UW Applied Mathematics

Computing Spectra of Linear Operators Using Finite Differences

J. Nathan Kutz

Department of Applied Mathematics
University of Washington
Seattle, WA 98195-2420
Email: (kutz@amath.washington.edu)

Stability Workshop, Seattle, September 6-8, 2006

Introduction: Spectral Stability

UW Applied Mathematics

Consider the nonlinear PDE evolution

$$\dot{u} = N(x, u, u_x, u_{xx}, \ldots)$$

Equilibrium solution

$$N(x, U, U_x, U_{xx}, \ldots) = 0$$

Linear stability

perturbation

$$u(x,t) = U(x) + \underbrace{\epsilon v(x,t)}_{}$$

B. Deconinck and J. N. Kutz, J. Comp. Phys. (2006)

Associated Eigenvalue Problem

Separation of variables

$$v(x,t) = w(x)e^{\lambda t}$$

Eigenvalue problem $\mathscr{L}[U(x)] = \frac{\partial N}{\partial U} + \frac{\partial N}{\partial U_x}\partial_x + \frac{\partial N}{\partial U_{xx}}\partial_x^2 + \cdots$

$$\mathscr{L}[U(x)]w(x) = \lambda w(x)$$

Linear stability

$\mathrm{Re}(\lambda) \leq 0$ **spectrally stable**

$\mathrm{Re}(\lambda) > 0$ **spectrally unstable**

Finite Differences and Taylor Series

Taylor expand

$$f(t+\Delta t) = f(t) + \Delta t \frac{df(t)}{dt} + \frac{\Delta t^2}{2!}\frac{d^2 f(t)}{dt^2} + \frac{\Delta t^3}{3!}\frac{d^3 f(c_1)}{dt^3}$$

$$f(t-\Delta t) = f(t) - \Delta t \frac{df(t)}{dt} + \frac{\Delta t^2}{2!}\frac{d^2 f(t)}{dt^2} - \frac{\Delta t^3}{3!}\frac{d^3 f(c_2)}{dt^3}$$

Add

$$\frac{df(t)}{dt} = \overset{\text{approximation}}{\frac{f(t+\Delta t) - f(t-\Delta t)}{2\Delta t}} - \overset{\text{error}}{\frac{\Delta t^2}{6}\frac{d^3 f(c)}{dt^3}}$$

slope formula with error

Higher-Order Accuracy

UW Applied Mathematics

Taylor expand again

$$f(t+\Delta t)-f(t-\Delta t) = 2\Delta t\frac{df(t)}{dt} + \frac{2\Delta t^3}{3!}\frac{d^3f(t)}{dt^3} + \frac{\Delta t^5}{5!}\left(\frac{d^5f(c1)}{dt^5} + \frac{d^5f(c2)}{dt^5}\right)$$

$$f(t+2\Delta t)-f(t-2\Delta t) = 4\Delta t\frac{df(t)}{dt} + \frac{16\Delta t^3}{3!}\frac{d^3f(t)}{dt^3} + \frac{32\Delta t^5}{5!}\left(\frac{d^5f(c3)}{dt^5} + \frac{d^5f(c4)}{dt^5}\right)$$

8 x (first) and subtract

$$\frac{df(t)}{dt} = \underbrace{\frac{-f(t+2\Delta t)+8f(t+\Delta t)-8f(t-\Delta t)+f(t-2\Delta t)}{12\Delta t}}_{\text{approximation}} + \underbrace{\frac{\Delta t^4}{30}f^{(5)}(c)}_{\text{error}}$$

slope formula with improved error

Finite Difference Tables

$O(\Delta t^2)$ center-difference schemes

$$f'(t) = [f(t+\Delta t) - f(t-\Delta t)]/2\Delta t$$
$$f''(t) = [f(t+\Delta t) - 2f(t) + f(t-\Delta t)]/\Delta t^2$$
$$f'''(t) = [f(t+2\Delta t) - 2f(t+\Delta t) + 2f(t-\Delta t) - f(t-2\Delta t)]/2\Delta t^3$$
$$f''''(t) = [f(t+2\Delta t) - 4f(t+\Delta t) + 6f(t) - 4f(t-\Delta t) + f(t-2\Delta t)]/\Delta t^4$$

$O(\Delta t^4)$ center-difference schemes

$$f'(t) = [-f(t+2\Delta t) + 8f(t+\Delta t) - 8f(t-\Delta t) + f(t-2\Delta t)]/12\Delta t$$
$$\begin{aligned} f''(t) = [-f(t+2\Delta t) + 16f(t+\Delta t) - 30f(t) \\ +16f(t-\Delta t) - f(t-2\Delta t)]/12\Delta t^2 \end{aligned}$$
$$\begin{aligned} f'''(t) = [-f(t+3\Delta t) + 8f(t+2\Delta t) - 13f(t+\Delta t) \\ +13f(t-\Delta t) - 8f(t-2\Delta t) + f(t-3\Delta t)]/8\Delta t^3 \end{aligned}$$
$$\begin{aligned} f''''(t) = [-f(t+3\Delta t) + 12f(t+2\Delta t) - 39f(t+\Delta t) + 56f(t) \\ -39f(t-\Delta t) + 12f(t-2\Delta t) - f(t-3\Delta t)]/6\Delta t^4 \end{aligned}$$

neighboring points determine accuracy

Forward and Backward Differences

UW Applied Mathematics

$O(\Delta t^2)$ forward- and backward-difference schemes

$$
\begin{aligned}
f'(t) &= [-3f(t) + 4f(t+\Delta t) - f(t+2\Delta t)]/2\Delta t \\
f'(t) &= [3f(t) - 4f(t-\Delta t) + f(t-2\Delta t)]/2\Delta t \\
f''(t) &= [2f(t) - 5f(t+\Delta t) + 4f(t+2\Delta t) - f(t+3\Delta t)]/\Delta t^3 \\
f''(t) &= [2f(t) - 5f(t-\Delta t) + 4f(t-2\Delta t) - f(t-3\Delta t)]/\Delta t^3
\end{aligned}
$$

