More on the Automatic Search for Differential Trails in NORX (Work in Progress)

V. Velichkov

University of Luxembourg, SnT

Directions in Authenticated Ciphers (DIAC) 2016
September 26, 2016, Nagoya, Japan

Outline

Resistance of NORX Against DC

- Designers give bounds against DC [AJN14]
- Use SAT-solver; memory is exhausted for more rounds
- Bounds based on best trails for up to $F^{2.0}$ rounds

W	32				64			
Scenario	init_N	$\text{init}_{N,K}$	rate	full	init_N	$\text{init}_{N,K}$	rate	full
$F^{0.5}$	6	2	2	0	6	2	2	0
$F^{1.0}$	(60)	22	10	2	(53)	22	12	2
$F^{1.5}$	(60)	(40)	(31)	12	(53)	(35)	(27)	12
$F^{2.0}$	(61)	(45)	(34)	(27)	(51)	(37)	(30)	(23)

Scenario: modify the nonce (init_N), nonce + key ($\text{init}_{N,K}$), rate words (rate), full state (full).

Motivation

Research Goal

Provide tighter bounds than the ones reported by [AJN14]

Disclaimer: Work-in-progress; preliminary results.

Our Contributions

1. New algorithm for finding optimal trails – **Best Search (BS)**
2. Heuristic version of BS – **Heuristic Search (HS)**
3. New (sub-optimal) trails on up to $F^{2.0}$ rounds with HS

W	32				64			
Scenario	init_N	$\text{init}_{N,K}$	rate	full	init_N	$\text{init}_{N,K}$	rate	full
$F^{0.5}$	6	2	2	0	6	2	2	0
$F^{1.0}$	84	22	10	2	88	22	12	2
$F^{1.5}$	319	210	205	12	413	263	245	12
$F^{2.0}$	606	476	493	354	809	656	645	561

Outline

Overview of NORX

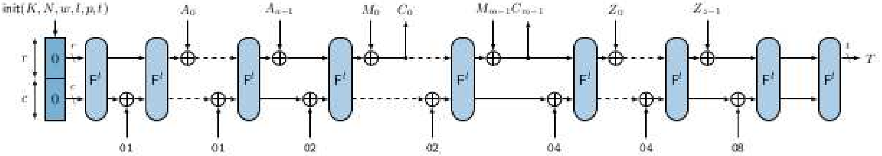
- Word size: $W \in \{32, 64\}$ bits
- Rounds: $1 \leq R \leq 63$
- Parallelism: $0 \leq D \leq 255$
- Tag size: $|A| \leq 10W$

NORXW-R-D	Nonce (2W)	Key (4W)	Tag (4W)	Classification
NORX64-4-1	128	256	256	Standard
NORX32-4-1	64	128	128	Standard
NORX64-6-1	128	256	256	High security
NORX32-6-1	64	128	128	High security
NORX64-4-4	128	256	256	High throughput

Credits: Philipp Jovanovic, ESORICS 2014.

NORX = NO(T A)RX

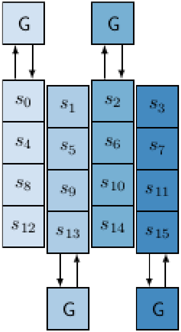
- Sponge structure based on the monkeyDuplex construction
- LRX permutation: $\oplus, \ggg, \wedge, \ll 1$
- Non-linear component: $H(x, y) = (x \oplus y) \oplus ((x \wedge y) \ll 1)$
- No S-box; No modular addition

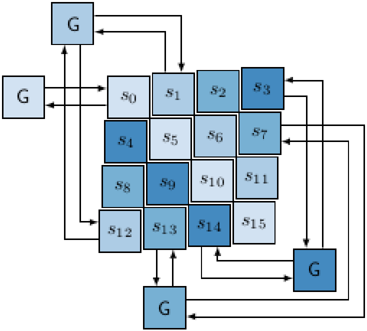


Credits: NORX Specification `https://norx.io/`

PERMUTATION F

F is composed of G-**circuit** (next slide) applied in parallel first to each of the 4 columns of the state followed by an application to each of the 4 diagonals of the state: $F = F_{\text{dia}} \circ F_{\text{col}}$

