

송전 손실 계산 패키지 개발을 위한 UML 적용 연구

이찬주\* · 신중린  
건국대학교 전기공학과

박종배  
인양대학교 전기·전자공학과

An Application of the Unified Modeling Language  
to develop the Transmission Loss Calculation Package

Chan-Joo Lee\* · Joong-Rin Shin  
Konkuk University  
jrshin@kkucc.konkuk.ac.kr

Jong-Bae Park  
Anyang University  
jbaepark@ayoc.anyang.ac.kr

**Abstract** - This paper presents an application of UML(Unified Modeling Language) to develop the Transmission Loss Calculation Package (TLCP) and shows a feasibility of UML design for the power system analyzing software. In the rapidly restructuring electric power industry, it is required the development of efficient and transparent transmission pricing mechanism. This should be based to calculate an accurate transmission loss to prompt a fair competition and to give unbiased signals to all the market players. For those reasons, it is necessary to develop a software system for transmission loss calculation.

In this paper, we design the TLCP based on UML, and describe using Rational Rose 2000.

1. 서 론

계획과 운영이 수직통합된 전력산업은 최근에 독점체제에서 구조개편을 추진해왔다. 전력산업 구조개편에 따라서 경쟁적 전력시장이 형성되고 이러한 경쟁시장에서 시장참여자는 경쟁을 할 것이다. 전력산업 구조개편은 모든 시장참여자에게 송전망 접속에 차별을 둘 수 없으며, 발전부분의 분할을 통한 경쟁시장을 추진중에 있으며 CBP(Cost-Base Generation Pool)에 적용될 입찰, 급전 그리고 정산의 시장운영 규칙이 만들어졌다. 우리나라는 지역별로 전력과 수요가 편중되어 있어, 송전선로 혼잡현상이 심각한 상태이며 각 지역별로 발전 및 수요로 인한 송전손실의 차이가 크다. 따라서 전력에너지 자원의 효율적인 배분과 공정경쟁을 촉진하기 위해서는 이의 배분방법이 중요하며 완전경쟁 시장을 위한 효율적이고 투명한 송전비용 메카니즘의 개발이 필요하다. 이러한 결과로 모든 시장참여자들에게 적절하고 경제적인 가격신호의 제공과 공정한 경쟁을 위한 정확한 송전손실 계산이 대단히 중요하다. 이러한 이유로 송전손실 계산 시스템의 개발이 필요하다.

본 논문에서 우리는 송전손실 계산 시스템에 적용할 UML(Unified Modeling Language)기반의 한계손실 계수계산 패키지(Marginal Loss Factor Calculation Package)를 설계하였고, 전력계통 소프트웨어 설계에 잠재적인 가능성을 제시하였다.

2. 본 론

2.1 UML

지난 수세기 동안, 많은 진보된 종류의 소프트웨어 엔지니어링 기법들이 전력계통 공학분야에 적용되어 왔었다. 구조적 프로그래밍 기법과 객체지향 개념은 소프트웨어의 효율성을 증가시켜 기존 프로그래밍의 한계성을 극복하였고 재사용성과 신뢰성을 증가시켰다. 그러나 소프트웨어의 복잡화와 대형화에 따른 유지보수의 어려움과 시스템의 설계 측면에서 많은 문제가 발생했다[1].

UML(Unified Modeling Language)은 1990년 후반에 Rational Software사의 Booch, Jacobson, 그리고 Rumbaugh에 의해서 개발되었다[1,2,3]. UML은 소프트웨어 중심의 산출물을 가시화하고, 명세화하며, 구축하고, 그리고 문서화하는데 사용되는 비주얼 프로그래밍 언어이며, 넓은 범위에 걸쳐서 복잡하고 규모가 큰 시스템을 효율적으로 관리할 수 있는 언어이다. UML은 많은 서로 다른

형태의 복잡한 시스템을 정의하는데 사용되며, 또한 요구사항 기술서에서부터 완성된 시스템의 테스트까지 전반적인 시스템의 개발에 적용된다. 또한 점진적이고 반복적인 개발 프로세스를 통해서 설계시 발생할 수 있는 위험요소를 최소화하고 시스템설계를 최적화한다.

