A Framework for Applying First-Order Methods to General Convex Conic Optimization Problems

Jim Renegar

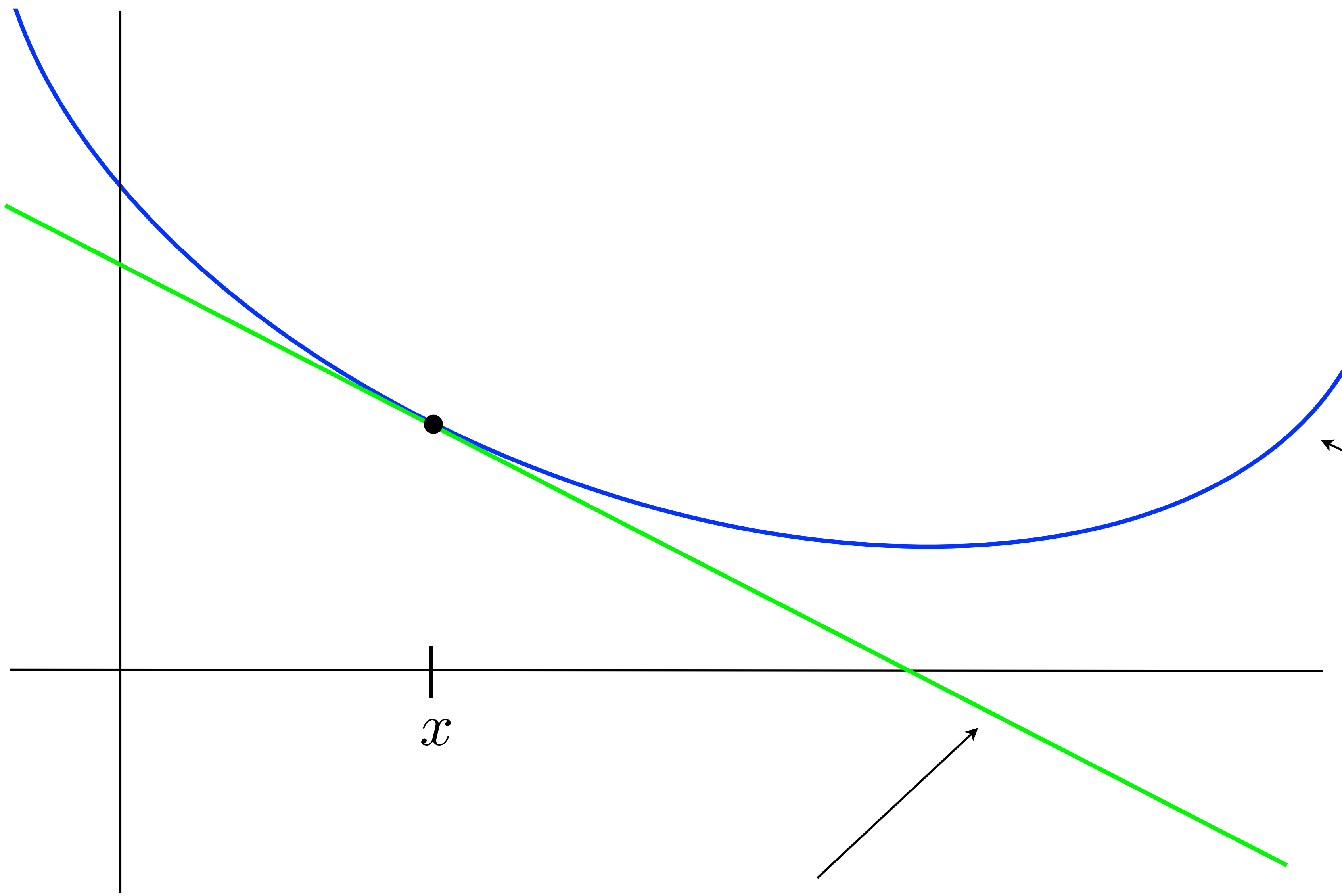
School of Operations Research
Cornell University

LNMB Lunteren Conference, 2016

convex function (assume lower semicontinuous)

$$
\begin{array}{ll}
\min & f(x) \\
\text{s.t.} & x \in Q
\end{array}
$$

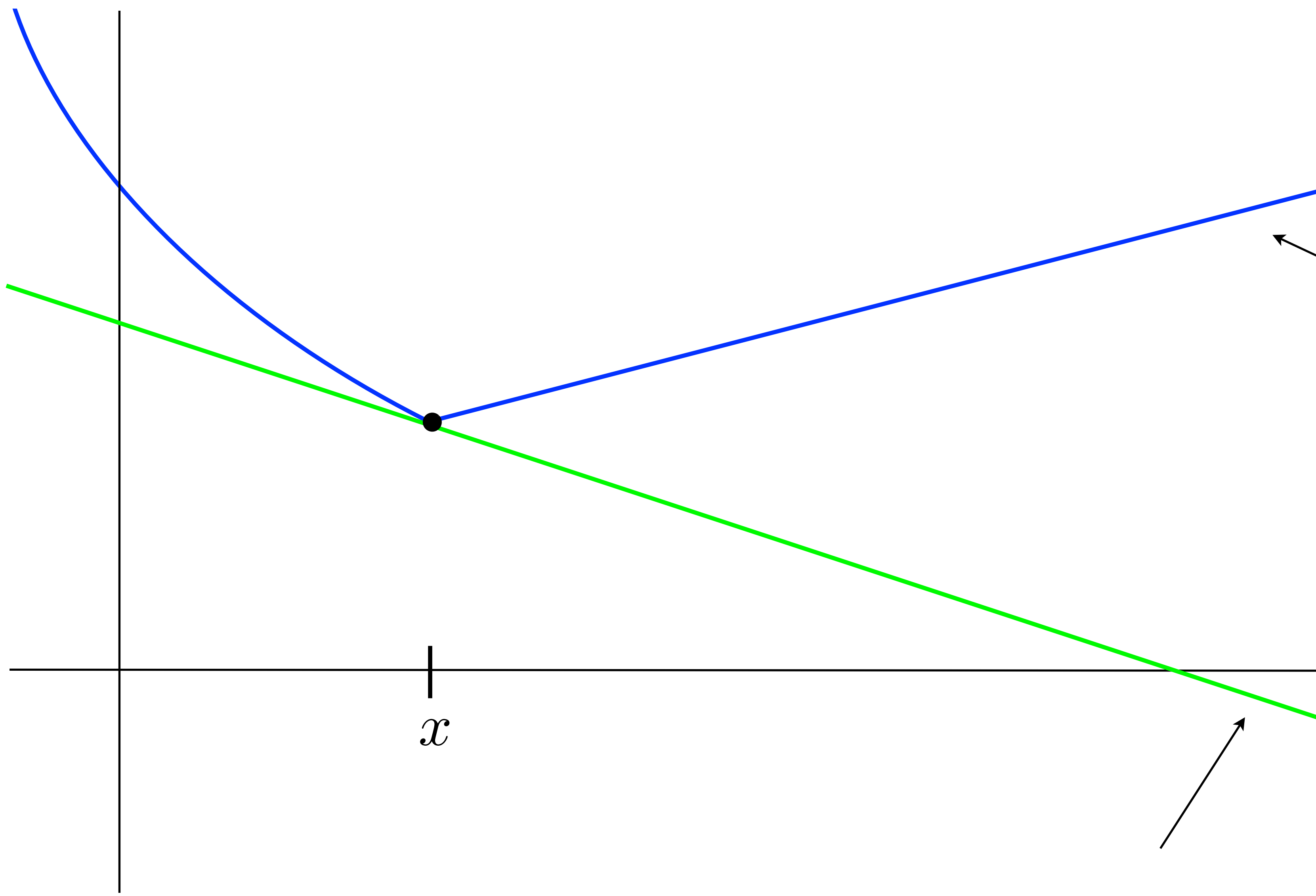
closed, convex set



convex function (assume lower semicontinuous)

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & x \in Q \end{array}$$

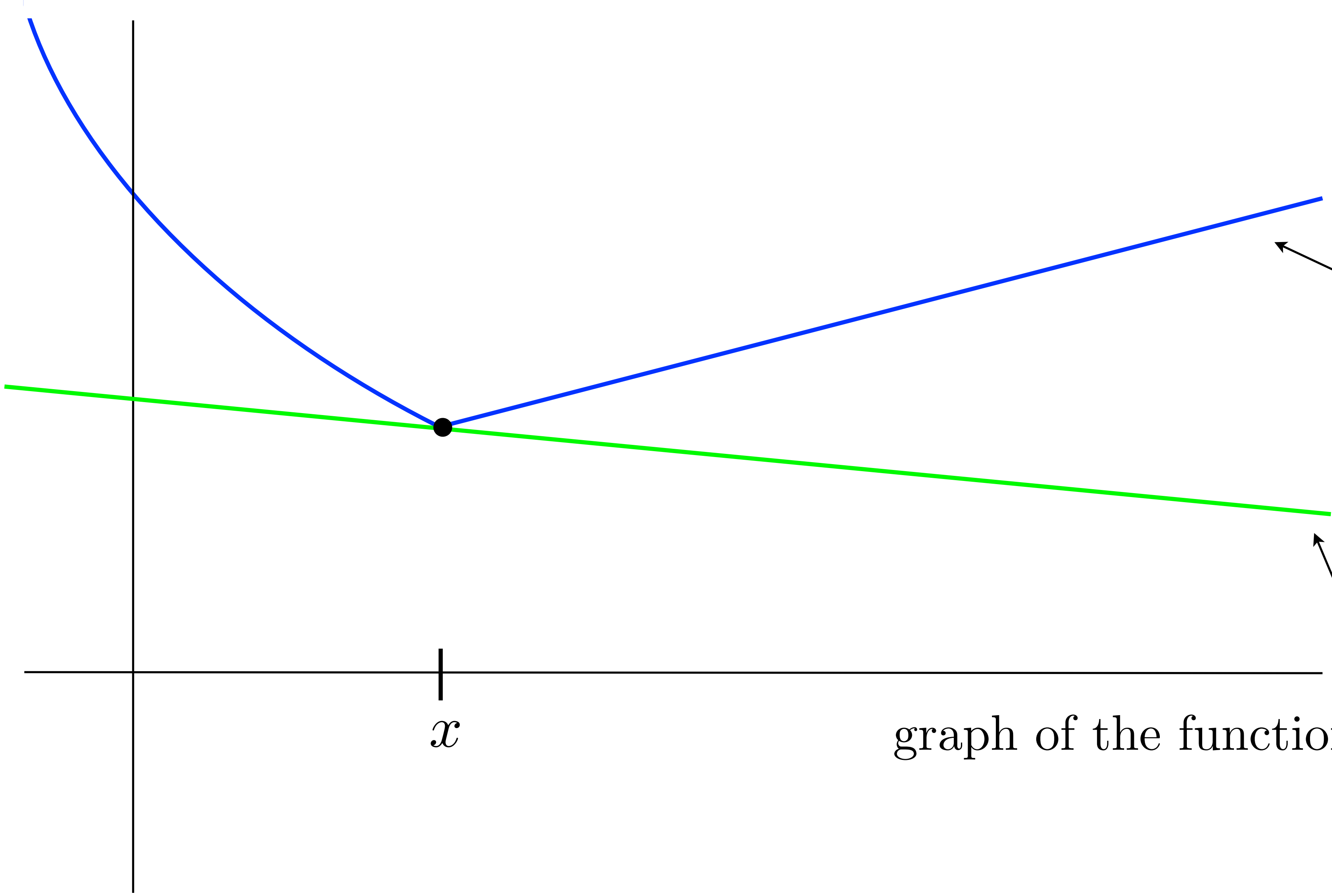
closed, convex set



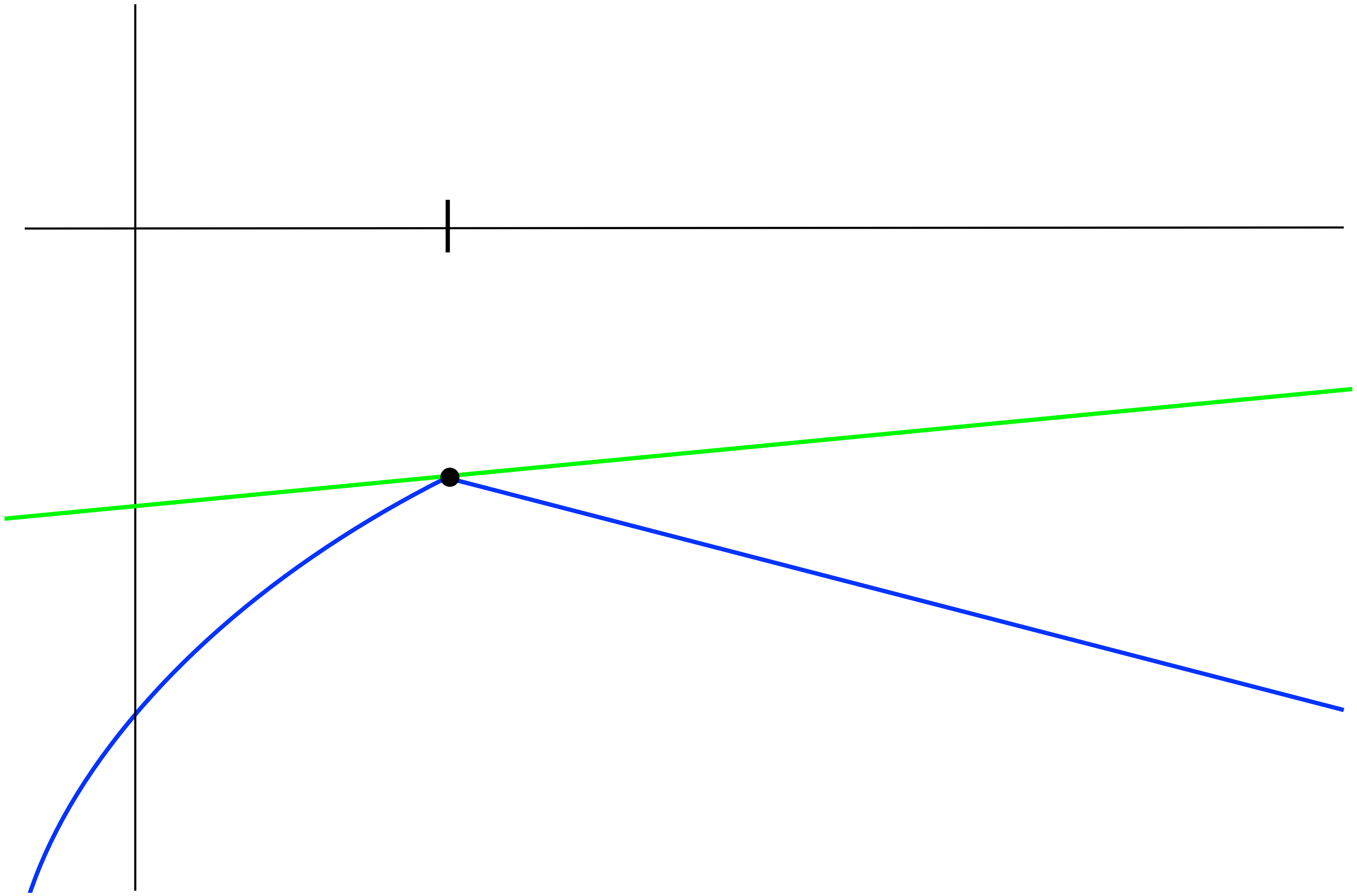
convex function (assume lower semicontinuous)

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & x \in Q \end{array}$$

closed, convex set



The set of all subgradients at x is denoted $\partial f(x)$ – the “subdifferential” at x



For a concave function, supgradients play the analogous role.

convex function (assume lower semicontinuous)

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & x \in Q \end{array}$$

closed, convex set

Assume f is Lipschitz-continuous on an open neighborhood of Q:

$$|f(x) - f(y)| \leq M \|x - y\|$$

Lipschitz constant

Goal: Compute $x \in Q$ satisfying $f(x) \leq f^* + \epsilon$

optimal value

A typical subgradient method:

Initialize: $x_0 \in Q$

Iterate: Compute $g_k \in \partial f(x_k)$, and let $x_{k+1} = P_Q\left(x_k - \frac{\epsilon}{\|g_k\|^2} g_k \right)$

where P_Q *is projection onto* Q

A typical theorem:

an optimal solution

$$\ell \geq \left(\frac{M\|x_0 - x^*\|}{\epsilon} \right)^2 \quad \Rightarrow \quad \min_{k \leq \ell} f(x_k) \leq f^* + \epsilon$$

In the special case of linear programming this becomes ...

$$\begin{array}{rl} \min & c^T x \\ \text{s.t.} & Ax \geq b \end{array} \qquad \text{so } Q = \{x : Ax \geq b\}$$

Then, of course, the objective function is Lipschitz continuous:

$$|c^T x - c^T y| \leq \underbrace{\|c\|}_{\text{Lipschitz constant}} \|x - y\|$$

Goal: Compute x satisfying $Ax \geq b$ and $c^T x \leq \underbrace{z^*}_{\text{optimal value}} + \epsilon$

A typical subgradient method:

Initialize: $x_0 \in Q$

Iterate: Let $x_{k+1} = P_Q(x_k - \frac{\epsilon}{\|c\|^2} c)$

where P_Q is projection onto Q

A typical theorem:

$$\ell \geq \left(\frac{\|c\| \, \|x_0 - \overbrace{x^*}^{\text{an optimal solution}}\|}{\epsilon} \right)^2 \quad \Rightarrow \quad \min_{k \leq \ell} c^T x_k \leq z^* + \epsilon$$

But in general, projecting onto $Q = \{x : Ax \geq b\}$
is no easier than solving linear programs!!!

