TRƯỜNG ĐẠI HỌC SÂN KHẤU – ĐIỆN ẢNH HÀ NỘI
KHOA CÔNG NGHỆ ĐIỆN ẢNH – TRUYỀN HÌNH

KỸ THUẬT AUDIO – VIDEO

CHƯƠNG 4

Cơ sở xử lý tín hiệu số

Fourier · STFT · Bộ lọc · Tích chập

5 tiết · Dành cho sinh viên Đạo diễn âm thanh, Quay phim, Đạo diễn điện ảnh – truyền hình, Dựng phim

Hà Nội – 2026

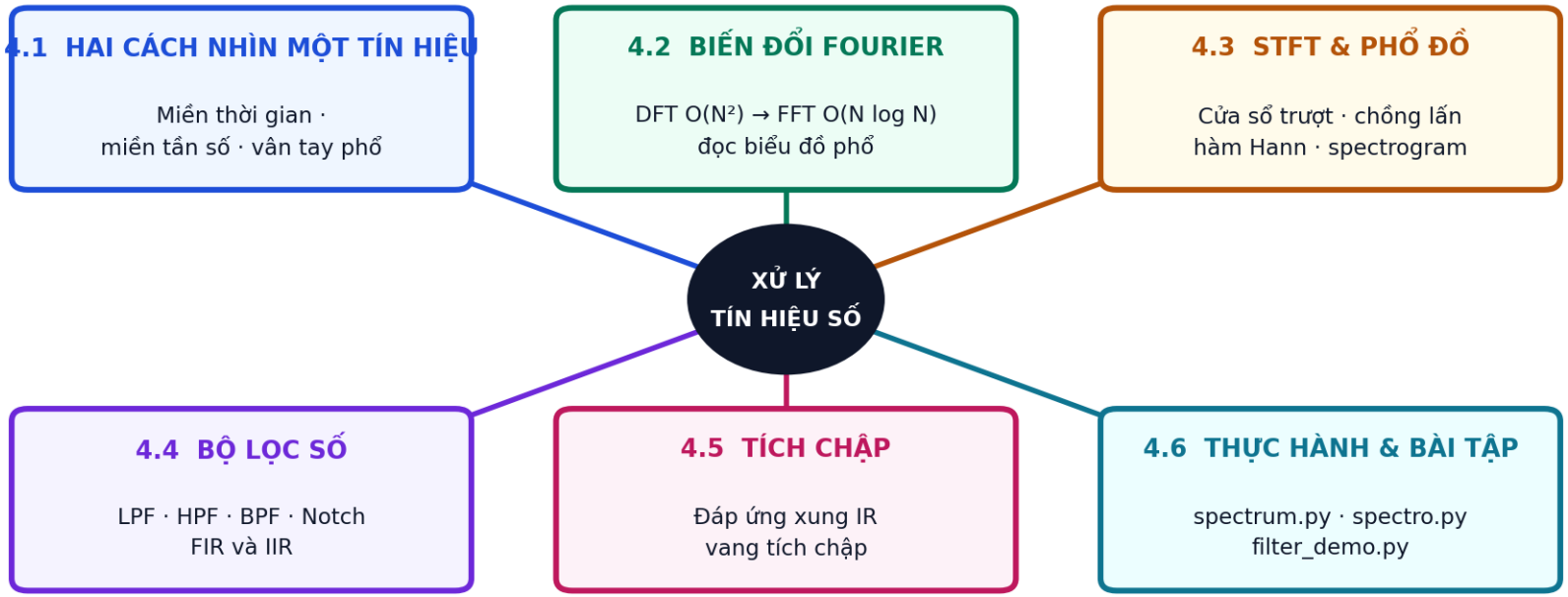
Ba chương trước cho ta dữ liệu, chương này cho ta công cụ

Chương 3 kết thúc bằng một `numpy.ndarray` trong bộ nhớ. Từ đây trở đi ta nghiên cứu cách **quan sát, giải nghĩa và can thiệp** trực tiếp vào mảng dữ liệu đó.

- **Chương 2:** sóng âm → điện áp. **Chương 3:** điện áp → dãy số. **Chương 4:** dãy số → hiểu và biến đổi được.
- Điểm mấu chốt của cả chương: cùng một tín hiệu có **hai cách nhìn bổ sung lẫn nhau** — như chụp một bức tượng từ hai góc để có hình dung toàn diện.
- Bốn công cụ sẽ học: **Fourier/FFT** (phân rã phổ), **STFT** (phổ theo thời gian), **bộ lọc số** (chỉnh âm sắc), **tích chập** (in đặc tính không gian).
- Mọi plugin EQ, reverb, denoise, pitch-shift trong DAW đều là ứng dụng của đúng bốn công cụ này. Không có công cụ thứ năm.

BẢN ĐỒ CHƯƠNG

Sáu mảng kiến thức của Chương 4



4.1 đặt vấn đề, 4.2 và 4.3 là công cụ quan sát, 4.4 và 4.5 là công cụ can thiệp, 4.6 là thực hành kiểm chứng.

TIẾT 1

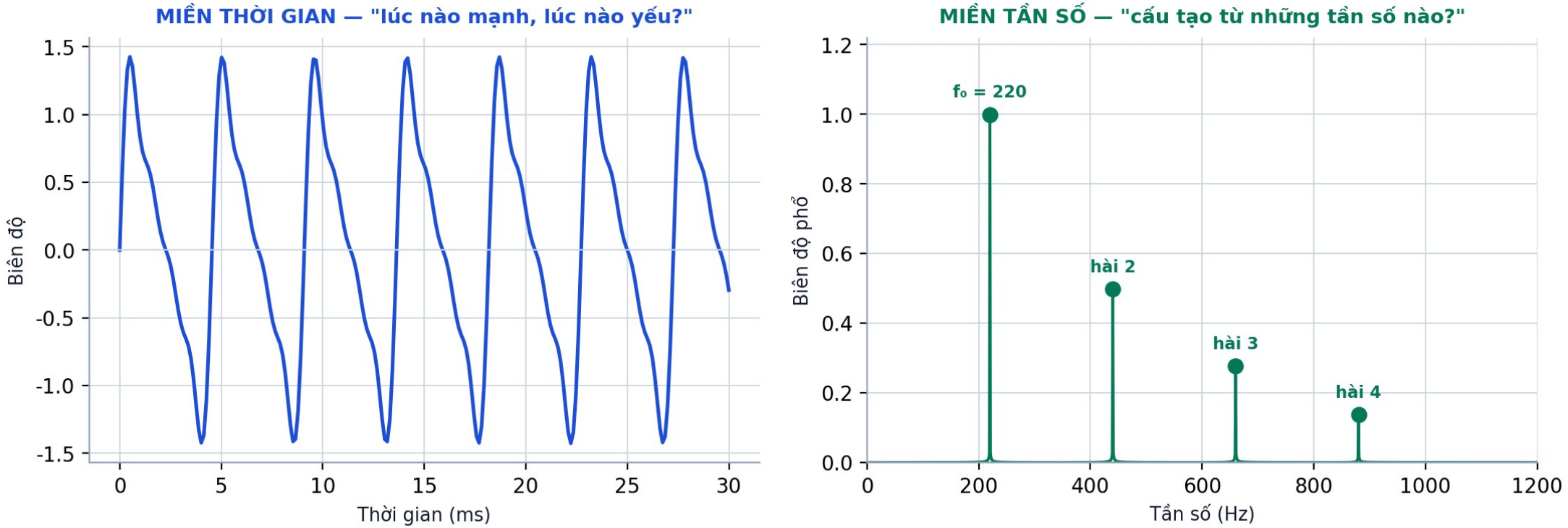
Hai cách nhìn một tín hiệu: thời gian và tần số

- Miền thời gian: "lúc nào mạnh, lúc nào yếu?"
- Miền tần số: "cấu tạo từ những tần số nào?"
- Ẩn dụ lăng kính tách ánh sáng trắng
- Vân tay phổ của nhạc cụ, giọng nói và tiếng động



MỤC 4.1.1

Cùng một tín hiệu — hai cách nhìn



Cùng MỘT tín hiệu — hai cách nhìn bổ sung nhau, như chụp một bức tượng từ hai góc

Trái: biên độ biến thiên theo thời gian. Phải: cùng tín hiệu ấy phân rã thành f_o = 220 Hz và ba hài âm.

ẨN DỤ SƯ PHẠM

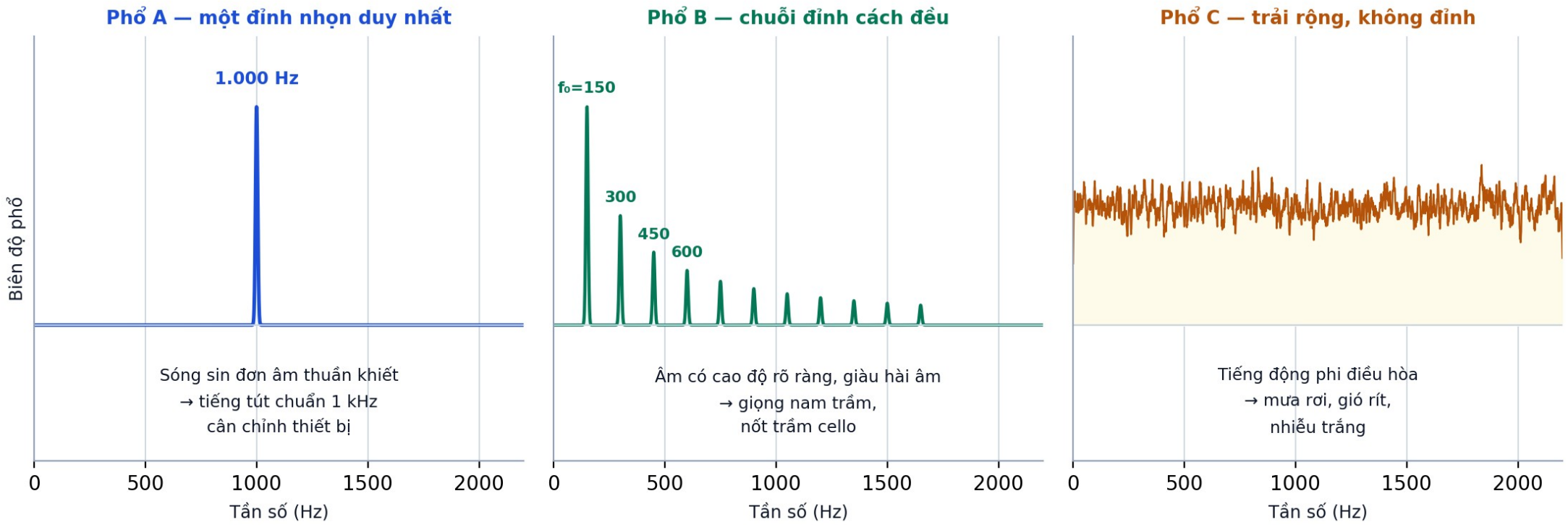
Lăng kính tách ánh sáng trắng

Một tia sáng trắng chiếu qua khối lăng kính bị khúc xạ và tách thành dải cầu vồng bảy sắc. Mỗi màu đơn sắc ứng với một bước sóng — một "tần số" ánh sáng cụ thể.

- Phép biến đổi sang miền tần số hoạt động y hệt: nó là một **chiếc lăng kính toán học**.
- Nó phân rã một tín hiệu âm thanh phức tạp thành các **sóng sin đơn tần** cấu thành.
- Khi tai nghe một hợp âm piano, ta nghe **một khối âm thanh hài hòa thống nhất**.
- Nhưng nhìn ở miền tần số, ta thấy rõ hợp âm đó gồm **chính xác những nốt nào** — Đồ, Mi, Sol — hiện thành các vạch năng lượng nhọn hoắt.

MỤC 4.1.2

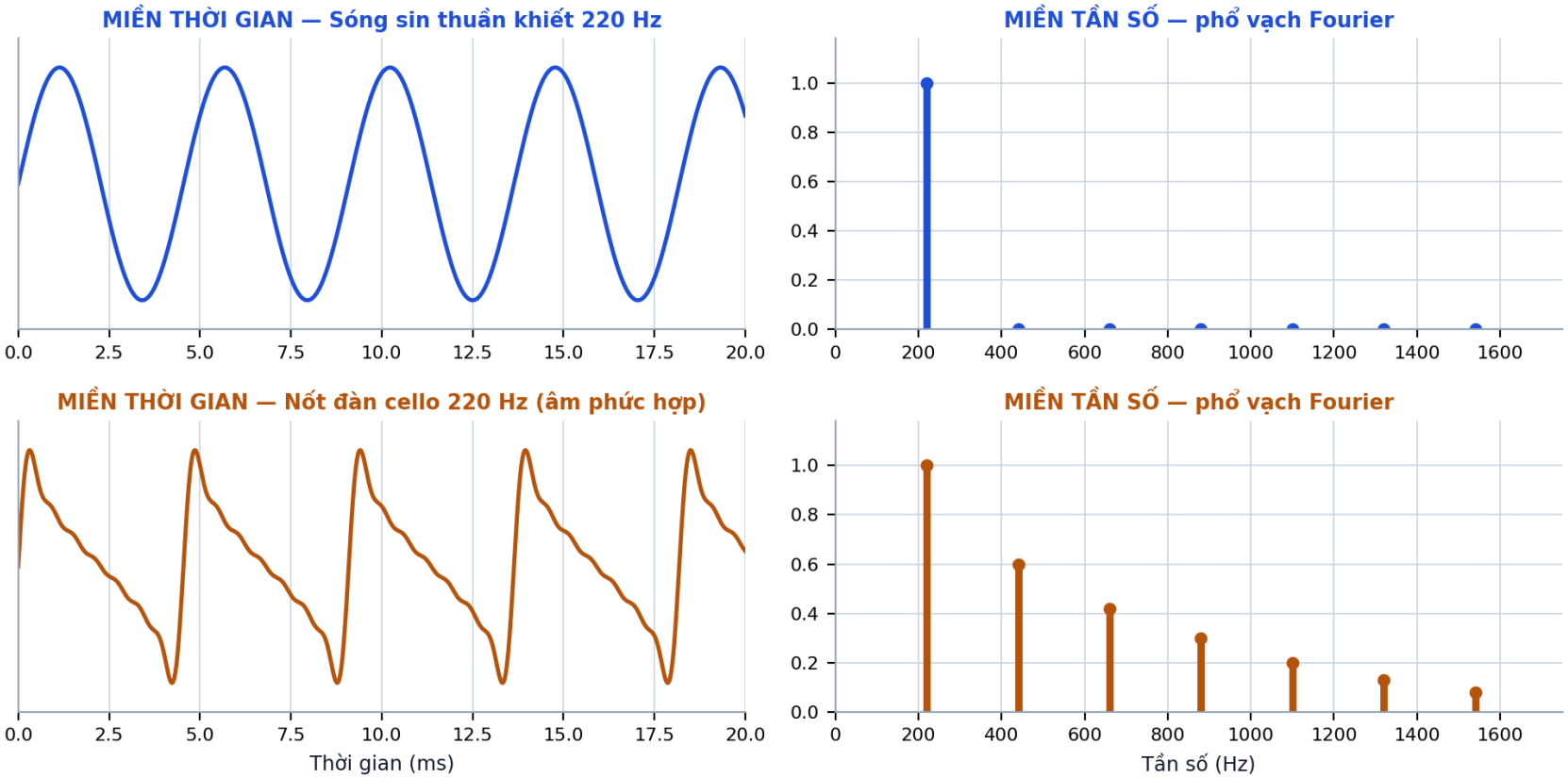
Vân tay phổ — ba dạng đồ thị cần thuộc



Nhìn hình dạng phổ là nhận ra loại nguồn âm, không cần bật tai nghe.

