Solving large scale eigenvalue problems

Lecture 1, Feb 21, 2018: Introduction

http://people.inf.ethz.ch/arbenz/ewp/

Peter Arbenz
Computer Science Department, ETH Zürich
E-mail: arbenz@inf.ethz.ch

Introduction: Survey on lecture

1. Introduction (today)
   - What makes eigenvalues interesting?
   - Some examples.
2. Some linear algebra basics
   - Definitions
   - Similarity transformations
   - Schur decompositions
   - SVD
   - Jordan normal forms
   - Functions of matrices
3. Newton's method for linear and nonlinear eigenvalue problems
4. The QR Algorithm for dense eigenvalue problems
5. Vector iteration (power method) and subspace iterations

Introduction: Survey on lecture (cont.)

6. Krylov subspaces methods
   - Arnoldi and Lanczos algorithms
   - Krylov-Schur methods
7. Davidson/Jacobi-Davidson methods
8. Rayleigh quotient minimization for symmetric systems
9. Locally-optimal block preconditioned conjugate gradient (LOBPCG) method

Lecture notes at
http://people.inf.ethz.ch/arbenz/ewp/lnotes.html

Literature

- Z. Bai, J. Demmel, J. Dongarra, A. Ruhe, and H. van der Vorst. *Templates for the Solution of Algebraic Eigenvalue Problems: A Practical Guide*. SIAM, Philadelphia, 2000.
- Y. Saad. *Numerical Methods for Large Eigenvalue Problems*. SIAM, Philadelphia, 1992. Revised version 2011.
- G. W. Stewart. *Matrix Algorithms II: Eigensystems*. SIAM, Philadelphia, 2001.
- G. H. Golub and C. F. van Loan. *Matrix Computations*, 4th edition. Johns Hopkins University Press. Baltimore, 2012.
- J. W. Demmel. *Applied Numerical Linear Algebra*. SIAM, Philadelphia, 1997.

Organization

- 12–13 lectures
- No lecture on April 4 (easter break) and May 30.
- Complementary exercises
  - To get hands-on experience
  - Based on MATLAB
- Examination
  - First week of semester break (week of June 4)
  - 30' oral
  - No testat required

- Introduction
  - What makes eigenvalues interesting?
  - Example 1: The vibrating string
  - Numerical methods for solving 1-dimensional problems
  - Example 2: The heat equation
  - Example 3: The wave equation
  - The 2D Laplace eigenvalue problem
  - (Cavity resonances in particle accelerators)
  - Spectral clustering
  - Google's PageRank
  - (Other sources of eigenvalue problems)

- In physics, eigenvalues are usually connected to vibrations. (violin strings, drums, bridges, sky scrapers)
  Prominent examples of vibrating structures.
  - On November 7, 1940, the Tacoma narrows bridge collapsed, less than half a year after its opening. Strong winds excited the bridge so much that the platform in reinforced concrete fell into pieces.
  - A few years ago the London millennium footbridge started wobbling in a way that it had to be closed. The wobbling had been excited by the pedestrians passing the bridge, see https://www.youtube.com/watch?v=eAXVa__XWZ8
- Electric fields in cyclotrons (particle accelerators)
- The solutions of the Schrödinger equation from quantum physics and quantum chemistry have solutions that correspond to vibrations of the, say, molecule it models. The eigenvalues correspond to energy levels that molecule can occupy.

Many characteristic quantities in science *are* eigenvalues:

- decay factors,
- frequencies,
- norms of operators (or matrices),
- singular values,
- condition numbers.

Notations

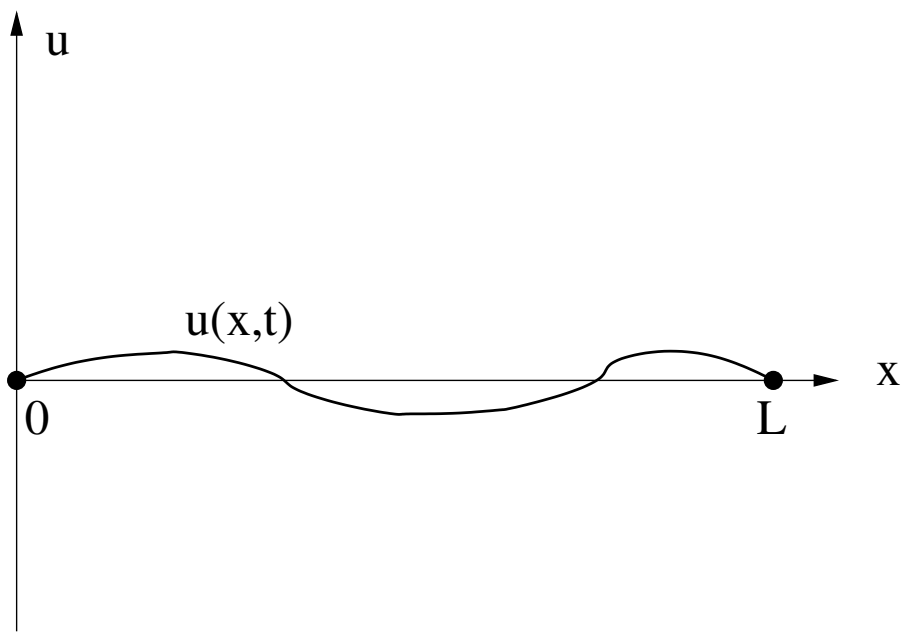
Scalars : lowercase letters, a, b, c. . ., and $\alpha, \beta, \gamma \ldots$.

Vectors : boldface lowercase letters, $\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}, \ldots$.

Matrices : uppercase letters, A, B, C. . ., and $\Gamma, \Delta, \Lambda, \ldots$.

Example 1: The vibrating string

A vibrating string fixed at both ends.

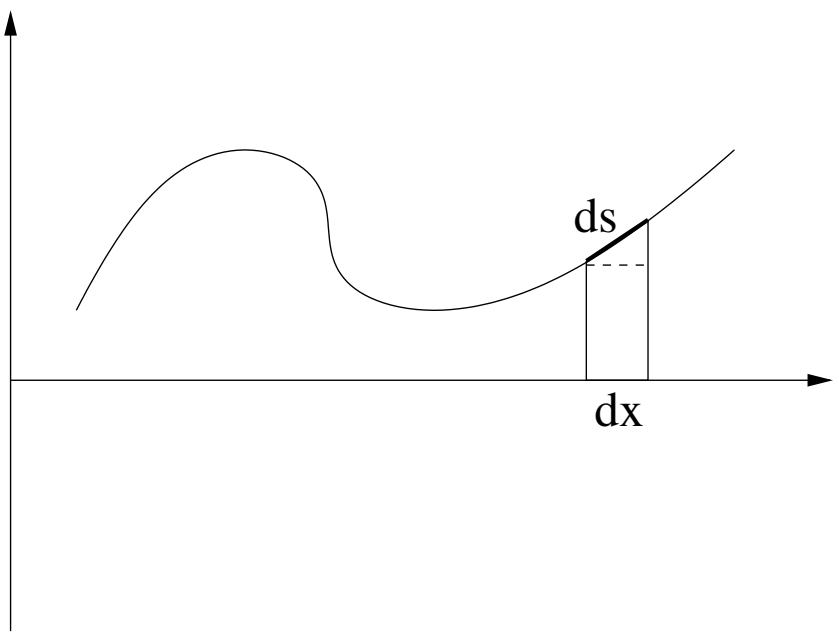


- $u(x,t)$: The displacement of the rest position at x, $0 < x < L$, and time t.
- Assume $\left|\frac{\partial u}{\partial x}\right|$ is small.
- $v(x,t)$:the velocity of the string at position x and at time t.

The kinetic energy of a string

The kinetic energy of a string section ds of mass $dm = \rho\, ds$:

$$dT = \frac{1}{2} dm\, v^2 = \frac{1}{2} \rho\, ds \left(\frac{\partial u}{\partial t}\right)^2 . \qquad (1)$$



- $ds^2 = dx^2 + \left(\frac{\partial u}{\partial x}\right)^2 dx^2$

$$\Rightarrow \frac{ds}{dx} = \sqrt{1 + \left(\frac{\partial u}{\partial x}\right)^2}$$
$$= 1 + \text{h.o.t.}$$

h.o.t. = higher order terms.

The kinetic energy of a string (cont.)

Plugging this into (1) and omitting also the second order term (leaving just the number 1) gives

$$dT = \frac{\rho\, dx}{2} \left(\frac{\partial u}{\partial t} \right)^2 .$$

The kinetic energy of the whole string:

$$T = \int_0^L dT(x) = \frac{1}{2} \int_0^L \rho(x) \left(\frac{\partial u}{\partial t} \right)^2 dx$$

The potential energy of the string

1. the stretching times the exerted strain τ,

$$\tau \int_0^L ds - \tau \int_0^L dx = \tau \int_0^L \left(\sqrt{1 + \left(\frac{\partial u}{\partial x} \right)^2} - 1 \right) dx$$
$$= \tau \int_0^L \left(\frac{1}{2} \left(\frac{\partial u}{\partial x} \right)^2 + \text{ h.o.t.} \right) dx$$

2. exterior forces of density f,

$$- \int_0^L f u dx.$$

The potential energy of the string:

$$V = \int_0^L \left(\frac{\tau}{2} \left(\frac{\partial u}{\partial x} \right)^2 - fu \right) dx. \qquad (2)$$

T: kinetic energy
V: potential energy

$$I(u) = \int_{t_1}^{t_2} (T - V)\, dt = \frac{1}{2} \int_{t_1}^{t_2} \int_0^L \left[\rho(x) \left(\frac{\partial u}{\partial t} \right)^2 - \tau \left(\frac{\partial u}{\partial x} \right)^2 + 2fu \right] dx\, dt \tag{3}$$

- $u(x, t)$ is differentiable with respect to x and t
- satisfies the boundary conditions (BC)

$$u(0, t) = u(L, t) = 0, \qquad t_1 \leq t \leq t_2, \tag{4}$$

- satisfies the initial conditions and end conditions,

$$\begin{aligned} u(x, t_1) &= u_1(x), \\ u(x, t_2) &= u_2(x), \end{aligned} \qquad 0 < x < L. \tag{5}$$

