Pose Graph

David Arnon

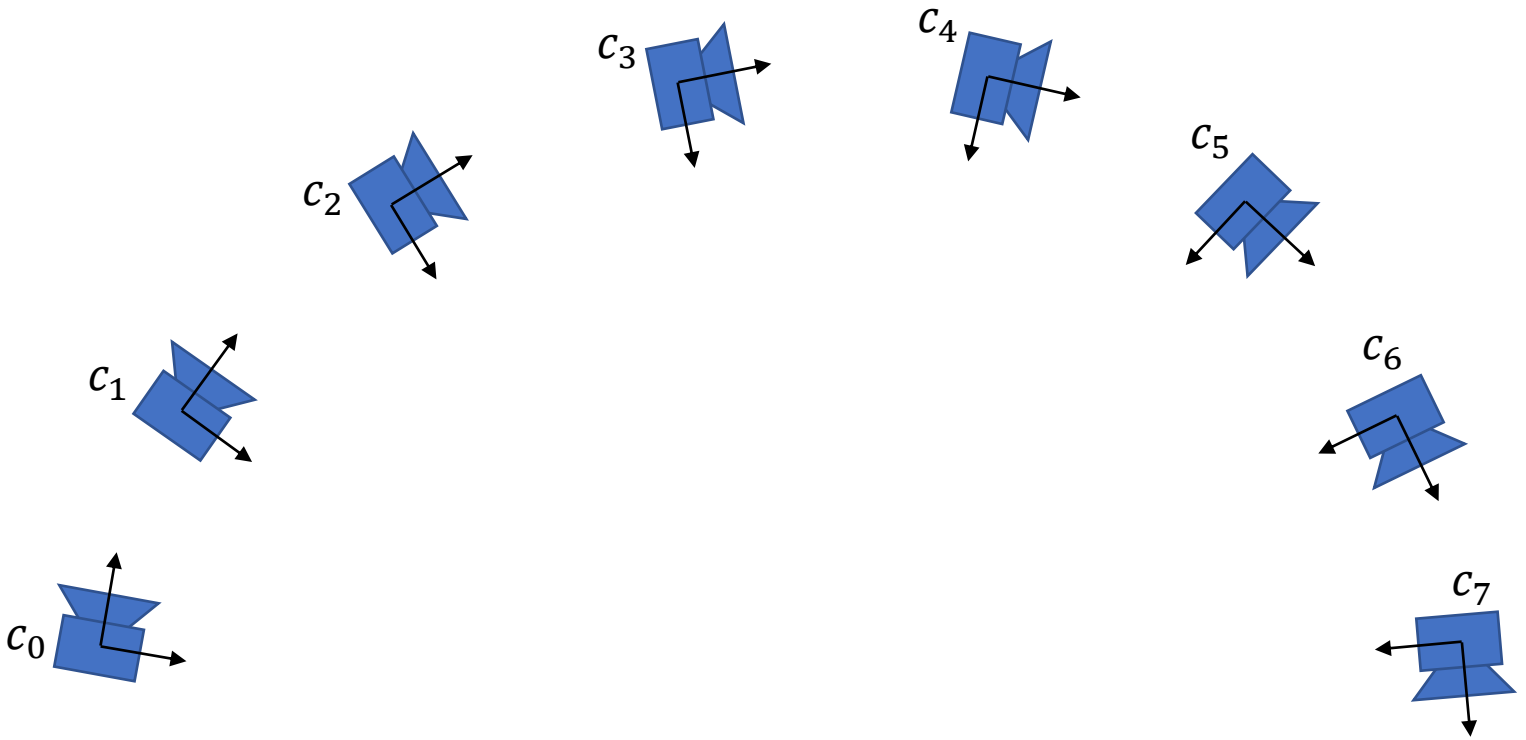
GTSAM
StereoCamera

- `StereoCamera(Pose3 leftCamPose, Cal3_S2Stereo K)`
- `StereoPoint2 = StereoCamera.project(Point3)`
- `Point3 = StereoCamera.backproject(StereoPoint2)`
- Location Likelihood:
  `gtsam.PriorFactorPose3(pose_key, base_pose, uncertainty)`

