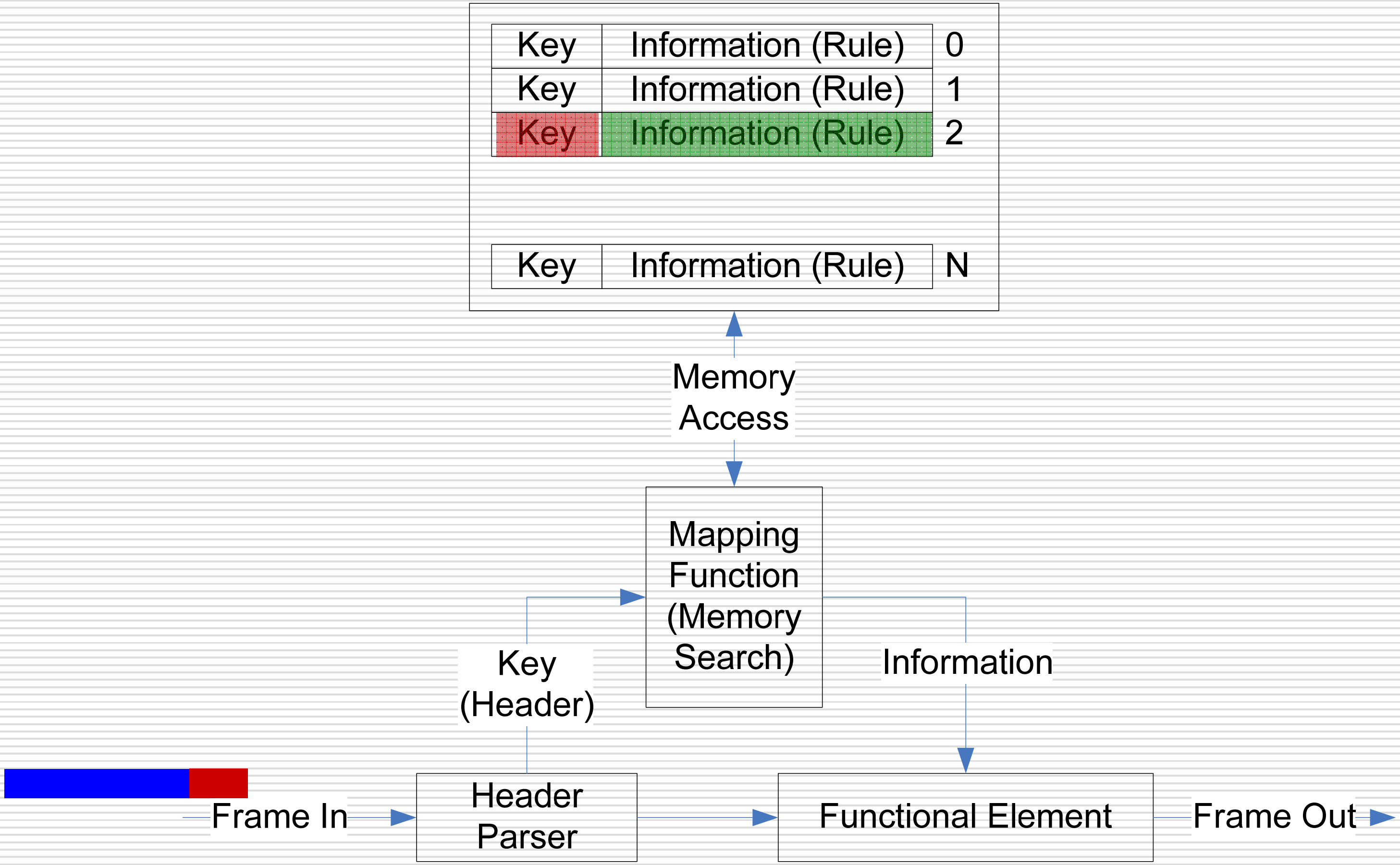
On the Impact of Caching on high Performance Packet Classifiers

Harald Widiger, Andreas Tockhorn, Dirk Timmermann
University of Rostock
Institute of Applied Microelectronics
and Computer Engineering

