

ÚVOD

V roce 1969 publikoval *J. Vlach* teoretické základy nového algoritmu inverzní Laplaceovy transformace (*ILT*) k časové analýze lineárních obvodů [43]. Postupem času byl tento algoritmus zdokonalován a rozšiřován za účelem simulace obvodů jak se soustředěnými [35],[36],[37], tak i rozloženými parametry [30]. Algoritmus našel své uplatnění v řadě simulačních programů, zejména pro simulaci spínaných obvodů [2],[3],[33]. V současnosti dochází k jeho dalšímu rozpracování v zahraničí i u nás [4],[5],[6],[7],[46],[47].

Metoda *ILT* je zajímavá i z pedagogického hlediska. Objasnění teoretických východisek metody je pěknou ukázkou aplikace teorie komplexní proměnné při řešení technického problému. Kromě toho některé "finesy" této metody se těsně váží k základům teorie dynamických systémů. Umožňují pochopit vztah mezi diferenciální rovnicí, přenosovou funkcí a stavovým popisem, vztahy mezi přirozenou, vynucenou a úplnou odezvou systému a mechanismus přechodu systému do periodického ustáleného stavu.

Metoda je snadno programovatelná, takže umožňuje rychlé ověření na osobním počítači.

Uvedený algoritmus byl rozvíjen i v rámci řešení grantového projektu. Pokusili jsme se o jeho zpřesnění pro delší výpočetní časy [47]. Na principu *ILT* dnes pracují procedury časové analýzy v návrhovém systému NAF [6], programy pro simulaci spínaných obvodů v kmitočtové [11] i časové [47] doméně a programy pro navrhování diskrétních systémů z analogových prototypů metodami signálových invariancí [8],[12],[14]. Algoritmu jsme dále využili k řešení problému numerické inverze transformace *D* [9]. Jiným vyústěním prací na časové analýze je algoritmus výpočtu členů Maclaurinovy řady signálu z jeho operátorového popisu [10].

Předložená zpráva shrnuje výše vyjmenované výsledky naší práce v průběhu prvního roku řešení projektu.

1. PODSTATA ALGORITMU INVERZNÍ LAPLACEOVY TRANSFORMACE A JEHO ZÁKLADNÍCH APLIKACÍ

1.1. Podstata algoritmu

Vzorec pro inverzní Laplaceovu transformaci (ILT)

$$v(t) = \frac{1}{2\pi j} \int_{c-j\infty}^{c+j\infty} V(p) e^{pt} dp \quad (1.1)$$

Lze vyčíslit pomocí reziduové věty, známe-li póly Laplaceova obrazu $V(p)$. Tato podmínka však bývá v mnoha případech omezující. Existuje způsob, jak uskutečnit výpočet bez znalosti těchto pólů.

Funkci e^z aproximujeme tzv. Padéovou racionální lomenou funkcí $R_{N,M}(z)$. Tato funkce má následující vlastnosti:

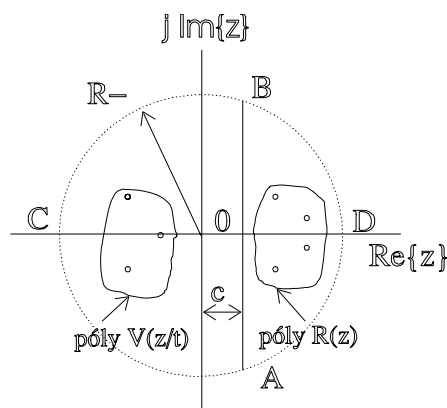
- a) V čitateli je polynom řádu N , ve jmenovateli polynom řádu M .
- b) Prvních $M + N + 1$ členů Taylorovy řady $R_{N,M}(z)$ odpovídá přesně prvním $M + N + 1$ členům Taylorovy řady funkce e^z .
- c) Póly $R_{N,M}(z)$ jsou vesměs jednoduché a všechny leží v otevřené pravé komplexní polorovině.

Způsob výpočtu koeficientů $R_{N,M}(z)$ je uveden např. v [42]. Pro vybraná čísla N, M jsou zde vyčísleny i konkrétní hodnoty těchto koeficientů.

Po náhradě $e^z = R_{N,M}(z)$ lze tedy psát

$$v(t) = \frac{1}{2\pi j t} \int_{c'-j\infty}^{c'+j\infty} V\left(\frac{z}{t}\right) R_{N,M}(z) dz \quad (1.2)$$

Dá se předpokládat, že pro rostoucí $z = pt$ bude aproximace méně přesná a tudíž že pro rostoucí čas bude (1.2) dávat nepřesné výsledky. To je pravda, i když ve skutečnosti chyba neroste neomezeně, nýbrž dochází k její stabilizaci (s výjimkou periodických originálů). Problém nezanedbatelné chyby pro velké časy však lze obejít aplikací tzv. "resetting principle" [42], jak ukážeme dále.



Obr. 1.1. K odvození metody ILT.

Předpokládejme pro jednoduchost, že $V(p)$ je racionální lomená funkce s póly v levé komplexní polorovině. Metoda však v principu není omezena tímto předpokladem. Z obr. 1.1 vyplývá, že výpočet integrálu (1.2) metodou reziduí lze provést dvěma způsoby:

1. Integrací podél křivky ABCA: $\int_{AB} + \int_{BCA} = 2\pi j \sum_{vp-lech V(z/t)} \text{res}\{V(z/t) R_{N,M}(z)\}$
2. Integrací podél křivky ABDA: $\int_{AB} + \int_{BDA} = -2\pi j \sum_{vp-lech R_{N,M}(z)} \text{res}\{V(z/t) R_{N,M}(z)\}$

za předpokladu $\int_{BCA} \rightarrow \int_{BDA} \rightarrow 0$ pro $R \rightarrow \infty$.

