BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

ĐẶNG VÂN TRƯỜNG

VỀ MỘT THUẬT TOÁN SINH SỐ GIẢ NGẪU NHIÊN DỰA TRÊN PHƯƠNG PHÁP TẠO DÃY PHI TUYẾN LỒNG GHÉP VỚI BẬC LỚN

LUẬN ÁN TIẾN SỸ KỸ THUẬT

HÀ NỘI – 2022

BỘ THÔNG TIN VÀ TRUYỀN THÔNG

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

ĐẶNG VÂN TRƯỜNG

VỀ MỘT THUẬT TOÁN SINH SỐ GIẢ NGẪU NHIÊN DỰA TRÊN PHƯƠNG PHÁP TẠO DÃY PHI TUYẾN LỒNG GHÉP VỚI BẬC LỚN

Chuyên ngành: Kỹ thuật điện tử

Mã số: 9.52.02.03

LUẬN ÁN TIẾN SĨ KỸ THUẬT ĐIỆN TỬ

NGƯỜI HƯỚNG DẪN KHOA HỌC

GS. TSKH. NGUYỄN XUÂN QUỲNH

HÀ NỘI – 2022

LỜI CAM ĐOAN

Nghiên cứu sinh xin cam đoan đây là công trình nghiên cứu của chính mình. Các số liệu, kết quả trong luận án là trung thực và chưa từng được công bố trong bất cứ công trình của bất kỳ tác giả nào khác.

Người cam đoan

Đặng Vân Trường

LỜI CẢM ƠN

Luận án tiến sỹ này được nghiên cứu sinh thực hiện tại Học viện Công nghệ Bưu chính Viễn thông dưới sự hướng dẫn khoa học của GS.TSKH Nguyễn Xuân Quỳnh. Nghiên cứu sinh xin được bày tỏ lòng biết ơn sâu sắc đối với GS.TSKH Nguyễn Xuân Quỳnh, TS. Lê Chí Quỳnh, TS Ngô Đức Thiện, các thầy đã định hướng khoa học, chỉ dẫn thực hiện những nhiệm vụ cần thiết cũng như tạo các điều kiện thuận lợi để luận án này được hoàn thành.

Nghiên cứu sinh xin được trân trọng cảm ơn Viện Khoa học Công nghệ Mật mã và Ban Cơ yếu Chính phủ đã tạo điều kiện để nghiên cứu sinh hoàn thành nhiệm vụ nghiên cứu.

Nghiên cứu sinh cũng xin chân thành cảm ơn Lãnh đạo Học viện Công nghệ Bưu chính Viễn thông, Khoa Đào tạo sau đại học và các đồng nghiệp đã luôn hỗ trợ, tạo điều kiện để hoàn thành công trình nghiên cứu này.

Cuối cùng là sự biết ơn tới gia đình, bạn bè, đồng nghiệp đã thông cảm, động viên giúp đỡ nghiên cứu sinh có thêm nghị lực để hoàn thành luận án này.

Hà nội – 2022.

MỤC LỤC

DANH MỤC CÁC KÝ HIỆU

Ký hiệu	Ý nghĩa
GF(p)	Trường Galois với đặc số p
$GF(p^n)$	Mở rộng trường Galois bậc n với đặc số p
$\oplus$	Phép cộng số nguyên trên trường GF(p) hoặc phép logic XOR nếu p=2
$\otimes$	Phép nhân số nguyên trên trường GF(p)
$\sum_{i=0}^{n-1} A_i$	Phép cộng tích lũy trên trường GF(p)
$Z(x^n)/g(x)$	Phép tính modulo đa thức
$\frac{S(d)}{g(d)}$	Phép tính modulo đa thức
A^T	Phép chuyển vị ma trận
$\{b_n\}$	Chuỗi các phần tử
$D[b_n]$	Biến đổi D của một chuỗi $\{b_n\}$
∞	Vị trí của dãy con chứa toàn phần tử 0 trong thứ tự lồng ghép I_P^T

DANH MỤC CÁC CHỮ VIẾT TẮT

Viết tắt	Tên tiếng Anh	Tên tiếng Việt
2G	Second Generation	Mạng thế hệ hai
3G	Third Generation	Mạng thế hệ ba
ACF	Auto Correlation Function	Hàm tự tương quan
ASG	Alternative Stop and Go	Dãy Stop-and-Go lần lượt
BTS	Base transceiver station	Trạm phát sóng cơ sở
CCF	Cross Correlation Function	Hàm tương quan chéo
CCIP	Conditional cochannel interference probability	Nhiễu đồng kênh có điều kiện
CDMA	Code Division Multiple Access	Đa truy nhập phân chia theo mã
CN	Core Network	Mạng lõi
CS	Chanel Switching	Chuyển mạch kênh
ELS	Equipvalence Linear Span	Khoảng tương đương tuyến tính
FDD	Frequency Division Duplex	Ghép song công phân chia theo tần số
FDMA	Frequence Division Mutiplex Access	Đa truy nhập phân chia theo tần số
FIR	Finite Impulse Response	Đáp ứng xung hữu hạn
GPS	Global Positioning System	Hệ thống định vị toàn cầu
GPRS	General Packet Radio Service	Dịch vụ gói vô tuyến
GSM	Global System For Mobile Communicatons	Hệ thống toàn cầu cho truyền thông di động
IMT	International Mobile Telecommunications	Thông tin di động toàn cầu
ISI	Inter-Symbol Interference	Nhiễu giữa các ký hiệu
ITU	International Telecommunications Union	Hiệp hội Viễn thông Quốc tế
LFSR	Linear Feedback Shift Register	Thanh ghi dịch phản hồi tuyến tính

LMS	Least Mean Square	Bình phương trung bình bé nhất
LP	Linear Predictor	Dự đoán tuyến tính
LTE	Long-term evolution	Phát triển dài lâu
LTP	Long Term Predictor	Dự đoán thời gian dài
MIMO	Multiple-input and multiple-output	Đa đầu vào và đa đầu ra
MISO	Multiple Input single Output	Đa đầu vào đơn đầu ra
NGN	Next Genneration Netword	Mạng viễn thông thế hệ mới
PN	Pseudo Noise	Chuỗi giả nhiễu
PRNG	Pseudo Random Number Generator	Bộ sinh số giả ngẫu nhiên
PSTN	Public Switched Telephone Network	Mạng điện thoại chuyển mạch công cộng
QoS	Quality of Service	Chất lượng dịch vụ
QPSK	Quadrature Phase Shift Keying	Khoá dịch pha vuông góc
TDD	Time Division Duplex	Ghép song công phân chia theo thời gian
TDMA	Time Division Multiplex Access	Đa truy nhập phân chia theo thời gian
UMTS	Universal Mobile Telephone System	Hệ thống viễn thông di động toàn cầu
WCDMA	Wideband Code Division Multiple Access	Đa truy nhập phân chia theo mã băng rộng
SDR	Software Define Radio	Vô tuyến điều khiển bằng phần mềm
RNG	Random Number Generator	Bộ sinh số giả ngẫu nhiên
TFlop/s	Teta FLOP per second	Số phép toán dấu phảy động trên một giây (tính theo đơn vị 10^{12})

DANH MỤC CÁC HÌNH VẼ

DANH MỤC CÁC BẢNG BIỂU

MỞ ĐẦU

1. Lý do chọn đề tài

Bài toán tạo ra các dãy số giả ngẫu nhiên là bài toán luôn được quan tâm nghiên cứu phát triển trong những năm gần đây, phục vụ nhiều yêu cầu trong thực tế sử dụng của ngành công nghệ thông tin nói chung và công nghệ viễn thông nói riêng. Dãy giả ngẫu nhiên được sử dụng phổ biến nhất là dãy m, cũng gọi là m-dãy. Các bộ tạo m-dãy được S.W. Golomb đặt nền móng từ thập kỷ 1960[21], dựa trên lý thuyết trường Galois.

Nhà toán học Stephen Wolfram đã nhấn mạnh rằng thuật toán m-dãy là thuật toán được sử dụng nhiều nhất trong lịch sử hiện đại [61]. Dãy giả ngẫu nhiên dựa trên m-dãy có các tính chất thống kê rất tốt phục vụ cho việc xáo trộn dữ liệu, cùng với giá trị hàm tương quan và tự tương quan rất nhỏ. Việc cài đặt các m-dãy có thể thực hiện rất hiệu quả chỉ bằng các mạch logic đơn giản cũng như trên phần mềm. Đó cũng là lý do các dãy giả ngẫu nhiên dựa trên m-dãy có nhiều ứng dụng rất rộng rãi trong công nghệ viễn thông hiện nay. Từ việc xác định độ lệch thời gian của tín hiệu GPS, phương pháp phân kênh theo mã sử dụng trong CDMA, họ thuật toán mã hóa bảo vệ kênh GSM A5/1, A5/2 và A5/3, hoặc thuật toán xáo trộn dữ liệu phục vụ các kênh truyền thông SATA, SDH đều sử dụng các biến thể của m-dãy. Trong thập niên 1980, các thuật toán mật mã dựa trên m-dãy cũng rất phát triển, có nhiều đóng góp trong việc bảo mật các hệ thống thông tin quân sự của các cường quốc lớn. Các nhà khoa học đã đưa ra các giải pháp xây dựng nên các hệ mã dòng dựa trên m-dãy, trong đó đưa ra các cấu trúc riêng, kết hợp nhiều m-dãy để tăng tính phi tuyến cũng như các tính chất mật mã của hệ mật [2] [19] [27].

Việc nghiên cứu phát triển các lý thuyết và thuật toán dựa trên m-dãy vẫn không ngừng được thực hiện. Trong hội nghị Asia Crypt 2004, chuyên gia mật mã Shamir đã có bài trình bày “Stream Ciphers: Dead or Alive?”[52], trong đó chỉ rõ các lợi thế và hướng phát triển của mã dòng và m-dãy. Ở Việt Nam có nhóm nghiên cứu dẫn đầu là TS. Lê Chí Quỳnh đã có nhiều năm theo đuổi hướng nghiên cứu

riêng về cấu trúc lồng ghép để xây dựng các bộ tạo dãy lồng ghép dựa trên m-dãy. Các tác giả tập trung vào vấn đề nghiên cứu lý thuyết, hướng đến việc xây dựng các dãy có độ phức tạp cao và tính chất tốt. Tuy nhiên về mặt thực hành, một số phương pháp đã đưa ra còn có độ phức tạp tính toán lớn, khó có khả năng ứng dụng trong thực tế. Các yêu cầu cụ thể hiện nay về các đường truyền dữ liệu băng thông rất lớn đều yêu cầu các dãy có bậc lớn (thực tế CDMA sử dụng dãy có chu kỳ 2^{42}-1, còn một số phiên bản SDH cũng đã sử dụng dãy có chu kỳ 2^{57}-1). Để có thể ứng dụng trong kỹ thuật mật mã, yêu cầu về bậc của dãy còn tiếp tục tăng lên để chống lại sức mạnh tính toán của các siêu máy tính và các thiết bị thám mã chuyên dụng.

Các phương pháp tạo dãy phi tuyến lồng ghép dựa trên m-dãy đã có

Các nghiên cứu của nhóm tác giả dẫn đầu là TS. Lê Chí Quỳnh đã xây dựng nên các bộ tạo dãy lồng ghép dựa trên m-dãy [1] [30] [50]. Đây là một thiết kế riêng biệt "kiểu Việt Nam" về bộ tạo dãy giả ngẫu nhiên dựa trên m-dãy, dựa trên lý thuyết biến đổi d và hàm vết. Dãy lồng ghép và dãy phi tuyến lồng ghép dựa trên m-dãy được đưa ra đã thỏa mãn các tính chất về phân bố thống kê, tính tương quan và đặc biệt dãy phi tuyến lồng ghép có tính phi tuyến được gia cố để có thể ứng dụng trong kỹ thuật mật mã.

Dãy lồng ghép (Interleaved sequence) là một kiến trúc riêng để xây dựng một dãy giả ngẫu nhiên dựa trên một m-dãy ban đầu với các tham số được lựa chọn theo yêu cầu. Dãy đầu ra có các tính chất tốt về phân bố và tương quan như đã trình bày trong công bố [J1].

Dãy phi tuyến lồng ghép là một phát triển của dãy lồng ghép, trong đó sử dụng 2 m-dãy ban đầu, song kết hợp với nhau theo phương pháp đặc trưng của dãy lồng ghép với mục tiêu đưa ra dãy đầu ra có tính phi tuyến cao hơn so với dãy lồng ghép .

Có 3 phương pháp tìm thứ tự lồng ghép đã được nghiên cứu [J1]

- Phương pháp sử dụng Biến đổi – d.
- Phương pháp sử dụng toán tử vết.

- Phương pháp tính trực tiếp m hàng đầu tiên của ma trận.

Dãy phi tuyến lồng ghép dựa trên m-dãy

Để tăng tính chất phi tuyến của dãy lồng ghép, nhóm tác giả [60] đã đề xuất phương án sử dụng tham số của hai dãy đầu vào có cùng bậc nhưng sinh bởi hai đa thức sinh khác nhau $f(x)$ và $g(x)$. Theo phương pháp xây dựng dãy lồng ghép, hai dãy đầu vào nói trên sẽ sinh ra hai dãy lồng ghép với các dãy con sinh bởi đa thức con $f_1(x)$ và $g_1(x)$ tương ứng. Nếu ta thay thế thứ tự lồng ghép của dãy con thứ nhất bằng thứ tự lồng ghép của dãy con thứ hai, dãy đầu ra sẽ là kết quả lồng ghép kết hợp của hai dãy đầu vào. Các tác giả đã chứng minh rằng dãy lồng ghép kết hợp này có tính chất phi tuyến tốt hơn dãy lồng ghép ban đầu, do đó nó được gọi là dãy phi tuyến lồng ghép.

Tiến sỹ Lê Chí Quỳnh đã đặt nền móng cho dãy lồng ghép từ năm 1986 [49] [50], các nghiên cứu này đã được các tác giả khác công nhận và tiếp tục ứng dụng, phát triển thêm [2] [24].

Tiếp sau đó TS Lê Minh Hiếu tiếp tục nghiên cứu về dãy lồng ghép tam phân và dãy phi tuyến lồng ghép [30]. Trong thời gian gần đây, tiến sỹ Bùi Lai An đã phát triển dãy lồng ghép đa cấp, đa chiều trong luận án tiến sỹ năm 2012 [1].

Trong những năm gần đây, nhóm nghiên cứu của tiến sỹ Lê Chí Quỳnh vẫn tiếp tục nghiên cứu phát triển dãy phi tuyến lồng ghép, khai thác khả năng cài đặt trong thiết bị phần cứng và ứng dụng vào nhiều bài toán thực tế như bài toán cảm biến nén, bài toán thủy vân số [51][55][56].

Ứng dụng của dãy giả ngẫu nhiên trong mật mã và các thách thức

Trong kỹ thuật mật mã hiện đại, ngoài sự đóng góp rất hiệu quả của các hệ mật khóa công khai, các hệ mật sử dụng khóa bí mật vẫn được sử dụng cho nhiệm vụ bảo mật phần lớn nội dung thông tin. Các hệ mật sử dụng khóa bí mật bao gồm các hệ mã khối và các hệ mã dòng. Một nhánh lớn trong các hệ mã dòng là phát triển của các bộ sinh số ngẫu nhiên dựa trên m-dãy.

M-dãy có các tính chất ngẫu nhiên rất tốt để ứng dụng trong các bộ tạo số giả ngẫu nhiên với mục đích trải đều phổ tín hiệu trong một khoảng. Khi ứng dụng m-dãy vào kỹ thuật mật mã, ngoài tính ngẫu nhiên ta cần quan tâm tới tính tuyến tính, đặc tính phân bố tương quan. Các tấn công phân tích mã đối với m-dãy trước hết khai thác các đặc tính này.

Tấn công phân tích mã đầu tiên đối với m-dãy là tấn công tuyến tính với kỹ thuật được đề ra bởi Massey, còn gọi là thuật toán Belekamp-Massey[41]. Nếu chỉ sử dụng một m-dãy đơn lẻ để mã hóa luồng thông tin, kẻ tấn công chỉ cần có được 2^n bit của dãy là đủ điều kiện tìm ra đa thức sinh của dãy, từ đó khôi phục toàn bộ dãy.

Các dãy giả ngẫu nhiên dựa trên m-dãy ứng dụng trong mật mã đều không dùng một m-dãy mà thường ghép nhiều m-dãy với nhau để tăng tính phi tuyến. Khi này phương pháp phân tích tuyến tính của Massey không thể áp dụng trực tiếp, các nhà phân tích đã đề xuất nhiều phương pháp phân tích khác. Trong luận án này tác giả đã phân tích phương pháp tính giá trị tương quan địa phương, là phương pháp có ảnh hưởng trực tiếp tới độ an toàn của dãy lồng ghép.

Các yêu cầu về độ an toàn của dãy giả ngẫu nhiên sử dụng trong mật mã

Cùng với sự phát triển không ngừng của khoa học công nghệ, năng lực xử lý của các hệ thống máy tính ngày càng tăng lên. Điều này làm cho yêu cầu tương ứng với các hệ mật cũng cần phải tăng lên. Một trong những tham số quan trọng nhất của hệ mã dòng dựa trên m-dãy là độ dài chu kỳ dãy, tham số này được quyết định bởi độ lớn của đa thức đặc trưng. Trong kỹ thuật mật mã không sử dụng một m-dãy đơn lẻ, song yêu cầu về kích thước dãy còn cần phải lớn hơn gấp nhiều lần.

Song song với yêu cầu về độ an toàn, các bài toán mật mã thực hành cũng đặt ra yêu cầu về hiệu năng tính toán. Với băng thông đường truyền liên tục được nâng cao, các thuật toán mật mã cũng cần đạt hiệu năng tương ứng để có thể ứng dụng trong thực tế.

Tính cấp thiết của đề tài

Với tiềm năng của dãy phi tuyến lồng ghép có thể ứng dụng trong kỹ thuật mật mã, một trong những yêu cầu cần thiết là cần có một thuật toán đủ hiệu quả để cài đặt dãy phi tuyến lồng ghép với bậc lớn cho đủ mức an toàn cần thiết, đồng thời cần khả thi về thực hành và đạt hiệu năng thực tế chấp nhận được.

Các nghiên cứu trước đây về dãy lồng ghép và dãy phi tuyến lồng ghép đều chỉ tập trung vào khía cạnh lý thuyết như chu kỳ dãy, hàm phân bố, hàm tương quan, khoảng tương đương tuyến tính. Các thử nghiệm trước đây hầu hết chỉ thực hiện trên các dãy có bậc thấp. Với cỡ bậc thỏa mãn các yêu cầu của kỹ thuật mật mã, một số phương pháp sinh dãy lồng ghép trước đây sẽ trở thành không khả thi trong thực hành.

Do vậy, một mục tiêu chính mà luận án này tập trung giải quyết là đưa ra một thuật toán hiệu quả để sinh dãy phi tuyến lồng ghép với bậc lớn, cùng với các đánh giá về độ phức tạp tính toán, độ phức tạp lưu trữ cũng như các đánh giá thực nghiệm. Đồng thời, tác giả luận án cũng đề xuất hiệu chỉnh một phương pháp sinh dãy để có thể sử dụng dãy phi tuyến lồng ghép trong kỹ thuật mật mã đảm bảo độ an toàn thực tế.

2. Mục tiêu nghiên cứu

Nội dung nghiên cứu của luận án nhằm vào các mục tiêu chính sau đây:

(i) Nghiên cứu các phương pháp sinh dãy phi tuyến lồng ghép, các phương pháp tấn công phân tích dãy giả ngẫu nhiên để từ đó đề xuất một phương pháp khả thi trong thực hành để sinh dãy phi tuyến lồng ghép với bậc lớn.

(ii) Đề xuất một thuật toán hiệu quả để sinh dãy phi tuyến lồng ghép với bậc lớn, phân tích đánh giá thuật toán đã đề xuất về độ phức tạp tính toán, độ phức tạp lưu trữ và các tính toán thực nghiệm.

3. Đối tượng nghiên cứu

Đối tượng nghiên cứu của luận án tập trung vào các phương pháp xây dựng dãy giả ngẫu nhiên dựa trên m-dãy ứng dụng trong cả truyền thông và kỹ thuật mật mã, bao gồm các vấn đề

(i) Phương pháp xây dựng dãy lồng ghép và dãy phi tuyến lồng ghép dựa trên m-dãy;

(ii) Thuật toán sinh dãy phi tuyến lồng ghép hiệu quả, khả thi trong thực tế;

(iii) Phương pháp đánh giá hệ mã dòng sử dụng dãy giả ngẫu nhiên dựa trên m-dãy.

4. Phạm vi nghiên cứu

Phạm vi nghiên cứu của luận án là các phương pháp, thuật toán xây dựng dãy giả ngẫu nhiên dựa trên m-dãy và các ứng dụng của dãy giả ngẫu nhiên dựa trên m-dãy trong truyền thông và kỹ thuật mật mã.

5. Phương pháp nghiên cứu

Dựa vào các lý thuyết về dãy giả ngẫu nhiên, các công cụ toán học để phân tích các đặc tính của dãy phi tuyến lồng ghép dựa trên m-dãy.

Sử dụng nguyên tắc lập trình để xây dựng thuật toán khả thi trong thực tế. Sử dụng lý thuyết độ phức tạp tính toán để phân tích độ phức tạp của thuật toán về mặt thời gian cũng như về tài nguyên sử dụng.

6. Nội dung nghiên cứu

- Nghiên cứu về lý thuyết m-dãy và ứng dụng của m-dãy trong mật mã, các bộ tạo dãy giả ngẫu nhiên dựa trên m-dãy với một số bộ tạo dãy cụ thể đã được ứng dụng trong mật mã.

- Nghiên cứu các phương pháp sinh dãy phi tuyến lồng ghép dựa trên m-dãy, đề xuất giải pháp sinh dãy phi tuyến lồng ghép có thể áp dụng trong ứng dụng bảo mật thông tin.

- Nghiên cứu, đề xuất thuật toán sinh dãy phi tuyến lồng ghép với đủ bậc lớn, thỏa mãn các yêu cầu ứng dụng của kỹ thuật mật mã. Đánh giá độ phức tạp tính toán và độ phức tạp lưu trữ của thuật toán.

7. Ý nghĩa khoa học và thực tiễn

Về mặt lý thuyết, luận án đã đề xuất một thuật toán hiệu quả để sinh dãy phi tuyến lồng ghép với bậc rất lớn cùng với các phân tích về mức độ hiệu quả của thuật toán. Luận án cũng đề xuất một phương pháp sinh dãy giả ngẫu nhiên dựa trên m-dãy và dãy phi tuyến lồng ghép có thể sử dụng trong kỹ thuật mật mã.

Về ý nghĩa thực tiễn, kết quả nghiên cứu đã đưa ra một thuật toán mã dòng có thể được đưa vào ứng dụng trong ngành Cơ yếu Việt Nam.

8. Bố cục của luận án

Ngoài phần mở đầu, phần kết luận và phần phụ lục, luận án gồm ba chương với bố cục như sau.

Chương 1: Tổng quan về bộ tạo dãy giả ngẫu nhiên dựa trên m-dãy, trình bày tổng quan về lý thuyết trường Galois, m-dãy trên trường GF(p) với giá trị đặc số p>2 và các tính chất của m-dãy. Trong chương này cũng phân tích một số ứng dụng của m-dãy trong lĩnh vực truyền thông cũng như trong kỹ thuật mật mã và nghiên cứu một số bộ tạo dãy giả ngẫu nhiên dựa trên m-dãy đang được sử dụng

Chương 2: Các phương pháp xây dựng dãy phi tuyến lồng ghép dựa trên m-dãy. Chương này đề cập tới thiết kế bộ tạo dãy giả phi tuyến lồng ghép dựa trên m-dãy và một số phương pháp xây dựng dãy lồng ghép, dãy phi tuyến lồng ghép dựa trên m-dãy, trong đó đi sâu vào phương pháp phân rã theo bước cho m-dãy và ứng dụng phân rã theo bước để đề xuất một phương pháp sinh dãy phi tuyến lồng ghép sử dụng các nội dung từ công bố [J2] của tác giả.

Chương 3: Thuật toán sinh dãy phi tuyến lồng ghép với bậc lớn ứng dụng trong kỹ thuật mật mã. Trong chương này trước hết tác giả thực hiện đánh giá một số phương pháp tấn công m-dãy điển hình. Phần tiếp theo là trình bày về thuật toán sinh dãy phi tuyến lồng ghép với bậc lớn tùy ý, cùng với các đánh giá

về lý thuyết cũng như tính toán thực hành cho việc cài đặt thực tế thuật toán này, theo công bố [J3]. Tác giả cũng đề xuất một cải tiến cho phương pháp sinh dãy luân phiên bằng cách đưa dãy phi tuyến lồng ghép thành một thành phần của dãy luân phiên.

Phần kết luận: tổng kết các đóng góp chính của luận án, khả năng ứng dụng và các vấn đề cần tiếp tục nghiên cứu;

CHƯƠNG 1 : TỔNG QUAN VỀ BỘ TẠO DÃY GIẢ NGẪU NHIÊN DỰA TRÊN M-DÃY

Các dãy chu kỳ tối đa (m-dãy) đã được đề xuất từ khá lâu, với xuất phát điểm từ lý thuyết trường Galois, và được thể hiện rõ trong các phương pháp xây dựng dãy cụ thể. Các dãy này đã có nhiều ứng dụng trong kỹ thuật điện tử, kỹ thuật viễn thông và đặc biệt là kỹ thuật mật mã.

Lý thuyết Galois đề cập tới nhiều khái niệm đại số phức tạp, nhưng các chi tiết cần thiết ứng dụng lý thuyết này để xây dựng m-dãy thì có thể trình bày thông qua các kiến thức số học sơ cấp, đặc biệt là phép tính modulo. Việc mở rộng trường Galois thành trường đa thức GF(p^n) cũng có thể diễn giải trên quan điểm các phép tính đa thức một cách rõ ràng. Trong phần đầu của chương này, tác giả sẽ trình bày chi tiết về các thông tin cần thiết của trường Galois GF(p) cũng như mở rộng trường GF(p^n) để có thể xây dựng nên m-dãy [12] [38].

Phần tiếp theo của chương sẽ trình bày về việc xây dựng cụ thể một m-dãy trên trường đa thức GF(p^n), với hai phương pháp xây dựng là phương pháp Galois và phương pháp Fibonacci [39]. Vấn đề thực hiện cài đặt m-dãy trên máy tính cũng được giới thiệu, cùng với một số bộ tạo dãy giả ngẫu nhiên đã được xây dựng dựa trên m-dãy và có ứng dụng rộng rãi trong kỹ thuật mật mã [19] [27].

1.1. Khái niệm trường Galois

1.1.1. Khái niệm trường Galois

Trường Galois [12] hay còn gọi là trường modulo theo đặc số p với p là số nguyên tố, ký hiệu là **GF(p)**, bao gồm tập xác định gồm p số nguyên trong khoảng [0, 1, …, p-1]và hai phép toán ký hiệu là ⊕ và ⊗, cụ thể là:

- Phép toán cộng, ký hiệu là ⊕, là phép cộng theo modulo p. Phần tử "0" của phép toán là giá trị 0. Phép toán ngược của phép cộng là phép trừ, cũng chính là phép trừ theo modulo (Chú ý là tập xác định của GF(p) không bao gồm các số

âm, vì thế kết quả phép trừ sẽ được hiệu chỉnh theo modulo *p* để được một giá trị nằm trong tập xác định của GF(*p*)).

Ví dụ: Trong trường GF(5) ta có:

$3 \oplus 4 = 2$ vì $3 + 4 = 7 = 2 \pmod 5$.

Ngược lại ta có $1 - 3 = 3 \pmod 5$.

- Phép toán nhân, ký hiệu là ⊗, là phép nhân theo modulo *p*. Phần tử "1" của phép toán là giá trị 1. Phép toán ngược của phép nhân modulo là phép chia modulo, phép chia modulo được định nghĩa khác với phép chia số nguyên thông thường. Cụ thể là $c = a/b \ mod \ p$ nghĩa là giá trị *c* thỏa mãn: $b \otimes c \equiv a \ mod \ p$ (≡ ký hiệu phép đồng dư), trong đó c là một số nguyên trong tập xác định của GF(p), và b có giá trị khác 0. Để thỏa mãn điều kiện mọi phép nhân đều có ngược, đặc số p cần phải là số nguyên tố.

Ví dụ: Trong trường GF(7) ta có:

$2 \otimes 4 = 1$ vì $2 \otimes 4 = 8 = 1 \ (mod \ 7)$.

$6 / 5 = 4$ vì $4 \otimes 5 = 20 = 6 \ (mod \ 7)$.

Để tính được giá trị phép chia modulo, trong thực tế ta sử dụng thuật toán Euclid mở rộng [5] để tính nghịch đảo modulo của một số, sau đó tính giá trị phép chia theo công thức:

$$C = a/b \ (mod \ p) = a \otimes b^{-1} \ (mod \ p) . \qquad (1.1)$$

Thuật toán Euclid mở rộng

Xuất phát từ thuật toán Euclid để tìm ước chung lớn nhất (gcd) của hai số nguyên không âm, và từ Định lý Bézout chỉ ra rằng, nếu $d = GCD(a, b)$ thì tồn tại hai số nguyên x, y sao cho $d = x\,a + y\,b$, người ta đã mở rộng thuật toán Euclid để giải phương trình Diophantine (là phương trình có dạng $ax+by = c$)[43]. Thuật toán Euclid mở rộng được viết bằng giả mã như sau:

Thuật toán 1.1*: Thuật toán Euclid mở rộng*

```
def ext_gcd(a,b):
    m = a
    n = b
    xm = 1
    ym = 0
    xn = 0
    yn = 1
    while (n != 0):
        q = m // n # chia lấy phần nguyên
        r = m % n # chia lấy phần dư
        xr = xm - q*xn
        yr = ym - q*yn
        m = n
        xm = xn
        ym = yn
        n = r
        xn = xr
        yn = yr
    return (xm, ym) # m = gcd(a,b) = xm * a + ym * b
```

Ta có thể rút gọn thuật toán để tính a^{-1} mod n bằng cách tính ext_gcd(a, n), trong đó chỉ cần lấy giá trị *xm*, không cần các tính toán với *ym*. Chú ý rằng chỉ có thể tính được giá trị a^{-1} mod n trong trường hợp $gcd(a,n) = 1$ (hay m =1 trong thuật toán trên).

