An evaluation of key factors affecting the design of efficient beam-&-slab floor systems.

Natasha K. Hirt

# 1.0 Motivation and Background (1000-1500)

The world needs efficient buildings. Between 2006 and 2019, global built floor area increased by 65%. The International Energy Agency (IEA) estimates that by 2050, it will double again. As construction accelerates to meet the needs of a larger and more demanding population, so will the environmental impact of the building sector. Engineers working to reduce the embodied carbon of the built environment can either reduce the amount of material they use or use materials with lower carbon coefficients; the product of these two values is the embodied carbon of the structure. Embodied carbon currently accounts for approximately 35% of annual global carbon emissions, and 7% results from the embodied carbon of building construction materials specifically, such as concrete, steel, and aluminum.[1] To mitigate the environmental impact of future construction, we require novel construction methods that achieve the same safety and stiffness as conventional structures with fewer emissions.

Flexural systems such as beams and slabs are prime candidates for innovation and improvement, as they rank among a building’s most material-intensive components. Materials generally perform poorly under flexure compared to pure compression or tension, and due to this intrinsic inefficiency, slabs typically constitute between 40–80% of the embodied energy of structures up to 80 stories.[2–4] Further, the mass of the floors affects the sizes of the columns and foundations required to support them, so lighter floors yield a lighter overall building.

Efficient floor slabs are not a uniquely modern idea; precedents for efficient floor design have arisen whenever construction materials were in limited supply. Notably in pre-WWII Italy, a 1939 ban on steel reinforcement (an anti-autarkic material) prompted radical explorations into new structural forms.[5] Engineer and architect Pier Luigi Nervi excelled in this environment, developing new prefabrication and design methods for concrete and wire-mesh-reinforced ferrocement. His style is characterized by structural slabs and vaults whose ribs run not orthogonally, as is traditional, but flow along the isostatic lines of the structure. The structural forms were determined using a combination of classical plate theory, strain gauge and photo-elastic experiments, and intuition, and allowed for the economical construction of large spans at a time of great scarcity.[6]

As technology and mathematical knowledge evolved, analytical methods have consistently found that better slab design has significant material savings potential. Morley’s instigating work in 1966,[7] which was further developed by Rozvany and collaborators in the following decades,[8] concerns itself with the minimum arrangement of steel reinforcement within concrete slabs so as to achieve optimal moment distributions. Bolbotowski et al. have verified Rozvany’s results computationally and generalized them to a wider range of load-dependent problems.[9] Despite the closed-form nature of the problem, its applicability to a range of boundary conditions, and the generality made possible by numerical methods, Morley’s early concern that the “necessary reinforcement arrangement is… too complex for practical use”[7] has proven true in standard construction practice.

Most recently, access to powerful computational tools has prompted renewed interest in optimal floor system design. In particular, the motivating environmental concerns have underscored the need for practical, readily implementable solutions. The two principal approaches to this challenge have been shaped beams and optimal beam layouts.[10]

In an investigation on shaped beams, Ismail et al. show that carving out underutilized material from a reinforced concrete (RC) beam can reduce its volume by up to 55%, [10] a result later replicated in timber.[11] Both of these materials can be cast or cut into the desired cross-section. Steel, which is the beam material discussed in this paper, has not been fully explored in the realm of shape optimization,[12] owing both to the efficient form of the conventional steel I-beam and to the difficulty of variably shaping hot-rolled steel.[13]

The second approach, optimal beam layouts, has several entry points, of which the most salient is continuum topology optimization across an entire slab.[14] By algorithmically iterating over a three-dimensional design domain, continuum topology optimization produces a complex pattern of concrete ribs bespoke to the loading and support conditions of that particular slab.[15] Despite considerable material savings, the manufacture of nonstandard slabs is energy, material, and time-intensive, and a full life-cycle analysis would be necessary to evaluate their full environmental impact.[16] Alternative strategies for slab layout optimization include ground-structure topology optimization. Whiteley et al. used this method, which consists of varying the stiffness of members within a dense network until only the essential members remain, to guide the design of an RC concrete slab.[17] The optimization algorithm used minimized total structural volume within the constraints of compliance with Eurocode 2, and both nearest-neighbor and fully-connected networks were found to be suitable starting points. Hearkening back to Nervi’s methods for economical design, the suggested formwork for the resulting slab would be single-sided and reusable for slabs with identical boundary conditions.

@ figure of drawn precedents

In sum, both analytical and experimental evidence suggest that, when used, optimal flexural systems can significantly reduce the embodied carbon of buildings. Thus far, however, a generalized, code-compliant design strategy that leverages current construction methods has not emerged, leading to a separation between standard engineering practice and theoretically optimal systems.

This paper presents a method for analyzing and developing new slab geometries using computational tools. These tools allow the discovery of new, potentially nonobvious, structural configurations that are efficient, aesthetically pleasing, code-compliant, and use commercially available materials and construction methods. As such, it employs four out of Fang et al.’s thirteen design strategies for reducing structural embodied carbon. (0a Estimating embodied carbon from bottom-up quantities; 1: Exploring or optimizing the parametric design space; 3: Using less material; 9. Exploiting standardization and/or customization).[18]

The section below outlines a design strategy for designing composite slabs with nonstandard steel beam layouts. These layouts can be assembled using conventional steel construction techniques, and the RC slabs modeled are also standard. A comparative analysis shows how different design decisions made by the engineer and architect can affect the efficiency of the slab, and how this tool can be used in combination with parametric design workflows to find optimal layouts.

