Математичка гимназија

МАТУРСКИ РАД

- из математике -

Брза Фуријеова трансформација

Ученик:
Марко Шишовић IVд

Ментор:
др Раде Лазовић

Београд, јун 2020.

Садржај

1

Увод у Фуријеову анализу

У овом матурском раду биће обрађена брза Фуријеова трансформација, алгоритам за израчунавање дискретне Фуријеове трансформације (ДФТ). Ипак, са циљем бољег пазумевања ДФТ, у овом уводу биће укратко објашњене различите врсте Фуријеове анализе.

1.1 Фуријеов ред

Фуријеов ред је развој периодичне реалне функције $f(x)$ периода T на бесконачну суму синуса и косинуса, формално записано на следећи начин:

$$F(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty}(a_n cos(n\frac{2\pi}{T}x) + b_n sin(n\frac{2\pi}{T}x))$$

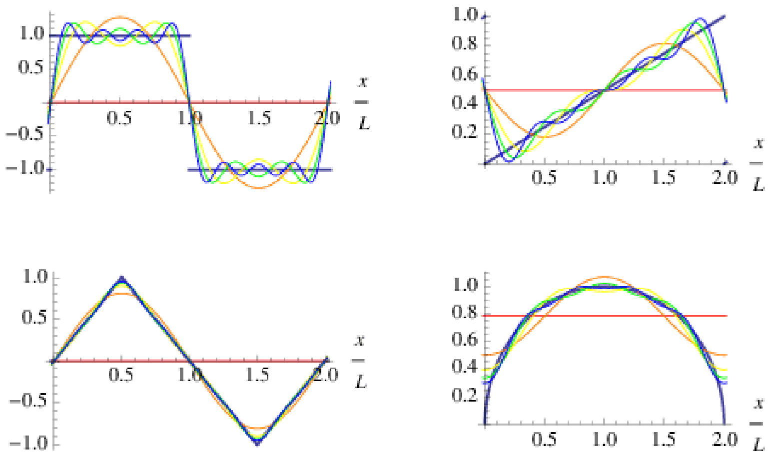
где Фуријеови коефицијенти за $n = 1, 2, 3...$ износе:

$$a_0 = \frac{2}{T}\int\limits_{-\frac{T}{2}}^{\frac{T}{2}} f(x)\mathrm{d}t,\ a_n = \frac{2}{T}\int\limits_{-\frac{T}{2}}^{\frac{T}{2}} f(x)cos(n\frac{2\pi}{T}x)\mathrm{d}t$$

$$b_n = \frac{2}{T}\int\limits_{-\frac{T}{2}}^{\frac{T}{2}} f(x)sin(n\frac{2\pi}{T}x)\mathrm{d}t$$

При добијању ових вредности користи се својство ортогоналности ових синусоида, тако што множимо обе стране са једном од њих и потом интегралимо обе стране.

Формула се може записати и на следећи начин:



Слика 1.1: Примери Фуријеових развоја различитих функција

$$F(x) = \sum_{n=-\infty}^{\infty} c_n e^{in\frac{2\pi}{T}x}$$

где коефицијенти за $n = 1, 2, 3...$ износе:

$$c_0 = \frac{a_0}{2},\ c_n = \frac{1}{2}(a_n - ib_n)$$

$$c_{-n} = \frac{1}{2}(a_n + ib_n)$$

Еквивалентност ова два записа следи директно из Ојлерове формуле $e^{i\phi} = cos(\phi) + isin(\phi)$. Ако уврстимо формуле за a_n и b_n у претходну формулу за Фуријеов развој добијамо:

$$F(x) = \sum_{n=-\infty}^{\infty} \frac{1}{T}\left(\int_{-\frac{T}{2}}^{\frac{T}{2}} f(x)e^{-in\frac{2\pi}{T}x}\mathrm{d}x\right)e^{in\frac{2\pi}{T}x}$$

Фуријеов ред $F(x)$ функције $f(x)$ конвергира функцији:

$$\overline{f}(x_0) = \frac{1}{2}\left(\lim_{x\to x_0^-} f(x) + \lim_{x\to x_0^+} f(x)\right)$$

Одавде такође видимо да почетна фунцкија мора имати коначно много тачака прекида и недефинисаности.

1.2 Фуријеова трансформација

Помоћу Фуријеовог реда могуће је апроксимирати периодичне функције користећи синусоиде чије су фреквенције умношци фреквенције те функције. Међутим, корисно би било када не бисмо били ограничени на периодичне функције. Тако долазимо до Фуријеове трансформације која користи синусоиде свих реалних фреквенција. Ако је $G(\mathbb{R})$ фамилија функција облика $g(x) : \mathbb{R} \to \mathbb{C}$ и $f(x)$ функција из те фамилије, онда је њена Фуријеова трансформација задата формулом:

$$F(\omega) = \int\limits_{-\infty}^{\infty} f(x)e^{-i\omega x}\mathrm{d}x$$

док је инверзна трансформација:

$$f(x) = \frac{1}{2\pi}\int\limits_{-\infty}^{\infty} F(\omega)e^{i\omega x}\mathrm{d}\omega$$

