On Variable Bit-Rotations in SHA-1-like Hash Functions

Christian Rechberger

Institute for Applied Information Processing and Communications (IAIK) - Krypto Group

Faculty of Computer Science
Graz University of Technology

ECRYPT

The work described in this paper has been supported by the European Commission through the IST Programme under Contract IST-2002-507932 ECRYPT.

Information Society Technologies

MAIN THEME

Why was SHA-1 designed the way it is?

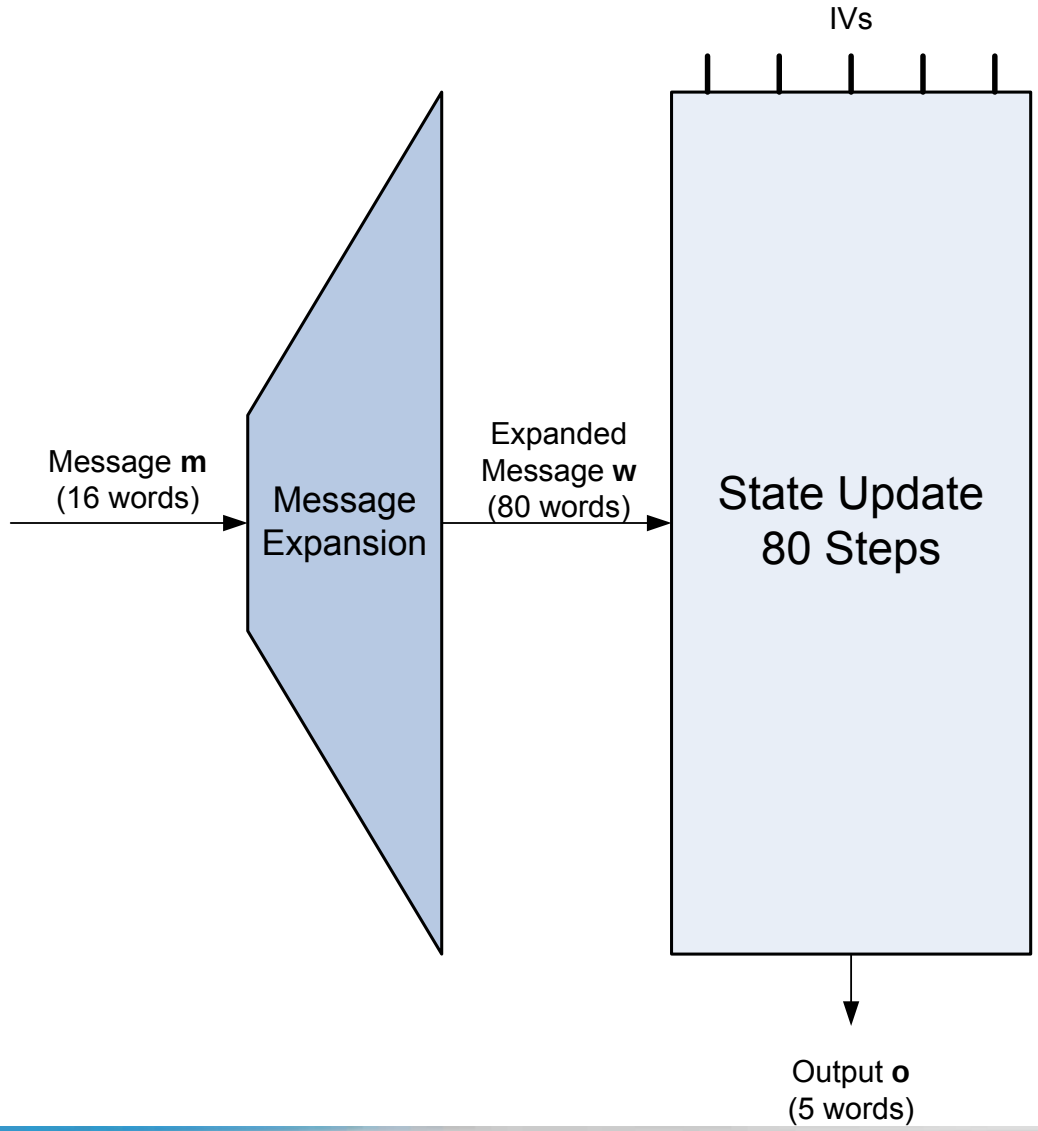
Overview

- Design history of SHA-1
- Review of basic design elements of SHA-1
  - Rotation in the Message Expansion
  - Recurrence-relation of the Message Expansion
  - Non-linear functions in the State Update
  - **Rotations in the State Update**
- Comparing variable and constant rotations
- Comparing constant rotations
- Conclusions

Overview

- Design history of SHA-1
- Review of basic design elements of SHA-1
  - Rotation in the Message Expansion
  - Recurrence-relation of the Message Expansion
  - Non-linear functions in the State Update
  - **Rotations in the State Update**
- Comparing variable and constant rotations
- Comparing constant rotations
- Conclusions

Outline of SHA-1



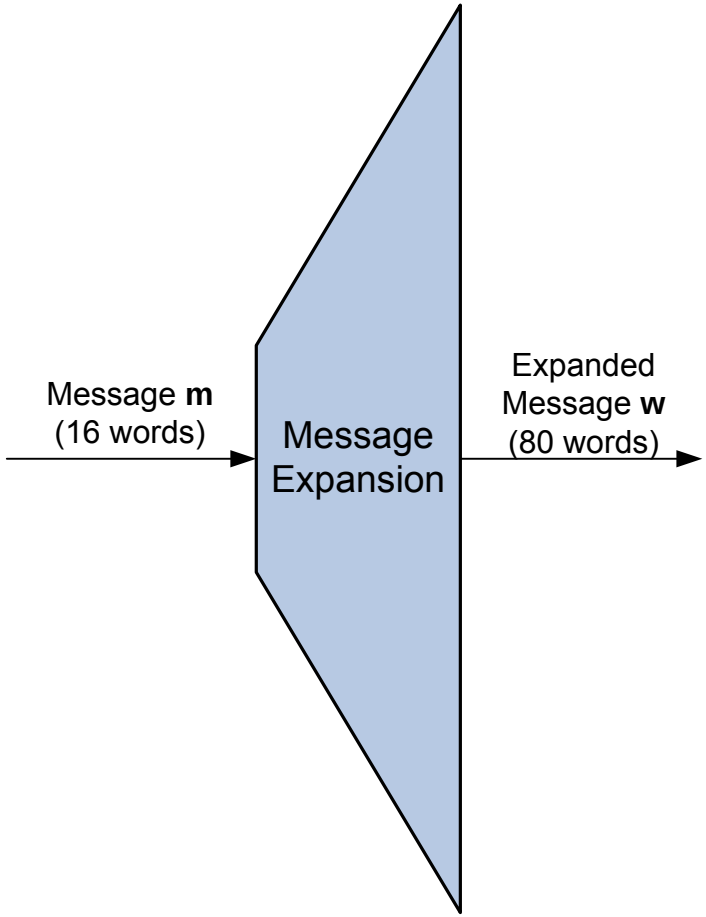
Design history of SHA-1

		Steps	#Chaining Variables	Message Expansion	Rotations
MD4	1990	48	4	Permutation	variable
MD5	1991	64	4	Permutation	variable
SHA-0	1992	80	5	Recurrence	const
SHA-1	1994	80	5	Recurrence + Bit-Rotation	const