Thuật toán 1.2*: Thuật toán Euclid mở rộng tính nghịch đảo* a^{-1} *mod b*

```
def ext_gcd(a,b):
    m = a
    n = b
    xm = 1
    xn = 0
    while (n != 0):
```

```
        q = m // n # chia lấy phần nguyên
        r = m % n # chia lấy phần dư
        xr = xm - q*xn
        m = n
        xm = xn
        n = r
        xn = xr
    if m != 1 # m = gcd(a,b)
        return (-1) # không tính được giá trị nghịch đảo
    return (xm) # xm = a-1 mod b
```

1.1.2 Phép mở rộng trường GF(p^n)

Xét vector n số nguyên trong đó các phần tử nằm trong trường GF(p):

$A = \{a_0, a_1, \ldots, a_{n-1}\}$.

Ta có thể biểu diễn vector này như một đa thức với biến x:

$A(x) = a_0 + a_1x + a_2x^2 + \ldots + a_{n-1}x^{n-1}$.

Ta gọi trường đa thức GF(p^n) là tập các đa thức nói trên cùng với hai phép tính cộng đa thức và nhân đa thức, lấy modulo theo một đa thức $g(x)$ có bậc n, trong đó các hệ số đều nằm trong GF(p), nghĩa là tuân thủ quy tắc của phép cộng và phép nhân trong GF(p). Trường đa thức này ký hiệu là $Z(x^n)/g(x)$. Sau này ta sẽ dùng ký hiệu phép chia hoặc phân số để biểu diễn phép modulo đa thức.

Chú ý là phép modulo theo đa thức $g(x)$ sẽ tính theo nguyên tắc modulo theo hệ số bậc cao nhất, nghĩa là một vector $S(x)$ với biểu diễn thành đa thức có bậc không nhỏ hơn n sẽ cần chia cho $g(x)$ để lấy được phần dư là một đa thức kết quả $S'(x)$ có bậc nhỏ hơn n, gọi là modulo của $S(x)$ theo $g(x)$. Trong trường hợp này, hai đa thức được so sánh với nhau theo bậc cao nhất của đa thức chứ không chỉ so sánh giá trị của từng phần tử.

Ví dụ:

$S(x) = x^3 + 2x^2 + 1$, $g(x) = 2x^3 + x + 2$ (xét trên GF(3)).

Theo logic thông thường, ta có thể cho rằng $S(x) < g(x)$ (do so sánh hệ số tương ứng với bậc cao nhất). Tuy nhiên kết quả phép modulo $S(x)$ mod $g(x)$ là:

$S'(x) = S(x) - 2*g(x) = 2x^2 + x$.

Điều này để bảo đảm $S'(x)$ luôn có bậc nhỏ hơn $g(x)$.

1.1.3 Xây dựng m-dãy từ trường GF(p^n)

Tác giả Golomb [21] đã chỉ ra rằng từ vector $A(x)$ như trên, sau mỗi bước nhân $A(x)$ với x theo modulo $g(x)$ và lấy ra một phần tử của $A(x)$, ta sẽ sinh ra một dãy số giả ngẫu nhiên có chu kỳ có thể lên tới 2^n-1. Trong lễ tang của Golomb năm 2016, nhà toán học Stephen Wolfram đã đánh giá rằng đây có lẽ là ý tưởng thuật toán được sử dụng nhiều nhất trong lịch sử, với hàng tỷ tỷ lần sinh bit giả ngẫu nhiên trên các thiết bị điện tử trên toàn thế giới[63].

Dãy số này còn được gọi là dãy có chu kỳ cực đại (maximum length sequence), hay gọi là m-dãy (m-sequence)[21]. Do phép nhân với x làm cho toàn bộ đa thức được dịch sang một phía, nên $A(x)$ còn gọi là thanh ghi dịch. Nội dung $A(x)$ tại mỗi thời điểm gọi là trạng thái hiện thời của dãy. Trạng thái m-dãy sau t bước bắt đầu từ trạng thái $S(x)$ là:

$$A(x) = S(x)*x^t/g(x) . \quad (1.2)$$

Để có thể sinh ra được dãy trên, đa thức $g(x)$ cần phải là đa thức bất khả quy (không có một ước đa thức nào). Trong trường hợp $g(x)$ có ước đa thức, có thể dãy sinh ra sẽ bị suy biến thành dãy chứa toàn các bit 0 sau một số bước. Khi $g(x)$ là đa thức bất khả quy, dãy sinh ra sẽ có chu kỳ tối đa là 2^n-1.

Để dãy đạt chu kỳ cực đại, đa thức $g(x)$ cần phải là đa thức nguyên thủy (còn gọi là đa thức nguyên tố - primitive polynomial), hay $g(x)$ là phần tử sinh của trường đa thức GF(p^n). Điều đó có nghĩa là $x^i/g(x)$ với $i = 0.. p^n$-1 sẽ sinh ra toàn bộ các thành phần của tập đa thức $A(x)$ tương ứng với tập các vector A. Chú ý rằng GF(p^n) chỉ là một trường hữu hạn đầy đủ khi $g(x)$ là đa thức nguyên thủy, với $g(x)$ bất kỳ thì ta chỉ có thể có vành đa thức.

Ta có thể tìm được các đa thức nguyên thủy từ tính chất sau:

Tính chất 1[40]**:** *Đa thức g(x) là đa thức nguyên thủy của vành đa thức GF(p^n) khi và chỉ khi g(x) là đa thức bất khả quy và số nguyên k nhỏ nhất mà g(x) là ước của đa thức x^k-1 là k = p^n-1.*

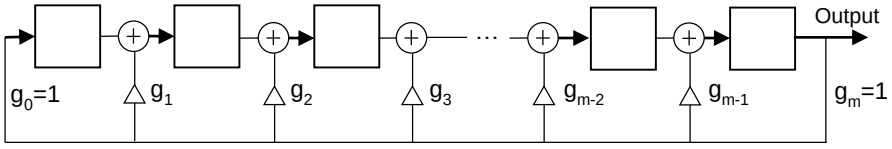
Có một phương pháp khác để kiểm tra đa thức bất khả quy *g(x)* với bậc n có là đa thức nguyên thủy hay không, bằng cách kiểm tra xem *g(x)* có phải là ước của các đa thức x^k-1 hay không, với *k* là các ước của số nguyên $N = p^n$-1.

1.1.4. Phương pháp xây dựng m-dãy trên trường đa thức GF(p^n):

Phương pháp Galois

Nội dung vector *A(x)* được lưu trữ trong một dãy các ô nhớ liên tiếp, gọi là thanh ghi. Sau một chu kỳ thời gian, giá trị mỗi ô nhớ được dịch sang ô nhớ bên cạnh (có thể thay đổi hoặc không, tùy vào phương pháp xây dựng dãy). Giải pháp lưu trữ và biến đổi này được gọi là thanh ghi dịch (Shift Register), đây là phương pháp cơ bản để xây dựng các dãy trên trường đa thức.

Các giá trị của *A(x)* được tính theo (1.2) [21], trong đó toàn bộ các giá trị của *A(x)* sẽ có thể thay đổi sau mỗi lần dịch chuyển, nếu giá trị ô nhớ đầu tiên của thanh ghi (ô nhớ sẽ bị loại bỏ sau phép dịch) là khác 0.



Hình 1.1 Sơ đồ xây dựng m-dãy theo Galois

Trong phương pháp Galois, các phần tử của thanh ghi đều có thể bị thay đổi sau mỗi bước, hoặc sẽ chỉ thực hiện dịch thanh ghi nếu $g_m = 0$.

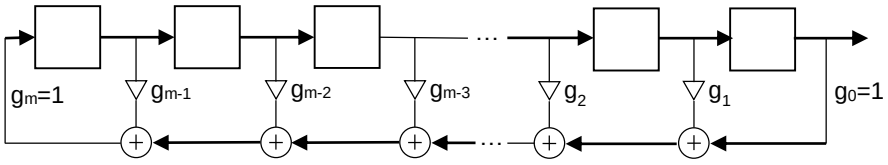
Phương pháp Fibonacci

Các giá trị của *A(x)* được tính theo phương pháp truy hồi, trong đó toàn bộ các giá trị của *A(x)* luôn dịch chuyển 1 bước, riêng giá trị đầu tiên (tương ứng với

x^0) sẽ được tính toán từ *A(x)* trước đó theo đa thức *g(x).* Phương pháp này dựa vào tính chất cơ bản của m-dãy.

Tính chất 2[20]*: Trong một m-dãy, giá trị của n+1 phần tử liên tiếp luôn là phụ thuộc tuyến tính với hệ số là đa thức sinh g(x).*

$$\sum_{i=0}^{n} A_i \otimes g_{n-i} = 0 \qquad (1.3)$$



Hình 1.2 Sơ đồ xây dựng m-dãy theo Fibonacci

Phương pháp này còn được gọi là thanh ghi dịch phản hồi tuyến tính (LFSR – Linear Feedback Shift Register), là phương pháp phổ biến nhất để xây dựng m-dãy.

Đã có chứng minh rằng kết quả sinh ra theo phương pháp Fibonacci cũng chính là kết quả sinh ra theo phương pháp Galois, nhưng giá trị khởi đầu của thanh ghi dịch sẽ thay đổi theo định lý 1.1.

Định lý 1.1[38]**:** *Một m-dãy sinh bởi phương pháp Fibonacy từ trạng thái khởi đầu là F = (F_0, …, F_{n-1}) và đa thức sinh g(x), cũng có thể sinh ra bằng phương pháp Galois từ giá trị khởi đầu F' =* (F'_0, …, F'_{n-1}) với $F'_i = \sum_{j=0}^{i} F_{i-j} g_{n-j}$.

Chú ý với các m-dãy trên GF(p) với p>2

Các tài liệu về m-dãy hầu hết được trình bày trong trường hợp dãy nhị phân, tức là xét trường hợp trường GF(2). Trong trường hợp đó, phép cộng và phép trừ là trùng nhau và trùng với phép tính logic tuyển loại trừ (XOR), phép nhân trùng với phép tính logic AND, phép nghịch đảo không cần thực hiện do trong GF(2) chỉ có thể có một giá trị duy nhất khác 0 là giá trị 1, và nghịch đảo của 1 là chính nó. Khi xem xét m-dãy trên trường GF(*p*) với *p*>2, khi thực hiện phép modulo đa thức cũng phải thực hiện đầy đủ phép nhân và phép trừ thay vì chỉ là phép XOR. Cụ thể hàm phản hồi được tính như sau:

$$A_n = -g_n^{-1} \sum_{i=0}^{n-1} A_i \otimes g_{n-1-i} \,. \qquad (1.4)$$

Ở đây phép nhân và phép cộng tích lũy đều được tính trên trường GF(p), hay tính theo modulo p.

Cài đặt thuật toán sinh m-dãy theo phương pháp Fibonacy trên máy tính

Thanh ghi A sẽ được lưu trữ trong một mảng số nguyên, mỗi phần tử mảng là một phần tử của vector theo đúng thứ tự. Mảng A được gán giá trị khởi đầu bằng giá trị trạng thái ban đầu của m-dãy. Đa thức sinh *g(x)* cũng được lưu trữ trong mảng số nguyên tương ứng g.

Trong mỗi vòng lặp ta sẽ tiến hành tính giá trị phản hồi theo (1.3) bằng cách thực hiện một vòng lặp nội bộ bên trong.

Giá trị đầu ra của thanh ghi là giá trị A[0] trước khi dịch chuyển.

Sau đó ta dịch toàn bộ thanh ghi về phía ô số 0, giá trị A[n-1] còn trống sẽ được gán bằng giá trị phản hồi vừa tính được.

Toàn bộ thuật toán có thể trình bày bằng giả mã như sau:

Thuật toán 1.3*: Thuật toán sinh 1 bit cho m-dãy*

```
An = 0;
For i=0 to n-1
        An = (An + A[i] * g[n-1-j]) % p // % ký hiệu phép modulo
Next i
An = An * InvMod (g[n], p) // InvMod: hàm thực hiên nghịch đảo
                                        // theo modulo
An = SubMod (0, An, p) // SubMod: hàm thực hiện phép trừ
                                        // theo modulo
For i=1 to n-1
        A[i-1] = A[i]
Next i
A[n-1] = An
```

Chú ý: Nếu ta thực hiện thuật toán trên trường GF(2), khi đó phép nhân được thay bằng phép AND, phép cộng modulo thay bằng phép XOR, việc lấy nghịch đảo modulo và trừ modulo không cần thực hiện. Khi đó ta sẽ không còn các phép tính số học trên số nguyên mà chỉ còn các phép tính logic trên bit.

Trong thuật toán trên cần thực hiện phép tính nghịch đảo theo modulo p. Thực tế ta có thể tránh việc tính nghịch đảo này bằng giải pháp chọn g_n = p-1, khi đó $-g_n^{-1}\ mod\ p$ = 1, hoặc ta có thể tính trước giá trị $-g_n^{-1}\ mod\ p$ để tăng tốc độ tính toán của thuật toán. Chú ý là việc biến đổi đa thức $g(x)$ bằng cách nhân với một số nguyên trong GF(p) không làm thay đổi đặc tính cũng như tính chất của dãy sinh ra.

1.2. Ứng dụng của dãy giả ngẫu nhiên dựa trên m-dãy

1.2.1 Môt số ứng dụng phổ biến của dãy giả ngẫu nhiên dựa trên m-dãy

Trong thực tế các chuỗi thanh ghi dịch luôn được sử dụng trong hầu hết các trường hợp khi các bit dữ liệu được truyền đi trong các hệ thống truyền thông hiện đại, máy tính và nhiều các thiết bị điện tử khác. Mặc dù có rất nhiều công nghệ khác nhau với các tên gọi khác nhau, trong chúng thường chứa các chuỗi thanh ghi dịch phản hồi tuyến tính (PN, pseudonoise, M-FSR, LFSR, truyền thông trải phổ, MLS, SRS, PRBS,…).

Trong lĩnh vực mạng điện thoại di động, việc sử dụng chuỗi thanh ghi dịch có nhiều thay đổi trong những năm qua. Mạng 2G dựa trên TDMA, sử dụng chuỗi thanh ghi dịch trong việc mã hóa bảo vệ dữ liệu. Mạng 3G là môi trường truyền thông sử dụng CDMA, trong đó các chuỗi thanh ghi dịch có đóng góp chính trong việc phân chia miền tần số. Các mạng 4G thường sử dụng kết hợp các khe thời gian và khe tần số, không liên quan trực tiếp đến các chuỗi thanh ghi dịch, mặc dù vẫn sử dụng đến CRC để xử lý toàn vẹn dữ liệu khi cửa sổ tần số trùng nhau. Mạng 5G được thiết kế phức tạp hơn với thích ứng linh hoạt để sử dụng các khe thời gian và

tần số một cách tối ưu. Nhưng một số kênh của 5G thường được phân bổ cho các "tín hiệu dẫn đường", hoạt động bằng cách truyền các chuỗi thanh ghi dịch .

Trong hầu hết các trường hợp, các thanh ghi dịch được sử dụng là các thanh ghi dịch sinh ra các chuỗi có độ dài tối đa. Một thuộc tính cơ bản của các chuỗi đó là chúng có tổng số 0 và 1 tương đương nhau. Các nghiên cứu sau đó cho thấy rằng chúng cũng có cùng số cặp bit 00, 01, 10 và 11 và ngay cả tần suất của các khối bít lớn hơn cũng tương đương nhau. Thuộc tính cân bằng theo từng bit và bộ bít sẽ gần đúng với bất kỳ chuỗi ngẫu nhiên nào đủ dài chứa các bit 0 và 1. Nhưng với chuỗi thanh ghi dịch độ dài tối đa, các thuộc tính này luôn luôn chính xác tuyệt đối. Các chuỗi này theo một nghĩa nào đó có một số ý nghĩa của sự ngẫu nhiên, nhưng theo một cách rất hoàn hảo, có thể thực tế là chúng không phải là ngẫu nhiên, mà thay vào đó có một cấu trúc có tổ chức, rất rõ ràng. Cấu trúc này của chuỗi làm cho các thanh ghi dịch phản hồi tuyến tính không trực tiếp sử dụng được trong kỹ thuật mật mã. Nhưng dạng chuỗi này phù hợp đối với các yêu cầu cơ bản về việc xáo trộn dữ liệu và các hệ mã hóa đơn giản.

Ứng dụng thanh ghi dịch trong việc ngẫu nhiên hóa dữ liệu

Một mục tiêu rất phổ biến chỉ là để biến một tín hiệu thành dạng nhiễu trắng (white noise). Điều này rất phổ biến khi ta muốn truyền dữ liệu có chứa chuỗi rất nhiều bit 0 (hoặc 1) liên tục, làmcác thiết bị điện tử thu nhận chuỗi này có thể bị nhầm lẫn nếu chúng thấy kênh truyền giữ im lặng (mức "0") quá lâu. Ta có thể tránh được vấn đề này bằng cách xáo trộn dữ liệu gốc, sử dụng cách kết hợp nó với một chuỗi thanh ghi dịch, từ đó luôn luôn có sự thay đổi bit dữ liệu trên đường truyền. Đó là những gì đang được sử dụng trong Wi-Fi, Bluetooth, USB, TV kỹ thuật số, Ethernet cung như hầu hết mọi loại bus dữ liệu nối tiếp khác (PCIe, SATA, v.v.).

Để ngẫu nhiên hóa tín hiệu, ta cho dòng bit đầu vào đi vào thanh ghi dịch bậc n. Giá trị đầu ra của thanh ghi dịch được tính tương tự như giá trị phản hồi của m-dãy, song không sử dụng để phản hồi mà sử dụng làm dữ liệu sẽ truyền trên kênh.

Tới đầu thu tín hiệu, các bit tín hiệu từ kênh sẽ đi vào thanh ghi dịch với các thông số (bậc, đa thức sinh) trùng với thanh ghi dịch bên phát. Giá trị đầu ra của thanh ghi dịch này cũng được tính tương tự như giá trị phản hồi của m-dãy.

Nếu dữ liệu truyền trên kênh là chính xác (không có lỗi), giá trị đầu ra của thanh ghi dịch sẽ trùng khớp với giá trị dòng bit tín hiệu cần truyền, song bị giữ chậm n vị trí. Trong trường hợp xuất hiện 1 bit lỗi, bit lỗi này sẽ làm cho n bit đầu ra ở bên thu bị sai, song các bit tiếp theo vẫn nhận giá trị đúng.

Đặc tính quan trong của kênh dữ liệu được áp dụng thanh ghi dịch là dòng bit dữ liệu trên kênh luôn có tính giả ngẫu nhiên, không phụ thuộc vào nội dung dòng dữ liệu thực sự cần truyền. Đặc tính này giúp phổ tín hiệu san đều ngay cả khi dữ liệu đầu vào chứa các đoạn rất nhiều bit 0 hoặc bit 1 đứng liền nhau.

1.2.2. Mật mã dòng và ứng dụng của m-dãy trong mã dòng

Khái niệm mã dòng

Theo Meneze [43]: Mã dòng là luồng các ký tự riêng lẻ được mã hóa (thường là các chữ số nhị phân) của một bản rõ tại một thời điểm, sử dụng một chuyển đổi mã hóa thay đổi theo thời gian. Ngược lại, mật mã khối luôn đồng thời mã hóa một nhóm ký tự của bản rõ bằng cách sử dụng một phép chuyển đổi mã hóa cố định.

Mật mã khối hoạt động với một chuyển đổi cố định trên các khối lớn dữ liệu văn bản; mật mã luồng hoạt động với sự biến đổi theo thời gian trên các chữ số văn bản riêng lẻ.

Lợi thế của mã dòng

Mật mã dòng là một xu hướng rõ ràng trong 30 năm qua, luôn được thúc đẩy bởi những thay đổi công nghệ cơ bản. Trong tương lai gần, chưa có khả năng thực hiện các tấn công để phá vỡ các hệ mật mã dòng.

So sánh với mã khối, lợi thế của mã khối là có sẵn các tiêu chuẩn mã khối được bảo trợ như mã DES, AES. Việc thiết kế mã khối cũng có nhiều lựa chọn để

xây dựng các khối đa năng và có các phân tích kỹ hơn về các vấn đề an toàn. Mã khối cũng được hỗ trợ rất tốt từ các tài liệu giáo khoa và các khóa học chính quy.

Tuy nhiên mã dòng vẫn còn các lợi thế khác để có thể sử dụng trong các thuật toán mã hóa ngày hôm nay. Đó là kích thước mã nhỏ hơn khi triển khai trong các thiết phần cứng tối thiểu, tốc độ mã hóa cao hơn (trong một số trường hợp); độ trễ đầu vào - đầu ra nhỏ hơn do không cần thu thập đủ khối dữ liệu. Với mã dòng ta chỉ cần các giao thức đơn giản để xử lý các đầu vào có kích thước nhỏ hoặc kích thước thay đổi.

Tuy nhiên, tầm quan trọng của mã dòng đang giảm dần do các lý do sau: Phần cứng đang trở nên lớn hơn và rẻ hơn, nhiều ứng dụng có thể được xử lý hoàn toàn bằng phần mềm. Vấn đề tốc độ cũng thường không phải là vấn đề quan trọng nhất khi thực hiện mã hóa dữ liệu. Các gói dữ liệu tiêu chuẩn hiện nay như gói chuyển mạch ATM khiến ta không có nhu cầu xử lý các tín hiệu đầu vào có kích thước nhỏ hoặc kích thước thay đổi

Tuy vậy, mật mã dòng sẽ vẫn giữ được thế cạnh tranh trong hai loại ứng dụng: Các lược đồ định hướng sử dụng phần cứng với kích thước đặc biệt nhỏ (xét theo số cổng, năng lượng tiêu thụ điện… như thiết bị RFID); hoặc các lược đồ hướng phần mềm yêu cầu tốc độ đặc biệt cao như bộ định tuyến cáp quang…

Xu hướng phát triển của mật mã dòng

Cho đến những năm 1960, mọi nơi đều sử dụng đến mật mã dòng: Các dịch vụ quân sự và ngoại giao, các tổ chức gián điệp, các nhà cung cấp viễn thông, công ty lớn, v.v. Các sơ đồ mã thường là khóa sử dụng một lần, hoặc một sơ đồ liên quan dựa trên sự phát sinh giả ngẫu nhiên dựa trên điện – cơ như máy Enigma. Các máy tính Mainframe đã xuất hiện, nhưng chỉ được sử dụng nhiều hơn trong việc phân tích mã chứ không phải việc mã hóa dữ liệu.

Vào năm 1960, các thiết bị mã hóa điện tử dựa trên bóng bán dẫn bắt đầu xuất hiện Các thiết bị mới có rất ít bộ nhớ, do đó mật mã luồng tiếp tục phổ biến hơn nhiều so với mật mã khối. Từ đó dẫn đến sự phổ biến của một sơ đồ thiết kế

mới của mã dòng: thanh ghi dịch phản hồi tuyến tính. Thiết kế này được hỗ trợ bởi một lý thuyết toán học phát triển tốt từ lý thuyết trường Galois cùng với những bổ sung cần thiết của Golomb.

Cuối thế kỷ 20, đã có sự xuất hiện của mật mã khối hiện đại. Do các máy tính, vệ tinh, điện thoại bắt đầu sử dụng các gói định hướng khối. Các cổng, bộ nhớ và bộ vi xử lý dựa trên VLSI bắt đầu xuất hiện, vượt qua được các hạn chế về tốc độ và kích thước vi mạch. Vì thế mật mã khối trở nên dễ dàng thực thi. Các dịch vụ quân sự tiếp tục sử dụng mật mã dòng, nhưng các ứng dụng thương mại yêu cầu sử dụng mật mã khối.

Một số triển khai cụ thể đã tiến hành thay thế mật mã dòng bằng mật mã khối. Mạng di động GSM trước đây sử dụng hệ mã dòng A5/x ở thế hệ 2G, khi chuyển sang 3G đã thay thế bằng mã khối Kasumi. Mạng không dây Wi-Fi trước đây sử dụng mã dòng RC4 trong phiên bản 802.11a/b, tới các phiên bản hiện nay đều sử dụng mã khối AES trong việc mã hóa dữ liệu trên đường truyền.

Một số xu hướng thiết kế các hệ mật mã dòng mới hiện nay bao gồm:

- Sử dụng các word 32/64 bit làm thành phần tính toán cơ bản để tăng tốc độ xử lý trong các vi xử lý hiện đại.
- Sử dụng lệnh chuyên biệt của hệ vi xử lý .
- Có thể vay mượn các thành phần từ mật mã khối: Hộp thế S-box, bộ chuyển vị
- Tránh các cấu trúc tuyến tính và Trộn các miền đại số.

Để có các chỉ dẫn chung về thiết kế hệ mật mã dòng mới, ta cần tuân theo một số nguyên tắc sau:

- Sử dụng các thiết kế tối giản.
- Nghiên cứu các cuộc tấn công nguyên thủy và tấn công tổng quát mới.
- Thiết kế cấu trúc khóa hai cấp.

- Tránh các kỹ thuật thám mã cổ điển đã được các kẻ tấn công nghiên cứu kỹ.
- Thêm các cơ chế tăng cường bảo mật, giảm phụ thuộc trong các thiết kế hệ mã dòng mới.

So sánh độ mạnh của mật mã dòng và mật mã khối

Mật mã dòng dường như trở nên yếu hơn so với mật mã khối vì các lý do sau:

Các phương pháp tấn công vào mật mã khối (như tấn công vi sai) cũng áp dụng được cho mật mã dòng; Song các phương pháp tấn công vào mật mã dòng (như tấn công tương quan) không thể áp dụng cho mật mã khối.

Mặt khác các tấn công đại số sẽ hữu ích hơn đối với các thuật toán mã hóa dòng (đặc biệt là các thuật toán dựa trên LFSR). Việc đoán và thiết lập các tấn công vào mật mã dòng có thể khôi phục khóa hoặc bất kỳ trạng thái nào của bộ tạo khóa.

Với một hệ mật mã khối mới, ta luôn có một bộ công cụ hoàn thiện để đánh giá tính bảo mật của nó. Với một hệ mật mã dòng mới, lỗ hổng của nó có nhiều khả năng là theo một dạng duy nhất. Với một mật mã khóa công khai mới, có khả năng nó không đủ an toàn.

Ứng dụng thanh ghi dịch trong mật mã dòng

Ta biết rằng một m-dãy độc lập có tính tuyến tính hoàn toàn, do đó không thể sử dụng trực tiếp trong kỹ thuật mật mã. Giải pháp thường được sử dụng là kết hợp nhiều m-dãy với nhau để thiết kế lên một hệ mã dòng [19][26]. Khi đó hệ mã dòng sẽ có tính phi tuyến cao, đồng thời chu kỳ của dòng bit khóa thường bằng tích các chu kỳ của các m-dãy thành phần.

Để mã hóa một nội dung dữ liệu, người dùng cần sử dụng một cụm từ khóa bí mật chia sẻ trước (passphrase). Từ cụm passphrase, người mã hóa sẽ tính ra dãy bit được sử dụng làm giá trị khởi đầu cho các trạng thái trong của các m-dãy là thành phần của hệ mã dòng. Từ giá trị khởi đầu này, hệ mã dòng sẽ sinh ra chuỗi

bit khóa có độ dài bằng độ dài bản rõ. Bản mã sẽ được tính bằng phép XOR bản rõ và chuỗi bit khóa theo vị trí tương ứng.

$$\text{CipherText} = \text{PlainText} \oplus \text{KeyStream} \quad (1.5)$$

Ở phía người giải mã cũng có cụm passphrase giống như cụm passphrase ở bên mã hóa. Cụm passphrase này có thể được thỏa thuận từ trước hoặc truyền qua một kênh truyền an toàn theo một cách nào đó. Để giải mã dữ liệu, người giải mã sẽ tính ra dãy bit được sử dụng làm giá trị khởi đầu cho các trạng thái trong của các m-dãy là thành phần của hệ mã dòng. Từ giá trị khởi đầu này, hệ mã dòng sẽ sinh ra chuỗi bit khóa có độ dài bằng độ dài bản mã. Bản rõ sẽ được tính bằng phép XOR bản mã và chuỗi bit khóa theo vị trí tương ứng.

$$\text{PlainText} = \text{CipherText} \oplus \text{KeyStream} \quad (1.6)$$

Bài toán tấn công thám mã với hệ mật dựa trên thanh ghi dịch

Từ một số bít khóa thu được, cần tính toán tìm ra giá trị khởi đầu của các dãy, từ đó khôi phục toàn bộ chuỗi bit khóa. Các nhà thám mã tìm cách phân tích thiết kế của các hệ mã dòng, cố gắng tìm ra các điểm yếu về tính tuyến tính hoặc tính tương quan, từ đó đưa ra phương pháp tìm ra giá trị khởi đầu của các dãy sao cho tiêu tốn ít tài nguyên và công sức nhất, bao gồm yêu cầu về độ dài chuỗi bit khóa thu được và số bước tính toán cần thiết. Thông thường các tấn công thám mã với thanh ghi dịch chỉ giảm bớt độ phức tạp tính toán so với việc vét cạn toàn bộ không gian các giá trị khởi đầu của mọi m-dãy cấu tạo nên hệ mã dòng, song ít nhất người tấn công cũng phải vét cạn giá trị khởi đầu của một dãy trong số các dãy đó.

Bảng 1.1 Năng lực tính toán của các hệ thống siêu máy tính Tháng 6/2021

TT	Hệ thống	Số lõi CPU	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
1	Supercomputer Fugaku - Supercomputer Fugaku A64FX 48C 2.2GHz Tofu interconnect D Fujitsu RIKEN Center for Computational Science Japan	7 630 848	442 010	537 212	29 899
2	Summit - IBM Power System AC922 IBM POWER9 22C 3.07GHz NVIDIA Volta GV100 Dual-rail Mellanox EDR Infiniband IBM DOE/SC/Oak Ridge National Laboratory United States	2 414 592	148 600	200 794	10 096
3	Sierra - IBM Power System AC922 IBM POWER9 22C 3.1GHz NVIDIA Volta GV100 Dual-rail Mellanox EDR Infiniband IBM / NVIDIA / Mellanox DOE/NNSA/LLNL United States	1 572 480	94 640	125 712	7 438

Yêu cầu của m-dãy trong mật mã

Tính chất mật mã: có đủ độ an toàn chống lại tấn công tuyến tính và tấn công tương quan.

