

# Digital Twin Models: BIM Meets NetLogo

Milan KOŘÍNEK\*, Ondřej TÁZLAR and Kamila ŠTEKEROVÁ

<sup>1</sup> University of Hradec Králové, Hradec Králové, Czech Republic; milan.korinek@uhk.cz; Ondrej.tazlar@uhk.cz; kamila.stekerova@uhk.cz

\* Corresponding author: milan.korinek@uhk.cz

**Abstract:** Digital twin is a computerized model of a physical device or system. Digital twins open new opportunities for operative decision making. Agent-Based Models (ABM) and simulations are core part of digital twins, especially in case of modelling humans in the spaces they act in. Microscopic simulations at the individual level have to be integrated with realistic models of environments. Here digital twins benefit from Building Information Models (BIM) but importing of BIM files into ABM platforms is not common. This problem is addressed in our paper. Principles of BIM are presented briefly. We suggest enhancing NetLogo modelling platform with BIM as the first step towards the development of digital twins models within NetLogo. The first part of the paper describes BIM formats and 3D visualization, including extraction of 3D model parts from IFC format. Parsing an IFC file using python is proposed, with recalculations of coordinates. It is also shown how the content of IFC can be reused in ABM and how simulations can affect BIM backwards.

**Keywords:** agent-based model; building information models; digital twin; IFC; NetLogo; simulation

**JEL Classification:** C63; C88

## 1. Introduction

A digital twin (DT) is a computerized model of a physical device or system that represents all functional features and links with the working elements (Chen, 2017). Three main counterparts of DT are a physical artefact, a digital counterpart, and data that connects them. The principles of DT modelling were initially applied in the aerospace sector and product manufacturing, recent applications are into smart cities, system engineering, robotics, healthcare, and medicine (Barricelli et al., 2019; Boje et al., 2020).

According Yokogawa (2019), DT technology can benefit multiple levels of the organization including enterprise insight (such as part of an enterprise-wide balanced scorecard), capability assurance with real-time monitoring and control of work processes, value chain optimization or integration of automation and control. Briefly, DTs open new opportunities for operative decision making.

Technologically, DTs build on artificial intelligence, Big Data processing techniques, cloud computing, high dimensional data coding, sensor technologies, and Internet of Things (Barricelli et al., 2019). Agent-Based Models (ABM) are core part of DTs, especially in case of modelling humans in the spaces they act in. Microscopic simulation at the individual level has to be integrated with a realistic model of the environment. This approach was successfully

applied in urban digital twins, a novel method in urban planning, see e.g. (Dembski et al., 2020) for a prototype of an urban digital twin for the town of Herrenberg in Germany. Virtual Singapore project (2021) offers four main capabilities to stakeholders (i.e. government, citizens, businesses and researchers). These capabilities are virtual experimentation, test-bedding, decision-making and research and development. A Digital Twin Helsinki (2021) was developed as a testing tool open to the public, also for mitigating climate change and improving energy efficiency, nowadays it also becomes leader in virtual tourism.

Building Information Modelling (BIM) is a set of technologies, processes and policies enabling multiple stakeholders to collaboratively design, construct and operate a facility in virtual space (BIM Dictionary, 2021). BIM is generally understood as an overarching term to describe a variety of activities in object-oriented Computer Aided Design, which supports the representation of building elements in terms of their 3D geometric and functional attributes and relationships (Ghaffarianhoseini, 2017). Recent trend is a representation of 4th and 5th dimension within BIM by adding temporal perspective and cost estimation (Boje et al. 2020).

Thanks to accurate information about the structure of buildings, it is desirable to directly use BIM for various simulations (Teo, 2016) which can result in process optimization. The question is how to integrate BIM within Agent-Based Models (ABM) such as models of visitors' flow in public spaces or evacuation models where parameters of buildings would be extracted from BIM. For synthesis of ABM platforms and resources, see e.g. (Abar, 2017).

