

## 개선된 NVP를 이용한 새로운 고장점 표정 계산 모델

장용원<sup>1</sup>, 김원하<sup>2</sup>, 한승수<sup>3</sup>

<sup>1</sup>차세대전력기술연구센터, "경희대학교 전자정보대학, <sup>2</sup>"명지대학교 정보공학과

<sup>3</sup>"전화: (031) 330-6761

## Fault Location Calculation using Improved NVP Model

Yong-Won Jang<sup>1</sup>, Won-Ha Kim<sup>2</sup>, Seung-Soo Han<sup>3</sup>

<sup>1</sup>NPTC, "School of Electronic and Information, Kyung Hee University

<sup>2</sup>"Department of Information Engineering, Myongji University

<sup>3</sup>E-mail : shan@mju.ac.kr

**Abstract** - 전력 에너지는 안정하고 신뢰할 수 있도록 고장에 대한 빠른 대처가 필요하다. 고장시 빠른 수리를 위해서는 보수 승무원에게 고장 위치를 정확하게 알려주어 올바른 위치에 도착할 수 있도록 고장점 표정 알고리즘의 정확도가 요구된다. 본 논문에서는 기존 1회선 분기점을 갖는 병행 2회선 송전선로의 고장점 표정 알고리즘을 이용하여 정확한 고장 위치를 찾는 방법으로 개선된 NVP(N-version programming) 모델을 적용한 새로운 계산 방법을 제안한다. 송전선로의 고장 데이터는 EMTP(Electro Magnetic Transients Program)을 사용하여 154[kV], 25[km] 분기된 병행 2회선 송전선로에서 고장지점과 고장 저항의 데이터를 이용하여 시뮬레이션 했다.

### 1. 서 론

송전선로에서 발생하는 각종 계통함수 정보를 상시 감시하여 고장발생시 신속한 고장복구를 위하여 정확한 위치를 찾아주는 고장점 표정 장치는 전력계통에 중요한 항목으로 대두되었다[1].

송전선로 상에서 고장발생시 고장점까지의 거리를 계산하는 고장점 표정알고리즘은 최근 십여 년 동안 활발히 연구되어왔다. 석용대상 계통은 크게 1회선 계통[2], 병행 2회선 계통[3], 분기점을 갖는 단단자 계통[1][4], 2회선 계통에서 전선회선의 영상분 전류를 이용한 방법으로 구분할 수 있다[5]. 병행 2회선으로 가설된 송전선로의 한 회선이 T분기된 경우 분기되지 않은 회선 1선 지각 사고 발생시 자기단의 정보만을 이용한 방법도 발표되고 있다[6].

알고리즘[6]을 기반으로 유도된 거리방정식은 시간 합수이며 미지수가 2개인 3차 복수 비선형방정식으로 해를 구하는 방법으로 Newton-Raphson 반복 계산법을 이용했다. 이 방법은 계산할 때 변수의 초기값을 결정해주어야 하며 결정된 초기값에 따라 계산 결과의 수렴 여부가 결정되는 문제가 있다. 이 문제를 해결하기 위해 software fault tolerance 기술을 이용한다. 이는 소프트웨어적인 고장이 시스템 일부에 발생하더라도 시스템 전체가 그 영향으로 고장 나지 않게 처리하는 기술이다[7].

본 논문에서는 1회선 분기점을 갖는 병행 2회선 송전선로의 고장점 표정 알고리즘으로 유도된 거리 방정식은 미지수가 2개인 3차 복수 비선형방정식이다. 기존에는 비선형 방정식의 해를 구하기 위해 Newton-Raphson 반복 계산법을 이용했다. 그러나 Newton-Raphson 방법은 계산 초기에 미지수의 초기값을 결정해줘야 하는데 이 초기값에 따라 계산 결과의 수렴 여부가 결정되는 문제를 가지고 있다. Newton-Raphson 반복법의 문제를 해결하고 신뢰할 수 있는 결과를 얻기 위해 NVP 모델을

적용했다. NVP 모델에서 사용한 version의 수는 3개이며 각각의 종류는 Newton-Raphson, Genetic Algorithm, Algebraic Equation 이다. 또한 Newton-Raphson의 미지수 초기값 결정 문제를 해결하기 위해 Genetic Algorithm으로 얻은 결과를 초기값으로 결정했다. Algebraic Equation은 MATLAB에 있는 함수를 이용하여 적용했다.

