БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ
ИНСТИТУТ ПО МАТЕМАТИКА И ИНФОРМАТИКА

# А В Т О Р Е Ф Е Р А Т

**на**

# Д И С Е Р Т А Ц И Я

# НОВИ ЕВРИСТИЧНИ МЕТОДИ ЗА ГЕНЕРИРАНЕ НА БИЕКТИВНИ ЗАМЕСТИТЕЛНИ ТАБЛИЦИ С ДОБРИ КРИПТОГРАФСКИ СВОЙСТВА

**на**

**Георги Велков Иванов**

докторант на самостоятелна подготовка

за придобиване на научната и образователна степен
„ДОКТОР“
по научна специалност
01.01.12 - информатика

**Научни консултанти:**
доц. д-р Светла Никова
доц. дмн Емил Колев

2016 г.

# С Ъ Д Ъ Р Ж А Н И Е

**Увод** 2

**Обзор на дисертацията** 11

**Апробация на резултатите** 23

**Авторска справка** 24

**Благодарности** 26

**Публикации по дисертацията** 26

**Списък с цитирания** 27

**Библиография** 29

# Увод

В динамичното настояще, обусловено най-вече от развитието на информационните технологии, осигуряването на адекватна защита на информацията във всичките ѝ форми е от все по-жизненоважно значение. Освен потенциалните финасови загуби, пробиването на тази защита може да доведе и до загуба на човешки животи.

Математическата дисциплина, която отговаря за защитата на информацията, е криптографията. От друга страна, криптоанализът се стреми да пробие тази защита. Криптографията и криптоанализът вървят паралелно и формират криптологията.

Модерните криптосистеми, които са съставени от необходимите добри блокови или поточни шифри и хеш функции, гарантирт защитата на информацията във всичките ѝ аспекти. Най-важните сред тях са конфиденциалност, автентификация и интегритет. Осигуряването на конфиденциалност на информацията означава, че съдържанието ѝ е достъпно само за оторизирано откриване. В противен случай, когато е осъществен нерагламентиран достъп, то се пази в тайна и е абсолютно незначещо. Наличието на интегритет означава, че със сигурност информацията не е била променяна - било то преднамерено или непреднамерено. Автентификацията е процес, при който се проверява автентичността на субекта, който изпраща информацията. За да може да бъде устойчива на мощните статистически атаки, като линеен и диференциален криптоанализ [2, 3, 16], една криптосистема трябва да включва в себе си подходящите криптографски алгоритми (шифри), които да осигуряват достатъчно високи нива на *confusion* и *diffusion* в процеса на криптиране [25]. Наличието на *confusion* означава, че откритият текст драстично се променя при криптирането и че връзката му със закрития текст е много сложна. Добрата *diffusion* означава, че малки промени в открития текст ще доведат до големи промени в закрития текст (лавинен ефект). Най-лесният начин за осигуряване на *confusion* е чрез добавяне на нелинейност към криптиращия процес. Тъй като линейните операции са лесно обратими, то колкото по-нелинейна е връзката между явния и закрития текст, толкова по-сложна ще е и връзката между тях. Обикновено, единствените нелинейни компоненти на блоковите шифри са заместителните таблици (S-boxes), съпоставящи на всеки $m$ входни бита други $n$ изходни такива. Поради това, те са единствените, които внасят *confusion* в криптиращия процес, което ги прави от жизненоважно значение за криптографската устойчивост на блоковия шифър като цяло. Биективните S-boxes, от своя страна, са особено интересни, защото представляват пермутация на $n$ входни бита и са обратими. За да може да се осигури нужната устойчивост на блоковия шифър към атаките на криптоанализа, изграждащите го S-boxes трябва да удовлетворяват определен

набор от критерии и да притежават някои основни криптографски свойства. Най-често, за да се счита един S-box за добър от криптографска гледна точка, е необходимо той да притежава следните криптографски свойства: *баланс* (*balance*), висока *нелинейност* (*nonlinearity*), ниска *differential uniformity*, ниска *автокорелация* (*autocorrelation*), висока *алгебрична степен* (*algebraic degree*), липса на *фиксирани точки* (*fix points*) и др.

Всяка $(m \times n)$ S-box $S$ може да се разглежда като векторна булева функция, която е съставена от $m$ булеви функции: $S = (f_1, f_2, \ldots, f_m) : B^n \to B^n$, където $f_i : B^n \to B$, за $i = 1, 2, \ldots, m$, се наричат координатни булеви функции на $S$. За да притежава $S$ горните свойства, задължително координатните й булеви функции $f_1, f_2, \ldots, f_m$ също трябва да ги притежават. Важно е да се отбележи обаче, че това е необходимо, но не е достатъчно условие. Необходимо е още и всички линейни комбинации на координатните булеви функции с изключение на нулевата да ги притежават. Тези функции се наричат компонентни булеви функции на $S$ и се означават с $g_v(x) = v_1 f_1(x) \oplus \ldots \oplus v_m f_m(x)$, за $v \in B^m \setminus \{0\}$. Компонентните булеви функции на всеки S-box играят важна роля при преминаването от многомерния към едномерния случай и превеждането на теорията на блоковите шифри и на заместителните таблици на езика на булевите функции.

Поради тяхната особена важност за криптографската устойчивост на блоковите шифри, S-boxes са се изучавали много усилено през последните няколко десетилетия. В литературата са налични много методи за S-box генериране, но всички те могат да се обособят в три основни класа: алгебрични конструкции, псевдо-случайно генериране и евристични техники.

Алгебричните конструкции се основават предимно на математически принципи и вече доказани връзки. Например, биективният $(8 \times 8)$ S-box на AES (Advanced Encryption Standard) се базира на обратна функция в крайното поле $GF(2^8)$, която е последвана от афинна трансформация за премахване на фиксираните точки [8]. Биективните $(n \times n)$ S-boxes, базирани на обратна функция в крайното поле $GF(2^n)$, за четни $n > 6$, са най-добрите намерени до момента и притежават оптималната, поне засега, комбинация от основни критпографски свойства: $N_{inv} = 2^{n-1} - 2^{\frac{n}{2}}$, $\delta_{inv} = 4$, $deg(S_{inv}) = n - 1$ и $AC(S_{inv})_{max} = 2^{\frac{n}{2}+1}$ [8, 22]. Тези стойности не са оптимално възможните. Такива са стойностите на така наречените *перфектно нелинейни* или *bent* $(n \times m)$ S-boxes $S(n, m)$: $N_{opt} = 2^{n-1} - 2^{\frac{n}{2}-1}$, $\delta_{opt} = 2^{n-m}$ и $AC(S_{opt})_{max} = 0$ [21]. Те обаче не са балансирани, не могат да са пермутации, т.е. биективни, съществуват само за четни $n \geq 2m$ и са с ниска алгебрична степен $deg(S_{opt}) \leq \frac{n}{2}$. Поради това, *bent* S-boxes не се използват в практиката. Намирането на биективен $(n \times n)$ S-box $S$, който да има нелинейност $N_S \in (N_{inv}, N_{opt})$,

е нерешена задача в криптографията през последните няколко десетилетия. По тази причина мнозина смятат, че биективните $(n \times n)$ S-boxes, базирани на обратна функция в крайното поле $GF(2^n)$, са оптималните, които съществуват. Но докато не се намери доказателство за несъществуването на S-boxes, които са по-добри от $S_{inv}$, няма защо да се изключва възможността за тяхното съществуване. Друг отворен криптографски проблем е намирането на биективен $(n \times n)$ S-box с $\delta = 2$, където $n$ е четно и по-голямо от 6. Тези S-boxes се наричат *почти перфектно нелинейни пермутации* (*almost perfect nonlinear permutations - APNP*) [21]. Единственият такъв S-box, който е намерен досега, е $(6 \times 6)$ *APN* пермутацията на Dillon от [4]. Въпреки че $S_{inv}$ са най-добрите намерени S-boxes, то те са ограничен брой и всичките са афинно еквивалентни. Абсолютно всички компонентни булеви функции на всяка една S-box от типа $S_{inv}$ или нейна производна след афинна трансформация са от един и същи клас на еквивалентност. Наличието на тази проста алгебрична структура и липсата на каквото и да е разнообразие сред тях би било пагубно за всички такива S-boxes при евентуален успешен криптоанализ на кой да е S-box $S_{inv}$. Към настоящия момент тази опасност е чисто имагинерна, но темпът на развитие на алгебричния криптоанализ [7] е сигурен знак за сериозни притеснения в близко бъдеще и създава поле за размисъл на S-box дизайнерите. Fuller в [9] дефинира понятието *линейна остатъчност* (*linear redundancy*) на един $(n \times m)$ S-box $S$, определено на база брой различни класове на еквивалентност, към които принадлежат всичките му $2^m - 1$ компонентни булеви функции. Колкото са повече тези класове, толкова е по-малка *линейната остатъчност* на $S$, а оттам и толкова по-сложна и различна е алгебричната структура, т.е. толкова е по-добре. Очевидно, всеки S-box от типа $S_{inv}$ притежава пълна *линейна остатъчност*, защото компонентните му булеви функции са от един единствен клас на еквивалентност [1, 27]. *Tweaking* методът, базиран на инвертиране на двойки елементи на $S_{inv}$ с цел получаване на нов S-box, който е с нулева *линейна остатъчност*, е представен в [9]. В случая при $n = 8$, най-добрият получен S-box $S$ е с комбинация от свойства: $(N_S,\ \delta_S,\ deg(S),\ AC(S)_{max}) = (106, 6, 7, 56)$, която значително остъпва на съответната комбинация на $S_{inv}$: $(N_{S_{inv}},\ \delta_{S_{inv}},\ deg(S_{inv}),\ AC(S_{inv})_{max}) = (112, 4, 7, 32)$. С дефинирането на *линейната остатъчност* възниква още един въпрос, който може също да се разглежда като отворен криптографски проблем. Той е: "Съществува ли биективен $(n \times n)$ S-box, който да притежава същата комбинация от свойства като на $S_{inv}$, но да няма пълна *линейна остатъчност*"? Такъв S-box, на който компонентните булеви функции принадлежат дори и само на два различни класа на еквивалентност, ще е еднакво добър както $S_{inv}$ по отношение на всички криптографски критерии, няма да е еквивалентен на него и ще има по-сложна алгебрична структура, която ще е толкова по-добра, колкото различните класове са повече на брой.

