

Diplomová práce



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra elektromagnetického pole

Symetrie víceportových anténních systémů

Samuel Trávníček

Vedoucí: Ing. Jan Kraček, Ph.D.

Druhý vedoucí: doc. Ing. Pavel Hazdra, Ph.D.

Studijní program: Elektronika a komunikace

Specializace: Rádiové komunikace a systémy

Květen 2022



ZADÁNÍ DIPLOMOVÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Trávníček** Jméno: **Samuel** Osobní číslo: **478068**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra elektromagnetického pole**
Studijní program: **Elektronika a komunikace**
Specializace: **Rádiové komunikace a systémy**

II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

Symetrie víceportových anténních systémů

Název diplomové práce anglicky:

Symmetry of Multiport Antenna Systems

Pokyny pro vypracování:

Seznamte se s teorií a realizací víceportových anténních systémů. Zaměřte se na důsledky symetrie pro jejich charakteristiky a zvolte si příklad víceportového anténního systému pro demonstraci. Charakterizaci víceportového anténního systému proveďte na základě modelování v simulátoru elektromagnetického pole.

Seznam doporučené literatury:

- [1] W. L. Stutzman, G. A. Thiele, Antenna Theory and Design, 3rd Ed., John Wiley & Sons, Hoboken, NJ, USA, 2012.
- [2] J. Sahalos, Orthogonal Methods for Array Synthesis: Theory and the ORAMA Computer Tool, John Wiley & Sons, Hoboken, NJ, USA, 2007.
- [3] R. McWeeny, Symmetry: An Introduction to Group Theory and Its Applications, Pergamon Press, Oxford, UK, 1963.

Jméno a pracoviště vedoucí(ho) diplomové práce:

Ing. Jan Kraček, Ph.D. katedra elektromagnetického pole FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

doc. Ing. Pavel Hazdra, Ph.D. katedra elektromagnetického pole FEL

Datum zadání diplomové práce: **14.02.2022** Termín odevzdání diplomové práce: _____

Platnost zadání diplomové práce: **19.02.2024**

Ing. Jan Kraček, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

16. 5. 2022

Datum převzetí zadání

Podpis studenta

Poděkování

Rád bych poděkoval svým vedoucím diplomové práce panu Ing. Janu Kračkovi, Ph.D. a panu doc. Ing. Pavlu Hazdrovi, Ph.D. za příjemnou spolupráci a podporu v průběhu tohoto semestru.

Prohlášení

Prohlašuji, že jsem zadanou diplomovou práci na téma „Symetrie víceportových anténních systémů“ vypracoval sám s přispěním vedoucích práce a používal jsem pouze literaturu uvedenou v práci. Zároveň souhlasím s tím, aby byla moje práce půjčována nebo zveřejňována pod podmínkou souhlasu katedry.

Abstrakt

Práce zabývá především automatizací vytváření modelu kruhově symetrické anténní řady (monopólů) s jedním aktivním prvkem a daným počtem pasivních prvků s induktivními/kapacitními zátěžemi umístěných podél kružnic kolem aktivního prvku. Zátěže slouží k tvarování výsledného vyzařovacího diagramu anténní řady. Vyzařovací diagramy jsou optimalizovány metodou Particle Swarm Optimization. Je zde zkoumán vliv narušení symetrie (resp. změn symetrie) a vliv změn parametrů anténní řady na tvar její vyzařovací charakteristiky.

Klíčová slova: víceportové antény, symetrie, optimalizace rojem částic, simulace elektromagnetického pole

Vedoucí: Ing. Jan Kraček, Ph.D.

Druhý vedoucí: doc. Ing. Pavel Hazdra, Ph.D.

Abstract

This work focuses on the automation of the generation of circular monopole antenna array with one active element and a given number of passive elements with inductive/capacitive loads placed along circles around the active element. The loads are used to shape the resulting radiation pattern of the antenna array. The radiation patterns are optimized using the Particle Swarm Optimization method. The effects of changes in the symmetry and the parameters of the antenna array on the shape of its radiation pattern are investigated.

Keywords: multiport antennas, symmetries, particle swarm optimization, electromagnetic field simulation

Obsah

1 Úvod	1
1.1 Cíle práce	1
2 Symetrie a jejich matematický popis	3
2.1 Symetrická grupa	3
2.2 Příklad sestavení symetrické grupy	3
2.3 Operátory transformace	5
2.4 Ireducibilní reprezentace a bázové funkce	6
3 Optimalizace	9
3.1 Princip	9
3.2 Volba kritéria	9
3.3 Metody optimalizace	11
3.3.1 Metoda největšího spádu	11
3.3.2 Genetické algoritmy	11
3.3.3 Optimalizace rojem částic	12
3.3.4 Další metody optimalizace	14
4 Generování modelu anténní řady	15
4.1 Propojení CST Microwave Studia s Matlabem	16
4.2 Použití knihoven v Matlabu	16
4.3 Generátor anténní řady	18
4.3.1 Příklad vygenerování řady	18
4.3.2 Parametrický skript	20
5 Nalezení optimálního vyzařovacího diagramu	21
5.1 Syntéza vyzařovacího diagramu anténní řady	22
5.2 Syntéza referenčního vyzařovacího diagramu	23
5.3 Optimalizátor reaktancí a budícího proudu	24
5.4 Parametrický skript	27
5.5 Výsledky optimalizace	27
6 Stabilita vyzařovacího diagramu	31
6.1 Vliv vzájemné orientace vnitřního a vnějšího kruhu	31
6.2 Vliv rozmítání reaktancí	33
6.3 Vliv požadavků na směrovost	34
7 Závěr	37
Literatura	39
A Použití knihovny CST_App	41
A.1 Vytvoření prázdného projektu	41
A.2 Nastavení frekvenčního rozsahu projektu	41
A.3 Nastavení okrajových podmínek	41
A.4 Nastavení počtu buněk na vlnovou délku	42
A.5 Přidání field monitoru	42
A.6 Přidání válce	42
A.7 Přidání diskretního portu	42
A.7.1 S-parametrový port	43
A.7.2 Proudový port	43
A.8 Přidání RLC prvku	43
A.9 Uložení a zavření projektu	43
A.10 Další funkce	44
B Použití knihovny TCSTInterface	45
B.1 Otevření projektu	45
B.2 Spuštění řešiče	45
B.3 Získání S-parametrů	45
B.4 Export vyzařovacích diagramů	46
B.5 Uložení a zavření projektu	47
C Vykreslení vyzařovacích diagramů	49
D Výpočet skalárního součinu vyzařovacích diagramů	51
E Výsledky optimalizace - tabulky	53

Obrázky

2.1 Osy symetrie čtverce	4
3.1 Struktura tenzoru vyzařovacího diagramu	10
3.2 Použitý sférický souřadný systém	10
3.3 Grafické znázornění Metody největšího spádu	12
3.4 Příklad křížení metodou „Cut and splice“	13
3.5 Možnosti chování částice na hranicích prostoru	14
4.1 Příklady analyzovaných anténních řad (pohled shora)	15
4.2 Ukázka <i>History Listu</i> a makra v jazyce VBA	17
4.3 Příklad vygenerované anténní řady se dvěma kruhy pasivních prvků ..	19
5.1 Schéma pro odvození rovnic proudů a napětí	22
5.2 Příklady referenčních vyzařovacích diagramů pro hodnoty parametrů $\phi_G = \pi$, $L = 1$, $A = 1$ a různé q_ϕ .	25
5.3 Ovlivněné a neovlivněné vyzařovací diagramy portů anténní řady	25
5.4 Schematické znázornění konfigurací pro optimalizaci	28
5.5 Příklady optimalizovaných vyzařovacích diagramů a referenční vyzařovací diagram	29
6.1 Příklad vygenerované anténní řady se dvěma kruhy pasivních prvků ..	32
6.2 Graf závislosti chyby Δ_3 na vzájemné orientaci dvou kruhů anténní řady	32
6.3 Graf závislosti budícího proudu I_E na vzájemné orientaci dvou kruhů anténní řady	33
6.4 Graf závislosti chyby Δ_3 na střední kvadratické chybě reaktancí zátěží	34
6.5 Graf závislosti chyby Δ_3 na parametru q_ϕ referenčního vyzařovacího diagramu	35
6.6 Příklady referenčních a optimalizovaných vyzařovacích diagramů na hranici přijatelnosti pro různé q_ϕ	35
D.1 Modely koule	52

Tabulky

2.1 Operace symetrie grupy D_4	5
2.2 Ireducibilní reprezentace prvků grupy D_4	6
5.1 Parametry pro vygenerování první sady anténních řad	27
5.2 Průměrné výsledky optimalizace	28
6.1 Nalezené optimální reaktance pro konfiguraci $\alpha = 0$ rad	32
6.2 Nalezené optimální reaktance pro konfiguraci $\alpha = \pi/12$ rad	33
E.1 Výsledky optimalizace (část 1/3)	53
E.3 Výsledky optimalizace (část 2/3)	54
E.5 Výsledky optimalizace (část 3/3)	55

Kapitola 1

Úvod

Roku 1887 se německý fyzik, pan Heinrich Hertz, pokoušel pomocí prstence z drátu a malého jiskřičky dokázat přítomnost elektromagnetického pole, které vznikne průchodem střídavého proudu vodičem. Byla to první anténa. V dnešní době máme k dispozici analytické a numerické metody pro návrh antén. Ve většině případů používáme právě numerické metody, protože ty analytické často poskytují pouze orientační výsledky a takto navržené antény občas nejsou optimální. Mnohé případy nejsou dokonce ani analyticky řešitelné, pak nezůstává jiná možnost, než použít nějaký simulátor elektromagnetického pole [1]. Numerické metody poskytují tak přesné výsledky, jak přesný je model analyzované struktury. Rozložení elektrického a magnetického pole je získáno přímo z řešení Maxwellových rovnic. Z nich pak dále můžeme získat užitečné informace, jako jsou S-parametry, impedanční matice a vyzařovací diagramy. Nevýhodou numerických metod oproti analytickým je delší výpočetní čas. Ten je ještě mnohem delší pro vyzařující struktury, jako jsou antény.

1.1 Cíle práce

Práce je rozdělena do třech hlavních částí. V první části se zabývá především automatizací vytváření modelu kruhově symetrické anténní řady (monopólů) s jedním aktivním prvkem a daným počtem pasivních prvků s induktivními/kapacitními zátěžemi umístěných podél kružnic kolem aktivního prvku. Zátěže slouží k tvarování výsledného vyzařovacího diagramu anténní řady. Automatizace je dosaženo propojením CST Microwave Studio (dále jen MWS), jakožto elektromagnetického simulátoru, s Matlabem, a výměnou dat mezi nimi.

Druhá část se věnuje nalezení způsobu, jak zkrátit čas výpočtu simulace, a nalezení optimálního vyzařovacího diagramu. K tomuto účelu jsou zátěže na pasivních prvcích nahrazeny nejprve proudovými porty, které slouží k získání vyzařovacích diagramů všech portů (včetně aktivního prvku). Dále jsou nahrazeny za S-parametrové porty, pomocí kterých je získána impedanční matice. Z takto získaných parametrů se pak pro zadané reaktance zátěží na pasivních prvcích vypočítá pomocí analytických metod výsledný vyzařovací diagram anténní řady. Ten je následně optimalizován metodou *Particle swarm optimization* (Optimalizace rojem částic), dále jen *PSO*.

Ve třetí části je zkoumán vliv narušení symetrie (resp. změn symetrie) a vliv změn parametrů anténní řady na tvar její vyzařovací charakteristiky.

Kapitola 2

Symetrie a jejich matematický popis

Pokud mluvíme o operaci symetrie, máme na mysli takovou transformaci, která zobrazí geometrické těleso samo na sebe. Po transformaci tedy nelze určit, zda se orientace tělesa změnila či nikoliv. Takovou transformaci můžeme matematicky popsat například pomocí bijektivního zobrazení, které zobrazí body tělesa na jiné body tělesa. Dalším způsobem popisu transformací jsou matice.

2.1 Symetrická grupa

Skupinu operací symetrie nazýváme symetrická grupa. V souvislosti s anténami a jejich používáními geometrii má symetrická grupa obvykle konečný počet prvků. Každý prvek symetrické grupy vyjadřuje jednu operaci symetrie. Počet prvků v grupě nazýváme řádem grupy. Operacemi symetrie mohou být identita, rotace, zrcadlení a další.

Každá symetrická grupa obsahuje minimálně jeden prvek a tím je identita. Značíme jí E . Ta zobrazí všechny body geometrického tělesa samy na sebe. Tato operace je také ekvivalentní rotaci o 360 stupňů. Operaci rotace často značíme C_n , kde n vyjadřuje úhel rotace jako $360^\circ/n$. Dále pak za n připojujeme písmeno vyjadřující osu, kolem které rotujeme. Příkladem je třeba operace C_{4z} , která vyjadřuje rotaci kolem osy z v kartézském souřadném systému o 90 stupňů. Rotaci o záporný úhel vyjádříme exponentem -1 . Tedy rotaci o -90 stupňů (resp. o 270 stupňů) zapíšeme jako C_{4z}^{-1} .

Násobením prvků symetrické grupy vykonáváme jednotlivé operace symetrie v daném pořadí. Násobením dvou prvků symetrické grupy vždy dostaneme další prvek stejné grupy. Pro každý prvek symetrické grupy existuje inverzní prvek takový, že vynásobením těchto dvou prvků dostaneme identitu. Prvky symetrické grupy spolu vzájemně obecně nekomutují, tedy záleží na pořadí operací.

2.2 Příklad sestavení symetrické grupy

Jako příklad bylo převzato odvození pro čtverec z článku [2]. Operaci symetrie (transformaci) budeme značit T . Symetrickou grupu pravidelného n -úhelníka

Op. symetrie T	Matice $\mathbf{R}(T)$	Operace $P(T)\mathbf{J}(\mathbf{r})$
E	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} J_x(x, y, z) \\ J_y(x, y, z) \\ J_z(x, y, z) \end{pmatrix}$
C_{4z}	$\begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} J_y(-y, x, z) \\ -J_x(-y, x, z) \\ J_z(-y, x, z) \end{pmatrix}$
C_{2z}	$\begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} -J_x(-x, -y, z) \\ -J_y(-x, -y, z) \\ J_z(-x, -y, z) \end{pmatrix}$
C_{4z}^{-1}	$\begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} -J_y(y, -x, z) \\ J_x(y, -x, z) \\ J_z(y, -x, z) \end{pmatrix}$
C_{2x}	$\begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$	$\begin{pmatrix} J_x(x, -y, -z) \\ -J_y(x, -y, -z) \\ -J_z(x, -y, -z) \end{pmatrix}$
C_{2y}	$\begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$	$\begin{pmatrix} -J_x(-x, y, -z) \\ J_y(-x, y, -z) \\ -J_z(-x, y, -z) \end{pmatrix}$
C_{2a}	$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix}$	$\begin{pmatrix} J_y(y, x, -z) \\ J_x(y, x, -z) \\ -J_z(y, x, -z) \end{pmatrix}$
C_{2b}	$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix}$	$\begin{pmatrix} -J_y(-y, -x, -z) \\ -J_x(-y, -x, -z) \\ -J_z(-y, -x, -z) \end{pmatrix}$

Tabulka 2.1: Operace symetrie grupy D_4

2.3 Operátory transformace

Až do této chvíle jsme brali v úvahu pouze transformaci souřadného systému (kartézských souřadnic). Při práci s charakteristickými módy nás hlavně zajímá, co se stane s vektorovými funkcemi povrchových proudových hustot $\mathbf{J}(\mathbf{r}) = (J_x(\mathbf{r}), J_y(\mathbf{r}), J_z(\mathbf{r}))^T$ na vyzářovací struktuře antény. Proto zavedeme operátor transformace $P(T)$, který transformuje původní funkce na nové podle dané operace symetrie. Operátor $P(T)$ vyjádříme pomocí transformační matice $\mathbf{R}(T)$ ve vztahu (dle [2]):

$$\mathbf{J}'(\mathbf{r}) = P(T)\mathbf{J}(\mathbf{r}) = \mathbf{R}(T)\mathbf{J}(\mathbf{R}^{-1}(T)\mathbf{r}). \quad (2.1)$$

Pojďme si nyní dosazením do rovnice odvodit, jak by vypadal operátor pro operaci C_{4z} v trojrozměrném kartézském souřadném systému:

$$\begin{aligned} \mathbf{J}'(\mathbf{r}) &= P(T)\mathbf{J}(\mathbf{r}) = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \mathbf{J} \left(\begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} \right) = \\ &= \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \mathbf{J} \left(\begin{pmatrix} -y \\ x \\ z \end{pmatrix} \right) = \begin{pmatrix} -J_y(-y, x, z) \\ J_x(-y, x, z) \\ J_z(-y, x, z) \end{pmatrix}. \end{aligned}$$

Stejným způsobem můžeme odvodit i další operátory pro ostatní operace symetrie. Kompletní seznam operátorů pro grupu D_4 je v tabulce 2.1.

