

객체지향기법을 이용한 3상 배전 계통 조류계산 Software의 개발

염재필, 최정환, 김광호
강원대학교 전기공학과

Development of Three Phase Power Flow Analysis Software using Object-oriented Programming in Distribution System

Jae-Pil Eom, Jung-Hwan Choi, Kwang-Ho Kim
Dept. of Electrical Engineering Kangwon National University

Abstract - This study developed the window based power flow analysis software for distribution system. This system was developed by using Visual C++ 6.0 in Microsoft Windows 98 environment; therefore, it utilized many advantages of windows based programming for providing user-friendly interface. The backward/forward sweep algorithm is implemented in this software for power flow analysis. Especially, this software includes the newly developed classes of complex function, matrix function, and power flow calculation procedure, so it has high flexibility for enhancement and modifications.

1. 서 론

최근 들어 객체 지향 기법은 프로그램의 신뢰성과 재사용성, 유지보수의 용이성 등의 장점으로 인해 소프트웨어 개발에 있어서 효과적으로 사용되고 있다. 본 연구에서는 이러한 객체 지향 기법의 장점을 활용하여 소프트웨어를 개발하여, 향후 소프트웨어를 손쉽게 개선할 수 있는 환경을 만들고, 또한 윈도우 프로그래밍 기법을 사용하여 사용자가 보다 편리하게 접근할 수 있는 GUI(Graphic User Interface) 환경을 갖는 배전 계통 3상 조류계산 소프트웨어를 개발하였다.

본 소프트웨어에서는 배전계통 3상 조류계산을 위해 방사상의 구조를 갖는 배전계통(radial distribution system)에 적합한 Backward/forward sweeping 알고리즘이 사용되었다. 본 소프트웨어는 Microsoft Windows 98 환경 하에서 MS Visual C++ 6.0으로 개발되었기 때문에 윈도우 프로그램의 다양한 장점을 그대로 활용할 수 있다. 특히 각종 파라미터의 입력 및 배전 계통의 구성을 마우스 이벤트를 이용한 Drag and Drop Play로 수행하도록 하여, 효율적이고 신속한 해석이 가능하게 하였다. 또한 업데이트 효율성을 증진시키기 위해 각종 수학적 클래스와 GUI(graphic user interface) 클래스 그리고 조류계산 알고리즘을 체계적으로 객체화하였다.

2. 배전 계통 조류계산 알고리즘의 정식화

조류계산의 문제는 어떤 특정한 단자나 모선의 조건에서 계통에 관한 전력의 흐름과 전압에 대한 계산으로 구성되므로, 이러한 계산은 궁극적으로 각 모선의 전력을 결정하기 위하여 모선전압의 크기와 위상각을 결정하는 문제로 귀결된다.

본 논문에서 제안된 3상 배전 계통 조류계산 알고리즘은 기존의 방법과 달리 Admittance Matrix를 쓰지 않고 KVL(Kirchhoff's Voltage Law), KCL(Kirchhoff's Current Law)를 이용하여 크게 3단계에 걸쳐 수행된다.

2.1 Line Modeling

배전 선로는 송전선로에 비해 길이가 짧고 연가등을 통한 선로정수의 평행이 이루어 질 수 없으므로 Mutual 성분이 고려되어야 한다.

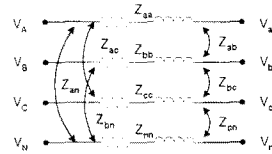


그림 1. 3상 상호 임피던스

대칭 좌표법을 이용해 위의 선로를 아래와 같이 전개하고, 중성점의 전압을 0이라고 보면 식(2)와 같이 모델링이 된다.

