NAZARBAYEV UNIVERSITY

School of Engineering and Digital Sciences

Bachelor of Engineering in
Mechanical and Aerospace Engineering

Determining the shape of arbitrary-shaped nanoelements from EM wave scattering

(Final Capstone Project Report)

by

Meruyert Toleuova, Nurtas Kaliyev

Lead Supervisor: Prof. Konstantinos Kostas

Co-Supervisor: Prof. Constantinos Valagiannopoulos

April, 2025

Declaration

We, Meruyert Toleuova, Nurtas Kaliyev, hereby declare that this report, entitled "Determining the shape of arbitrary-shaped nanoelements from EM wave scattering" is the result of our own project work except for quotations and citations which have been duly acknowledged. We also declare that it has not been previously or concurrently submitted for any other degree at Nazarbayev University or elsewhere.

Signature:

Name: Meruyert Toleuova
Date: April 29, 2025

Signature:

Name: Nurtas Kaliyev
Date: April 29, 2025

Acknowledgments

Both authors would like to acknowledge the Nazarbayev University Collaborative Research Project (CRP): “Waves Interaction with CArbon Nanotubes - WICAN”, Grant Award Nr. 20122022CRP1607, which funded their capstone project work.

Abstract

Precise determination of nanoscale structures is a key issue in nanotechnology and materials science. Conventional approaches to nano-element characterization usually rely on simplifying assumptions and idealized geometries, whose practical applicability is seriously restricted when dealing with real, arbitrarily shaped nano-objects. In this project, the inverse, generally ill-posed, problem of shape determination of arbitrary nano-elements by scattered electromagnetic waves is considered. A computationally efficient framework was developed using scattering data of appropriate resolution, which will enable the identification and accurate reconstruction of nano-element shapes.

We begin by formulating the relevant scattering problem in an isogeometric-analysis-based boundary element method setting and subsequently employ hybrid optimization techniques to solve the inverse scattering problem. Regularization methods have been used to alleviate the inherent ill-posedness, while global and local optimization algorithms have been incorporated for appropriately addressing the shape reconstruction problem in different stages. The numerical approach is verified against both simple and complex shapes with known and unknown geometrical characteristics, which demonstrates the capacity of the implemented approach to accurately identify nanoscale geometries in a widely range of settings.

The achieved results show that the developed method facilitates the identification and reconstruction of simple or complex shapes remarkably without sacrificing accuracy or computational efficiency. It is also demonstrated that the developed methodology holds the potential to be extended to handle different kinds of nanoscale elements as well as significantly larger structures. This fact permits its extension to object/signal identification for military applications, non-intrusive inspection, remote sensing, and many more. Such examples of shape identification with a limited range of the scattering signal are included in this work to demonstrate the applicability of the developed approach to inverse problems found in real-life applications.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Electromagnetic (EM) waves are used to transport energy and information between different places with frequencies, ranging from radio and microwave to the optical band to X-rays [3]. Part of this energy is scattered; the part that is reradiated in directions different from the incident path. The scattered field operates as a reservoir of information about a target as Maxwell's equations reveal its size details, as well as its shape information and material characteristics. A variety of technologies in optics, telecommunications, radar, and remote sensing find their foundation in EM-wave scattering as it grants access to probe or image invisible objects. Given an observed scattered field, solving the inverse scattering problem to determine an object's characteristics is both important and difficult to achieve. In this work, we are focusing on the scattering phenomenon exhibited by arbitrary-shaped nanotubes (including Carbon Nanotubes - CNTs or others) when they are illuminated by plane or cylindrical waves. Specifically, we are targeting the inverse problem which aims to determine the shape, or other properties, of the interacting object(s) by analyzing the scattered signal. In contrast to the forward problem, i.e., estimating scattering for a given illuminated object, the inverse problem assumes that scattering is given and varying properties of the interacting object or EM wave field are to be determined. Inverse problems are generally ill-posed in the sense that they may not satisfy one or more of the Hadamard criteria:

- Having a solution (existence);
- Having a unique solution (uniqueness);
- Having a solution that depends continuously on the parameters or input data (smoothness/stability).

This inherent problem in inverse design further complicates the problem of determining the shape from scattered EM wave data [4]. As a common problem in inverse design with scattering, we may mention here, as an indicative example, the abrupt shape changes that can occur by a relatively small change in the scattered signal, which demonstrates the high sensitivity and poor robustness of the problem when not properly formulated. Thus, a general treatise of the problem requires a solid theoretical background in physics and mathematics, along with advanced and heterogeneous numerical techniques to tackle the problems at hand. A large body of the relevant literature relies on idealized geometries – such as spheres or cylinders – which are amenable to analytical solutions. However,

as one may easily understand, such approaches cannot easily address real-life problems where the configuration intricacies and shapes tend to be arbitrarily and highly varied. It thus necessary to adapt a more general approach that can handle numerically shapes and configurations that can be found in practical applications.

The Boundary Element Method (BEM) can be used to solve Maxwell's equations through boundary-integral equations (BIE), which operate exclusively on the scatterer's surface boundaries. When using appropriate basis functions to discretize the boundary only, the unknown quantities can be reduced dramatically while far-field radiation conditions are satisfied automatically through the Green function kernel. This project uses BEM to help model the scattering of EM waves from objects at the nanoscale dimension where the cross-sections correspond to closed 2-D curves (including conductive nanotubes with defined surface conductivity). Solving the boundary integrals yields nanotube surface currents which are used, via the same BIE, to determine the scattered field in any direction without requiring truncation of an infinite domain - a crucial benefit for inverse scattering. Traditional BEM implementations use piecewise-polynomial functions, but through the integration of IsoGeometric Analysis (IGA) in our framework, these are substituted with spline basis functions used in the boundary representation, which ultimately helps to achieve high accuracy and efficiency when dealing with complex curved shapes. Typically, these spline basis functions correspond to Non-Uniform Rational B-Splines (NURBS) basis which are commonly used in the representation of geometry in CAD systems [4].

The method of IsoGeometric Analysis (IGA), introduced by Hughes et al. in 2005 [5], mandates the use of the same spline basis functions utilized in CAD systems for curves and surfaces to be used in the representation of the unknown physical quantities. The exact representation of circles and other conic sections which NURBS representation allows, as well as its capacity to accurately represent a range of smooth and non-smooth shapes, enables the elimination of geometry-approximation errors while providing high-order smooth fields. The implementation results in improved accuracy for each degree of freedom above standard low-order finite and boundary elements. We use the acronym IGABEM to denote the coupling of the Boundary Element Method with the IGA paradigm as it appears in relevant literature; see for example [6, 7]. The parametrization of scatterer boundaries through NURBS enables IGABEM to use the same basis for approximating the unknown fields. The combined method delivers multiple significant advantages to users. IGABEM meshes have an advantage over classical BEM meshes by exactly representing smooth curved boundaries. The NURBS basis delivers smoothness that the solution inherits, which makes it an excellent choice for representing electromagnetic surface currents. The authors in [8, 9] established a complete convergence mechanism for IGABEM in electromagnetic applications and proved it delivered both swift convergence together with precise solution accuracy during scattering measurement. Handling inverse problems requires such features since a manageable NURBS control point set can efficiently generate various object during

optimization or reconstruction. The forward scattering problem is solved for any nanotube shape by IGABEM while the optimizer receives precise analytical sensitivities about the control-point coordinates. The hybrid optimizer used in our work (see also §2.6) benefits from these sensitivities when using measured data for unknown shape recovery, and every optimization step maintains physical validity through exact NURBS geometry.

For the implementation of the identified approach, MATLAB was selected as it includes all the needed tools, either internally or via toolboxes, to implement the IGABEM approach for the problem at hand. These include tools for representing and manipulating CAD geometry (NURBS Toolbox), numerical integration and linear algebra functions for materializing the boundary element method, as well as global and local optimizers for solving the inverse problem.

IGABEM is also of theoretical appeal beyond practical values in a series of recent studies on carbon nanotubes' interactions with electromagnetic waves. In a research by Kostas and Valagiannopoulos, they use an IGABEM-based shape optimization scheme to optimize the shape of a single nanotube under illumination by a plane wave. They searched for the cross section that maximizes the electric field energy stored inside the tube with a compact NURBS boundary model and a hybrid global-local optimizer. Robust over a wide range of wavelengths and incidence angles, they hold more than two orders of magnitude more energy than an equivalent circular tube and are also optimal [10].

Aznabayev et al. then followed this path for pairs of nanotubes. Building off this, they first chart how two identical shapes behave as one as a function of distance, size, and material parameters, finding naturally resonant configurations, and optimizing the tube cross-section, given fixed area and separation, to strengthen the coupling further. The resulting shapes have internal field strengths more than 30 times higher than the best circular pair baseline, and where the circular pair was weak, improvements go up to 180 times. Importantly, these "coupled" shapes exhibit high performance under any perturbations of the illumination angle or tube size, implying that the method can be used in practical metasurface or array designs [11].

Iznat et al. pursued a different optimization target where 'quasi transparent' nanotubes having very low surface conductivity were considered. In this case, the aim of this paper is to maximize the far-field scattering by modifying the cross-section shape. It was demonstrated that spike-like geometries of the same cross sectional area, but with enlarged circumference, result in maximal far-field scattering; considerably larger than that of a smooth circle. The same spiky motif is repeated throughout a large range of conductivities and sizes, suggesting a general principle for enhancing particles of low contrast [2]. Importantly, the same performance of the forward IGABEM can be employed in reverse, i.e., reconstruct an unknown shape from measured scattering, which is exactly one of the reasons that this capstone project exists. These research contributions in these works demonstrate that: (i) shape-optimized nanotubes can outperform standard ones

(i.e., circular) by orders of magnitude in scattering or field concentration, (ii) using robust NURBS parametric models allows a relatively small number of control points to cover a rich design space efficiently and robustly, and (iii) even under broad operating conditions, the optimized shapes retain their advantages showing its practicality. The wider appeal of shape-controlled electromagnetic design is evidenced also by other works in the pertinent literature; for example, optimization of dielectric nanorods with gradients by Araujo et al. [12].

With regards to applications beyond the studied, we envision a wide range of applicable fields and disciplines. Reconstructing object shapes from scattered signals can be advantageous for improved target identification [13] in many fields. For example, detection systems of aircraft or other objects at a distance utilize radar scattering signatures to recognize them; thus, having a tool to accurately infer shape from such data could be used to better detect stealthy targets or hidden objects. Optimized nanoelements put into telecommunications and photonics can serve as an intentional source of scattering that results in improved antennas, filters, or beam steering devices. Bioimaging techniques based on EM waves (e.g. microwaves, terahertz waves) usually use EM waves to non-invasively probe tissues, which often solve the inverse scattering problem to get images or shapes of hidden structures such as tumors [14]. These kinds of applications directly motivated to consider arbitrary-shaped nanostructures and focus on a solution ready to use for real implementation. These developments take place in the context of the Waves Interaction CArbon Nanotubes (WICAN) NU-funded CRP project. Two ongoing master's theses are further developing the IGABEM framework for more complex scenarios as a part of the research. Designing metasurfaces composed of many nanotubes with optimized shapes is considered an avenue to get emergent effects, such as almost perfect absorption or anomalous reflection of EM waves. The other active area of studies includes practical constraints such as limited angle or single-source illumination in the inverse algorithms to better mimic conditions in radar and imaging systems, where one cannot surround the object with sources or receivers. Initial results suggest that despite these constraints, our approach would still allow for shape reconstruction through efficient global optimization enhanced regularization.

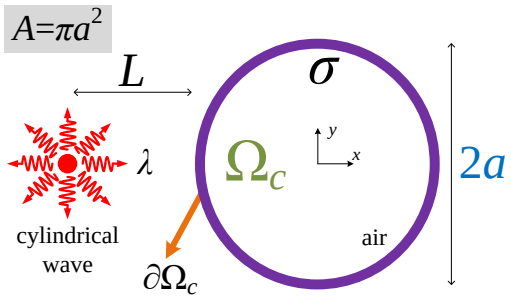
The rest of this report is structured as follows: In Chapter 2 (Methodology), the definition of the EM scattering problem using IGABEM is presented. Then, proceeds with the explanation of the inverse-shape reconstruction framework where a discussion regarding the NURBS-based parametric models is also being held. The boundary integral formulation producing surface currents, as well as global and local optimization algorithms exploited to recover a possibly complex but unknown shape from given scattered data are also discussed in this chapter. Chapter 3 (Implementation) then discusses particular software and computational aspects. In this work, the IGABEM solver is implemented in MATLAB, covering data structure, integration routines, and also the handling of collocation

and quadrature. The implementation is demonstrated to be accurate, to converge properly and computationally efficiently by means of a series of verification tests comparing against analytical solutions. In Chapter 4 (Results and Discussion), the outcomes of the Capstone project are presented. The results for every case being solved are presented, starting from the solver on a simple circular shape, then proceeding to examine several inverse-reconstruction case studies with constraints, full-angle, or limited-angle illumination, among others. Reconstruction accuracy, convergence properties, and robustness to data imperfections are each analyzed for each case, and insights on the method's practical strengths and weaknesses are drawn. Finally, Chapter 5 (Conclusion and Future Work) synthesizes the key findings: using IGABEM and advanced optimization, shape recovery of nanoscale scatterers can be achieved with a high level of reliability and computational performance. At the end, the concluding chapter closes by outlining promising extensions and future studies: generalization of the problem at different scales, consideration and impact of the surrounding real-world medium, or experimental validation, which can continue the work, based on this project.

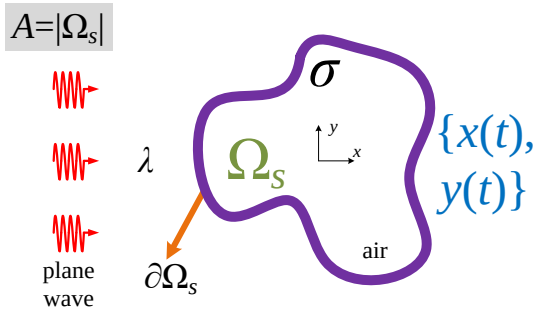
Chapter 2

Methodology

2.1 Electromagnetic Scattering



(a) Circular CNT with area A and surface conductivity at L+a distance from the point source



(b) Arbitrarily shaped CNT with the same properties

Figure 2.1: Examples of circular and arbitrary-shaped nanotubes illuminated by spherical and plane EM waves; see [1].

According to [1], the scattered electric field at an arbitrary observation point $\mathbf{P}$ may be computed by the mathematical derivation presented below. Specifically, a nanotube with complex surface conductivity σ, illuminated by cylindrical or planar electromagnetic waves with wavelength λ is assumed in sequel; see Fig. 2.1. First, the wave equation derived directly from Maxwell's equations is considered:

$$\left[\nabla^2 + k_0^2\right] E(\mathbf{P}) = \mathrm{i}k_0\eta_0 J(\mathbf{P}), \tag{2.1}$$

where $k_0 = 2\pi/\lambda$ is the wavenumber, and $\eta_0 = 120\pi$ is the wave impedance into vacuum, while $J(\mathbf{P})$ is the current density at the given location. Applying the corresponding Green's function of the linear differential operator in Eq. (2.1), with an arbitrary domain point and the source point denoted by $\mathbf{P}$ and $\mathbf{P}'$, respectively, the following derivation is obtained:

$$\begin{aligned}
\left[\nabla^2 + k_0^2\right] G(\mathbf{P},\mathbf{P}') &= -\,\delta(\|\mathbf{P}-\mathbf{P}'\|) \Rightarrow \\
-\mathrm{i}k_0\eta_0 \left[\nabla^2 + k_0^2\right] G(\mathbf{P},\mathbf{P}')J(\mathbf{P}') &= \mathrm{i}k_0\eta_0\delta(\|\mathbf{P}-\mathbf{P}'\|)J(\mathbf{P}') \Rightarrow \\
\left[\nabla^2 + k_0^2\right] \left(-\mathrm{i}k_0\eta_0 \int_\Omega G(\mathbf{P},\mathbf{P}')J(\mathbf{P}')ds_{P'}\right) &= \mathrm{i}k_0\eta_0 \int_\Omega \delta(\|\mathbf{P}-\mathbf{P}'\|)J(\mathbf{P}')ds_{P'} \Rightarrow \\
\left[\nabla^2 + k_0^2\right] \left(-\mathrm{i}k_0\eta_0 \int_\Omega G(\mathbf{P},\mathbf{P}')J(\mathbf{P}')ds_{P'}\right) &= \mathrm{i}k_0\eta_0 J(\mathbf{P}),
\end{aligned} \tag{2.2}$$

where δ is the two-dimensional Dirac delta function. By comparing Eq. (2.1) and Eq. (2.2), we can easily see that:

$$E(\mathbf{P}) = -\mathrm{i}k_0\eta_0 \int_\Omega G(\mathbf{P},\mathbf{P}')J(\mathbf{P}')ds_{(\mathbf{P}')}, \tag{2.3}$$

where the two-dimensional Green's function in free-space is given by:

$$G(\mathbf{P},\mathbf{P}') = -\frac{\mathrm{i}}{4}H_0^{(2)}\left(k_0\|\mathbf{P}-\mathbf{P}'\|\right), \tag{2.4}$$

with $H_0^{(2)}$ being the zeroth order Hankel function of the second kind.

Under the assumption that the current, which produces the scattered field, is concentrated at the boundary surface of the domain Ω, and using Ohm's Law, $J(\mathbf{P}') = \sigma E(\mathbf{P}')\delta(\mathbf{P}'-\partial\Omega)$ [15], the following expression is obtained from Eq. (2.3):

$$E_{\mathrm{scat}}(\mathbf{P}) = -\mathrm{i}k_0\sigma\eta_0 \oint_{\partial\Omega} G(\mathbf{P},\mathbf{P}')E(\mathbf{P}')d\ell_{(\mathbf{P}')}, \tag{2.5}$$

where σ is assumed constant along the boundary $\partial\Omega$.

