MOTIVATION

Numerical problems are everywhere in geometric modeling!

Quick summary!
Mostly for common ground: You may already know this material. First half is important; remainder summarizes interesting recent tools.
Numerical analysis is a huge field.

Patterns, algorithms, & examples common in geometry.
EXAMPLES

How to flatten a mesh?

Object space (3D)  Texture space (2D)

“Mesh Parameterization: Theory and Practice” by Kai Hormann, Bruno Lévy, Alla Sheffer
EXAMPLES

How to flatten a mesh?

\[
\min_{u_i, v_i \in \mathbb{R}^2} f(u_1, v_1, \ldots, u_n, v_n)
\]

Fit a target 2D shape?

Triangle distortion?

More sharp corners?
EXAMPLES

How to animate a character?

rest pose  LBS with [Jacobson et al. 2011]  our method

“Linear Subspace Design for Real-Time Shape Deformation” by Yu Wang, Alec Jacobson, Jernej Barbič, Ladislav Kavan
EXAMPLES

How to animate a character?

\[
\min_{x_i,y_i \in \mathbb{R}^2} f(x_1, y_1, \ldots, x_n, y_n)
\]

s.t. \( x_j - x'_j = 0 \)

\( y_j - y'_j = 0 \)

Triangle distortion?

Some smoothness?

j \in J

Known positions
\[ \min_{x \in \mathbb{R}^n} f(x) \]

\[ \text{s.t. } g(x) = 0 \]

\[ h(x) \geq 0 \]
\[
\min_{x \in \mathbb{R}^n} f(x)
\]

s.t. \( g(x) = 0 \)

\( h(x) \geq 0 \)
\[
\min_{x \in \mathbb{R}^n} f(x)
\]
subject to
\[
\begin{align*}
g(x) &= 0 \\
h(x) &\geq 0
\end{align*}
\]
Equality constraints
\[
\min_{x \in \mathbb{R}^n} f(x)
\quad \text{s.t. } g(x) = 0
\quad h(x) \geq 0
\]

Inequality constraints
EXAMPLES

\[ Ax = b \quad \iff \quad \min_{x \in \mathbb{R}^n} \| Ax - b \|_2^2 \]
EXAMPLES

\[ Ax = \lambda x \quad \leftrightarrow \quad \min_{x \in \mathbb{R}^n} \| Ax \|_2^2 \]

s.t. \[ \|x\| - 1 = 0 \]
ROUGH PLAN

• *intro* Matrices and Eigenvalues
• Linear problems
• Unconstrained optimization
• Equality-constrained optimization
• Variational problems
ROUGH PLAN

- (intro) Matrices and Eigenvalues
- Linear problems
- Unconstrained optimization
- Equality-constrained optimization
- Variational problems
How are those matrices special?

\[
\begin{pmatrix}
1 & -2 \\
-1 & 2 \\
5 & 0
\end{pmatrix}
\quad \begin{pmatrix}
1 & -1 & 0 \\
-1 & 2 & 0 \\
0 & 0 & 5
\end{pmatrix}
\quad \begin{pmatrix}
0 & -1 & -2 \\
1 & 0 & 4 \\
2 & -4 & 0
\end{pmatrix}
\]

\[
\begin{pmatrix}
\cos(0.4) & -\sin(0.4) & 0 \\
\sin(0.4) & \cos(0.4) & 0 \\
0 & 0 & 1
\end{pmatrix}
\quad \begin{pmatrix}
1 & 0 & 3 \\
0 & 1 & 4 \\
1 & & 1
\end{pmatrix}
\]
MATRICES

How are those matrices special?

Diagonal:
\[
\begin{pmatrix}
1 & -2 \\
-2 & 5
\end{pmatrix}
\]

Symmetric:
\[
\begin{pmatrix}
1 & -1 & 0 \\
-1 & 2 & 0 \\
0 & 0 & 5
\end{pmatrix}
\]

Skew-symmetric:
\[
\begin{pmatrix}
0 & -1 & -2 \\
1 & 0 & 4 \\
2 & -4 & 0
\end{pmatrix}
\]

Orthogonal:
\[
\begin{pmatrix}
\cos(0.4) & -\sin(0.4) & 0 \\
\sin(0.4) & \cos(0.4) & 0 \\
0 & 0 & 1
\end{pmatrix}
\]

Um something?
MATRICES

What is the geometric meaning of those?

\[
\begin{pmatrix}
1 & -2 \\
-2 & 5
\end{pmatrix}
\]

Diagonal

\[
\begin{pmatrix}
\cos(0.4) & -\sin(0.4) & 0 \\
\sin(0.4) & \cos(0.4) & 0 \\
0 & 0 & 1
\end{pmatrix}
\]

Orthogonal

\[A^{-1} = A^T\]

\[
\begin{pmatrix}
1 & 0 & 3 \\
0 & 1 & 4 \\
0 & 0 & 1
\end{pmatrix}
\]

Um something?
INTRO TO INTRO

What is the geometric meaning of those?

\[
\begin{pmatrix}
1 & -2 \\
5 & \end{pmatrix}
\]

Non-uniform scaling

\[
\begin{pmatrix}
\cos(0.4) & -\sin(0.4) & 0 \\
\sin(0.4) & \cos(0.4) & 0 \\
0 & 0 & 1
\end{pmatrix}
\]

Rotation

\[
\begin{pmatrix}
1 & 0 & 3 \\
0 & 1 & 4 \\
0 & 0 & 1
\end{pmatrix}
\]

Translation in 2D*
MATRICES

Eigenvalues and eigenvectors

\[ Ax = \lambda x \] 

or

\[ \mathcal{L}[f(x)] = \lambda f(x) \]  

*discrete*  

*continuous*
MATRICES

• Geometric transformations
• Linear operators on functions
EIGENVECTORS/VALUES

Geometric meaning?

Vectors which only scale
MATRIX IS POSITIVE DEFINITE

\[ A \succeq 0 \]

\[ \iff \]

All eigenvalues are nonnegative

Geometric meaning?
\[
\min_{x \in \mathbb{R}^n} f(x) \\
\text{s.t. } g(x) = 0 \\
\quad h(x) \geq 0
\]
ROUGH PLAN

• (intro) Matrices and Eigenvalues

• Linear problems

• Unconstrained optimization

• Equality-constrained optimization

• Variational problems
VECTOR SPACES AND LINEAR OPERATORS

\[
\mathcal{L}[\vec{x} + \vec{y}] = \mathcal{L}[\vec{x}] + \mathcal{L}[\vec{y}]
\]
\[
\mathcal{L}[c\vec{x}] = c\mathcal{L}[\vec{x}]
\]
ABSTRACT EXAMPLE

$C^\infty(\mathbb{R})$

$L[f] := \frac{df}{dx}$

Eigenvectors?
INFINITE DIMENSIONS

\[
\begin{align*}
A & \quad \vec{x} \\
\text{matrix vector} \\
\vec{x} & \mapsto A \vec{x}
\end{align*}
\]
LINEAR SYSTEM OF EQUATIONS

\[
\begin{pmatrix}
A \\
\end{pmatrix}
\begin{pmatrix}
\vec{x}
\end{pmatrix}
=
\begin{pmatrix}
\vec{b}
\end{pmatrix}
\]

Simple “inverse problem”
COMMON STRATEGIES

• Gaussian elimination
  – $O(n^3)$ time to solve $Ax=b$ or to invert

• But: Inversion is unstable and slower!

• Never ever compute $A^{-1}$ if you can avoid it.
SIMPLE EXAMPLE

\[
\frac{d^2 f}{dx^2} = g, \quad f(0) = f(1) = 0
\]

\[
\begin{pmatrix}
-2 & 1 \\
1 & -2 & 1 \\
1 & -2 & 1 \\
1 & -2 & 1 \\
\vdots & & \\
1 & -2 & 1 \\
1 & -2 & 1
\end{pmatrix}
\begin{pmatrix}
f_1 \\
f_2 \\
\vdots \\
f_n
\end{pmatrix}
=
\begin{pmatrix}
g_1 \\
g_2 \\
\vdots \\
g_n
\end{pmatrix}
\]
STRUCTURE?

\[
\begin{pmatrix}
-2 & 1 \\
1 & -2 & 1 \\
1 & -2 & 1 \\
\vdots & \vdots & \vdots \\
1 & -2 & 1 \\
1 & -2 & 1
\end{pmatrix}
\]
LINEAR SOLVER
CONSIDERATIONS

• Never construct $A^{-1}$ explicitly
  (if you can avoid it)

• Added structure helps
  Sparsity, symmetry, positive definiteness, bandedness

\[
\text{inv}(A) \ast b \ll (A' \ast A) \backslash (A' \ast b) \ll A \backslash b
\]
LINEAR SYSTEMS: SOLVERS

• **Direct** (*explicit* matrix)
  – **Dense**: Gaussian elimination/LU, QR for least-squares
  – **Sparse**: Reordering (SuiteSparse, Eigen)

• **Iterative** (*apply* matrix repeatedly)
  – **Positive definite**: Conjugate gradients
  – **Symmetric**: MINRES, GMRES
  – **Generic**: LSQR
GENERIC ADVICE

Generic tools are often not too effective!
Try the simplest solver first.
VERY COMMON: SPARSITY

Induced by the connectivity of the triangle mesh.

Iteration of CG has local effect ⇒ Precondition!
FOR IFT 6113

- No need to implement a linear solver
- If a matrix is sparse, your code should store it as a sparse matrix!

https://eigen.tuxfamily.org/dox/group__TutorialSparse.html
ROUGH PLAN

• *(intro)* Matrices and Eigenvalues
• Linear problems
• Unconstrained optimization
• Equality-constrained optimization
• Variational problems
UNCONSTRAINED OPTIMIZATION

\[
\min_{x \in \mathbb{R}^n} f(x)
\]

Trivial when \( f(x) \) is linear

Easy when \( f(x) \) is quadratic

Hard in case of generic non-linear.
UNCONSTRAINED OPTIMIZATION

\[ \min_{x \in \mathbb{R}^n} f(x) \]
NOTIONS FROM CALCULUS

\[ f(x, y) = 3x^2 - y \]
\[ \nabla f = \left( \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right) = (6x, -1) \]

Geometric meaning?
NOTIONS FROM CALCULUS

\[ f : \mathbb{R}^n \rightarrow \mathbb{R} \]

\[ \nabla f = \left( \frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_n} \right) \]

https://en.wikipedia.org/?title=Gradient
NOTIONS FROM CALCULUS

\[ f : \mathbb{R}^n \rightarrow \mathbb{R} \]

\[ \nabla f = \left( \frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \ldots, \frac{\partial f}{\partial x_n} \right) \]

https://en.wikipedia.org/?title=Gradient
NOTIONS FROM CALCULUS

\[ f : \mathbb{R}^n \rightarrow \mathbb{R}^m \]

\[ \rightarrow (Df)_{ij} = \frac{\partial f_i}{\partial x_j} \]

NOTIONS FROM CALCULUS

\[ f : \mathbb{R}^n \rightarrow \mathbb{R} \rightarrow H_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j} \]

\[ f(x) \approx f(x_0) + \nabla f(x_0)^\top (x - x_0) + (x - x_0)^\top H f(x_0) (x - x_0) \]

http://math.etsu.edu/multicalc/prealpha/Chap2/Chap2-5/10-3a-t3.gif
OPTIMIZATION TO ROOT-FINDING

\[ \nabla f(x) = 0 \] (unconstrained)

Critical point

Saddle point

Local min

Local max

\[ f(x) \]

x
CONVEX FUNCTIONS

\[ f''(x) > 0 \]

https://en.wikipedia.org/wiki/Convex_function
CONVEX FUNCTIONS

\[ H(x) \geq 0 \]

https://en.wikipedia.org/wiki/Convex_function
SPECIAL CASE: LEAST-SQUARES

\[
\min_x \frac{1}{2} \| Ax - b \|_2^2
\]

\[\rightarrow \min_x \frac{1}{2} x^\top A^\top Ax - b^\top Ax + \| b \|_2^2\]

\[\implies A^\top Ax = A^\top b\]

Normal equations
(better solvers for this case!)
The Matrix Cookbook
Petersen and Pedersen

UNCONSTRAINED OPTIMIZATION

\[ \min_{x} f(x) \]
Gradient descent

\[ x_{k+1} = x_k - \alpha_k \nabla f(x_k) \]

Multiple optima!
Line search

BASIC ALGORITHMS
BASIC ALGORITHMS

\[ \lambda_0 = 0, \lambda_s = \frac{1}{2} \left( 1 + \sqrt{1 + 4\lambda_{s-1}^2} \right), \gamma_s = \frac{1 - \lambda_2}{\lambda_{s+1}} \]

\[ y_{s+1} = x_s - \frac{1}{\beta} \nabla f (x_s) \]

\[ x_{s+1} = (1 - \gamma_s) y_{s+1} + \gamma_s y_s \]

**Quadratic convergence on convex problems!**
(Nesterov 1983)

A very cool intro: [https://distill.pub/2017/momentum/](https://distill.pub/2017/momentum/)
Newton's Method

\[ x_{k+1} = x_k - \left[ H f(x_k) \right]^{-1} \nabla f(x_k) \]
BASIC ALGORITHMS

\[ x_{k+1} = x_k - M_k^{-1} \nabla f(x_k) \]

• (Often sparse) approximation from previous samples and gradients
• Inverse in closed form!

Quasi-Newton: BFGS and friends
EXAMPLE: SHAPE INTERPOLATION

Figure 5: Interpolation and extrapolation of the yellow example poses. The blending weights are 0, 0.35, 0.65, 1.0, and 1.25.

Figure 6: Interpolation of an adaptively meshed and strongly twisted helix with blending weights 0, 0.25, 0.5, 0.75, 1.0.

Fröhlich and Botsch. “Example-Driven Deformations Based on Discrete Shells.” CGF 2011.
INTERPOLATION PIPELINE

Roughly:

1. **Linearly interpolate** edge lengths and dihedral angles.

\[
\ell_e^* = (1 - t) \ell_e^0 + t \ell_e^1 \\
\theta_e^* = (1 - t) \theta_e^0 + t \theta_e^1
\]

2. **Nonlinear** optimization for vertex positions.

\[
\min_{x_1, \ldots, x_m} \lambda \sum_{e} w_e (\ell_e(x) - \ell_e^*)^2 \\
+ \mu \sum_{e} w_b (\theta_e(x) - \theta_e^*)^2
\]

*Sum of squares: Gauss-Newton*
SOFTWARE

- Matlab: fminunc or minfunc
- C++: libLBFGS, dlib, others

Typically provide functions for function and gradient (and optionally, Hessian).

Try several!
SOME TRICKS

Lots of small elements:  \( \| x \|_2^2 = \sum_i x_i^2 \)
Lots of zeros: \( \| x \|_1 = \sum_i |x_i| \)
Uniform norm: \( \| x \|_\infty = \max_i |x_i| \)
Low rank: \( \| X \|_* = \sum \sigma_i \)

Mostly zero columns: \( \| X \|_{2,1} = \sum_j \sqrt{\sum_i x_{ij}^2} \)

Smooth: \( \int \| \nabla f \|_2^2 \)
Piecewise constant: \( \int \| \nabla f \|_2 \)

???: Early stopping

Regularization
SOME TRICKS

Multiscale/graduated optimization
ROUGH PLAN

• (intro) Matrices and Eigenvalues
• Linear problems
• Unconstrained optimization
• Equality-constrained optimization
• Variational problems
LAGRANGE MULTIPLIERS: IDEA

$$\min_x f(x) \quad \text{s.t.} \quad g(x) = 0$$

$$f(x) = c$$
LAGRANGE MULTIPLIERS: IDEA

\[
\min_x f(x) \\
\text{s.t. } g(x) = 0
\]

- Decrease \( f \): \(-\nabla f\)
- Violate constraint: \( \pm \nabla g \)
LAGRANGE MULTIPLIERS: IDEA

\[ \min_x f(x) \]
\[ \text{s.t. } g(x) = 0 \]

Want:
\[ \nabla f \parallel \nabla g \]
\[ \implies \nabla f = \lambda \nabla g \]
USE OF LAGRANGE MULTIPLIERS

Turns constrained optimization into unconstrained root-finding.

\[ \nabla f(x) = \lambda \nabla g(x) \]

\[ g(x) = 0 \]
\[
\min_{x \in \mathbb{R}^n} f(x) \quad \text{quadratic}
\]

s.t. \quad g(x) = 0 \quad \text{linear}
QUADRATIC WITH LINEAR EQUALITY

\[
\begin{align*}
\min_x & \quad \frac{1}{2} x^\top A x - b^\top x + c \\
\text{s.t.} & \quad M x = v
\end{align*}
\]

(assume A is symmetric and positive definite)
QUADRATIC WITH LINEAR EQUALITY

\[ \min_x \frac{1}{2} x^\top Ax - b^\top x + c \]

s.t. \[ Mx = \nu \]

(assume \( A \) is symmetric and positive definite)

\[
\begin{pmatrix}
A & M^\top \\
M & 0
\end{pmatrix}
\begin{pmatrix}
x \\
\lambda
\end{pmatrix}
= 
\begin{pmatrix}
b \\
\nu
\end{pmatrix}
\]
\[
\min_{x \in \mathbb{R}^n} f(x) \\
\text{s.t. } g(x) = 0
\]
MANY OPTIONS

• **Reparameterization**
  Eliminate constraints to reduce to unconstrained case

• **Newton’s method**
  Approximation: quadratic function with linear constraint

• **Penalty method**
  Augment objective with barrier term, e.g. $f(x) + \rho |g(x)|$
EXAMPLE: SYMMETRIC EIGENVECTORS

\[ f(x) = x^\top Ax \implies \nabla f(x) = 2Ax \]

\[ g(x) = \|x\|^2 \implies \nabla g(x) = 2x \]

\[ \implies Ax = \lambda x \]
EXAMPLE: MESH EMBEDDING

G. Peyré, mesh processing course slides
LINEAR SOLVE FOR EMBEDDING

\[ x_i \in \mathbb{R}^2 \]

\[
\min_{x_1, \ldots, x_{|V|}} \sum_{(i,j) \in E} w_{ij} \| x_i - x_j \|_2^2
\]

s.t. \( x_v \) fixed \( \forall v \in V_0 \)

- \( w_{ij} \equiv 1 \): Tutte embedding
- \( w_{ij} \) from mesh: Harmonic embedding

Assumption: \( w \) symmetric.
What if $V_0 = \emptyset$?
NONTRIVIALITY CONSTRAINT

\[
\begin{align*}
\min_x & \quad \|Ax\|_2 \\
\text{s.t.} & \quad \|x\|_2 = 1
\end{align*}
\mapsto A^\top Ax = \lambda x
\]

Prevents trivial solution \( x \equiv 0 \).

Extract the smallest eigenvalue.

\[
\begin{align*}
\min_u \quad & u^\top L_C u \\
& u^\top B e = 0 \quad \text{Easy fix} \\
& u^\top B u = 1
\end{align*}
\]

\[ L_C u = \lambda B u \]
BASIC IDEA OF EIGENALGORITHMS

\[ A\vec{v} = c_1 A\vec{x}_1 + \cdots + c_n A\vec{x}_n \]
\[ = c_1 \lambda_1 \vec{x}_1 + \cdots + c_n \lambda_n \vec{x}_n \text{ since } A\vec{x}_i = \lambda_i \vec{x}_i \]
\[ = \lambda_1 \left( c_1 \vec{x}_1 + \frac{\lambda_2}{\lambda_1} c_2 \vec{x}_2 + \cdots + \frac{\lambda_n}{\lambda_1} c_n \vec{x}_n \right) \]
\[ A^2 \vec{v} = \lambda_1^2 \left( c_1 \vec{x}_1 + \left( \frac{\lambda_2}{\lambda_1} \right)^2 c_2 \vec{x}_2 + \cdots + \left( \frac{\lambda_n}{\lambda_1} \right)^2 c_n \vec{x}_n \right) \]
\[ \vdots \]
\[ A^k \vec{v} = \lambda_1^k \left( c_1 \vec{x}_1 + \left( \frac{\lambda_2}{\lambda_1} \right)^k c_2 \vec{x}_2 + \cdots + \left( \frac{\lambda_n}{\lambda_1} \right)^k c_n \vec{x}_n \right). \]
TRUST REGION METHODS

Example: Levenberg-Marquardt

\[
\begin{aligned}
\min_{\delta x} & \quad \frac{1}{2} \delta x^\top H \delta x + w^\top x \\
\text{s.t.} & \quad \|\delta x\|^2_2 \leq \Delta \\
\downarrow & \quad (H + \lambda I)\delta x = -w
\end{aligned}
\]

