$L_\infty$ Optimization and Quasi-convex Optimization

Fredrik Kahl

Chalmers, Sweden
$L_\infty$ Optimization and Quasi-convex Optimization

PART 1: Outline

- Introduction: why $L_\infty$ optimization?
- Convexity and quasi-convexity
- Examples
- Globally optimal algorithms
- Outliers and quasi-convexity
Multiple View Geometry and Geometric Reconstruction Problems

- Given images, reconstruct:
  - Scene geometry (structure)
  - Camera positions (motion)

Unknown camera positions
Benchmarking Long-Term Localization

High-quality night-time images

Seasonal changes, (sub)urban

Seasonal changes, urban; Low-quality night-time images

Sattler, Maddern, Toft, Torii, Hammarstrand, Stenborg, Safari, Okutomi, Pollefeys, Sivic, Kahl, Pajdla
Benchmarking 6DOF Outdoor Visual Localization in Changing Conditions, CVPR 2018
Multi-view optimization methods

- Algebraic cost-functions
  - Example: 8-point algorithm by Longuet-Higgins
  - Example: DLT for absolute pose

  + Simple and fast
  - Unstable. Cost function makes no sense.

- Minimal solvers
  - Example: 5-point algorithm by Nistér

  + Fast and good for robust estimation (RANSAC)
  - Only small dimensional problems
Multi-view optimization methods, cont’d

- Bundle adjustment
  + Good for refinement. ML estimate.
  - Requires good initial solution.

- $L_\infty$-optimization and convex optimization
  + Computes globally optimal solutions
  + Cost function based on reprojection errors
  + Good for detecting outliers
  - Bad with outliers
$L_\infty$ Optimization
and
Quasi-convex Optimization
The triangulation problem

- Given known camera positions and matched points
  \[ x_1 \leftrightarrow x_2 \leftrightarrow \ldots \leftrightarrow x_n \]
- Find the 3D point \( x \) that maps to these points.
Triangulation:

- Knowing $P$ and $P'$
- Knowing $x$ and $x'$
- Compute $\hat{X}$ such that

$$x = Px ; \quad x' = P'x$$
Triangulation in presence of noise

- In the presence of noise, back-projected lines do not intersect.

Rays do not intersect in space

Measured points do not lie on corresponding epipolar lines
Problem formulation

Cost function: Geometrically & statistically meaningful

\[ \min \left\| \begin{bmatrix} d(\tilde{x}, x) \\ \vdots \end{bmatrix} \right\|_{L_p} \]

measured image point
reprojected image point
Perspective cameras

\[ d(\tilde{x}, x)^2 \text{ can be written as rational function} \]

reprojected image point

measured image point
Do local minima occur?

Consider the following three-view triangulation problem \textit{in the plane}.

Contour plot of the $L_2$-error function

Three local minima
Two-view triangulation

Cost function

\[ C(X) = d(x, \hat{x})^2 + d(x', \hat{x}')^2 \]
Multiple local minima

- Cost function may have local minima.
- Shows that gradient-descent minimization may fail.

**Left:** Example of a cost function with three minima.

**Right:** Cost function for a perfect point match with two minima.
Projective Projection

Camera matrix – $3 \times 4$ matrix

$$P = \begin{pmatrix} p^1^T \\ p^2^T \\ p^3^T \end{pmatrix}$$

3D point represented in homogeneous coordinates

$$x = (x, y, z, 1)^\top$$

Projected image point is given by

$$x = \begin{pmatrix} u \\ v \end{pmatrix} = Px = \begin{pmatrix} p^1^T x / p^3^T x \\ p^2^T x / p^3^T x \end{pmatrix}$$
Given point $x$ that should map to image point $x$, error is

$$d(x, PX)^2 = \left( u - \frac{p^1^T x}{p^3^T x} \right)^2 + \left( v - \frac{p^2^T x}{p^3^T x} \right)^2$$

$$= \left( \frac{u(p^3^T x) - p^1^T x}{p^3^T x} \right)^2 + \left( \frac{v(p^3^T x) - p^2^T x}{p^3^T x} \right)^2$$

$$= \frac{f(x)^2 + g(x)^2}{\lambda(x)^2}$$

- All of $f$, $g$ and $\lambda$ are linear in $(x,y,z)$

Error to be minimized is

$$\sum_{i=1}^{N} d(x, P_i x)^2 = \sum_{i=1}^{N} \frac{f_i(x)^2 + g_i(x)^2}{\lambda_i(x)^2}$$
The main difficulty