Као што видимо ове формуле су врло сличне формулама за Фуријеов развој што је и очекивано, Ако апериодичну функцију посматрамо као периодичну чији период тежи бесконачности, а кружна фреквенција нули и суме запишемо као интеграле, добићемо баш ове формуле:

$$f(x) = \lim_{T\to\infty}\sum_{n=-\infty}^{\infty}\frac{1}{T}(\int\limits_{-\frac{T}{2}}^{\frac{T}{2}} f(x)e^{-in\frac{2\pi}{T}x}\mathrm{d}x)e^{in\frac{2\pi}{T}x},\ \frac{2\pi}{T}\mapsto \mathrm{d}\omega,\ n\frac{2\pi}{T}\mapsto\omega$$

$$= \frac{1}{2\pi}\int\limits_{-\infty}^{\infty}(\int\limits_{-\frac{T}{2}}^{\frac{T}{2}} f(x)e^{-i\omega x}\mathrm{d}x)e^{i\omega x}\mathrm{d}\omega = \frac{1}{2\pi}\int\limits_{-\infty}^{\infty}F(\omega)e^{i\omega x}\mathrm{d}\omega$$

Представљање функције у уобичајеном облику називамо и облик у временском домену, јер ако је та функција сигнал, она је функција од времена, а Фуријеову трансформацију називамо обликом у фреквенцијском домену јер је она функција од фреквенције или кружне фреквенције, у зависности од записа.

1.3 Поасонова формула

Посматрајмо функцију $f(x) \in G(\mathbb{R})$ и њену Фуријеову трансформацију $\mathcal{F}(f(x)) = F(\omega)$. Ако је $f(x)$ таква да је њена периодична сумација $f_T(x) = \sum_{n=-\infty}^{\infty} f(x+nT)$ ограничена функција, тада важи

$$c_n = \frac{1}{T}F(\frac{2\pi n}{T})$$

за свако $n \in \mathbb{Z}$, где су c_n комплексни коефицијенти Фуријеовог развоја те периодичне сумације. Ово директно следи из формула за c_n и $F(\frac{2\pi n}{T})$ и применом смене $y = x + kT$:

$$c_n = \frac{1}{T}\int\limits_{-\frac{T}{2}}^{\frac{T}{2}} f_T(x)e^{-in\frac{2\pi}{T}x}\mathrm{d}x = \frac{1}{T}\int\limits_{-\frac{T}{2}}^{\frac{T}{2}}\sum_{k=-\infty}^{\infty} f(x+kT)e^{-in\frac{2\pi}{T}x}\mathrm{d}x =$$

$$= \frac{1}{T}\sum_{k=-\infty}^{\infty}\int\limits_{-\frac{T}{2}}^{\frac{T}{2}} f(x+kT)e^{-in\frac{2\pi}{T}x}\mathrm{d}x = \frac{1}{T}\sum_{k=-\infty}^{\infty}\int\limits_{kT-\frac{T}{2}}^{kT+\frac{T}{2}} f(y)e^{-in\frac{2\pi}{T}y}e^{-2\pi ik}\mathrm{d}y =$$

$$= \frac{1}{T}\sum_{k=-\infty}^{\infty}\int\limits_{kT-\frac{T}{2}}^{kT+\frac{T}{2}} f(y)e^{-in\frac{2\pi}{T}y}\cdot 1\cdot\mathrm{d}y = \frac{1}{T}\int\limits_{-\infty}^{\infty} f(y)e^{-in\frac{2\pi}{T}y}\mathrm{d}y =$$

$$= \frac{1}{T}F(n\frac{2\pi}{T}) = \frac{1}{T}F(\frac{2\pi n}{T})$$

Ако сада периодичну сумацију представимо преко њеног Фуријеовог развоја добијамо:

$$f_T(x) = \sum_{n=-\infty}^{\infty} c_n e^{in\frac{2\pi}{T}x},$$

$$\boxed{f_T(x) = \sum_{n=-\infty}^{\infty}\frac{1}{T}F(\frac{2\pi n}{T})e^{in\frac{2\pi}{T}x}} \tag{1.1}$$

Због релативне симетричности Фуријеове трансформације и њеног инверза веома слично се доказује и обрнуто тврђење:

$$\boxed{\sum_{k=-\infty}^{\infty} F(\omega + k\frac{2\pi}{T}) = T\sum_{n=-\infty}^{\infty} f(nT)e^{-inT\omega}} \tag{1.2}$$

Ове две формуле су веома значајне јер омогућавају да помоћу дискретног скупа информација о Фуријеовој трансформацији функције можемо извести периодичну сумацију оригиналне функције, и обрнуто, помоћу дискретног скупа информација о оригиналној функцији можемо одредити периодичну сумацију Фуријеове трансформације те функције, што нас директно уводи у следећу област.