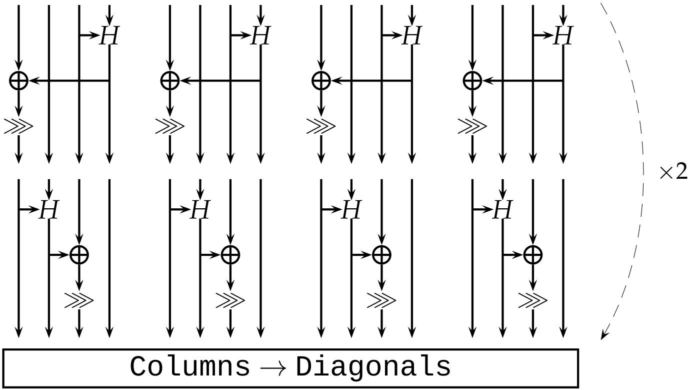




Credits: NORX Specification `https://norx.io/`

HALF-ROUND: $F^{0.5} = F_{\text{col}}$

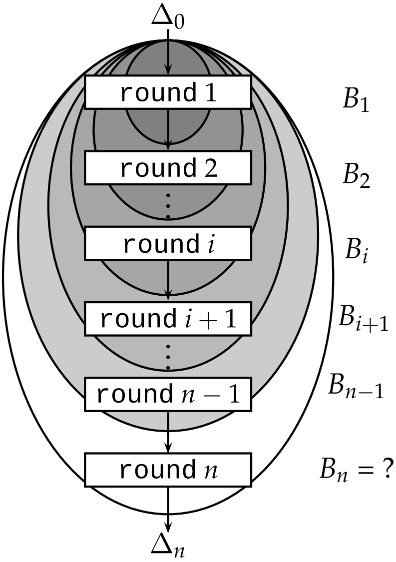
4 parallel app. of the G-circuit, where $H(x, y) = (x \oplus y) \oplus ((x \wedge y) \ll 1)$.



Note: Initialization = F^8; Data processing = F^4.

Outline

Matsui's Algorithm



Input: best p for $n-1$ rounds:

$B_1, B_2, \ldots, B_{n-1}; \bar{B}_n$

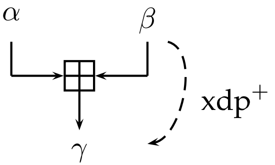
Output: best p for n rounds:

B_n

if $P_1 P_2 \ldots P_i B_{n-i} \geq \bar{B}_n$: $i \leftarrow i+1$

if $i = n$: $B_n \leftarrow \bar{B}_n$

MATSUI'S ALGORITHM FOR ARX [BVLC16]



$$2^{-\mathrm{hw}(\neg \mathrm{eq}(\alpha,\beta,\gamma) \wedge \mathrm{mask}_{n-1})}$$ [LM01]

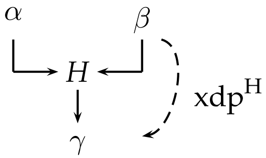
Proposition (Monotonicity of xdp^+)

xdp^+ *is monotonously decreasing with the word size* w *of* α, β, γ*:*

$$\tilde{p}_1 \geq \tilde{p}_2 \ldots \geq \tilde{p}_{w-1} \geq \tilde{p}_w = \mathrm{xdp}^+(\alpha, \beta \to \gamma) \ ,$$

where $\tilde{p}_i = \mathrm{xdp}^+(\alpha[i-1:0], \beta[i-1:0] \to \gamma[i-1:0]) : \ w \geq i \geq 1$ *, is the probability of the partial differential composed of the i LS bits of* α, β, γ.

MONOTONICITY OF xdp^{H}



$$H(x, y) = (x \oplus y) \oplus ((x \wedge y) \ll 1)$$

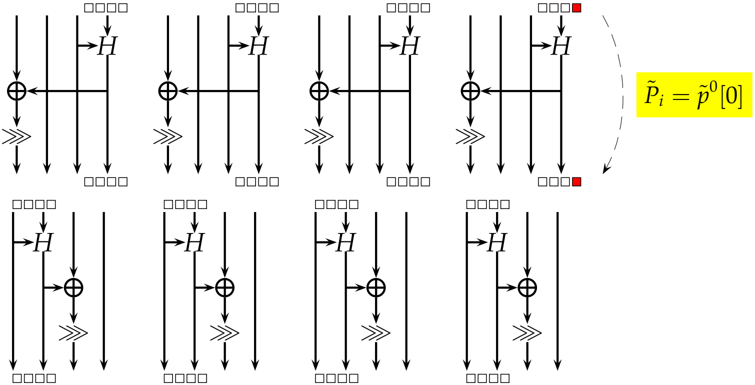
The Differential Probability of H [AJN14]

$$\text{xdp}^{\text{H}}(\alpha, \beta \to \gamma) = \begin{cases} 2^{-\text{hw}((\alpha \vee \beta) \ll 1)} & \iff (\alpha \oplus \beta \oplus \gamma) \wedge (\neg((\alpha \vee \beta) \ll 1)) = 0 \\ 0, & \text{otherwise} \end{cases}$$

