Derivative-based Optimization for Applications in Physics

Andrea Walther
Institut für Mathematik
Humboldt-Universität zu Berlin and ZIB

Keynote Lecture
4th MODE Workshop on Differentiable Programming for Experimental Design
September 24, 2024

Berlin Mathematics Research Center
MATH+

Outline

Where are Derivatives Needed?

- Optimization:

$$
\begin{array}{lll}
\text{unbounded:} & \min f(x), & f : \mathbb{R}^n \to \mathbb{R} \\
\text{bounded:} & \min f(x), & f : \mathbb{R}^n \to \mathbb{R} \\
 & c(x) \ = \ 0, & c : \mathbb{R}^n \to \mathbb{R}^m \\
 & h(x) \leq 0, & h : \mathbb{R}^n \to \mathbb{R}^l
\end{array}
$$

Where are Derivatives Needed?

- Optimization:

$$\begin{array}{lll} \text{unbounded:} & \min f(x), & f:\mathbb{R}^n \to \mathbb{R} \\ \text{bounded:} & \min f(x), & f:\mathbb{R}^n \to \mathbb{R} \\ & c(x) \; = \; 0, & c:\mathbb{R}^n \to \mathbb{R}^m \\ & h(x) \le 0, & h:\mathbb{R}^n \to \mathbb{R}^l \end{array}$$

- Solution of nonlinear equation systems

$$F(x) \; = \; 0, \quad F \; : \; \mathbb{R}^n \; \to \; \mathbb{R}^n$$

  Newton method requires $F'(x) \in \mathbb{R}^{n\times n}$

Where are Derivatives Needed?

- Optimization:

$$
\begin{array}{llll}
\text{unbounded:} & \min f(x), & & f : \mathbb{R}^n \to \mathbb{R} \\
\text{bounded:} & \min f(x), & & f : \mathbb{R}^n \to \mathbb{R} \\
 & c(x) & = 0, & c : \mathbb{R}^n \to \mathbb{R}^m \\
 & h(x) & \leq 0, & h : \mathbb{R}^n \to \mathbb{R}^l
\end{array}
$$

- Solution of nonlinear equation systems

$$F(x) = 0, \quad F : \mathbb{R}^n \to \mathbb{R}^n$$

  Newton method requires $F'(x) \in \mathbb{R}^{n \times n}$

- Simulation of complex system
  - definition
  - integration of differential equations using implicit methods

Berlin Mathematics Research Center

Where are Derivatives Needed?

- Optimization:

$$\begin{array}{lrcll} \text{unbounded:} & \min f(x), & & & f:\mathbb{R}^n \to \mathbb{R} \\ \text{bounded:} & \min f(x), & & & f:\mathbb{R}^n \to \mathbb{R} \\ & c(x) & = & 0, & c:\mathbb{R}^n \to \mathbb{R}^m \\ & h(x) & \leq & 0, & h:\mathbb{R}^n \to \mathbb{R}^l \end{array}$$

- Solution of nonlinear equation systems

$$F(x) = 0, \quad F : \mathbb{R}^n \to \mathbb{R}^n$$

  Newton method requires $F'(x) \in \mathbb{R}^{n\times n}$

- Simulation of complex system
  - definition
  - integration of differential equations using implicit methods
- Sensitivity analysis
- Real-time control

Frequent Situation

Computing Derivatives

Given:

Description of functional relation as

- formula $y = F(x)$ $\Rightarrow$ explicit expression $y' = F'(x)$
- computer program $\Rightarrow$?

Computing Derivatives

Given:

Description of functional relation as

- formula $y = F(x)$ $\Rightarrow$ explicit expression $y' = F'(x)$
- computer program $\Rightarrow$?

Task:

Computation of derivatives taking

- requirements on exactness
- computational effort

into account

Finite Differences

Idea: Taylor expansion, $f : \mathbb{R} \to \mathbb{R}$ smooth then

$$
\begin{aligned}
 & f(x+h) &=& f(x) + hf'(x) + h^2 f''(x)/2 + h^3 f'''(x)/6 + \dots \\
\Rightarrow & f(x+h) &\approx& f(x) + hf'(x) \\
\Rightarrow & Df(x) &=& \frac{f(x+h) - f(x)}{h}
\end{aligned}
$$

Finite Differences

Idea: Taylor expansion, $f : \mathbb{R} \to \mathbb{R}$ smooth then

$$
\begin{aligned}
& f(x+h) &=& \; f(x) + hf'(x) + h^2 f''(x)/2 + h^3 f'''(x)/6 + \dots \\
\Rightarrow \quad & f(x+h) &\approx& \; f(x) + hf'(x) \\
\Rightarrow \quad & Df(x) &=& \; \frac{f(x+h) - f(x)}{h}
\end{aligned}
$$

- simple derivative calculation (only function evaluations!)
- inexact derivatives
- computation cost often too high

  $F : \mathbb{R}^n \to \mathbb{R} \quad \Rightarrow \quad \mathrm{OPS}(\nabla F(x)) \sim (n+1)\mathrm{OPS}(F(x))$

Analytic Differentiation

- symbolic derivatives, e.g.,

$$f(x) = \exp(\sin(x^2)) \Rightarrow$$
$$f'(x) = \exp(\sin(x^2)) \, * \, \cos(x^2) \, * \, 2x$$

Analytic Differentiation

- symbolic derivatives, e.g.,

$$f(x) = \exp(\sin(x^2)) \Rightarrow$$
$$f'(x) = \exp(\sin(x^2)) \ * \ \cos(x^2) \ * \ 2x$$

- analytic optimality conditions for optimal control problems, i.e.,

$$\min J(x,u) \quad \text{such that} \quad x' = f(x,u) \quad + \quad \text{IC ?}$$

Analytic Differentiation

- symbolic derivatives, e.g.,

$$f(x) = \exp(\sin(x^2)) \Rightarrow$$
$$f'(x) = \exp(\sin(x^2)) \;*\; \cos(x^2) \;*\; 2x$$

- analytic optimality conditions for optimal control problems, i.e.,

$$\min J(x,u) \quad \text{such that} \quad x' = f(x,u) \quad + \quad \text{IC ?}$$

  derivative computation based on
  - sensitivity equation
  - adjoint equation $\lambda' = -f_x(x,u)^\top \lambda \quad + \quad \text{TC}$

  $\Rightarrow J'(y(u),u)$ based on continuous adjoint

Analytic Differentiation

- symbolic derivatives, e.g.,

$$f(x) = \exp(\sin(x^2)) \Rightarrow$$
$$f'(x) = \exp(\sin(x^2)) \ * \ \cos(x^2) \ * \ 2x$$

- analytic optimality conditions for optimal control problems, i.e.,

$$\min J(x,u) \quad \text{such that} \quad x' = f(x,u) \quad + \quad \text{IC ?}$$

  derivative computation based on
  - sensitivity equation
  - adjoint equation $\qquad \lambda' = -f_x(x,u)^\top \lambda \quad + \quad \text{TC}$

