Multivariate semi-algebraic super-resolution by semi-definite programming: numerical examples

Didier Henrion[1,2,3]

Draft of January 20, 2014

1 Total variation minimization

We consider examples of optimization problems

$$\begin{array}{ll} \inf & \|\mu\|_{TV} \\ \text{s.t.} & \langle a_i, \mu\rangle = b_i, \quad i = 1,\ldots,m \end{array} \tag{1}$$

where the minimization is with respect to a signed Borel measure $\mu \in \mathscr{M}(X;\mathbb{R})$ supported on a given compact set $X \subset \mathbb{R}^n$, subject to a finite number of linear constraints

$$\langle a_i, \mu\rangle := \int a_i(x) d\mu(x)$$

where each $a_i \in \mathscr{C}(X;\mathbb{R})$ is a given continuous function and $b_i \in \mathbb{R}$ a given real number, and the objective function is the total variation of μ.

Decomposing

$$\mu = \mu^+ - \mu^-$$

for some nonnegative Borel measures μ^+, μ^-, optimization problem (1) can be rewritten equivalently as a linear programming (LP) problem in the cone $\mathscr{M}^+(X)$ of nonnegative elements of $\mathscr{M}(X)$:

$$\begin{array}{rcll} p^* & = & \inf & \langle 1, \mu^+\rangle + \langle 1, \mu^-\rangle \\ & & \text{s.t.} & \langle a_i, \mu^+\rangle - \langle a_i, \mu^-\rangle = b_i, \quad i = 1,\ldots,m \\ & & & \mu^+ \geq 0, \quad \mu^- \geq 0. \end{array} \tag{2}$$

[1]CNRS; LAAS; 7 avenue du colonel Roche, F-31077 Toulouse; France. henrion@laas.fr

[2]Université de Toulouse; UPS, INSA, INP, ISAE; UT1, UTM, LAAS; F-31077 Toulouse; France

[3]Faculty of Electrical Engineering, Czech Technical University in Prague, Technická 2, CZ-16626 Prague, Czech Republic

If function a_i is the i-th entry of a column vector $a \in \mathscr{C}(X;\mathbb{R}^m)$, problem (2) is the dual of the following LP problem in the cone $\mathscr{C}^+(X)$ of nonnegative elements of $\mathscr{C}(X)$:

$$\begin{array}{rcll} d^* & = & \sup & b^T u \\ & & \text{s.t.} & z^+(x) := 1 - a^T(x)u \geq 0 \quad \text{a.e. } x \in X \\ & & & z^-(x) := 1 + a^T(x)u \geq 0 \quad \text{a.e. } x \in X \end{array} \tag{3}$$

where the maximization is w.r.t. a vector $u \in \mathbb{R}^m$. Denoting

$$\|z\|_\infty := \sup_{x \in X} |z(x)|$$

remark that LP problem (3) can be also written as

$$\begin{array}{rcll} d^* & = & \sup & b^T u \\ & & \text{s.t.} & \|a^T(x)u\|_\infty \leq 1. \end{array}$$

and that there is no duality gap, i.e. $p^* = d^*$.

Our numerical examples are solved with the interface GloptiPoly 3 which is designed to generate semidefinite relaxations of measure LP problems with polynomial data. So we assume that the functions $a_i(x)$ in LP problem (2) are multivariate polynomials, and for notational simplicity, we let $a_i(x) := x^{\alpha_i}$ where $\alpha_i \in \mathbb{N}^n$ are given[1]. Note that the choice of monomials is only motivated for notational simplicity, and that other choices of polynomials (e.g. Chebyshev polynomials) are typically preferable numerically[2]. Semidefinite relaxations are then solved with SeDuMi 1.3, an implementation of a primal-dual interior-point algorithm.

2 Univariate example

We want to recover the measure

$$\mu := \delta_{-3/4} + \delta_{1/2} - \delta_{1/8}$$

on the disconnected set $X := [-1, -1/2] \cup [0, 1]$ which can be modeled as the polynomial superlevel set $X = \{x \in \mathbb{R} \,:\, g_1(x) \geq 0\}$ for the choice

$$g_1(x) := -(x+1)(x+1/2)x(x-1).$$

In LP (2) we let $a_i(x) := x^i$ and $b_i := (-3/4)^i + (1/2)^i - (1/8)^i$ for $i = 0, 1, 2 \ldots, k$. The following GloptiPoly 3 code allows to construct a semidefinite relaxation of given order k for this measure reconstruction problem:

```
k = input('relaxation order=');
mpol xp
```

[1] We use the multi-index notation $x^\beta := \prod_{j=1}^n x_j^{\beta_j}$ for every $x = (x_1, \ldots, x_n) \in \mathbb{R}^n$ and $\beta = (\beta_1, \ldots, \beta_n) \in \mathbb{N}^n$