제안한 방법의 실험을 위한 송전선로의 고장 데이터는 EMTP(Electro Magnetic Transients Program)을 사용하여 154[kV], 25[km] 분기된 병행 2회선 송전선로에서 고장지점과 고장저항의 데이터를 얻었으며 구현은 MATLAB을 이용해서 시뮬레이션 했다.

### 2. 본 론

#### 2.1 고장점 표정 알고리즘

알고리즘[6]에서 제안한 1회선 분기점을 갖는 병행 2회선 송전선로의 방정식은 식 (1)과 같은 거리  $p$ 와 저항  $R_F$ 에 관한 3차 방정식으로 유도된다. 성분  $\alpha$ 는 시간에 따른 상수로 구할 수 있다.

$$a_3p^3 + a_2p^2 + a_1p + s_1R_Fp + a_0 + s_0R_F = 0 \quad (1)$$

거리 방정식을 계산하여 사고 지점  $p$ 를 찾는 과정은 그림1과 같다. DFT(Discrete Fourier Transform)를 사용하여 역상분 회로에서 전압과 전류를 계산한 후 타 회선의 영상 전류와 고장 전류를 알기 위해 분배계수를 이용한다. 구한 전압과 전류를 이용하여 비선형 3차 거리 방정식을 유도한다.

고장 위치를 찾기 위해 비선형 3차 방정식을 계산해야 한다. 보통 사용하는 계산법으로 Newton-Raphson 반복법을 이용한다.

하지만 Newton-Raphson 방법은 초기값을 어떤 값으로 선택하느냐에 따라 수렴여부를 결정하게 된다. 이러한 문제를 해결하기 위해서 Genetic Algorithm을 사용하여 global solution의 영역을 탐색한 후, 그 영역에 있는 점을 초기값으로 사용하여 Newton-Raphson 방법의 결과를 신뢰할 수 있도록 했다.

#### 2.2 제안한 NVP 계산 모델

그림 1은 논문[6]에서 제안한 순서를 블록도로 표시한 것이다. 계산 순서 중 비선형 3차 방정식을 유도하는 과정까지는 기존 과정에 준하여 구현한다. 본 논문에서 제안한 새로운 비선형방정식의 계산 모델은 그림1의 과정 중에 유도된 거리방정식 계산 과정에 속한다.

기존 논문에서는 비선형 방정식을 계산하기 위해 Newton-Raphson 반복 계산법을 사용했으나 미지수의 초기값 설정 문제가 있어날 수 있어 미리 고장 거리의 근사한 값을 찾아 주는 최적화 알고리즘을 사용하여 보

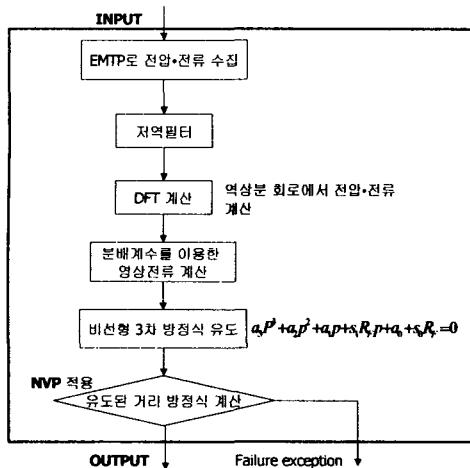


그림 1. 1회선 분기점을 갖는 병행 2회선 송전선로의 계산 과정

정한다. 또한 시스템 내부의 소프트웨어 오류로 인한 고장 거리의 오차를 줄이기 위해 NVP 모델을 제안한다. NVP 모델은 나수의 version을 이용하여 병렬로 처리된 결과를 모아 최상의 결과 값을 결정해 주는 모델이다. 원하는 결과를 찾아주는 DM(Decision Making)은 시스템의 특성에 맞게 실험적으로 만들어야 하며 DM의 성능에 따라 결과의 정확도가 결정된다.

본 연구에서 사용한 NVP 모델의 version의 수는 3개이고 각각의 종류는 Newton-Raphson(NR), Genetic Algorithm(GA), Algebraic Equation(AE)이다. Newton-Raphson의 초기값 결정 문제를 해결하기 위해 GA를 먼저 구하여 나온 결과값을 Newton-Raphson의 초기값으로 대입하여 문제를 해결했다. NVP 모델의 결과를 결정해 주는 DM은 version의 결과 중에 식(1)을 0에 가까운 값으로 만드는 결과를 선택하도록 했다.

