

객체지향기법을 이용한 밭관개조직 관망해석 시스템 개발

Development of Upland Irrigation Network Analysis System
Using Object-Oriented Programming(OOP)

이 성 학* · 정 하 우 (서울대)
Lee, Sung Hack · Chung, Ha Woo

Abstract

Upland Irrigation Network Analysis System(UINAS) used Object-Oriented Programming (OOP). The results of using OOP is definition of objects and class hierarchy for UINAS, Objects of UINAS are consist of the Pipe, Sprinkler, Valve, Pump, Tee, Bend and Contractions. The class hierarchy have cooperative design for FEM in analysing the irrigation network. Therefore UINAS have a flexibility in additioning the network components.

I. 서론

고압관수로를 이용한 관개시스템에 있어서 관망의 해석은 필수적이며, 또한 많은 연구가 이루어져 왔다. 관망의 해석은 비선형방정식을 포함한다. 따라서 반복적인 방법에 의하여 해를 구해야만 한다. 관망의 해석에 있어 비선형방정식을 풀기 위한 몇 가지 방법이 제시되어 졌다. 이러한 방법에는 Hardy-Cross법, Newton-Raphson법, 선형이론해석, 유한요소법을 이용한 해석 등이 있다.^[6]

Hardy-Cross법은 각 요소의 교차점에서 연속방정식을 만족시키도록 유량을 가정한 다음, 가정된 유량을 점차적으로 보정하여 각 관망의 유량을 구하는 방법이다. 그러나 관망이 복잡하고 클 경우 해의 수렴속도가 늦은 단점이 있다. Newton-Raphson법은 동시에 모든 절점의 유량과 수두를 계산함으로써 해의 수렴속도를 빠르게 하였으나, 최초의 가정 유량에 따라 해의 수렴성이 크게 좌우되며, 각 유량에 대한 미분치를 구하여야 하는 번거로움이 있다. 선형이론방법은 비선형 에너지 방정식을 선형화하여 해석하는 방법으로 정확성이 높기 때문에 관망해석에 널리 이용된다. 그러나 유량조절 밸브등과 같은 관로 요소에 대한 수리계산에 어려움이 있다.^[6] 유한요소법을 이용한 관망 해석법은 커다란 관망에 있어 쉽게 적용할 수 있고 적은 컴퓨터용량으로도 계산이 가능하며 기존의 유한요소 코드를 재 사용할 수 있을 뿐만 아니라 관망에 있어 중요한 요소(관, 펌프, 스프링클러, 밸브 등)를 고려하여 해석할 수 있는 장점이 있다.^{[3][4][5][6]}

객체지향기법이란 어떠한 문제를 자료와 그 자료를 가공하는 방법을 하나의 객체로 정의하고 정의된 객체를 중심으로하여 문제를 해결하려고 하는 프로그래밍의 방법론이다. 이 방법론의 특징은 객체의 개념과 자료를 추상화함으로써 실세계와 비슷하게 프로그램을 구성할수 있으며 시스템개발의 용이성과 함께 재사용으로 인한 개발비용의 절감을 들수 있다.^{[1][2][7]}

본 연구의 목적은 고압관수로 관개시스템의 관망해석에 있어 유한요소법과 객체지향기법을 적용하여 발관개조직 설계자동화 시스템을 개발하는 것이다.

II. 관망의 해석

1. 유한요소법을 이용한 관망의 해석

유한요소법을 이용한 관망의 해석은 Bralts(1985)등이 점적관개에 있어 적용한 것을 시작으로 하여 Haghghi(1988)등은 관로망의 요소중 bend와 Tee에 대하여 적용한 바 있으며, Haghghi등(1989)은 급확대, 급축소, 밸브와 부스터 펌프에 대하여 그리고 Saldivia(1990)은 살수관개시스템에 대하여 적용한 바 있다.

1) 기본이론

관망의 해석은 에너지 방정식과 연속방정식을 이용하여 각 절점의 수두와 유량을 구하는 것이다. 관망의 하나의 요소에 대한 에너지 방정식은 다음과 같다.

$$Z_i + \frac{P_i}{\gamma} + \frac{1}{g} \frac{V_i^2}{2} + H_{pp} = Z_j + \frac{P_j}{\gamma} + \frac{1}{g} \frac{V_j^2}{2} + h_L \quad (1)$$

Z : 절점 위치수두, P : 절점 압력, γ : 물의 단위중량, H_{pp} : 펌프에 의한 상승수두

h_L : 관 및 요소에 의한 수두손실, V : 단위중량에 대한 속도

부스터 펌프를 제외한 관 및 요소에 대한 수두손실 h_p , h_c 는 일반적으로

$$h_p = K_p Q^m, \quad h_c = K_c \frac{V^2}{2g} \quad (2)$$

과 같이 정리될 수 있다.

