BSP Project Description: BSP-3 Building Information Modelling from a Robot Map

Sunday 29th October, 2023 - 22:46

Kevin Moreira da Silva
University of Luxembourg
Email: kevin.moreira.001@student.uni.lu

This report has been produced under the supervision of:
Jose Luis SANCHEZ LOPEZ
University of Luxembourg
Email: joseluis.sanchezlopez@uni.lu
Special thanks to Mr. Muhammad SHAHEER

1. Main required competencies

1.1. Scientific main required competencies

This project explores the definitions, working principles, fundamentals, dynamics, and the theory of construction of S-graphs.

Having some knowledge on S-graphs is helpful prior to the project as the topic will be explored in its entirety and will be the primary focus of the scientific research.
To achieve the task of exploring S-graphs in depth, certain tools and environments are necessary.
BIM models is the most fundamental aspect of creating S-graphs, in the methodology followed by me and the team behind this project. An understanding of BIM models, their uses and basic functionalities is important.
To complement S-graphs, knowing about the working fundamentals of robots and sensors is recommended, it gives insight on why certain strategies are adopted when forming the S-graphs.

1.2. Technical main required competencies

In my previous BSP, I used digital construction plans in a BIM format to extract specific data using Dynamo for Revit. In this BSP, which is a continuation of my previous BSP, the reverse premise will be done. Standardized input data will be utilized to alter an existing BIM or create a new BIM building plan.

The process will be automated and generalized, such that the tool can generate a BIM from any input data so long as it follows certain restrictions or alter an already existing BIM.
To do this task, certain tools are used and consequently basic working knowledge of the uses of these tools is necessary.

The most fundamental tool is Revit, which accommodates BIM models. However, most of the manipulation will be done in Dynamo for Revit, it is a tool for Revit, which allows for manipulation of BIM models.

Dynamo is Python compatible, so Python scripts can be run inside dynamo scripts to do more specific tasks. Finally BimVision, a visualising tool for BIM models, is helpful. It allows the user and client to see the changes in real time that the dynamo script actuates on the existing BIM.

2. A Scientific Deliverable 1

In my last BSP-2, my scientific question was as follows:" How can the information available in BIM models be used by robots for their autonomous navigation?".
I will be continuing to explore this question, focusing more on the information available and its usage on robots, S-graphs.

Previously, in the Scientific part, I worked with BIM (Building Information Modeling) and explored what data was relevant from BIMs, methods, and techniques of extracting said data and briefly introduced S-graphs. In this BSP-3, on the Scientific part, I want to further explore this scientific question. The scientific research explores BIMs with an emphasis on S-graphs, what they are, how are they generated, their purpose, etc.

2.1. Context

Information available in BIM models is vast, in this project the aim is to explore S-graphs.

S-graphs contain topological and metric-semantic information about a building [1]. This information can be extrapolated from BIM models, and, from direct and odometrical robot data [1].
Me, personally, will not be working on the robot, however, scientific background on the working principles of how robot data is used will be explored. So, Robotics, and Sensor Input/Output and Computer Vision will be explored.
To complement the robot data, BIM models, which contain geometrical, semantic, and topological data from a building are explored [1]. This means exploring BIM as a digital data format. Furthermore, data within BIM must be handled. This permits us to access specific data, alter data if needed, and overall, allow available data useful. So, Data Handling and Data Processing is a key component.
Finally, to utilize all available data, Python and Dynamo for Revit are used, so Programming is necessary. To conceptualize S-graphs, Computer Vision and Graphs are necessary.

There are a lot of topics to cover, but the goal is not to explore them all deeply, but to cover some topics, just enough, to be able to explore S-graph in depth. By the end of the scientific part, certain sub-questions should be answered.

2.2. Functionalities

Some sub-questions, necessary to understand S-graphs are proposed: How to construct an S-graph? why make an S-graph? What does an S-graph provide to the robot? What information in an S-graph can the robot use? How can the robot use that information? Etc.
These are not set in stone; However, they serve as a guideline for providing a path for exploring and understanding S-graphs.

To answer the S-graph questions, some other questions arise, such as: why use BIM as a format?
This is quite a relevant question, because, answering this question by focusing on the proper aspects, gives an understanding of the reasoning behind certain technical decisions in making S-graphs.
Other questions relating to robots and sensor I/O will be addressed as well. These will be explored in much less detail, however.
Finally, after all the necessary information is introduced, S-graphs will be explored.

