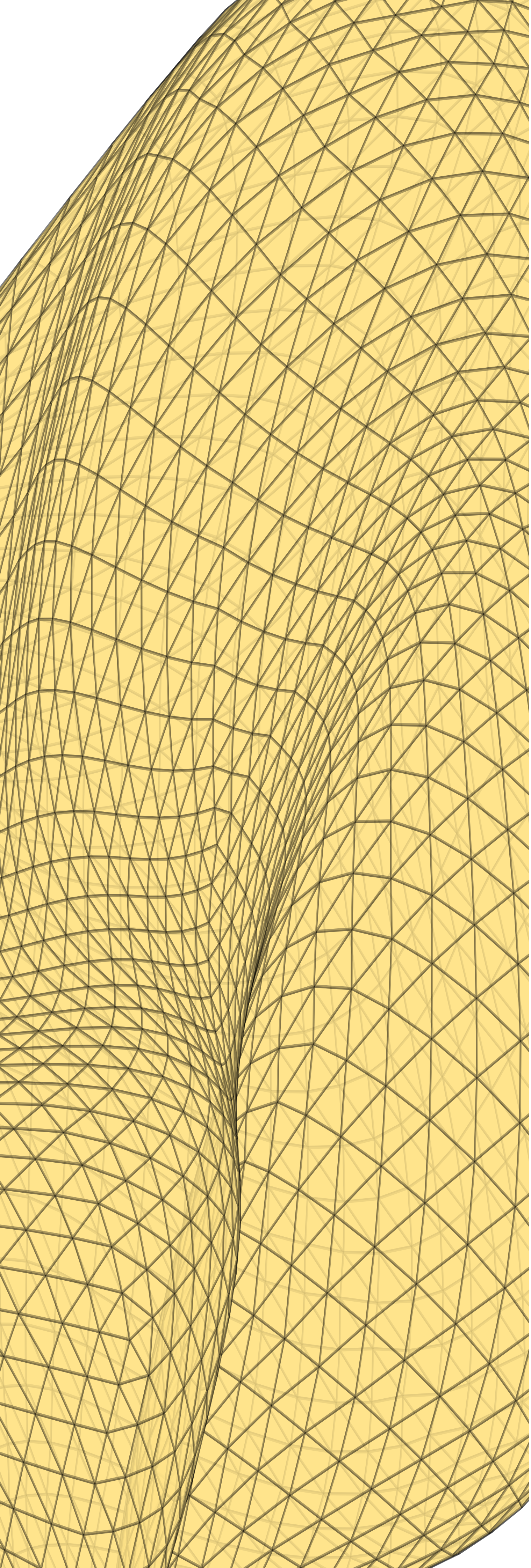
FLOOR PLAN GENERATION

Raban Ohlhoff - 000457528
Master in Architecture

Architecture and Design
Faculté d'Architecture La Cambre-Horta
Université libre de Bruxelles
Belgium - Brussels
April 7, 2022

Contents



Introduction

The increasing application of computer science in a wide variety of industries has become reality since the vulgarization of technology and the facilitated access by the broad public. The applications are as diverse as the underlying mechanisms and range from simple code-based optimizations to simulations in every direction to fully digitized frameworks. That scientific fields such as natural, structural, economic and engineering sciences benefit from increasing digitization is fairly natural and may be easily explained by the relevance of mathematical modeling, computational simulations, and the need for significant calculating power, but this interaction is far less intuitive when one considers the example of the artistic or humanistic fields. This report examines the interaction between computer science and the field of architecture, which is situated intermediate between science and art. In particular, it examines the topic of automatic floor plan generation as a starting point for exploring applications such as simulation, analysis, data manipulation, and machine learning based on the generated data. Each of these topics has been extensively researched by the academic world in the past few years and therefore provides extensive documentation of scientific reports. This explains why this work is largely based on third party work, but attempts to bring the individual topics together in a relevant way to create a workflow that can potentially provide new insights. Due to the significant amount of existing research and information, a extensive state of the art and thoroughly reading of relevant research is essential.
In addition, a significant emphasis is placed on the interaction of computer-based algorithms and the inherent creativity of the architectural profession, which has proven difficult to combine with digital means. Thus, this is neither a purely scientific work nor an exclusively experimental investigation, but rather the structured documentation of a problematic investigation with sensitive consideration of interdisciplinary aspects.

Abstract

This project tries to intervene in the architectural design process by applying different prediction models in order to pre-select certain volumetrics based on user-defined properties. To achieve this, several steps with different software library and algorithm application are needed, since the domains to be obtained vary from geometric to environmental simulations. First, the focus is on the parametric generation of an apartment floor plan using Python, Blender's Sverchok and various algorithms such as Voronoi Diagrams, KD-Trees and Genetic Algorithms. After generating a manifold spatial configuration, it can be analyzed from a variety of geometric aspects using the Python library Topologic. It is important that these two steps are in constant feedback exchange to combine geometric data with spatial analysis evaluation and to save the information in a file format appropriate for the data. These formats can vary from simple CSV files to database formats or graph data.

Furthermore, the evaluation stage can be complemented by physical and environmental analyses such as the application of light and radiation based simulation using Radiance, Vi-Suite, Honeybee and OpenStudio. It is also possible to perform a simulation of the energy performance of the architectural object using Energy+, Vi-Suite-Energy and OpenStudio's core component.

The evaluated geometric synthetic data can then be used as training data for a machine learning model after adapting the data shape with Pandas using SciKit-learn, Tensorflow or Pytorch.

Whether it will be a CNN, GAN or simpler algorithms like Random Forest will be determined based on the data properties. The last step concerns the reconstruction of the data in geometric form into a three-dimensional model, which will eventually be extended, enhanced, stored and displayed in a common open source BIM format such as IFC using IFCOpenshell, BlenderBIM, Topologic, Opencascade and Homemaker.

Special attention will be paid to the consistent use of python-based libraries to ensure the best possible compatibility of the individual software interactions. Furthermore, for didactic, ideological and compatibility reasons, only open source projects will be employed. During the execution of the individual steps, possible problems, suggestions, solutions, proposals for improvement and, just as important, ideas for further research of the topic are noted, which are described in detail in the following report.

This project tries to explore an unconventional approach in the connection between architectural generative design, spatial and environmental simulation in combination with machine learning.

Framework

This report is part of a long line of research on the interaction between machine learning and architectural design. The following project was developed in the framework of the Architecture and Design Studio at the Université libre de Bruxelles, which is characterized by an openness to new technologies in connection with architecture. The studio's focus is to raise interdisciplinary questions between technology, research and architecture that can be freely developed without having to remain in their limited domains.

The aim is to develop hypotheses about the future of architecture, taking into account all areas that may be of interest in terms of the design process of the project. In every aspect of the project development, it is important to take into account the analysis of new scientific knowledge and to explore the achievements of current science. Yet, it is also important to base experiments on inventions that have emerged throughout history.

This studio is not primarily about inventing novel objects, but rather to deepen the study of a given topic with all existing publications and research in order to formulate a relevant research hypothesis. The goal is then to begin a process of experimentation accompanied by meticulous documentations. The goal of this process is to find answers to the questions raised in the preliminary stage. In this way, the experimentation cycle will advance the research and may possibly lead to a concrete solution, but in any case will raise a new set of questions that can be pursued subsequently.

The operation of the studio is closely linked to the FabLab of the Faculty of Architecture, providing access to many tools for designing and experimenting. This space will also act as a library of skills acquired by each individual, and thus the emergence of collective knowledge. As a result of the restructuring around the Covid 19 pandemic, the workshop has become a paperless studio, which means that the visual representations and research objects are mostly digital.

Hypothesis

The use of intelligent neural networks and models trained through machine learning to optimize traditionally manual processes has become the norm nowadays. Machine learning is no longer limited to computer science, but extends to any domain as long as a database to be analyzed is involved or can be created. It is therefore not surprising that learning models have also found their way into architectural optimization.

However, the application of machine learning in architecture is wide-ranging and can be useful in any project development process. For example, intelligent parametrization can help find the adequate shape even before a concrete project is modeled, a mechanical analysis of the existing conditions and framework can be helpful to determine an approximate volumetry. Furthermore, it is possible to generate through intelligent algorithms the partitioning of the internal space and thus propose several adapted plans, which can lead to a qualitatively enhanced experience for the occupants. This optimization is not limited to the two-dimensional space and can therefore provide suggestions for optimal circulation or optimization of daylight incidence throughout the building. Besides the conceptual phase, it is equally possible to optimize the BIM model through various machine learning algorithms. All these processes are no longer visions of the future, but rather have become the norm, although often automated and therefore not directly visible. This report will mainly focus on the application of pretrained models, pseudo intelligent and evolutionary algorithms in the conceptual phase to attempt to automatize the generation of floor-plans in order to obtain an optimized plan through subsequent steps.

Premise of this work is the assumption that there is a direct re-

lationship between external conditions such as space connections, solar radiation, shading, humidity, wind flow, heat formation, soil conditions, air quality, pedestrian circulation or traffic congestion and the occupant's perceived quality of living.
The main hypothesis treated in this report addresses whether and to what extent learning algorithms can simplify, accelerate, and/or optimize the architectural design process. Is the increasing application of intelligent architecture considered purposeful or will traditional values be lost? Is the automation of processes traditionally performed by humans again just an idealized label or are there tangible benefits for both clients and architects? Furthermore, this thesis will investigate to what extent it is possible to perform a complete workflow from generation to simulation, analysis, prediction and back to generation without resorting to proprietary software and thus describe a step towards the democratization of architecture and its digital tools.

State of the Art

Since this project deals with diverse topics, the collection of previous works is accordingly divided into hierarchical subgroups. First, the role and emergence of computational design with parametric, generative and algorithmic design as subgroups should be considered. Its origins go back to 1960, when Sutherland's SKETCHPAD software was a major step towards the automation of architectural drawings and the digital parameterization of the relationships between individual geometric units. In the following years, several individual approaches to building information modeling inspired by important computational design conferences developed. With the establishment of various computer-aided design software, the way to visual programming interfaces was paved and with it the access to parametric and generative design tools for the masses. In this work, the geometric parametric tools apart from backend python coding will focus on the visual interface of the Sverchok addon for the three-dimensional processing software blender, which makes use of various python libraries. The first goal of this work, the automatic generation of synthetic floorplans, is as a topic widely researched and this with many very diverse tools, which can be roughly divided into simple algorithms, intelligent algorithms and machine learning enhanced algorithms. The group of simple algorithms is subdivided into geometric space partitioning by Voronoi diagrams and its derivatives like Delaunay triangulation, Lloyd's Algorithm, Orthogonal Voronoi Diagram and weighted Voronoi Diagrams. Also mentioned are aproximating schemes for subdivision such as Cutmull-Clark and interpolating ones such as Butterly subdivision Surfaces. Kdimensional trees, originally from the family of such algorithms, differ from the previous ones, despite the essential similarities, by a subdivision of input vertices as opposed to polygons, which thus allows a sensitive control over control points. The method of slicing tree structure forms the connection between point and area oriented subdivision. Another method of simple algorithms is the shape grammar methodology which allows rule based geometry generation and thus space for variation in defined limits. This topic has gone through many variations such as the CGA shape or parametric shape grammar and is recognized as a programming language in its own right through successful simulation of turing machines. Far enough away from these methods to define themselves as a group of their own are physics solver based methods like attraction force models or MagnetizingFPG algorithm to some degree. In this paper, however, we will mainly focus on the latest and more interesting topology based method developed by Wassim jabi. Topologic is not a layout generation method but rather an opencascade based geometry processor that processes non-manifold topologies and thus reduces the architectural spaces to simple cells and cell complexes. This method allows layout generation by combining the above mentioned methods and paves the way to a simple geometry to file workflow. Furthermore, topologic's boundary representation method facilitates environmental simulations for the subsequent plan evaluation stages. However, by far the most significant advantage of using topologic is the analytical approach to continuous space and relations awareness, which allows the arbitrary addition of apertures such as doors and windows, as well as the graph generation of various topological parameters and last but not least an interface to common BIM file types such as IFC, BREP and JSON.

More important in the literature of floorplan generation than the mentioned approaches are intelligent methods like evolutionary algorithms since the 1990s. These are input populations that go through biology-inspired mechanisms such as reproduction, mutation, recombination, and selection through fitness evolutionary phases, resulting in an optimization of the fitness function. Evolutionary algorithms can be combined with simple geometric methods by means of goal definition and can also be used for optimization by means of evaluation analyses. Computational intelligence also includes machine learning and neural networks, which are by far the most widely used methods in the field of automated floorplan generation. However, the above methods differ drastically from artificial neural network applications in that the latter are based on the learning of features in defined training datasets. In the context of plan generation, these datasets are real plans designed by humans. Thus, the computer generated floor plans resemble the input drawings but leave room for differentiation which is limited by the learning process based on existing plans or evaluation based on suitably defined examples.

Process

This work is an experimentally exploratory approach, meaning the primary goal is not to achieve an optimized process, but rather to critically analyze the individual steps. Thus, by repeatedly questioning the method, insights can be gained that will be beneficial in the following phases. Moreover, an essential focus lies on answering the formulated questions presented in the hypothesis, which means that the individual steps should be reected on several layers in order to gain not only technical, but also moral, ethical, and social insights.

First, the topic of parametric automated plan generation must be addressed in depth. In today's world, artificial intelligence is used as a kind of selling point, a solution to all complex problems, which raises the question of whether this assessment is truthful, or is this term simply associated with idealized solutions? Once

these questions are answered and a concrete concept has emerged from the abstract term, it becomes possible to think about connections between architecture and machine learning that simplify existing design processes, blueprints, simulations or constructive procedures.
Just as important as a clear understanding of the subject matter an in-depth review of the available libraries and their functions, in order to create a customized network diagram covering the interactions. The open source community, through platforms such as Github, Gitlab, and OSArch-community, provides an appropriate and direct exchange with developers and interested individuals and a complete understanding of the functions and procedures, as well as, in most cases, detailed documentation. With the help of various forums and exchanges with developers, it is possible to gain a comprehensive understanding of the software in question in a relatively short period of time, thus advancing the ideas of the project.