Độ phức tạp tính toán: tương đương với các mã khối hiện đại (A5/3 có cấu trúc 3 m-dãy với tổng bậc 128, chu kỳ chung cỡ 2^{128}). Với sự phát triển của các hệ thống tính toán hiệu năng cao cùng với các thiết bị tính toán dựa trên GPU/ASIC, yêu cầu về độ phức tạp tính toán ngày càng tăng. Nếu như trong những năm 1990, độ phức tạp tính toán cỡ 2^{50} đã được coi là an toàn, trong thời gian gần đây độ phức tạp tính toán của hệ thống siêu máy tính mạnh nhất đã đạt tới 2^{60} phép tính/s như trong công bố trên trang web Top500 trong bảng 1.1 [63]. Vì thế các hệ mã dòng sử dụng m-dãy thường yêu cầu các dãy thành phần có bậc tối thiểu 128.

1.3. Một số bộ tạo dãy giả ngẫu nhiên dựa trên m-dãy

Trong phần này tác giả sẽ giới thiệu một số thiết kế bộ tạo dãy giả ngẫu nhiên có khả năng ứng dụng trong mật mã, trong đó dựa vào thành phần chính là các m-dãy thành phần. Các bộ tạo dãy này đều đã được nghiên cứu phát triển trong thời gian dài, bằng cách kết hợp các m-dãy, các tác giả đã xây dựng nên các bộ tạo dãy giả ngẫu nhiên có tính phi tuyến cao, phân bố đều, chống lại được nhiều tấn công thám mã áp dụng cho mã dòng [7][19][27].

1.3.1 Bộ tạo dãy Gold

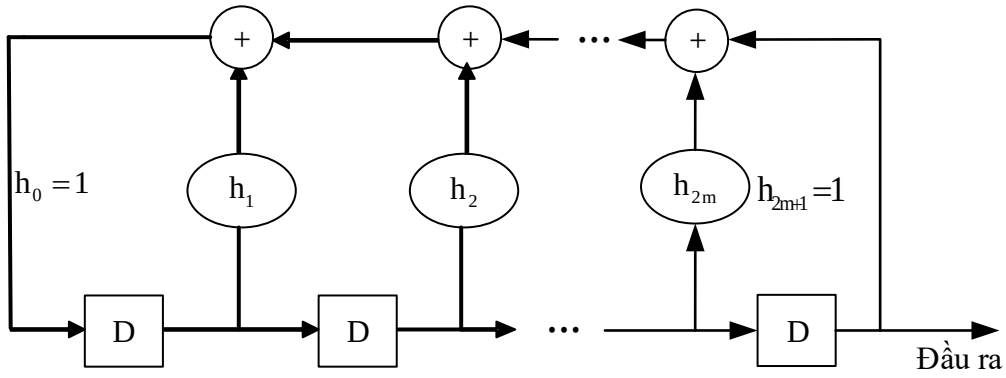
Hầu hết trong các ứng dụng như trong thông tin vũ trụ, vệ tinh dải rộng, thông tin di động nhiều địa chỉ… số dãy cần thiết trong tập hợp là rất lớn. Trong một hệ thống truy cập ngẫu nhiên hoặc các hệ thống hỗn hợp, kích thước cần thiết của tập hợp có thể dễ dàng vượt qua con số vài trăm. Với sự ra đời của dãy Gold vào những năm 60 của thế kỉ trước đã đưa ra một lớp chứa đựng một số lượng lớn các dãy thỏa mãn tính chất ACF và CCF với một mức độ chấp nhận được[19].

Dãy Gold có thể được cấu tạo từ bất kì cặp dãy m nào. Gọi $a = \{a_n\}$, $b = \{b_n\}$ là một cặp m dãy có chu kì $N = 2^m - 1$ được sinh ra từ các đa thức nguyên thủy $h_1(d)$ và $h_2(d)$ có bậc m. Tập hợp Gold kí hiệu là $G(a,b)$ được cấu tạo như sau:

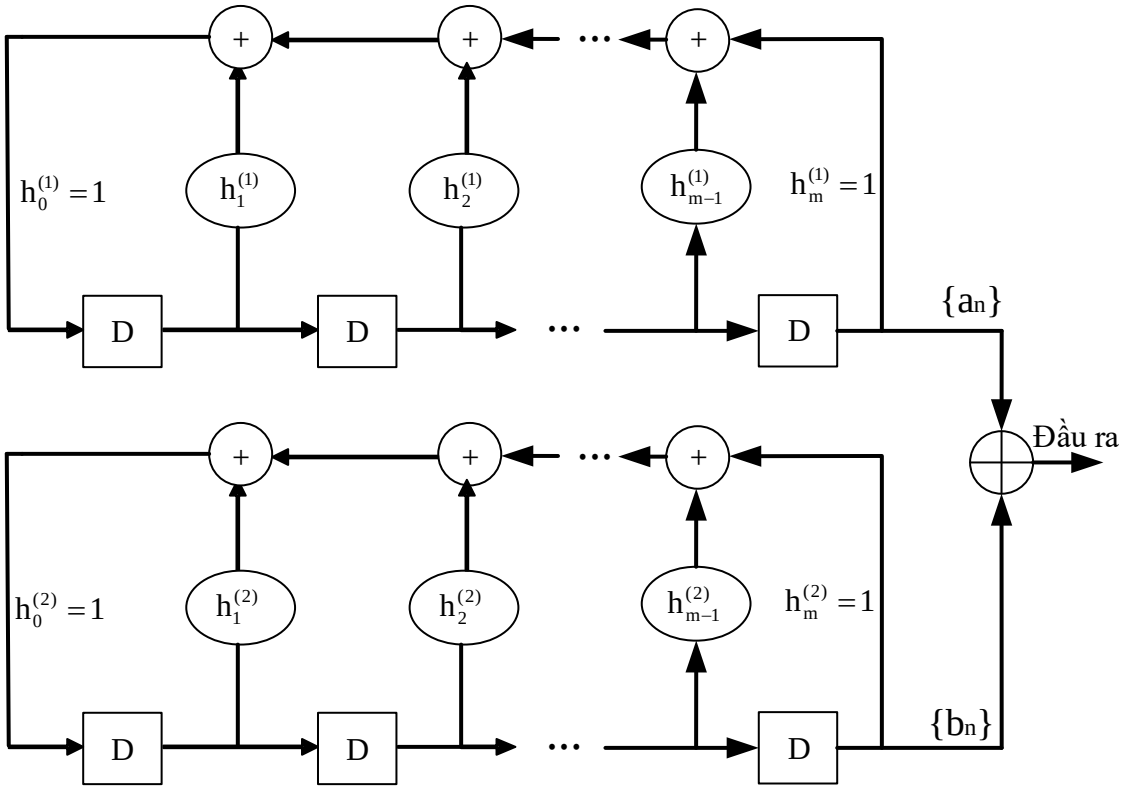
$$G(a,b) = \{a, b, a \oplus b, a \oplus Tb, a \oplus T^2b, \ldots, a \oplus T^{N-1}b\} \quad (1.7)$$

trong đó, T là phép dịch dãy, $G(a,b)$ chứa $N + 2 = 2^m + 1$ dãy có chu kì $N = 2^m - 1$.

Giả sử đa thức $h(d) = h_1(d)h_2(d)$ thì tập hợp tất cả các dãy được tạo bởi $h(d)$ là tập hợp các dãy được tạo bởi $a \oplus b$. Vậy, có 2 phương pháp khác nhau để có thể tạo ra dãy Gold có chu kì $N = 2^m - 1$ bằng việc sử dụng các thanh ghi phản hồi được mô tả như trong hình 1.3 và 1.4.



Hình 1.3 LFSR tạo dãy Gold kiểu I



Hình 1.4 LFSR tạo dãy Gold kiểu II

Vì đa thức $h_1(d)$ và $h_2(d)$ là hai đa thức nguyên thủy, nên ta có:

$$deg(h(d)) = deg(h_1(d)) + deg(h_2(d)) = 2m\ . \qquad (1.8)$$

Điều này có nghĩa là dãy Gold thỏa mãn quan hệ hồi quy tuyến tính bậc $2m$, hay nói cách khác ELS của dãy Gold là $2m$. Mặc dù giá trị này cao hơn bậc của đa thức $h_1(d)$ và $h_2(d)$ nhưng nó vẫn còn thấp hơn nhiều so với yêu cầu của các hệ thống bảo mật.

Vậy, dãy Gold tạo nên một tập hợp lớn các dãy có tính chất tương quan ACF và CCF tốt nhưng giá trị khoảng tuyến tính chưa cao.

Ví dụ 2.3: Cho dãy Gold có $m = 5$, theo bảng 1.2 ta có với $m = 5$ có tất cả 6 đa thức nguyên thủy. Ta chọn $h_1(d) = d^5 + d^2 + 1$, $h_2(d) = d^5 + d^4 + d^3 + d^2 + 1$ và các dãy a và b lần lượt được tạo bởi đa thức $h_1(d)$ và $h_2(d)$. Ta có:

a = (11100011011101010000100101100 11) ,

b = (10001010110100001100100111110 11) .

Tập hợp các dãy Gold tương ứng có chu kì $N = 2^5 - 1 = 31$ là $N + 2 = 33$ dãy được biểu diễn trong bảng 1.2.

Bảng 1.2 Các dãy Gold có chu kì N = 31, kích thước M = 33

STT	*G(a,b)*	Dãy Gold
1	*a*	11100011011101010000100101100 11
2	*b*	10001010110100001100100111111011
3	$a \oplus b$	01101001101001011100000010010 00
4	$a \oplus Tb$	00100110000111010110110110011 10
5	$a \oplus T^2b$	00000001110000010011101100011 01
6	$a \oplus T^3b$	10010010001011110001000001011 00
7	$a \oplus T^4b$	01011011110110000000010111111 00
8	$a \oplus T^5b$	00111111001000111000111100101 00
9	$a \oplus T^6b$	00001101010111100100101001000 00
10	$a \oplus T^7b$	00010100011000001010100011110 10
11	$a \oplus T^8b$	00011000111111111101100110101 11
12	$a \oplus T^9b$	10011110101100000110000100000 01
13	$a \oplus T^{10}b$	11011101100101111011110101010 10
14	$a \oplus T^{11}b$	01111100000001000101001101111 11
15	$a \oplus T^{12}b$	10101100110011011010010001101 01
16	$a \oplus T^{13}b$	11000100101010010101111111100 00
17	$a \oplus T^{14}b$	01110000100110110010001000100 10
18	$a \oplus T^{15}b$	00101010100000100001110011000 11
19	$a \oplus T^{16}b$	10000111100011101000001110110 11

STT	*G(a,b)*	Dãy Gold
20	$a \oplus T^{17}b$	11010001000010001100110000000111
21	$a \oplus T^{18}b$	11111010010010111111010111101001
22	$a \oplus T^{19}b$	11101111111010100111110000011110
23	$a \oplus T^{20}b$	01100101001110101011000011100101
24	$a \oplus T^{21}b$	10100000010100101101010010011000
25	$a \oplus T^{22}b$	01000010111001101110011101001l0
26	$a \oplus T^{23}b$	00110011101111001111111100111001
27	$a \oplus T^{24}b$	10001011000100011111001011101l0
28	$a \oplus T^{25}b$	01010111010001110111010010100l1
29	$a \oplus T^{26}b$	10111001011011000011011110000l0
30	$a \oplus T^{27}b$	01001110011110011001011000010l1
31	$a \oplus T^{28}b$	10110101111100110100011011011l1
32	$a \oplus T^{29}b$	11001000001101100010111010111l1
33	$a \oplus T^{30}b$	11110110110101001001101010001l0

Với hai dãy bất kỳ *a,b*, *G(a,b)* có các giá trị hàm tương quan sẽ nhận các giá trị [18]:

$$R_{a'b'}(\tau) = \begin{cases} -1, -1 - 2^{\frac{m+1}{2}}, -1 + 2^{\frac{m+1}{2}}, & (m \bmod 2) \neq 0 \\ -1, -1 - 2^{\frac{m+2}{2}}, -1 + 2^{\frac{m+2}{2}}, & (m \bmod 4) = 0 \end{cases} \quad (1.9)$$

1.3.2 Bộ tạo dãy tựa Gold

Dãy tựa Gold (Gold-like) [7] được định nghĩa như sau: cho *m* là một số chẵn và *q* là một số nguyên sao cho gcd(*q*, 2^m-1) = 3. Gọi *u* là một dãy m có chu kỳ $N = 2^m - 1$ tạo nên bởi *h(d)* và $b^{(k)}$ với *k* = 0, 1, 2, là tập hợp các dãy nhận được bằng

cách lấy mẫu T^ka theo q. Vận dụng tính chất dịch và cộng của dãy m ta thấy $b^{(k)}$ có chu kỳ $N' = N/3$ và được tạo nên bởi đa thức $h'(d)$ mà nghiệm của nó là lũy thừa bậc q của nghiệm của $h(d)$.

Lớp các dãy tựa Gold được tạo bởi:

$$H(a,b) = \{a, a \oplus b^{(0)}, a \oplus T^1 b^{(0)}, \ldots, a \oplus T^{N'-1} b^{(0)},$$
$$a \oplus b^{(1)}, a \oplus T^1 b^{(1)}, \ldots, a \oplus T^{N'-1} b^{(1)}, \quad (1.9)$$
$$a \oplus b^{(2)}, a \oplus T^1 b^{(2)}, \ldots, a \oplus T^{N'-1} b^{(2)}\}$$

Rõ ràng là tập hợp $H(a,b)$ chứa $(N + 1) = 2^m$ dãy có chu kỳ N.

Hàm tương quan của các dãy này có thế nhận các giá trị như sau [18]:

$$R_{a'b'}(\tau) = -1, -1 - 2^{\frac{m+2}{2}}, -1 + 2^{\frac{m+2}{2}}, -1 - 2^{\frac{m}{2}}, -1 + 2^{\frac{m}{2}} . \quad (1.10)$$

Khoảng tuyến tính của dãy tựa Gold được cho bởi $L = deg[h(d)] = m + m = 2m$.

<u>Ví dụ 2.4:</u> Chọn $m = 4$, $q = 9$, ta có gcd(q, $2^m - 1$) = gcd(9, 15) = 3. Ta có thể tạo một dãy m có độ dài $N = 2^4 - 1 = 15$ như sau:

$a = \{a_n\}$ = (000100110101111) .

Lấy mẫu dãy $T^k a$ với bước lấy mẫu $q = 9$, k = 0, 1, 2, ta được:

$b^{(0)}$ = (011110111101111) ,

$b^{(1)}$ = (000110001100011) ,

$b^{(2)}$ = (010100101001010) .

Tập các dãy tựa Gold $H(a,b)$ có kích thước $M = 16$ được biểu diễn trong bảng 1.3.

Với: $a' = a \oplus T^3 b^{(0)}$ = (110011011010100) ,

$b' = a \oplus T^1 b^{(2)}$ = (101101100111011) .

Ta có:

$\{R_{a'b'}(\tau)\}$ = {-9,3,3,-1,-5,7,3,-5,-1,3,7,-5,-5,7,3}.

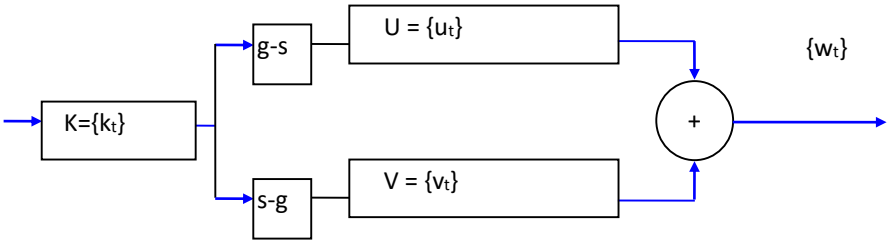
Bảng 1.3 Dãy tựa Gold có chu kì N = 15, kích thước M = 16

STT	*H(a,b)*	Dãy tựa Gold
1	*a*	000100110101111
2	$a \oplus T^0 b^{(0)}$	011010001000000
3	$a \oplus T^1 b^{(0)}$	111001001110001
4	$a \oplus T^2 b^{(0)}$	111111000010010
5	$a \oplus T^3 b^{(0)}$	110011011010100
6	$a \oplus T^4 b^{(0)}$	101011101011000
7	$a \oplus T^0 b^{(1)}$	000010111001100
8	$a \oplus T^1 b^{(1)}$	001000101101001
9	$a \oplus T^2 b^{(1)}$	011100000100011
10	$a \oplus T^3 b^{(1)}$	110101010110111
11	$a \oplus T^4 b^{(1)}$	100111110011110
12	$a \oplus T^0 b^{(2)}$	010000011100101
13	$a \oplus T^1 b^{(2)}$	101101100111011
14	$a \oplus T^2 b^{(2)}$	010110010000110
15	$a \oplus T^3 b^{(2)}$	100001111111101
16	$a \oplus T^4 b^{(2)}$	001110100001010

1.3.3 Bộ tạo dãy luân phiên

Bộ tạo dãy luân phiên (The Alternating Step Generator) là sự kết hợp khéo léo giữa hai bộ tạo dãy Stop- Go thông qua dãy điều khiển D' Bruijn [27]. Bộ tạo này đã phát huy được các đặc tính tốt của các dãy thành phần: các m-dãy và dãy

D' Bruijn, đồng thời nó cũng tích hợp được tính miễn dịch tương quan, một trong các yêu cầu quan trọng của các kiểu tạo khoá thuật toán hiện nay.



Hình 1.5 Mô hình bộ tạo dãy luân phiên

Giả sử: $K=\{k_t\}_{t\geq 0}$ là dãy D' Bruijn bậc k; $U=\{u_t\}$ và $V=\{v_t\}$ là hai m-dãy bậc tương ứng L, M nguyên tố cùng nhau.

Khi đó đầu ra $W=\{w_t\}_{t\geq 1}$ của bộ tạo dãy luân phiên như trong hình 1.5 sẽ được cho bởi công thức sau:

$$w_t = u_{f(t)} \oplus v_{f^*(t)},\ t\geq 1 \qquad (1.11)$$

trong đó $f(t)= \sum_{s=0}^{t-1} k_s$, $f^*(t)= t- f(t)$.

Các tính chất của bộ tạo dãy luân phiên

Tính chất 1 (Chu kỳ và Độ phức tạp tuyến tính)

Giả sử:

a) K là dãy D' Bruijn chu kỳ 2^k;

b) U, V là các m-dãy chu kỳ tương ứng p, q với các đa thức đặc trưng $p(x)$, $q(x)$ có bậc L, M nguyên tố cùng nhau.

Khi đó, chu kỳ T và độ phức tạp tuyến tính λ của dãy W sẽ được cho bởi công thức sau:

$$T= K.p.q; \qquad (1.12)$$

$(L+M)\ 2^{k-1} < \lambda \leq (L+M)\ 2^k$.

Tính chất 2 (Phân bố tần số các bộ r-tupe)

Với các giả thiết như trong Tính chất 1, và giả sử $d\leq \min\{L, M\}$.

Đặt $\sigma=(\sigma_0, \sigma_1,..., \sigma_{d-1})\in\{0,1\}^d$. Khi đó:

$$\frac{1}{T}\text{card}\{ t\in Z_T : w_{t+i}= \sigma_i, \forall i \in Z_d\} = \frac{1}{2^d} + o\left(\frac{1}{2^{L-d}}\right) + o\left(\frac{1}{2^{M-d}}\right) . \quad (1.13)$$

Tính chất 3 (Tính chất tương quan)

Cũng với các giả thiết như trên, hàm tự tương quan của dãy luân phiên sẽ được ước lượng bởi công thức sau:

$$C(\tau) = \frac{t_0.pq - t_1.p - t_2.q + t_3}{2^k.pq} \quad (1.14)$$

trong đó t_i, i=0..3, là chỉ số trùng giữa các pha thứ 0 và thứ τ của dãy, đồng thời chúng thoả mãn các ràng buộc

$0\le t_0, t_1, t_2, t_3\le 2^k$; $t_0+ t_1+ t_2+ t_3= 2^k$.

Chứng minh các Tính chất 2, 3 có thể tham khảo trong [27].

Tính chất 4 (Lực lượng của bộ tạo dãy)

Giả sử k, L, M là các số cố định cho trước thoả mãn các yêu cầu đã nêu trong mô hình bộ tạo. Khi đó số lượng các dãy có thể tạo được qua mô hình bộ tạo dãy luân phiên ứng với các tham số này là:

$$K_w = 2^{2^{k-1}-k} . \frac{\Phi(2^L-1)}{L} . \frac{\Phi(2^M-1)}{M} . \quad (1.15)$$

Nhận xét:

Các tính chất lý thuyết trên đây cho thấy dãy luân phiên có độ phức tạp tuyến tính rất cao, có hàm tự tương quan đủ nhỏ, và mô hình bộ tạo có tính miễn dịch tương quan, do đó có thể được dùng tạo khoá trong mật mã khi chọn các tham số thích hợp.

Từ các nghiên cứu về khả năng miễn dịch với các tấn công phân tích bộ tạo [19], ta thấy rằng với các tấn công mạnh nhất hiện nay và với các giả thiết rộng rãi nhất đối với thám mã, bộ tạo dãy luân phiên hoàn toàn miễn dịch với các kiểu tấn

công đó. Thông qua các khảo sát này, ta có thể làm chủ các thông số, các giới hạn đảm bảo an toàn cho bộ tạo dãy trong thực tế sử dụng.

1.3.4 Dãy lồng ghép và dãy phi tuyến lồng ghép

Dãy lồng ghép (Interleaved sequence) là một kiến trúc riêng do nhóm nghiên cứu của TS. Lê Chí Quỳnh đề xuất, với mục tiêu xây dựng một dãy giả ngẫu nhiên từ một m-dãy ban đầu với các tham số lồng ghép được lựa chọn theo nguyên tắc riêng. Các tác giả đã xây dựng các dãy lồng ghép có độ phức tạp cao và tính chất tốt về phân bố và tương quan [49] [50]. Các công bố của nhóm về dãy lồng ghép đã được một số tác giả nước ngoài quan tâm, công nhận và trích dẫn như một "kiến trúc dãy kiểu Việt Nam" [23]. Các chi tiết về kiến trúc dãy lồng ghép sẽ được phân tích trong phần 2.1 và 2.2 của luận án.

Có một số nghiên cứu ở nước ngoài cũng sử dụng khái niệm tiếng Anh "interleaved sequence"[33][59], về bản chất cũng sử dụng giải pháp đan xen các bit từ một hoặcnhiều dãy, song cách tiếp cận có nhiều khác biệt so với kiến trúc dãy được đề cập trong luận án này.

Kế tiếp các nghiên cứu ban đầu về dãy lồng ghép, TS Lê Minh Hiếu tiếp tục nghiên cứu về dãy lồng ghép tam phân và dãy phi tuyến lồng ghép [30]. Tiếp đó, tiến sỹ Bùi Lai An đã phát triển dãy lồng ghép đa cấp, đa chiều 2012 [1].

Dãy phi tuyến lồng ghép là một phát triển của dãy lồng ghép, trong đó sử dụng 2 m-dãy ban đầu, song kết hợp với nhau theo phương pháp đặc trưng của dãy lồng ghép với mục tiêu đưa ra dãy đầu ra có tính phi tuyến cao hơn so với dãy lồng ghép. sử dụng tham số của hai dãy đầu vào có cùng bậc nhưng sinh bởi hai đa thức sinh khác nhau $f(x)$ và $g(x)$. Theo phương pháp xây dựng dãy lồng ghép, hai dãy đầu vào nói trên sẽ sinh ra hai dãy lồng ghép với các dãy con sinh bởi đa thức con $f_1(x)$ và $g_1(x)$ tương ứng. Nếu ta thay thế thứ tự lồng ghép của dãy con thứ nhất bằng thứ tự lồng ghép của dãy con thứ hai, dãy đầu ra sẽ là kết quả lồng ghép kết hợp của hai dãy đầu vào. Các tác giả đã chứng minh rằng dãy lồng ghép kết hợp này có tính chất phi tuyến tốt hơn dãy lồng ghép ban đầu, do đó nó được gọi là dãy

phi tuyến lồng ghép. Tính chất "phi tuyến" được đề cập ở đây có nghĩa là dãy mới tạo ra có độ phức tạp tuyến tính lớn hơn nhiều so với dãy lồng ghép ban đầu, tuy nhiên nếu xét từng đoạn kích thước nhỏ của dãy mới thì ta vẫn có thể nhận ra sự phụ thuộc tuyến tính. Trong phần 2.3 của luận án sẽ đề cập các chi tiết cụ thể về dãy phi tuyến lồng ghép.

1.4 Kết luận chương I

Trong chương này tác giả đã trình bày một cách đơn giản và rõ ràng về trường Galois và mở rộng trường Galois bằng cách chỉ sử dụng các khái niệm toán học đơn giản. Việc xây dựng m-dãy từ trường Galois cũng được trình bày cụ thể với hai phương pháp xây dựng theo Galois và Fibonacy, cũng như sự liên hệ giữa hai phương pháp này. Trong chương cũng phân tích sự khác biệt của trường Galois trong trường hợp chung của đặc số p không phải là giá trị p=2, nhất là khi xây dựng m-dãy trên GF(p^n)

Về ứng dụng của m-dãy, tác giả đã phân tích một số ứng dụng thông dụng của m-dãy và đi sâu phân tích về ứng dụng m-dãy trong các hệ mã dòng. Một số hệ mã dòng thông dụng cũng được giới thiệu và phân tích, cùng với một số giới thiệu về dãy lồng ghép và dãy phi tuyến lồng ghép.

CHƯƠNG 2 : CÁC PHƯƠNG PHÁP SINH DÃY PHI TUYẾN LỒNG GHÉP DỰA TRÊN M-DÃY

Hướng nghiên cứu về dãy lồng ghép và dãy phi tuyến lồng ghép được nhóm nghiên cứu của TS. Lê Chí Quỳnh phát triển từ những năm 1980 [30] [50]. Trong chương này tác giả trình bày chi tiết về kiến trúc dãy lồng ghép, dãy phi tuyến lồng ghép và các phương pháp sinh dãy lồng ghép, bao gồm các phương pháp đã được phát triển [J1] và phương pháp do tác giả đề xuất trong công bố [J3].

2.1. Kiến trúc dãy lồng ghép

2.1.1 Biểu diễn dãy bằng biến đổi d

Về mặt lý thuyết, các dãy được biểu diễn theo cơ sở α, ví dụ như: Biểu diễn hàm vết (Trace function) đã được sử dụng rộng rãi để phân tích cấu trúc lồng ghép [24][39]. Trong phần này, ta sẽ chỉ ra rằng, cách biểu diễn đa thức không chỉ hiệu quả mà còn có một số lợi thế nhất định. Để chứng minh, ta chọn trường hợp khi độ dài của chuỗi $L \neq q^n - 1$, với q là một số nguyên tố trong đó hàm vết không xác định và do đó không thể áp dụng được lý thuyết hàm vết [40] [42]. Tuy nhiên, trong trường hợp này cách biểu diễn đa thức vẫn có thể áp dụng được [15]. Công cụ toán học để chuyển đổi các chuỗi thành đa thức là biến đổi d (d - Transform). Trong luận án này, biến đổi *d* sẽ được sử dụng để phân tích các dãy trên trường GF(p^n) .

Biểu diễn biến đổi *d* của một chuỗi $\{b_n\}$ trên GF(p^n) được ký hiệu là $D[b_n]$ (hoặc F) và xác định bởi công thức

$$\mathrm{D}[b_n] = F = \sum_{i=0}^{m} b_i d^i \,, b_i \in \{\mathrm{GF}(p)\} \qquad (2.1)$$

Ví dụ l: Đặt $\{b_n\}$={2 2 0 2 1 1 0 1}, biểu diễn biến đổi d của $\{b_n\}$ là

$$D[b_n] = 2 + 2d + 2d^3 + d^4 + d^5 + d^7.$$

Biến đổi ngược của *D* là $D^{-1} = \{b_n\}$.

Do đó, biến đổi d của chuỗi sẽ có dạng đa thức theo biến d trên GF (p) và điều này đã được sử dụng như một quy ước trong việc phân tích tín hiệu của các hệ thống truyền dữ liệu và CDMA [15][30].

Một số tính chất của đa thức bậc n trên trường GF(p) (với p là số nguyên tố) sẽ được tóm tắt dưới đây:

- Số mũ của đa thức $Q(d)$ là giá trị nhỏ nhất của n sao cho $Q(d)$ chia hết cho $1-d^n$, tức là, $(1-d^n) / Q(d)$ là một đa thức có bậc hữu hạn.
- Một đa thức $Q(d)$ được gọi là bất khả quy (irreducible) nếu không tìm được đa thức có bậc lớn hơn 1 mà chia hết được $Q(d)$.
- Hai đa thức gọi là nguyên tố cùng nhau khi không tìm được đa thức có bậc lớn hơn 1 mà chia hết được cho cả hai đa thức ban đầu.
- Một đa thức bất khả quy (irreducible) bậc m là đa thức nguyên thủy (primitive – còn gọi là đa thức nguyên tố) hoặc đa thức có số mũ cực đại nếu số mũ của nó là $p^m - 1$.

Cho một đa thức $Q(d)$ bậc m, đa thức đối ứng của nó là $d^mQ(1/d)$ và ta biết rằng đa thức đối ứng của một đa thức bất khả quy cũng là đa thức bất khả quy; đồng thời đa thức đối ứng của một đa thức nguyên thủy cũng là đa thức nguyên thủy.

Biến đổi d của một chuỗi tuần hoàn có dạng $R(d) / (1-d^l)$, trong đó l là chu kỳ của chuỗi và $R(d)$ là một đa thức bậc nhỏ hơn l trong d trên trường GF(p). Nói chung, có thể chỉ ra rằng, biến đổi d của chuỗi tuần hoàn theo thời gian có dạng $p(d)/Q(d)$ trong đó cả $p(d)$ và $Q(d)$ đều là các đa thức trên trường Galois. Nếu $p(d)$ và $Q(d)$ là nguyên tố cùng nhau, chu kỳ của chuỗi tuần hoàn được biểu thị bằng $p(d)/Q(d)$ chính là số mũ của $Q(d)$.