2.2 한계손실계수

일반적으로 한계손실계수(MLF)는 발전기 출력 또는 부하량의 아주 작은 증분변화가 생길 때 생기는 손실의 증가 또는 감소를 나타낸다[4]. 손실을 고려한 경우, 계통내의 부하를 공급하기 위해서는 부하량과 손실량을 합한 양을 발전기들이 공급해 주어야 하므로 식(1)과 같이 표현할 수 있다.

$$\sum_{i=1}^N P_i^G = \sum_{i=1}^N P_i^L + P_{loss} \quad (1)$$

여기서,  $P_i^G$  : i지역의 발전량,  $P_i^L$  : i지역의 부하량  
 $P_{loss}$  : 계통손실,  $i$  : 개별지역

MLF는 식(2)와 같이 계산할 수 있다.

$$(MLF)_i = \frac{\Delta P_i^G}{\Delta P_i^L} = \frac{\Delta P_i^L + \Delta P_{loss}}{\Delta P_i^L} \quad (2)$$

$$= 1 + \frac{\Delta P_{loss}}{\Delta P_i^L}$$

먼저 주어진 계통상황에서 계통손실을 계산한 다음에 계통내의 전체 부하를 아주 작게 증가시킨다. MLF를 구하고자 하는 모선을 기준모선으로 조류계산을 하여 계통손실의 변화를 측정한다. 이때, 개별지역 i에서의 한계손실계수는 부하의 변화량에 대한 발전의 변화량으로 계산된다.

식(2)에서 알 수 있듯이, 부하가 집중된 지역의 MLF는 1보다 작게 나타난다. 그 이유는 부하가 집중된 지역의 발전은 전체송전량을 줄일 수 있고 손실을 줄일 수 있는 확률이 높기 때문이다. 여기에서 구한 것은 발전에 대한 한계손실물이다. 부하에 대한 한계손실물은 발전 한계손실물의 역이다.

2.3 Modeling Approach

MLFCP (Marginal Loss Factor Calculation) 개발하기 위해서, 본 논문은 다음과 같은 단계를 거쳐서 모델링에 접근하였다. 또한 Rational Rose 2000을 사용하여 모델링하였고, Visual C++로 MLFCP를 구현하였다.

2.3.1 MLFCP 분석단계

분석 단계는 문제영역에서 발생하는 메카니즘을 클래스(Class)와 객체(Object)의 추상화를 통해서 나타낸다. 분석단계에서 쓰임세도 (Use Case Diagram), 활동도(Activity Diagram), 교류도 (Interaction Diagram)를 도해하고, 시스템에 맞춘 클래스를 도출한다.

MLFC시스템 쓰임새 모델은 쓰임새 관점에서 시스템을 나타낸다. 이러한 관점은 MLFCP 개발에 전반적인 영향을 주므로 대단히 중요하다. 또한 쓰임새는 MLFCP의 논리적, 물리적 계층에도 영향을 준다. 분석단계의 MLFCP 개발과정은 처음에 쓰임새도를 통하여 MLFC 시스템의 영역을 설정하고 그림 1과 같이 도해하였다.

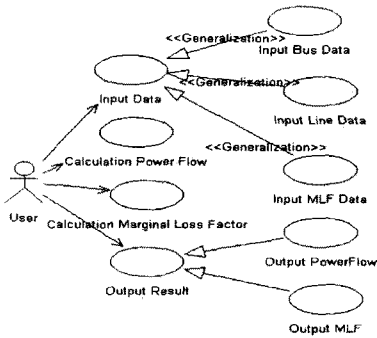


그림 1 MLFC 시스템의 쓰임새도  
Fig.1 MLFC Use-Case Diagram

Input Data, Calculation PowerFlow, Calculation MLF, Output Result같은 쓰임새는 User 액터에 의해 초기화된다. 또한 Input BusData, Input LineData, Input MLFData use-case는 Input Data 쓰임새로부터 상속을 받는다.

