

대규모 전력계통의 해석을 위한 소프트웨어 개발

김 한준*, 박형국, 황갑주
울산대학교 전기전자정보시스템공학부

Development of Software for Large Scaled Power System Application

Han-Jun Kim, Hyung-Kug Park, Kab-Ju Hwang
School of Electrical Engineering, University of Ulsan

Abstract - 전력계통의 조류계산과 고장계산은 전력계통의 운용과 해석 및 계획의 기반이 되는 산법으로 계속 연구 개선되고 있으며, 최근에는 컴퓨터 분야의 발달에 따라 소형 컴퓨터 레벨에서도 대규모 계통을 해석할 수 있는 소프트웨어들이 개발되고 있다.

본 연구를 통하여 퍼스널 컴퓨터 레벨에서 대규모의 실제계를 해석할 수 있는 소프트웨어를 개발하였다. 산법에는 조류계산과 고장계산 모두 어드미턴스행렬을 기반으로 하였으며, 스파스 프로그래밍 기법을 도입함으로써 대규모 실제계의 해석을 가능하게 하였다.

개발된 소프트웨어는 이미 전력회사에서 그 성능을 검증받은 PTI사의 소프트웨어인 PSS/E와의 입력자료 호환성을 갖도록 하였고 사용자의 접근을 쉽게 할 수 있는 그래픽 사용자 인터페이스 환경을 구현하였다. 개발된 산법의 성능을 검증하기 위하여 PSS/E 20모선 샘플계통, IEEE 샘플계통 및 한국전력 실제계에 적용하였다.

1. 서론

전력계통에서 평상시 각 선로나 설비에 흐르는 전력의 분포를 파악하거나, 선로나 설비에 고장이 발생하였을 경우 전압 및 전류의 변화를 산정하는 것은 현재 계통의 운용은 물론 새로운 설비계획을 검토하는데 매우 중요하다. 정상상태에서 발전기나 모선의 전압이나 선로에 흐르는 전력의 양을 계산하는 것을 전력조류계산(Power Flow Calculation)이라고 하며, 계통의 각 선로나 설비에 1선 지락 또는 선간 단락 등의 고장이 발생했을 경우 고장전류 및 계통 각 부분의 전압과 전류를 계산하는 것을 고장계산(Fault Calculation)이라고 한다. [1]

그동안 컴퓨터 하드웨어 기술의 발달로 퍼스널 컴퓨터 레벨에서도 수 천 모선의 대규모 계통을 해석할 수 있게 되었으며, 최근에는 계산속도와 수렴성에 못지 않게, 개발된 소프트웨어의 사용상 편의성이나 유지보수 문제에 관한 연구도 활발하게 이루어지고 있다. [2]

전력계통해석을 필요로 하는 전력회사의 실무부서에서는 대부분 PTI사의 PSS/E를 사용하고 있는데, 주된 이유는 빠른 속도와 다양한 기능을 지원할 뿐만 아니라 많은 사용자들에 의해 그 계산결과가 검증되었기 때문이다. [3] 그런데 최근의 컴퓨터 사용환경에서 볼 때, 제 2세대 언어인 FORTRAN으로 출발하여 변형시켜온 PSS/E는 윈도우즈 사용환경에 익숙한 사용자에게는 사용상 불편한 점이 많다. [3]

따라서 본 연구에서는 PSS/E에서 실무자가 많이 사용하는 기능을 손쉽게 접근할 수 있도록 개발하였다.

개발된 소프트웨어는 기존의 PSS/E 사용자의 자료호환성을 고려하여 입력자료의 형태를 같도록 설계하였으며, 대상계통의 규모나 계산속도를 PSS/E에 필적하도록 개발하였다.

이를 위해 먼저 윈도우 환경에 적합한 프로그래밍 언어로서 Visual Basic을 사용하였고, 소프트웨어의 메모리 문제와 속도 문제를 좀더 효율적으로 개선하기 위해서 전력계통의 시스템 행렬을 구성할 때 비영요소만 배열에 저장하는 스파스 프로그래밍 기법을 도입하였고, 자코비안 행렬 및 어드미턴스 행렬의 역변환 과정에서 발생하는 생성항을 감소하기 위해서 모선의 서열화 기법을 도입하였다. [4]

개발된 소프트웨어의 성능을 평가하기 위해서 IEEE 샘플계통과 우리나라의 실제계통 및 PSS/E에서 검증용으로 제공하는 20모선 샘플 계통을 대상으로 테스트를 하였으며, 그 결과를 PTI사의 PSS/E와 비교하여 제량들을 확인하였다.

