Chapter 4

Semidefinite programming

> *Prior to 1984, linear and nonlinear programming,*[4.1] *one a subset of the other, had evolved for the most part along unconnected paths, without even a common terminology.* (*The use of "programming" to mean "optimization" serves as a persistent reminder of these differences.*)
>
> −Forsgren, Gill, & Wright, 2002 [158]

Given some practical application of convex analysis, it may at first seem puzzling why a search for its solution ends abruptly with a formalized statement of the problem itself as a constrained optimization. The explanation is: typically we do not seek analytical solution because there are relatively few. (§3.5.2, §C) If a problem can be expressed in convex form, rather, then there exist computer programs providing efficient numerical global solution. [183] [423] [424] [422] [367] [353] The goal, then, becomes conversion of a given problem (perhaps a nonconvex or combinatorial problem statement) to an equivalent convex form or to an *alternation* of convex subproblems convergent to a solution of the original problem:

By the *fundamental theorem of Convex Optimization,* any locally optimal point (solution) of a convex problem is globally optimal. [63, §4.2.2] [324, §1] Given convex real objective function g and convex feasible set $\mathcal{D}\subseteq \operatorname{dom} g$, which is the set of all variable values satisfying the problem constraints, we pose a generic convex optimization problem

$$\begin{array}{ll} \underset{X}{\text{minimize}} & g(X) \\ \text{subject to} & X\in\mathcal{D} \end{array} \tag{685}$$

[4.1] nascence of polynomial-time *interior-point methods* of solution [382] [420].
Linear programming $\subset$ (convex $\cap$ nonlinear) programming.

where constraints are abstract here in membership of variable X to convex feasible set $\mathcal{D}$. Inequality constraint functions of a convex optimization problem are convex while equality constraint functions are conventionally affine, but not necessarily so. Affine equality constraint functions, as opposed to the superset of all convex equality constraint functions having convex level sets (§3.4.0.0.4), make convex optimization tractable.

Similarly, the problem

$$\begin{array}{ll} \underset{X}{\text{maximize}} & g(X) \\ \text{subject to} & X \in \mathcal{D} \end{array} \tag{686}$$

is called *convex* were g a real concave function and feasible set $\mathcal{D}$ convex. As conversion to convex form is not always possible, there is much ongoing research to determine which problem classes have convex expression or relaxation. [35] [61] [165] [294] [363] [162]

4.1 Conic problem

> *Still, we are surprised to see the relatively small number of submissions to semidefinite programming (SDP) solvers, as this is an area of significant current interest to the optimization community. We speculate that semidefinite programming is simply experiencing the fate of most new areas: Users have yet to understand how to pose their problems as semidefinite programs, and the lack of support for SDP solvers in popular modelling languages likely discourages submissions.*
>
> −*SIAM News*, 2002. [126, p.9]

(*confer* p.140) Consider a *conic problem* (p) and its dual (d): [311, §3.3.1] [255, §2.1] [256]

$$\text{(p)} \quad \begin{array}{ll} \underset{x}{\text{minimize}} & c^{\mathrm{T}}x \\ \text{subject to} & x \in \mathcal{K} \\ & Ax = b \end{array} \qquad \begin{array}{ll} \underset{y,s}{\text{maximize}} & b^{\mathrm{T}}y \\ \text{subject to} & s \in \mathcal{K}^* \\ & A^{\mathrm{T}}y + s = c \end{array} \quad \text{(d)} \tag{301}$$

where $\mathcal{K}$ is a closed convex cone, $\mathcal{K}^*$ is its dual, matrix A is fixed, and the remaining quantities are vectors.

When $\mathcal{K}$ is a polyhedral cone (§2.12.1), then each conic problem becomes a *linear program*; the selfdual nonnegative orthant providing the prototypical primal linear program and its dual. [98, §3-1][4.2] More generally, each optimization problem is convex when $\mathcal{K}$ is a closed convex cone. Solution to each convex problem is not necessarily unique; the optimal solution sets $\{x^\star\}$ and $\{y^\star, s^\star\}$ are convex and may comprise more than a single point.

[4.2] Dantzig explains reasoning behind a nonnegativity constraint: ... *negative quantities of activities are not possible.* ... *a negative number of cases cannot be shipped.*

4.1.1 a semidefinite program

When $\mathcal{K}$ is the selfdual cone of positive semidefinite matrices $\mathbb{S}^n_+$ in the subspace of symmetric matrices $\mathbb{S}^n$, then each conic problem is called a *semidefinite program* (SDP); [294, §6.4] primal problem (P) having matrix variable $X \in \mathbb{S}^n$ while corresponding dual (D) has *slack variable* $S \in \mathbb{S}^n$ and vector variable $y = [y_i] \in \mathbb{R}^m$: [11] [12, §2] [430, §1.3.8]

$$\begin{array}{lll} & \underset{X\in\mathbb{S}^n}{\text{minimize}} & \langle C\,,\,X\rangle \\ \text{(P)} & \text{subject to} & X \succeq 0 \\ & & A\,\operatorname{svec} X = b \end{array} \qquad \begin{array}{lll} & \underset{y\in\mathbb{R}^m,\ S\in\mathbb{S}^n}{\text{maximize}} & \langle b\,,\,y\rangle \\ \text{(D)} & \text{subject to} & S \succeq 0 \\ & & \operatorname{svec}^{-1}(A^{\mathrm{T}}y) + S = C \end{array} \tag{687}$$

This is the *prototypical semidefinite program* and its dual, where matrix $C \in \mathbb{S}^n$ and vector $b \in \mathbb{R}^m$ are fixed as is

$$A \triangleq \begin{bmatrix} \operatorname{svec}(A_1)^{\mathrm{T}} \\ \vdots \\ \operatorname{svec}(A_m)^{\mathrm{T}} \end{bmatrix} \in \mathbb{R}^{m\times n(n+1)/2} \tag{688}$$

because $\{A_i \in \mathbb{S}^n,\ i = 1 \dots m\}$ is given. Thus

$$\begin{aligned} A\,\operatorname{svec} X &= \begin{bmatrix} \langle A_1\,,\,X\rangle \\ \vdots \\ \langle A_m\,,\,X\rangle \end{bmatrix} \\ \operatorname{svec}^{-1}(A^{\mathrm{T}}y) &= \sum_{i=1}^{m} y_i A_i \end{aligned} \tag{689}$$

The vector inner-product for matrices is defined in the Euclidean/Frobenius sense in the isomorphic vector space $\mathbb{R}^{n(n+1)/2}$; *id est*,

$$\langle C\,,\,X\rangle \triangleq \operatorname{tr}(C^{\mathrm{T}}X) = \operatorname{svec}(C)^{\mathrm{T}}\operatorname{svec} X \tag{38}$$

where $\operatorname{svec} X$ defined by (56) denotes symmetric vectorization.

> *In a national planning problem of some size, one may easily run into several hundred variables and perhaps a hundred or more degrees of freedom. ... It should always be remembered that any mathematical method and particularly methods in linear programming must be judged with reference to the type of computing machinery available. Our outlook may perhaps be changed when we get used to the super modern, high capacity electronic computor that will be available here from the middle of next year.* −Ragnar Frisch [160]

The Simplex method of solution for linear programming, invented by Dantzig in 1947 [98], is now integral to modern technology. The same cannot yet be said for semidefinite programming whose roots trace back to systems of positive semidefinite linear inequalities studied by Bellman & Fan in 1963 [32] [108] who provided saddle convergence criteria. Interior-point methods for numerical solution of linear programs can be traced back to the logarithmic barrier of Frisch in 1954 and Fiacco & McCormick in 1968 [153]. Karmarkar's polynomial-time interior-point method sparked a log-barrier renaissance

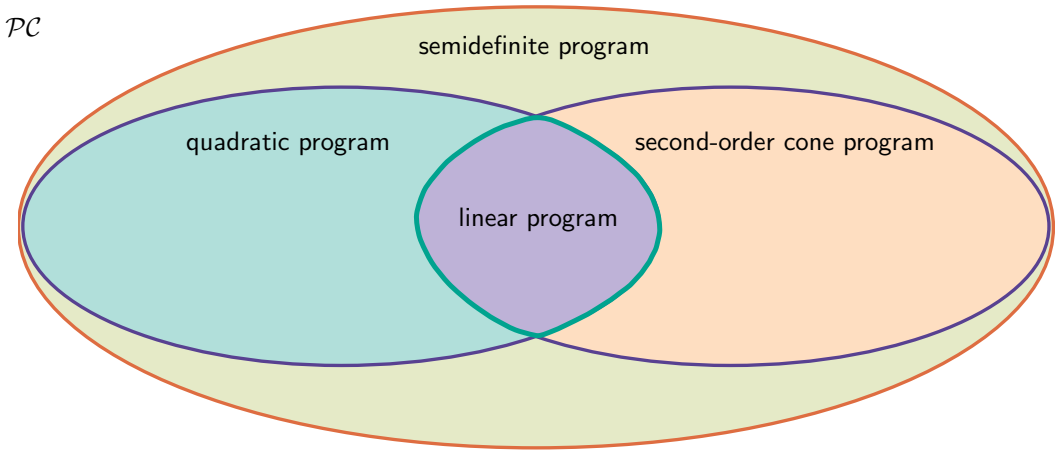


Figure 86: Venn diagram of programming hierarchy. Semidefinite program is a subset of convex program $\mathcal{PC}$. Semidefinite program subsumes other convex program classes excepting geometric program. Second-order cone program and quadratic program each subsume linear program. Nonconvex program $\backslash\mathcal{PC}$ comprises those for which convex equivalents have not yet been found.

in 1984, [291, §11] [420] [382] [294, p.3] but numerical performance of contemporary general-purpose semidefinite program solvers remains limited: Computational intensity for dense systems varies as $\mathrm{O}(m^2n)$ (m constraints $\ll n$ variables) based on interior-point methods that produce solutions no more relatively accurate than `1E-8`. There are no solvers capable of handling in excess of n=100,000 variables without significant, sometimes crippling, loss of precision or time.[4.3] [36] [293, p.258] [70, p.3]

Nevertheless, semidefinite programming has recently emerged to prominence because it admits a new class of problem previously unsolvable by convex optimization techniques, [61] and because it theoretically subsumes other convex techniques: (Figure **86**) linear programming and *quadratic programming* and *second-order cone programming*.[4.4] Determination of the Riemann mapping function from complex analysis [303] [30, §8, §13], for example, can be posed as a semidefinite program.

4.1.2 Maximal complementarity

It has been shown [430, §2.5.3] that contemporary interior-point methods [421] [306] [294] [12] [63, §11] [158] (developed *circa* 1990 [165] for numerical solution of semidefinite

[4.3] *Heuristics are not ruled out by SIOPT; indeed I would suspect that most successful methods have (appropriately described) heuristics under the hood - my codes certainly do. ... Of course, there are still questions relating to high-accuracy and speed, but for many applications a few digits of accuracy suffices and overnight runs for non-real-time delivery is acceptable.*
−Nicholas I. M. Gould, Stanford alumnus, SIOPT Editor in Chief

[4.4] Second-order cone programming was born in the 1990s; it is not posable as a quadratic program. [264]

programs) can converge to a solution of *maximal complementarity*; [192, §5] [429] [269] [172] not a vertex solution but a solution of highest cardinality or rank among all optimal solutions.[4.5]

This phenomenon can be explained by recognizing that interior-point methods generally find solutions relatively interior to a feasible set by design.[4.6] [7, p.3] Log barriers are designed to fail numerically at the feasible set boundary. So low-rank solutions, all on the boundary, are rendered more difficult to find as numerical error becomes more prevalent there.

4.1.2.1 Reduced-rank solution

A simple rank reduction algorithm, for construction of a primal optimal solution $X^\star$ to (687P) satisfying an upper bound on rank governed by Proposition 2.9.3.0.1, is presented in §4.3. That proposition asserts existence of feasible solutions with an upper bound on their rank; [27, §II.13.1] specifically, it asserts an extreme point (§2.6.0.0.1) of *primal feasible set* $\mathcal{A} \cap \mathbb{S}^n_+$ satisfies upper bound

$$\operatorname{rank} X \leq \left\lfloor \frac{\sqrt{8m+1}-1}{2} \right\rfloor \tag{272}$$

where, given $A \in \mathbb{R}^{m\times n(n+1)/2}$ and $b \in \mathbb{R}^m$,

$$\mathcal{A} \triangleq \{X \in \mathbb{S}^n \mid A \operatorname{svec} X = b\} \tag{690}$$

is the affine subset from primal problem (687P).

4.1.2.2 Coexistence of low- and high-rank solutions; analogy

That low-rank and high-rank optimal solutions $\{X^\star\}$ of (687P) coexist may be grasped with the following analogy: We compare a proper polyhedral cone $\mathcal{S}^{\mathbf{3}}_+$ in $\mathbb{R}^{\mathbf{3}}$ (illustrated in Figure 87) to the positive semidefinite cone $\mathbb{S}^{\mathbf{3}}_+$ in isometrically isomorphic $\mathbb{R}^{\mathbf{6}}$, difficult to visualize. The analogy is good:

- $\operatorname{int}\mathbb{S}^{\mathbf{3}}_+$ is constituted by rank-3 matrices.
  $\operatorname{int}\mathcal{S}^{\mathbf{3}}_+$ has three dimensions.
- boundary $\partial\mathbb{S}^{\mathbf{3}}_+$ contains rank-0, rank-1, and rank-2 matrices.
  boundary $\partial\mathcal{S}^{\mathbf{3}}_+$ contains 0-, 1-, and 2-dimensional faces.
- the only rank-0 matrix resides in the vertex at the origin.
- Rank-1 matrices are in one-to-one correspondence with extreme directions of $\mathbb{S}^{\mathbf{3}}_+$ and $\mathcal{S}^{\mathbf{3}}_+$. The set of all rank-1 symmetric matrices in this dimension

  $$\left\{G \in \mathbb{S}^{\mathbf{3}}_+ \mid \operatorname{rank} G = 1\right\} \tag{691}$$

  is not a connected set.

[4.5] This characteristic might be regarded as a disadvantage to interior-point methods of numerical solution, but this behavior is not certain and depends on solver implementation.

[4.6] Simplex methods, in contrast, find vertex solutions. [98, p.158] [16, p.2]

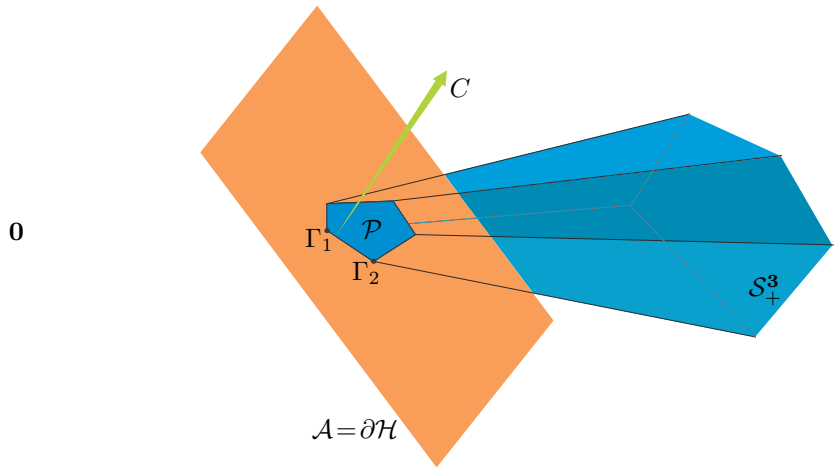


Figure 87: Visualizing positive semidefinite cone in high dimension: Proper polyhedral cone $\mathcal{S}_+^{\mathbf{3}} \subset \mathbb{R}^{\mathbf{3}}$ representing positive semidefinite cone $\mathbb{S}_+^{\mathbf{3}} \subset \mathbb{S}^{\mathbf{3}}$; analogizing its intersection $\mathbb{S}_+^{\mathbf{3}} \cap \partial\mathcal{H}$ with hyperplane. Number of facets is arbitrary (an analogy not inspired by eigenvalue decomposition). The rank-0 positive semidefinite matrix corresponds to origin in $\mathbb{R}^{\mathbf{3}}$, rank-1 positive semidefinite matrices correspond to edges of polyhedral cone, rank-2 to facet relative interiors, and rank-3 to polyhedral cone interior. Vertices Γ_1 and Γ_2 are extreme points of polyhedron $\mathcal{P} = \partial\mathcal{H} \cap \mathcal{S}_+^{\mathbf{3}}$, and extreme directions of $\mathcal{S}_+^{\mathbf{3}}$. A given vector C is normal to another hyperplane (not illustrated but independent w.r.t $\partial\mathcal{H}$) containing line segment $\overline{\Gamma_1\Gamma_2}$ minimizing real linear function $\langle C\,,\, X\rangle$ on $\mathcal{P}$. (*confer* Figure 29, Figure 33)

- Rank of a sum of members $F+G$ in Lemma 2.9.2.9.1 and location of a difference $F-G$ in §2.9.2.12.1 similarly hold for $\mathbb{S}^{\mathbf{3}}_{+}$ and $\mathcal{S}^{\mathbf{3}}_{+}$.

- Euclidean distance from any particular rank-3 positive semidefinite matrix (in the cone interior) to the closest rank-2 positive semidefinite matrix (on the boundary) is generally less than the distance to the closest rank-1 positive semidefinite matrix. (§7.1.2)

- distance from any point in $\partial\mathbb{S}^{\mathbf{3}}_{+}$ to $\operatorname{int}\mathbb{S}^{\mathbf{3}}_{+}$ is infinitesimal (§2.1.7.1.1).
  distance from any point in $\partial\mathcal{S}^{\mathbf{3}}_{+}$ to $\operatorname{int}\mathcal{S}^{\mathbf{3}}_{+}$ is infinitesimal.

- faces of $\mathbb{S}^{\mathbf{3}}_{+}$ correspond to faces of $\mathcal{S}^{\mathbf{3}}_{+}$ (*confer* Table **2.9.2.3.1**):

	k	$\dim\mathcal{F}(\mathcal{S}^{\mathbf{3}}_{+})$	$\dim\mathcal{F}(\mathbb{S}^{\mathbf{3}}_{+})$	$\dim\mathcal{F}(\mathbb{S}^{\mathbf{3}}_{+}\ni\text{rank-}k\text{ matrix})$
boundary	0	0	0	0
	1	1	1	1
	2	2	3	3
interior	3	3	6	6

  Integer k indexes k-dimensional faces $\mathcal{F}$ of $\mathcal{S}^{\mathbf{3}}_{+}$. Positive semidefinite cone $\mathbb{S}^{\mathbf{3}}_{+}$ has four kinds of faces, including cone itself $(k=3,\ \text{boundary} + \text{interior})$, whose dimensions in isometrically isomorphic $\mathbb{R}^{\mathbf{6}}$ are listed under $\dim\mathcal{F}(\mathbb{S}^{\mathbf{3}}_{+})$. Smallest face $\mathcal{F}(\mathbb{S}^{\mathbf{3}}_{+}\ni\text{rank-}k\text{ matrix})$ that contains a rank-k positive semidefinite matrix has dimension $k(k+1)/2$ by (222).

- For $\mathcal{A}$ equal to intersection of m hyperplanes having linearly independent normals, and for $X\in\mathcal{S}^{\mathbf{3}}_{+}\cap\mathcal{A}$, we have $\operatorname{rank}X\leq m$; the analogue to (272).

  Proof. With reference to Figure **87**: Assume one $(m=1)$ hyperplane $\mathcal{A}=\partial\mathcal{H}$ intersects the polyhedral cone. Every intersecting plane contains at least one matrix having rank less than or equal to 1; *id est*, from all $X\in\partial\mathcal{H}\cap\mathcal{S}^{\mathbf{3}}_{+}$ there exists an X such that $\operatorname{rank}X\leq 1$. Rank 1 is therefore an upper bound in this case.

  Now visualize intersection of the polyhedral cone with two $(m=2)$ hyperplanes having linearly independent normals. The hyperplane intersection $\mathcal{A}$ makes a line. Every intersecting line contains at least one matrix having rank less than or equal to 2, providing an upper bound. In other words, there exists a positive semidefinite matrix X belonging to any line intersecting the polyhedral cone such that $\operatorname{rank}X\leq 2$.

  In the case of three independent intersecting hyperplanes $(m=3)$, the hyperplane intersection $\mathcal{A}$ makes a point that can reside anywhere in the polyhedral cone. The upper bound on a point in $\mathcal{S}^{\mathbf{3}}_{+}$ is also the greatest upper bound: $\operatorname{rank}X\leq 3$. ♦

4.1.2.2.1 Example. *Optimization over* $\mathcal{A} \cap \mathcal{S}^{\mathbf{3}}_{+}$.
Consider minimization of the real linear function $\langle C\,,\,X\rangle$ over

$$\mathcal{P} \triangleq \mathcal{A} \cap \mathcal{S}^{\mathbf{3}}_{+} \tag{692}$$

a polyhedral feasible set;

$$\begin{array}{rll} f_0^\star \triangleq & \underset{X}{\text{minimize}} & \langle C\,,\,X\rangle \\ & \text{subject to} & X \in \mathcal{A} \cap \mathcal{S}^{\mathbf{3}}_{+} \end{array} \tag{693}$$

As illustrated for particular vector C and hyperplane $\mathcal{A} = \partial\mathcal{H}$ in Figure 87, this linear function is minimized on any X belonging to the face of $\mathcal{P}$ containing extreme points $\{\Gamma_1\,,\,\Gamma_2\}$ and all the rank-2 matrices in between; *id est*, on any X belonging to the face of $\mathcal{P}$

$$\mathcal{F}(\mathcal{P}) = \{X \mid \langle C\,,\,X\rangle = f_0^\star\} \cap \mathcal{A} \cap \mathcal{S}^{\mathbf{3}}_{+} \tag{694}$$

exposed by the hyperplane $\{X \mid \langle C\,,\,X\rangle = f_0^\star\}$. In other words, the set of all optimal points $X^\star$ is a face of $\mathcal{P}$

$$\{X^\star\} \;=\; \mathcal{F}(\mathcal{P}) \;=\; \overline{\Gamma_1\Gamma_2} \tag{695}$$

comprising rank-1 and rank-2 positive semidefinite matrices. Rank 1 is the upper bound on existence in the feasible set $\mathcal{P}$ for this case $m = 1$ hyperplane constituting $\mathcal{A}$. The rank-1 matrices Γ_1 and Γ_2 in face $\mathcal{F}(\mathcal{P})$ are extreme points of that face and (by transitivity (§2.6.1.2)) extreme points of the intersection $\mathcal{P}$ as well. As predicted by analogy to Barvinok's Proposition 2.9.3.0.1, the upper bound on rank of X existent in the feasible set $\mathcal{P}$ is satisfied by an extreme point. The upper bound on rank of an optimal solution $X^\star$ existent in $\mathcal{F}(\mathcal{P})$ is thereby also satisfied by an extreme point of $\mathcal{P}$ precisely because $\{X^\star\}$ constitutes $\mathcal{F}(\mathcal{P})$;[4.7] in particular,

$$\{X^\star \in \mathcal{P} \mid \operatorname{rank} X^\star \leq 1\} \;=\; \{\Gamma_1\,,\,\Gamma_2\} \;\subseteq\; \mathcal{F}(\mathcal{P}) \tag{696}$$

As all linear functions on a polyhedron are minimized on a face, [98] [268] [290] [297] by analogy we so demonstrate coexistence of optimal solutions $X^\star$ of (687P) having assorted rank. □

4.1.2.3 Previous work

Barvinok showed, [25, §2.2] when given a positive definite matrix C and an arbitrarily small neighborhood of C comprising positive definite matrices, there exists a matrix $\tilde{C}$ from that neighborhood such that optimal solution $X^\star$ to (687P) (substituting $\tilde{C}$) is an extreme point of $\mathcal{A} \cap \mathbb{S}^n_+$ and satisfies upper bound (272).[4.8] Given arbitrary positive definite C, this means nothing inherently guarantees that an optimal solution $X^\star$ to problem (687P) satisfies (272); certainly nothing given any symmetric matrix C, as the problem is posed. This can be proved by example:

[4.7] and every face contains a subset of the extreme points of $\mathcal{P}$ by the *extreme existence theorem* (§2.6.0.0.2). This means: because the affine subset $\mathcal{A}$ and hyperplane $\{X \mid \langle C\,,\,X\rangle = f_0^\star\}$ must intersect a whole face of $\mathcal{P}$, calculation of an upper bound on rank of $X^\star$ ignores counting the hyperplane when determining m in (272).

[4.8] Further, the set of all such $\tilde{C}$ in that neighborhood is open and dense.

4.1.2.3.1 Example. (Ye) *Maximal Complementarity.*
Assume dimension n to be an even positive number. Then the particular instance of problem (687P),

$$\begin{array}{ll}\underset{X\in\mathbb{S}^n}{\text{minimize}} & \left\langle \begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & 2I \end{bmatrix}, X \right\rangle \\ \text{subject to} & X \succeq 0 \\ & \langle I\,,\, X\rangle = n \end{array} \tag{697}$$

has optimal solution

$$X^\star = \begin{bmatrix} 2I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \in \mathbb{S}^n \tag{698}$$

with an equal number of twos and zeros along the main diagonal. Indeed, optimal solution (698) is a terminal solution along the *central path* taken by the interior-point method as implemented in [430, §2.5.3]; it is also a solution of highest rank among all optimal solutions to (697). Clearly, rank of this primal optimal solution exceeds by far a rank-1 solution predicted by upper bound (272). □

4.1.2.4 Later developments

This rational example (697) indicates the need for a more generally applicable and simple algorithm to identify an optimal solution $X^\star$ satisfying Barvinok's Proposition 2.9.3.0.1. We will review such an algorithm in §4.3, but first we provide more background.

4.2 Framework

4.2.1 Feasible sets

Denote by $\mathcal{D}$ and $\mathcal{D}^*$ the convex sets of primal and dual points respectively satisfying the primal and dual constraints in (687), each assumed nonempty;

$$\begin{aligned} \mathcal{D} &= \left\{ X\in\mathbb{S}^n_+ \mid \begin{bmatrix} \langle A_1\,,\, X\rangle \\ \vdots \\ \langle A_m\,,\, X\rangle \end{bmatrix} = b \right\} = \mathcal{A}\cap\mathbb{S}^n_+ \\ \mathcal{D}^* &= \left\{ S\in\mathbb{S}^n_+\,,\; y=[y_i]\in\mathbb{R}^m \mid \sum_{i=1}^m y_iA_i + S = C \right\} \end{aligned} \tag{699}$$

These are the *primal feasible set* and *dual feasible set*. Geometrically, primal feasible $\mathcal{A}\cap\mathbb{S}^n_+$ represents an intersection of the positive semidefinite cone $\mathbb{S}^n_+$ with an affine subset $\mathcal{A}$ of the subspace of symmetric matrices $\mathbb{S}^n$ in isometrically isomorphic $\mathbb{R}^{n(n+1)/2}$. $\mathcal{A}$ has dimension $n(n+1)/2-m$ when the vectorized A_i are linearly independent. Dual feasible set $\mathcal{D}^*$ is a Cartesian product of the positive semidefinite cone with its inverse image (§2.1.9.0.1) under affine transformation[4.9] $C-\sum y_iA_i$. Both feasible sets are convex, and

[4.9] Inequality $C-\sum y_iA_i\succeq 0$ follows directly from (687D) (§2.9.0.1.1) and is known as a *linear matrix inequality*. (§2.13.5.1.1) Because $\sum y_iA_i\preceq C$, matrix S is known as a *slack variable* (a term borrowed from linear programming [98]) since its inclusion raises this inequality to equality.

the objective functions are linear on a Euclidean vector space. Hence, (687P) and (687D) are convex optimization problems.

4.2.1.1 $\mathcal{A} \cap \mathbb{S}^n_+$ emptiness determination via Farkas' lemma

4.2.1.1.1 Lemma. *Semidefinite Farkas' lemma.*
Given set $\{A_i \in \mathbb{S}^n,\ i=1\dots m\}$, vector $b=[b_i]\in\mathbb{R}^m$, and affine subset

$$\text{(690)}\quad \mathcal{A} = \{X\in\mathbb{S}^n \,|\, \langle A_i\,,\, X\rangle = b_i\,,\ i=1\dots m\} \ \ni\ \{A\ \operatorname{svec} X \,|\, X\succeq 0\}\ \text{(379) is closed,}$$

then primal feasible set $\mathcal{A}\cap\mathbb{S}^n_+$ is nonempty if and only if $y^{\mathrm{T}}b\geq 0$ holds for each and every vector $y=[y_i]\in\mathbb{R}^m$ such that $\sum_{i=1}^m y_iA_i\succeq 0$.

Equivalently, primal feasible set $\mathcal{A}\cap\mathbb{S}^n_+$ is nonempty if and only if $y^{\mathrm{T}}b\geq 0$ holds for each and every vector $\|y\|=1$ such that $\sum_{i=1}^m y_iA_i\succeq 0$. ⋄

Semidefinite Farkas' lemma provides necessary and sufficient conditions for a set of hyperplanes to have nonempty intersection $\mathcal{A}\cap\mathbb{S}^n_+$ with the positive semidefinite cone. Given

$$A = \begin{bmatrix} \operatorname{svec}(A_1)^{\mathrm{T}} \\ \vdots \\ \operatorname{svec}(A_m)^{\mathrm{T}} \end{bmatrix} \in \mathbb{R}^{m\times n(n+1)/2} \qquad (688)$$

semidefinite Farkas' lemma assumes that a convex cone

$$\mathcal{K} = \{A\ \operatorname{svec} X \mid X\succeq 0\} \qquad (379)$$

is closed per membership relation (319) from which the lemma springs: [248, §I] $\mathcal{K}$ closure is attained when matrix A satisfies the *cone closedness invariance corollary* (p.157). Given closed convex cone $\mathcal{K}$ and its dual from Example 2.13.5.1.1

$$\mathcal{K}^* = \{y \mid \sum_{j=1}^m y_jA_j \succeq 0\} \qquad (386)$$

then we can apply membership relation

$$b\in\mathcal{K} \ \Leftrightarrow\ \langle y\,,\, b\rangle \geq 0\ \ \forall\, y\in\mathcal{K}^* \qquad (319)$$

to obtain the lemma

$$b\in\mathcal{K} \ \Leftrightarrow\ \exists\, X\succeq 0\ \ni\ A\operatorname{svec} X = b \ \Leftrightarrow\ \mathcal{A}\cap\mathbb{S}^n_+\neq\emptyset \qquad (700)$$

$$b\in\mathcal{K} \ \Leftrightarrow\ \langle y\,,b\rangle \geq 0\ \ \forall\, y\in\mathcal{K}^* \ \Leftrightarrow\ \mathcal{A}\cap\mathbb{S}^n_+\neq\emptyset \qquad (701)$$

The final equivalence synopsizes *semidefinite Farkas' lemma.*

While the lemma is correct as stated, a positive definite version is required for semidefinite programming [430, §1.3.8] because existence of a feasible solution in the cone

interior $\mathcal{A} \cap \operatorname{int}\mathbb{S}^n_+$ is required by *Slater's condition*[4.10] to achieve 0 duality gap (optimal primal−dual objective difference §4.2.3, Figure **62**). Geometrically, a positive definite lemma is required to insure that a point of intersection closest to the origin is not at infinity; *e.g*, Figure **48**. Then given $A \in \mathbb{R}^{m\times n(n+1)/2}$ having rank m, we wish to detect existence of nonempty primal feasible set interior to the PSD cone;[4.11] (382)

$$b \in \operatorname{int}\mathcal{K} \;\;\Leftrightarrow\;\; \langle y\,, b\rangle > 0 \;\; \forall\, y \in \mathcal{K}^*,\;\; y \neq \mathbf{0} \;\;\Leftrightarrow\;\; \mathcal{A} \cap \operatorname{int}\mathbb{S}^n_+ \neq \emptyset \tag{702}$$

Positive definite Farkas' lemma is made from proper cones, $\mathcal{K}$ (379) and $\mathcal{K}^*$ (386), and membership relation (325) for which $\mathcal{K}$ closedness is unnecessary:

4.2.1.1.2 Lemma. *Positive definite Farkas' lemma.*
Given l.i. set $\{A_i \in \mathbb{S}^n,\; i=1\dots m\}$ and vector $b=[b_i]\in\mathbb{R}^m$, make affine set

$$\mathcal{A} = \{X \in \mathbb{S}^n \,|\, \langle A_i\,, X\rangle = b_i\,,\; i=1\dots m\} \tag{690}$$

Primal feasible cone interior $\mathcal{A} \cap \operatorname{int}\mathbb{S}^n_+$ is nonempty if and only if $y^{\mathrm{T}}b > 0$ holds for each and every vector $y=[y_i]\neq\mathbf{0}$ such that $\sum_{i=1}^m y_i A_i \succeq 0$.

Equivalently, primal feasible cone interior $\mathcal{A} \cap \operatorname{int}\mathbb{S}^n_+$ is nonempty if and only if $y^{\mathrm{T}}b > 0$ holds for each and every vector $\|y\|=1 \;\ni\; \sum_{i=1}^m y_i A_i \succeq 0$. ◇

4.2.1.1.3 Example. *"New" Farkas' lemma.*
Lasserre [248, §III] presented an example in 1995, originally offered by Ben-Israel in 1969 [33, p.378], to support closedness in *semidefinite Farkas' Lemma* 4.2.1.1.1:

$$A \triangleq \begin{bmatrix} \operatorname{svec}(A_1)^{\mathrm{T}} \\ \operatorname{svec}(A_2)^{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad b = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{703}$$

Intersection $\mathcal{A} \cap \mathbb{S}^n_+$ is practically empty because the solution set

$$\{X \succeq 0 \;|\; A\,\operatorname{svec} X = b\} \;=\; \left\{ \begin{bmatrix} \alpha & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 0 \end{bmatrix} \succeq 0 \;|\; \alpha \in \mathbb{R} \right\} \tag{704}$$

is positive semidefinite only asymptotically ($\alpha \to \infty$). Yet $\sum_{i=1}^m y_i A_i \succeq 0 \Rightarrow y^{\mathrm{T}}b \geq 0$ the dual system erroneously indicates nonempty intersection because $\mathcal{K}$ (379) violates a closedness condition of the lemma; *videlicet*, for $\|y\|=1$

$$y_1 \begin{bmatrix} 0 & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & 0 \end{bmatrix} + y_2 \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \succeq 0 \;\;\Leftrightarrow\;\; y = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \;\;\Rightarrow\;\; y^{\mathrm{T}}b = 0 \tag{705}$$

[4.10] Slater's sufficient constraint qualification is satisfied whenever any primal or dual *strictly feasible solution* exists; *id est*, any point satisfying the respective affine constraints and relatively interior to the convex cone. [347, §6.6] [42, p.325] If the cone were polyhedral, then Slater's constraint qualification is satisfied when any feasible solution exists (relatively interior to the cone or on its relative boundary). [63, §5.2.3]

[4.11] Detection of $\mathcal{A} \cap \operatorname{int}\mathbb{S}^n_+ \neq \emptyset$ by examining $\operatorname{int}\mathcal{K}$ instead is a trick need not be lost.

On the other hand, *positive definite Farkas' Lemma* 4.2.1.1.2 certifies that $\mathcal{A} \cap \operatorname{int} \mathbb{S}_+^n$ is empty; what we need to know for semidefinite programming.

Lasserre suggested addition of another condition to *semidefinite Farkas' lemma* (§4.2.1.1.1) to make a new lemma having no closedness condition. But *positive definite Farkas' lemma* (§4.2.1.1.2) is simpler and obviates the additional condition proposed. □

4.2.1.2 Theorem of the alternative for semidefinite programming

Because these Farkas' lemmas follow from membership relations, we may construct alternative systems from them. Applying the method of §2.13.2.1.1, then from *positive definite Farkas' lemma* we get

$$\begin{gathered} \mathcal{A} \cap \operatorname{int} \mathbb{S}_+^n \neq \emptyset \\ \text{or in the alternative} \\ y^{\mathrm{T}} b \leq 0\,, \quad \sum_{i=1}^m y_i A_i \succeq 0\,, \quad y \neq \mathbf{0} \end{gathered} \tag{706}$$

Any single vector y satisfying the alternative certifies $\mathcal{A} \cap \operatorname{int} \mathbb{S}_+^n$ is empty. Such a vector can be found as a solution to another semidefinite program: for linearly independent (vectorized) set $\{A_i \in \mathbb{S}^n,\ i=1 \dots m\}$

$$\begin{aligned} \underset{y}{\text{minimize}} \quad & y^{\mathrm{T}} b \\ \text{subject to} \quad & \sum_{i=1}^m y_i A_i \succeq 0 \\ & \|y\|^2 \leq 1 \end{aligned} \tag{707}$$

If an optimal vector $y^\star \neq \mathbf{0}$ can be found such that $y^{\star \mathrm{T}} b \leq 0$, then primal feasible cone interior $\mathcal{A} \cap \operatorname{int} \mathbb{S}_+^n$ is empty.

4.2.1.3 Boundary-membership criterion

(*confer* (701)(702)) From boundary-membership relation (329), for proper cones $\mathcal{K}$ (379) and $\mathcal{K}^*$ (386) of linear matrix inequality,

$$b \in \partial\mathcal{K} \quad \Leftrightarrow \quad \exists\, y \neq \mathbf{0} \ \ni\ \langle y\,,\, b \rangle = 0\,, \quad y \in \mathcal{K}^*, \quad b \in \mathcal{K} \quad \Leftrightarrow \quad \partial\mathbb{S}_+^n \supset \mathcal{A} \ \cap\ \mathbb{S}_+^n \neq \emptyset \tag{708}$$

Whether vector $b \in \partial\mathcal{K}$ belongs to cone $\mathcal{K}$ boundary, that is a determination we can indeed make; one that is certainly expressible as a feasibility problem: Given linearly independent set[4.12] $\{A_i \in \mathbb{S}^n,\ i=1 \dots m\}$, for $b \in \mathcal{K}$ (700)

$$\begin{aligned} \text{find} \quad & y \neq \mathbf{0} \\ \text{subject to} \quad & y^{\mathrm{T}} b = 0 \\ & \sum_{i=1}^m y_i A_i \succeq 0 \end{aligned} \tag{709}$$

[4.12] From the results of Example 2.13.5.1.1, vector b on the boundary of $\mathcal{K}$ cannot be detected simply by looking for 0 eigenvalues in matrix X. We do not consider a skinny-or-square matrix A because then feasible set $\mathcal{A} \cap \mathbb{S}_+^n$ is at most a single point.

Any such nonzero solution y certifies that affine subset $\mathcal{A}$ (690) intersects the positive semidefinite cone $\mathbb{S}^n_+$ only on its boundary; in other words, nonempty feasible set $\mathcal{A} \cap \mathbb{S}^n_+$ belongs to the positive semidefinite cone boundary $\partial\mathbb{S}^n_+$.

4.2.2 Duals

The dual objective function from (687D) evaluated at any feasible solution represents a lower bound on the primal optimal objective value from (687P). We can see this by direct substitution: Assume the feasible sets $\mathcal{A} \cap \mathbb{S}^n_+$ and $\mathcal{D}^*$ are nonempty. Then it is always true:

$$\begin{aligned} \langle C \,,\, X\rangle &\geq \langle b \,,\, y\rangle \\ \left\langle \sum_i y_i A_i + S \,,\, X \right\rangle &\geq [\, \langle A_1 \,,\, X\rangle \cdots \langle A_m \,,\, X\rangle \,]\; y \\ \langle S \,,\, X\rangle &\geq 0 \end{aligned} \tag{710}$$

The converse also follows because

$$X \succeq 0\,,\;\; S \succeq 0 \;\;\Rightarrow\;\; \langle S \,,\, X\rangle \geq 0 \tag{1574}$$

Optimal value of the dual objective thus represents the greatest lower bound on the primal. This fact is known as the *weak duality theorem* for semidefinite programming, [430, §1.3.8] and can be used to detect convergence in any primal/dual numerical method of solution.

4.2.2.1 Dual problem statement is not unique

Even subtle but equivalent restatements of a primal convex problem can lead to vastly different statements of a corresponding dual problem. This phenomenon is of interest because a particular instantiation of dual problem might be easier to solve numerically or it might take one of few forms for which analytical solution is known.

Here is a canonical restatement of prototypical dual semidefinite program (687D), for example, equivalent by (194):

$$\text{(D)}\quad \begin{array}{ll} \underset{y\in\mathbb{R}^m,\; S\in\mathbb{S}^n}{\text{maximize}} & \langle b \,,\, y\rangle \\ \text{subject to} & S \succeq 0 \\ & \operatorname{svec}^{-1}(A^{\mathrm{T}} y) + S = C \end{array} \quad\equiv\quad \begin{array}{ll} \underset{y\in\mathbb{R}^m}{\text{maximize}} & \langle b \,,\, y\rangle \\ \text{subject to} & \operatorname{svec}^{-1}(A^{\mathrm{T}} y) \preceq C \end{array} \tag{687$\tilde{\text{D}}$}$$

Dual feasible cone interior in $\operatorname{int}\mathbb{S}^n_+$ (699) (689) thereby corresponds with canonical dual ($\tilde{\text{D}}$) feasible interior

$$\operatorname{rel\,int}\tilde{\mathcal{D}}^* \triangleq \left\{ y\in\mathbb{R}^m \mid \sum_{i=1}^m y_i A_i \prec C \right\} \tag{711}$$

4.2.2.1.1 Exercise. *Primal prototypical semidefinite program.*
Derive prototypical primal (687P) from its canonical dual (687$\tilde{\text{D}}$); *id est*, demonstrate that particular connectivity in Figure **88**. ▼

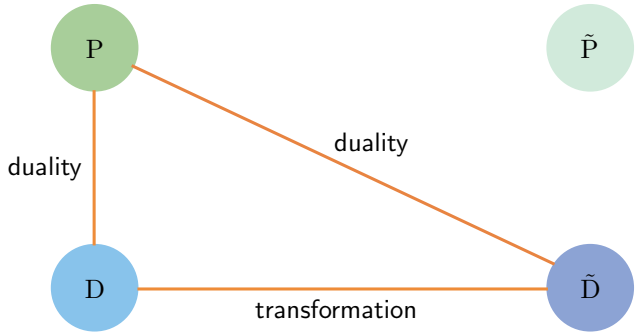


Figure 88: Connectivity indicates paths between particular primal and dual problems from Exercise 4.2.2.1.1. More generally, any path between primal problems P (and equivalent P̃) and dual D (and equivalent D̃) is possible: implying, any given path is not necessarily circuital; dual of a dual problem is not necessarily stated in precisely same manner as corresponding primal convex problem, in other words, although its solution set is equivalent to within some transformation.

4.2.3 Optimality conditions

When primal feasible cone interior $\mathcal{A} \cap \operatorname{int} \mathbb{S}^n_+$ exists in $\mathbb{S}^n$ or when canonical dual feasible interior $\operatorname{rel\,int} \tilde{\mathcal{D}}^*$ exists in $\mathbb{R}^m$, then these two problems (687P) (687D) become strong duals by Slater's sufficient condition (p.249). In other words, the primal optimal objective value becomes equal to the dual optimal objective value: there is no duality gap (Figure **62**) and so determination of convergence is facilitated; *id est*, if $\exists\, X \in \mathcal{A} \cap \operatorname{int} \mathbb{S}^n_+$ or $\exists\, y \in \operatorname{rel\,int} \tilde{\mathcal{D}}^*$ then

$$\begin{aligned} \langle C \,,\, X^\star \rangle &= \langle b \,,\, y^\star \rangle \\ \left\langle \sum_i y_i^\star A_i + S^\star \,,\, X^\star \right\rangle &= [\, \langle A_1 \,,\, X^\star \rangle \cdots \langle A_m \,,\, X^\star \rangle \,]\; y^\star \\ \langle S^\star \,,\, X^\star \rangle &= 0 \end{aligned} \tag{712}$$

where $S^\star$, $y^\star$ denote a dual optimal solution.[4.13] We summarize this:

4.2.3.0.1 Corollary. *Optimality and strong duality.* [378, §3.1] [430, §1.3.8]
For semidefinite programs (687P) and (687D), assume primal and dual feasible sets $\mathcal{A} \cap \mathbb{S}^n_+ \subset \mathbb{S}^n$ and $\mathcal{D}^* \subset \mathbb{S}^n \times \mathbb{R}^m$ (699) are nonempty. Then

- $X^\star$ is optimal for (687P)
- $S^\star$, $y^\star$ are optimal for (687D)
- duality gap $\langle C, X^\star \rangle - \langle b \,,\, y^\star \rangle$ is 0

[4.13] Optimality condition $\langle S^\star \,,\, X^\star \rangle = 0$ is called a *complementary slackness condition,* in keeping with linear programming tradition [98], that forbids dual inequalities in (687) to simultaneously hold strictly. [324, §4]

if and only if

i) $\exists\, X \in \mathcal{A} \cap \operatorname{int} \mathbb{S}^n_+$ **or** $\exists\, y \in \operatorname{rel\,int} \tilde{\mathcal{D}}^*$

and

ii) $\langle S^\star\,,\, X^\star \rangle = 0$ ◇

For symmetric positive semidefinite matrices, requirement **ii** is equivalent to the *complementarity* (§A.7.4)

$$\langle S^\star\,,\, X^\star \rangle = 0 \;\;\Leftrightarrow\;\; S^\star X^\star = X^\star S^\star = \mathbf{0} \tag{713}$$

Commutativity of diagonalizable matrices is a necessary and sufficient condition [218, §1.3.12] for these two optimal symmetric matrices to be simultaneously diagonalizable. Therefore

$$\operatorname{rank} X^\star + \operatorname{rank} S^\star \leq n \tag{714}$$

Proof. To see that, the product of symmetric optimal matrices $X^\star, S^\star \in \mathbb{S}^n$ must itself be symmetric because of commutativity. (1563) The symmetric product has diagonalization [12, cor.2.11]

$$S^\star X^\star = X^\star S^\star = Q\,\Lambda_{S^\star}\Lambda_{X^\star}\,Q^{\mathrm{T}} = \mathbf{0} \;\;\Leftrightarrow\;\; \Lambda_{X^\star}\Lambda_{S^\star} = \mathbf{0} \tag{715}$$

where Q is an orthogonal matrix. The product of the nonnegative diagonal Λ matrices can be $\mathbf{0}$ if their main diagonal zeros are complementary or coincide. Due only to symmetry, $\operatorname{rank} X^\star = \operatorname{rank} \Lambda_{X^\star}$ and $\operatorname{rank} S^\star = \operatorname{rank} \Lambda_{S^\star}$ for these optimal primal and dual solutions. (1549) So, because of the complementarity, the total number of nonzero diagonal entries from both Λ cannot exceed n. ♦

When equality is attained in (714)

$$\operatorname{rank} X^\star + \operatorname{rank} S^\star = n \tag{716}$$

there are no coinciding main diagonal zeros in $\Lambda_{X^\star}\Lambda_{S^\star}$, and so we have what is called *strict complementarity*.[4.14] Logically it follows that a necessary and sufficient condition for strict complementarity of an optimal primal and dual solution is

$$X^\star + S^\star \succ 0 \tag{717}$$

4.2.3.1 solving primal problem via dual

The beauty of Corollary 4.2.3.0.1 is its conjugacy; *id est*, one can solve either the primal or dual problem in (687) and then find a solution to the other via the optimality conditions. When a dual optimal solution is known, for example, a primal optimal solution is any primal feasible solution in hyperplane $\{X \mid \langle S^\star\,,\, X\rangle = 0\}$.

[4.14] distinct from maximal complementarity (§4.1.2).

