

Revit 기반 구조 모델링의 철근 배근을 위한 Dynamo 활용 방안 연구

강진옥^{1*}, 김재흥¹, 박준수¹, 윤희택²

¹(주)도화엔지니어링 철도부, ²한국철도기술연구원 첨단궤도토목본부

Dynamo Utilization for Reinforcement of Revit-Based Structural Modeling

Jin-Ook Kang^{1*}, Jae-Heung Kim¹, Jun-Soo Park¹, Hee-Taek Yoon²

¹Railway Department, Dohwa Engineering Co. Ltd

²Advanced Railroad Civil Engineering Division, Korea Railroad Research Institute

요약 첨단기술의 도입을 통한 생산성 향상을 위한 노력은 건설산업 전반에 걸쳐 지속적으로 추진되어져 왔다. 이러한 노력의 일환으로 Building Information Modeling(이하 BIM) 기술은 다양한 건설분야 실무에 적용되고 있다. 하지만 대부분 골조 모델링에 국한되어 있으며 BIM을 활용한 철근 모델링은 반복적인 수작업으로 인한 시간소요, BIM Tool에서 자체적으로 제공하는 철근 모델링 기능의 한계 등의 이유로 실무 적용이 제한되고 있다. 이에 본 연구는 파라메트릭 모델링에 비주얼 프로그래밍(Dynamo)을 결합·활용해 철근 배근 업무를 자동화하여 설계 실무자들의 BIM에 대한 접근성과 업무 효율성을 높이는 방법을 고안하였다. 이를 위해 토목 철근 콘크리트 구조물 중 비교적 형상이 복잡한 L형 옹벽을 Dynamo 알고리즘의 개발 대상으로 선정하고, 다양한 매개변수를 선정한 후 BIM 파라메트릭 모델링을 진행하였다. 또한, Dynamo와 Excel을 활용하여 BIM 객체에 철근 배근을 자동화하는 알고리즘을 구축하였다. 이를 통해 기존의 상용 BIM Tool에서 제공하는 기능을 이용하였을 때보다 작업시간을 현저히 절감하였고, 알고리즘의 일부만 수정하면 기둥, 보, 기초 등 다양한 부재나 유사 프로젝트에서 해당 알고리즘을 재사용하여 BIM 기반 철근 배근 모델링 업무의 생산성에 크게 기여할 수 있다.

Abstract Efforts to improve construction productivity by adapting advanced technologies have been made continuously throughout the construction industry. Building information modeling (BIM) is being applied in various construction fields. However, BIM application is limited mostly to structural frame modeling. In addition, the BIM tool has not been widely used in practice owing to its time-consuming and repetitive work and its limited rebar modeling functions. Therefore, this study proposes a method to increase engineers' accessibility and work efficiency in utilizing BIM by automating the rebar placement tasks via parametric modeling and visual programming (Dynamo). An L-shaped retaining wall, which has a relatively complex shape among the civil reinforced structures, was selected as the target of algorithm development and conducted the parametric modeling after selecting various parameters. In addition, an algorithm for the rebar arrangement automation on the BIM object was constructed using Dynamo and Excel. The results showed that working hours were lower than those taken using the rebar modeling function of the existing commercial BIM tools. Moreover, after modifying only a part of the algorithm, reusing these algorithms in various components or similar projects can contribute to the productivity of BIM-based rebar arrangement modeling tasks.

Keywords : BIM, Structural Modeling, Revit, Dynamo, Rebar Arrangement Automation

본 논문은 국토교통부 산하 국토교통과학기술진흥원 재원으로 한국철도기술연구원의 지원(No. 22RBIM-C158185-03)에 의해 수행되었음.

*Corresponding Author : Jin-Ook Kang(Dohwa Engineering Co. Ltd)

email: jin.kang@dohwa.co.kr

Received June 21, 2022

Accepted November 4, 2022

Revised September 13, 2022

Published November 30, 2022

1. 서론

1.1 연구배경 및 목적

3D Building Information Modeling(이하 BIM) 기반 모델링 기술은 건축, 토목 등 건설산업 전 분야의 설계 및 시공과정에서 적용되고 있다. 또한 프로젝트의 유형과 목적 또는 발주자의 요구 등에 따라 BIM이 적용되는 공종과 그 범위는 다양하며, 각 단계별, 공정별로 BIM 모델링 또한 상이하다.

최근까지 토목 분야에서도 BIM 시범사업이 수행되었고, 사업을 수행한 기업 및 관련 업체들은 일정 수준이상의 노하우를 축적하였다. 구조물을 모델링하고 수량을 산출하는 등의 프로세스에는 큰 문제를 겪고 있지 않으나 직접 또는 간접적으로 기존의 CAD를 활용하는 등 완전한 BIM 설계를 수행하고 있지는 않다[1].

BIM 지침 또는 발주처의 과업지시서 등을 살펴보면 철근 콘크리트 구조물의 경우 설계단계 이후에서는 설계 수준(Level of Detail, 이하 LOD) 350이상을 적용토록 되어 있어 철근 배근까지도 과업의 범위로 요구하고 있다. 기존 BIM 저작 도구에서 철근 배근 모델링 기능을 제공하지만 오랜 작업시간이 소요되고 많은 인력을 모델링에 투입하게 되어 BIM 모델 작성에 대한 생산성 저해 요인이 되며 실용적이지 못하다.

BIM 기술은 확장되어가고 있는데 정작 LOD에 맞도록 BIM 모델링을 하기 위해서는 수동으로 일일이 철근을 배근 작업해야 하는 실정이다. 물론, 각종 프로그래밍 언어를 활용하여 BIM 자동화를 연구한 프로그램도 있다. 하지만 이는 대부분 건축 분야에 한정되어 있으며 해당 프로그램을 제작할 때 전문적인 프로그래밍 지식을 요구하기 때문에 설계 실무자들이 원하는 환경에 맞게 알고리즘을 구축하기에는 제한이 있다.

