Boundary intervals and visualization of AE-solution sets for interval systems of linear equations

Irene A. Sharaya

Institute of Computational Technologies SB RAS
Novosibirsk, Russia

SCAN'2012, Novosibirsk, September 23–29, 2012

I would like to present:

- **new program** for visualization of AE-solution sets,
- **boundary intervals method** as a base of this program.

Introduction

Definition of AE-solution set

Let us be given

$$\boldsymbol{A},\ \boldsymbol{A}^{\forall}, \boldsymbol{A}^{\exists} \in \mathbb{IR}^{m\times n}, \qquad \boldsymbol{b},\ \boldsymbol{b}^{\forall}, \boldsymbol{b}^{\exists} \in \mathbb{IR}^{m},$$

$$\boldsymbol{A} = \boldsymbol{A}^{\forall} + \boldsymbol{A}^{\exists}, \qquad \boldsymbol{b} = \boldsymbol{b}^{\forall} + \boldsymbol{b}^{\exists},$$

$$\forall(i,j)\ \ \boldsymbol{A}^{\forall}_{ij} \cdot \boldsymbol{A}^{\exists}_{ij} = 0, \qquad (\forall i)\ \ \boldsymbol{b}^{\forall}_{i} \cdot \boldsymbol{b}^{\exists}_{i} = 0.$$

We will refer to the set

$$\Xi_{AE} = \{x \in \mathbb{R}^n \mid (\forall A' \in \boldsymbol{A}^{\forall})(\forall b' \in \boldsymbol{b}^{\forall})(\exists A'' \in \boldsymbol{A}^{\exists})(\exists b'' \in \boldsymbol{b}^{\exists})$$
$$(A' + A'')x = b' + b''\}$$

as AE-solution set for the interval linear system $\boldsymbol{A}x = \boldsymbol{b}$.

Definitions and theory of AE-solution sets for interval systems of linear equations was proposed by Sergey P. Shary.

(See e.g.
S.P. Shary, A new technique in systems analysis under interval uncertainty and ambiguity,
Reliable Computing, 8 (2002), No. 5, pp. 321–419,
http://www.nsc.ru/interval/shary/Papers/ANewTech.pdf)

Particular cases of AE-solution sets

The united solution set

$$\Xi_{uni} = \{x \in \mathbb{R}^n \mid (\exists A \in \boldsymbol{A})(\exists b \in \boldsymbol{b})\ (Ax = b)\},$$

the tolerable solution set

$$\Xi_{tol} = \{x \in \mathbb{R}^n \mid (\forall A \in \boldsymbol{A})(\exists b \in \boldsymbol{b})\ (Ax = b)\},$$

and the controllable solution set

$$\Xi_{ctl} = \{x \in \mathbb{R}^n \mid (\forall b \in \boldsymbol{b})(\exists A \in \boldsymbol{A})\ (Ax = b)\}$$

are particular cases of the AE-solution sets.

Geometric properties of AE-solution set

The intersection of an AE-solution set with a closed orthant is a convex polyhedron determined by system of linear inequalities

$$\begin{cases} -A'x \leqslant -\underline{b}^{\exists} - \overline{b}^{\forall}, \\ A''x \leqslant \overline{b}^{\exists} + \underline{b}^{\forall}, \end{cases}$$

where

$$A'_{ij} = \begin{cases} \left(\overline{A}^{\forall} + \underline{A}^{\exists}\right)_{ij} & \text{for } x_j < 0, \\ \left(\underline{A}^{\forall} + \overline{A}^{\exists}\right)_{ij} & \text{otherwise,} \end{cases} \qquad A''_{ij} = \begin{cases} \left(\underline{A}^{\forall} + \overline{A}^{\exists}\right)_{ij} & \text{for } x_j < 0, \\ \left(\overline{A}^{\forall} + \underline{A}^{\exists}\right)_{ij} & \text{otherwise.} \end{cases}$$

The whole AE-solution set is a polyhedral set.

It may be nonconvex, nonconnect, unbounded.

Problem

Given $\boldsymbol{A}^{\forall}$, $\boldsymbol{A}^{\exists}$, $\boldsymbol{b}^{\forall}$, $\boldsymbol{b}^{\exists}$,

with $n \in \{2, 3\}$, $m \in \mathbb{N}$,

we have to "see" AE-solution set Ξ_{AE}.

Known programs for visualization of AE-solution sets

- Siegfried Rump, Intlab function `plotlinsol` in MATLAB
- Walter Krämer and Gregor Paw, Java applet
- Walter Krämer and Sven Braun, package in Maple
- Evgenija Popova, online programs for united solution set, AE-solution set and parametric AE-solution set
- Irene Sharaya, file-program in PostScript

Drawbacks of these visualization programs

author(s)	solution type	size of system	process unbounded sets	process thin sets
Rump Z.	USS	3×3	$-$	$+$
Krämer W., Paw G.	USS	3×3	$\mp$	$\mp$
Krämer W., Braun S.	USS	3×3	$\mp$	$\mp$
Popova E.D.	USS	3×3	$\mp$	$-$
Popova E.D.	AEss	2×2	$\mp$	$-$
Sharaya I.A.	AEss	2×2	$+$	$+$

Drawbacks of these visualization programs

author(s)	solution type	size of system	process unbounded sets	process thin sets
Rump Z.	USS	3×3	$-$	$+$
Krämer W., Paw G.	USS	3×3	$\mp$	$\mp$
Krämer W., Braun S.	USS	3×3	$\mp$	$\mp$
Popova E.D.	USS	3×3	$\mp$	$-$
Popova E.D.	AEss	2×2	$\mp$	$-$
Sharaya I.A.	AEss	2×2	$+$	$+$

Drawbacks of these visualization programs

author(s)	solution type	size of system	process unbounded sets	process thin sets
Rump Z.	USS	3×3	$-$	$+$
Krämer W., Paw G.	USS	3×3	$\mp$	$\mp$
Krämer W., Braun S.	USS	3×3	$\mp$	$\mp$
Popova E.D.	USS	3×3	$\mp$	$-$
Popova E.D.	AEss	2×2	$\mp$	$-$
Sharaya I.A.	AEss	2×2	$+$	$+$

