

고속철도 자동 선로 설계 시스템의 설계 및 구현

이수봉*, 김우생*, 김주락**
*광운대학교 컴퓨터 공학부
**한국 철도 기술 연구원
e-mail : sblee@cs.kwangwoon.ac.kr

Design and Implementation of Auto Design System for High Speed Railroad

Soo-bong Yi*, Woo-saeng Kim*, Joo-Rak Kim**
*Dept. of Computer Engineering, Kwangwoon University
**Korea Railroad Research Institute

요 약

한국에 고속 철도가 도입됨에 따라 고속 철도 전용의 선로 설계 시스템이 필요하게 되었다. 그러나 이러한 고속 철도 전용의 시스템은 국내에서 개발되지 않아, 외국의 시스템을 사용하는 경우가 많았다. 따라서 본 논문에서는 한국 실정에 맞는 고속 철도 자동 선로 설계 시스템을 설계하고 구현해 보았다.

1. 서론

고속철도 선로 자동설계 시스템이란 고속 철도 전용의 선로를 위한 설계도를 자동으로 작성해 주는 시스템이다. 필요한 데이터를 입력하면 시스템이 처리하여 그에 맞는 설계도를 만들어 낸다[1]. 이러한 시스템은 고속철도가 존재하는 다른 국가에는 이미 개발되어 있지만 한국에는 그러한 시스템이 없었다. 그래서 본 연구는 이러한 자동 선로 시스템의 국내 개발을 목표로 진행하였다.

고속철도 선로 자동설계 시스템은 도면 설계 응용 프로그램인 AutoCad 와 AutoCad 를 설치하면 제공되는 Cad.ocx 그리고 그것을 쉽게 사용할 수 있는 프로그램 개발 도구인 Visual Basic 으로 만들어 졌다. Visual Basic 에서 명령을 내리면 그것을 받아 Cad.ocx 가 명령을 중계하여, AutoCad 로 보내고 AutoCad 는 내려진 명령을 실제로 처리하는 구조로 되어 있다. 이러한 기법을 VBA(Visual Basic for Application)라 하는데[2], 새로운 도면 처리 응용 프로그램을 개발할 필요 없이 필요한 기능을 구현할 수 있고, 데이터베이스를 지원하며 개발이 용이한

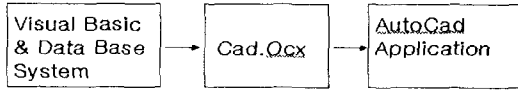
Visual Basic 을 사용함으로써, 개발비용 및 시간을 절약할 수 있다. 또한 한국 지형을 고려한 수식을 적용하여, 최적화된 설계 도면을 만들 수 있도록 하였다.

본 논문은 2 장에서는 Cad.ocx 에 대한 설명을, 3 장에서는 자동선로 설계 시스템에 대한 설명 및 기법에 관한 설명을, 4 장에서는 구현과정을, 그리고 마지막 5 장에서는 본 연구의 결론으로 구성하였다.

2. Auto Cad ActiveX (Cad.ocx)

AutoCad 는 설계 도구로서 많이 사용되어 왔다. 이 응용 프로그램을 설치하면, Visual Basic 에서 AutoCad 의 기능을 프로그램으로 제어할 수 있는 ActiveX 를 사용할 수 있다. 이것을 이용하면 따로 도면을 디스플레이 시켜줘야 할 부분을 제작하지 않고도 손쉽게 AutoCad 의 화면을 빌려서 사용자가 원하는 도면을 프로그램으로 제작 할 수 있다. 또한 데이터베이스를 지원하는 Visual Basic 과 연동하여 사용이 가능하다는 점에서 자동 설계 시스템에 적합한 형태라고 생각 할 수 있다. 그림 1 은 ActiveX 즉 Cad.ocx

가 Visual Basic 과 AutoCad 사이를 중계하여 명령을 처리하는 과정을 보여주고 있다.

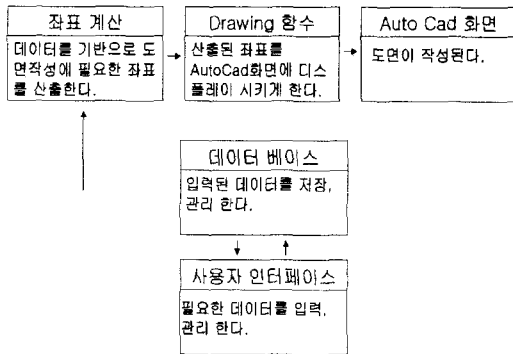


<그림 1 > Cad.ocx 관계도

3. 고속철도 선로 자동설계 시스템

3.1 시스템 구성

고속철도 선로 자동설계 시스템의 구조는 그림 2 처럼 크게 다섯 가지 부분으로 구성되어 있다. 실제 도면을 화면에 디스플레이 시키는 부분, 도면을 생성하기 위한 좌표와 수치를 계산하는 부분, 좌표를 AutoCad 화면에 디스플레이 시키기 위해 Cad.ocx 의 함수를 조합한 Drawing 함수 부분, 수식에 필요한 입력수치를 저장, 관리해 주는 데이터베이스 부분, 그리고 사용자가 데이터를 입력, 관리 하는 사용자 인터페이스로 나눌 수 있다. 데이터베이스에 저장된 데이터를 기반으로 좌표를 생성한 다음, 이 좌표 값을 Drawing 함수에 입력하게 되면 이 함수들은 AutoCad 화면에 도면을 그리게 된다.