이에 본 연구에서는 설계 실무자들이 쉽게 BIM 객체를 제어하고, 철근 배근을 자동화하여 작업 효율성을 증대시키는 방안을 제안하고자한다.

1.2 연구범위 및 방법

철근 콘크리트 설계는 골조형태 및 기능에 따라 콘크리트 부재, 주철근, 부철근, 띠철근 등의 구성요소가 조합된 3차원적인 설계라 할 수 있다. 즉 이러한 다양한 구성요소를 만들고 배치하는 것이 철근 콘크리트 구조 설계의 기본적인 형태라 할 수 있다. 철근 콘크리트 구조물은 건축물 외형에 비해 비교적 단순한 형태를 갖지만 두 가지 구조물의 연결부분은 배열 및 구성요소 등을 고려

하면 철근 배근이 매우 복잡하다. 이러한 형태는 BIM Tool로 일반적인 철근 배근은 가능하나 사용자가 짧은 시간에 정확한 형태를 구현한다는 것은 매우 어렵다. 특히 단면 형상이 불연속적으로 변화하는 경우, 단면 하나 하나를 모델링하는 것은 비효율적이다. 또한 단면마다 상세한 철근 배근 모델링을 위해 요구되는 시간과 노력에 비해 그 적용 효과는 높지 않으며, 설계변경 등의 변수에 일일이 대응하기도 어렵다고 할 수 있다[2].

본 연구에서는 비교적 간단한 선형 구조물인 옹벽을 대상으로 반복적인 수작업을 자동화하고, 설계변경에 유연하게 대응하기 위하여 Dynamo를 활용한 철근 배근의 파라메트릭 기반 자동화 모델링 방법을 제안하고자 한다.

2. 이론 및 선행연구 고찰

2.1 BIM 모델링 방식

BIM 모델링 방식은 다이렉트 모델링, 파라메트릭 모델링으로 구분된다.

다이렉트 모델링이란 이미 프로그램에서 형상이 결정되어있는 모델을 활용하는 기법을 의미한다. 정해진 변수의 값만 변경하여 사용하면 되기 때문에 작업속도가 빠르지만 원하는 형상을 직접 만들거나 필요한 변수를 직접 생성할 수 없어 사용할 수 있는 상황이 제한적이다[3].

파라메트릭 모델링은 형상을 구성하는 요소를 사용자가 직접 매개변수로 지정하여 모델링하는 방법이다. 이를 통해 사용자는 매개변수의 값을 변경하여 모델의 형상을 제어할 수 있고 형상뿐만 아니라 BIM 정보를 수정하거나 재활용할 수 있기 때문에 BIM 업무 생산성을 높일 수 있는 모델링 기술이다[4].

본 연구에서는 직접 매개변수를 생성하고 객체를 작성하는 파라메트릭 모델링 방법을 활용하였다.

2.2 Revit 및 Dynamo에 대한 고찰

BIM 소프트웨어는 Autodesk Revit(이하 Revit), Tekla, Allplan 등 다양하다. Revit은 BIM 소프트웨어 중 하나로, 프로그램 내에서 구조, 건축, MEP 등 다양한 분야간 협업이 가능한 솔루션을 제공한다. Fig. 1에서 볼 수 있듯, NBS (National Building Specification)에서 발간한 '10th Annual BIM Report 2020'에 의하면, Revit이 가장 선호하는 BIM 소프트웨어이며 시장 점유율도 가장 높다[5].

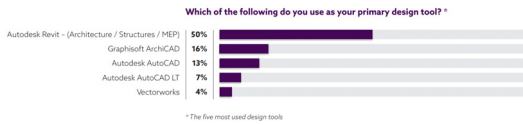


Fig. 1. Primary BIM tools - TOP 5

Revit에서는 Dynamo라는 비주얼 프로그래밍 개발 도구를 자체적으로 제공한다. Dynamo는 특정 기능을 가진 노드끼리 연결하여 알고리즘 구축이 가능하다. 사용자가 논리적 구조만 명확하게 세운다면 별도의 프로그래밍 지식이 없어도 쉽게 알고리즘을 생성할 수 있다. Dynamo는 단순히 형상만 디자인하는 것이 아니라 Revit의 매개변수 또는 정보를 연동하고 제어할 수 있다. 특히, 프로그래밍 언어인 C#과 Python도 지원되기 때문에 다른 프로그램과 연계하는 확장성도 가진다. 이러한 이유로 Dynamo는 부재의 자동 모델링, 물량 검증, 자동 물량 산출, 부재 배치, 도면화, 시각화, 각종 오류체크 등 다양한 부분에 활용되고 있다[6].

2.3 선행연구의 고찰

건설업계에서 BIM의 중요성을 인식하고 이의 활용이 확산되고 있지만, 그중에 Dynamo를 적용한 자동화 연구는 슬릿 케이스 모델링 자동화[7], 항만 시설물 모델링 자동화[8] 등 BIM 객체의 형상 모델링 자동화에 대한 연구가 대부분이며 Dynamo를 활용한 철근 배근 자동화에 관한 연구는 확인되지 않고 있다.

또한, 반복적인 수작업으로 생산성이 떨어지는 BIM 철근 모델링을 자동화하고자 하는 연구도 지속되고 있지만, 대부분 기둥, 보, 슬래브 등 건축 분야에 한정되어 있으며 전문적인 프로그래밍 언어를 사용하여 개발되기 때문에 설계 실무자들이 직접 업무환경에 맞게 수정하여 활용하기에 어려운 실정이다[9].

