^{(k)^{-\frac{1}{2}} } \|^2 \leq \| P^{(k) \frac{1}{2}} E^{(k)} P^{(k)^{-\frac{1}{2}} } \|^2 \end{aligned}$$

$$\begin{aligned} & \left( I - \frac{P^{(k) \frac{1}{2}} s^{(k)} s^{(k)T} P^{(k) \frac{1}{2}}}{s^{(k)T} P^{(k)} s^{(k)}} \right) \left\| \right. \\ & \left. + \left\| \frac{P^{(k) \frac{1}{2}} (y^{(k)} - A^* s^{(k)}) s^{(k)T} P^{(k) \frac{1}{2}}}{s^{(k)T} P^{(k)} s^{(k)}} \right\| \right\| \\ & \leq (1 - \theta^{(k)}) \| P^{(k) \frac{1}{2}} E^{(k)} P^{(k)^{-\frac{1}{2}} } \|^2 \\ & + \frac{\| P^{(k) \frac{1}{2}} (y^{(k)} - A^* s^{(k)}) \|^2}{\| P^{(k) \frac{1}{2}} s^{(k)} \|^2} \end{aligned}$$

여기서

$$\theta^{(k)} = \left( \frac{1}{2 \| P^{(k) \frac{1}{2}} E^{(k)} s^{(k)} \|^2} \right) \left( \frac{\| P^{(k) \frac{1}{2}} E^{(k)} s^{(k)} \|^2}{\| P^{(k) \frac{1}{2}} s^{(k)} \|^2} \right)^2$$

그리고  $E^{(k+1)} s^{(k)} = y^{(k)} - A^* s^{(k)}$ 로부터

$$\begin{aligned} & \| P^{(k+1) \frac{1}{2}} E^{(k+1)} P^{(k+1)^{-\frac{1}{2}} } \|^2 \leq (1 - \theta^{(k)}) \| P^{(k) \frac{1}{2}} E^{(k)} \\ & P^{(k)^{-\frac{1}{2}} } \|^2 + \frac{1}{\alpha} \| P^{(k+1) \frac{1}{2}} (y^{(k)} - A^* s^{(k)}) \|^2 \\ & + \frac{\| P^{(k) \frac{1}{2}} (y^{(k)} - A^* s^{(k)}) \|^2}{\| P^{(k) \frac{1}{2}} s^{(k)} \|^2} \end{aligned}$$

위의 식과 참고문헌 [3]의 theorem 5.2의 결과는 본 방법 1의 locally superlinear 수렴을 보여 준다. 여기서 식(9)와 (10)은 근본적으로 같은 식임을 밝혀 둔다.

參 考 文 獻

- [1] C.G. Broyden, "A class of methods for solving nonlinear simultaneous equations," *Math. Comp.*, vol.19, no.92, pp.577-593, Oct. 1965.
- [2] J.E. Dennis, JR. and J.J. More, "Quasi-Newton method, motivation and theory," *SIAM Review*, vol.19, no.1, pp.46-89, Jan. 1977.
- [3] G.C. Goodwin and R.L. Payne, *Dynamic system identification: experimental design and data analysis*, Academic Press, New York, pp.175-208, 1977.
- [4] Jietae Lee and Zeungnam Bien, *A Recursive Identification for Solving a Set of Nonlinear Equation*. in Preparation.
- [5] J.J. More, B.S. Garbow and K.E. Hillstrom, "Teoting unconstrained optimization softward," *ACM Trans. Math. Software*, vol.7, no.1, pp.17-41, March 1981.
- [6] M.G. Singh and A. Title, *System: Decomposition, Optimization and Control*. Pergamon Press, New York pp.329-383, 1978.