2

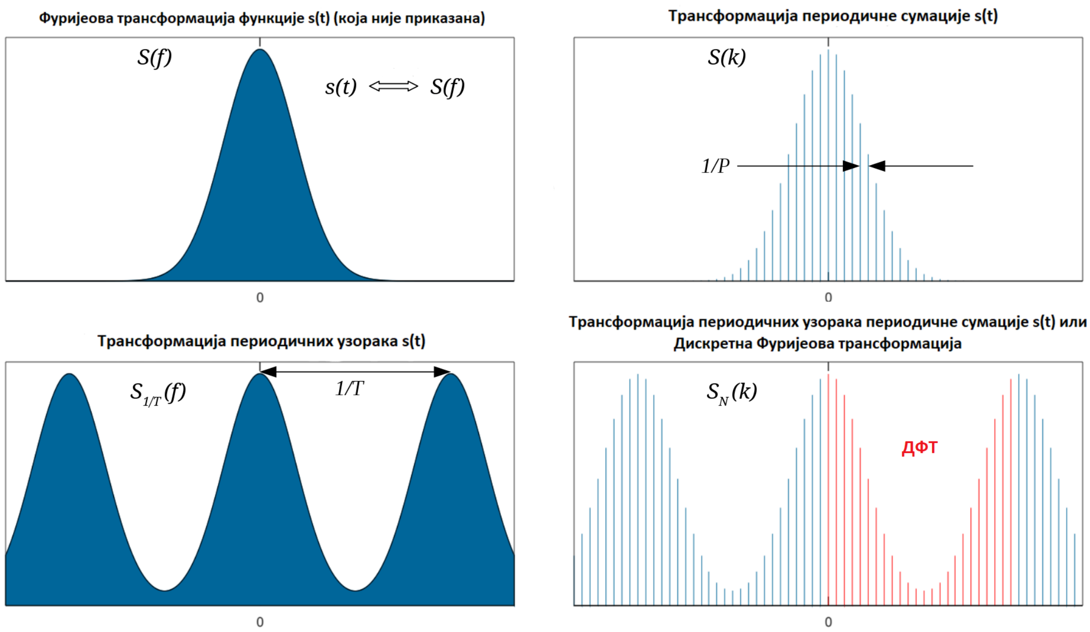
Дискретна Фуријеова трансформација

2.1 Увод и дефиниција

Коначни циљ нам је да помоћу рачунара обрађујемо сигнале (функције од времена). За то нам је потребно да радимо са коначним скупом информација. Дакле, као улаз не можемо имати континуалну функцију и не можемо је посматрати на бесконачном временском интервалу.

У стварности је већина сигнала ограниченог временског трајања, то јест можемо их представити функцијама које на неком интервалу $[0, L)$ имају вредност одговарајућег интензитета сигнала, док су ван тог интервала нула. То значи да ће нам периодична сумација те функције са периодом L подједнако добро говорити о тој функцији као и сама та функција. Користећи ту информацију и једнакост 1.1 закључујемо да су нам потребни само узорци Фуријеове трансформације у тачкама $\frac{2\pi k}{L}$, $k \in \mathbb{Z}$ да бисмо представили почетну функцију. Такође, када имамо велики број тачака, можемо добро апроксимирати Фуријеову трансформацију неком нумеричком методом, на пример полиномском интерполацијом.

Такође, мораћемо узорковати ту функцију, то јест, записати њене вредности у тачкама nT; $n \in [0, N)$; n, $N \in \mathbb{N}_0$. Углавном ћемо сматрати $NT = L$. Користећи једнакост 1.2 из тог низа узорака моћи ћемо да одредимо периодичну сумацију Фуријеове трансформације са периодом $1/T$. Доста сигнала има ограничен опсег фреквенција које садрже, на пример, запис људског говора је већином у звучном опсегу фреквенција. То значи да ако вршимо узорковање са довољно великом фреквенцијом, можемо сачувати готово све информације о Фуријеовој трансформацији те функције помоћу израчунате периодичне сумације Фуријеове трансформације.



Слика 2.1: Графички приказ различитих врста Фуријеове анализе

Уведимо сада ознаке $F[k] = \sum_{m=-\infty}^{\infty} F(\frac{2\pi k}{L} + m\frac{2\pi}{T})$ и $f[n] = f(nT)$. Уврштавањем у формулу 1.2, подељеном са T, добијамо:

$$F[k] = \sum_{n=0}^{N-1} f[n]e^{-i\frac{2\pi nk}{N}}$$

Сума има коначан број чланова јер сматрамо да је функција 0 ван датог интервала. Убрштавањем у формулу 1.1, коју притом множимо са T (јер смо преходну делили са T), добијамо:

$$f[n] = \frac{1}{N}\sum_{k=0}^{N-1} F[k]e^{i\frac{2\pi nk}{N}}$$

Поново, сума је коначна, јер сматрамо да смо све ненулте вредности Фуријеове трансформације укључили у овој суми њене периодичне сумације.

