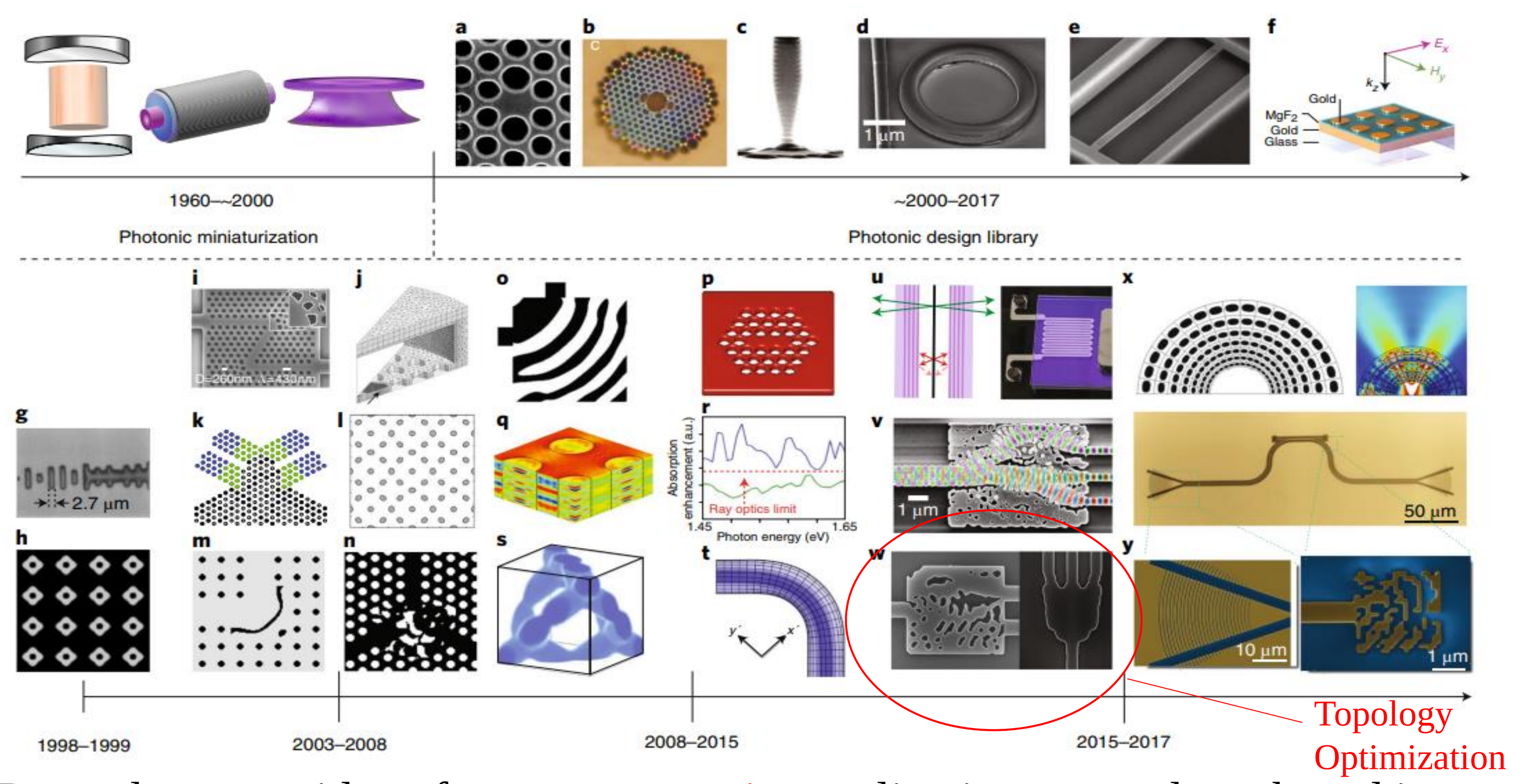
CUDA GPU-Based Software Development for the Design and Characterization of Next-Generation Optoelectronic Circuits

