

다양한 소프트웨어 개발환경에서의 최적설계 프레임워크

염근철*, 이세정**

Design Optimization Framework on Various Software Development Environments

Ycom, K.-C.* and Lee, S.J.**

ABSTRACT

This paper concerns about how and why design frameworks for optimization should consider various software development environments such as MATLAB, VB, VBscript, Python, Tcl, PHP, Perl, and JAVA. The frameworks can be utilized by many engineers who have a basic concept about the optimization theory and/or basic knowledge about the computer programming languages. The framework will integrate a number of remote CAE tools, automatically execute them for design optimization, and have the capabilities of post-processing of data such as objective functions, state variables and design variables using a third-party spreadsheet program like Excel. The prototype framework developed in this study will be applied to various examples of optimization problems and show the validity of the proposed method of a framework implementation.

Key words : Optimization, Design framework, Excel, Software development, CAE

1. 서 론

주어진 자원과 시간내에서 제품을 설계하기 위해서는 설계 전 분야에 걸친 설계 요소들을 통합적으로 고려하여 설계가 이루어져야 한다. 단일 분야에서의 목적을 달성하기 위한 설계 최적화는 이미 오래 전부터 연구되어 왔으나, 시스템이 점차 복잡해지고 제품의 고부가가치화, 개발기간의 단축, 개발 비용의 절감 등이 요구됨에 따라 최근에는 설계 프레임워크를 이용한 최적화 적용 연구가 활발히 진행되고 있다¹⁾.

설계 프레임워크를 이용하여 최적설계를 하는 것은 원격 또는 지역 단말기에 위치한 최적설계 모듈, 데이터베이스, CAD 그리고 각종 CAE 코드와 같은 자원들을 분산 컴퓨팅 환경에서 사용자가 쉽게 연결하고, 자동 실행시켜 설계를 수행하는 과정을 말한다.

현재까지 위와 같은 기능을 수행할 수 있도록 개발된 프레임워크로는 iSIGHT²⁾, ModelCenter³⁾,

VisualDOC⁴⁾ 등이 대표적인 소프트웨어들이며, 각각의 특징을 가지고 발전하여 왔다.

하지만 이들 프레임워크들에는 몇 가지 공통된 문제점이 있다. 첫 번째 문제점은, 각 프레임워크에 내장된 모듈의 독립성이 보장되지 않는다는 것이다. 예를 들어 사용자가 iSIGHT의 최적설계 모듈만을 독립적으로 사용하고 싶지만 iSIGHT는 이러한 기능을 제공하지 않는다. 이것은 설계 프레임워크를 설계하는데 있어서 각 모듈들간의 의존성을 두었기 때문에 생기는 문제이다. 즉, 최적설계 모듈을 사용하여 최적설계를 하기 위해서는 전처리(pre-processing) 기능과 후처리(post-processing) 기능을 최적설계 모듈과 꼭 연결시켜야 하기 때문이다.

두 번째 문제점으로서 엑셀을 이용한 후처리 문제를 들 수 있다. 대부분의 설계 프레임워크는 자체적으로 개발한 GUI 형태의 후처리 기능을 내장하고 있다. 이것은 설계 프레임워크 자체에서 설계 진행 과정을 관찰하고 결과 데이터를 관리 하는데 도움이 된다. 하지만 엑셀을 이용하여 데이터를 관리하는 사용자 입장에서는 데이터를 엑셀 데이터로 변환하는 과정을 거쳐야 하므로 데이터의 양이 많을 경우 매우 번거로운 작업

*현대자동차 차량해석팀

**교신저자, 종신회원, 서울시립대학교 기계정보공학과

- 논문투고일: 2004. 02. 20

- 심사완료일: 2005. 06. 01

이 될 수 있다.

