Proceedings of the IASS 2024 Symposium
Redefining the Art of Structural Design
August 26-30, 2024, Zurich Switzerland
Philippe Block, Giulia Boller, Catherine DeWolf,
Jacqueline Pauli, Walter Kaufmann (eds.)

Towards Automated Building Life Cycle Assessments: A Novel Approach Using Large Language Models and the COMPAS Framework

Li CHEN*, Heidi SILVENNOINEN*[a], Catherine DE WOLF[a], Daniel HALL[b], Tom VAN MELE*, Philippe BLOCK*

* Block Research Group, Institute of technology in Architecture, ETH Zurich
Stefano-Franscini-Platz 1, HIB E 45, 8093 Zurich, Switzerland
li.chen@arch.ethz.ch

[a] Circular Engineering for Architecture, Institute of Construction & Infrastructure Management, ETH Zurich
[b] Department of Management in the Built Environment, TU Delft

Abstract

The Architecture Engineering and Construction (AEC) industry, a significant contributor to global greenhouse gas emissions, urgently requires innovative approaches to upscale life cycle assessment (LCA) for both new and existing projects to meet emission reduction targets. This can be done through further automation of LCA. However, traditional methods for conducting LCAs face challenges due to the manual and time-consuming process of data synthesis from heterogeneous sources and the interoperability issues among various software tools used by different stakeholders. This research introduces a novel automated LCA workflow leveraging the synergies between large language models (LLMs) and the open-source computational framework COMPAS. By combining the cognitive capabilities of LLMs for processing unstructured data with the computational precision of COMPAS for handling 3D geometries, we propose a system capable of conducting LCAs across diverse data sources with minimal manual intervention. Our methodology involves segmenting the LCA process into several modules, including material data extraction, geometry data conversion, material and geometry data integration, life cycle inventory (LCI) data association, LCA calculation, and output integration. This modular approach streamlines the LCA process, improving scalability, and also facilitates its adaptation to different project scales and complexities. A prototypical implementation using the NEST HiLo project demonstrates the feasibility and efficiency of the proposed approach. It shows particular promise for reducing the manual labour involved in integrating data that is stored in inconsistent ways. This research marks a step towards achieving fully automated LCA workflows, enhancing the capacity for comprehensive environmental assessments in both the planning stages of ongoing projects and the evaluation of existing built projects, thus promoting more sustainable construction practices across the industry.

Keywords: LCA, BIM, Interoperability, LLMs, COMPAS, Software integration, embodied emission, AEC, Automation, Autonomous Agent

1. Introduction

The building industry is responsible for ca. 12.3 Gt of greenhouse gas emissions annually, accounting for over a third of global energy and process-related CO2 equivalent emissions [1]. Embodied emissions contribute significantly to this total: according to the International Energy Association, non-operational emissions from buildings cause an estimated 2.5 Gt of CO2 equivalent emissions (20% of the building industry's total) globally in 2022 [1], while Röck et al. [2] estimated that embodied emissions typically account for 20-50% of new buildings' emissions over 50 years. Life cycle assessment (LCA) is needed for both ongoing projects, to ensure that emissions targets are met, as well as for vast amounts of existing built projects in order to create a database to establish benchmarks for embodied carbon (De Wolf and Lieve [3]). Accordingly, many LCA tools exist to automate the calculation of embodied emissions in a building design, typically by multiplying the Bill of Quantities (BoQ) for materials with the relevant environmental characteristics, which can include, e.g., embodied emissions and life cycle scenarios.

However, in practice, gathering accurate data on the quantity of materials and linking them to environmental data is difficult. This involves an elaborate data synthesis process across interdisciplinary stakeholders including architects, engineers, manufacturers and LCA experts. Interoperability issues often arise among the large variety of professional software options that each stakeholder must rely on (De Gaetani et al. [4]). The adoption of Building Information Modelling (BIM) alone does not eliminate this problem. While BIM software can serve as an effective central coordination tool, specialised applications for each individual discipline are still necessary, as no single software can perform all tasks (De Gaetani et al. [4]). In order to generate a BoQ, a dedicated person of integration (typically the BIM manager) is required to manually inspect all relevant material information across data sources and input them by hand into a BIM software in a consistent way (Carvalho et al. [5], Sampaio et al. [6]). For large and complex projects, this can be time-consuming, error-prone and often only happens towards the project end, while LCA can be much more beneficial if performed in the early project stages of pre-design, concept design and developed design (Meex et al. [7], Hollberg et al. [8]). Furthermore, the vast majority of existing built projects were not realised using BIM, so gathering accurate material data for those projects remains a big challenge.

Despite many attempts in both industry and academia to build more automated LCA pipelines based on various specific software platforms, a path towards a fully automated LCA workflow that can work universally across all types of input data sources, remains out of reach. This can be explained by the fact that the process of data synthesis cannot be tackled by hard-written computer programs alone; more importantly, it requires human-like cognitive capabilities to process and integrate information from heterogeneous data sources, which are often fuzzy and unpredictable, as well as to associate them with appropriate environmental impact data using expert knowledge.

This research aims to solve the challenge of manual data synthesis for LCA. To do so, we propose to combine large language models (LLMs) with COMPAS (Van Mele et al. [9]), a Python-written, open-source computational framework, which can act as a computational solver and bridge to a vast amount of software applications used in the Architecture Engineering and Construction (AEC) industry. LLMs are revolutionary in their potential capacity to perform cognitive tasks at a level comparable to human experts (OpenAI et al. [10], Katz et al. [11], Luo et al. [12], Singhal et al. [13]), and linking them to COMPAS compensates for their shortcomings in accurate 3D-geometric computations. The goal is to create a novel software pipeline that enables fully automated LCA, which can be used universally with a diverse range of data sources.