- Function
  \[
  \frac{f_i(x)^2 + g_i(x)^2}{\lambda_i(x)^2}
  \]
  is not convex.
- Sum of non-convex functions can have several minima.
Triangulation cost – cross-section

\[ \frac{f(x)^2 + g(x)^2}{\lambda(x)^2} \]
Convex Optimization

— convex and quasi-convex functions, convex cones
Convex set

A set $D \subset \mathbb{R}^n$ is **convex** if the line joining points $x_0$ and $x_1$ lies inside $D$.

Intersection of convex sets is convex.
Convex function

$D$ – a domain in $\mathbb{R}^n$.

A convex function $f : D \rightarrow \mathbb{R}$ is one that satisfies, for any $x_0$ and $x_1$ in $D$:

$$f((1 - \lambda)x_0 + \lambda x_1) \leq (1 - \lambda)f(x_0) + \lambda f(x_1).$$

Line joining $(x_0, f(x_0))$ and $(x_1, f(x_1))$ lies above the function graph.
Convex optimization

The generic convex optimization problem is:

Minimize the convex function $f(x)$ over a convex set $D$.

Properties:

1. A local minimum is also a global minimum
2. The sum of convex functions is also convex
**Quasi-convex function**

**Definition:** Given a function $f : \mathbb{R}^n \rightarrow \mathbb{R}$, the $\alpha$-sublevel set of $f$ is the set

$$S_\alpha(f) = \{x \in \mathbb{R}^n | f(x) \leq \alpha\}$$

A function is quasi-convex if all its sublevel sets are convex.

- A convex function is quasi-convex, but not conversely.
Quasi-convex optimization problem

Minimize \( f(x) \)
subject to \( g_i(x) \leq 0 \)

where

- \( f \) is quasi-convex.
- \( g_i \) is convex for all \( i \).
Quasiconvex Functions

Sublevel Sets: \( S_\delta(f) = \{ \Theta \mid f(\Theta) \leq \delta \} \)
Quasiconvex Functions

Sublevel Sets: $S_\delta(f) = \{ \Theta | f(\Theta) \leq \delta \}$
Quasiconvex Functions

A function $f$ is **quasiconvex** if its sublevel sets $S_{\delta}(f)$ are convex for all $\delta$. 

---

![Graphs showing quasiconvex and non-quasiconvex functions](image_url)
Quasiconvex Functions

\( f \) is quasiconvex if its sublevel sets \( S_\delta(f) \) are convex \( \forall \delta \).

Quasiconvex

Not Quasiconvex
Sublevel set is convex – function is quasi-convex
Maximum of quasi-convex functions

If functions $f_i$, $i = 1, \ldots, r$ are quasi-convex, then the function

$$f(x) = \max_i f_i(x)$$

is quasi-convex.

**Proof:** $S_\alpha(f) = \bigcap_i S_\alpha(f_i)$.

Intersection of convex sets is convex.
Minimax Optimization

Find $\min_x \max_i f_i(x)$
Such that $x \in D$

Alternatively:

Find $\min_{x,s} s$
Subject to $f_i(x) \leq s$ $\forall i$
and $x \in D$
Second Order Cone Constraints

Consider the function $C(x)$:

$$C(x) = \frac{\| (f(x), g(x)) \|}{\lambda(x)}$$

with

$$\lambda(x) \geq 0$$

Second Order Cone Programs (SOCP)

Convex problems with constraint functions

\[ \|A_i x + b_i\| \leq c_i^T x + d_i \]

are called SOCP.

More general than LP, but less general than SDP.

