Semidefinite Programming
L. Vandenberghe and S. Boyd

SIAM Review, 38(1): 49-95,
March 1996

Presented by Robin Hewitt
for CSE 252C, Fall 2004

Paper is available online at
http://www.stanford.edu/~boyd/sdp.html

Relevance

Combinatorial Optimization (e.g. Min Cut/Max Cut).

J. Keuchel, C. Schnörr, C. Schellewald, D. Cremers, Unsupervised Image Partitioning with Semidefinite Programming. Luc Van Gool (Ed.), *Pattern Recognition* (24th DAGM Symposium, Zurich), Lecture Notes in Computer Science, Vol. 2449, Springer, Berlin, 141-149, 2002.

Image Segmentation using SDP, 2002:
http://wwwcms.brookes.ac.uk/~philiptorr/

P.H.S. Torr. Solving Markov Random Fields using Semi Definite Programming. *Ninth International Workshop on Artificial Intelligence and Statistics*, 2003. http://wwwcms.brookes.ac.uk/~philiptorr

Relevance

J. Keuchel, C. Schellewald, D. Cremers, C. Schnörr, Convex Relaxations for Binary Image Partitioning and Perceptual Grouping. B. Radig, S. Florczyk (Eds.), *Pattern Recognition* (23rd DAGM Symposium, Munich), Lecture Notes in Computer Science, Vol. 2191, Springer, Berlin, 353-360, 2001.

K. Q. Weinberger and L. K. Saul (2004), Unsupervised learning of image manifolds by semidefinite programming. To appear in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (CVPR-04), Washington D.C.

C. Schellewald, C. Schnörr, Subgraph Matching with Semidefinite Programming. *Proceedings IWCIA* (International Workshop on Combinatorial Image Analysis), Palermo, Italy, May 14.-16.2003.

What is Semidefinite Programming?

- Optimization method that uses matrices as variables.
- Solvable in polynomial time.
- Can accommodate a wide range of problems – both real-valued and discrete.

- Least squares and Linear Programming are special cases of SDP.

SDP Problem Statement

minimize $c^{\mathrm{T}} x$
subject to $\mathbf{F}(x) \geq 0$
where

$$x, c \in \mathbf{R}^n$$

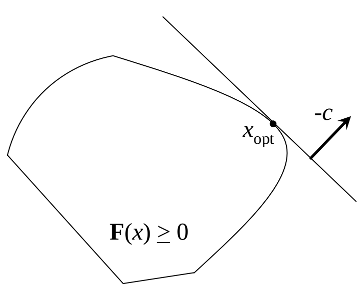
$$\mathbf{F}(x) \equiv \mathbf{F}_0 + \sum_{i=1}^{m} x_i \mathbf{F}_i$$

$\mathbf{F}(x)$ represents m+1 symmetric matrices. The inequality means that $\mathbf{F}(x)$ is positive semidefinite.

Geometric Interpretation

$\mathbf{F}(x) \geq 0$ represents the feasible region for x. So x is also a positive semidefinite matrix that's been vectorized.

The optimal value is the boundary point farthest in the $-c$ direction.



Geometry of an SDP

SDP Problem Statement Revisited

Vandenberghe-Boyd:

minimize $c^{\mathrm{T}} x$

subject to $\mathbf{F}(x) \geq 0$.

$x, c \in \mathbf{R}^n$

$$\mathbf{F}(x) \equiv \mathbf{F}_0 + \sum_{i=1}^{m} x_i \mathbf{F}_i$$

More commonly:

minimize $\mathbf{C} \bullet \mathbf{X}$

subject to $\mathbf{A}_i \bullet \mathbf{X} = \mathrm{b}_i$,

$i = 1, \ldots, m$

$\mathbf{X} \geq 0$.

$\mathbf{C} \bullet \mathbf{X}$ = inner product = $Tr(\mathbf{C}^{\mathrm{T}} \mathbf{X})$.

