Efficient Optimization for Robust Bundle Adjustment

Zhongnan Qu

Final Presentation of Master Thesis

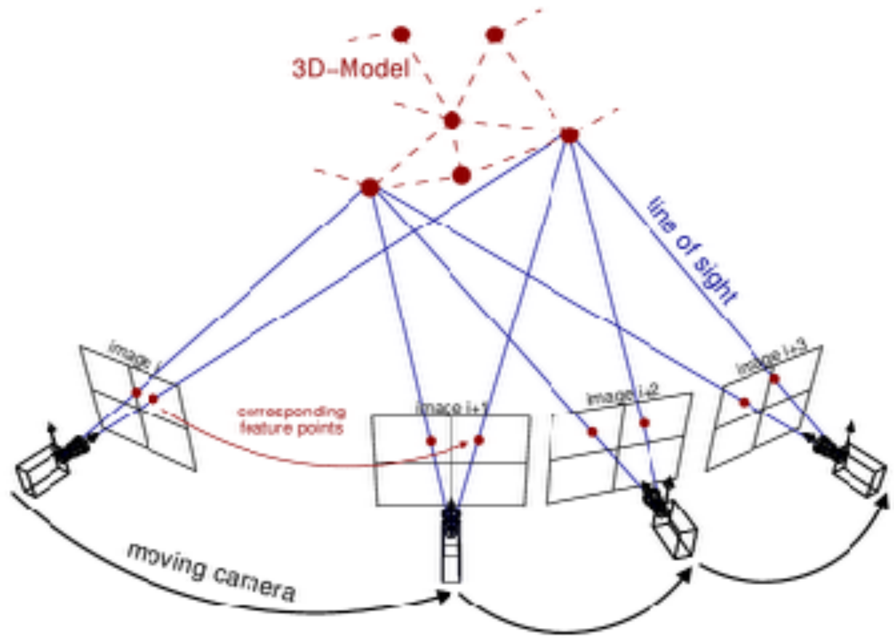
Prof. Dongheui Lee, Shile Li,
Prof. Daniel Cremers, Dr. Tao Wu, Nikolaus Demmel, Emanuell Laude

Human-centered Assistive Robotics
Technische Universität München

Motivation

Why are we addressing this problem?

- large-scale frame collections
- 3D Geometry Reconstruction
- SLAM



Structure of Motion

Estimating three-dimensional structures from two-dimensional image sequences

Problem Formulation

Problem

Bundle Adjustment:

Jointly optimize the parameters (feature positions and camera parameters) based on frame collections

Criterion: minimizing reprojection errors

Challenges in traditional implementations:

- **Efficiency** in large-scale dataset (curse of dimensionality)
- **Accurate** linearization of highly non-linear model
- Numerical **accuracy** in non-convex optimization
- **Robustness** facing bad conditioned or near singular matrix

Related Work

- Levenberg-Marquardt Algorithm solved by Cholesky decomposition [B. Triggs, Springer 1999]
- Truncated Newton‘s Method coupled with Schur complement [S. Agarwal, ECCV 2010, C. Wu, CVPR 2012]
- Distributed Bundle Adjustment, D-ADMM BA [K. Ramamurthy, ICCVW 2017]

Overview of My Master Thesis

Baselines in bundle adjustment

- LMA-CD: Levenberg-Marquardt step solved by Cholesky decomposition [B. Triggs, Springer 1999]
- BAL: Truncated Newton's method proposed in [S. Agarwal, ECCV 2010]

Damped inexact Newton's method (DINM), improving efficiency, accuracy, robustness:

- Implicit matrix form with cell-structure
- Conjugate gradient solver coupled with Schur complement
- Increment step bounded by scaled trust region
- Introducing the damping matrix into inexact Newton's method
- Mixed updating process for damping factor and trust region radius

Further improve the performance:

- Adaptive switched Schur complement mode for DINM (ADINM)
- Local parameterization (local DINM)
- Reweighted potential function (IRLS)

General Foundation

- Reprojection error: difference between reprojected point and its observation

$$F_{ij} = c'_{ij} - b_{ij} = \begin{pmatrix} c'_{ij,x} \\ c'_{ij,y} \end{pmatrix} - \begin{pmatrix} b_{ij,x} \\ b_{ij,y} \end{pmatrix}$$