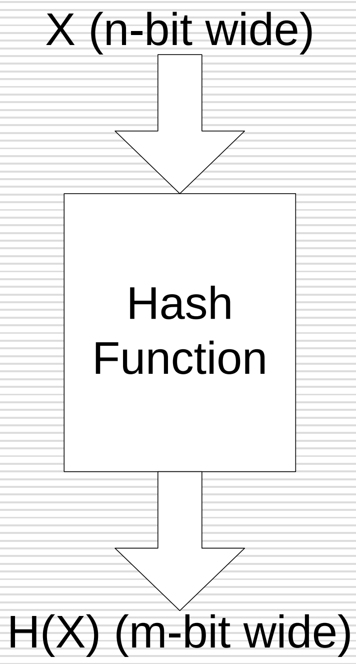
Outline

- Classification Problem
- Hash-based Packet Classification
- Cache Architecture
- Performance
- Conclusion

Classification Problem

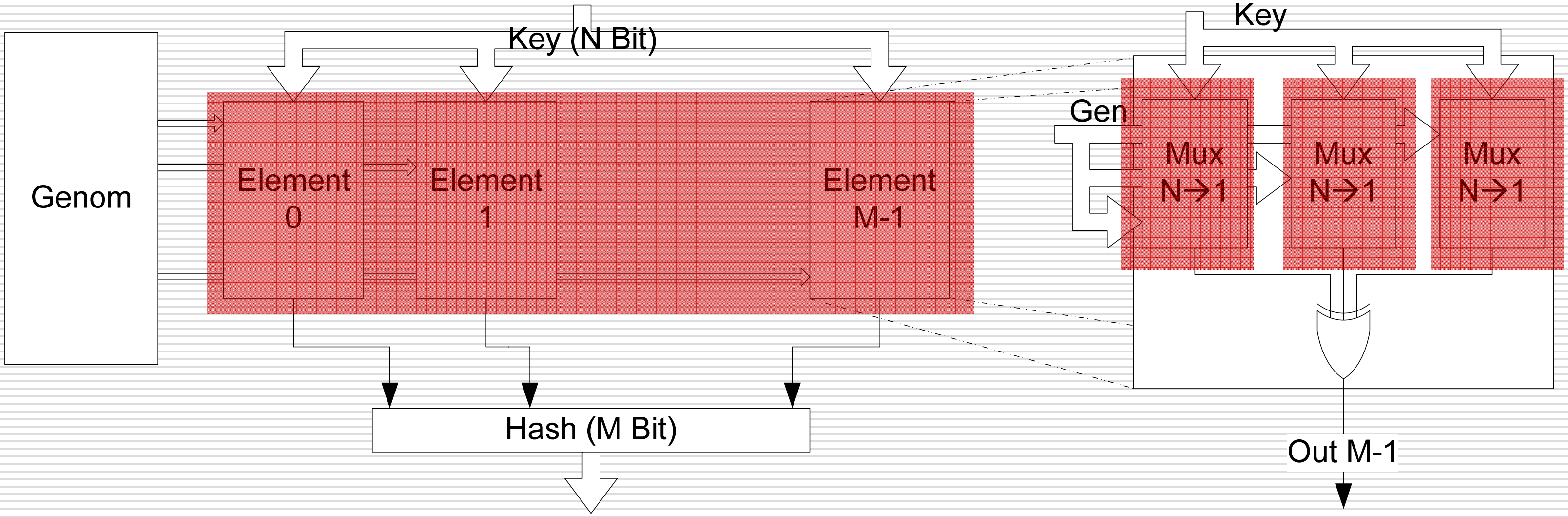