2. 조류계산

전력조류를 해석하기 위한 산법으로는 가우스-자이델법, 뉴턴-랩슨법, 고속 분할법 등을 들 수 있다. 본 연구에서는 대규모 계통의 적용을 위하여 고속분할법을 근간으로 개발하였다. [5]

전력계통의 전력방정식을 편차방정식의 형태로 적으면

$$\Delta P_i = P_i^{SP} - V_i \sum_{k=1}^N V_k (G_{ik} \cos \theta_{ik} + B_{ik} \sin \theta_{ik})$$

(1)

$$\Delta Q_i = Q_i^{SP} - V_i \sum_{k=1}^N V_k (G_{ik} \sin \theta_{ik} - B_{ik} \cos \theta_{ik})$$

이 된다. 뉴턴-랩슨법에서는 이들 전력편차를 없애기 위하여

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} H & N \\ M & L \end{bmatrix} \begin{bmatrix} \Delta \theta \\ \Delta V \end{bmatrix}$$

(3)

와 같은 수정 행렬방정식으로 표현한다. 식 (3)에서 부행렬 H, N, M, L을 Jacobian이라고 하며 각각 편미분 $\frac{\partial P_i}{\partial \theta_i}$, $\frac{\partial P_i}{\partial V_i}$, $\frac{\partial Q_i}{\partial \theta_i}$, $\frac{\partial Q_i}{\partial V_i}$ 을 구하여 대입함으로써

전압과 위상각의 수정량을 구하게 된다. 뉴턴-랩슨법은 빠른 수렴특성을 가지는 반면 부행렬을 저장하는데 많은 메모리가 필요하며 매 반복시 Jacobian의 역행렬을 구해야 하는 단점이 있다.

고속분할법은 뉴턴-랩슨법에서 Jacobian 행렬의 저장과 역행렬 계산의 부담을 줄이기 위해 식 (3)을 다음과 같이 분할하여 구하게 된다.

$$[\Delta P] = [B'] [\Delta \theta]$$

(4)

$$[\Delta Q] = [B''] [\Delta V]$$

(5)

위 식에서 [B']과 [B'']은 각각 [H]와 [L]과 같은 구조의 스파스(Sparse) 행렬일 뿐만 아니라 정수값을 가지는 행렬이다. 따라서 조류계산 과정에서 한번만 구

해 놓으면 되므로 계산시간이 매우 빠르다.

3. 고장계산

고장계산은 전력 시스템 해석의 중요한 분야 중 하나이며, 여러 형태의 고장이 발생하였을 경우 각 모선의 전압과 선로전류를 계산함으로써 이 결과를 바탕으로 보호계전기의 정정, 차단기의 차단용량 결정, 각 계통 기기의 과전류 등을 검토하는데 필요하다.

3상 단락고장과 같은 3상 평형고장은 일반적인 조류계산과 같이 한 상에 대한 계산을 통해 해석이 가능하지만, 1선지락, 선간단락 등과 같은 대부분의 고장은 각상이 불평형을 이루기 때문에 이러한 불평형 고장계산을 위해 대칭좌표법을 도입하여 불평형고장을 계산한다. [2]

3.1 모선 고장

고장이 발생 하였을 경우 테브넨 등가 회로를 구성하고 대칭 좌표법을 이용 고장점에 흐르는 전류를 구함으로써 각 모선의 전압 및 선로의 조류를 구하게 된다.

표 1은 각 고장별 고장점에 흐르는 a 상에 대한 대칭성분 전류를 나타낸 것이다.