Biến đổi d của chuỗi $\{b_n\}$ sinh ra từ bộ thanh ghi dịch phản hồi tuyến tính (LFSR) được xác định bởi công thức:

$$b(d) = \frac{S(d)}{g(d)} \qquad (2.2)$$

Trong đó $g(d)$ có bậc n là đa thức sinh của LFSR và $S(d)$ có bậc nhỏ hơn n xác định giá trị ban đầu của thanh ghi tương ứng với một phiên bản dịch bit vòng quanh của $\{b_n\}$. Khi $g(d)$ là đa thức nguyên thủy, chuỗi sinh ra từ LFSR được gọi là m-dãy và ta có p^n-l giá trị $S(d)$ là các trạng thái ban đầu có thể có của LFSR đó.

Các cặp biến đổi d được đưa ra trong [30]. Quy trình xây dựng dựa trên biến đổi d để tạo các dãy nhị phân phi tuyến lồng ghép được đưa ra trong [J1]. Các tác giả đã mở rộng kết quả của đối với trường hợp dãy được dùng là p-phân , trong đó chỉ ra cách áp dụng các quy trình cho các trường hợp tam phân.

2.1.2 Kiến trúc dãy lồng ghép

Với một m-dãy $\{b_i\}$ được sinh bởi đa thức sinh $f(x)$ trên trường GF(p^n). Trong trường hợp $n=m.l$, từ các giá trị $L = p^n$-1, $N = p^m$-1 ta tính ra bước lồng ghép

$$T = \frac{L}{N}. \tag{2.3}$$

Ta xây dựng lên dãy lồng ghép $\{b_i\}$ bằng cách lồng ghép (T-1) dãy con thành phần, mỗi dãy có độ dài $N = q^m$-l. Các dãy con có được bằng cách áp dụng phép phân rã theo bước (decimation) trên dãy $\{b_i\}$ với bước nhảy bằng T

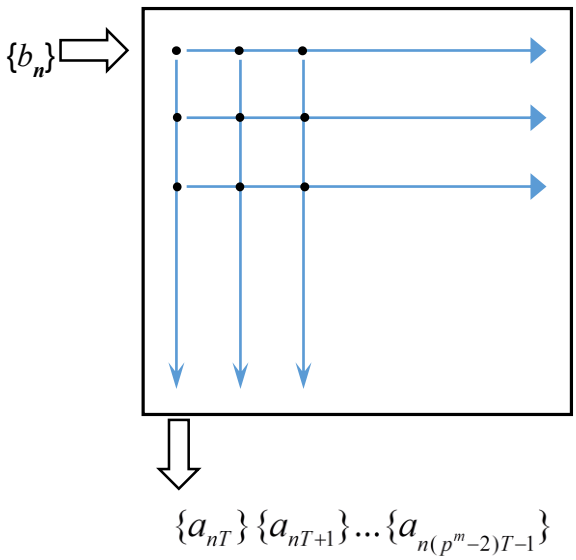
Khi phép phân rã theo bước bắt đầu từ bit đầu tiên của $\{b_i\}$ (ô giá trị đầu tiên của $\{b_i\}$), ta thu được dãy con:

$$\{a_{nT}\} = \{a_0, a_T, \dots, a_{(p^m-2)T}\}. \tag{2.4}$$

Tương tự như vậy, với vị trí bắt đầu nhảy bước là t, ta thu được dãy con:

$$\{a_{nT+t}\} = \{a_t, a_{T+t}, \dots, a_{(p^m-2)T+t}\}. \tag{2.5}$$

Do đó, xét trên miền thời gian, các dãy con này (sắp xếp theo cột) có thể được coi là ghép kênh theo bước thời gian T $\{a_{nT}\}\{a_{nT+1}\}...\{a_{n(p^m-2)T-1}\}$ để đặt vào T khe thời gian như trong sơ đồ dưới đây:



Hình 2.1 Kiến trúc dãy lồng ghép

Ví dụ 2.1: Cho $n = 4$, $m = 2$ và α là phần tử sinh của trường GF(3^4) với đa thức sinh là đa thức nguyên thủy $g(d) = 1 + d^3 + 2d^4$ trên trường GF(3). Ký hiệu $\{b_n\}$ là m-dãy sinh bởi $g(d)$. Ta có:

$\{b_n\}$ = {1 0 0 0 1 0 0 2 1 0 1 1 1 2 0 0 2 2 0 1 0 2 2 1 1 0 1 0 1 2 1 2 2 1 2 0 1
2 2 2 2 0 0 0 2 0 0 1 2 0 2 2 2 1 0 0 1 1 0 2 0 1 1 2 2 0 2 0 2 1 2 1 1 2
1 0 2 1 1 1} .

Áp dụng phép nhảy bước trên dãy $\{b_n\}$ với bước nhảy $T = 10$ ta có được các dãy con $\{a_n\} = \{b_{n*10}\}$ và sắp xếp lại các dãy con đó thành ma trận như sau:

$$M = \begin{vmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 2 & 1 & 0 \\ 1 & 1 & 1 & 2 & 0 & 0 & 2 & 2 & 0 & 1 \\ 0 & 2 & 2 & 1 & 1 & 0 & 1 & 0 & 1 & 2 \\ 1 & 2 & 2 & 1 & 2 & 0 & 1 & 2 & 2 & 2 \\ 2 & 0 & 0 & 0 & 2 & 0 & 0 & 1 & 2 & 0 \\ 2 & 2 & 2 & 1 & 0 & 0 & 1 & 1 & 0 & 2 \\ 0 & 1 & 1 & 2 & 2 & 0 & 2 & 0 & 2 & 1 \\ 2 & 1 & 1 & 2 & 1 & 0 & 2 & 1 & 1 & 1 \end{vmatrix}$$

So sánh các cột của ma trận M với giá trị biểu diễn bit trong bảng 2.1, ta có được thứ tự lồng ghép I_P^T (cũng là danh sách các bước dịch của dãy con) như sau:

$$I_P^T = \{4, 6, 6, 2, 5, \text{“}\infty\text{”}, 2, 0, 5, 6\}. \quad (2.6)$$

Trong đó giá trị ∞ biểu diễn vị trí của dãy con chứa toàn phần tử 0 .

2.1.3 Giải pháp chung để xây dựng dãy lồng ghép

Phân tích các đặc điểm của các dãy con, ta thấy rằng các dãy con đều là dịch pha từ một dãy toàn chu kỳ sinh ra bởi đa thức $f_1(x)$ trên trường GF(p^m). Giá trị đa thức sinh $f_1(x)$ được tính toán bằng cách sử dụng biến đổi d trên dãy đầu ra thu được từ ma trận lồng ghép theo công thức:

$$f_1(d) = gcd\left(a_n, d^{p^n-1} - 1\right) \quad (2.7)$$

Ta cũng có thể áp dụng thuật toán Belekamp-Massey trình bày trong phần 3.1.2 để tìm đa thức sinh $f_1(x)$. Trong các phần mềm mô phỏng tác giả sử dụng phương pháp này.

Để có được tất cả các dãy con, ta chỉ cần xác định tập các bước dịch pha tương ứng với mỗi dãy con gọi là tập thứ tự lồng ghép, ký hiệu là I_P. Từ các bước dịch pha này, ta có thể xây dựng dược dãy đầu ra mà không cần thực hiện tính toán dãy đầu vào.

2.2. Các phương pháp để xây dựng dãy lồng ghép p-phân

Công việc chính cần thực hiện để xây dựng dãy lồng ghép là xác định tập các bước dịch pha I_P. Để thực hiện điều này có thể sử dụng 3 phương pháp: Mở rộng dãy sử dụng biến đổi d, phân rã qua hàm vết α, hoặc phương pháp tính trực tiếp các hàng đầu tiên của ma trận lồng ghép[60].

2.2.1 Phương pháp mở rộng dãy sử dụng biến đổi d

Cho $\{b_n\}$ là một m-dãy sinh bởi đa thức sinh $g(d)$ có bậc n, chu kỳ L thỏa mãn các điều kiện:

$L = p^n - 1 = p^{l*m} - 1 = T.(p^m - 1) = T.N$, $n = l.m$, $T = (p^n - 1)/(p^m - 1)$.

Gọi $b(d)$ là biến đổi d của $\{b_n\}$, theo (2.2) ta có:

$$b(d) = \frac{S(d)}{g(d)} \tag{2.8}$$

trong đó $S(d)$ là trạng thái khởi đầu của m-dãy.

Ta luôn có thể biểu diễn $b(d)$ theo dạng:

$$b(d) = \sum_{i=0}^{T-1} d^i F_i(d^T) \tag{2.9}$$

trong đó $F_i(d)$ là dãy con được sinh từ đa thức sinh $g_1(d)$ có bậc m, chu kỳ N và được xác định trong biến đổi d theo công thức:

$$F_i(d) = \frac{S_i(d)}{g_1(d)}, i = 0,1 \dots, T-1 \tag{2.10}$$

với $S_i(d)$ là trạng thái khởi đầu của dãy con và $g_1(d)$ là đa thức sinh tương ứng của dãy đó.

Điều này được suy ra trực tiếp từ các thuộc tính của biến đổi d vì $\{b_n\}$ có thể được xây dựng bằng cách lồng ghép các pha T của $\{F_n\}$. Các pha cụ thể của $\{F_n\}$ trong đó thứ tự lồng ghép có thể được xác định thông qua 3 bước.

Bước 1: mở rộng $F_i(d)$ lên T lần (chèn T-1 số 0 vào giữa hai bit liên tiếp của $F_i(d)$), trong biến đổi d, nó tương đương với việc thay thế d bằng d^T trong công thức:

$$F_i(d^T) = \frac{S_i(d^T)}{g_1(d^T)} \ . \tag{2.11}$$

Bước 2: Biểu diễn theo biến đổi d của $\{b_n\}$ theo cách để xen kẽ các $F_i(d)$, (hoặc chèn T pha khác nhau của $F_i(d)$ để tạo thành $b(d)$):

$$b(d) = \sum_{i=0}^{T-1} d^i F_i(d^T) = \sum_{i=0}^{T-1} d^i \frac{S_i(d^T)}{g_1(d^T)} . \tag{2.12}$$

Sau đó đặt tử số trong (2.8) thành:

$$G(d) = \sum_{i=0}^{T-1} d^i S_i\left(d^T\right) . \tag{2.13}$$

Thay (2.12), (2.13) vào (2.8) ta có:

$$\mathrm{G(d)} = \frac{S(d).g_1(d^T)}{g(d)}. \quad (2.14)$$

Bước 3:

- Đặt $d^T = D$.

- Tìm các bước dịch pha $F_i(D) = \frac{S_i(D)}{g_1(D)}$.

Sau đó nhóm lại biến đổi d của $b(d)$ thành:

$$b(d) = \sum_{i=0}^{s-1} d^i F_i(D) . \quad (2.15)$$

So sánh phần $F_i(D)$ với bảng biến đổi d của từng phần của dãy, ta có thể xây dựng lên toàn bộ các bậc lồng ghép I_P^S.

Toàn bộ quy trình trên sẽ được minh họa rõ trong ví dụ sau:

Ví dụ 2.2: Cho $g(d) = 1 + d^3 + 2d^4$ trên GF(3^4) với các tham số :

$n = 4$, $m = 2$, $l=2$, $L = 80$, $N = 8$, $T = 80/8 = 10$.

$g_1(d) = 1 + d + 2d^2$.

$$F_1(d^S) = \frac{S_i(d^S)}{g_l(d^S)} = \frac{S_i(d^{10})}{g_l(d^{10})} \quad . \quad (2.16)$$

$$G(d) = \frac{S(d)\cdot g_1(d^{10})}{g(d)}. \quad (2.17)$$

Vì ta không cần phải quan tâm tới một giá trị khởi đầu cụ thể của $b(d)$, ta hoàn toàn có thể giả sử là $S(d) = 1$ mà không làm mất tính tổng quát. Khi này ta có:

$$G(d) = \frac{g_1(d^{10})}{g(d)} = \frac{1+d^{10}+2d^{20}}{1+d^3+2d^4}. \quad (2.18)$$

$$G(d) = d^{16} + d^{15} + d^{14} + d^{13} + d^{12} + d^{10} + d^9 + d^8 + d^7 + d^6 + d^4 + 2d^3 + 1 .$$

$$b(d) = \frac{G(d)}{g_1(d^{10})} = \frac{\mathrm{d}^{16} + \mathrm{d}^{15} + \mathrm{d}^{14} + \mathrm{d}^{13} + \mathrm{d}^{12} + \mathrm{d}^{10} + \mathrm{d}^9 + \mathrm{d}^8 + \mathrm{d}^7 + \mathrm{d}^6 + \mathrm{d}^4 + 2\mathrm{d}^3 + 1}{1+d^{10}+2d^{20}} \quad (2.19)$$

đặt $d^{10} = D$ và sắp xếp lại $b(d)$ theo nhóm của số mũ d như sau:

$$b(d)= \frac{(1+\mathrm{D})+(2\mathrm{D})\mathrm{d}^2+(2+\mathrm{D})\mathrm{d}^3+(1+\mathrm{D})\mathrm{d}^4+(\mathrm{D})\mathrm{d}^5+(1+\mathrm{D})\mathrm{d}^6+(1)\mathrm{d}^7+(1)\,\mathrm{d}^8+(2)\mathrm{d}^9}{1+D+2D^2} \quad (2.20)$$

So sánh các phần tử của (2.20) với nội dung bảng 2.1, ta có được bậc lồng ghép của $\{b_n\}$ là:

IP = {5, "∞", 2, 0, 5, 6, 5, 7, 7, 3} .

Ta cũng thấy rằng giá trị của I_P^T ở đây hoàn toàn trùng khớp với kết quả thu được trong phần 2.1.2.

Bảng 2.1 Biến đổi d của m-dãy

$g_1(d)$	Dãy con	Dạng bit	Chỉ số pha	$S_i(d)$	$S(D)$
$1 + d + 2d^2$	T^0W	2 2 0 2 1 1 0 1	0	$2 + d$	$2 + D$
	T^1W	2 0 2 1 1 0 1 2	1	$2 + 2d$	$2 + 2D$
	T^2W	0 2 1 1 0 1 2 2	2	$2d$	$2D$
	T^3W	2 1 1 0 1 2 2 0	3	2	2
	T^4W	1 1 0 1 2 2 0 2	4	$1 + 2d$	$1 + 2D$
	T^5W	1 0 1 2 2 0 2 1	5	$1 + d$	$1 + D$
	T^6W	0 1 2 2 0 2 1 1	6	d	D
	T^7W	1 2 2 0 2 1 1 0	7	1	1
$1+ 2d + 2d^2$	T^0Z	2 1 0 1 1 2 0 2	0	$2 + 2d$	$2 + 2D$
	T^1Z	1 0 1 1 2 0 2 2	1	$1 + 2d$	$1 + 2D$
	T^2Z	0 1 1 2 0 2 2 1	2	d	D
	T^3Z	1 1 2 0 2 2 1 0	3	1	1
	T^4Z	1 2 0 2 2 1 0 1	4	$1 + d$	$1 + D$
	T^5Z	2 0 2 2 1 0 1 1	5	$2 + d$	$2 + D$
	T^6Z	0 2 2 1 0 1 1 2	6	$2d$	$2D$
	T^7Z	2 2 1 0 1 1 2 0	7	2	2

2.2.2. Phương pháp phân rã m-dãy sử dụng hàm vết

Trong [42] mối quan hệ giữa lũy thừa của phần tử sinh α của dãy và biến đổi d đã được giải thích rất rõ ràng.

Vì cả hai biểu diễn của dãy lồng ghép (thông qua giữa lũy thừa của phần tử sinh α và thông qua biến đổi *d*) là tương đương với nhau, nên tập thứ tự lồng ghép cũng có thể được xác định bằng hai phương pháp.

Trong trường hợp biểu diễn dãy lồng ghép theo lũy thừa của α (còn gọi là hàm vết – Trace function), bậc lồng ghép được xác định như sau

Với *m, n* là các số nguyên dương mà *m* chia hết *n*, α là phần tử sinh của trường hữu hạn GF(p^n) và:

$$T = L/N = (p^n\text{-}1)/(p^m\text{-}1). \tag{2.21}$$

Khi đó, hàm vết $Tr_m^n(x)$ sẽ ánh xạ các phần tử của GF(p^n) vào GF(p^m) theo quan hệ sau [38]:

$$Tr_m^n(x) = \sum_{k=0}^{\frac{n}{m}-1} x^{p^{mk}} . \tag{2.22}$$

Bậc lồng ghép sẽ là: $I_P = I_P^0, I_P^1, \dots, I_P^{T-1}$.

Với các giá trị được tính theo công thức:

$$I_P^j = \begin{cases} i \quad if \quad Tr_m^n(\alpha^j) = \alpha^{iT} & i = 0,1,\dots,p^m - 1 \\ \infty \quad if \quad Tr_m^n(\alpha^j) = 0 & j = 0,1,\dots,T-1 \end{cases} \tag{2.23}$$

Ví dụ 2.3:

Xét hàm vết ánh xạ từ GF(3^4) vào GF(3^2) với đa thức sinh nguyên thủy:

$f(x) = x^4 + x + 2$

và các tham số được chọn là : $n = 4$, $m=2$, $L = 3^4\text{-}1 = 80$; $m = 2$, $N = 3^2\text{-}1 = 8$ và $T = L/N = 10$.

Tính toán giá trị của hàm vết ánh xạ từ GF(3^4) vào GF(3^2) ta có :

$$Tr_m^n(\alpha) = \sum_{k=0}^{n/m-1} \alpha^{3^{2k}} = \alpha + \alpha^9 . \quad (2.24)$$

Bảng các giá trị của α^{iT} như sau:

$i = 0 \Rightarrow \alpha^0 = 1$.

$i = 1 \Rightarrow \alpha^{10} = 1 + 2\alpha + \alpha^2 + \alpha^3$.

$i = 2 \Rightarrow \alpha^{20} = \alpha + 2\alpha^2 + 2\alpha^3$.

$i = 3 \Rightarrow \alpha^{30} = 1 + \alpha + 2\alpha^2 + 2\alpha^3$.

$i = 4 \Rightarrow \alpha^{40} = 2$.

$i = 5 \Rightarrow \alpha^{50} = 2 + \alpha + 2\alpha^2 + 2\alpha^3$.

$i = 6 \Rightarrow \alpha^{60} = 2\alpha + \alpha^2 + \alpha^3$.

$i = 7 \Rightarrow \alpha^{70} = 2 + 2\alpha + \alpha^2 + \alpha^3$.

- Cho giá trị j chạy từ 0 tới T-1 ta có:

$j = 0 \Rightarrow Tr(\alpha^0) = Tr(1) = 1 + 1 = 2 = \alpha^{40} \Rightarrow I_P^0 = 4$.

$j = 1 \Rightarrow Tr(\alpha^{10}) = \alpha + \alpha^9 = 2\alpha + \alpha^2 + \alpha^3 = \alpha^{60} \Rightarrow I_P^1 = 6$.

$j = 2 \Rightarrow Tr(\alpha^{20}) = \alpha^2 + \alpha^{18} = 2\alpha + \alpha^2 + \alpha^3 = \alpha^{60} \Rightarrow I_P^2 = 6$.

$j = 3 \Rightarrow Tr(\alpha^{30}) = \alpha^3 + \alpha^{27} = \alpha + 2\alpha^2 + \alpha^3 = \alpha^{20} \Rightarrow I_P^3 = 2$.

$j = 4 \Rightarrow Tr(\alpha^{40}) = \alpha^4 + \alpha^{36} = 2 + \alpha + 2\alpha^2 + 2\alpha^3 = \alpha^{50} \Rightarrow I_P^4 = 5$.

$j = 5 \Rightarrow Tr(\alpha^{50}) = \alpha^5 + \alpha^{45} = 0 \Rightarrow I_P^5 = \infty$.

$j = 6 \Rightarrow Tr(\alpha^{60}) = \alpha^6 + \alpha^{54} = 2 + \alpha + 2\alpha^2 + 2\alpha^3 = \alpha^{50} \Rightarrow I_P^6 = 5$.

$j = 7 \Rightarrow Tr(\alpha^{70}) = \alpha^7 + \alpha^{63} = 2\alpha + \alpha^2 + \alpha^3 = \alpha^{60} \Rightarrow I_P^7 = 6$.

Như vậy ta có bậc lồng ghép là:

$$I_P^T = \{4, 6, 6, 2, 5, \infty, 2, 0, 5, 6\} \quad (2.25)$$

Có thể thấy rằng bậc lồng ghép tìm được cũng hoàn toàn trùng khớp với kết quả thu được trong phần 2.2.1

2.2.3 Phương pháp tính trực tiếp tập thứ tự lồng ghép

Thực tế khi xây dựng chương trình trên máy vi tính để cài đặt hai phương pháp phân rã m-dãy đã nêu trên, việc cài đặt cả hai phương pháp trên là không hiệu quả, đặc biệt là khi độ dài dãy tăng lên. Ta chỉ có thể lập được bảng 2.1 nếu tổng kích thước của bảng có thể lưu hiệu quả trong bô nhớ máy tính (cỡ 10^9 phần tử). Đồng thời ta cần phải thực hiện đầy đủ T bước tinh toán để xây dựng bậc lồng ghép gồm T phần tử. Do đó các tác giả đã đưa ra một quy trình nhanh để tìm ra những phần tử đầu tiên của bậc lồng ghép[60].

Trước hết ta sinh ra phần đầu của chuỗi $\{b_n\}$ với trạng thái ban đầu được cho trước, nhưng thay vì tạo chuỗi đầy đủ chu kỳ (p^n-1 phần tử), ta chỉ cần tạo $m*T$ giá trị đầu tiên.

Tiếp đó ta sẽ sắp xếp lại các giá trị này bằng cấu trúc xen kẽ như định nghĩa của dãy lồng ghép, từ đó ta có thể nhận được trực tiếp các trạng thái ban đầu của dãy con $F_i(d)$. Vị trí của mỗi trạng thái này trong dãy con độ dài đầy đủ chính là giá trị tương ứng trong I_P^T.

Từ vị trí của trạng thái này (liên quan đến thứ tự xen kẽ), Ta chỉ cần sao chép N giá trị đồng thời của F^* vào chuỗi đầu ra. Nếu chuỗi con $F_i(d)$ có kích thước quá lớn, ta có thể xây dựng lại dãy từ trạng thái ban đầu mà ta vừa tìm thấy.

2.3. Xây dựng dãy phi tuyến lồng ghép

2.3.1 Kiến trúc dãy phi tuyến lồng ghép

Trong kiến trúc dãy phi tuyến, dãy đầu ra $\{b_n\}$ về bản chất vẫn là một m-dãy và có tính tuyến tính với bậc tuyến tính rất nhỏ. Nếu chỉ xét 1 đoạn ngắn của dãy, khi áp dụng thuật toán Belekamp-Massey (trình bày trong phần 3.1.2) ta sẽ thu được kết luận rằng đoạn dữ liệu đó được sinh bởi một đa thức bậc m. Trong trường hợp đoạn dữ liệu của dãy có chứa điểm ghép nối hai dãy con, thuật toán Belekamp-Massey cũng chỉ ra độ phức tạp tuyến tính bằng n.

Để tăng tính phi tuyến cho dãy đầu ra, nếu ta giữ nguyên thứ tự lồng ghép I_P^T nhưng thay các dãy con $\{a_n\}$ bằng các dãy con tương đương (về mặt độ lớn hàm

tự tương quan ACF và độ cân bằng), ta sẽ có được một dạng m-dãy phi tuyến, còn gọi là dãy tựa - m hoặc dãy GMW[49].

Như vậy đầu vào của dãy phi tuyến lồng ghép là hai m-dãy có cùng bậc n và đa thức sinh là $g(d)$ và $f(d)$ tương ứng. Ta sẽ áp dụng phương pháp sinh dãy lồng ghép cho cả hai m-dãy này với cùng các tham số n, m, T, từ đó tìm ra hai đa thức sinh cho dãy con tương ứng $g_1(d)$ và $f_1(d)$, cùng với hai thứ tự lồng ghép I_P^T và $I_P^{T\prime}$. Sử dụng thứ tự lồng ghép I_P^T nhưng lấy dãy con theo đa thức sinh $f_1(d)$,, ta thu được dãy đầu ra $\{e_n\}$ là dãy phi tuyến lồng ghép.

Thực tế ta có thể không cần tìm thứ tự lồng ghép $I_P^{T\prime}$ của dãy thứ hai, song đa thức sinh $f_1(d)$ vẫn cần xác định để tính được thứ tự lồng ghép I_P^T của dãy thứ nhất.

Trong phần tiếp theo, ta sẽ phân tích các tính chất của dãy phi tuyến lồng ghép để chứng minh rằng dãy mới tạo ra có tính phi tuyến cao hơn dãy lồng ghép.

2.3.2 Hàm tương quan của dãy phi tuyến lồng ghép

Hàm tương quan chéo (CCF - Cross-correlation function): Để tổng quát hóa, ta sẽ xem xét hàm tương quan chéo $R_{a,b}(\tau)$. Trước hết, trong trường hợp $a = b$ ta có hàm tự tương quan (autocorrelation function – ACF) $R_a(\tau)$.

Gọi $a, b \in A$ là hai dãy tam phân trong tập các dãy A. Cả a và b đều tuần hoàn với chu kỳ $L = 3^n-1$. Hàm tương quan chéo giữa a và b là $R_{a,b}(\tau)$ và được định nghĩa là:

$$R_{a,b}(\tau) = \sum_{i=0}^{L-1} \omega^{b_{i+\tau} - a_i} \quad , 0 \le \tau < L \tag{2.26}$$

trong đó $\omega = \exp\left(j\frac{2\pi}{3} \right)$ và phép cộng $(i + \tau)$ theo modulo L.

Tương tự như vậy, hàm tự tương quan của dãy a là :

$$R_a(\tau) = \sum_{i=0}^{L-1} \omega^{a_{i+\tau} - a_i} \text{ , } 0 \le \tau < L \text{ .} \quad (2.27)$$

Với các dãy tam phân, ta có định lý sau:

Định lý 2.1[J1]: Cho $\{a_i\}$ là dãy giả ngẫu nhiên trên trường GF(3^n) với chu kỳ $L = 3^n$-1, hàm tự tương quan của $\{a_i\}$ theo công thức:

$$R_a(\tau) = \begin{cases} 3^n - 1 & \tau = 0 \bmod (3^n - 1), \\ -1 & \tau \ne 0 \bmod (3^n - 1). \end{cases} \quad (2.28)$$

khi và chỉ khi tần số N_c là số lần xuất hiện ký tự $c \in$GF(3) trong 1 chu kỳ đầy đủ của dãy các sai khác $\{a_i\text{-}a_{i+\tau}\}$ giữa dãy $\{a_i\}$ và dãy dịch chuyển τ bước của nó $\{a_{i+\tau}\}$, thỏa mãn điều kiện:

$$N_c = \begin{cases} 3^{n-1} - 1 & c = 0; \\ 3^{n-1} & if \ c = 1; \\ 3^{n-1} & c = 2. \end{cases} \quad (2.29)$$

Ví dụ 5: Gọi $b(d)$ là dãy tam phân sinh bởi đa thức $f(d) = 1 + d^3 + 2d^4$, ta có:

$\{b_n\}$ = {1 0 0 0 1 0 0 2 1 0 1 1 1 2 0 0 2 2 0 1 0 2 2 1 1 0 1 0 1 2 1 2 2 1 2 0 1 2 2 2 2 0 0 0 2 0 0 1 2 0 2 2 2 1 0 0 1 1 0 2 0 1 1 2 2 0 2 0 2 1 2 1 1 2 1 0 2 1 1 1}.

Hàm tự tương quan của dãy $\{b_n\}$ là:

$$R_b(\tau) = \sum_{i=0}^{L-1} \omega^{b_{i+\tau} - b_i} \text{ , } 0 \le \tau < L \quad (2.30)$$

$R_b(\tau)$ = {80, -1, -1, -1, -1, -1, -1, -1,-1,-1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -l, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1).

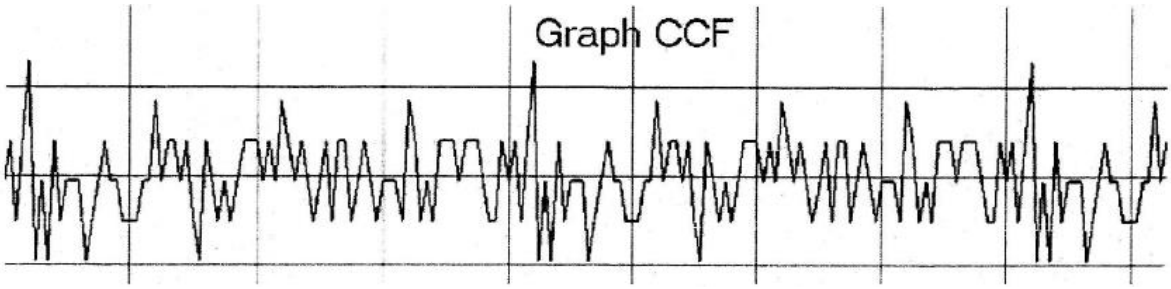
Xét dãy $a(d)$ là dãy tam phân sinh bởi đa thức $g(d) = 1 + d + 2d^2 + 2d^3 + 2d^4$,

$\{a_n\}$ = {1 0 0 0 1 2 2 1 1 1 0 0 2 2 0 1 0 0 1 0 1 0 2 2 1 0 2 1 2 1 1 0 1 1 1 1 2 1 0 1 2 1 0 1 2 1 1 2 2 2 0 0 1 1 0 2 0 0 2 0 2 0 1 1 2 1 1 2 1 2 2 0 2 2 2 2 1 2 0 1}.

Hàm tương quan chéo giữa *a* và *b* là:

$$R_{a,b}(\tau) = \sum_{i=0}^{L-1} \omega^{b_{i+\tau} - a_i} \ , 0 \leq \tau < \mathrm{L} \qquad (2.31)$$

$R_{a,b}(\tau)$ ={-1, 8, -10, 8, 26, -19, -1, -19, 8, -10, -1, -1, -1, -19, -10, -1, 8, -1, -1, -10, -10, -10, -l,-1, 17,-1,8, 8,-1, 8,-10,-19, 8,-1,-10,-1.-10,-1, 8, 8, 8,-1, 8, -1,17,8,-1, 8,-1,-10,-1,8, -10, 8, 8, -10, -1, 8, -1, -10, -1, -1, -1, -10,17,8, -10, -1, -10, 8, 8, 8, -1, 8, 8, 8, -1, -10, -10, 8}.



