

객체지향기법을 적용한 하이브리드 과도안정도 해석

박 지 호
경북대학교

Hybrid Transient Stability Analysis Using Object-oriented method

Park Ji Ho
Kyungpook National University

Abstract - In this paper, we simulate power system transient stability using object-oriented programming(OOP). OOP is a more flexible method than procedural programming. There are several advantages in dynamic system simulation using OOP. We also calculate critical fault clearing time using energy functions for detailed models.

1. 서 론

최근의 컴퓨터 소프트웨어의 발달은 전력계통 분야가 당면한 문제점을 해결할 수 있는 실마리를 제공하고 있다. 최근 전력계통 분야에 대한 적용이 활발히 연구되고 있는 객체지향 기법은 프로그램의 유연성과 기능면에서 기존의 방법들보다 월등함이 확인되고 있다. 본 논문에서는 하이브리드 방법을 이용하여 전력계통의 과도 안정도해석을 위한 발전기, 여자기 등의 모델링기법에 대하여 논하고, 발전기의 상세모델의 시간모의를 통한 에너지함수를 구성하고 과도 안정도 여유를 계산한다. 시간영역 모의 방법의 장점은 모델링의 제한이 없다는 점이다. 또한 안정도 현상의 자세한 고찰을 위한 시스템의 모든 양들의 시간응답을 볼 수 있다. 하지만 시간영역모의 방법은 미분방정식을 풀기 위하여 스텝바이스텝으로 적분하기 때문에 매우 느리다. 그래서 안정도 여유를 계산하기 위해서는 많은 시간이 필요하다. 또한 안정도 한계를 계산하기 위한 시스템의 민감도 정보가 부족하다. 반면 에너지함수를 이용한 직접법은 안정도 지수를 만들 수 있는 장점을 지니고 있다. 따라서 본 논문에서는 발전기의 상세모델을 이용한 시간 영역의 모의와 에너지 함수를 이용한 안정도 여유를 계산하는 하이브리드 방식을 이용하여 안정도 해석을 한다.

2. 본 론

2.1 시스템의 객체지향적 모델링

본 논문에서 제시한 기법을 이용한 전력계통 과도안정도 해석을 위한 모델링은 그림1과 같이 4단계로 구성된다.

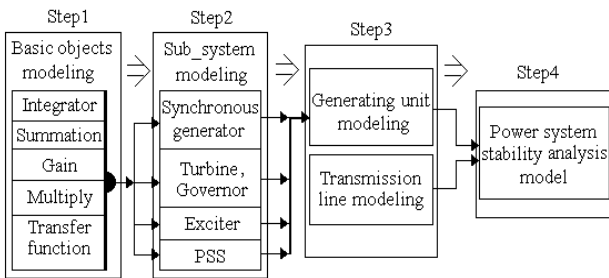


그림 1 객체지향기법을 사용한 전력계통 안정도 모델링 단계

2.1.1 기본 객체의 모델링

기본소자들을 객체로 표현하기 위해 C++언어의 class 키워드를 사용하여 객체(object)로 구성한다. 덧셈기, 곱셈기, 적분기, 이득기 클래스가 중요 기본소자이고 다음은 덧셈기의 예이다.

```
class objectinfo{
private: char* name,id; int num;
public: char* getname(); char getid();
};
class Summation:public objectinfo
```

```
private:double sum_data;
public:void sum(int,double*);
void setInit(double*);
};
```

2.2 서브시스템의 모델링

위에서 구성된 기본 객체들을 사용하여 안정도 해석을 위한 발전기, 여자기, 안정기, 조속기 등을 모델링할 수 있다.

1) 발전기 모델링

그림 2는 동기발전기 이축모델의 과도 유기전압에 대한 블록선도로서 각 블록에 대하여 객체 이름을 주었다.

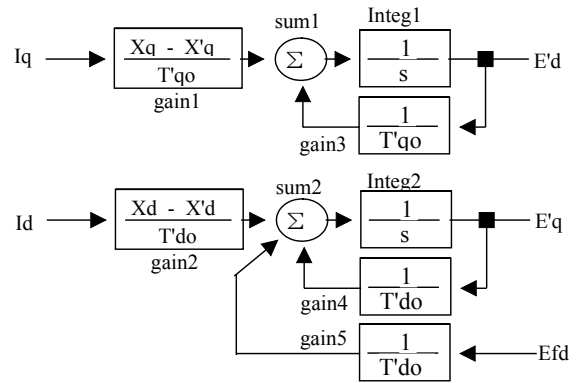


그림 2 동기발전기 이축모델

gain1~gain5는 이득기 객체, sum1, sum2는 덧셈기 그리고 Integ1, Integ2는 적분기 객체이다. 표1은 블록선도를 사용하여 구성된 각 객체간의 입력력 관계를 나타낸다. 이 입력정보를 사용하여 실제 계산에서 필요한 객체간의 연결순서가 이진트리 구성기법으로 구해진다.

