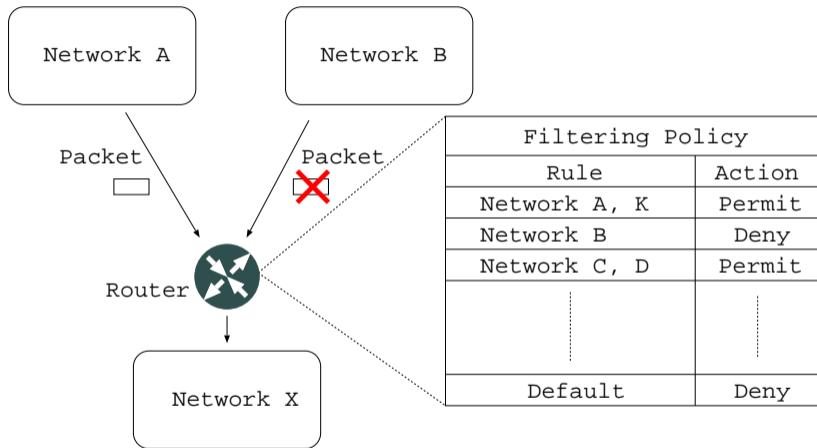


決定木を用いた Run-Based Trie の探索法

原田崇司

神奈川大学大学院 理学研究科 情報科学専攻 田中研究室

パケットフィルタリング

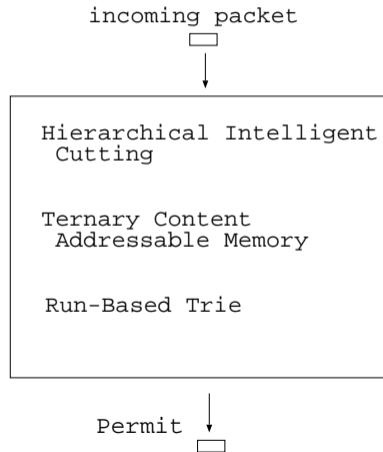
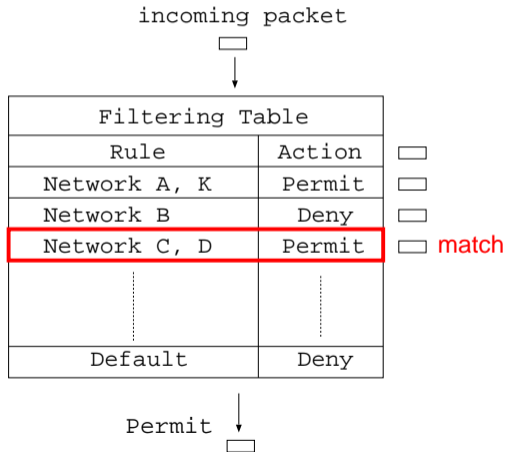


入ってくるパケットをポリシーに従ってルータで分類

パケットフィルタリングの方法

線型探索

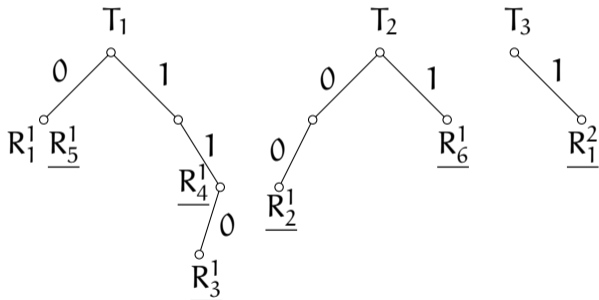
その他



Run-Based Trie (三河, 田中, 2011)

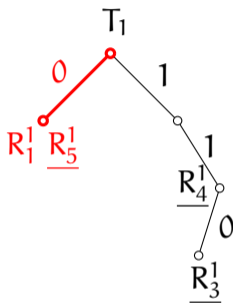
連の開始位置 i ビット目ごとにトライ T_i を構成

Filter	F
R_1	$0 * 1$
R_2	$* 0 0$
R_3	$1 1 0$
R_4	$1 1 *$
R_5	$0 * *$
R_6	$* 1 *$



Simple Search でパケット 011 を分類

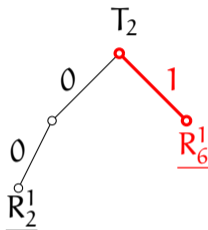
Filter	F
R_1	$0 * 1$
R_2	$* 0 0$
R_3	$1 1 0$
R_4	$1 1 *$
R_5	$0 * *$
R_6	$* 1 *$



最優先ルール	R_1	R_2	R_3	R_4	R_5	R_6
$-1 \rightarrow 5$	$0 \rightarrow 1$	0	0	0	$0 \rightarrow 1$	0

Simple Search でパケット 011 を分類

Filter	F
R ₁	0 * 1
R ₂	* 0 0
R ₃	1 1 0
R ₄	1 1 *
R ₅	0 * *
R ₆	* 1 *



最優先ルール	R ₁	R ₂	R ₃	R ₄	R ₅	R ₆
5	1	0	0	0	1	0 → 1

Simple Search でパケット 011 を分類

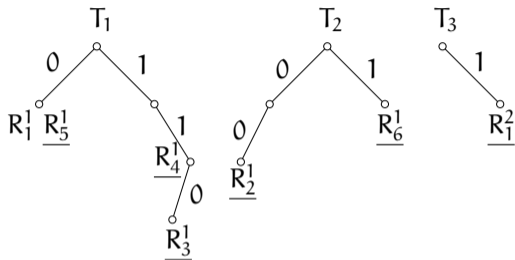
Filter	F
R_1	0 * 1
R_2	* 0 0
R_3	1 1 0
R_4	1 1 *
R_5	0 * *
R_6	* 1 *