세 번째 문제점은 텍스트 입력 파일을 이용한 문제 구성이 지원되지 않는다는 것이다. 사용자가 처음 CAE 코드와 같은 공학용 소프트웨어를 사용할 때 주로 GUI로 지원되는 전처리기를 이용한다. 하지만 해당 작업 과정이 반복 될수록 전처리기의 의존도는 낮아지고 텍스트 입력파일을 직접 만들어 작업을 진행하게 된다. 이러한 사용자의 성향은 설계 프레임워크에서도 똑같이 적용될 수 있다. 즉, 설계 프레임워크를 이용하여 최적설계를 수행하기 위해 문제 구성을 하는데, 작업 초기에는 프레임워크에서 지원하는 GUI를 이용하지만 반복 작업이 계속 되어 설계 프레임워크에 익숙해 지면 GUI보다는 사용자가 직접 텍스트 형태의 입력 파일을 만들어 설계 환경을 구성하고 싶어한다.

본 연구에서는, 위에서 언급한 바와 같이, 현재 개발되어 사용중인 설계 프레임워크들의 공통된 문제점을 개선하기 위하여, MATLAB^[5], Python^[6], VB^[7], VBS^[8], Tcl^[9], PHP^[10], Perl^[11], Java^[12]의 8개 프로그램 언어 환경에서 사용할 수 있는 최적설계 프레임워크를 COM(Coponent Object Model)^[13] 기술을 이용하여 개발한다. 즉, GUI를 이용한 문제 구성 순서를 배제하고, 사용자가 가장 잘 다룰 수 있는 프로그램 언어로, 원격에 위치한 설계 자원들을 통합하여 최적설계를 실행할 수 있는 환경, 즉 프레임워크를 개발해야 하는 것이다.

또한 이번에 개발된 최적설계 프레임워크는 최적설계 인터페이스 모듈, 분산 컴퓨팅 모듈, 엑셀 후처리 인터페이스 모듈, 파일래퍼가 각각의 독립성을 유지할 수 있도록 설계되어 사용자가 원하는 모듈만을 독립적으로 사용할 수 있도록 하였다. 이와 더불어 후처리들 엑셀을 통하여 자동화 함으로써 엑셀 상에서 최적설계 과정을 관찰할 수 있으며 결과 데이터 역시 엑셀에서 확인하고 차트화 하여 사용자의 추가 작업 없이 엑셀에서 데이터 관리가 가능하도록 하는 작업 환경을 제공한다.

마지막으로 이번 개발과정에서 습득한 8개 프로그램 언어의 COM 클라이언트의 특성, 즉 각 언어에 따른 실수형 배열 처리에 관한 VARIANT type을 정리하여 향후 COM기반의 API지원 설계 프레임워크 개발에 필요한 기초 자료를 제시하였다.

2. 최적설계 프레임워크 개념 및 아키텍처

여기서는 서론에서 언급한 최적설계 프레임워크를 개

발하기 위해서 필요한 기술 및 구현 방법에 대해서 언급하겠다. 즉, 8개 프로그램 언어에서 사용 가능한 인터페이스 모듈을 만들기 위해 사용한 COM 기술과 VARIANT type에 대해서 정리하고, 전체 프레임워크를 구현하는 설계 개념 및 아키텍처를 설명한다.

2.1 COM 개요 및 언어별 VARIANT types

COM은 바이너리 컴포넌트로 만들어진 어플리케이션을 수용하는 소프트웨어 아키텍처이다. COM은 컴파일된 코드로 구성되었기 때문에 소스코드는 COM을 지원하는 어떤 프로그램 언어로 작성될 수 있다. 컴포넌트는 전체 어플리케이션의 새 컴파일 없이 간단히 교환되어질 수 있으므로 어플리케이션을 업그레이드하기가 편리하다. 또한, 컴포넌트의 위치가 어플리케이션에 노출되어 있어, 어플리케이션의 수정 없이 분리된 프로세스 뿐만 아니라 원격 시스템에서도 호출 된다. COM을 사용하면, 개발자들과 사용자 들은 다른 프로그램 제공자들에 의해서 만들어진 어플리케이션에 특화된 컴포넌트들을 선택하여 사용할 수 있을 뿐만 아니라 그들을 통합하여 완벽한 어플리케이션을 만들 수 있다. 즉, 프로그램 언어에 상관없이 컴퓨터 프로그램이 가능하게 되는 것이다^[14]. 따라서 최적설계 프레임워크를 구성하는 기능을 COM을 이용하여 설계 함으로서 사용자가 원하는 프로그램 언어 환경에서 동일한 이름으로 같은 기능을 사용할 수 있다.