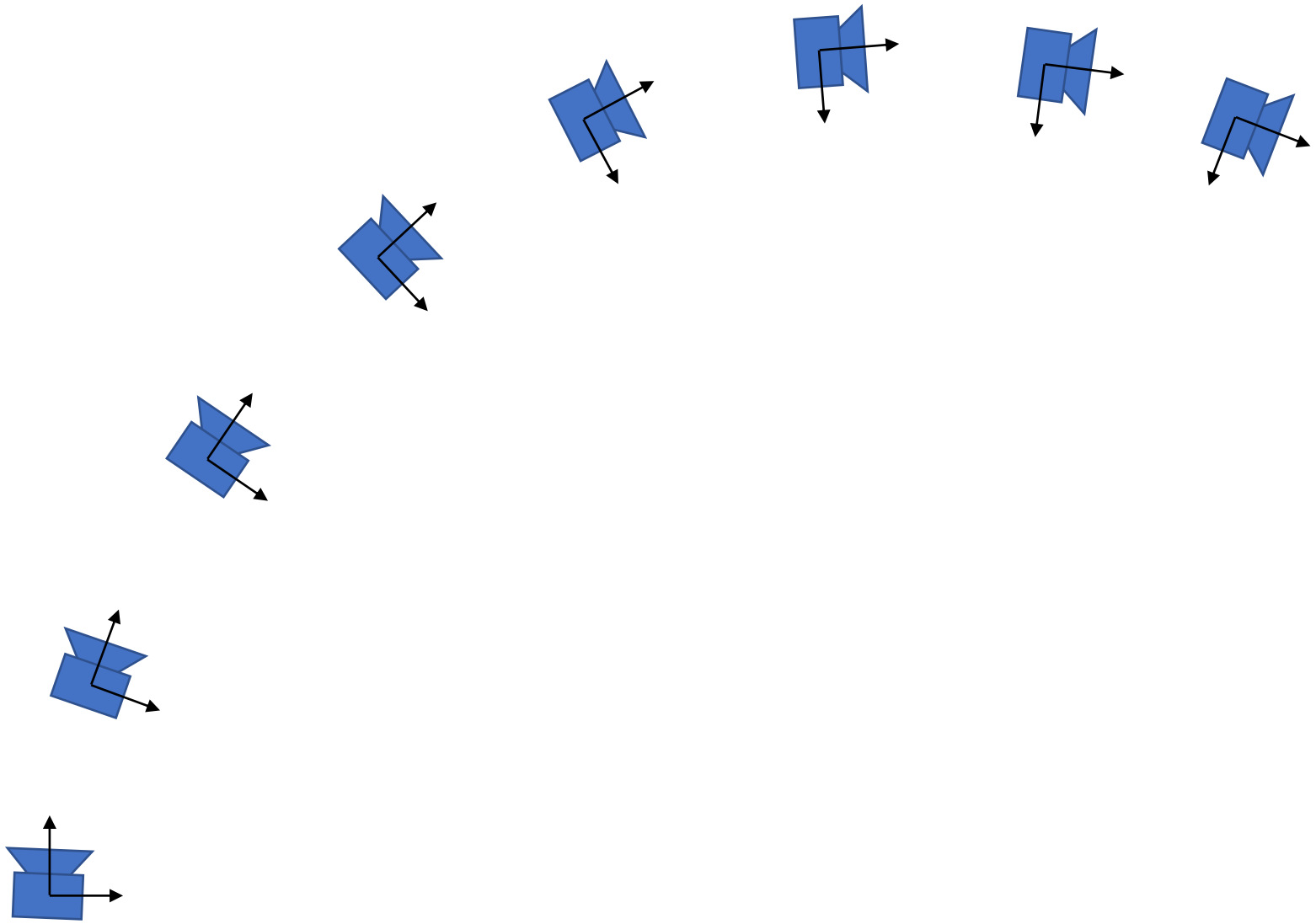
Relative Nonlinear Factor

- `relative_pose = pose_c0.between(pose_c1)`
- $c_0 = [R_0|t_0], \quad c_1 = [R_1|t_1]$
- $R_0 \underbrace{(R_\Delta x + t_\Delta)}_{c_1 \to c_0} + t_0 = R_1 x + t_1$
- $R_\Delta = R_0^T R_1$
- $t_\Delta = R_0^T (t_1 - t_0)$
- `BetweenFactorPose3(c0, c1, relative_pose, noiseCov)`

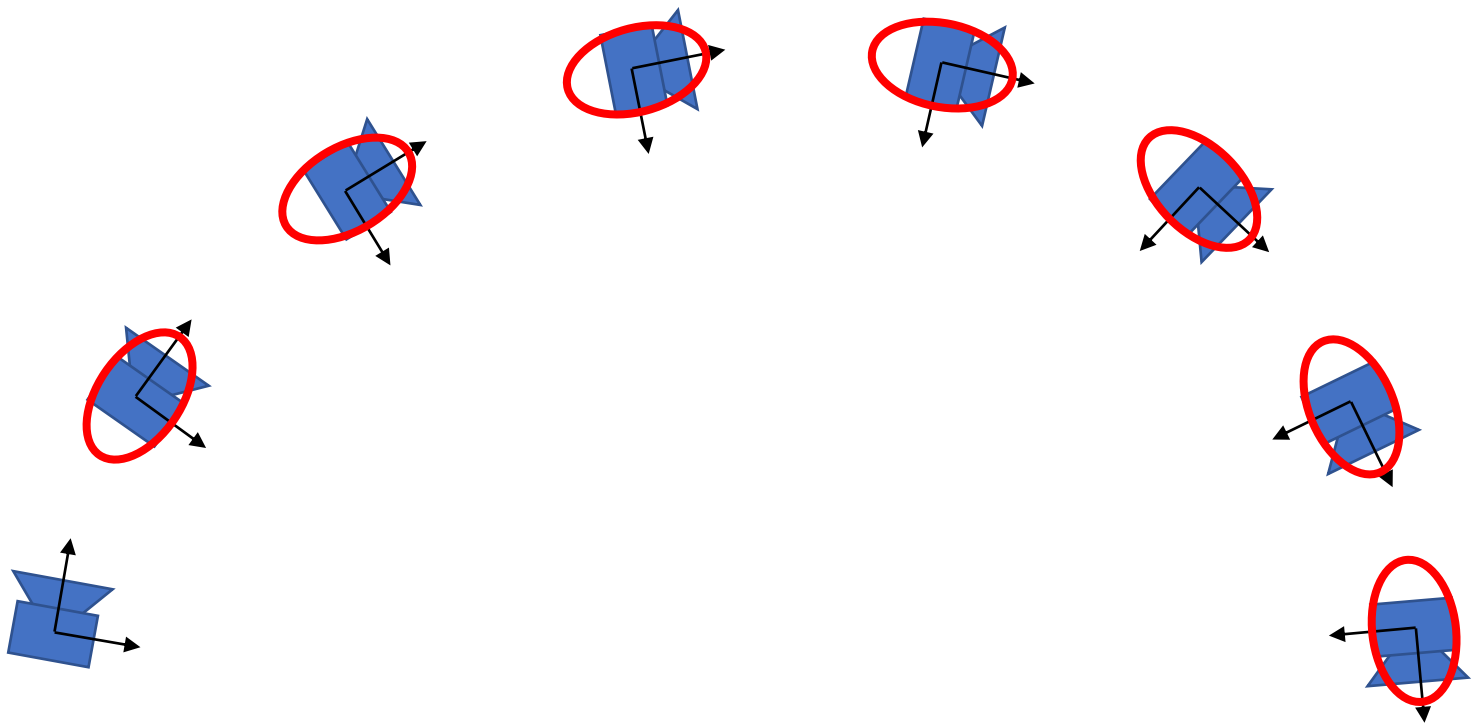
Estimation Error



Estimation Error



Absolute Error

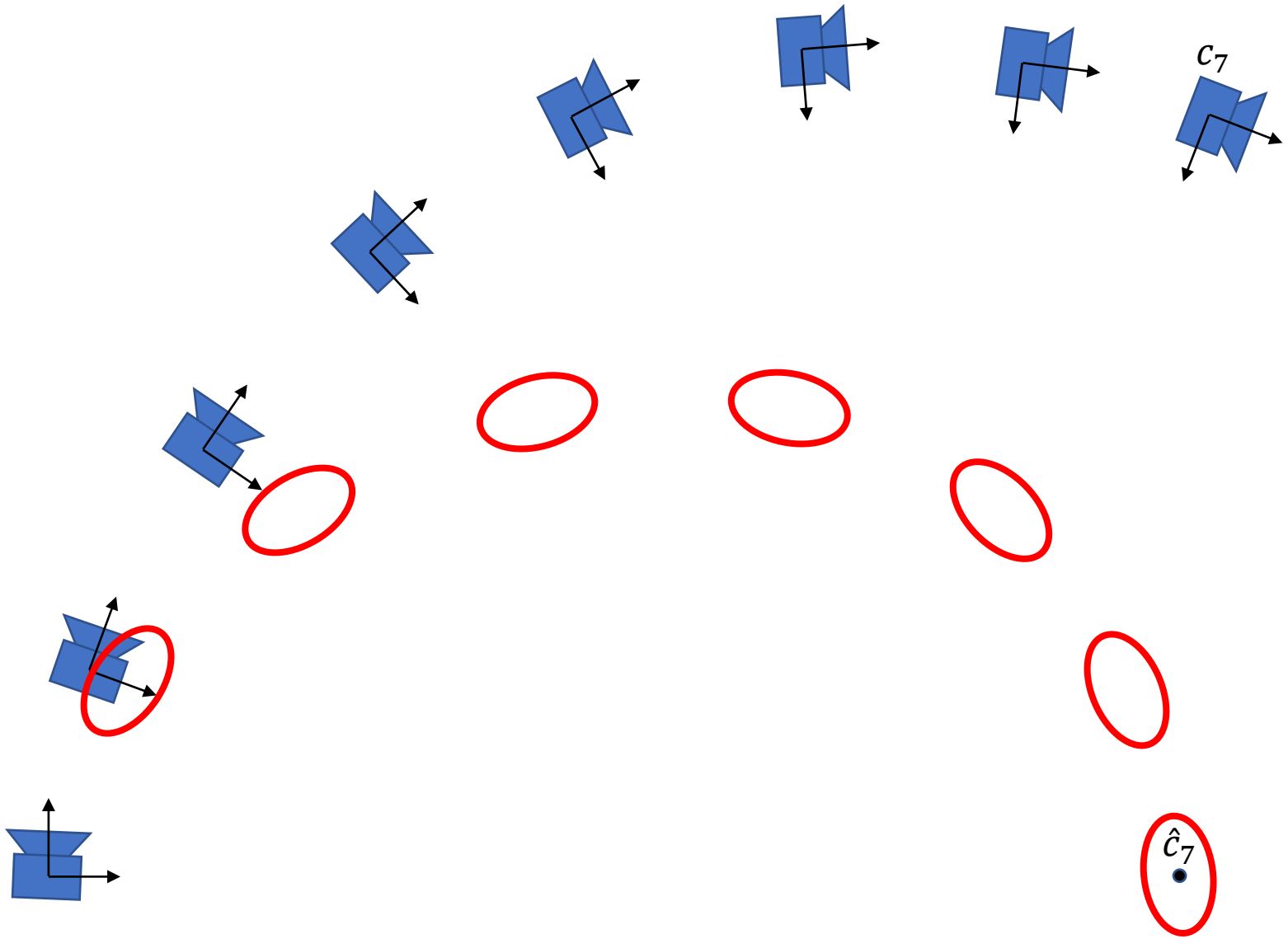


$$c_1 \stackrel{!}{=} \hat{c}_1$$

$$\vdots$$

$$c_7 \stackrel{!}{=} \hat{c}_7$$

Absolute Error