$$\begin{bmatrix} V_a \\ V_b \\ V_c \\ V_n \end{bmatrix} = \begin{bmatrix} Z_{aa} & Z_{ab} & Z_{ac} & Z_{an} \\ Z_{ab} & Z_{bb} & Z_{bc} & Z_{bn} \\ Z_{ac} & Z_{bc} & Z_{cc} & Z_{cn} \\ Z_{an} & Z_{bn} & Z_{cn} & Z_{nn} \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ I_c \\ I_n \end{bmatrix} \quad (1)$$

$$Z_L = \begin{bmatrix} Z_{aa} - \frac{(Z_{an})^2}{Z_{nn}} & Z_{ab} - \frac{(Z_{an}) \cdot (Z_{bn})}{Z_{nn}} & Z_{ac} - \frac{(Z_{an}) \cdot (Z_{cn})}{Z_{nn}} \\ Z_{ab} - \frac{(Z_{bn}) \cdot (Z_{an})}{Z_{nn}} & Z_{bb} - \frac{(Z_{bn})^2}{Z_{nn}} & Z_{bc} - \frac{(Z_{bn}) \cdot (Z_{cn})}{Z_{nn}} \\ Z_{ac} - \frac{(Z_{cn}) \cdot (Z_{an})}{Z_{nn}} & Z_{bc} - \frac{(Z_{bn}) \cdot (Z_{cn})}{Z_{nn}} & Z_{cc} - \frac{(Z_{cn})^2}{Z_{nn}} \end{bmatrix} \quad (2)$$

$$= \begin{bmatrix} Z_{11} & Z_{12} & Z_{13} \\ Z_{21} & Z_{22} & Z_{23} \\ Z_{31} & Z_{32} & Z_{33} \end{bmatrix}$$

2.2 Step1

각 노드에서의 부하전류를 구한다.

이때 루트노드를 포함한 모든 노드는 전압의 크기 1, 위상각은 0으로 초기화한다.

$$\begin{bmatrix} I_{ia}^{(k)} \\ I_{ib}^{(k)} \\ I_{ic}^{(k)} \end{bmatrix} = \begin{bmatrix} (S_{ia}/V_{ia}^{(k-1)})^* \\ (S_{ib}/V_{ib}^{(k-1)})^* \\ (S_{ic}/V_{ic}^{(k-1)})^* \end{bmatrix} - \begin{bmatrix} Y_{ii} & & \\ & Y_{ib} & \\ & & Y_{ic} \end{bmatrix} \begin{bmatrix} V_{ia}^{(k-1)} \\ V_{ib}^{(k-1)} \\ V_{ic}^{(k-1)} \end{bmatrix} \quad (3)$$

2.2 Step2

각 노드에 유입되는 브랜치 전류를 합산하기 위해 Backward Sweep을 한다. 그 방향은 노드의 끝단에서부터 루트노드로 거슬러 올라가는 방향이며, 식(4)와 같은 수식으로 표현될 수 있다.

2.3 Step3

각 노드에서의 전압을 업데이트 하기 위해 Forward Sweep을 한다. 이번에는 루트 노드부터 맨 말단 노드로 수행하게 되며, 식(5)와 같은 수식으로 표현된다.

2.4 조건식 판별

식(6)과 같은 식에 의해 각 노드에서 변화된 전압의 차이를 구한 후 이 값이 허용치보다 크면 Step1-Step3

까지 반복 수행하고 작으면 계산을 종료한다.

$$\begin{bmatrix} I_{jBra-a}^{(k)} \\ I_{jBra-b}^{(k)} \\ I_{jBra-c}^{(k)} \end{bmatrix} = \begin{bmatrix} I_{ia}^{(k)} \\ I_{ib}^{(k)} \\ I_{ic}^{(k)} \end{bmatrix} + \sum_{m \in M} \begin{bmatrix} I_{mBra-a}^{(k)} \\ I_{mBra-b}^{(k)} \\ I_{mBra-c}^{(k)} \end{bmatrix} \quad (4)$$