Proposition (Monotonicity of xdp^{H})

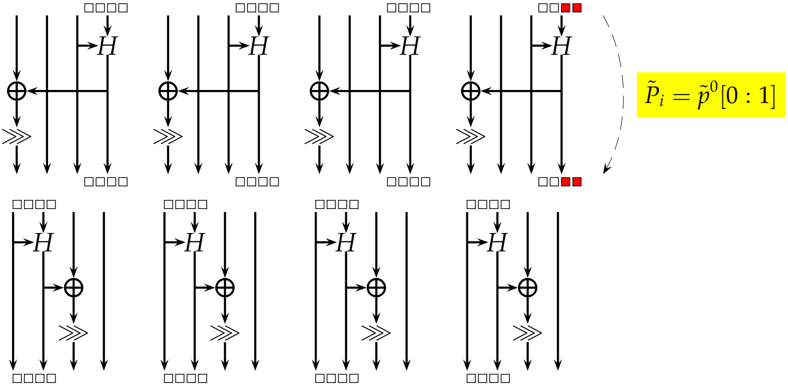
xdp^{H} *is monotonously decreasing with the word size of* α, β, γ.

Best Search (BS): Rounds i and $i+1$



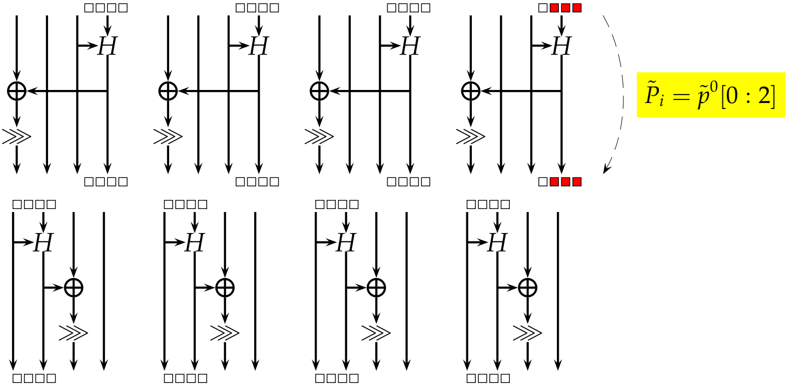
if $P_1 P_2 \ldots \tilde{P}_i B_{n-i} \geq \bar{B}_n$ assign next bit

Best Search (BS): Rounds i and $i+1$



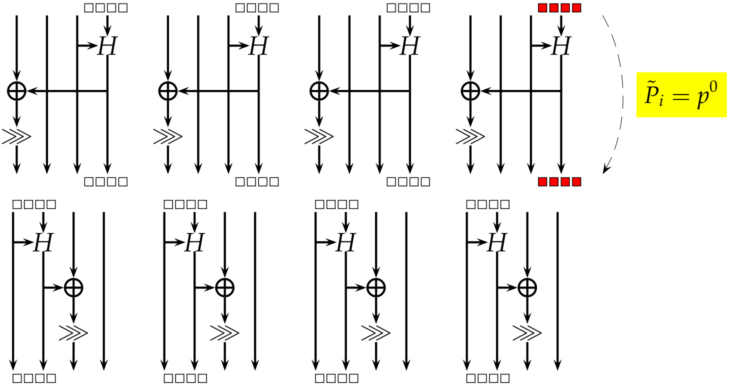
if $P_1 P_2 \ldots \tilde{P}_i B_{n-i} \geq \bar{B}_n$ assign next bit