The boundary condition, with E_{back} denoting the background field, becomes:

$$E_{\mathrm{back}}(\mathbf{P}) + E_{\mathrm{scat}}(\mathbf{P}) = E(\mathbf{P}). \tag{2.6}$$

where every point $\mathbf{P} \in \partial\Omega$. If we now substitute Eq. (2.5) into Eq. (2.6), the following integral equation is obtained:

$$E(\mathbf{P}) + \mathrm{i}k_0\sigma\eta_0 \oint_{\partial\Omega} G(\mathbf{P},\mathbf{P}')E(\mathbf{P}')d\ell_Q = E_{\mathrm{back}}(\mathbf{P}), \tag{2.7}$$

where $E(\mathbf{P})$, $\mathbf{P} = (x,y) \in \partial\Omega$, is the unknown electric field on the boundary. With regards to the background electric field, E_{back}, appearing above, we distinguish two cases: (a) the cylindrical wave produced from a point wave source, as shown on the Fig. 2.1a, and (b) the plane wave shown in Fig. 2.1b. The former case yields

$$E_{\mathrm{back}}(x,y) = -\frac{k_0\eta_0 I}{4}H_0^{(2)}\left(k_0\sqrt{(x+a+L)^2+y^2}\right), \tag{2.8}$$

which can be simplified to $E_{\mathrm{back}}(x,y) = H_0^{(2)}\left(k_0\sqrt{(x+a+L)^2+y^2}\right)$ by assuming that the source current I is of an appropriate value. For the plane wave case, E_{back} reads:

$$E_{\mathrm{back}}(x,y) = e^{-\mathrm{i}k_0(x\cos\theta + y\sin\theta)}, \tag{2.9}$$

which can be simplified to $E_{\mathrm{back}}(x,y) = e^{-\mathrm{i}k_0 x}$, when the plane wave is aligned with the horizontal x-axis, i.e., $\theta = 0$. Note here that θ corresponds to the angle between the x-axis in Fig. 2.1b and the wave propagation direction.

Considering now cylindrical coordinates, with (r,ϕ) and (r_0,ϕ_0) denoting the free-space point and the source position, respectively, Eq. (2.4) takes the following form:

$$G(r,\phi,r_0,\phi_0) = -\frac{\mathrm{i}}{4}H_0^{(2)}\left(k_0\sqrt{r^2+r_0^2-2rr_0\cos(\phi-\phi_0)}\right). \tag{2.10}$$

According to [16], the Hankel function has the following large-argument expansion:

$$H_\nu^{(2)}(z) \sim \sqrt{\frac{2}{\pi z}} e^{-\mathrm{i}(z-\frac{1}{2}\nu\pi-\frac{1}{4}\pi)}, \quad (-2\pi < \arg z < \pi). \tag{2.11}$$

Now, the expression for G can be obtained as $r \to \infty$ while r_0 remains fixed. Assuming that $\sqrt{r^2 + r_0^2 - 2rr_0\cos(\phi-\phi_0)} \approx r$, Newton's approximation is applied for the phase:

$$\begin{aligned} G(r,\phi,r_0,\phi_0) &\approx -\frac{\mathrm{i}}{4}\sqrt{\frac{2\mathrm{i}}{\pi k_0 r}} e^{-\mathrm{i}k_0 r\sqrt{1+\frac{r_0^2}{r^2}-2\frac{r_0}{r}\cos(\phi-\phi_0)}} \xrightarrow{r_0 \ll r} \\ G(r,\phi,r_0,\phi_0) &\approx -\frac{\mathrm{i}}{4}\sqrt{\frac{2\mathrm{i}}{\pi}}\frac{e^{-\mathrm{i}k_0 r}}{\sqrt{k_0 r}} e^{-\mathrm{i}k_0 r_0\cos(\phi-\phi_0)}. \end{aligned} \tag{2.12}$$

The z component of the scattering electric field from an arbitrary-shaped nanotube, as shown in Eq. (2.5), can be found as:

$$\begin{aligned} E_z(r_P,\phi_P) &= -\mathrm{i}k_0\sigma\eta_0 \oint_{\partial\Omega} G(r_P,\phi_P,r_{P'},\phi_{P'})E(r_{P'},\phi_{P'})d\ell_{P'} \xrightarrow{(2.12)} \\ E_z(r_P,\phi_P) &\approx -\frac{k_0\sigma\eta_0\sqrt{2\mathrm{i}}}{4\sqrt{\pi}}\frac{e^{-\mathrm{i}k_0 r_P}}{\sqrt{k_0 r_P}} U(\phi_P), \end{aligned} \tag{2.13}$$

where the voltage-like quantity, $U(\phi_P)$, is given by:

$$U(\phi_P) = \oint_{\partial\Omega} e^{-\mathrm{i}k_0 r_{P'}\cos(\phi_P-\phi_{P'})} E(r_{P'},\phi_{P'})d\ell_{P'}. \tag{2.14}$$

Finally, after applying Poynting's Theorem for the field in Eq. (2.5), the scattered power per unit length in the z-axis can be computed by the following expression:

$$P_{\text{scat}} = \frac{k_0}{16\pi\eta_0}|\sigma\eta_0|^2 \int_0^{2\pi} |U(\phi)|^2\, d\phi. \tag{2.15}$$

2.2 Iso-Geometric Analysis Boundary Element Method

As previously discussed (see chapter 1), the isogeometric analysis approach employs the basis functions used in the description of the domain (and its boundary) to express the unknown field quantity. Therefore, in the context of the boundary element method, the spline basis functions used in representing $\partial\Omega$ in Fig. 2.1 should be also used in expressing the unknown electric field $E(\mathbf{P})$. Specifically, the domain boundary $\partial\Omega$ is represented as a regular parametric NURBS curve $r(t), t \in I$, with $\mathcal{I} = \{t_0, t_1, \dots, t_{n+k}\}$ being the so-called knot-vector that partitions I and comprises $(n+k+1)$ non-decreasing components. Consequently, a NURBS curve is expressed as follows:

$$r(t) \coloneqq \sum_{t=0}^{n} \mathbf{b}_i N_{i,k}(t), \quad t \in I = [t_{k-1}, t_{n-1}], \tag{2.16}$$

where $\{N_{i,k}\}_{i=0}^n$ are the B-Spline basis functions defined in $\mathcal{I}$, with k being their order.

The projection of Eq. (2.16) on the 2D Euclidean space is:

$$\bar{r}(t) = \sum_{i=0}^{n} \bar{\mathbf{b}}_i \frac{w_i N_{i,k}(t)}{\sum_{j=0}^{n} w_j N_{j,k}(t)} = \sum_{i=0}^{n} \bar{\mathbf{b}}_i R_{i,k}(t), \tag{2.17}$$

where $\mathbf{b}_i = [x_i/w_i, y_i/w_i]^T$ are the control points in the 2D Euclidean space, and $\{R_{i,k}(t)\}_{i=0}^{n}$ are the rational B-Spline basis functions.

The IGA approach combines design with analysis; therefore, the unknown quantity $E(r_{P'}, \phi_{P'})$ in Eq. (2.14), needs to be defined with the same basis functions $R_{i,k}(t)$ appearing in Eq. (2.17). Hence, the assumed representation of the electric field becomes:

$$\tilde{E}(r_{P'}, \phi_{P'}) = \tilde{E}(x(t), y(t)) \coloneqq \sum_{i=0}^{n} e_i R_{i,k}(t), \quad t \in I, \tag{2.18}$$

where the rational basis function are defined over the same knotvector $\mathcal{I}$. By substituting Eq. (2.18) into the Eq. (2.7), we obtain the following continuous approximation of the integral equation:

$$\sum_{i=0}^{n} e_i R_{i,k}(t) + \mathrm{i}k_0\sigma\eta_0 \sum_{i=0}^{n} e_i \oint_{\partial\Omega} G(\mathbf{P}, \mathbf{P}') R_{i,k}(t) d\ell_Q = E_{\text{back}}(\mathbf{P}). \tag{2.19}$$

A more accurate approximation can be found by generating a sequence of nested finite-dimensional spline spaces $\mathcal{S}_k(\mathcal{I}^{(0)}) = \mathcal{S}_k(\mathcal{I}) \subset \mathcal{S}_k(\mathcal{I}^{(1)}) \subset \ldots \subset \mathcal{S}_k(\mathcal{I}^{(m)})$, where $m \in \mathbb{N}^*$, is the number of knots inserted in $\mathcal{I}$.

Additionally, to improve convergence, a collocation method that utilizes the Greville abscissae [17, 18] is applied. Collocation method provides a solution that satisfies the given equation at the given collocation points. The Greville abscissae are knot averages, defined as $\hat{t}_j = \frac{1}{k-1}(t_{i+1} + t_{i+2} + \ldots + t_{i+k-1})$, which guarantee collocation close to maximum of the corresponding basis function and therefore, help in maintaining a good condition number for the matrix in the resulting linear system. Finally, Eq. (2.7), after collocating at $n + m + 1$ Greville points, yields the linear system below:

$$\sum_{i=0}^{n+m} e_i R_{i,k}^{(m)}(\hat{t}_j) + \mathrm{i}k_0\sigma\eta_0 \sum_{i=0}^{n+m} e_i \int_I G(\hat{t}_j, \tau) R_{i,k}^{(m)}(\tau) \|\dot{\mathbf{r}}(\tau)\| d\tau = E_{\text{back}}(\hat{t}_j), \tag{2.20}$$

where obviously $j = 0, 1, \ldots, n + m$. Solving Eq. (2.20) produces the control values, $\{e_i\}_{i=0}^{n+m}$ that are then plugged in Eq. (2.18) to find the spline approximation of the electric field on the boundary.

Consequently, Eq. (2.14) can be evaluated using the results from the electric field approximation:

$$U(\phi) = \int_I e^{-\mathrm{i}k_0 \|r(\tau)\| \cos(\phi - \varphi(\tau))} \tilde{E}(\tau) \|\dot{r}(\tau)\| d\tau, \tag{2.21}$$

where $\varphi(\tau) = \arctan\left(y(\tau)/x(\tau)\right)$. Finally, Eq. 2.21 can be plugged in Eq. 2.15 for calculating the scattering power per unit length in the z-axis.

2.3 Inverse Problem Formulations

By obtaining the scattering profile of conducting objects, we may utilize the obtained information in various applications, ranging from radar and telecommunication systems to biomedical technologies for non-invasive imaging [19, 20]. In this context, our capstone project focuses on solving seven different inverse problems which are applicable, to varying extents, in these fields. A summary of these problems' characteristics is presented in Table 2.1. The inverse problems considered in this work include the following cases: (i-ii) unknown shape with some known material and integral geometric characteristics, i.e., (i) area, and material properties (electric surface conductivity) or (ii) circumference, and material properties, (iii) unknown electric surface conductivity, (iv) unknown position of the light source, (v) unknown shape with known material properties, but no information on integral geometric characteristics, (vi) unknown shape, material properties and integral geometric characteristics with two given scattering signals, and finally (vii) a variation of (vi) in which only parts of the scattering signals are known. For all cases, the L^2-norm, between given and calculated scattering profiles, is used as the cost function. The additional constraint on limited knowledge of the scattering signals can be applied to any of the problems above, but in the scope of this project it was only use with Case 6 setup, resulting in the last case, Case 7, appearing in Table 2.1.

First, we consider a given scattering signal $U_0(\phi)$ (see Eq. (2.14)) coming from a nanotube placed in vacuum and illuminated by either cylindrical or planar electromagnetic waves with a wavelength λ. The nanotube's material and cross-sectional area are known, but not its boundary shape. We further assume here that Ω denotes the region enclosed by the nanotube, $\partial\Omega$ its boundary shape, $A_0 = |\Omega|$ the known cross-sectional area, and finally σ corresponds to the electric conductivity of the known material. We now want to determine the unknown shape $\partial\Omega$ which produces that scattering signal. If an arbitrary-shaped nanotube can be uniquely determined by an appropriate vector $\mathbf{v}$, with boundary $\partial\Omega_s$ and area $A_s(\mathbf{v}) = |\Omega_s|$, the problem of determining $\mathbf{v}$ so that $U_s(\phi) \approx U_0(\phi)$ (and consequently $\partial\Omega \approx \partial\Omega_s$) can be formulated as the constrained minimization problem shown below.

$$\begin{aligned} \text{find } \mathbf{v}^* : \quad & \mathrm{f}(\mathbf{v}^*) = \min_{\mathbf{v}} \int_0^{2\pi} (U_0(\phi) - U_S(\mathbf{v}, \phi))^2 \, d\phi \\ \text{subject to : } \quad & \mathbf{v} \in \mathcal{V} \subset \mathbb{R}^n, \\ & |A_S(\mathbf{v}) - A_0| < \epsilon \ll 1, \\ & \text{with } A_0,\ \lambda,\ \sigma\eta_0,\ U_0(\phi) \text{ given}, \end{aligned} \tag{2.22}$$

where n is the number of parameters used for determining the shape of the nanotube, and ϵ is an arbitrarily small positive number corresponding to the area constraint tolerance.

The second inverse problem is defined similarly to the first one, but with a known a cross-sectional circumference $C_0 = |\partial\Omega|$ substituting the equiareal constraint. If

Table 2.1: Summary of known and unknown variables for studied inverse problems.

No.	Unknown variables	Known values & Constraints[†]
1	Geometry (encoded with a vector $\mathbf{v}$)	Area, $A(\mathbf{v})$ (constraint) Scattering profile, $U_0(\phi)$, $\phi = 0, \ldots, 2\pi$ Electric surface conductivity, σ Position of the EM wave source, $\mathbf{p}$
2	Geometry, $\mathbf{v}$	Circumference, $C(\mathbf{v})$ (constraint) Scattering profile, $U_0(\phi)$, $\phi = 0, \ldots, 2\pi$ Electric surface conductivity, σ Position of the EM wave source, $\mathbf{p}$
3	Material properties (electric surface conductivity, σ)	$\sigma \in \mathcal{C} \subset \mathbb{C}$ (constraint) Scattering profile, $U_0(\phi)$, $\phi = 0, \ldots, 2\pi$ Position of the EM wave source, $\mathbf{p}$ Geometry, $\mathbf{v}$
4	Position of the EM wave source, $\mathbf{p}$ (For plane waves, only direction is needed)	$\mathbf{p} \in \mathcal{P} \subset \mathbb{R}^2$ (constraint) Scattering profile, $U_0(\phi)$, $\phi = 0, \ldots, 2\pi$ Electric surface conductivity, σ Geometry, $\mathbf{v}$
5	Geometry, $\mathbf{v}$	Scattering profile, $U_1(\phi)$, $\phi = 0, \ldots, 2\pi$ Scattering profile, $U_2(\phi)$, $\phi = 0, \ldots, 2\pi$ Electric surface conductivity, σ Positions of EM sources, $\mathbf{p}_1$ and $\mathbf{p}_2$
6	Geometry, $\mathbf{v}$ Electric surface conductivity, σ	Scattering profile, $U_1(\phi)$, $\phi = 0, \ldots, 2\pi$ Scattering profile, $U_2(\phi)$, $\phi = 0, \ldots, 2\pi$ $\sigma \in \mathcal{C} \subset \mathbb{C}$ (constraint) Positions of EM sources, $\mathbf{p}_1$ and $\mathbf{p}_2$
7[‡]	Geometry, $\mathbf{v}$ Electric surface conductivity, σ	Scattering profile, $U_1(\phi)$, $\phi = \phi_1, \ldots, \phi_2$ Scattering profile, $U_2(\phi)$, $\phi = \phi_1, \ldots, \phi_2$ with $\lvert\phi_2 - \phi_1\rvert < 2\pi$ $\sigma \in \mathcal{C} \subset \mathbb{C}$ (constraint) Positions of EM sources, $\mathbf{p}_1$ and $\mathbf{p}_2$

[†] In all cases where $\mathbf{v}$ is unknown, the additional constraint $\mathbf{v} \in \mathcal{V} \subset \mathbb{R}^n$ is used.

[‡] Variation of case 6 with limited range of scattering signals known.

$C_S(\mathbf{v}) = |\partial\Omega_S|$ denotes the circumference of the sought-for shape, the resulting optimization problem can be written as:

$$\begin{aligned} \text{find}\,\mathbf{v}^* : \quad & \mathrm{f}(\mathbf{v}^*) = \min_{\mathbf{v}} \int_0^{2\pi} (U_0(\phi) - U_S(\mathbf{v},\phi))^2\, d\phi \\ \text{subject to} : \quad & \mathbf{v} \in \mathcal{V} \subset \mathbb{R}^n, \\ & |C_S(\mathbf{v}) - C_0| < \epsilon \ll 1, \\ & \text{with}\, C_0,\ \lambda,\ \sigma\eta_0,\ U_0(\phi)\, \text{given}, \end{aligned} \tag{2.23}$$

where in this case ϵ corresponds to the arc-length constraint tolerance.