  $\Rightarrow J'(y(u),u)$ based on continuous adjoint

  - legacy code (large number of lines) $\Rightarrow$ closed form not available

Analytic Differentiation

- symbolic derivatives, e.g.,

$$f(x) = \exp(\sin(x^2)) \Rightarrow$$
$$f'(x) = \exp(\sin(x^2)) \ * \ \cos(x^2) \ * \ 2x$$

- analytic optimality conditions for optimal control problems, i.e.,

$$\min J(x,u) \quad \text{such that} \quad x' = f(x,u) \quad + \quad \text{IC ?}$$

derivative computation based on

  - sensitivity equation
  - adjoint equation $\qquad \lambda' = -f_x(x,u)^\top \lambda \quad + \quad \text{TC}$

  $\Rightarrow J'(y(u),u)$ based on continuous adjoint

  - legacy code (large number of lines) $\Rightarrow$ closed form not available consistent derivative information?!

Example from Optimal Control

$$\min \frac{1}{2}\int_0^1 2x(t)^2 + u(t)^2 dt \quad \text{s.t.} \quad x' = \frac{1}{2}x(t) + u(t), \quad x(0) = 1$$

Example from Optimal Control

$$\min \frac{1}{2}\int_0^1 2x(t)^2 + u(t)^2 dt \quad \text{s.t.} \quad x' = \frac{1}{2}x(t) + u(t), \quad x(0) = 1$$

$$\lambda(t) = \frac{2e^{3-t} - 2e^{2t}}{e^{t/2}(2+e^3)}$$

$$\lambda(1) = 0$$

Example from Optimal Control

$$\min \frac{1}{2}\int_0^1 2x(t)^2 + u(t)^2 dt \quad \text{s.t.} \quad x' = \frac{1}{2}x(t) + u(t), \quad x(0) = 1$$

$$\lambda(t) = \frac{2e^{3-t} - 2e^{2t}}{e^{t/2}(2+e^3)}$$
$$\lambda(1) = 0$$

→

$$x^*(t) = (2e^{3t} + e^3)/(e^{3t/2}(2 + e^3))$$
$$u^*(t) = 2(e^{3t} - e^3)/(e^{3t/2}(2 + e^3))$$

Example from Optimal Control

$$\min \tfrac{1}{2}\int_0^1 2x(t)^2 + u(t)^2 dt \quad \text{s.t.} \quad x' = \tfrac{1}{2}x(t) + u(t), \quad x(0) = 1$$

$$\lambda(t) = \frac{2e^{3-t} - 2e^{2t}}{e^{t/2}(2+e^3)}$$

$$\lambda(1) = 0$$

$$x^*(t) = (2e^{3t} + e^3)/(e^{3t/2}(2+e^3))$$

$$u^*(t) = 2(e^{3t} - e^3)/(e^{3t/2}(2+e^3))$$

$$\min \tfrac{h}{2}\sum_{i=0}^{N} 2x_{i+1/2}^2 + u_{i+1/2}^2$$

$$x_{i+1/2} = x_i + \tfrac{h}{2}(x_i + u_i)$$

$$x_{i+1} = x_i + \tfrac{h}{2}(\tfrac{1}{2}x_{i+1/2} + u_{i+1/2})$$

Example from Optimal Control

$$\min \tfrac{1}{2}\int_0^1 2x(t)^2 + u(t)^2 dt \quad \text{s.t.} \quad x' = \tfrac{1}{2}x(t) + u(t), \quad x(0) = 1$$

$$\lambda(t) = \frac{2e^{3-t} - 2e^{2t}}{e^{t/2}(2+e^3)}$$
$$\lambda(1) = 0$$

→

$$x^*(t) = (2e^{3t} + e^3)/(e^{3t/2}(2+e^3))$$
$$u^*(t) = 2(e^{3t} - e^3)/(e^{3t/2}(2+e^3))$$

$$\min \tfrac{h}{2}\sum_{i=0}^{N} 2x_{i+1/2}^2 + u_{i+1/2}^2$$
$$x_{i+1/2} = x_i + \tfrac{h}{2}(x_i + u_i)$$
$$x_{i+1} = x_i + \tfrac{h}{2}(\tfrac{1}{2}x_{i+1/2} + u_{i+1/2})$$

→

$$x_i = 1, \; x_{i+1/2} = 0$$
$$u_i = -\tfrac{4+h}{2h}x_i, \; u_{i+1/2} = 0$$

W. Hager, Numerische Mathematik, 2000

Algorithmic Differentiation (AD)

aka Automatic Differentiation

= Differentiation of computer programs implementing $F : \mathbb{R}^n \mapsto \mathbb{R}^m$

Algorithmic Differentiation (AD)

aka Automatic Differentiation

= Differentiation of computer programs implementing $F : \mathbb{R}^n \mapsto \mathbb{R}^m$

Main Products:

- Quantitative dependence information (local):
  - Weighted and directed partial derivatives
  - Error and condition number estimates . . .
  - Lipschitz constants, interval enclosures . . .

Algorithmic Differentiation (AD)

aka Automatic Differentiation

= Differentiation of computer programs implementing $F : \mathbb{R}^n \mapsto \mathbb{R}^m$

Main Products:

- Quantitative dependence information (local):
  - Weighted and directed partial derivatives
  - Error and condition number estimates ...
  - Lipschitz constants, interval enclosures ...
- Qualitative dependence information (regional):
  - Sparsity structures, degrees of polynomials
  - Ranks, eigenvalue multiplicities ...

Algorithmic Differentiation (AD)

aka Automatic Differentiation

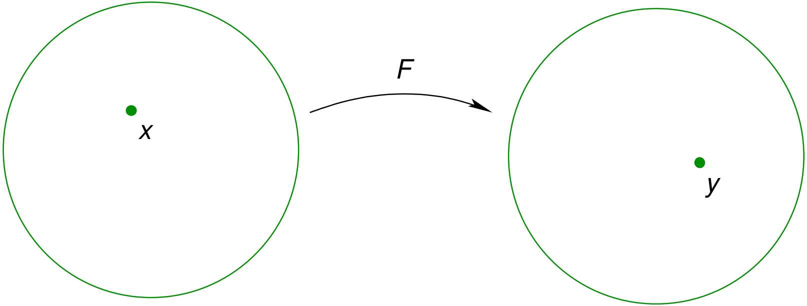
= Differentiation of computer programs implementing $F : \mathbb{R}^n \mapsto \mathbb{R}^m$

Main Products:

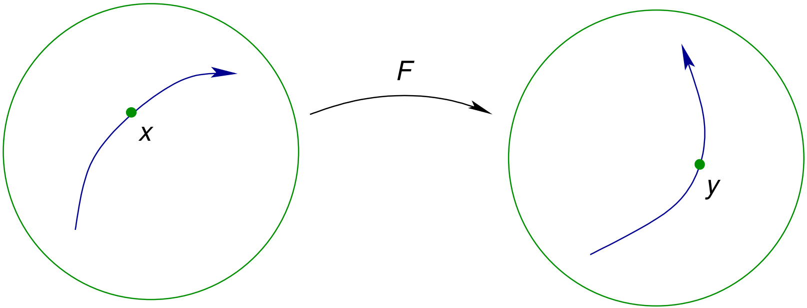
- Quantitative dependence information (local):
  - Weighted and directed partial derivatives
  - Error and condition number estimates . . .
  - Lipschitz constants, interval enclosures . . .
- Qualitative dependence information (regional):
  - Sparsity structures, degrees of polynomials
  - Ranks, eigenvalue multiplicities . . .