# 2.0 Methodology (2000-3000)

Restructure

2.1 Conceptual overview  
2.2 Structural design decisions  
2.2.1 Load distribution and rebar layout

## 2.1 Conceptual overview

This method outlines a computational strategy for analyzing the loads imposed on a beam network by a concrete slab. It allows for the design, calculation, and optimization of slab geometries that typical analysis methods are insufficient for using a combination of vector-based geometric analysis and finite elements. The method is written in accordance with ACI 318-19, ASCE 7-22, and AISC-16 design codes.[19–21]

@@ Flowchart

The method follows the outline of a standard steel design workflow. Beam-and-slab systems are typically designed by arranging secondary beams, or joists, parallel to the short edge of a rectangular slab. The maximum span of the spanning system, which may be uniaxial, orthogonally biaxial, or isotropic depending on the material and reinforcement choice, is then determined according to the spacing between the joists along the load-bearing vector of the slab. From this, the required depth, and thus the dead load, of the slab is determined. The joists are sized last. The area of the slab that each joist is responsible for supporting is known as its tributary area, which in a conventional slab is frequently a rectangle. This area, multiplied by the unit dead load of the slab, is assumed to be uniformly distributed across the joist, and the joist is sized according to the maximum moment and shear imposed upon it.

A collage of different patterns

Description automatically generated

The calculations associated with a rectangular, conventionally supported slab are relatively straightforward, as slab dimensions, spanning system, and joist spacing are the main geometric factors affecting load distribution. Introducing more complex geometries, such as those of a beam network, complicates the load calculations. Due to the indeterminate nature of floor slabs, precise load paths can only be approximated, anterd tributary areas are more challenging to determine (section 2.2). The method below outlines three strategies for analyzing the loads incident on irregular beam networks for isotropic (section 2.3), uniaxial (section 2.4), and orthogonally biaxial slabs (section 2.5). Section 2.6 describes how maximum spans and slab depth are determined, and section 2.7 outlines two algorithms for optimal beam sizing, one continuous and one using discrete options from the ASCE catalog of W-sections. This includes constraints for maximum assembly depth. Finally, section 2.8 addresses the design of the parametric slabs used for portions of the analysis.

This method is implemented in Julia v1.9.3 using the finite element analysis package Asap.jl.[22]

## 2.2 Tributary area calculation

Calculating the tributary areas of the beams in a network first requires a geometric description of the slab. When inputted, the beam network is described as a finite-element model (FEM) consisting of nodes, elements, and user-defined boundary conditions. The positions of the nodes (vertices) and connectivity of the elements (edges) are extracted and used to define a graph. The data structures are backward compatible with the original FEM and can be used to assign the results from geometric analysis to physical structural elements.

As the slab's external perimeter is not automatically embedded in the graph network, a convex hull is formed using a quickhull algorithm.[23] Quickhull for planar surfaces uses a divide-and-conquer approach to find the external boundaries of a cloud of vertices. The line connecting the bottom-leftmost and top-rightmost vertices is used to initialize the algorithm. Two triangles are drawn, connecting to the farthest point to the left and right of the line. The algorithm works recursively, using the new lines defined by the edges of the triangles as initializers, until there exist no more points outside of the edges of the last-drawn triangles. The convex hull vertices are retrieved and sorted clockwise.

A diagram of a triangle with lines and dots

Description automatically generated

The remainder of the graph analysis is based on half-edges, which define their directionality and are used to find which edges connect to form a cell. Interior edges are adjacent to two faces of the network, so to describe both faces, they must be traversed in two directions: from start to end and from end to start. Exterior edges, however, are only adjacent to one face, so they can only be traversed in one direction.

Once the perimeter of the slab is established, linear connections between adjacent convex hull vertices are cross-referenced with the connectivity matrix of the graph. If a graph is nonconvex and a corresponding edge is not found, the algorithm inserts a new edge to achieve convexity on the perimeter. The list of elements in the FEM is adjusted accordingly. All the exterior edges are assigned one half-edge, and the remaining edges in the graph are assigned two. These data are stored as an adjacency dictionary, where the adjacencies of each node are stored in clockwise order.

A diagram of a triangle with lines and dots

Description automatically generated

Connectivity network:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **A** | **B** | **C** | **D** | **E** |
| **A** | 0 | 1 | 0 | 0 | 0 |
| **B** | 0 | 0 | 1 | 0 | 1 |
| **C** | 0 | 0 | 0 | 1 | 0 |
| **D** | 0 | 0 | 0 | 0 | 1 |
| **E** | 1 | -1 | 0 | 0 | 0 |

Adjacency dictionary:

**A:** B

**B:** E, C

**C:** D

**D:** E

**E:** A, B

Finally, cells are found by iterating through the sorted adjacency dictionary. For example, node A is connected to B, B to E, and E to A. The cells ABE closes once A is reached again, and the adjacencies used are removed from the dictionary. This leaves just cells BCDE as a possible path. Once all minimum paths are traversed, the adjacency dictionary is empty, and the algorithm outputs the cells.

Currently, only convex networks can be analyzed in this manner. This has structural reasons: a convex corner in a beam network is likely to undergo large distortions without contributing to the stability of the structure. Nonconvex cells in the beam network are treated as invalid, and the slab is not analyzed.

Once the cells are found, they can be divided into tributary areas according to the desired slab type.

## 2.3 Isotropic slab loads

An isotropic slab has vector-independent material properties, so loads can flow equally easily in all directions. The base method builds off the assumption made in standard practice that loads flow to the nearest beam. Moreover, we assume that the loads incident upon a beam flow perpendicular to the beam axis. One way of solving this problem would be to discretize the cell into small squares, find the closest perpendicular point on each perimeter beam to a given square, and assign its load to the nearest beam at that point. This rasterization method is limited by rasterization density, however. A fine raster would yield accurate results, but at the cost of significant compute.

A vector-based method circumvents this issue by only performing vector intersections where necessary. Using bisectors, we divide the space into sub-cells that correspond to the regions of the cell closest to each beam. The sub-cell is further subdivided into rectangular strips running perpendicular to the beam. The area of the individual strips, each of which has width *s*, is then used to assign point loads along the beam.

A drawing of a rectangle

Description automatically generated

In an equilateral triangle, drawing bisectors from each corner produces a single intersection point. However, due to the nature of irregular polygons, the intersection of bisectors may not be so clean. Often, two bisectors will intersect before any others. In a quadrilateral, as shown in Figure @a, this yields two intersections. Drawing a line between these points suffices to delineate sub-cells. In higher-order polygons, there may be three or more intersections, requiring a recursive approach until the base condition of one or two intersections is reached. The full sub-cells are reconstructed from the layers of recursion.