Required for incorporating boundary conditions

asymmetric neighboring points

Numerical Round-Off

Consider the error in approximating the first derivative

$$\frac{dy}{dt} = \frac{-f(t+2\Delta t) + 8f(t+\Delta t) - 8f(t-\Delta t) + f(t-2\Delta t)}{12\Delta t} + E(y(t), \Delta t)$$

The error includes round-off and truncation

$$E = \underbrace{\frac{-e(t+2\Delta t) + 8e(t+\Delta t) - 8e(t-\Delta t) + e(t-2\Delta t)}{12\Delta t}}_{\text{round-off}} + \underbrace{\frac{\Delta t^4}{30}\frac{d^5 y(c)}{dt^5}}_{\text{truncation}}$$

Assume round-off $|e(t+\Delta t)| \leq e_r$ and $M = \max\{|y'''''(c)|\}$

$$|E| = \frac{3e_r}{2\Delta t} + \frac{\Delta t^4 M}{30}$$ minimum at $$\boxed{\Delta t = \left(\frac{45 e_r}{4M}\right)^{1/5}}$$

round-off error dominates below $\Delta t \approx 10^{-3}$

Boundary Conditions: Pinned $u(0)=u(l)=0$

UW Applied Mathematics

$$\frac{\partial^2 u}{\partial x^2} \rightarrow \frac{1}{\Delta x^2}\begin{bmatrix} -2 & 1 & 0 & \cdots & 0 & 0 \\ 1 & -2 & 1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & & \\ \vdots & & & & & \vdots \\ & & & & & 0 \\ \vdots & \cdots & 0 & 1 & -2 & 1 \\ 0 & 0 & \cdots & 0 & 1 & -2 \end{bmatrix}\begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix}$$

pinned boundaries $\quad u_0 = u_{n+1} = 0$

tri-diagonal matrix structure

Boundary Conditions: Periodic $u(0) = u(l)$

$$\frac{\partial^2 u}{\partial x^2} \rightarrow \frac{1}{\Delta x^2} \begin{bmatrix} -2 & 1 & 0 & \cdots & 0 & 1 \\ 1 & -2 & 1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & & \\ \vdots & & & & & \vdots \\ & & & & & 0 \\ \vdots & \cdots & 0 & 1 & -2 & 1 \\ 1 & 0 & \cdots & 0 & 1 & -2 \end{bmatrix} \begin{pmatrix} u_0 \\ u_1 \\ \vdots \\ u_n \end{pmatrix}$$

periodic boundaries $\qquad u_0 = u_{n+1}$

tri-diagonal matrix structure with corners

Boundary Conditions: No Flux $\frac{\partial u(0)}{\partial x} = \frac{\partial u(l)}{\partial x} = 0$

no flux condition

$$\frac{\partial u(0)}{\partial x} = \frac{-3u_0 + 4u_1 - u_2}{2\Delta x} = 0 \quad \rightarrow \quad u_0 = \frac{4}{3}u_1 - \frac{1}{3}u_2$$

$$\frac{\partial^2 u}{\partial x^2} \rightarrow \frac{1}{\Delta x^2} \begin{bmatrix} -2/3 & 2/3 & 0 & \cdots & 0 & 0 \\ 1 & -2 & 1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & & \\ \vdots & & & & & \vdots \\ & & & & & 0 \\ \vdots & \cdots & 0 & 1 & -2 & 1 \\ 0 & 0 & \cdots & 0 & 2/3 & -2/3 \end{bmatrix}$$

no longer symmetry matrix

General Boundaries

General (Sturm-Liouville) boundary conditions

$$\alpha_1 u(0) + \beta_1 \frac{\partial u(0)}{\partial x} = \gamma_1$$
$$\alpha_2 u(l) + \beta_2 \frac{\partial u(l)}{\partial x} = \gamma_2$$

Difficult to incorporate into matrix structure

- **shooting methods**
- **relaxation methods**

no longer symmetry matrix

Algorithm

- choose domain length and discretization size
- construct linear operator
- implement boundary conditions
- use eigenvalue/eigenvector solver: $O(N^3)$
  (or shooting/relaxation methods)
- construct eigenfunctions