SOCP is easily solvable by off-the-shelf software.
Bisection

Finding the minimum \( s \) so that all cone constraints are satisfied.
The minimax solution

1. The $L_{\infty}$ solution to the triangulation problem is to find

   \[ \min_{x} \max_{i} d(x_i, P_i x) \]

2. Function is quasi-convex on domain in front of cameras.

Do local minima occur?

Consider the following three-view triangulation problem in the plane.

Contour plot of the $L_\infty$-error function
Problem 2: partial structure and motion

- Assume calibrated cameras.
- Assume rotations are known.
- Given points $x_{ij}$ in several views, find the positions of points $x$ and cameras $C$ that minimize the projection errors.
Direction (unit) vectors from cameras (blue) to points (black) are given: Find the positions of the cameras and points.
Problem 3: Homographies

- Given known 3D points $u_i$ on a plane and corresponding image points $u'_i$.
- Find the homography that maps corresponding points, i.e., $u'_i \simeq Hu_i$. 
Problem 4: SfM using a Reference Plane

- Given a reference plane and corresponding image points.
- Find
  1. Interimage homographies
  2. Cameras and 3D points that map to corresponding image points.
Dinosaur Reconstruction
Example on a Real Sequence

Algorithm (reference plane):
1. Track image points
2. Compute interimage homographies
3. Compute cameras and 3D points
Flexible object tracking

M. Salzmann, R. Hartley, P. Fua, ICCV 2007
Other Problems

1. Camera resection.
2. Projections from $\mathcal{P}^n \rightarrow \mathcal{P}^m$
3. Minimax vanishing point estimation.
5. Projective triangulation.
6. Projective SfM given plane correspondence. (Hartley-Kahl)

Problem: How to incorporate rotations into this methodology?
Implementation

- Convex optimization based on SeDuMi
- Convex feasibility problem:
  - 0.05s for 3-view triangulation
  - 1s for 2270 cone constraints with 36 views and 328 points
- Improved Bisection method
- Typically 5-10 iterations required to reach optimum within $10^{-5}$ pixels

MATLAB Toolbox available on my homepage

See also: Pierre Moulon, PhD Thesis, 2013
Outliers

— detection/removal of outliers
Why are Outliers a Problem?

**Problem:** Find line of best fit

**Measurements:** $X_i = (x_i, y_i)$

**Parameters:** $\Theta = \{a, b\}$

**Error functions:** $f_i(\Theta) = (y_i - ax_i - b)^2$

$L_2$ optimization:
$$\min_{a,b} \sum_i (y_i - ax_i - b)^2$$

$L_\infty$ optimization:
$$\min_{a,b} \max_i (y_i - ax_i - b)^2$$

$X_9$ is an OUTLIER. We need to remove it!
Overview

When the $L_\infty$-idea was first introduced, it was considered a major drawback its sensitivity to outliers.

Now, one of its strengths.

Many different ideas and approaches for detection and removal introduced last few years.

- Outlier detection [Sim-Hartley].
- Abstract LP-approach [Li].
- Minimize infeasibility [Seo and Ke-Kanade].
- Verification strategy [Olsson-Enqvist-Kahl].
How to define an outlier?

Suppose only two error functions $f_1(\Theta)$ and $f_2(\Theta)$.

Choose a threshold $\delta_{in}$.

Then either $f_1(\Theta)$ or $f_2(\Theta)$ has to be removed such that

$$\min_{\Theta} \max_{i \in I_{in}} f_i(\Theta) < \delta_{in}$$

where $I_{in} = \{1\}$ or $I_{in} = \{2\}$. But which one?

It is inherently **AMBIGUOUS**.
**Definition of an Outlier**

We have error functions $f_i(\Theta)$ indexed by $i$ in an index set $I$.

Choose a threshold $\delta_{in}$.

Choose largest subset $I_{in}$ (the inlier set) that satisfies
\[
\min_{\Theta} \max_{i \in I_{in}} f_i(\Theta) < \delta_{in}
\]

An **inlier** is any measurement in $I_{in}$
An **outlier** is any measurement not in $I_{in}$.