A typical subgradient method:

Initialize: $x_0 \in Q$

Iterate: Let $x_{k+1} = P_Q(\, x_k - \frac{\epsilon}{\|c\|^2} c \,)$

where P_Q is projection onto Q

But in general, projecting onto $Q = \{x : Ax \geq b\}$ is no easier than solving linear programs!!!

There are ways, however, to use a subgradient method to "solve" an LP.

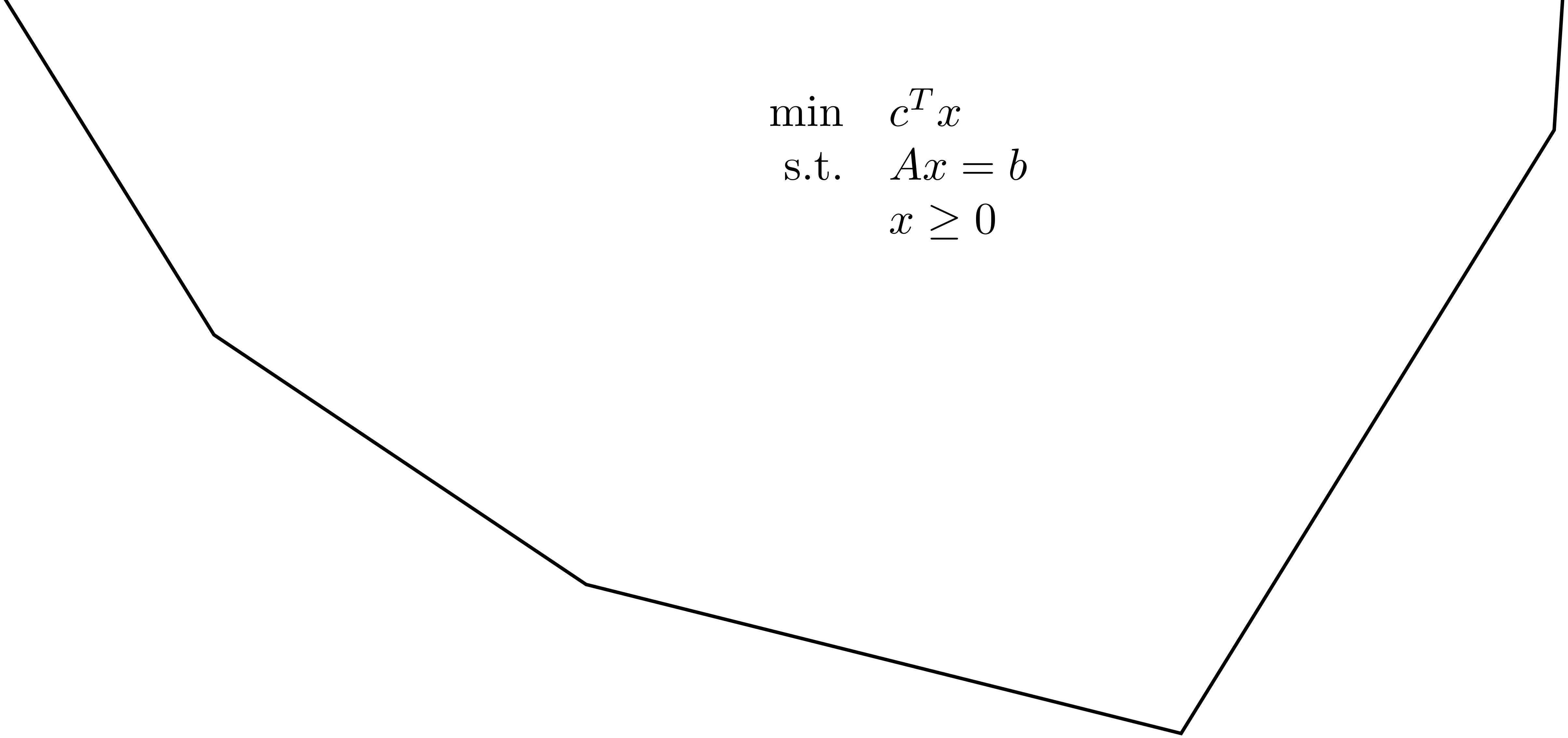
For example, here is a way to "approximate" an LP by an unconstrained convex optimization problem:

$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Ax \geq b \end{array} \quad \approx \quad \min \; c^T x + \underbrace{\gamma}_{\substack{\text{user-chosen} \\ \text{postive constant}}} \max\{0, b_i - \alpha_i^T x : i = 1, \ldots, m\}$$

However, the optimal solution for the problem on the right will not necessarily be feasible for LP.

In the literature, in fact, the only subgradient methods producing feasible iterates require the feasible region to be "simple."

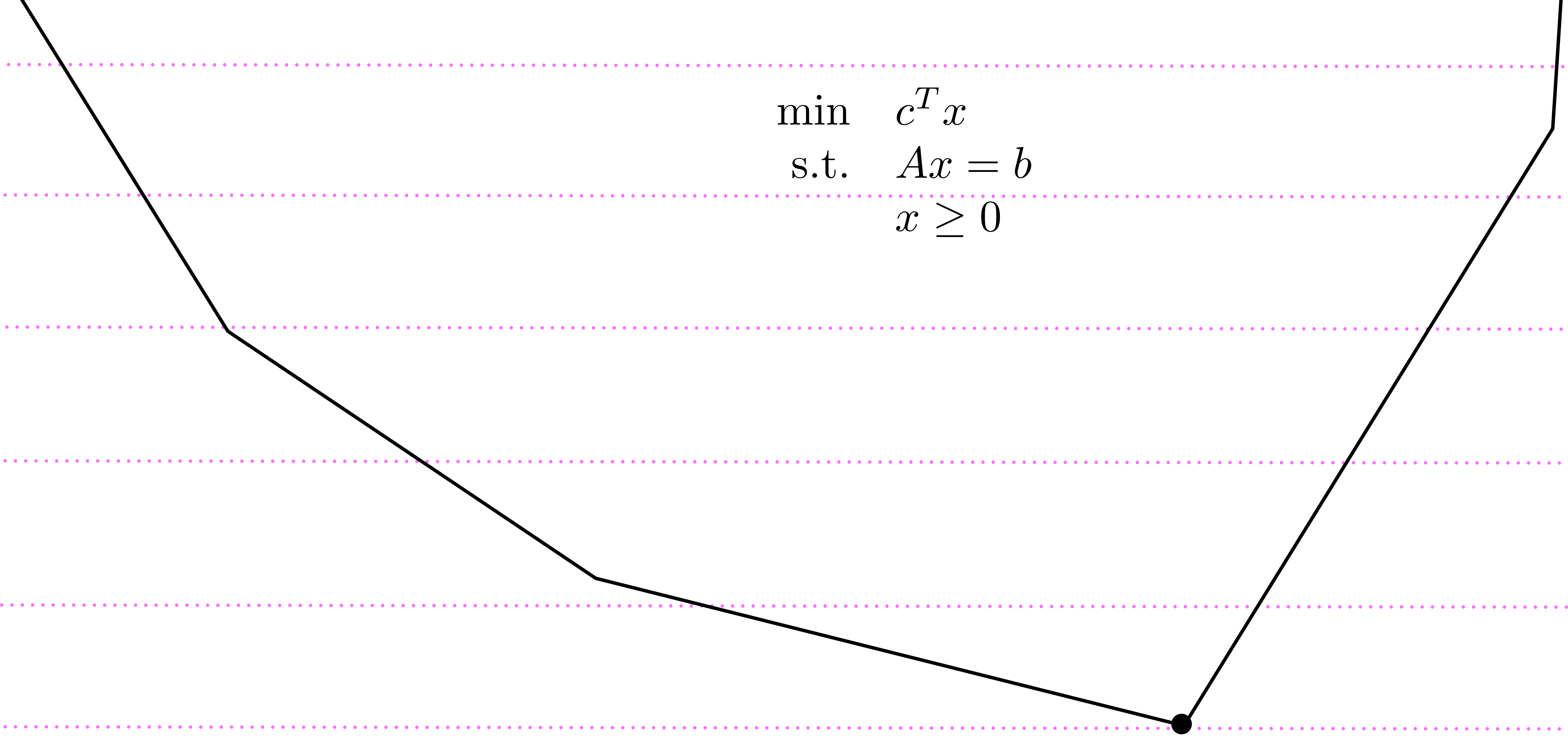
"Why is this?"



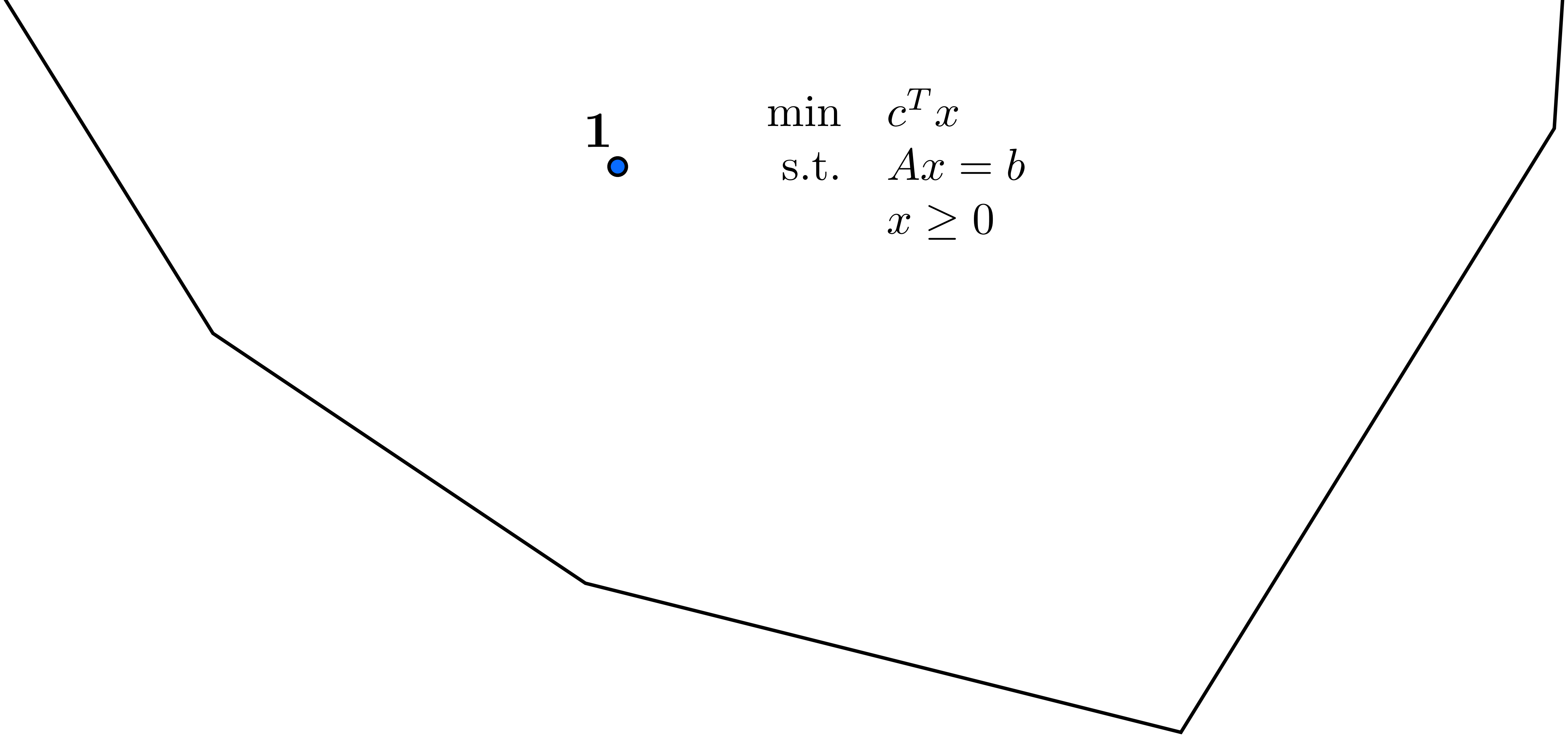
$$
\begin{aligned}
\min \quad & c^T x \\
\text{s.t.} \quad & Ax = b \\
& x \geq 0
\end{aligned}
$$

Think of this 2-dimensional plane
as being the slice of $\mathbb{R}^n$
cut out by $\{x : Ax = b\}$.

$$
\begin{aligned}
\min \quad & c^T x \\
\text{s.t.} \quad & Ax = b \\
& x \geq 0
\end{aligned}
$$



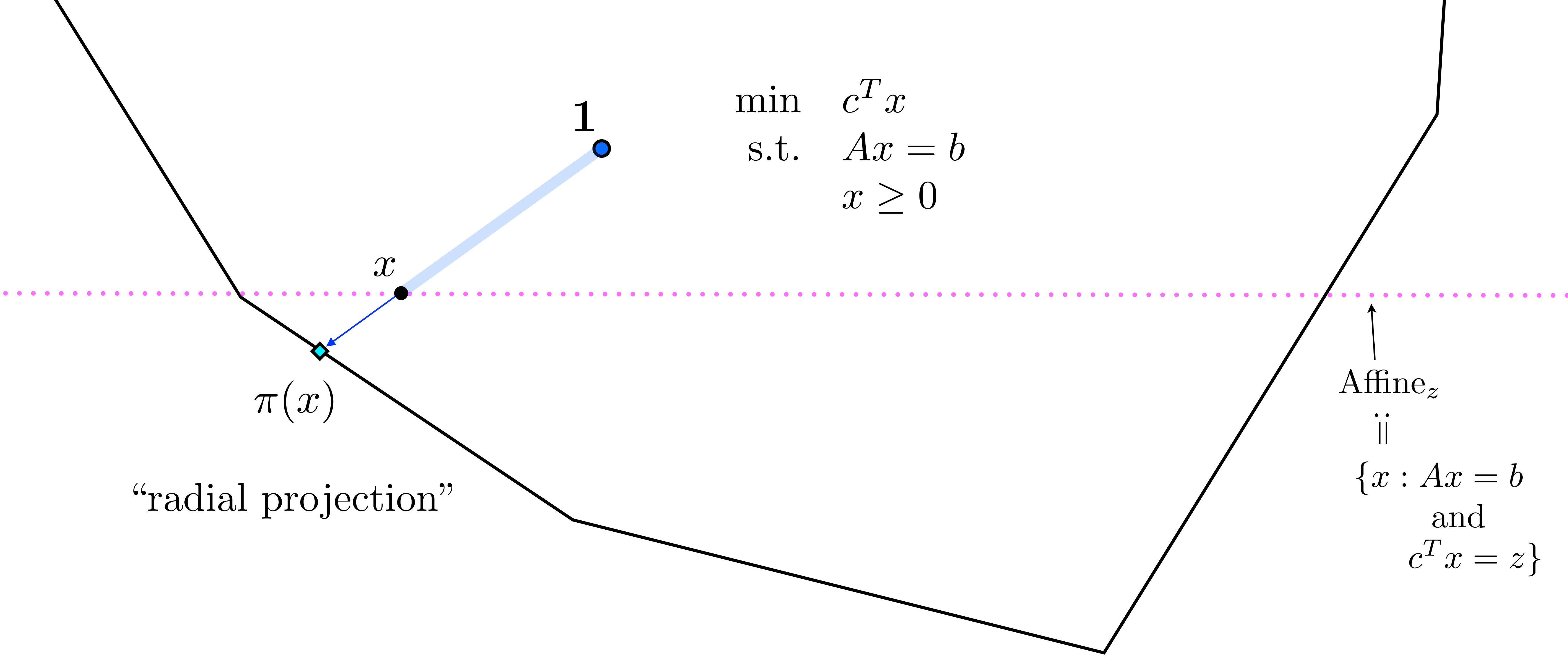
Assume the objective function $x \mapsto c^T x$ is constant on horizontal slices.