The aim of the article is to propose procedures by which it would be possible to apply BIM information in ABM. The first part describes BIM formats and 3D visualization. Then parsing an IFC file using python is proposed, with recalculations of coordinates. It is also shown how the content of IFC can be reused in ABM and how simulations can affect BIM backwards.

## **2. Methodology**

In the field of infrastructure planning, different modelling approaches are applicable such as Geographic Information System (GIS), Building information models (BIM) or City Information models (CIM). Although they can be used separately, it is possible to combine them together in planning: GIS is used for analyze and work with geographic data and solve spatial problems. GIS coordinates can be used for building location. BIM is a system that primarily focuses on life cycle of a building. BIM also includes 3D models of building and its equipment. Information on multiple buildings can be managed using CIM which is focused on a city infrastructure. CIM includes tools for description and analysis of roads, streets and public spaces (Tah, 2017).

### *2.1. Building Information Model (BIM)*

Building information model is not only a 3D model of the building, but this format also includes information needed for the creation of new buildings and their management, including information on the entire life cycle of the building (Bryde, 2013). The ISO 19650 standardization has made it possible to use and share BIM across sectors.

A part of the information model is also a visual representation of the building, which can be stored in various proprietary formats (RVT, NVD, DWG) and non-proprietary formats (IFC,

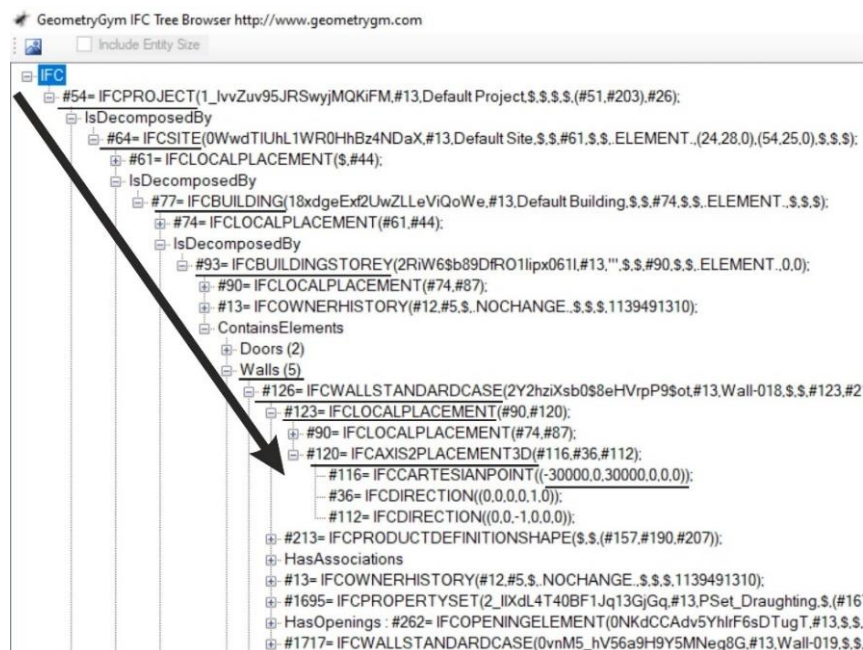
XML types, COBie). Autodesk's proprietary formats are *RVT* and *NVD* and can only be opened with Autodesk's software, *DWG* format can be opened in any CAD-based software. Revit DXF files also can be used in AnyLogic to develop structures and building where agents operate.

## 2.2 Industry Foundation Classes (IFC) Format

*IFC* is standardized (ISO 167391), open format, which can be used to transfer information between applications. However, it is still only an intermediary between applications, and some information may be lost when exporting BIM to *IFC* (Solihin, 2015). *COBie* is a human readable subset of the *IFC* format.

*IFC* file can be opened in any text browser. The file stores a lot of detailed information, which makes it widely applicable. The whole 3D model of the building is arranged hierarchically using links. Usually, only local information and a reference to superior object are attached to each element. It is possible to list hierarchy using tools such as *IFCTreeViewer*. The 3D model can be viewed in *FreeCad*, *usBIM.viewer* or *Autodesk Revit*.

