A Collection of Thoughts on the Keccak Construction

Alexander Scheel

Iowa State University

TODO

MATH 490 | Advisor: Dr. Clifford Bergman | SHA-3 | hash_framework

1. A Series of Introductions

This paper presents the work of the author towards the completion of the requirements for the MATH 490 Independent Study course under Dr. Clifford Bergman at Iowa State University of Science and Technology. This work was an extension of the author's Honors Project under Dr. Eric Bergman and utilized the resulting hash_framework project. Additional artifacts related to this project can be seen in the keccak-attacks repository.

Within this section are a series of introductions which provide necessary background on the topics of cryptography, hash functions, the development of Keccak/SHA-3, and its structure. While certain sections can skipped if the reader has the prerequisite knowledge, hopefully all readers find the material engaging and useful. Following these introductions, this paper presents the analysis of the Keccak/SHA-3 hash function before concluding with a final evaluation and further work.

Introduction on the Topics of Cryptography. While there have been many applications of mathematics, few are as demanding and shrouded in as much secrecy as cryptography. Cryptography exists because of the fundamental need of civilizations, governments, and individuals to keep secrets secure from devoted adversaries. While modern cryptography combines the disciplines of mathematics, computer science, and computer engineering, prior to the turn of the 20th century, cryptography lacked much of its modern rigor.

Prior to the 20th century, cryptography was split into three major branches: ciphers, codes, and stenography.

Introduction on the Topics of Hash Functions. Within cryptography's collection of algorithms, few are are as useful as hash functions have proven to be to cryptographers and non-cryptographers alike. A hash function maps arbitrary length binary strings to binary strings of a fixed length.

Introduction on the Development of Keccak/SHA-3. FIPS 202 [1].

Introduction to Terminology. This section contains a collection terminology useful for discussing SHA-3 and Boolean Satisfiability.

E-mail: alexander.m.scheelgmail.com

We define $\Sigma = \{0, 1\}$ to be the alphabet.

We define the set $W = \{1, 2, 4, 8, 16, 32, 64\}$ to be the powers of two typically used in constructing the Keccak widths; $w = 64$ is standardized for use in SHA-3, though any power of two can be used.

For a given $w \in W$, we define $S_w = \Sigma^{25w}$, to be the set of all binary strings of length $25w$; these represent the possible states (which are binary strings that map onto an indexible array A described later) and each round permutation maps $S_w \mapsto S_w$.

A binary string $s \in S_w$ can be indexed directly (in a zero-indexed manner, i.e., the starting bit of s is denoted $s[0]$), or via a 3 dimensional structure of size 5 by 5 by w. This is typically done in conjunction with an uppercase letter, $A = s$, and then $A[x, y, z] = s[w(5y + x) + z]$. This is in accordance with FIPS 202 [1].

Introduction on the Structure of SHA-3. SHA-3 consists of two parts: a core permutation function, KECCAK-f, and a domain extender, the KECCAK sponge function. As discussed by the Keccak authors, additional security is given by choosing the permutation functions to be bijective, though they need not be.

2. Mathematical Properties of the Five Round Functions

In the following section, we detail various mathematical properties of the five permutation functions which make up the core round function of Keccak. In most cases, we seek to provide mathematical proofs of these properties. In all cases, we rely on external code and Boolean Satisfiability for computerized proofs of these theorems, in ways independent of the ways proved here. In a few cases, we provide references into the existing literature.

Properties of θ. In this section, we show that θ is bijective, that XOR distributes through θ, give a method for finding the inverse of θ, and give the order of θ.

Let w be a fixed power of 2. Since S_w is of finite size, it suffices to show that, $\forall x, y \in S_w$, $x \neq y \Rightarrow \theta(x) \neq \theta(y)$. Assume the hypothesis: then $x \oplus y \neq 0^{25w}$.

Lemma 2.1.

$$\theta(a) = 0^{25w} \iff a = 0^{25w}$$

Proof. This follows from the definition of θ: note that $\theta(0^{25w}) = 0^{25w}$. If $a \neq 0^{25w}$, then there exists an index set I_1 such that $\forall i \in I_1$, $a[i] = 1$, and $\forall j \notin I_1$, $a[j] = 0$. Then $\theta(a) \neq 0^{25w}$, which follows from the construction of θ. (Note that each output bit of θ is composed of the XOR of 11 values in a fixed pattern).

□

Lemma 2.2. $\forall a, b \in S_w$,

$$\theta(a \oplus b) = \theta(a) \oplus \theta(b)$$

Proof. This follows from the definition of θ: note that θ is composed entirely of XORs and that XOR is commutative and associative.