그림 2는 그림 1에서 유도된 거리 방정식 계산 과정을 NVP 알고리즘을 적용한 새로운 계산 모델이다. 식 (1)에 해당하는 상수들을 계산하기 위해 다음과 같이 계산한다.

- Step 1: 식 (1)에 해당하는 상수를 계산한다.
- Step 2: 입력받은 데이터를 사용하여 GA를 수행한다.
- Step 3: 첫 세대에 구한 GA 결과를 이용해서

Newton-Raphson 반복법을 계산한다.

- Step 4: 과정 2의 결과를 받아 GA를  $k$  세대만큼 재생성하여 결과를 계산한다.

- Step 5: 입력받은 데이터를 사용하여 AE를 계산한다.

- Step 6: 과정 3, 4, 5에서 구한 각 version의 결과( $V_i$ )를 모아 각각의 결과가 0과 1 사이에 위치하는지 검사한다. 만약 위치하지 않는 version의 결과가 있으면 제외시킨다.

$$0 < V_i < 1 \quad i=1,2,3$$

- Step 7: 각 Version의 결과( $V_i$ ) 중에 서로 균접하는 2개의 결과를 찾아 평균값을 취한다( $V$ ).

- Step 8: 과정 7에서 구한 결과( $V$ )를 36개 모아 최소제곱법을 이용하여 최소 제곱 오차가 0.1보다 작은  $V_k$ 를 계산한다.

- Step 9: 과정 8에서 구한 의 범위가 분기점의 왼쪽인지 아니면 바깥쪽인지를 계산하여 고장 거리를 찾아준다.

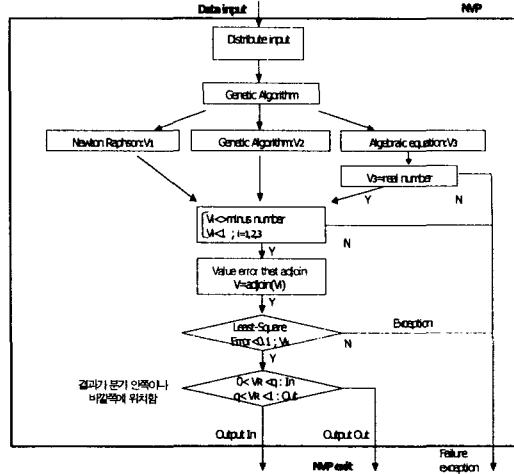


그림 2. 거리 방정식을 계산하기 위한 NVP 모델

### 3. 실험 과정 및 결과

본문의 알고리즘을 검증하기 위해 EMTP를 사용하여 154[kV] 병행회선길이 25[km]의 계통에서 사고모의를 하였다. 선로전체를 1로 하는 [pu]단위를 사용한다. GA의 조건은 300세대를 재생성하고 염색체의 길이는 15비트를 사용하여 최소화 문제로 적용했다.

제안한 NVP 모델은 3개의 verison을 사용하여 각각의 종류는 Newton-Raphson, Genetic Algorithm, Algebraic Equation이다. Newton-Raphson의 초기값은 Genetic Algorithm의 1세대 배양으로 계산된 결과를 가지고 적용한다. Algebraic Equation은 MABTAL의 함수를 이용하여 계산했다. Genetic Algorithm은 세대 수와 배양 조건에 따라 결과에 영향을 준다. 실험에 사용한 조건은 최대 반복 횟수(maxgen)는 100에서 100단위로 증가하여 300까지 3단계로 나누어 계산했고, 집단 크기(popszie)는 50에서 50단위로 증가하여 150까지 3단계로 나누어 계산했으며 염색체의 길이(lchrom)=10으로 고정하여 총 실험은 36가지의 각기 다른 실험 조건의 데이터를 가지고 9번 반복하여 계산했다. 이전 세대의 최적 개체를 다음 세대에 전달 해주기 위해 엘리트 전략(elitist strategy)을 적용했다.

