

Dynamo for Revit 在基坑围护结构建模中的应用

吴 樊 闫 智 孔祥平

(上海城建信息科技有限公司,上海 200120)

【摘 要】本文基于 Dynamo for revit 参数化建模的思想实现了快速、高效放置钢板桩和结构钻孔桩,并且快速准确赋值每个桩参数信息。与传统手动建模方法相比,参数化建模可实现快速建模并保证所建模型的精度,大幅度提高工作效率并保证工作质量,可将此方法推广到类似应用中。

【关键词】BIM; Dynamo; 参数化建模

【中图分类号】TU117 **【文献标识码】**A

【版权声明】本文被《土木工程信息技术》、中国知网重要会议论文全文数据库(CPCD)收录上网,未经授权严禁登载。

1 BIM 的发展现状

国内外近几年都在大力推广 BIM 技术,这个以数字化和信息化集成的技术在国内外都得到了快速认可^[9]。BIM 技术集成了工程全生命周期的信息,可以帮助工程技术人员对工程项目的具体信息准确把控,使工程各个阶段紧密联合,BIM 技术实施以三维数字设计与工程软件作为支撑,二者形成了可视化的数字建筑模型^[1]。

可视化的数字模型是 BIM 技术中应用的基础,建筑信息模型几何信息与非几何信息的完整度、精细度和准确度是 BIM 技术应用的关键^[11]。Autodesk Revit 是一款三维建模软件,在国内市场应用最为广泛,软件可以进行建筑、结构、风系统、水系统、电系统、场布等建模工作,而其自带的参数化功能可实现构件的参数化生成、编辑及属性赋值等功能,可以大大提高工作效率,提升建筑信息模型的精度与品质。目前,Revit 现有的建模方式有四种,自由形式建模,自带族库建模,外部引入族建模,Dynamo for Revit 参数化建模^[2]。

Dynamo 是一个基于 Revit 的可视化编程平台,Dynamo 为 BIM 工程师提供了一个可视化编程工

具^[10],让 BIM 工程师通过定义程序流程,探索参数化的方案设计和自动化建模与模型检查工作流^[3]。Dynamo 可以根据读取到的线路信息生成、调整和放置 Revit 的族^[7],更快地解决设计细节,通过 Dynamo 可以实现盾构隧道模型、地质模型、桥梁模型等复杂模型的快速创建。

2 项目概况

钢板桩以其承载力好,水密性好,耐久性好,施工简便环保,作业高效并可回收反复使用的特点,经常使用在土木工程中。在进行围护结构建模的时候需要放置钢板桩,同一项目所需要放置钢板桩的样式通常是一样的,但是每个钢板桩的尺寸差异较大,对于一个 2 万 m² 的场地,放置钢板桩是一个麻烦而又耗时的过程。

项目要求:第一步按照表格给的尺寸创建三维模型;第二步将模型按照图纸放置到对应位置;第三步将表格中的参数输入到对应的钢板桩上;第四步根据图纸对钢板桩进行编码,便于后期放入到平台监测和维护;最后给每个钢板桩添加标注,便于后期出图。钢板桩的尺寸大小不一样需要建立多个模型,人工放置模型又是一件十分费时费力的任

【作者简介】 吴樊(1993 -),男,本科,助理工程师,主要研究方向:工程信息化;闫智(1993 -),女,硕士研究生,助理工程师,主要研究方向:工程信息化;孔祥平(1989 -),硕士研究生,工程师,主要研究方向:工程信息化。

