MỤC LỤC

Chương 8. Hiệu ứng phần 1 5
8.1 Vị trí của chương trong cuốn sách 6
8.2 Đường bao mức và bộ dò mức 8
8.3 Bộ nén động 12
8.4 Bộ giới hạn và bộ cổng 19
8.5 Lọc lược — khi tín hiệu gặp lại chính nó 23
8.6 Hồi tiếp — một mũi tên đổi tất cả 26
8.7 Bậc thang độ trễ và trễ điều biến 28
8.8 Tự kiểm tra giữa chặng 31
8.9 Thực hành, bài tập và tổng kết 33

DANH MỤC HÌNH VẼ

Hình 8.1 Năm mục kỹ thuật của chương 8.....6
Hình 8.2 Chỗ mà chương 7 dừng lại — và là chỗ chương 8 bắt đầu.....7
Hình 8.3 Ba bộ dò mức khác nhau trên cùng một tín hiệu.....9
Hình 8.4 Hai hằng số thời gian nhìn trên một tín hiệu bậc thang.....10
Hình 8.5 Khoảng cách giữa hai kiểu dò phụ thuộc hẳn vào loại tư liệu.....11
Hình 8.6 Đường đặc tuyến tĩnh, và cùng thông tin ấy đọc theo lượng giảm gain.....13
Hình 8.7 Ba bề rộng đầu gối trên cùng một tỉ số 4:1.....14
Hình 8.8 Gain bù đưa đỉnh trở lại chỗ cũ, và khi ấy mới thấy độ to tăng.....15
Hình 8.9 Một bộ nén làm việc trên đoạn phỏng vấn hai người cách micro không đều..15
Hình 8.10 Thời gian tấn công quyết định gai có vượt qua hay không.....16
Hình 8.11 Nhả quá ngắn so với nhịp của tư liệu sinh ra hiện tượng "bơm".....16
Hình 8.12 Bốn tham số của bộ nén và câu hỏi mà mỗi tham số trả lời.....17
Hình 8.13 Kết quả của chuỗi nén, giới hạn và chuẩn hoá — và cái giá của nó.....18
Hình 8.14 Bộ giới hạn trên bản nhạc đã đẩy vượt trần 8 decibel.....20
Hình 8.15 Tăng tỉ số bộ nén lên cao không giữ được trần; nhìn trước thì có.....20
Hình 8.16 Đỉnh mẫu đúng trần −1,00 dBFS, mà đỉnh thật là +1,41 dBTP.....21
Hình 8.17 Bộ cổng hạ nền giữa các câu, tiếng nói không bị ảnh hưởng.....22
Hình 8.18 Hai lỗi điển hình của bộ cổng, đo được bằng số.....23
Hình 8.19 Sơ đồ khối và đáp ứng xung của bộ lọc lược FIR.....24
Hình 8.20 Trễ càng dài thì rãnh càng dày, và cách đọc độ trễ ra từ phổ đồ.....25
Hình 8.21 Dấu của g quyết định rãnh nằm ở đâu; độ lớn quyết định rãnh sâu bao nhiêu 25
Hình 8.22 Sơ đồ có hồi tiếp, và một xung vào cho ra cả một chuỗi vọng.....26
Hình 8.23 g quyết định đuôi vọng dài bao nhiêu và còn nghe được mấy tiếng.....27
Hình 8.24 Điều kiện ổn định, và hệ quả của hồi tiếp lên phổ.....28
Hình 8.25 Bậc thang độ trễ — cùng một khối, sáu hiệu ứng.....29
Hình 8.26 Flanger trên nhiễu trắng, và vì sao nó quét.....30
Hình 8.27 Cấu trúc bài tự kiểm tra giữa chặng — ba phần, mười điểm.....31
Hình 8.28 Chương 8 để lại gì cho các chương sau.....36

DANH MỤC BẢNG BIỂU

Bảng 8.1 Hai họ hiệu ứng dựng trên khối nào của chương 6.....7
Bảng 8.2 Bốn thứ gần giống mà thuộc chương khác.....8
Bảng 8.3 Ba đại lượng trước và sau chuỗi xử lý.....19
Bảng 8.4 Ba phần và nội dung được hỏi.....32
Bảng 8.5 Bảng ôn nhanh — bảy con số và một sơ đồ.....32
Bảng 8.6 Bảy bài tập của chương 8.....34
Bảng 8.7 Mười hàm mới của chương 8.....36

CHƯƠNG 8
HIỆU ỨNG PHẦN 1

Mục tiêu của chương

Sau chương này, người đọc phải làm được những việc sau.

- Giải thích được vì sao mọi bộ xử lý khoảng động đều bắt đầu bằng một bộ dò mức, và chọn được giữa dò theo đỉnh và dò theo RMS.
- Cài đặt được bộ dò mức hai hằng số thời gian, và giải thích được quy ước 63 phần trăm.
- Vẽ và đọc được đường đặc tuyến tĩnh của bộ nén, kể cả phần đầu gối mềm, và tính được gain bù.
- Đọc được đường giảm gain và từ đó nói được ngưỡng đang quá cao hay quá thấp.
- Giải thích được vì sao tỉ số nén cao *không* thay thế được bộ giới hạn, và vì sao bộ giới hạn cần nhìn trước.
- Chỉnh được bộ cổng đủ bốn tham số và tránh được hai lỗi điển hình: ngưỡng quá cao và thiếu thời gian giữ.
- Tính được vị trí các rãnh của bộ lọc lược theo cả hai chiều, và dùng được chiều ngược để chẩn đoán một bản thu nghe rỗng.
- Giải thích được điều kiện ổn định của vòng hồi tiếp, tính được RT60, và đặt được g theo RT60 mong muốn.
- Nêu được bậc thang độ trễ và giải thích được vì sao các mốc nằm ở đúng những chỗ ấy.

Hai họ hiệu ứng, một khối chung

Chương này mở phần hiệu ứng bằng hai họ hiệu ứng: xử lý khoảng động (mục 8.2 – 8.4) và hiệu ứng trễ (mục 8.5 – 8.7).

Cả hai đều dựng thẳng trên những khối đã viết ở chương 6 — xử lý theo khối có trạng thái, và phép trễ. Không có phép toán nào trong chương này khó hơn một phép nhân và một phép cộng.

Chương này cũng trả món nợ mà chương 7 để lại: bản nhạc chuẩn hoá không tới đích vì trần chặn. Câu trả lời bằng số nằm ở cuối mục 8.3.

Mục 8.8 là phần tự kiểm tra giữa chặng, nội dung từ chương 1 tới chương 7 — nên đọc ngay từ đầu chương.

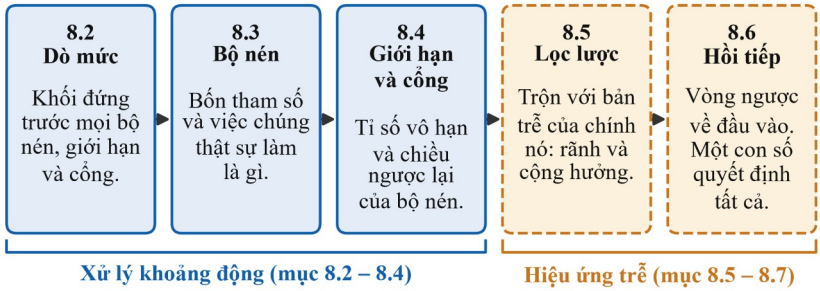
8.1 Vị trí của chương trong cuốn sách

8.1.1 Phần hiệu ứng bắt đầu từ đâu

Ba chương vừa qua đi theo một mạch rõ ràng. Chương 5 và chương 6 dạy cách *làm ra* một bản dựng chạy được. Chương 7 dạy cách *đo* xem bản dựng ấy đã nghe được chưa, và hạ nhiễu nền xuống.

Phần tiếp theo mang tên "Hiệu ứng" và hỏi câu tiếp theo: làm sao thay đổi tiếng một cách có chủ đích. Chương này mở phần ấy bằng hai họ hiệu ứng, và hai họ ấy không phải chọn ngẫu nhiên — chúng là hai họ dựng thẳng trên những khối đã viết ở chương 6.

- **Họ thứ nhất — xử lý khoảng động.** Bộ nén, bộ giới hạn, bộ cổng. Cả ba đều là một phép nhân gain thay đổi theo thời gian, và cả ba đều cần đúng một khối: bộ dò mức, tức là xử lý theo khối *có mang trạng thái* của mục 6.9.
- **Họ thứ hai — hiệu ứng trễ.** Lọc lược, flanger, chorus, tiếng vọng, hồi tiếp. Cả nhóm dựng trên đúng một khối: phép trễ của mục 6.4.

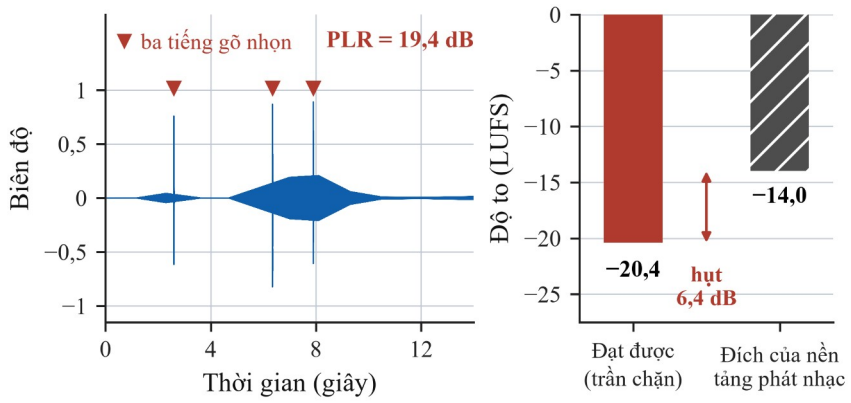


Hình 8.1 Năm mục kỹ thuật của chương 8

8.1.2 Câu hỏi mở đầu — món nợ của chương 7

Chương 7 mục 7.3.4 dừng lại ở một chỗ chưa xong. Bản nhạc cổ điển có đỉnh cao hơn độ to trung bình 19,4 decibel; khi chuẩn hoá về đích −14 LUFS của một nền tảng phát nhạc thì trần true peak chặn lại, và độ to dừng ở −20,4 LUFS — hụt 6,4 decibel so với đích.

Kết luận của chương 7 là: "vấn đề nằm ở bản dựng chứ không nằm ở phép chuẩn hoá; muốn to hơn thì phải nén nhẹ trước". Rồi chương 7 dừng, vì lúc ấy chưa có bộ nén.



(a) Bản nhạc cổ điển của chương 7 (b) Độ to sau chuẩn hoá

Hình 8.2 Chỗ mà chương 7 dừng lại — và là chỗ chương 8 bắt đầu

Chương này làm đúng việc ấy và đo lại. Cuối mục 8.3 sẽ có con số: chuỗi nén rồi giới hạn rồi chuẩn hoá đưa độ to từ −20,4 lên −15,5 LUFS ở cùng một trần −1 dBTP, tức là to hơn gần 5 decibel; cái giá là dải độ to thu từ 17,4 xuống 12,1 *lu*. Con số ấy là câu trả lời cho món nợ, và cũng là bài học chính của cả phần hiệu ứng: mọi hiệu ứng đều là một cuộc đánh đổi đo được.

8.1.3 Hai họ, một khối chung

Có một lý do sư phạm khiến hai họ hiệu ứng này nằm chung một chương, dù nghe thì chúng chẳng giống nhau chút nào.

Bảng 8.1 Hai họ hiệu ứng dựng trên khối nào của chương 6

Họ hiệu ứng	Khối của chương 6 mà nó dùng lại	Cái mới ở chương 8
Khoảng động (8.2 – 8.4)	Xử lý theo khối *có trạng thái* (mục 6.9). Đường bao mức chính là một biến trạng thái mang qua từng mẫu	Hai hằng số thời gian khác nhau cho chiều lên và chiều xuống; đường đặc tuyến tĩnh
Hiệu ứng trễ (8.5 – 8.7)	Phép trễ số nguyên mẫu và bộ đệm vòng (mục 6.4)	Trộn tín hiệu với chính bản trễ của nó; và đưa đầu ra vòng ngược về đầu vào

Một cách nói gọn về cả chương

Một bộ nén là một phép *nhân*, với hệ số nhân thay đổi theo mức của chính tín hiệu. Toàn bộ mục 8.2 – 8.4 là chuyện tính cái hệ số ấy cho khéo.

Một hiệu ứng trễ là một phép *cộng*, cộng tín hiệu với bản trễ của chính nó. Toàn bộ mục 8.5 – 8.7 là chuyện chọn độ trễ và chọn xem bản trễ lấy từ đầu vào hay từ đầu ra.

Không có phép toán nào trong chương này khó hơn một phép nhân và một phép cộng. Cái khó nằm ở chỗ chọn tham số, và chọn có căn cứ nghĩa là đo được.