Objectives

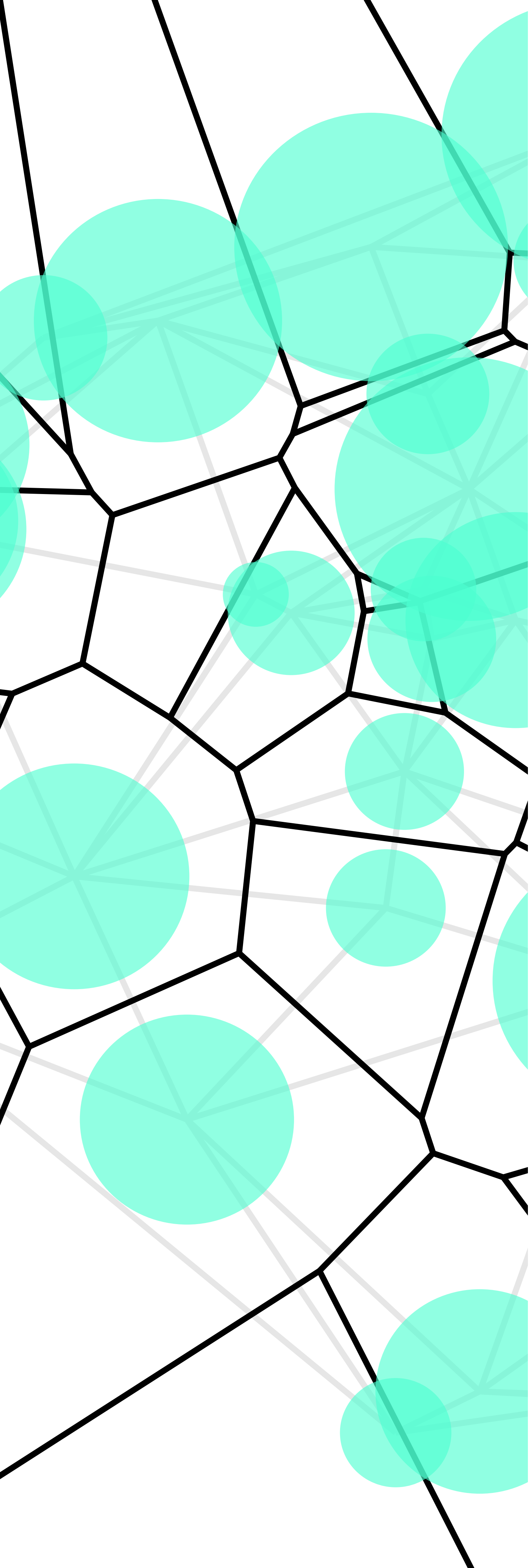
The experimental freedom explained above also leads to flexibility in terms of the defined aims. In general, there are objectives for each stage, but this does not mean that a step has failed if they turn out differently than expected or formulated in advance. Thus, the stage of parametric generation has as a desired result the generation of synthetically generated models that can be constrained by certain parameters. These can be the number of living rooms, the number of residents, the area, the volume or the shape of the floor plan. The end result of this phase should be a database of the different geometries that describes each situation as accurately as possible while still requiring a minimum amount of points. In the next stage, the simulations would generate new evaluation values that could be added to the geometric database. Further, the next stage is to train a model that describes the relationship between the individual values as accurately as possible by means of a graph. Finally, the last stage is to provide an accurate visualization of the predicted scenario.

Roadmap

- [ ] Information Gathering
  - [ ] Search Resources
    - [ ] Code Repositories
    - [ ] Academic Papers
    - [ ] Books / Reports / Articles
    - [ ] Student Works
  - [ ] Create Bibliography
  - [ ] Adequate Software
    - [ ] Code Documentation
    - [ ] Explanatory Resources
- [ ] Geometric Generation
  - [ ] Available Methods
    - [ ] Algorithms

      - ☐ Evolutionary Generation
      - ☐ KD-Tree
      - ☐ Pulga Physics
      - ☐ Shape Grammar
      - ☐ Shape Packing
      - ☐ Orthogonal Convex Hull
      - ☐ Voronoi
      - ☐ Orthogonal Voronoi
      - ☐ Lloyd
      - ☐ Delaunay
      - ☐ Power Diagram
      - ☐ Polygon Division
      - ☐ Recursive Subdivision
      - ☐ Recursive Bisection
      - ☐ Recursive Angled Bisection
      - ☐ Surface Population
      - ☐ Point Cloud
    - ☐ Machine Learning
      - ☐ CNN
      - ☐ GAN
      - ☐ Random Forest
  - ☐ Parametric Generation
    - ☐ Sverchok
    - ☐ Grasshopper
  - ☐ Evaluation
    - ☐ Data Set Search
    - ☐ Comparison
    - ☐ Esthetic Verification
    - ☐ Functional Verification
    - ☐ Ideological Verification
      - ☐ A Pattern Language
- ☐ Simulations / Analysis
  - ☐ Familiarization
    - ☐ Environmental
      - ☐ Energy
        - ☐ Energy+
        - ☐ Openstudio-Energy
        - ☐ Ladybug
        - ☐ Vi-Energy
      - ☐ Lightning
        - ☐ Radiance
        - ☐ HoneyBee
        - ☐ Openstudio
        - ☐ Vi-Radiance
      - ☐ Air
    - ☐ Spatial

        - ☐ TopologicPY
        - ☐ IFC
        - ☐ Blender BIM
      - ☐ Structural
    - ☐ Usage
    - ☐ Data Type Examination
    - ☐ Verification
- ☐ Data Manipulation
    - ☐ Database Types
    - ☐ Data Preparation
- ☐ Machine Learning
    - ☐ Methods
- ☐ Visualization
    - ☐ Back Feeding
    - ☐ Geometry Viewer
- ☐ Pertinence Study
    - ☐ Usage Application

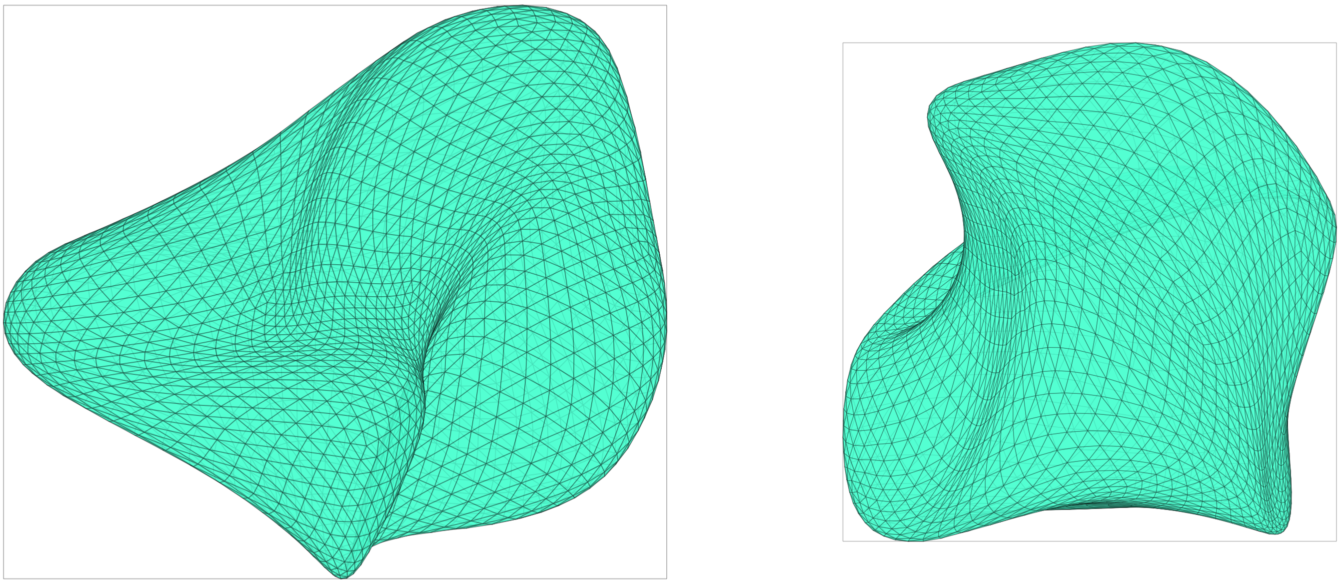


Layout Generation

The first stage of this work deals with the automated generation of different apartment layouts using different geometry processing software. An alternative to the generation approach is the acquisition of existing architectural datasets provided by real estate and research groups. Advantages of reality-based datasets are the certainty of architectural feasibility and construction of the individual plans, but disadvantages are a conservative approach in creative terms and a high risk of traditional bias due to the predominance of local and traditionally influenced architectural methods. Therefore, this experimental phase is primarily concerned with testing different floorplan generation methods and their derivatives as well as their possible combination. Simple algorithms like voronoi diagrams, intelligent methods like evolutionary algorithms and data driven approaches like neural networks are explained and evaluated by different experiments. Criteria for the evaluation are simplicity in the construction process, computing power, time and memory, integration and interaction with used software, purity of the generated geometry, degree of readiness for the simulation stage, simplicity of the layout for data storage but above all spatial quality, creativity in form finding and feasibility. After experimentation, the most appropriate method or combination of algorithms for the following steps is selected and an appropriate set of different layouts is generated. In order to increase the variation in the synthetic dataset, results from different methods can be mixed, but they must not deviate from the generally defined quality level. Furthermore, it is a requirement to understand the functionality of the algorithms sufficiently to allow slight adaptations and combinations with other functions and to achieve variation in the individual applications. In general, simplicity is preferred to complexity, creativity to ordinariness and variation to repetition.

Genetic Algorithm

The Genetic Algorithm developed by Holland and De Jong is an optimization model based on Darwin's natural selection-based theory of evolution in biology. Its operation is based on biological mechanisms such as crossover, recombination, mutation and selection by fitness evaluation acting on a population of defined size. Due to their mode of operation, genetic algorithms and their derivatives are particularly adaptable and offer a parallelization of the solution finding. However, for optimal application, the right parameters have to be set, such as crossover and mutation rate, population size, iterations and fitness boost. These parameters are problem-dependent and there is a risk of an undesired and non-optimized result if the starting conditions are set incorrectly. The construction of an evolutive algorithm begins with the determination of the variable function to be optimized, which consists of an unlimited number of components and has a tendency towards zero. After the framework conditions of the algorithm have been defined, the variables to be varied must also be determined, whereby it is important to ensure a proportional relationship between parameter variance and fitness evaluation function. In Figure 27, the surface area of the X- and Y-dimensional bounding box of an irregular three-dimensional body was defined as the fitness function. The population of the genetic algorithm acts on the three Eulerian rotation axes of the body and thus achieves a minimization of the Z-section of the irregular shape by iterative mutation and fitness evaluation.

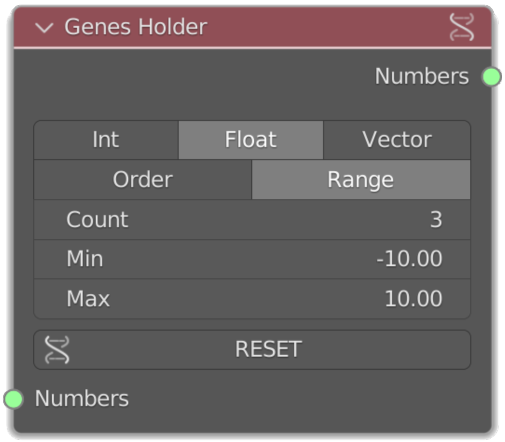


(a)

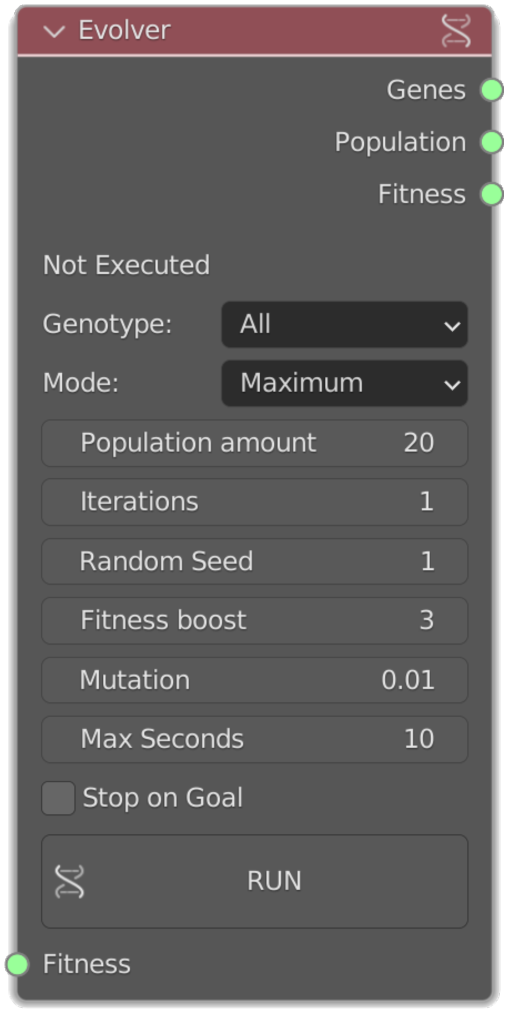
(b)

Figure 1: (a) blah (b) blah

With increasing iterations and an increase of the population number, the optimum is progressively approached, but the question arises how such a method works theoretically and how it proves itself on more complex problems? In Python there are several established evolutionary algorithm libraries with different integration mechanisms ranging from native python implementation to Keras and Pytorch or even scikit-learn integration. Since this work is primarily focused on generation and manipulation of visual and three-dimensional entities, the python natieve Blender implementation of the sverchok Genes solver was chosen as the experimentation tool. This provides a seamless integration into a visual scripting environment and thanks to sverchok's Blender integration, Blender's features such as the variety of available export file formats can be easily used.



(a)



(b)

Figure 2: (a) blah (b) blah

In order to test the floorplan-layout-generation capabilities of the evolution algorithm in Blender, different rectangles with defined X and Y sizes are first generated parametrically, each of these space representations being anchored by its origin location and rotation in two-dimensional space. The shape of each unit remains constant and represents the space sizes of the different apartment components. However, the lacolization and orientation of these modules remains free and is defined as a population parameter.

