# MỤC LỤC

# GIỚI THIỆU TỔNG QUAN

**GIÁO TRÌNH LÝ THUYẾT THÔNG TIN**

# MỤC ĐÍCH

❖ Giáo trình này sẽ cung cấp cho người đọc những khối kiến thức cơ bản của lý thuyết thông tin như: Độ do lượng tin (Measure of Information), Sinh mã tách được (Decypherable Coding), Kênh truyền tin rời rạc không nhớ (Discrete Memoryless Channel) và Sửa lỗi trên kênh truyền (Error Correcting Codings).

- **Liên quan đến Độ đo lượng tin**, giáo trình sẽ trình bày các khái niệm cơ bản về thông tin, entropy, một số công thức, tính chất, các định lý quan trọng của entropy và cách tính lượng tin.

- **Về Sinh mã tách được**, giáo trình sẽ giới thiệu đến người học các vấn đề về yêu cầu của bài toán sinh mã, giải mã duy nhất, cũng như mã tức thời và giải thuật kiểm tra mã tách được. Các định lý quan trọng được đề cập trong nội dung này là: Định lý Kraft (1949), Định lý Shannon (1948) và Định lý sinh mã Huffman.

- **Về kênh truyền tin rời rạc không nhớ**, giáo trình sẽ giới thiệu mô hình kênh truyền theo 2 khía cạnh vật lý và toán học. Các khái niệm về dung lượng kênh truyền, phân lớp kênh truyền, định lý về dung lượng kênh truyền, cũng như các khái niệm trong kỹ thuật truyền tin và phương pháp xây dựng lược đồ giải mã tối ưu cũng được trình bày trong môn học này.

- **Vấn đề Sửa lỗi** (hay **xử lý mã sai) trên kênh truyền** là một vấn đề rất quan trọng và được quan tâm nhiều trong môn học này. Các nội dung được giới thiệu đến các bạn sẽ là Nguyên lý Khoảng cách Hamming, các định lý về Cận Hamming, phương pháp kiểm tra chẵn lẻ, các lược đồ sửa lỗi, Bảng mã Hamming và Bảng mã xoay vòng.

❖ Hơn nữa, hầu hết các vấn đề nêu trên đều được đưa vào nội dung giảng dạy ở các bậc Đại học của một số ngành trong đó có ngành Công nghệ thông tin. Do đó, để có một tài liệu phục vụ công tác giảng dạy của giáo viên cũng như việc học tập và nghiên cứu của sinh viên, chúng tôi mạnh dạn biên soạn giáo trình này nhằm giúp cho sinh viên có một tài liệu tự học và nghiên cứu một cách hiệu quả.

# YÊU CẦU

❖ Sau khi học xong môn này, sinh viên phải có được những khả năng sau:
- Hiểu các khái niệm về về thông tin, Entropy, Entropy của một phân phối, Entropy của nhiều phân phối, Entropy có điều kiện, Độ đo lượng tin. Vận dụng giải quyết các bài toán về xác định lượng tin.
- Biết khái niệm về mã tách được, mã không tách được, bảng mã tối ưu. Hiểu Định lý Kraft (1949), Định lý Shannon (1948), Định lý sinh mã Huffman và phương pháp sinh mã Huffman. Vận dụng để sinh bảng mã tách được tối ưu, nhận biết được bảng mã như thế nào là bảng mã tối ưu và có thể vận dụng để viết các chương trình sinh mã, giải mã (hay viết chương trình nén và giải nén). Từ đây, các sinh viên có thể tự nghiên cứu các loại bảng mã khác để vận dụng cho việc mã hóa và bảo mật thông tin một cách hiệu quả.

- Biết các khái niệm về kênh truyền tin rời rạc không nhớ, dung lượng kênh truyền và phân lớp kênh truyền. Hiểu định lý về dung lượng kênh truyền, phương pháp xây dựng lược đồ giải mã tối ưu và cách tính xác suất truyền sai trên kênh truyền.
- Biết các khái niệm về khoảng cách Hamming, nguyên lý khoảng cách Hamming, các định lý về Cận Hamming, phương pháp kiểm tra chẵn lẻ, các lược đồ sửa lỗi, Bảng mã Hamming và Bảng mã xoay vòng.
- Vận dụng các kiến thức học được để thiết kế một hệ thống truyền nhận dữ liệu với quy trình cơ bản: mã hóa, giải mã và bảo mật thông tin.

❖ Lý thuyết thông tin cũng là một trong các môn học khó của ngành Công nghệ thông tin vì nó đòi hỏi người học phải có kiến thức cơ bản về toán và xác suất thống kê. Do đó, đòi hỏi người học phải tự bổ sung các kiến thức cơ bản về toán và xác suất thống kê cho mình (nếu thiếu), tham gia lớp học đầy đủ và làm các bài tập theo yêu cầu của môn học thì mới tiếp thu kiến thức môn học một cách hiệu quả.

# NỘI DUNG CỐT LÕI

Giáo trình gồm 5 chương được trình bày trong 45 tiết giảng cho sinh viên chuyên ngành Công nghệ thông tin, trong đó có khoảng 30 tiết lý thuyết và 15 tiết bài tập mà giáo viên sẽ hướng dẫn cho sinh viên trên lớp.

Chương 1: Giới thiệu. *Chương này trình bày các nội dung có tính tổng quan về môn học bao gồm: các đối tượng nghiên cứu, mô hình lý thuyết thông tin theo quan điểm của nhà toán học Shannon, khái niệm về lượng tin biết và chưa biết, định lý cơ bản của kỹ thuật truyền tin.*

Chương 2: Độ đo lượng tin. *Chương này trình bày các vấn đề cơ bản về entropy, các tính chất của entropy, entropy của nhiều biến, entropy có điều kiện, các định lý về quan hệ giữa các entropy và lượng tin của một sự kiện.*

Chương 3: Sinh mã tách được. *Nội dung chính của chương này bao gồm các khái niệm về mã tách được, quan hệ giữa mã tách được và độ dài mã, tính tối ưu của độ dài mã.*

Chương 4: Kênh truyền. *Các nội dung được trình bày trong chương này bao gồm khái niệm về kênh truyền tin rời rạc không nhớ, các mô hình truyền tin ở khía cạnh vật lý và toán học, dung lượng trên kênh truyền, phân lớp các kênh truyền. Phương pháp xây dựng lược đồ giải mã tối ưu và cách tính xác suất truyền sai cũng được giới thiệu trong chương này.*

Chương 5: Sửa lỗi. *Chương này trình bày các nội dung cốt lõi sau: khái niệm về khoảng cách Hamming, nguyên lý khoảng cách nhỏ nhất Hamming, bổ đề về tự sửa lỗi và định lý Cận Hamming. Chương này cũng giới thiệu về bộ mã kiểm tra chẵn lẻ, phương pháp kiểm tra chẵn lẻ, lược đồ sửa lỗi tối ưu, mã Hamming và mã xoay vòng.*

# KẾT THỨC TIÊN QUYẾT

Để học tốt môn học này, đòi hỏi sinh viên phải nắm vững các môn học có liên quan như: xác suất thống kê, đại số boole (phép toán Modulo 2 và đa thức nhị phân). Các môn học có liên quan và có thể tham khảo thêm như kỷ thuật số, hệ điều hành, mạng máy tính.

# TÀI LIỆU THAM KHẢO

1. David J.C. Mackey, *Information Theory, Infernce, and Learning Algorithms,* CamBridge University Express-2003.
2. G.J.ChaiTin, *Algorithmic Information Theory*, CamBridge University Express-1992.
3. Sanford Goldman, *Information Theory.*
4. http://www.inference.phy.cam.ac.uk/mackay/info-theory/course.html.
5. http://en.wikipedia.org/wiki/Information_theory.
6. http://www-2.cs.cmu.edu/~dst/Tutorials/Info-Theory/.
7. http://cscs.umich.edu/~crshalizi/notebooks/information-theory.html.
8. http://www.lecb.ncifcrf.gov/~toms/paper/primer/primer.pdf.
9. http://www.cs.ucl.ac.uk/staff/S.Bhatti/D51-notes/node27.html.
10. http://guest.engelschall.com/~sb/hamming/.
11. http://www2.rad.com/networks/1994/err_con/hamming.htm

# PHƯƠNG PHÁP HỌC TẬP

Để phục vụ cho mục tiêu nâng cao khả năng tự học tập và tự nghiên cứu của sinh viên, giáo trình này được biên soạn cùng với các giáo trình khác thuộc chuyên ngành Công nghệ thông tin của Khoa Công nghệ thông tin và Truyền thông – Đại Học Cần Thơ theo dự án **ASVIET002CNTT "Tăng cường hiệu quả đào tạo và năng lực đào tạo của sinh viên khoa Công nghệ Thông tin-Đại học Cần Thơ"**. Chúng tôi đã cố gắng trình bày giáo trình này một cách có hệ thống các nội dung theo bố cục các chương ứng với các khối kiến thức nêu trên, mỗi chương được được trình bày theo bố cục của các bài học và mỗi bài học giới thiệu đến người học một vấn đề nào đó trong số các vấn đề của một khối kiến thức tương ứng với một chương. Khi học xong các bài học của một chương, người học sẽ có một khối kiến thức cần thiết tương ứng cho môn học. Nội dung của các bài học đều được đưa vào các ví dụ để người học dễ hiểu, tùy theo từng vấn đề mà người học cần phải học và nghiên cứu trong thời lượng từ 1 đến 2 tiết tự học cho một bài học trong một chương. Như vậy, để học tốt môn học này, trước hết sinh viên cần phải:

- Học đầy đủ các môn học tiên quyết, bổ sung những kiến thức cơ bản về toán và xác suất thống kê (nếu thiếu).
- Học và nghiên cứu kỹ từng chương theo trình tự các chương được trình bày trong giáo trình này. Trong từng chương, học các bài theo thứ tự được trình bày, sau mỗi bài phải làm bài tập đầy đủ (nếu có).
- Tham gia lớp đầy đủ, thảo luận các vấn đề tồn tại chưa hiểu trong quá trình tự học.
- Sau mỗi chương học, phải nắm vững các khái niệm, các định nghĩa, các công thức tính toán và vận dụng giải các bài toán có tính chất tổng hợp được giới thiệu ở cuối chương.
- Vận dụng kiến thức có được sau khi học xong các chương để giải một số bài tập tổng hợp ở cuối giáo trình, từ đó giúp cho người học hiểu sâu hơn về môn học và có thể giải quyết các vấn đề tương tự trong thực tế.

Việc cho ra đời một giáo trình với những mục đích như trên là không đơn giản khi khả năng và kinh nghiệm của người soạn còn có hạn, nhiều khái niệm, thuật ngữ dùng trong giáo trình chưa được định nghĩa một cách chính thống. Vì vậy giáo trình này chắc không tránh khỏi những khiếm khuyết, rất mong nhận được sự góp ý của các đồng nghiệp và người đọc.

# CHƯƠNG 1: GIỚI THIỆU

## 1: Mục tiêu

Sau khi hoàn tất bài học này bạn có thể biết:

- Đối tượng nghiên cứu,
- Mô hình lý thuyết thông tin theo quan điểm Shannon,
- Các khái niệm về Lượng tin biết và lượng tin chưa biết,
- Định lý cơ sở của kỹ thuật truyền tin,
- Khái niệm chung về dung lượng kênh truyền,
- Vấn đề sinh mã và giải mã.

### Đối tượng nghiên cứu

Lý thuyết thống kê về thông tin được xây dựng trên hai hướng khác nhau bởi hai nhà toán học Shannon (1948) và Wiener (1949). Lý thuyết thông tin nghiên cứu quá trình xử lý tín hiệu như sau:

Đầu vào (input): nhận tín hiệu từ một lĩnh vực cụ thể, tức là tín hiệu xuất hiện theo các ký hiệu (symbol) từ một tập hợp cho trước và theo phân phối xác suất đã biết.
Tín hiệu được truyền đi trên kênh truyền (channel) và có thể bị nhiễu cũng theo một phân phối xác suất nào đó. Kênh truyền có thể được hiểu dưới hai nghĩa:
Dưới nghĩa vật lý: kênh truyền là một hệ thống truyền tín hiệu (dây dẫn, mạch, sóng, ...) và gây nhiễu tùy thao chất lượng của hệ thống.

Dưới nghĩa toán học: kênh truyền là các phân phối xác suất xác định trên lớp các tín hiệu đang xét ở đầu nhận tín hiệu (output).

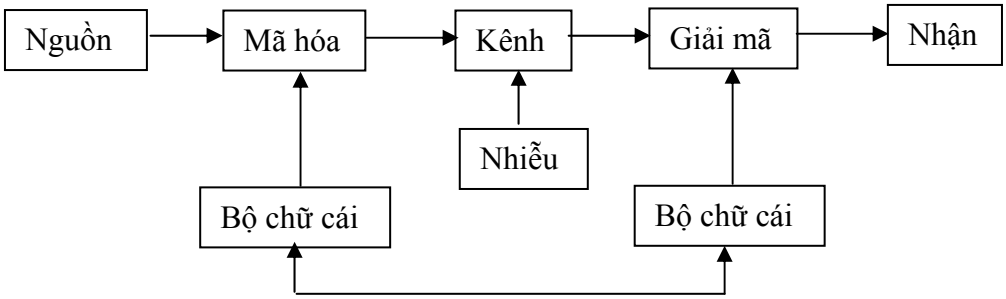
Ở đầu ra (output): dựng lại tín hiệu chân thật nhất có thể có so với tín hiệu ở đầu vào.
Shannon xây dựng mô hình lý thuyết thông tin trên cơ sở giải quyết bài toán: sinh mã độ dài tối ưu khi nhận tín hiệu đầu vào. Tín tối ưu được xét trên 3 yếu tố sau:
Phân phối xác suất của sự xuất hiện của các tín hiệu.

Tính duy nhất của mã và cho phép tự điều chỉnh mã sai nếu có với độ chính xác cao nhất. Giải mã đồng thời tự động điều chỉnh mã hoặc xác định đoạn mã truyền sai.

Trong khí đó, Wiener lại nghiên cứu phương pháp xử lý tín hiệu ở đầu ra: ước lượng tối ưu chuỗi tín hiệu so với chính nó khi nhận ở đầu vào không qua quá trình sinh mã. Như vậy phương pháp Wiener được áp dụng trong những trường hợp con người không kiểm soát được quá trình truyền tín hiệu. Môn "xử lý tín hiệu" đã đề cập đến vấn đề này.

### Mô hình lý thuyết thông tin theo quan điểm Shannon

Lý thuyết thông tin được xét ở đây theo quan điểm của Shannon. Đối tượng nghiên cứu là một hệ thống liên lạc truyền tin (communication system) như sơ đồ dưới đây:




### Diễn giải:

- Nguồn (source) thông tin còn gọi là thông báo cần được truyền ở đầu vào (Input).
- Mã hóa (encode) là bộ sinh mã. Ứng với một thông báo, bộ sinh mã sẽ gán cho một đối tượng (object) phù hợp với kỹ thuật truyền tin. Đối tượng có thể là:
  - Dãy số nghị phân (Digital) dạng: 01010101, cũng giống như mã máy tính.
  - Sóng liên tục (Analog) cũng giống như truyền radio.
- Kênh (channel) là phương tiện truyền mã của thông tin.
- Nhiễu (noise) được sinh ra do kênh truyền tin. Tùy vào chất lượng của kênh truyền mà nhiễu nhiều hay ít.
- Giải mã (decode) ở đầu ra (output) đưa dãy mã trở về dạng thông báo ban đầu với xác suất cao nhất. Sau đó thông báo sẽ được chuyển cho nới nhận. Trong sơ đồ trên, chúng ta quan tâm đến 2 khối mã hóa và giải mã trong toàn bộ môn học.

### Lượng tin biết và chưa biết

Một biến ngẫu nhiên (BNN) X luôn mang một lượng tin nào đó. Nếu X chưa xảy ra (hay ta chưa biết cụ thể thông tin về X) thì lượng tin của nó là chưa biết, trong trường hợp này X có một lượng tin chưa biết. Ngược lại nếu X đã xảy ra (hay ta biết cụ thể thông tin về X) thì lượng tin về biến ngẫu nhiên X coi như đã biết hoàn toàn, trong trường hợp này X có một lượng tin đã biết.

Nếu biết thông tin của một BNN X thông qua BNN Y đã xảy ra thì ta có thể nói: chúng ta chỉ biết một phần lượng thông tin của X đó trên cơ sở biết Y.

### Ví dụ về lượng tin biết và chưa biết

Ta xét ví dụ về một người tổ chức trò chơi may rủi khách quan với việc tung một đồng tiền "có đầu hình – không có đầu hình". Nếu người chơi chọn mặt không có đầu hình thì thắng khi kết quả tung đồng tiền là không có đầu hình, ngược lại thì thua. Tuy nhiên người tổ chức chơi có thể "ăn gian" bằng cách sử dụng 2 đồng tiền "Thật- Giả" khác nhau sau:

+ Đồng tiền loại 1 (hay đồng tiền thật): đồng chất có 1 mặt có đầu hình.

+ Đồng tiền loại 2 (hay đồng tiền giả ): đồng chất, mỗi mặt đều có 1 đầu hình.

Mặc dù người tổ chức chơi có thể "ăn gian" nhưng quá trình trao đổi 2 đồng tiền cho nhau là ngẫu nhiêu, vậy liệu người tổ chức chơi có thể "ăn gian" hoàn toàn được không? Hay lượng tin biết và chưa biết của sự kiện lấy một đồng tiền từ 2 đồng tiền nói trên được hiểu như thế nào?

Ta thử xét một trường hợp sau: nếu người chơi lấy ngẫu nhiên 1 đồng tiền và sau đó thực hiện việc tung đồng tiền lấy được 2 lần. Qua 2 lần tung đồng tiền, ta đếm được số đầu hình xuất hiện. Dựa vào số đầu hình xuất hiện, ta có thể phán đoán được người tổ chức chơi đã lấy được đồng tiền nào.

Chẳng hạn: Nếu số đầu hình đếm được sau 2 lần tưng là 1 thì đồng tiền đã lấy được là đồng tiền thật. Ngược lại nếu số đầu hình đếm được là 2 thì đồng tiền đã lấy được có thể là thật hay cũng có thể là giả. Như vậy, ta đã nhận được một phần thông tin về loại đồng tiền qua số đầu hình đếm được sau 2 lần tung. Ta có thể tính được lượng tin đó bằng bao nhiêu? *(Việc tính lượng tin này sẽ được thảo luận sau).* Dưới đây là một số bảng phân phối của bài toán trên:

Gọi BNN X về loại đồng tiền (X=1 nếu lấy được đồng tiền loại 1 và X=1 nếu lấy được đồng tiền loại 2 được lấy).
Khi đó phân phối của X có dạng:

| X | 1 | 2 |
|---|---|---|
| P | 0.5 | 0.5 |

Đặt BNN Y là BNN về số đầu hình đếm được sau 2 lần tung. Khi đó ta có thể xác định được phân phối của Y với điều kiện xảy ra của X trong 2 trường hợp sau.

Phân phối của Y khi biết X=1 có dạng:

| Y/X=1 | 0 | 1 | 2 |
|---|---|---|---|
| P | 0.25 | 0.5 | 0.25 |

Phân phối của Y khi biết X=2 có dạng:

| Y/X=2 | 0 | 1 | 2 |
|---|---|---|---|
| P | 0 | 0 | 1 |

### Định lý cơ sở của kỹ thuật truyền tin

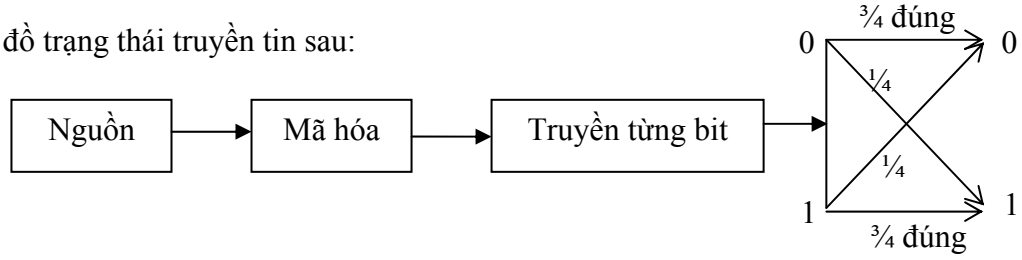
Trong " A New Basic of Information Theory (1954)", Feinstein đã đưa ra định lý sau: "Trên một kênh truyền có nhiễu, người ta luôn có thể thực hiện một phương pháp truyền sao cho đạt được sai số nhỏ hơn sai số cho phép (nhỏ bất kỳ) cho trước đối với kênh truyền."

Chúng ta sẽ không chứng minh định lý, thay vào đó, chúng ta sẽ tham khảo đến các minh họa giảm nhiễu trong các nội dung tiếp theo của bài học.

### Mô tả trạng thái truyền tin có nhiễu

Giả sử, một thông báo được truyền đi trên một kênh truyền nhị phân rời rạc. Thông báo cần truyền được mã hóa thành dãy số nhị phân (0,1) và có độ dài được tính theo đơn vị bit. Giả sử 1 bit truyền trên kênh nhiễu với xác suất 1/4 (hay tính trung bình cứ truyền 4 bit thì có thể nhiễu 1 bit).

Ta có sơ đồ trạng thái truyền tin sau:



### Minh họa kỹ thuật giảm nhiễu

Trong kỹ thuật truyền tin, người ta có thể làm giảm sai lầm khi nhận tin bằng cách truyền lặp lại 1 bit với số lẻ lần.

**Ví dụ:** truyền lặp lại 3 cho 1 bit cần truyền (xác suất nhiễu 1 bit bằng 1/4). Khi nhận 3 bit liền nhau ở cuối kênh được xem như là 1 bit. Giá trị của bit này được hiểu là 0 (hay 1) nếu bit 0 (bit 1) có số lần xuất hiện nhiều hơn trong dãy 3 bit nhận được liền nhau (hay giải mã theo nguyên tắc đa số). Ta cần chứng minh với phương pháp truyền này thì xác suất truyền sai thật sự < 1/4 (xác suất nhiễu cho trước của kênh truyền).

### Sơ đồ truyền tin:

| Bit truyền | Tuyền lặp 3 lần | Nhận 3 bit | Giải mã |
|---|---|---|---|
| 0 → | 000 → | 000 | 0 |
| | 000 | 001 | 0 |
| | 000 | 010 | 0 |
| | 000 | 100 | 0 |
| | 000 | 101 | 1 |
| | 000 | 011 | 1 |
| | 000 | 110 | 1 |
| | 000 | 111 | 1 |
| 1 → | 111 → | 000 | 0 |
| | 111 | 001 | 0 |
| | 111 | 010 | 0 |
| | 111 | 100 | 0 |
| | 111 | 011 | 1 |
| | 111 | 110 | 1 |
| | 111 | 111 | 1 |
| | 111 | 111 | 1 |

Thật vậy:

Giả sử $X_i$ xác định giá trị đúng hay sai của bit thứ i nhận được ở cuối kênh truyền với $X_i$ =1 nếu bit thứ i nhận được là sai và $X_i$ =0 nếu bit thứ i nhận được là đúng. Theo giả thiết ban đầu của kênh truyền thì phân phối xác suất của $X_i$ có dạng Bernoulli b(1/4):

| $X_i$ | 1 | 0 |
|---|---|---|
| P | 3/4 | 1/4 |

Gọi $Y = \{X_1 + X_2 + X_3\}$ là tổng số bit nhận sai sau 3 lần truyền lặp cho 1 bit. Trong trường hợp này Y tuân theo phân phối Nhị thức B(p,n), với p=1/4 (xác suất truyền sai một bit) và q =3/4 (xác suất truyền đúng 1 bit):
Y ~ B(i,n) hay

$p(Y = i) = C_n^i . p^i q^{n-i}$

Trong đó: $C_n^i = \frac{n!}{i!(n-i)!}$

Vậy truyền sai khi Y ∈ {2, 3} có xác xuất là:

$P_{sai}$= P(y≥2) = P(Y=2) + P(Y=3) = B(2,3) + B(2,3)

Hay $Psai = (C_3^2(\frac{1}{4})^2.(\frac{3}{4})^1) + (C_3^3(\frac{1}{4})^3(\frac{3}{4})^0) = \frac{10}{64} < \frac{1}{4}$ (đpcm).

### Chi phí phải trả cho kỹ thuật giảm nhiễu

Theo cách thức lặp lại như trên, ta có thể giảm sai lầm bao nhiêu cũng được (lặp càng nhiều thì sai càng ít), nhưng thời gian truyền cũng tăng lên và chi phí truyền cũng sẽ tăng theo.
Hay ta có thể hiểu như sau:
Lặp càng nhiều lần 1 bit => thời gian truyền càng nhiều => chi phí càng tăng.

### Khái niệm về dung lượng kênh truyền

Ví dụ trên cho chúng ta thấy cần phải xác định một thông số cho truyền tin để đảm bảo sai số chấp nhận được và đồng thời tốc độ truyền cũng không quá chậm.

Khái niệm "dung lượng" kênh truyền là khái niệm rất cơ bản của lý thuyết truyền tin và là một đại lượng vật lý đồng thời cũng là đại lượng toán học (có đơn vị là bit). Nó cho phép xác định tốc độ truyền tối đa của mỗi kênh truyền. Do đó, dựa vào dung lượng kênh truyền, người ta có thể chỉ ra tốc độ truyền tin đồng thời với một phương pháp truyền có sai số cho phép.

### Vấn đề sinh mã

Từ kỹ thuật truyền tin trên cho ta thấy quá trình sinh mã và giải mã được mô tả như sau: một đơn vị thông tin nhận được ở đầu vào sẽ được gán cho một ký hiệu trong bộ ký hiệu sinh mã. Một ký hiệu mã được gán n lần lặp lại (dựa vào dung lượng của kênh truyền, ta có thể xác định được n). Thiết bị sinh mã (Coding device/ Encoder) sẽ thực hiện quá trình sinh mã.

Như vậy, một đơn vị thông tin từ nguồn phát tin sẽ được thiết bị sinh mã gán cho một dãy n ký hiệu mã. Dãy ký hiệu mã của 1 đơn vị thông tin được gọi là một từ mã (Code word). Trong trường hợp tổng quát, người ta có thể gán một khối ký tự mã cho một khối thông tin nào đó và được gọi là một từ mã.

### Vấn đề giải mã

Ở cuối kênh truyền, một thiết bị giải mã (Decoding device/ Decoder) sẽ thực hiện quá trình ngược lại như sau: kiểm tra dãy ký hiệu mã để quyết định giải mã về một từ mã và đưa nó về dạng khối tin ban đầu.

**Ví dụ:**

Khối tin ban đầu : 01010101
Khối ký hiệu mã ở đầu truyền (lặp 3 lần): 000111000111000111000111.
Khối ký hiệu mã ở đầu nhận : 001110100111011001000111
Khối tin nhận được cuối cùng : 01011001 (sai 2 bit so với khối tin ban đầu)

Do đó làm sao để đưa khối tin nhận được về khối tin ban đầu 01010101, đây chính là công việc của bộ giải mã (Decoder).

Một vấn đề quan trọng cần lưu ý là phải đồng bộ giữa tốc độ nạp thông tin (phát tín hiệu) với tốc độ truyền tin. Nếu tốc độ nạp thông tin bằng hoặc lớn hơn so với tốc độ truyền tin của kênh, thì cần phải giảm tốc độ nạp thông tin sao cho nhỏ hơn tốc độ truyền tin.

# CHƯƠNG 2: ĐỘ ĐO LƯỢNG TIN

**Mục tiêu:** trình bày các khái niệm về độ đo lượng tin chưa biết và đã biết về một biến ngẫu nhiên X. Tính toán các lượng tin này thông qua định nghĩa và các tính chất của Entropy từ một hay nhiều biến ngẫu nhiên.

## BÀI 2.1: ENTROPY

### Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:

- Hiểu được các khái niệm Entropy,
- Biết Entropy của một sự kiện và Entropy của một phân phối,
- Hiểu Định lý dạng giải tích của Entropy,
- Biết Bài toán về cây tìm kiếm nhị phân và
- Làm kiến thức cơ sở để hiểu và học tốt các bài học tiếp theo.

### Ví dụ về entropy

Trước hết, ta cần tìm hiểu một ví dụ về khái niệm độ do của một lượng tin dựa vào các sự kiện hay các phân phối xác suất ngẫu nhiên như sau:
Xét 2 BNN X và Y có phân phối sau:

X={1, 2, 3, 4, 5} có phân phối đều hay p(X=i) = 1/5.
Y={1, 2} cũng có phân phối đều hay p(Y=i) = 1/2.

Bản thân X và Y đều mang một lượng tin và thông tin về X và Y chưa biết do chúng là ngẫu nhiên. Do đó, X hay Y đều có một lượng tin không chắc chắn và lượng tin chắc chắn, tổng của 2 lượng tin này là không đổi và thực tế nó bằng bao nhiêu thì ta chưa thể biết. Lượng tin không chắc chắn của X (hay Y) được gọi là Entropy.

Tuy nhiên, nếu X và Y có tương quan nhau thì X cũng có một phần lượng tin không chắc chắn thông qua lượng tin đã biết của Y (hay thông tin về Y đã được biết). Trong trường hợp này, một phần lượng tin không chắc chắn của thông qua lượng tin đã biết của Y được gọi là Entropy có điều kiện.

### Nhận xét về độ đo lượng tin

Rõ ràng, ta cần phải xây dựng một đại lượng toán học rất cụ thể để có thể đo được lượng tin chưa biết từ một biến ngẫu nhiên. Một cách trực quan, lượng tin đó phải thể hiện được các vấn đề sau:

Một sự kiện có xác suất càng nhỏ thì sự kiện đó ít xảy ra, cũng có nghĩa là tính không chắc chắn càng lớn. Nếu đo lượng tin của nó thì nó cho một lượng tin không biết càng lớn.
Một tập hợp các sự kiện ngẫu nhiên (hay Biến ngẫu nhiên) càng nhiều sự kiện có phân phối càng đều thì tính không chắc chắn càng lớn. Nếu đo lượng tin của nó thì sẽ được lượng tin không biết càng lớn. Hay lượng tin chắc chắn càng nhỏ.