2. Related works

A number of reviews have investigated approaches to automate LCA based on BIM or 3D models (Guignone et al. [14], Potrč Obrecht et al. [15], Köck et al. [16], Chen et al. [17], Zheng et al. [18], Soust-Verdaguer et al. [19]). Approaches found in recent publications can be broadly split into five categories: BIM-software plugins, visual programming workflows, standalone tools using industry foundation class (IFC) files, Extract Transform Loaded (ETL) tools and machine learning tools. However, none of them solve the problem of data synthesis across platforms in a satisfactory way.

2.1. BIM software plugins

One popular strategy for automating LCA is through plugins for BIM software, e.g.Tally [20], OneClickLCA [21], LCA Link [22] plugins for Revit. These plugins can directly consume BoQs generated from BIM software and conduct the LCA without leaving the platform. An obvious limitation is that these plugins can only be used inside their host platforms such as Revit, while some stakeholders might need to operate on alternatives like Archicad (Guignone et al. [14]). Although a product like OneClickLCA provides similar plugins for a wider range of BIM platforms, it still does not address the initial challenge of manually integrating material information from various specialised software into a centralised BIM model; as previously stated, a mono-software solution for all stakeholders is unrealistic. In addition, manual association of LCA inventory data to each of the building components is still unavoidable, which can become a time-consuming task in large projects.

2.2. Visual programming

Another approach to automating LCA is through parametric models created with visual programming tools such as Grasshopper for Rhinoceros and Dynamo for Revit (Bueno et al. [23], Hollberg and Ruth [24], Bombyx [25]). Compared to ready-built plugins, this approach offers better flexibility and potentially a higher level of automation, but at the cost of a steeper learning curve (Guignone et al. [14]). However, the issue of platform-specificity persists, as these visual programming workflows are only executable inside their host platforms, which also makes it challenging for them to directly interact with data provided by other software.

2.3. Standalone tools + IFC

A more platform-agnostic way to conduct an automated LCA is to use an open standard such as IFC [26], a prevalent data-exchange format for BIM-related applications; thus, the pipeline can work with a much larger variety of software (Alwan and Ilhan Jones [27], Ebertshäuser et al. [28], LLatas et al. [29], Forth et al. [30]). However, not all design software provide functionalities for straightforward IFC export, an example being Rhinoceros, a widely used program for architectural modelling and computational design. Even if all software used in a design workflow directly supports IFC, the exported IFC files often contain material information in inconsistent formats. As a consequence, a standalone interface often must be provided to manually re-assign material information to each element.

2.4. ETL tools

A fourth proposal for automated LCA relies on using Extract-Transform-Load (ETL) tools to create data pipelines that extract material information directly from native file formats like .rvt for Revit (Shadram et al. [31]). Although, in theory, this method can work with any data source, it may suffer scalability issues, since new data pipelines have to be manually configured per software and per data storing pattern. The high cost and un-transparent pricing terms of dedicated ETL software such as FME [32] can be another concern.

2.5. Machine-learning tools

There is also an increasing trend of conducting LCA using machine-learning (ML) models that are trained end-to-end to predict embodied emission directly from high-level input parameters such as building typology, structural types, number of floors, total areas, etc. (Fang et al. [33], Fenton et al. [34]). The challenge of this approach is that the accuracy of the ML model highly depends on the quality of training data and model architecture. With the decision-making process inside the end-to-end machine-learning models acting like a black box, there is little way to validate such an approach other than comparing the results against a large and high-quality test set of ground-truth data, which is hardly accessible (D'Amico et al. [35]).

Among existing approaches, one important yet unaddressed issue is how to effectively integrate the inconsistently stored material data from multiple information sources. While a tool based on LLMs can potentially solve this problem in an automated way at scale, research has not yet been published about a detailed implementation.

3. Overview of architecture

The goal of this research is to explore a novel workflow leading towards a fully automated LCA pipeline, being developed to work with a diverse range of data sources. More specifically, the goal is to overcome the key bottleneck of manual data synthesis by utilising the COMPAS framework in conjunction with Large Language Models. In our pipeline, software packages from the COMPAS ecosystem are used to build the underlying infrastructure and act as the "Computational Solver", while large language models such as GPT-4 are used as the "Cognitive Solver" to perform tasks that previously could only be tackled by human experts. Together, they form a powerful autonomous agent that is able to execute advanced geometric calculations and complex cognitive tasks of logical reasoning at the same time (Figure 1).

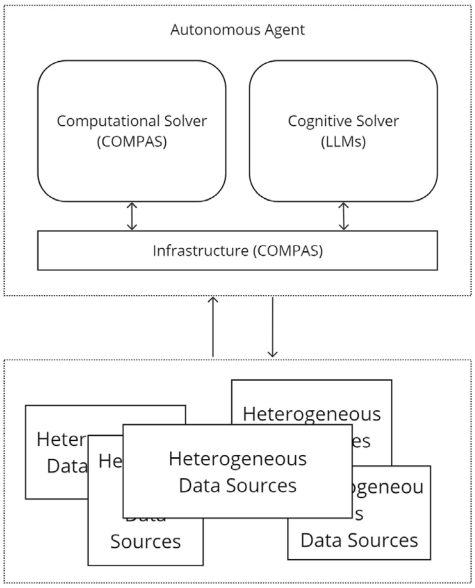

Figure 1. Concept diagram of the proposed system

Our system divides the data-synthesis process into several independent modules (Figure 2), each carrying out one of the following tasks:

1. Material Data Extraction, whereby information about the material type of each building element is extracted and catalogued from different data sources;
2. Geometry Data Conversion, which involves converting the geometric data of building elements into a uniform representation for further calculation;
3. Material and Geometry Data Integration, in which the material and geometric data are combined to create a BoQ, after checking and eliminating any duplicate elements;
4. LCI Data Association, whereby appropriate life cycle inventory items are assigned to each material type presented in the BoQ;
5. LCA Calculation, whereby the material quantities are multiplied with associated emission factors provided by life cycle inventory items; and,
6. Output Integration, an optional step to integrate the LCA results back into the input sources for future decision making.