$\mathbf{C}$, $\mathbf{X}$, $\mathbf{A}_i$ square and symmetric.

Five-Minute Linear Algebra Review

Quadratic Forms:

Definition:

$$a_1 x_1^2 + a_2 x_2^2 + \ldots + a_n x_n^2 +$$
(all possible terms of the form $a_n x_i x_j$ for $i < j$).

Examples:

$$a_1 x_1^2 + a_2 x_2^2 + a_3 x_1 x_2$$

$$a_1 x_1^2 + a_2 x_2^2 + a_3 x_3^2 + a_4 x_1 x_2 + a_5 x_1 x_3 + a_6 x_2 x_3$$

Linear Algebra Review, cont.

Quadratic Forms, cont.

Example:

$$a_1x_1^2 + a_2x_2^2 + a_3x_1x_2$$

Matrix Representation:

$$\begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} a_1 & \frac{1}{2}a_3 \\ \frac{1}{2}a_3 & a_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

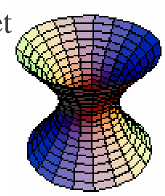
$x^T\mathbf{A}x = \mathbf{A}x \bullet x = x \bullet \mathbf{A}x$. $\mathbf{A}$ is symmetric.

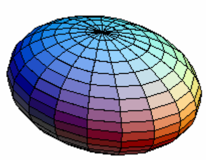
Linear Algebra Review, cont.

Graphs of quadratic equations are called quadratic surfaces.

Hyperboloid of one sheet

$$\frac{x^2}{l^2} + \frac{y^2}{m^2} - \frac{z^2}{n^2} = 1$$





Ellipsoid

$$\frac{x^2}{l^2} + \frac{y^2}{m^2} + \frac{z^2}{n^2} = 1$$

Linear Algebra Review, cont.

Positive Definite:

Definitions:

A quadratic form $x^{T}\mathbf{A}x$ is *positive definite* if $x^{T}\mathbf{A}x > 0$.

A symmetric matrix **A** is a *positive definite matrix* if $x^{T}\mathbf{A}x > 0$ is a positive definite quadratic form.

Theorem:

A symmetric matrix A is *positive definite* iff all its eigenvalues are positive.

Linear Algebra Review, cont.

Positive Definite:

Notation:

$\mathbf{A} > 0$ means **A** is *positive definite*.

$\mathbf{A} > \mathbf{B}$ means $\mathbf{A} - \mathbf{B} > 0$.

Positive Semidefinite:

For positive *semi*definite, replace > 0 by ≥ 0.

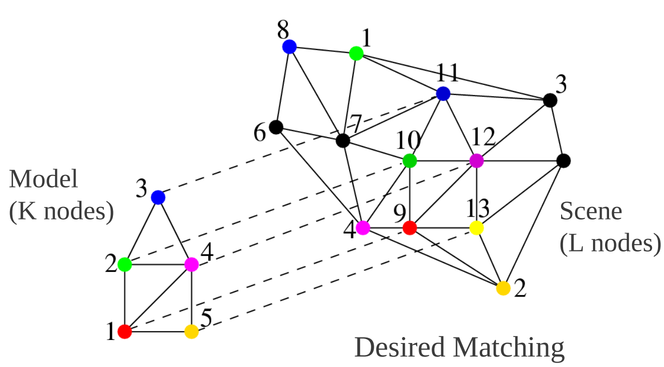
How To Set Up an SDP Problem

- Subgraph matching example.
- Class separation example.
- Handling nonlinearities.

Example – Subgraph Matching

Source: C. Schellewald and C. Schnörr
http://www.cvgpr.uni-mannheim.de/cschelle/graphmatching.html

The problem: find a model graph inside a scene graph.