Một phân phối xác suất càng lệch nhiều (có xác xuất rất nhỏ và rất lớn) thì tính không chắc chắn càng ít và do đó sẽ có một lượng tin chưa biết càng nhỏ so với phân phối xác suất đều hay lượng tin chắc chắn của nó càng cao.

## Khái niệm entropy

Trong tiếng việt ta chưa có từ tương đương với từ Entropy, tuy nhiên chúng ta có thể tạm hiểu hiểu thoáng qua trước khi đi vào định nghĩa chặc chẽ về mặt toán học của Entropy như sau:

**Entropy** là một đại lượng toán học dùng để đo lượng tin không chắc (hay lượng ngẫu nhiên) của một sự kiện hay của phân phối ngẫu nhiên cho trước. Hay một số tài liệu tiếng anh gọi là Uncertainty Measure.

## Entropy của một sự kiện

Giả sử có một sự kiện A có xác suất xuất hiện là p. Khi đó, ta nói A có một lượng không chắc chắn được đo bởi hàm số h(p) với $p \subseteq [0,1]$. Hàm h(p) được gọi là Entropy nếu nó thoả 2 tiêu đề toán học sau:

Tiên đề 01: h(p) là hàm liên tục không âm và đơn điệu giảm.
Tiên đề 02: nếu A và B là hai sự kiện độc lập nhau, có xác suất xuất hiện lần lượt là $p_A$ và $p_B$. Khi đó, $p(A,B) = p_A.p_B$ nhưng $h(A,B) = h(p_A) + h(p_B)$.

## Entropy của một phân phối

Xét biến ngẫu nhiên X có phân phối:

| X | $x_1$ | $x_2$ | $x_3$ | … | $x_M$ |
|---|---|---|---|---|---|
| P | $p_1$ | $p_2$ | $p_3$ | … | $p_M$ |

Nếu gọi $A_i$ là sự kiện $X=x_i$, (i=1,2,3,..) thì Entropy của $A_i$ là: $h(A_i)= h(p_i)$
Gọi Y=h(X) là hàm ngẫu nhiên của X và nhận các giá trị là dãy các Entropy của các sự kiện $X=x_i$, tức là $Y=h(X)=\{h(p_1), h(p_2), …, h(p_n)\}$.

Vậy, Entropy của X chính là kỳ vọng toán học của Y=h(X) có dạng:
$H(X)=H(p_1, p_2, p_3, …,p_n) = p_1h(p_1)+ p_2h(p_2)+…+p_nh(p_n)$.
Tổng quát:

$$H(X) = \sum_{i=1}^{n} p_i h(p_i)$$

## Định lý dạng giải tích của Entropy

Định lý: Hàm $H(X) = H(p_1, p_2,...,p_M) = C\sum_{i=1}^{M} p_i \log(p_i)$

C = const >0
Cơ số logarithm là bất kỳ.
Bổ đề: h(p)=-Clog(p).
Trường hợp C=1 và cơ số logarithm = 2 thì đơn vị tính là bit.
Khi đó: $h(p)=-\log_2(p)$ (đvt: bit) và

$$H(X) = H(p_1, p_2, ..., p_M) = -\sum_{i=1}^{M} p_i \log_2(p_i)$$

Qui ước trong cách viết: $\log(p_i) = \log_2(p_i)$

## Ví dụ minh họa

Nếu sự kiện A có xác suất xuất hiện là 1/2 thì h(A)=h(1/2)= -log(1/2) = 1 (bit)
Xét BNN X có phân phối sau:

| X | $x_1$ | $x_2$ | $x_3$ |
|---|---|---|---|
| P | 1/2 | 1/4 | 1/4 |

H(X) = H(1/2, 1/4, 1/4) = -(1/2log(1/2)+1/4log(1/4)+1/4log(1/4)) =3/2 (bit)

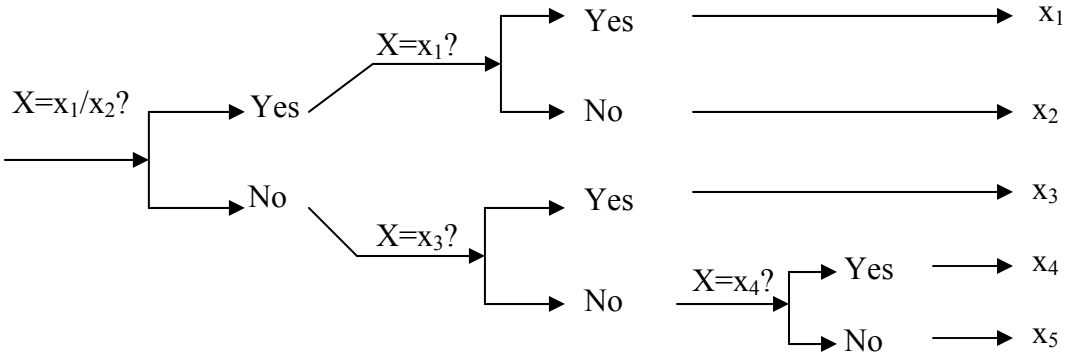
## Bài toán về cây tìm kiếm nhị phân-Đặt vấn đề

Giả sử, tìm 1 trong 5 người có tên biết trước sẽ xuất hiện theo phân phối sau:

| X | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ |
|---|---|---|---|---|---|
| P | 0,2 | 0,3 | 0,2 | 0,15 | 0,15 |

Trong đó: $x_1$, …$x_5$ lần lượt là tên của 5 người mà ta cần nhận ra với cách xác định tên bằng câu hỏi đúng sai (yes/no).

Sơ đồ dưới đây minh họa cách xác định tên của một người:




## Bài toán về cây tìm kiếm nhị phân - Diễn giải

Theo sơ đồ trên:
Để tìm $x_1$, $x_2$, $x_3$ với xác suất tương ứng là 0.2, 0.3, 0.2 ta chỉ cần tốn 2 câu hỏi.
Để tìm $x_4$, $x_5$ với xác suất tương ứng 0.15, 0.15 thì ta cần 3 câu hỏi.
Vậy:
Số câu hỏi trung bình là: 2 x (0,2+0,3+0,2) + 3 x (0,15+0,15) = 2.3
Mặt khác: Entropy của X: H(X)= H(0.2, 0.3, 0.2, 0.15, 0.15)=2.27.

Ta luôn có số câu hỏi trung bình luôn ≥ H(X) (theo định lý Shannon sẽ trình bày sau). Vì số câu hỏi trung bình trong trường hợp này xấp sỉ H(X) nên đây là số câu hỏi trung bình tối ưu để tìm ra tên chính xác của một người. Do đó, sơ đồ tìm kiếm trên là sơ đồ tối ưu.

Sinh viên tự cho thêm 1 hay 2 sơ đồ tìm kiếm khác và tự diễn giải tương tự - xem như bài tập.

## Bài tập

Tính H(X) với phân phối sau:

| X | $x_1$ | $x_2$ | $x_3$ |
|---|---|---|---|
| P | 1/3 | 1/3 | 1/3 |

Tính H(Y) với phân phối sau:

| Y | $x_1$ | $x_2$ | $x_3$ | $x_4$ |
|---|---|---|---|---|
| P | 1/6 | 2/6 | 1/6 | 2/6 |

# BÀI 2.2: CÁC TÍNH CHẤT CỦA ENTROPY

## Mục tiêu:

Sau khi hoàn tất bài học này bạn có thể:

- Hiểu các tính chất cơ bản của Entropy,
- Hiểu định lý cực đại của Entropy,
- Vận dụng giải một số bài toán về Entropy,
- Làm cơ sở để vận dụng giải quyết các bài toán tính dung lượng kênh truyền.

## Các tính chất cơ bản của Entropy

Xét biến ngẫu nhiên X = {$x_1$, $x_2$, …, $x_M$}. Entropy của biến ngẫu nhiên X có các tính chất:

1. Hàm số f(M) = H($\frac{1}{M}$,…,$\frac{1}{M}$) đơn điệu tăng.
2. Hàm số f(ML) = f(M)+f(L).
3. H($p_1$, $p_2$, …, $p_M$) = H($p_1 + p_2 + \ldots + p_r$, $p_{r+1} + p_{r+2} + \ldots + p_M$)

$$+ (p_1 + p_2 + \ldots + p_r) H\left(\frac{p_1}{\sum_{i=1}^{r} p_i}, \ldots, \frac{p_r}{\sum_{i=1}^{r} p_i}\right)$$

$$+ (p_{r+1} + p_{r+2} + \ldots + p_M) H\left(\frac{p_{r+1}}{\sum_{i=r+1}^{M} p_i}, \ldots, \frac{p_M}{\sum_{i=r+1}^{M} p_i}\right)$$

4. H(p, 1-p) là hàm liên tục theo P.

## Minh họa tính chất 1 và 2

Minh họa tính chất 1:

Trong trường hợp biến ngẫu nhiên X có phân phối đều

Entropy của X như sau :

$$H(X) = H\left(\frac{1}{M}, \frac{1}{M}, \cdots, \frac{1}{M}\right) = -\frac{1}{m}\log\frac{1}{M} - \frac{1}{M}\log\frac{1}{M}, \ldots, -\frac{1}{M}\log\frac{1}{M} = -M\frac{1}{M}\log\frac{1}{M}$$

=> $H(X) = -\log\frac{1}{M} = \log M$ là hàm đơn điệu tăng

Minh họa tính chất 2:

Trong trường hợp 2 biến ngẫu nhiên X, Y độc lập có phân phối đều với BNN X có M sự kiện và BNN Y có L sự kiện.

Gọi f(M), f(L) lần lượt là Entropy của X, của Y. Theo tính chất 2 của Entropy ta có f(ML)=f(M)+f(L)

## Minh họa tính chất 3 và 4

Minh họa tính chất 3:

Xét con xúc sắc có 6 mặt với xác suất xuất hiện các mặt được cho trong bảng sau:

| X | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $x_6$ |
|---|---|---|---|---|---|---|
| P | 10% | 20% | 25% | 25% | 15% | 5% |

Ta có thể gom các sự kiện $x_1$, $x_2$, $x_3$ lại thành một sự kiện mới là $x_{123}$ có xác suất xuất hiện là 55%, gom sự kiện $x_5$ và $x_6$ lại thành sự kiện $x_{56}$ có xác suất 20%.

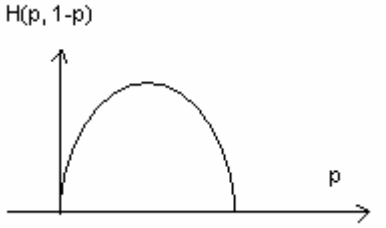
Ta được một nhiến ngẫu nhiên mới $X^*$ có phân phối sau:

| $X^*$ | $x_{123}$ | $x_4$ | $X_{56}$ |
|---|---|---|---|
| P | 55% | 25% | 20% |

Đến đây các bạn có thể áp dụng công thức để tính, so sánh các Entropy và nhận xét tính chất 3. Phần này xem như bài tập cho sinh viên.

## Minh họa tính chất 4:

Để hiểu tính chất thứ 4, ta xét dạng đồ thị của hàm số H(p, 1-p ):




Rõ ràng H(p, 1-p) là một hàm liên tục theo p.

## Định lý cực đại của entropy

**Định lý:** $H(p_1, p_2, \ldots, p_M) \le \log(M)$

Trong đó: đẳng thức xảy ra khi và chỉ khi $p_1 = \ldots = p_M = 1/M$

**Bổ đề:** cho 2 bộ $\{p_1, p_2, \ldots, p_M\}$ và $\{q_1, q_2, \ldots, q_M\}$ là các bộ số dương bất kỳ và

$$\sum_{i=1}^{M} p_i = \sum_{i=1}^{M} q_i$$

Khi đó, ta có $H(p_1, p_2, \ldots, p_M) = -\sum_{i=1}^{M} p_i \log_2 p_i \le -\sum_{i=1}^{M} p_i \log_2 q_i$ (*)

Đẳng thức xảy ra khi $p_i = q_i$ với $\forall i = 1,..,M$.

## Chứng minh định lý cực đại của Entropy

Chứng minh bổ đề:

Theo toán học ta luôn có thể chứng minh được $\ln(x) \le x-1$ với $x>0$ và đẳng thức đúng khi $x=1$.

Đặt $x = q_i/p_i$ Suy ra $\ln(q_i/p_i) \le q_i/p_i - 1$ (và đẳng thức đúng khi $q_i = p_i$ với mọi i).

$$\Leftrightarrow \sum_{i=1}^{M} p_i \ln \frac{q_i}{p_i} \le \sum_{i=1}^{M} (q_i - p_i) = 1 - 1 = 0$$

$$\Leftrightarrow -\sum_{i=1}^{M} p_i \ln p_i \le -\sum_{i=1}^{M} p_i \ln q_i \quad \text{(đẳng thức xảy ra khi } q_i = p_i\text{).} \qquad (1)$$

Theo toán học ta có $\ln x = \log_2 x / \log_2 e$ (2)

Từ (1) và (2), ta có $-\sum_{i=1}^{M} p_i \log p_i \le -\sum_{i=1}^{M} p_i \log q_i$ (đẳng thức xảy ra khi $q_i = p_i$.)

Chứng minh định lý:

Đặt $q_i \frac{1}{M}, \quad \forall i$

Từ bổ đề, ta có:

$$-\sum_{i=1}^{M} p_i \log_2 p_i \le -\sum_{i=1}^{M} p_i \log_2 \frac{1}{M} = \log_2 M \sum_{i=1}^{M} p_i = \log_2 M$$

và đẳng thức chỉ xảy ra khi $p_i=\frac{1}{M}, \quad \forall i$ (đpcm).

## Bài tập

**Bài 1:** Cho 2 biến ngẫu nhiên X, Y độc lập nhau có phân phối sau:

| X | $x_1$ | $x_2$ |
|---|---|---|
| P | 1/2 | 1/2 |

| Y | $y_1$ | $y_2$ | $y_3$ | $y_4$ |
|---|---|---|---|---|
| P | 1/4 | 1/4 | 1/4 | 1/4 |

Tính H(X), H(Y).
**Bài 2:** Kiểm tra lại kết quả của của bài 1 bằng tính chất 2.
**Bài 3:** Cho biến ngẫu nhiên X có phân phối sau:

| X | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $x_6$ |
|---|---|---|---|---|---|---|
| P | 10% | 20% | 25% | 25% | 15% | 5% |

Ta có thể gom các sự kiện $x_1$, $x_2$, $x_3$ lại thành một sự kiện mới là $x_{123}$ có xác suất xuất hiện là 55%, gom sự kiện $x_5$ và $x_6$ lại thành sự kiện $x_{56}$ có xác suất 20%.
Ta được một nhiến ngẫu nhiên mới $X^*$ có phân phối sau:

| $X^*$ | $x_{123}$ | $x_4$ | $x_{56}$ |
|---|---|---|---|
| P | 55% | 25% | 20% |

- Tính entropy của X, $X^*$ và kiểm tra lại tính chất 3.
- Kiểm tra lại định lý cực đại từ dữ liệu cho trên.

# BÀI 2.3: ENTROPY CỦA NHIỀU BIẾN

## Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:

- Hiểu biết các định nghĩa Entropy của nhiều biến và Entropy có điều kiện,
- Hiểu mối quan hệ giữa H(X,Y) với H(X) và H(Y) khi X, Y độc lập,
- Hiểu mối quan hệ giữa H(X,Y) với H(X) và H(Y) khi X, Y tương quan,
- Vận dụng mối quan hệ gữa các Entropy để tính các Entropy một cách hiệu quả,
- Vận dụng Entropy có điều kiện để làm cơ sở tính lượng tin trong bài học kế tiếp

## Định nghĩa Entropy của nhiều biến

Giả sử: X và Y là 2 biến ngẫu nhiên cho trước với $p_{ij} = p(X=x_i,Y=y_j)$ ($\forall$ i=1,..,M và j=1,…,L).

Khi đó, Entropy H(X,Y) có dạng:

$$H(X,Y) = -\sum_{i=1}^{M}\sum_{j=1}^{L} p(x_i, y_j)\log_2 p(x_i, y_j)$$

Hay

$$H(X,Y) = -\sum_{i=1}^{M}\sum_{j=1}^{L} p_{ij}\log_2 p_{ij}$$

Một cách tổng quát:

$$H(x_1,\ldots,x_n) = -\sum_{X_1,\cdots,X_n} p(x_1,...,x_n)\log_2 p(x_1,x_2,...,x_n)$$

## Ví dụ Entropy của nhiều biến

Cho 2 BNN X và Y độc lập nhau và có các phân phối:

| X=1 | 0 | 1 |
|---|---|---|
| P | 0.5 | 0.5 |

| Y | 0 | 1 | 2 |
|---|---|---|---|
| P | 0.25 | 0.5 | 0.25 |

## Tính H(X,Y).

- Lập phân phối của P(X,Y)

| X,Y | X=0,Y=0 | X=0,Y=1 | X=0,Y=2 | X=1,Y=0 | X=1,Y=1 | X=1,Y=2 |
|---|---|---|---|---|---|---|
| P(X,Y) | 0.125 | 0.25 | 0.125 | 0.125 | 0.25 | 0.125 |

- H(X,Y) =H(0.125, 0.25, 0.125, 0.125, 0.25, 0.125)=2.5 (Bit)

## Định nghĩa Entropy có điều kiện

Entropy của Y với điều kiện $X=x_i$ (i=1,..,M) được định nghĩa là:

$$H(Y/X=x_i) = -\sum_{j=1}^{L} p(y_j/x_i)\log p(y_j/x_i)$$

Entropy của Y với điều kiện X xảy ra được định nghĩa là:

$$H(Y/X)=\sum_{i=1}^{M}p(x_i)H(Y/X=x_i)$$

## Ví dụ Entropy có điều kiện

Xét biến ngẫu nhiên X và biến ngẫu nhiên Y có tương quan nhau. Các phân phối như sau:

| X | 1 . | 2 |
|---|---|---|
| P | 0.5 | 0.5 |

Phân phối của Y có điều kiện X:

| Y/X=1 | 0 | 1 | 2 |
|---|---|---|---|
| P | 0.25 | 0.5 | 0.25 |

| Y/X=2 | 0 | 1 | 2 |
|---|---|---|---|
| P | 0 | 0 | 1 |

**Entropy của Y/X=1 và Y/X=2 như sau :**

H(Y/X=1)=H(0.25, 0.5 , 0.25)= -0.25 log0.25 – 0.5 log0.5-0.25 log0.25
=0.5 + 0.5 + 0.5= 1.5 (Bit)

H(Y/X=2)= H(0; 0; 1)= 0 (Bit)

**Entropy của Y khi X xảy ra:**

H(Y/X)=P(X=1) H(Y/X=1)+ P(X=2) H(Y/X=2)=(0.5x1.5) + ((0.5x0)=0.75 (Bit).

## Quan hệ giữa H(X,Y) với H(X) và H(Y) khi X, Y độc lập

**Định lý 1:** H(X,Y)≤ H(X)+H(Y) và đẳng thức xảy ra khi X, Y độc lập

**Chứng minh:**

Ta có:

$$P(x_i)=\sum_{j=1}^{L}p(x_i,y_j)$$

$$P(y_i)=\sum_{i=1}^{M}p(x_i,y_j)$$

$$H(X)=-\sum_{i=1}^{M}p(x_i)\log_2 p(x_i)=-\sum_{i=1}^{M}\sum_{j=1}^{L}p(x_i,y_j)\log_2 p(x_i)$$

$$H(Y)=-\sum_{j=1}^{L}p(y_j)\log_2 p(y_j)=-\sum_{i=1}^{M}\sum_{j=1}^{L}p(x_i,y_j)\log_2 p(y_j)$$

$$\Rightarrow H(X)+H(Y)=-\sum_{i=1}^{M}\sum_{j=1}^{L}p(x_i,y_j)[\log_2 p(x_i)+\log_2 p(y_j)]$$

$$\Rightarrow H(X)+H(Y)=-\sum_{i=1}^{M}\sum_{j=1}^{L}p(x_i,y_j)[\log_2 p(x_i)p(y_j)] \quad (1)$$

Đặt $q_{ij}=p(x_i)p(y_j)$

$$\Rightarrow -\sum_{i=1}^{M}\sum_{j=i}^{L} p_{ij}\log_2 q_{ij} \geq -\sum_{i=1}^{M}\sum_{j=1}^{L} p_{ij}\log_2 p_{ij} \qquad (2)$$

Đẳng thức xảy ra khi $p(x_i, y_j)=p_{ij}=q_{ij}=p(x_i)p(y_j)$ hay X , Y độc lập nhau.

(Theo bổ đề định lý cực đại)

Mặt khác:

$$H(X,Y) = -\sum_{i=1}^{M}\sum_{j=1}^{L} p(x_i,y_j)\log_2 p(x_i,y_j) = -\sum_{i=1}^{M}\sum_{j=1}^{L} p_{ij}\log_2 p_{ij} \qquad (3)$$

Từ (1), (2) và (3), ta có $H(X,Y)\leq H(X)+H(Y)$ và đẳng thức xảy ra khi X, Y độc lập (đpcm)

## Hệ quả:

$H(X_1, \ldots, X_n) \leq H(X_1)+\ldots+H(X_n)$

$H(X_1,\ldots X_n; Y_1,\ldots,Y_n) \leq H(X_1,\ldots X_n)+ H(Y_1,\ldots,Y_n)$

## Quan hệ giữa H(X,Y) với H(X) và H(Y) khi X, Y tương quan

**Định lý 2:** H(X,Y)=H(X)+H(Y/X)=H(Y)+H(X/Y).

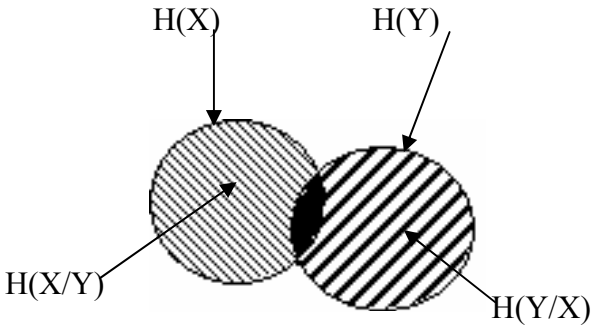
**Định lý 3:** H(Y/X)≤ H(Y) và Dấu đẳng thức xảy ra khi và chỉ khi X và Y độc lập nhau.

**Chứng minh định lý 2:**

$$\begin{aligned} H(X,Y) &= -\sum_{i=1}^{M}\sum_{j=1}^{L} p(x_i,y_j)\log_2 p(x_i,y_j) \\ &= -\sum_{i=1}^{M}\sum_{j=1}^{L} p(x_i,y_j)\log_2[p(x_i).p(y_j/x_i)] \\ &= -\sum_{i=1}^{M}\sum_{j=1}^{L} p(x_i,y_j)\log_2 p(x_i) - \sum_{i=1}^{M}\sum_{j=1}^{L} p(x_i,y_j)\log_2 p(y_j/x_i) \\ &= H(X) + H(Y/X) \end{aligned}$$

Tương tự ta có: H(X,Y)=H(Y)+H(X/Y)

**Chứng minh định lý 3:**
Từ định lý 1 và định lý về quan hệ giữa các Entropy, ta có:
$H(X,Y)=H(X)+H(Y/X)\leq H(X)+H(Y) \Rightarrow H(Y/X) \leq H(Y)$.




Sinh viên tự chứng minh

## Bài tập

Xét BNN X và BNN Y có tương quan nhau. Các phân phối như sau:

| X | 1 . | 2 |
|---|---|---|
| P | 0.5 | 0.5 |

Phân phối của Y có điều kiện X:

| Y/X=1 | 0 | 1 | 2 |
|---|---|---|---|
| P | 0.25 | 0.5 | 0.25 |

| Y/X=2 | 0 | 1 | 2 |
|---|---|---|---|
| P | 0 | 0 | 1 |

1. Tính các Entropy sau: H(X), H(Y).
2. Tính các Entropy có điều kiện sau: H(X/Y), H(Y/X).
3. Tính các Entropy sau: H(X,Y).
4. Từ kết quả câu 1,2 và 3 hãy minh họa các định lý 1, 2 và 3 cho bài học.

# BÀI 2.4: MINH HỌA CÁC ENTROPY

## Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:

- Biết được Yêu cầu của bài toán,
- Biết cách xác định các phân phối ngẫu nhiên của bài toán,
- Vận dụng các bài học trước để tính các Entropy H(X), H(Y) và H(X,Y),
- Vận dụng các bài học trước để tính các Entropy có điều kiện H(X/Y) và H(Y/X),
- Nhận xét và so sánh quan hệ giữa các Entropy
- Ngoài ra còn giúp bạn ôn tập và hiểu rõ hơn các công thức tính Entropy.

## Yêu cầu của bài toán

Ta xét ví dụ về một người tổ chức trò chơi may rủi khách quan với việc tung một đồng tiền "có đầu hình – không có đầu hình". Nếu người chơi chọn mặt không có đầu hình thì thắng khi kết quả tung đồng tiền là không có đầu hình, ngược lại thì thua. Tuy nhiên người tổ chức chơi có thể "ăn gian" bằng cách sử dụng 2 đồng tiền "Thật- Giả" khác nhau sau:

+ Đồng tiền loại 1 (hay đồng tiền thật): đồng chất có 1 mặt có đầu hình.
+ Đồng tiền loại 2 (hay đồng tiền giả ): đồng chất, mỗi mặt đều có 1 đầu hình.

Mặc dù người tổ chức chơi có thể "ăn gian" nhưng quá trình trao đổi 2 đồng tiền cho nhau là ngẫu nhiêu, vậy liệu người tổ chức chơi có thể "ăn gian" hoàn toàn được không? Hay lượng tin biết và chưa biết của sự kiện lấy một đồng tiền từ 2 đồng tiền nói trên được hiểu như thế nào?

Ta thử xét một trường hợp sau: nếu người tổ chức chơi lấy ngẫu nhiên 1 đồng tiền và sau đó thực hiện việc tung đồng tiền lấy được 2 lần. Qua 2 lần tung đồng tiền, ta đếm được số đầu hình xuất hiện. Dựa vào số đầu hình xuất hiện, ta có thể phán đoán được người tổ chức chơi đã lấy được đồng tiền nào.

Chẳng hạn: Nếu số đầu hình đếm được sau 2 lần tưng là 1 thì đồng tiền đã lấy được là đồng tiền thật, ngược lại nếu số đầu hình đếm được là 2 thì đồng tiền đã lấy được có thể là thật hay cũng có thể là giả. Như vậy, ta đã nhận được một phần thông tin về loại đồng tiền qua số đầu hình đếm được sau 2 lần tung. Ta có thể tính được lượng tin đó bằng bao nhiêu? *(Việc tính lượng tin này sẽ được thảo luận sau)*.

## Xác định các phân phối ngẫu nhiên của bài toán

Đặt X là biến ngẫu nhiên về loại đồng tiền.
Phân phối của X:

| X | 1 | 2 |
|---|---|---|
| P | 0.5 | 0.5 |

Đặt biến ngẫu nhiên Y là số đầu hình đếm được sau 2 lần tung:

Phân phối của Y khi nhận được đồng tiền có 1 mặt có đầu hình (Y/X=1)

| Y/X=1 | 0 | 1 | 2 |
|---|---|---|---|
| P | 0.25 | 0.5 | 0.25 |

Phân phối của Y khi nhận được đồng tiền có 2 mặt đều có đầu hình (Y/X=2)

| Y/X=2 | 0 | 1 | 2 |
|---|---|---|---|
| P | 0 | 0 | 1 |

Tìm phân phối của Y:

P(Y=0) = p(X=1)p(Y=0/X=1)+p(X=2)p(Y=0/X=2) = 0,5 x 0,25 +0,5 x 0 =0.125
P(Y=1) = p(X=1)p(Y=1/X=1)+p(X=2)p(Y=1/X=2) = 0,5 x 0,5 +0,5 x 0 =0.250
P(Y=2) = p(X=1)p(Y=2/X=1)+p(X=2)p(Y=2/X=2) = 0,5 x 0,25 + 0,5 x 1=0.625

| Y | 0 | 1 | 2 |
|---|---|---|---|
| P | 0.125 | 0.25 | 0.625 |

## Minh họa Entropy H(X), H(Y) và H(X,Y)

Entropy của X:
H(X) = H(0.5, 05)
= -(0.5)log(0.5) -(0.5)log(0.5) = 1 (bit)
Entropy của Y:
H(X) = H(0.125, 0.25, 0.625)
= -(0.125)log(0.125) + (0.25)log(0.25) + (0.625)log(0.625) = 1.2988 (bit)

Entropy của X và Y: H(X,Y)
Xem như bài tập dành cho các bạn sinh viên
Entropy của Y/X là trung bình của các entropy $Y/X=x_i$.
Vậy, Entropy của Y có điều kiện X: $H(Y/X)=\sum_{i=1}^{M} p(x_i).H(Y/X=x_i)$
Tương tự: H(Y,Z/X), H(Z/X,Y)

## Minh họa Entropy H(X/Y) và H(Y/X)

## Tính Entropy của Y khi biết X: H(Y/X)

H(Y/X=1) = H(0.25, 0.5 , 0.25)
= -(0.25log0.25 + 0.5log0.5 + 0.25log0.25)= 1.5 (bit)
H(Y/X=2)= H(0, 0, 1)= 0
H(Y/X)= p(X=1)H(Y/X=1)+ p(X=2)H(Y/X=2)= 0.5 x 1.5 + 0.5 x 0= 0.75 (bit)

## Tính Entropy của X khi biết Y: H(X/Y)

Xem như bài tập dành cho các bạn sinh viên (Gợp ý: bạn nên lập các phân phối cho các trường hợp (X/Y=0), (X/Y=1) và (X/Y=2).

## Minh họa quan hệ giữa các Entropy

Xem như bài tập dành cho các bạn sinh viên.
Gợi ý: sau khi bạn tính H(X,Y) và H(X/Y), bạn dựa vào các định lý 1,2 và 3 cùng với các kết quả đã tính được để so sánh và minh họa.

