

# 建築パラメトリックデザインのための対話的動線計画最適化 —平面計画、動線計画最適化手法の研究とプラグイン開発—

## 1. 背景

近年、建築設計においてコンピューテーショナルデザインが一般的になってきており、それに伴ってパラメトリックに建築の設計を行う手法も広まりつつある。特に3DCADソフト「Rhinceros」のプラグイン「Grasshopper」は、設計の様々なフェーズで使用されており、パラメトリックな意匠の設計や、構造、環境のシミュレーションで盛んに用いられている。パラメトリックに行った設計の最適化も普及しつつあり、中でも構造設計では、遺伝的アルゴリズム（GA）やその他様々な最適化手法が取り入れられている。

一方平面設計において、パラメトリックに設計を変化させたり最適化を行ったりするような取り組みはいまだ下火であり、事例も少ない。しかし、初期の平面計画や動線計画は建物の品質に直結するため最適化を研究する意義は大きいと考える。

## 2. 目的

本プロジェクトでは平面設計の新たな手法として量子アニーリングによる最適化を提案し、パラメトリックデザインの新たな可能性を広げることを目的とした。この手法によってこれまで困難であった、試行錯誤にかかる時間の短縮と機能や意匠の合理性の確保、これらを両立させながらの最適化を目指す。そのために、動線計画をアニーリングによって高速かつ対話的に最適化できるソフトウェアとしてGrasshopperのプラグインの開発を目標とした。

## 3. ソフトウェア開発内容

### ・ゾーニング最適化手法の開発

平面計画におけるゾーニングや部屋配置を最適化するための手法として二次割当問題を応用した手法を開発した（図1）。

この手法では、敷地をグリッド状に分割し、各グリッドにゾーンを割当てることによってゾーンの配置を行う。定式化には二次割当問題を応用しており、二次割当問題におけるフローとして部屋間のフローを設定し、かつゾーンがひとまとまりとなるようゾーン自身のフロー（フロー行列における主対角線上の要素）の値を大きく設定した。

また、二次割当問題以外にもゾーンの矩形化、ゾーンの方角指定のための目的関数や、ゾーンの重複を禁止したり、ゾーンの面積を所定の範囲に収めたりするための制約条件を与えている。それぞれ、ゾーンの矩形化では、ゾーン間の境界線の長さを最小化する目的関数を、ゾーンの方角指定では各グリッドの指定した方角からの距離（座標）を最小化する目的関数を作成した。また、制約条件として、ゾーンの重複禁止では各グリッドに割り当てられるゾーンが一つになるような条件の定式化、ゾー

ン面積の制約ではそれぞれのゾーンの面積と指定した面積との差が許容誤差範囲内に収まるような定式化を行った。

それぞれの目的関数や制約条件に適切な正規化、係数の設定を行い二次割当問題による目的関数に足し合わせて理論模型を作成、最適化を行う。これらによって多様な条件でのゾーニングの最適化を可能とした。

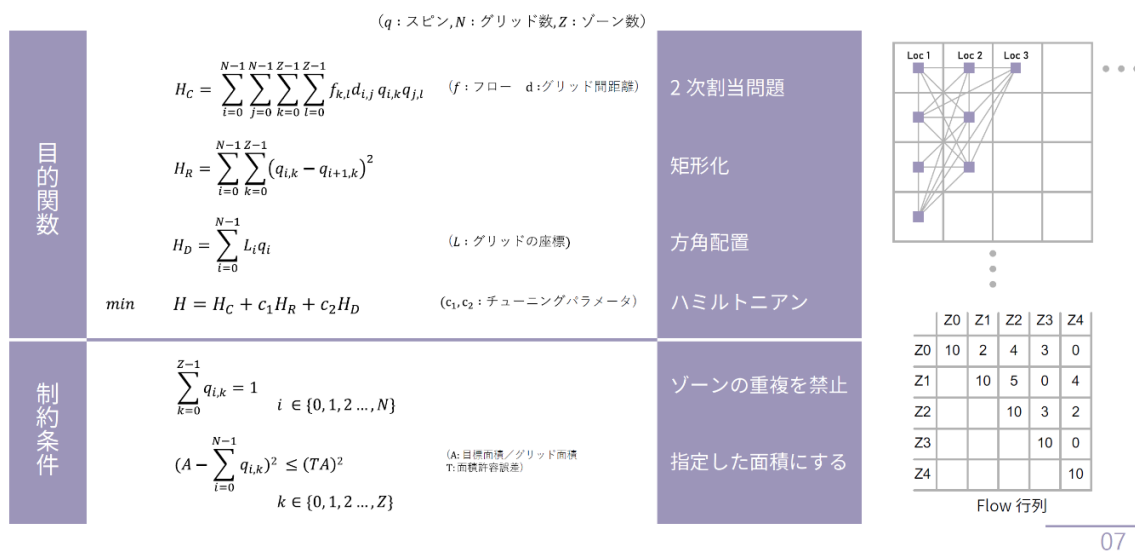


図 1: ゾーニング最適化問題の定式化 (成果報告会プレゼン資料より引用)

