Improving the Search Algorithm for the Best Differential/Linear Trails of Bit-Permutation-Based Ciphers

Supplementary Information

Jingsui Weng[1,2], Wentao Zhang[1,2(✉)], Ting Peng[1,2], and Tianyou Ding[1,2]

[1] Key Laboratory of Cyberspace Security Defense, Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
{wengjingsui, zhangwentao✉, pengting, dingtianyou}@iie.ac.cn
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China

1 Introduction

This paper provides supplementary information specific to the paper *Improving the Search Algorithm for the Best Differential/Linear Trails of Bit-Permutation-Based Ciphers* to make our work more accessible to the reader.

Organization. The paper is organized as follows. In Section 2, we describe in detail the framework of our improved algorithm. The differential distinguishers of `KNOT-256`, `KNOT-384` and `KNOT-512` for the cluster searches, and the merging weights strategy we employ to prevent numerical overflow, are given in Section 3. Section 4 presents detailed results of our experiments for seven bit-permutation-based ciphers: `RECTANGLE`, `KNOT`, `PRESENT`, `GIFT`, `LBlock`, `TWINE` and `WARP`.

2 Algorithms

2.1 Related Equations

For ease of understanding, we present the equations involved in the pseudocode, which are explained in detail in the paper *Improving the Search Algorithm for the Best Differential/Linear Trails of Bit-Permutation-Based Ciphers*.

The whole search space of all possible r-round differential trails for an r-round bit-permutation-based SPN cipher is divided into the following subsets:

$$(\bigcup_{N_A, v, \Delta} \mathcal{D}_{r,N_A,v,\Delta}) \bigcup \mathcal{D}_{r,3,0,0}. \tag{1}$$

Let $LB_{fw}[v, N_A, \Delta]$ be equal to

$$\min_{\Delta'}(w(\Delta \xrightarrow{SL} \Delta') + FWLB[v-1, N_A, LT(\Delta')]). \tag{2}$$

Initialize the $\boldsymbol{BWLB[r_{bw}, N_A{+}1, LT^{-1}(\varDelta)]}$. When $r_{bw} = 1$, $BWLB[r_{bw}, N_A + 1, LT^{-1}(\varDelta)]$ is evaluated by

$$\begin{cases} w_{bw_{min}}(LT^{-1}(\varDelta)) & if\ CN_A(LT^{-1}(\varDelta)) \geq N_A + 1 \\ infinite & otherwise \end{cases}. \quad (3)$$

When $r_{bw} \geq 2$, $BWLB[r_{bw}, N_A + 1, LT^{-1}(\varDelta)]$ is initialized by

$$\max_{1 \leq i < r_{bw}} (BWLB[r_{bw} - i, N_A + 1, LT^{-1}(\varDelta)] + LB[i, N_A + 1, 0, 0])\ (r_{bw} \geq 2). \quad (4)$$

Initialize the $\boldsymbol{FWLB[r_{fw}, N_A, LT(\varDelta)]}$. When $r_{fw} = 1$, $FWLB[r_{fw}, N_A, LT(\varDelta)]$ is evaluated by

$$\begin{cases} w_{fw_{min}}(LT(\varDelta)) & if\ CN_A(LT(\varDelta)) \geq N_A \\ infinite & otherwise \end{cases}. \quad (5)$$

When $r_{fw} \geq 2$, $FWLB[r_{fw}, N_A, LT(\varDelta)]$ is initialized by

$$\max_{1 \leq i < r_{fw}} (FWLB[i, N_A, LT(\varDelta)] + LB[r_{fw} - i, N_A, 0, 0])\ (r_{fw} \geq 2). \quad (6)$$

Initialize the $\boldsymbol{LB[r, N_A, v, \varDelta]}$. When $r = 1$, $LB[r, N_A, 0, 0]$ $(N_A \in \{1, 2, 3\})$ is estimated by

$$sw_{min} \times N_A \quad (7)$$

When $r \geq 2$ and $N_A \in \{1, 2\}$, we initialize $LB[r, N_A, v, \varDelta]$ using Equation 8:

$$w_{bw_{min}}(\varDelta) + FWLB[r - 1, N_A, LT(\varDelta)], \quad (8)$$

or Equation 9:

$$BWLB[v - 1, N_A + 1, LT^{-1}(\varDelta)] + LB_{fw}[r - v + 1, N_A, \varDelta]. \quad (9)$$

We traverse v and $\varDelta$ to initialize $LB[r, N_A, 0, 0]$ $(N_A \in \{1, 2\})$:

$$\min(\min_{v,\varDelta}(LB[r, N_A, v, \varDelta]), LB[r, N'_A, 0, 0])\ (N_A < N'_A \leq 3). \quad (10)$$

When $r \geq 2$ and $N_A = 3$, $LB[r, N_A, 0, 0]$ is initialized by

$$\max_{1 \leq i < r} (LB[i, 3, 0, 0] + LB[r - i, 3, 0, 0]). \quad (11)$$

Update the array $\boldsymbol{BWLB}$. After completing the backward direction of the search of the subset $\mathcal{D}_{r,N_A,v,\varDelta}$, we update $BWLB[v - 1, N_A + 1, LT^{-1}(\varDelta)]$ as

$$Bc_{bw} - LB_{fw}[r - v + 1, N_A, \varDelta], \quad (12)$$

If $BWLB[v{-}1, N_A{+}1, LT^{-1}(\varDelta)]$ is updated, it is necessary to update $BWLB[i, N_A + 1, LT^{-1}(\varDelta)]$ $(v - 1 < i < r)$ as follows:

$$\max_{v-1 \leq k \leq i} (BWLB[k, N_A + 1, LT^{-1}(\varDelta)] + LB[i - k, N_A + 1, 0, 0]) \quad (13)$$

