Objective-First Nanophotonic Design Plan

Jesse Lu

November 21, 2012

Contents

Chapter 1

Introduction

1.1 Problem statement

Our goal is to create a software package to enable the design of nanophotonic devices. Specifically, our software produces designs for *linear* nanophotonics devices based on *desired coupling strengths* between various optical ports.

To put things mathematically, we want to create software to solve the following problem,

$$\begin{aligned} \text{minimize} \quad & f(x) + g(z) && \text{(1.1a)} \\ \text{subject to} \quad & A(z)x - b(z) = 0 && \text{(1.1b)} \end{aligned}$$

where

- x and z are the variables representing the *field* our device produces and the *structure* of the device respectively,
- $f(x)$ and $g(z)$ are our *design objectives*, which tell us the desirable properties that we would like our device to achieve, and
- $A(z)x - b(z)$ is the *physics residual*, the underlying physical laws which must be met.

In general, we need to consider multiple fields produced by the device. Our problem statement is then

$$\begin{aligned} \text{minimize} \quad & \sum_{i}^{N} f_i(x_i) + g(z) && \text{(1.2a)} \\ \text{subject to} \quad & A_i(z)x_i - b_i(z) = 0, \quad \text{for } i = 1, \ldots, N. && \text{(1.2b)} \end{aligned}$$

1.2 Structure of our software

In reality, our software package consists of various modules which are designed to be maximally orthogonal. The various modules are illustrated in figure 1.1.

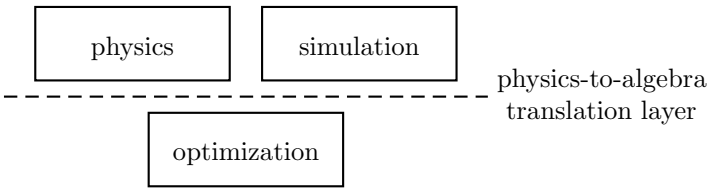

Figure 1.1: Structure of our software. The physics and simulation modules provide a complete description of the problem. The translation module takes this description and casts it in the language of linear algebra, which the optimization module can then use to produce various designs.

Although all modules are provided by our software, the majority of the innovation lies in the optimization module. Since the optimization module works entirely in the realm of linear algebra, a translation module is provided in order to cast concepts from electromagnetics into this form. Finally, physics and simulation modules are provided to allow the user to describe the design problem in physical, rather than mathematical terms.

Chapter 2

Theory

2.1 Mathematically rigorous statement of the problem

As previously stated, the problem we want to solve is

$$\text{minimize} \quad \sum_{i}^{N} f_i(x_i) + g(z) \tag{2.1a}$$

$$\text{subject to} \quad A_i(z)x_i - b_i(z) = 0, \quad \text{for } i = 1, \ldots, N. \tag{2.1b}$$

To now be more precise,

- $x_i \in \mathbf{C}^m$ are the field variables,
- $z \in \mathbf{R}^n$ is the structure variable,
- $f_i(x_i) \in \mathbf{C}^m \to \mathbf{R}$ are the field design objectives,
- $g(z) \in \mathbf{R}^n \to \mathbf{R}$ is the structure design objective,
- $A_i(z)x_i - b_i(z)$ are the physics residuals, with
- $A_i(z) \in \mathbf{C}^{m \times m}$ and
- $b_i(z) \in \mathbf{C}^m$.

2.1.1 Definition of physics residual

The physics residual, $A_i(z)x_i - b_i(z)$, corresponds to the electromagnetic wave equation

$$(\nabla \times \mu^{-1} \nabla \times -\omega^2 \epsilon) E = -i\omega J \tag{2.2}$$

via

- $\nabla \times \mu^{-1}\nabla \times -\omega^2\epsilon \rightarrow A_i(z)$, where ϵ is a function of z,
- $E \rightarrow x_i$, and
- $-i\omega J \rightarrow b_i(z)$, typically constant with respect to z.

2.1.2 Bi-affine property of the physics residual

Critically, (2.2) is not only linear in E, but is also affine in ϵ. This allows us to form the extremely useful relationship,

$$A_i(z)x_i - b_i(z) = B_i(x_i)z - d_i(x_i) = 0, \tag{2.3}$$

where $B(x_i) \in \mathbf{C}^{n\times n}$ and $d(x_i) \in \mathbf{C}^n$.