Lze dokázat, že poslední předpoklad je splněn, je-li řád jmenovatele racionální lomené funkce v integrandu (1.2) alespoň o dvojku vyšší než řád čitatele [42]. V praxi to znamená, že očekáváme-li kupříkladu, že $V(p)$ může mít řád čitatele stejný jako řád jmenovatele, ale ne vyšší, pak je třeba volit $N \leq M - 2$. Za tohoto předpokladu využijeme druhé možnosti výpočtu a píšeme

$$v(t) = -\frac{1}{t} \sum_{p \text{---} \text{le}ch R_{N,M}(z)} \text{res}\{V(z/t)R_{N,M}(z)\} = -\frac{1}{t} \sum_{i=1}^M K_i V(z_i/t), \quad (1.3)$$

kde z_i jsou jednoduché póly $R_{N,M}(z)$ a K_i jsou příslušná rezidua. Dvojice z_i, K_i jsou pro různé N, M uvedeny např. v [35],[36],[37].

Vzorec (1.3) umožňuje počítat $v(t)$ s výjimkou $t = 0$ pomocí předem vypočtených konstant z_i, K_i , aniž je třeba znát póly obrazu $V(p)$. Pro praktické výpočty je výhodné zvolit M sudé, protože pak ke každému pólu z_i existuje komplexně sdružený pól a ke každému reziduu K_i komplexně sdružené reziduum. Za tohoto předpokladu se výpočty dvakrát urychlí [42].

Přesnost výpočtu $v(t)$ roste se součtem $M + N$. Pro často používanou konfiguraci $M = 10, N = 8$ je metoda ekvivalentní metodě absolutně stabilní integrace 18. řádu.

1.2. Inverzní Laplaceova transformace maticové rovnice

K přednostem algoritmu patří to, že se dá aplikovat přímo na maticovou rovnici. Při simulaci dynamických systémů obvykle vycházíme buď z rovnice modifikované metody uzlových napětí (MMUN)

$$\mathbf{G}\mathbf{v}(t) + \mathbf{C} \frac{d\mathbf{v}(t)}{dt} = \mathbf{D}\mathbf{w}(t) \quad \text{časová oblast} \quad (1.4a)$$

$$(\mathbf{G} + p\mathbf{C})\mathbf{V}(p) = \mathbf{D}\mathbf{W}(p) + \mathbf{C}\mathbf{v}(0) \quad \text{Laplaceova oblast} \quad (1.4b)$$

nebo ze stavových rovnic

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{w}(t) \quad \text{časová oblast} \quad (1.5a)$$

$$(p\mathbf{E} - \mathbf{A})\mathbf{X}(p) = \mathbf{B}\mathbf{W}(p) + \mathbf{x}(0) \quad \text{Laplaceova oblast} \quad (1.5b)$$

Porovnáním (1.4) a (1.5) zjišťujeme jistou podobnost v tom smyslu, že stavová rovnice (1.5) je speciálním případem obecnější rovnice (1.4), kdy matice $\mathbf{G}$ a $\mathbf{C}$ mohou být singulární. Proto se v dalším zaměříme pouze na rovnice (1.4).

Známe-li počáteční podmínku $\mathbf{v}(0)$ a Laplaceův obraz buzení $\mathbf{W}(p)$, určíme časový průběh vektoru $\mathbf{v}(t)$ takto:

- a) Do rovnice (1.4b) postupně dosazujeme $p = z_i/t, i = 1, 2, \dots, M$ a vyřešením určíme vektory $\mathbf{V}(z_i/t)$.
- b) Vektor $\mathbf{v}(t)$ vypočteme z (1.3), kde namísto $V(z_i/t)$ bude $\mathbf{V}(z_i/t)$.

Tento postup vede vždy k cíli i v případě singulárních matic $\mathbf{G}$ a $\mathbf{C}$.

Ani volbou vysokých čísel N, M však nezabráníme tomu, aby přesnost metody klesala s rostoucím časem simulace. Tento jev lze účinně potlačit aplikací tzv. "resetting principle".

1.3. Resetting principle

Podstata spočívá v tom, že časovou osu rozdělíme na úseky takové délky, které zaručují zanedbatelnou chybu výpočtu. Na konci každého úseku se řešení $\mathbf{v}(t)$ stává vektorem počátečních podmínek $\mathbf{v}(0)$ pro další úsek. Tímto způsobem si můžeme dovolit v rámci relativně krátkého úseku použít i méně přesnou aproximaci $R_{N,M}(z)$ a zkrátit tak simulační čas.

Tento princip lze použít i při *ILT* racionálně lomené přenosové funkce tím, že přenosové funkci přiřadíme stavovou rovnici (1.5b) [36],[42].

1.4. Integrace vpřed a vzad (dvoukroková metoda)

Zajímavou aplikací "resetting principle" je tzv. *dvoukroková metoda* [33]. Lze ji použít k přesnému stanovení časového průběhu a jeho charakteristik pro $t = 0$. V prvním kroku se určí vektor signálu pro $t = t_{krok}$, což je zároveň vektor počátečních podmínek pro druhý krok. Nyní se aplikuje přesně stejný krok "zpět proti času", tj. $t = -t_{krok}$. Po tomto kroku se dostaneme do počátku souřadnic, tj. získáme $\mathbf{v}(0)$. Takto lze rovněž identifikovat mohutnosti případných Diracových impulsů v počátku, pokud aplikujeme dvoukrokovou metodu na integrál signálu.

