

イジングマシンを用いた救助経路の最適化に関する一検討

長南和希^{†1} 小松一彦^{†2} 佐藤雅之^{†3} 小林広明^{†3}

東北大学工学部機械知能航空工学科^{†1} 東北大学サイバーサイエンスセンター^{†2}
東北大学情報科学研究科^{†3}

1 はじめに

津波災害時の救出計画は、救助者の分布や移動可能経路、ボートなどの救出資源の数や配置など様々な情報を総合的に考慮し、最適な資源割り当てを行う必要があるため、人の力だけで膨大な情報の組み合わせから最適な解を導くのは困難である。

そこで本研究では、効果的な救助計画の立案を目的とし、イジングマシンを用いて津波災害時の救出経路の最適化を行う。イジングマシンにより求めた解が制約を満たす割合や経路距離の評価を通じて、資源割り当ての最適性を検討する。

2 QUBO 形式による問題の定式化

本研究では津波災害時に浸水した道路をボートで移動し、分散する救助対象者を救助することを課題として設定する。各ボートが配備位置から救助者の位置まで向かった後、安全な避難所まで救助者を連れていくまでの経路選択を組み合わせ最適化問題として定式化し、イジングマシンを用いて最適化を行う。ここでボート数と配備位置、避難者の分布と人数を初期条件として与えられるものとする。

組み合わせ最適化問題をイジングマシンで解くためには、問題を式(1)で表される QUBO (Quadratic Unconstrained Binary Optimization) 形式で定式化する必要がある。

$$E(q) = \sum_{i,j} Q_{i,j} q_i q_j + \sum_i b_i q_i \quad (1)$$

ここで、 q_i はバイナリ変数であり、 $Q_{i,j}$ は QUBO 行列と呼ばれ組み合わせのコストを表現する。イジングマシンは式(1)で示されるエネルギー関数 $E(q)$ を最小化するような解を探索する。

3 ボート配備と経路の最適化

ボート配備・経路の最適化において、救助者の人数に応じた適切な数のボート分配や、各ボートの移動距離がなるべく短くなるように、両者を考慮して最適化を行う必要がある。また、浸水した道路を機動性が低いボートで移動する状況を想定しているため、なるべく道は被らないようにして複数ボートの同一経路上での混雑を避け、経路移動時間を最小化する。そこで本研究では、救出経路の最適化を行うにあたって次に示す3つの最適化を行う。

3.1 ボート分配の最適化

ボート分配の最適化の方針として、避難者分布の各位置で全員が救助されるまでに必要なボートの往復回数の最大値を最小化する。計算方法としては、二分探索を用いて達成可能な往復回数を探索する。計算量は救助者の分布位置数を N 、探索上限を M として $O(N \log M)$ となる。

3.2 ボート行先の最適化

ボート行先の最適化方針として、各ボートの移動距離の総和を最小化する行先を指定する。このため以下のように QUBO 形式に定式化し、イジングマシンを用いて解を探索することで各ボートの行先を決める。

$$E(x) = \sum_i C_{i,i} x_i + \lambda \sum_{u \in S} \left(\sum_{k=1}^K x_{u,k} - f(u) \right)^2 + \lambda \sum_{v \in G} \left(\sum_{k=1}^K x_{v,k} - g(v) \right)^2 \quad (2)$$

式(2)の $C_{i,i}$ は経路 i の距離を表現し、関数 $f(u), g(v)$ は始点 u から出る経路数、終点 v に向かう経路数の制約をそれぞれ表す。関数 $f(u), g(v)$ は一つの場所に複数のボートが配備されている場合や、救助者の人数に応じたボート数分配に対応する。式(2)を最小化する組み合わせをイジングマシンに解かせることで、ボート行先の最適化方針に従った解が得られる。

3.3 渋滞解消のための最適化

経路選択の最適化方針として、各ボートはなるべく同じ道路をたどらないようにしつつ、短い距離で目的地に

A Study on Optimization of Rescue Routes Using an Ising Machine

Kazuki Chonan^{†1}, Kazuhiko Komatsu^{†2}, Masayuki Sato^{†3}, Hiroaki Kobayashi^{†3}

Department of Mechanical and Aerospace Engineering, Tohoku University^{†1}

Cyberscience Center, Tohoku University^{†2}

Graduate School of Information Sciences, Tohoku University^{†3}

向かうことを考える。まず、始点 u -終点 v の各組み合わせにおいて、経路候補として距離の短い経路を複数用意する。次に式 (3) に示す定式化を行い、イジングマシンを用いて各 (u, v) の経路候補から他の経路と同じ道を通らないようなものを1つずつ決定する [1]。

$$E(x) = \sum_e \left(\sum_m C_{e,m} x_m \right)^2 + \lambda \sum_{u=1}^N \left(\sum_{k=1}^K x_{u,k} - 1 \right)^2 \quad (3)$$