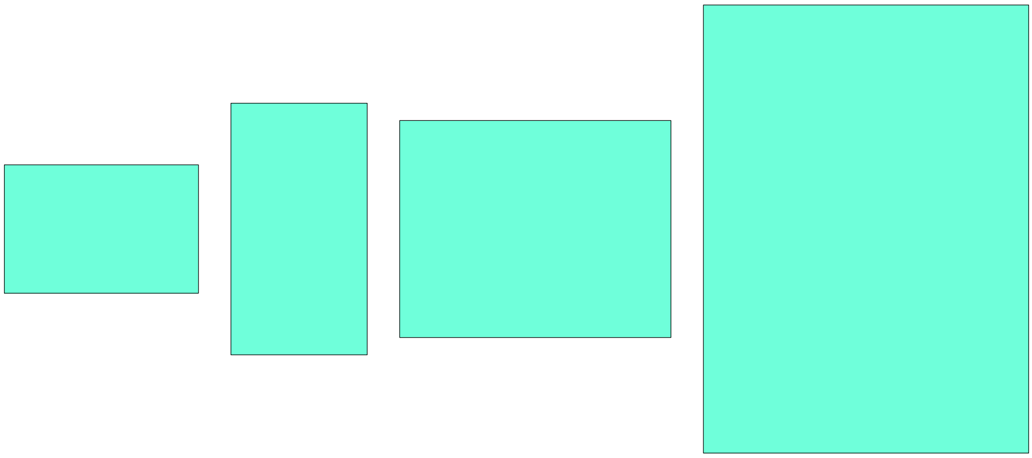
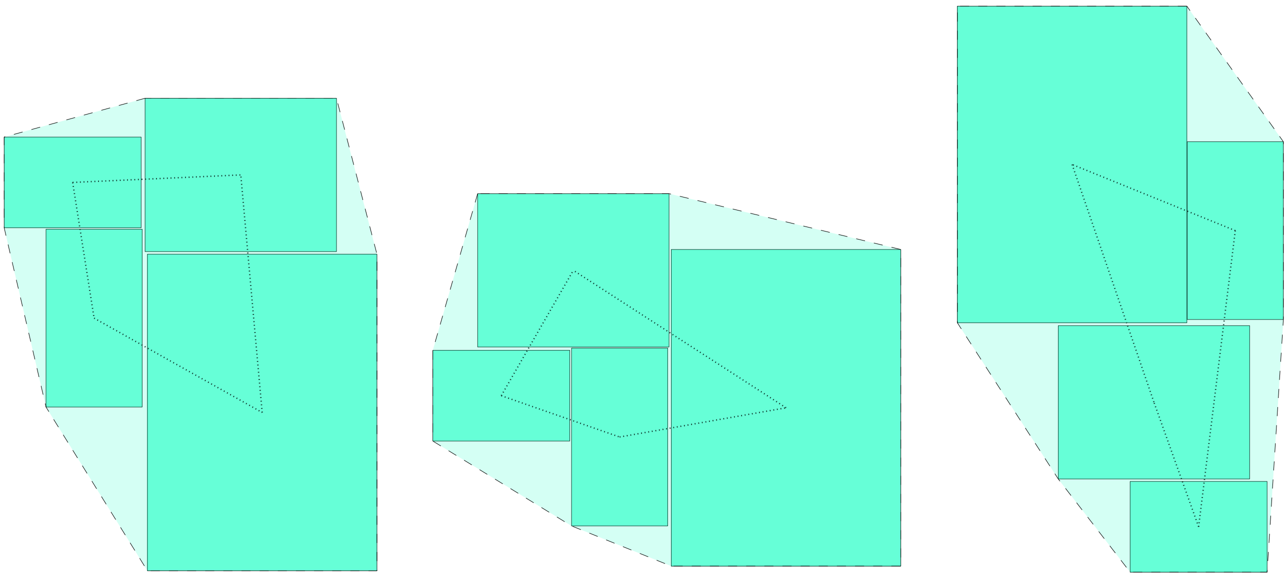


Figure 3: test

The fitness function is composed of the total length of the path of the UV connection of the individual rectangle origins, added to the total area of the two-dimensional convex hull with respect to the Z-axis of the overall geometry. Thus, the smaller distance between the individual units is rewarded in parallel by the total area and the individual distance of the center points, while also avoiding the overlap of the individual rectangles. This is achieved by an irradiation function which evaluates the number of geometries formed by a constant check of the boolean intersection to see whether the total number increases. If this is the case, a defined irradiation variable is added to the fitness function. The population size is set to 500, the fitness boost to 5, the mutation rate to 0.3 with an iteration of 5.



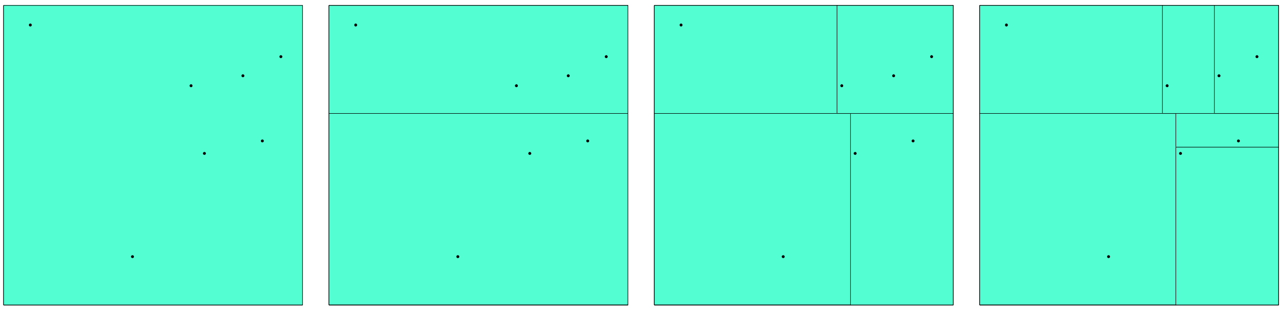
(a) (b) (c)

Figure 4: (a) blah (b) blah (c) blah

The different results generated parametrically by changing the random seed of the algorithm give random results with an amazing variance. Advantages of this method are the possibility of parameterization during the execution of the algorithm, a parallelization of the optimization, the possibility of defining different spatial relations by multipliers, the adaptability of the framework conditions and a variability of the obtained results. However, the main disadvantage is the constant distance between the spaces, which complicates the generation of boundary representation geometries and thus the complexity with respect to later environmental simulations or topological analyses. Furthermore, this increases the impurity of the geometry files to be stored. Possible solutions to this problem are a change of the framework and an extension of the irradiation function due to the overlap of the single units, which could integrate the size of the overlap surface. Furthermore, it would be possible to implement a second method that could lead to the cleanup of the overall geometry, but there is a risk of altering the basic geometry by causing alternating angles.

K-D Tree

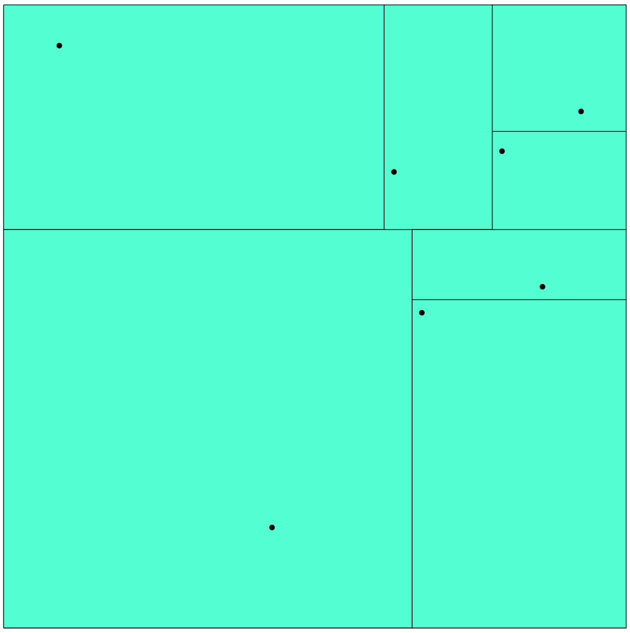
The k-dimensional tree data structure is a space partitioning primarily used in computer science for search algorithms and thus belongs to the family of binary space partitioning trees. It is the partitioning of pointclouds in a k-dimensional space. In this work I will limit myself to the two-dimensional space, since the floorplan layout can be described sufficiently in this dimension. In such a data structure, the dataset is divided into branches by nodes and branches, creating successive levels. Each of these branches leads to a leaf, which carries the coordinates of exactly one point.



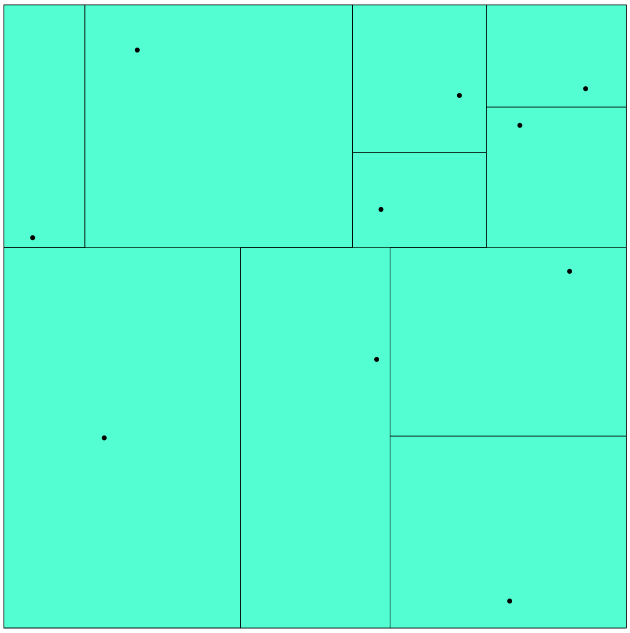
(a) (b) (c) (d)

Figure 5: (a) blah (b) blah (c) blah (d) blah

The functionality of such a data tree can vary, but the basic principles remain the same and can be implemented in a simple way in python without using external libraries, building the tree from the root upwards. First, the entire set of data points is considered and a sorting axis is determined based on the depth of the points. After these points have been sorted according to the axis, the node point is determined. This is located on the median of the point list and determines the coordinates of the subarea, which in the following step divides the point list into two child lists. The orientation axis of this intersection is also determined by the depth of the dataset to be split. If the data set to be divided consists of only one point, the leaf of the tree is reached and the iteration is terminated at this node. Thus, by repeating these steps, the complete tree is created using the logarithm.



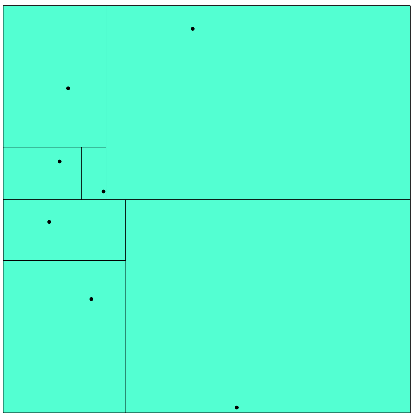
(a)



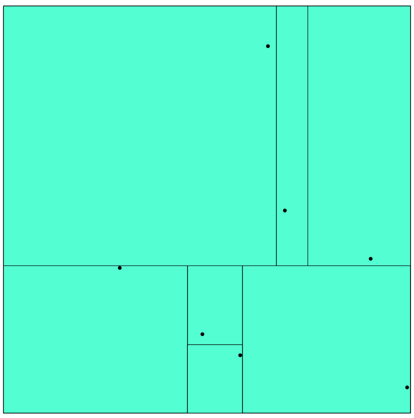
(b)

Figure 6: (a) blah (b) blah

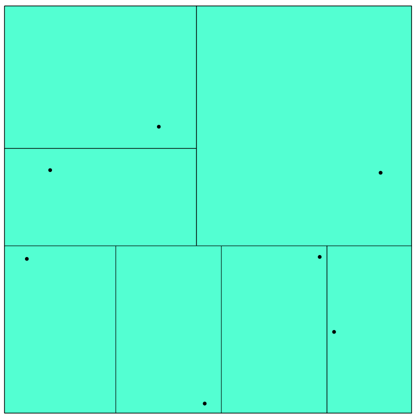
In Blender's Sverchok, creating a spatial K-d tree partition is relatively simple and requires only list manipulation, mathematical operators, slicing planes and loops. The points to be split and the corresponding planar area can be freely defined and are described in the examples shown by X and Y size defined rectangles and randomly selected points on this area. To avoid too small sheet spaces, a minimum distance between these points is defined. The number of these points is unlimited and can be defined accordingly, whereby their number determines the number of spaces of the floorplan.



(a)



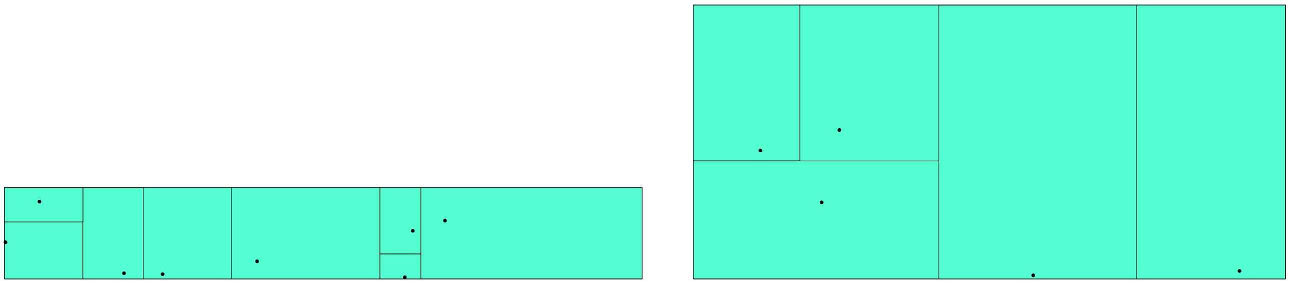
(b)



(c)

Figure 7: (a) blah (b) blah (c) blah

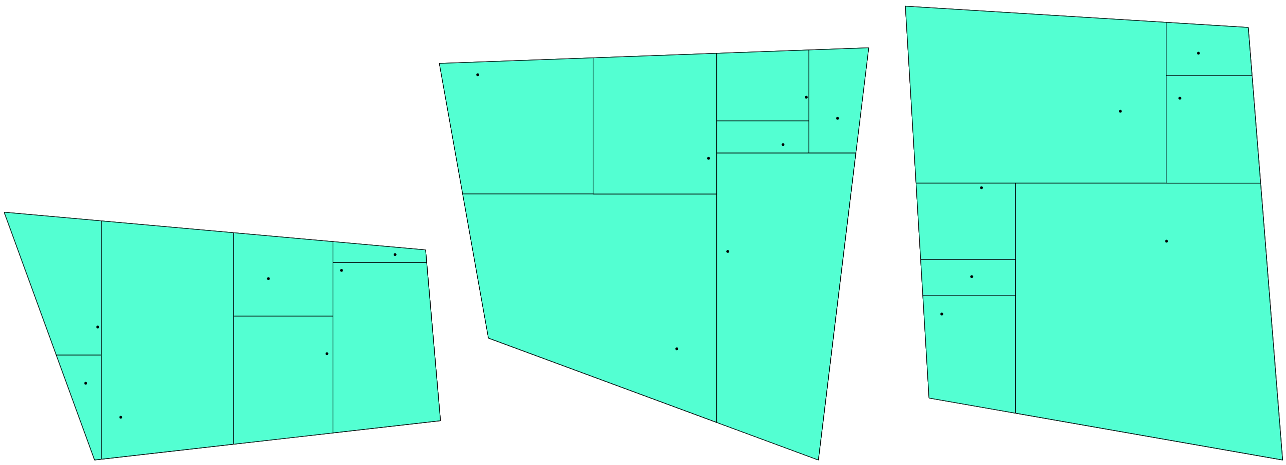
The main advantages of the described method are its speed and simplicity, the generation of natively connected units which simplify the export to boundary condition geometries, the possibility to parameterize the generation process, the determination of localizations of the individual spaces and the choice of the underlying volumetry of the contours of the plan.