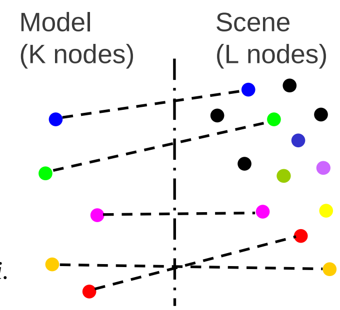


Subgraph Matching, cont.

Without SDP, we can minimize node-to-node correspondence error.

Min $d^T x$.
s.t. linear constraints*
where $x \in \{0,1\}^{KL}$, $d \in \mathbf{R}^{KL}$.

x – all possible bipartite pairs.
d – node-to-node distances for all x_i.
(a dissimilarity measure).



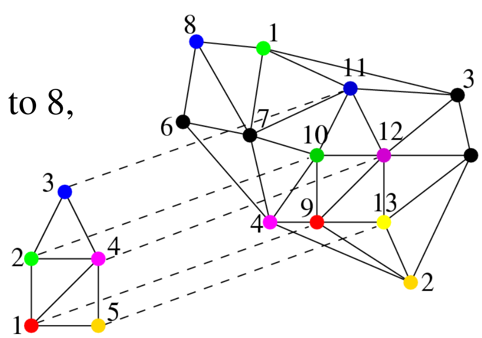
Bigraph of Matches

*The constraints enforce desired matching behavior. Example: each scene graph node should link to only one node in the model.

Subgraph Matching, cont.

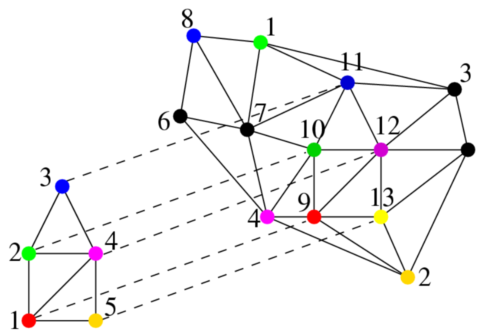
But, there's a problem with that.

Looking only at node similarity, we'd match 3 to 8, 5 to 2, etc.



Desired Matching

Example – Subgraph Matching

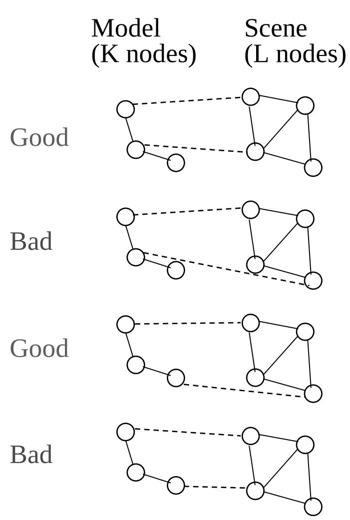


Desired Matching

We want a weighting that will favor node pairs with similar adjacencies in model and scene.

Solution: add a quadratic error term and model this as an SDP problem.

Subgraph Matching, cont.



Adjacency Error,
Q =

$$x^{\mathrm{T}}(\mathbf{A}_K \otimes \tilde{\mathbf{A}}_L)x = \sum_{ar}^{KL}\sum_{bs}^{KL}(\mathbf{A}_K)_{ab}(\tilde{\mathbf{A}}_L)_{rs}x_{ar}x_{bs}$$

$$+$$

$$x^{\mathrm{T}}(\tilde{\mathbf{A}}_K \otimes \mathbf{A}_L)x = \sum_{ar}^{KL}\sum_{bs}^{KL}(\tilde{\mathbf{A}}_K)_{ab}(\mathbf{A}_L)_{rs}x_{ar}x_{bs}$$

Subgraph Matching, cont.