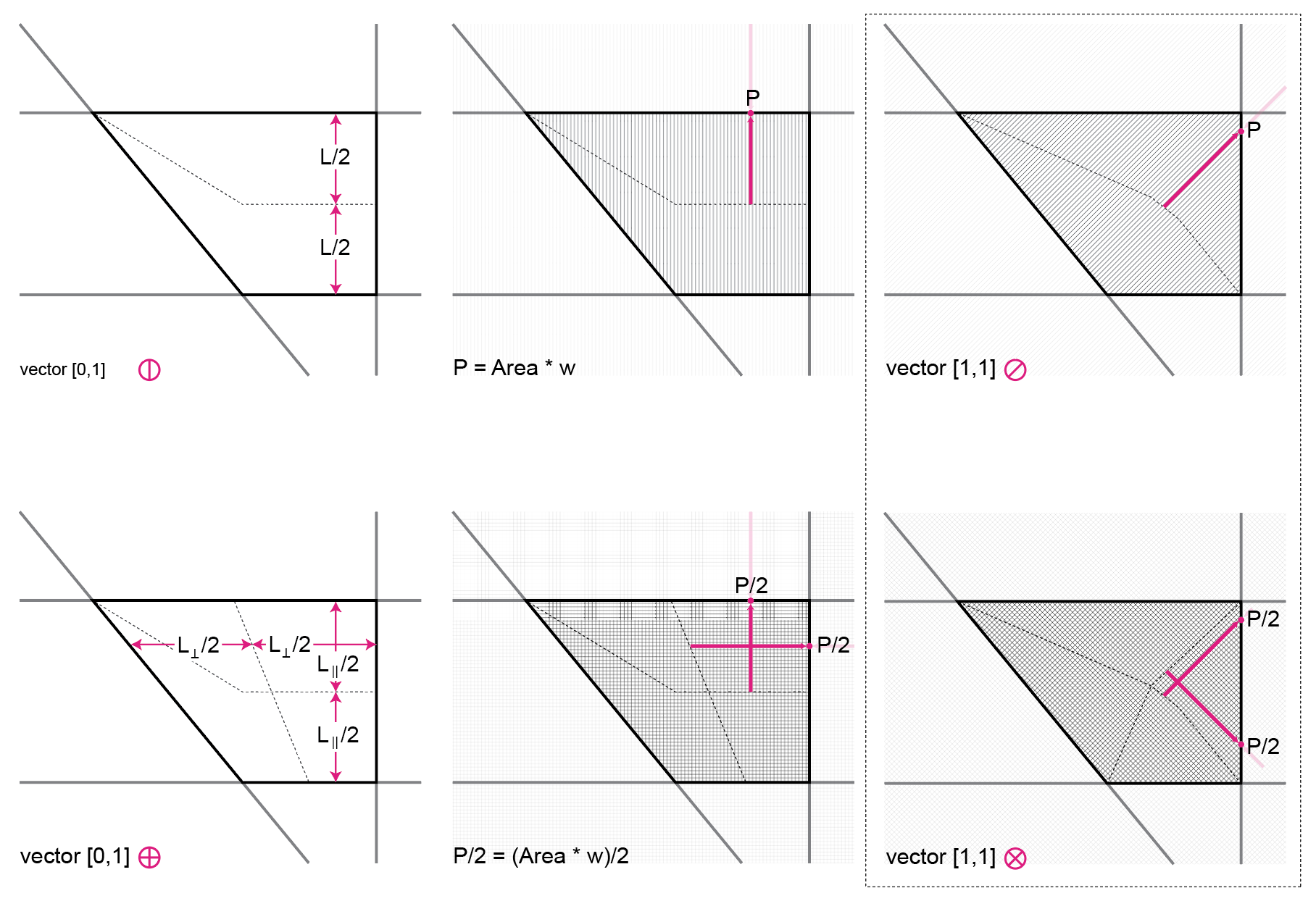
This method is specific to bisectors. For more complex subdivisions, such as ones determined by the stiffness of the individual beams, another approach is required. Relatively stiffer beams attract more load than relatively more flexible beams, so the angle of the subdivision line skews away from the stiffer beam, assigning it a greater tributary area. The problem is adjacent to finding the medial line of a polygon. Here, the problem is solved using a variation on wavefront progression.[24] In a typical wavefront progression, each edge is offset by unit distance, emulating ripples. The intersection events are solved, and the process is repeated until all wavefronts converge. To achieve a varying stiffness effect, the relative stiffnesses of the beams are analyzed (), and a set of corresponding subdivision vectors is produced. Each corner point is translated along that vector by unit distance. Connecting the points creates a new polygon, which shrinks with every iteration until convergence is reached.

Both methods are different ways of answering the same question. Nevertheless, there is no obvious answer for where the forces flow in an isotropic slab, so both are approximations. For most of the following analysis (section 3), the bisector method is chosen as it best approaches the standard methods used by engineers in practice. A direct comparison between the outcomes of the bisector and the stiffness-driven approach is given in section @@@.

## 2.4 Uniaxial slab loads

Uniaxial slabs only transfer load in one direction. In a concrete slab, these generally follow the arrangement of rebar or the corrugation of the underlying metal deck. Assuming again that the loads flow to the beam closest to them, vectors are projected from each beam in a cell. They are spaced distance *s* apart. To maintain regular spacing, the distance is measured perpendicular to the load vector and projected onto the beam. Vectors projected into the interior of the cell intersect with the opposite wall, and their intersection point is recorded. Vectors that do not intersect are discarded. This produces a vector field inside the cell. The length of the trimmed projection is halved and assigned as a point load along the beams that it intersects.

If a stiffness-based heuristic is desired, the projection is not halved but divided according to the relative stiffnesses of the start and end beams. This simulates how a load might be attracted to a stiffer beam along a single vector.