1.5. Operace se signály

Pomocí metody *ILT* lze získat z Laplaceova obrazu nejen časový průběh odpovídajícího signálu, ale i průběhy signálů z něj odvozených různými operacemi. Za zmínku stojí například položka 3 v Tab. 1.1, která ukazuje, že z Laplaceova obrazu signálu lze získat "průběhový" Laplaceův obraz, tj. jeho postupný vývoj pro rostoucí čas. Toho je možné využít při počítačové simulaci reálných spínaných obvodů současně v rovinách p i z [13].

	signál	Laplaceův obraz	algoritmus výpočtu <i>ILT</i>
1	$\frac{dv(t)}{dt}$	$pV(p) - v(0)$	$-\frac{1}{t} \sum_{i=1}^M K_i \frac{z_i}{t} V\left(\frac{z_i}{t}\right)$
2	$\int_0^t v(\alpha) d\alpha$	$V(p)/p$	$-\frac{1}{t} \sum_{i=1}^M K_i \frac{t}{z_i} V\left(\frac{z_i}{t}\right)$
3	$\int_0^t v(\alpha) e^{-s\alpha} d\alpha$	$V(p+s)/p$	$-\frac{1}{t} \sum_{i=1}^M K_i V\left(\frac{z_i}{t} + s\right) / \left(\frac{z_i}{t}\right)$
4	$v(t) * w(t)$	$V(p)W(p)$	$-\frac{1}{t} \sum_{i=1}^M K_i V\left(\frac{z_i}{t}\right) W\left(\frac{z_i}{t}\right)$

Tab. 1.1. Některé operace se signály a způsoby jejich výpočtů numerickou inverzí Laplaceova obrazu.

1.6. Další využití

Velmi slibné se jeví využití *ILT* k diskretizaci spojitých modelů lineárních systémů, tj. k převodu diferenciálních rovnic na diferenční rovnice podle určitého kritéria [5],[8],[12],[14]. To má svůj význam, chceme-li využít *CAD* systémů pro návrh analogových regulátorů nebo filtrů k návrhu obvodů diskrétních a číslicových. S otázkou diskretizace souvisí široké možnosti využití algoritmu *ILT* v simulačních programech pro spínané obvody [11],[13],[15].

2. NUMERICKÁ INVERZNÍ LAPLACEOVA TRANSFORMACE S VYUŽITÍM ČEBYŠEVOVY APROXIMACE

2.1. Úvod

Padeova aproximace, použitá ve Vlachově metodě *ILT* k aproximaci exponenciální funkce ve vzorci pro inverzní transformaci (1.1), vyniká vysokou přesností aproximace v okolí počátku souřadného systému. S rostoucím časem výpočtu (nezávisle proměnnou je v tomto případě čas) však chyba aproximace progresivně roste. Naším cílem bylo nalezení metody aproximace, jejíž průběh chyby by umožňoval jednorázový výpočet pro delší čas.

Podle pracovní hypotézy by touto metodou mohla být Čebyševova aproximace [34], [46], [47].

2.2. Teoretické základy

Uvažujme upravený tvar vzorce pro *ILT* (1.2). Nyní podrobíme funkci e^z aproximaci pomocí racionálně lomené funkce. Pro Čebyševovu aproximaci můžeme psát

$$T_{m,k}(z) = \frac{P_m(z)}{Q_m(z)} = \frac{\sum_{j=0}^m a_j T_j(z)}{\sum_{j=0}^k b_j T_j(z)}, \quad (2.1)$$

kde $P_m(z)$ a $Q_m(z)$ jsou mnohočleny stupňů m a k .

Nyní je nezbytné určit koeficienty a_j a b_j .

Čebyševova aproximace je založena na rozvoji funkce $f(z)$ v řadu [34]

$$f(z) = \frac{1}{z} c_0 + \sum_{j=0}^{\infty} c_j T_j(z). \quad (2.2)$$

Pro koeficienty c_j můžeme psát (pouze pro aproximaci funkce e^z)

$$c_j = 2I_j(1), \quad (2.3)$$

kde $I_j(1)$ je modifikovaná Besselova funkce prvního druhu.

Lze dokázat, že pro koeficienty a_j a b_j můžeme sestavit soustavu lineárních algebraických rovnic

$$\begin{aligned} a_0 &= \frac{1}{2} \sum_{i=0}^k b_i c_i, \\ a_j &= \frac{1}{2} \sum_{i=0}^k b_i (c_{|j-i|} + c_{j+i}), \quad j = 1, \dots, N, \quad N = m + k, \\ a_j &= 0, \quad j > m. \end{aligned} \quad (2.4)$$

Tato soustava obsahuje $N + 1$ rovnic pro $N + 2$ neznámých, avšak lze zvolit $b_0 = 1$.

V Tab. 2.1 je ukázáno srovnání čítelů a jmenovatelů funkcí $T_{8,10}$ a $P_{8,10}$. Označení T se vztahuje na Čebyševovu aproximaci a P na Padeovu.

2.3. Numerické výsledky

Analýzou dat, které byly získány naprogramováním obou metod, bylo zjištěno, že aproximace exponenciály pomocí Čebyševovy metody je přesnější v porovnání s Padeovou metodou.

