

1

モバイルネットワークのスライス埋め込み問題を対象とした量子遺伝子制御ネットワークアルゴリズムの適用手法

大阪大学 基礎工学部 情報科学科 村田研究室
関澤和希

特別研究報告発表 2025.2.13

1

2

研究背景

- **遺伝子制御ネットワーク (GRN) を用いた遺伝的アルゴリズム (GA) による通信ネットワークの動的な制御手法の研究[1]**
 - 映像分散処理システムのリソース割り当て制御の問題に対して適用
 - 最適解を事前に記憶させたGRNを用いることで記憶させた環境と類似する環境に短時間で適応可能
 - 課題が存在
 - GRNを用いないGAと比較して、行列計算による計算量の増大
- **量子ゲートを用いた量子遺伝子制御ネットワーク (QGRN) モデルに着目する**
 - 今後、量子コンピュータの発展による演算の高速化で計算時間の増大を解決できると期待
 - ただし、GRNと同等の特徴を示すかどうかは検証が必要

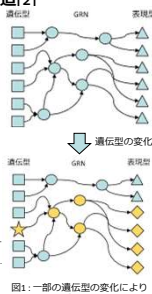
[1] S. Inoue, M. Yamauchi, D. Kominami, H. Shimomishi, and M. Murata, "Genetic algorithm with gene regulatory networks based optimization method for distributed video analysis system," in Proceedings of 27th Conference on Innovation in Clouds, Internet and Networks (ICIN), pp. 257-264, Mar. 2024.

2

3

遺伝子制御ネットワーク (GRN: Gene Regulatory Networks)

- 生物の遺伝子間の相互作用により形成されるネットワーク構造[2]
- GRNの特徴とネットワーク制御への応用
 - 一部の遺伝型の変化で表現型が大規模に変化[3]
 - 局所解の脱出[1]
 - 表現型の特徴を記憶し、将来の進化過程で再び発現[3]
 - 過去環境の最適解を記憶し、同様または類似した環境において利用[1]
 - 表現型の特徴を組み合わせた新たな表現型を生み出す機能が存在[4]



[1] S. Inoue, M. Yamauchi, D. Kominami, H. Shimomishi, and M. Murata, "Genetic algorithm with gene regulatory networks based optimization method for distributed video analysis system," in Proceedings of 27th Conference on Innovation in Clouds, Internet and Networks (ICIN), pp. 257-264, Mar. 2024.
[2] E. H. Davidson and D. H. Erwin, "Gene regulatory networks and the evolution of animal body plans," Science, vol. 311, pp. 796-800, Feb. 2006.
[3] P. A. Dinyarek, S. S. Puntambekar, and C. J. Gadgil, "Method for identification of sensitive nodes in boolean models of biological networks," IET Systems Biology, vol. 13, pp. 1-6, Feb. 2023.
[4] R. A. Watson, G. P. Wagner, M. Pavlicek, D. M. Weinreich, and R. Mills, "The evolution of phenotypic correlations and "developmental memory" Evolution, vol. 68, pp. 1324-1338, Apr. 2014.

図1: 一部の遺伝型の変化により表現型が大規模に変化

3

4

量子遺伝子制御ネットワーク (QGRN: Quantum GRN)[5]

- GRNにおける遺伝型同士の相互作用を量子演算子を用いて表現したモデル
 - 量子コンピュータに適用可能なモデルであるため、将来的には高速な計算が期待できる
 - 既存のGRNでは表現できない遺伝子間相互作用を表現可能
 - 発現する表現型および記憶する表現型を確率分布として扱うことが可能
 - 既存のGRNと比較して、複数の表現型の記憶に適している
- 動的なネットワーク制御問題に対するQGRNの適用可能性
 - 過去の最適解の想起による短時間の最適解の発見
 - 量子計算による計算時間の短縮
 - 大規模な問題に対する適用可能性
 - QGRNが変異した際の複数の表現型ビットの同時変異による局所解の脱出

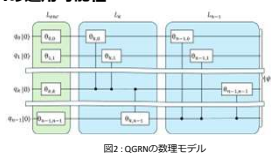


図2: QGRNの数理モデル

[5] C. Roman-Vichara and J. I. Cui, "Quantum gene regulatory networks," arXiv Quantum Information, vol. 9, July 2023.