COM을 이용하여 최적설계 프레임워크를 구현하는데 있어서 가장 중요한 부분은 실수형 배열 처리에 관한 것이다. 그 이유는 클라이언트 역할을 하는 프로그램 언어의 특성에 따라서 COM 서버 측 VARIANT의 SafeArray를 이용하여 각 언어의 실수형 배열을 처리하는 방법이 다르기 때문이다. 따라서 각 언어의 특성

Table 1. VARIANT types of each programming language

VARIANT TYPE Language	VT_ARRAY VT_R8	VT_ARRAY VT_VARIANT	VT_ARRAY VT_VARIANT VT_BYREF
MATLAB	●		
VB	●		
VBS			●
Python		●	
Perl	●	●	●
PHP		●	
Tcl		●	
JAVA	●		

을 모두 수용할 수 있도록 COM 서버, 즉 최적설계 프레임워크를 설계하는 것이 바람직하다. 이를 위해 각 언어에 따른 VARIANT type을 정리하여 Table 1에 나타내었다.

2.2 프레임워크 설계 개념

앞에서 소개한 COM 기술을 이용하여 8개 언어에서 사용 가능한 최적설계 프레임워크에 대한 설계 개념을 Fig. 1에 나타내었다. 최적설계 프레임워크는 사용자가 MATLAB, VB, VBS, Python, Java, Perl, Tcl, PHP중 하나의 언어를 기본적으로 사용할 수 있다는 가정에서 시작한다. 따라서 사용자는 익숙한 프로그램 언어로 외부에 위치한 다수의 CAE 도구들과 최적설계 모듈을 통합하여 최적설계 환경을 구성할 수 있다. 또한 엑셀을 이용하여 목적함수의 경향을 엑셀 차트를 통하여 관찰할 수 있으며, 설계 완료 후 작성되는 결과 파일로부터 데이터를 읽어 엑셀에 자동 저장시키고 사용자가 원하는 데이터에 대한 차트를 제공할 수 있다.

이를 위해 최적설계를 수행하는 ADS(Automated Design Synthesis)^[15] 인터페이스, 엑셀 후처리 인터페이스, 그리고 분산 컴퓨팅 환경을 구성하는 기능을 COM을 이용하여 설계 하였다. 또한, CAE 인터페이스 역할을 담당하는 파일래퍼를 별도로 개발하였다.

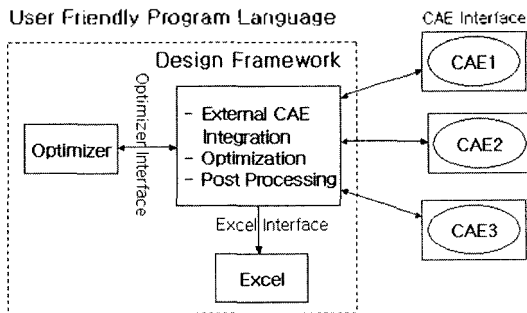


Fig. 1. Design concept of the design optimization framework.

2.3 프레임워크 아키텍처

전 절에서 8개 프로그램 언어에서 사용 가능한 최적설계 프레임워크에 대한 전반적인 설계 개념에 대해서 살펴 보았고, 여기서는 설계 프레임워크의 전체적인 구조를 Fig. 2를 기초로 설명하겠다. 그림에서와 같이 최적설계 프레임워크는 크게 메인 프레임워크, 해석 클라이언트, 해석 서버, 그리고 파일래퍼의 네 부분으로 구성되어 있다. 메인 프레임워크는 최적설계 툴인 ADS,

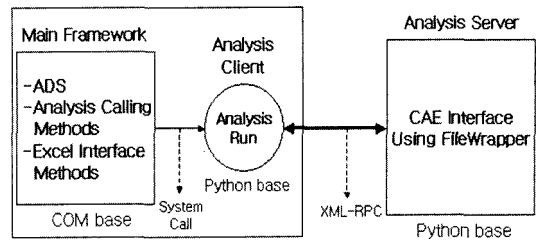


Fig. 2. Architecture of the design optimization framework.