ここで、式 (3) の第1項は道路 e を通る経路の重複に対するコストを表し、第2項は各 (u, v) に与えられた経路候補から1つ選ぶようにするための制約である。式 (3) のエネルギー関数 $E(x)$ を最小化するような解をイジングマシンを使って見つけることで渋滞解消の最適化方針に従った経路が得られる。

4 評価

4.1 評価環境

イジングマシンとして Fixstars Amplify AE を用いた。検証を行う対象地域は、南海トラフ地震によって津波災害が想定される高知県高知市とし、道路データは Open street map から Python ライブラリを用いて取得した。初期条件として与えるポート数は100台、救助者の分布数は50箇所、人数はランダムに与え、避難所の数は20箇所とした。地図上のノードからランダムに選んで、ポート、救助者および避難所の分布位置を決定した。

また最適化のための定式化を評価するために、イジングマシンを使用しない貪欲的なアルゴリズムを用いて同様にポート行先を指定し、得られた結果をイジングマシンの結果と比較する。貪欲的なアルゴリズムは、まだ行先が決まっていないポートのうち、最も距離が短いものを採用する。これによってポート数分配の制約を満たしつつ、ある程度距離が短い行先を指定することができる。

4.2 評価結果と考察

まず、得られた解が制約を満たしているかについて評価を行った。3.2節および3.3節の定式化に基づき QUBO を作成しイジングマシンを用いて最適化を行ったところ、得られた解は与えた制約をすべて満たすことを確認できた。

次にポート行先の最適化について、ポートの移動距離の評価を行った。得られた解に基づき、各ポートから救助者までの距離の総和を計算すると 56,966m となった。一方、貪欲的なアルゴリズムで得られた経路の距離の総和は 66,274m であった。このことから、イジングマシンを用いた結果は各ポートの移動距離が短い行先を指定できていると分かる。

最後に渋滞解消の最適化について、渋滞解消ありと

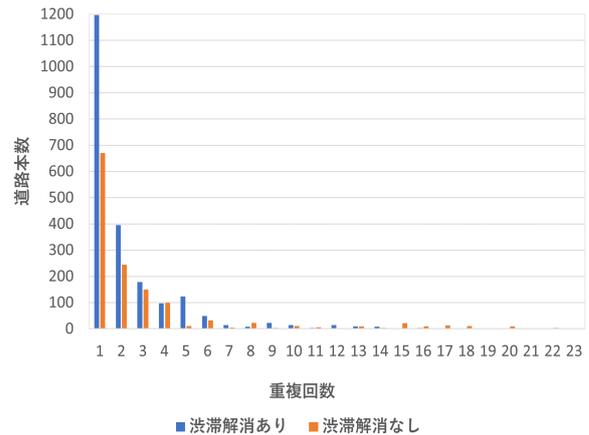


図1: ポートの経路が重複した道路の本数と重複回数の関係

なしの結果を比較した。図1に、渋滞解消の有無における重複回数と道路本数の関係を示す。これを見ると、渋滞解消を行った場合、重複数が1,2の道路が増加し、重複数15以上の道路が減少した。次に総移動距離については、渋滞解消ありのとき 244,739m、なしのとき 244,466m となった。したがって、渋滞解消の最適化を行うことで移動距離の増加を抑えつつ経路の重複を削減できた。渋滞は移動時間の著しい増加を招く恐れがあり、イジングマシンの方法が移動時間短縮に大きな効果得られることが期待できる。移動時間の検証は今後の課題である。

5 おわりに

本研究ではイジングマシンを用いて津波災害時の救助ポートの配備と移動経路の最適化を行った。3つの最適化方針をもとに定式化を行い解を求めたところ、ポート資源の適切な配置と救助先までの移動距離が短い経路を得ることができた。また、渋滞解消を行うことで重複する道路が少なくなることが確認できた。今後の課題として、移動時間の検証に加え、ポートや救助者の各分布の指定をランダムでなく、具体的なデータを用いて検証を行うことが挙げられる。

謝辞

本研究の一部は SIP3 期事業「量子および疑似量子アニーリングによる災害対応最適化問題の解決」の支援を受けて行われた。

参考文献

- [1] Yihui Liu, Kazuhiko Komatsu, Masahito Kumagai, Masayuki Sato, and Hiroaki Kobayashi. Performance evaluation of tsunami evacuation route planning on multiple annealing machines. In *Proceedings of the 20th ACM International Conference on Computing Frontiers*, CF '23, p. 185–188, New York, NY, USA, 2023. Association for Computing Machinery.