(a) (b)

Figure 8: (a) blah (b) blah (c) blah

The outlines of the individual shapes can assume any shape of the rectangles and are thus adaptable to the different frame conditions. Furthermore, it is possible without problems to determine more complex shapes as input geometries, from deformed rectangles to irregular polygons. However, this method also has its drawbacks, such as the difficulty of generating composite apartments in which individual units protrude beyond the floorplan contour as isolated shapes. Thus, it becomes complicated to generate shapes such as terraced garages or irregular facades.



(a) (b) (c)

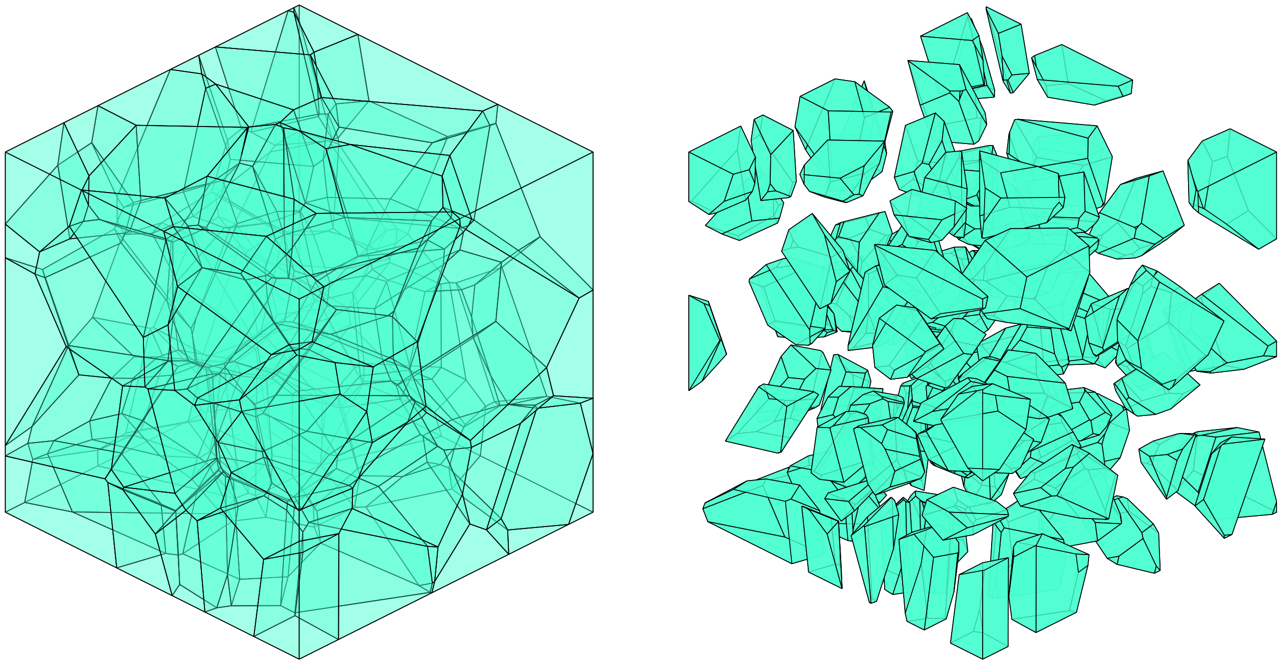
Figure 9: (a) blah (b) blah

KD-Tree and Genetic Algorithm

A possible combination between the k-dimensional space partitioning method and the evolutionary algorithm could be to optimize the variables of space contour, space number, but especially the point coordinates with a user defined fitness function according to the desired results. Such a combination would optimize the floorplan generation according to the objectives, but it would not solve the above mentioned problems.

Voronoi Diagram

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.



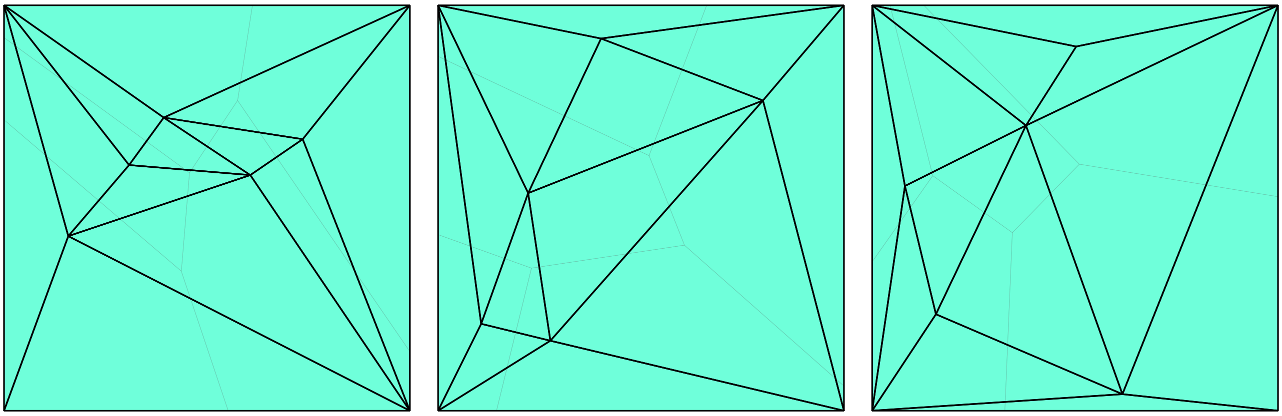
(a) (b)

Figure 10: (a) blah (b) blah (c) blah

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Delaunay Triangulation

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.



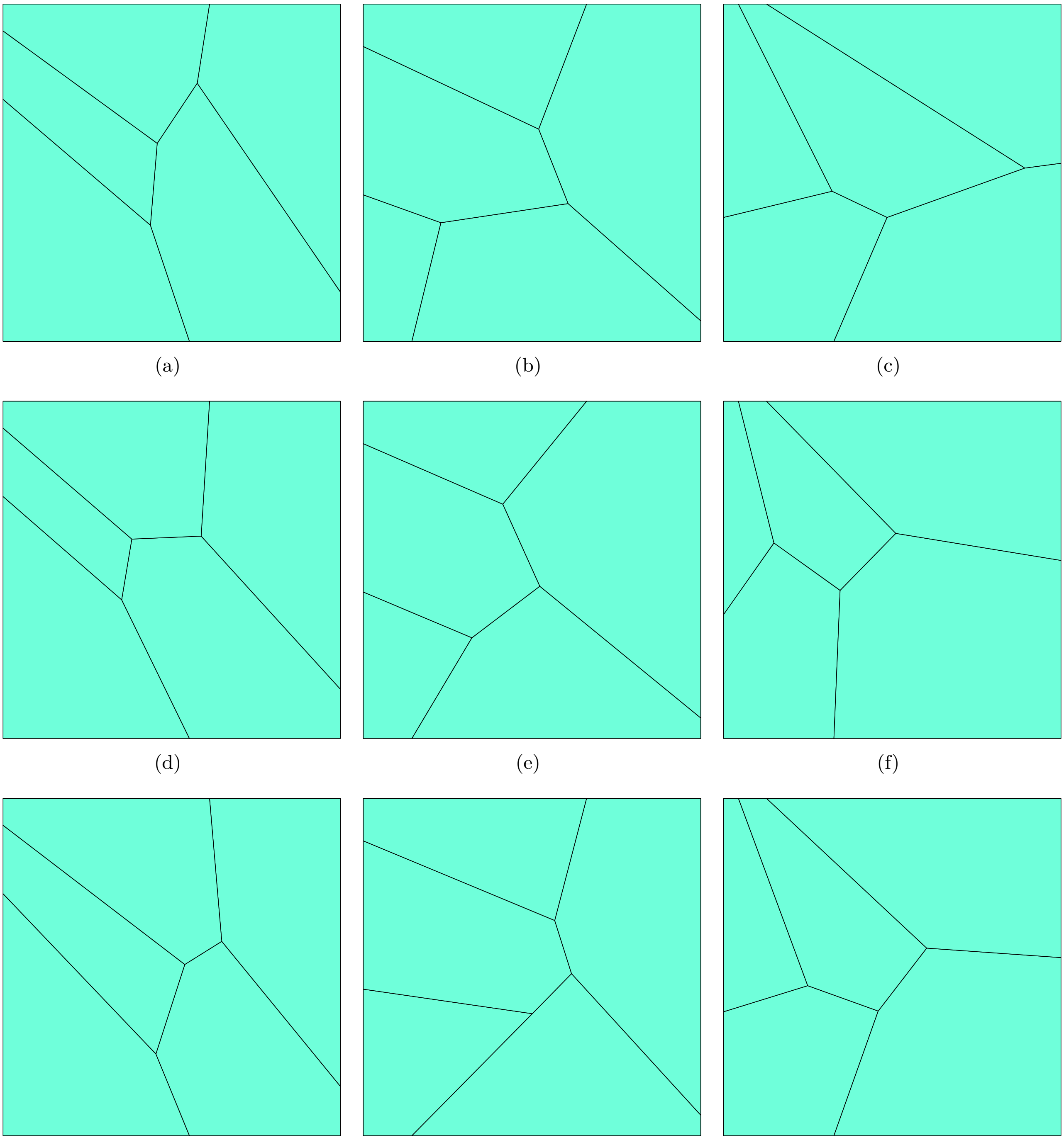
(a) (b) (c)

Figure 11: (a) blah (b) blah (c) blah

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Lloyd's Algorithm

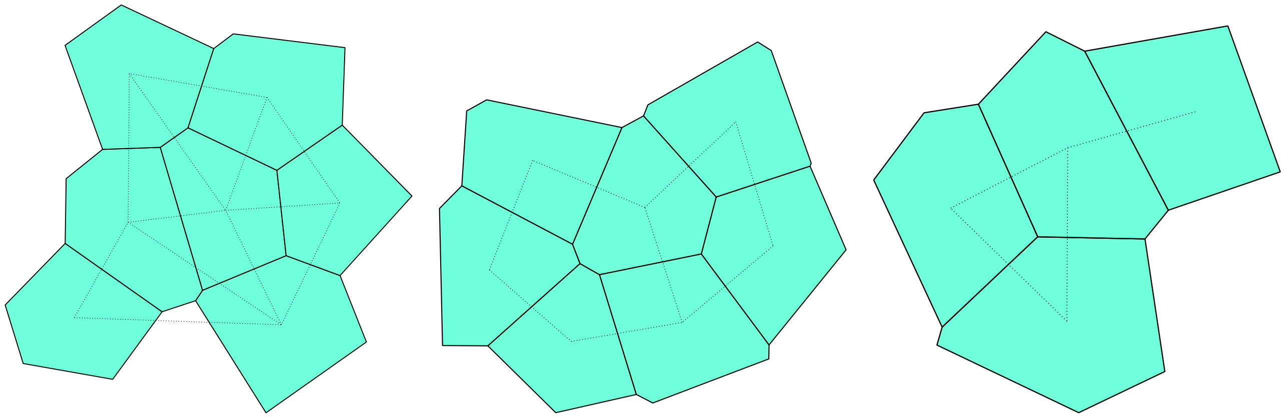
Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 12: (a) blah (b) blah (c) blah (d) blah (e) blah (f) blah (g) blah (h) blah (i) blah

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

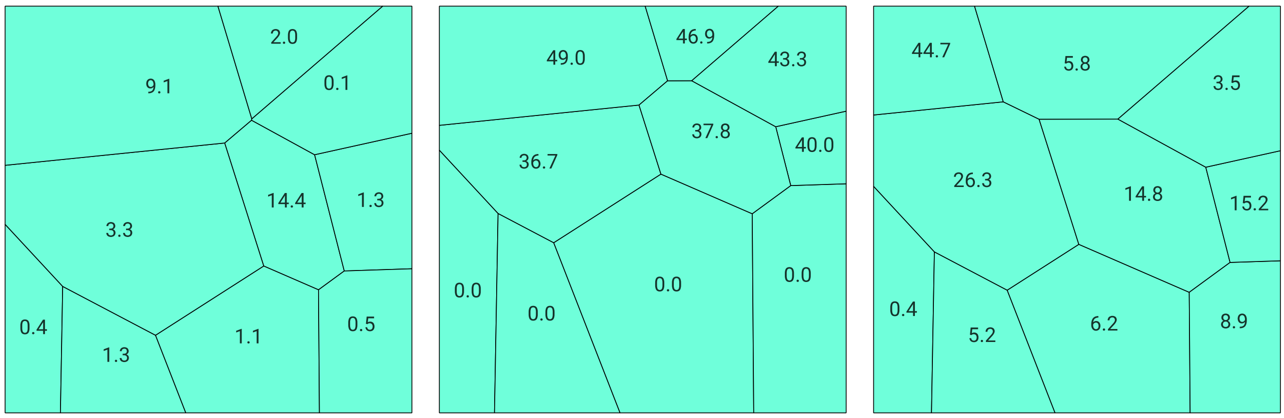


(a) (b) (c)

Figure 13: (a) blah (b) blah (c) blah

Laguerre-Voronoi

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.