Easily extends to vectors and higher dimensions

Example: Mathieu Equation

Classic Example

$$y'' + (a - 2q\cos(2x))y = 0 \iff -y'' + 2q\cos(2x)y = ay$$

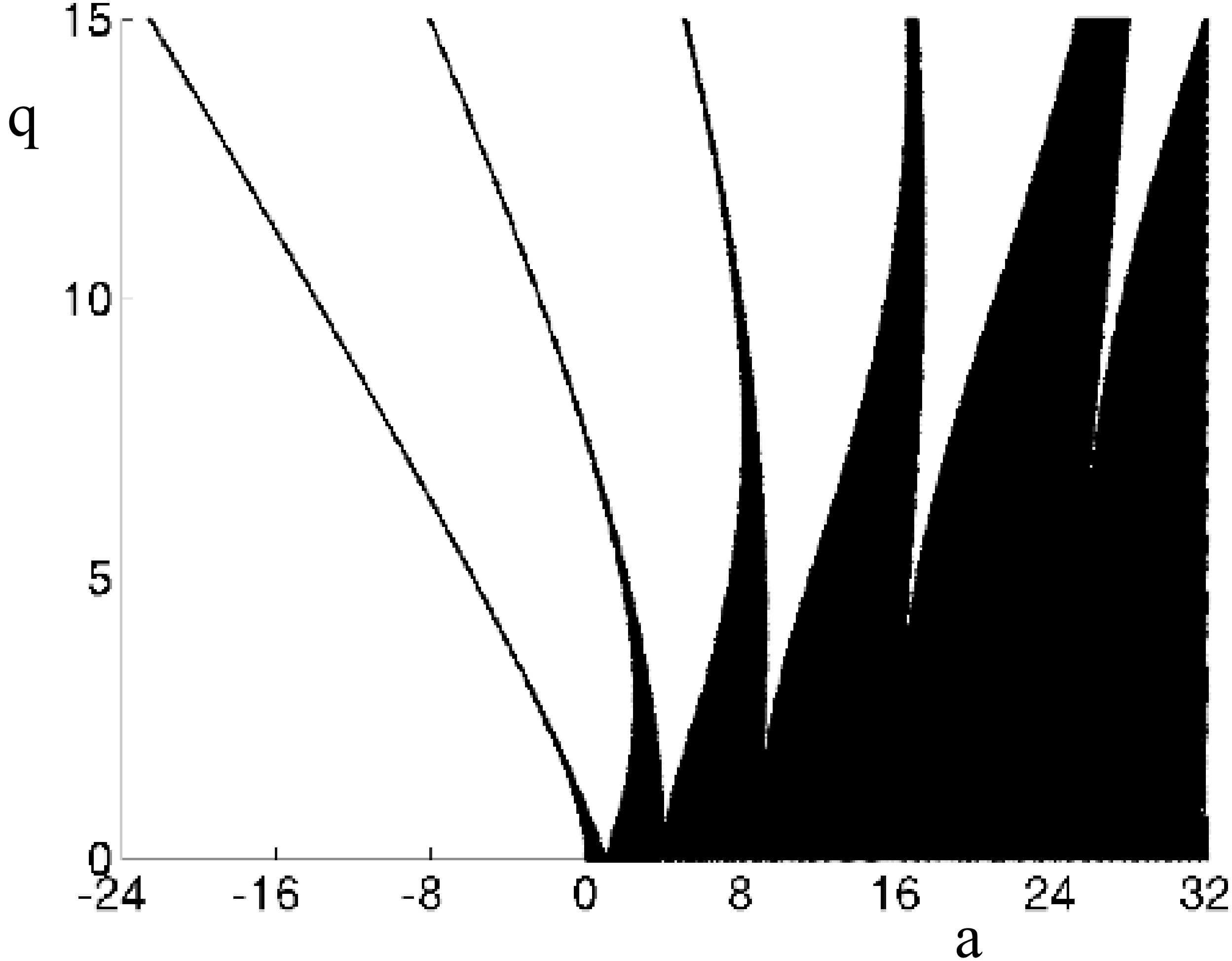
Operator $-\partial_x^2 + 2q\cos(2x)$ is self-adjoint (real spectrum)

$$-\frac{1}{\Delta x^2}\begin{bmatrix} -2 & 1 & 0 & \cdots & 0 & 1 \\ 1 & -2 & 1 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & & \\ \vdots & & & & & \vdots \\ & & & & & 0 \\ \vdots & \cdots & 0 & 1 & -2 & 1 \\ 1 & 0 & \cdots & 0 & 1 & -2 \end{bmatrix} + 2q \begin{bmatrix} \cos(2x_0) & 0 & 0 & \cdots & 0 & 0 \\ 0 & \cos(2x_1) & 0 & 0 & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & & \\ \vdots & & & & & \vdots \\ & & & & & 0 \\ \vdots & \cdots & 0 & 0 & \cos(2x_{n-1}) & 0 \\ 0 & 0 & \cdots & 0 & 0 & \cos(2x_n) \end{bmatrix}$$

compute with matlab, maple, mathematica, or homemade code

Spectrum for Mathieu Equation

UW Applied Mathematics



a is eigenvalue

Computing the Ground State

Convergence study and CPU time (q=2)

	Accuracy: 10^{-3}		Accuracy: 10^{-6}		Accuracy: 10^{-9}	
	Matrix size	CPU time	Matrix size	CPU time	Matrix size	CPU time
FDM2	239	0.43 sec	8000	1 hour	N/A	-
FDM4	52	0.01 sec	293	0.77 sec	1630	2.5 min

beyond Matlab7's ability

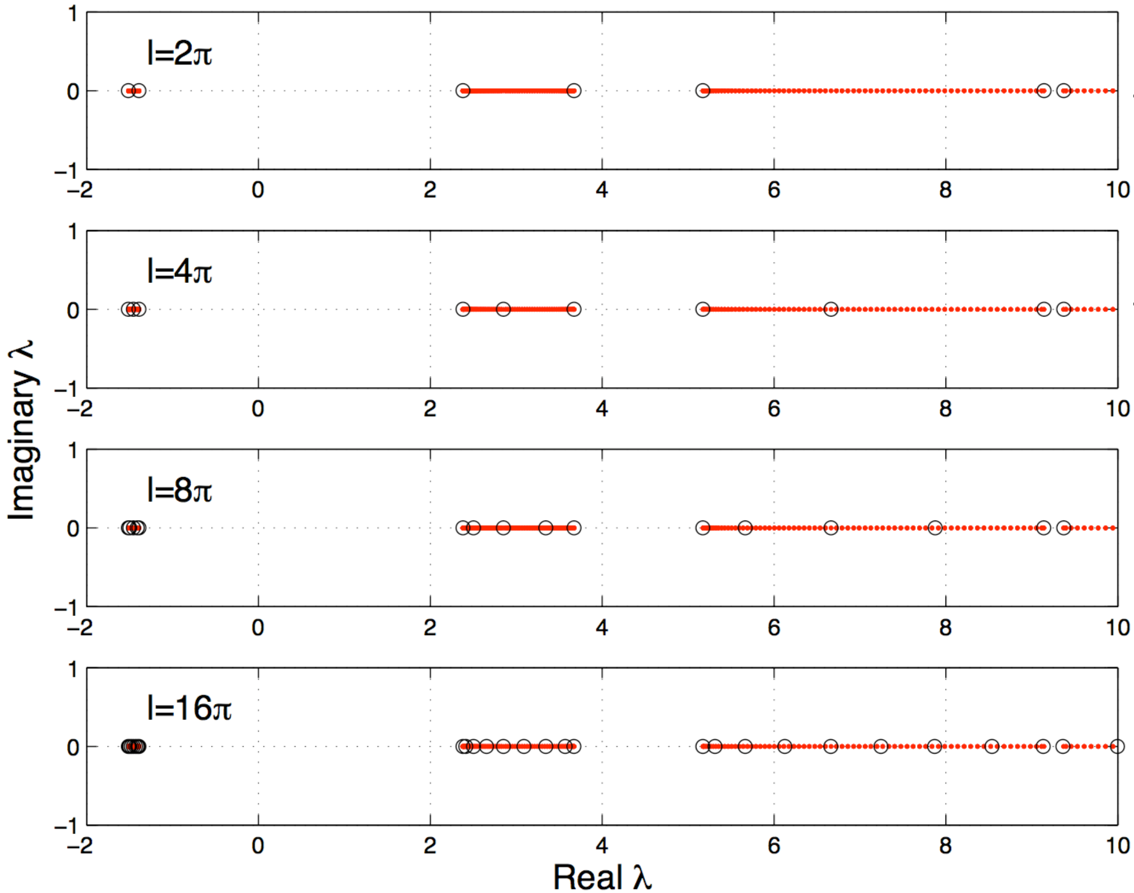
What about band-gap structure

- **increase domain length**
- **Floquet theory**

Calculating the Bands: Domain Length

UW Applied Mathematics

Increase the domain length (q=2)



doubling gives 8x computational increase

traditional way: very costly for recovering bands

Calculating the Bands: Floquet Theory

Make use of Floquet (Bloch) theory

larger period solutions

$$y(x+L) = \mathrm{e}^{\mathrm{i}\mu L} y(x)$$

with Floquet (characteristic) exponents $\mu \in [0, 2\pi/L)$

- **keep fixed domain**
- **discretize** $\mu_k = 2(k-1)/P, \; k = 1, \ldots, D$
- **solve D O(N³) equations**

Implementing Floquet Theory

Floquet theory modifies matrix corners

$$D_2^{(4,h)} = \frac{1}{12h^2} \begin{pmatrix} -30 & 16 & -1 & 0 & \cdots & \cdots & 0 & -e^{-i\theta} & 16e^{-i\theta} \\ 16 & -30 & 16 & -1 & 0 & \cdots & \cdots & 0 & -e^{-i\theta} \\ -1 & 16 & -30 & 16 & -1 & 0 & \cdots & \cdots & 0 \\ 0 & -1 & 16 & -30 & 16 & -1 & 0 & \cdots & \cdots \\ & & & \ddots & \ddots & \ddots & & & \\ \cdots & \cdots & 0 & -1 & 16 & -30 & 16 & -1 & 0 \\ 0 & \cdots & \cdots & 0 & -1 & 16 & -30 & 16 & -1 \\ -e^{i\theta} & 0 & \cdots & \cdots & 0 & -1 & 16 & -30 & 16 \\ 16e^{i\theta} & -e^{i\theta} & 0 & \cdots & \cdots & 0 & -1 & 16 & 30 \end{pmatrix}$$

with Floquet slices $\theta \in [0, 2\pi)$

UW Applied Mathematics

Floquet Theory vs. Domain Length

Compare methods for computing band (q=2)

band density

$$\delta_B = \frac{\max\limits_{a_k, a_{k+1} \in \sigma_B} (a_{k+1} - a_k)}{|\sigma_B|}$$

	$\delta_2 = 0.25$		$\delta_2 = 0.025$		$\delta_2 = 0.0025$	
	D	CPU time	D	CPU time	D	CPU time
FDM4 matrix size=1172	1	9 min	N/A	-	N/A	-
FDM4, $D > 1$	4	8 sec	32	1 min	310	10 min

beyond Matlab7's ability

Use Floquet Theory!

Example: Periodic NLS

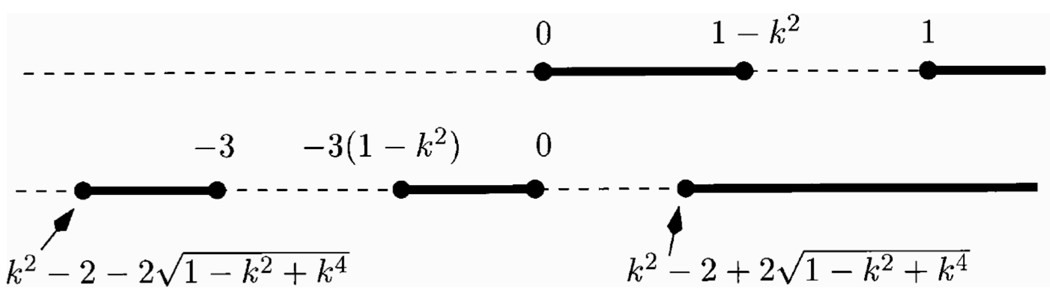
Consider the system

$$\begin{pmatrix} 0 & L_-(k) \\ -L_+(k) & 0 \end{pmatrix} \begin{pmatrix} U \\ V \end{pmatrix} = \lambda \begin{pmatrix} U \\ V \end{pmatrix}$$