务,针对项目特点,所以决定采用 Dynamo 参数化建模的方式生成模型。

3 技术路线

根据 Dynamo for Revit 参数化建模方法可以得技术路线如图 1 所示。

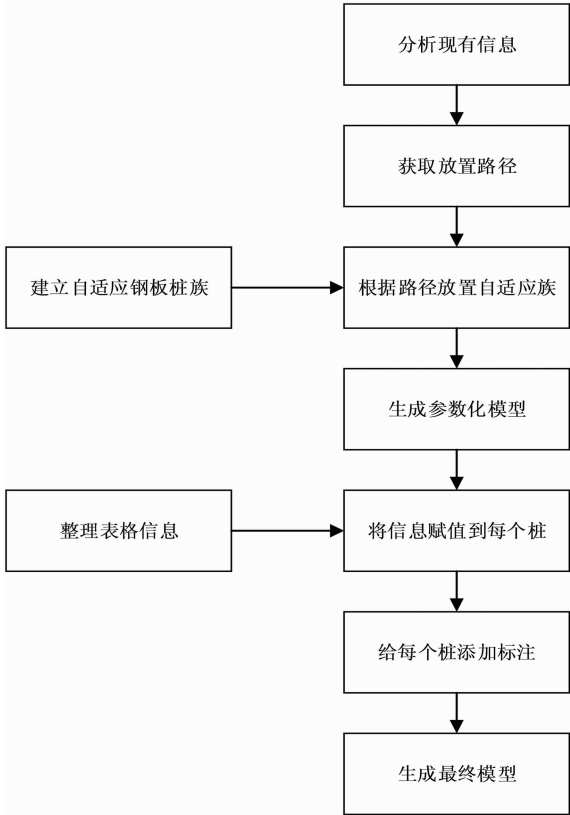


图 1 技术路线图

首先分析现有的项目概况知道需要解决什么问题即在已给的地形中放置钢板桩并赋予相应的属性信息,其次根据已有的项目信息获取可用的条件,即根据图纸获取放置路径,然后建立适合本项目特点的钢板桩自适应族,根据路径放置钢板桩,

接着整理需要复制的参数信息,用 Dynamo 所写的程序将整理好的信息赋值到每个模型中,最后给每个模型添加标注,生成最终的参数化模型。

4 基坑围护结构钢板桩的参数化模型构建方法研究

4.1 参数化设计方法

(1) 读取 CAD 并筛选出需要的线

用 Dynamo 读取导入到 Revit 中的 .dwg 文件的图层信息,对比结果筛选出需要的信息如图 2 所示。

(2) 取短线中点并两两成组

根据建模要求,结合建模思路,容易发现,我们只需获得放置 Sheetpile 的点,就可以放置 Sheetpile。对比图纸信息,我们需要读取图纸中短线的中点,然后将获取的点,按照 X 值递增顺序进行排序,之后再将排序后的点两两成组生成 List,每一组的两个点就可以精确放置一个 Sheetpile。

(3) 删除间断位置的点数组

根据实际情况如图 3 所示,Sheetpile 的放置路径不是一条连续的线,而是多条间断路径,所以需要将间断的地方从第 2 步中生成的 List 中去掉,如图 4 所示。