Псевдо-случайното генериране е най-наивният от трите подхода за S-box генерация. Спокойно може да бъде приобщен към случайното търсене, което го обрича на сигурен неуспех, особено когато размерностите растат. Броят на всички булеви функции на $n$ променливи е $2^{2^n}$, докато този на $(n \times m)$ S-boxes е $(2^m)^{2^n}$ и съответно на биективните $(n \times n)$ S-boxes сред тях $(2^n)^{2^n}$. Още когато $n = 5$, случайно да се уцели S-box, който да е добър от криптографска гледна точка при явната противопоставеност на повечето от критериите, е много малко вероятно. Обикновено, методът се състои от две основни техники. При първата, S-box стойностите се попълват от някаква таблица със случайни числа, което е крайно неефективно. Вероятността за това да се попадне на балансиран S-box е много малка, а какво остава ако трябва да са удовлетворени и всички останали криптографски свойства. Втората техника е по-надеждна от първата, защото при нея направо се генерира биективен S-box, който впоследствие се оценява спрямо останалите критерии. Техниката е по-успешна от първата, но е все така неефективна по отношение на крайния резултат. Напрмер, в случая на биективните $(8 \times 8)$ S-boxes, намерените най-добри представители са с *нелинейност* 100 [17, 18], която е далеч от $N_{S_{inv}} = 112$, при това без да се отчитат останалите критерии.

Евристичните техники донякъде могат да се асоциират със случайното търсене. Основната разлика е, че при тях търсенето се осъществява не в цялото пространство, а в някаква близка околност на предварително подбрана начална точка. Основната им цел е итеративно да се подобрят едно или повече свойства на някакво начално множество от S-boxes и в резултат да се получат голям брой по-добри S-boxes. Обикновено, получените решения са с по-лоши криптографски свойства в сравнение със свойствата на алгебрично конструираните S-boxes, но за сметка на това техният брой е значително по-голям и имат доста по-сложна и разнородна алгебрична структура. Макар и по-лоши, генерираните чрез евристични техники S-boxes са близки до най-добрите алгберични конструкции, както и също така са далеч по-добри от псевдо-случайно генерираните. Най-известните евристични техники са: *hill climbing* метод [17, 18], *simulated annealing* метод [6], както и множество генетични и имунни алгоритми [5, 19, 26]. Често в литературата се срещат и различни техни вариации и комбинации [13, 14, 18].

**Мотивация**

Често в практиката се налага да се използват множество различни S-boxes, които притежават еднакви или поне близки критпографски свойства. Обиновено, причината за това е необходимостта от децентрализация и разделяне на различни равнопоставени критпографски мрежи, или пък се желае тези S-boxes да служат като дълговременни

криптографски ключове, които да се подменят регулярно. Такъв набор от S-boxes не може да бъде получен нито чрез алгебрични конструкции, нито чрез псевдо-случайно генериране. Първите са много добри, но са малък брой и всичките са еквивалентни, а вторите са доста слаби от гледна точка на криптографските си свойства. Единствената оставаща възможност е да се търси някакъв евристичен подход, който да даде нужното разнообразие. Сред всичките налични в литературата евристични техники, двете, които успяват да получат най-добрите резултати по отношение на основните криптографски критерии в най-разпространения случай на $(8 \times 8)$ биективните S-boxes, са *Special genetic algorithm* [26] и *Gradient descent method* [14]. Първият успява да постигне максимална *нелинейност* 104, като не отчита останалите критерии, докато комбинацията от свойства на най-добрия S-box, получен от втория алгоритъм, е $(N, \delta, deg, AC_{max}) = (104, 8, 7, 80)$. Очевидно, ако се открие нов евристичен метод, който успешно да генерира достатъчно голямо количество нееквивалентни биективни $(n \times n)$ S-boxes с близки криптографски свойства до тези на $S_{inv}$, то той ще реши много задачи, свързани с горните сценарии. И това се отнася предимно за случая $n = 8$. Освен $S_{inv}$, които се генерират лесно и при $n > 8$ предвид известните неразложими полиноми за съответното крайно поле, за никой друг от известните в литературата методи не е известно да е успял да получи добри S-boxes, които са с размерности по-големи от $(8 \times 8)$. Това се дължи на особено тежките изчисления, свързани с размера на претърсваното пространство, който нараства много бързо с увеличаване на $n$. Стандартният подход отдолу-нагоре, където се стартира от някакъв случаен S-box, който стъпка по стъпка се подобрява, е недостатъчно ефективен по отношение качеството на получените решения още при $n = 8$. На настоящия етап на развитие на информационните технологии и при текущото състояние на изчислителната техника, всеки метод, различен от алгебрична конструкция и базиран на стандартния подход отдолу-нагоре, е обречен на неуспех при $n > 8$. С оглед успешно преодоляване на тези естествени ограничения, при по-големи размерности е необходимо използването на някакъв подход, който да е подобен на *tweaking* метода [9] и който да стартира от някое по-добро междинно положение, прескачайки достатъчно голям брой стъпки и сметки. Естествено, този подход ще може да се прилага и при $n = 8$, в случай че е по-успешен от останалите методи. За тази размерност е възможно, също така, да се търси и някакъв принципно нов метод, който да работи отдолу-нагоре и който да е по-успешен от всички известни методи от този тип досега.

**Цели**

Основната цел на изследванията, представени в настоящата дисертация, е свързана със създаване на ефективен инструментариум от няколко нови евристични метода за

генериране на много на брой биективни S-boxes, които са с размерности от $(6 \times 6)$ до $(16 \times 16)$ и които притежават близки до $S_{inv}$ криптографски свойства. Фрагментирана на по-малки съставни части, съответстващи на конкретните стъпки, които съпътстват постигането ѝ, тази цел се свежда до:

1. Подобряване на цялостното разбиране за сложните връзки и зависещото от тях взаимодействие между криптографските свойства на булевите функции и S-boxes.

   Въпреки че множество методи и техники за S-box генериране са изследвани през последните години, конструирането на S-boxes, които едновременно удовлетворяват в оптимална степен желаните криптографски критерии, е абсолютно невъзможно поради големия брой криптографски свойства, които са в конфликт помежду си. Очевидно, в такъв случай се търсят S-boxes, които са оптимални по отношение на криптографските критерии, имащи най-голяма тежест, и субоптимални спрямо по-маловажните и второстепенни критерии. Противоречието между критериите, в допълнение с изключително бързото нарастване на претърсваното пространство и на необходимите ресурси за тежките изчисления при увеличение на размерностите, превръщат дизайна на добри S-boxes в изкуство.

2. Прилагане на ефективна комбинация от известни евристични методи, за да може да се оптимизират най-ценните сред криптографските свойства.

   Целта е да се използва цялото наследено от известните евристични методи знание, което да се комбинира по подходящ начин, за да се получат нови S-boxes, които, ако не по-добри, то поне са еднакви по отношение на криптографските си свойства с най-добрите резултати на методите от този тип.

3. Разработване на нови евристични методи за генерация на биективни S-boxes, които са с размерности от $(6 \times 6)$ до $(16 \times 16)$ и които са криптографски силни.

   Новите евристични методи евентуално ще се получат вследствие на най-добрите резултати, получени от различните комбинации с евристични техники. Основната трудност, съпътстваща процеса на избор на най-ефективен метод за генериране, е свързана с нарастване на размерностите и конфликтите между криптографските критерии. В допълнение на изискването получените S-boxes да бъдат оптимални спрямо съществените критерии и субоптимални спрямо маловажните, те трябва

още да са много на брой и от всички размерности между $(6 \times 6)$ и $(16 \times 16)$.

4. Определяне на ефективността на разработените нови методи при производството на S-boxes по отношение на размерности, скорост и брой получени решения.

   В съответствие с възможните ефективни S-box размерности в интервала $[6, 16]$ и голямото разнообразие на получените S-boxes, е необходимо да бъдат извършени огромен обем изследвания и тестове, за да се определи степента на успех на новите генериращи методи.

5. Повишаване криптографската устойчивост на блоковите шифри чрез генериране и последващо имплементиране в тях на разноородни криптографски силни S-boxes от различни размерности.

   Целта е да се използват получените S-boxes и да се имплементират в съществуващи блокови шифри, за да се получи поне същата криптографска устойчивост.

**Принос**

В съответствие със зададените цели, приносът на извършените изследвания може да се обособи и сведе до изпълнението на следните по-малки стъпки:

1. След запознаване с взаимовръзките между отделните криптографски свойства в рамките на булевите и на векторните булеви функции сами по себе си, както и с тясната корелация между критериите в едномерния и многомерния случай, цялата необходима информация, която помага за показване курса на действие при дизайна на нови методи за S-box генериране, е налице и спомага за отсяването на всички безперспективни подходи.

