

量子アニーラーへの埋め込みに優れた QUBO 定式化手法の開発 —疎結合 QUBO による大規模・高精度量子アニーリングの実現—

1. 背景

量子アニーリングは、重ね合わせやトンネル効果などの量子効果を利用して組合せ最適化問題を効率的に解く手法として期待されている。しかし、実際の量子アニーラーの実用化にはハードウェア上の大きな制約が存在する。最新のマシンでも qubit 数は 5000 個程度であり、かつ物理 qubit 間の接続が限定的である。

量子アニーラーで組合せ最適化問題を解く際には、論理変数を物理 qubit へマッピングする埋め込みが必要となる。このとき、物理 qubit 間の接続が疎であるため、QUBO の論理変数間の接続が密な場合は直接埋め込むことができない。このような場合は物理 qubit の接続グラフ上で隣接した複数の qubit を 1 つの論理変数とみなしてマッピングを行う。このような複数の qubit を chain と呼ぶ。QUBO が密結合であるほど埋め込みは難しく、chain は長くなる傾向にある。このように chain が長くなると、埋め込み時に必要な物理 qubit 数が増加し、qubit 数に限りのある現在の量子アニーラーでは実行できる問題サイズに制限をかけてしまう。また、chain を構成する複数の物理 qubit が同じ値をとるように維持したり、結果として違う値となってしまう chain break によって、解の精度が劣化してしまうといった問題がある。

等式制約 $\sum x_i = c$ を QUBO として表現する際、既存手法ではペナルティ項 $H_{\text{penalty}} = (\sum x_i - c)^2$ として定式化されるため、結合数のオーダーが $O(N^2)$ と、非常に密結合な QUBO として定式化されていた。実際、TSP などの制約条件が絡む問題では、このような制約項由来の結合数が QUBO 全体の結合数を大きく押し上げ、密結合にできてしまっていた。そのため、上述した埋め込みの際の課題を引き起こし、実用化を進めるにあたって大きな壁となっていた。

2. 目的

本プロジェクトの目的は、上記の課題を制約条件の QUBO 定式化アルゴリズムの工夫で解決することである。具体的には、量子アニーラーへの埋め込みに特化するために、N-hot の等式・不等式制約条件の QUBO モデルの結合数を削減することを目指す。これにより、埋め込み時の chain 長を削減し、より大規模・複雑な問題の実行と解の精度向上といった効果を狙う。そして、現在実用化が発展途上である量子アニーラーの性能を最大限に引き出していく。

3. ソフトウェア開発内容

本プロジェクトでは、密結合の根本原因である長い式の二乗を回避するため、補助変数を追加して制約条件を分解して構築する新たなアルゴリズムを開発した。図 1 では、制約条件 $\sum_{i=1}^4 x_i = 1$ をネットワーク構造に従って分解していく様子を示す。このように、one-hot 制約については非常に簡潔な実装で分解できる。一般の N-hot 制約については、図 2 に示すように分割統治法を用いて再帰的に分解している。これら

を用いることで、効率的に制約条件を分解し、疎結合な QUBO として定式化することが可能になった。

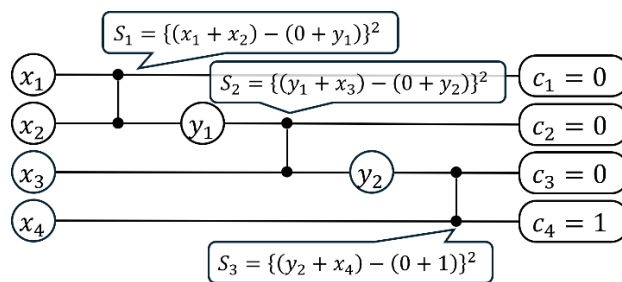


図 1 one-hot 制約の分解の様子

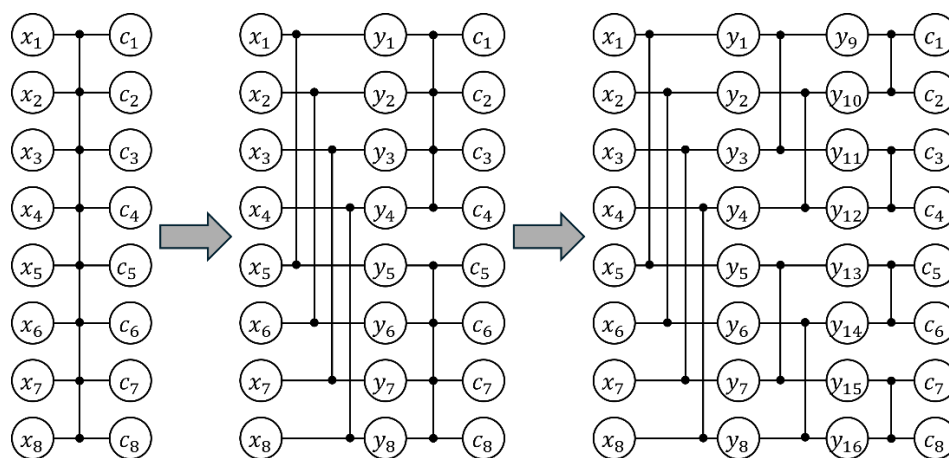


図 2 一般の N-hot 制約の分解の様子

これらの成果物を OSS ライブラリ sparse-qubo として公開した。実機への接続を念頭に設計されたライブラリである D-Wave 社の dimod と Fixstars Amplify 社の Amplify SDK に対応した。図 3 に示すように、ユーザーの既存コードをなるべく変更せずに済む設計にすることで、移行コストを最小限に抑えた。