4.2.3.1.1 Example. *Minimal cardinality Boolean.* [97] [35, §4.3.4] [363] (*confer* Example 4.5.1.5.1) Consider finding a *minimal cardinality* Boolean solution x to the classic linear algebra problem $Ax=b$ given noiseless data $A\in\mathbb{R}^{m\times n}$ and $b\in\mathbb{R}^m$;

$$\begin{array}{ll}\underset{x}{\text{minimize}} & \|x\|_0 \\ \text{subject to} & Ax=b \\ & x_i\in\{0,1\}\,,\quad i=1\ldots n\end{array} \tag{718}$$

where $\|x\|_0$ denotes cardinality of vector x (`a.k.a` 0-*norm*; not a convex function).

A minimal cardinality solution answers the question: "Which fewest linear combination of columns in A constructs vector b?" *Cardinality problems* have extraordinarily wide appeal, arising in many fields of science and across many disciplines. [336] [230] [187] [186] Yet designing an efficient algorithm to optimize cardinality has proved difficult. In this example, we also constrain the variable to be Boolean. The Boolean constraint forces an identical solution were the norm in problem (718) instead the 1-norm or 2-norm; *id est*, the two problems

$$(718)\quad \begin{array}{ll}\underset{x}{\text{minimize}} & \|x\|_0 \\ \text{subject to} & Ax=b \\ & x_i\in\{0,1\}\,,\quad i=1\ldots n\end{array} \;=\; \begin{array}{ll}\underset{x}{\text{minimize}} & \|x\|_1 \\ \text{subject to} & Ax=b \\ & x_i\in\{0,1\}\,,\quad i=1\ldots n\end{array} \tag{719}$$

are the same. The Boolean constraint makes the 1-norm problem nonconvex.

Given data

$$A=\begin{bmatrix}-1 & 1 & 8 & 1 & 1 & 0\\ -3 & 2 & 8 & \frac{1}{2} & \frac{1}{3} & \frac{1}{2}-\frac{1}{3}\\ -9 & 4 & 8 & \frac{1}{4} & \frac{1}{9} & \frac{1}{4}-\frac{1}{9}\end{bmatrix},\qquad b=\begin{bmatrix}1\\ \frac{1}{2}\\ \frac{1}{4}\end{bmatrix} \tag{720}$$

the obvious and desired solution to the problem posed,

$$x^\star=e_4\in\mathbb{R}^{\mathbf{6}} \tag{721}$$

has norm $\|x^\star\|_2=1$ and minimal cardinality; the minimum number of nonzero entries in vector x. The MATLAB backslash command `x=A\b`, for example, finds

$$x_{\text{M}}=\begin{bmatrix}\frac{2}{128}\\ 0\\ \frac{5}{128}\\ 0\\ \frac{90}{128}\\ 0\end{bmatrix} \tag{722}$$

having norm $\|x_{\text{M}}\|_2=0.7044$. Coincidentally, x_{M} is a 1-norm solution; *id est*, an optimal solution to

$$\begin{array}{ll}\underset{x}{\text{minimize}} & \|x\|_1 \\ \text{subject to} & Ax=b\end{array} \tag{517}$$

The pseudoinverse solution (rounded)

$$x_{\mathbf{P}} = A^{\dagger}b = \begin{bmatrix} -0.0456 \\ -0.1881 \\ 0.0623 \\ 0.2668 \\ 0.3770 \\ -0.1102 \end{bmatrix} \tag{723}$$

has least norm $\|x_{\mathbf{P}}\|_2 = 0.5165$; *id est*, the optimal solution to (§E.0.1.0.1)

$$\begin{array}{ll} \underset{x}{\text{minimize}} & \|x\|_2 \\ \text{subject to} & Ax = b \end{array} \tag{724}$$

Certainly none of the traditional methods provide $x^{\star} = e_4$ (721) because, and in general, for $Ax = b$

$$\left\|\arg\inf \|x\|_2\right\|_2 \leq \left\|\arg\inf \|x\|_1\right\|_2 \leq \left\|\arg\inf \|x\|_0\right\|_2 \tag{725}$$

We can reformulate this minimal cardinality Boolean problem (718) as a semidefinite program: First transform the variable

$$x \triangleq (\hat{x} + \mathbf{1})\tfrac{1}{2} \tag{726}$$

so $\hat{x}_i \in \{-1, 1\}$; equivalently,

$$\begin{array}{ll} \underset{\hat{x}}{\text{minimize}} & \|(\hat{x} + \mathbf{1})\tfrac{1}{2}\|_0 \\ \text{subject to} & A(\hat{x} + \mathbf{1})\tfrac{1}{2} = b \\ & \delta(\hat{x}\hat{x}^{\mathrm{T}}) = \mathbf{1} \end{array} \tag{727}$$

where δ is the main-diagonal linear operator (§A.1). By assigning (§B.1)

$$G = \begin{bmatrix} \hat{x} \\ 1 \end{bmatrix} \begin{bmatrix} \hat{x}^{\mathrm{T}} & 1 \end{bmatrix} = \begin{bmatrix} X & \hat{x} \\ \hat{x}^{\mathrm{T}} & 1 \end{bmatrix} \triangleq \begin{bmatrix} \hat{x}\hat{x}^{\mathrm{T}} & \hat{x} \\ \hat{x}^{\mathrm{T}} & 1 \end{bmatrix} \in \mathbb{S}^{n+1} \tag{728}$$

problem (727) becomes equivalent to: (Theorem A.3.1.0.7)

$$\begin{array}{ll} \underset{X \in \mathbb{S}^n,\ \hat{x} \in \mathbb{R}^n}{\text{minimize}} & \mathbf{1}^{\mathrm{T}}\hat{x} \\ \text{subject to} & A(\hat{x} + \mathbf{1})\tfrac{1}{2} = b \\ & G = \begin{bmatrix} X & \hat{x} \\ \hat{x}^{\mathrm{T}} & 1 \end{bmatrix} (\succeq 0) \\ & \delta(X) = \mathbf{1} \\ & \operatorname{rank} G = 1 \end{array} \tag{729}$$

where solution is confined to rank-1 vertices of the *elliptope* in $\mathbb{S}^{n+1}$ (§5.9.1.0.1) by the rank constraint, the positive semidefiniteness, and the equality constraints $\delta(X) = \mathbf{1}$. The

rank constraint makes this problem nonconvex; by removing it[4.15] we get the semidefinite program

$$\begin{array}{ll}\underset{X\in\mathbb{S}^n,\ \hat{x}\in\mathbb{R}^n}{\text{minimize}} & \mathbf{1}^{\mathrm{T}}\hat{x} \\ \text{subject to} & A(\hat{x}+\mathbf{1})\frac{1}{2} = b \\ & G = \begin{bmatrix} X & \hat{x} \\ \hat{x}^{\mathrm{T}} & 1 \end{bmatrix} \succeq 0 \\ & \delta(X) = \mathbf{1} \end{array} \tag{730}$$

whose optimal solution $x^\star$ (726) is identical to that of minimal cardinality Boolean problem (718) if and only if $\operatorname{rank} G^\star = 1$.

Hope[4.16] of acquiring a rank-1 solution is not ill-founded because 2^n elliptope vertices have rank 1, and we are minimizing an affine function on a subset of the elliptope (Figure 144) containing rank-1 vertices; *id est*, by assumption that the feasible set of minimal cardinality Boolean problem (718) is nonempty, a desired solution resides on the elliptope relative boundary at a rank-1 vertex.[4.17]

For that data given in (720), our semidefinite program solver `sdpsol` [423] [424] (accurate in solution to approximately `1E-8`)[4.18] finds optimal solution to (730)

$$\operatorname{round}(G^\star) = \begin{bmatrix} 1 & 1 & 1 & -1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 & 1 & 1 & -1 \\ -1 & -1 & -1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & -1 & 1 & 1 & -1 \\ 1 & 1 & 1 & -1 & 1 & 1 & -1 \\ -1 & -1 & -1 & 1 & -1 & -1 & 1 \end{bmatrix} \tag{731}$$

near a rank-1 vertex of the elliptope in $\mathbb{S}^{n+1}$ (Theorem 5.9.1.0.2); its sorted eigenvalues,

$$\lambda(G^\star) = \begin{bmatrix} 6.99999977799099 \\ 0.00000022687241 \\ 0.00000002250296 \\ 0.00000000262974 \\ -0.00000000999738 \\ -0.00000000999875 \\ -0.00000001000000 \end{bmatrix} \tag{732}$$

Negative eigenvalues are undoubtedly finite-precision effects. Because the largest eigenvalue predominates by many orders of magnitude, we can expect to find a good

[4.15] Relaxed problem (730) can also be derived via Lagrange duality; it is a dual of a dual program [*sic*] to (729). [322] [63, §5, exer.5.39] [416, §IV] [164, §11.3.4] The relaxed problem must therefore be convex having a larger feasible set; its optimal objective value represents a generally *loose* lower bound (1778) on the optimal objective of problem (729).

[4.16] A more deterministic approach to constraining rank and cardinality is in §4.6.0.0.12.

[4.17] Confinement to the elliptope can be regarded as a kind of normalization akin to matrix A column normalization suggested in [131] and explored in Example 4.2.3.1.2.

[4.18] A typically ignored limitation of interior-point solution methods is their relative accuracy of only about `1E-8` on a machine using 64-bit (*double precision*) floating-point arithmetic; *id est*, optimal solution $x^\star$ cannot be more accurate than square root of machine epsilon ($\epsilon = $ `2.2204E-16`). Nonzero primal−dual objective difference is not a good measure of solution accuracy.

approximation to a minimal cardinality Boolean solution by truncating all smaller eigenvalues. We find, indeed, the desired result (721)

$$x^\star = \text{round}\left(\begin{bmatrix} 0.00000000127947 \\ 0.00000000527369 \\ 0.00000000181001 \\ 0.99999997469044 \\ 0.00000001408950 \\ 0.00000000482903 \end{bmatrix}\right) = e_4 \tag{733}$$

These numerical results are solver dependent; insofar, not all SDP solvers will return a rank-1 vertex solution. □

4.2.3.1.2 Example. *Optimization over elliptope* versus 1*-norm polyhedron for minimal cardinality Boolean Example 4.2.3.1.1.*

A minimal cardinality problem is typically formulated via, what is by now, a standard practice [131] [71, §3.2, §3.4] of column normalization applied to a 1-norm problem surrogate like (517). Suppose we define a diagonal matrix

$$\Lambda \triangleq \begin{bmatrix} \|A(:,1)\|_2 & & & \mathbf{0} \\ & \|A(:,2)\|_2 & & \\ & & \ddots & \\ \mathbf{0} & & & \|A(:,6)\|_2 \end{bmatrix} \in \mathbb{S}^{\mathbf{6}} \tag{734}$$

used to normalize the columns (assumed nonzero) of given noiseless data matrix A. Then approximate the minimal cardinality Boolean problem

$$\begin{array}{ll} \underset{x}{\text{minimize}} & \|x\|_0 \\ \text{subject to} & Ax = b \\ & x_i \in \{0,1\}\,, \quad i = 1 \dots n \end{array} \tag{718}$$

as

$$\begin{array}{ll} \underset{\tilde{y}}{\text{minimize}} & \|\tilde{y}\|_1 \\ \text{subject to} & A\Lambda^{-1}\tilde{y} = b \\ & \mathbf{1} \succeq \Lambda^{-1}\tilde{y} \succeq 0 \end{array} \tag{735}$$

where optimal solution

$$y^\star = \text{round}(\Lambda^{-1}\tilde{y}^\star) \tag{736}$$

The inequality in (735) relaxes Boolean constraint $y_i \in \{0,1\}$ from (718); bounding any solution $y^\star$ to a nonnegative unit hypercube whose vertices are binary numbers. Convex problem (735) is justified by the *convex envelope*

$$\text{cenv}\,\|x\|_0 \;\text{ on }\; \{x \in \mathbb{R}^n \mid \|x\|_\infty \le \kappa\} \;=\; \frac{1}{\kappa}\|x\|_1 \tag{1455}$$

Donoho concurs with this particular formulation, equivalently expressible as a linear program via (513).

Approximation (735) is therefore equivalent to minimization of an affine function (§3.2) on a bounded polyhedron, whereas semidefinite program

$$\begin{array}{ll}\underset{X\in\mathbb{S}^n,\ \hat{x}\in\mathbb{R}^n}{\text{minimize}} & \mathbf{1}^{\mathrm{T}}\hat{x} \\ \text{subject to} & A(\hat{x}+\mathbf{1})\frac{1}{2} = b \\ & G = \begin{bmatrix} X & \hat{x} \\ \hat{x}^{\mathrm{T}} & 1 \end{bmatrix} \succeq 0 \\ & \delta(X) = \mathbf{1} \end{array} \tag{730}$$

minimizes an affine function on an intersection of the elliptope with hyperplanes. Although the same Boolean solution is obtained from this approximation (735) as compared with semidefinite program (730), when given that particular data from Example 4.2.3.1.1, Singer confides a counterexample: Instead, given data

$$A = \begin{bmatrix} 1 & 0 & \frac{1}{\sqrt{2}} \\ 0 & 1 & \frac{1}{\sqrt{2}} \end{bmatrix}, \qquad b = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \tag{737}$$

then solving approximation (735) yields

$$y^\star = \operatorname{round}\left(\begin{bmatrix} 1-\frac{1}{\sqrt{2}} \\ 1-\frac{1}{\sqrt{2}} \\ 1 \end{bmatrix}\right) = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{738}$$

(infeasible, with or without rounding, with respect to original problem (718)) whereas solving semidefinite program (730) produces

$$\operatorname{round}(G^\star) = \begin{bmatrix} 1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 \\ -1 & -1 & 1 & -1 \\ 1 & 1 & -1 & 1 \end{bmatrix} \tag{739}$$

with sorted eigenvalues

$$\lambda(G^\star) = \begin{bmatrix} 3.99999965057264 \\ 0.00000035942736 \\ -0.00000000000000 \\ -0.00000001000000 \end{bmatrix} \tag{740}$$

Truncating all but the largest eigenvalue, from (726) we obtain (*confer* $y^\star$)

$$x^\star = \operatorname{round}\left(\begin{bmatrix} 0.99999999625299 \\ 0.99999999625299 \\ 0.00000001434518 \end{bmatrix}\right) = \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} \tag{741}$$

the desired minimal cardinality Boolean result. □

4.2.3.1.3 Exercise. *Minimal cardinality Boolean art.*
Assess general performance of standard-practice approximation (735) as compared with the proposed semidefinite program (730). ▼

4.2.3.1.4 Exercise. *Conic independence.*
Matrix A from (720) is full-rank having three-dimensional nullspace. Find its four conically independent columns. (§2.10) To what part of proper cone $\mathcal{K} = \{Ax \mid x \succeq 0\}$ does vector b belong? ▼

4.2.3.1.5 Exercise. *Linear independence.*
Show why fat matrix A, from compressed sensing problem (517) or (522), may be regarded full-rank without loss of generality. In other words: Is a minimal cardinality solution invariant to linear dependence of rows? ▼

4.3 Rank reduction

> *... it is not clear generally how to predict* $\operatorname{rank} X^\star$ *or* $\operatorname{rank} S^\star$ *before solving the SDP problem.*
>
> −Farid Alizadeh, 1995 [12, p.22]

The premise of rank reduction in semidefinite programming is: an optimal solution $X^\star$ found does not satisfy Barvinok's upper bound (272) on rank. The particular numerical algorithm solving a semidefinite program may have instead returned a high-rank optimal solution (§4.1.2; *e.g*, (698)) when a lower-rank optimal solution was expected. Rank reduction is a means to adjust rank of an optimal solution to (687P), returned by a solver, until it satisfies Barvinok's upper bound with the optimal objective value unchanged.

4.3.1 Posit a perturbation of $X^\star$

Recall from §4.1.2.1, there is an extreme point of $\mathcal{A} \cap \mathbb{S}^n_+$ (690) satisfying upper bound (272) on rank. [25, §2.2] It is therefore sufficient to locate an extreme point of the intersection whose primal objective value (687P) is optimal:[4.19] [120, §31.5.3] [255, §2.4] [256] [8, §3] [309]

Consider again affine subset

$$\mathcal{A} = \{X \in \mathbb{S}^n \mid A \operatorname{svec} X = b\} \tag{690}$$

where for $A_i \in \mathbb{S}^n$

$$A \triangleq \begin{bmatrix} \operatorname{svec}(A_1)^{\mathrm{T}} \\ \vdots \\ \operatorname{svec}(A_m)^{\mathrm{T}} \end{bmatrix} \in \mathbb{R}^{m \times n(n+1)/2} \tag{688}$$

Given any optimal solution $X^\star$ to

$$\begin{array}{ll} \underset{X \in \mathbb{S}^n}{\text{minimize}} & \langle C \,,\, X \rangle \\ \text{subject to} & X \in \mathcal{A} \cap \mathbb{S}^n_+ \end{array} \tag{687P}$$

[4.19] *There is no known construction for Barvinok's tighter result* (277). −Monique Laurent, 2004

whose rank does not satisfy upper bound (272), we posit existence of a set of perturbations

$$\{t_j B_j \mid t_j \in \mathbb{R}\ ,\ B_j \in \mathbb{S}^n,\ j=1\dots n\} \tag{742}$$

such that, for some $0 \le i \le n$ and scalars $\{t_j\ ,\ j=1\dots i\}$,

$$X^\star + \sum_{j=1}^{i} t_j B_j \tag{743}$$

becomes an extreme point of $\mathcal{A} \cap \mathbb{S}^n_+$ and remains an optimal solution of (687P). Membership of (743) to affine subset $\mathcal{A}$ is secured for the i^{th} perturbation by demanding

$$\langle B_i\ , A_j \rangle = 0\ , \quad j=1\dots m \tag{744}$$

while membership to the positive semidefinite cone $\mathbb{S}^n_+$ is insured by small perturbation (753). Feasibility of (743) is insured in this manner, optimality is proved in §4.3.3.

The following simple algorithm has very low computational intensity and locates an optimal extreme point, assuming a nontrivial solution:

4.3.1.0.1 Procedure. *Rank reduction.* [403]
`initialize:` $B_i = \mathbf{0}\ \ \forall i$
`for` iteration `i=1...n`
{

1. `compute` a nonzero perturbation matrix B_i of $X^\star + \sum_{j=1}^{i-1} t_j B_j$
2. `maximize` t_i
   `subject to` $X^\star + \sum_{j=1}^{i} t_j B_j \in \mathbb{S}^n_+$

} ¶

A rank-reduced optimal solution is then

$$X^\star \leftarrow X^\star + \sum_{j=1}^{i} t_j B_j \tag{745}$$

4.3.2 Perturbation form

Perturbations of $X^\star$ are independent of constants $C \in \mathbb{S}^n$ and $b \in \mathbb{R}^m$ in primal and dual problems (687). Numerical accuracy of any rank-reduced result, found by perturbation of an initial optimal solution $X^\star$, is therefore quite dependent upon initial accuracy of $X^\star$.

4.3.2.0.1 Definition. *Matrix step function.* (*confer* §A.6.5.0.1)
Define the signum-like quasiconcave real function $\psi : \mathbb{S}^n \to \mathbb{R}$

$$\psi(Z) \triangleq \begin{cases} 1\ , & Z \succeq 0 \\ -1\ , & \text{otherwise} \end{cases} \tag{746}$$

The value -1 is taken for indefinite or nonzero negative semidefinite argument. △

Deza & Laurent [120, §31.5.3] prove: every perturbation matrix B_i , $i=1\dots n$, is of the form

$$B_i = -\psi(Z_i)R_iZ_iR_i^{\mathrm{T}} \in \mathbb{S}^n \tag{747}$$

where

$$X^\star \triangleq R_1R_1^{\mathrm{T}} , \qquad X^\star + \sum_{j=1}^{i-1} t_jB_j \triangleq R_iR_i^{\mathrm{T}} \in \mathbb{S}^n \tag{748}$$

where the t_j are scalars and $R_i\in\mathbb{R}^{n\times\rho}$ is full-rank and skinny where

$$\rho \triangleq \operatorname{rank}\left(X^\star + \sum_{j=1}^{i-1} t_jB_j\right) \tag{749}$$

and where matrix $Z_i\in\mathbb{S}^\rho$ is found at each iteration i by solving a very simple feasibility problem: [4.20]

$$\begin{array}{rl} \text{find} & Z_i\in\mathbb{S}^\rho \\ \text{subject to} & \langle Z_i \,,\, R_i^{\mathrm{T}}A_jR_i\rangle = 0\,, \qquad j=1\dots m \end{array} \tag{750}$$

Were there a sparsity pattern common to each member of set $\{R_i^{\mathrm{T}}A_jR_i\in\mathbb{S}^\rho,\ j=1\dots m\}$, then a good choice for Z_i has 1 in each entry corresponding to a 0 in the pattern; *id est*, a sparsity pattern complement. At iteration i

$$X^\star + \sum_{j=1}^{i-1} t_jB_j \;+\; t_iB_i \;=\; R_i(I - t_i\psi(Z_i)Z_i)R_i^{\mathrm{T}} \tag{751}$$

By fact (1539), therefore

$$X^\star + \sum_{j=1}^{i-1} t_jB_j \;+\; t_iB_i \succeq 0 \;\Leftrightarrow\; \mathbf{1} - t_i\psi(Z_i)\lambda(Z_i) \succeq 0 \tag{752}$$

where $\lambda(Z_i)\in\mathbb{R}^\rho$ denotes the eigenvalues of Z_i .

Maximization of each t_i in step 2 of the Procedure reduces rank of (751) and locates a new point on the boundary $\partial(\mathcal{A}\cap\mathbb{S}^n_+)$.[4.21] Maximization of t_i thereby has closed form;

$$(t_i^\star)^{-1} = \max\{\psi(Z_i)\lambda(Z_i)_j \;,\; j=1\dots\rho\} \tag{753}$$

[4.20] A simple method of solution is closed-form projection of a random nonzero point on that proper subspace of isometrically isomorphic $\mathbb{R}^{\rho(\rho+1)/2}$ specified by the constraints. (§E.5.0.0.6) Such a solution is nontrivial assuming the specified intersection of hyperplanes is not the origin; guaranteed by $\rho(\rho+1)/2 > m$. Indeed, this geometric intuition about forming the perturbation is what bounds any solution's rank from below; m is fixed by the number of equality constraints in (687P) while rank ρ decreases with each iteration i . Otherwise, we might iterate indefinitely.

[4.21] This holds because rank of a positive semidefinite matrix in $\mathbb{S}^n$ is diminished below n by the number of its 0 eigenvalues (1549), and because a positive semidefinite matrix having one or more 0 eigenvalues corresponds to a point on the PSD cone boundary (193). Necessity and sufficiency are due to the facts: R_i can be completed to a nonsingular matrix (§A.3.1.0.5), and $I - t_i\psi(Z_i)Z_i$ can be padded with zeros while maintaining equivalence in (751).

When Z_i is indefinite, direction of perturbation (determined by $\psi(Z_i)$) is arbitrary. We may take an early exit from the Procedure were Z_i to become $\mathbf{0}$ or were ρ to become equal to 1 (assuming a nontrivial solution) or were

$$\operatorname{rank}\big[\operatorname{svec} R_i^{\mathrm{T}}A_1R_i \;\; \operatorname{svec} R_i^{\mathrm{T}}A_2R_i \;\cdots\; \operatorname{svec} R_i^{\mathrm{T}}A_mR_i\,\big] = \rho(\rho+1)/2 \tag{754}$$

(274) which characterizes rank ρ of any [*sic*] extreme point in $\mathcal{A}\cap\mathbb{S}^n_+$. [255, §2.4] [256]

Proof. Assuming the form of every perturbation matrix is indeed (747), then by (750)

$$\operatorname{svec} Z_i \perp \big[\operatorname{svec}(R_i^{\mathrm{T}}A_1R_i) \;\; \operatorname{svec}(R_i^{\mathrm{T}}A_2R_i) \;\cdots\; \operatorname{svec}(R_i^{\mathrm{T}}A_mR_i)\,\big] \tag{755}$$

By orthogonal complement we have

$$\begin{aligned}&\operatorname{rank}\big[\operatorname{svec}(R_i^{\mathrm{T}}A_1R_i) \;\cdots\; \operatorname{svec}(R_i^{\mathrm{T}}A_mR_i)\,\big]^{\perp}\\ +\;&\operatorname{rank}\big[\operatorname{svec}(R_i^{\mathrm{T}}A_1R_i) \;\cdots\; \operatorname{svec}(R_i^{\mathrm{T}}A_mR_i)\,\big] = \rho(\rho+1)/2\end{aligned} \tag{756}$$

When Z_i can only be $\mathbf{0}$, then the perturbation is null because an extreme point has been found; thus

$$\big[\operatorname{svec}(R_i^{\mathrm{T}}A_1R_i) \;\cdots\; \operatorname{svec}(R_i^{\mathrm{T}}A_mR_i)\,\big]^{\perp} = \mathbf{0} \tag{757}$$

from which the stated result (754) directly follows. ♦

4.3.3 Optimality of perturbed $X^\star$

We show that the optimal objective value is unaltered by perturbation (747); *id est*,

$$\langle C\,,\; X^\star + \sum_{j=1}^{i} t_j B_j\rangle \;=\; \langle C\,,\; X^\star\rangle \tag{758}$$

Proof. From Corollary 4.2.3.0.1 we have the necessary and sufficient relationship between optimal primal and dual solutions under assumption of nonempty primal feasible cone interior $\mathcal{A}\cap\operatorname{int}\mathbb{S}^n_+$:

$$S^\star X^\star = S^\star R_1R_1^{\mathrm{T}} = X^\star S^\star = R_1R_1^{\mathrm{T}}S^\star = \mathbf{0} \tag{759}$$

This means $\mathcal{R}(R_1)\subseteq\mathcal{N}(S^\star)$ and $\mathcal{R}(S^\star)\subseteq\mathcal{N}(R_1^{\mathrm{T}})$. From (748) and (751) we get the sequence:

$$\begin{aligned}
&X^\star = R_1R_1^{\mathrm{T}}\\
&X^\star + t_1B_1 = R_2R_2^{\mathrm{T}} = R_1(I - t_1\psi(Z_1)Z_1)R_1^{\mathrm{T}}\\
&X^\star + t_1B_1 + t_2B_2 = R_3R_3^{\mathrm{T}} = R_2(I - t_2\psi(Z_2)Z_2)R_2^{\mathrm{T}} = R_1(I - t_1\psi(Z_1)Z_1)(I - t_2\psi(Z_2)Z_2)R_1^{\mathrm{T}}\\
&\;\vdots\\
&X^\star + \sum_{j=1}^{i} t_jB_j = R_1\left(\prod_{j=1}^{i}(I - t_j\psi(Z_j)Z_j)\right)R_1^{\mathrm{T}}
\end{aligned} \tag{760}$$

Substituting $C = \operatorname{svec}^{-1}(A^{\mathrm{T}}y^\star) + S^\star$ from (687),

$$\begin{aligned}\langle C\ ,\ X^\star + \sum_{j=1}^{i} t_j B_j\rangle &= \left\langle \operatorname{svec}^{-1}(A^{\mathrm{T}}y^\star) + S^\star\ ,\ R_1\left(\prod_{j=1}^{i}(I - t_j\,\psi(Z_j)Z_j)\right)R_1^{\mathrm{T}}\right\rangle \\ &= \left\langle \sum_{k=1}^{m} y_k^\star A_k\ ,\ X^\star + \sum_{j=1}^{i} t_j B_j\right\rangle \\ &= \left\langle \sum_{k=1}^{m} y_k^\star A_k + S^\star\ ,\ X^\star\right\rangle = \langle C\ ,\ X^\star\rangle\end{aligned} \tag{761}$$

because $\langle B_i\,,\,A_j\rangle = 0\ \ \forall\, i\,,j$ by design (744). ♦

4.3.3.0.1 Example. $A\,\delta(X) = b$.
This academic example demonstrates that a solution found by rank reduction can certainly have rank less than Barvinok's upper bound (272): Assume a given vector b belongs to the conic hull of columns of a given matrix A

$$A = \begin{bmatrix} -1 & 1 & 8 & 1 & 1 \\ -3 & 2 & 8 & \frac{1}{2} & \frac{1}{3} \\ -9 & 4 & 8 & \frac{1}{4} & \frac{1}{9} \end{bmatrix} \in \mathbb{R}^{m\times n}\ , \qquad b = \begin{bmatrix} 1 \\ \frac{1}{2} \\ \frac{1}{4} \end{bmatrix} \in \mathbb{R}^m \tag{762}$$

Consider the convex optimization problem

$$\begin{array}{ll} \underset{X\in\mathbb{S}^5}{\text{minimize}} & \operatorname{tr} X \\ \text{subject to} & X \succeq 0 \\ & A\,\delta(X) = b \end{array} \tag{763}$$

that minimizes the 1-norm of the main diagonal; *id est*, problem (763) is the same as

$$\begin{array}{ll} \underset{X\in\mathbb{S}^5}{\text{minimize}} & \|\delta(X)\|_1 \\ \text{subject to} & X \succeq 0 \\ & A\,\delta(X) = b \end{array} \tag{764}$$

that finds a solution to $A\,\delta(X) = b$. Rank-3 solution $X^\star = \delta(x_{\mathrm{M}})$ is optimal, where (*confer* (722))

$$x_{\mathrm{M}} = \begin{bmatrix} \frac{2}{128} \\ 0 \\ \frac{5}{128} \\ 0 \\ \frac{90}{128} \end{bmatrix} \tag{765}$$

Yet upper bound (272) predicts existence of at most a

$$\operatorname{rank}\text{-}\left(\left\lfloor \frac{\sqrt{8m+1}-1}{2} \right\rfloor = 2\right) \tag{766}$$

feasible solution from $m=3$ equality constraints. To find a lower rank ρ optimal solution to (763) (barring combinatorics), we invoke Procedure 4.3.1.0.1:

`Initialize:`
$C=I\,,\ \rho=3,\ A_j \triangleq \delta\big(A(j,:)\big)\,,\ j=1,2,3,\ X^\star=\delta(x_{\mathrm{M}})\,,\ m=3,\ n=5.$
{

Iteration `i=1`:

`Step 1:` $R_1=\begin{bmatrix} \sqrt{\frac{2}{128}} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \sqrt{\frac{5}{128}} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \sqrt{\frac{90}{128}} \end{bmatrix}.$

$$\begin{array}{rl} \text{find} & Z_1 \in \mathbb{S}^{\mathbf{3}} \\ \text{subject to} & \langle Z_1\,, R_1^{\mathrm{T}} A_j R_1\rangle = 0\,, \qquad j=1,2,3 \end{array} \tag{767}$$

A nonzero randomly selected matrix $Z_1\,$, having $\mathbf{0}$ main diagonal, is a solution yielding nonzero perturbation matrix $B_1\,$. Choose arbitrarily

$$Z_1 = \mathbf{1}\mathbf{1}^{\mathrm{T}} - I \in \mathbb{S}^{\mathbf{3}} \tag{768}$$

then (rounding)

$$B_1 = \begin{bmatrix} 0 & 0 & 0.0247 & 0 & 0.1048 \\ 0 & 0 & 0 & 0 & 0 \\ 0.0247 & 0 & 0 & 0 & 0.1657 \\ 0 & 0 & 0 & 0 & 0 \\ 0.1048 & 0 & 0.1657 & 0 & 0 \end{bmatrix} \tag{769}$$

`Step 2:` $t_1^\star=1$ because $\lambda(Z_1)=[\,-1\ \ -1\ \ 2\,]^{\mathrm{T}}$. So,

$$X^\star \leftarrow \delta(x_{\mathrm{M}}) + B_1 = \begin{bmatrix} \frac{2}{128} & 0 & 0.0247 & 0 & 0.1048 \\ 0 & 0 & 0 & 0 & 0 \\ 0.0247 & 0 & \frac{5}{128} & 0 & 0.1657 \\ 0 & 0 & 0 & 0 & 0 \\ 0.1048 & 0 & 0.1657 & 0 & \frac{90}{128} \end{bmatrix} \tag{770}$$

has rank $\rho \leftarrow 1$ and produces the same optimal objective value.

} □

4.3.3.0.2 Exercise. *Rank reduction of maximal complementarity.*
Apply rank reduction Procedure 4.3.1.0.1 to the *maximal complementarity example* (§4.1.2.3.1). Demonstrate a rank-1 solution; which can certainly be found (by Barvinok's Proposition 2.9.3.0.1) because there is only one equality constraint. ▼

4.3.4 thoughts regarding rank reduction

Because rank reduction Procedure 4.3.1.0.1 is guaranteed only to produce another optimal solution conforming to Barvinok's upper bound (272), the Procedure will not necessarily produce solutions of arbitrarily low rank; but if they exist, the Procedure can. Arbitrariness of search direction when matrix Z_i becomes indefinite, mentioned on page 262, and the enormity of choices for Z_i (750) are liabilities for this algorithm.

4.3.4.1 inequality constraints

The question naturally arises: what to do when a semidefinite program (not in prototypical form (687))[4.22] has linear inequality constraints of the form

$$\alpha_i^{\mathrm{T}} \operatorname{svec} X \preceq \beta_i \ , \quad i = 1 \dots k \tag{771}$$

where $\{\beta_i\}$ are given scalars and $\{\alpha_i\}$ are given vectors. One expedient way to handle this circumstance is to convert the inequality constraints to equality constraints by introducing a slack variable γ; *id est*,

$$\alpha_i^{\mathrm{T}} \operatorname{svec} X + \gamma_i = \beta_i \ , \quad i = 1 \dots k \ , \qquad \gamma \succeq 0 \tag{772}$$

thereby converting the problem to prototypical form.

Alternatively, we say the i^{th} inequality constraint is *active* when it is met with equality; *id est*, when for particular i in (771), $\alpha_i^{\mathrm{T}} \operatorname{svec} X^\star = \beta_i$. An optimal high-rank solution $X^\star$ is, of course, feasible (satisfying all the constraints). But for the purpose of rank reduction, inactive inequality constraints are ignored while active inequality constraints are interpreted as equality constraints. In other words, we take the union of active inequality constraints (as equalities) with equality constraints $A \operatorname{svec} X = b$ to form a composite affine subset $\hat{\mathcal{A}}$ substituting for (690). Then we proceed with rank reduction of $X^\star$ as though the semidefinite program were in prototypical form (687P).

4.4 Rank-constrained semidefinite program

We generalize the trace heuristic (§7.2.2.1), for finding low-rank optimal solutions to SDPs of a more general form:

4.4.1 rank constraint by convex iteration

Consider a *semidefinite feasibility problem* of the form

$$\begin{array}{ll} \underset{G \in \mathbb{S}^N}{\text{find}} & G \\ \text{subject to} & G \in \mathcal{C} \\ & G \succeq 0 \\ & \operatorname{rank} G \leq n \end{array} \tag{773}$$

[4.22] Contemporary numerical packages for solving semidefinite programs can solve a range of problems wider than prototype (687). Generally, they do so by transforming a given problem into prototypical form by introducing new constraints and variables. [12] [424] We are momentarily considering a departure from the primal prototype that augments the constraint set with linear inequalities.

where $\mathcal{C}$ is a convex set presumed to contain positive semidefinite matrices of rank n or less; *id est*, $\mathcal{C}$ intersects the positive semidefinite cone boundary. We propose that this rank-constrained feasibility problem can be equivalently expressed as iteration of the convex problem sequence (774) and (1800a):

$$\begin{array}{ll} \underset{G\in\mathbb{S}^N}{\text{minimize}} & \langle G\,,\, W\rangle \\ \text{subject to} & G\in\mathcal{C} \\ & G\succeq 0 \end{array} \tag{774}$$

where *direction vector*[4.23] W is an optimal solution to semidefinite program, for $0\leq n\leq N-1$

$$\sum_{i=n+1}^{N}\lambda(G^\star)_i \;=\; \begin{array}{ll} \underset{W\in\mathbb{S}^N}{\text{minimize}} & \langle G^\star,\, W\rangle \\ \text{subject to} & 0\preceq W\preceq I \\ & \operatorname{tr} W = N-n \end{array} \tag{1800a}$$

whose feasible set is a Fantope (§2.3.2.0.1), and where $G^\star$ is an optimal solution to problem (774) given some iterate W. The idea is to iterate solution of (774) and (1800a) until convergence as defined in §4.4.1.2:[4.24] (*confer* (810))

$$\sum_{i=n+1}^{N}\lambda(G^\star)_i \;=\; \langle G^\star,\, W^\star\rangle \;=\; \lambda(G^\star)^{\mathrm{T}}\lambda(W^\star) \;\triangleq\; 0 \tag{775}$$

defines *global convergence* of the iteration; a vanishing objective that is a certificate of global optimality but cannot be guaranteed. *Optimal direction vector* $W^\star$ is defined as any positive semidefinite matrix yielding optimal solution $G^\star$ of rank n or less to then convex equivalent (774) of feasibility problem (773):

$$(773)\qquad \begin{array}{ll} \underset{G\in\mathbb{S}^N}{\text{find}} & G \\ \text{subject to} & G\in\mathcal{C} \\ & G\succeq 0 \\ & \operatorname{rank} G\leq n \end{array} \quad\equiv\quad \begin{array}{ll} \underset{G\in\mathbb{S}^N}{\text{minimize}} & \langle G\,,\, W^\star\rangle \\ \text{subject to} & G\in\mathcal{C} \\ & G\succeq 0 \end{array} \qquad (774)$$

id est, any direction vector for which the last $N-n$ nonincreasingly ordered eigenvalues λ of $G^\star$ are zero.

In any semidefinite feasibility problem, a solution of least rank must be an extreme point of the feasible set.[4.25] This means there exists a hyperplane supporting the feasible set at that extreme point. Then there must exist a linear objective function such that this least-rank feasible solution optimizes the resultant semidefinite program.

[4.23] Search *direction* W is a hyperplane-normal pointing opposite to direction of movement describing minimization of a real linear function $\langle G\,,W\rangle$ (p.67).

[4.24] Proposed iteration is neither *dual projection* (Figure **179**) or *alternating projection* (Figure **183**). Sum of eigenvalues follows from results of Ky Fan (page 567). Inner product of eigenvalues follows from (1681) and properties of commutative matrix products (page 524).

[4.25] which follows by *extremes theorem* 2.8.1.1.1, by rank of a sum of positive semidefinite matrices (1555) (257), and by definition of extreme point (167) for which no convex combination can produce it: If a least rank solution were expressible as a convex combination of feasible points, then there could exist feasible matrices of lesser rank.

We emphasize that convex problem (774) is not a relaxation of rank-constrained feasibility problem (773); at global convergence, convex iteration (774) (1800a) makes it instead an *equivalent problem.*

4.4.1.1 direction matrix interpretation

(*confer* §4.5.1.2) The feasible set of direction matrices in (1800a) is the convex hull of outer product of all rank-$(N-n)$ orthonormal matrices; *videlicet*,

$$\operatorname{conv}\left\{UU^{\mathrm{T}} \mid U\in\mathbb{R}^{N\times N-n},\ U^{\mathrm{T}}U=I\right\} = \left\{A\in\mathbb{S}^N \mid I\succeq A\succeq 0\,,\ \langle I\,,\,A\rangle = N-n\right\} \quad (90)$$

This set (92), argument to $\operatorname{conv}\{\,\}$, comprises the extreme points of this Fantope (90). An optimal solution W to (1800a), that is an extreme point, is known in closed form (p.567): Given ordered diagonalization $G^\star = Q\Lambda Q^{\mathrm{T}} \in \mathbb{S}^N_+$ (§A.5.1), then direction matrix $W = U^\star U^{\star\mathrm{T}}$ is optimal and extreme where $U^\star = Q(:\,,\,n+1:N)\in\mathbb{R}^{N\times N-n}$. Eigenvalue vector $\lambda(W)$ has 1 in each entry corresponding to the $N-n$ smallest entries of $\delta(\Lambda)$ and has 0 elsewhere. By (221) (223), polar direction $-W$ can be regarded as pointing toward the set of all rank-n (or less) positive semidefinite matrices whose nullspace contains that of $G^\star$. For that particular closed-form solution W, consequently, (*confer* (812))

$$\sum_{i=n+1}^{N} \lambda(G^\star)_i \;=\; \langle G^\star,\, W\rangle \;=\; \lambda(G^\star)^{\mathrm{T}}\lambda(W) \;\geq\; 0 \quad (776)$$

This is the connection to cardinality minimization of vectors;[4.26] *id est*, eigenvalue λ cardinality (rank) is analogous to vector x cardinality via (812): for positive semidefinite X

$$\begin{array}{rcccl} \sum_i \lambda(X)_i = \operatorname{tr} X & = & \|X\|_2^* & \Leftrightarrow & \|x\|_1 \\ \sqrt{\sum_i \lambda(X)_i^2} = \sqrt{\operatorname{tr} X^2} & = & \|X\|_{\mathrm{F}} & \Leftrightarrow & \|x\|_2 \\ \max_i\{\lambda(X)_i\} & = & \|X\|_2 & \Leftrightarrow & \|x\|_\infty \end{array} \quad (777)$$

So that this method, for constraining rank, will not be misconstrued under closed-form solution W to (1800a): Define (*confer* (221))

$$\mathcal{S}_n \;\triangleq\; \{(I-W)G(I-W)\,|\,G\in\mathbb{S}^N\} \;=\; \{X\in\mathbb{S}^N \mid \mathcal{N}(X)\supseteq\mathcal{N}(G^\star)\} \quad (778)$$

as the symmetric subspace of $\operatorname{rank}\leq n$ matrices whose nullspace contains $\mathcal{N}(G^\star)$. Then projection of $G^\star$ on $\mathcal{S}_n$ is $(I-W)G^\star(I-W)$. (§E.7) Direction of projection is $-WG^\star W$. (Figure **89**) $\operatorname{tr}(WG^\star W)$ is a measure of proximity to $\mathcal{S}_n$ because its orthogonal complement is $\mathcal{S}_n^\perp = \{WGW\,|\,G\in\mathbb{S}^N\}$; the point being, convex iteration incorporating constrained $\operatorname{tr}(WGW) = \langle G\,,\,W\rangle$ minimization is not a projection method: certainly, not on these two subspaces.

[4.26] not trace minimization of a nonnegative diagonal matrix $\delta(x)$ as in [146, §1] [318, §2]. To make rank-constrained problem (773) resemble cardinality problem (529), we could make $\mathcal{C}$ an affine subset:

$$\begin{array}{ll} \text{find} & X\in\mathbb{S}^N \\ \text{subject to} & A\operatorname{svec} X = b \\ & X\succeq 0 \\ & \operatorname{rank} X\leq n \end{array}$$

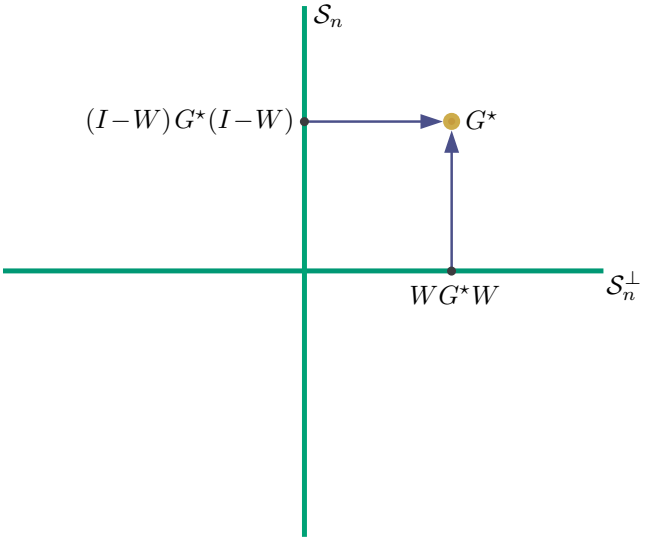


Figure 89: (*confer* Figure **180**) Projection of $G^\star$ on subspace $\mathcal{S}_n$ of rank$\leq n$ matrices whose nullspace contains $\mathcal{N}(G^\star)$. This direction W is closed-form solution to (1800a).

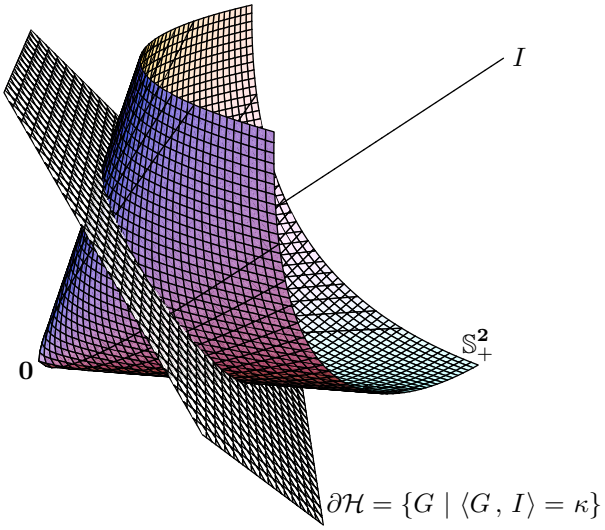


Figure 90: (*confer* Figure **106**) Trace heuristic can be interpreted as minimization of a hyperplane, with normal I, over positive semidefinite cone drawn here in isometrically isomorphic $\mathbb{R}^{\mathbf{3}}$. Polar of direction vector $W = I$ points toward origin.

Closed-form solution W to problem (1800a), though efficient, comes with a *caveat*: there exist cases where this projection matrix solution W does not provide the shortest route to an optimal rank-n solution $G^\star$; *id est*, direction W is not unique. So we sometimes choose to solve (1800a) instead of employing a known closed-form solution.

When direction matrix $W = I$, as in the trace heuristic for example, then $-W$ points directly at the origin (the rank-0 PSD matrix, Figure **90**). Vector inner-product of an optimization variable with direction matrix W is therefore a generalization of the trace heuristic (§7.2.2.1) for rank minimization; $-W$ is instead trained toward the boundary of the positive semidefinite cone.

4.4.1.2 convergence

We study convergence to ascertain conditions under which a direction matrix will reveal a feasible solution G, of rank n or less, to semidefinite program (774). Denote by $W^\star$ a particular optimal direction matrix from semidefinite program (1800a) such that (775) holds (feasible rank $G \leq n$ found). Then we define *global convergence* of the iteration (774) (1800a) to correspond with this vanishing vector inner-product (775) of optimal solutions.

Because this iterative technique for constraining rank is not a projection method, it can find a rank-n solution $G^\star$ **(**(775) will be satisfied**)** only if at least one exists in the feasible set of program (774).

4.4.1.2.1 Proof. Suppose $\langle G^\star, W \rangle = \tau$ is satisfied for some nonnegative constant τ after any particular iteration (774) (1800a) of the two minimization problems. Once a particular value of τ is achieved, it can never be exceeded by subsequent iterations because existence of feasible G and W having that vector inner-product τ has been established simultaneously in each problem. Because the infimum of vector inner-product of two positive semidefinite matrix variables is zero, the nonincreasing sequence of iterations is thus bounded below hence convergent because any bounded monotonic sequence in $\mathbb{R}$ is convergent. [274, §1.2] [43, §1.1] *Local convergence* to some nonnegative objective value τ is thereby established. ♦

Local convergence, in this context, means convergence to a *fixed point* of possibly infeasible rank. Only local convergence can be established because objective $\langle G\,,\, W \rangle$, when instead regarded simultaneously in two variables $(G\,,\, W)$, is generally multimodal. (§3.8.0.0.3)

Local convergence, convergence to $\tau \neq 0$ and definition of a *stall,* never implies nonexistence of a rank-n feasible solution to (774). A nonexistent rank-n feasible solution would mean certain failure to converge globally by definition (775) (convergence to $\tau \neq 0$) but, as proved, convex iteration always converges locally if not globally.