HÌNH 4.1

Hai cách nhìn trong giáo trình



Hình 4.1 — Dạng sóng theo thời gian và phổ vạch hài Fourier theo tần số.

CÂU HỎI TẠI LỚP

Một kỹ sư nhìn biểu đồ phổ thấy các đỉnh nhọn tại 150, 300, 450, 600, 750 Hz. Đây là nguồn âm gì?

A Năm nhạc cụ khác nhau chơi cùng lúc

B Một âm có cao độ với f_0 = 150 Hz và chuỗi hài âm

C Tiếng nhiễu trắng

D Năm nốt nhạc rời rạc của một hợp âm

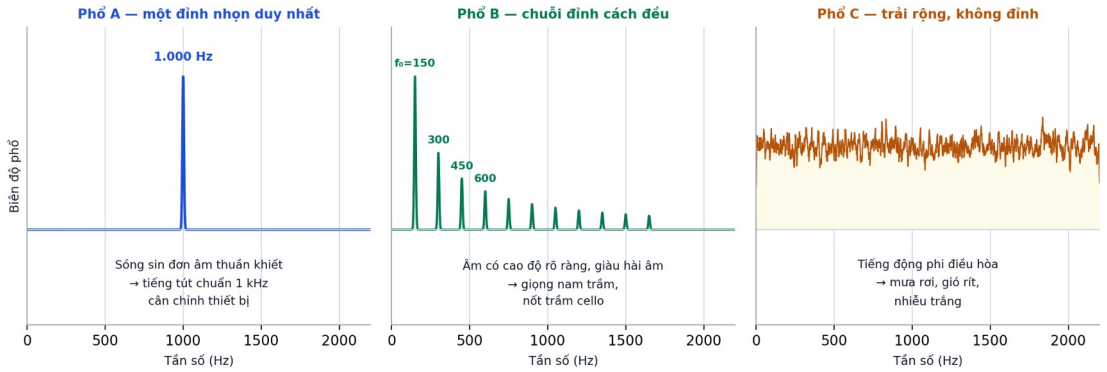
Suy nghĩ 30 giây — rồi trả lời

GIẢI THÍCH

Đáp án B — một âm có cao độ f_0 = 150 Hz

Các đỉnh cách đều nhau đúng 150 Hz và là bội số nguyên của đỉnh đầu — dấu hiệu chắc chắn của chuỗi hài âm từ MỘT nguồn.

- 150 Hz là **tần số cơ bản f_0** — quyết định cao độ nốt nhạc mà tai nghe được.
- 300 = $2f_0$, 450 = $3f_0$, 600 = $4f_0$, 750 = $5f_0$ — **chuỗi hài âm toàn phần**.
- Nếu là nhiều nguồn độc lập, các đỉnh sẽ **không** cách đều theo bội số nguyên như vậy.
- f_0 = 150 Hz ứng với **giọng nam trầm ấm** hoặc **nốt trầm của đàn cello**. Đây chính là đáp án phần B bài tập 1 của chương.



CHỐT LẠI MỤC 4.1

Chốt lại: hai miền quan sát

Miền thời gian — biên độ theo thời gian. Trả lời: *lúc nào mạnh, lúc nào yếu?*

Miền tần số — năng lượng theo tần số. Trả lời: *cấu tạo từ những tần số nào?*

Fourier là **lăng kính toán học** — bộc lộ tần số vốn đã có, không tạo thêm.

Phổ A: một đỉnh → đơn âm thuần. **Phổ B:** chuỗi đỉnh cách đều → âm có cao độ.

Phổ C: trải rộng không đỉnh → tiếng động phi điều hòa, nhiễu.

Nhìn phổ là **nhận ra nguồn âm**, không cần bật tai nghe.

TIẾT 2

Biến đổi Fourier và thuật toán FFT

- Ý tưởng Fourier: mọi tín hiệu là tổng các sóng sin
- DFT $O(N^2)$ và bước nhảy vọt của FFT $O(N \log N)$
- Đọc hiểu hai trục của biểu đồ phổ
- Quy luật đánh đổi độ phân giải thời gian – tần số



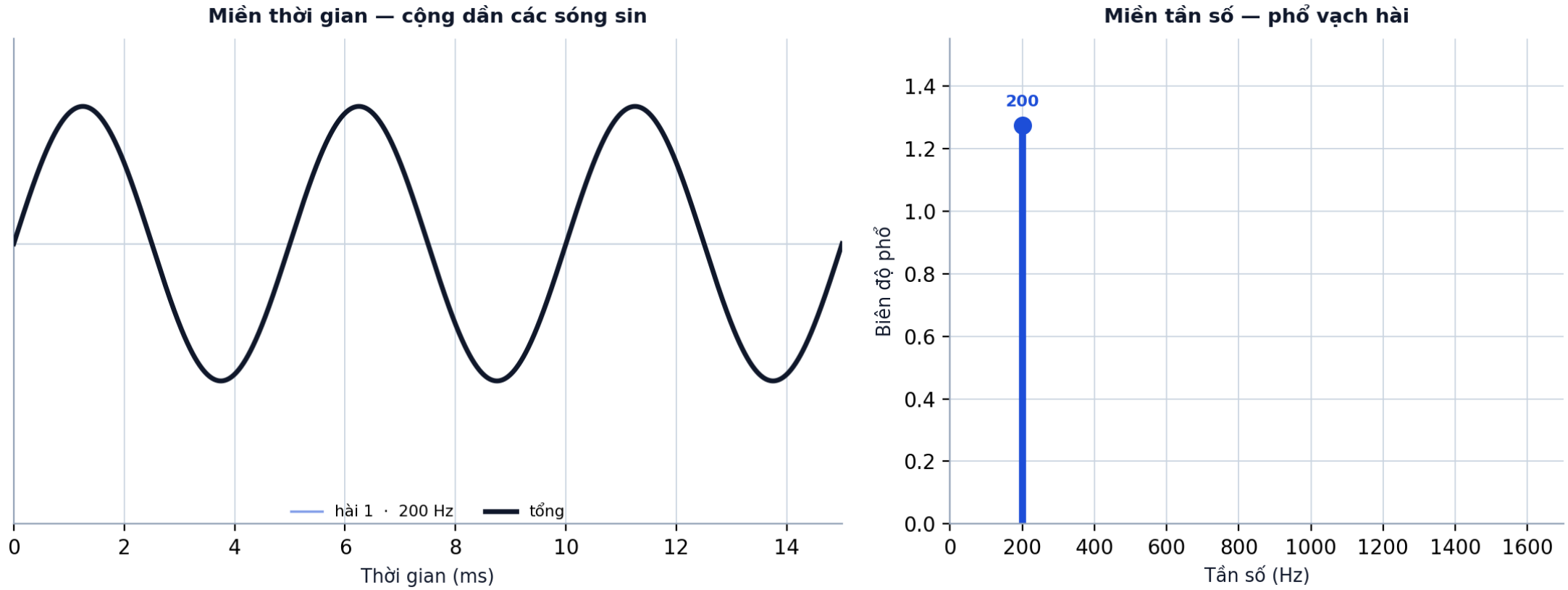
MỤC 4.2.1

Ý tưởng triết học của Joseph Fourier

Nền tảng toán học đứng sau việc chuyển tín hiệu giữa hai miền là **phép Biến đổi Fourier** — xây trên một ý tưởng vô cùng đẹp đẽ của nhà toán học Jean-Baptiste Joseph Fourier.

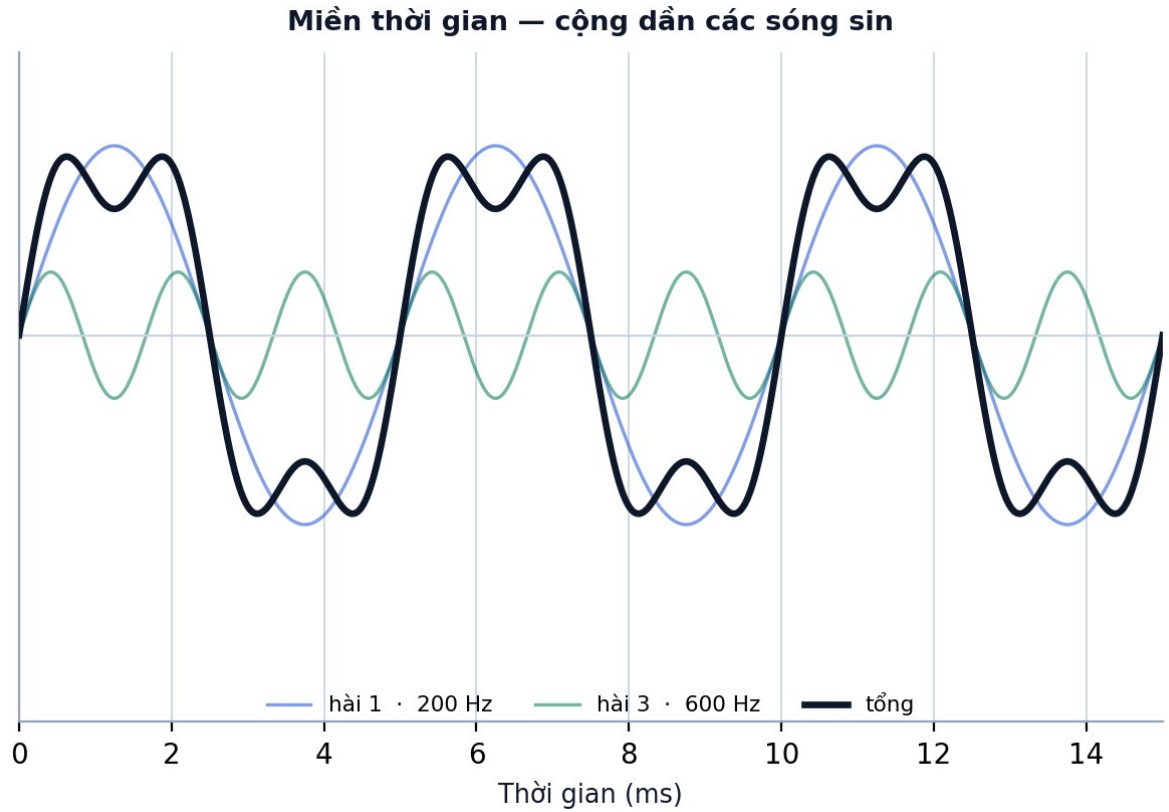
- **Mọi tín hiệu trong tự nhiên** — dù phức tạp, kỳ dị hay hỗn loạn đến đâu — đều có thể phân tích thành **tổng của nhiều sóng sin đơn tần thuần khiết**.
- Mỗi sóng sin thành phần có **tần số, biên độ và góc pha** hoàn toàn xác định.
- Biến đổi Fourier là công cụ **tìm ra công thức pha trộn** ẩn giấu bên trong tín hiệu.
- Nó tính chính xác: cần những sóng sin nào, tỷ lệ biên độ bao nhiêu, để khi cộng lại thu được **đúng tín hiệu gốc**.

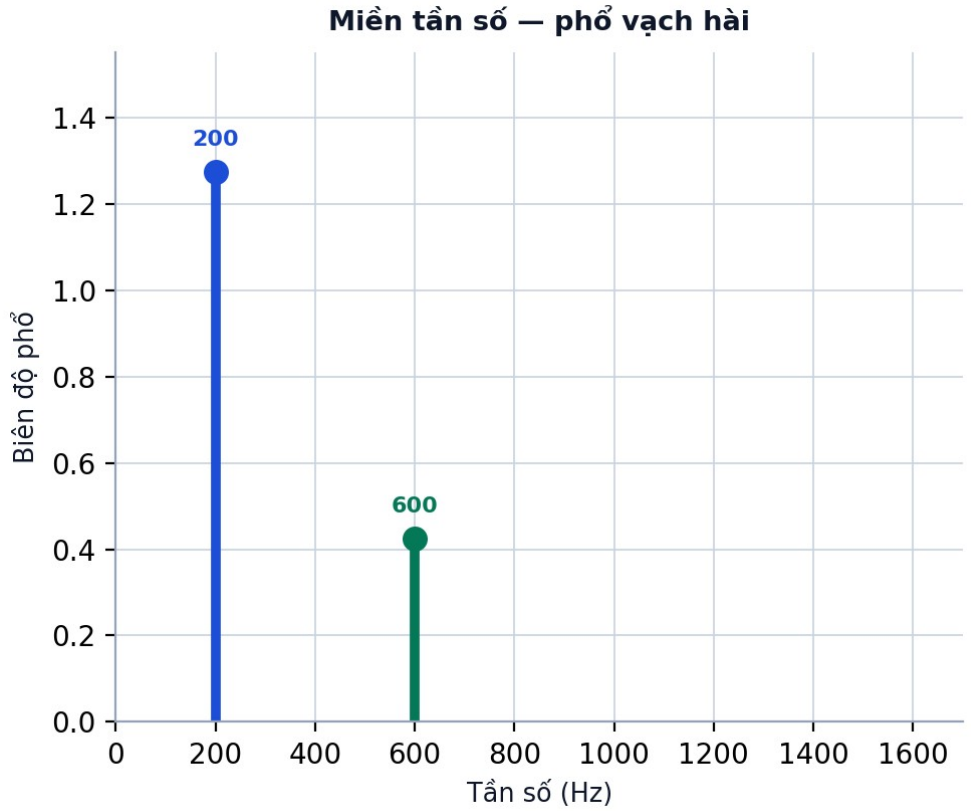
Bắt đầu từ một sóng sin duy nhất



Hài cơ bản 200 Hz. Bên phải: phổ chỉ có một vạch tại 200 Hz.

