ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA

BÁO CÁO BÀI TẬP LỚN
ĐẠI SỐ TUYẾN TÍNH
PHƯƠNG PHÁP FOURIER RỜI RẠC (DFT) ĐỂ KHỬ NHIỄU

Lớp: L13

GVHD: T.S Đặng Văn Vinh

Nhóm 1

Họ và tên	*MSSV*	*Lớp*
Bùi Thị Quỳnh Anh	2310057	L13
Đặng Đình Đức	2310762	L13
Vũ Nhật Huy	2311267	L13
Đào Huỳnh Khang	2311405	L13
Đặng Lê Mỹ Mỹ	2312149	L13
Cao Chí Nguyên	2312331	L13
Đặng Minh Quân	2312819	L13
Chu Quốc Tuấn	2313727	L13

Thành phố Hồ Chí Minh - 2023

Mục lục

I Cơ sở lý Thuyết

1 Giới thiệu

Trong toán học phép biến đổi Fourier rời rạc (DFT: Discrete Fourier Transform) còn được gọi là biến đổi Fourier hữu hạn, là một công cụ toán học được sử dụng rộng rãi nhất trong toán học ứng dụng, chẳng hạn như trong lĩnh vực xử lý tín hiệu và nén dữ liệu, biến đổi Fourier dựa trên ý tưởng các hàm có tính chất phù hợp có thể được biểu diễn bằng tổ hợp tuyến tính của các hàm lượng giác.

Euler là người đầu tiên đưa ra công thức cho các hệ số của chuỗi Fourier. Dựa trên các nghiên cứu của Euler, Clairaut đã công bố công thức đầu tiên cho phép biến đổi Fourier rời rạc (DFT) vào năm 1754 là cái chúng ta hiện biết. Năm 1805, Carl Friendrich Gauss (1777-1855) đã công bố công thức cho DFT. Năm 1822, Joseph Fourier (1768-1830) đưa ra công trình về dòng nhiệt cung cấp nền tảng cho phép biến đổi Fourier và nguồn gốc tên gọi của nó. Năm 1965, James Cooley và John Tukey, đã phát minh ra thuật toán biến đổi Fourier nhanh (FFT), thuật toán này đã cho phép một loạt các ứng dụng số trong thời đại kỹ thuật số ra đời. Các ứng dụng này bao gồm nén dữ liệu JPEG, phân tích cấu trúc tinh thể, giải phương trình vi phân từng phần, xử lí tín hiệu nhiễu,...

Một tín hiệu rời rạc là một tín hiệu được biểu thị dưới dạng một mảng các giá trị, mỗi giá trị được lấy mẫu tại một thời điểm cụ thể. Các giá trị này có thể đại diện cho các giá trị điện áp, cường độ âm thanh, hoặc bất kỳ giá trị nào khác thay đổi theo thời gian. DFT hoạt động bằng cách tính toán tích chập của tín hiệu với các hàm sin có tần số khác nhau. Kết quả của tích chập này là một mảng các giá trị, mỗi giá trị đại diện cho cường độ của tần số tương ứng.

2 Định nghĩa

Phép biến đổi Fourier rời rạc (DFT- Discrete Fourier Tranform) là một phương pháp toán học quan trọng trong xử lý tín hiệu và tính toán. Nó được sử dụng để chuyển đổi một số tín hiệu rời rạc từ miền thời gian sang miền tần số. Cụ thể, nếu chúng ta có một chuỗi số $x_0, x_1, \ldots, x_{N-1}$ ở trong miền thời gian thì DFT chuyển đổi chuỗi này thành một chuỗi số khác $X_0, X_1, \ldots, X_{N-1}$ trong miền tần số qua công thức sau:

$$X_k = \sum_{n=0}^{N-1} x_n . e^{\frac{-i2\pi}{N}kn}$$

trong đó x_n là tín hiệu trong miền thời gian, X_k là tín hiệu trong miền tần số và N là số lượng mẫu trong tín hiệu.