Drawbacks of these visualization programs

author(s)	solution type	size of system	process unbounded sets	process thin sets
Rump Z.	USS	3×3	$-$	$+$
Krämer W., Paw G.	USS	3×3	$\mp$	$\mp$
Krämer W., Braun S.	USS	3×3	$\mp$	$\mp$
Popova E.D.	USS	3×3	$\mp$	$-$
Popova E.D.	AEss	2×2	$\mp$	$-$
Sharaya I.A.	AEss	2×2	$+$	$+$

Drawbacks of these visualization programs

author(s)	solution type	size of system	process unbounded sets	process thin sets
Rump Z.	USS	3×3	$-$	$+$
Krämer W., Paw G.	USS	3×3	$\mp$	$\mp$
Krämer W., Braun S.	USS	3×3	$\mp$	$\mp$
Popova E.D.	USS	3×3	$\mp$	$-$
Popova E.D.	AEss	2×2	$\mp$	$-$
Sharaya I.A.	AEss	2×2	$+$	$+$

Drawbacks of these visualization programs

author(s)	solution type	size of system	process unbounded sets	process thin sets
Rump Z.	USS	3×3	$-$	$+$
Krämer W., Paw G.	USS	3×3	$\mp$	$\mp$
Krämer W., Braun S.	USS	3×3	$\mp$	$\mp$
Popova E.D.	USS	3×3	$\mp$	$-$
Popova E.D.	AEss	2×2	$\mp$	$-$
Sharaya I.A.	AEss	2×2	$+$	$+$

New program
for visualization
of AE-solution sets

This is a package in Matlab language

with subpackages for 2D and 3D cases.

2D-case. Notation

po$_k$ – intersection of Ξ_{AE} with k-th orthant (piece in orthant),

• – vertex of po,

/ – edge of po,

■ – interior of po,

⋮ – coordinate axis.

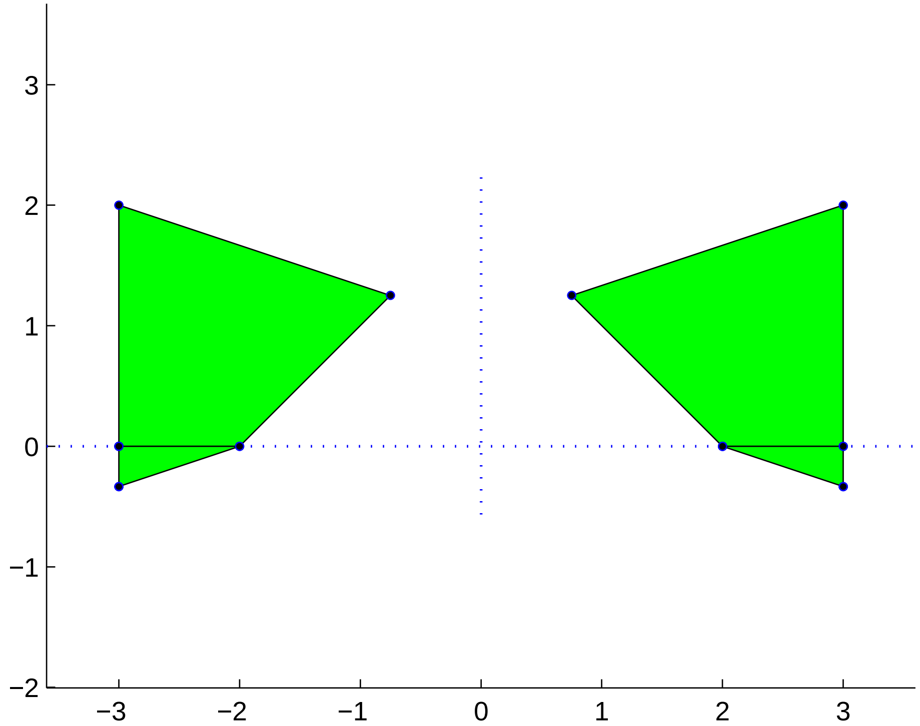
Capability for 2D tasks

1) arbitrary quantifiers

Example:

$$A = \begin{pmatrix} 1 & 0 \\ [-1, 1] & [1, 3] \end{pmatrix}, \; b = \begin{pmatrix} [-3, 3] \\ [2, 3] \end{pmatrix},$$

solution type — EE E / EA E



Capability for 2D tasks

2) rectangular matrix

Example (1000 rows):

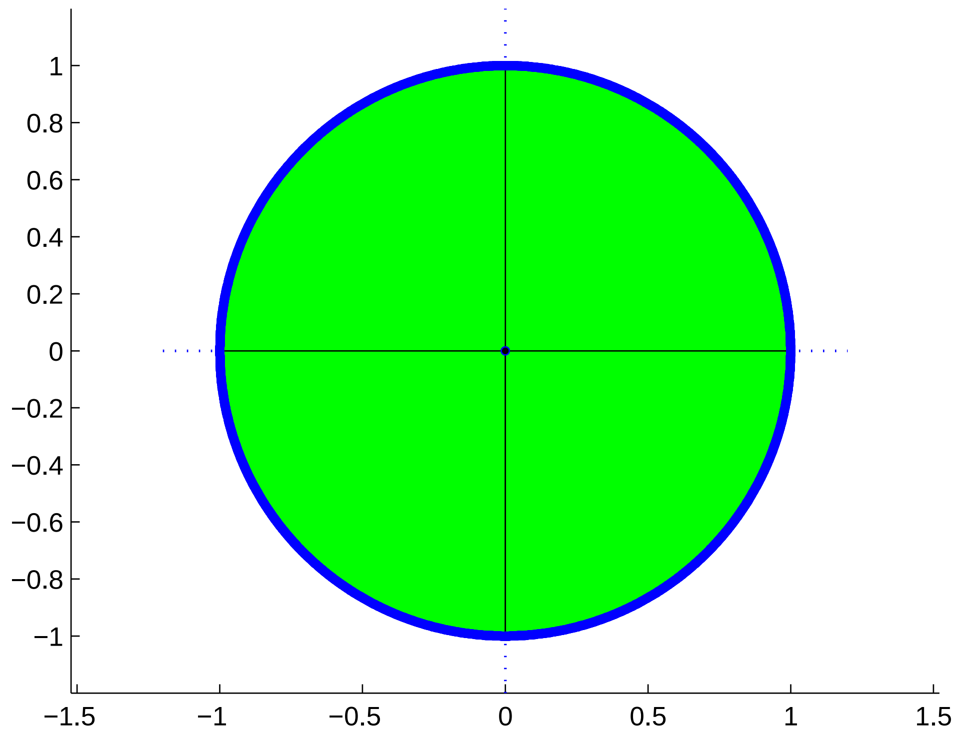
m=1000,

$\overline{A}_{i:} = \left(\sin\frac{\pi i}{2m},\ \cos\frac{\pi i}{2m}\right),$

$\underline{A}_{i:} = -\overline{A}_i,$

$\boldsymbol{b}_i = [-2, 1],$

solution type — tolerable.



