科学研究費助成事業　　研究成果報告書

平成　２６　年　　６　月　２２　日現在

機関番号：　２１６０２
研究種目：　基盤研究(C)
研究期間：　2011～2013
課題番号：　２３５０００２１

研究課題名（和文）高階ナローイングにもとづくＸＭＬ文書処理の検証技術

研究課題名（英文）Verification of XML Transformation based on Higher-Order Narrowing

研究代表者
鈴木　大郎（SUZUKI, TARO）

会津大学・コンピュータ理工学部・上級准教授

研究者番号：９０２７２１７９

交付決定額（研究期間全体）：（直接経費）　　2,800,000円、（間接経費）　　840,000円

研究成果の概要（和文）：XML変換の記号的な検証技術として、ヘッジ書換え系を対象としたナローイングに関する理論的性質について研究を行った。左線形、右平坦な構成子系というヘッジ書換系を対象としたナローイング計算系を設計し、その理論的性質について検討した。
変換対象のXML文書への変換関数の適用と、望ましくない変換結果をそれぞれ一般化したヘッジからなる等式をこのナローイング計算系に与えたとき、ナローイングが成功し、それに続くヘッジ単一化が成功すると望ましくない変換の例を得ることができる。この計算系が健全性、完全性、前単一化の決定可能性という理論的に望ましい性質を満たすことを示すことができた。

研究成果の概要（英文）：We investigated theoretical properties of narrowing for hedge rewrite systems as a verification of XML transformation. A narrowing calculus is designed for left-linear, right-flat constructor-based hedge rewrite systems. Given an equation with an application of transformation function to an XML document in the one side and an undesirable transformation result in the other side, the narrowing calculus produces a narrowing derivation and a unifier of the final equation in the derivation as an instance of undesirable transformation. We have shown that the narrowing calculus is sound and complete, and preunification problem associated with the calculus is decidable.

研究分野：　総合領域

科研費の分科・細目：　情報学・情報学基礎

キーワード：　ナローイング　XML　正規表現

１．研究開始当初の背景

XML 文書処理は情報化社会のインフラストラクチャを支える技術としてますます有用性が増しているが、その検証技術は十分に進んでいるとはいえない。また、検証技術の有効性を保証するための理論的研究も発展途上にある。

XML は主にインターネット上でやりとりされる構造化されたデータを記述するための枠組みとして、広く使われている。現在では、多くのデータやプロトコルが XML 文書として記述されており、それらを記述するための様々なフォーマット(XML ボキャブラリ)が存在している。このような状況下で、XML で記述されたデータへの問い合わせや、XML 文書データの更新や、ある XML ボキャブラリで書かれた XML 文書から他のボキャブラリに準拠した形式への変換などといった XML 文書処理が頻繁に発生している。このような処理のためには，Java，Perl，Python といった様々な既存のプログラミング言語や、XSLT，XQuery，XML Update Facility といった W3C により規格が決められた XML 処理言語で書かれたプログラムが使われる。しかし、このようなプログラム，特に後者(XSLT，XQuery，XML Update Facility)のプログラムが正しく文書処理を行っていることを検証する技術はあまり発達していない。

XML 文書処理の検証として重要なのは、静的型検査の正当性、およびアクセス制御ポリシーの妥当性検証である。静的型検査とは、XML ボキャブラリ間の変換プログラムが扱う変換前後の XML 文書の型(すなわち XML ボキャブラリ)が与えられているとき、実際の変換結果が与えられた型をみたすかどうかを変換を行う前(コンパイル時)に検査することである。アクセス制御ポリシーとは，XML 文書の各節点に対するアクセス権限を与える形式と、その形式により記述されるアクセス権限規則の集合である。