8.1.4 Những gì chương này không làm

Bảng 8.2 Bốn thứ gần giống mà thuộc chương khác

Thứ dễ gộp nhầm	Vì sao không thuộc chương 8	Thuộc chương nào
Cân bằng tần số (EQ), bộ lọc wah	Chúng đổi phổ theo *tần số* chứ không theo mức. Cần thiết kế bộ lọc biquad, một chủ đề riêng	Chương 9
Bộ vang (reverb)	Bộ vang là hàng chục bộ lọc lược IIR và tán xạ chạy song song. Chương này chỉ dựng một khối duy nhất của nó	Chương 9
Nén đa băng, nén nối tiếp trong mastering	Cần đã có cả bộ lọc phân băng của chương 9 lẫn bộ nén của chương này	Chương 14
Khử nhiễu, trừ phổ	Bộ cổng ở mục 8.4 hạ nền *giữa* các câu; nó không động tới nền nằm dưới tiếng nói. Hai việc khác nhau	Chương 7 (đã học)

8.1.5 Cách học chương này

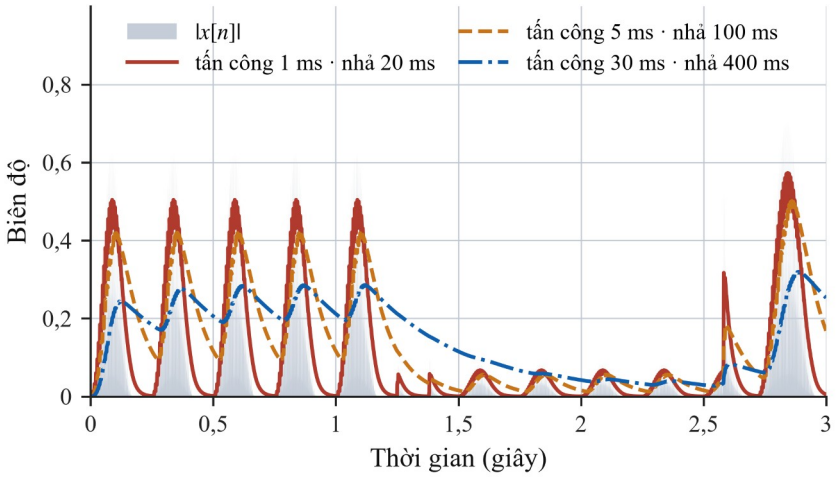
- **Nhìn đường giảm gain, đừng nhìn dạng sóng.** Mọi bộ nén, bộ giới hạn, bộ cổng đều hiển thị một đường "gain reduction". Người mới nhìn dạng sóng và đoán; người có nghề nhìn đường ấy và biết ngay bộ nén đang làm gì.
- **Mỗi tham số phải trả lời được một câu hỏi cụ thể.** Ngưỡng trả lời "từ mức nào"; tỉ số trả lời "mạnh tay bao nhiêu"; tấn công trả lời "có bắt được đầu tiếng không"; nhả trả lời "có bơm không". Chỉnh núm mà không trả lời được câu nào thì là mò.
- **Chương này có phần tự kiểm tra giữa chặng. Nội dung từ chương 1 tới chương 7. Cấu trúc** ở mục 8.8; nên đọc trước mục ấy ngay từ đầu chương.

8.2 Đường bao mức và bộ dò mức

8.2.1 Vì sao phải có bộ dò mức

Một bộ nén phải quyết định: ở mẫu thứ n này, hạ gain bao nhiêu. Muốn quyết định thì phải biết tín hiệu đang *to* cỡ nào. Nhưng "đang to cỡ nào" là một câu hỏi không có nghĩa nếu chỉ nhìn một mẫu — giá trị của một mẫu đơn lẻ dao động từ 0 tới đỉnh trong mỗi nửa chu kỳ.

Vì thế mọi bộ xử lý khoảng động đều bắt đầu bằng một khối chung: bộ dò mức, sinh ra một *đường bao* đi theo tín hiệu, lên nhanh và xuống chậm. Đường bao ấy mới là thứ mà đường đặc tuyến nén đọc vào.



Hình 8.3 Ba bộ dò mức khác nhau trên cùng một tín hiệu

Ba đường trên hình là ba bộ dò cùng thuật toán, chỉ khác hai con số. Đường đỏ bám sát từng âm tiết; đường xanh lam chỉ thấy xu hướng chung của cả câu. Bộ nén nào cũng chỉ phản ứng theo bức tranh mà bộ dò của nó nhìn thấy — nên trước khi trách bộ nén, phải xem bộ dò đang nhìn thấy gì.

8.2.2 Bộ lọc một cực — toàn bộ phép toán

Bộ dò mức là một bộ lọc thông thấp bậc một, viết bằng một dòng.

$$z[n] = \alpha z[n-1] + (1-\alpha)|x[n]|, \qquad \alpha = e^{-1/(\tau f_s)} \tag{8.1}$$

Trong đó z[n] là đường bao tại mẫu n, |x[n]| là trị tuyệt đối của mẫu vào, alpha là hệ số làm trơn nằm giữa 0 và 1, tau là hằng số thời gian tính bằng giây, và f dưới chỉ số s là tần số lấy mẫu.

Cách đọc công thức này rất trực tiếp: đường bao mới bằng một phần đường bao cũ cộng một phần giá trị mới. alpha gần 1 thì nó nhớ nhiều, đổi chậm; alpha gần 0 thì nó không giữ lại gì, đổi nhanh.

Quy ước 63 phần trăm — chỗ hay hiểu sai nhất

Hằng số thời gian tau *không* phải là thời gian để đường bao đi tới đích. Nó là thời gian để đường bao đi được 1 − 1/e, tức khoảng 63 phần trăm quãng đường còn lại.

Đường bao không bao giờ tới đích hoàn toàn — nó tiệm cận. Sau 3·tau nó mới đi được 95 phần trăm; sau 5·tau là 99 phần trăm.

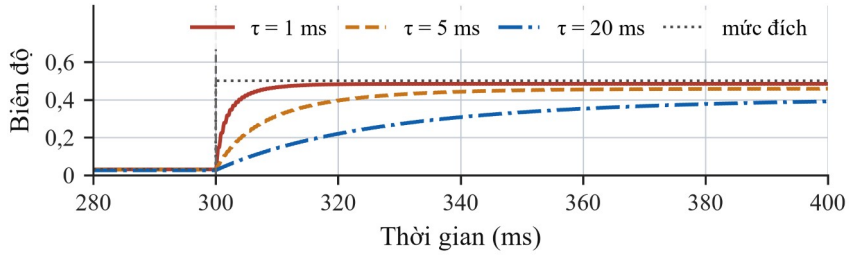
Vì thế một bộ nén ghi "tấn công 5 ms" vẫn còn đang hạ gain ở mili giây thứ mười lăm. Con số trên nhãn là hằng số thời gian, không phải thời gian hoàn tất.

8.2.3 Hai hằng số thời gian — tấn công và nhả

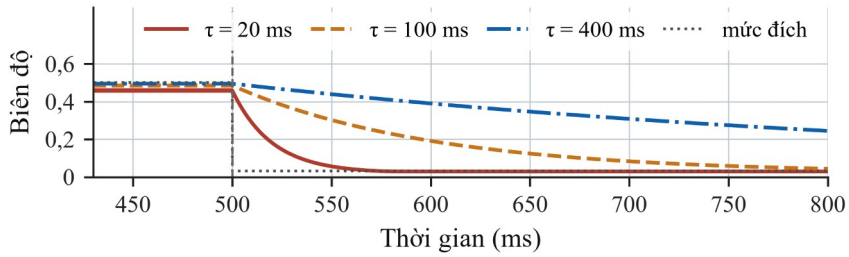
Nếu dùng một alpha duy nhất thì đường bao lên chậm bằng xuống, và nó sẽ bỏ lỡ mọi gai nhọn. Cái làm nên một bộ dò mức dùng được là chọn hệ số theo *chiều*: đi lên dùng một hệ số, đi xuống dùng hệ số khác.

$$\alpha = \alpha_{\text{tc}} \text{ khi } |x[n]| > z[n-1]; \qquad \alpha = \alpha_{\text{nhả}} \text{ khi } |x[n]| \le z[n-1] \tag{8.2}$$

Chỉ số dưới tc là tấn công (attack), nha là nhả (release). Thường thì tấn công ngắn hơn nhả nhiều lần: bắt gai thì phải nhanh, còn thả gain về thì phải từ tốn, nếu không sẽ nghe rõ tiếng gain nhấp nhô.



(a) Tấn công: mức nhảy lên tại 300 ms



(b) Nhả: mức tụt xuống tại 500 ms

Hình 8.4 Hai hằng số thời gian nhìn trên một tín hiệu bậc thang

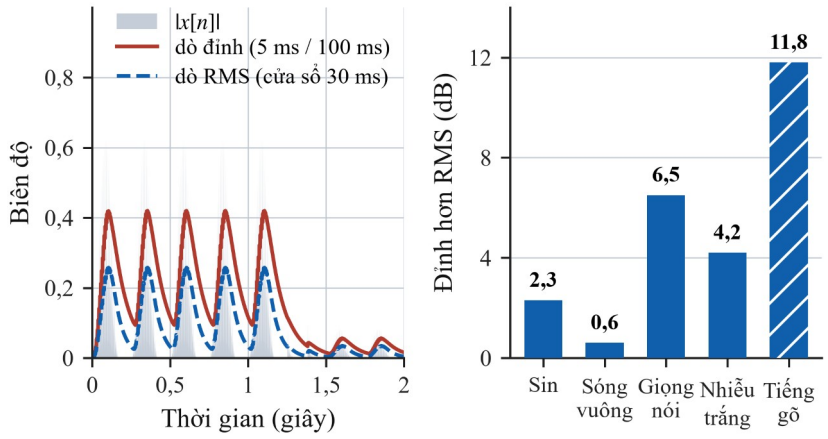
Tín hiệu bậc thang là công cụ chuẩn để nhìn hai hằng số này: mức nhảy từ −30 lên −6 dBFS ở mili giây thứ 300, rồi tụt xuống lại ở mili giây thứ 500. Đường chấm xám là mức đích; các đường màu là đường bao thật.

8.2.4 Dò theo đỉnh hay dò theo RMS

Có hai cách làm bộ dò, và chúng trả lời hai câu hỏi khác nhau.

- **Dò theo đỉnh.** Làm trơn trên |x[n]|, dùng hai hằng số thời gian như trên. Nó bám lấy từng gai. Đây là bộ dò của bộ giới hạn, vì bộ giới hạn có nhiệm vụ giữ trần cho từng mẫu.
- **Dò theo RMS.** Làm trơn trên x[n] bình phương với *một* hằng số thời gian duy nhất cho cả hai chiều, rồi lấy căn. Đây mới đúng là phép lấy trung bình bình phương, và nó gần với cảm nhận độ to hơn. Đây là bộ dò của bộ nén dùng cho nhạc.

> **Vì sao dò RMS không dùng hai hằng số**
>
> Nếu cho bộ dò RMS nhả chậm như bộ dò đỉnh thì nó cũng treo lại ở đỉnh và không còn là phép lấy trung bình nữa — hai bộ dò sẽ cho ra gần như cùng một đường.
>
> Đó là một lỗi cài đặt thật, và nó im lặng: mã chạy, kết quả có, chỉ là tham số "kiểu dò" mất tác dụng.
>
> Trong mô-đun của sách, hàm duong_bao_muc với kieu="rms" bỏ qua tham số tấn công và dùng nha_ms làm cửa sổ trung bình. Đó là cách viết trong Giannoulis và cộng sự (2012).



(a) Hai kiểu dò trên cùng tín hiệu (b) Khoảng cách giữa hai kiểu dò

Hình 8.5 Khoảng cách giữa hai kiểu dò phụ thuộc hẳn vào loại tư liệu

Cột phải của hình cho một kết quả đáng nhớ: khoảng cách giữa hai kiểu dò không phải một hằng số mà là một đặc tính của *tư liệu*. Sóng vuông gần như không chênh — 0,6 decibel; hình sin chênh 2,3 decibel, gần với con số lý thuyết 3,0; giọng nói chênh 6,5; còn tiếng gõ chênh tới 11,8 decibel.

Hệ quả thực hành: cùng một cài đặt bộ nén đặt lên hai loại tư liệu khác nhau sẽ cho hai kết quả khác hẳn, và đó không phải lỗi của bộ nén.

8.2.5 Cài đặt và kiểm

Liệt kê 8.1 Bộ dò mức — toàn bộ phần lõi

```
def he_so_thoi_gian(tau_ms, sr=44100):
    """alpha = exp(-1 / (tau * fs)), tau tính bằng giây."""
    if tau_ms <= 0:
        return 0.0                  # đổi tức thời
    return math.exp(-1.0 / (tau_ms / 1000.0 * sr))


def duong_bao_dinh(x, tan_cong_ms, nha_ms, sr=44100):
    a = np.abs(x)
    at = he_so_thoi_gian(tan_cong_ms, sr)
    re = he_so_thoi_gian(nha_ms, sr)
    ra = np.empty_like(a)
    z = 0.0                         # BIẾN TRẠNG THÁI
    for n in range(a.size):
        k = at if a[n] > z else re # chọn hệ số theo CHIỀU
        z = k * z + (1.0 - k) * a[n]
        ra[n] = z
    return ra
```

Biến z là biến trạng thái duy nhất của cả khối. Nếu xử lý tệp theo khối như mục 6.9 thì phải mang z qua giữa các khối; bỏ sót nó thì mỗi đầu khối đường bao lại bắt đầu từ 0, và kết quả là một tiếng "tách" đúng ở biên mỗi khối.

Bốn phép kiểm cho bộ dò mức

Đảo ngược: thoi_gian_tu_he_so(he_so_thoi_gian(5,0)) phải trả về đúng 5,0 mili giây.

Dò RMS trên hình sin biên độ 1 phải cho đúng 0,707 — mô-đun của sách cho sai số dưới 0,005.

Dò đỉnh luôn nằm *trên* dò RMS trên mọi tư liệu. Nếu có chỗ ngược lại thì bộ dò RMS đang dùng nhầm hai hằng số thời gian.

Khoảng cách đỉnh trừ RMS phải phụ thuộc tư liệu: sóng vuông dưới 1,5 decibel, tiếng gõ trên 8 decibel. Nếu hai con số ấy bằng nhau thì một trong hai bộ dò đang sai.

8.3 Bộ nén động

8.3.1 Bộ nén làm gì — nói bằng một câu

Bộ nén hạ gain khi tín hiệu vượt một ngưỡng, và để yên khi tín hiệu nằm dưới ngưỡng. Hệ quả là khoảng cách giữa chỗ to nhất và chỗ nhỏ nhất thu hẹp lại — từ đó có cái tên "nén khoảng động".

Sơ đồ của một bộ nén có bốn khối nối tiếp, và đọc đúng thứ tự ấy thì mọi chuyện sáng ra.

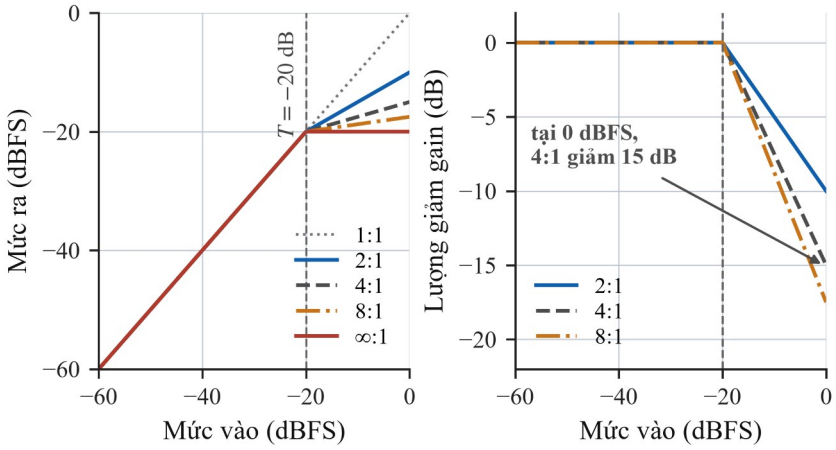
- **Dò mức.** Mục 8.2 vừa xong. Ra một đường bao.
- **Đổi sang decibel.** Vì đường đặc tuyến của bộ nén là một đường thẳng gãy khúc trong thang decibel, không phải trong thang tuyến tính.
- **Đường đặc tuyến tĩnh.** Mức vào bao nhiêu decibel thì mức ra bấy nhiêu — một hàm số thuần tuý, không nhớ gì cả.
- **Làm trơn trên gain, rồi nhân.** Đây là chỗ hai hằng số thời gian thật sự được áp vào. Chi tiết ở mục 8.3.6.