また、内部で用いるネットワークの構築アルゴリズムをユーザーが自由に設計できるようにした。このとき、アルゴリズム設計を支援するためのビジュアルライザを実装した。図 4 に示すように、元の変数と補助変数、分解後の制約条件の関係がグラフで図示され、QUBO に変換したときの結合と色を対応させた。

```

variables = dimod.variables.Variables(
    ... [f"x_{i}_{j}" for i in range(N) for j in range(N)]
)
bqm = dimod.BinaryQuadraticModel(dimod.BINARY)
for row in range(N):
    ... row_vars = variables[row * N : (row + 1) * N]
    ... bqm += create_constraint_dwave(row_vars, ConstraintType.EQUAL_TO, c1=c)
for col in range(N):
    ... col_vars = variables[col:N]
    ... bqm += create_constraint_dwave(col_vars, ConstraintType.EQUAL_TO, c1=c)
    
```

この行を
書き換える
だけでOK

図 3 sparse-qubo の使い方の例

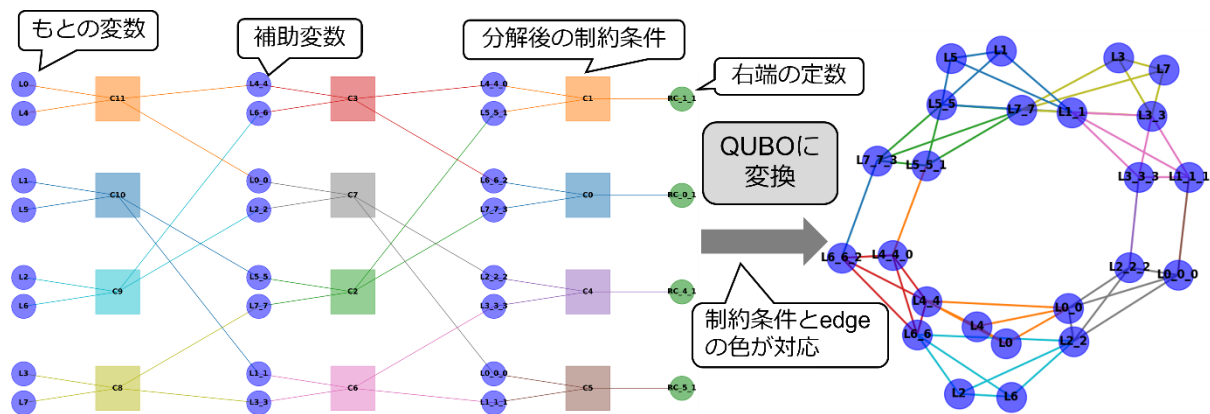


図 4 アルゴリズム設計用のビジュアライザ

4. 新規性・優位性

既存の等式制約の定式化では QUBO の結合数が $O(N^2)$ となるのに対し、本手法では結合数を $O(N \log N)$ (one-hot 制約に特化すれば $O(N)$) へと大幅に削減した。これにより、埋め込み時に必要な物理 qubit 数や平均 chain 長を大幅に削減でき、より大規模・複雑な問題を高精度に求解することが可能になった。実用的なシフトスケジューリング問題を用いた比較実験でも、本手法を用いることで安定活高品質な解が求まることを確認した。

5. 期待されるユーザー価値と社会へのインパクト

量子アニーリングはハードウェア（量子ビット数や結合トポロジー）の進化を待たなければ実用的な規模の問題が解けないとされてきた。本手法は、ソフトウェア側の定式化工夫のみでこの壁を突破し、「”今”使える技術」へと昇華させるものである。想定ユーザーとしては、組合せ最適化技術を活用したい企業の研究者やエンジニア、量子アニーリングのアプリケーション開発者が挙げられる。配送計画、シフト最適化、ポートフォリオ最適化など、制約条件が多数存在する実社会のあらゆる問題を定式化の際に、既存ライブラリの一部として最小コストで利用できる。

社会へのインパクトとしては、これまで問題規模が大きすぎて実機に乗らなかったりノイズでまともな解が出なかったりして、量子アニーリングの適用が諦められていた課題が、本手法を用いることで現行のハードウェアのままでも高精度に解けるようになる。これは量子アニーリングの社会実装を加速させる効果を持つ。また、このような実機を用いた量子アニーリングの普及が進むことで、さらなる量子アニーリングの研究開発が進められることも予想できる。このような相乗効果により、古典コンピュータでは到達できない性能による問題解決が現実になると考える。

6. 氏名（所属）

須田 浩平（東京大学 大学院情報理工学系研究科 電子情報学専攻）

2025 年度未踏ターゲット事業(量子コンピューティング技術を活用したソフトウェア開発分野)

(参考) 関連 URL

- arXiv: <https://arxiv.org/abs/2601.18108>
- OSS ライブラリ sparse-qubo: <https://github.com/KoheiSuda/sparse-qubo>