따라서 본 연구에서는 설계 실무자들이 객체 모델링부터 철근 배근 작업까지 하나의 프로그램 내에서 통합적으로 수행하며 생산성을 향상 시키는 방안을 제시하고자 한다. 이를 위해 Dynamo와 Excel을 활용한 BIM기반 철근 배근 설계 자동화 알고리즘을 개발하고자 한다.

3. 파라메트릭 모델링 및 철근 배근 모델링

3.1 매개변수 설정

한국철도시설공단에서 제공하는 철도표준도(노반편)[10]

에서 옹벽은 Fig. 2와 같이 중력식, 반중력식, L형, 역T형 4가지 형태로 구분되어있다.

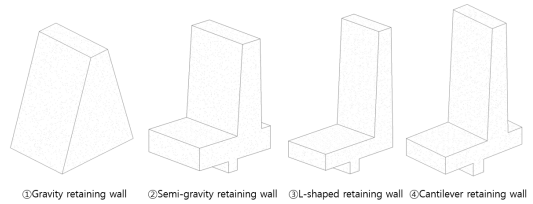


Fig. 2. Four types of retaining walls

본 연구는 L형 옹벽을 대상으로 이루어졌으며 Table 1과 Fig. 3에서 보는 바와 같이 매개변수를 정의하였다.

Table 1. Parameter settings (1)

Division	Type	Variable	Kind	Unit
Parameter	User input variable	Ⓐ Foundation thickness	Length	mm
		Ⓑ Retaining wall height	Length	mm
		Ⓒ Retaining wall slope width	Length	mm
		Ⓓ Retaining wall front slope width	Length	mm
		Ⓔ Retaining wall front position width	Length	mm
		Ⓕ Shear key thickness	Length	mm
		Ⓖ Shear key front position width	Length	mm
		Ⓗ Shear key width	Length	mm

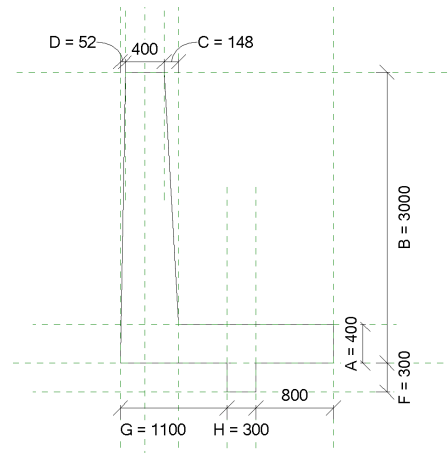


Fig. 3. Parameter settings (2)

3.2 Revit 철근 모델링

일반적으로 Revit에서의 철근 모델링은 Fig. 4와 같은 방식으로 이루어진다.

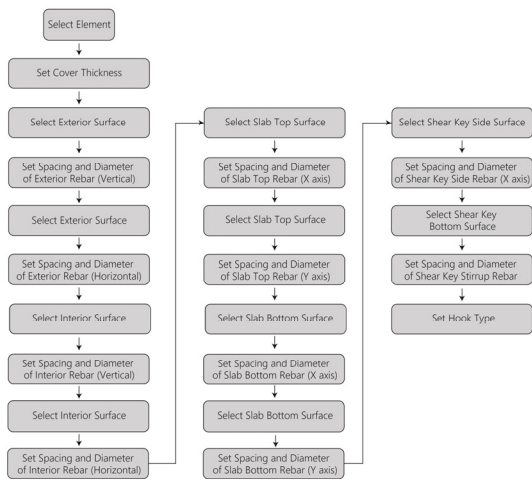


Fig. 4. Process of rebar arrangement automation by Revit

Autodesk사에서 제공하는 철근 배근 튜토리얼에도 철근이 배근될 객체를 일일이 선택하고 수동으로 철근을 모델링하는 방법을 제시하고 있다[11].

Fig. 5는 Revit에서 자체적으로 제공하는 철근 배근 UI를 활용하여 객체의 한 면에 철근을 모델링하는 방법을 나타낸다. 객체 자체에 피복 두께를 설정하고, 철근이 배치될 면을 선택 후, 배근될 철근의 직경과 위치를 일일이 지정하여 모델링한다. 하나의 면에 대한 철근 배치 작업이 완료되면 철근 단부의 후크 형태를 지정해준다. 이 방법을 객체별로, 배근될 위치별, 직경별로 반복 수행해야한다.

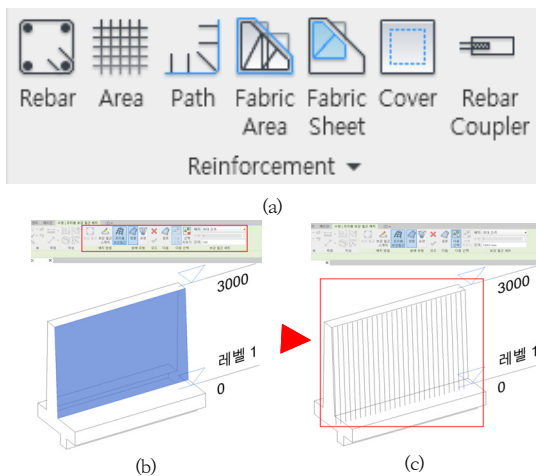


Fig. 5. Revit rebar modeling process
(a) Revit rebar function UI (b) Surface Selection
(c) Manual modeling of rebar

해당 방식을 적용하면 철근 작성시 별도의 지식이 없어도 Revit만 활용하여 직관적으로 모델링 할 수 있다는 장점이 있지만, 설계 변경시 수정이 어렵다는 단점이 있다.