In the third case, the boundary shape is known, but not the nanotube's material. Therefore, σ is the sought-for quantity. The optimization problem in this case reads:

$$\begin{aligned} \text{find}\,\sigma^* : \quad & \mathrm{f}(\sigma^*) = \min_{\sigma} \int_0^{2\pi} (U_0(\phi) - U_S(\sigma,\phi))^2\, d\phi \\ \text{subject to} : \quad & \sigma \in \mathcal{C} \subset \mathbb{C}, \\ & \text{with}\, \mathbf{v},\ \lambda,\ U_0(\phi)\, \text{given}. \end{aligned} \tag{2.24}$$

In the fourth case, the position $\mathbf{p}$ of the EM wave source is unknown (alternatively for plane waves, the direction of the incoming wave is unknown). Assuming now that the nanotube's centroid is at the origin, the optimization problem can be formulated as:

$$\begin{aligned} \text{find}\,\mathbf{p}^* : \quad & \mathrm{f}(\mathbf{p}^*) = \min_{\mathbf{p}} \int_0^{2\pi} (U_0(\phi) - U_S(\mathbf{p},\phi))^2\, d\phi \\ \text{subject to} : \quad & \mathbf{p} \in \mathcal{P} \subset \mathbb{R}^2, \\ & \text{with} \quad \mathbf{v},\ \lambda,\ \sigma\eta_0,\ U_0(\phi)\, \text{given}. \end{aligned} \tag{2.25}$$

The fifth case is potentially the most interesting, as information about the nanotube's geometry is assumed to be fully unknown. Here, the scattering signals from two non-coinciding point sources/directions are considered, leading to following shape-identification problem:

$$\begin{aligned} \text{find}\,\mathbf{v}^* : \quad & \mathrm{f}(\mathbf{v}^*) = \min_{\mathbf{v}} \int_0^{2\pi} (U_1(\phi) - U_{S_1}(\mathbf{v}))^2\, d\phi + \int_0^{2\pi} (U_2(\phi) - U_{S_2}(\mathbf{v}))^2\, d\phi \\ \text{subject to} : \quad & \mathbf{v} \in \mathcal{V} \subset \mathbb{R}^n, \\ & \text{with}\ \lambda,\ \sigma\eta_0,\ U_1(\phi),\ U_2(\phi)\, \text{given}, \end{aligned} \tag{2.26}$$

where U_1, U_2 correspond to the nanotube's scattering signal, when illuminated by the first and second point source/direction, respectively.

The sixth case is an extension of the fifth case adapted to practical applications. Here, in addition to geometry and its integral characteristics, electromagnetic conductivity is also unknown. The optimization problem has 2 additional unknown variables (as conductivity

is a complex quantity) and boundary constraints as follows:

$$\begin{aligned}
&\text{find}\,\mathbf{v}^*,\sigma^*: \quad \mathrm{f}(\sigma^*,\mathbf{v}^*)=\min_{\mathbf{v}}\int_0^{2\pi}(U_1(\phi)-U_{S_1}(\mathbf{v},\sigma))^2\,d\phi+\int_0^{2\pi}(U_2(\phi)-U_{S_2}(\mathbf{v},\sigma))^2\,d\phi\\
&\text{subject to}: \quad \mathbf{v}\in\mathcal{V}\subset\mathbb{R}^n,\\
&\qquad\qquad \sigma\in\mathcal{C}\subset\mathbb{C},\\
&\qquad\qquad \text{with }\ \lambda,\ U_1(\phi),\ U_2(\phi)\ \text{given},
\end{aligned} \tag{2.27}$$

where U_1, U_2, as in the fifth case, correspond to the nanotube's scattering signal, when illuminated by the first and second point source/direction, respectively.

The final case, case 7, is the hardest and more general case, where a limited portion of the scattering signals is assumed. Considering that α is the angle of illumination of the first EM wave source, the range was limited to $\phi\in[\alpha-\xi;\alpha+\xi]$, and therefore the values of ϕ_1 and ϕ_2 appearing in Table 2.1 become $\alpha-\xi$ and $\alpha+\xi$, respectively, where ξ can be adjusted depending on the problem at hand.

2.4 Parametric Models

Utilizing NURBS control points as direct design variables in IGA optimization can result in complex high-dimensional optimization problems that require a lot of constraints to avoid issues, such as self-intersecting or invalid shapes; see also [1]. Therefore, following parametric models were used to generate valid NURBS curves that also incorporate the automatic satisfaction of the integral geometric constraints (area/circumference) when needed:

1. Equal-sector-based parametric model (see [1], Fig. 2.2)

   Design vector $\mathbf{v}=(v_0,v_1,\ldots,v_{n-1})$, $\mathbf{v}\in\mathcal{V}=[0,1]^n$, with $n=2m$, is used to define the number of shape-controlling points (SCPs), m. Consequently, the polar coordinates can be found as follows:

   $$\begin{aligned}
   &\mathbf{q}_i=\left[r_i\cos\left(\phi_i+\frac{2\pi i}{m}\right),r_i\sin\left(\phi_i+\frac{2\pi i}{m}\right)\right]^T\\
   &\text{where}\quad r_i=v_i(r_{\max}-\epsilon)+\epsilon,\quad \phi_i=\frac{2\pi v_{i+m}}{m},\\
   &\qquad\qquad \mathbf{v}\in[0,1]^{2m},\ \text{i.e.},\ v_i,v_{i+m}\in[0,1],\ i=0,1,\ldots,m-1,
   \end{aligned} \tag{2.28}$$

   where ϵ is again an arbitrarily small positive number avoiding the placement of SCPs at the center, whereas $r_{\max}$ is the maximum distance from the point to the center.

2. Area-constrained equal-sector-based parametric model

This parametric model extends the first one by creating a shape that have user-input area A_0:

$$\begin{aligned} \mathbf{q}_i &= \left[r_i \cos\left(\phi_i + \frac{2\pi i}{m}\right), r_i \sin\left(\phi_i + \frac{2\pi i}{m}\right) \right]^T \\ \text{where} \quad & r_i = v_i(r_{\max} - \epsilon_r) + \epsilon_r, \quad \phi_i = \frac{2\pi v_{i+m}}{m}, \\ & |A(\mathbf{v}) - A_0| < \epsilon_A \\ & \mathbf{v} \in [0,1]^{2m}, \text{ i.e., } v_i, v_{i+m} \in [0,1],\ i = 0, 1, \ldots, m-1, \end{aligned} \tag{2.29}$$

where ϵ_r and ϵ_A are the arbitrarily small positive numbers corresponding to difference tolerance and area tolerance, respectively. The satisfaction of the area constraint is accomplished by scaling appropriately the generated shape, using a quick univariate minimization technique such as Newton's method.

3. Circumference-constrained equal-sector-based parametric model

This parametric model is another variation of the first model that creates a shape that have user-input circumference C_0:

$$\begin{aligned} \mathbf{q}_i &= \left[r_i \cos\left(\phi_i + \frac{2\pi i}{m}\right), r_i \sin\left(\phi_i + \frac{2\pi i}{m}\right) \right]^T \\ \text{where} \quad & r_i = v_i(r_{\max} - \epsilon_r) + \epsilon_r, \quad \phi_i = \frac{2\pi v_{i+m}}{m}, \\ & |C(\mathbf{v}) - C_0| < \epsilon_C \\ & \mathbf{v} \in [0,1]^{2m}, \text{ i.e., } v_i, v_{i+m} \in [0,1],\ i = 0, 1, \ldots, m-1, \end{aligned} \tag{2.30}$$

where ϵ_r and ϵ_C are the arbitrarily small positive numbers corresponding to difference tolerance and circumference tolerance, respectively. As in the previous case, the circumference constraint is handled via scaling.

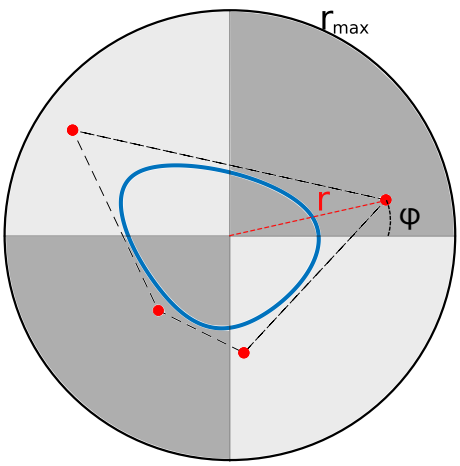


Figure 2.2: Equal-sector-based parametric model [2].

2.5 Methodological Accuracy of IGABEM

One of the main advantages of the IGABEM method is its substantially high accuracy and fast convergence rates, even when utilizing a small number of degrees of freedom [2]. Additionally, authors in [2] validated the same IGABEM approach used in this project by comparing its numerical approximation to the analytic expression for the electric field on the boundary of a circular nanotube illuminated by a cylindrical EM wave as shown in Fig. 2.3. They assess the deviation of the numerical solution using the L^{∞} and L^{2} norms as they apply increasing h- and p-refinements, where h-refinement is performed by knot insertion and p-refinement involves degree elevation of the spline bases representing the geometry and electric field. The results are in agreement with similar accuracy studies implemented in pertinent literature [5, 21].

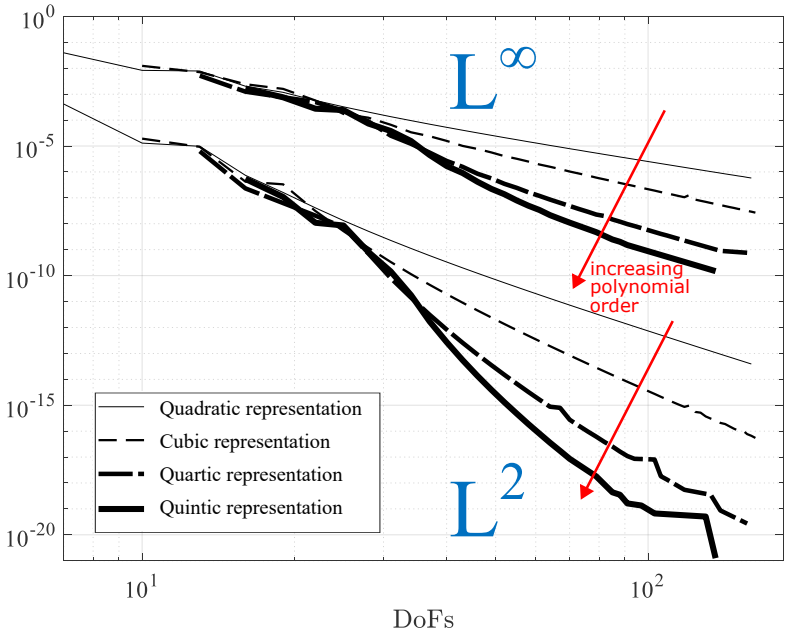


Figure 2.3: Difference between the analytical and numerical assessment of the electric field on the circular boundary, as determined by the L^{∞} and L^{2} norms, given an increasing number of polynomial order degrees of freedom (DoFs) [2].

2.6 Sampling Points and Degrees of Freedom Requirements

In the context of determining the shape of arbitrary nanoelements from EM wave scattering, the appropriate selection of sampling points and degrees of freedom (DOFs) is crucial to ensure accurate reconstruction while minimizing computational costs and time.

The Nyquist-Shannon Sampling Theorem provides a fundamental guideline for determining the minimum number of required sampling points [22]. It states that a

continuous signal can be completely reconstructed from its samples if the sampling frequency is at least twice the highest frequency present in the signal (the Nyquist rate). Nyquist–Shannon sampling theorem:

$$2B < f_s, \tag{2.31}$$

where B is the bandlimit of the signal, and f_s is the sufficient sampling rate.

In addition to sampling points requirement, the number of DOFs affect the number of collocation points required to accurately describe the unknown electric field. To identify the optimal number of DOFs, relative errors from power (Eq. (2.15)) and voltage (Eq. (2.14)) were computed while iteratively increasing the number of knots which are directly linked to the number of employed DOFs. The DOFs that provide errors less than 1% are incorporated into the final framework. The results from sapling points and DOFs requirement analysis are included in chapter 4.

2.7 Optimization using MATLAB

To solve the optimization problems formulated in the previous section, the following functions from MATLAB's Optimization and Global Optimization toolboxes were used:

- **`fmincon()`**: a gradient-based nonlinear programming optimizer supporting linear and non-linear constraints [23];
- **`patternsearch()`**: a direct search global optimizer supporting linear and non-linear constraints [24];
- a hybrid approach combining **`fmincon()`** and **`patternsearch()`**.

Initially, `fmincon()` was used to solve the preliminary inverse optimization problems that focused on relatively simple unknown geometrical boundaries. `fmincon()` searches for a constrained minimum of a scalar cost function by starting from an initial guess point. There are four optimization algorithms within `fmincon()`: Interior-Point, Sequential Quadratic Programming (SQP), Active-Set, and Trust-Region-Reflective Optimization [25, 26, 27, 28]. In our work, the SQP algorithm was mainly utilized due to its speed for relatively smooth objective functions when good estimation points are provided. SQP is strictly feasible with respect to bounds, which is beneficial for nonlinear optimization with complex objective functions and constraints [29]. The objective functions for `fmincon()` were implemented according to Eqs. (2.22),(2.23), and (2.26). Additionally, appropriate boundary constraints were applied for the shape-controlling parametric vector $\mathbf{v}$.

`patternsearch()` was initially implemented for cases 3 and 4 with unknown material properties and source signal position, respectively. The advantage of `patternsearch()`

for these cases is that it does not require information about the gradient of objective function. The direct search is efficient, when the objective function's gradient is hard to obtain, or it is non-differentiable or discontinuous [30]. `patternsearch()` uses a "Poll Method" by checking neighboring points in the current mesh at each iteration and comparing their values to the objective function value at current point [31]. This approach provides significant robustness to the solution as `patternsearch()` can simply take smaller steps if the current poll of points are not returning an improved or feasible point. The objective functions for `patternsearch()` are defined similarly with `fmincon()`. Note that for Eqs. (2.24) and (2.25), boundary constraints need to be adjusted accordingly.

After several trials of solving each inverse problem, a hybrid approach combining the two methods proved more appropriate as it stroke balance a between computational time and effectiveness in reaching the global optimum. As `patternsearch()` directly searches for the global minimum, it is used for the first iterations to correctly identify the neighborhood of the optimum point and subsequently, an estimate in this region is used as the initial guess for `fmincon()` to speed up the convergence process.

Chapter 3

MATLAB Implementation

The project involves extensive code development, including the implementation of the IGABEM approach and optimization problems and the utilization of NURBS Toolbox in MATLAB environment. The input characteristic parameters are as follows:

- Wavelength: $\lambda = 1$ length unit;
- Area and circumference: normalized with λ^2 and λ, respectively;
- Complex electric conductivity: $\sigma = 0.001 - 0.01\mathrm{i}$ (when given);
- Wave impedance into vacuum: $\eta_0 = 120\pi$ Ohm;
- Wavenumber: $k_0 = 2\pi/\lambda$;
- Distance from the point wave source: L (normalized with λ).

For the inverse problems with unknown shaped nanotubes illuminated with single source, the area and circumference values were set to:

- Area: $A_0 = 2$ square units;
- Circumference: $C_0 = 5.6224$ units.

The implementation started from inverse optimizing the simple shapes with unknown geometry but known area. Throughout the project, the code was constantly improved to prove the initial hypothesis that it is possible to identify nanoelements shapes and other characteristics knowing only scattering profiles. The latest full implementation can be found in the Appendix B, which shows the source code for the seventh case that is extended version of sixth case with additional constraint of limited sampling points range.

3.1 Integration of NURBS Toolbox

The curve generation in this framework heavily depends in NURBS Toolbox (see [32]). This toolbox provides a comprehensive set of MATLAB functions for creating, manipulating, and evaluating NURBS curves, surfaces, and volumes. It supports key operations such as interpolation, knot insertion, refinement, differentiation, and visualization. In the scope of this project, only functionalities for curves were used.

NURBS Toolbox functions are extensively used for parametric models. The latest version of the code utilizes the equal-sector-based parametric model as it does not require any information on area or circumference. Below is the function `freeform_geom()` that was developed for WICAN project (see [33]):

```
function [geom] = freeform_geom(params,max_r)
geom = [];
if isempty(params(params>1|params<0)) && max_r>0
    n = length(params);
    r = max_r*(0.99*params(1,:)+0.01);
    phi = (params(2,:)+(0:n-1))*2*pi/n;
    ctrl_pts = [r.*cos(phi);r.*sin(phi)];
    ctrl_pts = [ctrl_pts(:,end),ctrl_pts,ctrl_pts(:,1:2)];
    geom = nrbmak(ctrl_pts,0:1/(n+6):1); %assume cubics with uniform knots
    geom = nrbclamp(geom);
    %normalize knot vector
    geom.knots = (geom.knots-geom.knots(1))./(geom.knots(end)-geom.knots(1));
    geom = convertNRB(geom);
end

end
```

This function generates a closed freeform NURBS curve based on a set of normalized shape parameters `params` and a maximum radius `max_r`. It follows the Eq. (2.28), constructing the control points by mapping radial and angular positions. It further build a cubic NURBS curve, clamps and normalizes the knot vector, and converts the resulting curve into a standard NURBS format for further calculations.

The parametric models for area-constrained and circumference-constrained problems utilize MATLAB's `while`-loop until generated curve satisfies the input area tolerance or circumference tolerance.

3.2 IGABEM Functions

IGABEM functions were adapted from the project WICAN (see [33]). The function `igabem_one_tube_gpu` solves the Eq. (2.20). Inside this calculation, there is a function `E_back_theta` that corresponds to Eq. (2.9).

It is also relevant to introduce one of the objective functions as they extensively utilize IGABEM calculations. Below is the full implementation of the objective function for the seventh case, which is with additional constraint on sampling points range; see Eq. (2.27). The calculations of scattering profile correspond to Eq. (2.21).

```
function [res] = obj_fun_double_sigma_60(x, max_r_val, wavelength_val, angle_vals, phi_step_val, s
persistent max_r;
persistent signals;
persistent wavelength;
```

```
persistent angles;
persistent phi_step;
if nargin==6
    max_r = max_r_val;
    wavelength = wavelength_val;
    angles = angle_vals;
    phi_step = phi_step_val;
    signals = signal_vals;
else
    sigma = x(1,1)-x(2,1)*1i;
    DoFs = 60;
    integration_step = 1e-4;
    params = x(:,2:end);
    geom = freeform_geom(params,max_r); %change max_r accordingly
    knots_in = ceil((DoFs-geom.number)/(length(unique(geom.knots))-1));
    if knots_in<1
        knots_in=0;
    end
    geom = insertKnots(geom,knots_in);
    res = 0;
    for i=2:-1:1
        e_geom = igabem_one_tube_gpu(geom,wavelength,sigma,integration_step,angles(i));
        U = scat_phi_calc_GPU(geom,e_geom,wavelength,phi_step,integration_step);
        U = U(:,121:241);
        res = res+norm(signals(i,:)-U(2,:),2);
    end
end
```

Other objective functions were developed in the similar manner. The objective functions use `persistent` variables to store constant parameters that do not change throughout the optimization process. All objective functions compute L^2-norm between computed and given scattering profiles taking different unknown variables based on one of the 7 cases.