## 2.5 Orthogonal biaxial slab loads

Orthogonal biaxial slabs limit the flow of forces along two perpendicular vectors. Again, this effect is achieved through the arrangement of rebar within the slab, which is laid in a grid pattern. Like the isotropic slab, several possible approaches for approximating the flow of forces exist. Two were considered:

The first is a vector projection of the isotropic slab load along orthogonal vectors, where the load assigned to each direction is inversely proportional to the distance to the beam along that vector. The more distant beam would thereby receive less load. This yields usable results but does not directly simulate slab mechanics.

The second method, which is used in this analysis, replicates the rebar arrangement by overlaying two uniaxial analyses in perpendicular directions. In contrast to the uniaxial case, the final load is halved before it is assigned to the beam so that each direction is accorded 50% of the total load of the slab. Stiffness-based modifications are applied independently along each orthogonal vector, where they are treated like the uniaxial case.

## 2.6 Maximum span and assembly depth

The maximum span of a slab determines its minimum depth. According to ACI 318, the minimum depth required for deflection control of one-way continuous slabs is , and the minimum depth of two-way interior slabs without drop panels is , where is the maximum effective span of the slab.[19] Orthogonal and isotropic slabs are both treated as two-way slabs. The absolute minimum for all cases is 125 mm (0.125m) which is used if the calculated minimum depth is too shallow. This allows enough room for rebar.

To determine the maximum span of a cell in an isotropic two-way slab, we first extract the normal vector of each edge beam. The vector is drawn at each corner of the cell and tested for intersections against the remaining edge beams. If an intersection is found, the length of the line is recorded. This is repeated for each normal vector, and the maximum value of the list of spans is considered the maximum span. The maximum span may run along an edge, for instance, the long edge of a rectangle. This method allows us to find the longest distance over which moments act in a slab, as the farthest point from a given point along the perimeter of a polygon will always be a corner.[25]

A diagram of a triangle with lines and points

Description automatically generated

A similar test is performed for the uniaxial slab, but instead of using the normal vectors from the edge beams, intersection events are tested from the corners using the slab's load-carrying vector. The orthogonal slab is similar to the uniaxial case, but both vector directions are tested to find the maximum span.

The slab depth is directly determined by the maximum span according to the ACI minimum deflection depth. Because maximum spans are determined on a cell-by-cell basis, there are two options for sizing the concrete slab. First, the total slab can be uniformly sized according to the depth required to support the longest maximum span of the cells in its beam network. Second, cells can be individually sized according to their individual maximum spans. This affects the total volume of concrete used as well as the manufacturability of the slab, and the effect of this choice is displayed in figure @.

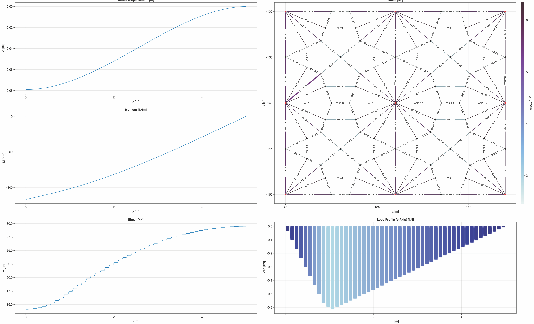
The mass of the slab is then equal to its depth multiplied by its unit area and the density of concrete (2400kg/m3). To convert to kN/m3, this value is multiplied by *g* (9.81m/s2).

## 2.7 Optimal beam sizing: continuous and catalog

Point loads are assigned by multiplying the area of the strip calculated in sections 2.3-2.5 by the sum of the unit dead load of the slab found in section 2.6, and the design loads: Live loads are assumed to be 30psf (1.44 kN/m3), and dead loads 50psf (2.39 kN/m3).[21]

To replicate the structural action of a frame, nodes designated as columns are supported by stiff 1m-tall columns. If no columns are designated by the user, the nodes on the convex hull (section 2.2) are used. The column bases are fixed. Beam joints at column nodes are released, and all other joints are moment connected.

To assign the point loads, the slab is again considered as an FEM. The geometric properties of the slab used in previous sections are now abstractly reduced to the calculated loads. Asap.jl allows point loads to be initialized at a parameter along the beam, from which the equivalent end-loads and moments are calculated. The cumulative effect of the point loads is then input into the load vector and solved along the beam for displacement, internal moment, and internal shear forces.[22] 200 samples are taken along the length of the beam.



@figure modify

To size an optimal structural member, its moment and shear capacity must exceed the nominal moment and shear forces, multiplied by a safety factor, at every point along the beam. Since, in this paper, the beams are sized prismatically and have the same cross-sectional properties throughout, we can size the beams to the maximum values of moment and shear. Once these constraints are met, its deflection is checked against a maximum deflection limit. The deflection limit is generally a serviceability limit and is placed at for floor beams, where is the length of the beam. Should the deflection limit be exceeded, the beam is sized up.[20]

W-sections are used throughout this paper. In standard practice, engineers are provided with a catalog of standard sections, which are iterated over until an appropriate section is found. Local displacement is checked by inputting the new section values into the FEM and checking the local vertical displacement of the element. The section catalog is sorted from smallest to largest (W4X13 to W43X335).

|  |
| --- |
| Pseudocode Discrete Section Sizing |
| ϕb = ϕv = 0.9  serviceability limit = 360  **maximum displacement** = **beam length** / serviceability limit  objective values = [ ]  valid sections = [ ] |
| *for* each **section** in **section catalog**  **beam volume** = **section area** \* **beam length**  *if* **section depth** > **maximum assembly depth – slab depth**  *continue*  *elseif* absolute(**section Mu**) < ϕb \* **Mn**  *continue*  *elseif* absolute(**section Vu**) < ϕv \* **Vn**  *continue*  *else*  check **local** **displacement**  *if* **local displacement** > **maximum displacement**  **penalty** = (**local displacement** – **maximum displacement**)2 \* 104  *else*  **penalty** = 0  *end*  **objective value** = **beam volume** + **penalty**  *end*  *append*(objective values, **objective value**)  *append*(valid sections, **section**)  *end*  **optimal section** = valid sections[*minimum index*(objective values)] |

An alternative method for sizing sections would be to treat the individual design parameters controlling section depth, flange width, flange thickness, and web thickness as continuous variables for optimization. This requires imposing more geometry constraints, for without the catalog, there is no automatic guarantee that the sections are valid according to AISC standards. They include:

1. Area > 0
2. 2 \* flange thickness ≤ section depth
3. Web area / compression flange area ≤ 10
4. Section depth / web thickness ≤ 260
5. 0.1 ≤ moment of inertia of compression flange / moment of inertia of section ≤ 0.9

Moment, shear, and deflection constraints also still apply. The constraints are combined into a single constraint vector that is evaluated at every iteration. The objective function remains the same as in the discrete case (beam volume + deflection penalty).[12] As the constraints and objectives are differentiable, it is possible to do a local gradient-based optimization using the CCSAQ algorithm, which converges faster than non-gradient-based methods. CCSAQ is a relative of the method of moving asymptotes (MMA) commonly used in structural optimization but uses local quadratic approximation instead of MMA’s more complex convex approximation.[26,27] For this optimization, MMA was found to be more likely to fall into a local minimum.

Since the initial frame is populated with arbitrary sections (W8X35), it can be productive to perform additional iterations for both discrete and continuous optimization, where the frame is re-initialized with the list of optimal sections. This allows any local minima to be avoided and can be used to verify deflection limits. Although only a single iteration is used for most of these results, the effect of additional iterations is shown in section @ for completeness.

## 2.8 Parametric and nonparametric slab design

Every slab in the dataset is designed as a continuous four-slab system with nine columns, of which eight are on the perimeter and one in the center. Individual slabs are 10mx12m, making the full assembly 20mx24m.

The slabs tested are designed in two dimensions in Rhino and converted into a JSON format compatible with FEM. An example is provided in Appendix @. The first set of “topology” slabs is manually designed and includes symmetrical, asymmetrical, and business-as-usual topologies. Two other sets of slabs are generated based on the template of a “grid” slab and a “star” slab, the latter of which is inspired by Nervi’s form-following geometries.

The grid parameterization is based on waffle slabs. The number of beams running in the x and y directions, respectively, is tested between 0 and 5 to find the optimal density of steel beams, beyond which the mass of added steel outweighs the benefits of slimmer slabs.

The star slab seeks to provide a quantitative solution to the best arrangement of force-following beams. Precedents that use principal moments frequently rely on manual adjustment to determine the optimal position for ribs, since there are infinite possible load lines one might follow. This design space allows for the discrete testing of different arrangements given the constraints of manufacturability using standard construction methods.

A collage of different squares

Description automatically generated

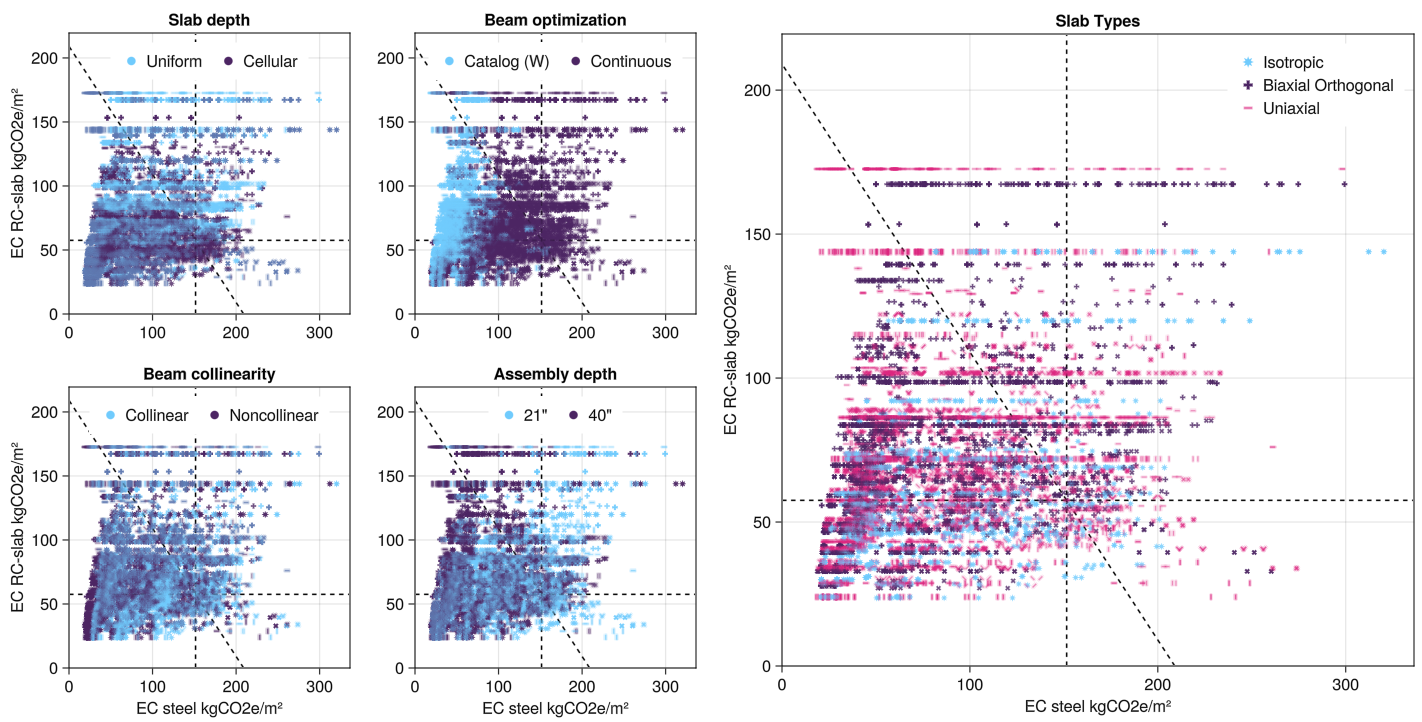
# 3.0 Results (3000)

To identify the key factors affecting floor slab efficiency, 12096 slabs were individually simulated and sized. Of these, 36 “topology” slabs (“t”) were of a unique design (figure @a) and 72 were parametric and varied across two axes of a design space. Half of the parametric slabs simulated fall within the “grid” design space (“g”), the other half within the “star” (“s”) (section 2.8). These slabs were then analyzed under combinations of several different design choices, including slab type and vector orientation (7 variations), slab depth (2 variations), beam optimization (2 variations), beam collinearity (2 variations), and maximum depth (2 variations). .