2. С изключение на алгебричните конструкции от типа $S_{inv}$, които са ограничен брой, всичките са еквивалентни и до голяма степен притежават една и съща алгебрична структура, като в същото време са с най-добрата открита до момента комбинация от криптографски свойства, за всички останали методи са проведени значителен брой изпитания. Изпробвани са различни комбинации сред тях, в това число и при различни размерности. Максималната получена *нелинейност* в случая $n = 8$ е 104, докато минималните стойности за *автокорелация* и $\delta$-*uniformity* са съответно 80 и

8. Що се отнася до едновременното удовлетворяване на основните криптографски критерии, с нито един метод или комбинация от методи не успяхме да получим $(8 \times 8)$ биективен S-box, който поне малко да се доближава до $S_{inv}$. Ситуацията става още по-лоша в случаите когато $n > 8$, заради по-голямото пространство, което се претърсва. Тогава, инспирирана от *tweaking* метода [9], се роди идеята за генетичен алгоритъм, работещ в обратен ред.

3. Идеята, която стои зад първия от двата нови методи за генериране, е базирана на реверсивен гентичен алгоритъм. Алгоритъмът започва своята работа от начална популация от $(n \times n)$ биективни S-boxes от типа $S_{inv}$ и техни производни след прилагане на афинна трансформация и търси S-boxes, които са близки до тях по отношение на основните криптографски свойства. Всеки един от трите различни варианта на алгоритъма успява относително бързо да произведе голям брой от биективни S-boxes, които са от всички размерности от $(8 \times 8)$ до $(16 \times 16)$, имат близки свойства до тези на $S_{inv}$, но по-сложна алгебрична структура от тях, и притежават малка, а в най-добрия случай и нулева, *линейна остатъчност*.

4. От гледна точка на ефективност на първия метод за генериране, всеки от трите му варианта показаха по време на изпитанията, че са достатъчно ефективни. С всеки следващ вариант ефективността по отношение на качеството на получените решения се повишаваше за сметка на скоростта на алгоритъма, която значително намаляваше с всяко добавяне на нова ценова функция и най-вече след всяко едно увеличаване на размерността.

5. Независимо от качеството на получените с първия генериращ метод S-boxes, все още много дизайнери на блокови шифри са притеснени от неслучайния им произход и от наличието на известно сходство в алгебричната им структура. Опасявайки се от наличие на скрита функционалност (задни врати - trap-doors) или от възможна уязвимост за в бъдеще към алгебрични атаки [7], те предпочитат да използват случайно генерирани S-boxes. След като нито псевдо-случайното генериране, нито някой от известните евристични методи е способен да получи решения със същото, или поне близко, качество като на $S_{inv}$, то някакъв нов метод, който се базира на съвършено различен подход, бе необходим. Така, възникна идеята за евристичен метод, базиран на имунен алгоритъм.

6. Идеята зад втория метод за S-box генериране се базира на модификация на имунен алгоритъм, работеща в стандартната посока отдолу-нагоре. Алгоритъмът стартира от някакъв случаен начален S-box, итеративно го изменя и търси S-boxes, които са по-добри от него по отношение на основните криптографски критерии. В резултат

от извършените множество изпитания в случая $n = 8$ се получиха големи масиви от биективни S-boxes, чиито свойства, макар и все още далеч от тези на $S_{inv}$, са по-добри от съответните свойства на останалите евристични методи. Също така, всички решения имат абсолютно случайна алгебрична структура и в болшинството от случаите нулева *линейна остатъчност*.

7. Вторият метод за генерация доказа своята ефективност за генериране на $(8 \times 8)$ биективни S-boxes. Заради случайния вход и големината на пространството, което се претърсва, методът не е приложим за размерности, по-големи от $(8 \times 8)$. Въпреки че произвежда по-добри S-boxes в сравнение с всички останали методи, базирани на стандартния подход отдолу-нагоре, той е по-неуспешен от *реверсивния генетичен алгоритъм* по отношение на криптографските свойства *нелинейност*, *differential uniformity* и *автокорелация*, и обикновено много по-успешен от него що се отнася до *линейната остатъчност* и комплексността на S-box алгебричната структура

8. Предвид подобните криптографски свойства на биективните S-boxes, получени с новите методи за генериране, много криптографски приложения, които се базират на блокови шифри, могат да бъдат захранени с различни криптографски силни S-boxes. От друга страна, в приложения, базирани на един и същи блоков шифър, получените криптографски еквивалентни по отношение на тяхната сила S-boxes могат да се използват за разграничаване на различни криптографски мрежи.

В случая $n = 8$, трите представени варианта на *reverse genetic algorithm* произвеждат хиляди биективни $(8 \times 8)$ S-boxes с *нелинейност* 104, 106, 108, 110 и $112 = N_{inv}$ за 2 дни работа на клъстер с 32 ядра. С изключение на метода в [28], който успя да произведе един S-box с $(N_S,\ \delta,\ deg(S),\ AC(S)_{max}) = (112, 6, 7, 32)$, притежаващ някаква *линейна остатъчност*, и ограничен брой от 92 неекивалентни $(8 \times 8)$ S-boxes с $(110, 6, 6$ и $7, 40)$, няма други налични в литературата методи, които да произвеждат голям брой S-boxes с *нелинейност*, по-голяма от 106. Що се отнася до *special immune algorithm*, за 10 дни работа на същия клъстер той успя да произведе 35 000 $(8 \times 8)$ S-boxes с $(104, 6, 6$ и $7, 88)$ и с нулева *линейна остатъчност*. В сравнение с резултатите, получени от първия метод, или със $S_{inv}$, тези S-boxes изглеждат да са с доста по-лоши криптографски свойства. Но, като се отчете, че първите или са афинно еквивалентни на $S_{inv}$, или до някаква степен имат тяхната алгебрична структура, то тези S-boxes стават по-привлекателни заради случайния си произход и очакваната по-добра устойчивост към алгебричните атаки [7].

В следващата таблица са сравнени най-добрите резултати на двойката нови методи за генерация с най-добрите $(8 \times 8)$ S-boxes, получени от известните в литаратурата методи.

| **Генериращи методи/свойства** | $N_S$ | $deg(S)$ | $AC(S)_{max}$ | $\delta$ |
|---|---|---|---|---|
| Pseudo-random Generation [17, 18] | 98 | - | - | - |
| 4-uniform Permut. Method [23, 24] | 98 | - | - | 4 |
| Tweaking Method [9] | 106 | 7 | 56 | 6 |
| Tweaking-based Method [28] | 110 | 7 | 40 | 4 |
| Tweaking-based Method [28] | 112 | 7 | 32 | 6 |
| Finite Field Inversion [22] | 112 | 7 | 32 | 4 |
| Hill climbing method [17] | 100 | - | - | - |
| Genetic Algorithm/Hill Climbing [18] | 100 | - | - | - |
| Simulated Annealing Method [6] | 102 | - | 80 | - |
| Special Genetic Algorithm [26] | 104 | - | - | - |
| Gradient Descent Method [14] | 104 | 7 | 80 | 8 |

В случая на $(16 \times 16)$ биективни S-boxes, приблизително за 3 месеца работа на същия клъстер и трите представени варианта на *reverse genetic algorithm* успяха да произведат големи множества със S-boxes, притежаващи криптографски свойства, които са много близо до тези на $S_{inv}$. Нещо повече, за прагове за *нелинейност* $N_{thr}$, които са избрани да са близки до $N_{S_{inv}}$, необходимото време за работа на всеки от вариантите се редуцира значително поради по-малкия брой извършени итерации. За такива S-boxes не е известно да са били получавани нито чрез псевдо-случайно генериране, нито с някои евристични подходи, така че за първи път се получават толкова много и толкова големи S-boxes, които да притежават добри криптографски свойства. Те могат да бъдат използвани при създаването на нови блокови шифри. По отношение на имплементацията и свързаните с нея ограничения, като необходима памет и скорост, очевидно тези S-boxes отстъпват пред $S_{inv}$. Те са много по-бавни и трябва да се съхраняват като големи *look-up tables* в *FPGA* матрици. Все пак, що се отнася до криптографската устойчивост на шифъра, тези S-boxes имат много по-сложна алгебрична структура и нямат линейна остатъчност. Следователно, са по-устойчиви към алгебричните атаки [7]. Заради големия си размер, те са много по-устойчиви също така и срещу линеен, диференциален, както и side-channel криптоанализ [2, 3, 15, 16].

# Обзор на дисертацията

В резултат от извършените изследвания и множеството проведени експерименти, съдържанието на настоящата дисертация е структурирано в 7 глави.

**Първата глава** се състои от увод, който е обособен в четири части: мотивация,

цели, резултати и структура на дисертацията.

Във **Втора глава** е представена онази част от теорията на булевите функции и на векторните булеви функции (наречени заместителни таблици или S-boxes), която има пряко отношение към криптографията и по-специално към криптографските алгоритми, използвани за осигуряване криптографската защита на информацията. Дефинирани са всички основни представяния и характеристики на заместителните таблици както и на булевите функции, които ги изграждат. Описани са различните типове S-boxes според тяхната размерност и свойства. Представени са най-известните атаки от криптоанализа, прилагани с цел преодоляване на криптографската устойчивост и разбиване на дадения алгоритъм, както и основните криптографски свойства, които трябва да притежават използваните в него S-boxes, за да се неутрализират тези атаки. Дадени са и някои специални булеви функции и S-boxes, притежаващи определени комбинации от полезни свойства. Накрая са показани взаимовръзките между криптографските критерии, както и търсените комбинации от критерии, които трябва да удовлетворява всеки добър S-box.

