様式Ｃ－１９

科学研究費助成事業（科学研究費補助金）研究成果報告書

平成２５年５月３１日現在

機関番号：３３９１０
研究種目：基盤研究（C）
研究期間：2010～2012
課題番号：２２５０００１９
研究課題名（和文）Glushkov オートマトンの拡張に基づく POSIX 正規表現の効率的照合手法

研究課題名（英文）An efficient POSIX regular expression matching via Glushkov automata with augmented transitions

研究代表者
奥居　哲（OKUI SATOSHI）
中部大学・工学部・准教授
研究者番号：00283515

研究成果の概要（和文）：正規表現のパターン照合はテキストデータの検索・加工に欠かせない技術であり，広く利用されている．しかしながら，POSIX 標準規格が定める正規表現のパターン照合（部分式の捕獲も考慮した最左最長一致による照合）については効率的な照合手法がほとんど知られておらず，バックトラックを用いて試行する非効率的なアルゴリズムが主に用いられてきた．本研究では，Glushkov オートマトンの独自拡張を用いたより効率的な照合手法を提案し，その正当性の証明を与え，計算量を解析し，試験実装により実際のいくつかの事例で著しい効率改善を確認した．

研究成果の概要（英文）：Regular expression matching is crucial for many applications such as text processing. Although POSIX 1003.2 standard requires (sub-) matching to follow the leftmost-longest rules, almost none of existing implementations, which rely on backtracking, are responsible to the requirement; they follow the greedy semantics, an alternative way more suited for backtracking, instead. This study has offered, based on a slight extension of Glushkov automata (aka, position automata), a new and more efficient matching algorithm that accommodates POSIX regular expression matching. We have given its rigorous correctness proof, and the exact computational cost at worst case. Our experimental implementation has shown significant improvements on efficiency in some practical cases.

交付決定額

（金額単位：円）

	直接経費	間接経費	合　計
２０１０年度	1,200,000	360,000	1,560,000
２０１１年度	1,000,000	300,000	1,300,000
２０１２年度	900,000	270,000	1,170,000
総　計	3,100,000	930,000	4,030,000

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：正規表現，POSIX，パターン照合，オートマトン

１．研究開始当初の背景

現在用いられている正規表現の照合手法は大きく２つに大別される．ひとつは形式言語理論に基づき正規表現から非決定性有限オートマトン(NFA)を経由して最終的に決定性有限オートマトン(DFA) を構成する(あるいはこの構成を実行時に模倣する)手法（Thompson 1968 他）て□,もうひとつはバックトラックによる解探索を行う手法（Spenser 1986 他）て□ある．一般に計算コストが低いのは前者て□あるか□，POSIX1003.2 て□定められた正規表現に対しては後者に基つ□く実装か□大半を占めてい

る．これは第一に，正規表現の後方参照（英語式には前方参照 forward reference）を実現するのに適しているからである．第二に，部分式の捕獲（部分式とマッチする文字列の利用）が容易に実現できるからである．一方，バックトラックで POSIX 最左最長規則を忠実に実現するには原理的に全解探索が必要であり，照合にかかる時間計算量が最悪の場合指数的に増大する．よって，後方参照を用いない正規表現に対しては前者の手法を用いることが望ましい．実際，後方参照は意味論上の曖昧さから積極的に用いられることは少なく，後方参照を用いない正規表現の効率的照合は実用上重要である．

前者の手法を POSIX の正規表現の照合に適用する際に最も問題となるのは，POSIX 最左最長規則に準拠した照合の曖昧さ除去(disambiguation) をどのように実現するかという点である．最左最長規則は正規表現の部分式の構造に依存しているが，DFA を構築する過程で正規表現の部分式に関する情報が欠落するからである．そこで，Laurikari （2000, 2001）は，中田と佐々（1991）の意味論的遷移規則付き有限オートマトン（automata with semantic rules）のアイデアに基づきタグ付き有限オートマトン（tagged automata）を導入し，POSIX 最左最長規則のバックトラックに依らない効率的実現が可能であることを主張した．また，この手法に基づいて TRE と呼ぶ POSIX 正規表現ライブラリを実装した．しかしながら，Laurikari の手法の正当性は示されておらず，POSIX の最左最長意味論に適合していないことが後に指摘された．Pike による Plan9 の正規表現実装も Laurikari と同 様の手法に基づいており，POSIX 最左最長規則には従っていない．Kuklewicz は Haskell（GHC）正規表現ライブラリ TDFA において軌道（orbit）と呼ぶ独自の工夫を導入し，Laurikari の手法の難点を回避したが，TDFA の手法の正当性については一切議論されていない．また，軌道の比較毎に入力に比例する計算時間を要するため，Laurikari の手法と比較して実行効率の低下が避けられない．

