Lineáris lépésszámú algoritmus $N=2^C$ hosszúságú diszkrét Fourier-transzformált számítására

DR. KOCSIS FERENC
BME Híradástechnikai Elektronika Intézet

DR. KOCSIS FERENC

1975-ben szerzett villamosmérnöki diplomát a BME Villamosmérnöki Karán, majd a Távközlési Kutató Intézetben kezdett dolgozni. Egyetemi doktori értekezését 1978-ban védte meg. 1983 szeptembere óta a BME HEI-ben dolgozik tudományos ösztöndíjasként, ahol a digitális jelfeldolgozás és jelszintézis algoritmikus kérdéseivel foglalkozik. Szakmai érdeklődési köre: rendszertechnika, digitális jelfeldolgozás, számítástechnika, algoritmusok elmélete.

ÖSSZEFOGLALÁS

A cikk $N=2^c$ hosszúságú adatsorozatok diszkrét Fourier transzformáltjának számítására mutat be új algoritmust. Az eljárás a fokozatos részekre osztáson alapszik: az N-pontos transzformációt egy N/2-pontos, egy N/4-pontos DFT és négy $\mathrm{mod}(z^{N/8}+1)$ szerinti polinomszorzat számítására bontja. A szükséges szorzások száma O(N).

1. Bevezetés

A dolgozatban a fokozatos részekre osztás elvét alkalmazó új számítási algoritmusról lesz szó. A számítási bonyolultságot a szükséges valós szorzások számával mérjük. Noha a jelfeldolgozó processzorok újabb generációjának megjelenésével (pl. TMS 32010) a korábban használt számítási modellek módosítására van szükség (azaz a legfontosabb bonyolultsági mérték a teljes műveletszám), bizonyos alkalmazásokban (pl. nagytömegű adat feldolgozása spektrumanalíziskor, geofizikában stb.) jól hasznosítható lineáris multiplikatív komplexitású diszkrét Fourier-transzformációs (DFT) algoritmus.

Az irodalomból jól ismert ([5]), hogy az $N=2^c$ hosszúságú DFT multiplikatív komplexitása $0(2^c)=2^{c+1}-c^2-3$, azonban nem ismert a gyakorlatban is jól használható $0(2^c)=0(N)$ komplexitású algoritmus nagyobb pontszámokra. Az első eredményről Duhamel és Hollmann cikke ([2]) számol be.

A javasolt algoritmus az eredeti 2^c-pontos transzformációt egyszerű algebrai átalakításokkal és kétváltozós számelméleti indextranszformáció alkalmazásával kisebb méretű transzformációk és valós periódikus konvolúciók, ill. polinomok $\mathrm{mod}(z^{N/8}+1)$ szerinti szorzatának számítására vezeti vissza.

2. Az algoritmus

Definíció szerint a DFT

$$(1)\qquad X(k)=\sum_{n=0}^{N-1} x(n)W_N^{kn} \quad 0\le k,\ n\le N-1$$

és $W_N^{kn}=\exp\left(-j\dfrac{2\pi}{N}kn\right)$. A továbbiakban $N=2^c$ és az $\{x(n)\}$ bemeneti adatok komplexek.

Az első lépés: a páros indexű transzformáltak számítása:

$$(2)\qquad X(2k)=\sum_{n=0}^{N-1} x(n)W_N^{2nk} \quad 0\le k\le N/2-1$$

Azonos algebrai átalakításokkal:

$$(3)\qquad X(2k)=\sum_{n=0}^{N/2-1} [x(n)+x(n+N/2)]W_{N/2}^{kn}$$

$$0\le k,\ n\le N/2-1$$

Azaz a páros indexű transzformált értékek $N/2$-pontos transzformációval számíthatók. Az új bemeneti adatvektor: $\{x(n)+x(n+N/2)\}$. Előállításához csupán komplex összeadásokra van szükség.