원격지의 해석도구를 제어하는 해석 호출 메소드 (method) 그리고 엑셀을 이용하여 후 처리를 하기위한 엑셀 인터페이스 메소드로 구성되어 있으며 이 모든 기능은 COM으로 인터페이스 되어 있다. 그리고 해석 클라이언트와 해석 서버는 분산 컴퓨팅 환경을 구성한다. 해석 클라이언트와 서버는 파이썬(Python)을 이용하여 구현되어 있으며 실질적으로 XML-RPC^[16] 기술을 이용하여 데이터 통신을 제어하므로, 서버 환경에 구애 받지 않고 사용이 가능하다. 즉, CAE 도구들이 MS-Windows, UNIX, LINUX등 다양한 운영체제에서 동작을 한다고 하여 플랫폼에 의존한 새로운 해석 서버의 도움 없이 분산 컴퓨팅 환경의 구성이 가능하다. 마지막으로 파일래퍼는 CAE 도구들의 인터페이스 역할을 담당하므로 해석 서버와 함께 동작하며 파일래퍼 역시 다양한 CAE 운영환경을 수용하기 위해 플랫폼 독립적인 언어인 파이썬을 이용하여 구현하였다.

이와 같이 구성된 최적설계 프레임워크는 사용자가 원하는 프로그램 언어 환경에서 함수를 호출하듯이, 필요한 메소드를 호출하여 사용하면 된다. 즉, 최적설계를 수행하기 위해서는, COM 인터페이스로 작성된 ADS를 정의하고, 외부의 CAE 도구들을 호출하기 위하여 해석호출 메소드를 사용하게 되며 후처리를 위해 엑셀 인터페이스 메소드를 호출하게 된다. 따라서 메소드의 호출을 통하여 해석 클라이언트와 해석 서버, 그리고 사용자에게 의해 미리 정의된 파일래퍼가 자동으로 실행되어 최적설계를 수행하게 된다.

4. 적용 문제

이번 장에서는 앞에서 설명한 최적설계 프레임워크 이 기계공학 문제에 어떻게 적용이 되는지, 문제점은 무엇인지를 집중적으로 검토한다. 이를 위해 3부제 트러스 최적화 문제^[17], 10부제 트러스 최적화 문제^[18], 그리고 Simple MDO 문제^[19]를 선택하였다. 이 문제들은 기존에 최적설계가 수행되어 최적화 값

에 대한 신뢰성을 가지고 있기 때문에 이번에 개발된 최적설계 프레임워크를 검증하는데 적절한 예제들이다.

적용 문제들은 이번에 개발된 최적설계 프레임워크를 이용하여 8개 언어에서 수행 되었으며, 자동으로 만들어진 엑셀 결과 데이터를 통하여 최적설계 프레임워크의 신뢰성을 확인한다.

4.1 3-부재 트러스 최적화 문제

이 문제는 Fig. 3에서와 같이 세 개의 트러스 부재로 구성되어 있는 구조물에 하중 P_1, P_2 가 작용한다. 목적함수는 부재의 총 부피를 최소화하는 것이며 제한조건은 각 부재의 응력이며, 설계변수는 부재의 단면적의 크기가 된다. 이 문제를 통하여 이번에 개발된 ADS 인터페이스 그리고 엑셀 후처리 인터페이스의 독립성 및 성능을 8개 언어에서 검증하기로 한다.

각 프로그램 언어에서 이 문제를 풀기 위해 설계 변수들의 초기값은 모두 0.5로 동일하게 설정하였으며 MMFD(Modified Method of Feasible Directions) 방법을 사용하였다.

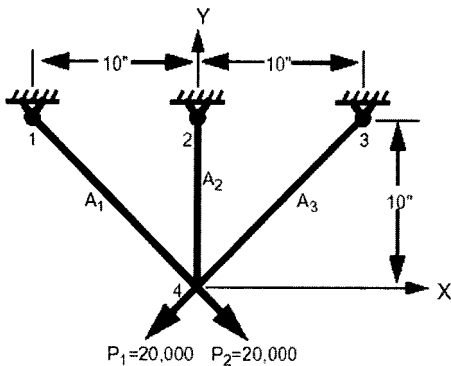


Fig. 3. Three bar truss optimization design.

