Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Tractable fitting with convex polynomials via sum-of-squares

Alessandro Magnani
Department of Electrical Engineering
Stanford University
Stanford, CA 94305-9510
Email: alem@stanford.edu

Sanjay Lall
Department of Aeronautics and Astronautics
Stanford University
Stanford, CA 94305-4035
Email: lall@stanford.edu

Stephen Boyd
Department of Electrical Engineering
Stanford University
Stanford, CA 94305-9510
Email: boyd@stanford.edu

***Abstract*— We consider the problem of fitting given data $(u_1, y_1), \ldots, (u_m, y_m)$ where $u_i \in \mathbf{R}^n$ and $y_i \in \mathbf{R}$ with a convex polynomial f. A technique to solve this problem using sum of squares polynomials is presented. This technique is extended to enforce convexity of f only on a specified region. Also, an algorithm to fit the convex hull of a set of points with a convex sub-level set of a polynomial is presented. This problem is a natural extension of the problem of finding the minimum volume ellipsoid covering a set. The algorithm, like that for the minimum volume ellipsoid problem, has the property of being invariant to affine coordinate transformations. We generalize this technique to fit arbitrary unions and intersections of polynomial sub-level sets.**

I. Introduction

We consider the problem of fitting given data

$$(u_1, y_1), \quad \ldots, \quad (u_m, y_m)$$

with $u_i \in \mathbf{R}^n$ and $y_i \in \mathbf{R}$ with a convex polynomial f.

Given polynomials $p_1, \ldots, p_w$ in n variables, we restrict the polynomials we are considering so that f has the form

$$f = c_1 p_1 + \cdots + c_w p_w,$$

where c_i for $i = 1, \ldots, w$, are reals. We would like to choose variables $c = (c_1, \ldots, c_w)$. For example, this description of f allows us to describe the set of polynomials of degree less than a constant or the polynomials of a specific degree.

Using least-squares fitting we obtain the problem

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^m (f(u_i) - y_i)^2 \\ \text{subject to} & f \quad \text{is convex,} \end{array} \tag{1}$$

One may also consider other norms but we will use the above formulation in this paper.

In some special cases the solution to this problem is known. If, for example, the polynomials p_i are such that f is affine in x and therefore convex, we have that f has the form $f = c_1 + c_2 x_1 + \cdots + c_{n+1} x_n$ and the problem becomes

$$\text{minimize} \quad \sum_{i=1}^m (f(u_i) - y_i)^2.$$

This is a least-squares problem with variable c_i and it has an analytical solution.

This work is supported in part by a Stanford Graduate Fellowship.

If instead the polynomials p_i have degree less than or equal to 2 then f is a quadratic form and can be written as

$$f(x) = x^T A x + b^T x + r,$$

where A, b, and r linearly depend on c. Since imposing the convexity of f is equivalent to imposing A to be positive semidefinite, the problem becomes

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^m (f(u_i) - y_i)^2 \\ \text{subject to} & f(x) = x^T A x + b^T x + r, \\ & A \succeq 0. \end{array}$$

This problem is a *semidefinite program* (SDP) [BV03] with variables A, b, and r and can be solved efficiently.

In the general case, if we consider the set $\mathcal{C}$ of coefficients such that f is convex

$$\mathcal{C} = \{c \mid f = c_1 p_1 + \cdots + c_d p_w, f \text{ is convex}\},$$

problem (1) can be rewritten as

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^m (f(u_i) - y_i)^2 \\ \text{subject to} & c \in \mathcal{C}. \end{array} \tag{2}$$

Since the set $\mathcal{C}$ is convex, this is a convex optimization problem. Nevertheless, since there is no known tractable description of the set $\mathcal{C}$ in general and so the problem is hard to solve.

We will consider a subset of $\mathcal{C}$ so that the problem becomes tractable. We will also show conditions under which one can solve the original problem exactly.