표 1 고장 유형에 따른 고장점 전류

고장 종류	대칭성분	고장점 전류
3상단락	Positive	$I^+(F) = \frac{V_f}{Z_{ii} + Z_f}$
1선지락	Positive	$I^+(F) = \frac{V_f}{Z_{ii}^+ + Z_{ii}^- + Z_{ii}^0 + 3Z_f}$
	Negative	$I^-(F) = \frac{V_f}{Z_{ii}^+ + Z_{ii}^- + Z_{ii}^0 + 3Z_f}$
	Zero	$I^0(F) = \frac{V_f}{Z_{ii}^+ + Z_{ii}^- + Z_{ii}^0 + 3Z_f}$
선간단락	Positive	$I^+(F) = \frac{V_f}{Z_{ii}^+ + Z_{ii}^- + Z_f}$
	Negative	$I^-(F) = -\frac{V_f}{Z_{ii}^+ + Z_{ii}^- + Z_f}$
	Zero	$I^0(F) = 0$
2선 지락	Positive	$I^+(F) = \frac{V_f}{Z_{ii}^+ + \frac{Z_{ii}^- (Z_{ii}^0 + 3Z_f)}{Z_{ii}^0 + Z_{ii}^- + 3Z_f}}$
	Negative	$I^-(F) = -I^+(F) \left[\frac{Z_{ii}^0 + 3Z_f}{Z_{ii}^0 + Z_{ii}^- + 3Z_f} \right]$
	Zero	$I^0(F) = -I^+(F) \left[\frac{Z_{ii}^-}{Z_{ii}^0 + Z_{ii}^- + 3Z_f} \right]$

3.2 선로 고장

선로 상에서 고장이 발생하였을 경우 그림 1과 같이 고장지점을 계통에 새로운 더미모선(Dummy Bus)으로 추가하고, 고장지점을 중심으로 양 모선의 거리에 비례하여 분할된 선로에 새로이 임피던스를 부여한다.

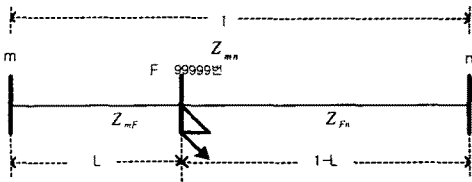


그림 1. 선로고장의 예

$$Z_{mF} = L \times Z_{mn}, \quad Z_{Fn} = (1-L) \times Z_{mn} \quad (6)$$

이때 새로 분할된 선로의 각 임피던스는 식 (6)과 같이 표현되며, 추가되는 더미모선의 모션번호로 99999를 부여한다. 더미모선을 포함한 새로운 계통이 구성되면 고장종류에 따라 모션고장과 같은 방법으로 고장계산이 이루어진다.

3.3 자동 연속모션 고장

임의의 모션에서 고장이 발생하면 고장 전류의 영향에 대해서 가장관심이 있는 부분은 고장 모션과 직접적으로 연결된 모션과 선로이다. 연속 모션 고장계산은 전체 또는 선택된 일부 모션에 대해서 각각의 모션에 대한 고장을 계산한 후 직접적으로 연결된 모션까지의 고장결과를 출력한다.

자동 연속모션 고장계산은 계통의 여러 위치(모션)에서 일련의 고장을 수행함으로써 각각의 모션과 연결된 선로가 다양한 조건을 가질 때 고장 사고에 대해 1차적으로 연결된 선로들에 미치는 고장의 영향을 알 수 있게 해준다. 개발된 소프트웨어에서는 연속모션 고장계산은 3상 단락 고장과 1선지락 고장에 대해서 수행되며, 전체 모션 또는 의미의 모션들을 선택할 수 있는 옵션을 제공하도록 하였다. 그림 2는 자동 연속모션 고장계산 입력 화면을 나타낸 것이다.

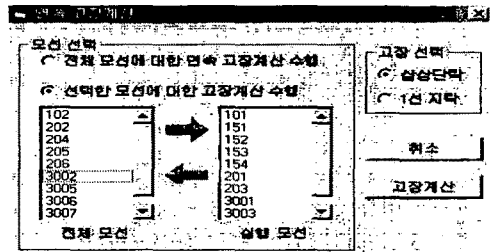


그림 2. 자동 연속고장 입력 화면

3.4 고장 계산의 간략화

고장계산의 간략화는 대상계통을 단순화하여 보다 빠른 고장 계산을 할 수 있도록 사용자가 다음과 같은 옵션을 선택할 수 있도록 개발하였다.

1. 고장 전 전압의 크기를 1, 위상은 0°로 설정.
2. 모든 부하의 크기는 0으로 설정.
3. 발전기 전력출력은 0으로 설정.
4. 모든 Shunt 요소들은 0으로 설정.
5. 선로 충전용량은 0으로 설정.
6. 변압기 권선비는 1로 설정.

