

Evoluční návrh filtru

Tomáš Müller

Katedra elektromagnetického pole, FEL ČVUT
Technická 2, 166 27 Praha 6
muller@kti.mff.cuni.cz

Abstract

Genetic and evolutionary algorithms are powerful general purpose optimization tools which model the principles of evolution. They are often capable of finding globally optimal solution even in the most complex search spaces. In this paper we outline an evolutionary (single population) algorithm for designing microwave filters from sequential arranged segments of coaxial wiring. We also present some results achieved on an example low-pass filter.

Úvod

Genetické a evoluční algoritmy jsou jedny ze základních optimalizačních metod založených na Darwinově teorii evoluce. Jako jejich hlavní výhoda se často označuje schopnost produkovat dobré, kvalitní řešení, a to i v případě značné rozlehlosti vstupních problémů. To je také důvodem, proč tyto algoritmy nacházejí uplatnění v nejrůznějších optimalizačních problémech.

Tento příspěvek se zabývá použitím evolučních algoritmů pro návrh jednoduchých mikrovlnných obvodů, které jsou složeny ze sekvenčně řazených obvodových prvků. Úkolem hledaného algoritmu je tedy zvolit počet obvodových prvků, jejich typ a parametry tak, aby byly splněny požadavky kladené na přenosovou funkci daného obvodu. Použitý algoritmus je demonstrován při návrhu mikrovlnného filtru, složeného z úseků koaxiálního vedení.

Genetické a evoluční algoritmy

Základní myšlenka všech genetických algoritmů vychází z vývoje populace. Populaci většinou představuje množina prvků, jakýchsi kandidátů na řešení. V našem případě těmito prvky mohou být například jednotlivé navrhované obvody. Algoritmus se snaží pomocí kombinace křížení a mutace jednotlivých členů populace postupně nalézt lepší a lepší řešení. [3, 4, 5]

Evoluční algoritmy můžeme chápat jako extrémní případ genetických algoritmů, kde velikost populace je 1. Vstupem je zde vždy jeden prvek, ze kterého je vytvořen jeden potomek. Tento potomek může být dále mutován, například použitím nějaké optimalizační metody. Na konci každé fáze je vybrán jeden z prvků do následné populace – původní prvek (rodič) či prvek z něj vytvořený (potomek). Potomek je z rodiče vytvářen změnou jednoho či více parametrů. V případě nediskrétních parametrů je pro výpočet změny hodnoty většinou použito normální rozdělení s nulovou střední hodnotou. Variance normálního rozdělení se může v průběhu výpočtu měnit, například exponenciálně klesat. [1, 3, 5]

Objektivní funkce – chyba nalezeného řešení

Vstupem při návrhu obvodu (například filtru) je většinou požadovaná útlumová charakteristika. Naším úkolem je nalézt co nejmenší počet za sebou zapojených dvojbranů tak, aby byly přenosové požadavky splněny. Dále se zaměříme pouze na úseky koaxiálního vedení, ztráty ve vedení a odrazy na rozhraní dvou vedení budeme zanedbávat.

Při výpočtu útlumové charakteristiky daného obvodu vyjdeme z vlnové kaskádní T-matice dvojbranu tvořeného koaxiálním vedením ve tvaru:

$$T_k = \begin{pmatrix} Ch - \frac{Z_v^2 + Z_0^2}{2Z_v Z_0} Sh & \frac{Z_v^2 - Z_0^2}{2Z_v Z_0} Sh \\ -\frac{Z_v^2 - Z_0^2}{2Z_v Z_0} Sh & Ch + \frac{Z_v^2 + Z_0^2}{2Z_v Z_0} Sh \end{pmatrix}, \text{ kde } \begin{matrix} Sh = \sinh(\gamma d) = \sin(\theta)j \\ Ch = \cosh(\gamma d) = \cos(\theta) \end{matrix}$$

kde Z_0 je impedance vstupu a výstupu dvojbranu, d je délka vedení,

$Z_v = \sqrt{\frac{l}{c}} = \frac{1}{2\pi} \sqrt{\frac{\mu}{\epsilon}} \ln\left(\frac{b}{a}\right)$ je vlnová impedance koaxiálního bezeztrátového vedení,

a a b jsou poloměry vnitřního a vnějšího vodiče koaxiálního vedení,

$\lambda = \frac{c_0}{f\sqrt{\epsilon_r}}$ je délka vlny ve vedení, c_0 je rychlost světla ve vakuu,

$\theta j = \gamma d = 2\pi \frac{d}{\lambda} j = 2\pi f d \frac{\sqrt{\epsilon_r}}{c_0} j$ je elektrická délka vedení.