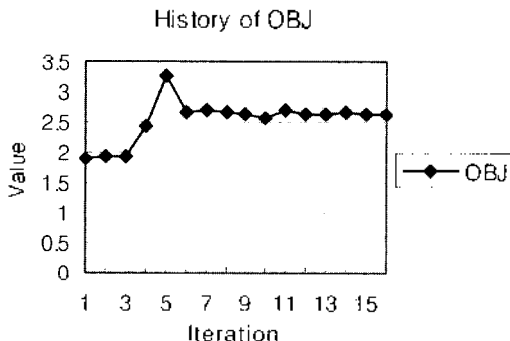


Fig. 4. Objective function history of three bar truss design.

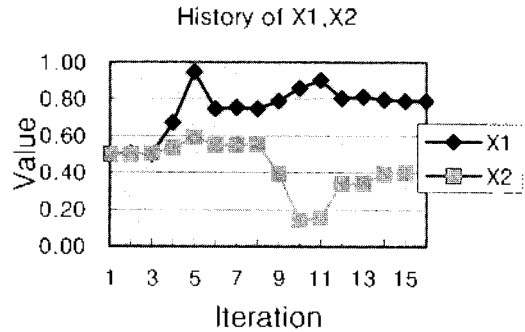


Fig. 5. Design variable histories of three bar truss design.

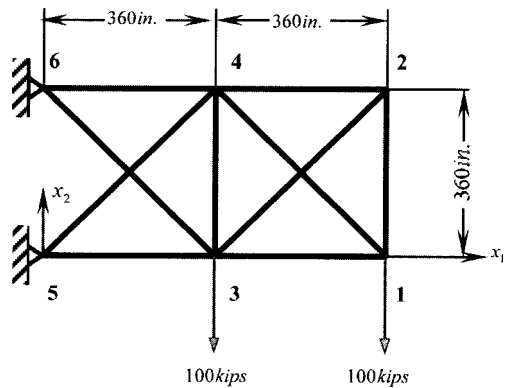


Fig. 6. Ten bar truss optimization design.

Fig. 5은 각 프로그램 언어에서 최적설계를 수행한 이후의 목적함수에 대한 그래프를 나타냈으며 Fig. 6은 설계 변수 X_1, X_2 에 대한 그래프이다. 목적함수 및 설계변수 그래프에서와 같이 각 프로그램 언어에서 동일한 값을 얻을 수 있었으며 참고문헌^[10]를 통하여 최적설계값이 일치함을 확인하였다.

4.2 10-부재 트러스 최적화 문제

두 번째 사례는 10부재 트러스 최적화 문제이다. 이 문제는 Fig. 6에서와 같이 두 질점, 5번과 6번이 벽에 고정되어 있고 나머지 두 질점, 1번과 3번에 100 kips의 하중을 받을 때, 주어진 주어진 허용 응력의 허용 범위를 만족하면서 전체 트러스의 중량을 최소화하는 각 부재의 단면적을 구하는 최적설계 문제이다.

이 문제를 통하여 첫 번째 문제에서 검증된 ADS 및 엑셀 후처리 인터페이스 뿐만 아니라 CAE 인터페이스 역할을 하는 파일랩퍼의 기능 및 분산 컴퓨팅 환경을 8개 언어에서 확인한다. 특히, 이 문제를 해석하기 위해 상용 CAE 툴인 ANSYS를 이용함으로써 본 연구에서 사용된 파일랩퍼가 상용 CAE 도구와 함께

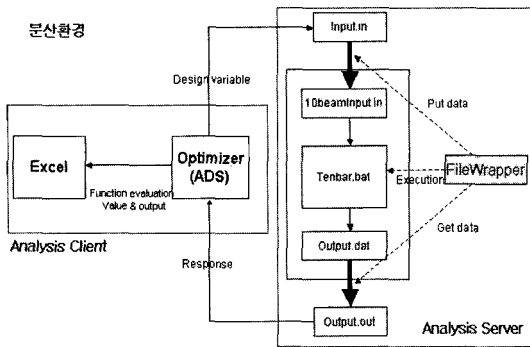


Fig. 7. Environment for solving ten bar truss optimization design.

