

Zobecněný Goertzelův algoritmus pro neceločíselné násobky základního harmonického kmitočtu

Goertzel algorithm generalized for non-integer multiples of
base frequency

Petr Sysel, Pavel Rajmic

sysel@feec.vutbr.cz, rajmic@feec.vutbr.cz

Fakulta elektrotechniky a komunikačních technologií VUT v Brně

Abstrakt: Článek se zabývá Goertzelovým algoritmem pro zjištění modulu a fáze harmonických složek signálu. Zaměřuje se na výhody klasického Goertzelova algoritmu oproti výpočtu pomocí diskrétní Fourierovy transformace. Dále je v článku odvozeno zobecnění Goertzelova algoritmu, které umožní jeho použití i pro neceločíselné násobky základního harmonického kmitočtu diskrétní Fourierovy transformace.

Abstract: The paper describes the Goertzel algorithm for determining the modulus and phase of the harmonics which are present in a signal. The paper emphasizes the advantages of the Goertzel algorithm compared to the discrete-time Fourier transform. Generalization of the algorithm, which allows us to use it also for non-integer multiples of the basic frequency, is presented as well.

Zobecněný Goertzelův algoritmus pro neceločíselné násobky základního harmonického kmitočtu

Petr Sysel, Pavel Rajmic

Ústav telekomunikací, Fakulta elektrotechniky a komunikačních technologií, VUT v Brně
Email: sysel@feec.vutbr.cz, rajmic@feec.vutbr.cz

Článek se zabývá Goertzelovým algoritmem pro zjištění modulu a fáze harmonických složek signálu. Zaměřuje se na výhody klasického Goertzelova algoritmu oproti výpočtu pomocí diskrétní Fourierovy transformace. Dále je v článku odvozeno zobecnění Goertzelova algoritmu, které umožní jeho použití i pro neceločíselné násobky základního harmonického kmitočtu diskrétní Fourierovy transformace.

1 Úvod

V případě diskrétního signálu se pro harmonickou analýzu používá diskrétní Fourierova transformace – DFT, viz definice (1). Kmitočty jednotlivých harmonických složek v DFT se vždy odvíjejí od délky transformace N a jsou celočíselnými násobky základního kmitočtu $\Delta f = \frac{f_{vz}}{N}$, kde f_{vz} představuje hodnotu vzorkovacího kmitočtu. Veličina Δf tedy udává kmitočtovou rozlišovací schopnost DFT. Pokud se délka transformace N neshoduje s periodou signálu, pak signál obsahuje složky o kmitočtech, který nejsou celočíselnými násobky základního harmonického kmitočtu, a dojde k prosakování spektra, což má za následek zkreslení hodnot složek DFT na celočíselných násobcích Δf .

I v případě, kdy chceme určit modul a fázi pouze jedné nebo několika harmonických složek, je nutné délku transformace N volit především s ohledem na požadovanou přesnost určení kmitočtu. Přitom výpočetní náročnost DFT roste přibližně kvadraticky s délkou transformace N a navíc většina vypočítaných hodnot je bez užítku zahozena. Pro takové případy byl odvozen Goertzelův algoritmus, který efektivně vypočítá hodnotu jedné složky DFT. Nevýhoda zaokrouhlování kmitočtu na nejbližší celočíselné násobky základního harmonického kmitočtu a prosakování spektra však setrvává.

V kapitole 2 nejprve ukážeme odvození klasického Goertzelova algoritmu. To je uděláno podrobně, neboť v části týkající se zobecnění bude zapotřebí se na jednotlivé kroky odvolávat. Provedeme srovnání výpočetní a paměťové náročnosti Goertzelova algoritmu a algoritmu rychlé Fourierovy transformace FFT. V kapitole 3 pak provedeme rozšíření Goertzelova algoritmu tak, že nebude nutné zaokrouhlovat kmitočty na celočíselné násobky základního harmonického kmitočtu. Takovéto zobecnění už bylo zmíněno, viz např. [2], ale je zde počítán pouze modul, nikoliv fáze harmonické složky.