S-graphs are the research topic, so they will be explored in depth. S-graphs will be introduced, definitions will be explained, what components make an S-graph, how to obtain these S-graphs components to make an S-graph, explain and define all these components, how to conglomerate the necessary components to make up an S-graph as well as any relevant information related to S-graphs within the scope of the scientific question.

2.3. Ideas

S-graphs are quite complex data. They comprise topological and metric-semantic data. These terms can be quite vague, so I will first give some meaning to these terms by defining them in the context of S-graphs and buildings.
Further on, I will explore which data is available from the accessible sources, BIM models and Robot Outputs.
As such, I will first try to explain what data is available, resulting in exploring geometric, semantic, and topological data from BIMs. Also exploring odometry and how sensors work in a basic sense in from robots. How to obtain this data from BIM models and from Robot Outputs is also relevant.
As such, BIM as a digital format and basics robotics and robot sensor IO will be explored.
Additionally, what data from these available data sources is relevant for the purposes of S-graphs.

Finally, everything S-graph within a delimited scope will be explored. If possible, a simpler and basic S-graph will be constructed as well.

3. A Technical Deliverable 1

The technical deliverable of this project is a tool, named “BIM blueprint manager “, upon which a standardized set of data is transformed into a BIM construction plan. This tool can be used to either create an entirely new BIM model, or, to alter a pre-existing BIM model with the desired properties mentioned in the input of the user.

3.1. Context

BIM blueprint manager or BIMbm, is a tool that, as the name implies, manages blueprints. These blueprints refer to, but not exclusively, construction plans.
These construction plans are digitalized into a BIM format. BIM is widely used, it is a digitalized blueprints format, and in the context of this project, BIM construction blueprints are handled.

Various tools and technologies are compatible with BIM models. Revit, is the tool of choice to handle BIM for this project. Revit renders BIM models, allows for manipulation, visualization, and many others. Revit is a broad software, and it has many sub tools which are compatible with it, one of them being Dynamo for Revit.
Dynamo for Revit, or Dynamo for short, is a plugin for Revit and a library for dynamo Nodes [3].
Unlike Revit, only allows for in place modification of BIM

models[1], Dynamo allows for the creation of scripts that can modify a BIM upon being executed.
Other worth mentioning tools, that, although not fundamental, do help in the completion of this task. Python 2 and Python 3 are compatible with Dynamo and will be used whenever some functionality is required that is not already provided by Dynamo's default dynamo Nodes.
Python 3 will be for certain more complex data handling and processing tasks, and Python 2 might be used to access certain older tools which were developed in Python 2 such as the library IronPython which was used in my previous BSP-2, in Dynamo, also in the context of S-graphs.
Lastly, BIM vision, a visualization tool for BIM models is to be used throughout the making of BIMbm. Although Revit offers some basic visualization features of BIMs, BIM vision is a tool specifically designed for that and has more functionalities, so, it is preferred over Revit. It allows the user and developer to see the changes made by BIMbm in real time clearly.

3.2. Functionalities

The goal of BIMbm is to allow the user to alter or create BIMs.

Dynamo is the main tool used for coding BIMbm because of the utility of the scripts that can be run to achieve a specific desired task.

3.2.1. Standardized Input. The script must receive a standardized input that contains the data to be modified.
A good example of standardized input data that is compatible with the desired final version of BIMbm is my BSP2 tool.
BSP2's technical tool extracts structural data from a construction's BIM and outputs it into an Excel file.
This data is combined in one single Excel file (.xlxs). The data is structured and organized into Excel sheets. Each Excel sheet refers to a single component type within the structure, e.g. walls. In each sheet, columns refer to different properties of the components e.g. Length. As can be seen in the attachment below.