쓰임새도를 정의한 후, 각 쓰임새도의 이벤트 흐름을 표현하는 활동도를 정의한다. 또한, 동시에 MLFCP의 설계에서 객체와 그들간의 관계를 보여주는 교류도(순차도, 협력도)를 도해한다. 동일한 객체교류에 대해서 서로 다른 관점으로 바라보는 두 종류의 교류도가 있다. 순차도(Sequence Diagram)는 객체의 시간적인 순서에 의해서 나타나고, 협력도(Collaboration) 객체와 그들간의 링크로 나타낸다. 그림 2는 "Calculation Marginal Loss Factor" 쓰임새도에 대한 시나리오를 바탕으로 MLFC 순차도를 도해한 것이다.

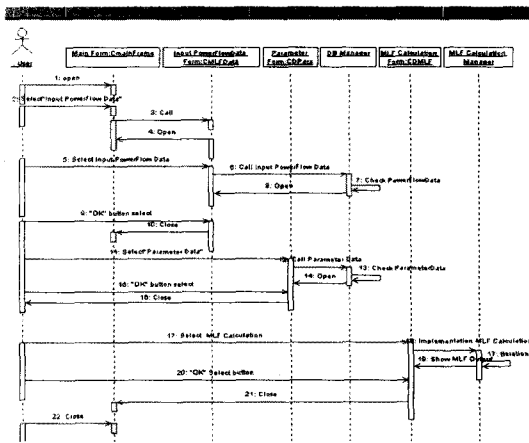


그림 2 "Calculation MLF" 쓰임새에 대한 순차도  
Fig.2 Sequence Diagram for "Calculation MLF" use case

### 2.3.2 MLFCP 설계 및 구현단계

설계단계는 분석단계에서 도출된 결과를 기술적인 해법을 통해 확장한다. 설계과정은 실제 구현환경을 감안하여 적용될 아키텍처 환경에서의 객체 및 클래스를 모델에 적용하여 분석단계 모델을 상세하게 재정의한다. 또한 추출된 클래스의 계층을 분류하고, 분류된 클래스를 구현하기 위한 알맞은 프로그래밍 언어를 선택한다. 본 논문은 MLFC 시스템을 상세 설계하기 위해 서브시스템(Subsystem) 설계, 패키지(Package)와 클래스(Class)도를 재정의하였다. 또한 분석단계의 협력도를 재정의하면서 추출된 객체들에 대해 클래스를 다시 정의하고, 객체들 간의 메시지들은 해당 클래스에서 정의한 행위(Operation)로 재정의한다. 이렇게 재정의하는 것은 이전의 분석단계의 단순한 흐름보다는 설계를 개발환경 및 분산환경까지도 고려하여 모델링을 진행하기 때문이다.

### 2.4 MLFCP 개발

정확한 한계손실계수의 계산은 경쟁적 전력 시장에서 모든 시장 참여자에게 적절한 가격신호를 제공하고, 부하에 대한 가격신호를 제공함으로써 공정한 경쟁을 가능하게 해준다. 이러한 이유로 경쟁체제하의 전력산업에서 정확한 한계손실계수를 계산하는 시스템이 필요하다. 본 논문에서는 한계손실계수 계산시스템에 필요한 패키지를 개발함으로써 재사용성과 신뢰성의 타당성을 제시한다. MLFCP에 포함된 각각의 클래스도는 Visual C++로 구현하였고, Data 패키지는 한진 설계통을 고려하여 모델링하였다.

본 논문에서 개발한 MLFCP는 아래의 그림 3과 같이 나타내었다. 각각의 패키지는 서로 의존관계(Dependency)를 갖고, Data 패키지와 MLFC 패키지는 오직 UserInterface 패키지를 통해서 정보교환이 이루어진다.

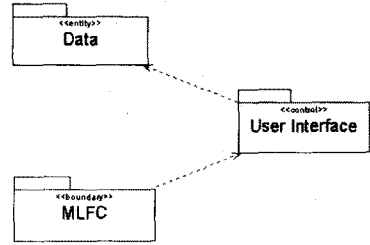


그림 3 MLFC 패키지도  
Fig.3 MLFC Package Diagram

그림 3은 MLFCP의 전반적인 개발구조를 시스템 아키텍처 관점에서 UML의 MVC (Model View Controller) 개념을 적용하여 나타내었다. 즉, 하위계층에 해당하는 Data 패키지가 변경되면 상위계층에 해당하는 MLFC 패키지와 중간계층에 해당하는 User Interface 패키지가 변경된다. 이것은 대규모 계통시스템을 분석하고 설계하는 과정에서 복잡성을 효율적으로 관리하기 위해 MLFC 패키지를 Jacobson의 계층 아키텍처(Layered Architecture)를 활용하여 나타낸 것이다[1,2].