1.1 Typický příklad použití Goertzelova algoritmu – DTMF

Goertzelův algoritmus se běžně používá pro detekci tónové volby DTMF (Dual-Tone Multi-Frequency) v telefonní technice, kde volba je určena pomocí současným zazněním dvou z celkem osmi kmitočtů [8]. Hodnoty kmitočtu každého ze dvou čtveřic signalizačních tónů byly zvoleny tak, aby kmitočty jejich vyšších harmonických nebo intermodulačních produktů byly od nich dostatečně vzdáleny. Zvolené kmitočty mají velmi vysokou hodnotu nejmenšího společného násobku. Při použití digitálního přijímače se vzorkovacím kmitočtem 8 kHz je tak perioda DTMF signálu několik desítek tisíc vzorků. V praxi se ovšem délka transformace N musí volit mnohem menší, a tak k prosakování spektra vždy dojde. Např. při $N = 205$ se namísto modulu přesného kmitočtu 770 Hz hledá na kmitočtu přibližně 780,5 Hz.

Hodnota $N = 205$ je v praxi často volena [6], neboť suma kvadrátů relativních odchylek signalizačních kmitočtů nabývá jednoho z lokálních minim právě pro tuto délku. Při této délce je odchylka přibližně rovna 1,4 %, přičemž povolená odchylka kmitočtu vysílače je 1,8 %. V některých aplikacích Goertzelova algoritmu může být tato systémová odchylka od přesného kmitočtu již za povolenou toleranci a Goertzelův algoritmus by nebylo možné použít. S přístupem ukázaným v tomto článku není vůbec nutné provádět zaokrouhlení detekovaného kmitočtu a je možné z konečně dlouhého signálu zjistit amplitudu i fázi složky o libovolném (i neceločíselném) kmitočtu. Počet operací přitom vzroste jen zanedbatelně.

1.2 Značení v textu

V textu uvažujeme diskrétní číslicový vstupní signál x délky N , tedy $\{x[n]\} = \{x[0], x[1], \dots, x[N-1]\}$. Symbol k bude značit pořadí harmonické složky v diskrétní

Fourierově analýze, tedy $k \in \mathbb{N}$. V pozdějších částech textu budeme pracovat i s $k \in \mathbb{R}$. Signál jednotkového skoku bude značen $\{u[n]\}$: $u[n] = 1$ pro $n \geq 0$, $u[n] = 0$ pro $n < 0$.

2 Klasický Goertzelův algoritmus

2.1 Odvození Goertzelova algoritmu

Algoritmus pocházející od G. Goertzela [3] slouží k výpočtu k -té složky DFT signálu $\{x[n]\}$ délky N , tj.

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi k \frac{n}{N}}, \quad k = 0, \dots, N-1. \quad (1)$$

Vynásobíme-li pravou stranu rovnice číslem $1 = e^{j2\pi k \frac{N}{N}}$, dostáváme ekvivalentně

$$X[k] = e^{j2\pi k \frac{N}{N}} \sum_{n=0}^{N-1} x[n] e^{-j2\pi k \frac{n}{N}}, \quad (2)$$

po úpravě pak

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j2\pi k \frac{n-N}{N}}. \quad (3)$$

Definujeme-li nyní posloupnost $\{h_k[n]\}$ s prvky $h_k[\ell] = e^{j2\pi k \frac{\ell}{N}} u[\ell]$, pak pravou stranu vyjádření (3) lze chápat jako diskretní lineární konvoluci signálů $\{x[n]\}$ a $\{h_k[n]\}$. Vskutku, neboť označíme-li výsledek této konvoluce $\{y_k[n]\}$, platí pro její prvky:

$$y_k[m] = \sum_{n=-\infty}^{\infty} x[n] h_k[m-n], \quad (4)$$

což lze s uvážením ohraničeného nosiče signálu $\{x[n]\}$ přepsat na

$$y_k[m] = \sum_{n=0}^{N-1} x[n] e^{j2\pi k \frac{m-n}{N}} u[m-n]. \quad (5)$$

Srovnáním (3) a (5) vidíme, že hledané $X[k]$ je vlastně N -tý vzorek této konvoluce, neboli

$$X[k] = y_k[N] \quad (6)$$

pro jakékoli fixně zvolené $k = 0, \dots, N-1$. Znamená to, že požadovanou hodnotu lze obdržet jako výstupní vzorek IIR lineárního systému s impulzní odezvou $\{h_k[n]\}$ v čase N .