2.4 Ireducibilní reprezentace a báze funkce

Pro každou symetrickou grupu existuje speciální sada lineárně nezávislých vektorových funkcí ψ , pro které platí, že pokud je transformujeme pomocí operátoru $P(T)$, můžeme je vyjádřit (dle [2]) jako lineární kombinací ostatních:

$$P(T)\psi_n^{(p)}(\mathbf{r}) = \sum_{m=1}^{d_p} \Gamma_{mn}^{(p)}(T)\psi_m^{(p)}(\mathbf{r}), \quad n = 1, 2, \dots, d_p. \quad (2.2)$$

Funkce $\psi_n^{(p)}$ se nazývá n -tá báze funkce p -té d_p -rozměrné reprezentace symetrické grupy. Výsledek operace T aplikované na funkci $\psi_n^{(p)}$ je ekvivalentní lineární kombinaci d_p báze funkcí $\psi_m^{(p)}$. Váhovací koeficienty $\Gamma_{mn}^{(p)}(T)$ jsou prvky d_p -rozměrné čtvercové matice $\mathbf{\Gamma}^{(p)}(T)$ (n značí řádky, m sloupce). Matice $\mathbf{\Gamma}^{(p)}(T)$ se nazývá ireducibilní reprezentace operace symetrie T . Řádky matic ireducibilních reprezentací jsou ortogonální (násobením koeficientů dvou řádků stejně rozměrných matic mezi sebou a součtem dostaneme nulu). Každá symetrická grupa má jednu ireducibilní reprezentaci, která má všechny členy rovné 1. Existuje 5 ireducibilních reprezentací pro čtverec, jsou shrnuty v tabulce 2.2.

T	E	C_{4z}	C_{2z}	C_{4z}^{-1}	C_{2x}	C_{2y}	C_{2a}	C_{2b}
$\Gamma^{(1)}(T)$	1	1	1	1	1	1	1	1
$\Gamma^{(2)}(T)$	1	1	1	1	-1	-1	-1	-1
$\Gamma^{(3)}(T)$	1	-1	-1	1	1	1	-1	-1
$\Gamma^{(4)}(T)$	1	-1	-1	1	-1	-1	1	1
$\Gamma^{(5)}(T)$	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$	$\begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}$	$\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$	$\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & -1 \\ -1 & 0 \end{pmatrix}$

Tabulka 2.2: Ireducibilní reprezentace prvků grupy D_4

Ireducibilní reprezentace získáme tak, že rozložíme matice jednotlivých operací symetrie na menší matice podél hlavní diagonály. Každá sada těchto menších matic napříč všemi operacemi symetrie vždy tvoří jednu ireducibilní

reprezentaci. Tyto matice už dále nemůžeme rozložit na méněrozměrné matice, protože bychom ztratili nějakou informaci. Pro grupu D_4 byly dvě reprezentace získány tak, že se všechny matice symetrických operací rozložily na dvě matice - jednu o velikosti 2×2 a jednu o velikosti 1×1 . Další reprezentace je jednotková, která je vždy přítomna. Poslední dvě byly získány na základě ortogonality a dalších vlastností ireducibilních reprezentací, které zde nebyly zmíněny. Jejich odvození však není v tuto chvíli podstatné, více podrobností lze dohledat například v [3].

Symetrie a jejich ireducibilní reprezentace souvisí s charakteristickými módy, kterými se však přímo tato práce nezabývá. Více o použití charakteristických módů v [4, 5, 6].

Kapitola 3

Optimalizace

Tato kapitola se zabývá metodami optimalizace, zvláště pak metodou PSO, která je v této práci použita pro optimalizaci vyzařovacích diagramů všech anténních řad.

3.1 Princip

Optimalizací máme na mysli nalezení optimálních hodnot parametrů tak, aby byl konečný výsledek co nejbližší požadovanému. V rámci této práce jsou optimalizovány reaktance zátěží [7, 8, 9, 10, 11] na pasivních prvcích anténní řady. Požadovaným výsledkem je nějaký vyzařovací diagram s daným předpisem. Pro porovnání dosaženého výsledku s požadovaným se používá tzv. účelová funkce (také kritérium, chybový povrch, cost function). Účelem optimalizace je nalezení minima (případně maxima) účelové funkce. Naivním řešením by bylo diskretizovat prostor daný počtem parametrů a jejich rozsahy a pro každý bod spočítat účelovou funkci. Z této množiny čísel by se pak vybralo minimum a řešením by byly příslušné parametry v daném bodě.

3.2 Volba kritéria

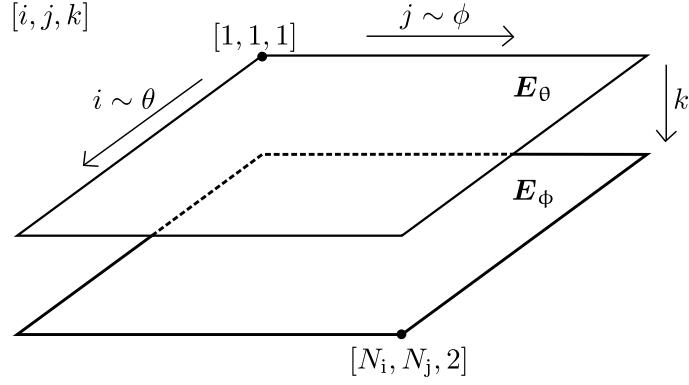
Označme tenzor prostorového vyzařovacího diagramu získaného pro jednu sadu reaktancí během optimalizace \mathbf{f}_A . Ten má strukturu podle obrázku 3.1, první dva indexy i a j odpovídají indexům úhlů θ a ϕ . Index k odpovídá složkám elektrického pole \mathbf{E}_θ a \mathbf{E}_ϕ . Dále referenční (požadovaný) vyzařovací diagram označíme \mathbf{f}_G . Všechny matematické operace v této sekci jsou uvažovány po složkách tenzorů.

Prvním možné kritérium je dané vztahem

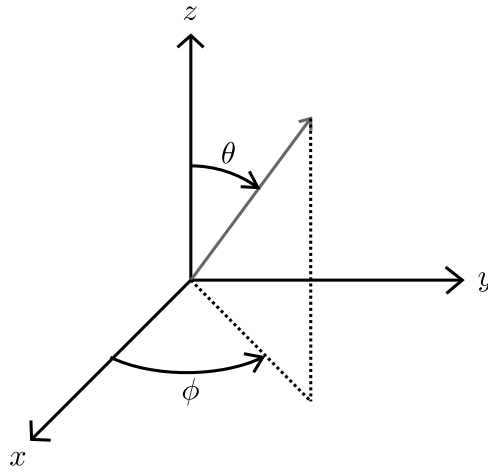
$$\Delta_1 = \frac{\|\mathbf{f}_G - \mathbf{f}_A\|}{\|\mathbf{f}_G\|}, \quad (3.1)$$

kde $\|x\|$ je norma definovaná jako

$$\|x\| = \langle x, x \rangle, \quad (3.2)$$



Obrázek 3.1: Struktura tenzoru vyzařovacího diagramu



Obrázek 3.2: Použitý sférický souřadný systém

kde $\langle x, y \rangle$ značí skalární součin x a y .¹ Toto kritérium je nejpřísnější, protože uvažuje amplitudy i fáze obou složek vyzařovacího diagramu \mathbf{E}_θ a \mathbf{E}_ϕ . Toto kritérium je vhodné zejména pokud chceme kromě tvaru amplitudové vyzařovací charakteristiky také optimalizovat fázovou vyzařovací charakteristiku. Pro účely této práce je ovšem toto kritérium zbytečně přísné.

Dalším možné kritérium je dané vztahem

$$\Delta_2 = \frac{\| |\mathbf{f}_G| - |\mathbf{f}_A| \|}{\|\mathbf{f}_G\|}, \quad (3.3)$$

kde $|\cdot|$ představuje absolutní hodnotu po složkách tenzoru. V tomto případě se uvažuje pouze amplitudová vyzařovací charakteristika. Toto kritérium je vhodné, pokud chceme optimalizovat zvláště složky \mathbf{E}_θ a \mathbf{E}_ϕ , ale nezajímá nás jejich fáze.

Posledním uvedeným kritériem je

$$\Delta_3 = \frac{\|\mathbf{E}_G - \mathbf{E}_A\|}{\|\mathbf{f}_G\|}, \quad (3.4)$$

¹Algoritmus pro výpočet skalárního součinu pro dva vyzařovací diagramy je uveden v příloze D.

kde $\mathbf{E}_{G,A}$ jsou matice amplitudy celkového pole spočítané jako

$$\mathbf{E}_{G,A} = \sqrt{|\mathbf{E}_{G\theta,A\theta}|^2 + |\mathbf{E}_{G\phi,A\phi}|^2}, \quad (3.5)$$

tedy prvky matice $\mathbf{E}_{G,A}$ tvoří velikosti vektorů danými složkami \mathbf{E}_θ a \mathbf{E}_ϕ . Toto kritérium je nejslabší ze všech, protože slučuje jednotlivé složky elektrického pole \mathbf{E}_θ a \mathbf{E}_ϕ a zanedbává jejich fáze. Pro účely této práce je poslední kritérium dostatečné, protože použité anténní řady se skládají pouze z monopólů umístěných do osy z , tedy nemají složku \mathbf{E}_ϕ .²

3.3 Metody optimalizace

3.3.1 Metoda největšího spádu

Metoda největšího spádu (angl. Gradient descent method) slouží pro nalezení lokálního minima účelové funkce. S oblibou se tato metoda používá pro strojové učení v neuronových sítích, protože tam se všechna lokální minima nacházejí velmi blízko toho globálního. Algoritmus je následující:

1. Zvolí se náhodný vektor \vec{x} v prostoru parametrů.
2. V každém bodě iterace se získá gradient účelové funkce ∇f v daném bodě \vec{x} , ten vždy ukazuje směrem k maximu. Směr k minimu je tedy vektor s opačným znaménkem.
3. Nový vektor \vec{x} se spočítá jako

$$\vec{x}_{n+1} = \vec{x}_n - \alpha \cdot \nabla f, \quad (3.6)$$

kde α je koeficient rychlosti optimalizace.

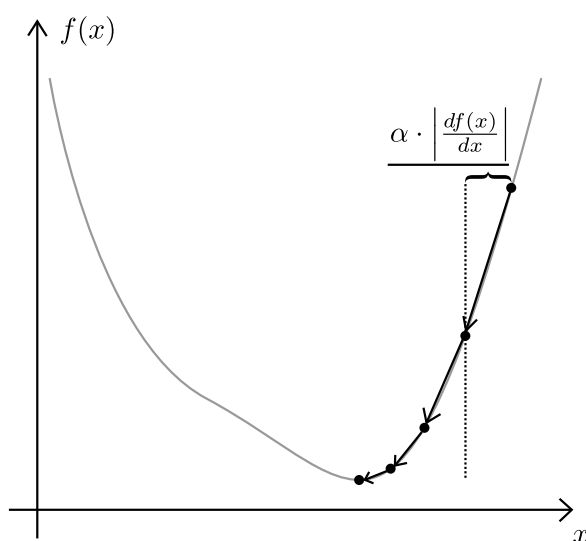
4. Konvergenci lze vyhodnotit například podle relativní změny vektoru \vec{x} .

Grafické znázornění metody pro jednorozměrnou účelovou funkci je na obrázku 3.3. V případě jednorozměrné funkce je $\nabla f = \frac{df(x)}{dx}$. Pro příliš malé α trvá optimalizace delší dobu (zabere větší počet kroků), naopak pro příliš velké α nemusí dojít ke konvergenci, místo toho může dojít například k oscilaci kolem optima.

3.3.2 Genetické algoritmy

Tato skupina metod je inspirována biologií. Je založena na populacích, kde každý jedinec populace představuje jedno řešení (vektor \vec{x}). Každé číslo v paměti počítače (ať už desetinné nebo celé) je reprezentováno bity. K bitům lze přistupovat jako ke genetickému kódu. Jedinci se pak navzájem mohou křížit, mutovat nebo se reprodukovat. Ve všech variantách je algoritmus podobný tomuto:

²Ve skutečnosti mají velmi zanedbatelnou složku \mathbf{E}_ϕ , protože se jednotlivé monopóly navzájem ovlivňují.



Obrázek 3.3: Grafické znázornění Metody největšího spádu

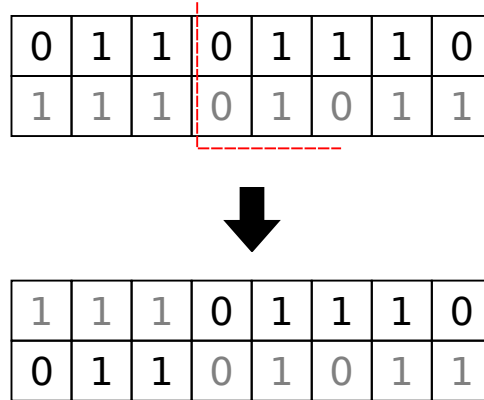
1. Vygeneruje se vybraný počet náhodných jedinců (počáteční populace).
2. V každém kroku iterace (epocha) se provede následující:
 - a. Dle implementace se vyberou jedinci (náhodně nebo podle nějakého kritéria).
 - b. Z těchto jedinců se vytvoří potomci a to křížením (část bitů se vezme z jednoho jedince a část z jiného, příklad na obrázku 3.4), mutací (náhodné změny bitů) a reprodukci (vytvoří se kopie jedince).
 - c. Vznikne nová populace. Pro každého jedince se spočítá účelová funkce (fitness funkce - maximalizuje se). Zvolené množství s nejvyšším hodnocením postoupí do další epochy.
3. Po zvoleném počtu epoch nebo při splnění ukončovací podmínky zůstane populace, která představuje řešení blízka optimálnímu.

Algoritmus hledá obecně globální maximum a existuje v mnoha variantách.

3.3.3 Optimalizace rojem částic

Anglicky Particle swarm optimization (PSO). Další z biologicky inspirovaných algoritmů. Algoritmus je založený na částicích (agentech), které mají polohový vektor \vec{r}^i , vektor rychlosti \vec{v}^i a hmotnost m^i , kde i je index částice. Částice se pohybují obdobně, jako objekty reálného světa podle Newtonových zákonů. Navíc mohou částice komunikovat mezi sebou. Algoritmus je následující:

1. Náhodně se vygeneruje zvolený počet částic (příliš malý počet nevede k nalezení globálního maxima, ale lokálního, příliš velký počet zpomaluje proces optimalizace). Minimální počet částic je $N + 1$, kde N je počet parametrů k optimalizaci (velikost vektoru \vec{x}).



Obrázek 3.4: Příklad křížení metodou „Cut and splice“

2. Každé částici je přiřazen náhodný polohový vektor a vektor rychlosti.
3. Částice mohou na své poloze spočítat fitness funkci. Také si pamatují polohu \vec{R} s nejvyšší samostatně nalezenou hodnotou fitness funkce a její hodnotu.
4. Částice se mohou zeptat ostatních částic na jejich uložené polohy nalezeného maxima fitness funkce.
5. V každém bodě iterace se provede následující:
 - a. Částice mají tendenci setrvat v rovnoměrném přímočarém pohybu, ale zároveň jsou ovlivňovány polohami nalezených maxim \vec{R}^i , ke kterým jsou přitahovány.
 - b. Jejich vektory rychlosti jsou aktualizovány takto:

$$\vec{v}_{n+1}^i = m^i \vec{v}_n^i + \alpha \left(\vec{R}_n^i - \vec{r}_n^i \right) + \beta \left(\vec{G}_n - \vec{r}_n^i \right) + \gamma \vec{u}, \quad (3.7)$$

kde

$$\vec{G}_n = \arg \max_i f \left(\vec{R}_n^i \right) \quad (3.8)$$

je poloha globálního optima, \vec{u} je jednotkový vektor náhodných čísel a koeficienty α , β a γ určují, jak velký význam mají jednotlivé atraktory (lokální, globální maximum a náhodný faktor) pro částice.