## 3.1 Parameter comparisons

The results of the different tests are shown below. Figures @a-@d show pairwise comparisons between the normalized embodied carbon of the business-as-usual and optimal variations of a) , b) beam optimization, c) beam collinearity, and d) assembly depth, whereas figure @e compares the different slab types and vector orientations. Table @ breaks down the mean and standard deviations for each comparison.

The embodied carbon is divided between steel (x-axis) and slab (y-axis). Steel embodied carbon is the embodied carbon of the steel beams of the grillage, excluding connections. Slab embodied carbon includes both rebar as well as concrete, where rebar is assumed to constitute 1% of the total slab volume in isotropic and uniaxial slabs, and 2% for biaxial orthogonal since the rebar is laid out in a two-directional grid. The volume of the rebar is increased by 5% to allow for turnups and cutoffs.



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Parameter** | **Variation** | **Mean /** kgCO2/m2 | | | **Standard Deviation /** kgCO2/m2 | | |
| *Steel* | *Slab* | *Total* | *Steel* | *Slab* | *Total* |
| Slab depth | Uniform |  |  |  |  |  |  |
| Cellular |  |  |  |  |  |  |
| Beam optimization | Catalog (W) |  |  |  |  |  |  |
| Continuous |  |  |  |  |  |  |
| Beam collinearity | Collinear |  |  |  |  |  |  |
| Noncollinear |  |  |  |  |  |  |
| Assembly depth | 21” |  |  |  |  |  |  |
| 40” |  |  |  |  |  |  |
| Slab type | Isotropic |  |  |  |  |  |  |
| Orthogonal biaxial |  |  |  |  |  |  |
| Uniaxial |  |  |  |  |  |  |
| Slab type &  vectors | Isotropic |  |  |  |  |  |  |
| Orthogonal biaxial (1,0) |  |  |  |  |  |  |
| Orthogonal biaxial (1,1) |  |  |  |  |  |  |
| Uniaxial (1,0) |  |  |  |  |  |  |
| Uniaxial (0,1) |  |  |  |  |  |  |
| Uniaxial (1,1) |  |  |  |  |  |  |
| Uniaxial (1,-1) |  |  |  |  |  |  |

## 3.1.1 Slab depth

Slab depth can be calculated uniformly or cellularly. When cell minimum depths are calculated (section 2.6), a cellular slab retains those depths on a cell-by-cell level whereas a uniform slab takes the maximum calculated slab depth and applies it to all the cells. The uniform slab requires less formwork and resembles a more typical construction method, whereas the cellular slab can produce material savings by removing material where it is unnecessary to support the loads across the maximum spans but would require more elaborate, and potentially wasteful, formwork. The embodied carbon of the formwork is not included in this analysis.

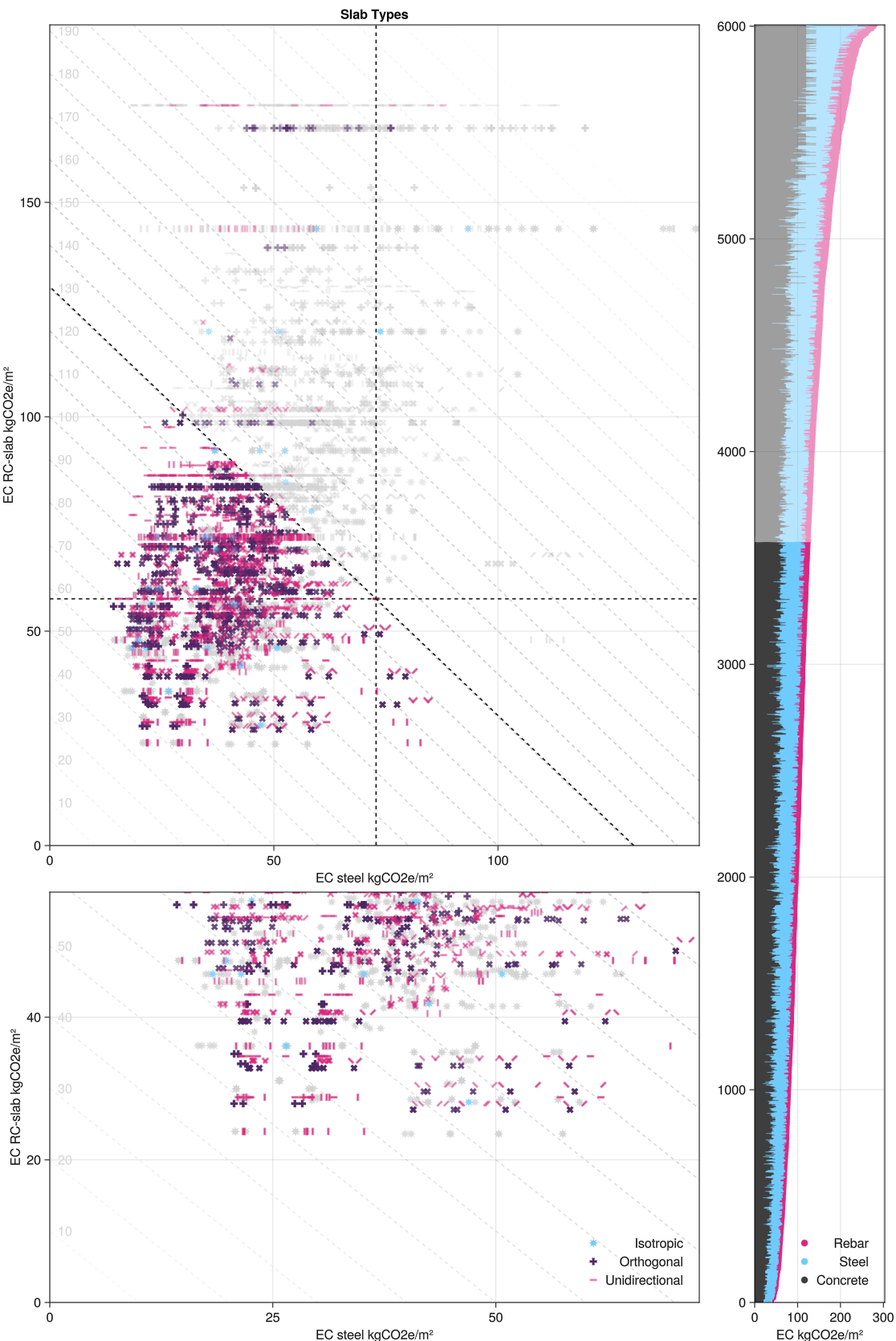
In general, the cellular slabs have less total embodied carbon than the uniform ones. Shallower slab depths impose lighter dead loads on the beams, allowing for smaller steel sizes, so both slab and steel embodied carbon are reduced. The uniform slab results are banded horizontally along slab embodied carbon, suggesting that there are several slabs which have the same maximum slab depth, for instance when a cell spans the full 10m width or 12m length of a slab, which occurs several times in the dataset (e.g. slabs p[3,2], g[1,5] and slabs p[3,3], p[7,4], p[9,3], g[4,1], respectively).

