SUSPECT: MINLP Special Structure Detector for Pyomo
Supplementary Material

Francesco Ceccon • John D. Siirola • Ruth Misener

A Rules of Interval Arithmetic

Given the intervals $[x_1^L, x_1^U]$ and $[x_2^L, x_2^U]$, the following rules apply for finite bounds:

1. $[x_1^L, x_1^U] + [x_2^L, x_2^U] = [x_1^L + x_2^L, x_1^U + x_2^U]$
2. $[x_1^L, x_1^U] - [x_2^L, x_2^U] = [x_1^L - x_2^U, x_1^U - x_2^L]$
3. $[x_1^L, x_1^U] \times [x_2^L, x_2^U] = [\min(x_1^L x_2^L, x_1^L x_2^U, x_1^U x_2^L, x_1^U x_2^U), \max(x_1^L x_2^L, x_1^L x_2^U, x_1^U x_2^L, x_1^U x_2^U)]$
4. $[x_1^L, x_1^U] \div [x_2^L, x_2^U] = [x_1^L, x_1^U] \times [(x_2^U)^{-1}, (x_2^L)^{-1}]$ if $0 \notin [x_2^L, x_2^U]$, $[-\infty, \infty]$ otherwise

Tables 2–5 list the more complex arithmetic rules that SUSPECT uses when a bound is not finite [33].

B Bound Tightening of Summation Expressions

Consider expression $g(\boldsymbol{x})$ with range $[g^L, g^U]$ that sums functions $h_i(\boldsymbol{x})$, $i = 1, \dots, N$ with range $[h_i^L, h_i^U]$.

$$g(\boldsymbol{x}) = h_1(\boldsymbol{x}) + h_2(\boldsymbol{x}) + \cdots + h_N(\boldsymbol{x}) = \sum_{i=1}^{N} h_i(\boldsymbol{x}) = \sum_{i=1, i\neq j}^{N} h_i(\boldsymbol{x}) + h_j(\boldsymbol{x}).$$

It follows that, for all $h_i(\boldsymbol{x})$ appearing in the summation $g(\boldsymbol{x})$, the bounds of $h_j(\boldsymbol{x})$ are:

$$g^L - \sum_{i=1, i\neq j}^{N} h_i^U \leq h_j(\boldsymbol{x}) \leq g^U - \sum_{i=1, i\neq j}^{N} h_i^L.$$

Proceeding similarly for linear expressions:

$$g(\boldsymbol{x}) = \sum_{i=1}^{N} \alpha_i x_i,$$

the new bounds of x_j are:

$$g^L - \sum_{i=1, i\neq j, \alpha_i>0}^{N} \alpha_i x_i^U - \sum_{i=1, i\neq j, \alpha_i<0} \alpha_i x_i^L \leq \alpha_j x_j \leq g^U - \sum_{i=1, i\neq j, \alpha_i>0}^{N} \alpha_i x_i^L - \sum_{i=1, i\neq j, \alpha_i<0}^{N} \alpha_i x_i^U$$

C Special Structure Detection

This section completes the monotonicity and convexity rules discussed in Section 4. Let k be the function defined as the composition $k = g \circ h : X \to Y$ of the functions $g : Z \to Y$ and $h : X \to Z$, where $X \subseteq \mathbb{R}^n$, $Y \subseteq \mathbb{R}$, $Z \subseteq \mathbb{R}^l$, and $\mathbf{dom}\, k = \{\boldsymbol{x} \in \mathbf{dom}\, h | h(\boldsymbol{x}) \in \mathbf{dom}\, g\}$. The function image h is positive if $h(\boldsymbol{x}) > 0$, $\forall \boldsymbol{x} \in X$, nonnegative if $h(\boldsymbol{x}) \geq 0$, $\forall \boldsymbol{x} \in X$, negative if $h(\boldsymbol{x}) < 0$, $\forall \boldsymbol{x} \in X$, and nonpositive if $h(\boldsymbol{x}) \leq 0$, $\forall \boldsymbol{x} \in X$.

The Tables 6–11 containing the special structure rules are read as Boolean logic tables, if we name the property (either monotonicity or convexity) of $k = g \circ h$ as C, and the conditions on the convexity, monotonicity and image of g and h as A_i, $i = 1 \dots n$, then each row of the table is equivalent to the implication $\bigwedge_{i=1}^{n} A_i \Rightarrow C$.

To understand the monotonicity rules [9] of $k(\boldsymbol{x})$ we compute the gradient of k

$$\nabla k(\boldsymbol{x}) = \nabla g(h(\boldsymbol{x})) \nabla h(\boldsymbol{x})$$

and it follows that

Table 2 Interval arithmetic rules for $[x_1^L, x_1^U] + [x_2^L, x_2^U]$ and $[x_1^L, x_1^U] - [x_2^L, x_2^U]$ [33].

Addition	$[-\infty, x_2^U]$	$[x_2^L, x_2^U]$	$[x_2^L, \infty]$	$[-\infty, \infty]$
$[-\infty, x_1^U]$	$[-\infty, x_1^U + x_2^U]$	$[-\infty, x_1^U + x_2^U]$	$[-\infty, \infty]$	$[-\infty, \infty]$
$[x_1^L, x_1^U]$	$[-\infty, x_1^U + x_2^U]$	$[x_1^L + x_2^L, x_1^U + x_2^U]$	$[x_1^L + x_2^L, \infty]$	$[-\infty, \infty]$
$[x_1^L, \infty]$	$[-\infty, \infty]$	$[x_1^L + x_2^L, \infty]$	$[x_1^L + x_2^L, \infty]$	$[-\infty, \infty]$
$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$
Subtraction	$[-\infty, x_2^U]$	$[x_2^L, x_2^U]$	$[x_2^L, \infty]$	$[-\infty, \infty]$
$[-\infty, x_1^U]$	$[-\infty, \infty]$	$[-\infty, x_1^U - x_2^L]$	$[-\infty, x_1^U - x_2^L]$	$[-\infty, \infty]$
$[x_1^L, x_1^U]$	$[x_1^L - x_2^U, \infty]$	$[x_1^L - x_2^U, x_1^U - x_2^L]$	$[-\infty, x_1^U - x_2^L]$	$[-\infty, \infty]$
$[x_1^L, \infty]$	$[x_1^L - x_2^U, \infty]$	$[x_1^L - x_2^U, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$
$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$

Table 3 Interval arithmetic rules for $[x_1^L, x_1^U] \times [x_2^L, x_2^U]$ [33].

Multiplication	$[x_2^L, x_2^U]$ $x_2^U \leq 0$	$[x_2^L, x_2^U]$ $x_2^L < 0 < x_2^U$	$[x_2^L, x_2^U]$ $x_2^L \geq 0$	$[0,0]$	$[-\infty, x_2^U]$ $x_2^U \leq 0$	$[-\infty, x_2^U]$ $x_2^U \geq 0$	$[x_2^L, \infty]$ $x_2^L \leq 0$	$[x_2^L, \infty]$ $x_2^U \geq 0$	$[-\infty, \infty]$
$[x_1^L, x_1^U], x_2^U \leq 0$	$[x_1^U \cdot x_2^U, x_1^L \cdot x_2^L]$	$[x_1^L \cdot x_2^U, x_1^L \cdot x_2^L]$	$[x_1^L \cdot x_2^U, x_1^U \cdot x_2^L]$	$[0,0]$	$[x_1^U \cdot x_2^U, \infty]$	$[x_1^L \cdot x_2^U, \infty]$	$[-\infty, x_1^L \cdot x_2^L]$	$[-\infty, x_1^U \cdot x_2^L]$	$[-\infty, \infty]$
$[x_1^L, x_1^U], x_1^L < 0 < x_1^U$	$[x_1^U \cdot x_2^L, x_1^L \cdot x_2^L]$	$[\min(x_1^L \cdot x_2^U, x_1^U \cdot x_2^L), \max(x_1^L \cdot x_2^L, x_1^U \cdot x_2^U)]$	$[x_1^L \cdot x_2^U, x_1^U \cdot x_2^U]$	$[0,0]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$
$[x_1^L, x_1^U], x_1^L \geq 0$	$[x_1^U \cdot x_2^L, x_1^L \cdot x_2^U]$	$[x_1^U \cdot x_2^L, x_1^U \cdot x_2^U]$	$[x_1^L \cdot x_2^L, x_1^U \cdot x_2^U]$	$[0,0]$	$[-\infty, x_1^L \cdot x_2^U]$	$[-\infty, x_1^U \cdot x_2^U]$	$[x_1^U \cdot x_2^L, \infty]$	$[x_1^L \cdot x_2^L, \infty]$	$[-\infty, \infty]$
$[0,0]$	$[0,0]$	$[0,0]$	$[0,0]$	$[0,0]$	$[0,0]$	$[0,0]$	$[0,0]$	$[0,0]$	$[0,0]$
$[-\infty, x_1^U], x_1^U \leq 0$	$[x_1^U \cdot x_2^U, \infty]$	$[-\infty, \infty]$	$[-\infty, x_1^U \cdot x_2^L]$	$[0,0]$	$[x_1^U \cdot x_2^U, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, x_1^U \cdot x_2^L]$	$[-\infty, \infty]$
$[-\infty, x_1^U], x_1^U \geq 0$	$[x_1^U \cdot x_2^L, \infty]$	$[-\infty, \infty]$	$[-\infty, x_1^U \cdot x_2^U]$	$[0,0]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$
$[x_1^L, \infty], x_1^L \leq 0$	$[-\infty, x_1^L \cdot x_2^L]$	$[-\infty, \infty]$	$[x_1^L \cdot x_2^U, \infty]$	$[0,0]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$
$[x_1^L, \infty], x_1^U \geq 0$	$[-\infty, x_1^L \cdot x_2^U]$	$[-\infty, \infty]$	$[x_1^L \cdot x_2^L, \infty]$	$[0,0]$	$[-\infty, x_1^L \cdot x_2^U]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[x_1^L \cdot x_2^L, \infty]$	$[-\infty, \infty]$
$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[0,0]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$

Table 4 Interval arithmetic rules for $[x_1^L, x_1^U] \div [x_2^L, x_2^U]$, $0 \notin [x_2^L, x_2^U]$ [33].