When a rank-n feasible solution to (774) exists, it remains an open problem to state conditions under which $\langle G^\star, W^\star \rangle = \tau = 0$ (775) is achieved by iterative solution of semidefinite programs (774) and (1800a). Then rank $G^\star \leq n$ and pair $(G^\star, W^\star)$ becomes a globally optimal fixed point of iteration. There can be no proof of global convergence because of the implicit high-dimensional multimodal manifold in variables $(G\,,\, W)$. When stall occurs, direction vector W can be manipulated to steer out; *e.g*, reversal of search direction as in Example 4.6.0.0.1, or reinitialization to a random

rank-$(N-n)$ matrix in the same positive semidefinite cone face (§2.9.2.3) demanded by the current iterate: given ordered diagonalization $G^\star = Q\Lambda Q^{\mathrm{T}} \in \mathbb{S}^N$, then $W = U^\star \Phi U^{\star\mathrm{T}}$ where $U^\star = Q(:, n{+}1{:}N) \in \mathbb{R}^{N\times N-n}$ and where eigenvalue vector $\lambda(W)_{1:N-n} = \lambda(\Phi)$ has nonnegative uniformly distributed random entries in $(0, 1]$ by selection of $\Phi \in \mathbb{S}_+^{N-n}$ while $\lambda(W)_{N-n+1:N} = \mathbf{0}$. Zero eigenvalues act as memory while randomness largely reduces likelihood of stall. When this direction works, rank and objective sequence $\langle G^\star,\, W\rangle$ with respect to iteration tend to be noisily monotonic.

4.4.1.2.2 Exercise. *Completely positive semidefinite matrix.* [41]

Given rank-2 positive semidefinite matrix $G = \begin{bmatrix} 0.50 & 0.55 & 0.20 \\ 0.55 & 0.61 & 0.22 \\ 0.20 & 0.22 & 0.08 \end{bmatrix}$, find a positive factorization $G = X^{\mathrm{T}}X$ (985) by solving

$$\begin{array}{ll} \underset{X\in\mathbb{R}^{\mathbf{2}\times\mathbf{3}}}{\text{find}} & X \succeq \mathbf{0} \\ \text{subject to} & Z = \begin{bmatrix} I & X \\ X^{\mathrm{T}} & G \end{bmatrix} \succeq 0 \\ & \operatorname{rank} Z \leq 2 \end{array} \tag{779}$$

via convex iteration. ▼

4.4.1.2.3 Exercise. *Nonnegative matrix factorization.*

Given rank-2 nonnegative matrix $X = \begin{bmatrix} 17 & 28 & 42 \\ 16 & 47 & 51 \\ 17 & 82 & 72 \end{bmatrix}$, find a nonnegative factorization

$$X = WH \tag{780}$$

by solving

$$\begin{array}{ll} \underset{A\in\mathbb{S}^{\mathbf{3}},\, B\in\mathbb{S}^{\mathbf{3}},\, W\in\mathbb{R}^{\mathbf{3}\times\mathbf{2}},\, H\in\mathbb{R}^{\mathbf{2}\times\mathbf{3}}}{\text{find}} & W\,,\, H \\ \text{subject to} & Z = \begin{bmatrix} I & W^{\mathrm{T}} & H \\ W & A & X \\ H^{\mathrm{T}} & X^{\mathrm{T}} & B \end{bmatrix} \succeq 0 \\ & W \geq \mathbf{0} \\ & H \geq \mathbf{0} \\ & \operatorname{rank} Z \leq 2 \end{array} \tag{781}$$

which follows from the fact, at optimality,

$$Z^\star = \begin{bmatrix} I \\ W \\ H^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} I & W^{\mathrm{T}} & H \end{bmatrix} \tag{782}$$

Use the known closed-form solution for a direction vector Y to regulate rank by convex iteration; set $Z^\star = Q\Lambda Q^{\mathrm{T}} \in \mathbb{S}^{\mathbf{8}}$ to an ordered diagonalization and $U^\star = Q(:, 3{:}8) \in \mathbb{R}^{\mathbf{8}\times\mathbf{6}}$, then $Y = U^\star U^{\star\mathrm{T}}$ (§4.4.1.1).

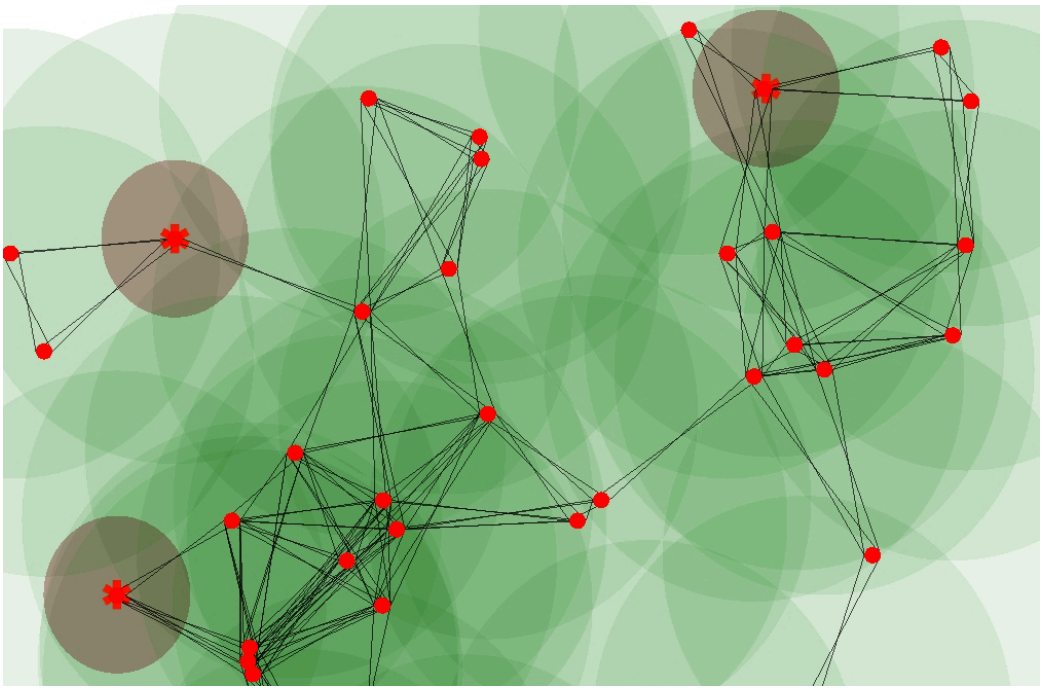


Figure 91: Sensor-network localization in $\mathbb{R}^{\mathbf{2}}$, illustrating connectivity and circular radio-range per *sensor*. Smaller dark grey regions each hold an *anchor* at their center; known fixed sensor positions. Sensor/anchor distance is measurable with negligible uncertainty for sensor within those grey regions. (Graphic by Geoff Merrett)

In summary, initialize Y then iterate numerical solution of (convex) semidefinite program

$$\begin{array}{ll} \underset{A\in\mathbb{S}^{\mathbf{3}},\, B\in\mathbb{S}^{\mathbf{3}},\, W\in\mathbb{R}^{\mathbf{3}\times\mathbf{2}},\, H\in\mathbb{R}^{\mathbf{2}\times\mathbf{3}}}{\text{minimize}} & \langle Z\,,\, Y\rangle \\ \text{subject to} & Z = \begin{bmatrix} I & W^{\mathrm{T}} & H \\ W & A & X \\ H^{\mathrm{T}} & X^{\mathrm{T}} & B \end{bmatrix} \succeq 0 \\ & W \geq \mathbf{0} \\ & H \geq \mathbf{0} \end{array} \tag{783}$$

with $Y = U^{\star}U^{\star\mathrm{T}}$ until convergence (which is global and occurs in very few iterations for this instance). ▼

Now, an application to optimal regulation of affine dimension:

4.4.1.2.4 Example. *Sensor-Network Localization and Wireless Location.*
Heuristic solution to a sensor-network localization problem, proposed by Carter, Jin,

Saunders, & Ye in [75],[4.27] is limited to two Euclidean dimensions and applies semidefinite programming (SDP) to little subproblems. There, a large network is partitioned into smaller subnetworks (as small as one *sensor* − a mobile point, whereabouts unknown) and then semidefinite programming and heuristics called SPASELOC are applied to localize each and every partition by two-dimensional distance geometry. Their partitioning procedure is one-pass, yet termed *iterative*; a term applicable only insofar as adjoining partitions can share localized sensors and *anchors* (absolute sensor positions known *a priori*). But there is no iteration on the entire network, hence the term "iterative" is perhaps inappropriate. As partitions are selected based on "rule sets" (heuristics, not geographics), they also term the partitioning *adaptive.* But no adaptation of a partition actually occurs once it has been determined.

One can reasonably argue that semidefinite programming methods are unnecessary for localization of small partitions of large sensor networks. [295] [90] In the past, these nonlinear localization problems were solved algebraically and computed by least squares solution to hyperbolic equations; called *multilateration.*[4.28] [244] [282] Indeed, practical contemporary numerical methods for global positioning (GPS) by satellite do not rely on convex optimization. [308]

Modern distance geometry is inextricably melded with semidefinite programming. The beauty of semidefinite programming, as relates to localization, lies in convex expression of classical multilateration: So & Ye showed [338] that the problem of finding unique solution, to a noiseless nonlinear system describing the common point of intersection of hyperspheres in real Euclidean vector space, can be expressed as a semidefinite program via distance geometry.

But the need for SDP methods in Carter & Jin *et alii* is enigmatic for two more reasons: 1) guessing solution to a partition whose intersensor measurement data or connectivity is inadequate for localization by distance geometry, 2) reliance on complicated and extensive heuristics for partitioning a large network that could instead be efficiently solved whole by one semidefinite program [240, §3]. While partitions range in size between 2 and 10 sensors, 5 sensors optimally, heuristics provided are only for two spatial dimensions (no higher-dimensional heuristics are proposed). For these small numbers it remains unclarified as to precisely what advantage is gained over traditional least squares: it is difficult to determine what part of their noise performance is attributable to SDP and what part is attributable to their heuristic geometry.

Partitioning of large sensor networks is a compromise to rapid growth of SDP computational intensity with problem size. But when impact of noise on distance measurement is of most concern, one is averse to a partitioning scheme because noise-effects vary inversely with problem size. [54, §2.2] (§5.13.2) Since an individual partition's solution is not iterated in Carter & Jin and is interdependent with adjoining partitions, we expect errors to propagate from one partition to the next; the ultimate partition solved, expected to suffer most.

Heuristics often fail on real-world data because of unanticipated circumstances.

[4.27] The paper constitutes Jin's dissertation for University of Toronto [232] although her name appears as second author. Ye's authorship is honorary.

[4.28] Multilateration − literally, *having many sides*; shape of a geometric figure formed by nearly intersecting lines of position. In navigation systems, therefore: Obtaining a *fix* from multiple lines of position. Multilateration can be regarded as noisy trilateration.

When heuristics fail, generally they are repaired by adding more heuristics. Tenuous is any presumption, for example, that distance measurement errors have distribution characterized by circular contours of equal probability about an unknown sensor-location. (Figure 91) That presumption effectively appears within Carter & Jin's optimization problem statement as affine equality constraints relating unknowns to distance measurements that are corrupted by noise. Yet in most all urban environments, this measurement noise is more aptly characterized by ellipsoids of varying orientation and eccentricity as one recedes from a sensor. (Figure 140) Each unknown sensor must therefore instead be bound to its own particular range of distance, primarily determined by the terrain.[4.29] The nonconvex problem we must instead solve is:

$$\begin{array}{ll} \underset{i\,,\,j\,\in\,\mathcal{I}}{\text{find}} & \{x_i\ ,\ x_j\} \\ \text{subject to} & \underline{d_{ij}} \le \|x_i - x_j\|^2 \le \overline{d_{ij}} \end{array} \tag{784}$$

where x_i represents sensor location, and where $\underline{d_{ij}}$ and $\overline{d_{ij}}$ respectively represent lower and upper bounds on measured distance-square from i^{th} to j^{th} sensor (or from sensor to anchor). Figure 96 illustrates contours of equal sensor-location uncertainty. By establishing these individual upper and lower bounds, orientation and eccentricity can effectively be incorporated into the problem statement.

Generally speaking, there can be no unique solution to the sensor-network localization problem because there is no unique formulation; that is the art of Optimization. Any optimal solution obtained depends on whether or how a network is partitioned, whether distance data is complete, presence of noise, and how the problem is formulated. When a particular formulation is a convex optimization problem, then the set of all optimal solutions forms a convex set containing the actual or true localization. Measurement noise precludes equality constraints representing distance. The optimal solution set is consequently expanded; necessitated by introduction of distance inequalities admitting more and higher-rank solutions. Even were the optimal solution set a single point, it is not necessarily the true localization because there is little hope of exact localization by any algorithm once significant noise is introduced.

Carter & Jin gauge performance of their heuristics to the SDP formulation of author Biswas whom they regard as vanguard to the art. [15, §1] Biswas posed localization as an optimization problem minimizing a distance measure. [48] [46] Intuitively, minimization of any distance measure yields compacted solutions; (*confer* §6.7.0.0.1) precisely the anomaly motivating Carter & Jin. Their two-dimensional heuristics outperformed Biswas' localizations both in execution-time and proximity to the desired result. Perhaps, instead of heuristics, Biswas' approach to localization can be improved: [45] [47].

The sensor-network localization problem is considered difficult. [15, §2] Rank constraints in optimization are considered more difficult. Control of affine dimension in Carter & Jin is suboptimal because of implicit projection on $\mathbb{R}^2$. In what follows, we present the localization problem as a semidefinite program (equivalent to (784)) having an explicit rank constraint which controls affine dimension of an optimal solution. We show how to achieve that rank constraint only if the feasible set contains a matrix of desired rank. Our problem formulation is extensible to any spatial dimension.

[4.29] A distinct contour map corresponding to each anchor is required in practice.

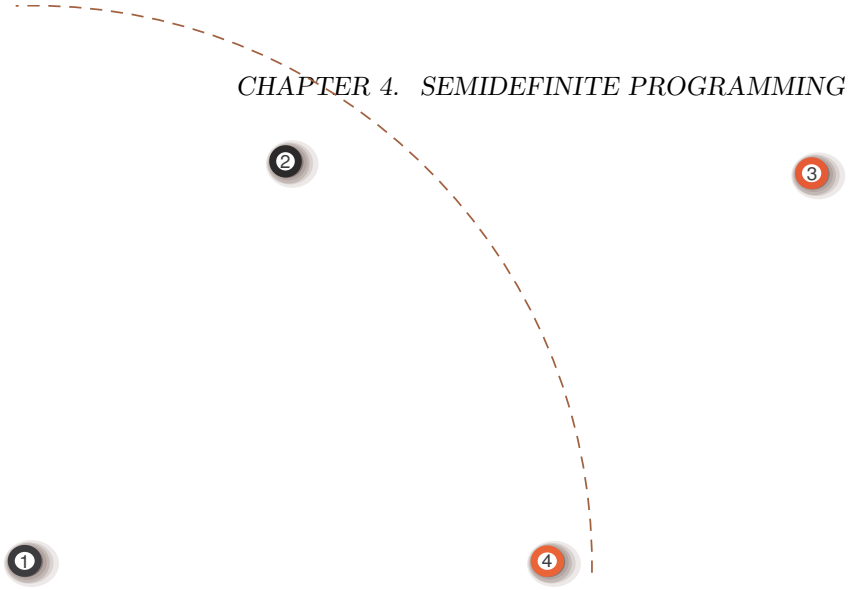


Figure 92: 2-lattice in $\mathbb{R}^2$, hand-drawn. Nodes 3 and 4 are anchors; remaining nodes are sensors. Radio range of sensor 1 indicated by arc.

proposed standardized test

Jin proposes an academic test in two-dimensional real Euclidean space $\mathbb{R}^2$ that we adopt. In essence, this test is a localization of sensors and anchors arranged in a regular triangular lattice. Lattice connectivity is solely determined by sensor radio range; a connectivity graph is assumed incomplete. In the interest of test standardization, we propose adoption of a few small examples: Figure **92** through Figure **95** and their particular connectivity represented by matrices (785) through (788) respectively.

$$\begin{array}{cccc} 0 & \bullet & ? & \bullet \\ \bullet & 0 & \bullet & \bullet \\ ? & \bullet & 0 & \circ \\ \bullet & \bullet & \circ & 0 \end{array} \tag{785}$$

Matrix entries *dot* $\bullet$ indicate measurable distance between *nodes* while unknown distance is denoted by ? (*question mark*). Matrix entries *hollow dot* $\circ$ represent known distance between anchors (to high accuracy) while zero distance is denoted 0. Because measured distances are quite unreliable in practice, our solution to the localization problem substitutes a distinct range of possible distance for each measurable distance; equality constraints exist only for anchors.

Anchors are chosen so as to increase difficulty for algorithms dependent on existence of sensors in their convex hull. The challenge is to find a solution in two dimensions close to the true sensor positions given incomplete noisy intersensor distance information.

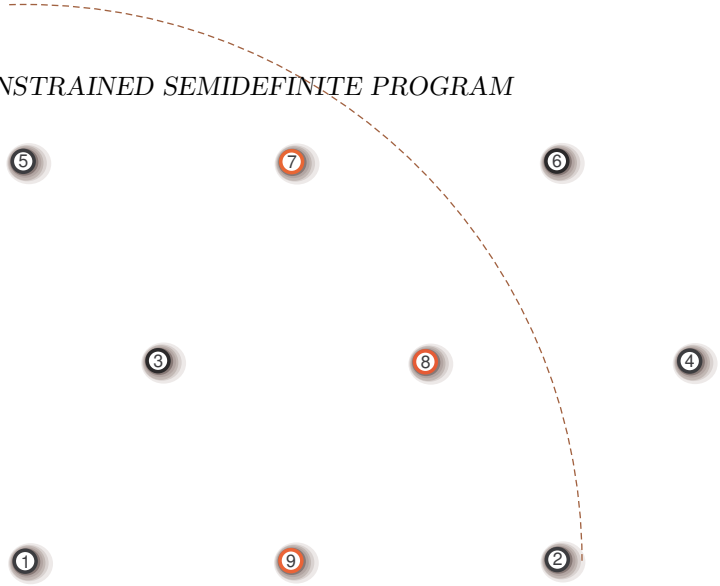


Figure 93: 3-lattice in $\mathbb{R}^2$, hand-drawn. Nodes 7, 8, and 9 are anchors; remaining nodes are sensors. Radio range of sensor 1 indicated by arc.

$$\begin{array}{ccccccccc}
0 & \bullet & \bullet & ? & \bullet & ? & ? & \bullet & \bullet \\
\bullet & 0 & \bullet & \bullet & ? & \bullet & ? & \bullet & \bullet \\
\bullet & \bullet & 0 & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\
? & \bullet & \bullet & 0 & ? & \bullet & \bullet & \bullet & \bullet \\
\bullet & ? & \bullet & ? & 0 & \bullet & \bullet & \bullet & \bullet \\
? & \bullet & \bullet & \bullet & \bullet & 0 & \bullet & \bullet & \bullet \\
? & ? & \bullet & \bullet & \bullet & \bullet & 0 & \circ & \circ \\
\bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \circ & 0 & \circ \\
\bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \circ & \circ & 0
\end{array} \tag{786}$$

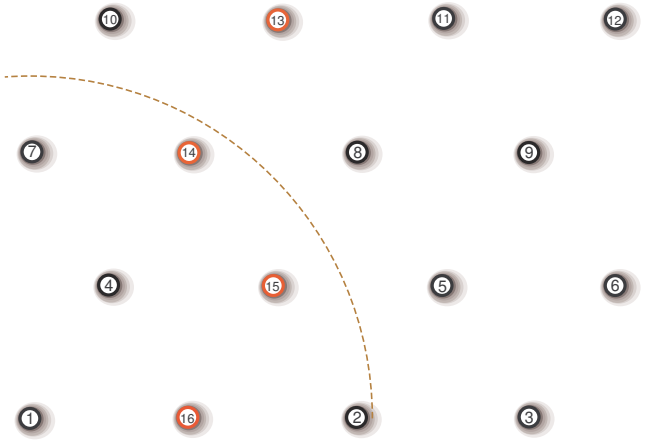


Figure 94: 4-lattice in $\mathbb{R}^2$, hand-drawn. Nodes 13, 14, 15, and 16 are anchors; remaining nodes are sensors. Radio range of sensor 1 indicated by arc.

$$
\begin{array}{cccccccccccccccc}
0 & ? & ? & \bullet & ? & ? & \bullet & ? & ? & ? & ? & ? & ? & ? & \bullet & \bullet \\
? & 0 & \bullet & \bullet & \bullet & \bullet & ? & \bullet & ? & ? & ? & ? & ? & \bullet & \bullet & \bullet \\
? & \bullet & 0 & ? & \bullet & \bullet & ? & ? & \bullet & ? & ? & ? & ? & ? & \bullet & \bullet \\
\bullet & \bullet & ? & 0 & \bullet & ? & \bullet & \bullet & ? & \bullet & ? & ? & \bullet & \bullet & \bullet & \bullet \\
? & \bullet & \bullet & \bullet & 0 & \bullet & ? & \bullet & \bullet & ? & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\
? & \bullet & \bullet & ? & \bullet & 0 & ? & \bullet & \bullet & ? & \bullet & \bullet & ? & ? & ? & ? \\
\bullet & ? & ? & \bullet & ? & ? & 0 & ? & ? & \bullet & ? & ? & \bullet & \bullet & \bullet & \bullet \\
? & \bullet & ? & \bullet & \bullet & \bullet & ? & 0 & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\
? & ? & \bullet & ? & \bullet & \bullet & ? & \bullet & 0 & ? & \bullet & \bullet & \bullet & ? & \bullet & ? \\
? & ? & ? & \bullet & ? & ? & \bullet & \bullet & ? & 0 & \bullet & ? & \bullet & \bullet & \bullet & ? \\
? & ? & ? & ? & \bullet & \bullet & ? & \bullet & \bullet & \bullet & 0 & \bullet & \bullet & \bullet & \bullet & ? \\
? & ? & ? & ? & \bullet & \bullet & ? & \bullet & \bullet & ? & \bullet & 0 & ? & ? & ? & ? \\
? & ? & ? & \bullet & \bullet & ? & \bullet & \bullet & \bullet & \bullet & \bullet & ? & 0 & \circ & \circ & \circ \\
? & \bullet & ? & \bullet & \bullet & ? & \bullet & \bullet & ? & \bullet & \bullet & ? & \circ & 0 & \circ & \circ \\
\bullet & \bullet & \bullet & \bullet & \bullet & ? & \bullet & \bullet & \bullet & \bullet & \bullet & ? & \circ & \circ & 0 & \circ \\
\bullet & \bullet & \bullet & \bullet & \bullet & ? & \bullet & \bullet & ? & ? & ? & ? & \circ & \circ & \circ & 0
\end{array}
\tag{787}
$$

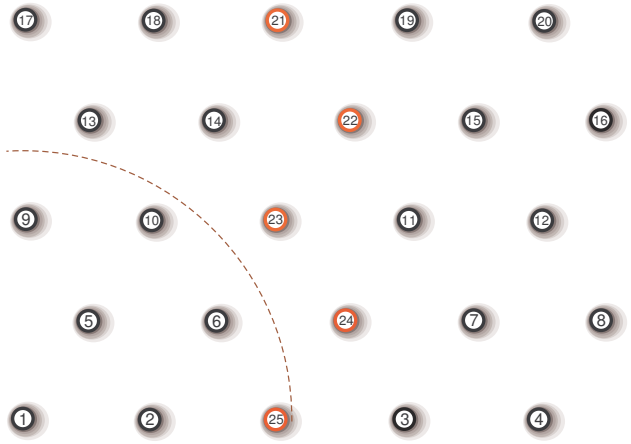


Figure 95: 5-lattice in $\mathbb{R}^2$. Nodes 21 through 25 are anchors.

$$
\begin{array}{ccccccccccccccccccccccc}
0 & \bullet & ? & ? & \bullet & \bullet & ? & ? & \bullet & ? & ? & ? & ? & ? & ? & ? & ? & ? & ? & ? & ? & ? & ? & ? \\
\bullet & 0 & ? & ? & \bullet & \bullet & ? & ? & ? & \bullet & ? & ? & ? & ? & ? & ? & ? & ? & ? & ? & ? & ? & \bullet & \bullet \\
? & ? & 0 & \bullet & ? & \bullet & \bullet & \bullet & ? & ? & \bullet & \bullet & ? & ? & ? & ? & ? & ? & ? & ? & ? & \bullet & \bullet & \bullet \\
? & ? & \bullet & 0 & ? & ? & \bullet & \bullet & ? & ? & ? & \bullet & ? & ? & ? & ? & ? & ? & ? & ? & ? & ? & \bullet & ? \\
\bullet & \bullet & ? & ? & 0 & \bullet & ? & ? & \bullet & \bullet & ? & ? & \bullet & \bullet & ? & ? & \bullet & ? & ? & ? & ? & ? & \bullet & ? & \bullet \\
\bullet & \bullet & \bullet & ? & \bullet & 0 & \bullet & ? & \bullet & \bullet & \bullet & ? & ? & \bullet & ? & ? & ? & ? & ? & ? & ? & ? & \bullet & \bullet & \bullet \\
? & ? & \bullet & \bullet & ? & \bullet & 0 & \bullet & ? & ? & \bullet & \bullet & ? & ? & \bullet & \bullet & ? & ? & ? & ? & ? & ? & \bullet & \bullet & \bullet \\
? & ? & \bullet & \bullet & ? & ? & \bullet & 0 & ? & ? & \bullet & \bullet & ? & ? & \bullet & \bullet & ? & ? & ? & ? & ? & ? & ? & \bullet & ? \\
\bullet & ? & ? & ? & \bullet & \bullet & ? & ? & 0 & \bullet & ? & ? & \bullet & \bullet & ? & ? & \bullet & \bullet & ? & ? & ? & ? & ? & ? & ? \\
? & \bullet & ? & ? & \bullet & \bullet & ? & ? & \bullet & 0 & \bullet & ? & \bullet & \bullet & ? & ? & ? & \bullet & ? & ? & \bullet & \bullet & \bullet & \bullet & \bullet \\
? & ? & \bullet & ? & ? & \bullet & \bullet & \bullet & ? & \bullet & 0 & \bullet & ? & \bullet & \bullet & \bullet & ? & ? & \bullet & ? & ? & \bullet & \bullet & \bullet & \bullet \\
? & ? & \bullet & \bullet & ? & ? & \bullet & \bullet & ? & ? & \bullet & 0 & ? & ? & \bullet & \bullet & ? & ? & \bullet & \bullet & ? & \bullet & \bullet & \bullet & ? \\
? & ? & ? & ? & \bullet & ? & ? & ? & \bullet & \bullet & ? & ? & 0 & \bullet & ? & ? & \bullet & \bullet & ? & ? & \bullet & \bullet & ? & ? & ? \\
? & ? & ? & ? & \bullet & \bullet & ? & ? & \bullet & \bullet & \bullet & ? & \bullet & 0 & \bullet & ? & \bullet & \bullet & \bullet & ? & \bullet & \bullet & \bullet & \bullet & ? \\
? & ? & ? & ? & ? & ? & \bullet & \bullet & ? & ? & \bullet & \bullet & ? & \bullet & 0 & \bullet & ? & ? & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & ? \\
? & ? & ? & ? & ? & ? & \bullet & \bullet & ? & ? & \bullet & \bullet & ? & ? & \bullet & 0 & ? & ? & \bullet & \bullet & ? & \bullet & ? & ? & ? \\
? & ? & ? & ? & \bullet & ? & ? & ? & \bullet & ? & ? & ? & \bullet & \bullet & ? & ? & 0 & \bullet & ? & ? & \bullet & ? & ? & ? & ? \\
? & ? & ? & ? & ? & ? & ? & ? & \bullet & \bullet & ? & ? & \bullet & \bullet & ? & ? & \bullet & 0 & \bullet & ? & \bullet & \bullet & \bullet & ? & ? \\
? & ? & ? & ? & ? & ? & ? & ? & ? & \bullet & \bullet & ? & \bullet & \bullet & \bullet & ? & \bullet & 0 & \bullet & \bullet & \bullet & \bullet & ? & ? \\
? & ? & ? & ? & ? & ? & ? & ? & ? & ? & \bullet & ? & ? & \bullet & \bullet & ? & ? & \bullet & 0 & \bullet & \bullet & ? & ? & ? \\
? & ? & ? & ? & ? & ? & ? & ? & ? & \bullet & ? & ? & \bullet & \bullet & \bullet & ? & \bullet & \bullet & \bullet & 0 & \circ & \circ & \circ & \circ \\
? & ? & ? & ? & ? & ? & ? & ? & ? & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet & ? & \bullet & \bullet & \bullet & \circ & 0 & \circ & \circ & \circ \\
? & ? & \bullet & ? & \bullet & \bullet & \bullet & ? & ? & \bullet & \bullet & \bullet & ? & \bullet & \bullet & ? & ? & \bullet & \bullet & ? & \circ & \circ & 0 & \circ & \circ \\
? & \bullet & \bullet & \bullet & ? & \bullet & \bullet & \bullet & ? & \bullet & \bullet & \bullet & ? & \bullet & \bullet & ? & ? & ? & ? & ? & \circ & \circ & \circ & 0 & \circ \\
? & \bullet & \bullet & ? & \bullet & \bullet & \bullet & ? & ? & \bullet & \bullet & ? & ? & ? & ? & ? & ? & ? & ? & ? & \circ & \circ & \circ & \circ & 0
\end{array}
\tag{788}
$$

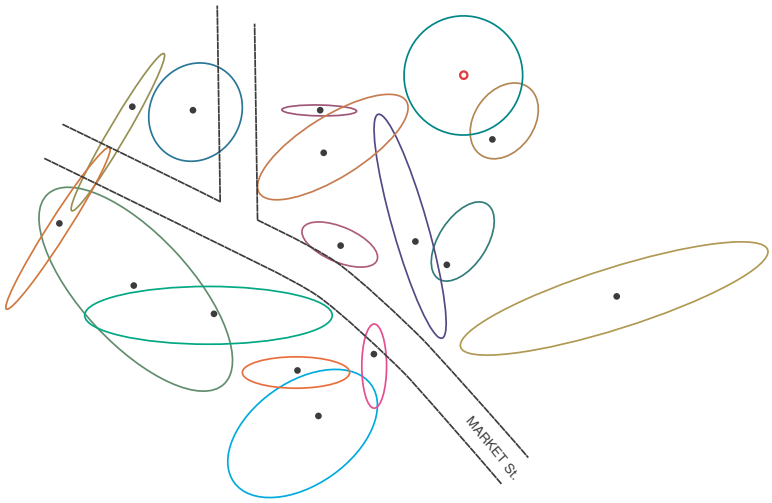


Figure 96: Location uncertainty ellipsoid in $\mathbb{R}^{\mathbf{2}}$ for each of 15 sensors $\bullet$ within three city blocks in downtown San Francisco. (Data by Polaris Wireless.)

problem statement

Ascribe points in a list $\{x_\ell \in \mathbb{R}^n,\ \ell = 1 \dots N\}$ to the columns of a matrix X;

$$X = [\, x_1 \ \cdots \ x_N \,] \in \mathbb{R}^{n \times N} \tag{76}$$

where N is regarded as cardinality of list X. Positive semidefinite matrix $X^{\mathrm{T}}X$, formed from inner product of the list, is a *Gram matrix*; [266, §3.6]

$$G = X^{\mathrm{T}}X = \begin{bmatrix} \|x_1\|^2 & x_1^{\mathrm{T}}x_2 & x_1^{\mathrm{T}}x_3 & \cdots & x_1^{\mathrm{T}}x_N \\ x_2^{\mathrm{T}}x_1 & \|x_2\|^2 & x_2^{\mathrm{T}}x_3 & \cdots & x_2^{\mathrm{T}}x_N \\ x_3^{\mathrm{T}}x_1 & x_3^{\mathrm{T}}x_2 & \|x_3\|^2 & \ddots & x_3^{\mathrm{T}}x_N \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ x_N^{\mathrm{T}}x_1 & x_N^{\mathrm{T}}x_2 & x_N^{\mathrm{T}}x_3 & \cdots & \|x_N\|^2 \end{bmatrix} \in \mathbb{S}_+^N \tag{985}$$

where $\mathbb{S}_+^N$ is the convex cone of $N \times N$ positive semidefinite matrices in the symmetric matrix subspace $\mathbb{S}^N$.

Existence of noise precludes measured distance from the input data. We instead assign measured distance to a range estimate specified by individual upper and lower bounds: $\overline{d_{ij}}$ is an upper bound on distance-square from i^{th} to j^{th} sensor, while $\underline{d_{ij}}$ is a lower bound. These bounds become the input data. Each measurement range is presumed different from the others because of measurement uncertainty; *e.g*, Figure **96**.

Our mathematical treatment of anchors and sensors is not dichotomized.[4.30] A sensor position that is known *a priori* to high accuracy (with absolute certainty) $\check{x}_i$ is called an *anchor*. Then the sensor-network localization problem (784) can be expressed equivalently: Given a number m of anchors and a set of indices $\mathcal{I}$ (corresponding to all measurable distances $\bullet$), for $0<n<N$

$$\begin{array}{ll}\underset{G\in\mathbb{S}^N,\ X\in\mathbb{R}^{n\times N}}{\text{find}} & X \\ \text{subject to} & \underline{d_{ij}} \le \langle G\ ,\ (e_i-e_j)(e_i-e_j)^{\mathrm{T}}\rangle \le \overline{d_{ij}} \qquad \forall (i,j)\in\mathcal{I} \\ & \langle G\ ,\ e_ie_i^{\mathrm{T}}\rangle = \|\check{x}_i\|^2\ , \quad i=N-m+1\ldots N \\ & \langle G\ ,\ (e_ie_j^{\mathrm{T}}+e_je_i^{\mathrm{T}})/2\rangle = \check{x}_i^{\mathrm{T}}\check{x}_j\ , \quad i<j\ , \quad \forall\, i,j\in\{N-m+1\ldots N\} \\ & X(:\,,N-m+1{:}N) = [\,\check{x}_{N-m+1}\ \cdots\ \check{x}_N\,] \\ & Z=\begin{bmatrix} I & X \\ X^{\mathrm{T}} & G\end{bmatrix} \succeq 0 \\ & \operatorname{rank} Z = n \end{array} \tag{789}$$

where e_i is the i^{th} member of the standard basis for $\mathbb{R}^N$. Distance-square

$$d_{ij} = \|x_i-x_j\|_2^2 = \langle x_i-x_j\ ,\ x_i-x_j\rangle \tag{972}$$

is related to Gram matrix entries $G\triangleq[g_{ij}]$ by vector inner-product

$$\begin{aligned} d_{ij} &= g_{ii}+g_{jj}-2g_{ij} \\ &= \langle G\ ,\ (e_i-e_j)(e_i-e_j)^{\mathrm{T}}\rangle = \operatorname{tr}(G^{\mathrm{T}}(e_i-e_j)(e_i-e_j)^{\mathrm{T}}) \end{aligned} \tag{987}$$

hence the scalar inequalities. Each linear equality constraint in $G\in\mathbb{S}^N$ represents a hyperplane in isometrically isomorphic Euclidean vector space $\mathbb{R}^{N(N+1)/2}$, while each linear inequality pair represents a convex Euclidean body known as *slab*.[4.31] By Schur complement (§A.4), any solution $(G\,,X)$ provides comparison with respect to the positive semidefinite cone

$$G\succeq X^{\mathrm{T}}X \tag{1025}$$

which is a convex relaxation of the desired equality constraint

$$\begin{bmatrix} I & X \\ X^{\mathrm{T}} & G\end{bmatrix} = \begin{bmatrix} I \\ X^{\mathrm{T}}\end{bmatrix}\begin{bmatrix} I & X\end{bmatrix} \tag{1026}$$

The rank constraint insures this equality holds, by Theorem A.4.0.1.3, thus restricting solution to $\mathbb{R}^n$. Assuming full-rank solution (list) X

$$\operatorname{rank} Z = \operatorname{rank} G = \operatorname{rank} X \tag{790}$$

[4.30] *Wireless location* problem thus stated identically; difference being: fewer sensors.

[4.31] an intersection of two parallel but opposing halfspaces (Figure 13). In terms of position X, this distance slab can be thought of as a thick *hypershell* instead of a hypersphere boundary.

convex equivalent problem statement

Problem statement (789) is nonconvex because of the rank constraint. We do not eliminate or ignore the rank constraint; rather, we find a convex way to enforce it: for $0<n<N$

$$\begin{array}{ll}\underset{G\in\mathbb{S}^N,\ X\in\mathbb{R}^{n\times N}}{\text{minimize}} & \langle Z\,,\,W\rangle \\ \text{subject to} & \underline{d_{ij}} \le \langle G\,,\,(e_i-e_j)(e_i-e_j)^{\mathrm{T}}\rangle \le \overline{d_{ij}} \qquad \forall (i,j)\in\mathcal{I} \\ & \langle G\,,\,e_ie_i^{\mathrm{T}}\rangle = \|\check{x}_i\|^2\,, \quad i=N-m+1\ldots N \\ & \langle G\,,\,(e_ie_j^{\mathrm{T}}+e_je_i^{\mathrm{T}})/2\rangle = \check{x}_i^{\mathrm{T}}\check{x}_j\,, \quad i<j\,, \quad \forall\, i,j\in\{N-m+1\ldots N\} \\ & X(:\,,N-m+1:N) = [\,\check{x}_{N-m+1}\ \cdots\ \check{x}_N\,] \\ & Z=\begin{bmatrix} I & X \\ X^{\mathrm{T}} & G\end{bmatrix} \succeq 0 \end{array} \tag{791}$$

Convex function $\operatorname{tr} Z$ is a well-known heuristic whose sole purpose is to represent convex envelope of $\operatorname{rank} Z$. (§7.2.2.1) In this convex optimization problem (791), a semidefinite program, we substitute a vector inner-product objective function for trace;

$$\operatorname{tr} Z \;=\; \langle Z\,,\,I\rangle \;\leftarrow\; \langle Z\,,\,W\rangle \tag{792}$$

a generalization of the trace heuristic for minimizing convex envelope of rank, where $W\in\mathbb{S}_+^{N+n}$ is constant with respect to (791). Matrix W is normal to a hyperplane in $\mathbb{S}^{N+n}$ minimized over a convex feasible set specified by the constraints in (791). Matrix W is chosen so $-W$ points in direction of rank-n feasible solutions G. For properly chosen W, problem (791) becomes an equivalent to (789). Thus the purpose of vector inner-product objective (792) is to locate a rank-n feasible Gram matrix assumed existent on the boundary of positive semidefinite cone $\mathbb{S}_+^N$, as explained beginning in §4.4.1; how to choose direction vector W is explained there and in what follows:

direction matrix W

Denote by $Z^\star$ an optimal composite matrix from semidefinite program (791). Then for $Z^\star\in\mathbb{S}^{N+n}$ whose eigenvalues $\lambda(Z^\star)\in\mathbb{R}^{N+n}$ are arranged in nonincreasing order, (Ky Fan)

$$\sum_{i=n+1}^{N+n}\lambda(Z^\star)_i \;=\; \begin{array}{ll}\underset{W\in\mathbb{S}^{N+n}}{\text{minimize}} & \langle Z^\star,\,W\rangle \\ \text{subject to} & 0\preceq W\preceq I \\ & \operatorname{tr} W = N\end{array} \tag{1800a}$$

which has an optimal solution that is known in closed form (p.567, §4.4.1.1). This eigenvalue sum is zero when $Z^\star$ has rank n or less.

Foreknowledge of optimal $Z^\star$, to make possible this search for W, implies iteration; *id est*, semidefinite program (791) is solved for $Z^\star$ initializing $W=I$ or $W=\mathbf{0}$. Once found, $Z^\star$ becomes constant in semidefinite program (1800a) where a new normal direction W is found as its optimal solution. Then this cycle (791) (1800a) iterates until convergence. When $\operatorname{rank} Z^\star=n$, solution via this convex iteration solves sensor-network localization problem (784) and its equivalent (789).

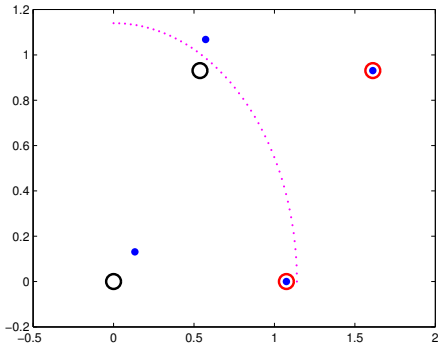


Figure 97: Typical solution for 2-lattice in Figure **92** with noise factor $\eta = 0.1$. Two red rightmost nodes are anchors; two remaining nodes are sensors. Radio range of sensor 1 indicated by arc; radius = 1.14. Actual sensor indicated by target ○ while its localization is indicated by bullet •. Rank-2 solution found in 1 iteration (791) (1800a) subject to reflection error.

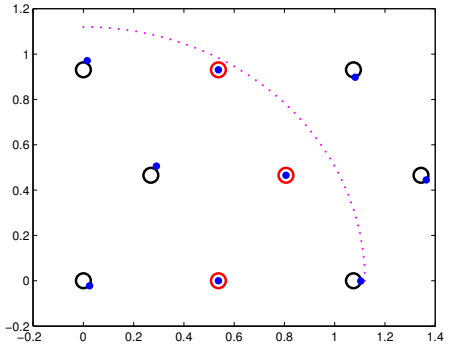


Figure 98: Typical solution for 3-lattice in Figure **93** with noise factor $\eta = 0.1$. Three red vertical middle nodes are anchors; remaining nodes are sensors. Radio range of sensor 1 indicated by arc; radius = 1.12. Actual sensor indicated by target ○ while its localization is indicated by bullet •. Rank-2 solution found in 2 iterations (791) (1800a).

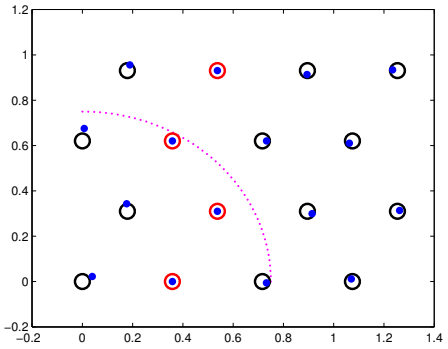


Figure 99: Typical solution for 4-lattice in Figure **94** with noise factor $\eta = 0.1$. Four red vertical middle-left nodes are anchors; remaining nodes are sensors. Radio range of sensor 1 indicated by arc; radius $= 0.75$. Actual sensor indicated by target $\bigcirc$ while its localization is indicated by bullet $\bullet$. Rank-2 solution found in 7 iterations (791) (1800a).

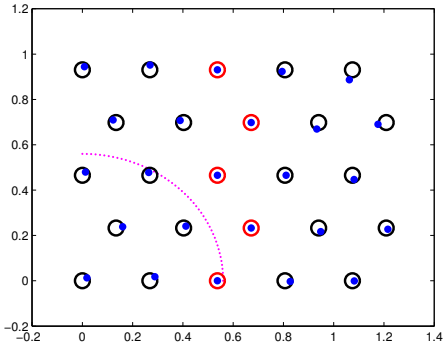


Figure 100: Typical solution for 5-lattice in Figure **95** with noise factor $\eta = 0.1$. Five red vertical middle nodes are anchors; remaining nodes are sensors. Radio range of sensor 1 indicated by arc; radius $= 0.56$. Actual sensor indicated by target $\bigcirc$ while its localization is indicated by bullet $\bullet$. Rank-2 solution found in 3 iterations (791) (1800a).

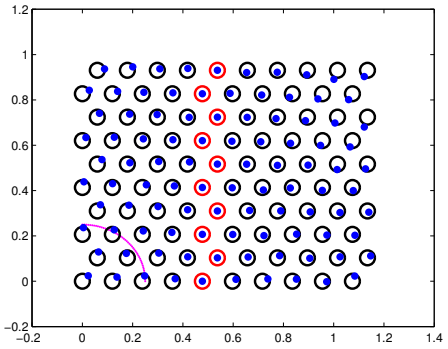


Figure 101: Typical solution for 10-lattice with noise factor $\eta = 0.1$ compares better than Carter & Jin [75, fig.4.2]. Ten red vertical middle nodes are anchors; the rest are sensors. Radio range of sensor 1 indicated by arc; radius $= 0.25$. Actual sensor indicated by target $\bigcirc$ while its localization is indicated by bullet $\bullet$. Rank-2 solution found in 5 iterations (791) (1800a).

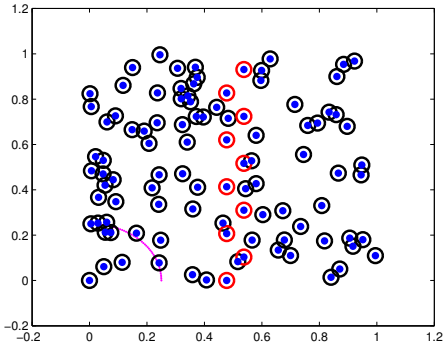


Figure 102: Typical localization of 100 randomized noiseless sensors ($\eta = 0$) is exact despite incomplete EDM. Ten red vertical middle nodes are anchors; remaining nodes are sensors. Radio range of sensor at origin indicated by arc; radius $= 0.25$. Actual sensor indicated by target $\bigcirc$ while its localization is indicated by bullet $\bullet$. Rank-2 solution found in 3 iterations (791) (1800a).

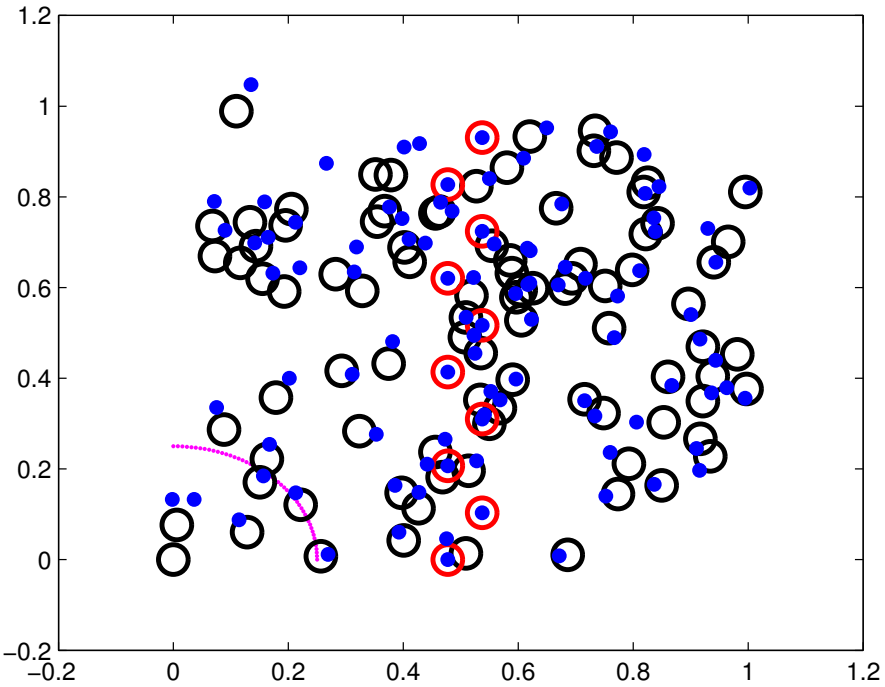


Figure 103: Typical solution for 100 randomized sensors with noise factor $\eta = 0.1$; worst measured average sensor error ≈ 0.0044 compares better than Carter & Jin's 0.0154 computed in 0.71s [75, p.19]. Ten red vertical middle nodes are anchors; same as before. Remaining nodes are sensors. Interior anchor placement makes localization difficult. Radio range of sensor at origin indicated by arc; radius $= 0.25$. Actual sensor indicated by target ○ while its localization is indicated by bullet •. After 1 iteration $\operatorname{rank} G = 92$, after 2 iterations $\operatorname{rank} G = 4$. Rank-2 solution found in 3 iterations (791) (1800a). (Regular lattice in Figure **101** is actually harder to solve, requiring more iterations.) Runtime for SDPT3 [367] under `cvx` [183] is a few minutes on 2009 vintage laptop Core 2 Duo CPU (Intel T6400@2GHz, 800MHz FSB).

numerical solution

In all examples to follow, number of anchors

$$m = \sqrt{N} \tag{793}$$

equals square root of cardinality N of list X. Indices set $\mathcal{I}$ identifying all measurable distances $\bullet$ is ascertained from connectivity matrix (785), (786), (787), or (788). We solve iteration (791) (1800a) in dimension $n = 2$ for each respective example illustrated in Figure **92** through Figure **95**.

In presence of negligible noise, true position is reliably localized for every standardized example; noteworthy insofar as each example represents an incomplete graph. This implies that the set of all optimal solutions having least rank must be small.

To make the examples interesting and consistent with previous work, we randomize each range of distance-square that bounds $\langle G\,,\,(e_i-e_j)(e_i-e_j)^{\mathrm{T}}\rangle$ in (791); *id est*, for each and every $(i,j)\in\mathcal{I}$

$$\begin{aligned}\overline{d_{ij}} &= d_{ij}(1+\sqrt{3}\,\eta\,\chi_l)^2\\ \underline{d_{ij}} &= d_{ij}(1-\sqrt{3}\,\eta\,\chi_{l+1})^2\end{aligned} \tag{794}$$

where $\eta = 0.1$ is a constant noise factor, χ_l is the l^{th} sample of a noise process realization uniformly distributed in the interval $(0,1)$ like `rand(1)` from MATLAB, and d_{ij} is actual distance-square from i^{th} to j^{th} sensor. Because of distinct function calls to `rand()`, each range of distance-square $[\,\underline{d_{ij}}\,,\,\overline{d_{ij}}\,]$ is not necessarily centered on actual distance-square d_{ij}. Unit stochastic variance is provided by factor $\sqrt{3}$.

Figure **97** through Figure **100** each illustrate one realization of numerical solution to the standardized lattice problems posed by Figure **92** through Figure **95** respectively. Exact localization, by any method, is impossible because of measurement noise. Certainly, by inspection of their published graphical data, our results are better than those of Carter & Jin. (Figure **101**, **102**, **103**) Obviously our solutions do not suffer from those compaction-type errors (clustering of localized sensors) exhibited by Biswas' graphical results for the same noise factor η.

localization example conclusion

Solution to this sensor-network localization problem became apparent by understanding geometry of optimization. Trace of a matrix, to a student of linear algebra, is perhaps a sum of eigenvalues. But to us, trace represents the normal I to some hyperplane in Euclidean vector space. (Figure **90**)

Our solutions are globally optimal, requiring: 1) no centralized-gradient postprocessing heuristic refinement as in [45] because there is effectively no relaxation of (789) at global optimality, 2) no implicit postprojection on rank-2 positive semidefinite matrices induced by nonzero $G-X^{\mathrm{T}}X$ denoting suboptimality as occurs in [46] [47] [48] [75] [232] [240]; indeed, $G^\star = X^{\star\mathrm{T}}X^\star$ by convex iteration.

Numerical solution to noisy problems, containing sensor variables well in excess of 100, becomes difficult via the holistic semidefinite program we proposed. When problem size is within reach of contemporary general-purpose semidefinite program solvers, then

the convex iteration we presented inherently overcomes limitations of Carter & Jin with respect to both noise performance and ability to localize in any desired affine dimension.

The legacy of Carter, Jin, Saunders, & Ye [75] is a sobering demonstration of the need for more efficient methods for solution of semidefinite programs, while that of So & Ye [338] forever bonds distance geometry to semidefinite programming. Elegance of our semidefinite problem statement (791), for constraining affine dimension of sensor-network localization, should provide some *impetus* to focus more research on computational intensity of general-purpose semidefinite program solvers. An approach different from interior-point methods is required; higher speed and greater accuracy from a simplex-like solver is what is needed. □

4.4.1.2.5 Example. *Nonnegative spectral factorization.* (*confer* §3.8.1.0.2)
Having found optimal real coefficient vectors $v^\star, u^\star$ for a sixteenth order magnitude square transfer function, evaluated along the $\jmath\omega$ axis (p.229),

$$|H(\jmath\omega)|^2 = H(\jmath\omega)H(-\jmath\omega) = \frac{1 + v_1^\star\omega^2 + v_2^\star\omega^4 + \ldots + v_8^\star\omega^{16}}{1 + u_1^\star\omega^2 + u_2^\star\omega^4 + \ldots + u_8^\star\omega^{16}} \qquad (657)$$

we wish to find real coefficients $b\,, a$ for corresponding Fourier transform

$$H(\jmath\omega) = \frac{1 + b_1\jmath\omega + b_2(\jmath\omega)^2 + \ldots + b_8(\jmath\omega)^8}{1 + a_1\jmath\omega + a_2(\jmath\omega)^2 + \ldots + a_8(\jmath\omega)^8} \qquad (654)$$

These coefficients $b\,, a\,, v^\star, u^\star$ are related through simultaneous nonlinear algebraic equations:

$$\begin{array}{ll}
v_1^\star = b_1^2 - 2b_2\ , & u_1^\star = a_1^2 - 2a_2 \\
v_2^\star = b_2^2 - 2b_1b_3 + 2b_4\ , & u_2^\star = a_2^2 - 2a_1a_3 + 2a_4 \\
v_3^\star = b_3^2 - 2b_2b_4 + 2b_1b_5 - 2b_6\ , & u_3^\star = a_3^2 - 2a_2a_4 + 2a_1a_5 - 2a_6 \\
v_4^\star = b_4^2 - 2b_3b_5 + 2b_2b_6 - 2b_1b_7 + 2b_8\ , & u_4^\star = a_4^2 - 2a_3a_5 + 2a_2a_6 - 2a_1a_7 + 2a_8 \\
v_5^\star = b_5^2 - 2b_4b_6 + 2b_3b_7 - 2b_2b_8\ , & u_5^\star = a_5^2 - 2a_4a_6 + 2a_3a_7 - 2a_2a_8 \\
v_6^\star = b_6^2 - 2b_5b_7 + 2b_4b_8\ , & u_6^\star = a_6^2 - 2a_5a_7 + 2a_4a_8 \\
v_7^\star = b_7^2 - 2b_6b_8\ , & u_7^\star = a_7^2 - 2a_6a_8 \\
v_8^\star = b_8^2\ , & u_8^\star = a_8^2
\end{array} \qquad (795)$$

Define a rank-one matrix

$$G(b) \triangleq \begin{bmatrix} 1 \\ b \end{bmatrix}\begin{bmatrix} 1 & b^{\mathrm{T}} \end{bmatrix} = \begin{bmatrix}
1 & b_1 & b_2 & b_3 & b_4 & b_5 & b_6 & b_7 & b_8 \\
b_1 & b_1^2 & b_1b_2 & b_1b_3 & b_1b_4 & b_1b_5 & b_1b_6 & b_1b_7 & b_1b_8 \\
b_2 & b_1b_2 & b_2^2 & b_2b_3 & b_2b_4 & b_2b_5 & b_2b_6 & b_2b_7 & b_2b_8 \\
b_3 & b_1b_3 & b_2b_3 & b_3^2 & b_3b_4 & b_3b_5 & b_3b_6 & b_3b_7 & b_3b_8 \\
b_4 & b_1b_4 & b_2b_4 & b_3b_4 & b_4^2 & b_4b_5 & b_4b_6 & b_4b_7 & b_4b_8 \\
b_5 & b_1b_5 & b_2b_5 & b_3b_5 & b_4b_5 & b_5^2 & b_5b_6 & b_5b_7 & b_5b_8 \\
b_6 & b_1b_6 & b_2b_6 & b_3b_6 & b_4b_6 & b_5b_6 & b_6^2 & b_6b_7 & b_6b_8 \\
b_7 & b_1b_7 & b_2b_7 & b_3b_7 & b_4b_7 & b_5b_7 & b_6b_7 & b_7^2 & b_7b_8 \\
b_8 & b_1b_8 & b_2b_8 & b_3b_8 & b_4b_8 & b_5b_8 & b_6b_8 & b_7b_8 & b_8^2
\end{bmatrix} \in \mathbb{S}^{\mathbf{9}} \qquad (796)$$

(Matrix $G(a)$ is similarly defined.) Observe that $v^\star$ in (795) is formed by summing antidiagonals of $G(b)$ whose entries alternate sign. A particular sum is specified by a

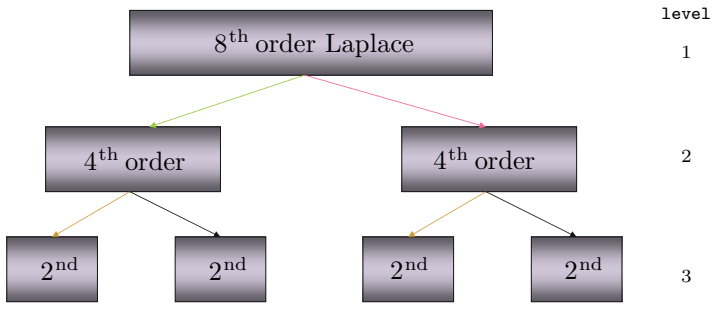


Figure 104: Nonnegative spectral factorization, high order bisection strategy. $\eta = 8^{\text{th}}$ order Laplace transform corresponds to $2\eta = 16^{\text{th}}$ order magnitude square transfer function. Because numerator v and denominator u are factored separately, number of factorizations $= 2(\log_2(\eta) - 1)$. In the text, double dots $\ddot{v}\,, \ddot{u}$ connote first bifurcation (`level` 2). Triple dots $\dddot{v}\,, \dddot{u}$ connote second bifurcations (`level` 3). Factors per level $= 2^{\texttt{level}-1}$.

predetermined symmetric constant matrix A_i (*confer* (57)) from a set $\{A_i \in \mathbb{S}^{\mathbf{9}},\ i = 1 \dots 8\}$. With

$$A = \begin{bmatrix} \operatorname{svec}(A_1)^{\mathrm{T}} \\ \vdots \\ \operatorname{svec}(A_8)^{\mathrm{T}} \end{bmatrix} \in \mathbb{R}^{\mathbf{8}\times\mathbf{9}(\mathbf{9}+\mathbf{1})/\mathbf{2}} \qquad (688)$$

as previously defined in §4.1.1, all the sums (795) may be stated as two linear equalities $A \operatorname{svec} G(b) = v^\star$ and $A \operatorname{svec} G(a) = u^\star$. Then the problem of finding coefficients b may be stated as a feasibility problem[4.32]

$$\begin{array}{ll} \underset{G \in \mathbb{S}^{\mathbf{9}}}{\text{find}} & b \in \mathbb{R}^{\mathbf{8}} \\ \text{subject to} & A \operatorname{svec} G = v^\star \\ & \begin{bmatrix} 1 \\ b \end{bmatrix} = G(:,1) \\ & b \succeq 0 \\ & (G \succeq 0) \\ & \operatorname{rank} G = 1 \end{array} \qquad (797)$$

The rank-one constraint is handled by convex iteration, as explained in §4.4.1. Positive semidefiniteness is parenthetical here because, for rank-one matrices, symmetry is necessary and sufficient (§A.3.1.0.7). □

[4.32] separately from the similar optimization problem to find vector a. Stability requires $a \succeq 0$ with additional constraints on a. Minimum phase requires $b \succeq 0$ plus more constraints on b that are missing from problem statement (797). Both stability and minimum phase may be enforced, subsequent to spectral factorization, by negating positive real parts of poles and zeros respectively in order to move them into the left half (Laplace) s-plane with no impact to $|H(\jmath\omega)|$.

4.4.1.2.6 Example. *Nonnegative spectral factorization* II.
The purpose of spectral factorization, in electronics, is to facilitate high order filter implementation in the form of passive and active circuitry. Cascades of second order (Laplace) sections are preferred because component sensitivity becomes manageable and because needed complex poles and zeros cannot be obtained from a first order section.

Nonnegative spectral factorization on a magnitude square transfer function, evaluated along the $\jmath\omega$ axis, was performed in Example 4.4.1.2.5 to recover its corresponding Fourier transform.[4.33] In this example, we nonnegatively decompose a high order magnitude square transfer function into a product of successively lower order magnitude square transfer functions. Once fourth order magnitude square functions are found, then corresponding second order Laplace transfer function coefficients are ascertained from (656) and then passive component values can be determined from those coefficients.

Our strategy, for an eighth order Laplace transfer function, is illustrated in Figure **104**. We begin at the tree's `level` 2 factorization. Nonnegative decomposition of a 16^{th} order magnitude square transfer function into two 8^{th} order functions

$$\frac{1+v_1^\star\omega^2+v_2^\star\omega^4+\ldots+v_8^\star\omega^{16}}{1+u_1^\star\omega^2+u_2^\star\omega^4+\ldots+u_8^\star\omega^{16}} = \frac{1+\ddot{v}_1\omega^2+\ddot{v}_2\omega^4+\ddot{v}_3\omega^6+\ddot{v}_4\omega^8}{1+\ddot{u}_1\omega^2+\ddot{u}_2\omega^4+\ddot{u}_3\omega^6+\ddot{u}_4\omega^8}\;\frac{1+\ddot{v}_5\omega^2+\ddot{v}_6\omega^4+\ddot{v}_7\omega^6+\ddot{v}_8\omega^8}{1+\ddot{u}_5\omega^2+\ddot{u}_6\omega^4+\ddot{u}_7\omega^6+\ddot{u}_8\omega^8} \tag{798}$$

implies these simultaneous algebraic identifications with known real coefficient vectors $v^\star, u^\star$:

$$\begin{array}{ll}
v_1^\star = \ddot{v}_1+\ddot{v}_5\ , & u_1^\star = \ddot{u}_1+\ddot{u}_5 \\
v_2^\star = \ddot{v}_2+\ddot{v}_6+\ddot{v}_1\ddot{v}_5\ , & u_2^\star = \ddot{u}_2+\ddot{u}_6+\ddot{u}_1\ddot{u}_5 \\
v_3^\star = \ddot{v}_3+\ddot{v}_7+\ddot{v}_1\ddot{v}_6+\ddot{v}_2\ddot{v}_5\ , & u_3^\star = \ddot{u}_3+\ddot{u}_7+\ddot{u}_1\ddot{u}_6+\ddot{u}_2\ddot{u}_5 \\
v_4^\star = \ddot{v}_4+\ddot{v}_8+\ddot{v}_1\ddot{v}_7+\ddot{v}_2\ddot{v}_6+\ddot{v}_3\ddot{v}_5\ , & u_4^\star = \ddot{u}_4+\ddot{u}_8+\ddot{u}_1\ddot{u}_7+\ddot{u}_2\ddot{u}_6+\ddot{u}_3\ddot{u}_5 \\
v_5^\star = \ddot{v}_4\ddot{v}_5+\ddot{v}_3\ddot{v}_6+\ddot{v}_2\ddot{v}_7+\ddot{v}_1\ddot{v}_8\ , & u_5^\star = \ddot{u}_4\ddot{u}_5+\ddot{u}_3\ddot{u}_6+\ddot{u}_2\ddot{u}_7+\ddot{u}_1\ddot{u}_8 \\
v_6^\star = \ddot{v}_4\ddot{v}_6+\ddot{v}_3\ddot{v}_7+\ddot{v}_2\ddot{v}_8\ , & u_6^\star = \ddot{u}_4\ddot{u}_6+\ddot{u}_3\ddot{u}_7+\ddot{u}_2\ddot{u}_8 \\
v_7^\star = \ddot{v}_4\ddot{v}_7+\ddot{v}_3\ddot{v}_8\ , & u_7^\star = \ddot{u}_4\ddot{u}_7+\ddot{u}_3\ddot{u}_8 \\
v_8^\star = \ddot{v}_4\ddot{v}_8\ , & u_8^\star = \ddot{u}_4\ddot{u}_8
\end{array} \tag{799}$$

Now define a rank-one matrix for the numerator

$$G(\ddot{v}) \triangleq \begin{bmatrix} 1 \\ \ddot{v} \end{bmatrix}\begin{bmatrix} 1 & \ddot{v}^{\mathrm{T}} \end{bmatrix} = \begin{bmatrix}
1 & \ddot{v}_1 & \ddot{v}_2 & \ddot{v}_3 & \ddot{v}_4 & \ddot{v}_5 & \ddot{v}_6 & \ddot{v}_7 & \ddot{v}_8 \\
\ddot{v}_1 & \ddot{v}_1^2 & \ddot{v}_1\ddot{v}_2 & \ddot{v}_1\ddot{v}_3 & \ddot{v}_1\ddot{v}_4 & \ddot{v}_1\ddot{v}_5 & \ddot{v}_1\ddot{v}_6 & \ddot{v}_1\ddot{v}_7 & \ddot{v}_1\ddot{v}_8 \\
\ddot{v}_2 & \ddot{v}_1\ddot{v}_2 & \ddot{v}_2^2 & \ddot{v}_2\ddot{v}_3 & \ddot{v}_2\ddot{v}_4 & \ddot{v}_2\ddot{v}_5 & \ddot{v}_2\ddot{v}_6 & \ddot{v}_2\ddot{v}_7 & \ddot{v}_2\ddot{v}_8 \\
\ddot{v}_3 & \ddot{v}_1\ddot{v}_3 & \ddot{v}_2\ddot{v}_3 & \ddot{v}_3^2 & \ddot{v}_3\ddot{v}_4 & \ddot{v}_3\ddot{v}_5 & \ddot{v}_3\ddot{v}_6 & \ddot{v}_3\ddot{v}_7 & \ddot{v}_3\ddot{v}_8 \\
\ddot{v}_4 & \ddot{v}_1\ddot{v}_4 & \ddot{v}_2\ddot{v}_4 & \ddot{v}_3\ddot{v}_4 & \ddot{v}_4^2 & \ddot{v}_4\ddot{v}_5 & \ddot{v}_4\ddot{v}_6 & \ddot{v}_4\ddot{v}_7 & \ddot{v}_4\ddot{v}_8 \\
\ddot{v}_5 & \ddot{v}_1\ddot{v}_5 & \ddot{v}_2\ddot{v}_5 & \ddot{v}_3\ddot{v}_5 & \ddot{v}_4\ddot{v}_5 & \ddot{v}_5^2 & \ddot{v}_5\ddot{v}_6 & \ddot{v}_5\ddot{v}_7 & \ddot{v}_5\ddot{v}_8 \\
\ddot{v}_6 & \ddot{v}_1\ddot{v}_6 & \ddot{v}_2\ddot{v}_6 & \ddot{v}_3\ddot{v}_6 & \ddot{v}_4\ddot{v}_6 & \ddot{v}_5\ddot{v}_6 & \ddot{v}_6^2 & \ddot{v}_6\ddot{v}_7 & \ddot{v}_6\ddot{v}_8 \\
\ddot{v}_7 & \ddot{v}_1\ddot{v}_7 & \ddot{v}_2\ddot{v}_7 & \ddot{v}_3\ddot{v}_7 & \ddot{v}_4\ddot{v}_7 & \ddot{v}_5\ddot{v}_7 & \ddot{v}_6\ddot{v}_7 & \ddot{v}_7^2 & \ddot{v}_7\ddot{v}_8 \\
\ddot{v}_8 & \ddot{v}_1\ddot{v}_8 & \ddot{v}_2\ddot{v}_8 & \ddot{v}_3\ddot{v}_8 & \ddot{v}_4\ddot{v}_8 & \ddot{v}_5\ddot{v}_8 & \ddot{v}_6\ddot{v}_8 & \ddot{v}_7\ddot{v}_8 & \ddot{v}_8^2
\end{bmatrix} \in \mathbb{S}^{\mathbf{9}} \tag{800}$$

(Matrix $G(\ddot{u})$ is defined similarly for the denominator.) Terms in (799) are picked out of $G(\ddot{v})$ by a predetermined symmetric constant matrix $\ddot{A}_i$ (*confer* (57)) from a set

[4.33] When there are no poles on the $\jmath\omega$ axis, a Laplace transform can be recovered from a Fourier transform by substitution $\jmath\omega \leftarrow s$.

$\{\ddot{A}_i \in \mathbb{S}^{\mathbf{9}},\ i=1\dots 8\}$. Populating rows of

$$A = \begin{bmatrix} \operatorname{svec}(\ddot{A}_1)^{\mathrm{T}} \\ \vdots \\ \operatorname{svec}(\ddot{A}_8)^{\mathrm{T}} \end{bmatrix} \in \mathbb{R}^{\mathbf{8}\times\mathbf{9}(\mathbf{9}+\mathbf{1})/\mathbf{2}} \qquad (688)$$

with vectorized $\ddot{A}_i$ (as in §4.1.1), sums (799) are succinctly represented by two linear equalities $A \operatorname{svec} G(\ddot{v}) = v^\star$ and $A \operatorname{svec} G(\ddot{u}) = u^\star$. Then this spectral factorization in $\ddot{v}$ may be posed as a feasibility problem

$$\begin{array}{ll} \underset{G\in\mathbb{S}^{\mathbf{9}}}{\text{find}} & \ddot{v} \in \mathbb{R}^{\mathbf{8}} \\ \text{subject to} & A \operatorname{svec} G = v^\star \\ & \begin{bmatrix} 1 \\ \ddot{v} \end{bmatrix} = G(:,1) \\ & \ddot{v} \succeq 0 \\ & (G \succeq 0) \\ & \operatorname{rank} G = 1 \end{array} \qquad (801)$$

Having found two 8^{th} order square spectral factors in nonnegative $\ddot{v}^\star$ from (801), two pairs of 4^{th} order `level` 3 factors remain to be found:

$$\frac{1+\ddot{v}_1^\star\omega^2+\ddot{v}_2^\star\omega^4+\ddot{v}_3^\star\omega^6+\ddot{v}_4^\star\omega^8}{1+\ddot{u}_1^\star\omega^2+\ddot{u}_2^\star\omega^4+\ddot{u}_3^\star\omega^6+\ddot{u}_4^\star\omega^8} = \frac{1+\ddot{v}_1\omega^2+\ddot{v}_2\omega^4}{1+\ddot{u}_1\omega^2+\ddot{u}_2\omega^4}\ \frac{1+\ddot{v}_3\omega^2+\ddot{v}_4\omega^4}{1+\ddot{u}_3\omega^2+\ddot{u}_4\omega^4} \qquad (802)$$

$$\frac{1+\ddot{v}_5^\star\omega^2+\ddot{v}_6^\star\omega^4+\ddot{v}_7^\star\omega^6+\ddot{v}_8^\star\omega^8}{1+\ddot{u}_5^\star\omega^2+\ddot{u}_6^\star\omega^4+\ddot{u}_7^\star\omega^6+\ddot{u}_8^\star\omega^8} = \frac{1+\ddot{v}_5\omega^2+\ddot{v}_6\omega^4}{1+\ddot{u}_5\omega^2+\ddot{u}_6\omega^4}\ \frac{1+\ddot{v}_7\omega^2+\ddot{v}_8\omega^4}{1+\ddot{u}_7\omega^2+\ddot{u}_8\omega^4} \qquad (803)$$

$$\begin{array}{ll} \ddot{v}_1^\star = \ddot{v}_1 + \ddot{v}_3\ , & \ddot{u}_1^\star = \ddot{u}_1 + \ddot{u}_3 \\ \ddot{v}_2^\star = \ddot{v}_2 + \ddot{v}_4 + \ddot{v}_1\ddot{v}_3\ , & \ddot{u}_2^\star = \ddot{u}_2 + \ddot{u}_4 + \ddot{u}_1\ddot{u}_3 \\ \ddot{v}_3^\star = \ddot{v}_1\ddot{v}_4 + \ddot{v}_2\ddot{v}_3\ , & \ddot{u}_3^\star = \ddot{u}_1\ddot{u}_4 + \ddot{u}_2\ddot{u}_3 \\ \ddot{v}_4^\star = \ddot{v}_2\ddot{v}_4\ , & \ddot{u}_4^\star = \ddot{u}_2\ddot{u}_4 \end{array} \qquad (804)$$

$$\begin{array}{ll} \ddot{v}_5^\star = \ddot{v}_5 + \ddot{v}_7\ , & \ddot{u}_5^\star = \ddot{u}_5 + \ddot{u}_7 \\ \ddot{v}_6^\star = \ddot{v}_6 + \ddot{v}_8 + \ddot{v}_5\ddot{v}_7\ , & \ddot{u}_6^\star = \ddot{u}_6 + \ddot{u}_8 + \ddot{u}_5\ddot{u}_7 \\ \ddot{v}_7^\star = \ddot{v}_5\ddot{v}_8 + \ddot{v}_6\ddot{v}_7\ , & \ddot{u}_7^\star = \ddot{u}_5\ddot{u}_8 + \ddot{u}_6\ddot{u}_7 \\ \ddot{v}_8^\star = \ddot{v}_6\ddot{v}_8\ , & \ddot{u}_8^\star = \ddot{u}_6\ddot{u}_8 \end{array} \qquad (805)$$

$$G(\ddot{v}) \triangleq \begin{bmatrix} 1 \\ \ddot{v} \end{bmatrix}\begin{bmatrix} 1 & \ddot{v}^{\mathrm{T}} \end{bmatrix} = \begin{bmatrix} 1 & \ddot{v}_1 & \ddot{v}_2 & \ddot{v}_3 & \ddot{v}_4 & \ddot{v}_5 & \ddot{v}_6 & \ddot{v}_7 & \ddot{v}_8 \\ \ddot{v}_1 & \ddot{v}_1^2 & \ddot{v}_1\ddot{v}_2 & \ddot{v}_1\ddot{v}_3 & \ddot{v}_1\ddot{v}_4 & \ddot{v}_1\ddot{v}_5 & \ddot{v}_1\ddot{v}_6 & \ddot{v}_1\ddot{v}_7 & \ddot{v}_1\ddot{v}_8 \\ \ddot{v}_2 & \ddot{v}_1\ddot{v}_2 & \ddot{v}_2^2 & \ddot{v}_2\ddot{v}_3 & \ddot{v}_2\ddot{v}_4 & \ddot{v}_2\ddot{v}_5 & \ddot{v}_2\ddot{v}_6 & \ddot{v}_2\ddot{v}_7 & \ddot{v}_2\ddot{v}_8 \\ \ddot{v}_3 & \ddot{v}_1\ddot{v}_3 & \ddot{v}_2\ddot{v}_3 & \ddot{v}_3^2 & \ddot{v}_3\ddot{v}_4 & \ddot{v}_3\ddot{v}_5 & \ddot{v}_3\ddot{v}_6 & \ddot{v}_3\ddot{v}_7 & \ddot{v}_3\ddot{v}_8 \\ \ddot{v}_4 & \ddot{v}_1\ddot{v}_4 & \ddot{v}_2\ddot{v}_4 & \ddot{v}_3\ddot{v}_4 & \ddot{v}_4^2 & \ddot{v}_4\ddot{v}_5 & \ddot{v}_4\ddot{v}_6 & \ddot{v}_4\ddot{v}_7 & \ddot{v}_4\ddot{v}_8 \\ \ddot{v}_5 & \ddot{v}_1\ddot{v}_5 & \ddot{v}_2\ddot{v}_5 & \ddot{v}_3\ddot{v}_5 & \ddot{v}_4\ddot{v}_5 & \ddot{v}_5^2 & \ddot{v}_5\ddot{v}_6 & \ddot{v}_5\ddot{v}_7 & \ddot{v}_5\ddot{v}_8 \\ \ddot{v}_6 & \ddot{v}_1\ddot{v}_6 & \ddot{v}_2\ddot{v}_6 & \ddot{v}_3\ddot{v}_6 & \ddot{v}_4\ddot{v}_6 & \ddot{v}_5\ddot{v}_6 & \ddot{v}_6^2 & \ddot{v}_6\ddot{v}_7 & \ddot{v}_6\ddot{v}_8 \\ \ddot{v}_7 & \ddot{v}_1\ddot{v}_7 & \ddot{v}_2\ddot{v}_7 & \ddot{v}_3\ddot{v}_7 & \ddot{v}_4\ddot{v}_7 & \ddot{v}_5\ddot{v}_7 & \ddot{v}_6\ddot{v}_7 & \ddot{v}_7^2 & \ddot{v}_7\ddot{v}_8 \\ \ddot{v}_8 & \ddot{v}_1\ddot{v}_8 & \ddot{v}_2\ddot{v}_8 & \ddot{v}_3\ddot{v}_8 & \ddot{v}_4\ddot{v}_8 & \ddot{v}_5\ddot{v}_8 & \ddot{v}_6\ddot{v}_8 & \ddot{v}_7\ddot{v}_8 & \ddot{v}_8^2 \end{bmatrix} \in \mathbb{S}^{\mathbf{9}} \qquad (806)$$

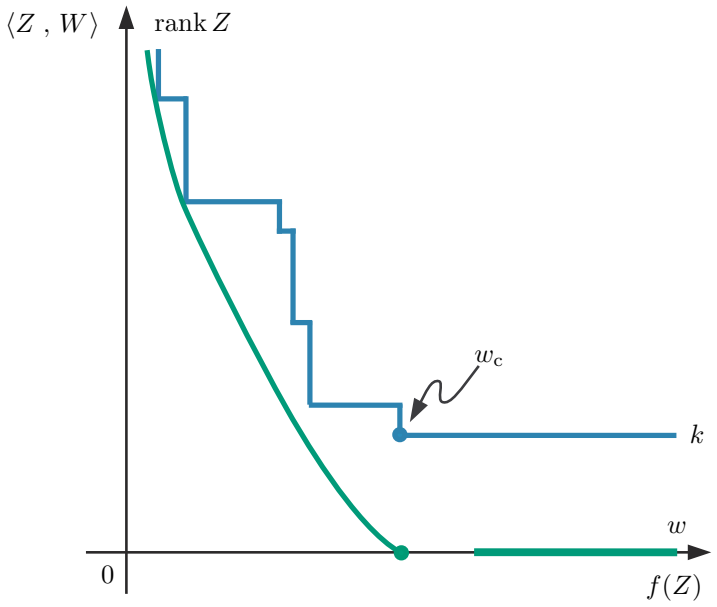


Figure 105: Regularization curve, parametrized by weight w for real convex objective f minimization (808) with rank constraint to k by convex iteration, illustrates discontinuity in f.

Setting

$$A = \begin{bmatrix} \operatorname{svec}(\ddot{A}_1)^{\mathrm{T}} \\ \vdots \\ \operatorname{svec}(\ddot{A}_8)^{\mathrm{T}} \end{bmatrix} \in \mathbb{R}^{\mathbf{8}\times\mathbf{9}(\mathbf{9}+\mathbf{1})/\mathbf{2}} \qquad (688)$$

then all `level` 3 (Figure **104**) nonnegative spectral factorization coefficients $\ddot{v}$ are found at once by solving

$$\begin{array}{rl} \underset{G\in\mathbb{S}^{\mathbf{9}}}{\text{find}} & \ddot{v} \in \mathbb{R}^{\mathbf{8}} \\ \text{subject to} & A \operatorname{svec} G = \ddot{v}^{\star} \\ & \begin{bmatrix} 1 \\ \ddot{v} \end{bmatrix} = G(:,1) \\ & \ddot{v} \succeq 0 \\ & (G \succeq 0) \\ & \operatorname{rank} G = 1 \end{array} \qquad (807)$$

The feasibility problem to find $\ddot{u}$ is similar. All second order Laplace transfer function coefficients can be found via (656). □

4.4.2 regularization

We test the convex iteration technique, for constraining rank, over a wide range of problems beyond localization of randomized positions (Figure **103**); *e.g*, *stress* (§7.2.2.7.1), *ball packing* (§5.4.2.2.6), and cardinality (§4.6). We have had some success introducing the direction matrix inner-product (792) as a *regularization* term[4.34]

$$\begin{array}{ll} \underset{Z\in\mathbb{S}^N}{\text{minimize}} & f(Z) + w\langle Z\,,\, W\rangle \\ \text{subject to} & Z \in \mathcal{C} \\ & Z \succeq 0 \end{array} \tag{808}$$

$$\begin{array}{ll} \underset{W\in\mathbb{S}^N}{\text{minimize}} & f(Z^\star) + w\langle Z^\star,\, W\rangle \\ \text{subject to} & 0 \preceq W \preceq I \\ & \operatorname{tr} W = N - n \end{array} \tag{809}$$

whose purpose is to constrain rank, affine dimension, or cardinality:

The abstraction, that is Figure **105**, is a synopsis; a broad generalization of accumulated empirical evidence: There exists a critical (smallest) weight $w_{\mathrm{c}}\bullet$ for which a minimal-rank constraint is just met. Graphical discontinuity can subsequently exist when there is a range of greater w providing required rank k but not necessarily increasing a minimization objective function f; *e.g*, §4.6.0.0.2. Positive scalar w is chosen via bisection so that $\langle Z\,,W\rangle$ just vanishes.

4.5 Constraining cardinality

The convex iteration technique for constraining rank can be applied to cardinality problems. There are parallels in its development analogous to how prototypical semidefinite program (687) resembles linear program (301) on page 240 [421]:

4.5.1 nonnegative variable

Our goal has been to reliably constrain rank in a semidefinite program. There is a direct analogy to linear programming that is simpler to present but, perhaps, more difficult to solve. In Optimization, that analogy is known as the *cardinality problem.*

Consider a feasibility problem $Ax=b$, but with an upper bound k on cardinality $\|x\|_0$ of a nonnegative solution x: for $A\in\mathbb{R}^{m\times n}$ and vector $b\in\mathcal{R}(A)$

$$\begin{array}{rl} \text{find} & x \in \mathbb{R}^n \\ \text{subject to} & Ax = b \\ & x \succeq 0 \\ & \|x\|_0 \leq k \end{array} \tag{529}$$

[4.34] called *multiobjective-* or *vector optimization.* Proof of convergence for this convex iteration is identical to that in §4.4.1.2.1 because f is a convex real function, hence bounded below, and $f(Z^\star)$ is constant in (809).

where $\|x\|_0 \leq k$ means[4.35] vector x has at most k nonzero entries; such a vector is presumed existent in the feasible set. Nonnegativity constraint $x \succeq 0$ is analogous to positive semidefiniteness; the notation means vector x belongs to the nonnegative orthant $\mathbb{R}^n_+$. Cardinality is quasiconcave on $\mathbb{R}^n_+$ just as rank is quasiconcave on $\mathbb{S}^n_+$. [63, §3.4.2]

4.5.1.1 direction vector

We propose that cardinality-constrained feasibility problem (529) can be equivalently expressed as iteration of a sequence of two convex problems: for $0 \leq k \leq n-1$

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{minimize}} & \langle x\,,\, y\rangle \\ \text{subject to} & Ax = b \\ & x \succeq 0 \end{array} \tag{156}$$

$$\sum_{i=k+1}^{n} \pi(x^\star)_i \;=\; \begin{array}{ll} \underset{y\in\mathbb{R}^n}{\text{minimize}} & \langle x^\star,\, y\rangle \\ \text{subject to} & 0 \preceq y \preceq \mathbf{1} \\ & y^{\mathrm{T}}\mathbf{1} = n-k \end{array} \tag{524}$$

where π is the (nonincreasing) presorting function. This sequence is iterated until ${x^\star}^{\mathrm{T}} y^\star$ vanishes; *id est*, until desired cardinality is achieved. But this *global convergence* cannot always be guaranteed.[4.36]

Problem (524) is analogous to the rank constraint problem; (p.266)

$$\sum_{i=k+1}^{N} \lambda(G^\star)_i \;=\; \begin{array}{ll} \underset{W\in\mathbb{S}^N}{\text{minimize}} & \langle G^\star,\, W\rangle \\ \text{subject to} & 0 \preceq W \preceq I \\ & \operatorname{tr} W = N-k \end{array} \tag{1800a}$$

Linear program (524) sums the $n-k$ smallest entries from vector x. In context of problem (529), we want $n-k$ entries of x to sum to zero; *id est*, we want a globally optimal objective ${x^\star}^{\mathrm{T}} y^\star$ to vanish: more generally, (*confer* (775))

$$\sum_{i=k+1}^{n} \pi(|x^\star|)_i \;=\; \langle\, |x^\star|\,,\, y^\star\rangle \;=\; |x^\star|^{\mathrm{T}} y^\star \;\triangleq\; 0 \tag{810}$$

defines *global convergence* for the iteration. Then $n-k$ entries of $x^\star$ are themselves zero whenever their absolute sum is, and cardinality of $x^\star \in \mathbb{R}^n$ is at most k. *Optimal direction vector* $y^\star$ is defined as any nonnegative vector for which

$$\text{(529)} \qquad \begin{array}{rl} \text{find} & x\in\mathbb{R}^n \\ \text{subject to} & Ax = b \\ & x \succeq 0 \\ & \|x\|_0 \leq k \end{array} \quad\equiv\quad \begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{minimize}} & \langle x\,,\, y^\star\rangle \\ \text{subject to} & Ax = b \\ & x \succeq 0 \end{array} \qquad \text{(156)}$$

Existence of such a $y^\star$, whose nonzero entries are complementary to those of $x^\star$, is obvious assuming existence of a cardinality-k solution $x^\star$.

[4.35] Although it is a metric (§5.2), cardinality $\|x\|_0$ cannot be a norm (§3.2) because it is not positively homogeneous.

[4.36] When it succeeds, a sequence may be regarded as a *homotopy* to minimal 0-norm.

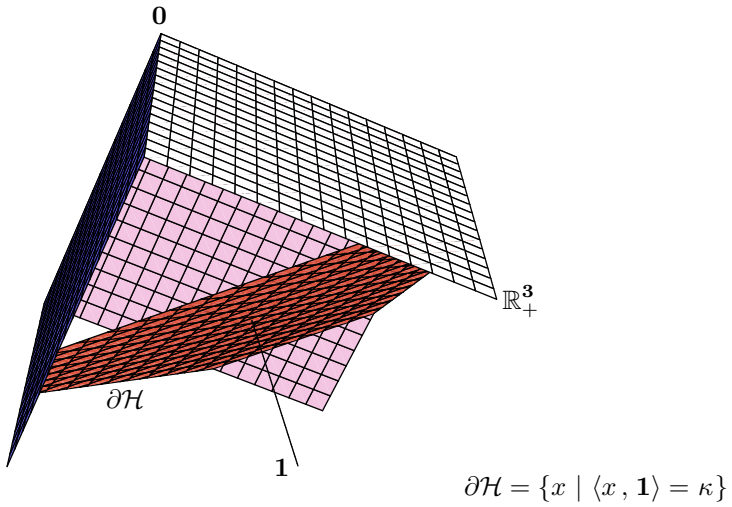


Figure 106: (*confer* Figure 90) 1-norm heuristic for cardinality minimization can be interpreted as minimization of a hyperplane, $\partial\mathcal{H}$ with normal $\mathbf{1}$, over nonnegative orthant drawn here in $\mathbb{R}^{\mathbf{3}}$. Polar of direction vector $y=\mathbf{1}$ points toward origin.

4.5.1.2 direction vector interpretation

(*confer* §4.4.1.1) Vector y may be interpreted as a negative search direction; it points opposite to direction of movement of hyperplane $\{x \mid \langle x\,,\,y\rangle = \tau\}$ in a minimization of real linear function $\langle x\,,\,y\rangle$ over the feasible set in linear program (156). (p.67) Direction vector y is not unique. The feasible set of direction vectors in (524) is the convex hull of all cardinality-$(n-k)$ one-vectors; *videlicet*,

$$\operatorname{conv}\{u\in\mathbb{R}^n \mid \operatorname{card} u = n-k\,,\;\; u_i\in\{0,1\}\} = \{a\in\mathbb{R}^n \mid \mathbf{1}\succeq a\succeq 0\,,\;\; \langle\mathbf{1}\,,a\rangle = n-k\} \tag{811}$$

This set, argument to $\operatorname{conv}\{\,\}$, comprises the extreme points of set (811) which is a *nonnegative hypercube slice.* An optimal solution y to (524), that is an extreme point of its feasible set, is known in closed form: it has 1 in each entry corresponding to the $n-k$ smallest entries of $x^\star$ and has 0 elsewhere. That particular polar direction $-y$ can be interpreted[4.37] by Proposition 7.1.3.0.3 as pointing toward the nonnegative orthant in the *Cartesian subspace,* whose basis is a subset of the Cartesian axes, containing all cardinality k (or less) vectors having the same ordering as $x^\star$. Consequently, for that closed-form solution, (*confer* (776))

$$\sum_{i=k+1}^{n} \pi(|x^\star|)_i \;=\; \langle |x^\star|\,,\,y\rangle \;=\; |x^\star|^{\mathrm{T}} y \;\geq\; 0 \tag{812}$$

[4.37] Convex iteration (156) (524) is not a projection method because there is no thresholding or discard of variable-vector x entries. An optimal direction vector y must always reside on the feasible set boundary in (524) page 292; *id est*, it is ill-advised to attempt simultaneous optimization of variables x and y.

When $y = \mathbf{1}$, as in 1-norm minimization for example, then polar direction $-y$ points directly at the origin (the cardinality-0 nonnegative vector) as in Figure **106**. We sometimes solve (524) instead of employing a known closed form because a direction vector is not unique. Setting direction vector y instead in accordance with an iterative inverse weighting scheme, called *reweighting* [176], was described for the 1-norm by Huo [223, §4.11.3] in 1999.

4.5.1.3 convergence can mean stalling

Convex iteration (156) (524) always converges to a *locally optimal solution,* a fixed point of possibly infeasible cardinality, by virtue of a monotonically nonincreasing real objective sequence. [274, §1.2] [43, §1.1] There can be no proof of global convergence, defined by (810). Constraining cardinality, solution to problem (529), can often be achieved but simple examples can be contrived that *stall* at a fixed point of infeasible cardinality; at a positive objective value $\langle x^\star,\, y\rangle = \tau > 0$. Direction vector y is then manipulated, as countermeasure, to steer out of local minima; *e.g*, complete randomization as in Example 4.5.1.5.1, or reinitialization to a random cardinality-$(n-k)$ vector in the same nonnegative orthant face demanded by the current iterate: y has nonnegative uniformly distributed random entries in $(0\,,1]$ corresponding to the $n-k$ smallest entries of $x^\star$ and has 0 elsewhere. Zero entries behave like memory or state while randomness greatly diminishes likelihood of a stall. When this particular heuristic is successful, cardinality and objective sequence $\langle x^\star,\, y\rangle$ *versus* iteration are characterized by noisy monotonicity.

4.5.1.4 algebraic derivation of direction vector for convex iteration

In §3.2.2.1.3, the compressed sensing problem was precisely represented as a nonconvex difference of convex functions bounded below by 0

$$\begin{array}{rl} \text{find} & x\in\mathbb{R}^n \\ \text{subject to} & Ax = b \\ & x \succeq 0 \\ & \|x\|_0 \leq k \end{array} \quad\equiv\quad \begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{minimize}} & \|x\|_1 - \|x\|_{\frac{n}{k}} \\ \text{subject to} & Ax = b \\ & x \succeq 0 \end{array} \tag{529}$$

where convex k-largest norm $\|x\|_{\frac{n}{k}}$ is monotonic on $\mathbb{R}^n_+$. There we showed how (529) is equivalently stated in terms of gradients

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{minimize}} & \left\langle x\,,\, \nabla\|x\|_1 - \nabla\|x\|_{\frac{n}{k}} \right\rangle \\ \text{subject to} & Ax = b \\ & x \succeq 0 \end{array} \tag{813}$$

because

$$\|x\|_1 = x^{\mathrm{T}}\nabla\|x\|_1\;, \qquad \|x\|_{\frac{n}{k}} = x^{\mathrm{T}}\nabla\|x\|_{\frac{n}{k}}\;, \qquad x \succeq 0 \tag{814}$$

The objective function from (813) is a directional derivative (at x in direction x, §D.1.6, *confer* §D.1.4.1.1) of the objective function from (529) while the direction vector of convex iteration

$$y = \nabla\|x\|_1 - \nabla\|x\|_{\frac{n}{k}} \tag{815}$$

is an objective gradient where $\nabla\|x\|_1 = \nabla\mathbf{1}^{\mathrm{T}}x = \mathbf{1}$ under nonnegativity and

$$\nabla\|x\|_k^n = \nabla z^{\mathrm{T}}x = \left.\begin{array}{rl} \underset{z\in\mathbb{R}^n}{\text{arg maximize}} & z^{\mathrm{T}}x \\ \text{subject to} & 0 \preceq z \preceq \mathbf{1} \\ & z^{\mathrm{T}}\mathbf{1} = k \end{array}\right\}, \qquad x \succeq 0 \tag{532}$$

is not unique. Substituting $\mathbf{1} - z \leftarrow z$ the direction vector becomes

$$y = \mathbf{1} - \begin{array}{rl} \underset{z\in\mathbb{R}^n}{\text{arg maximize}} & z^{\mathrm{T}}x \\ \text{subject to} & 0 \preceq z \preceq \mathbf{1} \\ & z^{\mathrm{T}}\mathbf{1} = k \end{array} \leftarrow \begin{array}{rl} \underset{z\in\mathbb{R}^n}{\text{arg minimize}} & z^{\mathrm{T}}x \\ \text{subject to} & 0 \preceq z \preceq \mathbf{1} \\ & z^{\mathrm{T}}\mathbf{1} = n - k \end{array} \tag{524}$$

4.5.1.5 optimality conditions for compressed sensing

Now we see how global optimality conditions can be stated without reference to a dual problem: From conditions (468) for optimality of (529), it is necessary [63, §5.5.3] that

$$\begin{aligned} x^\star &\succeq 0 && (1) \\ Ax^\star &= b && (2) \\ \nabla\|x^\star\|_1 - \nabla\|x^\star\|_k^n + A^{\mathrm{T}}\nu^\star &\succeq 0 && (3) \\ \langle \nabla\|x^\star\|_1 - \nabla\|x^\star\|_k^n + A^{\mathrm{T}}\nu^\star,\, x^\star\rangle &= 0 && (4\ell) \end{aligned} \tag{816}$$

These conditions must hold at any optimal solution (locally or globally). By (814), the fourth condition is identical to

$$\|x^\star\|_1 - \|x^\star\|_k^n + \nu^{\star\mathrm{T}}Ax^\star = 0 \qquad (4\ell) \tag{817}$$

Because a 1-norm

$$\|x\|_1 = \|x\|_k^n + \|\pi(|x|)_{k+1:n}\|_1 \tag{818}$$

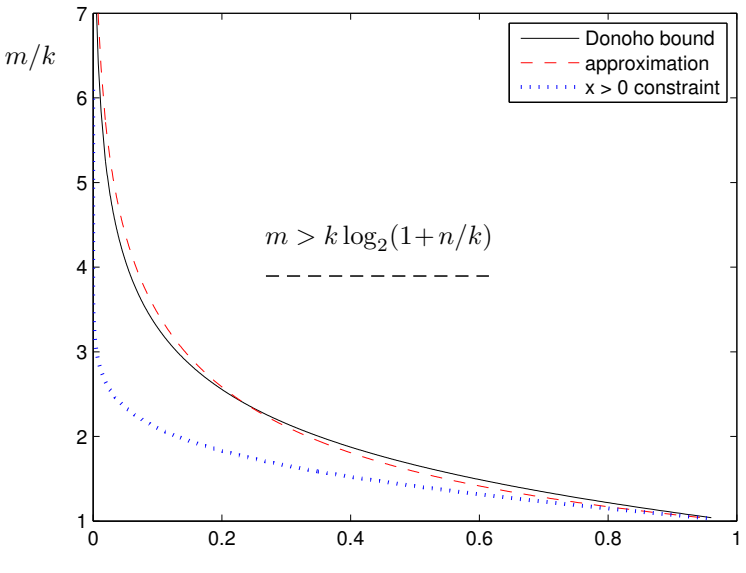
is separable into k largest and $n-k$ smallest absolute entries,

$$\|\pi(|x|)_{k+1:n}\|_1 = 0 \;\Leftrightarrow\; \|x\|_0 \le k \qquad (4g) \tag{819}$$

is a necessary condition for global optimality. By assumption, matrix A is fat and $b \neq \mathbf{0} \;\Rightarrow\; Ax^\star \neq \mathbf{0}$. This means $\nu^\star \in \mathcal{N}(A^{\mathrm{T}}) \subset \mathbb{R}^m$, and $\nu^\star = \mathbf{0}$ when A is full-rank. By definition, $\nabla\|x\|_1 \succeq \nabla\|x\|_k^n$ always holds. Assuming existence of a cardinality-k solution, then only three of the four conditions are necessary and sufficient for global optimality of (529):

$$\begin{aligned} x^\star &\succeq 0 && (1) \\ Ax^\star &= b && (2) \\ \|x^\star\|_1 - \|x^\star\|_k^n &= 0 && (4g) \end{aligned} \tag{820}$$

meaning, global optimality of a feasible solution to (529) is identified by a zero objective.



$$\begin{array}{ll} \underset{x}{\text{minimize}} & \|x\|_1 \\ \text{subject to} & Ax = b \end{array} \tag{517}$$

$$\begin{array}{ll} \underset{x}{\text{minimize}} & \|x\|_1 \\ \text{subject to} & Ax = b \\ & x \succeq 0 \end{array} \tag{522}$$

Figure 107: For Gaussian random matrix $A \in \mathbb{R}^{m\times n}$, graph illustrates Donoho/Tanner least lower bound on number of measurements m below which recovery of k-sparse n-length signal x by linear programming fails with overwhelming probability. Hard problems are below curve, but not the reverse; *id est*, failure above depends on proximity. Inequality demarcates *approximation* (dashed curve) empirically observed in [24]. Problems having nonnegativity constraint (dotted) are easier to solve. [133] [134]

4.5.1.5.1 Example. *Sparsest solution to* $Ax=b$. [73] [129]
(*confer* Example 4.5.2.0.4) Problem (720) has sparsest solution not easily recoverable by least 1-norm; *id est*, not by compressed sensing because of proximity to a theoretical lower bound on number of *measurements* m depicted in Figure **107**: for $A\in\mathbb{R}^{m\times n}$

- Given data from Example 4.2.3.1.1, for $m=3,\ n=6,\ k=1$

$$A=\begin{bmatrix} -1 & 1 & 8 & 1 & 1 & 0 \\ -3 & 2 & 8 & \frac{1}{2} & \frac{1}{3} & \frac{1}{2}-\frac{1}{3} \\ -9 & 4 & 8 & \frac{1}{4} & \frac{1}{9} & \frac{1}{4}-\frac{1}{9} \end{bmatrix},\qquad b=\begin{bmatrix} 1 \\ \frac{1}{2} \\ \frac{1}{4} \end{bmatrix} \tag{720}$$

  the sparsest solution to classical linear equation $Ax=b$ is $x=e_4\in\mathbb{R}^{\mathbf{6}}$ (*confer* (733)).

Although the sparsest solution is recoverable by inspection, we discern it instead by convex iteration; namely, by iterating problem sequence (156) (524) on page 292. From the numerical data given, cardinality $\|x\|_0=1$ is expected. Iteration continues until $x^{\mathrm{T}}y$ vanishes (to within some numerical precision); *id est*, until desired cardinality is achieved. But this comes not without a stall.

Stalling, whose occurrence is sensitive to initial conditions of convex iteration, is a consequence of finding a local minimum of a multimodal objective $\langle x\,,\,y\rangle$ when regarded as simultaneously variable in x and y. (§3.8.0.0.3) Stalls are simply detected as fixed points x of infeasible cardinality, sometimes remedied by reinitializing direction vector y to a random positive state.

Bolstered by success in breaking out of a stall, we then apply convex iteration to 22,000 randomized problems:

- Given random data for $m=3,\ n=6,\ k=1$, in MATLAB notation

$$\texttt{A}=\texttt{randn}(3\,,\,6),\quad \texttt{index}=\texttt{round}(5*\texttt{rand}(1))+1,\quad \texttt{b}=\texttt{rand}(1)*\texttt{A}(:\,,\,\texttt{index}) \tag{821}$$

  the sparsest solution $x\propto e_{\texttt{index}}$ is a scaled standard basis vector.

Without convex iteration or a nonnegativity constraint $x\succeq 0$, rate of failure for this minimal cardinality problem $Ax=b$ by 1-norm minimization of x is 22%. That failure rate drops to 6% with a nonnegativity constraint. If we then engage convex iteration, detect stalls, and randomly reinitialize the direction vector, failure rate drops to 0% but the amount of computation is approximately doubled. □

Stalling is not an inevitable behavior. For some problem types (beyond mere $Ax=b$), convex iteration succeeds nearly all the time. Here is a cardinality problem, with noise, whose statement is just a bit more intricate but easy to solve in a few convex iterations:

4.5.1.5.2 Example. *Signal dropout.* [132, §6.2]
Signal dropout is an old problem; well studied from both an industrial and academic perspective. Essentially *dropout* means momentary loss or gap in a signal, while passing through some channel, caused by some man-made or natural phenomenon. The signal lost is assumed completely destroyed somehow. What remains within the time-gap is system or idle channel noise. The signal could be voice over Internet protocol (VoIP), for

example, audio data from a *compact disc* (CD) or video data from a digital video disc (DVD), a television transmission over cable or the airwaves, or a typically ravaged cell phone communication, *etcetera.*

Here we consider signal dropout in a discrete-time signal corrupted by additive white noise assumed uncorrelated to the signal. The linear channel is assumed to introduce no filtering. We create a discretized windowed signal for this example by positively combining k randomly chosen vectors from a *discrete cosine transform* (DCT) basis denoted $\Psi \in \mathbb{R}^{n\times n}$. Frequency increases, in the Fourier sense, from DC toward Nyquist as column index of basis Ψ increases. Otherwise, details of the basis are unimportant except for its orthogonality $\Psi^{\mathrm{T}} = \Psi^{-1}$. Transmitted signal is denoted

$$s = \Psi z \in \mathbb{R}^n \tag{822}$$

whose upper bound on DCT basis coefficient cardinality $\operatorname{card} z \leq k$ is assumed known;[4.38] hence a critical assumption: transmitted signal s is sparsely supported ($k < n$) on the DCT basis. It is further assumed that nonzero signal coefficients in vector z place each chosen basis vector above the noise floor.

We also assume that the gap's beginning and ending in time are precisely localized to within a sample; *id est*, index ℓ locates the last sample prior to the gap's onset, while index $n-\ell+1$ locates the first sample subsequent to the gap: for rectangularly windowed received signal g possessing a time-gap loss and additive noise $\eta \in \mathbb{R}^n$

$$g = \begin{bmatrix} s_{1:\ell} & + & \eta_{1:\ell} \\ & & \eta_{\ell+1:n-\ell} \\ s_{n-\ell+1:n} & + & \eta_{n-\ell+1:n} \end{bmatrix} \in \mathbb{R}^n \tag{823}$$

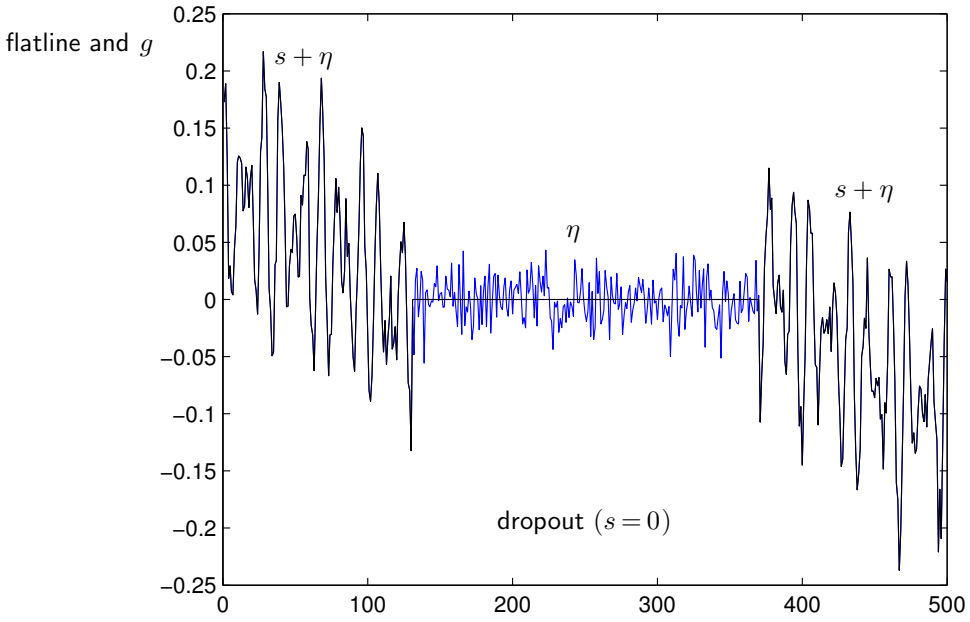
The window is thereby centered on the gap and short enough so that the DCT spectrum of signal s can be assumed static over the window's duration n. Signal to noise ratio within this window is defined

$$\mathrm{SNR} \triangleq 20 \log \frac{\left\| \begin{bmatrix} s_{1:\ell} \\ s_{n-\ell+1:n} \end{bmatrix} \right\|}{\|\eta\|} \tag{824}$$

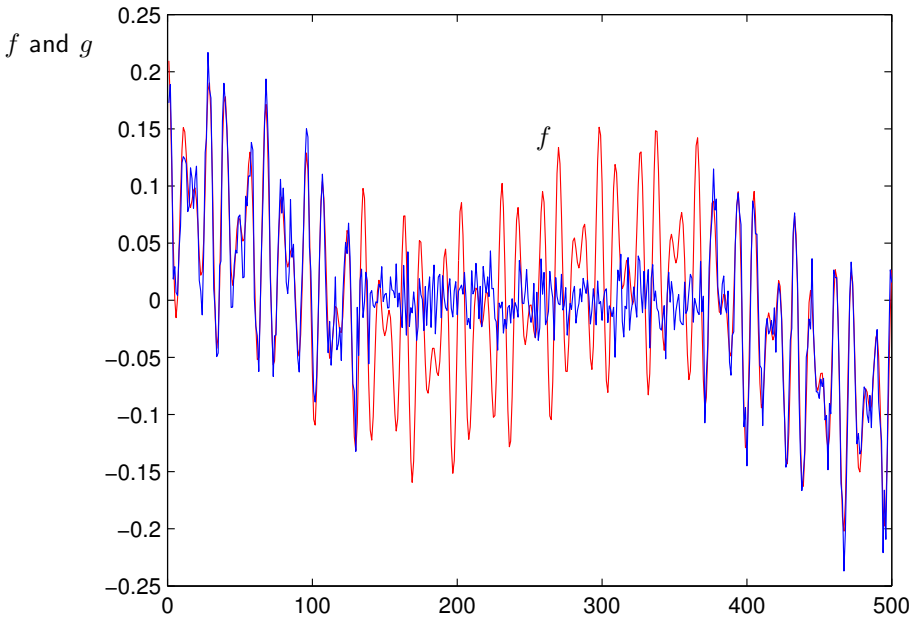
In absence of noise, knowing the signal DCT basis and having a good estimate of basis coefficient cardinality makes perfectly reconstructing gap-loss easy: it amounts to solving a linear system of equations and requires little or no optimization; with caveat, number of equations exceeds cardinality of signal representation (roughly $\ell \geq k$) with respect to DCT basis.

But addition of a significant amount of noise η increases level of difficulty dramatically; a 1-norm based method of reducing cardinality, for example, almost always returns DCT basis coefficients numbering in excess of minimal cardinality. We speculate that is because signal cardinality 2ℓ becomes the predominant cardinality. DCT basis coefficient cardinality is an explicit constraint to the optimization problem we shall pose: In presence of noise, constraints equating reconstructed signal f to received signal g are not possible.

[4.38] This simplifies exposition, although it may be an unrealistic assumption in many applications.

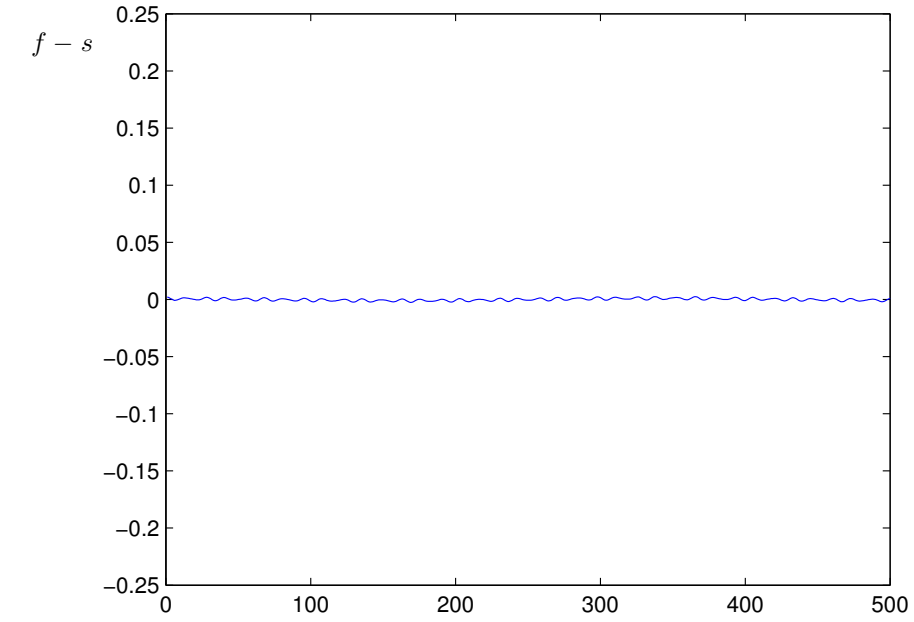


(a)

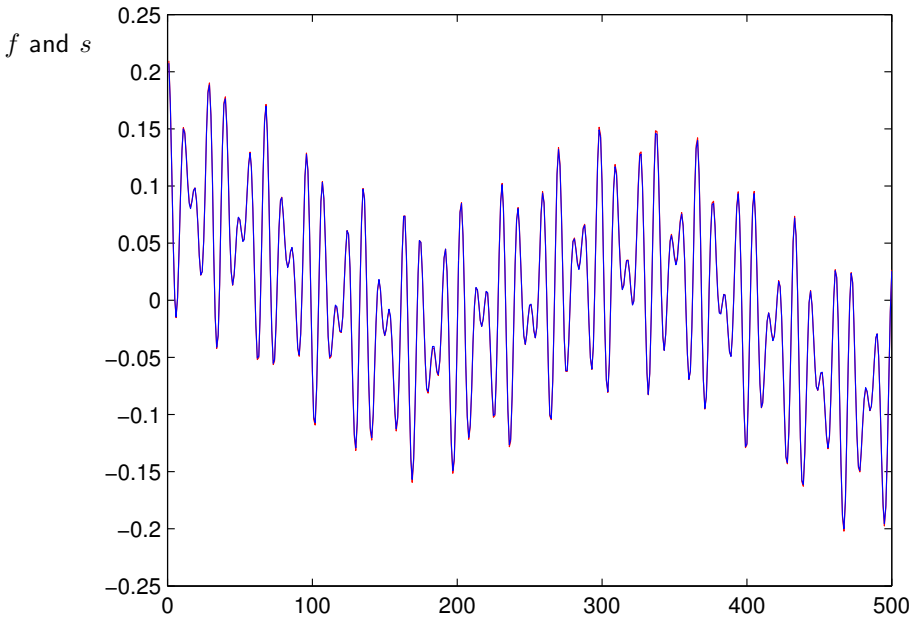


(b)

Figure 108: **(a)** Signal dropout in signal s corrupted by noise η (SNR = 10dB, $g = s + \eta$). Flatline indicates duration of signal dropout. **(b)** Reconstructed signal f (red) overlaid with corrupted signal g.



(a)



(b)

Figure 109: **(a)** Error signal power (reconstruction f less original noiseless signal s) is 36dB below s. **(b)** Original signal s overlaid with reconstruction f (red) from signal g having dropout plus noise.

We can instead formulate the dropout recovery problem as a best approximation:

$$\begin{array}{ll}\underset{x\in\mathbb{R}^n}{\text{minimize}} & \left\| \begin{bmatrix} f_{1:\ell} - g_{1:\ell} \\ f_{n-\ell+1:n} - g_{n-\ell+1:n} \end{bmatrix} \right\| \\ \text{subject to} & f = \Psi\, x \\ & x \succeq 0 \\ & \operatorname{card} x \leq k \end{array} \tag{825}$$

We propose solving this nonconvex problem (825) by moving the cardinality constraint to the objective as a regularization term as explained in §4.5; *id est*, by iteration of two convex problems until convergence:

$$\begin{array}{ll}\underset{x\in\mathbb{R}^n}{\text{minimize}} & \langle x\,,\, y\rangle + \left\| \begin{bmatrix} f_{1:\ell} - g_{1:\ell} \\ f_{n-\ell+1:n} - g_{n-\ell+1:n} \end{bmatrix} \right\| \\ \text{subject to} & f = \Psi\, x \\ & x \succeq 0 \end{array} \tag{826}$$

and

$$\begin{array}{ll}\underset{y\in\mathbb{R}^n}{\text{minimize}} & \langle x^\star,\, y\rangle \\ \text{subject to} & 0 \preceq y \preceq \mathbf{1} \\ & y^{\mathrm{T}}\mathbf{1} = n - k \end{array} \tag{524}$$

Signal cardinality 2ℓ is implicit to the problem statement. When number of samples in the dropout region exceeds half the window size, then that deficient cardinality of signal remaining becomes a source of degradation to reconstruction in presence of noise. Thus, by observation, we divine a reconstruction rule for this signal dropout problem to attain good noise suppression: ℓ must exceed a maximum of cardinality bounds; $2\ell \geq \max\{2k\,,\, n/2\}$.

Figure **108** and Figure **109** show one realization of this dropout problem. Original signal s is created by adding four ($k = 4$) randomly selected DCT basis vectors, from Ψ ($n = 500$ in this example), whose amplitudes are randomly selected from a uniform distribution above the noise floor; in the interval $[10^{-10/20},\, 1]$. Then a 240-sample dropout is realized ($\ell = 130$) and Gaussian noise η added to make corrupted signal g (from which a best approximation f will be made) having 10dB signal to noise ratio (824). The time gap contains much noise, as apparent from Figure **108**a. But in only a few iterations (826) (524), original signal s is recovered with relative error power 36dB down; illustrated in Figure **109**. Correct cardinality is also recovered ($\operatorname{card} x = \operatorname{card} z$) along with the basis vector indices used to make original signal s. Approximation error is due to DCT basis coefficient estimate error. When this experiment is repeated 1000 times on noisy signals averaging 10dB SNR, the correct cardinality and indices are recovered 99% of the time with average relative error power 30dB down. Without noise, we get perfect reconstruction in one iteration. [405, MATLAB code] □

4.5.1.6 Compressed sensing geometry with a nonnegative variable

It is well known that cardinality problem (529), on page 197, is easier to solve by linear programming when variable x is nonnegatively constrained than when not; *e.g*, Figure **75**, Figure **107**. We postulate a simple geometrical explanation:

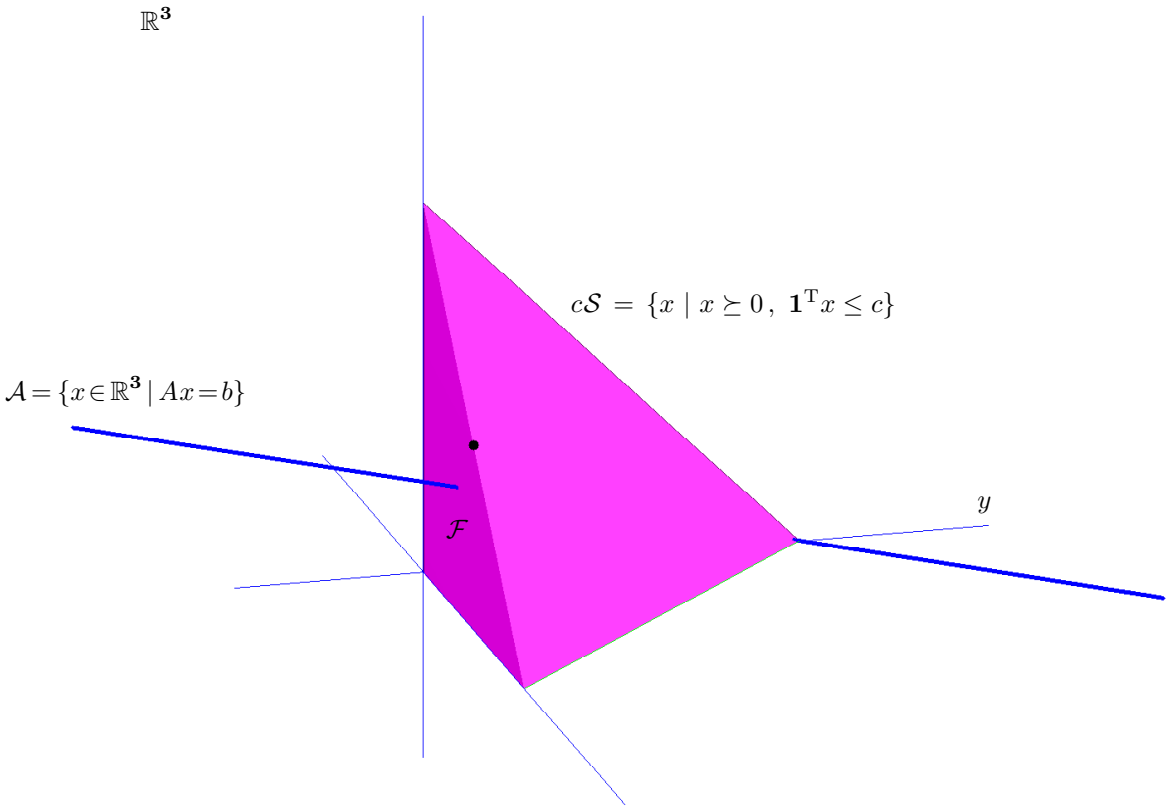


Figure 110: Simplex $\mathcal{S}$ is convex hull of origin and all cardinality-1 nonnegative vectors of unit norm (its vertices). Line $\mathcal{A}$, intersecting two-dimensional (cardinality-2) face $\mathcal{F}$ of nonnegative simplex $c\mathcal{S}$, emerges from $c\mathcal{S}$ at a cardinality-1 vertex. $\mathcal{S}$ equals nonnegative orthant $\mathbb{R}^3_+ \cap$ 1-norm ball $\mathcal{B}_1$ (Figure 74). Kissing point achieved when $\bullet$ (on edge) meets $\mathcal{A}$ as simplex contracts (as scalar c diminishes) under optimization (522).

Figure 74 illustrates 1-norm ball $\mathcal{B}_1$ in $\mathbb{R}^3$ and affine subset $\mathcal{A}$ defined $\{x\in\mathbb{R}^3 \mid Ax=b\}$. Prototypical compressed sensing problem, for $A\in\mathbb{R}^{m\times n}$

$$\begin{array}{ll}\underset{x}{\text{minimize}} & \|x\|_1 \\ \text{subject to} & Ax=b\end{array} \tag{517}$$

is solved when the 1-norm ball $\mathcal{B}_1$ kisses the affine subset.

If variable x is constrained to the nonnegative orthant

$$\begin{array}{ll}\underset{x\in\mathbb{R}^n}{\text{minimize}} & \|x\|_1 \\ \text{subject to} & Ax=b \\ & x\succeq 0\end{array} \equiv \begin{array}{ll}\underset{x\in\mathbb{R}^n}{\text{minimize}} & \mathbf{1}^{\mathrm{T}}x \\ \text{subject to} & Ax=b \\ & x\succeq 0\end{array} \equiv \begin{array}{ll}\underset{c\in\mathbb{R}\,,\ x\in\mathbb{R}^n}{\text{minimize}} & c \\ \text{subject to} & Ax=b \\ & x\in c\mathcal{S}\end{array} \tag{522}$$

then 1-norm ball $\mathcal{B}_1$ becomes nonnegative simplex $\mathcal{S}$ in Figure 110 where

$$c\mathcal{S} \;=\; \{[\,I\in\mathbb{R}^{n\times n}\;\; \mathbf{0}\in\mathbb{R}^n\,]a \mid a^{\mathrm{T}}\mathbf{1}=c\,,\; a\succeq 0\} \;=\; \{x \mid x\succeq 0\,,\; \mathbf{1}^{\mathrm{T}}x\le c\} \tag{827}$$

Nonnegative simplex $\mathcal{S}$ is the convex hull of its vertices. All $n+1$ vertices of $\mathcal{S}$ are constituted by standard basis vectors and the origin. In other words, all its nonzero extreme points are cardinality-1.

Affine subset $\mathcal{A}$ kisses nonnegative simplex $c^\star\mathcal{S}$ at optimality of (522). A kissing point is achieved at $x^\star$ for optimal $c^\star$ as $\mathcal{B}_1$ or $\mathcal{S}$ contracts. Whereas 1-norm ball $\mathcal{B}_1$ has only six vertices in $\mathbb{R}^3$ corresponding to cardinality-1 solutions, simplex $\mathcal{S}$ has three edges (along the Cartesian axes) containing an infinity of cardinality-1 solutions. And whereas $\mathcal{B}_1$ has twelve edges containing cardinality-2 solutions, $\mathcal{S}$ has three (out of total four) facets constituting cardinality-2 solutions. In other words, likelihood of a low-cardinality solution is higher by kissing nonnegative simplex $\mathcal{S}$ (522) than by kissing 1-norm ball $\mathcal{B}_1$ (517) because facial dimension (corresponding to given cardinality) is higher in $\mathcal{S}$.

Empirically, this observation also holds in other Euclidean dimensions (Figure 75, Figure 107).

4.5.1.7 cardinality-1 compressed sensing problem always solvable

In the special case of cardinality-1 feasible solution to nonnegative compressed sensing problem (522), there is a geometrical interpretation that leads to an algorithm.

Figure 110 illustrates a cardinality-1 feasible solution to problem (522) in $\mathbb{R}^3$; a vertex solution. But *first-octant* $\mathcal{S}$ of 1-norm ball $\mathcal{B}_1$ does not kiss line $\mathcal{A}$; which would be an optimality condition. How can we perform optimization and make $\mathcal{A}$ intersect $\mathcal{S}$ at a vertex? Assuming that nonnegative cardinality-1 solutions exist in the feasible set, it so happens:

4.5.1.7.1 Algorithm. *Deprecation.*

Columns of *measurement matrix* A, corresponding to high cardinality solution of (522)[4.39] found by Simplex method [98], may be *deprecated* and the problem solved

[4.39] Because signed compressed sensing problem (517) can be equivalently expressed in a nonnegative variable, as we learned in Example 3.2.0.0.1 (p.194), and because a cardinality-1 constraint in (517) transforms to a cardinality-1 constraint in its nonnegative equivalent (521), then this cardinality-1 recursive reconstruction algorithm continues to hold for a signed variable as in (517).

again with those columns missing. Such columns are recursively removed from A until a cardinality-1 solution is found. ¶

This algorithm intimates that either a solution to problem (522) is cardinality-1 or column indices of A, corresponding to a higher cardinality solution, do not intersect that index corresponding to a cardinality-1 feasible solution.

When problem (522) is first solved, in the example of Figure **110**, solution is cardinality-2 at a kissing point on that edge of simplex $c\mathcal{S}$ indicated by $\bullet$. Imagining that the corresponding cardinality-2 face $\mathcal{F}$ has collapsed, as a result of zeroing those two extreme points whose convex hull constructs that same edge $\bullet$ of $\mathcal{F}$, then the simplex collapses to a line segment along the y axis. When that line segment kisses $\mathcal{A}$, then the cardinality-1 vertex solution illustrated has been found.[4.40]

4.5.1.7.2 Proof (pending). *Deprecation algorithm* 4.5.1.7.1.
We require proof that a cardinality-1 feasible solution to (522) cannot exist within a higher cardinality optimal solution found by Simplex method; for only then can corresponding columns of A be eliminated without precluding cardinality-1 at optimality of the deprecated problem. Crucial is the Simplex method of solution because then an optimal solution is guaranteed to reside at a vertex of the feasible set. [98, p.158] [16, p.2] ■

Although it is more efficient (compared with our algorithm) to search over individual columns of matrix A for a cardinality-1 solution known *a priori* to exist, tables are turned when cardinality exceeds 1:

4.5.2 cardinality-k geometric presolver

This idea of deprecating columns has foundation in convex cone theory. (§2.13.4.3) Removing columns (and rows)[4.41] from $A\in\mathbb{R}^{m\times n}$ in a linear program like (522) (§3.2) is known in the industry as *presolving*;[4.42] the elimination of redundant constraints and identically zero variables prior to numerical solution. We offer a different and geometric presolver:

Two interpretations of the constraints from problem (522) are realized in Figure **111**. Assuming that a cardinality-k solution exists and matrix A describes a pointed polyhedral cone $\mathcal{K}=\{Ax \mid x\succeq 0\}$, as in Figure **111**b, columns are removed from A if they do not belong to the smallest face $\mathcal{F}$ of $\mathcal{K}$ containing vector b; those columns correspond to 0-entries in variable vector x (and *vice versa*). Generators of that smallest face always hold a minimal cardinality solution, in other words, because a generator outside the smallest face (having positive coefficient) would violate the assumption that b belongs to that face.

[4.40] A similar argument holds for any orientation of line $\mathcal{A}$ and cardinality-1 point of emergence from simplex $c\mathcal{S}$. This cardinality-1 reconstruction algorithm also holds more generally when affine subset $\mathcal{A}$ has any higher dimension $n-m$.
[4.41] Rows of matrix A are removed based upon linear dependence. Assuming $b\in\mathcal{R}(A)$, corresponding entries of vector b may also be removed without loss of generality.
[4.42] The conic technique proposed here can exceed performance of the best industrial presolvers in terms of number of columns removed, but not in execution time. This geometric presolver becomes attractive when a linear or *integer program* is not solvable by other means; perhaps because of sheer dimension.

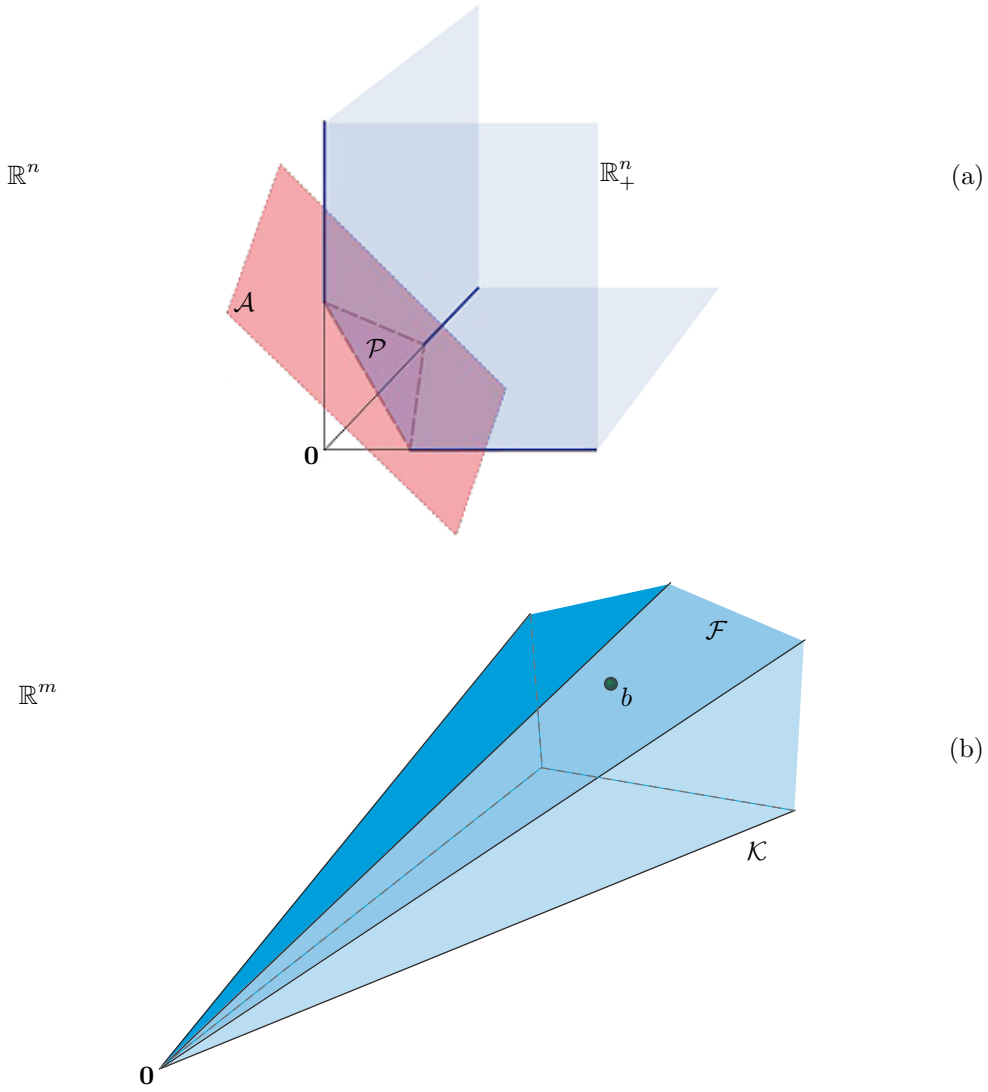


Figure 111: Constraint interpretations: **(a)** Halfspace-description of feasible set in problem (522) is a polyhedron $\mathcal{P}$ formed by intersection of nonnegative orthant $\mathbb{R}^n_+$ with hyperplanes $\mathcal{A}$ prescribed by equality constraint. (Drawing by Pedro Sánchez.) **(b)** Vertex-description of constraints in problem (522): point b belongs to polyhedral cone $\mathcal{K} = \{Ax \mid x \succeq 0\}$. Number of extreme directions in $\mathcal{K}$ may exceed dimensionality of ambient space.

Benefit accrues when vector b does not belong to relative interior of $\mathcal{K}$; there would be no columns to remove were $b \in \operatorname{rel}\operatorname{int}\mathcal{K}$ since the smallest face becomes cone $\mathcal{K}$ itself (Example 4.5.2.0.4). Were b an extreme direction, at the other end of the spectrum, then the smallest face is an edge that is a ray containing b; this geometrically describes a cardinality-1 case where all columns, save one, would be removed from A.

When vector b resides in a face $\mathcal{F}$ of $\mathcal{K}$ that is not cone $\mathcal{K}$ itself, benefit is realized as a reduction in computational intensity because the consequent equivalent problem has smaller dimension. Number of columns removed depends completely on geometry of a given problem; particularly, location of b within $\mathcal{K}$. In the example of Figure **111**b, interpreted literally in $\mathbb{R}^{\mathbf{3}}$, all but two columns of A are discarded by our presolver when b belongs to facet $\mathcal{F}$.

There are always exactly n linear feasibility problems to solve in order to discern generators of the smallest face of $\mathcal{K}$ containing b; the topic of §2.13.4.3.[4.43]

4.5.2.0.3 Exercise. *Minimal cardinality generators.*
Prove that generators of the smallest face $\mathcal{F}$ of $\mathcal{K} = \{Ax \mid x \succeq 0\}$ containing vector b always hold a minimal cardinality solution to $Ax = b$. ▼

4.5.2.0.4 Example. *Presolving for cardinality-2 solution to* $Ax = b$.
(*confer* Example 4.5.1.5.1) Again taking data from Example 4.2.3.1.1 ($A \in \mathbb{R}^{m\times n}$, desired cardinality of x is k), for $m=3$, $n=6$, $k=2$

$$A = \begin{bmatrix} -1 & 1 & 8 & 1 & 1 & 0 \\ -3 & 2 & 8 & \frac{1}{2} & \frac{1}{3} & \frac{1}{2}-\frac{1}{3} \\ -9 & 4 & 8 & \frac{1}{4} & \frac{1}{9} & \frac{1}{4}-\frac{1}{9} \end{bmatrix}, \qquad b = \begin{bmatrix} 1 \\ \frac{1}{2} \\ \frac{1}{4} \end{bmatrix} \tag{720}$$

proper cone $\mathcal{K} = \{Ax \mid x \succeq 0\}$ is pointed as proven by method of §2.12.2.2. A cardinality-2 solution is known to exist; sum of the last two columns of matrix A. Generators of the smallest face that contains vector b, found by the method in Example 2.13.4.3.1, comprise the entire A matrix because $b \in \operatorname{int}\mathcal{K}$ (§2.13.4.2.4). So geometry of this particular problem does not permit number of generators to be reduced below n by discerning the smallest face.[4.44] □

There is wondrous bonus to presolving when a constraint matrix is sparse. After columns are removed by theory of convex cones (finding the smallest face), some remaining rows may become $\mathbf{0}^{\mathrm{T}}$, identical to other rows, or nonnegative. When nonnegative rows appear in an equality constraint to $\mathbf{0}$, all nonnegative variables corresponding to nonnegative entries in those rows must vanish (§A.7.1); meaning, more columns may be removed. Once rows and columns have been removed from a constraint matrix, even more rows and columns may be removed by repeating the presolver procedure.

[4.43] Comparison of computational intensity for this conic presolver to a brute force search would pit combinatorial complexity, a binomial coefficient $\propto \binom{n}{k}$, against polynomial complexity; n linear feasibility problems plus numerical solution of the presolved problem.

[4.44] But a canonical set of conically independent generators of $\mathcal{K}$ comprise only the first two and last two columns of A.

4.5.3 constraining cardinality of signed variable

Now consider a feasibility problem equivalent to the classical problem from linear algebra $Ax=b$, but with an upper bound k on cardinality $\|x\|_0$: for vector $b\in\mathcal{R}(A)$

$$\begin{array}{rl} \text{find} & x\in\mathbb{R}^n \\ \text{subject to} & Ax=b \\ & \|x\|_0\leq k \end{array} \tag{828}$$

where $\|x\|_0\leq k$ means vector x has at most k nonzero entries; such a vector is presumed existent in the feasible set. Convex iteration (§4.5.1) utilizes a nonnegative variable; so absolute value $|x|$ is needed here. We propose that nonconvex problem (828) can be equivalently written as a sequence of convex problems that move the cardinality constraint to the objective:

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{minimize}} & \langle\, |x|\,,\, y\,\rangle \\ \text{subject to} & Ax=b \end{array} \equiv \begin{array}{ll} \underset{x\in\mathbb{R}^n,\ t\in\mathbb{R}^n}{\text{minimize}} & \langle\, t\,,\, y+\varepsilon\mathbf{1}\,\rangle \\ \text{subject to} & Ax=b \\ & -t\preceq x\preceq t \end{array} \tag{829}$$

$$\begin{array}{ll} \underset{y\in\mathbb{R}^n}{\text{minimize}} & \langle\, t^\star,\, y+\varepsilon\mathbf{1}\,\rangle \\ \text{subject to} & 0\preceq y\preceq\mathbf{1} \\ & y^{\mathrm{T}}\mathbf{1}=n-k \end{array} \tag{524}$$

where ε is a relatively small positive constant. This sequence is iterated until a direction vector y is found that makes $|x^\star|^{\mathrm{T}}y^\star$ vanish. The term $\langle\, t\,,\,\varepsilon\mathbf{1}\,\rangle$ in (829) is necessary to determine absolute value $|x^\star|=t^\star$ (§3.2) because vector y can have zero-valued entries. By initializing y to $(1-\varepsilon)\mathbf{1}$, the first iteration of problem (829) is a 1-norm problem (513); *id est*,

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n,\ t\in\mathbb{R}^n}{\text{minimize}} & \langle\, t\,,\,\mathbf{1}\,\rangle \\ \text{subject to} & Ax=b \\ & -t\preceq x\preceq t \end{array} \equiv \begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{minimize}} & \|x\|_1 \\ \text{subject to} & Ax=b \end{array} \tag{517}$$

Subsequent iterations of problem (829) engaging cardinality term $\langle\, t\,,\, y\,\rangle$ can be interpreted as corrections to this 1-norm problem leading to a 0-norm solution; vector y can be interpreted as a direction of search.

4.5.3.1 local convergence

As before (§4.5.1.3), convex iteration (829) (524) always converges to a locally optimal solution; a fixed point of possibly infeasible cardinality.

4.5.3.2 simple variations on a signed variable

Several useful equivalents to linear programs (829) (524) are easily devised, but their geometrical interpretation is not as apparent: *e.g*, equivalent in the limit $\varepsilon\to 0^+$

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n,\ t\in\mathbb{R}^n}{\text{minimize}} & \langle\, t\,,\, y\,\rangle \\ \text{subject to} & Ax=b \\ & -t\preceq x\preceq t \end{array} \tag{830}$$

$$\begin{array}{ll} \underset{y\in\mathbb{R}^n}{\text{minimize}} & \langle\, |x^\star|\,,\, y\rangle \\ \text{subject to} & 0 \preceq y \preceq \mathbf{1} \\ & y^{\mathrm{T}}\mathbf{1} = n-k \end{array} \tag{524}$$

We get another equivalent to linear programs (829) (524), in the limit, by interpreting problem (517) as infimum to a vertex-description of the 1-norm ball (Figure 74, Example 3.2.0.0.1, *confer* (516)):

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{minimize}} & \|x\|_1 \\ \text{subject to} & Ax = b \end{array} \equiv \begin{array}{ll} \underset{a\in\mathbb{R}^{2n}}{\text{minimize}} & \langle a\,,\, y\rangle \\ \text{subject to} & [\,A \;\; -A\,]a = b \\ & a \succeq 0 \end{array} \tag{831}$$

$$\begin{array}{ll} \underset{y\in\mathbb{R}^{2n}}{\text{minimize}} & \langle a^\star,\, y\rangle \\ \text{subject to} & 0 \preceq y \preceq \mathbf{1} \\ & y^{\mathrm{T}}\mathbf{1} = 2n-k \end{array} \tag{524}$$

where $x^\star = [\,I \;\; -I\,]a^\star$; from which it may be construed that any vector 1-norm minimization problem has equivalent expression in a nonnegative variable.

4.6 Cardinality and rank constraint examples

4.6.0.0.1 Example. *Projection on ellipsoid boundary.* [53] [161, §5.1] [263, §2] Consider classical linear equation $Ax = b$ but with constraint on norm of solution x, given matrices C, fat A, and vector $b \in \mathcal{R}(A)$

$$\begin{array}{ll} \text{find} & x \in \mathbb{R}^N \\ \text{subject to} & Ax = b \\ & \|Cx\| = 1 \end{array} \tag{832}$$

The set $\{x \mid \|Cx\| = 1\}$ (2) describes an ellipsoid boundary (Figure 15). This is a nonconvex problem because solution is constrained to that boundary. Assign

$$G = \begin{bmatrix} Cx \\ 1 \end{bmatrix} \begin{bmatrix} x^{\mathrm{T}}C^{\mathrm{T}} & 1 \end{bmatrix} = \begin{bmatrix} X & Cx \\ x^{\mathrm{T}}C^{\mathrm{T}} & 1 \end{bmatrix} \triangleq \begin{bmatrix} Cxx^{\mathrm{T}}C^{\mathrm{T}} & Cx \\ x^{\mathrm{T}}C^{\mathrm{T}} & 1 \end{bmatrix} \in \mathbb{S}^{N+1} \tag{833}$$

Any rank-1 solution must have this form. (§B.1.0.2) Ellipsoidally constrained feasibility problem (832) is equivalent to:

$$\begin{array}{ll} \underset{X\in\mathbb{S}^N}{\text{find}} & x \in \mathbb{R}^N \\ \text{subject to} & Ax = b \\ & G = \begin{bmatrix} X & Cx \\ x^{\mathrm{T}}C^{\mathrm{T}} & 1 \end{bmatrix} (\succeq 0) \\ & \operatorname{rank} G = 1 \\ & \operatorname{tr} X = 1 \end{array} \tag{834}$$

This is transformed to an equivalent convex problem by moving the rank constraint to the objective: We iterate solution of

$$\begin{array}{ll}\underset{X\in\mathbb{S}^N,\ x\in\mathbb{R}^N}{\text{minimize}} & \langle G\,,\,Y\rangle \\ \text{subject to} & Ax=b \\ & G=\begin{bmatrix} X & Cx \\ x^{\mathrm{T}}C^{\mathrm{T}} & 1\end{bmatrix}\succeq 0 \\ & \operatorname{tr}X=1\end{array} \tag{835}$$

with

$$\begin{array}{ll}\underset{Y\in\mathbb{S}^{N+1}}{\text{minimize}} & \langle G^\star,\,Y\rangle \\ \text{subject to} & 0\preceq Y\preceq I \\ & \operatorname{tr}Y=N\end{array} \tag{836}$$

until convergence. Initially $\mathbf{0}$, direction matrix $Y\in\mathbb{S}^{N+1}$ regulates rank. (1800a) Singular value decomposition $G^\star=U\Sigma Q^{\mathrm{T}}\in\mathbb{S}^{N+1}_+$ (§A.6) provides a new direction matrix $Y=U(:,2\!:\!N+1)U(:,2\!:\!N+1)^{\mathrm{T}}$ that optimally solves (836) at each iteration. An optimal solution to (832) is thereby found in a few iterations, making convex problem (835) its equivalent.

It remains possible for the iteration to stall; were a rank-1 G matrix not found. In that case, the current search direction is momentarily reversed with an added randomized element:

$$\mathtt{Y}=-\mathtt{U}(:,2\!:\!\mathtt{N}+1)*\left(\mathtt{U}(:,2\!:\!\mathtt{N}+1)'+\mathtt{randn}(\mathtt{N},1)*\mathtt{U}(:,1)'\right) \tag{837}$$

in MATLAB notation. This heuristic is quite effective for problem (832) which is exceptionally easy to solve by convex iteration.