Adaptive Hashing in Hardware



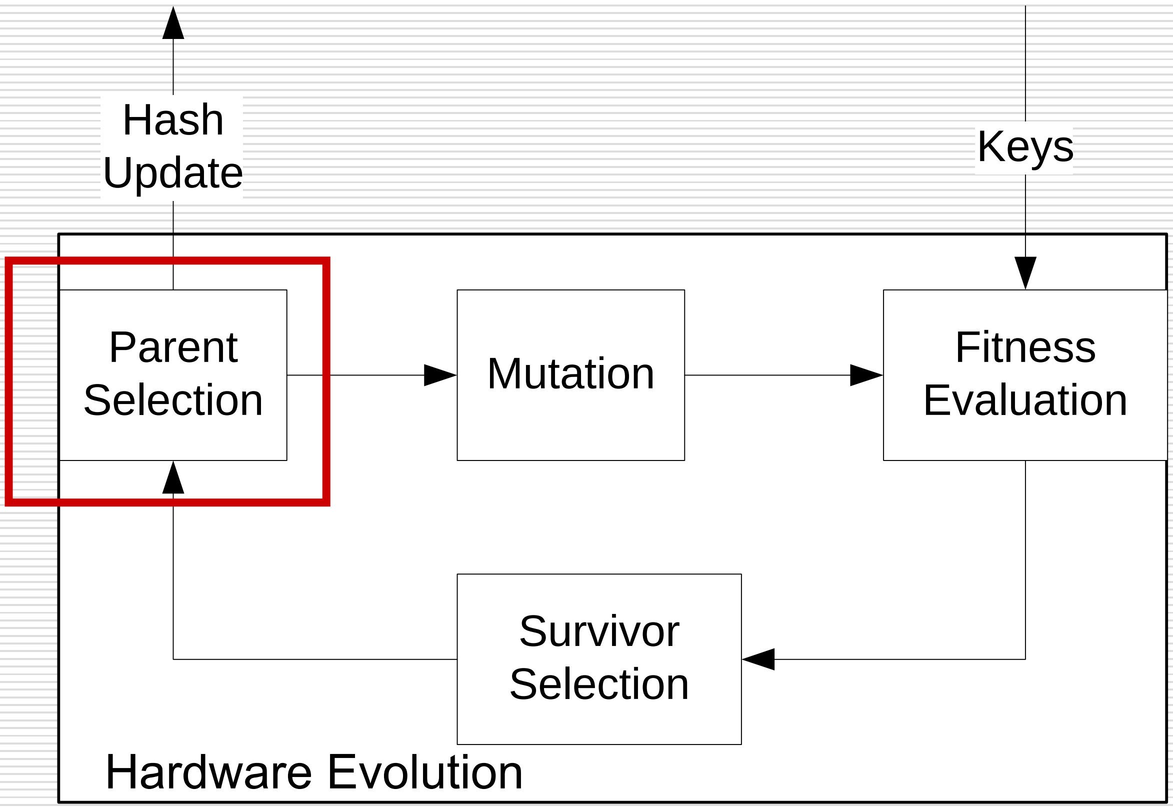
- Collision: $X \neq Y\ ; H(X) = H(Y)$
- Resolution
  - Rehashing H(H(X));
  - Linear H(X)+Prime
- Time Complexity: O(1)
- Memory Space : O(N)