4. 프로그래밍 기법

4.1 스퍼스 프로그래밍

수백개 이상의 모션을 갖는 실 계통을 대상으로 조류계산을 수행할 경우, 시스템 행렬은 보통 5(%) 미만의 비영요소로 이루어진 희소행렬의 특성을 가지고 있다. 따라서 이런 특성을 이용, 비영요소만을 저장함으로써 저장 장소와 계산시간의 향상을 꾀할 수 있다.

일반적으로 많이 사용하는 배열변수 값의 저장기법은 행렬요소 중 비영요소의 위치정보(행, 열)를 저장하고, 찾고자 하는 행렬요소는 저장된 위치정보를 이용한다.

모선수가 n 개, 선로의 수가 1개인 계통의 경우를 예로 들면, 행렬의 2행에 존재하는 비영요소는 표 2.a에서 $Idx(2)$ 에서 $Idx(3)-1$ 의 위치(3에서 5)에 존재하고 그 요소들의 열 정보 및 그 값은 표 2.a의 행정보를 바탕으로 표 2.b $Loc(3)$ 에서 $Loc(5)$ 사이에서 찾을 수 있다.

표 2.a 행렬의 대각요소 및 비대각요소의 인덱스

BusNo	G_{kk}	B_{kk}	Idx
1	G_{11}	B_{11}	1
2	G_{22}	B_{22}	3
3	G_{33}	B_{33}	6
⋮	⋮	⋮	⋮
$n+1$			$n+1$

표 2.b 행렬의 비대각요소 및 열 정보

Loc	Col	G_{km}	B_{km}
1	2	G_{12}	B_{12}
2	5	G_{15}	B_{15}
3	3	G_{23}	B_{23}
4	4	G_{24}	B_{24}
5	5	G_{25}	B_{25}
6	4	G_{34}	B_{34}
⋮	⋮	⋮	⋮
$n+1$			

4.2 모선 서열화

조류계산 및 고장계산 과정에서 자코비안 행렬과 어드미턴스 행렬의 삼각화가 필요하게 된다. 그런데 이 과정에서 발생하는 생성항으로 인하여, 스퍼스 기법을 사용함에도 불구하고, 상당히 많은 저장장소가 필요하게 되고 계산시간이 늘어나는 문제가 발생한다. 따라서 계수행렬의 삼각화 과정에서 발생하는 생성항을 최소화하기 위해서 계통 모선의 순서를 바꿀 필요가 있는데 이를 모선 서열화(Bus Ordering)라고 한다. Tinney 등은 모선 서열화를 위한 3가지 방법을 제안한 바 있다. [4] 즉,

[기법 1] 각 모선에 연결된 선로의 수가 가장 작은 모선의 순서(오름차순)로 모선 번호를 재배열한다.

[기법 2] 소거의 각 단계에서, 모선에 연결된 선로의 수가 가장 작은 모선의 순서로 모선 번호를 재배열한다. 이때 연결된 선로의 수가 같은 모선의 경우 모선번호는 임의로 지정한다.

[기법 3] 각 단계의 소거의 결과, 모선에 연결된 선로의 수가 가장 작은 모선이 먼저 모선번호를 부여받도록 재배열한다. 연결된 선로의 수가 같은 모선의 경우 모선번호는 임의로 지정한다.

본 연구에서는 Tinney의 두 번째 기법을 모선의 서열화 기법의 기반으로 프로그래밍 하였다. 그림 3은 IEEE14 모선에 대한 모선 서열화 결과를 나타낸 것이다. 그림에서 0은 삼각화 이전 행렬의 비영요소를 나타내며 x는 행렬의 삼각화로 생성되는 생성항을 나타낸다. 행렬의 삼각화 과정중 모선 서열화를 적용하지 않은 경우는 새로이 생성되는 항이 42개, 적용하였을 경우는 8개로 많은 차이를 보임을 알 수 있다.

1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0

1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0

그림 3.a Ordering 비 적용 시 삼각화 결과 그림 3.b Ordering 적용 시 삼각화 결과

5. 사용자 인터페이스

사용자가 조류계산과 고장계산을 보다 손쉽게 접근하기 위해서는 효과적인 사용자 인터페이스의 설계가 중요하다. 본 연구에서는 Visual Basic을 이용하여 편리한 그래픽 사용자 인터페이스를 구현하였다. 그림 4와 5는 조류계산 및 고장계산의 화면을 나타낸 것이다.