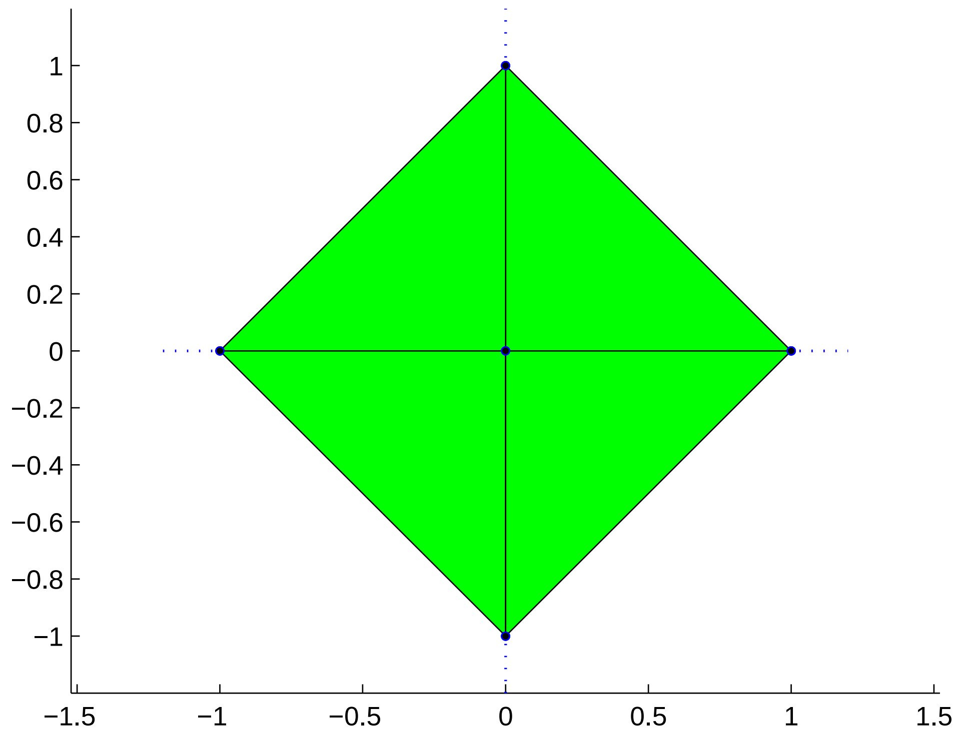
Capability for 2D tasks

2) rectangular matrix

Example (1 row):

$\boldsymbol{A} = ([-1, 1][-1, 1]),$
$\boldsymbol{b} = ([-1, 1]),$

solution type — tolerable.



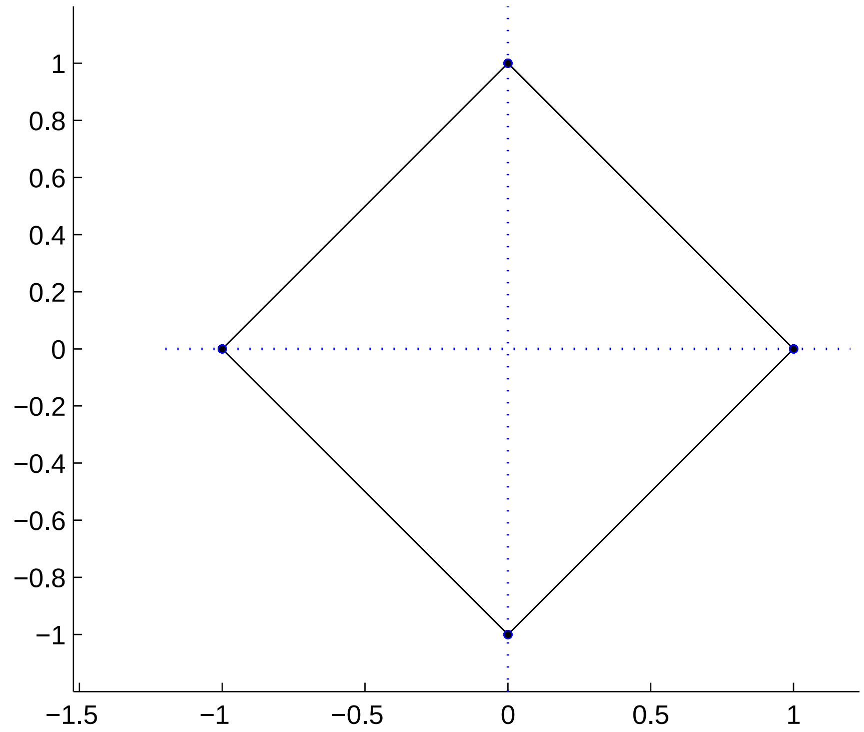
Capability for 2D tasks

3) **drawing thin sets** (vertices & edges of po$_k$) and **distinguishing between thin sets and sets with nonempty interior** (due to green interior – compare this example with previous one)

Example (bound of rhomb):

$$A = \begin{pmatrix} [-1,1] & [-1,1] \\ [-1,1] & [-1,1] \end{pmatrix}, \; b = \begin{pmatrix} [-1,1] \\ [1,2] \end{pmatrix},$$