Update the array $\boldsymbol{FWLB}$. Upon completion of the forward direction of the search, we update $FWLB[r-v, N_A, LT(\Delta')]$ as

$$Bc_r - BWLB[v-1, N_A+1, LT^{-1}(\Delta)] - w(\Delta \xrightarrow{SL} \Delta'). \quad (14)$$

When $FWLB[r-v, N_A, LT(\Delta')]$ is updated, $FWLB[i, N_A, LT(\Delta')]$ ($r-v < i < r$) is updated as

$$\max_{r-v \leq k \leq i}(FWLB[k, N_A, LT(\Delta')] + LB[i-k, N_A, 0, 0]) \quad (15)$$

2.2 The Framework of Algorithm

The complete search framework of our algorithm is described in Algorithm 1. Given a bit-permutation-based cipher, a number of rounds R, and the maximum number of rounds r_{pre} involving the pre-search phase, Algorithm 1 searches for the best r-round differential trail, where $r \in \{1, 2, ..., R\}$. The best 1-round trail is generated using the minimum non-zero weight of the S-box sw_{min}, and the search starts with $r = 2$. At the beginning of the search for r rounds, the conditional best r-round trail is generated by extending the best $(r-1)$-round trail forward and backward respectively, and the initial value of B_r is set to its weight. Then the variables in array $BWLB$, $FWLB$ and LB are initialized, followed by searching all subsets divided by Equation 1. During the search, we dynamically update the lower bounds, the candidate for the best r-round trail, and Bc_r. Upon completion, the best r-round trail and weight are obtained. If $r \leq r_{pre}$, additional searches are conducted for subsets $\mathcal{D}_{r,N_A,0,0}$ ($N_A \in \{2, 3\}$) whose tightest lower bounds were not found in the previous search. This allows us to determine the exact value of all $LB[r, N_A, 0, 0]$ ($N_A \in \{2, 3\}$). In the following, we provide detailed explanations for each procedure in Algorithm 1.

Procedure `InitLBArray()`. This procedure is used to initialize the array used to aid the search process. We first initialize $BWLB[r-1, N_A+1, LT^{-1}(\Delta)]$, $FWLB[r-1, N_A, LT(\Delta)]$ and $LB[r, N_A, v, \Delta]$ ($1 \leq v \leq r$), where $N_A \in \{1, 2\}$ and Δ belongs to the set of all possible differences that have N_A active S-boxes. Subsequently, we compute the initial value of $LB[r, N_A, 0, 0]$ ($N_A \in \{1, 2, 3\}$). In the case of $r = 2$, estimation of $LB[1, N_A, 0, 0]$ ($N_A \in \{1, 2, 3\}$) is conducted. The framework of the procedure is described in Algorithm 2.

Procedure `SearchSubset12()`. Given r and N_A, this procedure is employed to search the subsets $\mathcal{D}_{r,N_A,v,\Delta}$, where $v \in \{1, 2, ..., r\}$ and Δ belongs to the set of all possible differences that contain N_A active S-boxes. The search is conducted in ascending order of v. Before searching the subset $\mathcal{D}_{r,N_A,v,\Delta}$, we use the lower bound $LB[r, N_A, v, \Delta]$ to determine whether there is a better candidate of the best r-round trail in the subset. If not, we terminate the search of it and search the next one; otherwise, a detailed search is performed on the subset. It should be noted that upon completion of the search of a direction or a whole, the corresponding variable must to be updated in time. The framework of `SearchSubset12()` is described in Algorithm 6.

Procedure ***SearchSubset3().*** This procedure is used to search the subset $\mathcal{D}_{r,3,0,0}$. Before searching, the lower bound $LB[r, 3, 0, 0]$ is used to determined whether there is a better candidate of the best r-round trail in the subset. If so, perform a detailed search for the subset. The framework of the procedure is described in Algorithm 7. For most bit-permutation-based SPN ciphers, the best trail are likely to have exactly 1 or 2 active S-boxes at a certain round. Hence, the value of $LB[r, 3, 0, 0]$ is usually greater than or equal to Bc_r, and the search is usually terminated.

Procedure ***PreSearch().*** This procedure aims to search the subsets $\mathcal{D}_{r,N_A,0,0}$ ($N_A \in \{2, 3\}$) that did not get the actual lower bound in previous searches. According to the constraints of subset, we initialize Bc_r with the corresponding approach, and use the appropriate procedure to search for the lower bounds of minimum weight of trail within it. The framework of the procedure is described in Algorithm 8, where the value of V_{add} is determined by experience.

Algorithm 1 Our improved algorithm

Input: An R-round bit-permutation-based cipher; The maximum number of rounds r_{pre} involving the pre-search;
Output: Best differential trails from 2 to R rounds

Generate a best 1-round trail that satisfies $w(\Delta x_1 \xrightarrow{SL} \Delta y_1) = sw_{min}$ and set $B_1 \leftarrow sw_{min}$
for $r \leftarrow 2$ **to** R **do**
    Generate a conditional best r-round differential trail and set Bc_r as its weight
    InitLBArray(r)
    for $N_A \leftarrow 1$ **to** 2 **do**
        SearchSubset12(r, N_A)
    end for
    SearchSubset3(r)
    $B_r \leftarrow Bc_r$
    if $r \leq r_{pre}$ **then**
        PreSearch(r)
    end if
    for $N_A \leftarrow 2$ **to** 1 **do**
        $LB[r, N_A, 0, 0] \leftarrow$ the result of Equation 10
    end for
end for

