www.ahtakoru.com
February 8, 2016

Finding Minimum Volume Ellipsoid Enclosing N Points via Semi-Definite Programming

Ahmet Taha Koru[1]*

Abstract
The constraints which guarantees the ellipsoid to enclose N points in eucledian space is presented. Then, the volume of the ellipsoid is formulated as the cost function of the optimization problem. Semi-definite optimization problem is solved by Matlab.

Keywords
Semi-definite programming — minimize ellipsoid volume — yalmip — sdpt3

[1]*Department of Mechatronics, Yıldız Technical University, İstanbul, Turkey*
***Corresponding author**: ahtakoru@gmail.com

1. Optimization Problem

The set of points in an n-dimensional ellipsoid is the following set:

$$\xi = \left\{ x \,|\, (x-c)^T A (x-c) \le 1\,, x \in \mathbb{R}^{n\times 1} \right\}. \tag{1}$$

where $c \in \mathbb{R}^{n\times 1}$ is the center of ellipsoid and the symmetric matrix $A = A^T \in \mathbb{R}^{n\times n}$ involves the information of rotation and radii of the ellipsoid.

Consider there are N points in Euclidean space $\mathbb{R}^n$ to be enclosed by ellipsoid. Let those points be

$$\{ x_1, x_2, \dots, x_N \}.$$

Since those points are enclosed by ellipsoid, they are in the set in (1). Hence, for each point i, the inequality

$$(x_i - c)^T A (x_i - c) \le 1, \tag{2}$$

is satisfied.

In order to represent this inequality in linear matrix inequality (LMI) form, let me present the following change of variables,

$$\bar{A} = A^{1/2}, \qquad \bar{c} = -\bar{A}c. \tag{3}$$

Then, the inequality in the Equation (2) becomes

$$\begin{aligned} \left(\bar{A}x_i + \bar{c}\right)^T \left(\bar{A}x_i + \bar{c}\right) &\le 1 \\ 1 - \left(\bar{A}x_i + \bar{c}\right)^T \left(\bar{A}x_i + \bar{c}\right) &\ge 0, \end{aligned} \tag{4}$$

The Schur complement of Equation (4) is the following inequality:

$$\begin{bmatrix} I & \bar{A}x_i + \bar{c} \\ * & 1 \end{bmatrix} \succeq 0.$$

The radii of the ellipsoid before rotation is related to the eigenvalues of A. If we call eigenvalues as $\lambda_1, \lambda_2, \cdots, \lambda_n$, the radii with respect to principal axes are

$$\sqrt{\lambda_1^{-1}}, \sqrt{\lambda_2^{-1}}, \cdots, \sqrt{\lambda_n^{-1}}.$$

Hence, the volume of the ellipsoid is proportional to $\det A^{-1}$. The determinant function is neither convex nor concave. Instead, we minimize the logarithm of the determinant which is a concave function. Furthermore, to get rid of the inverse of the matrix, we minimize $-\log\det\bar{A}$ function which ends up with minimum volume ellipsoid.

Finding the minimum volume ellipsoid enclosing N points in Euclidean space is the following semi-definite programming problem:

$$\begin{aligned} \min \quad & -\log\det\bar{A} \\ \text{subject to} \quad & \bar{A} \succeq 0 \\ & \begin{bmatrix} I & \bar{A}x_i + \bar{c} \\ * & 1 \end{bmatrix} \succeq 0, \forall i \in \{1, 2, \cdots, N\} \end{aligned}$$

2. Solving via Matlab

In order to model semi-definite problems in Matlab, Yalmip toolbox is widely used which can be found free on internet [2]. Since the optimization problem involves log det, the SDPT3 toolbox is used as solver [3]. In order to visualise the result, i use the script in [4].

An example with 100 random points is solved to illustrate the algorithm. The result can be seen in Figure 1. The Matlab code is presented below.

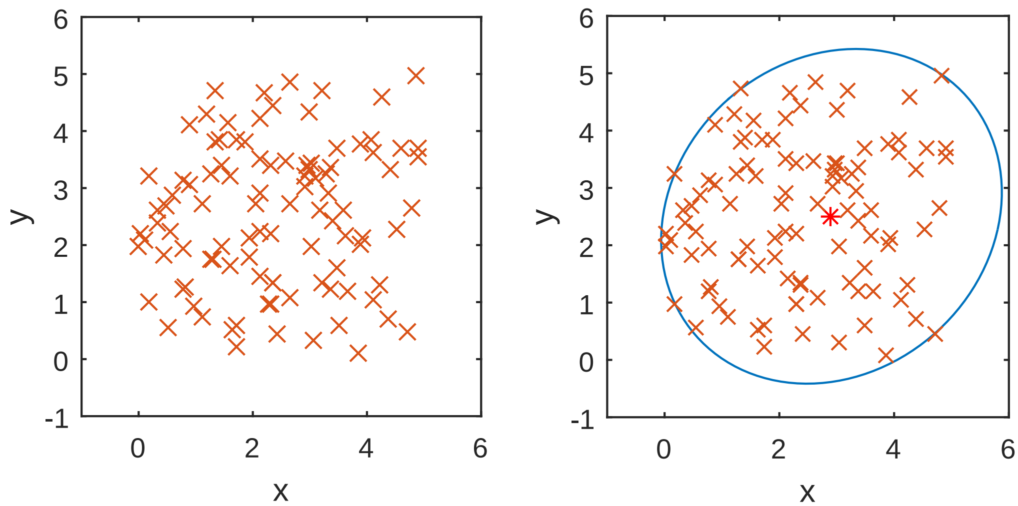

Figure 1. An example. Minimum ellipsoid enclosing 100 random points.

```
ops = sdpsettings('solver','sdpt3');

n = 2;   % The dimension of the ellipsoid
N = 100; % Number of points to be enclosed

points.x = rand(N,1)*5; % Generating ...
    random x coordinates for N points
points.y = rand(N,1)*5; % Generating ...
    random y coordinates for N points

A = sdpvar(n);
b = sdpvar(n,1);

Fset = [A ≥ 0]; % First condition, ...
    positive semi-definite A
% The LMI conditions for points to be ...
    enclosed by ellipsoid
for k = 1:length(points.x)
    pk = [points.x(k), points.y(k)];
    LMI1 = [eye(2)              , A*pk' + b;
            transpose(A*pk' + b), 1 ...
                        ];
    Fset = Fset + [LMI1 ≥ 0];
end
sol = optimize(Fset, [-logdet(A)], ops); % ...
    Optimizasyon problemini coz
% Converting optimal A and b from sdpvar ...
    to double
Ad = double(A); bd = double(b);

% Plotting the ellipsoid with the original ...
    A and b before change of var
Ellipse_plot(Ad^2, -inv(Ad)*bd);
hold on;
plot(points.x, points.y, 'x');
xlabel('x');
ylabel('y');
```

References

[1] Boyd, S., ve Vandenberghe, L. Convex optimization. Cambridge university press, 2004.

[2] Löfberg, J. ”YALMIP: A toolbox for modeling and optimization in MATLAB.” Computer Aided Control Systems Design, 2004 IEEE International Symposium on. IEEE, 2004.

[3] Toh, K. C., Todd, M. J., ve Tütüncü, R. H.. ”SDPT3—a MATLAB software package for semidefinite programming, version 1.3.” Optimization methods and software 11.1-4 (1999): 545-581.

[4] Moshtagh, N. ”Plot An Ellipse In Center Form”. 2016. http://www.mathworks.com/matlabcentral/fileexchange/13844-plot-an-ellipse-in–center-form-