



HOKKAIDO UNIVERSITY

Title	Studies on Permutation Set Manipulation Based on Decision Diagrams [an abstract of dissertation and a summary of dissertation review]
Author(s)	井上, 祐馬
Degree Grantor	北海道大学
Degree Name	博士(情報科学)
Dissertation Number	甲第12622号
Issue Date	2017-03-23
Doc URL	https://hdl.handle.net/2115/65364
Rights(URL)	https://creativecommons.org/licenses/by-nc-sa/2.1/jp/
Type	doctoral thesis
File Information	Yuma_Inoue_abstract.pdf, 論文内容の要旨



学 位 論 文 内 容 の 要 旨

博士の専攻分野の名称 博士（情報科学） 氏名 井上 祐馬

学 位 論 文 題 名

Studies on Permutation Set Manipulation Based on Decision Diagrams

（ 決定グラフに基づく順列集合処理に関する研究 ）

日常生活の中でよく、人はものに順序を付けることが必要になる。例として、優先順位に従ってものを並べ替えたり、たくさんの仕事を順に処理したり、web 検索の結果に順位を付けて表示したり、行きたい名所を順々に巡ったり、などが挙げられる。数学の用語では、このような順序付けのことを順列と呼ぶ。順列は集合上の全単射関数と見ることもできるため、他にも離散的な関数やマッチングなどの場面でも用いられる。

答えを順列で表せるような種々の問題に対し、“よい” 順列を 1 つ見つけてくるような手法は、数多く研究されてきた。例えば、巡回セールスマン問題は“最短” で移動できるような都市の訪問順列を求める問題であり、単一機械スケジューリング問題は“最小” ペナルティとなる仕事の処理順列を求める問題である。一方で、現実には道路や優先度の状況は刻一刻と変わる可能性がある。このような場合に答えを 1 つだけ求める手法を用いると、状況変化ごとに何度も“よい” 順列を計算し直す必要がある。

順列の列挙を行うことは、この問題に対応する手段の 1 つである。もし制約を満たす解を全て保存し、すぐに新たな状況下での“よい” 順列を求められるよう索引化することができれば、再計算するよりも高速に新たな“よい” 順列を知ることができる。また、解を全て列挙することには他にも利点がある。例えば、条件を満たす解の個数を数えたり、ランダムサンプリングを行ったり、追加の条件で絞り込んだものをさらに取り出したり、といったことが考えられる。様々な条件について、それを満たす順列を全て列挙する研究も盛んに行われてきた。しかしながら、順列の個数はその長さの階乗のオーダーで多くなっていくため、それほど大きくない集合の順列を列挙することも現実的ではなくなってしまう。

この組合せ爆発を回避するためのアイデアとして、保存・索引化に圧縮データ構造を利用することが考えられる。本論文では、順列決定グラフ (π DD) と呼ばれるデータ構造に着目する。このデータ構造は順列の集合を省スペースで表現できるのみならず、順列集合間の二項演算をサポートしている。さらに、保存している順列に対する書き換え操作や条件絞り込み操作などが圧縮された π DD の大きさに依存した時間で行える。これらの演算・操作は保存している順列の個数には直接依存しないため、順列の集合が π DD としてコンパクトに圧縮されていれば、 π DD 上での操作も高速に行えることが期待できる。

π DD は強力なデータ構造であるにも関わらず、順列を解とする問題に対する適用例が今までほとんど報告されていなかった。よって著者は、 π DD を様々な順列問題に適用するとともに、実験的な側面・理論的な側面の両方からその性能を解析する。第 3 章にて、実際に著者が発見した、 π DD の適用が有効であった 2 つの問題例、可逆回路のデバッグ・順列集合のサイクルタイプ同値類分割について、その適用法と性能を示す。これらの問題に対し、実際に解を列挙してそれを 1 つ 1 つデータ構造に追加していたのでは、階乗のオーダーの計算時間を回避できず、意味がない。したがって、著者は

個々の問題の性質を分析し、その性質を利用した π DD の構築アルゴリズムも提案している。これらのアルゴリズムは、 π DD の演算を巧みに利用したり、演算の回数そのものを減らすよう工夫を凝らしたものである。より詳細には、可逆回路デバッグ問題については、 π DD を用いない新たなアルゴリズムによって、1 つのみエラーを含む回路の高速なデバッグに成功した π DD を利用することで、このアルゴリズムを 2 つ以上のエラーを持つ回路に拡張できることも示した。また、サイクルタイプ同値類分割については、順列のサイクルが持つ性質が π DD による圧縮表現に適していることを示した上で、その利を活かした分割アルゴリズムを与えた。

また、 π DD の適用がうまくいかなかった例も存在する。そのような問題群に対し、著者は π DD に変更を加えた新たなデータ構造・Rot- π DD を提案するとともに、Rot- π DD を適用することで圧縮効率・データ構造の構築時間がともに向上することを示した。4 章では Rot- π DD を導入するとともに、問題例として、オイラー路の列挙・有向グラフのトポロジカル順序の列挙・パターン回避順列の列挙の 3 つを挙げ、その効果を実験的・理論的に評価する。より詳細には、オイラー路列挙問題については、動的計画法に基づく Rot- π DD の構築アルゴリズムを与え、Rot- π DD が π DD よりもこの問題に適していることを理論的に示した。トポロジカル順序の列挙については、オイラー路列挙問題同様の動的計画法を構築アルゴリズムに用いることができることを示した。また、新たな Rot- π DD 演算を提案し、これによって動的に新たな辺を加える操作に対応した索引化を可能にした。最後にパターン回避順列の列挙については、まずは π DD の構築アルゴリズムを与えた。さらにこのアルゴリズムにおいて、 π DD の代わりに Rot- π DD を用いると、アルゴリズムの途中で必要となる決定グラフの大きさが理論的に小さくなることを示した。これは Rot- π DD の利用によりアルゴリズムが高速化する可能性を示唆しており、実際に高速化が達成されることを実験により確認した。ここで、どのように π DD・Rot- π DD を使い分ければよいかという疑問が生まれる。4 章の最後では、さらなる実験を通じて、問題に応じた 2 つの決定グラフの選択方法についても検討する。

5 章ではさらに、決定グラフの構造に潜む順列問題である、決定グラフの変数順序付けについても取り扱う。変数順序の選び方は、決定グラフの圧縮効率に劇的に影響することが知られている。著者は主にグラフに含まれる部分グラフを表現する際の決定グラフについて、ヒューリスティクスに基づく変数順序付けアルゴリズムを与えた。このヒューリスティクスは、部分グラフを表す決定グラフの大きさが元のグラフのパス幅というグラフパラメータと関係しているという洞察に基づいたものである。このアルゴリズムによってよい変数順序が得られることを実験的評価により確認した。これは決定グラフを用いた多くの既存アルゴリズムの性能をさらに向上させる可能性がある結果とも言える。

本論文の一連の研究を通じて、 π DD および Rot- π DD を様々な問題に適用するための技法の発見や、問題に応じた性能の理論的・実験的評価により、いくつかの順列問題に対する順列決定グラフの適用が効果的であることを示した。また、複数の問題で共通して利用できるアイデアや、順列パラメータに着目した場合の効率の実験的評価などを与え、本研究で取り上げた問題以外への順列決定グラフの利用に対する示唆を与えた。さらに、既存の決定グラフを用いたアルゴリズムの性能向上に繋がる手法を与えた。これらの結果により、本論文で取り上げた問題の現実的応用や、他の多くの問題の効率的な解決に繋がるのが期待できる。