According to the principle of Hamilton a mechanical system behaves in a time interval $t_1 \leq t \leq t_2$ for given initial and end positions such that

$$I = \int_{t_1}^{t_2} L\,dt, \qquad L = T - V,$$

is **minimized**.
$u(x,t)$ such that $I(u) \leq I(w)$ for all w, that satisfy the initial, end, and boundary conditions.
$w = u + \varepsilon\, v$ with

$$v(0,t) = v(L,t) = 0, \qquad v(x,t_1) = v(x,t_2) = 0.$$

v is called a variation. $I(u + \varepsilon\, v)$ a function of ε.

$$I(u) \text{ minimal} \quad \Longleftrightarrow \quad \boxed{\frac{dI}{d\varepsilon}(u) = 0 \text{ for all admissible } v.}$$

Plugging $u + \varepsilon\, v$ into eq. (3), for all admissible v:

$$I(u+\varepsilon\, v) = \frac{1}{2}\int_{t_1}^{t_2}\int_{0}^{L}\left[\rho(x)\left(\frac{\partial(u+\varepsilon\, v)}{\partial t}\right)^2 - \tau\left(\frac{\partial(u+\varepsilon\, v)}{\partial x}\right)^2 + 2f(u+\varepsilon\, v)\right] d$$

$$= I(u) + \varepsilon\int_{t_1}^{t_2}\int_{0}^{L}\left[\rho(x)\frac{\partial u}{\partial t}\frac{\partial v}{\partial t} - \tau\frac{\partial u}{\partial x}\frac{\partial v}{\partial x} + 2fv\right] dx\, dt + \mathcal{O}(\varepsilon^2). \tag{6}$$

$$\frac{\partial I}{\partial \varepsilon} = \int_{t_1}^{t_2}\int_{0}^{L}\underbrace{\left[\rho\frac{\partial^2 u}{\partial t^2} - \tau\frac{\partial^2 u}{\partial x^2} + 2\, f\right]} v\, dx\, dt = 0$$

Euler-Lagrange equation

$$-\rho\frac{\partial^2 u}{\partial t^2} + \tau\frac{\partial^2 u}{\partial x^2} = 2\, f. \tag{7}$$

If the force is proportional to the displacement $u(x,t)$:

$$\boxed{\begin{gathered} -\rho(x)\frac{\partial^2 u}{\partial t^2} + \frac{\partial}{\partial x}\left(p(x)\frac{\partial u}{\partial x}\right) + q(x)u(x,t) = 0. \\ u(0,t) = u(1,t) = 0 \end{gathered}} \tag{8}$$

which is a special case of the Euler-Lagrange equation.

- $\rho(x) > 0$ mass density
- $p(x) > 0$ locally varying elasticity module.
- no initial and end conditions
- no *external* forces present in (8).

For simplicity assume that $\rho(x) = 1$.

To solve (8), we make the **ansatz**

$$u(x,t) = v(t)w(x). \tag{9}$$

With this ansatz (8) becomes

$$v''(t)w(x) - v(t)(p(x)w'(x))' - q(x)v(t)w(x) = 0. \tag{10}$$

separate the variables depending on t from those depending on x,

$$\frac{v''(t)}{v(t)} = \underbrace{\frac{1}{w(x)}(p(x)w'(x))' + q(x) = -\lambda}_{\textbf{Sturm–Liouville problem}} \quad \textit{for any } t \text{ and } x$$

$$-v''(t) = \lambda v(t) \Longleftrightarrow v(t) = a \cdot \cos(\sqrt{\lambda}t) + b \cdot \sin(\sqrt{\lambda}t), \quad \lambda > 0$$

Sturm–Liouville problem

$$-(p(x)w'(x))' + q(x)w(x) = \lambda w(x), \qquad w(0) = w(1) = 0. \tag{11}$$

- A value λ has a non-trivial solution w
- λ is called an **eigenvalue**;
- w is a corresponding **eigenfunction**.
- All eigenvalues of (11) are positive.
- (11) has infinitely many real positive eigenvalues $0 < \lambda_1 \leq \lambda_2 \leq \cdots$, $(\lambda_k \underset{k\to\infty}{\longrightarrow} \infty)$
- has a non-zero solution, $w_k(x)$, only for these particular values λ_k.

Solution of Euler-Lagrange Equation (8)

$$u(x,t) = w(x)\left[a \cdot \cos(\sqrt{\lambda}t) + b \cdot \sin(\sqrt{\lambda}t)\right]$$

$w_k(x)$ for the particular values λ_k

$$u(x,t) = \sum_{k=0}^{\infty} w_k(x)\left[a_k \cdot \cos(\sqrt{\lambda_k}\, t) + b_k \cdot \sin(\sqrt{\lambda_k}\, t)\right]. \quad (12)$$

The coefficients a_k and b_k are determined by initial and end conditions. u_0 and u_1 are given functions.

$$u(x,0) = \sum_{k=0}^{\infty} a_k w_k(x) = u_0(x),$$

$$\frac{\partial u}{\partial t}(x,0) = \sum_{k=0}^{\infty} \sqrt{\lambda_k}\, b_k w_k(x) = u_1(x),$$

- w_k form an orthogonal basis in the space of square integrable functions $L_2(0,1)$. Therefore, it is not difficult to compute the coefficients a_k and b_k.
- In concluding, we see that the difficult problem to solve is the eigenvalue problem (11). Knowing the eigenvalues and eigenfunctions the general solution of the time-dependent problem (8) is easy to form.
- Eq. (11) can be solved analytically only in very special situations, e.g., if all coefficients are constants. In general a numerical method is needed to solve the Sturm–Liouville problem (11).

Numerical methods for solving 1-dimensional problems

Three methods to solve the Sturm–Liouville problem

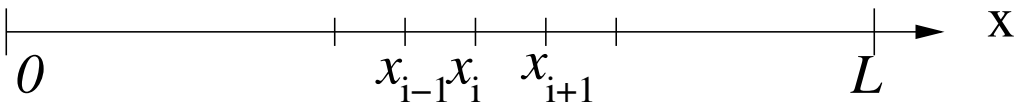
$$-(p(x)w'(x))' + q(x)w(x) = \lambda w(x)$$

with homogeneous Dirichlet boundary conditions

$$w(0) = w(1) = 0.$$

1. Finite difference method
2. The finite element method
3. Global functions

Approximate $w(x)$ by its values at the discrete points $x_i = ih$, $h = 1/(n+1)$, $i = 1, \ldots, n$.



At point x_i we approximate the derivatives by finite differences.

$$\frac{d}{dx}g(x_i) \approx \frac{g(x_{i+\frac{1}{2}}) - g(x_{i-\frac{1}{2}})}{h}.$$

For $g = p\frac{dw}{dx}$ we get

$$g(x_{i+\frac{1}{2}}) = p(x_{i+\frac{1}{2}})\frac{w(x_{i+1}) - w(x_i)}{h}$$

$$-\frac{d}{dx}\left(p\frac{dw}{dx}(x_i)\right) \approx -\frac{1}{h}\left[p(x_{i+\frac{1}{2}})\frac{w(x_{i+1})-w(x_i)}{h} - p(x_{i-\frac{1}{2}})\frac{w(x_i)-w(x_{i-1})}{h}\right]$$
$$= \frac{1}{h^2}\left[-p(x_{i-\frac{1}{2}})w_{i-1} + (p(x_{i-\frac{1}{2}}) + p(x_{i+\frac{1}{2}}))w_i - p(x_{i+\frac{1}{2}})w_{i+1}\right].$$

At the interval endpoints $w_0 = w_{n+1} = 0$.
In a matrix equation,

$$\begin{bmatrix} \frac{p(x_{\frac{1}{2}})+p(x_{\frac{3}{2}})}{h^2}+q(x_1) & -\frac{p(x_{\frac{3}{2}})}{h^2} & & \\ -\frac{p(x_{\frac{3}{2}})}{h^2} & \frac{p(x_{\frac{3}{2}})+p(x_{\frac{5}{2}})}{h^2}+q(x_2) & -\frac{p(x_{\frac{5}{2}})}{h^2} & \\ & -\frac{p(x_{\frac{5}{2}})}{h^2} & \ddots & \ddots \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_n \end{bmatrix} = \lambda \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \vdots \\ w_n \end{bmatrix},$$

$$A\mathbf{w} = \lambda\mathbf{w}. \tag{13}$$

- A is symmetric and tridiagonal.
- A is positive definite as well.
- A has just a few nonzeros: out of the n^2 elements of A only $3n-2$ are nonzero. This is a first example of a sparse matrix.

The finite element method

Find a twice differentiable function w with $w(0) = w(1) = 0$

$$\int_0^1 \left[-(p(x)w'(x))' + q(x)w(x) - \lambda w(x)\right] \phi(x)dx = 0$$

for all smooth functions ϕ that satisfy $\phi(0) = \phi(1) = 0$.

Integrate by parts and get the weak form of the problem:

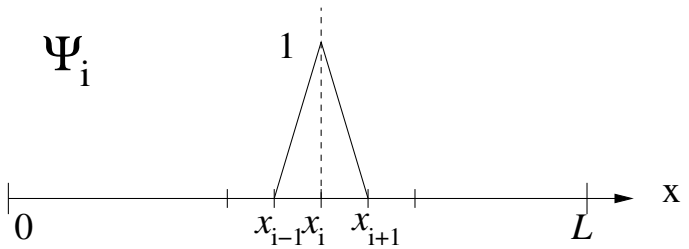
Find a differentiable function w with $w(0) = w(1) = 0$

$$\int_0^1 \left[p(x)w(x)'\phi'(x) + q(x)w(x)\phi(x) - \lambda w(x)\phi(x)\right] dx = 0 \tag{14}$$

for all differentiable functions ϕ that satisfy $\phi(0) = \phi(1) = 0$.

The finite element method (cont.)

A basis function of the finite element space: a hat function.



$$\text{The linear combination of } w(x) = \sum_{i=1}^{n} \xi_i \, \psi_i(x)$$

$$\psi_i(x) = \left(1 - \frac{|x - x_i|}{h}\right)_+ = \max\{0,\ 1 - \frac{|x - x_i|}{h}\}, \qquad (15)$$

is the function that is linear in each interval (x_i, x_{i+1}) and satisfies

$$\psi_i(x_k) = \delta_{ik} := \begin{cases} 1, & i = k, \\ 0, & i \neq k. \end{cases}$$

The finite element method (cont.)

- replace w by the linear combination $\sum \xi_i \psi_i(x)$
- replace testing 'against all ϕ' by testing against all ψ_j

Weak form becomes

$$\int_0^1 \left(-p(x)(\sum_{i=1}^{n} \xi_i \psi_i'(x))\psi_j'(x) + (q(x) - \lambda) \sum_{i=1}^{n} \xi_i \psi_i(x)\psi_j(x) \right) dx, \quad \text{for all } j,$$

$$\boxed{\sum_{i=1}^{n} \xi_i \int_0^1 \left(p(x)\psi_i'(x)\psi_j'(x) + (q(x) - \lambda)\psi_i(x)\psi_j(x)\right) dx = 0, \quad \text{for all } j.} \tag{16}$$

Rayleigh–Ritz–Galerkin equations.

Unknows: n values ξ_i and the eigenvalue λ.

The finite element method (cont.)

In matrix notation

$$A\mathbf{x} = \lambda M\mathbf{x} \tag{17}$$

$$a_{ij} = \int_0^1 \left(p(x)\psi_i'\psi_j' + q(x)\psi_i\psi_j\right)\, dx \quad \text{and} \quad m_{ij} = \int_0^1 \psi_i\psi_j\, dx$$

For **the specific case** $p(x) = 1 + x$ and $q(x) = 1$:

$$a_{kk} = \int_{(k-1)h}^{kh} \left[(1+x)\frac{1}{h^2} + \left(\frac{x-(k-1)h}{h}\right)^2\right] dx$$

$$+ \int_{kh}^{(k+1)h} \left[(1+x)\frac{1}{h^2} + \left(\frac{(k+1)h-x}{h}\right)^2\right] dx = 2(n+1+k) + \frac{2}{3}\frac{1}{n+1}$$

$$a_{k,k+1} = \int_{kh}^{(k+1)h} \left[(1+x)\frac{1}{h^2} + \frac{(k+1)h-x}{h}\cdot\frac{x-kh}{h}\right] dx = -n - \frac{3}{2} - k + \frac{1}{6}\frac{1}{n+1}$$

The finite element method (cont.)

We get:

$$M = \frac{1}{6(n+1)} \begin{bmatrix} 4 & 1 & & \\ 1 & 4 & \ddots & \\ & \ddots & \ddots & 1 \\ & & 1 & 4 \end{bmatrix}$$

A and M are symmetric tridiagonal and positive definite.

Global functions

Choose the $\psi_k(x)$ in the weak form (16) to be functions with global support

- differentiable
- satisfy the homogeneous boundary conditions

(The support of a function f is the set of arguments x for which $f(x) \neq 0$.)

$$\psi_k(x) = \sin k\pi x,$$

ψ_k are eigenfunctions of the 'nearby' problem

$$-u''(x) = \lambda u(x), \quad u(0) = u(1) = 0,$$

corresponding to the eigenvalue $\lambda_k = k^2\pi^2$.

Global functions (cont.)

The elements of matrix A are given by

$$a_{kk} = \int_0^1 \left[(1+x)k^2\pi^2 \cos^2 k\pi x + \sin^2 k\pi x\right]\, dx = \frac{3}{4}k^2\pi^2 + \frac{1}{2},$$

$$\begin{aligned} a_{kj} &= \int_0^1 \left[(1+x)kj\pi^2 \cos k\pi x \cos j\pi x + \sin k\pi x \sin j\pi x\right]\, dx \\ &= \frac{kj(k^2+j^2)((-1)^{k+j}-1)}{(k^2-j^2)^2}, \quad k \neq j. \end{aligned}$$

A numerical comparison: 1D eigenvalue problem

$$-((1+x)w'(x))' + w(x) = \lambda w(x)$$

$$w(0) = w(1) = 0$$

solve it with 3 different methods.

1. Finite differences
2. The finite element method
3. Global functions

Numerical solutions of problem

Finite difference method				
k	$\lambda_k(n=10)$	$\lambda_k(n=20)$	$\lambda_k(n=40)$	$\lambda_k(n=80)$
1	15.245	15.312	15.331	15.336
2	56.918	58.048	58.367	58.451
3	122.489	128.181	129.804	130.236
4	206.419	224.091	229.211	230.580
5	301.499	343.555	355.986	359.327
6	399.367	483.791	509.358	516.276
7	492.026	641.501	688.398	701.185
8	578.707	812.933	892.016	913.767
9	672.960	993.925	1118.969	1153.691
10	794.370	1179.947	1367.869	1420.585

Numerical solutions of problem (cont.)

Finite element method				
k	$\lambda_k(n=10)$	$\lambda_k(n=20)$	$\lambda_k(n=40)$	$\lambda_k(n=80)$
1	15.447	15.367	15.345	15.340
2	60.140	58.932	58.599	58.511
3	138.788	132.657	130.979	130.537
4	257.814	238.236	232.923	231.531
5	426.223	378.080	365.047	361.648
6	654.377	555.340	528.148	521.091
7	949.544	773.918	723.207	710.105
8	1305.720	1038.433	951.392	928.983
9	1702.024	1354.106	1214.066	1178.064
10	2180.159	1726.473	1512.784	1457.733

Numerical solutions of problem (cont.)

Global function method				
k	$\lambda_k(n=10)$	$\lambda_k(n=20)$	$\lambda_k(n=40)$	$\lambda_k(n=80)$
1	15.338	15.338	15.338	15.338
2	58.482	58.480	58.480	58.480
3	130.389	130.386	130.386	130.386
4	231.065	231.054	231.053	231.053
5	360.511	360.484	360.483	360.483
6	518.804	518.676	518.674	518.674
7	706.134	705.631	705.628	705.628
8	924.960	921.351	921.344	921.344
9	1186.674	1165.832	1165.823	1165.822
10	1577.340	1439.083	1439.063	1439.063

Numerical solutions of problem (cont.)

- The global function method is the most powerful of them all. The convergence rate is exponential.
- With the finite difference and finite element methods the eigenvalues exhibit quadratic convergence rates. If the mesh width h is reduced by a factor of $q = 2$, the error in the eigenvalues is reduced by the factor $q^2 = 4$.

  (Note thate there are higher order finite difference and finite element methods that give rise to higher convergence rates.)

Example 2: The heat equation

$u(\mathbf{x}, t)$: The instationary temperature distribution in an insulated container

$$\begin{aligned} \frac{\partial u(\mathbf{x}, t)}{\partial t} - \Delta u(\mathbf{x}, t) &= 0, && \mathbf{x} \in \Omega, \ t > 0, \\ \frac{\partial u(\mathbf{x}, t)}{\partial n} &= 0, && \mathbf{x} \in \partial\Omega, \ t > 0, \\ u(\mathbf{x}, 0) &= u_0(\mathbf{x}), && \mathbf{x} \in \Omega. \end{aligned} \tag{18}$$

Ω is a 3-dimensional domain with boundary $\partial\Omega$.
$\frac{\partial u}{\partial n}$: the derivative of u in direction of the outer normal vector $\mathbf{n}$
$u_0(\mathbf{x}), \mathbf{x} = (x_1, x_2, x_3)^T \in \mathbb{R}^3$, is a given bounded, sufficiently smooth function.

Laplace operator: $$\Delta u = \sum \frac{\partial^2 u}{\partial {x_i}^2}$$

Method of separation of variables:

$$u(\mathbf{x}, t) = v(t)w(\mathbf{x}).$$

If a constant λ can be found such that

$$\boxed{\begin{aligned} \Delta w(\mathbf{x}) + \lambda w(\mathbf{x}) = 0, \quad w(\mathbf{x}) \neq 0, \quad & \mathbf{x} \text{ in } \Omega, \\ \frac{\partial w(\mathbf{x}, t)}{\partial n} = 0, \quad & \mathbf{x} \text{ on } \partial\Omega, \end{aligned}} \tag{19}$$

the product $u = vw$ is a solution if and only if

$$\frac{dv(t)}{dt} + \lambda v(t) = 0,$$

If λ_n is an eigenvalue with corresponding eigenfunction w_n, then

$$e^{-\lambda_n t} w_n(\mathbf{x})$$

is a solution of the first two equations in (18).
Infinitely many real eigenvalues $0 \leq \lambda_1 \leq \lambda_2 \leq \cdots, (\lambda_n \underset{t\to\infty}{\longrightarrow} \infty)$.
An arbitrary bounded piecewise continuous function can be represented as a linear combination of the eigenfunctions $w_1, w_2, \ldots$.
The solution

$$u(\mathbf{x}, t) = \sum_{n=1}^{\infty} c_n e^{-\lambda_n t} w_n(\mathbf{x}),$$

where the coefficients c_n are determined by the initial conditions

$$u_0(\mathbf{x}) = \sum_{n=1}^{\infty} c_n w_n(\mathbf{x}).$$

The smallest eigenvalue is $\lambda_1 = 0$ with $w_1 = 1$ and $\lambda_2 > 0$. We can see that

$$u(\mathbf{x}, t) \underset{t\to\infty}{\longrightarrow} c_1.$$

The convergence rate towards this equilibrium is determined by the smallest *positive* eigenvalue λ_2 of (19):

$$\begin{aligned}\|u(\mathbf{x}, t) - c_1\| &= \|\sum_{n=2}^{\infty} c_n e^{-\lambda_n t} w_n(\mathbf{x})\| \leq \sum_{n=2}^{\infty} |e^{-\lambda_n t}| \|c_n w_n(\mathbf{x})\| \\ &\leq e^{-\lambda_2 t} \sum_{n=2}^{\infty} \|c_n w_n(\mathbf{x})\| \leq e^{-\lambda_2 t} \|u_0(\mathbf{x})\|.\end{aligned}$$

Note: we have assumed that the value of the constant function $w_1(\mathbf{x})$ is set to unity.

Example 3: The wave equation

$u(\mathbf{x}, t)$: air pressure in a volume with acoustically "hard" walls

$$\frac{\partial^2 u(\mathbf{x}, t)}{\partial t^2} - \Delta u(\mathbf{x}, t) = 0, \qquad \mathbf{x} \in \Omega,\ t > 0, \tag{20}$$

$$\frac{\partial u(\mathbf{x}, t)}{\partial n} = 0, \qquad \mathbf{x} \in \partial\Omega,\ t > 0, \tag{21}$$

$$u(\mathbf{x}, 0) = u_0(\mathbf{x}), \qquad \mathbf{x} \in \Omega, \tag{22}$$

$$\frac{\partial u(\mathbf{x}, 0)}{\partial t} = u_1(\mathbf{x}), \qquad \mathbf{x} \in \Omega. \tag{23}$$

Sound propagates with speed $-\nabla u$, along the (negative) gradient from high to low pressure.

Example 3: The wave equation (cont.)

Separation of variables leads again to equation (19) but now together with

$$\frac{d^2v(t)}{dt^2} + \lambda v(t) = 0. \tag{24}$$

The general solution of the wave equation has the form

$$u(x,t) = \sum_{k=0}^{\infty} w_k(x) \left[a_k \cdot \cos(\sqrt{\lambda_k}\, t) + b_k \cdot \sin(\sqrt{\lambda_k}\, t) \right]. \tag{12}$$

where the w_k, $k = 1, 2, \ldots$, are the eigenfunctions of the eigenvalue problem (19). The coefficients a_k and b_k are determined by (22) and (23).

Inhomogeneous problem

If a harmonic oscillation is forced on the system, an *inhomogeneous* problem is obtained,

$$\frac{\partial^2 u(\mathbf{x},t)}{\partial t^2} - \Delta u(\mathbf{x},t) = f(\mathbf{x},t). \tag{25}$$

The boundary and initial conditions are taken from (20)–(23). This problem can be solved by setting

$$\begin{aligned} u(\mathbf{x},t) &:= \sum_{n=1}^{\infty} \tilde{v}_n(t) w_n(\mathbf{x}), \\ f(\mathbf{x},t) &:= \sum_{n=1}^{\infty} \phi_n(t) w_n(\mathbf{x}). \end{aligned} \tag{26}$$

Inhomogeneous problem (cont.)

$\Longrightarrow$ $\tilde{v}_n$ has to satisfy equation

$$\frac{d^2\tilde{v}_n}{dt^2} + \lambda_n \tilde{v}_n = \phi_n(t). \tag{27}$$

If $\phi_n(t) = a \sin \omega t$, then the solution becomes

$$\tilde{v}_n = A_n \cos \sqrt{\lambda_n} t + B_n \sin \sqrt{\lambda_n} t + \frac{1}{\lambda_n - \omega^2} a \sin \omega t. \tag{28}$$

A_n and B_n are real constants determined by the initial conditions.

- If ω gets close to $\sqrt{\lambda_n}$, then the last term can be very large.
- If $\omega = \sqrt{\lambda_n}$, $\tilde{v}_n$ gets the form

$$\tilde{v}_n = A_n \cos \sqrt{\lambda_n} t + B_n \sin \sqrt{\lambda_n} t + at \sin \omega t. \tag{29}$$

Inhomogeneous problem (cont.)

$\tilde{v}_n$ is not bounded in time $\Longrightarrow$ is called resonance.

Remark: Vibrating membranes satisfy the wave equation. If the membrane (of a drum) is fixed at its boundary, the condition $u(\mathbf{x}, t) = 0$ is called Dirichlet boundary conditions.

Boundary Conditions:

$$u(\mathbf{x}, t) = g_D(\mathbf{x}), \quad \Rightarrow \text{ Dirichlet boundary conditions}$$

$$\frac{\partial u(\mathbf{x}, t)}{\partial n} = g_N(\mathbf{x}), \quad \Rightarrow \text{ Neumann boundary conditions}$$

$$\alpha u + \beta \frac{\partial u}{\partial n} = g, \quad \Rightarrow \text{ Mixed or Cauchy or Robin boundary conditions}$$

The 2D Laplace eigenvalue problem

$$-\Delta u(\mathbf{x}) = \lambda u(\mathbf{x}), \qquad \mathbf{x} \in \Omega, \tag{30}$$

with the more general boundary conditions

$$u(\mathbf{x}) = 0, \qquad \mathbf{x} \in C_1 \subset \partial\Omega, \tag{31}$$

$$\frac{\partial u}{\partial n}(\mathbf{x}) + \alpha(\mathbf{x})u(\mathbf{x}) = 0, \qquad \mathbf{x} \in C_2 \subset \partial\Omega. \tag{32}$$

C_1 and C_2 are disjoint subsets of $\partial\Omega$ with $C_1 \cup C_2 = \partial\Omega$.
In general not possible to solve exactly $\rightarrow$ numerical approx.
Two methods for the discretization of eigenvalue problems:

- Finite Difference Method
- Finite Element Method

The finite difference method

For simplicity, assume that the domain Ω is a square with sides of length 1: $\Omega = (0,1) \times (0,1)$. The eigenvalue problem

$$\begin{aligned} -\Delta u(x,y) &= \lambda u(x,y), && 0 < x, y < 1 \\ u(0,y) = u(1,y) &= u(x,0) = 0, && 0 < x, y < 1, \\ \frac{\partial u}{\partial n}(x,1) &= 0, && 0 < x < 1. \end{aligned} \tag{33}$$

This eigenvalue problem

- occurs in the computation of eigenfrequencies and eigenmodes of a homogeneous quadratic membrane with three fixed and one free side.
- can be solved analytically by separation of the two spatial variables x and y.

The finite difference method (cont.)

The eigenvalues are

$$\lambda_{k,l} = \left(k^2 + \frac{(2l-1)^2}{4} \right) \pi^2, \quad k, l \in \mathbb{N},$$

and the corresponding eigenfunctions are

$$u_{k,l}(x, y) = \sin k\pi x \sin \frac{2l-1}{2} \pi y.$$

Define a rectangular grid with grid points $(x_i, y_j), 0 \leq i, j \leq N$. The coordinates of the grid points are

$$(x_i, y_j) = (ih, jh), \qquad h = 1/N.$$

The finite difference method (cont.)

By a Taylor expansion, for sufficiently smooth functions u

$$-\Delta u(x,y) = \frac{1}{h^2}(4u(x,y) - u(x-h,y) - u(x+h,y) - u(x,y-h) - u(x,y+h)) + O(h^2)$$

At the interior grid points

$$4u_{i,j} - u_{i-1,j} - u_{i+1,j} - u_{i,j-1} - u_{i,j+1} = \lambda h^2 u_{i,j}, \quad 0 < i,j < N. \quad (34)$$

$$u_{i,j} \approx u(x_i, x_j)$$

The Dirichlet boundary conditions are replaced by the equations

$$u_{i,0} = u_{i,N} = u_{0,i}, \qquad 0 < i < N. \quad (35)$$

The finite difference method (cont.)

At the points at the upper boundary of Ω:

$$4u_{i,N} - u_{i-1,N} - u_{i+1,N} - u_{i,N-1} - u_{i,N+1} = \lambda h^2 u_{i,N}, \quad 0 \leq i \leq N. \tag{36}$$

$u_{i,N+1}$: a grid point outside of the domain
The Neumann boundary conditions suggest to reflect the domain at the upper boundary and to extend the eigenfunction symmetrically beyond the boundary. $u_{i,N+1} = u_{i,N-1}$.
Plugging it and multiply the new equation by the factor $1/2$ gives

$$2u_{i,N} - \frac{1}{2}u_{i-1,N} - \frac{1}{2}u_{i+1,N} - u_{i,N-1} = \frac{1}{2}\lambda h^2 u_{i,N}, \quad 0 < i < N. \tag{37}$$

The finite difference method (cont.)

The matrix equation

$$
\begin{pmatrix}
4 & -1 & 0 & -1 & & & & & & & & \\
-1 & 4 & -1 & 0 & -1 & & & & & & & \\
0 & -1 & 4 & 0 & 0 & -1 & & & & & & \\
-1 & 0 & 0 & 4 & -1 & 0 & -1 & & & & & \\
 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & & & & \\
 & & -1 & 0 & -1 & 4 & 0 & 0 & -1 & & & \\
 & & & -1 & 0 & 0 & 4 & -1 & 0 & -1 & & \\
 & & & & -1 & 0 & -1 & 4 & -1 & 0 & -1 & \\
 & & & & & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\
 & & & & & & -1 & 0 & 0 & 2 & $-\frac{1}{2}$ & 0 \\
 & & & & & & & -1 & 0 & $-\frac{1}{2}$ & 2 & $-\frac{1}{2}$ \\
 & & & & & & & & -1 & 0 & $-\frac{1}{2}$ & 2
\end{pmatrix}
\begin{pmatrix} u_{1,1} \\ u_{1,2} \\ u_{1,3} \\ u_{2,1} \\ u_{2,2} \\ u_{2,3} \\ u_{3,1} \\ u_{3,2} \\ u_{3,3} \\ u_{4,1} \\ u_{4,2} \\ u_{4,3} \end{pmatrix}
$$

$$
= \lambda h^2
\begin{pmatrix}
1 & & & & & & & & & & & \\
 & 1 & & & & & & & & & & \\
 & & 1 & & & & & & & & & \\
 & & & 1 & & & & & & & & \\
 & & & & 1 & & & & & & & \\
 & & & & & 1 & & & & & & \\
 & & & & & & 1 & & & & & \\
 & & & & & & & 1 & & & & \\
 & & & & & & & & 1 & & & \\
 & & & & & & & & & $\frac{1}{2}$ & & \\
 & & & & & & & & & & $\frac{1}{2}$ & \\
 & & & & & & & & & & & $\frac{1}{2}$
\end{pmatrix}
\begin{pmatrix} u_{1,1} \\ u_{1,2} \\ u_{1,3} \\ u_{2,1} \\ u_{2,2} \\ u_{2,3} \\ u_{3,1} \\ u_{3,2} \\ u_{3,3} \\ u_{4,1} \\ u_{4,2} \\ u_{4,3} \end{pmatrix}.
\tag{38}
$$

Matrix Eigenvalue Problem

For arbitrary $N > 1$,

$$\mathbf{u}_i := \begin{pmatrix} u_{i,1} \\ u_{i,2} \\ \vdots \\ u_{i,N-1} \end{pmatrix} \in \mathbb{R}^{N-1},$$

$$T := \begin{pmatrix} 4 & -1 & & \\ -1 & 4 & \ddots & \\ & \ddots & \ddots & -1 \\ & & -1 & 4 \end{pmatrix} \in \mathbb{R}^{(N-1)\times(N-1)},$$

$$I := \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix} \in \mathbb{R}^{(N-1)\times(N-1)}.$$

Matrix Eigenvalue Problem (cont.)

The discrete eigenvalue problem of size $N \times (N-1)$.

$$\begin{pmatrix} T & -I & & \\ -I & T & \ddots & \\ & \ddots & \ddots & -I \\ & & -I & \frac{1}{2}T \end{pmatrix} \begin{pmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_3 \\ \mathbf{u}_4 \end{pmatrix} = \lambda h^2 \begin{pmatrix} I & & & \\ & \ddots & & \\ & & I & \\ & & & \frac{1}{2}I \end{pmatrix} \begin{pmatrix} \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_{N-1} \\ \mathbf{u}_N \end{pmatrix}$$

Matrix eigenvalue problem:

$$\underbrace{A}_{\textbf{symmetric}} \mathbf{x} = \lambda \underbrace{M}_{\textbf{SPD}} \mathbf{x},$$

SPD: Symmetric Positive Definite

Matrix Eigenvalue Problem (cont.)

M is identity matrix $\Rightarrow$ special (generalized) eigenvalue problem.

Special (symmetric) eigenvalue problem: (39) left multiplication by

$$\begin{pmatrix} I & & & \\ & I & & \\ & & I & \\ & & & \sqrt{2}I \end{pmatrix}$$

$$\begin{pmatrix} T & -I & & \\ -I & T & -I & \\ & -I & T & -\sqrt{2}I \\ & & -\sqrt{2}I & T \end{pmatrix} \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \\ \frac{1}{\sqrt{2}}\mathbf{u}_4 \end{pmatrix} = \lambda h^2 \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \\ \frac{1}{\sqrt{2}}\mathbf{u}_4 \end{pmatrix}.$$

A property common to matrices obtained by the finite difference method are its **sparsity**.

(Ir)regular domains

- If the shapes of the domains get complicated
- If the boundary is not aligned with the coordinate axes

Finite Difference Method can be difficult to implement

$$\Downarrow$$

Finite Element Methods

The finite element method (FEM)

$(\lambda, u) \in \mathbb{R} \times V$ an eigenpair of 2D Laplace eigenvalue problem

$$\int_\Omega (\Delta u + \lambda u) v \, dx \, dy = 0, \quad \forall v \in V, \tag{39}$$

where V is vector space of bounded twice differentiable functions that satisfy the boundary conditions (31)–(32). By partial integration (Green's formula) this becomes

$$\int_\Omega \nabla u \nabla v \, dx \, dy + \int_{\partial\Omega} \alpha \, u \, v \, ds = \lambda \int_\Omega u \, v \, dx \, dy, \quad \forall v \in V, \tag{40}$$

or

$$a(u, v) = (u, v), \qquad \forall v \in V \tag{41}$$

The finite element method (FEM) (cont.)

where

$$a(u,v) = \int_\Omega \nabla u \, \nabla v \, dx \, dy + \int_{\partial\Omega} \alpha \, u \, v \, ds, \quad \text{and} \quad (u,v) = \int_\Omega u \, v \, dx \, dy.$$

We complete the space V with respect to the Sobolev norm

$$\sqrt{\int_\Omega (u^2 + |\nabla u|^2) \, dx \, dy}$$

to become a Hilbert space H. H is the space of quadratic integrable functions with quadratic integrable first derivatives that satisfy the Dirichlet boundary conditions (31)

$$u(x,y) = 0, \qquad (x,y) \in C_1.$$

The finite element method (FEM) (cont.)

(Functions in H in general do not satisfy the so-called natural boundary conditions (32).) One can show that the eigenvalue problem (30)–(32) is equivalent with the eigenvalue problem

$$\begin{aligned} &\text{Find } (\lambda, u) \in \mathbb{R} \times H \text{ such that} \\ &a(u, v) = \lambda(u, v) \quad \forall v \in H. \end{aligned} \tag{42}$$

(The essential point is to show that the eigenfunctions of (42) are elements of V.)

The Rayleigh–Ritz–Galerkin method

A set of linearly independent functions

$$\phi_1(x, y), \cdots, \phi_n(x, y) \in H, \tag{43}$$

These functions span a subspace S of H.
The problem (42) is solved where H is replaced by S.

$$\begin{aligned} &\text{Find } (\lambda, u) \in \mathbb{R} \times S \text{ such that} \\ &a(u, v) = \lambda(u, v) \quad \forall v \in S. \end{aligned} \tag{44}$$

With the Ritz ansatz

$$u = \sum_{i=1}^{n} x_i \phi_i, \tag{45}$$

The Rayleigh–Ritz–Galerkin method (cont.)

equation (44) becomes

$$\begin{aligned} &\text{Find } (\lambda, \mathbf{x}) \in \mathbb{R} \times \mathbb{R}^n \text{ such that} \\ &\sum_{i=1}^{n} x_i a(\phi_i, v) = \lambda \sum_{i=1}^{n} x_i (\phi_i, v), \quad \forall v \in S. \end{aligned} \tag{46}$$

Eq. (46) must hold for all $v \in S$, in particular for $v = \phi_1, \cdots, \phi_n$. But since the $\phi_i, 1 \leq i \leq n$, form a basis of S, equation (46) is equivalent with

$$\sum_{i=1}^{n} x_i a(\phi_i, \phi_j) = \lambda \sum_{i=1}^{n} x_i (\phi_i, \phi_j), \quad 1 \leq j \leq n. \tag{47}$$

The Rayleigh–Ritz–Galerkin method (cont.)

This is a matrix eigenvalue problem of the form

$$A\mathbf{x} = \lambda M\mathbf{x} \tag{48}$$

where

$$\mathbf{x} = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}, \quad A = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix}, \quad M = \begin{pmatrix} m_{11} & \cdots & m_{1n} \\ \vdots & \ddots & \vdots \\ m_{n1} & \cdots & m_{nn} \end{pmatrix} \tag{49}$$

with the stiffness matrix

$$a_{ij} = a(\phi_i, \phi_j) = \int_\Omega \nabla\phi_i \, \nabla\phi_j \, dx \, dy + \int_{\partial\Omega} \alpha \, \phi_i \, \phi_j \, ds$$

The Rayleigh–Ritz–Galerkin method (cont.)

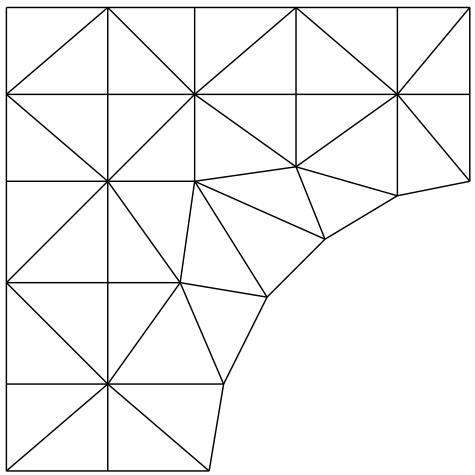
and the mass matrix

$$m_{ij} = (\phi_i, \phi_j) = \int_\Omega \phi_i \, \phi_j \, dx \, dy.$$

The finite element method (FEM) is a special case of the Rayleigh–Ritz method. In the FEM the subspace S and in particular the basis $\{\phi_i\}$ is chosen in a particularly clever way. For simplicity we assume that the domain Ω is a simply connected domain with a polygonal boundary, cf. Fig 63. (This means that the boundary is composed entirely of straight line segments.)

Triangulation

Triangulation of a domain Ω



Triangulation (cont.)

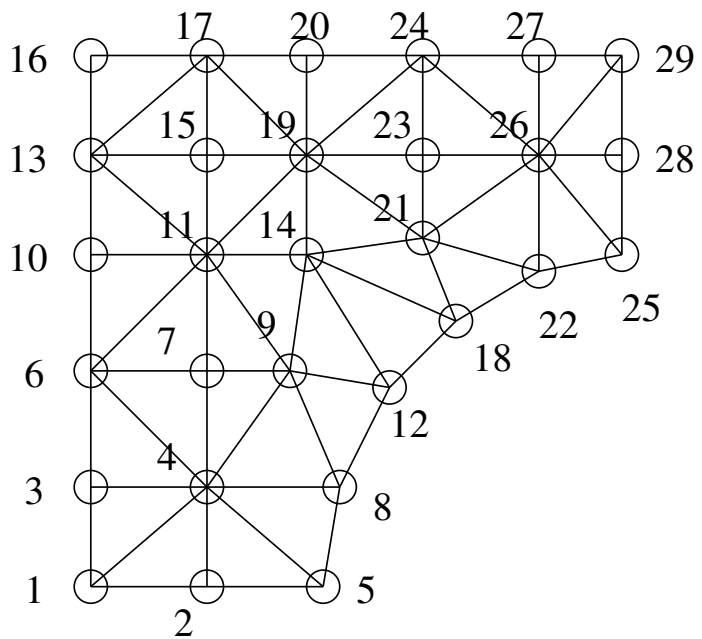
This domain is partitioned into triangular subdomains $T_1, \cdots, T_N$, so-called elements, such that

$$T_i \cap T_j = \varnothing \quad \text{for all } i \neq j, \quad \text{and} \quad \bigcup_e \overline{T_e} = \overline{\Omega}. \tag{50}$$

Finite element spaces for solving (30)–(32) are typically composed of functions that are continuous in Ω and are polynomials on the individual subdomains T_e. Such functions are called piecewise polynomials. Notice that this construction provides a subspace of the Hilbert space H but not of V, i.e., the functions in the finite element space are not very smooth and the natural boundary conditions are not satisfied.

Basis functions

The selection of the basis of the finite element space S. $S_1 \subset H$ is the space of continuous piecewise linear polynomials.



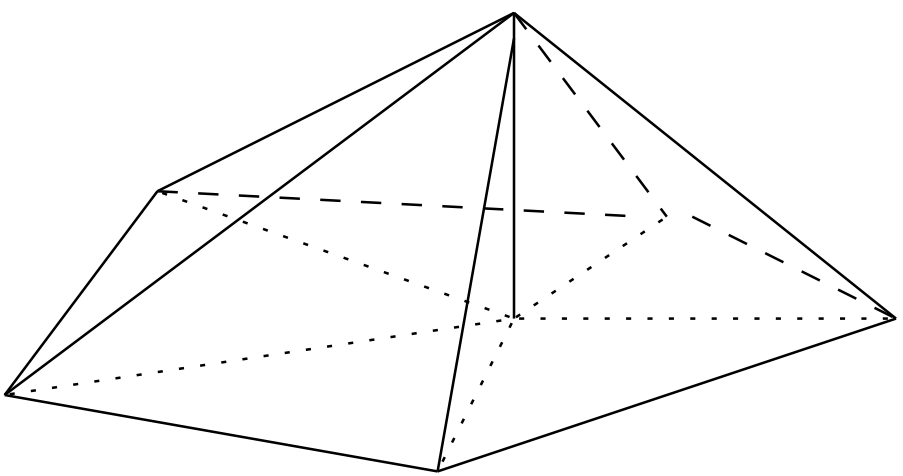
- Nodes, except those on the boundary portion C_1, are numbered from 1 to n.
- The coordinates of the i-th node be (x_i, y_i).

Basis functions (cont.)

$\phi_i(x, y) \in S_1$ is defined by

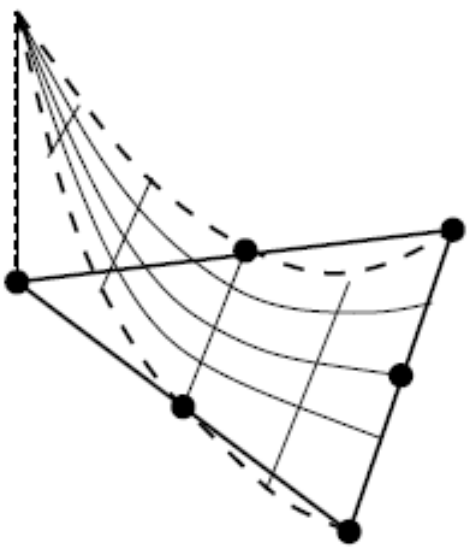
$$\phi_i((x_j, y_j)) := \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \tag{51}$$

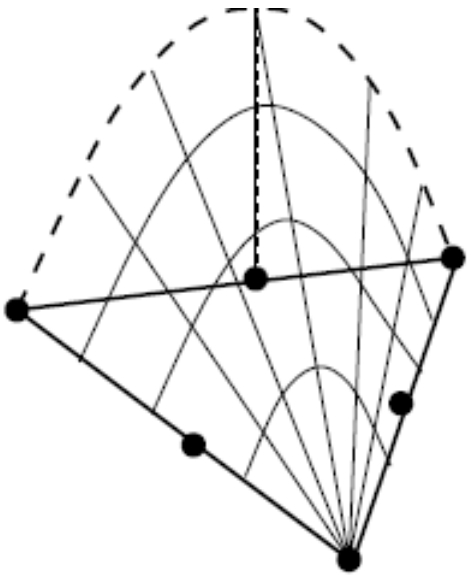
A typical basis function ϕ_i:
A piecewise linear basis function (or hat function)



Basis functions (cont.)

Another often used finite element element space is $S_2 \subset H$, the space of continuous, piecewise quadratic polynomials. These functions are (or can be) uniquely determined by their values at the vertices and edge midpoints of the triangle.





Basis functions (cont.)

One immediately sees that for most $i \neq j$

$$a(\phi_i, \phi_j) = 0, \quad (\phi_i, \phi_j) = 0. \tag{52}$$

The matrices A and M in (48) will be **sparse**. The matrix M is positive definite as

$$\mathbf{x}^T M \mathbf{x} = \sum_{i,j=1}^{N} x_i x_j m_{ij} = \sum_{i,j=1}^{N} x_i x_j (\phi_i, \phi_j) = (u, u) > 0,$$

$$u = \sum_{i=1}^{N} x_i \phi_i \neq 0,$$

(The ϕ_i are linearly independent and $||u|| = \sqrt{(u, u)}$ is a norm.)

Basis functions (cont.)

Similarly it is shown that

$$\mathbf{x}^T A\mathbf{x} \geq 0.$$

It is possible to have $\mathbf{x}^\mathbf{T}\mathbf{A}\mathbf{x} = 0$ for a nonzero vector $\mathbf{x}$. This is the case if the constant function $u = 1$ is contained in S. This happens if Neumann boundary conditions $\frac{\partial u}{\partial n} = 0$ are posed on the whole boundary $\partial\Omega$. Then,

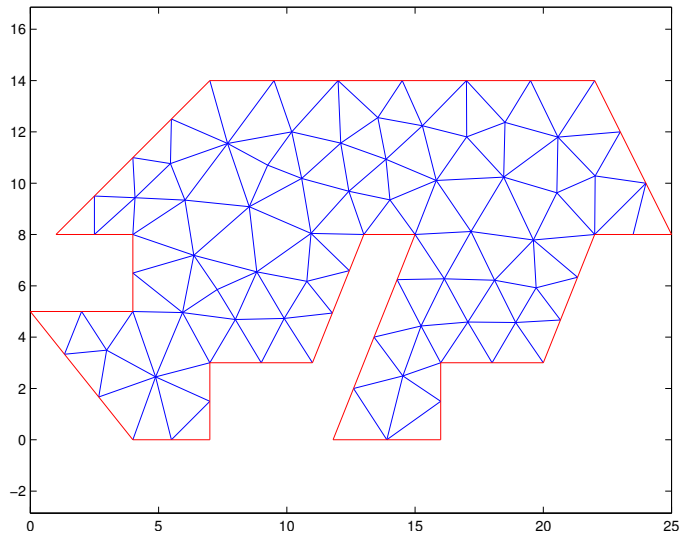
$$u(x, y) = 1 = \sum_i \phi_i(x, y),$$

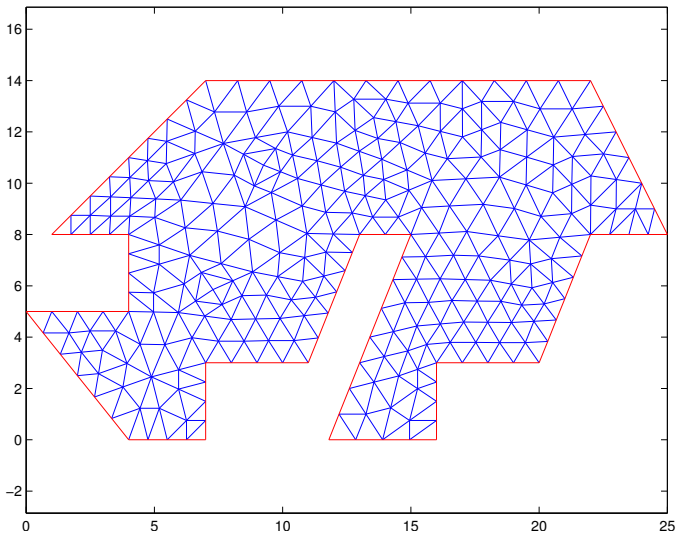
i.e., we have $\mathbf{x}^\mathbf{T}\mathbf{A}\mathbf{x} = 0$ for $\mathbf{x} = [1, 1, \ldots, 1]$.

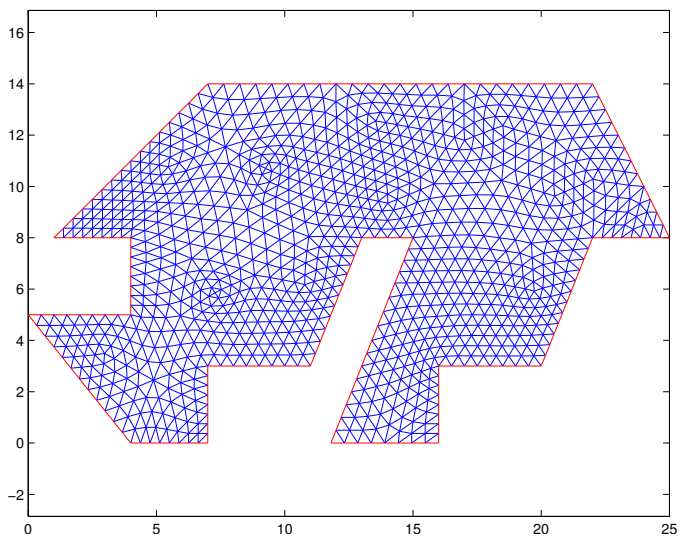
A numerical example: acoustic vibration problem

- Determine the acoustic eigenfrequencies and corresponding modes in the interior of a car.
- Interest in the manufacturing of cars, since an appropriate shape of the form of the interior can suppress the often unpleasant droning of the motor.
- The problem is 3D, but by separation of variables the problem can be reduced to 2D.
- If rigid, acoustically hard walls are assumed, the mathematical model of the problem is again the Laplace eigenvalue problem (19) together with Neumann boundary conditions. The domain is given in Fig. 70 where three finite element triangulations are shown with 87 (grid_1), 298 (grid_2), and 1095 (grid_3) vertices (nodes), respectively.

A numerical example: acoustic vibration problem (cont.)





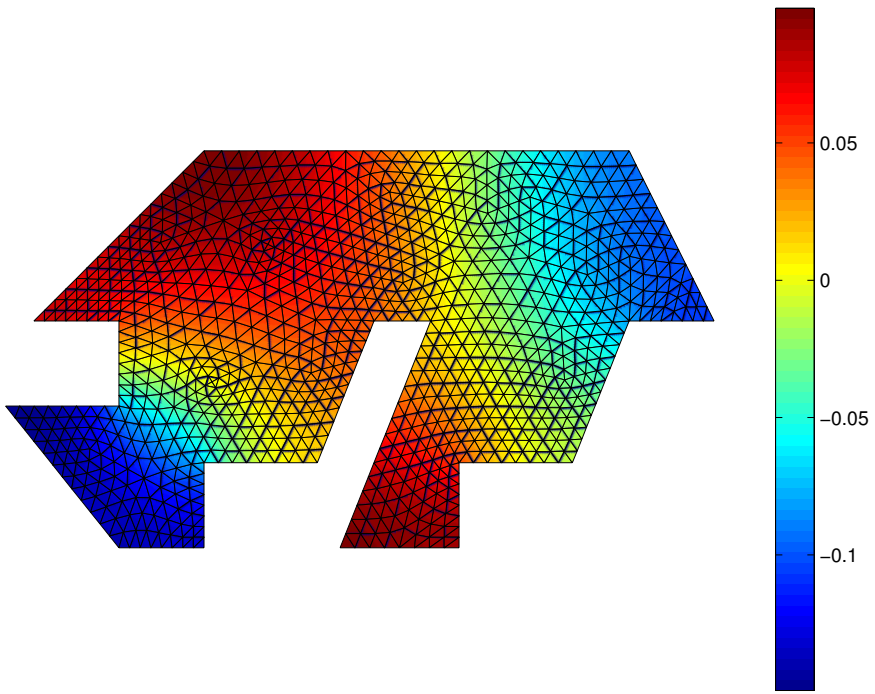


Numerical solutions of acoustic vibration problem

- the quadratic convergence rate
- The smallest eigenvalue is always zero.
- The corresponding eigenfunction is the constant function.

Finite element method			
k	$\lambda_k(\text{grid}_1)$	$\lambda_k(\text{grid}_2)$	$\lambda_k(\text{grid}_3)$
1	0.0000	-0.0000	0.0000
2	0.0133	0.0129	0.0127
3	0.0471	0.0451	0.0444
4	0.0603	0.0576	0.0566
5	0.1229	0.1182	0.1166
6	0.1482	0.1402	0.1376
7	0.1569	0.1462	0.1427
8	0.2162	0.2044	0.2010
9	0.2984	0.2787	0.2726
10	0.3255	0.2998	0.2927

Fourth eigenmode of the acoustic vibration problem



The difference of the pressure at a given location to the normal pressure. Large amplitudes means that the corresponding noise is very well noticable.

Spectral clustering

Goal: group a given set of data points $\mathbf{x}_1, \ldots, \mathbf{x}_n$ into k clusters such that members from the same cluster are (in some sense) close to each other and members from different clusters are (in some sense) well separated from each other.

A popular approach to clustering $\Longrightarrow$ similarity graphs.

$s(\mathbf{x}_i, \mathbf{x}_j) \geq 0$ between pairs of data points $\mathbf{x}_i$ and $\mathbf{x}_j$.

An undirected graph $G = (V, E) : V = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$.

Two vertices $\mathbf{x}_i, \mathbf{x}_j$ are connected by an edge if the similarity s_{ij} between $\mathbf{x}_i$ and $\mathbf{x}_j$ is sufficiently large.

A weight $w_{ij} > 0$ is assigned to the edge, depending on s_{ij}.

Spectral clustering (cont.)

If two vertices are not connected we set $w_{ij} = 0$. The weights are collected into a weighted adjacency matrix

$$W = (w_{ij})_{i,j=1}^{n} .$$

fully connected graph $w_{ij} = s(\mathbf{x}_i, \mathbf{x}_j)$.
Usually, this will only result in reasonable clusters if the similarity function models locality very well, e.g., $s(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}\right)$

k-nearest neighbors $\mathbf{x}_i, \mathbf{x}_j$ are connected if $\mathbf{x}_i$ is among the k-nearest neighbors of $\mathbf{x}_j$ or if $\mathbf{x}_j$ is among the k-nearest neighbors of $\mathbf{x}_i$ (then use $w_{ij} = s(\mathbf{x}_i, \mathbf{x}_j)$).

ϵ-neighbors $\mathbf{x}_i, \mathbf{x}_j$ are connected if their pairwise distance is smaller than ε for some $\varepsilon > 0$. Then, e.g., $w_{ij} = 1$.

Graph Laplacian

Let W be symmetric. Degree of vertex $\mathbf{x}_i$: $d_i = \sum_{j=1}^{n} w_{ij}$.

Let $D = \mathrm{diag}(d_1, d_2, \ldots, d_n)$. Then the graph Laplacian is defined as

$$L = D - W.$$

- The graph Laplacian has at least one zero eigenvalue.
- There is one zero eigenvalue per disconnected component of the graph.
  Eigenvectors = indicator vectors χ_{V_i}, of the components.
- Do not use the zero eigenvalues to determine the (number of) connected components.

Spectral clustering

We cannot simply compute the eigenvectors corresponding to the zero eigenvalues because

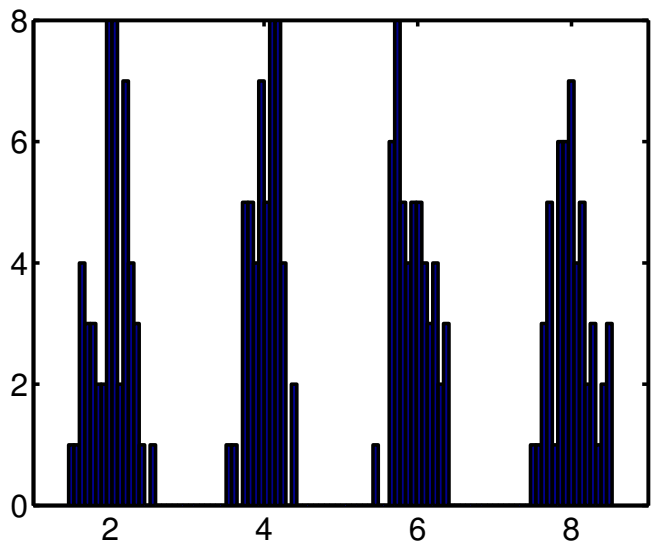
1. The eigenvectors would be mixed up.
   An eigensolver would give us $U = (\mathbf{v}_1, \ldots, \mathbf{v}_k)\, Q$
2. Don't want to compute disconnected components anyway.

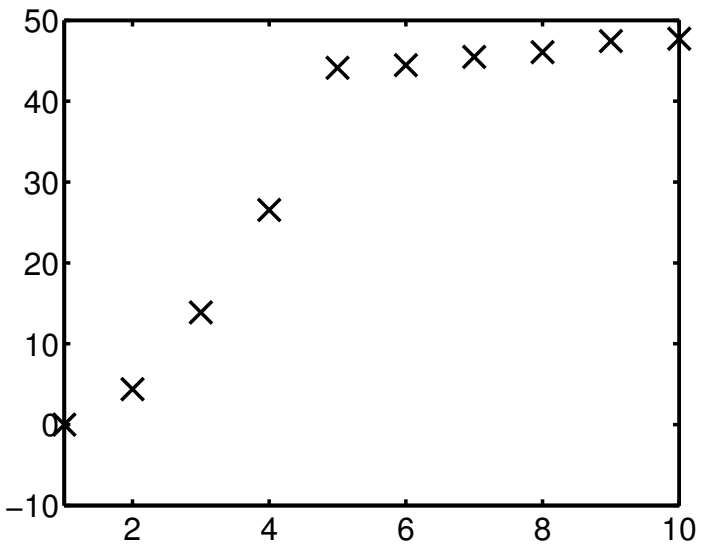
To find clusters we compute an eigenbasis belonging to the k smallest eigenvalues.

Spectral clustering (cont.)

Example: `m = 50; randn('state',0);`

`x = [2+randn(m,1)/4;4+randn(m,1)/4;6+randn(m,1)/4;8+randn(m`

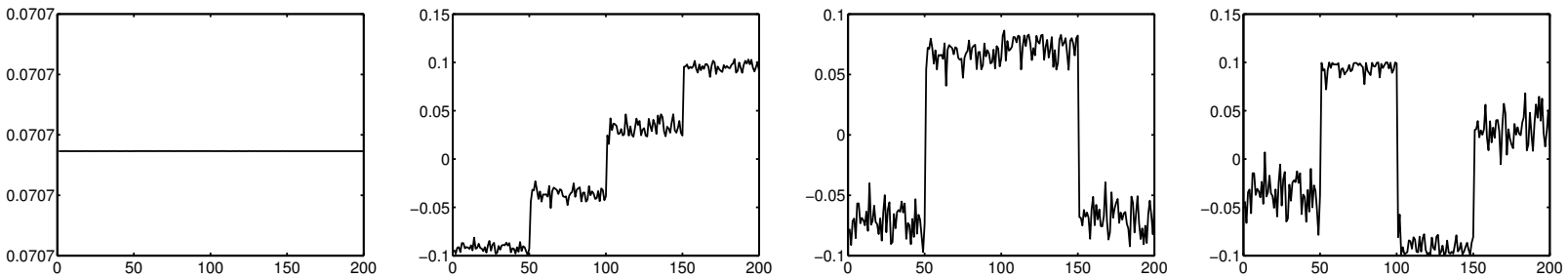




Histogram of the distribution of the entries of $\mathbf{x}$ and the eigenvalues of the graph Laplacian for the fully connected similarity graph with similarity function $s(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-|\mathbf{x}_i - \mathbf{x}_j|^2/2\right)$

Spectral clustering (cont.)

Eigenvectors of the graph Laplacian (4 smallest eigenvalues)



To solve the issue that the eigenbasis may be transformed by an arbitrary orthogonal matrix, we "transpose" the basis and consider the row vectors of U:

$U^T = \big(\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n\big), \quad \mathbf{u}_i \in \mathbb{R}^k.$

If U contained indicator vectors then each of the short vectors $\mathbf{u}_i$ would be a unit vector $\mathbf{e}_j$ for some $1 \leq j \leq k$.

Spectral clustering (cont.)