식(1)을 다시 정리하면 식(3)과 같다.

$$(Z_i - Z_j) + (H_i - H_j) = K_c \frac{V^2}{2g} - \frac{1}{g} \left(\frac{V_i^2}{2} - \frac{V_j^2}{2} \right) \quad (3)$$

식(3)의 우변을 $C_c Q$ 의 형태로 정리하면 식(4)와 같다.

$$(Z_i - Z_j) + (H_i - H_j) = C_c Q \quad (4)$$

C_c : 요소에 따라 정리, Q : $\frac{\pi D^2}{4} V$ 이며 D : 지름

여기서 각 요소의 절점에 대한 연속방정식을 적용하면 식(5),(6)과 같다.

$$-Q^{(e-1)} + Q^{(e)} = 0 \quad (5)$$

$$\begin{aligned} Q_i^{(e)} &= C_c (H_i - H_j) + C_c (Z_i - Z_j) \\ Q_j^{(e)} &= -C_c (H_i - H_j) - C_c (Z_i - Z_j) \end{aligned} \quad (6)$$

식(6)를 행렬의 형태로 다시 쓰면 식(7)과 같다.

$$\begin{pmatrix} Q_i^{(e)} \\ Q_j^{(e)} \end{pmatrix} = \begin{bmatrix} C_c & -C_c \\ -C_c & C_c \end{bmatrix} \begin{pmatrix} H_i \\ H_j \end{pmatrix} - \begin{bmatrix} -C_c & \Delta Z \\ C_c & \Delta Z \end{bmatrix}, \quad \Delta Z = Z_i - Z_j \quad (7)$$

이 행렬을 각 요소마다 구하여 강성도 행렬을 하여 강성도 행렬을 풀면, 각 점의 수두와 유량을 구할 수 있다. [4][5][6][7]

III. 객체지향기법

객체는 자료와 동작으로 이루어지며 이러한 자료와 동작을 구성하는 방법에는 객체의 자료나 메소드로의 접근은 메시지의 교환을 통해서만 가능하다. 이를 캡슐화(encapsulation)라하며, 기존객체의 확장사용이 상속성(inheritance)이며, 상속성을 이용하여 구성된 객체의 계층의 최상부에 있는 객체는 어떠한 문제를 가장 추상화된 형태로 표현하게 되는데 이를 추상화(abstraction)라고 하며, 가공의 대상이 다르면 가공의 방법 또한 다르게 적용할 수 있는 것을 객체의 다형성(polymorphism)이라고 한다. [2]

1. 객체지향언어

객체지향기법을 구현하기 위한 방법은 컴퓨터를 이용한 시뮬레이션을 위해 1960년대 후반에 개발된 SIMULAR-67이라는 프로그래머에서 찾을 수 있으며 본격적인 학문연구의 대상은 Smalltalk-80의 발표부터이다. 그 후 수많은 객체지향언어가 개발되었지만 대부분은 학문적인 수준에 머물렀으며 1980년대에 발표된 C++은 프로그래머들 사이에 널리 사용되고 있다.

Java™는 1995년도 Sun Microsystems에서 발표된 객체지향언어로 기존에 존재하던 언어에 기반을 두지 않고 인터넷, 멀티미디어, 멀티프로세싱과 같은 발전된 형태의 컴퓨팅 환경에 염두를 두고 개발되어 그래픽, 사용자 접속환경(Graphic User Interface), 네트워킹 등에 관한 객체를 직접 지원한다. Java™는 가상기계(virtual machine)개념을 적용하여 컴퓨터의 구조와 무관하게 Java™의 실행모듈이 설치된 컴퓨터에서는 프로그램의 수정 없이 실행이 가능하다. 이와 같이 객체지향기법과 언어를 이용하면 프로그램의 개발 및 확장이 용이하며 Java™언어를 이용 하므로써 프로그램의 호환성을 높일 수 있다. [1][7]

2. 관망 해석시스템의 구성

1)객체의 설계

객체지향기법을 적용하는 첫 단계로 객체를 분류하고 객체를 설계해야한다. 객체는 우선 일반적인 특징을 가지는 관, 살수기, 펌프, 요소의 클래스군으로 나누고, 이를 다시 종류에 따른 분류를 하여 객체를 구성하였으며 그 결과는 Table 1과 같다.