一方，本研究者らは近年文脈の捕獲が可能な正規表現の木言語拡張の研究(科研費基盤 (C)16500014，研究代表者：奥居）や Antimirov による partial derivative（Brzozoswki derivative の改良手法のひとつ）を拡張したパターン照合手法の研究に取り組んできたが，この過程で Glushkov オートマトン（ポジションオートマトン）を拡張することで Laurikari の手法の問題点を克服するという着想を得たものである．

２．研究の目的

POSIX の最左最長規則にしたがう正規表現の効率的な照合を実現しようとする上で，現在用いられている諸手法には以下のいずれかの問題がある．

（１） 現在主流を占めるバックトラックを用いる手法は，最左最長照合に適していない．原理的に全解探索をおこなう必要があるからである．
（２） 伝統的な（教科書的な）DFA を用いる方法は，正規表現全体の効率的な照合には適しているが，部分式の捕獲をどのように実現できるかが明らかにされていない．DFA を構築する過程で，もとの正規表現の部分式の情報が失われてしまうからである．
（３） タグなどの付加情報を利用して上記（２）の問題を克服しようという手法は最左最長規則を正確に実現していない．軌道を用いる手法は，照合の実行時に軌道の計算を必要とするため，通常の DFA と比べて実行効率が低下する．またこれらの手法は，その正当性が証明されていないという問題もある．

そこで，これらの問題点を解消した正規表現の新たな効率的照合手法を明らかにし，その正当性の証明を与えようというのが本研究の目的である．

本研究で取り組む具体的な研究課題は以下の 3 点である．

（１） POSIX の最左最長規則に基づく照合の厳密な定式化
（２） 正規表現から拡張された Glushkov NFA を構成するアルゴリズム，及びその NFA 上で最左最長規則にしたがう効率的な曖昧さ除去を実現するアルゴリズムの解明
（３） 前項の NFA から さらに DFA を導出するアルゴリズムとその効率的実装手法の解明
（４） 本手法の最左最長照合以外の照合への応用，特に現在最も一般的な貪欲照合への応用

また,各課題で明らかになったアルゴリズムの試験実装もおこない実際の効率改善を確認する．

３．研究の方法

本研究の目的を達成するために以下のような段階を踏む．

（１） 非形式的に与えられている最左最長

規則の意味を明確に形式化

構文解析木を用いた POSIX 最左最長規則の厳密な定式化をおこなう.POSIX の最左 最長規則は,自然言語（英語）で非形式的に与えられているため議論か不明確になり数学的に正確な議論かできない.そこで,正規表現によって受理される語の構文解析木をランク不定木で表現し,この構文解析木上の順序関係を用いて,POSIX 最左最長規則を厳密かつ簡潔に形式化する.これは,この後のアルコリスム構築,正当性証明の議論を厳密に進めるために不可欠な作業である.

（２） Glushkov NFA の独自拡張に基づく基本的な照合アルゴリズムの解明と,その正当性を証明および計算量の解析

本研究で中心的な役割を果たす拡張された Glushkov NFA の定式化と,与えられた正規表現を拡張された Glushkov NFA に変換するアルゴリズムの構築に移る.これについては,予備的研究において基本的な検討は既に終了し大きな問題なく実現できる見込みか得られているので,直ちに細部の検討に入るとともに,正当性の証明の検討に入る. これは以下のように進める.(1)与えられた正規表現の Glushkov NFA を元に正則変換器(regular transducer)を定義する.この変換器は入力された語をその構文解析木(の逆ポーラント記法に類する表現)に変換するものである.(2)与えられた正規表現に対して,それか受理する語の構文解析木の集合と,この変換器によって生成される構文解析木の集合か一致することを正規表現の構造に関する帰納法で証明する.(3)この変換器から導出 される拡張された Glushkov NFA における時点表示(様相;configuration)で,変換器の生成する構文解析木(の逆ポーラント表現)か与えられることを示す.