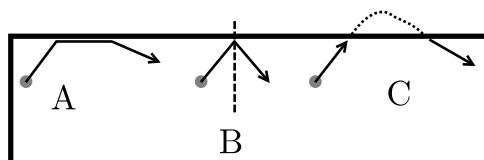
- c. Nové polohy se spočítají podle vztahu:

$$\vec{r}_{n+1}^i = \vec{r}_n^i + \Delta t \cdot \vec{v}_n^i, \quad (3.9)$$

kde Δt je zvolený časový krok.

- d. Pro novou polohu se spočítá fitness funkce a pokud je její hodnota vyšší než uložená, aktualizuje se poloha maxima.
- e. Pokud je nová poloha mimo hranice vymezeného prostoru optimalizovaných proměnných, pak může být tato situace řešena například jedním z následujících způsobů:

- (i) Normálová složka na hranici prostoru vektoru rychlosti částice je nulována (částice klouže po hranici, případ **A** na obrázku 3.5).
- (ii) Aplikuje se Snellův zákon (částice se odrazí pod stejným úhlem, pod kterým dopadla na hranici, případ **B** na obrázku 3.5).
- (iii) Částice vyletí z prostoru, ale v době, kdy je mimo prostor se neaktualizuje její nalezené maximum fitness funkce (případ **C** na obrázku 3.5).



Obrázek 3.5: Možnosti chování částice na hranicích prostoru

Pro účely této práce byla zvolena tato metoda optimalizace. K dispozici je také v Matlabu jako funkce *particleswarm* [12].³

3.3.4 Další metody optimalizace

Metod optimalizací existuje velké množství, jsou to například:

1. Evoluční algoritmy
2. Monte Carlo
3. Konvexní optimalizace
4. Multi-objektové optimalizace

Samotné MWS pak nabízí například tyto metody:

1. CMA-ES (The Covariance Matrix Adaptation Evolutionary Strategy)
2. TRF (Trust Region Framework)
3. Genetický algoritmus
4. PSO
5. Nelder Mead Simplex Algorithm
6. Interpolated Quasi Newton
7. Powellova metoda

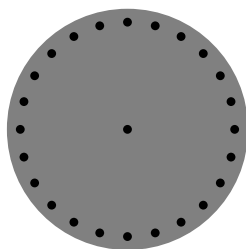
Popisy jednotlivých metod a jejich výhod a nevýhod jsou k dispozici na webových stránkách [13]. Proč v rámci této práce nebylo využito MWS pro optimalizaci vyzářovacích diagramů bude vysvětleno v kapitole 5.

³Je potřeba mít nainstalovaný *Global Optimization Toolbox*. Přesná implementace metody je popsána zde: <https://www.mathworks.com/help/gads/particle-swarm-optimization-algorithm.html>.

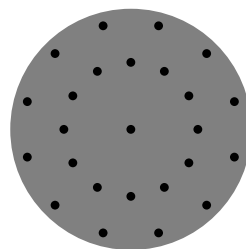
Kapitola 4

Generování modelu anténní řady

Tato práce se zabývá analýzou různých variant kruhově symetrické anténní řady monopólových antén (příklad dvou takových anténních řad je na obrázku 4.1; uprostřed je vždy aktivní monopól, kolem jsou kapacitně/induktivně zatížené pasivní monopóly, šedá plocha je zemní rovina, která je ale okrajovými podmínkami v MWS protažena do nekonečna).



(a) : s jedním kruhem pasivních prvků



(b) : se dvěma kruhy

Obrázek 4.1: Příklady analyzovaných anténních řad (pohled shora)

Vzhledem ke značnému množství konfigurací, které byly analyzovány, bylo časově velmi náročné všechny modely v MWS vytvořit manuálně. MWS do jisté míry umožňuje parametrizaci modelu. Takové řady by bylo možné vytvořit například pomocí jednoho monólu, který by se nakopíroval operací *Rotace* kolem aktivního prvku v ose z . Problém tohoto přístupu je nutnost takovou operaci provést pro každý kruh zvlášť. Navíc první prvek z kruhu musí být vždycky umístěn manuálně. To platí i pro zemní rovinu, přidání farfield monitorů pro požadovanou frekvenci, nastavení projektu (např. hraniční podmínky), umístění portů a zátěží, atd.

Další komplikace nastává při získávání a zpracování výstupních dat, jako jsou S-parametry a vyzařovací diagramy. Tyto je obvykle nutné vyexportovat do nějakého textového souboru (například ve formátu CSV). Poté mohou být data načtena do Matlabu a transformována do nějakého užitečnějšího formátu (matice, tenzory, struktury, ...).

Z výše uvedených důvodů se v rámci této práce (a pro případné budoucí projekty) přistoupilo k alternativnímu řešení a tím je automatizace všech úkonů pomocí Matlabu.

4.1 Propojení CST Microwave Studio s Matlabem

Pro propojení MWS s Matlabem byla nalezena knihovna *TCSTInterface*, která pro komunikaci využívá technologii COM operačního systému Windows. Knihovna je vhodná, jak uvádí na své webové stránce,¹ pro manipulaci s již existujícím projektem. V rámci této práce je použita pro spouštění řešiče (solveru) a pro extrakci výstupních dat. Na webové stránce je také odkaz² na jinou knihovnu *hgiddenss/CST_App*, která je zase vhodná pro vytváření projektů a vytváření komponent (přestože umí i spouštět řešič a extrahovat nějaká data v omezené míře).

Tyto dvě knihovny byly použity pro generování projektů, jejich nastavení, generování struktur, spouštění řešiče a extrakci dat. Bohužel samotné knihovny neobsahují všechny funkce, které byly potřeba pro realizaci projektů. Jedná se především o funkce pro vytváření proudových portů, sériových RLC prvků a nastavení počtu buněk na vlnovou délku.

MWS je založené na makrech v jazyce VBA (Visual Basic for Application). Veškeré operace prováděné v uživatelském prostředí jsou zaznamenány do tzv. *History Listu* (obrázek 4.2), odkud mohou být prohlíženy, upravovány a opakovaně vykonávány. Většinu operací provádí knihovny právě voláním VBA skriptů. Přestože existuje určitá forma dokumentace VBA pro MWS, mnohé vlastnosti tam nejsou popsány. Proto bylo využito právě *History Listu* pro implementaci chybějících funkcí. Upravené knihovny jsou k dispozici na GitLabu v repozitáři dedikovaném pro tuto práci.³ Kromě zmíněných knihoven obsahuje repozitář také všechny pomocné funkce použité v projektu a parametrické skripty pro generování anténních řad a optimalizaci jejich vyzářovacích charakteristik.

4.2 Použití knihoven v Matlabu

Pro načtení knihoven *TCSTInterface* a *CST_App* a všech ostatních pomocných funkcí stačí přidat cestu ke knihovně a zavolat funkci *loadLibs*. Za předpokladu, že je cesta k repozitáři s knihovnami v sousedním adresáři vypadá inicializace takto:⁴

```
1 addpath(' ../lib');
2 loadLibs();
```

Vytvoření prázdného projektu (případně otevření již existujícího) pomocí knihovny *CST_App* se provede při vytvoření instance:⁵

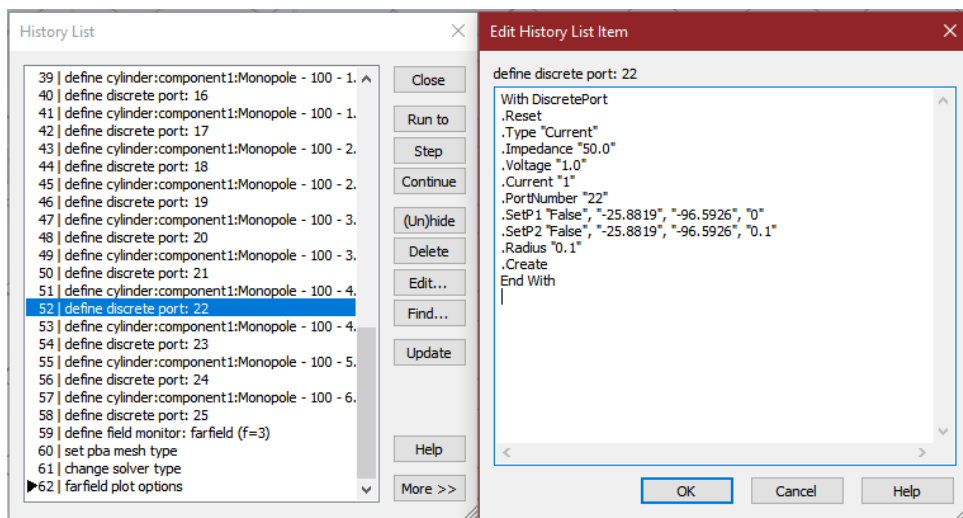
¹<https://www.mathworks.com/matlabcentral/fileexchange/72228-tcstinterface-cst-studio-suite-to-matlab-interface>

²https://www.mathworks.com/matlabcentral/fileexchange/67731-hgiddenss-cst_app

³<https://gitlab.com/sam.travnicek/thesis-symmetries>

⁴V novějších verzích Matlabu je možné zapisovat cesty k souborům pomocí dopředných lomítek nejen v operačních systémech Linux a macOS, ale i v operačním systému Windows.

⁵Pro komunikaci musí být CST Microwave Studio spuštěné.

Obrázek 4.2: Ukázka *History Listu* a makra v jazyce VBA

```
1 CST = CST_MicrowaveStudio(...
2   pathToDirectoryWithCstProject, cstProjectFileName);
```

Pokud adresář projektu ještě neexistuje, je automaticky vytvořen, pokud je potřeba. Konkrétní příklad by mohl vypadat takto:

```
1 CST = CST_MicrowaveStudio('CST_Projects', 'MyProject.cst');
```

Pomocí objektu *CST* pak můžeme provádět další operace nad projektem. Například uložení:

```
1 CST.save();
```

nebo zavření projektu:

```
1 CST.closeProject();
```

Podobným způsobem probíhá nastavení projektu a jsou vytvářeny komponenty (například válce představující monopóly a zemní rovinu). Všechny ostatní operace týkající se této práce jsou popsány v příloze A.

Otevření projektu knihovnou *TCSTInterface* probíhá takto:

```
1 CST = TCSTInterface();
2 CST.OpenProject(pathToProjectFile);
```

Případně přímo jedním voláním:

```
1 CST = TCSTInterface(pathToProjectFile);
```

V konkrétním případě by to tedy mohlo vypadat takto:

```
1 CST = TCSTInterface(fullfile('CST_Projects', 'MyProject.cst'));
```

Spuštění řešiče se provede:

```
1 CST.Solve();
```

A zavření projektu:

```
1 CST.CloseProject([], true);
```

Hodnota *true* říká knihovně, aby projekt před zavřením uložila. Všechny ostatní operace týkající se této práce jsou taktéž popsány v příloze B.

4.3 Generátor anténní řady

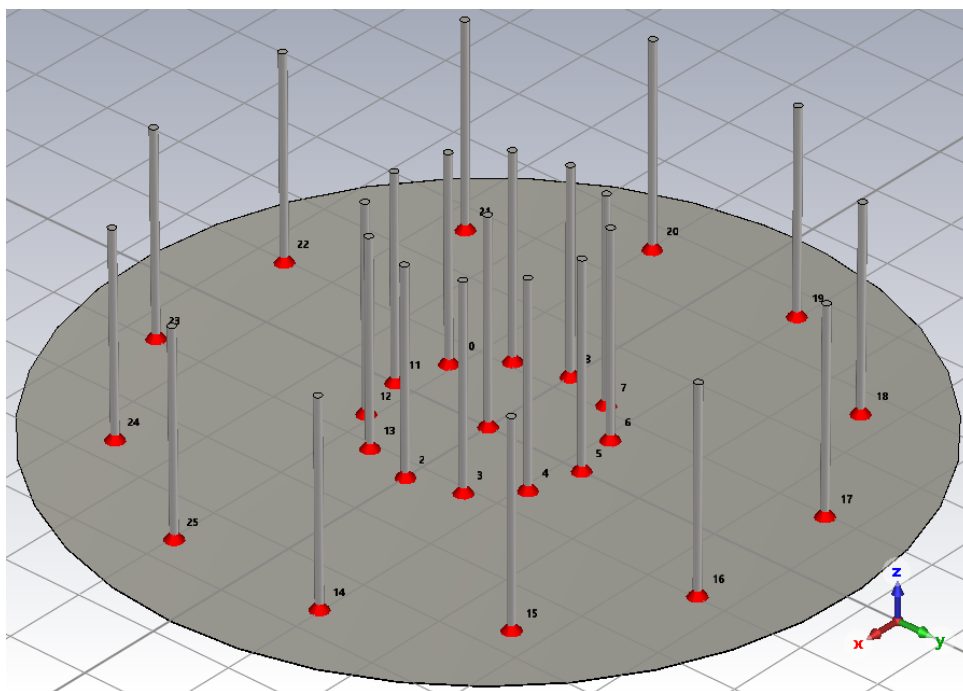
Generátor anténní řady byl implementován v podobě třídy s názvem *MonopoleArrayGenerator*. V repozitáři se nachází ve složce *lib/Custom*. Umožňuje vygenerovat kruhově symetrické monopólové anténní řady tak, jak jsou vyobrazeny na obrázku 4.1, s volitelným počtem a umístěním kruhů s pasivními prvky.

4.3.1 Příklad vygenerování řady

Vygenerování řady, za předpokladu, že máme objekt *CST* vytvořený pomocí knihovny *CST_App*, se provede následovně:

```
1 gen = MonopoleArrayGenerator;
2 gen.setInterface(CST);
3 gen.setMainMonopoleLength(25);
4 gen.setMonopoleRadius(0.5);
5 gen.setGroundPadding(10);
6 gen.setGroundThickness(0.1);
7 gen.setGap(0.1);
8
9 % addRing([radius rotAngle], monopoleLength, loadRLCs)
10 gen.addRing([12.5 0], 25, zeros(6, 3));
11 gen.addRing([37.5 pi/12], 25, zeros(6, 3));
12
13 gen.setPortType('Current');
14 gen.generateWithPorts();
```

Výsledná anténa je na obrázku 4.3. Na druhém řádku se nastaví rozhraní pro komunikaci s MWS (objekt *CST*). Na řádcích 3 až 7 jsou nastaveny obecné parametry - délka hlavního (aktivního) monopólu, poloměr monopólů (polovina jejich průřezu), mezera mezi vnějším kruhem a okrajem zemní roviny, tloušťka zemní roviny a mezera mezi zemní rovinou a monopóly. Mezi zemní rovinou a monopóly umožňuje generátor umístit buď proudové porty, S-parametrové porty nebo sériově zapojené RLC prvky (zátěže). Vygenerování s diskrétními porty se provede takto:



Obrázek 4.3: Příklad vygenerované anténní řady se dvěma kruhy pasivních prvků

```
1 gen.setPortType('Current'); % or 'SParam'
2 gen.generateWithPorts();
```

A vygenerování se zátěžemi se provede takto:

```
1 gen.generateWithLoads();
```

Metodou *addRing* se přidávají jednotlivé kruhy pasivních monopólů. Prvním parametrem je vektor udávající poloměr a rotaci kruhu. Druhým parametrem se nastavuje výška všech monopólů v daném kruhu. Třetím parametrem je matice impedancí jednotlivých zátěží. Ta má takovou strukturu, že na řádcích jsou jednotlivé hodnoty odporu, indukčnosti a kapacity (v základních jednotkách) pro každou zátěž. V případě, že je projekt vygenerován s diskretními porty, jsou tyto hodnoty ignorovány, tedy mohou být vyplněny například nulami. Počet řádků této matice určuje, kolik pasivních monopólů bude do kruhu umístěno. Ty jsou umístěny rovnoměrně. Úhel α mezi jednotlivými monopóly je dán vztahem

$$\alpha = \frac{2\pi}{N}, \quad (4.1)$$

kde N je počet monopólů v kruhu. Rotace kruhu je vztažena k ose x , na níž vždy leží první monopól kruhu. Čísla portů jsou přiřazována tak, že aktivní monopól má vždy číslo 1. Další čísla jsou přiřazována postupně, od prvního kruhu až k poslednímu, od prvního monopólu k poslednímu. Tedy vnitřní kruh obsahuje nejnižší čísla portů, vnější kruh nejvyšší. To je ukázáno na obrázku 4.3.