8.3.2 Ngưỡng và tỉ số — đường đặc tuyến tĩnh

Đường đặc tuyến tĩnh là hai đoạn thẳng nối nhau tại ngưỡng.

$$y_{\mathrm{dB}} = x_{\mathrm{dB}} \ \text{khi}\ x_{\mathrm{dB}} \le T\,; \qquad y_{\mathrm{dB}} = T + \frac{x_{\mathrm{dB}} - T}{R} \ \text{khi}\ x_{\mathrm{dB}} > T \tag{8.3}$$

Trong đó x chỉ số dB là mức vào tính bằng decibel, y chỉ số dB là mức ra, T là ngưỡng (threshold) và R là tỉ số nén (ratio). Dưới ngưỡng, đường đi thẳng — một decibel vào cho một decibel ra, bộ nén không làm gì cả. Trên ngưỡng, mỗi R decibel vào chỉ còn cho một decibel ra.



(a) Đường đặc tuyến tĩnh (b) Đọc theo lượng giảm gain

Hình 8.6 Đường đặc tuyến tĩnh, và cùng thông tin ấy đọc theo lượng giảm gain

Hai hình là một. Hình (a) vẽ mức ra theo mức vào; hình (b) vẽ hiệu của chúng, tức là lượng gain bị hạ. Người dùng bộ nén nhìn hình (b), vì đó chính là đường mà mọi phần mềm hiển thị.

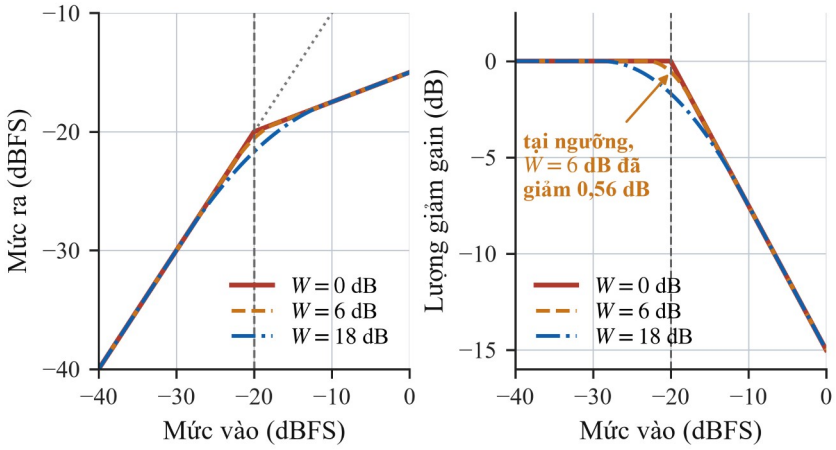
> **Tỉ số vô hạn chính là bộ giới hạn**
>
> Khi R tiến ra vô hạn, đoạn trên ngưỡng thành một đường nằm ngang: mọi mức vào trên ngưỡng đều cho ra đúng bằng ngưỡng.
>
> Đó là định nghĩa của bộ giới hạn. Nói cách khác, bộ giới hạn không phải một khối khác — nó là bộ nén ở một đầu mút của tham số R.
>
> Trong thực tế người ta coi từ khoảng 10:1 trở lên là đã sang lãnh địa bộ giới hạn. Nhưng như mục 8.4 sẽ cho thấy, tỉ số cao *không* đủ để giữ trần — còn thiếu một thứ nữa.

8.3.3 Đầu gối mềm

Đường gãy góc ở đúng ngưỡng có một nhược điểm nghe được: một tín hiệu dao động quanh ngưỡng sẽ bị bật tắt nén liên tục, và tai nghe ra tiếng "lép bép". Cách chữa là bo tròn chỗ gãy trong một khoảng W decibel quanh ngưỡng.

$$y_{\mathrm{dB}} = x_{\mathrm{dB}} + \left(\frac{1}{R} - 1\right)\frac{(x_{\mathrm{dB}} - T + W/2)^2}{2W}, \qquad |x_{\mathrm{dB}} - T| \le \frac{W}{2} \tag{8.4}$$

W là bề rộng đầu gối (knee width). Trong khoảng từ T − W/2 tới T + W/2, đường cong đi theo một parabol nối trơn hai đoạn thẳng; ngoài khoảng ấy công thức (8.3) vẫn đúng.



(a) Mức ra, tỉ số 4:1 (b) Lượng giảm gain

Hình 8.7 Ba bề rộng đầu gối trên cùng một tỉ số 4:1

Cái giá của đầu gối mềm — đo được

Đầu gối mềm bắt đầu nén *trước* ngưỡng đã đặt. Với T = −20 dB, R = 4 và W = 6 dB, ở đúng ngưỡng đã giảm 0,56 decibel, trong khi đầu gối cứng thì chưa giảm gì.

Đó là lý do một bộ nén đầu gối mềm hay có vẻ "nén nhiều hơn dự đoán", và cũng là lý do khi so hai bộ nén khác hãng thì phải hỏi bề rộng đầu gối trước khi so ngưỡng.

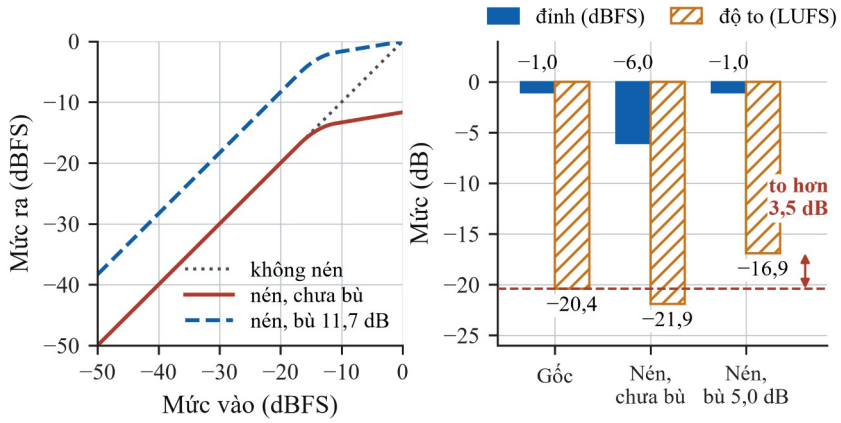
Quy tắc thực hành: đầu gối rộng cho giọng nói và nhạc nền; đầu gối cứng khi cần kiểm soát dứt khoát một dải mức cụ thể.

8.3.4 Gain bù — vì sao phải có

Nén xong thì mọi thứ nhỏ đi, nên nghe qua có vẻ kém hơn. Người mới thường dừng ở đó và kết luận bộ nén làm sai lệch tiếng. Kết luận ấy sai vì phép so sánh không công bằng: hai đoạn khác mức thì đoạn to hơn bao giờ cũng nghe hay hơn.

Gain bù (make-up gain) là một hằng số decibel cộng vào sau khi nén, để đưa mức trở lại chỗ cũ. Công thức chuẩn tính lượng bù sao cho một tín hiệu chạm 0 dBFS trở lại đúng 0 dBFS.

$$g_{\text{bù}} = -\left(T + \frac{0 - T}{R}\right) = T\left(\frac{1}{R} - 1\right) \tag{8.5}$$



(a) Đường đặc tuyến (b) Số đo trên bản nhạc cổ điển

Hình 8.8 Gain bù đưa đỉnh trở lại chỗ cũ, và khi ấy mới thấy độ to tăng

Hình (b) là kết quả đo thật trên bản nhạc cổ điển của chương 7, với bộ nén ngưỡng −14 dB, tỉ số 6:1, tấn công 0,5 ms, nhả 20 ms, đầu gối 6 dB. Đỉnh đi từ −1,0 xuống −6,0 rồi về đúng −1,0 dBFS sau khi bù 5,0 decibel. Trong khi đó độ to đi từ −20,4 lên −16,9 LUFS: to hơn 3,5 decibel ở *cùng* một mức đỉnh. Đó là toàn bộ lý do người ta dùng bộ nén.

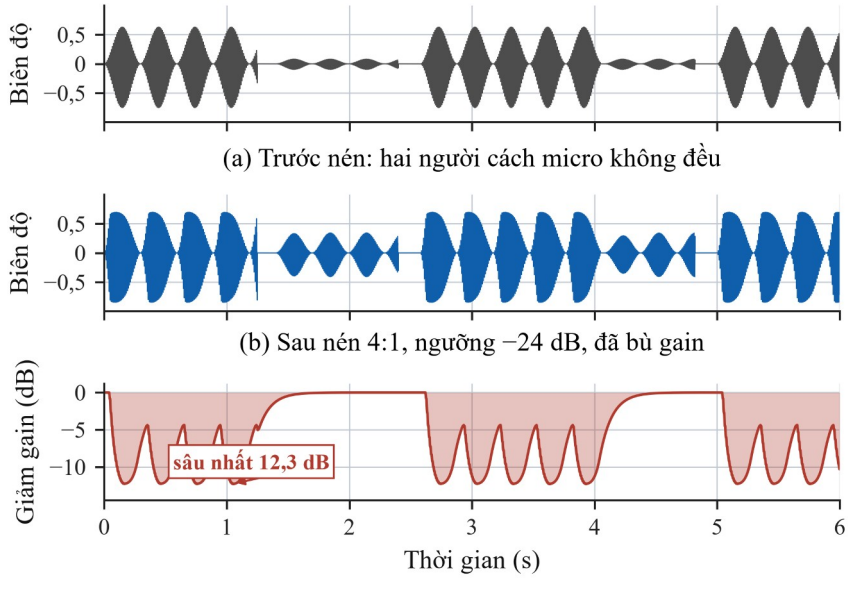
Công thức gain bù chỉ đúng cho tín hiệu chạm trần

Công thức (8.5) cho 11,7 decibel với T = −14 và R = 6. Nhưng trên tư liệu thật ở hình trên, lượng bù đúng chỉ là 5,0 decibel.

Lý do: công thức giả định tín hiệu chạm 0 dBFS. Bản nhạc thật đỉnh ở −1 dBFS và phần lớn thời gian nằm thấp hơn ngưỡng nhiều, nên nó không bị hạ nhiều như công thức giả định.

Cách làm đúng trong thực hành: *Đo* lượng đỉnh tụt sau khi nén, rồi bù đúng bằng ngần ấy. Dùng thẳng công thức sẽ đẩy tín hiệu vọt quá trần và cắt méo.

8.3.5 Bốn tham số và đọc đường giảm gain



(a) Trước nén: hai người cách micro không đều

(b) Sau nén 4:1, ngưỡng −24 dB, đã bù gain

(c) Đường giảm gain

Hình 8.9 Một bộ nén làm việc trên đoạn phỏng vấn hai người cách micro không đều

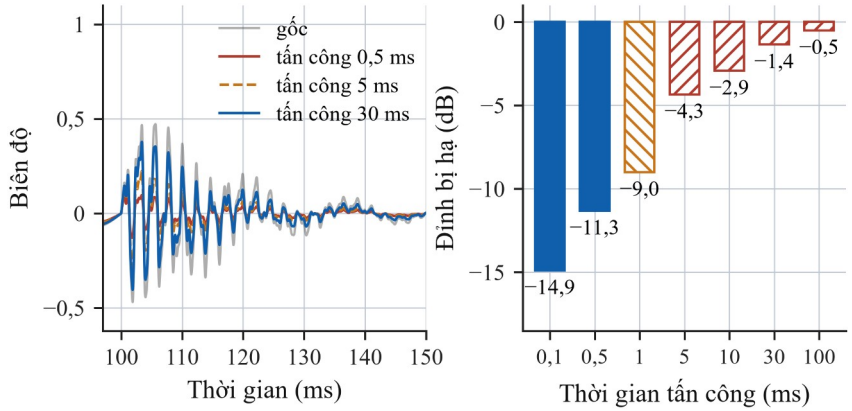
Hàng dưới của hình là thứ mà mọi bộ nén đều hiển thị và người mới hay bỏ qua. Nó cho biết bộ nén đang làm việc ở chỗ nào và nhiều tới đâu — ở đây sâu nhất là 12,3 decibel, đúng vào những câu của người ngồi gần micro.

Hai quy tắc đọc đường giảm gain

Nếu đường này gần như luôn bằng không thì ngưỡng đặt quá cao — bộ nén hầu như không làm gì.

Nếu nó không bao giờ về không thì ngưỡng quá thấp — bộ nén đang nén cả những chỗ không cần, và kết quả là tiếng bẹt đều một mức.

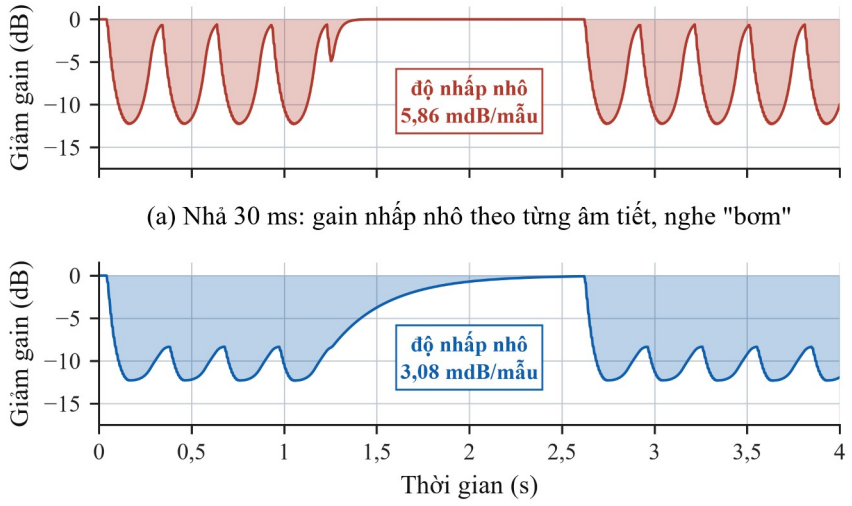
Đích cần nhắm: đường chạm đáy ở những chỗ *to nhất* và về không ở những chỗ bình thường.