Linwei Yuan[1], Guowu Zhang[1], Yuri Grinberg[2], Dan-Xia Xu[2], and Odile Liboiron-Ladouceur[1]

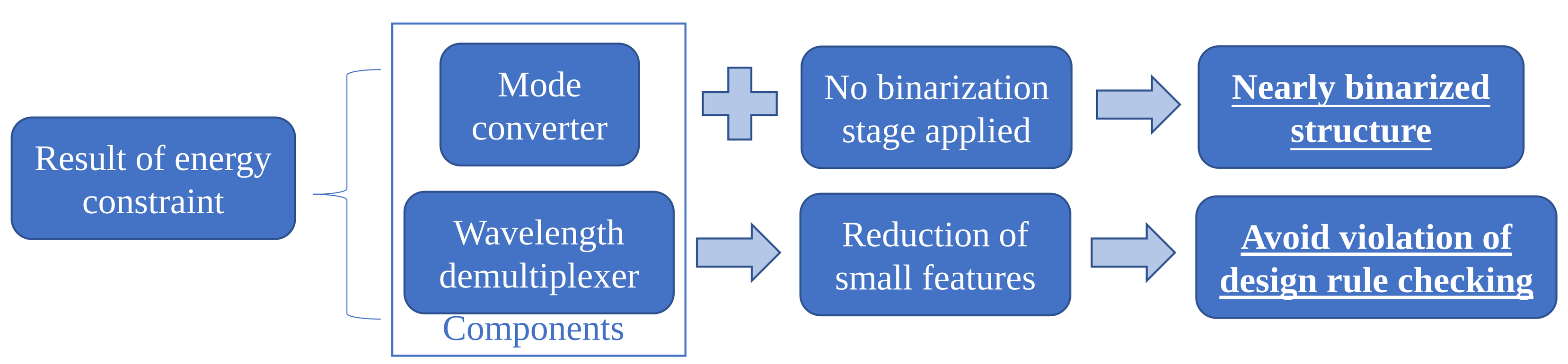
[1]Department of Electrical and Computer Engineering, McGill University, and [2]National Research Council Canada

Introduction

- Nowadays, photonics have a wide range of applications to fields including telecommunications, information processing, medicine, and so on.
- During the second half of the twentieth century, advancement in fabrication capabilities allowed photonic engineering to expand into the nanoscale. As shown below, many of photonic design templates were arising, and topological inverse design has become a crucial design and optimization methodology.



- Recently, a new idea of energy constraint application to topology-based inverse design has been proposed, which increases the robustness to fabrication imperfection and further improves the computationally intensive optimization.



- There are three advantages associated with applying this energy constraint:

Advantage 1	No extra binarization needed
Advantage 2	Less energy interacts with boundaries (more robust)
Advantage 3	Generation of small features is penalized

- Since the method is strongly computationally intensive, methods to shorten the computation time are widely explored, including usage of simulation software (SPINS etc.), parallel computing, and development of matrix algorithms.

Objective

With knowing the novelty of topological inverse design with energy constraint, the objective of the project is to explore possible solutions to accelerate the optimization process, using CUDA C++, NVidia GPU, and simulation tools.

Methodology

- Step 1: Understand the principle of inverse design with energy constraint
  - Conventionally, the optimization of nanophotonic devices are set as a minimization problem of desired design objective function $F_{obj}(\boldsymbol{E})$ with respect to a normalized design variable $\boldsymbol{\rho}(\mathbf{0} < \boldsymbol{\rho} < \mathbf{1})$, subject to the Maxwell's wave equation in the frequency domain:

$$\nabla \times \frac{1}{\mu_0 \mu_r} \nabla \times \boldsymbol{E} - \omega^2 \varepsilon_0 \varepsilon_r(\boldsymbol{\rho}) \boldsymbol{E} = -i\omega \boldsymbol{J}$$

electrical field distribution
vacuum electrical permittivity
frequency
curl operator
relative permeability (set to 1)
relative permittivity distribution
vacuum magnetic permeability
optical resource that excites the field