When $b\notin\mathcal{R}(A)$ then problem (832) must be restated as a projection:

$$\begin{array}{ll}\underset{x\in\mathbb{R}^N}{\text{minimize}} & \|Ax-b\| \\ \text{subject to} & \|Cx\|=1\end{array} \tag{838}$$

This is a projection of point b on an ellipsoid boundary because any affine transformation of an ellipsoid remains an ellipsoid. Problem (835) in turn becomes

$$\begin{array}{ll}\underset{X\in\mathbb{S}^N,\ x\in\mathbb{R}^N}{\text{minimize}} & \langle G\,,\,Y\rangle+\|Ax-b\| \\ \text{subject to} & G=\begin{bmatrix} X & Cx \\ x^{\mathrm{T}}C^{\mathrm{T}} & 1\end{bmatrix}\succeq 0 \\ & \operatorname{tr}X=1\end{array} \tag{839}$$

We iterate this with calculation (836) of direction matrix Y as before until a rank-1 G matrix is found. □

4.6.0.0.2 Example. *Orthonormal Procrustes.* [53]
Example 4.6.0.0.1 is extensible. An orthonormal matrix $Q\in\mathbb{R}^{n\times p}$ is characterized $Q^{\mathrm{T}}Q=I$. Consider the particular case $Q=[\,x\ \ y\,]\in\mathbb{R}^{n\times\mathbf{2}}$ as variable to a Procrustes problem (§C.3): given $A\in\mathbb{R}^{m\times n}$ and $B\in\mathbb{R}^{m\times\mathbf{2}}$

$$\begin{array}{ll}\underset{Q\in\mathbb{R}^{n\times\mathbf{2}}}{\text{minimize}} & \|AQ-B\|_{\mathrm{F}} \\ \text{subject to} & Q^{\mathrm{T}}Q=I\end{array} \tag{840}$$

which is nonconvex. By vectorizing matrix Q we can make the assignment:

$$G = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \begin{bmatrix} x^{\mathrm{T}} & y^{\mathrm{T}} & 1 \end{bmatrix} = \begin{bmatrix} X & Z & x \\ Z^{\mathrm{T}} & Y & y \\ x^{\mathrm{T}} & y^{\mathrm{T}} & 1 \end{bmatrix} \triangleq \begin{bmatrix} xx^{\mathrm{T}} & xy^{\mathrm{T}} & x \\ yx^{\mathrm{T}} & yy^{\mathrm{T}} & y \\ x^{\mathrm{T}} & y^{\mathrm{T}} & 1 \end{bmatrix} \in \mathbb{S}^{2n+1} \tag{841}$$

Now orthonormal Procrustes problem (840) can be equivalently restated:

$$\begin{array}{ll} \underset{X\,,\,Y\in\mathbb{S}\,,\,Z\,,\,x\,,\,y}{\text{minimize}} & \|A[\,x\;\;y\,]-B\|_{\mathrm{F}} \\ \text{subject to} & G = \begin{bmatrix} X & Z & x \\ Z^{\mathrm{T}} & Y & y \\ x^{\mathrm{T}} & y^{\mathrm{T}} & 1 \end{bmatrix} (\succeq 0) \\ & \operatorname{rank} G = 1 \\ & \operatorname{tr} X = 1 \\ & \operatorname{tr} Y = 1 \\ & \operatorname{tr} Z = 0 \end{array} \tag{842}$$

To solve this, we form the convex problem sequence:

$$\begin{array}{ll} \underset{X\,,\,Y\,,\,Z\,,\,x\,,\,y}{\text{minimize}} & \|A[\,x\;\;y\,]-B\|_{\mathrm{F}} \,+\, \langle G\,,\,W\rangle \\ \text{subject to} & G = \begin{bmatrix} X & Z & x \\ Z^{\mathrm{T}} & Y & y \\ x^{\mathrm{T}} & y^{\mathrm{T}} & 1 \end{bmatrix} \succeq 0 \\ & \operatorname{tr} X = 1 \\ & \operatorname{tr} Y = 1 \\ & \operatorname{tr} Z = 0 \end{array} \tag{843}$$

and

$$\begin{array}{ll} \underset{W\in\mathbb{S}^{2n+1}}{\text{minimize}} & \langle G^{\star},\, W\rangle \\ \text{subject to} & 0 \preceq W \preceq I \\ & \operatorname{tr} W = 2n \end{array} \tag{844}$$

which has an optimal solution W that is known in closed form (p.567). These two problems are iterated until convergence and a rank-1 G matrix is found. A good initial value for direction matrix W is $\mathbf{0}$. Optimal $Q^{\star}$ equals $[\,x^{\star}\;\;y^{\star}\,]$.

Numerically, this Procrustes problem is easy to solve; a solution seems always to be found in one or few iterations. This problem formulation is extensible, of course, to orthogonal (square) matrices Q. □

4.6.0.0.3 Example. *Combinatorial Procrustes problem.*
In case $A\,,B\in\mathbb{R}^n$, when vector $A=\Xi B$ is known to be a permutation of vector B, solution to orthogonal Procrustes problem

$$\begin{array}{ll} \underset{X\in\mathbb{R}^{n\times n}}{\text{minimize}} & \|A-XB\|_{\mathrm{F}} \\ \text{subject to} & X^{\mathrm{T}} = X^{-1} \end{array} \tag{1812}$$

is not necessarily a permutation matrix Ξ even though an optimal objective value of 0 is found by the known analytical solution (§C.3). The simplest method of solution finds permutation matrix $X^\star = \Xi$ simply by sorting vector B with respect to A.

Instead of sorting, we design two different convex problems each of whose optimal solution is a permutation matrix: one design is based on rank constraint, the other on cardinality. Because permutation matrices are sparse by definition, we depart from a traditional Procrustes problem by instead demanding a vector 1-norm which is known to produce solutions more sparse than Frobenius' norm.

There are two principal facts exploited by the first convex iteration design (§4.4.1) we propose. Permutation matrices Ξ constitute:

1) the set of all nonnegative orthogonal matrices,

2) all points extreme to the polyhedron (100) of doubly stochastic matrices.

That means:

1) norm of each row and column is 1 ,[4.45]

$$\|\Xi(:,i)\| = 1\,, \quad \|\Xi(i,:)\| = 1\,, \qquad i = 1\ldots n \tag{845}$$

2) sum of each nonnegative row and column is 1, (§2.3.2.0.4)

$$\Xi^{\mathrm{T}}\mathbf{1} = \mathbf{1}\,, \quad \Xi\mathbf{1} = \mathbf{1}\,, \quad \Xi \succeq \mathbf{0} \tag{846}$$

solution via rank constraint

The idea is to individually constrain each column of variable matrix X to have unity norm. Matrix X must also belong to that polyhedron, (100) in the nonnegative orthant, implied by constraints (846); so each row-sum and column-sum of X must also be unity. It is this combination of nonnegativity, sum, and sum square constraints that extracts the permutation matrices: (Figure **112**) given nonzero vectors $A\,, B$

$$\begin{array}{ll} \underset{X\in\mathbb{R}^{n\times n},\, G_i\in\mathbb{S}^{n+1}}{\text{minimize}} & \|A - XB\|_1 \;+\; w\sum_{i=1}^{n}\langle G_i\,,\, W_i\rangle \\ \text{subject to} & \left.\begin{array}{l} G_i = \left[\begin{array}{cc} G_i(1\!:\!n\,,1\!:\!n) & X(:,i) \\ X(:,i)^{\mathrm{T}} & 1 \end{array}\right] \succeq 0 \\ \operatorname{tr} G_i = 2 \end{array}\right\}\,, \quad i = 1\ldots n \\ & X^{\mathrm{T}}\mathbf{1} = \mathbf{1} \\ & X\mathbf{1} = \mathbf{1} \\ & X \geq \mathbf{0} \end{array} \tag{847}$$

[4.45] This fact would be superfluous were the objective of minimization linear, because the permutation matrices reside at the extreme points of a polyhedron (100) implied by (846). But as posed, only either rows or columns need be constrained to unit norm because matrix orthogonality implies transpose orthogonality. (§B.5.2) Absence of vanishing inner product constraints that help define orthogonality, like $\operatorname{tr} Z = 0$ from Example 4.6.0.0.2, is a consequence of nonnegativity; *id est*, the only orthogonal matrices having exclusively nonnegative entries are permutations of the Identity.

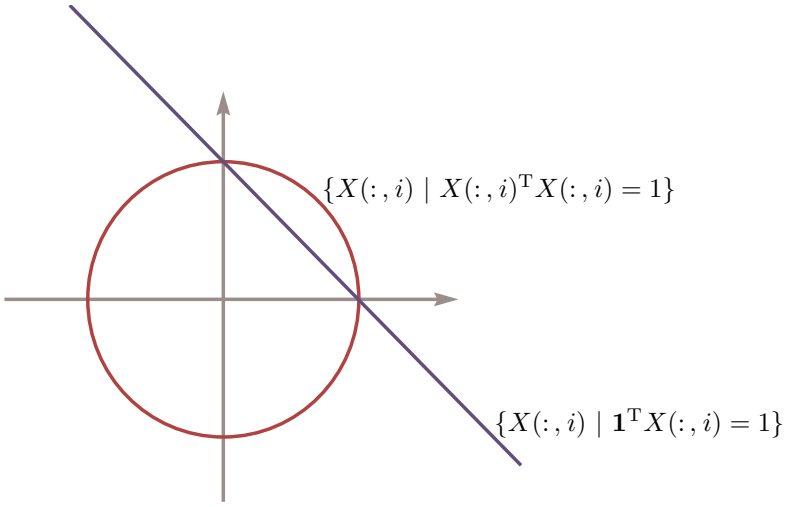


Figure 112: Permutation matrix i^{th} column-sum and column-norm constraint, abstract in two dimensions, when rank-1 constraint is satisfied. Optimal solutions reside at intersection of hyperplane with unit circle.

where $w \approx 10$ positively weights the rank regularization term. Optimal solutions $G_i^\star$ are key to finding direction matrices W_i for the next iteration of semidefinite programs (847) (848):

$$\left.\begin{array}{ll} \underset{W_i \in \mathbb{S}^{n+1}}{\text{minimize}} & \langle G_i^\star \,,\, W_i \rangle \\ \text{subject to} & 0 \preceq W_i \preceq I \\ & \operatorname{tr} W_i = n \end{array}\right\}, \qquad i = 1 \ldots n \tag{848}$$

Direction matrices thus found lead toward rank-1 matrices $G_i^\star$ on subsequent iterations. Constraint on trace of $G_i^\star$ normalizes the i^{th} column of $X^\star$ to unity because (*confer* p.377)

$$G_i^\star = \begin{bmatrix} X^\star(:,i) \\ 1 \end{bmatrix} \begin{bmatrix} X^\star(:,i)^{\mathrm{T}} & 1 \end{bmatrix} \tag{849}$$

at convergence. Binary-valued $X^\star$ column entries result from the further sum constraint $X\mathbf{1} = \mathbf{1}$. Columnar orthogonality is a consequence of the further transpose-sum constraint $X^{\mathrm{T}}\mathbf{1} = \mathbf{1}$ in conjunction with nonnegativity constraint $X \geq \mathbf{0}$; but we leave proof of orthogonality an exercise. The optimal objective value is 0 for both semidefinite programs when vectors A and B are related by permutation. In any case, optimal solution $X^\star$ becomes a permutation matrix Ξ.

Because there are n direction matrices W_i to find, it can be advantageous to invoke a known closed-form solution for each from page 567. What makes this combinatorial problem more tractable are relatively small semidefinite constraints in (847). (*confer* (843)) When a permutation A of vector B exists, number of iterations can be as small as 1. But this combinatorial Procrustes problem can be made even more challenging when vector A has repeated entries.

solution via cardinality constraint

Now the idea is to force solution at a vertex of permutation polyhedron (100) by finding a solution of desired sparsity. Because permutation matrix X is n-sparse by assumption, this combinatorial Procrustes problem may instead be formulated as a compressed sensing problem with convex iteration on cardinality of vectorized X (§4.5.1): given nonzero vectors $A\,, B$

$$\begin{array}{ll}\underset{X\in\mathbb{R}^{n\times n}}{\text{minimize}} & \|A-XB\|_1 \,+\, w\langle X\,,\,Y\rangle \\ \text{subject to} & X^{\mathrm{T}}\mathbf{1}=\mathbf{1} \\ & X\mathbf{1}=\mathbf{1} \\ & X\geq\mathbf{0}\end{array} \tag{850}$$

where direction vector Y is an optimal solution to

$$\begin{array}{ll}\underset{Y\in\mathbb{R}^{n\times n}}{\text{minimize}} & \langle X^\star,\,Y\rangle \\ \text{subject to} & \mathbf{0}\leq Y\leq\mathbf{1} \\ & \mathbf{1}^{\mathrm{T}}Y\mathbf{1}=n^2-n\end{array} \tag{524}$$

each a linear program. In this circumstance, use of closed-form solution for direction vector Y is discouraged. When vector A is a permutation of $B\,,$ both linear programs have objectives that converge to 0. When vectors A and B are permutations and no entries of A are repeated, optimal solution $X^\star$ can be found as soon as the first iteration.

In any case, $X^\star=\Xi$ is a permutation matrix. □

4.6.0.0.4 Exercise. *Combinatorial Procrustes constraints.*

Assume that the objective of semidefinite program (847) is 0 at optimality. Prove that the constraints in program (847) are necessary and sufficient to produce a permutation matrix as optimal solution. Alternatively and equivalently, prove those constraints necessary and sufficient to optimally produce a nonnegative orthogonal matrix. ▼

4.6.0.0.5 Example. *Tractable polynomial constraint.*

The set of all coefficients for which a multivariate polynomial were convex is generally difficult to determine. But the ability to handle rank constraints makes any nonconvex polynomial constraint transformable to a convex constraint. *All optimization problems having polynomial objective and polynomial constraints can be reformulated as a semidefinite program with a rank-1 constraint.* [302] Suppose we require

$$3+2x-xy\leq 0 \tag{851}$$

Identify

$$G=\begin{bmatrix}x\\ y\\ 1\end{bmatrix}\begin{bmatrix}x & y & 1\end{bmatrix}=\begin{bmatrix}x^2 & xy & x\\ xy & y^2 & y\\ x & y & 1\end{bmatrix}\in\mathbb{S}^{\mathbf{3}} \tag{852}$$

Then nonconvex polynomial constraint (851) is equivalent to constraint set

$$\begin{array}{l}\operatorname{tr}(GA)\leq 0\\ G_{33}=1\\ (G\succeq 0)\\ \operatorname{rank}G=1\end{array} \tag{853}$$

with direct correspondence to sense of trace inequality where G is assumed symmetric (§B.1.0.2) and

$$A = \begin{bmatrix} 0 & -\frac{1}{2} & 1 \\ -\frac{1}{2} & 0 & 0 \\ 1 & 0 & 3 \end{bmatrix} \in \mathbb{S}^{\mathbf{3}} \tag{854}$$

Then the method of convex iteration from §4.4.1 is applied to implement the rank constraint. □

4.6.0.0.6 Exercise. *Binary Pythagorean theorem.*
The technique in Example 4.6.0.0.5 is extensible to any quadratic constraint; *e.g*, $x^{\mathrm{T}}A\,x + 2b^{\mathrm{T}}x + c \le 0$, $x^{\mathrm{T}}A\,x + 2b^{\mathrm{T}}x + c \ge 0$, and $x^{\mathrm{T}}A\,x + 2b^{\mathrm{T}}x + c = 0$. Write a rank-constrained semidefinite program to solve (Figure **112**)

$$\begin{cases} x + y = 1 \\ x^2 + y^2 = 1 \end{cases} \tag{855}$$

whose feasible set is not connected. Implement this system in `cvx` [183] by convex iteration. ▼

4.6.0.0.7 Example. *High order polynomials.*
Consider nonconvex problem from Canadian Mathematical Olympiad 1999:

$$\begin{array}{ll} \underset{x\,,\,y\,,\,z\in\mathbb{R}}{\text{find}} & x\,,\,y\,,\,z \\ \text{subject to} & x^2y + y^2z + z^2x = \frac{2^2}{3^3} \\ & x + y + z = 1 \\ & x\,,\,y\,,\,z \ge 0 \end{array} \tag{856}$$

We wish to solve for, what is known to be, a tight upper bound $\frac{2^2}{3^3}$ on the constrained polynomial $x^2y + y^2z + z^2x$ by transformation to a rank-constrained semidefinite program. First identify

$$G = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \begin{bmatrix} x & y & z & 1 \end{bmatrix} = \begin{bmatrix} x^2 & xy & zx & x \\ xy & y^2 & yz & y \\ zx & yz & z^2 & z \\ x & y & z & 1 \end{bmatrix} \in \mathbb{S}^{\mathbf{4}} \tag{857}$$

$$X = \begin{bmatrix} x^2 \\ y^2 \\ z^2 \\ x \\ y \\ z \\ 1 \end{bmatrix} \begin{bmatrix} x^2 & y^2 & z^2 & x & y & z & 1 \end{bmatrix} = \begin{bmatrix} x^4 & x^2y^2 & z^2x^2 & x^3 & x^2y & zx^2 & x^2 \\ x^2y^2 & y^4 & y^2z^2 & xy^2 & y^3 & y^2z & y^2 \\ z^2x^2 & y^2z^2 & z^4 & z^2x & yz^2 & z^3 & z^2 \\ x^3 & xy^2 & z^2x & x^2 & xy & zx & x \\ x^2y & y^3 & yz^2 & xy & y^2 & yz & y \\ zx^2 & y^2z & z^3 & zx & yz & z^2 & z \\ x^2 & y^2 & z^2 & x & y & z & 1 \end{bmatrix} \in \mathbb{S}^{\mathbf{7}} \tag{858}$$

then apply convex iteration (§4.4.1) to implement rank constraints:

$$\begin{array}{ll}\underset{A\,,\,C\in\mathbb{S}\,,\,b}{\text{find}} & b \\ \text{subject to} & \operatorname{tr}(XE)=\frac{2^2}{3^3} \\ & G=\begin{bmatrix} A & b \\ b^{\mathrm{T}} & 1 \end{bmatrix}(\succeq 0) \\ & X=\begin{bmatrix} C & \begin{bmatrix}\delta(A)\\ b\end{bmatrix} \\ \begin{bmatrix}\delta(A)^{\mathrm{T}} & b^{\mathrm{T}}\end{bmatrix} & 1 \end{bmatrix}(\succeq 0) \\ & \mathbf{1}^{\mathrm{T}}b=1 \\ & b\succeq 0 \\ & \operatorname{rank}G=1 \\ & \operatorname{rank}X=1 \end{array} \tag{859}$$

where

$$E=\begin{bmatrix} 0&0&0&0&1&0&0\\ 0&0&0&0&0&1&0\\ 0&0&0&1&0&0&0\\ 0&0&1&0&0&0&0\\ 1&0&0&0&0&0&0\\ 0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0 \end{bmatrix}\frac{1}{2}\in\mathbb{S}^{\mathbf{7}} \tag{860}$$

[399, MATLAB code]. Positive semidefiniteness is optional only when rank-1 constraints are explicit by Theorem A.3.1.0.7. Optimal solution $(x\,,\,y\,,\,z)=(0\,,\frac{2}{3}\,,\frac{1}{3})$ to problem (856) is not unique. □

4.6.0.0.8 Exercise. *Motzkin polynomial.*
Prove $xy^2+x^2y-3xy+1$ to be nonnegative on the nonnegative orthant. ▼

4.6.0.0.9 Example. *Boolean vector satisfying $Ax\preceq b$.* (*confer* §4.2.3.1.1)
Now we consider solution to a discrete problem whose only known analytical method of solution is combinatorial in complexity: given $A\in\mathbb{R}^{M\times N}$ and $b\in\mathbb{R}^M$

$$\begin{array}{ll}\text{find} & x\in\mathbb{R}^N \\ \text{subject to} & Ax\preceq b \\ & \delta(xx^{\mathrm{T}})=\mathbf{1}\end{array} \tag{861}$$

This nonconvex problem demands a Boolean solution $[\,x_i=\pm 1,\ i=1\ldots N\,]$.

Assign a rank-1 matrix of variables; symmetric variable matrix X and solution vector x:

$$G=\begin{bmatrix} x\\ 1\end{bmatrix}\begin{bmatrix} x^{\mathrm{T}} & 1\end{bmatrix}=\begin{bmatrix} X & x\\ x^{\mathrm{T}} & 1\end{bmatrix}\triangleq\begin{bmatrix} xx^{\mathrm{T}} & x\\ x^{\mathrm{T}} & 1\end{bmatrix}\in\mathbb{S}^{N+1} \tag{862}$$

Then design an equivalent semidefinite feasibility problem to find a Boolean solution to

$Ax \preceq b$:

$$\begin{array}{ll} \underset{X\in\mathbb{S}^N}{\text{find}} & x \in \mathbb{R}^N \\ \text{subject to} & Ax \preceq b \\ & G = \left[\begin{array}{cc} X & x \\ x^{\mathrm{T}} & 1 \end{array}\right] (\succeq 0) \\ & \operatorname{rank} G = 1 \\ & \delta(X) = \mathbf{1} \end{array} \tag{863}$$

where $x_i^\star \in \{-1, 1\}$, $i=1 \dots N$. The two variables X and x are made dependent via their assignment to rank-1 matrix G. By (1711), an optimal rank-1 matrix $G^\star$ must take the form (862).

As before, we regularize the rank constraint by introducing a direction matrix Y into the objective:

$$\begin{array}{ll} \underset{X\in\mathbb{S}^N,\ x\in\mathbb{R}^N}{\text{minimize}} & \langle G\,,\, Y\rangle \\ \text{subject to} & Ax \preceq b \\ & G = \left[\begin{array}{cc} X & x \\ x^{\mathrm{T}} & 1 \end{array}\right] \succeq 0 \\ & \delta(X) = \mathbf{1} \end{array} \tag{864}$$

Solution of this semidefinite program is iterated with calculation of the direction matrix Y from semidefinite program (836). At convergence, in the sense (775), convex problem (864) becomes equivalent to nonconvex Boolean problem (861).

Direction matrix Y can be an orthogonal projector having closed-form expression, by (1800a), although convex iteration is not a projection method. (§4.4.1.1) Given randomized data A and b for a large problem, we find that stalling becomes likely (convergence of the iteration to a positive objective $\langle G^\star, Y\rangle$). To overcome this behavior, we introduce a heuristic into the implementation on *Wikimization* [389] that momentarily reverses direction of search (like (837)) upon stall detection. We find that rate of convergence can be sped significantly by detecting stalls early. □

4.6.0.0.10 Example. *Variable-vector normalization.*
Suppose, within some convex optimization problem, we want vector variables $x\,,\, y \in \mathbb{R}^N$ constrained by a nonconvex equality:

$$x\,\|y\| = y \tag{865}$$

id est, $\|x\| = 1$ and x points in the same direction as $y \neq \mathbf{0}$; *e.g*,

$$\begin{array}{ll} \underset{x\,,\,y}{\text{minimize}} & f(x\,,\,y) \\ \text{subject to} & (x\,,\,y) \in \mathcal{C} \\ & x\,\|y\| = y \end{array} \tag{866}$$

where f is some convex function and $\mathcal{C}$ is some convex set. We can realize the nonconvex equality by constraining rank and adding a regularization term to the objective. Make the

assignment:

$$G = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \begin{bmatrix} x^{\mathrm{T}} & y^{\mathrm{T}} & 1 \end{bmatrix} = \begin{bmatrix} X & Z & x \\ Z & Y & y \\ x^{\mathrm{T}} & y^{\mathrm{T}} & 1 \end{bmatrix} \triangleq \begin{bmatrix} xx^{\mathrm{T}} & xy^{\mathrm{T}} & x \\ yx^{\mathrm{T}} & yy^{\mathrm{T}} & y \\ x^{\mathrm{T}} & y^{\mathrm{T}} & 1 \end{bmatrix} \in \mathbb{S}^{2N+1} \tag{867}$$

where $X\,,\, Y \in \mathbb{S}^N$, also $Z \in \mathbb{S}^N$ [*sic*]. Any rank-1 solution must take the form of (867). (§B.1) The problem statement equivalent to (866) is then written

$$\begin{array}{ll} \underset{X\,,\, Y \in \mathbb{S}\,,\, Z\,,\, x\,,\, y}{\text{minimize}} & f(x\,,\, y) \,+\, \|X - Y\|_{\mathrm{F}} \\ \text{subject to} & (x\,,\, y) \in \mathcal{C} \\ & G = \begin{bmatrix} X & Z & x \\ Z & Y & y \\ x^{\mathrm{T}} & y^{\mathrm{T}} & 1 \end{bmatrix} (\succeq 0) \\ & \operatorname{rank} G = 1 \\ & \operatorname{tr}(X) = 1 \\ & \delta(Z) \succeq 0 \end{array} \tag{868}$$

The trace constraint on X normalizes vector x while the diagonal constraint on Z maintains sign between respective entries of x and y. Regularization term $\|X - Y\|_{\mathrm{F}}$ then makes x equal to y to within a real scalar; (§C.2.0.0.2) in this case, a positive scalar. To make this program solvable by convex iteration, as explained in Example 4.4.1.2.4 and other previous examples, we move the rank constraint to the objective

$$\begin{array}{ll} \underset{X\,,\, Y\,,\, Z\,,\, x\,,\, y}{\text{minimize}} & f(x\,,\, y) \,+\, \|X - Y\|_{\mathrm{F}} \,+\, \langle G\,,\, W \rangle \\ \text{subject to} & (x\,,\, y) \in \mathcal{C} \\ & G = \begin{bmatrix} X & Z & x \\ Z & Y & y \\ x^{\mathrm{T}} & y^{\mathrm{T}} & 1 \end{bmatrix} \succeq 0 \\ & \operatorname{tr}(X) = 1 \\ & \delta(Z) \succeq 0 \end{array} \tag{869}$$

by introducing a direction matrix W found from (1800a):

$$\begin{array}{ll} \underset{W \in \mathbb{S}^{2N+1}}{\text{minimize}} & \langle G^{\star},\, W \rangle \\ \text{subject to} & 0 \preceq W \preceq I \\ & \operatorname{tr} W = 2N \end{array} \tag{870}$$

This semidefinite program has an optimal solution that is known in closed form. Iteration (869) (870) terminates when $\operatorname{rank} G = 1$ and linear regularization $\langle G\,,\, W \rangle$ vanishes to within some numerical tolerance in (869); typically, in two iterations. If function f competes too much with the regularization, positively weighting each regularization term will become required. At convergence, problem (869) becomes a convex equivalent to the original nonconvex problem (866). □

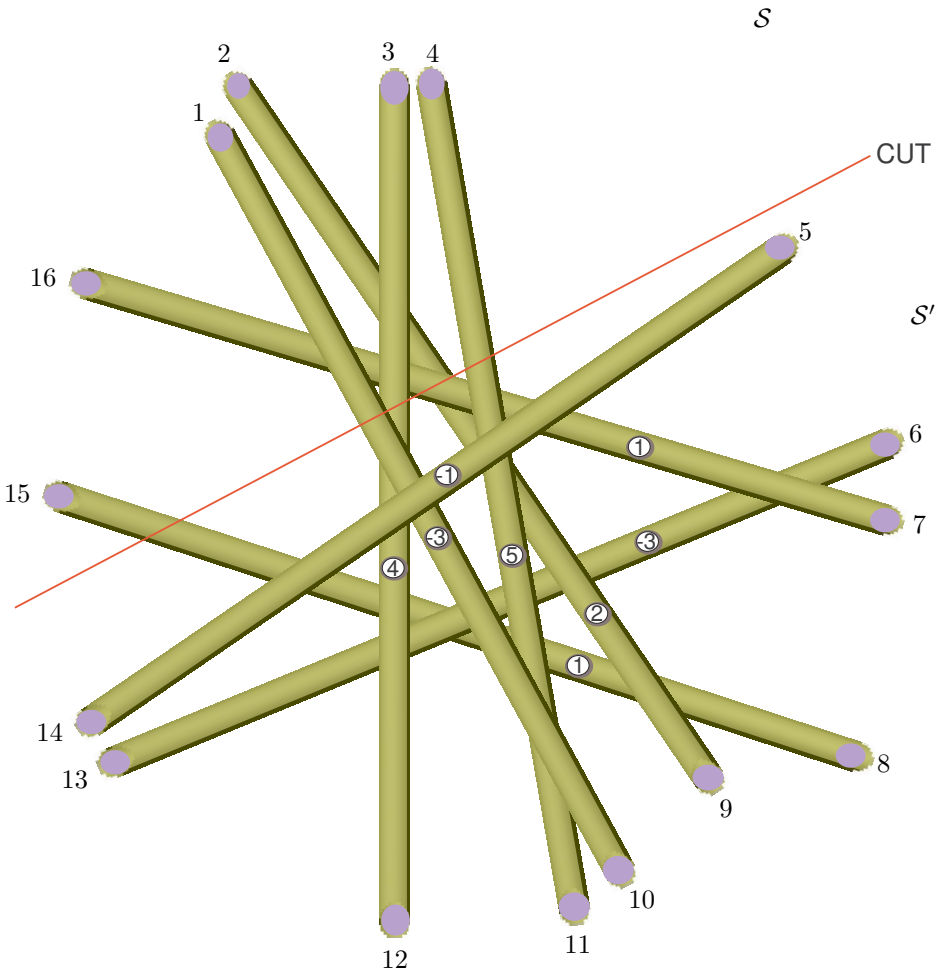


Figure 113: A CUT partitions nodes $\{i=1\ldots16\}$ of this graph into $\mathcal{S}$ and $\mathcal{S}'$. Linear arcs have circled weights. The problem is to find a cut maximizing total weight of all arcs linking partitions made by the cut.

4.6.0.0.11 Example. FAST MAX CUT. [120]

Let Γ be an n-node graph, and let the arcs $(i\,,j)$ of the graph be associated with ... weights a_{ij} . The problem is to find a cut of the largest possible weight, i.e., to partition the set of nodes into two parts $\mathcal{S}\,,\mathcal{S}'$ in such a way that the total weight of all arcs linking $\mathcal{S}$ and $\mathcal{S}'$ (i.e., with one incident node in $\mathcal{S}$ and the other one in $\mathcal{S}'$ [Figure **113**]*) is as large as possible.* −[35, §4.3.3]

Literature on the MAX CUT problem is vast because this problem has elegant primal and dual formulation, its solution is very difficult, and there exist many commercial applications; *e.g*, semiconductor design [136], quantum computing [426].

Our purpose here is to demonstrate how iteration of two simple convex problems can quickly converge to an optimal solution of the MAX CUT problem with a 98% success rate, on average.[4.46] MAX CUT is stated:

$$\begin{array}{ll}\underset{x}{\text{maximize}} & \sum_{1\le i<j\le n} a_{ij}(1-x_i\,x_j)\frac{1}{2} \\ \text{subject to} & \delta(xx^{\mathrm{T}}) = \mathbf{1}\end{array} \tag{871}$$

where $[a_{ij}]$ are real arc weights, and binary vector $x=[x_i]\in\mathbb{R}^n$ corresponds to the n nodes; specifically,

$$\begin{array}{lcl}\text{node } i\in\mathcal{S} & \Leftrightarrow & x_i = \;\; 1 \\ \text{node } i\in\mathcal{S}' & \Leftrightarrow & x_i = -1\end{array} \tag{872}$$

If nodes i and j have the same binary value x_i and x_j, then they belong to the same partition and contribute nothing to the cut. Arc $(i\,,j)$ traverses the cut, otherwise, adding its weight a_{ij} to the cut.

MAX CUT statement (871) is the same as, for $A=[a_{ij}]\in\mathbb{S}^n$

$$\begin{array}{ll}\underset{x}{\text{maximize}} & \frac{1}{4}\langle \mathbf{1}\mathbf{1}^{\mathrm{T}} - xx^{\mathrm{T}},\, A\rangle \\ \text{subject to} & \delta(xx^{\mathrm{T}}) = \mathbf{1}\end{array} \tag{873}$$

Because of Boolean assumption $\delta(xx^{\mathrm{T}})=\mathbf{1}$

$$\langle \mathbf{1}\mathbf{1}^{\mathrm{T}} - xx^{\mathrm{T}},\, A\rangle \;=\; \langle xx^{\mathrm{T}},\, \delta(A\mathbf{1}) - A\rangle \tag{874}$$

so problem (873) is the same as

$$\begin{array}{ll}\underset{x}{\text{maximize}} & \frac{1}{4}\langle xx^{\mathrm{T}},\, \delta(A\mathbf{1}) - A\rangle \\ \text{subject to} & \delta(xx^{\mathrm{T}}) = \mathbf{1}\end{array} \tag{875}$$

This MAX CUT problem is combinatorial (nonconvex).

Because an estimate of upper bound to MAX CUT is needed to ascertain convergence when vector x has large dimension, we digress to derive the dual problem: Directly from (875), its Lagrangian is [63, §5.1.5] (1507)

$$\begin{aligned}\mathfrak{L}(x\,,\nu) \;&=\; \tfrac{1}{4}\langle xx^{\mathrm{T}},\, \delta(A\mathbf{1}) - A\rangle \;+\; \langle \nu\,,\, \delta(xx^{\mathrm{T}}) - \mathbf{1}\rangle \\ &=\; \tfrac{1}{4}\langle xx^{\mathrm{T}},\, \delta(A\mathbf{1}) - A\rangle \;+\; \langle \delta(\nu)\,,\, xx^{\mathrm{T}}\rangle \;-\; \langle \nu\,,\, \mathbf{1}\rangle \\ &=\; \tfrac{1}{4}\langle xx^{\mathrm{T}},\, \delta(A\mathbf{1}+4\nu) - A\rangle \;-\; \langle \nu\,,\, \mathbf{1}\rangle\end{aligned} \tag{876}$$

[4.46] We term our solution to MAX CUT *fast* because we sacrifice a little accuracy to achieve speed; *id est*, only about two or three convex iterations, achieved by heavily weighting a rank regularization term.

where quadratic $x^{\mathrm{T}}(\delta(A\mathbf{1}+4\nu)-A)x$ has supremum 0 if $\delta(A\mathbf{1}+4\nu)-A$ is negative semidefinite, and has supremum ∞ otherwise. The finite supremum

$$g(\nu) \;=\; \sup_{x} \mathfrak{L}(x\,,\,\nu) \;=\; \begin{cases} -\langle \nu\,,\,\mathbf{1}\rangle\,, & A-\delta(A\mathbf{1}+4\nu)\succeq 0 \\ \infty & \text{otherwise} \end{cases} \tag{877}$$

is chosen to be the objective of minimization to dual (convex) problem

$$\begin{array}{ll} \underset{\nu}{\text{minimize}} & -\nu^{\mathrm{T}}\mathbf{1} \\ \text{subject to} & A-\delta(A\mathbf{1}+4\nu)\succeq 0 \end{array} \tag{878}$$

whose optimal value provides a least upper bound to MAX CUT, but is not tight ($\frac{1}{4}\langle xx^{\mathrm{T}},\,\delta(A\mathbf{1})-A\rangle < g(\nu)$, duality gap is nonzero). [171] In fact, we find that the bound's variance with problem instance is too large to be useful for this problem; thus ending our digression.[4.47]

To transform MAX CUT to its convex equivalent, first define

$$X = xx^{\mathrm{T}} \in \mathbb{S}^n \tag{883}$$

then MAX CUT (875) becomes

$$\begin{array}{ll} \underset{X\in\mathbb{S}^n}{\text{maximize}} & \frac{1}{4}\langle X\,,\,\delta(A\mathbf{1})-A\rangle \\ \text{subject to} & \delta(X)=\mathbf{1} \\ & (X\succeq 0) \\ & \operatorname{rank} X = 1 \end{array} \tag{879}$$

whose rank constraint can be regularized as in

$$\begin{array}{ll} \underset{X\in\mathbb{S}^n}{\text{maximize}} & \frac{1}{4}\langle X\,,\,\delta(A\mathbf{1})-A\rangle - w\langle X\,,\,W\rangle \\ \text{subject to} & \delta(X)=\mathbf{1} \\ & X\succeq 0 \end{array} \tag{880}$$

where $w\approx 1000$ is a nonnegative fixed weight, and W is a direction matrix determined from

$$\sum_{i=2}^{n}\lambda(X^\star)_i \;=\; \begin{array}{ll} \underset{W\in\mathbb{S}^n}{\text{minimize}} & \langle X^\star,\,W\rangle \\ \text{subject to} & 0\preceq W\preceq I \\ & \operatorname{tr} W = n-1 \end{array} \tag{1800a}$$

which has an optimal solution that is known in closed form. These two problems (880) and (1800a) are iterated until convergence as defined on page 266.

[4.47] Taking the dual of dual problem (878) would provide (879) but without the rank constraint. [164] Dual of a dual of even a convex primal problem is not necessarily the same primal problem; although, optimal solution of one can be obtained from the other.

Because convex problem statement (880) is so elegant, it is numerically solvable for large binary vectors within reasonable time.[4.48] To test our convex iterative method, we compare an optimal convex result to an actual solution of the MAX CUT problem found by performing a brute force combinatorial search of (875)[4.49] for a tight upper bound. Search-time limits binary vector lengths to 24 bits (about five days CPU time). 98% accuracy, actually obtained, is independent of binary vector length (12, 13, 20, 24) when averaged over more than 231 problem instances including planar, randomized, and toroidal graphs.[4.50] When failure occurred, large and small errors were manifest. That same 98% average accuracy is presumed maintained when binary vector length is further increased. A MATLAB program is provided on *Wikimization* [394]. □

4.6.0.0.12 Example. *Cardinality/rank problem.*
d'Aspremont, El Ghaoui, Jordan, & Lanckriet [99] propose approximating a positive semidefinite matrix $A \in \mathbb{S}^N_+$ by a rank-one matrix having constraint on cardinality c : for $0 < c < N$

$$\begin{array}{ll} \underset{z}{\text{minimize}} & \|A - zz^{\mathrm{T}}\|_{\mathrm{F}} \\ \text{subject to} & \operatorname{card} z \leq c \end{array} \tag{881}$$

which, they explain, is a hard problem equivalent to

$$\begin{array}{ll} \underset{x}{\text{maximize}} & x^{\mathrm{T}} A\, x \\ \text{subject to} & \|x\| = 1 \\ & \operatorname{card} x \leq c \end{array} \tag{882}$$

where $z \triangleq \sqrt{\lambda}\, x$ and where optimal solution $x^\star$ is a *principal eigenvector* (1793) (§A.5) of A and $\lambda = x^{\star\mathrm{T}} A\, x^\star$ is the *principal eigenvalue* [174, p.331] when c is true cardinality of that eigenvector. This is *principal component analysis* with a cardinality constraint which controls solution sparsity. Define the matrix variable

$$X \triangleq xx^{\mathrm{T}} \in \mathbb{S}^N \tag{883}$$

whose desired rank is 1, and whose desired diagonal cardinality

$$\operatorname{card} \delta(X) \equiv \operatorname{card} x \tag{884}$$

is equivalent to cardinality c of vector x. Then we can transform cardinality problem (882) to an equivalent problem in new variable X :[4.51]

[4.48] We solved for a length-250 binary vector in only a few minutes and convex iterations on a 2006 vintage laptop Core 2 CPU (Intel T7400@2.16GHz, 666MHz FSB).
[4.49] more computationally intensive than the proposed convex iteration by many orders of magnitude. Solving MAX CUT by searching over all binary vectors of length 100, for example, would occupy a contemporary supercomputer for a million years.
[4.50] Existence of a polynomial-time approximation to MAX CUT with accuracy provably better than 94.11% would refute NP-hardness; which Håstad believes to be highly unlikely. [197, thm.8.2] [198]
[4.51] A semidefiniteness constraint $X \succeq 0$ is not required, theoretically, because positive semidefiniteness of a rank-1 matrix is enforced by symmetry. (Theorem A.3.1.0.7)

$$\begin{array}{ll}\underset{X\in\mathbb{S}^N}{\text{maximize}} & \langle X\,,\,A\rangle \\ \text{subject to} & \langle X\,,\,I\rangle = 1 \\ & (X\succeq 0) \\ & \operatorname{rank} X = 1 \\ & \operatorname{card}\delta(X) \le c \end{array} \tag{885}$$

We transform problem (885) to an equivalent convex problem by introducing two direction matrices into regularization terms: W to achieve desired cardinality $\operatorname{card}\delta(X)$, and Y to find an approximating rank-one matrix X:

$$\begin{array}{ll}\underset{X\in\mathbb{S}^N}{\text{maximize}} & \langle X\,,\,A - w_1 Y\rangle - w_2\langle\delta(X)\,,\,\delta(W)\rangle \\ \text{subject to} & \langle X\,,\,I\rangle = 1 \\ & X\succeq 0 \end{array} \tag{886}$$

where w_1 and w_2 are positive scalars respectively weighting $\operatorname{tr}(XY)$ and $\delta(X)^{\mathrm{T}}\delta(W)$ just enough to insure that they vanish to within some numerical precision, where direction matrix Y is an optimal solution to semidefinite program

$$\begin{array}{ll}\underset{Y\in\mathbb{S}^N}{\text{minimize}} & \langle X^\star,\,Y\rangle \\ \text{subject to} & 0\preceq Y\preceq I \\ & \operatorname{tr} Y = N-1 \end{array} \tag{887}$$

and where diagonal direction matrix $W\in\mathbb{S}^N$ optimally solves linear program

$$\begin{array}{ll}\underset{W=\delta^2(W)}{\text{minimize}} & \langle\delta(X^\star)\,,\,\delta(W)\rangle \\ \text{subject to} & 0\preceq\delta(W)\preceq\mathbf{1} \\ & \operatorname{tr} W = N-c \end{array} \tag{888}$$

Both direction matrix programs are derived from (1800a) whose analytical solution is known but is not necessarily unique. We emphasize (*confer* p.266): because this iteration (886) (887) (888) (initial $Y, W = \mathbf{0}$) is not a projection method (§4.4.1.1), success relies on existence of matrices in the feasible set of (886) having desired rank and diagonal cardinality. In particular, the feasible set of convex problem (886) is a Fantope (91) whose extreme points constitute the set of all normalized rank-one matrices; among those are found rank-one matrices of any desired diagonal cardinality.

Convex problem (886) is neither a relaxation of cardinality problem (882); instead, problem (886) becomes a convex equivalent to (882) at global convergence of iteration (886) (887) (888). Because the feasible set of convex problem (886) contains all normalized (§B.1) symmetric rank-one matrices of every nonzero diagonal cardinality, a constraint too low or high in cardinality c will not prevent solution. An optimal rank-one solution $X^\star$, whose diagonal cardinality is equal to cardinality of a principal eigenvector of matrix A, will produce the least residual Frobenius norm (to within machine noise processes) in the original problem statement (881). □

`phantom(256)`

Figure 114: Shepp-Logan phantom from MATLAB *image processing toolbox.*

4.6.0.0.13 Example. *Compressive sampling of a phantom.*
In Summer 2004, Candès, Romberg, & Tao [73] and Donoho [129] released papers on perfect signal reconstruction from samples that stand in violation of Shannon's classical sampling theorem. These defiant signals are assumed sparse inherently or under some sparsifying affine transformation. Essentially, they proposed *sparse sampling theorems* asserting average sample rate independent of signal bandwidth and less than Shannon's rate.

MINIMUM SAMPLING RATE:

- OF Ω-BANDLIMITED SIGNAL: 2Ω ([301, §3.2] Shannon)
- OF k-SPARSE LENGTH-n SIGNAL: $k\log_2(1+n/k)$ (Figure **107** Candès/Donoho)

Certainly, much was already known about nonuniform or random sampling [37] [224] and about subsampling or *multirate systems* [95] [376]. Vetterli, Marziliano, & Blu [385] had congealed a theory of noiseless signal reconstruction, in May 2001, from samples that violate the Shannon rate. [404, *Sampling Sparsity*] They anticipated the sparsifying transform by recognizing: it is the *innovation* (onset) of functions constituting a (not necessarily bandlimited) signal that determines minimum sampling rate for perfect reconstruction. Average onset (sparsity), Vetterli *et alii* call, the *rate of innovation.* Vector inner-products that Candès/Donoho call *samples* or *measurements,* Vetterli calls *projections.* From those projections Vetterli demonstrates reconstruction (by digital signal processing and "root finding") of a Dirac comb, the very same prototypical signal from which Candès probabilistically derives minimum sampling rate [*Compressive Sampling and Frontiers in Signal Processing*, University of Minnesota, June 6, 2007]. Combining their terminology, we paraphrase a sparse sampling theorem:

- Minimum sampling rate, asserted by Candès/Donoho, $\propto$ Vetterli's rate of innovation (`a.k.a:` *information rate, degrees of freedom* [*ibidem* June 5, 2007]).

What distinguishes these researchers are their methods of reconstruction.

Properties of the 1-norm were also well understood by June 2004 finding applications in *deconvolution* of linear systems [87], constrained *linear regression* (*Lasso*) [366] [336], and *basis pursuit* [81] [229]. But never before had there been a formalized and rigorous sense that perfect reconstruction were possible by convex optimization of 1-norm when information lost in a subsampling process became nonrecoverable by classical methods. Donoho named this discovery *compressed sensing* to describe a nonadaptive perfect reconstruction method by means of linear programming. By the time Candès' and Donoho's landmark papers were finally published by IEEE in 2006, compressed sensing was old news that had spawned intense research which still persists; notably, from prominent members of the *wavelet* community.

Reconstruction of the *Shepp-Logan phantom* (Figure **114**), from a severely aliased image (Figure **116**) obtained by Magnetic Resonance Imaging (MRI), was the *impetus* driving Candès' quest for a sparse sampling theorem. He realized that line segments appearing in the aliased image were regions of high *total variation*. There is great motivation, in the medical community, to apply compressed sensing to MRI because it translates to reduced scan-time which brings great technological and physiological benefits. MRI is now about 35 years old, beginning in 1973 with Nobel laureate Paul Lauterbur from Stony Brook USA. There has been much progress in MRI and compressed sensing since 2004, but there have also been indications of 1-norm abandonment (indigenous to reconstruction by compressed sensing) in favor of criteria closer to 0-norm because of a correspondingly smaller number of measurements required to accurately reconstruct a sparse signal:[4.52]

5481 complex samples (22 radial lines, $\approx$256 complex samples per) were required in June 2004 to reconstruct a noiseless 256×256-pixel Shepp-Logan phantom by 1-norm minimization of an image-gradient integral estimate called *total variation*; *id est*, 8.4% subsampling of 65536 data. [73, §1.1] [72, §3.2] It was soon discovered that reconstruction of the Shepp-Logan phantom were possible with only 2521 complex samples (10 radial lines, Figure **115**); 3.8% subsampled data input to a (nonconvex) $\frac{1}{2}$-norm total-variation minimization. [79, §IIIA] The closer to 0-norm, the fewer the samples required for perfect reconstruction.

Passage of a few years witnessed an algorithmic speedup and dramatic reduction in minimum number of samples required for perfect reconstruction of the noiseless Shepp-Logan phantom. But minimization of total variation is ideally suited to recovery of any piecewise-constant image, like a phantom, because gradient of such images is highly sparse by design.

There is no inherent characteristic of real-life MRI images that would make reasonable an expectation of sparse gradient. Sparsification of a discrete image-gradient tends to preserve edges. Then minimization of total variation seeks an image having fewest edges. There is no deeper theoretical foundation than that. When applied to human brain scan or angiogram, with as much as 20% of 256×256 Fourier samples, we have observed[4.53] a 30dB

[4.52] Efficient techniques continually emerge urging 1-norm criteria abandonment; [84] [375] [374, §IID] *e.g*, five techniques for compressed sensing are compared in [38] demonstrating that 1-norm performance limits for cardinality minimization can be reliably exceeded.

[4.53] Experiments with real-life images were performed by Christine Law at Lucas Center for Imaging, Stanford University.

image/reconstruction-error ratio[4.54] barrier that seems impenetrable by the total-variation objective. Total-variation minimization has met with moderate success, in retrospect, only because some medical images are moderately piecewise-constant signals. One simply hopes a reconstruction, that is in some sense equal to a known subset of samples and whose gradient is most sparse, is that unique image we seek.[4.55]

The total-variation objective, operating on an image, is expressible as norm of a linear transformation (907). It is natural to ask whether there exist other sparsifying transforms that might break the real-life 30dB barrier (any sampling pattern @20% 256×256 data) in MRI. There has been much research into application of wavelets, discrete cosine transform (DCT), randomized orthogonal bases, splines, *etcetera*, but with suspiciously little focus on objective measures like image/error or illustration of difference images; the predominant basis of comparison instead being subjectively visual (Duensing & Huang, ISMRM Toronto 2008).[4.56] Despite choice of transform, there seems yet to have been a breakthrough of the 30dB barrier. Application of compressed sensing to MRI, therefore, remains fertile in 2008 for continued research.

Lagrangian form of compressed sensing in imaging

We now repeat Candès' image reconstruction experiment from 2004 which led to discovery of sparse sampling theorems. [73, §1.2] But we achieve perfect reconstruction with an algorithm based on vanishing gradient of a compressed sensing problem's Lagrangian, which is computationally efficient. Our *contraction method* (p.330) is fast also because matrix multiplications are replaced by fast Fourier transforms and number of constraints is cut in half by sampling symmetrically. Convex iteration for cardinality minimization (§4.5) is incorporated which allows perfect reconstruction of a phantom at 4.1% subsampling rate; 50% Candès' rate. By making neighboring-pixel selection adaptive, convex iteration reduces discrete image-gradient sparsity of the Shepp-Logan phantom to 1.9% ; 33% lower than previously reported.

We demonstrate application of discrete image-gradient sparsification to the $n\times n=256\times256$ Shepp-Logan phantom, simulating idealized acquisition of MRI data by radial sampling in the Fourier domain (Figure **115**).[4.57] Define a Nyquist-centric *discrete Fourier transform* (DFT) matrix

[4.54] Noise considered here is due only to the reconstruction process itself; *id est*, noise in excess of that produced by the best reconstruction of an image from a complete set of samples in the sense of Shannon. At less than 30dB image/error, artifacts generally remain visible to the naked eye. We estimate about 50dB is required to eliminate noticeable distortion in a visual A/B comparison.

[4.55] In vascular radiology, diagnoses are almost exclusively based on morphology of vessels and, in particular, presence of stenoses. There is a compelling argument for total-variation reconstruction of magnetic resonance angiogram because it helps isolate structures of particular interest.

[4.56] *I have never calculated the PSNR of these reconstructed images* [of Barbara]. −Jean-Luc Starck

The sparsity of the image is the percentage of transform coefficients sufficient for diagnostic-quality reconstruction. Of course the term "diagnostic quality" is subjective. ... I have yet to see an "objective" measure of image quality. Difference images, in my experience, definitely do not tell the whole story. Often I would show people some of my results and get mixed responses, but when I add artificial Gaussian noise to an image, often people say that it looks better. −Michael Lustig

[4.57] $\mathbb{k}$-*space* is conventional acquisition terminology indicating domain of the continuous raw data provided by an MRI machine. An image is reconstructed by inverse discrete Fourier transform of that data interpolated on a Cartesian grid in two dimensions.

$$F \triangleq \begin{bmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & e^{-\jmath 2\pi/n} & e^{-\jmath 4\pi/n} & e^{-\jmath 6\pi/n} & \cdots & e^{-\jmath(n-1)2\pi/n} \\ 1 & e^{-\jmath 4\pi/n} & e^{-\jmath 8\pi/n} & e^{-\jmath 12\pi/n} & \cdots & e^{-\jmath(n-1)4\pi/n} \\ 1 & e^{-\jmath 6\pi/n} & e^{-\jmath 12\pi/n} & e^{-\jmath 18\pi/n} & \cdots & e^{-\jmath(n-1)6\pi/n} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-\jmath(n-1)2\pi/n} & e^{-\jmath(n-1)4\pi/n} & e^{-\jmath(n-1)6\pi/n} & \cdots & e^{-\jmath(n-1)^2 2\pi/n} \end{bmatrix} \frac{1}{\sqrt{n}} \in \mathbb{C}^{n\times n} \tag{889}$$

a symmetric (nonHermitian) unitary matrix characterized

$$\begin{aligned} F &= F^{\mathrm{T}} \\ F^{-1} &= F^{\mathrm{H}} \end{aligned} \tag{890}$$

Denoting an unknown image $\mathcal{U}\in\mathbb{R}^{n\times n}$, its two-dimensional discrete Fourier transform $\mathfrak{F}$ is

$$\mathfrak{F}(\mathcal{U}) \triangleq F\mathcal{U}F \tag{891}$$

hence the inverse discrete transform

$$\mathcal{U} = F^{\mathrm{H}}\mathfrak{F}(\mathcal{U})F^{\mathrm{H}} \tag{892}$$

From §A.1.1 *no.*31 we have a vectorized two-dimensional DFT via Kronecker product $\otimes$

$$\operatorname{vec}\mathfrak{F}(\mathcal{U}) \triangleq (F\otimes F)\operatorname{vec}\mathcal{U} \tag{893}$$

and from (892) its inverse [182, p.24]

$$\operatorname{vec}\mathcal{U} \;=\; (F^{\mathrm{H}}\otimes F^{\mathrm{H}})(F\otimes F)\operatorname{vec}\mathcal{U} \;=\; (F^{\mathrm{H}}F\otimes F^{\mathrm{H}}F)\operatorname{vec}\mathcal{U} \tag{894}$$

Idealized radial sampling in the Fourier domain can be simulated by Hadamard product $\circ$ with a binary mask $\Phi\in\mathbb{R}^{n\times n}$ whose nonzero entries could, for example, correspond with the radial line segments in Figure **115**. To make the mask Nyquist-centric, like DFT matrix F, define a circulant [184] symmetric permutation matrix[4.58]

$$\Theta \triangleq \begin{bmatrix} \mathbf{0} & I \\ I & \mathbf{0} \end{bmatrix} \in \mathbb{S}^n \tag{895}$$

Then given subsampled Fourier domain (MRI $\Bbbk$-space) measurements in incomplete $K\in\mathbb{C}^{n\times n}$, we might constrain $\mathfrak{F}(\mathcal{U})$ thus:

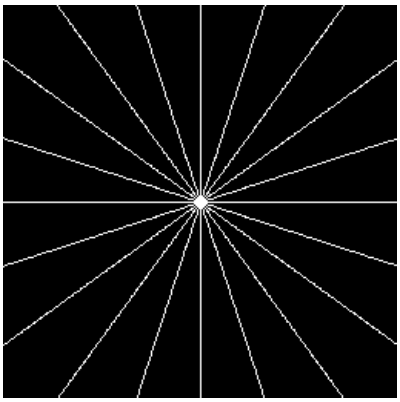
$$\Theta\Phi\Theta\circ F\mathcal{U}F = K \tag{896}$$

and in vector form, (42) (1888)

$$\delta(\operatorname{vec}\Theta\Phi\Theta)(F\otimes F)\operatorname{vec}\mathcal{U} = \operatorname{vec}K \tag{897}$$

Because measurements K are complex, there are actually twice the number of equality constraints as there are measurements.

[4.58] MATLAB `fftshift()`



Φ

Figure 115: MRI radial sampling pattern, in DC-centric Fourier domain, representing 4.1% (10 lines) subsampled data. Only half of these complex samples, in any halfspace about the origin in theory, need be acquired for a real image because of conjugate symmetry. Due to MRI machine imperfections, samples are generally taken over full extent of each radial line segment. MRI acquisition time is proportional to number of lines.

We can cut that number of constraints in half via vertical and horizontal mask Φ symmetry which forces the imaginary inverse transform to $\mathbf{0}$: The inverse subsampled transform in matrix form is

$$F^{\mathrm{H}}(\Theta\Phi\Theta \circ F\mathcal{U}F)F^{\mathrm{H}} = F^{\mathrm{H}}KF^{\mathrm{H}} \tag{898}$$

and in vector form

$$(F^{\mathrm{H}} \otimes F^{\mathrm{H}})\delta(\operatorname{vec}\Theta\Phi\Theta)(F\otimes F)\operatorname{vec}\mathcal{U} = (F^{\mathrm{H}}\otimes F^{\mathrm{H}})\operatorname{vec}K \tag{899}$$

later abbreviated

$$P\operatorname{vec}\mathcal{U} = f \tag{900}$$

where

$$P \triangleq (F^{\mathrm{H}}\otimes F^{\mathrm{H}})\delta(\operatorname{vec}\Theta\Phi\Theta)(F\otimes F) \in \mathbb{C}^{n^2\times n^2} \tag{901}$$

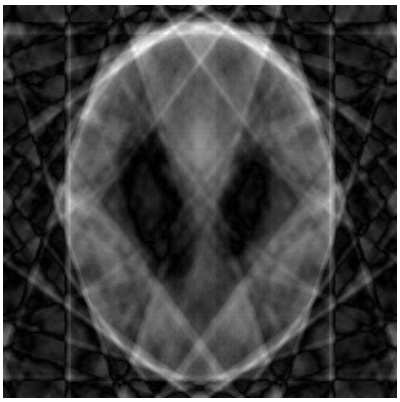
Because of idempotence $P = P^2$, P is a projection matrix. Because of its Hermitian symmetry [182, p.24]

$$P = (F^{\mathrm{H}}\otimes F^{\mathrm{H}})\delta(\operatorname{vec}\Theta\Phi\Theta)(F\otimes F) \;=\; (F\otimes F)^{\mathrm{H}}\delta(\operatorname{vec}\Theta\Phi\Theta)(F^{\mathrm{H}}\otimes F^{\mathrm{H}})^{\mathrm{H}} = P^{\mathrm{H}} \tag{902}$$

P is an orthogonal projector.[4.59] $P\operatorname{vec}\mathcal{U}$ is real when P is real; *id est*, when for positive even integer n

$$\Phi = \begin{bmatrix} \Phi_{11} & \Phi(1\,,2{:}n)\Xi \\ \Xi\Phi(2{:}n\,,1) & \Xi\Phi(2{:}n\,,2{:}n)\Xi \end{bmatrix} \in \mathbb{R}^{n\times n} \tag{903}$$

[4.59] (901) is a diagonalization of matrix P whose binary eigenvalues are $\delta(\operatorname{vec}\Theta\Phi\Theta)$ while the corresponding eigenvectors constitute the columns of unitary matrix $F^{\mathrm{H}}\otimes F^{\mathrm{H}}$.



$\mathrm{vec}^{-1} f$

Figure 116: Aliasing of Shepp-Logan phantom in Figure 114 resulting from $\Bbbk$-space subsampling pattern in Figure 115. This image is real because binary mask Φ is vertically and horizontally symmetric. It is remarkable that the phantom can be reconstructed, by convex iteration, given only $\mathcal{U}_0 = \mathrm{vec}^{-1} f$.

where $\Xi \in \mathbb{S}^{n-1}$ is the order-reversing permutation matrix (1828). In words, this necessary and sufficient condition on Φ (for a real inverse subsampled transform [301, p.53]) demands vertical symmetry about row $\frac{n}{2}+1$ and horizontal symmetry[4.60] about column $\frac{n}{2}+1$.

Define

$$\Delta \triangleq \begin{bmatrix} 1 & 0 & & & & \mathbf{0} \\ -1 & 1 & 0 & & & \\ & -1 & 1 & \ddots & & \\ & & \ddots & \ddots & \ddots & \\ & & & \ddots & 1 & 0 \\ \mathbf{0}^{\mathrm{T}} & & & & -1 & 1 \end{bmatrix} \in \mathbb{R}^{n\times n} \tag{904}$$

Express an image-gradient estimate

$$\nabla\mathcal{U} \triangleq \begin{bmatrix} \mathcal{U}\Delta \\ \mathcal{U}\Delta^{\mathrm{T}} \\ \Delta\mathcal{U} \\ \Delta^{\mathrm{T}}\mathcal{U} \end{bmatrix} \in \mathbb{R}^{4n\times n} \tag{905}$$

that is a simple first-order difference of neighboring pixels (Figure 117) to the right, left,

[4.60] This condition on Φ applies to both DC- and Nyquist-centric DFT matrices.

above, and below.[4.61] By §A.1.1 *no.*31, its vectorization: for $\Psi_i \in \mathbb{R}^{n^2\times n^2}$

$$\operatorname{vec}\nabla\mathcal{U} \;=\; \begin{bmatrix} \Delta^{\mathrm{T}}\otimes I \\ \Delta\otimes I \\ I\otimes\Delta \\ I\otimes\Delta^{\mathrm{T}} \end{bmatrix}\operatorname{vec}\mathcal{U} \;\triangleq\; \begin{bmatrix} \Psi_1 \\ \Psi_1^{\mathrm{T}} \\ \Psi_2 \\ \Psi_2^{\mathrm{T}} \end{bmatrix}\operatorname{vec}\mathcal{U} \;\triangleq\; \Psi\operatorname{vec}\mathcal{U} \in \mathbb{R}^{\mathbf{4}n^2} \tag{906}$$

where $\Psi\in\mathbb{R}^{\mathbf{4}n^2\times n^2}$. A total-variation minimization for reconstructing MRI image $\mathcal{U}$, that is known suboptimal [223] [74], may be concisely posed

$$\begin{array}{ll} \underset{\mathcal{U}}{\text{minimize}} & \|\Psi\operatorname{vec}\mathcal{U}\|_1 \\ \text{subject to} & P\operatorname{vec}\mathcal{U} = f \end{array} \tag{907}$$

where

$$f = (F^{\mathrm{H}}\otimes F^{\mathrm{H}})\operatorname{vec}K \in \mathbb{C}^{n^2} \tag{908}$$

is the known inverse subsampled Fourier data (a vectorized aliased image, Figure **116**), and where a norm of discrete image-gradient $\nabla\mathcal{U}$ is equivalently expressed as norm of a linear transformation $\Psi\operatorname{vec}\mathcal{U}$.

Although this simple problem statement (907) is equivalent to a linear program (§3.2), its numerical solution is beyond the capability of even the most highly regarded of contemporary commercial solvers.[4.62] Our only recourse is to recast the problem in Lagrangian form (§3.1.2.2.2) and write customized code to solve it:

$$\begin{array}{ll} \underset{\mathcal{U}}{\text{minimize}} & \langle\,|\Psi\operatorname{vec}\mathcal{U}|\,,\,y\,\rangle \\ \text{subject to} & P\operatorname{vec}\mathcal{U} = f \end{array} \qquad \text{(a)}$$

$$\equiv$$

$$\underset{\mathcal{U}}{\text{minimize}} \;\; \langle\,|\Psi\operatorname{vec}\mathcal{U}|\,,\,y\,\rangle \;+\; \tfrac{1}{2}\lambda\|P\operatorname{vec}\mathcal{U} - f\|_2^2 \qquad \text{(b)} \tag{909}$$

where multiobjective parameter $\lambda\in\mathbb{R}_+$ is quite large ($\lambda\approx$ `1E8`) so as to enforce the equality constraint: $P\operatorname{vec}\mathcal{U}-f=\mathbf{0} \;\Leftrightarrow\; \|P\operatorname{vec}\mathcal{U}-f\|_2^2=0$ (§A.7.1). We introduce a direction vector $y\in\mathbb{R}_+^{\mathbf{4}n^2}$ as part of a convex iteration (§4.5.3) to overcome that known suboptimal minimization of discrete image-gradient cardinality: *id est*, there exists a vector $y^\star$ with entries $y_i^\star\in\{0,1\}$ such that

$$\begin{array}{ll} \underset{\mathcal{U}}{\text{minimize}} & \|\Psi\operatorname{vec}\mathcal{U}\|_0 \\ \text{subject to} & P\operatorname{vec}\mathcal{U} = f \end{array} \;\equiv\; \underset{\mathcal{U}}{\text{minimize}}\; \langle\,|\Psi\operatorname{vec}\mathcal{U}|\,,\,y^\star\,\rangle \;+\; \tfrac{1}{2}\lambda\|P\operatorname{vec}\mathcal{U} - f\|_2^2 \tag{910}$$

Existence of such a $y^\star$, complementary to an optimal vector $\Psi\operatorname{vec}\mathcal{U}^\star$, is obvious by definition of global optimality $\langle\,|\Psi\operatorname{vec}\mathcal{U}^\star|\,,\,y^\star\rangle = 0$ (810) under which a cardinality-c optimal objective $\|\Psi\operatorname{vec}\mathcal{U}^\star\|_0$ is assumed to exist.

4.61 There is significant improvement in reconstruction quality by augmentation of a nominally two-point discrete image-gradient estimate to four points per pixel by inclusion of two polar directions. Improvement is due to centering; symmetry of discrete differences about a central pixel. We find small improvement on real-life images, $\approx$1dB empirically, by further augmentation with diagonally adjacent pixel differences.

4.62 for images as small as 128×128 pixels. Obstacle to numerical solution is not a computer resource: *e.g*, execution time, memory. The obstacle is, in fact, inadequate numerical precision. Even when all dependent equality constraints are manually removed, the best commercial solvers fail simply because computer numerics become nonsense; *id est*, numerical errors enter significant digits and the algorithm exits prematurely, loops indefinitely, or produces an infeasible solution.

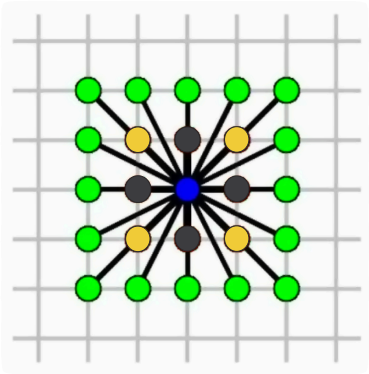


Figure 117: Neighboring-pixel stencil [375] for image-gradient estimation on Cartesian grid. Implementation selects adaptively from darkest four • about central. Continuous image-gradient from two pixels holds only in a limit. For discrete differences, better practical estimates are obtained when centered.

Because (909b) is an unconstrained convex problem, a zero objective function gradient is necessary and sufficient for optimality (§2.13.3); *id est*, (§D.2.1)

$$\Psi^{\mathrm{T}}\delta(y)\,\mathrm{sgn}(\Psi\,\mathrm{vec}\,\mathcal{U}) \;+\; \lambda P^{\mathrm{H}}(P\,\mathrm{vec}\,\mathcal{U} - f) = \mathbf{0} \tag{911}$$

Because of P idempotence and Hermitian symmetry and $\mathrm{sgn}(x) = x/|x|$, this is equivalent to

$$\lim_{\epsilon\to 0}\left(\Psi^{\mathrm{T}}\delta(y)\delta(|\Psi\,\mathrm{vec}\,\mathcal{U}| + \epsilon\mathbf{1})^{-1}\,\Psi + \lambda P\right)\mathrm{vec}\,\mathcal{U} \;=\; \lambda P f \tag{912}$$

where small positive constant $\epsilon\in\mathbb{R}_+$ has been introduced for invertibility. Speaking more analytically, introduction of ϵ serves to uniquely define the objective's gradient everywhere in the function domain; *id est*, it transforms absolute value in (909b) from a function differentiable almost everywhere into a differentiable function. An example of such a transformation in one dimension is illustrated in Figure **118**. When small enough for practical purposes[4.63] ($\epsilon\approx$`1E-3`), we may ignore the limiting operation. Then the mapping, for $0\preceq y\preceq\mathbf{1}$

$$\mathrm{vec}\,\mathcal{U}_{t+1} = \left(\Psi^{\mathrm{T}}\delta(y)\delta(|\Psi\,\mathrm{vec}\,\mathcal{U}_t| + \epsilon\mathbf{1})^{-1}\,\Psi + \lambda P\right)^{-1}\lambda P f \tag{913}$$

is a *contraction* in $\mathcal{U}_t$ that can be solved recursively in t for its unique *fixed point*; *id est*, until $\mathcal{U}_{t+1}\to\mathcal{U}_t$. [243, p.300] [219, p.155] Calculating this inversion directly is not possible

[4.63] We are looking for at least 50dB image/error ratio from only 4.1% subsampled data (10 radial lines in $\mathbb{k}$-space). With this setting of ϵ, we actually attain in excess of 100dB from a simple MATLAB program in about a minute on a 2006 vintage laptop Core 2 CPU (Intel T7400@2.16GHz, 666MHz FSB). By trading execution time and treating discrete image-gradient cardinality as a known quantity for this phantom, over 160dB is achievable.

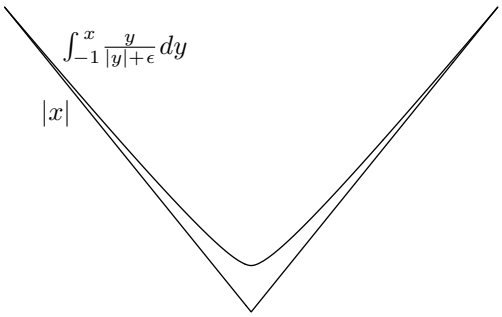


Figure 118: Real absolute value function $f_2(x) = |x|$ on $x \in [-1, 1]$ from Figure 72b superimposed upon integral of its derivative at $\epsilon = 0.05$ which smooths objective function.

for large matrices on contemporary computers because of numerical precision, so instead we apply the *conjugate gradient* method of solution to

$$\left(\Psi^{\mathrm{T}}\delta(y)\delta(|\Psi\,\mathrm{vec}\,\mathcal{U}_t| + \epsilon\mathbf{1})^{-1}\,\Psi + \lambda P\right)\mathrm{vec}\,\mathcal{U}_{t+1} \;=\; \lambda P f \tag{914}$$

which is linear in $\mathcal{U}_{t+1}$ at each recursion in the MATLAB program [390].[4.64]

Observe that P (901), in the equality constraint from problem (909a), is not a fat matrix.[4.65] Although number of Fourier samples taken is equal to the number of nonzero entries in binary mask Φ, matrix P is square but never actually formed during computation. Rather, a two-dimensional fast Fourier transform of $\mathcal{U}$ is computed followed by masking with $\Theta\Phi\Theta$ and then an inverse fast Fourier transform. This technique significantly reduces memory requirements and, together with contraction method of solution, is the principal reason for relatively fast computation.

convex iteration

By *convex iteration* we mean alternation of solution to (909a) and (915) until convergence. Direction vector y is initialized to $\mathbf{1}$ until the first fixed point is found; which means, the contraction recursion begins calculating a (1-norm) solution $\mathcal{U}^\star$ to (907) via problem (909b). Once $\mathcal{U}^\star$ is found, vector y is updated according to an estimate of discrete image-gradient cardinality c : Sum of the $4n^2 - c$ smallest entries of $|\Psi\,\mathrm{vec}\,\mathcal{U}^\star| \in \mathbb{R}^{\mathbf{4}n^2}$ is the optimal objective value from a linear program, for $0 \le c \le 4n^2 - 1$ (524)

$$\begin{aligned}\sum_{i=c+1}^{4n^2} \pi(|\Psi\,\mathrm{vec}\,\mathcal{U}^\star|)_i \;=\; &\underset{y\in\mathbb{R}^{\mathbf{4}n^2}}{\text{minimize}} && \langle\, |\Psi\,\mathrm{vec}\,\mathcal{U}^\star| \,,\, y\rangle \\ &\text{subject to} && 0 \preceq y \preceq \mathbf{1} \\ & && y^{\mathrm{T}}\mathbf{1} = 4n^2 - c\end{aligned} \tag{915}$$

where π is the nonlinear permutation-operator sorting its vector argument into nonincreasing order. *An optimal solution y to (915), that is an extreme point of its feasible*

[4.64] Conjugate gradient method requires positive definiteness. [166, §4.8.3.2]
[4.65] Fat is typical of compressed sensing problems; *e.g*, [72] [79].

set, is known in closed form: it has 1 *in each entry corresponding to the* $4n^2-c$ *smallest entries of* $|\Psi\,\mathrm{vec}\mathcal{U}^\star|$ *and has* 0 *elsewhere.* −p.293 Updated image $\mathcal{U}^\star$ is assigned to $\mathcal{U}_t$, the contraction is recomputed solving (909b), direction vector y is updated again, and so on until convergence which is guaranteed by virtue of a monotonically nonincreasing real sequence of objective values in (909a) and (915).

There are two features that distinguish problem formulation (909b) and our particular implementation of it [390, MATLAB code]:

1) An image-gradient estimate may engage any combination of four adjacent pixels. In other words, the algorithm is not locked into a four-point gradient estimate (Figure **117**); number of points constituting an estimate is directly determined by direction vector y .[4.66] Indeed, we find only $c=5092$ zero entries in $y^\star$ for the Shepp-Logan phantom; meaning, discrete image-gradient sparsity is actually closer to 1.9% than the 3% reported elsewhere; *e.g*, [374, §IIB].

2) Numerical precision of the fixed point of contraction (913) (≈`1E-2` for perfect reconstruction @−103dB error) is a parameter to the implementation; meaning, direction vector y is updated after contraction begins but prior to its culmination. Impact of this idiosyncrasy tends toward simultaneous optimization in variables $\mathcal{U}$ and y while insuring y settles on a boundary point of its feasible set (nonnegative hypercube slice) in (915) at every iteration; for only a boundary point[4.67] can yield the sum of smallest entries in $|\Psi\,\mathrm{vec}\mathcal{U}^\star|$.

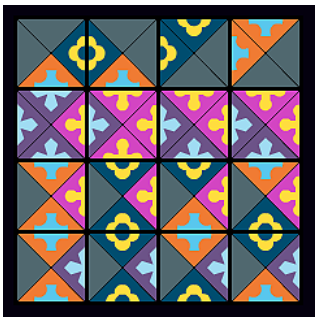
Perfect reconstruction of the Shepp-Logan phantom (at 103dB image/error) is achieved in a MATLAB minute with 4.1% subsampled data (2671 complex samples); well below an 11% least lower bound predicted by the sparse sampling theorem. Because reconstruction approaches optimal solution to a 0-norm problem, minimum number of Fourier-domain samples is bounded below by cardinality of discrete image-gradient at 1.9%. □

4.6.0.0.14 Exercise. *Contraction operator.*
Determine conditions on λ and ϵ under which $\Psi^{\mathrm{T}}\delta(y)\delta(|\Psi\,\mathrm{vec}\mathcal{U}_t|+\epsilon\mathbf{1})^{-1}\,\Psi+\lambda P$ from (914) is positive definite and (913) is a contraction. ▼

4.6.0.0.15 Example. *Eternity* II.
A tessellation puzzle game, playable by children, commenced world-wide in July 2007; introduced in London by Christopher Walter Monckton, 3rd Viscount Monckton of Brenchley. Called Eternity II, its name derives from an estimate of time that would pass while trying all allowable tilings of puzzle pieces before obtaining a complete solution. By the end of 2008, a complete solution had not yet been found although a $10,000 USD prize was awarded for a high score 467 (out of $480=2\sqrt{M}(\sqrt{M}-1)$) obtained by heuristic methods.[4.68] No prize was awarded for 2009 and 2010. Game-rules state that a $2M prize

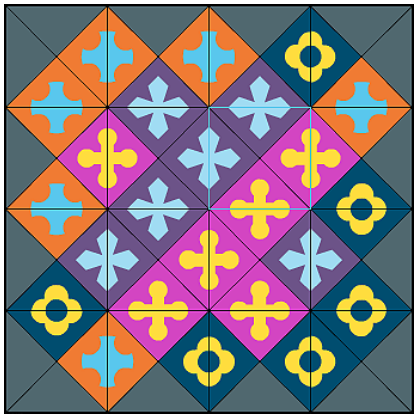
[4.66] This adaptive gradient was not contrived. It is an artifact of the convex iteration method for minimal cardinality solution; in this case, cardinality minimization of a discrete image-gradient.
[4.67] Simultaneous optimization of these two variables $\mathcal{U}$ and y should never be a pinnacle of aspiration; for then, optimal y might not attain a boundary point.
[4.68] That score means all but a few of the 256 pieces had been placed successfully (including the mandatory piece). Although distance between 467 to 480 is relatively small, there is apparently vast distance to a solution because no complete solution was found in 2009.

(a) pieces



1	5	9	13
2	6	10	14
3	7	11	15
4	8	12	16

(b) one solution



13	4	16	5
3	2	10	11
12	14	6	8
1	15	7	9

(c) colors

Figure 119: *Eternity* II is a board game in the puzzle genre. **(a)** Shown are all of the 16 puzzle pieces (indexed as in the tableau alongside) from a scaled-down computerized demonstration-version on the TOMY website. Puzzle pieces are square and triangularly partitioned into four colors (with associated symbols). Pieces may be moved, removed, and rotated at random on a 4×4 board. **(b)** Illustrated is one complete solution to this puzzle whose solution is not unique. The piece, whose border is lightly outlined, was placed last in this realization. There is no mandatory piece placement as for the full game, except the grey board-boundary. Solution time for a human is typically on the order of a minute. **(c)** This puzzle has four colors, indexed 1 through 4; grey corresponds to **0**.

would be awarded to the first person who completely solves the puzzle before December 31, 2010, but the prize remains unclaimed after the deadline.

The full game comprises $M = 256$ square pieces and a 16×16 gridded board (Figure **120**) whose complete tessellation is considered *NP-hard.*[4.69] [361] [113] A player may tile, retile, and rotate pieces, indexed 1 through 256, in any order face-up on the square board. Pieces are immutable in the sense that each is characterized by 4 colors (and their uniquely associated symbols), one at each edge, which are not necessarily the same per piece or from piece to piece; *id est*, different pieces may or may not have some edge-colors in common. There are $L = 22$ distinct edge-colors plus a solid grey. The object of the game is to completely tile the board with pieces whose touching edges have identical color. The boundary of the board must be colored grey.

full-game rules

1) Any puzzle piece may be rotated face-up in quadrature and placed or replaced on the square board.

2) Only one piece may occupy any particular cell on the board.

3) All adjacent pieces must match in color (and symbol) at their touching edges.

4) Solid grey edges must appear all along the board's boundary.

5) One mandatory piece (numbered 139 in the full game) must have a predetermined orientation in a predetermined cell on the board.

6) The board must be tiled completely (*covered*).

A scaled-down demonstration version of the game is illustrated in Figure **119**. Differences between the full game (Figure **120**) and scaled-down game are the number of edge-colors L (22 *versus* 4, ignoring solid grey), number of pieces M (256 *versus* 16), and a single mandatory piece placement interior to the board in the full game. The scaled-down game has four distinct edge-colors, plus a solid grey, whose coding is illustrated in Figure **119**c.

- $L = 22$ distinct edge-colors and number of puzzle pieces $M = 256$ and board-dimension $\sqrt{M} \times \sqrt{M} = 16 \times 16$ for the full game.

- There are $L = 4$ distinct edge-colors and $M = 16$ pieces and dimension $\sqrt{M} \times \sqrt{M} = 4 \times 4$ for the scaled-down demonstration board.

[4.69] Even so, combinatorial-intensity brute-force backtracking methods can solve similar puzzles in minutes given $M = 196$ pieces on a 14×14 test board; as demonstrated by Yannick Kirschhoffer. There is a steep rise in level of difficulty going to a 15×15 board.

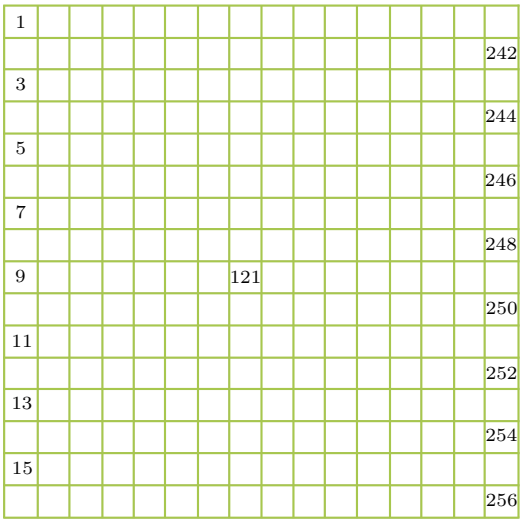


Figure 120: *Eternity* II full-game board (16×16, not actual size). Grid facilitates piece placement within unit-square cell; one piece per cell.

Euclidean distance intractability

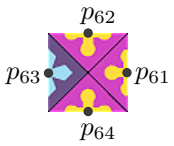
If each square puzzle piece were characterized by four points in quadrature, one point representing board coordinates and color per edge, then Euclidean distance geometry would be suitable for solving this puzzle. Since all interpoint distances per piece are known, this game may be regarded as a Euclidean distance matrix completion problem[4.70] in $\mathbb{EDM}^{\mathbf{4}M}$. Because distance information provides for reconstruction of point position to within an isometry (§5.5), piece translation and rotation are isometric transformations that abide by rules of the game.[4.71] Convex constraints can be devised to prevent puzzle-piece reflection and to quantize rotation such that piece-edges stay aligned with the board boundary. (§5.5.2.0.1)

But manipulating such a large EDM is too numerically difficult for contemporary general-purpose semidefinite program solvers which incorporate interior-point methods; indeed, they are hard-pressed to find a solution for variable matrices of dimension as small as 100. Our challenge, therefore, is to express this game's rules as constraints in a convex and numerically tractable way so as to find one solution from a googol of possible combinations.[4.72]

[4.70] (§6.7) Were edge-points ordered sequentially with piece number, then this EDM would have a block-diagonal structure of known entries.

[4.71] Translation occurs when a piece moves on the board in Figure **120**, rotation occurs when colors are aligned with a neighboring piece.

[4.72] There exists at least one solution; their exact number is unknown although Monckton insists they number in thousands. Ignoring boundary constraints and the single mandatory piece placement in the full game, a loose upper bound on number of combinations is $M!\,4^M = 256!\,4^{256}$. That number gets loosened: $150638!/(256!(150638-256)!)$ after presolving Eternity II (937).



$$P_6 = [\, p_{61} \;\; p_{62} \;\; p_{63} \;\; p_{64} \,]^{\mathrm{T}} \in \mathbb{R}^{\mathbf{4}\times L}$$

Figure 121: Demo-game piece P_6 illustrating edge-color $\bullet$ $p_{6j} \in \mathbb{R}^L$ counterclockwise ordering in j beginning from right.

permutation polyhedron strategy

To each puzzle piece, from a set of M pieces $\{P_i\,,\ i=1\ldots M\}$, assign an index i representing a unique piece-number. Each square piece is characterized by four colors, in quadrature, corresponding to its four edges. Each color $p_{ij} \in \mathbb{R}^L$ is represented by $e_\ell \in \mathbb{R}^L$ an L-dimensional standard basis vector or $\mathbf{0}$ if grey. These four edge-colors are represented in a $4\times L$-dimensional matrix; one matrix per piece

$$P_i \triangleq [\, p_{i1} \;\; p_{i2} \;\; p_{i3} \;\; p_{i4} \,]^{\mathrm{T}} \in \mathbb{R}^{\mathbf{4}\times L}, \qquad i=1\ldots M \tag{916}$$

In other words, each distinct nongrey color is assigned a unique corresponding index $\ell \in \{1\ldots L\}$ identifying a standard basis vector $e_\ell \in \mathbb{R}^L$ (Figure 119c) that becomes a vector $p_{ij} \in \{e_1 \ldots e_L\,, \mathbf{0}\} \subset \mathbb{R}^L$ constituting matrix P_i representing a particular piece. Rows $\{p_{ij}^{\mathrm{T}}\,,\ j=1\ldots 4\}$ of P_i are ordered counterclockwise as in Figure 121. Color data is given in Figure 122 for the demonstration board. Then matrix P_i describes the i^{th} piece, excepting its orientation and location on the board.

Our strategy to solve Eternity II is to first vectorize the board, with respect to the whole pieces, and then relax a very hard combinatorial problem: All pieces are initially placed, as in Figure 122, in order of their given index. Then the vectorized game-board has initial state, as in Figure 122, represented within a matrix

$$P \;\triangleq\; \begin{bmatrix} P_1 \\ \vdots \\ P_M \end{bmatrix} \in \mathbb{R}^{\mathbf{4}M\times L} \tag{917}$$

Moving pieces about the square board all at once corresponds to permuting pieces P_i on the vectorized board represented by matrix P, while rotating the i^{th} piece is equivalent to circularly shifting row indices of P_i (rowwise permutation). This permutation problem, as stated, is doubly combinatorial ($M!\,4^M$ combinations) because we must find a permutation of pieces ($M!$)

$$\Xi \in \mathbb{R}^{M\times M} \tag{918}$$

and a rotation $\Pi_i \in \mathbb{R}^{\mathbf{4}\times\mathbf{4}}$ of each individual piece (4^M) that solve the puzzle;

$$(\Xi \otimes I_4)\Pi P \;=\; (\Xi \otimes I_4)\begin{bmatrix} \Pi_1 P_1 \\ \vdots \\ \Pi_M P_M \end{bmatrix} \in \mathbb{R}^{\mathbf{4}M\times L} \tag{919}$$

$$
\begin{array}{ll}
P_1 & [\, e_3 \quad \mathbf{0} \quad \mathbf{0} \quad e_1 \,]^{\mathrm{T}} \\
P_2 & [\, e_2 \quad e_4 \quad e_4 \quad e_4 \,]^{\mathrm{T}} \\
P_3 & [\, e_2 \quad e_1 \quad \mathbf{0} \quad e_1 \,]^{\mathrm{T}} \\
P_4 & [\, e_4 \quad e_1 \quad \mathbf{0} \quad e_1 \,]^{\mathrm{T}} \\
P_5 & [\, \mathbf{0} \quad \mathbf{0} \quad e_3 \quad e_1 \,]^{\mathrm{T}} \\
P_6 & [\, e_2 \quad e_2 \quad e_4 \quad e_2 \,]^{\mathrm{T}} \\
P_7 & [\, e_2 \quad e_3 \quad \mathbf{0} \quad e_3 \,]^{\mathrm{T}} \\
P_8 & [\, e_4 \quad e_3 \quad \mathbf{0} \quad e_3 \,]^{\mathrm{T}} \\
P_9 & [\, \mathbf{0} \quad e_3 \quad e_3 \quad \mathbf{0} \,]^{\mathrm{T}} \\
P_{10} & [\, e_2 \quad e_2 \quad e_4 \quad e_4 \,]^{\mathrm{T}} \\
P_{11} & [\, e_2 \quad e_3 \quad \mathbf{0} \quad e_1 \,]^{\mathrm{T}} \\
P_{12} & [\, e_4 \quad e_1 \quad \mathbf{0} \quad e_3 \,]^{\mathrm{T}} \\
P_{13} & [\, \mathbf{0} \quad e_1 \quad e_1 \quad \mathbf{0} \,]^{\mathrm{T}} \\
P_{14} & [\, e_2 \quad e_2 \quad e_4 \quad e_4 \,]^{\mathrm{T}} \\
P_{15} & [\, e_2 \quad e_1 \quad \mathbf{0} \quad e_3 \,]^{\mathrm{T}} \\
P_{16} & [\, e_4 \quad e_3 \quad \mathbf{0} \quad e_1 \,]^{\mathrm{T}}
\end{array}
$$

Figure 122: Vectorized demo-game board has $M = 16$ matrices in $\mathbb{R}^{\mathbf{4}\times L}$ describing initial state of game pieces; 4 colors per puzzle-piece (Figure 121), $L = 4$ colors total in game (Figure 119c). So that color difference measurement remains unweighted, standard basis vectors in $\mathbb{R}^L$ represent color.

where

$$\Pi_i \in \{\pi_1\ ,\ \pi_2\ ,\ \pi_3\ ,\ \pi_4\} \triangleq \left\{ \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 0&0&0&1 \end{bmatrix}, \begin{bmatrix} 0&1&0&0\\ 0&0&1&0\\ 0&0&0&1\\ 1&0&0&0 \end{bmatrix}, \begin{bmatrix} 0&0&1&0\\ 0&0&0&1\\ 1&0&0&0\\ 0&1&0&0 \end{bmatrix}, \begin{bmatrix} 0&0&0&1\\ 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0 \end{bmatrix} \right\} \tag{920}$$

$$\Pi \triangleq \begin{bmatrix} \Pi_1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \Pi_M \end{bmatrix} \in \mathbb{R}^{\mathbf{4}M\times\mathbf{4}M} \tag{921}$$

Initial game-board state P (917) corresponds to $\Xi = I$ and $\Pi_i = \pi_1 = I\ \forall i$. Circulant [184] permutation matrices $\{\pi_1\ ,\ \pi_2\ ,\ \pi_3\ ,\ \pi_4\} \subset \mathbb{R}^{\mathbf{4}\times\mathbf{4}}$ correspond to clockwise piece-rotations $\{0°,\ 90°,\ 180°,\ 270°\}$.

Rules of the game dictate that adjacent pieces on the square board have colors that match at their touching edges as in Figure 119b.[4.73] A complete match is therefore equivalent to demanding that a constraint, comprising numeric color differences between $2\sqrt{M}(\sqrt{M}-1)$ touching edges, vanish. Because the vectorized board layout is fixed and its cells are loaded or reloaded with pieces during play, locations of adjacent edges in $\mathbb{R}^{\mathbf{4}M\times L}$ are known *a priori*. We need simply form differences between colors from adjacent edges of pieces loaded into those known locations (Figure 123). Each difference may be represented by a constant vector from a set $\{\Delta_i \in \mathbb{R}^{\mathbf{4}M},\ i = 1\ldots 2\sqrt{M}(\sqrt{M}-1)\}$. Defining sparse constant fat matrix

$$\Delta \triangleq \begin{bmatrix} \Delta_1^{\mathrm{T}} \\ \vdots \\ \Delta_{2\sqrt{M}(\sqrt{M}-1)}^{\mathrm{T}} \end{bmatrix} \in \mathbb{R}^{\mathbf{2}\sqrt{M}(\sqrt{M}-1)\times\mathbf{4}M} \tag{922}$$

whose entries belong to $\{-1, 0, 1\}$, then the desired constraint is

$$\Delta(\Xi \otimes I_4)\Pi P = \mathbf{0} \in \mathbb{R}^{\mathbf{2}\sqrt{M}(\sqrt{M}-1)\times L} \tag{923}$$

Boundary of the square board must be colored grey. This means there are $4\sqrt{M}$ boundary locations in $\mathbb{R}^{\mathbf{4}M\times L}$ that must have value $\mathbf{0}^{\mathrm{T}}$. These can all be lumped into one equality constraint

$$\beta^{\mathrm{T}}(\Xi \otimes I_4)\Pi P\mathbf{1} = 0 \tag{924}$$

where $\beta \in \mathbb{R}^{\mathbf{4}M}$ is a sparse constant vector having entries in $\{0, 1\}$ complementary to the known $4\sqrt{M}$ zeros (Figure 123).

By combining variables:

$$\Phi \triangleq (\Xi \otimes I_4)\Pi \in \mathbb{R}^{\mathbf{4}M\times\mathbf{4}M} \tag{925}$$

this square matrix becomes a structured permutation matrix replacing the product of permutation matrices. Partition the composite variable Φ into blocks

$$\Phi \triangleq \begin{bmatrix} \phi_{11} & \cdots & \phi_{1M} \\ \vdots & \ddots & \vdots \\ \phi_{M1} & \cdots & \phi_{MM} \end{bmatrix} \in \mathbb{R}^{\mathbf{4}M\times\mathbf{4}M} \tag{926}$$

[4.73] Piece adjacencies on the square board map linearly to the vectorized board, of course.

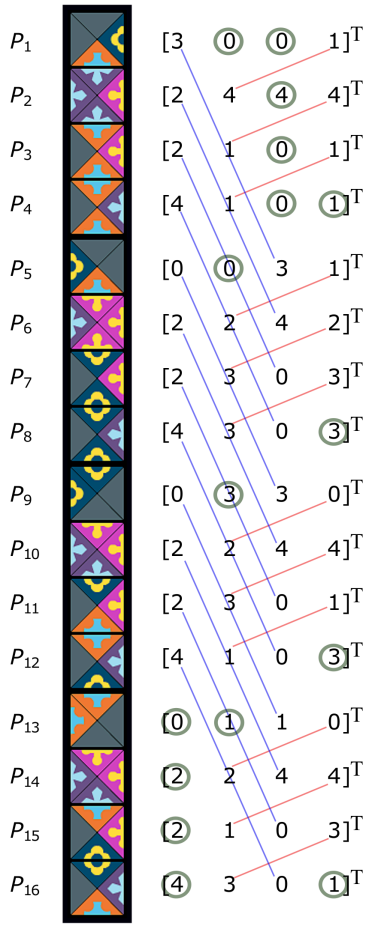
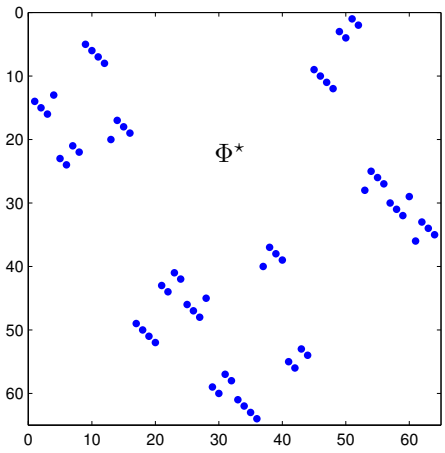


Figure 123: Initial piece state. ◯ indicate boundary β, line segments indicate differences Δ (922). Entries are indices ℓ identifying standard basis vectors $e_\ell \in \mathbb{R}^L$ from Figure **122**.



$M = 16$

Figure 124: Sparsity pattern for composite permutation matrix $\Phi^\star \in \mathbb{R}^{4M\times 4M}$ representing solution from Figure 119b. Each four-point cluster represents a circulant permutation matrix from (920). Any $M = 16$-piece solution may be verified on the TOMY website.

where $\Phi^\star_{ij} \in \{0, 1\}$ because (920)

$$\phi^\star_{ij} \in \{\pi_1\,,\ \pi_2\,,\ \pi_3\,,\ \pi_4\,,\ \mathbf{0}\} \subset \mathbb{R}^{4\times 4} \tag{927}$$

An optimal composite permutation matrix $\Phi^\star$ is represented pictorially in Figure 124. Now we ask what are necessary conditions on $\Phi^\star$ at optimality:

- $4M$-sparsity and nonnegativity.
- Each column has one 1. Each row has one 1.
- Entries along each and every diagonal of each and every 4×4 block $\phi^\star_{ij}$ are equal.
- Corner pair of 2×2 submatrices on antidiagonal of each and every 4×4 block $\phi^\star_{ij}$ are equal.

We want an objective function whose global optimum, if attained, certifies that the puzzle has been solved. Then, in terms of this Φ partitioning (926), the Eternity II problem is a minimization of cardinality

$$\begin{array}{ll} \underset{\Phi\in\mathbb{R}^{4M\times 4M}}{\text{minimize}} & \|\operatorname{vec}\Phi\|_0 \\ \text{subject to} & \Delta\,\Phi P = \mathbf{0} \\ & \beta^{\mathrm{T}}\Phi P\mathbf{1} = 0 \\ & \Phi\mathbf{1} = \mathbf{1} \\ & \Phi^{\mathrm{T}}\mathbf{1} = \mathbf{1} \\ & (I\otimes R_{\mathrm{d}})\Phi(I\otimes R_{\mathrm{d}}^{\mathrm{T}}) = (I\otimes S_{\mathrm{d}})\Phi(I\otimes S_{\mathrm{d}}^{\mathrm{T}}) \\ & (I\otimes R_{\phi})\Phi(I\otimes S_{\phi}^{\mathrm{T}}) = (I\otimes S_{\phi})\Phi(I\otimes R_{\phi}^{\mathrm{T}}) \\ & (e_{121}\otimes I_4)^{\mathrm{T}}\Phi\,(e_{139}\otimes I_4) = \pi_3 \\ & \Phi \succeq \mathbf{0} \end{array} \tag{928}$$

which is convex in the constraints where $e_{121}, e_{139} \in \mathbb{R}^M$ are members of the standard basis representing mandatory piece placement in the full game,[4.74] and where

$$R_{\mathrm{d}} \triangleq \begin{bmatrix} 1 & 0 & & \\ & 1 & 0 & \\ & & 1 & 0 \end{bmatrix} \in \mathbb{R}^{\mathbf{3}\times\mathbf{4}}, \quad S_{\mathrm{d}} \triangleq \begin{bmatrix} 0 & 1 & & \\ & 0 & 1 & \\ & & 0 & 1 \end{bmatrix} \in \mathbb{R}^{\mathbf{3}\times\mathbf{4}} \tag{929}$$

$$R_{\phi} \triangleq \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \in \mathbb{R}^{\mathbf{2}\times\mathbf{4}}, \quad S_{\phi} \triangleq \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{\mathbf{2}\times\mathbf{4}} \tag{930}$$

These matrices R and S enforce circulance.[4.75] Mandatory piece placement in the full game requires the equality constraint in π_3. Constraints $\Phi\mathbf{1}=\mathbf{1}$ and $\Phi^{\mathrm{T}}\mathbf{1}=\mathbf{1}$ confine nonnegative Φ to the permutation polyhedron (100) in $\mathbb{R}^{\mathbf{4}M\times\mathbf{4}M}$. The feasible set of problem (928) is an intersection of the permutation polyhedron with a large number of hyperplanes. Any vertex in the permutation polyhedron, which is the convex hull of permutation matrices, has minimal cardinality. (§2.3.2.0.4) The intersection contains a vertex of the permutation polyhedron because a solution $\Phi^{\star}$ cannot otherwise be a permutation matrix; such a solution is known to exist, so it must also be a vertex of the intersection.[4.76]

In the vectorized variable, problem (928) is equivalent to

$$\begin{array}{ll} \underset{\Phi\in\mathbb{R}^{\mathbf{4}M\times\mathbf{4}M}}{\text{minimize}} & \|\operatorname{vec}\Phi\|_0 \\ \text{subject to} & (P^{\mathrm{T}}\otimes\Delta)\operatorname{vec}\Phi = \mathbf{0} \\ & (P\mathbf{1}\otimes\beta)^{\mathrm{T}}\operatorname{vec}\Phi = 0 \\ & (\mathbf{1}_{4M}^{\mathrm{T}}\otimes I_{4M})\operatorname{vec}\Phi = \mathbf{1}_{4M} \\ & (I_{4M}\otimes\mathbf{1}_{4M}^{\mathrm{T}})\operatorname{vec}\Phi = \mathbf{1}_{4M} \\ & (I\otimes R_{\mathrm{d}}\otimes I\otimes R_{\mathrm{d}} - I\otimes S_{\mathrm{d}}\otimes I\otimes S_{\mathrm{d}})\operatorname{vec}\Phi = \mathbf{0} \\ & (I\otimes S_{\phi}\otimes I\otimes R_{\phi} - I\otimes R_{\phi}\otimes I\otimes S_{\phi})\operatorname{vec}\Phi = \mathbf{0} \\ & (e_{139}\otimes I_4\otimes e_{121}\otimes I_4)^{\mathrm{T}}\operatorname{vec}\Phi = \operatorname{vec}\pi_3 \\ & \Phi \geq \mathbf{0} \end{array} \tag{931}$$

This problem is abbreviated:

$$\begin{array}{ll} \underset{\Phi\in\mathbb{R}^{\mathbf{4}M\times\mathbf{4}M}}{\text{minimize}} & \|\operatorname{vec}\Phi\|_0 \\ \text{subject to} & E\operatorname{vec}\Phi = \tau \\ & \Phi \geq \mathbf{0} \end{array} \tag{932}$$

where $E \in \mathbb{R}^{\mathbf{2}L\sqrt{M}(\sqrt{M}-1)+\mathbf{8}M+\mathbf{13}M^{\mathbf{2}}+\mathbf{17}\times\mathbf{16}M^{\mathbf{2}}}$ is sparse and optimal objective value is $4M$; dimension of E is 864,593 × 1,048,576. A compressed sensing paradigm [73] is not inherent here. To solve this by linear programming, a direction vector is introduced for cardinality minimization as in §4.5. It is critical, in this case, to add enough random noise to the

[4.74] meaning that piece numbered 139 by the game designer must be placed in cell 121 on the vectorized board (Figure **120**) with orientation π_3 (p.338).