Phép biến Fourier rời rạc ngược (IDFT-Inverse Discrete Fourier Tranform) là phép biến đổi ngược của phép biến đổi Fourier rời rạc (DFT). Nó được sử dụng để chuyển đổi tín hiệu từ miền tần số sang miền thời gian. Cụ thể, khi chúng ta có một chuỗi số $X_0, X_1, \ldots, X_{N-1}$ trong miền tần số thì IDFT chuyển đổi chuỗi này thành một chuỗi số

$x_0, x_1, \ldots, x_{N-1}$ ở trong miền thời gian bằng cách sử dụng phép tính dưới đây:

$$x_n = \frac{1}{N}\sum_{k=0}^{N-1} X_k.e^{\frac{-i2\pi kn}{N}}$$

Phép biến đổi Fourier rời rạc là một biến đổi tuyến tính khả nghịch trong đó C ký hiệu tập các số phức. Nói cách khác, với mọi $N > 0$, mọi vectơ phức N chiều đều có một DFT và một IDFT, và chúng đều là các vectơ phức N chiều.

3 Tính chất

a) Tuyến tính

Với mọi $N > 0$, mọi vecto phức N chiều đều có một DFT (biến đổi Fourier rời rạc) và một IDFT (biển đổi Fourier rời rạc ngược) và chúng đều là vecto phức N chiều. Hay nói cách khác, nếu $x(n), y(n)$ có các biến đổi Fourier rời rạc lần lượt là $X(k), Y(k)$ thì:

$$ax(n) + by(n) \longleftrightarrow aX(k) + bY(k) \quad (a, b = const)$$

b) Tuần hoàn

Nếu ta tính biểu thức định nghĩa DFT tại mọi số nguyên k thay vì chỉ cho $k = 0, ..., N-1$ thì dãy số nhận được là một mở rộng tuần hoàn của DFT và có chu kì N. Tính tuần hoàn có được chứng minh trực tiếp từ định nghĩa:

$$X_{k+N} \triangleq \sum_{n=0}^{N-1} x_n.e^{\frac{-i2\pi}{N}(k+N)n} = \sum_{n=0}^{N-1} x_n.e^{\frac{-i2\pi}{N}kn} \underbrace{e^{-i2\pi n}}_{1} = \sum_{n=0}^{N-1} x_n.e^{\frac{-i2\pi}{N}kn} = X_k$$

c) Trực giao

Các vecto $e^{\frac{i2\pi}{N}kn}$ tạo thành một cơ sở trực giao của tập các vecto phức N chiều:

$$\sum_{n=0}^{N-1}\left(e^{\frac{i2\pi}{N}kn}\right)\left(e^{\frac{-i2\pi}{N}k'n}\right) = N\delta_{kk'}$$

trong đó $\delta_{kk'}$ là hàm delta Kronecker. Có thể dùng điều kiện trực giao để suy ra công thức cho IDFT từ định nghĩa của DFT, và điều kiện này tương đương với điều kiện unita dưới đây.

* Trong toán học, ký hiệu Kronecker delta là một hàm số của hai biến, thường là các số nguyên không âm. Hàm số có giá trị 1 nếu hai biến bằng nhau, và 0 nếu chúng khác nhau:

$$\delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$

d) Unita

Phép biến đổi DFT có thể được biểu diễn dưới dạng ma trận sau:

$$F = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & e^{\frac{-i2\pi}{N}} & e^{\frac{-i2\pi}{N}.2} & \ldots & e^{\frac{-i2\pi}{N}(N-1)} \\ 1 & e^{\frac{-i2\pi}{N}.2} & e^{\frac{-i2\pi}{N}.4} & \ldots & e^{\frac{-i2\pi}{N}2.(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & e^{\frac{-i2\pi}{N}(N-1)} & e^{\frac{-i2\pi}{N}2.(N-1)} & \ldots & e^{\frac{-i2\pi}{N}(N-1)^2} \end{bmatrix}$$