또한, Revit에서 철근은 고유 ID를 가진 객체에 종속되어 모델링 된다. 그렇기 때문에 동일한 철근 정보를 갖는 같은 유형의 웅벽이라도 해당 웅벽의 개수만큼 일일이 철근 정보를 수정하는 작업이 필요하다. 프로젝트의 규모가 커지고 웅벽의 배근 유형이 다양해질수록 수정에 필요한 작업시간도 비례하여 늘어난다.

3.3 Dynamo를 활용한 철근 배근 자동화

Dynamo를 활용한 철근 배근 작업을 하는 실무자의 작업 프로세스는 Fig. 6과 같다. 2D 도면을 확인하여 사용할 변수를 지정하고 Revit에서 매개변수를 생성하여 파라메트릭 모델을 작성한다. 모델의 유형별 철근 직경 및 배근 간격을 Excel 데이터로 작성한다. 연구에서 제시한 Dynamo 스크립트를 실행하면 자동적으로 Excel로부터 데이터를 추출, 이에 맞는 철근을 모델링한다.

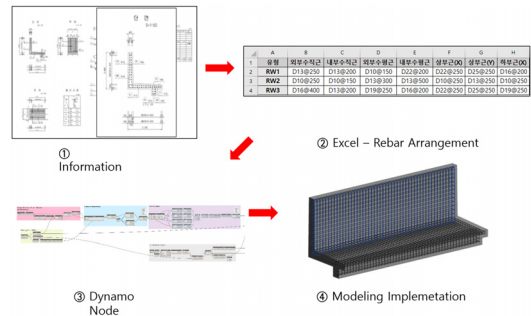


Fig. 6. BIM model implementation process

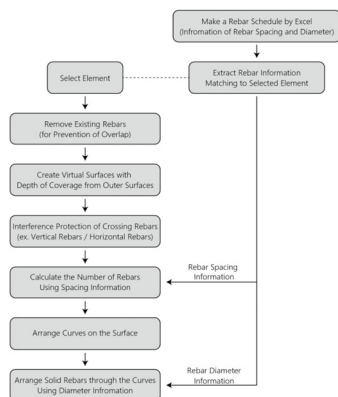


Fig. 7. Process of rebar arrangement automation by dynamo

Fig. 7는 Dynamo와 Excel을 활용한 알고리즘의 구동 프로세스를 나타낸다. Excel에 철근 정보를 입력하면 Dynamo에서 철근의 직경과 간격 데이터를 추출하고, 해당 데이터에 맞게 모델 객체에 Curve를 배치 후 철근 형상으로 자동적으로 모델링되도록 알고리즘을 구축하였다.

3.3.1 웅벽 유형별 철근 정보 입력

Fig. 8과 같이 Excel에 웅벽 유형에 따른 철근의 직경과 간격 정보를 입력하고 직관적으로 확인할 수 있도록 Excel 시트를 구성하였다.

TYPE	EX_VER	IN_VER	EX_HOR	IN_HOR	TOPN	TOP1	BOTN	BOT1	SHEAR_KEY_VER	SHEAR_KEY_HOR
RW1L-type	D13@250	D13@200	D10@150	D22@200	D22@250	D16@200	D16@250	D16@250	D16@250	D16@250
RW2L-type	D16@250	D10@150	D13@300	D13@300	D10@250	D13@250	D16@250	D13@250	D10@250	D13@250
RW3L-type	D16@400	D13@200	D16@300	D16@200	D22@250	D25@250	D16@250	D13@150	D22@250	D22@250
RW4L-type	D13@250	D13@200	D10@150	D22@200	D22@250	D16@200	D16@250	D16@250	D16@250	D16@250
RW5L-type	D16@250	D10@150	D13@300	D13@300	D10@250	D13@250	D16@250	D13@250	D10@250	D13@250
RW6L-type	D16@400	D13@200	D16@300	D16@200	D22@250	D25@250	D16@250	D13@150	D22@250	D22@250
RW7L-type	D13@250	D13@200	D10@150	D22@200	D22@250	D16@200	D16@250	D16@250	D16@250	D16@250
RW8L-type	D16@250	D10@150	D13@300	D13@300	D10@250	D13@250	D16@250	D13@250	D10@250	D13@250
RW9L-type	D16@400	D13@200	D16@300	D16@200	D22@250	D25@250	D16@250	D13@150	D22@250	D22@250
RW10L-type	D13@250	D13@200	D10@150	D22@200	D22@250	D16@200	D16@250	D16@250	D16@250	D16@250
RW11L-type	D16@250	D10@150	D13@300	D13@300	D10@250	D13@250	D16@250	D13@250	D10@250	D13@250
RW12L-type	D16@400	D13@200	D16@300	D16@200	D22@250	D25@250	D16@250	D13@150	D22@250	D22@250
RW13L-type	D13@250	D13@200	D10@150	D22@200	D22@250	D16@200	D16@250	D16@250	D16@250	D16@250
RW14L-type	D16@250	D10@150	D13@300	D13@300	D10@250	D13@250	D16@250	D13@250	D10@250	D13@250
RW15L-type	D16@400	D13@200	D16@300	D16@200	D22@250	D25@250	D16@250	D13@150	D22@250	D22@250

Fig. 8. Excel data input

3.3.2 Excel 데이터 추출 모듈

Dynamo에서 Fig. 9와 같이 'Data.ImportExcel' 노드와 Python Script를 사용하여 객체의 'Family Name'과 일치 여부를 판단하여 객체의 Family name에 맞는 철근 직경, 간격 데이터를 추출하는 Dynamo Script를 구성하였다.