표 1 동기발전기 이축 모델의 객체정보

Table 1 Object information of two axis model

object num.	name	input objects(num.)
0	gain1	Iq(외부연결객체출력)(Ex.)
1	gain2	Id(외부연결객체출력)(Ex.)
2	gain3	E'd (7)
3	gain4	E'q (8)
4	gain5	E'd(외부연결객체출력)(Ex.)
5	sum1	gain1(0) gain3(2)
6	sum2	gain2(1) gain4(3) gain5(4)
7	Integ1	sum1(5)
8	Integ2	sum2(6)

그림 2와 표1의 객체연결은 이진트리구조로 구성할 수 있다. 만 들어지는 트리의 개수는 적분기의 개수와 같다. 외부 연결트리는 여자기, 선로를 나타내는 트리와의 연결을 위한 것이다. 즉 외부의 여자기와 선로의 모델에 상관없이 트리로서 연결할 수 있다. 마찬가지로 발전기 모델에 상관없이 이와 같이 트리를 구성할 수 있다. 전달함수 객체는 각 객체의 파라미터에 따라 그 내부에서 기본 객체들의 연결관계가 자동적으로 성립되도록 구성하였다.

2) 여자기 모델링

그림 3은 IEEE type 1 여자기 모델을 나타낸 것이다. 4개의

전달함수 Transfer1~Transfer4는 모두 Transfer클래스의 객체이다. 이들 객체들은 각 파라미터가 다르므로 내부의 연결상태가 달라진다.

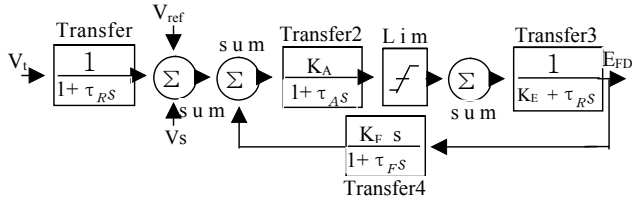


그림3 IEEE type 1 여자 시스템

2.3 전체 시스템 모델링

발전기와 선로와의 상호 연결은 각 객체가 가지는 버퍼를 통한 데이터 교환에 의해 이루어진다. 각 객체는 자신의 작업의 결과를 버퍼에 저장하고 상대 객체에게 넘겨주거나 상대적으로부터 결과를 얻어 오기만 하면 된다. 객체간의 연결에는 데이터의 교환만 있을 뿐 타 객체의 내부구조까지 알 필요가 없다.

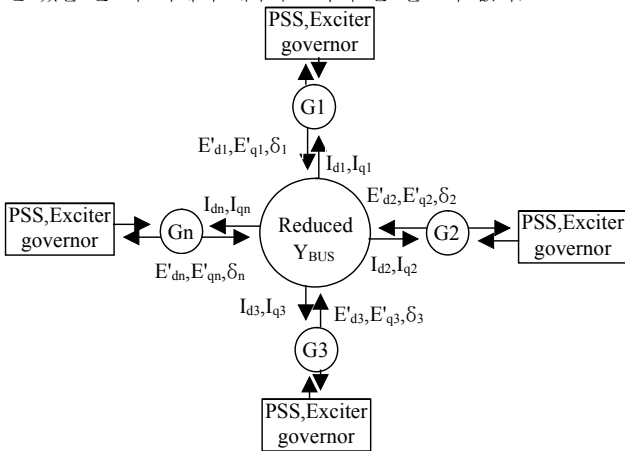


그림 4 다수기 시뮬레이션을 위한 객체연결

2.4 하이브리드 과도 안정도 해석

그림4와 같이 다수기 시스템의 상세모델을 이용하여 사고시스템의 사고궤도를 시간영역에서 계산하고 그 결과를 사고제거시스템의 에너지를 함수에 적용하여 사고 후의 에너지를 계산한다. COI (Center of Inertia) 표기법에서 시스템의 수학적 모델링은 다음과 같다.

$$M_i \frac{d\omega_i}{dt} = P_i - P_{ei} - \frac{M_i}{M_T} P_{COA} = f_i \quad (1a)$$

$$\frac{d\theta_i}{dt} = \tilde{\omega}_i, i = 1, N \quad (1b)$$

$$P_i = P_{mi} - E_i^2 G_{ii} \quad (1c)$$

$$P_{COA} = \sum_{i=1}^N P_i - P_{ei} \quad (1d)$$

$$P_{ei} = \sum_{j=1, j \neq i}^N C_{ij} \sin(\theta_i - \theta_j) + D_{ij} \cos(\theta_i - \theta_j) \quad (1e)$$

여기서 θ_i, ω_i =rotor angle and speed of Machine i
in the Centre of inertia(COI) frame

