High Speed Pattern Matching Algorithm Based on Deterministic Finite Automata with Faulty Transition Table

Jan Kaštil
Brno University of Technology
Bozetechova 2, 612 00 Brno, Czech Republic
ikastil@fit.vutbr.cz

Jan Kořenek
Brno University of Technology
Bozetechova 2, 612 00 Brno, Czech Republic
korenek@fit.vutbr.cz

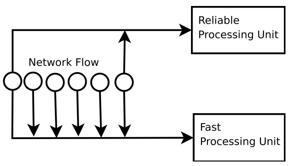
Motivation of the Research

Pattern matching can be used
- Application recognition
- Detection of the security treads

Advantages of Deterministic Finite Automata (DFA)
- Small state allows matching in network flows
- One memory access per transition

Limits of DFA
- Exponential blow-up in transition table
- Sparse transition table

Speed versus Reliability Trade-off

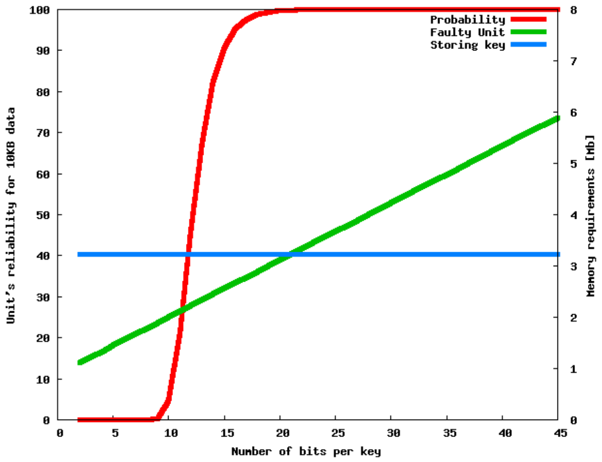


- Modern pattern matching units match only n-th packets
- Several parallel units are used for increasing throughput
  - Do not work on the flow level
- Increasing throughput of one unit is resource intensive
- As a flow speed increases few packets can elude matching

Briefly matching of every packet is better than good matching of every packet.

Experimental Results

- Typical behavior of memory savings by introducing faults
- Actual values depend on the complexity of the automaton
- Increase of memory savings with the size of automaton
- Easily parametrized

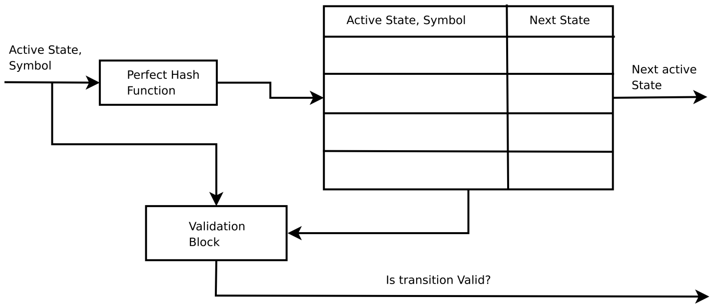


- Size of validation information for every transition can vary – X ax
- Red curve show probability of the correct matching in 10KB stream
- Blue line show memory requirements if whole combination of state and symbol is stored
- Green line is memory consumption of faulty implementation

Conclusion

- It is possible to improve throughput of the matching unit by introducing a small probability of faults per transition
- More than 20% of memory saved by 10% probability of failure in 10KB stream
- Suitable for extremely high speed

Perfect Hashing for fast Pattern Matching

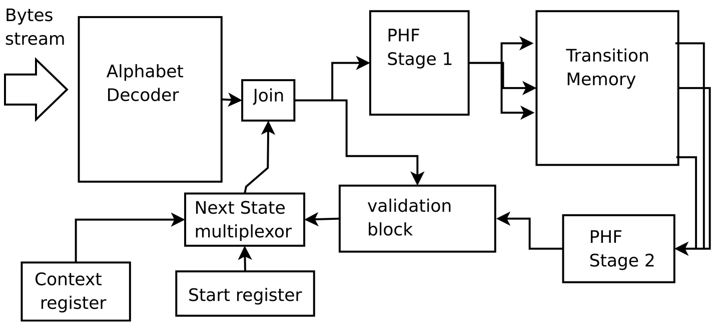


- Memory fully utilized
- Next State information is theoretically minimal for constant time access
- Validation information is surprisingly nearly optimal for sparse tables
  - Validation information takes more than two third of all memory requirements

Memory Reduction by Faults

- Possibility of incorrect transition allows reduction of validation memory
- Validation block compares hash fingerprints of transition
  - Probability of fault depends in the size of the fingerprint
  - Only false positives are possible in the validation block
  - Probability of the faulty transition is $P=\frac{1}{2^n}$ where n is the number of bits for fingerprint
- Allows addressing Speed x Reliability trade off at the flow level

Architecture of Faulty Matching Unit



- Alphabet decoder generates symbols of the multi-character alphabet
  - represents memory versus throughput trade off
- Active state and actual symbol is merged in join block
  - Active state can be start state, active state of the automaton or context information used for context switching between flows
- PHF blocks are responsible for reading the one transition from transition memory
- Validation block validate returned transition against input symbol
  - Invalid transition will result in resetting the automaton or stopping the matching process

Future work

- Evaluate the effect of the faulty matches to correct work of IDS
  - Faulty match can stop matching at different positions with correct result
  - Faulty match can accept words similar to given signatures
- Evaluate prototype design on the ComboV2 card
- Design and implement efficient method for grouping regular expression
- Design and implement efficient method for construction of multi-character automaton

evropský sociální fond v ČR
EVROPSKÁ UNIE
MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY
OP Vzdělávání pro konkurenceschopnost
INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