■ 4.3.2 Parametrický skript

Pro získání všech potřebných dat a jejich konverzi do souborů *.mat* byl vytvořen parametrický skript, který se nachází v repozitáři ve složce *arrgen* pod názvem *generateData.m*. Ten umožňuje nastavit všechny uvedené parametry z předchozí části. Jako jeden z parametrů se také zadává *DATA_ID*, který určuje, kam se vybraná data uloží. Data jsou obecně ukládána do sousední složky s názvem *Data*. Dále v ní se vytvoří složka s názvem hodnoty parametru *DATA_ID*. V té se nachází složka *CST*, určená pro uložení projektů a příslušných souborů MWS, a složka *Matlab* pro uložení výstupních dat v podobě souborů *.mat*. Díky tomuto uspořádání je možné zpětně dohledat geometrii každé konfigurace i všechny další případné informace, které nebyly vyexportovány do souborů *.mat*.

Skript pro potřeby analýzy vytvoří dva projekty MWS. Jeden s názvem *S-Ports.cst*, ten obsahuje anténní řadu s S-parametrovými porty a druhý s názvem *I-Ports.cst*, který obsahuje geometricky identickou řadu ale s proudovými porty. Pro oba projekty se spustí analýza a následně jsou vyexportovány S-parametry z prvního projektu a vyzařovací diagramy z druhého projektu.⁶ Nakonec jsou dopočítány impedanční matice z S-parametrů.

Výstupní data ve složce *Matlab* jsou rozdělena do souboru *coords.mat* obsahující vektory souřadnic θ a ϕ , souboru *Z.mat*, obsahující impedanční matice pro frekvenční rozsah projektu, a souborů *ff<x>.mat* ($\langle x \rangle$ je číslo portu), které obsahují vyzařovací charakteristiky jednotlivých portů.

Skript je podrobně zdokumentovaný pomocí komentářů a všechny parametry jsou v něm patřičně vysvětleny.

⁶Proudové vyzařovací diagramy je možné získat přepočtem z S-parametrových vyzařovacích diagramů, ale pro jednoduchost bylo zvoleno řešení pomocí dvou projektů.

Kapitola 5

Nalezení optimálního vyzařovacího diagramu

Vyzařovací diagramy jsou realizovány kruhově symetrickou anténní řadou s jedním aktivním monopólem uprostřed a kruhy pasivních monopólů zatížených sériovými RLC prvky soustředěných kolem aktivního. Změnou hodnot RLC prvků na pasivních monopólech lze měnit tvar vyzařovací charakteristiky anténní řady. Vzhledem k tomu, že rezistory jsou ztrátové prvky, nemá smysl jejich použití, naopak induktory a kapacitory jsou (v ideálním případě) beztrátové a mění fázi vyzařovacích charakteristik jednotlivých pasivních prvků. Superpozicí takto posunutých fází pak vzniknou různé celkové vyzařovací charakteristiky.

Aby byla případná fyzická konstrukce jednodušší, je vhodné použít pouze induktory nebo kapacitory. To znamená, že místo, abychom optimalizovali hodnoty induktorů a kapacitorů zvlášť, budeme optimalizovat reaktance zátěží X_i . Pokud je reaktance záporná (kapacitního charakteru), použijeme kapacitor o hodnotě

$$C_i = -\frac{1}{2\pi f_0 X_i}. \quad (5.1)$$

V případě, že je reaktance kladná (induktivního charakteru), použijeme induktor o hodnotě

$$L_i = \frac{X_i}{2\pi f_0}. \quad (5.2)$$

MWS umožňuje optimalizaci těchto prvků samo o sobě. Pro každý krok iterace optimalizačního procesu je ovšem nutné provést kompletní řešení Maxwellových rovnic v časové oblasti, jehož výpočet je časově velmi náročný (zvlášť v rámci této práce, kde jsou analyzovány prostorově náročnější struktury s velkým počtem portů). Proto byl zvolen alternativní postup a to kombinací numerických řešení získaných z MWS (kdy je kompletní řešení Maxwellových rovnic v časové oblasti provedeno pouze dvakrát pro každou anténní řadu) a analytických výpočtů.

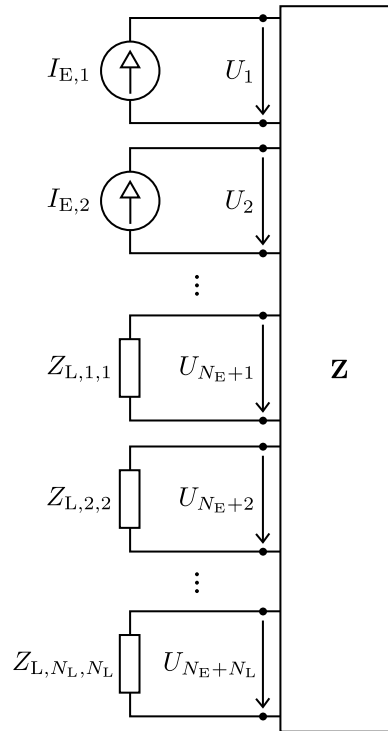
5.1 Syntéza vyzařovacího diagramu anténní řady

Předpokladem pro syntézu celkového vyzařovacího diagramu anténní řady jsou napočítané jednotlivé proudové vyzařovací diagramy aktivního a pasivních monopólů pro jednotkové budící proudy a impedanční matice (jak je popsáno v kapitole 4 a v přílohách A a B).

Pokud použijeme schéma pro anténní řadu podle obrázku 5.1, platí:

$$\vec{U} = \begin{pmatrix} \vec{U}_E \\ \vec{U}_L \end{pmatrix} = \begin{pmatrix} \mathbf{Z}_{EE} & \mathbf{Z}_{EL} \\ \mathbf{Z}_{LE} & \mathbf{Z}_{LL} \end{pmatrix} \begin{pmatrix} \vec{I}_E \\ \vec{I}_L \end{pmatrix} = \mathbf{Z}\vec{I}, \quad (5.3)$$

kde \vec{U} je vektor napětí a \vec{I} je vektor proudů na jednotlivých portech anténní



Obrázek 5.1: Schéma pro odvození rovnic proudů a napětí

řady. \mathbf{Z} je čtvercová matice impedančí, která dává proudy a napětí do souvislosti. Díky číslování portů (jak je uvedeno v kapitole 4) je možné rozdělit impedanční matici \mathbf{Z} na menší podmatice. \mathbf{Z}_{EE} je čtvercová matice o velikosti $[N_E \times N_E]$, kde N_E je počet aktivních prvků, která obsahuje vlastní a vzájemné impedance aktivních prvků (v tomto případě je pouze jeden aktivní prvek). \mathbf{Z}_{LL} je také čtvercová matice o velikosti $[N_L \times N_L]$, kde N_L je počet pasivních prvků, obsahující vlastní a vzájemné impedance pasivních prvků. \mathbf{Z}_{EL} je matice o velikosti $[N_E \times N_L]$ obsahující vzájemné impedance aktivních a pasivních prvků. \mathbf{Z}_{LE} je matice o velikosti $[N_L \times N_E]$ obsahující vzájemné

impedance aktivních a pasivních prvků a platí pro ní vztah¹

$$\mathbf{Z}_{\text{EL}} = \mathbf{Z}_{\text{EL}}^{\text{H}}. \quad (5.4)$$

Pro jeden aktivní prvek tvoří \mathbf{Z}_{EL} a \mathbf{Z}_{LE} řádkový a sloupcový vektor. Po rozepsání vztahu 5.3 dostaneme rovnice

$$\vec{U}_{\text{E}} = \mathbf{Z}_{\text{EE}}\vec{I}_{\text{E}} + \mathbf{Z}_{\text{EL}}\vec{I}_{\text{L}}, \quad (5.5)$$

$$\vec{U}_{\text{L}} = \mathbf{Z}_{\text{LE}}\vec{I}_{\text{E}} + \mathbf{Z}_{\text{LL}}\vec{I}_{\text{L}}. \quad (5.6)$$

Napětí \vec{U}_{L} a proudy \vec{I}_{L} jsou také svázány vztahem

$$\vec{U}_{\text{L}} = -\mathbf{Z}_{\text{L}}\vec{I}_{\text{L}}, \quad (5.7)$$

kde \mathbf{Z}_{L} je diagonální matice o velikosti $[N_{\text{L}} \times N_{\text{L}}]$, kde prvky na diagonále jsou impedance zátěží pasivních prvků anténní řady. Pokud zkombinujeme rovnice 5.6 a 5.7, dostaneme vztah

$$\vec{I}_{\text{L}} = -(\mathbf{Z}_{\text{L}} + \mathbf{Z}_{\text{LL}})^{-1} \mathbf{Z}_{\text{LE}}\vec{I}_{\text{E}}. \quad (5.8)$$

Proud \vec{I} pak mohou být vyjádřeny jako

$$\vec{I} = \begin{pmatrix} \mathbf{1} \\ -(\mathbf{Z}_{\text{L}} + \mathbf{Z}_{\text{LL}})^{-1} \mathbf{Z}_{\text{LE}} \end{pmatrix} \vec{I}_{\text{E}}, \quad (5.9)$$

kde $\mathbf{1}$ je matice identity o velikosti $[N_{\text{E}} \times N_{\text{E}}]$. Upravený vztah pro reaktivní zátěže pak vypadá takto:

$$\vec{I} = \begin{pmatrix} \mathbf{1} \\ -(\mathbf{j}\mathbf{X}_{\text{L}} + \mathbf{Z}_{\text{LL}})^{-1} \mathbf{Z}_{\text{LE}} \end{pmatrix} \vec{I}_{\text{E}}, \quad (5.10)$$

kde \mathbf{X}_{L} je diagonální matice reaktancí zátěží pasivních prvků.

Celkový vyzařovací diagram $f(\theta, \phi)$ lze pak zapsat jako součet vážených vyzařovacích diagramů jednotlivých portů

$$f(\theta, \phi) = \sum_{i=1}^{N_{\text{E}}+N_{\text{L}}} I_i f_{\text{p},i}(\theta, \phi), \quad (5.11)$$

kde I_i jsou prvky vektoru \vec{I} a $f_{\text{p},i}$ jsou proudové vyzařovací diagramy jednotlivých proudových portů buzených jednotkovým proudem.

5.2 Syntéza referenčního vyzařovacího diagramu

Vyzařovací diagram anténní řady se dá vyjádřit jako součin dvou částí (jak je popsáno v [14] na straně 286):

$$f(\theta, \phi) = EF \times AF, \quad (5.12)$$

¹Ze simulace tento vztah neplatí vzhledem k numerickým chybám. Rozdíly jsou ale relativně malé (v řádech setin až desetin ohmů) a tedy zanedbatelné.

kde EF (Element Factor) je vyzařovací diagram prvku (zjištěný v režimu open circuit), ze kterého je anténní řada složena (v tomto případě vyzařovací diagram monopólu), a AF (Array Factor) je koeficient anténní řady.

Pro monopól položený do osy z platí [15]:

$$EF(\theta) = \begin{cases} \frac{\cos(\frac{\pi}{2} \cos \theta)}{\sin \theta} & \text{pro } 0 \leq \theta < \frac{\pi}{2} \\ 0 & \text{pro } \frac{\pi}{2} \leq \theta \leq \pi \end{cases} \quad (5.13)$$

Koeficient kruhové anténní řady AF pak můžeme vyjádřit [16, 17] jako

$$AF(\theta, \phi) = \left| \cos \left(\frac{\phi - \phi_G}{2} \right) \right|^{q_\phi}, \quad (5.14)$$

kde ϕ_G je úhel rotace hlavního laloku kolem osy z a q_ϕ je exponent, kterým lze regulovat směrovost (šířku hlavního laloku). Zobecněním na libovolný počet laloků L a libovolnou amplitudu A dostaneme

$$AF(\theta, \phi) = A \left| \cos \left(\frac{L\phi - \phi_G}{2} \right) \right|^{q_\phi}. \quad (5.15)$$

Referenční diagram je pak dán vztahem

$$f_G(\theta, \phi) = \begin{cases} A \left| \cos \left(\frac{L\phi - \phi_G}{2} \right) \right|^{q_\phi} \cdot \frac{\cos(\frac{\pi}{2} \cos \theta)}{\sin \theta} & \text{pro } 0 \leq \theta < \frac{\pi}{2} \\ 0 & \text{pro } \frac{\pi}{2} \leq \theta \leq \pi \end{cases} \quad (5.16)$$

Ukázkové referenční diagramy jsou na obrázku 5.2. Uvedený vztah je nicméně pouze přibližný, protože nebere v úvahu zvlnění vyzařovacích diagramů jednotlivých portů vlivem sousedních prvků (jak ukazuje obrázek 5.3). Pro úlohu optimalizace v rámci této práce ale stačí.

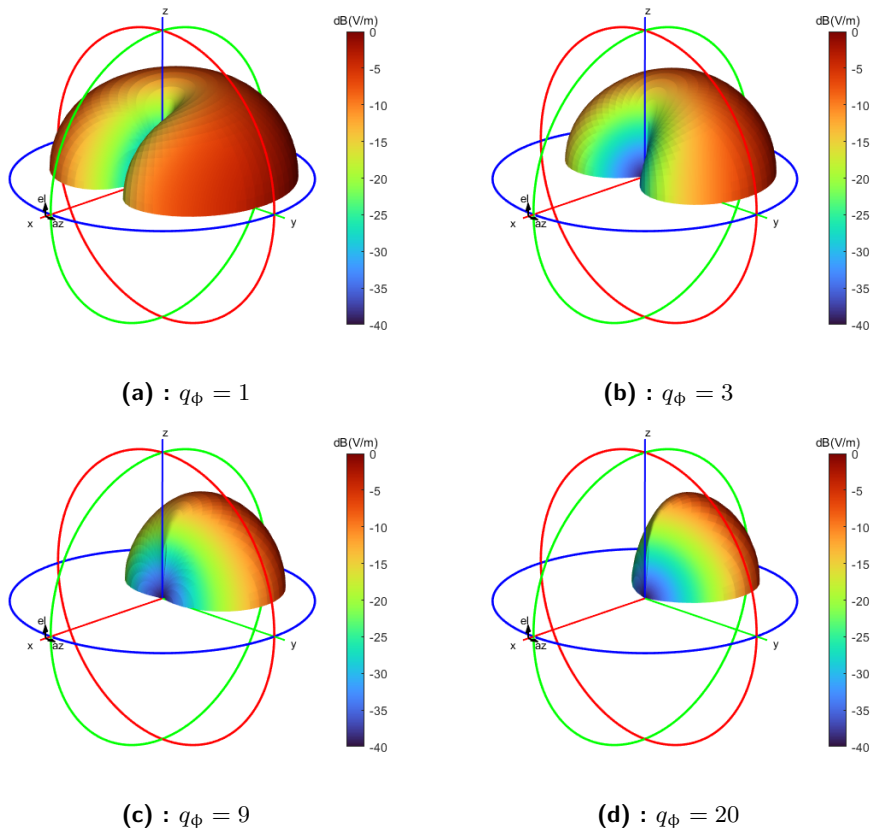
5.3 Optimalizátor reaktancí a budícího proudu

Předchozí vztahy byly implementovány v Matlabu třídou *ReactanceOptimizer* (v repozitáři ve složce *lib/Custom*). Ta se pro zadaný vyzařovací diagram pokusí nalézt optimální hodnoty reaktancí zátěží pasivních prvků a také velikost budícího proudu aktivního prvku (kvůli měřítku vyzařovacího diagramu). Pro optimalizaci využívá metodu *PSO*. Zde je ukázkový skript:

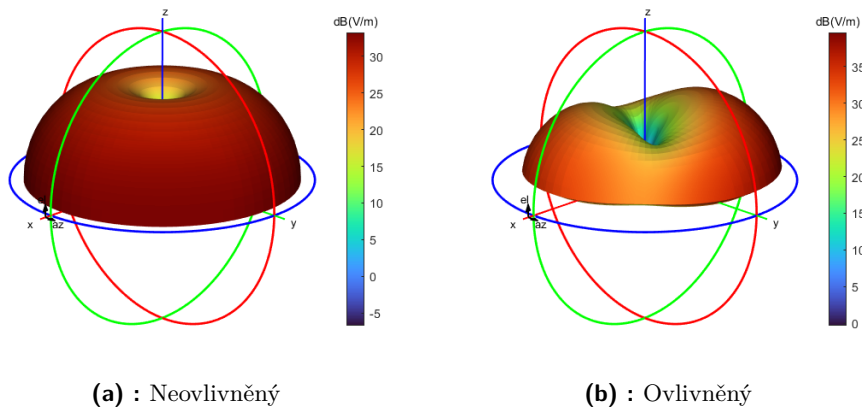
```

1 % ...
2 phi_G = 0;
3 q_phi = 15;
4 ffRef = @(theta, phi) ...
5     abs(cos((phi-phi_G)/2))^q_phi*(cos(pi/2*cos(theta))/sin(
6         theta));
7 ffopt = ReactanceOptimizer;
8 ffopt.setPortFarfields(portFF);
9 ffopt.setCoordinates(coords);

```



Obrázek 5.2: Příklady referenčních vyzářovacích diagramů pro hodnoty parametrů $\phi_G = \pi$, $L = 1$, $A = 1$ a různé q_ϕ



Obrázek 5.3: Ovlivněné a neovlivněné vyzářovací diagramy portů anténní řady

```

10 ffopt.setImpedanceMatrix(Zmat);
11 ffopt.setExcitationCurrentBounds([1 100]*1e-3);
12 ffopt.setReactanceBounds([-1.5e3 1.5e3]);
13 ffopt.setReferenceFF(ffRef);
14 ffopt.setOptimizationCriterion('sum');
15

```

```

16 tic;
17 ffopt.run();
18 toc;
19
20 optimizerOut = ffopt.getOptimizerOutput();
21 optimizerMinCost = ffopt.getMinimumCost();
22 resultFF = ffopt.getResultFF();
23 resultExcitCurr = ffopt.getResultExcitationCurrent();
24 resultReactances = ffopt.getResultReactances();

```

Proměnná *portFF* je *cell array* obsahující proudové vyzařovací diagramy jednotlivých portů, *coords* je struktura obsahující souřadnice θ a ϕ a *Zmat* je impedanční matice. Na řádce 4 je definován referenční vyzařovací diagram pomocí anonymní funkce tak, jak byl popsán dříve v této kapitole. Na řádce 7 je vytvořena instance optimalizátoru. Na řádcích 8 až 14 jsou pak nastaveny parametry optimalizátoru včetně intervalů hledaných reaktancí a budícího proudu. Optimalizační kritérium, které je nastaveno na řádce 14 může nabývat hodnot **full**, **mag** a **sum**, které odpovídají chybám Δ_1 , Δ_2 a Δ_3 z kapitoly 3 (v tomto pořadí).² Na řádce 17 je spuštěna samotná optimalizace (a je změřena doba jejího trvání). Na řádcích 20 až 24 jsou získány výstupy z optimalizátoru. Jsou to výstupy z funkce *particleswarm* (struktura obsahující například informaci o počtu iterací), nejnižší napočítaná hodnota účelové funkce (druhá mocnina chyby), výsledný optimalizovaný vyzařovací diagram, optimalizovaná hodnota budícího proudu a optimalizované hodnoty reaktancí zátěží.

Během optimalizace bylo empiricky zjištěno, že je vhodné nastavovat intervaly optimalizovaných reaktancí a budícího proudu v řádech ne větších než 10^3 . V případě, že jsou tyto intervaly větší, může skončit optimalizace s menším počtem iterací, než by bylo třeba. Je to způsobeno tím, že ukončovací podmínka funkce *particleswarm* je založena na relativní změně proměnných a ta je pro větší intervaly pro nějaký krok pochopitelně nižší než pro menší intervaly se stejným krokem.

Třída *ReactanceOptimizer* umí kromě optimalizace vyzařovacího diagramu také vygenerovat vyzařovací diagram na základě velikosti budícího proudu a reaktancí (toho je využito i během optimalizace). Ukázkový skript je následující:

```

1 Iexcit = 1;
2 X = [0.01 -0.12];
3 generatedFarfield = ffopt.generateFF(Iexcit, X);

```

Operace s vyzařovacími diagramy (vykreslování a skalární součin) jsou popsány v přílohách C a D).

²Ve skutečnosti se pro urychlení výpočtu používají druhé mocniny chyb.

5.4 Parametrický skript

Pro optimalizaci vyzařovacích diagramů byl vytvořen také parametrický skript, který se nachází v repozitáři ve složce *arrgen* pod názvem *processData.m*. Ten předpokládá, že jsou k dispozici data v podobě *.mat* souborů vygenerovaná skriptem *generateData.m* z kapitoly 4. Skript taktéž obsahuje parametr *DATA_ID*, který říká, s jakou sadou dat se pracuje. Dále obsahuje zdrojový kód obdobný tomu z předchozí sekce. Navíc umožňuje omezit vyzařovací diagram jen na určitý interval úhlu θ proměnnou *THETA_RANGE* a interval úhlu ϕ proměnnou *PHI_RANGE*.³

5.5 Výsledky optimalizace

Optimalizováno bylo celkem 36 různých geometrických konfigurací anténních řad. Řady byly tvořeny jedním nebo dvěma kruhy pasivních prvků s parametry z tabulky 5.1.

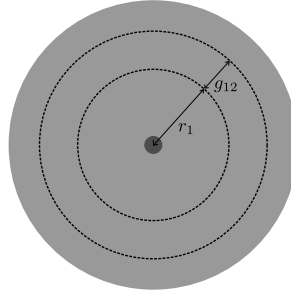
Frekvence f_0	3 GHz
Délka monopólů	$\lambda_0/4 = 25$ mm
Poloměr monopólů	0,5 mm
Okraj zemní roviny	10 mm
Tloušťka zemní roviny	0.1 mm
Mezera mezi zemní rovinou a monopóly	0.1 mm
Počet buněk blízko, dál od modelu	25, 25

Tabulka 5.1: Parametry pro vygenerování první sady anténních řad

Konfigurace byly nasimulovány pro počty pasivních prvků $N_L = 8, 12$ a 24 , poloměry vnitřního kruhu $r_1 = \lambda_0/8, \lambda_0/4$ a $\lambda_0/2$ a mezery mezi vnitřním a vnějším kruhem g_{12} o velikostech 0 (kruhy posazené do sebe), $\lambda_0/8, \lambda_0/4$ a $\lambda_0/2$. Schematické znázornění konfigurací je na obrázku 5.4. Vnější kruh byl vždy orientován tak, aby byly jeho prvky úhlově posazené mezi prvky vnitřního kruhu (tak, jak je vidět na obrázku 4.1b). Symetrie konfigurací s $8, 12$ a 24 prvky lze popsat dihedrálními symetrickými grupami D_4, D_6 a D_{12} (v tomto pořadí).

Optimalizace byla provedena pro těchto 36 konfigurací pro $q_\phi = 3$ pro různé hodnoty parametru ϕ_G referenčního vyzařovacího diagramu - a to pro hodnoty $0, \pi/4, \pi/6, \pi/8, \pi/12$ a $\pi/24$. V tabulce 5.2, jsou napočítány průměrné chyby $\overline{\Delta_3}$ a doby trvání optimalizace $\overline{T_{opt}}$ pro všechny úhly ϕ_G . Výsledky pro jednotlivé úhly jsou uvedeny v tabulkách v příloze E.

³Detailní popis použití skriptu je uveden v jeho komentářích.

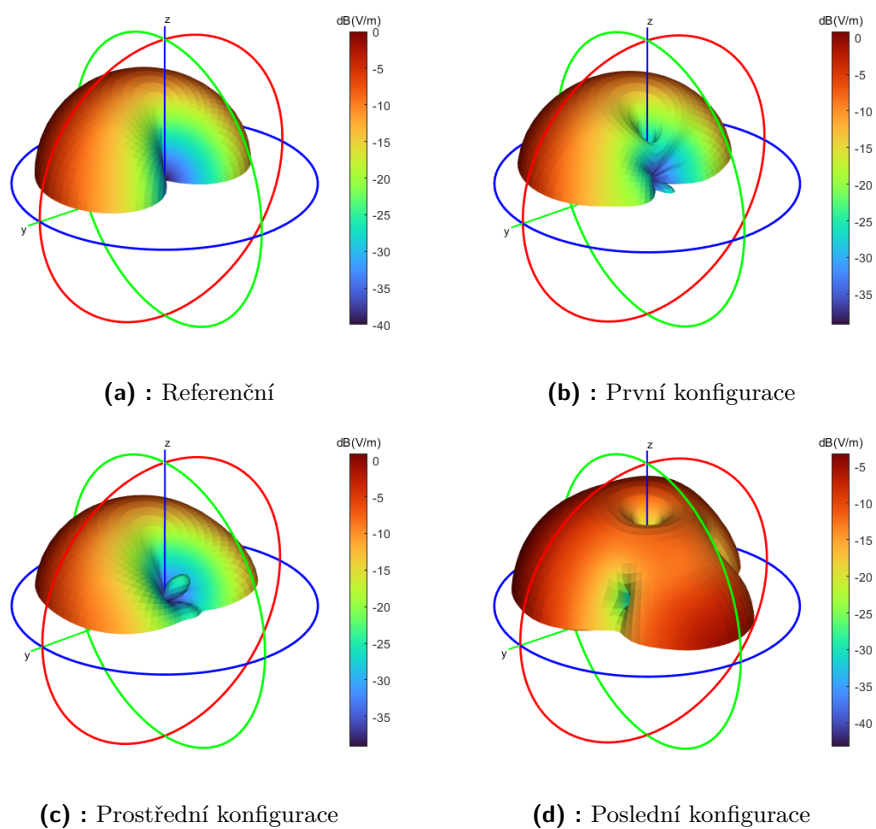


Obrázek 5.4: Schematické znázornění konfigurací pro optimalizaci

N_L	r_1	g_{12}	$\overline{\Delta_3}$	$\overline{T_{opt}}$ [s]	N_L	r_1	g_{12}	$\overline{\Delta_3}$	$\overline{T_{opt}}$ [s]
24	$\lambda/8$	$\lambda/4$	0.187	71.5	8	$\lambda/4$	0	0.216	4.0
24	$\lambda/8$	$\lambda/8$	0.191	74.3	8	$\lambda/4$	$\lambda/4$	0.216	2.9
24	$\lambda/4$	$\lambda/8$	0.192	82.3	8	$\lambda/8$	$\lambda/2$	0.218	3.6
24	$\lambda/8$	$\lambda/2$	0.193	34.5	8	$\lambda/8$	$\lambda/4$	0.222	4.9
24	$\lambda/4$	0	0.194	46.5	8	$\lambda/4$	$\lambda/8$	0.225	3.2
12	$\lambda/8$	$\lambda/2$	0.195	15.8	8	$\lambda/4$	$\lambda/2$	0.232	2.9
24	$\lambda/8$	0	0.195	48.9	24	$\lambda/2$	$\lambda/4$	0.236	30.3
8	$\lambda/8$	0	0.196	4.6	24	$\lambda/2$	$\lambda/8$	0.239	47.6
12	$\lambda/8$	$\lambda/4$	0.196	16.5	24	$\lambda/2$	$\lambda/2$	0.292	33.2
24	$\lambda/4$	$\lambda/4$	0.196	107.4	12	$\lambda/2$	$\lambda/4$	0.304	4.8
12	$\lambda/8$	0	0.196	11.6	24	$\lambda/2$	0	0.305	99.3
12	$\lambda/8$	$\lambda/8$	0.196	11.4	8	$\lambda/2$	0	0.340	2.8
24	$\lambda/4$	$\lambda/2$	0.197	60.5	12	$\lambda/2$	$\lambda/2$	0.351	4.9
8	$\lambda/8$	$\lambda/8$	0.198	4.4	12	$\lambda/2$	$\lambda/8$	0.358	5.2
12	$\lambda/4$	0	0.204	14.0	12	$\lambda/2$	0	0.364	6.6
12	$\lambda/4$	$\lambda/8$	0.206	10.5	8	$\lambda/2$	$\lambda/8$	0.385	3.0
12	$\lambda/4$	$\lambda/4$	0.209	6.0	8	$\lambda/2$	$\lambda/4$	0.461	2.4
12	$\lambda/4$	$\lambda/2$	0.210	8.8	8	$\lambda/2$	$\lambda/2$	0.476	3.0

Tabulka 5.2: Průměrné výsledky optimalizace

Z tabulky 5.2 vyplývá, že nejpřesnějších vyzářovacích diagramů bylo obecně dosaženo konfiguracemi s poloměrem r_1 menším než $\lambda/2$. Dále s 24 pasivními prvky, které měly ale dlouhé doby trvání optimalizace (v řádech desítek až stovek). Naopak nejhorších výsledků bylo obecně dosaženo konfiguracemi s 8 pasivními prvky, které měly velmi krátké doby trvání optimalizace (v řádech jednotek). Dobrým kompromisem mezi chybou a časem optimalizace je pak například posledních pět řádků tabulky vlevo a prvních šest řádků vpravo. Pro tyto konfigurace je chyba ještě přijatelná a jejich optimalizace trvá krátkou dobu. Na obrázku 5.5 je referenční vyzářovací diagram a vyzářovací diagramy (pro $\phi_G = 0$) první konfigurace (nejmenší chyba), prostřední konfigurace (poslední řádek vlevo, dobrý kompromis) a poslední konfigurace (největší chyba). Chyba byla velmi citlivá na změny úhlu ϕ_G .



Obrázek 5.5: Příklad optimalizovaných vyzářovacích diagramů a referenční vyzářovací diagram

Kapitola 6

Stabilita vyzařovacího diagramu

Tato kapitola se zabývá tím, jak změny symetrie a parametrů (reaktancí zátěží) anténní řady ovlivňují výsledný vyzařovací diagram (jak se mění chyba vůči referenčnímu vyzařovacímu diagramu).

6.1 Vliv vzájemné orientace vnitřního a vnějšího kruhu

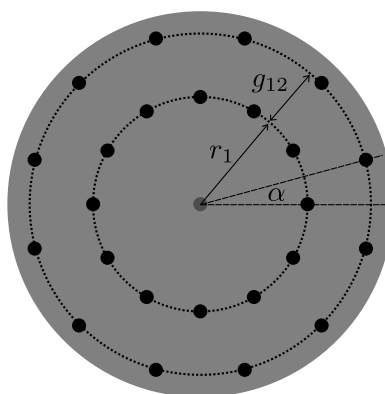
V této sekci je zkoumán vliv vzájemné rotace vnitřního a vnějšího kruhu. Bylo analyzováno celkem 9 různých orientací pro první anténní řadu ze sekce 5.5 (s nejmenší chybou; $N = 24$ pasivních prvků, $r_1 = \lambda/8$ a $g_{12} = \lambda/4$). Rotován byl vnější kruh kolem osy z a to pod úhly

$$\alpha_n = \frac{(n-1)\pi}{\frac{N}{2} \cdot (N_\alpha - 1)}; \quad n = 1, 2, \dots, N_\alpha = 9. \quad (6.1)$$

Schematické znázornění je na obrázku 6.1. Pro větší úhly se pak díky symetriím vlastnosti anténní řady periodicky opakují. Všechny tyto konfigurace lze popsat stejnou symetrickou grupou. Nejedná se však o dihedrální, jako tomu bylo v sekci 5.5 a to proto, že prvky vnějšího kruhu jsou vychýleny od středových úhlů.¹ V tomto případě se jedná o cyklickou symetrickou grupu $G = \{E, C_{12z}, C_{12z}^2, \dots, C_{12z}^{11}\}$.

Nejprve byly nalezeny optimální reaktance a velikost budícího proudu pro první konfiguraci ($\alpha = 0$ rad) a referenční vyzařovací diagram s libovolně zvolenými parametry $\phi_G = 0$ rad, $q_\Phi = 3$, $A = 130$ a $L = 1$. Výsledná velikost budícího proudu byla 7,5 A a hodnoty reaktancí z tabulky 6.1.

¹První konfigurace ($\alpha = 0$ rad), kdy jsou oba kruhy v zákrytu, a poslední konfigurace ($\alpha = \pi/12$ rad), tvoří výjimku a lze je popsat dihedrální symetrickou grupou D_{12} .