<그림 2> 시스템 도해

3.2 설계 기법

3.2.1 도면 표기 기법

도면을 디스플레이 시켜주는 부분은 도면을 직접적으로 작성해주는 AutoCad 와 AutoCad 를 제어하기 위해 Visual Basic 으로 제작된 Drawing 함수로 이루어진다. 도면은 AutoCad 를 설치하면 제공되는 Cad.ocx 를 통해 그려주는 것이 가능하다. ocx 에는 Function 을 통해 생성된 선이나, 점, 면들인 Object 와 Object 를 만들 수 있는 Function 이 제공된다[3]. 이러한 object 와 함수를 직접적으로 이용해도 도면제작은 가

능하지만, 코드의 중복이 심하다. 예를 들면 선을 그리는 경우 Update 함수를 함께 쓰지 않으면, 새로운 선은 화면에 표기되지 않는다. 즉 선을 그릴 때 마다 반복해서 코드에 Update 함수를 넣어야만 하는 것이다. 이러한 코드의 중복을 회피하기 위해서 제공되는 object 와 Function 을 조합하여 새로운 Function 을 정의하여 복합적인 명령을 한번에 내려줄 수 있게 하였다.

예 1 은 object 와 Function 을 조합한 제어 함수로 선을 그려주는 역할을 한다. AcadApp 나 lineobj 등은 Object 이고 AddLine(), update 같은 것은 function 이다. AcadApp 는 AutoCad 그 자체를 나타내주고 있다. Cad.ocx 를 프로젝트에 포함 시키고 AcadApp 를 프로그램 모듈에 선언해야만 AutoCad 에서 제공하는 모든 함수를 사용할 수 있다. AddLine(시작점, 끝점)은 선을 그려주는 함수이고 Update 는 그려진 선이 도면에 표기될 수 있도록 화면을 갱신해 준다.

```

Dim lineobj As Object
Set lineobj = AcadApp.mospace.AddLine(startpoint,
endpoint)
lineobj.Layer = "0"
acadApp.Update
    
```

<예 1> 선 그리기 함수 DrawLine()

표 1 은 ocx 의 함수를 조합한 대표적인 Function 의 이름과 기능을 설명하고 있다. 설계자는 이렇게 정의된 다양한 함수를 이용하여 보다 간편하게 프로그램을 제작할 수 있게 된다.

<표 1> 시스템의 Function

DrawLine	직선을 그려준다
DrawDashLine	점선을 그려준다.
DrawCircle	원을 그려준다
DrawPoint	점을 그려준다
DrawDimLine	치수선을 그려준다
DrawText	Text 를 표기한다.

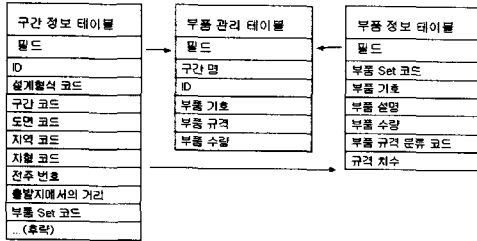
3.2.2 데이터 베이스 구조

데이터베이스는 크게 세가지 테이블로 분리되어 있다. 구간정보 테이블은 구간¹의 정보를 담고 있는 테이블로 동적인 생성이 가능하도록 되어 있다. 이 테이블은 도면 제작 시 필요한 기본적인 정보 및 수식에 들어가는 변수의 값들을 가지고 있다. 부품정보 테이블은 도면을 시공할 때 사용될 부품들의 정보를 담고 있는 것으로 부품정보를 도면에 표기하게 한다.

¹ 고속 철도의 구간을 의미. Ex) 서울 <-> 부산 의 철도구간

구간정보에서 입력된 데이터를 계산에 적용하여 부품 정보 테이블에서 알맞은 부품 세트를 결정해 도면에 표기한다. 부품관리 테이블은 한 구간에서 사용된 부품의 정보가 입력되며, 그것들을 관리한다. 사용한 부품들의 구간별, 전체 통계를 내거나 직접적인 관리를 하는데 주로 사용되며 사용자가 도면을 수정하는 경우 데이터가 변경된다.