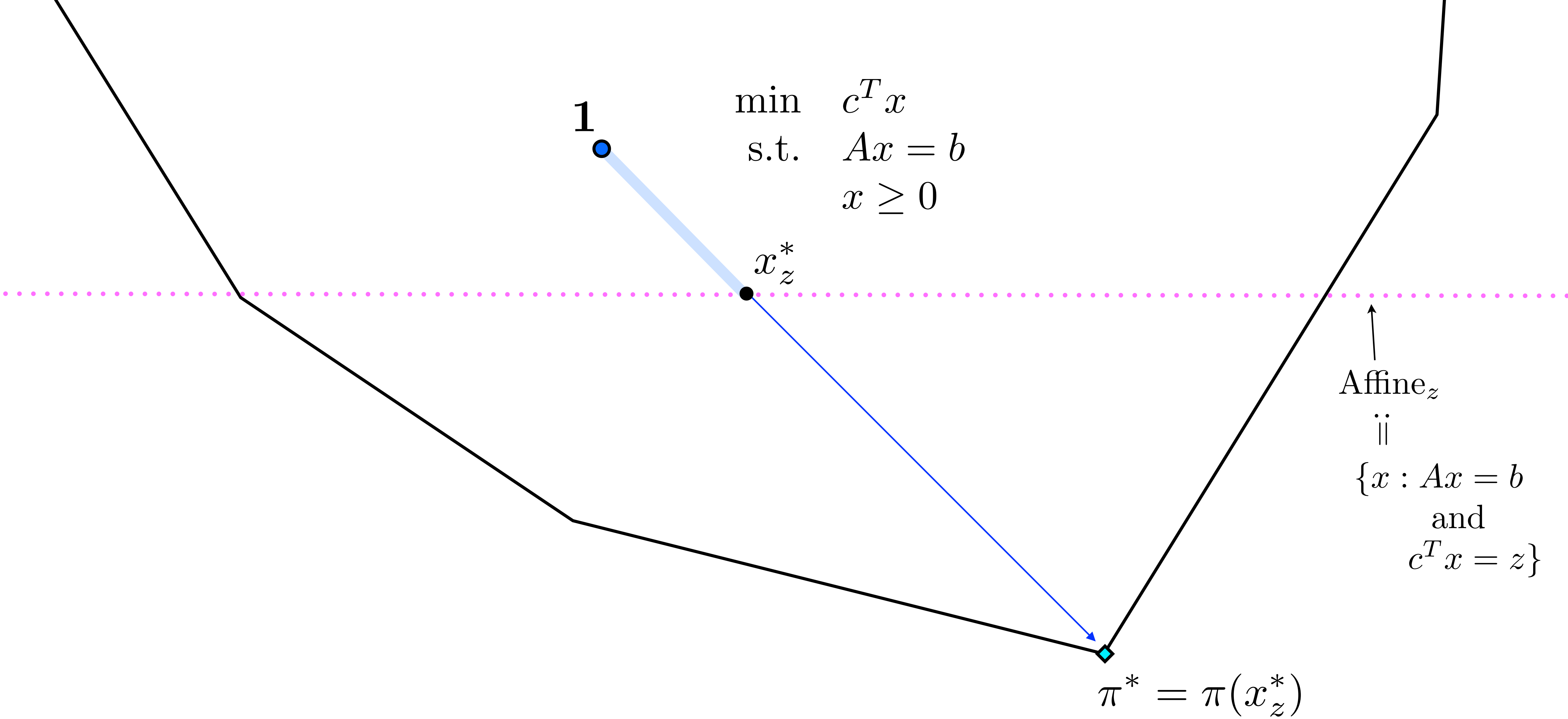


$$\begin{array}{rl} \min & c^T x \\ \text{s.t.} & Ax = b \\ & x \geq 0 \end{array}$$

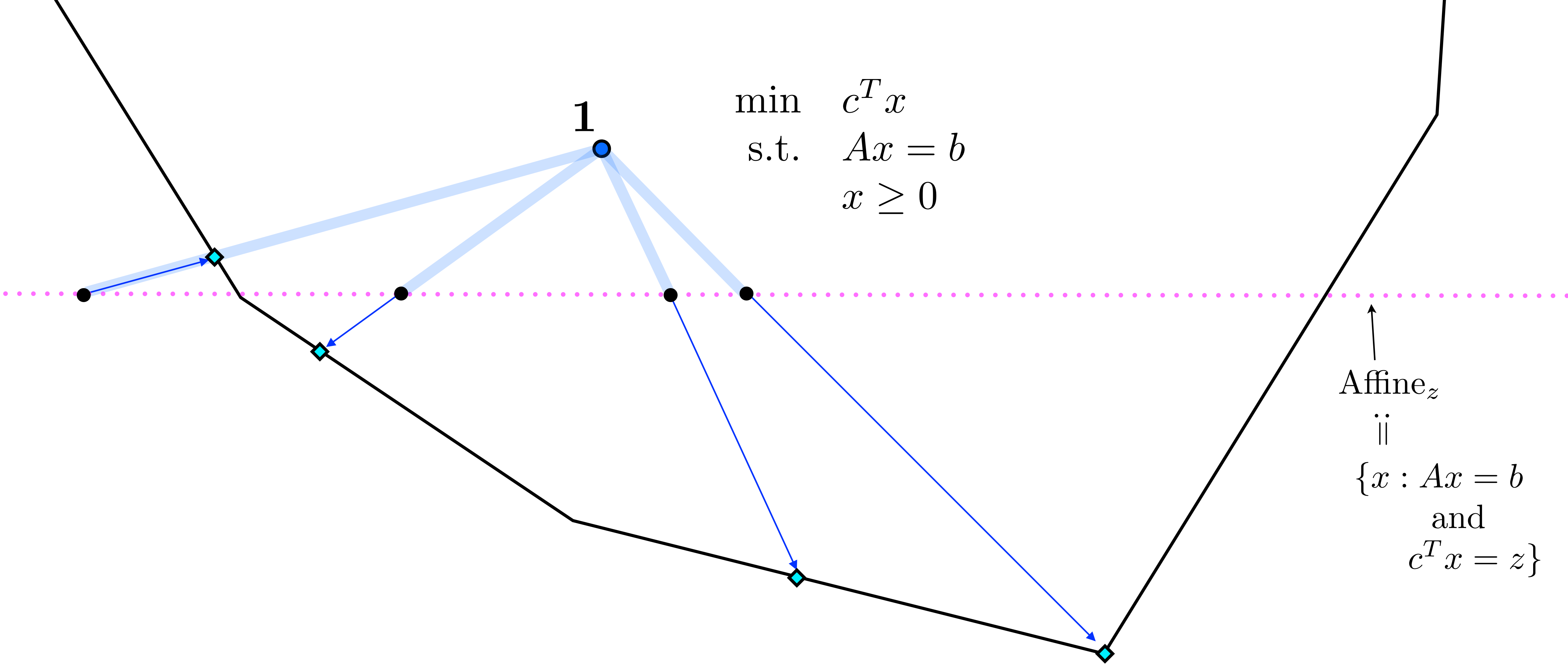
To begin with simplicity, assume the vector of all one's is feasible.

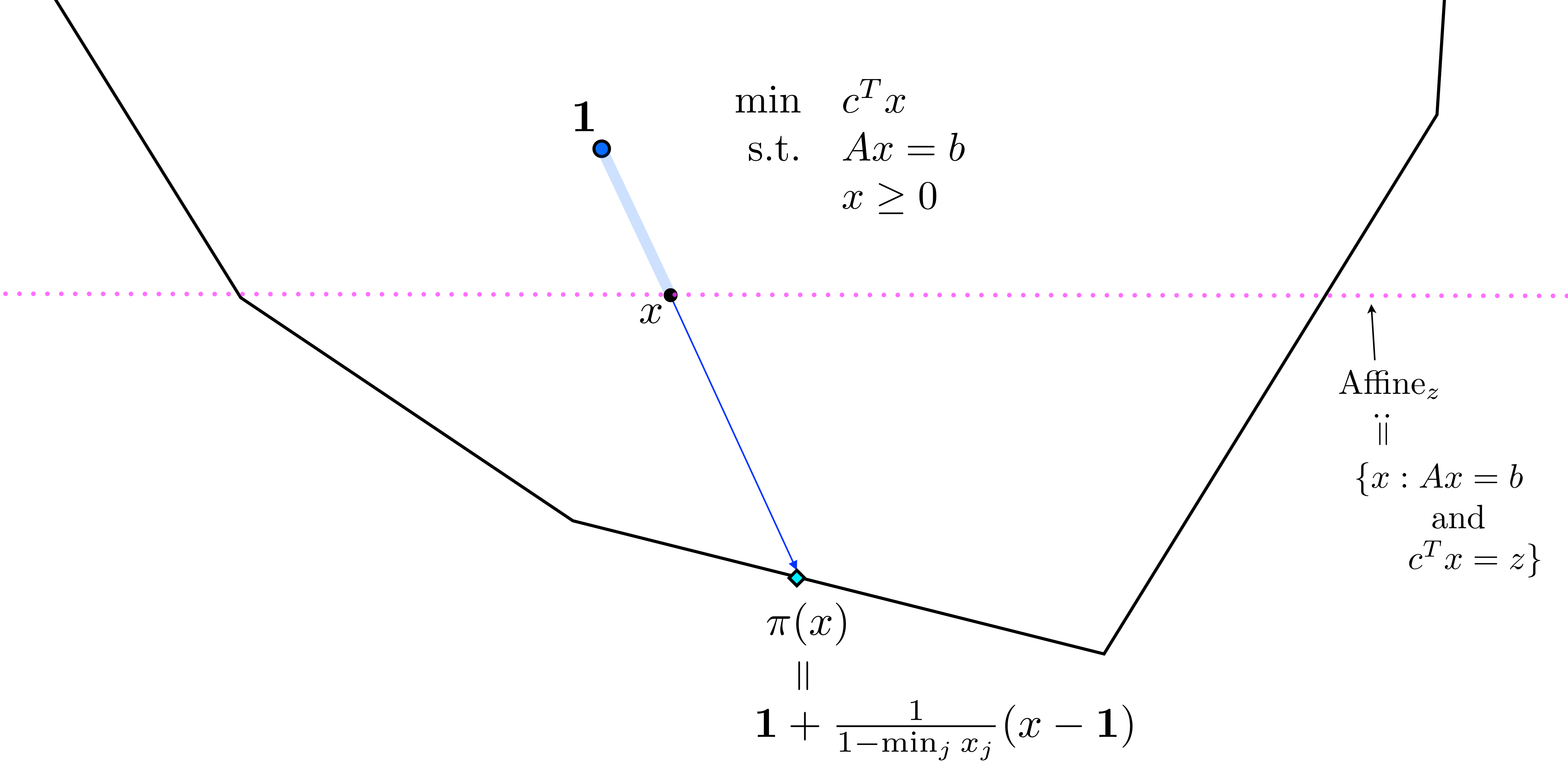
$$
\begin{array}{ll}
\min & c^T x \\
\text{s.t.} & Ax = b \\
 & x \geq 0
\end{array}
$$





$$\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Ax = b \\ & x \geq 0 \end{array}$$
$\mathbf{1}$
x_z^*
Affine_z
$\stackrel{\cdot\cdot}{\shortparallel}$
$\{x : Ax = b$
and
$c^T x = z\}$
$\pi^* = \pi(x_z^*)$


1
min $c^T x$
s.t. $Ax = b$
$x \geq 0$
Affine$_z$
$\{x : Ax = b$
and
$c^T x = z\}$

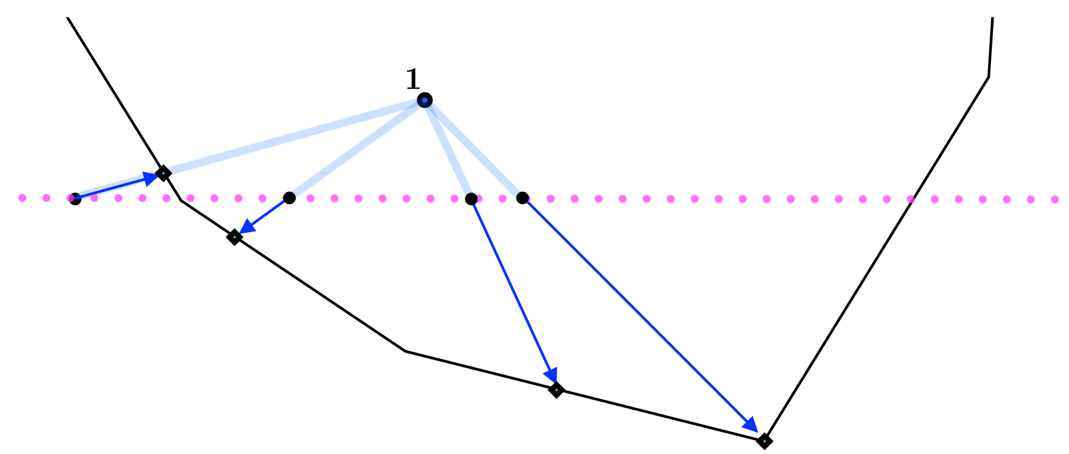


$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$
$\mathbf{1}$
x
$\pi(x)$
$\|$
$\mathbf{1} + \frac{1}{1 - \min_j x_j}(x - \mathbf{1})$
Affine_z
$\overset{\cdot\cdot}{\|}$
$\{x : Ax = b$
and
$c^T x = z\}$

$$\left.\begin{array}{ll} \min & c^T x \\ \text{s.t.} & Ax = b \\ & x \geq 0 \end{array}\right\} \text{LP}$$

z^*	optimal value, assumed finite
z	a fixed value satisfying $z < c^T \mathbf{1}$
Affine_z	$\{x : Ax = b \text{ and } c^T x = z\}$

$$x \mapsto \pi(x) := \mathbf{1} + \tfrac{1}{1-\min_j x_j}(x - \mathbf{1})$$



$$\begin{aligned} c^T \pi(x) &= c^T \mathbf{1} + \tfrac{1}{1-\min_j x_j}(c^T x - c^T \mathbf{1}) \\ &= c^T \mathbf{1} + \tfrac{1}{1-\min_j x_j} \underbrace{(\ z \ - c^T \mathbf{1})}_{\text{a negative constant}} \end{aligned}$$

Thus, for $x, y \in \text{Affine}_z$, $\quad c^T \pi(x) < c^T \pi(y) \quad \Leftrightarrow \quad \min_j x_j > \min_j y_j$

Theorem: LP is equivalent to

$$\begin{array}{ll} \max_x & \min_j x_j \\ \text{s.t.} & Ax = b \\ & c^T x = z \end{array}$$

The only constraints in the equivalent problem are linear equations.
It's thus easy to project onto the feasible region for the equivalent problem.