□

Combining Lemma 2.1 and Lemma 2.2, we have that:

$$
\begin{aligned}
x \oplus y = 0^{25w} &\iff \theta(x \oplus y) = \theta(0^{25w}) \\
&\iff \theta(x \oplus y) = 0^{25w} \\
&\iff \theta(x) \oplus \theta(y) = 0^{25w}
\end{aligned}
$$

and hence θ is bijective.

To construct the inverse of θ, note that $A'[x, y, z] = A[x, y, z] \oplus D[x, z]$; hence, $A[x, y, z] = A'[x, y, z] \oplus D'[x, z]$ for some $D' = D$. Since $D[x, z]$ is composed of several $C[x, z]$, where $C[x, z] = \oplus_{y=0}^{4} A[x, y, z]$, we can similarly define $C'[x, z]$ to be $C'[x, z] = \oplus_{y=0}^{4} A'[x, y, z]$. Then, we expect the inverse of θ to be of a similar form. This reduces to a linear algebra problem over boolean variables. We know that $D'[x, z] = D[x, z]$ in order to recover $A[x, y, z]$. Hence we can represent $D[x, z]$ as a series of bits of length $5 \times w$, where bit $i = z' + 5 \times x'$ is 1 if and only if $C[x', z']$ is used in the construction of $D[x, z]$. Further, we can view each of the $C'[x, z]$ as being the conjunction of three $C[x', z']$ in A, and thus can represent these as bit strings where bit $j = z' + 5 \times x'$ is 1 if and only if $C[x', z']$ is used to construct $C'[x, z]$. (That is, since $C'[x, z] = \oplus_{y=0}^{4} A'[x, y, z]$, $C'[x, z] = \oplus_{y=0}^{4} (A[x, y, z] \oplus D[x, z])$ and hence $C'[x, z] = \oplus_{y=0}^{4} (A[x, y, z] \oplus C[x', z'] \oplus C[x'', z''])$, and thus $C'[x, z] = C[x, z] \oplus C[x', z'] \oplus C[x'', z'']$, for some x, z, x', z', x'', z'' based on the definition of θ. Thus, giving each $C'[x, z]$ a constant $c_{x,z}$ for whether it is used in constructing $D'[x, z]$, we can form a system of linear equations and solve for the constants $c_{x,z}$ in each expression. Since there are $5 \times w$ variables and $5 \times w$ equations in each equation for $D'[x, z]$, this can be solved easily, yielding the inverse of θ.

Lastly, we have computed the order of the permutation θ for all w. We reproduce them here without proof; they were found by randomized search, and verified with SAT for $w = 1, 2, 4$ and 8. In general, the order is given by the expression $3 \times w$.

w	Order
1	3
2	6
4	12
8	24
16	48
32	96
64	192

Of ρ***.*** Since ρ is a simple permutation of the location of bits, it holds that $\rho(a \oplus b) = \rho(a) \oplus \rho(b)$.

It is obvious that the order of the ρ permutation is w: this is obvious from the construction of ρ.

Of π***.*** Since π is a simple permutation of the location of bits, it holds trivially that $\pi(a \oplus b) = \pi(a) \oplus \pi(b)$.

It is obvious that the order of the π permutation is 24: this is obvious from the construction of π.

Of χ***.*** It is non-trivial that the order of the χ function is a constant 4. However, this can be verified through the use of SAT to prove that χ^4 is the identity function. Hence $\chi^{-1} = \chi^3$.

Of ι***.*** Note that since ι is an XOR with a fixed value, it is obvious that ι is a bijection: for any w, for any i, and for all $x \in S_w$, $\iota(\iota(x, i), i) = x$, since $x \oplus \iota_i \oplus \iota_i = x$. Hence, ι is its own inverse and hence ι is bijective since the inverse is well defined for all $x \in S_w$.

Lastly, note that it is trivial that the order of the ι permutation is 2 by construction (due to the XOR).

Evaluation of the Orders of Composition of Permutations.

Generalizations of θ and χ.

Choice of Parameters and Ordering of Composition.

3. Marginal and Differential Properties of the Five Round Functions

Margin Properties.

Differential Properties.

Input Margin Impact on Differential Properties.

Output Margin Impact on Differential Properties.

4. Exhaustive Collision Searches

5. Fixed Point Attacks

Full Fixed Points.

Partial Fixed Points.

6. Conclusions and Further Work

7. Bibliography

1. (2015) Fips pub 202, sha-3 standard: Permutation-based hash and extendable-output functions. U.S.Department of Commerce/National Institute of Standards and Technology. http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf.