



# Multiplicative vs. Additive Half-Quadratic Minimization for Robust Cost Optimization

Christopher Zach, Guillaume Bourmaud

## ► To cite this version:

Christopher Zach, Guillaume Bourmaud. Multiplicative vs. Additive Half-Quadratic Minimization for Robust Cost Optimization. BMVC, 2018, Newcastle, United Kingdom. hal-01875291

**HAL Id: hal-01875291**

**<https://hal.science/hal-01875291>**

Submitted on 17 Sep 2018

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Multiplicative vs. Additive Half-Quadratic Minimization for Robust Cost Optimization

Christopher Zach  
christopher.zach@crl.toshiba.co.uk

Toshiba Research Europe  
Computer Vision Group  
Cambridge, UK

Guillaume Bourmaud  
guillaume.bourmaud@u-bordeaux.fr

University of Bordeaux  
Bordeaux, France

## Abstract

It has been experimentally shown in the literature that half-quadratic (HQ) lifting leads to very effective methods for robust cost minimization when combined with a joint optimization strategy. More precisely, the multiplicative formulation of HQ minimization was employed in these works. In this work we address the questions whether the complementary additive form of HQ lifting is beneficial for solving robust estimation problems. Additive HQ minimization is appealing due to its connection with a quadratic relaxation and because it fully pushes the difficulties induced by robust costs to independent terms in the (lifted) cost function. We also propose a “double lifting” method combining additive and multiplicative HQ minimization. We report numerical results for synthetic problems and standard bundle adjustment instances.

## 1 Introduction

Robust cost optimization is the task that consists in fitting parameters to data points containing outliers. This fundamental problem arises in many applications in 3D computer vision such as bundle adjustment [13], optical flow [9], registration of 3D surfaces [16], etc. When the proportion of outliers is reasonable, i.e typically less than 50%, using convex kernels<sup>1</sup>, such as Huber or Charbonnier kernels, per data point sufficiently mitigates the large residuals induced by outliers. However, when the data contains a large proportion of outliers, convex kernels are not “robust” enough and one has to rely on a quasi-convex kernel, such as Tukey’s biweight or Welsch kernels, to significantly downweight the influence of outliers. Nevertheless, summing quasi-convex functions (one for each data point/residual) creates many local minima and consequently makes the optimization problem really challenging.

In order to be useful for most computer vision applications, an optimization algorithm that addresses this challenging scenario should be both computationally efficient and able to escape poor local minima. One kind of algorithm, that satisfies these two requirements and turns out to be successful in practice [14], consists in embedding the original robust cost into a higher dimensional space by applying so-called *Multiplicative Half-Quadratic lifting* (M-HQ) [9]. An efficient non-linear-least-square (NLLS) solver is then used to jointly minimize the lifted cost w.r.t. both the parameters of interest and the *lifting variables*.

In this paper, we investigate a different type of lifting called *Additive Half-Quadratic lifting* (A-HQ) [10] and show that both A-HQ and M-HQ can be combined in order to reach better local minima than M-HQ alone while still leveraging the efficiency of an NLLS solver.

The rest of the paper is organized as follows: Section 2 discusses the related work while notations are introduced in Section 3. M-HQ, A-HQ and the combination we propose are described in Section 4. In Section 5, numerical evaluations of our approach are presented and a conclusion is provided in Section 6.

## 2 Related Work

In this section, the state of the art approaches for robust cost optimization are briefly described and compared to the novel algorithm we propose in this paper. The Iteratively Reweighted Least Squares (IRLS) algorithm [8] is the current workhorse for problems where a (very) good initialization of the parameters is available. This is due to the fact that IRLS seeks to directly minimize the robust cost with an NLLS solver. Unfortunately, IRLS gets easily trapped in a local minimum close to the initial value of the parameters. This (empirical) observation also holds for other established approaches that directly minimize the robust cost such as “Trigg’s correction” [13] and “square rooting the kernel” [9].

A different kind of algorithm that is usually referred to as *graduated optimization* algorithms [3, 4, 11, 12, 14] are explicitly designed to escape bad poor local minima. This is achieved by building a sequence of successively smoother versions of the original robust cost. On the one hand, solving that sequence of optimization problems (starting from the smoothest version and finishing with the original robust cost) usually allows to guide the optimization process towards a good local minimum. On the other hand, solving an entire sequence of optimization problems makes the whole approach computationally inefficient.

To the best of our knowledge, the only algorithm that meets our requirements (computational efficiency and ability to escape poor local minima) is the M-HQ based algorithm presented in [14, 12]. This approach consists in embedding the original robust cost into a higher dimensional space by introducing one *Multiplicative Lifting Variable* (M-LV) per data point. The lifted cost is then jointly minimized w.r.t. the parameters of interest and M-LVs using an NLLS solver. This joint optimization allows to escape bad local minima while each iteration is very efficient thanks to the NLLS solver. The case of multiple M-LVs per residuals is considered in [15]. In this paper, instead of embedding the original robust cost into a higher dimensional space as in [14], we investigate a different type of lifting called *Additive Half-Quadratic lifting* and demonstrate that both M-HQ and A-HQ can actually be combined to obtain an algorithm outperforming the M-HQ based algorithm of [14].