XML 文書は形式的にはヘッジ(同じラベルをもつ節点の子の数が不定な木)で表される。また、XML 文書の型は、正規ヘッジ表現(文字列を表す正規表現をヘッジを表すように拡張したもの)で表される。本研究代表者らは，XML 文書変換などの XML 文書処理をヘッジ間の書換えと捉え、正規ヘッジ表現により型付けされた高階ヘッジ書換え系を用いた計算モデルについて研究を行ってきた。この書換え系の静的型検査の正当性に関する研究、正規ヘッジ表現によるパターン照合に関する理論的研究、ヘッジ書換え系でのパターン照合の高速化のための理論に関する研究を通じて、書換えモデルが XML 文書処理の理論的基礎を与えるものとして有用であるという感触を得ている。さらに、 Jacquemard らによる書換えモデルによるアクセス制御ポリシーの妥当性検証に関する研究結果を検討し、書換えモデルによる XML 文書処理の検証が有用であるという考えをもった。彼らが得た結果をさらに拡張するために、本研究代表者と分担者が研究してきた型付きの高階書き換え系とナローイング(書換えで用いられるパターン照合を単一化(unification)に拡張したもの)を用いることで，より有用な検証技術のための理論が得られるのではないかと考えた。

２．研究の目的

本研究では、XML 文書処理の記号的な検証技術に関する理論的研究を行う。とくに、本研究代表者らがこの分野で貢献できる対象である、項書換え系による検証技術、とくに正規ヘッジ表現とナローイングにもとづく検証技術についての研究を行う。

XML 文書処理の検証技術として有用な、正規ヘッジ表現で型付けされた書換え系およびナローイングの定式化を与え、その理論的性質について調べることで、ナローイングを用いた検証技術の正当性を保証することを目指す。

また、欲張り戦略による曖昧さ解消ポリシーにもとづいた文字列と正規表現とのパターン照合を、マッチングオートマトンに用いて定式化する方法についても検討する。

３．研究の方法

まず XML 文書処理の検証のためのモデルとして適した書換え系とナローイングについての定式化を行う。ここでは，本研究代表者らが以前に提案したヘッジ書換え系に関する研究と、項書換え系に対するナローイングの研究を最大限に利用する。

次に、定式化したナローイングに関する理論的な性質の解明を行う。主に検討すべき性質は、完全性と到達可能性である。ナローイングは単一化を用いるため、ナローイングの対象(本研究ではヘッジ)に含まれる変数への代入がナローイングの過程で生成される。

完全性とは、与えられたヘッジからの書換えによって得られるすべてのヘッジに対応する(変数を含む)ヘッジと代入がナローイングによって得られることを保証する性質である。完全性を保証することで、書換えにより得られるヘッジの集合に関する性質をナローイングを使って抜けを生じることなく議論できるようになる。到達可能性とは、2つのヘッジの集合が与えられたとき，一方の集合に属する任意のヘッジからもう一方の集合に属するあるヘッジへのナローイングが存在するという性質である。到達可能性が決定可能かどうかを調べることは、XML 文書処理の検証にとって重要である。

また、欲張り戦略による曖昧さ解消ポリシーにもとづいた文字列と正規表現とのパターン照合では、以前に設計した、OSIX 標準規格準拠の曖昧さ解消ポリシーにもとづいたパターン照合を行うアルゴリズムをベースとする。２つの曖昧さ解消ポリシーを比較し、それらの間の違いを検討することで、POSIX

準拠でのアルゴリズムから、欲張り戦略でのアルゴリズムを導く。

４．研究成果

本研究の主要な成果は以下の通りである。

(1) XML 変換の検証のためのナローイング計算系の設計

ヘッジを扱うナローイング計算系の設計とその健全性・完全性を証明した。XML 検証のためのナローイングについて 2 通りの定式化を行い、そのうちの１つについて健全性と完全性を示すことができた。