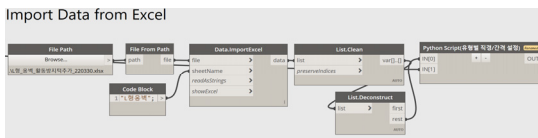


Fig. 9. Excel data extraction algorithm

```

Python Script(유형별 직경/간격 설정)
1. import sys
2. import clr
3. clr.AddReference('ProtoGeometry')
4. from Autodesk.DesignScript.Geometry import *
5.
6. dataEnteringNode = IN
7.
8. dim_list = [[] for x in range(len(IN[0])]
9. spc_list = [[] for x in range(len(IN[0])]
10. name_list = [v[0] for v in IN[0]]
11. for i,v in enumerate(IN[0]):
12.     for p,q in enumerate(v[1:]):
13.         dim_list[i].append(q.split("@")[0])
14.         spc_list[i].append(q.split("@")[1])
15.
16. for m,n in enumerate(name_list):
17.     if n==IN[1]:
18.         name_out = [n]
19.         dim_out = dim_list[m]
20.         spc_out = spc_list[m]
21.
22. OUT = [name_out, dim_out, spc_out]
    
```

Fig. 10. Listing diameter and spacing data of rebar

Excel 시트에 철근 정보를 '철근직경@배근간격'의 형식으로 입력하면 '@'를 기준으로 데이터를 구분하고 해당 데이터를 각각 직경 리스트, 간격 리스트에 구분하여 저장할 수 있는 알고리즘을 구성하였다. 해당 알고리즘을 구현하는 Python Script는 Fig. 10과 같다.

3.3.3 피복 두께 적용한 Surface 생성 모듈

객체의 외부면을 간격띄우기(Offset)한 후에 해당 Surface를 그대로 철근이 배근될 영역 Surface로 사용하면 Fig. 11 (a)에 표기된 부분에는 철근이 누락된다.

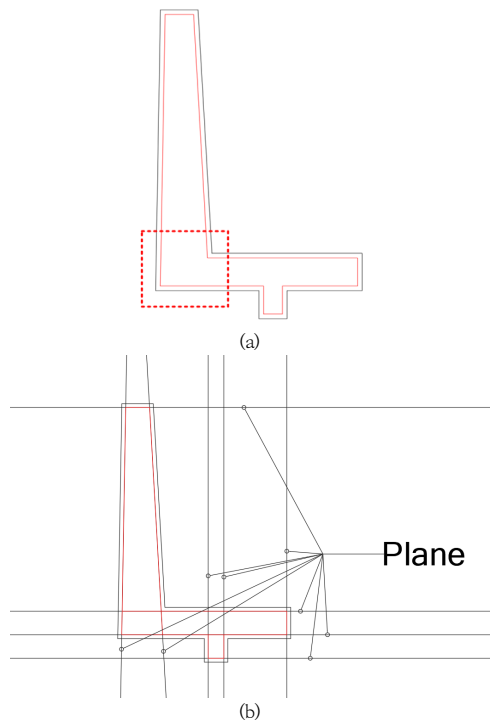
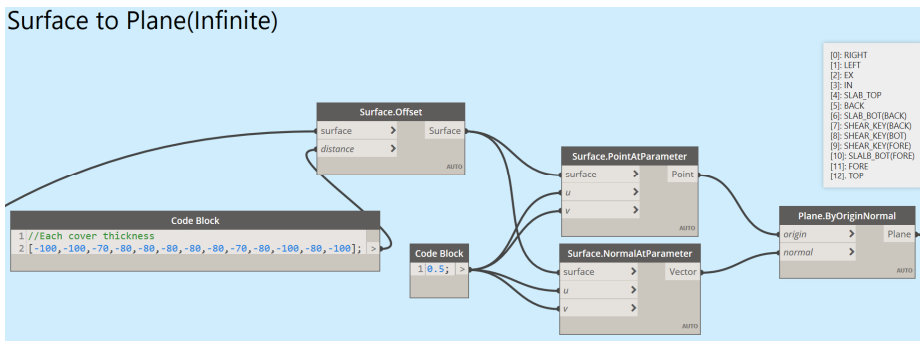
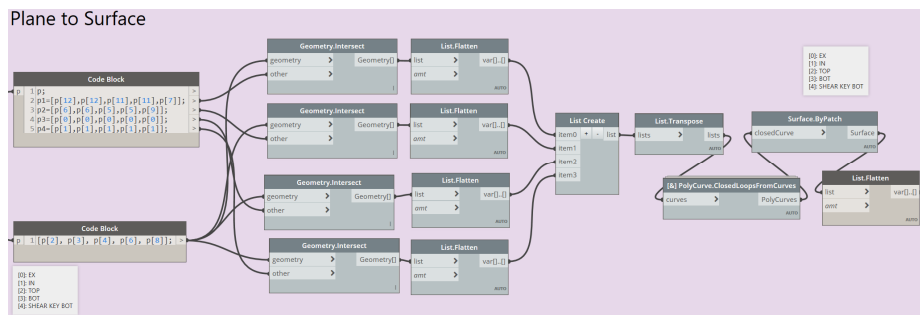


Fig. 11. Surface and plane
(a) Original surface (b) Revised surface

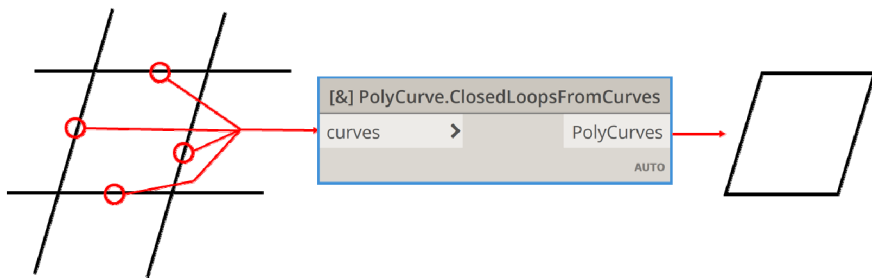
철근 누락을 방지하기 위해 배근 범위에 맞는 새로운 Surface를 생성하는 알고리즘을 구축하였다. Fig. 12 (a)는 피복 두께만큼 Offset된 Surface를 Plane(무한평면)으로 치환해주는 모듈을 나타낸다. Fig. 12 (b)와 같이 Plane간 교차선들을 추출하고 다시 Surface화하는 Dynamo Script를 구성하였다. Fig. 12 (c)는 교차선들이 만나 닫힌 Polycurve를 구성하는 기능을 가진 노드를 나타낸다[12]. Fig. 11 (b)는 해당 과정에서 생성된 Surface와 Plane을 입면에서 나타낸 모습이다.