Phép biến đổi ngược chính là ma trận nghịch đảo của ma trận trên:

$$F_N^{-1} = \frac{1}{N} F_N^*$$

Với hằng số chuẩn hóa unita $\frac{1}{\sqrt{N}}$, DFT trở thành một biến đổi unita, định nghĩa bởi một ma trận unita:

$$U = \frac{1}{\sqrt{N}} F$$

$$U^{-1} = U^*$$

$$|\det(U)| = 1$$

Nếu X_k và Y_k là các DFT của x_n và y_n thì theo định lý Plancherel:

$$\sum_{n=0}^{N-1} x_n y_n^* = \frac{1}{N} \sum_{k=0}^{N-1} X_k Y_k^*$$

trong đó dấu sao ký hiệu số phức liên hợp.Nếu ta coi DFT chỉ là một phép biến đổi tọa độ trong đó chỉ cần chỉ ra các thành phần của vectơ trong hệ tọa độ mới, thì mệnh đề trên chỉ nói rằng tích vô hướng của hai vectơ được giữ nguyên trong phép biến đổi unita DFT. Trong trường hợp đặc biệt khi $x = y$, điều này có nghĩa là độ dài vectơ cũng được giữ nguyên—đây chính là định lý Parseval (Định lý Parseval là một trường hợp đặc biệt của định lý Plancherel):

$$\sum_{n=0}^{N-1} |x_n|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X_k|^2$$

4 Ứng dụng

DFT rất hữu ích trong nhiều ứng dụng, bao gồm phân tích phổ tín hiệu đơn giản được nêu ở trên. Biết được cách biểu thị một tín hiệu dưới dạng kết hợp các sóng cho phép thao tác với tín hiệu đó và so sánh các tín hiệu khác nhau:

- Các tệp kỹ thuật số (jpg, mp3, v.v.) có thể được thu nhỏ bằng cách loại bỏ sự đóng góp từ các sóng ít quan trọng nhất trong tổ hợp.
- Các tập tin âm thanh khác nhau có thể được so sánh bằng cách so sánh các hệ số X_k của DFT.
- Sóng vô tuyến có thể được lọc để tránh "nhiễu" và lắng nghe các thành phần quan trọng của tín hiệu.

Các ứng dụng khác của DFT phát sinh vì nó có thể được tính toán rất hiệu quả bằng thuật toán biến đổi Fourier nhanh (FFT). Ví dụ: DFT được sử dụng trong các thuật toán tiên tiến để nhân các đa thức và số nguyên lớn với nhau; thay vì làm việc trực tiếp với phép nhân đa thức, việc tính DFT của các hàm đa thức và chuyển đổi bài toán nhân đa thức thành bài toán tương tự liên quan đến DFT của chúng sẽ nhanh hơn. Dưới đây là ba ứng dụng phổ biến nhất của DFT:

Phân tích phổ. Để tìm phổ của tín hiệu (liên tục và rời rạc), ta cần biết giá trị của phổ ở tất cả các thời điểm. Tuy nhiên, ta chỉ quan sát được tín hiệu trong một khoảng thời gian hữu hạn nên phổ tính được cũng chỉ là xấp xỉ của phổ chính xác. DFT ứng dụng rất hệu quả trong việc này.

Hình ảnh. Biến đổi Fourier rời rạc được sử dụng nhiều trong khử nhiễu và làm nét hình ảnh. Biến đổi Fourier rời rạc áp dụng cho ảnh số, được sử dụng để tách ảnh trong miền không gian thành các thành phần tần số. Tuy nhiên, nó chỉ áp dụng cho các điểm mẫu, vì thế nó không cho lại tất cả tần số hình thành ảnh số mà chỉ cho tập hợp điểm mẫu đủ lớn có khả năng mô tả đầy đủ hình ảnh trong miền không gian.

Y tế. Trên thế giới, biến đổi Fourier rời rạc được áp dụng cho định lượng bằng quang phổ hấp thụ trên nguyên lí: phổ tí đối của chất phân tích có thể triển khai như chuỗi Fourier hữu hạn từ thập niên 90. Kỹ thuật này áp dụng thành công để định lượng hỗn hợp chứa 2,3 thành phần.

II Ứng dụng khai triển Fourier để khử nhiễu âm thanh

Phép biến đổi $Y = F_n \times X$ được gọi là phép biến đổi rời rạc của vectơ X.

+ Vecto $Y = F_n.X$ có dạng $Y = A + iB$.