Overview

- Design history of SHA-1
- Review of basic design elements of SHA-1
  - Rotation in the Message Expansion
  - Recurrence-relation of the Message Expansion
  - Non-linear functions in the State Update
  - **Rotations in the State Update**
- Comparing variable and constant rotations
- Comparing constant rotations
- Conclusions

Outline of SHA-1 – Message Expansion



Recurrence relation

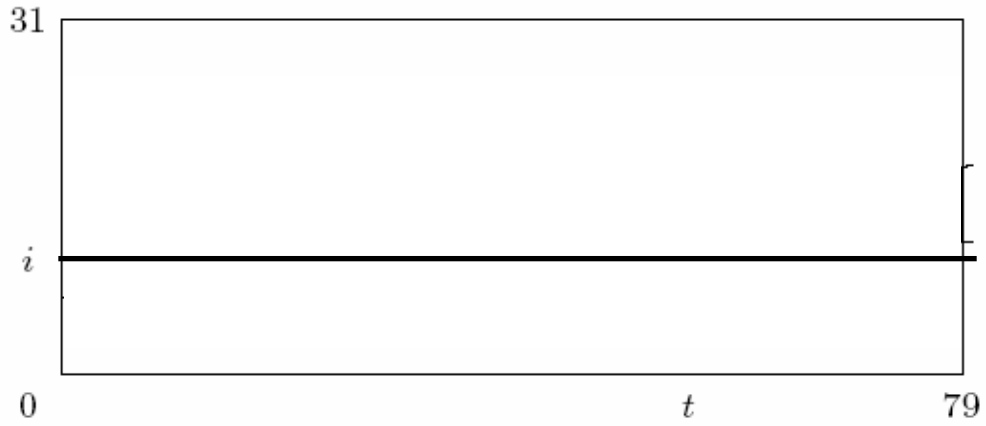
$$W_N = \begin{cases} M_N & for\,(0 \le N \le 15) \\ (W_{N-3} \oplus W_{N-8} \oplus W_{N-14} \oplus W_{N-16}) <<1 & for\,(16 \le N \le 79) \end{cases}$$

Effect of rotation

- Bits get connected over the 80 steps
- Bigger search space

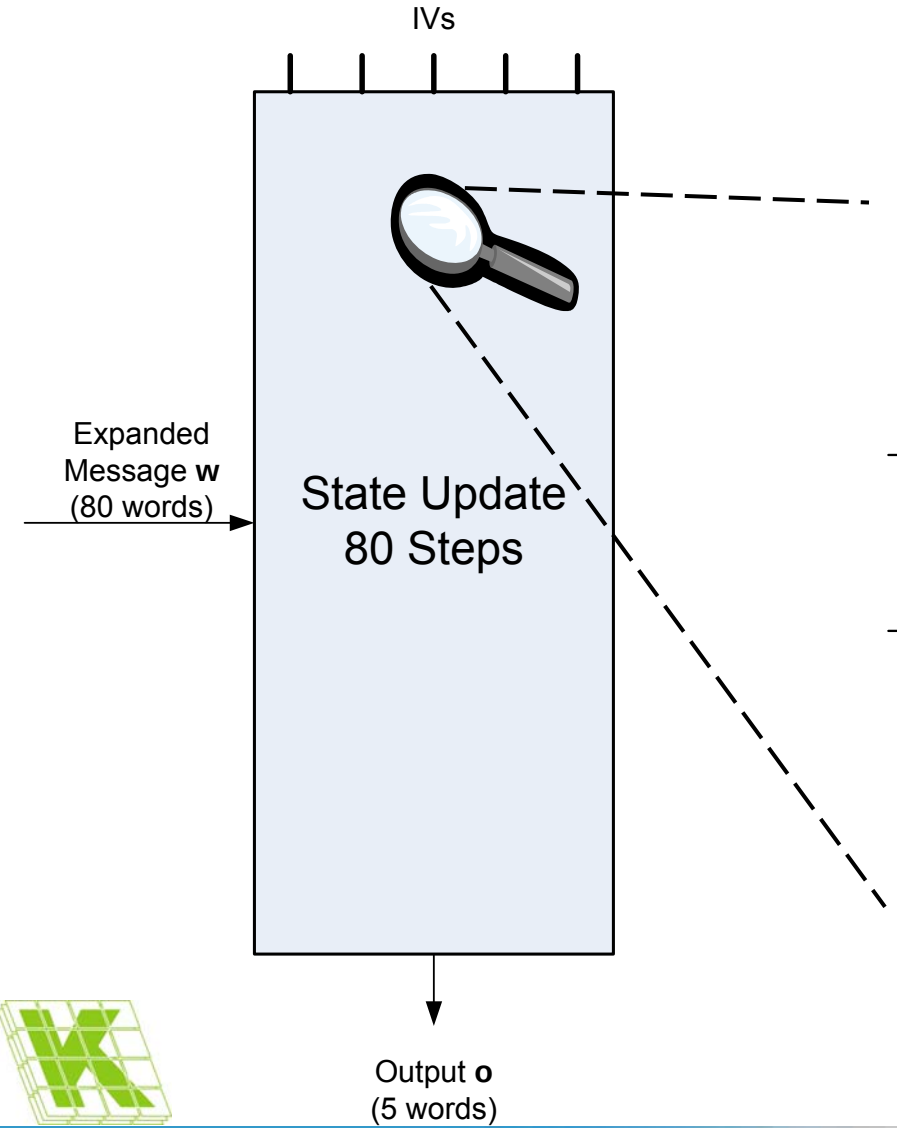
Diffusion Property of the SHA-0/1 Message Expansion

How does a change of one bit in the expanded message affect other bits?

