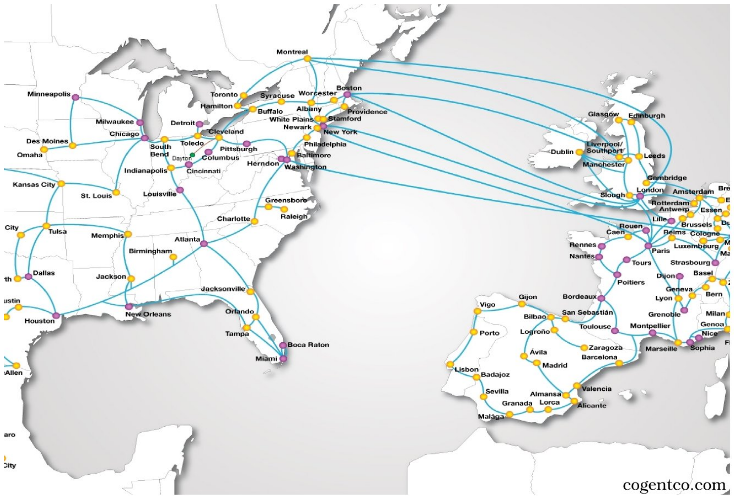
Nanophotonic Computational Design

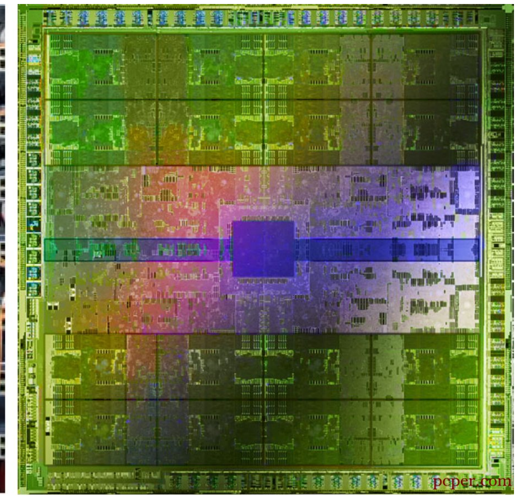
Jesse Lu

February 25, 2013

Takeaway: Taught a computer to design nanophotonic devices

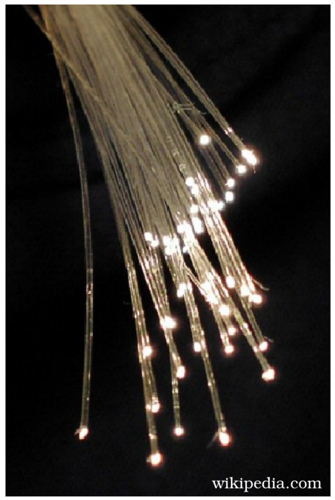
Part 1: Motivation

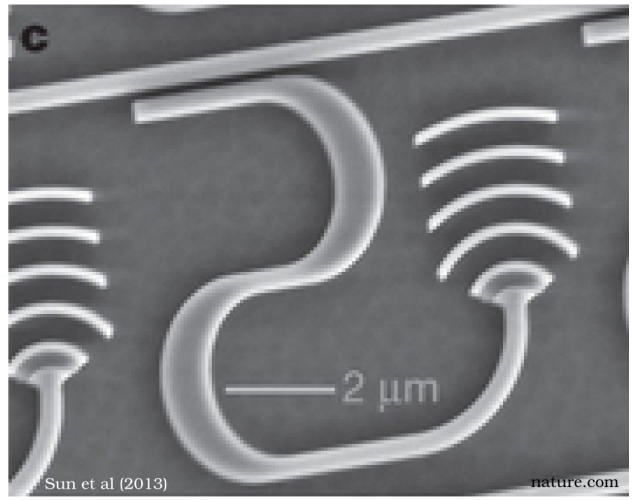




- As information grows, optical networks needed
  - across continents
  - within a datacenter
  - between chips and on-chip

- An on-chip optical network is a fundamentally new optical communications technology: *the integrated optical circuit*

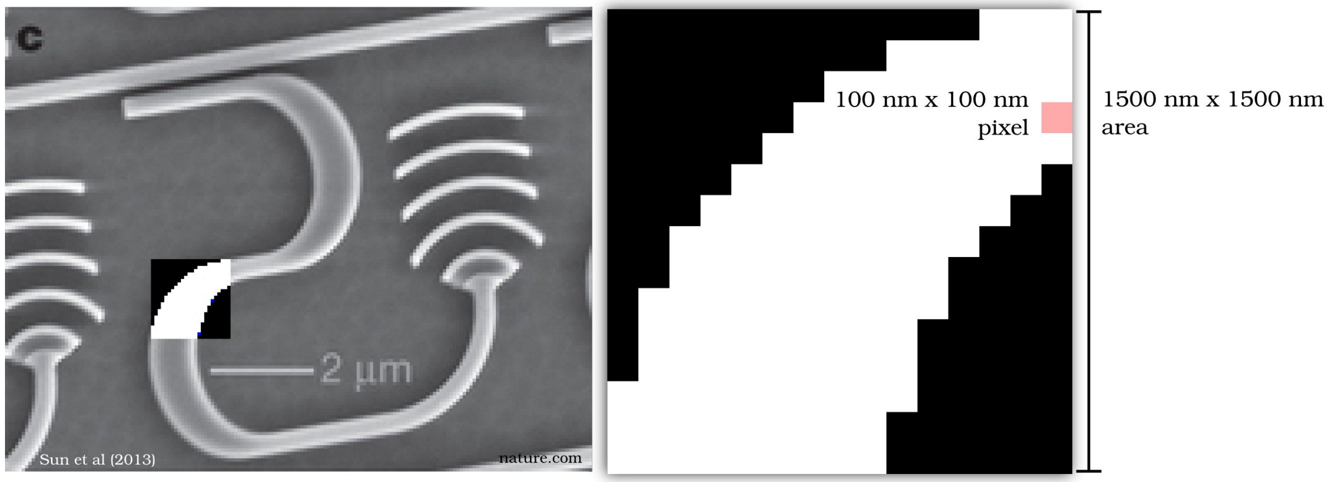




- Miniaturization drives
  - component price down
  - functionality up
  - design complexity (way) up