[4.75] Since $\mathbf{0}$ is the trivial circulant matrix, application is democratic over all blocks ϕ_{ij}.

[4.76] *Vertex* means *zero-dimensional exposed face* (§2.6.1.0.1); intersection with a strictly supporting hyperplane. There can be no further intersection with a feasible affine subset that would enlarge that face; *id est*, a vertex of the permutation polyhedron persists in the feasible set.

direction vector so as to insure a vertex solution [98, p.158]. For the demonstration game, in fact, choosing a direction vector randomly will find an optimal solution in only a few iterations.[4.77] But for the full game, numerical errors prevent solution of (932); number of equality constraints 864,593 is too large.[4.78] So again, we reformulate the problem:

canonical Eternity II

Because each block ϕ_{ij} of Φ (926) is optimally circulant having only four degrees of freedom (927), we may take as variable every fourth column of Φ:

$$\tilde{\Phi} \triangleq [\, \Phi(:,1) \;\; \Phi(:,5) \;\; \Phi(:,9) \;\; \cdots \;\; \Phi(:,4M-3) \,] \in \mathbb{R}^{\mathbf{4}M\times M} \tag{933}$$

where $\tilde{\Phi}_{ij} \in \{0,1\}$. Then, for $e_i \in \mathbb{R}^{\mathbf{4}}$

$$\Phi = (\tilde{\Phi}\otimes e_1^{\mathrm{T}}) + (I\otimes\pi_4)(\tilde{\Phi}\otimes e_2^{\mathrm{T}}) + (I\otimes\pi_3)(\tilde{\Phi}\otimes e_3^{\mathrm{T}}) + (I\otimes\pi_2)(\tilde{\Phi}\otimes e_4^{\mathrm{T}}) \in \mathbb{R}^{\mathbf{4}M\times\mathbf{4}M} \tag{934}$$

From §A.1.1 *no.*31 and *no.*40

$$\begin{aligned}\operatorname{vec}\Phi &= (I\otimes e_1\otimes I_{4M} + I\otimes e_2\otimes I\otimes\pi_4 + I\otimes e_3\otimes I\otimes\pi_3 + I\otimes e_4\otimes I\otimes\pi_2)\operatorname{vec}\tilde{\Phi}\\ &\triangleq Y\operatorname{vec}\tilde{\Phi} \in \mathbb{R}^{\mathbf{16}M^{\mathbf{2}}}\end{aligned} \tag{935}$$

where $Y \in \mathbb{R}^{\mathbf{16}M^{\mathbf{2}}\times\mathbf{4}M^{\mathbf{2}}}$. Permutation polyhedron (100) now demands that each consecutive quadruple of adjacent rows of $\tilde{\Phi}$ sum to 1: $(I\otimes\mathbf{1}_4^{\mathrm{T}})\tilde{\Phi}\mathbf{1}=\mathbf{1}$. Constraints in R and S (which are most numerous) may be dropped because circulance of ϕ_{ij} is built into Φ-reconstruction formula (934). Eternity II (931) is thus equivalently transformed

$$\begin{array}{ll}\underset{\tilde{\Phi}\in\mathbb{R}^{\mathbf{4}M\times M}}{\text{minimize}} & \|\operatorname{vec}\tilde{\Phi}\|_0\\ \text{subject to} & (P^{\mathrm{T}}\otimes\Delta)Y\operatorname{vec}\tilde{\Phi} = \mathbf{0}\\ & (P\mathbf{1}\otimes\beta)^{\mathrm{T}}Y\operatorname{vec}\tilde{\Phi} = 0\\ & (\mathbf{1}^{\mathrm{T}}\otimes I\otimes\mathbf{1}_4^{\mathrm{T}})\operatorname{vec}\tilde{\Phi} = \mathbf{1}\\ & (I\otimes\mathbf{1}_{4M}^{\mathrm{T}})\operatorname{vec}\tilde{\Phi} = \mathbf{1}\\ & (e_{139}\otimes e_1\otimes e_{121}\otimes I_4)^{\mathrm{T}}Y\operatorname{vec}\tilde{\Phi} = \pi_3 e_1\\ & \tilde{\Phi}\succeq\mathbf{0}\end{array} \tag{936}$$

whose optimal objective value is M with $\tilde{\Phi}^{\star}$-entries in $\{0,1\}$ and where $e_1\in\mathbb{R}^{\mathbf{4}}$ (§A.1.1 *no.*39) and $e_{121}\,, e_{139}\in\mathbb{R}^M$. In abbreviation

$$\begin{array}{ll}\underset{\tilde{\Phi}\in\mathbb{R}^{\mathbf{4}M\times M}}{\text{minimize}} & \|\operatorname{vec}\tilde{\Phi}\|_0\\ \text{subject to} & \tilde{E}\operatorname{vec}\tilde{\Phi} = \tilde{\tau}\\ & \tilde{\Phi}\succeq\mathbf{0}\end{array} \tag{937}$$

[4.77] This can only mean: there are many optimal solutions. A simplex-method solver is required for numerical solution; interior-point methods will not work. A randomized direction vector also works for Clue Puzzles provided by the toy maker: similar 6×6 and 6×12 puzzles whose solution each provide a clue to solution of the full game. Even better is a nonnegative uniformly distributed randomized direction vector having $4M$ entries (M entries, in case (937)), corresponding to the largest entries of $\Phi^{\star}$, zeroed.

[4.78] Saunders' program `lusol` can reduce that number to 797,508 constraints by eliminating linearly dependent rows of matrix E, but that reduction is not enough to overcome numerical issues with the best contemporary linear program solvers.

of reformulation (936), number of equality constraints is now 11,077; an order of magnitude fewer constraints than (932) from where sparse $\tilde{E} \in \mathbb{R}^{\mathbf{2}L\sqrt{M}(\sqrt{M}-1)+\mathbf{2}M+\mathbf{5}\times\mathbf{4}M^{\mathbf{2}}}$ replaces the E matrix. Number of columns in matrix $\tilde{E}$ has been reduced from a million to 262,144; dimension of $\tilde{E}$ goes to $11{,}077 \times 262{,}144$. But this dimension remains out of reach of most highly regarded contemporary academic and commercial linear program solvers because of numerical failure; especially disappointing insofar as sparsity of $\tilde{E}$ is high with only 0.07% nonzero entries $\in \{-1, 0, 1, 2\}$; element $\{2\}$ arising only in the β constraint which is soon to disappear after presolving.

Variable $\operatorname{vec}\tilde{\Phi}$ itself is too large in dimension. Notice that the constraint in β, which zeroes the board at its edges, has all positive coefficients. The zero sum means that all $\operatorname{vec}\tilde{\Phi}$ entries, corresponding to nonzero entries in row vector $(P\mathbf{1} \otimes \beta)^{\mathrm{T}}Y$, must be zero. For the full game, this means we may immediately eliminate 57,840 variables from 262,144. After zero-row and dependent row removal, dimension of $\tilde{E}$ goes to $10{,}054 \times 204{,}304$ with entries in $\{-1, 0, 1\}$.

polyhedral cone theory

Eternity II problem (937) constraints are interpretable in the language of convex cones: The columns of matrix $\tilde{E}$ constitute a set of generators for a pointed polyhedral cone $\mathcal{K} = \{\tilde{E}\operatorname{vec}\tilde{\Phi} \mid \tilde{\Phi} \succeq \mathbf{0}\}$. (§2.12.2.2) Even more intriguing is the observation: vector $\tilde{\tau}$ resides on that polyhedral cone's boundary. (§2.13.4.2.4)

We may apply techniques from §2.13.4.3 to prune generators not belonging to the smallest face of that cone, to which $\tilde{\tau}$ belongs, because generators of that smallest face must hold a minimal cardinality solution. Matrix dimension is thereby reduced:[4.79] The i^{th} column $\tilde{E}(:, i)$ of matrix $\tilde{E}$ belongs to the smallest face $\mathcal{F}$ of $\mathcal{K}$ that contains $\tilde{\tau}$ if and only if

$$\begin{array}{ll} \underset{\tilde{\Phi}\in\mathbb{R}^{\mathbf{4}M\times M},\ \mu\in\mathbb{R}}{\text{find}} & \tilde{\Phi},\ \mu \\ \text{subject to} & \mu\tilde{\tau} - \tilde{E}(:, i) = \tilde{E}\operatorname{vec}\tilde{\Phi} \\ & \operatorname{vec}\tilde{\Phi} \succeq 0 \end{array} \tag{375}$$

is feasible. By a transformation of Saunders, this linear feasibility problem is the same as

$$\begin{array}{ll} \underset{\tilde{\Phi}\in\mathbb{R}^{\mathbf{4}M\times M},\ \mu\in\mathbb{R}}{\text{find}} & \tilde{\Phi},\ \mu \\ \text{subject to} & \tilde{E}\operatorname{vec}\tilde{\Phi} = \mu\tilde{\tau} \\ & \operatorname{vec}\tilde{\Phi} \succeq 0 \\ & (\operatorname{vec}\tilde{\Phi})_i \geq 1 \end{array} \tag{938}$$

A minimal cardinality solution to Eternity II (937) implicitly constrains $\tilde{\Phi}^{\star}$ to be binary. So this test (938) of membership to $\mathcal{F}(\mathcal{K} \ni \tilde{\tau})$ may be tightened to a test of $(\operatorname{vec}\tilde{\Phi})_i = 1$;

[4.79] Column elimination can be quite dramatic but is dependent upon problem geometry. By method of convex cones, we discard 53,666 more columns via Saunders' `pdco`; a total of 111,506 columns removed from 262,144 leaving all remaining column entries unaltered. Following dependent row removal via `lusol`, dimension of $\tilde{E}$ becomes $7{,}362 \times 150{,}638$; call that A. Any process of discarding rows and columns prior to optimization is *presolving*.

id est, for $i=1\dots 4M^2$ distinct feasibility problems

$$\begin{array}{ll} \underset{\tilde{\Phi}\in\mathbb{R}^{4M\times M}}{\text{find}} & \tilde{\Phi} \\ \text{subject to} & \tilde{E}\,\mathrm{vec}\,\tilde{\Phi}=\tilde{\tau} \\ & \mathrm{vec}\,\tilde{\Phi}\succeq 0 \\ & (\mathrm{vec}\,\tilde{\Phi})_i=1 \end{array} \tag{939}$$

whose feasible set is a proper subset of that in (938). Real variable μ can be set to 1 because if it must not be, then feasible $(\mathrm{vec}\,\tilde{\Phi})_i=1$ could not be feasible to Eternity II (937). If infeasible here in (939), then the only choice remaining for $(\mathrm{vec}\,\tilde{\Phi})_i$ is 0; meaning, column $\tilde{E}(:,i)$ may be discarded after all columns have been tested. This tightened problem (939) therefore tells us two things when feasible: $\tilde{E}(:,i)$ belongs to the smallest face of $\mathcal{K}$ that contains $\tilde{\tau}$, and $(\mathrm{vec}\,\tilde{\Phi})_i$ constitutes a nonzero vertex-coordinate of permutation polyhedron (100). After presolving via this conic pruning method (with subsequent zero-row and dependent row removal), dimension of $\tilde{E}$ goes to $7{,}362\times 150{,}638$.

generators of smallest face are conically independent

Designate $A\in\mathbb{R}^{7362\times 150638}\triangleq\mathbb{R}^{m\times n}$ as matrix $\tilde{E}$ after having discarded all generators not in the smallest face $\mathcal{F}$ of cone $\mathcal{K}$ that contains $\tilde{\tau}$. The Eternity II problem (937) becomes

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{minimize}} & \|x\|_0 \\ \text{subject to} & Ax=b \\ & x\succeq 0 \end{array} \tag{940}$$

To further prune all generators relatively interior to that smallest face, we may subsequently test for conic dependence as described in §2.10 (280): for $i=1\dots 150{,}638$

$$\begin{array}{rl} \text{find} & x \\ \text{subject to} & Ax=A(:,i) \\ & x\succeq 0 \\ & x_i=0 \end{array} \tag{941}$$

where x is $\mathrm{vec}\,\tilde{\Phi}$ corresponding to columns of $\tilde{E}$ not previously discarded by (939).[4.80] If feasible, then column $A(:,i)$ is a conically dependent generator of the smallest face and must be discarded from matrix A before proceeding with test of remaining columns. It turns out, for Eternity II: generators of the smallest face, previously found via (939), comprise a minimal set; *id est*, (941) is never feasible and so no column of A can be discarded (A remains unaltered).[4.81]

affinity for maximization

Designate vector $b\in\mathbb{R}^m$ to be $\tilde{\tau}$ after discarding all entries corresponding to dependent rows in $\tilde{E}$; *id est*, b is $\tilde{\tau}$ subsequent to presolving. Then Eternity II resembles Figure **33**a

[4.80] Discarded entries in $\mathrm{vec}\,\tilde{\Phi}$ are optimally 0.

[4.81] One cannot help but notice a binary selection of variable by tests (939) and (941): Geometrical test (939) (smallest face) checks feasibility of vector entry 1 while geometrical test (941) (conic independence) checks feasibility of 0. Changing 1 to 0 in (939) is always feasible for Eternity II.

(not (b)) because variable x is implicitly bounded above by design; $\mathbf{1} \succeq x$ by confinement of Φ in (928) to the permutation polyhedron (100), for $i = 1 \dots 150{,}638$

$$1 \;=\; \begin{array}{ll} \underset{x}{\text{maximize}} & x_i \\ \text{subject to} & Ax = b \\ & x \succeq 0 \end{array} \tag{942}$$

Unity is always attainable, by (939). By (933) this means (§4.5.1.4)

$$M \;=\; \begin{array}{ll} \underset{y(x)\,,\,x}{\text{maximize}} & (\mathbf{1}-y)^{\mathrm{T}}x \\ \text{subject to} & Ax = b \\ & x \succeq 0 \end{array} \;\equiv\; \begin{array}{ll} \underset{x}{\text{maximize}} & \|x\|_{M}^{n} \\ \text{subject to} & Ax = b \\ & x \succeq 0 \end{array} \tag{943}$$

where

$$y = \mathbf{1} - \nabla\|x\|_{M}^{n} \tag{815}$$

is a direction vector from the technique of convex iteration in §4.5.1.1 and $\|x\|_{M}^{n}$ is a k-largest norm (§3.2.2.1, $k = M$). When upper bound M in (943) is met, solution x will be optimal for Eternity II because it must then be a Boolean vector with minimal cardinality M.

Maximization of convex function $\|x\|_{M}^{n}$ (monotonic on $\mathbb{R}_{+}^{n}$) is not a convex problem, though the constraints are convex. [325, §32] This problem formulation is unusual, nevertheless, insofar as its geometrical visualization is quite clear. We therefore choose to work with a complementary direction vector z, in what follows, in predilection for a mental picture of convex function maximization.

direction vector is optimal solution at global convergence

Instead of solving (943), which is difficult, we propose iterating a convex problem sequence: for $\mathbf{1} - y \leftarrow z$

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{maximize}} & z^{\mathrm{T}}x \\ \text{subject to} & Ax = b \\ & x \succeq 0 \end{array} \tag{944}$$

$$\begin{array}{ll} \underset{z\in\mathbb{R}^n}{\text{maximize}} & z^{\mathrm{T}}x^{\star} \\ \text{subject to} & 0 \preceq z \preceq \mathbf{1} \\ & z^{\mathrm{T}}\mathbf{1} = M \end{array} \tag{525}$$

Variable x is implicitly bounded above at unity by design of A. A globally optimal complementary direction vector $z^{\star}$ will always exactly match an optimal solution $x^{\star}$ for convex iteration of any problem formulated as maximization of a Boolean variable; here we have

$$z^{\star\mathrm{T}}x^{\star} \triangleq M \tag{945}$$

rumination

- Adding a few more clue pieces makes the problem harder in the sense that solution space is diminished; the target gets smaller.
- Because $z^\star = x^\star$, Eternity II can instead be formulated equivalently as a convex-quadratic maximization:

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{maximize}} & x^{\mathrm{T}}x \\ \text{subject to} & Ax = b \\ & x \succeq 0 \end{array} \tag{946}$$

  a nonconvex problem but requiring no convex iteration. If it were possible to form a nullspace basis Z, for A of about equal sparsity,[4.82] such that

$$x = Z\xi + x_{\mathrm{p}} \tag{115}$$

  then the equivalent problem formulation

$$\begin{array}{ll} \underset{\xi}{\text{maximize}} & (Z\xi + x_{\mathrm{p}})^{\mathrm{T}}(Z\xi + x_{\mathrm{p}}) \\ \text{subject to} & Z\xi + x_{\mathrm{p}} \succeq 0 \end{array} \tag{947}$$

  might invoke optimality conditions as obtained in [213, thm.8][4.83]. □

4.7 Constraining rank of indefinite matrices

Example 4.7.0.0.1, which follows, demonstrates that convex iteration is more generally applicable: to indefinite or nonsquare matrices $X \in \mathbb{R}^{m\times n}$; not only to positive semidefinite matrices. Indeed,

$$\begin{array}{ll} \underset{X\in\mathbb{R}^{m\times n}}{\text{find}} & X \\ \text{subject to} & X \in \mathcal{C} \\ & \operatorname{rank} X \le k \end{array} \quad \equiv \quad \begin{array}{ll} \underset{X,\,Y,\,Z}{\text{find}} & X \\ \text{subject to} & X \in \mathcal{C} \\ & G = \begin{bmatrix} Y & X \\ X^{\mathrm{T}} & Z \end{bmatrix} \\ & \operatorname{rank} G \le k \end{array} \tag{948}$$

Proof. $\operatorname{rank} G \le k \Rightarrow \operatorname{rank} X \le k$ because X is the projection of composite matrix G on subspace $\mathbb{R}^{m\times n}$. For symmetric Y and Z, any rank-k positive semidefinite composite matrix G can be factored into rank-k terms R; $G = R^{\mathrm{T}}R$ where $R \triangleq [\,B\ \ C\,]$ and $\operatorname{rank} B, \operatorname{rank} C \le \operatorname{rank} R$ and $B \in \mathbb{R}^{k\times m}$ and $C \in \mathbb{R}^{k\times n}$. Because Y and Z and $X = B^{\mathrm{T}}C$ are variable, (1554) $\operatorname{rank} X \le \operatorname{rank} B, \operatorname{rank} C \le \operatorname{rank} R = \operatorname{rank} G$ is tight. ♦

[4.82] Define *sparsity* as ratio of number of nonzero entries to matrix-dimension product. *For matrices, the average number of nonzeros per row or column is easier to understand and likely to be small for typical LP problems, independent of the dimensions.* —Michael Saunders

[4.83] *... the assumptions in Theorem 8 ask for the Q_i being positive definite* (*see the top of the page of Theorem 8*). *I must confess that I do not remember why.* —Jean-Baptiste Hiriart-Urruty

So there must exist an optimal direction vector $W^\star$ such that

$$\begin{array}{ll} \underset{X,Y,Z}{\text{find}} & X \\ \text{subject to} & X \in \mathcal{C} \\ & G = \begin{bmatrix} Y & X \\ X^{\mathrm{T}} & Z \end{bmatrix} \\ & \operatorname{rank} G \leq k \end{array} \quad \equiv \quad \begin{array}{ll} \underset{X,Y,Z}{\text{minimize}} & \langle G\,, W^\star \rangle \\ \text{subject to} & X \in \mathcal{C} \\ & G = \begin{bmatrix} Y & X \\ X^{\mathrm{T}} & Z \end{bmatrix} \succeq 0 \end{array} \tag{949}$$

Were $W^\star = I$, by (1785) the optimal resulting trace objective would be equivalent to the minimization of nuclear norm of X over $\mathcal{C}$. This means:

- (*confer* p.194) The argument of any nuclear norm minimization problem may be replaced with a composite semidefinite variable of the same optimal rank but doubly dimensioned.

Then Figure **90** becomes an accurate geometrical description of a consequent composite semidefinite problem objective. But there are better direction vectors than Identity I which occurs only under special conditions:

4.7.0.0.1 Example. *Compressed sensing, compressive sampling.* [318]
As our modern technology-driven civilization acquires and exploits ever-increasing amounts of data, everyone now knows that most of the data we acquire can be thrown away with almost no perceptual loss − witness the broad success of lossy compression formats for sounds, images, and specialized technical data. The phenomenon of ubiquitous compressibility raises very natural questions: Why go to so much effort to acquire all the data when most of what we get will be thrown away? Can't we just directly measure the part that won't end up being thrown away? −David Donoho [129]

Lossy data compression techniques like JPEG are popular, but it is also well known that compression artifacts become quite perceptible with signal postprocessing that goes beyond mere playback of a compressed signal. [236] [261] Spatial or audio frequencies presumed masked by a simultaneity are not encoded, for example, so rendered imperceptible even with significant postfiltering (of the compressed signal) that is meant to reveal them; *id est*, desirable artifacts are forever lost, so highly compressed data is not amenable to analysis and postprocessing: *e.g*, sound effects [102] [103] [105] or image enhancement (Adobe *Photoshop*).[4.84] Further, there can be no universally acceptable unique metric of perception for gauging exactly how much data can be tossed. For these reasons, there will always be need for raw (noncompressed) data.

In this example we throw out only so much information as to leave perfect reconstruction within reach. Specifically, the MIT logo in Figure **125** is perfectly reconstructed from 700 time-sequential samples $\{y_i\}$ acquired by the one-pixel camera illustrated in Figure **126**. The MIT-logo image in this example impinges a 46×81

[4.84] As simple a process as upward scaling of signal amplitude or image size will always introduce noise; even to a noncompressed signal. But scaling-noise is particularly noticeable in a JPEG-compressed image; *e.g*, text or any sharp edge.

Figure 125: Massachusetts Institute of Technology (MIT) logo, including its white boundary, may be interpreted as a rank-5 matrix. (Stanford University logo rank is much higher;) This constitutes *Scene* Y observed by the one-pixel camera in Figure **126** for Example 4.7.0.0.1.

array micromirror device. This mirror array is modulated by a pseudonoise source that independently positions all the individual mirrors. A single photodiode (one pixel) integrates incident light from all mirrors. After stabilizing the mirrors to a fixed but pseudorandom pattern, light so collected is then digitized into one sample y_i by analog-to-digital (A/D) conversion. This sampling process is repeated with the micromirror array modulated to a new pseudorandom pattern.

The most important questions are: How many samples do we need for perfect reconstruction? Does that number of samples represent compression of the original data?

We claim that perfect reconstruction of the MIT logo can be reliably achieved with as few as 700 samples $y = [y_i] \in \mathbb{R}^{\mathbf{700}}$ from this one-pixel camera. That number represents only 19% of information obtainable from 3726 micromirrors.[4.85] (Figure **127**)

Our approach to reconstruction is to look for low-rank solution to an *underdetermined* system:

$$\begin{array}{ll} \underset{X \in \mathbb{R}^{\mathbf{46}\times\mathbf{81}}}{\text{find}} & X \\ \text{subject to} & A \operatorname{vec} X = y \\ & \operatorname{rank} X \leq 5 \end{array} \tag{950}$$

where $\operatorname{vec} X$ is the vectorized (37) unknown image matrix. Each row of fat matrix A is one realization of a pseudorandom pattern applied to the micromirrors. Since these patterns are deterministic (known), then the i^{th} sample y_i equals $A(i,:) \operatorname{vec} Y$; *id est*, $y = A \operatorname{vec} Y$. *Perfect reconstruction* here means optimal solution $X^\star$ equals scene $Y \in \mathbb{R}^{\mathbf{46}\times\mathbf{81}}$ to within machine precision.

Because variable matrix X is generally not square or positive semidefinite, we constrain

[4.85] That number (700 samples) is difficult to achieve, as reported in [318, §6]. If a minimal basis for the MIT logo were instead constructed, only five rows or columns worth of data (from a 46×81 matrix) are linearly independent. This means a lower bound on achievable compression is about $5 \times 46 = 230$ samples plus 81 samples column encoding; which corresponds to 8% of the original information. (Figure **127**)

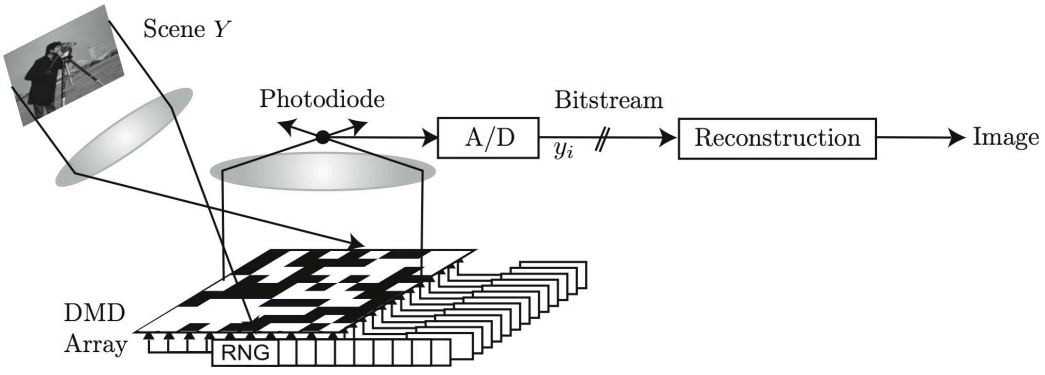


Figure 126: One-pixel camera. *Compressive imaging camera block diagram. Incident lightfield (corresponding to the desired image Y) is reflected off a digital micromirror device* (DMD) *array whose mirror orientations are modulated in the pseudorandom pattern supplied by the random number generators* (RNG). *Each different mirror pattern produces a voltage at the single photodiode that corresponds to one measurement y_i.* −[362] [409]

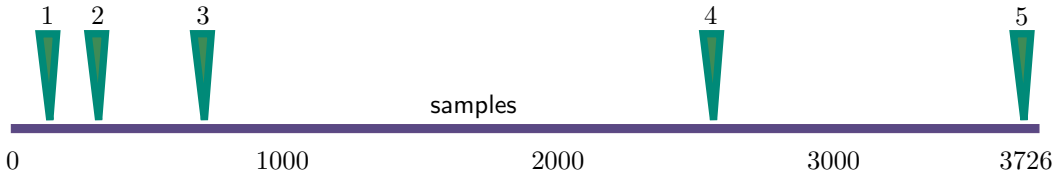


Figure 127: Estimates of compression for various encoding methods:
1) linear interpolation (140 samples),
2) minimal columnar basis (311 samples),
3) convex iteration (700 samples) can achieve lower bound predicted by compressed sensing (670 samples, $n=46\times 81$, $k=140$, Figure **107**) whereas nuclear norm minimization alone does not [318, §6],
4) JPEG @100% *quality* (2588 samples),
5) no compression (3726 samples).

its rank by rewriting the problem equivalently

$$\begin{array}{ll} \underset{W_1\in\mathbb{R}^{46\times 46},\, W_2\in\mathbb{R}^{81\times 81},\, X\in\mathbb{R}^{46\times 81}}{\text{find}} & X \\ \text{subject to} & A\,\mathrm{vec}\,X = y \\ & \operatorname{rank}\begin{bmatrix} W_1 & X \\ X^{\mathrm{T}} & W_2 \end{bmatrix} \le 5 \end{array} \tag{951}$$

This rank constraint on the composite (block) matrix insures $\operatorname{rank} X \le 5$ for any choice of dimensionally compatible matrices W_1 and W_2. But to solve this problem by convex iteration, we alternate solution of semidefinite program

$$\begin{array}{ll} \underset{W_1\in\mathbb{S}^{46},\, W_2\in\mathbb{S}^{81},\, X\in\mathbb{R}^{46\times 81}}{\text{minimize}} & \operatorname{tr}\left(\begin{bmatrix} W_1 & X \\ X^{\mathrm{T}} & W_2 \end{bmatrix} Z\right) \\ \text{subject to} & A\,\mathrm{vec}\,X = y \\ & \begin{bmatrix} W_1 & X \\ X^{\mathrm{T}} & W_2 \end{bmatrix} \succeq 0 \end{array} \tag{952}$$

with semidefinite program

$$\begin{array}{ll} \underset{Z\in\mathbb{S}^{46+81}}{\text{minimize}} & \operatorname{tr}\left(\begin{bmatrix} W_1 & X \\ X^{\mathrm{T}} & W_2 \end{bmatrix}^{\star} Z\right) \\ \text{subject to} & 0 \preceq Z \preceq I \\ & \operatorname{tr} Z = 46 + 81 - 5 \end{array} \tag{953}$$

(which has optimal solution known in closed form, p.567) until a rank-5 composite matrix is found.

With 1000 samples $\{y_i\}$, convergence occurs in two iterations; 700 samples require more than ten iterations but reconstruction remains perfect. Iterating more admits taking of fewer samples. Reconstruction is independent of pseudorandom sequence parameters; *e.g*, binary sequences succeed with the same efficiency as Gaussian or uniformly distributed sequences. □

4.7.1 rank-constraint midsummary

We find that this *direction matrix* idea works well and quite independently of desired rank or affine dimension. This idea of direction matrix is good principally because of its simplicity: When confronted with a problem otherwise convex if not for a rank or cardinality constraint, then that constraint becomes a linear regularization term in the objective.

There exists a common thread through all these Examples; that being, convex iteration with a direction matrix as normal to a linear regularization (a generalization of the well-known trace heuristic). But each problem type (per Example) possesses its own idiosyncrasies that slightly modify how a rank-constrained optimal solution is actually obtained: The *ball packing* problem in Chapter 5.4.2.2.6, for example, requires a problem sequence in a progressively larger number of balls to find a good initial value for the direction matrix, whereas many of the examples in the present chapter require an initial value of $\mathbf{0}$. Finding a Boolean solution in Example 4.6.0.0.9 requires a procedure to detect

stalls, while other problems have no such requirement. The combinatorial Procrustes problem in Example 4.6.0.0.3 allows use of a known closed-form solution for direction vector when solved via rank constraint, but not when solved via cardinality constraint. Some problems require a careful weighting of the regularization term, whereas other problems do not, and so on. It would be nice if there were a universally applicable method for constraining rank; one that is less susceptible to quirks of a particular problem type.

Poor initialization of the direction matrix from the regularization can lead to an erroneous result. We speculate one reason to be a simple dearth of optimal solutions of desired rank or cardinality;[4.86] an unfortunate choice of initial search direction leading astray. Ease of solution by convex iteration occurs when optimal solutions abound. With this speculation in mind, we now propose a further generalization of convex iteration for constraining rank that attempts to ameliorate quirks and unify problem types:

4.8 Convex Iteration rank-1

We now develop a general method for constraining rank that first decomposes a given problem via standard diagonalization of matrices (§A.5). This method is motivated by observation (§4.4.1.1) that an optimal direction matrix can be simultaneously diagonalizable with an optimal variable matrix. This suggests minimization of an objective function directly in terms of eigenvalues. A second motivating observation is that variable orthogonal matrices seem easily found by convex iteration; *e.g*, Procrustes Example 4.6.0.0.2.

4.8.1 rank-1 transformation

It turns out that this general method always requires solution to a rank-1 constrained problem regardless of desired rank ρ from the original problem. To demonstrate, we pose a semidefinite feasibility problem

$$\begin{array}{ll} \text{find} & X \in \mathbb{S}^n \\ \text{subject to} & A\,\operatorname{svec} X = b \\ & X \succeq 0 \\ & \operatorname{rank} X \leq \rho \end{array} \tag{954}$$

given an upper bound $0<\rho<n$ on rank, a vector $b\in\mathbb{R}^m$, and typically fat full-rank

$$A = \begin{bmatrix} \operatorname{svec}(A_1)^{\mathrm{T}} \\ \vdots \\ \operatorname{svec}(A_m)^{\mathrm{T}} \end{bmatrix} \in \mathbb{R}^{m\times n(n+1)/2} \tag{688}$$

where $A_i\in\mathbb{S}^n$, $i=1\ldots m$. So, for symmetric matrix vectorization svec as defined in (56),

$$A\,\operatorname{svec} X = \begin{bmatrix} \operatorname{tr}(A_1\,X) \\ \vdots \\ \operatorname{tr}(A_m\,X) \end{bmatrix} \tag{689}$$

[4.86] In Convex Optimization, an optimal solution generally comes from a convex set of optimal solutions; (§3.1.2.1) that set can be large.

This program (954) is a statement of the classical problem of finding a matrix X of maximum rank ρ in the intersection of the positive semidefinite cone with a given number m of hyperplanes in the subspace of symmetric matrices $\mathbb{S}^n$. [27, §II.13] [25, §2.2] Such a matrix is presumed to exist.

To begin transformation of (954), express the nonincreasingly ordered diagonalization (§A.5.1) of positive semidefinite variable matrix

$$X \triangleq Q\Lambda Q^{\mathrm{T}} = \sum_{i=1}^{n} \lambda_i Q_{ii} \in \mathbb{S}^n \tag{955}$$

which is a sum of rank-1 orthogonal projection matrices Q_{ii} weighted by eigenvalues λ_i where $Q_{ij} \triangleq q_i q_j^{\mathrm{T}} \in \mathbb{R}^{n\times n}$, $Q = [\,q_1 \cdots q_n\,] \in \mathbb{R}^{n\times n}$, $Q^{\mathrm{T}} = Q^{-1}$, $\Lambda_{ii} = \lambda_i \in \mathbb{R}$, and

$$\Lambda = \begin{bmatrix} \lambda_1 & & & \mathbf{0} \\ & \lambda_2 & & \\ & & \ddots & \\ \mathbf{0}^{\mathrm{T}} & & & \lambda_n \end{bmatrix} \in \mathbb{S}^n \tag{956}$$

where $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n \geq 0$. Recall the fact:

$$\Lambda \succeq 0 \;\Leftrightarrow\; X \succeq 0 \qquad (1539)$$

From orthogonal matrix Q in ordered diagonalization (955) of variable X, take a matrix

$$U \triangleq [\,u_1 \cdots u_\rho\,] \triangleq Q(:,1{:}\rho)\sqrt{\Lambda(1{:}\rho,1{:}\rho)} = \left[\sqrt{\lambda_1}\,q_1 \;\cdots\; \sqrt{\lambda_\rho}\,q_\rho\right] \in \mathbb{R}^{n\times\rho} \tag{957}$$

Then U has orthogonal but unnormalized columns;

$$X = UU^{\mathrm{T}} = \sum_{i=1}^{\rho} u_i u_i^{\mathrm{T}} \triangleq \sum_{i=1}^{\rho} U_{ii} = \sum_{i=1}^{\rho} \lambda_i q_i q_i^{\mathrm{T}} \in \mathbb{S}^n \tag{958}$$

Make an assignment

$$\begin{aligned} Z &= \begin{bmatrix} u_1 \\ \vdots \\ u_\rho \end{bmatrix} [\,u_1^{\mathrm{T}} \cdots u_\rho^{\mathrm{T}}\,] \qquad \in \mathbb{S}^{n\rho} \\ &= \begin{bmatrix} U_{11} & \cdots & U_{1\rho} \\ \vdots & \ddots & \vdots \\ U_{1\rho}^{\mathrm{T}} & \cdots & U_{\rho\rho} \end{bmatrix} \triangleq \begin{bmatrix} u_1 u_1^{\mathrm{T}} & \cdots & u_1 u_\rho^{\mathrm{T}} \\ \vdots & \ddots & \vdots \\ u_\rho u_1^{\mathrm{T}} & \cdots & u_\rho u_\rho^{\mathrm{T}} \end{bmatrix} \end{aligned} \tag{959}$$

Then transformation of (954) to its rank-1 equivalent is:

$$\begin{array}{ll} \underset{U_{ii}\in\mathbb{S}^n,\ U_{ij}\in\mathbb{R}^{n\times n}}{\text{find}} & X=\sum_{i=1}^{\rho} U_{ii} \\ \text{subject to} & Z=\begin{bmatrix} U_{11} & \cdots & U_{1\rho} \\ \vdots & \ddots & \vdots \\ U_{1\rho}^{\mathrm{T}} & \cdots & U_{\rho\rho} \end{bmatrix} (\succeq 0) \\ & A\,\operatorname{svec}\sum_{i=1}^{\rho} U_{ii}=b \\ & \operatorname{tr} U_{ij}=0 \qquad\qquad i<j=2\dots\rho \\ & \operatorname{rank} Z=1 \end{array} \tag{960}$$

Symmetry is necessary and sufficient for positive semidefiniteness of a rank-1 matrix. (§A.3.1.0.7) Matrix X is positive semidefinite whenever Z is. (§A.3.1.0.4, §A.3.1.0.2) This new problem always enforces a rank-1 constraint on matrix Z; *id est*, regardless of upper bound on rank ρ of variable matrix X, this equivalent problem always poses a rank-1 constraint. We propose solving (960) by iteration of convex problem

$$\begin{array}{ll} \underset{U_{ii}\in\mathbb{S}^n,\ U_{ij}\in\mathbb{R}^{n\times n}}{\text{minimize}} & \operatorname{tr}(Z\,W) \\ \text{subject to} & Z=\begin{bmatrix} U_{11} & \cdots & U_{1\rho} \\ \vdots & \ddots & \vdots \\ U_{1\rho}^{\mathrm{T}} & \cdots & U_{\rho\rho} \end{bmatrix} \succeq 0 \\ & A\,\operatorname{svec}\sum_{i=1}^{\rho} U_{ii}=b \\ & \operatorname{tr} U_{ij}=0 \qquad\qquad i<j=2\dots\rho \end{array} \tag{961}$$

with convex problem

$$\begin{array}{ll} \underset{W\in\mathbb{S}^{n\rho}}{\text{minimize}} & \operatorname{tr}(Z^{\star}\,W) \\ \text{subject to} & 0\preceq W\preceq I \\ & \operatorname{tr} W=n\rho-1 \end{array} \tag{962}$$

the latter providing direction of search W for a rank-1 matrix Z in (961). These convex problems (961) (962) are iterated until a rank-1 Z matrix is found (until the objective of (961) vanishes). Initial value of direction matrix W is the Identity. For subsequent iterations, an optimal solution to (962) has closed form (p.567).

Because of the nonconvex nature of a rank-constrained problem, there can be no proof of global convergence of this convex iteration. But this iteration always converges to a local minimum because the sequence of objective values is monotonic and nonincreasing; any monotonically nonincreasing real sequence converges. [274, §1.2] [43, §1.1] A rank ρ matrix X solving the original problem (954) is found when the objective in (961) converges to 0; a certificate of global optimality for the convex iteration. In practice, incidence of global convergence is quite high (99.99% [392]); failures being mostly attributable to numerical

accuracy. Upper bound ρ on rank of positive semidefinite matrix X is assured by rank-1 optimal matrix Z.

4.8.1.0.1 Example. *Singular value decomposition by convex iteration.* [173]
This diagonal decomposition technique (transformation to a rank-1 problem) is extensible to other problem types; *e.g*, [241, §III]. Rank-1 transformation makes singular value decomposition (SVD, §A.6) possible by convex iteration because orthogonality constraints may then be introduced. We learn that any uniqueness properties, the SVD of rank-ρ matrix $X \triangleq USV^{\mathrm{T}} \in \mathbb{R}^{m\times n}$ might enjoy, stem from demand for singular vector orthonormality.[4.87]

Assignment $Z \in \mathbb{S}_+^{2m\rho+n\rho+\rho+1}$ is key to finding the SVD of X by convex optimization:

$$
\begin{array}{ll}
\underset{H\,,\,J}{\text{find}} & U\,,\,\delta(S)\,,\,V \\
\text{subject to} & Z = \begin{bmatrix} 1 & \operatorname{vec}(H)^{\mathrm{T}} & \operatorname{vec}(U)^{\mathrm{T}} & \delta(S)^{\mathrm{T}} & \operatorname{vec}(V)^{\mathrm{T}} \\ \operatorname{vec} H & & & & \\ \operatorname{vec} U & & J & & \\ \delta(S) & & & & \\ \operatorname{vec} V & & & & \end{bmatrix} \succeq 0 \\
& \delta(S) \succeq 0 \\
& H = US \\
& X = HV^{\mathrm{T}} \\
& HU^{\mathrm{T}} \text{ symmetry} \\
& U^{\mathrm{T}}H \text{ perpendicularity} \\
& \operatorname{tr}\big(H(:,i)\,H(:,i)^{\mathrm{T}}\big) = S(i\,,i)^2 \qquad i=1\ldots\rho \\
& \operatorname{tr}\big(H(:,i)\,U(:,i)^{\mathrm{T}}\big) = S(i\,,i) \qquad i=1\ldots\rho \\
& H \text{ orthogonality} \\
& U\,,\,V \text{ orthonormality} \\
& \operatorname{rank} Z = 1
\end{array}
\tag{963}
$$

where variable matrix $J \in \mathbb{S}_+^{2m\rho+n\rho+\rho}$ is a large partition of Z, where given rank-ρ matrix $X \in \mathbb{R}^{m\times n}$ is subject to SVD in unknown orthonormal matrices $U \in \mathbb{R}^{m\times\rho}$ and $V \in \mathbb{R}^{n\times\rho}$ and unknown diagonal matrix of singular values $S \in \mathbb{R}^{\rho\times\rho}$, and where introduction of variable $H \triangleq US \in \mathbb{R}^{m\times\rho}$ makes identification of input $X = HV^{\mathrm{T}}$ possible within partition J. Orthogonality constraints on columns of H, within J, and orthonormality constraints on columns of U and V are critical; *videlicet*, $h \perp v \Leftrightarrow \operatorname{tr}(hv^{\mathrm{T}}) = 0$; $v^{\mathrm{T}}v = 1 \Leftrightarrow \operatorname{tr}(vv^{\mathrm{T}}) = 1$.

Symmetric matrix Z is positive semidefinite rank-1 at optimality, regardless of ρ. That rank constraint is the only nonconvex constraint in (963); the only constraint that cannot be directly implemented in a convex manner per partition J. But the rank constraint is handled well by convex iteration. Matlab implementation of SVD by convex iteration is intricate although incidence of global convergence is 99.99% [406], barring numerical error.

□

[4.87] Otherwise, there exist many similarly structured tripartite nonorthogonal matrix decompositions; in place of ρ nonzero singular values, diagonal matrix S would instead hold exactly ρ *coordinates*; orthonormal columns in U and V would become merely linearly independent.

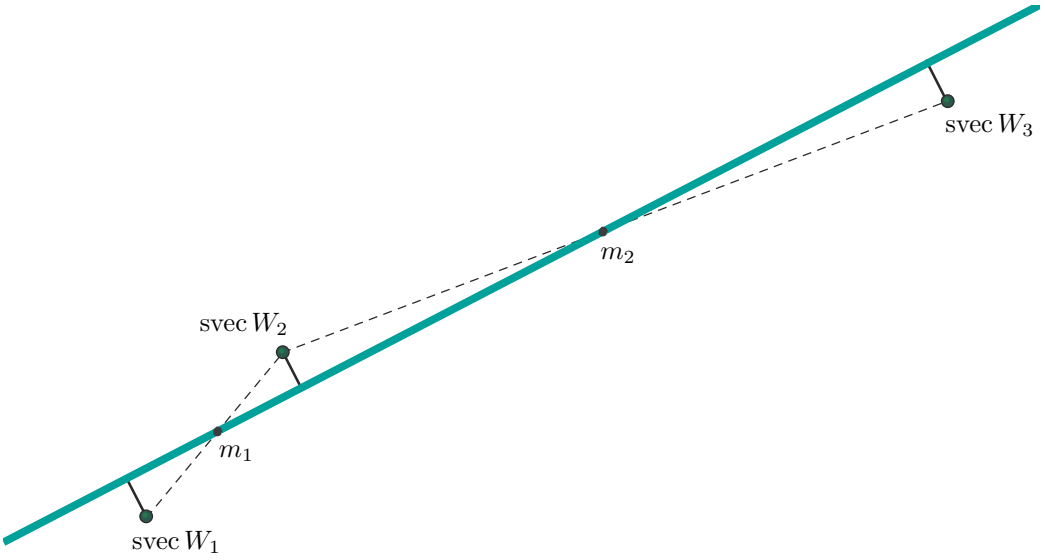


Figure 128: W_1, W_2, and W_3 represent the last three direction vectors in a sequence. m_1 represents the midpoint between direction vectors W_1 and W_2; m_2 is the midpoint of W_2 and W_3. Straight line passes through midpoints.

4.8.2 convex iteration accelerant

Convex iteration can be made to converge faster; sometimes, by orders of magnitude. The idea here is to determine whether the last three direction vectors are close to their fit to a straight line. When three direction vectors are close to a straight line, then the last direction vector may be replaced with its extrapolation along that line.

To reduce computation time, the fitted line is not a best fit. Instead, the midpoint between each pair of iteration-adjacent direction vectors is calculated (Figure **128**). A straight line is uniquely defined by two midpoints in any dimension. The distance of each direction vector to the line is calculated, then those three distances summed. When a sum is small, three direction vectors are deemed close to the line determined by them.

What is meant above by *close* and *small* depends on the particular problem type at hand. For the parameters and normalized random data chosen for two Matlab realizations [392] [406] on *Wikimization* (corresponding to problems (960) and (963)), *small* is numerically defined to be 1 or less in the statement `if straight < 1` whose purpose is to determine straightness of the last three direction vectors of convex iteration. The smaller the value of the normalized sum called `straight`, the closer the last three direction vectors are to a straight line. Variable `straight` is bounded below by 0 which indicates three direction vectors precisely on the line going through them.

If linear extrapolation goes too far, then the objective of convex iteration will increase or a solver may fail numerically. In either case, one must forget the last iteration and back up the linear extrapolation until the objective decreases. These techniques are illustrated by the Matlab programs.

4.8.2.0.1 Exercise. *SVD by convex iteration.*
Write every constraint in (963), beginning with $H = US$ downward excepting the rank constraint, as an affine expression of variable matrix J .[4.88] ▼

[4.88] Hint: [406].