## 3 Notations & Preliminaries

In this paper, we are interested in minimizing cost functions  $\Psi(\mathbf{x})$  (w.r.t.  $\mathbf{x}$ ) of the following form:

$$\Psi(\mathbf{x}) = \sum_i \psi(\|\mathbf{f}_i(\mathbf{x})\|) \quad (1)$$

where  $\mathbf{x} \in \mathbb{R}^p$  are the parameters of interest,  $\mathbf{f}_i : \mathbb{R}^p \mapsto \mathbb{R}^d$  is the residual function corresponding to data point  $\mathbf{y}_i \in \mathbb{R}^d$ ,  $\|\cdot\|$  is the L2-norm and  $\psi : \mathbb{R} \mapsto \mathbb{R}_0^+$  is a robust kernel. More

precisely,  $\psi$  should satisfy the following properties:

- i)  $\psi(0) = 0$  and  $\psi''(0) = 1$  which are normalization properties that allow to compare the robustness of different kernels, and
- ii)  $\phi : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$  defined via  $\phi(\|\mathbf{f}_i(\mathbf{x})\|^2/2) := \psi(\|\mathbf{f}_i(\mathbf{x})\|)$  is a concave and monotonically increasing function. Here the monotonically increasing property means that a large residual should lead to a higher cost than a small residual, while the concavity allows to mitigate the influence of large residuals. We also introduce the *weight function* of a robust kernel that will be used in the rest of the paper,

$$\omega(\|\mathbf{f}_i(\mathbf{x})\|) := \psi'(\|\mathbf{f}_i(\mathbf{x})\|)/\|\mathbf{f}_i(\mathbf{x})\| = \phi'(\|\mathbf{f}_i(\mathbf{x})\|^2/2). \quad (2)$$

## 4 Half-quadratic lifting of a robust cost function

In this section, we present three ways of lifting a robust cost function using half-quadratic lifting. We first briefly describe the M-HQ that was proposed in [14]. Then we present two contributions, one based on A-HQ and another one combining both A-HQ and M-HQ.

### 4.1 Multiplicative Half-Quadratic Lifting

Multiplicative Half-Quadratic lifting [8, 14] of a robust kernel  $\psi$  consists in introducing an M-LV and rewriting the robust kernel  $\psi$  as follows:

$$\psi(\|\mathbf{f}_i(\mathbf{x})\|) = \min_{v_i \in [0,1]} \psi^{\text{M-HQ}}(\mathbf{x}, v_i) \quad \text{with} \quad \psi^{\text{M-HQ}}(\mathbf{x}, v_i) = \frac{1}{2} v_i \|\mathbf{f}_i(\mathbf{x})\|^2 + \gamma(v_i), \quad (3)$$

where  $v_i \in [0, 1]$  is an M-LV and  $\gamma : [0, 1] \rightarrow \mathbb{R}_0^+$  is a convex and monotonically decreasing “bias” function. Intuitively, an M-LV multiplies the squared norm of the residual of a data point, thus if an M-LV reaches a value close to 0 (resp. 1), the corresponding data point is considered to be an outlier (resp. inlier). Plugging Eq. 3 in Eq. 1, we obtain the M-HQ lifted cost function

$$\Psi^{\text{M-HQ}}(\mathbf{x}, \{v_i\}_i) = \frac{1}{2} \sum_i \|\sqrt{v_i} \mathbf{f}_i(\mathbf{x})\|^2 + \sqrt{\gamma(v_i)}^2 = \frac{1}{2} \sum_i \left\| \frac{\sqrt{v_i} \mathbf{f}_i(\mathbf{x})}{\sqrt{\gamma(v_i)}} \right\|^2, \quad (4)$$

where we wrote  $\Psi^{\text{M-HQ}}$  as non-linear least-squares instance. In order to jointly optimize over the M-LVs and  $\mathbf{x}$  using an NLLS solver, the M-LVs are reparameterized by  $v = w(u)$  (which allows to avoid the constraint  $v \geq 0$ <sup>2</sup>), where  $w : \mathbb{R} \rightarrow \mathbb{R}_0^+$ . Two sensible choices for  $w$  are  $w(u) = u^2$  and  $w(u) = e^u$ . For instance, the numerically convenient choice  $w(u) = u^2$  and  $\gamma(v) = \tau^2(v-1)^2/4$  (and therefore  $\sqrt{\gamma(w(u))} = \tau(u^2-1)/2$ ) yields a smooth truncated kernel [17],

$$\psi_{ST}(\|\mathbf{f}\|) = \frac{\tau^2}{4} \left( 1 - \left[ 1 - \frac{\|\mathbf{f}\|^2}{\tau^2} \right]_+^2 \right). \quad (5)$$

The non-differentiability of  $\sqrt{\gamma(v)}$  at  $v = 1$  can be avoided e.g. by multiplication with  $\text{sgn}(v-1)$  (which leads to  $\sqrt{\gamma(v)} = \frac{\tau}{2}(v-1)$  instead of  $\sqrt{\gamma(v)} = \frac{\tau}{2}|v-1|$  for the above-mentioned kernel  $\psi_{ST}$ ).

In practice, one M-LV is introduced per data point but this only induces a moderate increase in run-time (e.g. by leveraging the Schur complement [14]) compared to IRLS while allowing the algorithm to escape poor local minima.

<sup>2</sup>For standard choices of  $\psi$ ,  $\gamma$  can be continuously extended to the domain  $\mathbb{R}_0^+$  (see e.g. [14]).

## 4.2 Additive Half-Quadratic Lifting

In this section we propose a novel lifting-based optimization approach by leveraging the additive variant of half-quadratic minimization [10]. It consists in introducing an A-LV and rewriting  $\psi$  as follows:

$$\psi(\|\mathbf{f}_i(\mathbf{x})\|) = \min_{\mathbf{p}_i} \psi^{\text{A-HQ}}(\mathbf{x}, \mathbf{p}_i) \quad \text{with} \quad \psi^{\text{A-HQ}}(\mathbf{x}, \mathbf{p}_i) = \frac{\alpha}{2} \|\mathbf{f}_i(\mathbf{x}) - \mathbf{p}_i\|^2 + \rho(\|\mathbf{p}_i\|), \quad (6)$$

where  $\mathbf{p}_i \in \mathbb{R}^d$  is an A-LV and  $\rho : \mathbb{R}_0^+ \rightarrow \mathbb{R}_0^+$  is generally a function similar to a robust kernel. Intuitively, an A-LV is subtracted from the residual of a data point, thus if an A-LV reaches a value close to the residual itself (resp.  $\mathbf{0}$ ), the corresponding data point is considered to be an outlier (resp. inlier). Since  $\rho$  has usually no closed-form expression for many robust kernels  $\psi$ , but at the same time  $\rho$  is very close to  $\psi$  for  $\alpha \gg 1$ , we replace  $\psi^{\text{A-HQ}}$  by its relaxation,

$$\tilde{\psi}^{\text{A-HQ}}(\mathbf{x}, \mathbf{p}_i) = \frac{\alpha}{2} \|\mathbf{f}_i(\mathbf{x}) - \mathbf{p}_i\|^2 + \psi(\|\mathbf{p}_i\|). \quad (7)$$

Observe that  $\tilde{\psi}^{\text{A-HQ}}$  is nothing else than using a quadratic penalizer for the equality constraint  $\mathbf{p}_i = \mathbf{f}_i(\mathbf{x})$ . In M-HQ one has non-convex interactions between lifting variables and the (linearized) residuals, whereas in A-HQ these interactions are convex and the non-convex aspects of the objective are entirely addressed by the lifting variables. This makes A-HQ an interesting choice. Plugging Eq. 7 in Eq. 1, we obtain the A-HQ lifted cost function

$$\Psi^{\text{A-HQ}}(\mathbf{x}, \{\mathbf{p}_i\}_i) = \frac{\alpha}{2} \sum_i \|\mathbf{f}_i(\mathbf{x}) - \mathbf{p}_i\|^2 + \psi(\|\mathbf{p}_i\|). \quad (8)$$

In order to minimize the A-HQ lifted cost function (Eq. 8) using an NLLS solver, we consider the following quadratic majorizer for  $\psi(\|\mathbf{p}_i\|)$ :

$$\psi(\|\bar{\mathbf{p}}_i + \Delta \mathbf{p}_i\|) \leq \omega(\|\bar{\mathbf{p}}_i\|) \frac{\|\bar{\mathbf{p}}_i + \Delta \mathbf{p}_i\|^2 - \|\bar{\mathbf{p}}_i\|^2}{2} + \psi(\|\bar{\mathbf{p}}_i\|). \quad (9)$$

where  $\omega(\cdot)$  is defined in Eq. 2. This majorizer arises in the derivation of the IRLS algorithm as an instance of the majorize-minimize principle (e.g. [10]). Each iteration of an NLLS solver (such as Gauss-Newton or Levenberg-Marquardt) then consists in:

- 1 - Applying a Gauss-Newton approximation of  $\|\mathbf{f}_i(\mathbf{x}) - \mathbf{p}_i\|^2$ , i.e. linearizing the term  $\mathbf{f}_i(\mathbf{x}) - \mathbf{p}_i$  around  $\bar{\mathbf{x}}$  and  $\bar{\mathbf{p}}_i$  (the current values of  $\mathbf{x}$  and  $\mathbf{p}_i$ ):  $\mathbf{f}_i(\bar{\mathbf{x}} + \Delta \mathbf{x}) - \bar{\mathbf{p}}_i - \Delta \mathbf{p}_i \approx \bar{\mathbf{r}}_i + \mathbf{J}_i \Delta \mathbf{x} - \bar{\mathbf{p}}_i - \Delta \mathbf{p}_i$ , where  $\bar{\mathbf{r}}_i = \mathbf{f}_i(\bar{\mathbf{x}})$  and  $\mathbf{J}_i$  is the Jacobian matrix of  $\mathbf{f}_i(\bar{\mathbf{x}} + \Delta \mathbf{x})$  around  $\Delta \mathbf{x} = \mathbf{0}$ .
- 2 - Replacing  $\psi(\|\bar{\mathbf{p}}_i + \Delta \mathbf{p}_i\|)$  by its quadratic majorizer (Eq. 9).
- 3 - Solving the resulting (large) linear least squares problem (or a damped version of it by adding a damping parameter  $\lambda$  to the diagonal of the approximated Hessian matrix). We now show how to efficiently solve this large linear system.

For simplicity, in the following we will focus on a two residual scenario but the results we will derive can be trivially extended to multiple residuals. From Eq. 8 and Eq. 9, introducing the notation  $\bar{\omega}_i = \omega(\|\bar{\mathbf{p}}_i\|)/\alpha$ , multiplying the cost by  $2/\alpha$  and linearizing the residuals as described above, we obtain the following linear least squares problem:

$$\mathring{\Psi}^{\text{A-HQ}}(\Delta \mathbf{x}, \Delta \mathbf{p}_1, \Delta \mathbf{p}_2) = \sum_i \|\bar{\mathbf{r}}_i + \mathbf{J}_i \Delta \mathbf{x} - \bar{\mathbf{p}}_i - \Delta \mathbf{p}_i\|^2 + \bar{\omega} \|\bar{\mathbf{p}}_i + \Delta \mathbf{p}_i\|^2 = \|\mathring{\mathbf{r}} + \mathring{\mathbf{J}} \Delta\|^2, \quad (10)$$

where (leaving zero entries blank)

$$\mathring{\mathbf{r}} = \begin{pmatrix} \bar{\mathbf{r}}_1 - \bar{\mathbf{p}}_1 \\ \bar{\mathbf{r}}_2 - \bar{\mathbf{p}}_2 \\ \sqrt{\bar{\omega}_1} \bar{\mathbf{p}}_1 \\ \sqrt{\bar{\omega}_2} \bar{\mathbf{p}}_2 \end{pmatrix} \quad \mathring{\mathbf{J}} = \begin{pmatrix} \mathbf{J}_1 & -\mathbf{I} & \\ \mathbf{J}_2 & & -\mathbf{I} \\ & \sqrt{\bar{\omega}_1} \mathbf{I} & \\ & & \sqrt{\bar{\omega}_2} \mathbf{I} \end{pmatrix} \quad \Delta = \begin{pmatrix} \Delta \mathbf{x} \\ \Delta \mathbf{p}_1 \\ \Delta \mathbf{p}_2 \end{pmatrix}.$$

Thus we obtain the following (approximated) Hessian and gradient:

$$\mathring{\mathbf{J}}^\top \mathring{\mathbf{J}} = \begin{pmatrix} \sum_i \mathbf{J}_i^\top \mathbf{J}_i & -\mathbf{J}_1^\top & -\mathbf{J}_2^\top \\ -\mathbf{J}_1 & (1 + \bar{\omega}_1) \mathbf{I} & \\ -\mathbf{J}_2 & & (1 + \bar{\omega}_2) \mathbf{I} \end{pmatrix} \quad \mathring{\mathbf{J}}^\top \mathring{\mathbf{r}} = \begin{pmatrix} \sum_i \mathbf{J}_i^\top (\bar{\mathbf{r}}_i - \bar{\mathbf{p}}_i) \\ (1 + \bar{\omega}_1) \bar{\mathbf{p}}_1 - \bar{\mathbf{r}}_1 \\ (1 + \bar{\omega}_2) \bar{\mathbf{p}}_2 - \bar{\mathbf{r}}_2 \end{pmatrix}.$$

Using the Schur complement and a damped approximated Hessian ( $\mathring{\mathbf{J}}^\top \mathring{\mathbf{J}} + \lambda \mathbf{I}$ ), we obtain for the update  $\Delta \mathbf{x}$ ,

$$\left( \lambda \mathbf{I} + \sum_i \frac{\bar{\omega}_i + \lambda}{1 + \bar{\omega}_i + \lambda} \mathbf{J}_i^\top \mathbf{J}_i \right) \Delta \mathbf{x} = - \sum_i \mathbf{J}_i^\top \frac{(\bar{\omega}_i + \lambda) \bar{\mathbf{r}}_i - \lambda \bar{\mathbf{p}}_i}{1 + \bar{\omega}_i + \lambda}. \quad (11)$$

Backsubstitution for  $\Delta \mathbf{p}_i$  yields

$$\Delta \mathbf{p}_i = (1 + \bar{\omega}_i + \lambda)^{-1} (\mathbf{J}_i \Delta \mathbf{x} + \bar{\mathbf{r}}_i - (1 + \bar{\omega}_i) \bar{\mathbf{p}}_i). \quad (12)$$

Equations 11 and 12 allow to efficiently solve the linear system appearing in each iteration of an NLLS solver since only a linear system of the size of  $\mathbf{J}_i^\top \mathbf{J}_i$  needs to be solved to obtain  $\Delta \mathbf{x}$  while a simple scalar inversion is required to compute  $\Delta \mathbf{p}_i$ . The system matrix appearing in Eq. 11 has the same non-zero pattern as the non-lifted problem. Thus, the novel algorithm we just derived meets our computational efficiency requirement. In the experiments, we will evaluate if this version of half-quadratic lifting allows to satisfy our second requirement, i.e. the ability to escape poor local minima.

**Link with IRLS** Since the A-HQ lifting approach we derived uses the same quadratic majorizer as IRLS, we investigate in this paragraph how the two approaches are related. Recall that  $\bar{\omega}_i = \omega_i / \alpha$  (with  $\omega_i = \omega(\|\bar{\mathbf{p}}_i\|)$ ). Thus we read

$$\frac{\bar{\omega}_i}{1 + \bar{\omega}_i} = \frac{\omega_i / \alpha}{1 + \omega_i / \alpha} = \frac{\omega_i}{\alpha + \omega_i} \quad \frac{1}{1 + \bar{\omega}_i} = \frac{1}{1 + \omega_i / \alpha} = \frac{\alpha}{\alpha + \omega_i}. \quad (13)$$

If we assume  $\bar{\mathbf{p}}_i = \bar{\mathbf{r}}_i$  and  $\lambda = 0$ , then the update for  $\mathbf{x}$  (Eq. 11) becomes

$$\left( \sum_i \frac{\bar{\omega}_i}{1 + \bar{\omega}_i} \mathbf{J}_i^\top \mathbf{J}_i \right) \Delta \mathbf{x} = - \sum_i \frac{\bar{\omega}_i}{1 + \bar{\omega}_i} \mathbf{J}_i^\top \bar{\mathbf{r}}_i. \quad (14)$$

Consequently, if  $\alpha \gg 1$ , then the factor  $(1 + \bar{\omega}_i)^{-1}$  in Eq. 14 is essentially constant and the equation for  $\Delta \mathbf{x}$  becomes the same as the IRLS update step. Further, in this setting we have (from Eq. 12):

$$\Delta \mathbf{p}_i = \frac{1}{1 + \bar{\omega}_i} (\mathbf{J}_i \Delta \mathbf{x} - \bar{\omega}_i \bar{\mathbf{r}}_i) = \frac{\alpha}{\alpha + \omega_i} \mathbf{J}_i \Delta \mathbf{x} - \frac{\omega_i}{\alpha + \omega_i} \bar{\mathbf{r}}_i.$$

Again, if  $\alpha \gg 1$ , then  $\Delta \mathbf{p}_i \approx \mathbf{J}_i \Delta \mathbf{x}$  and  $\bar{\mathbf{p}}_i + \Delta \mathbf{p}_i \approx \bar{\mathbf{r}}_i + \mathbf{J}_i \Delta \mathbf{x}$ , which is the new residual. Consequently, an iteration of the A-HQ lifting approach essentially reduces to an IRLS iteration for  $\bar{\mathbf{p}}_i \approx \bar{\mathbf{r}}_i$ ,  $\lambda$  close to zero and  $\alpha \gg 1$ .

### 4.3 Double Lifting

In the previous section, we investigated one way of applying A-HQ to lift a robust cost function. More precisely, we used a quadratic majorizer of  $\psi(\|\mathbf{p}_i\|)$  in order to be able to leverage the computational efficiency of an NLLS solver. Instead of using a quadratic majorizer, in this section, we propose to use the M-HQ method to lift  $\psi(\|\mathbf{p}_i\|)$  in Eq. 8 in order to obtain an NLLS instance. As a consequence, we call this method “Double Lifting” (DL). Using Eq. 8 and Eq. 3 we obtain the “doubly” lifted cost function:

$$\Psi^{\text{DL}}(\mathbf{x}, \{\mathbf{p}_i, u_i\}_i) = \frac{\alpha}{2} \sum_i \|\mathbf{f}_i(\mathbf{x}) - \mathbf{p}_i\|^2 + \frac{1}{2} \|\sqrt{w(u_i)} \mathbf{p}_i\|^2 + \sqrt{\gamma(w(u_i))}. \quad (15)$$

We now show how to efficiently minimize that cost function using an NLLS solver. In the following we will focus again on a single iteration of an NLLS solver and a two residual scenario but the results we will derive can be trivially extended to multiple residuals. We start by applying a Gauss-Newton approximation of each term in Eq. 15 around  $\bar{\mathbf{x}}$ ,  $\bar{\mathbf{p}}_1$ ,  $\bar{u}_1$ ,  $\bar{\mathbf{p}}_2$  and  $\bar{u}_2$  (the current values of  $\mathbf{x}$ ,  $\mathbf{p}_1$ ,  $u_1$ ,  $\mathbf{p}_2$  and  $u_2$ ) and obtain the following linear least squares problem:

$$\check{\Psi}^{\text{DL}}(\Delta \mathbf{x}, \Delta \mathbf{p}_1, \Delta u_1, \Delta \mathbf{p}_2, \Delta u_2) = \|\check{\mathbf{r}} + \check{\mathbf{J}} \Delta\|^2, \quad (16)$$

with  $\Delta = [\Delta \mathbf{x}; \Delta \mathbf{p}_1; \Delta u_1; \Delta \mathbf{p}_2; \Delta u_2]$  and

$$\check{\mathbf{r}} = \begin{pmatrix} \sqrt{\alpha}(\bar{\mathbf{r}}_1 - \bar{\mathbf{p}}_1) \\ \sqrt{\alpha}(\bar{\mathbf{r}}_2 - \bar{\mathbf{p}}_2) \\ \sqrt{\bar{w}_1} \cdot \bar{\mathbf{p}}_1 \\ \sqrt{\bar{w}_2} \cdot \bar{\mathbf{p}}_2 \\ \sqrt{2\bar{\gamma}_1} \\ \sqrt{2\bar{\gamma}_2} \end{pmatrix} \quad \check{\mathbf{J}} = \begin{pmatrix} \sqrt{\alpha} \mathbf{J}_1 & -\sqrt{\alpha} \mathbf{I} & & & & \\ \sqrt{\alpha} \mathbf{J}_2 & & & -\sqrt{\alpha} \mathbf{I} & & \\ & \sqrt{\bar{w}_1} \mathbf{I} & \frac{\bar{w}'_1}{2\sqrt{\bar{w}_1}} \bar{\mathbf{p}}_1 & & & \\ & & \sqrt{\bar{w}_2} \mathbf{I} & \frac{\bar{w}'_2}{2\sqrt{\bar{w}_2}} \bar{\mathbf{p}}_2 & & \\ & & & \frac{\bar{w}'_1 \bar{\gamma}'_1}{\sqrt{2\bar{\gamma}_1}} & & \\ & & & & \frac{\bar{w}'_2 \bar{\gamma}'_2}{\sqrt{2\bar{\gamma}_2}} & \end{pmatrix}.$$

$\bar{\mathbf{r}}_i = \mathbf{f}_i(\bar{\mathbf{x}})$ ,  $\mathbf{J}_i$  is the Jacobian matrix of  $\mathbf{f}_i(\bar{\mathbf{x}} + \Delta \mathbf{x})$  around  $\Delta \mathbf{x} = \mathbf{0}$ ,  $\bar{w}_i = w(\bar{u}_i)$ ,  $\bar{w}'_i = w'(\bar{u}_i)$ ,  $\bar{\gamma}_i = \gamma(\bar{w}_i)$  and  $\bar{\gamma}'_i$  is the derivative of  $\gamma(w(u_i))$  around the value  $w(\bar{u}_i)$ . Thus we obtain the following (approximated) Hessian and gradient:

$$\check{\mathbf{J}}^\top \check{\mathbf{J}} = \begin{pmatrix} \alpha \sum_i \mathbf{J}_i^\top \mathbf{J}_i & -\alpha \mathbf{J}_1^\top & & & -\alpha \mathbf{J}_2^\top \\ -\alpha \mathbf{J}_1 & (\alpha + \bar{w}_1) \mathbf{I} & \frac{\bar{w}'_1}{2} \bar{\mathbf{p}}_1 & & \\ & \frac{\bar{w}'_1}{2} \bar{\mathbf{p}}_1^\top & \frac{(\bar{w}'_1)^2}{4\bar{w}_1} \|\bar{\mathbf{p}}_1\|^2 + \frac{(\bar{w}'_1 \bar{\gamma}'_1)^2}{2\bar{\gamma}_1} & & \\ -\alpha \mathbf{J}_2 & & & (\alpha + \bar{w}_2) \mathbf{I} & \frac{\bar{w}'_2}{2} \bar{\mathbf{p}}_2 \\ & \frac{\bar{w}'_2}{2} \bar{\mathbf{p}}_2^\top & \frac{(\bar{w}'_2)^2}{4\bar{w}_2} \|\bar{\mathbf{p}}_2\|^2 + \frac{(\bar{w}'_2 \bar{\gamma}'_2)^2}{2\bar{\gamma}_2} & & \end{pmatrix}$$

$$\check{\mathbf{J}}^\top \check{\mathbf{r}} = \left[ \alpha \sum_i \mathbf{J}_i^\top (\bar{\mathbf{r}}_i - \bar{\mathbf{p}}_i); \alpha(\bar{\mathbf{p}}_1 - \bar{\mathbf{r}}_1) + \bar{w}_1 \bar{\mathbf{p}}_1; \bar{w}'_1 \frac{\|\bar{\mathbf{p}}_1\|^2 + 2\bar{\gamma}'_1}{2}; \alpha(\bar{\mathbf{p}}_2 - \bar{\mathbf{r}}_2) + \bar{w}_2 \bar{\mathbf{p}}_2; \bar{w}'_2 \frac{\|\bar{\mathbf{p}}_2\|^2 + 2\bar{\gamma}'_2}{2} \right].$$

