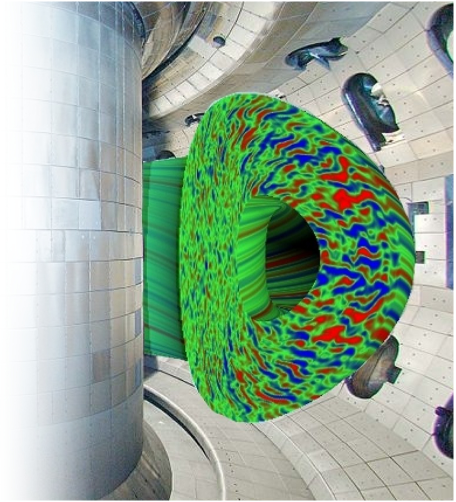
Useful Numerical Methods for Modern Computation

by
J. Candy



Presented at the
CPS-FR 2019 (MIT)
Cambridge, MA
26 August 2019

Goal and Outline

Simple illustrations of the power and elegance of numerical methods

Goal and Outline

Simple illustrations of the power and elegance of numerical methods

1. General Atomics history

Goal and Outline

Simple illustrations of the power and elegance of numerical methods

1. General Atomics history
2. Surprises in numerical integration

Goal and Outline

Simple illustrations of the power and elegance of numerical methods

1. General Atomics history
2. Surprises in numerical integration
3. Orthogonal polynomials (Legendre, Chebyshev) and curve fitting

Goal and Outline

Simple illustrations of the power and elegance of numerical methods

1. General Atomics history
2. Surprises in numerical integration
3. Orthogonal polynomials (Legendre, Chebyshev) and curve fitting
4. A pseudospectral method for pitch-angle scattering

Goal and Outline

Simple illustrations of the power and elegance of numerical methods

1. General Atomics history
2. Surprises in numerical integration
3. Orthogonal polynomials (Legendre, Chebyshev) and curve fitting
4. A pseudospectral method for pitch-angle scattering
5. Waves and how to advect them

Goal and Outline

Simple illustrations of the power and elegance of numerical methods

1. General Atomics history
2. Surprises in numerical integration
3. Orthogonal polynomials (Legendre, Chebyshev) and curve fitting
4. A pseudospectral method for pitch-angle scattering
5. Waves and how to advect them
6. Radial basis functions (RBF) in Plasma Physics

Goal and Outline

Simple illustrations of the power and elegance of numerical methods

1. General Atomics history
2. Surprises in numerical integration
3. Orthogonal polynomials (Legendre, Chebyshev) and curve fitting
4. A pseudospectral method for pitch-angle scattering
5. Waves and how to advect them
6. Radial basis functions (RBF) in Plasma Physics
7. Method of Fundamental Solutions (MFS) in Acoustics

Who is General Atomics?

Who is General Atomics?

1. **General Atomics** (GA) is a private contractor in San Diego

Who is General Atomics?

1. **General Atomics** (GA) is a private contractor in San Diego
2. The GA **Magnetic Fusion** division does DOE-funded research

Who is General Atomics?

1. **General Atomics** (GA) is a private contractor in San Diego
2. The GA **Magnetic Fusion** division does DOE-funded research
3. Hosts **DIII-D National Fusion Facility**

Founded on July 18, 1955 (photo 1957)

The General Atomic Division of General Dynamics

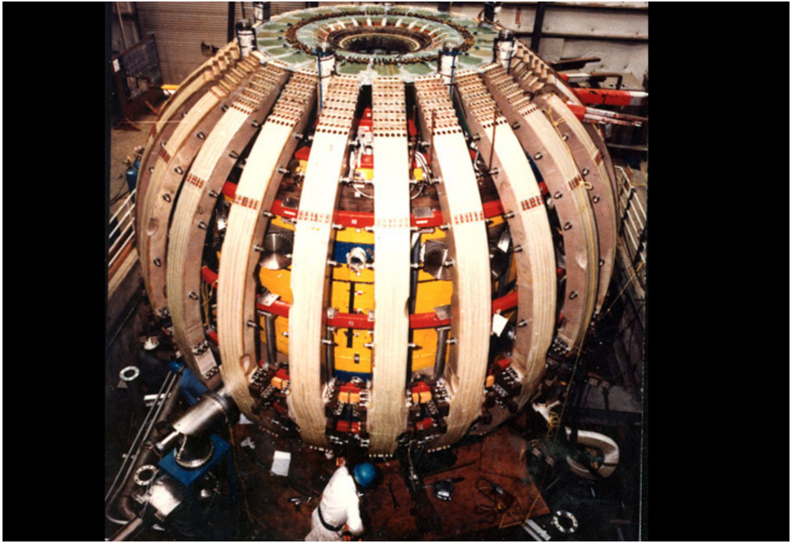
Laboratory formally dedicated on June 25th, 1959

John Jay Hopkins Laboratory for Pure and Applied Science

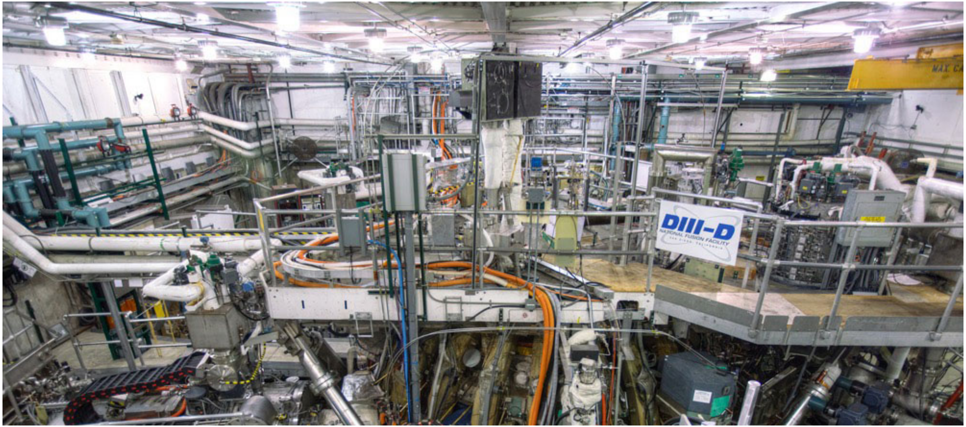
Present-day Campus (2019)

Retains feel of early architecture

Doublet III (1974)



DIII-D (Present day)



Surprises in numerical integration

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\pi x}$$

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\pi x}$$

Trapezoidal Rule

$$\int_a^b f(x)\,dx \simeq \Delta x \left[\frac{1}{2} f(x_0) + f(x_1) + f(x_2) + f(x_3) + \frac{1}{2} f(x_4) \right]$$

Simpson's Rule

$$\int_a^b f(x)\,dx \simeq \Delta x \left[\frac{1}{3} f(x_0) + \frac{4}{3} f(x_1) + \frac{2}{3} f(x_2) + \frac{4}{3} f(x_3) + \frac{1}{3} f(x_4) \right]$$

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\pi x} \simeq 7.352155820749956$$

n	E_{Trap}	E_{Simp}
5	1.453	1.907×10^{-1}

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\pi x} \simeq 7.352155820749956$$

n	$E_{\rm Trap}$	$E_{\rm Simp}$
5	1.453	1.907×10^{-1}
7	6.599×10^{-1}	4.337×10^{-2}

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\pi x} \simeq 7.352155820749956$$

n	E_{Trap}	E_{Simp}
5	1.453	1.907×10^{-1}
7	6.599×10^{-1}	4.337×10^{-2}
9	3.741×10^{-1}	1.447×10^{-2}

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\pi x} \simeq 7.352155820749956$$

n	E_{Trap}	E_{Simp}
5	1.453	1.907×10^{-1}
7	6.599×10^{-1}	4.337×10^{-2}
9	3.741×10^{-1}	1.447×10^{-2}
11	2.403×10^{-1}	6.079×10^{-3}

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\pi x} \simeq 7.352155820749956$$

n	E_{Trap}	E_{Simp}
5	1.453	1.907×10^{-1}
7	6.599×10^{-1}	4.337×10^{-2}
9	3.741×10^{-1}	1.447×10^{-2}
11	2.403×10^{-1}	6.079×10^{-3}
13	1.672×10^{-1}	2.973×10^{-3}

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\pi x} \simeq 7.352155820749956$$

n	E_{Trap}	E_{Simp}
5	1.453	1.907×10^{-1}
7	6.599×10^{-1}	4.337×10^{-2}
9	3.741×10^{-1}	1.447×10^{-2}
11	2.403×10^{-1}	6.079×10^{-3}
13	1.672×10^{-1}	2.973×10^{-3}
15	1.230×10^{-1}	1.618×10^{-3}

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\pi x} \simeq 7.352155820749956$$

n	E_{Trap}	E_{Simp}
5	1.453	1.907×10^{-1}
7	6.599×10^{-1}	4.337×10^{-2}
9	3.741×10^{-1}	1.447×10^{-2}
11	2.403×10^{-1}	6.079×10^{-3}
13	1.672×10^{-1}	2.973×10^{-3}
15	1.230×10^{-1}	1.618×10^{-3}
101	2.419×10^{-3}	6.363×10^{-7}

Numerical integration
Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\pi x} \simeq 7.352155820749956$$

n	E_{Trap}	E_{Simp}
5	1.453	1.907×10^{-1}
7	6.599×10^{-1}	4.337×10^{-2}
9	3.741×10^{-1}	1.447×10^{-2}
11	2.403×10^{-1}	6.079×10^{-3}
13	1.672×10^{-1}	2.973×10^{-3}
15	1.230×10^{-1}	1.618×10^{-3}
101	2.419×10^{-3}	6.363×10^{-7}

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\cos(\pi x)}$$

$$I = \int_{-1}^{1} dx\, e^{\cos(\pi x)}$$

Trapezoidal Rule (periodic)

$$\int_a^b f(x)\, dx \simeq \Delta x \left[f(x_0) + f(x_1) + f(x_2) + f(x_3) \right]$$

Simpson's Rule (periodic)

$$\int_a^b f(x)\, dx \simeq \Delta x \left[\frac{2}{3} f(x_0) + \frac{4}{3} f(x_1) + \frac{2}{3} f(x_2) + \frac{4}{3} f(x_3) \right]$$

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\cos(\pi x)} \simeq 2.53213175550401 7$$

n	$E_{\rm Trap}$	$E_{\rm Simp}$
5	1.095×10^{-2}	-1.701×10^{-1}

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\cos(\pi x)} \simeq 2.53213175550401 7$$

n	E_{Trap}	E_{Simp}
5	1.095×10^{-2}	-1.701×10^{-1}
7	8.995×10^{-5}	2.965×10^{-2}

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\cos(\pi x)} \simeq 2.53213175550401 7$$

n	E_{Trap}	E_{Simp}
5	1.095×10^{-2}	-1.701×10^{-1}
7	8.995×10^{-5}	2.965×10^{-2}
9	3.984×10^{-7}	-3.649×10^{-3}

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\cos(\pi x)} \simeq 2.53213175550401 7$$

n	$E_{\rm Trap}$	$E_{\rm Simp}$
5	1.095×10^{-2}	-1.701×10^{-1}
7	8.995×10^{-5}	2.965×10^{-2}
9	3.984×10^{-7}	-3.649×10^{-3}
11	1.101×10^{-9}	3.620×10^{-4}

Numerical integration

Illustrative example

$$I = \int_{-1}^{1} dx\, e^{\cos(\pi x)} \simeq 2.53213175550401 7$$

n	E_{Trap}	E_{Simp}
5	1.095×10^{-2}	-1.701×10^{-1}
7	8.995×10^{-5}	2.965×10^{-2}
9	3.984×10^{-7}	-3.649×10^{-3}
11	1.101×10^{-9}	3.620×10^{-4}
13	2.077×10^{-12}	-2.998×10^{-5}

Numerical integration

Excellent Resource

L. Trefethen, *The Exponentially Convergent Trapezoidal Rule*,
SIAM Review **56** (2014) 385

Orthogonal polynomials (Legendre, Chebyshev) and curve fitting

Polynomial Interpolation

Ancient method: Legendre Series Expansion

$$y(x) = \frac{1}{\epsilon^2 + x^2}$$

$$y(x) = \frac{1}{\epsilon^2 + x^2}$$

1. Write as Legendre series

$$y(x) = \frac{1}{\epsilon^2 + x^2} = \sum_{n=0}^{N-1} c_n P_n(x)$$

Polynomial Interpolation

Ancient method: Legendre Series Expansion

$$y(x) = \frac{1}{\epsilon^2 + x^2}$$

1. Write as Legendre series

$$y(x) = \frac{1}{\epsilon^2 + x^2} = \sum_{n=0}^{N-1} c_n P_n(x)$$

2. Multiply by P_m, integrate, use orthogonality

$$c_n = \left(n + \frac{1}{2}\right) \int_{-1}^{1} dx \, y(x) P_n(x)$$

Polynomial Interpolation

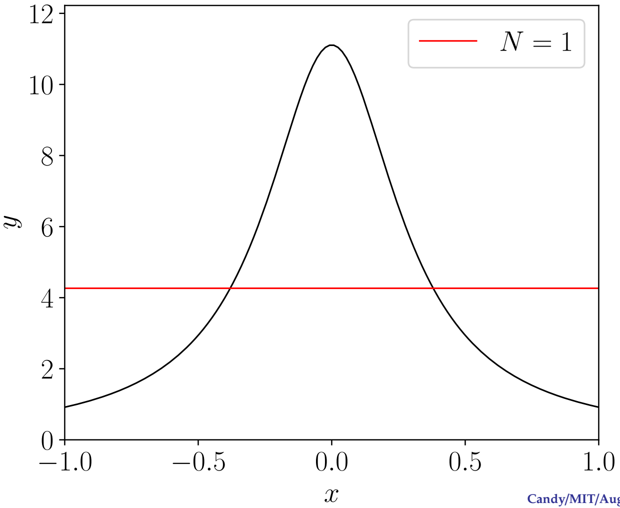
Ancient method: Legendre Series Expansion

$$\frac{1}{(0.3)^2 + x^2} = \sum_{n=0}^{N-1} c_n P_n(x)$$

Polynomial Interpolation

Ancient method: Legendre Series Expansion

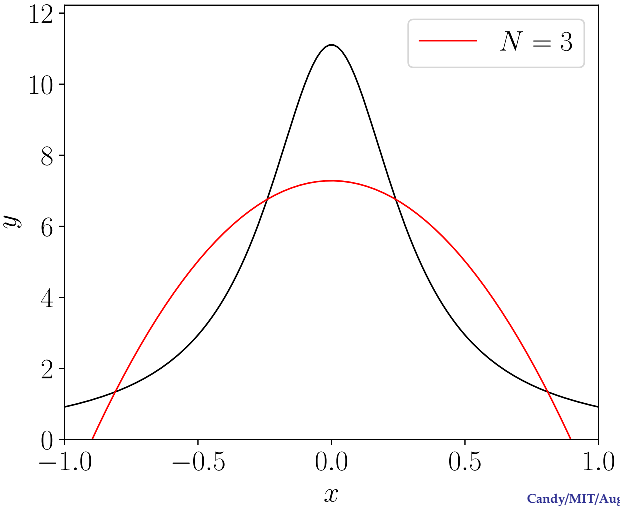
$$\frac{1}{(0.3)^2 + x^2} = \sum_{n=0}^{N-1} c_n P_n(x)$$



Polynomial Interpolation

Ancient method: Legendre Series Expansion

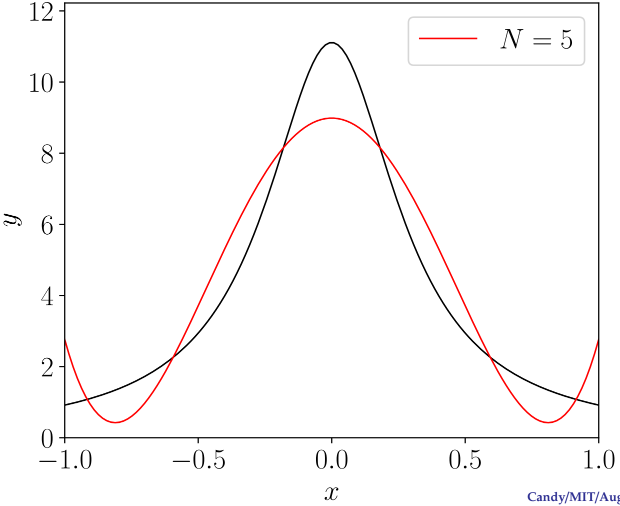
$$\frac{1}{(0.3)^2 + x^2} = \sum_{n=0}^{N-1} c_n P_n(x)$$



Polynomial Interpolation

Ancient method: Legendre Series Expansion

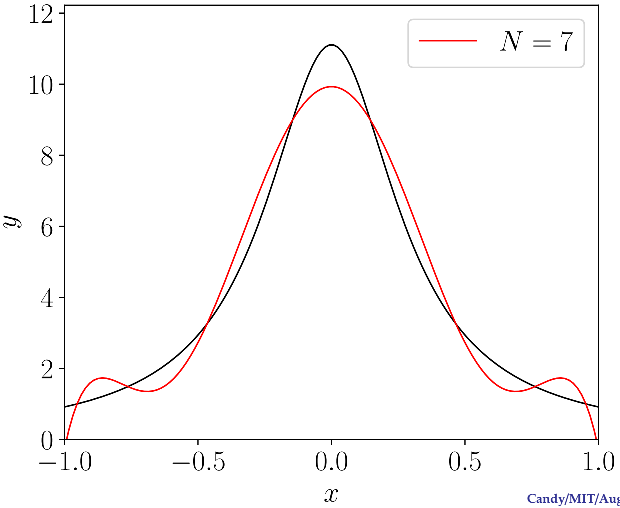
$$\frac{1}{(0.3)^2 + x^2} = \sum_{n=0}^{N-1} c_n P_n(x)$$



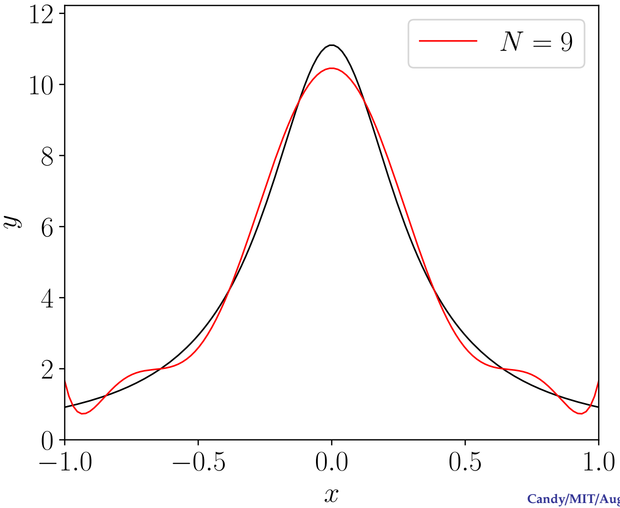
Polynomial Interpolation

Ancient method: Legendre Series Expansion

$$\frac{1}{(0.3)^2 + x^2} = \sum_{n=0}^{N-1} c_n P_n(x)$$



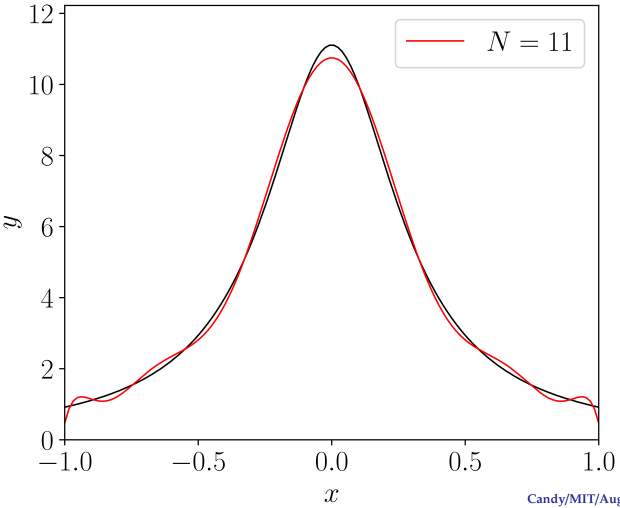
$$\frac{1}{(0.3)^2 + x^2} = \sum_{n=0}^{N-1} c_n P_n(x)$$