[2] A numerical analysis of the impact of the basis is however out of the scope of our work.

```
mup = meas(xp);
mpol xm
mum = meas(xm);
% build problem data
p = (0:k)'; % powers
B = zeros(size(p));
Xp = [-3/4,1/2]; Xm = [1/8];
for i = 1:length(Xp), B = B + Xp(i).^p; end  % positive atoms
for i = 1:length(Xm), B = B - Xm(i).^p; end % negative atoms
% build semidefinite relaxation
P = msdp(min(mass(mup)+mass(mum)),...
       mom(xp.^p)-mom(xm.^p)==B,...
       (xp+1)*(xp+1/2)*xp*(xp-1)<=0, (xm+1)*(xm+1/2)*xm*(xm-1)<=0);
% semidefinite program in SeDuMi format
[A,b,c,K,b0,s] = msedumi(P);
% solve semidefinite program
[x,y] = sedumi(A,b,c,K);
% extract moment matrices
z=c-A'*y;
Mm=reshape(z(K.f+K.l+(1:K.s(1)^2)),K.s(1),K.s(1));
Mp=reshape(z(K.f+K.l+K.s(1)^2+(1:K.s(2)^2)),K.s(2),K.s(2));
% lower bound on the objective function
disp(['lower bound=' num2str(b0+s*c'*x)]);
% try to extract the atoms
try
 mext(Mm,1,k) % positive atoms
 mext(Mp,1,k) % negative atoms
end
% display polynomial certificate
zp=rot90(x(1:K.f),-1);
X=linspace(-1,1,1e3);
close all
plot(X,polyval(zp,X),'k','linewidth',3)
hold on
plot([-1 1],[1 1],':k');plot([-1 1],[-1 -1],':k');
for i = 1:length(Xp), plot(Xp(i),1,'.r','markersize',30); end
for i = 1:length(Xm), plot(Xm(i),-1,'.b','markersize',30); end
axis([-1 1 -1.1 1.3])
```

With this script we obtain the following hierarchy of lower bounds p_k^* (displayed with four significant digits) on the objective function p^*:

k	1	2	3	4	5	6	7	8	9
p_k^*	1.000	1.000	1.000	1.973	2.417	2.448	2.861	2.876	3.000

Solving the semidefinite relaxation of order 9 on our standard PC takes 0.2 seconds, and global optimality is certified with a rank 2 moment matrix $M(y^+)$ and a rank 1 moment matrix $M(y^-)$ from which the 3 points can be extracted. On Figure 1 we represent the

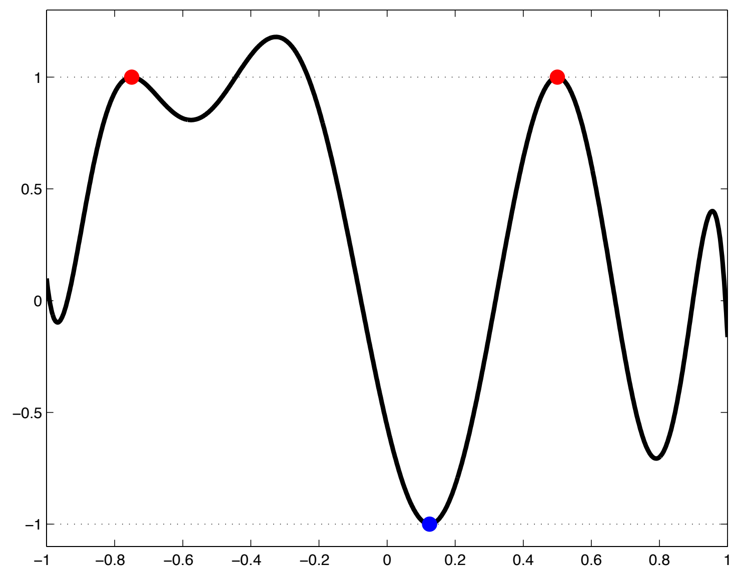

Figure 1: Degree 9 polynomial certificate for the univariate example.

degree 9 polynomial $\sum_{i=0}^{9} u_i x^i$ certifying global optimality. Indeed we can check that the polynomial attains the value $+1$ at the points $x = -3/4$, and $x = 1/2$ (in red), it attains the value -1 at the point $x = 1/4$ (in blue), while taking values between -1 and $+1$ on X. Notice in particular that the polynomial is bigger than $+1$ around $x = -1/4$, but this point is not in X.

3 Bivariate example

We want to recover the measure

$$\mu := \delta_{(-1/2,1/2)} + \delta_{(1/2,-1/2)} + \delta_{(1/2,1/2)} + \delta_{(0,0)} - \delta_{(0,-1/2)} - \delta_{(1/2,0)}$$