(a)



(b)



(c)

Fig. 12. Surface creation

(a) Convert surface into plane (b) Convert plane into surface (c) node - 'Closed loops from curves'

3.3.4 배치 철근 개수 산출 모듈

철근 개수는 길이를 배근 간격으로 나눈 후 소수점을 반올림한 후 1을 더하여 계산한다. 여기서 1을 더하는 이유는 배근 거리를 간격으로 나눌 때 단부의 시작점에 들어가는 철근이 계산되지 않기 때문이다. 또한, 수량의 계산은 소수의 이하 1위까지 구하고, 끝수는 반올림한다 [13]. 해당 식의 형태는 Eq. (1)과 같다.

$$n = L/s + 1 \quad (1)$$

Where, n denotes number of rebar, L denotes distance of rebar arrangement, s denotes rebar spacing

Surface에 X방향, Y방향으로 배치되는 철근을 구분하고 개수를 산출하는 모듈을 Fig. 13와 같이 Python Script로 구현하였다.

```

Python Script (# of rebar)
1: # Python 스크립트 및 DynamoScript, 객체지향 프로그래밍
2: import sys
3: import clr
4: import math
5: clr.AddReference("ProtoGeometry")
6: from Autodesk.DesignScript.Geometry import *

dataEnteringMode = IN

edge_len = IN[0]
str_spacing = IN[1]

2: # 소정된 길이 - (소정길이/수평간격(소수점 올림)) + 1
3: # 소정된 길이 - (소정길이/수직간격(소수점 올림)) + 1
spacing = [float(v) for v in str_spacing]

P = math.ceil - 소수점 올림
num = [math.ceil(p/q) + 1 for p, q in zip(edge_len, spacing)]

OUT = [num[0], num[1], num[2], num[3]]

flip_false = [num[0], num[1], num[2], num[3]]
flip_true = [num[2], num[3], num[1], num[0]]

OUT = [flip_false, flip_true]
    
```

Fig. 13. Number of rebar calculation algorithm

3.3.5 Curve 배치 및 Solid화 모듈

앞서 3.3.4 모듈에서 간격 데이터를 기반으로 산출된 개수의 Curve를 3.3.3에서 생성한 Surface에 배치하였다. 또한 배치된 Curve를 철근의 직경 데이터에 맞게 3D 철근 모양으로 Solid화 시키는 Dynamo Script를 구성하였다. 마지막으로 Solid화 한 철근 모델의 단부에 후크 여부를 판단하여 적용하는 모듈을 작성하였다. 해당 과정을 Fig. 14과 같이 Dynamo Script를 구성하였다.

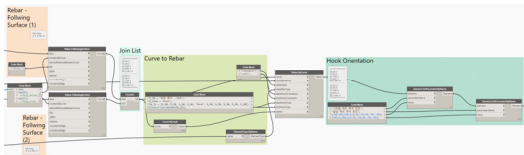


Fig. 14. Curve Arrangement and Convert Curve into Solid

4. 사례 검증

프로젝트에서는 다양한 유형의 옹벽과 여러개의 옹벽 모델이 필요하고, 철도표준도(노반편)에는 15가지 종류의 L형 옹벽이 명시되어있다. 따라서, Fig. 15와 같이 15개의 L형 옹벽을 모델링하고 해당 알고리즘을 적용하여 작업시간을 비교, 검증하였다.

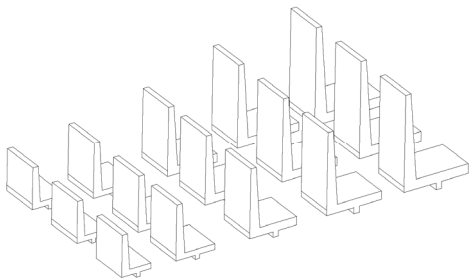


Fig. 15. Parametric models of retaining walls

작업자 한명이 Revit 내부 배근 기능만으로 15개의 옹벽에 철근 모델링을 했을 때, 약 12시간의 작업시간이 소요되었다.

반면, Dynamo와 Excel을 활용한 철근 배근 자동화 스크립트를 활용하였을 때, 알고리즘 작성시간을 제외하고 약 1시간의 작업시간이 소요되었다. 알고리즘만 구축된다면 Excel 데이터를 입력하고 객체를 선택하여 스크립트를 실행하는 작업만 반복하면 되기 때문에 작업시간과 노력을 절감할 수 있었다.

설계변경이 필요하다고 가정하였을 경우에도 유사한 결과가 나왔다. 15개의 옹벽의 철근 직경과 간격을 모두 다른 값으로 변경된다고 가정하고 진행하였다.

Revit 내부 배근 기능만을 활용한 방식에서는 해당 옹벽의 변경될 철근을 선택한 후, 일일이 철근의 직경 유형을 변경하고 간격을 조절해 주었다. 하나의 L형 옹벽을 수정하는데 대략 15분이 소요되었다. 단순 계산으로 15개의 옹벽의 철근을 수정하는데 약 225분이 소요된다.