最優先ルール	R_1	R_2	R_3	R_4	R_5	R_6
5 → 1	1 → 2	0	0	0	1	1

探索終了. パケット 011 に合致する最優先ルールは R_1

集合族 (Simple Search での T_i の辿り方の場合分け)



$$S_1 = \{\{\underline{R_1^1}, \underline{R_5^1}\}, \{\underline{R_4^1}\}, \{\underline{R_3^1}, \underline{R_4^1}\}, \phi\}$$

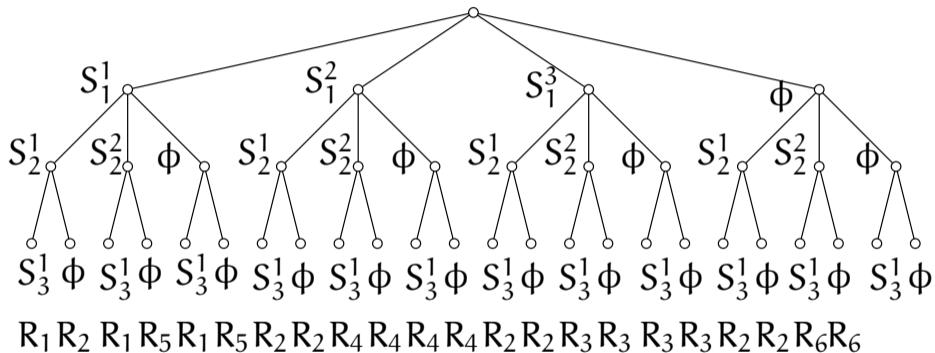
$$S_2 = \{\{\underline{R_2^1}\}, \{\underline{R_6^1}\}, \phi\}$$

$$S_3 = \{\{\underline{R_1^2}\}, \phi\}$$

集合族の直積 $|S_1| \times |S_2| \times |S_3|$ を取り, 対応するルールを付与
連の合致の組み合わせを全て列挙

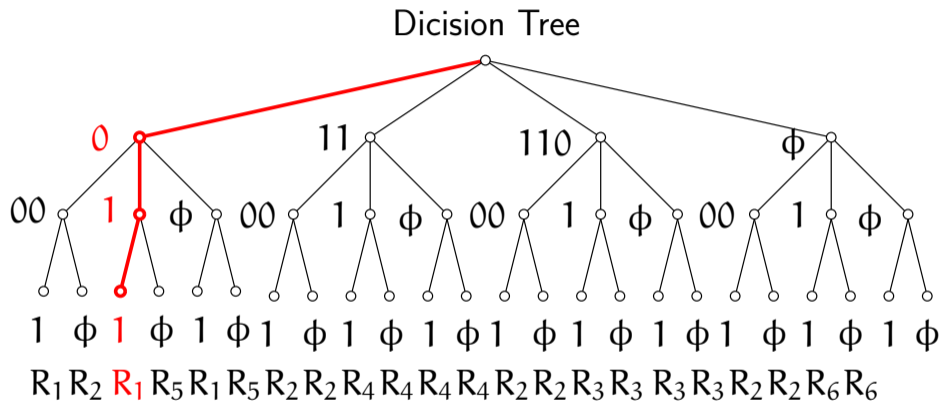
決定木

Decision Tree



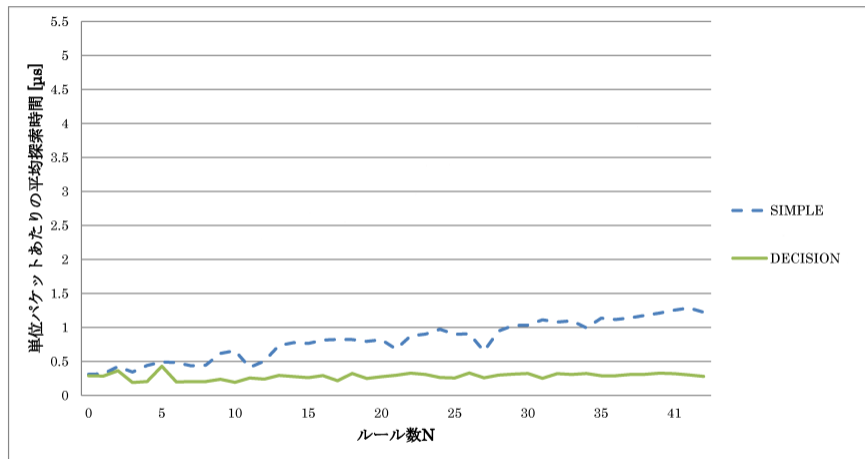
Run-Based Trie に従って決定木を辿るだけで，パケットを分類可能

決定木によりパケット 011 を探索



Run-Based Trie を用いて決定木を辿る。最優先ルールは R_1

実験結果



ビットの長さ dW 8 ビット

ルール数 N 1 ~ 45

パケット数 1000

※探索時間は1パケットあたりの平均探索時間をマイクロ秒単位で示している。

まとめと今後の課題

決定木を用いて Run-Based Trie を探索することにより、
フィルタリングルールの数 N に依存せず探索

今後の課題

- ▶ Simple Search の全ての辿り方を組み合わせるので、
決定木の空間計算量が膨大 → 枝刈りアルゴリズムが必要
- ▶ 決定木の空間計算量の算出
- ▶ プログラムを作成して、16 ビット以上での実装実験