

# CAQM: Convexity Analysis of Quadratic Maps

Anatoly Dymarsky, Elena Gryazina, Sergei Volodin, Boris Polyak

Name	Input	Call	Description	Return value	Exception
<b>Random map</b>	Dimensions $n, m$	<code>get_random_f(n, m, is_complex)</code>	Generates random map $f$	$[A, b]$	None
<b>Value at <math>x</math></b>	The point $x \in X$	<code>quadratic_map(A, b, x)</code>	Calculates $f(x)$	$y = f(x)$	None
<b>Product <math>c \cdot A</math></b>	Normal vector $c$	<code>get_Ac(A, c)</code>	Calculates $c \cdot A$	$A_c = c \cdot A$	None
<b>Get <math>H_c</math></b>	$c, y \in \mathbb{R}^m$	<code>get_H.c(A, b, c, y)</code>	Calculates $H_c = \begin{pmatrix} A_c & b_c \\ b'_c & -(c, y) \end{pmatrix}$	$H_c$	None
<b>Minimize <math>z(c)</math></b>	$c, c_+, \text{step } \beta$	<code>minimize_z.c(A, b, c, c_plus, beta_initial, max_step)</code>	Calculates $\inf_{c \in C_-} z(c)$	$[z, c\_array, z\_array]$	If failed
<b><math>\mathbb{R}^n</math> projection</b>		<code>project(A, b, c, x_0, delta_c, normal, search_area_size)</code>	Projects $c + \Delta c$ to $C_-$	$[c\_new, \text{lambda}]$	If failed
<b><math>\mathbb{C}^n</math> projection</b>		<code>project_descent(A, b, c, normal_1, normal_2)</code>	Projects $c$ to $C_-$	$[c\_new, \text{distance}]$	If failed
<b>Gradient <math>\frac{\partial z}{\partial c}</math></b>	Normal $c$	<code>get_dz.dc(A, b, c)</code>	Calculates $Q, \nabla z(c)$ , normal vectors $n_1, n_2$	$[Q, Q\_inv, k, v, \text{lambda\_min}, z, dz\_dc, \text{normal\_re}, \text{normal\_im}, \text{drho\_dc}]$	None
<b>Change of basis</b>	$c_+$	<code>change_basis(A, b, c_plus)</code>	Finds $\begin{cases} x = S(x' + x_0) \\ y = y' + y_0 \end{cases} \quad \text{s.t.} \quad \begin{cases} c_+ \cdot A_0 = I \\ c_+ \cdot b_0 = 0 \end{cases}$	$[A\_new, b\_new, x_0, y_0]$	None
<b>Boundary point <math>\partial F_c</math></b>	Normal $c$	<code>boundary_point(A, b, c)</code>	Calculates pre-image $x: f(x) \in \partial F_c$	$[x]$	If $c \cdot A < 0$