Po provedené aproximaci byly vypočteny póly a rezidua aproximačních funkcí a tyto funkce byly použity v algoritmu pro inverzní Laplaceovu transformaci, viz. např. [46], [47].

V Tab. 2.2 jsou porovnány póly a rezidua funkcí $T_{8,10}$ a $P_{8,10}$. Jsou ukázány pouze póly s kladnou

imaginární částí.

	Čitatele		Jmenovatele	
$T_{8,10}$	a_0	1.000000000000000000	b_0	1.000000000000000000
	a_1	0.444504231872440694	b_1	-0.555495768127559306
	a_2	0.091525919744705844	b_2	0.147021687872265150
	a_3	0.011441637375245691	b_3	-0.024498833099906473
	a_4	0.000953492832729215	b_4	0.002857443351096331
	a_5	0.000054482023110909	b_5	-0.000244835751428073
	a_6	0.000002095060660729	b_6	0.000015686837354772
	a_7	0.000000049862508655	b_7	-0.000000746475656442
	a_8	0.000000000566205573	b_8	0.000000025422924138
			b_9	-0.000000000564137588
$P_{8,10}$			b_{10}	0.000000000006254482
	a_0	1.000000000000000000	b_0	1.000000000000000000
	a_1	0.444444444444444444	b_1	-0.555555555555555556
	a_2	0.091503267973856209	b_2	0.147058823529511765
	a_3	0.011437908496732026	b_3	-0.024509803921568628
	a_4	0.000953159040394336	b_4	0.002859477124183007
	a_5	0.000054466223936819	b_5	-0.000245098039215686
	a_6	0.000002094855033632	b_6	0.000015711412770236
	a_7	0.000000049877500858	b_7	-0.000000748162512868
	a_8	0.000000000566789782	b_8	0.000000025505540211
$P_{8,10}$			b_9	-0.000000000566789782
			b_{10}	0.000000000006297664

Tab. 2.1. Srovnání čitateľů a jmenovatelů aproximačních funkcí Padéovy a Čebyševovy aproximace.

	Čebyšev	Pade
Póly	<i>Re</i> 1.18566855640266 E1	1.18300937391681 E1
	<i>Im</i> 1.59095994437192 E0	1.59375300588581 E0
	1.12455352814926 E1	1.12208537793951 E1
	4.78492420888555 E0	4.79296416756566 E0
	9.95444493277310 E0	9.93333383722175 E0
	8.02081170985597 E0	8.03310633426629 E0
	7.79724589072478 E0	7.78114626446461 E0
	1.13539262950610 E1	1.13688916490499 E1
	4.24475714793720 E0	4.23452249479700 E0
	1.49415752683083 E1	1.49570437812815 E1
Rezidua	1.63651281870040 E4	1.62866236805047 E4
	-1.42619701824349 E5	-1.39074711551605 E5
	-2.83283598819873 E4	-2.81781117130516 E4
	7.63224797152626 E4	7.43575823727417 E4
	1.47240212677406 E4	1.46297402523314 E4
	-1.97439509105781 E4	-1.91818081855018 E4
	-2.89473557752222 E3	-2.87041816103207 E3
	1.74202322094386 E3	1.67410948408430 E3
	1.33892004764833 E2	1.32165941247487 E2
	1.56753996997135 E1	1.74767479887716 E1

Tab. 2.2. Srovnání pólů a reziduí aproximačních funkcí Padéovy a Čebyševovy aproximace.

2.4. Shrnutí

Fakt, že prostá aproximace funkce e^z pomocí Čebyševovy metody je přesnější, dával dobré předpoklady pro použití této metody v algoritmu pro inverzní Laplaceovu transformaci. Analýzou simulačních výsledků bylo však zjištěno, že uvedený algoritmus je citlivější na chyby aproximace právě v okolí počátku souřadného systému. Vyšší přesnost aproximace Čebyševovy metody pro větší hodnoty nezávisle proměnné (času) se tedy nemohla uplatnit. Navíc aproximační chyba této metody osciluje, viz. např. [34], a nebylo tudíž možné nalézt pravidlo, které by jednoznačně předurčilo oblasti použití Čebyševovy aproximace.

V současné době jsme se tedy vrátili ke klasické Padéově aproximaci. Pokles její přesnosti při rostoucím čase simulace eliminujeme aplikací „resetting principle“.

3. ČASOVÁ ANALÝZA LINEÁRNÍCH SYSTÉMŮ

3.1. Úvod

V souvislosti s vývojem návrhového systému *NAF* [16],[21] bylo třeba vyřešit následující problém:

Jsou známy koeficienty přenosové funkce analogového filtru jako výsledek návrhového procesu. Je třeba simulovat vynucenou odezvu [28] filtru jak na typizované budicí signály (Diracův impuls, skoková funkce, lineárně narůstající impuls apod.), tak i na signály získané vzorkováním skutečných uživatelských signálů. Dále je požadován výpočet jedné periody odezvy v ustáleném stavu, chápeme-li segment budicího signálu jako jednu periodu jeho periodického opakování.

Pro účely analýzy a optimalizace aktivních filtrů pak bylo třeba zabezpečit výše uvedené činnosti za předpokladu, že filtr je popsán soustavou rovnic modifikované metody uzlových napětí (*MMUN*).

Vzhledem k tomu, že uvedené algoritmy slouží k interaktivní uživatelské analýze navrhovaných filtrů a k syntéze v optimalizačních smyčkách, jsou kladeny velké nároky na jejich rychlost.