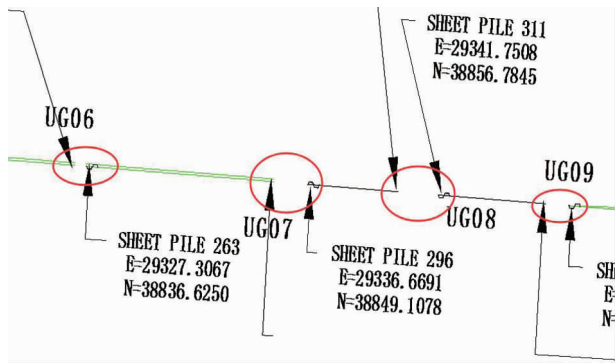


图 3 放置路径

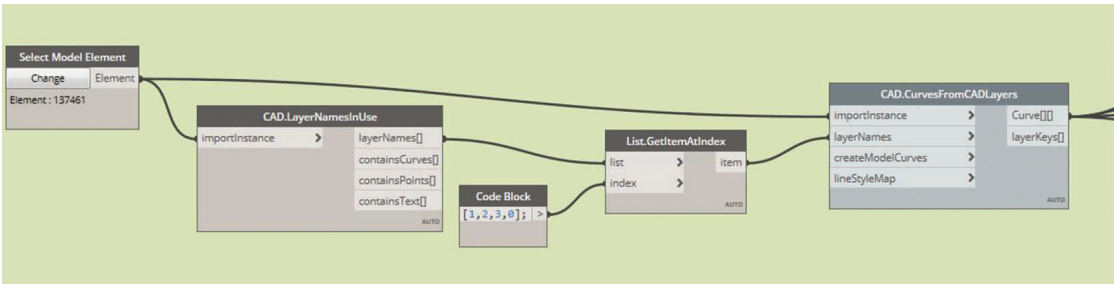


图 2 提取图层信息

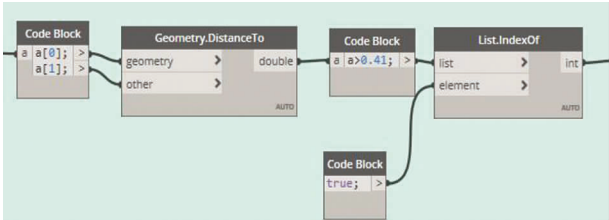


图 4 去除间断点

(4) 奇偶分组

Sheetpile 的方向是对称放置的,现在有两种方式放置,一是在 Revit 中用 Dynamo 参数化建模生成全部 Sheetpile,然后手动调整为对称模型,二是用 Dynamo 直接生成对称模型。因为用 Dynamo 生成也就是奇数的数组和偶数的数组处 Sheetpile 的方向是不一样的,所以要对第 2 步生成的 List 进行奇偶分组。Sheetpile 整体放置顺序由 X 轴从小到大放置,奇数组放置顺序为左边点 1 到右边点 2。偶数组是对称的,所以放置顺序为右边点 1 到左边点 2,如图 5 所示。

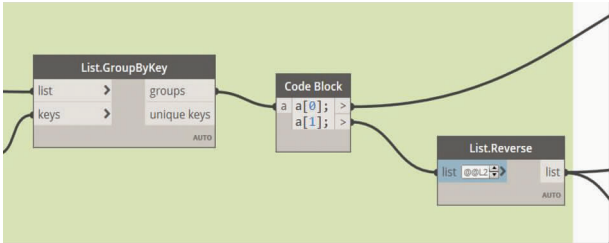


图 5 偶数组倒序

(5) 放置自适应族

Sheetpile 是自适应族,将表格按照 Mark 顺序重新排序,放置族。族有四个参数,其中每个 Sheetpile 相对应的参数不一样,将整理好的表格中的数据,按照参数名称为列,提取出来,并赋值到对应的 Sheetpile 相对应的参数上。

(6) 用标签族添加标注,然后手动调整标签族的位置。

生成模型结果如图 6 所示,模型参数信息如图 7 所示。

4.2 参数化建模方式与手动建模的效率对比

我们用 Dynamo 解决参数化建模的目的是为了保证工作质量的同时提高工作效率,如图 8 柱状图分析,我们在处理项目问题的时候将工作分为了四部分,分别为:在对应的点处放置自适应族;放好后

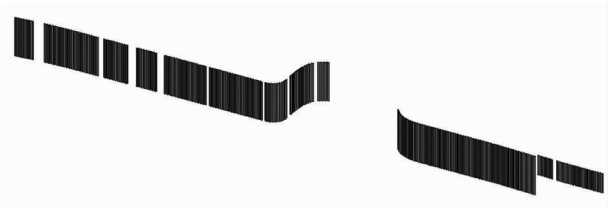


图 6 Sheetpile 成果图

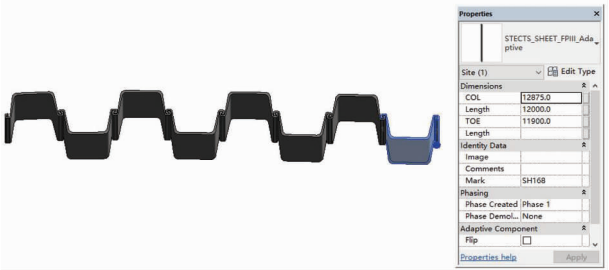


图 7 Sheetpile 信息参数

将偶数位的族翻转过来;在每个族中输入 Mark、Length、Top level 等信息;添加标注。四项工作分别耗时约 1 工时、0.5 工时、2.5 工时和 2.5 工时,而用 Dynamo 放置自适应族,只需要耗时约 50 分钟写好程序,然后用几分钟的时间运行程序。根据柱状图的最后一列可以明显看出,传统方式处理问题所耗费的时间约为用 Dynamo 参数化建模所耗费的时间 6.5 倍,而当所需要放置的模型量越大,这种数据结果对比程度更加明显,也就是说用 Dynamo 参数化建模的效率更加显著。