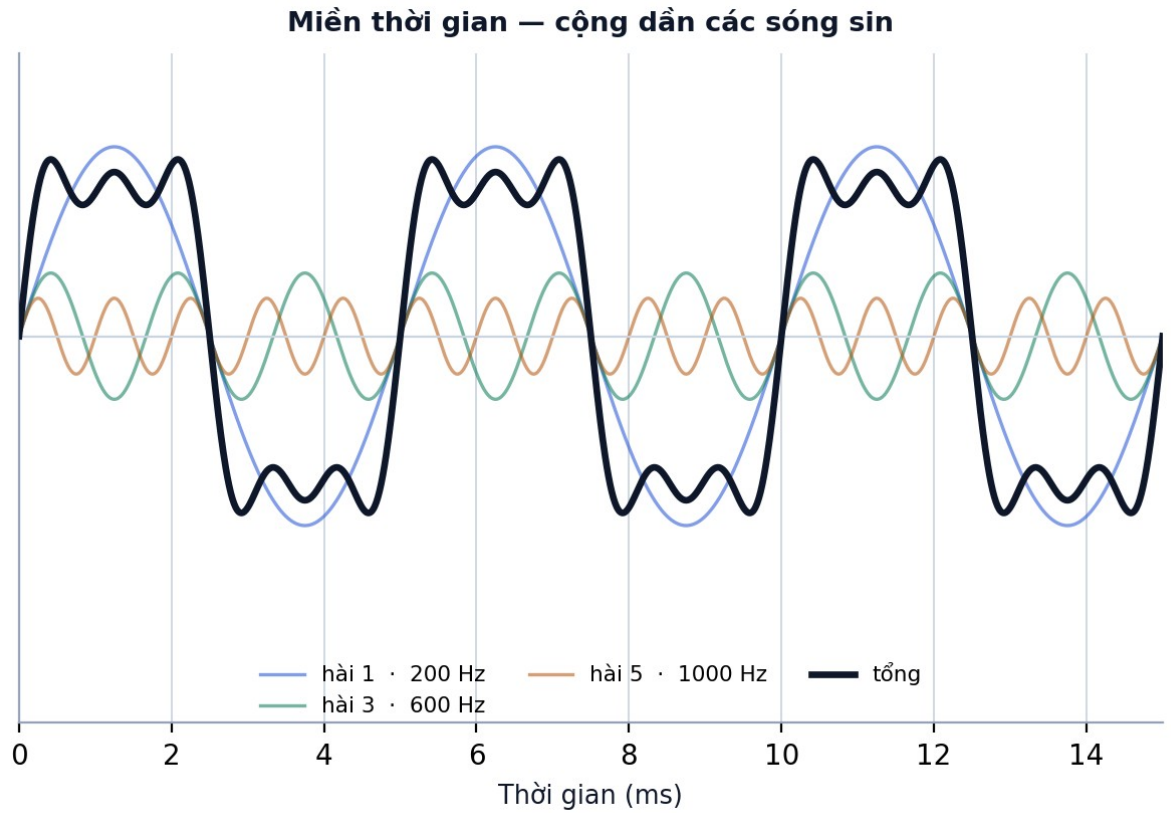
Cộng thêm hài bậc 3

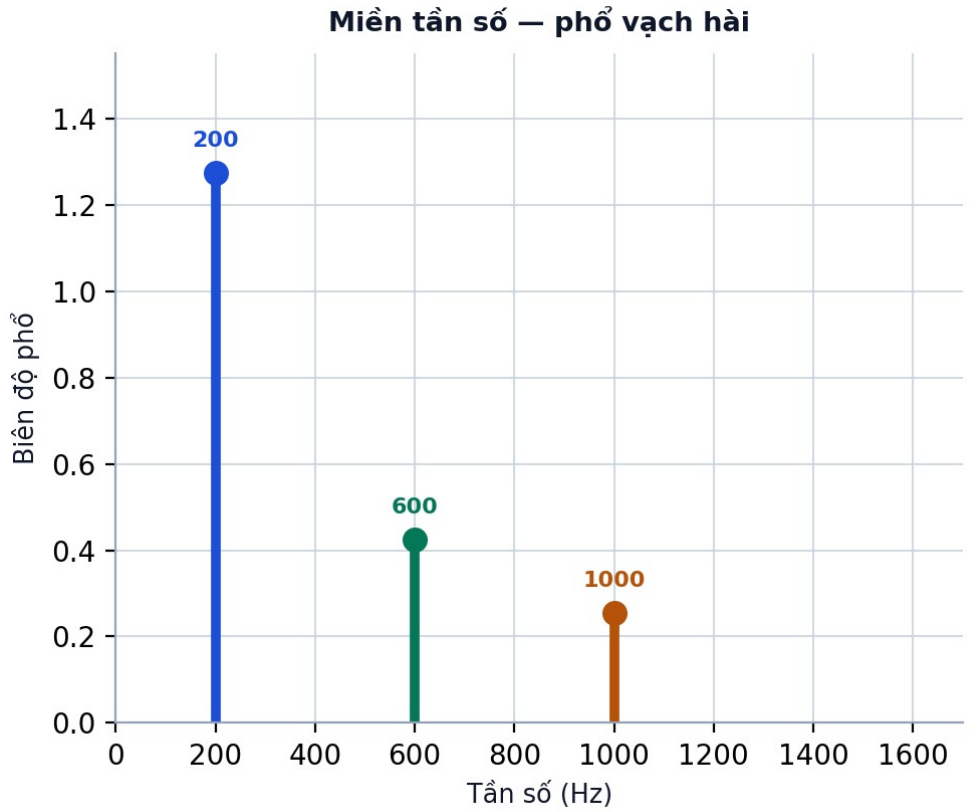




Thêm sóng 600 Hz với biên độ 1/3. Đỉnh sóng tổng bắt đầu bẹt xuống.

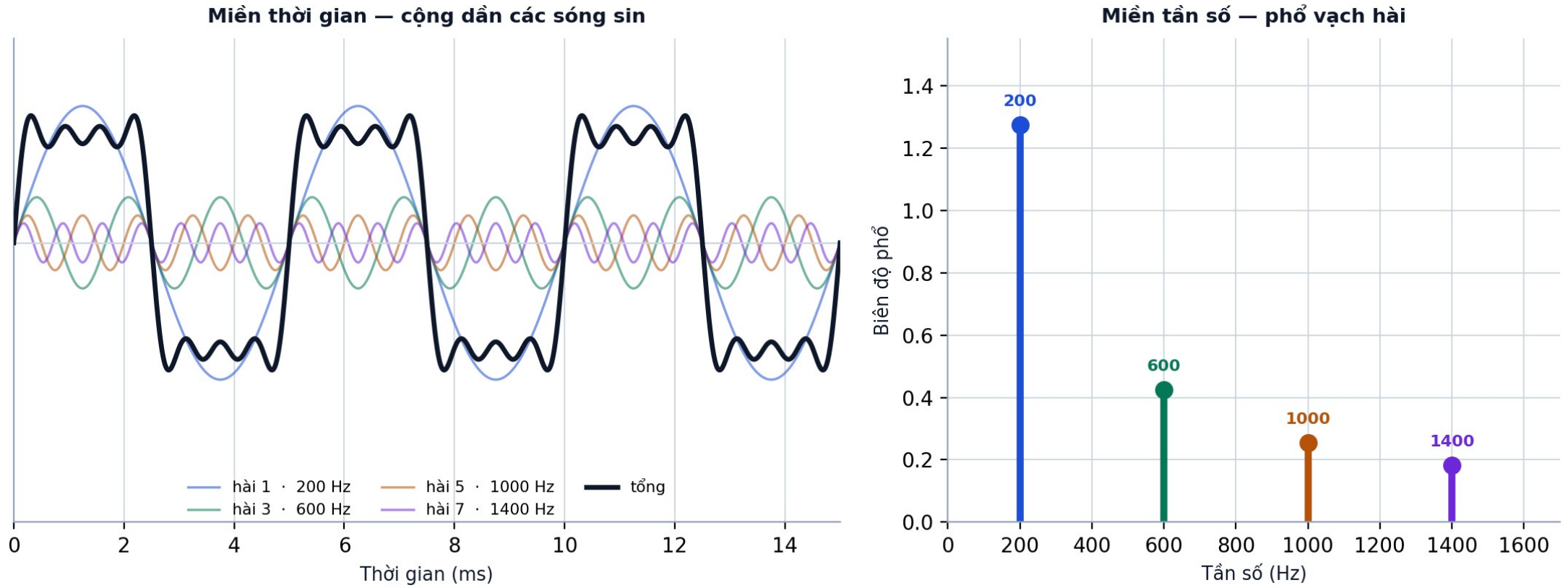
Thêm hài bậc 5





Thêm 1.000 Hz biên độ 1/5. Đỉnh phẳng dần, sườn dốc lên rõ hơn.

Bốn hài lẻ đã ra hình sóng vuông



Chỉ với 4 sóng sin, hình dạng vuông đã hiện rõ. Phổ bên phải là "công thức pha trộn" của nó.

CÔNG THỨC (4.1)

Biến đổi Fourier rời rạc — DFT

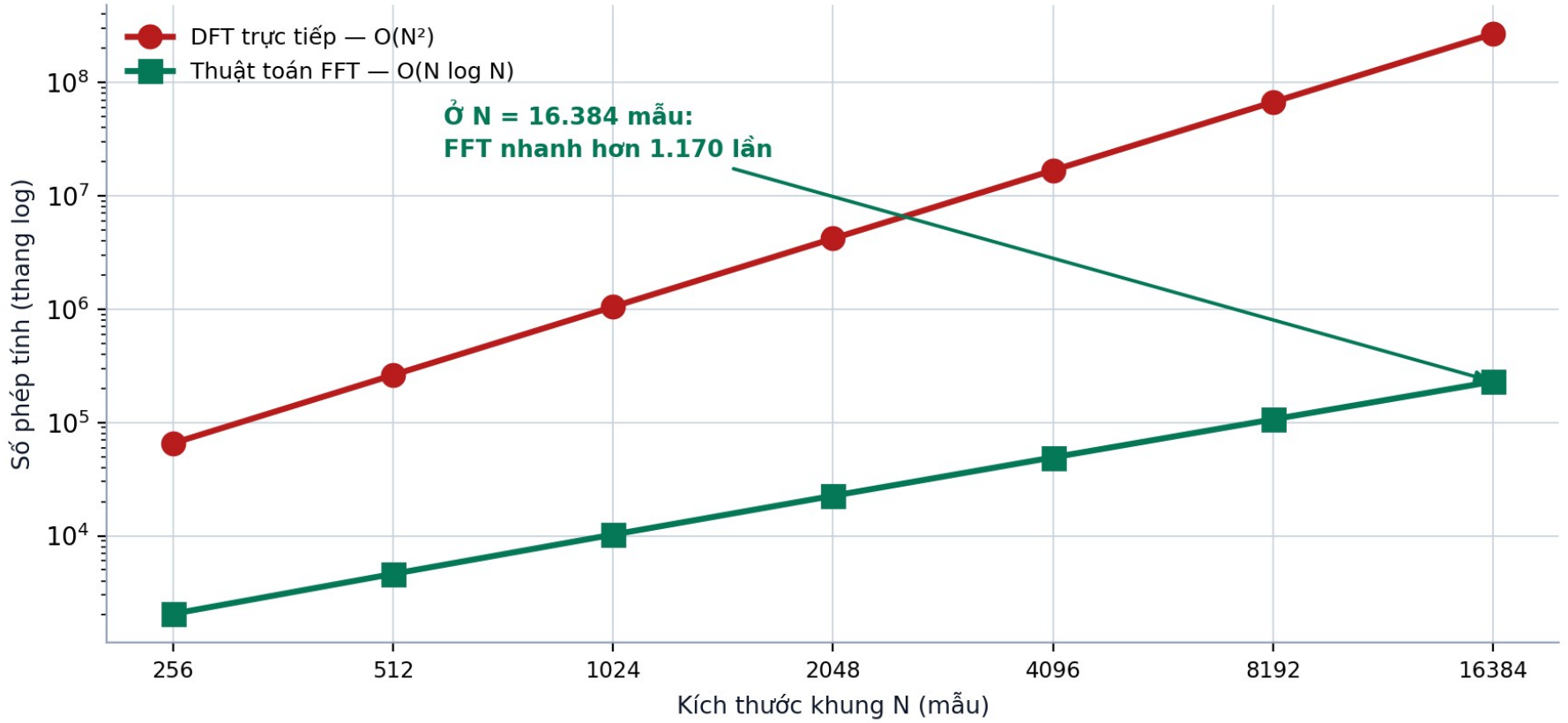
Với tín hiệu âm thanh số (một chuỗi mẫu biên độ rời rạc), phiên bản toán học tương ứng là **Biến đổi Fourier rời rạc (DFT)**.

$$X(k) = \sum_{n=0}^{N-1} x(n)\, e^{-j2\pi kn/N}$$

- **x(n)** — dãy mẫu trong miền thời gian; **X(k)** — hệ số phổ thứ k trong miền tần số.
- Mỗi hệ số X(k) cho biết **thành phần tần số thứ k** đóng góp biên độ và pha bao nhiêu.
- Tính DFT theo đúng công thức này đòi hỏi độ phức tạp **O(N²)** — máy tính không kham nổi theo thời gian thực với mảng âm thanh dài.

- Thuật toán **FFT (Fast Fourier Transform)** ra đời giảm độ phức tạp còn **O(N log N)**, tăng tốc hàng nghìn lần. FFT là công cụ tiêu chuẩn nhúng trong mọi thư viện DSP.

O(N²) so với O(N log N)



Ở khung 16.384 mẫu, FFT nhanh hơn DFT trực tiếp hơn một nghìn lần.

MỤC 4.2.2

Đọc hiểu biểu đồ phổ — hai trục và ba dạng

Một biểu đồ phổ tiêu chuẩn có hai trục cố định. Đọc đúng hai trục này là kỹ năng cơ bản của kỹ sư âm thanh.

- **Trục hoành (X):** tần số dao động, đơn vị **Hz hoặc kHz**. Thường hiển thị theo **thang logarit** cho tương thích với cảm nhận cao độ của tai người.
- **Trục tung (Y):** biên độ phổ hoặc mức năng lượng, thường đo bằng **dB hoặc dBFS**.
- **Dạng 1 — một đỉnh nhọn duy nhất:** sóng sin thuần, đơn âm tinh khiết.
- **Dạng 2 — nhiều đỉnh cách đều:** âm có cao độ kèm chuỗi hài (giọng nói, violin, sáo). Đỉnh đầu là $\mathbf{f_0}$, các đỉnh sau là hài bậc 2, 3, 4.
- **Dạng 3 — trải rộng, gồ ghề, không đỉnh rõ:** tiếng động phi điều hòa (Foley, va đập, nổ), tiếng ù nền, nhiễu trắng.

CHỐT LẠI MỤC 4.2

Chốt lại: Fourier và FFT

Ý tưởng Fourier: mọi tín hiệu = tổng các sóng sin đơn tần, mỗi sóng có tần số, biên độ, pha xác định.

Biểu đồ phổ: **trục X là tần số** (thường thang log), **trục Y là biên độ** (dB).

DFT là phiên bản rời rạc dùng cho mảng số; **FFT** là thuật toán tính DFT nhanh.

Ba dạng phổ cần thuộc: **một đỉnh · chuỗi đỉnh cách đều · trải rộng không đỉnh**.

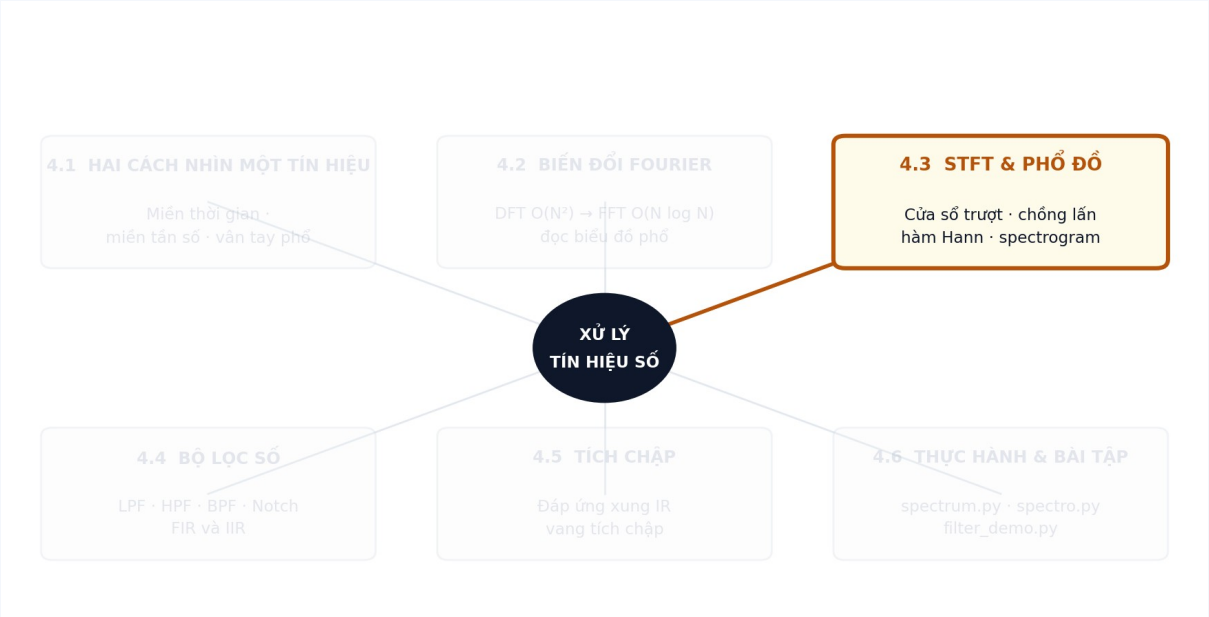
$O(N^2) \rightarrow O(N \log N)$ — cùng kết quả, nhanh hơn hàng nghìn lần ở khung lớn.