제안한 NVP 모델의 성능을 평가하기 위해 기존의 논문에서 계산한 Newton-Raphson과 제안한 NVP의 결과를 비교했다. 또한 계산 도중에 발생할 수 있는 오류에 강인성을 투정하기 위해 임의로 노이즈를 추가하여 강인성에 대한 테스트를 실행했다. 강인성 테스트의 환경은 사고 위치가 0.5[pu] 지점이고, 분기점의 위치는 0.8[pu] 지점일 때 Genetic Algorithm의 조건이 maxgen=100, popsize=60, lchrom=10 이다.

그림 3을 보면 각기 다른 실험 조건으로 테스트한 시뮬레이션의 수가 36가지이다. 각 시뮬레이션 조건에 따라 나오는 NVP 결과와 이전 논문에 결과인 NR 결과를 비교한 그래프이다. 이를 보면 거의 비슷한 결과를 보이나 전체적으로 NVP의 결과가 더 좋게 나온다. 명확한 비교를 위해 36번의 실험 오차를 누적한 누적 오차를 비교하면 NR=0.693924이고 NVP=0.68757가 되어 약 0.006354(0.91%)만큼 NVP의 오차가 적게 나온다. 그래프를 보면 분기점의 위치가 멀어질수록 오차가 커지고 고장 저항의 크기가 클수록 오차가 커짐을 알 수 있으나 시뮬레이션 수가 13에서 24까지 보면 균곡이 나타나는데 이는 분기점 왼쪽에 고장이 발생한 경우와 분기점 바깥쪽에 고장이 발생한 경우 사용된 방정식이 다르기 때문이다.

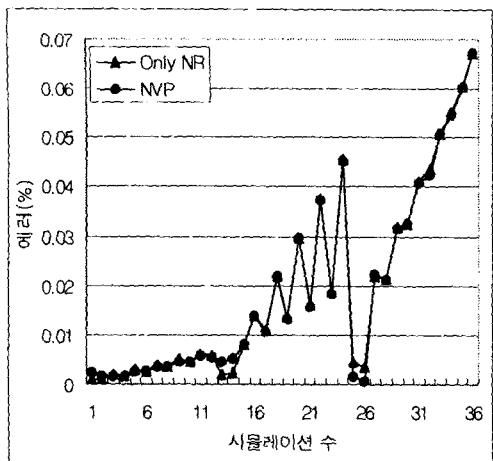


그림 3 NVP 결과와 오직 NR만 사용한 결과에 따른 에러 비교

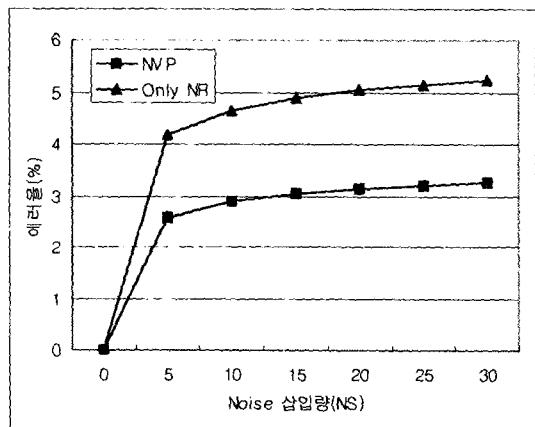


그림 4 강인성 테스트를 위한 노이즈 삽입에 따른 에러 비교  
(삽입 조건:  $a_2 = a_2 + a_2 \times NS$ )

그림 4는 시스템의 소프트웨어가 이상하여 오류가 계산식에 포함되거나 수수께끼 사과의 반율로에 의한 작은 오차가 누적되어 발생할 수 있는 계산상의 오류로부터 강인성을 테스트한 데이터이다. 임의의 노이즈를 상수  $a_2$ 에 다음 수식으로 추가하여 테스트했다.

$$\text{노이즈 삽입 방법: } a_2 = a_2 + a_2 \times NS$$

노이즈의 삽입에 영향을 주는 NS는  $a_2$ 의 배수로 증가시켜 고장 거리 방정식의 노이즈를 추가한다. 강인성 결과를 보면 기존 Newton-Raphson만 사용한 경우보다는 NVP를 적용한 경우가 오류에 강인함을 알 수 있다. NVP 모델을 적용한 제안한 방법이 기존 방법보다 약 40% 더 정확한 결과를 나타낸다. 오류에 대한 강인성은 고장위치에 대한 신뢰를 주는 요소로 시스템에 무엇보다 중요한 요소이다.