2.1.3 Definition of the field design objective

Although the field design objective $f_i(x_i)$ can take on virtually any form, we choose to define it very specifically as

$$f_i(x_i) = \sum_j I_+(|c_{ij}^\dagger x_i| - \alpha_{ij}) + I_+(\beta_{ij} - |c_{ij}^\dagger x_i|), \tag{2.4}$$

where $c_{ij} \in \mathbf{C}^m$ and I_+ is the indicator function on nonnegative reals,

$$I_+(u) = \begin{cases} 0 & u \geq 0, \\ \infty & u < 0. \end{cases} \tag{2.5}$$

Such a design objective implements the constraints $\alpha_{ij} \leq |c_{ij}^\dagger x_i| \leq \beta_{ij}$ which can be interpreted physically as constraining the power emitted into the optical modes represented by c_{ij}.

2.1.4 Choice of the structure design objective

In contrast to the narrow definition of the field design objective, the structure design objective is relatively unconstrained; taking on various forms to best suit the needs of the structure parameterization in use.

2.2 Properties of the problem

We now examine the general mathematical properties of the stated problem (2.1).

2.2.1 Convexity analysis

First we note that, as presented, (2.1) is non-convex in the variables x_i and z. Not only are the physics residuals $A_i(z)x_i - b_i(z)$ non-convex, but the field design objective, in that it implements the $\alpha_{ij} \leq |c_{ij}^\dagger x_i|$ constraint, is non-convex as well.

That our problem is non-convex means that it is fundamentally hard to solve because of the existence of multiple local minima. Additionally, even if we were to arrive at the global maxima, we would not have a straightforward way to verify global optimality. Lastly, fast convergence even to local minima may be difficult because methods such as Newton's method can not be directly applied to non-convex problems.

That said, (2.1) is *separably convex* in x_i and z if we ignore the non-convexity in the field design objectives, $f_i(x_i)$, and assume that the structure design objective, $g(z)$, is convex as well. This is given because of the bi-affine property of the physics residual, as shown in (2.3).

The separably convex, or *bi-convex*, properties of our problem open the door for the use of alternating direction algorithms, and specifically the alternating directions methods of multipliers (ADMM), for the implementation of a global optimization paradigm, as referred to in section 3.1. Of course, in this context we do not use the term "global" rigorously, but only to differentiate it from strategies that rely purely on local information.

Lastly, since our problem is not bi-convex in the case of non-convex field or structure design objectives, simple extensions to alternating direction algorithms are employed.

2.2.2 Optimality condition

Based on the problem definition in (2.1), we can write down a set of equations that, when met, signal that we have arrived at a locally optimal point. Otherwise known as the Karush-Kuhn-Tucker (KKT) conditions, these are, for (2.1),

$$|c_{ij}^\dagger x_i| - \beta_{ij} \leq 0, \tag{2.6a}$$

$$\alpha_{ij} - |c_{ij}^\dagger x_i| \leq 0, \tag{2.6b}$$

$$A_i(z)x_i - b_i(z) = 0, \tag{2.6c}$$

$$\nabla_z g(z) + \sum_i B(x_i)^\dagger \nu_i = 0, \tag{2.6d}$$

where $\nu_i \in \mathbf{C}^n$ are dual variables.

Chapter 3

The optimization module

3.1 Capabilities of the module

We have implemented various *optimization paradigms* and *structure parameterizations* which a user can arbitrarily combine in order to to solve (1.2).

Specifically, the user can choose to work in either a *local* or *global* optimization paradigm:

- the local paradigm uses the adjoint method to find small changes in the structure which will decrease the design objective;
- the global paradigm uses the objective-first method to arrive at a structure by forcing the design objective to be met from the start.

Also, the user can choose between the various structure parameterizations which include density, boundary, continuous, and combinatoric parameterizations. These parameterizations describe the structure as either

- a material density at every point in space;
- a boundary between objects of different materials;
- a set of continuously variable user-defined parameters; or
- a selective combination of user-defined objects.

3.2 Structure of the module

The module is naturally separated into two submodules, the optimization paradigm (or simply, paradigm) submodule and the structure parameterization (or simply, structure) submodule. The interaction between these two submodules is illustrated in figure 3.1.

In essense, a design is achieved via repeated calls to the structure submodule by the paradigm submodule, in which carefully chosen quadratic functions $Q(z)$ are minimized.