FFT bộc lộ tần số **vốn đã có** trong tín hiệu, không tạo thêm gì mới.

TIẾT 3

STFT và phổ đồ thời gian — spectrogram

- Quy luật đánh đổi độ phân giải thời gian và tần số
- Cửa sổ trượt, chồng lấn khung và hàm Hann
- Rò rỉ phổ và cách khắc phục
- Đọc hiểu ba trục của spectrogram



MỤC 4.2.3

Quy luật đánh đổi — không thể có cả hai

Trong xử lý tín hiệu số có một quy luật bất biến mang tính vật lý: **không thể đạt độ phân giải thời gian và độ phân giải tần số tối đa cùng một lúc.**

- Muốn **phân biệt rõ hai tần số nằm sát nhau**, FFT bắt buộc phải phân tích trên một đoạn tín hiệu **đủ dài** trong miền thời gian.
- Khung càng dài (N lớn) ⇒ **Δf càng mịn**, phân biệt cao độ càng chính xác — nhưng **thời điểm bị nhòe**, không biết tiếng đó vang lên ở mili-giây nào.
- Ngược lại, khung càng ngắn ⇒ **định vị thời gian càng chính xác** — nhưng dải tần thu được **thô và mờ**, không phân biệt được các nốt gần nhau.
- Đây không phải hạn chế của thuật toán mà là một **quy luật vật lý** — cùng bản chất với nguyên lý bất định Heisenberg trong vật lý lượng tử.

CÔNG THỨC (4.2)

Độ phân giải tần số theo kích thước khung

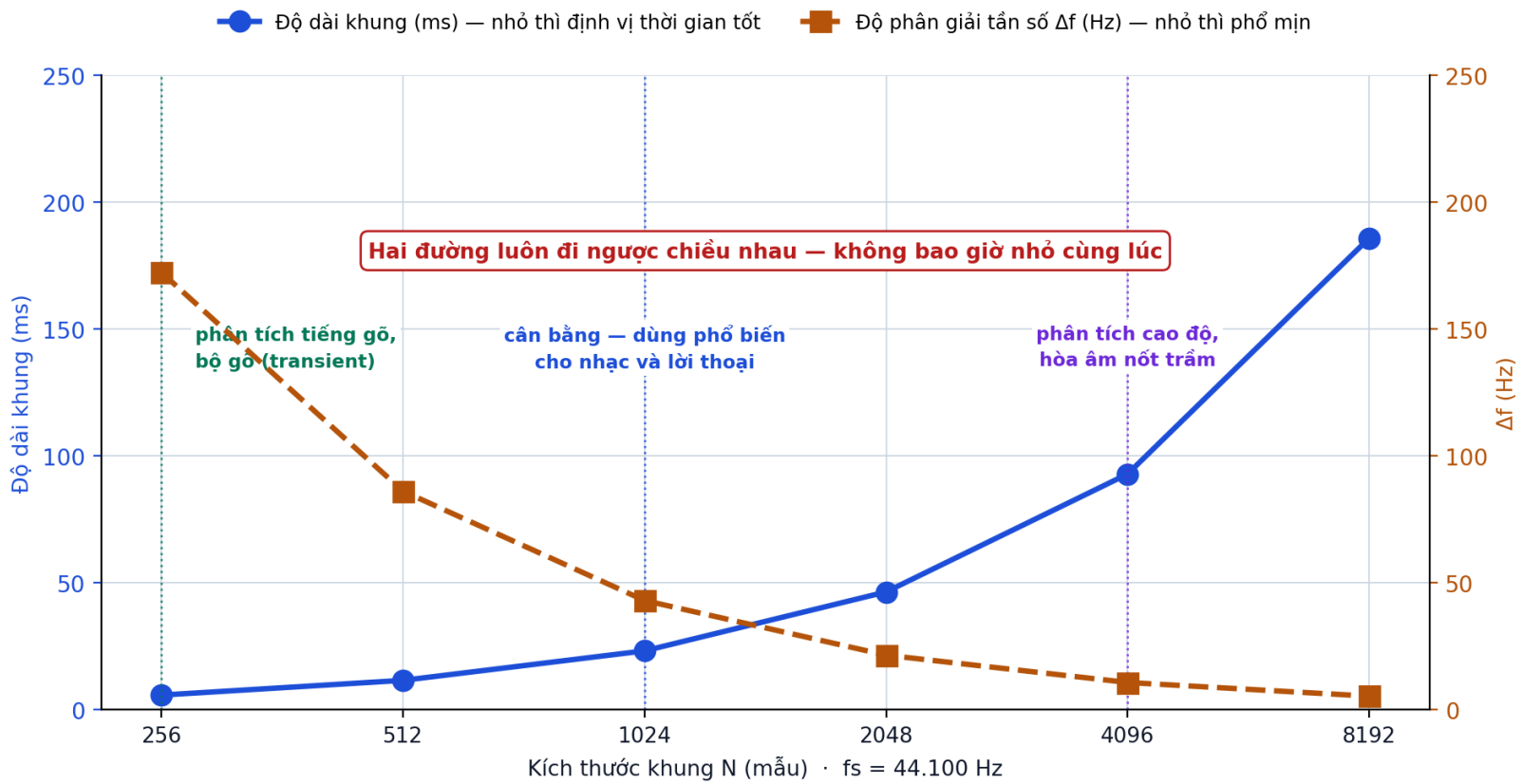
Hai công thức đi liền nhau và luôn ngược chiều nhau: **Δf = fs / N** và **T_khung = N / fs**.

$$\Delta f = \frac{f_s}{N} \quad (\text{Hz})$$

- Ở fs = 44.100 Hz, khung **256 mẫu**: Δf ≈ **172,3 Hz** (thô) nhưng khung chỉ **5,8 ms** (định vị rất tốt).
- Khung **1.024 mẫu**: Δf ≈ **43,1 Hz**, khung **23,2 ms** — điểm cân bằng, dùng phổ biến nhất.
- Khung **4.096 mẫu**: Δf ≈ **10,8 Hz** (mịn) nhưng khung tới **92,9 ms** (định vị kém).
- Nhân hai đại lượng lại: **Δf × T_khung = 1** — luôn luôn. Muốn cái này nhỏ thì cái kia phải lớn.

BẢNG 4.3 → BIỂU ĐỒ

Hai đường luôn đi ngược chiều nhau



Chọn N là chọn ưu tiên: định vị thời gian tốt, hay phân giải tần số mịn — không có lựa chọn tốt cả hai.

BẢNG 4.3

Bảng tra kích thước khung ở fs = 44.100 Hz

Kích thước khung N	Độ dài khung (ms)	Độ phân giải Δf (Hz)	Định vị thời gian / Ứng dụng
256 mẫu	256 / 44.100 ≈ 5,8 ms	44.100 / 256 ≈ 172,3 Hz	Rất tốt — phân tích tiếng gõ, bộ gõ (transient)
1.024 mẫu	1.024 / 44.100 ≈ 23,2 ms	44.100 / 1.024 ≈ 43,1 Hz	Cân bằng — phổ biến cho nhạc và lời thoại
4.096 mẫu	4.096 / 44.100 ≈ 92,9 ms	44.100 / 4.096 ≈ 10,8 Hz	Kém — phân tích cao độ, hòa âm nốt trầm

Sinh viên nên chụp lại bảng này. Khi cấu hình bộ phân tích phổ trong Audacity hay Python, đây là con số cần tra.

VÍ DỤ GIẢI TỪNG BƯỚC

Bài tập 2 — tính đánh đổi ở fs = 48.000 Hz

Tình huống: Phòng dựng phim đặt fs = **48.000 Hz**. (a) Tính Δf và độ dài khung khi N = **512 mẫu**. (b) Tính lại khi N = **4.800 mẫu**. (c) Cần phân biệt hai nốt trầm cách nhau chỉ **5 Hz** — nên chọn cấu hình nào, và cái giá phải trả là gì?

1. (a) N = 512: Δf = 48.000 / 512 = **93,75 Hz** (rất thô); T = 512 / 48.000 ≈ **10,7 ms** (định vị rất tốt).
2. (b) N = 4.800: Δf = 48.000 / 4.800 = **10 Hz** (mịn hơn nhiều); T = 4.800 / 48.000 = **100 ms** (định vị thô).
3. (c) Để phân biệt lệch 5 Hz, cần **Δf ≤ 5 Hz**. Cả hai cấu hình trên đều không đạt (93,75 và 10 đều lớn hơn 5).
4. Tăng khung lên tối thiểu **N = 9.600 mẫu**: Δf = 48.000 / 9.600 = **5 Hz** — vừa đủ.
5. Cái giá: T_khung = 9.600 / 48.000 = **200 ms**. Quá lớn với thính giác.

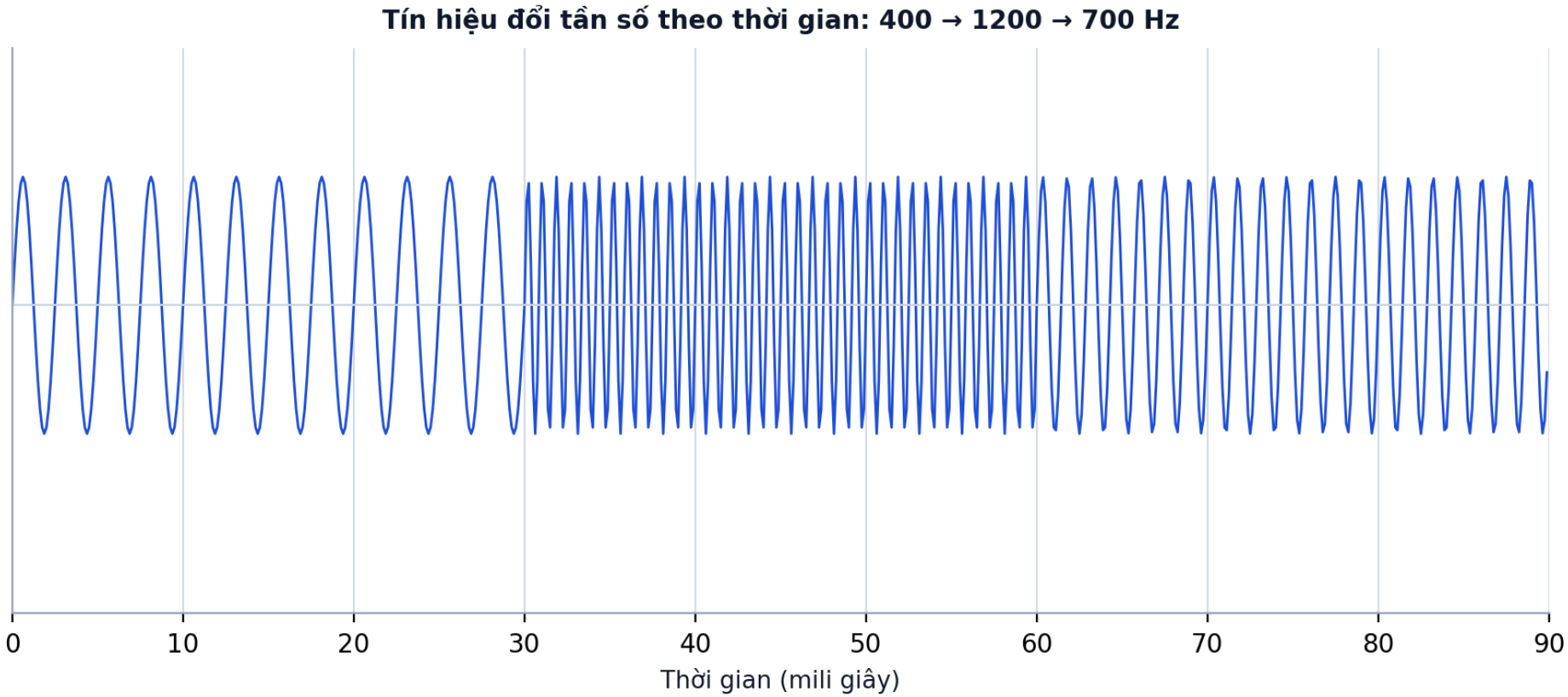
Kết luận nghề: Với khung 200 ms, thông tin biên độ bị nhòe rộng theo trục thời gian — kỹ sư **mất hoàn toàn khả năng định vị chính xác mili-giây** của các tiếng gõ đập transient. Muốn phân tích cả hai mặt thì phải chạy hai lần phân tích với hai cấu hình khác nhau.

Vì sao cần STFT

Một tác phẩm điện ảnh hay bản nhạc **không bao giờ đứng yên**: lời thoại, tiếng động, nốt nhạc liên tục biến đổi dọc trục thời gian.

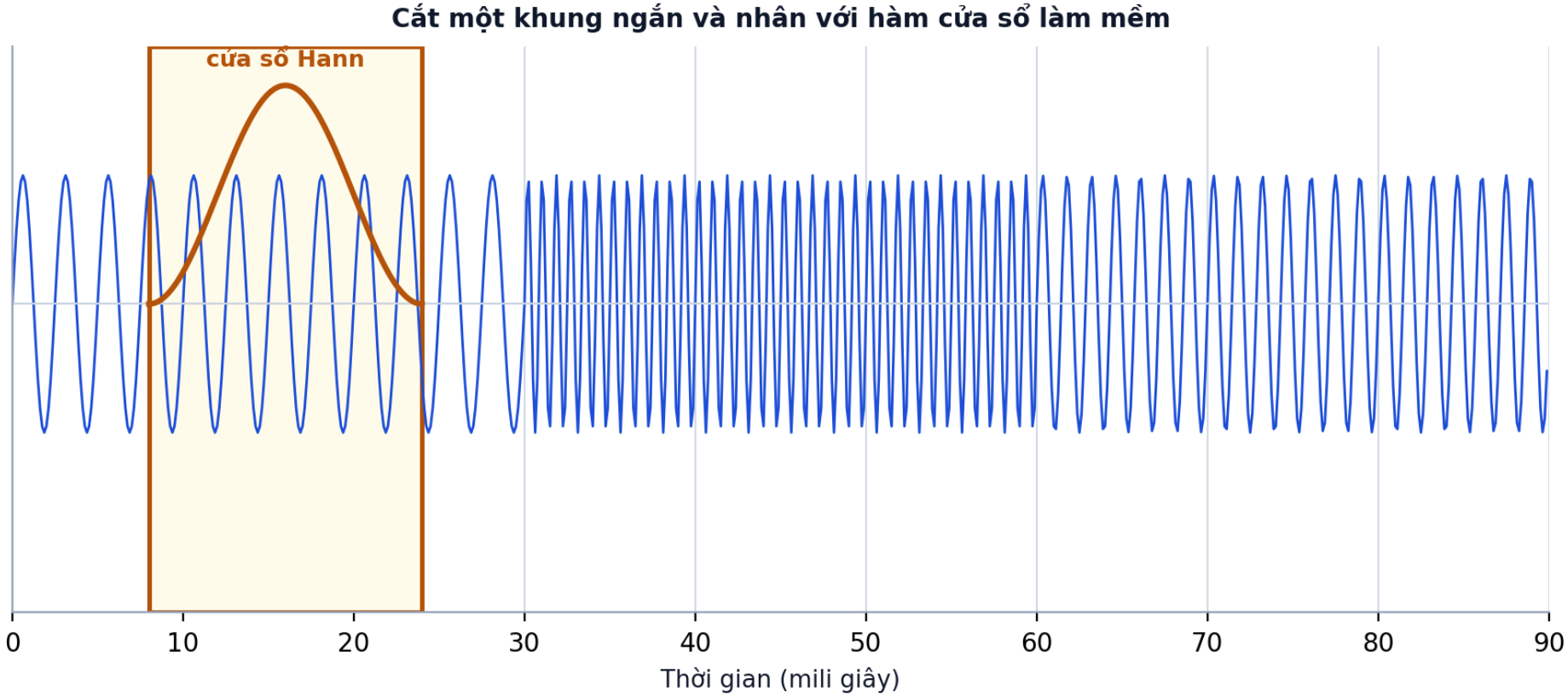
- Nếu áp dụng FFT cho **toàn bộ** file dài, ta chỉ nhận về **một biểu đồ phổ trung bình cộng của cả bài**.
- Ta biết bài có những tần số nào, nhưng **hoàn toàn bất lực** trong việc xác định tần số đó vang lên **lúc nào**.
- Giải pháp: **Biến đổi Fourier thời gian ngắn — STFT**. Thay vì phân tích một lần, dùng một **cửa sổ trượt** ngắn (vài chục mili-giây) trượt dọc tín hiệu.
- Tại mỗi vị trí dừng, trích đoạn ngắn trong cửa sổ và chạy FFT trên đoạn đó. Lặp từ đầu đến cuối ⇒ thu được **chuỗi phổ biến thiên liên tục theo thời gian**.