	A	B	C	D	E	F	G	H	I
1	VectorX	VectorY	VectorZ	ElementID	Length	Area	Width	Unconnec	Volume
2	-1351.566,	81.157,	0.000,	1441841	1354	37.97196	39.70757	302	16.23674
3	-1351.568,	81.157,	0.000,	1441842	1354.002	37.63363	39.52554	302	16.16234
4	-251.547,	15.105,	0.000,	1441843	252.0004	7.650012	53.96563	302	4.107006
5	-41.418,	-689.758,	0.000,	1441846	691.0002	12.89265	23.44055	302	4.891623
6	-95.423,	-1589.138,	0.000,	1441847	1592	29.76346	24.04531	302	11.5606
7	-93.625,	-1559.192,	0.000,	1441848	1562	30.52786	25.20308	302	11.8889
8	-14.116,	-235.077,	0.000,	1445057	235.5	4.6843	24.79078	302	1.763145
9	-37.682,	2.263,	0.000,	1445058	37.75	2.2801	58.33444	302	0.665042
10	-37.682,	2.263,	0.000,	1445059	37.75	2.2381	76.21854	302	0.868929
11	-38.676,	-644.090,	0.000,	1445060	645.2498	12.66743	25.32764	302	4.935482
12	14.116,	235.077,	0.000,	1476402	235.5	4.426546	21.74877	302	1.546794
13	-41.418,	-689.758,	0.000,	1476459	691.0002	12.28064	22.79839	302	4.757616
14	455.430,	-27.347,	0.000,	2347356	456.25	11.8845	9.919024	302	1.366717
15	-379.067,	22.762,	0.000,	2347799	379.75	8.2998	8.322633	302	0.954477
16	13.936,	232.082,	0.000,	2373268	232.4998	4.680945	15.99982	302	1.123427
17	-256.039,	15.374,	0.000,	2376175	256.5	5.02845	7.465135	302	0.578272
18	30.224,	503.343,	0.000,	2388008	504.2498	11.492	8.678418	302	1.321579
19	340.352,	-20.437,	0.000,	2477464	340.965	6.696974	16.58449	302	1.707728
20	-354.931,	21.313,	0.000,	2477467	355.5707	7.103014	16.86747	302	1.811269
21	12.797,	213.116,	0.000,	2477468	213.4998	5.6017	22.15417	302	1.428433
22	-633.609,	38.046,	0.000,	2477844	634.75	17.097	22.74314	302	4.359735
23	-203.384,	12.213,	0.000,	2479044	203.7505	3.786865	16.9429	304	1.049446
24	205.880,	-12.362,	0.000,	2493782	206.2506	5.921417	24.24173	302	1.509961
25	-1.918,	-31.942,	0.000,	2493924	32	0.9664	25.5	302	0.246432
26	250.549,	-15.045,	0.000,	2573915	251	5.01415	7.607019	302	0.576627
27	-241.066,	14.475,	0.000,	2586684	241.5	6.5987	23.07143	302	1.682669
28	1328.607,	-79.779,	0.000,	2657732	1331	36.3624	28.94793	302	11.63597
29	-354.932,	21.313,	0.000,	2662918	355.5716	7.103041	16.86749	302	1.811276
30	2.203,	36.684,	0.000,	2662925	36.75	1.4949	34.34694	302	0.3812
31	-253.044,	15.195,	0.000,	2662928	253.5	6.4079	21.34376	302	1.634015
32	-633.609,	38.046,	0.000,	2662992	634.75	18.0426	22.74869	302	4.360799
33	358.355,	-21.518,	0.000,	2663078	359.0005	7.281763	16.63384	302	1.80341
34	30.224,	503.343,	0.000,	2668701	504.2498	13.4135	10.12948	302	1.542552
35	40.833,	680.025,	0.000,	2671315	681.2498	14.2965	7.991238	302	1.644097
36	455.430,	-27.347,	0.000,	2673098	456.25	12.2042	10.18585	302	1.403483
37	-359.602,	21.593,	0.000,	2673099	360.2495	9.841186	10.40243	302	1.131736
38	-13.037,	-217.109,	0.000,	2673104	217.5002	4.924555	17.99334	302	1.181893
39	544.490,	-32.695,	0.000,	2673112	545.4704	13.20043	9.215262	302	1.518049
40	-13.037,	-217.109,	0.000,	2673113	217.5002	6.221205	10.89195	302	0.715439
41	-17.050,	-283.949,	0.000,	2673119	284.4603	5.832952	7.80832	302	0.67079
42	-389.548,	23.391,	0.000,	2677885	390.25	10.9173	10.65279	302	1.25549
43	-409.013,	24.560,	0.000,	2679314	409.75	9.30715	8.649453	302	1.070322
44	455.430,	-27.347,	0.000,	2734751	456.25	11.8845	9.919024	302	1.366718
45	-379.065,	22.762,	0.000,	2734752	379.7475	8.461731	8.485065	302	0.973099
46	-13.936,	-232.081,	0.000,	2734755	232.4994	4.819533	16.47356	302	1.156688
47	-256.036,	15.374,	0.000,	2734757	256.4975	6.810931	10.11147	302	0.783257
48	-29.745,	-495.357,	0.000,	2734760	496.2494	9.625633	7.386185	302	1.106948
49	250.548,	-15.045,	0.000,	2734763	250.9996	4.875539	7.396743	302	0.560687
50	-29.745,	-495.357,	0.000,	2734771	496.2494	13.19108	10.12212	302	1.516974
51	-40.354,	-672.039,	0.000,	2734777	673.2498	14.0741	7.960404	302	1.618521
52	455.430,	-27.347,	0.000,	2735641	456.25	11.8845	9.919024	302	1.366718
53	-379.065,	22.762,	0.000,	2735642	379.7475	8.299731	8.322618	302	0.954469
54	13.933,	232.042,	0.000,	2735646	232.4602	5.376347	18.3799	302	1.290323
55	-256.036,	15.374,	0.000,	2735648	256.4975	5.028381	7.465105	302	0.578264