Можемо приметити још једну ствар, а то је да се свака периодична функција може представити као периодична сумација неке друге функције, те дефиницију ДФТ можемо лако проширити и на периодичне дискретне (или узорковане) функције, где сумирамо по једном периоду. Штавише, некад има смисла и рачунати ДФТ било каквих низова бројева, када се примењује за нешто што се не односи на анализу сигнала.

2.2 Дискретна конволуција

Посматрајмо две коначне и дискретне функције, односно њихова проширења $f[n]$, $g[m]$, $f[n] = 0$ за $n \notin [0, N]$, $g[m] = 0$ за $m \notin [0, M]$, $n, m \in \mathbb{Z}$, $N, M \in \mathbb{N}$. Операција дискретне линеарне конволуције се означава са $*$ и једнака је:

$$(f * g)[x] =_{\text{def.}} \sum_{k=-\infty}^{\infty} f[k]g[x-k] = \sum_{k=0}^{N+M} f[k]g[x-k] \tag{2.1}$$

Функција која се добија ће бити не-нулта у највише $N + M + 1$ тачака. Дискретна конволуција је у овом облику корисна, рецимо за представљање коефицијената производа два полинома, па ако умемо да ефикасно одредимо конволуцију, то значи да умемо ефикасно и да помножимо полиноме. У следећем одељку биће доказана теорема конволуције, то јест показано како се дискретна конволуција може одредити алгоритмима за одређивање ДФТ.

Ако су с друге стране $f[n]$ $g[m]$ периодичне са истим периодом $N \in \mathbb{N}$, сума из 2.1 би била бесконачна, те се дефиниција коригује тако да сумирамо само по једном периоду, и то називамо цикличном конволуцијом,

$$(f * g)[x] =_{\text{def.}} \sum_{k=0}^{N-1} f[k]g[x-k].$$

Резултат је такође периодичан са периодом N, па у њему имамо N битних вредности.

2.3 Теорема дискретне цикличне конволуције

Ова теорема је разлог због ког се конволуција помиње у контексту алгоритама за одређивање ДФТ. Ако са $\mathcal{F}$ и $\mathcal{F}^{-1}$ означимо редом ДФТ и инверзни ДФТ и ако су $f[n]$ и $g[n]$ периодичне функције периода N дефинисане над $\mathbb{Z}$, онда је

$$\mathcal{F}^{-1}(\mathcal{F}(f[n])\mathcal{F}(g[n])) = f * g.$$

Теорема се доказује на следећи начин:

$$\begin{aligned}\mathcal{F}^{-1}(\mathcal{F}(f[n])\mathcal{F}(g[n]))[x] &= \frac{1}{N}\sum_{k=0}^{N-1}((\sum_{n=0}^{N-1} f[n]e^{\frac{-2\pi i}{N}nk})(\sum_{m=0}^{N-1} g[m]e^{\frac{-2\pi i}{N}mk})e^{\frac{2\pi i}{N}xk}) = \\ &= \frac{1}{N}\sum_{k=0}^{N-1}\sum_{n=0}^{N-1}\sum_{m=0}^{N-1} f[n]g[m]e^{\frac{-2\pi i}{N}(n+m-x)k} = \\ &= \frac{1}{N}\sum_{n=0}^{N-1}\sum_{m=0}^{N-1}\sum_{k=0}^{N-1} f[n]g[m]e^{\frac{-2\pi i}{N}(n+m-x)k}\end{aligned}$$

Сада ћемо размотрити два случаја за унутрашњу суму.

1. $n + m \neq x$

$$\sum_{k=0}^{N-1} f[n]g[m]e^{\frac{-2\pi i}{N}(n+m-x)k} = f[n]g[m]\frac{e^{\frac{-2\pi i}{N}(n+m-x)N} - 1}{e^{\frac{-2\pi i}{N}(n+m-x)} - 1} = 0$$

2. $n + m = x$

$$\sum_{k=0}^{N-1} f[n]g[m]e^{\frac{-2\pi i}{N}(n+m-x)k} = \sum_{k=0}^{N-1} f[n]g[m]\cdot 1 = Nf[n]g[m]$$

Ову ситуацију можемо описати параметарски $n = t$, $m = x - t$

Коначно, следи:

$$\mathcal{F}^{-1}(\mathcal{F}(f[n])\mathcal{F}(g[n]))[x] = \frac{1}{N}\sum_{t=0}^{N-1} Nf[t]g[x-t] = \sum_{t=0}^{N-1} f[t]g[x-t] = (f * g)[x]$$

Последица ове теореме је да помоћу три ДФТ-а (два обична и једног инверзног), можемо одредити конволуцију функција.

3

Брза Фуриједова трансформација

3.1 Увод

У прошлом поглављу смо видели да узлазне податке ДФТ-а представља N комплексних бројева, од којих се сваки рачуна као сума N производа. Ово се тривијално рачуна директном применом формуле и алгоритмом сложености $\mathcal{O}(n^2)$. Брза Фуриједова трансофмација (на даље ФФТ) је било који алгоритам који ово ради са сложеношћу $\mathcal{O}(n \log n)$. Дакле, постоји више оваквих алгоритама, сваки са предностима и манама. Такође, треба напоменути да се сви ови алгоритми могу користити и за одређивање инверзне ДФТ, с обзиром да су ове две операције готово идентичне.