В **Трета глава** са представени известните в литературата методи за генериране на (биективни) S-boxes, разделени в зависимиост от използвания подход и обособени в три основни класа: псевдо-случайно генериране, алгебрични конструкции и евристични техники. Направен е сравнителен анализ на трите типа техники и способността им да произвеждат добри решения. Показано е превъзходството на алгебричните конструкции по отношение на качеството на получените решения и удовлетворяването в съвкупност на необходимите криптографски критерии, като за сметка на това са споменати техният малък брой, еднаква алгебрична структура и еквивалентност, което за в бъдеще би се превърнало в проблем при очакваното развитие на алгебричния криптоанализ. Що се отнася до останалите две техники за генериране, обърнато е внимание на обречеността на случайното търсене и псевдо-случайните S-boxes за размерности, по-големи от $(4 \times 4)$, докато евристичният подход е упоменат в контекста на надеждните и успешни методи за генериране на голям брой нееквивалентни S-boxes, притежаващи качества, които са близки до тези на алгебричните конструкции, но с по-богата и разнородна алгебрична структура. Накарая, описани са основните евристични техники: hill climbing, simulated annealing, генетични и имунни алгоритми, както и различни техни комбинации.

**Четвърта глава** съдържа основната част на настоящата дисертация, свързана с двойка нови алгоритми за генериране на биективни S-boxes и техните варианти, които се базират на комбинации от известни евристични техники и едновременно с това се базират на два коренно различни подхода - реверсивният подход, където алгоритъмът

започва своята работа от определена междинна позиция и се придвижва отгоре-надолу, и стандартният подход отдолу-нагоре, при който се тръгва от някоя случайна позиция, за да се достигне друга, съответстваща на по-добри решения. Представени са описания на ***Реверсивен генетичен*** и ***Специализиран имунен алгоритъм***.

***Реверсивният генетичен алгоритъм*** като всеки друг генетичен алгоритъм работи с начална популация от възможни решения (S-boxes), която итеративно се преобразува в нови популации от по-добри решения по отношение на едно или няколко свойства чрез прилагане на три основни операции, инспирирани от биологичната еволюция - селекция, кръстосване и мутация. Родителите от всяка една от двойките от селектираната начална популация се кръстосват, в резултат на което от всяка двойка се раждат по две деца. Кръстосването се реализира чрез някаква функция $breeding(\cdot)$. По време на процеса на кръстосване се смесват гените на родителите с цел постигане на по-богато биологично разнообразие на потомството. Гените на всяко дете се подлагат и на мутация, която се реализира чрез някаква мутираща функция $mutation(\cdot)$, имаща за цел допълнително добавяне на случайност. От математическа гледна точка това допълнително вкарване на случайност в оптимизационния процес допринася да се избегне попадане в локален екстремум. Всички деца от потомството се подлагат на селекция, за да се определят най-добрите сред тях, които ще формират следващата родителска популация, и за да се отхвърлят останалите. Селектирането се реализира на база някаква оценъчна функция - фитнес функция $fitness(\cdot)$, която измерва степента на покритие на желаното свойство, или някоя ценова функция $cost(\cdot)$, която изчислява цената на съответното дете. След приключване на селекцията, всички деца, преминали успешно фитнес теста, заменят родителите си в родителската популация, с което целият цикъл се затваря и процесът преминава към следваща итерация, започвайки нов еволюционен цикъл към следващите поколения. За разлика от биологичната еволюция, представляваща непрекъснат процес, използваните за решаване на оптимизационни задачи различни генетични алгоритми спират своята работа след приключване на даден период от време или при достигане на някакъв предварително зададен праг за брой итерации.

Противно на конвенционалния генетичен алгоритъм, представеният в настоящата дисертация ***реверсивен генетичен алгоритъм (РГА)*** работи в обратен ред. Той стартира от начална популация, формирана от най-добрите намерени досега биективни $(n \times n)$ S-boxes, $S_{inv}$, които се базират на обратна функция в крайното поле $GF(2^n)$ и които притежават оптималната до момента комбинация от основни криптографски свойства. Идеята му е да успее да получи или популация, в която поне един S-box да е по-добър от $S_{inv}$ по отношение на кое да е от свойствата нелинейност (nonlinearity), differential uniformity или автокорелация (autocorrelation), което ще бъде равносилно с

успешно решаване на отворен криптографски проблем, или много на брой популации с различни нееквивалентни $(n \times n)$ биективни S-boxes, които са съвсем леко по-лоши по отношение на горните свойства, но за сметка на това са голям брой, по-разнородни са и са с далеч по-сложна алгебрична структура. Алгоритъмът е в четири модификации, които се базират на различни комбинации от една ценова и три фитнес функции, използвани за оценка на свойствата на децата по време на селекцията. Предназначението му е да получи добри биективни S-boxes от всички възможни размерности от $(6 \times 6)$ до $(16 \times 16)$. Колко точно да са добри и колко близки до $S_{inv}$ да са тези S-boxes се определя от зададените предварително прагове за нелинейност $N_{thr}$ и за differential uniformity $\delta_{thr}$. Тези прагове могат да служат и за естествени стопове на алгоритъма, при достигането на които от страна на всички деца от потомството той спира. В този случай, всички получени решения притежават съответните свойства, но техният брой е точно толкова голям, колкото е големината на популацията. За получаване на по-голяма възможност за избор и по-добра разнородност на решенията са необходими големи обеми от еднакви по отношение на криптографските свойства S-boxes. Това се постига като алгоритъмът продължи своята работа и след достигане на зададените прагове. Тогава, решението за стоп на алгоритъма се взима на база брой извършени итерации, изтекъл времеви период или достигнат минимален брой от необходими решения. Още една съществена разлика, отличаваща първия вариант на алгоритъма (***РГА1***) от останалите три, е в процедурата по кръстосване на родителските S-boxes и начина, по който се копират гените на децата от гените на родителите. Това се осъществява чрез промяна на един от параметрите в $breeding(\cdot)$ функцията, определящ реда на копиране на гените от съответния родител. И в четирите модификации на алгоритъма ролята на мутираща функция се играе от функцията $modeling(\cdot)$, чиято основна цел, освен вкарването на повече случайност в процеса и обогатяване на разнообразието, е също и да възстанови в детето биективното свойство, което евентуално се е загубило при копирането на едни и същи гени от двамата родители и последващото им дублиране в него.

Съответните за всеки от различните варианти на алгоритъма функции както и всички функции, които са еднакви във всеки вариант, имат следния вид:

- Кръстосваща функция $breeding(\cdot)$

  От родителската двойка $(P_i, P_j)$, една от всички възможни $\frac{T(T-1)}{2}$ двойки, където $T$ е големината на популацията, а $P_i$ и $P_j$ са биективни $(n \times n)$ S-boxes в началото от тип $S_{inv}$, се раждат децата $Ch_1$ и $Ch_2$: $(Ch_1, Ch_2) = breeding(P_i, P_j, CoP_1, CoP_2, cnt)$. Точките $CoP_1$ и $CoP_2$ са случайни за всяко кръстосване и указват двете позиции,

където се разделят генетичните вериги съответно на първия и втория родител. Числото $cnt$ е случайно и приема стойностите от 1 до 5, определящи реда, по който се копират гените - прав или обратен.

$cnt = 1$:
$Ch_1(x_k) = P_i(x_k)$, $k = 1, 2, \ldots, CoP_1$
$Ch_1(x_k) = P_j(x_k)$, $k = CoP_1 + 1, CoP_1 + 2, \ldots, 2^n$
$Ch_2(x_p) = P_j(x_p)$, $p = 1, 2, \ldots, CoP_2$
$Ch_2(x_p) = P_i(x_p)$, $p = CoP_2 + 1, CoP_2 + 2, \ldots, 2^n$

$cnt = 2$:
$Ch_1(x_{CoP_1+1-k}) = P_i(x_k)$, $k = 1, 2, \ldots, CoP_1$
$Ch_1(x_k) = P_j(x_k)$, $k = CoP_1 + 1, CoP_1 + 2, \ldots, 2^n$
$Ch_2(x_p) = P_j(x_p)$, $p = 1, 2, \ldots, CoP_2$
$Ch_2(x_p) = P_i(x_p)$, $p = CoP_2 + 1, CoP_2 + 2, \ldots, 2^n$

$cnt = 3$:
$Ch_1(x_k) = P_i(x_k)$, $k = 1, 2, \ldots, CoP_1$
$Ch_1(x_{2^n+CoP_1+1-k}) = P_j(x_k)$, $k = CoP_1 + 1, CoP_1 + 2, \ldots, 2^n$
$Ch_2(x_p) = P_j(x_p)$, $p = 1, 2, \ldots, CoP_2$
$Ch_2(x_p) = P_i(x_p)$, $p = CoP_2 + 1, CoP_2 + 2, \ldots, 2^n$

$cnt = 4$:
$Ch_1(x_k) = P_i(x_k)$, $k = 1, 2, \ldots, CoP_1$
$Ch_1(x_k) = P_j(x_k)$, $k = CoP_1 + 1, CoP_1 + 2, \ldots, 2^n$
$Ch_2(x_{CoP_2+1-p}) = P_j(x_p)$, $p = 1, 2, \ldots, CoP_2$
$Ch_2(x_p) = P_i(x_p)$, $p = CoP_2 + 1, CoP_2 + 2, \ldots, 2^n$

$cnt = 5$:
$Ch_1(x_k) = P_i(x_k)$, $k = 1, 2, \ldots, CoP_1$
$Ch_1(x_k) = P_j(x_k)$, $k = CoP_1 + 1, CoP_1 + 2, \ldots, 2^n$
$Ch_2(x_p) = P_j(x_p)$, $p = 1, 2, \ldots, CoP_2$
$Ch_2(x_{2^n+CoP_2+1-p}) = P_i(x_p)$, $p = CoP_2 + 1, CoP_2 + 2, \ldots, 2^n$

- Мутираща функция $modeling(\cdot)$

Всяко дете $Ch$, което е получено в резултат от кръстосването, може и да не е биективно и затова се подава на мутиращата функция: $Ch = modeling(Ch)$. Новополученото от изхода на функцията дете $Ch$ вече е $(n \times n)$ биективен S-box.