- Least square optimization

$$\psi_f = \sum_{(i,j)\in\mathcal{S}} \Psi(\|F_{ij}\|) = \sum_{(i,j)\in\mathcal{S}} \frac{1}{2}\| F_{ij}(x_i, \phi_j, t_j, y_j)\|^2$$

$$\min_{u} \psi_f(u) = \frac{1}{2}\|F(u)\|^2$$

- Iterative step in numerical optimization

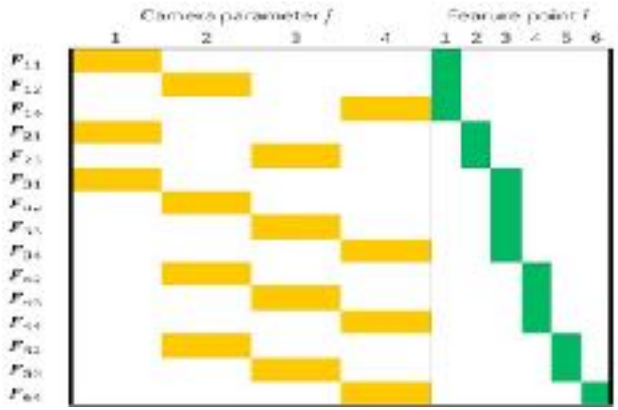
$$\psi_f(u + p) = \frac{1}{2}\|F(u) + J(u)p\|^2$$

$$\min_{p\in\mathbb{R}^s} \frac{1}{2}\|F^k - J^k p\|^2$$

DINM: Matrix Form

- Jacobian matrix:
  $\boldsymbol{J} = \nabla \boldsymbol{F} = \frac{d\boldsymbol{F}}{du}$
- Gradient vector:
  $\boldsymbol{g} = \boldsymbol{J}^T \boldsymbol{F}$
- Hessian matrix:
  $\boldsymbol{H} = \boldsymbol{J}^T \boldsymbol{J}$ $\quad \boldsymbol{H} = \begin{pmatrix} \boldsymbol{B} & \boldsymbol{E} \\ \boldsymbol{E}^T & \boldsymbol{C} \end{pmatrix}$

$\boldsymbol{J} =$



$\boldsymbol{H} =$

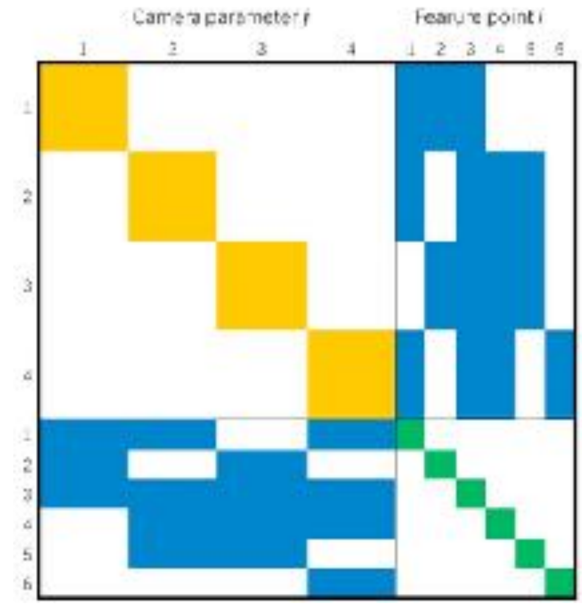


Table 4.1: Processing time and memory comparison

Implementation	EI	EI-sparse	II-sparse	II-cell
Time/s	201	194	1.32	0.37
Memory/MB	7.31	3.02	2.84	2.82

- More efficient in memory and computation time

DINM: Conjugate Gradient

- **Model function** in each iteration with trust region:

$$\min_{p \in \mathbb{R}^c} m_f^k(p) = \psi_f^k + g^{kT} p + \frac{1}{2} p^T H^k p \quad \text{s.t. } \|p\| \leq \Delta^k$$

$$H^k p = -g^k \text{ , if } \|p\| \leq \Delta^k \text{ and } H^k \text{ positive definite}$$

- General exact solver: Cholesky decomposition

- A exact solution of $\boldsymbol{p}$ in each iteration is inefficiency and unnecessary. System matrix $\boldsymbol{H}^k$ in Bundle adjustment is bad conditioned and near singular, so that decomposition always costs too much time and get an extremely inaccurate solution.
- Our inexact solver: **conjugate gradient**. Solution $\boldsymbol{p}$ is approached iteratively by the conjugate directions of $\boldsymbol{H}^k$.

- Computing conjugate directions only consumes little memory and time, also more robust to near singular problem.

DINM: Schur Complement

- Full equation system:

$$\begin{pmatrix} B & E \\ E^T & C \end{pmatrix} \begin{pmatrix} p_c \\ p_p \end{pmatrix} = - \begin{pmatrix} g_c \\ g_p \end{pmatrix}$$

$$g = \begin{pmatrix} g_c \\ g_p \end{pmatrix} \quad p = \begin{pmatrix} p_c \\ p_p \end{pmatrix}$$

$$H = \begin{pmatrix} B & E \\ E^T & C \end{pmatrix}$$

- Schur complement of C:

$$S_C = B - EC^{-1}E^T$$

$$v = -(-g_c + EC^{-1}g_p)$$

$$S_C p_c = -v$$

$$p_p = C^{-1}(-g_p - E^T p_c)\ .$$

$$C^{-1} = \begin{pmatrix} C_1^{-1} & & & \\ & C_2^{-1} & & \\ & & \ddots & \\ & & & C_m^{-1} \end{pmatrix}$$

Near linear complexity

Processing time comparison between different Schur complements

Implementation	H	S_C	S_B
Time per Newton-PCG/s	11.48	4.37	10.91
Time in 50 iteration/s	692	388	659

- Schur complement of ***C*** is more efficient generally.