BEST SEARCH (BS): ROUNDS i AND $i+1$



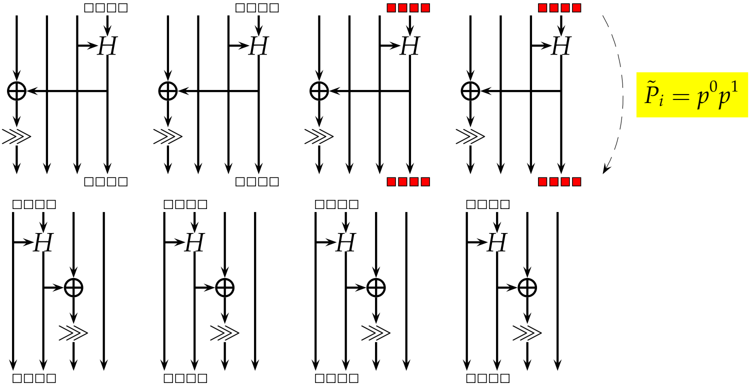
if $P_1 P_2 \ldots \tilde{P}_i B_{n-i} \geq \bar{B}_n$ assign next bit

BEST SEARCH (BS): ROUNDS i AND $i+1$



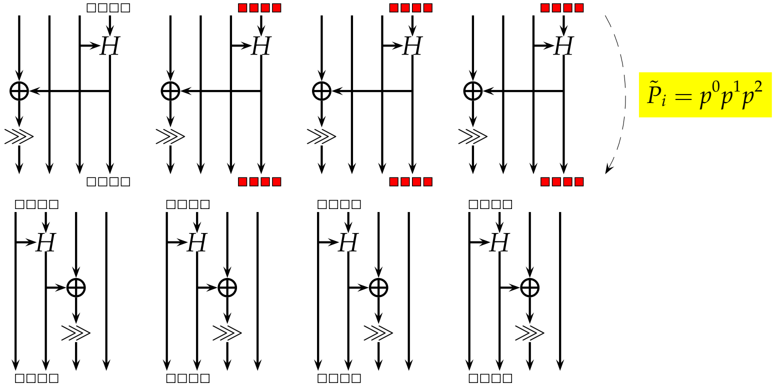
if $P_1 P_2 \ldots \tilde{P}_i B_{n-i} \geq \bar{B}_n$ assign next bit

Best Search (BS): Rounds i and $i+1$



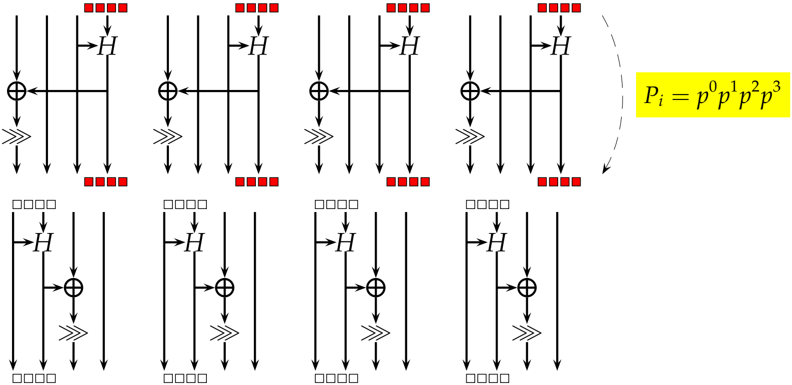
if $P_1 P_2 \ldots \tilde{P}_i B_{n-i} \geq \bar{B}_n$ assign next bit

BEST SEARCH (BS): ROUNDS i AND $i+1$



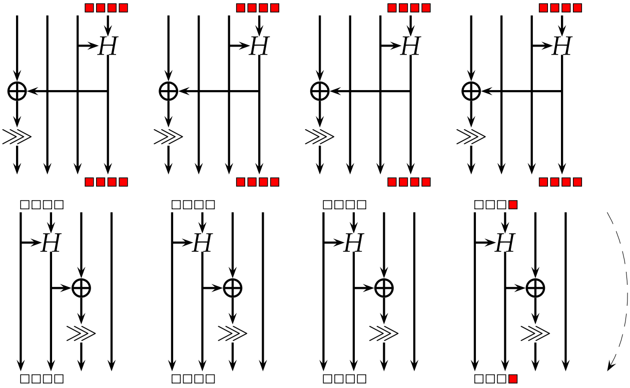
if $P_1 P_2 \ldots \tilde{P}_i B_{n-i} \geq \bar{B}_n$ assign next bit