(a) Ba thời gian tấn công (b) Mức hạ đỉnh

Hình 8.10 Thời gian tấn công quyết định gai có vượt qua hay không

Tấn công 0,1 mili giây hạ đỉnh của tiếng gõ tới 14,9 decibel; tấn công 100 mili giây chỉ hạ 0,5 decibel — gai vượt qua gần như nguyên vẹn. Cả hai đều là lựa chọn hợp lệ, tuỳ mục đích: tấn công nhanh bắt được gai nhưng làm mất cái "cộp" của trống, tiếng nghe bẹt đi; tấn công chậm giữ được đầu tiếng nhưng khi ấy bộ nén không giúp gì cho việc hạ đỉnh — đó là việc của bộ giới hạn.



(a) Nhả 30 ms: gain nhấp nhô theo từng âm tiết, nghe "bơm"

(b) Nhả 300 ms: gain đổi chậm, tai khó nhận ra

Hình 8.11 Nhả quá ngắn so với nhịp của tư liệu sinh ra hiện tượng "bơm"

"Bơm" (pumping) là tên gọi của hiện tượng nền nhạc phồng lên xẹp xuống theo từng âm tiết. Nguyên nhân luôn là thời gian nhả quá ngắn so với nhịp của tư liệu. Cách đo trong hình: lấy độ lệch chuẩn của biến thiên gain giữa hai mẫu liền nhau, tính bằng phần nghìn decibel trên mẫu. Nhả 30 mili giây cho 5,85; nhả 300 mili giây cho 3,09 — gần một nửa.



Hình 8.12 Bốn tham số của bộ nén và câu hỏi mà mỗi tham số trả lời

8.3.6 Cài đặt — và một chi tiết quyết định chất lượng

Có hai cách ghép các khối, và chúng cho kết quả khác nhau rõ rệt.

– **Làm trơn trên mức rồi mới tra đặc tuyến.** Cách này đơn giản hơn nhưng làm hai hằng số thời gian phụ thuộc vào tỉ số nén: đổi R thì tính cách của bộ nén cũng đổi theo, dù không đụng tới tấn công và nhả.

– **Làm trơn trên gain, sau khi đã tra đặc tuyến.** Cách này tách bạch: đặc tuyến quyết định nén *bao nhiêu*, hai hằng số thời gian quyết định đổi *nhanh chậm* thế nào. Đây là cách viết trong Giannoulis và cộng sự, và là cách mô-đun của sách dùng.

Liệt kê 8.2 Bộ nén động — sơ đồ bốn khối viết thẳng ra mã

```
def nen_dong(x, nguong_db=-20.0, ti_so=4.0, tan_cong_ms=5.0,
             nha_ms=100.0, dau_goi_db=0.0, bu_db=None, sr=44100):
    # 1. dò mức → 2. đổi sang decibel
    env = duong_bao_muc(x, tan_cong_ms, nha_ms, sr, "dinh")
    muc = lin_to_db(np.maximum(env, 1e-9))

    # 3. đường đặc tuyến TĨNH → lượng giảm gain mong muốn
    gr_dich = do_giam_gain_db(muc, nguong_db, ti_so, dau_goi_db)

    # 4. làm mượt trên GAIN, không phải trên mức
    at = he_so_thoi_gian(tan_cong_ms, sr)
    re = he_so_thoi_gian(nha_ms, sr)
    gr = np.empty_like(gr_dich)
    z = 0.0
    for n in range(gr_dich.size):
        v = gr_dich[n]
```

```
        k = at if v < z else re    # gain ĐI XUỐNG là tấn công
        z = k * z + (1.0 - k) * v
        gr[n] = z

    if bu_db is None:
        bu_db = -do_giam_gain_db(0.0, nguong_db, ti_so, dau_goi_db)
    return x * db_to_lin(gr + bu_db), gr
```

Một dòng dễ viết ngược

Trong vòng lặp làm trơn gain, điều kiện chọn hệ số là v < z, tức là gain *đi xuống* thì dùng hệ số tấn công.

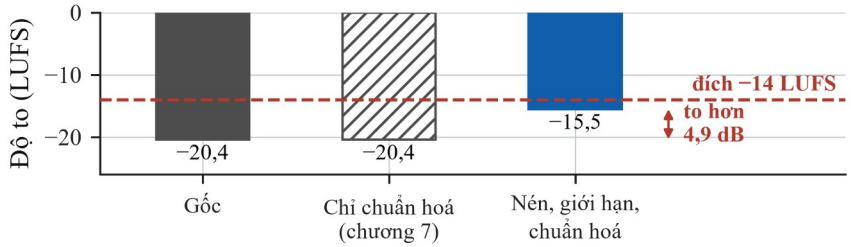
Viết ngược dấu thì mã vẫn chạy, vẫn ra tệp, chỉ là bộ nén hạ gain chậm và thả gain nhanh — đúng ngược với mọi bộ nén trên đời. Triệu chứng: gai đi qua hết mà tiếng vẫn bơm.

Phép kiểm: cho tín hiệu bậc thang từ −30 lên −6 dBFS, đo lượng giảm gain ở mili giây thứ 1, thứ 20 và ở trạng thái ổn định. Với tấn công 5 ms phải thấy 0,02 → 8,5 → 9,4 decibel. Nếu con số đầu đã gần con số cuối thì dấu đang ngược.

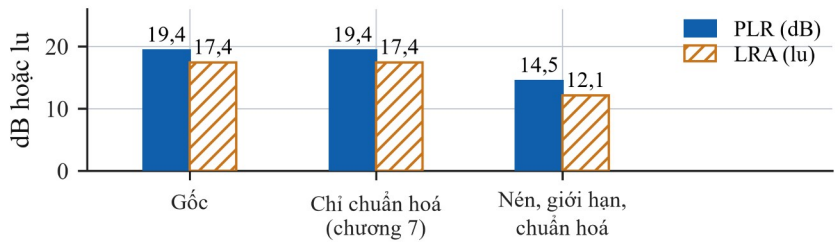
8.3.7 Trả món nợ của chương 7

Bây giờ đã đủ công cụ để làm việc mà chương 7 chưa hoàn thành. Chuỗi xử lý gồm bốn bước, và thứ tự không đổi được.

- Nén nhẹ — ngưỡng −24 dB, tỉ số 4:1, tấn công 10 ms, nhả 150 ms, đầu gối 6 dB.
- Đẩy mức lên 8 decibel — cố tình cho vượt trần.
- Giới hạn ở trần −1 dBFS với cửa sổ nhìn trước 5 ms (mục 8.4).
- Chuẩn hoá độ to về −14 LUFS với trần −1 dBTP (mục 7.3).



(a) Cùng trần −1 dBTP: độ to đạt được



(b) Cái giá phải trả: PLR và LRA

Hình 8.13 Kết quả của chuỗi nén, giới hạn và chuẩn hoá — và cái giá của nó

Bảng 8.3 Ba đại lượng trước và sau chuỗi xử lý

Đại lượng	Bản gốc	Chỉ chuẩn hoá (chương 7)	Nén → giới hạn → chuẩn hoá
Độ to tích hợp (LUFS)	−20,4	−20,4	−15,5
Hệ số đỉnh trên độ to PLR (dB)	19,4	19,4	14,5
Dải độ to LRA (*lu*)	17,4	17,4	12,1
Đỉnh thật (dBTP)	−1,0	−1,0	−1,0

Đọc bảng này cho kỹ — nó là cả phần hiệu ứng thu nhỏ

Cột giữa cho thấy phép chuẩn hoá đứng riêng *không* làm được gì: PLR không đổi, nên trần vẫn chặn ở đúng chỗ cũ. Chương 7 nói đúng.

Cột phải: PLR tụt từ 19,4 xuống 14,5 decibel, và nhờ đó độ to đạt được tăng gần 5 decibel ở *cùng* một trần −1 dBTP.

Cái giá nằm ở dòng LRA: dải độ to thu từ 17,4 xuống 12,1 *lu*. Với nhạc pop đó là đánh đổi bình thường; với nhạc cổ điển thì nhiều người nghe sẽ tiếc.

Không có con số nào ở đây là "đúng". Việc của người kỹ thuật là đặt cả bốn dòng lên bàn và để người quyết định nhìn thấy cái giá trước khi quyết.

Một kết quả ngược với trực giác — và nó thật

Nếu chỉ nén 4:1 với tấn công 10 mili giây mà *không* giới hạn, PLR không giảm mà *tăng*: từ 19,4 lên 22,9 decibel.

Lý do: tấn công 10 mili giây quá chậm để bắt ba tiếng gõ, nên chúng vượt qua nguyên vẹn, trong khi phần thân bản nhạc bị hạ đều. Đỉnh giữ nguyên mà độ to trung bình giảm — PLR vì thế lớn hơn.

Đây là chỗ nhiều người kết luận "bộ nén không có tác dụng" rồi tăng tỉ số lên cao hơn nữa. Cách chữa đúng là thêm một bộ *giới hạn* phía sau, chứ không phải nén mạnh tay hơn.

8.4 Bộ giới hạn và bộ cổng

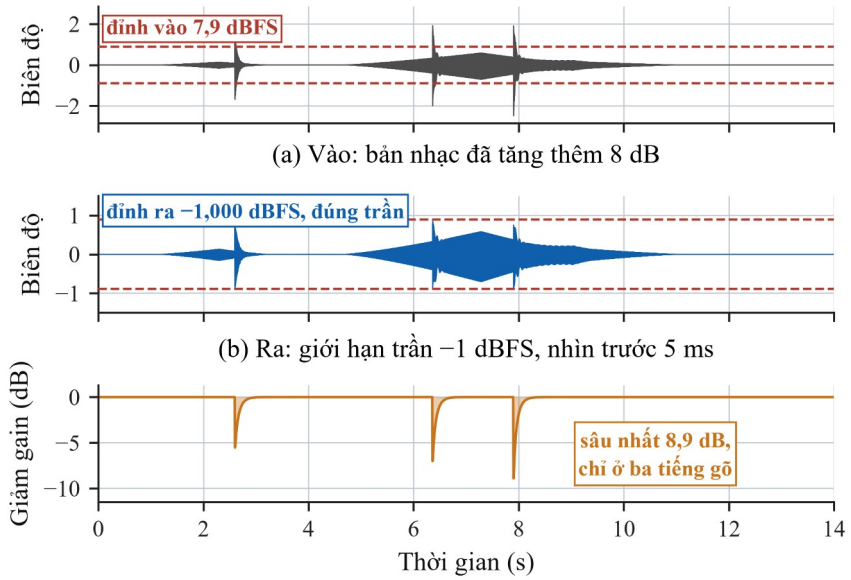
8.4.1 Bộ giới hạn khác bộ nén ở chỗ nào

Bộ giới hạn là bộ nén với tỉ số vô hạn — nhưng chỉ có thế thì chưa đủ. Điểm khác biệt thật nằm ở chỗ khác: bộ giới hạn *nhìn trước*.

Bộ nén phản ứng sau khi mức đã vượt ngưỡng, nên gai đầu tiên luôn vượt qua — dù tấn công có nhanh tới đâu, nó vẫn cần một khoảng thời gian khác không. Bộ giới hạn giải bài toán ấy bằng cách trễ tín hiệu lại một chút, và trong khoảng trễ đó nó đã kịp tính xong đường gain trước khi gai tới nơi.

$$g[n] = \min_{n \le k \le n+L} \min\left(1, \frac{A_{\text{trần}}}{|x[k]|}\right); \qquad y[n] = g[n]\,x[n-L] \tag{8.6}$$

A chỉ số tran là biên độ trần tính theo thang tuyến tính, L là số mẫu của cửa sổ nhìn trước. Phép cực tiểu trượt trên cửa sổ [n, n+L] bảo đảm gain tại mẫu n đã tính đến mọi mẫu sẽ tới trong L mẫu sắp tới; còn tín hiệu thì bị trễ đúng L mẫu để hai bên khớp nhau.



(a) Vào: bản nhạc đã tăng thêm 8 dB

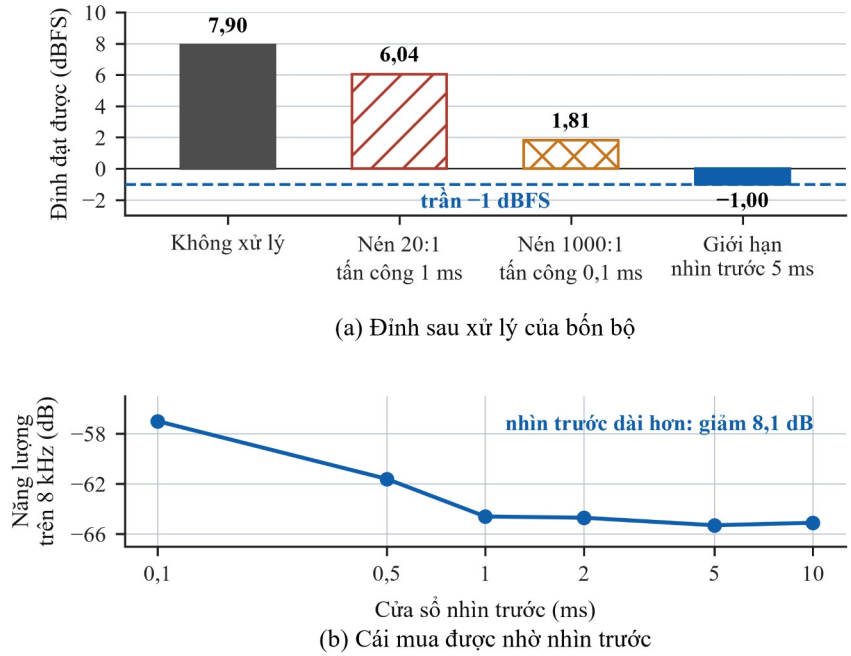
(b) Ra: giới hạn trần −1 dBFS, nhìn trước 5 ms

(c) Đường giảm gain của bộ giới hạn

Hình 8.14 Bộ giới hạn trên bản nhạc đã đẩy vượt trần 8 decibel

Đỉnh vào 7,9 dBFS, đỉnh ra −1,000 dBFS — đúng trần, không hơn một phần nghìn decibel. Và đường giảm gain chỉ chạm đáy ở ba tiếng gõ, sâu nhất 8,9 decibel; phần còn lại của bản nhạc không bị đụng tới. Đó là hình mẫu của một bộ giới hạn dùng đúng: nó làm rất ít việc, nhưng chỗ nào cần thì tuyệt đối.