$$\begin{bmatrix} V_{(i-1)a}^{(k)} \\ V_{(i-1)b}^{(k)} \\ V_{(i-1)c}^{(k)} \end{bmatrix} = \begin{bmatrix} V_{ia}^{(k)} \\ V_{ib}^{(k)} \\ V_{ic}^{(k)} \end{bmatrix} - \begin{bmatrix} Z_{11} & Z_{12} & Z_{13} \\ Z_{12} & Z_{22} & Z_{23} \\ Z_{13} & Z_{23} & Z_{33} \end{bmatrix} \begin{bmatrix} I_{(j)Bra-a}^{(k)} \\ I_{(j)Bra-b}^{(k)} \\ I_{(j)Bra-c}^{(k)} \end{bmatrix} \quad (5)$$

$$\Delta V_{ia}^{(k)} = V_{ia}^{(k)} - V_{ira}^{(k-1)}$$

$$\Delta V_{ib}^{(k)} = V_{ib}^{(k)} - V_{irb}^{(k-1)}$$

$$\Delta V_{ic}^{(k)} = V_{ic}^{(k)} - V_{irc}^{(k-1)}$$

전체적인 흐름도는 그림2와 같다.

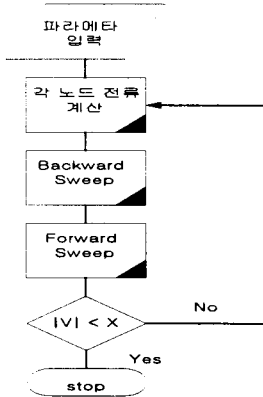


그림2. Backward/Forward Sweep Method Flow Chart

3. 배전 계통 조류계산 S/W의 구현

3.1 클래스의 구축

객체지향 기법은 데이터와 그것을 조작하는 함수를 동일 그룹으로 하는 추상 데이터형을 정의하고, 이렇게 정의된 변수를 객체로 간주하여 프로그램을 구성하는 기법으로써 상속(Inheritance), 캡슐화(Encapsulation), 다태성(Polymorphism)과 같은 여러 가지 특징을 가지고 있다.

따라서 객체지향 기법을 사용한 프로그램에서는 이런 객체를 클래스로써 얼마나 효율적으로 구성하였는가 하는 것이 프로그램의 성능을 나타내는 기준이 된다고 할 수 있다. 본 논문에서는 수학적 클래스, GUI 클래스, 조류계산 알고리즘 클래스 등 여러 가지 클래스를 사용하였지만 지면관계상 위의 3가지 클래스만 소개하도록 하였다.

3.1.1 수학적 클래스 구축

조류계산 과정에서 필요한 복소수 연산, 행렬 연산과 같은 수학적 함수들을 하나의 클래스로 만들어 객체화하고 상속을 통해 필요한 클래스에서 사용함으로써 불필요한 프로그램 코딩상의 중복을 피하고 다른 프로그램에도 다시 사용할 수 있는 Reusability를 증진시켰다. 그 상속 계층도의 일부를 나타낸 것이 그림3이다.

3.1.2 GUI 클래스 구축

GUI클래스는 크게 각 디바이스들의 Topology 데이터를 저장하기 위한 클래스인 CShape와 대화상자로부터 파라메타를 입력받고 저장하기 위한 각 디바이스의

클래스등이 있다.

또한 계통의 구성 및 파라메타의 입력을 마우스 이벤트를 이용해 구현하기 위한 함수를 CSymbolView 라는 클래스에 코딩하였다.

그림 4는 각 디바이스들의 파라메타를 입력받기 위한 대화상자이다.