Bước 1 — tín hiệu có tần số thay đổi theo thời gian



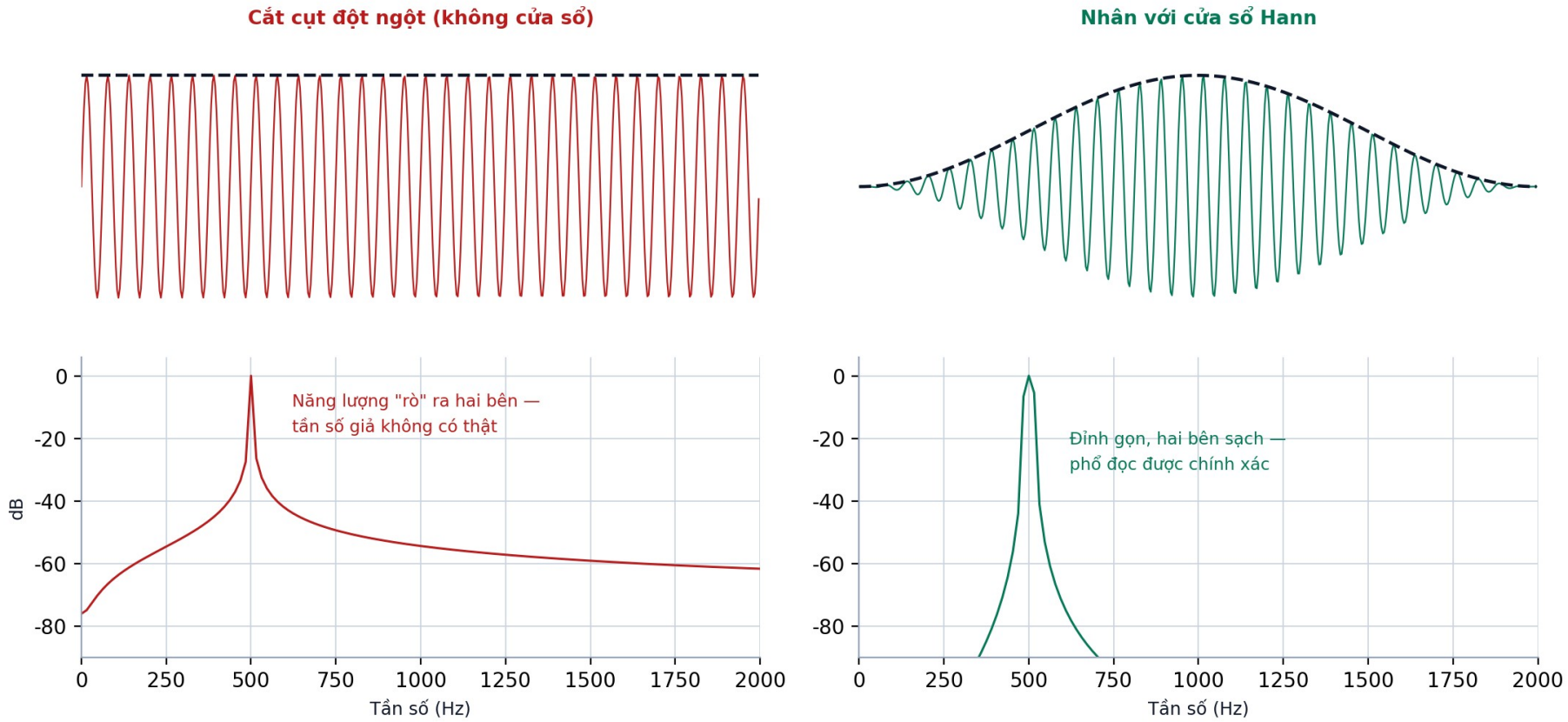
Ba đoạn: 400 Hz → 1200 Hz → 700 Hz. FFT toàn bộ chỉ cho biết có ba tần số, không cho biết thứ tự.

Bước 2 — cắt một khung ngắn và nhân với cửa sổ



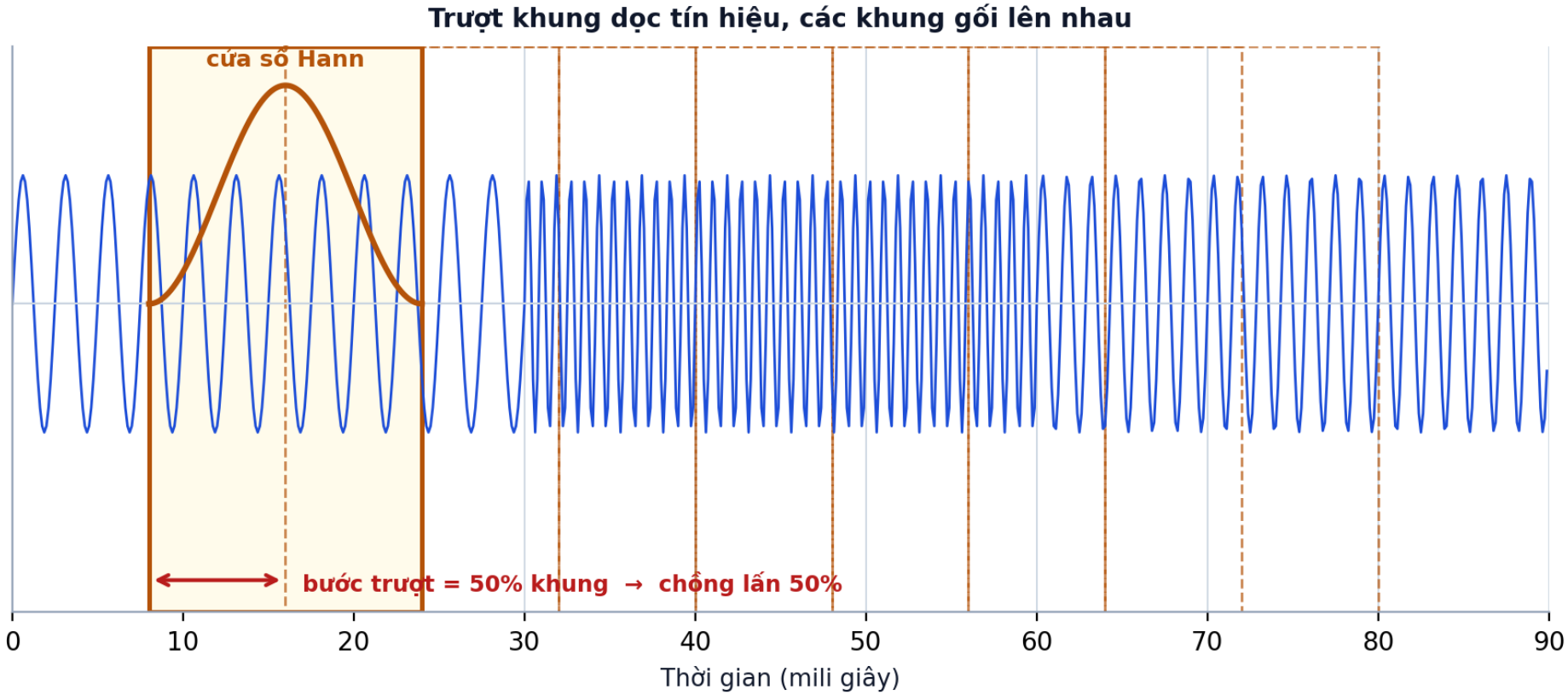
Đoạn tín hiệu trong khung được nhân với hàm cửa sổ Hann — biên độ triệt tiêu dần về 0 ở hai đầu.

Rò rỉ phổ — spectral leakage



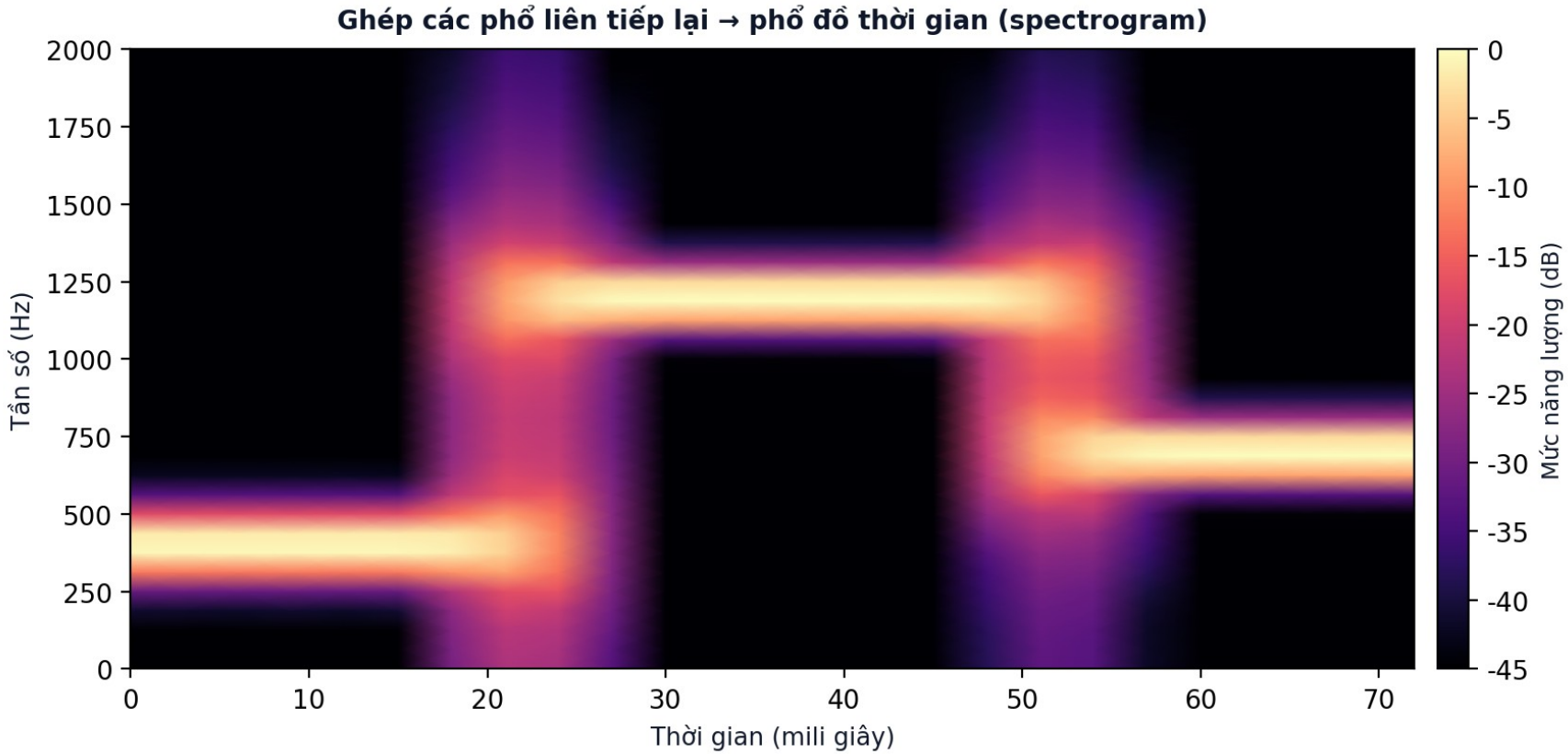
Trái: cắt cụt đột ngột làm năng lượng "rò" ra hai bên, sinh tần số giả. Phải: cửa sổ Hann cho đỉnh gọn, hai bên sạch.

Bước 3 — trượt khung, các khung gối lên nhau



Khung kề nhau chồng lấn 50% (hoặc 75%) để không mất thông tin tại ranh giới cắt.

Bước 4 — ghép các phổ lại thành spectrogram



Ma trận hai chiều: trục ngang là thời gian, trục dọc là tần số, màu sắc là năng lượng.

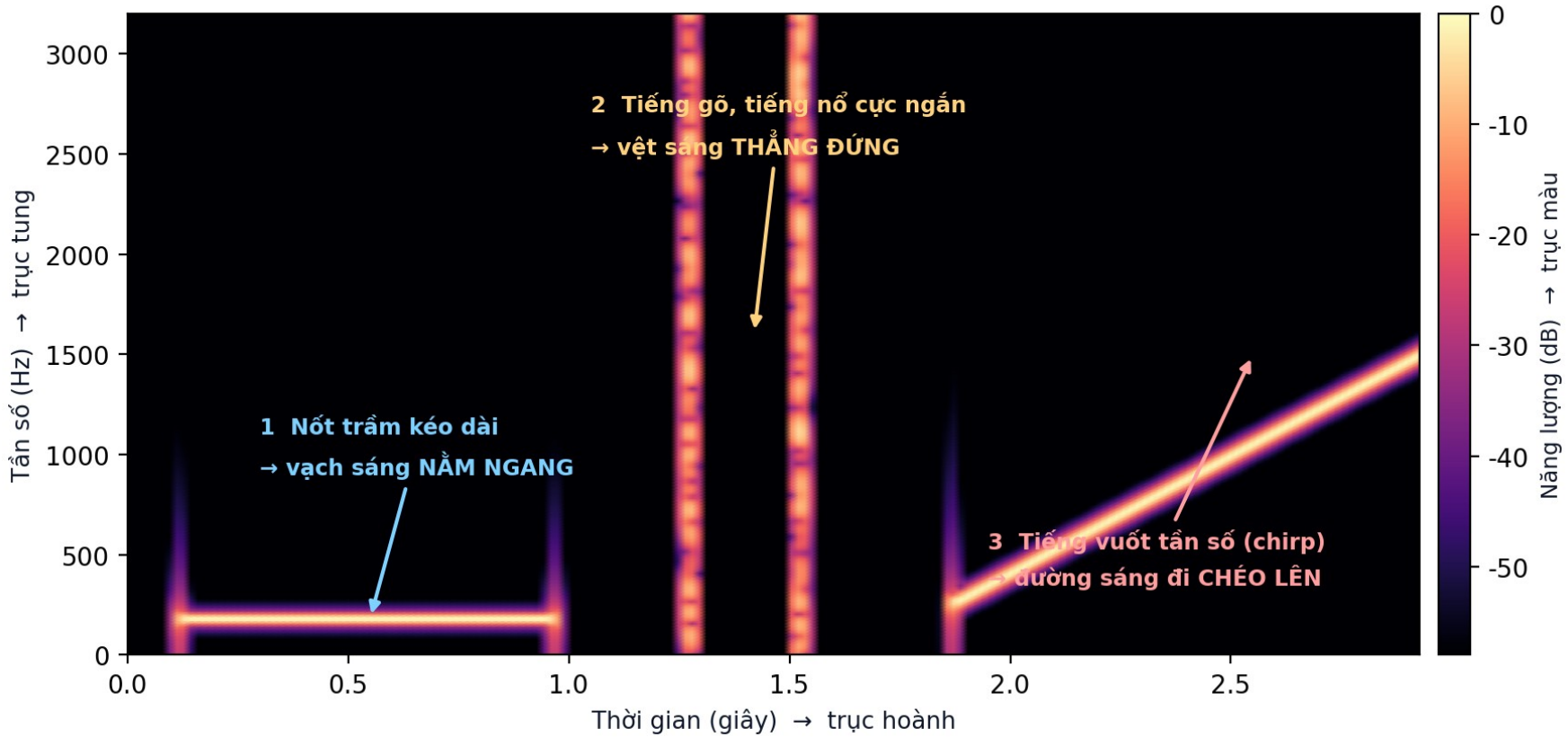
MỤC 4.3.3

Ba trục của một spectrogram

Kết quả STFT là ma trận hai chiều, trực quan hóa thành **phổ đồ thời gian (Spectrogram)** — biểu diễn đồng thời ba trục thông tin.