Algorithm 2 Procedure `InitLBArray()`

procedure InitLBArray(r)
  if r==2 **then**
    for $N_A \leftarrow 1$ **to** 3 **do**
      $LB[1, N_A, 0, 0] \leftarrow$ the result of Equation 7
    end for
  end if
  $LB[r, 3, 0, 0] \leftarrow$ the result of Equation 11
  for $N_A \leftarrow 2$ **to** 1 **do**
    for all $\Delta \leftarrow$ difference that has N_A active S-boxes **do**
      $BWLB[r-1, N_A+1, LT^{-1}(\Delta)] \leftarrow$ the result of Equation 3 or 4
      $FWLB[r-1, N_A, LT(\Delta)] \leftarrow$ the result of Equation 5 or 6
      for $v \leftarrow 1$ **to** r **do**
        $LB[r, N_A, v, \Delta] \leftarrow$ the result of Equation 9 or 8
      end for
    end for
    $LB[r, N_A, 0, 0] \leftarrow$ the result of Equation 10
  end for
end procedure

Algorithm 3 Procedure `FW-Round-(i)`

procedure FW-Round-(i)
  $\Delta x_i \leftarrow LT(\Delta y_{i-1})$
  if $i == r$ **then**
    $w_r \leftarrow w_{fw_{min}}(\Delta x_r),\ w_{sum} \leftarrow w_b + \sum_{k=v}^{r} w_k$
    if $w_{sum} \leq Bc_r$ **then**
      $Bc_r \leftarrow w_{sum}$
    end if
  else
    for all candidate of Δy_i **do**
      $w_i \leftarrow w(\Delta x_i \xrightarrow{SL} \Delta y_i)$
      $w_{sum} \leftarrow w_b + \sum_{k=v}^{i} w_k + LB[r-i, N_A, 0, 0]$
      if $w_{sum} \leq Bc_r$ and $CN_A(LT(\Delta y_i)) \geq N_A$ **then**
        FW-Round-(i+1)
      end if
    end for
  end if
  return to the upper procedure
end procedure

Algorithm 4 Procedure `BW-Round-(i)`

procedure BW-Round-(i)
  $\Delta y_i \leftarrow LT^{-1}(\Delta x_{i+1})$
  if $i == 1$ **then**
    $w_1 \leftarrow w_{bw_{min}}(\Delta y_1),\ w_{sum} \leftarrow \sum_{k=1}^{v-1} w_k + LB_{fw}[r-v+1, N_A, \Delta]$
    if $w_{sum} \leq Bc_{bw}$ **then**
      $Bc_{bw} \leftarrow w_{sum}$
    end if
  else
    for all candidate of Δx_i **do**
      $w_i \leftarrow w(\Delta y_i \xrightarrow{SL^{-1}} \Delta x_i)$
      $w_{sum} \leftarrow LB[i-1, N_A+1, 0, 0] + \sum_{k=i}^{v-1} w_k + LB_{fw}[r-v+1, N_A, \Delta]$
      if $w_{sum} \leq Bc_{bw}$ and $CN_A(LT^{-1}(\Delta x_i)) \geq N_A + 1$ **then**
        BW-Round-(i-1)
      end if
    end for
  end if
  return to the upper procedure
end procedure

Algorithm 5 Procedure `UpdateBWArray()` and `UpdateFWArray()`

procedure UpdateBWArray(v, N_A, Δ)
  $BW[v-1, N_A+1, LT^{-1}(\Delta)] \leftarrow$ the result of Equation 12
  for $i \leftarrow v$ **to** $r-1$ **do**
    $BW[i, N_A+1, LT^{-1}(\Delta)] \leftarrow$ the result of Equation 13
  end for
end procedure

procedure UpdateFWArray(v, N_A, Δ)
  $FW[r-v, N_A, LT(\Delta)] \leftarrow$ the result of Equation 14
  for $i \leftarrow r-v+1$ **to** $r-1$ **do**
    $FW[i, N_A, LT(\Delta)] \leftarrow$ the result of Equation 15
  end for
end procedure

Algorithm 6 Procedure `SearchSubset12()`

procedure SearchSubset12(r, N_A)
  for all $\Delta \leftarrow$ difference that have N_A active S-boxes **do**
    if $LB[r, N_A, 1, \Delta] \le Bc_r$ **then**
      $\Delta y_1 \leftarrow \Delta,\ w_1 \leftarrow w_{bw_{min}}(\Delta)$
      FW-Round-(2)
      UpdateFWArray$(1, N_A, \Delta)$
      $LB[r, N_A, 1, \Delta] \leftarrow Bc_r$
    end if
  end for
  for $v \leftarrow 2$ **to** r **do**
    for all $\Delta \leftarrow$ difference that have N_A active S-boxes **do**
      if $LB[r, N_A, v, \Delta] > Bc_r$ **then** continue
      end if
      $Bc_{bw} \leftarrow Bc_r,\ \Delta x_v \leftarrow \Delta$
      $LB_{fw}[r - v + 1, N_A, \Delta] \leftarrow$ the result of Equation 2
      BW-Round-$(v - 1)$
      UpdateBWArray(v, N_A, Δ)
      if find out a best trail for the first $(v - 1)$ rounds **then**
        $w_b \leftarrow Bc_{bw} - LB_{fw}[r - v + 1, N_A, \Delta]$
        if $v < r$ **then**
          for all candidate of Δ' **do**
            $\Delta y_v \leftarrow \Delta',\ w_v \leftarrow w(\Delta x_v \xrightarrow{SL} \Delta y_v)$
            $w_{sum} \leftarrow w_b + w_v + FWLB[r - v, N_A, LT(\Delta')]$
            if $w_{sum} \le Bc_r$ **then**
              FW-Round-$(v + 1)$
              UpdateFWArray(v, N_A, Δ')
            end if
          end for
        else
          $w_r \leftarrow w_{fw_{min}}(\Delta),\ Bc_r \leftarrow w_b + w_r$
        end if
      end if
      $LB[r, N_A, v, \Delta] \leftarrow Bc_r$
    end for
  end for
  return to the upper procedure
end procedure