(a) (b) (c)

Figure 14: (a) blah (b) blah (c) blah

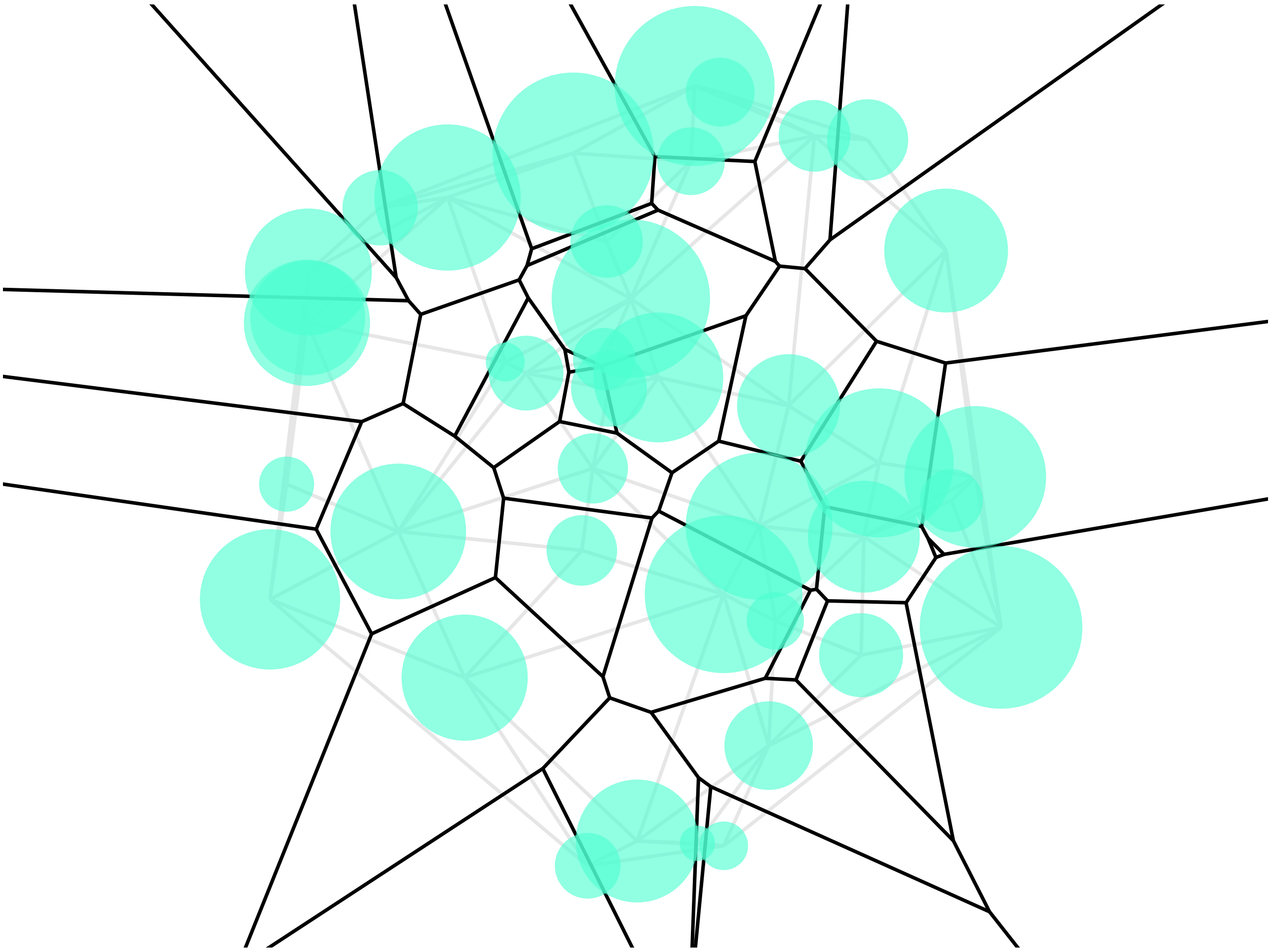


Figure 15: test

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Orthogonal Voronoi Diagram

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

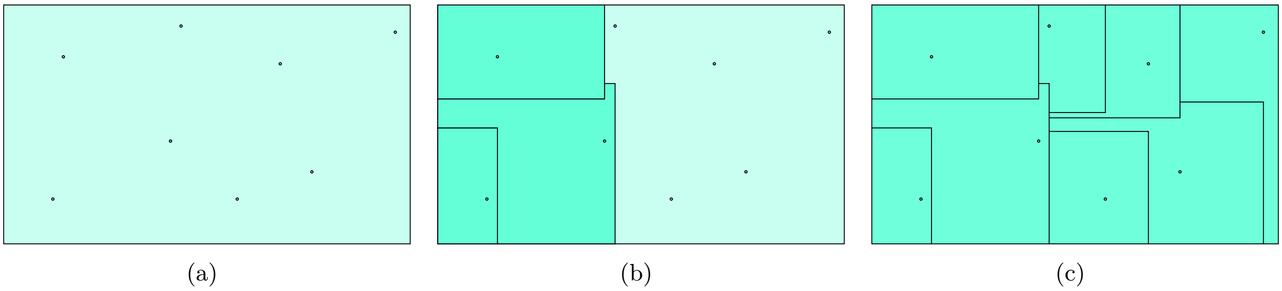


Figure 16: (a) blah (b) blah (c) blah

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Convex Hull

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

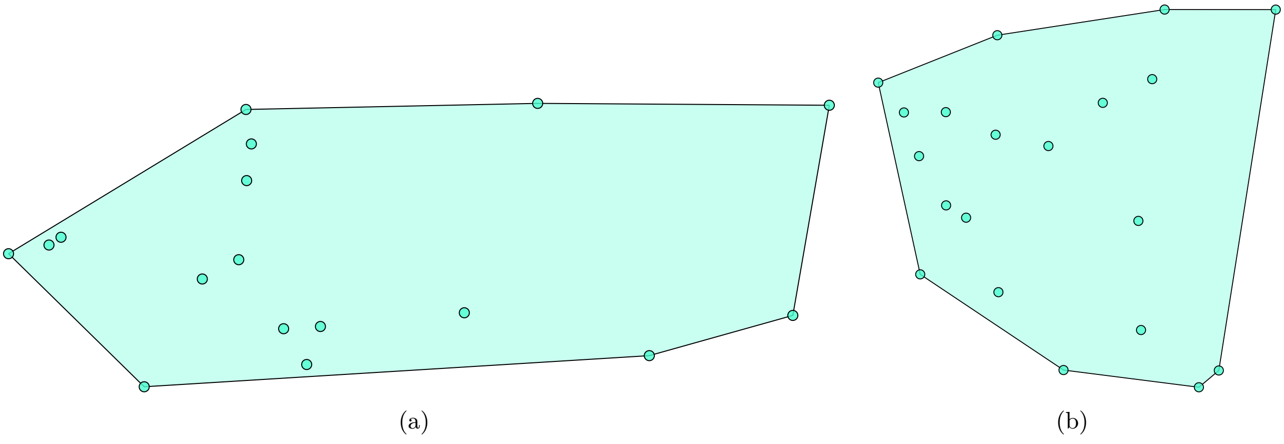


Figure 17: (a) blah (b) blah

Orthogonal Convex Hull

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

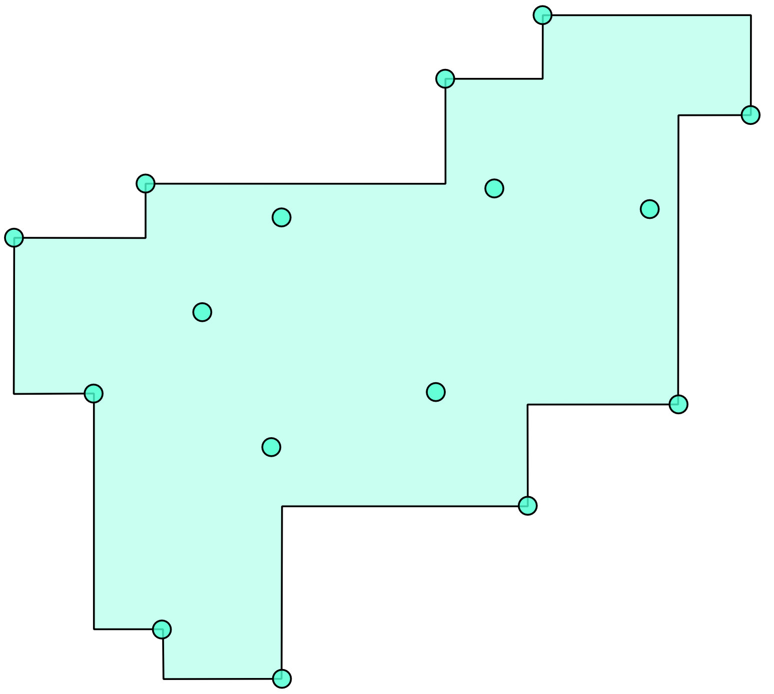
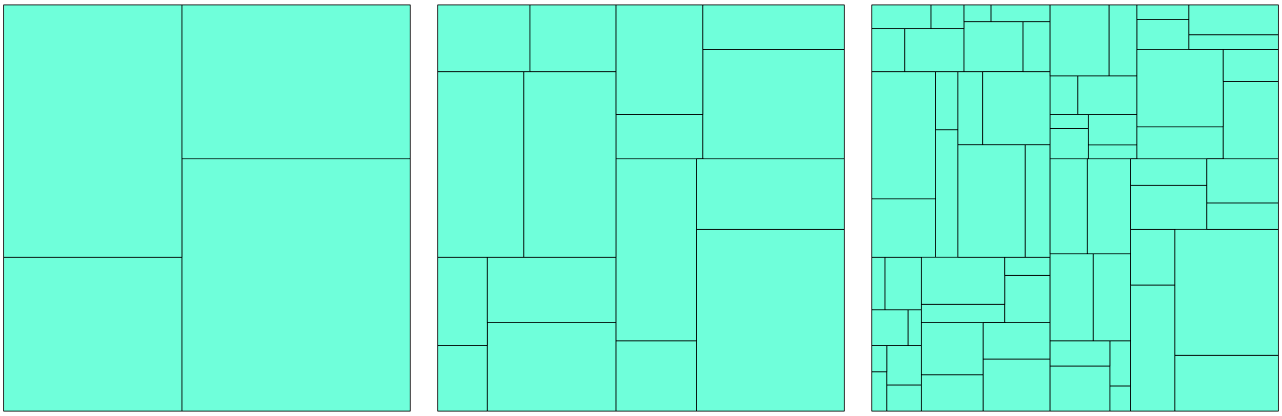


Figure 18: test

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Recursive Subdivision

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.



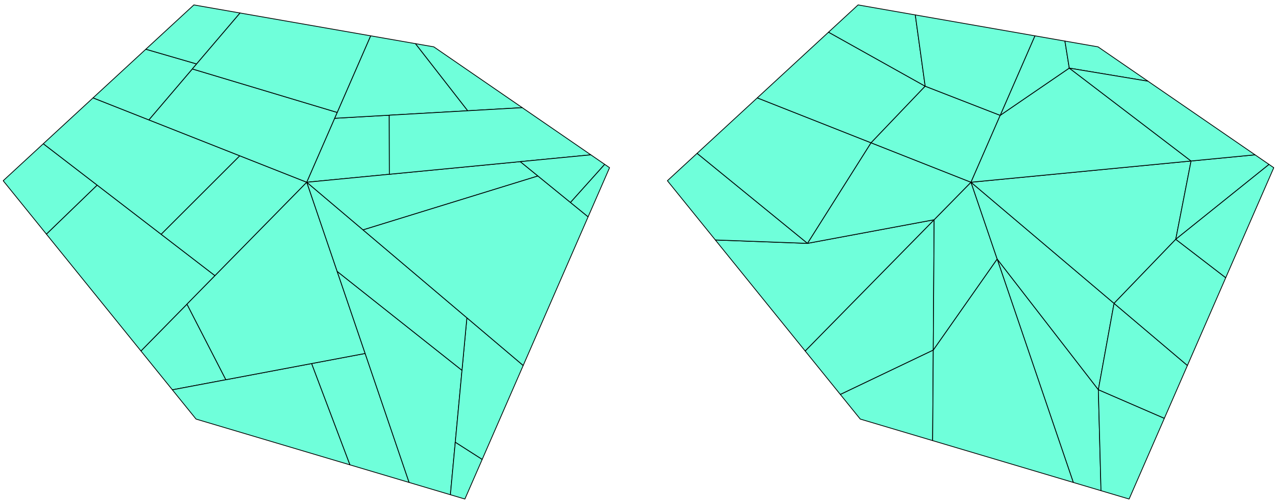
(a) (b) (c)

Figure 19: (a) blah (b) blah (c) blah

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Bent Bisection

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.



(a) (b)

Figure 20: (a) blah (b) blah

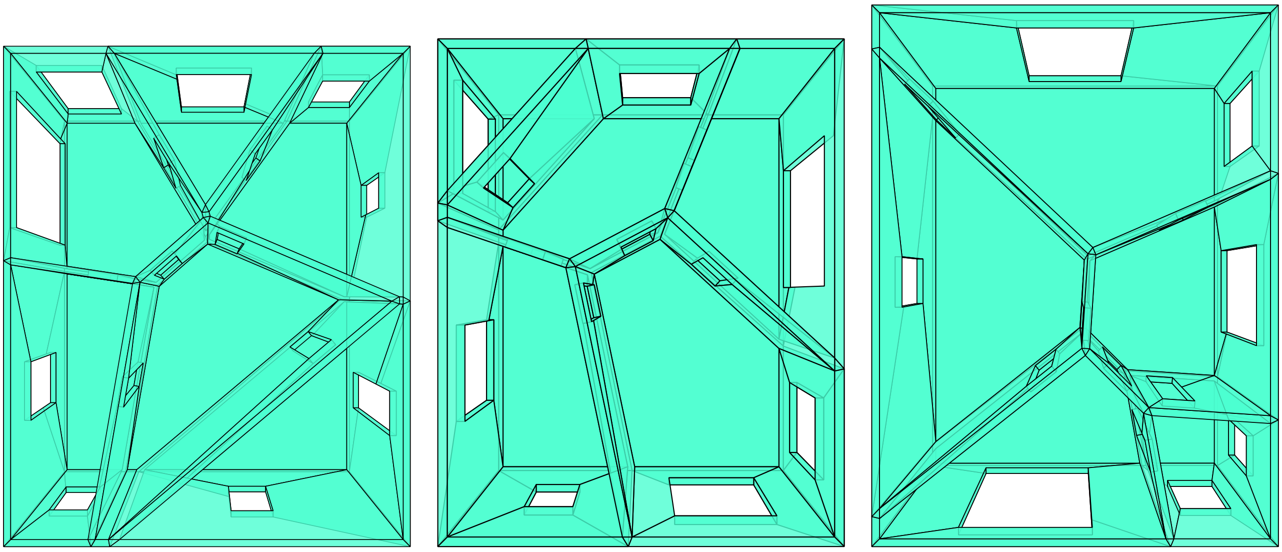
Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Shape Grammar

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Topologic

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.