A második lépés: a páratlan indexű transzformáltak számítása:

$$(4)\qquad X(2k+1)=\sum_{n=0}^{N-1} x(n)w_N^{(2k+1)n} \quad 0\le k\le N/2-1$$

A páros és a páratlan indexű tagokon végzendő műveleteket szétválasztva:

$$(5)\qquad X(2k+1)=\sum_{n=0}^{N/2-1} x(2n)W_N^{(2k+1)2n}+$$

$$+\sum_{n=0}^{N/2-1} x(2n+1)W_N^{(2k+1)(2n+1)}=$$

$$=Y(2k+1)+Z(2k+1) \quad 0\le k,\ n\le N/2-1$$

Átrendezve az első összegzést:

$$(6)\qquad Y(2k+1)=\sum_{n=0}^{N/4-1} \{x'(2n)W_N^{2n}\}W_{N/4}^{nk}$$

ahol $x\ (2n)=x(2n)-x[2(n+N/4)]$
és $0\le k, n\le N/4-1$.

$Y(2k+L)$ az $\{x'(2n)W_N^{2n}\}$ $(0\le n\le N/4-1)$, $N/4$ pontból álló bemeneti vektor, diszkrét Fourier-transzformáltja. Az adatvektor előállításához $N/4$ db komplex kivonásra és $N/4-2$ db nem triviális

Beérkezett: 1986. V. 5. (□).

komplex szorzásra van szükség. Továbbá $Y(2k+1)=Y[2(k+N/4)+1]$, $0\leq k\leq N/4-1$. Eddig az algoritmus lépései majdnem azonosak a radix-2 FFT lépésével.

A harmadik lépés: $Z(2k+1)$ számítása. $W_N(2k+1)(2n+1)$ értékű kitevője és $N=2^c$ egymáshoz relatív prímek, ha $0\leq k$, $n\leq N/2-1$. Noha $N=2^c$ $(c>2)$ esetén nem létezik primitív gyök, relatív prímek esetére mindig található a maradékrendszert leíró transzformáció. Egy lehetséges választás [3]:

$$(7)\qquad 2n+1=(3)^{n_1}(-1)^{n_2} \bmod N \quad 0\leq n_1\leq N/4-1,\quad n_2=0,1$$

Ezen indextranszformáció felhasználásával az egydimenziós $Z(2k+1)$ kétdimenziós periódikus konvolucióvá alakítható. A transzformált indexek:

$$(8)\qquad \begin{aligned} 2n+1&=3^{n_1}(-1)^{n_2} \bmod N\\ 2k+1&=3^{-k_1}(-1)^{-k_2} \bmod N\\ (0\leq n_1, k_1&\leq N/4-1 \quad \text{és} \quad n_2, k_2=0,1)\end{aligned}$$

$$(9)\qquad Z(k_1,k_2)=\sum_{n_2=0}^{1}\sum_{n_1=0}^{N/4-1} x(n_1,n_2)W_N^{3^{n_1-k_1}(-1)^{n_2-k_2}}$$

A kapott kétdimenziós konvolúciót polinomiális alakban felírva:

$$(10)\qquad Z(z_1,z_2)= =H(z_1,z_2)X(z_1z_2) \bmod (z_1^{N/4}-1) \bmod (z_2^2-1)$$

ahol $H(z_1,z_2)=\sum_{n_2=0}^{1}\sum_{n_1=0}^{N/4-1} W_N^{3^{n_1}(-1)^{n_2}} z_1^{n_1}z_2^{n_2}$

A kimeneti és bemeneti adatokat, valamint a közbülső eredményeket más alakban felírva:

$$X(z_1,z_2)=\sum_{n_1=0}^{N/4-1} X(n_1,0)z_1^{n_1}+z_2\sum_{n_1=0}^{N/4-1} X(n_1,1)z_1^{n_1}$$

$$Z(z_1,z_2)=\sum_{n_1=0}^{N/4-1} Z(k_1,0)z_1^{n_1}+z_2\sum_{n_1=0}^{N/4-1} Z(k_1,1)z_1^{n_1}$$

$$H(z_1,z_2)=\sum_{n_1=0}^{N/4-1} W^{N3^{n_1}}z_1^{n_1}+z_2\sum_{n_1=0}^{N/4-1} W_N^{-3^{n_1}}z_1^{n_1}$$

$$(11)\qquad \begin{aligned} X(z_1,z_2)&=X_0(z_1)+z_2X_1(z_1)\\ Z(z_1,z_2)&=Z_0(z_1)+z_2Z_1(z_1)\\ H(z_1,z_2)&=H(z_1)+z_2\overline{H}(z_1)\end{aligned}$$