- **Trục hoành:** thời gian thực chạy từ đầu đến cuối đoạn âm thanh (giây).
- **Trục dọc:** dải tần số từ 0 Hz đến giới hạn Nyquist **fs/2**.
- **Trục màu sắc:** mức năng lượng tại mốc thời gian và tần số tương ứng, thường quy đổi sang **dB**. Màu tối (đen, xanh sẫm) = năng lượng cực yếu; màu nóng sáng (vàng, đỏ, trắng) = năng lượng lớn.
- Ba trục trên một mặt phẳng — đó là lý do spectrogram là công cụ chẩn đoán mạnh nhất mà kỹ sư âm thanh có.

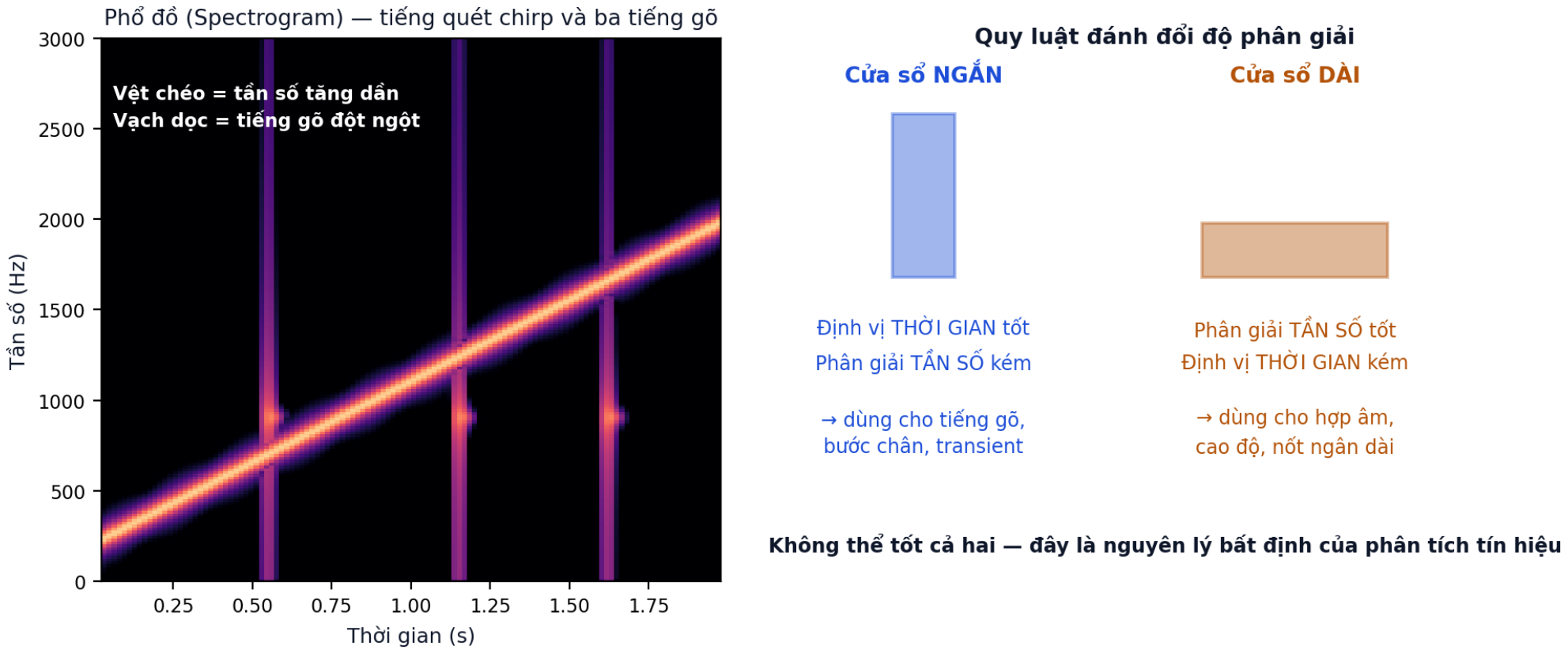
Ba dấu vết hình học cần nhận ra ngay



Mọi hành động âm thanh đều để lại một dấu vết hình học đặc trưng trên spectrogram.

HÌNH 4.2

Phổ đồ và quy luật đánh đổi trong giáo trình



Hình 4.2 — Phổ đồ thời gian và quy luật đánh đổi giữa độ phân giải thời gian và tần số.

Spectrogram chính là công cụ đã lật tẩy Fake Lossless ở Chương 3.

Vạch cắt brick-wall phẳng lỳ tại 16 kHz — bây giờ các em biết bức tranh đó được tính ra thế nào.

CHỐT LẠI MỤC 4.3

Chốt lại: đánh đổi và STFT

Δf = fs / N và **T_khung = N / fs** — hai đại lượng luôn ngược chiều, tích bằng 1.

Hàm cửa sổ Hann là bắt buộc — không có nó thì rò rỉ phổ sinh tần số giả.

Khung ngắn → định vị thời gian tốt, phổ thô. Khung dài → phổ mịn, thời gian nhòe.

Khung kề nhau **chồng lấn 50–75%** để không mất thông tin ở ranh giới.

N = 256 cho transient · **N = 1024 cân bằng** · N = 4096 cho cao độ.

Spectrogram ba trục: thời gian (ngang) · tần số (dọc) · năng lượng (màu).

STFT = trượt cửa sổ ngắn dọc tín hiệu, chạy FFT trên từng khung.

Ba dấu vết: **vạch ngang** = nốt kéo dài · **vệt đứng** = tiếng gõ · **đường chéo** = chirp.

TIẾT 4

Bộ lọc số và phép toán tích chập

- Bốn loại bộ lọc cơ bản và ứng dụng phim trường
- Phương trình sai phân — công thức tổng quát
- Hai họ bộ lọc FIR và IIR
- Tích chập và công nghệ vang tích chập



MỤC 4.4

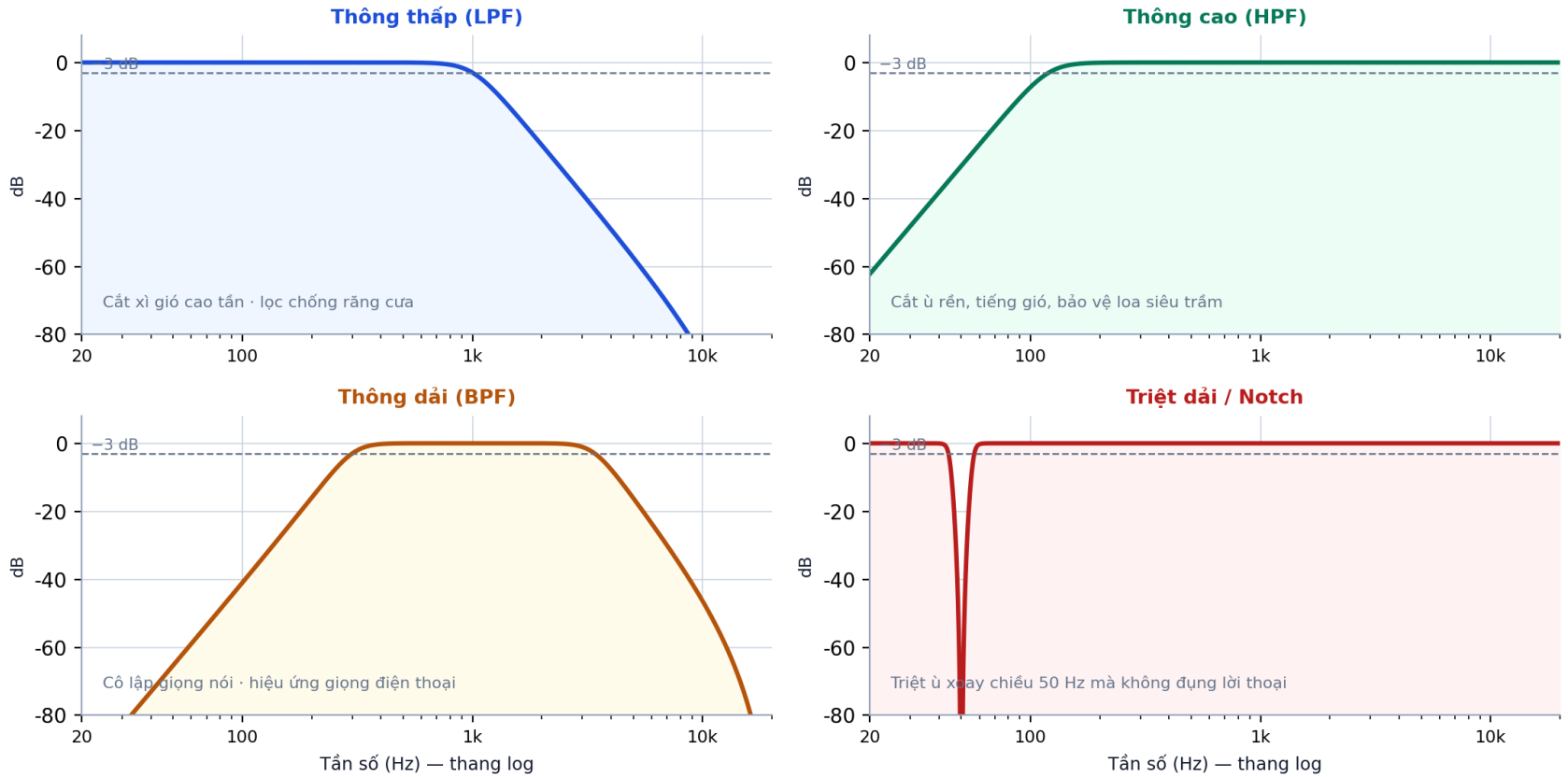
Bộ lọc số là gì

Khi đã có tư duy "hai cách nhìn tín hiệu" và quan sát được phổ tần số, việc can thiệp để tinh chỉnh âm sắc trở nên **rõ ràng và chủ động**.

- **Bộ lọc số (digital filter)** là một hệ thống toán học được lập trình để **giữ lại** một số dải tần mong muốn và **làm yếu đi** (suy hao) các dải tần không mong muốn.
- Nó không "xóa" tần số — nó **giảm biên độ** của dải tần đó xuống mức không còn nghe thấy.
- Mức suy hao đo bằng **dB**; tần số mà biên độ tụt xuống **−3 dB** gọi là **tần số cắt f_c**.
- Mọi EQ trong DAW đều là một tập hợp bộ lọc số. Núm "Low Cut" trên mixer chính là bộ lọc thông cao — chỉ là được đặt tên thân thiện hơn.

MỤC 4.4.1

Bốn loại bộ lọc số cơ bản



Đáp tuyến tần số thực tế của bốn bộ lọc Butterworth bậc 4 — dựng bằng chính scipy.signal mà sinh viên sẽ dùng.

BÀI TẬP 3

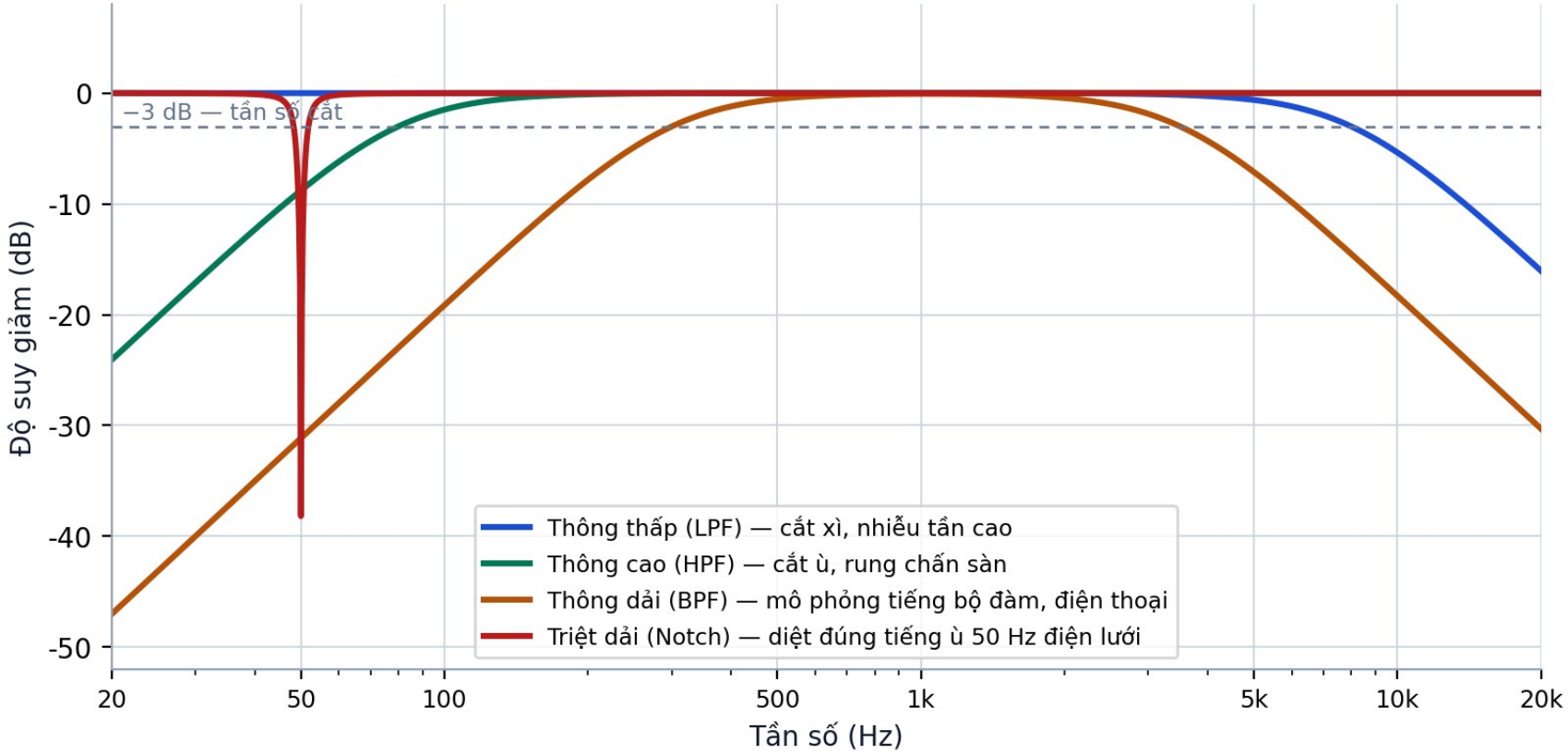
Chọn bộ lọc nào cho tình huống nào

Tình huống trên phim trường	Bộ lọc	Cài đặt
Tiếng thoại ngoài trời dính tạp âm gió trầm rền	Thông cao (HPF)	$f_c \approx 80 – 100$ Hz — cắt rền mà không đụng formant giọng
Cảnh đàm thoại qua điện thoại di động cổ	Thông dải (BPF)	Chỉ cho 300 – 3.400 Hz đi qua
Bản thu dính tiếng ù xoay chiều 50 Hz đều đặn	Triệt dải (Notch)	Rãnh cực hẹp tại đúng 50 Hz
Chống răng cưa trước bộ chuyển đổi ADC	Thông thấp (LPF)	f_c sát Nyquist — thường 20 kHz ở fs = 44,1 kHz

*Đây chính là bài tập 3 của chương — bốn tình huống, bốn bộ lọc, mỗi loại dùng đúng một lần. Sinh viên phải giải thích được **lý do chọn**, không chỉ nêu tên.*

HÌNH 4.3

Đáp tuyến bốn loại bộ lọc trong giáo trình



Hình 4.3 — Đáp tuyến tần số của bốn loại bộ lọc số: thông thấp, thông cao, thông dải và triệt dải.