Hình 2.2 Biểu đồ tương quan chéo 2 dãy trong ví dụ 2.3

2.3.3 Phân tích khoảng tương đương tuyến tính của các dãy phi tuyến lồng ghép

Để đánh giá mức độ phức tạp của các chuỗi phi tuyến, ta sẽ sử dụng khái niệm khoảng tương đương tuyến tính (Equivalent Linear Span - ELS [49]). ELS của một dãy được định nghĩa là bậc nhỏ nhất của đa thức sinh ra toàn bộ dãy đó. Ta biết rằng ELS có thể được tính bằng cách biểu diễn hàm vết cũng như biến đổi d [40] [49]. Trong luận án này, ta sẽ áp dụng biến đổi d để tính giá trị ELS.

Gọi biến đổi d của một dãy phi tuyến là:

$$c(d) = \frac{S_c(d)}{g_c(d)}. \qquad (2.32)$$

Bằng cách áp dụng giải thuật Euclid cho đa thức, ta có thể trực tiếp tìm được bậc của *c(d).*

Gọi $K(d) = \gcd(g_c(d), S_c(d))$.

khi đó $c(d) = \frac{K(d).S_c'(d)}{K(d).g_c'(d)} = \frac{S_c'(d)}{g_c'(d)}$, (2.33)

trong đó gcd$(S_c'(d), g_c'(d))$ = 1.

Giá trị ELS of *c(d)* cũng bằng bậc của $g_c'(d)$, chính là đa thức có bậc nhỏ nhất có thể sinh ra *c(d)*.

ELS = deg$(g_c'(d))$ = deg(g_c(*d*)) - deg(K(*d*)) . (2.34)

Ta sẽ trình bày quy trình theo từng bước để xác định ELS của dãy phi tuyến lồng ghép:

Bước 1:

Từ thứ tự lồng ghép I_P^T và giá trị của dãy mới tạo ra $\{e_n\}$, ta rút ra biến đổi d của dãy phi tuyến lồng ghép:

$c(d) = \sum_{i=0}^{S-1} d^i Z_i(d^S)$. (2.35)

Trong đó $Z_i(d^S)$ biểu diễn một dãy con với bước dịch cụ thể từ dãy $\{e_n\}$, theo mô tả trong I_P^S. Ta có thể thấy rằng:

$Z_i(d^S) = \frac{S_{e_i}(d^S)}{g_{e_S}(d^S)}$. (2.36)

trong đó $S_{e_i}(d^S)$ và $g_{e_S}(d^S)$ là bước dịch pha và đa thức sinh tương ứng của $\{e_n\}$. Vì thế ta có:

$c(d) = \sum_{i=0}^{S-1} \frac{d^i S_{e_i}(d^S)}{g_{e_T}(d^S)}$. (2.37)

Bước 2:

Áp dụng thuật toán Euclid vào công thức (2.37) ta sẽ có được đa thức sinh bậc nhỏ nhất sinh ra *c*(*d*) và từ đó nhận được giá trị ELS là :

ELS = deg $g_1(d^S)$ - *deg*(*K*(*d*)) (2.38)

trong đó *K*(*d*) = *gcd*(*G*(*d*),$g_1(d^S)$) .

Ví dụ 5: Xét m - dãy $\{b_n\}$ sinh bởi đa thức $g(d) = 1 + d^3 + 2d^4$ với các tham số:

$L = 3^4 - 1 = 80$, $N = 3^2 - 1 = 8$, T = 10 .

Thứ tự lồng ghép I_P^S tương ứng với $\{b_n\}$ được xác định theo phương pháp trình bày trong phần 2.3 là:

$I_P^T = \{5, "\infty", 2, 0, 5, 6, 5, 7, 7, 3\}$.

Ta sẽ thay thế dãy con của dãy lồng ghép bằng dãy sinh bởi đa thức:

$g_1(d) = 1 + 2d + 2d^2$.

Bằng cách áp dụng phương pháp mở rộng dãy theo biến đổi d, ta có

$G(D) = (2+D) + 0.d + (D)d^2 + (2+2D)d^3 + (2+D)d^4 + 2D.d^5 + (2+D)d^6 + 2.d^7 + 2.d^8$.

Thay thế D bằng d^{10} ta được:

$G(d) = 2 + 2d^3 + 2d^4 + 2d^6 + 2d^7 + 2d^8 + d^9 + d^{10} + d^{12} + 2d^{13} + d^{14} + 2d^{15} + d^{16}$.

$$c(d) = \frac{G(d)}{1+2d^{10}+2d^{20}} .$$

Áp dụng thuật toán Euclid, ta có:

$K(d) = \gcd(G(d), g_1(d^S))$,

$K(d) = 2d^8 + d^7 + 2d^6 + d^5 + d^3 + 2d + 2$.

Từ đó ta tính được giá trị ELS là:

$ELS = \deg g_1(d^T) - \deg(K(d)) = 20 - 8 = 12$.

Vì giá trị ELS bằng 12, lớn hơn ELS của dãy ban đầu $\{b_n\}$, ta có thể kết luận rằng dãy mới sinh ra $\{e_n\}$ có tính chất phi tuyến cao hơn so với dãy ban đầu.

2.3.4 Một số kết quả thực hành sinh dãy phi tuyến lồng ghép trên GF(p^n)

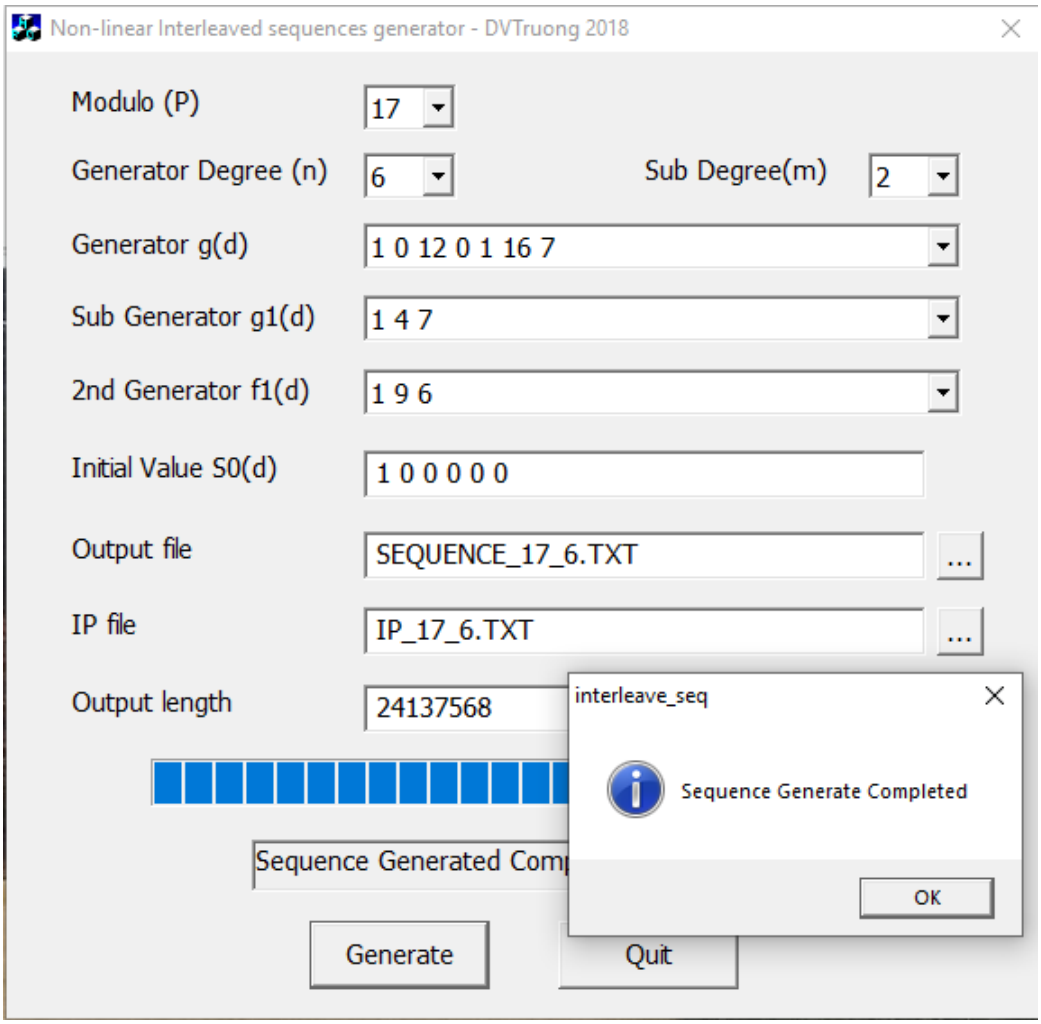
Các tác giả đã lập một chương trình máy tính bằng ngôn ngữ lập trình C để giả lập quá trình sinh các dãy phi tuyến lồng ghép trên GF(p^n) với các tham số cụ thể có thể lựa chọn trên giao diện phần mềm như hình 2.3 [60]:

Thử nghiệm thứ nhất

Các tham số của bộ tạo dãy là: p=5, n = 6, m = 3, đa thức sinh $g(d) = d^6 + 3d^5 + 2d^4 + 4d^3 + d + 2$, $g_1(d) = d^3 + d^2 + d + 3$. Từ đó ta có N = 15 624, L=124, T = 126.

Trước hết ta sinh ra m-dãy ban đầu $\{b_n\}$ và xác định thứ tự lồng ghép I_P^S:

I_P^T= (0, ∞, 63, 1, 107, 53, 9, 51, 88, 20, 45, 71, 115, 9, 56, 97, 105, 3, 17, 56, 33, 83, 25, 96, 45, 58, 111, 72, 120, 56, 68, 73, 75, 47, 54, 52, 10, 71, 88, 83, 87, 42, 5, 20, 109, 77, 43, 64, 74, 123, 115, 49, 83, 16, 48, 48, 21, 48, 122, 103, 9, 111, 96, 107, 77, 108, 98, 114, 13, 108, 4, 55, 29, 57, 58, 27, 95, 62, 5, 14, 90, 81, 61, 96,



Hình 2.3 Phần mềm mô phỏng sinh dãy phi tuyến lồng ghép

5, 41, 27, 65, 111, 108, 114, 98, 38, 81, 84, 78, 107, 106, 102, 91, 32, 109, 25, 97, 85, 13, 67, 2, 77, 101, 63, 50, 29, 22, 106, 60, 43, 0, 99, 75, 20, 108, 67, 112, 43, 62).

Toàn bộ chu kỳ của dãy con là: 1 0 0 2 3 0 1 0 4 3 3 2 1 3 0 4 2 4 2 3 3 3 0 3 3 4 4 3 1 4 1 2 0 0 4 1 0 2 0 3 1 1 4 2 1 0 3 4 3 4 1 1 1 0 1 1 3 3 1 2 3 2 4 0 0 3 2 0 4 0 1 2 2 3 4 2 0 1 3 1 3 2 2 2 0 2 2 1 1 2 4 1 4 3 0 0 1 4 0 3 0 2 4 4 1 3 4 0 2 1 2 1 4 4 4 0 4 4 2 2 4 3 2 3.

Tiếp theo, ta tính toán các thuộc tính của một dãy với các tham số tương ứng (cùng các giá trị p, n, m) nhưng đa thức sinh lại sử dụng đa thức $f(d) = d^6 + 2d^5 + d^4 + 4d^3 + 2d + 3$, $f_1(d) = d^3 + 3d^2 + 2$.

Trong trường hợp đó, thứ tự lồng ghép I_P^T được tính là :

$I_P'^T$ = (73, 114, 92, 61, 52, 46, 111, 13, 76, 58, 26, 62, 109, 57, 1, 53, 12, 7, 78, 35, 27, 35, 37, 122, 123, 29, 33, 123, 66, 55, 93, 113, 116, 13, 42, 26, 114, 29, 95, 30, 116, 29, 46, 18, 122, 120, 101, 64, 76, 10, 45, 42, 13, 13, 53, 52, 45, 54, 98, 28, 34, 76, 25, 82, 11, 89, 54, 87, 106, 44, 19, 79, 86, 96, 4, 27, 111, 14, 1, 44, 23, 0, 49, 64, 54, 9, 83, 99, 0, 107, 47, 60, 11, 42, 112, 117, 49, 2, 93, 14, 111, ∞,, 50, 78, 34, 68, 116, 61, 57, 112, 82, 8, 120, 57, 75, 51, 36, 87, 9, 20, 6, 82, 106, 4, 86, 100).

Sau cùng, ta sinh ra dãy phi tuyến lồng ghép bằng cách áp dụng thứ tự lồng ghép thứ hai $I_P'^T$ trong quy trình sinh dãy lồng ghép thứ nhất. Chuỗi kết quả đầu ra nhận các giá trị là :

1 1 4 2 1 0 3 4 3 4 1 1 1 0 1 1 3 3 1 2 3 2 4 0 0 3 2 0 4 0 1 2 2 3 4 2 0 1 3 1 3 2 2 2 0 2 2 1 1 2 4 1 4 3 0 0 1 4 0 3 0 2 4 4 1 3 4 0 2 1 2 1 4 4 4 0 4 4 2 2 4 3 2 3 1 0 0 2 3 0 1 0 4 3 3 2 1 3 0 4 2 4 2 3 3 3 0 3 3 4 4 3 1 4 1 2 0 0 4 1 0 2 0 3 …

Thử nghiệm thứ hai

Ta sẽ thử nghiệm với một dãy lớn hơn với các tham số p=17, n = 6, m= 2, polynomial $g(d) = d^6 + 12d^4 + d^2 + 16d + 7$, $g_1(d) = d^2 + 4d + 7$, từ đó ta có các giá trị tham số:

N = 24 137 568, , L=288, T = 83 811.

Ta cũng sinh ra m-dãy ban đầu $\{b_n\}$ và xác định thứ tự lồng ghép I_P^T:

I_P^T= (214, 109, ∞, 271, 145, 217, 133, 199, 87, 73, 269, 133, 155, 152, 226, 167, 207, 86, 228, 217, 264, 194, 45, 50, 56, 219, 26, 16, 136, 134, 180, 257, 110, 217, 197, 164, 99, 188, 261, 280, 249, 75, 193, 241, 93, 186, 127, 108, 227, 170, 5, 61, 164, 53, 223, 239, …).

Phần đầu tiên của dãy con là: 1 12 13 0 11 7 14 14 16 8 9 10 16 2 16 7 13 1 7 16 6 0 9 15 13 13 10 5 12 2 10 14 10 15 6 7 15 10 8 0 12 3 6 6 2 1 16 14 2 13 2 3 8 15 3 2 5 0 16 4 8 8 14 7 10 13 14 6 14 4 5 3 4 14 1 0 10 11 5 5 13 15 2 6 13 8 13 11 1 4 11 13 7 0 2 9 1 1 6 3 14 8 6 5 6 9 7 11 9 6 15 0 14 …

Sau đó giữ nguyên các giá trị p, n, m nhưng thay thế đã thức sinh thành $f(d) = d^6 + 10d^5 + 6d^4 + 11d^3 + 16d + 6$, $f_1(d) = d^2 + 9d + 6$. Khi đó ta có thứ tự lồng ghép mới:

$I_P'^T$ = (21, 127, 1, 145, 127, 181, 47, 186, 166, 42, 186, 9, 119, 280, 118, 222, 246, 1, 239, 131, 180, 164, 22, 260, 156, 36, 31, 182, 84, 278, 80, 152, 105, 37, 233, 95, 252, 75, 224, 164, 284, 29, 206, 278, 211, 62, 161, 251, 54, 164, 102, 43, 53, 181, 209, 200, 56, 285, 36, 198, 194, …).

Áp dụng thứ tự lồng ghép thứ hai $I_P'^T$ trong quy trình sinh dãy lồng ghép thứ nhất. Chuỗi kết quả đầu ra nhận các giá trị là :

1 3 15 4 15 14 9 2 14 15 12 0 1 13 9 9 3 10 7 4 3 11 3 13 12 14 13 3 16 0 7 6 12 12 4 2 15 11 4 9 4 6 16 13 6 4 10 0 15 8 16 16 11 14 3 9 11 12 11 8 10 6 8 11 2 0 3 5 10 10 9 13 4 12 9 16 9 5 2 8 5 9 14 0 4 1 2 2 12 6 11 16 12 …

2.4 Phương pháp phân rã theo bước để sinh dãy lồng ghép

Ngoài ba phương pháp sinh dãy phi tuyến lồng ghép đã được nghiên cứu, trong công bố [J2] tác giả luận án đã nghiên cứu phương pháp phân rã m-dãy theo bước (decimation) và từ đó đưa ra một phương pháp sinh dãy phi tuyến lồng ghép có tính ứng dụng cao trong thực hành.

2.4.1 Phương pháp phân rã m-dãy theo bước

Gọi *A* là một m-dãy với bậc *n*, chu kỳ 2^n-1 và phần tử sinh α. Ta sẽ sinh ra một dãy mới *A*(*T*) bằng cách lấy các bit cách nhau *T* vị trí từ dãy *A*, bắt đầu từ bit đầu tiên. *A*(*T*) được gọi là dãy phân rã (hay dãy decimation) theo bước *T* từ dãy *A*. Giá trị *T* gọi là bước phân rã.

Nếu bước phân rã *T* và độ dài chu kỳ của m-dãy là nguyên tố cùng nhau, thì dãy phân rã theo bước *T* từ m-dãy ban đầu cũng là một m-dãy với phần tử sinh α^T, các tham số khác của m-dãy phân rã đều giống như dãy ban đầu (bậc, chu kỳ, đa thức sinh …) . Nói cách khác, dãy phân rã được tạo có độ dài chu kỳ và các tính chất giống như dãy ban đầu. Nếu quy m-dãy phân rã về một m-dãy với phần tử sinh α như các biểu diễn m-dãy thông thường, ta sẽ có một m-dãy mới với đa thức sinh mới có cùng bậc với đa thức ban đầu và cũng là một đa thức nguyên thủy.

Nếu bước phân rã có giá trị $T= p^u$ thì dãy phân rã là một dịch pha của dãy ban đầu. Trong trường hợp này bước phân rã và độ dài chu kỳ của m-dãy luôn là nguyên tố cùng nhau.

Nếu bước phân rã *T* và độ dài chu kỳ của m-dãy không phải là nguyên tố cùng nhau, khi đó dãy phân rã nhận được sẽ không còn là m-dãy mà là một dãy tuần hoàn có chu kỳ $\frac{p^n-1}{\gcd(p^n-1,T)}$. Dãy tuần hoàn này cũng là một LFSR có thể sinh bởi đa thức sinh ban đầu với phần tử sinh α^T, hoặc khi quy về LFSR với phần tử sinh α ta sẽ được một đa thức đặc trưng là đa thức bất khả quy, nhưng không phải là đa thức nguyên thủy.

Trong trường hợp rất đặc biệt: khi bước phân rã thỏa mãn *T* = *L* / *N* theo các điều kiện của dãy lồng ghép, bậc của đa thức sinh cho dãy phân rã là ước của bậc đa thức sinh ban đầu. Đó chính là dãy con đầu tiên của dãy lồng ghép được trình bày trong phần 2.2.1 của trong chương này.

Thử nghiệm phương pháp phân rã với một m-dãy cụ thể

Chọn m-dãy bậc *n* = 23 được sinh bởi đa thức đặc trưng:

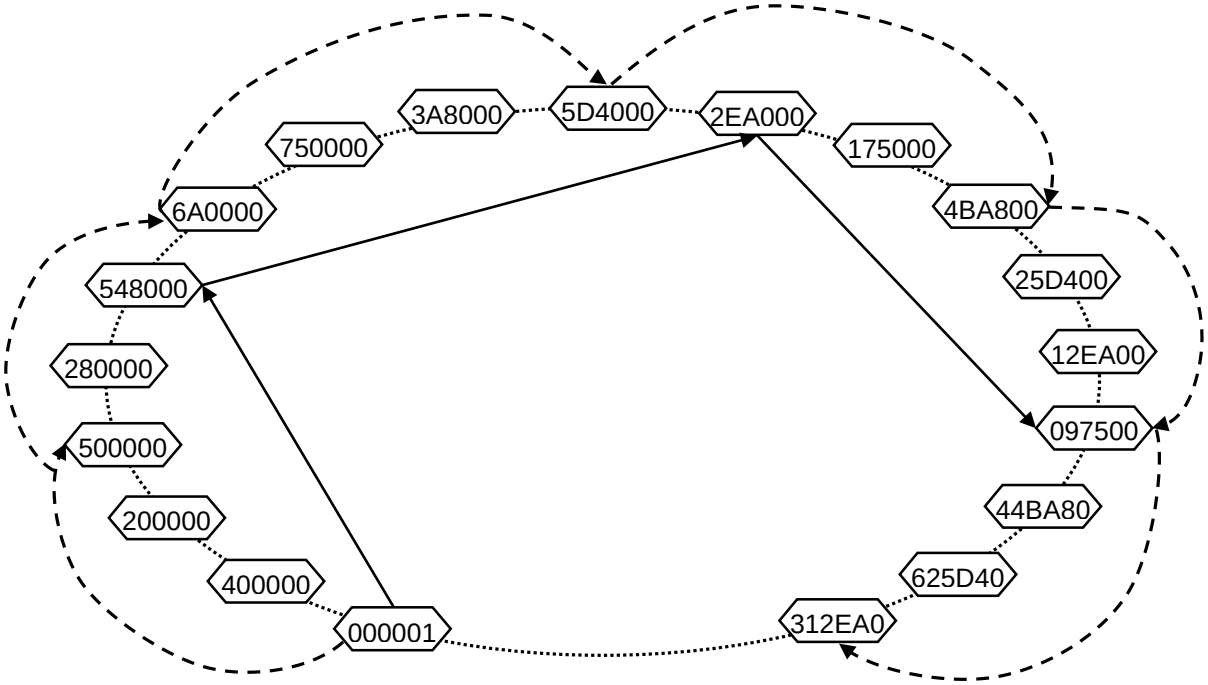
$f(x) = x^{23} + x^{18} + x^{15} + x^{14} + x^{11} + x^9 + x^5 + x^2 + 1.$

Với trạng thái khởi đầu $S_{(0)}$ = (1, 0, 0, …, 0), or (000001) theo cơ số 16 (trong phần này ta sẽ dùng giá trị số theo cơ số 16 để biểu diễn các trạng thái thanh ghi của m-dãy).

Áp dụng công thức tính giá trị phản hồi theo Fibonacy, ta có thể tính được 16 trạng thái liên liếp của thanh ghi m-dãy như sau:

400000, 200000, 500000, 280000, 548000, 6A0000, 750000, 3A8000, 5D4000, 2EA000, 175000, 4BA800, 25D400, 12EA00, 097500, 44BA80, 625D40, 312EA0

Chọn bước phân rã lần lượt bằng 3 và bằng 5, ta sẽ có các bước nhảy tương ứng như trong hình 2.4:



Hình 2.4 Phân rã m-dãy theo bậc 3 và 5

Tác giả đã xây dựng một ứng dụng nhỏ trên máy tính để thực nghiệm phương pháp phân rã m-dãy sử dụng ngôn ngữ lập trình C. Để xác định đa thức sinh mới của dãy phân rã, ta sử dụng thuật toán Belekamp-Massey trên dãy đầu ra.

Sử dụng phần mềm trên để sinh các dãy phân rã theo bước và phân tích các dãy đầu ra ta có các kết quả sau:

Với $T = 5$, dãy phân rã thu được là một m-dãy được sinh bởi đa thức:

$g_1(x) = x^{23} + x^{22} + x^{18} + x^{17} + x^{16} + x^{15} + x^{12} + x^{10} + x^7 + x^6 + x^5 + x^3 + x^2 + x + 1$.

Với T = 3, dãy phân rã thu được là một m-dãy được sinh bởi đa thức:

$g_2(x) = x^{23} + x^{20} + x^{19} + x^{18} + x^{17} + x^{15} + x^{14} + x^{12} + x^9 + x^7 + x^6 + x^5 + x^4 + x^2 + 1$.

Bảng 2.2 Kết quả phân rã m-dãy

TT	Bậc đa thức	Đa thức sinh	Bước phân rã	Đa thức mới
1	24	$x^{24} + x^{20} + x^{19} + x^{17} + x^{15} + x^{13} + x^{12} + x^{10} + x^8 + x^4 + 1$	11	$x^{24} + x^{22} + x^{20} + x^{17} + x^{14} + x^{11} + x^9 + x^8 + x^5 + x^3 + 1$
2	23	$x^{23} + x^{18} + x^{15} + x^{14} + x^{11} + x^9 + x^5 + x^2 + 1$	5	$x^{23} + x^{22} + x^{18} + x^{17} + x^{16} + x^{15} + x^{12} + x^{10} + x^7 + x^6 + x^5 + x^3 + x^2 + x + 1$
3	23	$x^{23} + x^{18} + x^{15} + x^{14} + x^{11} + x^9 + x^5 + x^2 + 1$	3	$x^{23} + x^{20} + x^{19} + x^{18} + x^{17} + x^{15} + x^{14} + x^{12} + x^9 + x^7 + x^6 + x^5 + x^4 + x^2 + 1$
4	17	$x^{17} + x^{10} + x^6 + x + 1$	19	$x^{17} + x^{11} + x^{10} + x^9 + x^6 + x + 1$
5	14	$x^{14} + x^{12} + x^9 + x^7 + x^6 + x + 1$	8	$x^{14} + x^{12} + x^9 + x^7 + x^6 + x + 1$
6*	21	$x^{21} + x^{18} + x^{16} + x^{13} + x^{10} + x^7 + x^4 + x^3 + x^2 + x + 1$	7	$x^{21} + x^{17} + x^{16} + x^{15} + x^{14} + x^{10} + x^5 + x^3 + 1$
7**	21	$x^{21} + x^{18} + x^{16} + x^{13} + x^{10} + x^7 + x^4 + x^3 + x^2 + x + 1$	16513	$x^7 + x^6 + x^5 + x^3 + x^2 + x + 1$

* Bước phân rã không nguyên tố cùng nhau với chu kỳ dãy, dẫn tới đa thức mới là đa thức bất khả quy song không phải đa thức nguyên thủy

** Bước phân rã không nguyên tố cùng nhau với chu kỳ dãy và thỏa mãn điều kiện kiến trúc dãy lồng ghép, dẫn tới đa thức mới có bậc là ước của bậc dãy ban đầu

Với dãy thử nghiệm thứ 5, ta thu được đa thức sinh cho dãy phân rã hoàn toàn giống như đa thức sinh ban đầu, do bước phân rã $T = 2^3$.

2.4.2. Giải pháp để xây dựng dãy phân rã một cách hiệu quả

Theo đúng các bước của phương pháp phân rã, ta cần tính toán T trạng thái trong của m-dãy để tạo ra 1 bit của dãy phân rã đầu ra.

Nếu ta sử dụng biến đổi - d để tính trạng thái mới (hoặc phương pháp Gaussian), thay vì tính toán từng trạng thái bằng cách nhân trạng thái trong hiện tại với d trong mỗi bước trên trường GF(q^n), ta có thể tính trực tiếp trạng thái bên trong sau bước T bằng trạng thái bên trong hiện tại nhân với d^T trên trường GF(q^n). Ta sẽ không cần phải lưu trạng thái trong của T-1 bước trung gian. Tuy nhiên trong quá trình tính toán đa thức trên trường GF(q^n) cần tính phép modulo đa thức với T hệ số. Quá trình tính toán này sẽ cần nhiều thời gian, đặc biệt khi giá trị T rất lớn.

Khi ta sử dụng phương pháp Fibonacci (là cách phổ biến để sinh m-dãy trong các vi xử lý), ta có thể thực hiện việc phân rã nhanh hơn bằng cách thực hiện các tính toán trước theo phương pháp sau:

Giả sử trạng thái trong của m-dãy được lưu trong một thanh ghi dịch ký hiệu $\{a_i\}$ (i = 0..n-1).

Công thức sinh bit phản hồi cho m-dãy theo Fibonacci như sau:

$$a_n = \sum_{i=0}^{n-1} a_i \cdot f_{n-i} \ . \qquad (2.39)$$

Ví dụ 2.4: ta lấy lại tham số trong thử nghiệm trên với bước phân rã được chọn là T = 5. Ta biết rằng đa thức sinh của dãy phân rã là :

$g_1(x) = x^{23} + x^{22} + x^{18} + x^{17} + x^{16} + x^{15} + x^{12} + x^{10} + x^7 + x^6 + x^5 + x^3 + x^2 + x + 1.$

Từ các bit của trạng thái ban đầu $S_{(0)} = \{a_0, a_1, \ldots, a_{22}\}$, sử dụng phương pháp Fibonacci để lập công thức cho cả T bit liên tiếp, tính theo tham số là n bit đầu vào như sau (sử dụng đa thức sinh của dãy ban đầu):

$$\boldsymbol{a_{23}} = a_0 \wedge a_5 \wedge a_8 \wedge a_9 \wedge a_{12} \wedge a_{14} \wedge a_{18} \wedge a_{21} \ .$$

$\boldsymbol{a_{24}} = a_1 \wedge a_6 \wedge a_9 \wedge a_{10} \wedge a_{13} \wedge a_{15} \wedge a_{19} \wedge a_{22}$.

$\boldsymbol{a_{25}} = a_2 \wedge a_7 \wedge a_{10} \wedge a_{11} \wedge a_{14} \wedge a_{16} \wedge a_{20} \wedge a_{23}$.

$= a_2 \wedge a_7 \wedge a_{10} \wedge a_{11} \wedge a_{14} \wedge a_{16} \wedge a_{20} \wedge a_0 \wedge a_5 \wedge a_8 \wedge a_9 \wedge a_{12} \wedge a_{14} \wedge a_{18} \wedge a_{21}$

$= a_0 \wedge a_2 \wedge a_5 \wedge a_7 \wedge a_8 \wedge a_9 \wedge a_{10} \wedge a_{11} \wedge a_{12} \wedge a_{16} \wedge a_{18} \wedge a_{20} \wedge a_{21}$.

$\boldsymbol{a_{26}} = a_3 \wedge a_8 \wedge a_{11} \wedge a_{12} \wedge a_{15} \wedge a_{17} \wedge a_{21} \wedge a_{24}$

$= a_1 \wedge a_3 \wedge a_6 \wedge a_8 \wedge a_9 \wedge a_{10} \wedge a_{11} \wedge a_{12} \wedge a_{13} \wedge a_{17} \wedge a_{19} \wedge a_{21} \wedge a_{22}$.