# BAI 2.5: ĐO LƯỢNG TIN (MESURE OF INFORMATION)

## Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:

- Biết bài toán tính lượng tin,
- Hiểu định nghĩa lượng tin,
- Biết cách tính lượng tin,
- Có thể vận dụng để tính lượng tin cho các bài toán tương tự.

## Đặt vấn đề bài toán

Ta xét ví dụ về một người tổ chức trò chơi may rủi khách quan với việc tung một đồng tiền "có đầu hình – không có đầu hình". Nếu người chơi chọn mặt không có đầu hình thì thắng khi kết quả tung đồng tiền là không có đầu hình, ngược lại thì thua. Tuy nhiên người tổ chức chơi có thể "ăn gian" bằng cách sử dụng 2 đồng tiền "Thật- Giả" khác nhau sau:

+ Đồng tiền loại 1 (hay đồng tiền thật): đồng chất có 1 mặt có đầu hình.
+ Đồng tiền loại 2 (hay đồng tiền giả ): đồng chất, mỗi mặt đều có 1 đầu hình.

Mặc dù người tổ chơi có thể "ăn gian" nhưng quá trình trao đổi 2 đồng tiền cho nhau là ngẫu nhiêu, vậy liệu người tổ chức chơi có thể "ăn gian" hoàn toàn được không? Ta thử xét một trường hợp sau: nếu người chơi lấy ngẫu nhiên 1 đồng tiền và sau đó thực hiện việc tung đồng tiền lấy được 2 lần. Qua 2 lần tung đồng tiền, ta đếm được số đầu hình xuất hiện. Dựa vào số đầu hình xuất hiện, hãy tính lượng tin về loại đồng tiền lấy được là bao nhiêu?

## Xác định các phân phối của bài toán

Đặt biến ngẫu nhiên X là loại đồng tiền, khi đó phân phối của X có dạng :

| X | 1 | 2 |
|---|---|---|
| P | 0.5 | 0.5 |

Đặt biến ngẫu nhiên Y là số đầu hình đếm được sau 2 lần tung. Khi đó ta có thể xác định được phân phối của Y trong 2 trường hợp sau.

Trường hợp 1: Phân phối của Y khi biết đồng tiền là thật (X=1) có dạng:

| Y/X=1 | 0 | 1 | 2 |
|---|---|---|---|
| P | 0.25 | 0.5 | 0.25 |

Trường hợp 2: Phân phối của Y khi biết đồng tiền là giả (X=2) có dạng:

| Y/X=2 | 0 | 1 | 2 |
|---|---|---|---|
| P | 0 | 0 | 1 |

Ta có thể tính dễ dàng phân phối của Y như sau:

| Y | 0 | 1 | 2 |
|---|---|---|---|
| P | 0.125 | 0.25 | 0.625 |

## Nhận xét dựa theo entropy

Từ các bảng phân phối trên, ta có:

Entropy của Y:

$H(Y) = H(0.125, 0.25, 0.625) = 1.3$ (bit)

Entropy của Y khi biết X

$H(Y/X=1) = H(0.25, 0.5 , 0.25)= 1.5$ (bit)

$H(Y/X=2)= H(0, 0, 1)= 0$

$H(Y/X)= p(X=1)H(Y/X=1)+ p(X=2)H(Y/X=2) = 0.75$ (bit)

Vậy, $H(Y) > H(Y/X)$

## Định nghĩa lượng tin

Từ nhận xét về quan hệ giữa các entropy ở trên, ta có thể định nghĩa lượng tin như sau:

**Định nghĩa:** Lượng tin (hay thông lượng) của X khi Y xảy ra là lượng chênh lệch giữa lượng không chắc chắn của X và lượng không chắc chắn của X khi Y xảy ra có quan hệ với X.

Ta có thể hiểu khái niệm này như sau: X và Y là 2 biến ngẫu nhiên nên chúng có 2 lượng tin không chắc chắn. Nếu X và Y độc lập, thì X xảy ra không ảnh hưởng tới Y nên ta vẫn không biết gì thêm về X và X giữ nguyên lượng không chắc chắn của nó. Trong trường hợp này lượng tin về X khi Y xảy ra là bằng 0. Nếu Y có tương quan với X thì khi Y xảy ra ta biết hoàn toàn về Y và một phần thông tin về X. Phần thông tin đó chính là lượng tin đã biết về X nhưng vẫn chưa biết hết về X. Bài toán ở đây là tính lượng tin đã biết về X khi Y xảy ra.

**Ký hiệu:** $I(X/Y) = H(X)-H(X/Y)$ là lượng tin đã biết về X khi Y đã xảy ra.

**Chú ý:** ta luôn có $I(X/Y) = I(Y/X)$

**Ví dụ:** xét lại ví dụ trên, ta có lượng tin về X khi biết Y là

$I(X/Y)= I(Y/X)= H(Y) – H(Y/X) = 1.3 – 0.75=0.55$ (bit).

## Bài tập

**1.** Thực hiện một phép thử con xúc sắc đồng chất đồng thời với một đồng tiền cũng đồng chất. Trong đó, con xúc sắc có các mặt điểm từ 1 đến 6, đồng tiền một mặt có đầu hình và mặt kia không có đầu hình. Trước tiên thử con xúc sắc, nếu số điểm ≤ 4 thì tung đồng tiền một lần, ngược lại thì tung đồng tiền hai lần. Tính lượng tin về số điểm con xúc sắc khi biết thông tin về số đầu hình đếm được.

**2.** Người ta thực hiện một khảo sát trên các sinh viên đại học về mối quan hệ giữa khả năng học tập với sở hữu phương tiện đi lại và tinh thần ái hữu. Kết quả cho thấy:
Trong tổng số sinh viên có 3/4 sinh viên hoàn thành chương trình học và 1/4 không hoàn thành. Trong số sinh viên hoàn thành chương trình học, 10% có xe con. Ngược lại, trong số sinh viên không hoàn thành chương trình học có tới 50% có xe con.

Tất cả sinh viên có xe con đều tham gia hội ái hữu sinh viên. Trong số sinh viên không có xe con (kể cả hoàn thành hay không hoàn thành khóa học) thì 40% sinh viên tham gia hội ái hữu sinh viên.

a. Tìm thông tin về trạng thái học tập của sinh viên khi biết điều kiện về phương tiện đi lại của họ.
b. Tìm thông tin về tình trạng học tập của sinh viên khi biết tinh thần ái hữu của họ.

**3.** Những người dân của một làng được chia làm 2 nhóm A và B. Một nửa nhóm A chuyên nói thật, 3/10 nói dối và 2/10 từ trối trả lời. Trong nhóm B: 3/10 nói thật, 1/2 nói dối và 2/10 từ trối trả lời. Giả sử p là xác suất chọn 1 người thuộc nhóm A và I(p) = I(Y/X) là lượng tin về người nói thật sau khi đã chọn nhóm, tính I(p), tìm $p^*$ sao $I(p^*) = Max(I(p)$ và tính $I(p^*)$.

# CHƯƠNG 3: SINH MÃ TÁCH ĐƯỢC (Decypherable Coding)

### Mục tiêu:

Phân này đề cập đến bài toán mã hóa (coding) các giá trị của một biến X. Khi mã các giá trị của X người ta phải sử dụng bảng ký tự mã (Coding Character Table) hay bảng chữ cái (Code Alphabet). Như vậy, một giá trị x của X sẽ được mã thành một từ mã (Code Word) w dưới dạng một dãy các ký tự mã với độ dài là n ký tự. Trong truyền tin, một dãy các giá trị của X được phát sinh và được mã thành một dãy liên tục các từ mã hay một dãy các ký tự mã lấy từ bảng ký tự mã. Vấn đề cần giải quyết là:

1. Khi nhận một dãy ký tự mã liên tục đó thì ta có thể giải mã thành một dãy các giá trị duy nhất của X hay không ? Nói cách khác, dãy ký tự mã này có tách được thành các từ mã một cách duy nhất hay không ?
2. Chỉ ra phương pháp xây dựng mã tách được tối ưu.

# BÀI 3.1: KHÁI NIỆM VỀ MÃ TÁCH ĐƯỢC

### Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:

- Biết yêu cầu của bài toán sinh mã,
- Hiểu khái niệm về bảng mã tách được và bảng mã không tách được,
- Hiểu khái niệm về bảng mã tức thời,
- Hiểu giải thuật kiểm tra tính tách được của một bảng mã,
- Vận dụng giải thuật kiểm tra tính tách được của một bảng mã để kiểm tra xem một bảng mã có phải là bảng mã tách được hay không.

### Đặt vấn đề bài toán sinh mã

Giả sử nguồn tin X xuất hiện và được ghi lại thông qua một thiết bị đặc biệt. Chẳng hạn như ảnh được ghi lại bằng máy ảnh, âm thanh được ghi lại bằng máy ghi âm, … Qua kênh truyền, những thông tin này cần phải được mã hóa cho phù hợp. Để có thể mã hóa người ta cần một bảng chữ cái gồm các chữ cái quy định trước (chẳng hạn bảng chữ cái la tinh, bảng mã nhị phân, … ). Mỗi giá trị của X sau đó được mã dưới dạng một dãy hữu hạn các chữ cái và ta gọi dãy hữu hạn các chữ cái gán cho một giá trị của x là một từ mã.

Ta xét BNN $X=\{x_1, x_2, \ldots, x_n\}$ có phân phối $\{p_1, p_2, \ldots, p_n\}$ được quan sát liên tục và độc lập. Dãy các giá trị nhận được gọi là thông báo (Message) có dạng $x_{i1}x_{i2}\ldots x_{in}$. Tập hợp $A=\{a_1, a_2, \ldots, a_n\}$ là tập hợp ký tự mã (Code Characters) hay là bảng chữ cái (Code Alphabet) dùng để sinh mã. Một giá trị $x_i \in X$ được gán bởi một dãy hữu hạn các ký tự mã được gọi là từ mã (Code word). Tập hợp gồm tất cả các từ mã gán cho tất cả các giá trị của X được gọi là bộ mã hay bảng mã (Code). Các từ mã phải khác nhau từng đôi một.

Bộ mã được gọi là tách được nếu như từ một dãy các ký tự mã nhận được liên tục (được mã hóa từ bộ mã này), ta luôn luôn giải mã được với kết quả duy nhất là dãy các giá trị gốc của X.

Shannon (1948) lần đầu tiên đã đưa ra định lý cơ sở về sinh mã tách được. Mc Millan (1956) đã chứng minh định lý về điều kiện cần và đủ của bảng mã tách được. Nhưng vấn đề sinh mã tách được chỉ được xét một cách chuẩn mực bởi Feinstein (1958), Abramson (1963) và Fano (1961). Sardinas(1960) và Patterson (1963) đã đưa ra định lý về giải thuật kiểm tra tính tách được của một bảng mã. Abramson (1963) đã đưa ra khái niệm bảng mã tức thời.

Trong phạm vi bài giảng này, bài toán sinh mã tối ưu được đặt ra ở đây là tìm ra một phương pháp sinh mã sao cho độ dài trung bình của các từ mã trong bộ mã là nhỏ nhất. Nghĩa là, nếu giá trị $x_i$ được gán bởi từ mã có độ dài $n_i$ thì bài toán sinh mã phải thỏa:

$$\sum_{i=1}^{n} p_i n_i \rightarrow Min$$

Huffman (1950) đã đưa ra qui trình xây dựng một bảng mã tối ưu thỏa yêu cầu này.

## Khái niệm về bảng mã không tách được

Bảng mã không tách được là bảng mã mà khi mã hóa thông báo **Msg** ta sẽ nhận được một dãy các từ mã **ws**, và khi giải mã dãy các từ mã **ws** thì ta có thể nhận được nhiều thông báo **Msg** khác nhau.

**Ví dụ:** Xét biến ngẫu nhiên $X=\{x_1, x_2, x_3, x_4\}$ có bảng mã $W=\{w_1=0, w_2=1, w_3=01, w_4=10\}$.

Giả sử thông báo nguồn có nội dung: $x_1x_2x_3x_4x_3x_2x_1$. Khi đó dãy mã tương ứng viết từ W có dạng: 0101100110.

Nếu giải mã tuần tự từ trái qua phải ta nhận kết quả: $x_1x_2x_1x_2x_2x_1x_1x_2x_2x_1$. Nhưng nếu bằng phương pháp khác ta có thể nhận được kết quả: $x_3x_3x_4x_3x_4$ và nhiều thông báo khác nữa.

**Nhận xét:** Bảng mã giải mã không tách được là bảng mã mà trong đó tồn tại ít nhất một từ mã này là mã khóa của một hay nhiều từ mã khác trong bộ mã (ví dụ từ mã $w_1=0$ hay $w_2=1$ là mã khóa của $w_3$).

## Bảng mã tách được

Bảng mã tách được là bảng mã mà khi mã hóa thông báo **Msg** ta sẽ nhận được dãy các từ mã **ws**, và khi giải mã dãy các từ mã **ws** thì ta chỉ nhận được một thông báo duy nhất là **Msg** ban đầu.

**Ví dụ:** Xét biến ngẫu nhiên $X=\{x_1, x_2\}$ có bảng mã tương ứng $W=\{w_1=0, w_2=01\}$.

Phương pháp giải mã được sử dụng như sau: chỉ giải mã khi nào đã nhận được đoạn mã với độ dài bằng độ dài của từ mã dài nhất.

Giả sử dãy mã nhận được (cần giải mã) là: 0010000101001.
Sử dụng phương pháp giải mã trên ta nhận được duy nhất dãy thông báo gốc:

$x_1x_2x_1x_1x_1x_2x_2x_1x_2$.

Có thể chi tiết hóa các bước giải mã dãy từ mã trên như sau:

Nhận được đoạn 00 -> Giải ra $x_1$ , còn lại 0.
Nhận tiếp 1 ->01 -> Giải ra $x_2$.
Nhận tiếp 00 -> Giải ra $x_1$, còn lại 0.

Nhận tiếp 0 -> 00 -> Giải ra $x_1$, còn lại 0.
Nhận tiếp 0 -> 00 -> Giải ra $x_1$, còn lại 0.
Nhận tiếp 1 -> 01 -> Giải ra $x_2$.
Nhận tiếp 01 -> Giải ra $x_2$.
Nhận tiếp 00 -> Giải ra $x_1$, còn lại 0.
Nhận tiếp 1 -> 01 -> Giải ra $x_2$.
Kết quả dãy thông báo là: $x_1x_2x_1x_1x_1x_2x_2x_1x_2$.

**Kết luận:** Bảng mã tách được là bảng mã mà trong đó không tồn lại từ mã này là mã khóa từ mã khác, tuy nhiên vẫn có thể tồn tại từ mã này là tiền tố (phần đầu) của từ mã kia.

## Khái niệm bảng mã tức thời

Bảng mã tức thời là bảng mã mà khi mã hóa thông báo **Msg** ta sẽ nhận được dãy các từ mã **ws**, và khi giải mã dãy các từ mã **ws** thì ta chỉ nhận được một thông báo duy nhất là **Msg** ban đầu. **Abramson đã chứng minh được kết quả sau: Bảng mã tức thời là bảng mã không tồn tại từ mã này là tiền tố của từ mã khác**.

**Ví dụ 1:** Bảng mã W={$w_1$=10; $w_2$=101; $w_3$=100} không phải bảng mã tức thời vì $w_1$ là tiền tố của $w_2$ và $w_3$.
**Ví dụ 2:** Bảng mã W={$w_1$=0, $w_2$=100, $w_3$=101, $w_4$=11} là bảng mã tức thời vì không tồn tại từ mã này là tiền tố của từ mã khác.

## Giải thuật kiểm tra tính tách được của bảng mã

Thủ tục sau đây do Sardinas (1960), Patterson (1963) và Abramson (1963) đưa ra nhằm kiểm tra xem một bảng mã nào đó có phải là bảng mã tách được (bảng mã cho phép giải mã duy nhất) hay không.

**Input**: Bảng mã W
**Output:** Kết luận bảng mã tách được hay không tách được.

## Giải thuật:

*Bước khởi tạo:* Gán tập hợp $S_0$=W.
**Bước 1**: xác định tập hợp $S_1$ từ $S_0$:
- Khởi tạo $S_1$={}
- Với $\forall$ $w_i$, $w_j \in S_0$, ta xét: nếu $w_i$=$w_j$A ($w_j$ là tiền tố của $w_i$) hoặc $w_j$=$w_i$ A ($w_i$ là tiền tố của $w_j$) thì thêm A (phần hậu tố) vào $S_1$.

**Bước k**: xác định tập hợp $S_k$ (k≥2) từ tập hợp $S_0$ và $S_{k-1}$:
- Khởi tạo: $S_k$={}
- Với $\forall$ $w_i \in S_0$ và $\forall$ $v_j \in S_{k-1}$, ta xét: nếu $w_i$=$v_j$A ($v_j$ là tiền tố của $w_i$) hoặc $v_j$=$w_i$ A ($w_i$ là tiền tố của $v_j$) thì thêm A (phần hậu tố) vào $S_k$.

Điều kiện để dừng vòng lặp:
Nếu $S_k$={} thì dừng và kết luận bảng mã tách được (k≥1).
Nếu tồn tại từ mã $w_i$ trong $S_k$ hay $S_k \cap S_0 \neq \varnothing$ thì dừng và kết luận bảng mã không tách được.
Nếu $S_k$=$S_{t<k}$ thì dừng và kết luận bảng mã tách được (k≥1).
Bài toán 1- yêu cầu

**Bài toán:** Kiểm tra xem bảng mã W={a, c, ad, abb, bad, deb, bbcde} có phải là bảng mã tách được hay không?

Áp dụng Giải thuật kiểm tra tính tách được của một bảng mã:
**Bước khởi tạo:** S0={a, c, ad, abb, bad, deb, bbcde}
**Bước 1:** Tính S1
Khởi tạo S1={}
Vì a là tiền tố của ad nên đưa phần hậu tố "d" vào S1 => S1={d}.
Vì a là tiền tố của abb nên đưa phần hậu tố "bb" vào S1 => S1={d, bb}.
Kiểm tra điều kiện dừng: không thỏa -> qua bước 2.

**Bước 2:** Tính S2 từ S0 và S1.
Khởi tạo S2={}.
Vì d ∈ S1 là tiền tố của deb ∈ S0 nên đưa phần hậu tố "eb" vào S2
=> S2={eb}
Vì bb∈ S1 là tiền tố của bbcde ∈ S0 nên đưa phần hậu tố "cde" vào S2
=> S2={eb, cde}
Kiểm tra điều kiện dừng: không thỏa -> qua bước 3.
Bài toán 1 - Áp dụng giải thuật
**Bước 3:** Tính S3 từ S0 và S2.
Khởi tạo S3={}.
Vì c∈ S0 là tiền tố của cde ∈ S2 nên đưa phần hậu tố "de" vào S3
=> S3={de}
Kiểm tra điều kiện dừng: không thỏa -> qua bước 4.
**Bước 4:** Tính S4 từ S0 và S3.
Khởi tạo S4={}.
Vì de∈ S3 là tiền tố của deb ∈ S0 nên đưa phần hậu tố "b" vào S4
=> S4={b}
Kiểm tra điều kiện dừng: không thỏa -> qua bước 5.

**Bước 5:** Tính S5 từ S0 và S4.
+ khởi tạo S5={}.
+ Vì b∈ S4 là tiền tố của bad ∈ S0 nên đưa phần hậu tố "ad" vào S5 => S5={ad}
+ Vì b∈ S4 là tiền tố của bbcde ∈ S0 nên đưa "bcde" vào S5
=> S5={ad, bcde}
Kiểm tra điều kiện dừng: Vì S5 có chứa từ mã ad nên dừng lại và kết luận đây là bảng mã không tách được.
Bài toán 2
**Bài toán:** Kiểm tra xem bảng mã W={010, 0001, 0110, 1100, 00011, 00110, 11110, 101011} có phải là bảng mã tách được không?

Áp dụng Giải thuật kiểm tra tính tách được của một bảng mã:
Bước khởi tạo và bước 1
- Tập hợp $S_0$ ={010, 0001, 0110, 1100, 00011, 00110, 11110, 101011}
- Tập hợp $S_1$ ={1}
Dành cho sinh viên tự làm các bước tiếp theo.

Kết quả gợi ý:
Tập hợp $S_2$ ={100, 1110, 01011}
Tập hợp $S_3$={11}

Tập hợp $S_4$={00, 110}
Tập hợp $S_5$={01, 0, 011, 110}
Tập hợp $S_6$={0, 10, 001, 110, 0011, 0110}
Tập hợp $S_6$ chứa từ mã 0110 nên bảng mã này không phải là bảng mã tách được.

## Bài tập

1. Hãy cho biết bảng mã sau có phải là bảng mã tách được hay không?
   W={$w_1$=00, $w_2$=01, $w_3$=0010, $w_4$=0111, $w_5$=0110}
2. Hãy lấy ví dụ một bảng mã tách được, và chứng minh nó là bảng mã tách được.

# BÀI 3.2: QUAN HỆ GIỮA MÃ TÁCH ĐƯỢC VÀ ĐỘ DÀI MÃ

## Mục tiêu

Sau khi hoàn tất bài học này bạn có thể hiểu:

- Định lý Kraft (1949),
- Định nghĩa cây bậc D cỡ K,
- Vấn đề sinh mã cho cây bậc D cỡ K,
- Vận dụng định lý Kraff để kiểm tra sự tồn tại bảng mã tách được và sinh bảng mã tách được.

## Định lý Kraftn(1949).

Gọi $X=\{x_1, x_2,\ldots, x_M\}$ là biến ngẫu nhiên chứa các giá trị cần truyền có phân phối là $P=\{p_1, p_2, \ldots, p_M\}$.

$A=\{a_1, a_2,\ldots,a_D\}$ là bộ ký tự sinh mã có D chữ cái (D được gọi là cơ số sinh mã).
Giá trị $x_i$ được mã hóa thành từ mã $w_i$ có độ dài là $n_i$.
Đặt $N=\{n_1, n_2,\ldots,n_M\}$ là tập hợp độ dài các từ mã.

## Định lý (Kraft- 1949):

Điều kiện cần và đủ để tồn tại bảng mã tức thời với độ dài $N=\{n_1,n_2,\ldots,n_M\}$ là

$$\sum_{i=1}^{M} D^{-n_i} \le 1$$

**Ví dụ 1:** Bộ mã $W=\{w_1, w_2, w_3\}$ với $M=3$; $n_1=1$; $n_2=2$; $n_3=3$; $D=2$

$$\sum_{i=1}^{M} D^{-n_i} = \frac{1}{2^1}+\frac{1}{2^2}+\frac{1}{2^3}=\frac{7}{8}<1$$

=> Tồn tại bảng mã tức thời.

## Ví dụ 2: Bộ mã $W=\{w_1, w_2, w_3\}$ với $M=3$; $n_1=n_2=1$; $n_3=2$; $D=2$

$$\sum_{i=1}^{M} D^{-n_i} = \frac{1}{2^1}+\frac{1}{2^1}+\frac{1}{2^2}=\frac{5}{4}>1$$

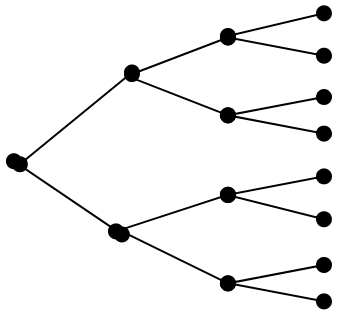
=> Không tồn tại bảng mã tức thời.
Đề nghị: sinh viên tìm hiểu nội dung tiếp theo và trở lại giải thích 2 ví dụ trên.

## Định nghĩa cây bậc D cỡ k.

**Định nghĩa:** Cây bậc D cỡ k là cây có hệ thống nút, cạnh thỏa điều kiện:

- Từ 1 nút có số cạnh đi ra không vượt quá D hay một nút có không quá D nút con.
- Nút cuối cùng (Nút lá) cách nút gốc không vượt quá k cạnh.

**Ví dụ:** cây bậc D=2 và cỡ k=3

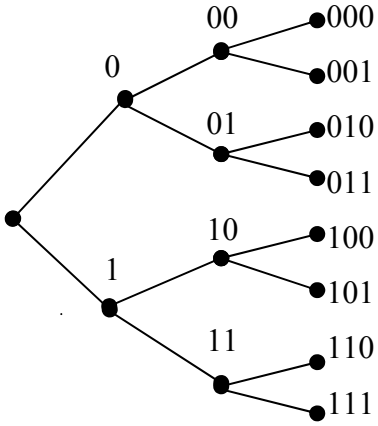


## Vấn đề sinh mã cho cây bậc D cỡ k

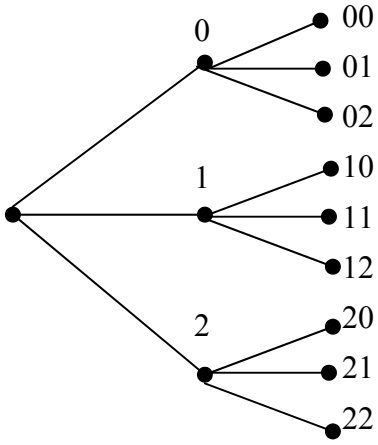
## Sinh mã cho các nút của cây bậc D cỡ K (trừ nút gốc):

Để đơn giản hóa: mỗi nút (trừ nút gốc) được ký hiệu bởi dãy ký hiệu của nút cha làm tiền tố + một ký tự bổ sung lấy từ tập hợp {0, 1, 2, …, D-1} thay cho bảng chữ cái A={$a_1$, $a_2$, …, $a_D$}.

**Ví dụ 1:** Cây bậc D=2 cỡ k=3




**Ví dụ 2:** Cây bậc D=3 cỡ k=2.




## Tính chất:

+ Các nút (trừ nút gốc) của cây đều được mã hóa từ bảng chữ cái {0, 1, 2,…, D-1}
+ Mỗi nút (đã mã hóa) có mã của nút kề trước là tiền tố.
+ Tổng số các nút lá bằng $D^k$ = tổng số các mã tức thời có thể có.

## Chứng minh định lý Kraft (Điều kiện cần)

Giả sử, cho trước bảng mã tức thời W={$w_1$, $w_2$,…, $w_M$} với N={$n_1 \leq n_2 \leq \ldots \leq n_M$}. Ta cần c/m:

$$\sum_{i=1}^{M} D^{-n_i} \leq 1$$

Xây dựng cây bậc D cỡ $n_M$ và sinh mã cho các nút trừ nút gốc với các ký tự mã lấy từ bảng chữ cái A = {0, 1, 2,…, D-1}. Mã tại mỗi nút (trừ nùt gốc) đều có khả năng được chọn là từ mã.

Như vậy, ta tiến hành chọn các từ mã cho bảng mã tức thời với qui tắc là: một nút nào đó được chọn để gán một từ mã thì tất cả các nút kề sau nút gán từ mã phải được xóa. Cụ thể như sau:

Chọn một nút có mã với độ dài mã là $n_1$ gán cho nó một từ mã $w_1$.

=> Tổng số nút lá được xóa tương ứng là $D^{n_M - n_1}$

Chọn một nút có mã với độ dài mã là $n_2$ gán cho nó một từ mã $w_2$.

=> Tổng số nút lá được xóa tương ứng là $D^{n_M - n_2}$

……

Chọn một nút có mã với độ dài mã là $n_n$ gán cho nó một từ mã $w_n$.

=> số nút lá được gán từ mã là $D^{n_M - n_M}$

Vậy số nút lá bị xóa hoặc được gán từ mã là:

$$=> D^{n_M - n_1} + D^{n_M - n_1} + \cdots + D^{n_M - n_M} = \sum_{i=1}^{M} D^{n_M - n_i} \le D^{n_M} = \text{tổng số nút lá.}$$

$$=> \sum_{i=1}^{M} D^{-n_i} \le 1 \text{ (đpcm)}$$

## Chứng minh định lý Kraft (Điều kiện đủ)

Giả sử: $\sum_{i=1}^{M} D^{-n_i} \le 1$, để cần chứng minh tồn tại bảng mã tức thời với $N=\{n_1, n_2, \ldots, n_M\}$, ta chỉ cần chỉ ra thủ tục xây dựng bảng mã tức thời như sau:

## Thủ tục tạo mã tức thời:

Xét $N=\{n_1, n_2, \ldots, n_M\}$ và cơ số sinh mã là D**:**

**Bước 1:** Ta xếp thứ tự $n_1 \le n_2 \le \ldots \le n_M$, xây dựng cây bậc D cỡ $k=n_M$ và sinh mã cho các nút .

**Bước 2:** Chọn nút bất kỳ trên cây có độ dài $n_1$ gán cho từ mã $w_1$ và xóa tất cả các nút kề sau nó.

**Bước 3:** Lặp lại các bước 2 đối với việc chọn các từ mã còn lại $w_2, \ldots, w_M$ ứng với $n_2, \ldots, n_M$.

=> Bảng mã $W=\{w_1, w_2, \ldots, w_M\}$ là bảng mã tức thời.

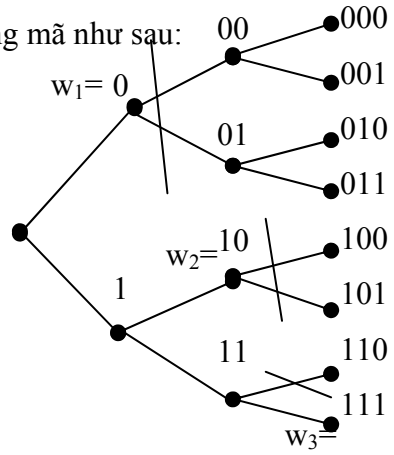
## Ví dụ minh họa định lý Kraft

**Ví dụ 1:** Xét bảng mã thỏa M=3, D=2, $n_1=1$, $n_2=2$, $n_3=3$. Vậy ta kiểm tra xem có tạo được bảng mã tức thời hay không?