ナローイングの一つ目の定式化は、文脈をもつヘッジ書換え系を使って、つねに XML 文書全体に対してナローイングを行うようなものである。この書換え系は、当初の計画で予定していた XML Update Facility での変換を定式化するものである。XML Update Facility で提案されている変換規則を書換え規則として記述すると、規則の左辺に現れない変数が右辺に出現する。通常のナローイングでは、このような変数(外変数)にはヘッジを代入することができないので、外変数に対応する位置で変換を行う場合をシミュレートできないという問題が生じる。そこで、書換え規則に文脈を導入し、ナローイングがつねに XML 文書全体に適用されるようにした。こうすることで、ナローイングにより導入された外変数に対応する位置で XML Update Facility による変換が行われた場合でも、変換列に対応するナローイング列を考えることができる。しかし、この方法では文脈単一化による代入の列挙が必要となるが、既存の文脈単一化は代入の存在性を保証するものだけである。そこで、先行研究を参考に代入の列挙方法を検討したが、有効な方法は得られなかった。そのため、この方法では完全性をもつナローイングを得ることはできなかった。

もう一つの定式化は、変換関数に XML 文書を与えることで行われる変換を検証するためのナローイングである。XML Update Facility による変換では、XML 文書(またはその一部)を直接変換可能な対象とみなすので、XML 文書に現れるラベル自体が変換関数名を表す。それに対して、この定式化が扱う変換では XSLT と同様に変換のための関数を別に用意するので、関数名は XML 文書のラベルと異なるものになる。これは構成子系と呼ばれる書換え系に対応する。また、XML Update Facility による返還とは異なり、変換には外変数のないヘッジ書換え規則を用いる。ここで扱う変換では、変換関数を XML 文書に適用し、関数記号がなくなるまで変換を行う。このような変換では、変換関数を表すヘッジ書換え系としては、左線形(書換え規則の左辺に同じ変数が現れない)、右平坦(書換え規則の右辺で関数記号がネストしない)な構成子系に制限してもあまり不便を感じることはない。そのため、この定式化では上記の性質を満たすヘッジ書換え系に対象を絞った。さらに、望ましくない変換が可能かどうかをナローイングで調べるために、左辺に変換の初期状況(XML 文書に変換関数に適用したもの)、右辺に望ましくない結果である XML 文書(ただし、それを特徴付ける箇所以外は変数としたヘッジ)からなる等式からナローイングを始めることとした。この等式の左辺だけにナローイングを適用し、両辺が等しくなるような代入が存在する等式で終わるナローイング列が得られれば、望ましくない変換が存在することになる。以上のように、問題を設定した上で、完全性などの望ましい性質をもつナローイング計算系の設計を行った。

その結果、上記のようなナローイングでは、高階単一化で使われる前単一化(preunification)アルゴリズムが有効であることを見出した。前単一化アルゴリズムとは、高階項の等式の両辺を等しくする代入を見つける際、両辺の対応する位置がともに変数になったらそれ以上の単一化を行わないようなアルゴリズムである。これを用いると単一化の探索空間を狭めることができる。一方で、一般的なヘッジ書換え系でのナローイングに前単一化アルゴリズムを用いると、このアルゴリズムで代入が行われない変数の位置で変換を行うような変換列をナローイング列で表現できなくなってしまう。しかし、ここで考えている変換列の場合、変換開始時点では関数記号はヘッジ(この場合は不定木)の根だけに現れ、変換の途中での関数記号も書換え規則により陽に導入されるものだけなので、対応するナローイング列でも絶対に変数になることはない。したがって、前単一化アルゴリズムを用いても任意の変換列をナローイング列で表現できるという利点がある。また、通常の高階単一化では前単一化アルゴリズムが成功すれば等式の両辺を等しくする代入が必ず存在するが、両辺がヘッジの場合には必ずしもそのような代入があるとは限らない。したがって、前単一化アルゴリズムを用いたナローイングによって、途中で失敗することなく終了するナローイング列が得られたとしても、望ましくない変換列が存在するとは言えない。なぜなら、得られたナローイング列から望ましくない変換列を得るための代入が存在しないかもしれないからである。これを解決するために、前単一化アルゴリズムを用いたナローイングにより途中で失敗しないナローイング列が得られたら、その列で最後に得られた等式に単一化アルゴリズムを適用することで、望ましくない変換列が存在するかどうかを調べる。変数を含むヘッジは文脈変数を含む二分木で表せるので、文脈単一化アルゴリズムを用いることができる。一つ目の定式化の説明で述べたように、既存の文脈単一化アルゴリズムは解の存在だけを保証するが、ここでの目的にはそれで十分なので、それを用いて

