

インターネットトラフィックの局所性を用いた 拡張パトリシアツリーの提案と評価

Evaluation of Enhanced Patricia Tree Taking Advantage of Traffic Locality of the Internet

80715037 明石大 (Dai Akashi) Supervisor : 西宏章 (Hiroaki Nishi)

1. 緒論

近年インターネットの発展に伴い、インターネット上の端末数が増加し続けている。端末数の増加は、ルータが管理する経路数を増大させ、ルーティングテーブルルックアップの速度の低下を招くという問題を引き起こす。今後 IPv6 やユビキタスネットワークの普及が進むにつれ、経路数は更に増大し続けることが予想され、これに対応するためにルーティングテーブルルックアップを高速化が必要がある。本研究ではルーティングテーブルルックアップの高速化に適したデータ構造を提案、評価する。

2. ルーティングテーブルルックアップ

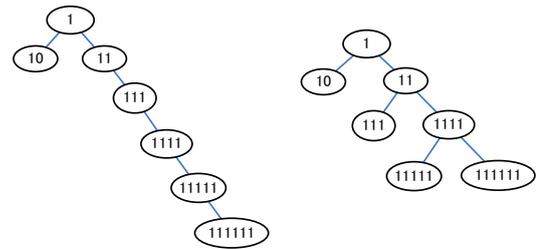
ルーティングテーブルルックアップは IP パケットヘッダの送信先 IP アドレスを基に、木構造のルーティングテーブルを探索することで行われる。ルーティングテーブルの各エントリはプレフィクス、出力先ポート情報、ネクストホップ情報を持ち、一致したエントリの出力先ポート情報に従ってパケットを転送する。

探索コストは木に含まれる冗長なノードを縮約することで削減することができる。また、IP トラフィックには、特定のエントリにアクセスが集中するといった局所性が存在し、この局所性を用いることで平均探索コストを削減できると考えられる。本研究では、従来手法により構築された木を縮約する拡張パトリシアツリーの構築アルゴリズムと、トラフィックの局所性を利用した木の提案を行う。

3. 拡張パトリシアツリー

代表的な従来研究における木として、トライやパトリシアツリーが挙げられる[1]。これらの木は検索結果を端末ノードに加えて中間ノードにも保持することで冗長なノードを削減したり、分岐を持たない余分なノードを除くことで木の縮約を図っている。しかしこれらの木はエントリの登録方向がプレフィクスの次のビットが0か1かによって決まる。そのため、テーブルエントリによっては木に偏りが生じてしまう。

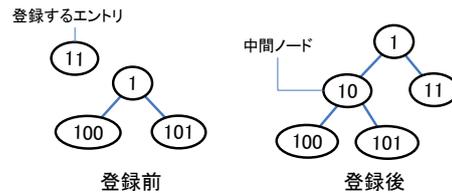
提案する拡張パトリシアツリーは、エントリの登録方向が1か0に依らず、空きポイントがある場合は必ず直下にエントリを登録するといった特徴を持つ。そのため、エントリの登録方向が偏りにくく、探索コストを削減することができる。拡張パトリシアツリーと従来手法で構築された木の比較を図1に示す。従来手法ではテーブルエントリのプレフィクスが1の連続であるため、11や111といったノードに空きポイントが存在しているにも関わらず、これらのポイントに対して登録を行えない。そのため木の深い位置にエントリを登録してしまっているが、拡張パトリシアツリーは空きポイントを利用することで木を浅くしている。



通常のパトリシアツリー 拡張パトリシアツリー

図1 拡張パトリシアツリーと従来手法の比較

また、従来手法ではプレフィクスの次のビットの値が1か0かによってエントリの登録方向が決まっていたが、拡張パトリシアツリーではプレフィクスの一致度を比較することで登録方向を決める。もしエントリを登録する空きポイントがない場合は、プレフィクスの一致度から中間ノードを新たに作成したり、木を再構築することで登録を行う。例として、各ノードのプレフィクスの一致度と、そこから導かれる構築パターンを図2に示す。プレフィクスの一致度から、新しいエントリが左子供にも右子供にも含まれず、左右の子供同士の一貫性が他のノードの一貫度よりも大きいことが分かる。そのため、左右の子供の間に中間ノードが作成され、空いたポイントにエントリが登録される。



- エントリと左(右)子供の一致度(1) < エントリのプレフィクス長(2)
- エントリと左(右)子供の一致度(1) < 左(右)子供のプレフィクス長(3)
- エントリと左(右)子供の一致度(1) < 左右の子供の一致度(2)

図2 拡張パトリシアツリーの構築例

4. トラフィック局所性を用いた拡張パトリシアツリー

前述したようにトラフィックの局所性を木の構築に利用することで、ルーティングテーブルルックアップの平均探索コストを削減できるものと考えられる。このような局所性を利用した過去研究として、Gupta らの提案した木がある[2]。Gupta らは JPEG などの圧縮アルゴリズムに用いられるハフマン符号化のアルゴリズムを二分探索木に応用することで木の構築を行っている。しかし二分探索木をベースとしているためエントリの配置に制限があり、柔軟な木の構築が行えない。本研究では関連研究として、提案手法と Gupta らの提案した木との比較を評価として用いる。