II. Convex polynomials via SOS

We first consider the problem of imposing convexity on a generic polynomial f in n variables of the form $f = c_1 p_1 + \cdots + c_w p_w$ where p_i, $i = 1, \ldots, w$, are given polynomials in n variables, $c = (c_1, \ldots, c_w) \in \mathbf{R}^w$, and d is the degree of f.

We know that a necessary and sufficient condition for f to be convex is that

$$h = s^T \nabla^2 f(x) s \geq 0 \quad \text{for all } x, s. \tag{3}$$

Notice that h is a polynomial expression with variables s and x and moreover is of the same degree d as f.

A polynomial $g(t)$ such that $g(t) \geq 0$ for all $t \in \mathbf{R}^n$ is called *positive semidefinite* (PSD). Therefore f is convex if and only if h is PSD.

Except for special cases (*e.g.*, $n = 1$ or $d = 2$), it is NP-hard to determine whether or not a given polynomial is PSD, let alone solve an optimization problem, with the coefficients of c as variables, with the constraint that h is PSD.

A famous sufficient condition for a polynomial to be PSD is that it has the form

$$g(x) = \sum_{i=1}^{r} q_i(x)^2,$$

for some polynomials q_i, with degree no more than $d/2$. A polynomial g that has this sum-of-squares form is called SOS.

The condition that a polynomial g be SOS (viewed as a constraint on its coefficients) turns out to be equivalent to an *linear matrix inequality* (LMI) ([Nes00], [Par00]). In particular a polynomial g of even degree w is SOS if and only if there exist monomials of degree less that $d/2$, $e_1, \ldots, e_s$ and a positive semidefinite matrix V such that

$$g = e^T V e. \tag{4}$$

Since the condition $g = e^T V e$ is a set of linear equality constraints relating the coefficients of g to the elements of V, we have that imposing the polynomial g to be SOS is equivalent to the positive semidefiniteness constraint that $V \succeq 0$ together with a set of linear equality constraints.

We will impose convexity on the polynomial f by requiring h to be SOS. We then clearly have

$$\mathcal{S} = \{c \mid h \text{ is SOS}\} \subseteq \mathcal{C}.$$

Since the condition of a polynomial being PSD is not equivalent to being SOS, in general $\mathcal{C} \neq \mathcal{S}$ and therefore by imposing h to be SOS, we are not considering all the possible convex polynomials but only a subset of them. Only in special cases does $\mathcal{S} = \mathcal{C}$, for example if $n = 1$ or $d = 2$.

As mentioned above, the advantage of h being SOS is that imposing this constraint can be cast as LMI and handled efficiently [BGFB94].

III. Function fitting via SOS

Using the approximation of the previous section to solve problem (1), we obtain

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{m} (f(u_i) - y_i)^2 \\ \text{subject to} & c \in \mathcal{S}. \end{array} \tag{5}$$

Equivalently, using the necessary and sufficient condition for a polynomial to be SOS, we obtain the problem

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{m} (f(u_i) - y_i)^2 \\ \text{subject to} & h = s^T \nabla^2 f(x) s = e^T V e \quad \text{for all } x, s \\ & V \succeq 0, \end{array} \tag{6}$$

where e is a vector of monomials in s and x and the variables are the matrix V and c. Since the equation $h = e^T V e$ is simply a set of linear equations in the coefficients of V and c, this problem can be cast as a semidefinite program for which there are efficient algorithms [BV03], [VB96].

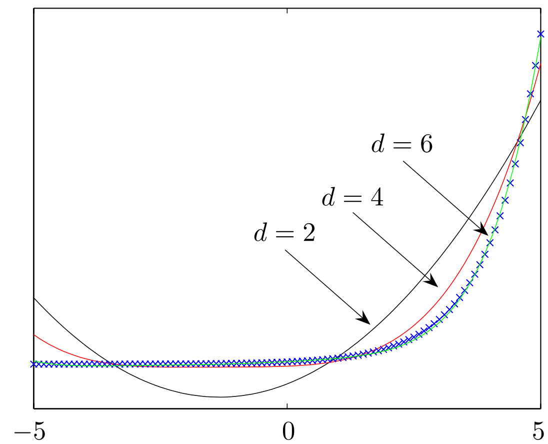

Fig. 1: Convex polynomial fitting example.