Odvodíme nyní přenosovou funkci $H_k(z)$ tohoto sys-

tému; je to transformace \mathcal{Z} impulzní odezvy [1], tedy

$$H_k(z) = \sum_{n=-\infty}^{\infty} h_k[n] z^{-n} \quad (7)$$

$$= \sum_{n=-\infty}^{\infty} e^{j2\pi k \frac{n}{N}} u[n] z^{-n} \quad (8)$$

$$= \sum_{n=0}^{\infty} e^{j2\pi k \frac{n}{N}} z^{-n} \quad (9)$$

$$= \sum_{n=0}^{\infty} \left(e^{j2\pi \frac{k}{N}} z^{-1} \right)^n, \quad (10)$$

což je geometrická řada, jejíž první člen je $e^{j2\pi k \frac{0}{N}} z^{-0} = 1$ a kvocient $q = e^{j2\pi \frac{k}{N}} z^{-1}$. Pro $|q| < 1$, tj. $|z| > 1$ je konvergentní se součtem, který je hledanou přenosovou funkcí:

$$H_k(z) = \frac{1}{1 - e^{j2\pi \frac{k}{N}} z^{-1}}. \quad (11)$$

Příslušná diferenciální rovnice je

$$y_k[n] = x[n] + e^{j2\pi \frac{k}{N}} y_k[n-1], \quad \text{při } y_k[-1] = 0. \quad (12)$$

Tato diferenciální rovnice prvního řádu však obsahuje násobení komplexním koeficientem, což je výpočetně nevýhodné. Pro úsporu výpočetní náročnosti se příslušná přenosová funkce rozšíří v čitateli i jmenovateli o faktor komplexně sdružený k $z = e^{j2\pi \frac{k}{N}}$ na

$$H_k(z) = \frac{1 - e^{-j2\pi \frac{k}{N}} z^{-1}}{(1 - e^{-j2\pi \frac{k}{N}} z^{-1})(1 - e^{j2\pi \frac{k}{N}} z^{-1})} \quad (13)$$

$$= \frac{1 - e^{-j2\pi \frac{k}{N}} z^{-1}}{1 - 2 \cos\left(\frac{2\pi k}{N}\right) z^{-1} + z^{-2}}. \quad (14)$$

Příslušná diferenciální rovnice tohoto IIR systému druhého řádu je

$$y_k[n] = x[n] - x[n-1] e^{-j2\pi \frac{k}{N}} + 2 \cos\left(\frac{2\pi k}{N}\right) y_k[n-1] - y_k[n-2] \quad (15)$$

při $x[-1] = y[-1] = y[-2] = 0$. Tuto strukturu lze popsat pomocí vnitřních stavových proměnných

$$s[n] = x[n] + 2 \cos\left(\frac{2\pi k}{N}\right) s[n-1] - s[n-2], \quad (16)$$

přičemž výstup je popsán rovnicí

$$y_k[n] = s[n] - e^{-j2\pi \frac{k}{N}} s[n-1] \quad (17)$$

a klademe $s[-1] = s[-2] = 0$. Graf signálových toků tohoto systému je vidět na obr. 1.

Stavový popis je výhodný z toho důvodu, že nás zajímá až výstup $y[N]$, a tudíž systém (16), kde se pracuje jen s reálnými čísly, iterujeme $(N+1)$ -krát (začínáme

Rovnice (18) znamená, že výpočet spektrálních hodnot je rychlejší než při použití celého FFT v případě, že K analyzovaných kmitočtů je méně než $\log_2 N$. Např. pro $K = 5$ kmitočtů leží ona hranice výhodnosti na úrovni $N = 2^5 = 32$ vzorků.

Při použití algoritmu FFT je nutné v paměti vyhradit místo minimálně pro $2N$ vzorků reprezentujících reálnou a imaginární složku celého bloku signálu. Dále se často předpočítají a do paměti uloží hodnoty jádra transformace \sin a \cos , kterých je celkem N . Výpočet FFT sice může probíhat bez přesouvání hodnot v paměti (zpracování in-place), ale s ohledem na možnost zahájit zpracování až po příchodu celého bloku a s tím spojené zpoždění výpočtu, je často použita vyrovnávací paměť o velikosti minimálně jednoho bloku $2N$. V případě reálných signálů postačuje velikost N . Celková paměťová náročnost algoritmu FFT je tak $4N$ pro reálné signály.