(a) (b) (c)

Figure 21: (a) blah (b) blah (c) blah

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

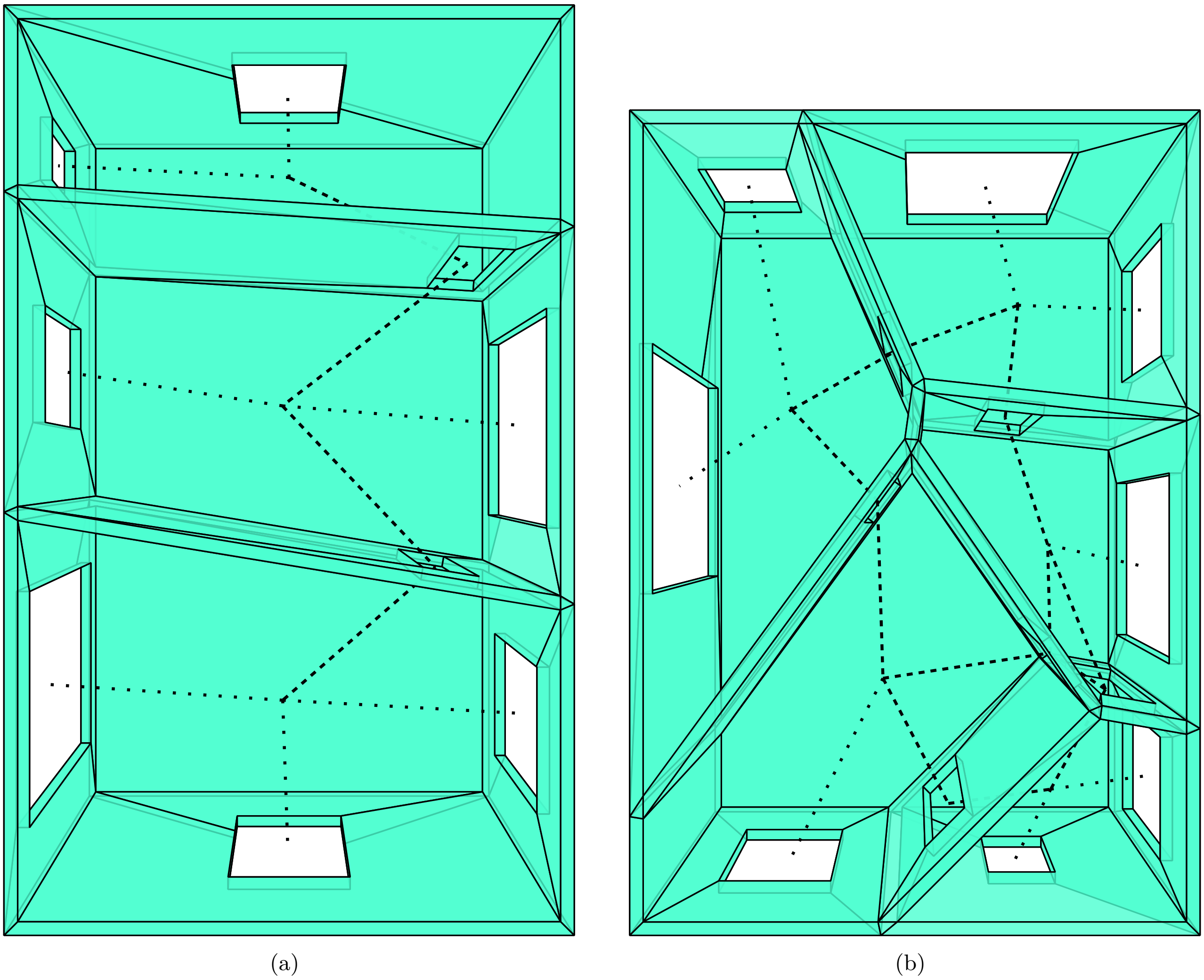
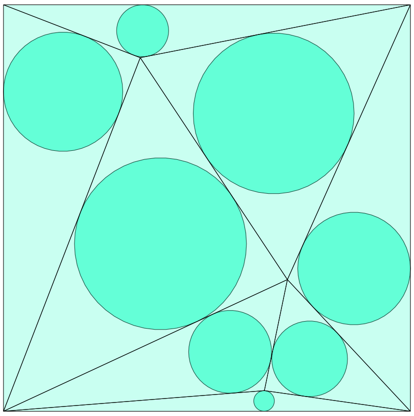


Figure 22: (a) blah (b) blah

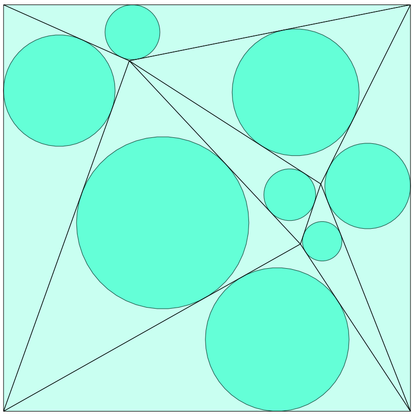
Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Shape Packing

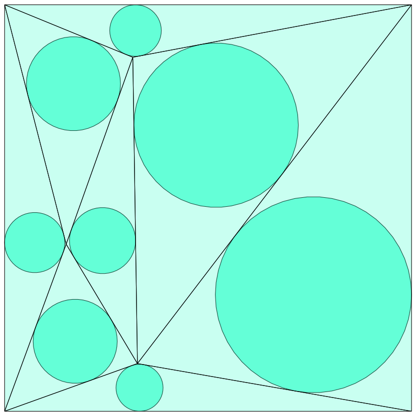
Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.



(a)



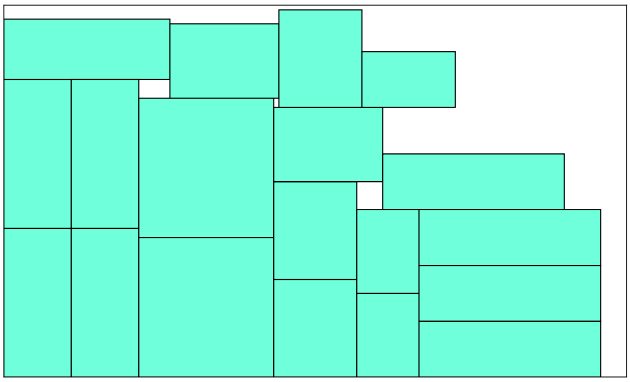
(b)



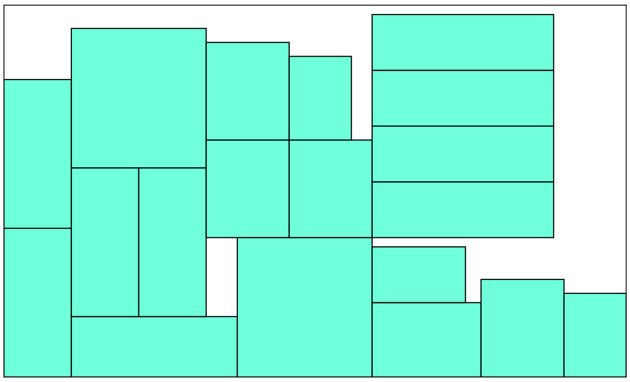
(c)

Figure 23: (a) blah (b) blah (c) blah

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.



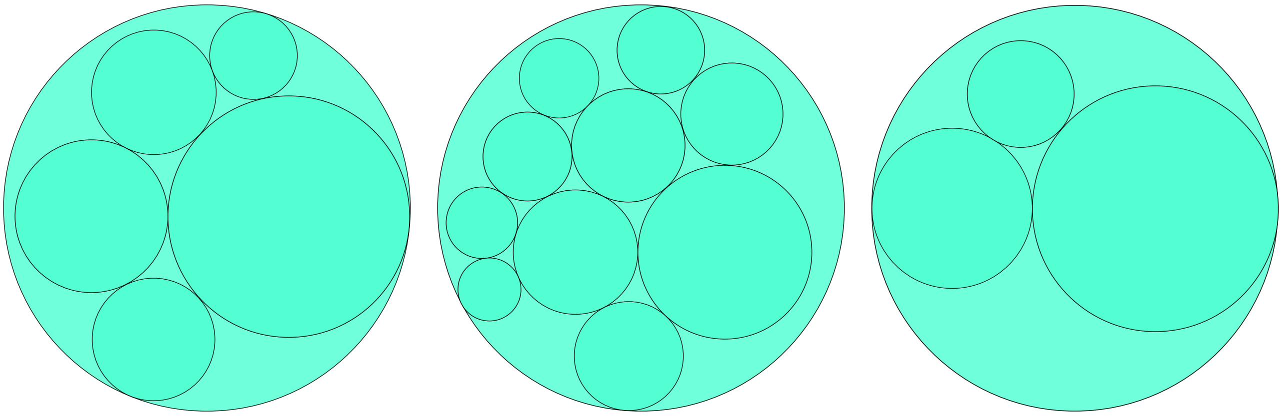
(a)



(b)

Figure 24: (a) blah (b) blah

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

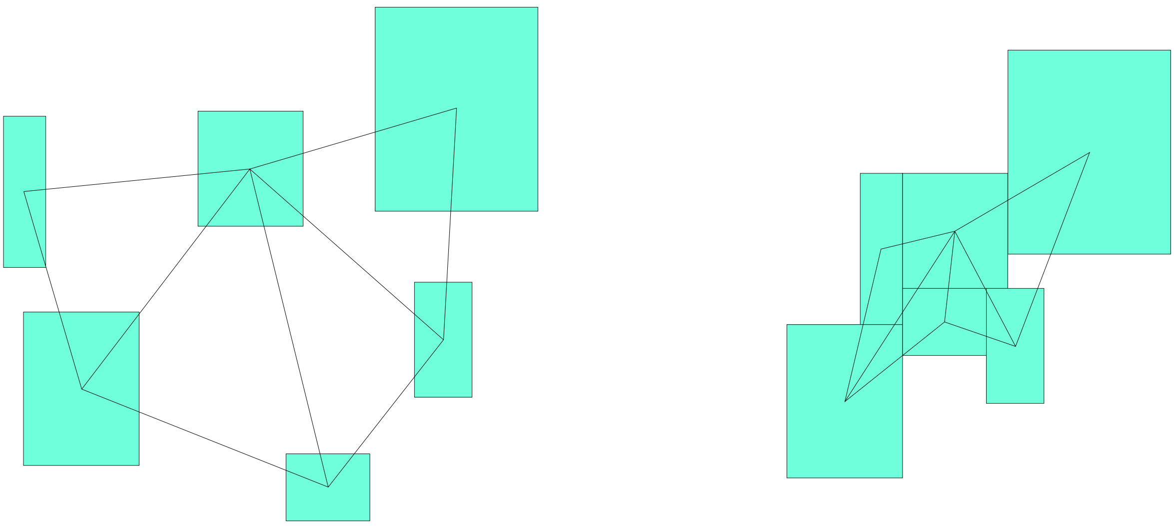


(a) (b) (c)

Figure 25: (a) blah (b) blah (c) blah

Physics Solver

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.



(a) (b)