さらに,本アルコリスムの計算量について検討する.本アルコリスムで計算コストに最も関係するのは,各探索ステップにおけるパスの順序比較である.そこで,合流しないパスに関しても事前にある 程度比較を行っておくことで計算量を削減する手法についても検討する.

（３） DFA への変換手法の解明

項目（２）で明らかにした NFA ベースの照合アルゴリズムをもとに DFA への変換を可能にする研究をおこなう. 通常の DFA は遷移間の優先順位の情報を表現するのに十分ではないため, 適切な DFA の拡張を検討する. 具体的には通常の DFA が NFA 状態の集合を DFA 状態とみなすのに対し, NFA 状態の列（リスト）を DFA 状態とみなす変更を導入する. また, 各々の DFA 遷移に対し, 対応する NFA 遷移を関連づける.

各項目の遂行にそって必要に応じて試験的な実装をおこない実際の効果を検証する. またパーサ・コンビネータの手法を用いた実装手法についても検討する.

本研究の主目的は最左最長規則を効率的に実現するオートマトンに基づく手法の解明であるが, 最左最長以外の照合の意味論への転用・応用の可能性についても随時検討する. 特に現在, 最左最長照合と並んで広く用いられている貪欲照合（greedy matching）への転用・応用が重要であり, 十分に検討する.

４．研究成果

本研究の主要な成果は以下の通りである.

（１） 部分式の捕獲も含めた最左最長規則の形式化

POSIX 1003.2 に与えられている最左最長規則は自然言語で非形式に記述されている. これは実装間の非互換性の一因となる. また, 照合アルゴリズムを正確に記述し照合アルゴリズムの正当性を厳密に証明しようとする際の妨げになる.

そこで本研究では, 形式言語理論やコンパイラ構成で用いられる構文解析木（導出木）の概念を利用することで, 最左最長規則の簡潔で厳密な定式化を与えた. 具体的には, 照合の解を表現するために「正形（canonical)」な解析木と呼ぶ概念を導入し, 与えられた照合問題に対して正形な解析木が有限個しか存在しないことを示した. さらに正形な解析木の間に厳格全順序を導入し, この順序のもとで最小となる解析木を用いて最左最長解を定義した.

（２） Glushkov NFA（ポジション NFA)の拡張に基づく正規表現の効率的な照合アルゴリズムの考案およびその計算量の解析

バックトラックを用いない正規表現の照合を実現するには，基本的には正規表現から構築される NFA 上での探索を並列しておこなえばよい（すなわち部分集合構成を照合実行時におこなえばよい)．ただし，この際に上記項目（１）で導入した順序関係を用いた枝刈りをおこない，最左最長解を与える経路が最終的に選択されるようにする必要がある．この枝刈りで問題になるのがε遷移の閉路の存在である．Frisch と Cardelli（2004）はこの問題を「problematic case」と呼び，閉路を検出するために状態数を２倍に増やした NFA を用いているが，照合効率を低下させる要因になっている．

そこで本研究では，正規表現の照合で通常用いられる Thompson の NFA の代わりに Glushkov NFA（ポジション NFA)を用いる手法を提案した．Glushkov NFA はε遷移をもたないため，上述の問題点は自動的に回避されることになる．その一方，Glushkov NFA の遷移は正形な解析木上の葉から葉への巡回を表しており，もとの正規表現の部分式の構造に関する情報を保存していないため，対処が必要になる．本研究では Glushkov NFA の遷移にもとの正規表現の部分式の情報を表すタグを付加することで部分式の情報を保持できるようにした．これは背景のところで述べた Laurikari のタグ付き NFA から着想を得たものであるが，Laurikari の方法とは異なり各遷移に複数のタグを許容する．正規表現の照合に Thompson NFA の代わりに Glushkov NFA の拡張を用いる手法は本研究の方法論的に独自な点である．

提案アルゴリズムは，基本的には入力文字列をひとつ読む毎に Glushkov NFA 上での部分集合構成を実行する．ただし，項目（１）で明らかにした順序を利用した経路比較をおこない，合流した経路の中で最も優先順位の高い経路のみを残し枝刈りをおこなう点が通常の部分集合構成とは異なる．この比較は任意の経路対毎に，それらが最初に分岐した時点以降のタグから正形な解析木上の深さ（高さ）を計算し比較するという簡単なものである．この計算は NFA の構築時に予め行うことが可能である．これは，背景のところで述べた Kuklewicz の手法において軌道の計算を照合実行時におこなう必要があるのと比べると利点である（DFA の構築については次の項目で述べる)．