Table 1. 시스템의 객체 분류

클래스군	클래스	비고
파이프	파이프	종류에 따라 세부적으로 분류된 것이 객체
펌프	펌프	
살수기	살수기	
	밸브	
	Bend	
	Tee	
	급확대관	
요소	급축소관	

2) 클래스 다이어그램

관망해석에 시스템에 있어 클래스 다이어그램은 크게 파이프군, 살수기군, 펌프군 그리고 요소군으로 나뉘며, 전체의 클래스에 대한 추상객체 아래 각각의 클래스군에 대한 일반적인 정보를 포함하는 소단위 추상클래스를 가진다. 이 소단위 추상클래스를 상속받아 각각의 요소에 대한 클래스를 구성한다. 관망요소 객체에 대한 클래스 다이어그램은 Figure 1.과 같다.

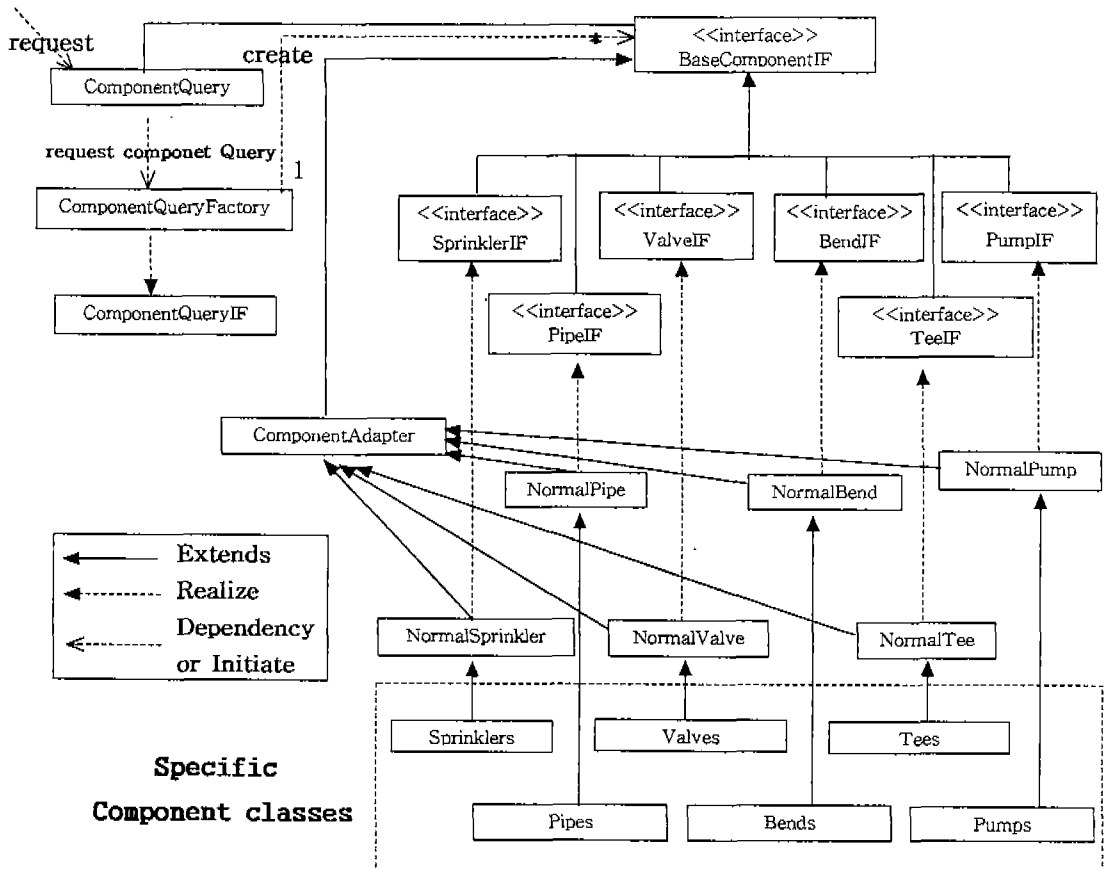


Figure 1. 시스템의 클래스 hierarchy

3) 클래스의 정의

객체는 유한요소법을 이용한 관망해석에 있어 강성도 행렬을 구성하는데 중점을 두고 설계하였다. 강성도 행렬을 구성하기 위하여서는 각 요소의 마찰계수를 우선 산정 하여야 하며 이를 위하여 각 마찰계수를 결정하는 요소를 계산하는 메소드와 변수를 구성한다. 그리고 모든 객체는 상위 객체로부터 좌표를 나타내는 부분은 상속을 받아 사용한다.