U Goertzelova algoritmu je nutné v paměti vyhradit místo pro uložení dvou stavových proměnných pro každý kmitočet, předpočítané hodnoty reálné a imaginární složky konstanty C pro každý kmitočet a reálnou a imaginární složku výsledné hodnoty. Pro vstupní signál není nutné použít vyrovnávací paměť, protože výpočet může probíhat současně s příchodem vzorků. Podobně výstupní signál je nutné přepsat až po příchodu posledního vzorku. V mnoha případech tak nebude nutné použít vyrovnávací paměť ani pro výstup. Celková paměťová náročnost Goertzelova algoritmu je tak $6K$.

Z hlediska paměťové náročnosti tak bude Goertzelův algoritmus výhodnější, pokud

$$\begin{aligned} 6K &< 4N \\ K &< \frac{2}{3}N, \end{aligned} \quad (19)$$

což bude ale splněno vždy, když bude splněna podmínka (18), neboť $\frac{2}{3}N > \log_2 N$.

3 Zobecněný Goertzelův algoritmus

Zdůrazněme, že vztah (2) platí pouze pro celočíselná k . Pouze v takovém případě délce vstupního signálu N odpovídá celistvý počet period transformačního jádra $e^{-j2\pi k \frac{n}{N}}$. V případě $k \in \mathbb{R}$ již ekvivalence rovnic (1) a (2) není zaručena. To souvisí s tím, že perioda transformačního jádra již nemusí odpovídat délce signálu, a tedy klasický algoritmus již není možné použít.

3.1 Rozšíření pro neceločíselná k

Pokud k není celé číslo, nýbrž reálné, nelze vlastně už mluvit o DFT (1), ale pouze o Fourierově transformaci s diskretním časem (DTFT), kterou vyjadřuje vzorec

$$X(\omega) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n}, \quad \omega \in \mathbb{R}. \quad (20)$$

Při označení $\omega_k = 2\pi \frac{k}{N}$ můžeme tedy psát

$$X(\omega_k) = \sum_{n=-\infty}^{\infty} x[n] e^{-j2\pi k \frac{n}{N}} \quad (21)$$

$$= \sum_{n=0}^{N-1} x[n] e^{-j2\pi k \frac{n}{N}}, \quad k, \omega_k \in \mathbb{R}, \quad (22)$$

s uvážením předpokladu konečného nosiče signálu $\{x[n]\}$.

Postup odvozením zobecněného Goertzelova algoritmu je zcela analogický s částí 2. Oproti předchozímu však rovnici (22) na začátku rozšíříme jedničkou ve tvaru

$$e^{j2\pi k \frac{N}{N}} \cdot e^{-j2\pi k \frac{N}{N}} = 1 \quad \text{pro } k \in \mathbb{R}, \quad (23)$$

čímž dostáváme

$$X(\omega_k) = e^{j2\pi k \frac{N}{N}} \cdot e^{-j2\pi k \frac{N}{N}} \sum_{n=0}^{N-1} x[n] e^{-j2\pi k \frac{n}{N}} \quad (24)$$

$$= e^{-j2\pi k \frac{N}{N}} \sum_{n=0}^{N-1} x[n] e^{j2\pi k \frac{N-n}{N}} \quad (25)$$

$$= e^{-j2\pi k} \sum_{n=0}^{N-1} x[n] e^{j2\pi k \frac{N-n}{N}}. \quad (26)$$

Všimněme si, že suma ve výrazu (26) je shodná s (3). A proto odvození zobecněného algoritmu může kopírovat odvození v části 2.1, s jedinou změnou, a to, že rovnice charakterizující výstup pomocí stavových proměnných (17) bude nyní mít tvar

$$y_k[n] = \left(s[n] - e^{-j\frac{2\pi k}{N}} s[n-1] \right) \cdot e^{-j2\pi k}. \quad (27)$$

Skutečně je tomu tak, protože korekční konstanta $e^{-j2\pi k}$ závisí pouze na indexu kmitočtové složky, který je pro každou složku konstantní po celou dobu výpočtu. Komplexní konstanta je přitom pro $k \in \mathbb{N}$ rovna jedničce, a tedy skutečně se jedná o zobecnění. Tedy vlastně jediná změna oproti klasickému algoritmu je závěrečné vynásobení touto konstantou.