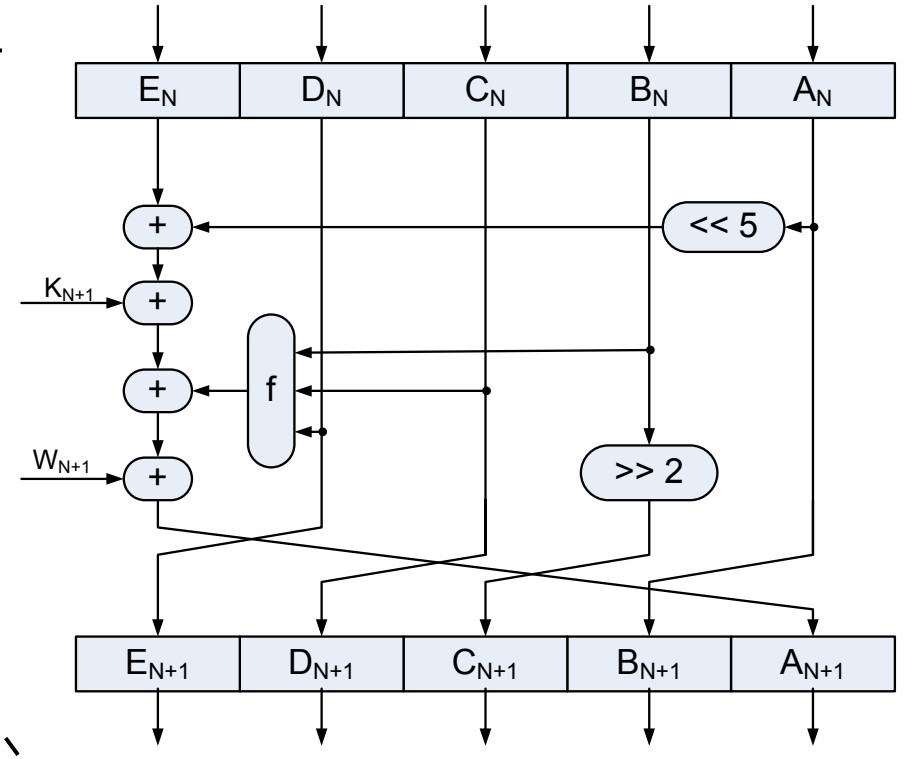


In the case of SHA-0, there is no such diffusion

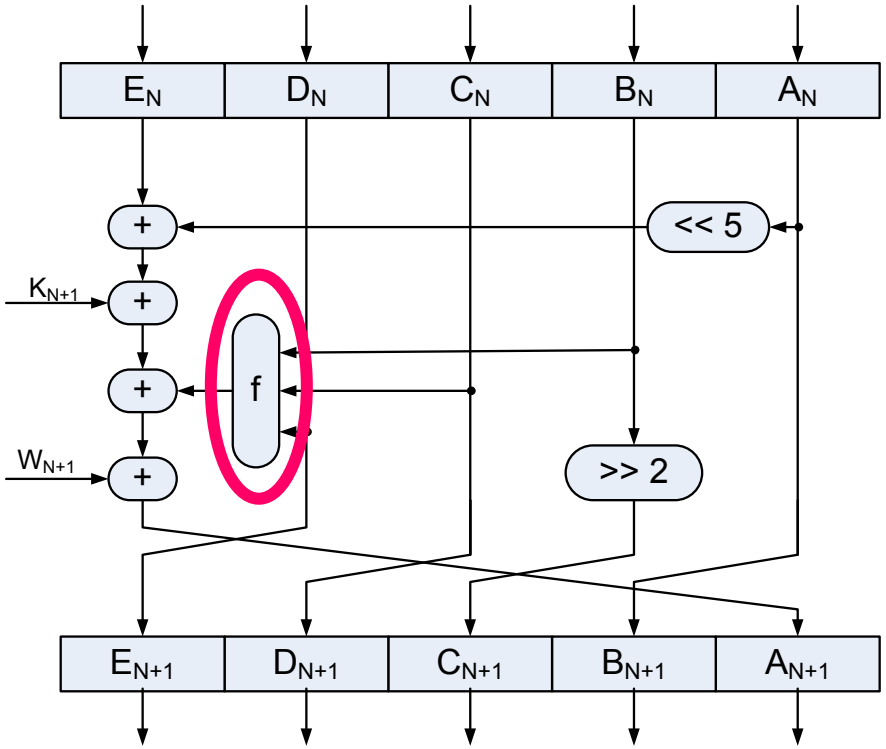
Outline of SHA – State Update



One round of the State Update



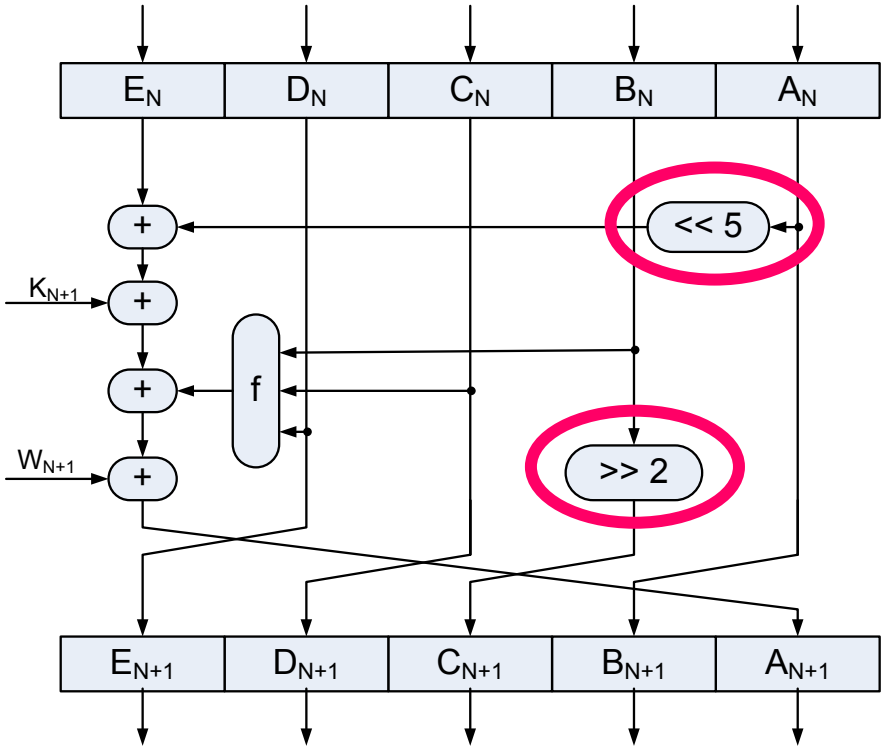
Non-linear functions in the State Update



Step i	Function f(i)	
	Name	Definition
0-19	IF	BC+(-B)D
20-39	XOR	B+C+D
40-59	MAJ	BC+BD+CD
60-79	XOR	B+C+D

Being 0-1 balanced is the only known design-criterion

Rotations in the State Update



How are these values chosen?

→ Could be ad-hoc

→ They could be the result of an optimization process!

Overview

- Design history of SHA-1
- Review of basic design elements of SHA-1
  - Rotation in the Message Expansion
  - Recurrence-relation of the Message Expansion
  - Non-linear functions in the State Update
  - **Rotations in the State Update**
- Comparing variable and constant rotations
- Comparing constant rotations
- Conclusions

The working principle of a collision attack applied to SHA

1. A linear approximation for SHA is constructed. → **Linear Approximation**
2. (Near)Collisions for the linear approximation are determined. → **Input Differential**
3. A collision for the real SHA is searched among the collisions for the linear approximation. → **Conditions & Final Collision Search**

The working principle of a Collision Attack applied to SHA

1. A linear approximation for SHA is constructed. → **Linear Approximation**
2. (Near)Collisions for the linear approximation are determined. → **Input Differential**
3. A collision for the real SHA is searched among the collisions for the linear approximation. → **Conditions & Final Collision Search**

The working principle of a Collision Attack applied to SHA

1. A linear approximation for SHA is constructed. → **Linear Approximation**
2. (Near)Collisions for the linear approximation are determined. → **Input Differential**
3. A collision for the real SHA is searched among the collisions for the linear approximation. → **Conditions & Final Collision Search**