Assumption: F differentiable in a neighbourhood of current argument x

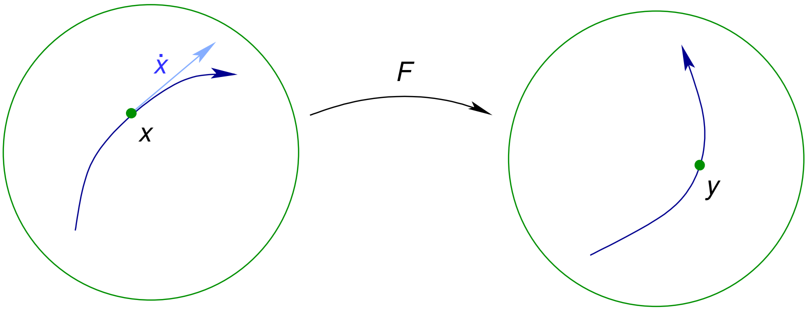
Forward mode AD = Tangents/Sensitivities



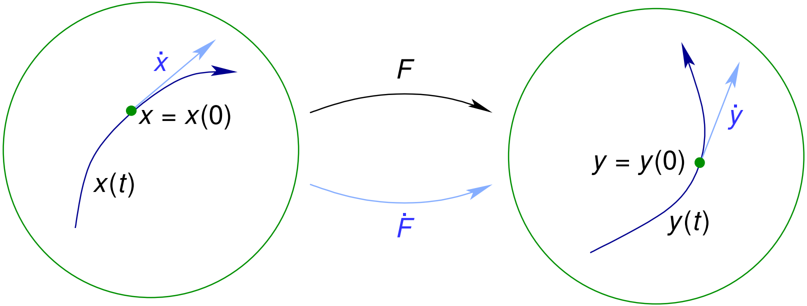
Forward mode AD = Tangents/Sensitivities



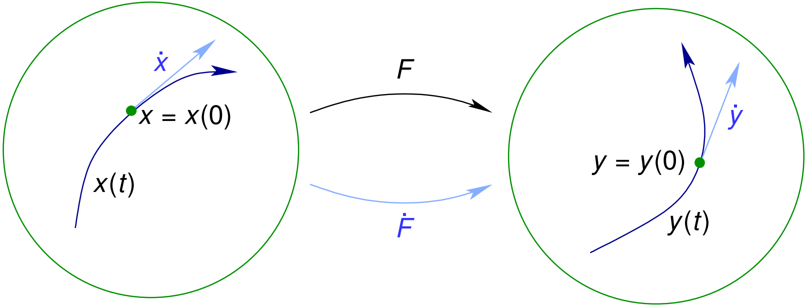
Forward mode AD = Tangents/Sensitivities



Forward mode AD = Tangents/Sensitivities

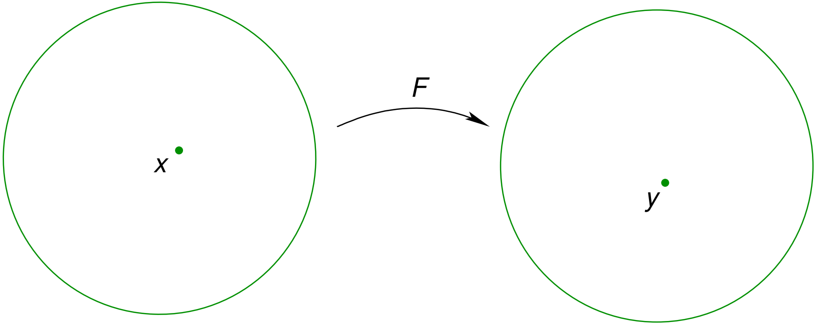


Forward mode AD = Tangents/Sensitivities

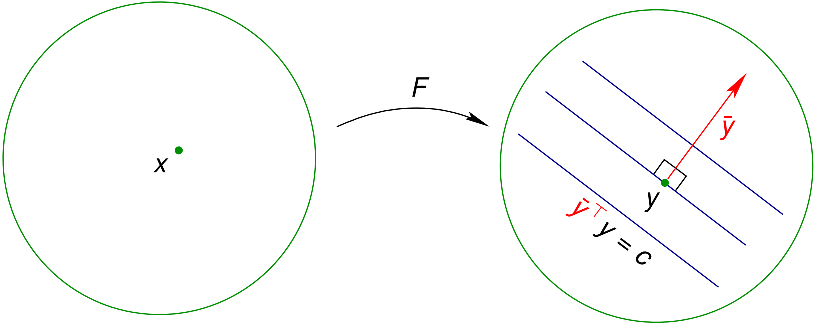


$$\dot{y}(t) = \frac{\partial}{\partial t} F(x(t)) = F'(x(t))\,\dot{x}(t) \equiv \dot{F}(x, \dot{x})$$