Best Search (BS): Rounds i and $i+1$



if $P_1 P_2 \ldots P_i B_{n-i} \geq \bar{B}_n$ assign next bit

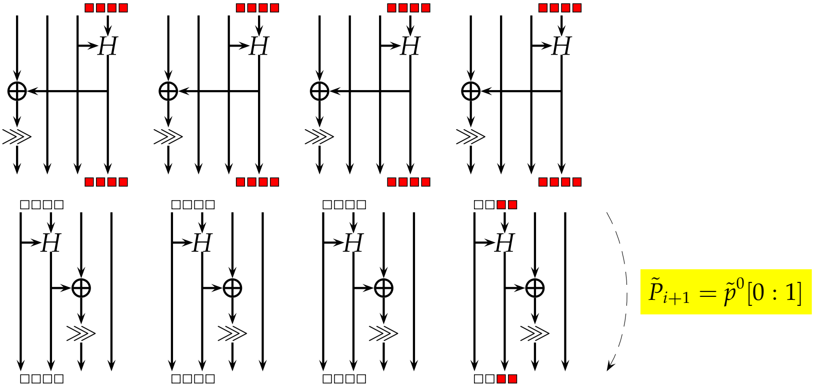
Best Search (BS): Rounds i and $i+1$



$\tilde{P}_{i+1} = \tilde{p}^{0}[0]$

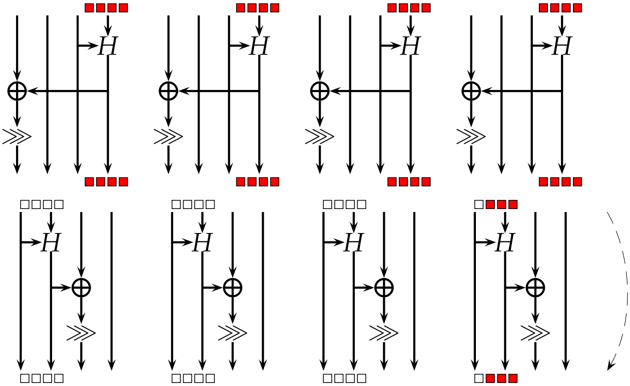
`if` $P_1 P_2 \ldots P_i \tilde{P}_{i+1} B_{n-i-1} \geq \bar{B}_n$ `assign next bit`

BEST SEARCH (BS): ROUNDS i AND $i+1$



if $P_1 P_2 \ldots P_i \tilde{P}_{i+1} B_{n-i-1} \geq \bar{B}_n$ assign next bit

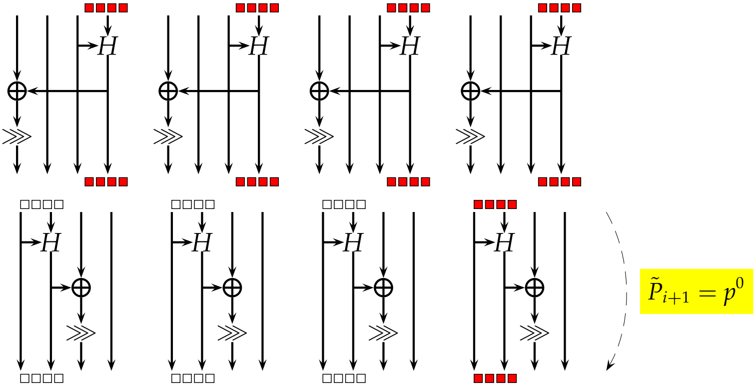
Best Search (BS): Rounds i and $i+1$



$\tilde{P}_{i+1} = \tilde{p}^0[0:2]$

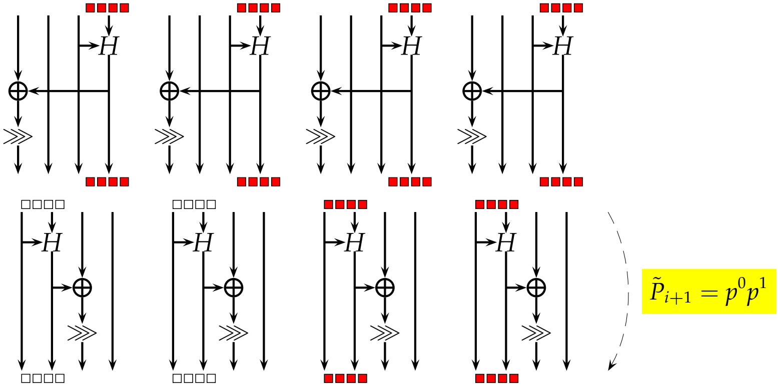
if $P_1 P_2 \ldots P_i \tilde{P}_{i+1} B_{n-i-1} \geq \bar{B}_n$ assign next bit