$$\left.\begin{array}{rl} \min & c^T x \\ \text{s.t.} & Ax = b \\ & x \geq 0 \end{array}\right\} \text{LP} \qquad \equiv \qquad \begin{array}{rl} \max_x & \min_j x_j \\ \text{s.t.} & Ax = b \\ & c^T x = z \end{array}$$

$x \mapsto \min_j x_j$ is the exemplary nonsmooth concave function

- Lipschitz continuous with constant $M = 1$
- Supgradients at x are the convex combinations of the standard basis vectors $e(k)$ for which $x_k = \min_j x_j$

Thus, projected supgradients at x are the convex combinations of the corresponding columns of the projection matrix

$$\bar{P} := I - \bar{A}^T (\bar{A}\,\bar{A}^T)^{-1} \bar{A} \quad \text{where } \bar{A} = \left[\begin{smallmatrix} A \\ c^T \end{smallmatrix}\right]$$

Hence, in implementing a supgradient method, one option in choosing a supgradient at x is simply to compute any column $\bar{P}_k$ for which $x_k = \min_j x_j$

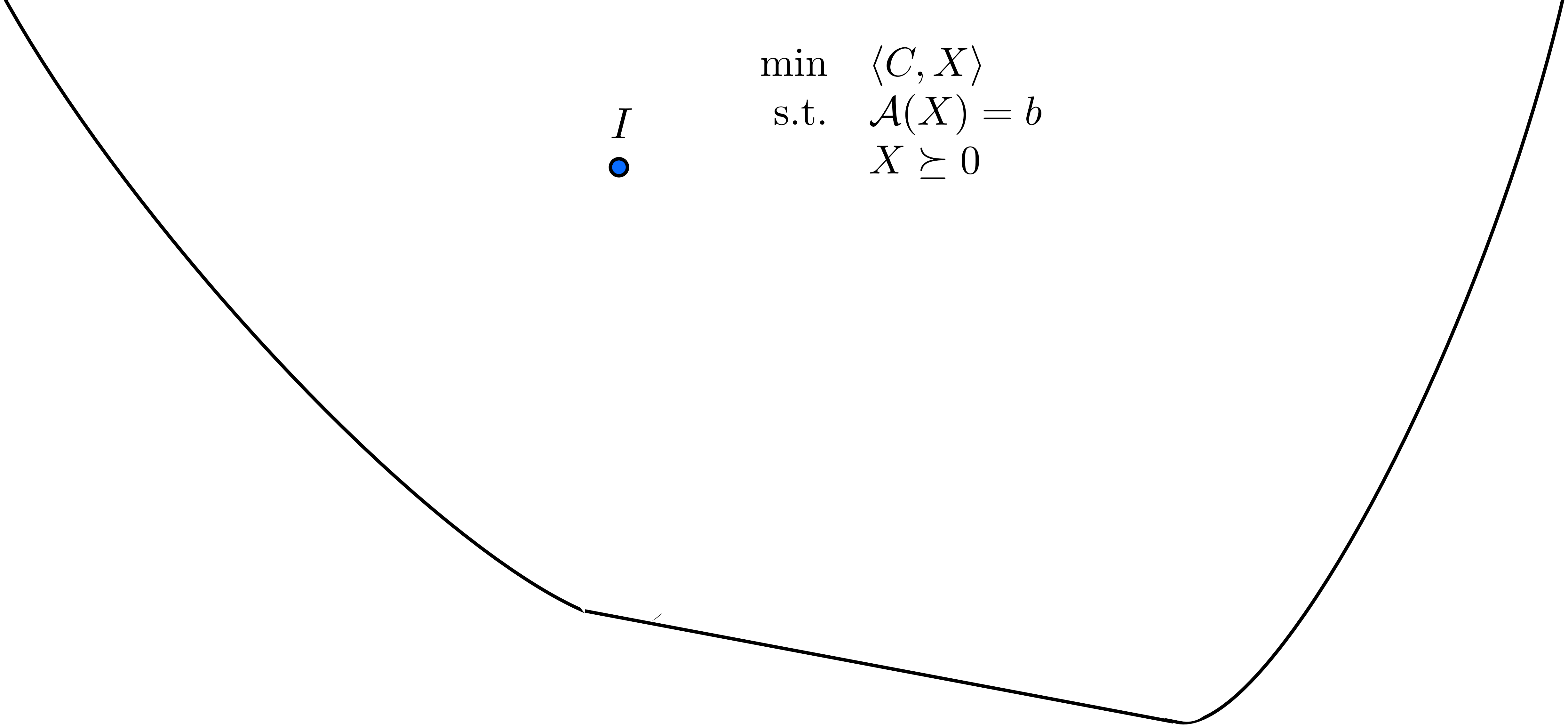
$$x_+ = x + \frac{\epsilon}{\|\bar{P}_k\|^2} \bar{P}_k$$

For large n, compute columns of $\bar{P}$ as needed

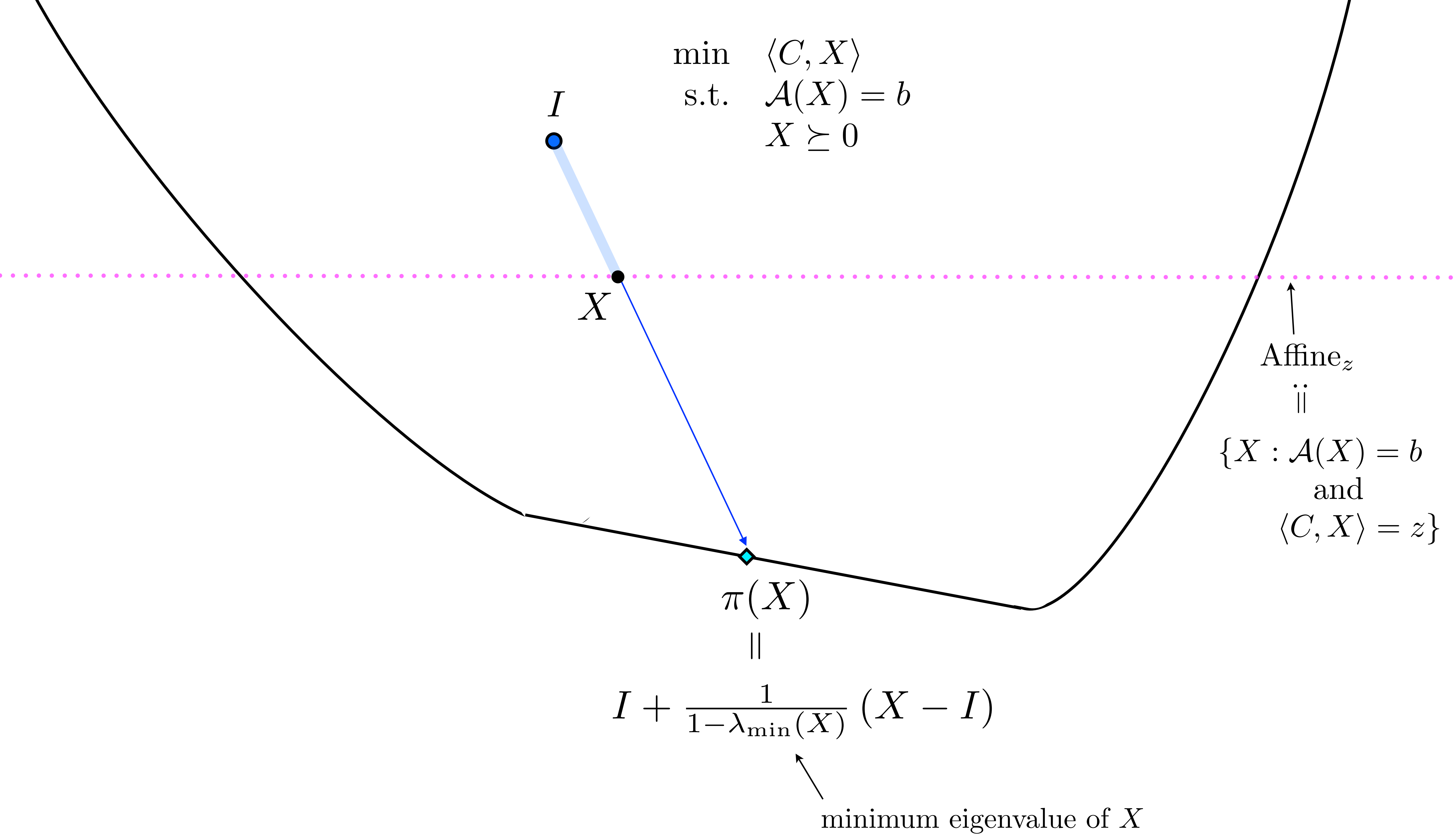
do not (cannot) compute (store in memory) all of $\bar{P}$

With a modest amount of preprocessing work, the cost of each iteration is proportional to the number of nonzero entries in A.

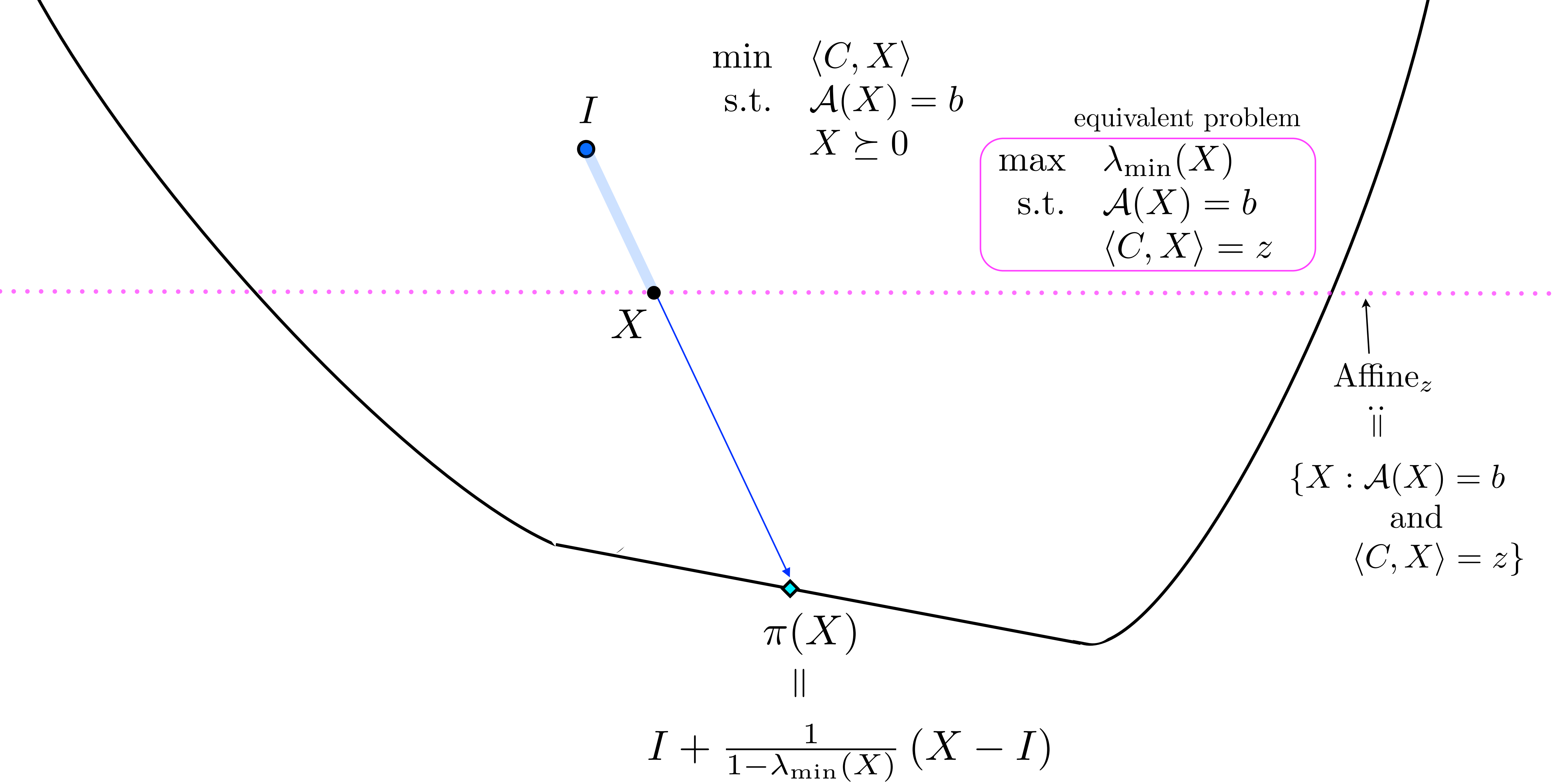
$$
\begin{array}{ll}
\min & \langle C, X \rangle \\
\text{s.t.} & \mathcal{A}(X) = b \\
& X \succeq 0
\end{array}
$$



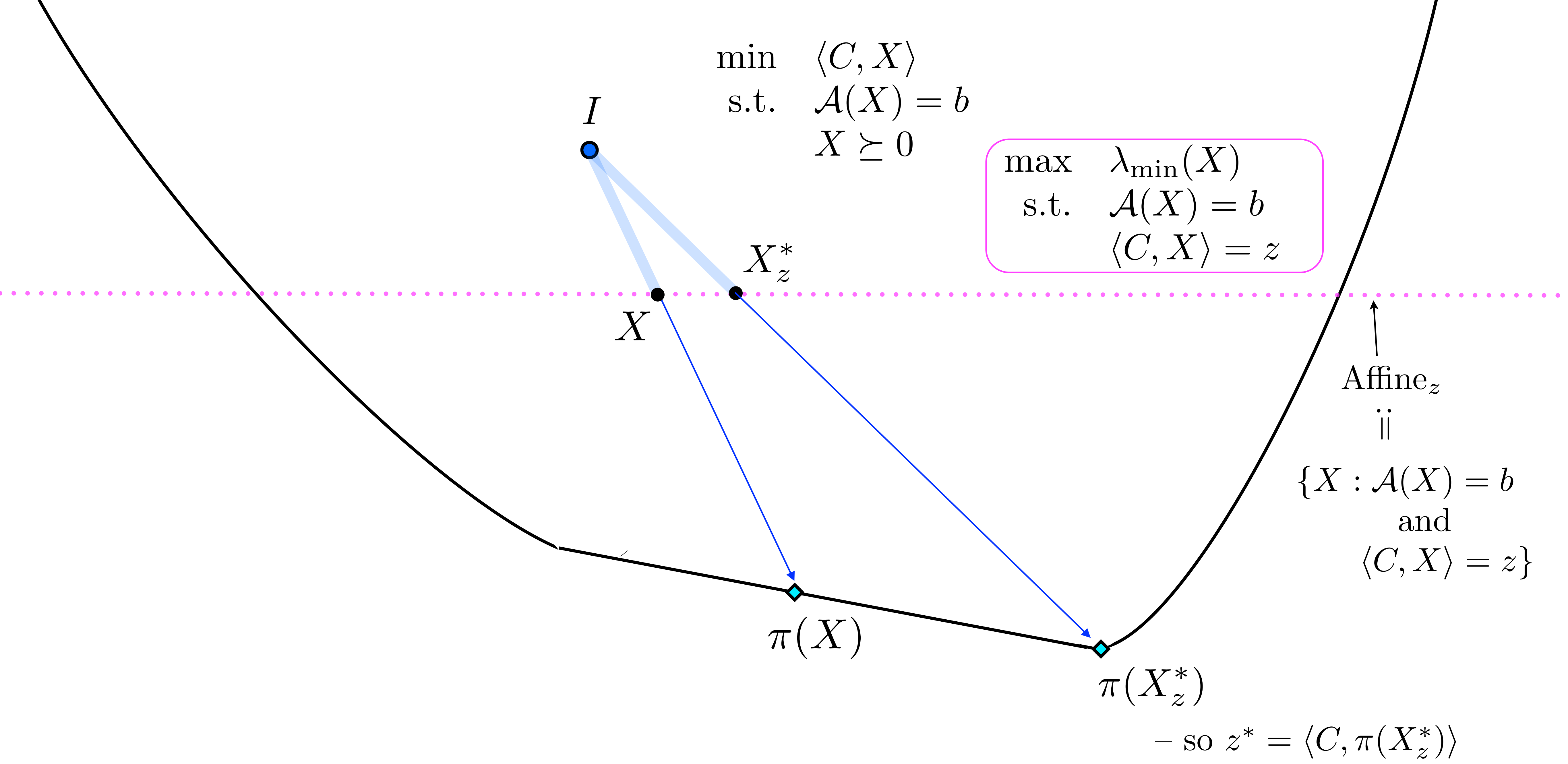
Now consider a semidefinite program,
and for simplicity, assume the identity matrix is feasible.



$\min \langle C, X\rangle$
$\text{s.t. } \mathcal{A}(X) = b$
$X \succeq 0$
I
X
Affine$_z$
$\vdots$
$\{X : \mathcal{A}(X) = b$
and
$\langle C, X\rangle = z\}$
$\pi(X)$
$\|$
$I + \frac{1}{1-\lambda_{\min}(X)}(X - I)$
minimum eigenvalue of X



$\min \quad \langle C, X\rangle$
$\text{s.t.} \quad \mathcal{A}(X) = b$
$X \succeq 0$
I
equivalent problem
$\max \quad \lambda_{\min}(X)$
$\text{s.t.} \quad \mathcal{A}(X) = b$
$\langle C, X\rangle = z$
X
Affine_z
$\{X : \mathcal{A}(X) = b$
and
$\langle C, X\rangle = z\}$
$\pi(X)$
$I + \frac{1}{1-\lambda_{\min}(X)}\,(X - I)$

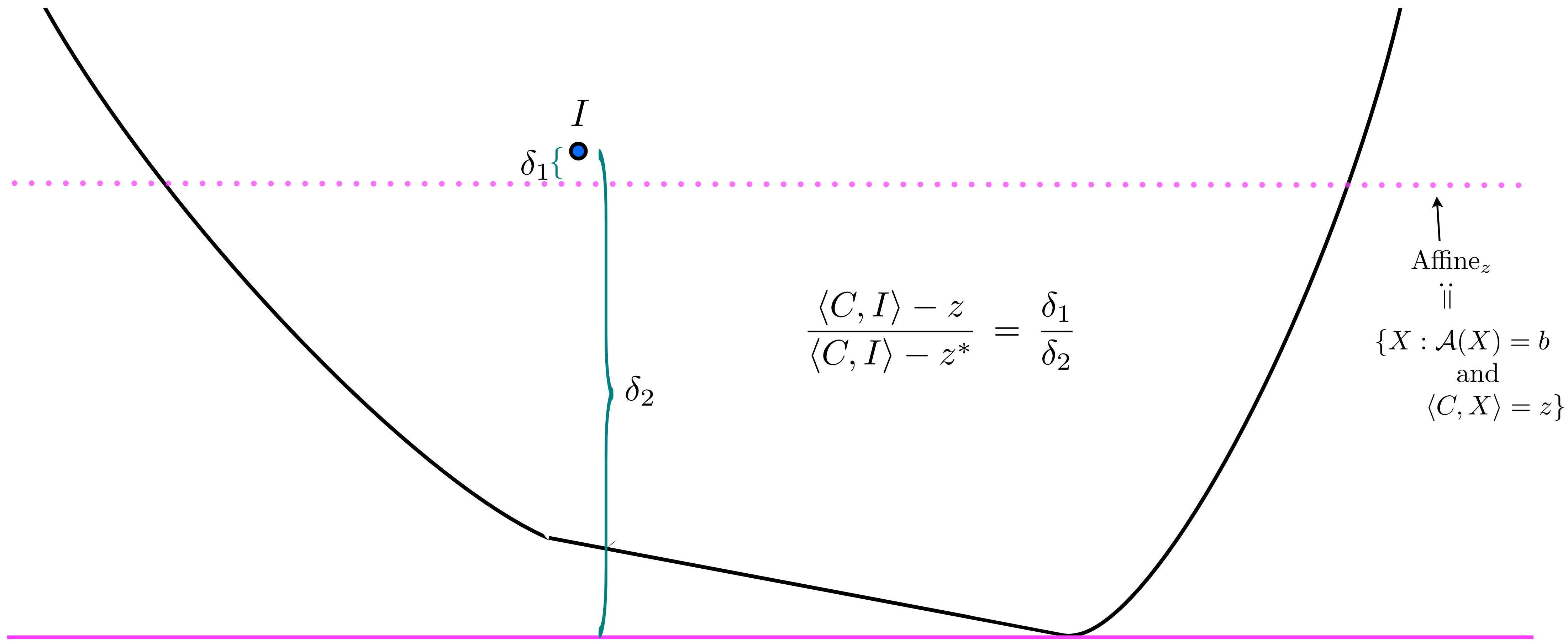


$$\textbf{Goal:}\ \text{Compute } X \text{ satisfying } \frac{\langle C, \pi(X)\rangle - z^*}{\langle C, I\rangle - z^*} \le \epsilon$$

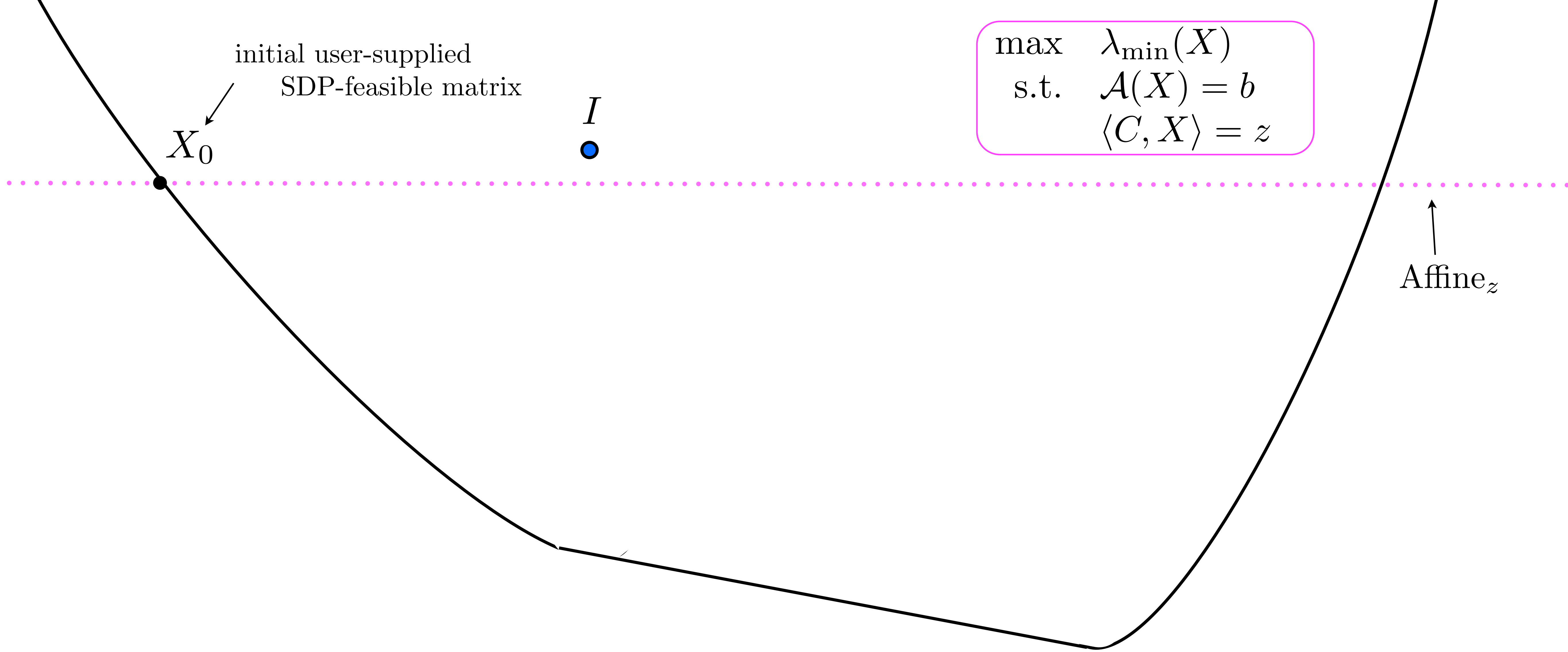
To accomplish this, how accurately does $\lambda_{\min}(X)$ *need to approximate* $\lambda_{\min}(X_z^*)$*?*

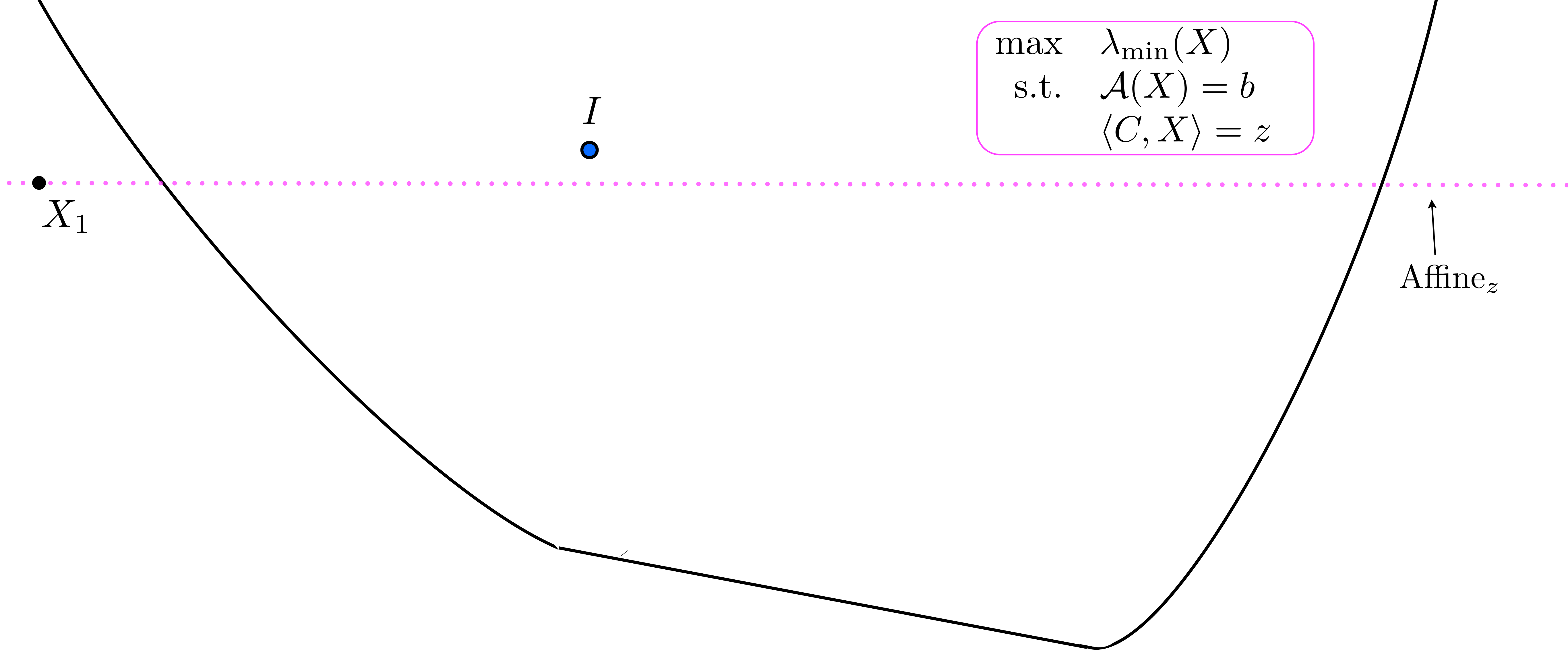
$$\left.\begin{array}{rl} \min & \langle C, X\rangle \\ \text{s.t.} & \mathcal{A}(X) = b \\ & X \succeq 0 \end{array}\right\} \text{SDP} \qquad\qquad \begin{array}{rl} \max & \lambda_{\min}(X) \\ \text{s.t.} & \mathcal{A}(X) = b \\ & \langle C, X\rangle = z \end{array}$$

$$\frac{\langle C, \pi(X)\rangle - z^*}{\langle C, I\rangle - z^*} \le \epsilon \quad \Leftrightarrow \quad \lambda_{\min}(X_z^*) - \lambda_{\min}(X) \le \frac{\epsilon}{1-\epsilon}\, \frac{\langle C, I\rangle - z}{\langle C, I\rangle - z^*}$$

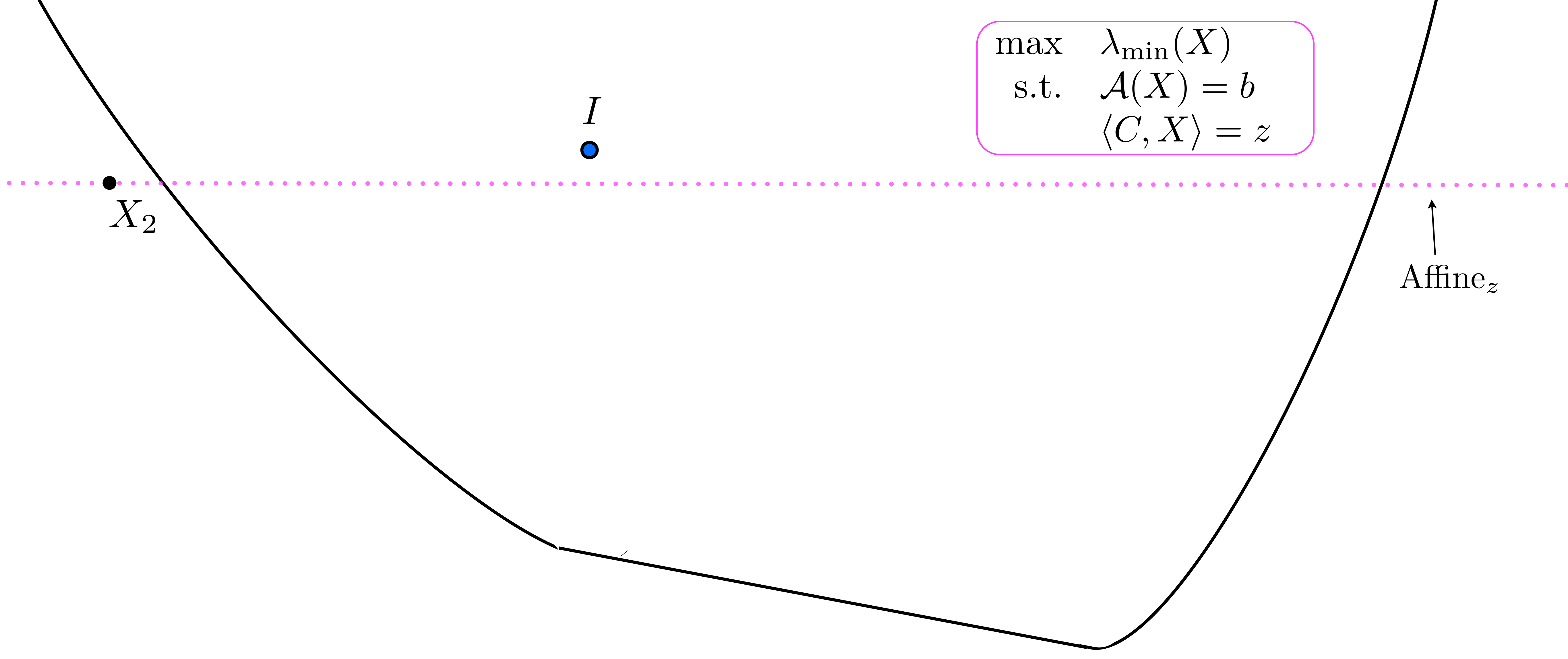


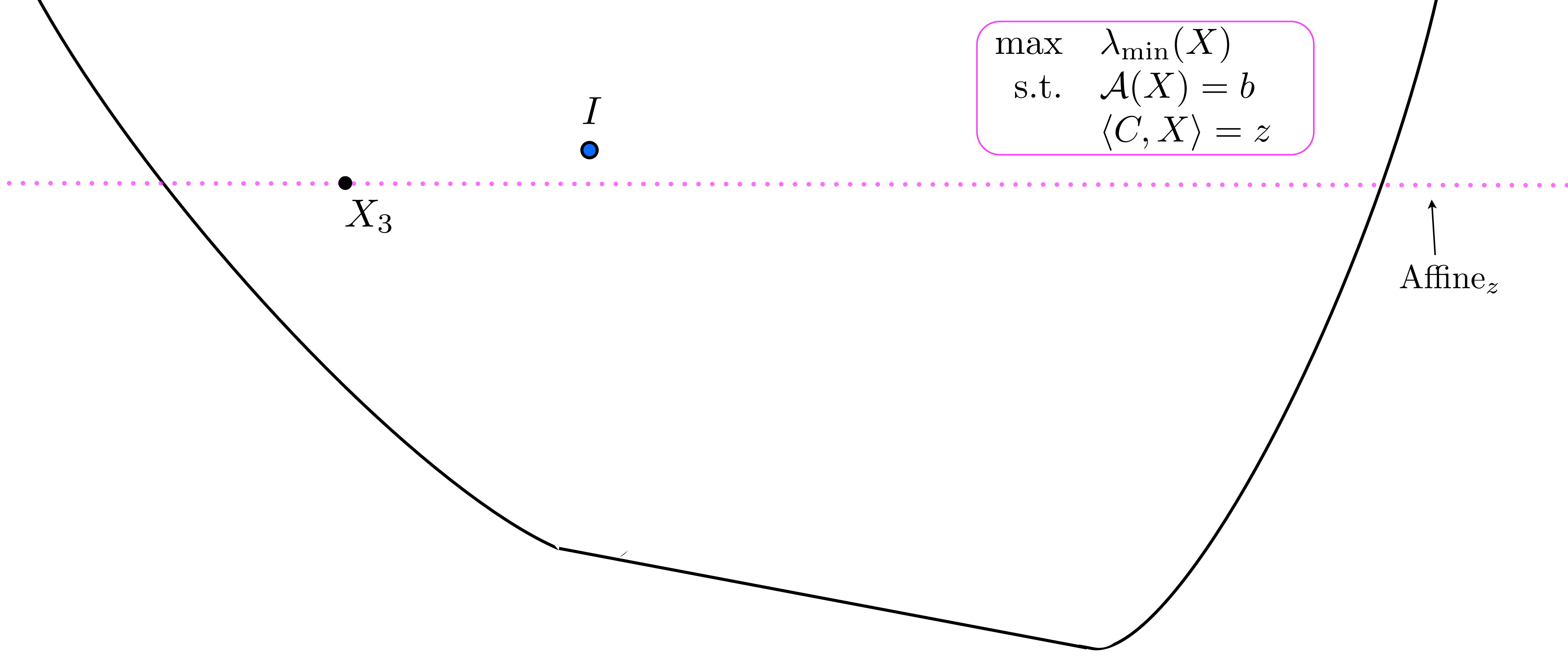
To get around the high accuracy required if the ratio is small,
and to get around having to know the ratio
(that is, having to know the optimal value z^*),
we apply a supgradient method to multiple layers
(details of which can be found in the arXiv posting) ...


initial user-supplied SDP-feasible matrix
X_0
I
max $\lambda_{\min}(X)$
s.t. $\mathcal{A}(X) = b$
$\langle C, X \rangle = z$
Affine_z

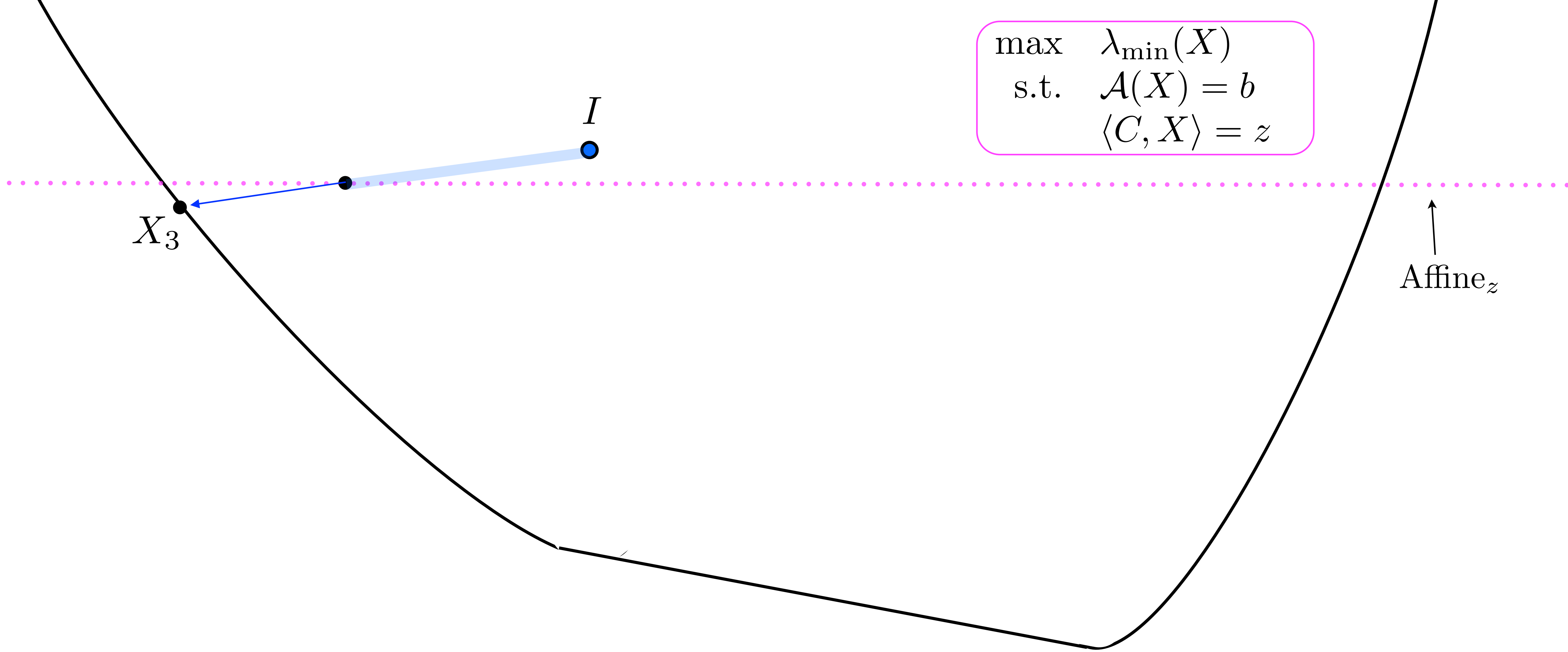


max $\lambda_{\min}(X)$
s.t. $\mathcal{A}(X) = b$
$\langle C, X \rangle = z$
I
X_1
Affine_z

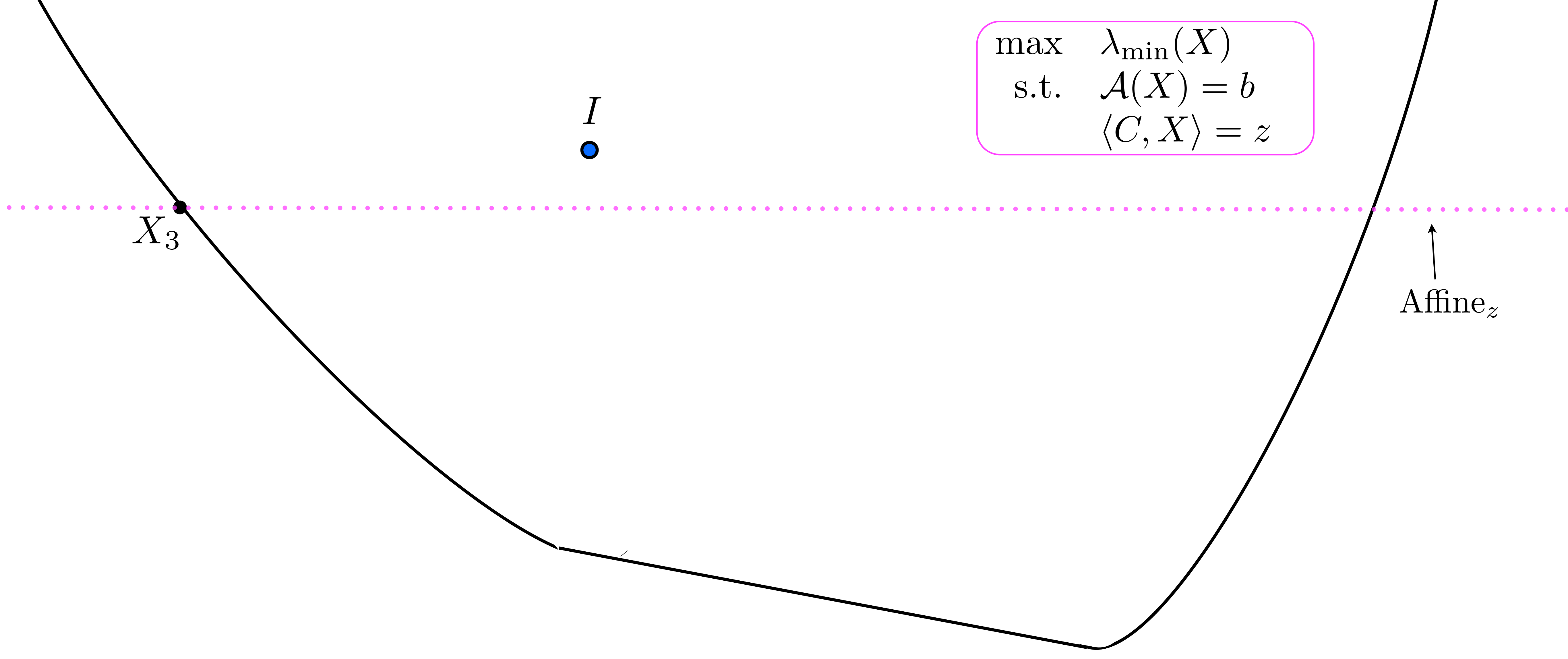

$$\begin{array}{ll} \max & \lambda_{\min}(X) \\ \text{s.t.} & \mathcal{A}(X) = b \\ & \langle C, X \rangle = z \end{array}$$
I
X_2
Affine_z

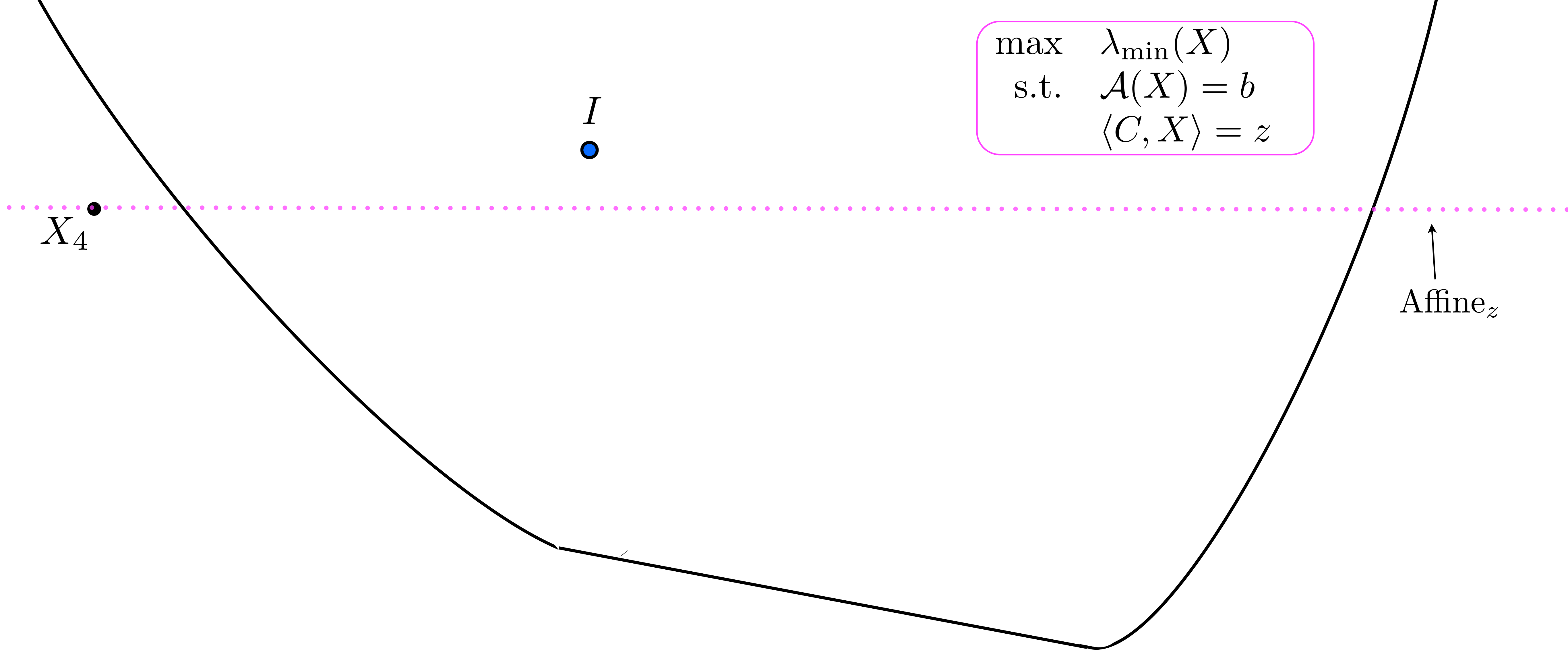


max $\lambda_{\min}(X)$
s.t. $\mathcal{A}(X) = b$
$\langle C, X\rangle = z$
I
X_3
Affine$_z$

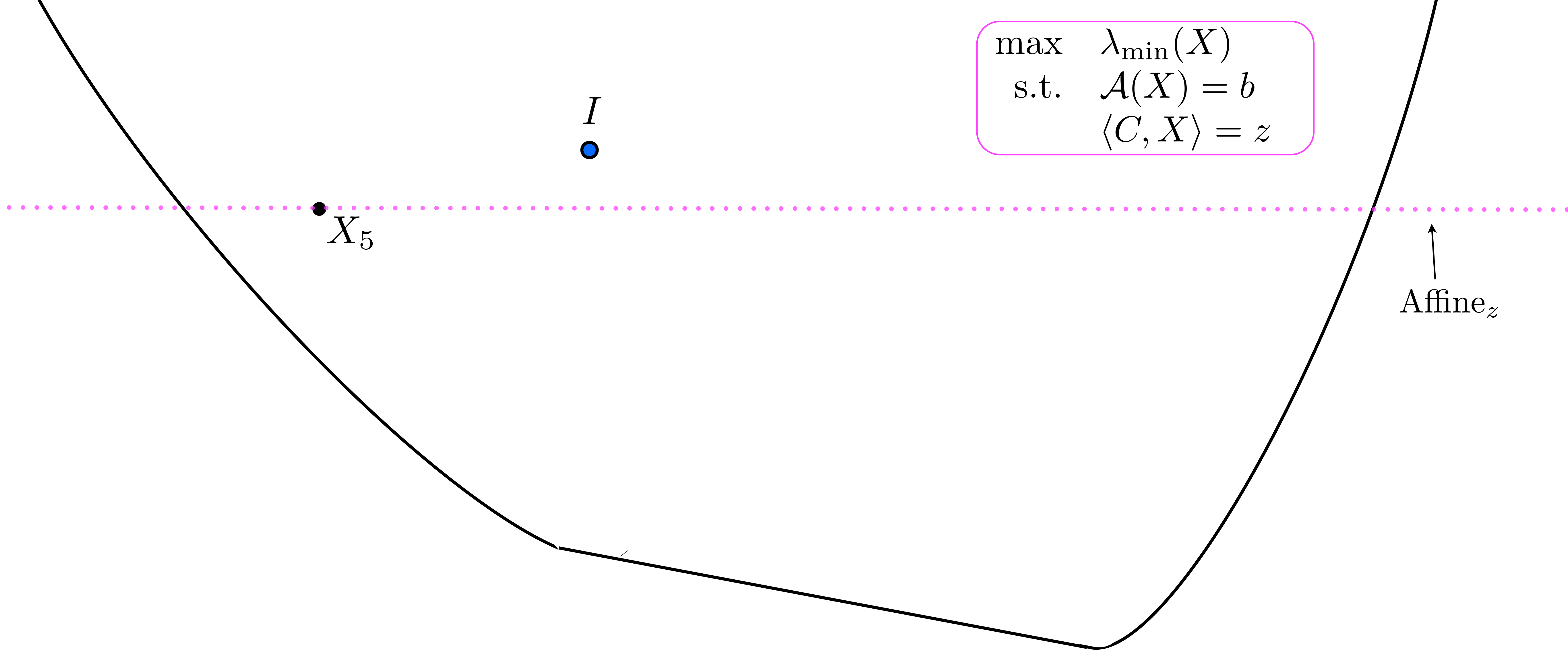


$\max \quad \lambda_{\min}(X)$
$\text{s.t.} \quad \mathcal{A}(X) = b$
$\langle C, X \rangle = z$
I
X_3
Affine_z

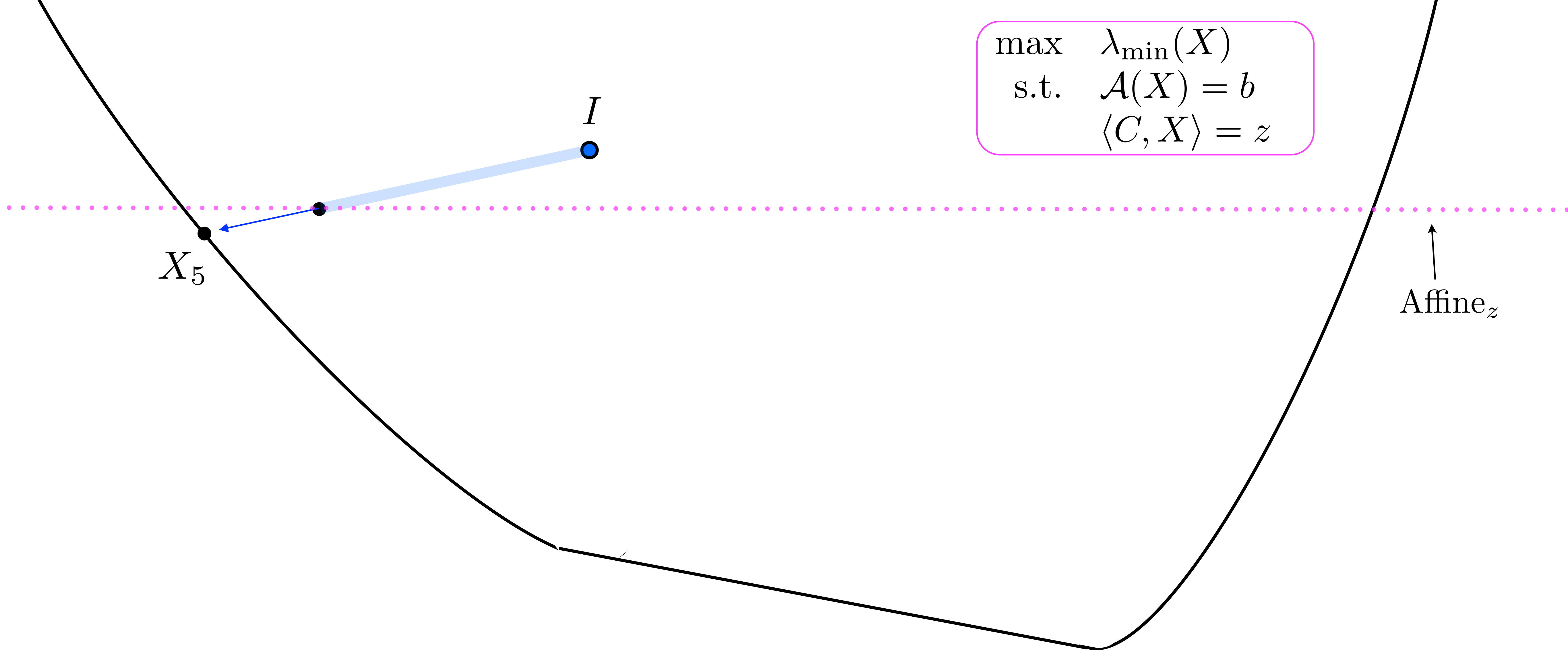

$\max \quad \lambda_{\min}(X)$
$\text{s.t.} \quad \mathcal{A}(X) = b$
$\langle C, X \rangle = z$
I
X_3
Affine_z

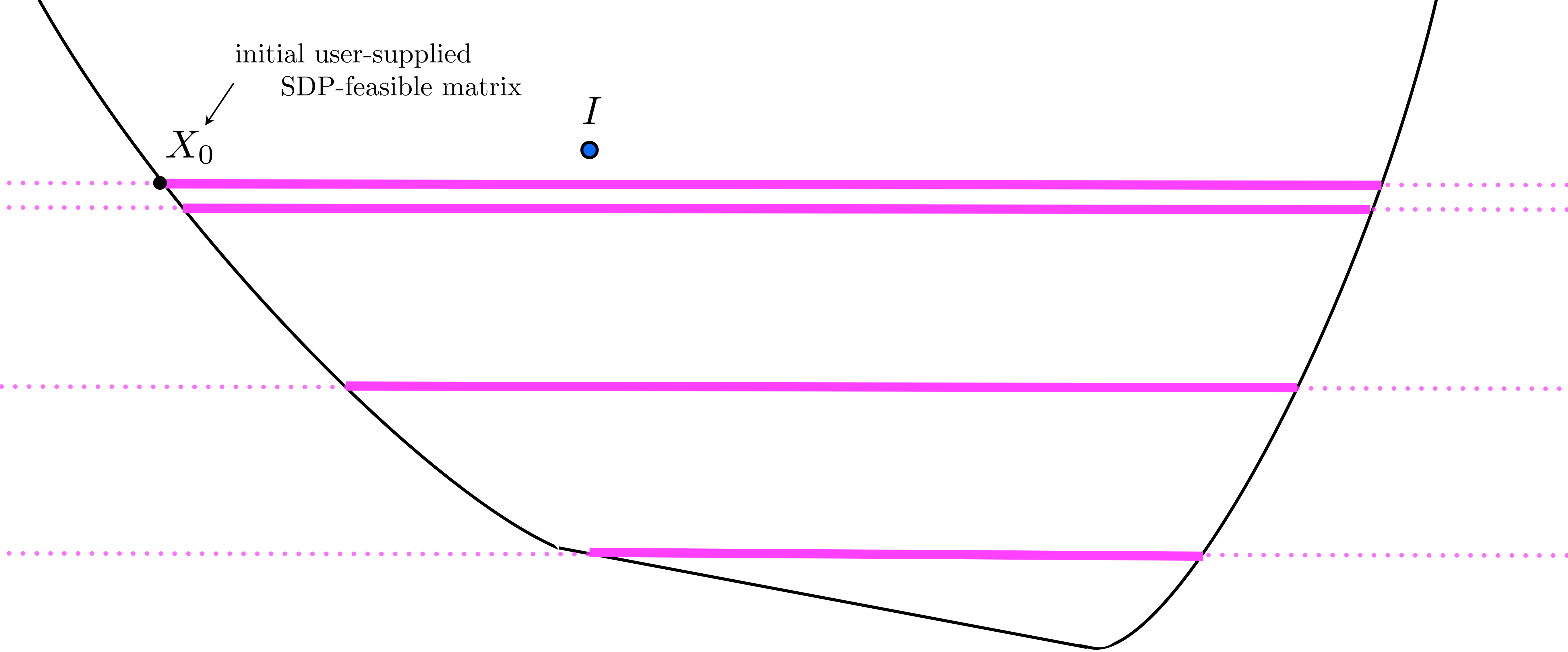


$\max \quad \lambda_{\min}(X)$
$\text{s.t.} \quad \mathcal{A}(X) = b$
$\langle C, X \rangle = z$
I
X_4
Affine_z



$$\begin{array}{ll} \max & \lambda_{\min}(X) \\ \text{s.t.} & \mathcal{A}(X) = b \\ & \langle C, X \rangle = z \end{array}$$
I
X_5
Affine_z


$\max \quad \lambda_{\min}(X)$
$\text{s.t.} \quad \mathcal{A}(X) = b$
$\langle C, X \rangle = z$
I
X_5
Affine_z



$=$ level set for SDP

$\mathrm{Diam} :=$ supremum of diameters of level sets for objective values $\leq \langle C, X_0 \rangle$

$$\left.\begin{array}{rl} \min & \langle C, X\rangle \\ \text{s.t.} & \mathcal{A}(X) = b \\ & X \succeq 0 \end{array}\right\} \text{SDP}$$	$$\begin{array}{rl} \max & \lambda_{\min}(X) \\ \text{s.t.} & \mathcal{A}(X) = b \\ & \langle C, X\rangle = z \end{array}$$
$$\frac{\langle C, \pi(X)\rangle - z^*}{\langle C, I\rangle - z^*} \leq \epsilon$$	$$\lambda_{\min}(X_z^*) - \lambda_{\min}(X) \leq \frac{\epsilon}{1-\epsilon} \frac{\langle C, I\rangle - z}{\langle C, I\rangle - z^*}$$

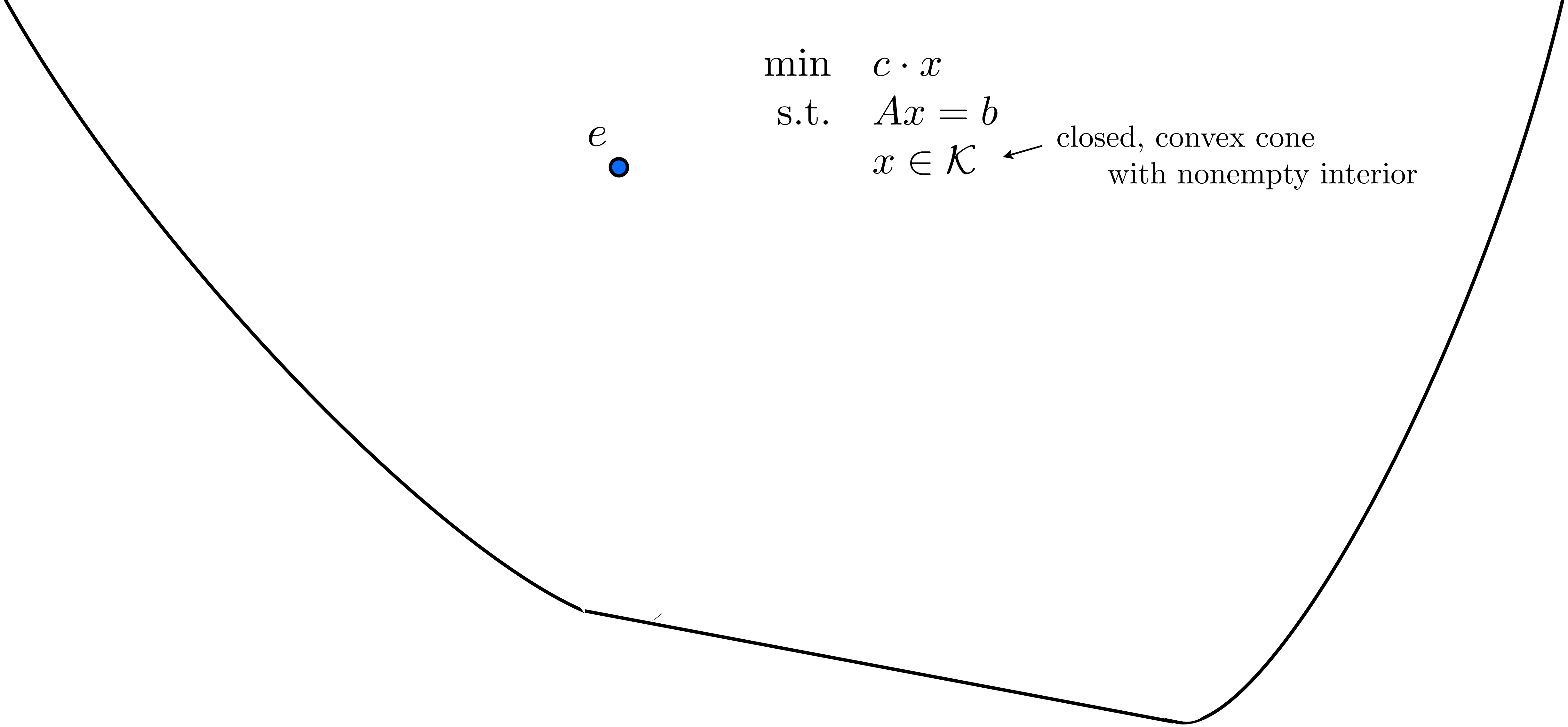
Thm:

$$\ell \geq 8 \text{ Diam}^2 \cdot \left(\frac{1}{\epsilon^2} + \frac{1}{\epsilon} \log_{4/3}\left(\frac{\langle C, I\rangle - z^*}{\langle C, I\rangle - \langle C, X_0\rangle} \right) + 1 \right)$$

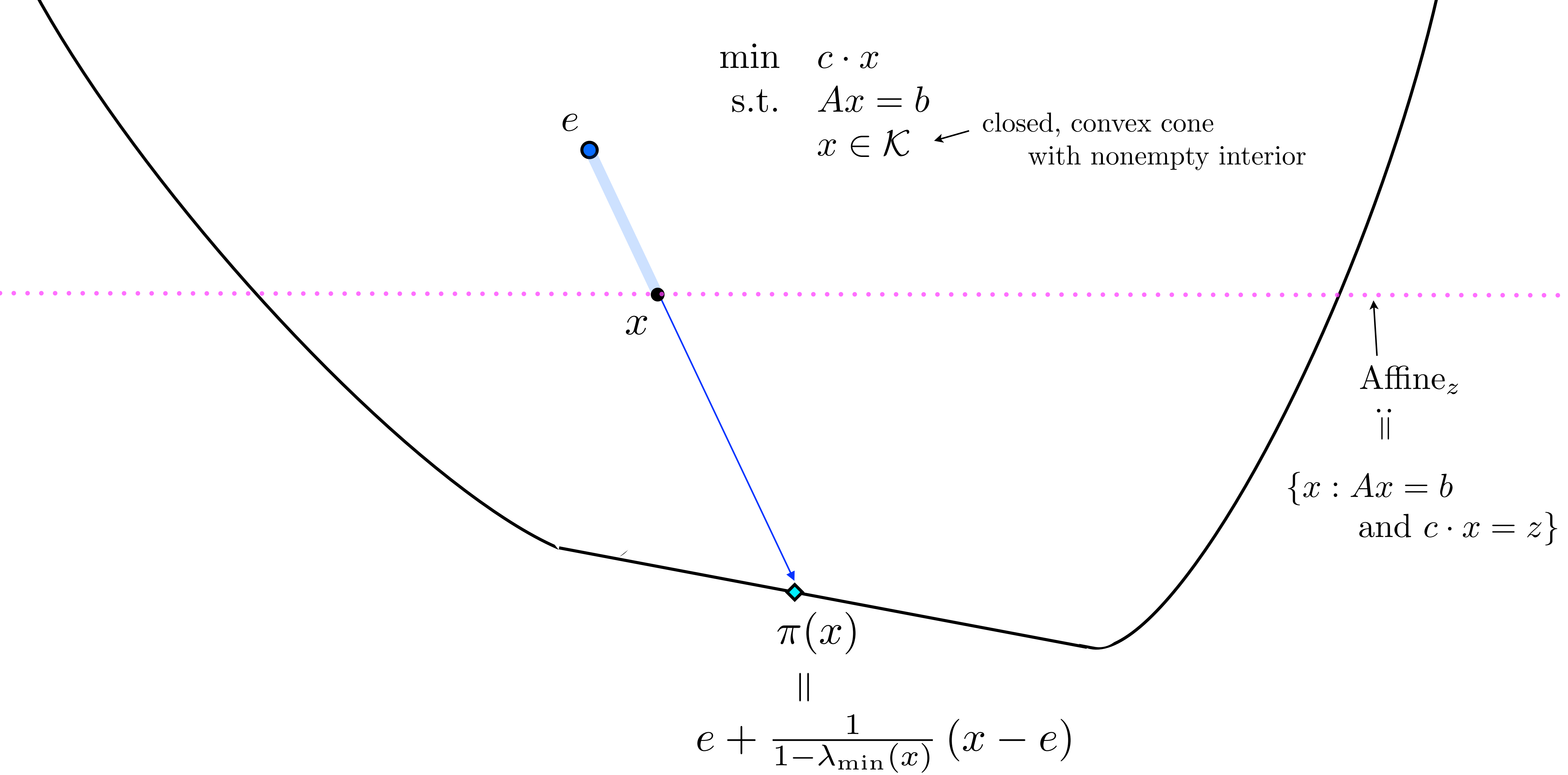
$$\Rightarrow \quad \min_{k \leq \ell} \frac{\langle C, \pi(X_k)\rangle - z^*}{\langle C, I\rangle - z^*} \leq \epsilon$$

$$\begin{aligned} \min \quad & c \cdot x \\ \text{s.t.} \quad & Ax = b \\ & x \in \mathcal{K} \end{aligned}$$

closed, convex cone with nonempty interior



Now consider a general convex conic optimization problem, and fix a strictly feasible point e.