An Evolvable Hash Function



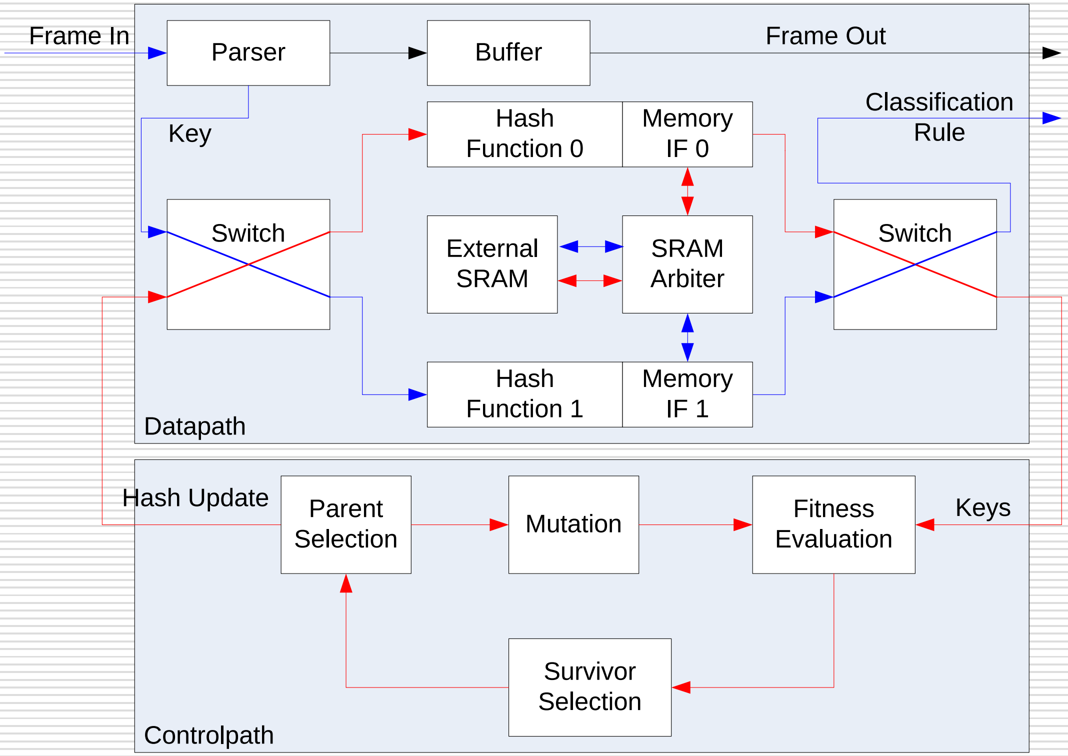
- Genome: $M \cdot 3 \cdot \log_2(N)$

Genetic Algorithm – Implementation in Hardware



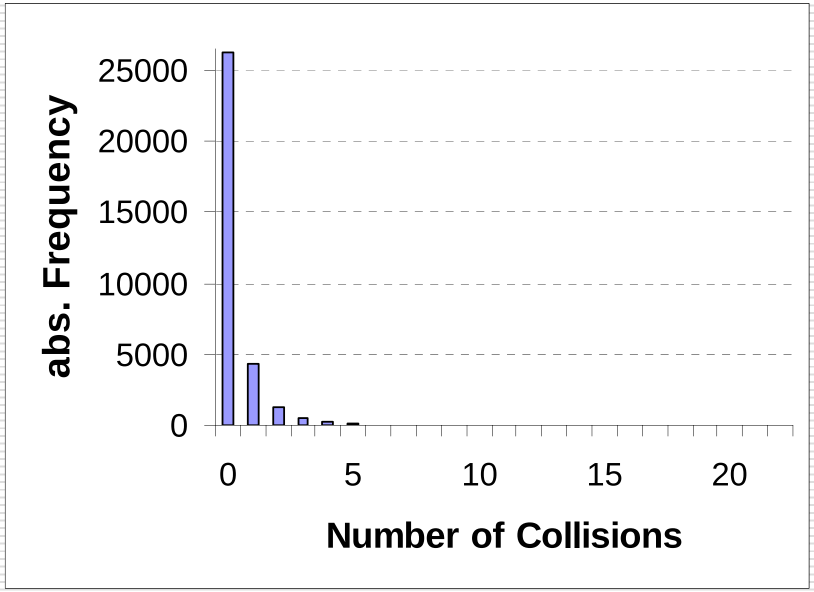
- μ Individuals
- λ Offspring
- Mutation Rate: 2/l
- Survivor Selection:μ new parents out of λ offspring and fittest old parent; (λ,μ)-elitist

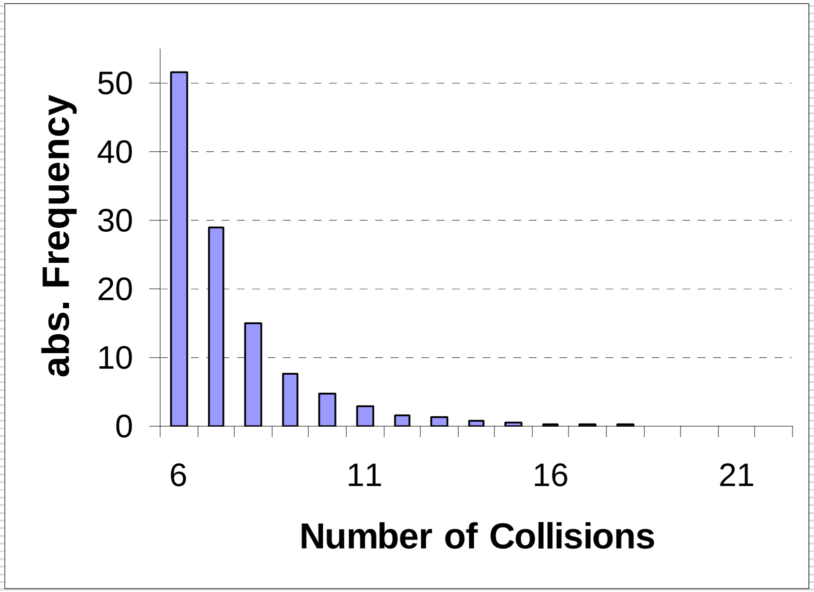
Hash-based Packet Classification - EPC



- Two parallel (data) paths
  - Packet classification
  - Hash function evolution
- Classification and Evolution work in parallel
- When finding a better hash function, data path is switched to the better one