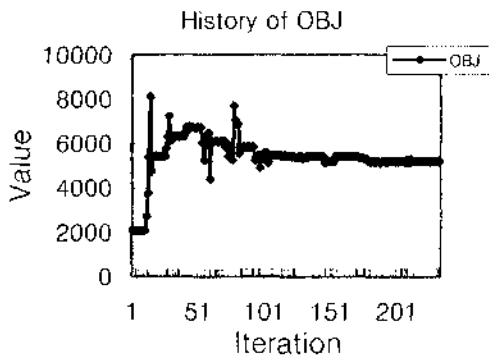


Fig. 8. Objective function history of ten bar truss design.

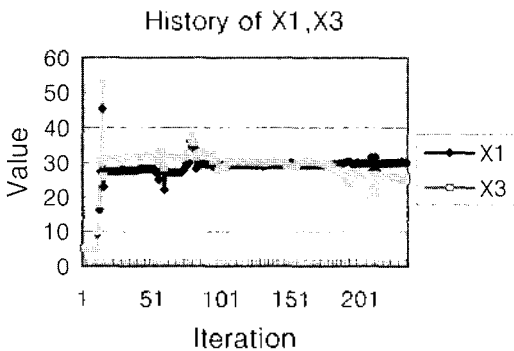


Fig. 9. Design variable histories of ten bar truss design.

사용이 가능함을 확인한다.

각 프로그램 언어에서 이 문제를 해결하기 위해 설계변수의 초기값으로 각각 2를 주었으며 MMFD 방법을 사용하였다.

이 문제를 분산 컴퓨팅 환경에서 실행시키기 위해 Fig. 7과 같은 환경을 만들었다. 즉, 최적설계를 수행하는 ADS 부분과 예상을 이용한 후처리 부분을 해석

클라이언트로 만들고 파일래퍼로 실행되는 ANSYS를 해석 서버 부분으로 분리하였다.

Fig. 8, Fig. 9는 위 문제를 8개 프로그램 언어에서 실행하여 얻은 목적함수와 설계 변수 X_1 , X_3 의 변화 추이를 나타내며 모든 프로그램 언어에서 동일한 결과를 얻었다.

4.3 Simple MDO 문제

마지막 문제는 MDO(Multidisciplinary Design Optimization)^[22] 검증을 위한 예제 중 NASA MDOB의 Heart Dipole분체를 쉽게 만든 Simple MDO 문제이다. 이 문제는 Heart Dipole 문제가 비선형성이 과도하여 초기값의 영향을 크게 받아 해에 대한 수렴의 여부를 알기 어렵기 때문에 초기 값의 영향을 크게 받지 않는 문제로 재 구성한 것이며 이에 대한 문제의 정식화를 아래에 나타내었다.

$$\begin{aligned}
 f_1 &= x_1^2 + x_2 - 2x_3 = 0 && \text{System Optimization} \\
 f_2 &= x_2^2 + x_3 - 2x_6 = 0 && \text{Determine } 0 < x_5, x_6, x_7, x_8 < 10 \\
 f_3 &= x_3^2 + x_4 - 2x_7 = 0 && \text{Minimize } OBJ = f_5 + f_6 + f_7 + f_8 \\
 f_4 &= x_4^2 + x_1 - 2x_3 = 0 && \text{Subject to } f_5 \geq 0, f_6 \geq 0, f_7 \geq 0, f_8 \geq 0 \\
 f_5 &= x_1x_3 + x_3^2 - 2 = 0 \\
 f_6 &= x_2x_3 + x_3^2 - 2 = 0 && \text{System Analysis} \\
 f_7 &= x_3x_4 + x_4^2 - 2 = 0 && \text{Find } x_1, x_2, x_3, x_4 \text{ for given } x_5, x_6, x_7, x_8 \\
 f_8 &= x_4x_1 + x_4^2 - 2 = 0 && \text{Satisfying } f_1 = f_2 = f_3 = f_4 = 0
 \end{aligned}$$

이 문제를 해결하기 위해 MDO 방법론 중 MDF (Multiple Discipline Feasible)^[23] 방법을 사용하였다. 특히, Fig. 10에서와 같이 최적설계 시스템과 해석 시스템으로 분리하고, 해석 시스템도 두개의 해석 서버로 분리 함으로써 이번 프레임워크 분산 컴퓨팅 환경에서 MDO 방법을 적용시킬 수 있음을 보여주고 있다.