The *IFC* file contains information in a hierarchical arrangement (see Figure 1). The GIS location of the object is presented on the first level, subsequent levels are used for location of the building, its floors, components of each floor (such as walls, windows, doors, or columns). These components are also broken into smaller parts. Each individual object contains a reference to the parent element and its parameters, including location and rotation. Here comes the first problem when reading *IFC* file: information on location and rotation is local only, related to the direct predecessor, which can also be described by local information. For this reason, it is necessary to search the entire hierarchy to extract object information from the lowest levels.



**Figure 1.** Example of IFC hierarchical structure from the top element to the start position of one of the definitions of walls. (Source: authors)

As mentioned earlier, not all ABM platforms can work with the *IFC* format. Moreover, it is not always necessary to transform the whole BIM model: only relevant parts can be

processed with respect to the aim of ABM, game engine or other engine (e.g. floor plans of the buildings for the purpose of simplified pedestrian movement models). It is possible to perform a one-way conversion of IFC file into a 2D or 3D model for ABM. This process may be sufficient, but if some of information and links from the IFC file is lost, it will not be possible, for example, to retroactively modify IFC data based on the simulation outputs.

### 3. IFC Parser

The reason for the need to create an IFC format parser was to create ABM working with BIM, it means to import environmental information which contains a description of real buildings. 3D model data was stored in IFC format. The obtained data were used for simulations in the *NetLogo* tool, but thanks to the export to CSV and the definition of a simple and readable structure, other tools can also benefit from this procedure.

For the needs of ABM, it is practical to extract information on individual rooms and corridors from IFC file. Walls, windows, and doors are basic building elements which are included in IFC file. The problem is the relative relationships between the objects. For our experimental purpose, the floor of the building is considered the highest level of IFC file.

Because the IFC format is a text format, it is possible to create a parser from scratch by reading individual lines. However, it was easier to use one of the already existing libraries, such as *IFCOpenShell* for the python programming language. Thanks to the references between objects, it is possible to proceed in both directions. The library allows to get a list of superior objects, as well as a list of subordinate objects. In this way, it is possible to navigate hierarchical dependencies between objects. The following section describes the most important parameters needed to develop a simple environment for ABM. The aim is to create the space of two rooms with windows and doors. Rooms are composed of walls. All walls can be extracted using a function *by\_type('IfcWall')* from *IFCOpenShell* library.

#### 3.1. Position of Object

The first parameter that needs to be read is the position. This is stored in the *IfcLocalPlacement* structure. In this structure, there is a reference to the position of the parent element, which would have to be considered, and to the local transformation of the current element. The transformation includes the local position and directions of the X and Z axes (the Y axis is calculated). The position is represented by Cartesian coordinates in the *IfcCartesianPoint* structure. These are the coordinates of the center of the wall at its beginning. The directions of the X and Z axes determine in which direction the wall is rotated. A parameter named *Axis* specifies the Z axis, and a parameter named *RefDirection* specifies the direction of the X axis. Both parameters are of type *IfcDirection*. For example, *RefDirection: IFCDirection(-1.0,0.0,0.0)* says that the wall faces left when viewed from above (provided it is not affected by the parent hierarchy). For better precision is good to calculate transition matrix and then use dot product to get global coordinates. The wall size, endpoint, or wall shape is not the part of the *IfcLocalPlacement* structure and these values needs to be read from the *IfcProductDefinitionShape* structure, which specifies the shape of the wall.

