



CONVEX RELAXATION STRATEGIES FOR COMPUTER GRAPHICS

Leticia Mattos Da Silva



OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX

Non-standard

Linear Programming
(LP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

Semi-Definite
Programming
(SDP)

Quadratic
Programming
(QP)

CONSTRAINED

UNCONSTRAINED

OUR END GOAL



let's use question words!

WHY

Why convex relaxation?

WHAT

What can be done via convex relaxation?

HOW

How is convex relaxation used?

Subskill: Answering questions in your own words
When you find the answer to a question in the text, try to think of another way to say the same thing.

Subskill: Noticing differences between the question and what you hear
Remember that there will be differences between the questions and the recordings. Remember that there will be differences between the questions and the recordings. You won't hear exactly the same words.

Music Festival
The first festival happened in 2010.
Open to young musicians under 25 years old.
Enjoy no more than 3 hours of music.
such as 4 different bands.
Cost for workshops: adults 5 \$
students 6 \$
Cost for concerts: adults 7 \$
students 8 \$
Organizers donate part of the money
9 %
Family activities including 10 \$

Prepare an experience day for your students

1 There was a festival in... the festival was in...
2 Listen to the interview and write down your ideas in exercise 1.

3 Look at the notes about Bratislava. Think of different ways in which you can hear the information.

4 Listen to the text again and answer the questions in your own words.

- 1 How long do you drive the Lamborghini for?
You drive _____ for 10 minutes.
- 2 What do you need to do in the escape room?
You need to _____.
- 3 Where do you spend the end of your day at?
The Bear Grylls Adventure?

- 4 Where will you go on the 3D-design course?
Why will you go there?

OUR END GOAL



let's use question words!

WHY

Why convex relaxation?

WHAT

What can be done via convex relaxation?

HOW

How is convex relaxation used?

we won't cover as much theory as a convex optimization course!

the examples of convex relaxation are non-exhaustive

we won't dive deep into all the technical aspects of each work

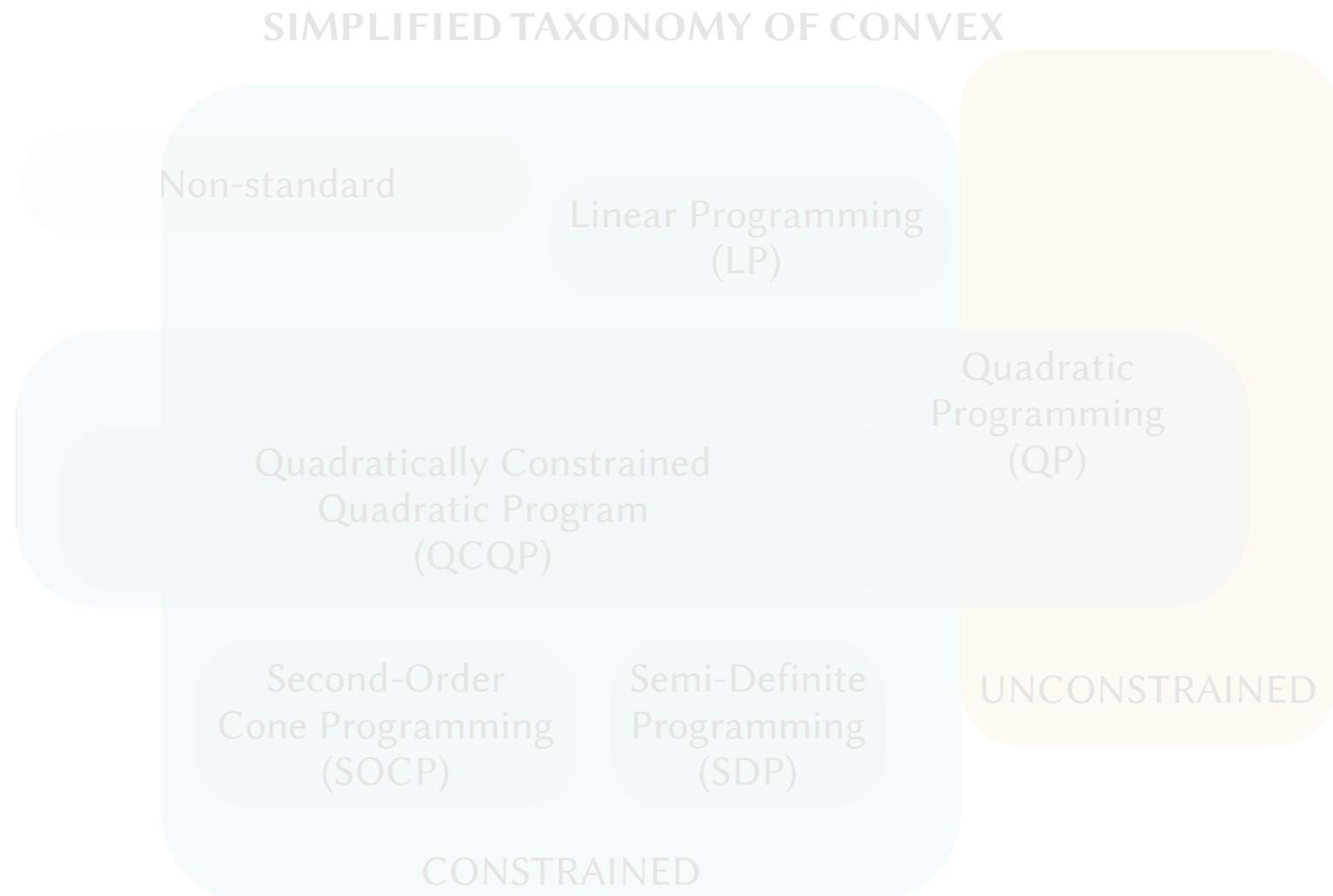
OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

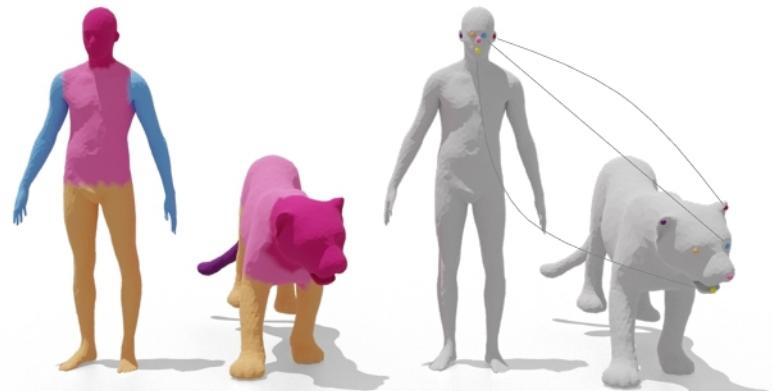


THE FUNDAMENTALS

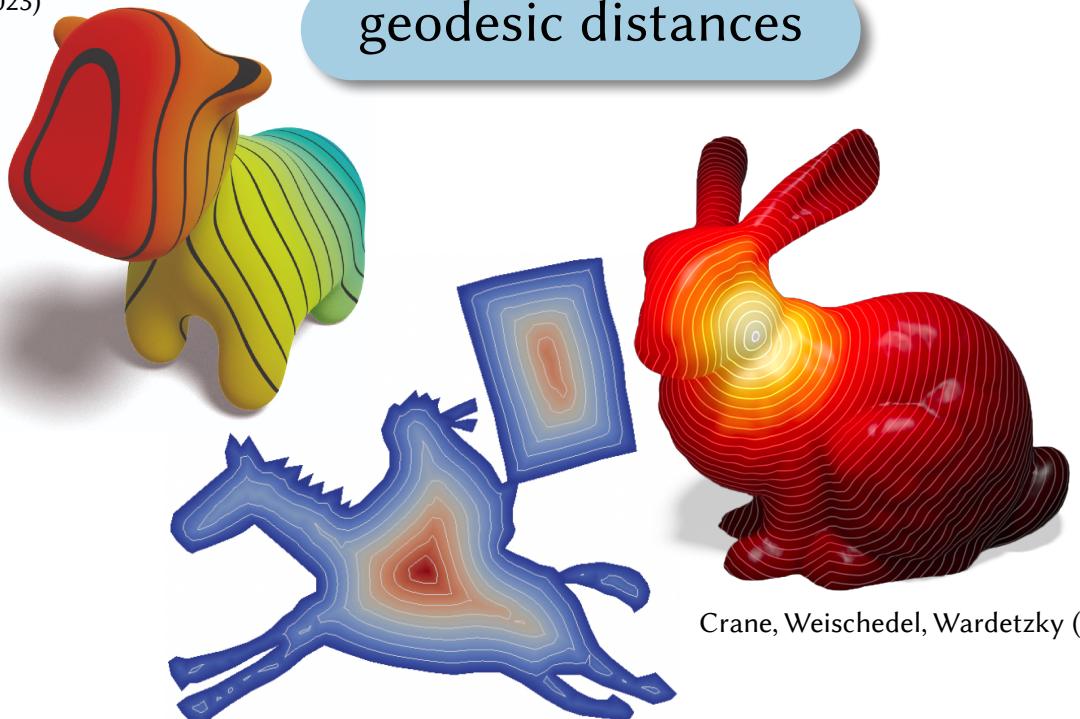
Computer graphics is full of hard math problems!

Edelstein, Guillén, Solomon,
Ben-Chen (2023)

shape correspondence



Abdelreheem, Eldekokey, Ovsjanikov, Wonka (2023)



Crane, Weischedel, Wardetzky (2013)

Belyaev & Fayolle (2015)

optimal transport



Solomon, Goes, Peyre, Cuturi, Butscher,
Nguyen, Du, Guibas (2015)

THE FUNDAMENTALS

Edelstein, Guillén, Solomon,
Ben-Chen (2023)

Computer graphics is full of hard math problems!

geodesic distances

shape co...

REFORMULATED AS AN OPTIMIZATION PROBLEM

$$\begin{aligned} & \min \text{ or } \max f(x) \\ & \text{subject to } x \in \Pi \end{aligned}$$

Abdelreheem, Eldesokey, Ovsjanikov, Wonka (2018)

optimal transport

Crane, Weischedel, Wardetzky (2013)

Solomon, Goes, Peyre, Cuturi, Butscher,
Nguyen, Du, Guibas (2015)

THE FUNDAMENTALS

**REFORMULATED AS AN
OPTIMIZATION PROBLEM**

$$\begin{aligned} & \text{min or max } f(x) \\ & \text{subject to } x \in \Pi \end{aligned}$$

THE FUNDAMENTALS

find the x that makes the **objective function** $f(x)$ as small or big as possible

REFORMULATED AS AN
OPTIMIZATION PROBLEM

\min or $\max f(x)$

subject to $x \in \Pi$

THE FUNDAMENTALS

find the x that makes the **objective function** $f(x)$ as small or big as possible

constraints define the set of all x that we are allowed to consider

**REFORMULATED AS AN
OPTIMIZATION PROBLEM**

\min or $\max f(x)$

subject to $x \in \Pi$

THE FUNDAMENTALS

find the x that makes the **objective function** $f(x)$ as small or big as possible

constraints define the set of all x that we are allowed to consider

the set of all x that satisfy the constraints is called **feasible**

**REFORMULATED AS AN
OPTIMIZATION PROBLEM**

\min or $\max f(x)$

subject to $x \in \Pi$

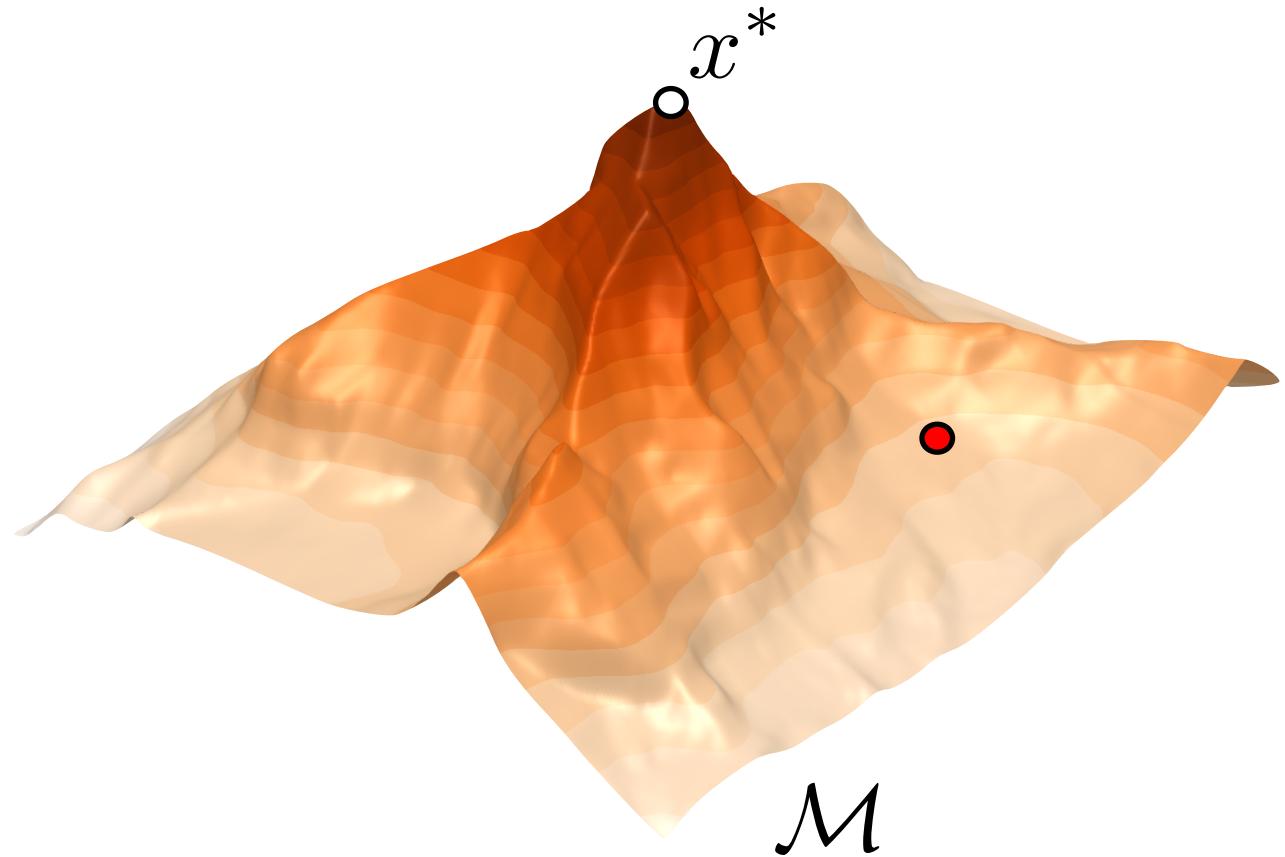
THE FUNDAMENTALS

$$x^* = \arg \max_{x \in \mathcal{M}} f(x)$$

From calculus...

$$\nabla f(x) = 0$$

Identifies the critical points



THE FUNDAMENTALS

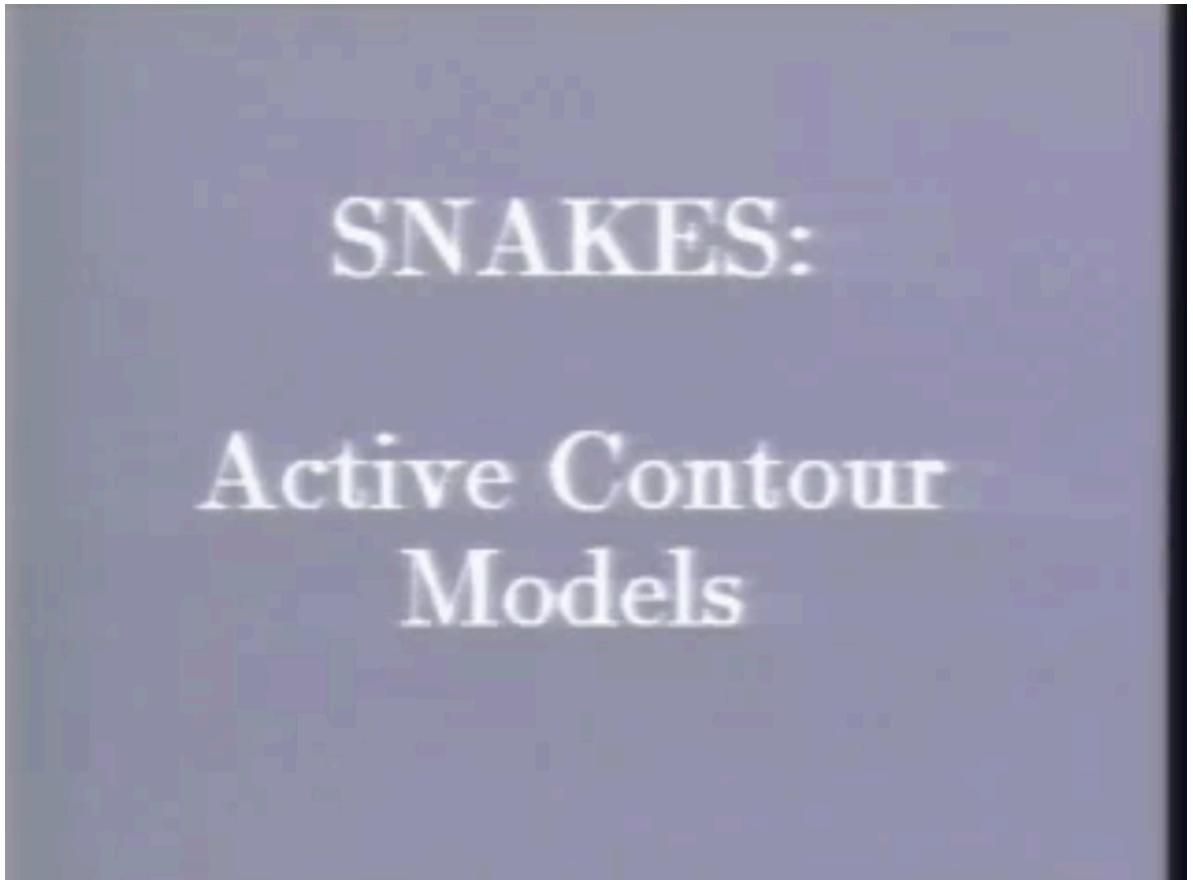
Snakes: Active Contour Models

Kass, Witkin and Terzopoulos (1988)

A snake is an energy-minimizing spline guided by external constraint forces and influenced by image forces that pull it toward features such as lines and edges.

$$v(s) = (x(s), y(s)), \ s \in [0, 1]$$

$$\text{minimize } \mathcal{E}_{\text{snake}}(v)$$



THE FUNDAMENTALS

Snakes: Active Contour Models
Kass, Witkin and Terzopoulos (1988)
A snake is an energy-minimizing spline
guided by external constraint forces and
influenced by image forces that pull it toward
features such as lines and edges.

$$v_{i+1} = v_i - \gamma \nabla \mathcal{E}_{\text{snake}}(v)$$

🔑 gradient descent

The result is **highly sensitive to initialization** because classical “snakes” models are **non-convex** energy minimization problems solved using gradient descent.

OUR ROADMAP

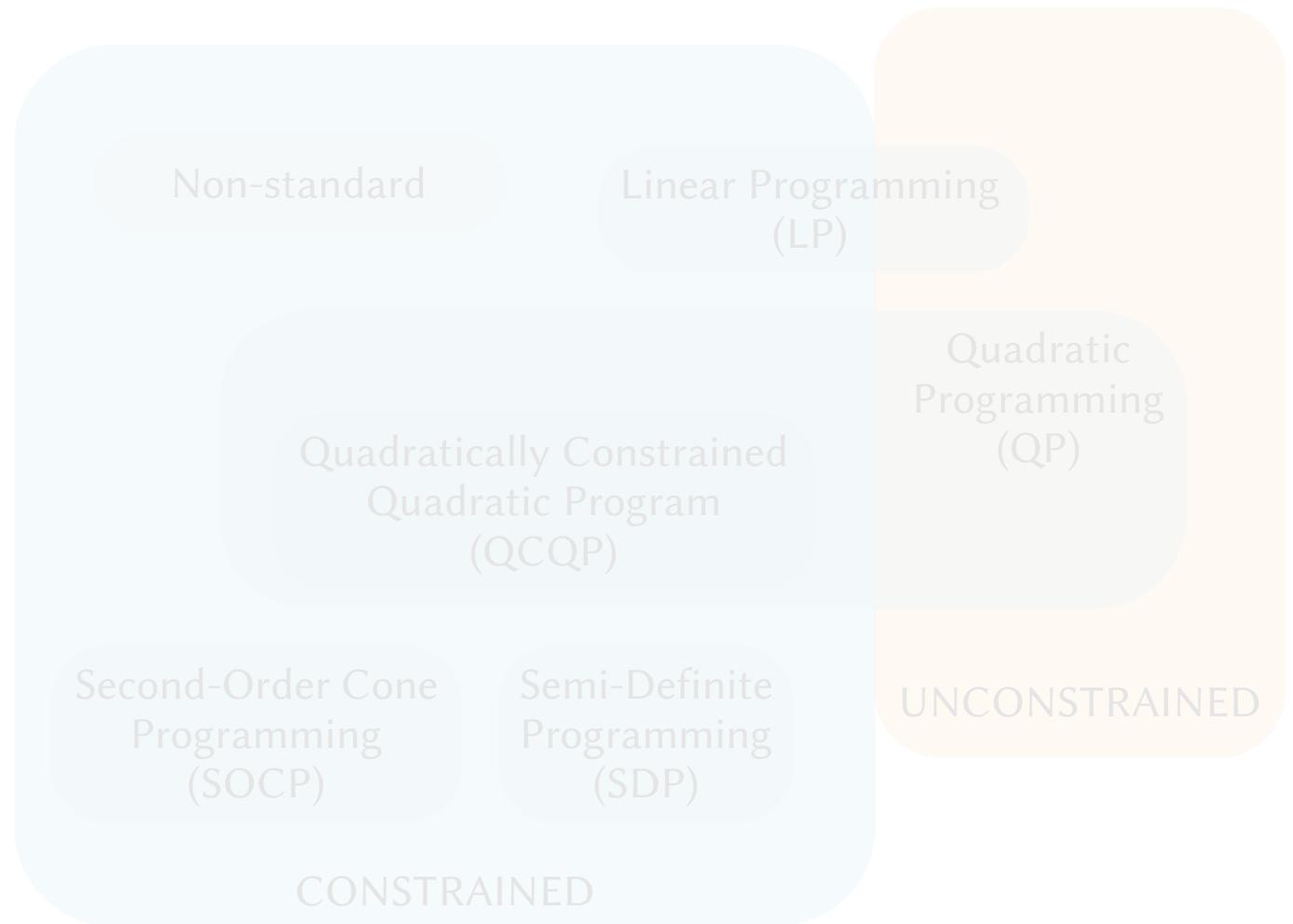
OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX PROBLEMS

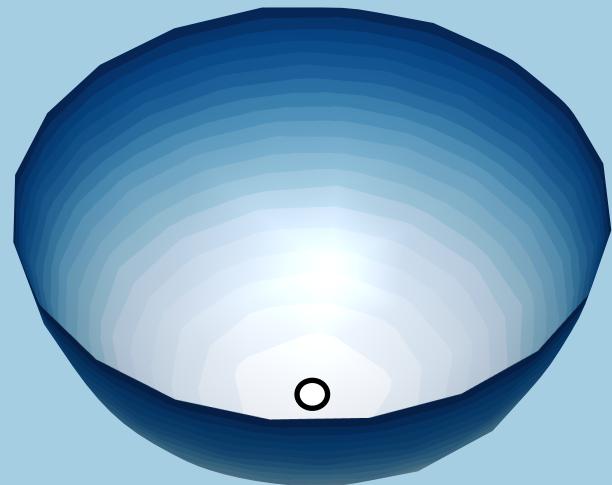


CONVEXITY

CONVEX

local optima are global optima

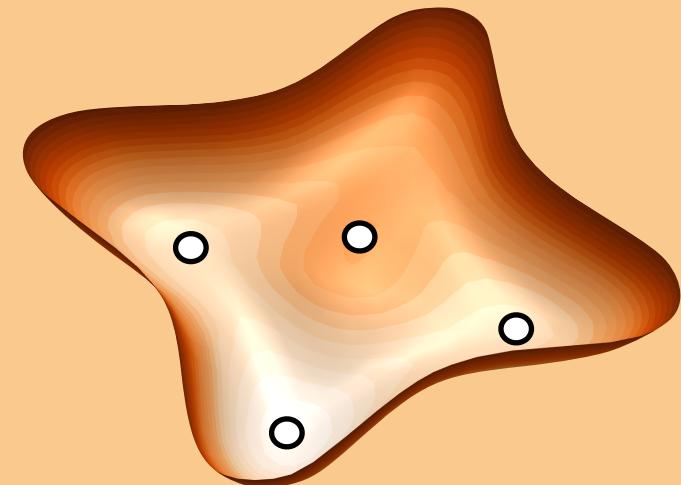
Convexity gives global optima



NON-CONVEX

multiple local optima

Often gets stuck on
local optima or fails

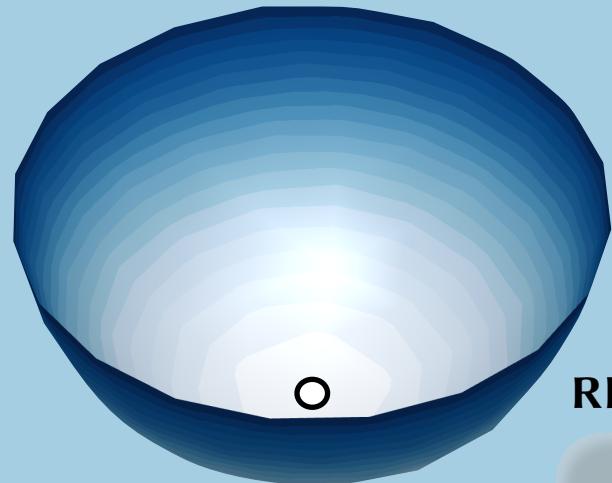


CONVEXITY

CONVEX

local optima are global optima

Convexity gives global optima



REQUIREMENTS:

the **objective function** $f(x)$ is convex

constraint set is convex

NON-CONVEX

multiple local optima

Often gets stuck on
local optima or fails



CONVEXITY

CONVEX

local optima are global optima

Convexity gives global optima

THE ISSUE WITH NON-CONVEX PROBLEMS:

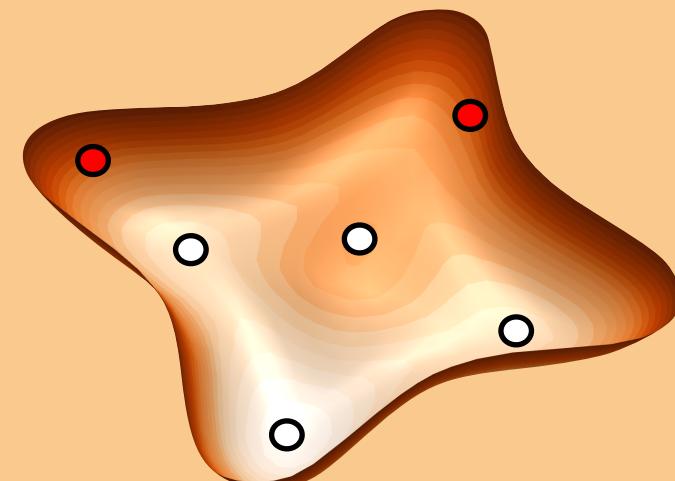
does not always recover the global optima

where we start matters!

NON-CONVEX

multiple local optima

Often gets stuck on
local optima or fails



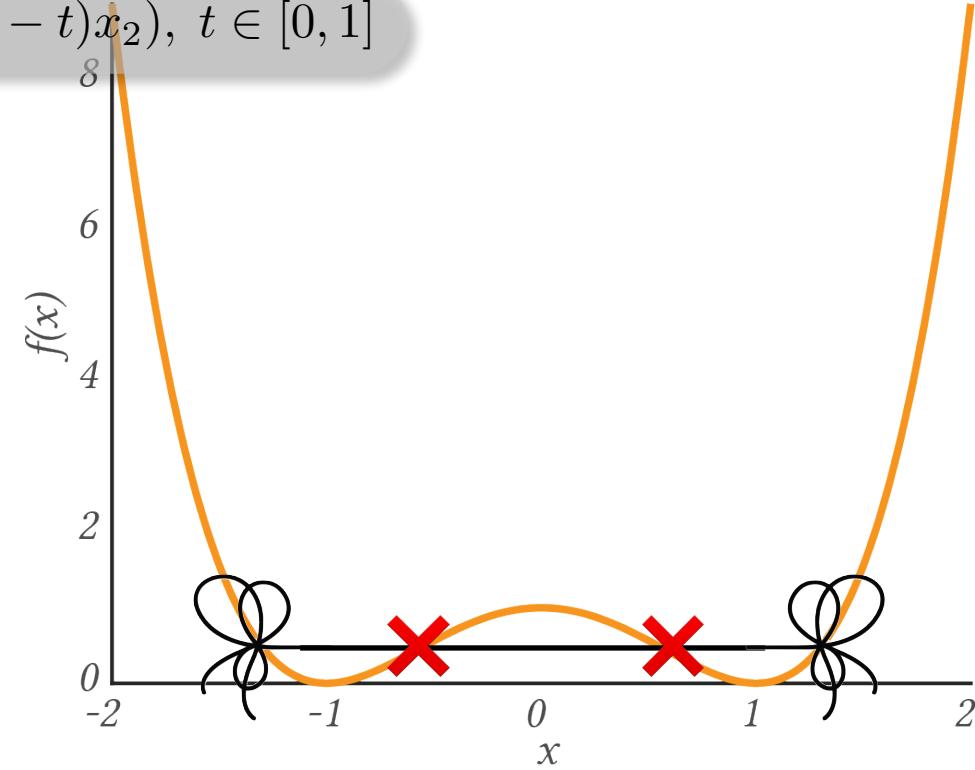
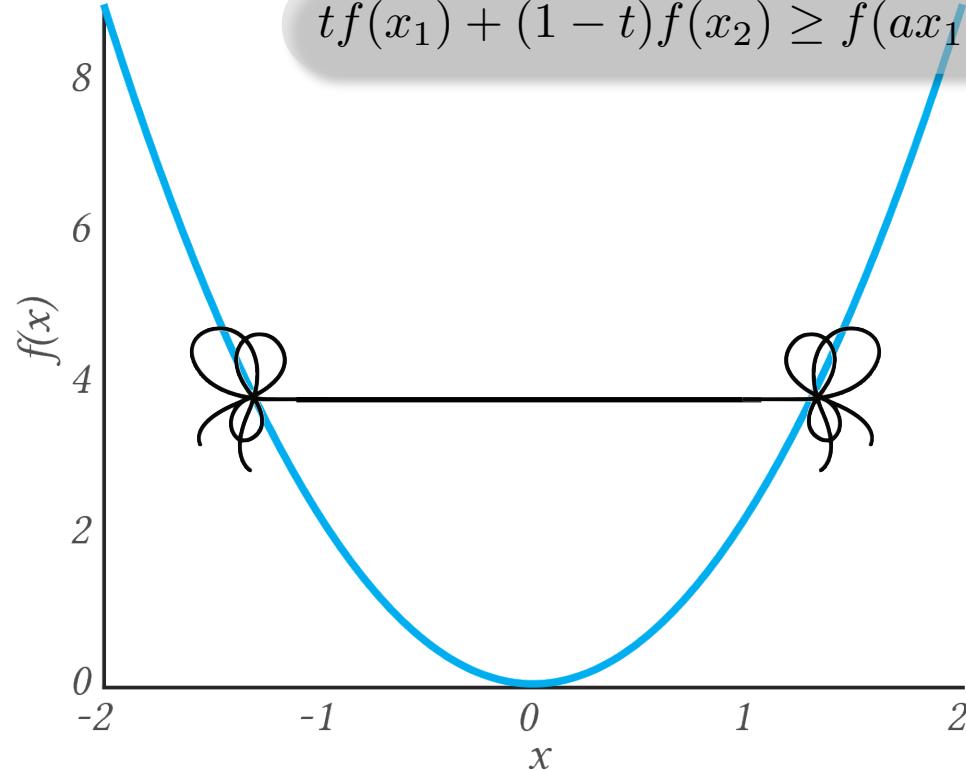
CONVEXITY

CONVEX FUNCTION

Convex functions lie below their **bowstrings**

mathematically

$$tf(x_1) + (1 - t)f(x_2) \geq f(tx_1 + (1 - t)x_2), \quad t \in [0, 1]$$



CONVEXITY

some standard convex functions...

linear functions

$$f(x) = c^\top x + d$$

norms

$$f(x) = \|x\|_p$$

exponential

$$f(x) = e^{ax}$$

certain powers

$$f(x) = x^a$$
$$x > 0, a \notin (0, 1)$$

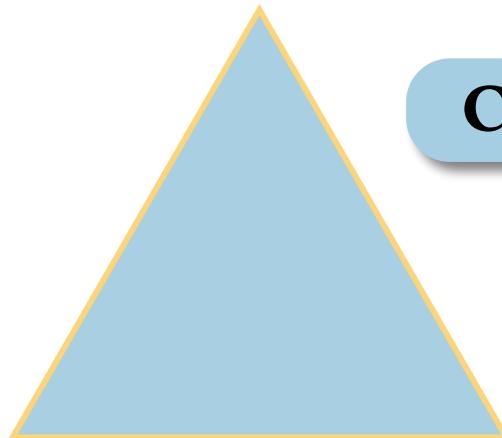
CONVEXITY

CONVEX SETS

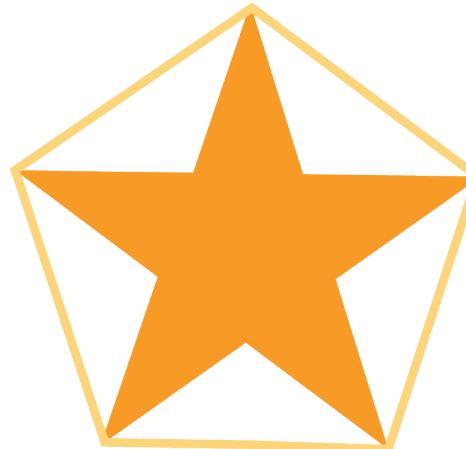
Convex sets **don't** lie below a rubber band

mathematically

$$tx_1 + (1 - t)x_2 \in \Pi, \forall x_1, x_2 \in \Pi, t \in [0, 1]$$



CONVEX

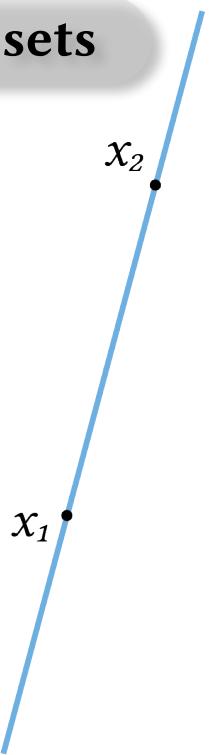


NON-CONVEX!

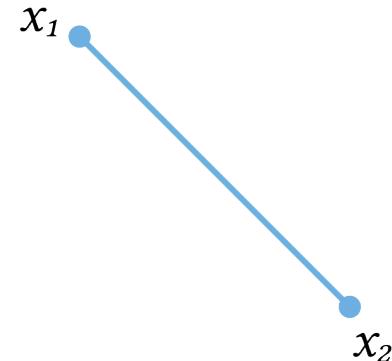
CONVEXITY

some standard convex sets...

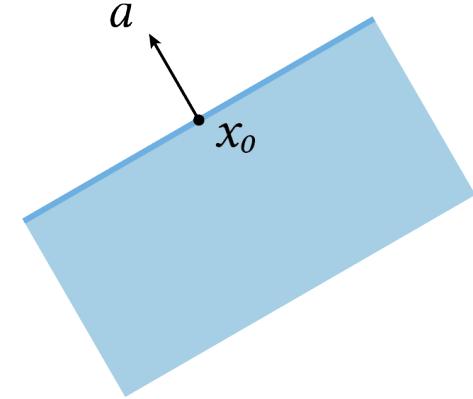
affine sets



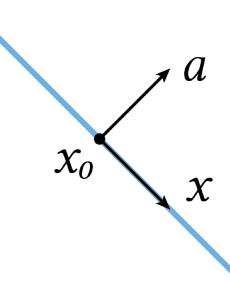
line segments



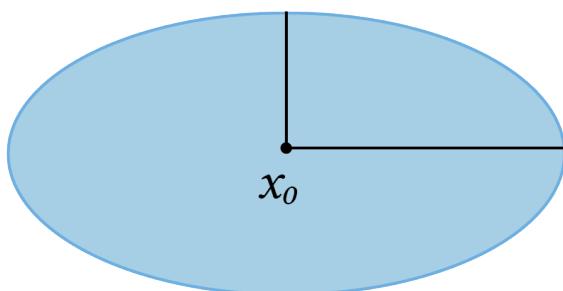
semi-planes



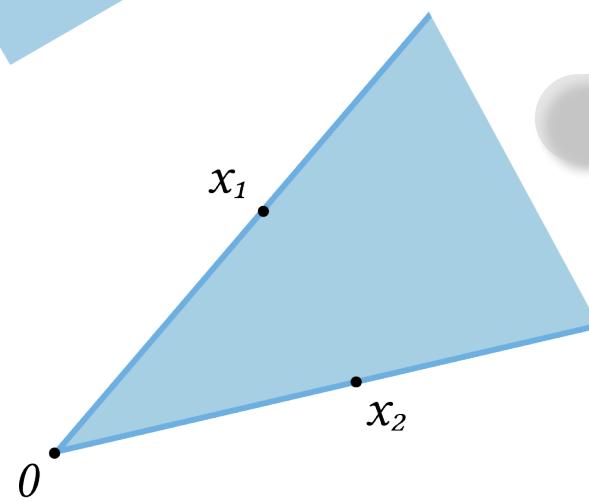
hyperplanes



ellipsoids



cones



OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX

Non-standard

Linear Programming
(LP)

Quadratic
Programming
(QP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

Semi-Definite
Programming
(SDP)

CONSTRAINED

UNCONSTRAINED

WHY CONVEXITY?

we can solve convex problems with efficiency and robustness

Convexity **gives global optima** and
theoretical guarantees

WHY CONVEXITY?

we can solve convex problems with efficiency and robustness

Convexity **gives global optima** and
theoretical guarantees

Algorithms are plentiful and “self-sufficient”

WHY CONVEXITY?

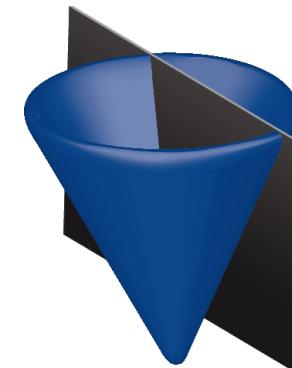
we can solve convex problems with efficiency and robustness

Convexity **gives global optima** and theoretical guarantees

Algorithms are plentiful and “self-sufficient”

Modern convex optimization software lets you **solve large, complex problems quickly**

🔧 help you prototype quicker!



mosek

GUROBI
OPTIMIZATION



YALMIP

SCS
SPLITTING CONIC SOLVER

WHY CONVEXITY?

we can solve convex problems with efficiency and robustness

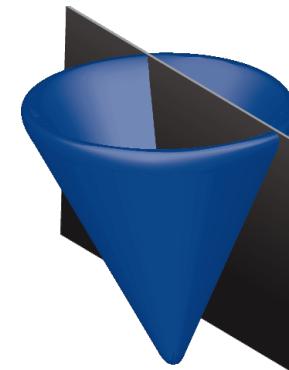
Convexity **gives global optima** and theoretical guarantees

Algorithms are plentiful and “self-sufficient”

Modern convex optimization software lets you **solve large, complex problems quickly**

Convex optimization software **helps you debug**

🔧 help you prototype quicker!



mosek



GUROBI
OPTIMIZATION

YALMIP



SCS

SPLITTING CONIC SOLVER

WHY CONVEXITY?

```
% From Boyd & Vandenberghe, "Convex Optimization"
% Joëlle Skaf - 09/26/05
%
% Solves the following QP with inequality constraints:
%       minimize    1/2x'*P*x + q'*x + r
%       s.t.      -1 <= x_i <= 1      for i = 1,2,3
% Also shows that the given x_star is indeed optimal
%
% Generate data
P = [13 12 -2; 12 17 6; -2 6 12];
q = [-22; -14.5; 13];
r = 1;
n = 3;
x_star = [1;1/2;-1];

% Construct and solve the model
fprintf(1,'Computing the optimal solution ...');
cvx_begin
    variable x(n)
    minimize ( (1/2)*quad_form(x,P) + q'*x + r)
    x >= -1;
    x <= 1;
cvx_end
fprintf(1,'Done! \n');

% Display results
disp('-----');
disp('The computed optimal solution is: ');
disp(x);
disp('The given optimal solution is: ');
disp(x_star);
```

🔧 help you prototype quicker!

```
Interior-point solution summary
Problem status : PRIMAL_AND_DUAL_FEASIBLE
Solution status : OPTIMAL
Primal. obj: 3.9124999974e+01    nrm: 3e+01    Viol. con: 3e-08    var: 0e+00
Dual.   obj: 3.9124999978e+01    nrm: 2e+00    Viol. con: 0e+00    var: 3e-09
Optimizer summary
Optimizer          -                               time: 0.02
    Interior-point - iterations : 10               time: 0.01
    Basis identification -                           time: 0.00
        Primal      - iterations : 0               time: 0.00
        Dual         - iterations : 0               time: 0.00
        Clean primal - iterations : 0              time: 0.00
        Clean dual   - iterations : 0              time: 0.00
    Simplex          -                               time: 0.00
        Primal simplex - iterations : 0            time: 0.00
        Dual simplex  - iterations : 0            time: 0.00
        Mixed integer - relaxations: 0           time: 0.00
-----
Status: Solved
Optimal value (cvx_optval): -21.625

Done!
-----
The computed optimal solution is:
1.0000
0.5000
-1.0000

The given optimal solution is:
1.0000
0.5000
-1.0000
```

WHY NOT CONVEXITY?

WHY NOT CONVEXITY?

**NON-CONVEX
PROBLEM**

What if you are given a bad hand?



WHY NOT CONVEXITY?

**NON-CONVEX
PROBLEM**

What if you are given a bad hand?



Convex Relaxation

SIMPLIFIED TAXONOMY OF CONVEX

Non-standard

Linear Programming
(LP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

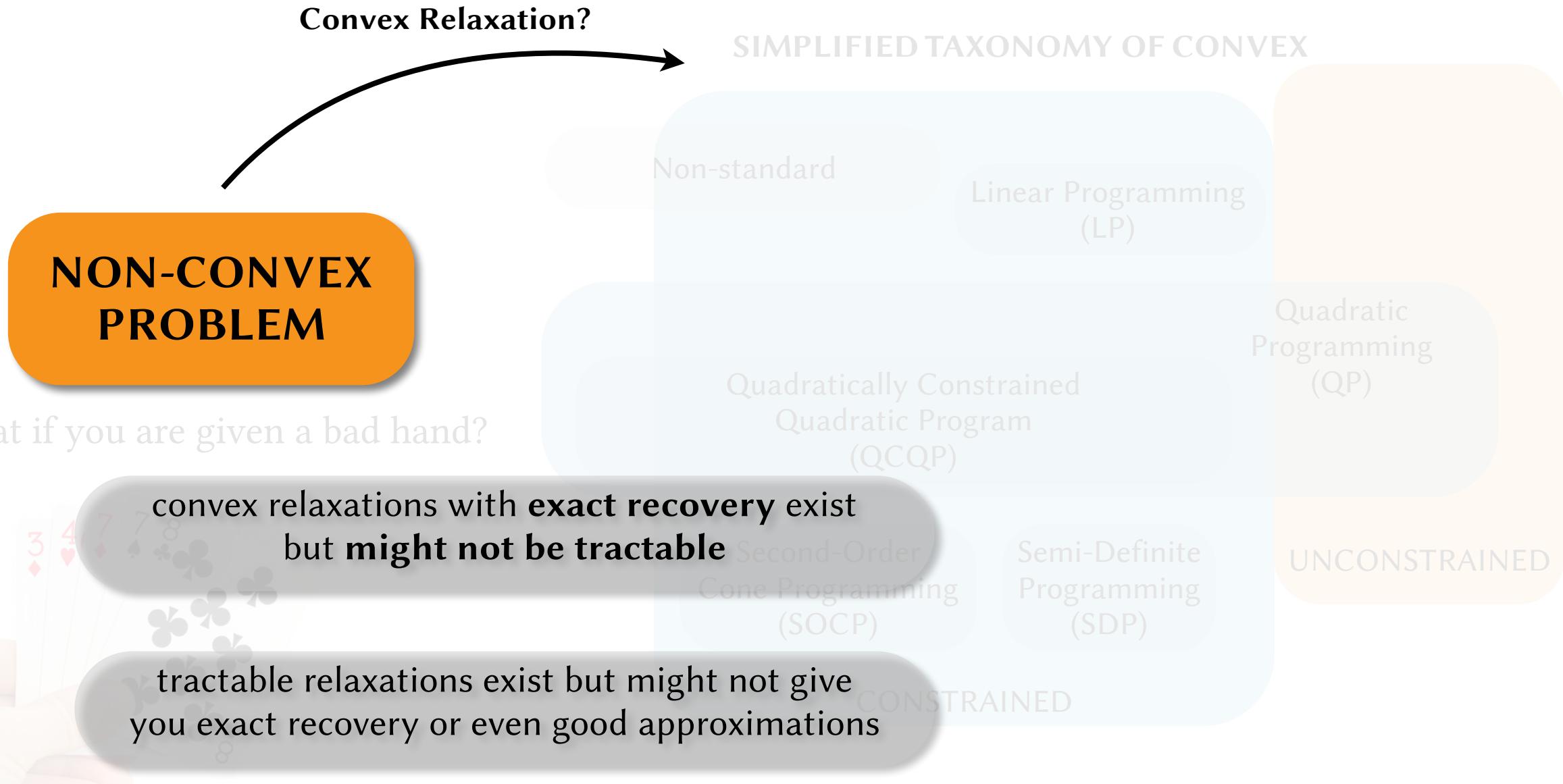
Semi-Definite
Programming
(SDP)

Quadratic
Programming
(QP)

CONSTRAINED

UNCONSTRAINED

WHY NOT CONVEXITY?



OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX

Non-standard

Linear Programming
(LP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

Semi-Definite
Programming
(SDP)

Quadratic
Programming
(QP)

CONSTRAINED

UNCONSTRAINED

OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX

Linear Programming
(LP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

Semi-Definite
Programming
(SDP)

Quadratic
Programming
(QP)

CONSTRAINED

UNCONSTRAINED

STANDARD CONVEX PROBLEMS

LINEAR PROGRAMMING (LP)

minimize $c^\top x + d$

subject to $Ax \leq b$

feasible set is a **convex polytope!**

find the variable x that...

minimizes the **linear objective function**

subject to the **linear inequality constraints**

STANDARD CONVEX PROBLEMS

LINEAR PROGRAMMING (LP)

minimize $c^\top x + d$
subject to $Ax \leq b$

feasible set is a **convex polytope!**

find the variable x that...

minimizes the **linear objective function**

subject to the **linear inequality constraints**

intersection of **semi-planes**

$$a_{11}x_{11} + a_{12}x_{12} + \cdots + a_{1n}x_{1n} \leq b_1$$

⋮

$$a_{m1}x_{m1} + a_{m2}x_{m2} + \cdots + a_{mn}x_{mn} \leq b_m$$

STANDARD CONVEX PROBLEMS

Optimal transport

Earth Mover's Distance

$$\mathcal{W}(\mu_0, \mu_1) := \inf_{\pi} \int \int d(x, y) d\pi(x, y)$$

s. t. $\pi \in \Pi(\mu_0, \mu_1)$

seeking a joint probability distribution



convex linear programming

STANDARD CONVEX PROBLEMS

Optimal transport

Earth Mover's Distance

$$\mathcal{W}(\mu_0, \mu_1) := \inf_{\pi} \int \int d(x, y) d\pi(x, y)$$

s. t. $\pi \in \Pi(\mu_0, \mu_1)$

seeking a joint probability distribution



convex linear programming



scales quadratically with the number of variables

STANDARD CONVEX PROBLEMS

Optimal transport

Earth Mover's Distance

$$\mathcal{W}(\mu_0, \mu_1) := \inf_{\pi} \int \int d(x, y) d\pi(x, y)$$

s. t. $\pi \in \Pi(\mu_0, \mu_1)$

seeking a joint probability distribution



convex linear programming



scales quadratically with the number of variables



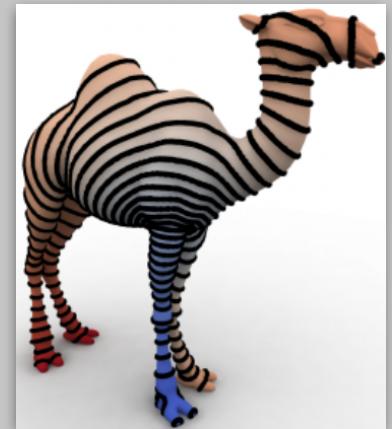
lesson: **convex does not always mean tractable!**

**Earth Mover's Distances
on Discrete Surfaces**

Solomon, Rustamov, Guibas, Butscher (2014)

$$\mathcal{W}(\mu_0, \mu_1) = \inf_J \int \|J(x)\| dx$$

s. t. $\nabla \cdot J(x) = \rho_1(x) - \rho_0(x)$
 $J(x) \cdot n(x) = 0 \quad \forall x \in \partial M$



OUR ROADMAP

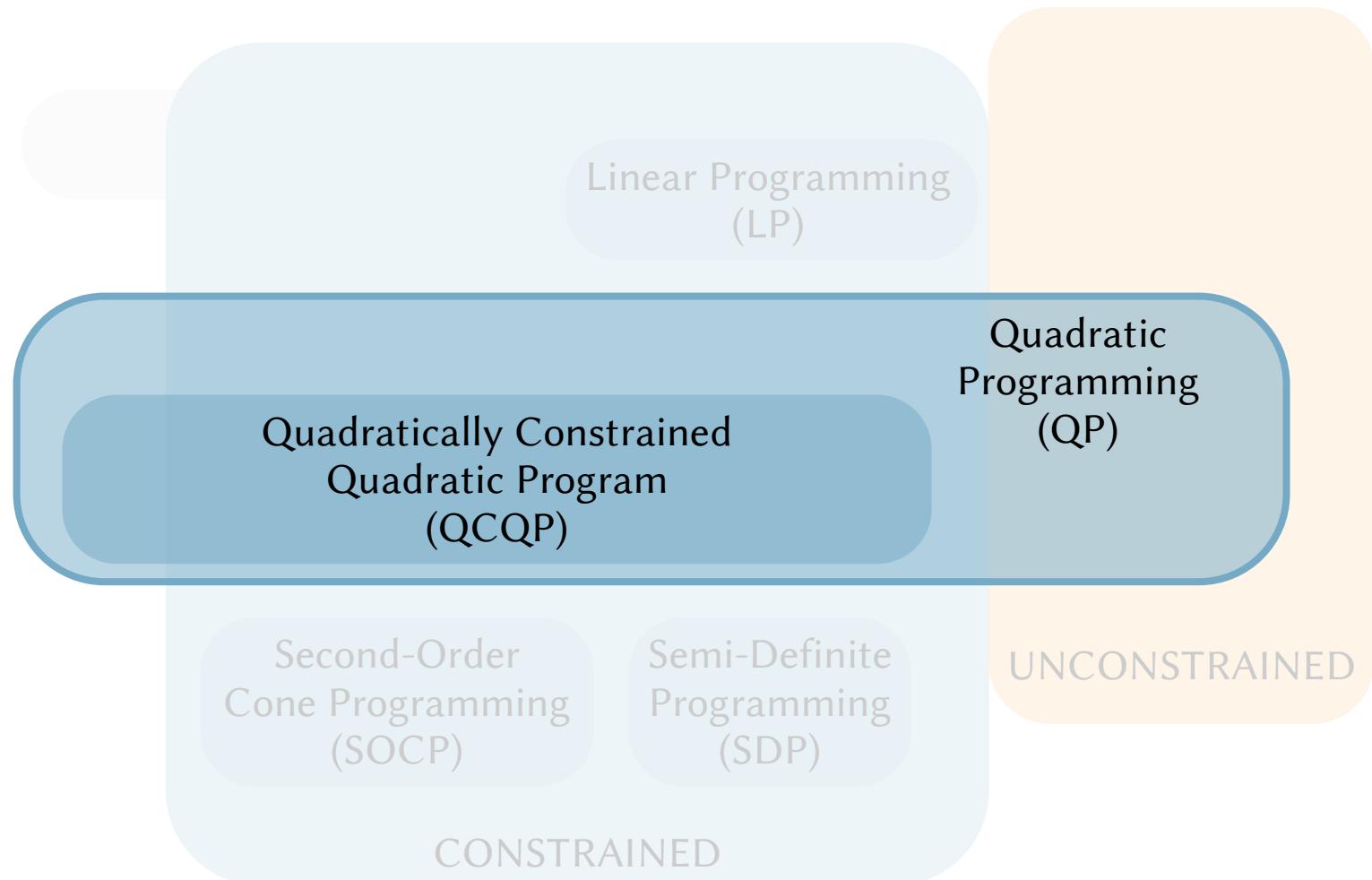
OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX



STANDARD CONVEX PROBLEMS

QUADRATIC PROGRAMMING (QP)

$$\text{minimize } \frac{1}{2}x^\top Qx + c^\top x + d$$

subject to $Ax \leq b$

$Q \in \mathbb{S}_+^n$ so the objective is **convex quadratic**

Positive Semi-Definite (PSD)

A symmetric matrix M is PSD if

$$v^\top M v \geq 0$$

for every vector v .

find the variable x that...

minimizes the **quadratic objective function**

subject to the **linear inequality constraints**

STANDARD CONVEX PROBLEMS

Shape deformation

Bounded Biharmonic Weights for Real-Time Deformation

Jacobson, Baran, Popović, Sorkine (2011)

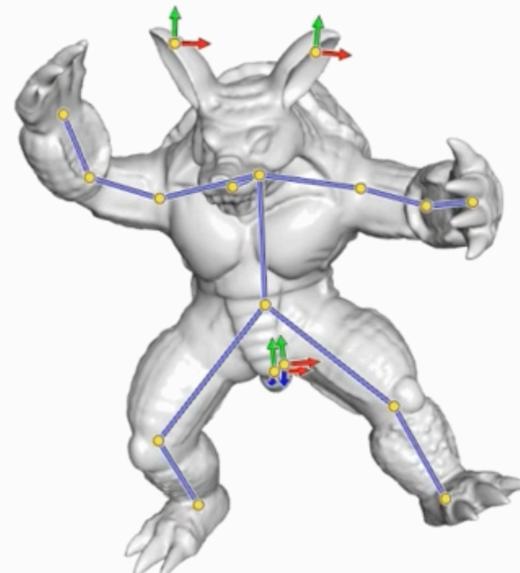
$$\arg \min_{w_j, j=1, \dots, m} \sum_{j=1}^m \frac{1}{2} \int_{\Omega} \|\Delta w_j\|^2 dV$$

$$\text{s. t. } w_j|_{H_k} = \delta_{jk}$$

$$w_j|_F \text{ is linear}$$

$$\sum_{j=1}^m w_j(\mathbf{p}) = 1$$

$$0 \leq w_j(\mathbf{p}) \leq 1, j = 1, \dots, m$$



OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX

Linear Programming
(LP)

Quadratic
Programming
(QP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

Semi-Definite
Programming
(SDP)

CONSTRAINED

UNCONSTRAINED

STANDARD CONVEX PROBLEMS

QUADRATICALLY CONSTRAINED QUADRATIC PROGRAMMING (QCQP)

$$\text{minimize } \frac{1}{2} x^\top Qx + c^\top x + d$$

$$\text{subject to } \frac{1}{2} x^\top P_i x + q_i^\top x + r_i \leq 0$$

$Q, P \in \mathbb{S}_+^n$ so the objective and constraints are **convex quadratic**

find the variable x that...

minimizes the **quadratic objective function**

subject to the **quadratic inequality constraints**

STANDARD CONVEX PROBLEMS

QUADRATICALLY CONSTRAINED QUADRATIC PROGRAMMING (QCQP)

$$\text{minimize } \frac{1}{2} x^\top Qx + c^\top x + d$$

$$\text{subject to } \frac{1}{2} x^\top P_i x + q_i^\top x + r_i \leq 0$$

$Q, P \in \mathbb{S}_+^n$ so the objective and constraints are **convex quadratic**

find the variable x that...

minimizes the **quadratic objective function**

subject to the **quadratic inequality constraints**

feasible set is the **intersection of ellipsoids!**

WHAT CAN I DONE?

Point-to-point

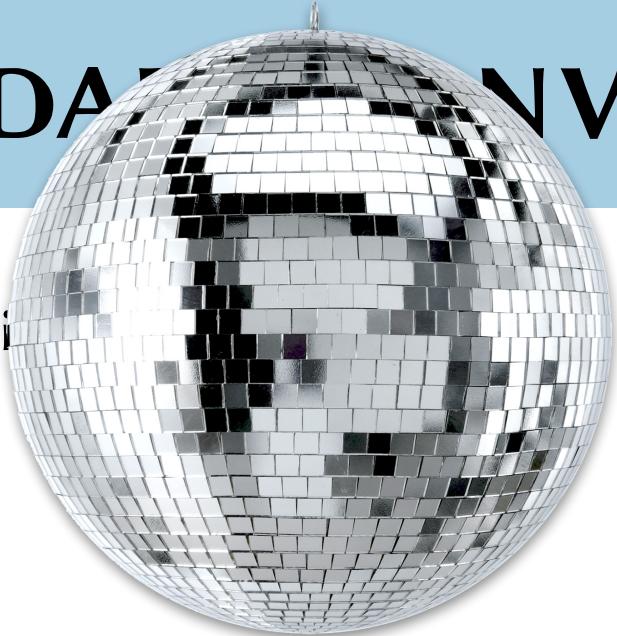


Procrustes matching

$$\begin{aligned} & \text{minimize} \|RP - QX\|_F^2 \\ & \text{s.t. } R^\top R = I, X \text{ a permutation} \end{aligned}$$

STANDARD CONVEX PROBLEMS

Point-to-point



Procrustes matching

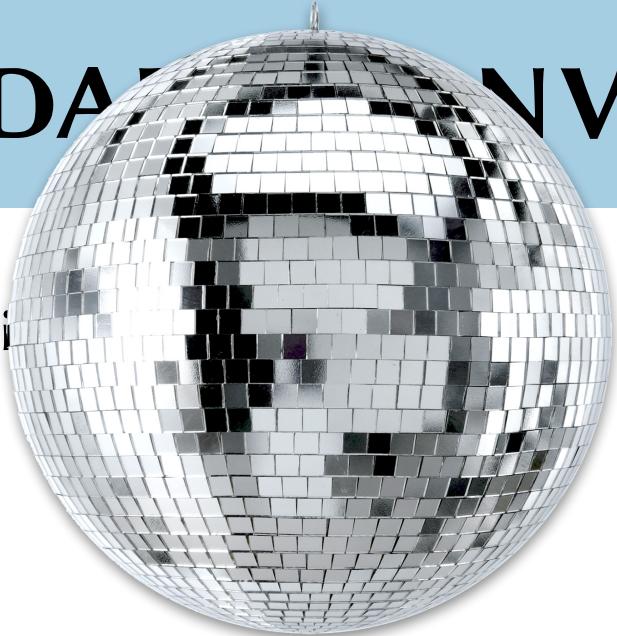
$$\begin{aligned} & \text{minimize} && \|RP - QX\|_F^2 \\ & \text{s.t. } R^\top R = I, X \text{ a permutation} \end{aligned}$$

Question: is this problem convex?



STANDARD CONVEX PROBLEMS

Point-to-point



Procrustes matching

$$\begin{aligned} & \text{minimize} && \|RP - QX\|_F^2 \\ & \text{s.t. } R^\top R = I, X \text{ a permutation} \end{aligned}$$

Question: is this problem convex? **NO**



STANDARD CONVEX PROBLEMS

Point-to-point



Procrustes matching

$$\begin{aligned} & \text{minimize } \|RP - QX\|_F^2 \\ & \text{s.t. } R^\top R = I, X \text{ a permutation} \end{aligned}$$

Question: is this problem convex? NO

Point Registration via Efficient Convex Relaxation

Maron, Dym, Kezurer, Kovalsky, Lipman (2016)

We will see the relaxation later...



OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX

Linear Programming
(LP)

Quadratic
Programming
(QP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

Semi-Definite
Programming
(SDP)

CONSTRAINED

UNCONSTRAINED

STANDARD CONVEX PROBLEMS

SECOND-ORDER CONE PROGRAMMING (SOCP)

minimize $f^\top x$

subject to $\|A_i x + b_i\|_2 \leq c_i^\top x + d_i$

$Fx = q$

inequality constraints are
second-order cones

find the variable x that...

minimizes the **linear objective function**

subject to the **cone and linear constraints**

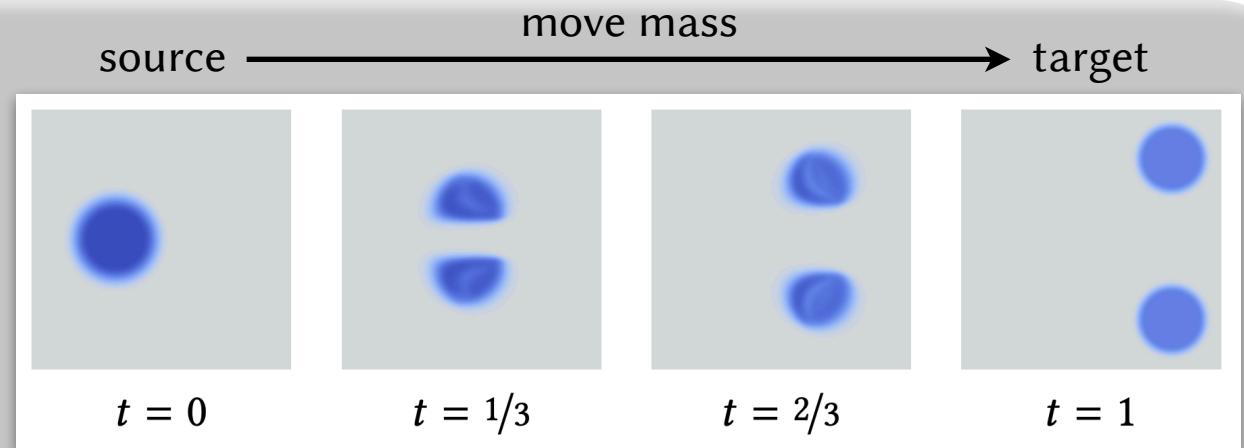
STANDARD CONVEX PROBLEMS

Optimal transport

Dynamical Optimal Transport on Discrete Surfaces

Lavenant, Claici, Chien, Solomon (2018)

$$\mathcal{W}(\mu_0, \mu_1) = \max_{\phi} \int \phi_1 d\mu_1 - \int \phi_0 d\mu_0$$
$$\text{s. t. } \partial_t \phi + \frac{1}{2} \|\nabla \phi\|^2 \leq 0$$



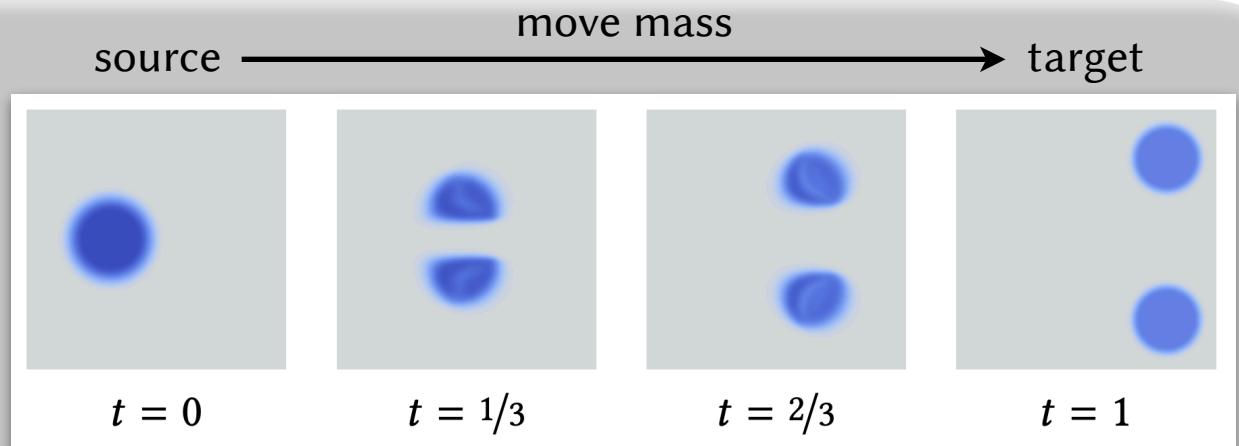
STANDARD CONVEX PROBLEMS

Optimal transport

Dynamical Optimal Transport on Discrete Surfaces

Lavenant, Claici, Chien, Solomon (2018)

$$\begin{aligned} \mathcal{W}(\mu_0, \mu_1) = \max_{\phi} & \int \phi_1 d\mu_1 - \int \phi_0 d\mu_0 \\ \text{s. t. } & \partial_t \phi + \frac{1}{2} \|\nabla \phi\|^2 \leq 0 \end{aligned}$$



quadratic constraint?

Quadratic
Inequality Constraints

$$x^\top Q x + c^\top x + d \leq 0$$

Cone constraints

$$\left\| Q^{1/2} x + \frac{1}{2} Q^{-1/2} c \right\|_2 \leq \left(\frac{1}{4} c^\top Q^{-1} c - d \right)^{1/2}$$

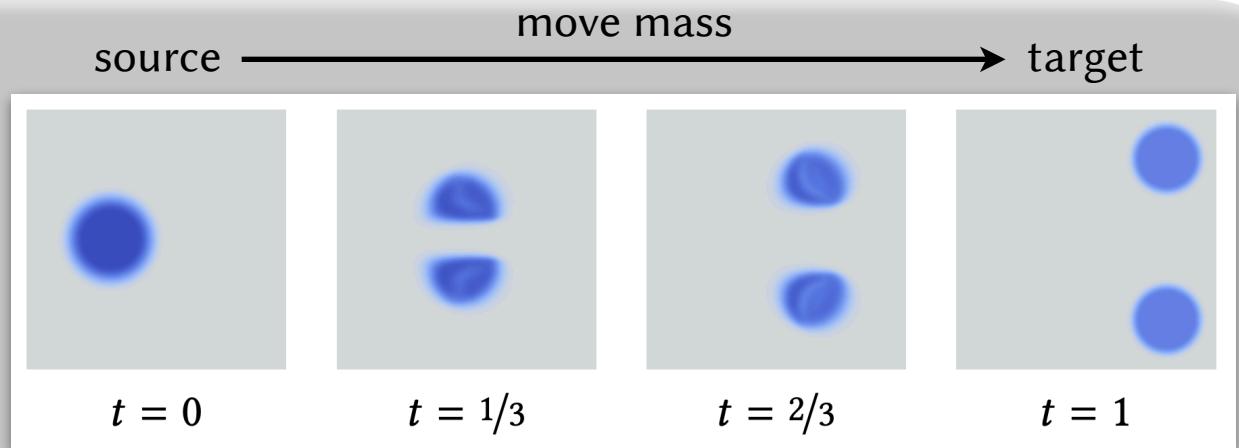
STANDARD CONVEX PROBLEMS

Optimal transport

Dynamical Optimal Transport on Discrete Surfaces

Lavenant, Claici, Chien, Solomon (2018)

$$\begin{aligned} \mathcal{W}(\mu_0, \mu_1) = \max_{\phi} & \int \phi_1 d\mu_1 - \int \phi_0 d\mu_0 \\ \text{s. t. } & \partial_t \phi + \frac{1}{2} \|\nabla \phi\|^2 \leq 0 \end{aligned}$$



We will see it again later...

Quadratic
Inequality Constraints

$$x^\top Q x + c^\top x + d \leq 0$$

quadratic constraint?

Cone constraints

$$\left\| Q^{1/2}x + \frac{1}{2}Q^{-1/2}c \right\|_2 \leq \left(\frac{1}{4}c^\top Q^{-1}c - d \right)^{1/2}$$

OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX

Linear Programming
(LP)

Quadratic
Programming
(QP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

Semi-Definite
Programming
(SDP)

CONSTRAINED

UNCONSTRAINED

STANDARD CONVEX PROBLEMS

SEMIDEFINITE PROGRAMMING (SDP)

minimize $\text{trace}(C^\top X)$

subject to $\text{trace}(A_i^\top X) \leq b_i$

$X \succeq 0$

constraint is a **linear
matrix inequality (LMI)**

find the variable x that...

minimizes the **linear objective function**

subject to the **LMI and PSD constraints**

STANDARD CONVEX PROBLEMS

Volumetric mesh deformations

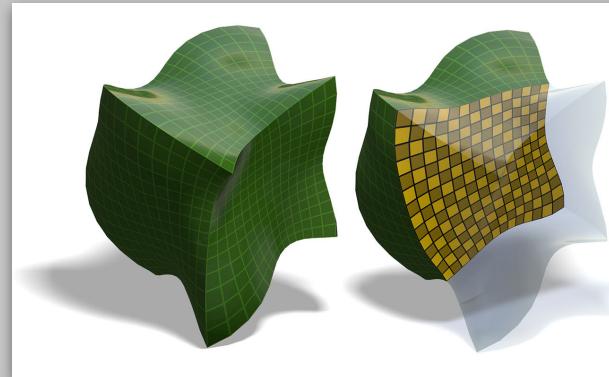
Controlling Singular Values with Semidefinite Programming

Kovalsky, Aigerman, Basri, Lipman (2014)

$$\min_{A \in \mathbb{R}^{n \times n}} \|A - B\|_F$$

$$\text{s.t. } \begin{pmatrix} \Gamma I & A \\ A^T & \Gamma I \end{pmatrix} \succeq 0$$

$$\frac{A + A^T}{2} \succeq \gamma I$$



OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX

Non-standard

Linear Programming
(LP)

Quadratic
Programming
(QP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

Semi-Definite
Programming
(SDP)

CONSTRAINED

UNCONSTRAINED

COFFEE BREAK

 Questions?

OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. PDE Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX

Non-standard

Linear Programming
(LP)

Quadratic
Programming
(QP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

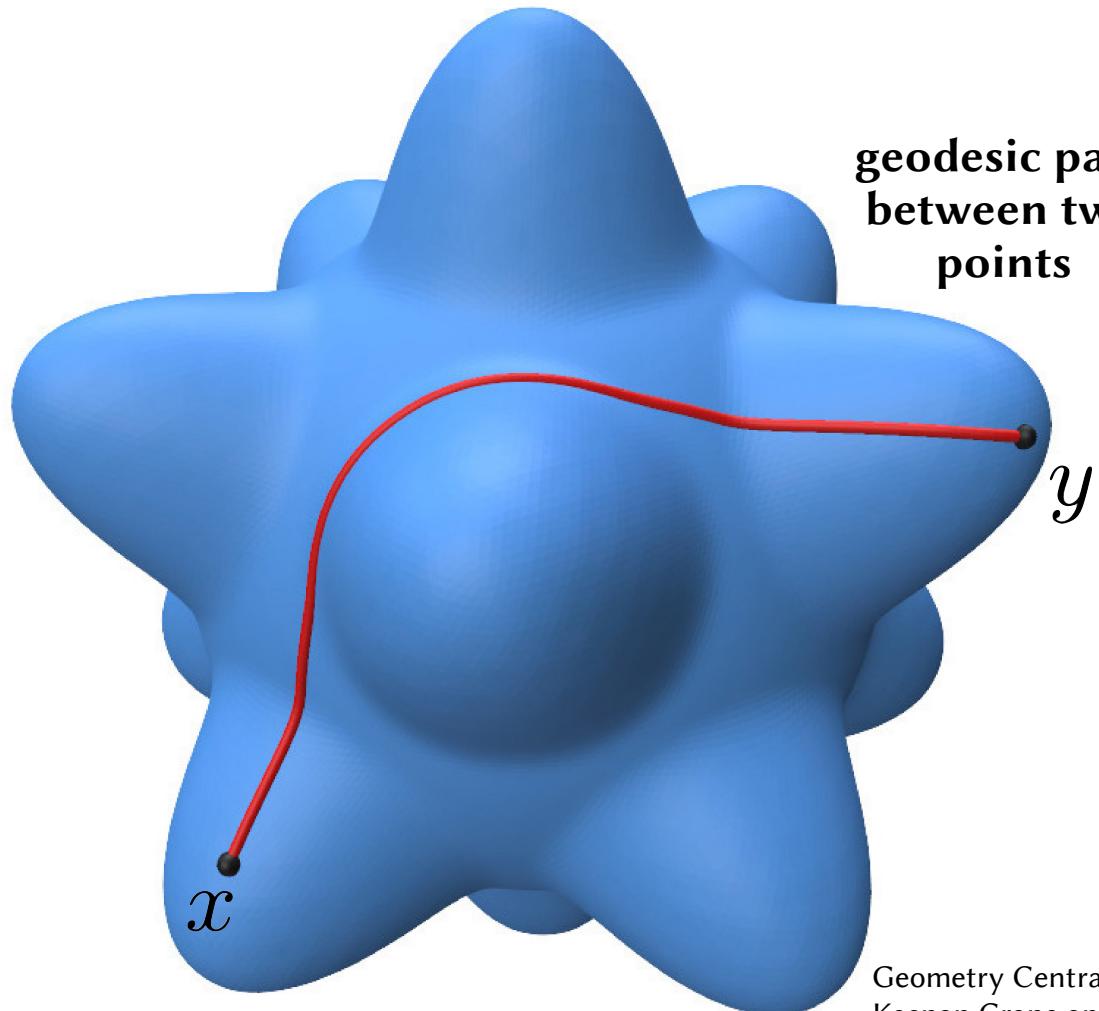
Semi-Definite
Programming
(SDP)

CONSTRAINED

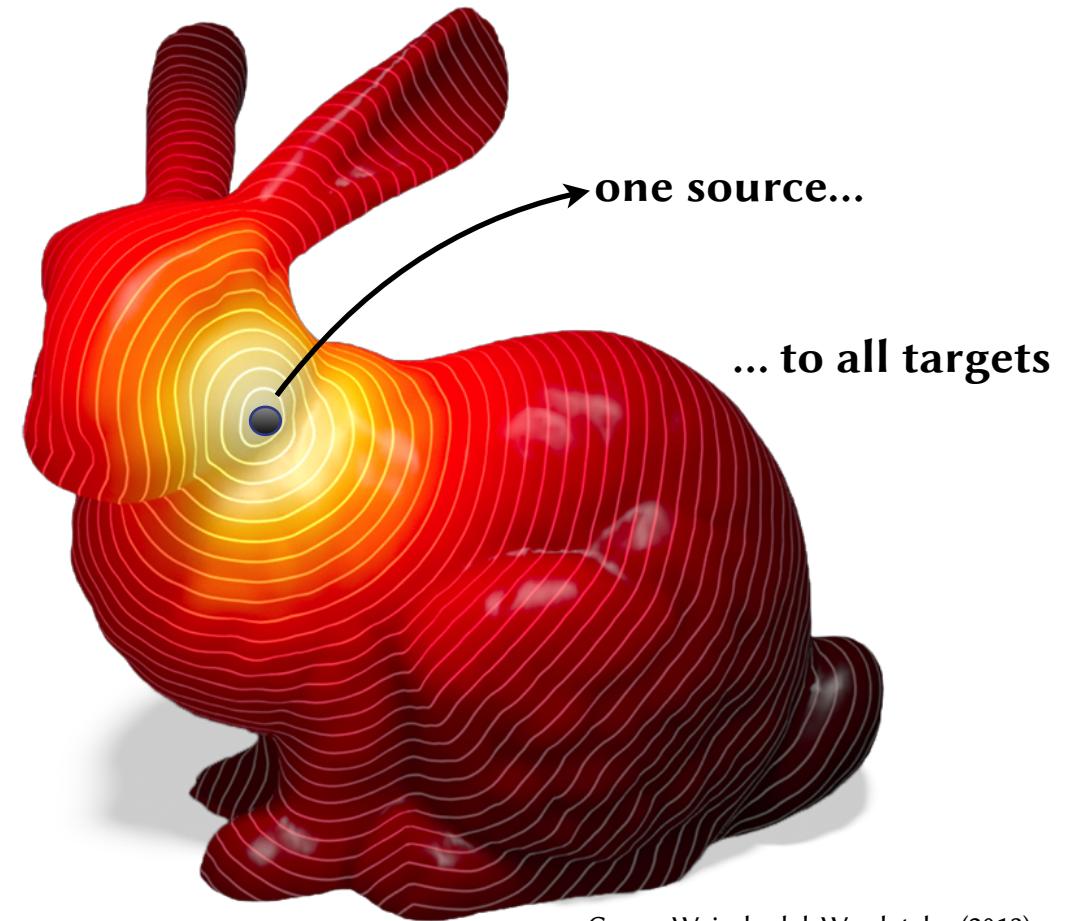
UNCONSTRAINED

VISCOSITY SOLUTIONS

Computing geodesic distances



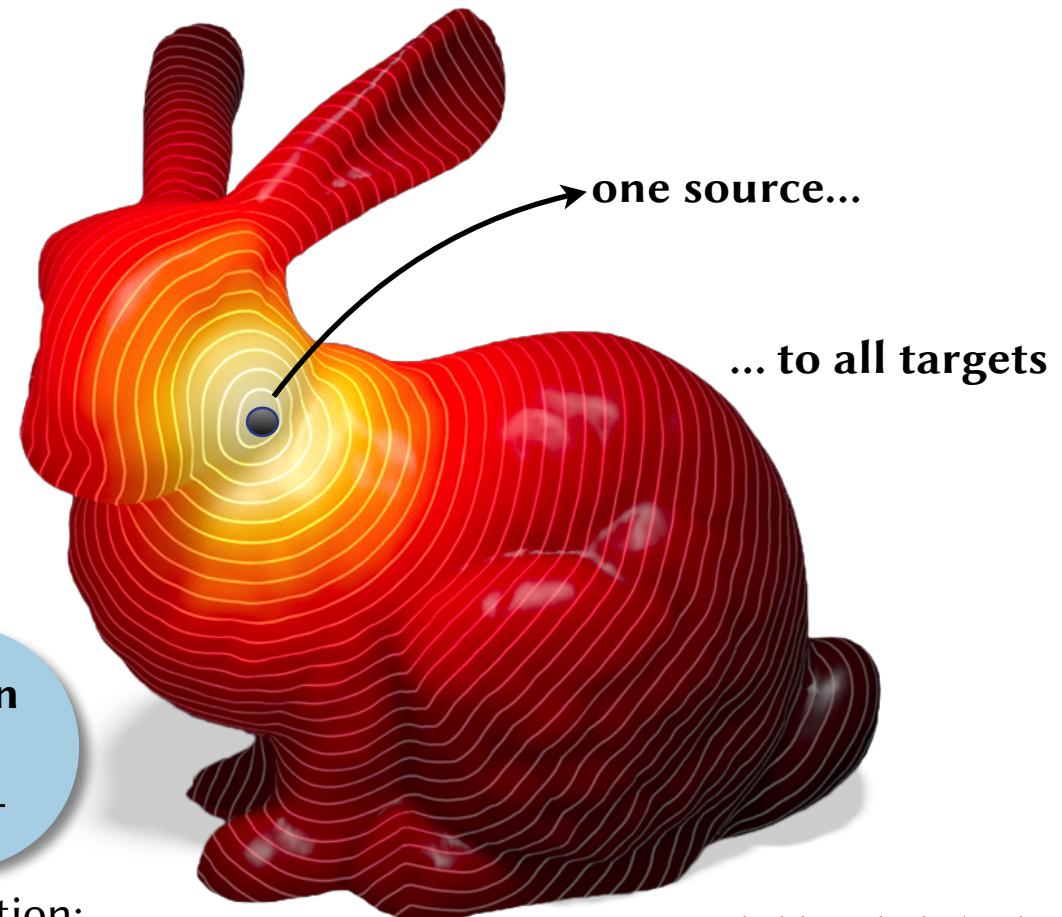
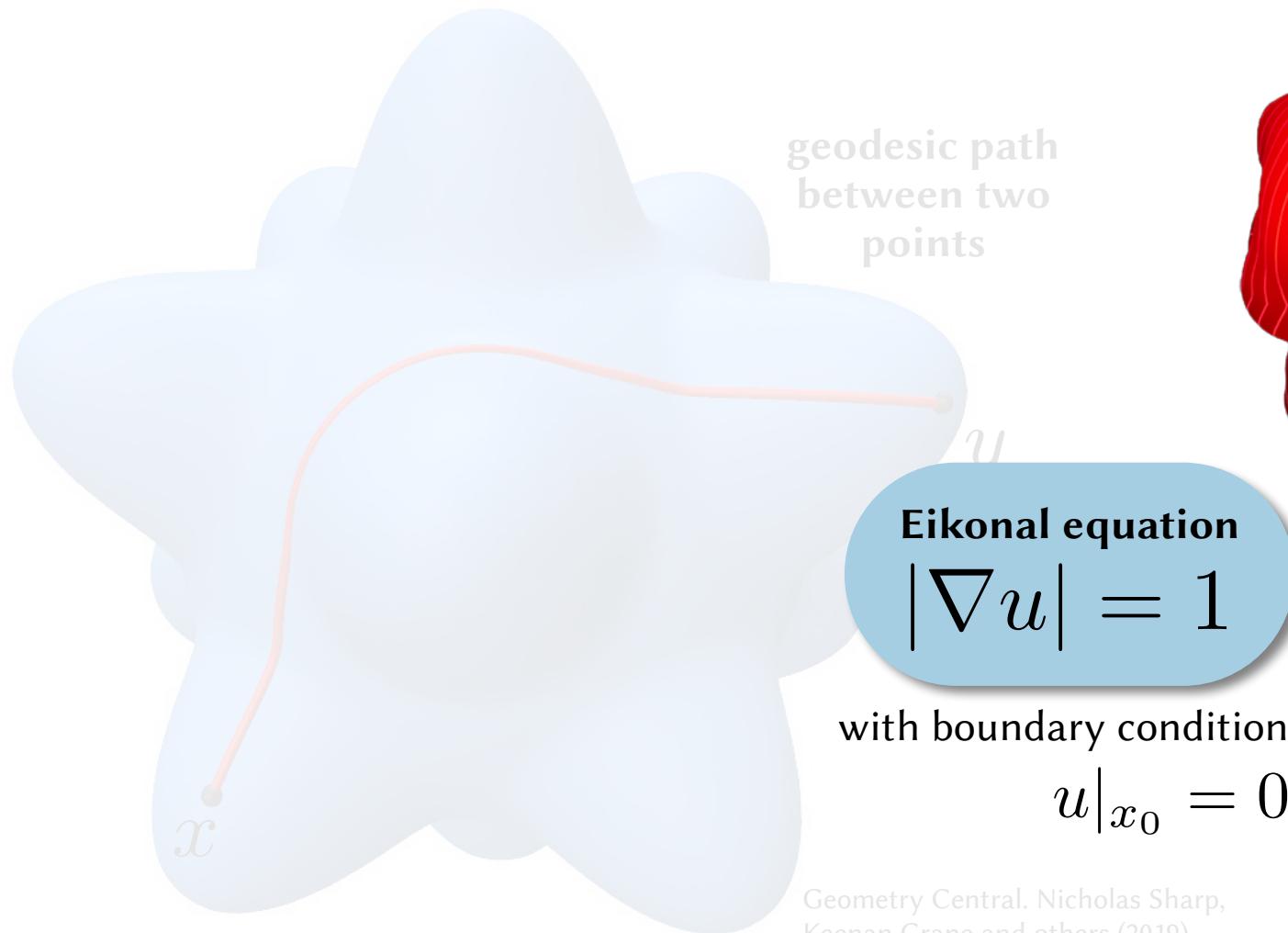
Geometry Central. Nicholas Sharp,
Keenan Crane and others (2019)



Crane, Weischedel, Wardetzky (2013)

VISCOSITY SOLUTIONS

Computing geodesic distances



VISCOSITY SOLUTIONS

Computing geodesic distances

An ADMM-based Scheme for Distance Function Approximation Belyaev & Fayolle (2020)

Numerical Algorithms (2020) 84:983–996
https://doi.org/10.1007/s11075-019-00789-5
ORIGINAL PAPER

An ADMM-based scheme for distance function approximation
Alexander Belyaev¹ · Pierre-Alain Fayolle²

Received: 6 March 2019 / Accepted: 24 July 2019 / Published online: 19 August 2019
© The Author(s) 2019

Abstract A novel variational problem for approximating the distance function (to a domain boundary) is proposed. It is shown that this problem can be efficiently solved by ADMM. A review of several other variational and PDE-based methods for distance function estimation is presented. Advantages of the proposed distance estimation method are demonstrated by numerical experiments. Applications of the method to the problems of surface curvature estimation and computing of a binary image are shown.

Keywords Distance function · Variational methods · Distance transform · Skeleton · Curvature

1 Introduction
Fast and accurate estimation of the distance to a surface (Ω) is important for a number of applications including redistricting level-set methods [17], wall distance models in turbulent homogeneous material modeling in computational mechanics [26], FEM extensions [2, 16], robotics [31] and meshing [31].

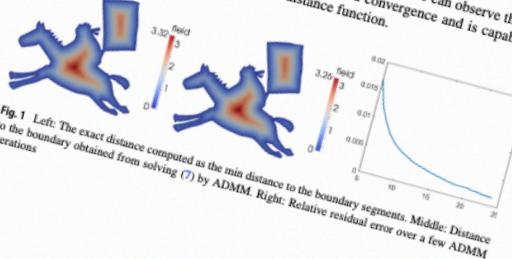
2 Numerical experiments: distance to a polygonal mesh using FEM and application to curvature computation
As described in Section 2, we use ADMM to minimize (7). This gives an iterative process, where the first step involves solving a Poisson problem of the form: $-\Delta\phi = f$. This Poisson problem is discretized and solved by the finite elements method. The computational domain bounded by $\partial\Omega$ is represented by a triangle mesh in 2D or a tetrahedral mesh in 3D. Linear basis functions are used at each node of the triangulation. The solution to the Poisson problem is obtained from numerically solving a linear system $A\Phi = b$, where the sparse matrix A , corresponding to a discretization of the Laplacian, is the same for each iteration of ADMM and can thus be prefactored (for example with the Cholesky decomposition).

3.1 Distance computation
Figure 1 illustrates the result obtained by our approach on a 2D polygonal domain with complex geometry. The left image visualizes the exact distance obtained by computing the minimum distance to any boundary segment. The middle image presents the distance computed by solving (7). The right image demonstrates how the relative residual error

$$\|\phi_{k+1} - \phi_k\|_2 / \|\phi_k\|_2$$

decreases with each iteration when solving (7) by ADMM. One can observe that (7) solved numerically by ADMM demonstrates a good convergence and is capable to deliver an accurate approximation of the distance function.

Fig. 1 Left: The exact distance computed as the min distance to the boundary segments. Middle: Distance to the boundary obtained from solving (7) by ADMM. Right: Relative residual error over a few ADMM iterations



Numerical Algorithms (2020) 84:983–996
987



VISCOSITY SOLUTIONS

Computing geodesic distances

An ADMM-based Scheme for Distance Function Approximation Belyaev & Fayolle (2020)



KEY INSIGHT

$$\max_{\phi} \int_{\Omega} \phi dx \text{ s.t. } |\nabla \phi| \leq 1, \phi|_{\partial\Omega} = 0$$

Numerical Algorithms (2020) 84:983–996
https://doi.org/10.1007/s11075-019-00789-5
ORIGINAL PAPER

An ADMM-based scheme for distance function approximation

Alexander Belyaev¹ · Pierre-Alain Fayolle²

Received: 6 March 2019 / Accepted: 24 July 2019 / Published online: 19 August 2019
© The Author(s) 2019

Abstract A novel variational problem for approximating the distance function (to a domain boundary) is proposed. It is shown that this problem can be efficiently solved by ADMM. A review of several other variational and PDE-based methods for distance function estimation is presented. Advantages of the proposed distance estimation method are demonstrated by numerical experiments. Applications of the method to the problems of surface curvature estimation and computing of a binary image are shown.

Keywords Distance function · Variational methods · Distance transform · Skeleton · Curvature

1 Introduction

Fast and accurate estimation of the distance to a surface (Ω) is important for a number of applications including region-level-set methods [17], wall distance models in turbulent homogeneous material modeling in computational mechanics [31], FEM extensions [2, 16], robotics [3], and meshing [31].

2 Numerical experiments: distance to a polygonal mesh using FEM and application to curvature computation

We consider first the problem of computing the distance function to $\partial\Omega$, a surface (curve in 2D), represented by a triangle mesh (a polygonal chain in 2D). As described in Section 2, we use ADMM to minimize (7). This gives an iterative process, where the first step involves solving a Poisson problem of the form: $-\Delta\phi = f$. This Poisson problem is discretized and solved by the finite elements method in 2D or a tetrahedral mesh in 3D. Linear basis functions are used at each node of the triangulation. The solution to the Poisson problem is obtained from numerically solving a linear system $A\Phi = b$, where the sparse matrix A , corresponding to a discretization of the Laplacian, is the same for each iteration of ADMM and can thus be prefactored (for example with the Cholesky decomposition).

3.1 Distance computation

Figure 1 illustrates the result obtained by our approach on a 2D polygonal domain with complex geometry. The left image visualizes the exact distance obtained by computing the minimum distance to any boundary segment. The middle image presents the distance computed by solving (7). The right image demonstrates how the relative residual error

$$\|\phi_{k+1} - \phi_k\|_2 / \|\phi_k\|_2$$

decreases with each iteration when solving (7) by ADMM. One can observe that (7) solved numerically by ADMM demonstrates a good convergence and is capable to deliver an accurate approximation of the distance function.

Fig. 1 Left: The exact distance computed as the min distance to the boundary segments. Middle: Distance to the boundary obtained from solving (7) by ADMM. Right: Relative residual error over a few ADMM iterations

Springer

VISCOSITY SOLUTIONS

Question: What if we want the distances to be regularized instead of approximate?

A Convex Optimization Framework for Regularized Geodesic Distances

Edelstein, Guillen, Solomon, Ben-Chen (2020)

Regularizer

$$\mathcal{E}(u) = \int_{\Omega} F(\nabla u(x), x) d \text{Vol}(x)$$

A Convex Optimization Framework for Regularized Geodesic Distances

Nestor Guillen
Texas State University
San Marcos, TX, USA
nestor@txstate.edu

Mirela Ben-Chen
Technion - Israel Institute of Technology
Haifa, Israel
mirela@cs.technion.ac.il

Michal Edelstein
Technion - Israel Institute of Technology
Haifa, Israel
smichale@cs.technion.ac.il

Justin Solomon
Massachusetts Institute of Technology (MIT)
Cambridge, MA, USA
jsolomon@mit.edu

Abstract
We propose a general convex optimization problem for computing regularized geodesic distances. We show that under mild conditions on the regularizer, the problem is well posed. We propose three different regularizers and provide analytical solutions in special cases, as well as corresponding efficient optimisation algorithms. Additionally, we show how to generalize the approach to the all-pairs case by formulating the problem on the product manifold, which leads to symmetric distances. Our regularized distances compare favorably to existing methods, in terms of robustness and ease of calibration.

ACM Reference Format:
Michal Edelstein, Nestor Guillen, Justin Solomon, and Mirela Ben-Chen. 2023. A Convex Optimization Framework for Regularized Geodesic Distances. In Special Interest Group on Computer Graphics and Interactive Techniques Conference Proceedings (SIGGRAPH '23 Conference Proceedings), August 6–10, 2023, Los Angeles, CA, USA. ACM, New York, NY, USA, 11 pages. <https://doi.org/10.1145/3588432.3591523>

1 INTRODUCTION
Distance computation is a central task in shape analysis. Distances are required for many downstream geometry processing applications, including shape correspondence, shape descriptors and remeshing. In many cases, however, exact geodesic distances are not required, and a distance-like function suffices. Moreover, it is often required to regularize the distance-like function to improve the performance of a downstream application.

The geometry processing community has proposed myriad methods for computing geodesic distances (Crane et al. 2020), including some regularized distances (Crane et al. 2013; Solomon et al. 2014). However, a unified framework, including a controlled and easily calibratable approach to regularization is still missing.

Keywords
geodesic distance, convex optimization, triangle meshes

Permissions to make digital or hard copies of part or all of this work for personal or classroom use in graduate without prior permission or payment are granted by copyright holder for third parties to copy or republish, provided that: (1) the full name of the copyright holder is given on the first page, (2) no derivative work is made, (3) no part of the material is sold in profit or commercial advantage, and (4) the full citation of this work must be honored on the first page. Copyright for shell-patch components of this work must be honored on the first page. Copyright for shell-patch components of this work must be honored on the first page. For other uses, contact the owner/author(s).
SIGGRAPH '23 Conference Proceedings, August 6–10, 2023, Los Angeles, CA, USA
© 2023 Copyright held by the owner/authors.
ACM ISBN 978-1-4503-7235-0.
<https://doi.org/10.1145/3588432.3591523>

Figure 1: Geodesic distances (a) may not have desired properties such as smoothness. We present here three examples of regularizers: (b,c) smoothness, (d,e) alignment to a vector field, and (e) boundary invariance.

Figure 13: Dirichlet regularized volumetric distances. (a) The input tetrahedral mesh. (b,c) Two cuts showing the distance to a point on the shoulder. (d,e) Distance to the boundary, where (d) is more smoothed than (c), i.e. has a larger α value.

Figure 7: Scale invariance. While the distances are different between the uniformly scaled models, the area of the smoothness (where the norm of the gradient is not 1) is similar for all meshes. See the text for details.

VISCOSITY SOLUTIONS

Question: What if we want the distances to be regularized instead of approximate?

A Convex Optimization Framework for Regularized Geodesic Distances
Edelstein, Guillen, Solomon, Ben-Chen (2023)

Regularizer

$$\mathcal{E}(u) = \int_{\Omega} F(\nabla u(x), x) d \text{Vol}(x)$$



VISCOSITY SOLUTIONS

Computing regularized geodesic distances.

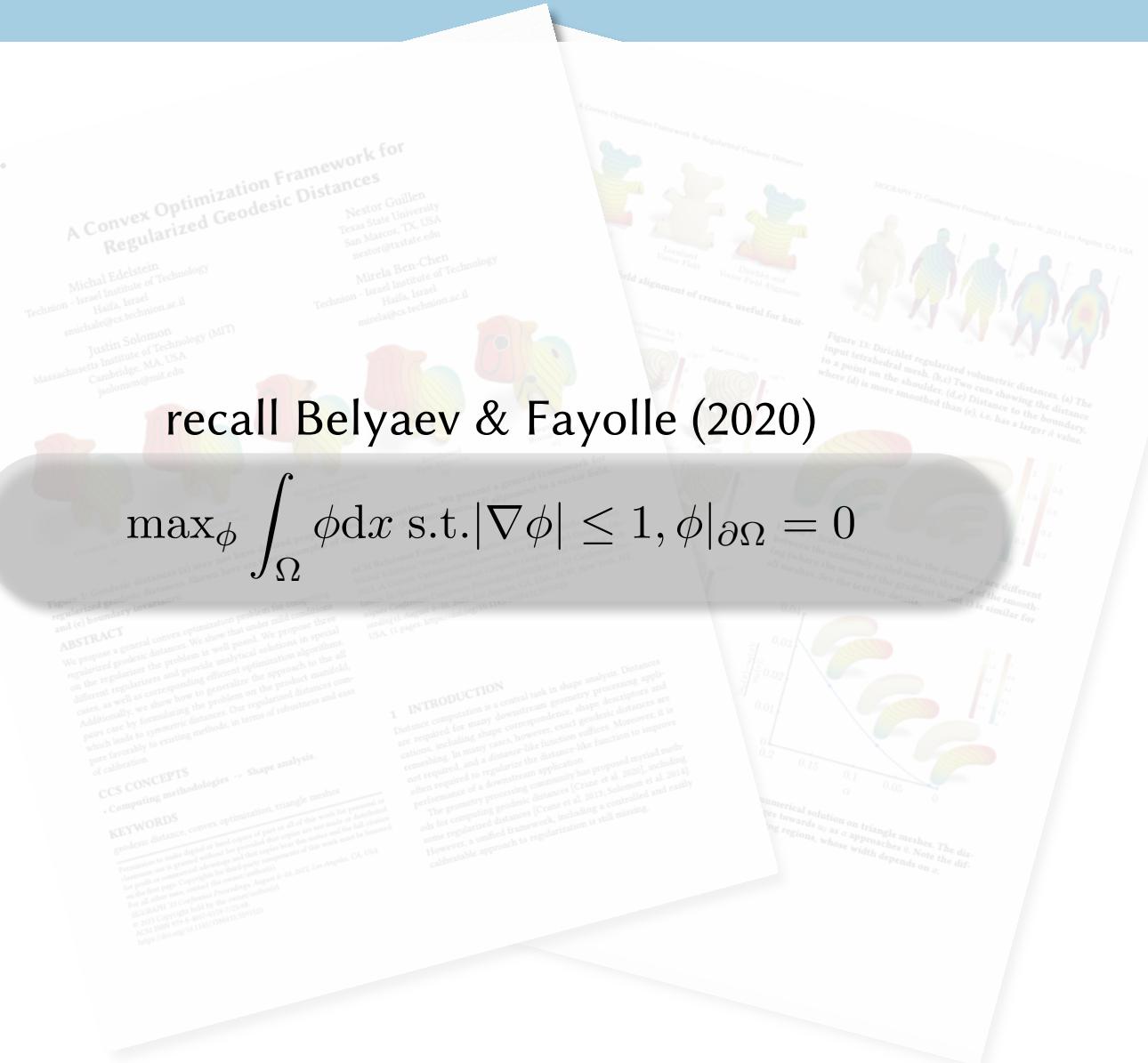
A Convex Optimization Framework for Regularized Geodesic Distances Edelstein, Guillen, Solomon, Ben-Chen (2023)



KEY INSIGHT

$$\min_u \mathcal{E}(u) - \int_{\mathcal{M}} u dx \\ \text{s.t.} |\nabla u| \leq 1, u|_{\partial\mathcal{M}} = 0$$

Same relaxation strategy: supersolutions



VISCOSITY SOLUTIONS

Question: What if we want to solve more general time-evolving PDE?

A Framework for Solving Parabolic Partial Differential Equations on Discrete Domains

Mattos Da Silva, Stein, Solomon (2024)

Second-order parabolic PDE

$$\frac{\partial u}{\partial t} + H(x, u, \nabla u) = \varepsilon \Delta u$$

A Framework for Solving Parabolic Partial Differential Equations on Discrete Domains

LETICIA MATTOS DA SILVA, Massachusetts Institute of Technology, USA
ODED STEIN, University of Southern California, USA
JUSTIN SOLOMON, Massachusetts Institute of Technology, USA

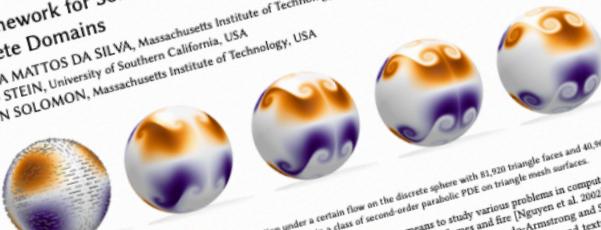


Fig. 1. We show the time evolution of the Fokker-Planck equation under a certain flow on the discrete sphere with 81,920 triangle faces and 40,962 vertices, obtained using our framework. Our method can be used for any equation in a class of second-order parabolic PDE on triangle mesh surfaces.

We introduce a framework for solving a class of parabolic partial differential equations on triangle mesh surfaces, including the Hamilton-Jacobi equation and the Fokker-Planck equation. PDE in this class often have nonlinear or stiff terms that cannot be resolved with standard methods on curved triangle meshes. To address this challenge, we leverage a splitting integrator combined with a convex optimization step to solve these PDE. Our framework can be used to compute entropic approximations of optimal transport沉tances on geometric domains, overcoming the numerical limitations of the state-of-the-art method. In addition, we demonstrate the versatility of our method to capture complex flows in geometry processing.

CSCS Concepts: Mathematics of computing → Partial differential equations; Computing methodologies → Shape analysis.

Additional Key Words and Phrases: diffusion, optimal transportation, Hamilton-Jacobi, Fokker-Planck

1. INTRODUCTION

The analysis of partial differential equations (PDE) is a ubiquitous technique in computer graphics, geometry processing, and adjacent fields. In particular, parabolic PDE describe a wide variety of phenomena. For example, instances of the Hamilton-Jacobi equation model the time evolution of front propagation and the evolution of functions undergoing nonlinear diffusion. As another example, the Fokker-Planck equation describes the evolution of density functions driven by stochastic processes. Each of these equations has a long history as means to study various problems in computer graphics, including modeling flames and fire [Nguyen et al. 2002], stochastic heat kernel estimation [Amentano-Armstrong and Siddiqi 2017], medial axis detection [Du and Qin 2004], and texture synthesis [Wilkin and Kaas 1991]. Hence, methods to solve this class of PDE over geometric domains are central in geometry processing.

Myriad numerical algorithms have been proposed for solving PDE in geometry processing. Unfortunately, the most popular algorithms are unsuitable for important regimes, such as capturing infinitesimal axis-symmetric flows [Amentano-Armstrong and Siddiqi 2017] or nonlinear phenomena. An interesting example involves the convolutional Wasserstein distance method for barcode computation [Solomon et al. 2015]. The aforementioned method is built on tiny amounts of diffusion, but relies on heuristics for choosing diffusion times. If the diffusion time step is too small, then the method fails to numerical inaccuracies, and if the step is too large, then the method results in approximations that quantitatively differ from the true type of flow. In this case, the challenge is that explicit integrators used for these PDE require time step restrictions to avoid numerical instability, and implicit integration schemes are not equivalent to solving a single linear system of equations, turning out to be too expensive.

To address these challenges, we propose a framework leveraging a splitting integration strategy and an appropriate spatial discretization to solve parabolic PDE over discrete geometric domains. The splitting allows us to leverage the implicit integration of a well-known PDE, the heat equation, and use a convex relaxation to deal with the challenging piece of the parabolic PDE. Empirically, our method overcomes limitations presented in section §4.1.4.

Another example involves nonlinear heat diffusion, the nonlinear G-equation, and the Fokker-Planck equation, all of which we solve efficiently on a variety of domains and time steps.

4 · Mattos Da Silva et al.

Proof. Let $u = \log v$, then this is a straightforward application of the chain rule:

$$\begin{aligned}\frac{\partial p}{\partial t} - \|\nabla u\|_2^2 &= \Delta u \\ \frac{\partial p}{\partial t} - \frac{1}{\varepsilon^2} \|\nabla v\|_2^2 &= \nabla \cdot (\frac{1}{\varepsilon^2} \nabla v) \\ &= -\frac{1}{\varepsilon^2} (\nabla v, \nabla v)_2 + \frac{1}{\varepsilon^2} \Delta v\end{aligned}$$

□
Fig. 3.1 only applies to strictly positive functions v . It is sufficient to check $v(x, 0) > 0$ and clearly $\partial v/\partial t \geq \Delta v$.

This paper is that the same numerical scheme can be generalized to handle the form (1) given certain continuity and H . In particular, we assume

u is continuous on $T^*M \times \mathbb{R}$. Then, whenever $w \leq u$, we have $H(x, q, u) - H(x, q, w)\|_2 \leq C$, where C is a suitable constant of the function H =

□
Fig. 4. These typical examples of vector fields Φ used when evolving second-order parabolic PDE that involve terms with vector fields.

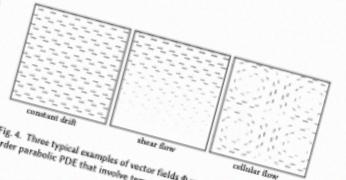


Fig. 4. These typical examples of vector fields Φ used when evolving second-order parabolic PDE that involve terms with vector fields.

Fig. 5. Time evolution of the Fokker-Planck equation (6) on a 100×100 triangle grid, obtained using our method, under constant drift (top), shear flow (middle), and no drift (bottom). See Figure 4 for an illustration of

where $W(t)$ is a Wiener process. We refer the reader to [Medved et al. 2020] for a review on the relationship between equations (6)-(7). The vector field Φ is typically known as the drift vector and ε as the diffusion coefficient, but we will call the latter the viscosity parameter for consistency throughout this paper.

3.3 Viscosity Solutions
We introduce a few definitions from [Crandall and Lions 1983] in the study of weak solutions to Hamilton-Jacobi equations. These definitions will be necessary in the proofs presented in section §4.1.4. We refer the reader to [Crandall et al. 1992] for a complete introduction to the theory of viscosity solutions and their applications to PDE.

Fig. 3.2 (Viscosity subsolutions, and resp., supersolutions).
An open subset in the manifold M . A function $u: V \rightarrow \mathbb{R}$ is a sub/super solution (resp., supersolution) of $\frac{\partial u}{\partial t} + H(x, q, u) = 0$ if the function $\varphi: V \rightarrow \mathbb{R}$ and every point $x \in V$ such

function $\varphi: V \rightarrow \mathbb{R}$ (resp., minimum) at x , we have a local maximum (resp., minimum) at x , we have

≤ 0 (resp., ≥ 0) if $\varphi + H(x, V\varphi, \varphi) \geq 0$.

viscosity solution). A function $u: V \rightarrow \mathbb{R}$ is a viscosity solution if it is both a sub/super

VISCOSITY SOLUTIONS

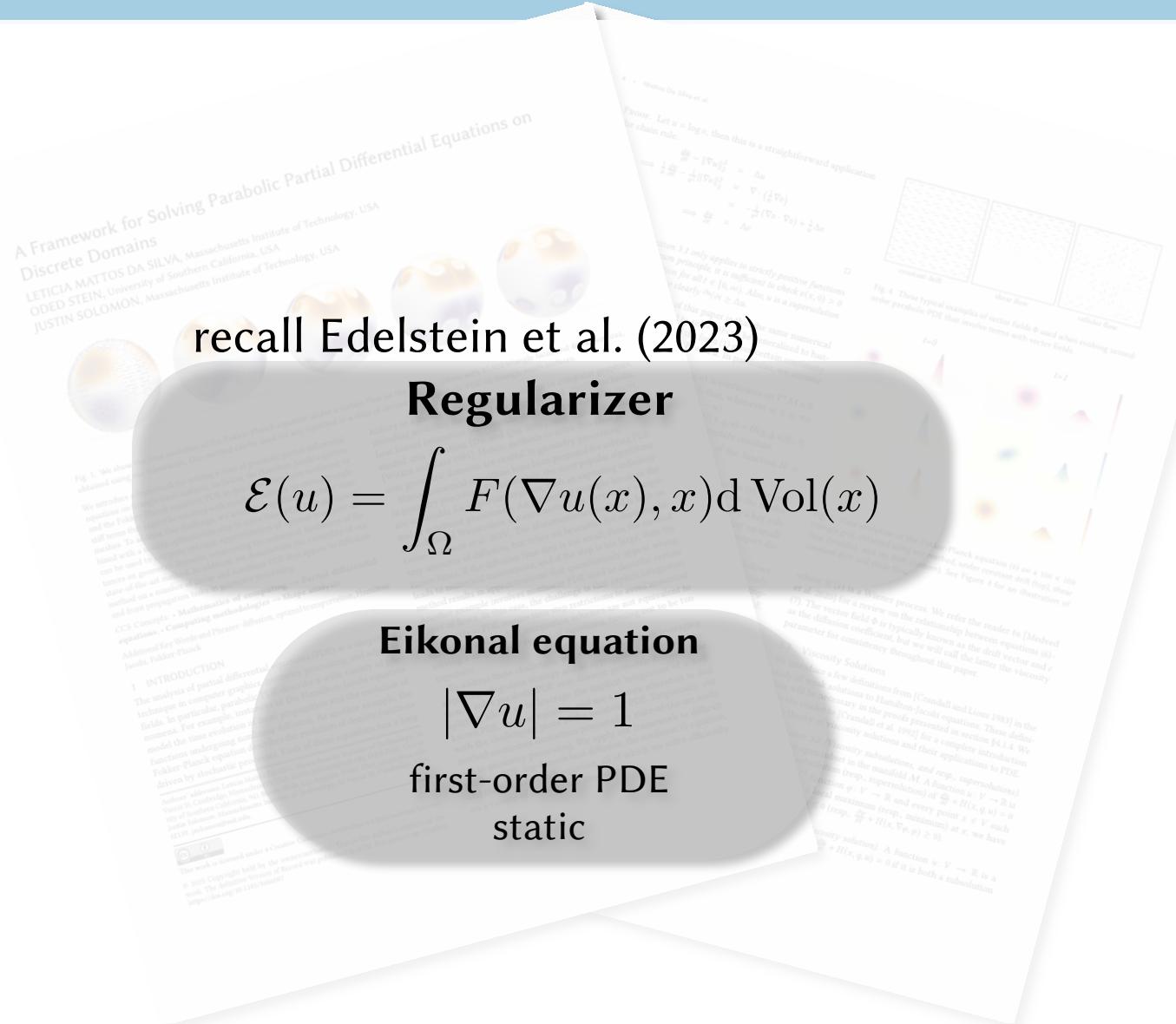
Question: What if we want to solve more general time-evolving PDE?

A Framework for Solving Parabolic Partial Differential Equations on Discrete Domains

Mattos Da Silva, Stein, Solomon (2024)

Second-order parabolic PDE

$$\frac{\partial u}{\partial t} + H(x, u, \nabla u) = \varepsilon \Delta u$$



VISCOSITY SOLUTIONS

Question: What if we want to solve more general time-evolving PDE?

A Framework for Solving Parabolic Partial Differential Equations on Discrete Domains

Mattos Da Silva, Stein, Solomon (2024)

Second-order parabolic PDE

$$\frac{\partial u}{\partial t} + H(x, u, \nabla u) = \varepsilon \Delta u$$

Same* relaxation strategy: supersolutions

first, apply splitting technique.

1. $\frac{\partial u}{\partial t} = \varepsilon \Delta u$ Linear solve

2. $\frac{\partial u}{\partial t} + H(x, \nabla u, u) = 0$ Convex relaxation

3. $\frac{\partial u}{\partial t} = \varepsilon \Delta u$ Linear solve, again

VISCOSITY SOLUTIONS

Question: What if we want to solve more general time-evolving PDE?



KEY INSIGHT

$$\begin{aligned} & \arg \min_u \int_{\mathcal{M}} u(x) \, d\text{Vol}(x) \\ \text{subject to } & \frac{1}{h}(u - u_n^{(1)}) + H(x, \nabla u, u) \geq 0 \\ & \text{for all } x \in \mathcal{M}. \end{aligned}$$

Second-order parabolic PDE

$$\frac{\partial u}{\partial t} + H(x, u, \nabla u) = \varepsilon \Delta u$$

Same* relaxation strategy: supersolutions

first, apply splitting technique.

$$1. \frac{\partial u}{\partial t} = \varepsilon \Delta u \text{ Linear solve}$$

$$2. \frac{\partial u}{\partial t} + H(x, \nabla u, u) = 0 \text{ Convex relaxation}$$

$$3. \frac{\partial u}{\partial t} = \varepsilon \Delta u \text{ Linear solve, again}$$

OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX

Non-standard

Linear Programming
(LP)

Quadratic
Programming
(QP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

Semi-Definite
Programming
(SDP)

CONSTRAINED

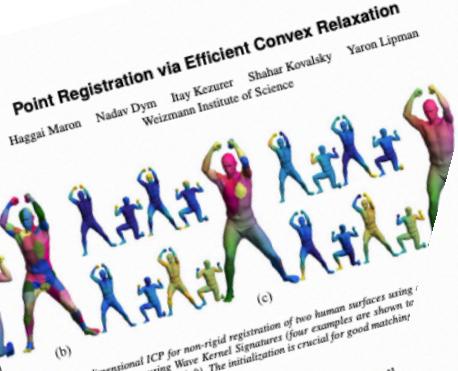
UNCONSTRAINED

MAPPING PROBLEMS

Point-to-point correspondence

Point Registration via Efficient Convex Relaxation

Maron, Dym, Kezurer, Kovalsky, Lipman (2016)



Point Registration via Efficient Convex Relaxation
Haggai Maron Nadav Dym Itay Kezurer Shahar Kovalsky Yaron Lipman
Weizmann Institute of Science

Figure 1: Initializing high-dimensional ICP for non-rigid registration of two human surfaces using random initialization; (c) initialization using Wave Kernel Signatures (four examples are shown) with correspondence (four examples are shown to its left). The initialization is crucial for good matches using PM-SDP which provides comparable result to (d). The initialization is crucial for good matches using PM-SDP which provides comparable result to (d).

Abstract

Point cloud registration is a fundamental task in computer graphics, and more specifically in rigid and non-rigid shape matching. The rigid shape matching problem can be formulated as the problem of simultaneously aligning and labeling two point clouds in 3D so that they are as similar as possible. We name this problem the Procrustes matching (PM) problem. The non-rigid shape matching problem can be formulated as a higher dimensional PM problem using the functional maps method. High dimensional PM problems are difficult non-convex problems which currently can only be solved locally using iterative closest point (ICP) algorithms or similar methods. Good initialization is crucial for obtaining a good solution.

We introduce a novel and efficient convex SDP (semidefinite programming) relaxation for the PM problem. The algorithm is guaranteed to return a correct global solution of the problem when matching two isometric shapes which are either asymmetric or laterally symmetric.

We show our algorithm gives state of the art results on shape matching datasets. We also show that our algorithm is state of the art results for anatomical classification of 3D medical collections.

Keywords: Point registration, Shape matching, C
Concepts: Computing methodologies → Shape matching, C

Permission to make digital or hard copies of all or portions of this work is granted without fee for personal classroom use and the flat fee of \$10.00 per article for others by others than ACM or its agents or the editor of a journal in which the article appears, provided that the full terms of this notice and the copyright notice are given, and that the full reference appears in a journal or other publication. This permission is granted on the condition that the copy is not resold in any form, in whole or in part, without the express written permission of ACM or its agents or the editor of the journal in which the article appears. SIGGRAPH '16 Technical Paper, July 24–28, 2016, Boston, MA, USA. © 2016 ACM. 978-1-4503-4279-3/16/07...
ISBN: 978-1-4503-4279-3/16/07

Local minimization. Since the feasible set of PM-SDP is larger than the feasible set of PM, the solution of PM-SDP in general may not contain orthogonal and the permutation matrices. We therefore project the solutions onto the feasible set of PM. We do this by locally minimizing PM using the output of PM-SDP to initialize the algorithm. The local minimization is done using an ICP-like algorithm which interleaves between minimizing over one of the matrices R , X while holding the other constant: fixing R results in a linear program, while for a fixed X there exists a closed-form solution [Gower and Dijksterhuis, 2004]. In Figure 4 we illustrate the doubly stochastic matrix X as computed from the PM-SDP relaxation and the permutation achieved after the projection. As shown, the PM-SDP output is already very similar to the projection result demonstrating the tightness of the PM-SDP relaxation. More details are in Appendix A.

The local minimization following the PM-SDP relaxation allows generalizing Theorem 2 to the inexact case:

Corollary 1. Let P, Q be point clouds satisfying the conditions of Theorem 5, and let P^*, Q^* be sufficiently small perturbations of P, Q . Then PM-SDP followed by the local minimization returns the unique (global) solution of PM for P^*, Q^* .

A quantitative evaluation of the optimality of PM-SDP is given in Figure 5, (e-f). We ran 80 random experiments for $d = 3$ and $d = 5$ with noise level $\sigma = 0.1$ and measured the optimal objective value achieved by PM-SDP in comparison to the global minimum and median value of the objective values found by the exhaustive algorithm. For visualization, we subtracted the value of the optimal value from all of the results. PM-SDP (black line) usually returns the optimal value (green) and always returns a better result than the median objective value (blue) of the exhaustive algorithm.

5 Evaluation

We test the tightness of the PM-SDP relaxation by comparing it to the ground truth obtained from an exhaustive brute sampling algorithm. The latter is only capable for low dimensional d , and we choose $d = 3$. The exhaustive algorithm densely samples a 10k points from a uniform distribution over $C(3)$ and uses each sample R_i as an initialization for the local minimization algorithm described above.

In Figure 5 we compare the histograms of optimal values achieved by the exhaustive sampling algorithm (in red) to the energy achieved by PM-SDP (in blue). The data for this experiment was generated by randomizing $Q \in \mathbb{R}^{3 \times n}$ according to a uniform distribution on $[0, 1]$, and setting $P = R^T Q X + \epsilon$, with $X \in \mathbb{U}_{d \times d}$, $R \in C(3)$ and noise $\epsilon \sim N_{\mu=0, \sigma=2}$. (a-d) show the results of a few typical runs with increasing amount of noise $\sigma = 0, 0.05, 0.1, 0.2$. We note that the number of local (sub-global) minima for the exhaustive sampling is surprisingly high for example, at noise level $\sigma = 0.1$ we found more than 1000 local energy minima. Additionally, the experiment in (a) verifies our theoretical exactness result as can be seen by the fact that the blue point achieves the left most value of the red histogram. When the noise level is low to

point $q \in C(3)$, $(\Phi_q^T(p), i = 1, \dots, d)$ and then assigning the d coordinates $(\Phi_q^T(p), i, q \in C(3))$ to each point $p \in P$, and similarly to every

Current approaches using this formulation, solve the resulting high dimensional PM problem using an ICP-type iterative algorithm; as this problem is known to have a vast number of local minima even for $d = 3$ (see Figure 5 (a-d)), initialization is crucial.

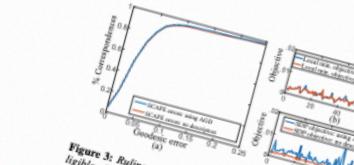


Figure 3: Ruling out matches with the AGD descriptor has a negligible effect on the quality of the relaxation: (a) depicts the results of PM-SDP on SCAPE dataset (Aguech et al., 2005) with and without AGD to rule out unlikely matches; (b) the objective after local minimization; and (c) objective value achieved by PM-SDP. The PM-SDP objective is lower for the unpruned version while the rest of the results are equivalent for both versions.

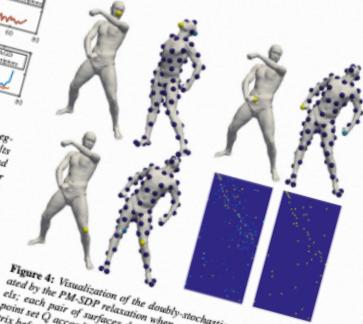


Figure 4: Visualization of the doubly-stochastic map X as generated by the PM-SDP relaxation when comparing two SCAPE meshes. Each pair of surfaces depicts a column of X by coloring the point set Q according to the corresponding value in X ; the X matrix before and after projection on the permutations is shown at the bottom-right.

Figure 5: (e-f) We ran 80 random experiments for $d = 3$ and $d = 5$ with noise level $\sigma = 0.1$ and measured the optimal objective value achieved by PM-SDP in comparison to the global minimum and median value of the objective values found by the exhaustive algorithm. For visualization, we subtracted the value of the optimal value from all of the results. PM-SDP (black line) usually returns the optimal value (green) and always returns a better result than the median objective value (blue) of the exhaustive algorithm.

6 Applications

6.1 Functional maps

The main application of our algorithm is non-rigid shape matching. We pose this problem as a high dimensional polygonal transformation in higher dimensional space. More specifically, we sample k points on the first shape and n points on the second shape uniformly, using farthest point sampling (Eldar et al., 1997) initialized with embedded $P, Q \in \mathbb{R}^d$. The embedding is done by first computing the first d eigenfunctions of the cut-weight Laplace-Beltrami (LB) operator (Pajek and Pajek, 1993) on each of the surfaces $\{\Phi_i^T(p)\}_{i=1}^d, \{\Phi_d^T(p)\}_{i=1}^d$ and then assigning the d coordinates

point $q \in C(3)$ to each point $p \in P$, and similarly to every point $q \in C(3)$. Following [Ovsjanikov et al., 2012] we pose this problem as a high dimensional polygonal transformation in higher dimensional space. More specifically, we sample k points on the first shape and n points on the second shape uniformly, using farthest point sampling (Eldar et al., 1997) initialized with embedded $P, Q \in \mathbb{R}^d$. The embedding is done by first computing the first d eigenfunctions of the cut-weight Laplace-Beltrami (LB) operator (Pajek and Pajek, 1993) on each of the surfaces $\{\Phi_i^T(p)\}_{i=1}^d, \{\Phi_d^T(p)\}_{i=1}^d$ and then assigning the d coordinates point $q \in C(3)$ to each point $p \in P$, and similarly to every point $q \in C(3)$. Current approaches using this formulation, solve the resulting high dimensional PM problem using an ICP-type iterative algorithm; as this problem is known to have a vast number of local minima even for $d = 3$ (see Figure 5 (a-d)), initialization is crucial.

MAPPING PROBLEMS

Relaxing QCQP into SDP-LC

Procrustes matching

$$\begin{aligned} & \text{minimize} && \|RP - QX\|_F^2 \\ & \text{s.t. } R^\top R = I, X \text{ a permutation} \end{aligned}$$



KEY INSIGHT

non-convex QCQP can be relaxed to SDP at the cost of exactness

MAPPING PROBLEMS

Relaxing QCQP into SDP-LC

Procrustes matching

$$\begin{aligned} & \text{minimize} \|RP - QX\|_F^2 \\ & \text{s.t. } R^\top R = I, X \text{ a permutation} \end{aligned}$$



KEY INSIGHT

non-convex QCQP can be relaxed to SDP at the cost of exactness

lots of steps... let's stick to the big ideas!

introduce new variable

$$Z = zz^\top, z = [\text{vec}(R), \text{vec}(X)]$$

and rewrite in standard quadratic

constraints are then linear, but also introduces PSD and $\text{rank}(1)$ constraints

drop the $\text{rank}(1)$ constraints

MAPPING PROBLEMS

Question: What if instead of relaxing the Procrustes matching, we relax the requirement that correspondences be exact point-to-point bijections?



lesson:

another strategy is to relax the problem *itself* to obtain an already convex formulation instead of performing a proper relaxation.

