A Sage Library For Analysis Of Nonlinear Binary Mappings

Anna Maria Eilertsen, Oleksandr Kazymyrov,
Valentyna Kazymyrova, Maxim Storetvedt

Selmer Center, Department of Informatics,
University of Bergen, Norway

CECC'14
May 21, 2014

Outline

Substitutions

Definition

Substitution box (S-box) is an arbitrary mapping of one alphabet to another.

Substitutions for cryptography

S-boxes used in cryptography often map elements from vector space $\mathbb{F}_2^n$ to $\mathbb{F}_2^m$.

Substitutions

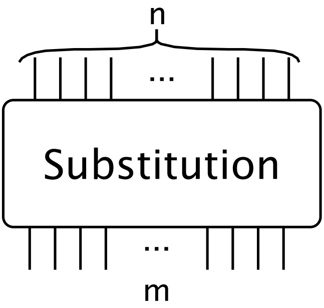


Figure : A Substitution Box

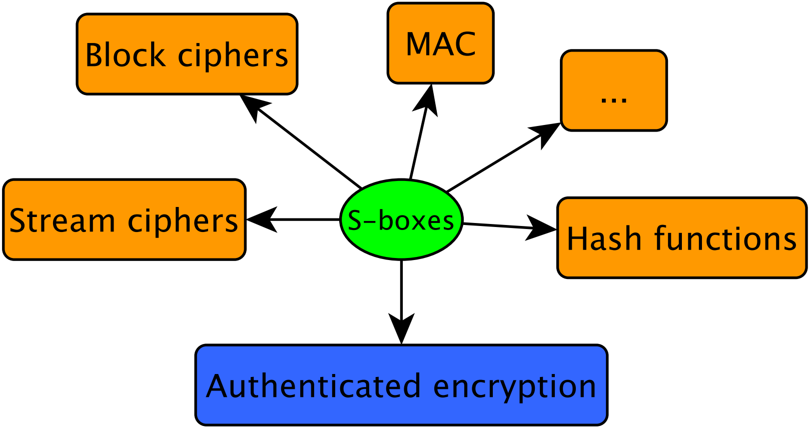
Possible variants

- $n > m$
- $n < m$
- $n = m$
  - $\#img(\text{S-box}) = 2^n$

Representations

- lookup tables
- vectorial Boolean functions
  - Boolean functions
- system of equations

Application of S-boxes



List of properties

Definition

An S-box is a mapping of an n-bit input message to an m-bit output message.

- Minimum degree
- Balancedness
- Nonlinearity
- Correlation immunity
- δ-uniformity
- Cyclic structure
- Algebraic immunity
- Absolute indicator
- Absence of fixed points
- Propagation criterion
- Sum-of-squares indicator
- ...

Cryptographic properties of S-boxes (1/5)

Definition

Let n and m be two positive integers. Any function $F : \mathbb{F}_2^n \mapsto \mathbb{F}_2^m$ is called an (n, m)-function or vectorial Boolean function.

δ-uniform

Arbitrary F is differentially δ-uniform if equation

$$b = F(x) + F(x + a),\ \forall a \in \mathbb{F}_2^n, \forall b \in \mathbb{F}_2^m, a \neq 0$$

has at most δ solutions.

Cryptographic properties of S-boxes (2/5)

Walsh transform

The Walsh transform of an (n, m)-function F at $(u, v) \in \mathbb{F}_2^n \times \mathbb{F}_2^m \backslash \{0\}$

$$\lambda(u, v) = \sum_{x \in \mathbb{F}_2^n} (-1)^{v \cdot F(x) \oplus u \cdot x}, \qquad (1)$$

where "$\cdot$" denotes inner products in $\mathbb{F}_2^n$ and $\mathbb{F}_2^m$ respectively.

Nonlinearity

$$NL(F) = 2^{n-1} - \frac{1}{2} \max_{v \in \mathbb{F}_2^{m*};\ u \in \mathbb{F}_2^n} |\lambda(u, v)|$$

Cryptographic properties of S-boxes (3/5)

Balancedness

An (n, m)-function F is called balanced if it takes every value of F_2^m the same number of times (2^{n-m}).