3.2 Основе Кули-Тјуки алгоритама

Овај алгоритам фунцкионише тако што улазни скуп података дели на мање скупове података, рекурзивно израчунава ФФТ тих мањих скупова, па израчунате ФФТ-ове користи за израчунавање почетног ФФТ-а. Сам алгоритам има неколико верзија у зависности од тога на колико мањих скупова се дели, али сви раде на истом математичком принципу.

Претпоставимо да се број улазних узорака функције N може представити као производ два природна броја $N_1, N_2 > 1$, дакле $N = N_1 N_2$. Делићемо улазне податке у групе на основу остатка при дељењу њиховог индекса са N_1. Сада ћемо формулу за израчунавање $F[k],\ k = N_2 k_1 + k_2,\ k_1 >= 0,\ 0 \leqslant k_2 < N_2$ записати мало другачије:

$$F[N_2k_1+k_2] = \sum_{n_1=0}^{N_1-1}\sum_{n_2=0}^{N_2-1} x_{N_1n_2+n_1} e^{\frac{-2\pi i}{N_1N_2}(N_1n_2+n_1)(N_2k_1+k_2)} =$$
$$= \sum_{n_1=0}^{N_1-1} \underbrace{e^{\frac{-2\pi i}{N}n_1k_2}}_{\text{ротирајући фактор}} \underbrace{\left(\sum_{n_2=0}^{N_2-1} x_{N_1n_2+n_1} e^{\frac{-2\pi i}{N_2}n_2k_2}\right)}_{k_2\text{-ти елемент ДФТ-а дужине } N_2} e^{\frac{-2\pi i}{N_1}n_1k_1}$$

Означени ротирајући фактор се често записује у облику $W_A^B,\ A,B\in\mathbb{N}_0$, где је $W_A = e^{\frac{-2\pi i}{A}}$, и представља један од комплексних A-тих корена јединице. Дакле, из ове формуле видимо да, ако смо рекурзивно израчунали дате ДФТ-ове дужине N_2, сваки елемент већег ДФТ-а можемо израчунати као суму N_1 чланова. Ово значи да ћемо у преласку између ова два рекурзивна нивоа имати N_1N операција сабирања. То значи да N_1 треба да буде што мањи и зато ћемо у сваком кораку за N_1 бирати прост број. Такође треба напоменути да цео израз представља један члан ДФТ-а дужине N_1, али с обзиром на то да се овај алгоритам углавном користи када су прости делиоци почетног броја мали, ове суме рачунамо тривијално. Ипак, ово може бити корисно када правимо хибридни алгоритам, чиме убрзавамо сваки ниво рекурзије.

Али шта ако почетни број има велике просте делиоце? Или шта ако у неком кораку рекурзије дођемо до простог броја, или је чак почетни број улазних бројева прост? Постоји више начина како решити овај проблем, један од њих је коришћење неког другог алгоритма, као на пример Блуштајновог.

Ако се ради о обради сигнала, можемо променити фреквенцију узорковања оригиналног сигнала, а најједноставнија метода јесте додавање нула (тзв. zero padding). Као што смо рекли на почетку поглавља о ДФТ, претпостављамо да је интензитет сигнала приближан нули ван датог интервала. Зато можемо да на крај низа узорака додамо произвољан низ нултих узорака. Притом овим не добијамо лошије решење, већ чак боље, јер је размак између узорака Фуријеове трансформације које добијамо ДФТ-ом обрнуто пропорционалан дужини интервала улаза.

Када имамо погодан број улазних података, можемо факторисати тај број и радити рекурзију, али се ова општија варијанта (у смислу да користимо разне просте делиоце) ређе користи. Кад већ можемо да бирамо број улазних података, ради једноставности алгоритма често се узимају степени малих простих пројева, нпр. 3^n, или најчешће 2^n. Варијанта овог алгоритма која увек улаз дели на два једнака дела назива се Кули-Тјуки алгоритам са основом 2, или лептир алгоритам. Овим у најгорем случајем дуплирамо величину улаза, и зато се алгоритам са мешовитом основом узима када је битна најоптималнија брзина, али је асимптотска сложеност иста.