A. Numerical example

We present a very simple example for $n = 1$, where the data u_i for $i = 1, \ldots, 100$, is obtained by uniformly sampling the interval $[-5, 5]$ and $y_i = \exp(u_i)$. In this case, since $\mathcal{S} = \mathcal{C}$ we can tractably solve problem (1). Figure 1 shows an example, where stars correspond to given data points.

IV. Minimum volume set fitting

In this section we address the problem of finding a convex set P, described through a sub-level set of a convex polynomial g, that contains a set of points and is close in some sense to them. We would like, for example, to find the minimum volume set P that includes all points u_i.

As before, given polynomials $p_1, \ldots, p_w$ we restrict ourselves to consider a polynomial g of the form

$$g = b_1 p_1 + \cdots + b_w p_w,$$

where we would like to choose $b \in \mathbf{R}^w$. Therefore we want to solve the problem

$$\begin{array}{ll} \text{minimize} & \text{volume}(P) \\ \text{subject to} & P = \{x \mid g(x) \leq 1\} \\ & u_i \in P \quad \text{for all } i = 1, \ldots, m, \\ & P \text{ is convex.} \end{array} \tag{7}$$

If for example g is a polynomial of degree 2, P will be the minimum volume ellipsoid containing all the data points. This is a well-known problem [BV03] and if we write g as $g = x^T A x + b^T x + r$ we then g is convex if and only if A is positive semidefinite. The above problem then becomes

$$\begin{array}{ll} \text{minimize} & \text{volume}(P) \\ \text{subject to} & u_i^T A u_i + b^T u_i + r \leq 1, \quad i = 1, \ldots, m \\ & A \succeq 0. \end{array}$$

We can assume without loss of generality that $g(x) \geq 0$ for all x, in which case the volume of P is proportional to

$\sqrt{\det A^{-1}}$ and we can write the problem as

$$\begin{array}{ll} \text{minimize} & \log\det A^{-1} \\ \text{subject to} & u_i^T A u_i + b^T u_i + r \leq 1, \quad i = 1, \ldots, m \\ & A \succeq 0 \\ & \left[\begin{array}{cc} A & b \\ b^T & r \end{array}\right] \succeq 0, \end{array}$$

where the last constraint is equivalent to $g(x) \geq 0$ for all x. This problem can be cast as an SDP [NN95].

In the general case the problem can be written as

$$\begin{array}{ll} \text{minimize} & \text{volume}(P) \\ \text{subject to} & u_i \in P \quad \text{for } i = 1, \ldots, m, \\ & h = s^T \nabla^2 g(x) s \geq 0 \quad \text{for all } x, s. \end{array}$$

Now not only is the second constraint hard to handle exactly, but there is also no known way to efficiently compute the volume of P. We propose a heuristic algorithm that tries to shrink the set P around the data points and that for $d = 2$ is equivalent to the minimum volume ellipsoid. This problem has possible applications in data mining [KNZ01], [LN95], robotics, and computer vision [KG99], [TCS+94], [RB97].

A. Pseudo minimum volume heuristic

The main idea of the algorithm is to increase the curvature of g along all directions so that the set P gets closer to the points u_i. Since the curvature of g along the direction s is proportional to

$$h = s^T \nabla^2 g(x) s,$$

we will write h in a specific form so that we can, at the same time, enforce h to be PSD and increase the curvature of g.