3.3 Optimization Implementation

To implement the inverse problems, MATLAB's Optimization and Global Optimization toolboxes were used. Optimization Toolbox offers functions that solve minimization or maximization problems to find parameters given certain constraints [34]. On the other hand, Global Optimization Toolbox provides functions for finding global solutions, where optimization problem can contain multiple minima or maxima [35].

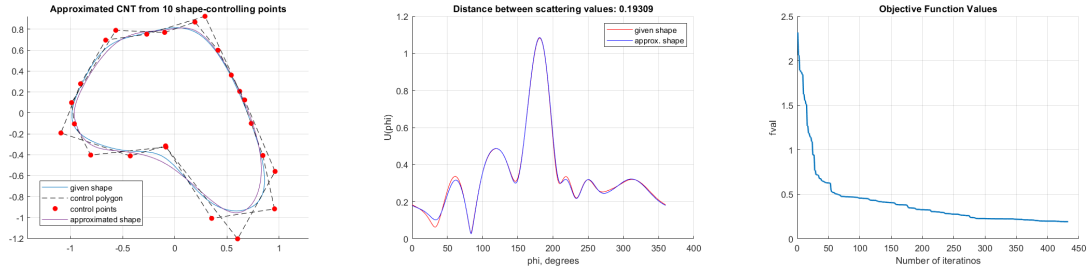
A hybrid approach was implemented utilizing both of these toolboxes. As it was discussed in the Section 2.7, this approach was more effective than using gradient-based or global optimization alone. First, `patternsearch()` was used to get the parameters near global solution. The example of solution convergence using only `patternsearch()`

can be seen in Fig. 3.1c. This algorithm was particularly chosen because it is a direct search optimizer that does not require information about the gradient of the objective function [24], which is efficient for this project because involved objective function's gradient are hard to obtain. Then, after around 1000-3000 function evaluations, depending on the case, the parameters from `patternsearch()` are passed to `fmincon()`; see [23]. In this work, `fmincon()`'s SQP algorithm was used due to its quick solution for relatively smooth objective functions. Fig. Below is the implementation of hybrid approach for Case 1:

```
inverse_area([],U_0,area,DoFs,wavelength,sigma,mode,0 ,...
distance,phi,integration_step);
options = optimoptions('patternsearch','Display','iter',...
'StepTolerance',1e-2,'MaxFunctionEvaluations',200,'OutputFcn',...
@outfun);
optparams0 = patternsearch(@inverse_area, x0, A, b, Aeq, beq,...
lb, ub, nonlcon, options);

opts = optimoptions('fmincon','Algorithm','sqp','Display',...
'iter','StepTolerance',1e-4,'MaxFunctionEvaluations',10000,...
'OutputFcn',@outfunc);
optparams = fmincon(@inverse_area, x0, A, b, Aeq, beq,...
lb, ub, nonlcon, opts);
```

The optimization part for each case was implemented in the similar manner to the examples above. The latest full implementation can be found in the Appendix B.



(a) Shape approximation. (b) Scattering profile. (c) Objective function.

Figure 3.1: Unknown shape with area parametric model solved using patternsearch() with $\phi = 360$.

3.4 Computational Time

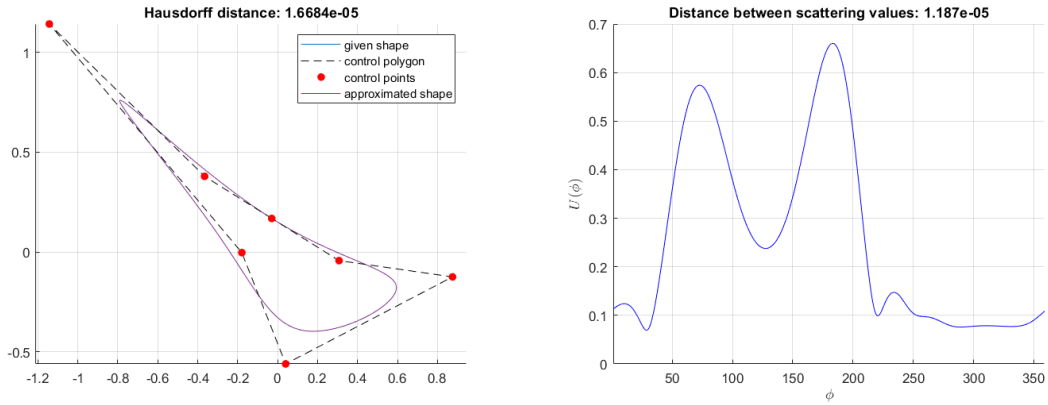
Computational time comparison of different algorithms are demonstrated in the Table 3.1. The corresponding results can be found in Figs. 3.1. The results are taken from area-

constrained simulations, as they were the most frequently tested for sampling points analysis. From the table, it is clearly seen how despite high number of sampling points and DOFs, hybrid approach is faster and requires less iterations to reach to the solution.

ϕ	DOFs	Algorithm	Time (s)	Iterations	Figure
360	120	`fmincon()`	19251	304	A.5
360	120	`patternsearch()`	26290	438	3.1
360	120	hybrid	16169	278	4.6

Table 3.1: Summary of area-constrained optimization results with corresponding phi numbers, DOFS, algorithms, computational times, and iteration counts.

The latest framework can solve Cases 6 and 7 with much higher computational efficiency. For example, Fig. 3.2 demonstrates the results that can be directly obtained right after simulation of Case 6. This particular solution took only 3028.476507 seconds and 275 iterations to complete.



(a) Shape approximation.

(b) Scattering profile.

Figure 3.2: Approximation of an arbitrary shape with 5 shape-controlling points in Case 6 with 3000 function evaluations during `patternsearch()`. The scattered signal corresponds to a plane wave.

Chapter 4

Results and Discussion

4.1 Sampling points and DOFs requirements

To reduce computational time and costs, it is crucial to identify the minimum optimal number of sampling points and DOFs.

An important discussion pertains to the relationship between the number of sampling points $\{\phi_i\}$ and the accuracy of the inverse optimization solution. Figure 4.1 demonstrates a preliminary study of the convergence to the true shape for an increasing number of sampled points from the signal. From this figure, we can observe that for the given signal and the area-constrained problem with an arbitrary shape determined by 10 shape-controlling points, approximately 40 equally distributed data points are enough to get an accurate result. In the older examples, 360 data points were collected (with a step of 1 degree) around the shape. By observing this graph, it was found that a smaller number of data points might be generally needed.

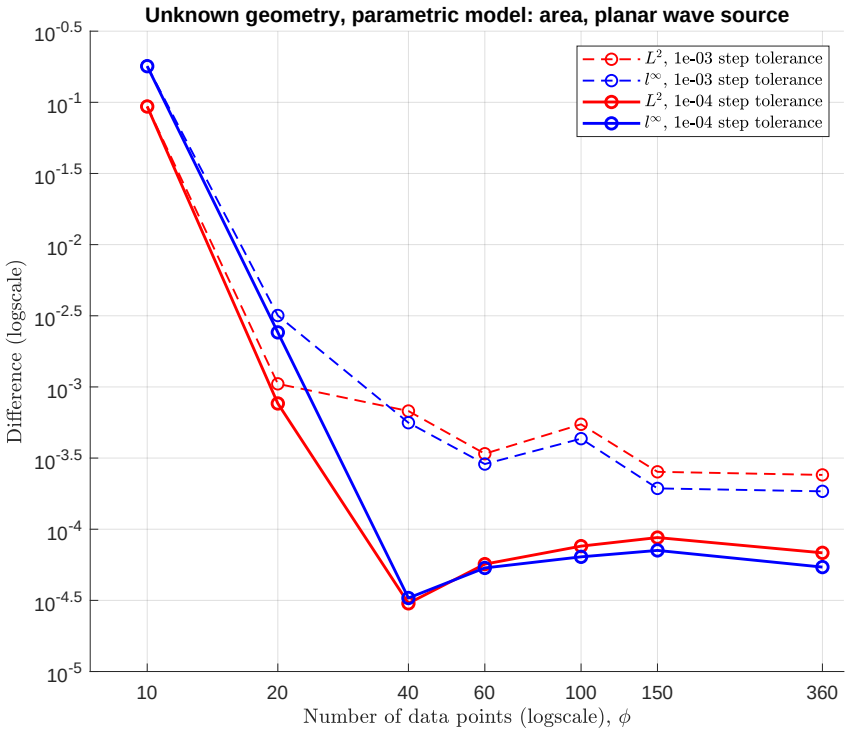


Figure 4.1: The graph of L^2- and $l^\infty-$difference against the number of sampling data points for different step tolerance values.

To research this observation further, the sampling theorem, also called as Nyquist–Shannon sampling theorem, was a crucial aid, which states that the sample rate must be at least twice the frequency of the signal to avert aliasing [22]. Using the Fourier transform of the scattering signal we may identify the important frequencies and subsequently determine the number of needed points. Figure 4.2 shows an example of Fourier transformation of the scattering profile obtained from the shape with 10 shape-controlling points, which is the most complex case among existing in the scope of inverse problem studies. It can be seen that the bandlimit is around 20; therefore, approximately 40 equally distributed data points are sufficient for accurate results. To ensure stability and accuracy, 60 data points were integrated into the framework.

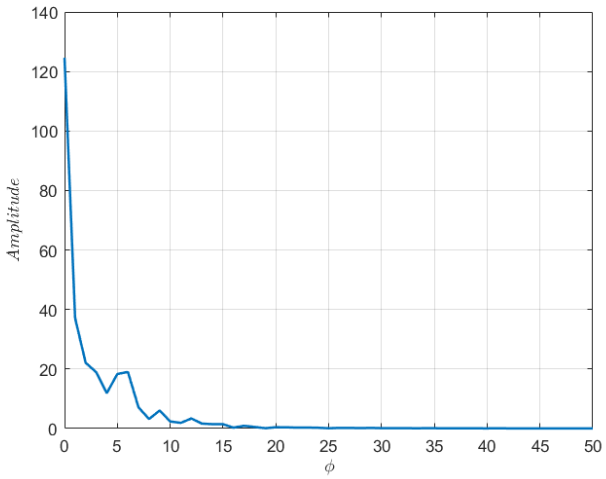
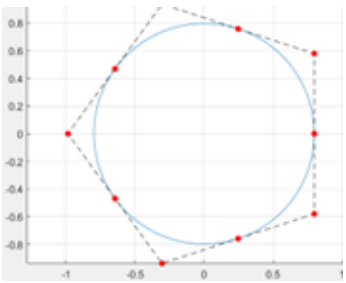
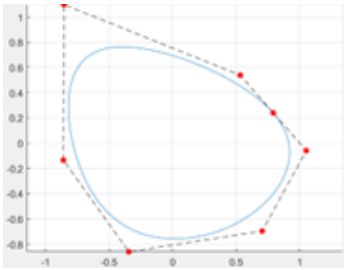


Figure 4.2: Fourier transform of scattering profile of an arbitrary shape with 10 shape-controlling points.

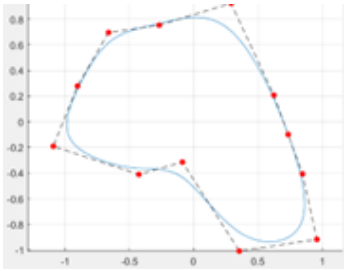
Similarly, it is crucial to identify the relationship between the number of DOFs and the accuracy of the scattering profiles. The optimal numbers of DOFs for each cross-sectional shape shown on Figure 4.3 that are being used for the analysis of inverse problems. The results were obtained by gradually increasing the number of inserted knots and parallelly computing the number of DOFs and relative errors from power and voltage. Fig. 2-4



(a) Circle.



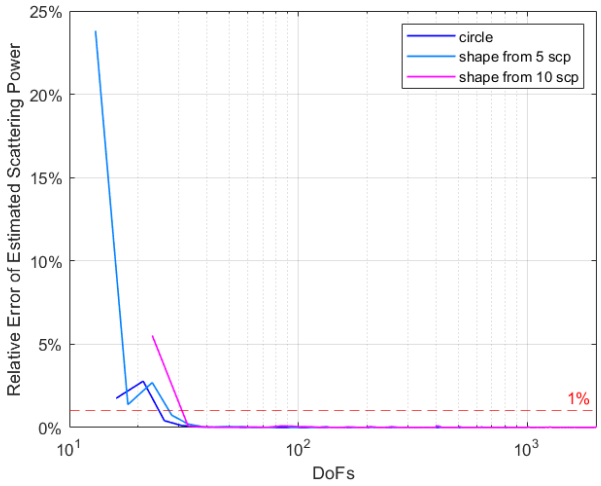
(b) Arbitrary shape from 5 SCPs.



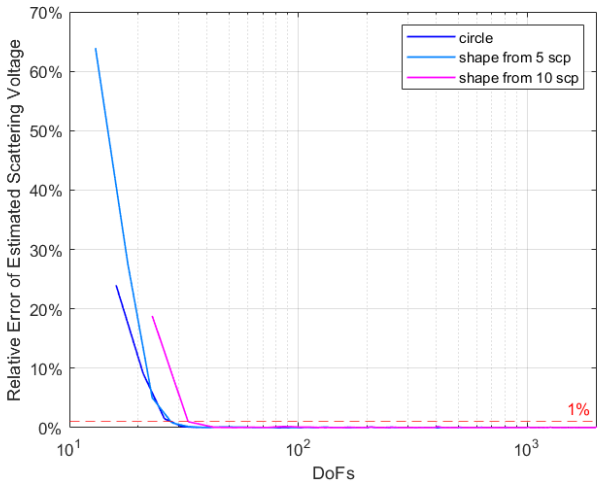
(c) Arbitrary shape from 10 SCPs.

Figure 4.3: Cross-sectional shapes used for inverse problem analysis.

demonstrate the graphs of DOFs analysis. It was calculated that 26, 28, and 33 DOFs are enough to get 1% relative error for circle, shape from 5 shape-controlling points, and shape from 10 shape-controlling points, respectively. To ensure stability and accuracy while using random and unknown shapes, 60 DOFs were integrated into the framework.



(a) Relative error of estimated scattering power against DOFs.



(b) Relative error of estimated scattering voltage against DOFs.

Figure 4.4: DOFs requirement analysis.

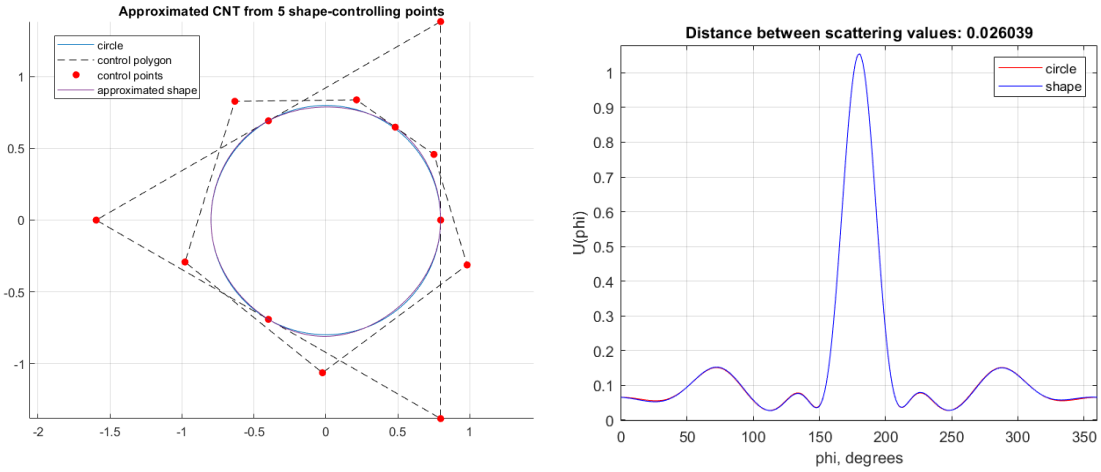
4.2 Optimization Results

For each shape approximation, the forward IGABEM approach was first used to obtain the scattering signal from the true shape for comparison purposes. Then, inverse optimization was executed with unknown variables and constraints as previously discussed for each optimization case. By using the forward approach, we could verify the obtained results. Convergence rates and histories were recorded for all cases. A selected number of cases

from the obtained results is indicatively presented in this chapter while the full range of cases with the corresponding analysis can be found in Appendix A.

4.2.1 Case 1: Unknown Geometry, Known Area

The first optimization problem solved was to identify the shape of a nanotube with known area and scattering profile when illuminated by a given plane wave. In Fig. 4.5a, the blue line represents the true shape, which is a circle. The purple line, which represents the resulting shape, practically coincides with the true shape. Similarly, Fig. 4.5b shows that voltage values at each angle are roughly the same for the true and approximating shape. Consequently, this provides a first verification of the capacity of the employed IGABEM approach to efficiently address the inverse optimization problems.



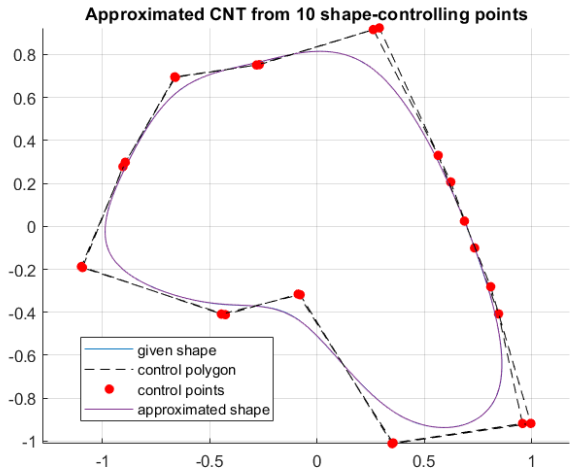
(a) Shape approximation.

(b) Scattering profile.