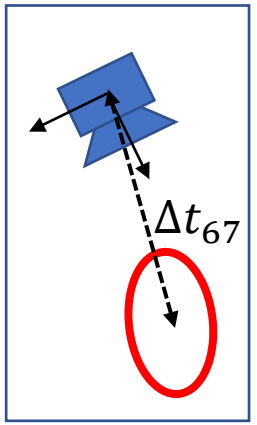


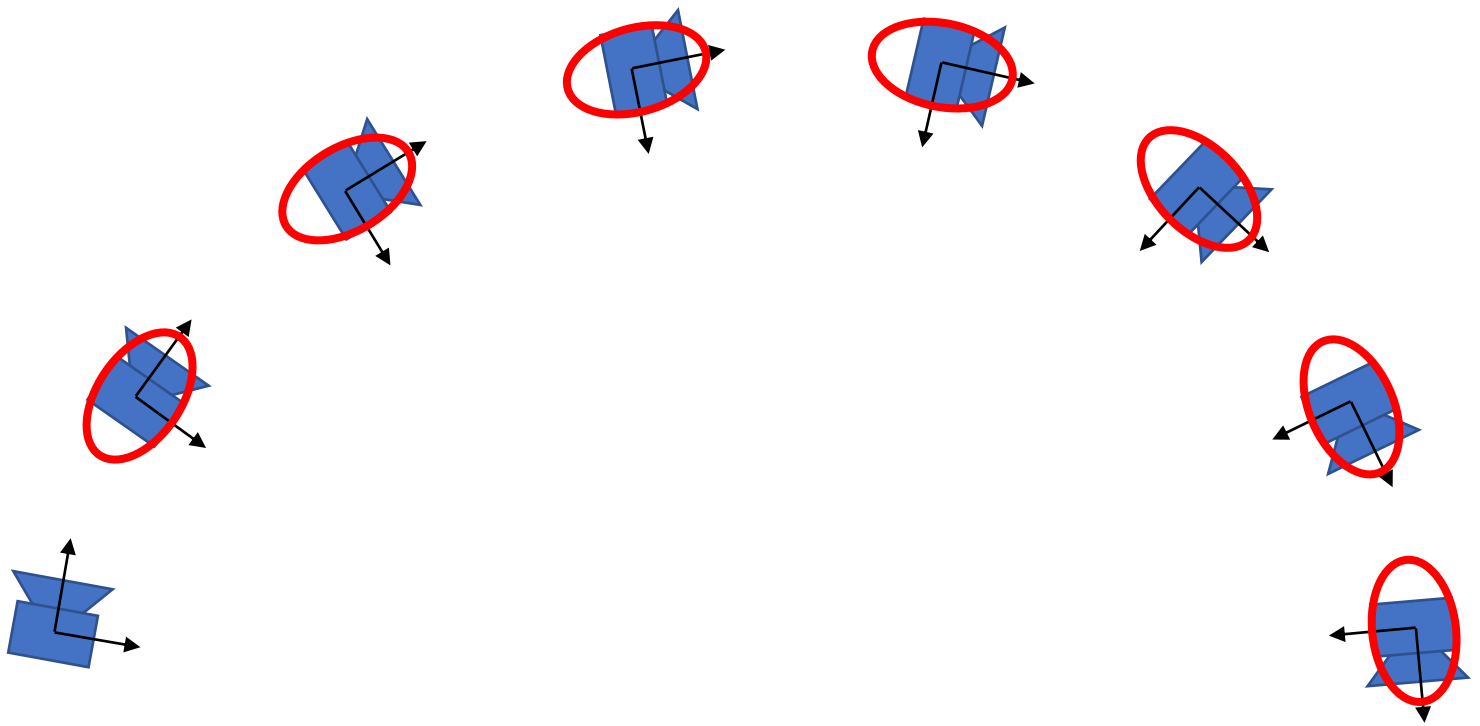
$$c_1 \stackrel{!}{=} \hat{c}_1$$

$$\vdots$$

$$c_7 \stackrel{!}{=} \hat{c}_7$$

Relative Error



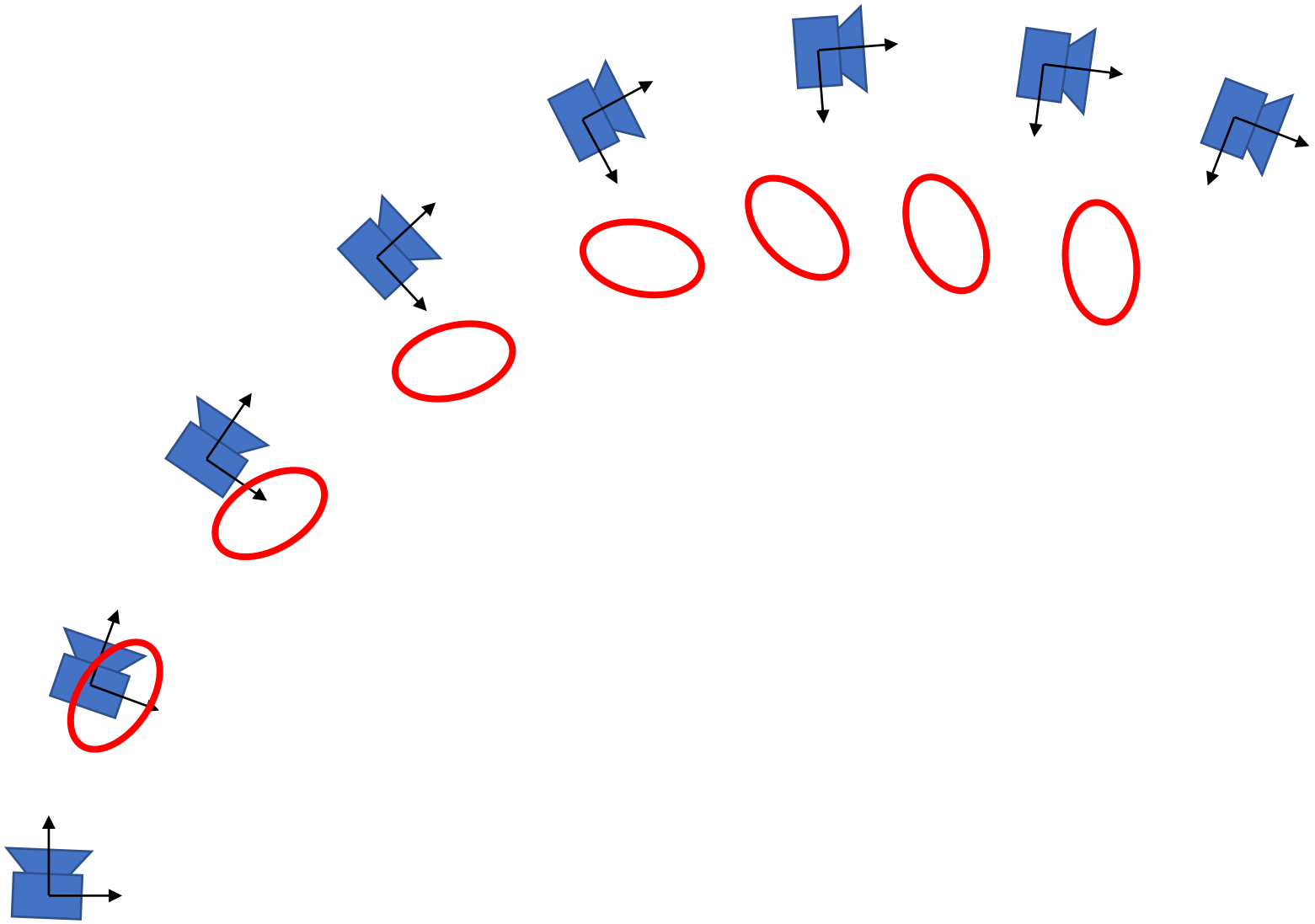