The first step, as before, is to impose

$$h = s^T \nabla^2 g(x) s \quad \text{is SOS},$$

or equivalently

$$\begin{array}{l} h = s^T \nabla^2 g(x) s = e^T V e \quad \text{for all } x, s \\ V \succeq 0, \end{array}$$

where e is a vector of monomials in x and s. In this way we have that g is convex. Similarly to the case of the minimum volume ellipsoid, we maximize the determinant of V which has the effect of increasing the curvature of g along all possible directions.

The heuristic becomes

$$\begin{array}{ll} \text{minimize} & \log\det V^{-1} \\ \text{subject to} & g(x) \geq 0 \quad \text{for all } x, \\ & s^T \nabla^2 g(x) s = e^T V e \quad \text{for all } x, s \\ & g(u_i) \leq 1 \quad i = 1, \ldots, m. \end{array}$$

Again replacing the positivity constraint $g(x) \geq 0$ by an SOS condition, we arrive at

$$\begin{array}{ll} \text{minimize} & \log\det V^{-1} \\ \text{subject to} & g \text{ is SOS} \\ & s^T \nabla^2 g(x) s = e^T V e \quad \text{for all } x, s \\ & g(u_i) \leq 1 \quad i = 1, \ldots, m, \end{array}$$

or equivalently

$$\begin{array}{ll} \text{minimize} & \log\det V^{-1} \\ \text{subject to} & g = h^T C h, \\ & C \succeq 0, \\ & s^T \nabla^2 g(x) s = e^T V e \quad \text{for all } x, s \\ & g(u_i) \leq 1 \quad i = 1, \ldots, m, \end{array} \tag{8}$$

where h is a vector of monomials in x and e is a vector of monomials in x and s. This problem can now be solved efficiently.

It is clear that for $d = 2$, the problem reduces to finding the minimum volume ellipsoid. Note that the matrix C is not unique and it depends on the choice of monomials e. It is also possible for the heuristic to fail; for example, if we choose a redundant set of monomials for e, then C may not be full rank and the determinant of C will be zero. One workaround for this is to use fewer monomials for e. Moreover we should notice that it is not strictly needed for e to be made out of monomials but any polynomial expression would work.

It can be shown (see Appendix) that, under some minor conditions, the solution to this problem has the nice property of being invariant to affine coordinate transformations. In other words, if P is the solution of the problem, by changing the coordinates of the points u_i through an affine transformation, we would have that the set P scaled by the same transformation, is an optimal point for the problem in the new set of coordinates.

1) Example: We show in a simple case how to derive the matrices C and V for problem (8). Suppose g has the form

$$g(x, y) = c_1 + c_2 x^4 y^4,$$

we can choose the vectors of monomials h as

$$h = (1, x^2 y^2).$$

With this choice of h the matrix C will be

$$C = \left[\begin{array}{cc} c_1 & 0 \\ 0 & c_2 \end{array}\right].$$

We then have

$$h = s^T \nabla^2 g(x) s = 12 c_3 x^2 y^4 s_1^2 + 12 c_3 x^4 y^2 s_2^2 + 32 x^3 y^3 s_1 s_2,$$

and by picking the vector of monomials e to be

$$e = (x y^2 s_1, x^2 y s_2),$$

we obtain

$$h = e^T V e = e^T \left[\begin{array}{cc} 12 c_3 & 16 c_3 \\ 16 c_3 & 12 c_3 \end{array}\right] e.$$

2) Numerical example: We show the result of the algorithm for a set of points corresponding to the simulated first 150 steps of a dynamical system. We pick g to be a generic polynomial of degree less than d. Figure 2 shows the level set for different degrees d of the polynomial g.

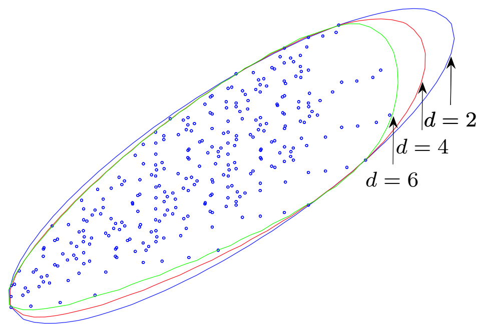

Fig. 2: Pseudo minimum volume example.