Method to assess attack complexity

1. A linear approximation for SHA is constructed. → **Linear Approximation**
2. (Near)Collisions for the linear approximation are determined. → **Input Differential**
3. A collision for the real SHA is searched among the collisions for the linear approximation. → **Conditions & Final Collision Search**

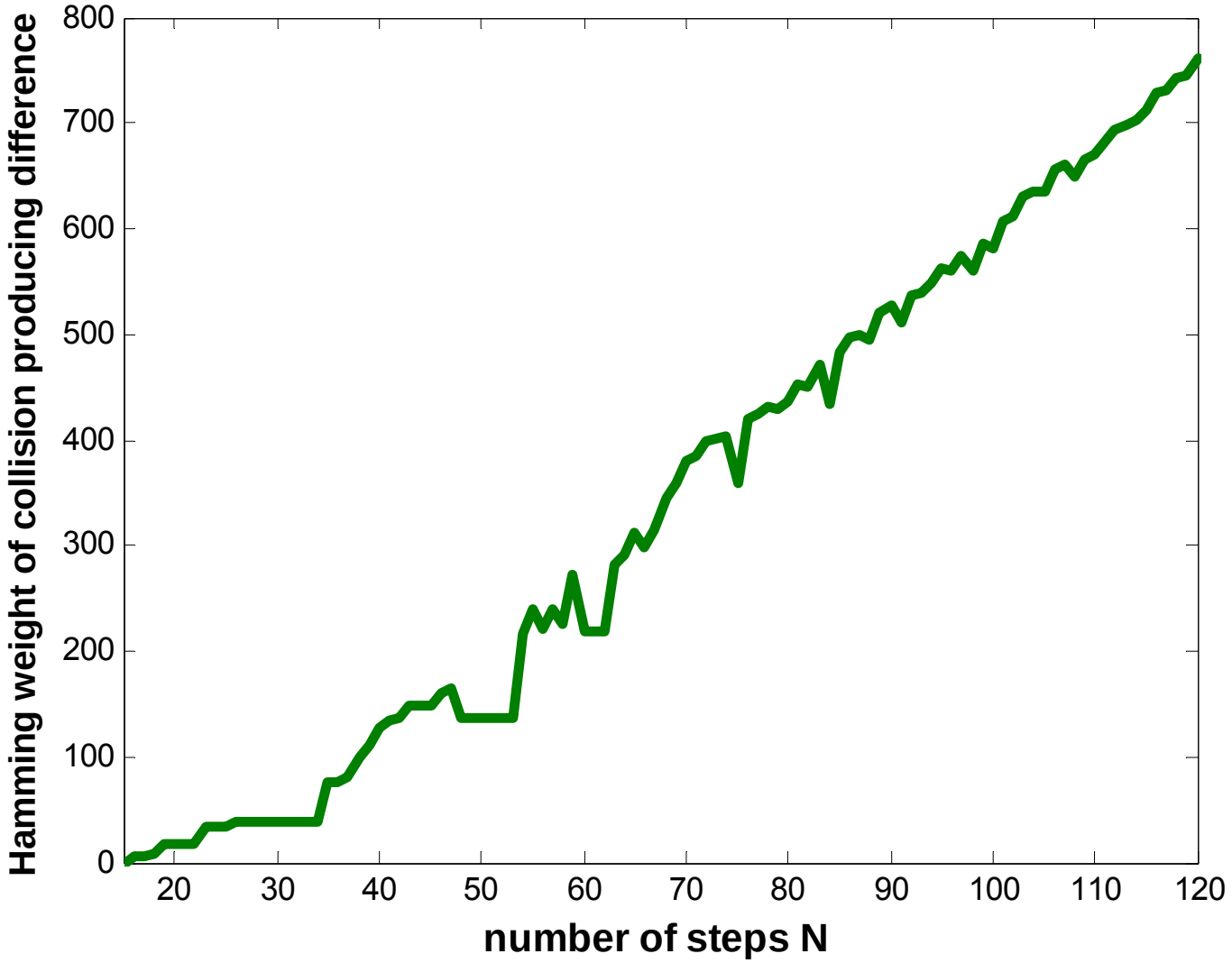
↓

Strong correlation between

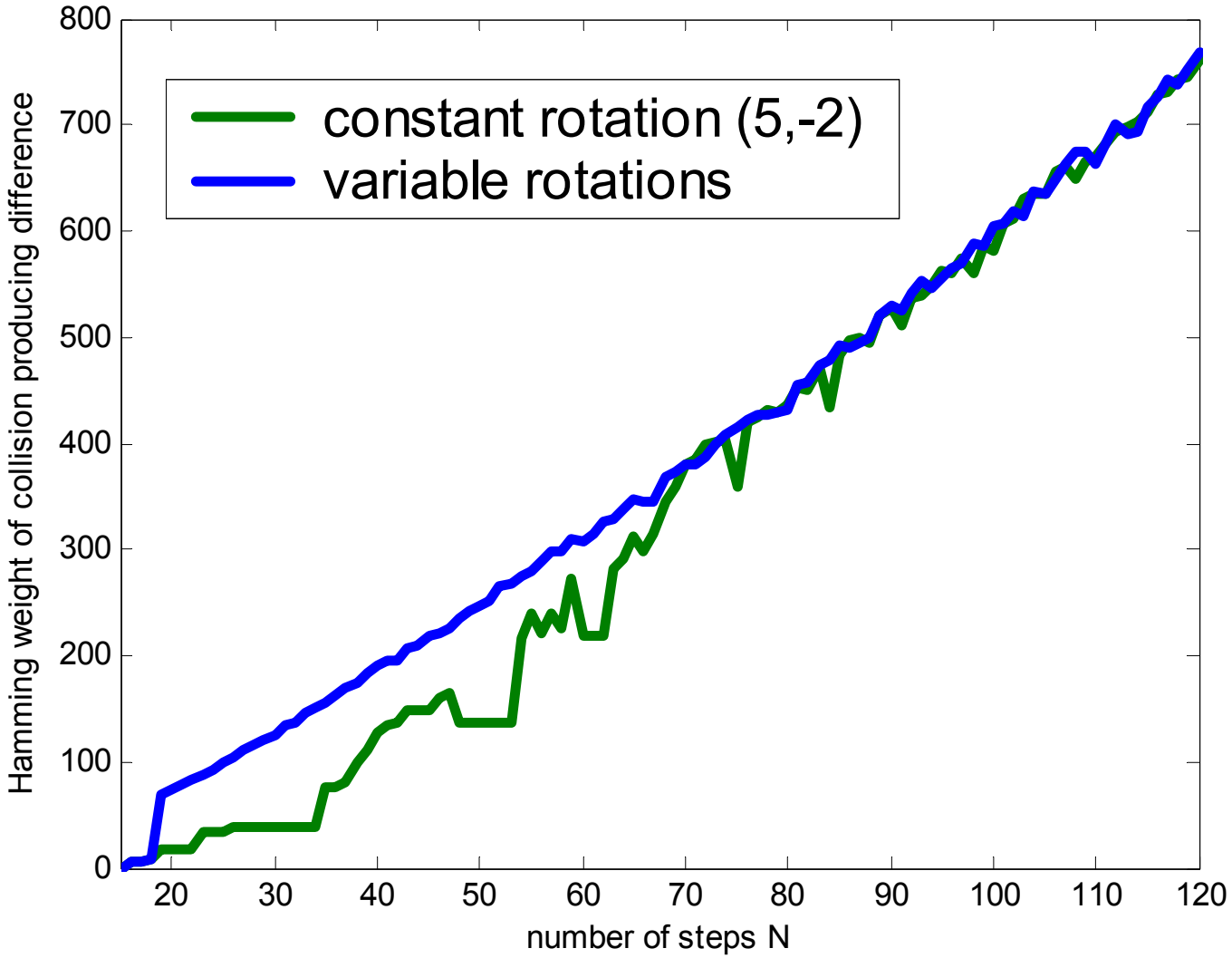
lowest weight of input difference(2) *and* #conditions(3)

