Early-Stage Building Lifecycle Optimization of Cost & Carbon Impact

CSHub Research Brief | Volume 2021, Issue 3 | Jingyi Liu | jingyil@mit.edu

MIT CONCRETE SUSTAINABILITY HUB

Revitalizing LCA Tools

CSHub researchers have developed a software tool that recommends building solutions in the early design stage to minimize life cycle cost and carbon impacts. While most existing life cycle tools rely on detailed building models or material assignments that make it challenging to explore sustainable strategies in the earliest phases, the CSHub tool can analyze conceptual designs when only building geometry is specified. It applies life cycle thinking by considering both the embodied-stage material usage and the operational-stage energy usage. The tool recommends multiple detailed design schemes that possess lower carbon impacts (kgCO2eq) and cost (in USD) and identifies the building attributes that are important to consider in achieving these objectives.

A Two-part Workflow

To make life cycle assessment (LCA) a seamless addition to the early design stage, researchers developed a workflow (a type of program) in Grasshopper, which is a plugin

Key Takeaways:

- This workflow provides sustainable strategies during the early design stage, which can save users time when building and modifying detailed models.
- The workflow also offers greater flexibility. It can recommend users a range of optimal attribute values instead of fixed values and allows them to balance design diversity, cost, and environmental impact while modifying the workflow to meet their demands.
- When following recommended attribute features and design solutions, the workflow can help save ~10% on cost and ~20% on the carbon impact on average. For a medium-sized office building, these savings could total $6 million over 50 years.

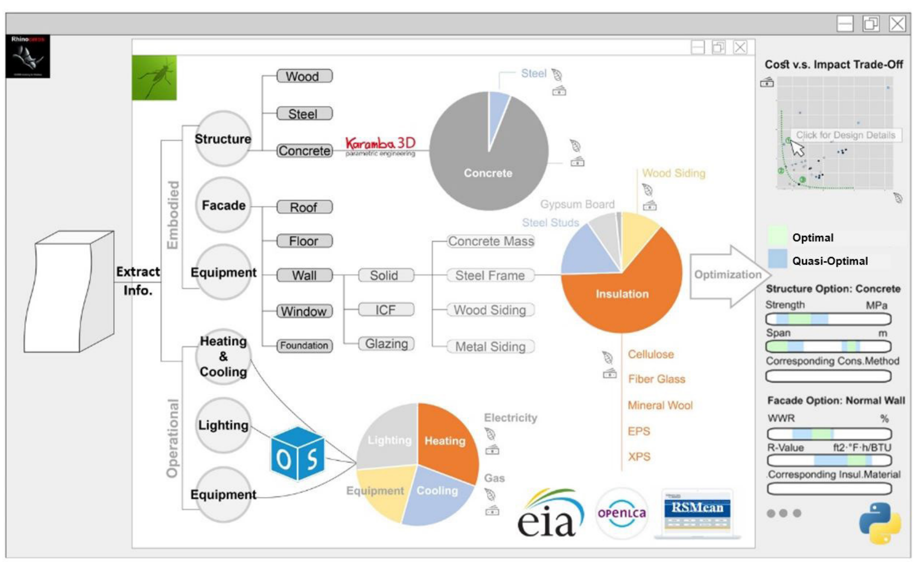

Fig. 1 An overview of the lifecycle optimization workflow in Grasshopper.

The workflow extracts the building's geometry information from Rhino and assigns scores to various design solutions based on lifecycle cost and carbon impact.