Ta có $\sum_{i=1}^{3} 2^{-n_i} = 2^{-1} + 2^{-2} + 2^{-3} = \frac{7}{8} < 1$

=> $W=\{w_1, w_2, w_3\}$ là bảng mã tức thời

Ta Xây dựng bảng mã như sau:




- Chọn $w_1=0$ , cắt bỏ các nút con của nút $w_1$.
- Chọn $w_2=10$, cắt bỏ các nút con của nút $w_2$.
- Chọn $w_3=111$

**Chú ý:** ngoài bảng mã tức thời chọn được ở trên, ta còn có thể sinh được nhiều bảng mã tức thời khác. *Đề nghị sinh viên đưa ra bảng mã tức thời khác.*

## Bài tập

1. Tìm 1 bảng mã tách được thỏa tính chất D = 2, k = 4?
2. Tìm tất cả các bảng mã tách được thỏa tính chất D=2, k=3?
3. Hãy chỉ ra bảng mã sau đây là bảng mã không tách được:
   W={w1=00, w2=1, w3=100, w4=110, w5=111}
4. Hãy tìm một bảng mã nhị phân tách được có ít nhất 5 từ mã thỏa điều kiện

$$\sum_{i=1}^{M} D^{-n_i} = 1$$

# BÀI 3.3: TÍNH TỐI ƯU CỦA ĐỘ DÀI MÃ

## Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:

- Hiểu định lý Shannon (1948),
- Biết được các tiêu chuẩn đánh giá bảng mã tối ưu tuyệt đối và bảng mã tối ưu tương đối,
- Điều kiện nhận biết một bảng mã tối ưu,
- Hiểu Định lý Huffman,
- Biết Phương pháp sinh mã Huffman,
- Vận dụng phương pháp sinh mã Huffman để sinh mã Huffman cho một thông báo,
- Vận dụng phương pháp sinh mã Huffman để viết chương trình nén.

## Định lý Shannon (1948)

Phát biểu định lý:

Đặt $\bar{n} = \sum_{i=1}^{M} p_i n_i$ là độ dài trung bình của bảng mã.

Khi đó $\bar{n} \geq \frac{H(X)}{\log_2 D}$

Dấu đẳng thức xảy ra khi và chỉ khi $p_i = D^{-n_i}$ hay $\sum_{i=1}^{M} D^{-n_i} = 1$

**Diễn giải**: Đối với mã tách được độ dài trung bình của mã sẽ có cận dưới là $\frac{H(X)}{\log_2 D}$. Nếu mã không tách được độ dài trung bình của nó có thể nhỏ hơn cận dưới. Nếu mã tách được không tối ưu thì độ dài của nó sẽ lớn hơn nhiều so với cận dưới, còn nếu mã tách được tối ưu thì độ dài trung bình của nó gần với cận dưới.
Bài toán đặt ra sẽ là tìm phương pháp xây dựng bảng mã tách được tối ưu.
Chú ý:

$$H_D(X) = -\sum p_i \log_D p_i$$

$$H_D(X) = \frac{H(X)}{\log_2 D} = \frac{-\sum p_i \log_2 p_i}{\log_2 D}$$

là entropy của X với cơ số D.

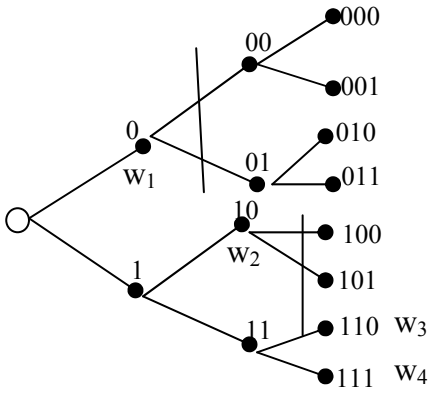
## Bảng mã tối ưu tuyệt đối

**Định lý:** Bảng mã được gọi là tối ưu tuyệt đối khi $\bar{n} = \frac{H(X)}{\log_2^D}$ hay $p_i = D^{-n_i}$

**Ví dụ:** xét biến ngẫu nhiên X={$x_1$, $x_2$, $x_3$, $x_4$}
Có phân phối: P={1/2, 1/4, 1/8, 1/8}
Có bảng mã W={$w_1$= 0, $w_2$=10, $w_3$=110, $w_4$=111}

Ta tính được độ dài trung bình từ mã: $\bar{n} = \frac{1}{2}*1 + \frac{1}{4}*2 + \frac{1}{8}*3 + \frac{1}{8}*3 = \frac{12}{8} = 1.75$

Tính Entropy của X: H(X)= H(0.5, 0.25, 0.125, 0.125) = 0.5 +0.5 + 0.375 + 0.375 =1.75
$Log_2D$=1.




W= {$w_1$, $w_2$, $w_3$, $w_4$} là bảng mã tối ưu tuyệt đối vì thỏa điều kiện:

$$\bar{n} = \frac{H(X)}{\log_2^D}$$

## Bảng mã tối ưu tương đối

**Định lý:** Bảng mã được gọi là tối ưu tương đối khi: $\frac{H(X)}{\log_2 D} \le \bar{n} < \frac{H(X)}{\log_2 D} + 1$

## Điều kiện nhận biết một bảng mã tối ưu

## Định lý (với D=2):

- Xác suất $p_i$ càng lớn thì độ dài $n_i$ của từ mã $w_i$ càng nhỏ.
- Giả sử $p_1 \ge p_2 \ge \ldots \ge p_M$. Nếu $p_i \ge p_{i+1} \ge p_{i+r}$ thì $n_i \le n_{i+1} \le n_{i+r}$ thì 2 từ mã tương ứng với 2 giá trị có xác suất nhỏ nhất có độ dài mã bằng nhau $n_{M-1} = n_M$.
- Trong các từ mã có độ dài bằng nhau và cùng bằng $n_M$ (dài nhất) thì tồn tại ít nhất 2 từ mã $w_{M-1}$ và $w_M$ có M-1 ký tự đầu giống nhau và ký tự thứ M khác nhau.

**Ví dụ:** xét bảng mã W={$w_1$=0, $w_2$=100, $w_3$=101, $w_4$=1101, $w_5$=1110}
Bảng mã trên không phải là bảng mã tối ưu vì 2 từ mã $w_4$, $w_5$ có độ dài lớn nhất =4 ký tự nhưng 3 ký tự đầu khác nhau.

**Ghi chú:** D > 2 được xét tương tự.

## Định lý Huffman

**Định lý:** Giả sử X có phân phối xác suất với thứ tự giảm dần sau:

| X | $x_1$ | $x_2$ | … | $x_M$ |
|---|---|---|---|---|
| P | $p_1 \ge$ | $p_2 \ge$ | … | $\ge p_M$ |

Giả sử bảng mã của X là W={$w_1$, $w_2$, …, $w_{M-1}$, $w_M$}.
Đặt $x_{M-1,M}$={$x_{M-1}$, $x_M$} có xác suất là $p_{M-1,M}=p_{M-1}+p_M$.
và $X^*$ = { $x_1$, $x_2$,…, $x_{M-1,M}$} có phân phối sau:

| X* | $x_1$ | $x_2$ | … | $x^*_{M-2}$ | $x^*_{M-1,M}$ |
|---|---|---|---|---|---|
| P | $P_1$ | $p_2$ | … | $p^*_{M-2}$ | $p^*_{M-1,M}$ |

Giả sử W* ={$w_1$, $w_2$, …, $w_{M-2}$, $x^*_{M-1,M}$} là bảng mã tối ưu của X*. Khi đó:
- $w_{M-1}=w^*_{M-1,M}$ + "0".
- $w_M = w^*_{M-1,M}$ + "1".

## Phương pháp sinh mã Huffman

Giả sử X có phân phối xác suất với thứ tự giảm dần sau:

| X | $x_1$ | $x_2$ | … | $x_M$ |
|---|---|---|---|---|
| P | $p_1 \geq$ | $p_2 \geq$ | … | $\geq p_M$ |

Thủ tục lùi (D=2):

**Khởi tạo:** Đặt $M_0$=M

Bước 1:

- Đặt $x_{M_0-1,M_0} = \{x_{M_0-1}, x_{M_0}\}$ có xác suất $p_{M_0-1,M_0} = p_{M_0-1} + p_{M_0}$
- Sắp xếp lại theo tứ tự giảm dần của xác suất ta nhận được dãy phân phối mới có $M_0$-1 phần tử như sau: $p_1, p_2, \cdots, p_{M_0-2}, p_{M_0-1,M_0}$

**Bước 2:** Lặp lại bước 1 với sự lưu vết

$$w_{M_0-1} = w_{M_0-1,M_0} + "0"$$

$$w_{M_0} = w_{M_0-1,M_0} + "1"$$

Giảm $M_0$: $M_0$=$M_0$-1, vòng lặp kết thúc khi $M_0$=2

(**Chú ý:** trong trường hợp tổng quát, vong lặp kết thúc khi $M_0 \leq$ D.)

Thủ tục tiến:

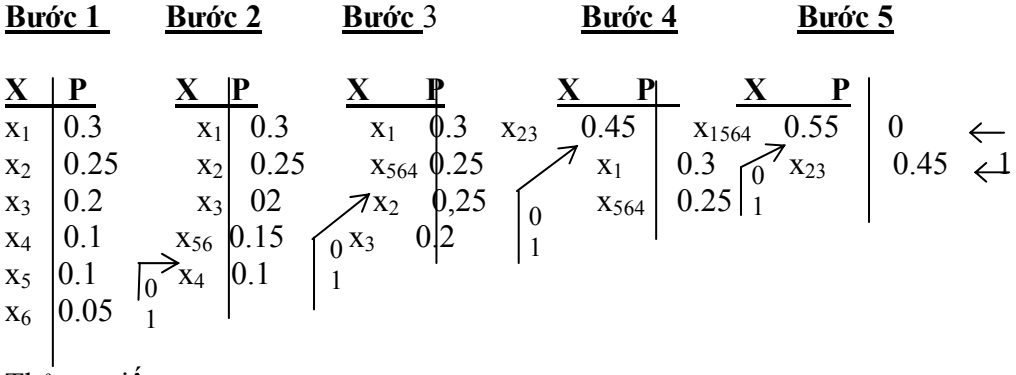
Đi ngược lại so với thủ tục lùi đồng thời xác định từ mã ở mỗi bước từ sự lưu vết ở thủ tục lùi.

Minh họa phương pháp sinh mã Huffman

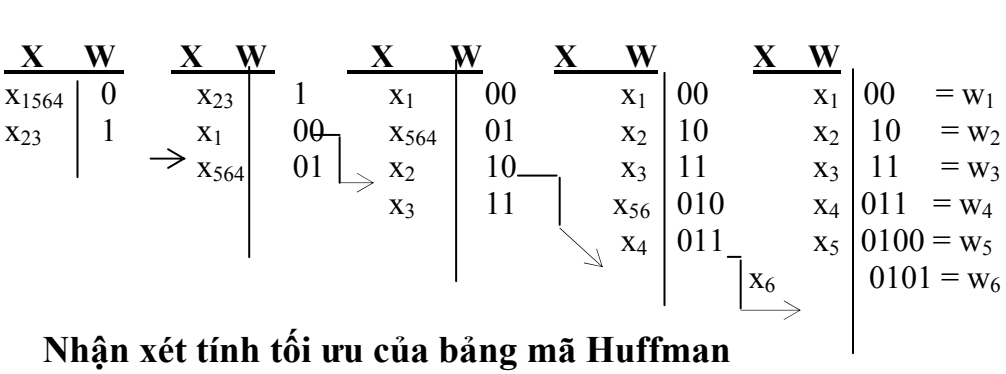
**Ví dụ 1:** sinh bảng mã nhị phân Huffman cho X có phân phối sau:

| X | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $x_6$ |
|---|---|---|---|---|---|---|
| P | 0.3 | 0.25 | 0.2 | 0.1 | 0.1 | 0.05 |

Thủ tục lùi:

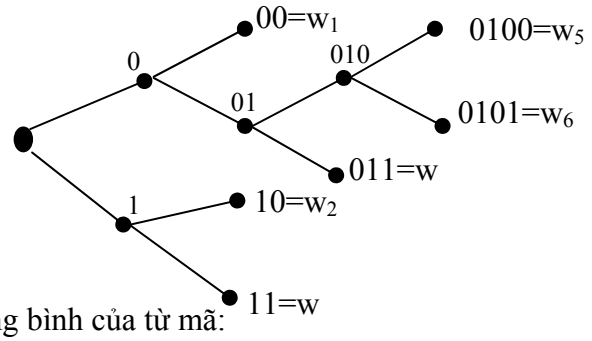



Thủ tục tiến:




## Nhận xét tính tối ưu của bảng mã Huffman

Vẽ cây Huffman của bảng mã trên:




Độ dài trung bình của từ mã:

$\bar{n}$ =(0.3 x 2)+ (0.25 x 2)+ (0.2 x 2) + (0.1 x 3) +(0.1 x 4) + (0.05 x 4) = 2.4

**Entropy của X:** H(X) = H(0.3, 0.25; 0.2, 0.1,0.1, 0.05)
= 2.4

**Nhận xét:** Do D = 2 và $\log_2 D=1$, Ta có $\bar{n}$ = H(X) nên bảng mã trên tối ưu tuyệt đối.

## Bài tập

1. Cho biến ngẫu nhiên X có phân phối sau:

| X | $x_1$ | $x_2$ | $x_3$ | $x_4$ |
|---|---|---|---|---|
| P | 0.4 | 0.3 | 0.2 | 0.1 |

2. Cho biến ngẫu nhiên Y có phân phối sau:

| Y | $y_1$ | $y_2$ | $y_3$ | $y_4$ | $y_5$ | $y_6$ | $y_7$ | $y_8$ | $y_9$ |
|---|---|---|---|---|---|---|---|---|---|
| P | 0.3 | 0.2 | 0.2 | 0.1 | 0.05 | 0.05 | 0.04 | 0.03 | 0.03 |

3. Cho đoạn văn bản "thoi the thi thoi thi the thoi thi the". Tìm bảng mã nhị phân Huffman dùng để mã hóa đoạn văn bản trên.
4. Thay từng ký tự trong đoạn văn bản trên thành một từ mã, cắt từng đoạn 8 bits đổi thành số thập phân. Cho biết dãy số thập phân kết quả.

# CHƯƠNG 4: KÊNH TRUYỀN

## Mục tiêu:

Trình bày mô hình truyền tin rời rạc từng ký tự mã độc lập lẫn nhau (phù hợp với đặc điểm của kênh). Mô hình này còn gọi là kênh truyền rời rạc không nhớ (Memoryless Discret Channel). Từ mô hình này người ta có thể xây dựng cách tính dung lượng kênh truyền và phương pháp phân loại đầu nhận để có thể giải mã tốt nhất.

## BÀI 4.1: KÊNH TRUYỀN RỜI RẠC KHÔNG NHỚ

### Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:

- Biết mô hình kênh truyền tin rời rạc không nhớ ở 2 khía cạnh vật lý và toán học.
- Khái niệm về lượng tin trên kênh truyền
- Định nghĩa dung lượng kênh truyền

### Giới thiệu

Trước hết, ta có thể hiểu khái niệm kênh truyền rời rạc và không nhớ ở bài học này như sau: khái niệm truyền rời rạc ở đây là truyền tuần tự các ký tự độc lập nhau (hay truyền từng ký tự một), còn khái niệm không nhớ ở đây là chỉ xét mối quan hệ giữa ký tự truyền và ký tự nhận được tương ứng, không xét đến mối quan hệ giữa ký tự nhận được với ký tự nhận được trước đó.

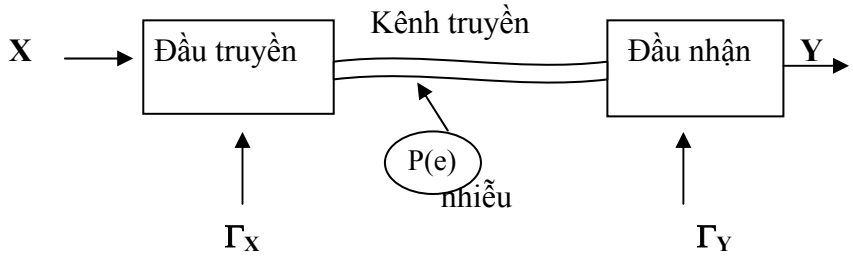
Khái niệm về một kênh truyền rời rạc dựa vào phân bố xác suất của tín hiệu ra phụ thuộc vào tín hiệu vào và trạng thái của kênh truyền đã được chuẩn hóa bởi Feinstein (1958) và Wolfowitz (1961). Dung lượng kênh (Channel Capacity) được xác định chính xác nhờ Muroya (1953) và Fano (1961). Giải thuật và chương trình tính dung lượng kênh đã được viết bởi Eisenberg (1963).

Định lý cơ bản về truyền tin đã chỉ ra rằng "với dung lượng kênh cho trước luôn có thể tìm ra một phương pháp truyền tin với lượng tin nhỏ hơn dung lượng kênh và đạt sai số nhỏ hơn sai số cho phép bất kỳ". Định lý cơ bản về truyền tin đã được Feinstein (1954, 1958) khảo sát. Các nhà khoa học Blackwell, Breinan (1958, 1959) và Thomasian (1961) đã lần lượt chỉnh lý để đạt chuẩn tốt hơn. Trong các nội dung tiếp theo của bài học, các bạn sẽ tìm hiểu về mô hình kênh truyền tin rời rạc không nhớ ở khia cạnh vật lý và toán học. Đặc biệt ở mô hình toán học sẽ chỉ ra cách xác định phân phối ở đầu ra dựa vào phân phối ở đầu vào.

### Mô hình vật lý

Một thông báo được cấu tạo từ các ký hiệu của một bảng chữ cái ở đầu truyền (input) và được truyền trên kênh. Thông báo được nhận ở cuối kênh (hay đầu nhận-output) và được giải mã theo bảng chữ cái ở đầu truyền. Mặt khác, từng ký tự ở đầu nhận có thể quan hệ với các ký tự ở đầu nhận trước đó, các ký tự ở đầu truyền và trạng thái của kênh truyền. Để đơn giản, ở đây chúng ta chỉ xét mô hình vật lý như sau:

Xét từng ký tự ở đầu nhận chỉ phụ thuộc vào ký tự ở đầu truyền tương ứng với nó, nếu kênh truyền có nhiễu thì một ký tự ở đầu truyền có thể được diễn giải (nhiễu) ra nhiều ký tự khác nhau ở đầu nhận và do đó tạo ra một phân phối xác suất có điều kiện cho ký tự ở đầu nhận. *Mô hình truyền tin rời rạc không nhớ là mô hình truyền tin chỉ xét mối quan hệ giữa ký tự truyền và ký tự nhận được tương ứng, không xét mối quan hệ giữa ký tự nhận được và ký tự nhận được trước đó.*
Mô hình:




Các qui ước:

- X: là biến ngẫu nhiên có giá trị cần truyền ở đầu truyền.
- Y: là biến ngẫu nhiên chứa giá trị có thể nhận được ở đầu nhận.
- $\Gamma_X$: là bảng chữ cái sinh mã ở đầu truyền.
- $\Gamma_Y$: là bảng chữ cái giải mã ở đầu nhận.
- X, Y, $\Gamma_X$, $\Gamma_Y$: đều hữu hạn và rời rạc.
- Truyền rời rạc từng ký tự và nhận cũng rời rạc từng ký tự.
- Ký tự nhận sau không phụ thuộc vào ký tự nhận trước.

## Mô hình toán học

Ta gọi:

- $\Gamma_X=\{x_1, x_2, \ldots, x_M\}$ là bộ ký tự sinh mã ở đầu truyền (input).
- $\Gamma_Y=\{y_1, y_2,\ldots,y_L\}$ là bộ ký tự giải mã ở đầu nhận (output).
- Biến ngẫu nhiên X lấy giá trị (đã mã hóa) trên $\Gamma_X$ và có phân phối $p(X=x_i)=p(x_i)$ với i=1,..,M.
- Biến ngẫu nhiên Y lấy giá trị (giải mã) trên $\Gamma_Y$ và có phân phối xác suất có điều kiện: $P(Y=y_j/X=x_i)=p(y_j/x_i)=p_{ij}$ với j=1,..,L.

Gọi $A=||p_{ij}||$ là ma trận truyền tin hay mô hình truyền tin của kênh truyền rời rạc không nhớ.
Với $i=\overline{1,M}$ , $j=\overline{1,L}$ và $p_{ij} = p(Y=y_j/X=x_i) = p(y_j/x_i)$ là xác suất nhận được giá trị $y_j$ khi đã truyền giá trị $x_i$.

Tính phân phối đầu nhận:

Ta có: $p(Y=y_j) = p(y_j) = \sum_{i=1}^{M} p(x_i).p(y_j / x_i)$

$$\Rightarrow p(yj) = \sum_{i=1}^{M} p(x_i).p(y_j / x_i)$$

$$= \sum_{i=1}^{M} p(x_i).p_{ij}$$

Vậy $\mathbf{p(y_j)= P_X^{'} .A_j}$ **(1)**

Một các tổng quát: $\mathbf{P'_Y = P'_X.A}$ **(2)**
Trong đó:

- $A_j$ là cột thứ j của A
- $P'_X = [p(x_1), p(x_2),…., p(x_M)]$.
- $P'_Y = [p(y_1), p(y_2),…., p(y_M)]$.

## Ví dụ xác định phân phối đầu nhận

Cho ma trận truyền tin như sau:

$$A = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \\ \\ \end{array} \begin{array}{c} \begin{array}{ccc} y_1 & y_2 & y_3 \end{array} \\ \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix} \end{array}$$

Xác suất truyền: $p(x_1)=0.5$ và $p(x_2)=p(x_3)= 0.25$.

Ta tìm phân phối của Y :
Ta có: $P_X^{'} =(0.5, 0.25, 0.25)$
Áp dụng công thức (1) ở trên ta được:

$p(y_1) = P_{x,}.A_1 = 0.375$
$p(y_2) = P_{x,}.A_2 = 0.3$
$p(y_3) = P_x.A_3 = 0.325$

$\Rightarrow PY' =(0.375, 0.3, 0.325)$

## Lượng tin trên kênh truyền

**Ví dụ:** cho ma trận truyền tin như sau:

$$A = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \\ \\ \end{array} \begin{array}{c} \begin{array}{ccc} y_1 & y_2 & y_3 \end{array} \\ \begin{bmatrix} 0.5 & 0.2 & 0.3 \\ 0.3 & 0.5 & 0.2 \\ 0.2 & 0.3 & 0.5 \end{bmatrix} \end{array}$$

Xác suất truyền: $p(x_1)=0.5$ và $p(x_2)=p(x_3)= 0.25$.
$X = \{x_1, x_2, x_3\}$ được xem như tập các ký tự truyền và $Y =\{y_1, y_2, y_3\}$ là tập các ký tự nhận.

Tính lượng tin trên kênh truyền:
Ta tìm phân phối của Y :
Ta có: $P_X^{'} =(0.5, 0.25, 0.25)$
Áp dụng công thức (1) ở trên ta được:

$p(y_1) = P_{x,}.A_1 = 0.375$
$p(y_2) = P_{x,}.A_2 = 0.3$
$p(y_3) = P_x.A_3 = 0.325$

$\Rightarrow P_Y^{'} =(0.375, 0.3, 0.325)$

Tính các Entropy:

$H(Y) = H(0.375, 0.3, 0.325) = 1.58$ (bit)
$H(Y/X=x_1) = H(0.5, 0.2, 0.3)= 1.49$ (bit)
$H(Y/X=x_2) = H(0.3, 0.5, 0.2)= 1.49$ (bit)
$H(Y/X=x_1) = H(0.2, 0.3, 0.5)= 1.49$ (bit)
$H(Y/X)= p(x_1).H(Y/X=x_1) + p(x_2).H(Y/X=x_2) + p(x_3).H(Y/X=x_3) = 1.49$ (bit)

***Lượng thông tin truyền trên kênh:*** $I (X/Y)= I (Y/X)= H(Y) - H(Y/X) = 0,09$ (bit)

## Định nghĩa dung lượng kênh truyền

Dựa vào ma trận truyền tin A, ta có thể dễ dàng tính lượng tin trên kênh truyền.

I(X/Y)= H(X)-H(Y/X)

= H(Y)-H(X/Y)

= I(Y/X)

Ta có I(X/Y)= H(Y)-H(Y/X), trong đó:

H(Y)= $H(P_X^{'}.A)$ phụ thuộc vào $P_X$.

H(Y/X) phụ thuộc vào $P_X$

Vậy: I(Y/X) phụ thuộc hoàn toàn vào $P_X$ và do đó I(Y/X) có thể đạt Max với $P_X$ xác định nào đó.

Ta định nghĩa: $C = \underset{\forall p(X)}{Max}\, I(X/Y)$ là dung lượng của kênh truyền (ĐVT: bit).

# BAI 4.2: CÁC DẠNG KÊNH TRUYỀN

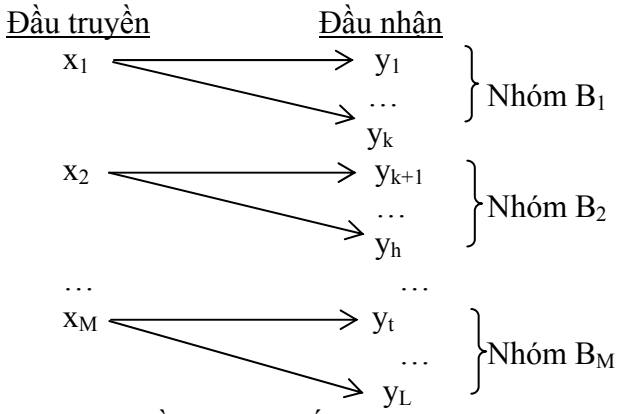
## Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:

- Biết kênh truyền không mất tin,
- Biết kênh truyền xác định,
- Biết kênh truyền không nhiễu,
- Biết kênh truyền không sử dụng được,
- Hiểu kênh truyền đối xứng,

## Hiểu định lý về dung lượng kênh truyền,Kênh truyền không mất tin

**Mô hình:** từ tập hợp các giá trị có thể nhận được ở đầu nhận $Y=\{y_1, y_2, \ldots, y_L\}$ được phân thành M nhóm $B_i$ tương ứng với các giá trị $x_i$ ở đầu truyền và xác suất để truyền $x_i$ với điều kiện đã nhận $y_j$ là $p(X= x_i /Y=y_j \in B_i)=1$ ( với $M < L$ ).

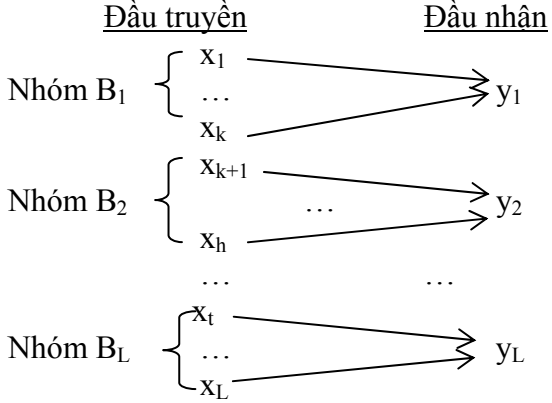



Đặc trưng của kênh truyền không mất tin là H(X/Y)=0. Có nghĩa là lượng tin chưa biết về X khi nhận Y là bằng 0 hay ta có thể hiểu khi nhận được Y thì ta hoàn toàn có thể biết về X.

**Dung lượng:** $C=\log_2 M$ (Sinh viên tự chứng minh, xem như bài tập)

## Kênh truyền xác định

**Mô hình:** từ tập hợp các giá trị có thể truyền ở đầu truyền được phân thành L nhóm $B_j$ tương ứng với các giá trị có thể nhận được $y_j$ ở đầu nhận và xác suất để nhận $y_j$ với điều kiện đã truyền $x_i$ là $p(Y=y_j/X=x_i \in B_j)=1$ (M>L).




Đặc trưng: của kênh truyền xác định là H(Y/X)=0. Có nghĩa là lượng tin chưa biết về Y khi truyền X bằng 0 hay khi truyền X thì ta biết sẽ nhận được Y.

**Dung lượng:** $C=\log_2 L$ (Sinh viên tự chứng minh, xem như bài tập)

## Kênh truyền không nhiễu

**Mô hình:** là sự kết hợp của kênh truyền xác định và kênh truyền không mất thông tin, truyền ký tự nào sẽ nhận được đúng ký tự đó.

| Đầu truyền | | Đầu nhận |
|---|---|---|
| $x_1$ | ⟶ | $x_1$ |
| $x_2$ | ⟶ | $x_2$ |
| … | | … |
| $x_M$ | ⟶ | $x_M$ |

Đặc trưng: H(X/Y)=H(Y/X)=0.

**Dung lượng:** $C=\log_2 L=\log_2 M$ *(Sinh viên tự chứng minh, xem như bài tập)*

**Ví dụ:** ma trận truyền tin của kênh truyền không nhiễu với M=L=3:

$$A=\begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \end{array}\begin{array}{c} \begin{array}{ccc} y_1 & y_2 & y_3 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{array}$$

## Kênh truyền không sử dụng được.

**Mô hình:** là kênh truyền mà khi truyền giá trị nào thì mất giá trị đó hoặc xác suất nhiễu thông tin trên kênh truyền lớn hơn xác suất nhận được.