Wall | Door | window | Room | Stairs | Sheet1 | +

3.2.2. BSPbm's Requirements. The BSPbm tool would alter values in BIM models according to the user's instructions.
The values to be changed will be input in an interface.
The user, when changing values, will not need to keep in mind the interrelationships of walls. This means that, if the user changes the data pertaining to an arbitrary Wall A into Wall A'; Consequently, Wall B which is originally connected to Wall A would then automatically have its starting point in the new endpoint of Wall A'. Resulting in a conflict-free and more convenient way to change data.

The tool will house similar features for other BIM components: walls, rooms, windows, and doors.
It will then generate a new BIM model with the altered data. The tool can alter or create entirely new models.

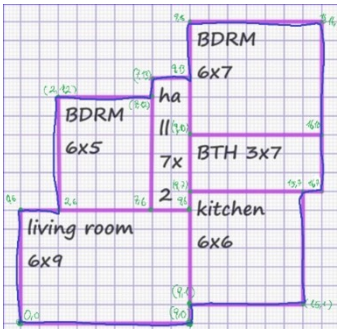
3.3. Ideas

The first functionality, and the only one to be implemented until now[2], is to create BIM files from available data.

Suppose the goal is to create a simple 2D planar view of a building, with a fence around it, such as in the picture below.

1. I believe Revit only allows for in place modifications but could not fact check this.

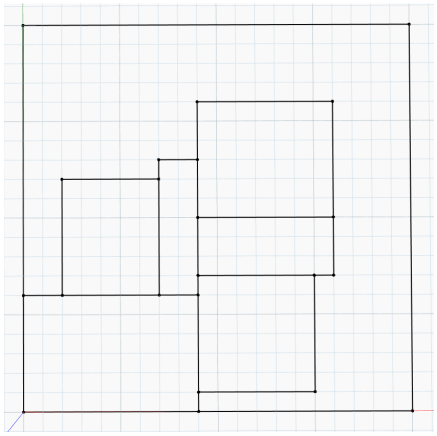
2. 29/10/2023



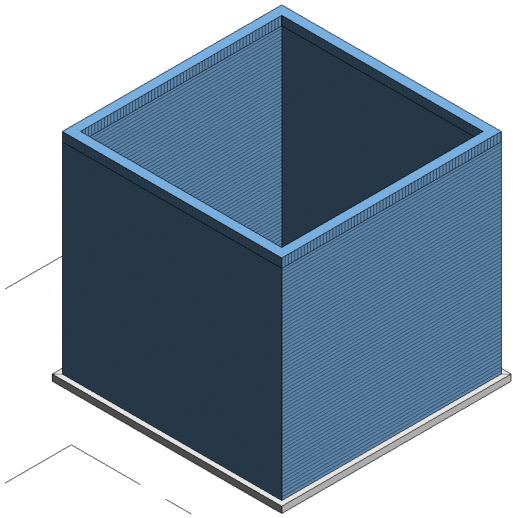
The premise is to input the information relevant to the creation of the building in some storage that can handle the data. For the earlier builds of BIMbm, the storage of input data is Excel .xlxs, later on, it most likely will be .CSV format.