Division	$[x_2^L, x_2^U]$ $x_2^U < 0$	$[x_2^L, x_2^U]$ $x_2^L > 0$	$[-\infty, x_2^U]$ $x_2^U < 0$	$[x_2^L, \infty]$ $x_2^L > 0$
$[x_1^L, x_1^U], x_1^U \leq 0$	$[x_1^U \div x_2^L, x_1^L \div x_2^U]$	$[x_1^L \div x_2^L, x_1^U \div x_2^U]$	$[0, x_1^L \div x_2^U]$	$[x_1^L \div x_2^L, 0]$
$[x_1^L, x_1^U], x_1^L < 0 < x_1^U]$	$[x_1^U \div x_2^U, x_1^L \div x_2^U]$	$[x_1^L \div x_2^L, x_1^U \div x_2^L]$	$[x_1^U \div x_2^U, x_1^L \div x_2^U]$	$[x_1^L \div x_2^L, x_1^U \div x_2^L]$
$[x_1^L, x_1^U], x_1^L \geq 0$	$[x_1^U \div x_2^U, x_1^L \div x_2^L]$	$[x_1^L \div x_2^U, x_1^U \div x_2^L]$	$[x_1^U \div x_2^U, 0]$	$[0, x_1^U \div x_2^L]$
$[0,0]$	$[0,0]$	$[0,0]$	$[0,0]$	$[0,0]$
$[-\infty, x_1^U], x_1^U \leq 0$	$[x_1^U \div x_2^L, \infty]$	$[-\infty, x_1^U \div x_2^U]$	$[0, \infty]$	$[-\infty, 0]$
$[-\infty, x_1^U], x_1^U \geq 0$	$[x_1^U \div x_2^U, \infty]$	$[-\infty, x_1^U \div x_2^L]$	$[x_1^U \div x_2^U, \infty]$	$[-\infty, x_1^U \div x_2^L]$
$[x_1^L, \infty], x_1^L \leq 0$	$[-\infty, x_1^L \div x_2^U]$	$[x_1^L \div x_2^L, \infty]$	$[-\infty, x_1^L \div x_2^U]$	$[x_1^L \div x_2^L, \infty]$
$[x_1^L, \infty], x_1^L \geq 0$	$[-\infty, x_1^L \div x_2^L]$	$[x_1^L \div x_2^U, \infty]$	$[-\infty, 0]$	$[0, \infty]$
$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$	$[-\infty, \infty]$

- k is nondecreasing if g and h are nondecreasing
- k is nondecreasing if g and h are nonincreasing
- k is nonincreasing if g is nondecreasing and h is nonincreasing
- k is nonincreasing if g is nonincreasing and h is nondecreasing

Expression monotonicity of an expression only depends on the monotonicity and bounds of its children, so SUSPECT does not compute derivatives at runtime and only relies on the rules derived from the previous base rules. For nondifferentiable functions, we produce rules for domains where they are differentiable. For example, the function $k(\boldsymbol{x}) = |h(\boldsymbol{x})|$ is not differentiable, but if we restrict $h(\boldsymbol{x})$ to be either nonnegative or nonpositive, then $k(\boldsymbol{x})$ reduces to $k(\boldsymbol{x}) = h(\boldsymbol{x})$ and $k(\boldsymbol{x}) = -h(\boldsymbol{x})$ respectively. Table 6 contains the rules to determine the monotonicity of unary functions, Table 8 contains the rules for binary functions, and Table 7 contains the rules for power functions.

Table 5 Interval arithmetic rules for $[x_1^L, x_1^U] \div [x_2^L, x_2^U]$, $0 \in [x_2^L, x_2^U]$ [33].

Division	$[0,0]$	$[x_2^L,0]$	$[0,x_2^U]$	$[-\infty,0]$	$[0,\infty]$	$[-\infty,\infty]$
$[x_1^L,x_1^U], x_1^U < 0$	Ø	$[x_1^U \div x_2^L, \infty]$	$[-\infty, x_1^U \div x_2^U]$	$[0,\infty]$	$[-\infty,0]$	$[-\infty,\infty]$
$[x_1^L,x_1^U], x_1^L \leq 0 \leq x_1^U$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$
$[x_1^L,x_1^U], x_1^L > 0$	Ø	$[-\infty, x_1^L \div x_2^L]$	$[x_1^L \div x_2^U, \infty]$	$[-\infty,0]$	$[0,\infty]$	$[-\infty,\infty]$
$[-\infty,x_1^U], x_1^U < 0$	Ø	$[x_1^U \div x_2^L, \infty]$	$[-\infty, x_1^U \div x_2^U]$	$[0,\infty]$	$[-\infty,0]$	$[-\infty,\infty]$
$[-\infty,x_1^U], x_1^U > 0$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$
$[x_1^L,\infty], x_1^L < 0$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$
$[x_1^L,\infty], x_1^L > 0$	Ø	$[-\infty, x_1^L \div x_2^L]$	$[x_1^L \div x_2^U, \infty]$	$[-\infty,0]$	$[0,\infty]$	$[-\infty,\infty]$
$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$	$[-\infty,\infty]$

Similarly to monotonicity detection, recursive rules compute convexity. We define the convexity of an expression as linear, convex, concave, or indefinite. Variables and constants are always linear. We can examine the rules used to determine the convexity of the composition $k = g \circ h$ [9]. First we consider the case where the function g depends on one argument only, that is when $l = 1$, $g : \mathbb{R} \to \mathbb{R}$, $h : \mathbb{R}^n \to \mathbb{R}$, and the second derivative of k is

$$k''(\boldsymbol{x}) = g''(h(\boldsymbol{x}))[\nabla h(\boldsymbol{x})]^2 + g'(h(\boldsymbol{x}))\nabla^2 h(\boldsymbol{x})$$