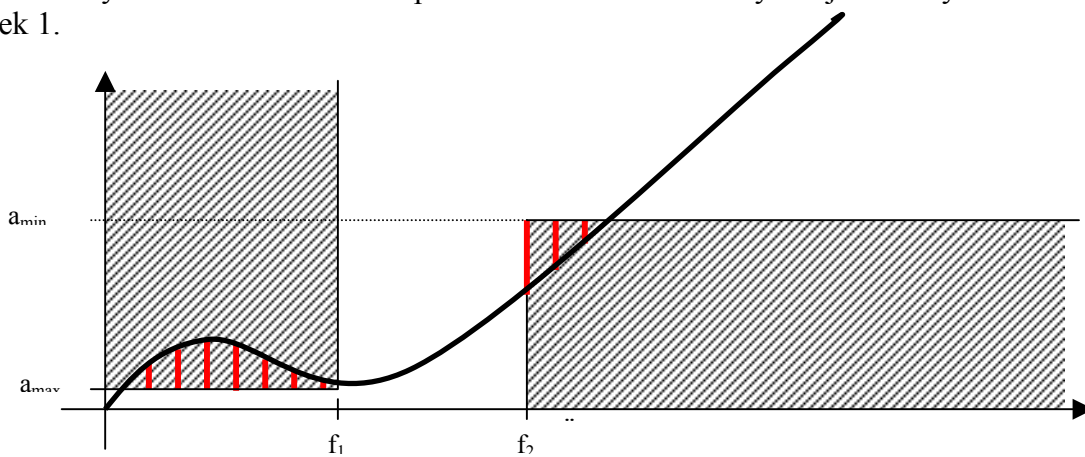
Vlnová kaskádní matice celého filtru pak bude součinem kaskádních matic jednotlivých vedení:

$$T = T_{k1} \cdot T_{k2} \cdots$$

Výsledný přenos je pak dán vztahem $G = S_{2,1} = \frac{1}{T_{2,2}}$

a požadovaný útlum filtru na dané frekvenci f lze spočítat dle vztahu $a(f) = 20 \log\left(|T(f)_{2,2}|\right)$

Abychom mohli porovnat dvě různá řešení, musíme si zvolit nějakou objektivní funkci, která nám bude říkat, jak moc se liší nalezené řešení (tj. spočtená útlumová funkce) od požadovaného řešení. To můžeme udělat například tak, že sečteme střední kvadratické odchylky útlumové charakteristiky navrženého filtru od požadované charakteristiky na jednotlivých frekvencích, viz obrázek 1.



Obr 1. Příklad výpočtu chyby pro navržené řešení DP filtru $(f_1, a_{\max}, f_2, a_{\min})$

Objektivní funkce je počítána na 100 různých frekvencích v rozsahu f_{\min} a f_{\max} , tento interval frekvencí je rozdělen logaritmičticky. Pro docílení přesnějších výsledků je dobré ještě přidat výpočet chyby na zlomových frekvencích f_1 a f_2 . Z předchozího odstavce je patrné, že se budeme snažit zvolenou chybovou funkci minimalizovat. Výsledné řešení by mělo mít tuto funkci nulovou.

Algoritmus

Evoluční návrh obvodu bude vycházet z jednoho náhodně zvoleného koaxiálního vedení. Abychom zamezili nesmyslným výsledkům, jsou jednotlivé parametry koaxiálního vedení omezeny vztahy: [6, 7]

délka vedení	$1\text{ mm} \leq d \leq 1\text{ m}$
vnitřní poloměr	$50\ \mu\text{m} \leq a \leq 5\text{ mm}$
vnější poloměr	$2\text{ mm} \leq b \leq 1\text{ cm}$

dále jsou zvoleny parametry: $\varepsilon = 4,0 \cdot \varepsilon_0$ $F/m = 4,0 \cdot 8,85418 \cdot 10^{-12} \text{ F/m}$, $Z_0 = 50\ \Omega$

$$\mu = \mu_v = \mu_0 = 4\pi \cdot 10^{-7} \text{ H/m}, \quad c_0 = 2,99792 \cdot 10^8 \text{ m/s}$$