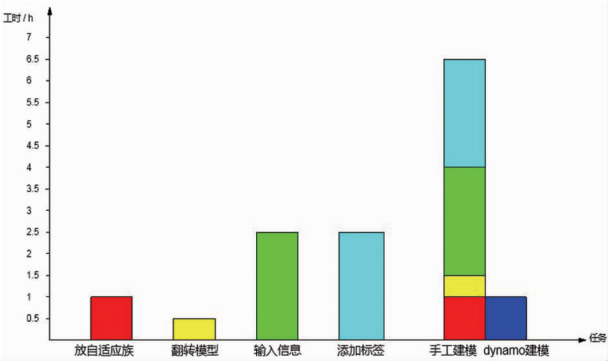


图 8 工作效率分析对比图

5 利用 Dynamo 快速放置 Revit 族并赋值属性信息

在实际工程中会遇到很多类似问题,比如在做桩模型的时候也可以用这个思路来解决问题。如图 9 所示,项目需要创建结构钻孔桩模型,其中桩的

类型有两种,每种类型有三种管径,总共有两千多个桩。如果手动放置大概需要消耗一个工作人员一天的工时,然后按照要求,两个桩之间不能有剪切现象,则再需一个工作人员一天的工时去取消剪切。所以我们借用上一个问题的思路,并根据这个的实际项目难点,来解决放桩问题。

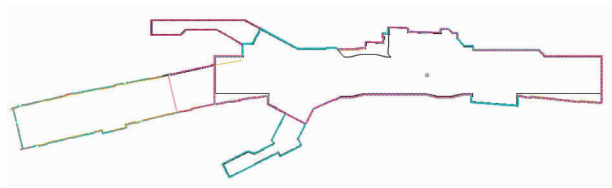


图 9 放置路径

从图 9 可以看出,此任务和上一个任务的区别在于,钢板桩是沿线放置的,但是此项目的桩是按照桩的中心点放置的。解决问题步骤如下:

- (1)用 CAD 处理图纸,筛选出来需要的信息,导入到 Revit 中。
- (2)用 Dynamo 提取导入到 Revit 中的 CAD 数据,选出需要的图层,读取需要的红、蓝两图层的信息,如图 10 所示。