on the box $X := [-1, 1]^2$. With the following GloptiPoly 3 script, which is essentially the same as in the univariate case, we construct and solve semidefinite relaxations of this problem:

```
k = input('relaxation order=');
mpol xp 2
mup = meas(xp);
mpol xm 2
mum = meas(xm);
% build problem data
p = genpow(3,k); p = p(:,2:end); % powers
b = zeros(size(p,1),1);
Xp = [-1/2 1/2; 1/2 -1/2; 1/2 1/2; 0 0]; Xm = [0 -1/2; 1/2 0];
for i = 1:size(Xp,1), b = b + Xp(i,1).^p(:,1) .* Xp(i,2).^p(:,2); end % positive atoms
for i = 1:size(Xm,1), b = b - Xm(i,1).^p(:,1) .* Xm(i,2).^p(:,2); end % negative atoms
```

```
% build semidefinite relaxation
ME = [mass(mup)-mass(mum)==b(1)];
for i = 2:size(p,1)
 ME = [ME; mom(xp(1)^p(i,1).*xp(2)^p(i,2))-mom(xm(1)^p(i,1).*xm(2)^p(i,2))==b(i)];
end
MC = [xp(1)^2<=1, xp(2)^2<=1, xm(1)^2<=1, xm(2)^2<=1];
P = msdp(min(mass(mup)+mass(mum)),ME,MC);
% semidefinite program in SeDuMi format
[A,b,c,K,b0,s] = msedumi(P);
% solve semidefinite program
[x,y] = sedumi(A,b,c,K);
% extract moment matrices
z=c-A'*y;
Mp=reshape(z(K.f+K.l+(1:K.s(1)^2)),K.s(1),K.s(1));
Mm=reshape(z(K.f+K.l+K.s(1)^2+(1:K.s(2)^2)),K.s(2),K.s(2));
% lower bound on the objective function
disp(['lower bound=' num2str(b0+s*c'*x)]);
% try to extract the atoms
try
 mext(Mp,2,k) % positive atoms
 mext(Mm,2,k) % negative atoms
end
% display polynomial certificate
close all
[X1,X2]=meshgrid(linspace(-1,1,1e3));
certif=x(1:K.f)'*mmon(xp,k);
X3=eval(vectorize(certif,'X1','X2'));
mesh(X1,X2,X3,'FaceColor','interp','EdgeColor','none','FaceLighting','phong');
view(-15,15)
axis vis3d
camlight left
xlabel x_1
ylabel x_2
colormap spring
hold on
for i = 1:size(Xp,1), plot3(Xp(i,1),Xp(i,2),1,'.r','markersize',30); end
for i = 1:size(Xm,1), plot3(Xm(i,1),Xm(i,2),-1,'.b','markersize',30); end
```

With this script we obtain the following hierarchy of lower bounds p_k^* (displayed with four significant digits) on the objective function p^*:

k	1	2	3	4	5	6	7	8	9	10	11	12
p_k^*	2.000	2.068	2.210	2.481	3.011	3.277	4.170	4.795	5.374	5.899	5.986	6.000

Solving the semidefinite relaxation of order 12 on our standard PC takes less than 3 seconds, and global optimality is certified with a rank 4 moment matrix $M(y^+)$ and a rank 2 moment matrix $M(y^-)$ from which the 6 points can be extracted with a relative accuracy around 10^{-6}. On Figure 2 we represent the degree 12 polynomial certifying

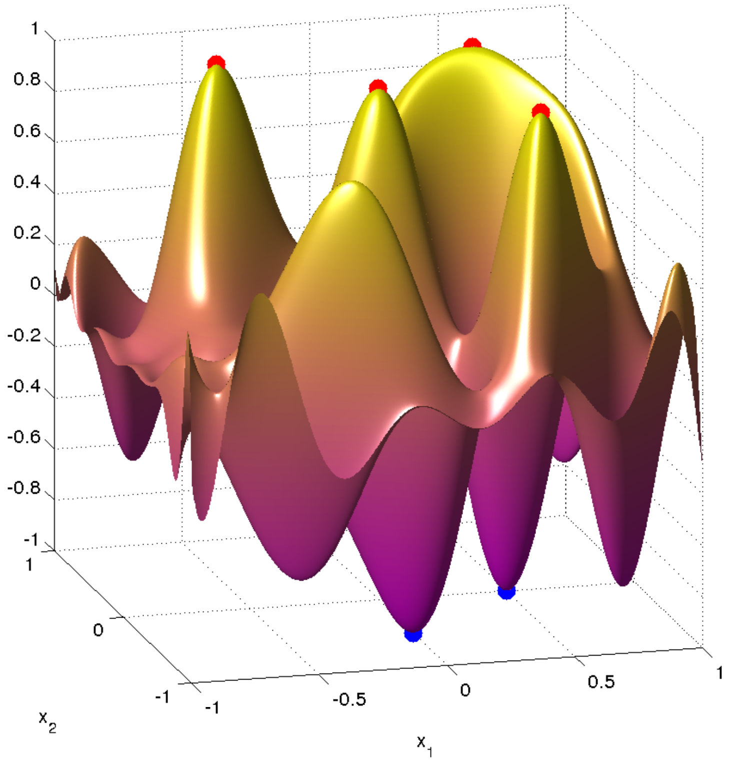

Figure 2: Degree 12 polynomial certificate for the bivariate example.

global optimality. Indeed we can check that the polynomial attains the value $+1$ at the 3 points $x \in \{(-1/2, 1/2), (1/2, -1/2), (0, 0)\}$, it attains the value -1 at the 2 points $x \in \{(0, -1/2), (1/2, 0)\}$ (in blue), while taking values between -1 and $+1$ on X.