$\boldsymbol{a_{27}} = a_4 \wedge a_9 \wedge a_{12} \wedge a_{13} \wedge a_{16} \wedge a_{18} \wedge a_{22} \wedge a_{25}$

$= a_0 \wedge a_2 \wedge a_4 \wedge a_5 \wedge a_7 \wedge a_8 \wedge a_{10} \wedge a_{11} \wedge a_{13} \wedge a_{20} \wedge a_{21} \wedge a_{22}$.

Như vậy trạng thái trong của thanh ghi sau T bước sẽ là:

$$S_{(5)} = \{a_5, a_6, \dots, a_{22}, \boldsymbol{a_{23}, a_{24}, a_{25}, a_{26}, a_{27}}\} . \quad (2.40)$$

Biểu diễn các hệ số dưới dạng ma trận như sau:

$$A = \begin{bmatrix} 0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ 0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0 \\ & & & & & & & & & \vdots & & & & & & & & & & & \\ 0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1 \\ 1&0&0&0&0&1&0&0&1&1&0&0&1&0&1&0&0&0&1&0&0&1&0 \\ 0&1&0&0&0&0&1&0&0&1&1&0&0&1&0&1&0&0&0&1&0&0&1 \\ 0&0&1&0&0&1&0&1&1&1&1&1&1&0&0&0&1&0&1&0&1&1&0 \\ 0&1&0&1&0&0&1&0&1&1&1&1&1&1&0&0&0&1&0&1&0&1&1 \\ 1&0&1&0&1&1&0&1&1&0&1&1&0&1&0&0&0&0&0&0&1&1&1 \end{bmatrix} \quad (2.41)$$

Với biểu diễn ma trận này, ta có thể tính trạng thái trong của thanh ghi sau T bước:

$$S_{(T)} = A\, S_{(0)}^T . \quad (2.42)$$

Sử dụng lặp lại công thức (2.42), ta có thể sinh ra toàn bộ dãy phân rã mà không cần tính các giá trị trung gian.

Chú ý là ta cần có được toàn bộ các giá trị các bit của trạng thái trong tại $S_{(5)}$ vì để tính giá trị bit a_{32} (bit tiếp theo trong quá trình phân rã dãy ban đầu) ra cần

toàn bộ các giá trị các bit của trạng thái trong tại $S_{(5)}$ để áp dụng cho (2.42). Tuy rằng ta có thể xây dựng công thức tính trực tiếp a_{32} từ $S_{(0)}$, nhưng ta không thể tính trước toàn bộ các công thức sinh cho từng bit của dãy phân rã.

Với $T < m$, ta có thể tính được các bit của dãy phân rã mà chỉ cần lưu T công thức trong (2.41), không cần thiết lưu toàn bộ ma trận A trong (2.42). Giải pháp này chỉ cần lưu trữ thông tin ít hơn, và hiệu năng sẽ tốt hơn so với phép nhân ma trận đầy đủ sử dụng (2.42).

Trong các thử nghiệm thực tế, nếu áp dụng giải pháp nêu trên, độ phức tạp tính toán không thay đổi giữa phương pháp truyền thống (tính trực tiếp theo (2.40)) và phương pháp sử dụng (2.41). Tuy nhiên trong phương pháp truyền thống, ta cần dịch chuyển thanh ghi dịch T lần, mỗi lần dịch chuyển 1 vị trí. Với phương pháp sử dụng (2.41) và (2.42) ta chỉ cần một lần dịch chuyển thanh ghi.

2.4.3 Phương pháp xây dựng dãy lồng ghép sử dụng phân rã theo bước

Trong phần 2.2.2 ta đã biết rằng dãy con thứ i của dãy lồng ghép chính là phân rã bậc T của m-dãy ban đầu với bước dịch pha bằng i. Áp dụng giải pháp xây dựng nhanh dãy phân rã trong phần 2.4.2, ta có thể xây dựng nên dãy con đầu tiên. Khi đã biết đa thức sinh của dãy con, ta không cần tính toán toàn bộ dãy con theo (2.41) hoặc (2.42) mà chỉ cần tính m bit đầu tiên của dãy con. Các bit còn lại được theo công thức sinh m-dãy (2.40) với bậc m và đa thức sinh của dãy con.

Trong khi tính m bit đầu tiên của dãy con, ta sẽ lưu lại m trạng thái trong của m-dãy ban đầu tương ứng. Từ m trạng thái trong này, áp dụng công thức sinh m-dãy (2.40) với bậc n và đa thức sinh của m-dãy ban đầu, ta sẽ lần lượt tính được m bit khởi đầu của các dãy con tiếp theo. Các bộ m bit khởi đầu này có thể sử dụng để sinh ra các dãy con tương ứng theo công thức sinh m-dãy với bậc m và đa thức sinh của dãy con. Ta cũng có thể sử dụng các bộ m bit khởi đầu để xác định bậc lồng ghép của dãy con tương ứng.

Như vậy khi sử dụng phương pháp phân rã theo bước, ta có thể tính trực tiếp mọi bộ *m* bit khởi đầu. Phương pháp này cần sử dụng dung lượng bộ nhớ để lưu *m* trạng thái của m-dãy ban đầu, tương ứng với kích thước *m.n*.

Thông thường trong thực tế ta chỉ cần sinh một phần của dãy lồng ghép với kích thước cho trước. Trong trường hợp đó, ta không cần thiết tính toán toàn bộ các phần tử của tập thứ tự lồng ghép I_P^T.. Ta chỉ cần tính các thứ tự lồng ghép tương ứng với số dãy con cần thiết để sinh ra đủ đầu ra với kích thước được yêu cầu.

2.5 Kết luận chương 2

Trong chương này, tác giả giới thiệu phương pháp xây dựng các dãy phi tuyến lồng ghép trên trường tam phân và *p*-phân có giá trị hàm tự tương quan và hàm phân phối rất tốt. Các phương pháp này dựa trên biến đổi *d*, áp dụng cho tất cả các chuỗi tuần hoàn với chu kỳ *L*.

Tiếp đó, bằng cách mở rộng chuỗi con $\{a_n\}$ hoặc phân tách chuỗi ban đầu $\{b_n\}$, ta có thể dễ dàng tìm ra thứ tự lồng ghép I_P^T để có thể lồng ghép các chuỗi con $\{a_n\}$ tạo thành chuỗi lồng ghép $\{e_n\}$. Phân tích một số thuộc tính thống kê của dãy mới tạo ra chỉ ra rằng các thuộc tính thống kê của các dãy lồng ghép tam phân là rất tốt về mặt hàm tương quan và hàm phân phối. Khi áp dụng phương pháp này cho dãy p-phân, ta sẽ nhanh chóng có được chu kỳ rất lớn của dãy lồng ghép đầu mà không cần tăng bậc của dãy lên quá nhiều. Cụ thể là trong thử nghiệm thứ hai, với *p*=17 ta có được chu kỳ khá lớn (~2.10^7) mà chỉ cần xử lý đa thức bậc 6, nếu sử dụng dãy nhị phân ta cần tới đa thức bậc 24 để có kết quả tương đương.

Phần cuối cùng của chương này giới thiệu một phương pháp mới để sinh dãy lồng ghép sử dụng kỹ thuật phân rã theo bước. Phương pháp này có thể áp dụng một trong thực tế để sinh một phần đầu tiên của dãy lồng ghép với kích thước cho trước. Khi giá trị bước lồng ghép *T* rất lớn, sử dụng phương pháp này cũng giúp ta không cần tính toán và lưu trữ toàn bộ tập các thứ tự lồng ghép I_P^T.

CHƯƠNG 3 : THUẬT TOÁN SINH DÃY PHI TUYẾN LỒNG GHÉP BẬC LỚN ỨNG DỤNG TRONG KỸ THUẬT MẬT MÃ

Nội dung chương này tập trung vào việc đề xuất một thuật toán hiệu quả để sinh dãy phi tuyến lồng ghép với bậc lớn, cùng với các phân tích đánh giá về lý thuyết cũng như tính toán thực nghiệm về độ phức tạp tính toán cũng như độ phức tạp lưu trữ của thuật toán, sử dụng các thông tin được tác giả luận án công bố trong [J2]. Trước khi đề xuất thuật toán sinh dãy phi tuyến lồng ghép, tác giả nghiên cứu một số phương pháp phân tích dãy giả ngẫu nhiên thường được sử dụng trong tấn công thám mã đối với hệ mã dòng xây dựng trên m-dãy.

3.1. Độ phức tạp tuyến tính của dãy giả ngẫu nhiên

Dãy giả ngẫu nhiên ngoài việc cần phải thỏa mãn những yêu cầu tổng thể về các tính chất giả ngẫu nhiên nói dung, chúng còn phải có các tính chất địa phương tốt liên quan tới tính chất ngoại suy tuyến tính và tính tương quan giữa các đoạn con một dãy giả ngẫu nhiên... Các khái niệm cơ bản sẽ được phân tích trong chương này là độ phức tạp tuyến tính[26] [40] và tương quan địa phương của các dãy nhị phân [3].

3.1.1. Khái niệm và tính chất cơ bản của độ phức tạp tuyến tính

Giả sử F là trường hữu hạn hoặc bất kỳ. Dãy $s_1, s_2,...$các phần tử của F được gọi là dãy ghi dịch tuyến tính bậc k, nếu tồn tại các hệ số $a_k, a_{k-1}, ..., a_0 \in F, a_k \neq 0$ sao cho:

$$a_k s_{i+k} + ...+ a_1 s_{i+1} + a_0 s_i = 0, \forall\ i = 1, 2, ... \quad (3.1)$$

Một cách tương đương, nếu tồn tại các hệ số $c_1, c_2, ..., c_k \in F$, sao cho quan hệ sau được thỏa mãn:

$$s_j = - \sum_{i=1}^{k} c_i.s_{j-i},\ j = k+1, k+2, ... \quad (3.2)$$

Qui ước dãy zêro 0, 0,... toàn các số không được gọi là dãy ghi dịch bậc 0.

Hiển nhiên một dãy ghi dịch được xác định hoàn toàn bởi quan hệ truy hồi (3.1) và các giá trị ban đầu $s_1, s_2,..., s_k$.

Định nghĩa 3.1[26]*:* Giả sử $s = s_1, s_2,...$ là một dãy tùy ý các phần tử của trường F. Giả sử n-là một số nguyên dương. Khi đó độ phức tạp tuyến tính $L_n(s)$ được xác định là số k-bé nhất sao cho dãy n-phần tử $s_1, s_2,... s_n$ trùng với n-số hạng đầu tiên của một dãy ghi dịch tuyến tính bậc k.

Độ phức tạp tuyến tính được xác định bởi tính chất sau:

Tính chất 3.1.

Giả thiết F = GF(q), q là số nguyên tố bất kỳ. Xét dãy $s = \{s_n\}_n$ với các phần tử thuộc GF(q). Khi đó tương ứng với dãy S ta có dãy $\{L_n(s)\}_n$ có tính chất sau.

Tính chất 1: Tính bị chặn: $0 \le L_n(s) \le n, \forall\ n \ge 1$.

Tính chất 2: Tính đơn điệu không giảm: $L_n(s) \le L_{n+1}(s), \forall\ n \ge 1$.

Tính chất 3: Mối quan hệ giữa các phần tử liên tiếp của dãy $\{L_n(s)\}_n$.

a) Nếu thanh ghi dịch tuyến tính độ dài $L_n(s)$ sinh ra dãy $s_1, s_2,... s_n$ mà cũng sinh ra dãy $s_1, s_2,... s_n, s_{n+1}$ thì $L_{n+1}(s) = L_n(s)$.

b) Nếu thanh ghi dịch tuyến tính độ dài $L_n(s)$ sinh ra dãy $s_1, s_2,... s_n$ nhưng không sinh ra dãy $s_1, s_2,..., s_n, s_{n+1}$ thì có hai khả năng xảy ra:

Nếu $2\ L_n(s) > n$, thì ta có $L_{n+1}(s) = L_n(s)$;

Nếu $2\ L_n(s) \le n$, thì $L_{n+1}(s) = n+1- L_n(s)$. (3.3)

3.1.2. Thuật toán tổng hợp độ phức tạp tuyến tính Berlekamp-Massey

Một số bổ đề phục vụ cho thuật toán Belekamp-Massey

Bổ đề 3.1. Nếu thanh ghi dịch tuyến tính độ dài L sinh ra dãy $s_0, s_1,..., s_{N-1}$ nhưng không sinh ra dãy $s_0, s_1,..., s_{N-1}, s_N$ thì khi đó bất kỳ thanh ghi dịch tuyến tính nào sinh ra dãy $s_0, s_1,..., s_{N-1}, s_N$ cũng sẽ có độ dài L' thỏa mãn:

$$L' \ge N+1- L\ .\qquad (3.4)$$

Bổ đề 3.2. Nếu thanh ghi dịch tuyến tính độ dài $L_N(s)$ sinh ra dãy $s_0, s_1,...$ s_{N-1} nhưng không sinh ra dãy $s_0, s_1,..., s_n, s_N$ thì

$$L_{N+1}(s) \geq max\{ L_N(s), N+1- L_N(s)\}. \quad (3.5)$$

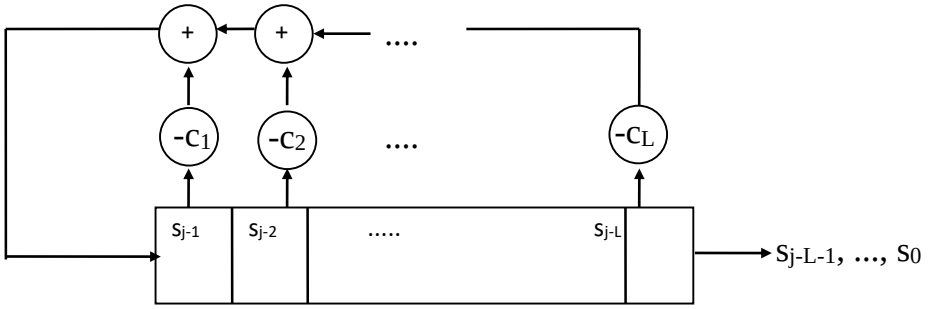
Thuật toán tổng hợp thanh ghi dịch phản hồi tuyến tính sinh ra một dãy cho trước

Sử dụng hai bổ đề nêu trên, ta sẽ trình bày thuật toán truy hồi tạo ra một trong những thanh ghi dịch phản hồi tuyến tính có độ dài $L_N(s)$ sinh ra dãy $s_0, s_1,...,$ s_{N-1} với $N = 1, 2, 3, ...$

Ký hiệu:

$$C(D) = 1 + c_1D + c_2D^2 + ... + c_LD^L \quad (3.6)$$

là đa thức liên kết với thanh ghi dịch phản hồi tuyến tính như trong Hình.3.1 có bậc tối đa là L. Nếu $L= 0$ ta lấy $C(D)= 1$ tương ứng với thanh ghi dịch độ dài 0.



Hình 3.1 Mô hình thanh ghi dịch phản hồi tuyến tính

Bây giờ với dãy s cho trước, ký hiệu:

$$\boldsymbol{C^{(N)}(D)} = 1+C_1^{(N)}D+...+C_{L_N(s)}^{(N)}D^{L_N(s)}. \quad (3.7)$$

là đa thức liên kết với thanh ghi dịch độ dài ngắn nhất $L_{(N)}(s)$ tạo ra dãy $s_0, s_1,..., s_N$.

Vậy ta có định lý sau:

Định lý 3.2 [40]**.** Nếu thanh ghi dịch tuyến tính độ dài $L_N(s)$ sinh ra dãy $s_0, s_1,... s_{N-1}$ mà cũng sinh ra dãy $s_0, s_1,..., s_{N-1}, s_N$ thì $L_{N+1}(s) = L_N(s)$. Ngược lại, nếu thanh ghi

dịch tuyến tính độ dài $L_N(s)$ sinh ra dãy $s_0, s_1,... s_{N-1}$ nhưng không sinh ra dãy $s_0, s_1,..., s_{N-1}, s_N$ thì $L_{N+1}(s) = max[L_N(s), N+1- L_N(s)]$.

Trong quá trình chứng minh Định lý 3.2 các tác giả đã xây dựng nên thuật toán tổng hợp thanh ghi dịch phản hồi tuyến tính độ dài ngắn nhất tạo ra dãy $s_0, s_1,... s_{n-1}$ như sau:

Thuật toán 3.1 Thuật toán tổng hợp LFSR (Berlekamp-Massey)

1) $1 \rightarrow C(D)$ $\quad 1 \rightarrow B(D)$ $\quad 1 \rightarrow x$

$0 \rightarrow L$ $\quad 1 \rightarrow b$ $\quad 0 \rightarrow N$

2) Nếu N = n, dừng thuật toán. Ngược lại tính

$$d = s_N + \sum_{i=1}^{L} c_i s_{N-i}.$$

3) Nếu d = 0, thì $x + 1 \rightarrow x$*, chuyển đến bước 6)*

4) Nếu $d \neq 0$ *và* $2L > N$*, thì*

$C(D) - d.b^{-1}.D^x.B(D) \rightarrow C(D)$

$x + 1 \rightarrow x$

và chuyển đến bước 6).

5) Nếu $d \neq 0$ *và* $2L \leq N$*, thì*

$C(D) \rightarrow T(D)$ *[lưu giữ tạm thời của C(D)]*

$C(D) - d.b^{-1}.D^x.B(D) \rightarrow C(D)$

$N + 1 - L \rightarrow L$

$T(D) \rightarrow B(D)$

$d \rightarrow b$

$1 \rightarrow x$

6) $N + 1 \rightarrow N$ *và quay về bước 2).*

Định lý 3.3[40]**.** Giả sử rằng thuật toán tổng hợp thanh ghi dịch LFSR Berlekamp-Massey được áp vào dãy $s_0, s_1,..., s_{n-1}$ và giả sử L, $C(D)$, x và $B(D)$ ký hiệu là các giá trị khi thuật toán dừng.

-Nếu $2L \leq n$, thì $C(D)$ là đa thức liên kết của thanh ghi dịch duy nhất độ dài tối thiểu L tạo ra dãy đó.

-Nếu $2L > n$, thì tập các đa thức :

$$\{C(D) + Q(D).D^x.B(D): \text{bậc của } Q(D) \text{ nhỏ hơn } 2L - n\}$$

là tập các đa thức liên kết của tất cả các thanh ghi dịch độ dài tối thiểu L sinh ra dãy đã cho.

3.1.3 Phân bố độ phức tạp tuyến tính của dãy ngẫu nhiên

Giả sử $F = \text{GF}(q)$ là trường hữu hạn q-phần tử. Với n là số nguyên dương bất kỳ, ta ký hiệu:

$$\Omega_n = [\text{GF}(q)]^n = \{(s_1, s_2,..., s_n), s_i \in F, i= 1, 2, .., n\}. \qquad (3.8)$$

là không gian của các dãy hữu hạn n-phần tử trên trường F. Như phần trên ta thấy gắn với mỗi dãy hữu hạn $s = (s_1, s_2,..., s_n)$ sẽ tồn tại thanh ghi dịch độ dài tối thiểu $L_n(s)$ sinh dãy đó.

Mệnh đề 3.8.

a) Nếu thanh ghi dịch tuyến tính độ dài $L_n(s)$ sinh ra dãy $s_1, s_2,... s_n$ mà cũng sinh ra dãy $s_1, s_2,... s_n, s_{n+1}$ thì $L_{n+1}(s) = L_n(s)$.

b) Nếu thanh ghi dịch tuyến tính độ dài $L_n(s)$ sinh ra dãy $s_1, s_2,... s_n$ nhưng không sinh ra dãy $s_1, s_2,..., s_n, s_{n+1}$ thì có hai khả năng xảy ra:

Nếu $2\ L_n(s) > n$, thì ta có $L_{n+1}(s) = L_n(s)$;

Nếu $2\ L_n(s) \leq n$, thì $L_{n+1}(s) = n+1- L_n(s)$. (3.9)

Từ Mệnh đề 2.8, ta thấy $L_n(s) < L_{n+1}(s)$ xảy ra (tương ứng với sự thay đổi bậc của thanh ghi dịch tuyến tính trong thuật toán Berlekamp-Massey) khi và chỉ

khi quan hệ truy hồi (2.1) đúng cho dãy n-phần tử $s_1, s_2,... s_n$ nhưng không đúng với dãy $(n+1)$ phần tử $s_1, s_2,... s_n, s_{n+1}$ và với điều kiện là:

$$2.L_n(s) \le n. \tag{3.10}$$

Lập luận trên sẽ được sử dụng trong bài toán ta sẽ xét sau đây.

Với các khái niệm đã nêu, ta ký hiệu:

$$D(n, \lambda) = \text{Card}\{ \boldsymbol{s} = (s_1, s_2,..., s_{n-1}, s_n) \in \Omega_n : L_n(\boldsymbol{s}) = \lambda\}.$$

Ta sẽ tìm công thức cho hàm phân bố $D(n, \lambda)$, với $n \in N$, và $0 \le \lambda \le n$.

Định lý 3.9. (F. G. Gustavson [26])

Hàm phân bố $D(n, \lambda)$, $n \in N$, $0 \le \lambda \le n$, được cho bởi công thức sau

$$D(n,\lambda) = \begin{cases} 1, & \lambda = 0; \\ q^{2\lambda-1}.q^*, & \lambda = 1,..,\alpha_n; \\ q^{2(n-\lambda)}.q^*, & \lambda = \alpha_n + 1,..,n. \end{cases} \tag{3.11}$$

trong đó, $q^* = q - 1$, và α_n là phần nguyên (dưới) của $(n/2)$.

Hệ quả 3.10.

Giá trị trung bình độ phức tạp tuyến tính của dãy hữu hạn n-phần tử lấy ngẫu nhiên đều trên không gian $[GF(q)]^n$ được đánh giá bởi công thức:

$$n/2 \le E(n) < (n+1)/2, \; n=1, 2,... \tag{3.12}$$

trong đó

$$E(n) = (1/q^n). \sum_{\lambda=0}^{n} \lambda.D(n;\lambda). \tag{3.13}$$

3.2. Tính chất tương quan địa phương của m-dãy

3.2.1. Khái niệm tương quan địa phương

Như ta đã biết, trong thực tế khi mã hóa một bản rõ nào đó trong hệ mã dòng đang xét, ta chỉ dùng đoạn khóa có độ dài M tương ứng với độ dài bản rõ để thực hiện phép mã hóa đó. Có thể giả thiết M là số cố định đối với một hệ thống mật mã thực tế nào đó. Nếu các đoạn khóa độ dài M xuất hiện ngẫu nhiên độc lập, đồng xác suất từ một nguồn khóa nào đó thì hệ mã dòng của ta là hệ mật hoàn thiện. Tuy nhiên nếu ta dùng khóa giả ngẫu nhiên, tức là các đoạn khóa độ dài M được sinh ra từ một thuật toán sinh dãy, khi đó ta cần quan tâm tới sự tương quan giữa các đoạn khóa độ dài M được lấy ngẫu nhiên đều từ nguồn khóa sinh ra.

Trước hết ta khảo sát tính chất này đối với khóa ngẫu nhiên lý tưởng.

Giả sử M là số tự nhiên cố định nào đó. Ký hiệu:

$$M = \{ (a_0, a_1, .., a_{M-1}), a_i \in \mathrm{GF}(2)\}.$$

Rõ ràng lực lượng của tập hợp M này là $\#M = 2^M$.

Giả sử ξ, η là các véc tơ ngẫu nhiên phân bố đều nhận giá trị trong không gian M. Khi đó véc tơ $\zeta = \xi \oplus \eta$ (với phép cộng véc tơ chính là phép XOR) cũng có phân bố đều trên không gian M.

Ký hiệu wt(ζ), $\zeta \in M$, là hàm trọng số của ζ. Khi đó wt(ζ) là biến ngẫu nhiên nhận giá trị trong tập hợp $\{0, 1, 2, ..., M\}$.

Đến đây ta có thể thiết lập bài toán như sau:

Giả thiết ζ lấy ngẫu nhiên đều trên không gian M. Tìm phân bố của biến ngẫu nhiên wt(ζ), trọng số của $\zeta \in M$.

Rõ ràng nếu $\zeta = (a_0, a_1, .., a_{M-1})$, thì $\mathrm{wt}(\zeta) = \sum_{i=0}^{M-1} a_i$, tức là wt(ζ) là số các bít 1 trong véc tơ ζ. Do đó, số các véc tơ ζ có $wt(\zeta) = k$, $0 \le k \le M$, sẽ chính bằng tổ hợp chập k trong M phần tử, tức là bằng C_M^k . Từ đó ta có hàm phân bố của biến ngẫu nhiên wt(ζ) có dạng sau:

$$P\{ wt(\zeta) = k, 0 \le k \le M\} = C_M^k / 2^M. \quad (3.14)$$

Dưới dạng bảng ta có:

wt(ζ)	0,	1,	... ,	k ,	...,	M-1,	M
P(wt(ζ))	$1/2^M$,	$M/2^M$ ...	$C_M^k / 2^M$, ...,	$M/2^M$,	$1/2^M$.		

Nhận xét:

a) Từ (3.14) ta thấy biến ngẫu nhiên wt(ζ), khi ζ lấy ngẫu nhiên đều trên không gian M, có phân bố đối xứng qua kỳ vọng của nó. Do đó nếu ta xét các mômen trung tâm cấp p tương ứng W_c^p, ta sẽ có $W_c^p = 0$, với p-lẻ. Ngoài ra Wainberg và Wolf cũng đã chỉ ra rằng:

$W_c^2 = M/4$; $W_c^4 = (3M^2 - 2M)/16$; $W_c^6 = (15M^3 - 30M^2 + 16M)/64$;

$W_c^p = 0$, với p-lẻ. (3.15)

Đây là những mômen quan trọng đầu tiên để ta làm căn cứ nhận định về phân bố giá trị tương quan địa phương đối với những nguồn giả ngẫu nhiên.

b) Khi nghiên cứu tương quan địa phương của một dãy nào đó, nếu ta chuyển sang việc khảo sát phân bố trọng số của các đoạn con độ dài M tương ứng trong một dãy đã biết, khi đó có thể dùng các công thức trong (3.14) để so sánh hai phân bố, trên cơ sở đó có thể kết luận sơ bộ về tính ngẫu nhiên địa phương trong phạm vi M của dãy đã cho.

3.2.2. Bài toán về tương quan địa phương của m-dãy

Giả sử $\{a_i\} = a_0, a_1, a_2, ..., a_{N-1}$ là một m-dãy nhị phân bậc r, có chu kỳ là:

$N = 2^r - 1$.

Đặt:

$M^* = \{ (a_i, a_{i+1}, .., a_{i+M-1}), i = 0, 1, 2, ..., N-1\}$, $M > 0$,

là tập các đoạn con độ dài M của m-dãy $\{a_i\}$. Rõ ràng lực lượng của M^* bằng $\#M^* = 2^r - 1 = N$.

Giả sử xét $M_1, M_2 \in M^*$, $M_1 \neq M_2$. Khi đó theo tính cất của m-dãy ta có $M^* = M_1 \oplus M_2 \in M^*$. Xét hàm tương quan giữa M_1 và M_2 thiết lập bởi công thức:

$$C(M_1, M_2) = (1/M)[M - 2wt(M^*)]. \quad (3.16)$$

Khi M_1, M_2 lấy ngẫu nhiên đều trên M^* , với $M_1 \neq M_2$, thì ta có thể xem rằng M^* cũng lấy ngẫu nhiên đều trên M^*. Do đó, từ (3.16) để xét phân bố giá trị tương quan giữa các đoạn con trong M^*, ta có thể chuyển xét (một cách tương đương) phân bố trọng số $wt(M^*)$, $M^* \in M^*$. Từ đó ta có bài toán sau:

Giả thiết véc tơ ζ^* lấy ngẫu nhiên đều trên không gian M^*. Tìm phân bố của biến ngẫu nhiên $wt(\zeta^*)$, trọng số của $\zeta^* \in M^*$.

Rõ ràng hàm phân bố của biến ngẫu nhiên $wt(\zeta^*)$ phụ thuộc vào các tham số độ dài đoạn con M, chu kỳ N và bản thân m-dãy đang xét $\{a_i\}$. Nếu $M \leq r$, theo tính chất phân bố của r-tuples, việc giải bài toán trên là rõ ràng. Tuy nhiên khi xét độ dài M nằm trong khoảng $r < M < N$, thì bài toán trở nên phức tạp cả về phương diện lý thuyết lẫn thực hành. Do đó để nhận diện phân bố trên, ta có thể đi tìm một số mômen đầu tiên của phân bố đó, trên cơ sở đó sẽ thấy được những đặc trưng hình học chủ yếu của phân bố cần tìm.

Vì vậy nhiệm vụ đặt ra trong phần này là thiết lập công thức tính mômen của biến ngẫu nhiên $wt(\zeta^*)$, $\zeta^* \in M^*$, so sánh với mômen của $wt(\zeta)$, $\zeta \in M$, trên cở sở đó nhận xét về tương quan địa phương của m-dãy đó.

3.2.3. Mômen phân bố trọng số của m-dãy

Giả sử $\{a_i\} = a_0, a_1, a_2, ..., a_{N-1}$ là m-dãy với các tham số ký hiệu như trên.

Xét dãy $\{b_i\}$ tương ứng với $b_i = 1 - 2.a_i$, $b_i \in \{-1, +1\}$.

Ký hiệu:

$$W_n = wt(a_n, a_{n+1}, .., a_{n+M-1}) = \sum_{i=0}^{M-1} a_{n+i} \quad (3.17)$$

là trọng số của M-tupel ($a_n, a_{n+1}, .., a_{n+M-1}$), n = 0, 1, 2, ..., N-1, với M > r.

Gọi A_ω bằng số các M-tupels trong M^* có $W_n = \omega$. Ta thấy nếu M > r thì ω chỉ nhận các giá trị từ 1 đến M (không nhận giá trị 0).

Giả sử $S_n = \sum_{i=0}^{M-1} b_{n+i}$. Khi đó rõ ràng là $S_n = M - 2.W_n$, và S_n nhận giá trị trong đoạn [$-M$, $+M$].

Ta cần tìm mômen của các phân bố $\{W_n\}$ và $\{S_n\}$.

