Recognizing Underlying Sparsity in Optimization

Sunyoung Kim[1] Masakazu Kojima[2] Philippe Toint[3]

[1]Ewha Woman's University, Seoul, Korea

[2]Tokyo Institute of Technology, Japan

[3]University of Namur, Belgium

Sparse Days, Cerfacs, June 2006

Outline

1. Introduction
2. A method for recognizing underlying sparsity
3. Conclusions and perspectives

Structure and efficiency

In scientific computations:

Problem structure	$\Longrightarrow$	Sparse linearization	$\Longrightarrow$	Efficient computation

Example:

local variables + local interaction $\longrightarrow$ sparsity pattern $\longrightarrow$ efficient factorizations

This talk's objective: explore the $\Longrightarrow$ implication in the context of optimization

Sparsity and optimization

Where is sparsity useful in nonlinear optimization?

- unconstrained: Newton's method:

$$H_k \Delta x_k = -\nabla_x f(x_k)$$

  with $H_k \approx \nabla_{xx} f(x_k)$;
- constrained: KKT system

$$\begin{pmatrix} H_k & A_k^T \\ A_k & 0 \end{pmatrix} \begin{pmatrix} \Delta x_k \\ \Delta \lambda_k \end{pmatrix} = - \begin{pmatrix} g(x_k) \\ 0 \end{pmatrix}$$

  with $H_k \approx \nabla_{xx} L(x_k, \lambda_k)$.
- our motivation: (sparse) semi-definite relaxations for polynomial problems

Partially invariance. . .

A common structure: partial invariance

$f(x)$ is partially invariant

$\Longleftrightarrow$

$f(x) = \bar{f}(u)$ with $u = Ax$ and A has low rank

$f(x)$ is partially separable

$\Longleftrightarrow$

$f(x) = \sum_{\ell=1}^{m} f_\ell(x)$ where each $f_\ell(x)$ is partially invariant

... and useful consequences

If $f(x)$ is partially invariant:

- range(A) is a subspace $\Rightarrow$ geometric concept (basis invariant)
- Hessian structure

$$\nabla_{xx} f(x) = A^T \nabla_{uu} \bar{f}(u) A$$

- invariant subspace:

$$\mathrm{Inv}(f) = \{w \in \mathbf{R}^n \mid f(x+w) = f(x) \quad \forall x \in \mathbf{R}^n\} = \mathrm{Null}(A)$$

- induced (Cartesian) sparsity:

$$e_\ell \in \mathrm{Inv}(f) \Longrightarrow [\nabla_{xx} f(x)]_{ij} = 0 \text{ for } i = \ell \text{ or } j = \ell$$

(in this case, $A = \boxed{}$)

Define

$$K(f) = \{\ell \mid e_\ell \in \mathrm{Inv}(f)\}$$

the sparsity index of f

The question

Griewank and T. (1981):

$f(x)$ smooth and $\nabla_{xx} f(x)$ sparse $\Longrightarrow f(x)$ partially separable

The question is to recognize the underlying sparsity:

Given $\{f_\ell(x)\}_{\ell \in M}$ a collection of partially invariant functions, is there a basis in which each $\nabla_{xx} f_\ell(x)$ has few nonzero rows and columns?

More specifically (for sparse SDP relaxations):

Can we choose a basis such that $\nabla_{xx} \left[\sum_{\ell \in M} f_\ell(x)\right]$ admits a sparse Cholesky factorization?

Sparsity and the basis

The tool of the trade

$z \longrightarrow x = Pz$ where $P = (p_1, \ldots, p_n)$ is nonsingular

We then consider the transformed functions

$$g_\ell(z) = f_\ell(Pz)$$

and the invariant spaces are preserved:

$$\mathrm{Inv}(g_\ell) = P^{-1}\mathrm{Inv}(f_\ell)$$

and

$$K(g_\ell) = \{j \mid e_j \in \mathrm{Inv}(g_\ell)\} = \{j \mid P^{-1}p_j \in \mathrm{Inv}(g_\ell)\} = \{j \mid p_j \in \mathrm{Inv}(f_\ell)\}$$

Thus,

sparsity can be increased by choosing p_j in (as many as possible) invariant subspaces

An example

Consider

$$\min_x \sum_{\ell=1}^{n} \left[x_\ell^2 - x_\ell\right] + \left[\sum_{i=1}^{n} x_i\right]^4$$

then

$$\mathrm{Inv}(f_j) = e_j^\perp \quad (\ell = 1, \dots, n) \text{ and } \mathrm{Inv}(f_{n+1}) = e^\perp$$

$$K(f_\ell) = \{1, \dots, n\} \setminus \{\ell\} \quad (\ell = 1, \dots, n) \text{ and } K(f_{n+1}) = \emptyset$$