„$^-$" a komplex konjugálást jelöli. Az $X(z_1,z_2)$ és $Z(z_1,z_2)$ polinomok együtthatóinak az $\{x(2n+1)\}$ és $\{Z(2k+1)\}$ vektorokból való előállítására csupán permutációkra van szükség. Először a mod (z_2^2-1) értékeket számítva:

$$(12)\qquad \begin{aligned} Z_0(z_1)&=[H(z_1)X_0(z_1)+\overline{H}(z_1)X_1(z_1)] \bmod (z_1^{N/4}-1)\\ Z_1(z_1)&=[\overline{H}(z_1)X_0(z_1)+H(z_1)X_1(z_1)] \bmod (z_1^{N/4}-1)\end{aligned}$$

Azonban $z_1^{N/4}-1=(z_1^{N/8}-1)(z_1^{N/8}+1)$. $Z_0(z_1)$ és

1. táblázat

Nem triviális szorzások száma

N	Új algoritmus	Winograd formula
8	4	4
16	20	20
32	70	108
64	192	267
2^7	476	612
2^8	1106	1335
2^9	2468	2820
2^{10}	6356	5835
2^{11}	11398	11916
2^{12}	23192	24135

$Z_1(z_1)$ meghatározható a polinomokra vonatkozó kínai maradéktétel felhasználásával ([3], [6], [9]) Vizsgáljuk először $H(z_1) \bmod (z_1^{N/8}-1)$ értékét. $H(z_1)$ definíciójából kiindulva

$$\left[H(z_1)=\sum_{n_1=0}^{N/4-1} W_N^{3^{n_1}}z_1^{n_1}\right]:$$

$$(13)\qquad H(z_1) \bmod (z_1^{N/8}-1)=\sum_{n_1=0}^{N/8-1}(W_N^{3^{n_1}}+W_N^{3^{n_1}+N/8})z_1^{n_1}$$

3^{n_1} értékét más alakban írva: $3^{n_1}=a\cdot 2^c+b$, ahol b nyilvánvalóan páratlan. Ismert tétel szerint [8] azonban $3^{N/8}=1+2^{c-1}+d\cdot 2^c$ alakban írható (a, b és d egészek, továbbá $N=2^c$). A (13) alatti összegzés együtthatói:

$$(14)\qquad W_N^{3^{n_1}}+W_N^{3^{n_1}+N/8}=W_N^b(1+W_N^{b\cdot 2^{c-1}})= =W_N^b[1+(-1)^b]=0 \quad (0\leq n_1<N/8) \text{ azaz}$$

$H(z_1) \bmod (z_1^{N/8}-1)=0$, ill. $\overline{H}(z_1) \bmod (z_1^{N/8}-1)=0$.

Következésképp $Z_0(z_1) \bmod (z_1^{N/8}-1)=0$

és $Z_1(z_1) \bmod (z_1^{N/8}-1)=0$,

vagyis $Z(z_1,z_2)$ meghatározásához csupán $H(z_1) \bmod (z^{N/8}+1)$, $\overline{H}(z_1) \bmod (z_1^{N/8}+1)$, $X_0(z_1) \bmod (z_1^{N/8}+1)$ és $X_1(z_1) \bmod (z_1^{N/8}+1)$ számítására van szükség [4 valós mod $(z^{N/8}+1)$ szerinti polinominális szorzat]. Mivel a $(z_1^{N/8}+1)$ polinom irreducibilis a racionális számok terében, a szükséges valós szorzások száma $4[2(N/8)-1]=N-4$. A $Z_0(z_1)$ és $Z_1(z_1)$ értékét meghatározó összefüggések (segédváltozók bevezetésével):

$$(15)\qquad \begin{aligned} m_1&=\frac{1}{2}H(z_1) \bmod (z_1^{N/8}+1)\\ m_2&=\frac{1}{2}\overline{H}(z_1) \bmod (z_1^{N/8}+1)\\ p_1&=X_0(z_1) \bmod (z_1^{N/8}+1)\\ p_2&=X_1(z_1) \bmod (z_1^{N/8}+1)\\ q_1&=(m_1+m_2)(p_1+p_2)\\ q_2&=(m_1-m_2)(p_1-p_2)\end{aligned}$$