- k is convex if g is convex and nondecreasing over Z and h is convex
- k is convex if g is convex and nonincreasing over Z and h is concave
- k is concave if g is concave and nondecreasing over Z and h is concave
- k is concave if g is concave and nonincreasing over Z and g is convex

If we consider the general case where $l \geq 1$, then the function k is:

$$k(\boldsymbol{x}) = g(h(\boldsymbol{x})) = g(h_1(\boldsymbol{x}), \ldots, h_l(\boldsymbol{x}))$$

its Hessian is

$$\nabla^2 k(\boldsymbol{x}) = \nabla h(\boldsymbol{x})^T \nabla^2 g(h(\boldsymbol{x}))\nabla h(\boldsymbol{x}) + \nabla g(h(\boldsymbol{x}))\nabla^2 h(\boldsymbol{x})$$

and we can derive the following rules:

- k is convex if g is convex and nondecreasing in each argument, and all h_i are convex
- k is convex if g is convex and nonincreasing in each argument, and all h_i are concave
- k is concave if g is concave and nondecreasing in each argument, and all h_i are concave
- k is concave if g is concave and nonincreasing in each argument, and all h_i are convex

Table 9 contains the rules to determine the convexity of unary functions, Table 11 contains the rules for binary functions, and Table 10 contains the rules for power functions.

D The `Bound` Interface

SUSPECT deals with floating point numerical issues by encapsulating all operations that can introduce numerical errors in the `Bound` interface. Users can provide an implementation of this interface that uses, for example, safe interval arithmetic. Table 12 contains a list of abstract methods that implementations have to implement.

Table 6 Monotonicity rules for unary functions. The monotonicity of $g \circ h$ is the conjunction of the conditions on the monotonicity and image of h, and some additional conditions, e.g. $|h|$ is nondecreasing if h is nondecreasing and nonnegative.

	$g \circ h$ monotonicity	h monotonicity	h image	Additional Conditions
$-h$	nondecreasing	nonincreasing	any	
	nonincreasing	nondecreasing	any	
$\lvert h \rvert$	nondecreasing	nondecreasing	nonnegative	
	nonincreasing	nonincreasing	nonnegative	
	nondecreasing	nonincreasing	nonpositive	
	nonincreasing	nondecreasing	nonpositive	
$\sqrt{h}$	nondecreasing	nondecreasing	any	
	nonincreasing	nonincreasing	any	
e^h	nondecreasing	nondecreasing	any	
	nonincreasing	nonincreasing	any	
$\log h$	nondecreasing	nondecreasing	any	
	nonincreasing	nonincreasing	any	
$\sin h$	nondecreasing	nondecreasing	any	$\cos h$ nonnegative
	nondecreasing	nonincreasing	any	$\cos h$ nonpositive
	nonincreasing	nonincreasing	any	$\cos h$ nonnegative
	nonincreasing	nondecreasing	any	$\cos h$ nonpositive
$\cos h$	nondecreasing	nonincreasing	any	$\sin h$ nonnegative
	nondecreasing	nondecreasing	any	$\sin h$ nonpositive
	nonincreasing	nondecreasing	any	$\sin h$ nonnegative
	nonincreasing	nonincreasing	any	$\sin h$ nonpositive
$\tan h$	nondecreasing	nondecreasing	any	
	nonincreasing	nonincreasing	any	
$\arcsin h$	nondecreasing	nondecreasing	any	
	nonincreasing	nonincreasing	any	
$\arccos h$	nondecreasing	nonincreasing	any	
	nonincreasing	nondecreasing	any	
$\arctan h$	nondecreasing	nondecreasing	any	
	nonincreasing	nonincreasing	any	

Table 7 Monotonicity rules for power functions. If both the base and the exponent are not constant, then we do not infer expression monotonicity. The monotonicity of $g \circ h$ is the conjunction of the monotonicity conditions and image of h, e.g. $g^{2i},\ i = 1, 2, \dots$ is nondecreasing if g is nondecreasing and nonnegative.

	$g \circ h$ monotonicity	g monotonicity	g image	Additional Conditions	
α^g	nondecreasing	nondecreasing	nonpositive	$0 < \alpha < 1$	
	nondecreasing	nonincreasing	nonnegative	$0 < \alpha < 1$	
	nondecreasing	nondecreasing	nonnegative	$\alpha \geq 1$	
	nondecreasing	nonincreasing	nonpositive	$\alpha \geq 1$	
g^α	constant	any	any	$\alpha = 0$	
	nondecreasing	nondecreasing	any	$\alpha = 1$	
	nonincreasing	nonincreasing	any	$\alpha = 1$	
	nondecreasing	nondecreasing	nonnegative	$\alpha = 2i$	$i = 1, 2, \dots$
	nondecreasing	nonincreasing	nonpositive	$\alpha = 2i$	$i = 1, 2, \dots$
	nonincreasing	nondecreasing	nonpositive	$\alpha = 2i$	$i = 1, 2, \dots$
	nonincreasing	nonincreasing	nonnegative	$\alpha = 2i$	$i = 1, 2, \dots$
	nondecreasing	nondecreasing	nonpositive	$\alpha = -2i$	$i = 1, 2, \dots$
	nondecreasing	nonincreasing	nonnegative	$\alpha = -2i$	$i = 1, 2, \dots$
	nonincreasing	nondecreasing	nonnegative	$\alpha = -2i$	$i = 1, 2, \dots$
	nonincreasing	nonincreasing	nonpositive	$\alpha = -2i$	$i = 1, 2, \dots$
	nondecreasing	nondecreasing	any	$\alpha = 2i + 1$	$i = 1, 2, \dots$
	nonincreasing	nonincreasing	any	$\alpha = 2i + 1$	$i = 1, 2, \dots$
	nondecreasing	nonincreasing	any	$\alpha = -2i + 1$	$i = 1, 2, \dots$
	nonincreasing	nondecreasing	any	$\alpha = -2i + 1$	$i = 1, 2, \dots$
	nondecreasing	nondecreasing	nonnegative	$\alpha > 0$	
	nonincreasing	nonincreasing	nonnegative	$\alpha > 0$	
	nondecreasing	nonincreasing	nonnegative	$\alpha < 0$	
	nonincreasing	nondecreasing	nonnegative	$\alpha < 0$	

