Optimization Course Project VI:

Optimization in Sensor Network Localization

1 Introduction

Sensor Network Localization (SNL) is a major topic in Data Science and Machine Learning. It is also closely related to Data Dimensionality Reduction, Phase Retrieval, Molecular Confirmation, and Graph Realization. The SNL problem is: Given possible anchors $\mathbf{a}_k \in R^d$, distance information $d_{ij},\ (i,j) \in N_x$, and $\hat{d}_{kj},\ (k,j) \in N_a$, find $\mathbf{x}_i \in R^d$ for all i such that

$$\begin{aligned} \|\mathbf{x}_i - \mathbf{x}_j\|^2 &= d_{ij}^2,\ \forall\ (i,j) \in N_x,\ i<j, \\ \|\mathbf{a}_k - \mathbf{x}_j\|^2 &= \hat{d}_{kj}^2,\ \forall\ (k,j) \in N_a, \end{aligned} \tag{1}$$

where $(i,j) \in N_x$ ($(k,j) \in N_a$) connects points $\mathbf{x}_i$ and $\mathbf{x}_j$ ($\mathbf{a}_k$ and $\mathbf{x}_j$) with an edge whose Euclidean length is d_{ij} ($\hat{d}_{kj}$). N_x and N_a denote the pairs of points whose distances are known. This system of quadratic equations are difficult to solve in general.

1.1 Global Optimization Model

There is a simple nonlinear least squares approach to solve (1):

$$\min_{\mathbf{x}_i} \quad \sum_{(i,j)\in N_x} \left(\|\mathbf{x}_i - \mathbf{x}_j\|^2 - d_{ij}^2\right)^2 + \sum_{(k,j)\in N_a} \left(\|\mathbf{a}_k - \mathbf{x}_j\|^2 - d_{kj}^2\right)^2 \tag{2}$$

which is an unconstrained nonlinear and non-convex minimization problem with potentially many local minimum solutions.

1.2 SOCP Relaxation Model

One can develop an SOCP relaxation for solving (1): Find vectors $\mathbf{x}_i$ to solve the feasibility problem:

$$\begin{array}{ll} \min_{\mathbf{x}_i} & \sum_i \mathbf{0}^T \mathbf{x}_i \\ \text{s.t.} & \|\mathbf{x}_i - \mathbf{x}_j\|^2 \le d_{ij}^2,\ \forall\ (i,j) \in N_x,\ i < j, \\ & \|\mathbf{a}_k - \mathbf{x}_j\|^2 \le \hat{d}_{kj}^2,\ \forall\ (k,j) \in N_a. \end{array} \tag{3}$$

This becomes a convex optimization problem, in particular, a second-order cone linear optimization problem described in the class.

1.3 SDP Relaxation Model

One also establish an SDP relaxation for solving (1): Find a symmetric matrix $Z \in S^{d+n}$ such that

$$\begin{array}{ll} \min_Z & \mathbf{0} \bullet Z \\ \text{s.t.} & Z_{1:d,1:d} = I, \\ & (\mathbf{0}; \mathbf{e}_i - \mathbf{e}_j)(\mathbf{0}; \mathbf{e}_i - \mathbf{e}_j)^T \bullet Z = d_{ij}^2,\ \forall\ i,j \in N_x,\ i < j, \\ & (\mathbf{a}_k; -\mathbf{e}_j)(\mathbf{a}_k; -\mathbf{e}_j)^T \bullet Z = \hat{d}_{kj}^2,\ \forall\ k,j \in N_a, \\ & Z \succeq \mathbf{0}. \end{array} \tag{4}$$

Note that $Z_{1:d,1:d} = I \in S^d$ can be realized through $d(d+1)/2$ linear equations. For example, if $d = 2$, we have $Z_{11} = 1$, $Z_{22} = 1$, and $Z_{12} = 0$.

The SDP relaxation model with possible noisy distance data can be formulated to minimize the L_1 norm of the total distance errors:

$$\begin{array}{ll} \min_{Z,\delta',\delta'',\hat{\delta}',\hat{\delta}''} & \sum_{(i,j)\in N_x} (\delta'_{ij} + \delta''_{ij}) + \sum_{(k,j)\in N_a} (\hat{\delta}'_{kj} + \hat{\delta}''_{kj}) \\ \text{s.t.} & Z_{1:d,1:d} = I, \\ & (\mathbf{0}; \mathbf{e}_i - \mathbf{e}_j)(\mathbf{0}; \mathbf{e}_i - \mathbf{e}_j)^T \bullet Z + \delta'_{ij} - \delta''_{ij} = d_{ij}^2,\ \forall\ i,j \in N_x,\ i < j, \\ & (\mathbf{a}_k; -\mathbf{e}_j)(\mathbf{a}_k; -\mathbf{e}_j)^T \bullet Z + \hat{\delta}'_{kj} - \hat{\delta}''_{kj} = \hat{d}_{kj}^2,\ \forall\ k,j \in N_a, \\ & Z \succeq \mathbf{0} \\ & \delta', \delta'', \hat{\delta}', \hat{\delta}'' \ge 0. \end{array} \tag{5}$$

In this case, the SDP solution from the relaxation

$$\bar{Z} = \begin{pmatrix} I & \bar{X} \\ \bar{X}^T & \bar{Y} \end{pmatrix}$$

often may not be rank d so that $\bar{X} \in R^{d\times n}$ cannot be the best possible localization of the n sensors.

This project is to develop efficient computational algorithms to solve the problem. You may randomly generated problems in 2D with 3 or more anchors, respectively, and many sensors to test your approaches.

You may set up a threshold radius such that the distance, with some possible noise, between any two points can be measured and it is known when the distance is below the threshold.

2 Possible Approaches

Below are suggested approaches for solving the problem

2.1 Unconstrained Nonlinear Optimization Search Approaches

Directly solve the global optimization model (2) using:

- First-order methods: gradient, conjugate-gradient, stochastic gradient, etc.
- Second-order methods: BFGS, trust-region, etc.
- Dimension-Reduced Second-Order Method (DRSOM) described in the class or [5].

2.2 Convex-Relaxation-First and Gradient-Second Approaches

Use the SOCP or SDP relaxation solution $\bar{X} = [\bar{\mathbf{x}}_1,\ \bar{\mathbf{x}}_2,\ ....,\ \bar{\mathbf{x}}_n]$ of (5) as the initial solution for solving model (2) by the Methods listed earlier. Are you able to estimate the position of the sensors well? Compare this approach to using Steepest Descent on (2) with random initialization.

2.3 Steepest Descent and Projection Method for SDP Relaxations

Unfortunately, the current available SDP solvers are still too time consuming for solving large-scale SDP problems. In this part, you are asked to implement one of the first-order SDP methods described in class to solve the SDP relaxation problem for SNL.

The SNL problem can be casted as

$$\min\ f(X) = \frac{1}{2}\|\mathcal{A}X - \mathbf{b}\|^2 \text{ s.t. } X \succeq \mathbf{0},$$

where

$$\mathcal{A}X = \begin{pmatrix} A_1 \bullet X \\ ... \\ A_m \bullet X \end{pmatrix},\ \mathcal{A}^T\mathbf{y} = \sum_{i=1} y_i A_i, \quad \text{and} \quad \nabla f(X) = \mathcal{A}^T(\mathcal{A}X - \mathbf{b}).$$

The SDM projection method described in class is to compute

$$\hat{X}^{k+1} = X^k - \frac{1}{\beta}\nabla f(X^k),$$

then project $\hat{X}^{k+1}$ back to the cone. One way for the projection is to use the eigendecomposition $\hat{X}^{k+1} = V\Lambda V^T$, where V are the eigenvectors and Λ the eigenvalues, and let

$$X^{k+1} = \text{Proj}_K(\hat{X}^{k+1}) = V\max\{\mathbf{0},\ \Lambda\}V^T.$$

The drawback is that the eigendecomposition may be costly in each iteration.

- Try just computing the few largest eigenpairs, say six largest λ_i with corresponding eigenvectors $\mathbf{v}_i$ and let:

  $$X^{k+1} = \sum_{i=1}^{6} \max\{0, \lambda_i\}\mathbf{v}_i\mathbf{v}_i^T.$$

  Typically, a few extreme eigenvalues of a symmetric matrix can be computed more efficiently. Here, we assume that the problem has only one anchor at the origin. One can find the true position later using two more anchor information.

- Any possible theoretical analysis of the projection algorithm?

2.4 ADMM Method for Sensor Network Localization

Another speed-up may be using ADMM approach. One can reformulate the nonlinear least squares model (2) as

$$\begin{array}{ll} \min & \sum_{(i,j)\in N_x} \left[(\mathbf{x}_i - \mathbf{x}_j)^T(\mathbf{y}_i - \mathbf{y}_j) - d_{ij}^2\right]^2 + \sum_{(k,j)\in N_a} \left[(\mathbf{a}_k - \mathbf{x}_j)^T(\mathbf{a}_k - \mathbf{y}_j) - d_{kj}^2\right]^2 \\ \text{s.t.} & \mathbf{x}_j - \mathbf{y}_j = \mathbf{0},\ \forall j. \end{array} \tag{6}$$

For fixed $\mathbf{y}$'s, the objective function is a linear square function of $\mathbf{x}$'s; and for fixed $\mathbf{x}$'s, the objective function is a linear square function of $\mathbf{y}$'s.

Develop an ADMM method to minimize the objective function by treating $\mathbf{x}$s and $\mathbf{y}$s as two blocks of variables so that each block optimization problem within any ADMM iteration is a convex quadratic minimization problem.

3 Project Goals

You may explore this new approach by generating large-scale noisy SNL problems up to 10,000 sensors and 1,000 anchors (2). The key questions are: what are the best approaches to efficiently estimate the positions of sensors? What are the differences among different methods? The comparison can include aspects such as algorithm design, theoretical analysis, computation time, and the approximation error of different algorithms.

References

[1] P. Biswas and Y. Ye, "Semidefinite programming for ad hoc wireless sensor network localization," in *Proceedings of the third international symposium on Information processing in sensor networks*, ACM Press, 2004, pp. 46–54.

[2] Pratik Biswas, T-C Liang, K-C Toh, Y. Ye, T-C Wang, "Semidefinite programming approaches for sensor network localization with noisy distance measurements," IEEE transactions on automation science and engineering, 3(4), 2006, pp. 360-371.

[3] L. Doherty, L. E. Ghaoui, and K. S. J. Pister, "Convex position estimation in wireless sensor networks." in *Proceedings of IEEE Infocom*, Anchorage, Alaska, April 2001, pp. 1655 –1663.

[4] A. So. A Semidefinite Programming Approach to the graph realization problem: Theory, Applications and Extensions *Phd Thesis, Stanford University*, 2007.

[5] C. Zhang, D. Ge, C. He, B. Jiang, Y. Jiang, Y. Ye, "DRSOM: A Dimension Reduced Second-Order Method and Preliminary Analyses" *arXiv preprint arXiv:2208.00208*, 2022.