Algorithm 7 Procedure `SearchSubset3()`

procedure SearchSubset3(r)
  $N_A \leftarrow 3,\ v \leftarrow 1,\ w_b \leftarrow 0$
  if $LB[r, N_A, 0, 0] \leq Bc_r$ **then**
    for all candidate of Δy_1 **do**
      $w_1 \leftarrow w_{bw_{min}}(\Delta y_1)$
      if $w_1 + LB[r-1, N_A, 0, 0] \leq Bc_r$ and $CN_A(LT(\Delta y_1)) \geq N_A$ **then**
        FW-Round-(2)
      end if
    end for
    $LB[r, N_A, 0, 0] \leftarrow Bc_r$
  end if
  return to the upper procedure
end procedure

Algorithm 8 Procedure `PreSearch()`

procedure PreSearch(r)
  for each subset $\mathcal{D}_{r,N_A,0,0}$ ($N_A \in \{2, 3\}$) that did not get the actual lower bound **do**
    if $N_A == 3$ **then**
      while $True$ **do**
        SearchSubset3(r)
        if find out a best trail within the subset $\mathcal{D}_{r,3,0,0}$ **then** break
        end if
        $Bc_r \leftarrow Bc_r + V_{add}$
      end while
    else
      Generate the conditional best r-round differential trail within $\mathcal{D}_{r,N_A,0,0}$
      $Bc_r \leftarrow$ the weight of the conditional trail
      SearchSubset12(r, N_A)
    end if
  end for
end procedure

3 Differential Distinguishers of `KNOT` for Cluster Searches

Merging Weights Strategy To mitigate the problem of numerical overflow caused by an excessive number of low-weight trails, we propose *merging the weights* during the search process. For example, if we find two trails both with a weight of w, by merging their weights, we only need to add one entry for weight $w + 1$ in the stored information, instead of adding two entries for weight w.

We performed cluster searches applying the *memoization* strategy and the *merging weights* strategy for the best trails found for `KNOT-256`, `KNOT-384`, and `KNOT-512`:

1. The 52-round differential distinguisher of `KNOT-256`:

$$\begin{aligned}&0x90000000000000000000000000000000\\&00000000000000900000000000000000\\&\qquad\qquad\to_{52}\\&\qquad\qquad 0x00000000000000000000000000000000\\&\qquad\qquad 0000000a000000000000000000000c00;\end{aligned}$$

2. The 76-round differential distinguisher of `KNOT-384`:

$$\begin{aligned}&0x00000000000000000000000000000000\\&00000000000000000000000000000000\\&000000000000000000000000090000a\\&\qquad\qquad\to_{76}\\&\qquad\qquad 0x00000007000000000000000000000000\\&\qquad\qquad 00000000000000000100000000000000\\&\qquad\qquad 00000000000000000000000000000000;\end{aligned}$$

3. The 100-round differential distinguisher of `KNOT-512`:

$$\begin{aligned}&0x90000000000000000000000000000000\\&00000000000000000000000000000000\\&00000000000000000000000000000000\\&00000000000000000000009000000000\\&\qquad\qquad\to_{100}\\&\qquad\qquad 0x00000000000000000000000000000000\\&\qquad\qquad 0000000000000c000000000000000000\\&\qquad\qquad 000000000000000a0000000000000000\\&\qquad\qquad 00000000000000000000000000000000.\end{aligned}$$

In Table 1, $w_{cluster1}$ and $t_{cluster1}$ represent the cluster weight and search time obtained by the *memoization* strategy, while $w_{cluster2}$ and $t_{cluster2}$ represent the cluster weight and search time obtained by the *memoization* strategy and the *merging weights* strategy. The cluster search using only the *memoization* strategy for `KNOT-512` resulted in numerical overflow, making it impossible to obtain a cluster weight. The *merging weights* strategy allows for consideration of trails exceeding the upper bound of weight during the cluster search. Consequently, although the search time increases with the *merging weights* strategy, we achieve more accurate cluster weights for `KNOT-256`, `KNOT-384` and `KNOT-512`.

Table 1: Experimental results of the cluster searches for KNOT.

Ciphers	r	B_r	$w_{cluster1}$	$t_{cluster1}$	$w_{cluster2}$	$t_{cluster2}$
KNOT-256	52	274	254.202	17.912s	253.806	73.764s
KNOT-384	76	402	368.248	36.575s	366.248	636.037s
KNOT-512	100	530	-	-	479.569	917.042s

- Constraints included limiting the maximum number of active S-boxes per round to no more than 3 and setting the upper bound UB_c to $B_r + 25$.

4 Experimental Results

We apply our improved algorithm to seven bit-permutation-based symmetric-key primitives: RECTANGLE, KNOT, GIFT, LBlock, TWINE and WARP, where three versions KNOT-256, KNOT-384, KNOT-512 of KNOT as well as two versions GIFT-64, GIFT-128 of GIFT are considered. The experimental results are presented in Table 2-11, where r indicates the rounds, B_r denotes the weight of the best r-round trail, and t denote the time of searching the best trail. Our experiments were performed on a PC (Intel(R) Core(TM) i9-9900 CPU @ 3.10GHz), and we used one core for each case.

It is worth noting that, due to the large state size of WARP, which has a state size of 128 bits, the search remains time-consuming even after using the number of active S-boxes to estimate the weight lower bound and enhance pruning efficiency. Consider the fact that the best r-round weight for WARP is equal to the minimum number of active S-boxes for r rounds multiplied by the minimum non-zero weight of the S-box sw_{min}. If the lower bound of the given subset is less than Bc_r, we first search for truncated trails within the subset where the product of the number of active S-boxes and sw_{min} is less than Bc_r. Then, we focus our practical searches on these truncated trails, which are instantiated with actual values to compute their weights and determine the best one. We update the candidate of the best r-round trail and Bc_r only when the weight of the best trail obtained through truncated trails is less than Bc_r.