Uvedené práce vyústily v tvorbu programových jednotek časové analýzy odladěných v Borland Pascalu 7. Jednotky slouží k rychlé časové analýze filtrů z jejich přenosových funkcí. Jednotky zabezpečují následující činnosti:

- 1) Výpočet impulsních a přechodných charakteristik včetně identifikace Diracových impulsů.
- 2) Výpočet vynucené odezvy filtru na obdélníkový a pilovitý signál.
- 3) Výpočet vynucené odezvy filtru na „uživatelský“ signál jehož vzorky jsou zapsány v datovém souboru.
- 4) Přímý výpočet periodického ustáleného stavu za předpokladu, že daný segment budicího signálu představuje jednu periodu jeho periodického pokračování.

Z literatury je známa řada rozdílných přístupů k řešení úlohy numerické inverze Laplaceova obrazu ve tvaru racionální lomené funkce. Uvedme například původní *Zakianovo* řešení [45], *Valsovu* metodu [41], metodu náhrady Laplaceova obrazu z -obrazem [18],[32], semisymbolickou analýzu časových odezev rozkladem Laplaceova obrazu na parciální zlomky [24],[44] apod. Posledně jmenovaná metoda sice poskytuje přímo analytické výrazy pro časovou odezvu, její přesnost však silně závisí na přesnosti výpočtu všech pólů Laplaceova obrazu. V případě vícenásobných pólů nebo pólů blízkých hodnot může metoda selhat. Ostatní jmenované metody nejsou založeny na výpočtu pólů, avšak jinou nepříjemnou vlastností je pokles přesnosti stanovení průběhu s narůstajícím časem, což vede na neúnosné simulační chyby v případě pomalu zanikajících nebo dokonce periodických odezev. Další nevýhodou je problematický výpočet periodického ustáleného stavu.

K realizaci výše uvedených činností se proto jako nejvýhodnější jevil Vlachův algoritmus numerické Laplaceovy transformace maticové stavové rovnice, která je vytvořena nad známou přenosovou funkcí systému tak, že dané přenosové funkci se přidruží dynamický systém v druhé kanonické struktuře [36]. To nám umožnilo i bezproblémový výpočet periodického ustáleného stavu. K zabezpečení dostatečné přesnosti analýzy stavu byl algoritmus doplněn „resetting principle“.

3.2. Stavový model přidružený k přenosové funkci

Uvažujme přenosovou funkci $K(p)$ lineárního dynamického systému n -tého řádu s jedním vstupem $w(t)$ a jedním výstupem $y(t)$:

$$K(p) = \frac{L\{v(t)\}}{L\{w(t)\}} = \frac{Y(p)}{W(p)} = \frac{\sum_{i=0}^m a_i p^i}{p^n + \sum_{i=1}^{n-1} b_i p^i}, m \leq n, n \geq 1, \quad (3.1)$$

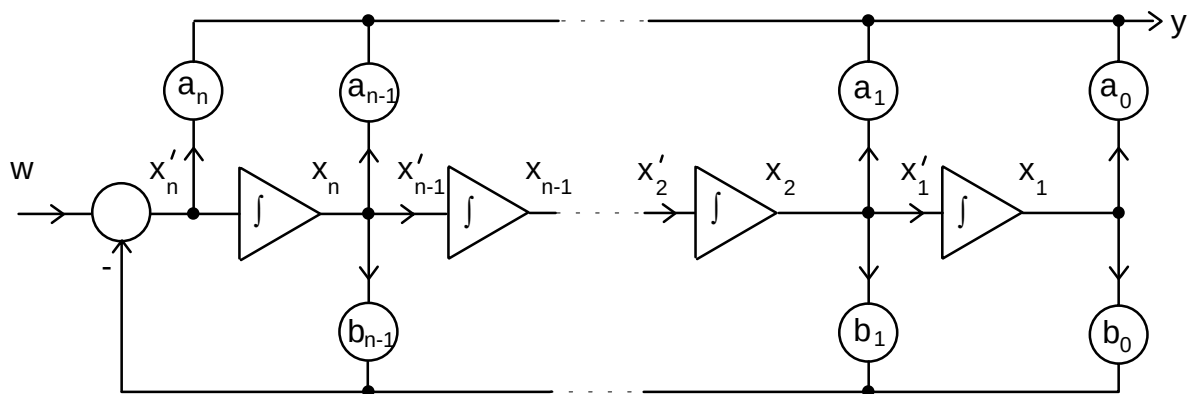
kde $L\{\}$ je operátor Laplaceovy transformace. Téhož přenosové funkci odpovídá nekonečně mnoho stavových modelů. Jeden z nich je znázorněn ve formě schématu na obr. 3.1. Jedná se o dynamický systém v 2. *kanonické struktuře* [28]. Zvolíme-li za stavové proměnné výstupy integrátorů, stavové rovnice budou ve tvaru

$$\frac{d}{dt}\mathbf{x} = \mathbf{Ax} + \mathbf{B}w, \quad (3.2)$$

$$y = \mathbf{C}\mathbf{x} + a_n w, \quad (3.3)$$

kde stavová matice $\mathbf{A}$ a vektory $\mathbf{B}$ a $\mathbf{C}$ mají následující strukturu:

$$\mathbf{A} = \begin{vmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ -b_0 & -b_1 & -b_2 & \dots & -b_{n-1} \end{vmatrix}, \mathbf{B} = \begin{vmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{vmatrix}, \mathbf{C} = \begin{vmatrix} a_0 - a_n b_0 & a_1 - a_n b_1 & \dots & a_{n-1} - a_n b_{n-1} \end{vmatrix}. \quad (3.4)$$



Obr. 3.1. 2. kanonická struktura dynamického systému n -tého řádu.