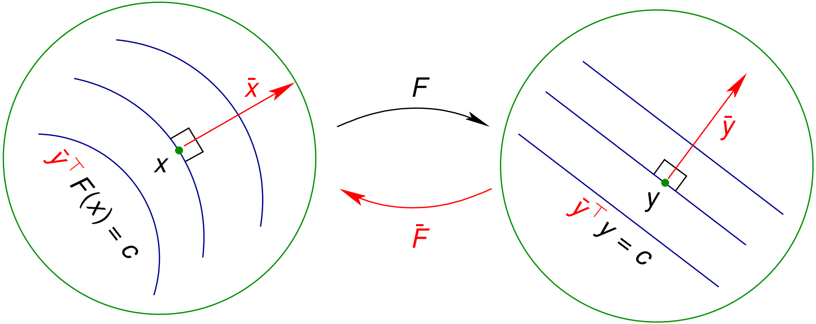
Reverse Mode AD = Discrete Adjoints



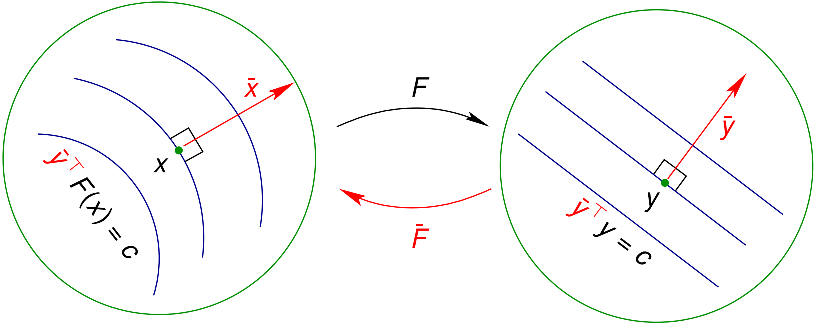
Reverse Mode AD = Discrete Adjoints



Reverse Mode AD = Discrete Adjoints



Reverse Mode AD = Discrete Adjoints



$$\bar{x} \equiv \bar{y}^\top F'(x) = \nabla_x \langle \bar{y}^\top F(x) \rangle \equiv \bar{F}(x, \bar{y})$$

Overview AD Theory and Tools

- Differentiation of computer programmes with working accuracy (Griewank, Kulshreshtha, Walther 2012)

Overview AD Theory and Tools

- Differentiation of computer programmes with working accuracy (Griewank, Kulshreshtha, Walther 2012)
- Forward mode: $\text{OPS}(F'(x)\dot{x}) \leq c\,\text{OPS}(F), \quad c \in [2, 5/2]$

Overview AD Theory and Tools

- Differentiation of computer programmes with working accuracy (Griewank, Kulshreshtha, Walther 2012)
- $$\text{Forward mode:} \quad \text{OPS}(F'(x)\dot{x}) \quad \leq \quad c\,\text{OPS}(F), \quad c \in [2, 5/2]$$

  = discrete analogon to sensitivity equation

Overview AD Theory and Tools

- Differentiation of computer programmes with working accuracy (Griewank, Kulshreshtha, Walther 2012)

- $$\begin{array}{llcll} \text{Forward mode:} & \mathrm{OPS}(F'(x)\dot{x}) & \leq & c\,\mathrm{OPS}(F), & c \in [2, 5/2] \\ \text{Reverse mode:} & \mathrm{OPS}(\bar{y}^\top F'(x)) & \leq & c\,\mathrm{OPS}(F), & c \in [3, 4] \\ & \mathrm{MEM}(\bar{y}^\top F'(x)) & \sim & \mathrm{OPS}(F), & \end{array}$$

Overview AD Theory and Tools

- Differentiation of computer programmes with working accuracy (Griewank, Kulshreshtha, Walther 2012)
- $$\begin{array}{llcll} \text{Forward mode:} & \mathrm{OPS}(F'(x)\dot{x}) & \leq & c\,\mathrm{OPS}(F), & c \in [2, 5/2] \\ \text{Reverse mode:} & \mathrm{OPS}(\bar{y}^\top F'(x)) & \leq & c\,\mathrm{OPS}(F), & c \in [3, 4] \\ & \mathrm{MEM}(\bar{y}^\top F'(x)) & \sim & \mathrm{OPS}(F), & \end{array}$$

  = discrete analogon to adjoint equation

Overview AD Theory and Tools

- Differentiation of computer programmes with working accuracy (Griewank, Kulshreshtha, Walther 2012)
- $$\begin{array}{llcll} \text{Forward mode:} & \mathrm{OPS}(F'(x)\dot{x}) & \leq & c\,\mathrm{OPS}(F), & c \in [2, 5/2] \\ \text{Reverse mode:} & \mathrm{OPS}(\bar{y}^\top F'(x)) & \leq & c\,\mathrm{OPS}(F), & c \in [3, 4] \\ & \mathrm{MEM}(\bar{y}^\top F'(x)) & \sim & \mathrm{OPS}(F), & \end{array}$$

$\Longrightarrow$ Gradients are cheap $\sim$ Function costs!!

Overview AD Theory and Tools

- Differentiation of computer programmes with working accuracy (Griewank, Kulshreshtha, Walther 2012)
- $$\begin{array}{llll} \text{Forward mode:} & \mathrm{OPS}(F'(x)\dot{x}) & \leq & c\,\mathrm{OPS}(F), \quad c \in [2, 5/2] \\ \text{Reverse mode:} & \mathrm{OPS}(\bar{y}^\top F'(x)) & \leq & c\,\mathrm{OPS}(F), \quad c \in [3, 4] \\ & \mathrm{MEM}(\bar{y}^\top F'(x)) & \sim & \mathrm{OPS}(F), \end{array}$$

$\Longrightarrow$ Gradients are cheap $\sim$ Function costs!!

- Combination: $\mathrm{OPS}(\bar{y}^\top F''(x)\dot{x}) \leq c\,\mathrm{OPS}(F), \quad c \in [7, 10]$
- Consistent derivative information!

Berlin Mathematics Research Center MATH+

Overview AD Theory and Tools

- Differentiation of computer programmes with working accuracy (Griewank, Kulshreshtha, Walther 2012)
-
$$\begin{array}{llcll} \text{Forward mode:} & \mathrm{OPS}(F'(x)\dot{x}) & \leq & c\,\mathrm{OPS}(F), & c \in [2, 5/2] \\ \text{Reverse mode:} & \mathrm{OPS}(\bar{y}^\top F'(x)) & \leq & c\,\mathrm{OPS}(F), & c \in [3, 4] \\ & \mathrm{MEM}(\bar{y}^\top F'(x)) & \sim & \mathrm{OPS}(F), & \end{array}$$

$\Longrightarrow$ Gradients are cheap $\sim$ Function costs!!

- Combination: $\mathrm{OPS}(\bar{y}^\top F''(x)\dot{x}) \leq c\,\mathrm{OPS}(F), \quad c \in [7, 10]$
- Consistent derivative information!
- AD-tools: ADOL-C, CoDiPack, Tapenade, INTLAB, ADiMat, ...
- General purpose tools: FEniCS, SU2, PyTorch, TensorFlow, ...

Overview AD Theory and Tools

- Differentiation of computer programmes with working accuracy (Griewank, Kulshreshtha, Walther 2012)
-

$$\begin{array}{llcll} \text{Forward mode:} & \mathrm{OPS}(F'(x)\dot{x}) & \leq & c\,\mathrm{OPS}(F), & c \in [2, 5/2] \\ \text{Reverse mode:} & \mathrm{OPS}(\bar{y}^\top F'(x)) & \leq & c\,\mathrm{OPS}(F), & c \in [3, 4] \\ & \mathrm{MEM}(\bar{y}^\top F'(x)) & \sim & \mathrm{OPS}(F), & \end{array}$$

$\Longrightarrow$ Gradients are cheap $\sim$ Function costs!!