In order to be able to apply the Schur complement and solve the previous linear system efficiently, we will need the matrix-inverse of the following submatrix of  $\check{\mathbf{J}}^\top \check{\mathbf{J}} + \lambda \mathbf{I}$ ,

$$\mathbf{A}_i := \begin{pmatrix} (\alpha + \bar{w}_i + \lambda) \mathbf{I} & \frac{\bar{w}'_i}{2} \bar{\mathbf{p}}_i \\ \frac{\bar{w}'_i}{2} \bar{\mathbf{p}}_i^\top & \frac{(\bar{w}'_i)^2}{4\bar{w}_i} \|\bar{\mathbf{p}}_i\|^2 + \frac{(\bar{w}'_i \bar{\gamma}'_i)^2}{2\bar{\gamma}_i} + \lambda \end{pmatrix} =: \begin{pmatrix} a_i \mathbf{I} & b_i \bar{\mathbf{p}}_i \\ b_i \bar{\mathbf{p}}_i^\top & c_i \end{pmatrix}. \quad (17)$$

It can be verified that

$$\mathbf{A}_i^{-1} = \frac{1}{a_i c_i - b_i^2 \|\mathbf{p}_i\|^2} \begin{pmatrix} \frac{a_i c_i - b_i^2 \|\mathbf{p}_i\|^2}{a_i} \mathbf{I} + \frac{b_i^2}{a_i} \mathbf{p}_i \mathbf{p}_i^\top & -b_i \mathbf{p}_i \\ -b_i \mathbf{p}_i^\top & a_i \end{pmatrix} = \begin{pmatrix} \mathbf{B}_i & \mathbf{g}_i \\ \mathbf{g}_i^\top & h_i \end{pmatrix}, \quad (18)$$

where  $\mathbf{B}_i$  is the upper left  $d \times d$  block of  $\mathbf{A}_i^{-1}$  and  $\mathbf{g}_i$  is the  $d$ -dimensional column vector at the top right of  $\mathbf{A}_i^{-1}$ . Using the Schur complement and a damped approximated Hessian ( $\check{\mathbf{J}}^\top \check{\mathbf{J}} + \lambda \mathbf{I}$ ), we obtain for the update  $\Delta \mathbf{x}$ ,

$$\begin{aligned} & \left( \sum_i \mathbf{J}_i^\top (\mathbf{I} - \alpha \mathbf{B}_i) \mathbf{J}_i + \lambda \mathbf{I} \right) \Delta \mathbf{x} = \\ & - \sum_i \mathbf{J}_i^\top \left( \bar{\mathbf{r}}_i - \bar{\mathbf{p}}_i + \mathbf{B}_i (\alpha (\bar{\mathbf{p}}_i - \bar{\mathbf{r}}_i) + \bar{w}_i \bar{\mathbf{p}}_i) + \bar{w}'_i \left( \frac{\|\bar{\mathbf{p}}_i\|^2}{2} + \bar{\gamma}'_i \right) \mathbf{g}_i \right). \end{aligned} \quad (19)$$

Backsubstitution for  $\Delta \mathbf{p}_i$  and  $u_i$  yields

$$\begin{pmatrix} \Delta \mathbf{p}_i \\ \Delta u_i \end{pmatrix} = -\mathbf{A}_i^{-1} \begin{pmatrix} \alpha (\bar{\mathbf{p}}_i - \bar{\mathbf{r}}_i) + \bar{w}_i \bar{\mathbf{p}}_i - \alpha \mathbf{J}_i \Delta \mathbf{x} \\ \bar{w}'_i \left( \frac{1}{2} \|\bar{\mathbf{p}}_i\|^2 + \bar{\gamma}'_i \right) \end{pmatrix}. \quad (20)$$

Equations 19 and 20 allow to efficiently solve the linear system appearing in each iteration of an NLLS solver. Thus our novel DL approach also fulfills our computational efficiency requirement. The ability of DL to escape poor local minima will be evaluated experimentally and compared against both M-HQ and A-HQ in section 5.

## 5 Experiments

Our straightforward C++ implementation of all algorithms is based on a sparse Levenberg-Marquardt solver leveraging column reordering and sparse  $\text{LDL}^\top$  decomposition. We also use the explicit Schur complement as described in the previous sections to handle the half-quadratic based approaches. Using a direct solver puts limits on the maximum bundle adjustment problem size. The parameter  $\alpha$  is set to 10 in all experiments, since for this value of  $\alpha$  the bias function  $\rho$  in the additive HQ formulation is already close to  $\psi$ . In terms of run-time the IRLS method is the fastest (per LM iteration), followed by the HQ variants. The relative run-time indices are IRLS: 1, M-HQ: 1.6, A-HQ: 1.45, DL: 1.72.

**Synthetic toy data:** We follow [15] in the setup of the synthetic robust mean instances in order to determine the behavior of the methods w.r.t. random initial values. The robust objective is given by

$$\min_{\theta} \sum_{i=1}^N \psi(\|\mathbf{y}_i - \theta\|), \quad (21)$$

where  $\psi$  is the Welsch kernel with parameter  $1/2$ . The  $N = 1000$   $d$ -dimensional data points consist of uniformly sampled in  $[-20, 20]^d$  outliers and inliers sampled from  $\mathcal{N}(\mu, \mathbf{I}_{d \times d})$ . Fig. 1 depicts the average objective values (and the respective standard deviation over 100 runs) reached by the different methods for varying choices of inlier ratios and  $d \in \{2, 3\}$ . Double lifting significantly outperforms multiplicative HQ in this scenario, and additive HQ is superior to M-HQ in most cases.



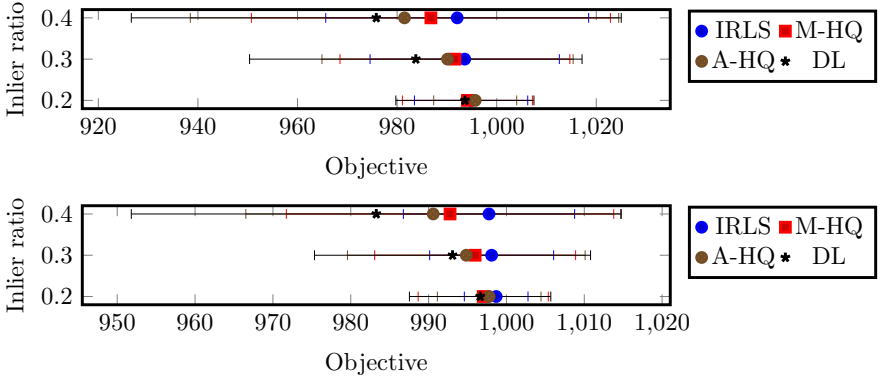


Figure 1: Synthetic results in 2D (top) and 3D (bottom).

**Bundle adjustment data:** Bundle adjustment is the prominent application of large scale robust cost minimization in 3D computer vision. We selected 10 up-to-medium sized problem instances from the “bundle adjustment in the large collection” [14].<sup>3</sup> The robust objective is given by

$$\Psi^{\text{BA}}(\{\mathbf{R}_i\}_i, \{\mathbf{t}_i\}_i, \{\mathbf{X}_j\}_j) := \sum_{i,j} \psi(\|\pi(\mathbf{R}_i \mathbf{X}_j + \mathbf{t}_i) - \mathbf{q}_{ij}\|), \quad (22)$$

where  $\mathbf{q}_{ij} \in \mathbb{R}^2$  is the observed image observation of the  $j$ -th 3D point  $\mathbf{X}_j \in \mathbb{R}^3$  point in the  $i$ -th image (which has associated parameters  $\mathbf{R}_i \in SO(3)$  and  $\mathbf{t}_i \in \mathbb{R}^3$ ).  $\pi(X) = (X_1/X_3, X_2/X_3)^T$  is the projection function of a pinhole camera model.  $\mathbf{q}_{ij}$  is measured on the normalized image plane, i.e. the original pixel coordinates are premultiplied by the (provided) inverse calibration matrix. We use both the Welsch and the smooth truncated kernel with parameter  $1/2$ . This parameter value renders the problem instances sufficiently difficult, as the initial inlier ratio of image observations varies between 14% and 50% (depending on the dataset). The inlier ratios obtained after robust cost minimization cluster around 60% for the best obtained local minima. We limit the number of LM iterations to at most 100 to avoid excessive run-times.

In order to separate between local minima induced by the non-linear bundle objective and the ones induced by the robustified cost function we first consider a *linearized* version of bundle adjustment, i.e. the residuals  $\mathbf{f}_{ij} = \pi(\mathbf{R}_i \mathbf{X}_j + \mathbf{t}_i) - \mathbf{q}_{ij}$  are replaced by their linearized versions w.r.t. the provided initial values. The non-robust objective is therefore convex (a linear least-squares instance), and the performance differences depicted in Fig. 2 indicate how well each method escapes poor local minima. In this setting A-HQ generally outperforms IRLS by a margin (and is competitive with M-HQ and double lifting for the Welsch kernel), and double lifting is slightly ahead of M-HQ.

Fig. 3 illustrates the reached objectives by the different methods for non-linear metric bundle adjustment. Due to the non-linearity already present in the non-robust objective, the results are more diverse than the ones in Fig. 2. In particular, dataset 2 is leading to slow convergence for both A-HQ and DL methods. Hence, a better method for robust cost minimization is no guarantee for reaching a better minimum under all circumstances.

<sup>3</sup>The datasets are in particular ladybug-318, ladybug-598, trafilgar-138, trafilgar-257, dubrovnik-150, dubrovnik-356, venice-245, venice-427, final-93 and final-394.

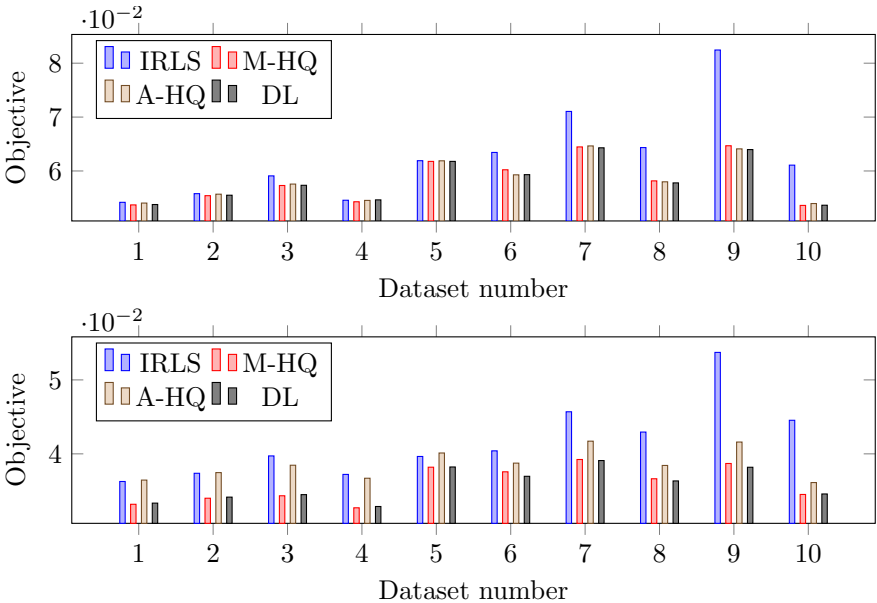


Figure 2: Results for linearized BA using the Welsch (top) and smooth truncated kernel (bottom).