solution type —
AA E
EE E



Capability for 2D tasks

4) auto-choose of Drawing Box

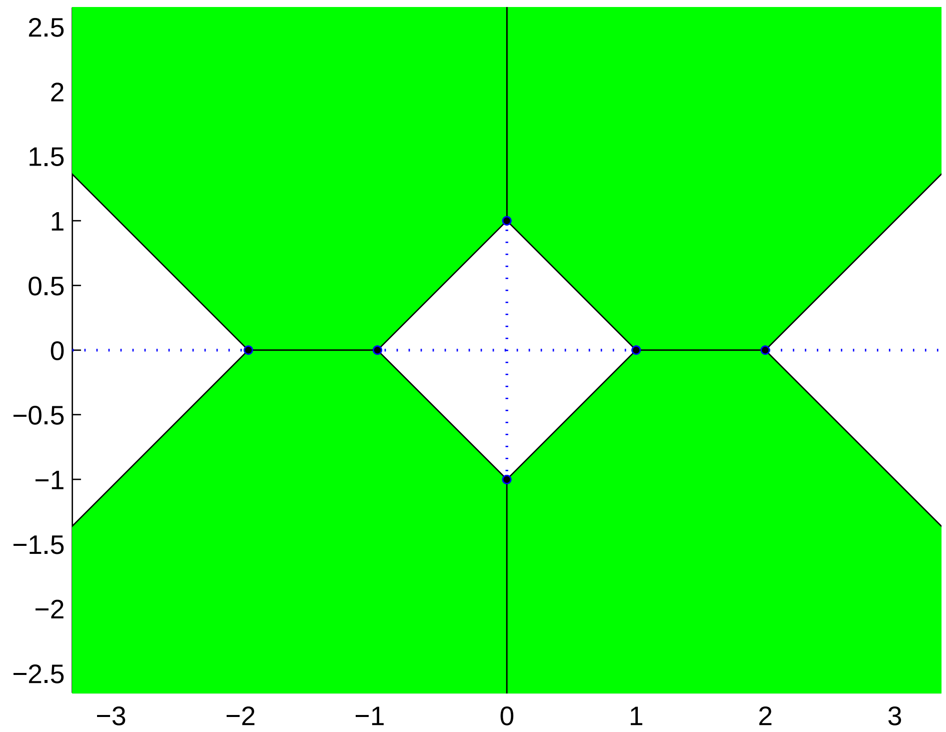
(even for unbounded sets)

Example (unbounded set):

$$\boldsymbol{A} = \begin{pmatrix} [-1,1] & [-1,1] \\ -1 & [-1,1] \end{pmatrix},$$

$$\boldsymbol{b} = \begin{pmatrix} 1 \\ [-2,2] \end{pmatrix},$$

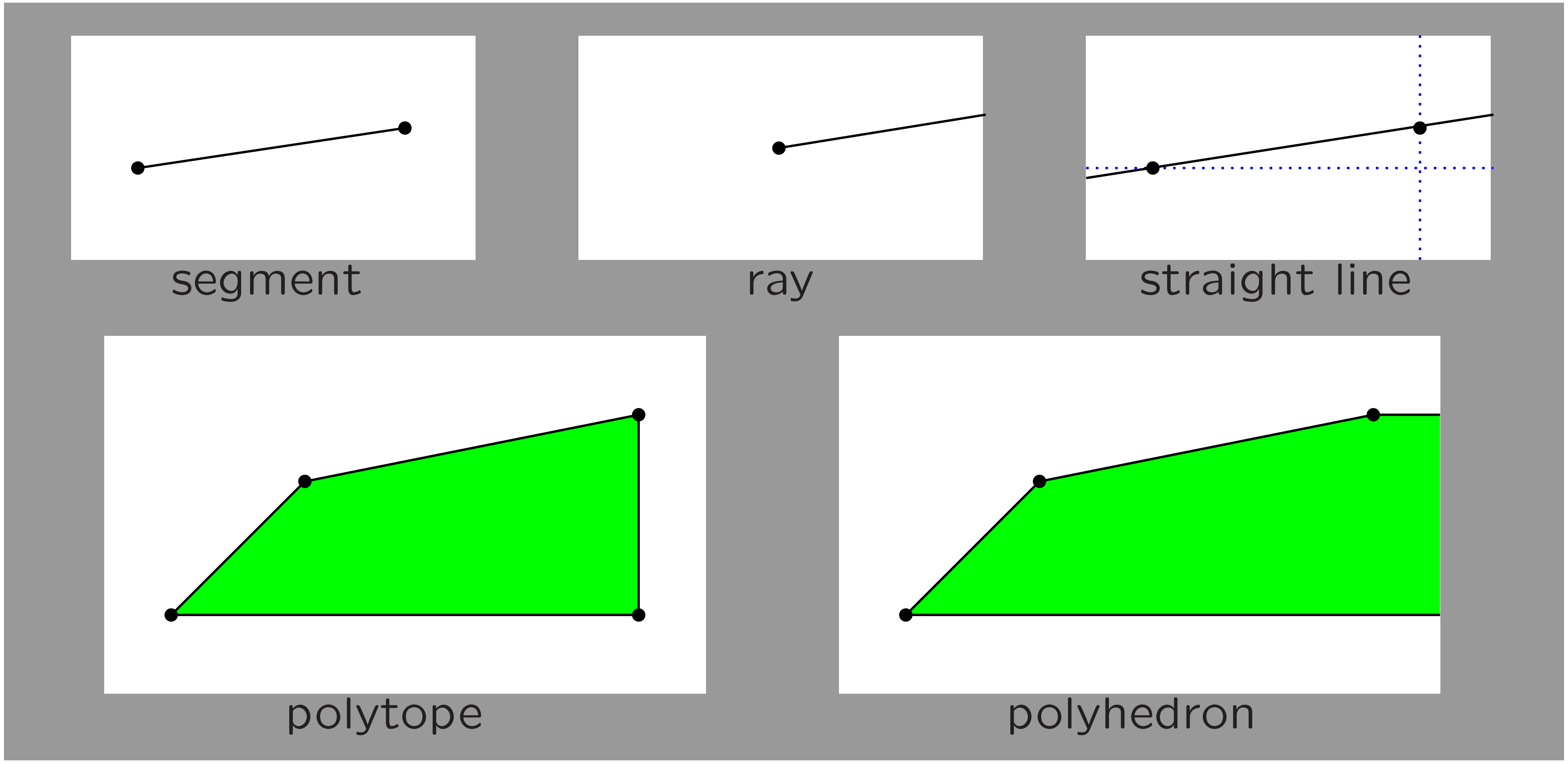
solution type — united.



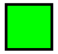
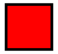
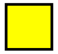
Capability for 2D tasks

5) distinguishing between bounded and unbounded sets
(unbounded set has points on the border of Drawing Box)

Examples:



3D-case. Notation

po_k –	intersection of Ξ_{AE} with k-th orthant (piece in orthant),
• –	vertex of po,
/ –	edge of po,
 –	real facet,
 –	cut facet,
 –	prescribed facet.