- Combination: $\mathrm{OPS}(\bar{y}^\top F''(x)\dot{x}) \leq c\,\mathrm{OPS}(F), \quad c \in [7, 10]$
- Consistent derivative information!
- AD-tools: ADOL-C, CoDiPack, Tapenade, INTLAB, ADiMat, ...
- General purpose tools: FEniCS, SU2, PyTorch, TensorFlow, ...

(Griewank, Walther 2008), (Naumann 2012), www.autodiff.org

Automatic Differentiation by OverLoading in C++

- ADOL-C version 2.7, available at COIN-OR since 2009, open source (GPL or ECL)
- based on operator overloading, trace as internal representation

Automatic Differentiation by OverLoading in C++

- ADOL-C version 2.7, available at COIN-OR since 2009, open source (GPL or ECL)
- based on operator overloading, trace as internal representation
- general-purpose AD tool with focus on functionalities
- interfaces to ColPack (Purdue University) and Ipopt (COIN-OR)

Automatic Differentiation by OverLoading in C++

- ADOL-C version 2.7, available at COIN-OR since 2009, open source (GPL or ECL)
- based on operator overloading, trace as internal representation
- general-purpose AD tool with focus on functionalities
- interfaces to ColPack (Purdue University) and Ipopt (COIN-OR)
- current developments
  - Julia interface ADOLC.jl
  - exploitation of fixed-point structure for second-order derivatives
  - generalized derivatives for nonsmooth functions

Piezoelectricity

Fundamental properties:

- Transformation of **mechanical energy** into **electrical energy**
- Transformation of **electrical energy** into **mechanical energy**

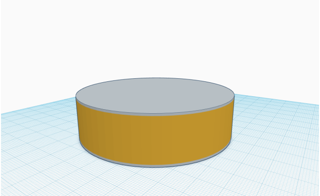
Piezoelectricity

Fundamental properties:

- Transformation of **mechanical energy** into **electrical energy**
- Transformation of **electrical energy** into **mechanical energy**

Used in wide range of applications:
Pressure Sensors, Ultrasonic Cleaning, Ultrasound Imaging, Piezoelectric Speakers, Electronic Toothbrushes, Instrument Pickups, Microphones, Piezoelectric Igniters, Electricity Generation, Tennis Racquets, . . .

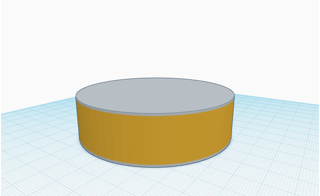
The Considered Setting



https://www.piceramic.de/de/produkte/piezokeramische-bauelemente/scheiben-staebe-und-zylinder/piezoelektrische-scheiben-1206710/

- Piezoceramics come in many shapes and sizes
  here: disk shaped ceramics (very popular, cheap(er) simulation)

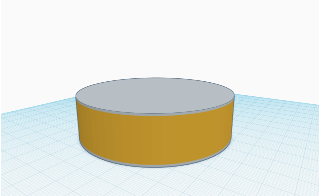
The Considered Setting



https://www.piceramic.de/de/produkte/piezokeramische-bauelemente/scheiben-staebe-und-zylinder/piezoelektrische-scheiben-1206710/

- Piezoceramics come in many shapes and sizes
  here: disk shaped ceramics (very popular, cheap(er) simulation)
- Thanks to Benjamin Jurgelucks and Veronika Schulze!
  Cooperation with Measurement Engineering Group, Prof. Henning, Univ Paderborn

The Considered Setting



https://www.piceramic.de/de/produkte/piezokeramische-bauelemente/scheiben-staebe-und-zylinder/piezoelektrische-scheiben-1206710/

- Piezoceramics come in many shapes and sizes
  here: disk shaped ceramics (very popular, cheap(er) simulation)
- Thanks to Benjamin Jurgelucks and Veronika Schulze!
  Cooperation with Measurement Engineering Group, Prof. Henning, Univ Paderborn
- Here: Consider only small loads $\Rightarrow$ Disregard thermal effects
  Nonlinear effects $\Rightarrow$ DFG research group NEPTUN

Inverse Problem - State of the Art

- Sensitivity too small for some parameter
  (using conventional methods or data provided by manufacturer)
  Up to 20% error not uncommon

Inverse Problem - State of the Art

- Sensitivity too small for some parameter
  (using conventional methods or data provided by manufacturer)
  Up to 20% error not uncommon

- Alternative approaches:
  - Use multiple differently shaped piezoceramics
    $\Rightarrow$ Leads to inconsistent datasets
  - Use additional measuremens of surface displacement
    $\Rightarrow$ Very expensive and still low sensitivity
  - Low sensitivity parameters are excluded from parameter identification

Inverse Problem - State of the Art

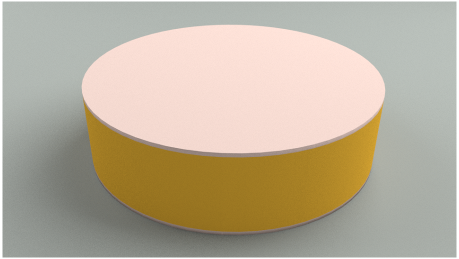
- Sensitivity too small for some parameter
  (using conventional methods or data provided by manufacturer)
  Up to 20% error not uncommon

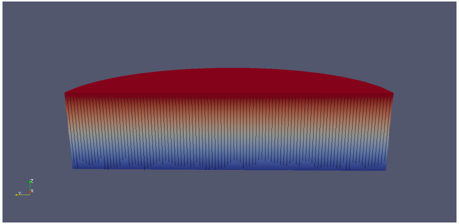
- Alternative approaches:
  - Use multiple differently shaped piezoceramics
    $\Rightarrow$ Leads to inconsistent datasets
  - Use additional measuremens of surface displacement
    $\Rightarrow$ Very expensive and still low sensitivity
  - Low sensitivity parameters are excluded from parameter identification

Goal: Identify all parameters using a single piezoceramic and impedance measurements