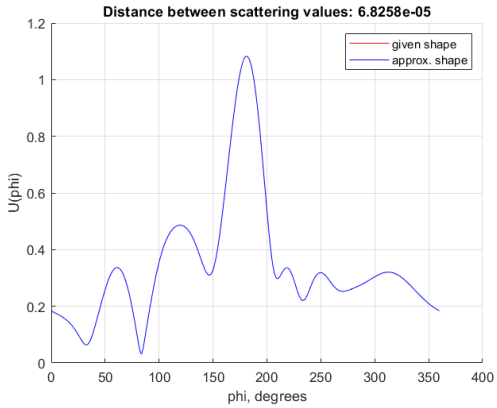
Figure 4.5: Circle identification for given area; scattered signal for the circle illuminated by a plane wave.

Generally, the calculation of the integrals in the objective functions, which resemble the L^2-norm, was performed using Simpson's 1/3 Rule, and the employed optimization method was the previously discussed hybrid approach. However, for test cases, numerical simplifications and/or `fmincon()` was used instead to quickly check the implementations. Figure 4.6 demonstrates the achieved shape for a scattering signal coming from an arbitrary-shaped nanotube. The optimization process required 336 iterations, resulting in an L^2-difference of $6.8258e-05$. Finally, the convergence history for objective function is depicted in Fig. 4.6c with different colors corresponding to the two optimizers used in the hybrid approach. It also confirms that the optimization method converges adequately.

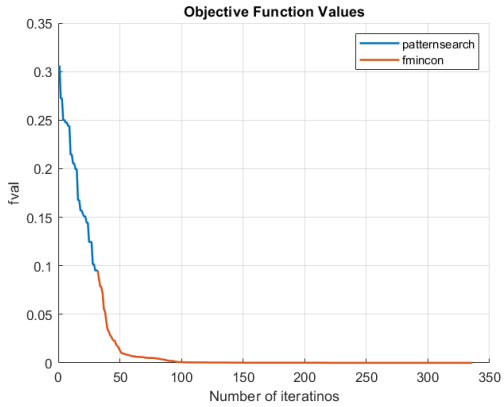
Similar approaches were implemented for scattering signals corresponding to a point source (cylindrical EM waves). Such an example is presented in Fig. 4.7. It can be seen that the optimization algorithm works equally well with point wave sources. Furthermore, this case shows the advantage of the hybrid approach. On the Fig. 4.7c, it is obvious that



(a) Shape approximation.



(b) Scattering profile.



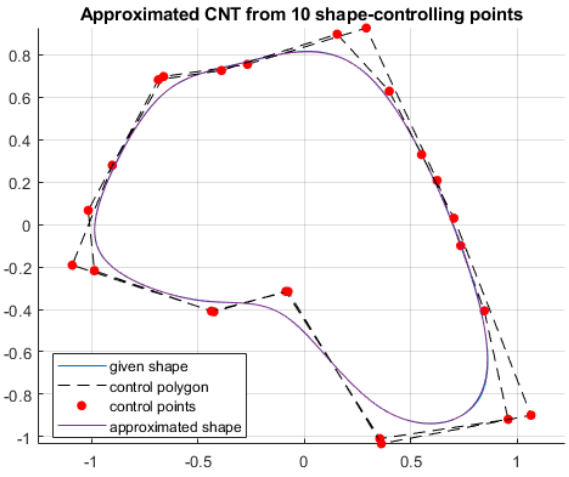
(c) Objective function.

Figure 4.6: Approximation of an arbitrary shape with 10 shape-controlling points and given area. The scattered signal corresponds to a plane wave.

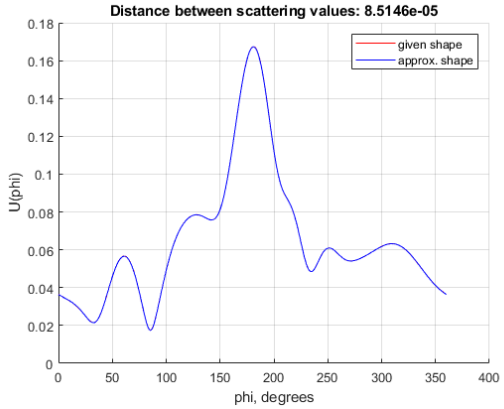
`patternsearch()` was converging slowly, but provided an appropriate initial point for `fmincon()` that was utilized after. The optimization process took only 216 iterations and resulted in an L^2-difference of $8.5146e-05$.

4.2.2 Case 2: Unknown Geometry, Known Circumference

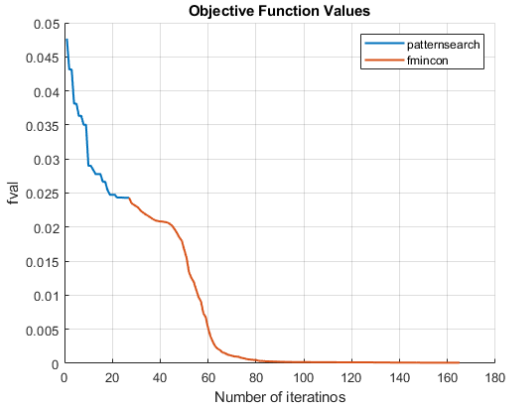
The circumference-constrained inverse optimization problem is similar to the area-constrained one. It follows a very similar implementation with the only differences being the constraint implementation and the employed parametric model. Figures 4.8 and 4.9 demonstrate indicative examples for the solution of this problem when scattering signals correspond to plane and cylindrical EM waves, respectively.



(a) Shape approximation.



(b) Scattering profile.



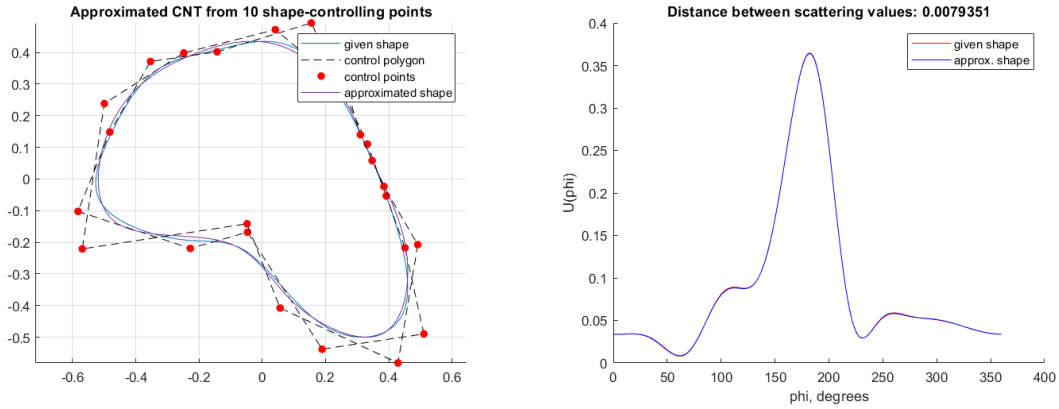
(c) Objective function.

Figure 4.7: Approximation of an arbitrary shape with 10 shape-controlling points and given area. The scattered signal corresponds to a cylindrical wave.

4.2.3 Case 3: Unknown Conductivity, Known Geometry

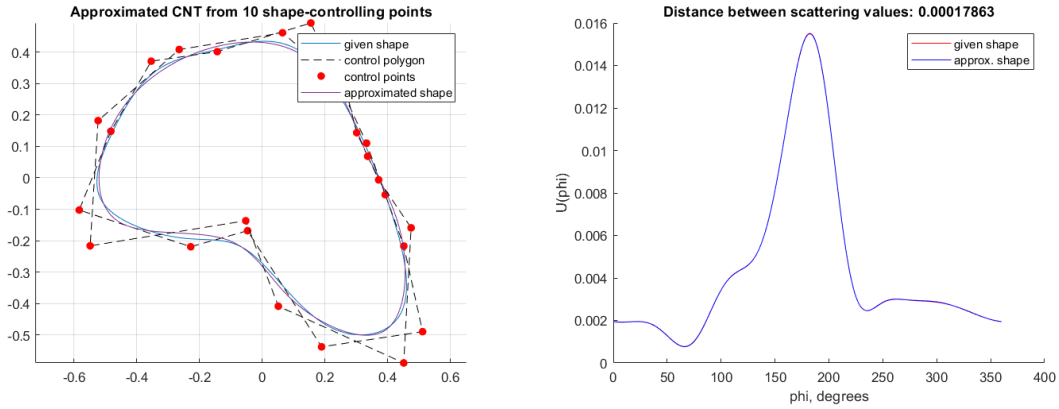
The third inverse problem focuses on material identification. This time, the geometry is known; thus, it is a geometry-constrained inverse optimization problem that aims to identify the complex electric conductivity value as shown in Eq. (2.24). Figure 4.10 depicts the results for this case. As it can be seen, the obtained scattering profile coincides with the given one, indicating a successful solution.

This case is unique from the previous two because it involves significantly less unknown parameters. Thus, the optimization for this problem converged much faster. If previous problems with unknown geometries took up to an hour to complete, this took only up to 5 minutes. Consequently, it is also considered one of the easiest optimization problem among all considered in the scope of this project. Due to this, `patternsearch()` tests were first performed on this case for comparison with `fmincon()` and hybrid approach to rapidly identify with optimization method would be the most time-efficient.



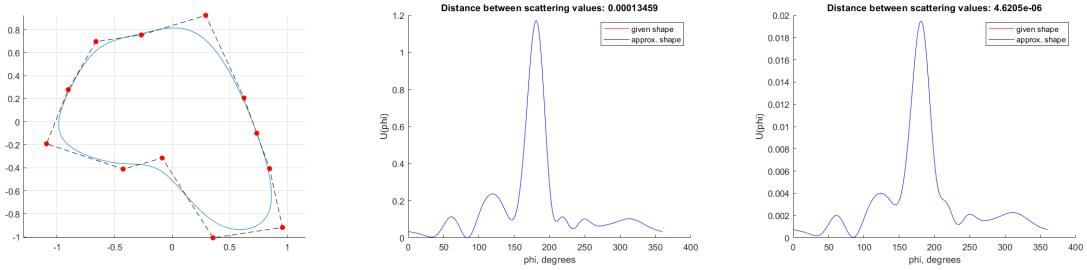
(a) Shape approximation. (b) Scattering profile.

Figure 4.8: Approximation of an arbitrary shape with 10 shape-controlling points and given circumference. The scattered signal corresponds to a plane wave.



(a) Shape approximation. (b) Scattering profile.

Figure 4.9: Approximation of an arbitrary shape with 10 shape-controlling points and given circumference. The scattered signal corresponds to a cylindrical wave.



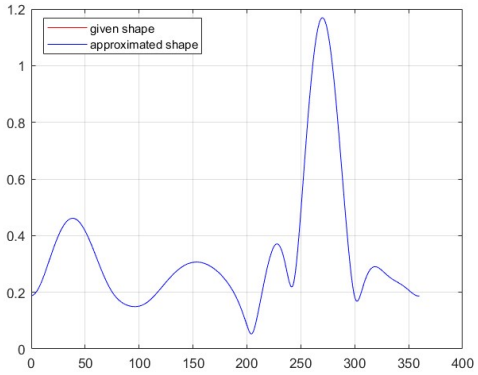
(a) Given shape. (b) Scattering profile from plane wave. (c) Scattering profile from cylindrical wave.

Figure 4.10: Conductivity value estimation and comparison of scattering profiles.

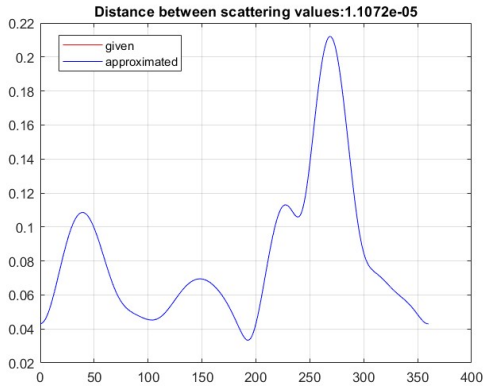
4.2.4 Case 4: Unknown Position, Known Geometry

Similarly to the Case 3, the position problem is geometry-constrained, see also Eq. (2.25). Figure 4.11 depicts the obtained results for this problem. Here, Fig. 4.11a shows the scattering results from the plane wave that is along a $\pi/2$ angle direction of $\pi/2$ angle;

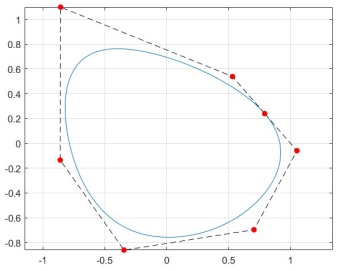
thus, hitting the nanotube at an angle of $3\pi/2$, which can be clearly in the scattering profile. The results are similar when a cylindrical wave is used; see Fig. 4.11b.



(a) Scattering profile from a plane wave forming an $\alpha = \pi/2$ angle with horizontal axis.



(b) Scattering profile from a point source positioned at $(2, \pi/2)$.



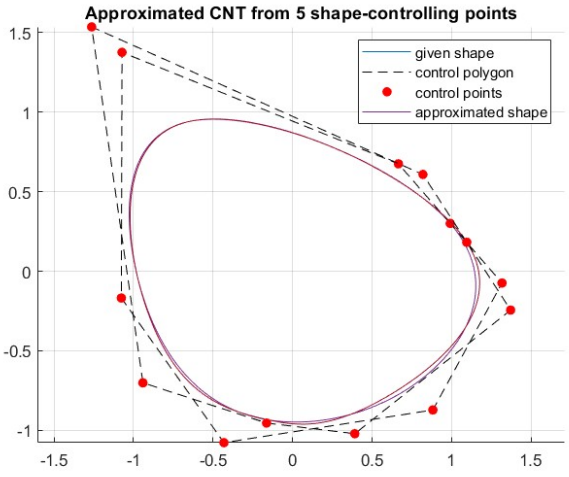
(c) Given shape.

Figure 4.11: Identification of wave characteristics (direction and location of point source) and comparison of the resulting scattering profiles.

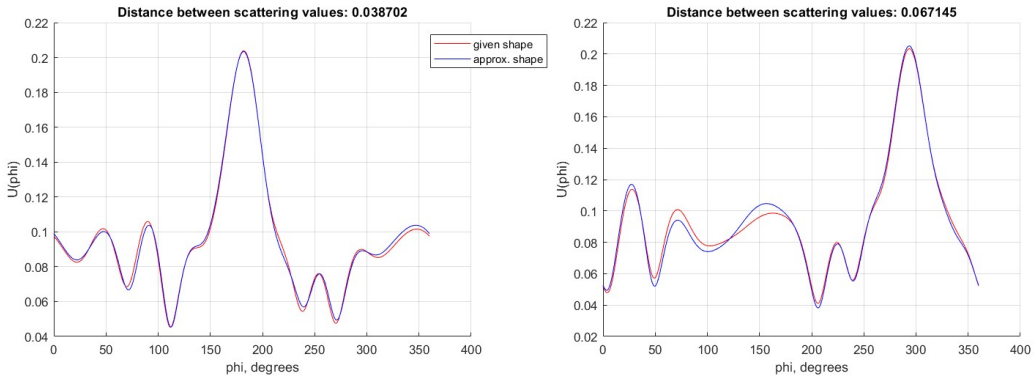
4.2.5 Case 5: Unknown geometry, scattering signals from two independent EM wave sources

Most importantly, we present indicative results for the inverse optimization problem that identifies the boundary shape when two scattering signals are given (from independent EM wave sources). Obviously, the two sources should not coincide or interfere with each other as we then need to employ a completely different approach. As there are no area or circumference constraints, the solution will generally need more time when compared to the previous inverse problems. Moreover, the shape approximation was not as accurate as in the other cases. However, this case was considered the most interesting and potentially useful among all inverse problems during previous semester as minimal information is provided. Normally, during the application of this approach in different devices, we would not be able to identify any information about the shape of the object of interest, but we could scatter the object with the second source from different position. Therefore, this inverse problem's optimization algorithm was further improved resulting in Case 6.

Fig. 4.12 demonstrate the preliminary results for this particular problem. The algorithm for this specific solution was entirely based on `fmincon()`. This case was not extensively tested due to the discovery of Case 6, which is the direct extension of this case but with unknown material properties; thus, it was not tested with hybrid approach optimization.



(a) Shape approximation.



(b) Scattering profiles.

Figure 4.12: Approximation of an arbitrary shape with 5 shape-controlling points; two scattering signal by two separate point sources are used.

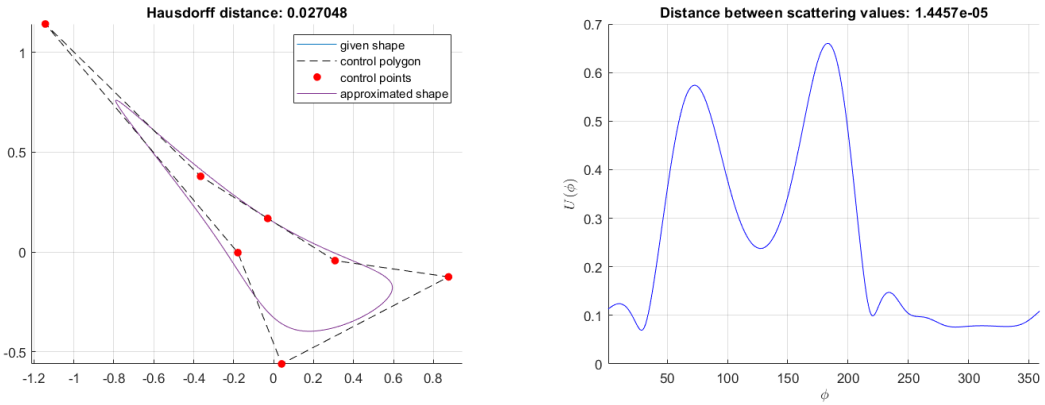
4.2.6 Cases 6 & 7: Unknown geometry and material, scattering signals from two independent EM wave sources

After incorporation of sampling points and DOFs requirements, modified framework was used to analyze the double source inverse problem with unknown geometry and material. This problem is the most practical one as there is even less information provided about unknown object. Additionally, more complex random shapes were tested to ensure framework's stability. One of the most interesting shapes was the one shown in Figure 4.13 as its scattering profile had two big spikes when illuminated horizontally. It can be seen that the MATLAB framework successfully approximated the shape. Additionally, the

computational time was reduced by almost 3 times. Figure 4.14 demonstrates the result for randomly picked arbitrary shape from 8 shape-controlling points, which further proves that the method can be used for more complex shapes.