Define a variable, **X**, s.t. it's positive semidefinite.

$$d^{\mathrm{T}} x + \mathrm{c} x^{\mathrm{T}} \mathbf{Q} x \rightarrow Tr\left[\begin{pmatrix} 0 & \frac{1}{2} d^{\mathrm{T}} \\ \frac{1}{2} d & \mathrm{c}\mathbf{Q} \end{pmatrix} \begin{pmatrix} 1 & x^{\mathrm{T}} \\ x & xx^{\mathrm{T}} \end{pmatrix}\right] \rightarrow Tr\left[\hat{\mathbf{Q}}\mathbf{X}\right]$$

x – all possible bipartite pairs.
d – node-to-node distances for all x_i.
Q – adjacency mismatch (quadratic) term, $\mathbf{Q} \in \{0,1\}^{KL,KL}$.
c – determines the importance of adjacency term.

Subgraph Matching, cont.

SDP Formulation:
minimize $\hat{\mathbf{Q}}^{\mathrm{T}} \bullet \mathbf{X}$
subject to $\mathbf{X} \geq 0$
(plus additional linear constraints.)

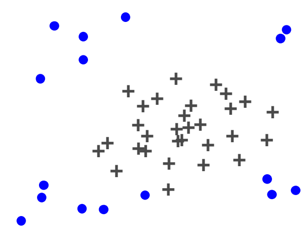
Example – Class Separation

Without SDP:

Separation with hyperplanes:

$a^T x_i + b \leq 0,\qquad i=1,\ldots,k,$

$a^T y_j + b \geq 0,\qquad j=1,\ldots,l.$



Example – Class Separation, cont.

With SDP:

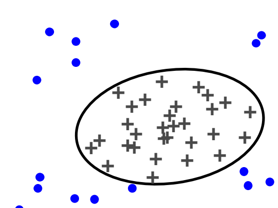
Separation with quadratic surfaces:

$(x_i)\mathbf{A}^T x_i + \mathbf{B}^T x_i + \mathbf{C} \leq 0,\qquad i=1,\ldots,k,$

$(y_j)\mathbf{A}^T y_j + \mathbf{B}^T y_j + \mathbf{C} \geq 0,\qquad j=1,\ldots,l.$

For an ellipsoid separator, add the constraint $\mathbf{A} > 0$.

To make the ellipsoid as spherical as possible, add an objective function to equalize eigenvalues.



Ellipsoid class separator

Tricks for Setting up an SDP Problem

$$\begin{array}{ll} \text{minimize} & \dfrac{(c^T x)^2}{d^T x} \\ \text{subject to} & \mathbf{A}x + \mathbf{B} \geq 0 \end{array}$$

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \mathbf{A}x + \mathbf{B} \geq 0 \\ & \dfrac{(c^T x)^2}{d^T x} \leq t \end{array}$$

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \begin{bmatrix} diag(\mathbf{A}x+\mathbf{B}) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & t & c^{\mathrm{T}}x \\ \mathbf{0} & c^{\mathrm{T}}x & d^{\mathrm{T}}x \end{bmatrix} \geq 0 \end{array}$$

- Slack variable
- Block diagonal
- Schur complement

Setting Up an SDP Problem, cont.

The convex constraints and data matrices may arise naturally from the application. This happened in the class separation example.

It may be necessary to devise an appropriate variable as in the subgraph matching example.

Both the objective function and the constraints must be linear. A nonlinear convex constraint may be expressible as a linear matrix inequality using some tricks.

Solving SDP Problems

- Use Newton's Method to approach x_{opt}.
- x_{opt} lies along a special path – the Central Path.
- Step direction and termination check utilize the *dual* SDP.

The Dual SDP

Primal SDP:

minimize $c^T x$
subject to $\mathbf{F}(x) \geq 0$.

$$x, c \in \mathbf{R}^n$$
$$\mathbf{F}(x) \equiv \mathbf{F}_0 + \sum_{i=1}^{m} x_i \mathbf{F}_i$$

Dual SDP:

maximize $-Tr\mathbf{F}_0\mathbf{Z}$
subject to $Tr\mathbf{F}_i\mathbf{Z} = c_i$, $i = 1,\ldots,m$
$\mathbf{Z} \geq 0$.