AD-enabled Optimization of the Electrodes

Fully covering electrodes

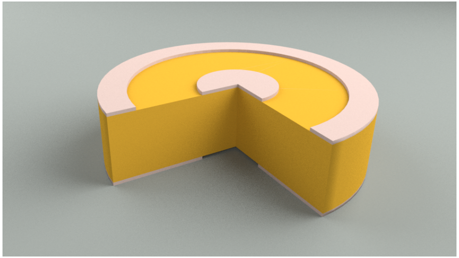


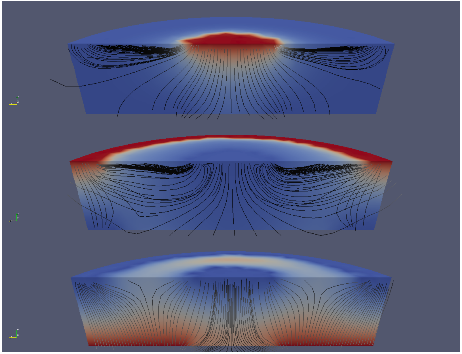


Thanks to B. Jurgelucks

AD-enabled Optimization of the Electrodes

Triple-ring electrodes





Thanks to B. Jurgelucks

Real Measurements

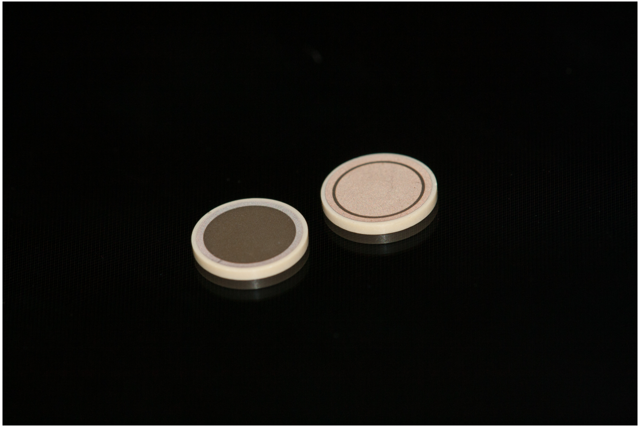
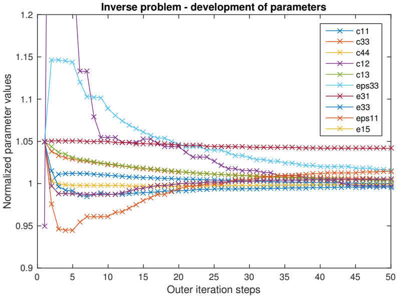
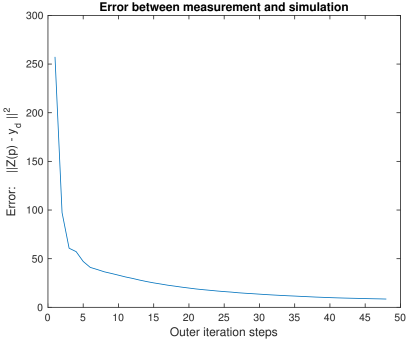


Photo by S. Olfert

Appropriate Version of AD-enabled Gauss-Newton Method





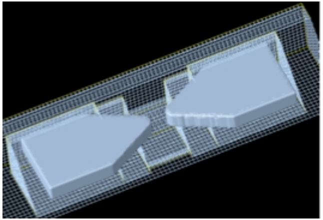
- None of the 10 (!) parameters diverges
- To the best of our knowledge this has not been possible with only one piezoceramic

Optical Nano-Structures

State of the art: Nano-structures are used to confine light

Simple example: Bow-tie antenna

- metallic nano structure
- two triangles and gap
- Size: 100 nm ($< \lambda_{\text{light}}$!)
- intensity enhancement in gap

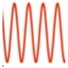


Possible Configurations

Simple setting:

- different structure

- “simple” excitation



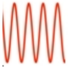
- pure metal
- extremely short dephasing

Possible Configurations

Simple setting:

- different structure

- “simple” excitation



- pure metal
- extremely short dephasing

Advanced setting:

- fixed structure

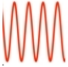
Berlin Mathematics Research Center
MATH+

Possible Configurations

Simple setting:

- different structure

- “simple” excitation

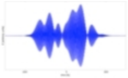


- pure metal
- extremely short dephasing

Advanced setting:

- fixed structure

- sophisticated excitation

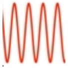


Possible Configurations

Simple setting:

- different structure

- “simple” excitation

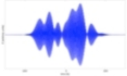


- pure metal
- extremely short dephasing

Advanced setting:

- fixed structure

- sophisticated excitation



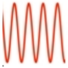
- add resonances in semiconductors

Possible Configurations

Simple setting:

- different structure

- “simple” excitation

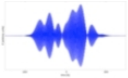


- pure metal
- extremely short dephasing

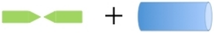
Advanced setting:

- fixed structure

- sophisticated excitation



- add resonances in semiconductors
- longer dephasing

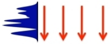


Berlin Mathematics Research Center MATH+

Test Case: Quantum Wire

Cooperation with T. Meier, M. Reichelt, Dep. Physik, Uni Paderborn

Generic configuration:

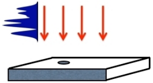


$\longleftarrow$ adaptable light puls $E(t)$

Test Case: Quantum Wire

Cooperation with T. Meier, M. Reichelt, Dep. Physik, Uni Paderborn

Generic configuration:



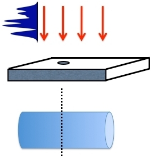
$\longleftarrow$ adaptable light puls $E(t)$

$\longleftarrow$ metal aperture with air hole

Test Case: Quantum Wire

Cooperation with T. Meier, M. Reichelt, Dep. Physik, Uni Paderborn

Generic configuration:



⟵ adaptable light puls $E(t)$

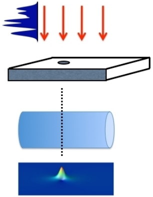
⟵ metal aperture with air hole

⟵ quantum wire

Test Case: Quantum Wire

Cooperation with T. Meier, M. Reichelt, Dep. Physik, Uni Paderborn

Generic configuration:



⟵ adaptable light puls $E(t)$

⟵ metal aperture with air hole

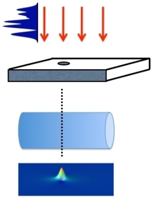
⟵ quantum wire

⟵ electron distribution?

Test Case: Quantum Wire

Cooperation with T. Meier, M. Reichelt, Dep. Physik, Uni Paderborn

Generic configuration:



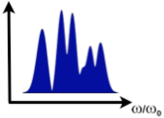
$\longleftarrow$ adaptable light puls $E(t)$

$\longleftarrow$ metal aperture with air hole

$\longleftarrow$ quantum wire

$\longleftarrow$ electron distribution?

Light puls:

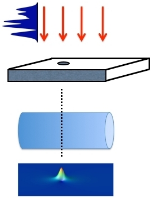


with $E(t) = \sum A_i \exp\left(-\left(\frac{t-t_i}{\delta t_i}\right)^2\right)\cos(\omega_i t + \phi_i)$

Test Case: Quantum Wire

Cooperation with T. Meier, M. Reichelt, Dep. Physik, Uni Paderborn

Generic configuration:



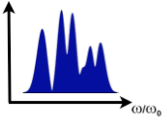
$\longleftarrow$ adaptable light puls $E(t)$

$\longleftarrow$ metal aperture with air hole

$\longleftarrow$ quantum wire

$\longleftarrow$ electron distribution?