4

5

研究目的と研究手順

- **研究目的**
 - 動的なネットワーク環境の変動が想定される問題設定に対するQGRNの適用可能性の評価
 - QGRNはGRNと同様の特徴を有するか
- **研究手順**
 - QGRNを用いた最適化アルゴリズムの構築
 - QGRNの記憶性能の検証
 - QGRNにより、既存のGRNと比較して記憶性能が向上するか
 - 動的な環境変動を想定したモバイルネットワークのスライス埋め込み問題に対するQGRNの適用検証
 - 期待されるQGRNの特性が現れるか
 - ネットワークスライシング
 - 1つの物理ネットワークを通信要求に応じて仮想的にスライスし、リソースを物理ネットワークに割り当てる技術




図3: ネットワークスライシング

5

6

量子遺伝子制御ネットワークを用いた最適化アルゴリズム

- 適応度の高い表現型を発現しやすいようQGRNを変異
 - 最適化問題の目的関数をFitness関数として設定
 - 各QGRNの発現する表現型の適応度をFitness関数をもとに算出
 - 最も適応度の高い表現型を最適化問題の解として適用
- **アルゴリズムの流れ**
 1. QGRNの集団を形成
 2. QGRNの適応度を評価
 - a. QGRNを観測し、表現型を取得
 - b. Fitness関数からQGRNの適応度を算出
 3. GAによるQGRNの最適化
 - a. 適応度の高いQGRNをエリートとして保存し、適応度の低いものにコピー
 - b. エリート以外に突然変異を適用
 4. 2, 3.を繰り返して問題に適合するQGRNに最適化

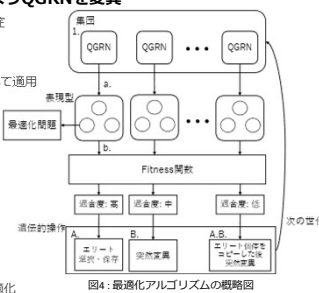


図4: 最適化アルゴリズムの概略図

6

記憶能力の評価

7

- 問題設定
 - 5つの表現型をGRN・QGRNにそれぞれ記憶させる[4]
- 評価結果
 - 従来のGRNと比較して記憶能力が向上
 - QGRN: 平均4.67個
 - GRN: 平均1.33個
 - 記憶させた表現型(赤字)と似た型(青字)が生成されている
 - 5つの表現型の各ビットにおいて0が多いor 1が多い
 - 表現型の特徴を組み合わせた新たな表現型を生み出す機能を有する

表2: 記憶能力の検証に用いる表現型

表現型	記憶させる表現型
表現型1	00100001111101
表現型2	00101010011111
表現型3	00001010111111
表現型4	00100001101001
表現型5	01110101110101

表3: QGRNが記憶した表現型

再現順位	シミュレーション1	シミュレーション2	シミュレーション3
1	00101010011111	00001010111111	00101010011111
2	00001010111111	01110101110101	00100001111101
3	00100001111101	00100001111101	00001010111111
4	00100001101001	00101010111111	00100001101001
5	00100001101001	00100011111101	01110101110101
6	01110101110101	00000000000000	00000000000000

表4: GRNが記憶した表現型

再現順位	シミュレーション1	シミュレーション2	シミュレーション3
1	01110101110101	01110101110101	11011110000010
2	10001010001010	01110101110001	00100001111101
3	01100011101001	01110001110101	00110101110101
4	01110001011101	01110101111101	10001010001010
5	01110001110101	10001010011010	00110001111101
6	01111101110101	00110101110101	01110101110101

[4] R. A. Watson, G. P. Wagner, M. Pavlov, D. M. Weisbach, and R. Mills, "The evolution of phenotypic correlations and "developmental memory" Evolution, vol. 68, pp. 1124-1138, Apr. 2014.