- Increasing design complexity requires additional degrees of freedom

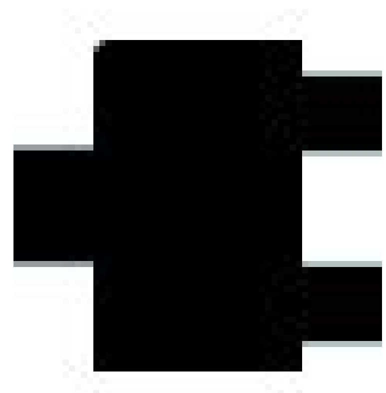
- Fortunately, we have a virtually unlimited amount



- Include/exclude per pixel gives us $2^{(15^2)} = 2^{225}$ possibilities, uncountable

- Only feasible solution: Humans describe, Computers design

```
device mux2
    in: {freq1, freq2}
    out1 <= freq1
    out2 <= freq2
```



Part 2: Theory

- First, cast electromagnetic wave equation into linear algebra terms

$$(\nabla \times \mu_0^{-1} \nabla \times -\omega^2 \epsilon) E + i\omega J = 0 \longrightarrow A(z)x - b = 0$$

where

$$\begin{aligned} E &\rightarrow x \\ \epsilon &\rightarrow z \\ \nabla \times \mu_0^{-1} \nabla \times -\omega^2 \epsilon &\rightarrow A(z) \\ -i\omega J &\rightarrow b \end{aligned}$$

- $A(z)x - b$ is called the *physics residual*

- Secondly, formulate our optimization objective

$$f(x) = (\alpha - |c^\dagger x|)^2$$

- $c^\dagger x$ is equivalent to overlap integral $\int E_t^* E$

- To design linear devices, objective chosen to be overlap integral with target field at output port

- $f(x)$ is called the *field design objective*

- Typically,

$$\begin{aligned} \text{minimize} \quad & (\alpha - |c^\dagger x|)^2 \\ \text{subject to} \quad & A(z)x - b = 0 \end{aligned}$$

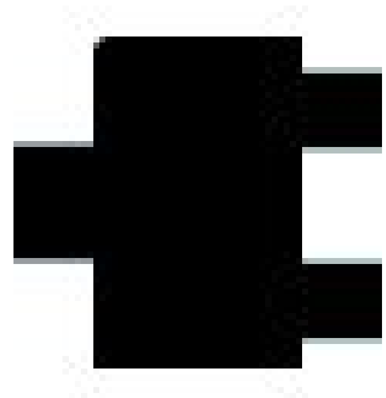
- Efficient algorithm known as the *adjoint method*

- Our alternative formulation, known as *objective-first*

$$\begin{aligned} \text{minimize} \quad & \|A(z)x - b\|^2 \\ \text{subject to} \quad & |c^\dagger x| = \alpha \end{aligned}$$

- Perfect performance always enforced, even at the expense of breaking physical laws

- Results in *soft-physics* solves



- Key insight: soft-physics solution suggests optimal structure

$$\begin{array}{rl} \text{minimize} & \|A(z)x - b\|^2 \\ \text{subject to} & |c^\dagger x| = \alpha \end{array}$$

- Could be solved by iteratively solving for x and z
  - Known as *alternating directions*
  - Takes advantage of bi-linearity of the physics residual,

$$A(z)x - b = B(x)z - d(x)$$

- *Alternating directions method of multipliers (ADMM)* gives much faster convergence
  - Due to introduction of dual variable

- Full problem is multi-mode and multi-output
- Objective-first formulation:

$$\begin{array}{ll} \text{minimize} & \sum_{i}^{M} \|A_i(z)x_i - b_i\|^2 \\ \text{subject to} & \alpha_{ij} \le |c_{ij}^\dagger x_i| \le \beta_{ij}, \quad \text{for } i = 1, \ldots, M \text{ and } j = 1, \ldots, N_i \end{array}$$

- Adjoint method formulation:

$$\begin{array}{ll} \text{minimize} & \sum_{ij}^{M,N_i} \max\{\alpha_{ij} - |c_{ij}^\dagger x_i|, |c_{ij}^\dagger x_i| - \beta_{ij}, 0\} \\ \text{subject to} & A_i(z)x_i - b_i = 0, \quad \text{for } i = 1, \ldots, M \end{array}$$

Part 3: Implementation

- Implementation in Matlab (github.com/JesseLu/lumos)
- x is 3D and about a million variables
  - $A_i(z)^{-1}$ Solved in the cloud (AWS), scales arbitrary modes.

- z is about a thousand variables and can be solved locally

- Both objective-first and adjoint methods implemented
- Designs produced using a combination of steps
- Initial structure...

- Step 1: ob-1 with density

- Step 2: adjoint with density

- Step 3: adjoin with level-set

- Verification with larger simulation

Part 4: Results