Table 2. 관객체(NormalPipe Object)정의 예

클래스명	변수	메소드	비고
관 (NormalPipe)	-Node ID -Element ID -Head Loss Coefficient · 관마찰손실계수 · Manning-Chezy 지수 · Hazen-William 지수 · Darcy-Weisbach - Diameter - Velocity - Re - Junction Head - Flow Rate	- 매트릭스 구성 - 마찰손실계수 결정 - 흐름 판정	위치정보는 상위클래스로부터 상속
	-Picture Name -Text File Name	· 문자정보 제공 · 사진정보 제공	

3. 관망해석 시스템의 구축결과

발판개에 있어서 관망 해석을 위하여 시스템을 구성한 결과는 Figure 2와 같다. 그림은 임의의 관망을 구성한 모습을 나타내는 그림이다. 본 시스템은 그래픽에디터와 해석프로그램으로 구성되며, 시스템은 입력자료를 그래픽에디터를 사용하여 입력하며, 결과를 시각적으로 표현하여준다.

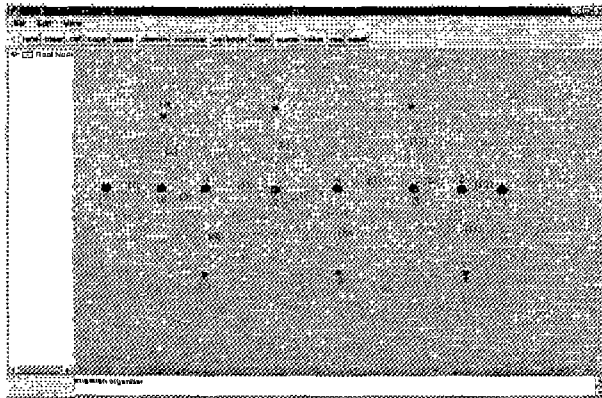


Figure 2. 시스템의 구축결과

III. 결론

본 연구에서는 발판개조직의 관망해석에 있어서 유한요소법과 객체지향기법을 도입하여 발판개조직에 있어 관망해석시스템을 개발하였다.

본 연구에서 개발한 발판개조직 관망해석시스템의 연구결과를 요약하면 다음과 같다.

1. 본 연구에서는 발판개조직 관망해석을 위하여 유한요소법을 적용하였다.
2. 객체지향기법을 적용하여 발판개조직 관망해석시스템의 객체의 속성과 메소드를 정의 하였으며, 객체별로 체계를 구성하였다.
3. 본 시스템에 있어 객체와 메소드는 발판개조직 관망해석을 위한 유한요소법에 사용되는 각요소를 중심으로 설계하였다.
4. 본 시스템은 객체지향적(Java™언어 사용) 설계/구축 하였으므로 앞으로 계속 발생하는 관망해석요소를 쉽게 시스템에 반영할 수 있으며 관망해석의 사용자 접속환경(GUI)등의 확장성을 가지도록 하였다.

IV. 참고문헌

1. 이호재, 1998, "동적 부하 조정 기법을 적용한 객체지향 유한요소 분산처리 시스템의 개발", 서울대학교 석사논문, pp.5-20
2. 윤정모, 한규정역, 1996, "객체지향 시스템 개발". 통일출판사, pp.4-28
3. Haghghi, K. et al, 1989, Modeling expansion/contraction, valve and booster pump inhydraulic pipe network analysis: a finite element approach, Trans. of the ASAE 32(6), pp.1945-1953
4. Saldiva, L. A. et al, 1990, Hydraulic analysis of sprinkler irrigation system components using the finite element method, Trans. of the ASAE 33(4), pp.1195-1202
5. Haghghi, K. et al, 1988, Finite element formulation of tee and bend components in hydraulic pipe network analysis, Trans. of the ASAE 33(4), pp.1750-1758
6. Bralts, V. F. et al, 1985, Finite element analysis of drip irrigation submain units. Trans. of the ASAE 28(3), pp.809-814
7. Arnold, Ken and James Gosling, 1996, "Java™ Programming Language", Addison-Wesley Publishing Company