#### 4. 결론

전력 에너지 누漏 안정하고 신뢰할 수 있도록 고장에 대한 빠른 대처가 필요하다. 송전선로의 빠른 수리를 위해 정확한 고장 위치를 찾는 것이 중요하다. 본 논문에서는 기존 논문의 알고리즘을 이용하여 정확한 고장 위치를

찾는 방법으로 NVP 모델을 적용한 새로운 알고리즘을 제안했다.

제안한 방법은 기존 논문보다 약 0.91% 에러가 줄었으며 내부 오류에 대한 강인성은 약 40% 더 강인했다. 또한 Newton-Raphson 방법은 계산 초기에 미지수의 초기값을 결정해줘야 하는 문제가 있는데 이를 해결하기 위해 Genetic Algorithm을 사용하여 해결했다. 제안한 NVP 계산 모델은 나수의 계산된 결과를 비교하여 보다 정확한 결과를 선택할 수 있어 예상치 못한 결과의 오류를 수정할 수 있는 강인성을 부여해줄 수가 있게 되었다.

#### 감사의 글

본 연구는 과학기술부 및 한국과학재단의 ERC 프로그램을 통한 지원으로 이루어졌으며 이에 감사를 드립니다.

#### 【참고 문헌】

- [1] T. Nagasawa, M. Abe, N. Otsuzuki, T. Emura, Y. Jikihara, M. Takeuchi, "Development of a New Fault Location Algorithm for Multi Terminal Two Parallel Transmission Lines", IEEE Trans. on PWRD, Vol. 7, No. 3, pp.1516-1532, 1992
- [2] M. S. Sachdev, R. Agarwal, "A Technique for Estimating Transmission Line Fault Locations from Digital Impedance Relay Measurements", IEEE Trans. on PWRD, Vol. 3, No. 1, pp.121-129, 1988
- [3] A. Wiszniewski, "Accurate Fault Impedance Locating Algorithm", IEE Proceedings, Vol. 130, pt. C, No. 6, pp.311-314, 1983
- [4] R. K. Aggarwal, D. V. Courcy, A. T. Johns, A. Kalam, "A Practical Approach to Accurate Fault Location on Extra High Voltage Teed Feeders", IEEE Trans. on PWRD, Vol. 8, No. 3, pp.874-883, 1993
- [5] 박홍규, 이명수, 이재규, 유석구, "송전선에서의 고장점 저작사고시 고장거리 추정에 관한 알고리즘", 대한전기학회 학계학술대회 논문집, pp.1363-1365, 1999
- [6] 권태원, 강상희, 최연송, 이승재, "1회선 분기점을 갖는 병행 2회선 송전선로의 고장점 표정 알고리즘", 대한전기학회 추계학술대회 논문집, pp. 116-118, 1999년 11월
- [7] Laura L. Pullum, "Software Fault Tolerance Techniques and Implementation", Artech House, 2001
- [8] 전강규, "유전알고리즘과 그 응용", 교우사, 2000
- [9] Sang Hee Kang, Seung Jae Lee, Young Jin Kwon, Yong Cheol Kang, "A Fault Location Algorithm for Parallel Transmission Line with a Teed Circuit," Power Engineering Society Summer Meeting, 2001, IEEE , Volume: 2, 15-19 July 2001 Pages:921 - 926 vol.2
- [10] Randell, B., "Software Structure for Software Fault Tolerance," IEEE Transactions on Software Engineering, 1, 2, June 1975, pp. 220-232.
- [11] Avizienis, A., and Chen, L., "On the Implementation of N Version Programming for Software Fault Tolerance During Execution," Proceedings 1st IEEE International Computer Science Applications Conference, Chicago, Illinois, November 1977, pp. 149-155.
- [12] Meliarr Smith P.M., B.Randell, "Software reliability: the role of programmed exception handling," SIGPLAN Notices 12(3), 1977, pp. 95 - 100.
- [13] Kim K.H. "Distributed execution of recovery blocks: an approach to uniform treatment of hardware and software faults, Proc. 4th International Conference on Distributed Computing Systems," IEEE Computer Society Press, 1984, pp. 526 - 532.
- [14] Yau S.S., R.C.Cheung, "Design of Self Checking Software," Proc. Int. Conf. on Reliable Software, 1975, pp. 450 - 457.