Konstanta $e^{-j2\pi k}$ ovlivňuje pouze fázi výsledku, nikoliv modul. To mj. znamená, že pokud chceme určit pouze modul složky s neceločíselným k , postačuje použít původní Goertzelův algoritmus. A také se tak někdy používá, viz např. [2]. V případech, kdy je nutné znát i fázi, např. chceme určit zpoždění signálu, je použití této konstanty nutné.

3.2 Redukce počtu iterací

V této části ukážeme, že poslední iteraci Goertzelova algoritmu není třeba provádět obvyklým způsobem, ale lze ji nahradit pouze jedním komplexním násobením.

Z rovnice (5) můžeme vyjádřit

$$\begin{aligned}
 y_k[N] &= \sum_{n=0}^{N-1} x[n] e^{-j2\pi k \frac{n-N}{N}} u[N-n] \\
 &= \sum_{n=0}^{N-1} x[n] e^{-j2\pi k \frac{n-N}{N}} \quad (28)
 \end{aligned}$$

a rovněž

$$\begin{aligned}
 y_k[N-1] &= \sum_{n=0}^{N-1} x[n] e^{-j2\pi k \frac{n-(N-1)}{N}} u[(N-1)-n] \\
 &= \sum_{n=0}^{N-1} x[n] e^{-j2\pi k \frac{n-N}{N}} e^{-j2\pi k \frac{1}{N}}. \quad (29)
 \end{aligned}$$

Srovnáním (28) a (29) dostáváme vztah mezi posledními dvěma vzorky konvoluce:

$$y_k[N] = y_k[N-1] \cdot e^{+j2\pi \frac{k}{N}}. \quad (30)$$

To znamená, že poslední iteraci obvyklého Goertzelova algoritmu je možné nahradit pouhým násobením konstantou $e^{j2\pi \frac{k}{N}}$. Vztah (30) platí pro $y_k[N]$ a $y_k[N-1]$ kvůli omezenému nosiči $x[n]$. Pro vzorky $y_k[N-1]$ a $y_k[N-2]$ už nic takového neplatí, díky členu $u[\cdot]$.

Dáme-li dohromady toto číslo s konstantou korigující fází pro necelá k , viz (27) v části 3.1, získáme celkovou korekční konstantu

$$e^{-j2\pi k} \cdot e^{+j2\pi \frac{k}{N}} = e^{-j2\pi \frac{k}{N}(N-1)}. \quad (31)$$

Tímto způsobem tedy můžeme získat zkrácený zobecněný algoritmus, který shrnujeme v obr. 3.

3.3 Výpočetní a paměťová náročnost

Výpočetní náročnost zobecněného Goertzelova algoritmu dle části 3.1 (bez zkrácení podle 3.2) se oproti klasickému algoritmu zvedne o 1 komplexní násobení (tzn. 4 reálná násobení a 2 reálné součty). Paměťová náročnost se zvýší o 2 pozice, které obsahují reálnou a imaginární část korekční konstanty.

Samotné zkrácení iteračního cyklu dle 3.2 vede sice k úspoře 2 součtů a 1 násobení, avšak násobení komplexní konstantou tento benefit znehodnocuje. To znamená, že zkracovat cyklus v případě celých k nemá smysl a je výhodnější zůstat u obvyklého algoritmu podle části 2.

Pokud zobecnění dle 3.1 spojíme se zkrácením dle 3.2, výpočetní náročnost poklesne o 2 součty a 1 násobení, ale na druhou stranu díky násobení komplexní konstantou (31) způsobí její zvýšení o 1 komplexní násobení. Celkově tedy máme nárůst o 3 reálná násobení. Co se týká paměťové náročnosti v tomto případě, oproti klasickému Goertzelovu algoritmu je nutné vyhradit dvě paměťová místa navíc pro reálnou a imaginární část konstanty.

```

Vstupem jsou kmitočety  $k \in \mathbb{R}$ , signál  $x$  délky  $N$ 
Výstupem je  $y$ , což je  $X(\omega_k)$  (DTFT) dle rovnice (20)

%Předpočítání konstant
A = 2*pi*k/N
B = 2*cos(A)
C = e^-j*A
D = e^-j*2*pi*k*(N-1)/N

%Stavové proměnné
s0 = 0
s1 = 0
s2 = 0

%Hlavní cyklus
for i = 0 : N - 2 %o jednu iteraci méně než klasicky
    s0 = x[i] + B * s1 - s2 % (16)
    s2 = s1
    s1 = s0
end

%Dokončující výpočty
s0 = x[N - 1] + B * s1 - s2 % odpovídá (16)
y = s0 - s1 * C
y = y * D % konstanta nahrazující poslední iteraci a
současně korigující fázi
  