Table 8 Monotonicity rules for binary functions in the form $g \circ h$. The monotonicity of $g \circ h$ is the conjunction of the monotonicity conditions and image of h, and some additional conditions, e.g. $g+h$ is nondecreasing if both g and h are nondecreasing.

	$g \circ h$ monotonicity	g monotonicity	g image	h monotonicity	h image
$g+h$	nondecreasing	nondecreasing	any	nondecreasing	any
	nonincreasing	nonincreasing	any	nonincreasing	any
gh	nondecreasing	nondecreasing	any	constant	nonnegative
	nondecreasing	nonincreasing	any	constant	nonpositive
	nonincreasing	nondecreasing	any	constant	nonpositive
	nonincreasing	nonincreasing	any	constant	nonnegative
	nondecreasing	nondecreasing	nonnegative	nondecreasing	nonnegative
	nondecreasing	nondecreasing	nonpositive	nonincreasing	nonnegative
	nondecreasing	nonincreasing	nonnegative	nondecreasing	nonpositive
	nondecreasing	nonincreasing	nonpositive	nonincreasing	nonpositive
	nonincreasing	nonincreasing	nonnegative	nonincreasing	nonnegative
	nonincreasing	nonincreasing	nonpositive	nondecreasing	nonnegative
	nonincreasing	nondecreasing	nonnegative	nonincreasing	nonpositive
	nonincreasing	nondecreasing	nonpositive	nondecreasing	nonpositive
g/h	nondecreasing	nondecreasing	any	constant	nonnegative
	nondecreasing	nonincreasing	any	constant	nonpositive
	nonincreasing	nondecreasing	any	constant	nonpositive
	nonincreasing	nonincreasing	any	constant	nonnegative
	nondecreasing	nondecreasing	nonnegative	nonincreasing	nonnegative
	nondecreasing	nondecreasing	nonpositive	nondecreasing	nonnegative
	nondecreasing	nonincreasing	nonnegative	nonincreasing	nonpositive
	nondecreasing	nonincreasing	nonpositive	nondecreasing	nonpositive
	nonincreasing	nonincreasing	nonnegative	nondecreasing	nonnegative
	nonincreasing	nonincreasing	nonpositive	nonincreasing	nonnegative
	nonincreasing	nondecreasing	nonnegative	nondecreasing	nonpositive
	nonincreasing	nondecreasing	nonpositive	nonincreasing	nonpositive

Table 9 Convexity rules for unary functions. The convexity of $g \circ h$ is the conjunction of the conditions on the convexity and image of h, and some additional conditions. For example, $|h|$ is convex if h is linear, or if h is convex and nonnegative, or if h is concave and nonpositive.

	$g \circ h$ convexity	h convexity	h image	Additional Conditions
$-h$	convex	concave	any	
	concave	convex	any	
$\|h\|$	convex	linear	any	
	convex	convex	nonnegative	
	convex	concave	nonpositive	
$\sqrt{h}$	concave	concave	nonnegative	
e^h	convex	convex	any	
$\log h$	concave	concave	nonnegative	
$\sin h$	concave	linear	any	$\sin h$ nonnegative
	concave	convex	any	$\sin h$ nonnegative and $\cos h$ nonpositive
	concave	concave	any	$\sin h$ nonnegative and $\cos h$ nonnegative
	convex	linear	any	$\sin h$ nonpositive
	convex	concave	any	$\sin h$ nonpositive and $\cos h$ nonpositive
	convex	convex	any	$\sin h$ nonpositive and $\cos h$ nonnegative
$\cos h$	concave	linear	any	$\cos h$ nonnegative
	concave	convex	any	$\cos h$ nonnegative and $\sin h$ nonpositive
	concave	concave	any	$\cos h$ nonnegative and $\sin h$ nonnegative
	convex	linear	any	$\cos h$ nonpositive
	convex	concave	any	$\cos h$ nonpositive and $\sin h$ nonpositive
	convex	convex	any	$\cos h$ nonpositive and $\sin h$ nonnegative
$\tan h$	convex	convex	any	$\tan h$ nonnegative
	concave	concave	any	$\tan h$ nonpositive
$\arcsin h$	convex	convex	any	$\arcsin h$ nonnegative
	concave	concave	any	$\arcsin h$ nonpositive
$\arccos h$	convex	concave	any	$\arccos h$ nonpositive
	concave	convex	any	$\arccos h$ nonnegative
$\arctan h$	convex	convex	nonpositive	
	concave	concave	nonnegative	

Table 10 Convexity rules for power functions. If both the base and the exponent are not constant, then we can't infer the convexity of the expression. The convexity of $g \circ h$ is the conjunction of the conditions on the convexity and image of h, and some additional conditions. For example, g^{2i}, $i = 1, 2, \ldots$ is convex if g is linear, or if g is convex and nonnegative, or if g is concave and nonpositive.

		g		Additional Conditions
	convexity	convexity	image	
α^g	convex	concave	any	$0 < \alpha < 1$
	convex	convex	any	$\alpha \geq 1$
g^α	linear	any	any	$\alpha = 0$
	linear	linear	any	$\alpha = 1$
	convex	convex	any	$\alpha = 1$
	concave	concave	any	$\alpha = 1$
	convex	linear	any	$\alpha = 2i \quad i = 1, 2, \ldots$
	convex	convex	nonnegative	$\alpha = 2i \quad i = 1, 2, \ldots$
	convex	concave	nonpositive	$\alpha = 2i \quad i = 1, 2, \ldots$
	convex	convex	nonpositive	$\alpha = -2i \quad i = 1, 2, \ldots$
	convex	concave	nonpositive	$\alpha = -2i \quad i = 1, 2, \ldots$
	concave	convex	nonnegative	$\alpha = -2i \quad i = 1, 2, \ldots$
	concave	concave	nonpositive	$\alpha = -2i \quad i = 1, 2, \ldots$
	convex	convex	nonnegative	$\alpha = 2i+1 \quad i = 1, 2, \ldots$
	concave	concave	nonpositive	$\alpha = 2i+1 \quad i = 1, 2, \ldots$
	convex	convex	nonnegative	$\alpha > 1$
	convex	concave	nonnegative	$\alpha < 0$
	concave	concave	nonnegative	$0 < \alpha < 1$
	concave	convex	nonnegative	$\alpha < 0$

Table 11 Convexity rules for binary functions in the form $g \circ h$. The convexity of $g \circ h$ is the conjunction of the conditions on the convexity and image of h, e.g. $g + h$ is convex if both g and h are convex.

	$g \circ h$	g		h	
	convexity	convexity	image	convexity	image
$g + h$	convex	convex	any	convex	any
	concave	concave	any	concave	any
gh	convex	convex	any	constant	nonnegative
	convex	concave	any	constant	nonpositive
	concave	convex	any	constant	nonpositive
	concave	concave	any	constant	nonnegative
g/h	convex	convex	any	constant	positive
	convex	concave	any	constant	negative
	concave	convex	any	constant	negative
	concave	concave	any	constant	positive
	convex	constant	nonnegative	concave	positive
	convex	constant	nonpositive	convex	negative
	concave	constant	nonnegative	convex	negative
	concave	constant	nonpositive	concave	positive

Table 12 Description of the `Bound` interface methods required by concrete interface implementations.

method	description
`is_zero(self)`	Return True if the bound is $[0,0]$
`is_positive(self)`	Return True if the lower bound is greater than 0
`is_negative(self)`	Return True if the upper bound is less than 0
`is_nonnegative(self)`	Return True if the lower bound is greater than or equal to 0
`is_nonpositive(self)`	Return True if the upper bound is less than or equal to 0
`tighten(self, other)`	Return a new bound which is the intersection of the two bounds
`add(self, other)`	Return a new bound which is the sum of the two bounds
`sub(self, other)`	Return a new bound which is the difference of the two bounds
`mul(self, other)`	Return a new bound which is the product of the two bounds
`div(self, other)`	Return a new bound which is the quotient of the two bounds
`equals(self, other)`	Return True if the two bounds are equal
`contains(self, other)`	Return True if `other` is contained in `self`
`zero()`	Return a new bound for which `is_zero` is True
`size(self)`	Return the size of the bound

E RecLMTD Convexity Detector Implementation

This section contains the implementation of a convexity detector for the reciprocal of log mean temperature difference with the limits defined:

$$\mathrm{RecLMTD}^{\beta}(x,y) = \begin{cases} \left(\frac{\ln(x/y)}{x-y}\right)^{\beta} & x \neq y, \\ 1/x^{\beta} & x = y, \end{cases}$$

with variables $x,y \in \mathbb{R}_+$, and constant parameter $\beta \geq -1$. We can extend SUSPECT to detect this type of expression by inspecting power expressions that match the subgraph shown in Figure 7 for the case $x \neq y$ and division expressions for the case $x = y$.