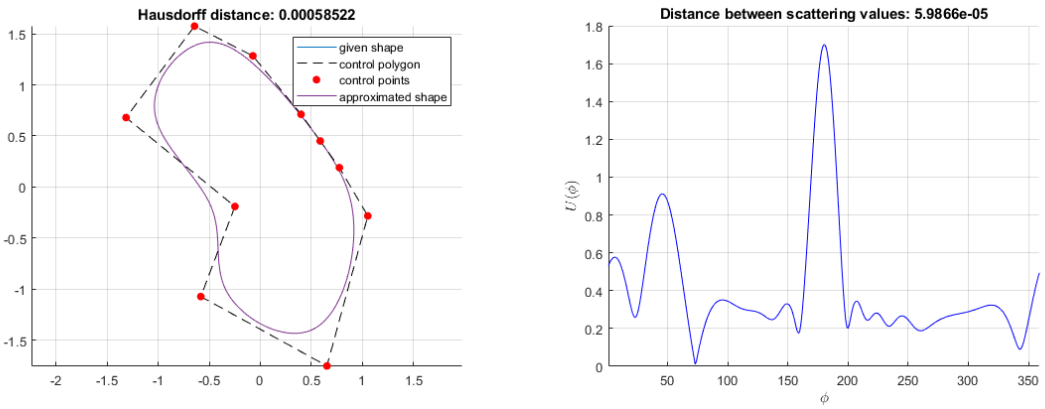
To further improve double source study and introduce real-life constraints, Case 6 code was modified to collect sampling points data from a limited range. Using scattering profiles from two different cylindrical wave sources and given data about signal properties, the goal was to deduce the shape and conductivity by minimizing the difference between computed and measured profiles from a 120° range. Figure 4.15 demonstrates the best results from this study obtained from complex shape from 5 shape-controlling points. Figure 4.15b further shows that although scattering profile difference was only calculated from the given range, it is enough to deduce the entire shape and match profiles.

All of the results for Cases 6 and 7 were collected using the latest framework that utilizes hybrid approach and optimized usage of L^2-difference calculations during objective function calculations. It also covers the usage of optimal sampling points and DOFs



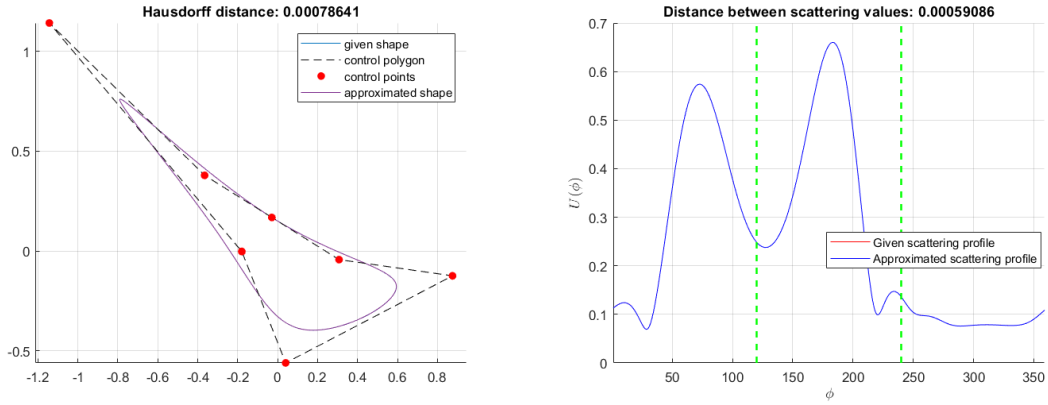
(a) Shape approximation. (b) Scattering profile.

Figure 4.13: Approximation of an intricate arbitrary shape with 5 shape-controlling points illuminated by two plane wave sources. The material in unknown.



(a) Shape approximation. (b) Scattering profile.

Figure 4.14: Approximation of an arbitrary shape with 8 shape-controlling points illuminated by two plane wave sources. The material in unknown.



(a) Shape approximation.

(b) Scattering profile.

Figure 4.15: Limited range approximation of an arbitrary shape with 5 shape-controlling points illuminated by two plane wave sources. The material in unknown.

requirement to reach time-efficient solutions.

Additionally, several optimization runs were performed starting with different initial parameters. They resulted in the same approximations of the shape shown in Fig. 4.15a. This is a strong indication that we have reached the global optimum and unique solutions are generated, at least for the tested shapes.

Finally, in Figs. 4.13, 4.14, and 4.15, it can be seen that given shapes and approximated shapes, as well as scattering profiles, almost overlap each other proving the accuracy of the IGABEM. Thus, the limitations from the previous semester were successfully tackled resulting in an improved framework with higher accuracy, less computational time, and potentially unique solutions.

4.3 Limitations

While this study successfully demonstrates an approach to determining the shape of nanoelements from electromagnetic wave scattering profiles, several limitations should be acknowledged:

- Although the sampling theorem and DOFs analysis results provided with substantial computational time reduction, a single optimization problem might take up to 1.5 hours in MATLAB. This can be a disadvantage in real-life applications, where immediate results might be needed. The problem may be solved by switching the framework's code to a faster programming language, such as C++.

- Sensitivity analyses of the scattering profile to changes in key parameters were not conducted, which limits the ability to analyze the robustness of the approach under varying conditions. Inverse problems were formulated and solved assuming ideal conditions. The developed method does not consider noise and interfering

signals. Additionally, the results were obtained only for vacuum conditions. Further improvements can be made by testing the framework under different real-world conditions and varying key parameters.

- Third, although we ensured to include all the necessary constraints to the optimization problems, the uniqueness of the obtained solutions has not yet been fully confirmed. There is a strong indication that solutions are unique because every optimization led to the same results regardless of its initial parameters. However, this has been tested only for a limited number of cases and results, which implies that it is possible that different parameter sets and therefore shapes may yield similar scattering profiles. Establishing a framework to verify the uniqueness of the solution will be a critical step in the validation of the method in the future.

Chapter 5

Conclusion and Future Work

5.1 Conclusions

This study aimed to develop a method for determining the shape of arbitrary nanoelements from electromagnetic wave scattering profiles, using the IGABEM approach and optimization techniques to minimize the difference between measured and predicted scattering data. The primary objectives were to establish a framework that can reliably reconstruct nanoelement's geometry and other characteristics, such as material properties and position with respect to the source signal. To achieve the objective, EM scattering was formulated, and a method involving the IGABEM approach and incorporating NURBS was implemented in MATLAB. The inverse optimization problems were defined and solved using gradient-based, direct, and hybrid algorithms. The results demonstrate great potential of the proposed method, successfully approximating the shapes and other parameters from scattering profiles in a controlled setting.

Limitations from the previous semester's work were addressed by studying the sampling points and DoFs requirements. As a result, it was identified that 6 times less sampling points and twice as few DoFs are required to obtain accurate results. These results were integrated into the solution of Cases 6 and 7, which involved the identification of the shape of the nanotube without any preliminary information about its geometry and material properties. With older framework, this problem would take several hours to solve. The computational time was effectively reduced, and even with the least favorable initial parameters, it takes up to 1.5 hours.

MATLAB implementation for all seven inverse problems demonstrate a great potential of the proposed methodology, resulting in successful and accurate approximations. Although the current implementation shows great potential, further refinements and addition of advanced numerical techniques are required to adapt the framework for real-world conditions. The limitations include improving computational efficiency to further decrease the optimization time. Additional studies need to be conducted to analyze method's sensitivity to varying key parameters and performance under actual conditions with noise and possible interfering signals.

Despite these challenges, the results largely support the initial hypothesis that nanoelement shapes can be accurately reconstructed from electromagnetic scattering data. The model has demonstrated significant potential for providing precise reconstructions under ideal conditions. While the methodology was applied to a specific set of scattering

profiles, its applicability extends to a broader range of scenarios. Specifically, the inverse problem involving an unknown geometry and material illuminated by two points, as defined by Eq. (2.27), shows great promise for applications in non-invasive scanning and military technologies, particularly when objects of interest are difficult to detect visually.

5.2 Future Work

While this work demonstrates the feasibility of identifying nanotube shapes and other characteristics from EM wave scattering profiles, several opportunities exist to improve and extend this work.

1. The complexity of the modeled nanotube geometries can be extended to match real-world nanotubes that often exhibit surface roughness and defects. Additionally, the methodology can be scaled to identify the shapes of larger micro- and macro-scale objects. Adding these factors can enhance the model's robustness and applicability.

2. Although the current inverse problem approach already consider limited-angle scattering, future work could investigate the impact of noise, incomplete data, and interfering signals. Regularization strategies or machine learning-assisted reconstructions can be implemented to maintain identification accuracy under such conditions.

3. Validating this framework with experimental data can be a crucial step. Experiments can be conducted to obtain measured scattering profiles from fabricated nanotubes to provide a real-world benchmark for the developed model and highlight any necessary model refinements.

4. The methodology developed here will be further studies as part of a master's thesis within the WICAN project. The thesis will focus on military applications of shape identification from EM wave scattering, where practical constraints such as single-source illumination, limited-angle scattering, and other will be incorporated.

5.3 Individual Contributions

Below is the Table 5.1 with individual team member contributions for this project.

Team Member	Contribution
Meruyert Toleuova	Literature review for Background. Literature review for Methodology. Inverse problem formulations. Implementation of optimization techniques. Integration of NURBS and IGABEM. Sampling points requirement analysis. DOFs requirement analysis. Overall refinements of the code. Development of objective functions and optimizations in MATLAB. Development of the MATLAB code for Cases 1-7. Data collection and Results for Cases 1-7. Discussion of results for Cases 1-7. Report preparation.
Nurtas Kaliyev	Literature review for Background and Introduction.

Table 5.1: Team member individual contributions.

Bibliography

[1] K. V. Kostas and V. C., "Optimally shaped nanotubes for field concentration," *Engineering Analysis with Boundary Elements*, vol. 169, Part B, 2024.

[2] N. Iznat, M. Bespayev, Y. Turarov, C. Valagiannopoulos, and K. Kostas, "Shaping quasi-transparent nanotubes into maximally strong em scatterers," *Engineering Analysis with Boundary Elements*, vol. 173, p. 106153, 2025.

[3] R. K. Mishra, J. Cherusseri, E. Allahyari, S. Thomas, and N. Kalarikkal, "Chapter 10 - small-angle light and x-ray scattering in nanosciences and nanotechnology," in *Thermal and Rheological Measurement Techniques for Nanomaterials Characterization* (S. Thomas, R. Thomas, A. K. Zachariah, and R. K. Mishra, eds.), Micro and Nano Technologies, pp. 233–269, Elsevier, 2017.

[4] G. A. Viano, "Ill-posed problems," in *Inverse Problems: An Interdisciplinary Study* (P. C. Sabatier, ed.), pp. 3–60, Springer, 1985.

[5] T. Hughes, J. Cottrell, and Y. Bazilevs, "Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, pp. 4135–4195, 2005.

[6] K. Kostas, M. Fyrillas, C. Politis, A. Ginnis, and P. Kaklis, "Shape optimization of conductive-media interfaces using an IGA-BEM solver," *Computer Methods in Applied Mechanics and Engineering*, vol. 340, pp. 600–614, 2018.

[7] K. Kostas, C. Politis, I. Zhanabay, and P. Kaklis, "A physics-informed parametrization and its impact on 2D IGABEM analysis.," *Engineering with Computers*, vol. 40, pp. 3663—3682, 2024.

[8] J. Dölz, S. Schöps, F. Wolf, and S. Kruz, "Isogeometric boundary elements in electromagnetism: Rigorous analysis, fast methods, and examples," *arXiv preprint arXiv:1807.03097*, 2019.

[9] Y. Ma, X. Zhang, and S. Chen, "Broadband electromagnetic scattering analysis with isogeometric boundary element method accelerated by frequency-decoupling and model order reduction techniques," *International Conference on Computational & Experimental Engineering and Sciences*, vol. 27, no. 2, pp. 60–73, 2023.

[10] K. Kostas and V. Costas, "Optimally shaped nanotubes for field concentration," *Journal of Applied Physics*, vol. 134, p. 045102, July 2023.

[11] A. A., A. R., F. A., C. V., and K. K., "Maximal electromagnetic coupling between arbitrary–shaped nanotubes," *Partial Differential Equations in Applied Mathematics (submitted)*, 2025. Manuscript submitted for review.

[12] J. C. Araújo C., C. Engström, and E. Wadbro, "Shape optimization for the strong routing of light in periodic diffraction gratings," *Journal of the Optical Society of America B*, vol. 38, no. 12, pp. 3451–3462, 2021.

[13] E. F. Knott, J. F. Schaeffer, and M. T. Tuley, *Radar Cross Section*. Raleigh, NC: SciTech Publishing, 2 ed., 2004.

[14] M. Pastorino, *Microwave Imaging*. Hoboken, NJ: John Wiley & Sons, 2010.

[15] S. Hughes, "Current, continuity equation, resistance, Ohm's law," 2005. [Online; accessed 22-November-2008].

[16] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions: with Formulas, Graphs, and Mathematical Tables*. Dover, 1965.

[17] L. Piegl and W. Tiller, *The Nurbs Book, 2nd Edition*. Springer Verlag, 1997.

[18] D. Wang, Q. Dongliang, and X. Li, "Superconvergent isogeometric collocation method with greville points," *Computer Methods in Applied Mechanics and Engineering*, 2021.

[19] A. Canós Valero, H. K. Shamkhi, A. Kupriianov, T. Weiss, A. Pavlov, D. Redka, V. Bobrovs, Y. Kivshar, and A. Shalin, "Superscattering emerging from the physics of bound states in the continuum," *Nature Communications*, vol. 14, 08 2023.

[20] Z. Ruan and S. Fan, "Superscattering of light from subwavelength nanostructures," *Phys. Rev. Lett.*, vol. 105, p. 013901, Jun 2010.

[21] J. Cottrell, T. Hughes, and A. Reali, "Studies of refinement and continuity in isogeometric structural analysis," *Computer Methods in Applied Mechanics and Engineering*, vol. 196, no. 41, pp. 4160–4183, 2007.

[22] A. Jerri, "The shannon sampling theorem—its various extensions and applications: A tutorial review," *Proceedings of the IEEE*, vol. 65, no. 11, pp. 1565–1596, 1977.

[23] MathWorks, Inc., "fmincon - constrained nonlinear optimization," 2024. Accessed: 2024-11-26.

[24] MathWorks, Inc., "patternsearch - pattern search optimization," 2024. Accessed: 2024-11-26.

[25] R. H. Byrd, J. C. Gilbert, and J. Nocedal, "A trust region method based on interior point techniques for nonlinear programming," *Mathematical Programming*, vol. 89, no. 1, pp. 149–185, 2000.

[26] T. F. Coleman and Y. Li, "An interior, trust region approach for nonlinear minimization subject to bounds," *SIAM Journal on Optimization*, vol. 6, pp. 418–445, 1996.

[27] T. F. Coleman and Y. Li, "On the convergence of reflective newton methods for large-scale nonlinear minimization subject to bounds," *Mathematical Programming*, vol. 67, no. 2, pp. 189–224, 1994.

[28] M. J. D. Powell, "A fast algorithm for nonlinearly constrained optimization calculations," in *Numerical Analysis* (G. A. Watson, ed.), vol. 630 of *Lecture Notes in Mathematics*, pp. 144–157, Springer-Verlag, 1978.

[29] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering, New York: Springer, 2 ed., 2006.

[30] MathWorks, Inc., "What is direct search?," 2024. Accessed: 2024-11-26.

[31] MathWorks, Inc., "How pattern search polling works," 2024. Accessed: 2024-11-26.

[32] M. H, "NURBS Toolbox by D.M. Spink." https://www.mathworks.com/matlabcentral/fileexchange/26390-nurbs-toolbox-by-d-m-spink, 2025. MATLAB Central File Exchange. Retrieved April 26, 2025.

[33] Nazarbayev University, "Waves interaction with carbon nanotubes (wican)." https://research.nu.edu.kz/en/projects/waves-interaction-with-carbon-nanotubes-wican, 2025. Accessed: 2025-04-28.

[34] MathWorks, "Optimization toolbox documentation." https://www.mathworks.com/help/optim/index.html, 2025. Accessed: 2025-04-28.

[35] MathWorks, "Global optimization toolbox documentation." https://www.mathworks.com/help/gads/index.html, 2025. Accessed: 2025-04-28.

Appendices

Appendix A

Visual Results

A.1 Case 1

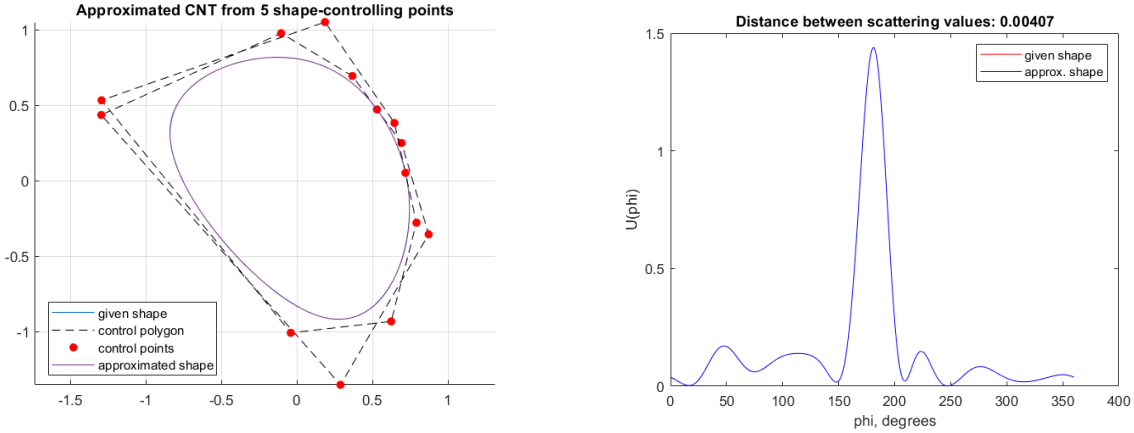
Figs. A.1 and A.2 represent the area-constrained approximation results for various shapes illuminated by planar wave. Figs. A.3, A.4, and A.5 demonstrate the similar results for point wave.