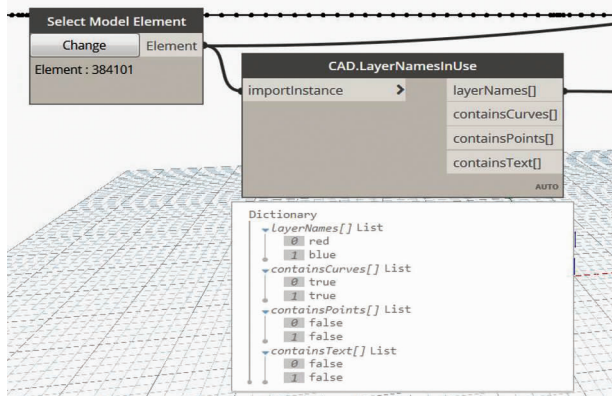


图 10 获取图层信息

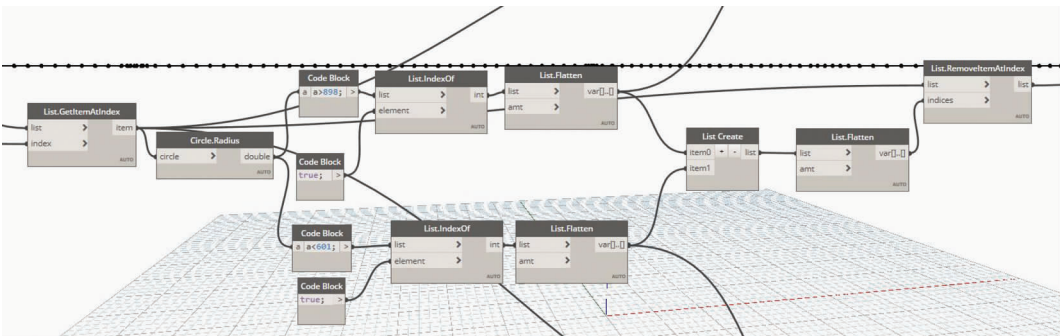


图 11 处理图层信息

(3)处理图层信息:因为桩径分为三种:600, 750, 900,所以每个图层都按照桩径划分为三类,如图 11 所示。

(4)桩是圆柱形桩,创建的桩族中心点是桩的圆心,所以放置桩的位置是 CAD 图中圆的圆心,即需提取圆中心点的数据。

(5)选择每个桩径对应的族,结合圆心点位置,用 Dynamo 生成参数化模型。

(6)Revit 生成的构件默认自动剪切,所以调用不连接节点,使放置好自动剪切的桩分开,如图 12 所示。

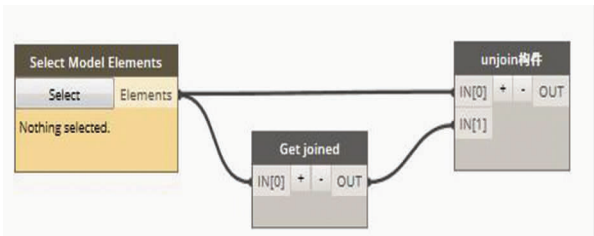


图 12 Getjoined 节点和 unjoin 节点

Dynamo 自带节点中没有让两个构件不连接的节点,但是 Revit API 中有 GetJoinedElements 找出连接构件命令和 UnjoinGeometry 使构件不连接命令,要想实现这两个命令我们可以在 Dynamo 中用代码调用 Revit 中的 API 写成节点^[5],让其可以判断读取到的数据,并让数据之间不连接^[4]。

首先将两个类型的三种尺寸规格的桩通过 Dynamo 放置到准确位置,然后每个类型的桩分别排列成一个 List,然后通过 Get joined 节点获取与放置的第一个类型的桩相互连接的桩,并生成列表。最后,通过 Unjoin 节点将放置的第一个类型的桩与通过 Get joined 节点获取的与之相连接的桩,取消相互连接。

此两个节点的部分关键代码如下：

```
doc = DocumentManager. Instance. CurrentDBDocument
items1 = UnwrapElement( IN[0] )
def getjoin( doc, item1 ):
    try:
        out = JoinGeometryUtils. GetJoinedElements
        ( doc, item1 )
        return out
    except: return False
TransactionManager. Instance. EnsureInTransaction
( doc )
if isinstance( IN[0], list ):
    a = [ getjoin( doc, x ) for x in items1 ]
else:
    a = getjoin( doc, items1 )
TransactionManager. Instance. TransactionTaskDone( )
JE = [ ]
if isinstance( a, list ):
    for je in a:
        #for aa in je:
            JE. append( Document. GetElement( doc, aa ).
ToDSType( True ) for aa in je )
else:
    JE. append( Document. GetElement( doc, aa ).
ToDSType( True ) for aa in a )
#LJE = [ JE ]
OUT = JE
doc = DocumentManager. Instance. CurrentDBDocument
items1 = UnwrapElement( IN[0] )
items2 = UnwrapElement( IN[1] )
def unjoin( doc, item1, item2 ):
    try:
        JoinGeometryUtils. UnjoinGeometry ( doc, item1, item2 )
        return True
    except: return False
TransactionManager. Instance. EnsureInTransaction
( doc )
if isinstance( IN[0], list ):
    if isinstance( IN[1], list ):
```

```
for x,y in zip( items1, items2 ):
    OUT = [ unjoin( doc, x, a ) for a in y ]
else: OUT = [ unjoin( doc, x, items2 ) for x in items1 ]
else:
    if isinstance( IN[1], list ): OUT = [ unjoin( doc, items1, x ) for x in items2 ]
    else: OUT = unjoin( doc, items1, items2 )
TransactionManager. Instance. TransactionTaskDone( )
生成模型结果如图 13 所示。
```



图 13 生成桩结果图

与建钢板桩模型相比,本节参数化建立结构钻孔桩模型都是用 Dynamo 节点库写好程序,然后参数化生成模型,但是也有区别,本节在使用现有的节点已经不能快速完成模型,于是调用 Revit 的 API 写了新的适合本项目的节点解决了问题。两种方法都能生成模型,但是对待不同的项目应该具体问题具体分析,选择出最适合本项目的方法,快速高效地解决问题。