pro každý kus koaxiálního vedení musí také platit: $a < b$ a $b < d$

Algoritmus návrhu filtru pracuje v iteracích. Vždy se pracuje pouze s jedním obvodem (populace má velikost 1). V každé iteraci je provedena změna tohoto obvodu. Takovou změnou může být přidání nového prvku, výměna pořadí dvou prvků (filtr má sekvenční strukturu za sebou jdoucích vedení), přidání nového prvku nebo odebrání prvku. Pokud byla změna úspěšná, tj. pokud se snížila hodnota chybové funkce, bude nový filtr vybrán do následující populace. Takový filtr je následně ještě optimalizován. Pokud ale změna nevedla k přiblížení se požadavkům na výsledný filtr, je tato změna stornována – do následující populace je vybrán rodič. Iterace se v cyklu opakují, dokud není nalezen takový obvod, který splňuje všechny požadavky, nebo dokud není dosaženo maximálního počtu iterací.

Prvním krokem každé iterace je tedy vybrání operace, která bude nad budovaným obvodem provedena. Při evoluci obvodu máme na výběr čtyři následující operace:

1. **Přidání nového prvku** (pravděpodobnost výběru 0,05)
Náhodně je zvolen nový kus koaxiálního vedení a jeho umístění v obvodu.
2. **Odebrání prvku** (pravděpodobnost výběru operace 0,05)
Náhodně je zvolen jeden z prvků obvodu a ten je odebrán.
3. **Výměna dvou prvků** (pravděpodobnost 0,2)
Náhodně jsou zvoleny dva prvky v obvodu a ty jsou prohozeny.
4. **Změna jednoho parametru jednoho prvku** (pravděpodobnost 0,7)
Náhodně je vybrán jeden z prvků obvodu a jeden z jeho parametrů – vnější poloměr, vnitřní poloměr či jeho délka (se stejnou pravděpodobností). Tento parametr je pak změněn o náhodný počet kroků. Velikost kroku byla zvolena $10\ \mu\text{m}$ pro délku vedení a $1\ \mu\text{m}$ pro oba poloměry. Dále je zajištěno splnění výše uvedených podmínek.

Další možností by zde bylo použít normálního rozdělení či volit parametr náhodně v celém intervalu jeho možných hodnot. Při použití normálního rozdělení se dosahovalo přibližně stejných výsledků, volba parametru v celém rozsahu se ukázala jako nevýhodná.

Dalším krokem každé iterace je výpočet chybové funkce. Pokud je tato funkce nulová, je nalezeno požadované řešení a výpočet je ukončen. Pokud je chyba větší, než byla u výchozího obvodu dané iterace, je provedená změna stornována a iterace je ukončena. Program tedy pokračuje další iterací se stejným výchozím obvodem.