望ましくない変換列が存在するかどうかを調べることができる。
以上のような考察の結果から、前単一化アルゴリズムとナローイングを組み合わせたナローイング計算系を設計した。これを用いた検証の流れは以下のようになる。まず、与えられた等式に対して上記のナローイング計算系によってナローイングを行い、望ましくない変換を表すナローイング列の候補を列挙する。その各々に対して、既存の文脈単一化アルゴリズムを適用し、それが実際に望ましくない変換列を表しているかどうかをチェックする。
このナローイング計算系について、(a)健全性 (b)完全性 (c)前単一化の決定可能性の３つの性質が成り立つことを示せた。性質(c)はナローイングの過程で行われる前単一化が等式のサイズに比例した時間で終わることと、異なる代入を作り出す分岐が有限(高々２つ)であることから、前単一化の探索空間が有限になることを示している。

(2) マッチングオートマトンを用いた欲張り戦略にもとづく文字列パターン照合アルゴリズムの設計
XML 文書の変換では、変換規則に現れる変数への代入は一意には決まらず、複数存在する。(1)で述べたナローイングでは、それらすべてを列挙して、どのような場合にも対応できるものを考慮した。一般には、変数に割り当てられた正規ヘッジ表現型を利用して、正規木表現と文字列とのパターン照合に対して何らかの曖昧さ解消ポリシーを導入することで、可能な代入を一つに絞ることがよく行われる。その際よく使われるポリシーは欲張り戦略(greedy semantics)と呼ばれる。そこで、文字列と正規表現とのパターン照合を欲張り戦略による曖昧さ解消ポリシーにもとづいて、マッチングオートマトンを用いて定式化することを目指した。以前に我々が設計した POSIX 標準規格に準拠した曖昧さ解消ポリシーにもとづくパターン照合では、パターン照合の様子を表す解析木を導入した。曖昧さ解消ポリシーから選択される結果に対応する解析木が、与えられた正規表現と文字列とのパターン照合を表す解析木の中で最小のものとなるような順序を定式化した。本研究では、まず欲張り戦略によって選択される結果に対応する解析木が最小になるような順序を定式化することからはじめた。その結果、そのような順序の定義が、POSIX 標準規格準拠の場合の順序の定義の一部を用いるだけで得られることを見出した。つぎに、実際にパターン照合を行うオートマトンの設計を、POSIX 標準規格準拠の場合のオートマトンをベースにして行った。その結果、オートマトンそのものは両者でほぼ同じとなることがわかった。唯一の違いは、クリーネ閉包がネストしたときに選択される結果の違いから生じる部分だけであることがわかった。つぎに、このオートマトンを用いたパターン照合アルゴリズムを、解析木に関する順序にもとづいて設計した。その結果、欲張り戦略にもとづくパターン照合アルゴリズムは、POSIX 標準規格に準拠したパターン照合アルゴリズムと比べて、劇的に単純化できることがわかった。単純化のカギとなるのは、オートマトンの初期状態から文字列のプレフィックスを読んだ後に到達可能な状態の間に厳格な全順序が成り立つという性質である。この性質を用いることで、POSIX 標準規格に準拠したアルゴリズムでは必要となった補助的なグローバル変数が不要になり、パターン照合を行う際に必ず必要になるグローバル変数だけ用意すればよくなった。アルゴリズムが単純になったことにより、実質的な計算量も減らすことができた。POSIX 標準規格に準拠したアルゴリズムでは、最悪の場合の計算量は正規表現に最も頻繁に現れる文字数の二乗と文字列の長さとの積に比例していたが、欲張り戦略では正規表現に最も頻繁に現れる文字数と文字列の長さの積に実質的に比例する。欲張り戦略にもとづく正規表現と文字列との間のバックトラックのないパターン照合アルゴリズムとしては、Frischeと Caldelli による研究が有名である。今年度得られたアルゴリズムは、この研究と比べて以下の利点がある。まず、計算量の実質的なオーダは変わらないものの、後者は正規表現に現れるシンボル数と文字列の長さの積なので、本研究の結果の方が同じ性能のコンピュータで実行したときにはほとんどの場合定数倍の速度向上効果が得られる。また、後者は文字列そのものが正規表現と一致する場合にだけ成功するが、本研究の結果では部分文字列が正規表現と一致する場合にも成功するので、彼らのアルゴリズムよりも適用範囲が広い。