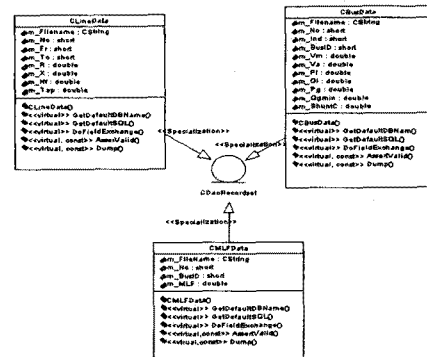


그림 4 Data 패키지의 클래스도  
Fig.4 Class Diagram in Data Package

MLFC 패키지와 의존관계를 갖는 Data 패키지는 그림 4와 같이 설계통에 적용할 수 있도록 모델링하였다. 그림 4에서 보면, 각각의 Data 클래스가 데이터베이스에 해당하는 CDaoRecordset 클래스로부터 상속을 받는다. 이것은 사용자가 가지고 있는 대상계통에 대한 데이터를 입력받을 수 있도록 모델링하였다.

MLFCP의 클래스도에서 한계손실계수가 조류계산의 결과에 의존한다는 점을 고려하여 그림 5와 같이 각각의 서브시스템 패키지에 의존관계를 주었다. 여기서, 각각의 서브시스템은 시스템 인터페이스를 가지고 있고 이것에 의해 행위 오퍼레이션이 제공된다.

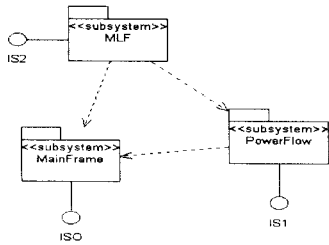


그림 5 MLFC 패키지의 서브시스템 다이어그램  
Fig.5 MLFCP Subsystem Package Diagram

시스템의 계층구조를 패키지로 정의하고나면, 패키지별로 분석단계에서 추출된 클래스에 대하여 각각의 속성 및 오퍼레이션 (Operation)을 상세 설계한다. PowerFlow 패키지에 포함된 클래스 다이어그램은 그림 6에 나타내었다.

PowerFlow 협력도로부터 유도된 MLFC PowerFlow 클래스도에는 각각의 CPowerFlowView, CProgressBar, CPowerFlowDoc, CPowerFlowApp와 같은 클래스가 포함되어있다.

CProgressBar, CPowerFlowDoc, CPowerFlowApp 클래스는 각각 CProgressBar, CDocument, CWinApp 클래스로부터 상속 받는다. 그리고 CPowerFlowView 클래스는 Visual C++에서 기본적으로 제공하는 CDaoFieldInfo, CDaoTableDef, CDaoTableDef 클래스와 연관관계 (Association)를 갖는다.

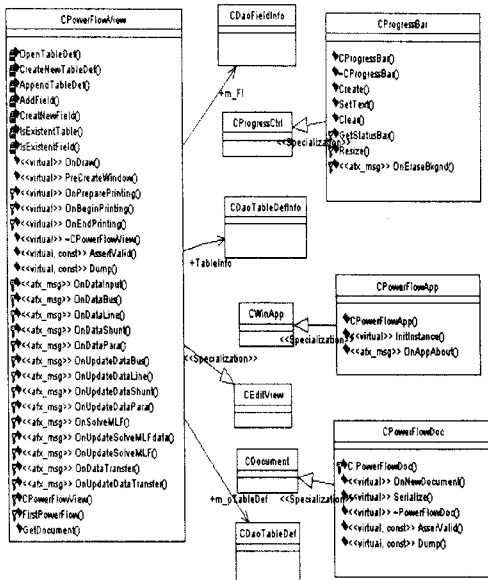


그림 6 PowerFlow 서브시스템의 클래스도  
Fig.6 Class Diagram of PowerFlow Substem

계통의 조류계산의 결과는 한계손실계수의 입력 데이터로 사용되어진다. 한계손실계수는 우선 지역내의 가상의 발전기의 위치를 지정하고 입력의 모선의 부하를 아주 작게 증가시킨후 한계손실계수 방정식에 의해서 계산된다.