## 3.1.2 Beam optimization

Figure @b shows the difference between discretely selecting steel sections from the AISC catalog of W-sections and continuously optimizing over the shape parameters of a section until the minimum shape is found. Since this parameter only affects the steel after slab loads have been applied to it, the primary shift is horizontal. Continuous beam optimization makes a significant difference to the total efficiency, suggesting that a greater range of available sections in the catalog might allow for more overall efficiency in the design of standard slabs. @ comment on mean and SD

## 3.1.3 Beam collinearity

After optimization, collinearity can be imposed in post-processing by scaling all sequences of collinear members to the maximum section size in that sequence. This reflects the standard practice of using monolithic beams for longer spans and connecting perpendicular members to them, for instance at a T-joint, rather than dividing the beam into optimal section sizes. This simplifies both the connections and the assembly process. As in section 3.1.2,



@Figure: geometry choices (star graph where vectors point the many options?)

@Figure: different slab depths line graph

1. Architecture 2030. (n.d.). *Why The Built Environment – Architecture 2030*. Retrieved March 19, 2024, from https://www.architecture2030.org/why-the-built-environment/

2. Foraboschi, P., Mercanzin, M., & Trabucco, D. (2014). Sustainable structural design of tall buildings based on embodied energy. *Energy and Buildings*, *68*, 254–269. https://doi.org/10.1016/j.enbuild.2013.09.003

3. Bischof, P., Mata-Falcón, J., & Kaufmann, W. (2022). Fostering innovative and sustainable mass-market construction using digital fabrication with concrete. *Cement and Concrete Research*, *161*, 106948. https://doi.org/10.1016/j.cemconres.2022.106948

4. Kromoser, B. (2020). *Ressourceneffizientes Bauen mit Betonfertigteilen. Material—Struktur—Herstellung* (pp. 307–356). https://doi.org/10.1002/9783433610206.ch3

5. Iori, T., & Poretti, S. (1960). *Pier Luigi Nervi’s Works for the 1960 Rome Olympics*.

6. Halpern, A. B., Billington, D. P., & Adriaenssens, S. (2013). *The ribbed floor slab systems of pier luigi nervi*. *54*, 127–136.

7. Morley, C. T. (1966). The minimum reinforcement of concrete slabs. *International Journal of Mechanical Sciences*, *8*(4), 305–319. https://doi.org/10.1016/0020-7403(66)90031-2

8. Rozvany, G. I. N. (1976). *Optimal Design of Flexural Systems: Beams, Grillages, Slabs, Plates and Shells* (First Edition). Pergamon.

9. Bolbotowski, K., He, L., & Gilbert, M. (2018). Design of optimum grillages using layout optimization. *Structural and Multidisciplinary Optimization*, *58*(3), 851–868. https://doi.org/10.1007/s00158-018-1930-6

10. Ismail, M. A., Mayencourt, P. L., & Mueller, C. T. (2021). Shaped beams: Unlocking new geometry for efficient structures. *Architecture, Structures and Construction*, *1*(1), 37–52. https://doi.org/10.1007/s44150-021-00003-y

11. Mayencourt, P., & Mueller, C. (2020). Hybrid analytical and computational optimization methodology for structural shaping: Material-efficient mass timber beams. *Engineering Structures*, *215*, 110532. https://doi.org/10.1016/j.engstruct.2020.110532

12. Carruth, M. A., Allwood, J. M., & Moynihan, M. C. (2011). The technical potential for reducing metal requirements through lightweight product design. *Resources, Conservation and Recycling*, *57*, 48–60. https://doi.org/10.1016/j.resconrec.2011.09.018

13. Lee, K. J., Hirt, N. K., & Mueller, C. T. (2024). *Geometry, strength, and efficiency: Tracing the standardization of North American structural steel, 1888-present*. 8th International Congress on Construction History, Zürich.

14. Meibodi, M. A. ; J. (2018). Smart Slab. Computational design and digital fabrication of a lightweight concrete slab. *ACADIA // 2018: Recalibration. On Imprecisionand Infidelity. [Proceedings of the 38th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA) ISBN 978-0-692-17729-7] Mexico City, Mexico 18-20 October, 2018, Pp. 434-443*. https://papers.cumincad.org/cgi-bin/works/paper/acadia18\_434

15. Jewett, J. L., & Carstensen, J. V. (2019). Topology-optimized design, construction and experimental evaluation of concrete beams. *Automation in Construction*, *102*, 59–67. https://doi.org/10.1016/j.autcon.2019.02.001

16. Li, L., Yan, J., & Xing, Z. (2013). Energy requirements evaluation of milling machines based on thermal equilibrium and empirical modelling. *Journal of Cleaner Production*, *52*, 113–121. https://doi.org/10.1016/j.jclepro.2013.02.039