Stavová rovnice dynamiky (3.2) má v operátorové oblasti tvar

$$(p\mathbf{E} - \mathbf{A})\mathbf{X}(p) = \mathbf{B}W(p) + \mathbf{x}(0), \quad (3.5)$$

kde $\mathbf{X}(p)$ a $W(p)$ jsou Laplaceovy obrazy stavového vektoru $\mathbf{x}(t)$ ($n \times 1$) a vstupního signálu $w(t)$; $\mathbf{x}(0)$ je vektor počátečních podmínek v čase $t = 0$ a $\mathbf{E}$ je jednotková matice ($n \times n$).

Výstupní rovnice (3.3) se dá s přihlédnutím k struktuře vektoru $\mathbf{C}$ vyjádřit takto:

$$y(t) = \sum_{i=0}^{n-1} (a_i - a_n b_i) x_{i+1}(t) + a_n w(t), \quad (3.6)$$

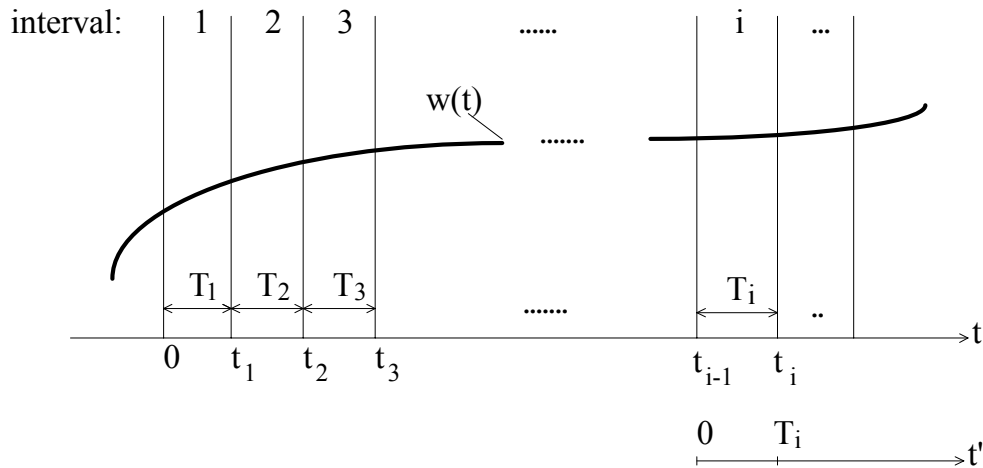
kde $x_i, i=1,2,...,n$ jsou prvky stavového vektoru.

Tímto způsobem jsme problém časové analýzy filtru z jeho přenosové funkce převedli na problém numerické inverzní Laplaceovy transformace maticové rovnice (3.5). Výsledkem této inverze bude časový vektor $\mathbf{x}(t)$. Výstupní signál pak počítáme z rovnice (3.6) jako lineární kombinaci stavových veličin a vstupního signálu.

3.3. Analýza vynucené odezvy

Tzv. "resetting principle" [36] je objasněn na obr. 3.2. Rozdělme časovou osu na dílčí intervaly podle obrázku. Nad každým i -tým intervalem aplikujeme algoritmus inverzní Laplaceovy transformace a tím získáme vektor $\mathbf{x}(t_i)$ z rovnice (1.3). Tuto rovnici lze přepsat pro i -tý interval následovně:

$$(p_i \mathbf{E} - \mathbf{A})\mathbf{X}^i = \mathbf{B}W^i(p_i) + \mathbf{x}(t_{i-1}). \quad (3.7)$$



Obr. 3.2. K objasnění „resetting principle“.

Na pravé straně této rovnice je stavový vektor $\mathbf{x}(t_{i-1})$, počítaný v posledním kroku. W je Laplaceův obraz vstupního signálu za předpokladu, že počátek časové osy umístíme na začátek řešeného časového intervalu. Odpovídající lineární transformace časové osy $t' = t - t_{i-1}$ je znázorněna na obr. 3.2.

Rovnice (3.7) musí být v souladu s [42] opakovaně řešena pro několik předem vypočtených komplexních konstant p_i . V nejjednodušším případě se jedná o jedinou konstantu

$$p_i = (1 + j)/T_i$$

přiřazenou i -tému intervalu [4],[6].

V [4] jsme popsali účinnou metodu výpočtu vektoru $\mathbf{X}^i$. Tato metoda je založena na LU rozkladu matice $p_i \mathbf{E} - \mathbf{A}$. V případě konstantního kroku $T_i = T$ je LU rozklad proveden jen jednou. Řešení rovnice (3.7) pak probíhá velmi ekonomickým způsobem pro různé pravé strany. Jakmile je nalezen vektor $\mathbf{X}^i$, určí se časová odezva inverzní Laplaceovou transformací

$$\mathbf{x}(t_i) = -2 \operatorname{Im}\{\mathbf{X}^i\} / T_i. \quad (3.8)$$

Výstupní signál $y(t_i)$ v čase $t' = t_i$ je pak počítán z výstupní rovnice (3.6).

Výše uvedeným způsobem lze získat vzorky výstupního signálu v okamžicích $t_1, t_2, t_3, \dots$. Časové kroky T_i musí být voleny dostatečně krátké, aby se neprojevovaly chyby jednoduchého algoritmu ILT (3.8).