Best Search (BS): Rounds i and $i+1$



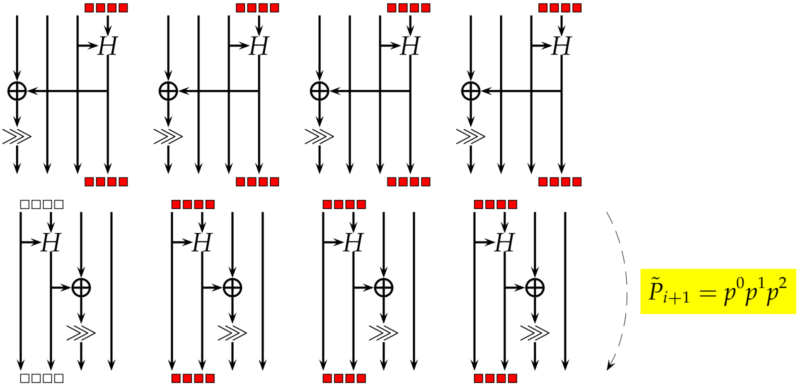
if $P_1 P_2 \ldots P_i \tilde{P}_{i+1} B_{n-i-1} \geq \bar{B}_n$ assign next bit

Best Search (BS): Rounds i and $i+1$



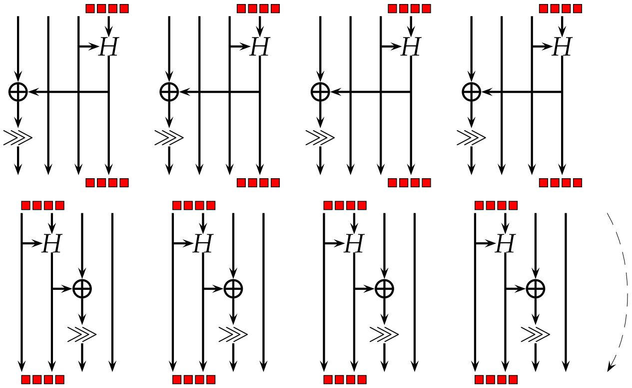
`if` $P_1 P_2 \dots P_i \tilde{P}_{i+1} B_{n-i-1} \geq \bar{B}_n$ `assign next bit`

Best Search (BS): Rounds i and $i+1$



`if` $P_1 P_2 \ldots P_i \tilde{P}_{i+1} B_{n-i-1} \geq \bar{B}_n$ `assign next bit`

Best Search (BS): Rounds i and $i+1$



$P_{i+1} = p^0 p^1 p^2 p^3$

if $P_1 P_2 \ldots P_i P_{i+1} B_{n-i-1} \geq \bar{B}_n$ assign next bit

Complexity of BS

Time and Memory Requirements

- Negligible memory (good)
- Time is proportional to num. of rounds and word size (bad)
- Feasible up to $F^{0.5}/F^{1.0}$ rounds (SAT-solver covers up to $F^{1.0}/F^{1.5}$)

Time Measurements for NORX32 and NORX64

- **BS** $F^{0.5}/F^{1.0} \approx$ 49 days; negl. RAM (3.4GHz PC)
- **SAT** $F^{1.0}/F^{1.5} \approx$ 8 hours; 49 GB RAM [AJN14]

Heuristic Search (HS)

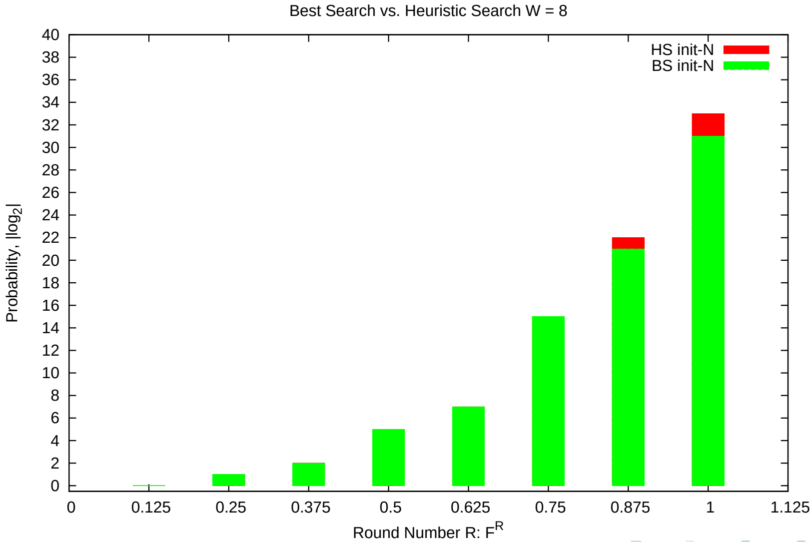
Heuristics to lower the time complexity of BS