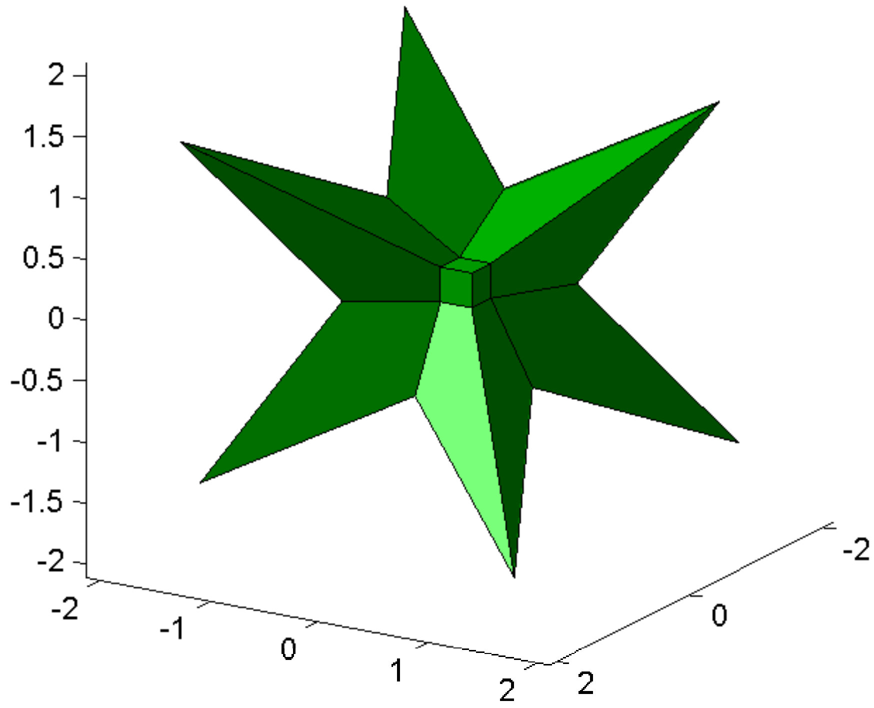
Capability for 3D tasks

1) availability of Matlab tools (zoom, rotation, light, ...)

Example (Neumaier star):

$$\boldsymbol{A} = \begin{pmatrix} 3.5 & [0,2] & [0,2] \\ [0,2] & 3.5 & [0,2] \\ [0,2] & [0,2] & 3.5 \end{pmatrix}, \ \boldsymbol{b} = \begin{pmatrix} [-1,1] \\ [-1,1] \\ [-1,1] \end{pmatrix},$$

solution type — united.

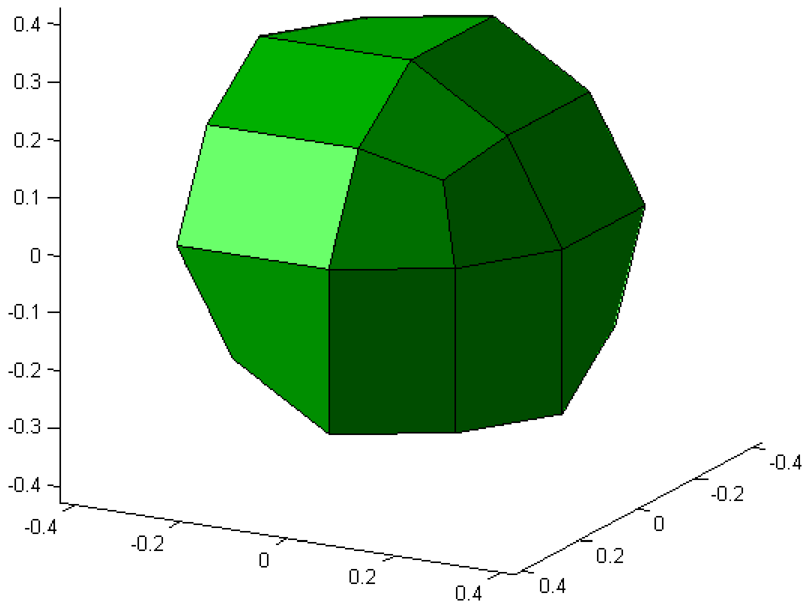


Capability for 3D tasks

2) arbitrary quantifiers

Example (diamond):

$$\boldsymbol{A} = \begin{pmatrix} 3.5 & [0,2] & [0,2] \\ [0,2] & 3.5 & [0,2] \\ [0,2] & [0,2] & 3.5 \end{pmatrix}, \ \boldsymbol{b} = \begin{pmatrix} [-1,1] \\ [-1,1] \\ [-1,1] \end{pmatrix},$$



solution type — tolerable.

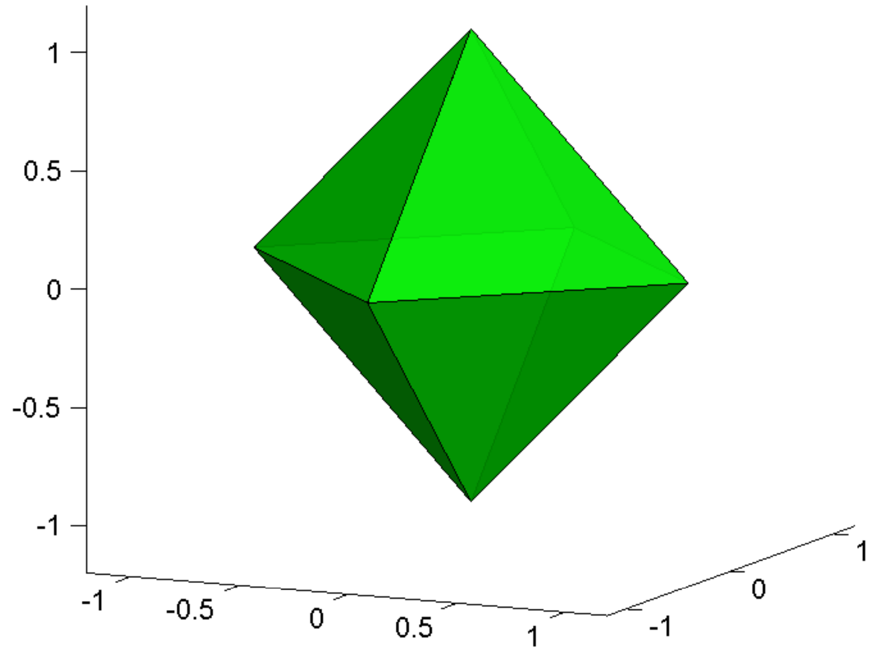
Capability for 3D tasks

3) rectangular matrix

Example (1 row):

$\boldsymbol{A} = ([-1,1][-1,1][-1,1])$,
$\boldsymbol{b} = ([-1,1])$,

solution type — tolerable.



