SWAQ

特定の問題にちょっと強くなった量子アニーリングシミュレーター

岡田颯斗

大阪府立四條畷高等学校

目次

自己紹介

どうもこんにちは

- 名前：岡田颯斗（高校 3 年生）
- 趣味・興味：
  - 競技数学
  - 量子コンピュータ

前提

量子アニーリングは組合せ最適化問題を解く手段の一つ

よく扱われる問題の例：

彩色問題

- 隣り合う場所は異なる色で塗分ける

巡回セールスマン問題

- 複数の街を最短経路ですべて訪れる

彩色問題

彩色問題：リンク

だがしかし

量子アニーリングは組合せ最適化問題を解く手段の一つ

彩色問題

- 隣り合う場所は異なる色で塗分ける

巡回セールスマン問題

- 複数の街を最短経路ですべて訪れる

だがしかし

量子アニーリングは組合せ最適化問題を解く手段の一つ

彩色問題

- 隣り合う場所は異なる色で塗分ける

巡回セールスマン問題

- 複数の街を最短経路ですべて訪れる

しかし、量子アニーリングは最適化問題を効率よく解けるかというと...

背景

量子アニーリングはm個の中からn個選ぶのが苦手

なぜなら…

- 制約がペナルティ項として目的関数につけられてしまう
- すると問題が非本質な方向へ最適化される

$$\begin{aligned} &\text{minimize} \quad H_{object} \\ &\text{subject to} \quad H_{constraint} = c \end{aligned}$$

量子アニーリングは m 個の中から n 個選ぶのが苦手

なぜなら...

- 制約がペナルティ項として目的関数につけられてしまう
- すると問題が非本質な方向へ最適化される

$$\text{minimize} \quad H_{object} + \underbrace{(H_{constraint} - c)^2}_{H_{penalty}}$$

式で見ると

彩色問題

$$\text{minimize} \quad \sum_{i,j \in Adj} \sum_{k \in color} q_{i,k} q_{j,k}$$

$$\text{subject to} \quad \sum_{i \in vertics} q_{i,k} = 1$$

↓複雑化

$$\text{minimize} \quad \sum_{i,j \in Adj} \sum_{k \in color} q_{i,k} q_{j,k} + \sum_{k \in color} \left(\sum_{i \in Adj} q_{i,k} - 1 \right)^2$$

巡回セールスマン問題

$$\text{minimize} \quad \sum_{i,j \in C} \sum_{k=0}^{n} w_{i,j} q_{i,k} q_{j,k+1}$$

$$\text{subject to} \quad \sum_{i \in C} q_{i,k} = 1$$

$$\sum_{k=0}^{n} q_{i,k} = 1$$

↓複雑化

$$\text{minimize} \quad \sum_{i,j \in C} \sum_{k=0}^{n} w_{i,j} q_{i,k} q_{j,k+1} + \sum_{k=0}^{n} \left(\sum_{i \in C} q_{i,k} - 1 \right)^2 + \sum_{i \in C} \left(\sum_{k=0}^{n} q_{i,k} - 1 \right)^2$$