8.4.2 Vì sao tỉ số cao không thay thế được nhìn trước



(a) Đỉnh sau xử lý của bốn bộ

(b) Cái mua được nhờ nhìn trước

Hình 8.15 Tăng tỉ số bộ nén lên cao không giữ được trần; nhìn trước thì có

Đây là chỗ hay nhầm nhất của cả chương. Cùng một tín hiệu vượt trần 8,9 decibel, bộ nén tỉ số 20:1 tấn công 1 mili giây vẫn để đỉnh vượt qua ở 6,04 dBFS; tăng lên 1000:1 với tấn công 0,1 mili giây cũng chỉ xuống được 1,81 dBFS. Bộ giới hạn cho đúng −1,00.

Hình (b) trả lời câu hỏi tiếp theo: cửa sổ nhìn trước dài thêm thì mua được gì. Không phải trần chặt hơn — trần đã tuyệt đối rồi ngay cả với 0,1 mili giây. Cái mua được là đường gain đổi ở chỗ tín hiệu còn nhỏ, nên ít sinh vụn cao tần hơn: năng lượng trên 8 kHz giảm 8,1 decibel khi đi từ 0,1 lên 10 mili giây.

Cái giá của nhìn trước là độ trễ

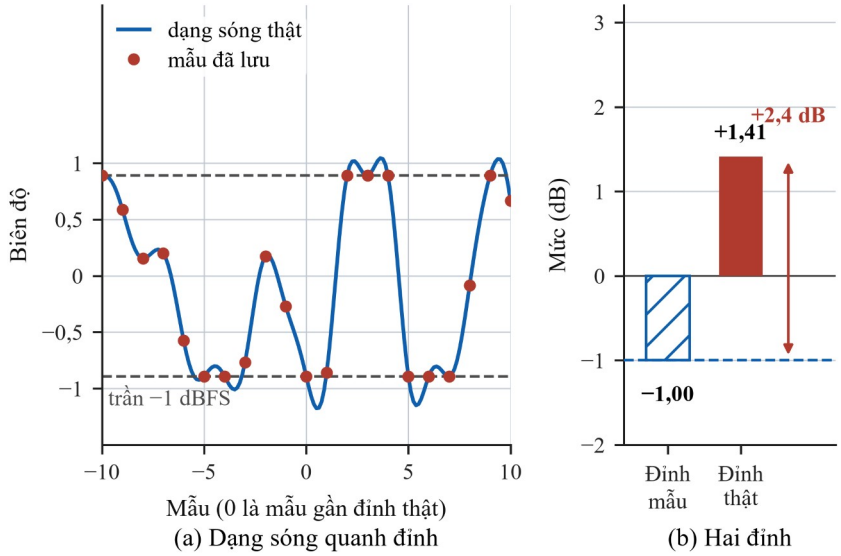
Bộ giới hạn nhìn trước 5 mili giây làm cả tín hiệu chậm đi đúng 5 mili giây. Riêng độ trễ ấy thì không nghe ra.

Nhưng độ trễ *cộng dồn* khi ghép nhiều khối nối tiếp, và nó thành vấn đề ở hai chỗ: khi trộn với một đường tín hiệu song song không qua bộ giới hạn (sinh ra đúng bộ lọc lược của mục 8.5), và khi xử lý thời gian thực cho người biểu diễn tự nghe lại tiếng của họ.

Vì thế mọi phần mềm dựng đều có chức năng "bù trễ" tự động, và khi tự viết chuỗi xử lý thì phải tự cộng lấy.

8.4.3 Trần mẫu không phải trần thật — nối lại với chương 7

Bộ giới hạn bảo đảm không *mẫu* nào vượt trần. Nó không bảo đảm được đường cong đi qua các mẫu ấy cũng không vượt — đúng cái đỉnh liên mẫu đã học ở mục 7.4.



Hình 8.16 Đỉnh mẫu đúng trần −1,00 dBFS, mà đỉnh thật là +1,41 dBTP

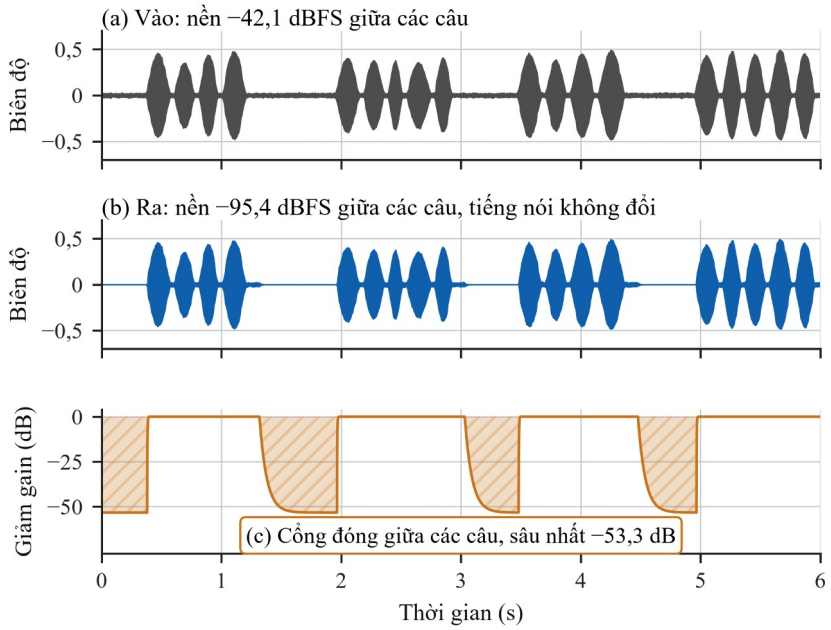
Trên tư liệu gắt — nhiễu giới hạn băng đã cắt méo — bộ giới hạn giữ đỉnh mẫu ở đúng −1,00 dBFS, trong khi đỉnh thật lên tới +1,41 dBTP, tức là vượt trần 2,4 decibel. Bộ giải mã có tổn hao của nền tảng phát sẽ cắt méo đúng ở đó.

– **Cách chữa thứ nhất:** đặt trần theo dBTP thay vì dBFS, tức là dùng phép đo true peak của mục 7.4 làm đầu vào cho bộ giới hạn.

– **Cách chữa thứ hai:** chạy bộ giới hạn ở tần số lấy mẫu gấp bốn lần rồi hạ về — cách mà các bộ giới hạn thương mại gọi là "true peak limiting".
– **Cách chữa thứ ba, rẻ nhất:** hạ trần xuống −1,5 hoặc −2 dBFS và chấp nhận mất chừng ấy độ to. Với phần lớn công việc, đây là lựa chọn đúng.

8.4.4 Bộ cổng — hình ảnh soi gương của bộ nén

Bộ cổng (gate) hạ gain khi tín hiệu nằm *dưới* ngưỡng. Nó ngược chiều với bộ nén về mọi mặt, và dùng để hạ nền giữa các câu nói.



Hình 8.17 Bộ cổng hạ nền giữa các câu, tiếng nói không bị ảnh hưởng

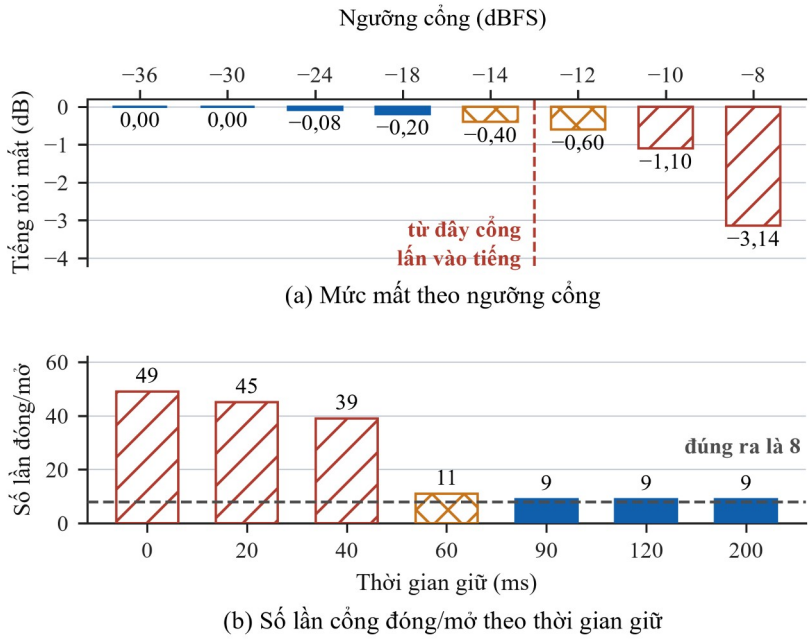
Nền đi từ −42,1 xuống −95,4 dBFS ở những chỗ không có tiếng, còn các câu nói thì không mất một phần trăm decibel nào.

> **Cổng không phải là khử nhiễu**
>
> Bộ cổng chỉ hạ nền ở những chỗ *không có tiếng*. Nền nằm dưới tiếng nói vẫn còn nguyên — cổng không biết tách hai thứ ấy ra.
>
> Vì thế cổng không thay thế phép trừ phổ của mục 7.5. Hai thứ giải hai bài toán khác nhau và trong thực tế hay dùng cùng nhau: trừ phổ hạ nền chung, cổng dọn nốt những khoảng lặng.
>
> Có một cái lợi phụ ít người nói tới: cổng không sinh nhiễu nhạc, vì nó không đụng gì tới phổ.

Bộ cổng có bốn tham số, ba trong đó giống bộ nén. Tham số thứ tư là thứ riêng của nó và cũng là thứ hay bị bỏ sót.

– **Ngưỡng.** Dưới mức này thì đóng cổng.
– **Tỉ số.** Đóng mạnh tay tới đâu. Tỉ số vô hạn là cắt hẳn về im lặng; tỉ số vừa phải chỉ hạ nền xuống vài decibel — khi ấy người ta gọi nó là bộ nới (expander).
– **Tấn công và nhả.** Như bộ nén.

– **Thời gian giữ (hold).** Cổng vẫn mở thêm ngần ấy sau khi tín hiệu đã tụt dưới ngưỡng. Không có nó thì cổng đóng mở lập bập ngay trong một từ.



Hình 8.18 Hai lỗi điển hình của bộ cổng, đo được bằng số

Hình (a): ngưỡng đặt quá cao thì cổng cắt vào tiếng. Với đoạn thử trong hình, tới ngưỡng −12 dBFS đã mất 0,60 decibel, tới −8 dBFS thì mất 3,14 decibel — nghe ra là đuôi từ bị cụt và phụ âm nhẹ bị mất.

Hình (b): không có thời gian giữ thì cổng đóng mở 49 lần trong khi đúng ra chỉ nên 8 lần — bốn câu, mỗi câu một lần mở một lần đóng. Mỗi khoảng ngắt giữa hai âm tiết là một lần đóng thừa, và tai nghe ra là tiếng "lập bập". Thời gian giữ 120 mili giây đưa con số về 9.

Ba con số khởi đầu cho bộ cổng

Ngưỡng: đặt cao hơn mức nền chừng 8 – 12 decibel, đo trên chính đường bao mà cổng dùng, không phải đo bằng RMS.

Thời gian giữ: 80 – 150 mili giây cho tiếng nói. Ngắn hơn thì lập bập, dài hơn thì cổng không kịp đóng giữa hai câu nhanh.

Nhả: 100 – 300 mili giây. Nhả quá nhanh thì đuôi tiếng bị cắt ngang nghe rất nhân tạo — đây là dấu hiệu dễ nhận nhất của một bộ cổng chỉnh ẩu.

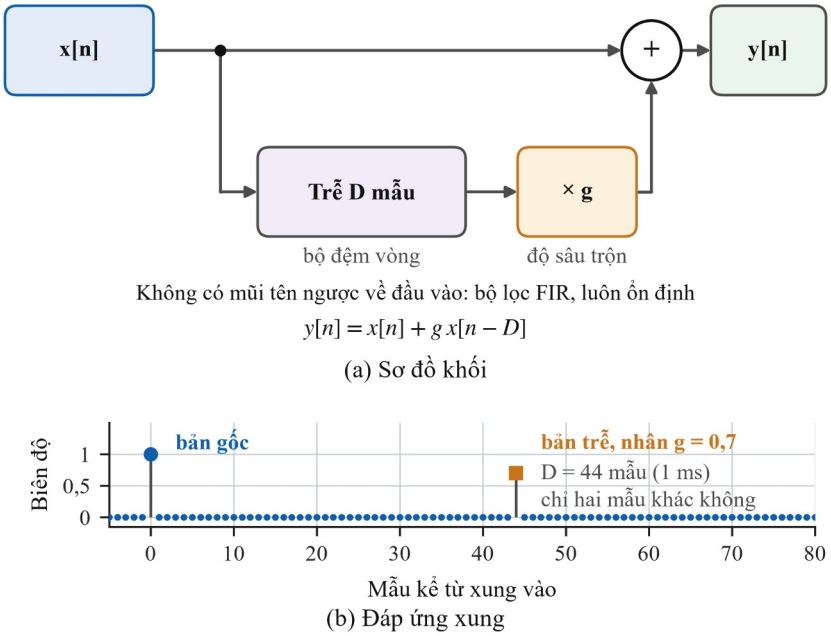
8.5 Lọc lược — khi tín hiệu gặp lại chính nó

8.5.1 Một phép cộng và một phép trễ

Nửa sau của chương đổi hẳn họ hiệu ứng. Không còn phép nhân gain thay đổi theo mức nữa; từ đây tới hết chương chỉ có một phép cộng: cộng tín hiệu với bản trễ của chính nó.

$$y[n] = x[n] + g\,x[n - D]\,; \qquad |H(f)| = \left|1 + g\,e^{-j2\pi fD/f_s}\right| \tag{8.7}$$

D là độ trễ tính bằng số mẫu, g là độ sâu trộn, f dưới chỉ số s là tần số lấy mẫu. Vế phải của công thức là đáp ứng biên độ, tính thẳng ra từ định nghĩa biến đổi Fourier của một dãy chỉ có hai mẫu khác không.