Например, ако $(Ch_1, Ch_2) = breeding(P_i, P_j, CoP_1, CoP_2, cnt)$ и децата са от вида:

$Ch_1 = [a_1, a_2, \ldots, a_{CoP_1-1}, a_{CoP_1}, a_{CoP_1+1}, \ldots, a_{2^n-1}, a_{2^n}]$
$Ch_2 = [b_1, b_2, \ldots, b_{CoP_2-1}, b_{CoP_2}, b_{CoP_2+1}, \ldots, b_{2^n-1}, b_{2^n}]$,

то всяка от редиците с гени $\{a_1, a_2, \ldots, a_{CoP_1}\}$ и $\{a_{CoP_1+1}, a_{CoP_1+2}, \ldots, a_{2^n}\}$ на $Ch_1$, както и всяка от редиците $\{b_1, b_2, \ldots, b_{CoP_2}\}$ и $\{b_{CoP_2+1}, b_{CoP_2+2}, \ldots, b_{2^n}\}$ на $Ch_2$ ще се състоят от различни елементи заради биективността на родителите си $P_i$ и $P_j$. Тъй като $\{a_1, a_2, \ldots, a_{CoP_1}\}$ е част от $P_i$ и $\{a_{CoP_1+1}, a_{CoP_1+2}, \ldots, a_{2^n}\}$ е част от $P_j$, то някои от елементите в редицата $\{a_1, a_2, \ldots, a_{CoP_1}\}$ може да се дублират в редицата $\{a_{CoP_1+1}, a_{CoP_1+2}, \ldots, a_{2^n}\}$. Биективността на всяко от децата се възстановява чрез функцията $modeling(\cdot)$ по следния начин:

Последователно, един по един, се проверяват елементите $\{a_{CoP_1+1}, a_{CoP_1+2}, \ldots, a_{2^n}\}$ за това, дали съответният елемент вече се срещнал в $\{a_1, a_2, \ldots, a_{CoP_1}\}$. Ако се срещне повтарящ се елемент, то последователно и по случаен начин се генерират нови елементи, докато не се срещне такъв, който още не се е срещнал. Той подменя засегения дубликат и се преминава към следващия отдясно елемент в редицата. Накрая, съответното дете вече е пермутация.

- Фитнес функции $fitness\_N(\cdot)$, $fitness\_\delta(\cdot)$ и $fitness\_component\_N(\cdot)$

  Всяко дете $Ch$, което вече е мутирало и е биективен $(n \times n)$ S-box, се подлага на фитнес тест. В зависимост от съответната модификация на алгоритъма, в теста се използва самостоятелно или в комбинация някоя от функциите $fitness\_N(\cdot)$, $fitness\_\delta(\cdot)$ и $fitness\_component\_N(\cdot)$. Съответните функции имат вида:

  $fitness\_N(Ch) = N_{Ch}$, където $N_{Ch}$ е нелинейността на детето $Ch$

  $fitness\_\delta(Ch) = \delta_{Ch}$, където $\delta_{Ch}$ е differential uniformity на детето $Ch$

  $fitness\_component\_N(f) = num$, където $num$ е точно броят на компонентните

булеви функции на $Ch$, които са с по-голяма нелинейност от $N_{inv} = 2^{n-1} - 2^{\frac{n}{2}}$

Понятията нелинейност, differential uniformity, компонентни булеви функции, както и всички останали понятия, свързани с тях, се дефинират както следва:

Ако всяко едно от децата (биективен $(n \times n)$ S-box) се разгледа като векторна булева функция $Ch = (f_1, f_2, \ldots, f_n) : B^n \to B^n$, където $f_i : B^n \to B$ за $i = 1, 2, \ldots, n$, са координатните булеви функции на $Ch$, то всички линейни комбинации без нулевата на координатните функции, $< v, Ch >= v_1 f_1(x) \oplus \ldots \oplus v_n f_n(x)$, за $v \in B^n \setminus \{0\}$, са булеви функции, наречени компонентни булеви функции на $Ch$.

Под нелинейност на булевата функция $f : B^n \to B$ се разбира неотрицателното число $N_f$, което се дефинира като $N_f = 2^{n-1} - \frac{1}{2} \max_{(w \in B^n)} |\widehat{F}_f(w)|$, където $\widehat{F}_f(w)$ е трансформацията на Уолш-Адамар на булевата функция $f(x)$. Нелинейността е ограничена отгоре от $2^{n-1} - 2^{\lfloor \frac{n}{2} \rfloor - 1}$, като равенството се достига от bent функциите.

Ако $f : B^n \to B$ е дадена булева функция, то трансформацията на Уолш-Адамар $\widehat{F}_f : B^n \to [-2^n, 2^n] \subset Z$ се дефинира като $\widehat{F}_f(w) = \sum_{x \in B^n} (-1)^{f(x) \oplus < w, x >}$, $\forall w \in B^n$

Ако детето $Ch$ е биективен $(n \times n)$ S-box, то под нелинейност на $Ch$ се разбира неотрицателното число $N_{Ch}$, дефинирано с:

$$\begin{aligned} N_{Ch} &= min_{(v \in B^n \setminus \{0\})}\ N_{v_1 f_1(x) \oplus v_2 f_2(x) \oplus \ldots \oplus v_n f_n(x)}, \text{ или} \\ N_{Ch} &= 2^{n-1} - \frac{1}{2} max_{(v \in B^n \setminus \{0\})}\ max_{(w \in B^n)} |\ \widehat{F}_{v_1 f_1(x) \oplus \ldots \oplus v_n f_n(x)}(w)| \end{aligned}$$

Ако детето $Ch$ е биективен $(n \times n)$ S-box, то под differential uniformity ($\delta$-uniformity) на $Ch$ се разбира числото $\delta$, дефинирано с: $\delta = max_{(\alpha \in B^n \setminus \{0\})}\ max_{(\beta \in B^n)}\ \delta(\alpha, \beta)$, където $\delta(\alpha, \beta) = |\{x \in B^n | Ch(x) \oplus Ch(x \oplus \alpha) = \beta\}|$.

- Ценова функция $cost(\cdot)$

Мутиралото дете $Ch$, което е биективен $(n \times n)$ S-box и което е подложено на фитнес тест, в зависимост от варианта се предава и на ценовата функция $cost(\cdot)$, изчисляваща цената му $cost(Ch) = C_{Ch}$ по следния начин [26]:

$$C_{Ch} = \sum_{c=(c_1, c_2, \ldots, c_n) \in B^n \setminus \{0\}} \sum_{\omega \in B^n} |\widehat{F}_{c_1 f_1 \oplus c_2 f_2 \oplus \ldots \oplus c_n f_n}(\omega) - 21|^7$$

- Фитнес тест

  В зависимост от съответната модификация на алгоритъма, всяко от децата $Ch$, което е подложено на фитнес тест, успешно ще премине теста и ще премине към следващата итерация на алгоритъма, ако са изпълнени следните условия:

  За ***РГА1***: $N_{Ch} \geq N_{thr}$

  За ***РГА2***: $N_{Ch} \geq N_{thr}$ и $C_{Ch} < C_P$

  За ***РГА3***: $N_{Ch} \geq N_{thr}$, $\delta_{Ch} \leq \delta_{thr}$ и $C_{Ch} < C_P$

  За ***РГА3М***: $N_{Ch} \geq N_{thr}$, $\delta_{Ch} \leq \delta_{thr}$, $C_{Ch} < C_P$ и $num_{Ch} > 0$

**Коментари**.

Представеният нов евристичен метод - ***реверсивен генетичен алгоритъм (РГА)*** е съвместна разработка с д-р Николай Николов и доц. д-р Светла Никова. Първи вариант ***РГА1*** е докладван на международна конференция BalkanCrypt 2013, проведена в град София. ***РГА1*** и ***РГА2*** са докладвани на National Coding Theory Seminar 2013, който се проведе в град Велико Търново. Всичките три варианта на новия метод за генериране ***РГА*** са публикувани в Cryptology ePrint Archive 801/2014 и JOURNAL of Cryptography and Communications 8(2), 247-276.

***Специализираният имунен алгоритъм*** представлява клас методи за решаване на различни оптимизационни задачи, известен като *Изкуствени имунни системи*, които от своя страна са инспирирани от биологичната имунна система. Основната задача на биологичната имунна система е свързана с предпазването на защитавания организъм от проникване в него на злонамерени микроорганизми (вируси, бактерии, паразити и др.), известни като патогени. Имунната система съдържа няколко защитни слоя, отговарящи за прихващането и елиминирането на проникнали патогени. Най-общо, тя може да бъде разглеждана като съставена от две взаимосвързани подсистеми - вродена и придобита имунна система. Докато първата е резултат от еволюцията на съответния биологичен вид и се наследява по рождение от предишните поколения, то придобитата имунна система е адаптивна и отговаря за изграждане на защитата на организъма към точноопределен патоген. Тя поддържа памет за вече състояли се инфектирания с дадени патогени, което

й помага бързо да се справи при последваща тяхна поява и реинфекция. Най-важните клетки на придобитата имунна система, непосредствено отговорни за идентифицирането и унищожаването на патогените, са така наречените бели кръвни телца, известни още като левкоцити или лимфоцити. При проникване на вече познат патоген отговорът е много бърз. От паметта се извлича видът на левкоцита, който е най-успешен срещу този патоген, и за много кратко време съответната клетка се клонира достатъчно голям брой пъти, докато се натрупа необходимата ударна мощ, която да се хвърли срещу патогена. В случай че проникналият патоген е нов и не е бил още засичан, отговорът на системата е значително по-бавен. Тя хвърля срещу него различни левкоцити, за да се определи най-успешните сред тях. От подбраните няколко вида клетки посредством клонирането им се образуват съответният брой армии, които атакуват патогена. По време на процеса на клониране, поради малки грешки при копиране, някои от клетките мутират и леко се видоизменят. Това спомага за подобряване на разнообразието от клетки и получаване на нови левкоцити, които евентуално са по-силни и по-успешни срещу патогена. Отново се селектират най-добрите, клонират се и се пускат в настъпление. Процесът се зацикля до момента на победата и пълното унищожение на патогена, когато в паметта се записва видът на левкоцита, който е победил.