Collision Distribution



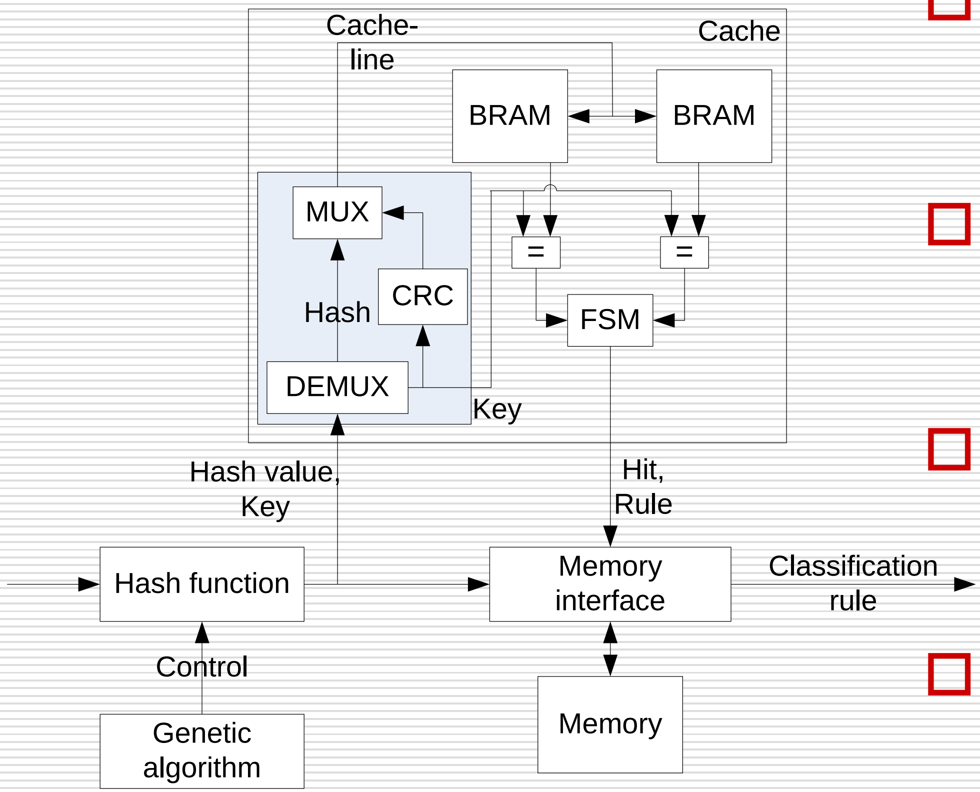


- 32768 keys
- Up to 22 collisions
- Number of keys with many collisions very small

Caching in Packet Classifiers

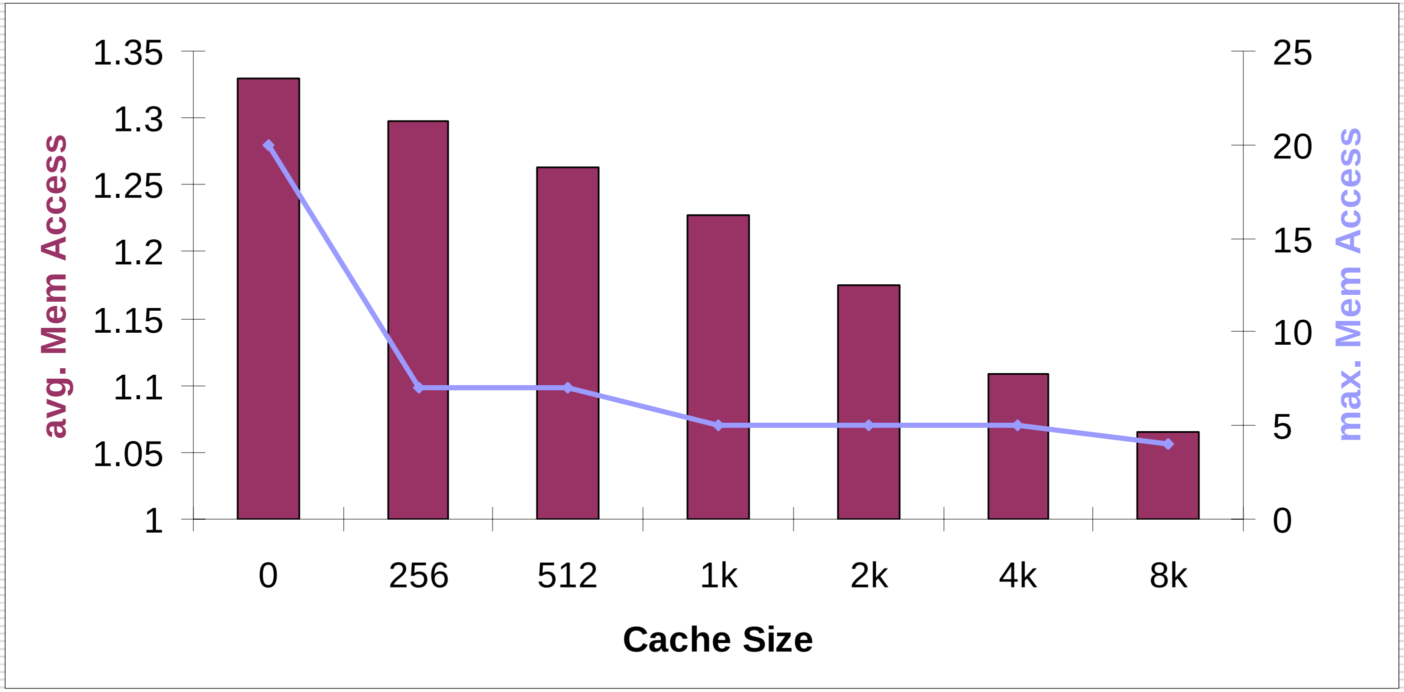
- Number of collisions = memory accesses → Cache the entries with most collisions
- Cache: memory accesses = 1 → do not cache entries that occur most often