Dynamo와 Excel을 활용한 수정작업은 다음과 같은 방식으로 진행되었다. Excel 시트에서 모든 유형의 옹벽에 대해 철근 직경과 배근 간격을 수정하고 객체 선택 후 Dynamo를 실행시키면 작업이 완료된다. 해당 수정 방식으로는 15개의 옹벽 철근을 수정하는데 약 20분이 소요되었다.

Table 2와 같이 Dynamo와 Excel을 활용하여 철근 자동배근 알고리즘을 구축해두면 알고리즘을 작성하는 시간이 소요되지만 작업 효율을 획기적으로 증가시킬 수 있다.

작업 기간은 PC 성능과 작업자의 작업속도에 따라 다소 차이가 발생할 수 있지만, 본 연구에서 동일한 작업자가 같은 사양의 PC로 진행하였기 때문에 해당 변수는 고려하지 않았다.

Dynamo 알고리즘 파일과 Excel 양식을 작성해두면 추후 유사 프로젝트를 진행시 재사용할 수 있어 새롭게 알고리즘을 작성하는 시간을 절감할 수 있다.

Table 2. Work efficiency

	Modeling Method	Working Time	Work Efficiency (Revit / Dynamo)
First modeling	Revit	720 min	1200%
	Dynamo+Excel	60 min	
Chagne Requirement	Revit	225 min	1125%
	Dynamo+Excel	20 min	

5. 결론 및 향후 연구 진행 방향

최근 BIM 전면 도입이 추진되고 공공공사에 BIM 도입이 의무화되고 있다. 건설분야를 필두로 BIM 기술의 도입이 추진되고 있다. 철근 정보 모델링은 실무자들이 Revit 내부 배근 기능으로 수동으로 철근을 배근하거나 별도의 개발된 프로그램을 사용하고 있다. 하지만 Revit 내부 기능으로 모델링시 많은 시간과 노력이 소요되고, 별도로 프로그램을 개발하기 위해서는 C, C# 등 전문적인 프로그래밍 지식이 필요하다. 기존에 개발된 철근을 자동 배근해주는 프로그램도 있지만 이는 건축분야에 특화되어있어 사용이 제한적이며 설계 실무자가 원하는 환경에 맞춰 수정하여 사용할 수도 없는 실정이다.

본 연구는 Dynamo와 Excel을 활용하는 방식으로 철근 배근 자동화 분야에 접근하여 알고리즘 개발을 진행하였다. 실무자들의 접근성과 활용성을 높이고자 Revit 기반의 Dynamo와 Excel을 활용하였다. 그리고, 연구를 통해 내린 결론은 다음과 같다.

첫째, Revit의 내부 기능만 사용하였을 때보다 작업 효율을 높일 수 있다. 처음 알고리즘을 작성에는 대략 2일이 소요되었지만 한번 작성해두면 유사 프로젝트에 적용이 가능하며 작업 효율을 높일 수 있다.

둘째, 설계변경 시 수정 작업시간이 절감된다. Revit의 내부 배근 기능만 사용하는 방식과 본 연구에서 개발한 Dynamo 알고리즘을 사용하였을 때 수정 작업시간은 약 10배 정도의 차이가 있었다. 설계가 변경되면 전자의 방식은 해당 객체를 찾아 철근을 각각 선택하여 정보를 변경하거나 재모델링을 해야한다. 3.2절에서 언급하였듯이 Revit 철근은 고유 ID를 가진 객체에 종속되는 개념이다. 그렇기 때문에 같은 유형의 옹벽 객체가 2개 이상 있어도 각각의 철근을 따로 수정해야한다. 이에 반해, 해당 Dynamo 알고리즘을 활용 시 Excel 데이터 수정 후, Dynamo를 실행하는 것만으로 철근 모델의 변경이 가능해 작업시간을 단축할 수 있다. 실제 작업시 철근 배근 작업시간이 단축되었고, 설계변경시에도 작업시간 절감이 가능하다는 것을 확인하였다.

그리고 향후 연구에서는 몇 가지 사항을 개선하고자 한다.

첫째, 배근 자동화 라이브러리를 추가 확장하고자 한다. 본 연구에서는 해당 알고리즘을 L형 옹벽에 적용한 예시만 기술하였지만, 개발된 Dynamo Script를 일부 수정하게 되면 기둥, 교대, 교각 등 다양한 토목 구조부재에 대한 철근 배근 자동화 Script도 쉽게 구성할 수 있

다. 따라서 본 연구에서 개발된 알고리즘을 기반으로 부재별 Script를 만들어 라이브러리를 구축하게되면 설계 실무자는 단순하게 Excel 데이터 수정과 Dynamo Script 실행만으로 다양한 부재에 대하여 철근을 자동적으로 배근할 수 있다.

둘째, Excel 파일에 철근 항복강도(Fy)에 따른 후크길이 및 이음/정착 길이 데이터를 추가하여 보다 정확한 물량을 산출하고자 한다. 본 연구에서는 약산식으로 계산된 이음/정착 길이가 적용되어있다. 하지만 철근 항복강도에 따라 후크길이 및 이음/정착 길이를 산정한다면 보다 정확한 물량 데이터 산출이 가능하다.

셋째, BIM 모델 수정 시, 철근 배근이 실시간으로 수정되도록 알고리즘을 개선하고자한다. 연구에서는 철근 배근을 자동화하여 작업시간을 감소시켰다. 추가적으로 모델의 변경에 따라 실시간으로 배근도 수정되도록 개선하여 모델 수정 후 다시 Dynamo Script를 실행시키는 시간을 절감하고자한다.