Hình 8.19 Sơ đồ khối và đáp ứng xung của bộ lọc lược FIR

Đáp ứng xung của cả bộ lọc chỉ có hai mẫu khác không: mẫu gốc và mẫu trễ đã nhân g. Không có mũi tên nào đi ngược về đầu vào, nên đây là bộ lọc FIR và nó luôn ổn định — không có cách nào làm nó kêu rú.

Đây là một lỗi trước khi là một hiệu ứng

Bộ lọc lược không phải là thứ ai đó nghĩ ra để làm hiệu ứng. Nó *tự sinh ra* mỗi khi một tín hiệu gặp lại bản trễ của chính nó.

Ba tình huống thường gặp: hai micro cùng thu một nguồn ở hai khoảng cách khác nhau; một đường tín hiệu bị nối lặp trong bàn trộn; và cộng tín hiệu gốc với tín hiệu đã đi qua một khối có trễ, chẳng hạn bộ giới hạn nhìn trước của mục 8.4.

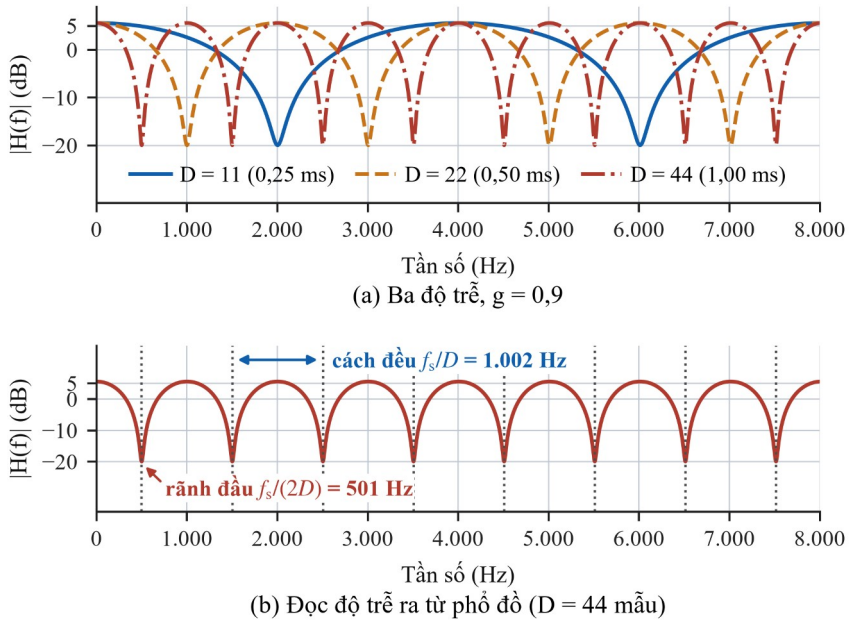
Chương 6 mục 6.3.5 đã gặp nó dưới dạng một lỗi và đo được rãnh sâu 22,3 decibel ở 501 hertz. Chương này gọi tên nó và cho công thức.

8.5.2 Rãnh nằm ở đâu — hai công thức đáng thuộc

Đáp ứng biên độ có các rãnh sâu cách đều nhau như răng lược, và vị trí của chúng tính được bằng hai công thức rất gọn.

$$f_{\text{rãnh}} = \frac{(2m+1)f_s}{2D}, \qquad \Delta f = \frac{f_s}{D} \tag{8.8}$$

m chạy từ 0 trở lên. Rãnh đầu tiên nằm ở một nửa của f dưới chỉ số s chia D, rồi các rãnh sau cách đều nhau đúng f dưới chỉ số s chia D.



(a) Ba độ trễ, g = 0,9

(b) Đọc độ trễ ra từ phổ đồ (D = 44 mẫu)

Hình 8.20 Trễ càng dài thì rãnh càng dày, và cách đọc độ trễ ra từ phổ đồ

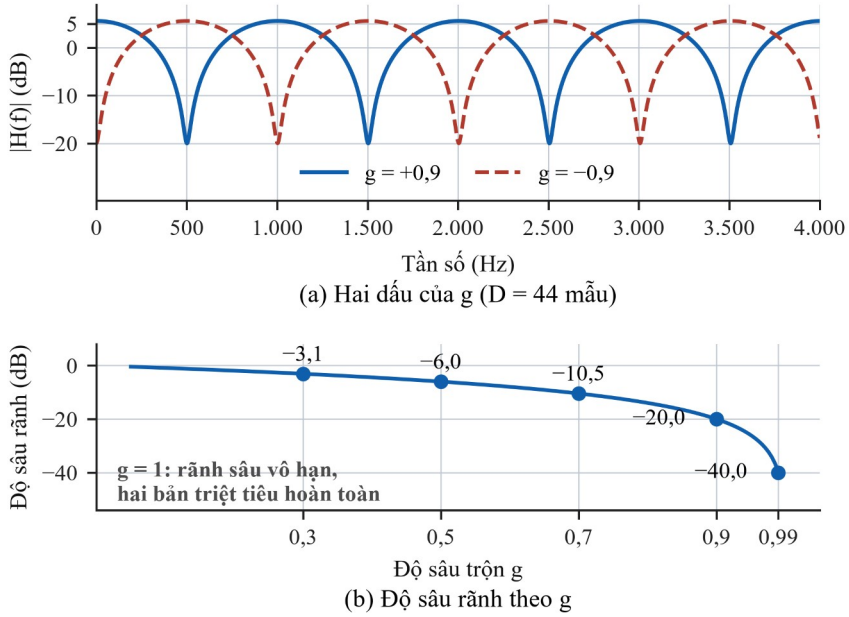
Hai công thức này đi được cả hai chiều

Chiều thuận: biết độ trễ thì tính ra ngay các rãnh nằm ở đâu. Với D = 44 mẫu ở 44,1 kHz, rãnh đầu ở 501 hertz và các rãnh cách nhau 1.002 hertz.

Chiều ngược, và đây mới là chiều hay dùng: nhìn thấy các rãnh đều nhau trên phổ đồ thì tính ngược ra D, tức là tìm được đúng độ trễ đang gây ra sự cố.

Đó là cách chẩn đoán tiếng "rỗng như nói qua ống" trong một bản thu: mở phổ đồ, đo khoảng cách giữa hai rãnh, chia tần số lấy mẫu cho khoảng cách ấy, ra ngay số mẫu trễ.

8.5.3 Dấu và độ lớn của g



(a) Hai dấu của g (D = 44 mẫu)

(b) Độ sâu rãnh theo g

Hình 8.21 Dấu của g quyết định rãnh nằm ở đâu; độ lớn quyết định rãnh sâu bao nhiêu

Với g dương, rãnh nằm ở những tần số mà bản trễ ngược pha với bản gốc. Đổi dấu g thì mọi thứ đảo lại: chỗ vừa là rãnh trở thành đỉnh và ngược lại.

Còn độ sâu rãnh thì chỉ phụ thuộc độ lớn của g: g bằng 0,5 khoét rãnh 6,0 decibel; g bằng 0,9 khoét 20,0; g bằng 0,99 khoét tới 40,0 decibel. Khi g bằng đúng 1 thì hai bản triệt tiêu hoàn toàn và rãnh sâu vô hạn.

Vì sao trộn hai micro cùng mức là trường hợp bất lợi nhất

Trộn hai micro cùng thu một nguồn, kéo hai fader lên bằng nhau, chính là g xấp xỉ 1. Đó là giá trị khoét rãnh sâu nhất. Vì thế quy tắc "ba trên một" của người thu thanh: micro thứ hai phải đặt xa nguồn ít nhất gấp ba lần khoảng cách của micro thứ nhất, để mức của nó thấp hẳn xuống và g nhỏ đi.

Cách chữa sau khi đã thu: hoặc dịch một trong hai đường đi đúng D mẫu cho hai bản trùng pha, hoặc hạ hẳn mức một đường xuống.

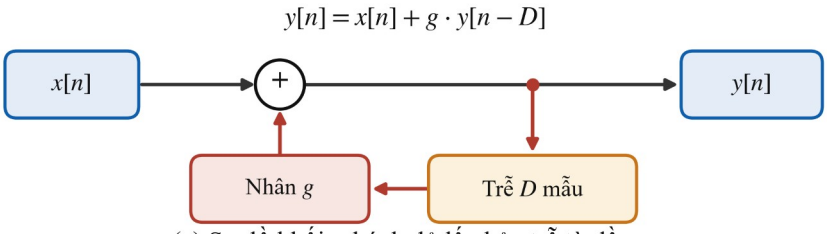
8.6 Hồi tiếp — một mũi tên đổi tất cả

8.6.1 Lấy bản trễ từ đầu ra thay vì đầu vào

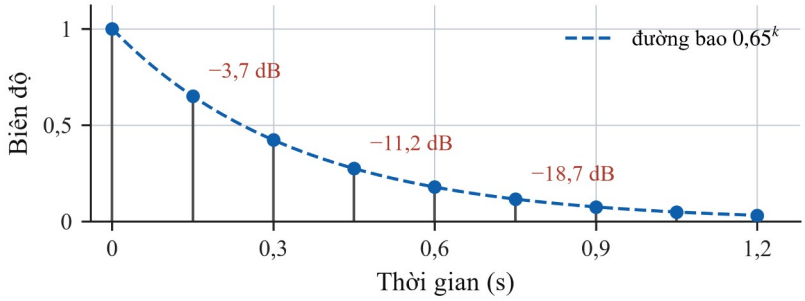
Công thức của bộ lọc lược FIR lấy bản trễ từ *đầu vào*. Đổi đúng một chữ — lấy bản trễ từ *đầu ra* — thì được một hệ khác hẳn.

$$y[n] = x[n] + g\,y[n - D]\,; \qquad |g| < 1 \tag{8.9}$$

Điều kiện |g| < 1 không phải là một lời khuyên mà là điều kiện ổn định của hệ. Với |g| ≥ 1 thì mỗi vòng tín hiệu lại to hơn vòng trước và tổng phân kỳ.



(a) Sơ đồ khối: nhánh đỏ lấy bản trễ từ đầu ra



(b) Một xung vào, cả chuỗi vọng ra (D = 150 ms, g = 0,65)

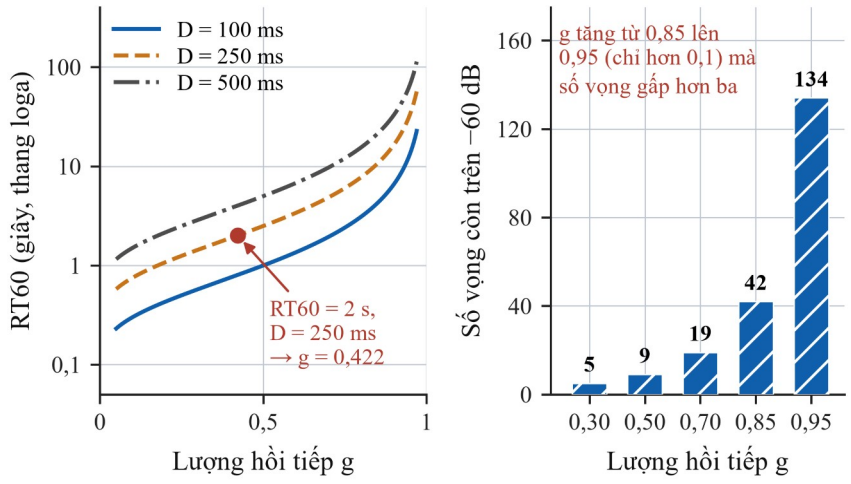
Hình 8.22 Sơ đồ có hồi tiếp, và một xung vào cho ra cả một chuỗi vọng

Một xung vào cho ra một chuỗi vọng cách đều nhau D mẫu, biên độ tụt theo cấp số nhân g, g bình phương, g mũ ba. Với g bằng 0,65 thì tiếng vọng thứ nhất thấp hơn gốc 3,7 decibel, thứ ba thấp hơn 11,2 và thứ năm thấp hơn 18,7 decibel.

8.6.2 RT60 — đặt g có căn cứ

Đường bao của chuỗi vọng là một hàm mũ, nên câu hỏi "đuôi vọng dài bao nhiêu" có lời giải kín. Quy ước của ngành là đo tới lúc tiếng tụt 60 decibel, gọi là RT60.

$$\mathrm{RT}_{60} = \frac{D}{f_s} \cdot \frac{-60}{20 \log_{10} |g|} \tag{8.10}$$



(a) RT60 theo g

(b) Số vọng nghe được

Hình 8.23 g quyết định đuôi vọng dài bao nhiêu và còn nghe được mấy tiếng

Công thức đi được cả hai chiều. Chiều thuận: g bằng 0,5 với D bằng 250 mili giây cho RT60 là 2,49 giây. Chiều ngược, và đây là chiều hay dùng: muốn RT60 đúng 2 giây với D bằng 250 mili giây thì g phải bằng 0,422.

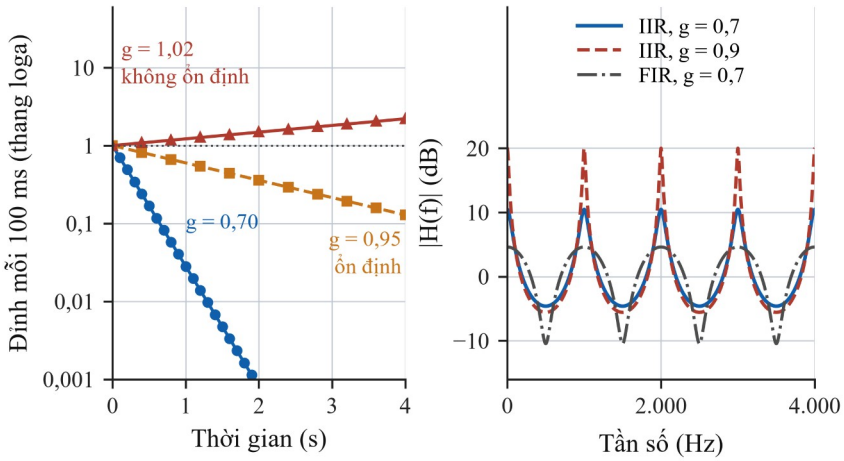
Vì sao núm feedback" bao giờ cũng khó chỉnh ở cuối hành trình

Chú ý thang loga ở hình (a): g đi từ 0,9 lên 0,99 làm RT60 tăng gấp mười lần.