Capability for 3D tasks

3) rectangular matrix

Example (100 rows):

$k = 10,$

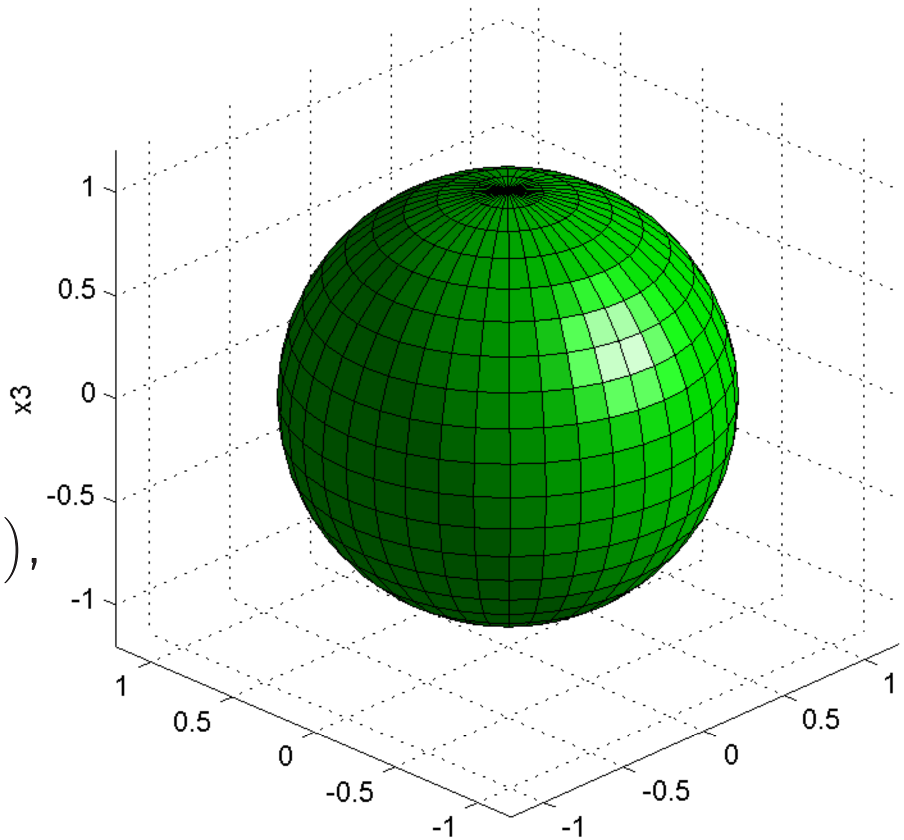
$\alpha, \beta = \frac{\pi}{4k} : \frac{\pi}{2k} : \frac{(2k-1)\pi}{4k},$

$\overline{A}_{i:} = \big(\cos(\alpha)\cos(\beta),\ \sin(\alpha)\cos(\beta),\ \sin(\beta)\big),$

$\underline{A}_{i:} = -\overline{A},$

$\boldsymbol{b}_i = [-2, 1],$

solution type — tolerable.

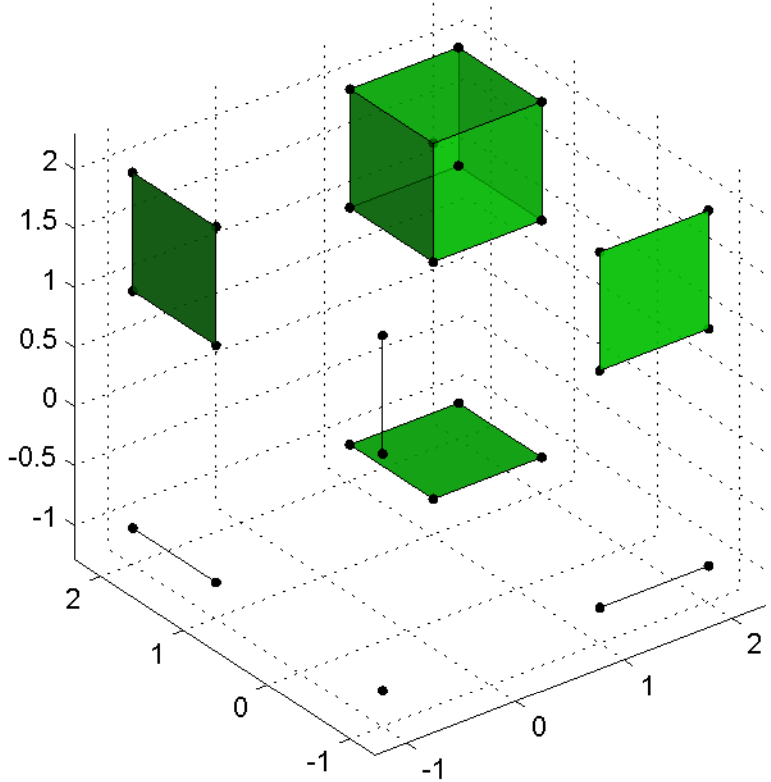


Capability for 3D tasks

4) drawing thin sets (input argument ‘OrientPoints’ must be equal 1)

Example:

$$\begin{array}{cc} \boldsymbol{A} & \boldsymbol{b} \\ \begin{pmatrix} [-1,1] & 0 & 0 \\ 0 & [-1,1] & 0 \\ 0 & 0 & [-1,1] \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} & \begin{pmatrix} 1 \\ 1 \\ 1 \\ [-1,2] \\ [-1,2] \\ [-1,2] \end{pmatrix}, \end{array}$$



solution type — united.