A.2 Case 2

Figs. A.6 and A.7 represent the alternative shape approximation for the second inverse problem with circumference constraint.

A.3 Study for Sampling Data Points

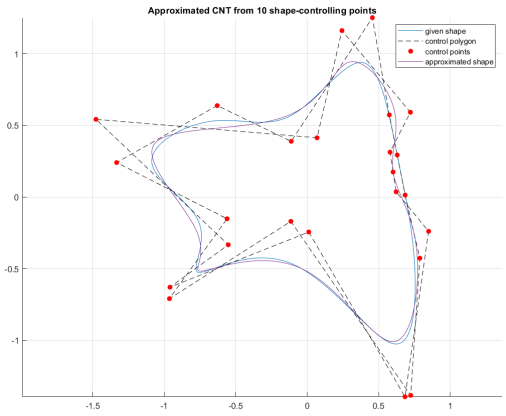
Figs. A.8-A.14 shows the results for approximation with varying number of sampling data points.



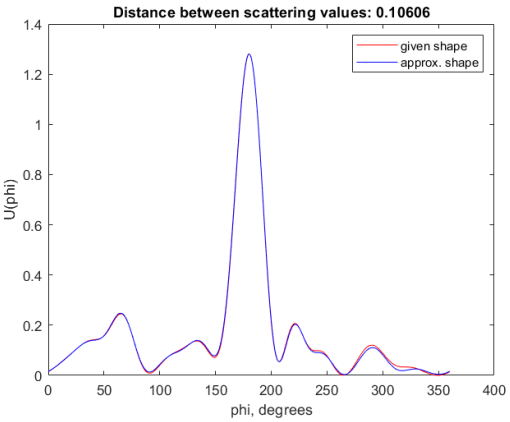
(a) Shape approximation.

(b) Scattering profile.

Figure A.1: Approximation of an arbitrary shape, consisting of 5 shape-controlling points, scattered by a planar wave source, with area parametric model.

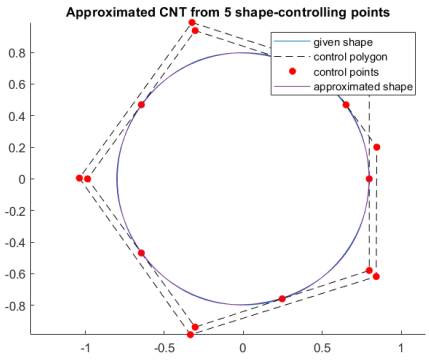


(a) Shape approximation.

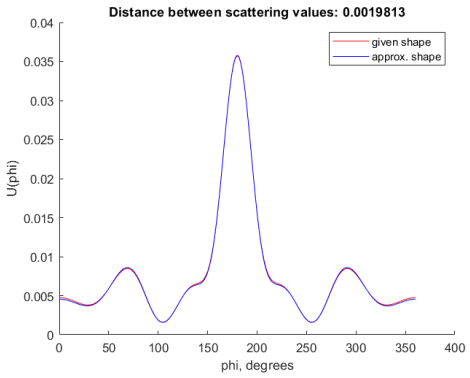


(b) Scattering profile.

Figure A.2: Approximation of an arbitrary shape, consisting of 10 shape-controlling points, scattered by a planar wave source, with area parametric model.

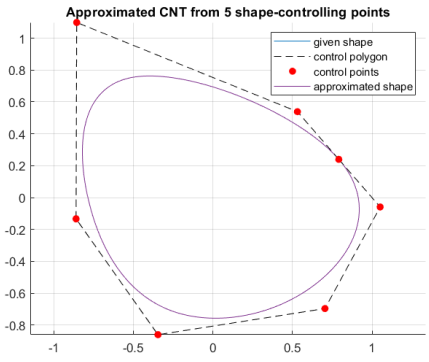


(a) Shape approximation.

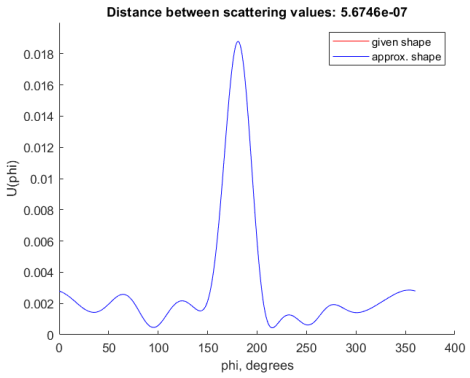


(b) Scattering profile.

Figure A.3: Approximation of a circle scattered by a point wave source, with area parametric model.

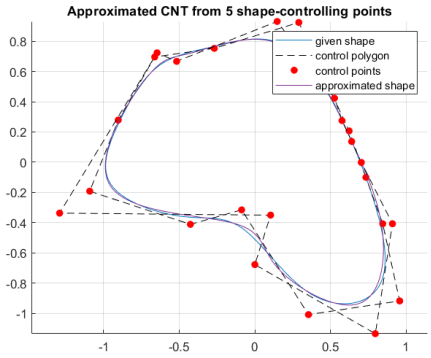


(a) Shape approximation.

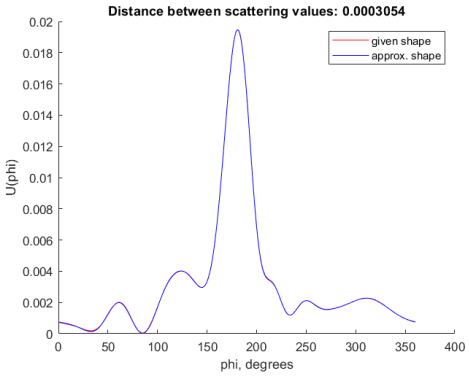


(b) Scattering profile.

Figure A.4: Approximation of an arbitrary shape, consisting of 5 shape-controlling points, scattered by a point wave source, with area parametric model.

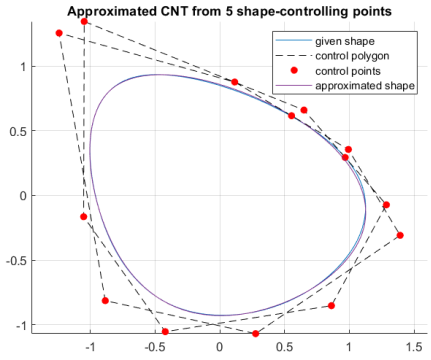


(a) Shape approximation.

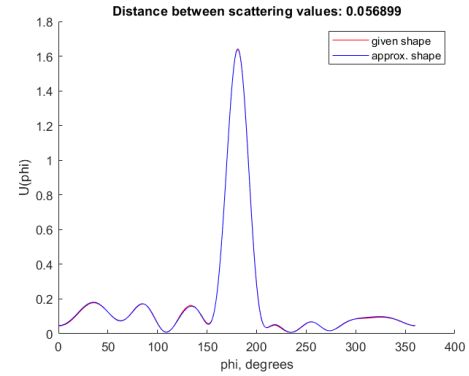


(b) Scattering profile.

Figure A.5: Approximation of an arbitrary shape, consisting of 10 shape-controlling points, scattered by a point wave source, with area parametric model.

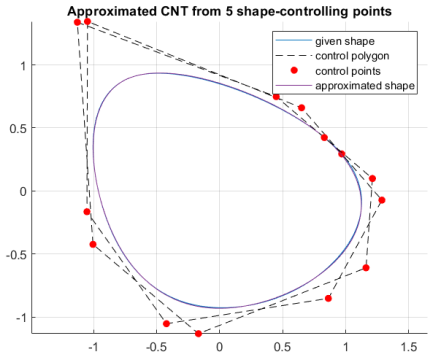


(a) Shape approximation.

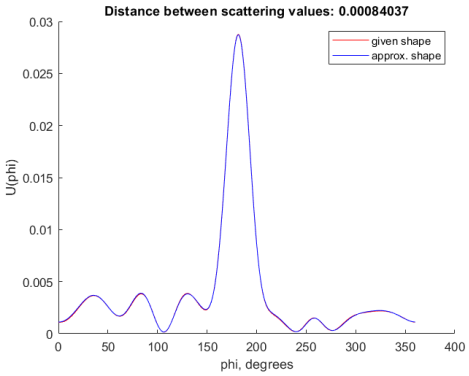


(b) Scattering profile.

Figure A.6: Approximation of an arbitrary shape, consisting of 5 shape-controlling points, scattered by a planar wave source, with circumference parametric model.

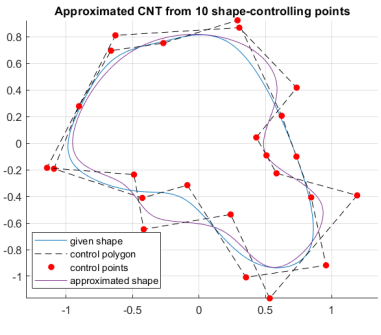


(a) Shape approximation.

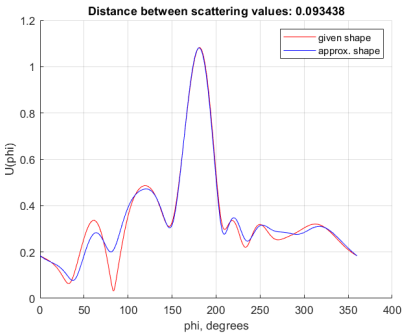


(b) Scattering profile.

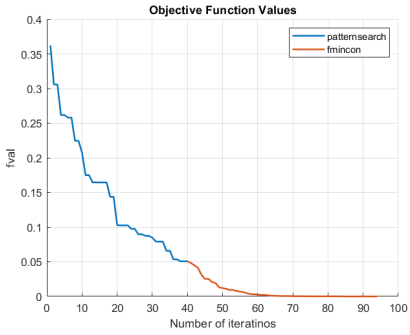
Figure A.7: Approximation of an arbitrary shape, consisting of 5 shape-controlling points, scattered by a point wave source, with circumference parametric model.



(a) Shape approximation.

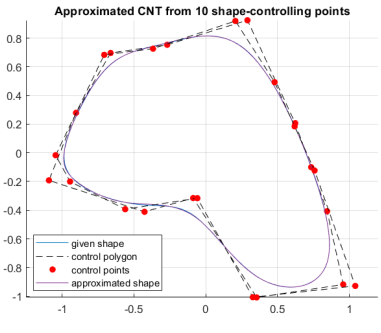


(b) Scattering profile.

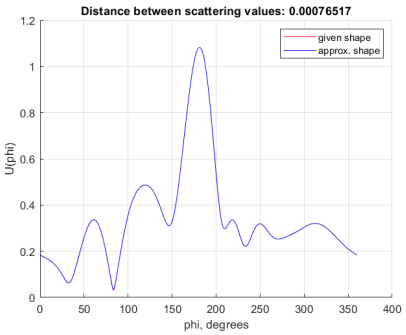


(c) Objective function.

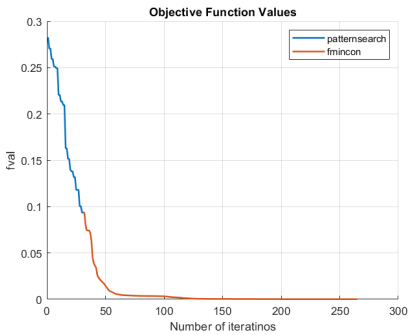
Figure A.8: Unknown shape with area parametric model solved with $\phi = 10$.



(a) Shape approximation.

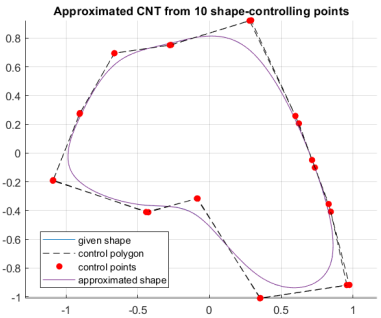


(b) Scattering profile.

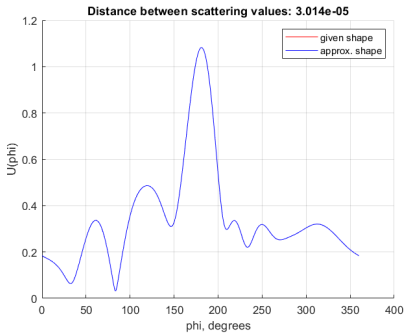


(c) Objective function.

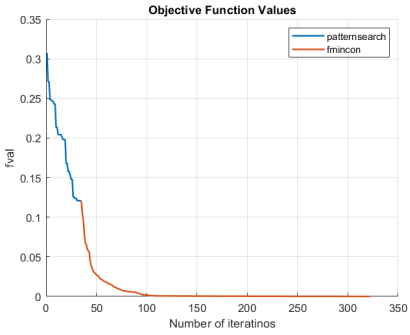
Figure A.9: Unknown shape with area parametric model solved with $\phi = 20$.



(a) Shape approximation.

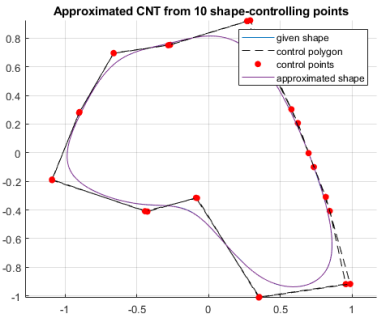


(b) Scattering profile.

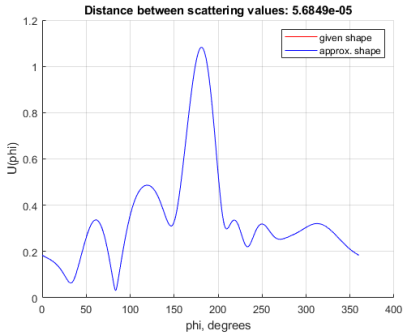


(c) Objective function.

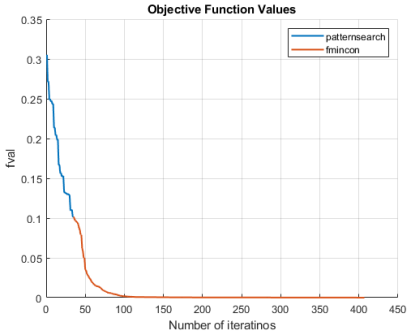
Figure A.10: Unknown shape with area parametric model solved with $\phi = 40$.



(a) Shape approximation.

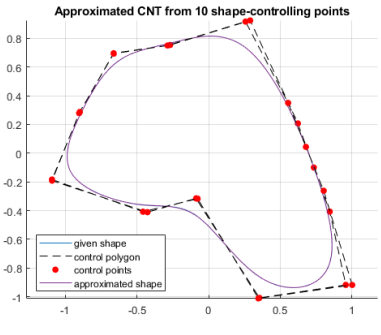


(b) Scattering profile.

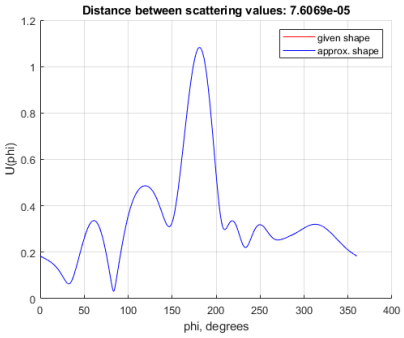


(c) Objective function.

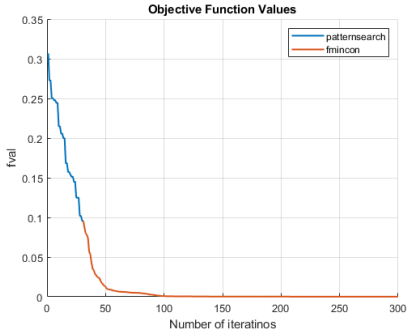
Figure A.11: Unknown shape with area parametric model solved with $\phi = 60$.



(a) Shape approximation.

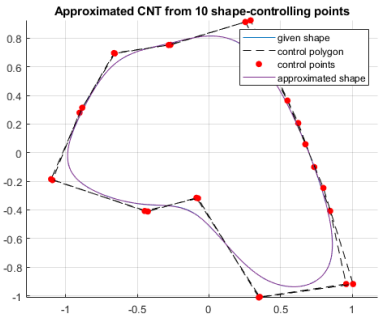


(b) Scattering profile.

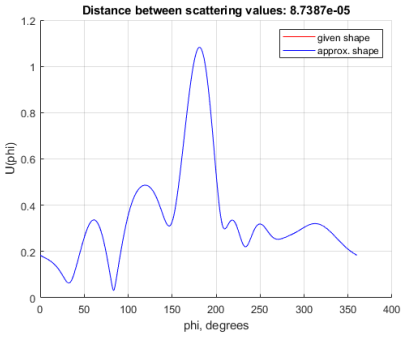


(c) Objective function.

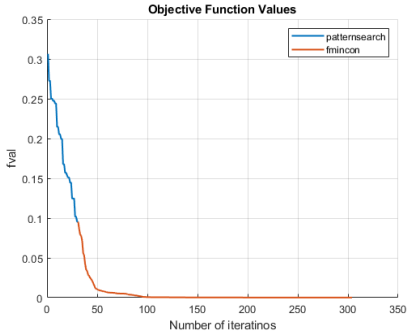
Figure A.12: Unknown shape with area parametric model solved with $\phi = 100$.