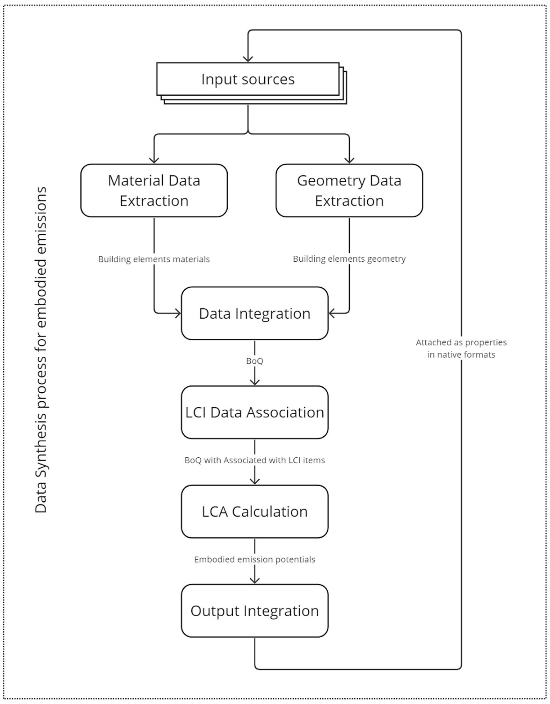

Figure 2. Modules of the data-synthesis process

Such division ensures distinct functional boundaries of each module, so that they can be improved independently in future research, therefore simplifying the complex topic of LCA automation through a strategic segmentation. Additionally, these modules can also be used for a wide range of related tasks such as cost estimation, structural analysis, and many other types of environmental impact analysis.

To test this system, we are building a prototypical implementation using NEST HiLo (Block et al. [36]). NEST HiLo is an experimental building unit on Empa and Eawag's NEST platform [37]. It is a modular building unit containing many innovative components such as a lightweight funicular floor system, an integrated doubly curved thin shell roof, and an adaptive solar façade, etc. (Block et al. [36]) (Figure 3). Despite the unit's moderate scale, its mix of conventional and experimental building components makes it a well-suited case study for exploring the challenge of data synthesis in building LCA.

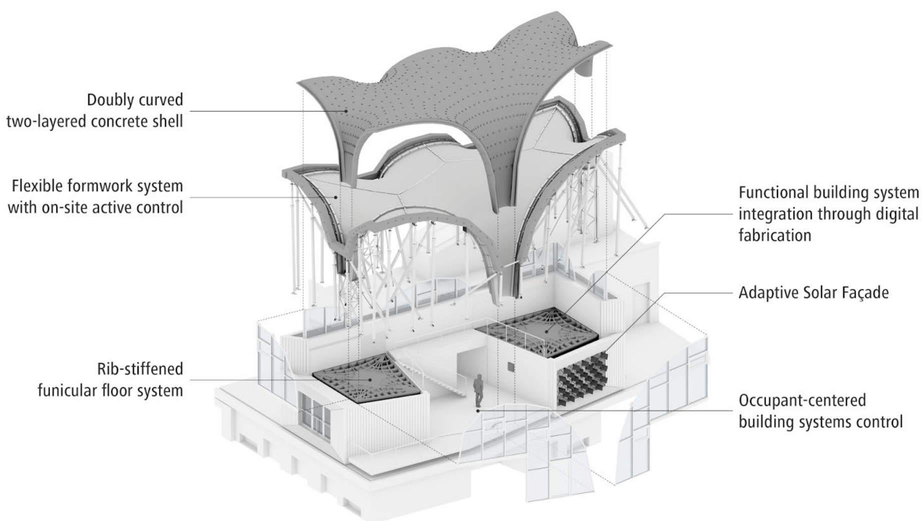

Figure 3. Innovative components of NEST HiLo

4. Implementation details

The following project files from HiLo have been collected so far for an early implementation: two BIM models (.ifc), with one being an architectural model exported from Revit, and one detailed geometry model for the façade exported from Solidworks; 44 Rhinoceros files (.3dm), containing detailed geometry for the two lightweight funicular floors together with all the temporary formworks, detailed geometry for the structure of roof, together with full building context; In total, 23,389 individual geometric elements are found in these files.

4.1. Material data extraction

Extracting material information from multiple software platforms is always challenging because few rules have been established for how they should be stored, even with a standardised data-exchange format like IFC. This reality stays the same for the HiLo files, where the material information is linked to building elements in many different ways. In several Rhinoceros files, the material name is expressed in their layer names; in some other files, the name can be a substring of the object name or sometimes vaguely mentioned in the description of a block instance. In the IFC file exported from Revit, the material information can be obtained from a custom property named "Structural Material", while in the other IFC file exported from Solidworks, the equivalent property is simply named "Material". The language used for material names can be either German or English. For a large number of elements, the material was never explicitly assigned. Nevertheless, if a human expert takes a look at the full property sheet and the shape of the component, an educated guess can often be made with relatively high confidence.