Към класа на *изкуствените имунни системи* спадат няколко често използвани за оптимизационни задачи алгоритми, като *Клонална селекция*, *Негативна селекция* и *Имунни мрежи* [5]. От гледна точка на криптографията, а в частност и на теорията на булевите и векторните булеви функции, до момента в литературата не са известни резултати, свързани с генериране на S-boxes чрез имунни алгоритми. Представеният в настоящата дисертация ***специализиран имунен алгоритъм*** за генериране на $(n \times n)$ биективни S-boxes се базира на лека модификация на алгоритъма за *клонална селекция*. Алгоритъмът за *клонална селекция* стартира работата си с някакъв начален репертоар (популация) от левкоцити (S-boxes) и селектира тези сред тях, които са най-подходящи да се изправят срещу проникналия патоген. Селекцията се базира на някаква оценъчна функция - фитнес функция $fitness(\cdot)$ или ценова функция $cost(\cdot)$. След като се изберат най-подходящите леквоцити, започва клонирането им посредством някаква клониращa функция $clone(\cdot)$, докато се получат хиляди техни клонинги. По време на клонирането клетките леко се изменят, заради възникнали малки грешки при копиране. Процесът се нарича *соматична хипермутация* и се реализира посредством някаква мутираща функция $mutation(\cdot)$, чиято основна цел е увеличиване на разнообразието. В резултат от клонирането и мутацията, боеспособността и адаптивността на организма чувствително са се подобрили чрез дублиране и вариативност. Впоследствие, формираната популация от клонинги отново се селектира, за да се подберат най-добрите клетки. Те започват да се клонират и т.н... Цикълът итеративно се завърта, до момента когато се стигне до

крайната популация с клонинги, определена чрез някакъв критерий за стоп - праг, брой итерации или време. Най-добрите левкоцити от последната популация са най-добрите търсени решения (S-boxes).

За разлика от представения по-горе ***реверсивен генетичен алгоритъм (РГА)***, ***специализираният имунен алгоритъм (СИА)*** разчита на стандартния подход отдолу-нагоре. Също така, противно на ***РГА*** и на алгоритъма за *клонална селекция*, вместо от цяла начална популация, той стартира своята работа само от една единствена начална имунна клетка (биективен $(n \times n)$ S-box). Алгоритъмът е подбрана специална комбинация от модифициран *hill climbing* метод за S-boxes (*MHCM*) [17] и модифициран алгоритъм за *клонална селекция (MCSA)* [5]. Началното решение $S_0$ се подава на *MHCM*, който на всяка стъпка разменя местатата на два негови елемента и изчислява цената на новополучения S-box $S$ чрез ценовата функция $cost(\cdot)$. Ако $cost(S) < cost(S_0)$, процесът итеративно продължава от $S$. Иначе, нов S-box $S_0$ се генерира и се започва отначало. Накрая, в резултат от *MHCM*, е получен S-box $S$, който е с минимална цена. Той се подава на *MCSA*. Първоначално $S$ се клонира чрез $clone(\cdot)$ и се създават три негови идентични клонинга. Получените четири идентични S-boxes $S$ се разделят по двойки и всяка двойка се подлага на различна мутация, реализирана посредством две мутиращи функции $mutate_1(\cdot)$ и $mutate_2(\cdot)$. Получават се нови четири различни S-boxes $S_1$, $S_2$, $S_3$ и $S_4$. Всеки от тях се оптимизира по отношение на цената си посредством *MHCM*, докато не се получат четири нови S-boxes $S'_1$, $S'_2$, $S'_3$ и $S'_4$, които са с минимлна цена. Селектира се от тях само един S-box - този, $S'$, който е с минимална цена. Ако СТОП критерият е удовлетворен, алгоритъмът спира и $S'$ е търсеното оптимално решение. В противен случай, $S'$ подменя $S$ на входа на *MCSA* и стартира нова итерация на алгоритъма, с което цикълът е затворен.

Конкретното описание на използваните в алгоритъма функции е следното:

- Ценова функция $cost(\cdot)$

  Ценовата функция, която пресмята цената на съответния левкоцит $S$, се дефинира по следния начин:

  $$cost(S) = cost_1(S).cost_2(S).cost_3(S),$$

  където всяка от трите ценови функции $cost_1(\cdot)$, $cost_2(\cdot)$ и $cost_3(\cdot)$ има следния вид:

- Ценова функция $cost_1(\cdot)$

Ценовата функция $cost_1(\cdot)$, пресмятаща цената на $S$, се дефинира с:

$$cost_1(S) = \sum_{c<d \in B^n \setminus \{0\}} \sum_{\omega \in B^n} \mid |\widehat{F}_{c_1 f_1 \oplus c_2 f_2 \oplus \ldots \oplus c_n f_n}(\omega)|^3 - |\widehat{F}_{d_1 f_1 \oplus d_2 f_2 \oplus \ldots \oplus d_n f_n}(\omega)|^3 \mid^7$$

- Ценова функция $cost_2(\cdot)$

  Ценовата функция $cost_2(\cdot)$, пресмятаща цената на $S$, се дефинира с:

  $$cost_2(S) = \sum_{c=(c_1,c_2,\ldots,c_n) \in B^n \setminus \{0\}} \sum_{\omega \in B^n} |\widehat{F}_{c_1 f_1 \oplus c_2 f_2 \oplus \ldots \oplus c_n f_n}(\omega) - 21|^7,$$

  която е взаимствана от [26].

- Ценова функция $cost_3(\cdot)$

  Ценовата функция $cost_3(\cdot)$, пресмятаща цената на $S$, се дефинира с:

  $$cost_3(S) = \sum_{\delta_{00} \neq \delta_{\alpha\beta}} (\delta_{\alpha\beta} - 1)^2 . (\delta_{\alpha\beta} - 2)^2 . (\delta_{\alpha\beta} - 4)^2,$$

  където $\delta_{\alpha\beta} = \delta(\alpha, \beta) = |\{x \in B^n | S(x) \oplus S(x \oplus \alpha) = \beta\}|, \forall \alpha \in B^n, \forall \beta \in B^n$

- Мутираща функция $mutate_1(\cdot)$

  Мутиращата функция $mutate_1(\cdot)$, изменяща $S$, се дефинира с:

  $$S' = mutate_1(S),$$

  където $S'$ се получава от $S$ чрез разменяне на два съседни елемента. Позициите $p-1$ и $p$ на тези елементи зависят от числото $p$ в интервала $[2, 2^n]$, което се избира по случаен начин при всяко изпълнение на функцията. Това гарантира, че когато и да се изпълни функцията $mutate_1(\cdot)$, полученият S-box ще е различен от този, получен при предишното изпълнение.

- Мутираща функция $mutate_2(\cdot)$

  Мутиращата функция $mutate_2(\cdot)$, изменяща $S$, се дефинира с:

  $$S'' = mutate_2(S),$$

където $S''$ се получава от $S$ по следния начин: изменя се блок от съседни елементи в $S$ със случайно избрана дължина $q \in [2, 8]$. Ако $q$ е четно число, то елементите се разменят симетрично спрямо позицията $p$, която е случайно избрана и която разделя блока на две части с еднаква дължина. В противен случай, ако $q$ е нечетно число, то елементът по средата на позиция $p$ (случайно избрана) остава на място, а останалите, симетрични спрямо него елементи от блока, два по два си разменят местата. Случайният избор на числата $q$ и $p$ при всяко изпълнение на функцията гарантира, че при всяко последващо нейно изпълнение винаги ще се получава нов S-box, различен от предишния.

Алгоритъмът има модификация, която е предназначена конкретно за случая $(6 \times 6)$ и която търси поне една пермутация $S$, която да е *почти перфектно нелинейна* ($APN \iff \delta = 2$) и която да не е афинно еквивалентна на единствената такава открита досега в [4]. Промяната се състои единствено в ценовата функция $cost(\cdot)$, която вместо произведение на три различни ценови функции, е модификация на функцията $cost_3(\cdot)$:

$$cost(S) = \sum_{\delta_{00} \neq \delta_{\alpha\beta}} (\delta_{\alpha\beta} - 1)^2,$$

където $\delta_{\alpha\beta} = \delta(\alpha, \beta) = |\{x \in B^n | S(x) \oplus S(x \oplus \alpha) = \beta\}|, \forall \alpha \in B^n, \forall \beta \in B^n$

**Коментари**

***Специализираният имунен алгоритъм (СИА)*** също е разработен съвместно с д-р Николай Николов и доц. д-р Светла Никова. Докладван е на втора международна конференция по криптология BalkanCryptSec 2015, проведена в град Копер, Словения.

В **Пета глава** са представени всички резултати, получени от двата нови метода за генерация по време на множеството проведени изпитания за всички четни размерности от $(6 \times 6)$ до $(16 \times 16)$. Разделена е на две части - по една за ***РГА*** и ***СИА***. В първата част е направена сравнителна характеристика на получените от всеки от вариантите на ***РГА*** биективни S-boxes в случаите $n = 8, 12$ и 16 със съответните $S_{inv}$, базирани на обратна функция в крайното поле $GF(2^n)$. Във втората част са представени резултатите на ***СИА*** за $n = 8$ и са сравнени с всички останали методи за генериране на биективни $(8 \times 8)$ S-boxes. Също така, показани са и резултатите на модификацията на алгоритъма за случая $n = 6$, където се използва изменена ценова функция с цел получаване на $APN$ пермутация, която не е еквивалентна на тази от [4]. Не на последно място, сравнени са

резултатите на двата нови метода за генериране, като също така са сравнени помежду си и отделните техни варианти.