Fig. 10을 통하여 Simple Problem을 간단히 설명하면, 최적설계 시스템에서 설계변수 x_5, x_6, x_7, x_8 이 두 부분으로 나누어져 각각의 해석 시스템으로 보내지고 그 값들을 이용하여 각각의 해석 시스템에서 고정점 반복법을 통하여 x_1, x_2, x_3, x_4 를 구하여 이 값들을 최적설계 시스템으로 보내는 형태로 구성되어 있다.

이 문제도 앞의 문제들과 마찬가지로 8개 언어에서 실행을 시켰으며 그에 대한 목적함수와 설계변수 값은 Fig. 11, Fig. 12와 같이 동일한 값을 얻을 수 있었다. 특히, 설계변수 값이 모두 1이 나오는 것을 볼 수 있으므로 최적설계가 정확히 이루어 졌음을 알 수 있다.

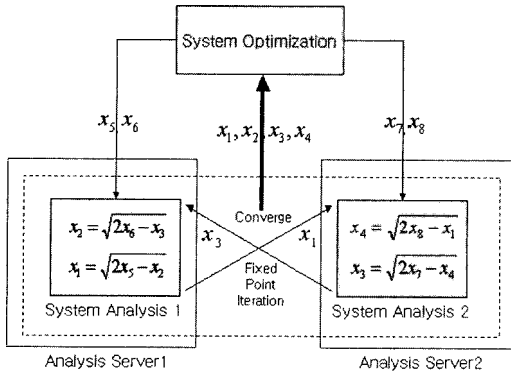


Fig. 10. Environment for Simple MDO problem.

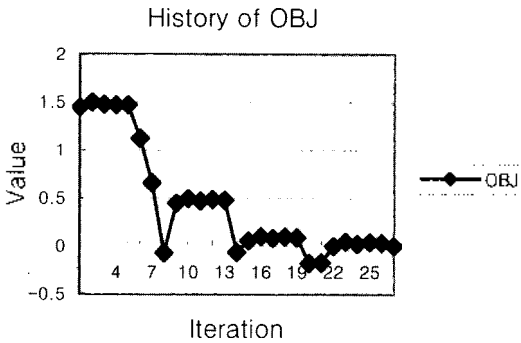


Fig. 11. Objective function history of Simple MDO.

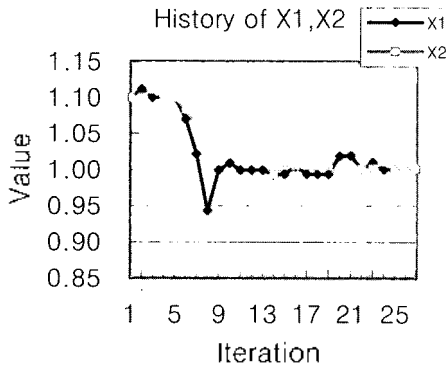


Fig. 12. Design variable histories of Simple MDO.

5. 결 론

본 연구에서는 MATLAB, VB, Python, VBS, Tcl, Perl, PHP, Java의 8개 프로그램 언어에서 사용할 수 있는 분산 환경 최적설계 프레임워크를 개발하였다. 이번에 개발된 최적설계 프레임워크는 사용자가 최적설계에 대한 기본 개념이 있고 8개 프로그램 언어 중 하

나의 프로그램 언어에 대한 기초 지식이 있을 경우 원격에 위치한 CAE 자원을 통합하고 이를 자동 실행시켜 최적설계를 수행할 수 있는 환경을 제공한다. 또한 최적설계 모듈 인터페이스, 엑셀 후 처리 인터페이스, 파일래퍼, 분산 컴퓨팅 기능에 대한 서로의 의존성을 제거하여 사용자가 원하는 기능만 따로 사용할 수 있게 만들었다. 즉, 사용자가 액셀 후 처리 기능만을 이용하고 싶다면 다른 3개의 기능을 제외하고 엑셀 후 처리 기능만을 이용할 수 있도록 최적설계 프레임워크를 설계 하였다.