DINM: Scaled Trust Region

- Bundle adjustment is poorly scaled, using Ellipsoid trust region:

$$\min_{\tilde{\boldsymbol{p}} \in \mathbb{R}^s} \widetilde{m}_f^k(\tilde{\boldsymbol{p}}) = \psi_f^k + (\tilde{\boldsymbol{g}}^k)^T \tilde{\boldsymbol{p}} + \frac{1}{2} \tilde{\boldsymbol{p}}^T \widetilde{\boldsymbol{H}}^k \tilde{\boldsymbol{p}} \quad \text{s.t. } \|\tilde{\boldsymbol{p}}\| \leq \Delta^k$$

$$\tilde{\boldsymbol{g}}^k = \mathrm{D}^{-1} \boldsymbol{g}^k \qquad \tilde{\boldsymbol{p}} = \mathrm{D} \boldsymbol{p} \qquad \widetilde{\boldsymbol{H}}^k = \mathrm{D}^{-1} \boldsymbol{H}^k \mathrm{D}^{-1}$$

- Design a new scaling matrix, **D** is generated from the optimal step in numerical derivatives.

- Scaling matrix **D** balance the sensitivities on different directions in argument space → more accurate solution.

DINM: Damping

- Damping matrix:

$$\boldsymbol{H}_\mu = \boldsymbol{J}^T\boldsymbol{J} + \mu\boldsymbol{D}$$

$$\boldsymbol{D} = diag(\boldsymbol{H}) \qquad \boxed{\boldsymbol{D} = \mathbf{I}}$$

- Damping factor:

μ is a nonnegative damping factor which controls the switching between Gauss Newton Algorithm and Gradient Descent Algorithm.

- Mixed updating process:

Damping factor and trust region radius are jointly updated in each iteration, according to how well the reduction of their model functions approximate the reduction of the real objective function.

$$\rho^k = \frac{\psi_f(\boldsymbol{u}^k) - \psi_f(\boldsymbol{u}^k + \boldsymbol{p}^k)}{m_f^k(\mathbf{0}) - m_f^k(\boldsymbol{p}^k)}$$

- Damping weakens the matrix problem of near singularity and bad conditioned.

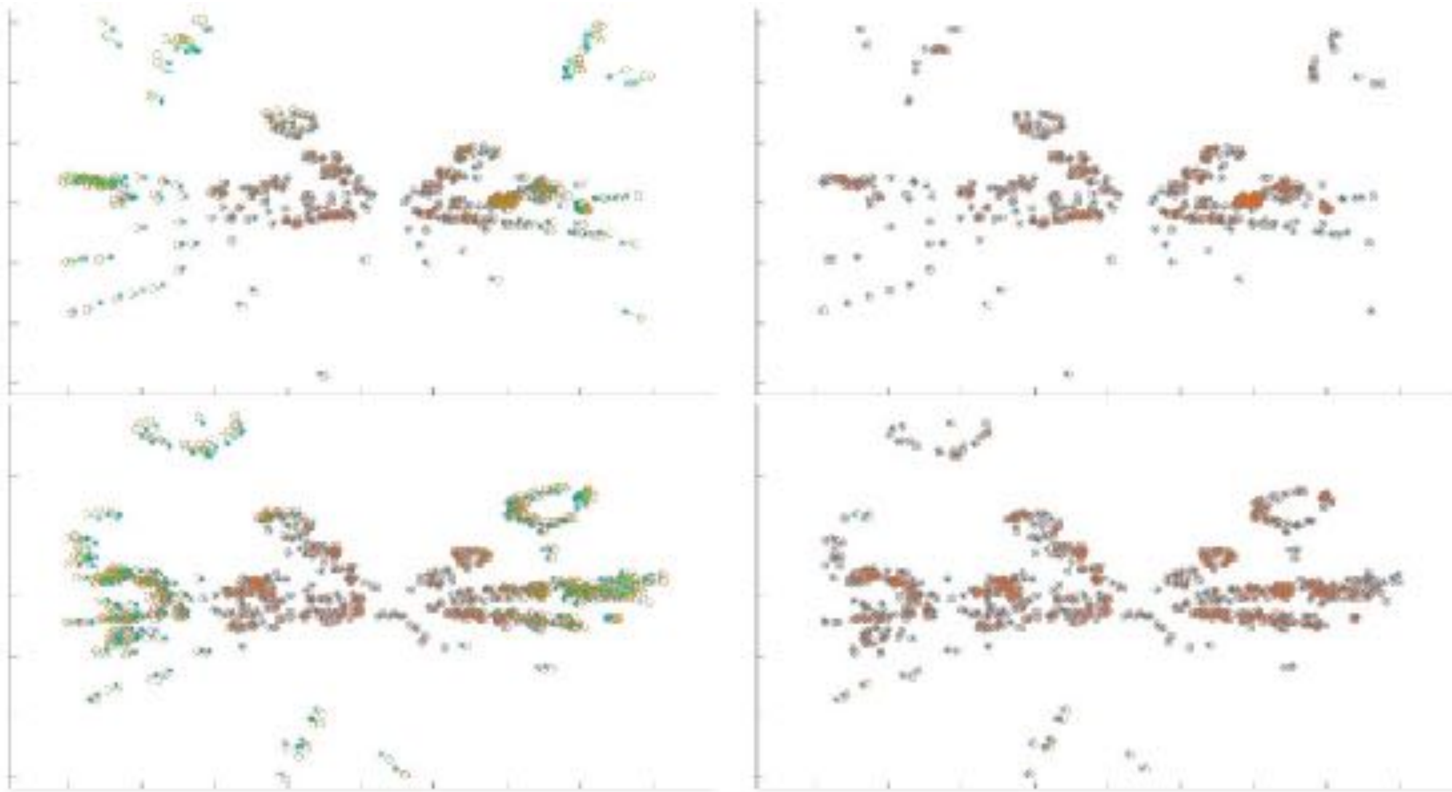
Evaluation

- Reprojection Error Plots:

∘: observations
* : reprojected positions
- : reprojection error

14

37



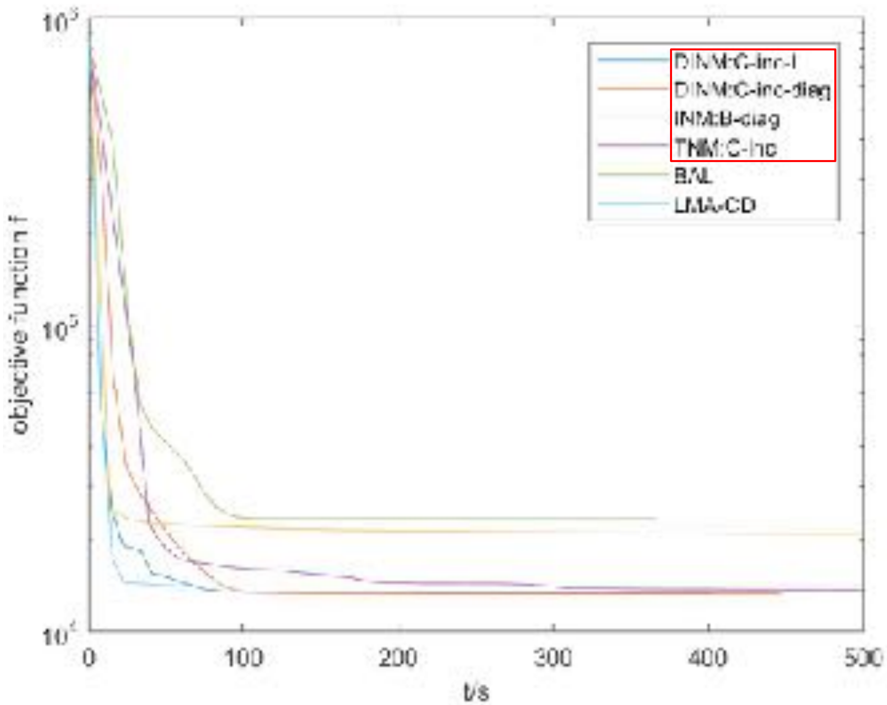
Before Optimization

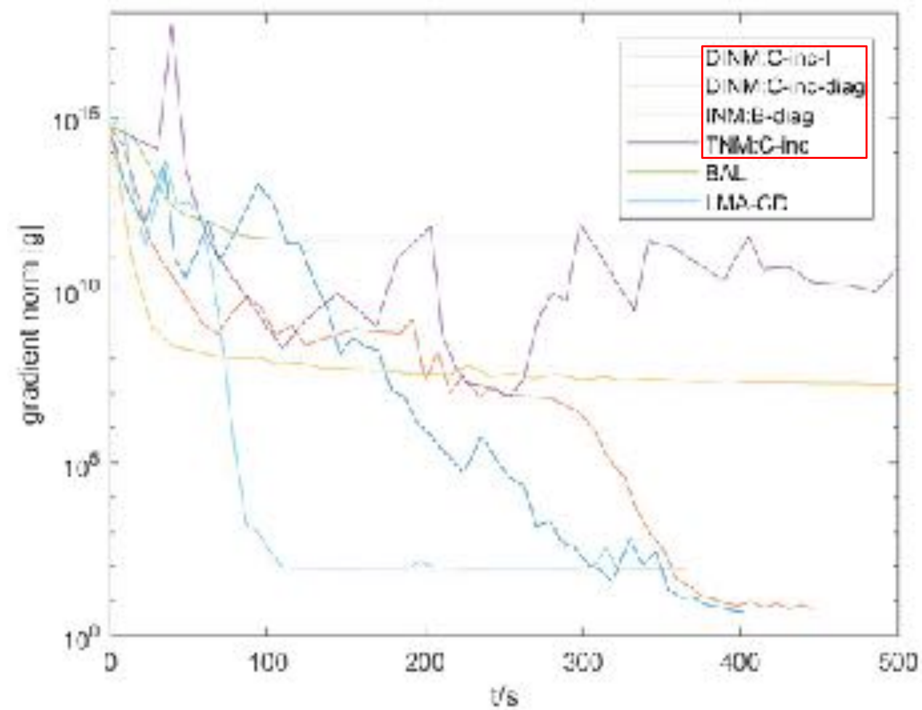
After Optimization

49 images frames, 7776 feature points, 31843 observations

Evaluation

- Objective function and gradient norm:



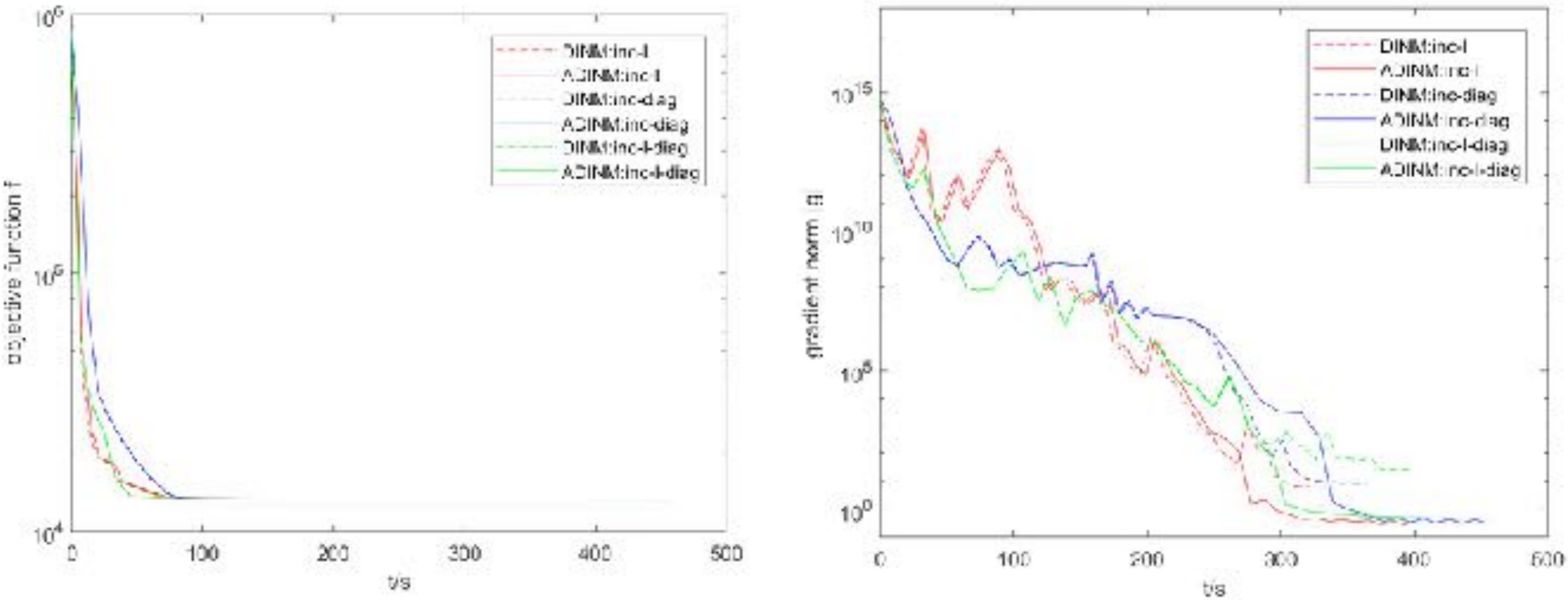


49 images frames, 7776 feature points, 31843 observations

- “DINM:C-inc-I” owns the best performance.

Adaptive DINM

- Switching between the mode of Schur complement of ***C*** and the mode of Schur complement of ***B*** in each iteration according to the condition numbers of both matrices.