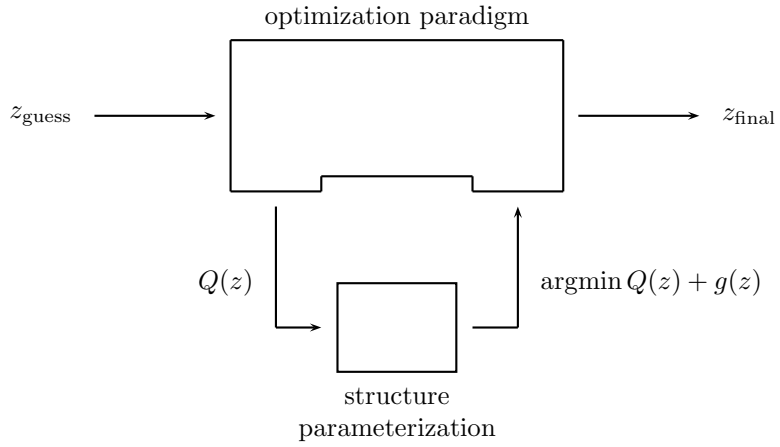

Figure 3.1: Basic layout of the module. The optimization paradigm submodule accepts an initial structure z_{guess} and returns a final, optimized structure z_{final}. This is accomplished by repeatedly passing quadratic functions $Q(z)$ to the structure parameterization module, which returns an updated structure z which minimizes $Q(z) + g(z)$.

3.3 The paradigm submodule

3.3.1 The local optimization paradigm

The local optimization paradigm applies the adjoint method to solve (2.1) in a manner analogous to the steepest-descent strategy. Specifically, we compute the derivative of the total field design objective with respect to z,

$$\nabla_z F = \sum_i^N \nabla_z f_i(x_i), \tag{3.1}$$

and then pass

$$Q(z) = \frac{1}{2}\|z - z_0\|^2 + \kappa \nabla_z F^\dagger (z - z_0) \tag{3.2}$$

to the structure submodule, for some κ.

The analogy with a steepest-descent strategy is apparent since the global minimum of $Q(z)$ can be found via

$$\nabla_z Q(z) = (z - z_0) + \kappa \nabla F = 0, \tag{3.3}$$

resulting in

$$z = z_0 - \kappa \nabla_z F, \tag{3.4}$$

from which we see that the role of κ is to determine the step-size in the direction of steepest-descent.

Computation of $\nabla_z F$

The derivatives of the individual field design objectives are found via

$$\frac{d}{dz} f_i(x_i) = \frac{\partial f_i}{\partial x_i} \frac{dx_i}{dz}. \tag{3.5}$$

To obtain dx_i/dz we first differentiate the corresponding physics residual, $A_i(z)x_i - b_i(z)$,

$$A_i(z)dx_i + dA_i(z) - db_i(z) = A_i(z)dx_i + B_i(x_i)dz, \tag{3.6}$$

where $dA_i(z) - db_i(z) = B_i(x_i)dz$ as a result of the bi-affine property of the physics residual (2.3).

Assuming that physics is already satisfied, $A_i(z)x_i - b_i(z) = 0$, and that we want to keep the physics residual at 0, the following condition must then be satisfied,

$$A_i(z)dx_i = -B_i(x_i)dz \tag{3.7}$$

from which we obtain

$$\frac{dx}{dz} = -A_i(z)^{-1} B_i(x_i). \tag{3.8}$$

Efficient calculation of dx/dz is almost always impossible since $B_i(x_i) \in \mathbf{C}^{n \times n}$. At the same time, since $\partial f_i / \partial x_i \in \mathbf{C}^{1 \times m}$, we instead compute $df_i(x_i)/dz$ via

$$\frac{d}{dz} f_i(x_i) = -\frac{\partial f_i}{\partial x_i} A_i(z)^{-1} B_i(x_i) = -\left(A_i(z)^\dagger \frac{\partial f_i}{\partial x_i}^\dagger \right)^\dagger B_i(x_i) \tag{3.9}$$

which requires only one solve of $A_i^\dagger$ as opposed to n solves of A_i.

Computation of $\partial f_i(x_i)/\partial x_i$

The definition of $f_i(x_i)$ as given in (2.4) is not differentiable since any deviation away from the power constraints results in $f_i = \infty$. In order to make the constraints differentiable, then, we use the following relaxed indicator function I_+^{rel} in place of I_+,

$$I_+^{\text{rel}}(u) = \begin{cases} 0 & u \geq 0, \\ \frac{1}{a}|u|^p & u < 0, \end{cases} \tag{3.10}$$

where a is a normalization factor and $p \in (0, \infty]$. In order to guarantee a well-defined value of $\partial f_i(x_i)/\partial x_i$ even for $p = \infty$, we can let $a = \max_i f_i(x_i)$.

3.3.2 The global optimization paradigm

3.4 The structure submodule