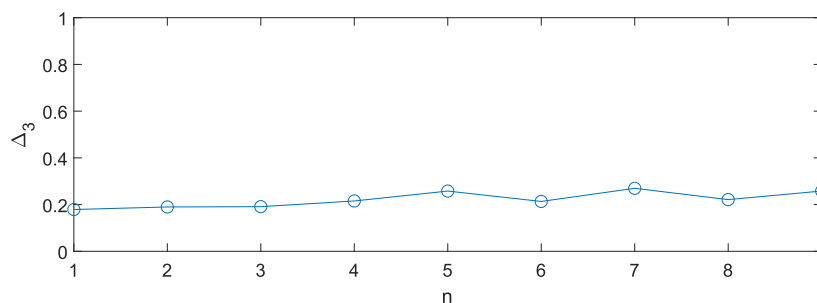


Obrázek 6.1: Příklad vygenerované anténní řady se dvěma kruhy pasivních prvků

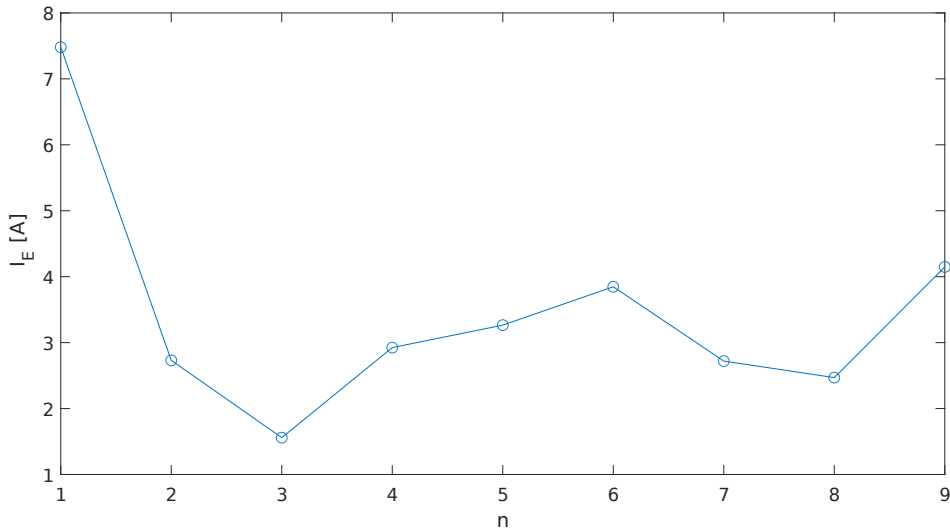
n	X_n [Ω]	n	X_n [Ω]	n	X_n [Ω]	n	X_n [Ω]
1	-50,8	13	1500,0	13	-102,4	19	-1500,0
2	-48,0	14	29,0	14	-1500,0	20	11,0
3	-69,7	15	-2,1	15	-1500,0	21	-1497,8
4	86,6	16	1500,0	16	1,7	22	-1499,1
5	71,9	17	-144,2	17	-222,4	23	-126,0
6	-1,8	18	-73,5	18	-41,5	24	-158,6

Tabulka 6.1: Nalezené optimální reaktance pro konfiguraci $\alpha = 0$ rad

Pro všechny konfigurace s napočítanými parametry pak byly vyhodnoceny hodnoty chyby Δ_3 . Závislost byla vynesena do grafu na obrázku 6.2. Z grafu je patrné, že změnou vzájemné orientace dvou kruhů nedochází k výraznému ovlivnění chyby. Ovlivněna je ale jiná veličina a tím je velikost budícího proudu. Zdá se, že rotací vnějšího kruhu je možné optimalizovat jeho velikost. Závislost je vynesena do grafu na obrázku 6.3.



Obrázek 6.2: Graf závislosti chyby Δ_3 na vzájemné orientaci dvou kruhů anténní řady



Obrázek 6.3: Graf závislosti budícího proudu I_E na vzájemné orientaci dvou kruhů anténní řady

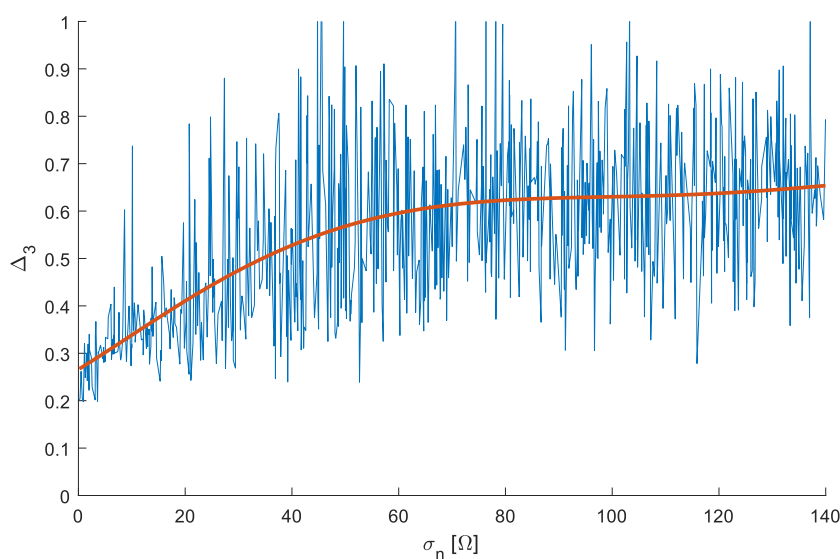
6.2 Vliv rozmítání reaktancí

V této sekci je zkoumán vliv změny reaktancí na výsledný vyzařovací diagram. Pro testování byla zvolena konfigurace z předchozí sekce pro $\alpha = \pi/12$ rad. Opět pro ní byl nalezen optimální vyzařovací diagram. Výsledná hodnota budícího proudu vyšla 2,1 A a hodnoty reaktancí z tabulky 6.2.

n	X_n [Ω]	n	X_n [Ω]	n	X_n [Ω]	n	X_n [Ω]
1	-467,7	7	1500,0	13	-1500,0	19	-1499,4
2	-1500,0	8	1498,6	14	41,6	20	-1499,7
3	-1498,8	9	58,3	15	-1500,0	21	-1500,0
4	15,3	10	1382,8	16	23,2	22	56,6
5	-28,0	11	-308,7	17	90,3	23	1497,3
6	-10,2	12	-617,0	18	-1499,2	24	55,3

Tabulka 6.2: Nalezené optimální reaktance pro konfiguraci $\alpha = \pi/12$ rad

Bylo vygenerováno 1000 náhodných vektorů odchylek reaktancí $\Delta\vec{X}_n$. Pro každý vektor $\vec{X} + \Delta\vec{X}_n$ byla napočítána střední kvadratická chyba (RMSE) reaktance σ_n a odpovídající chyba Δ_3 vůči referenčnímu vyzařovacímu diagramu. Jejich závislost byla vynesena do grafu na obrázku 6.4. Hodnoty byly proloženy polynomem šestého řádu (červená křivka). Dle očekávání pro rostoucí σ_n obecně rostla i chyba Δ_3 .

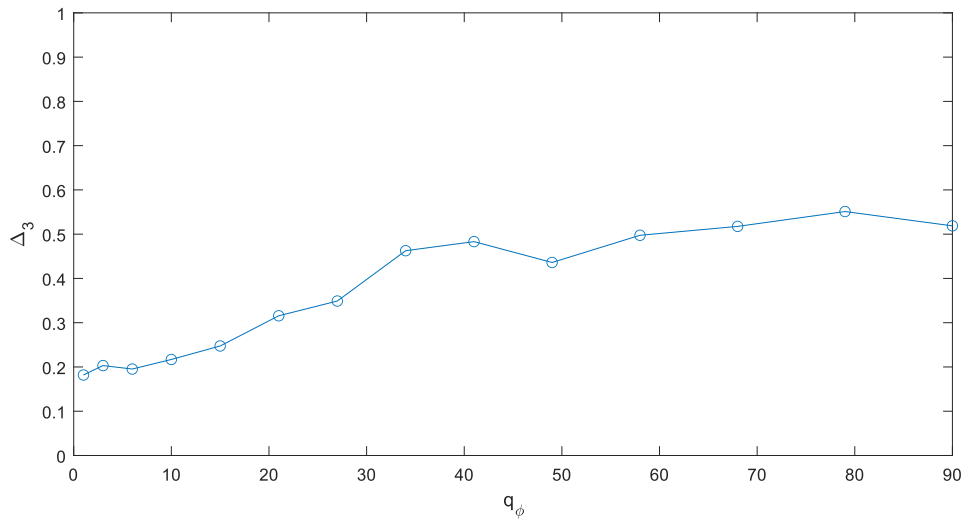


Obrázek 6.4: Graf závislosti chyby Δ_3 na střední kvadratické chybě reaktancí zátěží

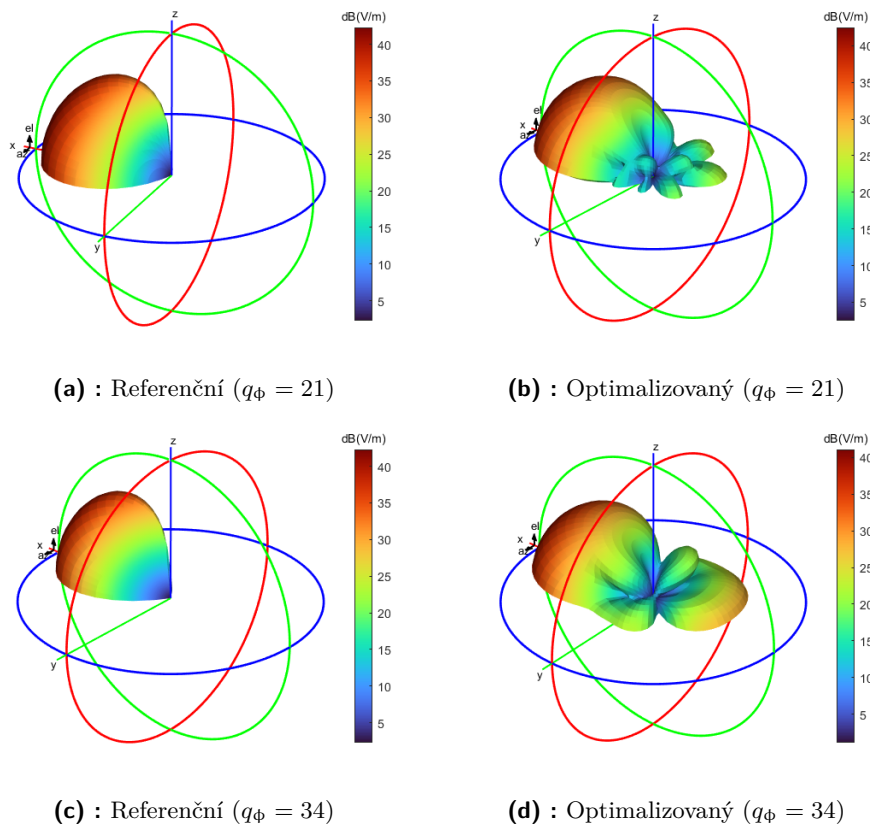
6.3 Vliv požadavků na směrovost

V této sekci je zkoumána schopnost kruhově symetrické anténní řady splnit požadavky na směrovost určenou parametrem q_ϕ referenčního vyzařovacího diagramu. K tomuto účelu byla opět vybrána konfigurace z předchozí sekce (tedy ta, která měla v průměru nejlepší výsledky při optimalizaci v sekci 5.5). Ostatní parametry referenčního vyzařovacího diagramu byly zvoleny $\phi_G = 0$ rad, $A = 130$ a $L = 1$. Pro každé q_ϕ byla spočítána chyba Δ_3 . Výsledná závislost Δ_3 na q_ϕ byla vynesena do grafu na obrázku 6.5

V grafu lze zjistit, že přibližně až do hodnoty $q_\phi = 21$ lze vyzařovací diagram považovat za přijatelný. Na obrázku 6.6 jsou pro porovnání referenční a optimalizované vyzařovací diagramy pro hodnoty $q_\phi = 21$ a $q_\phi = 34$ (sousední vypočtené hodnoty).



Obrázek 6.5: Graf závislosti chyby Δ_3 na parametru q_ϕ referenčního vyzářovacího diagramu



Obrázek 6.6: Příklady referenčních a optimalizovaných vyzářovacích diagramů na hranici přijatelnosti pro různé q_ϕ

Kapitola 7

Závěr

V kapitole 4 se podařilo propojit MWS s Matlabem a plně automatizovat generování kruhově symetrických anténních řad, jejich analýzu a export všech potřebných dat.

V kapitole 5 byly odvozeny analytické vztahy pro syntézu vyzařovacích diagramů obecných anténních řad s libovolným množstvím aktivních prvků a pasivních prvků zatížených obecnými impedancemi. Těch bylo dále využito pro optimalizaci vyzařovacích diagramů kruhově symetrických anténních řad. Byly prezentovány výsledky optimalizace v podobě tabulek a grafů zobrazujících výsledné vyzařovací diagramy a jejich parametry. Nejlepších výsledků bylo dosaženo s větším množstvím pasivních prvků, které ale byly časově náročnější na optimalizaci.

V poslední kapitole byly prozkoumány různé vlivy na stabilitu vyzařovacích diagramů. Ukázalo se, že vzájemná rotace dvou kruhů použité antény neměla na vyzařovací diagram žádný výrazný vliv, zato měla potenciál pro optimalizaci budícího proudu na aktivním prvku. Byla testována stabilita vůči rozmítání reaktancí zátěží na pasivních prvcích, která s rostoucí chybou obecně klesala. Nakonec byly ukázány limity anténní řady s ohledem na požadovanou směrovost.

V rámci této práce bylo vygenerováno celkem 90 různých MWS projektů (vždy po dvojicích pro proudové a S-parametrové porty). Výpočetní čas trval dohromady zhruba 4 celé dny. To jsou v průměru na každou dvojici přibližně 2 hodiny výpočetního času. Výpočet vyzařovacího diagramu anténní řady s jedním aktivním prvkem a 24 pasivními prvky trvá v MWS přibližně 35 sekund. Stejný výpočet pomocí analytických vztahů (uvedených v předchozích kapitolách) v Matlabu trvá přibližně 400 μ s. Jeden krok optimalizace trvá asi 120 ms. Počet kroků iterace při optimalizaci takového diagramu je přibližně 1000. Pokud bychom vyzařovací diagram optimalizovali v MWS, trvalo by to tedy přibližně 35000 sekund (téměř 10 hodin). V Matlabu by optimalizace trvala 120 s, ale s nutností napočítat proudové vyzařovací diagramy a impedanční matici (tedy +2 hodiny). Optimalizace v Matlabu je tedy pro takovou anténní řadu asi pětikrát rychlejší než v MWS (to je opravdu velmi hrubý odhad). I to je velká úspora, ale skutečná výhoda se ukáže ve chvíli, kdy chceme jednu konfiguraci optimalizovat pro různé vyzařovací diagramy. S jejich rostoucím počtem roste i úspora výpočetního času.



