D-030

VPA を用いた XMLStream 向け CEP エンジン

CEP Engine Formed with VPA for XML Streams

内田 友樹† 松田 達希† 藤田 悟‡
Yuki Uchida Tatsuki Matsuda Satoru Fujita

1. はじめに

近年，高速ネットワーク技術の発達により時々刻々と変化していく大規模な情報をリアルタイムに取得し，活用できるようになった．次々と流れてくるデータはストリームデータと呼ばれ，このデータを分析して価値を見出す CEP(Complex Event Processing)技術に注目が集まっている．

一方，Web サービスの普及に伴い様々なインターネット上でのデータ交換フォーマットとして XML が用いられるようになってきた．XML とは，タグを用いた文書データ構造化のための汎用フォーマットであり，個々のデータの属性や論理構造をタグの中に容易に表現できる．現在，株価情報，気象情報，センサ等のデータが XML ストリームとして扱われており，これを高速に処理するための CEP エンジンが求められている．

XML ストリームを扱うシステムとして XFilter[4]と YFilter[5]が開発されている．これらは単純な検索要求にしか応えることができず，複雑なイベント処理を行う必要がある CEP には適さない．さらに複雑な要求に応えられる VPA(Visibly Pushdown Automaton)[2]ベースの XSeq[1]が提案されているが，時系列を組み込んだ検索ができるものの，単一ストリームしか扱うことができない．

本研究では，複数の XML ストリームに対して検索を行うために QLMXS(Query Language for Multiple XML Streams)と呼ばれる問い合わせ言語を開発した．この言語では検索エンジンのコアに VPA を利用している．一つの検索要求から複数の VPA が生成されることがあり，それらを組み合わせて高速に実行することが求められる．

本稿では，QLMXS の検索を実現する VPA エンジンを提案する．そして VPA の特性を生かしたエンジンの最適化を行い，高速な処理を目指す．

2. VPA

VPA はプッシュダウンオートマトンの制約を強めたものであり，スタック操作がプッシュ，ポップ，インターナルの 3 種類に分かれていることが特徴である．スタック操作が明確化されたことにより，VPA は，和集合，積集合，補集合，連結，クリーネ*に対して閉じた性質を持っている．そのため有限状態オートマトンと同等の最適化を行うことが可能である．また，スタックの特性を活かし，XML，JSON ファイルのような入れ子構造のデータをモデル化することに適したオートマトンであると言える．

† 法政大学大学院 情報科学研究科, Graduate School of Computer and Information Sciences, Hosei University

‡ 法政大学 情報科学部, Faculty of Computer and Information Sciences, Hosei University

3. QLMXS

QLMXS は，XPath やその他 CEP 向け問い合わせ言語[1][3]を参考に設計されており， XML ストリームから単純なデータの検索・抽出だけでなく, 複数の XML に跨って解析を行うための複雑な条件の記述が可能である．

Example. return stocks/stock/price >> stock_stream2
from stocks/stock/price << stock_stream

上記の Example は，流れてきた株情報データから売値のみを抽出し，新しいストリームとして出力するためのクエリである．

4. QLMXS 検索エンジン

この検索エンジンでは，問い合わせ言語 QLMXS を用い，検索エンジンのコアには QLMXS の複雑なパターンを表現することができる VPA を用いる．開発した QLMXS 検索エンジンの概要を下記の図 1 に示す．

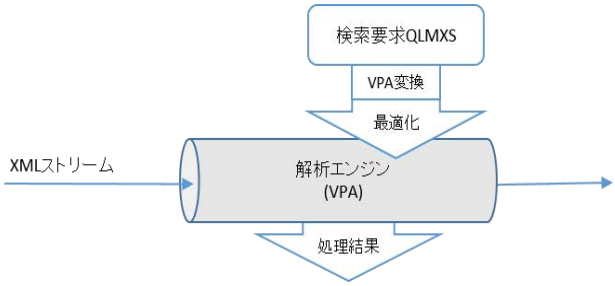

図 1 QLMXS 検索エンジン

蓄積されたデータを解析する従来のビックデータ処理とは違い，CEP では次々に送られてくるデータを高速に解析するために，事前に解析エンジンを作成しておく必要がある．本システムでは問い合わせ言語 QLMXS を基に VPA ベースの解析エンジンを生成する．この解析エンジンでは，XML の階層構成の解釈のためにプッシュダウンスタックを利用する．オートマトンは非決定性を持っており，複製可能なトークンを用いて状態遷移を繰り返し，最終状態に到達した時に検索要求が満たされたと判定する．

5. エンジンの最適化

CEP では大量に送られてくるデータをリアルタイムに処理しなければならないため，エンジンには高速性が求められる．そのためエンジンの最適化をいくつか実行する．

5.1 VPA 単体の高速化

XSeq と同様に状態数の最適化，非決定性の最適化機能を実装する．

5.1.1 状態数の最適化

XML スキーマが利用可能な場合，条件等で直接参照されていない上位の XML 要素の記述を VPA から削除する．これにより削除した部分でのトークン操作を行う必要がなくなるため大幅な高速化を図ることができる．

5.1.2 非決定性の最適化

QLMXS から VPA に自動変換する時，非決定性が生じる．VPA の非決定性部分ではトークンが大量に複製され，速度低下の原因となる．非決定性の主な要因は子要素に同じものがいくつあるか分からないために生じる状態の自己ループである．そのため同じ子要素が無いことをスキーマから確認できた場合，その部分の非決定性を失くすことで最適化を図る．