Now apply *k*-means clustering:

1. Compute cluster centers $\mathbf{c}_\ell$ as cluster means:

$$\mathbf{c}_\ell = \sum_{i \text{ in cluster } \ell} \mathbf{u}_i \Big/ \sum_{i \text{ in cluster } \ell} 1.$$

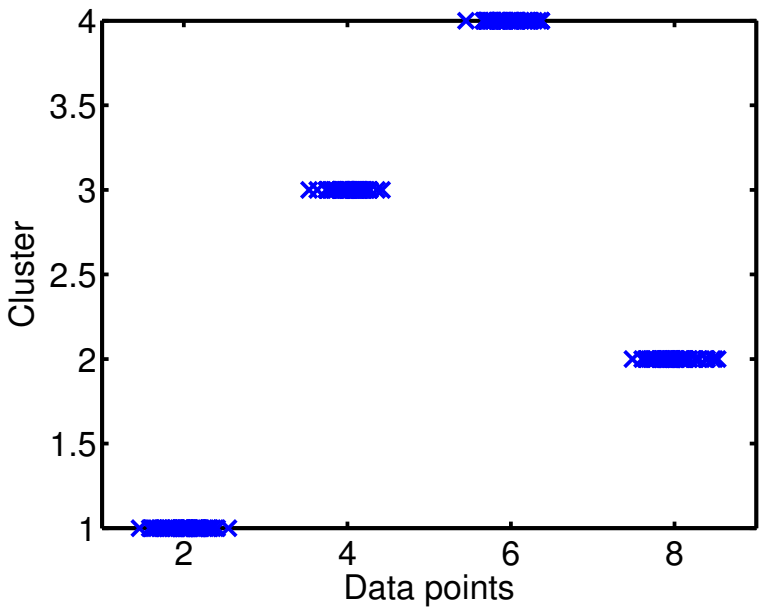
2. Assign each $\mathbf{u}_i$ to the cluster with the nearest cluster center.
3. Goto Step 1.

The algorithm is stopped when the assigned clusters do not change in an iteration.

Spectral clustering (cont.)

Example: continued:

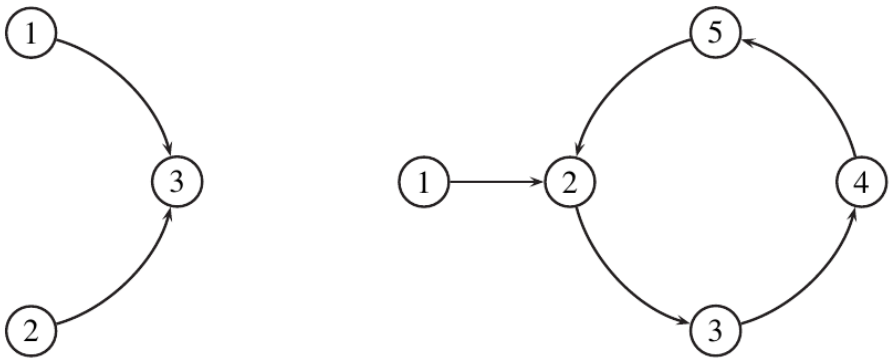
The k-means algorithm applied to the previous eigenbasis converges in 2 iteration steps and results in the following clustering:



Google's page rank

- One of the reasons why Google is such an effective search engine is the PageRank that determines the importance of a web page.
- PageRank is determined entirely by the link structure of the World Wide Web.
- Then, for any particular query, Google finds the pages on the Web that match that query and lists those pages in the order of their PageRank.
- Let's imagine a surfer going from page to page by randomly choosing an outgoing link from one page to get to the next.

Google's page rank (cont.)



- To escape dead ends, a random page of the web is chosen.
- To avoid cycles, at a fraction of time, a random page of the web is chosen.
- This theoretical random walk is known as a Markov chain or Markov process.

Google's page rank (cont.)

- Let W be the set of (reachable) web pages and let $n = |W|$.
- Connectivity matrix $G \in \mathbb{R}^{n \times n}$:

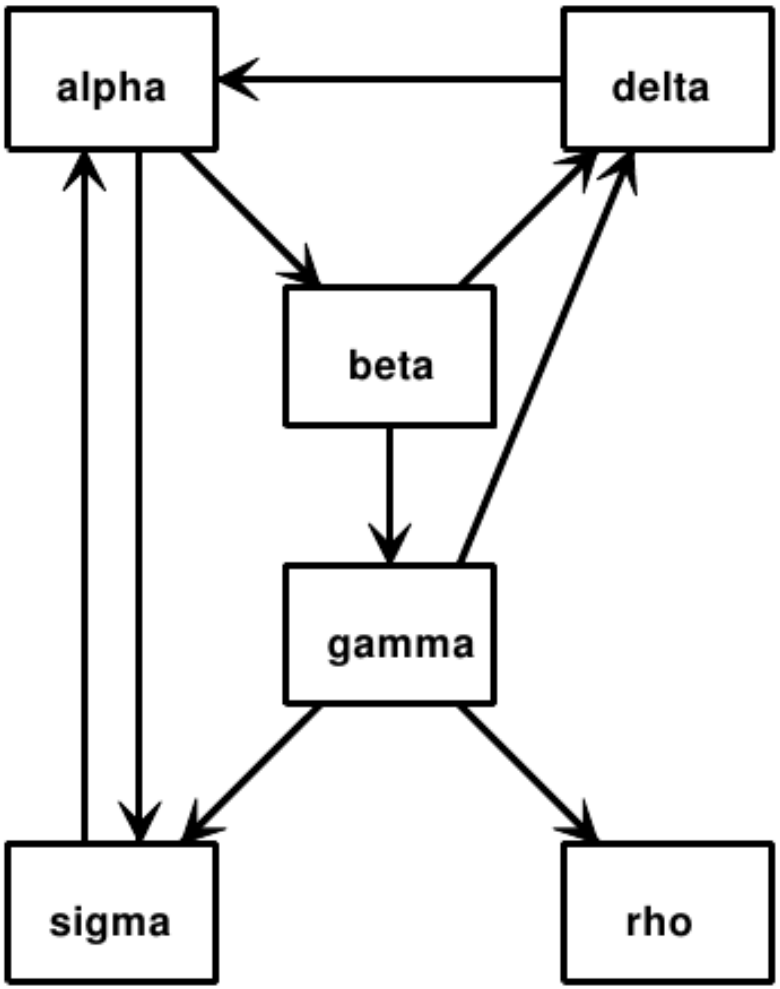
$$g_{ij} = \begin{cases} 1 & \text{there is a hyperlink } j \mapsto i, \\ 0 & \text{otherwise.} \end{cases}$$

`nnz(G)` = number of hyperlinks in W.
Let r_i and c_j be the row and column sums of G:

$$r_i = \sum_j g_{ij}, \qquad c_j = \sum_i g_{ij}.$$

$\Longrightarrow$ r_i = in-degree, c_j = out-degree of the jth page.
($c_j = 0$ is a dead end)

Google's page rank (cont.)



α	β	γ	δ	ρ	σ
1	2	3	4	5	6

$$G = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Google's page rank (cont.)

- Let A be the matrix with elements

$$a_{ij} = \begin{cases} g_{ij}/c_j & \text{if } c_j \neq 0 \\ 1/n & \text{if } c_j = 0 \text{ (dead end).} \end{cases}$$

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & \frac{1}{6} & 1 \\ \frac{1}{2} & 0 & 0 & 0 & \frac{1}{6} & 0 \\ 0 & \frac{1}{2} & 0 & 0 & \frac{1}{6} & 0 \\ 0 & \frac{1}{2} & \frac{1}{3} & 0 & \frac{1}{6} & 0 \\ 0 & 0 & \frac{1}{3} & 0 & \frac{1}{6} & 0 \\ \frac{1}{2} & 0 & \frac{1}{3} & 0 & \frac{1}{6} & 0 \end{bmatrix}$$

- Let $\mathbf{e} = (1, 1, \ldots, 1)^T$. Then $A^T\mathbf{e} = \mathbf{e}$ (or $\mathbf{e}^T A = \mathbf{e}^T$). So, $1 \in \sigma(A^T) = \sigma(A)$.

Google's page rank (cont.)

- To be able to escape cycles or strong components we follows the links only with a probability α.
- With probability $1-\alpha$ we choose a random page.
- We replace A by the matrix

$$\tilde{A} = \alpha A + (1-\alpha)\mathbf{p}\mathbf{e}^T,$$

  where $\mathbf{p}$ is a personalization vectors. ($\mathbf{p}$ has nonnegative elements that sum to 1, $\|\mathbf{p}\|_1 = 1$.
- We may choose $\mathbf{p} = \mathbf{e}/n$.
- Note that $\mathbf{e}^T\tilde{A} = \mathbf{e}^T$
- Most of the elements of A are very small. If $n = 4 \cdot 10^9$ and $\alpha = 0.85$, then the probability of jumping from one page to another without following a link is $\delta = 3.75 \cdot 10^{-11}$.

Google's page rank (cont.)

The Perron–Frobenius theorem applies. It states that a nonzero solution of the equation

$$\mathbf{x} = \tilde{A}\mathbf{x}$$

exists and is unique to within a scaling factor. If this scaling factor is chosen so that

$$\sum_{i=1}^{n} x_i = 1$$

then $\mathbf{x}$ is the state vector of the Markov chain and is Google's PageRank. The elements of $\mathbf{x}$ are all positive and less than one. This vector $\mathbf{x}$ is the eigenvector corresponding to the largest eigenvalue of $\tilde{A}$. It can be determined by vector iteration, aka. power method.

MATLAB code

```
function [x,cnt] = pagerankpow(G)
% PAGERANKPOW  PageRank by power method with no matrix operations.
% x = pagerankpow(G) is the PageRank of the graph G.
% [x,cnt] = pagerankpow(G) also counts the number of iterations.
% There are no matrix operations.  Only the link structure
% of G is used with the power method.

% Link structure
[n,n] = size(G);
for j = 1:n
   L{j} = find(G(:,j)); % set of links coming into node j
   c(j) = length(L{j}); % in-degree
end

% Power method
p = .85;  delta = (1-p)/n;
```

MATLAB code (cont.)

```
x = ones(n,1)/n;  z = zeros(n,1);
cnt = 0;
while max(abs(x-z)) > .0001
   z = x;
   x = zeros(n,1);
   for j = 1:n
      if c(j) == 0
         x = x + z(j)/n;
      else
         x(L{j}) = x(L{j}) + z(j)/c(j);
      end
   end
   x = p*x + delta;
   cnt = cnt+1;
end
```