Ký hiệu W^p, S^p là mômen bậc p của các phân bố tương ứng. Theo định nghĩa ta có:

$$\begin{aligned} W^p &= \frac{1}{N}\sum_{n=0}^{N-1} W_n^p = \frac{1}{N}\sum_{\omega=1}^{M} \omega^p . A_\omega \\ S^p &= \frac{1}{N}\sum_{n=0}^{N-1} S_n^p = \frac{1}{N}\sum_{\omega=1}^{M} (M-2\omega)^p . A_\omega \end{aligned} \quad (3.18)$$

Các mômen trung tâm cấp p tương ứng xác định bởi công thức:

$$\begin{aligned} W_c^p &= \frac{1}{N}\sum_{n=0}^{N-1} (W_n - W^1)^p ; \\ S_c^p &= \frac{1}{N}\sum_{n=0}^{N-1} (S_n - S^1)^p . \end{aligned} \quad (3.19)$$

Và hiển nhiên có:

$$S_c^p = (-2)^p . W_c^p . \quad (3.20)$$

Để thuận tiện ta tính các mômen S^p của phân bố $\{S_n\}$ tương ứng với dãy $\{b_i\}$.

Công thức tính mômen bậc 1.

Từ (3.19) ta có:

$$S^1 = \frac{1}{N}\sum_{n=0}^{N-1} S_n = \frac{1}{N}\sum_{n=0}^{N-1}\sum_{i=0}^{M-1} b_{n+i} = \frac{1}{N}\sum_{i=0}^{M-1}\sum_{n=0}^{N-1} b_{n+i} = -\frac{M}{N}. \quad (3.21)$$

(Chú ý: $\sum_{n=0}^{N-1} b_{n+i} = -1$, do tính chất của m-dãy).

**) Công thức tính mômen bậc 2.*

$S^2 =$

$$\frac{1}{N}\sum_{n=0}^{N-1} S_n{}^2 = \frac{1}{N}\sum_{n=0}^{N-1}(\sum_{i=0}^{M-1} b_{n+i})^2 = \frac{1}{N}\sum_{n=0}^{N-1}(M + 2\sum_{i=0}^{M-2}\sum_{j=i+1}^{M-1} b_{n+i}b_{n+j}) = M + \frac{2}{N}\sum_{i=0}^{M-2}\sum_{j-i+1}^{M-1}\sum_{n=0}^{N-1} b_{n+i}b_{n+j}.$$

Từ tính chất của nhóm nhân $\{b_{n+i}\}.\{b_{n+j}\} = \{b_{n+k}\}$, với $i \neq j$, nên ta có

$$S^2 = M + (2/N)\{\ C_M^2 .(-1)\} = M.[1- (M-1)/N] \quad (3.22)$$

**) Công thức tính mômen bậc 3.*

Tiếp tục như trên ta có:

$S^3 =$

$$= \frac{1}{N}\sum_{n=0}^{N-1}(\sum_{i=0}^{M-1} b_{n+i})^3$$
$$= \frac{1}{N}\sum_{n=0}^{N-1}\{(3M-2).\sum_{i=0}^{M-1} b_{n+i} + 3!\sum_{i=0}^{M-3}\sum_{j=i+1}^{M-2}\sum_{k=j+1}^{M-1} b_{n+i}.b_{n+j}.b_{n+k}\}$$
$$= \frac{-M(3M-2)}{N} + \frac{3!}{N}\left\{\sum_{i=0}^{M-3}\sum_{j=i+1}^{M-2}\sum_{k=j+1}^{M-1}\left(\sum_{n=0}^{N-1} b_{n+i}.b_{n+j}.b_{n+k}\right)\right\}.$$

Chú ý rằng trong số hạng thứ hai của biểu thức cuối cùng có C_M^3 tổng dạng:

$$\left(\sum_{n=0}^{N-1} b_{n+i}.b_{n+j}.b_{n+k}\right).$$

Do đó để tính số hạng thứ hai, ta ký hiệu B_3 là số các bộ ba (i, j, k), với $0 \leq i < j < k \leq M-1$, sao cho:

$$\{b_{n+i}\}.\{b_{n+j}\} = \{b_{n+k}\} \quad (3.23)$$

Tức là:

$$b_{n+i}.b_{n+j} = b_{n+k},\ \forall\ n = 0, 1, 2, ..., N\text{-}1. \tag{3.24}$$

Do tính chất của m-dãy, nên số các bộ ba (i, j, k) còn lại là $(C_M^3 - B_3)$ sẽ cho đẳng thức:

$$\{b_{n+i}\}.\{b_{n+j}\} = \{b_{n+k'}\},\ k' \notin [0, M\text{-}1],\ k' \neq k \bmod N. \tag{3.25}$$

Cũng từ tính chất tự tương quan của m-dãy,ta có:

$$\left(\sum_{n=0}^{N-1} b_{n+i}b_{n+j}b_{n+k}\right) = \begin{cases} N, & \text{nếu có (3.23)} \\ -1, & \text{nếu có (3.25)} \end{cases}. \tag{3.26}$$

Do vậy:

$$\left\{\sum_{i=0}^{M-3}\sum_{j=i+1}^{M-2}\sum_{k=j+1}^{M-1}\left(\sum_{n=0}^{N-1} b_{n+i}.b_{n+j}.b_{n+k}\right)\right\} = \left[N.B_3 + \left(C_M^3 - B_3\right)(-1)\right]. \tag{3.27}$$

Từ đó ta có:

$$S^3 = -\frac{M^3}{N} + 3!\frac{N+1}{N}.B_3, \tag{3.28}$$

ở đây B_3 là số các bộ ba (i, j, k), $0 \le i < j < k \le M\text{-}1$, sao cho

$$\{b_{n+i}\}.\{b_{n+j}\} = \{b_{n+k}\}. \tag{3.29}$$

**) Công thức tính mômen bậc 4.*

Tương tự như trên ta có:

$$\begin{aligned} S^4 &= \\ &= \frac{1}{N}\sum_{n=0}^{N-1}\left(\sum_{i=0}^{M-1} b_{n+i}\right)^4 \\ &= M(3M-2) - \frac{M(M-1)^2(M+2)}{N} + 4!\frac{N+1}{N}.B_4 \end{aligned} \tag{3.30}$$

trong đó, B_4 là số các bộ bốn (i_1, i_2, i_3, i_4), $0 \le i_1 < i_2 < i_3 < i_4 \le M$ - 1, sao cho

$$\{b_{n+i_1}\}.\{b_{n+i_2}\}.\{b_{n+i_3}\} = \{b_{n+i_4}\}. \tag{3.31}$$

*)Khái quát công thức tính mômen bậc p.

Bằng qui nạp ta có thể chỉ ra rằng các mômen S^p, $p \geq 1$, được tính bởi công thức sau:

$$\begin{aligned} S^p &= \\ &= \frac{1}{N}\sum_{n=0}^{N-1}\Big(\sum_{i=0}^{M-1} b_{n+i}\Big)^p \\ &= F(0,p,M) + \frac{1}{N}\sum_{k=1}^{p} F(k,p,M).\{-C_M^2 + (N+1).B_k\}. \end{aligned} \tag{3.32}$$

ở đây B_k là số các bộ k-tuples: (i_1, i_2, i_3,..., i_k), $0 \leq i_1 < i_2 < i_3 \ldots < i_k \leq M$-1, $k \geq 3$ sao cho

$$\{b_{n+i_1}\}.\{b_{n+i_2}\}\ldots\{b_{n+i_{k-1}}\} = \{b_{n+i_k}\}. \tag{3.33}$$

Và $B_0 = B_1 = B_2 = 0$.

Các hệ số $F(k, p, M)$ là các số hạng bậc k trong khai triển chính tắc của tổng $\Big(\sum_{i=0}^{M-1} b_{n+i}\Big)^p$ và $F(k, p, M) \neq 0$, khi cả k và p cùng chẵn hoặc cùng lẻ.

Chẳng hạn:

$F(1,1,M) = 1!$.

$F(0,2,M) = M$; $F(2,2,M) = 2!$.

$F(1,3,M) = 3M$-2; $F(3,3,M) = 3!$.

$F(0,4,M) = M(3M$-2); $F(2,4,M) = 4(3M$-4); $F(4,4,M) = 4!$.

$F(1,5,M) = 15(M$-1)^2 +1; $F(3,5,M) = 60(M$-2); F(5,5,M) = 5! .

3.2.5. Thuật toán tính B_3

Như phần trước đã chỉ ra B_3 là số các bộ ba (i, j, k) với $0 \leq i < j < k \leq M$-1, sao cho: $\{b_{n+i}\}.\{b_{n+j}\} = \{b_{n+k}\}$.

Chuyển sang ngôn ngữ của m-dãy ban đầu, hệ thức trên có nghĩa là

$a_{n+i} \oplus a_{n+j} = a_{n+k}, \forall\ n = 0, 1,..., N\text{-}1$

$\Leftrightarrow a_n = a_{n-c_j} \oplus a_{n-d_j}, \forall\ n = 0, 1,..., N\text{-}1,$

trong đó: $1 \le c_j < d_j \le M\text{-}1$.

Đẳng thức trên có nghĩa là đa thức $g(x) = 1 + x^{c_j} + x^{d_j}$, $1 \le c_j < d_j \le M\text{-}1$, là đa thức đặc trưng của dãy $\{a_i\}$.

Giả sử $f(x) = 1 + \sum_{i=1}^{r} c_i x^i$ là đa thức đặc trưng nguyên thủy sinh dãy $\{a_i\}$. Khi đó ta có $f(x)$ là ước của $g(x)$.

Như vậy, từ một bộ ba (i, j, k) với $0 \le i < j < k \le M\text{-}1$, sao cho

$$a_{n+i} \oplus a_{n+j} = a_{n+k}, \forall\ n = 0, 1,..., N\text{-}1,$$

ta tìm được một đa thức $g(x) = 1 + x^{c_j} + x^{d_j}$ thỏa mãn $f(x) \mid g(x)$, trong đó $c_j = k - j$, $d_j = k - i$, với $1 \le c_j < d_j \le M\text{-}1$. Và dĩ nhiên số bộ ba (i, j, k) khác nhau cho cùng một cặp (c, d) sẽ là $(M\text{-}d_j)$ bộ.

Ngược lại, ta có thể chứng minh rằng, nếu có tam thức:

$g(x) = 1 + x^{c_j} + x^{d_j}$, với $1 \le c_j < d_j \le M\text{-}1$, sao cho $f(x) \mid g(x)$, thì ta sẽ tìm được cả thảy $(M\text{-}d_j)$ bộ ba (i, j, k) với $0 \le i < j < k \le M\text{-}1$, sao cho:

$a_{n+i} \oplus a_{n+j} = a_{n+k}, \forall\ n = 0, 1,..., N\text{-}1.$

Vậy ta có:

$$B_3 = \sum_{j=1}^{B_3^*} (M - d_j), \tag{3.34}$$

trong đó, B_3^* là số các tam thức $g(x) = 1 + x^{c_j} + x^{d_j}$, với $1 \le c_j < d_j \le M\text{-}1$, sao cho $f(x) \mid g(x)$.

Bây giờ ta sẽ nêu thuật toán đơn giản để tính B_3^* và d_j.

Cơ sở của thuật toán

Giả sử cho *f(x)* là đa thức sinh dãy $\{a_i\}$ có bậc r. Ký hiệu

$$\{a_i\}_M = (1, \underbrace{0, 0, ..., 0}_{(r-1)\text{-bít } 0}, 1, a_{r+1}, ..., a_{M+r-1}),$$

là đoạn $(M + r)$ bít đầu tiên được sinh từ *f(x)* với trạng thái ban đầu *r*-bít là

$$(1, 0, ..., 0).$$

Giả sử $g(x) = 1 + x^{d-c} + x^d$, là tam thức với $1 \le d - c < d \le M-1$, sao cho $f(x) \mid g(x)$. Khi đó do *g(x)* cũng sinh ra dãy $\{a_i\}$ nên áp vào đoạn $\{a_i\}_M$ ta sẽ có:

$$\underbrace{(0, 0, ..., 0)}_{(r-1)\text{-bít}} \oplus (a_{c+1}, a_{c+2}, ..., a_{c+r-1}) = (a_{d+1}, a_{d+2}, ..., a_{d+r-1})$$

$$\Leftrightarrow \quad (a_{c+1}, a_{c+2}, ..., a_{c+r-1}) = (a_{d+1}, a_{d+2}, ..., a_{d+r-1}) \qquad (3.35)$$

-Ngược lại, nếu trong đoạn $\{a_i\}_M$ tồn tại *c, d* sao cho (3.24) xảy ra. Khi đó từ chỗ $\{a_i\}$ nhận *f(x)* là đa thức đặc trưng, nên ta có thể chứng minh được $\{a_i\}$ cũng nhận tam thức $g(x) = 1 + x^{d-c} + x^d$, là đa thức sinh ra nó. Và từ chỗ *f(x)* là đa thức đặc trưng tối thiểu nên suy ra $f(x) \mid g(x)$.

Từ các phân tích trên ta có: số các tam thức $g(x) = 1 + x^c + x^d$, với $1 \le c < d \le M-1$, sao cho $f(x) \mid g(x)$, đúng bằng số cặp $(r-1)$-tuples trùng nhau dạng sau:

$$(a_{c+1}, a_{c+2}, ..., a_{c+r-1}) \equiv (a_{d+1}, a_{d+2}, ..., a_{d+r-1}).$$

Do vậy ta có thuật toán để tìm B_3^* và d_j, tức là tìm B_3 như sau:

Thuật toán tìm B_3

Bước 1: Cho f(x) là đa thức nguyên thủy bậc r.

Sinh ra đoạn $\{a_i\}_M$ với trạng thái ban đầu r-bít (1, 0, 0, ..., 0).

Đặt $B_3^* := 0$, $B_3 := 0$.

Bước 2: Lần lượt so sánh các (r-1)-tupels trong đoạn $\{a_i\}_M$. Nếu xảy ra đẳng thức

$(a_{c+1}, a_{c+2}, ..., a_{c+r-1}) \equiv (a_{d+1}, a_{d+2}, ..., a_{d+r-1})$.

thì tăng B_3^* lên một đơn vị, và tăng B_3 lên (M-d) đơn vị.

Tức là ta có đoạn câu lệnh

For c:=1 to (M-2)

do begin

for d:= (c+1) to (M-1)

if $(a_{c+1}, a_{c+2}, ..., a_{c+r-1}) \equiv (a_{d+1}, a_{d+2}, ..., a_{d+r-1})$.

then $B_3^* := B_3^* + 1$, $B_3 := B_3 + (M-d)$.

end.

End.

Bước 3: In ra các giá trị B_3^* và B_3.

3.2.6. Thuật toán tính B_4

Tương tự như phần lập công thức tính B_3 ta cũng có công thức để tính B_4 như sau:

$$B_4 = \sum_{j=1}^{B_4^*} (M - d_j). \qquad (3.36)$$

Trong đó B_4^* là số các đa thức $g(x) = 1 + x^{b_j} x^{c_j} + x^{d_j}$, với $1 \le b_j < c_j < d_j \le M-1$, sao cho $f(x) \mid g(x)$.

Giả sử $f(x)$ là đa thức nguyên thủy sinh dãy $\{a_i\}$ như trên. Ký hiệu

$\{a_i\}_M = (0, \underbrace{1, 1, ..., 1}_{\text{r-bít 1}}, 0, a_{r+2}, a_{r+3}, ..., a_{r+M-1})$,

là ($M+r$) -số hạng đầu tiên của dãy $\{a_i\}$ sinh bởi $f(x)$ với trạng thái ban đầu r-bit là (1, 1, ..., 1).

Tương tự như phần trước ta có tứ thức $g(x) = 1 + x^{d-c} + x^{d-b} + x^d$, với $1 \leq b < c < d \leq M$-1, sẽ thỏa mãn $f(x) \mid g(x)$ khi và chỉ khi trong đoạn $\{a_i\}_M$ trên tìm được bộ ba r-tupels thỏa mãn:

$(a_{b+1}, a_{b+2}, ..., a_{b+r}) \oplus (a_{c+1}, a_{c+2}, ..., a_{c+r}) \oplus (a_{d+1}, a_{d+2}, ..., a_{d+r}) \equiv (1, 1, ..., 1)$.

Từ đó ta có thuật toán đơn giản để tính B_4 như sau.

* Thuật toán tìm B_4.

Bước 1: Cho f(x) là đa thức nguyên thủy bậc r.

Sinh ra đoạn $\{a_i\}_M$ với trạng thái ban đầu r-bít (1, 1, 1, ..., 1).

Đặt $B_4^* := 0$, $B_4 := 0$.

Bước 2: For b:= 1 to M-3

do begin for c:= b+1 to M-2

do begin for d:= c+1 to M-1

If $(a_{b+1}, a_{b+2}, ..., a_{b+r}) \oplus (a_{c+1}, a_{c+2}, ..., a_{c+r}) \equiv (a_{d+1} \oplus 1, a_{d+2} \oplus 1, ..., a_{d+r} \oplus 1)$

Then

$B_4^* := B_4^* + 1$, $B_4 := B_4 + (M-d)$.

end. end.

End.

Bước 3: In ra các giá trị B_4^* và B_4.

3.2.6. Nhận xét về tương quan địa phương của m-dãy

Xét m-dãy $\{a_i\}$ sinh bởi đa thức nguyên thủy $f(x)$ bậc r. Từ (3.21) nếu lấy giá trị M sao cho $r < M << N$, ở đây $N = 2^r - 1$ là chu kỳ của m-dãy. Khi đó ta có

thể coi $S^1 = -M/N \approx 0$. Do đó ta sẽ có các công thức xấp xỉ cho các mômen trung tâm bậc 1, 2, 3, 4 của phân bố trọng số các đoạn con M-bit của m-dãy $\{a_i\}$ như trong Bảng 3.1.

Bảng 3.1 Mô men trung tâm của phân bố trọng số các đoạn con

Mô men bậc p	Dãy ngẫu nhiên lý tưởng; W_c^p	m-dãy chu kỳ N; $W_c^p = S_c^p / (-2)^p$
1	0	≈ 0
2	M/4	$(M/4)[1-(M-1)/N]$ $\approx M/4$
3	0	$-(M^3/8N) + (3/4).[(N+1)/N].B_3$ $\approx (3/4)B_3$
4	$(3M^2 - 2M)/16$	$(3M^2-2M)/16-[M(M-1)^2(M+2)]/16N+$ $+(3/2)\,[(N+1)/N].B_4$ $\approx(3M^2-2M)/16 +(3/2)B_4$

Từ Bảng 3.1, so sánh các mômen trung tâm bậc 1, 2, 3, 4 của một m-dãy với các mômen trung tâm tương ứng của dãy ngẫu nhiên lý tưởng, ta thấy rằng một m-dãy có thể xem là có phân bố giá trị tương quan địa phương ngẫu nhiên đối với các đoạn độ dài M, nếu có $B_3 = B_4 = 0$ với các tham số $[M, f(x)]$ tương ứng. Một nhận xét nữa là các mômen bậc 1, 2 (theo các công thức (3.21), (3.22)) không phụ thuộc vào các m-dãy cụ thể mà chỉ phụ thuộc vào hai tham số độ dài M của các đoạn con đang xét và chu kỳ N của m-dãy. Các m-dãy cùng chu kỳ chỉ phân biệt nhau (trên khía cạnh tương quan địa phương) khi ta xét tới các mômen bậc 3, 4 và cao hơn.. Điều này cũng nói lên rằng, mỗi m-dãy có tập M^* khác nhau (tuy cùng lực lượng), m-dãy nào có tập M^* làm cho $B_3 = B_4 = 0$, ta hy vọng rằng m-dãy đó sẽ có dáng điệu ngẫu nhiên hơn trong tập hợp các m-dãy cùng chu kỳ đang xét.

Tóm lại, trong quá trình đánh giá các tính chất của dãy giả ngẫu nhiên, ta cần quan tâm tới tính chất ngẫu nhiên địa phương của chúng, liên quan tới tính chất ngoại suy tuyến tính và tính tương quan giữa các đoạn con một dãy giả ngẫu nhiên...

3.3. Đề xuất thuật toán sinh dãy giả ngẫu nhiên phi tuyến lồng ghép với bậc lớn

3.3.1 Các khó khăn khi sinh dãy giả ngẫu nhiên phi tuyến lồng ghép với bậc lớn

Đi sâu phân tích phương pháp sinh dãy lồng ghép, ta có thể thấy phần công việc lớn nhất cần thực hiện trong thực hành là việc xác định giá trị ban đầu của các dãy con thành phần. Cả 4 phương pháp xác định giá trị ban đầu đã nghiên cứu trước đây đều yêu cầu số bước tính toán rất lớn khi bậc của dãy ban đầu tăng lên.

Trong phương pháp sử dụng biến đổi d, ta cần phải tính toán với một đa thức có bậc p^{mT}

Với phương pháp sử dụng hàm vết cần thực hiện tính giá trị hàm vết

$$Tr_n^m(x) = \sum_{k=0}^{\frac{n}{m}-1} x^{q^{mk}}. \quad (3.37)$$

Khi giá trị n (là của bậc của dãy ban đầu) tăng lên, giá trị của m cũng tăng lên tương ứng. Điều này khiến số mũ của thành phần x là q^{mk} tăng lên tới mức không khả thi trong thực tế.

Trong phương pháp tính toán trực tiếp chỉ yêu cầu sinh ra m dòng của ma trận chứa toàn bộ chu kỳ từ dãy ban đầu. Với các tham số đã nêu ở trên, ta cần sinh ra $m.T$ phần tử.

Với phương pháp tính toán sử dụng phép phân rã theo bước ta cũng cần tính toán giá trị đa thức d^T trên trường GF(p^n).

Giả sử ta chọn n=24, q=7, m=8, khi đó T = 3,32.10^{13} và các yêu cầu tính toán đã nêu trở thành không khả thi về tính toán (computational infeasibility).

Một giới hạn khác cần quan tâm là về không gian lưu trữ. Phương pháp sử dụng biến đổi d và phương pháp tính toán trực tiếp đều hướng đến việc tìm ra toàn bộ tập thứ tự lồng ghép I_P^T, khi giá trị T tăng lên thì yêu cầu về không gian lưu trữ

cần thiết để lưu tập thứ tự lồng ghép này cũng trở lên khó khả thi. Với phương pháp phân rã theo bước, ta đã xác định chỉ tính một phần cần thiết của I_P^T, còn phương pháp sử dụng hàm vết cũng có thể hiệu chỉnh để chỉ cần tính một phần cần thiết của I_P^T.

Các yêu cầu của dãy lồng ghép áp dụng trong kỹ thuật mật mã.

Từ hai phương pháp phân tích dãy giả ngẫu nhiên đã đề cập trong phần 3.1 và 3.2, ta thấy rằng khi áp dụng thuật toán tổng hợp độ phức tạp tuyến tính cho đầu ra của dãy lồng ghép và phi tuyến lồng ghép, trong hầu hết các trường hợp ta đều nhận được kết luận là dãy được sinh ra bởi m-dãy thành phần có bậc m. Chỉ khi đoạn dữ liệu đem phân tích có sự tiếp nối giữa hai dãy con, khi này độ phức tạp tuyến tính được tăng lên, song không vượt quá độ phức tạp chung của dãy gốc có bậc n. Việc áp dụng bài toán tương quan địa phương cũng đưa ra kết quả tương tự. Như vậy để có thể áp dụng dãy lồng ghép trong kỹ thuật mật mã để bảo mật thông tin, ta cần thiết kế bộ tạo dãy sao cho bậc của dãy con thành phần đã thỏa mãn các yêu cầu về bảo mật. Đây cũng là lý do tác giả luận án không tiếp tục phát triển hướng nghiên cứu về dãy lồng ghép đa cấp, do dãy con thành phần ở mức sau cùng của dãy lồng ghép đa cấp có bậc rất nhỏ so với bậc của dãy gốc, làm suy giảm tính an toàn về mật mã của dãy đầu ra.

Với các yêu cầu của kỹ thuật mật mã đã trình bày trong phần 1.2.2, ta thấy rằng yêu cầu phổ biến với các m-dãy thành phần là độ lớn bậc $n \geq 128$.

Vì m là ước của n nên:

$$m \leq n/2. \tag{3.38}$$

Do giá trị bước lồng ghép được tính là:

$$T = \frac{L}{N} = \frac{p^n - 1}{p^m - 1}. \tag{3.39}$$

Với m, n đủ lớn ta có thể lấy xấp xỉ:

$$T \approx p^{n-m}. \tag{3.40}$$

Để thỏa mãn điều kiện $n \geq 128$ như đã nêu ta cần có:

$T \geq p^{n/2}$ hay $T \geq p^{64}$. (3.41)

Với yêu cầu của bước lồng ghép nêu trên cả bốn phương pháp sinh dãy phi tuyến lồng ghép đã trình bày ở trên đều khó có thể áp dụng trong thực tế.

3.3.2 Thuật toán sinh dãy giả ngẫu nhiên phi tuyến lồng ghép với bậc lớn

Để có thể sinh dãy giả ngẫu nhiên phi tuyến lồng ghép với bậc đủ lớn thỏa mãn yêu cầu của kỹ thuật mật mã, ta cần giải quyết việc tiền xử lý để xây dựng tập thứ tự lồng ghép I_P^T. Trong phần tiếp theo tác giả luận án sẽ trình bày thuật toán để tính toán giá trị đa thức d^T trên trường GF(p^n) trong trường hợp T rất lớn, để có thể áp dụng phương pháp sinh dãy phi tuyến lồng ghép bằng kỹ thuật phân rã theo bước.

Thuật toán tiền xử lý tìm thứ tự lồng ghép

Theo công thức sinh m-dãy, trạng thái của m-dãy sau T bước bắt đầu từ giá trị khởi đầu S(0) là:

$$S_{(T)} = \frac{S_{(0)} \times d^T}{g(d)} \quad (3.42)$$

Ta chú ý tới giá trị tham số T:

$T = \frac{N}{L}$ với $L = p^m$-1 và $N = p^n$-1. (3.43)

Chú ý là $n = m*l$, vậy ta có thể viết:

$N = p^{m*l}$-1 hay $N = (p^m)^l - 1$. (3.44)

Để đơn giản, đặt $Q = p^m$, ta có thể viết

$T = \frac{Q^l - 1}{Q - 1} = Q^{l\text{-}1} + Q^{l\text{-}2} + \ldots + Q + 1$. (3.45)

Nói một cách khác, nếu biểu diễn giá trị T theo cơ số Q là

(l số 1)

$T = 111 \ldots 111_Q$. (3.46)

Nếu biểu diễn T theo cơ số p, biểu diễn của T cũng chỉ có l số 1, chen giữa các số 1 là m-1 số 0:

$$T = 100...00100...00100 \ldots 001_p . \quad (3.47)$$

Ta sẽ tìm cách tính nhanh giá trị đa thức

$$U_T(d) = \frac{d^T}{g(d)} . \quad (3.48)$$

Ta có:

$$U_1(d) = d , \quad (3.49)$$

$$U_Q(d) = \frac{d^Q}{g(d)} , \quad (3.50)$$

$$U_{Q^k}(d) = \frac{\prod_{j=1}^{Q} U_{Q^{k-1}}(d)}{g(d)} , \quad (3.51)$$

$$U_T(d) = \prod_{k=0}^{l-1} U_{Q^k}(d). \quad (3.52)$$

Vì $Q = p^m$ có thể lên tới hàng tỷ hoặc thậm chí cao hơn khi bậc của đa thức ban đầu tăng lên, ta cần có phương pháp hiệu quả để tính được các công thức (3.50), (3.52).

Để tính $U_Q(d)$ ta sẽ tính lần lượt từng bước như sau. Trước hết tính trực tiếp đa thức:

$$U_p(d) = \frac{d^p}{g(d)} . \quad (3.53)$$

Chú ý rằng:

$$d^{p^k} = d^{p^{k-1} \times p} = \left(d^{p^{k-1}}\right)^p . \quad (3.54)$$

Nên ta có:

$$U_{p^k}(d) = \left(U_{p^{k-1}}(d)\right)^p . \quad (3.55)$$

Ta có thể tính $U_{p^k}(d)$ theo công thức sau:

$$U_{p^k}(d) = \frac{\prod_{j=1}^{p} U_{p^{k-1}}(d)}{g(d)}, \quad (3.56)$$

$$U_Q(d) = U_{p^m}(d) . \quad (3.57)$$

Để tính $U_{Q^k}(d)$ từ $U_{Q^{k-1}}(d)$ Ta sẽ sử dụng phương pháp tương tự.

Chú ý rằng $U_{p^0 Q^{k-1}}(d) = U_{Q^{k-1}}(d)$ nên ta có:

$$U_{p^i Q^{k-1}}(d) = \frac{\prod_{j=1}^{p} U_{p^{i-1} Q^{k-1}}(d)}{g(d)}, \quad (3.58)$$

$$U_{Q^k}(d) = U_{p^m Q^{k-1}}(d) . \quad (3.59)$$

Cuối cùng, sử dụng các giá trị tính bởi (3.59) ta tính được đa thức (3.51).

Với giá trị khởi đầu $S_{(0)}$, sử dụng (3.42) ta tính được $S_{(T)}$. Tiếp tục sử dụng (3.42) song thay $S_{(0)}$ bằng $S_{(T)}$ ta sẽ có $S_{(2T)}$ … từ đó tính được *m* bộ trạng thái của dãy ban đầu ứng với m phần tử bắt đầu các cột trong ma trận. Lấy *m* phần tử đầu tiên trong bộ trạng thái này chính là giá trị *m* ô trong cột đầu tiên trong ma trận.

Có được *m* bộ trạng thái này, ta có thể sử dụng (3.42) để tính *m* bộ trạng thái ở cột tiếp theo… Tùy vào kích thước yêu cầu của dãy đầu ra mà ta sẽ tính số bộ trạng thái khởi đầu các cột một cách tùy ý, hoặc cũng có thể tính trạng thái khởi đầu cho một cột mỗi khi cần dùng tới cột đó.

Phương pháp bình phương và nhân

Có một cách hiệu quả hơn để tính các công thức (3.50) và (3.52). Khi ta có biểu diễn của p dưới dạng nhị phân thành tập các bit $\{p_i\}$ với i=0..r, sau đó sử dụng phương pháp bình phương và nhân để tính đa thức kết quả của công thức (3.50)như sau:

Đặt $U^* = 1$

$V_{tmp} = U_{p^k}(d)$

Với i chạy từ 0 tới r, ta lần lượt tính

Nếu $p_i = 1$

$U^* = U^* \; x \; V_{tmp}$

$$V_{tmp} = \frac{(V_{tmp})^2}{g(d)}$$

Sau r bước ta có $U_{p^k}(d) = \mathrm{U}^*$.