トラフィックの局所性を利用して木を構築する場合、アクセスの集中しているエントリを木の浅い位置に登録することが望ましい。そのためには対象となるエントリと、そのエントリの祖先となるノードとの間に中間ノードを作成する必要がある。よって、図3に示すように中間ノード以下のノードは1段深い位置に移動してしまう。そのため木の構築には、対象となるエントリを浅い位置に登録することで削減できる探索コストと、中間ノードを作成することで増加してしまう探索コストを算出し、トータルコストからエントリの移動にメリットがあるかどうかを求める必要がある。ここでノードRを起点とし、その子孫であるノード*i*との間に中間ノードを作成した場合のメリット $M(R)$ を以下の式で表す。

$$M(R) = P(i) \times (H(i) - 1) - (P(R) - P(i)) \quad (1)$$

$$= P(i)H(i) - P(R)$$

ここで $P(i)$ はノード*i*に対するアクセス頻度、 $H(i)$ はノード*i*とノードRの高低差を示し、 $P(i) \times (H(i) - 1)$ はエントリを移動することで得られる削減コスト、 $P(R) - P(i)$ は中間ノードの作成により生じる増加コストを示す。

提案手法では拡張パトリシアツリーの各ノードに対して、この $M(R)$ が非負かつ最大となるノードを探索し、木の構築を行い、平均探索コストの評価を行う。

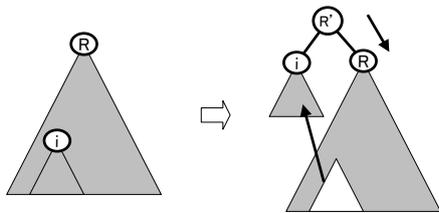


図3 トラフィック局所性を利用した木

5. 評価

まず拡張パトリシアツリーを評価する。表1にパトリシアツリーと拡張パトリシアツリーの空きポイント数、使用したルーティングテーブルのAS番号を示す。AS番号は国ごとによって発行される各組織が保有するネットワークの識別番号である。表1から拡張パトリシアツリーはパトリシアツリーと違い空きポイントを保持していないことが分かる。また拡張パトリシアツリーとパトリシアツリーを二進木と四進木で構築し、深さ毎のエントリ数を比較した。図4は二進木の、図5は四進木のエントリ分布を示す。二進木の時は分布のグラフが重なりほとんど違いが見られないが、四進木の時に、パトリシアツリーに比べ拡張パトリシアツリーのエントリが浅い位置に登録されていることが分かる。また、最大深さを15から14に削減した。これは拡張パトリシアツリーが空きポイントを持たないような木を構築するためであり、拡張パトリシアツリーが多進木に適した木であることを示している。

表1 各木に含まれる空きポイント数の比較

AS 番号	エントリ数	空きポイント数	
		パトリシアツリー	拡張パトリシアツリー
AS1221	178,925	1,908	0
AS4637	139,519	1,466	0
AS6447	273,391	2,947	0
AS65000	178,984	1,905	0

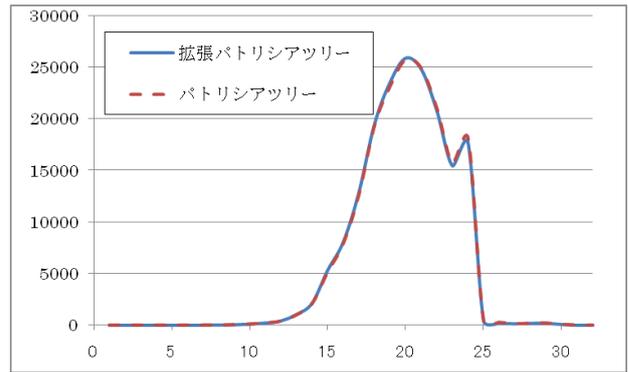


図4 二進木のエントリ分布

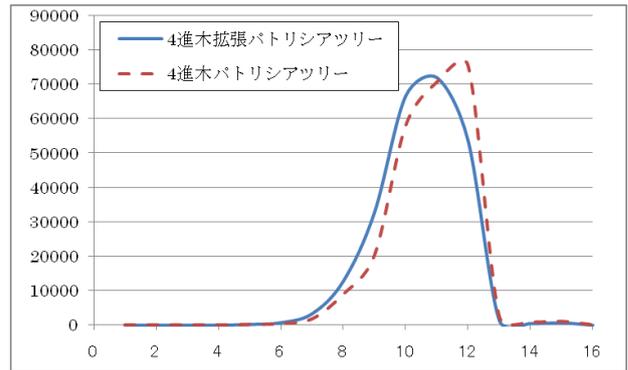


図5 四進木のエントリ分布

次にトラフィック局所性を利用した拡張パトリシアツリーを構築し、ルーティングテーブルルックアップを行い、平均探索コストを評価した。シミュレーションにはWIDEでキャプチャした100Mbpsのバックボーントレースを使用した。提案手法と通常の拡張パトリシアツリー、Guptaらの提案した木による探索コストの比較を表2に示す。提案手法は元の木よりも平均探索コストが約1.9倍程小さく、Guptaらの提案した木よりも0.99回程小さくなった。

表2 各木を用いた場合の平均探索コスト

拡張パトリシアツリー	局所性を用いた拡張パトリシアツリー	二分探索	Gupta らの提案した木
17.28	9.06	17.30	10.05

6. 結論

空きポイントを保持しないことで従来手法よりも木を縮約する拡張パトリシアツリーと、トラフィックの局所性を用いることで平均探索コストを削減する木を提案、評価した。拡張パトリシアツリーは従来手法よりも多進木に適した構築アルゴリズムを持ち、木の最大高さを削減した。また局所性を利用した拡張パトリシアツリーは、従来手法よりも平均探索コストを削減した。

参考文献

- [1] Ruiz-Sanchez, M. A, Biersack, E.W, and Dabbous, W.: Survey and taxonomy of IP address lookup algorithms, Network, IEEE, vol.15, pp.8-23 (2001).
- [2] Pankaj, G, Balaji, P., and Stephen, B.: Near-Optimal Routing Lookups with Bounded Worst Case Performance, IEEE INFOCOM, Vol.3, pp.1184-1192 (2000).