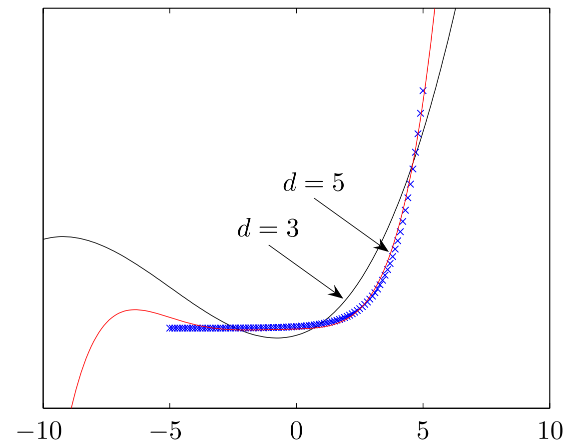

Fig. 3: Conditional convex polynomial fitting.

V. Conditional convex polynomial fitting

In this section we want to solve problem (1) with the relaxed constraint that f is convex only over a set P and not on the entire space

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{m}(f(u_i) - y_i)^2 \\ \text{subject to} & f \quad \text{is convex on } P. \end{array} \tag{9}$$

We require that the set P contains the points u_i and that is convex. Moreover the set P should be described as the sub-level set of a polynomial g

$$P = \{x \mid g(x) \leq 1\}.$$

For example, the previously presented algorithm gives us a set P with the required properties that can be used to solve problem (9).

We will write a sufficient condition for f to be convex over the set P. We will show that for P compact this condition has a nice property that allows to prove a stronger result. For $h = s^T \nabla^2 f s$ we define

$$l(x, s) = h(x, s) + w(x, s)(1 - g(x)), \tag{10}$$

where w is a sum of squares polynomial. It is clear that if l is SOS then the function f will be convex over the set P. Vice versa it can be shown [Sch91] that if P is compact and h is strictly positive over P, there exist SOS polynomials l and w so that (10) holds.

Therefore, by using this condition to impose convexity of f over P, the problem becomes

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{m}(f(u_i) - y_i)^2 \\ \text{subject to} & s^T \nabla^2 f s - w(x, s)(1 - g(x)) \quad \text{is SOS} \\ & w \quad \text{is SOS}. \end{array}$$

Notice that we have a wide range of choice for the polynomial w since the only constraint is that it should be SOS. Therefore we cannot solve this problem because to describe the polynomial w we would need an infinite number of variables. Nevertheless we should notice that if we were able to solve this problem and P was compact, we would be able to find a polynomial for which the cost function is no greater than the optimal value of

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{m}(f(u_i) - y_i)^2 \\ \text{subject to} & f \quad \text{is strictly convex on } P. \end{array} \tag{11}$$

To make this problem tractable we can, for example, impose the maximum degree of w to be less than a given constant t. In this case, w will have the form

$$w = h^T W h,$$

where h is a vector of all monomials of degree less or equal than $t/2$ and W is a generic positive semidefinite matrix.

Once we fix the order of the polynomial w, the problem can be cast as a convex program (SDP) and solved efficiently. We obtain the problem

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{m}(f(u_i) - y_i)^2 \\ \text{subject to} & s^T \nabla^2 f s - w(x, s)(1 - g(x)) \quad \text{is SOS} \\ & w = h^T W h, \\ & W \succeq 0, \end{array}$$

where the variables are c and W and h is a vector of monomials of degree less or equal than $t/2$. By increasing t we obtain larger problems that in the limit tend to a solution for which the cost function is not greater than the optimal value of problem (11).