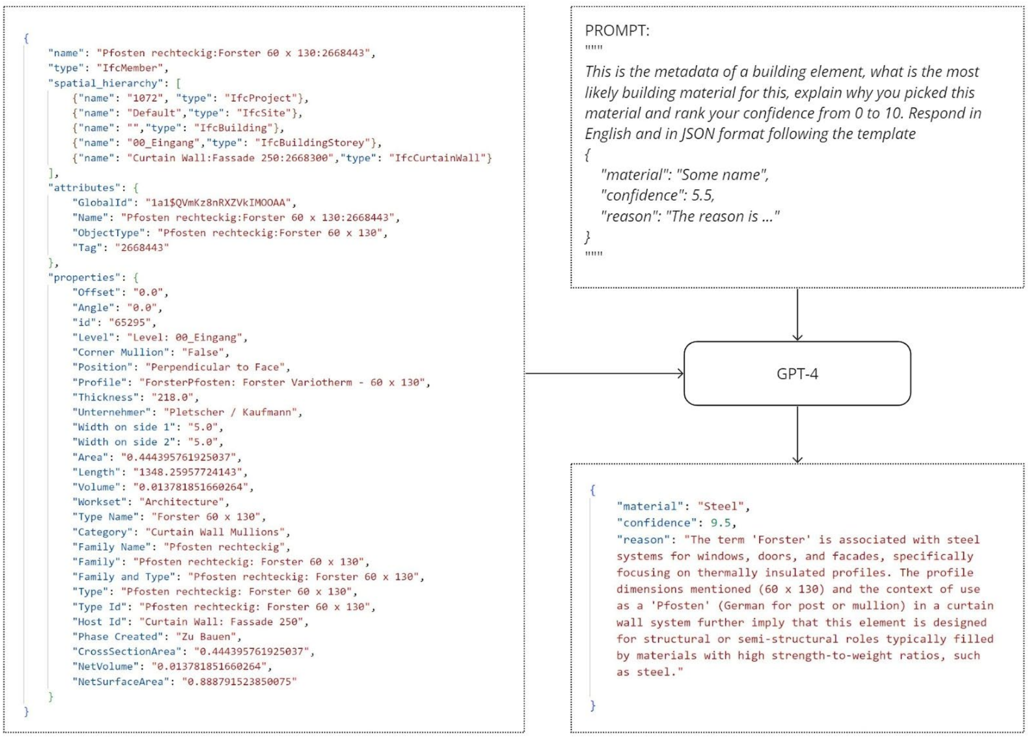

Figure 4. Example prompt extracting implicitly stored material data from a building element

Writing custom scripts to accommodate such arbitrary data storing patterns is overwhelming and unrealistic. However, LLMs can be a perfect candidate to automate such tasks with significantly less development work. As a demonstration, we tested the ability of LLMs to identify the materials of a sampled list of one thousand building elements found in the different types of files containing various metadata, including object names, geometry types, custom properties, spatial hierarchies, layer names etc. GPT4 was prompted, through the OpenAI API, to identify the most likely building material of each component based on this information. Besides the material name, we also asked it to explain itself in a short sentence, accompanied by a score of confidence, so we can inspect and audit LLMs' decision-making process in future analysis (Figure 4). We additionally requested GPT-4 to produce the outputs in structured JSON format so they can be directly used as inputs in the next part of the pipeline, without manual intervention. For the elements with low confidence scores, we repeat the prompt but this time accompanied by a visual aid of the element's 3D rendering, highlighted from its context, which in many cases makes the confidence score significantly higher (Figure 5). This step can be automated as well with the help of compas_viewer, a standalone visualiser from COMPAS. For the test sample of one thousand elements, the average response time from GPT-4 is 4.2s per request, and can be reduced to 1.5s if omitting the field "reason" which is used for research inspection purpose. The total cost for one thousand prompts is approximately 10 USD. This means the entire material extraction process of 23,389 building elements from this project can be completed by an LLM in roughly 12 hours at the cost below 250 USD, while the same workload could take a human operator several weeks at a cost of different order of magnitude.