Figure 26: (a) blah (b) blah

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

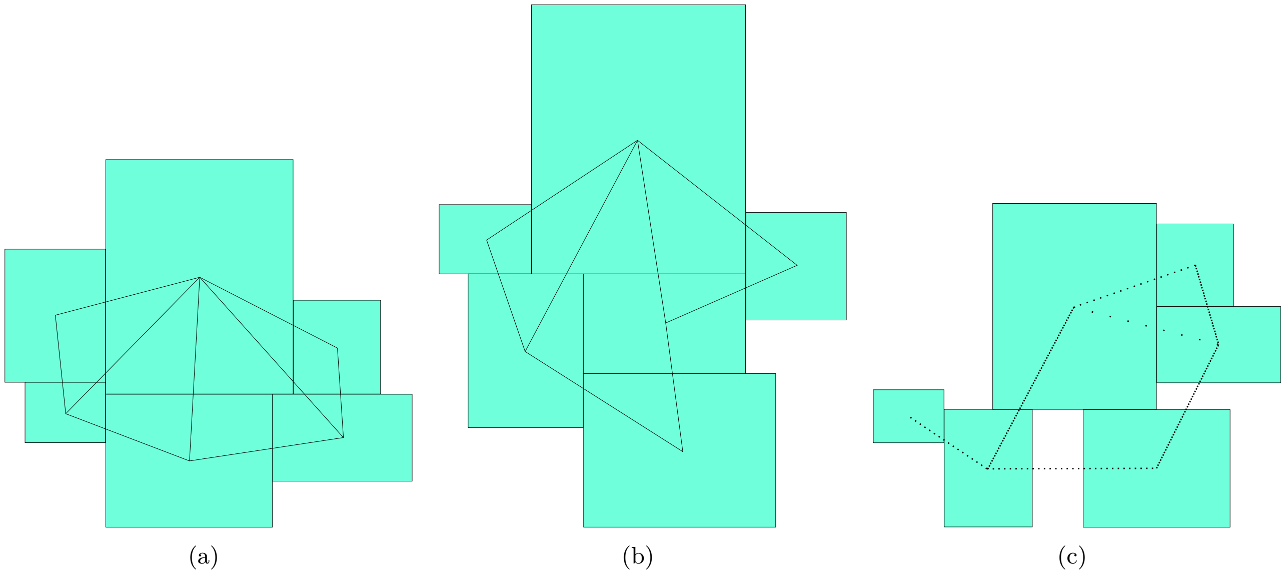


Figure 27: (a) blah (b) blah (c) blah

Physics Solver and Genetic Algorithm

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Data Driven Approaches

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Method of Choice

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Analysis

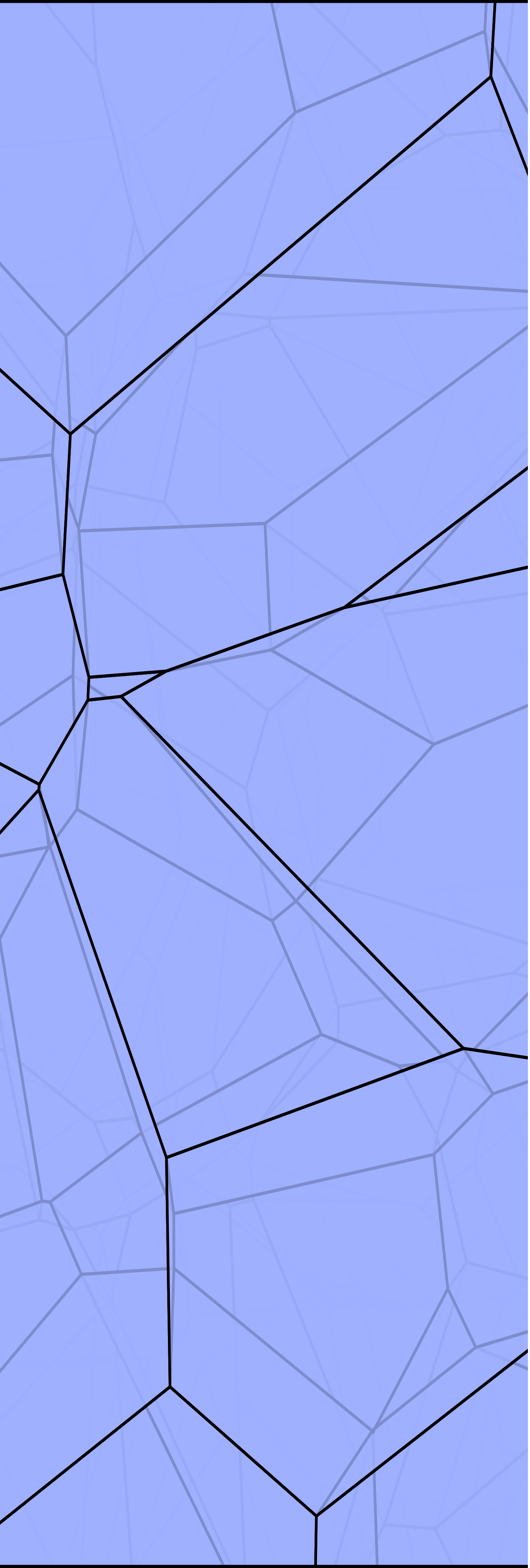
The first stage of this work deals with the automated generation of different apartment layouts using different geometry processing software. An alternative to the generation approach is the acquisition of existing architectural datasets provided by real estate and research groups. Advantages of reality-based datasets are the certainty of architectural feasibility and construction of the individual plans, but disadvantages are a conservative approach in creative terms and a high risk of traditional bias due to the predominance of local and traditionally influenced architectural methods. Therefore, this experimental phase is primarily concerned with testing different floorplan generation methods and their derivatives as well as their possible combination. Simple algorithms like voronoi diagrams, intelligent methods like evolutionary algorithms and data driven approaches like neural networks are explained and evaluated by different experiments. Criteria for the evaluation are simplicity in the construction process, computing power, time and memory, integration and interaction with used software, purity of the generated geometry, degree of readiness for the simulation stage, simplicity of the layout for data storage but above all spatial quality, creativity in form finding and feasibility. After experimentation, the most appropriate method or combination of algorithms for the following steps is selected and an appropriate set of different layouts is generated. In order to increase the variation in the synthetic dataset, results from different methods can be mixed, but they must not deviate from the generally defined quality level. Furthermore, it is a requirement to understand the functionality of the algorithms sufficiently to allow slight adaptations and combinations with other functions and to achieve variation in the individual applications. In general, simplicity is preferred to complexity, creativity to ordinariness and variation to repetition.

Topologic

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.


```
[
    {
        "brep": "topology_00001",
        "cellApertures": [],
        "cellDictionaries": [],
        "dictionary": {
            "color": [
                1.0,
                1.0,
                1.0,
                1.0
            ],
            "id": "2af15e2c-3b00cebd41f8",
            "name": "None",
            "type": "Face"
        },
        "edgeApertures": [],
        "edgeDictionaries": [],
        "faceApertures": [],
        "faceDictionaries": [],
        "vertexApertures": [],
        "vertexDictionaries": []
    }
]
```


Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

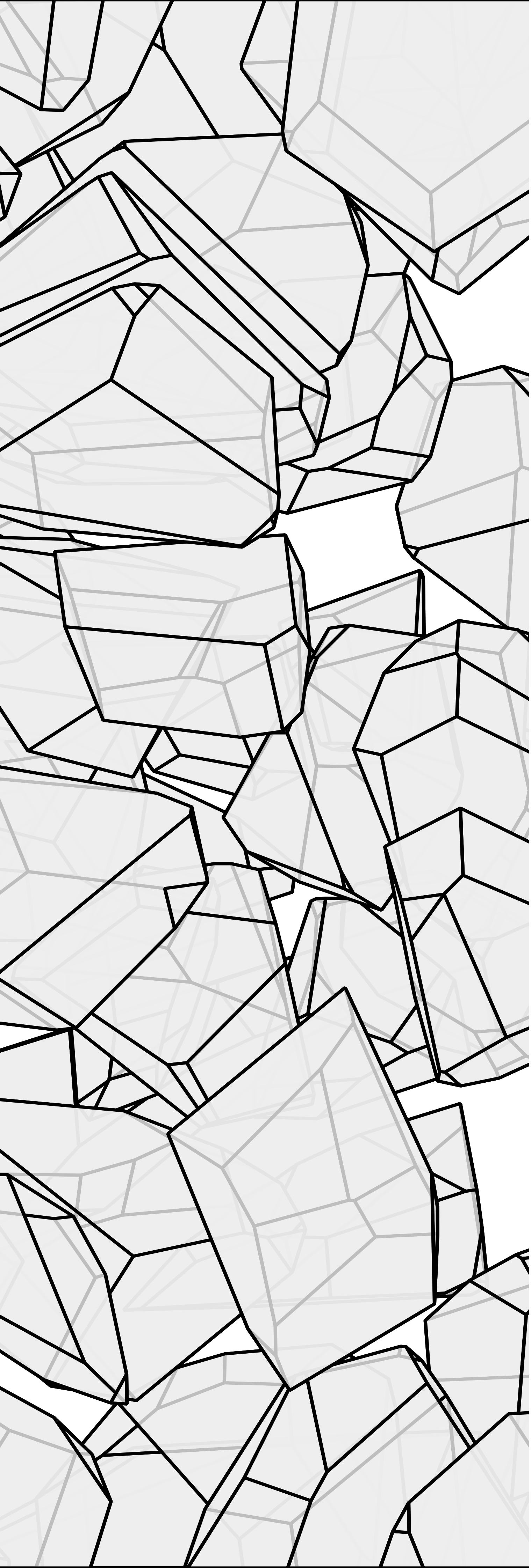


Data

The first stage of this work deals with the automated generation of different apartment layouts using different geometry processing software. An alternative to the generation approach is the acquisition of existing architectural datasets provided by real estate and research groups. Advantages of reality-based datasets are the certainty of architectural feasibility and construction of the individual plans, but disadvantages are a conservative approach in creative terms and a high risk of traditional bias due to the predominance of local and traditionally influenced architectural methods. Therefore, this experimental phase is primarily concerned with testing different floorplan generation methods and their derivatives as well as their possible combination. Simple algorithms like voronoi diagrams, intelligent methods like evolutionary algorithms and data driven approaches like neural networks are explained and evaluated by different experiments. Criteria for the evaluation are simplicity in the construction process, computing power, time and memory, integration and interaction with used software, purity of the generated geometry, degree of readiness for the simulation stage, simplicity of the layout for data storage but above all spatial quality, creativity in form finding and feasibility. After experimentation, the most appropriate method or combination of algorithms for the following steps is selected and an appropriate set of different layouts is generated. In order to increase the variation in the synthetic dataset, results from different methods can be mixed, but they must not deviate from the generally defined quality level. Furthermore, it is a requirement to understand the functionality of the algorithms sufficiently to allow slight adaptations and combinations with other functions and to achieve variation in the individual applications. In general, simplicity is preferred to complexity, creativity to ordinariness and variation to repetition.

Learning

The first stage of this work deals with the automated generation of different apartment layouts using different geometry processing software. An alternative to the generation approach is the acquisition of existing architectural datasets provided by real estate and research groups. Advantages of reality-based datasets are the certainty of architectural feasibility and construction of the individual plans, but disadvantages are a conservative approach in creative terms and a high risk of traditional bias due to the predominance of local and traditionally influenced architectural methods. Therefore, this experimental phase is primarily concerned with testing different floorplan generation methods and their derivatives as well as their possible combination. Simple algorithms like voronoi diagrams, intelligent methods like evolutionary algorithms and data driven approaches like neural networks are explained and evaluated by different experiments. Criteria for the evaluation are simplicity in the construction process, computing power, time and memory, integration and interaction with used software, purity of the generated geometry, degree of readiness for the simulation stage, simplicity of the layout for data storage but above all spatial quality, creativity in form finding and feasibility. After experimentation, the most appropriate method or combination of algorithms for the following steps is selected and an appropriate set of different layouts is generated. In order to increase the variation in the synthetic dataset, results from different methods can be mixed, but they must not deviate from the generally defined quality level. Furthermore, it is a requirement to understand the functionality of the algorithms sufficiently to allow slight adaptations and combinations with other functions and to achieve variation in the individual applications. In general, simplicity is preferred to complexity, creativity to ordinariness and variation to repetition.



Conclusion

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula. Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Discussion

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio. Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris. Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris. Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris. Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Further Readings

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris. Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Future Works

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Used Software

Geometry Manipulation

Blender (Version 3.2) [Computer Software]. (1994). Retrieved from https://www.blender.org

SciPy (Version 1.8.0) [Computer Software]. (2001). Retrieved from https://scipy.org/

Sverchok (Version 1.1.0) [Computer Software]. (2012). Retrieved from https://github.com/nortikin/sverchok

Shapely (Version 1.8.0) [Computer Software]. (2007). Retrieved from https://github.com/shapely/shapely

FreeCAD (Version 0.19.4) [Computer Software]. (2002). Retrieved from https://www.freecadweb.org

IfcOpenShell (Version 220330) [Computer Software]. (2011). Retrieved from https://blenderbim.org

Simulation

Topologic (Version 0.6.0) [Computer Software]. (2014). Retrieved from https://topologic.app

Radiance (Version 5.3) [Computer Software]. (1985). Retrieved from https://www.radiance-online.org

EnergyPlus (Version 22.1.0) [Computer Software]. (1996). Retrieved from https://energyplus.net

OpenFOAM (Version 9) [Computer Software]. (2004). Retrieved from https://openfoam.org

OpenStudio (Version 3.3.0) [Computer Software]. (2008). Retrieved from https://openstudio.net

Tools

Numpy (Version 1.22.3) [Computer Software]. (1995). Retrieved from https://numpy.org/

Pandas (Version 1.4.2) [Computer Software]. (2008). Retrieved from https://pandas.pydata.org/

LaTeX (Version 2022.02.24) [Computer Software]. (1984). Retrieved from https://www.latex-project.org

Inkscape (Version 1.1.2) [Computer Software]. (2003). Retrieved from https://inkscape.org

Bibliography

[1] Asli Agirbas. Building energy performance of complex forms-test simulation of minimal surface-based form optimization. 2020.

[2] Rita Aguiar, Carmo Cardoso, et al. Algorithmic design and analysis fusing disciplines. 2017.

[3] Feyza Nur Aksin and Semra Arslan Selçuk. Use of simulation techniques and optimization tools for daylight, energy and thermal performance-the case of office module (s) in different climates. 2021.

[4] Firas Al-Douri. The employment of digital simulation in the planning departments in us cities-how does it affect design and decision-making processes? 2018.

[5] Amer Al-Jokhadar and Wassim Jabi. Humanising the computational design process: Integrating parametric models with qualitative dimensions. 2016.

[6] Pantea Alambeigi, Canhui Chen, Jane Burry, and Eva Cheng. Shape the design with sound performance prediction: a case study for exploring the impact of early sound performance prediction on architectural design. 2017.

[7] Christopher Alexander. *A pattern language: towns, buildings, construction.* Oxford university press, 1977.

[8] Abdulrahman Alymani, Wassim Jabi, and Padraig Corcoran. Machine learning methods for clustering architectural precedents-classifying the relationship between building and ground. 2020.

[9] Jayedi Aman, Nusrat Tabassum, James Hopfenblatt, Jong Bum Kim, and MD Haque. Optimizing container housing units for informal settlements-a parametric simulation & visualization workflow for architectural resilience. 2021.

[10] Dulce Andino and Sheng-Fen Chien. Embedding garifuna shape grammars in a parametric design software. *Blucher Design Proceedings*, 1(7):202–206, 2013.

[11] Mesrop Andriasyan, Alessandra Zanelli, Gevorg Yeghikyan, Rob Asher, and Hank Haeusler. Algorithmic planning and assessment of emergency settlements and refugee camps. 2020.

[12] V Anuradha and Vennila Thirumavalavn Minal Sabnis. Voronoi diagram voro [schemata]: Application of interactive weighted voronoi diagrams as an alternate master-planning framework for business parks. 2008.

[13] Logan Armstrong, Guy Gardner, and Christina James. Evolutionary solar architecture-generative solar design through soft forms and rigid logics. 2016.

[14] Hardik Arora, Jessica Bielski, Viktor Eisenstadt, Christoph Langenhan, Christoph Ziegler, Klaus-Dieter Althoff, and Andreas Dengel. Consistency checker-an automatic constraint-based evaluator for housing spatial configurations. 2021.

[15] Imdat As and Prithwish Basu. *The Routledge Companion to Artificial Intelligence in Architecture.* Routledge, Taylor & Francis Group, 2021.