77 Massachusetts Ave, 1-374, Cambridge, MA 02139 | cshub@mit.edu | cshub.mit.edu

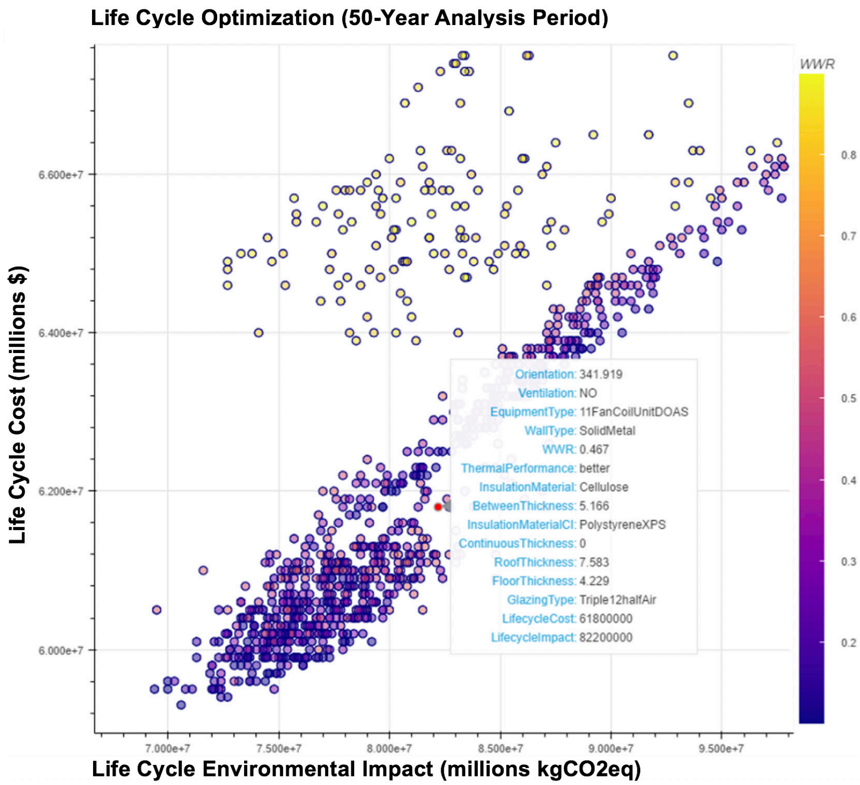

Fig. 2 A Pareto Front Diagram showing the trade-off between cost and carbon over a 50-year analysis period. Each point represents one design solution. Points colored in purple feature non-glazing facades and points colored in yellow feature glazing facades. Users can click on each point for construction details. For simplicity, the other axis (design diversity) is not shown.

in the popular design software, Rhino. The workflow contains two parts.

The first part recommends optimal ranges for numerical attributes, such as window-to-wall ratio & wall R-value, and rankings for categorical choices, such as wall type and insulation material for a specified building geometry. To produce these recommendations, the workflow generates numerous designs with attribute values randomly selected based on a range of market and building standards; quantifies the energy consumption, cost, and embodied carbon of each design solution; and then scores those solutions based on life cycle cost and greenhouse gas emissions (or carbon impacts) (**See Fig. 1**).

The second step recommends a separate set of detailed design solutions based on the optimization of life cycle cost, life cycle carbon impact, and design diversity. Using genetic optimization, the workflow iterates increasingly optimal designs based on the three objectives above: the final result is a 3D Pareto front diagram with three objectives as three axes. **Fig. 2**, which includes only two of those axes, shows a trade-off between the objectives of lifecycle cost and lifecycle carbon footprint. Though not shown in Fig 2, design diversity—the breadth of solutions considered—allows users to better meet their unique needs by increasing the range of potential solutions.

Obtaining Optimal Designs

When analyzing a conceptual geometry of a medium-size office building in Boston, **Fig. 3** shows the optimal and quasi-optimal ranges recommended for various attributes, which are the output of the workflow. Optimal means the best balance between cost and carbon impact, whereas quasi-optimal relaxes the definition of optimal to consider more designs. For example, the optimal range for window-to-wall ratio (WWR) accumulates at very small values, whereas the quasi-optimal range expands the limit. Taking wall R-value as another example, quasi-optimal ranges are either very small or large because of the trade-off between insulation cost and thermal performance. A thicker layer of insulation improves energy savings, but it is costly in terms of material usage.

