Outlier-Robust Spatial Perception: Hardness, General-Purpose Algorithms, and Guarantees
Vasileios Tzoumas, Pasquale Antonante, Luca Carlone
Laboratory for Information & Decision Systems, Massachusetts Institute of Technology

1. Motivation and Contributions

Motivation: Spatial perception is the backbone of many robotics applications, spanning a broad range of research problems, such as see Fig. 1:
- (a) point cloud alignment (3D registration);
- (b) relative pose estimation from camera images (two-view geometry);
- (c) localization and mapping (SLAM).

But in the presence of incorrect data associations, sensor malfunctions, or even adversarial attacks, that is, in general, outliers, spatial perception is jeopardized. Although techniques to handle outliers exist, they can fail in unpredictable manners, e.g., RANSAC, or have exponential runtime (e.g., branch-and-bound).

Contributions: We advance the state-of-the-art in outlier rejection by making three contributions.
- (1) Prove the extreme hardness of outlier rejection even in non-polynomial runtime.
- (2) Provide the first per-instance (post-run) sub-optimality guarantees for any algorithm that may be used for outlier rejection. As such, they can be used to assess the approximation performance of, e.g., RANSAC.
- (3) Propose a general-purpose, linear runtime algorithm to remove outliers, termed adaptive trimming (ADAPT). ADAPT is agnostic to an outlier-free threshold, and is self-correcting.

We demonstrate ADAPT on 3D registration, two-view geometry, and SLAM. The results show that ADAPT outperforms several state-of-the-art methods across applications while being a general-purpose method.

3. Outlier Rejection is Inapproximable

Definition (Quasi-polynomial algorithm): An algorithm is quasi-polynomial if it runs in $2^{(polynomial)} n$ time, where $n$ is the input size, and $c$ a constant.

- A polynomial algorithm is also quasi-polynomial.
- Yet, a quasi-polynomial algorithm is asymptotically faster than exponential-time.

Theorem (Inapproximability): Consider the linear MTS problem, per (2):

$$\min_{C \in \mathbb{R}^{m \times n}} \sum_{i=1}^{m} \min_j |a_{ij}x_j| \leq \epsilon.$$  (1)

Let $C$ be the number of measurements $|M|$;  $x$ the true number of outliers; $\epsilon$ the optimal value of the parameter under estimation.

For any $\delta \in (0,1)$, there exist polynomial $\lambda_1(m)$ and $\lambda_2(n) = \omega(n^{O(1)})$ and Instances of (3) where $\epsilon < \frac{\lambda_2(m)}{\lambda_1(m)}$ such that there exist no quasi-polynomial algorithm that can return a solution $C$ to (1) $\epsilon$-close to $\frac{\epsilon}{\lambda_1(m)}$. This holds true even the algorithm knows $x$, and that $\epsilon$ exists.

Theorem implies extreme hardness of MTS:
Even if we inform the algorithms with the true number of outliers $x$, it is impossible for even quasi-polynomial algorithms to find a good set of outliers in $\epsilon$ time. Surprisingly, this remains true even if we allow the algorithms to cheat by rejecting more measurements than $\epsilon$ (i.e., $\lambda_1(m)$).

4. Post-run Sub-Optimality Guarantees

Remark (Approximability beyond the worst-case): Although our inapproximability result suggests that no efficient algorithm will do well in the worst-case, we can still be happy if we can evaluate a posteriori if an algorithm computed a good solution for any other given problem instance.

Theorem (Post-run sub-optimality bound): Consider the MTS problem, with optimal solution denoted by $C^*$. For any candidate solution $C$, let:

- $r(C) = \max_{i \in \mathbb{R}} \sum_{j \in \mathbb{R}} |a_{ij}x_j|$ be the residual error given the outlier rejection $C$.
- $r(C^*) = \max_{i \in \mathbb{R}} \sum_{j \in \mathbb{R}} |a_{ij}x_j|$ be the optimal residual error when all measurements are rejected.
- $x^* = r(C^*)$, i.e., $x^*$ is the residual error for the optimal solution $C^*$.

Then:

$$r(C) - r(C^*) \leq |x| \epsilon$$  (2)

where:

$$|x| \epsilon$$  quantifies through (4) the distance between the residual $r(C)$ of the algorithm’s output and the residual $x^*$ of the optimal solution.

The bound is algorithm agnostic: It does not take assumption on the way $x$ is generated.

The bound is computable in $O(n)$ time.

-bound is shown tight via simulations (see paper).

5. A General-Purpose Algorithm: ADAPT

Algorithm 1: Adaptive Trimming (ADAPT)

- (10) has error of at least 25 meters. For the complete figure, see [11].

ADAPT is adaptive, self-correcting, with linear runtime. At each iteration:
- (Adaptive) For each candidate outlier, (Self-correcting) Revises candidate outliers: Inliers incorrectly picked as outliers can be reconsidered as inliers later on.

ADAPT outperforms specialized outlier rejection methods (in particular for SLAM) while being a general-purpose algorithm.

Experiments

Our experimental tests demonstrate: ADAPT outperforms RANSAC in terms of accuracy and scalability; and outperforms specialized outlier rejection methods (in particular for SLAM) while being a general-purpose algorithm. Finally, the tests show that the sub-optimality bound in (4) is informative and can be used to assess the outlier rejection outcomes.

3D Registration


Compared methods: We benchmark ADAPT against Fast Global Registration (FGR) [3], and the three-point RANSAC.

SLAM

Experimental setup: We test ADAPT on standard 2D and 3D SLAM datasets: HIT (2D), Intel (2D), CSAIL (2D), and Sphero250 (3D) [4,5]. ADAPT uses SE-Sync [5] as global solver.

Compared methods: We benchmark ADAPT against DC5 [6]; we report DC5 results for three choices of the robust kernel size: {1,10,100} (default is 1).

Two-view Geometry

Experimental setup: We test ADAPT on both synthetic data and on the WS/GT sequence of the EuRoC dataset [7]. ADAPT uses BriBrie’s QCP relaxation [8] as global solver.

Compared methods: We benchmark ADAPT against Huster’s five-point [9] and the eight-points algorithm [10] within RANSAC.

References