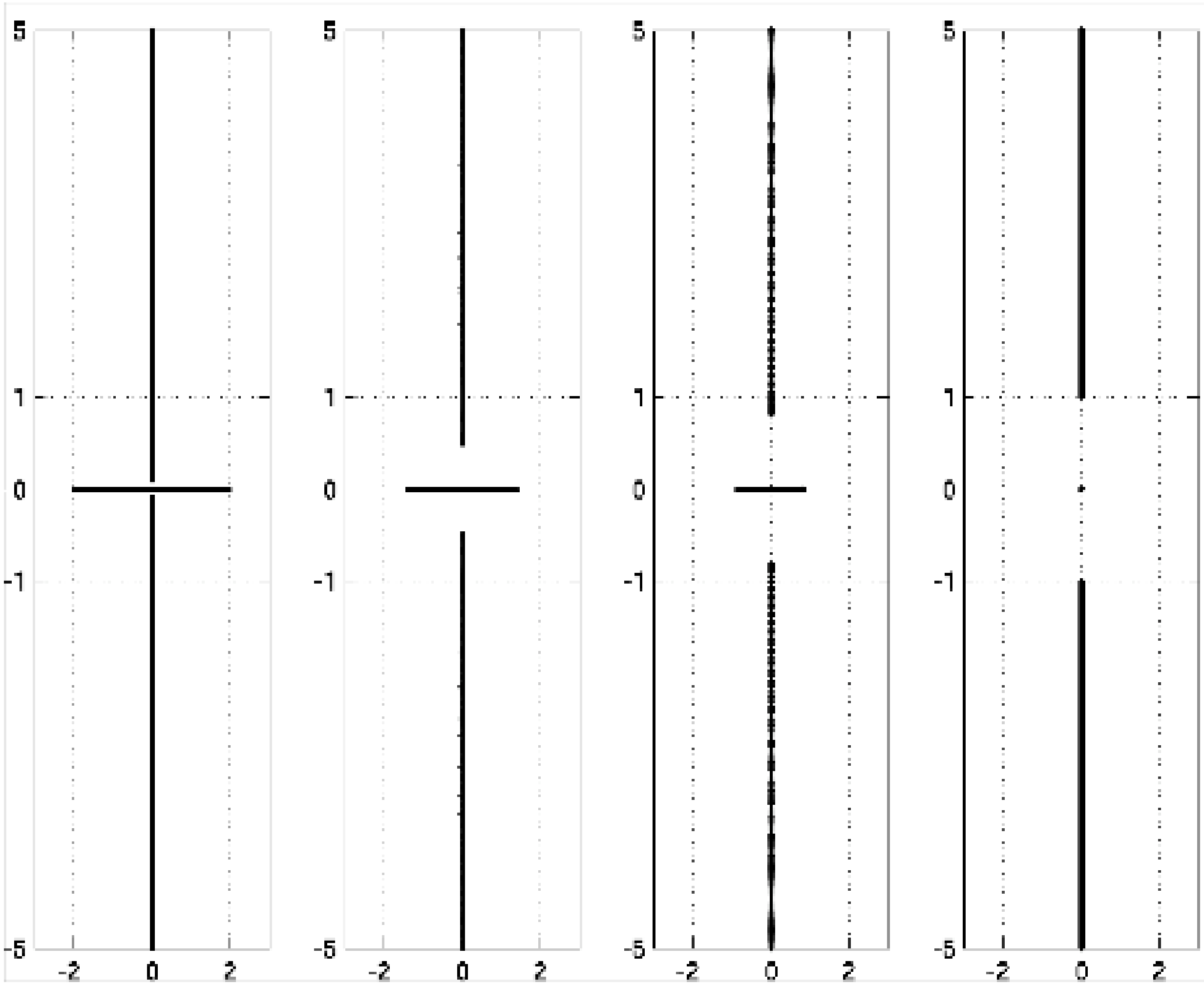
with

$$L_-(k)u = -u'' + (2k^2 \mathrm{sn}^2(x,k) - k^2)u$$
$$L_+(k)v = -v'' + (6k^2 \mathrm{sn}^2(x,k) - 4 - k^2)v$$