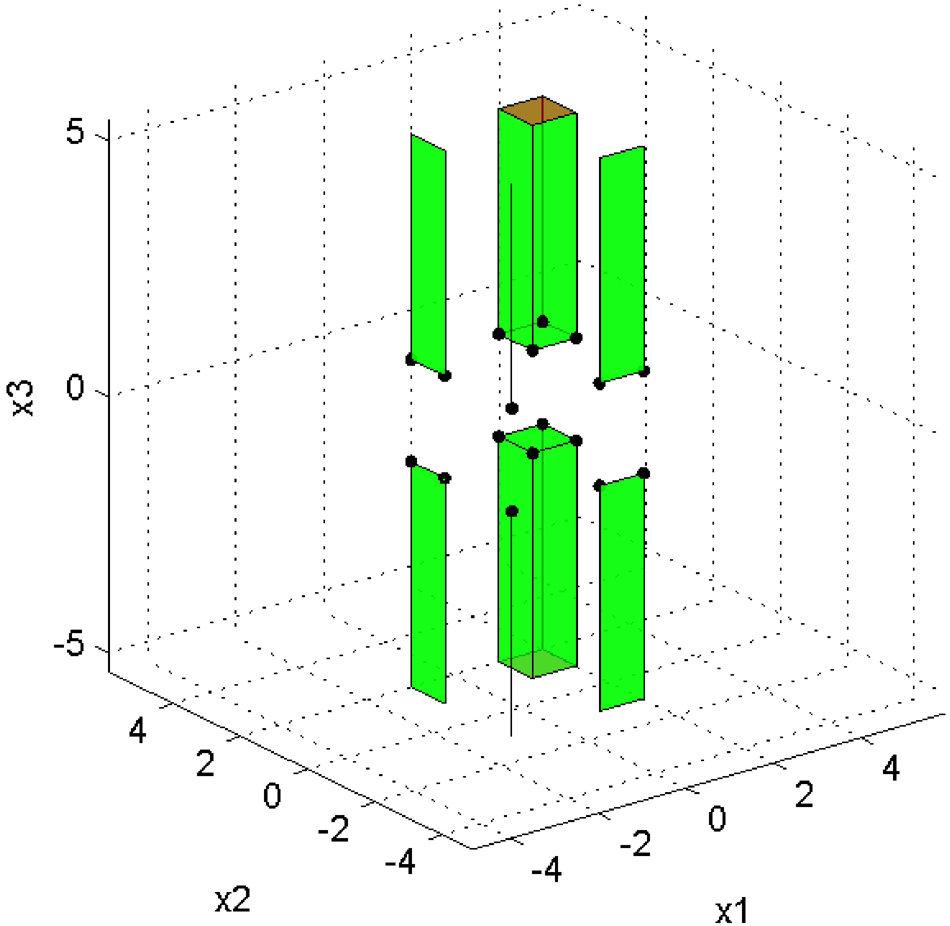
Capability for 3D tasks

5) auto-choose of Drawing Box (even for unbounded sets)

Example:

$$\begin{pmatrix} [-1,1] & 0 & 0 \\ 0 & [-1,1] & 0 \\ 0 & 0 & [-1,1] \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}_{\boldsymbol{A}} \quad \begin{pmatrix} 1 \\ 1 \\ 1 \\ [-1,2] \\ [-1,2] \end{pmatrix}_{\boldsymbol{b}},$$

solution type — united.



Capability for 3D tasks

6) distinguishing between bounded and unbounded sets

Main characteristic — unbounded set has points
on the facets of auto-choosed Drawing Box,

complementary characteristics —

- cut facet has not vertices,
- 2 dimensional cut facet is red.

(Compare two previous examples.)

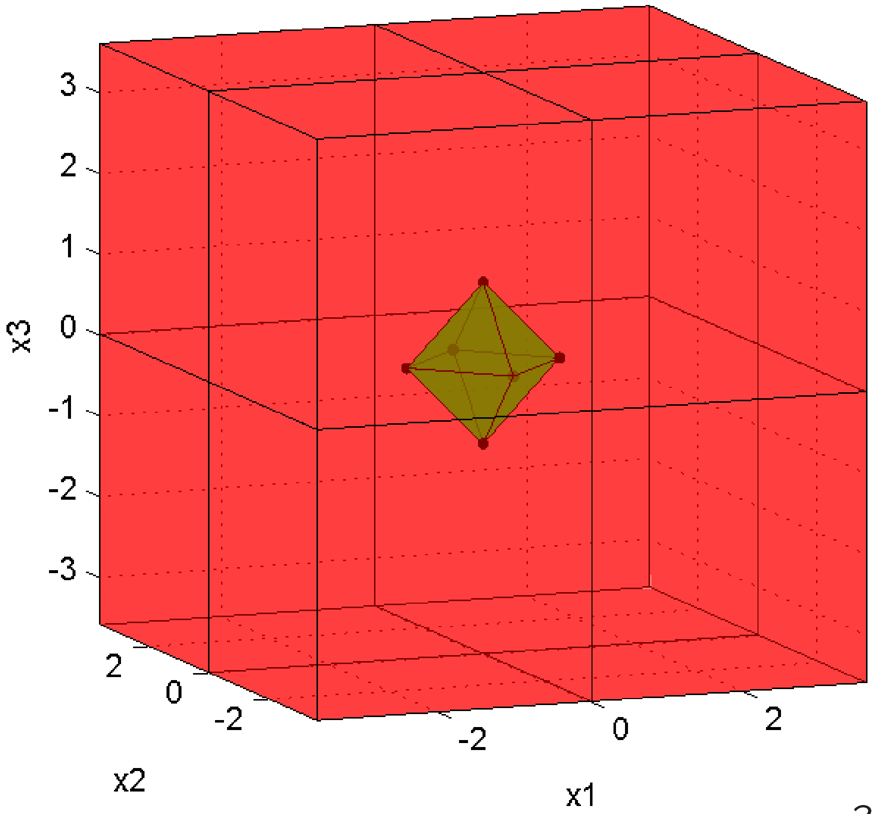
Capability for 3D tasks

6) transparency
always for cut and prescribed facets
and as input argument for real facets

Example ($\mathbb{R}^3$ with cave):

$\boldsymbol{A} = ([-1, 1][-1, 1][-1, 1])$,
$\boldsymbol{b} = [1, 2]$,

solution type — united.