5.2 複数 VPA の最適化

一つの QLMXS から複数の VPA が生成される時や複数の問い合わせを同時に処理したい時に，解析エンジンでは複数の VPA を並行して処理することができる．各 VPA の先頭からの共通の状態での入力に対してのトークン操作は同一であると考えられるため，この共通状態を合成した新たな VPA を作成することで最適化を図る．これにより，トークンの無駄な操作が減り，VPA のスケーラビリティが向上する．

6. 検証

5.2 で述べた複数 VPA の最適化を行うことによる効果を実験により検証する．実験環境を表 1 に示す．表 2 の問合せ Q1 と Q2 を VPA に変換し，解析エンジンにより並行に処理した時と，Q1 と Q2 の VPA の共通状態を一つに合成したもの(問い合わせ式で書くならば Q3)を処理した時，そして 5.1.1 で述べた最適化をした時の速度比較を行う．

図 2 の実験結果より，共通状態を合成した Q3 では，Q1+Q2 より 1.3 倍ほど処理速度が向上した．さらに 5.1.1 の最適化を行った Q3(5.1.1)では，2 倍程の速度向上が伺える．そして，XML のデータ容量が大きくなればなるほど最適化前との処理時間の差は大きくなることが示された．

7. 考察

Q1 と Q2 は，/issue/articles/article を共通部分として持ち，これを VPA に変換すると非決定な遷移がそれぞれ 3 個生じ，処理に時間がかかる．一方，Q3 では，これらを共通化することによりトークン数を半分にし，処理時間が短くなる．処理する VPA の数やサイズが大きくなればなるほど，共通状態を合成する最適化手法は，大きな効果が現れると考える．

XML を VPA で扱う時には，オープンタグが入力されて状態を遷移する時に，遷移前の状態にトークンを複製して、対応したクローズタグがくるまで保持しておく必要がある．そして XML の構造解釈のため VPA の先頭に近い状態ほどトークンを複製して保持している時間が長くなる．そのため 5.1.1 で述べた状態数の最適化は 5.2 で述べた複数 VPA の最適化と合わせて利用することで非常に有効であると考える．このことは，表 3 の各トークン操作回数の差に顕著に表れている．

表 1　実験環境

CPU	Intel(R) Core(TM) i5-3340M 2.7GHz x 2
メモリ	8GB
XML	SIGMOD Record [6]

表 2　問合せセット

問合せ	問合せ式
Q1	/issue/articles/article/title/text()
Q2	/issue/articles/article/endPage/text()
Q3	/issue/articles/article[title/text() or endPage/text()]

表 3　データ容量 467KB 時のトークン操作回数

Q1+Q2	Q3	Q1+Q2(5.1.1)	Q3(5.1.1)
209492	106250	69384	36196

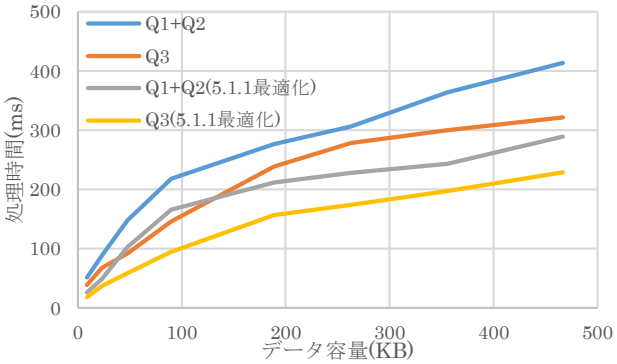

図 2　速度比較実験結果

8. まとめ

本稿では，QLMXS 検索エンジンの実装と VPA の最適化について述べた．XML を VPA で扱うための非決定性による速度低下の原因を明確にし，それを最適化することである程度改善することができた．そして実験により証明することができた．

今後の課題として，本稿では触れていない VPA の特性を生かした最適化を進めることによるエンジンの高速化と，実験で扱ったような単純な検索だけではなく，QLMXS で要求されるストリーム分割や複数の VPA を連携させる等のより複雑な処理に対応できるエンジンの開発を進める．

参考文献

[1] Mozafari B., Zeng K., Zaniolo C., “High-Performance Complex Event Processing over XML Streams”, Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, p. 253-264 (2012).
[2] Alur R., Madhusudan P., “Visibly Pushdown Languages”, Proceedings of the thirty-sixth annual ACM symposium on Theory of computing, p. 202-211(2004).
[3] Demers A. J., Gehrke J., Panda B., Riedewald M., Sharma V., White W. , “A General Purpose Event Monitoring System”, CIDR, Vol. 7, p. 412-422 (2007).
[4] Altınel M., Franklin M., “Efficient filtering of XML documents for selective dissemination of information”, Proc. of the 26th Int’l Conference on Very Large Data Bases, (2000).
[5] Diao Y., Altinel M., Franklin M., Zhang H., Fischer P., “Path sharing and predicate evaluation for high-performance XML filtering”, Transactions on Database Systems, 28(4), p. 467-516 (2003).
[6] XML Data Repository http://www.cs.washington.edu/research/xmldatasets/www/repository.html