Apply Tracklet Association to Robotic Data

Group 3: David E. Bernal, Aldo Pareja, and Chenghui Zhou

Carnegie Mellon University

Motivation

Applications

- **Labeling multiple moving objects**
  - Only positional data is known
  - Visual detection of objects with similar appearance
  - Laser sensors do not label the detections
- **Prediction of future dynamics**
  - Infer which observations correspond to what objects
  - Predict future positions of the object
- **Many application areas**
  - Law enforcement, military, robotic sports, and service robot fields

Goal

Track multiple objects from non-labeled positional information using convex optimization tools

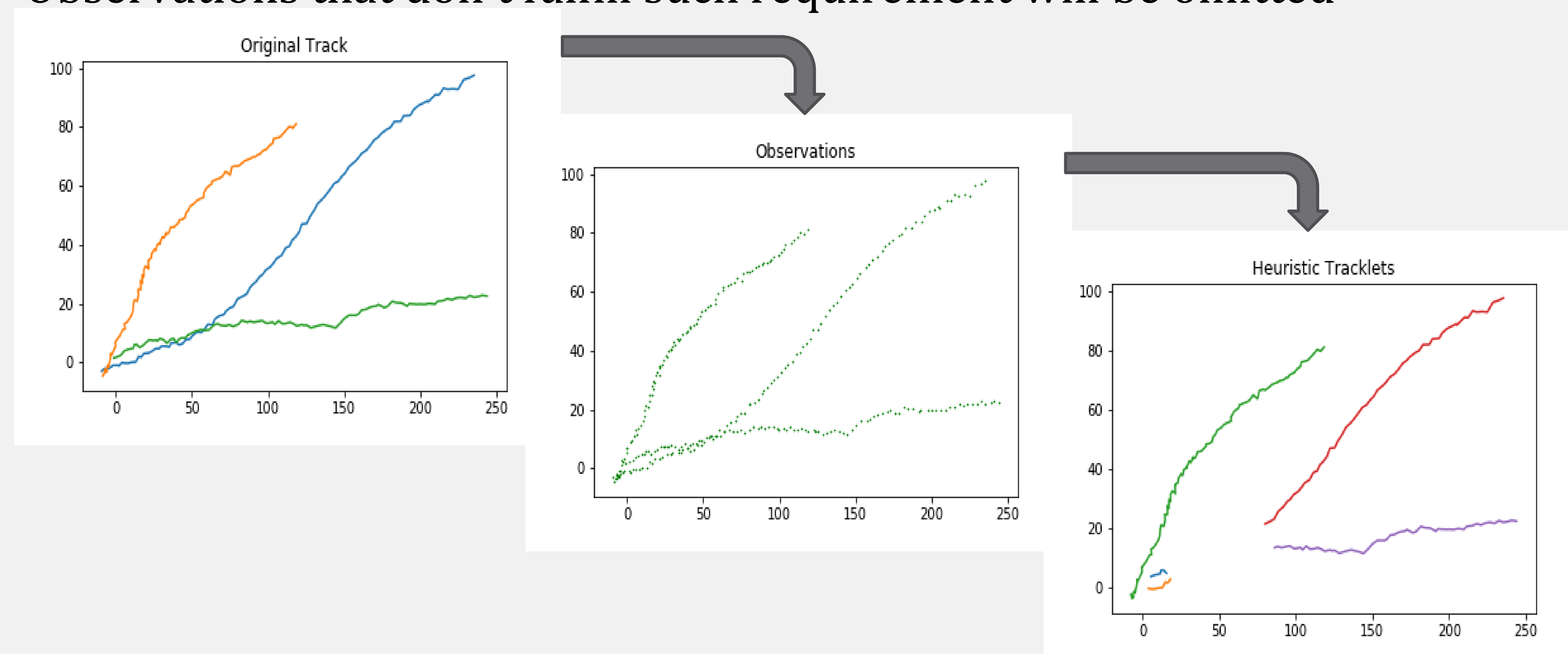
Data and Initial Tracklet Generation

Short high confidence tracklets:

Joining observations which are close and far away from other observations

$$\left\| y_t - y_{t+1}^1 \right\|_1 < 0.3 \left\| y_t - y_{t+1}^2 \right\|_1$$

Observations that don't fulfill such requirement will be omitted



Maximum Similarity Assignment

- After computing the similarity matrix P we join tracklets into recovered trajectories by solving a **Generalized Linear Assignment** problem which is mixed integer

$$\max_X \sum_{i,j} P_{i,j} X_{i,j}$$

$$s.t. \sum_i X_{i,j} \leq 1, \forall j; \sum_j X_{i,j} \leq 1, \forall i; X_{i,j} \in \{0,1\}, \forall i,j$$

- We used the Branch and Bound algorithm together with the Simplex Method implemented in the solver Gurobi 7.5

GUROBI OPTIMIZATION

Project pipeline

Simulate moving objects

Using Linear dynamic system
- Variables amount of objects
- Noise level in the state vector
- Random initial points
- Necessary intersection

Generate initial tracklets

Group sequence of observations or **tracklets**
- Short and high confidence
- Heuristic