PowerFlow 클래스도를 정의한 후, MLF 서브시스템에 해당하는 클래스도 그림 7과 같이 도해하였다. 국내 설계에 대한 해석을 위해서 PSS/E 조류계산 입력 데이터의 형식을 CKepco 클래스로 정의하였다. CKepco 클래스에는 한계손실계수 계산하기 위한 PSS/E조류계산 입력 데이터를 속성과 행위 통해 9개 부분으로 구분하였다: 전체 데이터, 모선 데이터, 부하 데이터, 부하 데이터, 발전기 데이터, 선로 데이터, 변압기 데이터, Area 데이터, Shunt Capacitor 데이터, Zone 데이터.

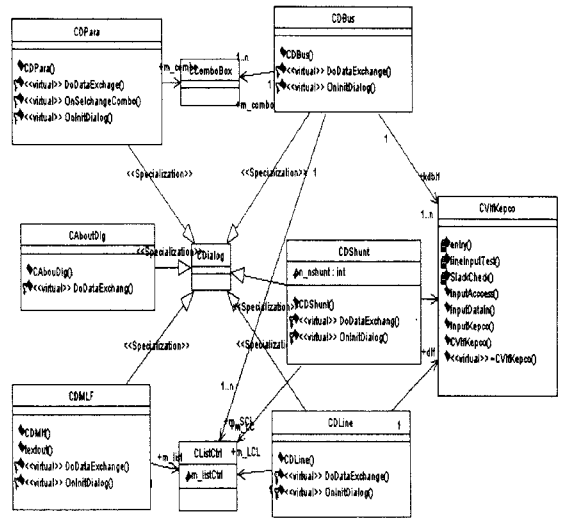


그림 7 MLF 서브시스템의 클래스도  
Fig.7 Class Diagram of MLF Subsystem

CDLine, CDSshunt, CDBus 클래스는 CKepco 클래스와 연관관계를 갖고 CKepco 클래스와 CListCtrl 클래스를 제외한 나머지 클래스들은 CDialog 클래스로부터 상속을 받는다. 조류계산을 수행한 PSS/E 데이터는 한계손실계수의 입력데이터로 CDMLF 클래스의 DoDataExchange()를 통해서 수행된다. 그리고, 한계손실계수의 계산은 CDMLF() 오퍼레이션을 통해서 수행된다.

### 3. 결론

본 논문은 대규모 전력 시스템 소프트웨어 설계에 UML (Unified Modeling Language)의 적용가능성을 검토하였으며, 구체적으로 UML 기반의 한계손실계산 패키지를 개발하였다. 본 논문에서 개발된 MLFCP의 장점은 다음과 같다.

- 대규모 전력계통 소프트웨어 시스템의 요구사항에 대해 효율적으로 관리하고, 유지·보수가 용이하다.
- 객체지향 기법으로 설계된 전력계통 교육용 프로그램이나 대규모 전력계통 소프트웨어 패키지 개발에 재사용과 신뢰성을 제공한다.

본 논문의 UML을 이용한 MLFCP 설계는 구현단계의 클래스 다이어그램까지 이루어졌다. 향후 컴포넌트 다이어그램을 통한 TLCP 시스템을 개발할 예정이다.

### [참 고 문 헌]

- [1] G. Booch, J. Rumbaugh, I. Jacobson, "The Unified Modeling Language User Guide", Addison Wesley, 1998
- [2] T. Quatrani, "Visual Modeling with Rational Rose 2000 and UML", Addison Wesley, 1999
- [3] Rational Rose, "Introduction to Rational Rose 98, Student Manual, Version 4.5", May, 1998
- [4] A.J. Wood, B.F. Wollenberg, "Power Generation, Operation and Control", Wiley, p.29-130, 1996
- [5] NEMMCO, "Treatment of Loss Factors in the National Electricity Market", Nov. 1999
- [6] H.J. Kohler, U. Nickel, J. Niere, A.Zundorf, "Integrating UML diagrams for production control systems", Proceedings of the International Conference on Software Engineering, pp. 241-251, 2000
- [7] B. P. Douglass, "Real-Time UML", Addison-Wesley, 1998