MAPPING PROBLEMS

Soft Maps Between Surfaces

Solomon, Nguyen, Butscher, Ben-Chen, Guibas (2012)



KEY INSIGHT

Replace the “all or nothing” bijection constraints with linear probabilistic ones.

allow mass-splitting

The figure shows a white 3D pig model on the left and a purple pig model on the right. Colored patches on the white pig map onto the purple pig, demonstrating the non-injective nature of the mapping where multiple patches from the source map to the same target area.

Volume 31 (2012), Number 5

Soft Maps Between Surfaces

Justin Solomon, Andy Nguyen, Adrian Butscher, Mirela Ben-Chen, Leonidas Guibas
Geometric Computing Group, Stanford University

Figure 1: Soft maps from one model to another computed using our optimization technique. The colored patches on the leftmost model are mapped to the colored distributions over the models on the right. These soft maps acknowledge discrete left-right point-to-point correspondences and front-back symmetries as well as localized ambiguities including slippage along the pig's back.

Abstract
The problem of mapping between two non-isometric surfaces admits ambiguities on both local and global scales. For instance, symmetries can make it possible for multiple maps to be equally acceptable, and stretching, slippage, and compression introduce difficulties deciding exactly where each point should go. Since most algorithms for point-to-point and even sparse mapping struggle to resolve these ambiguities, in this paper we introduce soft maps, a probabilistic relaxation of point-to-point correspondence that explicitly incorporates ambiguities in the mapping process. In addition to computing a continuous theory of soft maps, we show how they can be represented using probability matrices and computed for given pairs of surfaces through a convex optimization explicitly trading off between continuity, conformity to geometric descriptors, and spread. Given that our correspondences are encoded in matrix form, we also illustrate how low-rank approximation and other linear algebraic tools can be used to analyze, simplify, and represent both individual and collections of soft maps.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric algorithms, languages, and systems

Figure 6: (a) The first eight SVD basis vectors from the first map in Figure 5, sorted by decreasing singular value, and (b) the basis vectors “untangled” using [SCHWIG11] to better show their support. Bases are colored using the scale below the image. They respect symmetries and are spread depending on the usefulness of ϕ for mapping each patch.

Figure 7: A plot of the singular values from our decomposition. These singular values have a relatively long tail, so low-rank approximations of A can be obtained by projecting onto a reduced basis; Figure 8 shows such a projection onto the first 10 basis vectors.

Figure 8: A 3D rendering of a white pig model being mapped onto a purple pig model using the basis from Figure 6; it is visible that the original map (Figure 1) is more accurate than the basis.

© 2012 The Author(s). Computer Graphics Forum © 2012 The Eurographics Association and Blackwell Publishing Ltd, 0146-080X/12 \$10.00
DOI: 10.1111/j.1467-8659.2012.03748.x

OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX

Non-standard

Linear Programming
(LP)

Quadratic
Programming
(QP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

Semi-Definite
Programming
(SDP)

CONSTRAINED

UNCONSTRAINED

OPTIMAL TRANSPORT

Optimal transport

Monge's original
formulation

$$\int_M \frac{1}{2} c(x, T(x)) d\mu_0(x)$$

s. t. $T_{\#}\mu_0 = \mu_1$

NON-CONVEX!



seeking a deterministic map T
that maps points to points.

OPTIMAL TRANSPORT

Optimal transport

Kantorovich relaxation

$$\min_{\pi} \int \int \frac{1}{2} c(x, y) d\pi(x, y)$$

s. t. $\pi \in \Pi(\mu_0, \mu_1)$



seeking a joint probability distribution.

Earth Mover's Distances on Discrete Surfaces

Solomon, Rustamov, Guibas, Butscher (2014)



Convolutional Wasserstein Distances: Efficient Optimal Transportation on Geometric Domains.

Solomon, de Goes, Peyré, Cuturi, Butscher, Nguyen, Du, Guibas (2015)



OPTIMAL TRANSPORT

Optimal transport

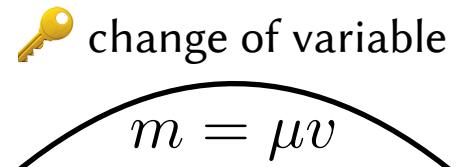
Benamou-Brenier

$$\min_{\mu_t, v_t} \int_0^1 \int_M \frac{1}{2} \|v_t(x)\|^2 d\mu_t(x) dt$$

$$\text{s. t. } \partial\mu_t + \nabla \cdot (\mu v) = 0$$

$$\mu(0) = \mu_0, \mu(1) = \mu_1$$

NON-CONVEX!



$$\min_{\mu_t, m_t} \int_0^1 \int_M \frac{1}{2} \frac{\|m_t(x)\|^2}{\mu_t} d\mu_t(x) dt$$

$$\text{s. t. } \partial\mu_t + \nabla \cdot m = 0$$

$$\mu(0) = \mu_0, \mu(1) = \mu_1$$

CONVEX

OPTIMAL TRANSPORT

Optimal transport

Dynamical Optimal Transport on Discrete Surfaces

Lavenant, Claici, Chien, Solomon

Lavenant, Claici, Chien, Solomon

Dynamical Optimal Transport on Discrete Surfaces
 M.A.N.T., Université Paris-Sud
 Massachusetts Institute of Technology
 Massachusetts Institute of Technology

Dynamical Optics
 HUGO LAVENANT, Université Paris-Sud
 SEBASTIEN CLAICI, Massachusetts Institute of Technology
 EDWARD CHIEN, Massachusetts Institute of Technology
 JUSTIN SOLOMON, Massachusetts Institute of Technology

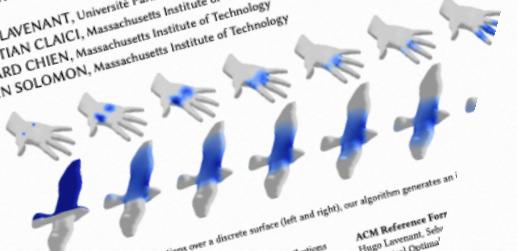


Fig. 1. Given two probability distributions over a discrete surface (left and right), our algorithm generates an ϵ -approximation to the theory of optimal transport. Unlike previous approaches, our method is based on a geometric interpretation of optimal transport for discrete surfaces.

Fig. 1. Given two probability distributions over a discrete surface into account. We propose a technique for interpolating between probability distributions on discrete surfaces, based on the theory of optimal transport. Unlike previous attempts that use linear programming, our method is based on a dynamical formulation of the optimal transport problem for flat domains by Benamou and Brenier [2000], adapted to discrete surfaces. Our structure-preserving construction yields a Riemannian metric on the (finite-dimensional) space of probability distributions on a discrete surface, which translates the so-called Célestini calculus to discrete language. From a practical perspective, our technique provides a smooth interpolation between a gradient distributions on discrete surfaces with less diffusion than state-of-the-art algorithms involving entropic regularization. Beyond interpolation, we show how our discrete notion of optimal transport extends to other tasks, such as distribution-valued Dantzig problems and time integration of gradient methodologies — Shape analysis, Mathematical image processing, Convex optimization, Partial differential equations.

CCS Concepts: Computing methodologies → Interpolation; Optimal Transport, Wasserstein Distar

... all or part of this
... article may be reproduced, stored in a
... retrieval system, or transmitted, in any form or by any means,
... electronic, mechanical, photocopying, recording or otherwise,
... without prior permission or written agreement from the
... publisher.

Institute on
Patent Law
University of Regensburg
93040 Regensburg, Germany
Postfach 2555
Phone: +49 941 943-1000
Fax: +49 941 943-1090
E-mail: patent@ur.de
Internet: www.patent.ur.de

https://www.industrydocuments.ucsf.edu/docs/230953 Page 1 of 1-16

Dynamical Optimal Transport on Discrete Surfaces • 259

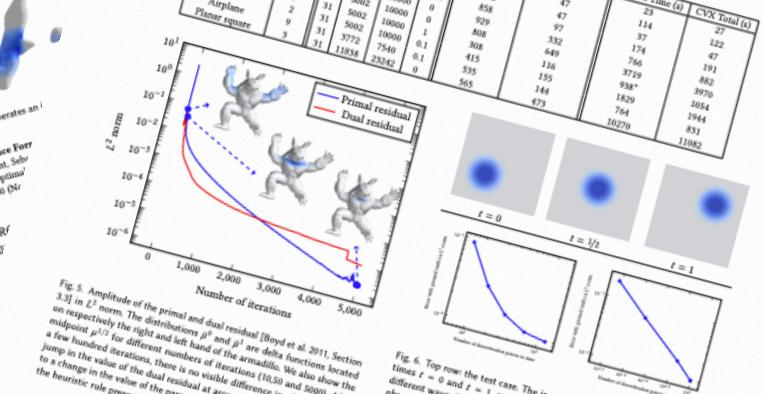


Fig. 5. Amplitude of the primal and dual residual [Boyd et al. 2011, Section 3.3] in L^2 norm. The distributions ρ^1 and ρ^2 are dual functions located respectively on the right and left half of the armature. We also show the midpoint rule for different numbers of iterations (10,30 and 500). After a few hundred iterations, there is no visible difference in L^2 . There is a jump in the value of the dual residual at around 4600 iterations. It is due to a change in the value of the parameter r , which is updated according to the heuristic rule presented in Section 3.4 of [Boyd et al. 2011].

As indicated in Section 3.4.1 of [Boyd et al. 2011],

As indicated in Section 3, it is not known whether the mesh refinement converges when the mesh size Δx goes to zero.

In Figure 6, however, we present some experiments indicating that the algorithm converges to the true Wasserstein distance when the time discretization is concerned, one could likely adapt the method of article.

is likely to be true.

norm (in g)

ACM Trans.

Digitized by srujanika@gmail.com

Digitized by srujanika@gmail.com

For more information about the study, please contact Dr. Michael J. Hwang at (319) 356-4550 or via email at mhwang@uiowa.edu.

Digitized by srujanika@gmail.com

Graph. Vis. 37, No. 6, Article 250. Publication date: November 1998.

OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX

Non-standard

Linear Programming
(LP)

Quadratic
Programming
(QP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

Semi-Definite
Programming
(SDP)

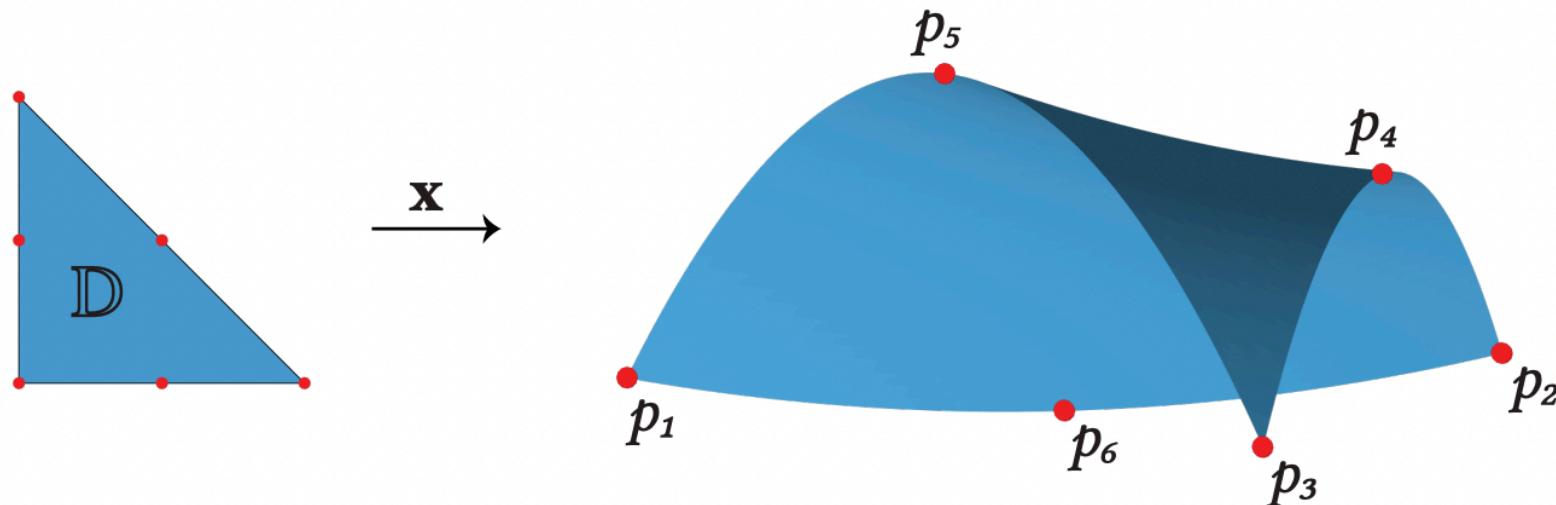
CONSTRAINED

UNCONSTRAINED

SOS RELAXATION



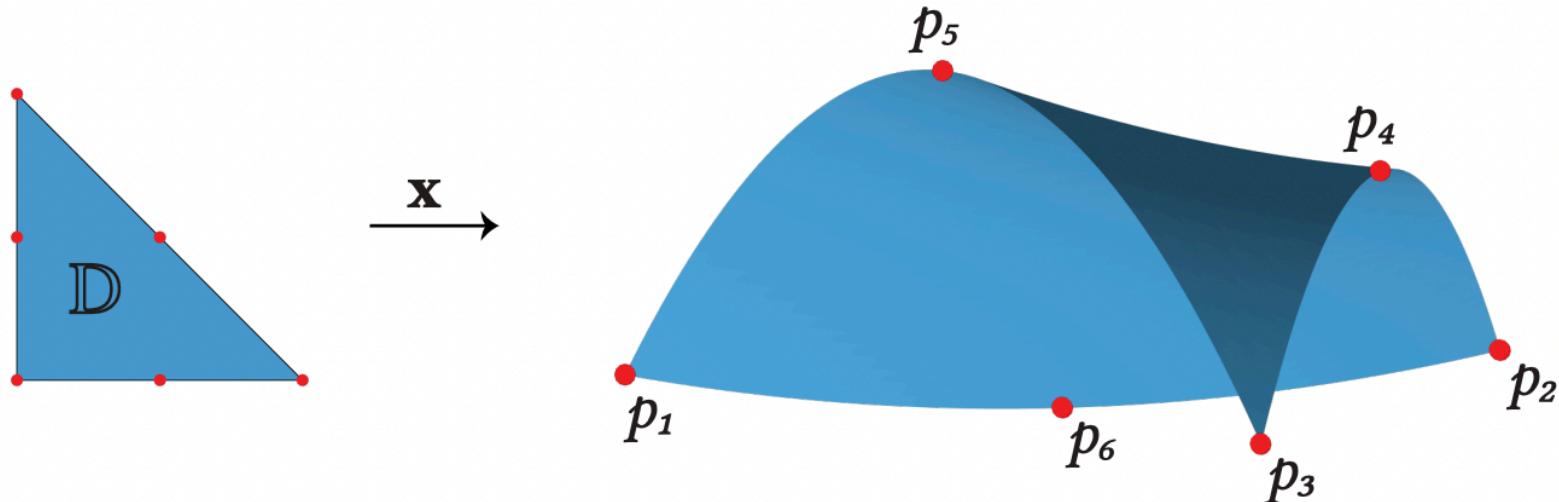
This will connect back to SDP relaxation



SOS RELAXATION



This will connect back to SDP relaxation



Putinar's Positivity Theorem

If f is polynomial, then f being nonnegative on \mathbb{D} is equivalent to f being expressible as a sum of squares of polynomials.

SOS RELAXATION

Typical Optimization Problem

$$\min_{x \in \mathbb{D}} f(x)$$

equivalent to

Reformulation with Linear Objective

$$\max \lambda$$

$$\text{s.t. } f(x) - \lambda \geq 0$$

$$x \in \mathbb{D}$$

Putinar's Positivity Theorem

If f is polynomial, then f being nonnegative on \mathbb{D} is equivalent to f being expressible as a sum of squares of polynomials.

SOS RELAXATION

Typical Optimization Problem

$$\min_{x \in \mathbb{D}} f(x)$$

Reformulation with Linear Objective

$$\begin{aligned} & \max \lambda \\ \text{s.t. } & f(x) - \lambda \geq 0 \\ & x \in \mathbb{D} \end{aligned}$$

Putinar's Positivity Theorem

If f is polynomial, then f being nonnegative on \mathbb{D} is equivalent to f being expressible as a sum of squares of polynomials.

Can solve via SOS relaxation if:

domain compact with polynomial boundary

objective is polynomial function

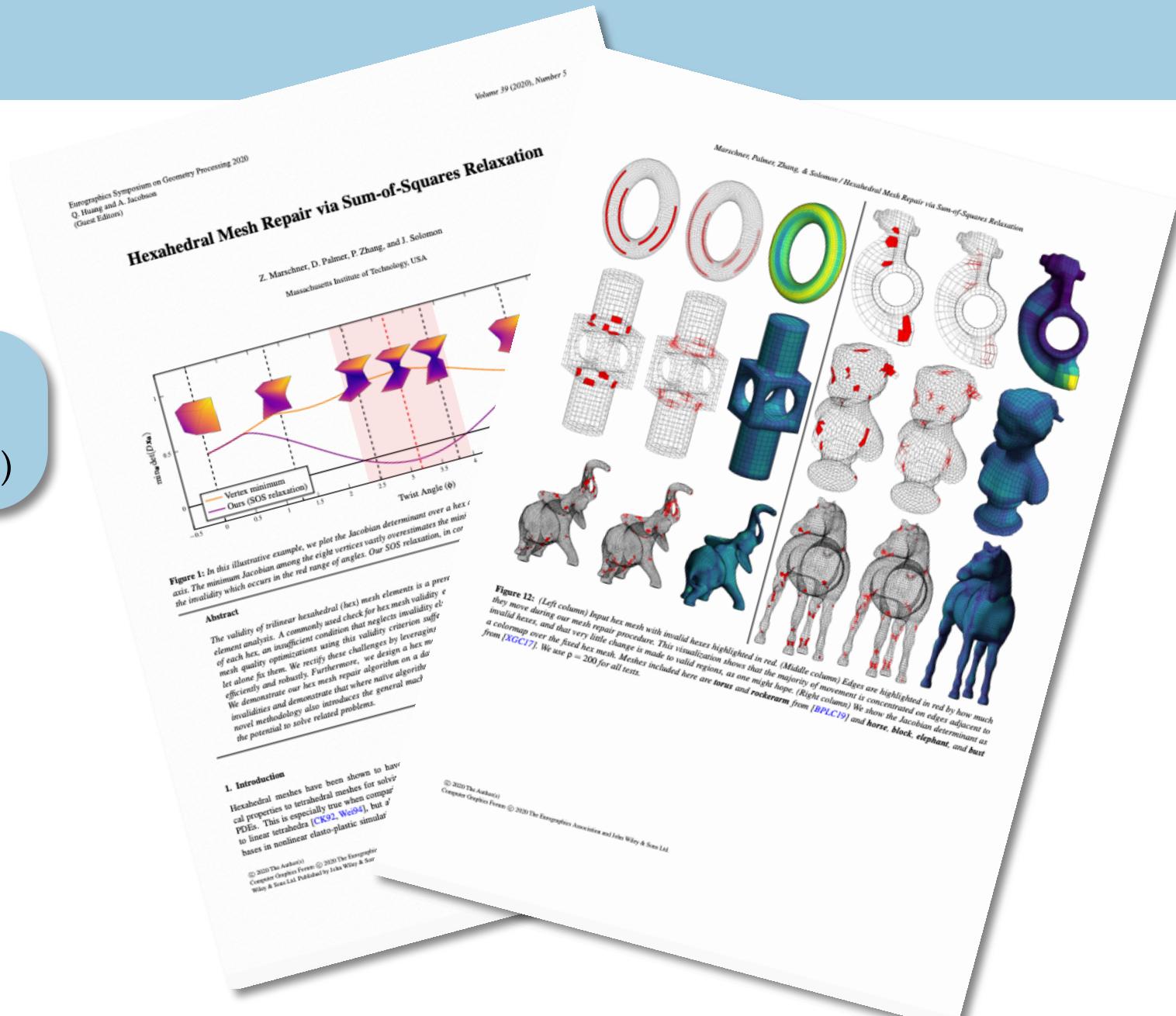
Solve resulting problem via SDP relaxation!

SOS RELAXATION

Mesh repair

Hexahedral Mesh Repair via Sum-of-Squares Relaxation

Marschner, Palmer, Zhang, Solomon (2020)



OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX

Non-standard

Linear Programming
(LP)

Quadratic
Programming
(QP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

Semi-Definite
Programming
(SDP)

CONSTRAINED

UNCONSTRAINED

CONVEX SUBSTRUCTURES



A number of optimization problems cannot be recast as fully convex. A *hidden convexity* can at least isolate the non-convex part. While this strategy is not a proper convex relaxation, it can lead to optimization algorithms that enjoy some of the benefits of convex optimization.