17. Whiteley, J., Liew, A., He, L., & Gilbert, M. (2023). Engineering design of optimized reinforced concrete floor grillages. *Structures*, *51*, 1292–1304. https://doi.org/10.1016/j.istruc.2023.03.116

18. Fang, D., Brown, N., De Wolf, C., & Mueller, C. (2023). Reducing embodied carbon in structural systems: A review of early-stage design strategies. *Journal of Building Engineering*, *76*, 107054. https://doi.org/10.1016/j.jobe.2023.107054

19. American Concrete Institute. (n.d.). *ACI 318-19 Building Code Requirements for Structural Concrete*.

20. American Institute of Steel Construction. (2016). *ANSI/AISC 360-16 Specification for structural steel buildings*. https://www.aisc.org/publications/steel-standards/aisc-360/

21. American Society of Civil Engineers. (2023). *ASCE/SEI 7-22 Minimum Design Loads and Associated Criteria for Buildings and Other Structures*. https://ascelibrary.org/doi/10.1061/9780784415788.sup1

22. Lee, K. J. (2023). *aSAP* [Julia]. https://github.com/keithjlee/Asap (Original work published 2021)

23. Greenfield, J. S. (1990). *A Proof for a QuickHull Algorithm*. https://www.semanticscholar.org/paper/A-Proof-for-a-QuickHull-Algorithm-Greenfield/051334587ea704e17f5c7c4a191ac84938265e66

24. Palfrader, P., & Held, M. (2015). Computing Mitered Offset Curves Based on Straight Skeletons. *Computer-Aided Design and Applications*, *12*, 414–424. https://doi.org/10.1080/16864360.2014.997637

25. Bertsimas, D., Tsitsiklis, J. N., & Tsitsiklis, J. (1997). *Introduction to Linear Optimization* (unknown edition). Athena Scientific.

26. JuliaOpt. (2024). *NLopt* [Julia]. JuliaOpt. https://github.com/JuliaOpt/NLopt.jl (Original work published 2013)

27. Svanberg, K. (1987). The method of moving asymptotes—A new method for structural optimization. *International Journal for Numerical Methods in Engineering*, *24*(2), 359–373. https://doi.org/10.1002/nme.1620240207

# Outline

1. Motivation and background – why does this matter (1000-1500 words)
   1. Climate change
      1. What is a flexural system
      2. Despite the existence of many different floor topologies and construction methods, there is a lack of understanding of the interaction between the slab surface and its supporting beam network, a superstructure I here call the beam grillage.
      3. The standard beam grillage used in construction is the primary-secondary beam system, where secondary beams carry loads from their tributary areas to the primary beams, which then carry them into the columns. This paper concerns the diverse arrangement of secondary beams and the flow of tributary loads to the primary beams.
   2. Literature Review
      1. Pictures (unsplash?) of normative slabs and nervi; construction image before rebar is cast
      2. Topology optimization of structures
         1. Moh (beams + floors)
         2. Ashley
      3. An infinite arrangement of secondary beams is possible. ETH has shown that topology optimized slabs can lead to significant material efficiency gains but they require large robotics facilities to design and manufacture the formwork. There has not yet been an investigation into the flexible production of optimal slabs using conventional manufacturing methods such as steel welding.
      4. Analytical methods (Rozvany) are inflexible (mathematically complex) and the results are difficult to size for a real structure (he calculates the section sizes according to the moment values, but we can’t really check against building code (?)), Bolbotowski’s numerical analysis is not prescriptive at all but shows results consistent with those of Rozvany
      5. Floor slabs have historically been supported by secondary beams because they are easy to manufacture and analysis difficult without computational means. This paper describes a geometric method for calculating the tributary areas and sizing the beam layout with reference to real structural code and manufacturing methods
2. Methodology
   1. Scope – steel with reinforced concrete slabs?
   2. Tributary area
      1. Slabs are indeterminate, anything we do is a best guess but we can at least have a relatively good guess
   3. Topology optimization of beams
   4. Picture of all topologies and their names
   5. Lateral stability not considered – slabs act as diaphragms
3. Levers + do they matter  
   This section discusses the differences between different forms of analysis used in this paper. To portray the primary differences, take the Grasshopper slabs and make small multiples
   1. Discrete vs continuous (can implement depth restriction by bounding the set) ✓
   2. Compare constant sizing vs. individually sized ✓
   3. Minimum slab depth (21”, 40”)
   4. Slab topology:
      1. One way slab on deck
      2. Two way slab of reinforced concrete, orthonormal or normal (and if orthonormal, which vector)
   5. Slab geometry: Cellular vs. uniform sizing ✓
   6. Make a table with mean/total SD for pairwise comparisons
   7. Isotropic and biaxial 40” varying vectors
   8. Uniform mass distribution over slabs
4. M e g a p l o t
   1. Significant choices
   2. Fade out the ones worse than the standard slab
5. Grillage topology
   1. 4-6 discrete choices with large variations
6. Grillage geometry
   1. Description of parametric model

Three cases — each of these cases can be parametrically adjusted to plot out the optimal design for each design space.

* 1. Case 1: regular slab (4 bays) ~ the grid!
  2. Case 2: self-derived slab ~ the star! (nervi and eth-inspired)

1. Assembly depth
   1. 2-4 topologies, hold all else constant, run 5-10 max depths and plot line plot with x-axis being max depth an y-axis total EC
   2. Utilities (have not looked at this at all?)
2. Case study, working through the layered decision tree
   1. Business as usual
      1. Relevant levers – continuously sized, uniform slab depth
   2. Bad choice
   3. Best choice that we can find
      1. Add post-processing stage of shaving additional material off the steel beams (you can carve it from the middle of the steel beam or the flanges of the beam, which can then be recycled for other use)
3. Future work
   1. Different materials (full reinforced-concrete structural system that can be cast in place), timber floor
   2. Topology optimization for the optimal discovery of the perfect floor slab (in defined design spaces such as the ones shown in section 7

# Questions

* What are the most important factors in efficient floor design?
* Are there trade-offs between different design choices (is there a pareto front?)
* How sensitive is a given design to different geometric, topological, sizing, and slab choices?