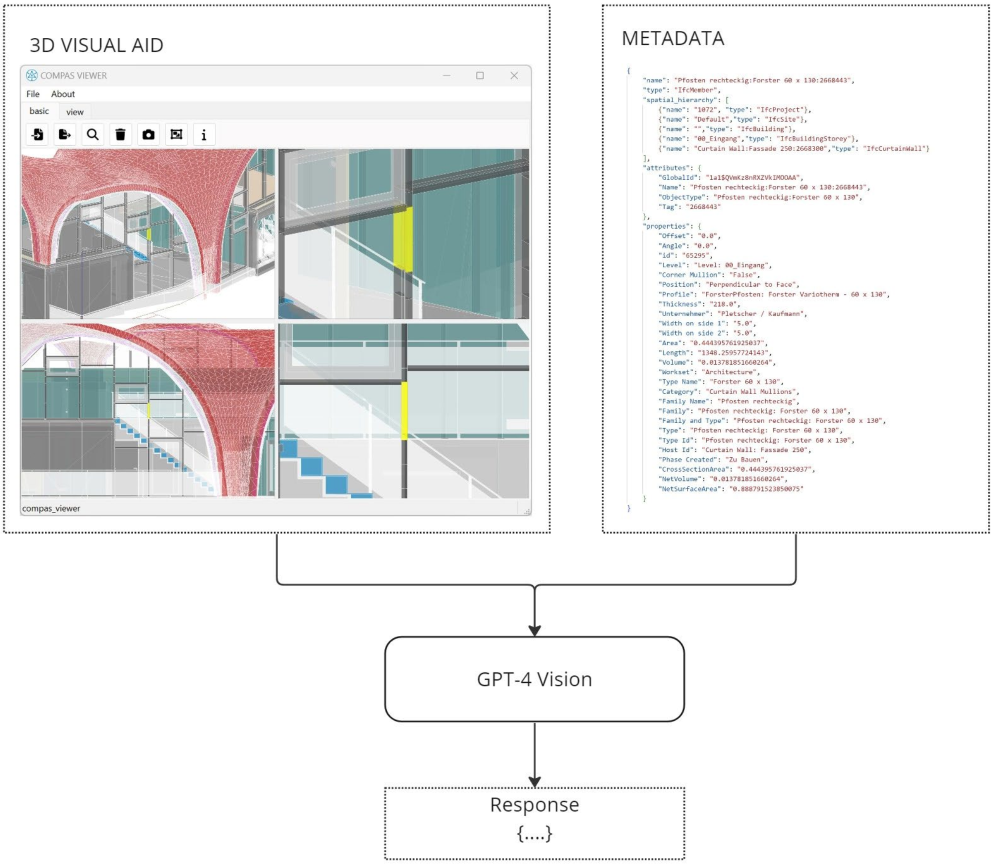

Figure 5. Second-round prompted with 3D visual aid

4.2. Geometric data conversion

The second step is to convert the geometric data of building elements from different formats to an uniform representation so that we can calculate their volumes and run geometric checks like collision detection to resolve conflicts and duplications. This part relies on several COMPAS packages: compas_ifc is used for parsing IFC files, and compas_rhino for loading geometries and traversing contents in Rhinoceros files. We then convert all geometries to Boundary Representations (BRep) using compas_occ, which provides a rich set of APIs to interact with BReps for operations like volume calculation and collision detection.

4.3. Material and geometry data integration

The next step is to take the outputs from the first two modules and use them to generate a cohesive BoQ. The main goal here is to resolve potential conflicts and duplication of building elements that are provided by multiple file sources at different levels of detail (LOD) and project stages. For example, while the full architectural model of HiLo already contains the coarse geometries of the façade elements like window panels and steel frames, the dedicated façade model offers much more detailed versions of the same elements, which should be used instead of the coarse versions (Figure 7). Another example is the reinforced concrete base for the curved roof from a Rhinoceros file, where the concrete geometry is defined as a simple solid box, excluding the steel rebar geometry buried inside (Figure 8).

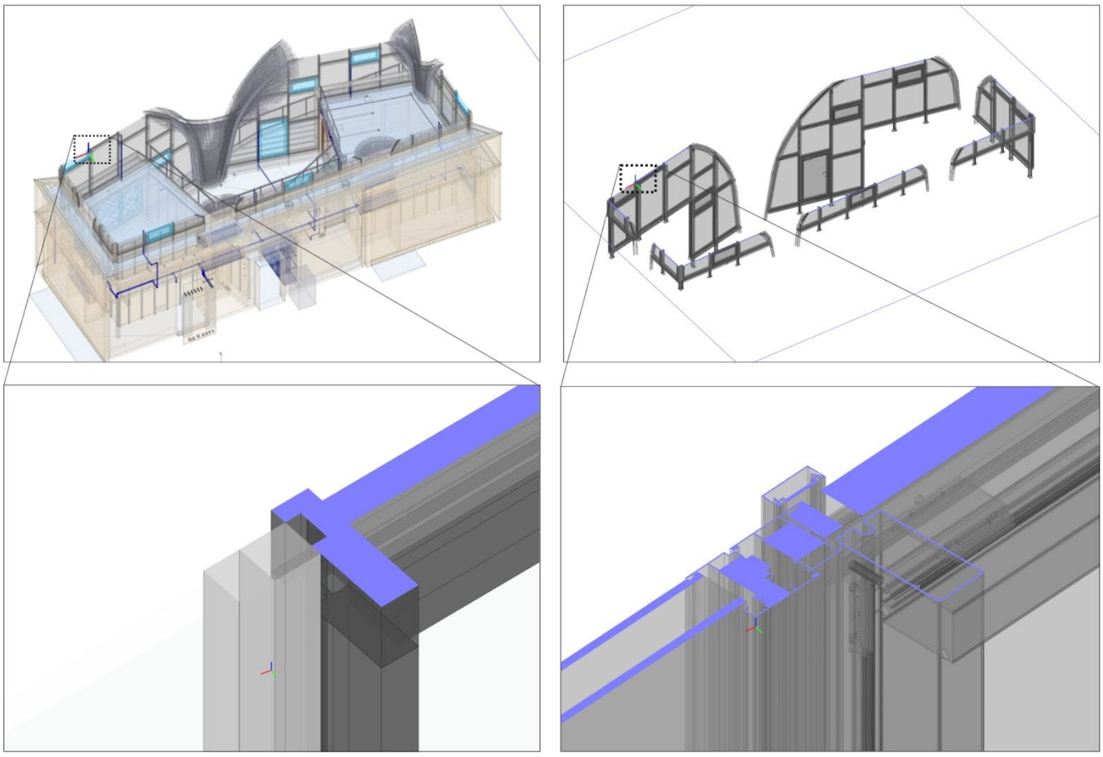

Figure 7. Same building elements with geometries of different LOD from two IFC files