7

スライス埋め込み問題に対する適用検証: 問題設定

8

- ネットワークの設定
 - 物理ネットワーク
 - ユーザ端末2台, エッジサーバ2台, クラウド2台の階層構造
 - ネットワークスライス
 - 2つのスライス方法を設定
 - リンクとアプリケーションをそれぞれ物理ネットワーク上に埋め込む

図6: 物理ネットワークの設定 図7: ネットワークスライス1 図8: ネットワークスライス2

8

表現型とスライス埋め込み方法とのマッピング

9

- スライスの埋め込み先を決定するため、表現型は14bitで表現
 - アプリケーションの埋め込み先はエッジ・クラウドの数字に対応

クラウド10へ クラウド11へ エッジ01へ エッジ01へ

[00 01 10 11 10 11 01]

リンク リンク application1 application2 リンク application3

図8: 表現型とスライス埋め込みの対応

9

評価環境

10

- 最適解となる表現型が異なる3つのネットワーク環境を設定
 - 環境A→環境B→環境Cのサイクルを30分間隔で繰り返す
- 実行環境
 - 量子フレームワークであるQiskitを使用して、古典コンピュータ上で実行
- GA (QGRNを用いない) とQGRNとの比較

表5: 実測に用いた実行環境

項目	内容
プロセッサ名	Intel(R) Xeon(R) Gold 6226R CPU @ 2.90GHz
スレッド数	64スレッド
コア数	32コア(16コア×2ソケット)
メモリ容量	131.65 GB
OS	Ubuntu 20.04.6 LTS
使用したソフトウェア	Python 3.11.6
使用したフレームワーク	Qiskit 1.2.4

図9: 変動する環境設定

10

評価結果: 実時間に対するFitnessの推移

11

- QGRNはGAでは到達困難な最適解へ到達可能
 - QGRNは複数の表現型が同時に変化することで局所解を回避

表6: 環境ごとの最適なFitnessの値

環境	最適解のFitness
環境A	-2263.22
環境B	-8168.18
環境C	-6733.23

図10: 環境A,B,Cが30分ごとに変動する環境下でのfitnessの推移

11

評価結果: 実時間に対するFitnessの推移

12

- QGRNはGAでは到達困難な最適解へ到達可能
 - QGRNは複数の表現型が同時に変化することで局所解を回避

表6: 環境ごとの最適なFitnessの値

環境	最適解のFitness
環境A	-2263.22
環境B	-8168.18
環境C	-6733.23

図11: 環境A,B,Cが30分ごとに変動する環境下でのfitnessの推移

12

評価結果：最適解到達性について

13

- **最適解に到達した割合 (最適解に到達した回数/環境変動回数)**
 - QGRN：高確率 ↔ GA：低確率
 - 一度の変異で複数の表現型が同時に変異するQGRNの特性により、局所解への収束を回避したため
- **最適解に到達するまでの時間**
 - GAでは最適解に到達しにくく、ほとんどの場合は30分以内で最適解に到達しなかった
 - パラメータ調整は今後の課題
 - 最適解に到達したケースに限った比較でも、環境A→BではQGRNの方が短時間で最適解に到達
 - 複数の表現型ビットの同時変異によるもの

表7：環境変動時におけるQGRNとGAの最適解到達時間
(最適解に到達したケースのみ)

評価方法	環境C→A	環境A→B	環境B→C	
QGRN	最適解到達率	100% (48/48 回)	100% (48/48 回)	95.83% (46/48 回)
	平均値 (秒)	37.13	431.83	427.23
	中央値 (秒)	22.99	322.43	334.70
	標準偏差 (秒)	56.12	412.11	398.59
GA	最適解到達率	18.75% (9/48 回)	25% (12/48 回)	54.17% (26/48 回)
	平均値 (秒)	37.25	709.61	29.58
	中央値 (秒)	0	596.34	0.15
	標準偏差 (秒)	143.68	444.47	157.31

13

まとめと今後の課題

14

- **QGRNの記憶性能の検証**
 - 既存のGRNと比較して、表現型の記憶能力が向上
 - 表現型の特徴を組み合わせた新たな表現型を生み出す機能を有することを確認
- **動的な環境変動が想定される問題へのQGRNの適用性を検証**
 - GAと比較して安定して最適解に到達可能
 - 遺伝子間の制御作用の変動による複数の表現型ビットの同時変異により局所解を回避
 - 環境の最適解を記憶している様子は見られなかった
- **今後の課題**
 - ネットワーク制御問題の適用における最適解の記憶
 - Fitness関数やGAの設定を工夫する必要がある
 - 適用する問題規模の拡大
 - 拡大可能な規模感の確認
 - NWの階層化による量子ビット数の削減

14