```

Obr. 3: Goertzelův algoritmus zobecněný, se zkráceným iteračním cyklem. Barevně jsou vyznačeny změny oproti klasickému Goertzelovu algoritmu z obr. 2.

Je vidět, že zvýšení výpočetní i paměťové náročnosti zobecněného algoritmu je zanedbatelné.

Hlavní výhodou zkrácení podle části 3.2 je především v tom, že není nutné provádět N -tou iteraci a v této době je možné již zahájit zpracování dalšího příchozího vzorku $x[N]$, pokud je algoritmus nasazen v kontinuálním provozu.

4 Software

Dvě funkce pro prostředí Matlab jsou k dispozici na adrese <http://bitbucket.org/signalteam/goertzel>. První funkce `goertzel_classic.m` realizuje klasický Goertzelův algoritmus pro celočíselná k , zobecněný výpočet pro reálná k provádí `goertzel_general_shortened.m`. Struktura programů koresponduje s algoritmy z obr. 2 a 3. Indexování vektorů v MATLABu ovšem začíná jedničkou a nikoliv nulou jako v teoretickém popisu.

5 Závěr

V článku byl prezentován zobecněný Goertzelův algoritmus, který umožňuje použít i neceločíselné násobky základní harmonické, a tím tedy počítat Fourierovu transformaci s diskretním časem (DTFT). Výhodou je tedy zejména, že v aplikacích nejrůznějšího typu, kde se Goertzelův algoritmus využívá, požadované kmitočty není

nutné zaokrouhlovat, a tím obdržet značně přesnější výsledky. V článku je ukázáno, že tohoto zobecnění je dosaženo se zanedbatelným zvýšením výpočetní a paměťové náročnosti.

Poděkování

Tento článek vznikl za podpory výzkumného záměru MSM 0021630513, projektů COST OC08057 a GAČR 102/09/1846.

Literatura

- [1] Davídek, V.: *Implementace algoritmů číslicového zpracování signálů v reálném čase*. Praha: ČVUT, první vydání, 2004, ISBN 80-01-03114-4, 171 s.
- [2] Gay, S. L.; Hartung, J.; Smith, G. L.: Algorithms for Multi-Channel DTMF Detection for the WE DSP32 Family. In *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, Glasgow: IEEE, 1989, ISSN 1520-6149, s. 1134–1137.
- [3] Goertzel, G.: An algorithm for the evaluation of finite trigonometric series. *American Mathematical Monthly*, ročník 65, č. 1, 1958: s. 34–35, ISSN 0002-9890.
- [4] Lyons, R. G.: *Understanding digital signal processing*. New Jersey (USA): Prentice Hall PTR, druhé vydání, 2004, ISBN 0-13-108989-7.
- [5] Wikipedia contributors: Goertzel algorithm. In *Wikipedia: the free encyclopedia*, St. Petersburg (Florida): Wikipedia Foundation, 29. 6. 2005, 19. 1. 2010 [cit. 6. 4. 2010].
URL <http://en.wikipedia.org/wiki/Goertzel>.
- [6] Mock, P.: Add DTMF generation and decoding to DSP-uP designs. In *Digital signal processing applications with the TMS320 family*, ročník 1, New Jersey (USA): Prentice-Hall, 1987, ISBN 0-13-212466-1, s. 543–557.
- [7] Oppenheim, A. V.; Schafer, R. W.; Buck, J. R.: *Discrete-time signal processing*. New Jersey (USA): Prentice-Hall, druhé vydání, 1998, ISBN 0-13-754920-2, 870 s.
- [8] Q.23: *Technical Features of Push-Button Telephone Sets*. ITU-T, Geneva (Switzerland), 1988.