Đặc trưng: H(X/Y)=H(Y/X)= max

**Dung lượng:** C=0 (Sinh viên tự chứng minh, xem như bài tập)

**Ví dụ:** kênh truyền có ma trận truyền tin như sau:

$$A=\begin{pmatrix} \varepsilon & 1-\varepsilon \\ \varepsilon & 1-\varepsilon \end{pmatrix}$$

## Kênh truyền đối xứng

**Mô hình:** là kênh truyền mà ma trận truyền tin có đặc điểm sau:

+ Mỗi dòng của ma trận A là một hoán vị của phân phối $P=\{p'_1, p'_2, \ldots, p'_L\}$
+ Mỗi cột của ma trận A là một hoán vị của $Q=\{q'_1, q'_2, \ldots, q'_M\}$

**Ví dụ:** cho kênh truyền đối xứng có ma trận truyền tin như sau:

$$A = \begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \end{array}\begin{array}{c} \begin{array}{ccc} y_1 & y_2 & y_3 \end{array} \\ \begin{bmatrix} 1/2 & 1/3 & 1/6 \\ 1/3 & 1/6 & 1/2 \\ 1/6 & 1/2 & 1/3 \end{bmatrix} \end{array}$$

## Xây dựng công thức tính dung lượng kênh truyền đối xứng

Do H(Y/X) không phụ thuộc vào phân phối của X => Max của I(X/Y) được quy về mã của H(Y). Hay

$$C = Max\, I(X/Y) = Max(H(Y) - H(Y/X))$$

Ta có thể tính dễ dàng:

$$H(Y/X) = -\sum_{j=i}^{L} p'_j \log p'_j = const$$

Do đó:

$$C = Max\, I(X/Y) = MaxH(Y) + \sum_{j=i}^{L} p'_j \log p'_j$$

Do H(Y)<= logL => ta cần chứng tỏ "=" xảy ra khi $p_1=p_2=...=p_L=1/L$

Xét trường hợp $P(X=x_i)=1/M$, với mọi i => chứng minh $P(Y=y_j)=1/L$ với mọi j

Thật vậy :

$$P(Y = y_j) = \sum_{i=1}^{M} P(Y = y_j, X = x_i)$$

$$= \sum_{i=1}^{M} P(X = x_i) P(Y = y_j / X = x_i) = \frac{1}{M} \sum_{i=1}^{M} P_{ij} = \frac{1}{M} q_i$$

Từ A ta nhận thấy:

$$A = \begin{pmatrix} p_{11} & \cdots & p_{1L} \\ \cdots & \cdots & \cdots \\ p_{M1} & \cdots & p_{ML} \end{pmatrix} => \sum_A = \text{tổng các phần tử của A.}$$

$$\text{Do } \sum_A = \sum_A^{+hang} = \sum_A^{+\cot} => M = L\sum_{i=i}^{M} q_i => \sum_{i=i}^{M} q_i = \frac{M}{L}$$

$$=> P(Y = y_j) = \frac{1}{M}\frac{M}{L} = \frac{1}{L} => H(Y) = -\sum p' P(Y = y_j) \log P(Y = y_j) = \log L = Max$$

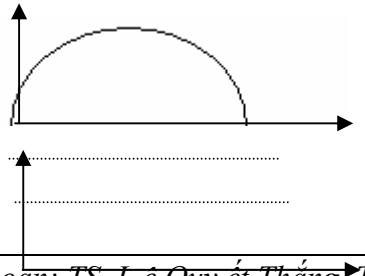
=> H(Y) đạt max là logL khi $P(Y=y_j)=1/L$ hoặc $P(X=x_i)=1/M$

Vậy: C= log L – H(p'1, p'2, …, p'L ) hay $C = \log L + \sum_{j=1}^{L} p_j \log p_j$

**Chú ý:** trường hợp kênh 1 bit với nhiễu β

Ma trận truyền tin $A = \begin{pmatrix} 1-\beta & \beta \\ \beta & 1-\beta \end{pmatrix}$

Dung lượng C=1+(1-β) log(1-β)+βlogβ = 1- H(β, 1-β)



1 – H(β,1-β)

## Định lý về dung lượng kênh truyền

Giả sử ma trạn A có dạng vuông và có ma trận nghịch đảo là $A^{-1}$
Ký hiệu $A=||p_{ij}||$ với i=1,2,...,M và j =1,2,...,M
$A^{-1}=||q_{ij}||$ với i=1,2,...,M và j =1,2,...,M
Đặt tham số $d_k=\sum_{j=1}^{M} q_{jk} \exp_2\left[-\sum_{i=1}^{M} q_{ji} H(Y/X=x_i)\right], \forall k=\overline{1,M}$
Nếu $d_k>0$ thì dung lượng kênh truyền có dạng:

$$C = Log\left\{\sum_{j=1}^{M} \exp_2\left[-\sum_{i=1}^{M} q_{ji} H(Y/X=x_i)\right]\right\}$$

Giá trị cực đại đạt khi tín hiệu vào X=X* thỏa phân phối $P(X^*=x_k)=2^{-C}d_k$
Hay C=max I(X/Y)=I(X*/Y)
Chú ý:

- Điều kiện $d_k>0$ cho phép hàm I(X/Y) là hàm lồi => Tồn tại Max tuyệt đối tại phân phối của X* với $p(X^*=x_k)=2^{-C} d_k =p_k$ (với mọi k).
- Nếu điều kiện ma trận vuông hoặc ma trận ngịch đảo không thỏa thì giá trị cực đại max sẽ nằm trên đường biên của miền xác định $\{p_k>0$ và $-\Sigma p_k=1\}$

## Bài tập

1. Cho một kênh truyền có ma trận truyền tin như sau:

$$\begin{array}{c} \\ x_1 \\ x_2 \\ x_3 \end{array}\begin{bmatrix} 1/2 & 1/3 & 1/6 \\ 1/3 & 1/6 & 1/2 \\ 1/6 & 1/2 & 1/3 \end{bmatrix}$$
$$\begin{array}{ccc} y_1 & y_2 & y_3 \end{array}$$

   Tính dung lượng kênh truyền.
2. Chứng minh các công thức tính dung lượng kênh truyền trên.

# BÀI 4.3: LƯỢC ĐỒ GIẢI MÃ

## Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:
- Biết đặt vấn đề bài toán giải mã,
- Hiểu các khái niệm cơ bản của kỹ thuật truyền tin,
- Biết và hiểu các dạng sai số cơ bản của kỹ thuật truyền tin,
- Hiểu phương pháp xây dựng lược đồ giải mã tối ưu,
- Vận dụng xây dựng lược đồ giải mã tối ưu và tính các dạng xác suất truyền sai.

## Đặt vấn đề bài toán giải mã

Phân tích yêu cầu giải mã:

Khi truyền giá trị $x_i$, ta sẽ nhận được $y_j$.

Đối với kênh truyền không nhiễu thì $y_j$ chính là $x_i$. Đối với kênh truyền có nhiễu thì $y_j$ có thể khác $x_i$. Do đó ta cần tìm cách giải mã $y_j$ về giá trị $x_i$ khi kênh truyền có nhiễu.

Phép phân hoạch các giá trị ở đầu nhận:

Phép phân hoạch tập các giá trị ở đầu nhập $y_j \in Y$ là phép phân chia tập Y thành các tập con $B_i$ sao cho:

1. $\begin{cases} B_i \cap B_j = \varnothing \\ \bigcup_{i=1}^{M} B_i = Y \end{cases} \quad (\forall\ i \neq j)$

2. Khi nhận $y_j \in B_i$ thì giải mã về $x_i$.

## Ví dụ bài toán giải mã

Cho tập các từ mã truyền X và tập các dãy n bit nhận được Y như sau:

X={0000, 0101, 1110, 1011}

Y={0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111,
1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111}

Giả sử ta có thể phân hoạch tập Y thành các tập con $B_i$ như sau:

$B_1$={0000, 1000, 0001, 0010}

$B_2$={0101, 1101, 0100, 0111}

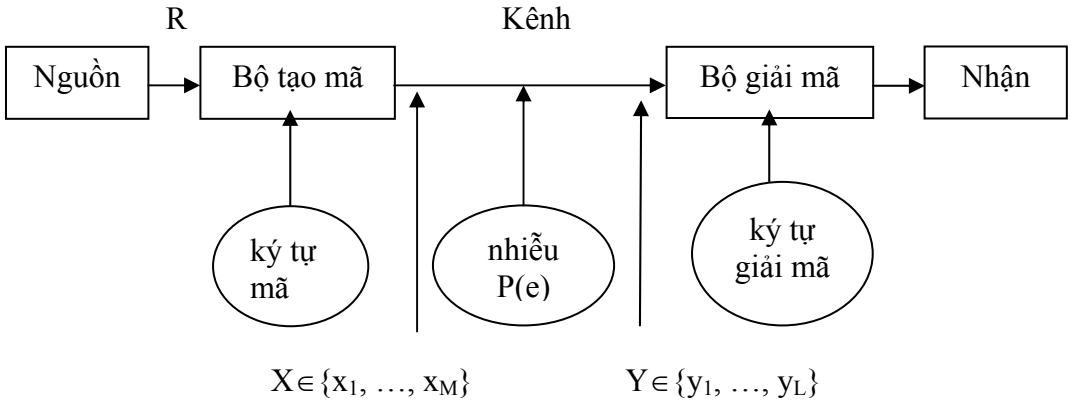
$B_3$={1110, 0110, 1111, 1100}

$B_4$={1011, 0011, 1010, 1001}

Giả sử nhận $y_j$ = 0011 thì giải mã về $x_4$ = 1011 vì $y_j \in B_4$.

## Các khái niệm cơ bản của kỹ thuật truyền tin

Xét sơ đồ truyền tin như sau:




Diễn giải:

- Nguồn phát tín hiệu (hay thông báo) với vận tốc R (tín hiệu/giây).
- Tín hiệu được mã hóa từ bộ ký tự mã.
- Tín hiệu mã hóa được truyền trên kênh với vận tốc C (ký tự/giây), C đồng thời là dung lượng của kênh truyền.
- Tín hiệu truyền trên kênh có thể bị nhiễu với xác suất P(e).
- Trước khi nhận, tín hiệu mã hóa được giải mã theo một phương thức tối ưu và độ chính xác cao nhất có thể có.

Bài toán đặt ra ở đây: tìm giải pháp tạo mã sao cho sai số đầu nhận có xác suất nhỏ hơn ε bất kỳ ($\varepsilon < P(e)$) đồng thời với đồng bộ hóa: vận tốc phát thông báo ở nguồn R và vận tốc truyền tải ≤ C (C là dung lượng kênh).

Các khái niệm cơ bản:

**Từ mã:** là dãy n ký tự truyền hay dãy n ký tự nhận đúng.

**Bộ mã (S,n):** là tập hợp gồm S từ mã với độ dài mỗi từ mã đều bằng n và được ký hiệu là $x^{(1)}$, …, $x^{(s)}$.

**Lược đồ giải mã**: là một hàm gán cho một dãy n ký tự nhận được $y_j$ một từ mã của bộ mã $W = \{w_1, w_2, \ldots, w_s\}$. Ký hiệu: $g(y_j) = w_i$

**Lược đồ giải mã tối ưu**: là lược đồ giải mã sao cho tổng xác suất truyền sai là nhỏ nhất hay tổng xác suất truyền đúng là lớn nhất.

Nghĩa là: khi nhận $y_j$ thì ta giải mã về $w_i^*$ sao cho:

$$P(w_i^*/y_j) = \text{Max}\{P(w_k/y_j)\}$$
$$\forall w_k \in W$$

## Ví dụ minh họa các khái niệm cơ bản

Giả sử kênh truyền từng bit với C=1, nguồn phát thông báo với tốc độ R=2/5 bit/giây (R<C). Để thuận lợi cho mã hóa và giảm nhiễu, ta xét từng khoảng thời gian n = 5 giây. Như vậy trong khoảng thời gian n = 5 giây, ta có:

- Tập hợp các tín hiệu khác nhau là $2^{nR} = 4$. Giả sử 4 tín hiệu là $m_1$, $m_2$, $m_3$, $m_4$.
- Số bit được phát ra là nR=2 bit và một tín hiệu dạng $m_i$ được kết cấu bởi một dãy các bit.

- Quá trình mã hóa các tín hiệu $m_1$, $m_2$, $m_3$, $m_4$ cần chú ý là: mỗi $m_i$ cần được mã hóa với số bit tối đa là nC=5 bit. Vậy, ta có thể mã hóa các tín hiệu $m_i$ theo 2 cách sau:

Cách 1:

$m_1$=00000
$m_2$=01101
$m_3$=11010
$m_4$=10111

Cách 2:

$m_1$=00
$m_2$=01
$m_3$=10
$m_4$=11

Nếu sử dụng cách 1 với độ dài 5 bit, trong đó 5 bit có thể hiểu là có 2 bit thông tin cần truyền và 3 bit con lại là 3 bit được bổ sung để phát hiện nhiễu theo một phương pháp nào đó sẽ được đề cập ở các nội dung tiếp theo sau. Với cách mã hóa này, ta có nhiều khả năng phát hiện và sửa sai do nhiễu.

Nếu sử dụng cách 2 thì trường hợp có 1 bit truyền sai sẽ dẫn đến trùng lặp sang một trong các tín hiệu khác. Ví dụ truyền $m_1$=00 và nhận 2 bit là 01 (do nhiễu), trong trường hợp này 01 chính là $m_2$, đây là một tín hiệu đúng nên ta không thể phát hiện có nhiễu hay không nhiễu.

Như vậy, trong khoảng thời gian truyền và dung lượng kênh cho phép, ta cần mã hóa mỗi tín hiệu càng dài càng tốt nhưng không được vượt quá độ dài mã cho phép. Trường hợp với thời gian n=5 và c= 1 bit thì nC=5 là số bit tối đa có thể truyền nên ta chỉ mã hóa tín hiệu với độ dài mã tối đa là 5 bit.

## Các dạng sai số cơ bản

Xác suất truyền sai từ mã $x_i$: $p(e/x_i)= \sum p(Y=y_j \notin B_i/X=x_i)$

Xác suất truyền sai trung bình: $p(e) = \sum_{i=1}^{M} p(X = x_i)\, p(e / x_i)$

Xác suất truyền sai lớn nhất: $p_m(e) = \underset{i=\overline{1,M}}{Max}\, p(e / x_i)$

## Phương pháp xây dựng lượt đồ giải mã tối ưu

Theo công thức Bayes:
Ta có: $P(w_k/y_j) = [p(w_k).p(y_j/w_k)] / p(y_j)$ với $(\forall w_k \in W)$
Từ định nghĩa lược đồ giải mã tối ưu:
$\Rightarrow$ tìm $w_k$ sao cho $P(w_k/y_j) \rightarrow Max \Leftrightarrow p(w_k).p(y_j/w_k) \rightarrow Max$.

Như vậy, ta có thể xây dựng lược đồ giải mã tối ưu theo các bước sau:

**Bước 0**: Khởi tạo các $B_i = \phi$ $(\forall i)$

**Bước lặp:** xét với mọi $y_j \in Y$

+ Tính:

$p(w_1).p(y_j/w_1)$
$p(w_2).p(y_j/w_2)$
…
$p(w_M).p(y_j/w_M)$

+ So sánh các giá trị tính trên và chọn giá trị $w^*_i$ sao cho $p(w^*_i).p(y_j/w^*_i)$= Max $\{p(w_k).p(y_j/w_k)\}$ $(\forall w_k \in W)$
+ $B_i = B_i + \{y_j\}$ và $g(y_j) = w^*_i$.

## Minh họa xây dựng lược đồ giải mã tối ưu

## Bài toán:

Cho ma trận truyền tin A và xác suất ở đầu truyền như sau:

$$\begin{array}{c} x_1 \\ x_2 \\ x_3 \end{array}\begin{bmatrix} 1/2 & 1/3 & 1/6 \\ 1/3 & 1/6 & 1/2 \\ 1/6 & 1/2 & 1/3 \end{bmatrix}$$

$$y_1 \quad y_2 \quad y_3$$

Với $p(x_1)=1/2$; $p(x_2)=p(x_3)=1/4$. Hãy xây dựng lược đồ giải mã tối ưu.

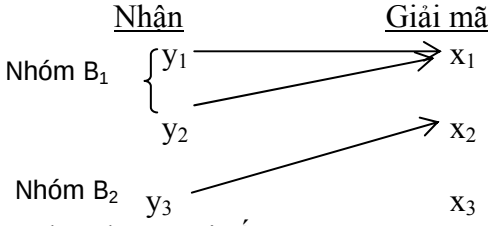
Áp dụng phương pháp xây dựng lược đồ giải mã tối ưu:
**Bước 0:** $B_1=\{\}$; $B_2=\{\}$; $B_3=\{\}$;
**Bước 1:** Nhận giá trị $y_1$, ta tính:
+ $p(x_1).p(y_1/x_1)= 1/2.1/2 = 1/4$ (Max)
+ $p(x_2).p(y_1/x_2)= 1/4.1/3 = 1/12$
+ $p(x_3).p(y_1/x_3)= 1/4.1/6 = 1/24$

Do $p(x_1).p(y_1/x_1)$ lớn nhất nên liệt kê $y_1$ vào tập hợp $B_1$ tương ứng với $x_1$.
=> $B_1=\{y_1\}$.
**Bước 2:** Nhận giá trị $y_2$, ta tính:
+ $p(x_1).p(y_2/x_1)= 1/2 \,.\, 1/3 = 1/6$ (Max)
+ $p(x_2).p(y_2/x_2)= 1/4 \,.\, 1/6 = 1/24$
+ $p(x_3).p(y_2/x_3)= 1/4 \,.\, 1/2 = 1/8$

Do $p(x_1).p(y_1/x_1)$ lớn nhất nên liệt kê $y_2$ vào tập hợp $B_1$ tương ứng với $x_1$.
=> $B_1=\{y_1, y_2\}$.
**Bước 3:** Nhận giá trị $y_3$, ta tính:
+ $p(x_1).p(y_3/x_1)= 1/2 \,.\, 1/6 = 1/12$
+ $p(x_2).p(y_3/x_2)= 1/4 \,.\, 1/2 = 1/8$ (Max)
+ $p(x_3).p(y_3/x_3)= 1/4 \,.\, 1/3 = 1/12$

Do $p(x_1).p(y_2/x_1)$ lớn nhất nên liệt kê $y_3$ vào tập hợp $B_2$ tương ứng với $x_2$.
=> $B_2=\{y_3\}$.

Kết quả:
Phân hoạch: $B_1=\{y_1, y_2\}$, $B_2=\{y_3\}$ và $B_3=\{\}$.
Lược đồ giải mã tối ưu:

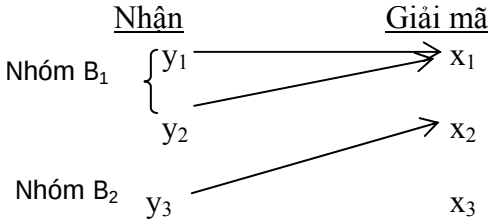



Minh họa cách tính các sai số
Xét lại ví dụ minh họa xây dựng lược đồ giải mã tối ưu trên, ta có:
- Ma trận truyền tin A:

$$\begin{array}{c} x_1 \\ x_2 \\ x_3 \end{array}\begin{bmatrix} 1/2 & 1/3 & 1/6 \\ 1/3 & 1/6 & 1/2 \\ 1/6 & 1/2 & 1/3 \end{bmatrix}$$
$$\begin{array}{ccc} y_1 & y_2 & y_3 \end{array}$$

- Xác suất ở đầu truyền: $p(x_1)=1/2$; $p(x_2)=p(x_3)=1/4$.

- Lược đồ giải mã tối ưu:




- Phân hoạch: $B_1=\{y_1, y_2\}$, $B_2=\{y_3\}$ và $B_3=\{\}$.

Tính các xác suất truyền sai:
*Xác suất truyền sai một từ mã:*
Xác suất truyền sai từ mã $x_1$: $p(e/x_1)= \sum p(Y=y_j \notin B_1/X=x_1)$
$= p(y_3/x_1) =1/6$
Xác suất truyền sai từ mã $x_2$: $p(e/x_2)= \sum p(Y=y_j \notin B_2/X=x_2)$
$= p(y_1/x_2) + p(y_2/x_2) =1/3+1/6=1/2$
Xác suất truyền sai từ mã $x_3$: $p(e/x_3)= \sum p(Y=y_j \notin B_3/X=x_3)$
$= p(y_1/x_3) + p(y_2/x_3) + p(y_3/x_3) =1/6+1/3+1/2=1$
Xác suất truyền sai trung bình: $p(e) = \sum_{i=1}^{M} p(X = x_i)\, p(e/x_i)$
$\Rightarrow p(e)=p(x_1).p(e/x_1) + p(x_2).p(e/x_2) + p(x_3).p(e/x_3) = 1/2.1/6 + 1/4.1/2 + 1/4.1 = 11/24$
*Xác suất truyền sai lớn nhất:* $p_m(e) = \underset{i=\overline{1,M}}{Max}\, p(e/x_i)$
$\Rightarrow pm(e) = Max\{ p(e/x_1), p(e/x_2), p(e/x_3)\} = p(e/x_3) =1$

## Bài tập 1

1. Cho ma trận truyền tin sau:

$$\begin{array}{c} x_1 \\ x_2 \\ x_3 \end{array}\begin{bmatrix} 1/2 & 1/3 & 1/6 \\ 1/3 & 1/6 & 1/2 \\ 1/6 & 1/2 & 1/3 \end{bmatrix}$$

$y_1 \quad y_2 \quad y_3$

Biết xác suất ở đầu truyền: $p(x_1)=5/10$, $p(x_2)=3/10$, $p(x_3)=2/10$.

- Tính dung lượng kênh truyền.
- Xây dựng lược đồ giải mã tối ưu.
- Tính các sai số p(e) và $p_m(e)$.

2. Cho ma trận truyền tin sau:

$$\begin{array}{c} x_1 \\ x_2 \\ x_3 \end{array}\begin{bmatrix} 7/12 & 3/12 & 2/12 \\ 2/12 & 7/12 & 3/12 \\ 3/12 & 2/12 & 7/12 \end{bmatrix}$$

$y_1 \quad y_2 \quad y_3$

Biết xác suất ở đầu truyền: $p(x_1)=1/3$, $p(x_2)=1/3$, $p(x_3)=1/3$.

- Tính dung lượng kênh truyền.
- Xây dựng lược đồ giải mã tối ưu
- Tìm các sai số p(e) và $p_m(e)$.

## Bài Tập 2

1. Cho ma trận truyền tin sau:

$$\begin{array}{c} x_1 \\ x_2 \\ x_3 \end{array}\begin{pmatrix} 1/2 & 1/3 & 1/6 \\ 1/6 & 1/2 & 1/3 \\ 1/3 & 1/6 & 1/2 \end{pmatrix}$$

$y_1 \quad y_2 \quad y_3$

Biết $p(x_1)=1/2$, $p(x_2)=1/4$, $p(x_3)=1/4$.

- Tính dung lượng kênh truyền.
- Xây dựng lược đồ giải mã tối ưu.
- Tính các sai số p(e) và $p_m(e)$.

2. Cho ma trận truyền tin sau:

$$\begin{array}{c} x_1 \\ x_2 \\ x_3 \end{array}\begin{pmatrix} 7/10 & 2/10 & 1/10 \\ 1/10 & 7/10 & 2/10 \\ 2/10 & 1/10 & 7/10 \end{pmatrix}$$

$y_1 \quad y_2 \quad y_3$

Biết xác suất truyền $p(x_1)=0.4$, $p(x_2)=0.4$, $p(x_3)=0.2$.

- Tính dung lượng kênh truyền.
- Xây dựng lược đồ giải mã tối ưu.
- Tính các sai số p(e) và $p_m(e)$.

# CHƯƠNG 5: SỬA LỖI

**Mục tiêu:** Xây dựng nguyên tắc sửa lỗi dựa vào khoảng cách Hamming. Trên nguyên tắc này, phương pháp sửa lỗi "kiểm tra chắn lẻ (parity check)" được xây dựng và tạo ra quy trình sửa lỗi tối ưu và phù hợp với công nghệ truyền tin hiện nay.

## BÀI 5.1: NGUYÊN LÝ KHOẢNG CÁCH NHỎ NHẤT HAMMING

### Mục tiêu:

Sau khi hoàn tất bài học này bạn có thể hiểu:
- Định nghĩa khoảng cách Hamming
- Kênh truyền đối xứng nhị phân và lược đồ giải mã tối ưu
- Quan hệ giữa xác suất giải mã và khoảng cách Hamming
- Nguyên lý khoảng cách nhỏ nhất của Hamming.

### Khoảng cách Hamming

**Định nghĩa:** cho $v_1$ và $v_2$ là 2 dãy nhị phân dài n bit, ta gọi khoảng cách Hamming giữa 2 dãy $v_1$, $v_2$ là số bit tương ứng khác nhau. Ký hiệu: $d(v_1, v_2)$.

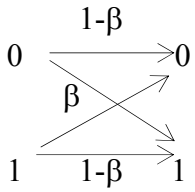
Ví dụ:

$v_1$=10101010
$v_2$=10101111

Ta nhận thấy rằng bit thứ 6 và bit thứ 8 giữa giữa $v_1$ và $v_2$ là khác nhau nên số bit tương ứng khác nhau giữa $v_1$ và $v_2$ là 2. Do đó, ta nói khoảng cách Hamming giữa $v_1$ và $v_2$ là 2 hay $d(v_1, v_2) = 2$

### Kênh truyền đối xứng nhị phân và lược đồ giải mã tối ưu

Xét kênh truyền đối xứng nhị phân. Giả sử ta truyền các dãy từ mã nhị phân có độ dài n bits với xác suất truyền sai 1 bit là β.




Gọi $W = \{w_1, w_2,\ldots,w_s\}$ là tập s từ mã truyền, độ dài mỗi từ mã đều bằng n bit.

$V = \{v_1, v_2,\ldots., v_2^n\}$ là tập các dãy n bit nhận được ở cuối kênh với W có phân phối đều, xác suất để nhận $v_j$ khi truyền $w_i$ là $p(v_j/w_i) = p_{ij.}$

Theo lược đồ giải mã tối ưu ta có: khi nhận $v_j$ thì giải mã về $w_i^*$ sao cho:

$$P(w_i^*/v_j) = \text{Max}\{P(w_k/v_j)\}$$
$$(\forall w_i \in W)$$

Ta có: $P(w_k/y_j) = [p(w_k).p(y_j/w_k)] / p(y_j)$ với $(\forall w_k \in W)$

$\Rightarrow P(w_k/y_j) \rightarrow \text{Max} \Leftrightarrow p(w_k).p(y_j/w_k) \rightarrow \text{Max}.$

Do W có phân phối đều nên $P(w_k/y_j) \to Max \Leftrightarrow p(y_j/w_k) \to Max$

Vậy: để tìm $w_i^*$ sao cho $P(w_i^*/v_j) = Max\{P(w_k/v_j)\}$ ta chỉ cần tìm $w_i^*$ sao cho

$P(v_j/ w_i^*) = Max\{P(v_j/ w_k)\}$ (chỉ dựa vào ma trân truyền tin A)

## Ví dụ kênh truyền đối xứng nhị phân

Xét ma trận truyền tin A và xác suất ở đầu truyền như sau:

$$A = \begin{matrix} w_1 \\ w_2 \\ w_3 \end{matrix}\begin{bmatrix} 1/2 & 1/3 & 1/6 \\ 1/3 & 1/6 & 1/2 \\ 1/6 & 1/2 & 1/3 \end{bmatrix} \quad \text{và } p(w_1) = p(w_2) = p(w_3) = 1/3.$$

$v_1$ $v_2$ $v_3$

dựa vào lược đồ giải mã tối ưu ta có:

- Nhận $v_1$ giải mã về $w_1$
- Nhận $v_2$ giải mã về $w_3$
- Nhận v3 giải mã về w2.

## Quan hệ giữa xác suất giải mã và khoảng cách Hamming

Giả sử nhận được v:

Xét 2 từ mã $w_1$ và $w_2$ cần chọn để giải mã cho v.

+ Gọi $d_1=d(v, w_1)$, $d_2=d(v,w_2)$.

+ Ta có: $p(v/w_1)= \beta^{d_1}(1-\beta)^{n-d_1}$ (xác suất đế nhận v khi truyền $w_1$).

$P(v/w_2)= \beta^{d_2}(1-\beta)^{n-d_2}$ (xác suất đế nhận v khi truyền $w_2$).

So sánh xác suất: $\dfrac{p(v/w_1)}{p(v/w_2)} = \dfrac{\beta^{d_1}(1-\beta)^{n-d_1}}{\beta^{d_2}(1-\beta)^{n-d_2}} = \left(\dfrac{1-\beta}{\beta}\right)^{d_2-d_1}$

Nếu nhiễu $0 < \beta < ½$ thì $\dfrac{1-\beta}{\beta} > 1$

Do đó: $P(v/w_1) > p(v/w_2) \Leftrightarrow d_1 < d_2$

**Nhận xét:** xác suất giải mã càng lớn thì **khoảng cách Hamming** càng nhỏ.

## Nguyên lý Hamming

**Định lý:** trên kênh truyền đối xứng nhị phân với s từ mã ở đầu truyền có độ dài n bit, lược đồ giải mã tối ưu có thể thay thế bằng lược đồ giải mã theo khoảng cách Hamming với nguyên lý: nếu nhận được v, ta sẽ giải ra $w^*_i$

sao cho $d(v,w^*_i)=Min\, d(v,w_k)$ (với $\forall w_k \in W$).

## Ví dụ: xét bộ mã $W=\{w_1=00000, w_2=10011, w_3=11100, w_4=01111\}$

Giả sử nhận được dãy v=01011.

ta có: $d(v,w_1)=3$; $d(v,w_2)=2$; $d(v,w_3)=4$; $d(v,w_4)=1$.

vậy v được giải về $w_4$ vì khoảng cách Hamming giữa v và $w_4$ là nhỏ nhất.

## Bài tập

1. Cho bộ mã W={$w_1$=000000, $w_2$=101010, $w_3$=111000, $w_4$=111111} và nhận được dãy v=010111, khi đó giải mã về từ mã nào? diễn giải?
2. Cho bộ mã W={w1=000000, w2=010101, w3=000111, w4=111111} và Nhận được dãy v=010111, khi đó giải mã về từ mã nào? diễn giải?

# BÀI 5.2: BỔ ĐỀ VỀ TỰ SỬA LỖI VÀ CẬN HAMMING

## Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:

- Biết được Bổ đề về tự sửa lỗi,
- Hiểu Định lý về cận Hamming,
- Biết phân loại được các dạng lỗi,
- Làm cơ sở lý thuyết cho các phương pháp sửa lỗi được trình bài trong các bài học tiếp theo.

## Bổ đề về tự sửa lỗi

**Đặt vấn đề:** một từ mã w dài n bit khi được truyền tuần tự từng bit có thể sai e bit. Vấn đề đặt ra là khoáng cách (Hamming) giữa các từ mã và sai số e quan hệ với nhau như thế nào để có thể phân biệt tốt nhất đồng thời tất cả các từ mã? Bổ đề sau xác định quan hệ này.

Bổ đề:
Xét bộ mã $W=\{w_1, w_2, \ldots, w_s\}$ gồm có s từ mã nhị phân dài n bit và 1 số nguyên dương e.