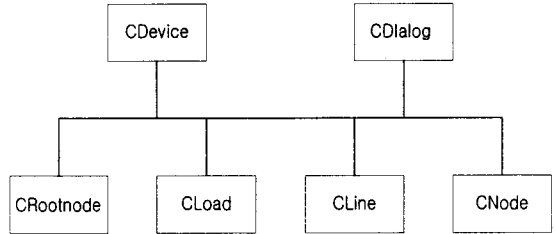
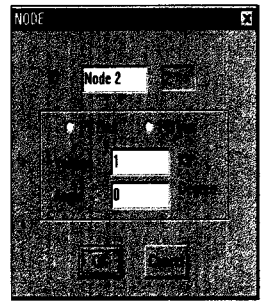
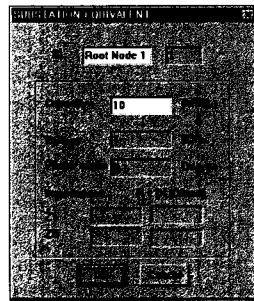
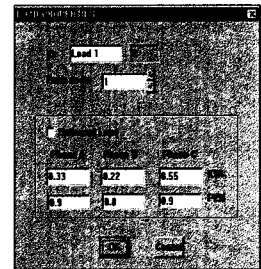
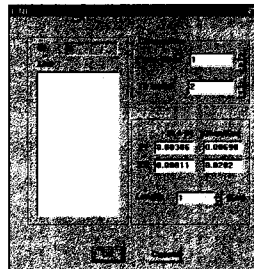


그림3. 수학적 모델 클래스 CDevice의 상속 계층도



(a) CRootnode 대화상자

(b) CNode 대화상자



(c) CLine 대화상자

(d) CLoad 대화상자

그림4. 각 디바이스의 대화상자 클래스

3.1.3 Backward/Forward Sweep Method의 클래스 구축

3상 배전선 조류계산을 하기 위한 알고리즘인 Backward/Forward Sweep 클래스는 복소수·행렬의 계산에 대한 멤버 함수가 있는 CDevice를 상속받아 사용하며, 그 자체가 캡슐화 되어 있기 때문에 다른 프로그램에서도 독립적으로 쓰일 수 있다. 클래스의 계층도는 그림5와 같다.

3.2 GUI(graphic user interface) 구현

그림6은 본 논문에서 개발한 Software의 Main Frame으로써 각각의 객체는 마우스 이벤트에 의해 구성되어지고 파라메타의 입력 또한 그림4에의 대화상자를 통해 이루어진다.

이때 사용되는 마우스 이벤트는 왼쪽, 오른쪽 마우스 버튼의 down 과 up 그리고 이동에 관한 것이며 이들을 통해 "Drag And Drop Play" 와 같은 Visual하고 편

리한 Interface가 구현된다.

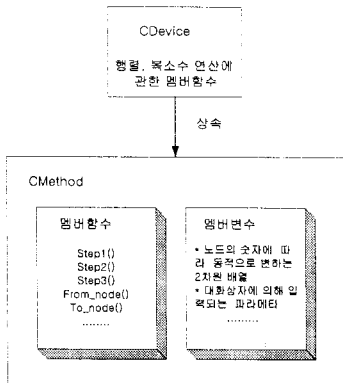


그림5. CMethod 클래스 계층도

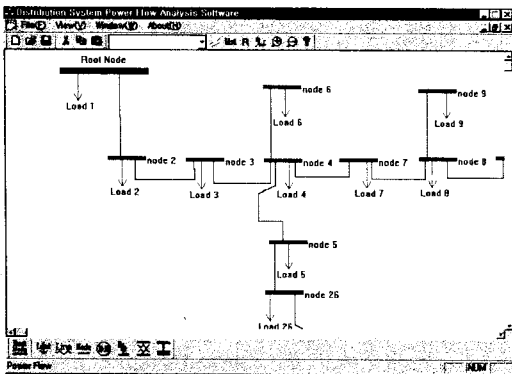


그림6. Software의 메인 프레임

4. 시뮬레이션

그림7은 3상으로 구성된 36개의 노드를 가진 배전 선로 모델 계통으로써, 시뮬레이션에 사용된 라인은 ACSR 160 mm², 1800 mm 길이의 완금이 설치된 Y 결선 다중접지 모델이다. 1회선용 선로 data는 표1과 같고, 부하 데이터는 표2와 같다.