Literatura

- [1] CHEPALA, Anil, Vincent FUSCO a Neil BUCHANAN. Active Circular Retro-Directive Array. *IEEE Transactions on Antennas and Propagation* [online]. 2019, **67**(10), 6677-6679 [cit. 2022-05-18]. ISSN 0018-926X. Dostupné z: doi:10.1109/TAP.2019.2927628
- [2] PEITZMEIER, Nikolai a Dirk MANTEUFFEL. Upper Bounds and Design Guidelines for Realizing Uncorrelated Ports on Multimode Antennas Based on Symmetry Analysis of Characteristic Modes. *IEEE Transactions on Antennas and Propagation* [online]. 2019, **67**(6), 3902-3914 [cit. 2022-01-26]. ISSN 0018-926X. Dostupné z: doi:10.1109/TAP.2019.2905718
- [3] MCWEENY, Roy. *Symmetry: An Introduction to Group Theory and Its Applications*. Oxford, UK: Pergamon Press, 1963. ISBN 978-0-486-42182-7.
- [4] HARRINGTON, R. a J. MAUTZ. *Optimization of radar cross section of N-port loaded scatterers*. *IEEE Transactions on Antennas and Propagation* [online]. 1974, **22**(5), 697-701 [cit. 2022-05-18]. ISSN 0096-1973. Dostupné z: doi:10.1109/TAP.1974.1140862
- [5] LONSKY, Tomas, Pavel HAZDRA a Jan KRACEK. Characteristic Modes of Dipole Arrays. *IEEE Antennas and Wireless Propagation Letters* [online]. 2018, **17**(6), 998-1001 [cit. 2022-05-18]. ISSN 1536-1225. Dostupné z: doi:10.1109/LAWP.2018.2828986
- [6] MARAK, Karoly, Jan KRACEK a Sandor BILICZ. Antenna Array Pattern Synthesis Using an Iterative Method. *IEEE Transactions on Magnetics* [online]. 2020, **56**(2), 1-4 [cit. 2022-05-18]. ISSN 0018-9464. Dostupné z: doi:10.1109/TMAG.2019.2952809
- [7] HARRINGTON, R. a J. MAUTZ. *Control of radar scattering by reactive loading*. *IEEE Transactions on Antennas and Propagation* [online]. 1972, **20**(4), 446-454 [cit. 2022-05-18]. ISSN 0096-1973. Dostupné z: doi:10.1109/TAP.1972.1140234
- [8] HARRINGTON, R. a J. MAUTZ. *Pattern synthesis for loaded N-port scatterers*. *IEEE Transactions on Antennas and Propagation* [online]. 1974, **22**(2), 184-190 [cit. 2022-05-18]. ISSN 0096-1973. Dostupné z: doi:10.1109/TAP.1974.1140785

- [9] THIEL, D.V. *Impedance variations in Controlled Reactance Parasitic Antennas*. In: 2005 IEEE Antennas and Propagation Society International Symposium [online]. IEEE, 2005, s. 671-674 [cit. 2022-05-18]. ISBN 0-7803-8883-6. Dostupné z: doi:10.1109/APS.2005.1552343
- [10] HARRINGTON, R. *Reactively controlled directive arrays*. IEEE Transactions on Antennas and Propagation [online]. 1978, **26**(3), 390-395 [cit. 2022-05-18]. ISSN 0096-1973. Dostupné z: doi:10.1109/TAP.1978.1141852
- [11] YUSUF, Y. a XUN GONG. A Low-Cost Patch Antenna Phased Array With Analog Beam Steering Using Mutual Coupling and Reactive Loading. *IEEE Antennas and Wireless Propagation Letters* [online]. 2008, **7**, 81-84 [cit. 2022-05-18]. ISSN 1536-1225. Dostupné z: doi:10.1109/LAWP.2008.916689
- [12] *Particle swarm optimization - MATLAB particleswarm* [online]. [cit. 2022-05-10]. Dostupné z: <https://www.mathworks.com/help/gads/particleswarm.html>
- [13] AUTOMATIC OPTIMIZATION: CST STUDIO SUITE. *Dassault Systemes* [online]. [cit. 2022-05-11]. Dostupné z: <https://www.3ds.com/products-services/simulia/products/cst-studio-suite/optimization/>
- [14] BALANIS, Constantine A. *Antenna theory: analysis and design*. 3rd ed. Hoboken: Wiley-Interscience, 2005. ISBN 978-0-471-66782-7.
- [15] STUTZMAN, Warren L. a Gary A. THIELE. *Antenna theory and design*. 3rd ed. Hoboken: Wiley, c2013. ISBN 978-0-470-57664-9.
- [16] SAHALOS, John. *Orthogonal Methods for Array Synthesis: Theory and the ORAMA Computer Tool*. Hoboken, NJ, USA: John Wiley, 2007. ISBN 978-0-470-01741-8.
- [17] CUI, Tong, Xiongzhi ZHU, Jinling ZHANG a Zhanqi ZHENG. Circular Antenna Array Synthesis Technique Including Mutual Coupling Using Unit-excitation Active Element Pattern. In: *2018 12th International Symposium on Antennas, Propagation and EM Theory (ISAPE)* [online]. IEEE, 2018, 2018, s. 1-3 [cit. 2022-05-18]. ISBN 978-1-5386-7302-7. Dostupné z: doi:10.1109/ISAPE.2018.8634013

Příloha A

Použití knihovny *CST_App*

A.1 Vytvoření prázdného projektu

(případně otevření již existujícího):

```
1 projectDir = 'ExampleProjectDir'
2 projectFileName = 'ExampleProject.cst'
3
4 CST = CST_MicrowaveStudio(projectDir, projectFileName);
```

A.2 Nastavení frekvenčního rozsahu projektu

(v MWS editoru dostupné pod **Simulation** → **Frequency**):¹

```
1 minFreq = 2;
2 maxFreq = 4;
3
4 CST.setFreq(minFreq, maxFreq);
```

A.3 Nastavení okrajových podmínek

(dostupné pod **Simulation** → **Boundaries**):

```
1 CST.setBoundaryCondition(...
2     'xmin','open add space','xmax','open add space',...
3     'ymin','open add space','ymax','open add space',...
4     'zmin','electric','zmax','open add space')
```

Možnosti jsou *Open*, *Open add space*, *Electric*, *Magnetic*, *Periodic*, *Unit cell* a *conducting wall*. V tomto případě byla použita podmínka *electric* pro *zmin*, která realizuje nekonečnou zemní rovinu.

¹Frekvence a další veličiny, jako kapacity a indukčnosti, se nastavují v jednotkách projektu - ve výchozím stavu mm, Kelvin, GHz, s, V, A, S, Ohm, nH a pF.

A.4 Nastavení počtu buněk na vlnovou délku

(dostupné pod **Simulation** → **Global Properties** → **Cells per wavelength**):

```
1 numCellsNear = 25;
2 numCellsFar = 25;
3
4 CST.configureCells(numCellsNear, numCellsFar);
```

A.5 Přidání field monitoru

(dostupné pod **Simulation** → **Field Monitor**):

```
1 fieldType = 'farfield';
2 fieldMonitorFrequency = 3;
3
4 CST.addFieldMonitor(fieldType, fieldMonitorFrequency);
```

Možnosti jsou *Efield*, *Hfield*, *Surfacecurrent*, *Powerflow*, *Current*, *Powerloss*, *Eenergy*, *Elossdens*, *Lossdens*, *Henergy*, *Farfield*, *Temperature*, *Fieldsource*, *Spacecharge*, *ParticleCurrentDensit* a *Electrondensity*.

A.6 Přidání válce

(dostupné pod **Modeling**):

```
1 innerRadius = 0;
2 outerRadius = 4;
3 orientation = 'z'; % axis 'x', 'y' or 'z'
4 xRange = 0; % represents x position if orientation = 'z'
5 yRange = 0; % represents y position if orientation = 'z'
6 zRange = [1 100];
7 name = 'Example Cylinder';
8 componentName = 'component1';
9 material = 'PEC';
10
11 CST.addCylinder(...
12     outerRadius, innerRadius, orientation, ...
13     xRange, yRange, zRange, ...
14     name, componentName, material)
```

A.7 Přidání diskrétního portu

(dostupné pod **Simulation**):

■ A.7.1 S-parametrový port

```

1 xRange = [0 0];
2 yRange = [0 0];
3 zRange = [0 1];
4 radius = 0.1;
5 impedance = 50;
6
7 CST.addDiscretePort(xRange, yRange, zRange, radius, impedance);

```

■ A.7.2 Proudový port

```

1 xRange = [0 0];
2 yRange = [0 0];
3 zRange = [0 1];
4 radius = 0.1;
5 current = 1;
6
7 CST.addCurrentDiscretePort(...
8     xRange, yRange, zRange, radius, current);

```

■ A.8 Přidání RLC prvku

(dostupné pod **Simulation**):

```

1 name = 'Example RLC Element';
2 xRange = [0 0];
3 yRange = [0 0];
4 zRange = [0 1];
5 R = 0;
6 L = 0;
7 C = 10e-12;
8 elemType = 'Serial'; % 'Serial' or 'Parallel'
9
10 CST.addRLCLumpedElement(...
11     name, xRange, yRange, zRange, R, L, C, elemType);

```

Upozornění: u RLC prvků se zadávají veličiny v základních jednotkách (Ohm, Henry, Farad)!

■ A.9 Uložení a zavření projektu

```

1 CST.save();
2 CST.closeProject();

```

■ A.10 Další funkce

Další funkce knihovny nebyly v rámci této práce použity, nicméně je možné si je prohlédnout přímo ve zdrojovém kódu (kde jsou také popsány) v souboru *lib/CST_App/CST_MicrowaveStudio.m* v repozitáři.

Příloha B

Použití knihovny *TCSTInterface*

B.1 Otevření projektu

```
1 projectDir = 'ExampleProjectDir'
2 projectFileName = 'ExampleProject.cst'
3 projectPath = fullfile(projectDir, projectFileName);
4
5 CST = TCSTInterface();
6 CST.OpenProject(projectPath);
```

B.2 Spuštění řešiče

(v MWS editoru dostupné pod **Home** → **Start Simulation**):¹

```
1 CST.Solve();
```

B.3 Získání S-parametrů

(v editoru jsou k dispozici jako grafy):

```
1 runID = 0;
2 [S, Freq, Zref, InfoS] = CST.GetSParams(runID, [], [], [], []);
```

Do proměnné S jsou uloženy matice S-parametrů pro všechny frekvence v rozsahu frekvencí projektu tvořící trojrozměrný tenzor. Získání matice S-parametrů pro první frekvenci se tedy získá takto:

```
1 S_f1 = S(:, :, 1);
```

Potom například parametr S_{13} se získá takto:

¹Ve výchozím stavu je vybrán řešič v časové doméně. Momentálně knihovna neumožňuje měnit typ řešiče v programu, musí být změněn manuálně.

```
1 S_13 = S_f1(1,3);
```

Proměnná *Freq* je sloupcový vektor frekvencí, pro které se S-parametry napočítaly. Frekvence jsou v základních jednotkách (Hertz)! Proměnná *Zref* obsahuje referenční impedanci projektu (ve výchozím stavu 50Ω). V proměnné *InfoS* jsou uloženy informace z MWS popisující strukturu proměnné *S*. Pro konverzi S-parametrů na impedance byla vytvořena funkce *convertScatToImp*:

```
1 Z = convertScatToImp(S, Zref);
2 Z_f1 = Z(:, :, 1);
3 Z_13 = Z_f1(1,3);
```

B.4 Export vyzařovacích diagramů

```
1 [Field, InfoF] = CST.ExportFarField();
```

Proměnná *Field* je struktura obsahující vyzařovací diagramy, frekvenci, pro kterou byly počítány, matice souřadnic θ a ϕ a další dodatečné informace o formátu těchto dat. Takto se získají matice souřadnic θ a ϕ :

```
1 thetaMat = Field.THETA;
2 phiMat = Field.PHI;
```

Matice jsou ale redundantní, protože jsou vždy v jednom směru hodnoty neměnné, takto se získají pouze vektory:

```
1 thetaVec = thetaMat(:,1);
2 phiVec = phiMat(1,:);
```

Pro získání jednotlivých komponent \mathbf{E}_θ a \mathbf{E}_ϕ byla vytvořena funkce *getE-Components*:

```
1 portNumber = 1;
2 frequencyIndex = 1;
3
4 E_comps = getEComponents(Field, portNumber, frequencyIndex);
5 E_theta = E_comps(:, :, 1);
6 E_phi = E_comps(:, :, 2);
```

Matice *E_theta* a *E_phi* obsahují hodnoty daných složek pro úhly θ a ϕ (v tomto pořadí). Hodnoty pro $\theta = 0^\circ$ a $\phi = 0^\circ$ jsou duplikované na posledních indexech (tedy $\theta = 180^\circ$ a $\phi = 360^\circ$). Pro jejich odstranění (například pro korektní výpočet integrálu) je potřeba provést následující operaci:

```
1 E_theta = E_theta(1:end-1, 1:end-1);
2 E_phi = E_phi(1:end-1, 1:end-1);
```


■ B.5 Uložení a zavření projektu

```
1 CST.CloseProject([], true);
```

Prvním parametrem lze specifikovat cestu projektu, který se má uložit, nebo lze uložit současný otevřený projekt vyplněním `[]`. Druhý parametr říká, zda se má projekt před zavřením uložit nebo ne. Prostým voláním

```
1 CST.CloseProject();
```

se zavře současný otevřený projekt bez uložení.

Příloha C

Vykreslení vyzařovacích diagramů

V příloze B bylo popsáno, jak vyexportovat vyzařovací diagramy z MWS. Vektory obsahující souřadnice θ a ϕ byly uloženy do proměnných *thetaVec* a *phiVec*. Jednotlivé složky E_θ a E_ϕ byly uloženy do proměnných *E_theta* a *E_phi*. Pro jejich vykreslení byly v rámci této práce vytvořeny dvě funkce - *plotRadiationPatternYUp* a *plotRadiationPatternZUp*. Obě vykreslují stejným způsobem, ale u té první lze vyzařovací diagram rotovat kolem osy *y*, u té druhé kolem osy *z*. Protože se často modelují antény tak, aby vyzařovaly ve směru osy *z*, bude zde demonstrováno použití té druhé funkce.

Vykreslení složky E_θ samostatně provede tento kód:

```
1 figure;
2 plotRadiationPatternZUp(E_theta, thetaVec, phiVec);
3 colormap Turbo;
```

Takto proběhne vykreslení vizuálně velmi podobné tomu v MWS. Nicméně nyní je vyzařovací charakteristika vykreslena v lineárním měřítku. Pro praktické účely je vhodné nadefinovat funkci, která vyzařovací diagram převede do logaritmického měřítka:

```
1 function f_dB = getFieldIndB(f, dynRange)
2     f_dB = 20*log10(f);
3     threshold = max(max(f_dB))-dynRange;
4     f_dB(f_dB < threshold) = threshold;
5 end
```

Tato funkce navíc omezí dynamický rozsah na zadanou hodnotu v decibelech. Vykreslení (s omezením například na rozsah 40 dB) pak proběhne následovně:

```
1 figure;
2 plotRadiationPatternZUp(...
3     getFieldIndB(E_theta, 40), thetaVec, phiVec);
4 colormap Turbo;
```

Pro zobrazení jednotek lze přidat tento řádek:

```
1 title(colorbar, 'dB(V/m)');
```

Nakonec pro vykreslení celkového vyzařovacího diagramu sloučíme složky E_{θ} a E_{ϕ} :

```
1 figure;
2 E_tot = sqrt(abs(E_theta).^2 + abs(E_phi).^2);
3 plotRadiationPatternZUp(...
4     getFieldIndB(E_tot, 40), thetaVec, phiVec);
5 colormap Turbo;
```

Pokud máme složky v jednom tenzoru (proměnná E_{comps} z přílohy B), pak lze celkový vyzařovací diagram také získat sumou:

```
1 E_tot = sqrt(sum(abs(E_comps).^2, 3));
```

Příloha D

Výpočet skalárního součinu vyzařovacích diagramů

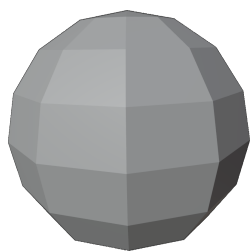
Pro potřeby optimalizace bylo nutné implementovat funkci pro výpočet skalárního součinu vyzařovacích diagramů. Byly implementovány dvě funkce - *getInnerProductFF* a *getInnerProductFFComp*. Liší se pouze tím, že ta první vyžaduje všechny složky vyzařovacího diagramu (\mathbf{E}_θ a \mathbf{E}_ϕ , to je proměnná *E_comps* z přílohy B) a spočítá celkový skalární součin i součiny po složkách, ta druhá pracuje pouze s jednou složkou (proměnná *E_theta* nebo *E_phi* z přílohy B). Obě funkce se nacházejí v repozitáři ve složce *lib/Custom*.