Lưu đồ thuật toán tính $U_{p^k}(d)$ được thể hiện trong hình 3.2.

Bằng phương pháp tương tự ta tính được đa thức kết quả của công thức (3.52) như sau:

Đặt $U^* = 1$ *và* $V_{tmp} = U_{p^{i-1}Q^{k-1}}(d)$

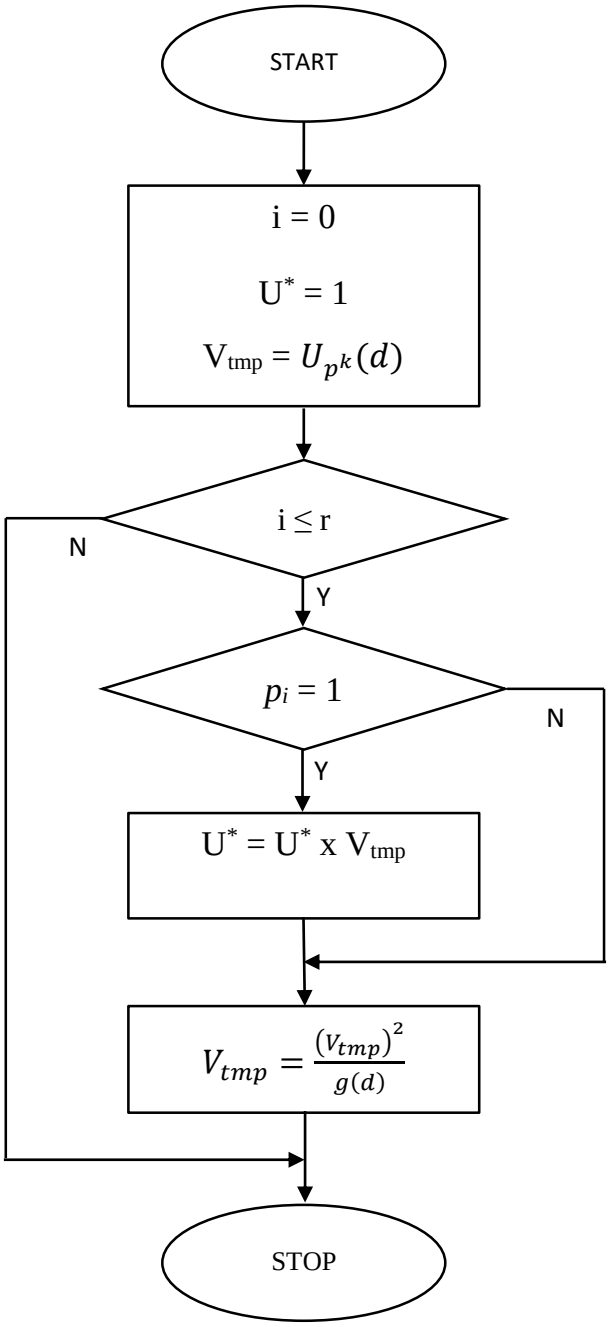
Với i chạy từ 0 tới r, ta lần lượt tính

Nếu $p_i = 1$

$U^* = U^* \; x \; V_{tmp}$

$$V_{tmp} = \frac{(V_{tmp})^2}{g(d)}$$

Sau r bước ta có $U_{p^iQ^{k-1}}(d) = \mathrm{U}^*$



Hình 3.2 Lưu đồ thuật toán tính $\boldsymbol{U_{p^k}(d)}$

Thuật toán sinh dãy phi tuyến lồng ghép dựa trên m-dãy

Khi có trạng thái khởi đầu của một cột trong ma trận lồng ghép, ta sử dụng phần sau của phương pháp trình bày trong mục 2.4.3 để xác định các giá trị còn lại của cột.

Với thuật toán tiền xử lý nêu trên, ta tìm ra được giá trị m phần tử đầu tiên của cột thứ nhất trong ma trận lồng ghép (là m bit số 0 trong m bộ trạng thái). Từ các giá trị này ta có thể xây dựng phần đầu tiên của dãy lồng ghép bằng dãy con đầu tiên.

Để tiếp tục xây dựng các phần tiếp theo của dãy lồng ghép, ta tìm giá thứ tự lồng ghép tiếp sau đó bằng cách sử dụng từng trạng thái trong $S_{(kT)}$ để xác định các trạng thái trong $S_{(kT+1)}$ qua công thức sinh m-dãy, từ đó có được giá trị khởi đầu của cột n+1. Như vậy ta có thể xây dựng dãy lồng ghép có độ dài bất kỳ mà không cần tính trước toàn bộ bảng thứ tự lồng ghép.

Để sinh dãy phi tuyến lồng ghép, ta áp dụng các bước tương tự như với dãy lồng ghép đối với dãy đầu vào thứ nhất, song riêng việc sinh ra giá trị các cột lại sử dụng dãy con từ dãy đầu vào thứ hai.

3.3.3 Đánh giá độ phức tạp của thuật toán sinh dãy giả ngẫu nhiên phi tuyến lồng ghép với bậc lớn

Để tính được $U_Q(d)$ ta cần (m-1).(p-1) phép nhân đa thức trên trường GF(p^n) để tính công thức (3.41). Trong trường hợp tính bằng phương pháp bình phương và nhân, số phép nhân cần tính là (m-1).$\log_2 p$ (Giả sử thời gian tính phép bình phương và phép nhân đa thức là tương đương nhau)

Để tính được $U_{Q^k}(d)$ từ $U_{Q^{k-1}}(d)$ ta cần tính các phép nhân đa thức trên trường GF(p^n). Xét trường hợp sử dụng phương pháp bình phương và nhân thay cho (19) và (21) thì số phép nhân đa thức là:

$$v_{mulq} = (m\text{-}1).\log_2 p \ . \qquad (3.60)$$

Để tính được $U_T(d)$ theo (3.36) ta cần sử dụng thêm (l-1) phép nhân đa thức. Tổng số phép nhân đa thức trên trường GF(p^n) cần tính là:

$$v_q = (l\text{-}1).(m\text{-}1).\log_2 p + (l\text{-}1). \qquad (3.61)$$

Chú ý là khi n $\rightarrow \infty$, m và l đều có cỡ tương đương n. Ta có thể coi $log_2 p$ là hằng số, xét phép nhân đa thức trên trường GF(p^n) có độ phức tạp tính toán là n thì độ phức tạp tính toán của v_q tiệm cận với O(n^3).

So sánh với phương pháp bình phương và nhân áp dụng trực tiếp cho giá trị số mũ T bằng một thuật toán tương tự như *Thuật toán 1*. Ta sẽ biểu diễn giá trị của T thành $log_2 T$ bit $\{t_i\}$ để áp dụng bình phương và nhân.

Số phép bình phương đa thức cần tính là:

$$v_{mul2} = log_2 T = (n\text{-}m)*log_2 p = m.(l\text{-}1).log_2 p \qquad (3.62)$$

Số phép nhân cần tính sẽ có giá trị trung bình là $v/2$ (do phân bố bit 0/1 trong biểu diễn nhị phân của T là đều nhau).

Vậy tổng số phép nhân đa thức cần tính là

$$v_2 = {}^3\!/_2 \, . \, v_{mul2} \, , \qquad (3.63)$$

$$v_2 = {}^3\!/_2 \; m.(l\text{-}1).log_2 p \, . \qquad (3.64)$$

Phương pháp tính theo biểu diễn cơ số p có được lợi thế hơn phương pháp biểu diễn nhị phân bởi vì trong biểu diễn cơ số p của T có rất nhiều phần tử bằng 0 theo (3.47), trong khi biểu diễn nhị phân của T không có được lợi thế này.

Để so sánh cụ thể số bước tính toán giữa hai trường hợp tính toán với cơ số p và tính toán theo phương pháp bình phương và nhân trực tiếp (trên biểu diễn cơ số 2 của T) ta tính toán các giá trị cụ thể của vq và v2 trong một số trường hợp như trong bảng 3.2. Ta tính chính xác giá trị T, sau đó chuyển đổi sang nhị phân và đếm số bit để có $\boldsymbol{n_{bit1\text{-}2}}$, do đó trong một số trường hợp số bit 1 không phải là $log_2 T/2$ như trong (3.46). Ví dụ ở dòng 4, ta có $T = 678\,223\,072\,850_{10}$, biểu diễn nhị phân là:

T = 10011101111010010011111011001110010100102 .

Như vậy $\boldsymbol{n_{bit1\text{-}2}}$ nhận giá trị 23 thay vì $log_2 T/2 = 20$.

Bảng 3.2 Số bước tính toán tiền xử lý cho dãy lồng ghép

STT	p	n	m	v_{mulq}	$n_{bit1\text{-}q}$	v_q	T	v_{mul2}	$n_{bit1\text{-}2}$	v_2
1	2	24	8	17	3	**20**	65 793	17	3	**20**
2	3	24	8	32	3	**35**	43 053 283	26	12	**38**
3	7	24	8	48	3	**51**	33 232 936 334 403	45	24	**69**
4	7	28	14	42	2	**44**	678 223 072 850	40	23	**63**
5	13	12	3	36	4	**40**	10 609 328 380	34	17	**51**
6	13	18	9	36	2	**38**	10 604 499 374	34	17	**51**
7	29	12	3	45	4	**49**	14 507 740 823 580	44	19	**63**
8	31	12	3	45	4	**49**	26 440 509 694 144	45	12	**57**

So sánh 2 bảng trên, ta thấy phương pháp tính toán với cơ số p có hiệu quả tốt hơn phương pháp tính toán bình phương và nhân trực tiếp .

Với cơ số $p = 2$, hai phương pháp có số bước giống nhau do cùng là tính toán trên cơ số 2.

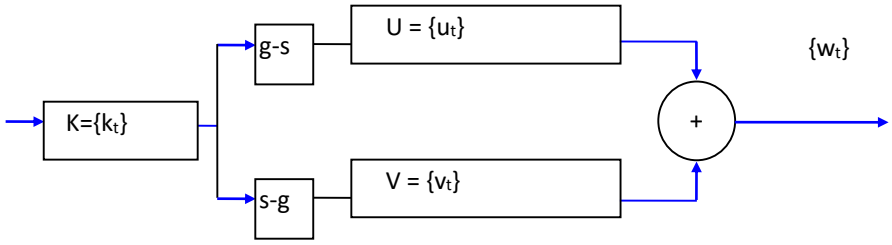
Với cơ số $p = 3$ là một giá trị rất nhỏ, phương pháp tính toán với cơ số p chỉ giúp tăng một phần nhỏ hiệu quả so với phương pháp tính toán bình phương và nhân trực tiếp.

Trong trường hợp cơ số p có giá trị lớn, phương pháp tính toán trên cơ số p có hiệu quả tốt hơn hẳn so với phương pháp tính toán bình phương và nhân trực tiếp, cụ thể là số bước tính toán ít hơn khoảng 25% .

3.4 Đề xuất phương pháp sinh dãy giả ngẫu nhiên an toàn sử dụng dãy phi tuyến lồng ghép

3.4.1 Bộ tạo dãy luân phiên phi tuyến lồng ghép

Sử dụng bộ tạo dãy luân phiên đã trình bày trong phần 1.3.3, trong đó ta chọn dãy thành phần thứ nhất là một dãy phi tuyến lồng ghép, trong khi dãy thành phần thứ hai vẫn giữ nguyên là m-dãy và dãy điều khiển vẫn giữ nguyên dãy



Hình 3.3 Mô hình bộ tạo dãy luân phiên phi tuyến lồng ghép

D'Brujin.

Giả sử: $K=\{ k_t\}_{t\geq 0}$ là dãy D' Bruijn bậc k; $U=\{u_t\}$ là m-dãy bậc L, $V=\{v_t\}$ là dãy phi tuyến lồng ghép bậc M, trong đó L và M nguyên tố cùng nhau.

3.4.2 Các tính chất của bộ tạo dãy luân phiên phi tuyến lồng ghép

Từ các phân tích về tính chất của dãy luân phiên trình bày trong phần 1.3.3, ta có thể áp dụng để đưa ra các tính chất tương ứng của dãy luân phiên phi tuyến lồng ghép như sau:

Tính chất 1: (*Chu kỳ và độ phức tạp tuyến tính*)

Ta biết rằng dãy phi tuyến lồng ghép có chu kỳ là 2^n-1.

Vậy chu kỳ của dãy đầu ra vẫn giống như trường hợp dãy luân phiên là:

$$K.(2^M-1)(2^N-1) . \qquad (3.65)$$

Riêng tính tương quan của dãy luân phiên phi tuyến lồng ghép có một chút thay đổi so với dãy luân phiên. Do tính tương quan địa phương của dãy phi tuyến lồng ghép phụ thuộc phần lớn vào dãy con sinh được sử dụng trong đó, vì thế tính tương quan của dãy luân phiên phi tuyến lồng ghép sẽ được tính bằng

$$C(\tau) = \frac{t_0.pq^* - t_1.p - t_2.q^* + t_3}{2^k.pq}. \quad (3.66)$$

ở đây t_i, i=0..3, là chỉ số trùng giữa các pha thứ 0 và thứ τ của dãy, p là chu kỳ của dãy U, q* là chu kỳ của dãy con của dãy phi tuyến lồng ghép

Tính chất 2: (*lực lượng của bộ tạo dãy)*

So với dãy luân phiên, bộ tạo dãy luân phiên phi tuyến lồng ghép có thêm tham số đầu vào *m* và m-dãy thứ hai để xây dựng lên dãy phi tuyến lồng ghép. Vì thế lực lượng của bộ tạo dãy luân phiên phi tuyến lồng ghép trở thành:

$$K_w = M.2^{2^{k-1}-k}.\frac{\Phi\left(2^L-1\right)}{L}.\frac{\Phi^2\left(2^M-1\right)}{M^2} \quad (3.67)$$

Lực lượng các bộ tạo dãy trở lên lớn hơn rất nhiều so với bộ tạo dãy luân phiên ban đầu.

Các tính chất còn lại : Phân bố tần số các bộ r-tupe, tính chất tương quan được giữ nguyên như đối với dãy luân phiên, do dãy phi tuyến lồng ghép giữ được các tính chất đó từ m-dãy tương ứng.

3.5 Kết luận chương 3

Trong chương này tác giả đã đề xuất một thuật toán mới, có thể coi là một phương pháp mới để sinh dãy phi tuyến lồng ghép với bậc lớn. Việc đánh giá độ phức tạp tính toán cho thấy thuật toán này có độ phức tạp tính toán ở mức hàm mũ, chính xác là cỡ $O(n^3)$ so với bậc *n* của đa thức sinh m-dãy. Bằng cách khai thác một đặc điểm của tham số của dãy lồng ghép, thuật toán này có lợi thế lớn hơn thuật toán bình phương và nhân thông thường. Thuật toán này giúp cho việc sinh dãy phi tuyến lồng ghép trở lên hiệu quả trong thực hành khi bậc của dãy lớn lên theo các yêu cầu trong kỹ thuật mật mã.

Qua phân tích một số phép tấn công phân tích mã đối với dãy giả ngẫu nhiên dựa trên m-dãy, tác giả đã đề xuất bộ tạo dãy luân phiên phi tuyến lồng ghép là kết quả việc ứng dụng dãy phi tuyến lồng ghép vào bộ tạo dãy luân phiên, từ đó đạt được các tính chất mật mã tốt có thể áp dụng trong thực tế.

KẾT LUẬN

Trong phạm vi luận án, tác giả đã nghiên cứu về cơ sở toán học, lý thuyết xây dựng dãy giả ngẫu nhiên theo phương pháp phi tuyến lồng ghép dựa trên m-dãy, cùng với việc nghiên cứu về các phương pháp đánh giá an toàn dãy giả ngẫu nhiên trên khía cạnh mật mã. Từ các nghiên cứu đó, tác giả đã phân tích và đưa ra thuật toán hiệu quả để thực hiện sinh dãy phi tuyến lồng ghép với bậc lớn cùng với các phân tích về hiệu quả lý thuyết cũng như các thử nghiệm kiểm chứng trên máy vi tính. Trong quá trình thực hiện luận án, tác giả đã có một số đóng góp khoa học mới, cụ thể như sau:

(i) Đề xuất một giải pháp sinh dãy phi tuyến lồng ghép dựa trên kỹ thuật phân rã theo bước và kỹ thuật tính một phần thứ tự lồng ghép. Giải pháp này có thể ứng dụng trong cài đặt thực tế để sinh ra một đoạn có kích thước tùy ý của dãy phi tuyến lồng ghép.

(ii) Đề xuất một thuật toán hiệu quả để sinh dãy phi tuyến lồng ghép với bậc lớn, phân tích đánh giá thuật toán đã đề xuất về độ phức tạp tính toán, độ phức tạp lưu trữ và kết quả tính toán thực nghiệm. Độ phức tạp tính toán của thuật toán tiệm cận với $O(n^3)$ với n là bậc của đa thức sinh m-dãy. Bằng cách khai thác một đặc điểm của tham số của dãy lồng ghép, thuật toán này có lợi thế lớn hơn so với thuật toán bình phương và nhân thông thường.

Với những đóng góp khoa học nêu trên, luận án là cơ sở để tác giả có thể đề xuất một hệ mã dòng có thể ứng dụng trong thực tế đáp ứng nhu cầu bảo mật thông tin trong Ban Cơ yếu. Ngoài việc ứng dụng dãy phi tuyến lồng ghép trong kỹ thuật mật mã, còn rất nhiều lĩnh vực kỹ thuật có thể ứng dụng dãy phi tuyến lồng ghép như một bộ tạo dãy giả ngẫu nhiên với các mục đích khác nhau.

Các vấn đề cần tiếp tục nghiên cứu

Việc đề xuất một thuật toán mật mã mới cần phải xem xét rất kỹ về tính an toàn của thuật toán trên nhiều khía cạnh trước khi có thể đưa vào sử dụng thực tế, cần có các nghiên cứu sâu về việc phân tích mã đối với dãy lồng ghép và phi tuyến lồng ghép, cũng như dãy luân phiên phi tuyến lồng ghép

Một công việc khác cần tiếp tục nghiên cứu là giải pháp để cài đặt hiệu quả các dãy trên GF(p^n) với số p nguyên tố lớn (p>2) trên cả hai môi trường: phần mềm máy tính và các thiết bị xử lý trực tiếp bằng phần cứng. Ta cũng cần nghiên cứu về việc sử dụng hiệu quả dãy đầu ra trên GF(p^n), có thể là một phương pháp chuyển đổi dữ liệu giữa hệ q-phân và hệ nhị phân.

DANH MỤC CÁC CÔNG TRÌNH ĐÃ CÔNG BỐ CỦA LUẬN ÁN

[J1] Hieu Le Minh, **Truong Dang Van**, Binh Nguyen Thanh and Quynh Le Chi, "Design and Analysis of Ternary m-sequences with Interleaved Structure by d-Transform", Journal of Information Engineering and Applications, vol.5, no.8, pp.93-101, 2015.

[J2] **Truong Dang Van**, Quynh Le Chi, "Applying M-sequences Decimation to Generate Interleaved Sequence", Journal of Science and Technology on Information security, No 2.CS (14) 2021, pp. 85-88

[J3] **Đặng Vân Trường, "**Một phương pháp hiệu quả để sinh dãy giả ngẫu nhiên kiểu lồng ghép phi tuyến với bậc lớn**",** Tạp chí Khoa học Công nghệ Thông tin và Truyền thông (Journal of Science and Technology on Information and Communications), 2021

TÀI LIỆU THAM KHẢO

Tài liệu tiếng Việt

1) Bùi Lai An, "Về một cấu trúc tổng quát của mã tựa ngẫu nhiên phi tuyến đa cấp – đa chiều theo kiểu lồng ghép", Luận án tiến sĩ kỹ thuật, Học viện CN BCVT, 2012.

2) T. V. Trường và L. Đ. Tân, "Nghiên cứu về m-dãy trong các bộ tạo dãy giả ngẫu nhiên", Tạp chí Nghiên cứu KHKT&CN quân sự, Trung tâm KHKT&CN quân sự, Bộ Quốc phòng, số 6, 2004, trang 61-66.

3) T. V. Trường, "Một số tính chất địa phương của m-dãy", Tạp chí KHKTMM - Ban Cơ yếu Chính phủ, số 2 -1993, trang 31-33.

Tài liệu tiếng Anh

4) M. Antweiler, “Cross-correlation of p-ary GMW sequences”, IEEE Trans Inform. Theory, vol 40, pp. 1253-1261, 1994.

5) L.D. Baumert, Cyclic Difference Sets, ser. Lecture Notes in Mathematics. Springer-Verlag, 1971.

6) E.R. Berlekamp, “*Algebric Coding Theory”*, New York, McGraw-Hill, 1968.

7) S. Boztas and P. V. Kumar: Binary Sequences with Gold-Like Correlation but Lager Linear Span. IEEE Transaction on Information Theory, Vol 40, No.2, March-1994, pp 532-537.

8) G. Cattaneo, G. De Maio, and U. F. Petrillo. “Security issues and attacks on the GSM standard: a review”. Journal of Universal Computer Science, vol. 19, no. 16, pp. 2437–2452, 2013.

9) S.D Cardell, A. Fúster-Sabater and V. Requena, “Interleaving Shifted Versions of a PN-Sequence”, Mathematics 9(687), 2021.

10) A. Chang, P. Gaal, S.W. Golomb, G. Gong, “On a conjectured ideal autocorrelation sequence and a related triple-error correcting cyclic code, IEEE Trans Inform. Theory, vol. 46 no. 2 , pp. 680-687, 2000.

11) C. Ding, T. Helleseth and K.Y. Lam, “Several classes of binary sequences with three-level autocorrelation”, IEEE Trans Inform. Theory, vol. 45 no. 7, pp. 2606-2612, 1999.

12) R.G. Gallager, *Galois Field*, MIT, 1992.

13) R.A. Games, “Crosscorelation of m-sequences and GMW-sequences with the same primitive polynomial”, Discrete Applied Mathematics, vol.12, pp. 139-146, 1985.

14) J. D. Gibson, “Challenges in speech coding research”. Ogunfunmi, T., Togneri, R., Narasimha, M.S. (eds.) “Speech and Audio Processing for Coding, Enhancement and Recognition”, pp. 19–39. Springer, Berlin 2015.

15) A. Gill, “Linear sequential circuits:, Mc Grawhill, Newyork, 1996.

16) R.G Gitlin & J. F. Hayer, "Timming recovery and scramblers in data transmission, Bell Syst Tech Journal", vol 54, no3, pp 589-593, March 1975.

17) X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in Proc. AISTATS, 2010, pp. 249–256.

18) J. D. Golic and R. Menicocci, "Edit Distance Correlation Attack on the Alternating Step Generator", CRYPTO' - 97, pp 499-512, 1997.

19) R. Gold, “Optimal binary sequences for spread spectrum multiplexing”. IEEE Transaction on Information Theory, Vol IT-13, pp 154-156, 1967.

20) R. Gold, “Maximal recursive sequences with 3-value recursive cross-correlation functions”, IEEE Trans Inform. Theory, vol 14, pp. 154 -156, 1968.

21) S.W. Golomb, "*Shift Register Sequences*", San Francisco, Holden-Day, 1967.

22) S. W. Golomb and G. Gong, "Signal Design for Goog Correlation for Wireless Communication", *Cryptography and Radar*, Cambrigde University Press, 2005.

23) G. Gong: "New design for signal sets with low cross correlation, balance property and large linear span - GF(p) case", IEEE Trans. Inform. Theory, vol 48, no. 11, pp.2847-2867, Nov. 2002.

24) G. Gong: "Theory and application of q-ary interleaving sequences". IEEE Trans. Inform. Theory, vol41, pp. 400-41l, March 1995.

25) G. Gong and G. Z.Xiao, “Synthesis and Uniqueness of m-Sequences over GF(q^n) as n-Phase Sequences over GF(q)”. IEEE Trans. on Commu., Vol. 42, No. 8, pp. 2501-2505, 1994.

26) F. G. Gustavson: Analysis of the Berlekamp-Massey Linear Feedback Shift-Register Synthesis Algorithm. IBM Journal Res. Develop. May 1976.

27) C. G. Günther: "Alternating Step Generators controlled by de Brujin Sequeces", EUROCRYPT'- 87, pp 5-14, 1987.

28) H. Han, S. Zhang, L. Zhou and X Liu, " Decimated m-sequences families with optimal partial Hamming correlation ", Cryptography and Communications 12, pp 405-413, 2020

29) S. Hara an R. Prasad, "Overview of multicarrier CDMA", IEEE Commun. Mag. Vol. 35, pp. 126-133, 1997.

30) L.M. Hieu & L.C. Quynh, "Design and Analysis of Sequences with Interleaved Structure by d-Transform," IETE Journal of Research, vol. 51, no. l, pp.61-67, Jan-Feb. 2005.

31) J. He, "Interleaved Sequences Over Finite Fields", Doctor thesis, Carleton University, Ottawa, Canada 2013.

32) S. Jianga , Z. Daia , G. Gong, "On interleaved sequences over finite fields", Discrete Mathematics 252, 2002, pp 161-175.

33) S. Jianga , Z. Daia , G. Gong, "Notes on q-ary Interleaved Sequences", Chinese Science Bulletin volume 45, 2002, pp 502-507.

34) Kaur, M., Gianey, H.K., Singh, D.; Sabharwal, "Multi-objective differential evolution based random forest for e-health applications". Mod. Phys. Lett. B 33(05), 1950022, 2019.

35) E. L. Key, "An Analysic of the Structure and Complexity of Nonlinear Binary Sequence Generators". IEEE Trans. Inform. Theory, Vol IT-22, No-6, November 1976, pp 732-736.

36) A. Klein, "Linear Feedback Shift Registers", *Stream Ciphers*, pp 17-58, Springer, 2013.

37) A.M. Kondoz, "*Digital Speech*", 2nd Edition, Wiley, 2004.

38) A. Klapper; A.H. Chan; M. Goresky, "Cascaded GMW sequences", IEEE Trans, Inform. Theory ,Vol.39, no 1, pp. 177 - 183, 1993.

39) X.D. Lin and K.H. Chang: "Optimal PN sequences design for quasisynchronous CDMA communication systems", IEEE Trans. Comm.vol 45. pp 221-226. Feb 1997.

40) R. Lidl & H. Niedemeiter, *Introduction to finite field and their application*, Cambridge University press 2000.

41) J. L. Massey, “Shift register synthesic and BCH decoding”. IEEE Trans. Inform. Theory, vol IT-15, pp 122-127, 1969.

42) R. J. McEliece, *Finite Field for Computer Scientists and Engineers*. Boston, MA: Kluwer, 1987.

43) A. Menezes, P. van Oorschot, S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.

44) H. Meyr, M.Moeneclaey, and S.A. Fechtel. “Digital Communication Receviers”. John Wiley & Sons, INC, 1998.

45) J.S. No, "P-ary unified sequences: P-ary extended d form sequences with the ideal autocorrelation property", IEEE Trans. Inform. Theory, vol 48, no. 9, pp. 2540-2546, Sept 2002.

46) J.S. No, S.W. Golomb, G. Gong, "Binary pseudo-random sequences of period 2^m-1 with ideal autocorrelation", IEEE Trans. Inform. Theory, vol. 44 no. 2, pp. 814-817, 1998.

47) R.L Perterson, R.E Zeemer & D.B Both, "*Introduction to spread spectrum*", Prentice Hall Int Inc 1995.

48) J.G. Proakis and Masoud Salehi. “*Communication System Engineering*”, volume Second Edition. Prentice Hall, 2002.

49) L.C. Quynh and S.Prasad, "A class of binary cipher sequences with best possible correlation funtion." IEEE Proceeding Part F .Dec 1985. vol 132.pp.560-570.

50) L.C. Quynh and S. Prasad, “Equivalent linear span analysis of binary sequences having an interleaved structure”. IEE Proc., Vol. 133, F, No. 3, June 1986, pp 288-292.

51) Quynh L.C, Cuong N.Le, Thang P.X, "A hardware oriented method to generate and evaluate nonlinear interleaved sequences with desired properties", Journal of Information Engineering and Applications, Vol.6, No.7, 2016.

52) Alan Shamir, "Stream Cipher: Dead or Alive ?", Asia Crypt, 2014.

53) Singh, D.; Kumar, “A comprehensive review of computational dehazing techniques”. Arch. Comput. Methods Eng. 26(5), 1395–1413 (2019)

54) Scholtz R A, Welch L R, “GMW sequences”. IEEE Trans. Inform. Theory, Vol. 30(3), pp. 548–553, 1984.

55) N.V. Son, et. al, "FPGA Implemen-tation of Optimal PN-Sequences by Time-Multiplexing Technique", International Conference on Engineering Research and Applications (ICERA), 2019.

56) Son N.V, Dinh D.X and Quynh L.C, “Some new insights into design and analysis of sequences of interleaved structure”, Journal of Xidian University, Volume 14, Issue 12, 2020.

57) X. H. Tang and F. Z. Fan, "A class of PN sequences over GF (P) with low correlation zone", IEEE Trans. Inform. Theory, vol.41, no.4, pp. 1644-1649, May 2001.

58) X. H. Tang and F. Z. Fan, "Large families of Generalized d-form sequences with Low correlation and Large linear span Based on the Interleaved technique", 2003.

59) K. Tsuchiya, Y. Nogami and S. Uehara, "Interleaved sequences of geometric sequences binarized with Legendre symbol of two types", IEICE Trans. Fundamential of Electronics, Communication and Computing Science, 2017

60) Truong D.V, Binh N.T, Hieu L.M and Quynh L.C, “Construction of Nonlinear q-ary m-sequences with Interleaved Structure by d-Transform”, IEEE ICCE 2018, pp.389-392, 2018.

61) B. Walke, S. Seidenberg, M. P. Althoft. "*UTMS: The fundamental*", John Wiley, 2003.

62) Nam Yul Yu, “On Periodic Correlation of Binary Sequences”, Doctor thesis, University of Waterloo, Canada, 2005.

Trang Web

63) Stephen Wolfram, Solomon Golomb (1932–2016) *https://blog.stephenwolfram.com/2016/05/solomon-golomb-19322016/*

64) TOP 500 Super computer *https://www.top500.org/lists/top500/2021/11/*