1. Nếu $d(w_i, w_j) \geq 2e+1$ (với $\forall\ i \neq j$ )
   Khi đó: tất cả các dãy nhận được v có số bit lỗi ≤ e thì v có thể tự điều chỉnh (hay tự sửa lỗi).
2. Nếu $d(w_i, w_j) \geq 2e$ (với $\forall\ i \neq j$ )
   Khi đó: tất cả các dãy nhận được v có số bit lỗi < e thì v có thể tự điều chỉnh. Tất cả các dãy nhận được có số bit lỗi = e thì ta chỉ phát hiện là v có lỗi và không thể tự điều chỉnh được.
3. Ngược lại;
   Nếu v có số chữ số bit lỗi ≤ e và có thể tự điều chỉnh thì $d(w_i, w_j) \geq 2e+1$ (với $\forall\ i \neq j$ ).
   Nếu v có số chữ số bit lỗi ≤ e-1 tự điều chỉnh được và tất cả các tín hiệu với số chữ số bit lỗi ≤ e được phát hiện thì khoảng cách giữa các từ mã luôn thỏa: $d(w_i,w_j) \geq 2e$ (với $\forall\ i \neq j$ ).

## Chứng minh và minh họa bổ đề

**a.** Giả sử: $d(w, w') \geq 2e+1$ với $\forall\ i \neq j$ . Nếu w và w' có cùng khoảng cách đối với dãy v thì $d(v,w)=d(v,w') \geq e+1$. Vậy , nếu $d(v, w^*) \leq e$ thì v có thể được giải mã ra w*.

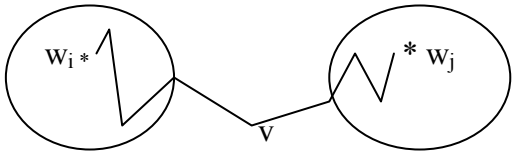
**b.** Nếu $d(w_i,w_j) \geq 2e$ với $\forall\ i \neq j$, có khả năng có v, w và w' với số chữ số lỗi là: $d(v,w)=d(v,w')=e$ ($d(v,w)+ d(v,w') \geq d(w,w') \geq 2e$). Có thể phát hiện ra các từ mã gần v, nhưng do tồn tại cùng lúc nhiều từ mã gần nhất với v dẫn đến không giải mã được, ngược lại hoàn toàn tương tự.

Minh họa:

**a.** $d(w_i, w_j)= 2e+1= 7,\ e=3$

Nếu $v \in B_i$ thì v được giải mã về $w_i$

Nếu $v \in B_j$ thì v được giải mã về $w_j$




b. $d(w_i, w_j) = 2e = 8$ $(e = 4, e - 1=3)$

nếu $v \notin B_i$ , $v \notin B_j$ => các điểm cách tâm khoảng cách 3 thì luôn được giải mã, còn các điểm cách tâm 4 thì chỉ phát hiện lỗi chứ không thể giải mã được.

**c.** Mã 3 chiều (x, y, z) bắt đầu từ gốc 000. Cứ một tín hiệu thay đổi thì mã bị đẩy đi theo 1 cạnh, chẳng hạn:

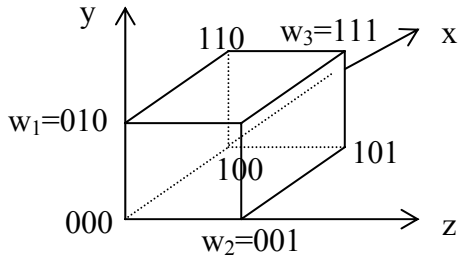
000 cách 010, 001 bởi 1 cạnh,

011 cách 010, 111 và 001 bởi 1 cạnh.

Như vậy, nếu ta chọn $w_1$=010, $w_2$=001, $w_3$=111 thì khoảng cách giữa chúng là 2

$d(w_1, w_2)=d(w_1, w_3)=d(w_2, w_3)=2$

vậy nếu có lỗi phát sinh thì chỉ phát hiện chứ không sửa được.




## Cận Hamming.

**Đặt vấn đề:** trong tổng số $2^n$ dãy nhị nhân dài n bit có thể chọn ra bao nhiêu dãy để tạo thành một bộ mã có thể tự điều chỉnh được e bit lỗi. Định lý cận Hamming cho chúng ta xác định số từ mã có độ dài n bit với giả thiết: có khả năng tự sửa được e bit lỗi (điều kiện cần tự sửa lỗi).

**Định lý:** Nếu bộ mã W có s từ mã có độ dài n bit có thể tự sửa được e bit lỗi thì

$$s \le \frac{2^n}{\sum_{i=1}^{e} C_n^i}$$

Ghi chú: $C_n^i = n!/(i!*(n-i)!)$

**Chứng minh:**

Xét từ mã nhị phân $w_i$ có độ dài n bit và có khả năng tự sửa được e bit lỗi.

Số dãy $v_j$ sai khác với $w_i$ từ 0 đến e bit là : $C_n^0 + C_n^1 + C_n^2 + ... + C_n^e = \sum_{i=0}^{e} C_n^i$

Tương ứng với s từ mã, tổng số dãy $v_j$ có thể tự sửa lỗi là : $s.\sum_{i=0}^{e} C_n^i \le 2^n$

($2^n$ là tổng số dãy nhị phân dài n bits).

$$\Rightarrow \quad s \le \frac{2^n}{\sum_{i=1}^{e} C_n^i}$$

### Phân các dạng lỗi

Giả sử ta truyền từ mã n bit $w_i \in W$ ( $1 \le i \le s$) và nhận được dãy n bit $v_j$ ( $1 \le j \le 2^n$).
Các loại lỗi có thể phát hiện sau:

Lỗi có thể tự điều chỉnh:
Trong trường hợp này tồn tại duy nhất từ mã $w^*_i$ sao cho $d(v_j, w^*_i) = \text{Min } d(v_j, w_k)$ với $\forall w_k \in W$.
=> $v_j$ được giải mã về $w^*_i$

Lỗi chỉ phát hiện không điều chỉnh được:
Trong trường hợp này tồn tại từ mã $w^*_i$ và $w^{**}_i$ sao cho
$d(v_j, w^*_i) = d(v_j, w^{**}_i) = \text{Min } d(v_j, w_k)$ với $\forall w_k \in W$
=> $v_j$ không thể giải mã chính xác.

Lỗi không phát hiện được.
Trong trường hợp ta giải mã ra $w^*_i$ nhưng khác với $w_i$ đã truyền.

### Bài tập

1. Cho n=7 và e=2, hãy áp dụng định lý cận Hamming cho biết số từ mã tối đa của bộ mã W.
2. Cho n=7 và e=2, hãy áp dụng định lý cận Hamming cho biết số từ mã tối đa của bộ mã W.
3. Hãy cho một ví dụ cụ thể minh họa các trường hợp phân loại lỗi.

# BÀI 5.3: MÃ KIỂM TRA CHẴN LẺ

### Mục tiêu:

Sau khi hoàn tất bài học này bạn có thể:
- Hiểu bộ mã kiểm tra chẵn lẻ,
- Hiểu phương pháp kiểm tra chẵn lẻ,

- Biết tính chất cơ bản của phương pháp kiểm tra chẵn lẻ,
- Hiểu và vận dụng tốt phương pháp sinh mã kiểm tra chẵn lẻ,
- Hiểu và vận dụng tốt Định lý quan hệ giữa độ dài mã n, số bit kiểm tra m và số lỗi tự sửa e,
- Vận dụng cho các bài học tiếp theo.

## Bộ mã kiểm tra chẵn lẻ

Bộ mã kiểm tra chẵn lẻ là bộ mã gồm s từ mã, trong đó mỗi từ mã có dạng sau:

$w' = \underbrace{r_1 r_2 r_3 \ldots r_m}_{\text{m bit kiểm tra}} \underbrace{r_{m+1} r_{m+2} \ldots r_{m+k}}_{\text{k bit thông tin}}$ (với n = m+k).

**Ghi chú:** trong một số trường hợp sinh mã theo phương pháp kiểm tra chẵn lẻ, thứ tự các bit kiểm tra và các bit thông tin có thể xen kẻ nhau (theo một thứ tự nào đó, chẳng hạn như mã Hamming,…) hay cũng có thể theo một thứ tự khác (theo quy ước khác). Ở đây, ta chọn thứ tự các bit kiểm tra chẵn lẻ và các bit thông tin như trên để dễ tính toán nhưng vẫn mất tính tổng quát hóa.

Trong đó: w' viết theo dong là chuyển vị của w (w được viết theo cột)

+ $r_i$: là bit thứ i của từ mã ( $1 \le i \le n$).
+ n: độ dài của từ mã hay số bit của từ mã chẵn lẻ.
+ m: số bit kiểm tra.
+ k = n-m: số bit thông tin $\Rightarrow s=2^k$ (vì với k bit thông tin thì ta chỉ có thể biểu diên tối đa $2^k$ trạng thái thông tin k bit).
+ Đoạn kiểm tra: gồm m bit dùng để kiểm tra mã sai.
+ Đoạn thông tin: gồm k bit thông tin.

Mỗi đoạn mã thông tin có duy nhất một đoạn mã kiểm tra và được xác định bởi hệ phương trình tuyến tính nhị phân sau:

$$\begin{cases} a_{11}r_1 + a_{12}r_2 + \ldots + a_{1n}r_n = 0 \\ a_{21}r_1 + a_{22}r_2 + \ldots + a_{2n}r_n = 0 \\ \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \\ a_{n1}r_1 + a_{n2}r_2 + \ldots + a_{nn}r_n = 0 \end{cases}$$

Gọi $A=||a_{ij}|| = A_{m \times n}$ , $a_{ij} \in \{0,1\}$, $i=\overline{1,m}$ , $j=\overline{1,n}$ . Ma trận A được gọi là ma trận kiểm tra chẵn lẻ có hạng là m (hay Rank(A) = m).

Các phép toán trong Modulo 2 (+,-):

$0 + 1 = 1 + 0 = 1$; $0 - 1 = 1 - 0 = 1$;
$1 + 1 = 1 - 1 = 0$;

## Phương pháp kiểm tra chẵn lẻ

Gọi $w'=r_1r_2\ldots r_n$ là từ mã truyền (hay dãy n bit truyền) và $v'=r_1r_2\ldots r_n$ là dãy n bit nhận được.

**Qui ước:** v', w' (lần lượt là chuyển vị của v và w) được viết theo dòng. Còn v, w được viết theo cột.

Nếu A.v = 0 thì v = w, ta gọi v là chẵn (trường hợp nhận đúng)

Nếu A.v ≠ 0 thì v ≠ w, ta gọi v là lẻ (trường hợp nhận sai).

Ta gọi z = v-w là bộ lỗi giữa v và w. Nghĩa là tại các vị trí z = {0} thì bit nhận được tương ứng là bit đúng và tại các vị trí z = {1} thì bit nhận được tương ứng là bit sai (hay bit lỗi).
Ta gọi C = A.v là bộ sửa lỗi (hay bộ điều chỉnh lỗi).
Ta có C = A.z = A.(v-w) = A.v-A.w = A.v $\Rightarrow$ **C = A.v = A.z**

**Tính chất của bộ sửa lỗi:** dãy n bit nhận được v và bộ lỗi tương ứng có cùng bộ điều chỉnh.

## Phương pháp sinh mã kiểm tra chẵn lẻ

Giả sử: cho trước ma trận kiểm tra chẵn lẻ A với Rank(A) = m.
Tìm bộ mã chẵn lẻ W={$w_1$, $w_2$, $w_3$,…,$w_s$}

**Bước 0:** Xác định các giá trị n, m, k, s

Độ dài của từ mã n= số cột của ma trận A.
Số bit kiểm tra m= số dòng của ma trận A.
Số bit thông tin: k = n-m.
Số từ mã s=$2^k$ của bộ mã.

**Bước i:** Tìm các từ mã thứ i ($1 \le i \le s$):

Gọi $kp_i$ là triển khai nhị phân k bit của số i
Từ mã cần tìm là: $w'_i=r_1r_2..r_mkp_i$
Giải hệ phương trình $A.w_i=0$ để tìm m bit kiểm tra ứng với k bit thông tin ($kp_i$) đã biết
=> từ mã $w_i$

## Ví dụ sinh mã kiểm tra chẵn lẻ

Xây dựng bộ mã kiểm tra chẵn lẻ được sinh từ ma trận kiểm tra A như sau:

$$A=\begin{bmatrix}1&0&0&1&1&0\\0&1&1&1&0&1\\1&0&1&1&0&1\end{bmatrix}\quad \text{Rank(A) = 3}$$

Bước 0:

n=6 (= số dòng của ma trận A)
m=3 (= số cột của ma trận A)
Số bit thông tin k = n – m = 3 => Số từ mã s=$2^k$=8 từ mã.

**Bước i:** Tìm từ mã thứ i ($1 \le i \le s$):

$w'_1=r_1r_2r_3000$ (000 là triển khai nhị phân k=3 bits của số i=0)
$w'_1=r_1r_2r_3001$ (001 là triển khai nhị phân k=3 bits của số i=1)
$w'_2=r_1r_2r_3010$ (010 là triển khai nhị phân k=3 bits của số i=2)
$w'_3=r_1r_2r_3011$ (011 là triển khai nhị phân k=3 bits của số i=3)
$w'_4=r_1r_2r_3100$ (100 là triển khai nhị phân k=3 bits của số i=4)
$w'_5=r_1r_2r_3101$ (101 là triển khai nhị phân k=3 bits của số i=5)
$w'_6=r_1r_2r_3110$ (110 là triển khai nhị phân k=3 bits của số i=6)
$w'_7=r_1r_2r_3111$ (111 là triển khai nhị phân k=3 bits của số i=7)

Giải hệ phương trình $A.w_1=0$

$$\begin{bmatrix}1&0&0&1&1&0\\0&1&1&1&0&1\\1&0&1&1&0&1\end{bmatrix}\begin{bmatrix}r_1\\r_2\\r_3\\0\\0\\1\end{bmatrix}=\begin{bmatrix}0\\0\\0\end{bmatrix} \Rightarrow \begin{cases}r_1=0\\r_2+r_3=1\\r_1+r_3=1\end{cases} \Rightarrow \begin{cases}r_1=0\\r_2=0\\r_3=1\end{cases} \Rightarrow w'_1=001001$$

Giải hệ phương trình $A.w_2=0$

$$\begin{bmatrix}1&0&0&1&1&0\\0&1&1&1&0&1\\1&0&1&1&0&1\end{bmatrix}\begin{bmatrix}r_1\\r_2\\r_3\\0\\1\\0\end{bmatrix}=\begin{bmatrix}0\\0\\0\end{bmatrix} \Rightarrow \begin{cases}r_1=1\\r_2+r_3=0\\r_1+r_3=0\end{cases} \Rightarrow \begin{cases}r_1=1\\r_2=1\\r_3=1\end{cases} \Rightarrow w'_2=111010$$

Giải tương tự cho các trường hợp còn lại ta có:

$w'_0=000000$, $w'_3=110011$, $w'_4=110100$,
$w'_5=111101$, $w'_6=001110$, $w'_7=000111$.

$\Rightarrow$ W={000000, 001001, 111010, 110011, 110100, 111101, 001110, 000111}

## Định lý quan hệ giữa độ dài mã n, số bit kiểm tra m và số lỗi tự sửa e

Điều kiện cần (Cận Hamming):
Điều kiện cần để bộ mã chẵn lẻ có độ dài n bit có thể tự sửa được e bit lỗi với k bit thông tin và m bit kiểm tra là:

$$2^m \ge \sum_{i=0}^{e} C_n^i$$

Điều kiện đủ ( ĐK Vasharmov-Gilbert-Sacks):
Điều kiện đủ để bộ mã kiểm tra chẵn lẻ có độ dài n bit với m bit kiểm tra chẵn lẻ có thể tự sửa được e bit lỗi là:

$$2^m > \sum_{i=0}^{2e-1} C_{n-1}^i$$

Ghi chú: $C_n^i = n!/(i!*(n-i)!)$

### Ví dụ tìm m nhỏ nhất từ n và e

Giả sử biết trước n=7 và e=1. Tìm số bit kiểm tra tối thiểu cần thiết của bộ mã chẵn lẻ.
Theo định lý điều kiện cần (Cận Hamming):

Ta có: $2^m \geq \sum_{i=0}^{e} C_n^i$

$\Leftrightarrow \quad 2^m \geq \sum_{i=0}^{e=1} C_7^i \quad (*)$

m = 1 $\Rightarrow$ (*) sai.

m = 2 $\Rightarrow$ (*) sai.

m $\geq$ 3 $\Rightarrow$ (*) đúng.

Vậy số bit kiểm tra tối thiểu cần thiết là m = 3.

### Ví dụ tìm e lớn nhất từ m và n

Giả sử cho trước m=3, k=2. Tìm số bit lỗi lớn nhất có thể tự sửa e?
Theo định lý điều kiện đủ (ĐK Vassharmov-Gilbert-Sacks):

$2^m \geq \sum_{i=0}^{2e-1} C_{n-1}^i \Leftrightarrow 2^3 \geq \sum_{i=0}^{2e-1} C_{5-1}^i \quad (*)$

e =1 $\Rightarrow$ (*) đúng.

e > 1 $\Rightarrow$ (*) sai.

Vậy số bit lỗi lớn nhất có thể tự sửa là e = 1.

### Bài tập

**1.** Xây dựng bộ mã kiểm tra chẵn lẻ được sinh từ ma trận kiểm tra A như sau:

$$A = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

**2.** Tìm bộ mã kiểm tra chẵn lẻ được sinh từ ma trận kiểm tra A như sau:

$$A = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

- **Gợi ý giải bài tập 1 & 2:** dựa vào phương pháp sinh mã kiểm tra chẵn lẻ và tham khảo ví dụ sinh mã kiểm tra chẵn lẻ.

**3.** Xét bộ mã kiểm tra chẵn lẻ độ dài 15 bit có thể tự sửa được 1 bit lỗi trên đường truyền, hãy cho biết số bit kiểm tra chẵn lẻ tối thiểu?
**4.** Xét bộ mã kiểm tra chẵn lẻ độ dài 8 bit với 4 bit kiểm tra chẵn lẻ. Hãy cho biết số lỗi tự sửa tối đa của bộ mã?
**Gợi ý giải bài tập 3 & 4:** dựa vào đinh lý Điều kiện cần (Cận Hamming) và Điều kiện đủ (ĐK Varshamov-Gilbert-Sacks).

# BÀI 5.4: NHÓM CỘNG TÍNH VÀ BỘ TỪ MÃ CHẴN LẺ

## Mục tiêu.

Sau khi hoàn tất bài học này bạn có thể:

- Hiểu Khái niệm nhóm cộng tính,
- Biết các tính chất của bộ mã chẵn lẻ,
- Vận dụng sinh ma trận kiểm tra chắn lẻ từ bộ mã kiểm tra chẵn lẻ.
- Vận dụng tốt phương pháp sinh bộ mã kiểm tra chẵn lẻ từ các từ mã độc lập tuyến tính của bộ mã.

## Khái niệm nhóm cộng tính.

Đặt vấn đề:
Như chúng ta đã biết, phương pháp sinh mã kiểm tra chẵn lẻ giúp ta sinh bộ mã kiểm tra chẵn lẻ với số từ mã tương ứng là $s=2^k$. Với phương pháp này, ta phải xác định từng từ mã một (bằng cách giải hệ phương trình tuyến tính nhị phân). Giả sử: k=5 ta phải xác định $s=2^5$ =32 từ mã hay k=10 ta phải xác định $s=2^{10}$=1024 từ mã,…Điều này sẽ mất nhiều thời gian nếu k càng lớn. Vấn đề đặt ra ở đây là tìm ra một phương pháp sinh bộ mã kiểm tra chẵn lẻ nhanh hơn về mặt thời gian. Phương pháp sinh mã kiểm tra chẵn lẻ dựa theo lý thuyết nhóm sẽ giải quyết vấn đề này.

Khái niệm nhóm cộng tính:
Nhóm G được gọi là một nhóm cộng tính nếu G có các tính chất:

- $\forall$ a, b $\in$ G $\Rightarrow$ a+b $\in$ G ( tính chất cộng).
- $\forall$ a, b, c $\in$ G $\Rightarrow$ a + (b + c)= (a + b) + c ( tính chất kết hợp).
- $\exists \varnothing \in$ G sao cho $\varnothing$ + a = a + $\varnothing$ = a, $\forall$a$\in$ G ($\varnothing$ là Identity Element của G).
- $\forall$ a $\in$ G $\exists$ -a$\in$G : a + (-a)=$\varnothing$

Nhóm G là nhóm hoán vị (nhóm Aben) nếu $\forall$a,b $\in$ G=> a + b = b + a.

Ví dụ:

- Tập hợp các số nguyên với phép + thông thường là nhóm Aben.
- Tập hợp các số nhị phân có độ dài n bit cùng với phép + trong Modulo 2 tạo thành nhóm Aben.

## Tính chất của bộ mã chẵn lẻ

Tính tương đương của bộ mã nhóm cộng tính và bộ từ mã kiểm tra chẵn lẻ được thể hiện qua 2 định lý sau:

**Định lý 1:** tập hợp các từ mã trong bộ mã kiểm tra chẵn lẻ là một nhóm cộng tính.
(Đề nghị sinh viên chứng minh định lý này dựa vào các tính chất của nhóm cộng tính)

**Định lý 2:** Nếu tập hợp W là tập các dãy nhị phân với độ dài các dãy cùng bằng n và W là một nhóm Aben với phép cộng Modulo 2 thì W có thể xem như một bộ mã kiểm tra chẵn lẻ được sinh ra từ ma trận A có dạng như sau:

$$A = \begin{bmatrix} & b_{11} & b_{12} & \dots & b_{1k} \\ I_m & b_{21} & b_{22} & \dots & b_{2k} \\ & \dots & \dots & \dots & \dots \\ & b_{m1} & b_{m2} & \dots & b_{mk} \end{bmatrix}$$

Trong đó:

- Ma trận A có m dòng và n cột.
- $I_m$ : là ma trận đơn vị cấp m.
- k: là số dãy nhị phân (hay từ mã) độc lập tuyến tính lớn nhất.
- n: là độ dài của từ mã và m = n-k:
- bij: được xác định bằng cách dựa vào hệ phương trình tuyến tính (*) và k từ mã độc lập tuyến tính như sau:

$w'_i = \underbrace{r_1 r_2 r_3 \dots r_m}_{\text{Đoạn kiểm tra}} \ \underbrace{r_{m+1} r_{m+2} \dots r_n}_{\text{Đoạn thông tin}} \ (\forall i = \overline{1,k})$

$$(*)\begin{cases} r_1 = b_{11} r_{m+1} + \dots + b_{1k} r_{m+k} \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ r_m = b_{m1} r_{m+1} + \dots + b_{mk} r_{m+k} \end{cases}$$

Thế k từ mã độc lập tuyến tính vào hệ pt (*) để tìm các $b_{ij} \Rightarrow$ ma trận A.

## Ví dụ minh họa

Xét tập hợp M gồm có 8 dãy nhị phân dài 6 bits như sau:

| | $r_1$ | $r_2$ | $r_3$ | $r_4$ | $r_5$ | $r_6$ | |
|---|---|---|---|---|---|---|---|
| $w'_0$ = | 0 | 0 | 0 | 0 | 0 | 0 | |
| $w'_1$ = | 1 | 0 | 1 | 0 | 0 | 1 | |
| $w'_2$ = | 1 | 1 | 0 | 0 | 1 | 0 | |
| $w'_3$ = | 0 | 1 | 0 | 1 | 0 | 1 | |
| $w'_4$ = | 0 | 1 | 1 | 0 | 1 | 1 | ($w'_1$+$w'_2$) |
| $w'_5$ = | 1 | 1 | 1 | 1 | 0 | 0 | ($w'_1$+$w'_3$) |
| $w'_6$ = | 1 | 0 | 0 | 1 | 1 | 1 | ($w'_2$+$w'_3$) |
| $w'_7$ = | 0 | 0 | 1 | 1 | 1 | 0 | ($w'_1$+$w'_2$+$w'_3$) |

Ta thấy $\{w_1, w_2, w_3\}$ là tập hợp lớn nhất các từ mã độc lập tuyến tính từ tập hợp M:

| | | | | | | |
|---|---|---|---|---|---|---|
| $w'_1$ = | 1 | 0 | 1 | 0 | 0 | 1 |
| $w'_2$ = | 1 | 1 | 0 | 0 | 1 | 0 |
| $w'_3$ = | 0 | 1 | 0 | 1 | 0 | 1 |

$\Rightarrow$ n=6 và k=3. => m = n – k = 3.

Như vậy: ma trận kiểm tra chẵn lẻ có dạng như sau:

$$A = \begin{bmatrix} & b_{11} & b_{12} & b_{13} \\ I_3 & b_{21} & b_{22} & b_{23} \\ & b_{31} & b_{32} & b_{33} \end{bmatrix}$$

Các $b_{ij}$ ( $\forall i, i = \overline{1,3}$ ) được xác định từ hệ phương trình tuyến tính nhị phân sau:

$$\Rightarrow \begin{cases} \begin{pmatrix}1\\1\\0\end{pmatrix} = b_{11}\begin{pmatrix}0\\0\\1\end{pmatrix} + b_{12}\begin{pmatrix}0\\1\\0\end{pmatrix} + b_{13}\begin{pmatrix}1\\0\\1\end{pmatrix} \\ \begin{pmatrix}0\\1\\1\end{pmatrix} = b_{21}\begin{pmatrix}0\\0\\1\end{pmatrix} + b_{22}\begin{pmatrix}0\\1\\0\end{pmatrix} + b_{23}\begin{pmatrix}1\\0\\1\end{pmatrix} \\ \begin{pmatrix}1\\0\\0\end{pmatrix} = b_{31}\begin{pmatrix}0\\0\\1\end{pmatrix} + b_{32}\begin{pmatrix}0\\1\\0\end{pmatrix} + b_{33}\begin{pmatrix}1\\0\\1\end{pmatrix} \end{cases}$$

$$\Rightarrow \begin{cases} b_{11} = 1 \;\; b_{12} = 1 \;\; b_{13} = 1 \\ b_{21} = 1 \;\; b_{22} = 1 \;\; b_{23} = 0 \\ b_{31} = 1 \;\; b_{32} = 0 \;\; b_{33} = 1 \end{cases}$$

$$\Rightarrow A = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}$$

Vậy ta có thể sử dụng nhóm M như là một bộ mã kiểm tra chẵn lẻ.

## Phương pháp sinh mã kiểm tra chẵn lẻ nhanh

Bước khởi tạo: xác định các giá trị n, m, k, s.
Bước 1: sinh k từ mã độc lập tuyến tính (đltt).
Bước 2: cộng tổ hợp các từ mã:

+ Cộng các tổ hợp của 2 từ mã từ k mã đltt => có $C_k^2$ từ mã.

---

+ Cộng các tổ hợp của k từ mã từ k từ mã đltt => có $C_k^k$ từ mã.

Bước 3: Cộng s-1 từ mã đã tìm được để tìm từ mã cuối cùng => $C_k^0 = 1$ từ mã.

Tổng số từ mã s= $\sum_{i=0}^{k} C_k^i = 2^k$ từ mã.

## Ví dụ sinh mã kiểm tra chẵn lẻ nhanh

Tìm bộ mã nhóm khi biết trước ma trận kiểm tra $A = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$

Bước khởi tạo: n = 6, m = 3, k = 3, s = $2^k$ = 8.

Bước 1: Sinh k = 3 từ độc lập truyến tính: $w'_1$=001001, $w'_2$=111010, $w'_3$=110100
Bước 2: Cộng tổ hợp các từ mã.
+ Cộng các tổ hợp 2 từ mã đltt:
$w'_4=w'_1+w'_2$=110011
$w'_5=w'_1+w'_3$=111101
$w'_6=w'_2+w'_3$=001110
+ Cộng các tổ hợp 3 từ mã đltt:
$w'_7=w'_1+w'_2+w'_3$=001111
Bước 3: xác định từ mã cuối cùng:
$w'_0=w'_1+w'_2+w'_3+w'_4+w'_5+w'_6+w'_7$=000000

## Bài tập

**1.** Sử dụng phương pháp sinh mã nhanh cho bộ mã từ ma trận kiểm tra A như sau:

$$A=\begin{bmatrix}1&0&0&1&1&1\\0&1&1&1&0&1\\1&0&1&1&0&1\end{bmatrix}$$

**2.** Sử dụng phương pháp sinh mã nhanh cho bộ mã từ ma trận kiểm tra A trong các trường hợp sau:

$$A=\begin{pmatrix}0&0&0&1&0&1\\0&0&1&0&1&0\\0&1&0&1&0&0\\1&0&1&0&0&0\end{pmatrix};\quad A=\begin{pmatrix}0&0&1&1&1&1\\0&1&1&1&1&0\\1&1&1&1&0&0\\1&1&1&0&0&1\end{pmatrix};\quad A=\begin{pmatrix}1&0&0&0&1&1\\0&1&0&0&1&0\\0&0&1&0&1&1\\0&0&0&1&0&1\end{pmatrix}$$

# BÀI 5.5: LƯỢC ĐỒ SỬA LỖI TỐI ƯU

## Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:
- Biết được vấn đề của bài toán,
- Hiểu Định nghĩa Hiệp hợp,
- Vận dụng để xây dựng lược đồ sửa lỗi theo các hiệp hợp,
- Vận dụng để xây dựng lược đồ sửa lỗi thông qua bộ sửa lỗi,
- Vận dụng tính Xác suất truyền đúng cho lược đồ sửa lỗi,
- Kiến thức đạt được sẽ là cơ sở để các bạn có thể ứng dụng cho việc thiết kế một hệ, thống mã hóa, giải mã và bảo mật thông tin.

## Đặt vấn đề

Trong một hệ thống liên lạc truyền tin, bên cạnh các yêu cầu thiết bị (như nguồn phát, bộ mã hóa, kênh truyền, bộ giải mã,…) đảm bảo tốt cho việc truyền và nhận dữ liệu thì còn có các khía cạnh khác như phương pháp mã hóa và giải mã sao cho tối ưu là phần rất quan trọng trong hệ thống. Vấn đề luôn được đặt ra ở đây là làm thế nào để chỉ ra một phương pháp giải mã tối ưu, có nghĩa là hệ thống phải có khả năng phát hiện và sửa lỗi một cách chính xác nhất có thể có khi nhiễu xảy ra. Đây chính là vấn đề chính được thảo luận trong suốt bài học này.