Polynomial Interpolation

Ancient method: Legendre Series Expansion

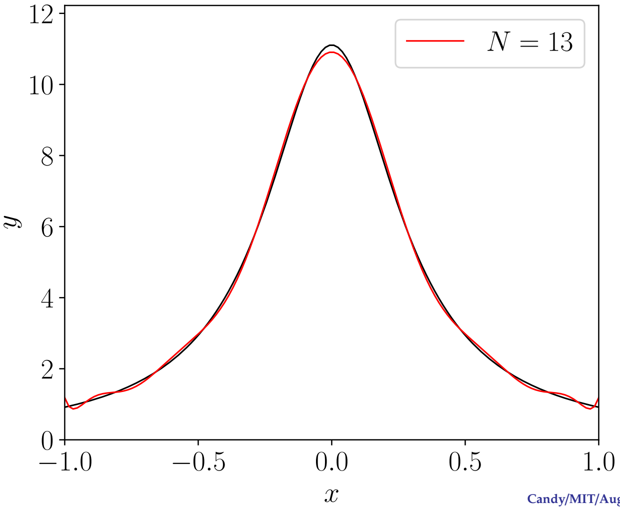
$$\frac{1}{(0.3)^2 + x^2} = \sum_{n=0}^{N-1} c_n P_n(x)$$



Polynomial Interpolation

Ancient method: Legendre Series Expansion

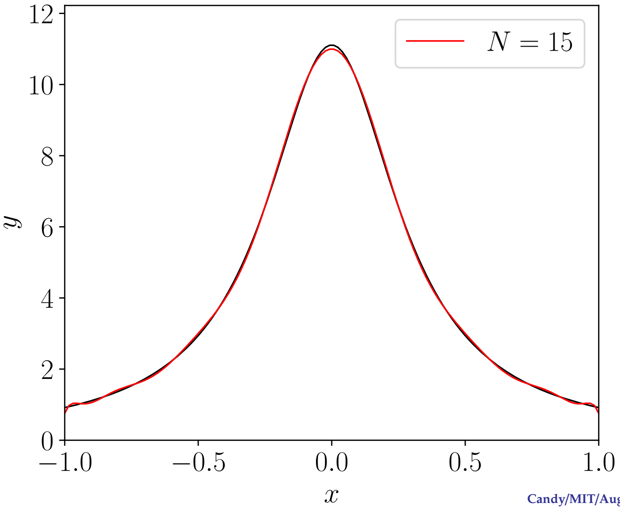
$$\frac{1}{(0.3)^2 + x^2} = \sum_{n=0}^{N-1} c_n P_n(x)$$



Polynomial Interpolation

Ancient method: Legendre Series Expansion

$$\frac{1}{(0.3)^2 + x^2} = \sum_{n=0}^{N-1} c_n P_n(x)$$

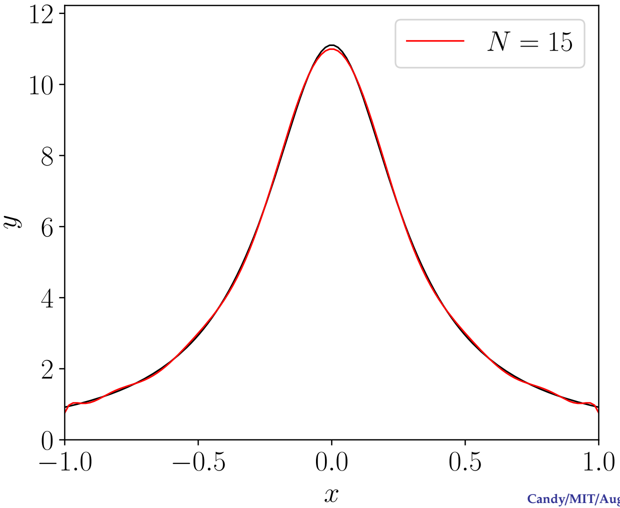


Polynomial Interpolation

Ancient method: Legendre Series Expansion

$$\frac{1}{(0.3)^2 + x^2} = \sum_{n=0}^{14} c_n P_n(x)$$
$$= \sum_{n=0}^{14} a_n x^n$$

Just a finite sum of **monomials**



Conclusion:

Polynomial approximation is good

Conclusion:
Polynomial approximation is good

Question:
So why bother with orthogonal polynomials?

Polynomial Interpolation

So we can use a simple interpolating polynomial, right?

$$\underbrace{\frac{1}{(0.3)^2 + x_i^2}}_{y_i} = \sum_{n=0}^{14} a_n x_i^n$$

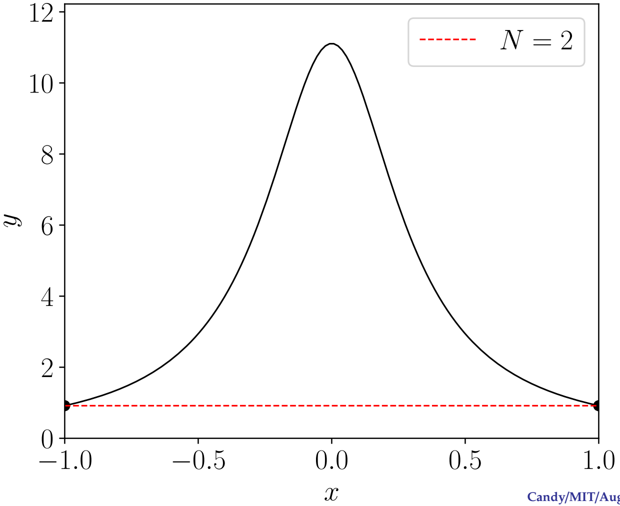
This is just a linear system: $y_i = M_{in} a_n$

Polynomial Interpolation

Fit interpolating polynomial at equally-spaced points

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ are equally-spaced on [-1,1]

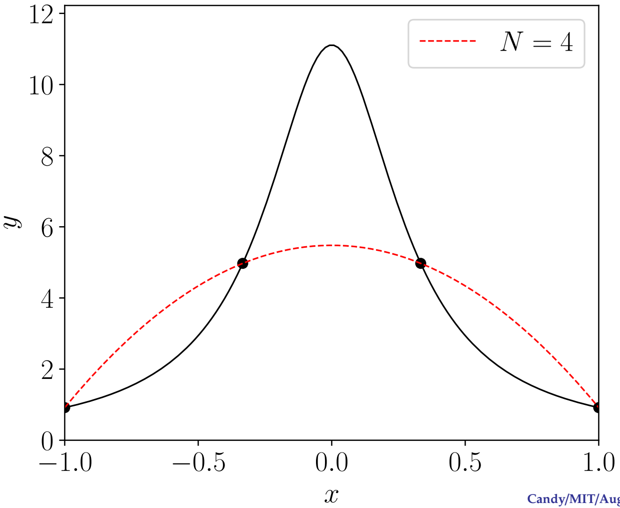


Polynomial Interpolation

Fit interpolating polynomial at equally-spaced points

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ are equally-spaced on [-1,1]

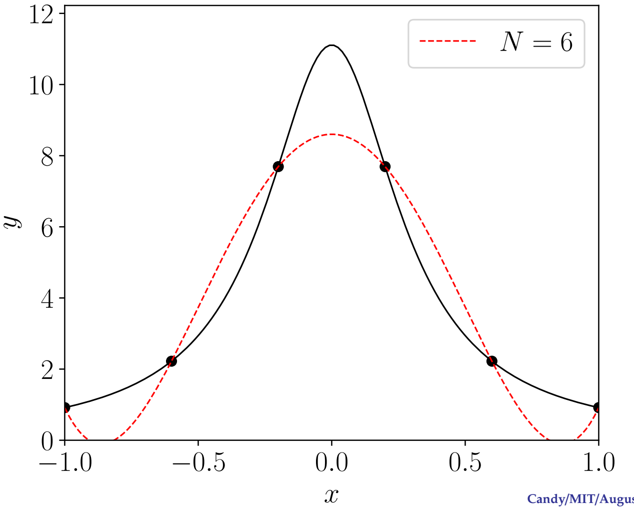


Polynomial Interpolation

Fit interpolating polynomial at equally-spaced points

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ are equally-spaced on [-1,1]

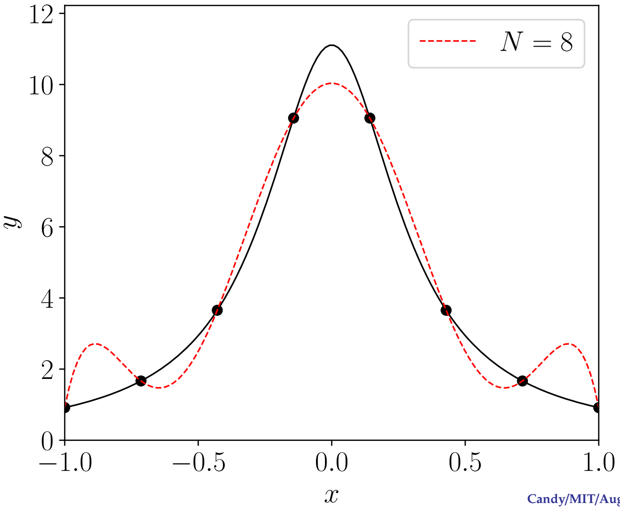


Polynomial Interpolation

Fit interpolating polynomial at equally-spaced points

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ are equally-spaced on [-1,1]

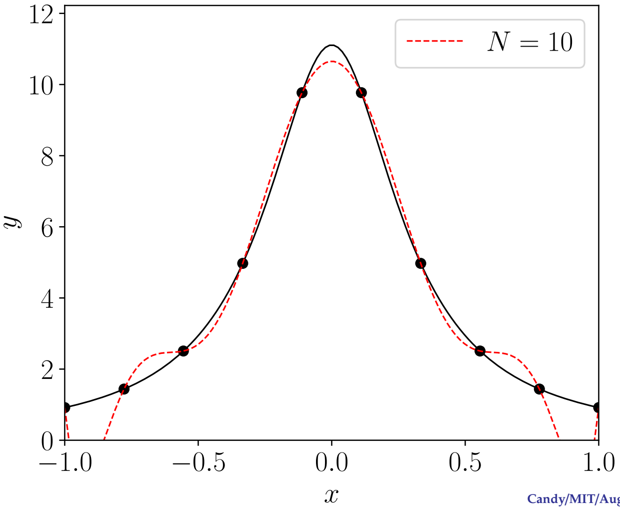


Polynomial Interpolation

Fit interpolating polynomial at equally-spaced points

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ are equally-spaced on [-1,1]

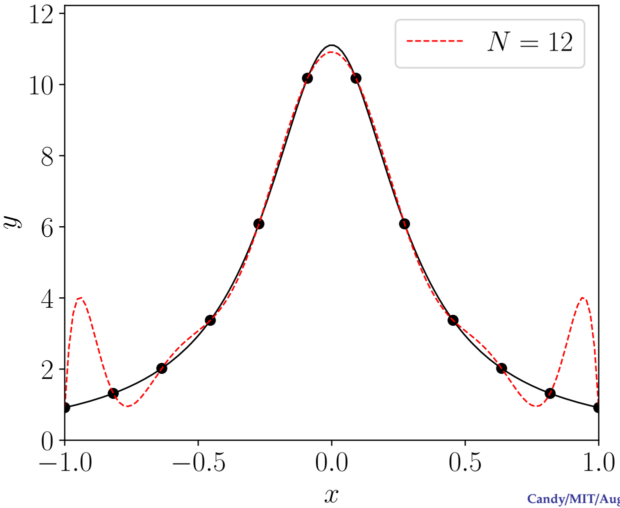


Polynomial Interpolation

Fit interpolating polynomial at equally-spaced points

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ are equally-spaced on [-1,1]

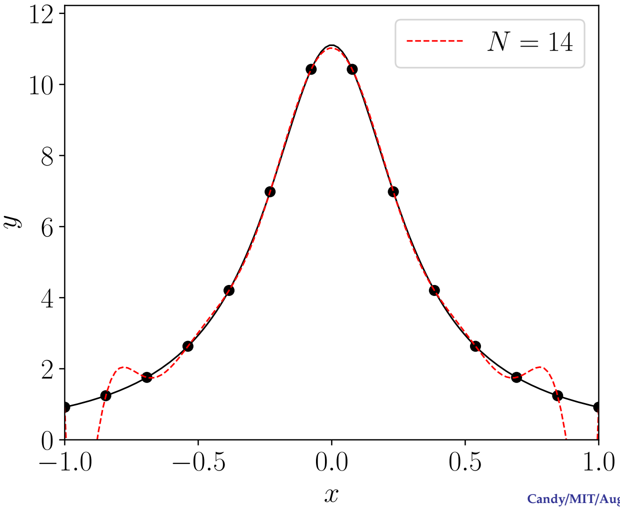


Polynomial Interpolation

Fit interpolating polynomial at equally-spaced points

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ are equally-spaced on [-1,1]

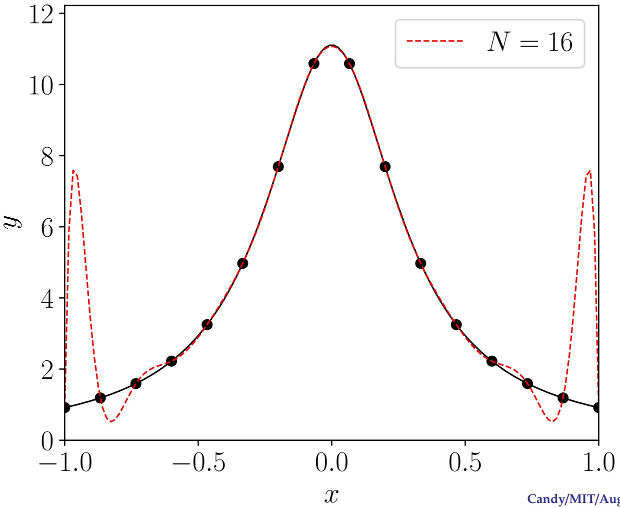


Polynomial Interpolation

Fit interpolating polynomial at equally-spaced points

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ are equally-spaced on [-1,1]



Updated conclusion:

Polynomial approximation can fail if applied naively

Orthogonal Polynomials

Some reminders

Legendre (1752-1833)

$$\int_{-1}^{1} dx\, P_m(x) P_n(x) = \frac{\delta_{mn}}{n + 1/2}$$

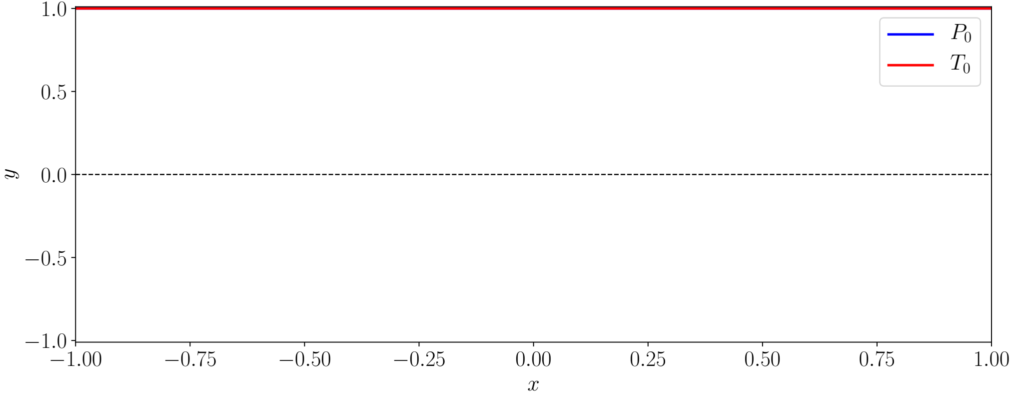
Chebyshev (1821-1894)

$$\int_{-1}^{1} \frac{dx}{\sqrt{1-x^2}}\, T_m(x) T_n(x) = \frac{\pi}{2}\, \delta_{mn}(1 + \delta_{n0})$$

Orthogonal Polynomials

Some reminders

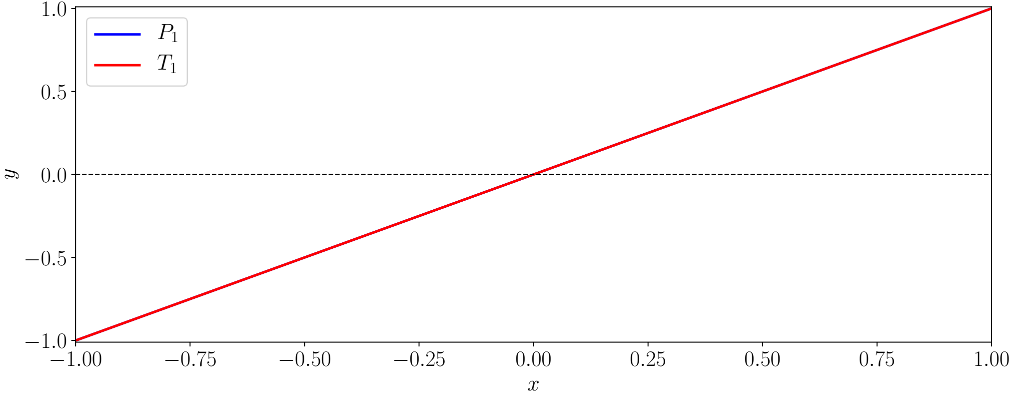
$$\int_{-1}^{1} dx\, P_m(x) P_n(x) = \frac{\delta_{mn}}{n+1/2} \qquad \int_{-1}^{1} \frac{dx}{\sqrt{1-x^2}} T_m(x) T_n(x) = \frac{\pi}{2} \delta_{mn}(1+\delta_{n0})$$



Orthogonal Polynomials

Some reminders

$$\int_{-1}^{1} dx\, P_m(x) P_n(x) = \frac{\delta_{mn}}{n + 1/2} \qquad \int_{-1}^{1} \frac{dx}{\sqrt{1 - x^2}} T_m(x) T_n(x) = \frac{\pi}{2} \delta_{mn} (1 + \delta_{n0})$$

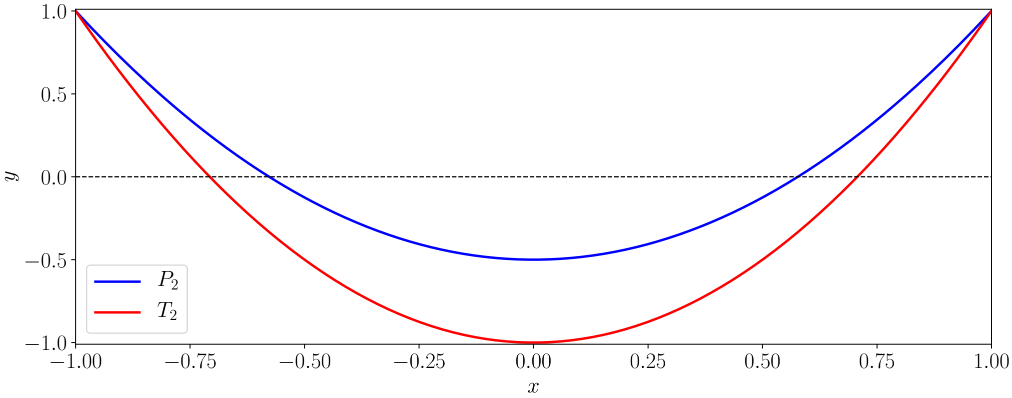


Orthogonal Polynomials

Some reminders

$$\int_{-1}^{1} dx\, P_m(x) P_n(x) = \frac{\delta_{mn}}{n+1/2}$$

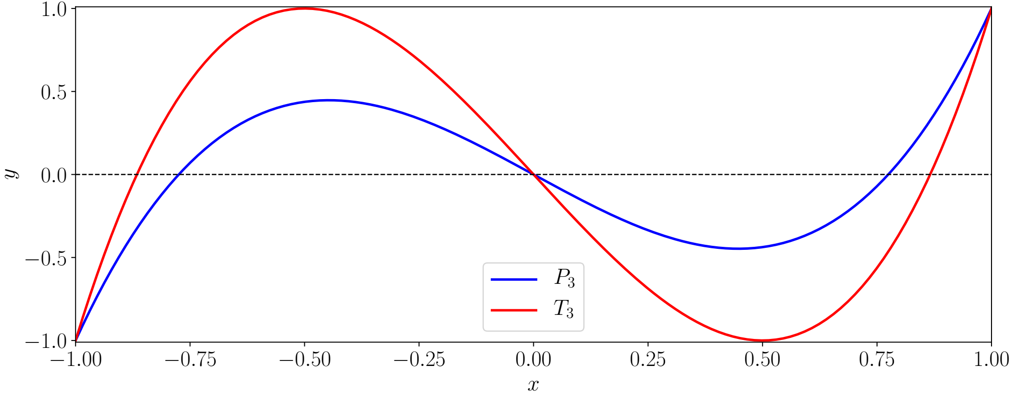
$$\int_{-1}^{1} \frac{dx}{\sqrt{1-x^2}}\, T_m(x) T_n(x) = \frac{\pi}{2}\,\delta_{mn}(1+\delta_{n0})$$



Orthogonal Polynomials

Some reminders

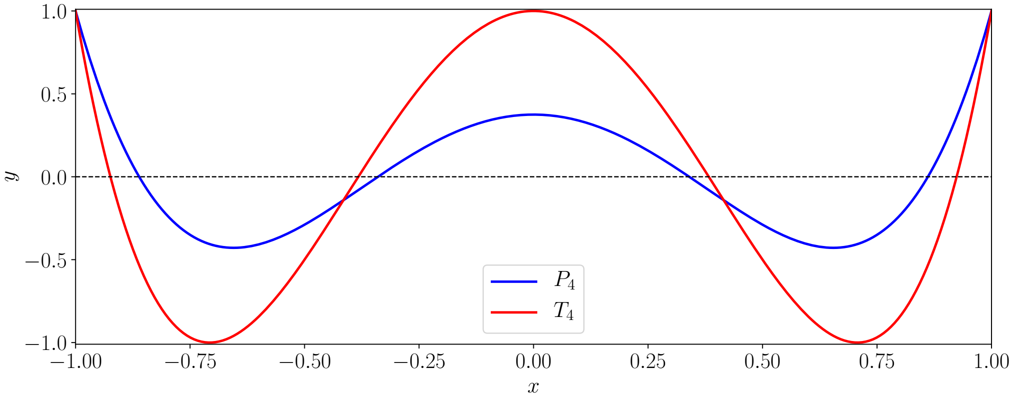
$$\int_{-1}^{1} dx\, P_m(x) P_n(x) = \frac{\delta_{mn}}{n+1/2} \qquad \int_{-1}^{1} \frac{dx}{\sqrt{1-x^2}}\, T_m(x) T_n(x) = \frac{\pi}{2}\, \delta_{mn}(1+\delta_{n0})$$



Orthogonal Polynomials

Some reminders

$$\int_{-1}^{1} dx\, P_m(x) P_n(x) = \frac{\delta_{mn}}{n+1/2} \qquad \int_{-1}^{1} \frac{dx}{\sqrt{1-x^2}}\, T_m(x) T_n(x) = \frac{\pi}{2}\,\delta_{mn}(1+\delta_{n0})$$

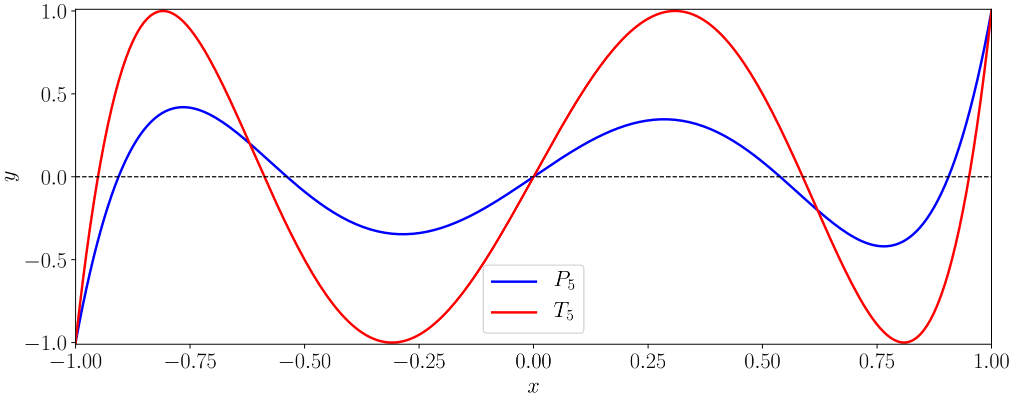


Orthogonal Polynomials

Some reminders

$$\int_{-1}^{1} dx\, P_m(x) P_n(x) = \frac{\delta_{mn}}{n+1/2}$$

$$\int_{-1}^{1} \frac{dx}{\sqrt{1-x^2}}\, T_m(x) T_n(x) = \frac{\pi}{2}\,\delta_{mn}(1+\delta_{n0})$$

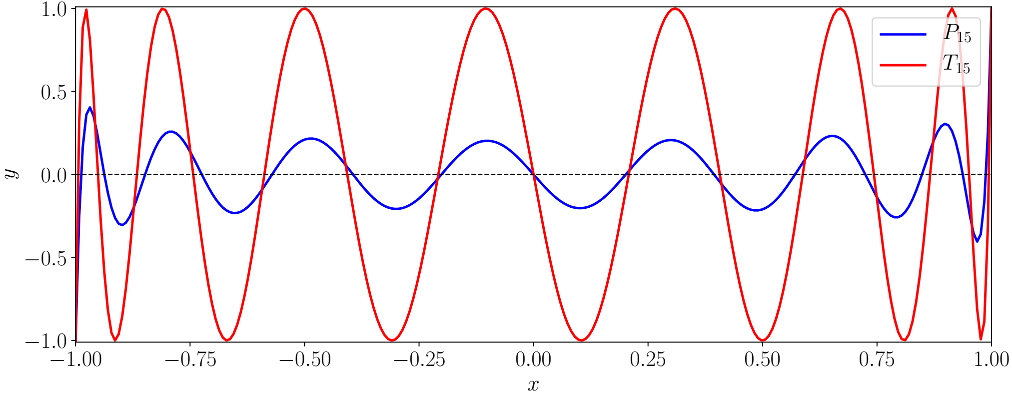


Orthogonal Polynomials

Some reminders

$$\int_{-1}^{1} dx\, P_m(x) P_n(x) = \frac{\delta_{mn}}{n+1/2}$$

$$\int_{-1}^{1} \frac{dx}{\sqrt{1-x^2}}\, T_m(x) T_n(x) = \frac{\pi}{2}\, \delta_{mn}(1+\delta_{n0})$$



Connection of integration to interpolation

Observation:

Gauss quadrature rules place nodes at zeros of orthogonal polynomials

Connection of integration to interpolation

Quardature nodes/roots in python

```
>>> import numpy.polynomial as p
>>> xa,wa = p.legendre.leggauss(4)
>>> xb,wb = p.chebyshev.chebgauss(4)
>>> print xa,xb
[-0.86113631 -0.33998104  0.33998104  0.86113631]
[ 0.92387953  0.38268343 -0.38268343 -0.92387953]
>>> print wa,wb
[0.34785485 0.65214515 0.65214515 0.34785485]
[0.78539816 0.78539816 0.78539816 0.78539816]
```

Chebyshev note: $x_i = \cos[\pi(2i-1)/(2n)] \quad w_i = \pi/n$

Polynomial Interpolation

Place nodes at zeros of P_N

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

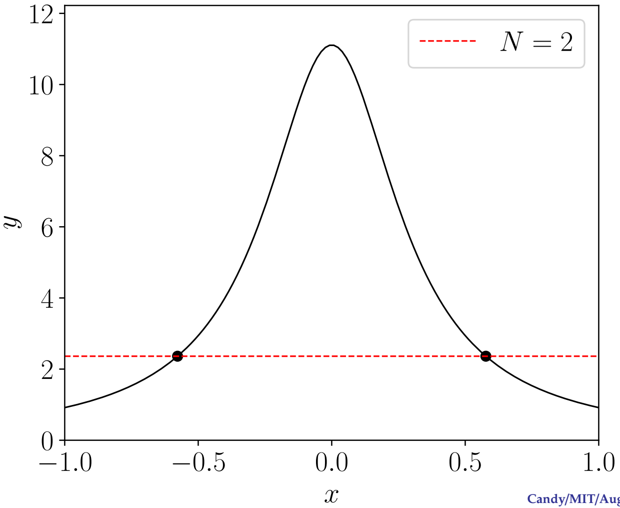
$\{x_i\}$ are the zeros of $P_N(x)$

Polynomial Interpolation

Place nodes at zeros of P_N

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ are the zeros of $P_N(x)$

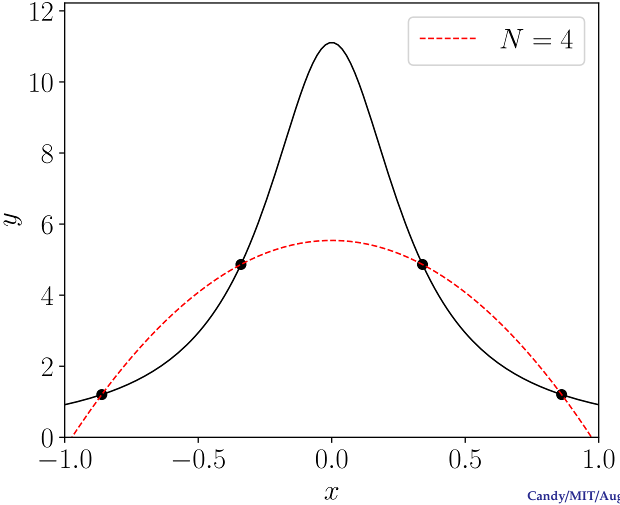


Polynomial Interpolation

Place nodes at zeros of P_N

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ **are the zeros of** $P_N(x)$

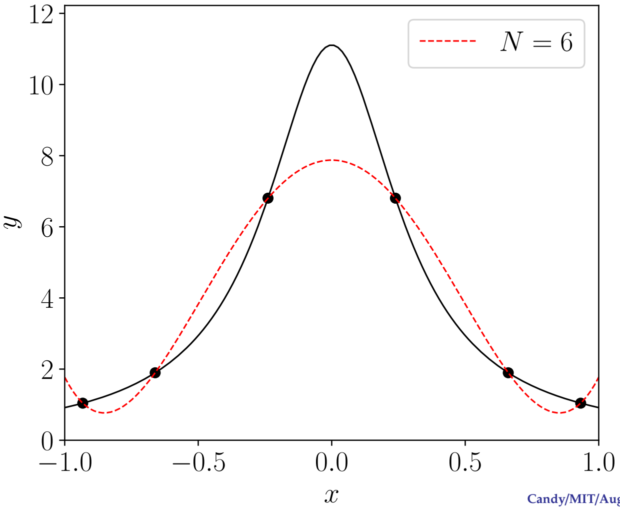


Polynomial Interpolation

Place nodes at zeros of P_N

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ are the zeros of $P_N(x)$

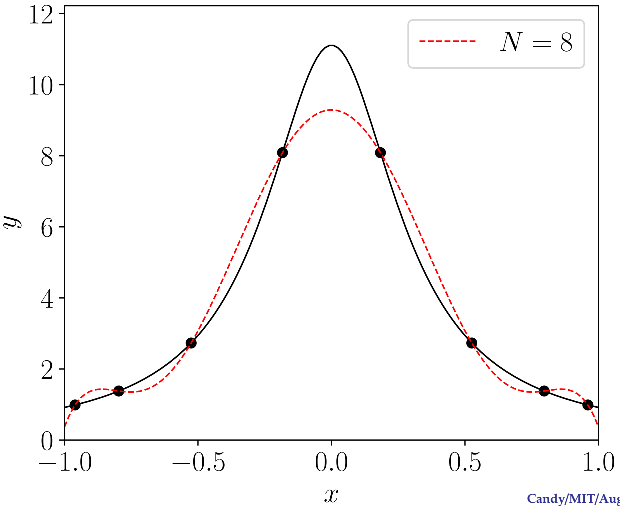


Polynomial Interpolation

Place nodes at zeros of P_N

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ are the zeros of $P_N(x)$

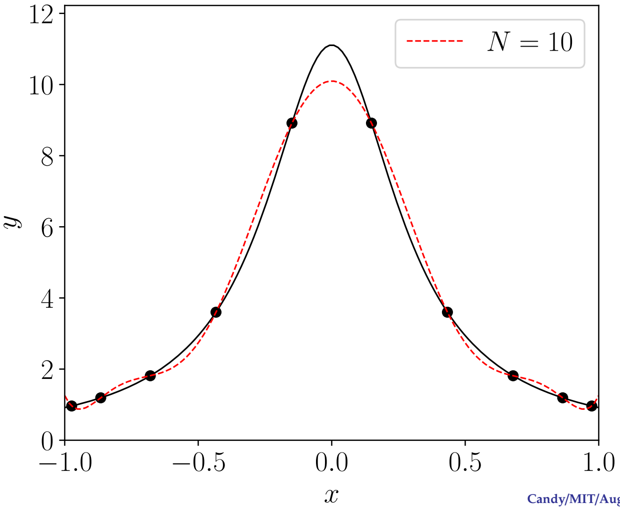


Polynomial Interpolation

Place nodes at zeros of P_N

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ **are the zeros of** $P_N(x)$

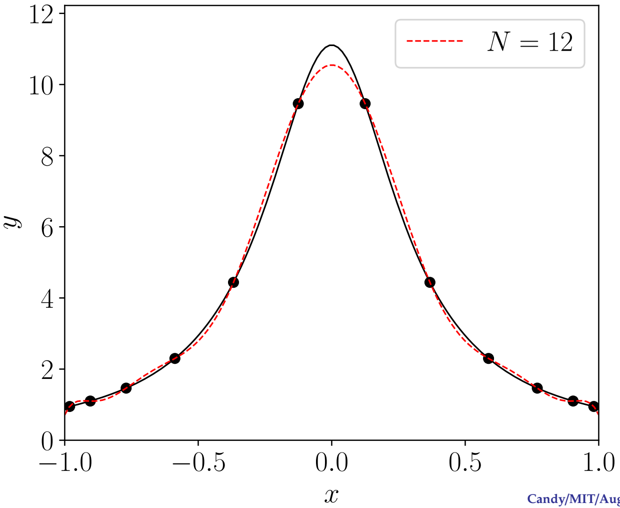


Polynomial Interpolation

Place nodes at zeros of P_N

$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ are the zeros of $P_N(x)$

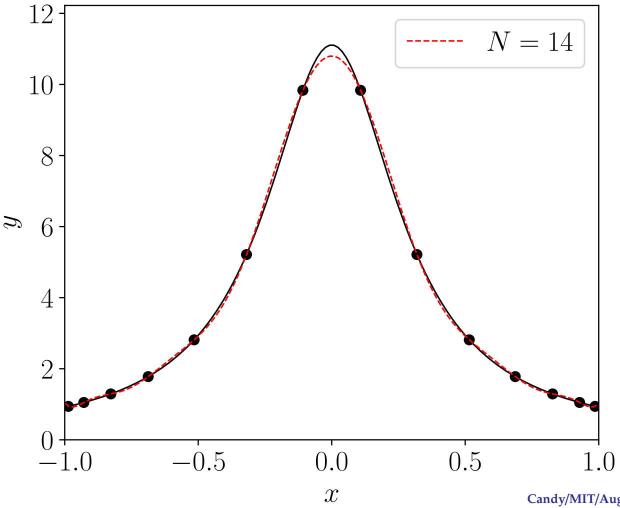


Polynomial Interpolation

Place nodes at zeros of P_N

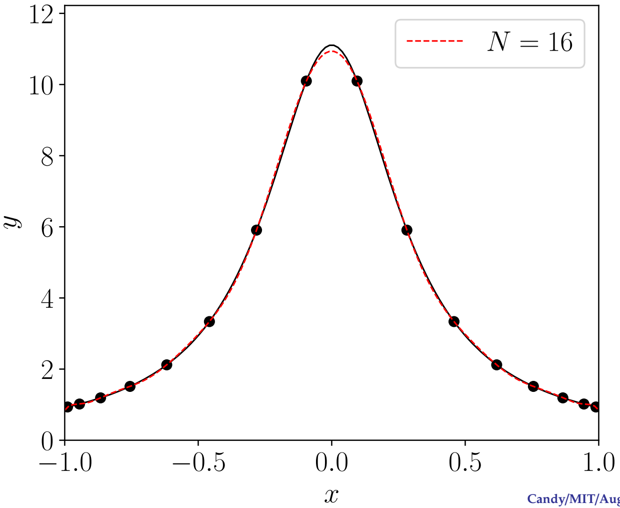
$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ are the zeros of $P_N(x)$



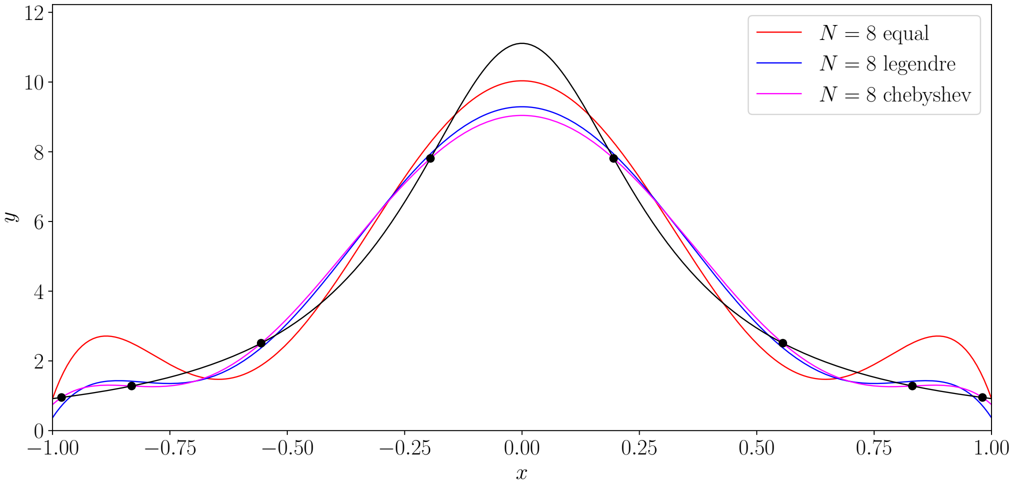
$$\frac{1}{(0.3)^2 + x_i^2} = \sum_{n=0}^{N-1} a_n x_i^n$$

$\{x_i\}$ are the zeros of $P_N(x)$



Polynomial Interpolation

Comparison of fit quality



A pseudospectral method for pitch-angle scattering

Example from kinetic theory

Legendre Pseudospectral method for pitch-angle scattering

$$\frac{\partial f}{\partial t} = \nu \frac{\partial}{\partial x}(1 - x^2)\frac{\partial f}{\partial x}$$

- Time-evolution due to **pitch-angle scattering**

Example from kinetic theory

Legendre Pseudospectral method for pitch-angle scattering

$$\frac{\partial f}{\partial t} = \nu \frac{\partial}{\partial x}(1 - x^2)\frac{\partial f}{\partial x}$$

- Time-evolution due to **pitch-angle scattering**
- Start with $f(t)$

Example from kinetic theory

Legendre Pseudospectral method for pitch-angle scattering

$$\frac{\partial f}{\partial t} = \nu \frac{\partial}{\partial x}(1 - x^2)\frac{\partial f}{\partial x}$$

- Time-evolution due to **pitch-angle scattering**
- Start with $f(t)$
- Evolve to $f(t + \Delta t)$

Example from kinetic theory

Legendre Pseudospectral method for pitch-angle scattering

$$\frac{\partial f}{\partial t} = \nu \frac{\partial}{\partial x}(1 - x^2)\frac{\partial f}{\partial x}$$

- Time-evolution due to **pitch-angle scattering**
- Start with $f(t)$
- Evolve to $f(t + \Delta t)$
- Should conserve number

$$\frac{\partial}{\partial t}\int_{-1}^{1} dx\, f(x) = 0 \quad (1)$$

Example from kinetic theory

Legendre Pseudospectral method for pitch-angle scattering

$$\frac{\partial f}{\partial t} = \nu \frac{\partial}{\partial x}(1 - x^2)\frac{\partial f}{\partial x}$$

- Time-evolution due to **pitch-angle scattering**
- Start with $f(t)$
- Evolve to $f(t + \Delta t)$
- Should conserve number

$$\frac{\partial}{\partial t}\int_{-1}^{1} dx\, f(x) = 0 \tag{1}$$

- No conditions at $x = \pm 1$ except regularity

Example from kinetic theory

Legendre Pseudospectral method for pitch-angle scattering

$$\frac{\partial f}{\partial t} = \nu \frac{\partial}{\partial x}(1 - x^2)\frac{\partial f}{\partial x}$$

- We can move between node and spectral bases

$$f(x_i) = \sum_n c_n P_n(x_i) \longrightarrow f_i = A_{in} c_n \quad \text{where} \quad A_{in} = P_n(x_i)$$

Example from kinetic theory

Legendre Pseudospectral method for pitch-angle scattering

$$\frac{\partial f}{\partial t} = \nu \frac{\partial}{\partial x}(1 - x^2)\frac{\partial f}{\partial x}$$

- We can move between node and spectral bases

$$f(x_i) = \sum_n c_n P_n(x_i) \longrightarrow f_i = A_{in} c_n \quad \text{where} \quad A_{in} = P_n(x_i)$$

- The evolution equation at node *i* becomes

$$\frac{\partial f_i}{\partial t} = -\nu \sum_n n(n+1) c_n P_n(x_i) = -\nu L_{in} c_n$$

$$\text{where} \quad L_{in} = n(n+1) P_n(x_i)$$

Legendre Pseudospectral method for pitch-angle scattering

Timestep is a matrix-vector multiply

$$\frac{\partial f}{\partial t} = \nu \frac{\partial}{\partial x}(1 - x^2)\frac{\partial f}{\partial x}$$

- The evolution equation at node i is

$$\frac{\partial f_i}{\partial t} = -\nu C_{ik} f_k \qquad C_{ik} = L_{in}(A^{-1})_{nk}$$

$$L_{in} = n(n+1)P_n(x_i)$$

$$A_{in} = P_n(x_i)$$

Legendre Pseudospectral method for pitch-angle scattering

Timestep is a matrix-vector multiply

$$\frac{\partial f}{\partial t} = \nu \frac{\partial}{\partial x}(1-x^2)\frac{\partial f}{\partial x}$$

- The evolution equation at node i is

$$\frac{\partial f_i}{\partial t} = -\nu C_{ik} f_k \qquad \begin{aligned} C_{ik} &= L_{in}(A^{-1})_{nk} \\ L_{in} &= n(n+1)P_n(x_i) \\ A_{in} &= P_n(x_i) \end{aligned}$$

- Implicit time advance, with $\bar{f} = f(t+\Delta t)$

$$\frac{\bar{f}_i - f_i}{\Delta t} = -\nu C_{ik}\bar{f}_k \quad \longrightarrow \quad (\delta_{ik} + \nu\Delta t C_{ik})\bar{f}_k = f_i$$

Legendre Pseudospectral method for pitch-angle scattering

Timestep is a matrix-vector multiply

$$\frac{\partial f}{\partial t} = \nu \frac{\partial}{\partial x}(1 - x^2)\frac{\partial f}{\partial x}$$

- Spectrally-accurate discrete approximation

$$\bar{f}_k = M_{ik} f_i$$

Legendre Pseudospectral method for pitch-angle scattering

Timestep is a matrix-vector multiply

$$\frac{\partial f}{\partial t} = \nu \frac{\partial}{\partial x}(1-x^2)\frac{\partial f}{\partial x}$$

- Spectrally-accurate discrete approximation

$$\bar{f}_k = M_{ik} f_i$$

- Conserves number to machine precision

Legendre Pseudospectral method for pitch-angle scattering

Timestep is a matrix-vector multiply

$$\frac{\partial f}{\partial t} = \nu \frac{\partial}{\partial x}(1 - x^2)\frac{\partial f}{\partial x}$$

- Spectrally-accurate discrete approximation

$$\bar{f}_k = M_{ik} f_i$$

- Conserves number to machine precision
- Regularity at $x = \pm 1$ without accuracy loss or instability

Legendre Pseudospectral method for pitch-angle scattering

Timestep is a matrix-vector multiply

$$\frac{\partial f}{\partial t} = \nu \frac{\partial}{\partial x}(1 - x^2)\frac{\partial f}{\partial x}$$

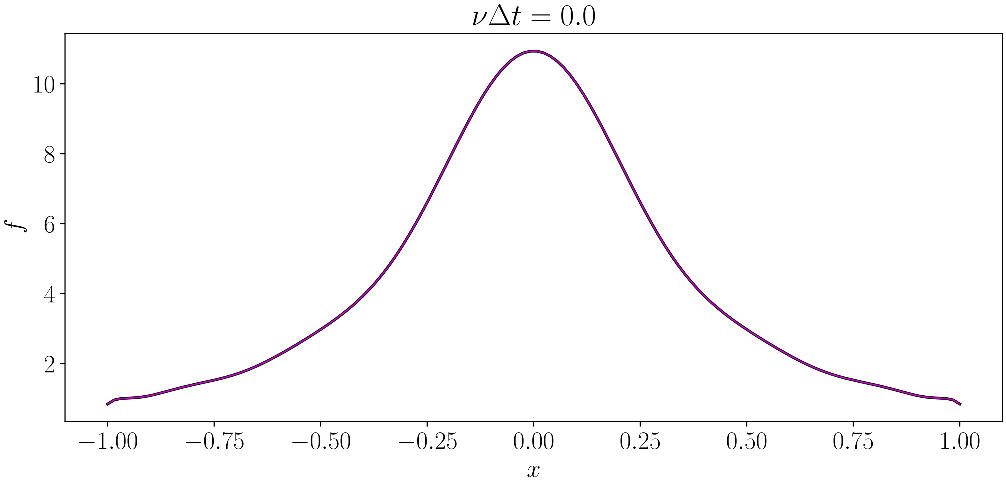
- Spectrally-accurate discrete approximation

$$\bar{f}_k = M_{ik} f_i$$

- Conserves number to machine precision
- Regularity at $x = \pm 1$ without accuracy loss or instability
- Matrix-vector multiply (DGEMV) well-suited for multicore or accelerator optimization

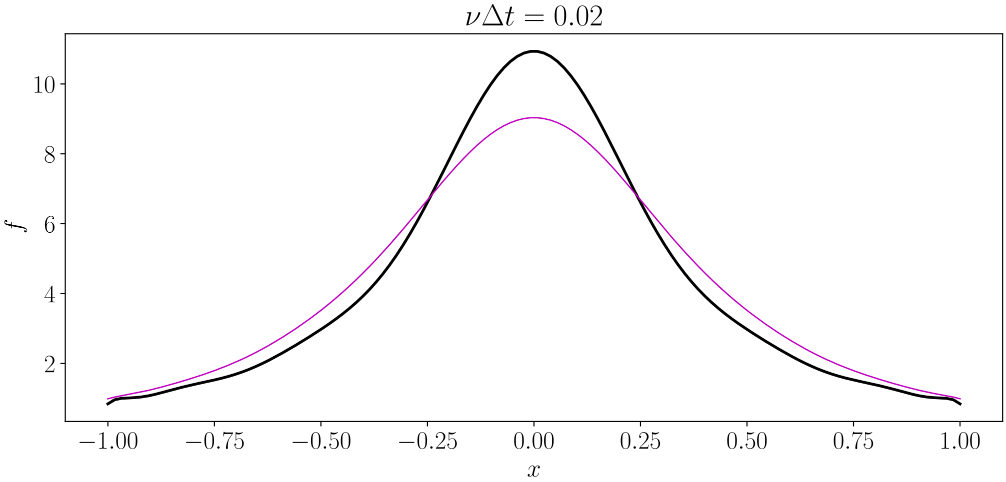
Legendre Pseudospectral method for pitch-angle scattering

Density exactly conserved ; smooth solutions a boundaries



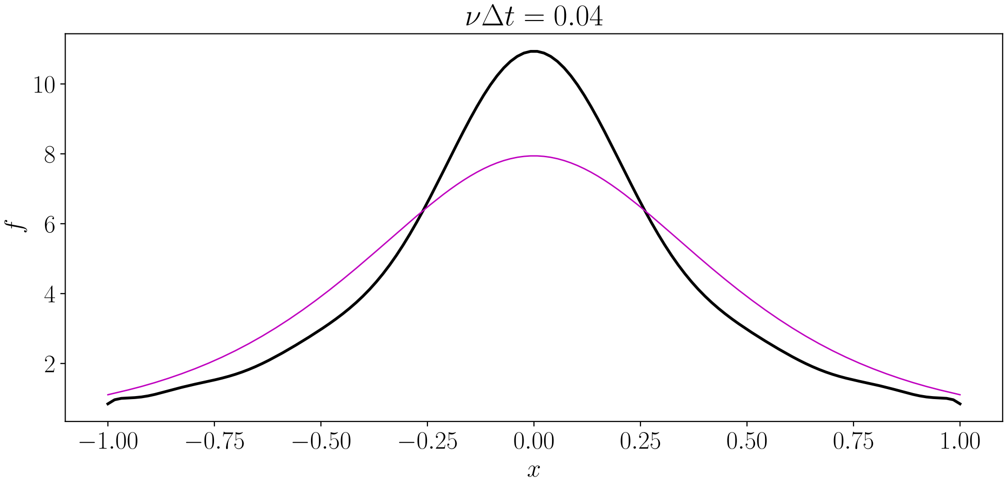
Legendre Pseudospectral method for pitch-angle scattering

Density exactly conserved ; smooth solutions a boundaries



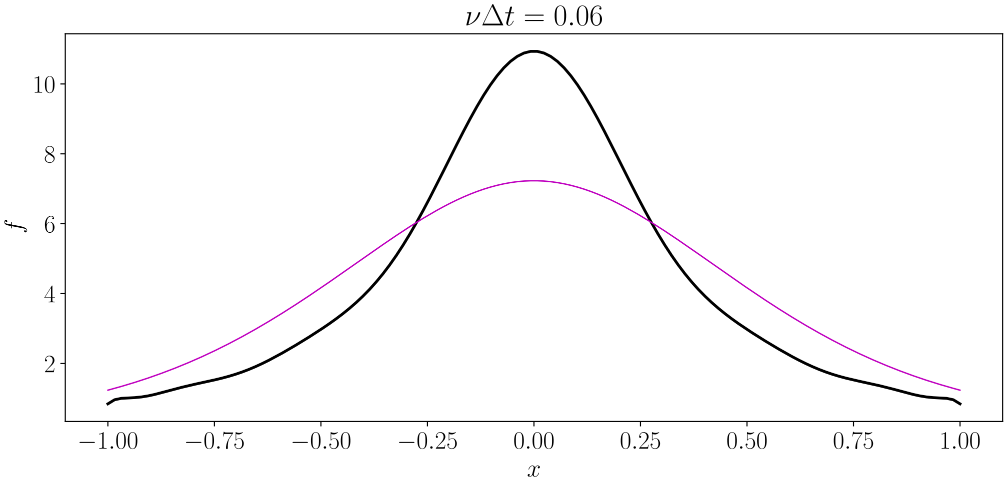
Legendre Pseudospectral method for pitch-angle scattering

Density exactly conserved ; smooth solutions a boundaries



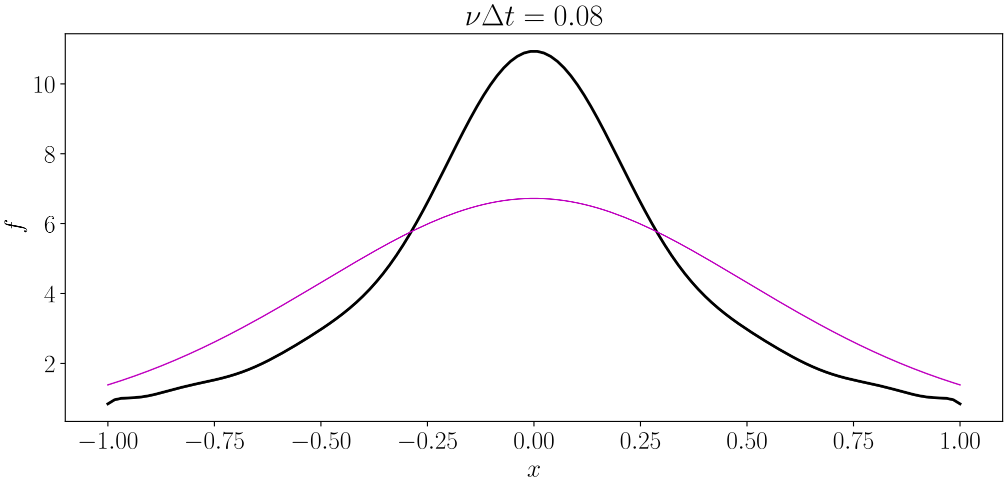
Legendre Pseudospectral method for pitch-angle scattering

Density exactly conserved ; smooth solutions a boundaries



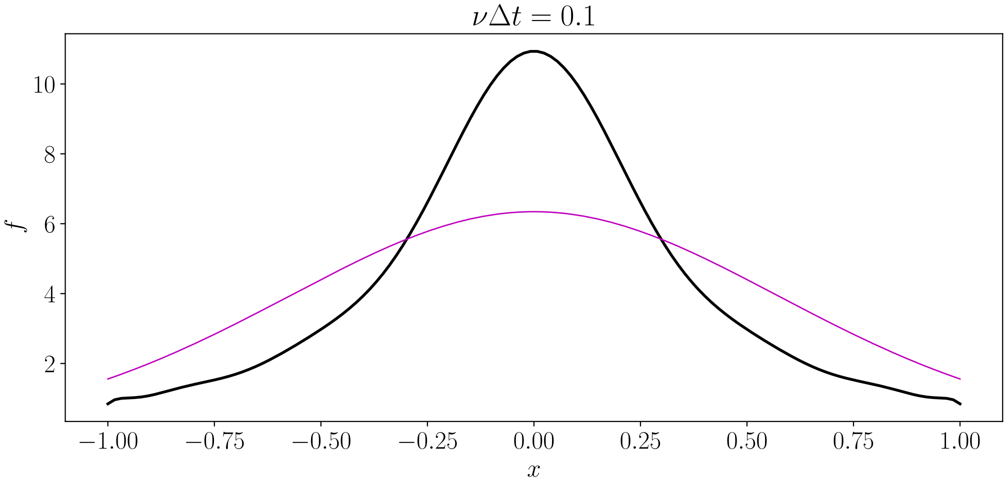
Legendre Pseudospectral method for pitch-angle scattering

Density exactly conserved ; smooth solutions a boundaries



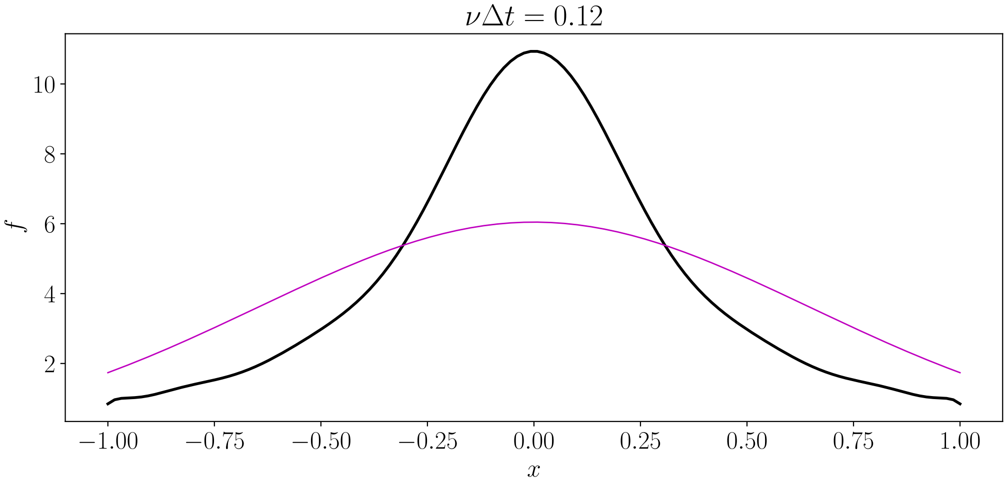
Legendre Pseudospectral method for pitch-angle scattering

Density exactly conserved ; smooth solutions a boundaries



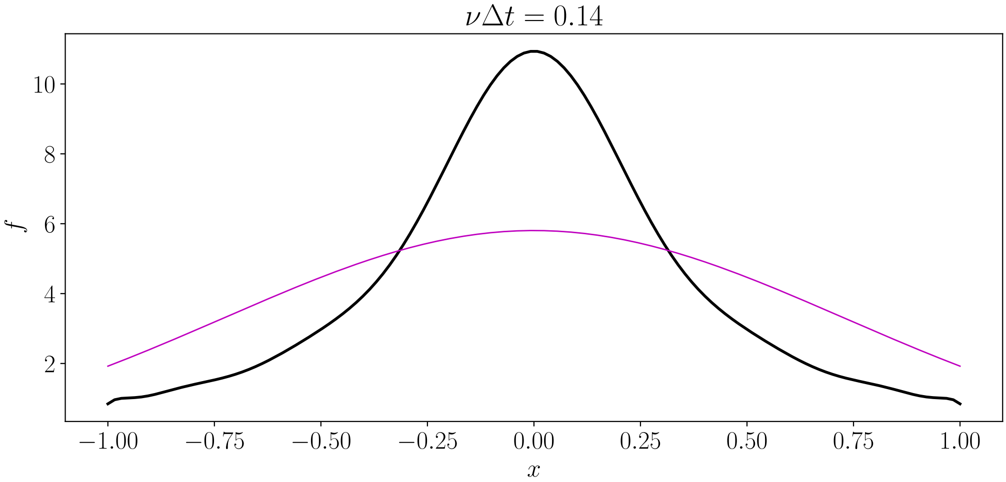
Legendre Pseudospectral method for pitch-angle scattering

Density exactly conserved ; smooth solutions a boundaries



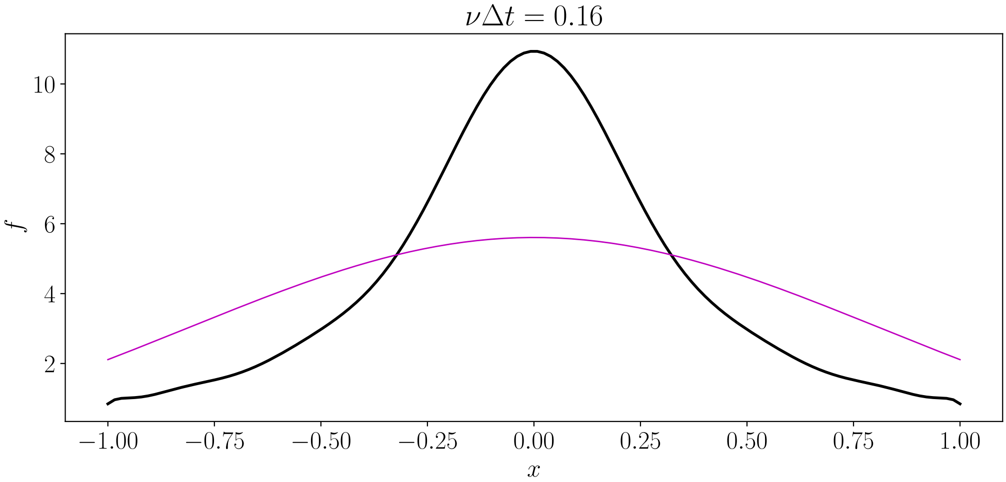
Legendre Pseudospectral method for pitch-angle scattering

Density exactly conserved ; smooth solutions a boundaries



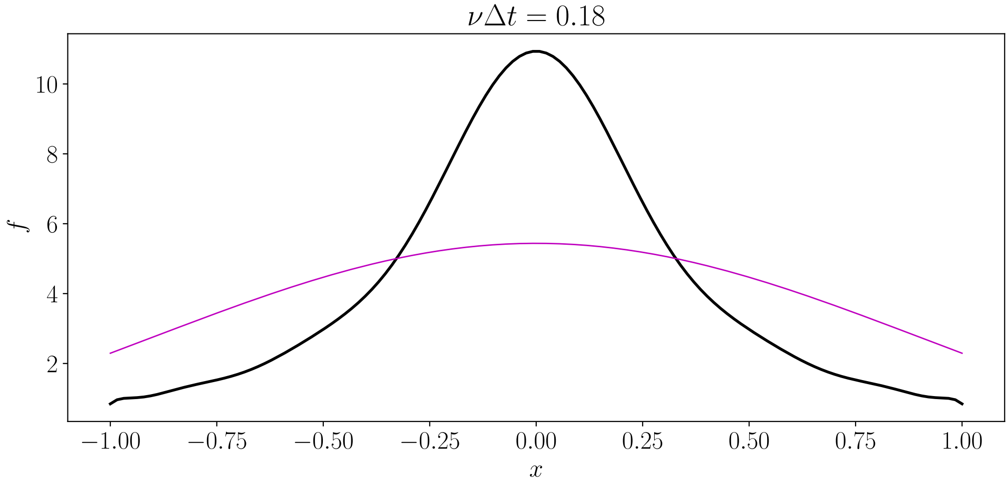
Legendre Pseudospectral method for pitch-angle scattering

Density exactly conserved ; smooth solutions a boundaries



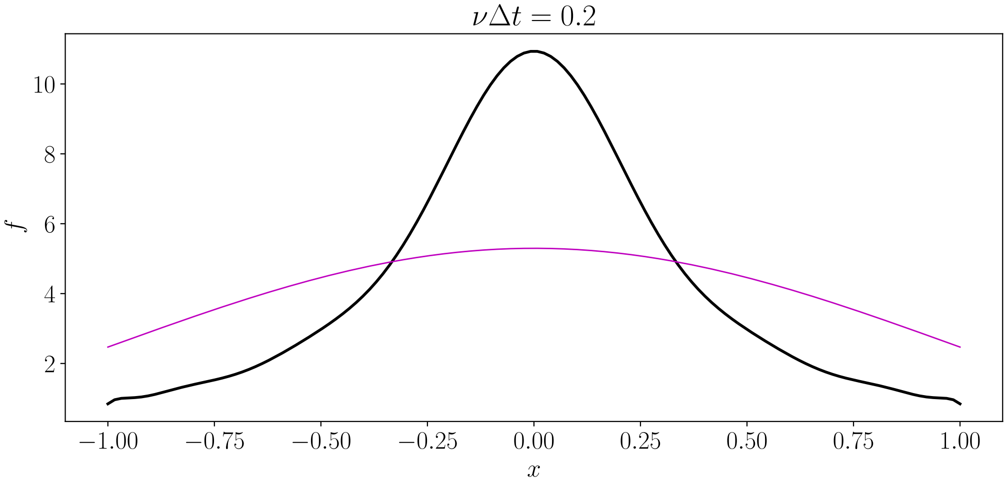
Legendre Pseudospectral method for pitch-angle scattering

Density exactly conserved ; smooth solutions a boundaries



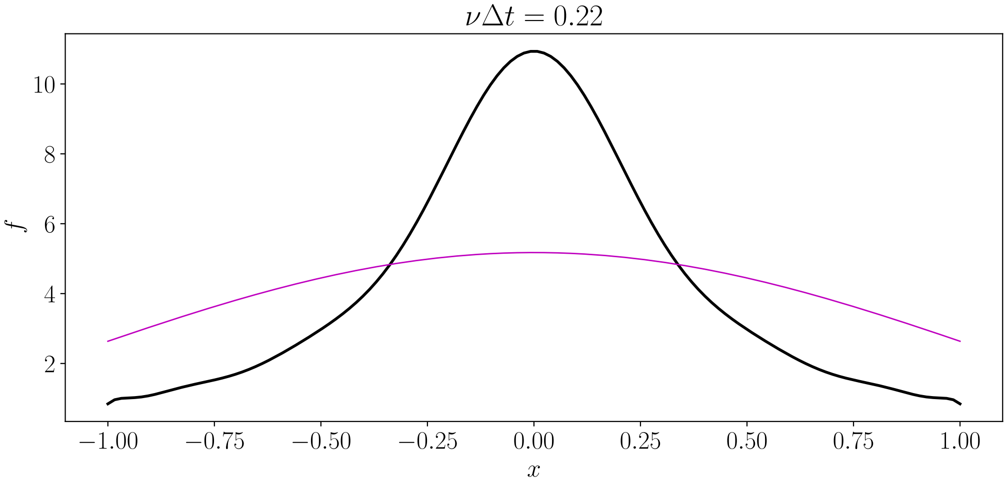
Legendre Pseudospectral method for pitch-angle scattering

Density exactly conserved ; smooth solutions a boundaries



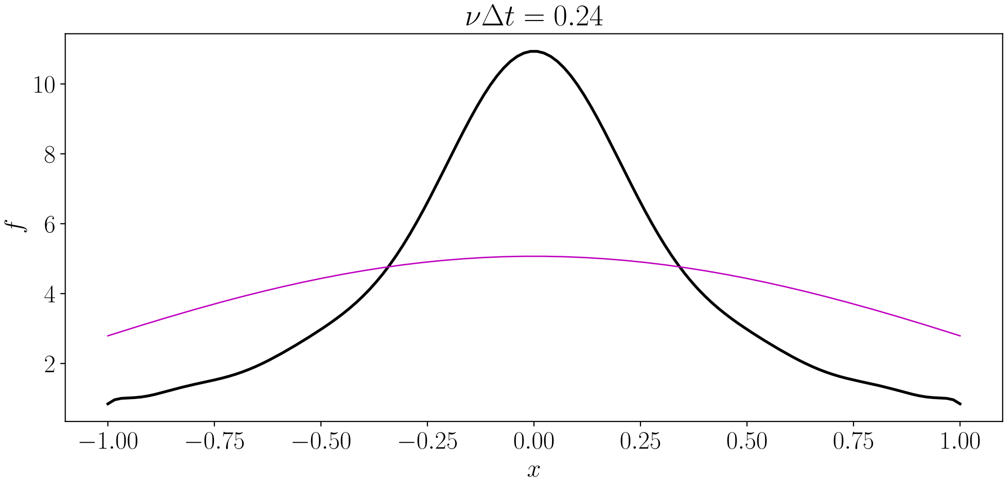
Legendre Pseudospectral method for pitch-angle scattering

Density exactly conserved ; smooth solutions a boundaries



Legendre Pseudospectral method for pitch-angle scattering

Density exactly conserved ; smooth solutions a boundaries



Waves and how to advect them

One-dimensional advection

Advection is a fundamental process in plasma fluid and kinetic theory

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} = 0 \quad \longrightarrow \quad f(x - vt)$$

One-dimensional advection

Advection is a fundamental process in plasma fluid and kinetic theory

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} = 0 \quad \longrightarrow \quad f(x - vt)$$

1. A successful numerical approach is **not intuitive**

One-dimensional advection

Advection is a fundamental process in plasma fluid and kinetic theory

$$\frac{\partial f}{\partial t} + v \frac{\partial f}{\partial x} = 0 \quad \longrightarrow \quad f(x - vt)$$

1. A successful numerical approach is **not intuitive**
2. Replacing derivatives by centered-differences is not enough

One-dimensional advection

Advection is a fundamental process in plasma fluid and kinetic theory

$$\boxed{\frac{\partial f}{\partial t} + v \frac{\partial f}{\partial x} = 0 \quad \longrightarrow \quad f(x - vt)}$$

1. A successful numerical approach is **not intuitive**
2. Replacing derivatives by centered-differences is not enough
3. Problems associated with **poorly resolved waves**

Canonical wrong way to solve the equation

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_{p+1} - f_{p-1}}{2\Delta x}\right) = 0$$

Canonical wrong way to solve the equation

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_{p+1} - f_{p-1}}{2\Delta x}\right) = 0$$

- Ansatz for numerical solution (Von Neumann stability analysis):

$$f_p(t) = e^{i(kp\Delta x - \omega t)}$$

Canonical wrong way to solve the equation

$$\frac{\partial f_p}{\partial t} + v \left(\frac{f_{p+1} - f_{p-1}}{2\Delta x} \right) = 0$$

- Ansatz for numerical solution (Von Neumann stability analysis):

$$f_p(t) = e^{i(kp\Delta x - \omega t)}$$

- Instead of $\omega = ck$, we get

$$\omega = c \, \frac{\sin k\Delta x}{\Delta x}$$

Canonical wrong way to solve the equation

$$\frac{\partial f_p}{\partial t} + v \left(\frac{f_{p+1} - f_{p-1}}{2\Delta x} \right) = 0$$

- Ansatz for numerical solution (Von Neumann stability analysis):

$$f_p(t) = e^{i(kp\Delta x - \omega t)}$$

- Instead of $\omega = ck$, we get

$$\omega = c\,\frac{\sin k\Delta x}{\Delta x}$$

- Obviously there is a problem for $k\Delta x \sim 1$!

Canonical wrong way to solve the equation

$$\frac{\partial f_p}{\partial t} + v \left(\frac{f_{p+1} - f_{p-1}}{2\Delta x} \right) = 0$$

- Most poorly resolved wave (grid-to-grid) has $\lambda = 2\Delta x$ or $k\Delta x = \pi$

Canonical wrong way to solve the equation

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_{p+1} - f_{p-1}}{2\Delta x}\right) = 0$$

- Most poorly resolved wave (grid-to-grid) has $\lambda = 2\Delta x$ or $k\Delta x = \pi$
- **Phase speed**

$$v_{\mathrm{phase}} = \frac{\omega}{k} = c\,\frac{\sin k\Delta x}{k\Delta x} \longrightarrow 0$$

Canonical wrong way to solve the equation

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_{p+1} - f_{p-1}}{2\Delta x}\right) = 0$$

- Most poorly resolved wave (grid-to-grid) has $\lambda = 2\Delta x$ or $k\Delta x = \pi$
- **Phase speed**

$$v_{\text{phase}} = \frac{\omega}{k} = c\,\frac{\sin k\Delta x}{k\Delta x} \longrightarrow 0$$

- **Group velocity**

$$v_{\text{group}} = \frac{\partial \omega}{\partial k} = c\,\cos k\Delta x \longrightarrow -c$$

Solution is to damp poorly-resolved waves

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_{p+1} - f_{p-1}}{2\Delta x}\right) = 0$$

Solution is to damp poorly-resolved waves

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_{p+1} - f_{p-1}}{2\Delta x}\right) - v\,\frac{\Delta x}{2}\left(\frac{f_{p+1} - 2f_p + f_{p-1}}{(\Delta x)^2}\right) = 0$$

- How to deal with poorly-resolved waves?

Solution is to damp poorly-resolved waves

$$\frac{\partial f_p}{\partial t} + v \left(\frac{f_{p+1} - f_{p-1}}{2\Delta x} \right) - v \, \frac{\Delta x}{2} \left(\frac{f_{p+1} - 2f_p + f_{p-1}}{(\Delta x)^2} \right) = 0$$

- How to deal with poorly-resolved waves?
- Add a second-derivative to smooth the solution (kill poorly-resolved waves)

Solution is to damp poorly-resolved waves

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_{p+1} - f_{p-1}}{2\Delta x}\right) - v\,\frac{\Delta x}{2}\left(\frac{f_{p+1} - 2f_p + f_{p-1}}{(\Delta x)^2}\right) = 0$$

- How to deal with poorly-resolved waves?
- Add a second-derivative to smooth the solution (kill poorly-resolved waves)
- Result looks like 1-sided differencing (so-called first-order upwind scheme)

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_p - f_{p-1}}{\Delta x}\right) = 0$$

Solution is to damp poorly-resolved waves

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_{p+1} - f_{p-1}}{2\Delta x}\right) - v\,\frac{\Delta x}{2}\left(\frac{f_{p+1} - 2f_p + f_{p-1}}{(\Delta x)^2}\right) = 0$$

- How to deal with poorly-resolved waves?
- Add a second-derivative to smooth the solution (kill poorly-resolved waves)
- Result looks like 1-sided differencing (so-called first-order upwind scheme)

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_p - f_{p-1}}{\Delta x}\right) = 0$$

- This is the **1st-order upwind** method

Solution is to damp poorly-resolved waves

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_{p+1} - f_{p-1}}{2\Delta x}\right) - v\,\frac{\Delta x}{2}\left(\frac{f_{p+1} - 2f_p + f_{p-1}}{(\Delta x)^2}\right) = 0$$

Solution is to damp poorly-resolved waves

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_{p+1} - f_{p-1}}{2\Delta x}\right) - v\,\frac{\Delta x}{2}\left(\frac{f_{p+1} - 2f_p + f_{p-1}}{(\Delta x)^2}\right) = 0$$

- Poorly-resolved waves now strongly damped

Solution is to damp poorly-resolved waves

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_{p+1} - f_{p-1}}{2\Delta x}\right) - v\,\frac{\Delta x}{2}\left(\frac{f_{p+1} - 2f_p + f_{p-1}}{(\Delta x)^2}\right) = 0$$

- Poorly-resolved waves now strongly damped
- Total dissipation decreases as Δx is reduced

Solution is to damp poorly-resolved waves

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_{p+1} - f_{p-1}}{2\Delta x}\right) - v\,\frac{\Delta x}{2}\left(\frac{f_{p+1} - 2f_p + f_{p-1}}{(\Delta x)^2}\right) = 0$$

- Poorly-resolved waves now strongly damped
- Total dissipation decreases as Δx is reduced
- Dissipation vanishes as $(\Delta x)^3$ for **3rd-order upwind**

Solution is to damp poorly-resolved waves

$$\frac{\partial f_p}{\partial t} + v\left(\frac{f_{p+1} - f_{p-1}}{2\Delta x}\right) - v\,\frac{\Delta x}{2}\left(\frac{f_{p+1} - 2f_p + f_{p-1}}{(\Delta x)^2}\right) = 0$$

- Poorly-resolved waves now strongly damped
- Total dissipation decreases as Δx is reduced
- Dissipation vanishes as $(\Delta x)^3$ for **3rd-order upwind**
- Dissipation vanishes as $(\Delta x)^5$ for **5th-order upwind**

Compact differences

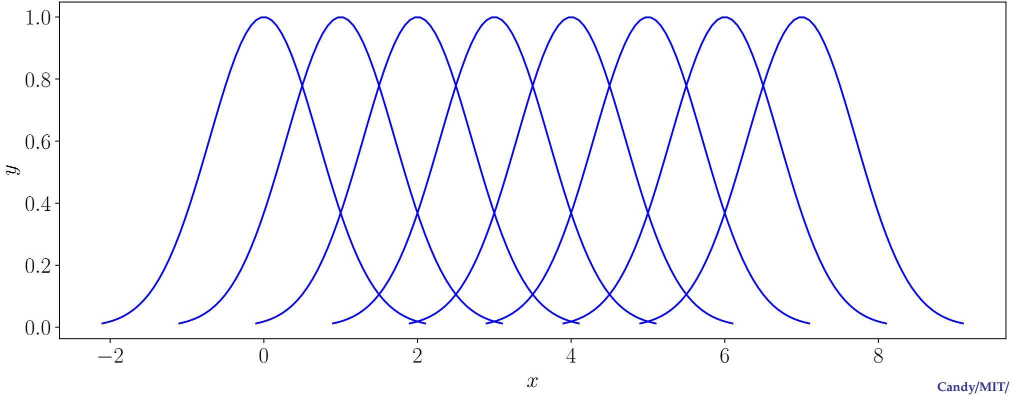
$$\frac{f_{i+1} - f_{i-1}}{2\Delta x} = \frac{1}{6}\left[\left(\frac{df}{dx}\right)_{i+1} + 4\left(\frac{df}{dx}\right)_{i} + \left(\frac{df}{dx}\right)_{i-1}\right] + \mathcal{O}(\Delta x^4)$$

- This is the 4th-order **compact difference scheme**
- Solve a **tridiagonal system** for $(df/dx)_i$
- Compact schemes better than FD schemes at intermediate resolution
- Compact 4th-order better than explicit 6th-order at intermediate resolution

Radial Basis Function Expansion

Not an orthogonal basis

$$f(x) = \sum_{n=0}^{N-1} c_n e^{-(x-x_i)^2/\Delta^2}$$



Here, $r = |\mathbf{x}|$ is length in n dimensions, such that $\mathbf{x} \in \mathcal{R}^n$

$$
\begin{aligned}
\textbf{Gaussian(GRBF)} \quad \phi(r) &= e^{-(\varepsilon r)^2} \\
\text{Multiquadric} \quad \phi(r) &= \sqrt{1 + (\varepsilon r)^2} \\
\text{Inverse Multiquadric} \quad \phi(r) &= \frac{1}{\sqrt{1 + (\varepsilon r)^2}} \\
\text{Inverse Quadratic} \quad \phi(r) &= \frac{1}{1 + (\varepsilon r)^2} \\
\text{Polyharmonic Spline} \quad \phi(r) &= r, r^3, r^5, \ldots \\
\text{Thin-Plate Spline} \quad \phi(r) &= r^2 \ln(r)
\end{aligned}
$$

RBF Interpolation [Fornberg, SIAM J. Sci. Comput. 33 (2011)]

Mesh-free approximation in n-dimensions

1. Pick a radial basis function $\phi(r)$

RBF Interpolation [Fornberg, SIAM J. Sci. Comput. 33 (2011)]

Mesh-free approximation in n-dimensions

1. Pick a radial basis function $\phi(r)$
2. Collect some scattered data $\{\mathbf{x}_k, f_k\}$ for $k = 1, 2, \ldots, N$.

RBF Interpolation [Fornberg, SIAM J. Sci. Comput. 33 (2011)]
Mesh-free approximation in n-dimensions

1. Pick a radial basis function $\phi(r)$
2. Collect some scattered data $\{\mathbf{x}_k, f_k\}$ for $k = 1, 2, \ldots, N$.
3. Make an RBF interpolant

$$s(\mathbf{x}, \varepsilon) = \sum_{k=1}^{N} \lambda_k \phi(|\mathbf{x} - \mathbf{x}_k|)$$

RBF Interpolation [Fornberg, SIAM J. Sci. Comput. 33 (2011)]

Mesh-free approximation in n-dimensions

1. Pick a radial basis function $\phi(r)$
2. Collect some scattered data $\{\mathbf{x}_k, f_k\}$ for $k = 1, 2, \ldots, N$.
3. Make an RBF interpolant
$$s(\mathbf{x}, \varepsilon) = \sum_{k=1}^{N} \lambda_k \phi(|\mathbf{x} - \mathbf{x}_k|)$$
4. Compute the coefficients λ_k as the solution of the linear system
$$A_{jk}\lambda_k = f_j \quad \text{where} \quad A_{jk} = \phi(|\mathbf{x}_j - \mathbf{x}_k|)$$

RBF Interpolation [Fornberg, SIAM J. Sci. Comput. 33 (2011)]

Mesh-free approximation in n-dimensions

1. Pick a radial basis function $\phi(r)$
2. Collect some scattered data $\{\mathbf{x}_k, f_k\}$ for $k = 1, 2, \ldots, N$.
3. Make an RBF interpolant

$$s(\mathbf{x}, \varepsilon) = \sum_{k=1}^{N} \lambda_k \phi(|\mathbf{x} - \mathbf{x}_k|)$$

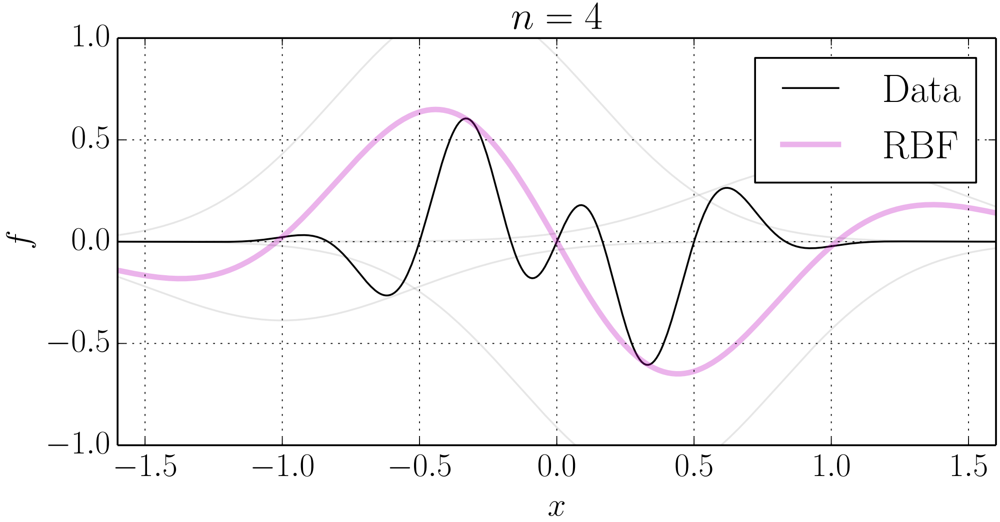
4. Compute the coefficients λ_k as the solution of the linear system

$$A_{jk}\lambda_k = f_j \quad \text{where} \quad A_{jk} = \phi(|\mathbf{x}_j - \mathbf{x}_k|)$$

5. As $\varepsilon \to 0$ accuracy increases but A_{jk} becomes ill-conditioned

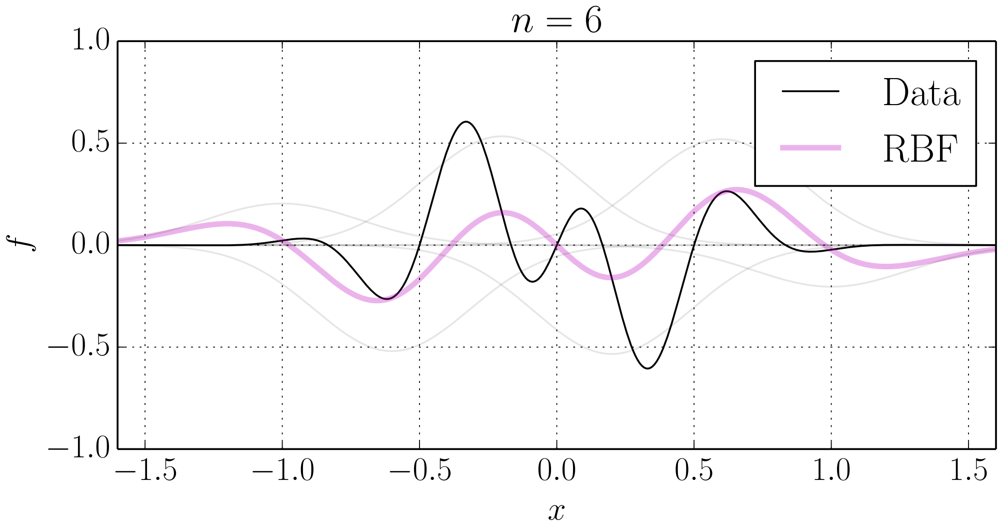
GRBF Illustration

$N = 4$



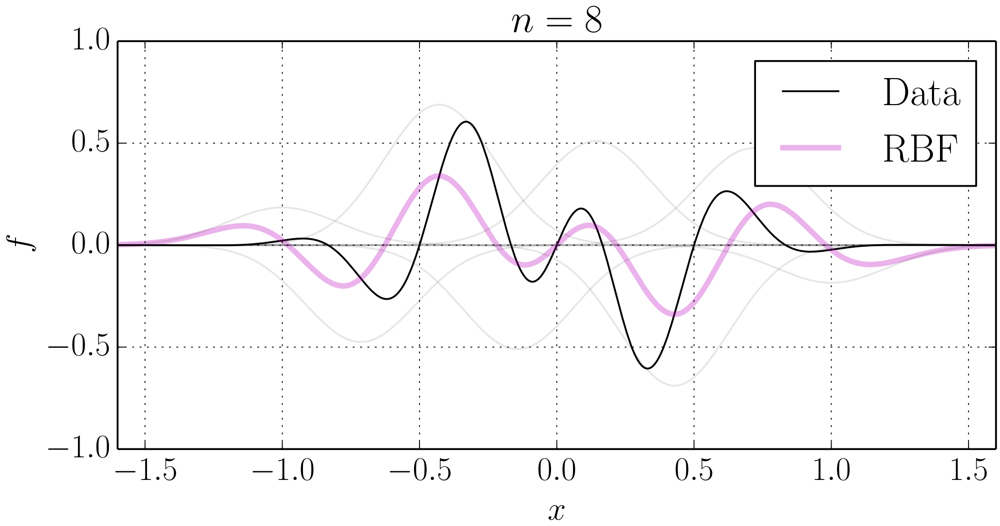
GRBF Illustration

$N = 6$



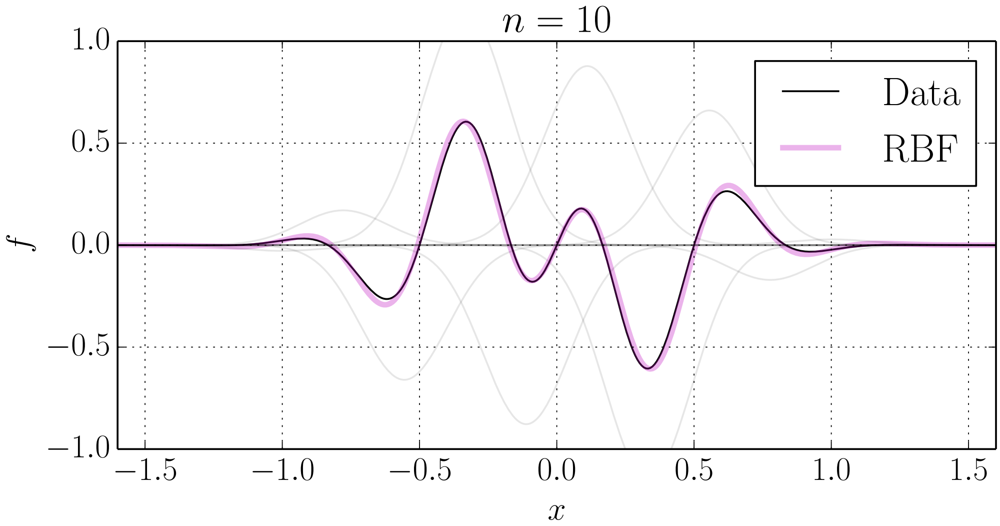
GRBF Illustration

$N = 8$



GRBF Illustration

$N = 10$



6D Fokker-Planck-Landau Equation

Everything: MHD, gyrokinetics, cyclotron waves, you name it.

$$\frac{\partial f_a}{\partial t} + \mathbf{v} \cdot \nabla f_a + \frac{e_a}{m_a} \left(\mathbf{E} + \frac{\mathbf{v}}{c} \times \mathbf{B} \right) \cdot \frac{\partial f_a}{\partial \mathbf{v}} =$$

$f_a(\mathbf{x}, \mathbf{v}, t) \longrightarrow$ 6D distribution function for species a

6D Fokker-Planck-Landau Equation

Everything: MHD, gyrokinetics, cyclotron waves, you name it.

$$\frac{\partial f_a}{\partial t} + \mathbf{v} \cdot \nabla f_a + \frac{e_a}{m_a} \left(\mathbf{E} + \frac{\mathbf{v}}{c} \times \mathbf{B} \right) \cdot \frac{\partial f_a}{\partial \mathbf{v}} = \sum_b C_{ab}(f_a, f_b) + S_a$$

$f_a(\mathbf{x}, \mathbf{v}, t) \longrightarrow$ 6D distribution function for species a

$C_{ab} \longrightarrow$ Nonlinear (bilinear) Collision Operator

6D Fokker-Planck-Landau Equation

Everything: MHD, gyrokinetics, cyclotron waves, you name it.

$$\frac{\partial f_a}{\partial t} + \mathbf{v} \cdot \nabla f_a + \frac{e_a}{m_a} \left(\mathbf{E} + \frac{\mathbf{v}}{c} \times \mathbf{B} \right) \cdot \frac{\partial f_a}{\partial \mathbf{v}} = \sum_b C_{ab}(f_a, f_b) + S_a$$

$f_a(\mathbf{x}, \mathbf{v}, t) \longrightarrow$ 6D distribution function for species a

$C_{ab} \longrightarrow$ Nonlinear (bilinear) Collision Operator

This is a **complex integro-differential** equation in $\mathbf{v}$

Nonlinear Coulomb Collision Operator
Numerical treatment extremely challenging

Collision operator

$$C_{ab}(f_a, f_b) = \frac{\partial}{\partial \mathbf{v}} \cdot \left[\mathbf{A}_{ab} f_a + \frac{\partial}{\partial \mathbf{v}} \cdot (\mathbb{D}_{ab} f_a) \right]$$

Friction and diffusion coefficients

$$\mathbf{A}_{ab} = L^{ab} \left(1 + \frac{m_a}{m_b} \right) \frac{\partial \varphi_b}{\partial \mathbf{v}} \qquad \mathbb{D}_{ab} = -L^{ab} \frac{\partial^2 \psi_b}{\partial \mathbf{v} \partial \mathbf{v}}$$

Rosenbluth potentials

$$\varphi_b(\mathbf{v}, t) = -\frac{1}{4\pi} \int d\mathbf{v}' f_b(\mathbf{v}', t) \frac{1}{|\mathbf{v} - \mathbf{v}'|}$$

$$\psi_b(\mathbf{v}, t) = -\frac{1}{8\pi} \int d\mathbf{v}' f_b(\mathbf{v}', t) |\mathbf{v} - \mathbf{v}'|$$

Connection with plasmas and systems close to equilibrium

- The distribution of particles in the DIII-D core is nearly

$$f_a = \frac{n_a}{(\pi/\gamma_a)^{3/2}} e^{-\gamma_a (\mathbf{v}-\mathbf{V})^2} \quad \text{where} \quad \gamma_a \doteq \frac{m_a}{2T_a}$$

Connection with plasmas and systems close to equilibrium

- The distribution of particles in the DIII-D core is nearly

$$f_a = \frac{n_a}{(\pi/\gamma_a)^{3/2}} e^{-\gamma_a (\mathbf{v}-\mathbf{V})^2} \quad \text{where} \quad \gamma_a \doteq \frac{m_a}{2T_a}$$

- The error is typically less than 1% (gyrokinetic ordering)

Connection with plasmas and systems close to equilibrium

- The distribution of particles in the DIII-D core is nearly

$$f_a = \frac{n_a}{(\pi/\gamma_a)^{3/2}} e^{-\gamma_a (\mathbf{v}-\mathbf{V})^2} \quad \text{where} \quad \gamma_a \doteq \frac{m_a}{2T_a}$$

- The error is typically less than 1% (gyrokinetic ordering)
- This is a **Gaussian RBF** with center $\mathbf{V}$ and width γ_a

Connection with plasmas and systems close to equilibrium

- The distribution of particles in the DIII-D core is nearly

$$f_a = \frac{n_a}{(\pi/\gamma_a)^{3/2}} e^{-\gamma_a(\mathbf{v}-\mathbf{V})^2} \quad \text{where} \quad \gamma_a \doteq \frac{m_a}{2T_a}$$

- The error is typically less than 1% (gyrokinetic ordering)
- This is a **Gaussian RBF** with center $\mathbf{V}$ and width γ_a
- GRBFs are a **natural basis** for plasma simulation

Connection with plasmas and systems close to equilibrium

- The distribution of particles in the DIII-D core is nearly

$$f_a = \frac{n_a}{(\pi/\gamma_a)^{3/2}} e^{-\gamma_a (\mathbf{v}-\mathbf{V})^2} \quad \text{where} \quad \gamma_a \doteq \frac{m_a}{2T_a}$$

- The error is typically less than 1% (gyrokinetic ordering)
- This is a **Gaussian RBF** with center $\mathbf{V}$ and width γ_a
- GRBFs are a **natural basis** for plasma simulation
- **Magic** property: $C_{aa}(f_a, f_a) = 0$

GRBFs = Shifted Maxwellians

Universal functions in statistical mechanics

Basic strategy: Expand f_a in series of shifted Maxwellians

$$f_a(\mathbf{x},\mathbf{v},t) = \sum_i w_i^a(\mathbf{x},t) \left(\frac{\gamma_i}{\pi}\right)^{3/2} e^{-\gamma_i(\mathbf{v}-\mathbf{v}_i)^2}$$

Fluid moments: simple, elegant weighted sums:

$$n_a = \sum_i w_i^a(\mathbf{x},t) \quad \text{and} \quad n_a \mathbf{V}_a = \sum_i w_i^a(\mathbf{x},t)\, \mathbf{v}_i(\mathbf{x})$$

Free parameters: widths (bases) and centers (mesh):

$$\{\gamma_i, \mathbf{v}_i\} \quad \text{... or alternatively } \gamma_i = \frac{m_a}{2\tau_i^a}$$

Can Evaluate Rosenbluth Potentials Analytically

Result is exact in the space of GRBFs

Rosenbluth potentials are evaluated **exactly**:

$$
\begin{aligned}
f_a(\mathbf{v}) &= \sum_i w_i^a \left(\frac{\gamma_i}{\pi}\right)^{3/2} e^{-y^2} \\
-4\pi\,\varphi_a(\mathbf{v}) = \int d\mathbf{v}'\, f_b(\mathbf{v}',t)\frac{1}{|\mathbf{v}-\mathbf{v}'|} &= \sum_i w_i^a\,(\gamma_i)^{1/2}\,\frac{\mathrm{Erf}(y)}{y} \\
-8\pi\,\psi(\mathbf{v}) = \int d\mathbf{v}'\, f_b(\mathbf{v}',t)|\mathbf{v}-\mathbf{v}'| &= \sum_i w_i^a\,(\gamma_i)^{-1/2}\left[\left(y+\frac{1}{2y}\right)\mathrm{Erf}(y)+\frac{1}{\sqrt{\pi}}e^{-y^2}\right]
\end{aligned}
$$

where $y = \sqrt{\gamma_i}|\mathbf{v}-\mathbf{v}_i|$.

Collision operator also has analytic form

Result is exact in the space of GRBFs

The full collision operator is simply a sum of products of weights

$$C_{ab}(f_a, f_b) = \sum_i \sum_k w_i^a \, w_k^b \, C_{ik}^{ab}(\mathbf{v})$$

where

$$C_{kl}^{ab}(\mathbf{v}) = L^{ab} \left[\frac{m_a}{m_b} f_i^a f_k^a + \left(\frac{m_a}{m_b} - 1 \right) \frac{\partial \varphi_i^a}{\partial \mathbf{v}} \cdot \frac{\partial f_k^b}{\partial \mathbf{v}} - \frac{\partial^2 \psi_i^a}{\partial \mathbf{v} \partial \mathbf{v}} : \frac{\partial^2 f_k^b}{\partial \mathbf{v} \partial \mathbf{v}} \right] .$$

Nonlinear GRBF equations

Fluid-like equations with no velocity dependence

Fluid equations for $w_1^a,\, w_2^2,\, \ldots,\, w_N^a$

$$\sum_{i=1}^{N} M_{ij} \left\{ \frac{\partial w_i^a}{\partial t} + \mathbf{v}_j \cdot \nabla w_i^a + \frac{e_a}{\tau_i^a} \left[(\mathbf{v}_i - \mathbf{v}_j) \cdot \mathbf{E} + \frac{1}{c} (\mathbf{v}_i \times \mathbf{v}_j) \cdot \mathbf{B} \right] w_i^a \right\}$$

$$= \sum_{i,k,b} C_{ik,j}^{ab}\, w_i^a\, w_k^b$$

Projection weights: $M_{ij} = \left(\frac{\gamma_j}{\pi} \right)^{3/2} e^{-\gamma_j (\mathbf{v}_i - \mathbf{v}_j)^2} \,, \quad \tau_i^a = \frac{m_a}{2\gamma_i}$

Centers: $\mathbf{v}_1(\mathbf{x}),\, \mathbf{v}_2(\mathbf{x}),\, \ldots,\, \mathbf{v}_N(\mathbf{x})$

Widths: $\gamma_1(\mathbf{x}),\, \gamma_2(\mathbf{x}),\, \ldots,\, \gamma_N(\mathbf{x})$

Fluid Moments

Exact in the space of GRBFs

$$\text{Density} \quad n_a = \sum_i w_i^a$$

$$\text{Velocity} \quad n_a \mathbf{V}_a = \sum_i w_i^a \, \mathbf{v}_i(\mathbf{x})$$

$$\text{Temperature} \quad \frac{3}{2} n_a T_a = \sum_i w_i^a \left[\frac{3}{2} \tau_i^a + \frac{1}{2} m_a (\mathbf{v}_i - \mathbf{V}_a)^2 \right]$$

$$\text{Momentum flux} \quad (\Pi_a)_{\mu\nu} = \sum_i w_i^a \, m_a \left[\tau_i^a + (\mathbf{v}_i)_\mu (\mathbf{v}_i)_\nu \right]$$

$$\text{Energy flux} \quad \mathbf{Q}_a = \sum_i w_i^a \, \mathbf{v}_i \left[\frac{5}{2} \tau_i^a + \frac{1}{2} m_a v_i^2 \right]$$

Nonlinear bi-Maxwellian relaxation

Standard test problem for numerical solvers

1. Single-species plasma:

$$C_{aa}[f_a, f_a] = \gamma_{aa} \left[f_a f_a - \frac{\partial^2 \psi_a}{\partial \mathbf{v} \partial \mathbf{v}} : \frac{\partial^2 f_a}{\partial \mathbf{v} \partial \mathbf{v}} \right]$$

Nonlinear bi-Maxwellian relaxation

Standard test problem for numerical solvers

1. Single-species plasma:

$$C_{aa}[f_a, f_a] = \gamma_{aa} \left[f_a f_a - \frac{\partial^2 \psi_a}{\partial \mathbf{v} \partial \mathbf{v}} : \frac{\partial^2 f_a}{\partial \mathbf{v} \partial \mathbf{v}} \right]$$

2. Normalize time to the collision time

$$\tau = \gamma t \quad \text{where} \quad \gamma \doteq \left(\frac{e_a^2}{m_a \epsilon_0} \right)^2 \ln \Lambda$$

Nonlinear bi-Maxwellian relaxation

Standard test problem for numerical solvers

1. Single-species plasma:

$$C_{aa}[f_a,f_a] = \gamma_{aa} \left[f_a f_a - \frac{\partial^2 \psi_a}{\partial \mathbf{v} \partial \mathbf{v}} : \frac{\partial^2 f_a}{\partial \mathbf{v} \partial \mathbf{v}} \right]$$

2. Normalize time to the collision time

$$\tau = \gamma t \quad \text{where} \quad \gamma \doteq \left(\frac{e_a^2}{m_a \epsilon_0} \right)^2 \ln \Lambda$$

3. Solve the initial value problem

$$\frac{\partial f_a}{\partial \tau} = f_a f_a - \frac{\partial^2 \psi_a}{\partial \mathbf{v} \partial \mathbf{v}} : \frac{\partial^2 f_a}{\partial \mathbf{v} \partial \mathbf{v}}$$

Nonlinear bi-Maxwellian relaxation
Standard test problem for numerical solvers

1. Single-species plasma:

$$C_{aa}[f_a,f_a] = \gamma_{aa}\left[f_a f_a - \frac{\partial^2\psi_a}{\partial\mathbf{v}\partial\mathbf{v}} : \frac{\partial^2 f_a}{\partial\mathbf{v}\partial\mathbf{v}}\right]$$

2. Normalize time to the collision time

$$\tau = \gamma t \quad \text{where} \quad \gamma \doteq \left(\frac{e_a^2}{m_a\epsilon_0}\right)^2 \ln\Lambda$$

3. Solve the initial value problem **by applying the RBF expansion**

$$\frac{\partial f_a}{\partial\tau} = f_a f_a - \frac{\partial^2\psi_a}{\partial\mathbf{v}\partial\mathbf{v}} : \frac{\partial^2 f_a}{\partial\mathbf{v}\partial\mathbf{v}}$$

Nonlinear bi-Maxwellian relaxation

Initial state

1. Choose a **bi-Maxwellian** distribution function

$$f_s(\mathbf{v}, \tau_0) = \sum_{i=1}^{2} \exp[-\beta_i(\mathbf{v} - \mathbf{v}^i)^2]$$

with $\mathbf{v}^{1,2} = (\pm 3, 0, 0)$ and $\beta_{1,2} = 1/5$

Nonlinear bi-Maxwellian relaxation

Initial state

1. Choose a **bi-Maxwellian** distribution function

$$f_s(\mathbf{v}, \tau_0) = \sum_{i=1}^{2} \exp[-\beta_i(\mathbf{v} - \mathbf{v}^i)^2]$$

with $\mathbf{v}^{1,2} = (\pm 3, 0, 0)$ and $\beta_{1,2} = 1/5$

2. Project the initial state to the RBF basis

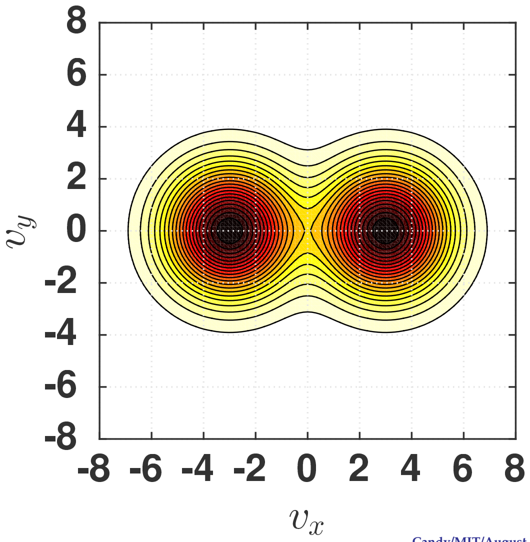
Nonlinear bi-Maxwellian relaxation
Initial state

1. Choose a **bi-Maxwellian** distribution function

$$f_s(\mathbf{v}, \tau_0) = \sum_{i=1}^{2} \exp[-\beta_i(\mathbf{v} - \mathbf{v}^i)^2]$$

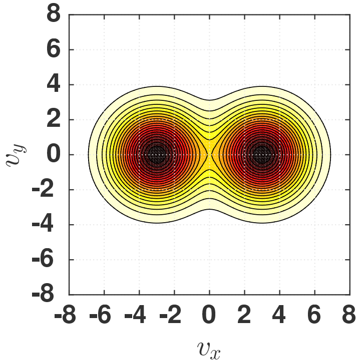
with $\mathbf{v}^{1,2} = (\pm 3, 0, 0)$ and $\beta_{1,2} = 1/5$

2. Project the initial state to the RBF basis



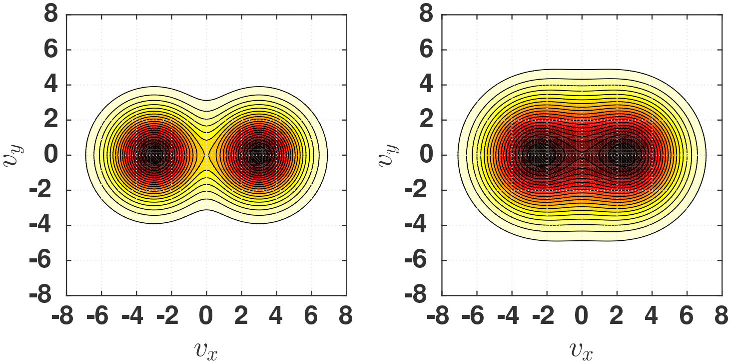
Nonlinear bi-Maxwellian relaxation

Contours illustrating time-evolution



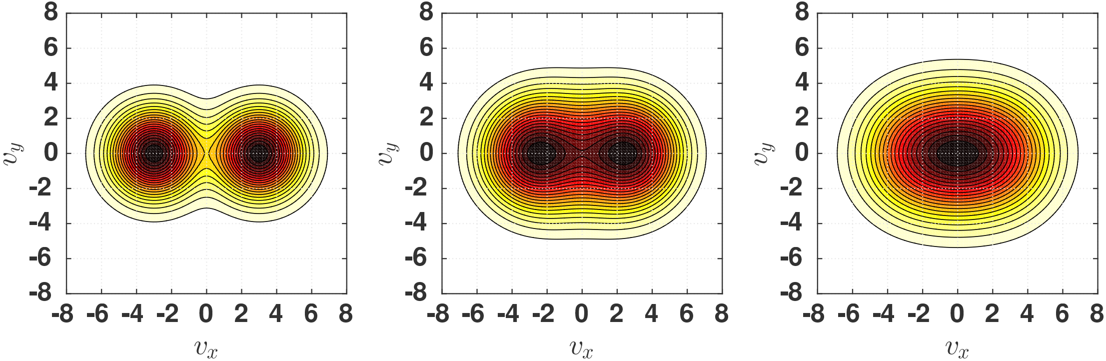
Nonlinear bi-Maxwellian relaxation

Contours illustrating time-evolution



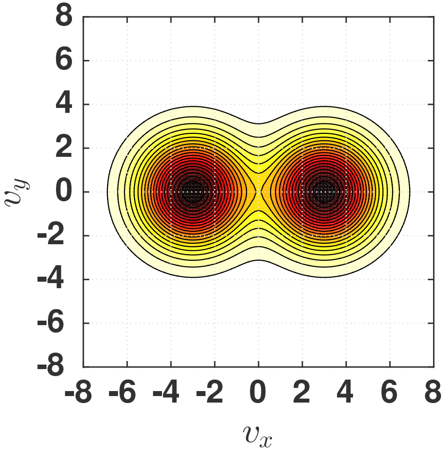
Nonlinear bi-Maxwellian relaxation

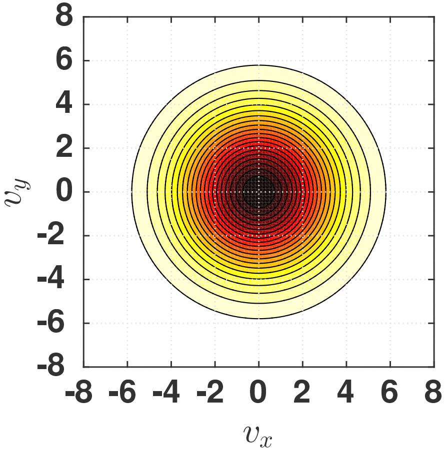
Contours illustrating time-evolution



Nonlinear bi-Maxwellian relaxation

Contours illustrating time-evolution

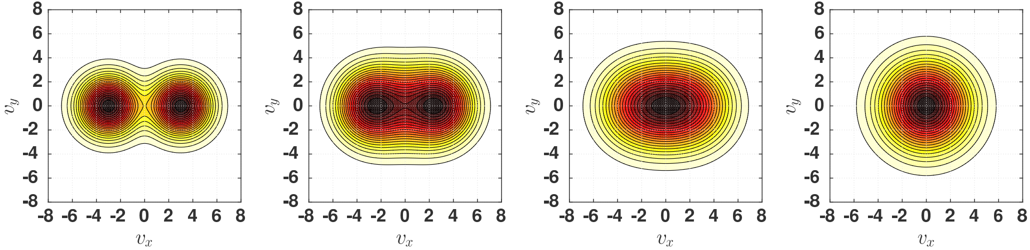




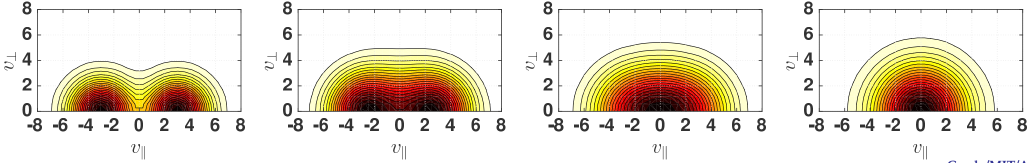
Nonlinear bi-Maxwellian relaxation

3D versus 2D

3D $F_s^i(\mathbf{v}) = (\gamma_s^i/\pi)^{3/2} \exp\left[-\gamma_s^i(\mathbf{v} - \mathbf{v}_s^i)^2\right]$



2D $F_s^i(\mathbf{v}) = (\gamma_s^i/\pi)^{3/2} I_0(2\gamma_s^i v_{s,\perp}^i v_\perp) e^{-\gamma_s^i(v_\parallel - v_{s,\parallel}^i)^2 - \gamma_s^i(v_\perp^2 + (v_{s,\perp}^i)^2)}$



Helmholtz Equation (3D)

Green's function is an RBF

$$\left(\nabla^2 + k^2\right) p(\mathbf{x}) = 0$$

- Green's function for exterior problem is an **RBF**

$$G_j(\mathbf{x}) = \frac{e^{ikR_j(\mathbf{x})}}{4\pi R_j(\mathbf{x})} \;, \quad \text{with} \quad R_j = \left|\mathbf{x} - \mathbf{x}_j\right| \;.$$

$$p(\mathbf{x}) = \sum_{j=1}^{n_s} c_j G_j(\mathbf{x})$$

- $p(\mathbf{x})$ solves **Helmholtz equation** but not (yet) **boundary conditions**

Helmholtz Equation (3D)

Choose coefficients to satisfy BCs

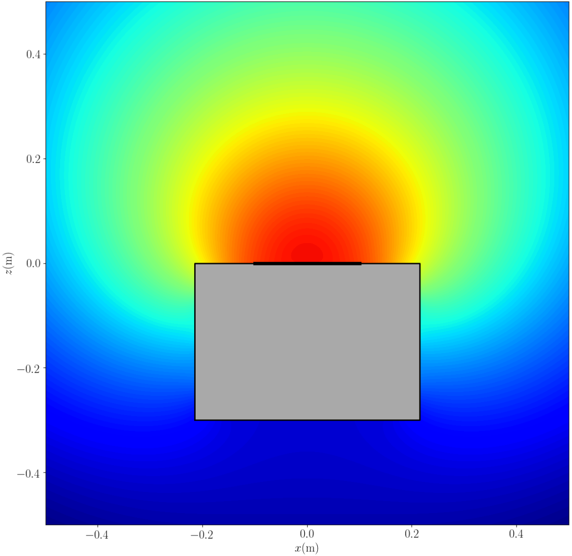
$$\left(\nabla^2 + k^2\right) p(\mathbf{x}) = 0$$

$$p(\mathbf{x}) = \sum_{j=1}^{n_s} c_j G_j(\mathbf{x})$$

- Choose c_j to satisfy boundary conditions ($\partial p/\partial n$ on surface)
- Put sources at locations $\mathbf{x}_j$ inside boundary regions
- Try to keep sources equally-spaced to avoid ill-conditioning

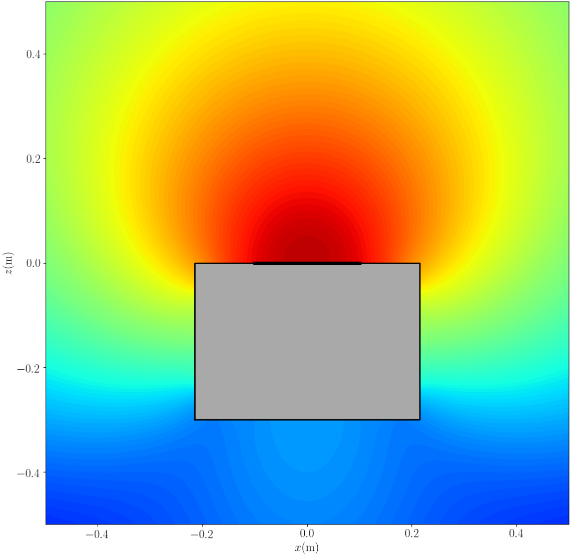
Helmholtz Equation (3D)

Piston radiator in rectangular box



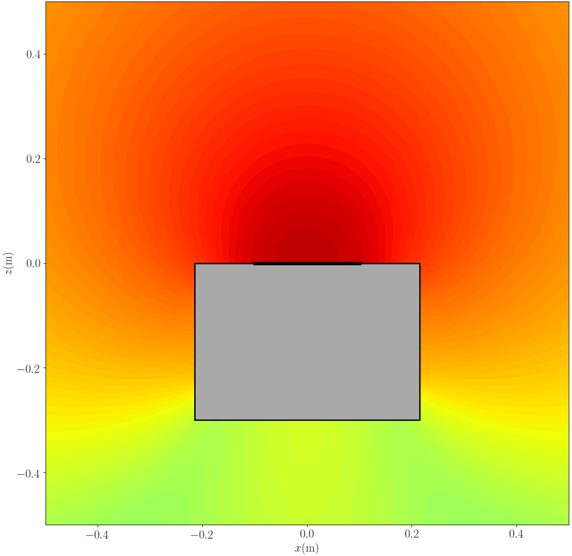
Helmholtz Equation (3D)

Piston radiator in rectangular box



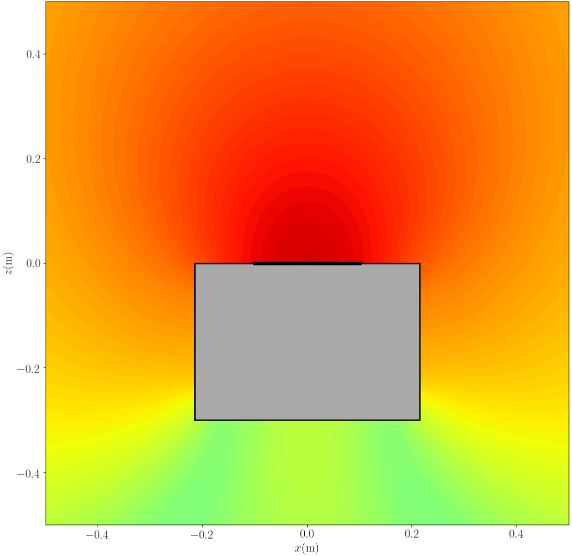
Helmholtz Equation (3D)

Piston radiator in rectangular box



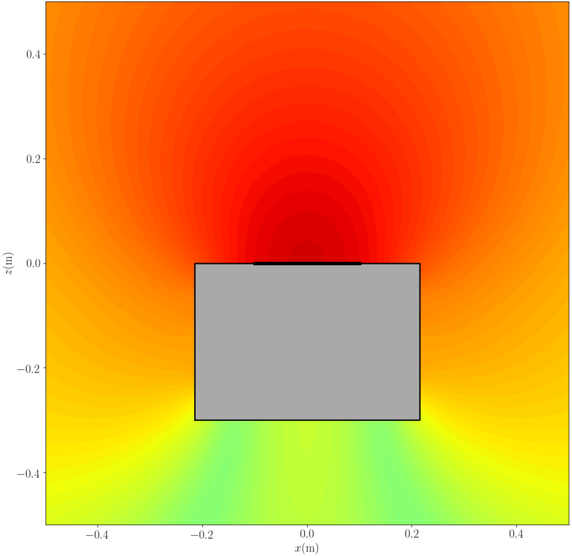
Helmholtz Equation (3D)

Piston radiator in rectangular box



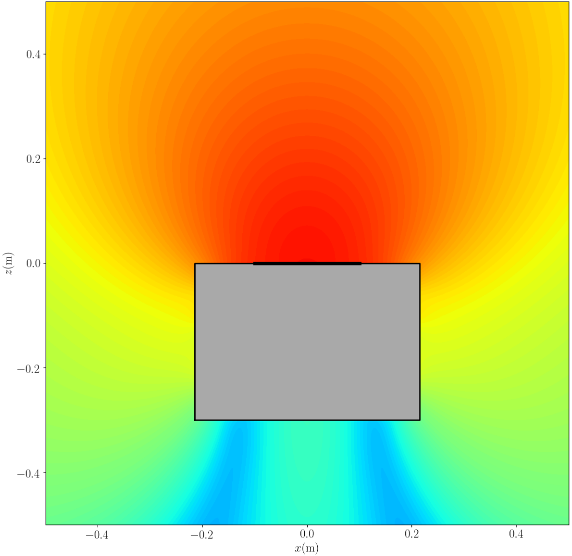
Helmholtz Equation (3D)

Piston radiator in rectangular box



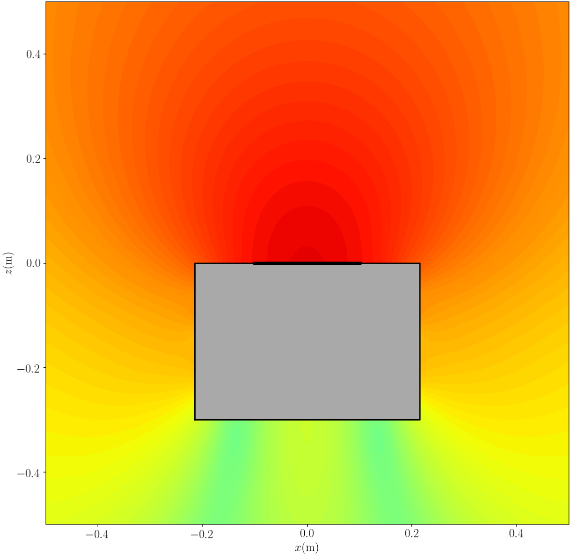
Helmholtz Equation (3D)

Piston radiator in rectangular box



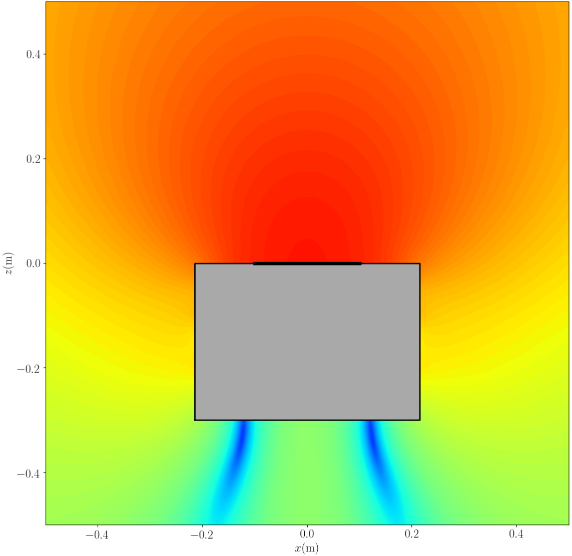
Helmholtz Equation (3D)

Piston radiator in rectangular box



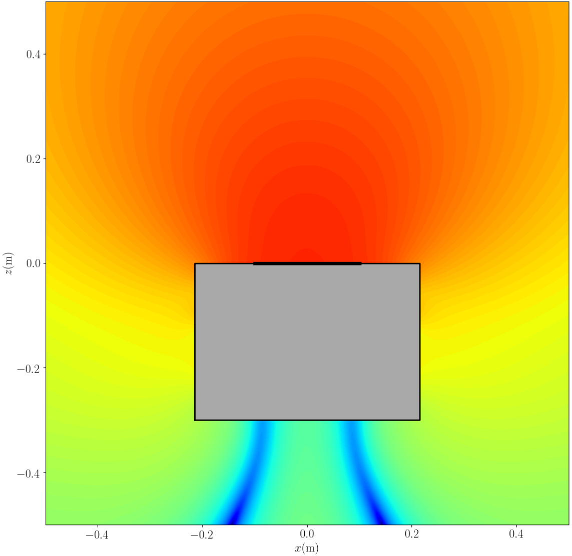
Helmholtz Equation (3D)

Piston radiator in rectangular box



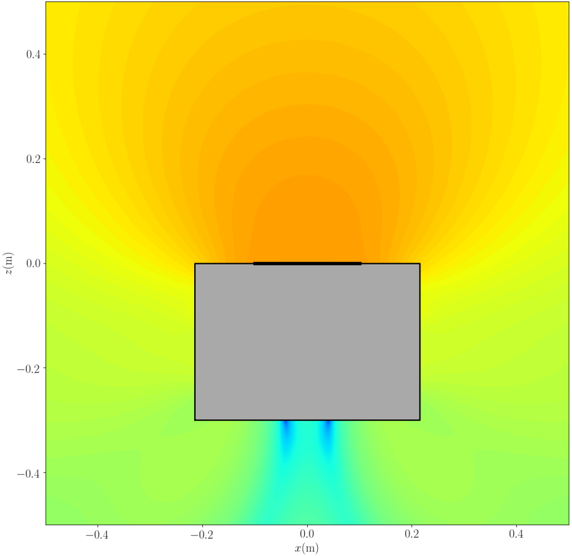
Helmholtz Equation (3D)

Piston radiator in rectangular box



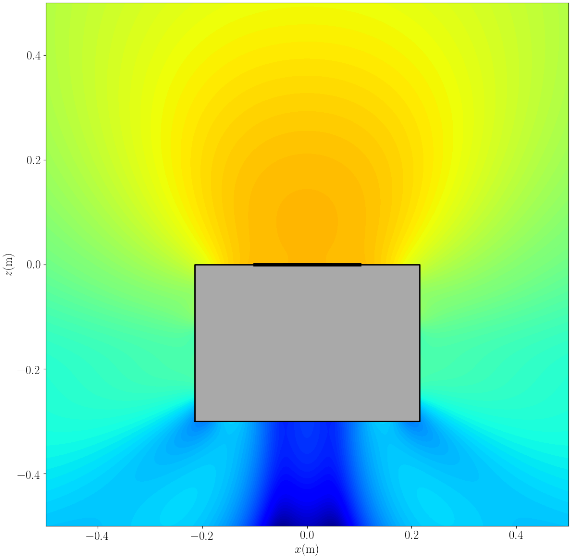
Helmholtz Equation (3D)

Piston radiator in rectangular box



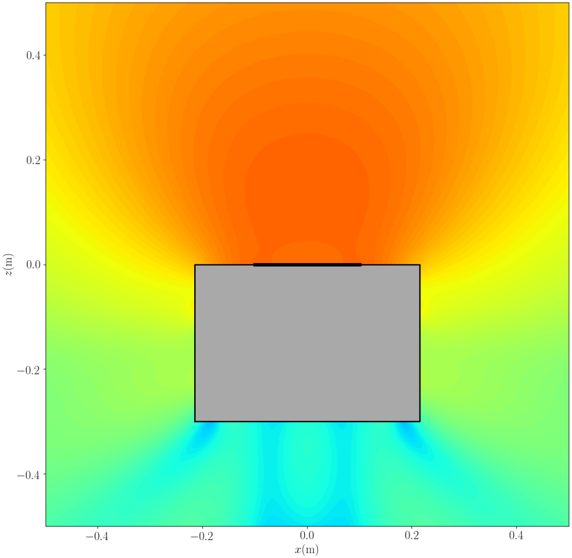
Helmholtz Equation (3D)

Piston radiator in rectangular box



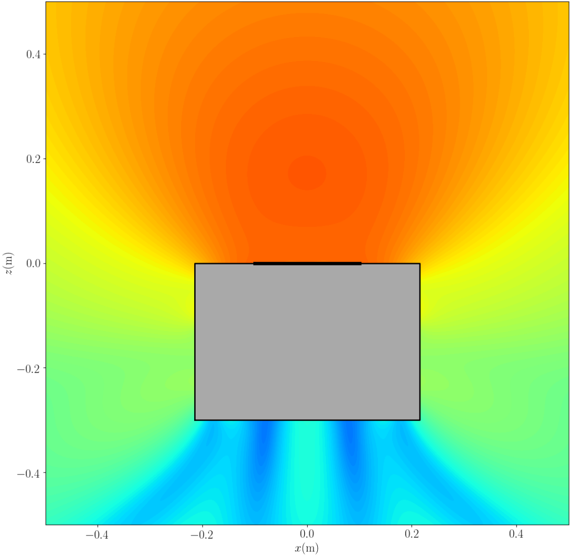
Helmholtz Equation (3D)

Piston radiator in rectangular box



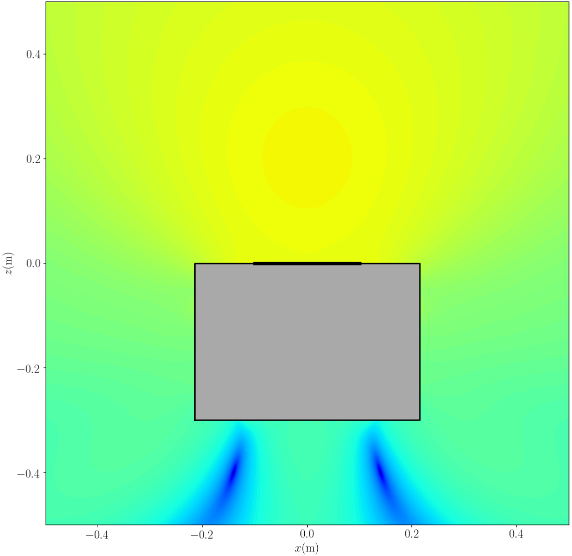
Helmholtz Equation (3D)

Piston radiator in rectangular box



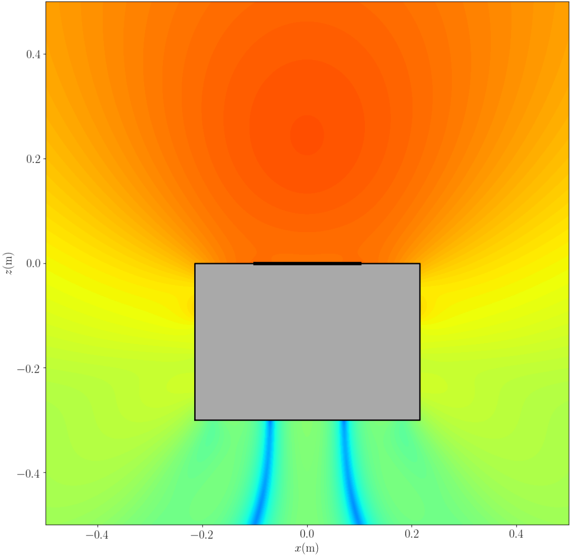
Helmholtz Equation (3D)

Piston radiator in rectangular box



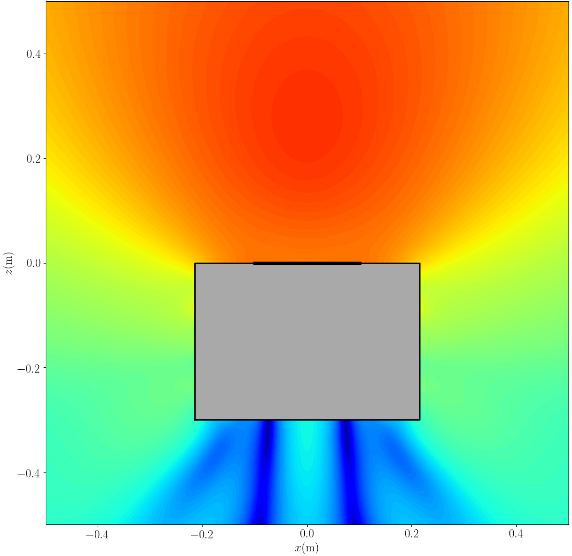
Helmholtz Equation (3D)

Piston radiator in rectangular box



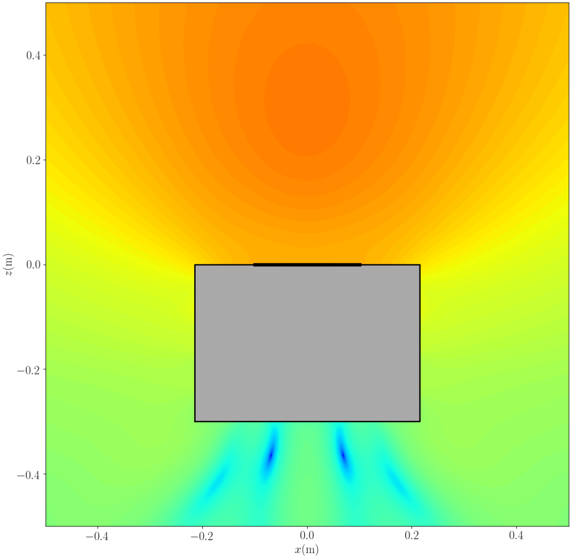
Helmholtz Equation (3D)

Piston radiator in rectangular box



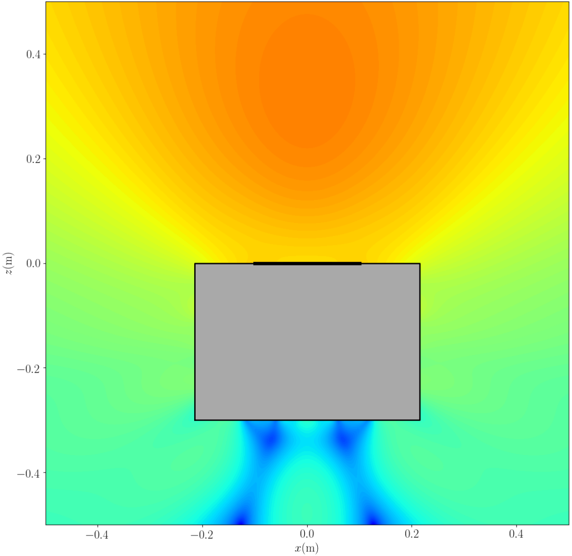
Helmholtz Equation (3D)

Piston radiator in rectangular box



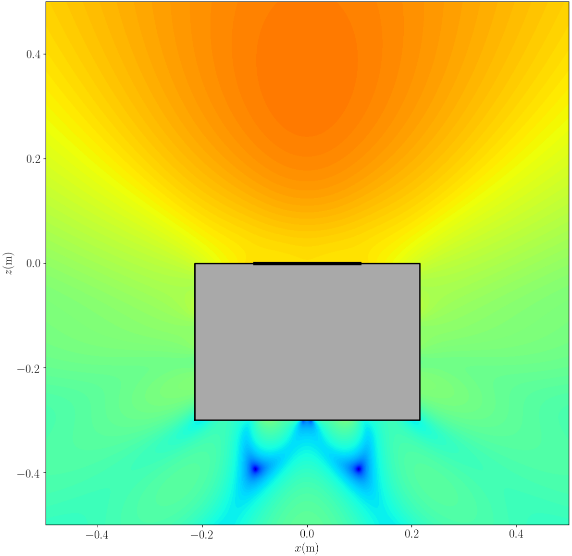
Helmholtz Equation (3D)

Piston radiator in rectangular box



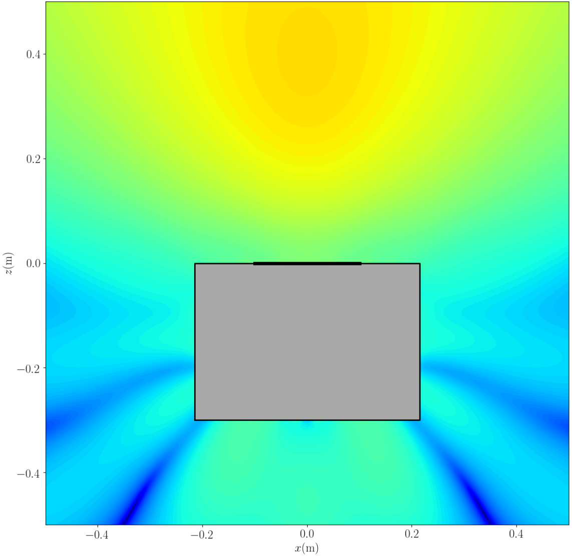
Helmholtz Equation (3D)

Piston radiator in rectangular box



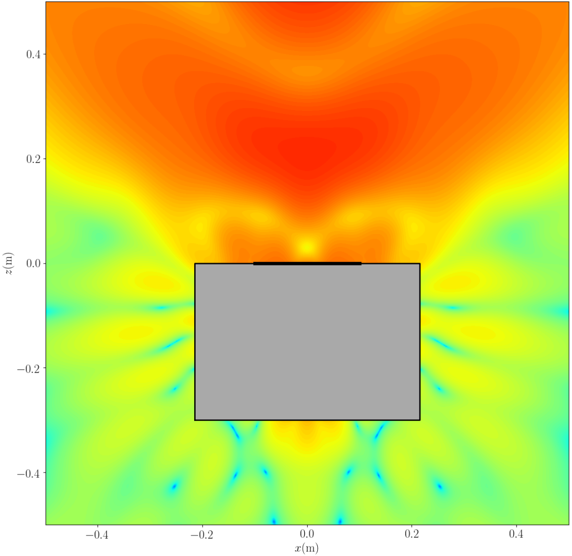
Helmholtz Equation (3D)

Piston radiator in rectangular box



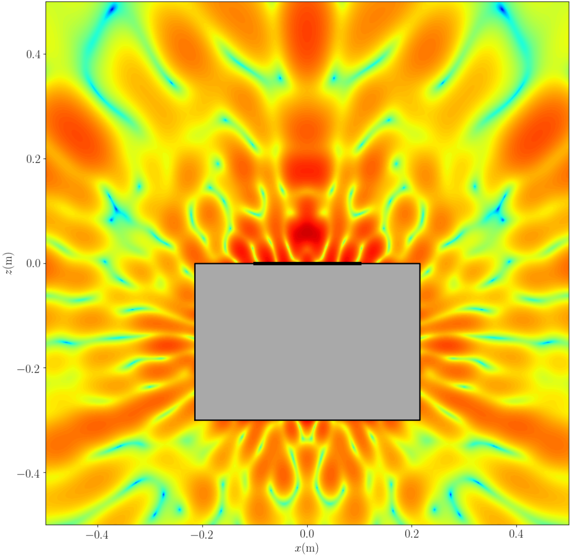
Helmholtz Equation (3D)

Piston radiator in rectangular box



Helmholtz Equation (3D)

Piston radiator in rectangular box



Helmholtz Equation (axisymmetric 2D)

Place equally-spaced sources just inside tweeter

- Simulation of loudspeaker driver on infinite baffle
- Construct source ring for axisymmetric problem

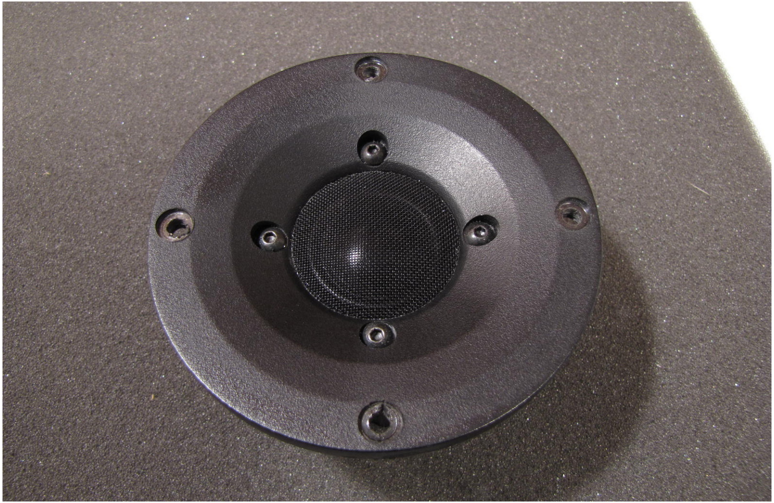
$$\bar{G}_j(r,z) = \oint \frac{d\phi}{2\pi} \frac{e^{ikR_j}}{4\pi R_j} \; ,$$

$$R_j = R_j(r, r_j, z, z_j, \phi - \phi_j)$$

- Use trapezoidal integration with refinement until convergence

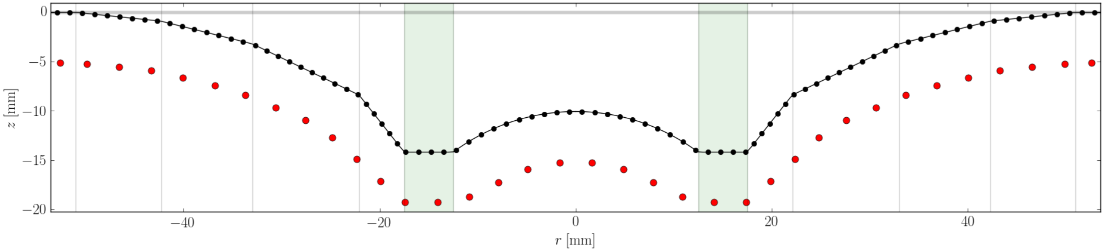
Helmholtz Equation (axisymmetric 2D)

Place equally-spaced sources just inside tweeter



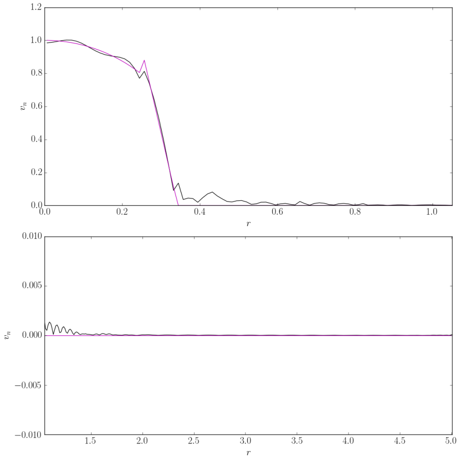
Helmholtz Equation (axisymmetric 2D)

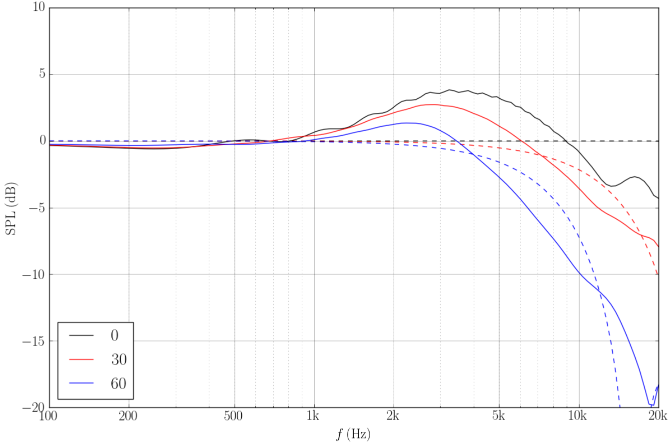
Place equally-spaced sources just inside tweeter



Helmholtz Equation (axisymmetric 2D)

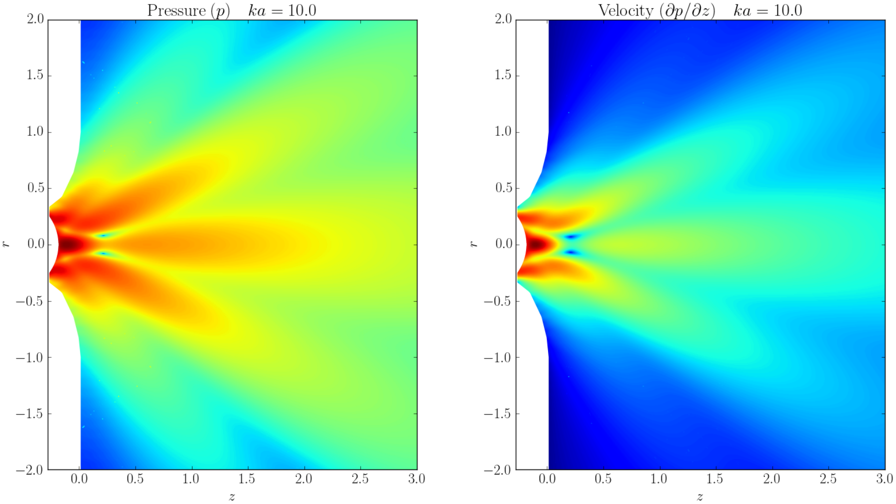
Place equally-spaced sources just inside tweeter





Helmholtz Equation

Choose coefficients to satisfy BCs




FIN