Phương trình sai phân

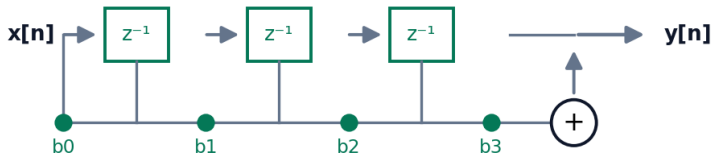
Về mặt toán học, bộ lọc số hoạt động **trực tiếp trong miền thời gian**: tính từng mẫu đầu ra y[n] bằng một tổ hợp tuyến tính có trọng số.

$$y(n) = \sum_{k=0}^{M} b_k x(n-k) - \sum_{k=1}^{N} a_k y(n-k)$$

- **b_k** — trọng số áp lên các mẫu **đầu vào** hiện tại x[n] và quá khứ x[n−k].
- **a_k** — trọng số áp lên các mẫu **đầu ra** vừa tính trước đó y[n−k] (nhánh hồi tiếp).
- Chọn bộ trọng số khác nhau ⇒ ra đáp tuyến tần số khác nhau ⇒ ra loại bộ lọc khác nhau.
- Điều bất ngờ: bộ lọc làm việc trong **miền thời gian**, chỉ bằng phép nhân và cộng — nhưng hiệu quả của nó thể hiện ở **miền tần số**. Đó là vẻ đẹp của lý thuyết Fourier.

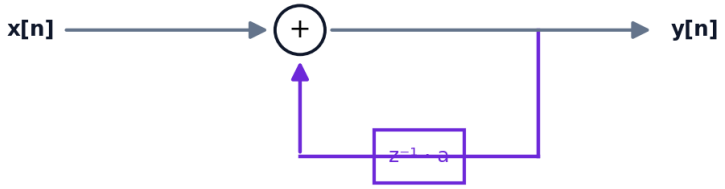
FIR và IIR — khác nhau ở nhánh hồi tiếp

FIR — không hồi tiếp



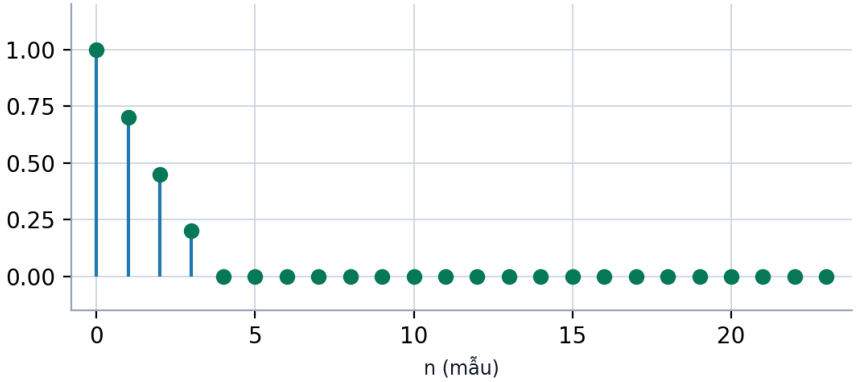
y[n] chỉ phụ thuộc mẫu VÀO hiện tại và quá khứ

IIR — có nhánh hồi tiếp

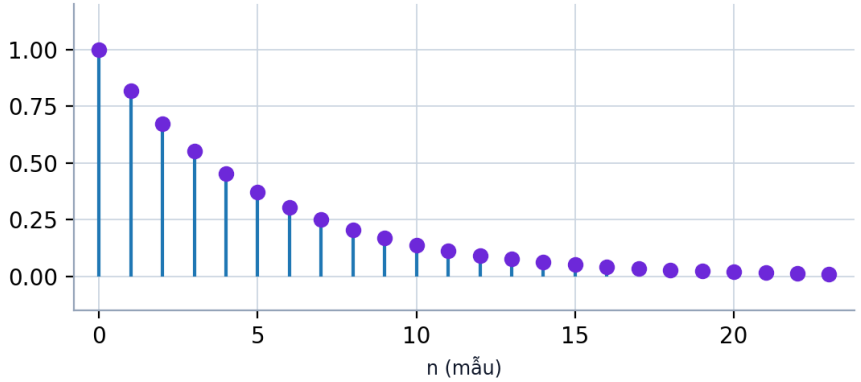


y[n] phụ thuộc cả mẫu RA vừa tính trước đó

Đáp ứng xung HỮU HẠN — tắt sau 4 mẫu



Đáp ứng xung VÔ HẠN — nhỏ dần nhưng không bao giờ hết



FIR: đáp ứng xung tắt sau vài mẫu. IIR: đáp ứng xung nhỏ dần nhưng về lý thuyết không bao giờ hết.

MỐI LIÊN KẾT

Hiệu ứng tiếng vọng Echo chính là một bộ lọc IIR

Ý tưởng "mẫu đầu ra phụ thuộc vào mẫu đầu ra trước đó" nghe rất trừu tượng — nhưng thực tế nó chính là **sơ đồ khối của hiệu ứng tiếng vọng có hồi tiếp**.

- Mỗi tiếng vọng dội lại sau một khoảng trễ là **một bản sao của mẫu đầu ra trước đó**.
- Bản sao đó được **giảm âm lượng** (Feedback Gain < 1) rồi **cộng gộp ngược** vào tín hiệu đầu vào hiện tại.
- Cấu trúc lặp lại vô hạn này chính là mô hình toán học của **một bộ lọc IIR**.
- Vì Feedback Gain < 1 nên các tiếng vọng nhỏ dần và tắt. Nếu đặt Gain ≥ 1, hệ **mất ổn định** — tiếng vọng to dần vô hạn, đúng hiện tượng mất ổn định của IIR.
- Các hiệu ứng âm thanh không phải khái niệm độc lập xa lạ — chúng đều xây trên **cùng bộ khung toán học của bộ lọc số**.

MỤC 4.5.1

Đáp ứng xung — dấu vân tay của một không gian

Mỗi hệ thống âm học — một rạp hát, một hành lang bê tông, một thùng loa, một bộ lọc số — đều có một "dấu vân tay" vật lý đặc trưng duy nhất gọi là **đáp ứng xung (Impulse Response — IR)**.

- Đáp ứng xung là **toàn bộ sóng âm dội lại** mà hệ thống phát ra khi ta kích thích nó bằng một **xung tức thời cực ngắn**.
- Về lý thuyết đó là hàm delta Dirac; trong thực tế là một tiếng **"tách"** hoặc tiếng nổ đanh gọn chứa năng lượng phân bổ đều ở mọi tần số.
- **Tích chập** một tín hiệu x[n] với đáp ứng xung h[n] của một hệ thống chính là hành vi **"in dấu"** toàn bộ đặc tính âm học của hệ thống đó lên tín hiệu.
- Kết quả y[n] mang theo trọn vẹn độ sâu, các đợt phản xạ sớm và đuôi vang của không gian.

CÔNG THỨC (4.4)

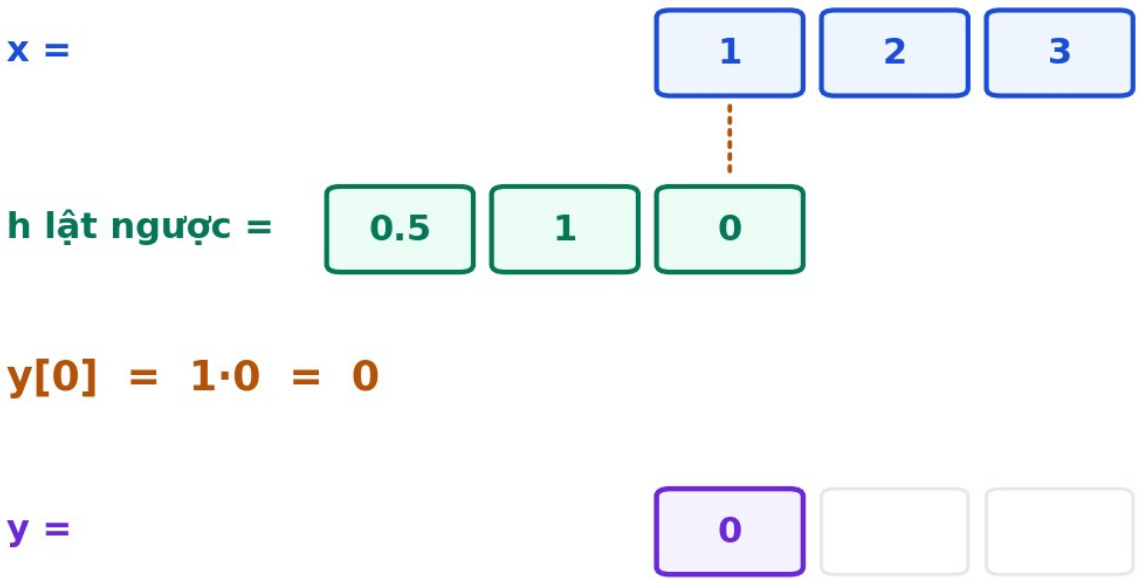
Phép toán tích chập

Công thức trông phức tạp, nhưng ý nghĩa với người làm âm thanh lại vô cùng trực quan: **in đặc tính hệ thống lên tín hiệu**.

$$y(n) = x(n) * h(n) = \sum_{k} x(k)\, h(n-k)$$

- **x(k)** — tín hiệu đầu vào; **h(n−k)** — đáp ứng xung đã lật ngược và trượt.
- Phép tính thực hiện bằng cách **trượt ngược dãy h qua dãy x** và tính tổng tích các phần tử chồng lấn tại mỗi bước.
- Độ dài kết quả: **len(x) + len(h) − 1** — dài hơn tín hiệu vào vì mang theo cả phần "đuôi" của đáp ứng xung.
- Đáp ứng xung của một phòng thật có hàng chục nghìn mẫu ⇒ đuôi tích chập rất dày. Đó là lý do vang tích chập ngốn CPU khủng khiếp.

Tính tay bước 0



Trượt dãy h đã lật ngược qua dãy x, mỗi bước cộng tích các ô chồng nhau

Bước 0 / 4

x = [1, 2, 3] ∗ h = [0, 1, 0,5]. Chỉ một ô chồng nhau: y[0] = 1·0 = 0.

Tính tay bước 1

x = 1 2 3

h lật ngược = 0.5 1 0

y[1] = 1·1 + 2·0 = 1

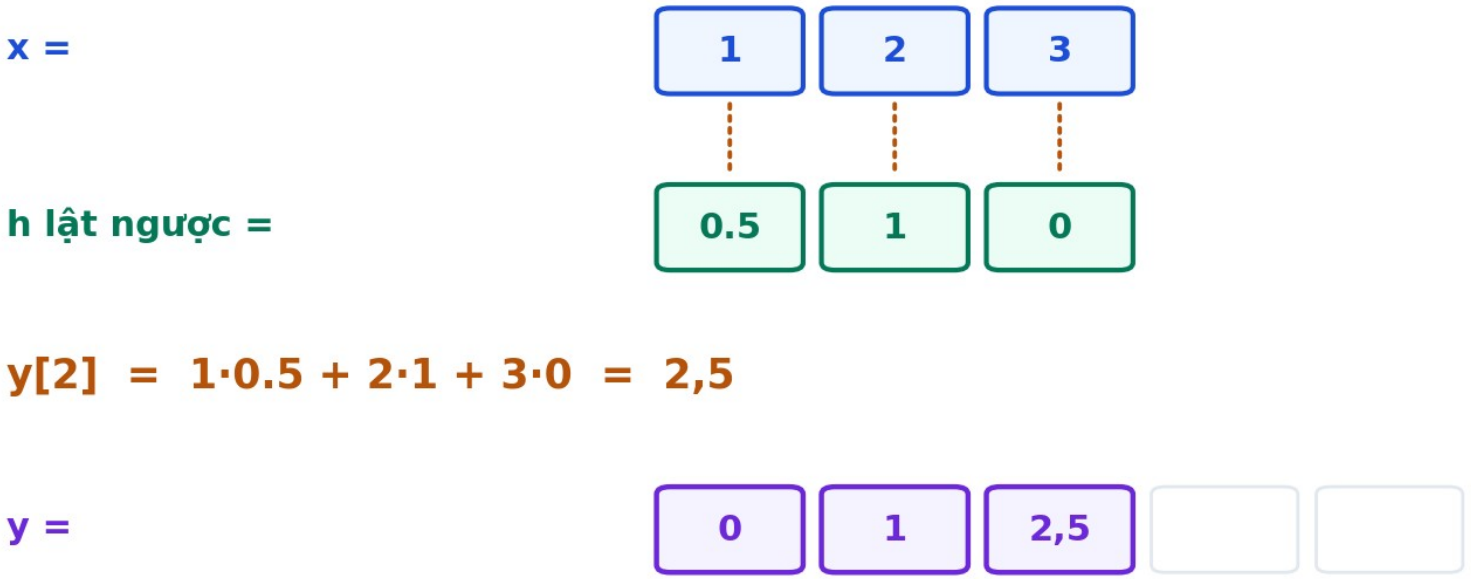
y = 0 1

Trượt dãy h đã lật ngược qua dãy x, mỗi bước cộng tích các ô chồng nhau

Bước 1 / 4

Hai ô chồng nhau: y[1] = 1·1 + 2·0 = 1.

Tính tay bước 2 — chồng lấn nhiều nhất



Trượt dãy h đã lật ngược qua dãy x, mỗi bước cộng tích các ô chồng nhau

Bước 2 / 4

Ba ô chồng nhau: y[2] = 1·0,5 + 2·1 + 3·0 = 2,5.

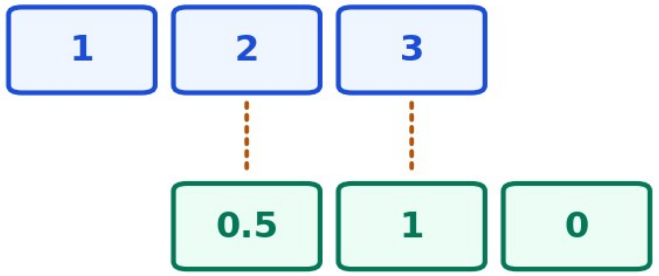
Tính tay bước 3

x = 1 2 3

h lật ngược = 0.5 1 0

y[3] = 2·0.5 + 3·1 = 4

y = 0 1 2,5 4



Trượt dãy h đã lật ngược qua dãy x, mỗi bước cộng tích các ô chồng nhau

Bước 3 / 4

y[3] = 2·0,5 + 3·1 = 4. Kết quả đã vượt quá độ dài của x — đây là phần "đuôi".

Tính tay bước 4 — kết quả cuối

x = 1 2 3

h lật ngược = 0.5 1 0

y[4] = 3·0.5 = 1,5

y = 0 1 2,5 4 1,5

Trượt dãy h đã lật ngược qua dãy x, mỗi bước cộng tích các ô chồng nhau

Bước 4 / 4

y = [0; 1; 2,5; 4; 1,5] — dài 5 mẫu = 3 + 3 − 1, đúng công thức.