Index set $I$ is made up of two subsets - $I_{in}$ (inlier set) and $I_{out}$ (outlier set).
\[
I = I_{in} \cup I_{out}
\]

$I_{in} = \{3, 4\}$ and $I_{out} = \{1, 2\}$
How Do We Remove Outliers?

- **Method 1:** RANSAC
  - Relies on random sampling to find a set of measurements containing only inliers.
  - Can only be used on problems where solution can be computed quickly and from only a small number of measurements.

- **Method 2:** Throw out measurements with largest residual
  - Solve optimization problem.
  - Remove measurements with largest residual.
  - Repeat first two steps until an acceptable max residual is achieved.

For this to work, the set of measurements with largest residual must contain outliers. **BUT THIS IS NOT ALWAYS THE CASE!**
Outlier Removal Strategy

Outlier removal strategy:
- Solve optimization problem
- Remove measurements with largest residual
Outlier Removal Strategy

Outlier removal strategy:
- Solve optimization problem
- Remove measurements with largest residual

Why does strategy fail for general $L_2$ or $L_\infty$ problems?
For general $L_2$ or $L_\infty$ problems, the set of measurements with largest residual does not necessarily contain outliers.

BUT strategy works for certain $L_\infty$ problems!
We show that, under certain conditions, the measurements with largest residual are guaranteed to contain outliers.
**What Conditions Are Needed?**

**Theorem:** *(Under certain conditions)*
Consider a minimax problem with solution $\min_{\Theta} \max_{i \in I} f_i(\Theta) = \delta_{opt}$. Suppose there exists $I_{in} \subset I$ for which $\min_{\Theta} \max_{i \in I_{in}} f_i(\Theta) < \delta_{in} < \delta_{opt}$. Then $I_{supp}$ must contain at least one index $i$ not in $I_{in}$.

**In English:** The support set must contain at least one outlier.

**Condition A:** *(Under certain conditions)*
If $f_0$ is a function not in the support set for a minimax problem, then we can remove $f_0$ without decreasing the $L_\infty$ error $\delta_{opt}$. That is, if $0 \notin I_{supp}$, then $\min_{\Theta} \max_{i \in I \setminus \{0\}} f_i(\Theta) = \min_{\Theta} \max_{i \in I} f_i(\Theta) = \delta_{opt}$.

**In English:** If $f_0 \notin I_{supp}$, then $f_0$ should not be constraining our solution. So we can remove $f_0$ without affecting the $L_\infty$ error $\delta_{opt}$. 
Quasiconvexity Is Needed For Condition A To Hold

- $A, B, C$ are the sublevel sets of 3 error functions $f_{i_A}, f_{i_B}, f_{i_C}$.
- $f_{i_C}$ is QC $\Rightarrow$ $C$ is a convex set
- $f_{i_A}, f_{i_B}$ are not QC $\Rightarrow$ $A, B$ are nonconvex sets
- $\Theta_{opt} = A \cap B \cap C$

- $\Theta_{opt} \notin bd(C) \Rightarrow f_{i_C} \notin I_{supp} = \{i_A, i_B\}$
- Suppose we remove $f_{i_C}$. 
Quasiconvexity is needed for condition A to hold.

- $A, B, C$ are the sublevel sets of 3 error functions $f_{i_A}, f_{i_B}, f_{i_C}$.
- $f_{i_C}$ is QC $\Rightarrow$ $C$ is a convex set.
- $f_{i_A}, f_{i_B}$ are not QC $\Rightarrow$ $A, B$ are nonconvex sets.
- $\Theta_{opt} = A \cap B \cap C$

- $\Theta_{opt} \notin bd(C) \Rightarrow f_{i_C} \notin I_{supp} = \{i_A, i_B\}$
- Suppose we remove $f_{i_C}$.

- Since $A, B$ are not convex, the solution may jump to $\Theta'$ where $f_{i_A}(\Theta') < \delta_{opt}$ and $f_{i_B}(\Theta') < \delta_{opt}$.
- That is, because $A, B$ are not convex, it is possible to remove $f_{i_C} \notin I_{supp}$ and obtain a lower $L_\infty$ error $\delta'$ at $\Theta'$. 