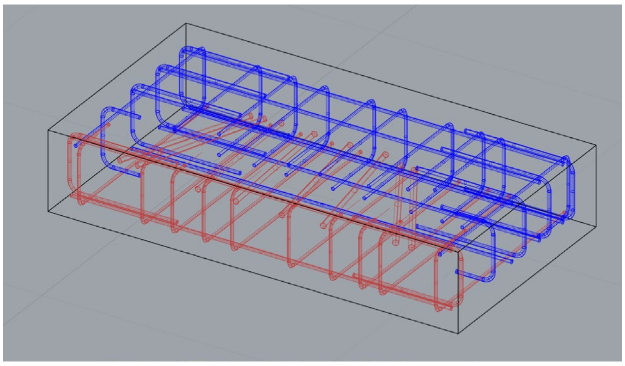

Figure 8. Example element where volume of internal geometries should subtracted

In order to resolve such conflicts, a computational decision tree is constructed, taking both material and geometry information extracted earlier as inputs. For example, if we detected two building elements colliding, we first check if the two elements are assigned the same material or name; if yes, that is likely because they both represent the same element but at a different level of detail. In such a case, we should keep the more detailed geometry (higher surface area, lower volume); if not, it is likely a situation similar to reinforced concrete. In such a case, we should accordingly subtract the volume of the inner shell from the outer ones when calculating the BoQ, effectively subtracting the volume of the rebar from the concrete. We iterate through all pairs of elements until all conflicts are resolved (Image 9). At the end of this step, a clean BoQ can be generated.

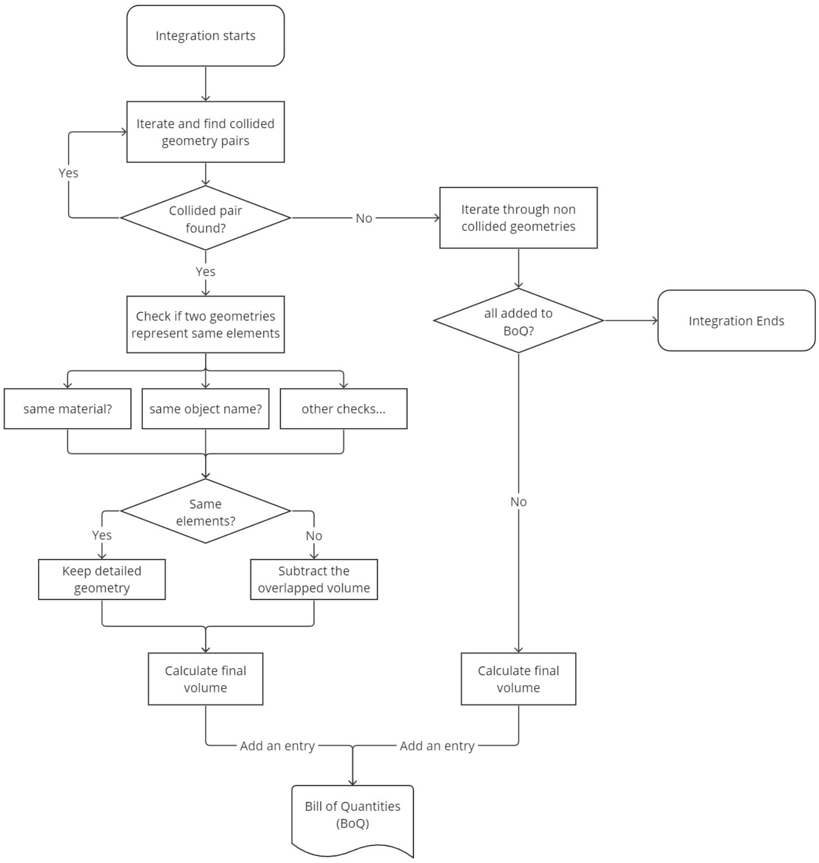

Figure 9. Flowchart for data integration

4.4. LCI data association

Once the BoQ is created, the next step is to associate each material in the BoQ with corresponding inventory items from an LCA database with appropriate functional units. This can be done again by iteratively prompting LLMs providing material to be matched with a list of detailed information about available LCI items. With GPT-4 currently supporting a maximum text window of 128k tokens [38], it is possible to feed the entire inventory data of a typical LCA database such as KBOB [39] directly into a single prompt, and get back a matched inventory item. In the case of larger databases, techniques such as vectorisation and chunking can be used (Schwaber-Cohen [40]).

4.5. LCA Calculation

The LCA calculation consists simply of multiplying the material quantities from the BoQ to emission potential factors provided by associated inventory items from the LCA database. In the case the functional unit is not volume-based, unit conversions will be needed, which is relatively easy to automate.

4.6. Output integration

A last optional step is to integrate the LCA results back into the input source files, so that they become directly accessible in stakeholders' original authoring platforms. In the case of this project, the LCA results are integrated back using the same packages that were used for reading data. Compas_rhino can write LCA data as User Data in Rhino objects, and compas_ifc can be used to write LCA results into the property set called "Pset_EnvironmentalImpactIndicators", which is officially recommended for such purposes.

5. Discussion

The main contribution of this research is to explore a path towards a fully automated LCA workflow by overcoming the challenge of manual data synthesis from heterogeneous data sources. Such a process is time-consuming, error-prone and difficult to automate with conventional tools. Our strategy has been to use a combination of COMPAS framework and LLMs, where COMPAS acts as the infrastructure and computational solver, and LLMs as the cognitive solver. Together, they form a powerful autonomous agent that demonstrates potential to automate the majority of tasks involved in the estimation of embodied emission at scale without human intervention. We use NEST HiLo as a case study to build an early implementation of this system. Compared to existing approaches, our system is more open and customisable, and involves significantly less manual work at potentially a much lower cost. By integrating the cognitive ability of LLMs to process inconsistent and unstructured inputs, such a system can provide unparalleled flexibility to work seamlessly with any information sources with arbitrary data storing patterns. This capability is not only useful for conducting LCA in the early project stages, where the material information is scattered and unintegrated but will also make it significantly easier to conduct assessments on countless built projects that were not centrally planned with tools like BIM.