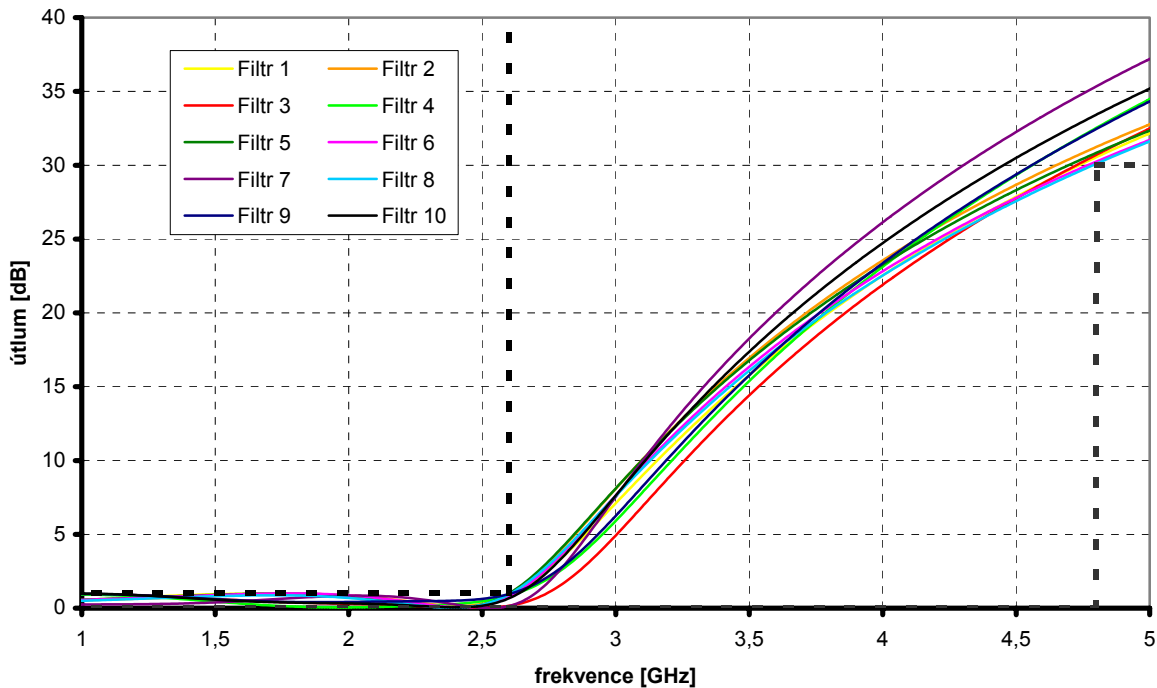
V případě, že se chyba snížila, je provedena optimalizace. Návrh optimalizační metody byl motivován algoritmy lokálního prohledávání [2, 3]. Optimalizace také pracuje v iteracích. V každé iteraci je na začátku náhodně vybrán prvek filtru a jeden z jeho parametrů. Tento parametr je následně optimalizován. Optimalizace se provádí opakovanou změnou parametru daným směrem (přičítání/odečítání hodnoty), dokud se chybová funkce filtru snižuje. Optimalizace je ukončena pokud je dosaženo maximálního počtu iterací, nebo pokud se několikrát za sebou nezdaří iterací parametr vůbec změnit.

Program byl napsán v jazyce JAVA. Jednou z hlavních výhod zvolené implementace je jednoduchá rozšířitelnost programu o další typy dvojbranů. Program lze tedy použít pro návrh filtrů složených i z jiných prvků, než jsou úseky koaxiálních vedení. Program může být také rozšířen o složitější popis útlumových požadavků.

Příklad

Příkladem je návrh filtru, který propouští signály do 2,6 GHz s útlumem do 1 dB a signály nad 4,8 GHz tlumí alespoň o 30 dB. Filtr je přizpůsoben na vstupu i na výstupu na impedanci $50\ \Omega$.

Program byl nad tímto zadáním spuštěn desetkrát a ve všech případech dokázal zadaný filtr bez problémů nalézt, viz obrázek 2. Frekvenční rozsah výpočtu chyby (f_{min} , f_{max}) byl zvolen (100MHz, 20GHz). Počet prvků (úseků koax. vedení) nalezeného řešení kolísá mezi 5 a 8, počet iterací se pohyboval mezi 60 a 190. Doba potřebná pro výpočet nikdy nepřesáhla 20 s (měřeno na počítači Pentium III 1GHz, 128 MB RAM, java JDK 1.3.1.).



Obr. 2. Přenos výsledného filtru

Závěr

Práce jasně ukazuje, že evoluční algoritmy lze s úspěchem použít při návrhu mikrovlnných obvodů. Implementovaný program je navíc jednoduše rozšiřitelný pro návrh libovolného obvodu složeného z kaskádně řazených dvojbranů. Další vývoj by mohl směřovat k návrhu obecnějších mikrovlnných obvodů, například mikropáskových planárních obvodů.

Odkazy

- [1] T. Bäck, F. Hoffmeister, H. P. Schwefel. *A survey of evolution strategies*. Proc. of the Fourth International Conference on Genetic Algorithms and their Applications, San Diego, California, USA, 1991.
- [2] K. Marriot, P. J. Stuckey. *Programming with Constraints: An Introduction*. The MIT Press, 1998
- [3] Z. Michalewicz, D. B. Fogel. *How to Solve It: Modern Heuristics*. Springer, 2000
- [4] D. Whitley. *A Genetic Algorithm Tutorial*. Technical report, Computer Science Department, Colorado State University, 1997.
- [5] A. H. Wright. *Genetic algorithms for real parameter optimization*. Foundations of Genetic Algorithms, pages 205--218. Morgan Kaufmann Publishers, San Mateo, CA, 1991.
- [6] Z. Škvor. *CAD pro VF techniku*. Skriptum ČVUT, 1999
- [7] J. Vrba. *Mikrovlnné obvody*. Skriptum ČVUT, 1999