### 3.2. Shape of Object

*IfcProductDefinitionShape* provides all the important information needed to build the shape of an object. An important parameter is the representation with the *SweptSolid* parameter. Part of this representation is information about the extraction of the object in a certain direction, the length of the extraction and also the shape by which the extraction is intended. The inclusion of the *IfcRectangleProfileDef* parameter says that the shape is of the rectangle type, and its *XDim* and *YDim* parameters determine the width and length of the rectangles. It is also possible to obtain the coordinates of the center of this extracted object. The appearance of the wall does not have to be defined by a geometric shape. It can arise, for example, from a curve that is specified by several points of type *IfcCartesianPoint*. It depends on the shape and method of creating the model.

### 3.3. Other Parameters

Many parameters are a part of the IFC forma, including the definition of units of measure within *IfcUnitAssignment* structure. These parameters must be considered, because for individual objects the values are only as a decimal number without units.

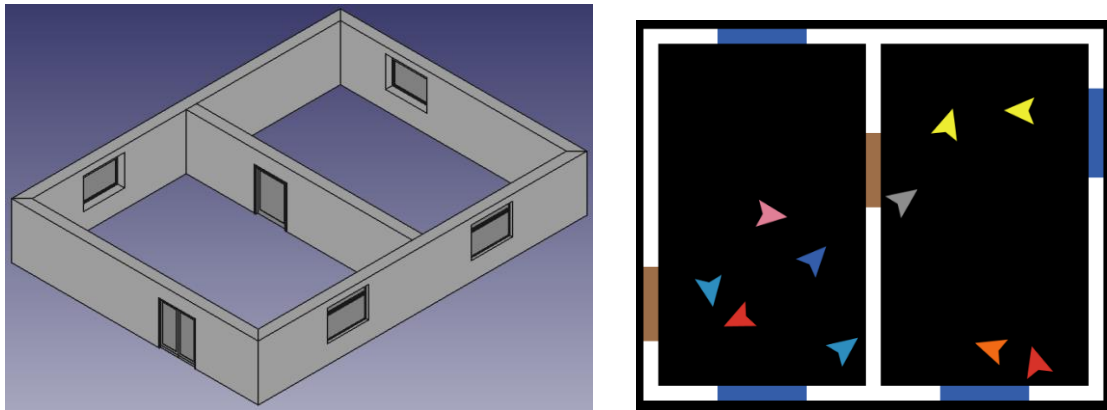
Objects contain *GUID* – parameter important for their identification. It is a unique identifier of the type *IfcGloballyUniqueID*, which is unique for the whole model. With this identifier, it is possible to uniquely identify an element and it is also possible to search for it in the hierarchy.

### 3.4. From IFC to CSV

If all the information about the objects is known, it can be exported to any format and structure that is needed. In the case of the *NetLogo* tool, it is advisable to use the csv format for input. Therefore, a simple structure was designed for a specific case, containing the GUID, object name, start position, end position, width, and reference to the parent object. Depending on the situation, it would be possible to extend the structure by, for example, the Z coordinate. In this way, information about the height of the rooms, the location of the windows or, for example, the height of the door could be stored. Thanks to this, it would be possible to create simple 2.5D or 3D simulations dealing with, for example, the volume of gases at different heights depending on the parameters of the space.

## 4. Results and Discussion

Part of the 3D model of BIM was converted to *NetLogo* using IFC file (Figure 2). With information about objects, it was possible to create corresponding 2D representation of two rooms with doors and windows. The final CSV file can be imported to other simulations platforms too. The main disadvantage of the method is that it is not a full 1:1 conversion of IFC file to CSV file. Missing information has to be added by python script. The complete logic of the IFC format is not considered, nor is it the way 3D models should be created.



**Figure 2.** Example of 3D visualization from IFC file (left) and its transformation to NetLogo 2D visualization with agents (right). Source: authors

## 5. Conclusions

Authors of review (Fuller et al., 2020) noticed that the number of papers on Digital Twins in manufacturing is significantly higher compared to papers on Digital Twins for smart cities and they identify it as the research gap. We agree and to contribute to this area, we propose to enhance NetLogo with BIM as the first step towards the development of DT models within NetLogo. We explained how it is possible convert IFC files into the specification of the environment for ABM. It was described how to get building object information and how to manage parameters that are not explicitly defined in IFC file. The creation of environment in NetLogo from IFC files was presented.