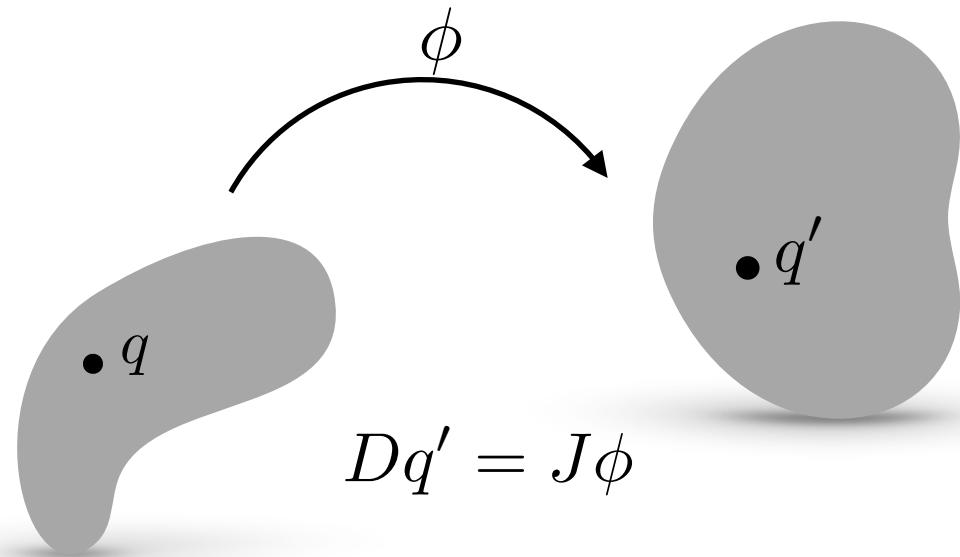
CONVEX SUBSTRUCTURES

🚧 Identifying convex substructure in distortion energies

Distortion energy

$$E(q) := \sum_i w_i f((Dq)_i)$$

NON-CONVEX!

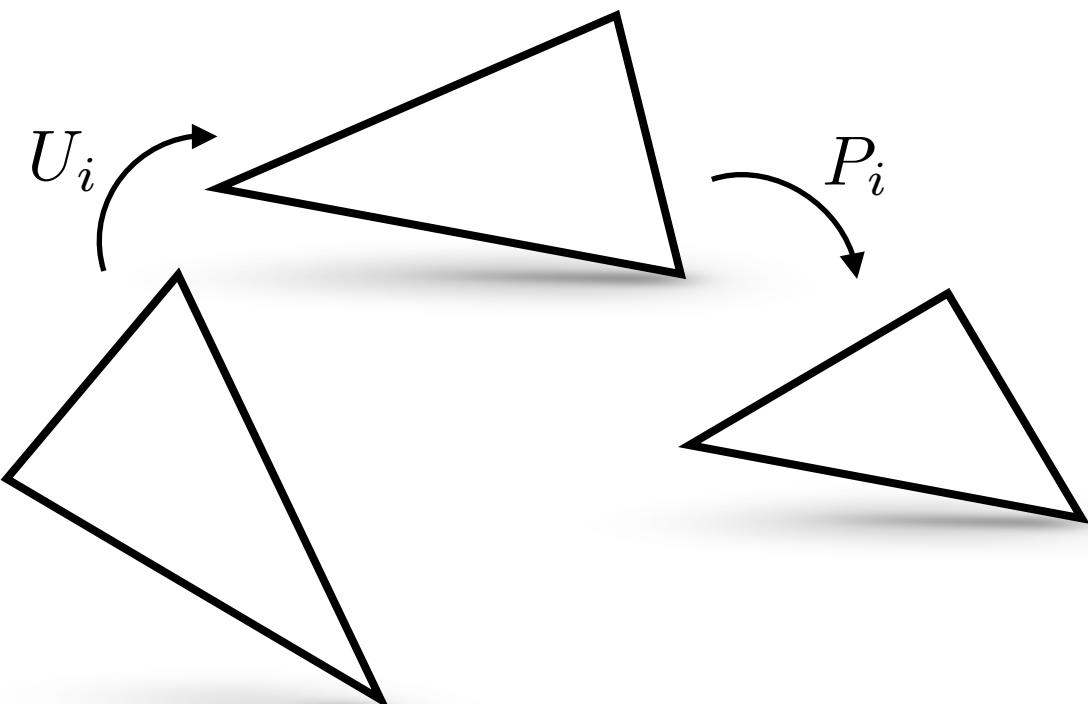


Deformation gradient

CONVEX SUBSTRUCTURES

Apply polar decomposition to each Jacobian matrix $J_i = U_i P_i$

$$\begin{aligned} \min_{U \in SO(d), P \in \mathcal{S}_+^d} E(Dq) &= \sum_i w_i f(P_i) \\ \text{s. t. } (Dq)_i - U_i P_i &= 0 \end{aligned}$$



Stein, Li, Solomon (2021)

A Splitting Scheme for Flip-Free Distortion Energies

Oded Stein*, Jiajin Li†, and Justin Solomon‡

Abstract. We introduce a robust optimization method for flip-free distortion energies used, for example, in parametrization, deformation, and volume correspondence. This method can minimize a variety of distortion energies, such as the symmetric Dirichlet energy and our new symmetric gradient energy. We identify energies, such as the symmetric Dirichlet energy and our new symmetric gradient energy, that are non-convex, non-smooth, and have a complex structure of distortion energies. The scheme results in an efficient method to deal with a non-convex, non-smooth nature of distortion energies. The scheme results in an efficient method when the global step involves a single matrix multiplication and the local steps are closed-form per-triangle/per-tetrahedron expressions that are highly parallelizable. The resulting general-purpose optimization algorithm exhibits robustness to flipped triangles and tetrahedra in initial data as well as during the optimization. We establish the convergence of the proposed algorithm under certain conditions and demonstrate applications to parametrization, deformation, and volume correspondence.

Key words. computer graphics, optimization, nonconvex optimization, parametrization, ADMM
AMS subject classifications. 65K10, 90C26, 65D18, 68U05

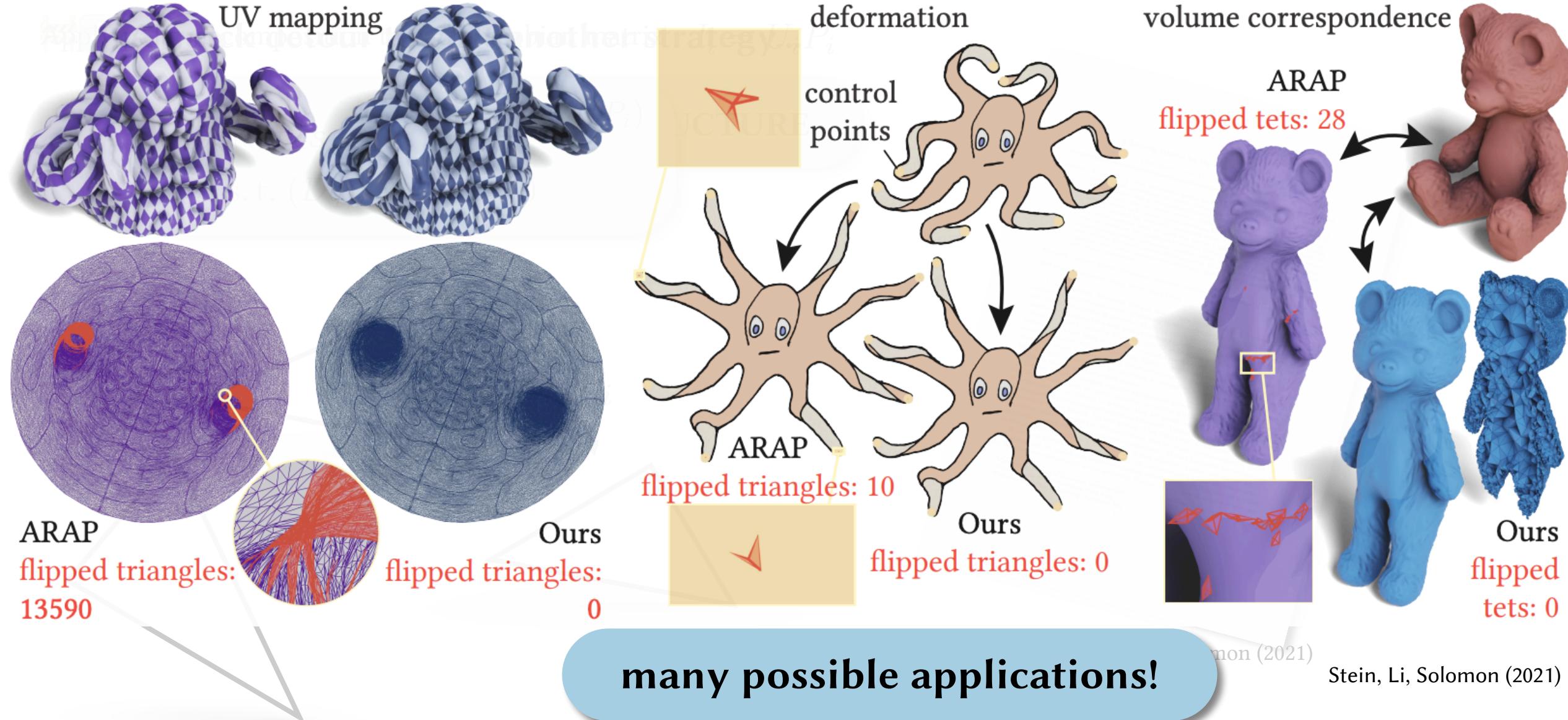
1. Introduction. Distortion energies measure how much a mapping from one shape to another deforms the initial shape. Minimizing these energies can yield maps between domains with just little distortion as possible. Minimization of distortion energies such as UV mapping constraints is employed in a wide array of computer graphics applications, such as UV mapping (where one seeks to embed a 3D surface into 2D while minimizing distortion, Figure 1 left), deformation (where parts of a surface or volume are deformed, and the goal is to find the overall deformation with least distortion, Figure 1 center), and volume correspondence (two boundary surfaces are given, and a distortion-minimizing map between the two volumes is desired, Figure 1 right). We are interested in computing maps that minimize distortion energies on triangle and tetrahedral meshes.

Flip-free distortion energies comprise an important subset of distortion energies. A mapping that minimizes such an energy will never invert (flip) a triangle or tetrahedron. Flip-free distortion energies are difficult to optimize: they are usually non-linear and non-convex, and singularities that correspond to collapsed elements. Typical optimization methods based on line-search require feasible, flip-free iterates. Thus, they must exercise great care to avoid singularities, where they will fail. For applications such as volume correspondence set, it can struggle with the problem's non-convexity in both objective function and feasible set.

We focus on distortion energies that depend only on the mapping's Jacobian, are invariant to rotations, and are convex over symmetric positive definite matrices. This includes popular flip-free distortion energies such as the symmetric Dirichlet energy, as well as our new symmetric

*Massachusetts Institute of Technology, Cambridge, MA
 †The Chinese University of Hong Kong, Hong Kong
 ‡Massachusetts Institute of Technology, Cambridge, MA

CONVEX SUBSTRUCTURES



CONVEX SUBSTRUCTURES

Question: What happens if we add dynamics?

Lagrangian Mechanics

$$\min \mathcal{L}, \quad \mathcal{L} = K - E$$

convex concave



the solution of this optimization problem gives
the equations of motion

Not obvious that the hidden
convexity will be helpful!

CONVEX SUBSTRUCTURES

Question: What happens if we add dynamics?

Lagrangian Mechanics

$$\min \mathcal{L}, \quad \mathcal{L} = K - E$$

convex concave



the solution of this optimization problem gives
the equations of motion

Not obvious that the hidden
convexity will be helpful!

Mostly Convex Reformulation

$$\min K' + E'$$

convex hidden convexity



Preserve first-order optimality constraints!

simplified preview
(not the full story)



CONVEX SUBSTRUCTURES

Question: What happens if we add dynamics?

Lagrangian Mechanics

Variational Elastodynamic Simulation

Mattos Da Silva, Sellán, Pacheco-Tallaj, Solomon (2025)

Solution gives equations of motion

Not obvious that the hidden convexity will be helpful!

Variational Elastodynamic Simulation

LETICIA MATTOS DA SILVA, Massachusetts Institute of Technology
SILVIA SELLÁN, Massachusetts Institute of Technology
NATALIA PACHECO-TALLAJ, Massachusetts Institute of Technology
JUSTIN SOLOMON, Massachusetts Institute of Technology

Fig. 1. It's Flying Rubber! Our method allows for the simulation of elastic objects, such as this robbery bouncy blob (click here to see the accompanying video).

Numerical schemes for time integration are the cornerstone of dynamical simulations for deformable solids. The most popular time integrators for isotropic distortion energies rely on nonlinear root-finding solvers, most prominently, Newton's method. These solvers are computationally expensive and sensitive to ill-conditioned Hessians and poor initial guesses; these difficulties can particularly hamper the effectiveness of variational integrators, whose momentum conservation properties require reliable root-finding integration. To tackle these difficulties, this paper shows how to express variational time integration for a large class of elastic energies as an optimization problem with a "hidden" convex substructure. This hidden convexity suggests using optimization techniques with rigorous convergence analysis, guaranteed inversion free elements, and conservation of physical invariants up to tolerance/numerical precision. In particular, we propose an Alternating Direction Method of Multipliers (ADMM) algorithm combined with a proximal operator step to solve our formulation. Especially, our integrator improves the performance of elastic simulation tasks, as we demonstrate in a number of examples.

CCS Concepts • Computing methodologies → Physical simulation

Additional Key Words and Phrases: Time integration methods, convex optimization, ADMM

1 INTRODUCTION

An elegant property of classical mechanics is that it admits a wide variety of formulations and interpretations. Most prominently, Newton's laws determine the state of a system by evolving a second-order system of differential equations. An equivalent variational formulation provided by Lagrangian mechanics shows that the equations of motion can be understood as critical points of an action functional, which maps entire trajectories to scalar values. This formulation helps derive conserved quantities of physical interest like the Hamiltonian and momenta.

In the discrete setting, myriad time integration methods have been proposed to solve the equations of motion, with the goal of evolving the system's state over discrete time steps while maintaining stability and numerical accuracy. For instance, implicit Euler integration is used to take large time steps stably, with first-order accuracy. Explicit Runge-Kutta methods offer higher accuracy under moderate time-step restrictions for stability. These well-known integrators, however, do not conserve physical invariants: implicit integrators like Euler's tend to introduce numerical damping, losing energy over time even in the absence of dissipative forces, while explicit integrators like Runge-Kutta can increase the energy of the system over time.

Inspired by these shortcomings, variational time integrators adapt the Lagrangian formulation, exploiting discrete Hamilton principles, i.e., discrete principles of stationary actions to conserve momenta and approximately conserve energy. Implicit variational integrators typically rely on iterative nonlinear solvers such as Newton's method to advance the state of the system. Thus, despite their conservation properties, these methods can be computationally expensive, be sensitive to the choice of initial conditions for the root-finding procedure, and lack convergence analysis.

In this paper, we propose an alternative to classical nonlinear solvers used in variational integrators for elastodynamic simulation. In particular, rather than expressing time integration using a generic root-finding or nonsmooth optimization problem, we derive an equivalent optimization-based formulation that is jointly convex in all of its variables except the rotational component of the unknown deformation gradient. Leveraging this reformulation, we provide an Alternating Direction Method of Multipliers (ADMM) algorithm combined with a simple proximal operator step to solve

4.1 Hidden Convexity in the Action Functional

Following the notation of §3.2, let $\hat{q}^k \in \mathbb{R}^{n \times C}$ be our estimate of $q(t)$. In time kk , again, let $\hat{q}^{k+1} = (q^{k+1}, -\dot{q}^k)/h$ be the velocity, and define $\hat{q} = (q^{k+1}, \tau q^k)^T/2$ as the midpoint approximation. We estimate q^{k+1} applying Hamilton's principle to the action functional combined with the polar decomposition of the Jacobian. In particular, we define a constrained action functional using a discretization as follows:

$$\mathcal{L}^d = \sum_k h \left[K(\hat{v}^{k+1}) - E(\hat{p}^{k+1}) \right],$$

subject to $D_i \hat{q}_{mid}^{k+1} - U_i^{k+1} \hat{p}_i^{k+1} = 0 \quad \forall i.$ (6)

The formulation parametrizes the mesh by discrete position space and velocity on a staggered grid. The constrained action functional in (6) preserve the discrete grid and the system and thus preserve the resulting integrator. Moreover, by Noether's theorem, the discrete Lagrangian yields exactly second-order accuracy [Fetecau et al., 2014].

By applying Hamilton's principle to the action functional (6), we find that the resulting integrator is symplectic. If the discrete grid is jointed and symmetric, then the action functional is given by

$$I = \sum_k h \left(\frac{1}{2} \sum_i \hat{p}_i^k \cdot \nabla f(\hat{p}_i^k) \right). \quad (7)$$

OUR ROADMAP

OPTIMIZATION

1. The Fundamentals
2. Convexity
3. Why Convexity?
4. Standard Convex Problems

CONVEX RELAXATION

5. Viscosity Solutions
6. Mapping Problems
7. Optimal Transport
8. SOS Relaxations
9. Convex Substructures

SIMPLIFIED TAXONOMY OF CONVEX

Non-standard

Linear Programming
(LP)

Quadratically Constrained
Quadratic Program
(QCQP)

Second-Order
Cone Programming
(SOCP)

Semi-Definite
Programming
(SDP)

Quadratic
Programming
(QP)

CONSTRAINED

UNCONSTRAINED

SUGGESTED READING

OPTIMIZATION

Snakes: Active Contour Models
Kass, Witkin and Terzopoulos (1988)

STANDARD CONVEX PROBLEMS

Bounded Biharmonic Weights for Real-Time Deformation

Jacobson, Baran, Popović, Sorkine (2011)

Controlling Singular Values with Semidefinite Programming

Kovalsky, Aigerman, Basri, Lipman (2014)

RELATED READING

Convex Optimization

Boyd & Vandenberghe (2004)

SUGGESTED READING

VISCOSITY SOLUTIONS

An ADMM-based Scheme for Distance Function Approximation

Belyaev & Fayolle (2020)

A Convex Optimization Framework for Regularized Geodesic Distances

Edelstein, Guillen, Solomon, Ben-Chen (2023)

A Framework for Solving Parabolic Partial Differential Equations on Discrete Domains

Mattos Da Silva, Stein, Solomon (2024)

RELATED READING

Geodesics in Heat : A New Approach to Computing Distance Based on Heat Flow

Crane, Weischedel, Wardetzky (2013)

The Discrete Geodesic Problem

Mitchell, Mount, Papadimitriou (1987)

Fast Exact and Approximate Geodesics on Meshes

Surazhsky, Surazhsky, Kirsanov, Gortler, Hoppe (2005)

SUGGESTED READING

MAPPING PROBLEMS

Point Registration via Efficient Convex Relaxation

Maron, Dym, Kezurer, Kovalsky, Lipman (2016)

Soft Maps Between Surfaces

Solomon, Nguyen, Butscher, Ben-Chen, Guibas (2012)

RELATED READING

Blended Intrinsic Maps

Kim, Lipman, Funkhouser (2011)

OPTIMAL TRANSPORT

Earth Mover's Distances on Discrete Surfaces

Solomon, Rustamov, Guibas, Butscher (2014)

Convolutional Wasserstein Distances

Solomon, de Goes, Peyré, Cuturi, Butscher, Nguyen, Du, Guibas (2015)

Dynamical Optimal Transport on Discrete Surfaces

Lavenant, Claici, Chien, Solomon

SUGGESTED READING

SOS RELAXATIONS

Hexahedral Mesh Repair via Sum-of-Squares Relaxation

Marschner, Palmer, Zhang, Solomon (2020)

RELATED READING

Sum-of-Squares Geometry Processing

Marshner, Zhang, Palmer, Solomon (2021)

Sum-of-Squares Collision Detection for Curved Shapes and Paths

Marschner*, Zhang*, Solomon, Tamstorf (2023)

CONVEX SUBSTRUCTURES

A Splitting Scheme for Flip-Free Distortion Energies

Stein, Li, Solomon (2022)

Variational Elastodynamic Simulation

Mattos Da Silva, Sellán, Pacheco-Tallaj, Solomon (2025)

THANK YOU FOR ATTENDING!

Questions?



Presenter's website