Hamming weight of input difference is enough for our purposes

Hamming weights for variants of SHA-1



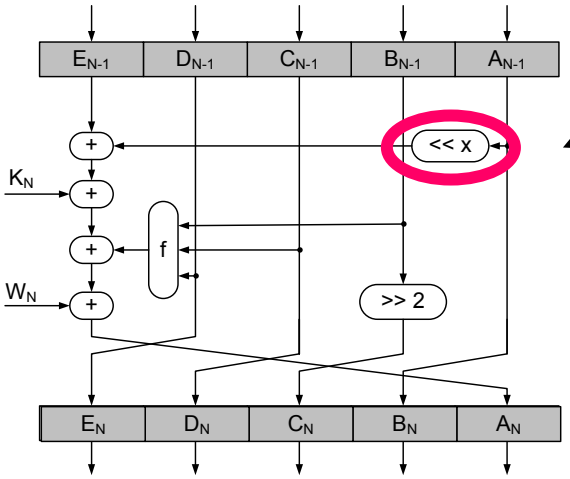
Comparison of constant and variable rotations



Overview

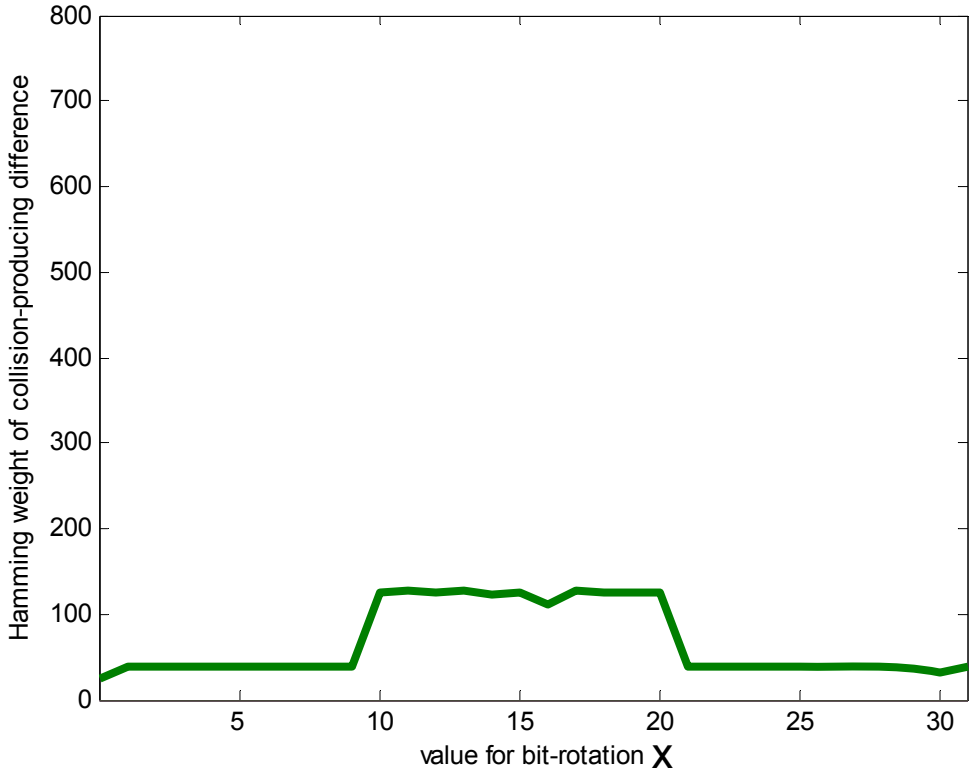
- Design history of SHA-1
- Review of basic design elements of SHA-1
  - Rotation in the Message Expansion
  - Recurrence-relation of the Message Expansion
  - Non-linear functions in the State Update
  - **Rotations in the State Update**
- Comparing variable and constant rotations
- Comparing constant rotations
- Conclusions