BÀI TẬP 4

Tính tích chập và giải nghĩa vật lý

Tình huống: Tính tích chập của tín hiệu thoại ngắn **x = [2, 0, 1]** với đáp ứng xung **h = [1, 1]**. Dãy kết quả nói lên hệ thống này thực hiện hiệu ứng vật lý gì?

1. y[0] = x[0]·h[0] = 2 · 1 = **2**
2. y[1] = x[1]·h[0] + x[0]·h[1] = 0·1 + 2·1 = **2**
3. y[2] = x[2]·h[0] + x[1]·h[1] = 1·1 + 0·1 = **1**
4. y[3] = x[2]·h[1] = 1 · 1 = **1** ⇒ kết quả **y = [2, 2, 1, 1]**, dài 3 + 2 − 1 = 4 mẫu.
5. Giải nghĩa: h = [1, 1] nghĩa là **y[n] = x[n] + x[n−1]** — cộng mẫu hiện tại với chính nó trễ đi một bước.

Kết luận nghề: Đây là mô hình tối giản của một **bộ lọc làm mịn / làm nhòe dải cao** — tức một bộ lọc thông thấp đơn giản. Nó tạo ra một tiếng dội tức thời siêu ngắn dính liền vào âm thanh gốc. Chỉ hai con số [1, 1] đã đủ mô tả một hệ thống vật lý có thật.

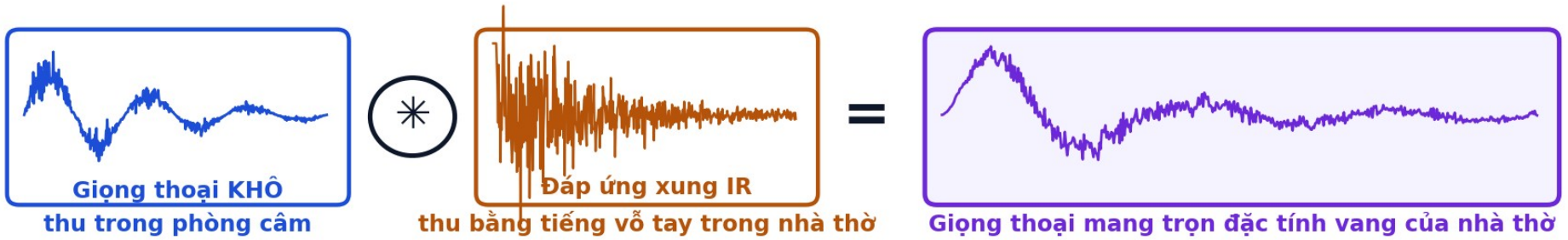
TIẾT 5

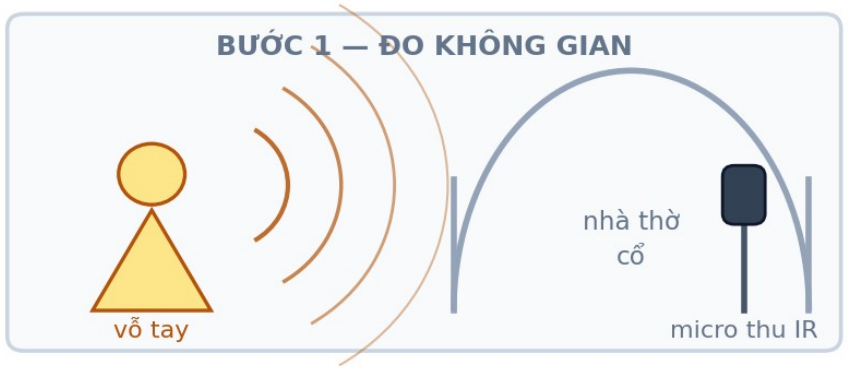
Vang tích chập, thực hành và tổng kết chương

- Convolution Reverb — công nghệ tạo vang trung thực nhất
- Ba bài thực hành Python: phổ · spectrogram · bộ lọc
- Bài tập bối cảnh trường quay và phòng hậu kỳ
- Tổng kết sáu điểm cốt lõi của chương



Vang tích chập — Convolution Reverb





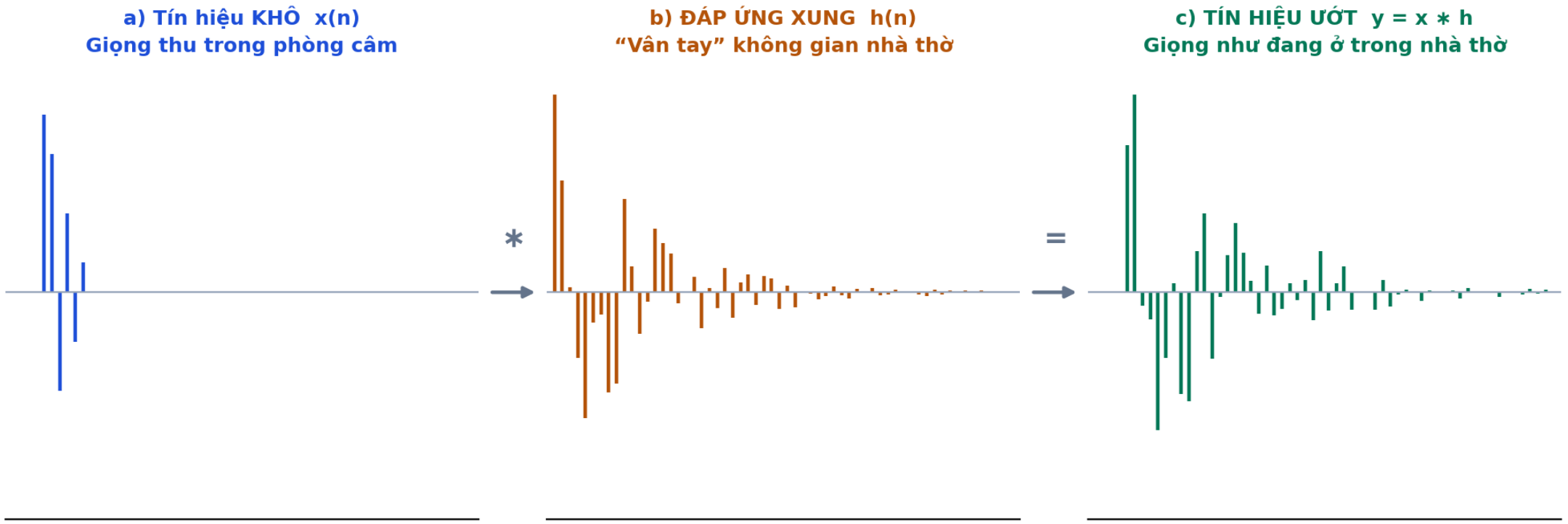
BƯỚC 2 — DÙNG TRONG PHÒNG HẬU KỲ

Nạp tệp IR vào plugin Convolution Reverb.
Mọi lời thoại đi qua đều nghe như thu ngay tại nhà thờ đó — không cần đưa đoàn phim tới hiện trường lần nữa.

Đo IR một lần, dùng được mãi mãi: mọi lời thoại tích chập với IR đó đều nghe như thu tại chính không gian ấy.

HÌNH 4.4

Phép tích chập trong giáo trình



Hình 4.4 — Tín hiệu khô kết hợp với đáp ứng xung không gian tạo ra tín hiệu mang đặc tính vang dội tự nhiên.

CÂU HỎI TẠI LỚP

Vì sao vang tích chập ngốn CPU nặng hơn nhiều so với reverb thuật toán thông thường?

Gợi ý: IR của một nhà thờ dài khoảng 3 giây ở 48 kHz. Với mỗi mẫu âm thanh đầu ra, máy phải làm bao nhiêu phép nhân?

Suy nghĩ 30 giây — rồi trả lời

GIẢI THÍCH

Đáp án — mỗi mẫu ra cần hàng trăm nghìn phép nhân

IR nhà thờ 3 giây ở 48 kHz có 144.000 mẫu. Theo định nghĩa tích chập, mỗi mẫu đầu ra là tổng của 144.000 phép nhân.

- Một giây âm thanh có 48.000 mẫu ⇒ **48.000 × 144.000 ≈ 6,9 tỷ phép nhân** cho mỗi giây, mỗi kênh.
- Reverb thuật toán chỉ dùng vài chục dòng trễ và bộ lọc ⇒ vài trăm phép tính mỗi mẫu.
- Giải pháp thực tế: dùng **định lý tích chập** — tích chập trong miền thời gian tương đương **phép nhân đơn giản trong miền tần số**. Chuyển sang miền tần số bằng FFT, nhân, rồi chuyển ngược.
- Đó là kỹ thuật **fast convolution** — và nó chỉ khả thi vì có thuật toán FFT đã học ở tiết 2. Ba công cụ của chương này nối vào nhau.

Mục 4.6 — Hướng dẫn thực hành

Ba bài Python: vẽ phổ vạch hài · vẽ spectrogram tín hiệu chirp · thiết kế bộ lọc Butterworth và thẩm âm

BÀI THỰC HÀNH 4.6.1

src/spectrum.py — vẽ phổ vạch hài Fourier

Sinh tín hiệu sóng sin thuần và sóng vuông, chạy FFT để so sánh trực quan **vân tay phổ** của hai loại sóng cơ bản theo đúng lý thuyết chuỗi hài.

- Sinh **sóng sin thuần 440 Hz** (nốt La chuẩn) và **sóng vuông 220 Hz** ở fs = 44.100 Hz.
- Dùng `np.fft.rfft(x)` cho tín hiệu số thực, chuẩn hóa chia `len(x)/2`; trục tần số bằng `np.fft.rfftfreq(len(x), 1/fs)`.
- Vẽ bằng `plt.stem` để thấy rõ các vạch phổ rời rạc.
- Kết quả cần quan sát: sóng sin cho **đúng một vạch tại 440 Hz**; sóng vuông 220 Hz cho **chuỗi hài LẺ**: 220, 660, 1.100, 1.540 Hz…
- Đây là bằng chứng thực nghiệm cho lý thuyết Fourier đã dựng dần ở tiết 2.

src/spectro.py — spectrogram của tín hiệu chirp

Tạo tín hiệu **quét tần số (Chirp)** chạy tuyến tính từ 200 Hz lên 4.000 Hz trong 3 giây, rồi vẽ spectrogram bằng STFT.

- `signal.chirp(t, f0=200, f1=4000, t1=3, method='linear')` — sinh tín hiệu quét tần.
- `signal.spectrogram(x, fs, nperseg=1024, noverlap=512)` — khung 1.024 mẫu, chồng lấn 50%.
- Chuyển năng lượng sang thang dB: `10 * np.log10(Sxx + 1e-12)` rồi vẽ bằng `plt.pcolormesh`.
- Kết quả: một **đường chéo sáng rực** đi từ (0 s, 200 Hz) lên (3 s, 4.000 Hz) — đúng dấu vết thứ ba đã học ở tiết 3.
- Bài tập mở rộng bắt buộc: đổi `nperseg` xuống **256** rồi lên **4.096**, quan sát độ phân giải thời gian và tần số đổi ngược chiều nhau.

src/filter_demo.py — thiết kế bộ lọc và thẩm âm

Dùng bộ lọc **Butterworth** để lọc thông thấp và thông cao trên tệp `input.wav`, xuất kết quả ra tệp để nghe và **thẩm âm sự thay đổi vật lý**.

- `sos = signal.butter(4, fc / (fs/2), btype='low', output='sos')` — bậc 4, tần số cắt **chuẩn hóa theo Nyquist fs/2**.
- `y = signal.sosfilt(sos, x)` — chạy bộ lọc. Dùng dạng `sos` thay vì `b, a` vì ổn định số học tốt hơn ở bậc cao.
- Xuất hai tệp: **`out_lowpass.wav`** (cắt tại 1.000 Hz) và **`out_highpass.wav`** (cắt tại 3.000 Hz).
- **Thẩm âm bằng tai nghe kiểm âm:** bản lowpass nghe **tối sầm, ngột ngạt, ấm** như âm thanh truyền qua bức tường gạch dày.
- Bản highpass nghe **mỏng dính, sắc bén, rào rạt**, xa xăm như tiếng còi từ loa phát thanh di động.

CHỐT LẠI MỤC 4.6

Bốn bài tập bối cảnh trường quay

Bài 1 — đọc phổ: ba biểu đồ không nhãn. Phổ A một đỉnh 1.000 Hz → **đơn âm chuẩn**. Phổ B chuỗi 150/300/450… → **âm có cao độ f_0 = 150 Hz**. Phổ C trải rộng → **tiếng động phi điều hòa**.

Bài 2 — đánh đổi: ở fs = 48 kHz, N = 512 cho Δf = 93,75 Hz / 10,7 ms; N = 4.800 cho 10 Hz / 100 ms; muốn phân biệt 5 Hz phải dùng N = 9.600, trả giá 200 ms.

Bài 3 — chọn bộ lọc: gió trầm → **HPF 80–100 Hz** · giọng điện thoại → **BPF 300–3.400 Hz** · ù 50 Hz → **Notch 50 Hz** · chống răng cưa → **LPF sát Nyquist**.

Bài 4 — tích chập: x = [2,0,1] ∗ h = [1,1] = **[2, 2, 1, 1]**. Hệ h = [1,1] là **y[n] = x[n] + x[n−1]** — bộ lọc thông thấp làm mịn đơn giản.

CHỐT LẠI MỤC 4.8

Tóm tắt Chương 4 — sáu điểm cốt lõi

1. Hai miền bổ sung: miền thời gian (biên độ theo thời gian) và miền tần số (năng lượng theo tần số) — hai cách nhìn cùng một mảng số.

2. Biến đổi Fourier: FFT phân rã tín hiệu thành tổng các sóng sin đơn tần. Đỉnh cao nhất của phổ chỉ rõ tần số mạnh nhất.

3. Đánh đổi độ phân giải: khung càng dài → tần số càng mịn nhưng thời gian càng nhòe. $\Delta f = fs/N$, $T = N/fs$, tích luôn bằng 1.

4. STFT và Spectrogram: trượt cửa sổ ngắn gối chồng lấn, chạy FFT từng khung. Ba trục: thời gian · tần số · năng lượng theo màu.

5. Bộ lọc số: LPF cắt treble · HPF cắt ù bass · BPF lọc thông thoại · Notch triệt ù 50 Hz. FIR không hồi tiếp ổn định tuyệt đối; IIR có hồi tiếp hiệu suất cao.

6. Tích chập và vang tích chập: in đáp ứng xung của một không gian lên tín hiệu thoại khô, tạo Convolution Reverb trung thực bậc nhất.

NHÌN LẠI TOÀN BỘ

Bốn chương — một hành trình liền mạch

Kết thúc Chương 4, sinh viên đã đi trọn con đường của một tín hiệu âm thanh, từ không khí tới thuật toán.

- **Chương 1 — trong không khí:** sóng dọc, bốn đại lượng của sóng, decibel, tai người và ISO 226.
- **Chương 2 — trong dây dẫn:** micro và loa, chuỗi thiết bị analog, bốn cấp mức, cân bằng XLR, SNR và dải động.
- **Chương 3 — trong bộ nhớ:** lấy mẫu và Nyquist, lượng tử hóa và độ sâu bit, dBFS, định dạng tệp, neural codec.
- **Chương 4 — trong thuật toán:** Fourier và FFT, STFT và spectrogram, bộ lọc số, tích chập và vang tích chập.
- Mọi phần mềm âm thanh — từ Audacity tới các plugin hàng nghìn đô — đều đứng trên đúng bốn tầng kiến thức này. Không có tầng thứ năm.

Hết Chương 4.

Các chương tiếp theo của học phần sẽ dùng đúng bộ công cụ này để lập trình hiệu ứng không gian, xử lý giọng nói và ứng dụng trí tuệ nhân tạo trong âm thanh.

Tài liệu tham khảo đầy đủ ở cuối Chương 4, trang 77 giáo trình.