	A	B
1	0	0
2	9	0
3	9	1
4	15	1
5	15	7
6	16	7
7	16	10
8	16	16
9	9	16
10	9	13
11	7	13
12	7	12
13	2	12
14	2	6
15	0	6
16	7	6
17	9	6
18	9	7
19	9	10
20	20	0
21	0	20
22	20	20

The data represents points with x and y coordinates in the A and B columns respectively.
An earlier version of the tool BIMbm creates a simple BIM from an Excel sheet filled with data.
This is the output from the Excel sheet of data.

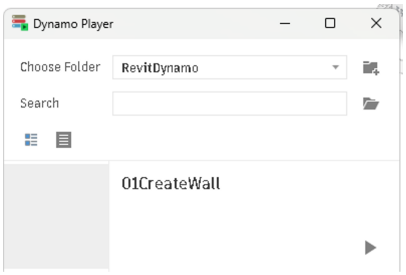


This method can be applied to bigger and more complex BIMs.
Furthermore, this concept can be expanded upon. For instance, we can give these connected dots, wall properties.



The final product is supposed to allow the end user to change any existing BIM file. Walls can be changed without disrupting the file, meaning that the integrity of the building is preserved, and subsidiary changes are made to accommodate for a modified wall. Doors and windows are also planned to be modifiable.

A group of walls can be used to create a room, and existing rooms will be retained as a room when the walls are modified. All these functionalities are expected to be in a plugin dropdown list, whereby the user can click a plugin and it will run certain functionalities. Here is an abstract example of this concept. The final goal of the technical deliverable is



to have, either a drop-down list of ready-to-use plugins for various useful purposes or, perhaps make an application with a GUI, but, functionality is prioritized so the main focus in making usable scripts first.

References

[1] Robot Localization, information manipulation and extraction of plans and autonomous robot S-graph navigation
Robot Localization using Situational Graphs and Building Architectural Plans
Muhammad Shaheer, Hriday Bavle, Jose Luis Sanchez-Lopez, Holger Voos 2022 Sept 23 arXiv preprint arXiv:2209.11575 https://doi.org/10.48550/arXiv.2209.11575

[2] S-Graphs+,
Advanced Situational Graphs for Robot Navigation in Structured Indoor Environments Hriday Bavle, Jose Luis Sanchez-Lopez, Muhammad Shaheer, Javier Civera, Holger Voos 2022 Nov 16 arXiv preprint arXiv:2211.08754 https://doi.org/10.48550/arXiv.2211.08754

[3] DynamoRevit https://github.com/DynamoDS/DynamoRevit

4. Plagiarism statement

I declare that I am aware of the following facts:

- I understand that in the following statement the term "person" represents a human or **ANY AUTOMATIC GENERATION SYTEM**.
- As a student at the University of Luxembourg I must respect the rules of intellectual honesty, in particular not to resort to plagiarism, fraud or any other method that is illegal or contrary to scientific integrity.
- My report will be checked for plagiarism and if the plagiarism check is positive, an internal procedure will be started by my tutor. I am advised to request a pre-check by my tutor to avoid any issue.
- As declared in the assessment procedure of the University of Luxembourg, plagiarism is committed whenever the source of information used in an assignment, research report, paper or otherwise published/circulated piece of work is not properly acknowledged. In other words, plagiarism is the passing off as one's own the words, ideas or work of another person, without attribution to the author. The omission of such proper acknowledgement amounts to claiming authorship for the work of another person. Plagiarism is committed regardless of the language of the original work used. Plagiarism can be deliberate or accidental. Instances of plagiarism include, but are not limited to:
  1) Not putting quotation marks around a quote from another person's work
  2) Pretending to paraphrase while in fact quoting
  3) Citing incorrectly or incompletely
  4) Failing to cite the source of a quoted or paraphrased work
  5) Copying/reproducing sections of another person's work without acknowledging the source
  6) Paraphrasing another person's work without acknowledging the source
  7) Having another person write/author a work for oneself and submitting/publishing it (with permission, with or without compensation) in one's own name ('ghost-writing')
  8) Using another person's unpublished work without attribution and permission ('stealing')
  9) Presenting a piece of work as one's own that contains a high proportion of quoted/copied or paraphrased text (images, graphs, etc.), even if adequately referenced

  Auto- or self-plagiarism, that is the reproduction of (portions of a) text previously written by the author without citing that text, i.e. passing previously authored text as new, may be regarded as fraud if deemed sufficiently severe.