(3) 月・惑星探査データのための XML にもとづくデータフォーマットの設計
本研究は、月・惑星探査データにおいて用いられているデータフォーマットについて、情報科学的な観点から再検討を行い、より使いやすい形を模索するとともに、その試験的な実装と検証を目的として行った。月・惑星探査データのフォーマットとしては、PDS (Plaranetary Data System) と呼ばれるシステムで採用されているフォーマットが 30 年もの間利用されてきている。しかし、このフォーマットには、データの構造化がなされていない、データの改版に対応していない、異なる種類のデータを混在させることができない、など設計の古さに伴う種々の問題が生じている。また、ファイルはテキストとバイナリの混在であるため、データをチェックする際にはいちいちヘッダをみるかファイルを直接オープンする必要があり、大量のデータを抱える現状では解析効率を著しく低下させる要因となっている。本研究では、 PDS

に代わるものとしてXMLにもとづく新たなデータフォーマットを設計し、それをXPEF(XML-based Planetary Exploration data Format)と名づけた。 XPEF は、XML を基盤とし、Open Office XML Formats にみられるような、XML によるデータ記述とバイナリ(またはテキスト) の実データの組み合わせをアーカイブ化する形式により構成される。元データ情報ファイルのフォーマット(XML ボキャブラリ)を設計するために、PDSで使われるフォーマットで既存の月・惑星科学観測データがどのように記述されているかを調査し、それにもとづいてXML フォーマットに必要な要素と属性の洗い出しを行った。ここで用いたのは、NASA により公開された月探査機クレメンタインが撮影した月の全球をカバーする画像データである。この画像を納めた PDS ファイルのヘッダ部分特に重要なもの、および、他の観測データにも共通に存在するであろうものを抽出し、XPEFで用いるXML フォーマットを設計した。また、PDS データを XPEF のアーカイブデータに変換するための方法についても検討を行った。

５．主な発表論文等
（研究代表者、研究分担者及び連携研究者には下線）

〔雑誌論文〕（計３件）

Taro Suzuki, Jun'ya Terazono and Takafumi Hayashi. Design and Implementation of New Generation Data Format for Lunar and Planetary Exploration. Transaction of JSASS, Aerospace and Technology Japan, Vol.10, Tk_15-Tk_18 (2012) [査読有]

奥居哲、増田拓也、藤田佳宏、鈴木大郎. 決定性有限オートマトンによる正規表現の貪欲な照合. 情報科学リサーチジャーナル, Vol.20, pp.97-104 (2012) [査読無]

奥居哲. メモ化について. 情報科学リサーチジャーナル, Vol.21, pp.67-68 (2014) [査読無]

〔学会発表〕（計１件）

Taro Suzuki, Jun'ya Terazono and Takafumi Hayashi. Design and Implementation of New Generation Data Format for Lunar and Planetary Exploration. The 28th International Symposium on Space Technology and Science, 2011-k-25:1-4 (2011) [査読無]

〔図書〕（計０件）

〔産業財産権〕
○出願状況（計０件）

○取得状況（計０件）

〔その他〕
テクニカルレポート
Satoshi Okui and Taro Suzuki. Disambiguation in Regular Expression Matching via Position Automata with Augmented Transitions, Technical Report No.2013-002, The University of Aizu (2013)

６．研究組織
(1)研究代表者
鈴木　大郎（SUZUKI TARO）
会津大学コンピュータ理工学部・上級准教授
研究者番号：９０２７２１７９

(2)研究分担者
奥居　哲（OKUI SATOSHI）
中部大学工学部・准教授
研究者番号：００２８３５１５