辞書化回路と遅延ＦＦを利用したNFAパターンマッチング回路の最適化手法

渡辺友浩†，山口喜教†，前田敦司†
†筑波大学　システム情報工学研究科　305-8573　茨城県つくば市天王台 1-1-1
E-mail:　†{tomohiro, yamaguti, maeda}@ialab.cs.tsukuba.ac.jp

1. はじめに

近年、インターネットが普及するのに従って、多くの問題が発生してきている。特に、外部からの不正侵入による秘密情報の漏洩や改竄や、ネットワークのサービスを妨害するような攻撃など、セキュリティ上の問題が社会問題として取り上げられている。これらの侵入や攻撃を検知するためのシステムとして、ネットワーク侵入検知システム（NIDS）の研究開発が行われ、実用化もされている。NIDS をソフトウェアで実装した場合には、侵入検知に必須な部分であるパターンマッチング処理を行う部分がボトルネックとなり、ネットワークの高速化に NIDS の性能が追随できない事態が生じている。その問題を解決するために、パターンマッチング処理を行う部分を FPGA などの書き換え可能なデバイス上に実装する様々な方法が研究されている[1][2][3]。本報告では、そのようなハードウェア化の手法のうち、非決定性オートマトン（NFA）を用いたパターンマッチング回路の効率的な実装に関して述べる。

2. パターンマッチング回路の回路規模縮小

2.1 NFAによるパターンマッチング回路

非決定性オートマトン（NFA）を用いたパターンマッチング回路に関する研究[1][2][3]では、パターンの集合を1つの正規表現で表し、それに対する NFA を構築する方法をとっている。本研究では Snort[4]のルールオプションを図 1 に示すようにハードウェア化したパターンマッチング回路として実現する。そのルールオプションには文字列しか含まれない。そのため、正規表現のうち、扱うのは1文字の一致と連結のみであり、和や閉包は扱わない。

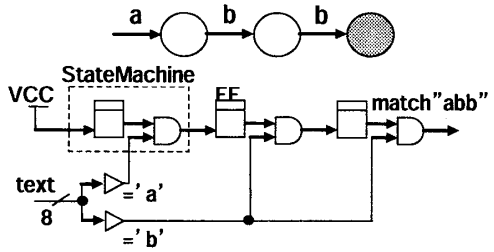

図 1、NFA と対応するパターンマッチング回路

The optimization approach of a NFA pattern matching circuit using compact circuit and delay FlipFlops by dictionary approach
Tomohiro WATANABE†, Yoshinori YAMAGUCHI† and Atsushi MAEDA†
†Graduate School of System and Information Technology, University of Tsukuba Tennoudai 1-1-1, Tsukuba-shi, Ibaraki, 305-8573, Japan

2.2 辞書化回路による回路規模縮小

NFA パターンマッチング回路では、マッチングパターンに含まれる文字列の増加に伴い、NFA のステート数が増大し、回路規模も大きくなってしまう。これを解決する手法として、辞書化回路による回路規模縮小[5]が考えられる。この手法では、データ圧縮技術で用いられる「辞書」を回路化することで回路全体を縮小化する。

例として、“abcdbcd”というパターンを圧縮すると、辞書コード(1)を“bcd”とすることで、このパターンは、“a(1)(1)”と表される。この回路は、図 2 に示すようなものとなるが、ここで新たに、辞書部のパターンマッチング結果を待つための遅延発生部が必要となる。

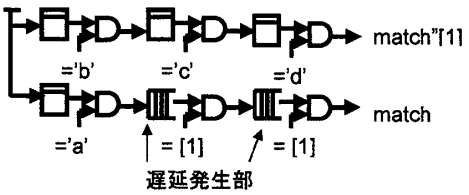

図 2、回路規模縮小を行った回路構成

この遅延発生部には、たとえば Xilinx 社の FPGA ではシフトレジスタ LUT（SRL16）を利用することができ、FF を利用してシフトレジスタを構成するよりも、回路規模の削減に役立つ。しかし、LUT を使っているため、回路自体の処理速度の低下を招く[5]という欠点がある。回路の処理速度の低下を招かないようにするためには、必要な遅延クロック分だけ FF を連結することで遅延を発生させればよいが、その場合には、遅延クロックの増大に伴い、FF の増大に伴うハードウェア量の増加が問題となる。本報告では、FF を利用しつつハードウェアの増加を抑制する手法を提案する。

3. 辞書化回路のための遅延 FF とその適用条件

FF を用いた遅延回路のハードウェア量を削減する最も単純な手法として、入力があってから遅延クロック分だけ状態遷移し、出力するという順序回路が考えられる。この回路の状態は初期状態と遅延させるための状態からなるので、3 クロック遅延であるなら、2 つの FF で表せ、7クロック遅延であるなら、3 つの FF で表せる。そのため、FF を連結した遅延回路よりもこの回路を利用した方が回路の削減につながる。図 3 にあるように、これは遅延回路に入力があったことを表す状態と、遅延するための状態からなる。遅延回路に入力があると、状態0から状態1に遷移し、その後、遅延クロック分だけ状態遷移