Funkce *getInnerProductFF* je definována takto:

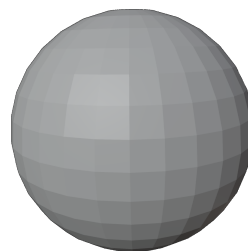
```
1 function [innerProductTotal,innerProductTheta,innerProductPhi]=
   getInnerProductFF(ff1, ff2)
2     % ff1, ff2 ... farfield(nTheta,nPhi,2)
3     % third dimension contains E components: E_theta, E_phi
4
5     if size(ff1,1) ~= size(ff2,1) || size(ff1,2) ~= size(ff2, 2)
6         error('Farfields size mismatch!');
7     end
8
9     nTheta = size(ff1,1);
10    nPhi = size(ff1,2);
11    dTheta = pi/(nTheta-1);
12    dPhi = pi/(nPhi-1);
13
14    thetaVec = sin((0:nTheta-1)*dTheta);
15    thetaGrid = repmat(thetaVec', 1, nPhi);
16    dS = 2*thetaGrid*sin(dTheta/2)*dPhi;
17    % dS = sin(thetaGrid)*dTheta*dPhi;
18
19    integrandTheta = conj(ff1(1:end-1,1:end-1,1)).*ff2(1:end
-1,1:end-1,1).*dS(1:end-1,1:end-1);
20    integrandPhi = conj(ff1(1:end-1,1:end-1,2)).*ff2(1:end-1,1:
end-1,2).*dS(1:end-1,1:end-1);
21
22    integralTheta = sum(sum(integrandTheta));
```

```
23     integralPhi = sum(sum(integrandPhi));
24
25     innerProductTheta = integralTheta;
26     innerProductPhi = integralPhi;
27     innerProductTotal = integralTheta+integralPhi;
28 end
```

kde argumenty $ff1$ a $ff2$ svojí strukturou odpovídají E_comps . Na řádcích 9 až 12 se spočítají kroky v úhlech θ a ϕ . Pro potřeby výpočtu plošky dS je napočítána matice úhlů θ odpovídajících jednotlivým bodům vyzařovacích diagramů. Na řádku 17 je zakomentovaný výpočet, který by odpovídal ideální plošce na kouli. Protože je však integrál počítán na modelu koule, který je složen z polygonů, jednotlivé plošky tvoří trochu odlišný povrch. To je ukázáno na obrázku D.1. Pro jemnější rozlišení je rozdíl méně patrný. Tento rozdíl je do jisté míry kompenzován alternativním výpočtem na řádku 16. Nakonec jsou na řádcích 19 a 20 spočítány integrandy. Na řádcích 22 a 23 jsou pak integrály nahrazeny sumami.



(a) : S hrubým rozlišením



(b) : S jemným rozlišením

Obrázek D.1: Modely koule

Příloha E

Výsledky optimalizace - tabulky

N_L	r_1	g_{12}	Δ_3	T_{opt} [s]	N_L	r_1	g_{12}	Δ_3	T_{opt} [s]
12	$\lambda/8$	$\lambda/2$	0,183	15,2	12	$\lambda/8$	$\lambda/4$	0,182	34,5
12	$\lambda/8$	$\lambda/4$	0,183	15,9	12	$\lambda/8$	$\lambda/2$	0,185	8,5
12	$\lambda/4$	$\lambda/8$	0,185	12,0	24	$\lambda/8$	$\lambda/4$	0,187	58,5
24	$\lambda/8$	$\lambda/4$	0,185	116,4	8	$\lambda/4$	$\lambda/8$	0,188	2,6
12	$\lambda/4$	$\lambda/2$	0,187	11,4	24	$\lambda/4$	$\lambda/8$	0,189	53,7
24	$\lambda/4$	$\lambda/8$	0,187	87,3	12	$\lambda/4$	0	0,190	22,9
12	$\lambda/8$	$\lambda/8$	0,187	10,7	24	$\lambda/4$	0	0,191	39,7
24	$\lambda/4$	$\lambda/4$	0,188	109,2	24	$\lambda/8$	$\lambda/8$	0,194	70,8
12	$\lambda/4$	0	0,190	6,6	12	$\lambda/4$	$\lambda/4$	0,194	6,9
24	$\lambda/4$	0	0,190	44,1	12	$\lambda/8$	$\lambda/8$	0,195	29,2
8	$\lambda/4$	$\lambda/2$	0,194	4,4	8	$\lambda/8$	0	0,195	1,4
24	$\lambda/8$	0	0,194	63,7	24	$\lambda/4$	$\lambda/2$	0,196	59,5
24	$\lambda/8$	$\lambda/2$	0,195	41,4	8	$\lambda/8$	$\lambda/8$	0,196	3,9
24	$\lambda/8$	$\lambda/8$	0,195	30,3	24	$\lambda/8$	0	0,196	33,0
8	$\lambda/8$	0	0,196	2,2	12	$\lambda/8$	0	0,200	11,9
12	$\lambda/8$	0	0,196	8,5	8	$\lambda/4$	0	0,201	3,6
8	$\lambda/8$	$\lambda/8$	0,200	4,5	12	$\lambda/4$	$\lambda/2$	0,203	11,2
8	$\lambda/4$	0	0,201	3,2	8	$\lambda/4$	$\lambda/2$	0,205	1,7
24	$\lambda/4$	$\lambda/2$	0,201	72,7	24	$\lambda/4$	$\lambda/4$	0,214	32,4
12	$\lambda/4$	$\lambda/4$	0,202	5,7	8	$\lambda/8$	$\lambda/2$	0,214	3,3
8	$\lambda/4$	$\lambda/4$	0,215	2,5	12	$\lambda/4$	$\lambda/8$	0,215	13,0
8	$\lambda/4$	$\lambda/8$	0,228	2,1	8	$\lambda/4$	$\lambda/4$	0,216	2,4
24	$\lambda/2$	$\lambda/4$	0,231	33,0	24	$\lambda/2$	$\lambda/4$	0,226	26,9
8	$\lambda/8$	$\lambda/4$	0,236	2,5	24	$\lambda/8$	$\lambda/2$	0,227	21,3
8	$\lambda/8$	$\lambda/2$	0,243	4,2	8	$\lambda/8$	$\lambda/4$	0,233	4,1
24	$\lambda/2$	$\lambda/8$	0,246	23,8	24	$\lambda/2$	$\lambda/8$	0,256	30,4
24	$\lambda/2$	$\lambda/2$	0,275	61,3	24	$\lambda/2$	$\lambda/2$	0,284	40,1
24	$\lambda/2$	0	0,298	53,8	24	$\lambda/2$	0	0,300	66,9
8	$\lambda/2$	0	0,300	2,9	12	$\lambda/2$	$\lambda/4$	0,301	5,5
12	$\lambda/2$	$\lambda/4$	0,317	4,8	12	$\lambda/2$	$\lambda/8$	0,327	4,5
12	$\lambda/2$	0	0,320	10,9	12	$\lambda/2$	$\lambda/2$	0,355	6,4
12	$\lambda/2$	$\lambda/2$	0,336	3,9	8	$\lambda/2$	$\lambda/8$	0,355	3,4
12	$\lambda/2$	$\lambda/8$	0,370	3,5	12	$\lambda/2$	0	0,385	3,9
8	$\lambda/2$	$\lambda/8$	0,403	2,6	8	$\lambda/2$	0	0,385	2,2
8	$\lambda/2$	$\lambda/4$	0,455	2,5	8	$\lambda/2$	$\lambda/2$	0,467	2,3
8	$\lambda/2$	$\lambda/2$	0,472	2,6	8	$\lambda/2$	$\lambda/4$	0,491	1,9

(a) : $\phi_G = 0$

(b) : $\phi_G = \pi/4$

Tabulka E.1: Výsledky optimalizace (část 1/3)

N_L	r_1	g_{12}	Δ_3	T_{opt} [s]	N_L	r_1	g_{12}	Δ_3	T_{opt} [s]
24	$\lambda/4$	$\lambda/2$	0,185	24,5	12	$\lambda/8$	$\lambda/4$	0,181	18,0
24	$\lambda/4$	$\lambda/4$	0,185	156,2	24	$\lambda/4$	$\lambda/2$	0,186	76,7
24	$\lambda/4$	$\lambda/8$	0,186	97,0	24	$\lambda/8$	$\lambda/4$	0,187	34,9
24	$\lambda/8$	$\lambda/8$	0,187	31,3	24	$\lambda/8$	$\lambda/8$	0,187	152,4
12	$\lambda/4$	$\lambda/4$	0,188	5,1	24	$\lambda/8$	$\lambda/2$	0,189	42,5
24	$\lambda/8$	$\lambda/2$	0,189	40,6	24	$\lambda/4$	$\lambda/4$	0,191	92,4
12	$\lambda/8$	$\lambda/2$	0,192	10,6	12	$\lambda/4$	0	0,191	10,2
24	$\lambda/8$	$\lambda/4$	0,192	43,8	8	$\lambda/8$	$\lambda/8$	0,192	7,1
8	$\lambda/8$	0	0,195	9,8	24	$\lambda/4$	0	0,192	60,9
12	$\lambda/8$	0	0,196	10,5	12	$\lambda/4$	$\lambda/8$	0,194	6,5
24	$\lambda/2$	$\lambda/8$	0,197	121,4	12	$\lambda/8$	0	0,196	10,8
24	$\lambda/8$	0	0,197	8,7	24	$\lambda/8$	0	0,197	12,6
8	$\lambda/8$	$\lambda/8$	0,198	3,6	8	$\lambda/8$	0	0,198	2,1
12	$\lambda/4$	0	0,199	26,0	8	$\lambda/4$	$\lambda/4$	0,199	5,6
24	$\lambda/4$	0	0,200	48,4	12	$\lambda/4$	$\lambda/2$	0,201	7,5
12	$\lambda/8$	$\lambda/8$	0,201	9,2	8	$\lambda/4$	0	0,201	5,0
12	$\lambda/8$	$\lambda/4$	0,206	12,3	8	$\lambda/8$	$\lambda/2$	0,201	4,4
8	$\lambda/4$	$\lambda/4$	0,216	2,5	8	$\lambda/4$	$\lambda/2$	0,202	3,7
12	$\lambda/4$	$\lambda/2$	0,217	7,9	12	$\lambda/8$	$\lambda/8$	0,203	8,1
24	$\lambda/2$	$\lambda/4$	0,217	36,8	24	$\lambda/4$	$\lambda/8$	0,206	104,0
8	$\lambda/4$	$\lambda/2$	0,219	1,9	8	$\lambda/8$	$\lambda/4$	0,208	4,2
8	$\lambda/8$	$\lambda/4$	0,224	10,7	12	$\lambda/8$	$\lambda/2$	0,227	12,3
8	$\lambda/8$	$\lambda/2$	0,234	3,4	24	$\lambda/2$	$\lambda/8$	0,240	27,8
8	$\lambda/4$	$\lambda/8$	0,270	2,6	8	$\lambda/4$	$\lambda/8$	0,268	3,7
12	$\lambda/4$	$\lambda/8$	0,276	4,1	12	$\lambda/4$	$\lambda/4$	0,272	4,8
12	$\lambda/2$	$\lambda/4$	0,281	4,0	24	$\lambda/2$	$\lambda/4$	0,278	28,1
24	$\lambda/2$	0	0,292	223,0	24	$\lambda/2$	$\lambda/2$	0,290	15,5
8	$\lambda/4$	0	0,293	1,8	24	$\lambda/2$	0	0,291	144,4
8	$\lambda/2$	0	0,302	2,8	12	$\lambda/2$	$\lambda/4$	0,320	4,8
24	$\lambda/2$	$\lambda/2$	0,309	36,9	12	$\lambda/2$	0	0,341	6,0
12	$\lambda/2$	0	0,324	4,6	12	$\lambda/2$	$\lambda/8$	0,342	6,0
12	$\lambda/2$	$\lambda/2$	0,350	4,9	8	$\lambda/2$	0	0,369	3,3
12	$\lambda/2$	$\lambda/8$	0,357	7,9	12	$\lambda/2$	$\lambda/2$	0,381	3,6
8	$\lambda/2$	$\lambda/8$	0,368	4,2	8	$\lambda/2$	$\lambda/8$	0,383	2,4
8	$\lambda/2$	$\lambda/4$	0,439	2,7	8	$\lambda/2$	$\lambda/4$	0,442	2,5
8	$\lambda/2$	$\lambda/2$	0,463	4,4	8	$\lambda/2$	$\lambda/2$	0,473	3,6

(a) : $\phi_G = \pi/6$ (b) : $\phi_G = \pi/8$

Tabulka E.3: Výsledky optimalizace (část 2/3)

N_L	r_1	g_{12}	Δ_3	T_{opt} [s]	N_L	r_1	g_{12}	Δ_3	T_{opt} [s]
24	$\lambda/8$	$\lambda/2$	0,178	41,4	24	$\lambda/4$	$\lambda/4$	0,182	129,8
12	$\lambda/4$	$\lambda/8$	0,183	14,9	24	$\lambda/8$	$\lambda/2$	0,183	20,0
24	$\lambda/8$	$\lambda/4$	0,188	79,1	24	$\lambda/8$	$\lambda/4$	0,185	96,5
24	$\lambda/4$	0	0,189	51,5	12	$\lambda/4$	$\lambda/8$	0,185	12,8
24	$\lambda/4$	$\lambda/8$	0,189	100,0	12	$\lambda/4$	$\lambda/2$	0,187	11,9
24	$\lambda/8$	$\lambda/8$	0,191	69,7	12	$\lambda/8$	$\lambda/2$	0,187	38,3
12	$\lambda/8$	$\lambda/8$	0,193	5,5	8	$\lambda/8$	$\lambda/2$	0,191	4,0
24	$\lambda/4$	$\lambda/2$	0,193	75,6	24	$\lambda/8$	$\lambda/8$	0,194	91,2
8	$\lambda/8$	$\lambda/4$	0,193	4,2	24	$\lambda/8$	0	0,194	68,3
12	$\lambda/8$	$\lambda/2$	0,194	10,0	8	$\lambda/4$	$\lambda/8$	0,194	5,0
12	$\lambda/8$	0	0,194	19,1	24	$\lambda/4$	$\lambda/8$	0,195	52,1
24	$\lambda/8$	0	0,194	107,0	12	$\lambda/8$	0	0,196	8,8
8	$\lambda/8$	0	0,195	8,4	8	$\lambda/8$	0	0,196	3,9
12	$\lambda/4$	$\lambda/4$	0,198	6,7	24	$\lambda/4$	0	0,198	34,1
8	$\lambda/8$	$\lambda/8$	0,200	3,3	12	$\lambda/8$	$\lambda/8$	0,199	5,8
8	$\lambda/4$	$\lambda/8$	0,202	3,0	8	$\lambda/4$	0	0,199	6,0
8	$\lambda/4$	0	0,202	4,5	8	$\lambda/8$	$\lambda/8$	0,201	3,9
8	$\lambda/4$	$\lambda/4$	0,211	2,3	12	$\lambda/4$	$\lambda/4$	0,201	6,8
24	$\lambda/4$	$\lambda/4$	0,216	124,5	12	$\lambda/8$	$\lambda/4$	0,206	9,6
12	$\lambda/8$	$\lambda/4$	0,217	9,0	12	$\lambda/4$	0	0,212	11,8
8	$\lambda/8$	$\lambda/2$	0,222	2,4	24	$\lambda/2$	$\lambda/8$	0,221	52,2
8	$\lambda/4$	$\lambda/2$	0,224	3,2	24	$\lambda/4$	$\lambda/2$	0,222	54,0
24	$\lambda/2$	$\lambda/4$	0,224	27,8	8	$\lambda/8$	$\lambda/4$	0,235	3,3
12	$\lambda/4$	0	0,240	6,8	8	$\lambda/4$	$\lambda/4$	0,241	2,1
12	$\lambda/4$	$\lambda/2$	0,267	2,8	24	$\lambda/2$	$\lambda/4$	0,243	29,2
24	$\lambda/2$	$\lambda/8$	0,273	29,7	24	$\lambda/2$	$\lambda/2$	0,283	15,6
12	$\lambda/2$	$\lambda/4$	0,286	5,6	12	$\lambda/2$	$\lambda/4$	0,318	4,1
24	$\lambda/2$	$\lambda/2$	0,311	29,9	12	$\lambda/2$	0	0,332	9,3
24	$\lambda/2$	0	0,317	56,3	24	$\lambda/2$	0	0,335	51,7
8	$\lambda/2$	0	0,321	3,6	8	$\lambda/4$	$\lambda/2$	0,348	2,3
12	$\lambda/2$	$\lambda/2$	0,333	4,4	12	$\lambda/2$	$\lambda/2$	0,348	6,5
12	$\lambda/2$	$\lambda/8$	0,370	5,5	8	$\lambda/2$	0	0,361	2,1
8	$\lambda/2$	$\lambda/8$	0,394	2,5	12	$\lambda/2$	$\lambda/8$	0,382	3,8
8	$\lambda/2$	$\lambda/4$	0,463	2,7	8	$\lambda/2$	$\lambda/8$	0,407	2,9
12	$\lambda/2$	0	0,480	4,6	8	$\lambda/2$	$\lambda/4$	0,477	2,3
8	$\lambda/2$	$\lambda/2$	0,498	2,1	8	$\lambda/2$	$\lambda/2$	0,486	2,9

(a) : $\phi_G = \pi/12$ (b) : $\phi_G = \pi/24$

Tabulka E.5: Výsledky optimalizace (část 3/3)