[16] Imdat As, Siddharth Pal, and Prithwish Basu. Artificial intelligence in architecture: Generating conceptual design via deep learning. *International Journal of Architectural Computing*, 16(4):306–327, 2018.

[17] Fan Bao, Dong-Ming Yan, Niloy J Mitra, and Peter Wonka. Generating and exploring good building layouts. *ACM Transactions on Graphics (TOG)*, 32(4):1–10, 2013.

[18] Catarina Belém, Luís Santos, and António Leitão. On the impact of machine learning: Architecture without architects? 2019.

[19] Martin Bielik, Sven Schneider, Florian Geddert, and Dirk Donath. Addis building configurator: Computational design tool for efficient planning of mass housing in addis ababa. 2013.

[20] Jessica Bielski, Christoph Langenhan, Babara Weyand, Markus Neuber, Viktor Eisenstadt, and Klaus-Dieter Althoff. Topological queries and analysis of school buildings based on building information modeling (bim) using parametric design tools and visual programming to develop new building typologies. 2020.

[21] Christopher Boon, Corey Griffin, Nicholas Papaefthimious, Jonah Ross, and Kip Storey. Optimizing spatial adjacencies using evolutionary parametric tools: using grasshopper and galapagos to analyze, visualize, and improve complex architectural programming. *Perkins+ Will Research Journal*, 7(2):25–37, 2015.

[22] Nathan C Brown and Caitlin T Mueller. Design variable analysis and generation for performance-based parametric modeling in architecture. *International Journal of Architectural Computing*, 17(1):36–52, 2019.

[23] Inês Caetano, Luís Santos, and António Leitão. Computational design in architecture: Defining parametric, generative, and algorithmic design. *Frontiers of Architectural Research*, 9(2):287–300, 2020.

[24] Luísa G Caldas and Luís Santos. Generation of energy-efficient patio houses with gene_arch: Combining an evolutionary generative design system with a shape grammar. 2012.

[25] Victor Calixto and Gabriela Celani. A literature review for space planning optimization using an evolutionary algorithm approach: 1992-2014. 2015.

[26] D Campo, S Manninger, M Sanche, et al. The church of ai: An examination of architecture in a posthuman design ecology [c]. In *Proceedings of the 24th International Conference on Computer-Aided Architectural Design Research in Asia: Intelligent and Informed, CAADRIA 2019*, pages 767–772. CAADRIA Hong Kong, China, 2019.

[27] Matias Del Campo, Sandra Manninger, Leete Jane Wang, and Marianne Sanche. Sensibilities of artificial intelligence. In *Design Modelling Symposium Berlin*, pages 529–538. Springer, 2019.

[28] Giuseppe Canestrino. On the influence of evolutionary algorithm (ea) optimization in architectural design: A reflection through an architectural envelope's shadowing system design. 2021.

[29] Giuseppe Canestrino, Greco Laura, Francesco Spada, and Roberta Lucente. Generating architectural plan with evolutionary multiobjective optimization algorithms: a benchmark case with an existent construction system. 2020.

[30] Silvio Carta. Self-organizing floor plans. *Harvard Data Science Review HDSR*, 2021.

[31] Stanislas Chaillou. Ai architecture towards a new approach (2019). *URL https://www. academia. edu/39599650/AI_Architecture_Towards_a_New_Approach.*

[32] H Chalabee. Performance-based architectural design: optimisation of building opening generation using generative algorithms. *Master Sustain Archit Stud Univ Sheff Sch Archit*, 2013.

[33] Ioannis Chatzikonstantinou. A 3-dimensional architectural layout generation procedure for optimization applications: Dc-rvd. 2014.

[34] Aikaterini Chatzivasileiadi, Anas M Hosney Lila, Simon Lannon, and Wassim Jabi. The effect of reducing geometry complexity on energy simulation results. 2018.

[35] Paul Coates, Christian Derix, Ing Stefan Paul Krakhofer, and AbdulMajid Karanouh. Generating architectural spatial configurations. two approaches using voronoi tessellations and particle systems. 2005.

[36] Jan Cudzik and Kacper Radziszewski. Artificial intelligence aided architectural design. 2018.

[37] Subhajit Das, Colin Day, John Hauck, John Haymaker, and Diana Davis. Space plan generator: Rapid generationn & evaluation of floor plan design options to inform decision making. 2016.

[38] Matias del Campo. Architecture, language and ai-language, attentional generative adversarial networks (attngan) and architecture design. 2021.

[39] Matias del Campo, Alexandra Carlson, and Sandra Manninger. How machines learn to plan. 2020.

[40] Gözdenur Demir. Analysis of space layout using attraction force model and quadratic assignment problem. Master's thesis, Middle East Technical University, 2014.

[41] Theodoros Dounas, Wassim Jabi, and Davide Lombardi. Topology generated non-fungible tokens: blockchain as infrastructure for a circular economy in architectural design. 2021.

[42] Gavrilov Egor, Schneider Sven, Dennemark Martin, and Koenig Reinhard. Computer-aided approach to public buildings floor plan generation. magnetizing floor plan generator. *Procedia Manufacturing*, 44:132–139, 2020.

[43] Viktor Eisenstadt, Klaus-Dieter Althoff, and Christoph Langenhan. Student graduation projects in the context of framework for ai-based support of early conceptual phases in architecture. In *LWDA*, pages 174–179, 2020.

[44] Viktor Eisenstadt, Hardik Arora, Christoph Ziegler, Jessica Bielski, Christoph Langenhan, Klaus-Dieter Althoff, and Andreas Dengel. Comparative evaluation of tensor-based data representations for deep learning methods in architecture. 2021.

[45] Viktor Eisenstadt, Hardik Arora, Christoph Ziegler, Jessica Bielski, Christoph Langenhan, Klaus-Dieter Althoff, and Andreas Dengel. Exploring optimal ways to represent topological and spatial features of building designs in deep learning methods and applications for architecture. 2021.

[46] Viktor Eisenstadt, Christoph Langenhan, and Klaus-Dieter Althoff. Generation of floor plan variations with convolutional neural networks and case-based reasoning–an approach for unsupervised adaptation of room configurations within a framework for support of early conceptual design. In *eCAADe SIGraDi Conference, Porto*, 2019.

[47] Viktor Eisenstadt, Christoph Langenhan, Klaus-Dieter Althoff, and Andreas Dengel. Improved and visually enhanced case-based retrieval of room configurations for assistance in architectural design education. In *International Conference on Case-Based Reasoning*, pages 213–228. Springer, 2020.

[48] Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau, and Christian Gagné. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, 13:2171–2175, jul 2012.

[49] E Friedrich, S Hanna, and C Derix. Emergent form from structural optimisation of the voronoi polyhedra structure. Generative Art International Conference, 2007.

[50] Ahmed Fawzy Gad. Pygad: An intuitive genetic algorithm python library, 2021.

[51] Katarzyna Grzesiak-Kopeć, Barbara Strug, and Grażyna Ślusarczyk. Evolutionary methods in house floor plan design. *Applied Sciences*, 11(17):8229, 2021.

[52] Tim Head, Manoj Kumar, Holger Nahrstaedt, Gilles Louppe, and Iaroslav Shcherbatyi. scikit-optimize/scikit-optimize, October 2021.

[53] Ruizhen Hu, Zeyu Huang, Yuhan Tang, Oliver Van Kaick, Hao Zhang, and Hui Huang. Graph2plan: Learning floorplan generation from layout graphs. *ACM Transactions on Graphics (TOG)*, 39(4):118–1, 2020.

[54] Mohamed Naeim A Ibrahim. Computing architectural layout. 2011.

[55] Tim Ireland. Emergent space diagrams: The application of swarm intelligence to the problem of automatic plan generation. 2009.

[56] Wassim Jabi. The potential of non-manifold topology in the early design stages. 2015.

[57] Wassim Jabi, Robert Aish, Simon Lannon, Aikaterini Chatzivasileiadi, and Nicholas Mario Wardhana. Topologic-a toolkit for spatial and topological modelling. 2018.

[58] Wassim Jabi, Aikaterini Chatzivasileiadi, Nicholas Mario Wardhana, Simon Lannon, and Robert Aish. The synergy of non-manifold topology and reinforcement learning for fire egress. 2019.

[59] Wassim Jabi, Barbara Grochal, and Adam Richardson. The potential of evolutionary methods in architectural design. 2013.

[60] Sam Conrad Joyce and Ibrahim Nazim. Limits to applied ml in planning and architecture-understanding and defining extents and capabilities. 2021.

[61] Ahti Kalervo, Juha Ylioinas, Markus Häikiö, Antti Karhu, and Juho Kannala. Cubicasa5k: A dataset and an improved multi-task model for floorplan image analysis. In *Scandinavian Conference on Image Analysis*, pages 28–40. Springer, 2019.

[62] Nariddh Khean, Lucas Kim, Jorge Martinez, Ben Doherty, Alessandra Fabbri, Nicole Gardner, and M Hank Haeusler. The introspection of deep neural networks-towards illuminating the black box-training architects machine learning via grasshopper definitions. 2018.

[63] Katja Knecht and Reinhard König. Generating floor plan layouts with kd trees and evolutionary algorithms. In *Generative Art Conf*, pages 238–253, 2010.

[64] Davide Lombardi, Theodoros Dounas, Lok Hang Cheung, and Wassim Jabi. Blockchain grammars for validating the design process. 2020.

[65] Yueheng Lu, Runjia Tian, Ao Li, Xiaoshi Wang, and Garcia del Castillo Lopez Jose Luis. Cubigraph5k-organizational graph generation for structured architectural floor plan dataset. 2021.

[66] Danil Nagy, Damon Lau, John Locke, Jim Stoddart, Lorenzo Villaggi, Ray Wang, Dale Zhao, and David Benjamin. Project discover: An application of generative design for architectural space planning. In *Proceedings of the Symposium on Simulation for Architecture and Urban Design*, pages 1–8, 2017.

[67] Nelson Nauata, Kai-Hung Chang, Chin-Yi Cheng, Greg Mori, and Yasutaka Furukawa. House-gan: Relational generative adversarial networks for graph-constrained house layout generation. In *European Conference on Computer Vision*, pages 162–177. Springer, 2020.

[68] Nelson Nauata, Sepidehsadat Hosseini, Kai-Hung Chang, Hang Chu, Chin-Yi Cheng, and Yasutaka Furukawa. House-gan++: Generative adversarial layout refinement networks. *arXiv preprint arXiv:2103.02574*, 2021.

[69] David Newton. Deep generative learning for the generation and analysis of architectural plans with small datasets. 2019.

[70] Maciej Nisztuk and Paweł Myszkowski. Tool for evolutionary aided architectural design. hybrid evolutionary algorithm applied to multi-objective automated floor plan generation. 2019.

[71] Maciej Nisztuk and Paweł B Myszkowski. Hybrid evolutionary algorithm applied to automated floor plan generation. *International Journal of Architectural Computing*, 17(3):260–283, 2019.

[72] Neri Oxman. Material-based design computation: Tiling behavior. 2009.

[73] Yuzhe Pan, Jin Qian, and Yingdong Hu. A preliminary study on the formation of the general layouts on the northern neighborhood community based on gaugan diversity output generator. In *The International Conference on Computational Design and Robotic Fabrication*, pages 179–188. Springer, 2020.

[74] A Papapavlou. *Structural Evolution: A genetic algorithm method to generate structurally optimal Delaunay triangulated space frames for dynamic loads.* PhD thesis, UCL (University College London), 2008.

[75] Greig Paterson, Sung Min Hong, Dejan Mumovic, and Judit Kimpian. Real-time environmental feedback at the early design stages. 2013.

[76] Bruno Postle. On pattern languages, design patterns and evolution.

[77] Mohammad Rahmani Asl, Subhajit Das, Barry Tsai, Ian Molloy, and Anthony Hauck. Energy model machine (emm)-instant building energy prediction using machine learning. 2017.

[78] Wiesław Rokicki and Ewelina Gawell. Voronoi diagrams–architectural and structural rod structure research model optimization. *MAZOWSZE Studia Regionalne*, (19):155–164, 2016.

[79] Richard Schaffranek. Parallel planning: An experimental study in spectral graph matching. In *Proceedings of the 10th International Space Syntax Symposium*, 2015.

[80] Adam Sebestyen and Jakub Tyc. Machine learning methods in energy simulations for architects and designers-the implementation of supervised machine learning in the context of the computational design process. 2020.

[81] Krishnendra Shekhawat, Nitant Upasani, Sumit Bisht, and Rahil Jain. Gplan: Computer-generated dimensioned floorplans for given adjacencies. *arXiv preprint arXiv:2008.01803*, 2020.

[82] Manav Mahan Singh, Patricia Schneider-Marin, Hannes Harter, Lang Werner, and Philipp Geyer. Applying deep learning and databases for energyefficient architectural design. In *Proceedings of the 38th International Online Conference on Education and Research in Computer Aided Architectural Design in Europe*, pages 79–87. eCAADe (Education and Research in Computer Aided Architectural Design in Europe), 2020.

[83] Kihoon Son and Kyung Hoon Hyun. A framework for multivariate data based floor plan retrieval and generation. 2021.

[84] Sherif Tarabishy, Stamatios Psarras, Marcin Kosicki, and Martha Tsigkari. Deep learning surrogate models for spatial and visual connectivity. *International Journal of Architectural Computing*, 18(1):53–66, 2020.

[85] Mahdi Valitabar, Mohammadjavad Mahdavinejad, and Peiman Pilechiha Henry Skates. Data-driven design of adaptive façades: View, glare, daylighting and energy efficiency. 2021.

[86] Yan-Chao Wang, Feng Lin, and Hock-Soon Seah. Orthogonal voronoi diagram and treemap. *arXiv preprint arXiv:1904.02348*, 2019.

[87] Nicholas Mario Wardhana, Wassim Jabi, Aikaterini Chatzivasileiadi, and Nikoleta Petrova. A spatial reasoning framework based on non-manifold topology. 2019.

[88] Shermeen Yousif and Daniel Bolojan. Deep-performance-incorporating deep learning for automating building performance simulation in generative systems. 2021.

[89] Xin Zhao, Yunsong Han, and Linhai Shen. Multi-objective optimisation of a free-form building shape to improve the solar energy utilisation potential using artificial neural networks. 2021.

[90] Hao Zheng, Keyao An, Jingxuan Wei, and Yue Ren. Apartment floor plans generation via generative adversarial networks. 2020.