이와 더불어 엑셀을 이용하여 최적설계 과정 중에 나오는 목적함수를 실시간 차트화 하여 진행 과정을 관찰 할 수 있으며 사용자가 원하는 설계 데이터에 대한 결과 값 역시 차트화 할 수 있도록 자동화 하였다.

마지막으로 이번 개발 과정에서 얻게 된 다양한 프로그램 언어의 COM 클라이언트의 특성을 파악하여 향후 COM 기반의 API 지원 설계 프레임워크 개발에 필요한 인터페이스 방법론을 제시할 수 있는 자료를 얻었다. 이 자료는 실수 배열을 COM을 이용하여 처리하는데 있어서 COM 클라이언트의 특성에 따라서 COM 서버의 설계가 달라지는 분체점을 해결하는데 도움을 줄 것이다. 즉, 8개 프로그램 언어의 COM 클라이언트를 분석하여 얻은 VARIANT 타입의 특징을 정리하고 이를 바탕으로 COM 클라이언트의 배열 처리에 대한 공통된 특성을 파악하여 향후 설계 프레임워크 개발에 도움이 될 것이다.

참고문헌

1. 이재우, "다분야 통합최적설계: 설계개념, 연구 동향 및 응용", 한국항공우주학회지, 제29호, 제5호, pp. 139-148, 2001.
2. <<http://www.engineous.com>>
3. <<http://www.phoenix-int.com>>
4. <<http://www.vrand.com>>
5. <<http://www.mathworks.com>>
6. <<http://www.python.or.kr>>
7. <<http://msdn.microsoft.com/korea/msdn/vbasic>>
8. <<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/script56/html/vbstutor.asp>>
9. <<http://www.activestate.com/Solutions/Programmer/Tcl.plex>>
10. <<http://man.phpschool.com>>
11. <<http://www.hostinglogin.com/help/perl/>>
12. <<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/vjcore98/html/vjovrprogrammersguide.asp>>
13. Grimes, R., Templeman, J., Stockton, A., Watson, K. and Reilly, G, "Beginning ATL 3 COM Pro-

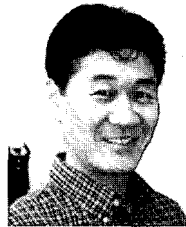
gramming”, Wrox Press Ltd., Birmingham, UK, 2000.

14. <http://www.mathworks.com/access/helpdesk/help/techdoc/matlab_external/matlab_external.shtml>
15. Vanderplaats, G. N., “ADS Manual Ver. 2.01”, Engineering Design Optimization, Inc., Santa Barbara, CA, USA, 1987.
16. <<http://www.xml-rpc.com>>
17. <<http://h18009.www1.hp.com/fortran/visual/>>
18. <<http://www.python.or.kr:8080/python/Internet/xml-rpc.html>>
19. 노현석, “데이터베이스 기반의 최적설계 인터페이스 개발”, 한양대학교 대학원 석사학위 논문, 2002.
20. Haug, E. J. and Arora, J. S., “Applied Optimal Design”, Wiley&Sons, New York, U.S.A, pp. 171-173.
21. 이세정, “Simple Problem for MDO”, 서울시립대학교 기계정보공학과, 2000.
22. 조상오, 이세우, 변영환, “MDO 프레임워크 개발을 위한 해석 및 최적화 과정 통합에 관한 연구”, 한국항공우주학회지, 제30권, 제7호, pp. 1-10, 2002.
23. 황원준, “다분야통합최적설계 문제 구현을 위한 상용 설계 프레임워크 벤치마크”, 한양대학교 대학원 석사학위 논문, 2001.



염근철

1996년~2001년 서울시립대학교 성밀기
계공학과
2002년~2004년 서울시립대학교 기계정
보공학과
2004년~ 현대자동차 차량해석팀



이세정

1976년~1980년 서울대학교 기계공학
1980년~1982년 한국과학기술원 기계공학
1986년~1989년 펜실베이니아주립대학 기
계공학
1993년~ 서울시립대학교 교수