3.3 Лептир алгоритам

Предуслов да бисмо радили ову варијацију Кули-Тјуки алгоритма јесте да је почетни број облика $N = 2^n, n \in \mathbb{N}_0$. Сада можемо математичке изразе за овај случај једноставније записати. Улазне податке раздвајамо на оне са парним и непарним индексима. Претпоставимо да је $0 <= k < N/2$. Тада је

$$F[k] = \sum_{n=0}^{N-1} x_k e^{\frac{-2\pi i}{N} nk} = \sum_{m=0}^{N/2-1} x_{2m} e^{\frac{-2\pi i}{N} 2mk} + \sum_{m=0}^{N/2-1} x_{2m+1} e^{\frac{-2\pi i}{N}(2m+1)k} =$$
$$= \underbrace{\sum_{m=0}^{N/2-1} x_{2m} e^{\frac{-2\pi i}{N/2} mk}}_{k\text{ти члан ДФТ парних}} + e^{\frac{-2\pi i}{N} k} \underbrace{\sum_{m=0}^{N/2-1} x_{2m+1} e^{\frac{-2\pi i}{N/2} mk}}_{k\text{ти члан ДФТ непарних}} = E[k] + e^{\frac{-2\pi i}{N} k} O[k]$$

Овде су E и O низови који редом представљају ДФТ-ове улазних елемената са парним, и оних са непарним индексима. Оба имају по $N/2$ чланова и зато смо ограничили k. Срећом, можемо искористити ове вредности за израчунвање и осталих $N/2$ $F[k]$. Користећи идентитете $e^{-2\pi i} = 1$ и $e^{-\pi i} = -1$ имамо:

$$F[k + N/2] = \sum_{m=0}^{N/2-1} x_{2m} e^{\frac{-2\pi i}{N/2} m(k+N/2)} + e^{\frac{-2\pi i}{N} k} \sum_{m=0}^{N/2-1} x_{2m+1} e^{\frac{-2\pi i}{N/2} m(k+N/2)} =$$
$$= E[k] - e^{\frac{-2\pi i}{N} k} O[k] = E[k] - w_N^k O[k]$$

Овде се $w_N = -e^{\frac{-2\pi i}{N}}$ назива се ротирајући фактор.

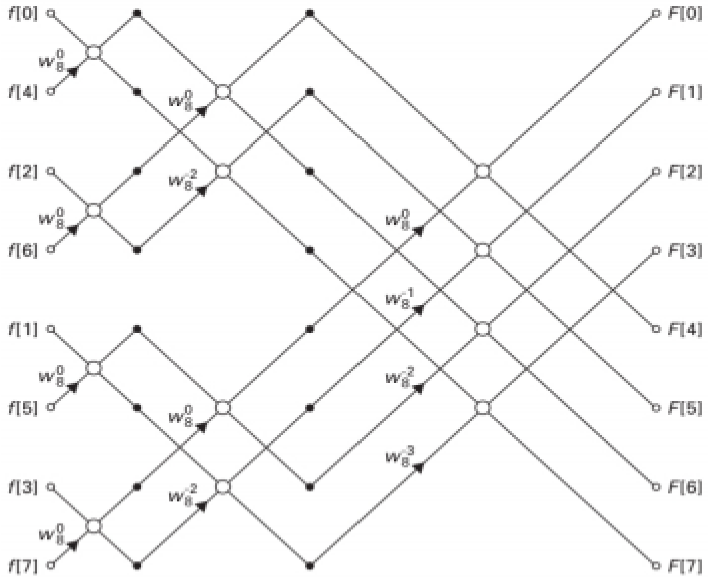
Сада, када имамо ове везе, можемо да направимо скицу алгоритма:

1. Ако је $N = 2$ рачунамо ДФТ на тривијалан начин, и излазимо из ове гране рекурзије

2. Раздвајамо улазне елементе на два низа, оне са парним и оне са непарним индексима

3. Рекурзивно израчунамо ДФТ ова два нова низа

4. Користећи формуле дате изнад израчунавамо ДФТ и враћамо резултат.



Слика 3.1: Шематски приказ лептир алгоритма

Временска сложеност $\mathcal{O}(n \log n)$ овог алгоритма се прилично лако доказује, и скоро је очигледна, с обзиром на то да имамо $\mathcal{O}(\log n)$ нивоа рекурзије и за сваки користимо $\mathcal{O}(n)$ операција.

За крај ћемо погледати како алгоритам ради на конкретном примеру. Задат нам је улазни низ:

$$[f[0], f[1], f[2], f[3], f[4], f[5], f[6], f[7]] = [0, 1, 0, 1, 1, 0, 1, 0]$$

Први корак је поделити улаз на две половине, по парности индекса, које ћемо прво одредити, а затим та решења искористити за добијање коначног резултата. Сада имамо два низа:

$$[0, 0, 1, 1], [1, 1, 0, 0]$$

Пошто базни случај представља ситуацију када низ има 2 елемента, оба низа ћемо поделити још једном:

$$[0, 1], [0, 1], [1, 0], [1, 0]$$

Сада рачунамо ДФТ сваког пара директно по формули, па добијамо:

$$[0 \cdot w_2^{0\cdot 0} + 1 \cdot w_2^{0\cdot 1},\ 0 \cdot w_2^{1\cdot 0} + 1 \cdot w_2^{1\cdot 1}], [0 \cdot w_2^{0\cdot 0} + 1 \cdot w_2^{0\cdot 1},\ 0 \cdot w_2^{1\cdot 0} + 1 \cdot w_2^{1\cdot 1}],$$

$$[1 \cdot w_2^{0\cdot 0} + 0 \cdot w_2^{0\cdot 1},\ 1 \cdot w_2^{1\cdot 0} + 0 \cdot w_2^{1\cdot 1}], [1 \cdot w_2^{0\cdot 0} + 0 \cdot w_2^{0\cdot 1},\ 1 \cdot w_2^{1\cdot 0} + 0 \cdot w_2^{1\cdot 1}]$$