**Шеста глава** е разделена на три отделни части - по една съответно за разделите **Обобщение**, **Принос на дисертацията** и **Бъдеща работа**. Направено е резюме на всички проведени изследвания. Разгледани са поставените цели и е разяснена степента, до която те са изпълнени в съответствие с получените резултати. Накрая са разгледани някои възможни модификации, които биха могли да се изпробват за в бъдеще с цел евентуално получаване на още по-добри резултати.

В **Седма глава** е поместен **Appendix**. Разделен е на 5 основни части, във всяка от които са представени екземпляри от най-добрите решения, получени от всеки един от алгоритмите заедно с неговите варианти, в случая $n = 8$. За ***РГА*** са представени по 5 биективни $(8 \times 8)$ S-boxes, съответстващи на всеки от четирите варианта и на прагове за *нелинейност* $N_{thr}$: 104, 106, 108, 110 и 112. За ***СИА*** е показан един представител на най-добрите биективни $(8 \times 8)$ S-boxes, получен при случаен вход на алгоритъма.

# Апробация на резултатите

Първият резултат [20], който е включен в дисертацията, е в съавторство с Виолета Дъчева и с д-р Николай Николов. Всички останали резултати [10, 11, 12] са получени в съавторство с доц. д-р Светла Никова и д-р Николай Николов.

Публикувани са в следните международни научни списания или в proceedings на международни конференции:

- Proceedings of International Scientific Conference MATTEX 2014, Shumen 2014 [20]
- IACR Cryptology ePrint Archive (2014) [10]
- International Conference on Cryptography and Information Security Balkan Crypt Sec 2015 [11]
- Journal Cryptography and Communications. Descrete Structures, Boolean Functions and Sequences (2016) [12]

Докладвани са на:

- International Conference on Cryptography and Information Security BalkanCrypt, Sofia, Bulgaria, 2013 [10]

- National Coding Theory Seminar, Veliko Turnovo, Bulgaria, 2013 [10]
- International Conference on Cryptography and Information Security BalkanCryptSec, Koper, Slovenia, 2015 [11]

# Авторска справка

По мнение на автора, основните приноси на дисертационния труд са:

- Представяне на проблема за генериране на голям брой криптографски добри $(n \times n)$ биективни S-boxes за $n \in [6, 16]$ чрез метода на псевдо-случайното генериране или чрез алгебрични конструкции:

  (1) Направено е сравнение между всички известни типове методи за генериране и възможностите им за получаване на голям брой неекивалентни биективни S-boxes, за всяка размерност от $(6 \times 6)$ до $(16 \times 16)$, които са криптографски добри.

- Представяне на проблема за генериране на биективни $(n \times n)$ S-boxes, притежаващи близки криптографски свойства до тези на $S_{inv}$:

  (2) Разработени са първите два варианта на ***реверсивен генетичен алгоритъм***, генериращи за първи път такъв голям брой и толкова големи биективни S-boxes, които са с близки свойства до $S_{inv}$. Получените най-добри комбинации от свойства за $n = 8$ са: $(N,\ \delta,\ deg,\ AC_{max}) = (110, 6, 7, 40)$ и $(112 = N_{inv}, 6, 7, 32)$. При $n = 12$ и $n = 16$ с ***РГА1*** са получени $(n \times n)$ S-boxes с комбинации от свойства, съответно $(1984 = N_{inv}, 6, 11, 128)$ и $(32512 = N_{inv}, 6, 15, 512)$, каквито, изключвайки $S_{inv}$, досега не са докладвани в литературата.

- Представяне на проблема за генериране на биективни $(n \times n)$ S-boxes, които имат близки криптографски свойства до тези на $S_{inv}$ и нулева *линейна остатъчност*:

  (3) Разработен е трети вариант на ***реверсивен генетичен алгоритъм (РГА3)***, генериращ биективни $(n \times n)$ S-boxes, които за първи път са такъв голям брой, са толкова близки до $S_{inv}$ по отношение на криптографските си свойства и са с толкова по-сложна алгебрична структура от $S_{inv}$ (нулева *линейна остатъчност*). Най-добрите S-box комбинации от свойства, които са получени в случая $n = 8$, са: $(N,\ \delta,\ deg,\ AC_{max},\ LinRed) = (110, 4 = \delta_{inv}, 7, 40, zero)$ и $(112 = N_{inv}, 6, 7, 32, some)$.

- Представяне на проблема за генериране на биективни $(8 \times 8)$ S-boxes, близки до $S_{inv}$, когато се стартира от случайни S-boxes:

  (4) Разработен е първи вариант на ***специализиран имунен алгоритъм***, който за първи път генерира такъв голям брой $(8 \times 8)$ биективни S-boxes, които са със следните свойства: $(N,\ \delta,\ deg,\ AC_{max}) = (104, 6, 7, 88)$, при условие че се стартира от случаен S-box. *Нелинейност* 104 досега е достигната само от 2 метода - Special Genetic Algorithm [26] и Gradient Descent Method [14]. При първия метод изобщо не се споменава свойството *differential uniformity*, докато при втория - минималната получена $\delta$ е 8. Постигнатата от ***СИА1*** комбинация $(N,\ \delta) = (104, 6)$ се получава за първи път, стартирайки от случен $(8 \times 8)$ биективен S-box. Въпреки че е далеч все още от комбинацията на $S_{inv}$: $(N_{inv},\ \delta_{inv}) = (112, 4)$ или на S-boxes от някои други методи, базирани на алгебрични конструкции, получената комбинация успя още да скъси дистанцията. Нещо повече, достигната е от голям брой различни S-boxes, притежаващи нулева *линейна остатъчност*.

- Представяне на проблема за генериране на биективни $(6 \times 6)$ $APN$ пермутации чрез евристични методи, когато се стартира от $APN$ S-box:

  (5) Разработен е втори вариант на ***специализиран имунен алгоритъм (СИА2)***, генериращ голям брой $(6 \times 6)$ биективни S-boxes, които са с близки свойства до $S_{inv}$ и са $APN$. Независимо че всички до момента получени решения са еквивалентни на намереното в [4], до намиране на доказателство за несъществуване на други, нееквивалентни на единствената намерена, $(6 \times 6)$ $APN$ пермутации, изследването не е лишено от смисъл.

В крайна сметка, в случая когато се търсят биективни $(8 \times 8)$ S-boxes, са представени два нови и съвършено различни по отношение на своята същност евристични подхода, способни да произведат достатъчно голям брой S-boxes, чиито криптографски свойства са близки до тези на $S_{inv}$. Първият подход е значително по-бърз, тъй като разчита на генетичен алгоритъм, работещ в обратен ред, но за сметка на това получените решения не са със случаен произход и донякъде споделят същата алгебрична структура като на $S_{inv}$. От друга страна, вторият метод тръгва от абсолютно случаен S-box и достига до нов, значително по-добър от него, който притежава случайна алгебрична структура и нулева *линейна остатъчност*. Въпреки че дава по-добри резултати спрямо основните криптографски критерии от всички известни методи с изключение на някои алгебрични

конструкции, все пак той е по-слаб от $S_{inv}$. Що се отнася до по-големите размерности, например като $n = 16$, ***реверсивен генетичен алгоритъм (РГА1)*** е първият метод, който получава толкова големи биективни S-boxes, които едновременно с това са много на брой и с много близки криптографски свойства до свойствата на $S_{inv}$ и на останалите алгебрични конструкции.

# Благодарности

Бих искал да изразя своята голяма благодарност към научните си ръководители доц. д-р Светла Никова от КУ Льовен, Белгия, и доц. дмн. Емил Колев от ИМИ-БАН за актуалните задачи, проявеното разбиране и цялата оказана подкрепа през периода на докторантурата. Благодаря също на г-жа Виолета Дъчева и г-н Светослав Александров за това, че ме подтикнаха да стартирам тази докторантура, както и на проф. дмн. Цонка Байчева за проявения интерес към цялостната ми работата, помощта и за всички ценни съвети. Но най-вече благодаря на моя основен съавтор, колега и приятел д-р Николай Николов за това, че с негова помощ изградихме един чудесен взаимодопълващ се тандем между инженер и математик, и успяхме да изгенерираме и впоследствие на практика да реализираме някои чудесни идеи, свързани с евристични алгоритми и приложението им за намиране на добри S-boxes. Изказвам благодарност и на семейството си, на всички приятели и колеги за съдействието им, за тяхната отзивчивост, както и за това, че ме изтърпяха. Не на последно място, благодаря на всички мои критици - доброжелатели и недоброжелатели, за това че допълнително ме амбицираха и по този начин помогнаха да се превърна в един по-мотивиран и още по-вярващ в собствените си способности човек.

# Публикации по дисертацията

Основните резултати на изследванията, свързани с дисертацията, са представени в следните публикации:

1. [20] N. Nikolov, G. Ivanov, and V. Ducheva. *Generation of Substitution Tables with Good Cryptographic Properties using Artificial Immune Algorithms*. International Scientific Conference MATTEX 2014, Shumen, 2014.

2. [10] G. Ivanov, N. Nikolov, and S. Nikova. *Reversed Genetic Algorithms for generation of bijective S-boxes with good cryptographic properties*. Cryptology ePrint Archive, Report 2014/801 2014.