Estimate similarity

Minimize **Rank of Hankel Matrix**
- Determines complexity of tracklet
- Tracklets with similar dynamics should have high similarity
- Exclude time conflicts
- Solve convex relaxation of problem as an Semi-definite program (SDP)

Recover trajectories

Solve an Assignment problem
- Formulate as Integer Program
- Say which tracklets should be stitched based on similarities
- Get recovered data from SDP

Linear dynamic Systems

Simulate moving object

$$y_t = Ox_t + \varepsilon_t$$
$$x + (t+1) = Tx_t + \delta_t$$

- y_t is the position at time t
- x_t contains position and velocity at time t
- ε_t, δ_t are normal distributed noises with level given by covariances $\text{diag}(\Sigma), \text{diag}(\Delta)$
- T, O are transition and observation matrices

Similarity Measure Between Tracklets

Dynamics Complexity

- The complexity of a sequence of observations or **tracklet** $\alpha = \{y_i\}_{i=1}^T$ is the order of a linear regression:

$$y_t = \Sigma_{i=1}^T a_i y_{t-i}$$

- This order with the **Rank of the Hankel Matrix** for the sequence:

$$H_\alpha = \begin{bmatrix} y_1 & \cdots & y_m \\ y_2 & \cdots & y_{m+1} \\ \vdots & \cdots & \vdots \\ y_{T-m+1} & \cdots & y_T \end{bmatrix}$$

Similarity Measure

- If two tracklets α_i, α_j have **similar dynamics** then they are likely to be parts of a bigger tracklet.
- **Similarity matrix**:

$$P_{ij} \doteq \begin{cases} -\infty & if\ \alpha_i\ and\ \alpha_j\ conflict\ in\ time \\ & \dfrac{NSV_\sigma(H_{\alpha_i}) + NSV_\sigma(H_{\alpha_j})}{\min_{\beta_{i,j}} NSV_\sigma(H_{\alpha_j,\alpha_i})} \end{cases}$$

- Instead of Rank (NP-Hard) we use a thresholded nuclear norm (NSV_σ) as a convex surrogate.
- H_{α_j,α_i} is given by a **joint tracklet** $\alpha_{ij} = [\alpha_i\ \ \beta_{i,j}\ \ \alpha_j]$

Missing Data Recovery

- Minimizing a Rank is an NP-Hard problem so we solve a convex relaxation to recover $\beta_{i,j}$, becoming an Semi-Definite Program (SDP):

$$\min_{\beta_{i,j},X,Z} tr(X) + tr(Z)$$

$$s.t \begin{bmatrix} X & H_{\alpha_j,\alpha_i} \\ H_{\alpha_j,\alpha_i}^T & Z \end{bmatrix} \succcurlyeq 0$$

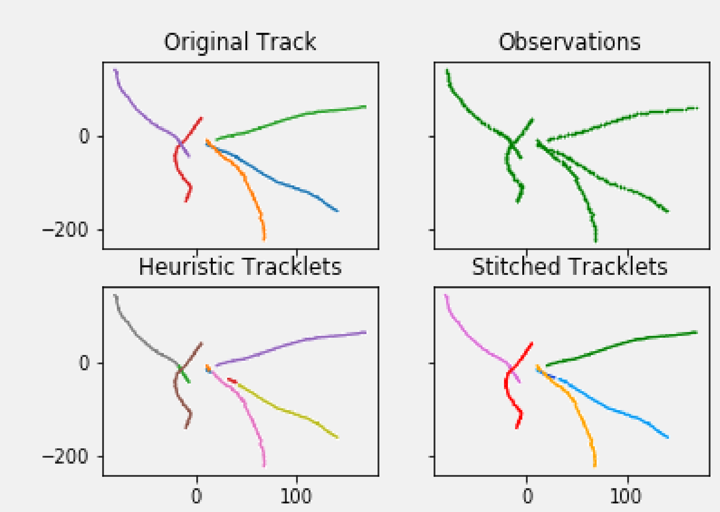
$$X = X^T \succcurlyeq 0, Z = Z^T \succcurlyeq 0$$

- Solved using the solver Mosek 8.0 together with CVX using a conic interior point algorithm

mosek CVX RESEARCH

Illustrative Example and Results

5 tracklets recovery



Example for 5 trajectories

- Split for 9 short tracklets
- Recovered original trajectories and missing dynamics

Accuracy Calculation

$$Accuracy = \frac{\#\ correctly\ recovered\ tracklets}{\#\ tracklets}$$

Results

Objects	diag(Σ), diag(Δ)	Accuracy
5	0.01, 0.05	98.55%
3	0.01, 0.05	98.00%
5	0.1, 0.5 (noisy)	85%
3	0.1, 0.5 (noisy)	84.77%

Conclusions and Future Steps

- We were able to recover original trajectories from raw observations using Convex Optimization tools with high accuracy and moderate computational cost.
- Next step is to try it with real robotic sensors outputs

Carnegie Mellon University – Final Presentation Project Convex Optimization 10-725 – December 14 2017