## Định nghĩa Hiệp hợp

Gọi $W=\{w_1, w_2, \ldots, w_s\}$ là bộ mã kiểm tra chẵn lẻ.
$V=\{v_1, v_2, \ldots, v_{2^n}\}$ là tập hợp các dãy n bit có thể nhận được ở cuối kênh.
Ta gọi một hiệp hợp của W trong V là tập hợp có dạng z + W (z là bộ lỗi)

**Ví dụ:** Cho ma trận kiểm tra chẵn lẻ sau:

$$A=\begin{bmatrix}1 & 0 & 1 & 0\\ 0 & 1 & 1 & 1\end{bmatrix}$$

Từ A, ta có thể xây dựng được bộ mã tương ứng sau: $W=\{w'_0=0000, w'_1=0101, w'_2=1110, w'_3=1011\}$.

Ta có thể thấy rằng, các bộ lỗi một bit khác nhau có thể có là z={1000, 0100, 0010, 0001}. Do đó các hiệp hợp ứng với các bộ lỗi 1 bit sẽ là:

| | $w_0$ | $w_1$ | $w_2$ | $w_3$ | |
|---|---|---|---|---|---|
| | 0000 | 0101 | 1110 | 1011 | |
| Hiệp hợp 1 | 1000 | 1101 | 0110 | 0011 | (với $z_1$=1000) |
| Hiệp hợp 2 | 0100 | 0001 | 1010 | 1111 | (với $z_2$=0100) |
| Hiệp hợp 3 | 0010 | 0111 | 1100 | 1001 | (với $z_3$=0010) |
| Hiệp hợp 4 | 0001 | 0100 | 1111 | 1010 | (với $z_4$=0001) |

Trong đó: hiệp hợp $i = w_i + z_i$, các bạn có thể xét thêm các bộ lỗi sai 2 bit, 3 bit, … để được các hiệp hợp ứng với các bộ lỗi sai 2 bit, 3bit,….

## Lược đồ sửa lỗi theo các hiệp hợp

**Bước 1:** Lập bảng các hiệp hợp ứng với các bộ lỗi cần thiết

- Dòng đầu tiên viết các từ mã $w_i \in W$.
- Các dòng tiếp theo ứng với cột $w_0 = 00\ldots00$ viết các bộ lỗi z (các bộ lỗi 1 bit, 2 bit,…).
- Các dòng ở cột thứ i được xác định bởi $z + w_i$

**Bước 2:** Quá trình giải mã

Giải mã: khi nhận v, ta xác định cột thứ i chứa v và giải mã về $w_i$ tương ứng.

**Ví dụ:** xây dựng lược đồ sửa lỗi theo các hiệp hợp cho bộ mã được sinh từ ma trận kiểm tra chẵn lẻ sau:

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

Từ A, ta có thể xây dựng được bộ mã tương ứng sau: W={w'$_0$=0000, w'$_1$=0101, w'$_2$=1110, w'$_3$=1011}.

**Bước 1:** Lập bảng các hiệp hợp ứng với các bộ lỗi cần thiết:

Ta xây dựng các hiệp hợp ứng với các bộ lỗi sai 1 bit. Vậy z={1000, 0100, 0010, 0001}.

| | $w_0$ | $w_1$ | $w_2$ | $w_3$ | |
|---|---|---|---|---|---|
| | 0000 | 0101 | 1110 | 1011 | |
| Hiệp hợp 1 | 1000 | 1101 | 0110 | 0011 | (với $z_1$=1000) |
| Hiệp hợp 2 | 0100 | 0001 | 1010 | 1111 | (với $z_2$=0100) |
| Hiệp hợp 3 | 0010 | 0111 | 1100 | 1001 | (với $z_3$=0010) |
| Hiệp hợp 4 | 0001 | 0100 | 1111 | 1010 | (với $z_4$=0001) |

**(Bảng các hiệp hợp)**

**Bước 2:** Quá trình giải mã:

Giả sử nhận v = 0111. Tra tìm v trên bảng các Hiệp hợp ta có v ở cột 1. Do đó, v được giải mã về $w_1$ = 0101.

Giả sử nhận v = 1010. Tra tìm v trên bảng các Hiệp hợp ta có v ở cột 2 hay cột 3. Do đó, v được giải mã về $w_2$ hay $w_3$, trong trường hợp này giải mã không chính xác. Đề nghị các bạn lưu ý và cho ý kiến của bạn về các trường hợp giải mã không chính xác này.

## Lược đồ sửa lỗi thong qua bộ lỗi

Để xây dựng lược đồ sửa lỗi thông qua bộ sửa lỗi, ta dựa vào tính chất của bộ sửa lỗi. Như vậy ta có thể thấy lược đồ giải mã gồm 2 bước sau:

**Bước 1:** Lập bảng sửa lỗi: Bộ lỗi (Z) – Bộ sửa lỗi (C=A*Z).

**Bước 2:** Quá trình sửa lỗi

- Khi nhận được dãy n bit $v \in V$, ta xác định bộ điều lỗi C cho v với C=A.v
- Tra bảng sửa lỗi để tìm bộ lỗi $z_0$ ứng với C.
- Giải mã $w=v+z_0$.

Ví dụ minh họa lược đồ sửa lỗi 1 bit

Xét bộ mã được sinh từ ma trận $A = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$

Bộ mã tương ứng được xác định là: $w_1$=000000, $w_2$=101101, $w_3$=111010, $w_4$=010111

(Đề nghị các bạn tham khảo phương pháp sinh mã chẵn lẻ và xây dựng lại bộ mã từ ma trận kiểm tra chẵn lẻ A).

Lược đồ sửa lỗi:
Bước 1: Lập bảng sửa lỗi: Bộ lỗi- Bộ điều chỉnh (e = 1)

| | Bộ lỗi (z') | Bộ điều chỉnh (C'=A.z) | |
|---|---|---|---|
| Bộ 0 lỗi | 000000 | 0000 | 1 Bộ |
| Bộ lối 1 bit | 100000 | ← 1000 | |
| | 010000 | 0100 | |
| | 001000 | 0010 | 6 Bộ |
| | 000100 | 0001 | |
| | 000010 | 1110 | |
| | 000001 | 1011 | |

Bước 2: Quá trình sửa lỗi

- Giả sử nhận v=001101, tính C = A.v = 1000
- Tra bảng sửa lỗi để tìm bộ lỗi $z_0$ ứng với C, ta có $z_0$ = 100000
- Giải mã w = v + $z_0$ = 001101 + 100000 = 101101 = $w_2$

## Ví dụ minh họa lược đồ sửa lỗi 2 bit

Lược đồ sửa lỗi:
Bước 1: Lập bảng sửa lỗi: Bộ lỗi- Bộ điều chỉnh (e = 2)

| | Bộ lỗi (z') | Bộ điều chỉnh (C'=A.z) | |
|---|---|---|---|
| Bộ lỗi 2 bit | 110000 | 1100 | |
| | 101000 | 1010 | |
| | 100100 | 1001 | |
| | 100010 | 0110 | |
| | 100001 | 0011 | 7 Bộ |
| | 011000 | 0110 | |
| | 010100 | 0101 | |

(Tất cả các bộ 2 lỗi còn lại có trùng bộ điều chỉnh với các bộ ở trên)

Bước 2: Quy trình sửa lỗi

- Giả sử nhận v=100111, tính C = A.v = 1100
- Tra bảng sửa lỗi để tìm bộ lỗi $z_0$ ứng với C, ta có $z_0$ = 110000
- Giải mã w = v + $z_0$ = 100111 + 110000 = 010111 = $w_4$

### Ví dụ minh họa lược đồ sửa lỗi 3 bit

Lược đồ sửa lỗi:
Bước 1: Lập bảng sửa lỗi: Bộ lỗi- Bộ điều chỉnh (e = 3)

| | z' | C=A.z | |
|---|---|---|---|
| Bộ lỗi 3 bit | 110100 | 1101 | 2 Bộ |
| | 110001 | 0111 | |

(Tất cả các bộ 3 lỗi còn lại có trùng bộ điều chỉnh với các bộ ở trên)

Bước 2: Quy trình sửa lỗi
Giả sử nhận v=011001, tính C = A.v = 1101
Tra bảng sửa lỗi để tìm bộ lỗi $z_0$ ứng với C, ta có $z_0$ = 110100
Giải mã w=v + $z_0$ = 011001 + 110100 = 101101 = $w_2$
Chú ý:
Tổng số bộ điều chỉnh = $2^m$. Trong một số trường hợp, bộ mã chẵn lẻ chỉ cho phép phát hiện lỗi trên đường truyền và không thể giải mã chính xác do tổng số bộ điều chỉnh = $2^m$ và số bộ lỗi có thể lớn hơn nhiều (so với tổng số bộ điều chỉnh).

### Xác suất truyền đúng

Gọi $N_i$ là số bộ lỗi ứng với i lỗi có thể tự sửa, khi đó xác suất truyền đúng và tự điều chỉnh sẽ là:

$$P(e') = \sum_{i=0}^{n} Ni.\beta i.(1-\beta i)^{n-i}$$

Với n là độ dài từ mã
**Ví dụ:** xét trường hợp các ví dụ trên với n= 6 và tự sửa e = 3 bit lỗi. Áp dụng công thức trên ta có:

$$P(e') = \sum_{i=0}^{3} Ni.\beta i.(1-\beta i)^{n-i} = (1-\beta)^6 + 6\beta(1-\beta)^5 + 7\beta^2(1-\beta)^4 + 2\beta^3(1-\beta)^3$$

### Bài tập

1. Cho ma trận kiểm tra chẵn lẻ sau:

$$A = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

- Xây dựng bộ mã kiểm tra chẵn lẻ.
- Minh họa quy trình sửa lỗi 1 bit.

2. Từ kết quả của bài tập 1, hãy minh họa lược đồ sửa lỗi thông qua bộ điều chỉnh trong các trường hợp lỗi 1 bit, 2 bit. Tính xác suất truyền đúng cho các trường hợp có thể tự sửa được.

# BÀI 5.6: MÃ HAMMING

### Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:

- Hiểu Mã Hamming,
- Hiểu tính chất của mã Hamming.

## Mã Hammin

Mã Hamming là một dạng mã nhóm (mã kiểm tra chẵn lẻ) được xác định từ ma trận kiểm tra chẵn lẻ A có dạng sau:

- Cột thứ j của ma trận A là biểu diễn nhị phân m bit (m là số bit kiểm tra chẵn lẻ) của số j theo qui ước biểu diễn nhị phân của số j được viết theo thứ tự từ dưới lên trên (viết theo cột), tương đương với viết từ trái sang phải (viết theo dòng).
- Các bit ở vị trí $2^i$ ( i = 0, 1, 2, …) được chọn làm bit kiểm tra.

**Ví dụ 1:** biểu diễn nhị phân của số j = 3 có m = 3 bit như sau:

Viết theo dòng: 011 (viết từ trái sang phải)

Viết theo cột: $\begin{pmatrix}1\\1\\0\end{pmatrix}$ (viết từ dưới lên)

**Ví dụ 2:** ma trận kiểm tra chẵn lẻ với n=6, m=3 có thể viết như sau:

$$A=\begin{bmatrix}1&0&1&0&1&0\\0&1&1&0&0&1\\0&0&0&1&1&1\end{bmatrix}$$

Từ mã Hamming có dạng: $w=r_1r_2r_3r_4r_5r_6$. Trong đó, $r_1r_2r_4$ là các bit kiểm tra và $r_3r_5r_6$ là các bit thông tin (vì các bit ở vị trí $2^i$ (với i = 0, 1, 2, …) được chọn làm bits kiểm tra).

## Tính chất

Nếu cho trước số bit (m) và số bit lỗi tự sửa (e) thì số bit tối đa của bộ mã Hamming (n) có thể được ước lượng từ bất đẳng thức sau:

$$2^m \geq \sum_{i=o}^{e} C_n^i$$

## Ví dụ minh họa

Tìm bộ mã Hamming với n = 6 và m =3

Ta có thể viết ngay ma trận kiểm tra chẵn lẻ cho bộ mã Hamming

$$A=\begin{bmatrix}1&0&1&0&1&0\\0&1&1&0&0&1\\0&0&0&1&1&1\end{bmatrix}$$

Từ A $\Rightarrow$ k = n – m = 3.

Các bit ở các vị trí 1, 2, 4 được chọn làm các bit kiểm tra.

=> số từ mã của bộ mã Hamming là $s = 2^k = 8$

Tìm k từ mã độc lập tuyến tính có dạng:

$w'_1=r_1r_20r_401$

$w'_2=r_1r_20r_410$

$w'_3=r_1r_21r_400$

Giải các hệ phương trình: $A.w_1=0$, $A.w_2=0$, $A.w_3=0$

Các từ mã còn lại được xác định theo phương pháp sinh mã nhanh.

**Ghi chú:** *Kết quả chi tiết xây dựng bảng mã Hamming dành cho sinh viên tự làm.*

## Bài tập

1. Viết ma trận kiểm tra chẵn lẻ cho bộ mã Hamming với n = 15.
2. Từ kết quả bài tập 1, hãy tìm các từ mã Hamming độc lập tuyến tính tương ứng.
3. Xét bộ mã Hamming với số bit kiểm tra cho trước là m, khi đó:
   - Độ dài mã tối thiểu là bao nhiêu?
   - Độ dài mã tối đa là bao nhiêu?

# BÀI 5.7: THANH GHI LÙI TỪNG BƯỚC

## Mục tiêu

Sau khi hoàn tất bài học này bạn có thể biết:

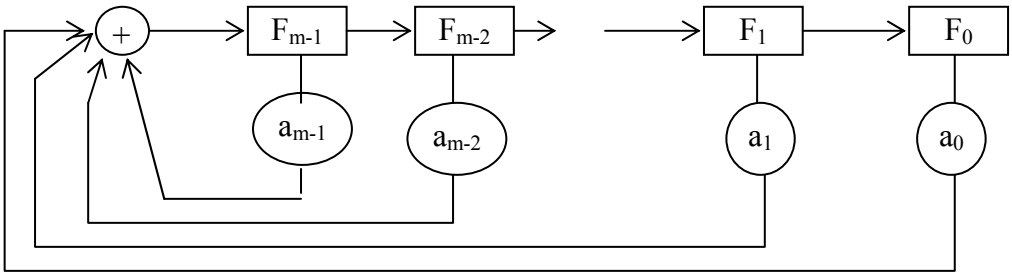
- Đặt vấn đề về thanh ghi lùi từng bước,
- Cách biểu diễn vật lý của thanh ghi,
- Cách biểu diễn toán học của thanh ghi,
- Tìm chu kỳ của thanh ghi.

## Đặt vấn đề

Như chúng ta đã biết, phương pháp sinh bộ mã kiểm tra chẵn lẻ dựa trên lý thuyết nhóm cho phép chúng ta sinh mã nhanh bằng cách chỉ sinh ra k từ mã độc lập tuyến tính trong tổng số $s=2^k$ từ mã, từ k từ mã này ta có thể xác định các từ mã còn lại (bằng cách cộng tổ hợp các từ mã). Vấn đề đặt ra ở đây là làm sao để tìm ra một phương pháp sinh mã khác sao cho số từ mã sinh ban đầu nhỏ hơn k (k là số từ mã độc lập tuyến tính của bộ mã kiểm tra chẵn lẻ) và từ đây ta có thể xác định nhanh các từ mã còn. Cụ thể dựa trên mô hình của thanh ghi lùi từng bước có thể giải quyết được vấn đề này.

## Biểu diễn vật lý của thanh ghi

Để gọi một cách ngắn gọn, ta qui ước gọi thanh ghi thay vì goi thanh ghi lùi từng bước. Biểu diễn vật lý của thanh ghi có thể thấy như hình vẽ dưới đây:




- $F_{m-1}, F_{m-2}, \ldots, F_1, F_0$ : các bit lưu trữ dữ liệu nhị phân.
- $a_{m-1}, a_{m-2}, \ldots, a_1, a_0$ : các công tắc (switch) dùng để đóng (=1) hay mở ( =0).
- $\oplus$ : là bộ làm tính cộng trong phép toán mudulo 2 sau mỗi xung đồng hồ với dữ liệu do các bit của thanh ghi gửi về.

**Quá trình dịch chuyển lùi từng bước:** sau mỗi xung đồng hồ thì dữ liệu trong bit $F_i$ sẽ được chuyển về lưu trữ ở bit $F_{i-1}$ ($F_1$-> $F_0$; $F_2$->$F_1$; …; $F_{m-2}$->$F_{m-3}$; $F_{m-1}$->$F_{m-2}$). Tất cả các giá trị trên các $F_i$ (trước khi có xung điện) sẽ được chuyển về bộ cộng (tùy theo các công tắc đóng hay mở), tổng của các giá trị này sẽ được đưa vào lưu trữ ở bit $F_{m-1}$.

Ta sẽ nghiên cứu thanh ghi này cụ thể hơn trong các nội dung tiếp theo nhằm tìm ra một phương pháp sinh mã mà ta có thể gọi là mã xoay vòng. Đây cũng là một dạng mã kiểm tra chẵn lẻ.

## Biểu diễn toán học của thanh ghi

Mục tiêu của việc biểu diễn toán học là để tìm ra các mô hình tính toán phục vụ cho việc nghiên cứu sinh mã xoay vong chẵn lẻ từ thanh ghi.

Gọi $x = (x_0, x_1, \ldots, x_{m-2}, x_{m-1})$ là giá trị các bit của thanh ghi tại thời điểm trước khi có nhịp xung đồng hồ.

$x' = (x'_0, x'_1, \ldots, x'_{m-2}, x'_{m-1})$ là giá trị các bit của thanh ghi sau khi có nhịp xung đồng hồ.

Khi đó ta có:

$x'_0 = x_1$

$x'_1 = x_2$

$x'_2 = x_3$

---

$x'_{m-2} = x_{m-1}$

$x'_{m-1} = a_0x_0 + a_1x_1 + \ldots + a_{m-1}x_{m-1}$.

Hay viết theo dạng ma trận ta có $x' = T.x$

Trong đó:

$$T=\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & ... & 0 & 0 \\ ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & 0 & 0 & .. & 0 & 1 \\ a_0 & a_1 & a_2 & a_3 & ... & a_{m-2} & a_{m-1} \end{bmatrix}$$

- T: Ma trận vuông cấp m.
- Dòng cuối của ma trân: là các hệ số: $a_0, a_1, \ldots, a_{m-1}$.
- Gốc trên bên phải: là ma trận đơn vị cấp m-1.

T được gọi là ma trận đặc trưng của thanh ghi lùi từng bước.

Quá trình dịch chuyển lùi từng bước của thanh ghi:

Gọi $x^{(0)}=\begin{pmatrix} x_0 \\ x_2 \\ x_3 \\ \vdots \\ x_{m-1} \end{pmatrix}$ là véc tơ chỉ giá trị của thanh ghi tại thời điểm đang xét.

Giá trị của thanh ghi sau 1 xung đồng hồ là $x^{(1)}=T.x^{(0)}$

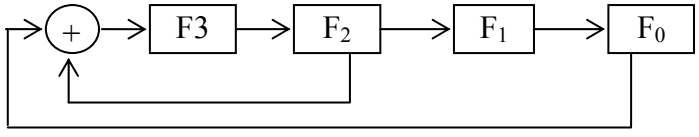
Giá trị của thanh ghi sau 2 xung đồng hồ là $x^{(2)}=T.x^{(1)}=T^2.x^{(0)}$

Giá trị của thanh ghi sau 3 xung đồng hồ là $x^{(3)}=T.x^{(2)}=T^3.x^{(0)}$

---

## Ví dụ thanh ghi lui từng bước

Cho thanh ghi lui từng bước sau:




Từ thanh ghi, ta có: m=4, $a_0$=1, $a_1$=0, $a_2$=1, $a_3$=0.

Ma trận đặc trưng của thanh ghi: $T=\begin{bmatrix}0&1&0&0\\0&0&1&0\\0&0&0&1\\1&0&1&0\end{bmatrix}$

## Chu kỳ của thanh ghi

Như đã trình bày ở trên về quá trình dịch chuyển lùi từng bước của thanh ghi:

Nếu ta gọi $x^{(0)}=\begin{pmatrix}x_0\\x_2\\x_3\\\vdots\\x_{m-1}\end{pmatrix}$ là véc tơ chỉ giá trị của thanh ghi tại thời điểm khởi tạo thì các giá trị của thanh ghi ở các thời điểm tiếp theo như sau:

Giá trị của thanh ghi sau 1 xung đồng hồ là $x^{(1)}=T.x^{(0)}$
Giá trị của thanh ghi sau 2 xung đồng hồ là $x^{(2)}=T.x^{(1)}=T^2.x^{(0)}$
Giá trị của thanh ghi sau 3 xung đồng hồ là $x^{(3)}=T.x^{(2)}=T^3.x^{(0)}$
---

Giá trị của thanh ghi sau n xung đồng hồ là $x^{(n)}=T.x^{(n-1)}=T^n.x^{(0)}$ (bởi vì số trạng thái thông tin khác nhau có thể có là $2^m$)

Vậy chu kỳ của thanh ghi là số xung nhịp đồng hồ để thanh ghi lặp lại trạng thái ban đầu. Nghĩa là nếu $x^{(0)}\neq 0$ và $\exists\ n>0$ sao cho $x^{(n)} = x^{(0)}$ thì ta nói n là chu kỳ của thanh ghi.

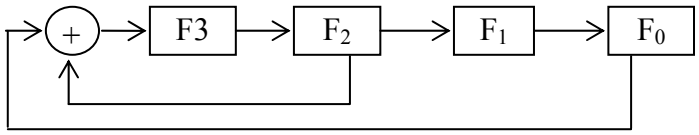
Lưu ý:
Cách viết biểu diễn nhị phân cho giá trị của $x^{(i)}$ theo thứ tự từ trên xuống (theo cột), tương ứng với viết từ trái sang phải (theo dòng). Ví dụ: biểu diễn nhị phân của $x^{(i)} = 3$ có m = 3 bit như sau:

Viết theo dòng: $x^{(i)} = 011$ (viết từ trái sang phải)

Viết theo cột: $x^{(i)} = \begin{pmatrix}0\\1\\1\end{pmatrix}$ (viết từ trên xuống)

## Ví dụ tìm chu kỳ của thanh ghi

Cho thanh ghi lui từng bước như hình sau:




Từ thanh ghi ta có: m=4, $a_0=1$, $a_1=0$, $a_2=1$, $a_3=0$.

Ma trận đặc trưng của thanh ghi: $T=\begin{bmatrix}0&1&0&0\\0&0&1&0\\0&0&0&1\\1&0&1&0\end{bmatrix}$

Đặc giá trị khởi tạo của thanh ghi $x^{(0)}=1=\begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix}=\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$

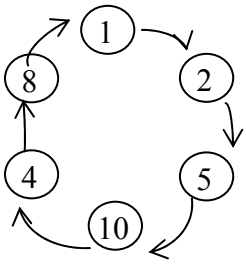
Tìm chu kỳ:

$$X^{(1)}=T.x^{(0)}=\begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \Rightarrow \quad x^{(2)}=T.x^{(1)}=\begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} \Rightarrow \quad x^{(3)}=T.x^{(2)}=\begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

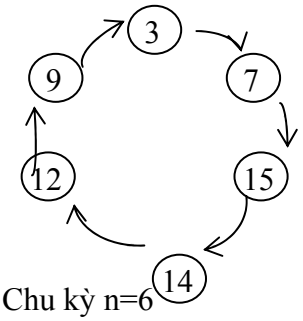
$$\Rightarrow \quad x^{(4)}=T.x^{(3)}=\begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \Rightarrow \quad x^{(5)}=T.x^{(4)}=\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \Rightarrow \quad x^{(6)}=T.x^{(5)}=\begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}=x^{(0)}$$

Tương tự:

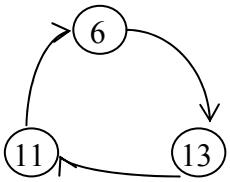
+ Khi chọn $x^{(0)} = 3$ thi ta cũng có chu kỳ n = 6.
+ Khi chọn $x^{(0)} = 6$ thì ta có chu kỳ n = 3.
+ Khi chọn $x^{(0)} = 0$ thì ta có chu kỳ n = 1.



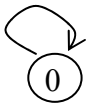

Chu kỳ n=6




Chu kỳ n=6




Chu kỳ n=3

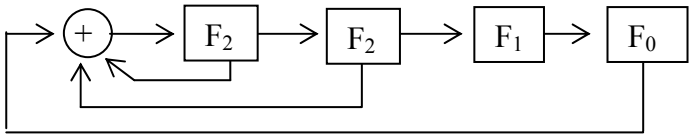



Chu kỳ n=1

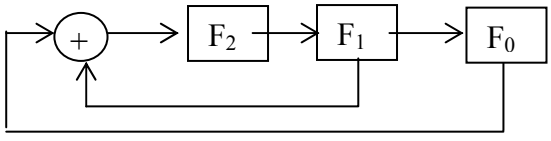
Thanh ghi trên có 4 chu kỳ.

### Bài tập

1. Tìm các chu kỳ của thanh ghi lui từng bước như hình sau:




2. Tìm các chu kỳ của thanh ghi lui từng bước như hình sau:




# BÀI 5.8: MÃ XOAY VÒNG

### Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:

- Biết cách xác định ma trận kiểm tra chẵn lẻ cho mã xoay vòng (hay còn gọi là mã vòng),
- Hiểu định nghĩa mã xoay vòng,
- Vận dụng xây dựng bộ mã xoay vòng,
- Vận dụng phương pháp sinh nhanh bộ mã xoay vòng để sinh bộ mã kiểm tra chẵn lẻ.

## Ma trận kiểm tra chẵn lẻ mã xoay vòng

**Định nghĩa:** ma trận kiểm tra chẵn lẻ được thiết kế từ thanh ghi lùi từng bước là ma trận có dạng sau:

$A=[x^{(0)}| T x^{(0)}|T^2 x^{(0)} |\ldots|T^{n-1} x^{(0)}]$ với n là chu kỳ của thanh ghi (n > m)

Trong đó:

- T là ma trận đặc trưng của thanh ghi.
- $x^{(0)} \neq 0$: là giá trị khởi tạo của thanh ghi.
- n : là chiều dài của từ mã và cũng là chu kỳ của thanh ghi.
- m: là số bit kiểm tra hay số bit của thanh ghi.

**Ví dụ:** xét lại ví dụ tìm chu kỳ thanh ghi, nếu chọn giá trị khởi tạo của thanh ghi là $x^{(0)} = 1$ thì ta có ma trận kiểm tra với chu kỳ n=6 như sau:

$$A = [\, x^{(0)} \;\; x^{(1)} \;\; x^{(2)} \;\; x^{(3)} \;\; x^{(4)} \;\; x^{(5)} \,] = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

## Định nghĩa mã xoay vòng

Mã xoay vòng là mã kiểm tra chẵn lẻ được sinh ra từ ma trận kiểm tra chẵn lẻ ứng với chu kỳ n của thanh ghi lùi từng bước có dạng như:

$$A=[x^{(0)}| Tx^{(0)}|T^2x^{(0)} |\ldots|T^{n-1}x^{(0)} ]$$

Ví dụ: xét lại ma trận kiểm tra chẵn lẻ ở trên

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \text{ (chu kỳ n = 6)}$$

Ta có n = 6, m = 3, k = 2 $\Rightarrow$ $s = 2^k = 2^2 = 4$ từ mã.

Áp dụng Phương pháp sinh mã nhanh bộ mã kiểm tra chẵn lẻ ta có bộ mã kiểm tra chẵn lẻ gồm 4 từ mã sau : $w_0 = 000000$, $w_1 = 101010$, $w_2 = 010101$, $w_4 = 111111$, đây chính là một trong các bộ mã xoay vòng sinh từ thanh ghi lùi từng bước nêu trên ***(Các bước sinh mã nhanh để nghị các bạn tự làm)***

## Phương pháp sinh nhanh bộ mã xoay vòng

Cách sinh nhanh k từ mã độc lập tuyến tính của bộ mã vòng từ $a_0, a_1, a_2, \ldots, a_{m-1}$:

**Bước 1:** sinh mã xoay vòng đầu tiên

Sinh mã xoay vòng đầu tiên có dạng $w_1=a_0a_1a_2\ldots a_{m-1}\,1000\ldots\underbrace{00}_{k-1 \text{ bit } 0}$

**Bước 2:** sinh k -1 từ mã độc lập tuyến tính còn lại

$w_2$= $0a_0a_1a_2\ldots a_{m-1}1\underbrace{000\ldots0}_{\text{k-2 bit 0}}$ (dịch $w_1$ sang phải 1 bit).

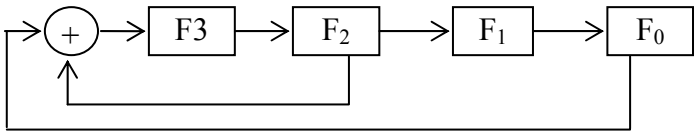
…….

$w_k$= $\underbrace{000\ldots00}_{\text{k-1 bit 0}}a_0a_1a_2\ldots a_{m-1}1$ (dịch từ $w_{k-1}$ sang phải 1 bit).

**Bước 3:** xác định các từ mã còn lại của bộ mã
Các từ mã còn lại gồm ($2^k$ – k từ mã) được xác định bằng cách cộng tổ hợp của 2, 3, …, k từ mã từ k từ mã độc lập tuyến tính ở trên.

## Ví dụ sinh nhanh bộ mã xoay vòng

Cho thanh ghi lui từng bước như hình sau:

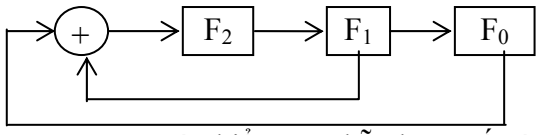



Từ thanh ghi, ta có: m=4, n=6, $a_0$=1, $a_1$=0, $a_2$=1, $a_3$=0.