Despite the significant potential of our approach, a number of limitations still need to be fully investigated and resolved. First of all, a thorough comparison will be conducted between the LLM-based solution and conventional manual procedures in terms of error rates, time efficiency, and financial costs. Secondly a larger variety of data sources can be included into the pipeline, such as native formats of other applications, text-based documents, excel sheets, and even 2D drawings. Finally, the integration and benchmark using different LLMs other than OpenAI's GPT-4, especially open-source alternatives like Llama 2 will be included, to reduce reliance on proprietary models. The authors are therefore currently working on a full implementation and validation of the proposed system.

References

[1] 'Tracking Clean Energy Progress 2023 – Analysis', IEA. Accessed: Apr. 08, 2024. [Online]. Available: https://www.iea.org/energy-system/buildings#tracking

[2] M. Röck *et al.*, 'Embodied GHG emissions of buildings – The hidden challenge for effective climate change mitigation', *Appl. Energy*, vol. 258, p. 114107, Jan. 2020, doi: 10.1016/j.apenergy.2019.114107.

[3] D. Wolf and C. (Catherine E. Lieve), 'Low carbon pathways for structural design : embodied life cycle impacts of building structures', Thesis, Massachusetts Institute of Technology, 2017. Accessed: Apr. 08, 2024. [Online]. Available: https://dspace.mit.edu/handle/1721.1/111491

[4] C. I. De Gaetani, M. Mert, and F. Migliaccio, 'Interoperability Analyses of BIM Platforms for Construction Management', *Appl. Sci.*, vol. 10, no. 13, Art. no. 13, Jan. 2020, doi: 10.3390/app10134437.

[5] J. P. Carvalho, I. Alecrim, L. Bragança, and R. Mateus, 'Integrating BIM-Based LCA and Building Sustainability Assessment', *Sustainability*, vol. 12, no. 18, Art. no. 18, Jan. 2020, doi: 10.3390/su12187468.

[6] A. Z. Sampaio, G. Azevedo, and A. Gomes, 'BIM Manager Role in the Integration and Coordination of Construction Projects', *Buildings*, vol. 13, no. 8, Art. no. 8, Aug. 2023, doi: 10.3390/buildings13082101.

[7] E. Meex, A. Hollberg, E. Knapen, L. Hildebrand, and G. Verbeeck, 'Requirements for applying LCA-based environmental impact assessment tools in the early stages of building design', *Build. Environ.*, vol. 133, pp. 228–236, Apr. 2018, doi: 10.1016/j.buildenv.2018.02.016.

[8] A. Hollberg, G. Genova, and G. Habert, 'Evaluation of BIM-based LCA results for building design', *Autom. Constr.*, vol. 109, p. 102972, Jan. 2020, doi: 10.1016/j.autcon.2019.102972.

[9] T. Van Mele, G. Casas, L. Chen, and C. Kasirer, 'COMPAS'. COMPAS Association. Accessed: Sep. 16, 2023. [Online]. Available: https://compas.dev/

[10] OpenAI *et al.*, 'GPT-4 Technical Report'. arXiv, Mar. 04, 2024. doi: 10.48550/arXiv.2303.08774.

[11] D. M. Katz, M. J. Bommarito, S. Gao, and P. Arredondo, 'GPT-4 Passes the Bar Exam'. Rochester, NY, Mar. 15, 2023. doi: 10.2139/ssrn.4389233.