The dual SDP is also an SDP. The original SDP is called the primal SDP.

The Duality Gap

The dual SDP gives us bounds for the optimal value of the primal SDP (and vice versa). Assume both **Z** and *x* are feasible. Then

$$c^{\mathrm{T}} x + Tr\mathbf{F}_0\mathbf{Z} = \sum_{i=1}^{m} Tr\mathbf{F}_i\mathbf{Z}x_i + Tr\mathbf{F}(x)\mathbf{Z} \geq 0.$$

Also, $Tr\mathbf{F}_0\mathbf{Z} \geq 0$ since $\mathbf{F}_0 = \mathbf{F}_0^{\mathrm{T}} \geq 0$ and $\mathbf{Z} = \mathbf{Z}^T \geq 0$.
So $-Tr\mathbf{F}_0\mathbf{Z} \leq c^{\mathrm{T}}x$.

Thus, in the dual SDP, the objective value for any feasible **Z** is smaller than or equal to the objective value of any feasible *x* in the primal SDP.

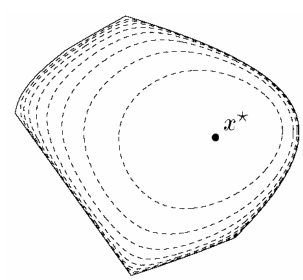
This difference is the duality gap, η.

$$\eta \equiv c^{\mathrm{T}}x + Tr\mathbf{F}_0\mathbf{Z} = Tr\mathbf{F}(x)\mathbf{Z}.$$

Barrier Function and Analytic Center

Define a barrier function φ(x) s.t. φ(x) is finite iff $\mathbf{F}(x) > 0$ and becomes infinite as x approaches the boundary of the feasible region. Such a function is

$$\Phi(x) \equiv \begin{cases} \log\det \mathbf{F}(x)^{-1} & \text{if } \mathbf{F}(x) > 0 \\ +\infty & \text{otherwise} \end{cases}$$



Contour lines for the barrier function

The point where φ(x) is minimized is called the Analytic Center, x^*. This is *not* a geometric center. It depends only on $\mathbf{F}(x)$.

Central Path

Given the inequality

$c^T x = \gamma$

$\mathbf{F}(x) > 0$

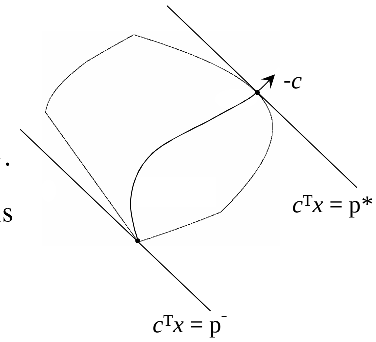
where $p^* < \gamma < p^- \equiv \sup\{c^T x \mid \mathbf{F}(x) > 0\}$.

The analytic center of this new system is

$$x^*(\gamma) \equiv \operatorname{argmin} \quad \log\det \mathbf{F}(x)^{-1}$$
$$\text{subject to} \quad \mathbf{F}(x) > 0$$
$$c^T x = \gamma$$

As γ approaches p^* from above, $x^*(\gamma)$ converges to the optimal point.

The curve $x^*(\gamma)$ is called the *central path* for the SDP.



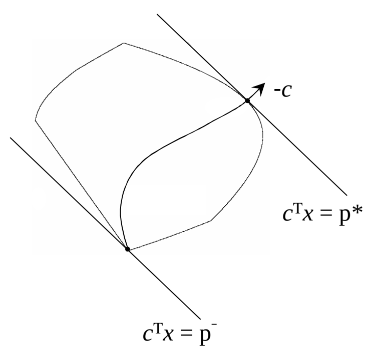
Central Path

Central Path, cont.

Points on the central path yield dual-feasible matrices of the form $\mathbf{Z} = \mathbf{F}(x^*(\gamma))^{-1}\mathbf{F}_i = \lambda c_i$, $i = 1,\ldots m$, where λ is a Lagrange multiplier.