Comparing different constant rotations – Results for A

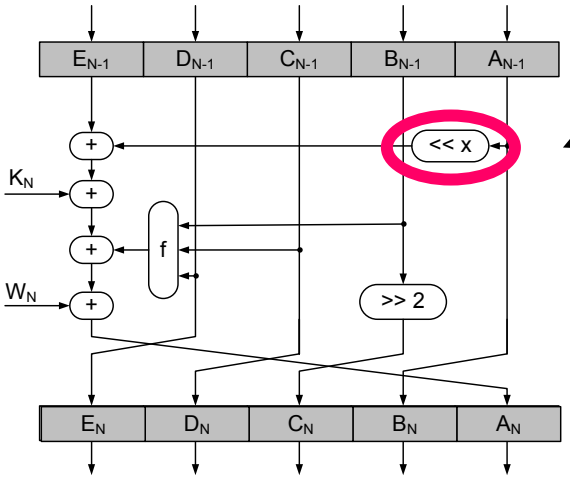


Rotating ***A*** by ***x*** bits to the left

$0 \leq x \leq 31$

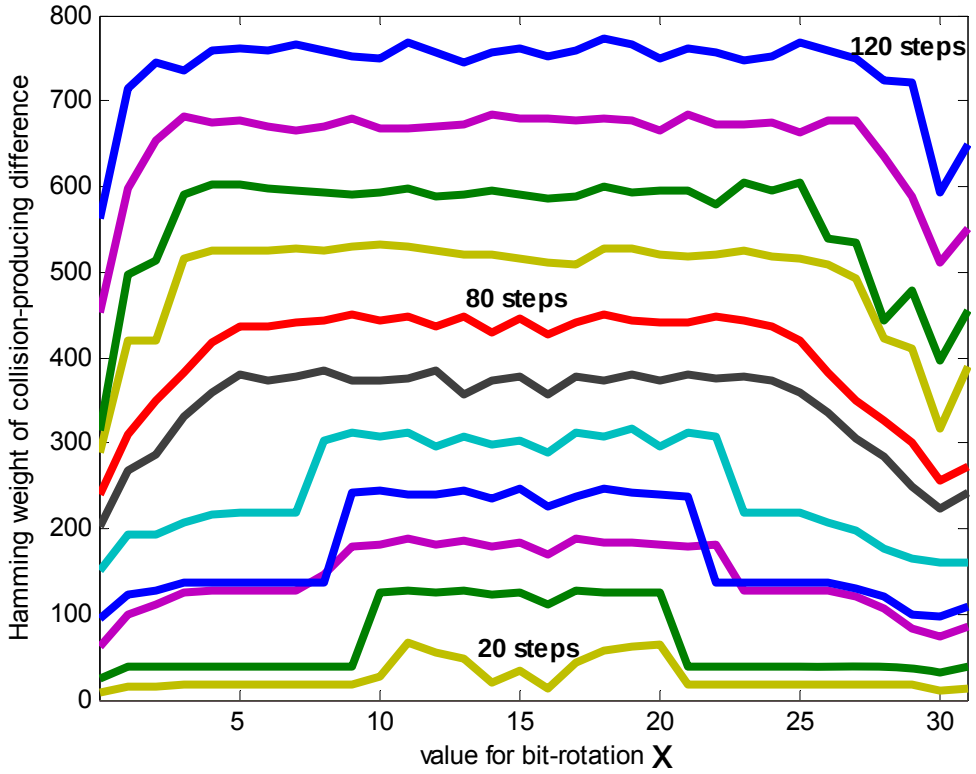


Comparing different constant rotations – Results for A

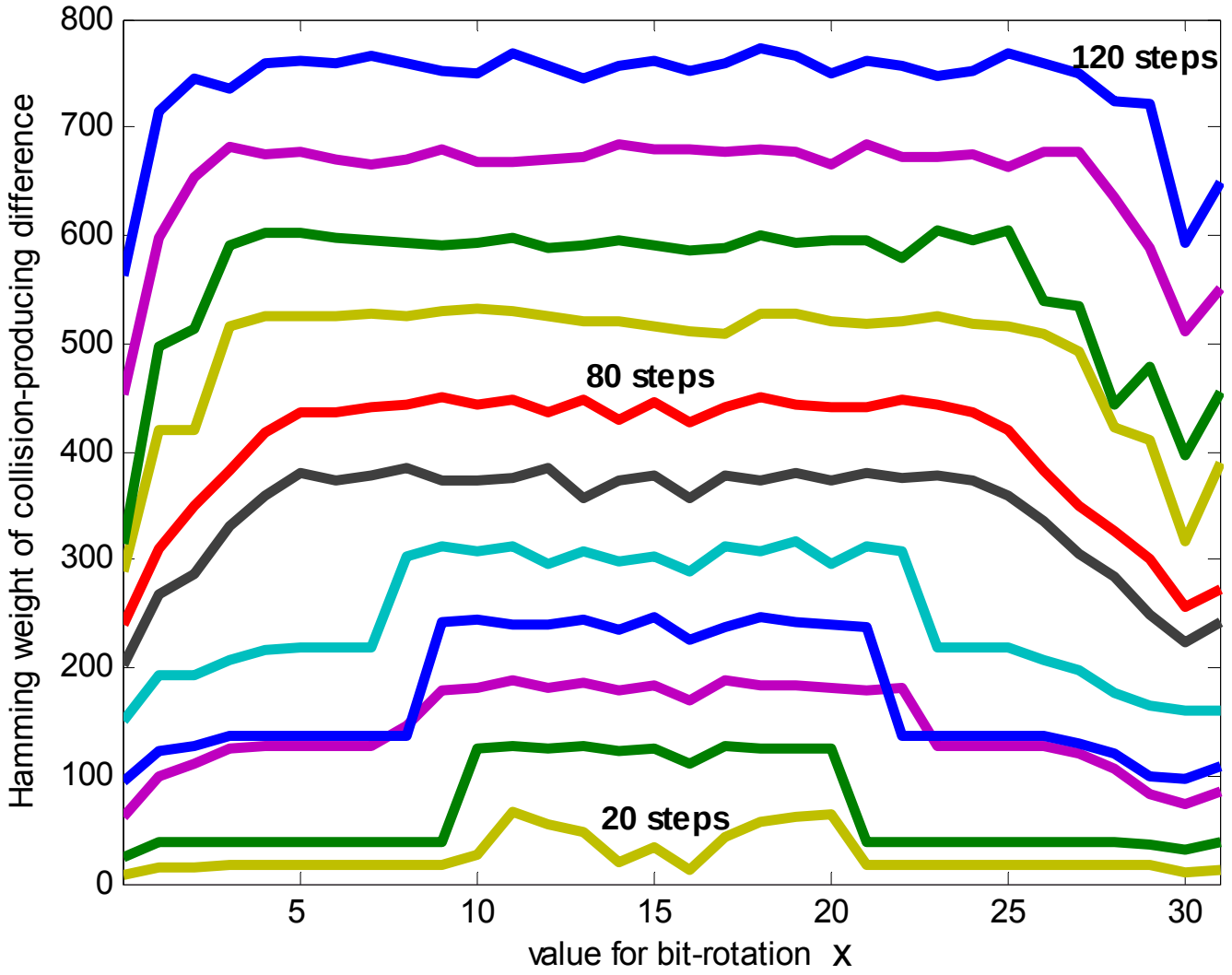