and Jacobi sine function $\mathrm{sn}(x,k)$

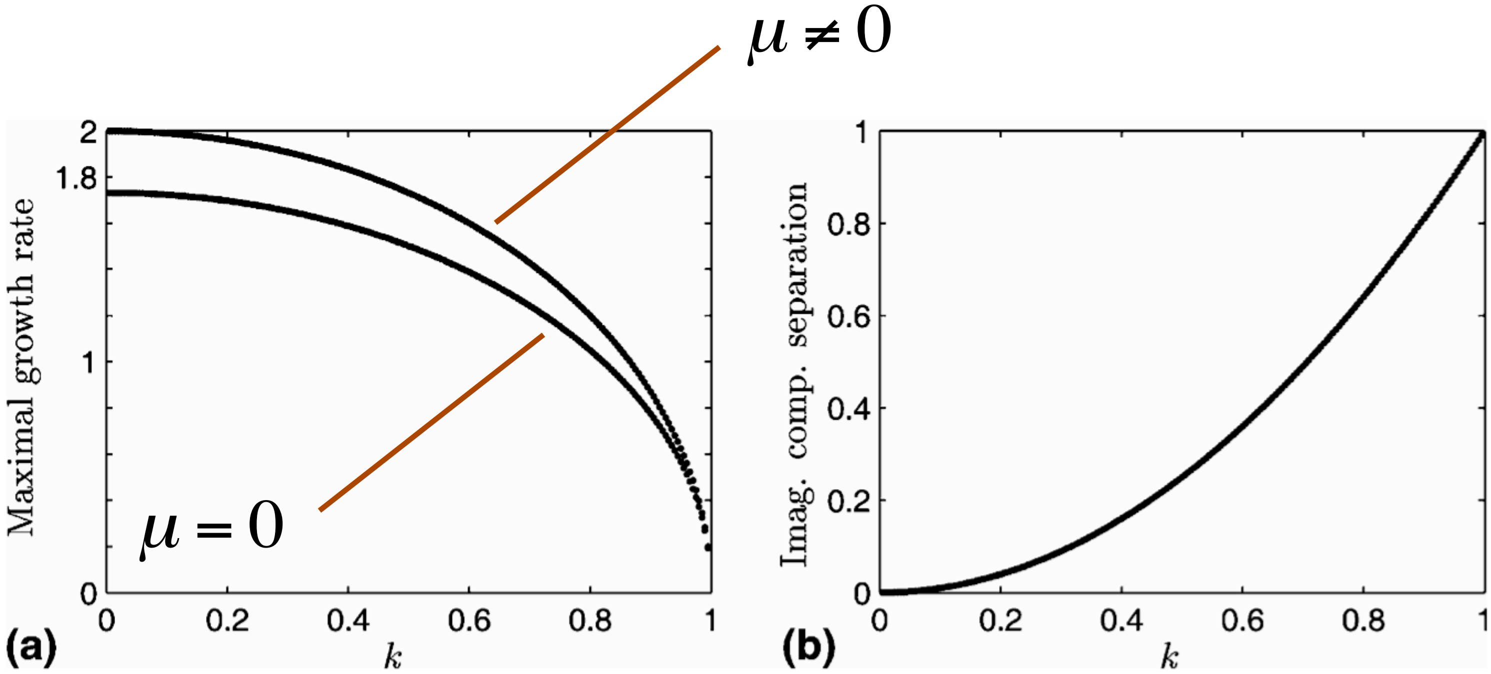


Spectrum of Periodic NLS



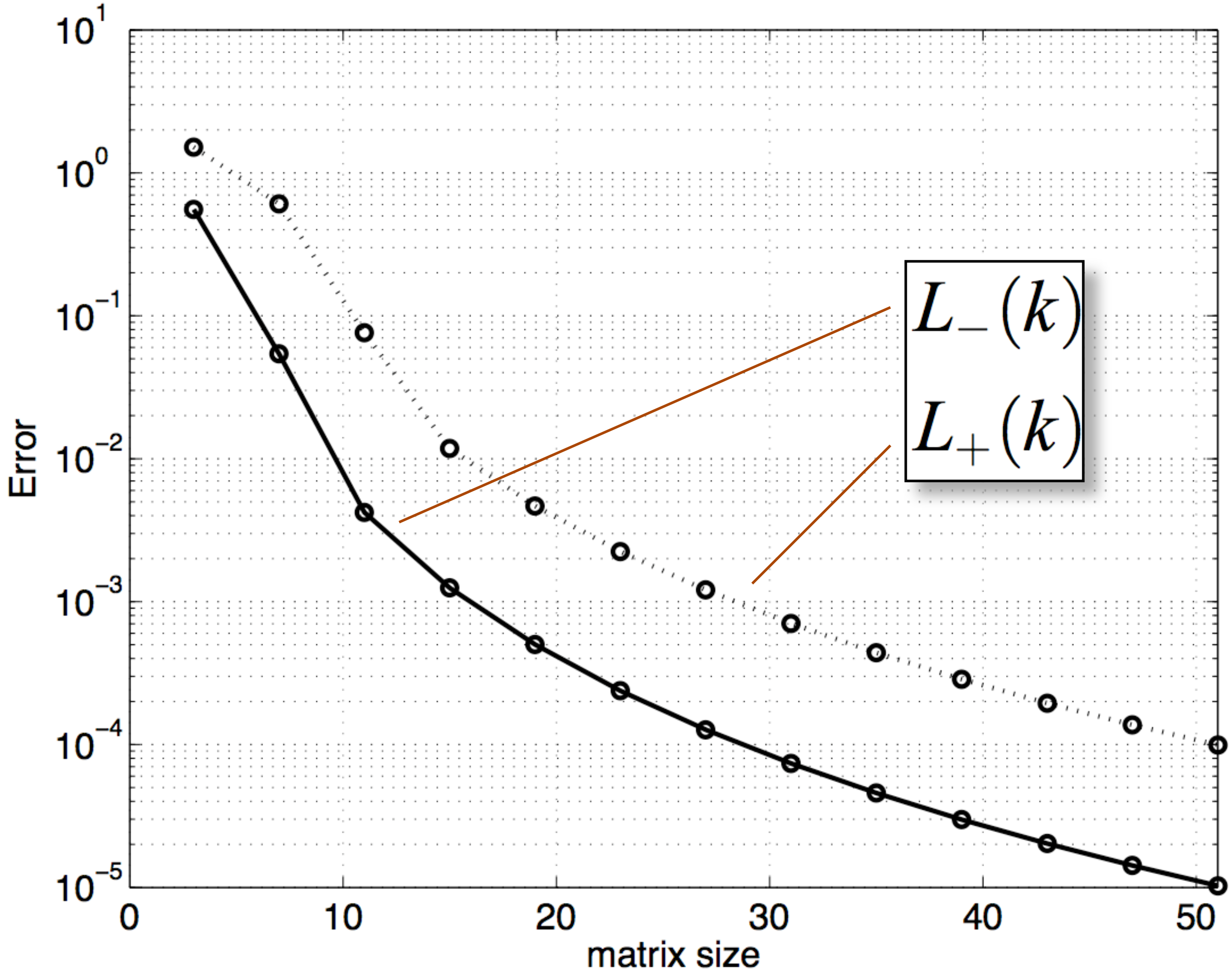
UW Applied Mathematics

Importance of Floquet Slicing



UW Applied Mathematics

Accuracy and Convergence



Example: 2D Mathieu Equation

Consider

$$-\frac{1}{2}(\psi_{xx}+\psi_{yy})+f(x,y)\psi=\lambda\psi$$

with

$$f(x,y)=A\sin^2 x\sin^2 y$$

Operation Count: $O((N^2)^3)=O(N^6)$

Laplacian in 2D

Consider $$\nabla^2\psi = \frac{\partial^2\psi}{\partial x^2} + \frac{\partial^2\psi}{\partial y^2}$$

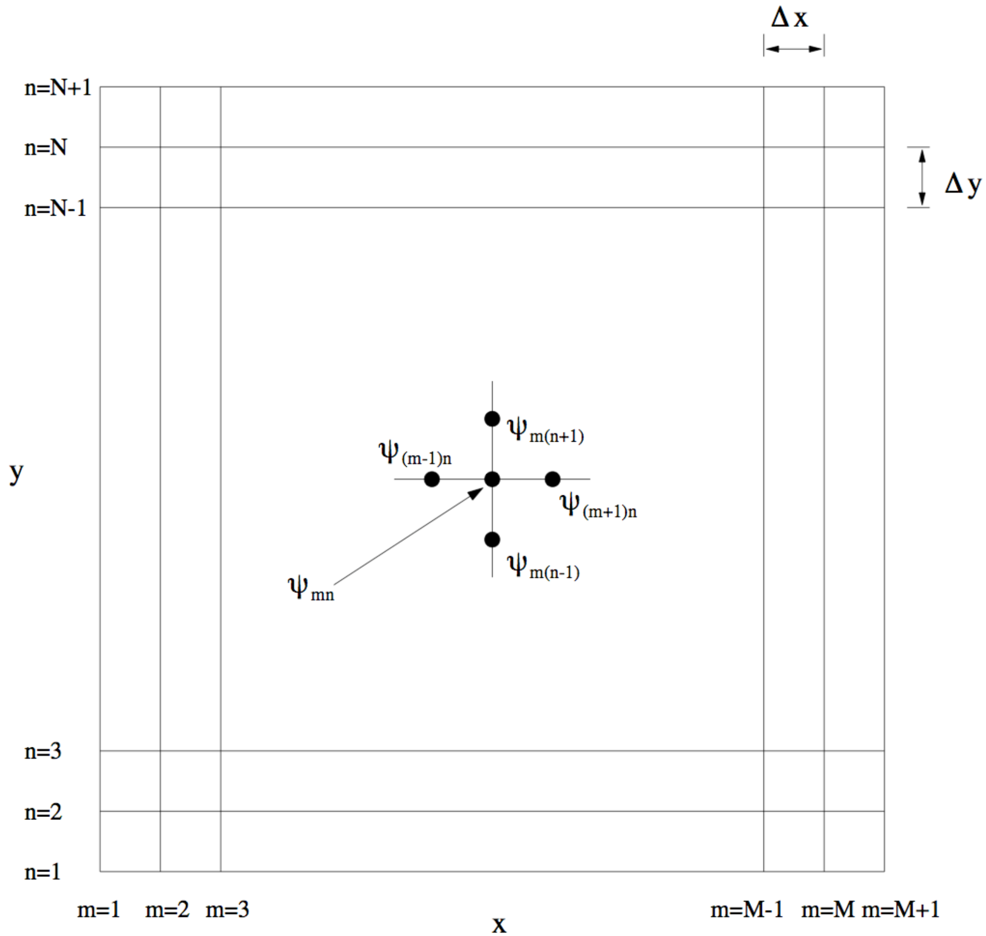
Discretize: $$\frac{\psi(x+\Delta x,y,t) - 2\psi(x,y,t) + \psi(x-\Delta x,y,t)}{\Delta x^2} + \frac{\psi(x,y+\Delta y,t) - 2\psi(x,y,t) + \psi(x,y-\Delta y,t)}{\Delta y^2}$$