Table 2: Experimental results of RECTANGLE

Differential property

r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	2		6	18	<0.001s	11	46	0.005s	16	71	0.008s	21	96	0.008s
2	4	<0.001s	7	25	<0.001s	12	51	0.009s	17	76	0.008s	22	101	0.007s
3	7	<0.001s	8	31	0.001s	13	56	0.013s	18	81	0.007s	23	106	0.007s
4	10	0.001s	9	36	<0.001s	14	61	0.015s	19	86	0.008s	24	111	0.008s
5	14	<0.001s	10	41	0.002s	15	66	0.012s	20	91	0.007s	25	116	0.008s

Pre-search time: 0.005s Search time: 0.134s Total time: 0.139s

Linear property

r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	2		6	20	<0.001s	11	50	0.004s	16	80	0.387s	21	108	0.198s
2	4	<0.001s	7	26	<0.001s	12	56	0.129s	17	84	0.703s	22	114	0.270s
3	8	0.001s	8	32	0.002s	13	62	0.083s	18	90	0.002s	23	120	1.044s
4	12	<0.001s	9	38	<0.001s	14	68	0.042s	19	96	0.434s	24	126	0.598s
5	16	<0.001s	10	44	0.008s	15	74	0.286s	20	102	0.043s	25	132	1.423s

Pre-search time: 0.076s Search time: 5.657s Total time: 5.733s

Table 3: Experimental results of KNOT-256

Differential property

r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	2		12	60	0.214s	23	119	0.014s	34	178	0.015s	45	236	0.010s
2	4	<0.001s	13	66	0.461s	24	124	0.011s	35	183	0.018s	46	242	0.019s
3	7	0.001s	14	71	0.475s	25	130	0.014s	36	188	0.016s	47	247	0.015s
4	10	0.001s	15	76	0.314s	26	135	0.013s	37	194	0.021s	48	252	0.011s
5	14	0.001s	16	82	0.185s	27	140	0.010s	38	199	0.019s	49	258	0.017s
6	18	0.001s	17	87	0.059s	28	146	0.014s	39	204	0.013s	50	263	0.011s
7	25	0.001s	18	92	0.015s	29	151	0.020s	40	210	0.019s	51	268	0.008s
8	32	0.001s	19	98	0.014s	30	156	0.009s	41	215	0.014s	52	274	0.014s
9	40	0.004s	20	103	0.013s	31	162	0.015s	42	220	0.013s			
10	49	0.048s	21	108	0.010s	32	167	0.015s	43	226	0.015s			
11	55	0.138s	22	114	0.016s	33	172	0.009s	44	231	0.013s			

Pre-search time: 0.058s Search time: 2.397s Total time: 2.455s

Linear property

r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	2		12	58	0.066s	23	124	0.352s	34	190	0.604s	45	256	0.954s
2	4	0.001s	13	64	0.126s	24	130	0.377s	35	196	0.630s	46	262	0.970s
3	8	0.002s	14	70	0.182s	25	136	0.398s	36	202	0.659s	47	268	1.004s
4	12	0.001s	15	76	0.232s	26	142	0.426s	37	208	0.703s	48	274	1.038s
5	16	0.002s	16	82	0.216s	27	148	0.441s	38	214	0.722s	49	280	1.074s
6	20	0.002s	17	88	0.229s	28	154	0.477s	39	220	0.782s	50	286	1.102s
7	26	0.001s	18	94	0.246s	29	160	0.514s	40	226	0.789s	51	292	1.125s
8	34	0.002s	19	100	0.271s	30	166	0.510s	41	232	0.826s	52	298	1.158s
9	40	0.006s	20	106	0.298s	31	172	0.532s	42	238	0.840s			
10	46	0.013s	21	112	0.322s	32	178	0.569s	43	244	0.886s			
11	52	0.028s	22	118	0.332s	33	184	0.574s	44	250	0.909s			

Pre-search time: 0.518s Search time: 24.523s Total time: 25.041s

Table 4: Experimental results of `KNOT-384`

Differential property														
r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	2		17	87	0.055s	33	172	0.009s	49	258	0.014s	65	343	0.011s
2	4	<0.001s	18	92	0.013s	34	178	0.014s	50	263	0.011s	66	348	0.009s
3	7	0.002s	19	98	0.015s	35	183	0.011s	51	268	0.009s	67	354	0.014s
4	10	0.002s	20	103	0.011s	36	188	0.010s	52	274	0.014s	68	359	0.010s
5	14	0.002s	21	108	0.009s	37	194	0.015s	53	279	0.011s	69	364	0.009s
6	18	0.001s	22	114	0.014s	38	199	0.010s	54	284	0.009s	70	370	0.013s
7	25	0.002s	23	119	0.010s	39	204	0.009s	55	290	0.014s	71	375	0.011s
8	32	0.002s	24	124	0.008s	40	210	0.014s	56	295	0.013s	72	380	0.009s
9	40	0.006s	25	130	0.014s	41	215	0.012s	57	300	0.009s	73	386	0.013s
10	49	0.057s	26	135	0.010s	42	220	0.009s	58	306	0.014s	74	391	0.011s
11	55	0.166s	27	140	0.008s	43	226	0.015s	59	311	0.011s	75	396	0.009s
12	60	0.235s	28	146	0.014s	44	231	0.011s	60	316	0.009s	76	402	0.015s
13	66	0.510s	29	151	0.011s	45	236	0.009s	61	322	0.015s			
14	71	0.474s	30	156	0.008s	46	242	0.016s	62	327	0.011s			
15	76	0.322s	31	162	0.014s	47	247	0.011s	63	332	0.009s			
16	82	0.206s	32	167	0.010s	48	252	0.009s	64	338	0.014s			
Pre-search time: 0.030s Search time: 2.716s Total time: 2.746s														