Capability for 3D tasks

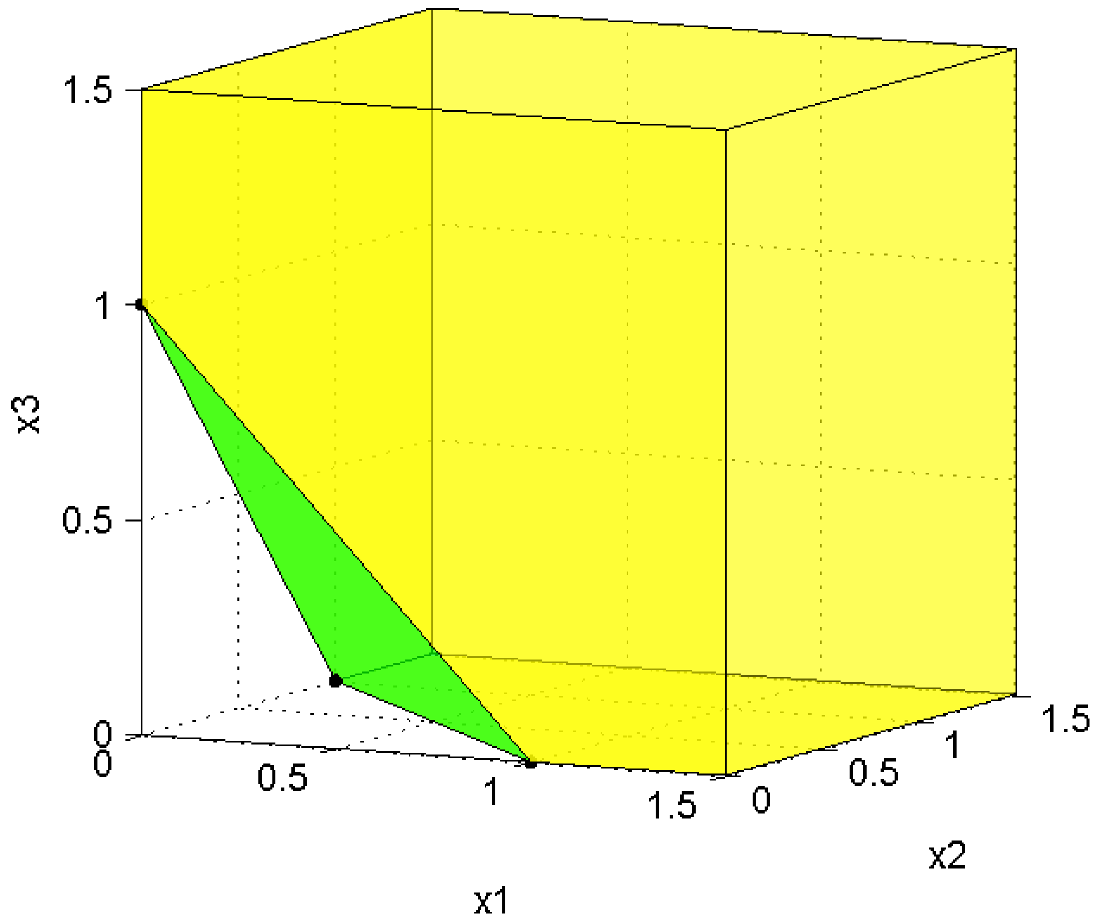
7) **Prescribed Box** as optional input argument

Example ($\mathbb{R}^3$ with cave):

$\boldsymbol{A} = ([-1, 1][-1, 1][-1, 1])$,
$\boldsymbol{b} = [1, 2]$,

solution type — united,

Prescribed Box —
([0,1.5] [0,1.5] [0,1.5]).



The codes of the presented program

are open and available from

http://www.nsc.ru/interval/Programing/

Basic ideas of the program:

How to draw the polytope?

How to draw thin sets?

How to draw unbounded sets?

How to find ordered list of vertices for polytope, wich is described as a system of 2D linear inequalities?

How to draw the polytope?

To use Matlab functions `fill` and `fiil3`.

(They draw 2D polytope by ordered list of its vertices in $\mathbb{R}^2$ and $\mathbb{R}^3$ respectively.)

How to draw thin sets?

To draw 1D and 2D facets by functions `fill` and `fill3` and to draw vertices using functions `plot` or `scatter`.

How to draw unbounded sets?

To find $\square\left(\bigcup_i \text{vertices}(\text{po})_i\right)$,
to increase the received interval,
and to use the increased interval as a Cut Box.

How to find ordered list of vertices for polytope, wich is described as a system of 2D linear inequalities?

To use a boundary interval method.

Boundary interval method

Boundary interval method ‘was born’ this year.

It is assigned for visualization of

solution set to system of linear inequalities,
solution set to system of two-sided linear inequalities,
and AE-solution set to system of interval linear equations.

Basic terms of the method are

boundary interval
and boundary interval matrix.

Boundary interval (definition)

Let us be given the system of linear inequalities $Ax \geqslant b$ with $A \in \mathbb{R}^{m\times 2}$, $b \in \mathbb{R}^m$. If the set $\{\, x \mid (A_{i:}x = b_i) \,\&\, (Ax \geqslant b) \,\}$ for $i \in \{1, \ldots, m\}$ is not empty, we call it *boundary interval*.

A boundary interval, as a set of points on the plane, may be a single point, a segment, a ray, and a straight line.

Boundary interval (How to evaluate?)

For i

1) go to inner coordinate of straight line $A_{i:}x = b_i$,

   i.e. replace x by $\dfrac{b_i}{||A_{i:}||_2} A_{i:}^\top + (-A_{i2}, A_{i1})^\top t$,

3) evaluate interval $[\underline{t}, \overline{t}]$ of inner coordinate t

   from 1D system of linear inequalities,

4) rewrite points $\underline{t}$ and $\overline{t}$,

   in outer coordinates.

Boundary interval (How to write it?)

$$\Big(\quad \underbrace{* \quad *}_{\substack{\text{begining} \\ \in \\ \overline{\mathbb{R}}^2}} \quad \underbrace{* \quad *}_{\substack{\text{end} \\ \in \\ \overline{\mathbb{R}}^2}} \quad \underset{\substack{\text{inequality} \\ \text{number} \\ \in \\ \mathbb{N}}}{*} \quad \Big)$$

Boundary interval matrix

(What knowledge about solution set it gives?)

$$\begin{array}{ccccccc} (& * & * & * & * & * &) \\ (& * & * & * & * & * &) \\ & & . & . & . & & \\ (& * & * & * & * & * &) \\ (& * & * & * & * & * &) \end{array} \longrightarrow$$

- emptiness
- boundedness
- ordered list of vertices

THANK YOU