Listing 1 Detector class for the $\mathrm{RecLMTD}^{\beta}$ expression defined in Equation (E).

```
from suspect.ext import ConvexityDetector, Convexity
import suspect.dag.expressions as dex


class RecLMTDDetector(ConvexityDetector):
    def register_handlers(self):
        return {
            dex.DivisionExpression: self.visit_division,
            dex.PowExpression: self.visit_power,
        }

    def visit_division(self, expr, ctx):
        # Method definition in Listing 2

    def visit_power(self, expr, ctx):
        # Method definition in Listing 3

    def _convexity_from_beta(self, beta):
        """Return expression convexity based on value of exponent."""
        if beta > 0:
            return Convexity.Convex
        elif beta == 0:
            return Convexity.Linear
        elif -1 <= beta < 0:
            return Convexity.Concave
```

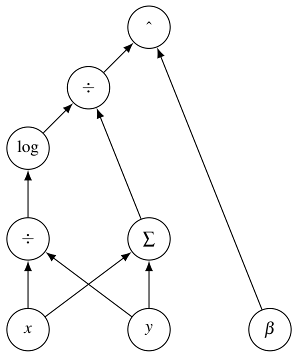

Fig. 7 Graph structure of RecLMTD$^{\beta}$ expression. Vertices x and y are positive variables, and node log represent the natural log.

Listing 2 Detector method for the expression $1/x^{\beta}$ defined in Equation (E).

```
def visit_division(self, expr, ctx):
    num, den = expr.children
    # numerator has to be 1.0
    if not isinstance(num, dex.Constant) or exp.value != 1.0:
        return

    # denominator has to be power with beta >= -1.0
    if not isinstance(den, dex.PowExpression):
        return
    base, expo = den.children
    if not isinstance(expo, dex.Constant) or expo.value < -1.0:
        return
    if not isinstance(base, dex.Variable):
        return

    bound = ctx.bound[base]
    if bound.is_nonnegative():
        return self._convexity_from_beta(expo.value)
```

Listing 3 Detector method for the expression $(\ln(x/)/(x-y))^{\beta}$ defined in Equation (E).

```
def visit_power(self, expr, ctx):
    base, expo = expr.children
    # check exponent is a constant term >= -1.0
    if not isinstance(expo, dex.Constant) or expo.value < -1.0:
        return
    # check base is a division
    if not isinstance(base, dex.DivisionExpression):
        return

    num, den = base.children
    # Check numerator is log expression
    if not isinstance(num, dex.LogExpression):
        return
    # Check denominator is linear expression
    if not isinstance(den, dex.LinearExpression):
        return

    # Check expression inside log is a division
    if not isinstance(num.children[0], dex.DivisionExpression):
        return

    # Check the denominator linear expression has two elements
    if len(den.children) != 2:
        return

    x, y = den.children
    bound_x = ctx.bound[x]
    bound_y = ctx.bound[y]
    c_x, c_y = den.coefficients

    inner_div = num.children[0]
    inner_num, inner_den = inner_div.children

    # Check both x and y are positive
    if bound_x.lower_bound <= 0 or bound_y.lower_bound <= 0:
        return

    # Use coefficients to determine which variable has to
    # be the numerator and denominator in the inner division
    if c_x == 1 and c_y == -1:
        # x - y
        if inner_num is x and inner_den is y:
            return self._convexity_from_beta(expo.value)
    elif c_x == -1 and c_y == 1:
        # y - x
        if inner_num is y and inner_den is x:
            return self._convexity_from_beta(expo.value)
```

Listing 4 Example `setup.py` file used to register the detector of Listing 1 with SUSPECT.

```
from setuptools import setup, find_packages

setup(
    name='rec_lmtd_detector',
    packages=find_packages(),
    entry_points={
        # register rec_lmtd_detector as SUSPECT covexity detector plugin
        'suspect.convexity_detection': [
            'rec_lmtd=rec_lmtd_detector:RecLMTDDetector'
        ]
    },
    # ensure SUSPECT package is installed
    requires=['cog-suspect']
)
```

F Results

This Appendix contains a more detailed breakdown of the computational results. Table 13 contains a list of instances that SUSPECT is able to process correctly, but that because of the problem size it takes over 10 seconds of processing time. Table 14 contains a list of instances where SUSPECT is not able to terminate within the 5 minutes timeout. Finally, Table 15 contains the problems where we don't identify convexity correctly, instances marked with a dagger† fail because of numerical errors when computing eigenvalues.

Table 13 List of instances where we correctly detected the convexity properties in over 10 seconds.

arki0008	arki0018	autocorr_bern35-35
autocorr_bern40-30	autocorr_bern40-40	autocorr_bern45-23
autocorr_bern45-34	autocorr_bern45-45	autocorr_bern50-25
autocorr_bern50-38	autocorr_bern50-50	autocorr_bern55-28
autocorr_bern60-30	camshape400	camshape800
chp_shorttermplan2c	crudeoil_lee4_10	crudeoil_li21
crudeoil_pooling_dt1	crudeoil_pooling_dt2	crudeoil_pooling_dt3
crudeoil_pooling_dt4	edgecross20-040	edgecross22-096
edgecross24-057	edgecross24-115	ex8_2_3b
ex8_2_5b	faclay30	faclay30h
faclay33	faclay35	gasprod_sarawak81
genpooling_meyer10	genpooling_meyer15	hvb11
hydroenergy3	ibs2	infeas1
jbearing100	jbearing50	jbearing75
lop97ic	lop97icx	multiplants_stg6
netmod_dol2	nuclear104	nuclear10a
nuclear10b	nuclear49a	nuclear49b
polygon100	pooling_foulds3pq	pooling_foulds3stp
pooling_foulds3tp	pooling_foulds4pq	pooling_foulds4stp
pooling_foulds4tp	pooling_foulds5pq	pooling_foulds5stp
pooling_foulds5tp	pooling_sppa0pq	pooling_sppa0stp
pooling_sppa0tp	pooling_sppa5pq	pooling_sppa5stp
pooling_sppa5tp	portfol_classical200_2	portfol_robust200_03
portfol_shortfall200_05	qap	space960
supplychainp1_053050	supplychainr1_053050	telecomsp_njlata
telecomsp_pacbell	torsion50	turkey
unitcommit2		