1. Time Limit (TL)
   - How long to wait before the recursive call is terminated
2. Maximum Number of Tries (MT)
   - Number of times that we restart the search after the TL is exceeded.
3. Branch Factor Percentage x (BF)
   - In x out of 100 cases branch and explore both 0 and 1 for a given bit. In the remaining cases set the bit to 0.
   - Intuition: due to $\mathrm{xdp}^{\mathrm{H}}(\alpha, \beta \rightarrow \gamma) = 2^{-\mathrm{hw}((\alpha \vee \beta) \ll 1)}$

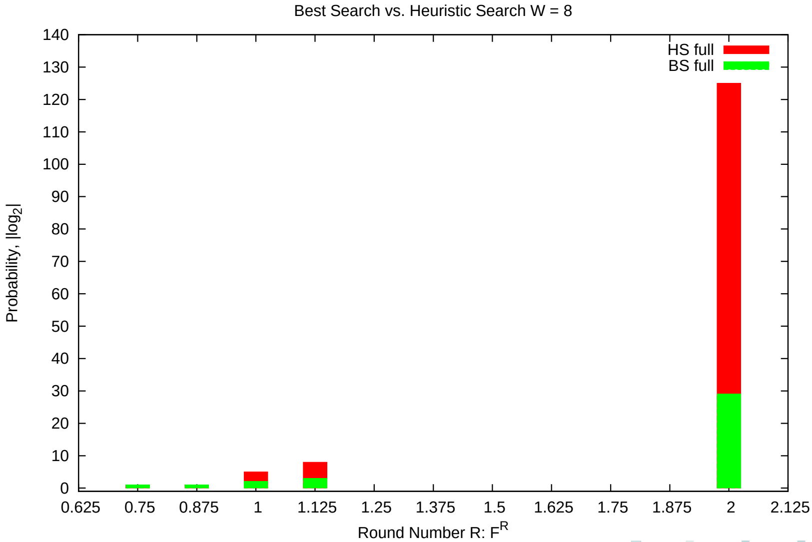
TL, sec	1	1	1	30	60	60	**600**	3600
MT	65535	1024	512	256	256	256	**64**	1
BF	30	50	75	50	50	50	**75**	100