Absence of Fixed Points

A substitution must not have fixed point, i.e.

$$F(a) \neq a, \quad \forall a \in \mathbb{F}_2^n.$$

Cryptographic properties of S-boxes (4/5)

The algebraic normal form (ANF) of any (n, m)-function F always exists and is unique:

$$F(x) = \sum_{I \subseteq \{1,\dots,n\}} a_I \left(\prod_{i \in I} x_i \right) = \sum_{I \subseteq \{1,\dots,n\}} a_I x^I, \; a_I \in \mathbb{F}_2^m$$

The algebraic degree of F

$$deg(F) = \max\{|I| \mid a_I \neq 0\}$$

Minimum degree

The minimum algebraic degree of all the component functions of F is called the minimum degree.

Cryptographic properties of S-boxes (5/5)

Arbitrary substitution can be represented as the system of equations

$$\begin{cases} g_1(x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m) = 0; \\ g_2(x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m) = 0; \\ \ldots \\ g_r(x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_m) = 0. \end{cases} \tag{2}$$

Algebraic immunity

The algebraic immunity $AI(F)$ of any (n, m)-function F is the minimum algebraic degree of all functions in (2).

Outline

System for Algebra and Geometry Experimentation (Sage)

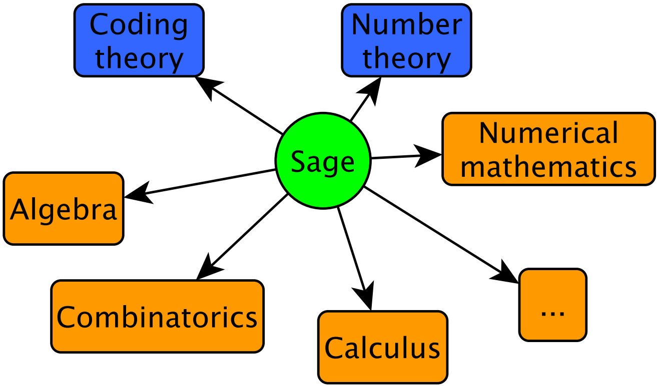


Figure : One can use Sage for ...

System for Algebra and Geometry Experimentation (Sage)

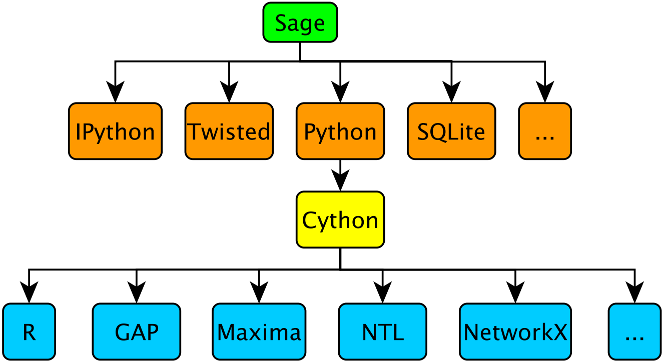
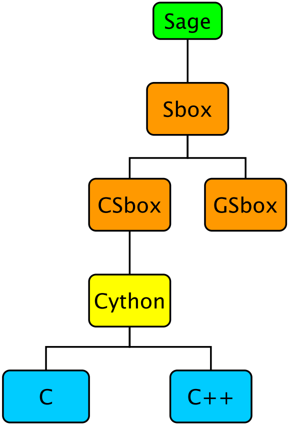


Figure : Sage components

General overview of the Sbox library



Design principles

- Orientation on arbitrary n and m
- Code optimization for performance
- Implementation of known cryptographic indicators

List of supported indicators (CSbox.sage)

- Minimum degree
- Balancedness
- Nonlinearity
- Correlation immunity
- δ-uniformity
- Cyclic structure
- Algebraic immunity
- Absolute indicator
- Absence of fixed points
- Propagation criterion
- Sum-of-squares indicator
- ...

Generation of substitutions (GSbox.sage)

- Gold
- Kasami
- Welch
- Niho
- Inverse
- Dobbertin
- Dicson
- APN for $n = 6$
- Optimal permutation polynomials for $n = 4$
- Polynomial
- ...

Unification of the functions

generate_sbox calls different methods based on parameters method and T which define generation method and equivalence respectively.

Additional functionality

- Extra functions
    - Resilience (balancedness and correlation immunity)
    - Maximum of linear approximation table
    - Check APN property (optimized)
- Convert linear functions to matrices and vice versa
- Apply EA- and CCZ-equivalence
- Generation of substitutions
    - Based on user-defined polynomial (trace supported)
    - Random substitution/permutation
    - With predefined properties
- Input/output
    - Set and get S-boxes as lookup tables
    - Get univariate representation/system of equations
    - Convert polynomial to/from internal representation

Outline

An example

Theorem (Browning, K. A., et al/Budaghyan, L.)