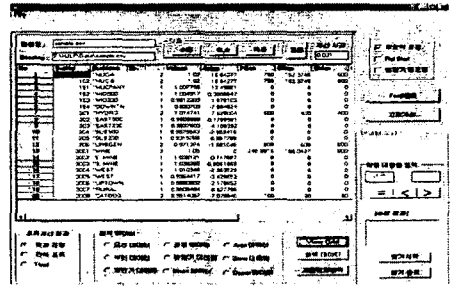


그림 4. 조류계산 주 화면

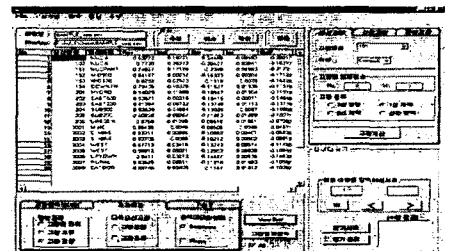


그림 5. 고장계산 주 화면

다른 응용프로그램의 활용을 위하여 입력 데이터 및 계산결과 데이터는 화면은 물론 파일로 출력되며, 데이터 편집을 용이하게 하기 위하여 화면에서 직접 편집을 할 수 있을 뿐만 아니라 엑셀과 연계할 수 있다.

대규모 계통을 다룰 때는 데이터의 양이 방대해져서 원하는 데이터를 검색하는 일이 어려워진다. 이를 위해 원하는 데이터 또는 설정된 범위의 데이터를 찾아주는 탐색기능도 지원하도록 하였다.

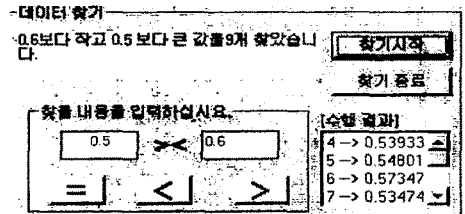


그림 6. 데이터 찾기 화면

그림 7은 프로그램의 수행과정을 나타낸 흐름도이다.

6. 사례연구

본 연구를 통하여 개발된 소프트웨어의 성능을 검증하기 위하여 IEEE-14모선, IEEE-57모선, IEEE-118 모선, PSS/E 20 모선 샘플계통과 한국전력 805 모선에 대해서 적용하였으며, 그 결과를 PSS/E 와 비교하였다.

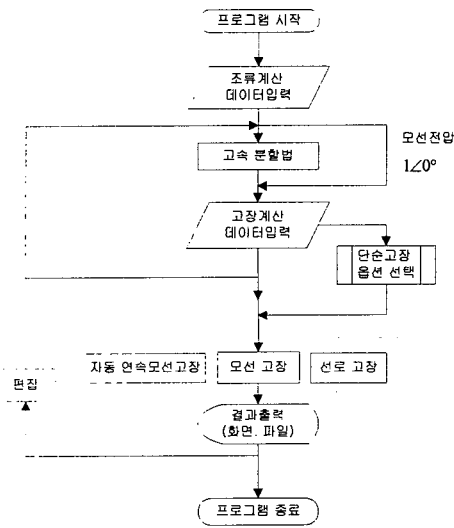


그림 7. 프로그램의 수행과정

그림 8은 IEEE 샘플계통을 대상으로 전력조류계산을 수행하였을 경우의 수렴회수 및 최대전압편차를 나타낸 것이다. 이들 샘플계통의 경우는 모두 5회의 반복회수 내에서 수렴이 이루어지고 있음을 알 수 있다.

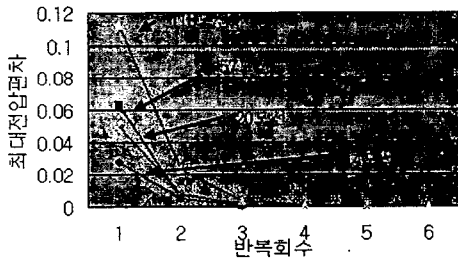


그림 8. 샘플계통의 수렴특성

표 3은 805모선의 실제계에 대해 본 연구에서 개발된 프로그램과 PSS/E의 고속 분할법 결과를 비교 분석한 것이다. 조류계산 결과의 화면 출력을 고려할 경우 사용자가 조류계산결과 인식까지 걸리는 시간은 PSS/E보다 10배 이상 빠르다. 표 3에서 알 수 있듯이 본 연구에서 개발된 프로그램의 성능이 실제계 적용에도 유효함을 알 수 있다. 표 3 과 표 4 의 'PSC'는 본 연구에서 개발된 프로그램을 의미한다.