Outline

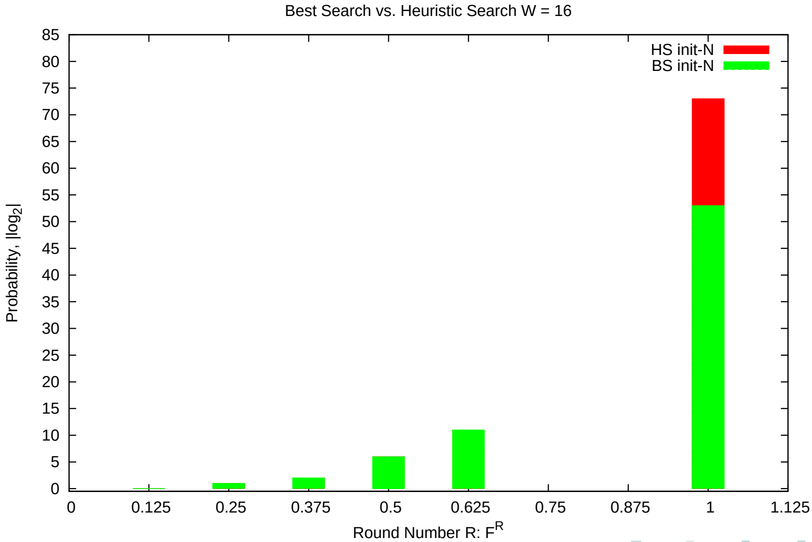
NORX8: BS vs. HS, scenario: init$_N$



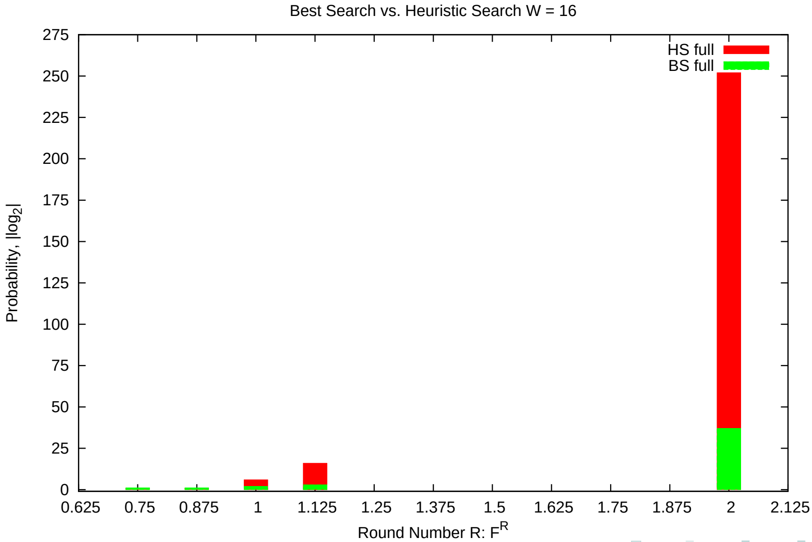
NORX8: BS vs. HS, scenario: full



NORX16: BS vs. HS, scenario: init$_N$



NORX16: BS vs. HS, scenario: full



OUTLINE

Conclusions and Future Work

W	32				64			
Scenario	init_N	$\text{init}_{N,K}$	rate	full	init_N	$\text{init}_{N,K}$	rate	full
$F^{0.5}$	6	2	2	0	6	2	2	0
$F^{1.0}$	84	22	10	2	88	22	12	2
$F^{1.5}$	319	210	205	12	413	263	245	12
$F^{2.0}$	606	476	493	354	809	656	645	561

- New heuristic algorithm for diff. search in NORX
- Matsui-like; Negligible memory; Manageable time
- Accuracy drastically degrades in the num. rounds and word size

Conclusions and Future Work

W	32				64			
Scenario	init_N	$\text{init}_{N,K}$	rate	full	init_N	$\text{init}_{N,K}$	rate	full
$F^{0.5}$	6	2	2	0	6	2	2	0
$F^{1.0}$	84	22	10	2	88	22	12	2
$F^{1.5}$	319	210	205	12	413	263	245	12
$F^{2.0}$	606	476	493	354	809	656	645	561

- New trails on up to $F^{2.0}$ rounds $\implies$ new diff. bounds
- Designers' bounds are too pessimistic (designer's PoV)
- Our bounds are too optimistic (designer's PoV)
- Need better heuristics; new (algorithmic) optimizations; BS + SAT

Conclusions and Future Work

W	32				64			
Scenario	init_N	$\text{init}_{N,K}$	rate	full	init_N	$\text{init}_{N,K}$	rate	full
$F^{0.5}$	6	2	2	0	6	2	2	0
$F^{1.0}$	84	22	10	2	88	22	12	2
$F^{1.5}$	319	210	205	12	413	263	245	12
$F^{2.0}$	606	476	493	354	809	656	645	561

Main Message

NORX is a conservative design with large security margin against DC

Conclusions and Future Work

W	32				64			
Scenario	init_N	$\text{init}_{N,K}$	rate	full	init_N	$\text{init}_{N,K}$	rate	full
$F^{0.5}$	6	2	2	0	6	2	2	0
$F^{1.0}$	84	22	10	2	88	22	12	2
$F^{1.5}$	319	210	205	12	413	263	245	12
$F^{2.0}$	606	476	493	354	809	656	645	561

Main Message

NORX is a conservative design with large security margin against DC

Thank you for your attention!

Questions?

References I

Jean-Philippe Aumasson, Philipp Jovanovic, and Samuel Neves.
Analysis of NORX: Investigating differential and rotational properties.
In Progress in Cryptology-LATINCRYPT 2014, pages 306–324. Springer International Publishing, 2014.

Alex Biryukov, Vesselin Velichkov, and Yann Le Corre.
Automatic search for the best trails in ARX: Application to block cipher SPECK.
In Fast Software Encryption–FSE, 2016.

REFERENCES II

Helger Lipmaa and Shiho Moriai.
Efficient Algorithms for Computing Differential Properties of Addition.
In Mitsuru Matsui, editor, FSE, volume 2355 of LNCS, pages 336–350. Springer, 2001.