**Bước 1:** Sinh mã xoay vòng đầu tiên
$w_1$=101010
**Bước 2:** Sinh k -1 từ mã độc lập tuyến tính còn lại
$w_2$=010101
**Bước 3:** Xác định các từ mã còn lại của bộ mã
$w_3$ =111111 ($w_1$+$w_2$), $w_0$ =000000 ($w_1$+$w_2$ + $w_3$)
Bộ mã vòng vừa sinh là W={000000, 101010, 010101, 111111)

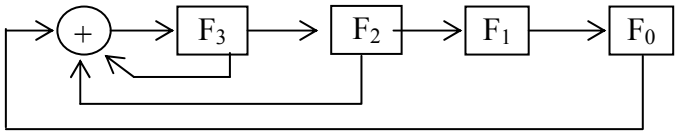
## Bài tập

1. Cho thanh ghi lùi từng bước sau:




   - Tìm ma trận kiểm tra chẵn lẻ có số cột n > 4
   - Từ kết quả câu a, xác định bộ mã xoay vòng tương ứng.
   - Tìm bộ mã xoay vòng theo phương pháp sinh nhanh bộ mã xoay vòng

2. Cho thanh ghi lùi từng bước sau:




   - Tìm ma trận kiểm tra chẵn lẻ có số cột n > 4
   - Từ kết quả câu a, xác định bộ mã xoay vòng tương ứng.
   - Tìm bộ mã xoay vòng theo phương pháp sinh nhanh bộ mã xoay vòng.

# BÀI 5.9: ĐA THỨC ĐẶC TRƯNG CỦA THANH GHI

## Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:

- Hiểu định nghĩa đa thức đặc trưng của thanh ghi,
- Hiểu Quan hệ giữa chu kỳ n, đa thức đặc trưng và đa thức $(x^n + 1)$,
- Vận dụng sinh thanh ghi lùi từng bước,
- Làm cơ sở để vận dụng sinh bộ mã vòng.

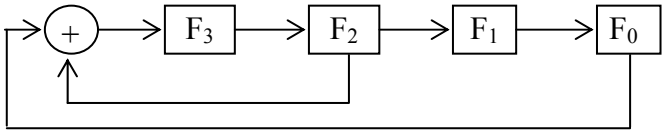
## Định nghĩa đa thức đặc trưng của thanh ghi

**Định nghĩa:** đa thức đặc trưng của thanh ghi có ma trận đặc trưng là T là đa thức có dạng

$g_m(x)=a_0 + a_1x+ a_2 x^2+ \ldots+a_{m-1}x^{m-1} + x^m$.

với $a_0, a_1, a_2,\ldots, a_{m-1}$ là các công tắc của thanh ghi và m là số bit của thanh ghi

**Ví dụ:** xét lại thanh ghi như hình sau:



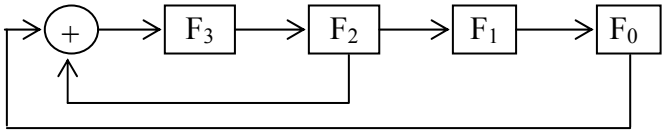

$a_0 = 1, a_1= 0, a_2 = 1, a_3 = 0$

Đa thức đặc trưng của thanh ghi có dạng: $g_m(x)=1 + x^2 + x^4$.

## Quan hệ giữa chu kỳ n, đa thức đăc trưng và đa thức $(x^n + 1)$

Đa thức đặc trưng của thanh ghi $g_m(x)=a_0 + a_1x+ a_2 x^2+ \ldots+a_{m-1}x^{m-1} + x^m$ luôn chia hết đa thức $(x^n + 1)$.

**Ví dụ:** xét lại thanh ghi lui từng bước như hình sau:



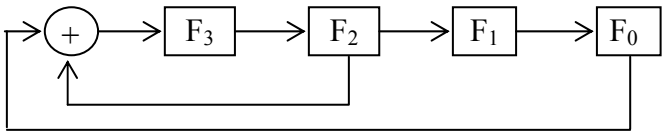

Từ thanh ghi ta có thể xác định các kết quả sau:

- $a_0 = 1, a_1= 0, a_2 = 1, a_3 = 0$
- Đa thức đặc trưng của thanh ghi có dạng: $g_4(x)=1 + x^2 + x^4$.
- Thanh ghi này có chu kỳ n = 6.

Thực hiện phép chia đa thức $(x^6 + 1) : (1 + x^2 + x^4) = (x^2 + 1) \Rightarrow$ chia hết.

**Ghi chú:** phép toán trên đa thức nhị phân vẫn là phép toán Modulo 2.

*Ví dụ*: xét lại thanh ghi lui từng bước như hình sau:




$a_0 = 1, a_1= 0, a_2 = 1, a_3 = 0$

đa thức đặc trưng của thanh ghi có dạng: $g_4(x)=1 + x^2 + x^4$.

thanh ghi này có chu kỳ n = 6 và $(x^6 + 1) : 1 + x^2 + x^4 = x^2 + 1$.

## Thủ tục sinh thanh ghi lùi từng bước

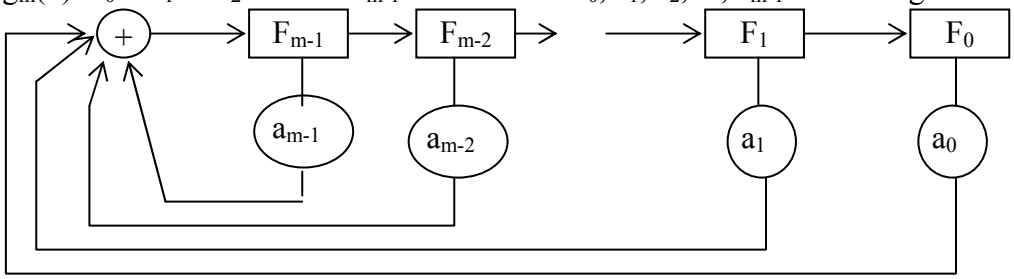
Để sinh thanh ghi lùi từng bước với số bit là m và có chu kỳ n, ta có thể thực hiện theo các bước sau:

**Bước 1:** xác định đa thức đặc trưng của thanh ghi

- Tìm 2 đa thức $g_m(x)=a_0 + a_1x+ a_2 x^2+ \ldots+a_{m-1}x^{m-1} + x^m$
  và $h_k(x)=h_0 + h_1x+ h_2x^2 + \ldots+h_{k-1}x^{k-1} + x^k$ sao cho $(x^n + 1) = g_m(x)* h_k(x)$.
- Nếu $\exists (x^n + 1) = g_m(x)* h_k(x)$ thì ta chọn $g_m(x)$ làm đa thức đặc trưng cho thanh ghi (vì số bit kiểm tra của bộ mã là m) và thực hiện bước 2.
- Ngược lại: không tồn tại thanh ghi theo yêu cầu.

**Bước 2:** vẽ thanh ghi

Từ $g_m(x)=a_0 + a_1x+ a_2 x^2+ \ldots+a_{m-1}x^{m-1} + x^m \Rightarrow a_0, a_1, a_2,\ldots, a_{m-1} \Rightarrow$ thanh ghi có dạng:




## Ví dụ minh họa

Thiết kế thanh ghi có m=3 bit và chu kỳ n=7, ta thực hiện theo 2 bước sau:

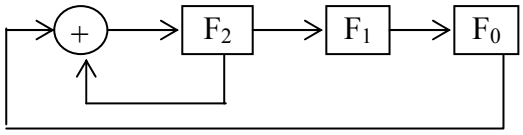
**Bước 1:** Xác định đa thức đặc trưng của thanh ghi

Ta có $(x^7 + 1) : (1 + x^2 + x^3) = (1 + x^2 + x^3 + x^4)$

Do m=3 nên chọn $g_3(x) = (1 + x^2 + x^3)$ làm đa thức đặc trưng của thanh ghi.

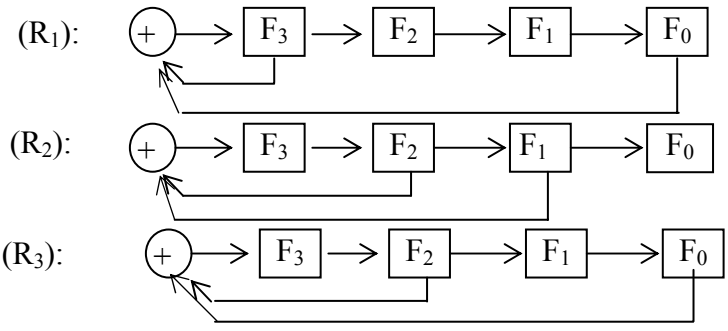
**Bước 2:** Vẽ thanh ghi

Từ $g_3(x) = (1 + x^2 + x^3)$ ta có, $a_0=1$, $a_1=0$, $a_2=1$




## Bài tập

1. Trong các thanh ghi sau đây, thanh ghi nào sinh ra bộ mã vòng có độ dài n=15 bit?




2. Nêu các bước cần thiết để thiết kế bộ mã xoay vòng độ dài 15 bit với số bit kiểm tra là 4. Vẽ sơ đồ thanh ghi dạng tổng quát.

# Bài 5.10: PHƯƠNG PHÁP SINH MÃ XOAY VÒNG

## Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:

- Hiểu các phương pháp sinh mã vòng,
- Biết bảng liệt kê một số đa thức đặc trưng,
- Vận dụng để sinh mã vòng theo nhiều cách khách nhau.

## Đặt vấn đề

Để sinh bộ mã kiểm tra chẵn lẻ, ta có thể dựa theo nhiều phương pháp khác nhau như: sinh mã dựa theo lý thuyết nhóm, mã Hamming,... Vấn đề đặt ra ở đây là làm sao để sinh bộ mã xoay vòng với độ dài n bit và m bit kiểm tra chẵn lẻ. Phương pháp sinh mã xoay vòng dựa trên lý thuyết về đa thức đặc trưng nhị phân của thanh ghi giúp ta có cái nhìn tổng quát về vấn đề sinh bộ mã xoay vòng theo nhiều cách khác nhau.

## Phương pháp sinh bảng mã xoay vòng

Để sinh mã xoay vòng độ dài n bit với m bit kiểm tra và k bit thông tin, ta có thể thực hiện theo các bước sau:

**Bước 1:** tìm 2 đa thức $g_m(x)=a_0 + a_1x+ a_2 x^2+ \ldots+a_{m-1}x^{m-1} + x^m$

và $h_k(x)=h_0 + h_1x+ h_2x^2 + \ldots+h_{k-1}x^{k-1} + x^k$ sao cho $(x^n + 1) = g_m(x)* h_k(x)$.

Nếu $\exists\ (x^n + 1) = g_m(x)* h_k(x)$ thì chuyển sang bước 2

Ngược lại không thể sinh bộ mã vòng theo yêu cầu.

**Bước 2:** ta có thể sinh bộ mã xoay vòng theo các cách như dưới đây:

**Cách 1:** Chọn đa thức $g_m(x)=a_0 + a_1x+ a_2 x^2+ \ldots+a_{m-1}x^{m-1} + x^m$

$\Rightarrow a_0, a_1, a_2,\ldots, a_{m-1}$

$\Rightarrow$ thanh ghi $\Rightarrow$ ma trận đặc trưng T

$\Rightarrow$ chu kỳ n $\Rightarrow$ ma trận kiểm tra chẵn lẻ A.

$\Rightarrow$ Bộ mã xoay vòng.

**Cách 2:** chọn đa thức $g_m(x)=a_0 + a_1x+ a_2 x^2+ \ldots+a_{m-1}x^{m-1} + x^m$

$\Rightarrow a_0, a_1, a_2,\ldots, a_{m-1}$

$\Rightarrow$ Sinh nhanh k từ mã độc lập tuyến tính với từ mã sinh độc lập tuyến tính đầu tiên có dạng: $w_1=a_0a_1a_2\ldots a_{m-1}1000\underbrace{\ldots 00}_{\text{k-1 bit 0}} \Rightarrow$ Bộ mã xoay vòng.

***Cách 3:*** chọn $h_k(x)=h_0 + h_1x+ h_2x^2 + \ldots+h_{k-1}x^{k-1} + x^k$ làm đa thức sinh ma trận kiểm tra chẵn lẻ cho bộ mã vòng có dạng:

$$\begin{pmatrix} 0 & 0 & - & - & - & 0 & 0 & 1 & h_{k-1} & - & - & - & h_1 & h_0 \\ 0 & - & - & - & 0 & 0 & 1 & h_{k-1} & - & - & - & h_1 & h_0 & 0 \\ - & - & - & - & - & - & - & - & - & - & - & - & - & - \\ 0 & 1 & k_{k-1} & - & - & - & h_1 & h_0 & 0 & 0 & - & - & - & 0 \\ 1 & h_{k-1} & - & - & - & h_1 & h_0 & 0 & 0 & - & - & - & 0 & 0 \end{pmatrix} \Big\updownarrow m$$

$\longleftrightarrow$ (m-1) bits

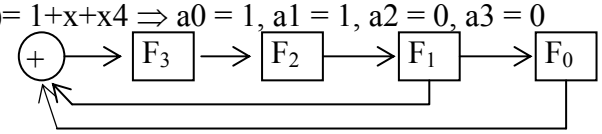
$\Rightarrow$ Sinh bộ mã xoay vòng theo Phương pháp sinh nhanh bộ mã xoay vòng.

**Nhận xét:** kết quả theo 3 cách sinh bộ mã xoay vòng nói trên la như nhau (cho cùng bộ mã).

## Ví dụ minh họa 1

Thiết kế thanh ghi và sinh ma trận kiểm tra chẵn lẻ.

Chọn đa thức $g_m(x)= 1+x+x4 \Rightarrow a0 = 1, a1 = 1, a2 = 0, a3 = 0$




Ma trận đặc trưng của thanh ghi: $T=\begin{bmatrix}0&1&0&0\\0&0&1&0\\0&0&0&1\\1&1&0&0\end{bmatrix}$

Tìm chu kỳ của thanh ghi:

Chọn giá trị khởi tạo $x^{(0)}=1=\begin{pmatrix}0\\0\\0\\1\end{pmatrix}$

$x^{(1)}=T.x^{(0)}=\begin{pmatrix}0\\0\\1\\0\end{pmatrix}$; $x^{(2)}=Tx^{(1)}=\begin{pmatrix}0\\1\\0\\0\end{pmatrix}$; $x^{(3)}=Tx^{(2)}=\begin{pmatrix}1\\0\\0\\1\end{pmatrix}$; $x^{(4)}=Tx^{(3)}=\begin{pmatrix}0\\0\\1\\1\end{pmatrix}$; $x^{(5)}=Tx^{(4)}=\begin{pmatrix}0\\1\\1\\0\end{pmatrix}$

$x^{(6)}=Tx^{(5)}=\begin{pmatrix}1\\1\\0\\1\end{pmatrix}$; $x^{(7)}=Tx^{(6)}=\begin{pmatrix}1\\0\\1\\0\end{pmatrix}$; $x^{(8)}=Tx^{(7)}=\begin{pmatrix}0\\1\\0\\1\end{pmatrix}$; $x^{(9)}=Tx^{(8)}=\begin{pmatrix}1\\0\\1\\1\end{pmatrix}$; $x^{(10)}=Tx^{(9)}=\begin{pmatrix}0\\1\\1\\1\end{pmatrix}$

$x^{(11)}=Tx^{(12)}=\begin{pmatrix}1\\1\\1\\1\end{pmatrix}$;$x^{(12)}=Tx^{(11)}=\begin{pmatrix}1\\1\\1\\0\end{pmatrix}$;$x^{(13)}=Tx^{(12)}=\begin{pmatrix}1\\1\\0\\0\end{pmatrix}$;$x^{(14)}=Tx^{(13)}=\begin{pmatrix}1\\0\\0\\0\end{pmatrix}$; $x^{(15)}=T.x^{(14)}=\begin{pmatrix}0\\0\\0\\1\end{pmatrix}=x^{(0)}$

Ma trận kiểm tra chẵn lẻ :

$$A=\begin{pmatrix}0&0&0&1&0&0&1&1&0&1&0&1&1&1&1\\0&0&1&0&0&1&1&0&1&0&1&1&1&1&0\\0&1&0&0&1&1&0&1&0&1&1&1&1&0&0\\1&0&0&1&1&0&1&0&1&1&1&1&0&0&0\end{pmatrix}$$

$\Rightarrow$ Bộ mã xoay vòng vớin=14, m=4, k=11.

## Ví dụ minh họa 2

Chọn đa thức gm(x)= 1+x+x4 $\Rightarrow$ a0 = 1, a1 = 1, a2 = 0, a3 = 0.

**Bước 1:** Sinh mã xoay vòng đầu tiên

w1 =110010000000000

**Bước 2:** Sinh k -1 từ mã độc lập tuyến tính còn lại

w2 =011001000000000
w3 =001100100000000
w4 =000110010000000
w5 =000011001000000
w6 =000001100100000
w7 =000000110010000
w8 =000000011001000
w9 =000000001100100
w10=000000000110010
w11=000000000011001

**Bước 3:** Xác định các từ mã còn lại của bộ mã

(215 - 11) từ mã còn lại được xác định bằng cách cộng tổ hợp 2, 3, 4,.., k = 11 từ mã từ k=11 từ mã độc lập tuyến tính.

## Ví dụ minh họa 3

Chọn $h_k(x)= 1+ x + x^2 + x^3 +x^5 + x^7 + x^8 + x^{11}$ làm đa thức sinh ma trận kiểm tra chẵn lẻ cho bộ mã vòng $\Rightarrow h_0 = 1, h_1 = 1, h_2 = 1, h_3 = 1, h_4 = 0, h_5 = 1, h_6 = 0, h_7 = 1, h_8 =1, h_9 = 0, h_{10} = 0.$

$$A=\begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix} \Rightarrow \text{Bộ mã xoay vòng}$$

## Bảng liệt kê một số đa thức đặc trưng

| M | Đa thức | M | Đa thức |
|---|---|---|---|
| 3 | $1+x+x^3$ | 14 | $1+x+x^6+x^{10}+x^{14}$ |
| 4 | $1+x+x^4$ | 15 | $1+x+x^{15}$ |
| 5 | $1+x^2+x^5$ | 16 | $1+x+x^3+x^{12}+x^{16}$ |
| 6 | $1+x+x^6$ | 17 | $1+x^3+x^7$ |
| 7 | $1+x^3+x^7$ | 18 | $1+x^7+x^{18}$ |
| 8 | $1+x^2+x^3+x^4+x^8$ | 19 | $1+x+x^2+x^5+x^{19}$ |
| 9 | $1+x^4+x^9$ | 20 | $1+x^3+x^{20}$ |
| 10 | $1+x^3+x^{10}$ | 21 | $1+x^2+x^{21}$ |
| 11 | $1+x^2+x^{11}$ | 22 | $1+x+x^{22}$ |
| 12 | $1+x+x^4+x^6+x^{12}$ | 23 | $1+x^3+x^{23}$ |
| 13 | $1+x+x^3+x^4+x^{13}$ | 24 | $1+x+x^2+x^7+x^{24}$ |

## Bài tập

1. Tìm bộ mã vòng có độ dài 7 bit.
2. Tìm thanh ghi sinh bộ mã vòng có độ dài 15 bit.
3. Tìm thanh ghi sinh bộ mã vòng có độ dài 31 bit.

# BÀI TẬP TỔNG HỢP

## Mục tiêu

Sau khi hoàn tất bài học này bạn có thể:

- Hiểu rõ hơn về nội dung môn học.
- Vận dụng nội dung môn học để giải quyết một số bài tập tổng hợp.

## Bài 1

Xét một mô hình chẩn đoán bệnh từ các triệu chứng: A, B và C; để chẩn đoán 1 trong 4 bệnh: 1, 2, 3 và 4 với ma trận chẩn đoán (hay ma trận truyền tin).

| Bệnh / Triệu chứng | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| A | 0,6 | 0,3 | 0 | 0,1 |
| B | 0,2 | 0,6 | 0,2 | 0 |
| C | 0 | 0 | 0,3 | 0,7 |

**Yêu cầu:**

Câu 1: Vẽ *sơ đồ mô tả mô hình chẩn đoán bệnh* trên và *diễn giải các ý nghĩa* của sơ đồ.

Câu 2: Nếu phân phối của Triệu chứng có dạng:

| Triệu chứng | A | B | C |
|---|---|---|---|
| P | 0,5 | 0,3 | 0,2 |

Tính các lượng sau :

- Lượng ngẫu nhiên (Entropy) của Triệu chứng .
- Lượng ngẫu nhiên của Bệnh.
- Lượng ngẫu nhiên của Bệnh khi biết Triệu chứng.
- Lượng chẩn đoán đúng.(Lượng thông tin biết về Bệnh thông qua Triệu chứng) và tỷ lệ chẩn đoán đúng là bao nhiêu phần trăm.

Câu 3: Bây giờ người ta sử dụng 2 bit để mã thông tin về Triệu chứng (có 1 triệu chứng dự trữ) và 5 bit để mã các triệu chứng khi chẩn đoán bệnh trực tuyến. *Mô tả các đoạn* của dãy 5 bit trong phương pháp kiểm tra chẵn lẻ.

Câu 4: Nếu sử dụng ma trận kiểm tra chẵn lẻ dạng:

$$A = \begin{array}{|c|c|c|c|c|} \hline 1 & 1 & 1 & 0 & 1 \\ \hline 0 & 1 & 0 & 1 & 1 \\ \hline 1 & 0 & 0 & 1 & 1 \\ \hline \end{array}$$

*Tính các từ mã*.

*Xây dựng Bộ sửa lỗi* 1 bit dùng cho tự động sửa lỗi tối ưu trong quá trình chẩn đoán trực tuyến. Cho *một ví dụ*.

## Bài 2

Xét một kênh truyền tin đặc biệt dạng : Truyền X → Nhận Y.

Truyền một giá trị của X có thể nhận được nhiều giá trị khác nhau của Y với các xác suất khác nhau. Bảng xác suất truyền X và nhận các Y khác nhau được cho dưới đây:

| Y / X | $y_1$ | $y_2$ | $y_3$ | $y_4$ | $y_5$ | $y_6$ |
|---|---|---|---|---|---|---|
| $x_0$ | 0,6 | 0,1 | 0,1 | 0,05 | 0,05 | 0,1 |
| $x_1$ | 0,1 | 0,05 | 0,6 | 0,1 | 0,1 | 0,05 |
| $x_2$ | 0,05 | 0,1 | 0,1 | 0,05 | 0,6 | 0,1 |
| $x_3$ | 0,1 | 0,05 | 0,05 | 0,1 | 0,1 | 0,6 |

**Yêu cầu:**
Câu 1: Vẽ *sơ đồ mô tả kênh truyền tin* trên và *diễn giải các ý nghĩa* của sơ đồ.
Câu 2: Nếu phân phối của X có dạng :

| X | $x_0$ | $x_1$ | $x_3$ | $x_4$ |
|---|---|---|---|---|
| P | 0.5 | 0.25 | 0.15 | 0.1 |

tính *thông lượng về X truyền trên kênh*.
Câu 3: Phân phối của X cần có dạng như thế nào để thông lượng truyền trên kênh là lớn nhất. Tính dung lượng kênh truyền.
Câu 4: Bây giờ người ta sử dụng 2 bit để mã thông tin về X và 4 bit để mã các giá trị truyền trên kênh. *Mô tả các đoạn* của dãy 4 bit trong phương pháp kiểm tra chẵn lẻ.
Câu 5: Nếu sử dụng ma trận kiểm tra chẵn lẻ dạng:

$$A = \begin{array}{|c|c|c|c|} \hline 1 & 1 & 1 & 0 \\ \hline 0 & 1 & 0 & 1 \\ \hline \end{array}$$

*Tính các từ mã*.
*Xây dựng Bộ sửa lỗi* dùng cho tự động sửa lỗi tối ưu trong quá trình truyền tin. Cho *một ví dụ*.

## Bài 3

Người ta cần đánh giá kênh truyền tin và chuẩn bị thực hiện truyền một loại tín hiệu đặc biệt: X = $\{x_0, x_1, x_2, x_3\}$
Công việc đầu tiên là phải khảo sát kênh truyền. Kết quả khảo sát cho thấy:
*Kênh có thể truyền nhận được 8 giá trị khác nhau, để có khả năng phát hiện lỗi hoặc điều chỉnh lỗi.* Ma trận truyền tin có dạng:

| Y / X | $y_1$ | $y_2$ | $y_3$ | $y_4$ | $y_5$ | $y_6$ | $y_7$ | $y_8$ |
|---|---|---|---|---|---|---|---|---|
| $x_0$ | 0,6 | 0,1 | 0,05 | 0,05 | 0,05 | 0,05 | 0,05 | 0,05 |
| $x_1$ | 0,05 | 0,05 | 0,6 | 0,1 | 0,05 | 0,05 | 0,05 | 0,05 |
| $x_2$ | 0,05 | 0,05 | 0,05 | 0,05 | 0,6 | 0,1 | 0,05 | 0,05 |
| $x_3$ | 0,05 | 0,05 | 0,05 | 0,05 | 0,05 | 0,05 | 0,6 | 0,1 |

Yêu cầu:
Câu 1: Vẽ *sơ đồ mô tả kênh truyền tin* trên và *diễn giải các ý nghĩa* của sơ đồ. Nếu phân phối của X có dạng :

| X | $x_0$ | $x_1$ | $x_3$ | $x_4$ |
|---|---|---|---|---|
| P | 0.5 | 0.25 | 0.15 | 0.1 |

tính *thông lượng về X truyền trên kênh*.

Câu 2: Phân lớp các giá trị của Y về các lớp $B_0$, $B_1$, $B_2$, và $B_3$ dùng để giải mã tối ưu Y tốt nhất về các giá trị tương ứng của X.
Câu 3 : Bây giờ người ta sử dụng 2 bit để mã thông tin về X và 4 bit để mã các giá trị truyền trên kênh. Mô tả các đoạn của dãy 4 bit trong phương pháp kiểm tra chẵn lẻ.
Câu 4: Nếu sử dụng ma trận kiểm tra chẵn lẻ dạng:

$$A = \begin{array}{|c|c|c|c|} \hline 1 & 0 & 0 & 1 \\ \hline 0 & 1 & 1 & 1 \\ \hline \end{array}$$

*Tính các từ mã*.
*Xây dựng Bộ sửa lỗi* dùng cho tự động sửa lỗi tối ưu trong quá trình truyền tin. Cho *một ví dụ*.

## Bài 4

Xét một mô hình chẩn đoán bệnh từ các triệu chứng: A, B và C; để chẩn đoán 1 trong 4 bệnh: 1, 2, 3 và 4 với ma trận chẩn đoán (hay ma trận truyền tin)

| Bệnh / Triệu chứng | 1 | 2 | 3 | 4 |
|---|---|---|---|---|
| A | 0,5 | 0,3 | 0 | 0,2 |
| B | 0,1 | 0,2 | 0,7 | 0 |
| C | 0 | 0,1 | 0,3 | 0,6 |

## Yêu cầu:

Câu 1: Giả sử người ta biết thêm 3 triệu chứng gây bệnh khác đó là : D, E và F và muốn ghi lại các triệu chứng này thông qua bảng ký hiệu A = {+, - }.
Hãy kiểm tra tính tách được của bảng mã sau :

| Triệu chứng : X | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| Mã : W | + | -+ | ++- | --+- | ++-+ | -- |

Câu 2: Nếu các triệu chứng ở câu 1 có phân phối :

| Triệu chứng : X | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| P | 0.5 | 0.2 | 0.2 | 0.05 | 0.03 | 0.2 |

Giử sử có một người bệnh với 1 trong 5 triệu chứng trên đến khám bệnh và bác sĩ sẽ hỏi bệnh với nguyên tắc, sao cho người bệnh chỉ trả lời bằng 2 câu : Đúng hoặc Sai.

- Tìm phương pháp hỏi bệnh với số câu hỏi trung bình ít nhất.
- Tính số câu hỏi trung bình.
- Tính lượng ngẫu nhiên của Triệu chứng.
- Nhận xét gì về số câu hỏi trung bình và lượng ngẫu nhiên của triệu chứng.

Câu 3: Bây giờ sử dụng mô hình 3 triệu chứng {A, B, C} và 4 bệnh. Vẽ *sơ đồ mô tả mô hình chẩn đoán bệnh* và *diễn giải các ý nghĩa* của sơ đồ.

Câu 4: Từ kết quả câu 3, người ta sử dụng 2 bit để mã thông tin về Triệu chứng (có 1 triệu chứng dự trữ) và 5 bit để mã các triệu chứng khi chẩn đoán bệnh trực tuyến. *Mô tả các đoạn* của dãy 5 bit trong phương pháp kiểm tra chẵn lẻ.
Câu 5: Nếu sử dụng ma trận kiểm tra chẵn lẻ dạng:

$$A = \begin{array}{|c|c|c|c|c|} \hline 1 & 1 & 1 & 0 & 1 \\ \hline 0 & 1 & 0 & 1 & 1 \\ \hline 1 & 0 & 0 & 1 & 1 \\ \hline \end{array}$$

Tính các từ mã.

# TÀI LIỆU THAM KHẢO

12. G.J.ChaiTin, *Algorithmic Information Theory*, CamBridge University Express-1992.
13. David J.C. Mackey, *Information Theory, Infernce, and Learning Algorithms,* CamBridge University Express-2003.
14. Sanford Goldman, *Information Theory.*
15. http://www.inference.phy.cam.ac.uk/mackay/info-theory/course.html.
16. http://en.wikipedia.org/wiki/Information_theory.
17. http://www-2.cs.cmu.edu/~dst/Tutorials/Info-Theory/.
18. http://cscs.umich.edu/~crshalizi/notebooks/information-theory.html.
19. http://www.lecb.ncifcrf.gov/~toms/paper/primer/primer.pdf.
20. http://www.cs.ucl.ac.uk/staff/S.Bhatti/D51-notes/node27.html.
21. http://guest.engelschall.com/~sb/hamming/.
22. http://www2.rad.com/networks/1994/err_con/hamming.htm.