Cache Architecture



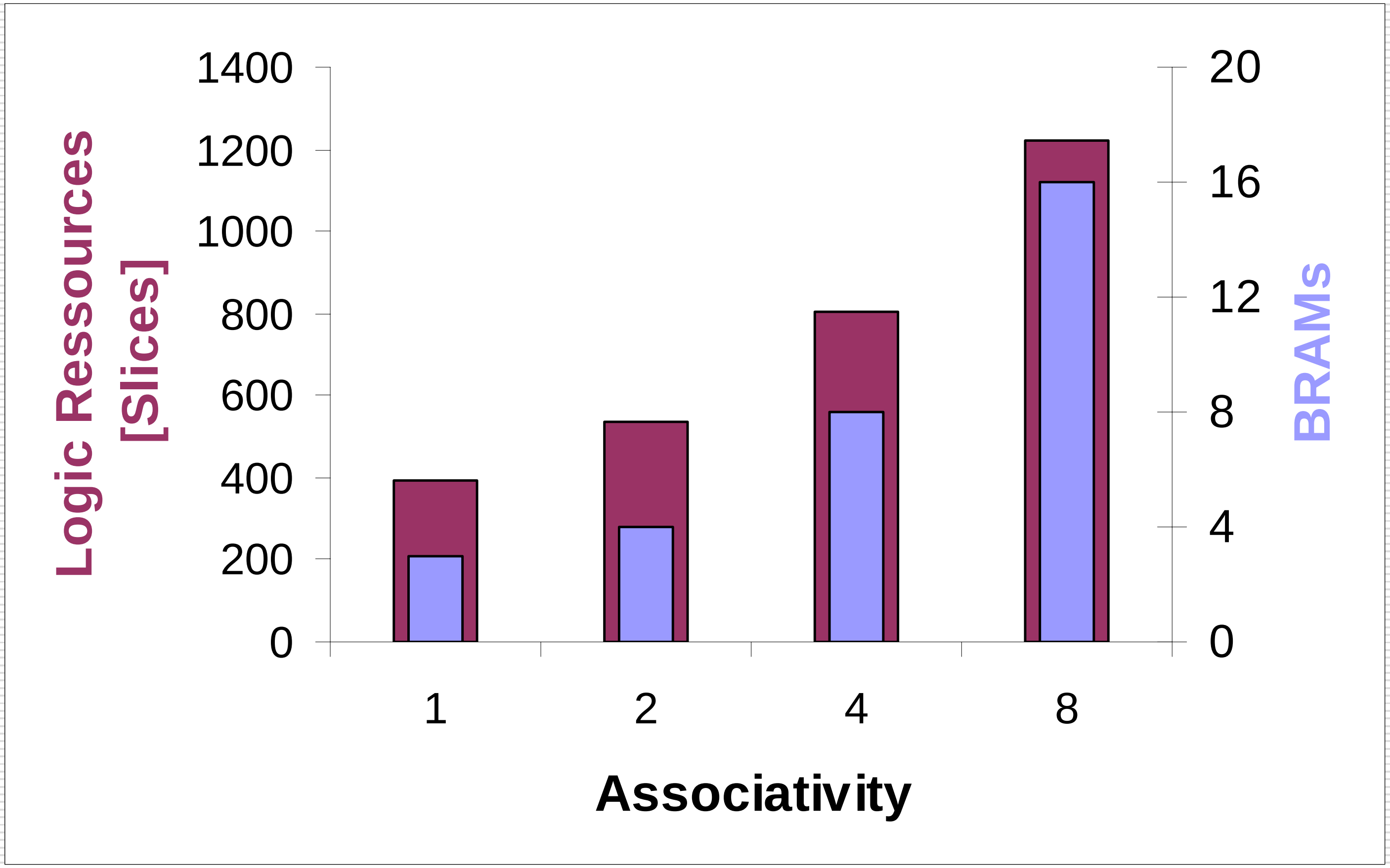
- Implemented in parallel to the data path
- Size and degree of associativity configurable
- for each degree one BRAM and a comparator
- Constraint: size/associativity = 2^N