Light puls:



with $E(t) = \sum A_i \exp\left(-\left(\frac{t-t_i}{\delta t_i}\right)^2\right) \cos(\omega_i t + \phi_i)$

Parameter: $A_i, \phi_i, \omega_i, t_i$ $\Rightarrow$ up to 120!

Mathematical Formulation

State equation:

$$
\begin{aligned}
\frac{\partial}{\partial t} p &= \frac{i}{\hbar}(\epsilon_0 - \epsilon_1)p + \frac{i}{\hbar}\mathbf{E}(t)\cdot\mathbf{d}\,(n_0 - n_1) \\
\frac{\partial}{\partial t} n_0 &= \frac{2}{\hbar}\mathrm{Im}\,[\mathbf{E}(t)\cdot\mathbf{d}\,p^*] \\
\frac{\partial}{\partial t} n_1 &= -\frac{2}{\hbar}\mathrm{Im}\,[\mathbf{E}(t)\cdot\mathbf{d}\,p^*] \\
1 &= n_1 + n_0
\end{aligned}
$$

$\Rightarrow$ Three complex-valued coupled differential equations
p, n_0 and n_1 distributed in space.

Mathematical Formulation

State equation:

$$\frac{\partial}{\partial t} p = \frac{i}{\hbar}(\epsilon_0 - \epsilon_1)p + \frac{i}{\hbar}\mathbf{E}(t) \cdot \mathbf{d}\,(n_0 - n_1)$$
$$\frac{\partial}{\partial t} n_0 = \frac{2}{\hbar}\mathrm{Im}\,[\mathbf{E}(t) \cdot \mathbf{d}\, p^*]$$
$$\frac{\partial}{\partial t} n_1 = -\frac{2}{\hbar}\mathrm{Im}\,[\mathbf{E}(t) \cdot \mathbf{d}\, p^*]$$
$$1 = n_1 + n_0$$

Target:
Maximize energy at given time and given place with constant energy

Mathematical Formulation

State equation:

$$
\begin{aligned}
\frac{\partial}{\partial t} p &= \frac{i}{\hbar}(\epsilon_0 - \epsilon_1)p + \frac{i}{\hbar}\mathbf{E}(t) \cdot \mathbf{d}\,(n_0 - n_1) \\
\frac{\partial}{\partial t} n_0 &= \frac{2}{\hbar}\mathrm{Im}\,[\mathbf{E}(t) \cdot \mathbf{d}\,p^*] \\
\frac{\partial}{\partial t} n_1 &= -\frac{2}{\hbar}\mathrm{Im}\,[\mathbf{E}(t) \cdot \mathbf{d}\,p^*] \\
1 &= n_1 + n_0
\end{aligned}
$$

Target:
Maximize energy at given time and given place with constant energy

= Maximize emitted radiation

$$
I_{rad} = \left|\omega^2 P(\omega)\right|^2 = \left|\omega^2\, 2\,\mathrm{Re}(d\,p)\right|^2
$$

Function Evaluation of $I_{rad}(f(g(x)))$

```
x[] ← (phase[], amplitude[], width[], point[])

for time=0 to Tfinal do
    if (time >= Tobs && time < Tobs+dt)
        eval_time_step1(x,int_tar)
    else
        eval_time_step2(x,int_tar)
    end if
end for

eval_target(int_tar,fitness)
```

Function Evaluation of $I_{rad}(f(g(x)))$

```
x[] ← (phase[], amplitude[], width[], point[])

for time=0 to Tfinal do
      if (time >= Tobs && time < Tobs+dt)
          eval_time_step1(x,int_tar)
      else
          eval_time_step2(x,int_tar)
      end if
 end for

eval_target(int_tar,fitness)
```

scenarios:

independents $\in \{20, 60, 120\}$ $\Rightarrow$ Reverse mode!

Function Evaluation of $I_{rad}(f(g(x)))$

```
x[] ← (phase[], amplitude[], width[], point[])

for time=0 to Tfinal do
      if (time >= Tobs && time < Tobs+dt)
           eval_time_step1(x,int_tar)
      else
           eval_time_step2(x,int_tar)
      end if
 end for

eval_target(int_tar,fitness)
```

scenarios:

# independents $\in \{20, 60, 120\}$	$\Rightarrow$	Reverse mode!
# time steps $\in \{16000, 32000, 160000\}$	$\Rightarrow$	Checkpointing!

Quantum Wire: Optimization

So far: Genetic algorithms

Now: L-BFGS and efficient gradient computation

- ADOL-C coupled with hand-coded adjoints
- Checkpointing (160 000 time steps!!)

$\Rightarrow$ TIME(gradient)/TIME(target function) < 7 despite of checkpointing!

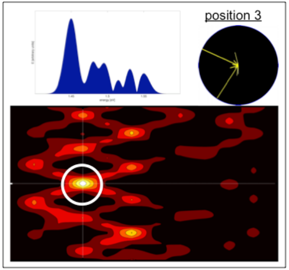
Quantum Wire: Optimization

So far: Genetic algorithms

Now: L-BFGS and efficient gradient computation

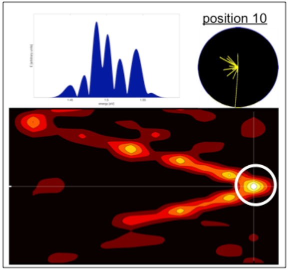
- ADOL-C coupled with hand-coded adjoints
- Checkpointing (160 000 time steps!!)

$\Rightarrow$ TIME(gradient)/TIME(target function) $<$ 7 despite of checkpointing!

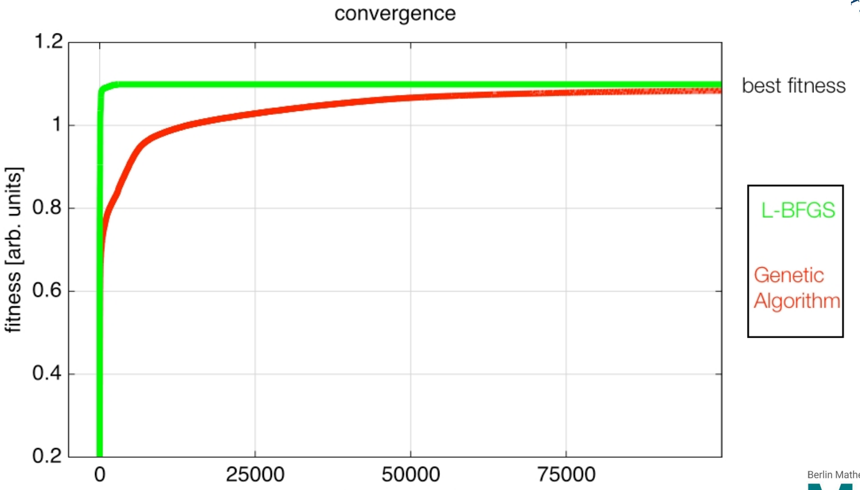


excite
•at **same** position
•at **same** time
•with **same** energy

optimize
•for **same** t_{opt}
•**different** positions



Quantum Wire: Comparison



Photonic Nano-Resonators

Joint work with F. Binkowski, J. Kullig, F. Betz, L. Zschiedrich, J. Wiersig, S. Burger