The duality gap along the central path is $\eta = Tr\mathbf{F}(x^*(\gamma))\, \mathbf{F}(x^*(\gamma))^{-1}/\lambda = n/\lambda$.

Thus the central path can be parameterized by the duality gap.



Central Path

A Possible Algorithm

1. Define a function g(x,**Z**) that combines the duality gap with the deviation from centrality. Pick g(x,**Z**) s.t. g is small when the duality gap is small.

2. Pick feasible initial points $x^{(0)}$ and $\mathbf{Z}^{(0)}$.

3. Take Newton steps that reduce g by a fixed amount each step.

4. Evaluate the duality gap at each step and terminate when this is sufficiently small.

A Possible Algorithm, cont.

Each Newton step involves minimizing g over a plane defined by the current x and **Z** and the current search directions, δx and $\delta\mathbf{Z}$.

δx and $\delta\mathbf{Z}$ are obtained by solving a set of linear equations. Most of the computational time is due to this step.

Once we've selected the search directions, δx and $\delta\mathbf{Z}$, we can draw a rectangle s.t. g within the rectangle has a unique local minimum that is also the global minimum. Also, the duality gap lies at one of the corners of this rectangle. We can use this as the termination criterion.

Performance

Convergence time is polynomial.

Actual performance depends on the initial choice for $x^{(0)}$ and $\mathbf{Z}^{(0)}$. Worst case complexity for convergence is $O(n^{1/2})$.

In practice, $O(\log n)$ or $O(n^{1/4})$ is typical.

Since most of the computation time goes to solving a set of linear equations, it may be worthwhile to exploit the problem structure to optimize this step. For example, sparse matrices or block-diagonal structure can be exploited to improve performance.

Conclusion

A wide class of convex optimization problems can be solved with SDP.

It's not much harder to solve an SDP problem than it is to solve an LP problem.

References and Resources

For Learning About SDP:

C. Helmberg. Semidefinite Programming for Combinatorial Optimization. *Habilitationsschrift*, TU Berlin, January 2000. ZIB-Report ZR-00-34, Konrad-Zuse-Zentrum Berlin, October 2000.
http://www.zib.de/PaperWeb/abstracts/ZR-00-34/

SDP References (to reference in a paper):

L. Vandenberghe and S. Boyd. Semidefinite programming. *SIAM Review*, 38(1):49-95, 1996.

H. Wolkowicz, R. Saigal and L. Vandenberghe, *Handbook on Semidefnite Programming*, Kluwer Academic Publishers, 2000.

References and Resources, cont.

Subgraph Matching:

C. Schellewald, C. Schnörr. Subgraph Matching with Semidefinite Programming. *Proceedings IWCIA* (International Workshop on Combinatorial Image Analysis), Palermo, Italy, May 14.-16.2003
http://www.cvgpr.uni-mannheim.de/cschelle/graphmatching.html

SDP Solvers:

The SDP website has links to software reviews:
http://www-user.tu-chemnitz.de/~helmberg/semidef.html

Matlab links to compatible optimization resources:
http://www.mathtools.net/MATLAB/Optimization/index.html

Extra Slides

Kronecker Product

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} \mathrm{a}_{11}\mathbf{B} & \cdots & \mathrm{a}_{1n}\mathbf{B} \\ \vdots & \ddots & \vdots \\ \mathrm{a}_{n1}\mathbf{B} & \cdots & \mathrm{a}_{nn}\mathbf{B} \end{bmatrix}.$$

If A is m×n and B is p×q, their Kronecker product is size (mp)×(nq).

Schur Complement

In a partitioned matrix, $\mathbf{M} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}$,

$S \equiv \mathbf{A} - \mathbf{B}\mathbf{D}^{-1}\mathbf{C}$ is the Schur complement of **D**.

If **M** is positive definite, S is also positive definite.