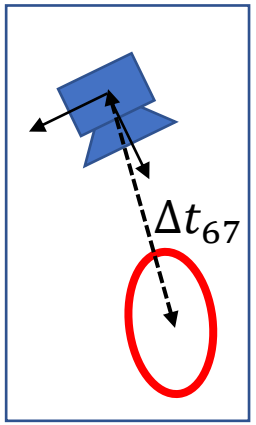
$$t_1 - t_0 \stackrel{!}{=} \Delta t_{01}$$

$$\vdots$$

$$t_7 - t_6 \stackrel{!}{=} \Delta t_{67}$$

Relative Error





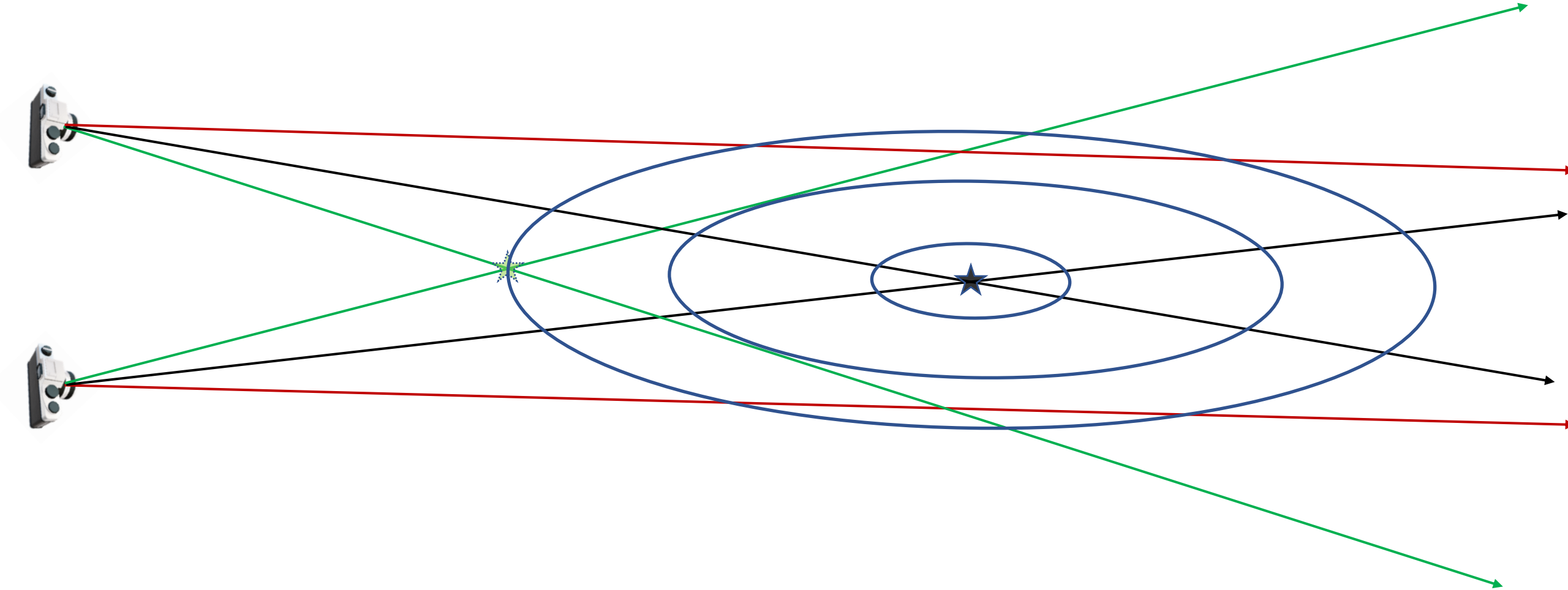
$$t_1 - t_0 \stackrel{!}{=} \Delta t_{01}$$

$$\vdots$$

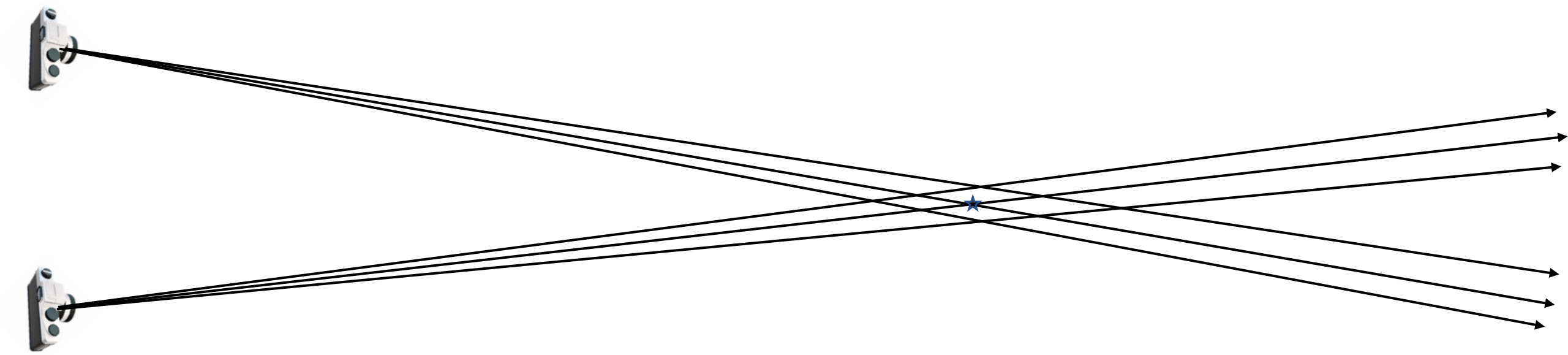
$$t_7 - t_6 \stackrel{!}{=} \Delta t_{67}$$