M_i = inertia constant of machine i

P_{M_i} =mechanical input of Machine i

E_i = internal voltage magnitude of Machine i

G_{ij}, B_{ij} = elements of the internal Y-matrix

$C_{ij}, D_{ij} = E_i E_j B_{ij}, E_i E_j G_{ij}$

$$V = V_{KE}(\omega) + V_{PE}(\theta) \quad (2a)$$

$$V_{KE}(\omega) = \frac{1}{2} \sum_{i=1}^N M_i \omega_i^2 \quad (2b)$$

$$V_{PE}(\theta) = - \sum_{i=1}^N P_i (\theta_i - \theta_i^s)$$

$$- \sum_{i=1}^{N-1} \sum_{j=i+1}^N [C_{ij} (\cos \theta_{ij} - \cos \theta_{ij}^s) + \int_{\theta_i^s + \theta_j^s}^{\theta_i + \theta_j} D_{ij} \cos \theta_{ij} d(\theta_i + \theta_j)] \quad (2c)$$

여기서 θ^s 는 stable equilibrium point

하이브리드 방법을 적용하여 임계사고제거시간을 계산하는 방법은 다음과 같다.

주어진 시스템에 대하여 사고발생 전후의 평형점을 계산한다. 사고 제거후의 시스템에 대하여 식(2a)-(2c)의 에너지 함수를 구성한다. 발전기를 상세 모델링한 시스템에 대하여 그림 4와 같은 객체연결구조를 가지고 사고시스템을 시뮬레이션한다. 즉 사고를 지속시킨 상태에서 사고궤도를 계산한다. 계산 사고궤도를 이용하여 식(2a)-(2c)의 에너지 함수를 계산한다. V_{PE} 가 최대가 되는 점을 찾고 이점의 에너지를 갖는 식(2a)의 V의 시간을 찾으려면 이것이 임계사고제거시간이다. 그림 5는 3기9모선시스템[4]의 시뮬레이션 결과이다. 회전자 각의 초기값은 $\delta_1 = 3.58^\circ, \delta_2 = 61.1^\circ, \delta_3 = 54.2^\circ$ 이고 사고후는 COI 표기법에서 $\delta_1 = -0.1649\text{rad}, \delta_2 = 0.4987\text{rad}, \delta_3 = 0.2344\text{rad}$ 이다.

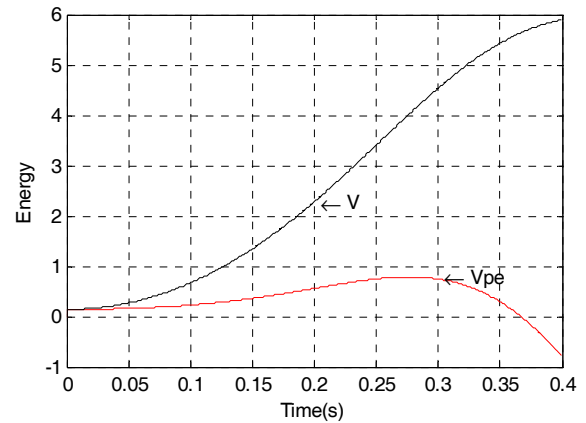


그림 5 총에너지와 포텐셜에너지

3. 결 론

전력계통은 여러 가지 동적시스템이 상호 결합되어 있기 때문에 보다 보편적이고 유연한 시뮬레이션 기법이 필요한데 기존의 프로그램방식은 시스템의 변화에 능동적이지 못하기 때문에 소프트웨어의 유지 및 재사용에 있어서 많은 제약이 있다. 이러한 이유로 객체지향기법을 적용하여 보다 유연한 방법을 구현하였다. 또한 시간영역의 시뮬레이션과 안정도 직접해석법인 에너지 함수를 결합시킨 하이브리드 방법을 이용하여 발전기 임계고장 제거시간을 구할 수 있다.

[참고 문헌]

- [1] M. Foley, A. Bose, W. Michell and A. Faustini, "An Object Based Graphical user Interface for Power System," IEEE Trans. on Power Systems. Vol. 8 No. 1, Feb. 1993, pp. 97-104
- [2] S. Lu, E. Swidenbank and B. W. Hogg, "An Object-oriented Power Plant Adaptive Control System Design Tool," IEEE Trans. on Energy Conversion, Vol. 10, No. 3, September. 1995, pp. 600-605
- [3] B. Hakavik and A.T. Holen, "Power System Modelling and Sparse Matrix Operations Using Object-oriented Programming," IEEE Trans. on Power Sytems, vol. 9, No. 2, May 1994
- [4] P.M. Anderson and A.A. Fouad, *Power System Control and Stability*, Iowa State University Press, 1977.
- [5] T. Athay, R. Podmore and S. Virmani, "A Practical Method For The Direct Analysis of Transient Stability".