Table 14 List of instances where we don't terminate before the 5 minutes timeout.

gams03	pooling_sppa9pq	pooling_sppa9stp	pooling_sppa9tp
pooling_sppb0pq	pooling_sppb0stp	pooling_sppb0tp	pooling_sppb2pq
pooling_sppb2stp	pooling_sppb2tp	pooling_sppc0pq	pooling_sppc0stp
pooling_sppc0tp			

Table 15: Detailed breakdown of instances where we don't identify convexity correctly. Instances marked with a dagger† fail because of numerical errors when computing eigenvalues.

	Constraints Convexity		Objective Convexity		Objective Type	
Instance	Expected	Detected	Expected	Detected	Expected	Detected
clay0203h	convex	indefinite	linear	linear	linear	linear
clay0204h	convex	indefinite	linear	linear	linear	linear
clay0205h	convex	indefinite	linear	linear	linear	linear
clay0303h	convex	indefinite	linear	linear	linear	linear
clay0304h	convex	indefinite	linear	linear	linear	linear
clay0305h	convex	indefinite	linear	linear	linear	linear
cvxnonsep_nsig20	convex	indefinite	linear	linear	linear	linear
cvxnonsep_nsig30	convex	indefinite	linear	linear	linear	linear
cvxnonsep_nsig40	convex	indefinite	linear	linear	linear	linear
cvxnonsep_pcon20	convex	indefinite	linear	linear	linear	linear
cvxnonsep_pcon30	convex	indefinite	linear	linear	linear	linear
cvxnonsep_pcon40	convex	indefinite	linear	linear	linear	linear
cvxnonsep_psig20	linear	linear	convex	indefinite	nonlinear	nonlinear
cvxnonsep_psig30	linear	linear	convex	indefinite	nonlinear	nonlinear
cvxnonsep_psig40	linear	linear	convex	indefinite	nonlinear	nonlinear

Continued on next page

Table 15 (continued)

Instance	Constraints Convexity Expected	Constraints Convexity Detected	Objective Convexity Expected	Objective Convexity Detected	Objective Type Expected	Objective Type Detected
eq6_1 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
fac3 †	linear	linear	convex	indefinite	quadratic	quadratic
gams01	convex	indefinite	convex	convex	nonlinear	nonlinear
gasoil100 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
gasoil200 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
gasoil400 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
gasoil50 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
jbearing25 †	linear	linear	convex	indefinite	quadratic	quadratic
johnall	indefinite	indefinite	indefinite	indefinite	nonlinear	polynomial
lip	linear	linear	convex	concave	nonlinear	nonlinear
methanol100 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
methanol200 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
methanol50 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
pinene100 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
pinene200 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
pinene50 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
pollut	linear	linear	convex	indefinite	nonlinear	nonlinear
popdynm100 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
popdynm200 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
popdynm25 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
popdynm50 †	indefinite	indefinite	convex	indefinite	quadratic	quadratic
portfol_buyin	convex	indefinite	linear	linear	linear	linear
portfol_card	convex	indefinite	linear	linear	linear	linear
portfol_roundlot	convex	indefinite	linear	linear	linear	linear
qp2 †	linear	linear	convex	indefinite	quadratic	quadratic
smallinvSNPr1b010-011 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr1b020-022 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr1b050-055 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr1b100-110 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr1b150-165 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr1b200-220 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr2b010-011 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr2b020-022 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr2b050-055 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr2b100-110 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr2b150-165 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr2b200-220 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr3b010-011 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr3b020-022 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr3b050-055 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr3b100-110 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr3b150-165 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr3b200-220 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr4b010-011 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr4b020-022 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr4b050-055 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr4b100-110 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr4b150-165 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr4b200-220 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr5b010-011 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr5b020-022 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr5b050-055 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr5b100-110 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr5b150-165 †	convex	indefinite	linear	linear	linear	linear
smallinvSNPr5b200-220 †	convex	indefinite	linear	linear	linear	linear
st_e17	convex	indefinite	linear	linear	linear	linear
st_qpc-m3a †	linear	linear	concave	indefinite	quadratic	quadratic
st_qpc-m3b †	linear	linear	concave	indefinite	quadratic	quadratic
st_qpc-m3c †	linear	linear	concave	indefinite	quadratic	quadratic
tls12	convex	indefinite	linear	linear	linear	linear
tls2	convex	indefinite	linear	linear	linear	linear

Continued on next page

Table 15 (continued)

Instance	Constraints Convexity		Objective Convexity		Objective Type	
	Expected	Detected	Expected	Detected	Expected	Detected
tls4	convex	indefinite	linear	linear	linear	linear
tls5	convex	indefinite	linear	linear	linear	linear
tls6	convex	indefinite	linear	linear	linear	linear
tls7	convex	indefinite	linear	linear	linear	linear
watercontamination0202r †	linear	linear	convex	indefinite	quadratic	quadratic
watercontamination0303r †	linear	linear	convex	indefinite	quadratic	quadratic