- More robust, more accurate, more efficient when matrix is bad conditioned

Local Parameterization (local DINM)

- Using local parameterization instead of global parameterization

$$R_g(\phi_j + \delta\phi_j) = \mathrm{Exp}(\phi_j + \delta\phi_j) \qquad \boldsymbol{R}_l(\boldsymbol{\phi}_j + \boldsymbol{\delta\phi}_j) = \boldsymbol{R}(\boldsymbol{\phi}_j)\mathrm{Exp}(\boldsymbol{\delta\phi}_j)$$

$$\boldsymbol{t}_g(\boldsymbol{t}_j + \boldsymbol{\delta t}_j) = \boldsymbol{t}_j + \boldsymbol{\delta t}_j \qquad \boldsymbol{t}_l(\boldsymbol{t}_j + \boldsymbol{\delta t}_j) = \boldsymbol{t}_j + \boldsymbol{R}(\boldsymbol{\phi}_j)\boldsymbol{\delta t}_j$$

$$\boldsymbol{p} = \{\{\boldsymbol{\delta\phi}_j\}_{j=1}^{\mathrm{n}}, \{\boldsymbol{\delta t}_j\}_{j=1}^{\mathrm{n}}, \{\boldsymbol{\delta y}_j\}_{j=1}^{\mathrm{n}}, \{\boldsymbol{\delta x}_i\}_{i=1}^{\mathrm{m}}\}$$

$$\boldsymbol{u} = \{\{\boldsymbol{\phi}_j\}_{j=1}^{\mathrm{n}}, \{\boldsymbol{t}_j\}_{j=1}^{\mathrm{n}}, \{\boldsymbol{y}_j\}_{j=1}^{\mathrm{n}}, \{\boldsymbol{x}_i\}_{i=1}^{\mathrm{m}}\} \qquad \boldsymbol{w} = \{\{\boldsymbol{R}_j\}_{j=1}^{\mathrm{n}}, \{\boldsymbol{t}_j\}_{j=1}^{\mathrm{n}}, \{\boldsymbol{y}_j\}_{j=1}^{\mathrm{n}}, \{\boldsymbol{x}_i\}_{i=1}^{\mathrm{m}}\}$$

$$\boldsymbol{u} + \boldsymbol{p} \qquad \boldsymbol{w} \boxplus \boldsymbol{p} = \{\{\boldsymbol{R}_j\mathrm{Exp}(\boldsymbol{\delta\phi}_j)\}_{j=1}^{\mathrm{n}}, \{\boldsymbol{t}_j + \boldsymbol{R}_j\boldsymbol{\delta t}_j\}_{j=1}^{\mathrm{n}}, \{\boldsymbol{y}_j + \boldsymbol{\delta y}_j\}_{j=1}^{\mathrm{n}}, \{\boldsymbol{x}_i + \boldsymbol{\delta x}_i\}_{i=1}^{\mathrm{m}}\}$$

- More accurate linearization than global parameterization.

Reweighted Potential (IRLS)

- The percentages whose absolute values are greater than 2 in reprojection error vector are 1.60%, and the corresponding probability is 0.2% in the Gaussian distribution with the same mean and variance.

- Huber loss function:

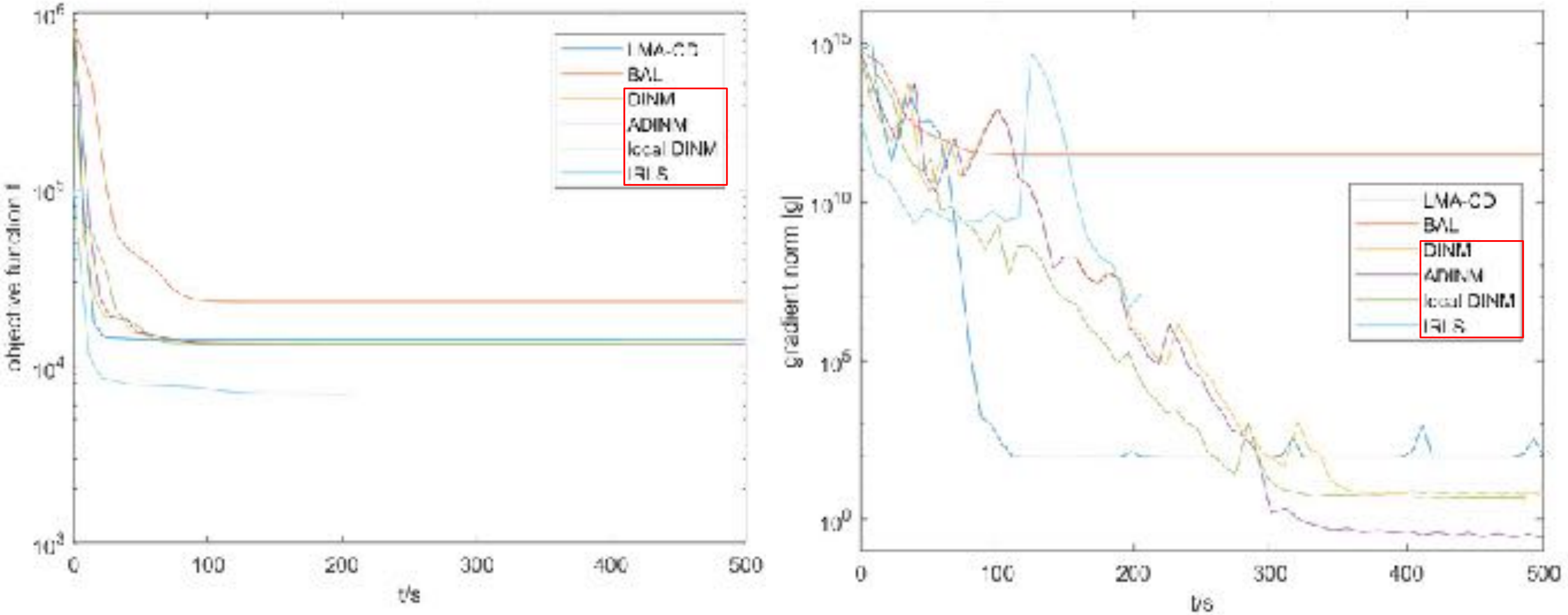
$$\psi_{f,\bullet(\gamma)}(\boldsymbol{u}) = \sum_{i=1}^{21} \Psi_{\bullet(\gamma)}(\|F_i\|)$$