  - With energy constraint objective function F_{energy} and subject to the formula above, the overall optimization will be formulated as:

$$\min_{\rho} F_{obj}(\boldsymbol{E}) = (1 - w_c) \times F_{EM}(\boldsymbol{E}) + w_c \times F_{energy}(\rho, \boldsymbol{E})$$

Pre-defined EM objective function for the device performance
weight factor that controls F_{energy}

- Step 2: Determine that it is the cross products of sparse matrices in those Maxwell equations that cost the most computation time
- Step 3: Investigate technologies that could be utilized for development of a new software to boost this sparse matrix–vector multiplication (SpMV):

NVIDIA GEFORCE GTX
NVIDIA CUDA
docker

- Step 4: Develop and compare the SpMV algorithms including diagonal format (DIA), ELLPACK format (ELL), coordinate format (COO), compressed sparse row format (CSR), and hybrid format (combinations of different formats):

Storage format	Array number	Array type
DIA	*2*	*value array; offset array*
ELL	*2*	*value array; index array*
COO	*3*	*value array; row array; column array;*
CSR	*3*	*value array; pointer array; index array;*

- Step 5: Choose CSR to do more experiments according to the comparison results and further researches
  - Here is an example of the CSR algorithm:

$$A = \begin{bmatrix} 1 & 7 & 0 & 0 \\ 0 & 2 & 8 & 0 \\ 5 & 0 & 3 & 9 \\ 0 & 6 & 0 & 4 \end{bmatrix}$$

$$ptr = [0\ 2\ 4\ 7\ 9]$$
$$indices = [0\ 1\ 1\ 2\ 0\ 2\ 3\ 1\ 3]$$
$$values = [1\ 7\ 2\ 8\ 5\ 3\ 9\ 6\ 4]$$

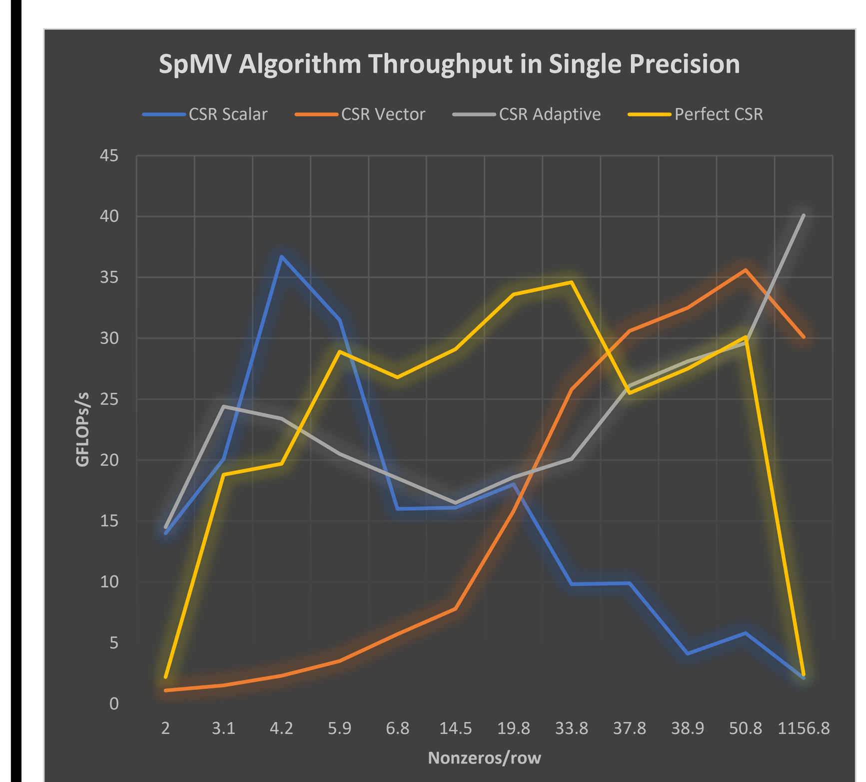
***The pointer array stores the accumulated number of values added in each row. The index array will provide the column index for each row number (row numbers are specified by "ptr"). The value array just stores data that corresponds to each index in "indices".**