што је, када се израчуна:

$$[1, -1], [1, -1], [1, 1], [1, 1]$$

Сада ћемо применити рекурзивну формулу за израчунавање ДФТ-ова дужине 4:

$$[1 + 1 \cdot w_4^0,\ \ -1 + (-1) \cdot w_4^1,\ 1 - 1 \cdot w_4^0,\ \ -1 - (-1) \cdot w_4^1],$$

$$[1 + 1 \cdot w_4^0,\ 1 + 1 \cdot w_4^1,\ 1 - 1 \cdot w_4^0,\ 1 - 1 \cdot w_4^1]$$

Добијамо:

$$[2, -1 + i, 0, -1 - i], [2, 1 - i, 0, 1 + i]$$

Исто тако добијамо ДФТ свих 8 елемената:

$$[2 + 2 \cdot w_8^0,\ (-1 + i) + (1 - i) \cdot w_8^1,\ 0 + 0 \cdot w_8^2,\ (-1 - i) + (1 + i) \cdot w_8^3,$$

$$2 - 2 \cdot w_8^0,\ (-1 + i) - (1 - i) \cdot w_8^1,\ 0 - 0 \cdot w_8^2,\ (-1 - i) - (1 + i) \cdot w_8^3]$$

Коначан резултат је:

$$[4,\ \ -1 + i(1 - \sqrt{2}),\ 0,\ \ -1 - i(1 + \sqrt{2}),\ 0,\ \ -1 + i(1 + \sqrt{2}),\ 0,\ \ -1 - i(1 - \sqrt{2})]$$

3.4 Блуштајнов алгоритам

У ситуацијама када желимо да одредимо ДФТ тачно одређене дужине, користан је Блуштајнов алгоритам. Он представља начин да било који ДФТ добијемо као цикличну конволуцију произвољне дужине. Ако уведемо ознаку за комплексни N-ти корен броја 1, $W = e^{\frac{2\pi i}{N}}$, имамо:

$$F[k] = \sum_{n=0}^{N-1} f[n] W^{-kn} = \sum_{n=0}^{N-1} f[n] W^{-kn} W^{\frac{1}{2}(n^2+k^2)} W^{-\frac{1}{2}(n^2+k^2)} =$$

$$= W^{-\frac{1}{2}k^2} \sum_{n=0}^{N-1} [x(n) W^{-\frac{1}{2}n^2}] W^{\frac{1}{2}(k-n)^2} = W^{-\frac{1}{2}k^2} (a * b)[k]$$

Дакле изразили смо овај ДФТ преко неке конволуције, где низови a и b представљају:

$$a_n = x[n]W^{-\frac{1}{2}n^2},\ n \in [0, N-1]$$

$$b_n = W^{\frac{1}{2}n^2},\ n \in [-N+1, N-1]$$

Ипак, ово и даље не можемо израчунати помоћу ДФТ, јер нам је потребна циклична конволуција, а не линеарна као што имамо. Срећом, решење за овај проблем је врло једноставно. Идеја је да направима две периодична низа чијих ће првих N чланова конволуције бити исти као тражених N чланова конволуције. Таква два низа постоје за свако $M \geqslant 2N-1$ и конструишу се на следећи начин (приказани су елементи од 0 до $M-1$, редом):

$$a_0, ..., a_{N-1},\ \underbrace{0, ..., 0}_{M-N \text{ нула}}$$

$$b_0, ..., b_{N-1},\ \underbrace{0, ..., 0}_{M-2N+1 \text{ нула}},b_{-N+1}, ...b_{-1}$$

Ова два низа очигледно испуњавају задате услове. Ова метода додавања нула и претварање линеарне конволуције у цикличну је корисна не само код од овог алгоритма. На пример, споменуто множење полинома степена N и M из претходног поглавља бисмо протворили у цикличну конволуцију додајући $M-1$ нула првом, и $N-1$ нула другом, како бисмо им изједначили дужине.

Пошто M може да буде било који број већи од $2N-1$, изабраћемо погодан број за неки други алгоритам, на пример степен броја 2. Потом ћемо конволуцију одредити неким другим алгоритмом користећи теорему о дискретној цикличној конволуцији, па из ње израчунати почетни ДФТ.