(a) Shape approximation.

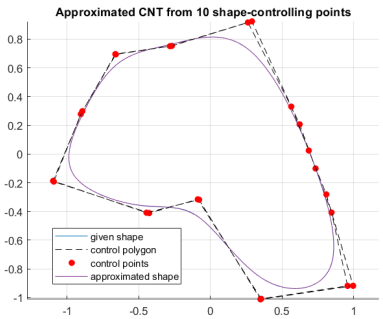


(b) Scattering profile.

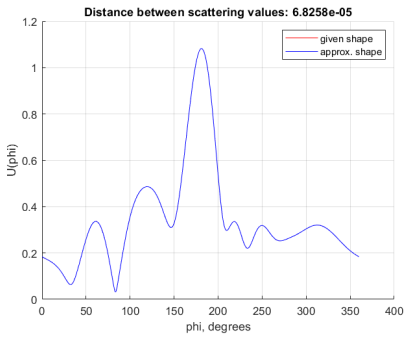


(c) Objective function.

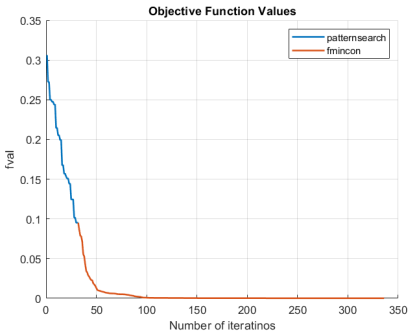
Figure A.13: Unknown shape with area parametric model solved with $\phi = 150$.



(a) Shape approximation.



(b) Scattering profile.



(c) Objective function.

Figure A.14: Unknown shape with area parametric model solved with $\phi = 360$.

Appendix B

Source Code

B.1 Implementation

Below is the latest source code for the limited sampling point range inverse problem as an example.

```
clear
close all

% addpath C: path to nurbs toolbox

% inputs and characteristics
angle = [0, 45];
wavelength = 1;
sigma = 0.001 - 0.01i; %this is the tube's complex electrical conductivity
integration_step = 1e-4; %step used for Simpson's integration
distance = 2;
phi_step_val = 1;
A = 2;
DoFs = 60;

% given unknown shape
% c = create_circle(sqrt(A/pi),[0,0],3,true);
% figure(1)
% hold on
% grid on
% nrbctrlplot(c);
% geom = convertNRB(c);
% knots_in = ceil((DoFs-geom.number)/(length(unique(geom.knots))-1));
% if knots_in<1
%     knots_in=0;
% end
% geom_compute = insertKnots(geom,knots_in);
%params = [0.6557    0.8491    0.6787    0.7431    0.6555; 0.0357    0.9340
0.7577    0.3922    0.1712];
params = [0.004634224134067,0.817303220653433,0.084435845510910,0.259870402850654,0.43141382
%%%%[0.228976968716819,0.152378018969223,0.538342435260057,0.078175528753184,0.10665277018058
%params = [0.6557 0.8491 0.6787 0.7431 0.6555 0.7060 0.2769 0.0971 0.6948 0.9502; 0.0357 0.9
geom = freeform_geom(params,A);
cen = curve_centroid(geom);
geom.coefs(1,:) = geom.coefs(1,:)-cen(1);
```

```
geom.coefs(2,:) = geom.coefs(2,:)-cen(2);
figure(1)
hold on
grid on
nrbctrlplot(convertNRB(geom));
knots_in = ceil((DoFs-geom.number)/(length(unique(geom.knots))-1));
if knots_in<1
    knots_in=0;
end
geom_compute = insertKnots(geom,knots_in);
e_geom1 = igabem_one_tube_gpu(geom_compute,wavelength,sigma,integration_step,angle(1)*pi/180);
U_given1 = scat_phi_calc_GPU(geom_compute,e_geom1,wavelength,1,integration_step);
U_given1_120 = U_given1(:,121:241);
e_geom2 = igabem_one_tube_gpu(geom_compute,wavelength,sigma,integration_step,angle(2)*pi/180);
U_given2 = scat_phi_calc_GPU(geom_compute,e_geom2,wavelength,1,integration_step);
U_given2_120 = U_given2(:,121:241);
t = 0:.001:1;
p1 = fnval(geom,t);

% optimization parameters
coefs = [1,repmat([4,2],1,floor((361-2)/2)),4,1];
coefs_120 = [1,repmat([4,2],1,floor((121-2)/2)),4,1];
max_r = 2;
nParams = 5;
r_p = rand(2,nParams);
s_p = 0.01*rand(2,1);

tic
%optimization
%obj_fun_double_sigma([], max_r, wavelength, angle*pi/180, phi_step_val, [U_given1(2,1:phi_step_va
obj_fun_double_sigma_60([], max_r, wavelength, angle*pi/180, phi_step_val, [U_given1_120(2,1:phi_s

%opts = optimoptions('patternsearch','Display','iter','MaxFunctionEvaluations',400);
opts = optimoptions('patternsearch','Display','iter','MaxFunctionEvaluations',3000);
%[x_p,fval_p,exitflag_p,output_p] = patternsearch(@obj_fun_double_sigma,[s_p r_p],[],[],[],[],zero
[x_p,fval_p,exitflag_p,output_p] = patternsearch(@obj_fun_double_sigma_60,[s_p r_p],[],[],[],[],ze
opts = optimoptions('fmincon','Display','iter','Algorithm','sqp','MaxFunctionEvaluations',10000);
%[x,fval,exitflag,output] = fmincon(@obj_fun_double_sigma,x_p,[],[],[],[],zeros(2,nParams+1),ones(
[x,fval,exitflag,output] = fmincon(@obj_fun_double_sigma_60,x_p,[],[],[],[],zeros(2,nParams+1),one
optsigma = x(1,1)-x(2,1)*1i;
optparams = x(:,2:end);
optgeom = freeform_geom(optparams,max_r);
cen = curve_centroid(optgeom);
optgeom.coefs(1,:) = optgeom.coefs(1,:)-cen(1);
optgeom.coefs(2,:) = optgeom.coefs(2,:)-cen(2);
knots_in = ceil((DoFs-optgeom.number)/(length(unique(optgeom.knots))-1));
if knots_in<1
```

```
    knots_in=0;
end
optgeom_compute = insertKnots(optgeom,knots_in);
toc

%results
e = igabem_one_tube_gpu(optgeom_compute,wavelength,optsigma,integration_step,angle(1)*pi/180
U = scat_phi_calc_GPU(optgeom_compute,e,wavelength,1,integration_step);
U_60 = U(:,121:241);
%compule L2-norm between signals
%L2norm = sqrt(1/3*coefs*((U_given1(2,:)-U(2,:)).^2)');
L2norm = sqrt(1/3*coefs_120*((U_given1_120(2,:)-U_60(2,:)).^2)');
%compute Hausdorff distance between shapes
p2 = fnval(optgeom_compute,t);
DH = HausdorffDist(p1',p2');

figure(1)
nrbctrlplot(convertNRB(optgeom));
legend('given shape','control polygon','control points','approximated shape');
title(['Hausdorff distance: ', num2str(DH)]);
figure(2)
hold on
grid on
plot(U_given1(1,:),U_given1(2,:),'r-');
%plot(U_given1_120(1,:),U_given1_120(2,:),'r-');
xlabel('$\phi$','Interpreter','latex')
%xlim([120,240]);
xlim([1,359]);
ylabel('$U(\phi)$','Interpreter','latex')
ylim([0,1e-4])
%plot(U(1,:),U(2,:),'b-');
plot(U_60(1,:),U_60(2,:),'b-');
title(['Distance between scattering values: ', num2str(L2norm)]);

function [res] = obj_fun_double_sigma_60(x, max_r_val, wavelength_val, angle_vals, phi_step_v
persistent max_r;
persistent signals;
persistent wavelength;
persistent angles;
persistent phi_step;
if nargin==6
    max_r = max_r_val;
    wavelength = wavelength_val;
    angles = angle_vals;
    phi_step = phi_step_val;
    signals = signal_vals;
else
    sigma = x(1,1)-x(2,1)*1i;
```

```
    DoFs = 60;
    integration_step = 1e-4;
    params = x(:,2:end);
    geom = freeform_geom(params,max_r); %change max_r accordingly
    knots_in = ceil((DoFs-geom.number)/(length(unique(geom.knots))-1));
    if knots_in<1
        knots_in=0;
    end
    geom = insertKnots(geom,knots_in);
    res = 0;
    for i=2:-1:1
        e_geom = igabem_one_tube_gpu(geom,wavelength,sigma,integration_step,angles(i));
        U = scat_phi_calc_GPU(geom,e_geom,wavelength,phi_step,integration_step);
        U = U(:,121:241);
        res = res+norm(signals(i,:)-U(2,:),2);
    end
end

function [geom] = freeform_geom(params,max_r)
geom = [];
if isempty(params(params>1|params<0)) && max_r>0
    n = length(params);
    r = max_r*(0.99*params(1,:)+0.01);
    phi = (params(2,:)+(0:n-1))*2*pi/n;
    ctrl_pts = [r.*cos(phi);r.*sin(phi)];
    ctrl_pts = [ctrl_pts(:,end),ctrl_pts,ctrl_pts(:,1:2)];
    geom = nrbmak(ctrl_pts,0:1/(n+6):1); %assume cubics with uniform knots
    geom = nrbclamp(geom);
    %normalize knot vector
    geom.knots = (geom.knots-geom.knots(1))./(geom.knots(end)-geom.knots(1));
    geom = convertNRB(geom);
end

end

function [curve_rep2] = convertNRB(curve_rep1)
%convert from MATLAB's rational spline curve format to NURBS toolbox format
%and vice versa
curve_rep2 = [];
if curve_rep1.form == "rB"
    curve_rep2 = nrbmak([curve_rep1.coefs(1:2,:);zeros(1,curve_rep1.number);curve_rep1.coefs(3,:)]
elseif curve_rep1.form == "B-NURBS"
    curve_rep2 = rsmak(curve_rep1.knots,curve_rep1.coefs([1:2,4],:));
end
end

function new_curve = insertKnots(curve, n)
%add n knots for each non-zero interval
```

```
k = curve.order;
uniqueKnots=unique(curve.knots(k:end-k+1));
l = length(uniqueKnots)-1;
knotsIn = zeros(1,l*n);
for i=1:l
    step = (uniqueKnots(i+1)-uniqueKnots(i))/(n+1);
    for j=1:n
        knotsIn((i-1)*n+j) = uniqueKnots(i)+j*step;
    end
end
   new_curve = fnrfn(curve,knotsIn);
end

function [e] = igabem_one_tube_gpu(geometry,lambda,sigma_val,step,angle)
if strcmp(geometry.form,'B-NURBS')
    geometry = convertNRB(geometry);
end
k0 = 2*pi/lambda;
h0 = 120*pi;
t0=0:step:1;
nPoints = length(t0);
greville = aveknt(geometry.knots,geometry.order);
greville = shift_greville(greville,t0);
DoFs = geometry.number;
N = spcol(geometry.knots,geometry.order,greville);
RBases = gpuArray((geometry.coefs(3,:).*N)./(N*geometry.coefs(3,:)')); %Rational bases @ grev
Nt = spcol(geometry.knots,geometry.order,t0);
Rt = (geometry.coefs(3,:).*Nt)./(Nt*geometry.coefs(3,:)');
fnvals = fntlr(geometry,2,t0);
coefs = gpuArray([1, repmat([4,2],1,floor((nPoints-2)/2)), 4, 1]);
LHS = gpuArray(zeros(DoFs,DoFs));
b = gpuArray(zeros(DoFs,1));
fngvals = fntlr(geometry,1,greville);
for i=1:DoFs
    g = fngvals(1:2,i);
    vals = gpuArray(Green(g,fnvals(1:2,:),lambda).*Rt'.*sqrt(fnvals(3,:).^2+fnvals(4,:).^2))
    LHS(i,:) = gpuArray(1i*k0*h0*sigma_val*step/3*coefs*transpose(vals));
    b(i) = gpuArray(E_back_theta(g,lambda,angle));
end
e_in_b = (RBases+LHS)\b;
e = rsmak(geometry.knots,[transpose(e_in_b).*geometry.coefs(3,:);geometry.coefs(3,:)]);


function [g] = shift_greville(g,t)
m = (g(1)+g(end))/2;
for j=1:length(g)
    tv = t(abs(g(j)-t)<1e-7);
    if ~isempty(tv)
```

```
        if tv<=m
            g(j)=g(j)+1e-6;
        else
            g(j)=g(j)-1e-6;
        end
    end
end
end

end

function [U_of_phi] = scat_phi_calc_GPU(geom, e, wavelength, step_phi, step)
t=geom.knots(1):step:geom.knots(end);
phi = 0:step_phi:360;
e_vals = gpuArray(fnval(e,t));
p = gpuArray(fntlr(geom,2,t));
coefs = gpuArray([1, repmat([4,2],1,floor((length(t)-2)/2)), 4, 1]);
U_of_phi = [phi;zeros(1,length(phi))];
for i=1:length(phi)-1
    U_of_phi(2,i) = abs(step/3*coefs*(exp(-1i*2*pi/wavelength*(sqrt(p(1,:).^2+p(2,:).^2)).*cos(phi
end
U_of_phi(2,end) = U_of_phi(2,1);
end

function [res] = Green(p0,p,lambda)
k0 = 2*pi/lambda;
k = 2; %2nd kind
nu = 0; %0 order
H0 = besselh(nu,k,k0*sqrt((p(1,:)-p0(1)).^2+(p(2,:)-p0(2)).^2));
res = -1i/4*H0;
end

function [res] = E_back_theta(p,lambda,theta)
        k0 = 2*pi/lambda*[cos(theta),sin(theta)];
        res = exp(-1i*k0*p);
end

function [geom,c] = freeform_geom_withArea(params,max_r,area0)
geom = [];
c = [0,0];
area_tol=1e-4;
diff_tol=1e-6;
if isempty(params(params>1|params<0)) && max_r>0
    n = length(params);
    ctrl_pts = compute_ctrl_pts(params,max_r);
    tmpgeom = nrbmak(ctrl_pts,0:1/(n+6):1); %assume cubics with uniform knots
    tmpgeom = nrbclamp(tmpgeom);
    area = curve_area(tmpgeom);
```

```
    while abs(area-area0)>area_tol
        pts1 = compute_ctrl_pts(params,max_r-diff_tol/2);
        pts2 = compute_ctrl_pts(params,max_r+diff_tol/2);
        g1 = nrbmak(pts1,0:1/(n+6):1);
        g2 = nrbmak(pts2,0:1/(n+6):1);
        g1 = nrbclamp(g1);
        g2 = nrbclamp(g2);
        darea = (curve_area(g2)-curve_area(g1))/diff_tol;
        max_r = max_r - (area-area0)/darea;
        ctrl_pts = compute_ctrl_pts(params,max_r);
        if any(isnan(ctrl_pts),'all') || any(isinf(ctrl_pts),'all')
            return;
        end
        tmpgeom = nrbmak(ctrl_pts,0:1/(n+6):1); %assume cubics with uniform knots
        tmpgeom = nrbclamp(tmpgeom);
        area = curve_area(tmpgeom);
    end
    geom = tmpgeom;
    %normalize knot vector
    geom.knots = (geom.knots-geom.knots(1))./(geom.knots(end)-geom.knots(1));
    geom = convertNRB(geom);
    c = curve_centroid(geom,area,false);
end

    function [res] = compute_ctrl_pts(params,rv)
        r = rv*(0.99*params(1,:)+0.01);
        phi = (params(2,:)+(0:n-1))*2*pi/n;
        res = [r.*cos(phi);r.*sin(phi)];
        res = [res(:,end),res,res(:,1:2)];
    end
end

function [geom,c] = freeform_geom_withC(params,max_r,C0)
geom = [];
c = [0,0];
length_tol=1e-4;
diff_tol=1e-6;
if isempty(params(params>1|params<0)) && max_r>0
    n = length(params);
    ctrl_pts = compute_ctrl_pts(params,max_r);
    tmpgeom = nrbmak(ctrl_pts,0:1/(n+6):1); %assume cubics with uniform knots
    tmpgeom = nrbclamp(tmpgeom);
    C = curve_length(tmpgeom);

    while abs(C-C0)>length_tol
        pts1 = compute_ctrl_pts(params,max_r-diff_tol/2);
        pts2 = compute_ctrl_pts(params,max_r+diff_tol/2);
        g1 = nrbmak(pts1,0:1/(n+6):1);
```

```
        g2 = nrbmak(pts2,0:1/(n+6):1);
        g1 = nrbclamp(g1);
        g2 = nrbclamp(g2);
        dC = (curve_length(g2)-curve_length(g1))/diff_tol;
        max_r = max_r - (C-C0)/dC;
        ctrl_pts = compute_ctrl_pts(params,max_r);
        if any(isnan(ctrl_pts),'all') || any(isinf(ctrl_pts),'all')
            return;
        end
        tmpgeom = nrbmak(ctrl_pts,0:1/(n+6):1); %assume cubics with uniform knots
        tmpgeom = nrbclamp(tmpgeom);
        C = curve_length(tmpgeom);
    end
    geom = tmpgeom;
    %normalize knot vector
    geom.knots = (geom.knots-geom.knots(1))./(geom.knots(end)-geom.knots(1));
    geom = convertNRB(geom);
    area = curve_area(tmpgeom);
    c = curve_centroid(geom,area,false);
    geom.coefs(1,:) = geom.coefs(1,:) - ones(1,geom.number)*c(1);
    geom.coefs(2,:) = geom.coefs(2,:) - ones(1,geom.number)*c(2);
    % c = curve_centroid(geom,area,false);
end

    function [res] = compute_ctrl_pts(params,rv)
        r = rv*(0.99*params(1,:)+0.01);
        phi = (params(2,:)+(0:n-1))*2*pi/n;
        res = [r.*cos(phi);r.*sin(phi)];
        res = [res(:,end),res,res(:,1:2)];
    end
end
```