[12] X. Luo *et al.*, ‘Large language models surpass human experts in predicting neuroscience results’. arXiv, Mar. 14, 2024. doi: 10.48550/arXiv.2403.03230.
[13] K. Singhal *et al.*, ‘Towards Expert-Level Medical Question Answering with Large Language Models’. arXiv, May 16, 2023. doi: 10.48550/arXiv.2305.09617.
[14] G. Guignone, J. L. Calmon, D. Vieira, and A. Bravo, ‘BIM and LCA integration methodologies: A critical analysis and proposed guidelines’, *J. Build. Eng.*, vol. 73, p. 106780, Aug. 2023, doi: 10.1016/j.jobe.2023.106780.
[15] T. Potrč Obrecht, M. Röck, E. Hoxha, and A. Passer, ‘BIM and LCA Integration: A Systematic Literature Review’, *Sustainability*, vol. 12, no. 14, Art. no. 14, Jan. 2020, doi: 10.3390/su12145534.
[16] B. Köck, A. Friedl, S. Serna Loaiza, W. Wukovits, and B. Mihalyi-Schneider, ‘Automation of Life Cycle Assessment—A Critical Review of Developments in the Field of Life Cycle Inventory Analysis’, *Sustainability*, vol. 15, no. 6, Art. no. 6, Jan. 2023, doi: 10.3390/su15065531.
[17] Z. Chen, L. Chen, X. Zhou, L. Huang, M. Sandanayake, and P.-S. Yap, ‘Recent Technological Advancements in BIM and LCA Integration for Sustainable Construction: A Review’, *Sustainability*, vol. 16, no. 3, Art. no. 3, Jan. 2024, doi: 10.3390/su16031340.
[18] B. Zheng, M. Hussain, Y. Yang, A. P. C. Chan, and H.-L. Chi, ‘Trade-offs between accuracy and efficiency in BIM-LCA integration’, *Eng. Constr. Archit. Manag.*, vol. ahead-of-print, no. ahead-of-print, Jan. 2023, doi: 10.1108/ECAM-03-2023-0270.
[19] B. Soust-Verdaguer, C. Llatas, and A. García-Martínez, ‘Critical review of bim-based LCA method to buildings’, *Energy Build.*, vol. 136, pp. 110–120, Feb. 2017, doi: 10.1016/j.enbuild.2016.12.009.
[20] ‘Tally’. Accessed: Apr. 07, 2024. [Online]. Available: https://www.choosetally.com/
[21] ‘One Click LCA | Revit | Autodesk App Store’. Accessed: Apr. 07, 2024. [Online]. Available: https://apps.autodesk.com/RVT/en/Detail/Index?id=3065869958781255107&appLang=en&os=Win64
[22] ‘LCAlink | carbon calculating | New Zealand’, LCAlink NZ. Accessed: Apr. 07, 2024. [Online]. Available: https://www.lcalink.co.nz
[23] C. Bueno, L. M. Pereira, and M. M. Fabricio, ‘Life cycle assessment and environmental-based choices at the early design stages: an application using building information modelling’, *Archit. Eng. Des. Manag.*, vol. 14, no. 5, pp. 332–346, Sep. 2018, doi: 10.1080/17452007.2018.1458593.
[24] A. Hollberg and J. Ruth, ‘LCA in architectural design—a parametric approach’, *Int. J. Life Cycle Assess.*, vol. 21, no. 7, pp. 943–960, Jul. 2016, doi: 10.1007/s11367-016-1065-1.
[25] Bombyx, ‘Bombyx-ETH/Bombyx’. Dec. 10, 2023. Accessed: Apr. 08, 2024. [Online]. Available: https://github.com/Bombyx-ETH/Bombyx
[26] ‘Industry Foundation Classes (IFC)’, buildingSMART Technical. Accessed: Sep. 16, 2023. [Online]. Available: https://technical.buildingsmart.org/standards/ifc/
[27] Z. Alwan and B. Ilhan Jones, ‘IFC-based embodied carbon benchmarking for early design analysis’, *Autom. Constr.*, vol. 142, p. 104505, Oct. 2022, doi: 10.1016/j.autcon.2022.104505.
[28] S. Ebertshäuser, K. Graf, P. Von Both, K. Rexroth, R. Di Bari, and R. Horn, ‘Sustainable building information modeling in the context of model-based integral planning’, *IOP Conf. Ser. Earth Environ. Sci.*, vol. 323, no. 1, p. 012113, Aug. 2019, doi: 10.1088/1755-1315/323/1/012113.
[29] C. LLatas, B. Soust-Verdaguer, A. Hollberg, E. Palumbo, and R. Quiñones, ‘BIM-based LCSA application in early design stages using IFC’, *Autom. Constr.*, vol. 138, p. 104259, Jun. 2022, doi: 10.1016/j.autcon.2022.104259.
[30] K. Forth, A. Braun, and A. Borrmann, ‘BIM-integrated LCA - model analysis and implementation for practice’, *IOP Conf. Ser. Earth Environ. Sci.*, vol. 323, no. 1, p. 012100, Aug. 2019, doi: 10.1088/1755-1315/323/1/012100.
[31] F. Shadram, T. D. Johansson, W. Lu, J. Schade, and T. Olofsson, ‘An integrated BIM-based framework for minimizing embodied energy during building design’, *Energy Build.*, vol. 128, pp. 592–604, Sep. 2016, doi: 10.1016/j.enbuild.2016.07.007.
[32] ‘FME-Pricing’, FME by Safe Software. Accessed: Apr. 07, 2024. [Online]. Available: https://fme.safe.com/pricing/

[33] D. Fang, S. V. Kuhn, W. Kaufmann, M. A. Kraus, and C. Mueller, 'Quantifying the influence of continuous and discrete design decisions using sensitivities', *Adv. Archit. Geom. 2023*, pp. 411–426, Oct. 2023, doi: 10.1515/9783111162683-031.
[34] S. K. Fenton, K. D. Rycke, and L. D. Laet, 'Predicting embodied carbon of building structure types through machine learning.', 2023.
[35] B. D'Amico *et al.*, 'Machine Learning for Sustainable Structures: A Call for Data', *Structures*, vol. 19, pp. 1–4, Jun. 2019, doi: 10.1016/j.istruc.2018.11.013.
[36] P. Block *et al.*, 'NEST HiLo: Investigating lightweight construction and adaptive energy systems', *J. Build. Eng.*, vol. 12, pp. 332–341, Jul. 2017, doi: 10.1016/j.jobe.2017.06.013.
[37] 'Empa - NEST - Homepage'. Accessed: Apr. 08, 2024. [Online]. Available: https://www.empa.ch/web/nest
[38] 'New models and developer products announced at DevDay'. Accessed: Apr. 08, 2024. [Online]. Available: https://openai.com/blog/new-models-and-developer-products-announced-at-devday
[39] K. der B. L. der öffentlichen B. KBOB, 'Ökobilanzdaten im Baubereich'. Accessed: Apr. 08, 2024. [Online]. Available: https://www.kbob.admin.ch/kbob/de/home/themen-leistungen/nachhaltiges-bauen/oekobilanzdaten_baubereich.html
[40] R. Schwaber-Cohen, 'Chunking Strategies for LLM Applications | Pinecone'. Accessed: Apr. 08, 2024. [Online]. Available: https://www.pinecone.io/learn/chunking-strategies/