Applications:

- probing single molecules with ultrahigh sensitivity
- designing nanoantennas with a tailored directivity
- large spontaneous emission rate or realizing efficient single-photon sources

Photonic Nano-Resonators

Joint work with F. Binkowski, J. Kullig, F. Betz, L. Zschiedrich, J. Wiersig, S. Burger

Applications:

- probing single molecules with ultrahigh sensitivity
- designing nanoantennas with a tailored directivity
- large spontaneous emission rate or realizing efficient single-photon sources

Furter aspects:

- computed by solving the source-free Maxwell's equations

Berlin Mathematics Research Center MATH+

Photonic Nano-Resonators

Joint work with F. Binkowski, J. Kullig, F. Betz, L. Zschiedrich, J. Wiersig, S. Burger

Applications:

- probing single molecules with ultrahigh sensitivity
- designing nanoantennas with a tailored directivity
- large spontaneous emission rate or realizing efficient single-photon sources

Furter aspects:

- computed by solving the source-free Maxwell's equations
- sensitivities of resonances are of interest for
  - better understanding of the underlying physical effects
  - an efficient optimization of corresponding photonic devices

Behaviour of Sensitivities at EPs

Exceptional points: spectral degeneracies
(eigenfrequencies and eigenmodes coalesce)

Behaviour of Sensitivities at EPs

Exceptional points: spectral degeneracies
(eigenfrequencies and eigenmodes coalesce)

- parametric fine-tuning is needed to achieve such non-Hermitian degeneracies
  $\Rightarrow$ exceptional points (EPs)

Berlin Mathematics Research Center

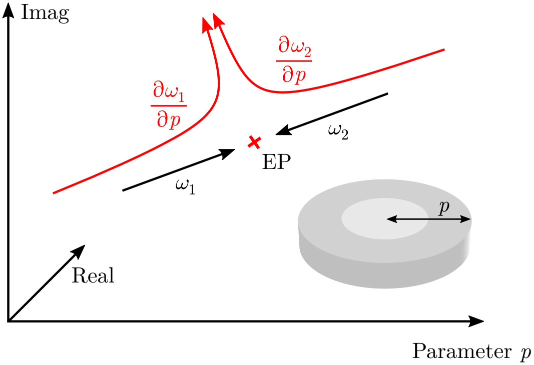
Behaviour of Sensitivities at EPs

Exceptional points: spectral degeneracies
(eigenfrequencies and eigenmodes coalesce)

- parametric fine-tuning is needed to achieve such non-Hermitian degeneracies
  $\Rightarrow$ exceptional points (EPs)
- EPs have been connected to many interesting effects including
  - ultra-sensitive sensors
  - control of light transport
  - electromagnetically induced transparency
  - optical amplifiers,...

Behaviour of Sensitivities at EPs

Exceptional points: spectral degeneracies
(eigenfrequencies and eigenmodes coalesce)

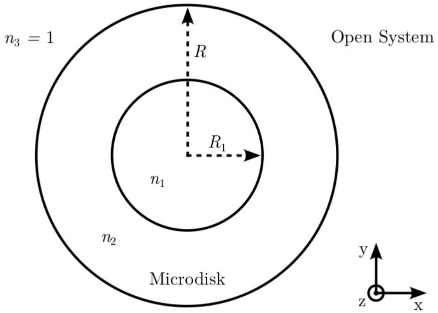


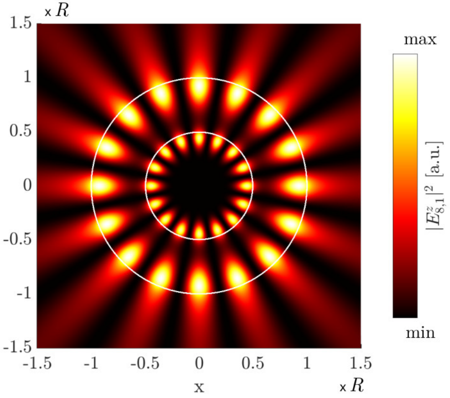
Calculation of EPs and Their Sensitivities

- RPExpand
  Software for Riesz projection expansion of resonance phenomena
  by F. Betz, F. Binkowski, S. Burger

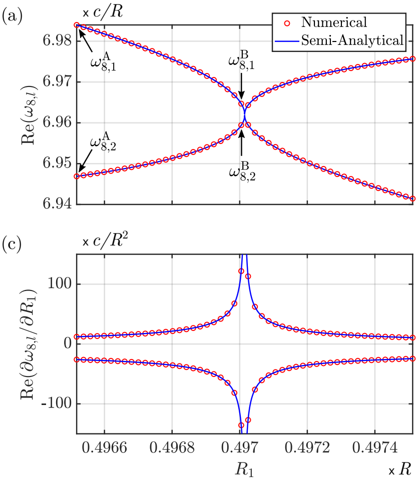
- JCMSuite
  part of JCMWave (commercial software)
  based on FEM combined with efficient contour-integral method
  implements own AD approach

The Considered Setting





Eigenfrequencies and Their Sensitivities



Conclusions

- Basics of Algorithmic Differentiation
  - Efficient evaluation of derivatives with working accuracy
  - Theory for basic modes complete, advanced AD?
  - Various tools/implementations available

Conclusions

- Basics of Algorithmic Differentiation
  - Efficient evaluation of derivatives with working accuracy
  - Theory for basic modes complete, advanced AD?
  - Various tools/implementations available
- Structure exploitation indispensable

Conclusions

- Basics of Algorithmic Differentiation
  - Efficient evaluation of derivatives with working accuracy
  - Theory for basic modes complete, advanced AD?
  - Various tools/implementations available
- Structure exploitation indispensable
- AD for applications in physics:
  - parameter identification for piezoceramics[1]
  - optimized excitation of nano antenna[2]
  - sensitivities of exceptional points of nano-resonators[3]

[1] e.g., L. Claes et al.: Inverse procedure for measuring piezoelectric material parameters using a single multi-electrode sample. J. Sensors and Sensor Systems 12 (1) (2023)

[2] e.g., M. Reichelt, A. Walther, T. Meier: Tailoring the high-harmonic emission in two-level systems and semiconductors by pulse shaping. JOSA B 29 (2) (2012)

[3] F. Binkowski, J. Kullig, F. Betz, L. Zschiedrich, A. Walther, J. Wiersig, S. Burger: Computing eigenfrequency sensitivities near exceptional points,Phys. Rev. Research 6 (2) (2024)