Let $\psi_{mn} = \psi(x_m, y_n)$

$$-4\psi_{mn} + \psi_{(m-1)n} + \psi_{(m+1)n} + \psi_{m(n-1)} + \psi_{m(n+1)}$$

must stack 2D data: periodic boundaries add structure

Laplacian in 2D



$$\psi = \begin{pmatrix} \psi_{11} \\ \psi_{12} \\ \vdots \\ \psi_{1(M+1)} \\ \psi_{21} \\ \psi_{22} \\ \vdots \\ \psi_{(N+1)M} \\ \psi_{(N+1)(M+1)} \end{pmatrix}$$

Laplacian in 2D

$$
\begin{bmatrix}
-4 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
1 & -4 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & -4 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & -4 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & -4 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 1 & -4 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & -4 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 1 & -4 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -4 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & -4 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & -4 & 1 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & -4 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -4 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & -4 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & -4 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & -4
\end{bmatrix}
$$

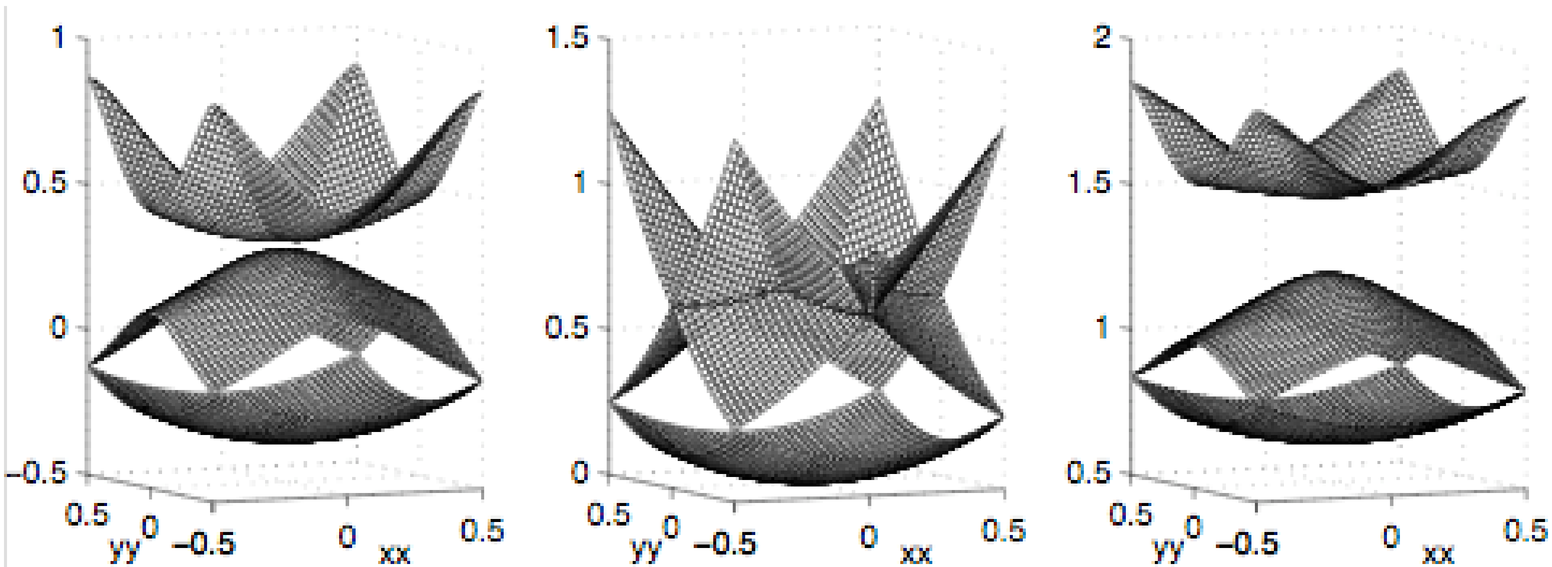
nx=ny=4

```
I=eye(n); % idendity matrix of size nXn
B=kron(I,A)+kron(A,I); % 2-D differentiation matrix
```