$\ldots$ where $\lambda_{\min}(x)$ is the scalar λ satisfying $x - \lambda\, e \in \text{boundary}(\mathcal{K})$

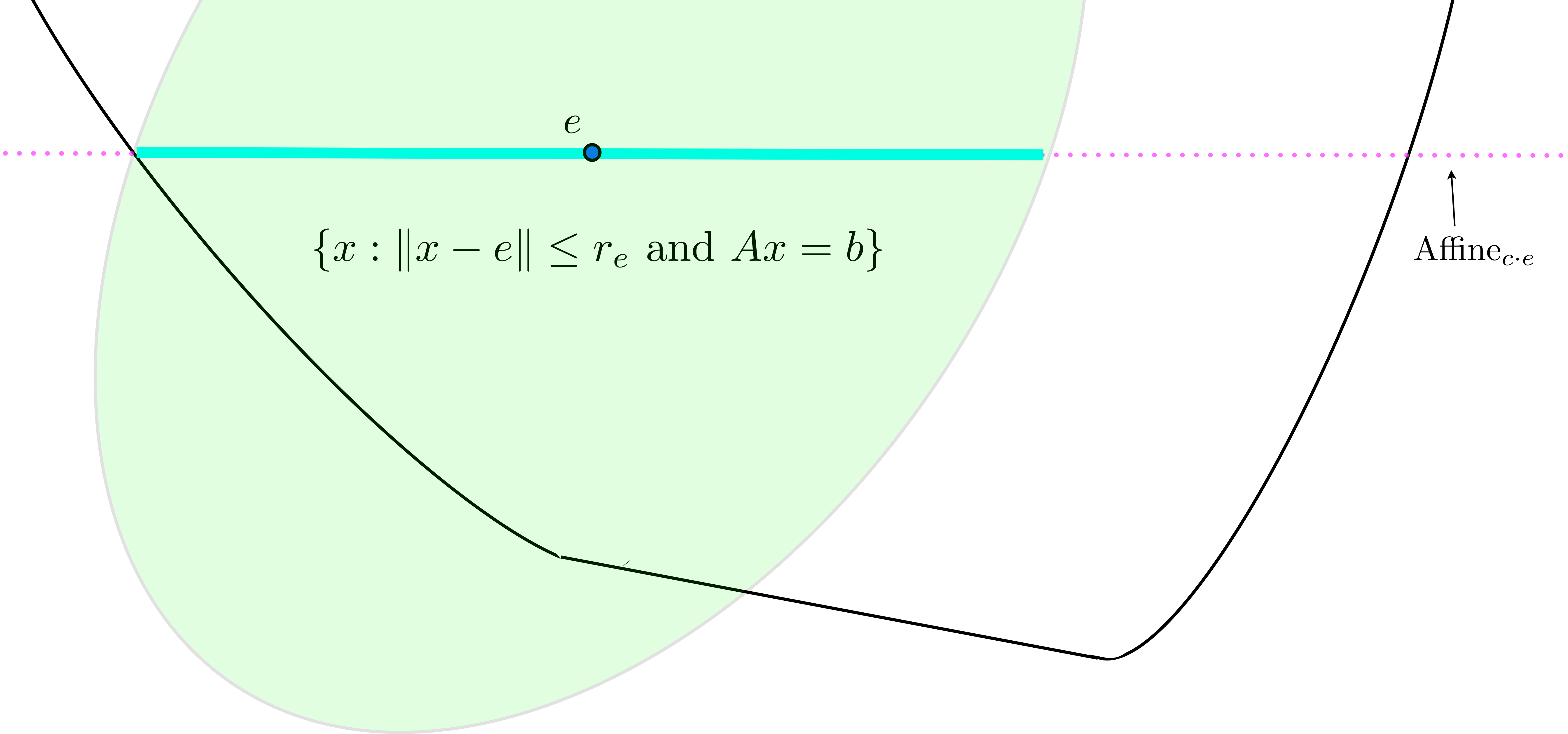
Prop: The map $x \mapsto \lambda_{\min}(x)$ is concave and Lipschitz continuous.

$$\begin{array}{ll} \min & c \cdot x \\ \text{s.t.} & Ax = b \\ & x \in \mathcal{K} \end{array}$$	$$\begin{array}{ll} \max & \lambda_{\min}(x) \\ \text{s.t.} & Ax = b \\ & c \cdot x = z \end{array}$$
$$\frac{c \cdot \pi(x) - z^*}{c \cdot e - z^*} \leq \epsilon \quad \Leftrightarrow$$	$$\lambda_{\min}(x_z^*) - \lambda_{\min}(x) \leq \frac{\epsilon}{1-\epsilon} \, \frac{c \cdot e - z}{c \cdot e - z^*}$$

even in this general setting
we have "if and only if"

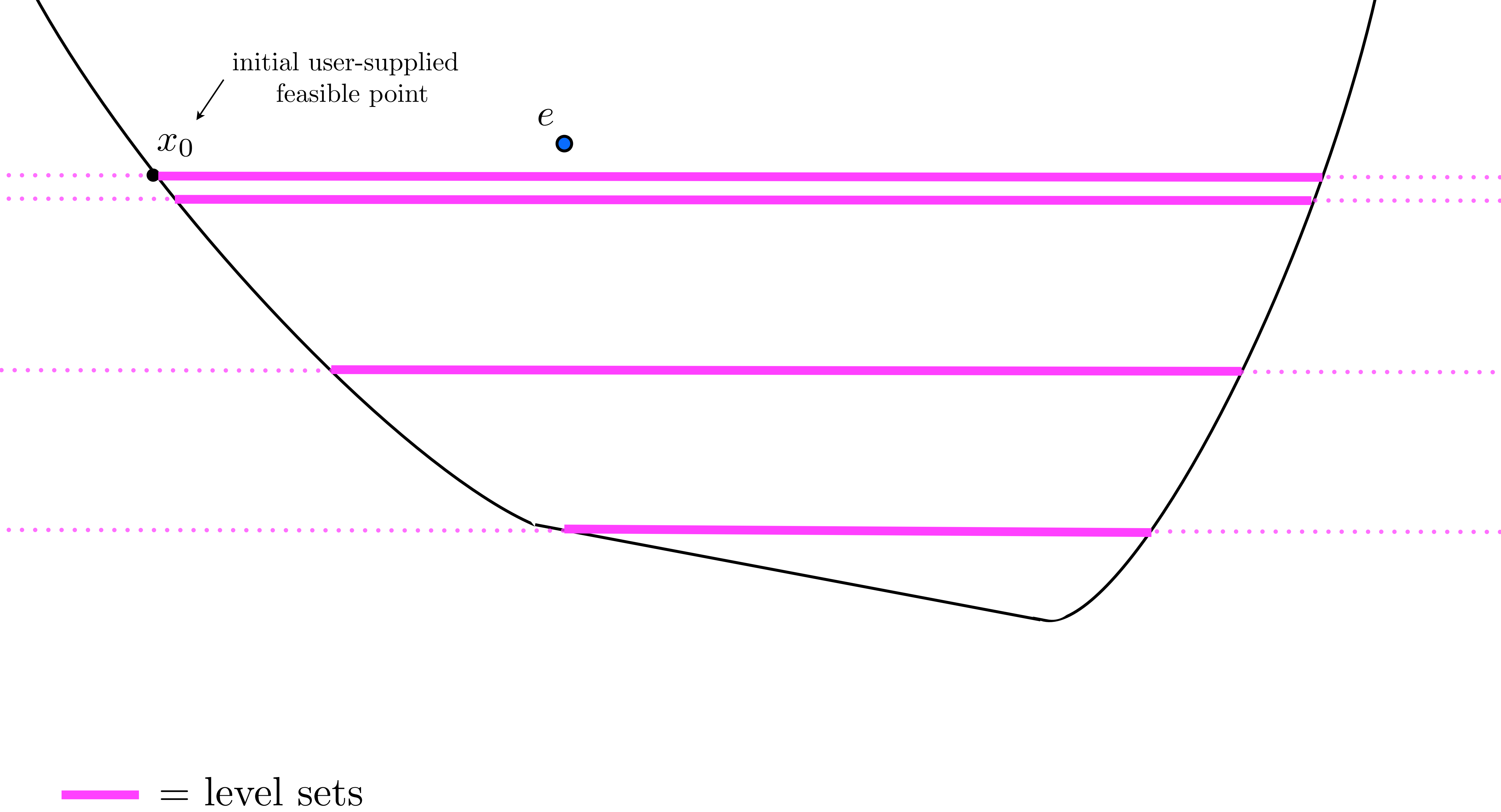
Whereas for linear programming we relied on the dot product,
and for SDP we relied on the trace product,
in this general setting
we allow computations to be done with respect to any inner product.

However, the extent to which the inner product
reflects the geometry of the cone $\mathcal{K}$ affects the Lipschitz constant ...



Prop: $|\lambda_{\min}(x) - \lambda_{\min}(y)| \leq \frac{1}{r_e} \|x - y\|$ for all $x, y \in \text{Affine}_z$ and for every z

(see arXiv posting for full explanation)



$$\text{Diam} := \text{ supremum of diameters of level sets for objective values } \leq c \cdot x_0$$

$$\begin{array}{rl} \min & c\cdot x \\ \text{s.t.} & Ax = b \\ & x \in \mathcal{K} \end{array}$$	$$\begin{array}{rl} \max & \lambda_{\min}(x) \\ \text{s.t.} & Ax = b \\ & c\cdot x = z \end{array}$$
$$\frac{c\cdot \pi(x) - z^*}{c\cdot e - z^*} \leq \epsilon$$	$$\lambda_{\min}(x_z^*) - \lambda_{\min}(x) \leq \frac{\epsilon}{1-\epsilon}\,\frac{c\cdot e - z}{c\cdot e - z^*}$$

Applying a supgradient method results in a sequence $x_0, x_1, \ldots$ for which ...

Thm:

Lipschitz constant $\leq 1/r_e$

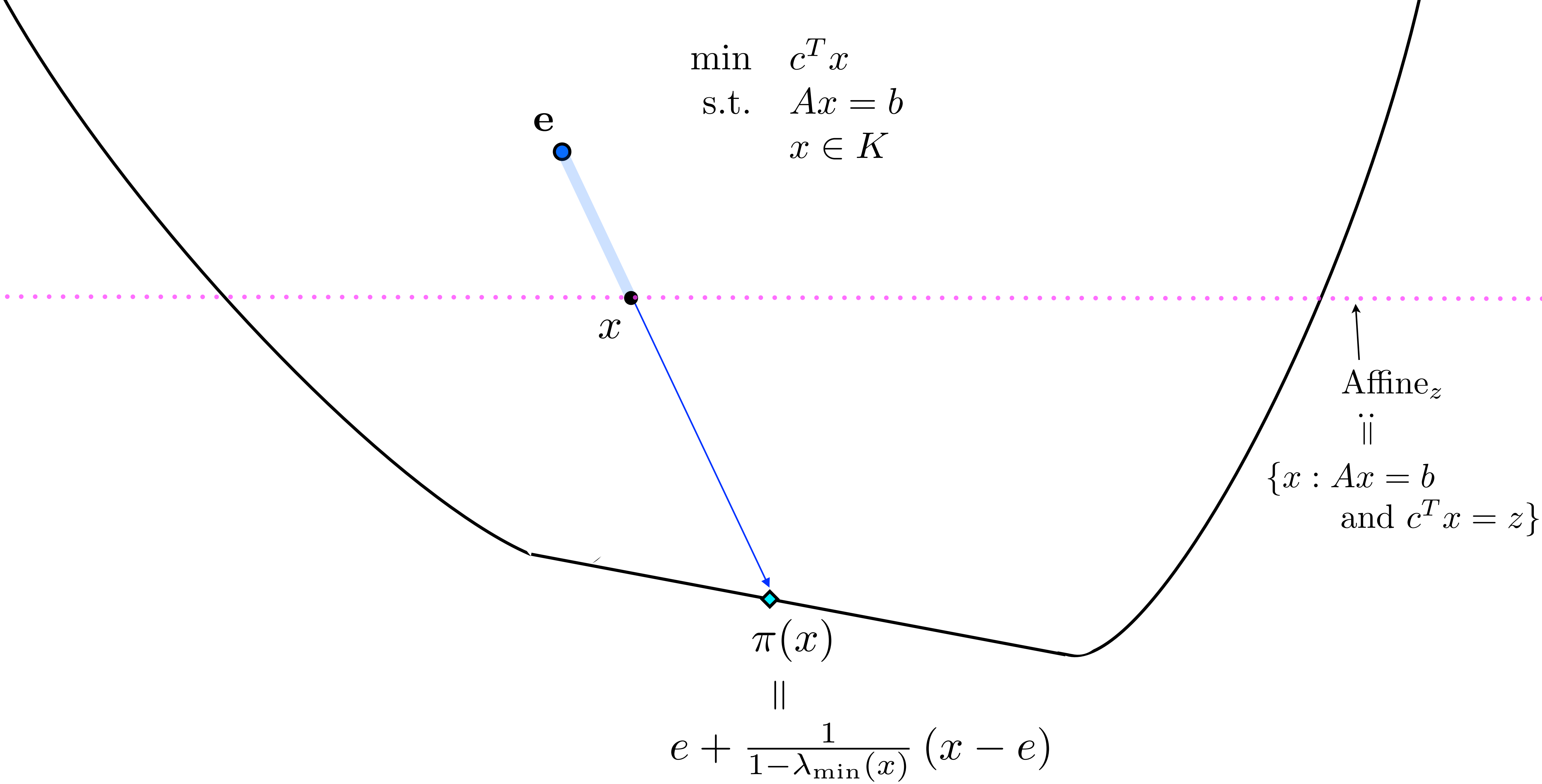
$$\ell \geq 8\,(M\,\mathrm{Diam})^2 \cdot \left(\frac{1}{\epsilon^2} + \frac{1}{\epsilon}\log_{4/3}\left(\frac{c\cdot e - z^*}{c\cdot e - c\cdot x_0}\right) + 1 \right)$$

$$\Rightarrow \quad \min_{k\leq \ell} \frac{c\cdot \pi(x_k) - z^*}{c\cdot e - z^*} \leq \epsilon$$

The main takeaway from the talk is the use of radial projection
to replace a general conic optimization problem
with an equivalent problem whose only constraints are linear equations.

This simple and natural approach
has not previously appeared in the literature, a blind spot.

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & Ax = b \\ & x \in K \end{aligned}$$



. . . where $\lambda_{\min}(x)$ is the scalar λ satisfying $x - \lambda e \in \text{boundary}(K)$

$$\begin{aligned} \min \quad & c \cdot x \\ \text{s.t.} \quad & Ax = b \\ & x \in \mathcal{K} \end{aligned} \quad \equiv \quad \begin{aligned} \max \quad & \lambda_{\min}(x) \\ \text{s.t.} \quad & Ax = b \\ & c \cdot x = z \end{aligned}$$

Thanks for listening!