표 3. PSS/E 와의 조류계산 결과 비교

적용 소프트웨어	계산시간 [sec]	모든데이터 결과 출력 시간[sec]	반복 회수	최대편차
PSS/E	0.5미만	20~25	4회	0.0001
PSC	0.672	1~2	6회	0.0002

표 4는 PSS/E 20모선 샘플계통에 대한 고장 종류별,

고장 후 전압을 비교한 것이며, 152번 모선에서 고장 발생, 고장점 임피던스는 0.01pu 일 경우, 3004번 모선에서의 고장 후 전압을 비교하였다.

표 4. 고장종류별 고장계산 결과 비교

		Positive		Negative		Zero	
		Re	Im	Re	Im	Re	Im
3상 단락	PSC	0.219	0.007	----	----	----	----
	PSS/E	0.219	0.007	----	----	----	----
	편차	0	0	----	----	----	----
1선 지락	PSC	0.833	-0.077	-0.177	-0.037	-0.225	-0.128
	PSS/E	0.832	-0.078	-0.178	-0.038	-0.222	-0.127
	편차	0.001	0.001	0.001	0.001	<u>0.003</u>	0.001
2선 지락	PSC	0.551	-0.039	0.333	-0.046	0.152	0.116
	PSS/E	0.551	-0.040	0.332	-0.046	0.150	0.115
	편차	0	0.001	0.001	0	0.002	0.001
선간 단락	PSC	0.722	-0.147	0.287	0.107	0	0
	PSS/E	0.722	-0.147	0.287	0.107	0	0
	편차	0	0	0	0	0	0

7. 결론

본 연구를 통하여 실 계통 해석이 가능한 소프트웨어를 개발하였으며, 연구내용을 요약하면 다음과 같다.

- 1) PSS/E와 같은 입력 데이터 양식을 사용함으로써 소프트웨어의 호환성을 갖게 하였다.
- 2) 조류계산과 고장계산 모두 어드미턴스 행렬을 기반으로 개발하였으며, 시스템 행렬의 비영요소만의 저장, 모선의 서열화 및 스퍼스 프로그래밍 기법을 도입함으로써 대규모 실 계통의 해석을 가능하게 하였다.
- 3) 편리한 그래픽 사용자 인터페이스를 통해 자료의 편집과 계산 수행 등 사용자의 접근을 용이하게 하였다.
- 4) 샘플계통과 실 계통의 사례를 대상으로 PSS/E의 결과와 비교함으로써 해의 유효성과 소프트웨어의 성능을 검증하였다.

전력시스템의 해석을 보다 다각적이고 포괄적으로 수행하기 위하여 조류계산 및 고장계산의 다양한 기능의 추가를 진행중이며, 안정도 해석 모듈을 개발하는 등의 연구도 계획하고 있다.

(참고 문헌)

- [1] M. A. PAI, Computer Techniques in Power System Analysis, Tata McGraw-Hill Publishing Company Limited, New Delhi, 1979.
- [2] Hadi Saadat, "Power System Analysis", McGraw-Hill book, Co, 1999.
- [3] "PSS/E Program Operation Manual", PTI, 2000
- [4] W. F. Tinney, "Compensation Method for Network Solution By Optimally Ordered Triangular Factorization", IEEE Transaction on Power Apparatus and Systems, Vol. PAS-91, pp123-127, Jan/Feb. 1972.
- [5] B. Stott, O. Alsac, "Fast Decoupled Load Flow" IEEE Trans. on PAS, Vol. 93, pp859-869, May/June, 1974
- [6] Chun-Liang Lee, Nanmming Chen, "Sparse Vector Method Improvements via Minimum Inverse Fill-In Ordering", IEEE Transaction on Power Systems, Vol. 8, no. 1, pp239-245, Feb. 1993.