Now choose

$$p_j = e_j - e_{j+1} \quad (j = 1, \dots, n-1) \text{ and } p_n = e_n$$

and the problem becomes

$$\min_z \sum_{\ell=1}^{n} \left[(z_\ell - z_{\ell-1})^2 - (z_\ell - z_{\ell-1})\right] + [z_n]^4$$

Then

$$K(g_\ell) = \{1, \dots, n\} \setminus \{\ell - 1, \ell\} \quad (\ell = 1, \dots, n) \text{ and } K(g_{n+1}) = \{1, \dots, n-1\}$$

the size of $K(g_\ell)$ are large evenly

The idea (1)

For $S \subseteq \{1, \ldots, m\}$ and let

$$\mathrm{Inv}[S] = \bigcap_{\ell \in S} \mathrm{Inv}(f_\ell)$$

Our objective: choose $p_j \in \mathrm{Inv}[S_j]$ for S_j as large as possible ($j = 1, \ldots, n$).

Let

$L_\ell(\mathcal{S}) = \{j \mid \ell \in S_j\}$ (the set of p_j that are invariant for f_ℓ)

Reformulate again:

Can we choose $p_1, \ldots, p_n$ such that the size of the $\{L_\ell(\mathcal{S})\}_{\ell=1}^m$ are large evenly?

The idea (2)

Finally (!), for $\mathcal{S} = (S_1, \ldots, S_n)$, define

$$\sigma(\mathcal{S}) = \text{ the vector } (\#L_1(\mathcal{S}), \ldots, \#L_m(\mathcal{S})) \text{ sorted by increasing values}$$

General idea:

> lexicographically maximize $\sigma(\mathcal{S})$
>
> subject to the existence of $p_1, \ldots, p_n$ with $S_j = \{\ell \mid p_j \in \mathrm{Inv}(f_\ell)\}$

- maximization makes the $L_\ell(\mathcal{S})$ large
- the lexicographic maximization make them *evenly* large

A sketch of the method

How do we solve that combinatorial problem?

- approximate solution only!
- use a greedy approach:
  - progressively increase the size of the problem (external loop)
  - progressively increase the size of the S_j (internal loop)
- ensure (almost certain) feasibility in two steps:
  - “weak” feasibility by a (cheap) probabilistic test
  - real feasibility by the structure of the greedy approach

$\Longrightarrow$ Complicated, but numerically tractable!

POPs over the unit simplex

Does it work?

Example 1: SDP relaxations of simple POPs over the unit simplex

Problem	$n = 4$ nnzL	$n = 4$ cpu	$n = 12$ nnzL	$n = 12$ cpu	$n = 200$ nnzL	$n = 200$ cpu
Rosenbrock	10 / 9	0.3 / 0.2	78 / 43	111.9 / 1.4	21100 / 606	∞ / 21.7
Broyden 3D	10 / 9	0.3 / 0.2	78 / 45	152.1 / 8.6	21100 / 819	∞ / 111.1
Woods	10 / 10	0.2 / 0.3	78 / 50	233.6 / 3.2	21100 / 936	∞ / 104.1

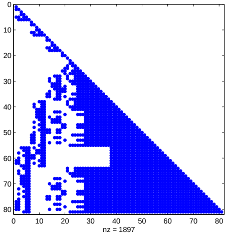
(before transformation/after transformation)

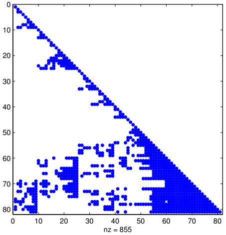
Concave quadratics

Example 2: SDP relaxations of concave quadratics with transportation constraints

unknown: a $m \times k$ matrix

$m = 5, k = 10$ nnzL	cpu	$m = 5, k = 20$ nnzL	cpu	$m = 9, k = 9$ nnzL	cpu	$m = 9, k = 9$ nnzL	cpu
228 / 136	4.7 / 2.0	380 / 757	∞ / 160.7	687 / 388	832.6 / 21.6	1897 / 855	∞ / 917.9





Indefinite quadratics

Example 2: SDP relaxations of rank 4 indefinite quadratics in the unit cube

$n = 10$		$n = 30$		$n = 100$	
nnzL	cpu	nnzL	cpu	nnzL	cpu
55 / 46	1.0 / 0.9	465 / 194	3261.9 / 7.3	5050 / 539	∞ / 55.3

Conclusions

- transformation useful, allowing the solution of previously unsolvable problems
- applications to other areas than SDP relaxation...
- many remaining questions:
  - basis conditioning
  - other measures of sparsity
  - alternative algorithms
  - (efficiency of sparse SDP relaxations)

...but this is a first encouraging step!

Thank you for your attention