3.4. Výpočet počáteční hodnoty odezvy a identifikace Diracova impulsu

Často je zapotřebí určit přesnou počáteční hodnotu odezvy $y(0_+)$. Analyticky může být tato hodnota zjištěna z limitní věty. Tento přístup může být snadno implementován pro strojový výpočet při inverzi obrazu ve tvaru racionální lomené funkce. Počáteční hodnota a mohutnost případného Diracova impulsu v počátku mohou být určeny jednoduchým výpočtem.

Limitní přístup však není vhodný pro počítačovou analýzu v případě ILT maticové rovnice. K výpočtu počáteční hodnoty lze použít *dvoukrokovou metodu* (viz část 1.4). V prvním kroku volíme dopředný integrační krok T_1 a vypočteme vektor $\mathbf{x}(T_1)$. Pak zvolíme integrační krok stejné délky zpět a získáme počáteční vektor $\mathbf{x}(0_+)$.

Pomocí dvoukrokové metody lze určit i mohutnost případného Diracova impulsu pro $t=0$. Podrobnosti jsou uvedeny v [33].

3.5. Odezva na uživatelský signál zadaný vzorky

Často jsou namísto analogového signálu $w(t)$ k dispozici pouze jeho vzorky, získané měřením, vzorkováním nebo jiným způsobem. Pak je zapotřebí provést interpolaci a získat tak analogový signál jako vstupní signál testovaného analogového systému. V Tab. 3.1 jsou uvedeny čtyři typy interpolací, které se ukázaly jako vhodné pro využití algoritmem *ILT*. Poznamenejme, že uvedená „impulsní interpolace“ není interpolací v klasickém slova smyslu, protože není splněna rovnost $w^{int}(t_i) = w(t_i)$.

Volba typu interpolace bude záviset jak na požadované přesnosti časové analýzy, tak na skutečném, resp. předpokládaném charakteru vstupního signálu, jehož vzorky jsou k dispozici. Rovněž je vhodné vzít v úvahu přenosové vlastnosti testovaného obvodu. Impulsní a skoková interpolace poskytují dobré výsledky, je-li splněna podmínka vzorkovacího teorému s dostatečnou rezervou. Systém však nesmí mít charakter horní propusti, protože rychlé změny, které jsou obsaženy ve vstupním signálu, nesmí být přenášeny do počítané odezvy. Výpočet Laplaceova obrazu $W^i(p)$ je pak snadný.

Impulsní (skoková) interpolace je dále výhodná, jestliže původní vstupní signál má charakter úzkých impulsů (impulsů typu *Sample-Hold*).

Lineární interpolace poskytuje velmi dobré výsledky pro různé typy obvodů.

Interpolace kubickými spliny je určena pro přesnou analýzu za předpokladu že původní vstupní signál je dostatečně „hladký“, avšak je vzorkován s relativně nízkým vzorkovacím kmitočtem blízkým *Nyquistovu* kmitočtu. Numerická inverze Laplaceova obrazu je v případě této interpolace časově nejnáročnější (viz též Tab. 3.1).

V programové jednotce vytvořené v Pascalu je přednastavena skoková interpolace jako kompromis mezi rychlostí a přesností. Uživatel programu *NAF* si však bude moci vybrat i jiný typ interpolace podle svých požadavků.

interpolace	časová oblast: $w^{int}(t')$	operátorová oblast: $W^i(p)$ $W^i(p_i), p_i = (1+j)/T_i$
impulsní	$w(t_{i-1})T_i\delta(t')$	$w(t_{i-1})T_i$ $w(t_{i-1})T_i$
skoková	$w(t_{i-1})$	$w(t_{i-1})/p$ $w(t_{i-1})T_i(1-j)/2$
lineární	$w(t_{i-1}) + [w(t_i) - w(t_{i-1})]t'/T_i$	$w(t_{i-1})/p + [w(t_i) - w(t_{i-1})]T_i/2$ $[w(t_{i-1}) - jw(t_i)]T_i/2$
kubickým splinem	$a_0^i + a_1^i t' + a_2^i t'^2 + a_3^i t'^3$	$a_0^i/p + a_1^i/p^2 + 2a_2^i/p^3 + 6a_3^i/p^4$ $\{a_0^i - a_2^i T_i^2 - 3a_3^i T_i^3 - j[a_0^i + a_1^i T_i + a_2^i T_i^2]\}T_i/2$
Poznámka: vzorkovaný signál $w(t_i)$, $i = 0, 1, 2, \dots$ krok č. i : $t' = t - t_{i-1}$		

Tab. 3.1. Různé typy interpolací a jejich začlenění do algoritmu ILT.

3.6. Výpočet periodického ustáleného stavu

Jednoduchá metoda hledání periodického ustáleného stavu je popsána v [32]. Některá vylepšení této metody byla uvedena v [18]. Metoda je založena na stavovém přístupu. Po náhradě derivací diferencemi je maticová diferenciální stavová rovnice převedena na soustavu algebraických rovnic s jednoduchým algoritmickým řešením.

Tato metoda tedy využívá jiných principů než je inverze Laplaceova obrazu. Abychom využili algoritmu *ILT* i k hledání periodického ustáleného stavu, použili jsme zcela jiný postup.

V případě periodického ustáleného stavu s opakovací periodou T_p platí rovnost vektorů $\mathbf{x}(0)$ a $\mathbf{x}(T_p)$:

$$\mathbf{x}(0) = \mathbf{x}(T_p) \quad (3.9)$$

Důležité je nalezení těchto vektorů. Výpočet periodického ustáleného stavu je pak snadnou záležitostí.

