Fast and powerful hashing using tabulation

Thorup, Mikkel

Published in:
36th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2016)

DOI:
10.4230/LIPIcs.FSTTCS.2016.1

Publication date:
2016

Document version
Publisher's PDF, also known as Version of record

Document license:
CC BY

Citation for published version (APA):
Fast and Powerful Hashing Using Tabulation*

Mikkel Thorup

University of Copenhagen, Dept. of Computer Science, Copenhagen, Denmark
mikkel2thorup@gmail.com

Abstract

Randomized algorithms are often enjoyed for their simplicity, but the hash functions employed
to yield the desired probabilistic guarantees are often too complicated to be practical. Here we
survey recent results on how simple hashing schemes based on tabulation provide unexpectedly
strong guarantees.

Simple tabulation hashing dates back to Zobrist [1970]. Keys are viewed as consisting of \( c \)
characters and we have precomputed character tables \( h_1, \ldots, h_q \) mapping characters to random
hash values. A key \( x = (x_1, \ldots, x_c) \) is hashed to \( h_1[x_1] \oplus h_2[x_2] \oplus \cdots \oplus h_c[x_c] \). This schemes is very
fast with character tables in cache. While simple tabulation is not even \( 4 \)-independent, it does
provide many of the guarantees that are normally obtained via higher independence, e.g., linear
probing and Cuckoo hashing.

Next we consider twisted tabulation where one character is ’twisted’ with some simple oper-
ations. The resulting hash function has powerful distributional properties: Chernoff-Hoeffding
type tail bounds and a very small bias for min-wise hashing.

Finally, we consider double tabulation where we compose two simple tabulation functions,
applying one to the output of the other, and show that this yields very high independence in
the classic framework of Carter and Wegman [1977]. In fact, w.h.p., for a given set of size
proportional to that of the space consumed, double tabulation gives fully-random hashing.

While these tabulation schemes are all easy to implement and use, their analysis is not.

This invited talk surveys results from the papers in the reference list. The reader is refered
to [8] for more details.

1998 ACM Subject Classification E.1 [Data Structures] Tables, E.2 [Data Storage Representa-
tions] Hash Table Representations, F.2.2 [Nonnumerical Algorithms and Problems] Sorting and
Searching, H.3 [Information Search and Retrieval] Search Process

Keywords and phrases Hashing, Randomized Algorithms

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2016.1

Category Invited Talk

References

1 Tobias Christiani, Rasmus Pagh, and Mikkel Thorup. From independence to expansion and

2 Søren Dahlgaard, Mathias Bæk Tejs Knudsen, Eva Rotenberg, and Mikkel Thorup. The
power of two choices with simple tabulation. In Proceedings of the 27th ACM-SIAM Sym-

* Research is partly supported by Advanced Grant DFF-0602-02499B from the Danish Council for
Independent Research.

© Mikkel Thorup.
Licensed under Creative Commons License CC-BY
36th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science
(FSTTCS 2016).
Editors: Akash Lal, S. Akshay, Saket Saurabh, and Sandeep Sen; Article No. 1; pp. 1:1–1:2
Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany
Fast and Powerful Hashing Using Tabulation