Az eredmény:

$$Z_0(z_1)=(q_1+q_2) \bmod (z_1^{N/8}+1) \tag{16}$$
$$Z_1(z_1)=(q_1-q_2) \bmod (z_1^{N/8}+1)$$

Az m_1 tényező tisztán valós, míg az m_2 tisztán képzetes. Így $Z(2k+1)$ számításához $(N-4)$ valós szorzásra van szükség.

3. Az algoritmus bonyolultsága

Az előzőek alapján már meghatározható a szükséges valós szorzások száma (egy komplex szorzást három valós szorzással számítva). Rekurzív formában:

$$0(2^c)=0(2^{c-1})+0(2^{c-3})+3(N/4-2)+N-4 \tag{17}$$

A differenciaegyenlet kezdőértékei: $0(2^3)=4$ és $0(2^4)=20$ (a Winograd-féle rövid optimális DFT algoritmusok). Az eredményeket táblázatba foglalva (1. táblázat) $c=12$ értékig és összehasonlítva a Winograd-összefüggésből $(0(2^c)=3(2^{c-1}-c^2-3)=$ $=0(N))$ kapott értékekkel látható, hogy a javasolt algoritmushoz valamivel kevesebb szorzásra van szükség, mint a lineáris Winograd-eljáráshoz.

4. Következtetések

Új, rekurzív módszert javasoltunk az $N=2^c$ hosszúságú DFT számítására. Az algoritmus bonyolultságát a szükséges valós szorzások számával mértük (nem számolva az egyéb aritmetikai és adatmozgatási műveleteket). Az azonos hosszúságú transzformáltakat számító radix-4 FFT programokhoz képest [7], az első kísérleti eredmények (N=32 és 64 hosszúságú transzformációk a TMS 32010 mikroprocesszoron programmal megvalósítva) mintegy 10%-os sebességnövekedést mutatnak.

IRODALOM

[1] *Aho, A. V.—Hopcroft, J. E.—Ullmann, J. D.:* „The Design and Analysis of Computer Algorithms", Addison-Wesley, 1970.

[2] *Duhamel, P.—Hollmann, H.:* „Existenoe of a 2^n FFT Algorithm with a Number of Multiplioations Lower than 2^{n+1}", Electron. Letters, vol. 20, no. 17, pp. 690—698, Aug. 16, 1984.

[3] *Rader, C. M.—McClellan, J. H.:* „Number Theory in Digital Signal Processing", Prentice-Hall, Englewood Cliffs, 1979.

[4] *Winograd, S.:* „On the Multiplicative Complexity of of the Discrete Fourier Transform", Advances in Mathematics, vol. 32, no. 2, pp. 83—117, May 1979.

[5] *Winograd, S.:* „Signal Processing and Complexity of Computation", Proc ICASSP'80, pp. 94—101, Apr. 9—11.

[6] *Kocsis, F.:* „Gyors eljárások a DFT számítására 1—2—3.", Híradástechnika, vol. 35, 1984/12, pp. 544—549, vol. 36, 1985/1, pp. 31—39, vol. 36, 1985/5, pp. 214—219.

[7] *Morris, L. R.:* „Structural Considerations for Large FFT Programs on the TI TMS 32010 DSP Microchip", Proc. ICASSP'85, pp. 1648—1651.

[8] *Niven, I.—Zuckermann, H. S.:* „Bevezetés a számelméletbe", Műszaki Könyvkiadó, Budapest, 1978.

[9] *Nussbaumer, H. J.:* „Fast Fourier Transform and Convolution Algorithms", Springer-Verlag, Berlin—Heidelberg—New York, 1981.