Je známo, že řešení rovnice (3.2) v časové oblasti sestává z vynucené odezvy $\mathbf{x}_v(t)$ při nulovém výchozím stavu a z přirozené odezvy $\mathbf{x}_p(t)$ při nulovém buzení. Pro čas $t = T_p$ tedy platí

$$\mathbf{x}(T_p) = \mathbf{x}_v(T_p) + \mathbf{x}_p(T_p) \quad (3.10)$$

Vektor přirozené odezvy je přímo úměrný vektoru počátečního stavu $\mathbf{x}(0)$:

$$\mathbf{x}_p(T_p) = \mathbf{g}^*(T_p) \mathbf{x}(0) \quad (3.11)$$

Pro stavový systém (3.2) je matice $\mathbf{g}^*(T_p)$ dána maticovou rovnicí

$$\mathbf{g}^*(T_p) = e^{\mathbf{A}T_p} \quad (3.12)$$

Úpravou rovnic (3.9), (3.10) a (3.11) dostáváme výsledek

$$\mathbf{x}(0) = [\mathbf{E} - \mathbf{g}^*(T_p)]^{-1} \mathbf{x}_v(T_p) \quad (3.13)$$

Vektor $\mathbf{x}_v(T_p)$ lze získat inverzní Laplaceovou transformací rovnice (3.2) přes periodu T_p při nulovém výchozím stavu. Matici $\mathbf{g}^*(T_p)$ získáme opět pomocí *ILT* rovnice (3.2) pro $W(p) = 0$. K výpočtu k -tého sloupce $\mathbf{g}^*(T_p)$ je zapotřebí položit $x_k(0) = 1$, $x_i(0) = 0$, $i \neq k$.

Pro výpočet matice $\mathbf{g}^*(T_p)$ se osvědčil rozvoj exponenciální funkce v (3.12) do Taylorovy řady a sečtení N jejích členů:

$$\mathbf{E} - e^{\mathbf{A}T_p} \approx - \sum_{k=1}^N \frac{(\mathbf{A}T_p)^k}{k!} \quad (3.14)$$

Počet členů N je volen tak, aby byl výpočet zatížen akceptovatelnou chybou ε . V průběhu výpočtu je zadaná hodnota ε konfrontována s výrazem

$$\varepsilon = \left\| \frac{(\mathbf{A}T_p)^N}{N!} \right\| \left\| \sum_{k=1}^N \frac{(\mathbf{A}T_p)^k}{k!} \right\|.$$

Symbolem $\| \cdot \|$ je označena norma matice.

Jakmile je proveden výpočet vektoru $\mathbf{x}(0)$, periodická ustálená odezva je získána algoritmem *ILT* z rovnice (3.2).

3.7. Časová analýza obvodu založená na modifikované metodě uzlových napětí

K uskutečnění výše popsané metody časové analýzy je nutné znát koeficienty přenosové funkce systému. V mnoha případech je však namísto této funkce k dispozici soustava rovnic, popisující daný obvod v rámci modifikované metody uzlových napětí (*MMUN*) [42]. Ukážeme, že i v tomto případě můžeme k časové analýze využít prezentované metody *ILT*.

MMUN vede na maticovou rovnici ve tvaru

$$(\mathbf{G} + p\mathbf{C})\mathbf{V}(p) = \mathbf{D}w(p) + \mathbf{C}\mathbf{v}(0), \quad (3.15)$$

kde $\mathbf{V}$ je vektor uzlových napětí a případných branových proudů, w je vstupní signál a $\mathbf{v}(0)$ je vektor počátečních podmínek. $\mathbf{G} + p\mathbf{C}$ je modifikovaná admitanční matice. $\mathbf{D}$ je incidenční vektor. Algoritmus počítačového sestavení těchto rovnic je všeobecně znám.

Po srovnání rovnic (3.15) a (3.5) zjišťujeme, že rovnice (3.15) představuje zobecněný stavový model lineárního systému. Aplikací algoritmu *ILT* můžeme stejně jako u stavových rovnic použít stejnou proceduru pro časovou analýzu. V případě výpočtu periodického ustáleného stavu však nelze k výpočtu matice $\mathbf{g}^*(T_p)$ použít rozvoj (3.14). Tuto matici nyní určíme pomocí algoritmu *ILT*.

3.8. Shrnutí

V této kapitole jsou analyzovány možnosti řešení lineárních obvodů v časové oblasti metodou inverzní Laplaceovy transformace s cílem jejich využití v systému *NAF* pro navrhování analogových filtrů. Je vyřešen problém analýzy vynucené odezvy i periodického ustáleného stavu pro systém zadaný jak přenosovou funkcí, tak i soustavou rovnic *MMUN*. Podrobně je řešena otázka výpočtu odezvy na uživatelský signál reprezentovaný vzorky dat.

Na základě popsaných principů byly vytvořeny a odladěny procedury časové analýzy v Borland Pascalu. Do současné verze programu *NAF* byla začleněna jednotka pro přesnou analýzu přechodných charakteristik navrhovaných filtrů. Do připravované profesionální verze budou zakomponovány i ostatní algoritmy.

Určitou nevýhodou použitého algoritmu *ILT* je nutnost práce s komplexní aritmetikou, paměťově náročné výpočty v případě nutnosti vysoce přesné simulace a nepříznivé vazby mezi přesností a rychlostí simulace. V následující etapě řešení projektu se proto pokusíme o nalezení řešení, které by dané nedostatky odstranilo.