つまり

量子アニーリングは組合せ最適化問題を解く手段の一つ

彩色問題

- 隣り合う場所は異なる色で塗分ける

つまり

量子アニーリングは組合せ最適化問題を解く手段の一つ

彩色問題

- 隣り合う場所は異なる色で塗分ける
- 各地点はちょうど一色で塗られる

巡回セールスマン問題

- 複数の街を最短経路ですべて訪れる

つまり

量子アニーリングは組合せ最適化問題を解く手段の一つ

彩色問題

- 隣り合う場所は異なる色で塗分ける
- 各地点はちょうど一色で塗られる

巡回セールスマン問題

- 複数の街を最短経路ですべて訪れる
- 各時間に訪れることのできるサイトはちょうど1か所

つまり

量子アニーリングは組合せ最適化問題を解く手段の一つ

彩色問題

- 隣り合う場所は異なる色で塗分ける
- 各地点はちょうど一色で塗られる

巡回セールスマン問題

- 複数の街を最短経路ですべて訪れる
- 各時間に訪れることのできるサイトはちょうど 1 か所
- 各地点ちょうど 1 回訪れる

つまり

量子アニーリングは組合せ最適化問題を解く手段の一つ

彩色問題

- 隣り合う場所は異なる色で塗分ける
- 各地点はちょうど一色で塗られる

巡回セールスマン問題

- 複数の街を最短経路ですべて訪れる
- 各時間に訪れることのできるサイトはちょうど 1 か所
- 各地点ちょうど 1 回訪れる

これらの制約を常に満たすように解を遷移させていけば効率が上がりそう

やってみた

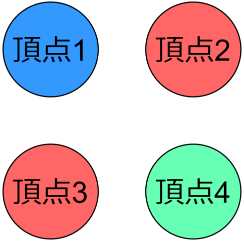
彩色問題

エネルギーが下がっていく様子を付けた 20×20 の彩色問題の動画

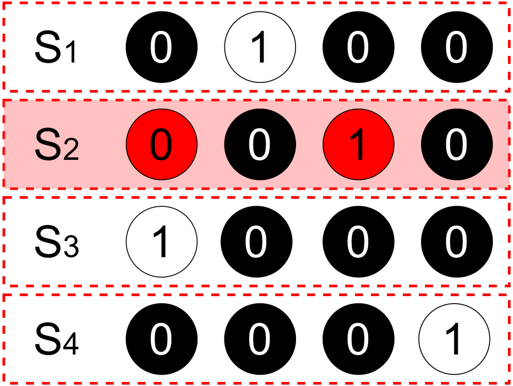
手法

有名な例：彩色問題

		赤	青	黄	緑
頂点1	S_1	0	1	0	0
頂点2	S_2	1	0	0	0
頂点3	S_3	1	0	0	0
頂点4	S_4	0	0	0	1



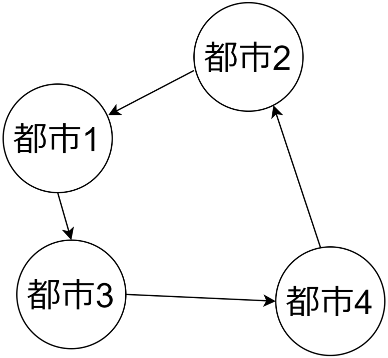
状態遷移方法

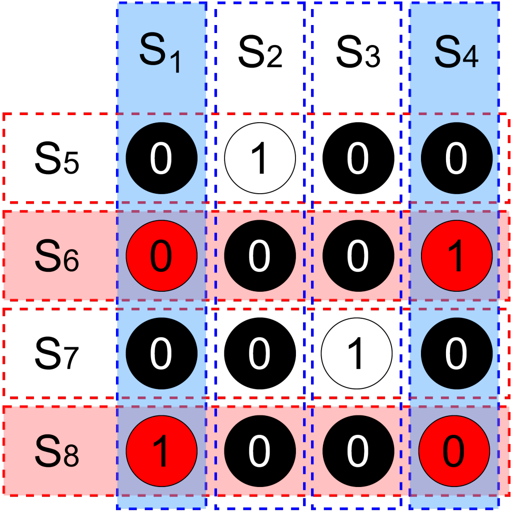


1. S_i を一つ選ぶ
2. 0 と 1 を交換する
3. 状態遷移完了

その次くらいに有名な例：巡回セールスマン

		都市1	都市2	都市3	都市4
		S_1	S_2	S_3	S_4
順番1	S_5	0	1	0	0
順番2	S_6	1	0	0	0
順番3	S_7	0	0	1	0
順番4	S_8	0	0	0	1





1. S_i を $1 \le i \le 4$ で一つ選ぶ
2.
3. 状態遷移完了

比較

比較

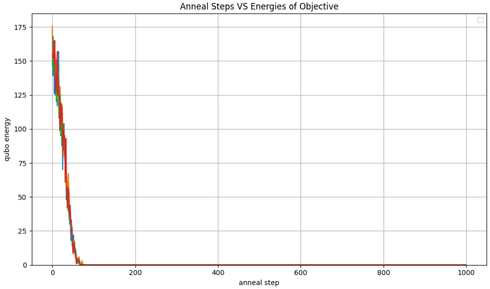


図 1: Swap based の QAS (lowest = 0)

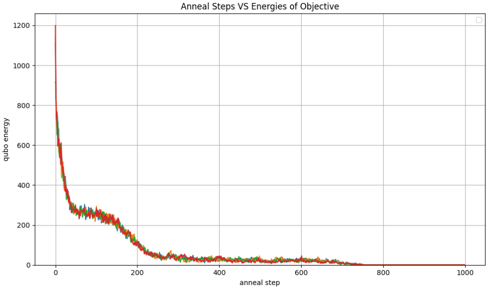
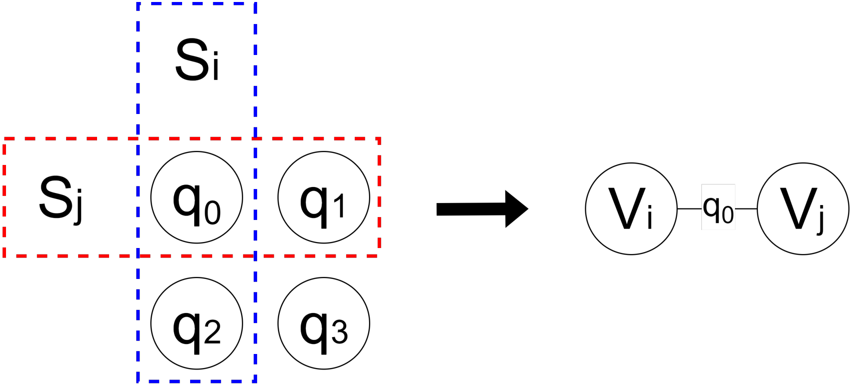