Linear property														
r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	2		17	88	0.310s	33	184	0.732s	49	280	1.333s	65	376	1.919s
2	4	<0.001s	18	94	0.322s	34	190	0.777s	50	286	1.372s	66	382	1.978s
3	8	0.002s	19	100	0.348s	35	196	0.799s	51	292	1.402s	67	388	2.101s
4	12	0.002s	20	106	0.396s	36	202	0.852s	52	298	1.450s	68	394	2.138s
5	16	0.002s	21	112	0.421s	37	208	0.878s	53	304	1.482s	69	400	2.067s
6	20	0.002s	22	118	0.437s	38	214	0.927s	54	310	1.523s	70	406	2.173s
7	26	0.002s	23	124	0.486s	39	220	0.957s	55	316	1.569s	71	412	2.190s
8	34	0.003s	24	130	0.488s	40	226	0.987s	56	322	1.630s	72	418	2.227s
9	40	0.006s	25	136	0.512s	41	232	1.068s	57	328	1.661s	73	424	2.262s
10	46	0.014s	26	142	0.538s	42	238	1.051s	58	334	1.659s	74	430	2.274s
11	52	0.033s	27	148	0.572s	43	244	1.104s	59	340	1.735s	75	436	2.342s
12	58	0.073s	28	154	0.589s	44	250	1.151s	60	346	1.742s	76	442	2.355s
13	64	0.135s	29	160	0.652s	45	256	1.180s	61	352	1.786s			
14	70	0.206s	30	166	0.647s	46	262	1.244s	62	358	1.819s			
15	76	0.245s	31	172	0.683s	47	268	1.254s	63	364	1.838s			
16	82	0.274s	32	178	0.730s	48	274	1.290s	64	370	1.872s			
Pre-search time: 1.093s Search time: 77.280s Total time: 78.373s														

Table 5: Experimental results of KNOT-512

Differential property

r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	2		21	108	0.011s	41	215	0.014s	61	322	0.017s	81	428	0.011s
2	4	<0.001s	22	114	0.017s	42	220	0.011s	62	327	0.014s	82	434	0.016s
3	7	0.001s	23	119	0.013s	43	226	0.017s	63	332	0.011s	83	439	0.014s
4	10	0.002s	24	124	0.010s	44	231	0.016s	64	338	0.015s	84	444	0.010s
5	14	0.001s	25	130	0.016s	45	236	0.011s	65	343	0.013s	85	450	0.017s
6	18	0.001s	26	135	0.014s	46	242	0.017s	66	348	0.010s	86	455	0.014s
7	25	0.001s	27	140	0.010s	47	247	0.015s	67	354	0.016s	87	460	0.011s
8	32	0.002s	28	146	0.017s	48	252	0.010s	68	359	0.013s	88	466	0.016s
9	40	0.006s	29	151	0.014s	49	258	0.016s	69	364	0.010s	89	471	0.013s
10	49	0.070s	30	156	0.011s	50	263	0.014s	70	370	0.016s	90	476	0.010s
11	55	0.200s	31	162	0.016s	51	268	0.010s	71	375	0.014s	91	482	0.016s
12	60	0.308s	32	167	0.014s	52	274	0.015s	72	380	0.010s	92	487	0.015s
13	66	0.685s	33	172	0.010s	53	279	0.014s	73	386	0.016s	93	492	0.010s
14	71	0.677s	34	178	0.017s	54	284	0.011s	74	391	0.014s	94	498	0.017s
15	76	0.456s	35	183	0.014s	55	290	0.017s	75	396	0.010s	95	503	0.014s
16	82	0.302s	36	188	0.012s	56	295	0.014s	76	402	0.016s	96	508	0.010s
17	87	0.076s	37	194	0.018s	57	300	0.013s	77	407	0.014s	97	514	0.016s
18	92	0.018s	38	199	0.014s	58	306	0.016s	78	412	0.011s	98	519	0.014s
19	98	0.017s	39	204	0.011s	59	311	0.014s	79	418	0.016s	99	524	0.011s
20	103	0.015s	40	210	0.017s	60	316	0.011s	80	423	0.014s	100	530	0.017s

Pre-search time: 0.091s Search time: 3.932s Total time: 4.023s

Linear property

r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	2		21	112	0.483s	41	232	1.232s	61	352	2.106s	81	472	3.000s
2	4	<0.001s	22	118	0.511s	42	238	1.272s	62	358	2.187s	82	478	3.024s
3	8	0.003s	23	124	0.546s	43	244	1.322s	63	364	2.227s	83	484	3.084s
4	12	0.002s	24	130	0.576s	44	250	1.368s	64	370	2.224s	84	490	3.127s
5	16	0.002s	25	136	0.605s	45	256	1.391s	65	376	2.468s	85	496	3.230s
6	20	0.003s	26	142	0.670s	46	262	1.458s	66	382	2.365s	86	502	3.474s
7	26	0.003s	27	148	0.697s	47	268	1.479s	67	388	2.447s	87	508	3.316s
8	34	0.003s	28	154	0.719s	48	274	1.539s	68	394	2.427s	88	514	3.333s
9	40	0.007s	29	160	0.748s	49	280	1.573s	69	400	2.457s	89	520	3.405s
10	46	0.016s	30	166	0.776s	50	286	1.694s	70	406	2.512s	90	526	3.431s
11	52	0.041s	31	172	0.819s	51	292	1.838s	71	412	2.557s	91	532	3.423s
12	58	0.089s	32	178	0.829s	52	298	1.857s	72	418	2.609s	92	538	3.639s
13	64	0.177s	33	184	0.894s	53	304	1.780s	73	424	2.618s	93	544	3.842s
14	70	0.250s	34	190	0.916s	54	310	1.905s	74	430	2.756s	94	550	3.589s
15	76	0.286s	35	196	0.938s	55	316	2.180s	75	436	2.700s	95	556	3.584s
16	82	0.317s	36	202	1.020s	56	322	2.384s	76	442	2.759s	96	562	3.694s
17	88	0.359s	37	208	1.091s	57	328	2.017s	77	448	2.833s	97	568	3.771s
18	94	0.389s	38	214	1.100s	58	334	2.145s	78	454	2.835s	98	574	3.870s
19	100	0.407s	39	220	1.139s	59	340	2.181s	79	460	2.907s	99	580	3.829s
20	106	0.445s	40	226	1.181s	60	346	2.181s	80	466	2.915s	100	586	3.952s