Fix (or adjust) damping parameter $\lambda > 0$. 
EXAMPLE: POLYCUBE MAPS


Align with coordinate axes

\[
\min_X \sum_{b_i} A(b_i; X) \| n(b_i; X) \|_1 \\
\text{s.t.} \quad \sum_{b_i} A(b_i; X) = \sum_{b_i} A(b_i; X_0)
\]

Preserve area

Note: Final method includes more terms!
Aside:
Convex Optimization Tools

versus

Sometimes work for non-convex problems...

Try lightweight options
ITERATIVELY REWEIGHTED LEAST SQUARES

\[
\min_x \sum_i \phi(x^\top a_i + b_i) \leftrightarrow \left\{ \begin{array}{l}
\min_{x,y_i} \sum_i y_i (x^\top a_i + b_i)^2 \\
\text{s.t. } y_i = \phi(x^\top a_i + b_i)(x^\top a_i + b_i)^{-2}
\end{array} \right\}
\]

“Geometric median”

\[
\min_x \sum_i \|x - p_i\|_2 \quad \Rightarrow \quad \begin{cases}
x & \leftarrow \min_x \sum_i y_i \|x - p_i\|_2^2 \\
y_i & \leftarrow \|x - p_i\|_2^{-1}
\end{cases}
\]

Repeatedly solve linear systems
ALTERNATING PROJECTION

$$\min_{p} d(p, p_0)$$

s.t. $$p \in C_1 \cap C_2 \cap \cdots \cap C_k$$

$d$ can be a Bregman divergence
AUGMENTED LAGRANGIANS

\[
\begin{align*}
\min_x & \quad f(x) \\
\text{s.t.} & \quad g(x) = 0 \\
\downarrow & \\
\min_x & \quad f(x) + \frac{\rho}{2} \|g(x)\|^2_2 \\
\text{s.t.} & \quad g(x) = 0
\end{align*}
\]

Add constraint to objective

Does nothing when constraint is satisfied
ALTERNATING DIRECTION METHOD OF MULTIPLIERS (ADMM)

\[ \min_{x,z} \ f(x) + g(z) \]

\[ \text{s.t.} \quad Ax + Bz = c \]

\[ \Lambda_\rho(x, z; \lambda) = f(x) + g(z) + \lambda^\top (Ax + Bz - c) + \frac{\rho}{2} \|Ax + Bz - c\|_2^2 \]

\[ x \leftarrow \arg \min_x \Lambda_\rho(x, z, \lambda) \]

\[ z \leftarrow \arg \min_z \Lambda_\rho(x, z, \lambda) \]

\[ \lambda \leftarrow \lambda + \rho(Ax + Bz - c) \]

To minimize $f(x)$ s.t. $x \in \mathcal{D}$:

$$s_k \leftarrow \begin{cases} \arg \min_s s^\top \nabla f(x_k) \\ \text{s.t. } s \in \mathcal{D} \end{cases}$$

$$\gamma \leftarrow \frac{2}{k + 2}$$

$$x_{k+1} \leftarrow x_k + \gamma(s_k - x_k)$$

Linearize objective, preserve constraints
ROUGH PLAN

• (intro) Matrices and Eigenvalues
• Linear problems
• Unconstrained optimization
• Equality-constrained optimization
• Variational problems
VARIATIONAL CALCULUS:  
BIG IDEA

Sometimes your unknowns are not numbers!

Can we use calculus to optimize anyway?
ON THE BOARD

\[
\min_f \int_\Omega \left\| \vec{v}(x) - \nabla f(x) \right\|_2^2 \, d\vec{x}
\]

\[
\min \int_\Omega f(x)^2 \, d\vec{x} = 1 \int_\Omega \left\| \nabla f(x) \right\|_2^2 \, d\vec{x}
\]
GÂTEAUX DERIVATIVE

\[ d\mathcal{F}[u; \psi] := \frac{d}{dh} \mathcal{F}[u + h\psi]|_{h=0} \]

Vanishes for all \( \psi \) at a critical point!

Analog of derivative at \( u \) in \( \psi \) direction