Matlab easily builds 2D Laplacian

Band Gap Structure

UW Applied Mathematics



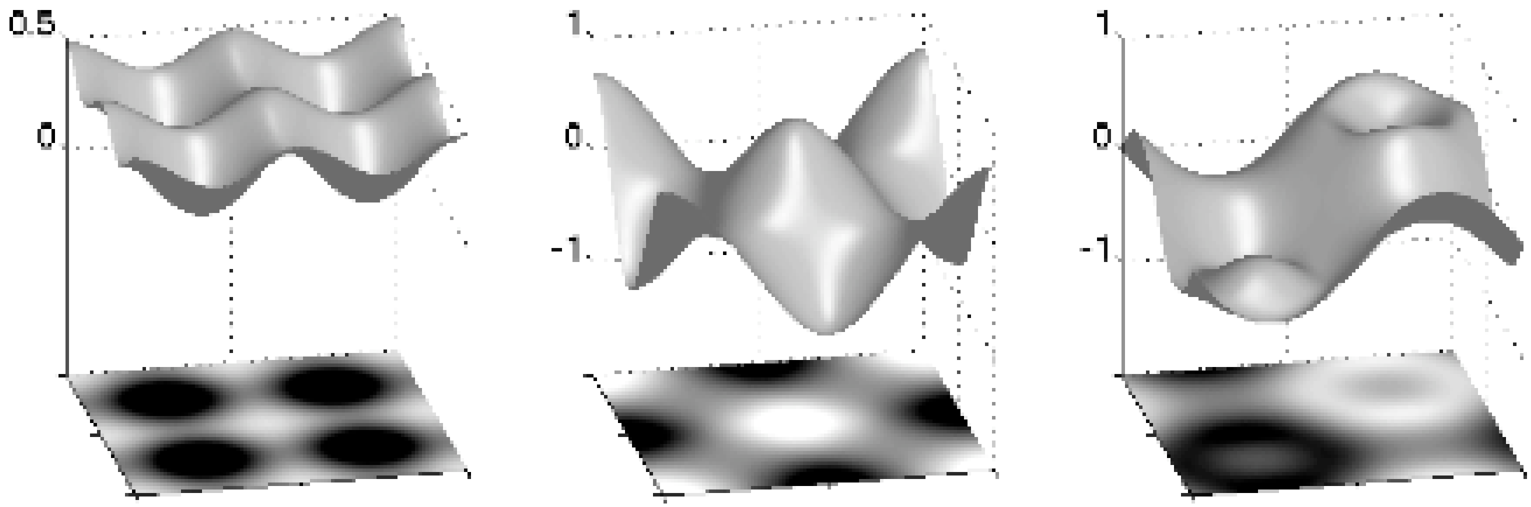
A=-0.3 A=0 A=1

First three band-gap structures

Quasi-momentum representation

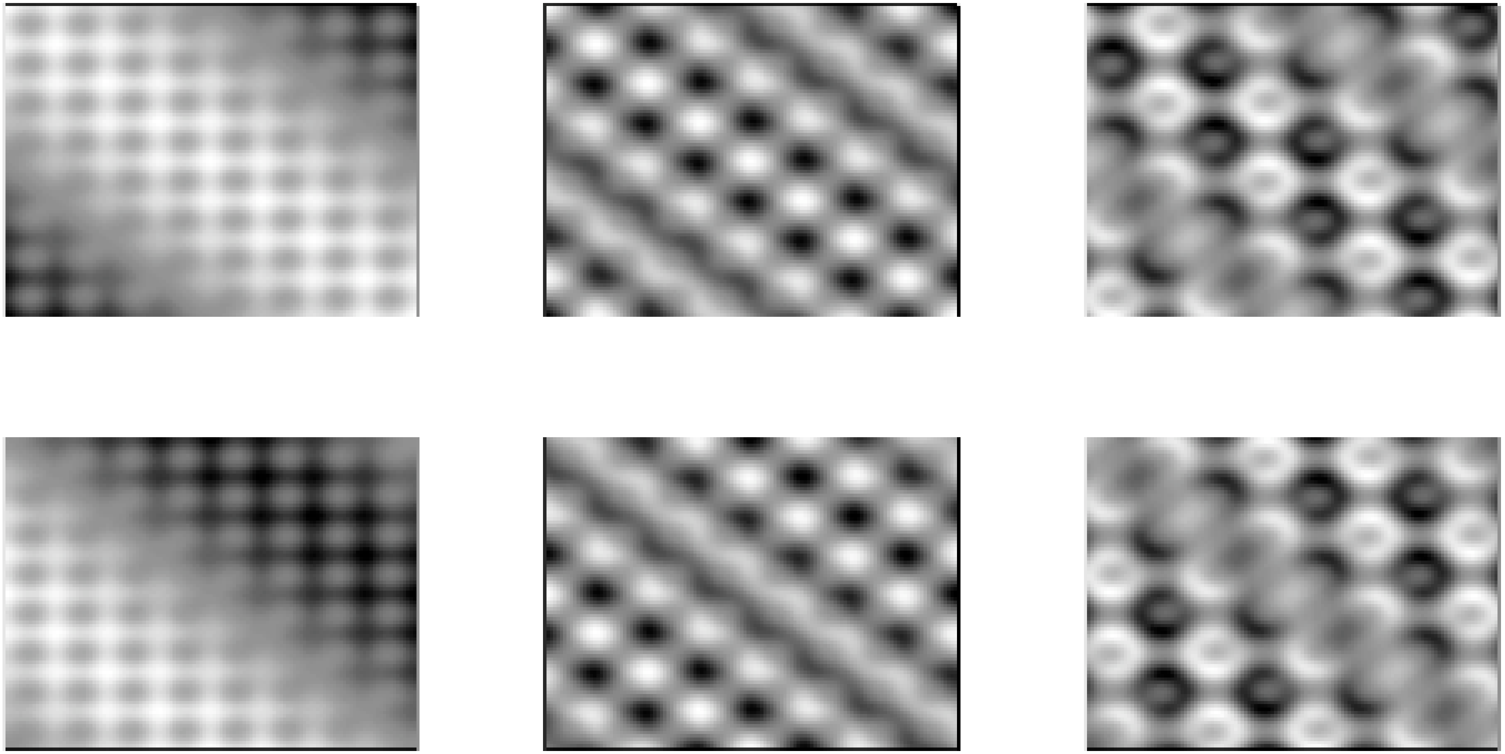
2D Dominant Eigenfunctions

UW Applied Mathematics



First three eigenfunctions for $\mu_x = \mu_y = 0$

2D Eigenfunctions



First three eigenfunctions for $\mu_x = \mu_y = 1/4$

UW Applied Mathematics

Summary and Conclusions

UW Applied Mathematics

- simple, simple, simple
- boundary conditions at edge of matrices
- eigenvalue solvers make use of sparse structure
- Floquet theory for resolution of bands
- costly/impractical for 2D-3D problems

B. Deconinck and J. N. Kutz, J. Comp. Phys. (2006)