図 2: 普通の QAS (lowest = 2630)

※縦軸の目盛りが異なることに注意

手法（再）

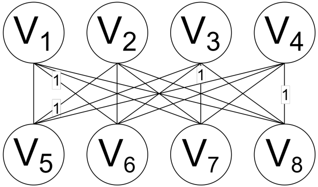
一般化をする



$S_i \cap S_j = q_k \rightarrow V_i$と V_j間に重み q_kのエッジが存在

巡回セールスマン問題再論

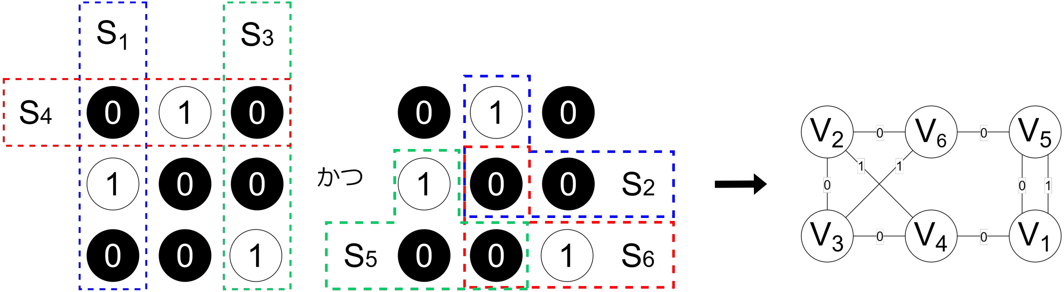
	S_1	S_2	S_3	S_4
S_5	0	1	0	0
S_6	1	0	0	0
S_7	0	0	1	0
S_8	0	0	0	1



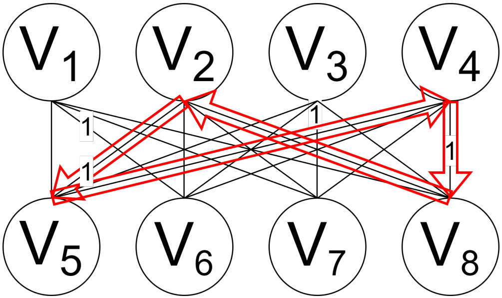
（重みは 1 のみ明記、0 は省略）

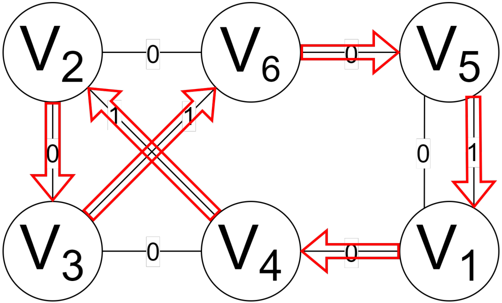
多くの問題が完全二部グラフに相当する

こんな問題は見たことないが



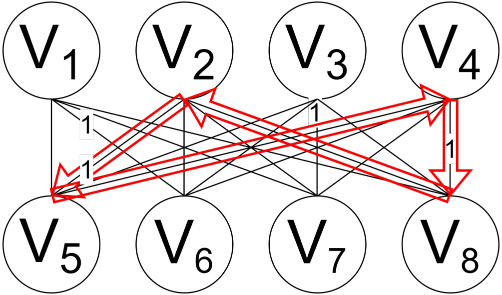
グラフで表現したことによる嬉しさ





問題によらず、ビットフリップ方法を統一的に説明できる

ビットフリップ方法



1. 条件を満たす閉路を探索する
   - 偶閉路
   - 重み 0,1 のエッジを交互に通る
2. 閉路上のエッジの重みをすべてフリップする
3. ビットの値をエッジの重みに対応するように更新

展望

これからしたいこと

- 一般化された制約条件下での条件を満たす閉路を探索するフェーズの考察
- より高次の制約条件への適用のための考察

参考文献