We also verified if allowing more LM iterations (our choice is 500) leads to a reduction in the performance gaps between the methods for metric BA. This appears not to be the case, and therefore the different methods reach actually different basins of convergence.

## 6 Conclusion

In this work we investigated the benefits of the additive variant of half-quadratic lifting (A-HQ) for robust cost minimization and compared its performances against multiplicative half-quadratic lifting (M-HQ). We also proposed a “double lifting” approach (DL) that combines both additive and multiplicative lifting. On synthetic data, we obtained a clear ordering of the ability of the different methods to escape bad local minima: DL outperformed both A-HQ and M-HQ while M-HQ fell behind A-HQ. On real bundle adjustment datasets, we observed different behaviors: A-HQ fell behind both by M-HQ and DL, while DL was comparable to and sometimes slightly outperformed M-HQ. To conclude, we believe that our novel DL approach is an interesting alternative to M-HQ with a moderate 10% increase in run-time in exchange for a stronger ability to escape bad local minima.

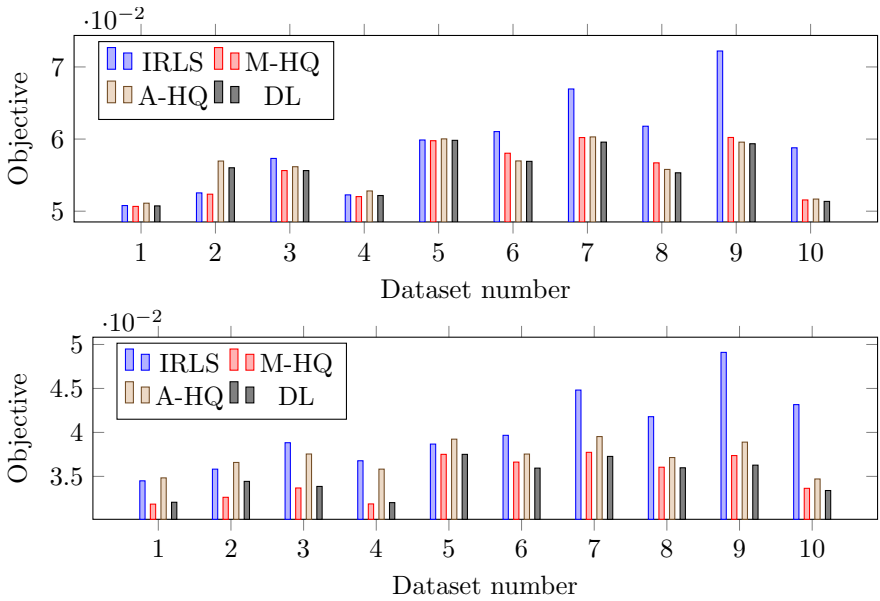


Figure 3: Results for metric BA using the Welsch (top) and smooth truncated kernel (bottom).

## References

- [1] Sameer Agarwal, Noah Snavely, Steven M Seitz, and Richard Szeliski. Bundle adjustment in the large. In *Proc. ECCV*, pages 29–42. Springer, 2010.
- [2] Michael J Black and Paul Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer vision and image understanding*, 63(1):75–104, 1996.
- [3] Andrew Blake and Andrew Zisserman. *Visual reconstruction*. 1987.
- [4] Daniel M Dunlavy and Dianne P O’Leary. Homotopy optimization methods for global optimization. Technical report, Sandia National Laboratories, 2005.
- [5] C. Engels, H. Stewénus, and David Nistér. Bundle adjustment rules. In *Photogrammetric Computer Vision (PCV)*, 2006.
- [6] Donald Geman and George Reynolds. Constrained restoration and the recovery of discontinuities. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(3):367–383, 1992.
- [7] Donald Geman and Chengda Yang. Nonlinear image recovery with half-quadratic regularization. *IEEE Transactions on Image Processing*, 4(7):932–946, 1995.
- [8] Peter J Green. Iteratively reweighted least squares for maximum likelihood estimation, and some robust and resistant alternatives. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 149–192, 1984.

- [9] Kenneth Lange, David R Hunter, and Ilsoon Yang. Optimization transfer using surrogate objective functions. *Journal of computational and graphical statistics*, 9(1):1–20, 2000.
- [10] Hossein Mobahi and John W Fisher. On the link between gaussian homotopy continuation and convex envelopes. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 43–56. Springer, 2015.
- [11] Hossein Mobahi and John W Fisher III. A theoretical analysis of optimization by gaussian continuation. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.
- [12] Kenneth Rose. Deterministic annealing for clustering, compression, classification, regression, and related optimization problems. *Proceedings of the IEEE*, 86(11):2210–2239, 1998.
- [13] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment – A modern synthesis. In *Vision Algorithms: Theory and Practice*, volume 1883 of *LNCS*, pages 298–372, 2000.
- [14] Christopher Zach. Robust bundle adjustment revisited. In *Proc. ECCV*, pages 772–787, 2014.
- [15] Christopher Zach and Guillaume Bourmaud. Iterated lifting for robust cost optimization. In *Proc. BMVC*, 2017.
- [16] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast global registration. In *Proc. ECCV*, pages 766–782. Springer, 2016.
- [17] Michael Zollhöfer, Matthias Nießner, Shahram Izadi, Christoph Rehmann, Christopher Zach, Matthew Fisher, Chenglei Wu, Andrew Fitzgibbon, Charles Loop, Christian Theobalt, et al. Real-time non-rigid reconstruction using an rgb-d camera. *ACM Transactions on Graphics (TOG)*, 33(4):156, 2014.