Као и у делу о лептир алгоритму, показаћемо како овај алгоритам функционише на конкретном примеру. Узмимо следећи улаз:

$$[f[0], f[1], f[2], f[3]] = [1, 0, 1, 0]$$

Формирајмо сада два низа за конволуцију:

$$a = [1 \cdot W^{-\frac{1}{2}0^2}, 0 \cdot W^{-\frac{1}{2}1^2}, 1 \cdot W^{-\frac{1}{2}2^2}, 0 \cdot W^{-\frac{1}{2}3^2}]$$

$$b = [W^{\frac{1}{2}3^2}, W^{\frac{1}{2}2^2}, W^{\frac{1}{2}1^2}, W^{\frac{1}{2}0^2}, W^{\frac{1}{2}1^2}, W^{\frac{1}{2}2^2}, W^{\frac{1}{2}3^2}]$$

Када то упростимо, и проширимо низове до дужине 8, добијамо:

$$a = [1, 0, -1, 0, 0, 0, 0, 0]$$

$$b = [1, \frac{\sqrt{2}}{2} + i\frac{\sqrt{2}}{2}, -1, \frac{\sqrt{2}}{2} + i\frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2} + i\frac{\sqrt{2}}{2}, -1, \frac{\sqrt{2}}{2} + i\frac{\sqrt{2}}{2}]$$

Овде нећемо приказивати како се добијају ови ДФТ-ови, јер смо лептир алгоритам већ видели на конкретном примеру. ДФТ-ови ова два низа су:

$$A = [0, 1 + i, 2, 1 - i, 0, 1 + i, 2, 1 - i]$$

$$B = [-1 + 2\sqrt{2} + i2\sqrt{2}, 1, 3, 1, -1 - 2\sqrt{2} - i2\sqrt{2}, 1, 3, 1]$$

Сада треба да сваки елемент из A помножимо са елементом на истој позицији из B и тако добијемо нови низ:

$$C = [0, 1 + i, 6, 1 - i, 0, 1 + i, 6, 1 - i]$$

Ако израчунамо инверзну ДФТ датог низа добијамо:

$$c = [2, 0, -2, 0, 1, 0, -1, 0]$$

Како бисмо добили почетни ДФТ треба узети прва 4 елемента овог низа и помножити их са одговарајућим ротирајућим факторима:

$$F = [2 \cdot W^{-\frac{1}{2}0^2}, 0 \cdot W^{-\frac{1}{2}1^2}, -2 \cdot W^{-\frac{1}{2}2^2}, 0 \cdot W^{-\frac{1}{2}3^2}]$$

$$F = [2, 0, 2, 0]$$

3.5 Хибридни ФФТ алгоритми

Сем алгоритама објашњених у овом поглављу, постоје још многи други ФФТ алгоритми, као што су Радеров алгоритам, алгоритам узајамно простих делилаца (prime-factor algorithm) и тако даље. Баш као што се Блуштајнов и, рецимо, лептир алгоритам користе заједно, најмодерније имплементације ФФТ користе све ове алгоритме, бирајући која комбинација је најбоља у зависности од улаза. У овом раду су представљени лептир и Блуштајнов алгоритам јер се већ помоћу ова два алгоритма може направити хбирид који испуњава услов сложености за свако N, а уједно су међу практичнијим и схватљивијим од ФФТ алгоритама.

4

Закључак

У овом раду је прво дат увод у Фуријеову анализу функција и сигнала са циљем да се боље и интуитивније схвати дискретна Фуријеова трансформација. Затим су објашњени дискретна Фуријеова трансформација, дискретна конволуција и веза између њих. На крају су детаљније објашњена два алгоритма за одређивање дискретне Фуријеове трансформације. Примена овог алгоритма је заиста широка, кроз рад је било пар назнака да иако се до дискретне Фуријеове трансформације првобитно дошло у сврху обраде функција и сигнала, примене има и за тотално различите ствари, на пример дискретну конволуцију можемо користити да бисмо израчунали производ два полинома.

За крај бих желео да се захвалим свом ментору и професору, др Радету Лазовићу, на указаној помоћи при изради рада, на одличном уводу у тему и за помоћ при налажењу литературе.

Литература

[1] Дискретне трансформације сигнала, примене и алгоритми, Катарина Ковачевић, Нови Сад, 2018

[2] https://mathworld.wolfram.com/FourierTransform.html, последњи пут прегледан 21. маја 2020.

[3] https://mathworld.wolfram.com/FourierSeries.html, последњи пут прегледан 21. маја 2020.

[4] https://en.wikipedia.org/wiki/Fourier_analysis, последњи пут прегледан 21. маја 2020.

[5] https://en.wikipedia.org/wiki/Poisson_summation_formula, последњи пут прегледан 21. маја 2020.

[6] Elementary Number Theory and Rader's FFT | SIAM Review | Vol. 59, No. 3 | Society for Industrial and Applied Mathematics

[7] https://en.wikipedia.org/wiki/Fast_Fourier_transform, последњи пут прегледан 21. маја 2020.

[8] https://en.wikipedia.org/wiki/Cooley-Tukey_FFT_algorithm, последњи пут прегледан 21. маја 2020.

[9] https://en.wikipedia.org/wiki/Convolution, последњи пут прегледан 21. маја 2020.

[10] https://ccrma.stanford.edu/ jos/mdft/Bluestein_s_FFT_Algorithm.html, последњи пут прегледан 21. маја 2020.

[11] The Design and Implementation of FFTW3, Matteo Frigo and Steven G. Johnson