시스템의 데이터베이스 구조는 세가지로 분리되어 있지만 그 하위개념의 테이블이 다수 존재한다. 부품 정보 테이블에서는 도면에서의 각 부품의 사용 위치에 따라 해당 테이블들이 따로 존재하고, 부품 관리 테이블 역시 부품의 사용 위치를 기준으로 분류된 테이블들이 있다. 구간 정보 테이블은 구간을 만들 때마다 테이블이 새로 생성되므로 숫자의 제한이 없다. 이러한 테이블들은 필요한 경우 서로 조인 하도록 되어 있고 변경 사항이 있는 경우 부품 관리 테이블의 수치를 갱신하게 하여 통계에 이상이 없도록 한다.



<그림 3> 테이블 구성도

3.2.3 좌표 생성 기법

도면을 생성하기 위해선 좌표가 필요하다. 이러한 좌표를 만들어 주는 것이 수식 알고리즘이다. 수식 알고리즘은 단순히 좌표를 표기해 주는 것 외에, 시공 시 필요한 부품의 크기, 무게, 종류 등을 결정하여, 기본적으로 제공된 입력치 즉 데이터베이스에 저장된 수치들 외에 수식에서 계산된 수치를 최종 도면 설계 시 적용하게 한다. 또한 시공자가 쉽게 이러한 결과를 알 수 있도록 표시해 줌으로서 도면제작자와 시공자의 편의를 최대한 충족시켰다. 수식 (1)은 절대좌표(0,0)에서 양쪽 전주 레일면 기준점 (x,y) 까지의 거리(x = Xtr)로 전주²의 기준이 되는 점을 구하는 식이다. 여기에 사용된 변수들은 구간정보 테이블에 입력된 수치가 적용된다[4].

Rtrx: 레일 사이의 길이
 Rwidx: 레일의 중심과 끝의 길이
 dl: 캔트 수치

$$Xtr = Rtrx / 2 + \sqrt{(Rwidx / 2)^2 - dl^2} \quad (1)$$

수식 (2)는 수식 (1)의 값을 기준으로 실제 도면을

구성하고 있는 브라켓³의 경사주 파이프⁴ 좌표를 생성해 내는 식이다. Cxtsx, Cstsy는 경사주 파이프의 시작점 x, y 좌표 그리고 Cxtsf, Cstfy는 끝점의 x, y 좌표이다. 그리고 각 수식에 들어가는 변수들은 사용자의 입력으로 정할 수 있다.

Location : 지면과의 거리
 hl : 브라켓 길이
 m : 집전면 / 경사주 파이프 길이
 stfit : 경사주 파이프 Fitting
 stlenl : 경사주 파이프 길이

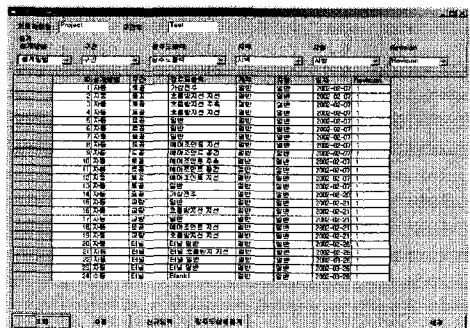
$$\begin{aligned} Cxtsx1 &= 2 * Xtr + Location - Stfit - Rtrx / 2 \\ Cxtsy1 &= hl + m \\ Cxtfx1 &= Cxtsx1 - stlenl * \cos(\alpha) - 0.15 * \cos(\alpha) \\ Cxtfy1 &= Cxtsx1 + stlenl * \sin(\alpha) - 0.15 * \sin(\alpha) \end{aligned} \quad (2)$$

수식 2로 구해진 좌표는 경사주 파이프의 시작위치()와 끝 위치()이며, 아래의 DrawLine 함수를 이용하여 경사주 파이프를 도면에 그린다.

DrawLine(Cxtsx1, Cxtsy1, Cxtfx1, Cxtfy1)

4. 구현

이 시스템은 MS SQL 데이터베이스와 AutoCad 가 설치된 상태에서 VBA 를 이용한 Visual Basic 으로 개발되었다. 데이터는 ADO 의 SQL 문으로 처리 되었으며 도면을 그리는데 필요한 Object 와 Function 은 Cad.ocx 에서 가져왔다[5]. 그림 4는 메인 화면을 나타낸다. 메인 화면에서는 데이터의 대표적인 속성들을 GRID 화면으로 보여준다. 이 곳에서 새로운 구간을 생성하거나, 수정할 수 있다. 메인 화면에서는 구간의 대표적인 속성을 지정하거나, 지정된 속성을 알 수 있다. 예를 들면 일반 선로나, 교량 선로, 혹은 터널 등으로 분류 시키거나, 전주의 형태를 결정하는 작업을 할 수 있다. 단 메인 화면에서는 대표적인 속성만을 지정할 수 있기 때문에 상세한 작업은 상세설계 창을 통해 수행한다.