Let α be a multiplicative generator of $\mathbb{F}_{2^6}$ with irreducible polynomial $f(x) = x^6 + x^4 + x^3 + x + 1$. Then the APN function

$$F(x) = \alpha x^3 + \alpha^5 x^{10} + \alpha^4 x^{24}$$

is CCZ-equivalent to an APN permutation over F_{2^6} with $\mathcal{L}(x, y) = (tr_{6/3}(\alpha^4 x) + \alpha tr_{6/3}(y), tr_{6/3}(\alpha x) + \alpha tr_{6/3}(\alpha^4 y))$, where $tr_{6/3} = x + x^{2^3}$, $y = F(x)$.

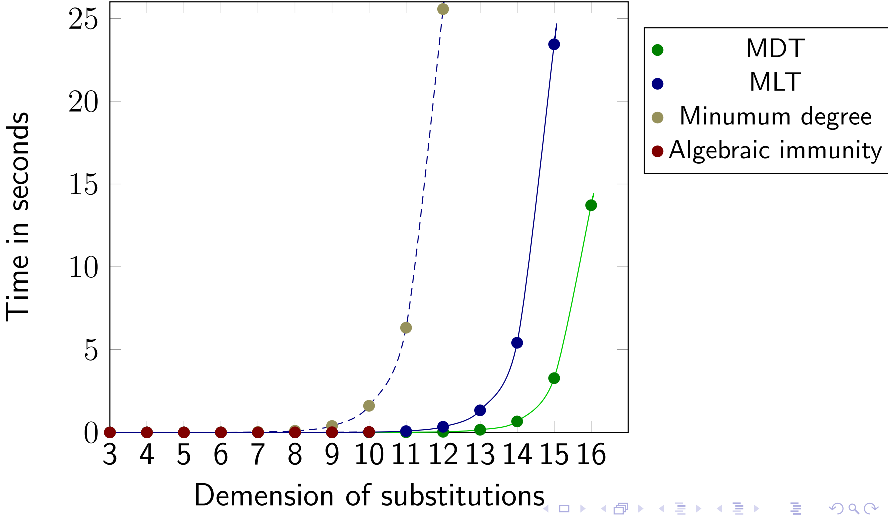
An example

```
sage: %runfile ./Sbox.sage
sage: S = Sbox(n=6,m=6)
sage: P = S.get_ring()
sage: g = S.get_mg()
a
sage: tr = S.Tr_pol(x=P("x"),n=6,m=3)
sage: tr
x^8 + x

sage: M1 = S.l2m(tr.subs(P("(%s)*x"%(g^4))))
sage: M2 = S.l2m(g*tr)
sage: M3 = S.l2m(tr.subs(P("(%s)*x"%(g))))
sage: M4 = S.l2m(g*tr.subs(P("(%s)*x"%(g^4))))
```

An example

```
sage: F = "g*x^3+g^5*x^10+g^4*x^24"
sage: S.generate_sbox(method="polynomial",G=F,T=
    ↪ "CCZ",M1=M1,M2=M2,M3=M3,M4=M4)
sage: S.is_bijection()
True
sage: S.is_APN()
True
sage: S.MDT()
2

sage: S = Sbox(n=6,m=6)
sage: S.generate_sbox(method='APN6')
sage: S.is_bijection()
True
sage: S.is_APN()
True
```

Performance



Comparison of known substitutions

Properties	AES	GOST R 34.11-2012	STB 34.101.31-2011	Kalyna's S-boxes	Next-generation S-boxes
δ-uniformity	4	8	8	8	8
Nonlinearity	112	100	102	96	104
Absolute Indicator	32	96	80	88	80
SSI	133120	258688	232960	244480	194944
Minimum Degree	7	7	6	7	7
Algebraic Immunity	2 (39)	3 (441)	3 (441)	3 (441)	3 (441)

Conclusions

- A high performance library to analyze and generate arbitrary binary nonlinear mappings
- Lots of cryptographic indicators and generation functions are included
- Functionality can be expanded quite easily
- Under development
- Hard to run for the first time
  - Works only in consoles
- Source code: `https://github.com/okazymyrov/sbox`