Số tiếng vọng còn nghe được cũng vậy: g bằng 0,85 cho 42 tiếng, g bằng 0,95 cho 134 tiếng — chỉ hơn nhau một phần mười mà số vọng gấp hơn ba.

Bản thân phép toán là hàm mũ chứ không phải bậc nhất. Vì thế khi tự viết giao diện, nên cho núm chạy theo RT60 chứ đừng cho chạy theo g — người dùng nghĩ bằng giây, không nghĩ bằng hệ số.

8.6.3 Khi |g| vượt 1 — chỗ mã không báo lỗi



(a) Điều kiện |g| < 1 (b) Đỉnh cộng hưởng của IIR

Hình 8.24 Điều kiện ổn định, và hệ quả của hồi tiếp lên phổ

Với g bằng 1,02 — chỉ hơn 1 đúng hai phần trăm — đường bao đi *lên* thay vì đi xuống. Vòng lặp vẫn chạy hết, tệp vẫn ghi ra được, không có ngoại lệ nào được ném ra. Chỉ là tệp ấy chứa tiếng rú mỗi lúc một to.

> **Kiểm tham số ngay lúc nhận, đừng đợi nghe rồi mới biết**
>
> Đây là mẫu mực của loại lỗi khó nhất trong lập trình âm thanh: mã chạy đúng, kết quả sai, và không có dấu hiệu gì cho tới lúc mở tệp ra nghe.
>
> Hàm on_dinh_hoi_tiep của mô-đun chỉ có một dòng — trả về abs(g) < 1 — nhưng gọi nó ngay ở đầu mỗi hàm có hồi tiếp là thói quen đáng có.
>
> Trong sản phẩm thật, nên chặn cứng g ở khoảng 0,95 và cho người dùng chỉnh RT60 chứ đừng cho chỉnh g. Như thế không bao giờ vượt được ngưỡng ổn định.

Hình (b) cho thấy hệ quả còn lại của hồi tiếp, và nó là cầu nối sang mục vang của chương 9. Bộ lọc lược FIR cho các *rãnh*; bộ lọc lược IIR cho các *đỉnh* cộng hưởng nhọn ở bội số của f dưới chỉ số s chia D. Một bộ vang chính là nhiều bộ lọc lược IIR như thế chạy song song với các độ trễ nguyên tố cùng nhau, để các đỉnh của chúng không chồng lên nhau thành một tiếng ngân giả.

8.7 Bậc thang độ trễ và trễ điều biến

8.7.1 Cùng một khối, sáu hiệu ứng

Đây là bảng đáng thuộc nhất của cả phần hiệu ứng. Cùng một khối trễ, chỉ đổi con số mili giây là ra sáu hiệu ứng nghe khác hẳn nhau. Không có khối nào mới, không có công thức nào mới.

Hiệu ứng	Độ trễ	Điều biến	Nghe ra là gì
Lọc lược	0,1 – 1 ms	không	Âm bị rỗng, như nói qua ống; thường là lỗi chứ không phải hiệu ứng
Flanger	1 – 10 ms	có	Tiếng quét như máy bay phản lực; có hồi tiếp thì càng rõ
Chorus	10 – 30 ms	có	Nghe như nhiều người cùng hát: dày lên mà không thành vọng
Doubling	30 – 50 ms	không	Như thu hai lần rồi chồng lên; vẫn là một tiếng
Slapback	50 – 120 ms	không	Một tiếng vọng rõ, không lặp; dấu ấn của nhạc rock thập niên 1950
Echo	Trên 120 ms	không	Tiếng vọng tách hẳn; có hồi tiếp thì lặp nhiều lần

Cùng một khối trễ, chỉ đổi con số mili giây là ra sáu hiệu ứng khác nhau.

Hình 8.25 Bậc thang độ trễ — cùng một khối, sáu hiệu ứng

Ranh giới giữa các bậc không sắc nét, và ba mốc dưới đây giải thích vì sao chúng nằm ở đúng những chỗ ấy.

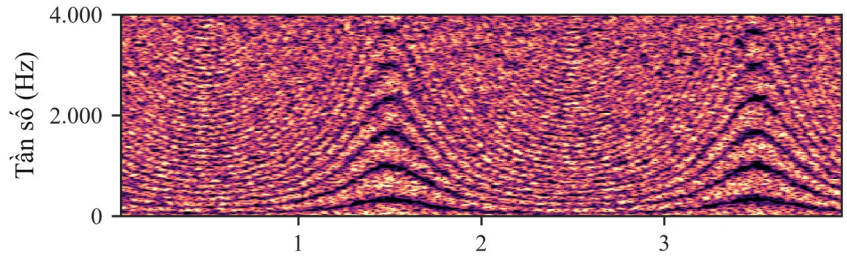
- **Khoảng 1 mili giây.** Dưới mốc này, rãnh đầu tiên nằm trên 500 hertz, tức là ngay giữa vùng nghe nhạy nhất. Tai không nghe ra hai tiếng mà nghe ra một tiếng bị đổi màu — đó là "âm rỗng".
- **Khoảng 30 mili giây.** Đây là ngưỡng mà tai bắt đầu tách được hai sự kiện, tuỳ tư liệu. Dưới ngưỡng thì hai bản nhập làm một tiếng dày hơn; trên ngưỡng thì bắt đầu nghe thành hai.
- **Khoảng 120 mili giây.** Trên mốc này thì tiếng vọng tách hẳn ra và đếm được. Dưới nó là "slapback" — một tiếng vọng nghe rõ mà vẫn gắn liền với tiếng gốc.

8.7.2 Trễ điều biến — gốc của flanger và chorus

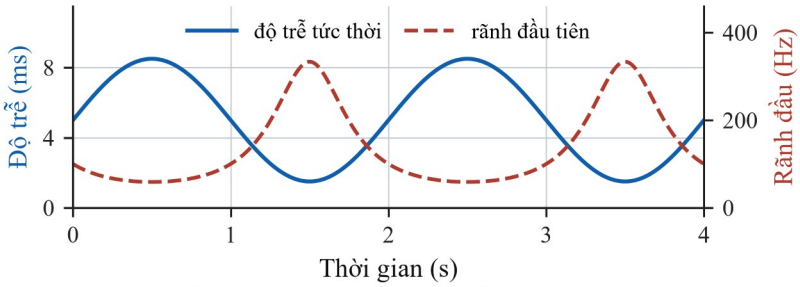
Ba trong sáu bậc ở bảng trên có cột "điều biến" đánh dấu có. Điều biến ở đây nghĩa là độ trễ *không* cố định mà chạy quanh một giá trị trung bình theo một dao động chậm.

$$D[n] = D_0 + A \sin(2\pi f_{\text{đb}}\, n / f_s) \tag{8.11}$$

D dưới chỉ số 0 là độ trễ trung bình, A là biên độ dao động, f dưới chỉ số db là tốc độ điều biến — thường dưới 2 hertz. Vì độ trễ đổi liên tục nên các rãnh của bộ lọc lược cũng trượt lên trượt xuống theo, và tai nghe ra tiếng quét đặc trưng.



(a) Phổ đồ của nhiễu trắng qua trễ điều biến: các vệt tối là rãnh



(b) Độ trễ và tần số rãnh đầu tiên biến thiên ngược chiều

Hình 8.26 Flanger trên nhiễu trắng, và vì sao nó quét

Hình (a) là phổ đồ của nhiễu trắng đã qua bộ trễ điều biến: các vệt tối cong lên cong xuống chính là các rãnh đang chạy. Hình (b) giải thích: đường tím là độ trễ tức thời, đường đỏ là tần số rãnh đầu tiên tính theo công thức (8.8). Hai đường ngược chiều nhau vì rãnh tỉ lệ nghịch với độ trễ.

Một chi tiết cài đặt không bỏ qua được

Độ trễ điều biến hầu như không bao giờ rơi vào đúng một số nguyên mẫu. Phải nội suy.

Nội suy tuyến tính là đủ cho phần lớn công việc và chỉ tốn một dòng — hàm np.interp làm ngay. Nhược điểm của nó là suy giảm nhẹ phần cao tần, và mức suy giảm ấy thay đổi theo độ trễ, nên bản thân phép nội suy cũng góp vào tiếng quét.

Muốn sạch hơn thì dùng nội suy Lagrange bậc ba hoặc bộ lọc thông toàn phần. Chương 9 sẽ quay lại chỗ này khi làm bộ trễ có độ trễ phân số đúng nghĩa.

Liệt kê 8.3 Trễ điều biến — mười dòng

```
def tre_dieu_bien(x, tre_tb_ms=5.0, sau_ms=2.0, toc_do_hz=0.5,
                  tron_g=0.7, sr=44100):
    n = x.size
    t = np.arange(n) / sr
    # độ trễ tức thời, tính bằng MẪU (số thực, không nguyên)
    d = (tre_tb_ms + sau_ms * np.sin(2 * np.pi * toc_do_hz * t))
    d = d / 1000.0 * sr
    i = np.arange(n) - d
    tre = np.interp(i, np.arange(n), x, left=0.0, right=0.0)
    return x + tron_g * tre
```

Ba cách đặt tham số cho ba hiệu ứng

Flanger: trễ trung bình 1 – 10 ms, biên độ dao động lớn so với trễ trung bình, tốc độ 0,2 – 1 Hz, trộn g lớn. Thêm hồi tiếp thì tiếng quét càng rõ.

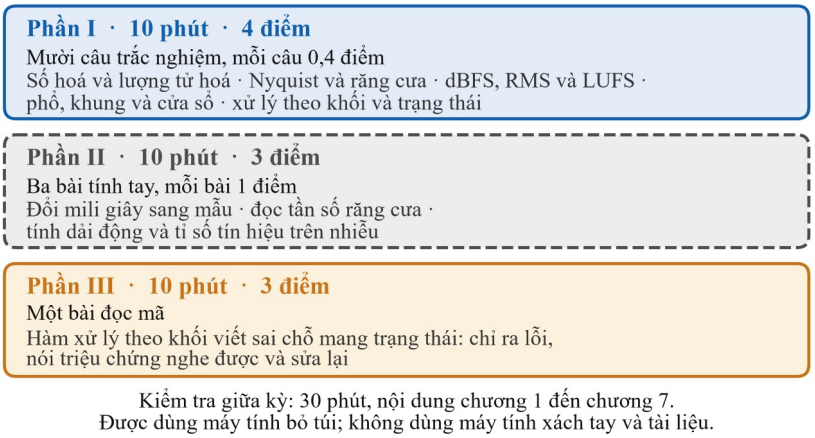
Chorus: trễ trung bình 10 – 30 ms, biên độ dao động *nhỏ*, tốc độ 0,5 – 2 Hz, trộn g vừa phải. Không hồi tiếp — có hồi tiếp là thành flanger.

Vibrato: giống chorus nhưng *không* cộng bản gốc, chỉ lấy bản trễ. Khi ấy không còn lọc lược nữa mà là dao động cao độ thuần tuý.

8.8 Tự kiểm tra giữa chặng

8.8.1 Cấu trúc bài tự kiểm tra

Trước khi sang phần cuối chương, nên dành ba mươi phút tự kiểm tra. Nội dung từ chương 1 tới chương 7 — không có phần nào của chính chương 8, vì nội dung chương này vừa mới đọc xong.



Hình 8.27 Cấu trúc bài tự kiểm tra giữa chặng — ba phần, mười điểm

Bảng 8.4 Ba phần và nội dung được hỏi

Phần	Dạng câu hỏi	Nội dung	Điểm
I	Mười câu trắc nghiệm	Số hoá và lượng tử hoá (chương 3) · Nyquist và răng cưa (chương 3) · dBFS, RMS và LUFS (chương 5, chương 7) · phổ, khung và cửa sổ (chương 4) · xử lý theo khối và trạng thái (chương 6)	4
II	Ba bài tính tay	Đổi mili giây sang mẫu và ngược lại · đọc tần số răng cưa từ một tần số vượt Nyquist · tính dải động và tỉ số tín hiệu trên nhiễu	3
III	Một bài đọc mã	Một hàm xử lý theo khối viết sai chỗ mang trạng thái — chỉ ra lỗi, nói triệu chứng nghe được, và sửa lại	3

8.8.2 Ôn tập có định hướng

Trước khi tự kiểm tra, nên rà lại đúng những chỗ sau, chứ không đọc lại cả bảy chương.

Bảng 8.5 Bảng ôn nhanh — bảy con số và một sơ đồ

Cần nhớ	Ở đâu	Vì sao quan trọng
Định lý lấy mẫu và tần số răng cưa	Chương 3, mục 3.2 và 3.3	Là câu hỏi phân biệt người hiểu và người thuộc lòng
Quan hệ số bit và dải động — 6,02 dB một bit	Chương 3, bảng quy đổi	Tính được trong đầu, không cần máy
dBFS, RMS, hệ số đỉnh	Chương 5, mục 5.4	Nền của cả chương 7 và chương 8
LUFS và vì sao nó khác RMS	Chương 7, mục 7.2	Chỗ hay trả lời chung chung nhất
Khung, chồng lấn, cửa sổ	Chương 4, mục 4.5	Cần cho câu hỏi về phổ đồ
Đổi mili giây sang mẫu	Chương 6, bảng quy đổi mục 6.4.6	Xuất hiện ở cả phần II lẫn phần III
Trạng thái mang qua giữa các khối	Chương 6, mục 6.9	Toàn bộ phần III xoay quanh chỗ này

Một lời về phần III

Phần đọc mã không kiểm tra khả năng nhớ cú pháp. Mã cho sẵn, và câu hỏi là: hàm này sai ở đâu, nghe ra thành cái gì, sửa thế nào.

Cách chuẩn bị tốt nhất không phải học thuộc mà là mở lại mã của bài tập chương 6, cố tình bỏ dòng mang trạng thái, chạy, rồi *nghe* kết quả. Nghe một lần thì nhớ mãi.

8.9 Thực hành, bài tập và tổng kết

8.9.1 Ba bài thực hành

Ba bài dưới đây làm ngay sau khi đọc xong chương, mỗi bài chừng mười phút, và cả ba đều phải *nghe* chứ không chỉ nhìn số.