$$\Psi_{Huber(\gamma)}(\|F_i\|) = \begin{cases} \|F_i\|^2/2, & \text{if } \|F_i\| \leq \gamma, \\ \gamma(\|F_i\| - \gamma/2), & \text{if } \|F_i\| > \gamma. \end{cases}$$

- Avoiding to outlier problems than without reweighting, but sometimes also not robust as without reweigting.

Evaluation

- Objective function and gradient norm on small-scale real dataset:

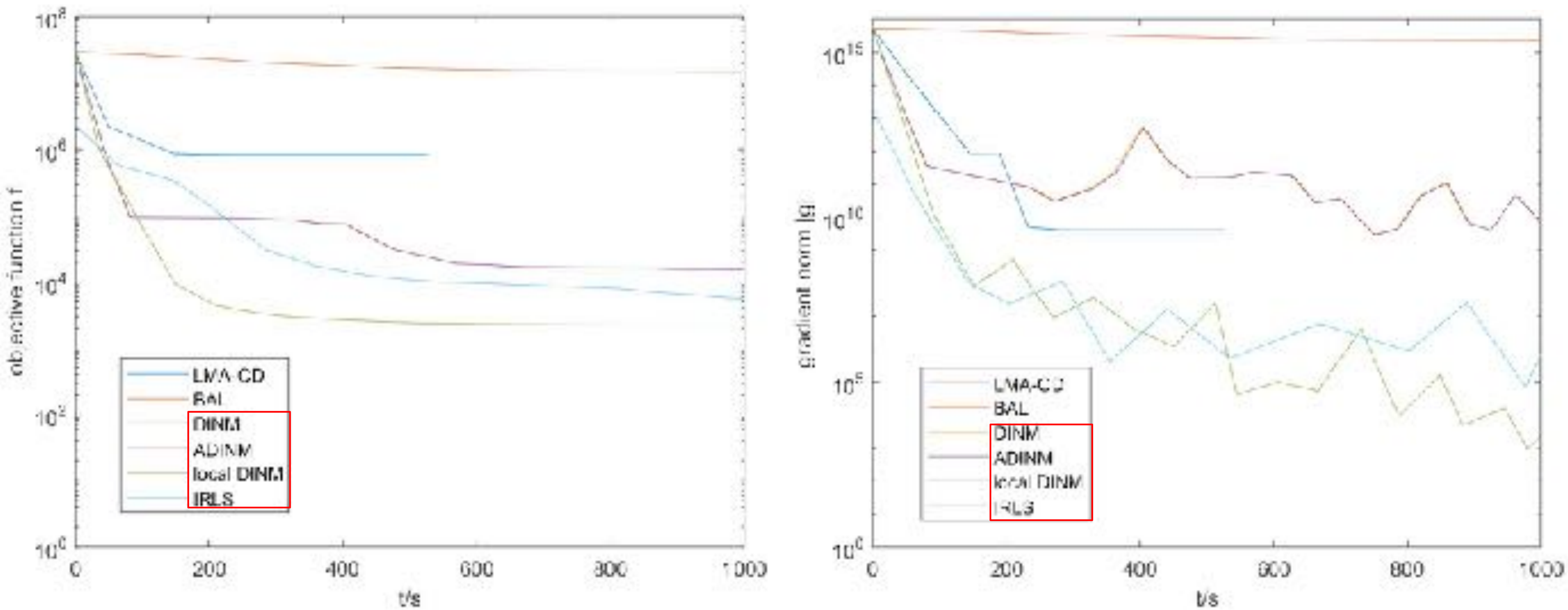


49 images frames, 7776 feature points, 31843 observations

- “ADINM” owns the best performance.