Pre-search time: 2.524s Search time: 174.379s Total time: 176.903s

Table 6: Experimental results of PRESENT

Differential property														
r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	2		8	32	0.003s	15	66	0.003s	22	96	0.003s	29	128	0.004s
2	4	<0.001s	9	36	0.003s	16	70	0.004s	23	100	0.003s	30	132	0.004s
3	8	0.003s	10	41	0.003s	17	74	0.004s	24	106	0.005s	31	136	0.003s
4	12	0.002s	11	46	0.004s	18	78	0.004s	25	110	0.004s			
5	20	0.002s	12	52	0.005s	19	82	0.002s	26	116	0.005s			
6	24	0.005s	13	56	0.003s	20	86	0.003s	27	120	0.002s			
7	28	0.002s	14	62	0.006s	21	90	0.003s	28	124	0.003s			
Pre-search time: 0s Search time: 0.100s Total time: 0.100s														

Linear property														
r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	2		8	28	0.006s	15	56	0.011s	22	84	0.007s	29	112	0.006s
2	4	<0.001s	9	32	0.006s	16	60	0.007s	23	88	0.009s	30	116	0.006s
3	8	0.006s	10	36	0.007s	17	64	0.010s	24	92	0.007s	31	120	0.007s
4	12	0.005s	11	40	0.007s	18	68	0.010s	25	96	0.008s			
5	16	0.005s	12	44	0.007s	19	72	0.009s	26	100	0.006s			
6	20	0.006s	13	48	0.010s	20	76	0.007s	27	104	0.006s			
7	24	0.006s	14	52	0.006s	21	80	0.007s	28	108	0.006s			
Pre-search time: 0.082s Search time: 0.206s Total time: 0.288s														

Table 7: Experimental results of GIFT-64

Differential property														
r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	1.415		7	28.415	0.001s	13	62	0.001s	19	92	<0.001s	25	122	0.001s
2	3.415	<0.001s	8	38	0.023s	14	68	0.012s	20	98	0.003s	26	128	0.003s
3	7	0.001s	9	42	0.002s	15	72	<0.001s	21	102	0.001s	27	132	<0.001s
4	11.415	<0.001s	10	48	0.007s	16	78	0.003s	22	108	0.003s	28	138	0.002s
5	17	<0.001s	11	52	0.001s	17	82	<0.001s	23	112	<0.001s			
6	22.415	0.001s	12	58	0.009s	18	88	0.003s	24	118	0.003s			
Pre-search time: 0s Search time: 0.080s Total time: 0.080s														

Linear property														
r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	2		7	26	0.001s	13	68	0.017s	19	104	0.001s	25	140	0.002s
2	4	<0.001s	8	32	0.002s	14	74	0.206s	20	110	0.062s	26	146	0.099s
3	6	<0.001s	9	40	0.002s	15	80	0.011s	21	116	0.001s	27	152	0.002s
4	10	<0.001s	10	50	0.016s	16	86	0.119s	22	122	0.086s			
5	14	0.001s	11	58	0.509s	17	92	0.002s	23	128	0.001s			
6	20	0.001s	12	62	<0.001s	18	98	0.059s	24	134	0.101s			
Pre-search time: 0.368s Search time: 1.402s Total time: 1.770s														

Table 8: Experimental results of `GIFT-128`

Differential property

r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	1.415		9	45.415	0.183s	17	96.415	3.368s	25	157.415	0.781h	33	210.415	0.572h
2	3.415	0.023s	10	49.415	0.075s	18	103.415	21.683s	26	162.415	190.578s	34	217.415	0.314h
3	7	0.013s	11	54.415	0.083s	19	110.83	39.355s	27	168.415	384.132s	35	224.83	0.740h
4	11.415	0.013s	12	60.415	0.173s	20	121.415	360.412s	28	174.415	0.296h	36	234.415	3.523h
5	17	0.017s	13	67.83	0.811s	21	126.415	13.200s	29	181.83	390.916s	37	240.415	630.109s
6	22.415	0.013s	14	79	24.645s	22	132.415	30.739s	30	193	2.612h	38	246.415	1.063h
7	28.415	0.016s	15	85.415	2.840s	23	139.415	170.537s	31	198.415	0.331h	39	253.415	0.617h
8	39	0.260s	16	90.415	1.321s	24	146.83	274.807s	32	204.415	286.450s	40	260.415	1.149h

Pre-search time: 176.240s   Search time: 12.783h   Total time: 12.832h

Linear property

r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	2		9	44	0.109s	17	102	0.018s	25	172	1.807h	33	234	1.015h
2	4	<0.001s	10	52	2.346s	18	112	150.076s	26	182	8.276h	34	242	3.670h
3	6	0.019s	11	62	5.737s	19	118	0.531s	27	188	5.102h	35	252	0.738h
4	10	0.020s	12	72	101.982s	20	128	299.405s	28	196	2.994h	36	260	8.954h
5	14	0.018s	13	76	0.017s	21	136	121.294s	29	202	1.516h	37	266	0.018s
6	20	0.020s	14	82	22.521s	22	148	0.553h	30	210	1.588h	38	274	185.825s
7	26	0.019s	15	90	119.870s	23	158	1.706h	31	216	2.565h	39	280	4.593h
8	34	0.062s	16	96	218.117s	24	164	0.726h	32	224	636.726s	40	286	0.017s