A. Numerical example

We solve the same numerical example presented in section III-A but imposing convexity only on the interval $[-5, 5]$. In this way we can, for example, fit using odd degree polynomials. We describe the interval with $g(x) = x^2 - 24$ and we fix the degree of w to be less or equal than 4. In particular figure 3 shows the result for a third and fifth order polynomial. Clearly the function is not convex on $\mathbf{R}$ but it is still convex on the interval $[-5, 5]$.

VI. Extensions

We present two simple extensions. The first one allows to fit a set of points described through the intersection of two sub-level sets of polynomials. The second one extends the results of the paper to a different class of polynomials.

A. Fitting the intersection of sub-level sets of polynomials

One simple extension of the pseudo minimum volume heuristic is to find a convex set P that fits a set of the form

$$K = \{x \mid f_i(x) \leq 0 \quad i = 1, 2\},$$

where f_i are polynomials.

We can write a sufficient condition for P to contain K. In particular, if we have

$$(1 - g(u_i)) + w_1 f_1 + w_2 f_2 - w_3 f_1 f_2 \quad \text{is SOS}, \qquad (12)$$

where w_i are SOS, clearly the set P will contain the set K. It can also be shown [Sch91] that if K is compact and P is such that $\hat{x} \in K$ implies $g(\hat{x}) < 1$ then there exist SOS polynomials w_i such that (12) is verified.

The heuristic can be modified to impose $K \subseteq P$ as

$$\begin{array}{ll} \text{minimize} & \log\det C^{-1} \\ \text{subject to} & g \text{ is SOS} \\ & s^T \nabla^2 g(x) s = e^T C e \quad \text{for all } x, s \in \mathbf{R}^n \\ & (1 - g(u_i)) + \sum_i w_i f_i - w_3 f_1 f_2 \quad \text{is SOS}, \\ & w_1, w_2, w_3 \quad \text{are SOS}. \end{array}$$

To be able to solve this problem, we need to impose some more constraints on the polynomials w_i since the only constraint is that they should be SOS. As we did before, we can impose them to have a maximum degree less than some constant, and the resulting optimization problem is an SDP. With similar techniques one may also also easily handle union and intersection of such sets.

B. Convexity along a line passing through a point

We can extends the techniques presented in this paper to a different class of polynomials [KG99], that are convex only when restricted to a line passing for a given point x_0.

Given a polynomial f and a point x_0, the property is equivalent to

$$h(x) = (x - x_0)^T \nabla^2 f(x)(x - x_0) \geq 0 \quad \text{for all } x.$$

In other words we are replacing the generic direction s in (3) along which the curvature of the polynomial is evaluated, with the direction $x - x_0$ that goes through the point x_0.

We can therefore apply the function fitting algorithm (5) and the pseudo minimum volume algorithm (8) for polynomials with this property by simply substituting s with $x - x_0$. We should point out that in this case the algorithm loses the property of being invariant under an affine coordinate transformation.

Appendix

Given problem (8), we would like to show the relationship between the solutions of it for two different systems of coordinates x, y such that $x = Ay + b$ where $\det A \neq 0$. In particular we have that, if u_i, v_i for $i = 1, \ldots, m$ are the points in the first and second system of coordinates respectively,

$$u_i = Av_i + b.$$

We will use subscript x to refer to the problem with the x coordinates and a y subscript for the problem in the y coordinates. We call, for example, e_x and h_x the vectors e and h in the first system of coordinates and e_y and h_y in the second. We also call $\hat{e}_y$ and $\hat{h}_y$ the vectors e_x and h_x where each component as been transformed in the other system of coordinates so that $x = Ay + b$ and $s_x = As_y$. Therefore each component of $\hat{h}_y$, for example, is a polynomial in y and $\hat{e}_y$ depends only on y and s_y. The same holds for $\hat{e}_x$ and $\hat{h}_x$ which are the vectors e_y and h_y in the other system of coordinates.

We make the assumption that the vector $\hat{e}_y$ can be represented as a linear combination of e_y and that $\hat{e}_x$ is a linear combination of e_x. Moreover we require the same property for the vectors h_x and h_y. This assumption is satisfied, for example, if h_x consists of all monomials up to a certain degree in x and the same choice is made for h_y. In other words, we require that in the two systems of coordinates, we can describe the same set of polynomials.

Given this property we have that

$$\begin{aligned} \hat{h}_y &= U_1 h_y, \\ \hat{e}_y &= U_2 e_y, \end{aligned}$$

where U_1 and U_2 are matrices that depend nonlinearly on A and b. So suppose that $g(x)$ is a feasible solution for the problem in the x coordinates. We will show that the polynomial $f(y) = g(Ay+b)$ is feasible in the second system of coordinates.

We have that

$$f(v_i) = g(Av_i + b) = g(u_i) \leq 1,$$

and therefore the points are included in the sub-level set. We also have that

$$\begin{aligned} f(y) &= g(Ay + b) \\ &= \hat{h}_y^T C_x \hat{h}_y \\ &= h_y^T U_1^T C_x U_1 h_y, \end{aligned}$$

where clearly $C_y = U_1^T C_x U_1 \succeq 0$, and

$$\begin{aligned} s_y^T \nabla^2 f(y) s_y &= s_y^T \nabla^2 g(Ay + b) s_y \\ &= s_y^T A^T \nabla_x^2 g(Ay + b) A s_y \\ &= \hat{e}_y^T V_x \hat{e}_y \\ &= e_y^T U_2^T V_x U_2 e_y, \end{aligned}$$

where $V_y = U_2^T V_x U_2 \succeq 0$. It is also clearly true that

$$\log\det V_x^{-1} = 2\log\det U_2 + \log\det V_y,$$

in other words the same polynomial after a coordinate transformation is still feasible for the second problem and produce a value which is the same except for a constant independent of the polynomial. Since the same applies in the other direction, we can conclude that an optimal solution to the first problem will be optimal for the second one too.

REFERENCES

[BGFB94] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory.* SIAM, 1994.

[BV03] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2003.

[KG99] D. Keren and C. Gotsman. Fitting curves and surfaces with constrained implicit polynomials. *IEEE Trans. Pattern Anal. Mach. Intell.*, 21(1):31–41, 1999.

[KNZ01] E. M. Knorr, R. T. Ng, and R. H. Zamar. Robust space transformations for distance-based operations. In *KDD '01: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 126–135. ACM Press, 2001.

[LN95] D. Levin and E. Nadler. Convexity preserving interpolation by algebraic curves and surfaces. *Numerical Algorithms*, 9:113–139, 1995.

[Nes00] Y. Nesterov. Squared functional systems and optimization problems. In J. Frenk, C. Roos, T. Terlaky, and S. Zhang, editors, *High Performance Optimization Techniques*, pages 405–440. Kluwer, 2000.

[NN95] Y. Nesterov and A. Nemirovskii. *Interior-point polynomial algorithms in convex programming*. SIAM studies in applied mathematics, 1995.

[Par00] P. A. Parrilo. *Structured Semidefi nite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*. PhD thesis, California Institute of Technology, 2000.

[RB97] E. Rimon and S. Boyd. Obstacle collision detection using best ellipsoid fi t. *Journal of Intelligent and Robotic Systems*, 18:105–126, 1997.

[Sch91] K. Schmudgen. The k-moment problem for compact semi-algebraic sets. *Math. Ann.*, 289:203–206, 1991.

[TCS+94] G. Taubin, F. Cukierman, S. Sullivan, J. Ponce, and D. J. Kriegman. Parameterized families of polynomials for bounded algebraic curve and surface fi tting. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 16:287–303, 1994.

[VB96] L. Vandenberghe and S. Boyd. Semidefi nite programming. *SIAM Review*, 38(1):49–95, 1996.