Rotating *A* by *x* bits to the left

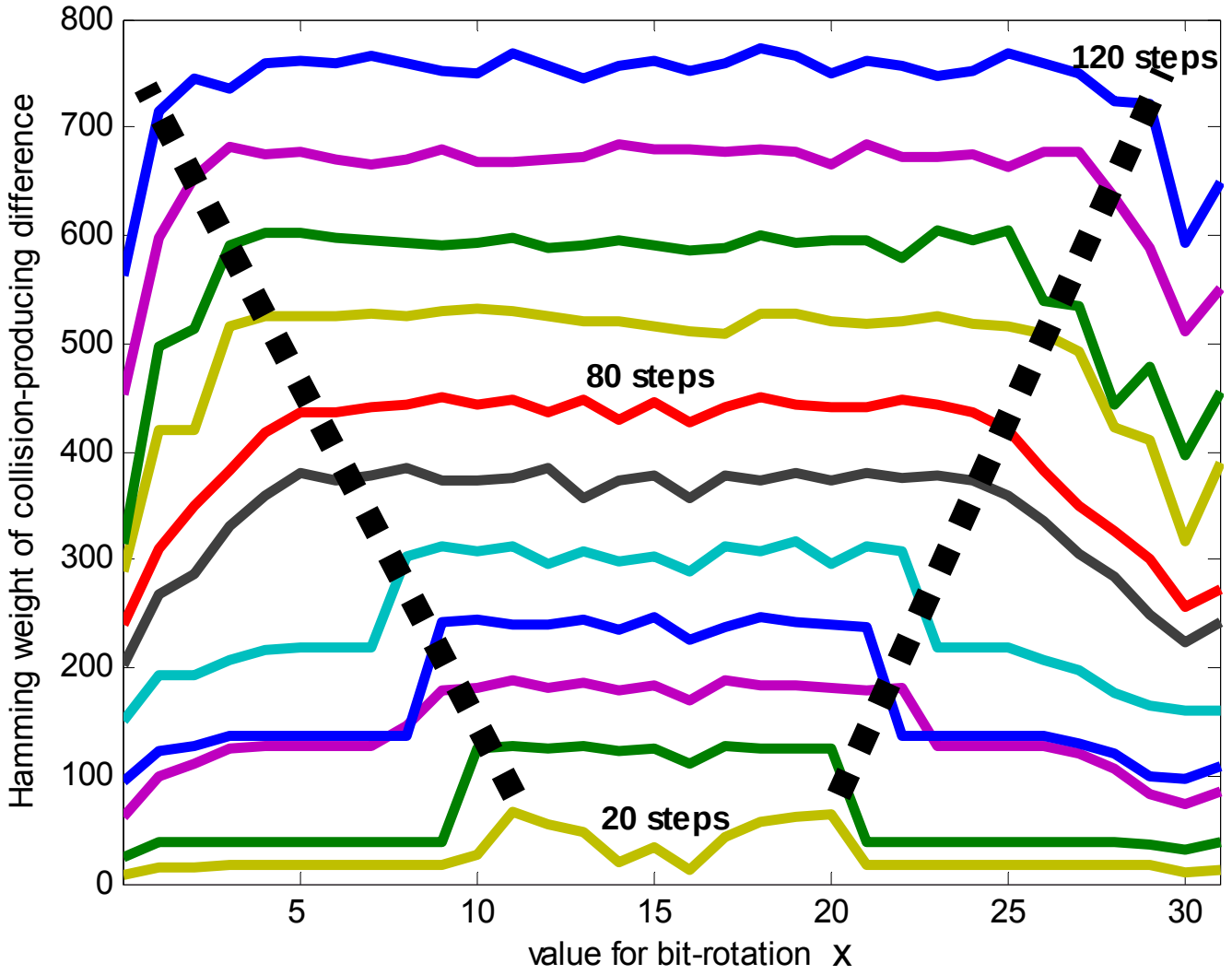
$0 \le x \le 31$



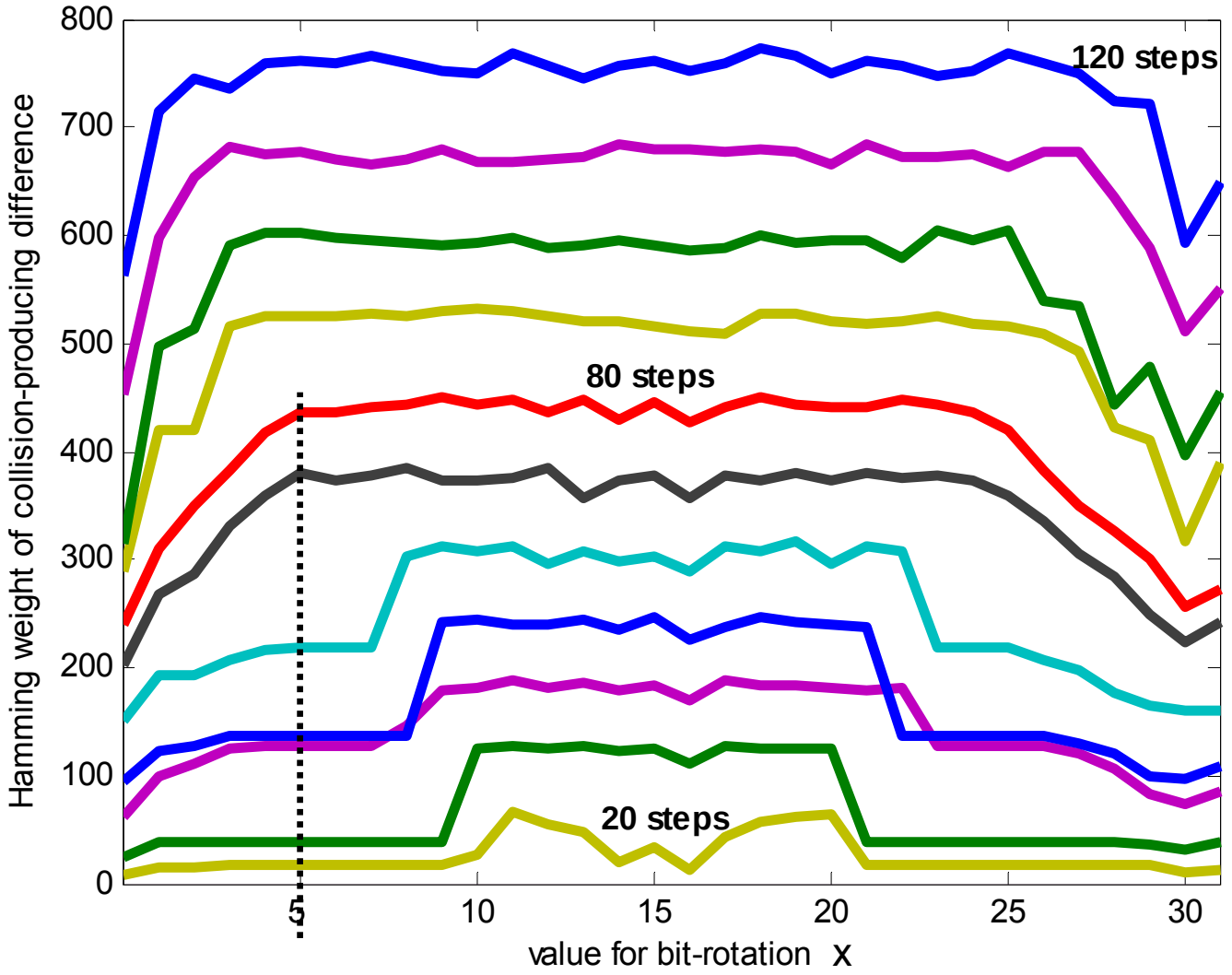
Comparing different constant rotations – Results for A



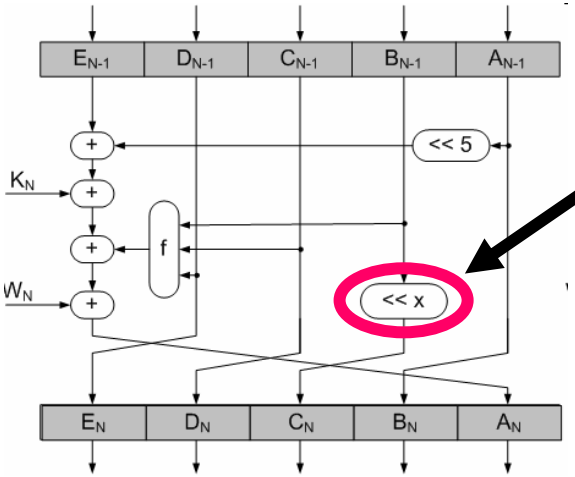
Comparing different constant rotations – Results for A



