

# 進捗管理・報告(2020/8/25)

## 1. 現在取り組んでいること

可逆コンピューティングの研究

## 2. 進捗状況

・可逆コンピューティングについてと、この分野で行われてきたことの調査

## 3. 前回からの進捗

課題：

可逆線形探索と、木構造の可逆深さ優先探索の関係性をまとめる。

↓

・木構造の深さ優先探索は、右手法(左手法)にしたがって必ず一意の順序で行われる(グラフでは閉路が存在するため、右手法に従うとループする)。したがって、図1の木構造は必ず

③→⑧→⑨→⑧→②→⑧→③→④→⑥→④→③

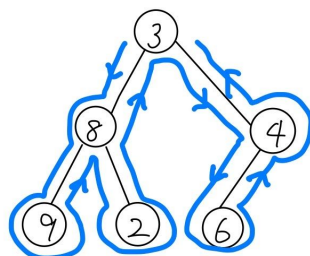


図1 木構造

の順で探索される。視点を変えれば、これらの要素が入ったレコードを線形探索していると捉えることができる。

深さ優先探索のような計算量を必要とせず探索順序が一意の列に並ぶアルゴリズムでは、計算量は線形探索と同等になるのではないだろうか？

幅優先探索などでも探索要素を一列に並べることができれば、一列に並んだ要素の探索は線形探索でできる？一列に並べることに計算量が必要？データ構造によって異なる？

・可逆線形探索では、入力データ構造が計算量に影響を与えていた。可逆線形探索でも、同様の問題が考えられる。例えば、単方向リストの木構造では、どのような探索順序を経てきたかをゴミ情報として保存する必要がある(単方向リストでは有効グラフしか表現できないが、有効グラフで表現した木構造は木構造なのだろうか...?)。また、出力の形式も問題である。インデックスを返す深さ優先探索は、出力であるインデックスでpreを消去できるため、1パスで実装可能である。しかし、出力がフラグであったり、要素の個数であったりした場合は、preを消去するためにBennett法を使うかゴミ情報を出力する必要が出てくる。

調査：

・Reversible Computation and Reversible Programming Languages

[https://docs.google.com/document/d/1tPj9I0C9S\\_BN2KPH8cw7Vqr5mDXK29cAfMb6WCRC0PU/edit#heading=h.scb358uk98yp](https://docs.google.com/document/d/1tPj9I0C9S_BN2KPH8cw7Vqr5mDXK29cAfMb6WCRC0PU/edit#heading=h.scb358uk98yp)

#### 4. 今後の課題

- ・ Analyzing Trade-offs in Reversible Linear and Binary Search Algorithms で提案されている線形探索の問題点を発見し、改善案を提案する。
  - ・ 大会発表に用いるスライドを作成する。
  - ・ 進捗報告を日報形式にする。

#### 5. その他

- ・ 2020 年度南山大学パツへ研究奨励金 I-A-2 について、申請等必要でしょうか？
- ・ r-while-web を公開しました <http://153.127.69.31/RWHILE-WEB/2>
- ・ このアプリケーションは、Dockerコンテナ内で動作しています。
  - Dockerイメージ [https://github.com/yokoyama-lab/OCaml-PHP-Apache2\\_Image](https://github.com/yokoyama-lab/OCaml-PHP-Apache2_Image)
- ・ ただ、なぜかコンパイルされた ri ファイルがうまく動いていないようで、プログラムの結果が返ってきません...(正確には "" が返ってきます)。ローカルでは動くので、r-while-web の問題ではなく、環境のせいだと思うのですが...
- ・ シンタックスハイライトは調整中です。