し、出力する。出力するときに、遅延回路に入力があった場合は、最初の状態0に戻るのではなく、入力があることを示す状態1に遷移する。この遅延回路は入力があってから、必要なクロック分だけ遅延させ、出力信号を出す。この回路を遅延順序回路と呼ぶ。

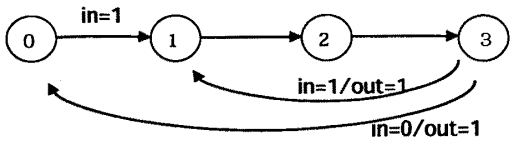

図3、　遅延回路の状態遷移

この遅延回路はFFを連結した回路に比べて回路規模は少ないという特長があるが、全ての遅延部に導入することができないという欠点がある。それは、この回路は1つのアクティブな状態しか持たないため、一般的なNFAの遅延回路に適用できないためである。すなわち、1度遅延回路に入力があると、遅延回路で設定されたクロック数の間の入力を認識できないという問題がある。しかしながら、パターンによっては、このような回路でも問題なく動作する。これらのことから、遅延回路はマッチングパターンによっては上に提示した遅延順序回路が適用できるものと適用できないものに分類できる。適用できない条件となる2つの場合の例を示す。

例えば、ルールが“ababa[cde]”として、辞書が“[cde]“である場合を想定する。入力が“…abababacde…”であると、辞書前の検査パターンである”ababa”は入力にマッチして、その2クロック後に再びマッチすることになる。提案する遅延順序回路は1度遅延回路に入力があると、遅延回路で設定されたクロック数の間の入力を認識できないので、初めの“ababa“が遅延順序回路に入力があると、3クロック以内の入力が認識できない。そのため、次の“ababa”はその入力の2クロック後に遅延順序回路に入力するので、認識できない。つまり、検知すべき入力パターンを検知できないという状況が生じる。

また、辞書前のパターン長が遅延クロック数よりも短い場合のときでも、適用できない場合が生じる。それは、例えば、ルールが“ab[cde]”で、辞書が“[cde]”の場合である。入力が“…ababcde…“であるとする。辞書前のパターン”ab“にマッチして、2クロック後に再びマッチする。上記と同様に、初めの”ab“の3クロック後は遅延順序回路は入力を認識できないために、検知すべきルールを検知できないという場合が生じる。しかしながら、上に挙げた以外のパターンでは、遅延順序回路が適用でき、その割合は全体の遅延回路の45%程度となる。

4.　評価

提案する遅延順序回路を導入したNFAパターンマッチング回路をXilinx社のISE9.2で論理合成し、FFを単純連結したものを遅延回路としたNFAパターンマッチング回路との比較を行う。

遅延順序回路を導入することで、最大周波数が20%程の低下を招いたが、5.7%のSlice数の削減をすることができた。

表1、FFを連結したものを遅延回路としたNFAパターンマッチング回路規模と最大周波数

ルール数	Slice	FF	LUT	最大周波数
128	525	994	763	431.22
256	885	1673	1218	408.497
512	1998	3773	2425	411.862
1024	4641	8602	4235	386.548
2048	9183	16956	7436	363.769

表2、遅延順序回路を導入したNFAパターンマッチング回路規模と最大周波数

ルール数	Slice	FF	LUT	最大周波数
128	549	868	959	341.88
256	917	1462	1579	325.203
512	1911	3210	3338	302.206
1024	4387	7513	6777	288.018
2048	8652	14770	12880	286.697

5.　今後の課題

上述の遅延順序回路が適応できない場合にでも、パターンによって、現在、遅延させている信号がルールと一致しないことが分かる場合がある。このような場合には、この遅延させている信号を廃棄し、遅延順序回路の状態を初期状態にすることが可能になる。つまり、遅延順序回路が適応できない場合でも、導入できるようになると考えられる。これらの検討及び評価を行いたい。

文献

[1] 栗原純, 丹羽雄平, 前田敦司, 山口喜教, 「FPGA/ソフトウェア協調処理による侵入検知システムの提案」, 信学技法, Vol.102, No.276, pp.11-16, 2002.

[2] R. Sidhu and V. K. Prasanna. Fast Regular Expression Matching using FPGAs. In IEEE FCCM 2001, pp.227-238, April 2001.

[3] B. L. Hutchings, R. Franklin and D. Carver, Assisting Network Intrusion Detection with Reconfigurable Hardware, Proceedings of the 10 th Annual IEEE FCCM'02, p.111, September 22-24, 2002.

[4] Sourcefire. Snort: The Open Source Network Intrusion Detection System. http://www.snort.org, 2003.

[5] 小野正人「ネットワークIDS向けの効率的なパターンマッチング回路の研究」筑波大学大学院博士課程システム情報工学研究科修士論文　2006年1月