6 结论与展望

用 Dynamo 参数化建模可以将复杂的问题简单化,沿用此种方法,可以解决很多建模问题^[6],当一个简单而又重复性的工作摆在你面前,可以使用参数化的方式解决,不仅高效而且准确^[8]。参数化建模具有以下优点:

(1) 提高建模速度。用参数化建模方法可以批量建模,相比于传统建模方法大大提高了建模效率。

(2) 确保模型精度。用参数化建模的方法可以避免人工失误,提高模型的准确率以及模型精度。

(3) 全过程可视化。Dynamo 是可视化的插件,整个参数化的过程都是可以在 Dynamo 或者是 Revit 中可以看到,如果有问题可以及时发现及时修正。

(4) 应用范围广。参数化模型具有属性信息可

以直接放在平台上管理,也可以直接进行出图。

在钢板桩的参数化建模中使用现有的节点可以解决问题,但在结构钻孔桩参数化建模的时候,使用自行编写的 Python 节点才能解决建模需求^[12]。上述表明,虽然 Dynamo 现有的节点能够解决大量的问题,但是节点是有限的,且不具有普遍适用性,这就跟 Revit 的族库一样,我们可以将自己写的节点扩充到节点库,让对解决实际工作进程和效益有着很大的帮助。

参考文献

[1] 闫智, 韩春华, 卢玉韬. 基于 BIM 快速生成高精度三维模型的方法研究及其应用探索[J]. 土木建筑工程信息技术, 2018, 10(4): 33-40.

[2] 许良梅, 刘梦, 李伟, 等. 基于 Revit 平台的 BIM 全过程正向设计[J]. 安徽建筑, 2019, 26(11): 79-80 + 102.

[3] 代振坤. 基于 BIM-Dynamo 的参数化预制构件探索[J]. 智能城市, 2018, 4(16): 30-31.

[4] 王维轩, 周东明. 基于 Revit 平台二次开发模式的研究[J]. 土木建筑工程信息技术, 2019, 11(1): 132-136.

[5] 王松. 可视化编程语言下的计算式设计插件—Dyna-

mo 初探[J]. 福建建筑, 2015, 9(11): 105-110.

[6] 代振坤. 基于 BIM-Dynamo 的参数化预制构件探索[J]. 智城建设, 2018, 16(14): 30-31.

[7] 韩继宗. 基于 Dynamo 的参数化设计研究[J]. 居业, 2019, (12): 36-38.

[8] 李庶安, 王喆, 孔晨光, 等. Dynamo 在桥梁缓和曲线段建模应用研究[J]. 公路交通科技(应用技术版), 2019, 15(4): 165-167.

[9] 幸金权, 文钦, 樊美娜. 基于 BIM 用 dynamo 实现物业管理系统可视化初探[J]. 智能城市, 2017, 3(4): 93.

[10] 林金华, 林武, 吴福居. 可视化编程在 BIM 参数化建模中的应用技术[J]. 工程建设与设计, 2018(22): 276-278.

[11] 欧阳业伟, 黄钟涛. Dynamo 在模板算量中的应用研究[J]. 山西建筑, 2019, 45(10): 256-258.

[12] 李文浩. 基于 Dynamo 的轨道交通工程计算式设计[C]. 中国岩石力学与工程学会、中国水利水电勘测设计协会、云南省岩土力学与工程学会. 第二届全国岩土工程 BIM 技术研讨会论文集. 中国岩石力学与工程学会、中国水利水电勘测设计协会、云南省岩土力学与工程学会: 云南省岩土力学与工程学会, 2017, 302-309.

Application of Dynamo for Revit in Modeling of Foundation Pit Retaining Structure

Wu Fan, Yan Zhi, Kong Xiangping

(Shanghai Urban-Construction Information Technology Co., Shanghai 200120, China)

Abstract: Based on the idea of parametric modeling in Dynamo for Revit, this paper realizes the fast and efficient placement of steel sheet piles and structural bored piles, and assigns the parameter information of each pile. Compared with the traditional manual modeling method, the parametric modeling method is able to achieve rapid modeling while ensuring the accuracy of the model, which greatly improves the work efficiency and ensures the work quality. This method can be extended to similar applications.

Key Words: BIM; Dynamo; Parametric Modeling