・ 通路配置最適化手法の開発

動線計画における通路の配置を最適化するための手段としてシュタイナー木問題 2 を応用した手法を開発した (図 2)。

この手法では、ゾーニングの最適化で得られた結果を元として、その境界線の頂点とユーザーが設定するドアとなる点から無向グラフを作成する。その無効グラフをつなぐ辺を通路とすることで、通路の配置をおこなう。ドアとなるポイントを必ず辺でつなぐ必要のある頂点、ゾーン境界線の頂点を必ずしも辺で繋ぐ必要のない頂点とし、シュタイナー木問題の定式化を行った。

グリッドを元に最適化を行うため、隣接行列を用いることによってグリッドに沿った通路の最適化を可能としている。

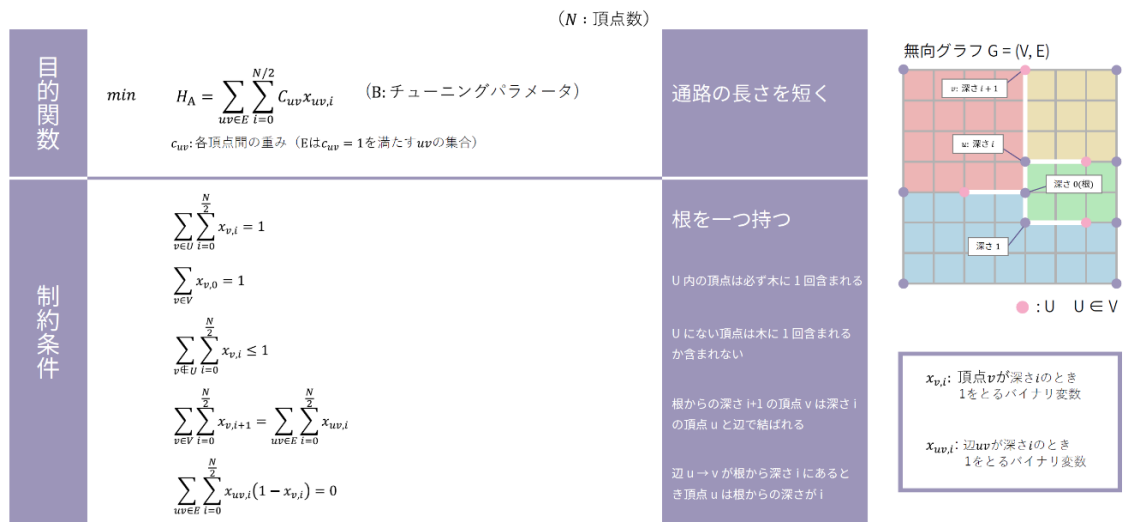


図 2 : 通路配置最適化問題の定式化 (成果報告会プレゼン資料より引用)

・ Grasshopper プラグイン開発及び公開

開発した最適化手法をもとに、Grasshopper 上で最適化を行うためのプラグイン開発を行った。このプラグインは、最適化を行う 2 つのコンポーネント (Zoning Optimizer、Corridor Layout Optimizer) とそのコンポーネントの入力や出力を補助する 5 つのコンポーネント (Site Grid、Result2Tree、Corridor Point、Zoning Optimizer(URL)、Corridor Layout Optimizer(URL))によって構成されている(図 3)。最適化を行うコンポーネントは、Grasshopper の外部での処理を可能とするプラグイン「Hops」と、処理を行うサーバーによって作成し、補助用のコンポーネントは Grasshopper 上のプログラムを配布するための機能「User Object」によって作成した。ユーザーはこれらのコンポーネントに適切な入力を行うことによって平面計画や動線計画の最適化を行い、パラメータを変化させることによって試行錯誤する。

また、Rhinceros/Grasshopper のプラグイン公開サイト「Food4Rhino」にて、作成した Grasshopper プラグインを「Qalico」として公開した (<https://www.food4rhino.com/en/app/qalico>)。