Comparing different constant rotations – Results for A

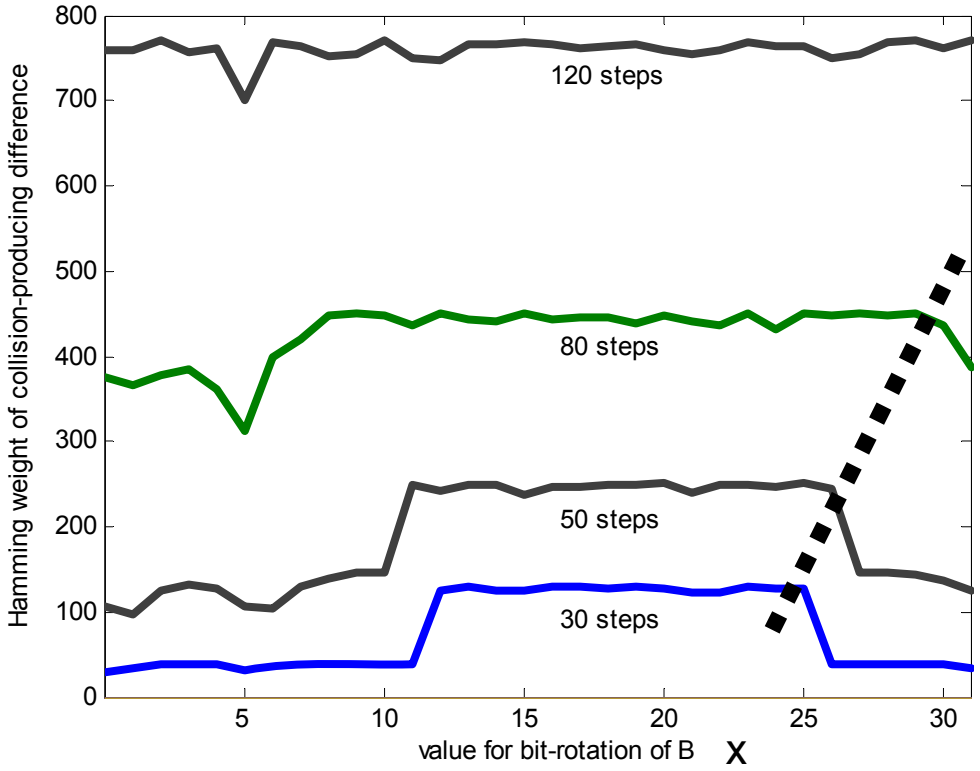


Comparing different constant rotations – Results for B



Rotating ***B*** by *x* bits to the left

$0 \le x \le 31$



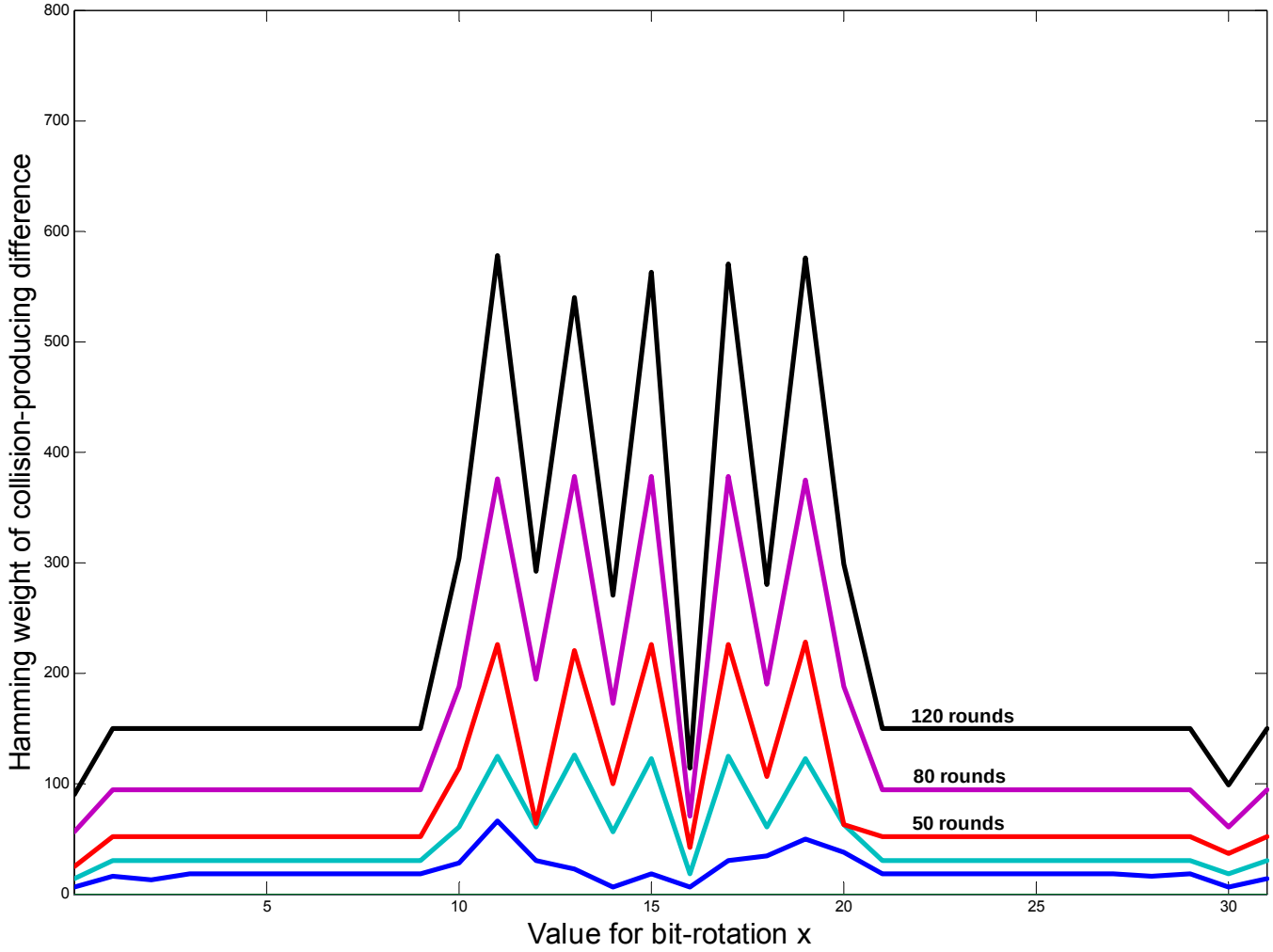
Why minimizing bit-rotations?

„*It‘s not only about security, it is also about performance*“

Consider platforms where constant-time shifter/rotators are not available.

e.g. i286, Pentium IV

Explanation does not hold for SHA-0



Overview

- Design history of SHA-1
- Review of basic design elements of SHA-1
  - Rotation in the Message Expansion
  - Recurrence-relation of the Message Expansion
  - Non-linear functions in the State Update
  - **Rotations in the State Update**
- Comparing variable and constant rotations
- Comparing different constant rotations
- Conclusions

Conclusions

- Design criteria for SHA-1 are unknown
- SHA-1 is based on MD4/MD5, but uses constant rotations instead of variable rotations
  - Both have advantages and disadvantages

- Particular values of the rotation (+5 and -2) could be the result of an optimization process
  - Given the 80 steps of the SHA-1 State Update, these are the minimal values to result in comparatively high attack complexities
  - Why? Advantages for platforms without constant-time shifters
  - Explanation does not hold for SHA-0 design