+ Vecto A chứa các hệ số α_t trong $\sum_0^{n-1} \alpha_t.\cos\dfrac{2\pi t}{n}$

+ Vecto B chứa các hệ số β_t trong $\sum_0^{n-1} \beta_t.\sin\dfrac{2\pi t}{n}$

Như vậy dùng phép biến đổi Fourier rời rạc, ta chuyển tín hiệu X ở miền thời gian thành tín hiệu ở miền tần số gồm tổng các hàm $\cos\dfrac{2\pi t}{n}$ và $\sin\dfrac{2\pi t}{n}$.
Giả sử tín hiệu gốc có dạng $\sum_0^{n-1} \alpha_t.\cos\dfrac{2\pi t}{n} + \beta_t.\sin\dfrac{2\pi t}{n}$ nhiễu. Sau khi phân tích Fourier rời rạc $Y = F_n.X$, so sánh với hình vẽ trước và sau khi phân tích, ta có thể xác định tần số tín hiệu chính và tần số nhiễu. Lọc bỏ tín hiệu nhiễu, chỉ giữ lại tín hiệu chính, sau đó phân tích Fourier ngược ta sẽ thu được tín hiệu đã khử nhiễu.

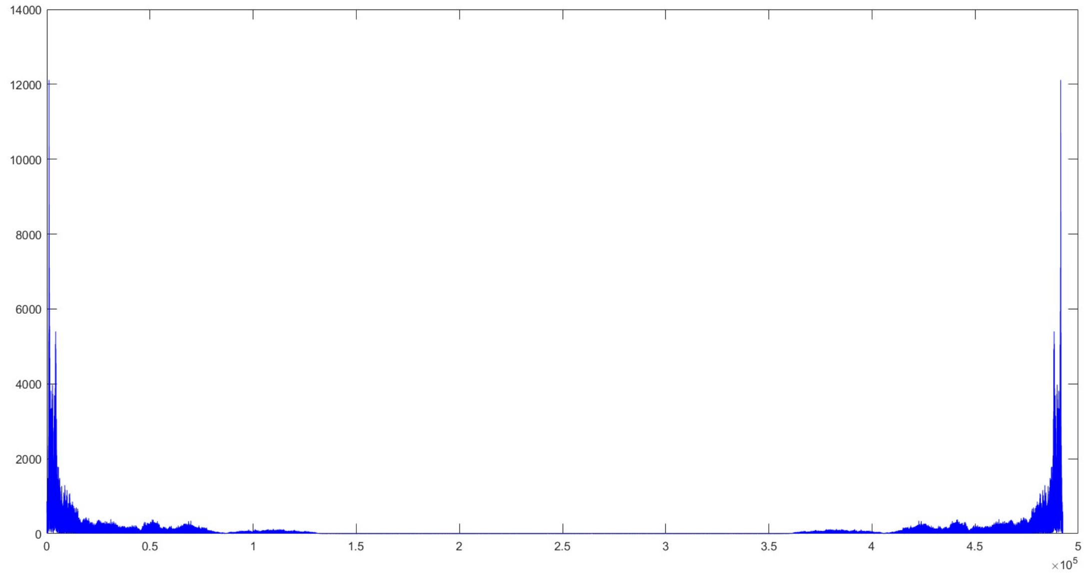
III Code Matlab

Các bước dùng phép biến đổi Fourier rời rạc để khử nhiễu:

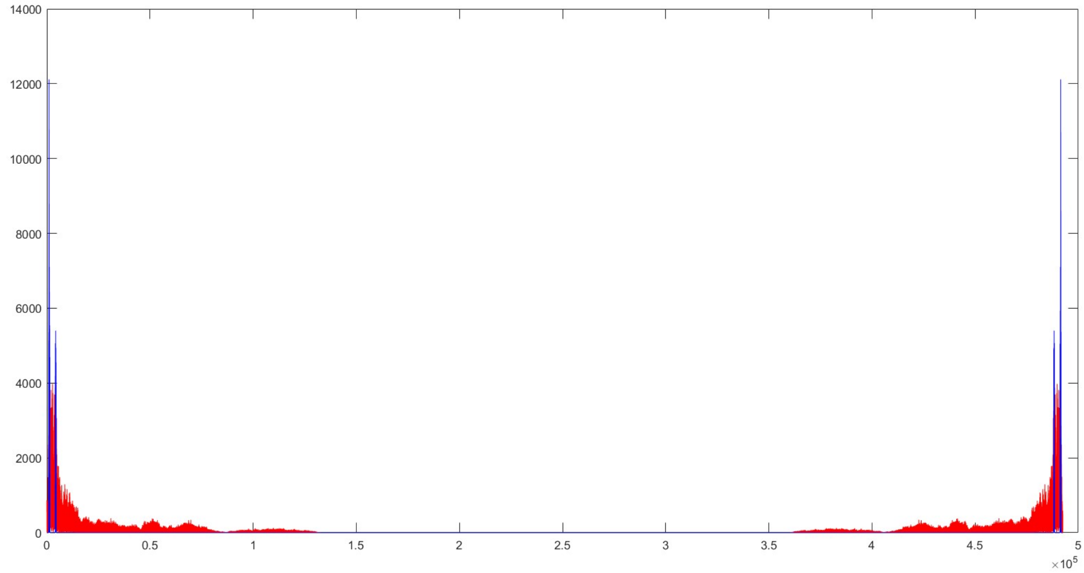
Bước 1. Số hóa một file âm thanh, ta có một véctơ: $y = audioread('filename');$
Bước 2. Phân tích Fourier rời rạc véctơ y: $Y = fft(y)$; Fast Fourier Transform.
Bước 3. Vẽ tin hiệu ban đầu. Phân tích tần số của tín hiệu để biết được tần số của tín hiệu chính và của tín hiệu bị nhiễu: $plot(abs(Y))$;
Bước 4. Lọc bỏ bớt các tần số của tín hiệu nhiễu chỉ giữ lại tần số của tín hiệu chính:

$M = max(abs(Y));$
$thresh = 0.5;$ (dựa vào tín hiệu ban đầu, xác định hệ số thresh để khử được nhiễu)
$sound(y);$
$Ythresh = (abs(Y) > thresh * M). * Y;$
$sum(abs(Ythresh) > 0)/L;$
Bước 5. Biến đổi Fourier ngược lại: $ythresh = real(ifft(Ythresh));$
Ví dụ: Ta sẽ áp dụng các bước trên để khử đoạn âm thanh có tiếng xe cộ đông đúc ngoài đường:

```
y = audioread('noise.wav');
L = length(y);
Y = fft(y);
sound(y);
plot(abs(Y),'r');
hold on;
M = max(abs(Y));
thresh = 0.5;
Ythresh = (abs(Y)>thresh*M).*Y;
sum(abs(Ythresh)>0)/L;
ythresh = real(ifft(Ythresh));
plot(abs(Ythresh),'b');
sound(ythresh);
```



Tín hiệu gốc của đoạn âm thanh



Tín hiệu âm thanh đã qua xử lý

IV Kết luận

Biến đổi Fourier hữu hạn (DFT) là một trong những công cụ mạnh mẽ nhất trong xử lý tín hiệu kỹ thuật số, nó cho phép chúng ta tìm ra phổ của tín hiệu có thời lượng hữu hạn.Thông qua bài tập lớn này, với sự phân chia công việc hợp lý cho từng thành viên và tinh thần trách nhiệm cao của mỗi người, nhóm em đã đạt được một số kết quả sau:

- Biết cách sử dụng Matlab để khử nhiễu âm thanh.
- Hiểu rõ ứng dụng của Fourier hữu hạn.
- Trau dồi kĩ năng làm việc nhóm hiệu quả.
- Có sự hứng thú với môn học.

Tài liệu tham khảo

1. Đặng Văn Vinh, Giáo trình Đại số tuyến tính, Đại học Quốc gia Thành phố Hồ Chí Minh.

2. Discrete-Time Signal Processing (Prentice Hall Signal Processing) 3rd Edition by Alan Oppenheim.

3. The Fourier Transform & Its Applications McGraw-Hill Science/Engineering/-Math; 3rd edition (June 8, 1999).

4. Digital Signal Processing Using MATLAB 1st Edition CL Engineering; 1st edition (July 11, 1999).

5. Meyer C.D, Matrix analysis and Applied linear algebra, SIAM, 2000, chapter 5, section 8.

6. Isaac Amidror, Mastering the Discrete Fourier Transform in One, Two or Several Dimensions: Pitfalls and Artifacts, Springer,2013.

7. Audio Filtering With MATLAB (Denoising) link