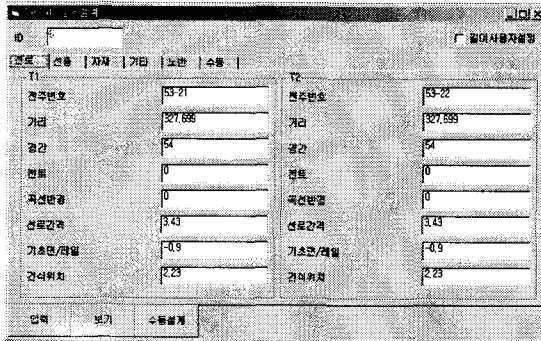
<그림 4> 메인 화면

² 선로의 측면에서 있는 기둥

³ 열차의 상단을 지지하는 받침대. 전주에 달려있다.

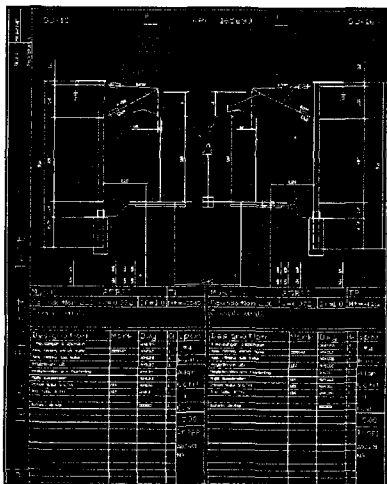
⁴ 브라켓의 한 부분

그림 5는 상세 설계 창으로 메인 화면의 GRID 화면으로 나타낸 구간부분을 더블클릭 하거나 상세설계 버튼을 통해 생성한다. 생성된 설계 창을 통해 각종 데이터를 입력 또는 수정한다. 데이터 입력, 수정은 좌측 전주, 우측 전주로 구분하도록 되어있다. 이 데이터들은 구간정보 테이블에 저장되고 처리되며, 이 데이터들을 기반으로 실제 도면을 그리게 된다. 중복되거나 비슷한 데이터를 계속 입력하는 것을 피하기 위해 새로운 도면을 만들 때 이전의 도면 데이터를 복사하여 필요한 것만 수정할 수 있게 하였다.



<그림 5> 상세 설계 창

그림 6는 시스템을 통해 완성된 전차 선로의 도면이다. 이러한 도면이 다수 모이면 하나의 구간이 완성되는 것이다. 도면은 선로와 전주를 중심으로 시공 시 필요한 구조물을 나타내고 있다. 화면 하단에서는 부품정보 테이블에서 얻어낸 값들을 표기하고 있으며, 부품 관리 테이블에서 얻을 수 있는 통계는 인체가 가능하도록 하고 있다. 이 외에도 시공 시 필요한 수치들은 모두 도면에 표기되어 있다. 파일은 AutoCad File의 포맷으로 자동 저장할 수 있도록 하였다.



<그림 6> 완성된 도면

5. 결론 및 향후 연구 방향

본 연구는 한국의 지형을 고려한 한국의 선로 환경에 맞는 선로 설계 시스템을 설계하고 구현하였다. 이것을 바탕으로 새로운 노선을 건설할 때 필요한 부분만 교체하여 계속적인 업그레이드가 가능할 것이다. 또한 이것을 개발하면서 축적한 기술과 방법론은 이와 유사한 다른 시스템을 개발하는 데에도 참고가 될 수도 있을 것이다.

향후 연구 방향으로서는 현 시스템에서 지원하지 않는 특수한 도면을 생성해 내는 기능을 추가해야 할 것이다. 지형의 특수성으로 인해 현 시스템에서 생성해 낼 수 없는 도면이 극소수이지만 존재하는데, 현재의 시스템에서는 이러한 도면을 제작할 수 없어서, 사용자가 목적하는 도면과 가장 유사한 도면을 제작한 후 직접 도면을 수정해야만 한다. 이러한 단점을 개선하여 보다 완전한 설계 시스템을 개발하여야 할 것이다.

참고문헌

- [1] 김지운, 유광균, 홍효식, 노성찬, “고속전차 선로 상세설계 S/W 구조”, 한국 철도학회 논문집 Vol.3, No.4, 2000
- [2] Ken Getz, Mike Gilbert, “(Inside Secret) VBA: Developer’s Handbook”, 삼각형, 1997
- [3] AutoCad Library Pattern in AutoCad2000 AutoCad Library
- [4] 한국 철도기술연구원, 고속전차 선로 상세설계 S/W 개발(II), 한국 고속철도 공단, 2001.12
- [5] David Sceppa, “Programing ADO”, 정보문화사