We need convex sublevel sets. Quasiconvexity is needed!
Quasiconvexity Is Needed For Condition A To Hold

We need convex sublevel sets. Quasiconvexity is needed!
...but Quasiconvexity Is Insufficient

If \( \{\Theta\}_{\text{soln}} \) is a single point, then QC is necessary and sufficient.
If \( \{\Theta\}_{\text{soln}} \) contains more than a single point, then QC is necessary but insufficient.

- \( f_1, f_2 \) are quasiconvex
- \( \min_{\Theta} \max_{i=1,2} f_i(\Theta) = \delta_{opt} \)
- \( \{\Theta\}_{\text{soln}} = \cap_{i=1,2} S_{\delta_{opt}}(f_i) = [a, b] \)
- But bisection algorithm only returns a single point \( \Theta_{opt} \in \{\Theta\}_{\text{soln}} \)

- \( f_1(\Theta_{opt}) < \delta_{opt} \Rightarrow f_1 \notin I_{\text{supp}} = \{2\} \)
- Suppose we remove \( f_1 \).

- Bisection algorithm will find a new solution \( \Theta' \) with a lower \( L_{\infty} \) error \( \delta' \).
  \( \Rightarrow \) Quasiconvexity is insufficient

\[ I_{\text{supp}} = \{i | f_i(\Theta_{opt}) = \delta_{opt}\} = \{2\} \]

Need smoothness condition on sublevel sets. Strict Quasiconvexity is needed!
Strict quasiconvexity: As $\delta$ decreases, the sublevel sets $S_{\delta}(f)$ must shrink smoothly. That is, no plateaus allowed.

**Definition:** $f$ is strictly QC if $\bigcup_{\mu<\delta} S_{\mu}(f) = \text{Int} \ S_{\delta}(f) \ \forall \ \delta$
Strict QC is sufficient

**Theorem:** Consider a minimax problem with solution \( \min_{\Theta} \max_{i \in I} f_i(\Theta) = \delta_{opt} \) where error functions \( f_i(\Theta) \) are all strictly quasiconvex. Suppose there exists \( I_{in} \subset I \) for which \( \min_{\Theta} \max_{i \in I_{in}} f_i(\Theta) < \delta_{in} < \delta_{opt} \). Then \( I_{supp} \) must contain at least one index \( i \) not in \( I_{in} \).

**In English:** If our error functions \( f_i(\Theta) \) are all strictly quasiconvex, then the support set must contain at least one outlier.

For a detailed proof, see:

What Does This All Mean?

If we can write a geometric vision problem as an $L_\infty$ optimization problem where the error functions $f_i(\Theta)$ are strictly quasiconvex then $I_{supp}$ must contain at least one outlier.

So by repeatedly throwing out part or all of $I_{supp}$, it should be possible to eventually remove outliers from a given problem.
Results - Reconstruction

- 4402 image points $x_{ij}$ used to recover 36 camera locations $C_i$ and 1381 scene points $X_j$.
- Gaussian noise added to 5% of the 4402 image points $x_{ij}$ (i.e. 220 outliers).

<table>
<thead>
<tr>
<th>Cycle</th>
<th>Max Residual</th>
<th>Size of $I_{supp}$</th>
<th>Remaining Outliers</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0390</td>
<td>10</td>
<td>210</td>
</tr>
<tr>
<td>2</td>
<td>0.0277</td>
<td>43</td>
<td>168</td>
</tr>
<tr>
<td>3</td>
<td>0.0196</td>
<td>54</td>
<td>123</td>
</tr>
<tr>
<td>4</td>
<td>0.0140</td>
<td>100</td>
<td>57</td>
</tr>
<tr>
<td>5</td>
<td>0.0080</td>
<td>72</td>
<td>23</td>
</tr>
<tr>
<td>6</td>
<td>0.0035</td>
<td>60</td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td>0.0019</td>
<td>36</td>
<td>4</td>
</tr>
</tbody>
</table>