References

- [1] H.L. Choi, K.S. Jang, H. Kim, M.H. Lee, M.S. Park, "Algorithm-based Railway Tunnel BIM Design Considering Railway Alignment", *Journal of KIBIM*, Vol.9, No.1 pp.1-10, Nov, 2019
DOI: <https://doi.org/10.13161/kibim.2019.9.1.001>
- [2] J.H. Rho, M.S. Park, H.S. Lee, S.H. Nam, "An Analysis of BIM Application for Rebar and Its Efficacy - Based on Malaysia Case Study", *Autumn Annual Conference of AIK, 2018*, Architectural Institute of Korea, Vol.38, No.1 pp.680-683, Apr, 2018.
- [3] S.M. Jeon, D.W. Kim, BIM parametric modeling, Technical Report, Ssangyong Engineering & construction, Korea, pp.20-24.
- [4] M. Ringsven, D. Bond, "A Study on the Application of BIM-enabled Interior Panel Design by the control of Parametric" *Journal of the Korean Institute of Interior Design*, Vol.25, No.6, pp.70-78, Dec. 2016. DOI: <https://doi.org/10.14774/JKIID.2016.25.6.070>
- [5] R. Waterhouse, 10th Annual BIM Report 2020, Technical Report, National Building Specification, U. K, pp.25.
- [6] S. Mengana, T. Mousiadis, Parametric BIM: Energy Performance Analysis Using Dynamo for Revit, Master's thesis, KTH Royal institute of Technology, Stockholm, Sweden, pp.1-55, 2016.
- [7] H.S. Kim, H.M. Lee, I.S. Lee, "Development of Automatic BIM Modeling System for Slit Caisson", *Journal of the Korea Academia-Industrial*, Vol. 21,

No. 11 pp. 510-518, 2020.

DOI: <https://doi.org/10.5762/KAIS.2020.21.11.510>

- [8] H.S. Kim, H.M. Lee, I.S. Lee, "A study on Development of Practical BIM Library for Automated BIM Design in Port and Harbor Infrastructure", *Journal of KIBIM*, Vol. 9, No. 4 pp. 21-30, 2019.
DOI: <https://doi.org/10.13161/kibim.2019.9.4.021>
- [9] C.H. Song, C.K. Kim, S.E. Lee, H.C. Choi, "Establishment of Rebar Quantity Estimation in BIM-based Initial Design Phase", *Computational Structural Engineering Institute of Korea*, Vol. 29, No. 5 pp. 447-454, 2016.
- [10] Korea Rail Network Authority, "Railway Standard Drawings: Roadbed", Korea, CRR5501-CRR5515, 2015
- [11] Autodesk, Place Rebar, Autodesk, c2022 [cited 2022 August 19], Available From: <https://knowledge.autodesk.com/support/revit/learn-explore/caas/CloudHelp/cloudhelp/2020/ENU/Revit-Model/files/GUID-D61AF551-AD15-4FEC-966F-C048E0885212-htm.html> (accessed Sep. 07, 2022)
- [12] Dynamo, v.2.10, Package Name: "Ampersand", 2022 January 25.
- [13] Seoul Housing & Communities Corporation, "Estimate Criteria for Building Construction", Korea, pp.2, 2012.

강 진 옥(Jin-Ook Kang)

[정회원]



- 1987년 2월 : 서울대학교 농공학과 학사
- 1989년 2월 : 한국과학기술원 토목공학과 석사
- 1989년 3월 ~ 1992년 9월 : 한국 전력기술(주) 사원/대리
- 1992년 9월 ~ 1996년 1월 : (주) 대우엔지니어링 대리/과장

- 2001년 8월 : 조지아공대(美) 토목공학과 박사
- 2002년 1월 ~ 2016년 12월 : 삼성물산 건설부문 차장/부장/상무, 토목엔지니어링팀 (BIM그룹 + 시공ENG그룹) 팀장 역임
- 2016년 12월 ~ 현재 : 도화엔지니어링 철도부 부사장, 철도.메트로 프로젝트 입찰.수행 관리 및 철도인프라 BIM 라이브러리 확장·개발 연구과제 (국가 RnD) 수행

<관심분야>

토목BIM, 철도BIM, 토목인프라 설계 및 계획

김 재 흥(Jae-Heung Kim)

[정회원]



- 2006년 2월 : 우송대학교 건축학과 (학사)
- 2006년 8월 ~ 2009년 5월 : 한아 종합건설 BIM설계팀 대리
- 2009년 5월 ~ 2013년 4월 : 삼성 물산 DDP현장 BIM설계팀 대리

• 2014년 5월 ~ 2017년 8월 : 현대건설 NMOQ현장 BIM 설계팀 과장

- 2019년 5월 ~ 2020년 5월 : 포치건축 BIM설계팀 팀장
- 2020년 5월 ~ 현재 : 도화엔지니어링 철도부 차장

<관심분야>

BIM, 설계BIM, 시공BIM, BIM 프로세스

박 준 수(Jun-Soo Park)

[정회원]



- 2020년 8월 : 한양대학교 건설환경공학과 (학사)
- 2020년 10월 ~ 2021년 12월 : ㈜창소프트아이엔아이
- 2022년 1월 ~ 현재 : 도화엔지니어링 철도부 사원

<관심분야>

BIM, Dynamo, 프로그래밍

윤 희 택(Hee-Taek Yoon)

[정회원]



- 1987년 2월 : 성균관대학교 토목공학과 공학석사
- 2000년 7월 : 한국철도기술연구원 선임연구원
- 2002년 8월 : 성균관대학교 토목환경공학과 공학박사
- 2008년 7월 ~ 현재 : 한국철도기술연구원 책임연구원

<관심분야>

토목BIM, 철도BIM