Cache Size vs. Performance

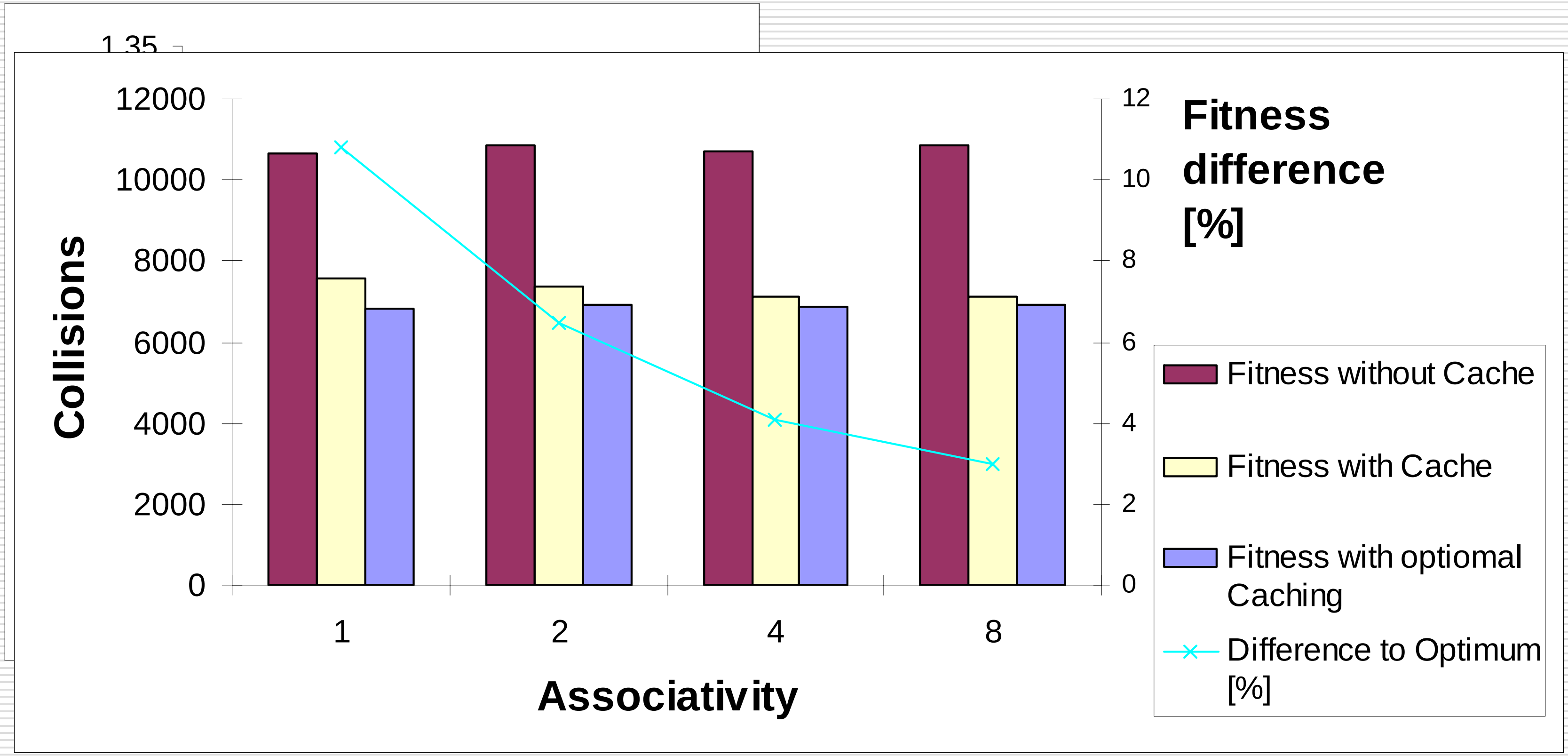


- 32768 keys
- Good results already when caching 0.78% of keys

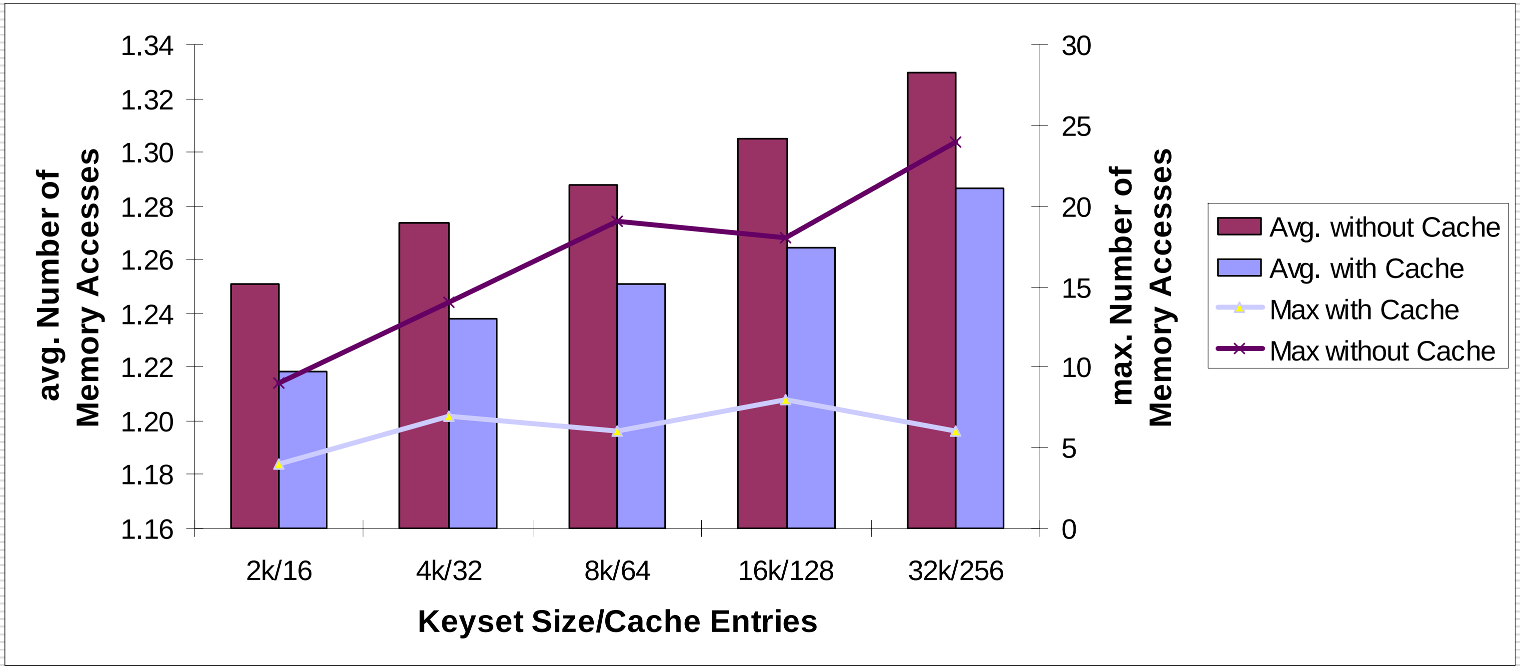
Associativity - Costs



Associativity - Performance



Performance Keyset Size



- With cache the max. number of memory accesses is largely constant

Conclusion

- When using Hashing a cache should be used to limit the max. number of memory accesses
- Even small caches are effective
- Higher associativity improves effectiveness gradually
- Even with lager keysets, it can be expected to have the max. number of memory accesses kept widely constant