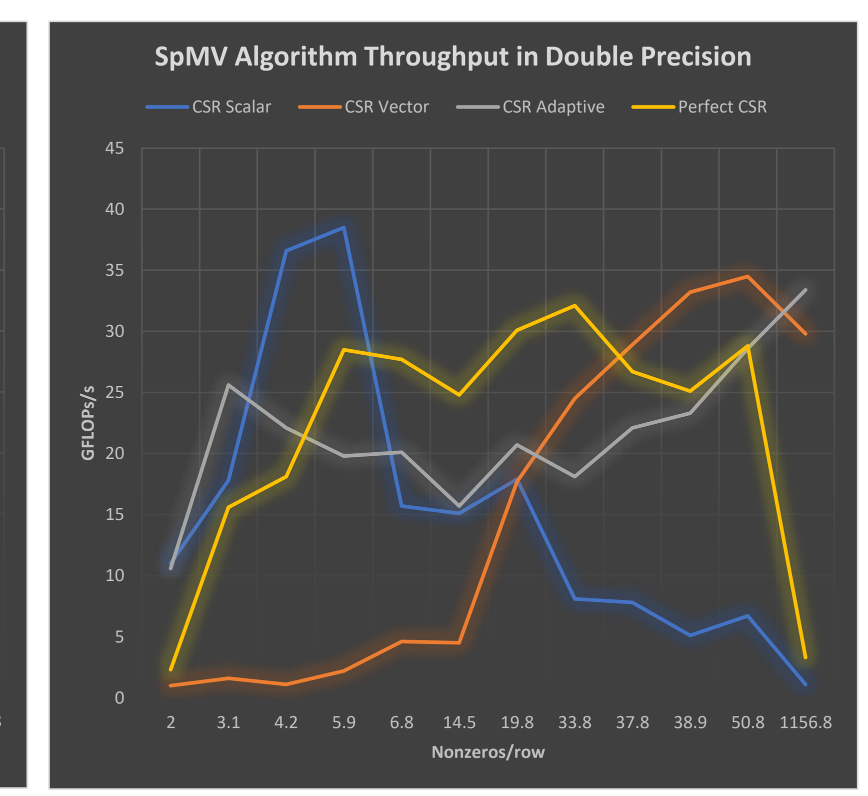
Results

- As a result, 4 deviations of CSR (CSR Scalar, CSR Vector, CSR Adaptive, and Perfect CSR) were investigated and measured with respect to their throughputs.
- As an example, we would input matrices with the following specifications:

Matrix Number	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*
Nonzeros	3M	4M	4M	5M	11K	10K	84K	5.6K	102K	2M	18K	1000M
Nonzeros/row	2	3.1	4.2	5.9	6.8	14.5	19.8	33.8	37.8	38.9	50.8	1156.8

- The result of the implemented algorithms are shown below in both single and double precision forms. As shown in the graphs, we observe that different CSR algorithms have their own strength: CSR Adaptive tends to compute faster when encountering sparse matrices with lots of nonzeros; When it comes to light-weight SpMV computation, CSR Scalar is simple and efficient enough.
- In general, from this result, we ideally could choose the most efficient SpMV algorithm according to the characteristics of input matrices.





Future work: We would incorporate the algorithms to an online open-source Maxwell calculator to accelerate topological inverse design process with energy constraint. Future work could also include building a machine learning unit to predict which SpMV algorithm to use when it receives an optical design.

References

Guowu Zhang, Dan-Xia Xu, Yuri Grinberg, and Odile Liboiron-Ladouceur. "Topological inverse design of nanophotonic devices with energy constraint." Optics Express 29, no. 8 (2021): 12681-12695.

Guowu Zhang, Dan-Xia Xu, Yuri Grinberg, and Odile Liboiron-Ladouceur. "Experimental demonstration of robust nanophotonic devices optimized by topological inverse design with energy constraint." Photonics Research.