Evaluation

- Objective function and gradient norm on medium-scale synthetic dataset:

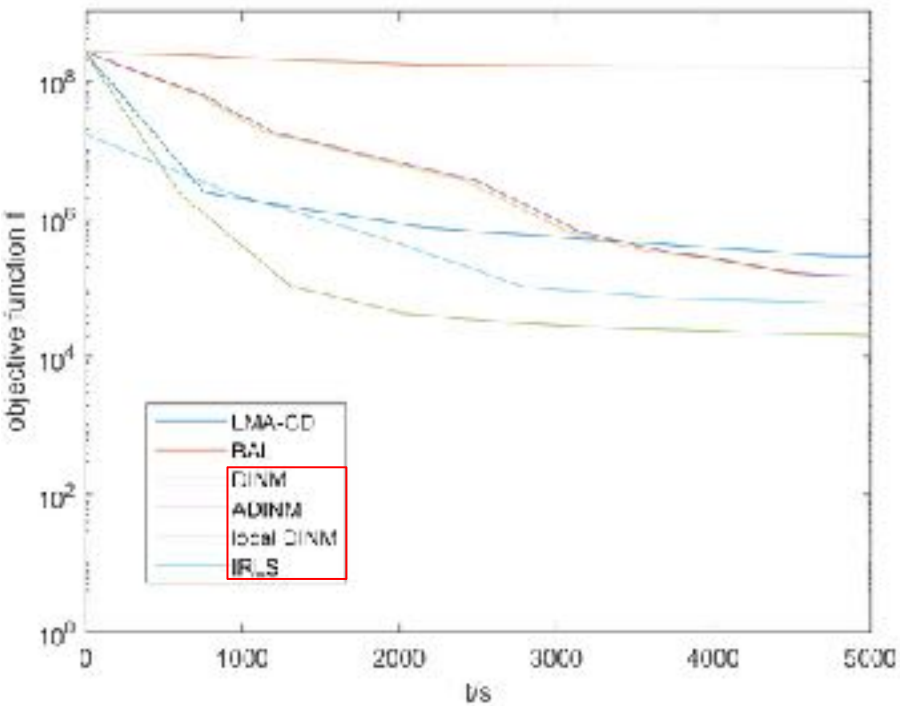


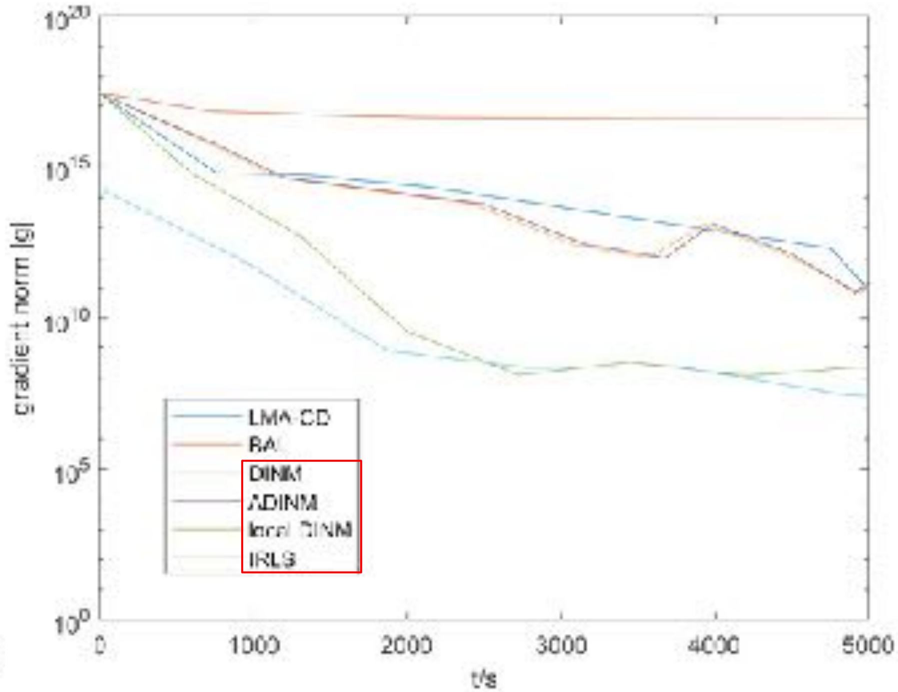
100 images frames, 4000 feature points, 228658 observations

- “local DINM” and “IRLS” own the best performance.

Evaluation

- Objective function and gradient norm on large-scale synthetic dataset:





500 images frames, 5000 feature points, 1477513 observations

- “local DINM” owns the best performance.

References

Triggs B, McLauchlan P F, Hartley R I, et al.
Bundle adjustment—a modern synthesis.
In: International workshop on vision algorithms. Springer, Berlin, Heidelberg, 1999: 298-372.

S. Agarwal, N. Snavely, S. M. Seitz, and R. Szeliski.
Bundle Adjustment in the Large.
In: *European Conference on Computer Vision (ECCV), pages 29–42, 2010*.

Ramamurthy K N, Lin C C, Aravkin A, et al.
Distributed Bundle Adjustment.
In: *arXiv:1708.07954, 2017.*