プラグイン公開に伴い、プラグインのアイコンの作成を吉成愛美理氏 (京都工芸繊維大学学生) に依頼。また、説明文やサンプルプログラムの作成を行い、ユーザーに抵抗なくプラグインの機能を使用してもらうための環境を整えた。

また、コンピューテーショナルデザインにおける量子アニーリング技術の普及のため、GitHub にプラグインのコード、バイナリを公開した。

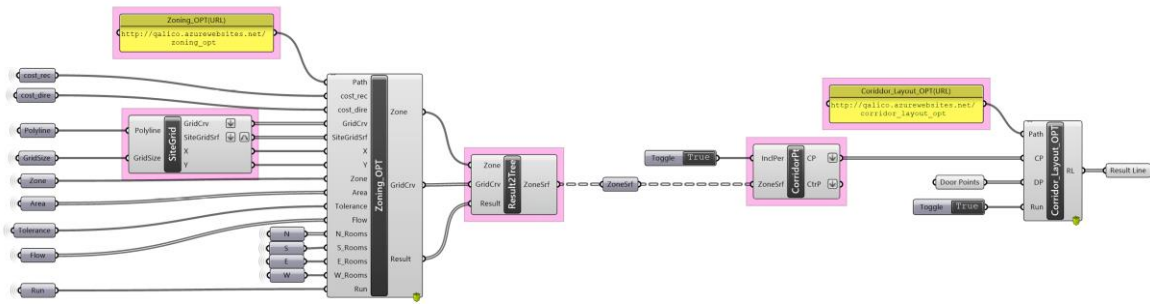


図 3：プラグインコンポーネントの使用例

#### 4. 新規性・優位性

従来の平面計画最適化や、動線計画最適化においては、手続き的な部屋の配置や制約を満たす部屋配置を列挙しそれを設計者が評価していくもの、グラフ理論を用いた部屋や動線の配置などが見られた。しかし、これらの手法では部屋の形状に多様性がない点や、最適化時間の短縮が難しい点などいくつかの問題点があった。

本プロジェクトで開発した最適化手法は、グリッドをもととした組み合わせ最適化を用いることによって形状に多様性を持たすことができ、またアニーリングマシンによる高速な最適化を実現した。

さらに、Grasshopper にプラグインとして実装することで導入を容易にし、かつ最適化結果の活用の幅を広げることが出来た。

#### 5. 期待されるユーザー価値と社会へのインパクト

本プロジェクトで作成したソフトウェアは、設計者や設計者をコンピューテーショナルデザインによって技術的にサポートする人、また設計を学ぶ学生をユーザーとして想定している。こうしたユーザーは平面設計に精通しており、ある程度平面計画を素早く構成することが可能である。そうしたユーザーに価値を提供するため、このソフトウェアでは最適化にかかる時間をなるべく短く抑えることが重要となる。そこで、アニーリングを用いることによって、大規模な建築の条件であってもユーザーが考えるよりも早く結果を出力できるものとした。これによって、素早く平面計画を得ることが出来るだけでなく、出力結果を評価し、その評価を入力条件へと反映させることで更に設計を改良していくことが可能となる。具体的には、ゾーニングについては小さな規模であれば 25 秒程度、大きな規模でも 1 分半ほどで最適化でき、通路配置についても規模に関わらず、適切な条件であれば 20 秒程度で最適化が可能となっている。Grasshopper で従来用いられている GA であれば、問題は異なるものの最適化パラメータがより少ない条件でも 10 分程度最適化を行う必要があるため、飛躍的な高速化が行えていることがわかる。

また、Grasshopper はユーザーがビジュアルコーディングを行う事によって自由に機能を付け足していくことが可能なため、本プロジェクトのソフトウェアで得られた結果を元に更にそこから立体化や形状の変化を行うことが可能である。

こうした量子アニーリングによる最適化が普及すれば、現状他のアルゴリズムが用いられている構造や環境の最適化への応用や、建設業での作業の効率化など、幅広い分野での応用が期待できる。

## 6. 氏名（所属）

田川 直樹（京都工芸繊維大学工芸科学部デザイン・建築学課程 3 年）

（参考）関連 URL

・ Food4Rhino 「Qalico」

<https://www.food4rhino.com/en/app/qalico>

・ GitHub 「Qalico\_server」

[https://github.com/TagawaN/Qalico\\_server/tree/master/Qalico\\_server\\_v011](https://github.com/TagawaN/Qalico_server/tree/master/Qalico_server_v011)