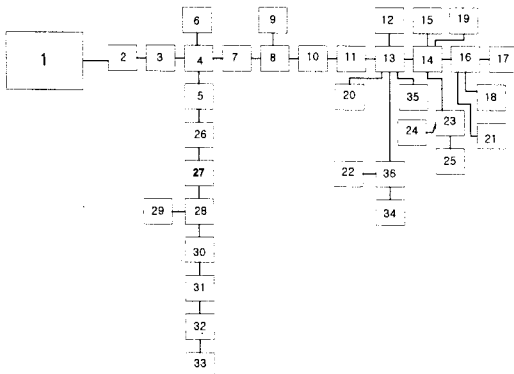


그림7. 배전선로 샘플 계통도

영상	0.0811+j0.202
정상	0.0386+j0.0698
역상	0.0386+j0.0698

표1. 라인 임피던스값 (PU)

Node NO.	A상(KVA/PF)	B상(KVA/PF)	C상(KVA/PF)
1	0	0	0
2	101.66/0.9	98.3/0.9	101.66/0.9
3	43.3/0.9	40.1/0.9	43.3/0.9
4	115/0.9	104/0.9	115/0.9
5	126.66/0.9	119.30/0.9	126.66/0.9
6	135/0.9	1350/0.9	135/0.9
...
31	19.18/0.9	16.4/0.9	19.18/0.9
32	2.25/0.9	2.25/0.9	2.25/0.9
33	9.6/0.9	9/0.9	9.6/0.9
34	14.15/0.9	14.33/0.9	14.15/0.9
35	19.7/0.9	16.33/0.9	19.7/0.9
36	13.7/0.9	12.5/0.9	13.7/0.9

표2. 부하 데이터(100MVA, 100MVAR, 22.9KV base)

그림8 결과에서 보여 주듯이 루트 노드로부터 멀어질수록 전압이 감소하는 것을 알 수 있고, 특히 A상과 C 상은 부하가 거의 비슷하여 그 결과가 거의 일치하며, B 상은 다른 상에 비해 부하가 작아 다른 상에 비해 약간 높은 전압을 나타내는 것을 볼 수 있다.

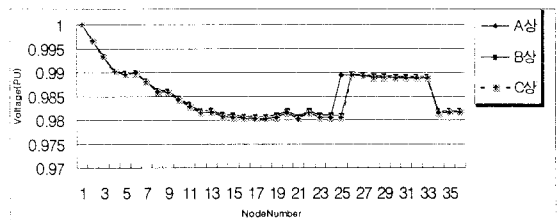


그림8. 샘플 계통 시뮬레이션 결과

5. 결 론

본 논문에서는 객체지향기법을 이용해 3상 배전선 조류계산 Software를 윈도우즈용 프로그램으로 개발했다.

특히, 배전계통은 송전계통과 달리 3상이 불평형하기 때문에 3상 각각에 대한 해석이 필요함에 따라 방사형 배전계통에 적합한 Backward/Forward Sweep Method를 3상으로 확장하여 제안하였고, 각각의 디바이스는 CDevice 라는 객체를 상속받았기 때문에 복소수·행렬연산에 관한 멤버 함수를 공유할 수 있어 프로그램을 효율적으로 코딩할 수 있었다.

또한 프로그램을 객체화했기 때문에 DR(Dispersed Resource), Shunt capacitor와 같은 향후 업데이트해야 할 디바이스들을 추가할 경우 프로그램의 다른 부분에 영향을 주지 않고 모듈형식으로 쉽게 추가될 수 있어 효율적이고 신뢰성 있는 업데이트 여건을 보장받을 수 있다.

(참 고 문 헌)

- [1] Carol S. Cheng, "A three-phase power flow method for real-time distribution system analysis", IEEE Transactions on power systems, Vol.10, No. 2, May 1995
- [2] 김건중의, "배전계통에서 객체지향기법을 이용한 삼상조류계산", 대한전기 학회 하계학술대회 논문집, 제 47권 11호, pp.1817-1823, 1998.11
- [3] Bjarne Stroustrup, "The C++ Programming Language : Second Edition", Addison-Wesley, pp. 55-292, 1991
- [4] Glenn W. Stagg, Ahmed H. El-Abiad, "Computer Methods in Power System Analysis", McGraw-Hill, 1968.