- **Bài 1 — dò mức.** Lấy một đoạn giọng nói, vẽ |x[n]| cùng ba đường bao với các cặp tấn công/nhả là 1/20 ms, 5/100 ms và 30/400 ms. Trả lời: cặp nào bám theo từng âm tiết, cặp nào chỉ thấy xu hướng câu.
- **Bài 2 — bộ nén.** Nén đoạn ấy với ngưỡng đặt sao cho đường giảm gain chạm đáy ở những câu to nhất và về không ở những câu bình thường. In ra lượng giảm sâu nhất. Rồi bù gain đúng bằng lượng đỉnh đã tụt và so độ to trước sau bằng hàm do_to_tich_hop của chương 7.
- **Bài 3 — lọc lược.** Trộn đoạn ấy với bản trễ 1 mili giây của chính nó ở g bằng 0,9. Nghe. Rồi vẽ phổ và đo khoảng cách giữa hai rãnh liền nhau; chia tần số lấy mẫu cho khoảng cách ấy và kiểm xem có ra đúng 44 mẫu không.

8.9.2 Bảy bài tập

Bảy bài dưới đây. Bài nào có đáp số thì đáp số in ngay sau đề — giá trị nằm ở *cách làm* và ở phần giải thích, không nằm ở con số.

Bảng 8.6 Bảy bài tập của chương 8

#	Đề bài	Đáp số
1	Ở tần số lấy mẫu 48 kHz, tính hệ số alpha của bộ dò mức có hằng số thời gian 10 ms. Sau bao lâu thì đường bao đi được 95 phần trăm quãng đường?	alpha = 0,997919. 95 % ứng với 3·tau, tức 30 ms
2	Bộ nén có ngưỡng −18 dBFS, tỉ số 3:1, đầu gối cứng. Mức vào −6 dBFS thì mức ra bao nhiêu, và lượng giảm gain bao nhiêu?	Ra −14 dBFS; giảm 8 dB
3	Vẫn bộ nén ở bài 2. Tính gain bù theo công thức (8.5), rồi giải thích trong ba câu vì sao trên tư liệu thật lượng bù đúng thường nhỏ hơn con số ấy.	Công thức cho 12,0 dB. Vì tư liệu thật không chạm 0 dBFS và phần lớn thời gian nằm dưới ngưỡng
4	Một bản thu nghe rỗng như nói qua ống. Mở phổ đồ thấy các rãnh cách đều nhau 400 Hz, tệp ở 48 kHz. Độ trễ gây ra là bao nhiêu mẫu và bao nhiêu mili giây? Rãnh đầu tiên ở đâu?	D = 120 mẫu = 2,5 ms; rãnh đầu ở 200 Hz
5	Vòng hồi tiếp có D bằng 180 ms. Muốn RT60 đúng 1,5 giây thì g bằng bao nhiêu? Kiểm lại bằng công thức (8.10).	g = 0,4365; thay ngược lại ra đúng 1,50 s
6	Chuỗi xử lý gồm ba bộ giới hạn nối tiếp, mỗi bộ nhìn trước 5 ms, tệp ở 44,1 kHz. Tổng độ trễ là bao nhiêu mẫu và bao nhiêu mili giây? Vì sao con số này quan trọng khi có một đường tín hiệu song song?	660 mẫu = 14,97 ms. Vì trộn với đường song song sẽ sinh bộ lọc lược có rãnh đầu ở 33,4 Hz
7	Viết một hàm nhận vào g và trả về số tiếng vọng còn trên −60 dB, có kiểm $\|g\| < 1$. Chạy thử với g bằng 0,6.	13 tiếng vọng

8.9.3 Mở rộng cho ai muốn đi xa hơn

– **Nén theo đường bên (side-chain).** Cho bộ dò mức đọc một tín hiệu *khác* với tín hiệu bị nén. Đây là cách làm ducking: nhạc nền tự nhỏ đi khi có giọng

đọc. Sửa hàm nen_dong để nhận thêm một tham số tín hiệu điều khiển — chỉ cần đổi một dòng.

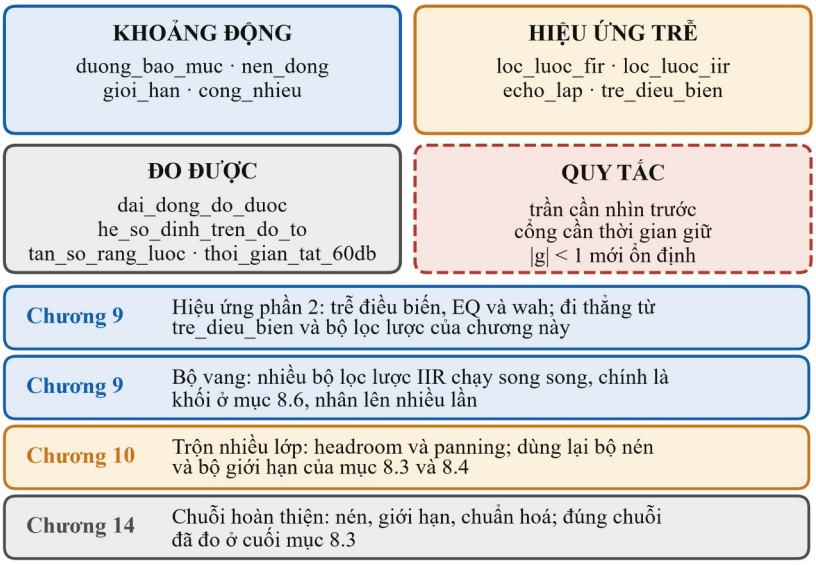
- **Nén song song.** Trộn tín hiệu gốc với bản đã nén rất mạnh, thay vì thay hẳn. Lưu ý: nếu bản nén có trễ thì phép trộn này chính là bộ lọc lược của mục 8.5 — phải bù trễ trước khi trộn.
- **Bộ giới hạn theo dBTP.** Ghép hàm true_peak_dbtp của chương 7 vào vòng của hàm gioi_han, để trần được giữ theo đỉnh thật chứ không theo đỉnh mẫu. Đo lại tư liệu gắt của mục 8.4.3 và xem còn vượt bao nhiêu.
- **Bộ lọc lược có độ trễ phân số.** Thay np.interp bằng nội suy Lagrange bậc ba và đo lại đáp ứng cao tần ở vài độ trễ khác nhau. Chương 9 sẽ cần đúng khối này.

8.9.4 Câu hỏi tự trả lời

Bảy câu dưới đây không có đáp án in trong sách. Trả lời được cả bảy nghĩa là đã nắm chương này.

- Vì sao bộ dò RMS dùng một hằng số thời gian còn bộ dò đỉnh dùng hai? Nếu cho bộ dò RMS dùng hai thì sai ở chỗ nào?
- Một bộ nén có đường giảm gain không bao giờ về không. Ngưỡng đang quá cao hay quá thấp? Nghe ra thành cái gì?
- Vì sao đầu gối mềm làm bộ nén "nén nhiều hơn dự đoán"? Con số nào ở mục 8.3.3 chứng minh điều đó?
- Tỉ số 1000:1 với tấn công 0,1 ms vẫn để đỉnh vượt qua ở 1,81 dBFS. Giải thích tại sao, chỉ bằng khái niệm hằng số thời gian.
- Bộ giới hạn giữ đỉnh mẫu tuyệt đối mà đỉnh thật vẫn vượt trần. Điều này mâu thuẫn ở chỗ nào và không mâu thuẫn ở chỗ nào?
- Hai micro cùng thu một nguồn, trộn cùng mức. Vì sao đây là trường hợp bất lợi nhất chứ không phải trường hợp trung bình?
- Vì sao nên cho người dùng chỉnh RT60 thay vì chỉnh g, dù hai đại lượng ấy tương đương nhau về mặt toán học?

8.9.5 Tổng kết — chương 8 để lại gì



Hình 8.28 Chương 8 để lại gì cho các chương sau

Bảng 8.7 Mười hàm mới của chương 8

Hàm	Làm gì	Dùng lại ở
he_so_thoi_gian	Đổi hằng số thời gian sang hệ số làm trơn	Mọi hàm còn lại của nhóm
duong_bao_muc	Bộ dò mức theo đỉnh hoặc theo RMS	Bộ nén, bộ giới hạn, bộ cổng
dac_tuyen_nen, do_giam_gain_db	Đường đặc tuyến tĩnh, có đầu gối mềm	Bộ nén
nen_dong	Bộ nén đủ bốn tham số cộng gain bù	Chương 14 — chuỗi hoàn thiện
gioi_han	Bộ giới hạn có nhìn trước, giữ trần tuyệt đối	Chương 14
cong_nhieu	Bộ cổng và bộ nới	Chương 14 — dọn kết quả AI
loc_luoc_fir, loc_luoc_iir	Hai bộ lọc lược	Chương 9 — flanger và bộ vang
tan_so_rang_luoc, khoang_cach_rang	Đọc độ trễ ra từ vị trí các rãnh	Chương 9 — chẩn đoán
echo_lap, on_dinh_hoi_tiep, thoi_gian_tat_60db, g_cho_rt60	Vòng hồi tiếp và cách đặt g có căn cứ	Chương 9 — bộ vang
tre_dieu_bien	Trễ thay đổi theo thời gian	Chương 9 — flanger, chorus, vibrato

Ba câu mang về

Một bộ nén là một phép nhân với hệ số thay đổi theo mức; một hiệu ứng trễ là một phép cộng với bản trễ của chính nó. Không có phép toán nào khó hơn thế trong cả chương.

Mỗi tham số phải trả lời được một câu hỏi cụ thể, và mỗi lựa chọn phải đo được cái giá của nó. Chuỗi ở mục 8.3.7 đổi 5,3 *lu* dải độ to lấy 4,9 decibel độ to — đó là một cuộc trao đổi có giá rõ ràng, không phải điều kỳ diệu.

Hai chỗ mã không báo lỗi mà kết quả vẫn sai: bỏ sót việc mang trạng thái của bộ dò mức qua giữa các khối, và để |g| vượt 1 trong vòng hồi tiếp. Cả hai đều kiểm được bằng một dòng đặt ngay đầu hàm.

Chương 9 tiếp tục phần hiệu ứng với hiệu ứng phần 2: trễ điều biến thành flanger và chorus đúng nghĩa, cân bằng tần số và bộ lọc wah, điều chế biên độ, rồi bộ vang và tích chập đáp ứng xung — nơi vòng hồi tiếp của mục 8.6 được nhân lên nhiều lần. Cần chuẩn bị trước: đọc lại mục 4.6 về đáp ứng tần số, và chuẩn bị sẵn một đoạn ghi guitar hoặc một đoạn trống có nhiều cao tần.

Tài liệu tham khảo của chương

Nguồn trực tiếp của các công thức trong bài

- Giannoulis, D., Massberg, M., Reiss, J. D. (2012) — Digital Dynamic Range Compressor Design: A Tutorial and Analysis, Journal of the Audio Engineering Society 60(6). Đây là nguồn *trực tiếp* của mục 8.2 và 8.3: sơ đồ bốn khối, đường đặc tuyến có đầu gối mềm, và lập luận vì sao phải làm trơn trên gain chứ không trên mức. Bài này đọc được với kiến thức của chương 4 và chương 6.
- Zölzer, U. (chủ biên) — DAFX: Digital Audio Effects. Sách gối đầu giường của cả phần hiệu ứng. Chương về xử lý khoảng động và chương về hiệu ứng trễ tương ứng đúng hai nửa của chương này.
- Smith, J. O. — Physical Audio Signal Processing, và Introduction to Digital Filters. Nguồn của phần lọc lược, hồi tiếp và điều kiện ổn định. Đọc miễn phí trên trang của tác giả tại Đại học Stanford.
- Reiss, J. D., McPherson, A. — Audio Effects: Theory, Implementation and Application. Viết cho người lập trình, nhiều mã giả và ít công thức phức tạp.

Chuẩn liên quan

- ITU-R BS.1770-4 và EBU R 128 — đã dùng ở chương 7, và chương này dùng lại ở mục 8.3.7 khi đo kết quả của chuỗi nén, giới hạn, chuẩn hoá.
- AES Technical Document AES-TD1004 — khuyến nghị về mức và dải động cho nội dung phát trực tuyến. Đây là chỗ trả lời câu hỏi "nén tới đâu là vừa" từ góc nhìn của cả ngành.

Công cụ nên biết

- Bất kỳ bộ nén nào có hiển thị đường giảm gain — bản miễn phí đi kèm Reaper, Ardour hoặc Audacity đều đủ. Điều quan trọng không phải chất lượng bộ nén mà là *nhìn* được đường giảm gain khi chỉnh từng núm.
- ffmpeg với bộ lọc acompressor và alimiter — để đối chiếu kết quả của mô-đun tự viết với một cài đặt độc lập.
- Một bộ phân tích phổ chạy thời gian thực, để nhìn các rãnh của bộ lọc lược chạy khi đổi độ trễ ở mục 8.5.

Đọc thêm theo hướng

- **Muốn đi sâu về bộ nén:** tìm đọc về bộ nén dùng ống chân không và bộ nén quang học — hai loại có "tính cách" rất riêng, và cái riêng ấy đến từ đường bao mức phi tuyến chứ không phải từ đường đặc tuyến. Đó là hướng mở rộng tự nhiên của mục 8.2.
- **Muốn đi sâu về bộ giới hạn:** tìm đọc về bộ giới hạn nhiều tầng và về "clipper" mềm. Chúng giải cùng bài toán của mục 8.4 nhưng đánh đổi khác: méo nhiều hơn mà đỡ bơm hơn.
- **Muốn đi sâu về hiệu ứng trễ:** tìm đọc về bộ lọc thông toàn phần và về mạng trễ phản hồi (feedback delay network). Đó chính là con đường từ mục 8.6 sang bộ vang của chương 9.

Một lưu ý về các con số trong bài

> **Cách đọc các con số**
>
> Các con số về đường đặc tuyến, về vị trí rãnh và về RT60 là kết quả tính chính xác từ công thức — chúng đúng với mọi tư liệu.
>
> Các con số về tư liệu (bản nhạc cổ điển, đoạn phỏng vấn, tiếng gõ) là kết quả *đo* trên tư liệu dựng bằng mã, mô tả trong chú thích từng hình. Trên tư liệu thật con số sẽ khác; điều không đổi là thứ tự và bậc độ lớn.
>
> Riêng bảng ở mục 8.3.7 nên tự đo lại trên tư liệu của chính mình. Bốn dòng của bảng ấy là bốn thứ cần đặt lên bàn mỗi lần quyết định nén tới đâu.
>
> Mọi phép tính trong bài sinh ra từ mô-đun sigtools.py, đã qua 298 phép tự kiểm.