Pre-search time: 7.227h   Search time: 46.322h   Total time: 53.549h

Table 9: Experimental results of `LBlock`

Differential property

r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	0		8	22	0.002s	15	66	0.204s	22	102	1.138s	29	135	0.369s
2	2		9	28	0.005s	16	72	0.265s	23	106	2.032s	30	141	0.673s
3	4	<0.001s	10	36	0.012s	17	76	0.506s	24	112	2.493s	31	146	2.915s
4	6	<0.001s	11	44	0.036s	18	82	0.774s	25	115	0.596s	32	151	2.310s
5	8	<0.001s	12	48	0.034s	19	86	0.844s	26	121	0.296s			
6	12	<0.001s	13	56	0.386s	20	92	1.118s	27	126	1.473s			
7	16	<0.001s	14	62	0.196s	21	96	0.887s	28	131	0.873s			

Pre-search time: 0.438s   Search time: 20.437s   Total time: 20.437s

Linear property

r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	0		8	22	0.002s	15	66	0.252s	22	100	0.677s	29	132	2.801s
2	2		9	28	0.005s	16	72	0.377s	23	104	0.275s	30	138	0.294s
3	4	<0.001s	10	36	0.012s	17	74	0.507s	24	110	0.671s	31	144	1.350s
4	6	<0.001s	11	44	0.038s	18	80	0.187s	25	112	1.197s	32	148	2.581s
5	8	<0.001s	12	48	0.040s	19	84	0.030s	26	118	0.046s			
6	12	<0.001s	13	54	0.219s	20	90	0.119s	27	124	0.441s			
7	16	<0.001s	14	60	0.012s	21	94	0.463s	28	130	1.492s			

Pre-search time: 0.380s   Search time: 14.088s   Total time: 14.468s

Table 10: Experimental results of `TWINE`

Differential property

r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	0		9	28	0.005s	17	77	2.826s	25	116	0.190s	33	155	0.553s
2	2		10	38	0.018s	18	83	1.214s	26	122	0.403s	34	161	0.419s
3	4	<0.001s	11	46	0.133s	19	88	0.870s	27	126	0.014s	35	166	0.348s
4	6	<0.001s	12	51	0.056s	20	94	1.502s	28	132	0.677s	36	172	1.419s
5	8	<0.001s	13	58	4.345s	21	97	1.045s	29	136	0.205s			
6	12	<0.001s	14	64	5.588s	22	103	0.720s	30	142	1.266s			
7	16	0.001s	15	68	4.736s	23	107	0.127s	31	146	0.282s			
8	22	0.001s	16	74	6.746s	24	113	0.766s	32	152	1.241s			

Pre-search time: 0.607s Search time: 37.716s Total time: 38.323s

Linear property

r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	0		9	28	0.005s	17	72	0.007s	25	108	0.001s	33	144	0.002s
2	2		10	36	0.013s	18	78	0.003s	26	114	0.001s	34	150	0.001s
3	4	<0.001s	11	44	0.044s	19	82	0.004s	27	118	0.003s	35	154	0.002s
4	6	<0.001s	12	48	0.070s	20	88	0.004s	28	124	0.002s	36	160	0.002s
5	8	<0.001s	13	54	0.016s	21	90	0.002s	29	126	0.002s			
6	12	<0.001s	14	60	0.011s	22	96	0.001s	30	132	0.001s			
7	16	<0.001s	15	64	0.007s	23	100	0.002s	31	136	0.002s			
8	22	0.002s	16	70	0.010s	24	106	0.002s	32	142	0.002s			

Pre-search time: 0.650s Search time: 0.224s Total time: 0.874s

Table 11: Experimental results of `WARP`

Differential property

r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	0		10	34	0.001s	19	132	22.447s	28	202	2.720h	37	290	966.541s
2	2		11	44	0.003s	20	140	10.310s	29	212	70.975s	38	300	0.383h
3	4	0.108s	12	56	0.202s	21	150	27.883s	30	218	6.207s	39	308	0.972h
4	6	<0.001s	13	68	2.712s	22	158	19.125s	31	228	18.211s	40	318	213.574s
5	8	<0.001s	14	80	13.145s	23	164	13.369s	32	240	88.656s	41	324	2.546h
6	12	0.001s	15	94	89.970s	24	170	15.344s	33	252	271.083s			
7	16	0.001s	16	104	41.125s	25	178	16.547s	34	262	266.790s			
8	22	0.001s	17	114	19.814s	26	186	30.166s	35	270	153.741s			
9	28	0.001s	18	122	8.193s	27	194	179.661s	36	280	212.332s			

Pre-search time: 623.313s Search time: 7.393h Total time: 7.566h

Linear property

r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t	r	B_r	t
1	0		10	34	0.002s	19	132	21.352s	28	202	105.050s	37	290	937.453s
2	2		11	44	0.005s	20	140	10.489s	29	212	67.206s	38	300	0.373h
3	4	0.001s	12	56	0.251s	21	150	26.402s	30	218	5.863s	39	308	0.972h
4	6	<0.001s	13	68	2.503s	22	158	18.786s	31	228	17.131s	40	318	221.591s
5	8	<0.001s	14	80	12.998s	23	164	21.389s	32	240	83.686s	41	324	2.608h
6	12	0.002s	15	94	88.943s	24	170	13.679s	33	252	247.202s			
7	16	0.001s	16	104	41.219s	25	178	17.601s	34	262	259.413s			
8	22	0.001s	17	114	18.803s	26	186	36.088s	35	270	146.967s			
9	28	0.001s	18	122	7.962s	27	194	0.621h	36	280	215.591s			

Pre-search time: 474.051s Search time: 5.308h Total time: 5.440h