さらに，提案アルゴリズムの計算コストの解析をおこなったところ，提案アルゴリズムの照合実行時の時間計算量は最悪の場合でも $O(mn(n+k))$ のオーダで抑えられることが分かった．ここで m は入力文字列の長さ（文字数)，n は正規表現中に最も多く出現する文字の出現数，k は捕獲したい部分式の数である．一般に k は比較的小さくｎのほうが支配的にはたらくので，これはおよそ $O(mn^2)$ である．これはバックトラックによるアルゴリズムの時間計算量が最悪の場合ｍに関して指数的に増大することを考えると，最悪の場合の計算オーダとしては著しい改善になっている．よって，後方参照を含まない正規表現の最左最長照合には，本アルゴリズムがより適していると考えられる．

提案アルゴリズムは C++で試験的に実装した．

（３） 貪欲照合を実現する DFA の構築

背景のところで述べたように現在の主流である正規表現の照合手法は Thompson NFA 上でバックトラック探索をおこなう方法であるが，これは最左最長照合の実現には不向きである．このため，バックトラックでより実現しやすい貪欲（greedy）な照合の意味論を採用するシステムが多い．このため貪欲照合をより効率的に実現するための手法の研究や新たな照合エンジンの開発がさかんになってきている．

そこで，本研究で提案した照合アルゴリズム（項目（２））を貪欲照合にたいしても応用できないかどうか研究をおこなった．その結果，貪欲照合を効率的に実行できるアルゴリズムを得ることができた．このアルゴリズムは，項目（２）で述べたアルゴリズムと同じく拡張された Glushkov NFA 上をバックトラックなしに並列的に探索する．この際，貪欲照合の意味論にしたがい優先順位の低い経路の枝刈りをおこなうが，

貪欲照合では経路の優先順位が途中で逆転することがないので，最左最長照合のアルゴリズムよりもさらに簡潔なアルゴリズムになる．さらに，この経路の枝刈りはコンパイル時に，すなわち照合したい文字列が与えられるよりも前にあらかじめ，おこなうことが可能である．これにより，貪欲照合を効率的に実行する DFA が得られた．

現在までに知られている最も効率的な貪欲照合のアルゴリズムのひとつは Fresch と Cardelli (1994) による２パスのアルゴリズムである．与えられた文字列を１パス目で逆方向から走査し，NFA の遷移を逆方向にたどり，あらかじめ不要な経路を発見しておく．これにより２パス目の探索ではバックトラックが不要になる．ただし，この解析は文字列が与えられてから行なう必要がある．また，無限ループを回避するために状態数を２倍に増やす必要がある．これに対して，本研究で得られた DFA パス 1 のような実行時の事前解析を必要としない．DFA は決定的な探索が可能なので，照合にかかる最悪の計算コストは文字列の長さ m に対して O(m) で済む．逆に本手法の欠点はコンパイル時に DFA を構築するための余分な計算コストとメモリのコストが必要な点である．経路の優先順位の情報を保持する必要から，本手法で構築される DFA は，伝統的な DFA と比較して状態数が増大する．

以上のことから，本手法はコンパイル時のコストはあまり気にならないが照合実行時のコストを重視するような応用（そのような利用場面は多々みられる）に適していると考えられる．

（４） 貪欲照合を実現する NFA を構築するパーサ・コンビネータの実現

パーサ・コンビネータ（構文解析器結合子は，オブジェクト指向言語や関数型言語において急速に利用が広まっているパーサ（構文解析器）の記述手段である．オブジェクトや高階関数を活用することで，複雑になりがちな構文解析器の記述をモジュール化し，保守性・再利用性を向上させる利点がある．

そこで項目（３）で述べた貪欲照合を実現する NFA ベースの照合アルゴリズムを関数型言語 Haskell を用いてコンビネータとして実現した．具体的には拡張 Glushkov NFA の非決定的な遷移をモナドとして定式化し，それに基づき正規表現の各演算子に対応するコンビネータを定義した．コンビネータを組み合わせて正規表現を記述すると，結果として照合をおこなう拡張 Glushkov NFA が得られる．コンビネータは Haskell の通常の高階関数に他ならないので，正規表現特有の記法を用いずにパターン照合器を記述できるようになった．例えば，プログラミング言語の識別子を認識するパターン照合器 **ident** は以下のように記述できる．

```
num   = letter ['0'..'9']
alpha = letter $
        ['a'..'z'] ++ ['A'..'Z']
alnum = alpha `alt` num
ident = alpha `cat` (rep0 alnum)
```

また，通常の Haskell の関数とシームレスに融合できるので，Haskell の高階関数を活用した見通しのよい記述が可能になった．

（５） 貪欲照合を行なう NFA/DFA の C++による試験実装

項目（３）で述べた手法に基づく貪欲照合をおこなう照合エンジンを C++で試験実装し DFA 化に伴う効率改善の程度を検証した．検証には R. Cox が例示した正規表現 (a+1){n}a{n} を用い n の値を 1 から 100 まで変化させ，拡張 Glushkov NFA ベースの照合エンジンと DFA ベースの照合エンジンの照合に要する計算時間を比較した．また参考として Cox らによる Google の正規表現エンジン RE2 との比較もおこなった．
その結果，n=100 の場合で Glushkov NFA ベースのエンジンは RE2 の 5 倍程度高速であるという結果が得られた．また，DFA ベースのエンジンは Glushkov NFA ベースのエンジンより約 100 倍程度高速であるという結果が得られた（n=100 の場合で）．これにより，項目（３）で述べた DFA への変換による効率改善の効果が確かめられた．

以上の研究成果のうち（１）と（２）については CIAA2010 で発表済である（ただし，正当性の証明については現在の段階のものが会津大テクニカルノート No.2013-002 にあ

る．これについては，証明の更なる改良・簡単化が済み次第，投稿の予定である)．項目（３）については情報科学リサーチジャーナル Vol20 にこれまでの結果がある．項目（４）については同 vol.19 で発表している．項目（５）の検証実験の詳細は関連する卒業研究論文に記載されている（これらの結果については投稿準備中である)．ソフトウェアについては，次節に問い合わせ先を記載している．

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）

①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，情報科学リサーチシ▯ャーナル Vol.20，pp.97-104（2013）［査読無］

②奥居 哲 ：拡張ポジションオートマトンを生成するモナド結合子，情報科学リサーチシ▯ャーナル Vol.19，pp.5-18（2013）［査読無］

③奥居 哲 ：正規表現のあいまいさ除去の効率的実現，情報科学リサーチシ▯ャーナル Vol.18，pp.95-96（2011）［査読無］

〔学会発表〕（計 3 件）

① Taro Suzuki: Matching Automaton for String Pattern with Greedy Semantics, 34th TRS Meeting, 2011-02-12, The University of Aizu, Japan

②Satoshi Okui: Disambiguation in Regular Expression Matching via Position Automata with Augmented Transitions, 15th International Conference on Implementation and Application of Automata (CIAA2010) LNCS6482 pp.231-240 (Springer) ,2010.08.12-15, University of Manitoba, Canada

③Satoshi Okui : POSIX-Compliant Disambiguation in Regular Expression Matching (Preliminary Report), Workshop on Symbolic Computation and Software Verification, 2010.04.08-09, University of Tsukuba, Japan

〔その他〕

○テクニカルレポート

① Satoshi Okui and Taro Suzuki: Disambiguation in regular Expression Matching via Position Automata with Augmented Transitions, Technical Note No. 2013-002, The University of Aizu (2013)

○ソフトウェア（試験実装）

①Haskell でモナド結合子として実現した貪欲照合のための照合エンジン
②C++で実装した貪欲照合をおこなう DFA ベースの照合エンジン
③C++で実装した最左最長照合をおこなう照合エンジン

入手・問い合わせ先：
①,②：okui@cs.chubu.ac.jp
③ ：taro@u-aize.ac.jp

６．研究組織
(1)研究代表者
奥居 哲（OKUI SATOSHI）
中部大学・工学部・准教授
研究者番号：00283515

(2)研究分担者
鈴木 大郎（SUZUKI TARO）
会津大学・コンピュータ理工学部・准教授
研究者番号：90272179