Gaussian

Linear Approximation



Gaussian
Linear Approximation

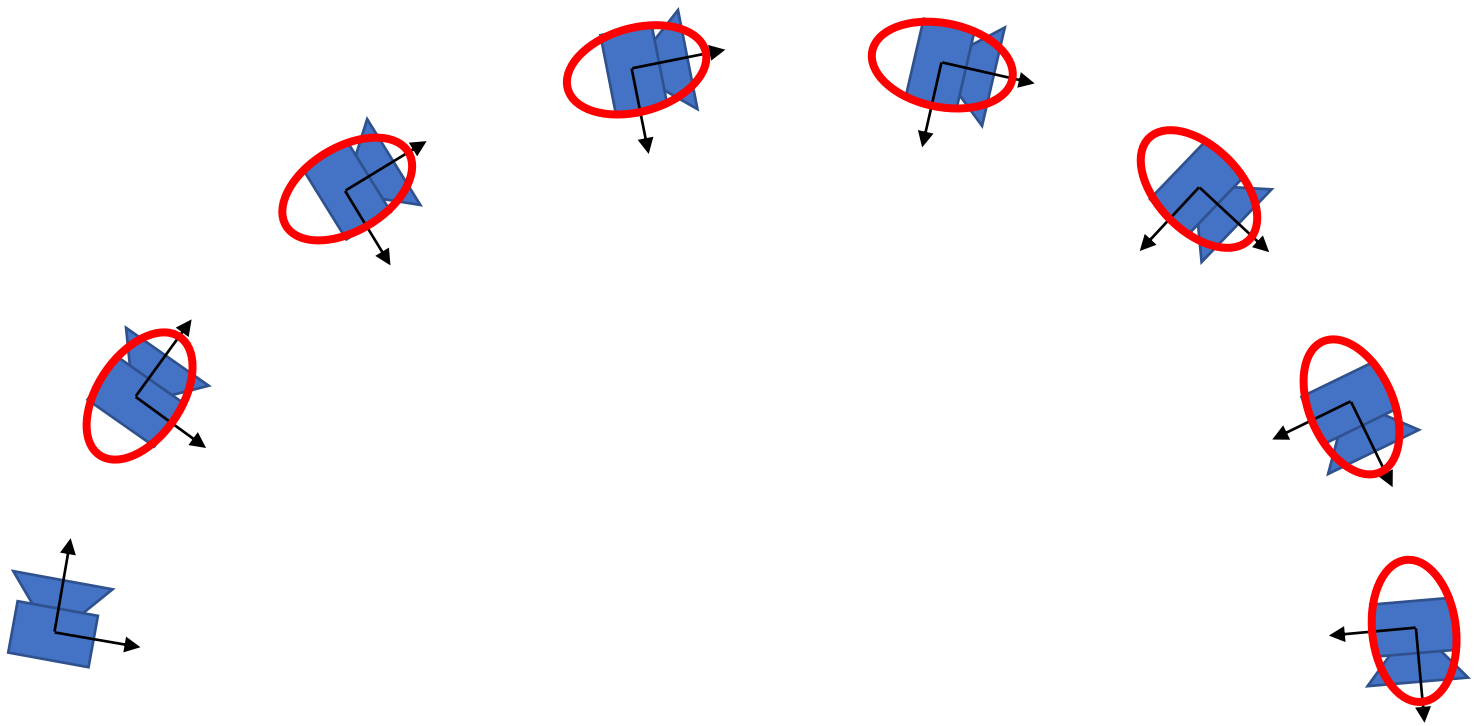


Relative POV Error

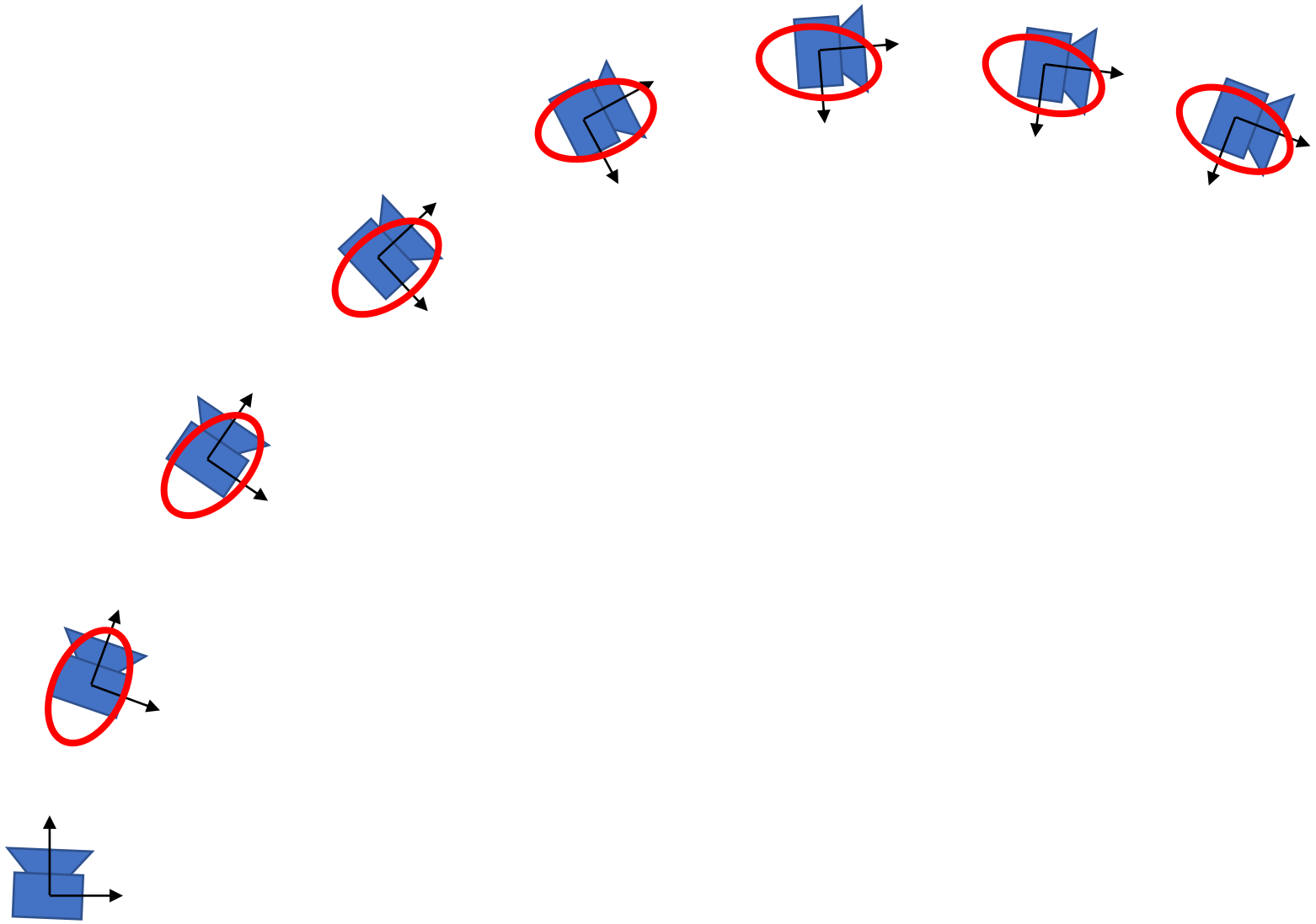
$$\Delta t_{01} \stackrel{!}{=} R_0^T(t_1 - t_0)$$

$$\vdots$$

$$\Delta t_{67} \stackrel{!}{=} R_6^T(t_7 - t_6)$$



Relative Nonlinear Error



$$\Delta t_{01} \stackrel{!}{=} R_0^T(t_1 - t_0)$$

$$\vdots$$

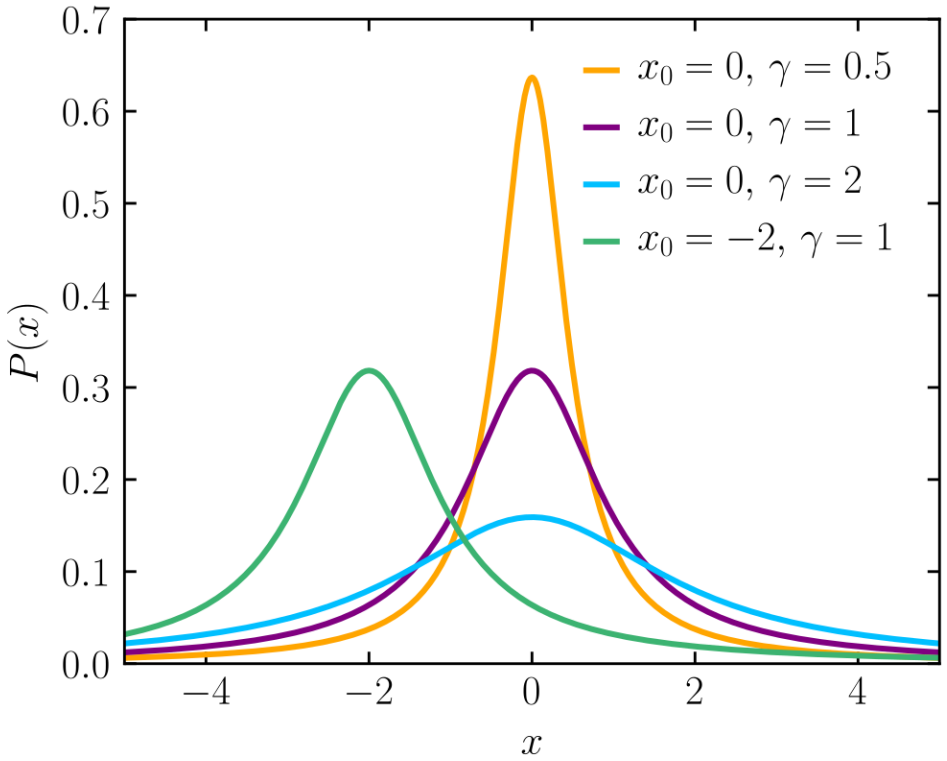
$$\Delta t_{67} \stackrel{!}{=} R_6^T(t_7 - t_6)$$

Robust Kernels

Robust Estimation

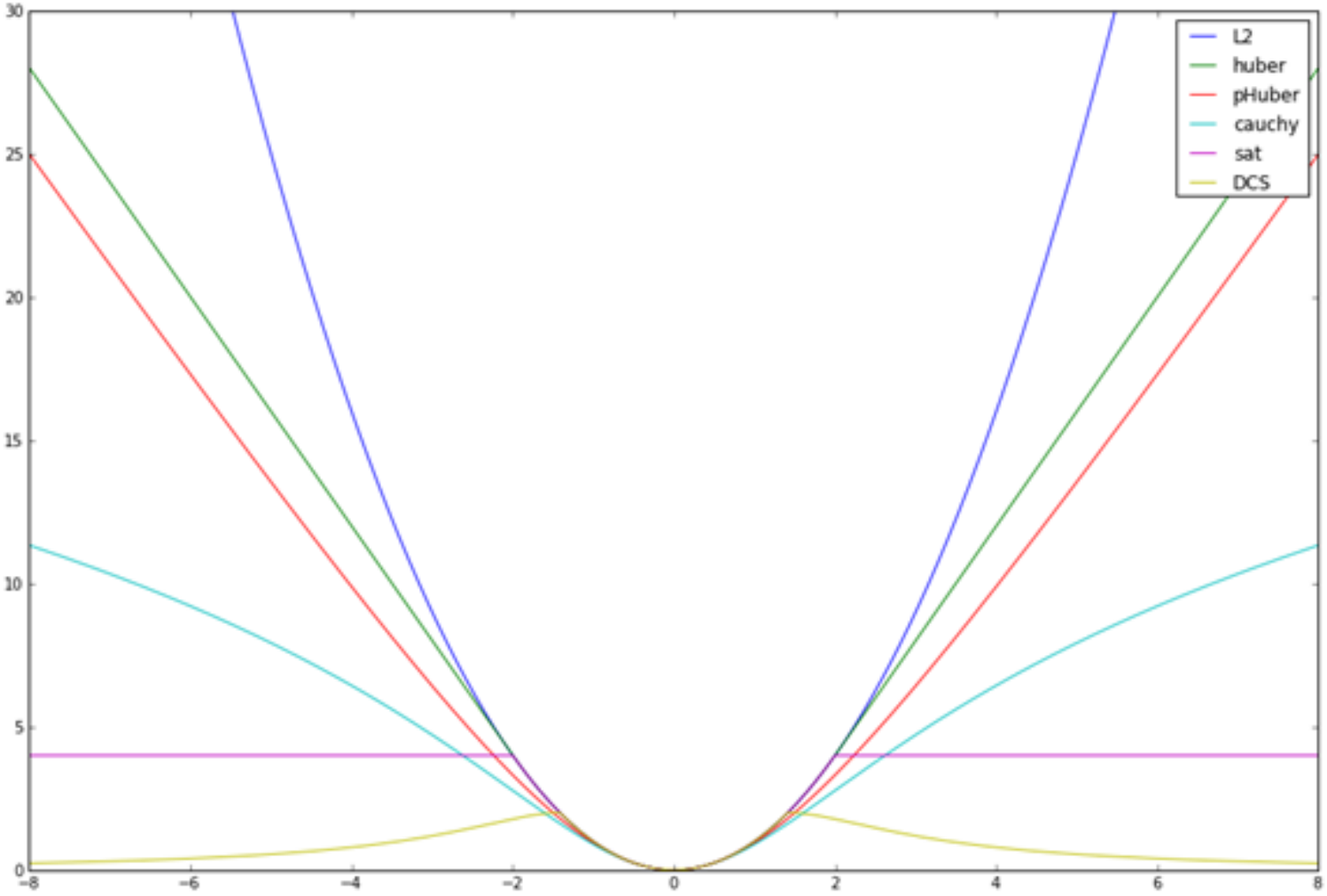
Kernels

- L_2: $L_\delta(x) = x^2$
- Huber: $L_\delta(x) = \begin{cases} x^2, & |x| < \delta \\ \delta(2|x| - \delta), & |x| \geq \delta \end{cases}$
- Saturated: $L_\delta(x) = \begin{cases} x^2, & |x| < \delta \\ \delta^2, & |x| \geq \delta \end{cases}$
- Cauchy: $L_\delta(x) = \delta^2 \log(1 + (x/\delta)^2)$
- Cauchy distribution: $Cauchy_{0,\gamma}(x) = \dfrac{1}{\pi\gamma\left(1 + \left(\dfrac{x}{\gamma}\right)^2\right)}$



Robust Estimation

Kernels



Robust Estimation
GTSAM Kernels

- **Look Ma, No RANSAC**
  https://gtsam.org/2019/09/20/robust-noise-model.html

- `gaussian_model = gtsam.noiseModel.Diagonal.Sigmas(np.array([1.0, 1.0, 1.0]))`
- `cauchy = gtsam.noiseModel.mEstimator.Cauchy.Create(2)`
- `cauchy_model = gtsam.noiseModel.Robust.Create(cauchy, gaussian_model)`