The impact of different numerical attributes on building performance is shown in **Fig. 4**: attributes with taller columns have greater impacts on performance (categorical attributes are not shown). For this case, it is clear that WWR has the biggest influence on low-impact and cost building designs.

By following the workflow's recommended attribute features and design solutions, it's possible to save, on average, around 10% on cost and 20% on carbon impact in various USA locations. For a medium-sized office building in Boston, a 10% cost saving corresponds to around $6 million over 50 years. Right now, the tool is undergoing preliminary testing and we intend to release it as a Grasshopper plugin for additional testing later this year.

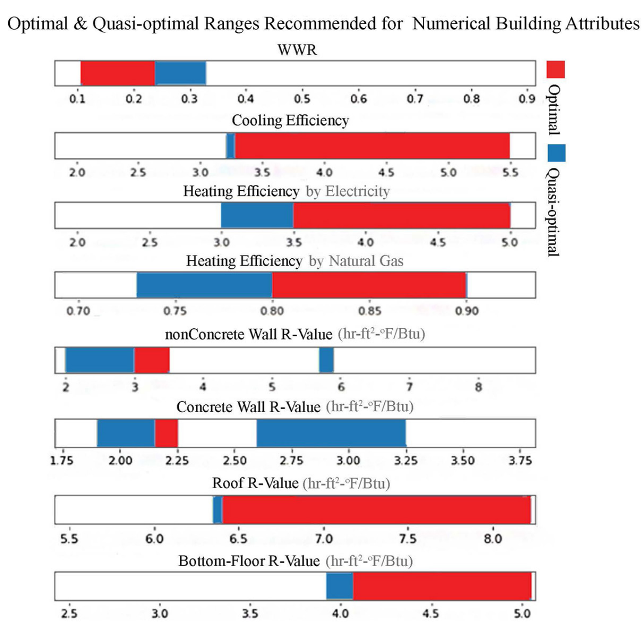

Fig. 3 Recommended ranges for numerical attributes

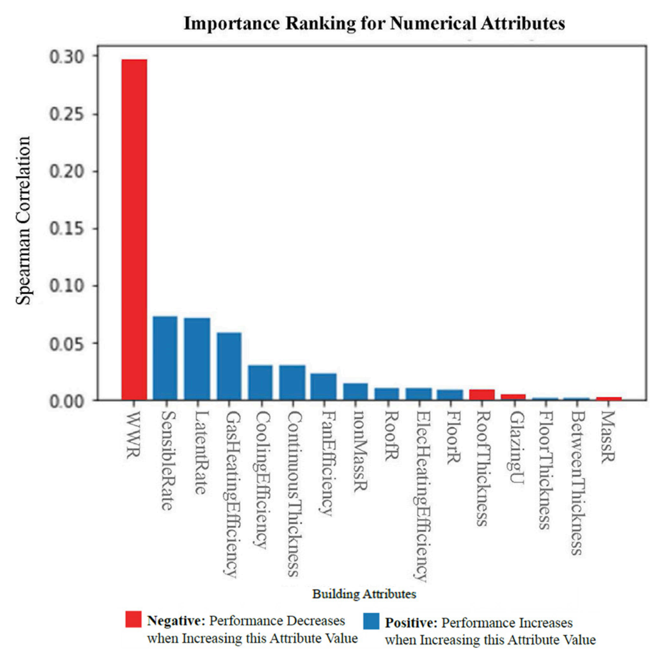

Fig. 4 Sensitivity analysis of numerical building attributes

Related Links:

- CSHub Buildings Research
- CSHub Buildings LCA Research
- CSHub Building and City Energy Consumption Research

Citation:

Liu, Jingyi. (2021). "Early-Stage Building Lifecycle Optimization of Cost & Carbon Impact." Research Brief. Volume 2021, Issue 3.

This research was carried out by CSHub with sponsorship provided by the Portland Cement Association and the Ready Mixed Concrete Research & Education Foundation. CSHub is solely responsible for content.

77 Massachusetts Ave, 1-374, Cambridge, MA 02139 | cshub@mit.edu | cshub.mit.edu