**Acknowledgments:** The financial support of the Specific Research Project Information and knowledge management and cognitive science in tourism 5. Prepared on the basis of consultations with the researchers of the BEST project ([www.bestsim.eu](http://www.bestsim.eu)).

## References

- Abar, S., Theodoropoulos, G. K., Lemarinier, P., & O'Hare, G. M. P. (2017). Agent Based Modelling and Simulation tools: A review of the state-of-art software. *Computer Science Review*, 24, 13–33. <https://doi.org/10.1016/j.cosrev.2017.03.001>
- Barricelli, B. R., Casiraghi, E., & Fogli, D. (2019). A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications. *IEEE Access*, 7, 167653-167671. <https://doi.org/10.1109/ACCESS.2019.2953499>
- BIM Dictionary. (2021). *Building Information Modelling*. <https://bimdictionary.com/en/building-information-modelling>
- Boje, C., Guerriero, A., Kubicki, S., & Rezgui, Y. (2020). Towards a semantic Construction Digital Twin: Directions for future research. *Automation in Construction*, 114, 103179. <https://doi.org/10.1016/j.autcon.2020.103179>
- Bryde, D., Broquetas, M., & Volm, J. M. (2013). The project benefits of Building Information Modelling (BIM). *International Journal of Project Management*, 31(7), 971-980. <https://doi.org/10.1016/j.ijproman.2012.12.001>
- Chen, Y. (2017). Integrated and Intelligent Manufacturing: Perspectives and Enablers. *Engineering*, 3, 588–595. <https://doi.org/10.1016/J.ENG.2017.04.009>
- Dembski, F., Wössner, U., Letzgus, M., Ruddat, M., & Yamu, C. (2020). Urban Digital Twins for Smart Cities and Citizens: The Case Study of Herrenberg, Germany. *Sustainability*, 12(6), 2307. <https://doi.org/10.3390/su12062307>
- Fuller, A., Fan, Z., Day, C., & Barlow, C. (2020). Digital Twin: Enabling Technologies, Challenges and Open Research. *IEEE Access*, 8, 108952-108971. <https://doi.org/10.1109/ACCESS.2020.299>

- Ghaffarianhoseini, A., Tookey, J., Ghaffarianhoseini, A., Naismith, N., Azhar, S., Efimova, O., & Raahemifar, K. (2017). Building Information Modelling (BIM) uptake: Clear benefits, understanding its implementation, risks and challenges. *Renewable and Sustainable Energy Reviews*, 75, 1046-1053. <https://doi.org/10.1016/j.rser.2016.11.083>
- Solihin, W., Eastman, C., & Lee, Y. C. (2015). Toward robust and quantifiable automated IFC quality validation. *Advanced Engineering Informatics*, 29(3), 739-756. <https://doi.org/10.1016/j.aei.2015.07.006>
- Tah, J. H. M., Oti, A. H., & Abanda, F. H. (2017). A state-of-the-art review of built environment information modelling (BeIM). *Organization, Technology and Management in Construction: an International Journal*, 9(1), 1638-1654. <https://www.doi.org/10.1515/otmcj-2016-0030>
- Teo, T. A., & Cho, K. H. (2016). BIM-oriented indoor network model for indoor and outdoor combined route planning. *Advanced Engineering Informatics*, 30(3), 268-282. <https://www.doi.org/10.1016/j.aei.2016.04.007>
- Virtual Helsinki. (2021). <https://www.virtualhelsinki.fi/>
- Virtual Singapore. (2021). <https://www.nrf.gov.sg/programmes/virtual-singapore>
- Yokogawa. (2019). *Digital Twin: The Key to Effective Decision-Making How a Digital Twin Delivers Superior and Sustainable Business Value*. White paper. <https://web-material3.yokogawa.com/>