3. [12] G. Ivanov, N. Nikolov, and S. Nikova. *Reversed Genetic Algorithms for generation of bijective S-boxes with good cryptographic properties*. Journal of Cryptography and Communications. Descrete Structures, Boolean Functions and Sequences, 8(2), 247-276. DOI 10.1007/s12095-015-0170-5

4. [11] G. Ivanov, N. Nikolov, and S. Nikova. *Cryptographically strong S-boxes generated by Modified Immune Algorithm*. Springer International Publishing Switzerland 2016. E. Pasalic and L.R.Knudsen (Eds.): BalkanCryptSec 2015, LNCS 9540, pp.1-12, 2016. DOI 10.1007/978-3-319-29172-7-3

# Списък с цитирания

Съответният брой на цитиране на публикациите е както следва:

- [10] G. Ivanov, N. Nikolov, and S. Nikova. *Reversed Genetic Algorithms for generation of bijective S-boxes with good cryptographic properties*. Cryptology ePrint Archive, Report 2014/801 2014 $\longrightarrow$ 2 пъти

    1. С.В. Поликарпов, А.А. Кожевников. *ПСЕВДО-ДИНАМИЧЕСКИЕ ПОДСТАНОВКИ: ИССЛЕДОВАНИЕ ЛИНЕЙНЫХ СВОЙСТВ*. izv-tn.tti.sfedu.ru
    2. С.В. Поликарпов, К.Е. Румянцев, А.А. Кожевников. *Исследование линейных характеристик псевдодинамических подстановок*. Журнал Известия Южного федерального университета. cyberleninka.ru, 2015

- [12] G. Ivanov, N. Nikolov, and S. Nikova. *Reversed Genetic Algorithms for generation of bijective S-boxes with good cryptographic properties*. Journal of Cryptography and Communications. Descrete Structures, Boolean Functions and Sequences, 8(2), 247-276. DOI 10.1007/s12095-015-0170-5 $\longrightarrow$ 5 пъти

    1. S. Picek. *Evolutionary Computation and Cryptology*. Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion. New York, NY, USA ©2016, pp 883-909, ISBN: 978-1-4503-4323-7, DOI:10.1145/2908961.2927003
    2. S. Picek, M. Cupic, and L. Rotim. *A New Cost Function for Evolution of S-boxes*. Evolutionary Computation, Massachusetts Institute of Technology Press, DOI:10.1162/$EVCOa$00191, 2016

3. H. Isa, N. Jamil, and M.R. Z'aba*Construction of Cryptographically Strong S-Boxes Inspired by Bee Waggle Dance*. New Generation Computing, 34: 221. DOI:10.1007/s00354-016-0302-2, Springer, 2016
4. N.B. Cuong, N.V. Long, and H.D. Linh. *Analyzing the influence of linear redundancy in S-boxes with affine equivalence within XSL-like round functions*. ctcrypt.ru, 2016
5. T. Kapuscinski, R.K. Nowicki, and C. Napoli. *Application of Genetic Algorithms in the Construction of Invertible Substitution Boxes*. 15th International Conference on Artificial Intelligence and Soft Computing ICAISC'16, Zakopane, Poland, June 12-16, 2016, Proceedings, Part I, pp 380-391, Springer, 2016, DOI $10.1007/978-3-319-39378-033$

- [11] G. Ivanov, N. Nikolov, and S. Nikova. *Cryptographically strong S-boxes generated by Modified Immune Algorithm*. Springer International Publishing Switzerland 2016. E. Pasalic and L.R.Knudsen (Eds.): BalkanCryptSec 2015, LNCS 9540, pp.1-12, 2016. DOI 10.1007/978-3-319-29172-7-3 $\longrightarrow$ 2 пъти

1. S. Picek. *Evolutionary Computation and Cryptology*. Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion. New York, NY, USA ©2016, pp 883-909, ISBN: 978-1-4503-4323-7, DOI:10.1145/2908961.2927003
2. S. Picek, M. Cupic, and L. Rotim. *A New Cost Function for Evolution of S-boxes*. Evolutionary Computation, Massachusetts Institute of Technology Press, DOI:10.1162/$EVCOa$00191, 2016

# Библиография


[1] E. Biham. Observations on the relations between bit-functions of many s-boxes. In *The 3rd NESSIE conference*, November 2002.

[2] E. Biham and A. Shamir. Differential cryptanalysis of des-like cryptosystems. In *Advances in Cryptology - CRYPTO'90*, volume 537 of *LNCS*, pages 2–21. Springer Verlag, 1991.

[3] E. Biham and A. Shamir. Differential cryptanalysis of des-like cryptosystems. *Journal of Cryptology*, 4:3–72, 1991.

[4] K. Browning, J. Dillon, M. McQuistan, and A. Wolfe. An apn permutation in dimension six. *Finite Fields: Theory and Applications, Contemporary Mathematics*, 518:33–42, 2010.

[5] J. Brownlee. *Clever Algorithms: Nature-Inspired Programming Recipes.* LuLu, first edition, January 2011.

[6] J.A. Clark, J.L. Jacob, and S. Stepney. The design of s-boxes by simulated annealing. *New Generation Computing Archive*, 23(3), September 2005.

[7] N. T. Courtois and J. Pieprzyk. Cryptanalysis of block ciphers with overdefined systems of equations. In *Advances in Cryptology - ASIACRYPT'02*, volume 2501 of *LNCS*, pages 267–287. Springer Verlag, 2002.

[8] J. Daeman and V. Rijmen. *The design of Rijndael: AES - The advanced Encryption Standard.* Springer Verlag, 2002.

[9] J. Fuller and W. Millan. Linear redundancy in s-boxes. In *FSE'03*, volume 2887 of *LNCS*, pages 74–86. Springer, 2003.

[10] G. Ivanov, N. Nikolov, and S. Nikova. Reversed genetic algorithms for generation of bijective s-boxes with good cryptographic properties. *IACR Cryptology ePrint Archive (2014), Report 2014/801, http://eprint.iacr.org/2014/801.pdf.*

[11] G. Ivanov, N. Nikolov, and S. Nikova. Cryptographically strong s-boxes generated by modified immune algorithm. In E. Pasalic and L.R. Knudsen, editors, *International Conference on Cryptography and Information Security BalkanCryptSec'15*, volume 9540 of *LNCS*, pages 1–12. Springer, 2016.

[12] G. Ivanov, N. Nikolov, and S. Nikova. Reversed genetic algorithms for generation of bijective s-boxes with good cryptographic properties. *Cryptography and Communications. Descrete Structures, Boolean Functions and Sequences*, 8(2):247–276, 2016.

[13] Y. Izbenko, V. Kovtun, and A. Kuznetsov. The design of boolean functions by modified hill climbing method. *http://eprint.iacr.org/2008/111.pdf, 01.09.2013.*

[14] O. Kazymyrov, V. Kazymyrova, and R. Oliynykov. A method for generation of high-nonlinear s-boxes based on gradient descent. *IACR Cryptology ePrint Archive (2013)*.

[15] L.Goubin, A.Martinelli, and M.Walle. Impact of s-boxes size upon side channel resistance and block cipher design. In *AFRICACRYPT'13*, volume 7918 of *LNCS*, pages 240–259. Springer, 2013.

[16] M. Matsui. Linear cryptanalysis method for des cipher. In *Advances in Cryptology єЂ“ EUROCRYPT'93*, volume 765 of *LNCS*, pages 386–397. Springer Verlag, 1994.

[17] W. Millan. How to improve the nonlinearity of bijective s-boxes. In *Australian Conference on Information Security and Privacy 1998*, volume 1438, pages 181–192. Springer Verlag, 1998.

[18] W. Millan, L. Burnett, G. Carter, A. Clark, and E. Dawson. Evolutionary heuristics for finding cryptographically strong s-boxes. In *ICICS'99*, volume 1726 of *LNCS*, pages 263–274. Springer, 1999.

[19] W. Millan, A. Clark, and E. Dawson. Heuristic design of cryptographically strong balanced boolean functions. In *Advances in Cryptology - EUROCRYPT'98*, volume 1403 of *LNCS*, pages 489–499. Springer Verlag, 1998.

[20] N. Nikolov, G. Ivanov, and V. Ducheva. Generation of substitution tables with good cryptographic properties using artificial immune algorithms. In *International Scientific Conference MATTEX'14, Shumen 2014*, 2014.

[21] K. Nyberg. Perfect nonlinear s-boxes. In *Advances in Cryptology - EUROCRYPT'91*, volume 547 of *LNCS*, pages 378–386. Springer Verlag, 1992.

[22] K. Nyberg. Differentially uniform mappings for cryptography. In *Advances in Cryptology - EUROCRYPT'93*, volume 765 of *LNCS*, pages 55–64. Springer Verlag, 1994.

[23] L. Qu, Y. Tan, C. Li, and G. Gong. More constructions of differentially 4-uniform permutations on $f_{2^{2k}}$. In *arxiv.org/pdf/1309.7423*, 2013.

[24] L. Qu, Y. Tan, C. Tan, and C. Li. Constructing differentially 4-uniform permutations over $f_{2^{2k}}$ via the switching method. *IEEE Transactions on Inform. Theory*, 59(7):4675–4686, 2013.

[25] C. E. Shannon. Communication theory of secrecy systems. *Bell System Technical Journal*, 28:656–715, 1949.

[26] P. Tesar. A new method for generating high non-linearity s-boxes. *Radioengineering*, 19(1):23–26, 2010.

[27] A.M. Youssef and S.E. Tavares. On some algebraic structures in the aes round function. Technical Report 2002/144, Cryptology ePrint Archive, 2002.

[28] Y. Yu, M. Wang, and Y. Li. Constructing differential 4-uniform permutations from known ones. *IACR Cryptology ePrint Archive (2011), Report 2011/047, http://eprint.iacr.org/2011/047.pdf*, 2011.