A Pseudocode for the interval Krawczyk method in one and two dimensions

Here we give the pseudocode for the interval Krawczyk method in one and two dimensions, which seeks zeros of a given function.

We use the following functions of the intervals:

Midpoint:

$$\text{mid } x = \frac{\overline{x} + \underline{x}}{2}$$

Radius:

$$\text{rad } x = \frac{\overline{x} - \underline{x}}{2}$$

Diameter:

$$\text{diam } x = \overline{x} - \underline{x}$$

Magnitude:

$$\text{mag } x = \max(|\underline{x}|, |\overline{x}|)$$

A.1 Krawczyk method in one dimension

For the factor C in (10) to not return "division by thin zero" error, we slightly change the denominator in this case.

Algorithm 1 1D Krawczyk operator

```
function C(f, x)
    if f'(mid x) == 0 then
        return 1/(f'(mid x) + 0.0001rad x)
    else
        return 1/f'(mid x)
    end if
end function
```

$$\boldsymbol{K}(\boldsymbol{x}) = \text{mid } \boldsymbol{x} - C(f, \boldsymbol{x}) f(\text{mid } \boldsymbol{x})(1 - C(f, \boldsymbol{x}) f'(\boldsymbol{x}))(\boldsymbol{x} - \text{mid } \boldsymbol{x})$$

A.2 Krawczyk method in two dimensions

The Jacobian in (12) has to be invertible; if it is not, we slightly modify it.

A.3 Krawczyk method in two dimensions with purity

In this algorithm, we specify the tolerance δ, by reaching which the intervals of this size which still have purity 0 are left out. The function $purity(f, x)$ evaluates the intersection of x and the domain of the function and returns 1 if x is fully inside of the domain, 0 if x is partially inside the domain, and -1 if x is outside of the domain. If during the bisection procedure the interval has purity 1, we apply the regular Krawczyk method, if

Algorithm 2 Krawczyk algorithm in 1D

```
A = [ ]                                                      ▷ Empty array of roots
δ = 10^{-10}
function KRAWCZYK(f, a)
    ▷ If the midpoint of the starting interval is 0, it should be slightly asymmetrized to avoid issues
    if mid a == 0 then
        a = [a̲, ā + 0.0001mag a]
    end if
    k_a = a ∩ K(f, a)
    if k_a ≠ ∅ then
        if diam k_a < δ then
            if k_a ⊂ a then
                                ▷ k_a contains a unique zero; add it to the array of roots
                k_a → A
            else
                                ▷ k_a contains a possible zero; add it to the array of roots
                k_a → A
            end if
        else
                          ▷ Bisect the current interval and apply the function recursively
            krawczyk(f, [k̲_a, mid k_a])
            krawczyk(f, [mid k_a, k̄_a])
        end if
    end if
end function
```

Algorithm 3 2D Krawczyk operator

function $\overrightarrow{Y}(f, \boldsymbol{x})$

  if det $\left(\frac{\partial \overrightarrow{f}}{\partial(\text{mid } \overrightarrow{x})}\right) == 0$ **then**

    return $1/\left(\frac{\partial \overrightarrow{f}}{\partial(\text{mid } \overrightarrow{x} + 0.0001\|\text{diam } \overrightarrow{x}\|)}\right)$

  else

    return $1/\left(\frac{\partial \overrightarrow{f}}{\partial(\text{mid } \overrightarrow{x})}\right)$

  end if

end function

$M(f, x) = 1 - Y(f, x)\frac{\partial \overrightarrow{f}}{\partial(\text{mid } \overrightarrow{x})}$

$K(f, x) = \text{mid } x - Y(f, x) f(\text{mid } x) + M(f, x)(x - \text{mid } x)$

Algorithm 4 Krawczyk algorithm in 2D

$A = [\,]$ ▷ Empty array of roots
$\delta = 10^{-10}$
function BISECT(a)
    $b = [\,]$
    $([\underline{a_1}, \text{mid } a_1], [\underline{a_2}, \text{mid } a_2]) \to b$
    $([\text{mid } a_1, \overline{a_1}], [\underline{a_2}, \text{mid } a_2]) \to b$
    $([\underline{a_1}, \text{mid } a_1], [\text{mid } a_2, \overline{a_2}]) \to b$
    $([\text{mid } a_1, \overline{a_1}], [\text{mid } a_2, \overline{a_2}]) \to b$
    return b
end function
function KRAWCZYK2D($f, \boldsymbol{a}$)
    $\boldsymbol{k}_a = \boldsymbol{a} \cap \boldsymbol{K}(f, \boldsymbol{a})$
    if $\boldsymbol{k}_a \neq \emptyset$ **then**
        if $(\text{diam } \boldsymbol{k}_a)_1 < \delta$ and $(\text{diam } \boldsymbol{k}_a)_2 < \delta$ **then**
            if $\boldsymbol{k}_a \subset \boldsymbol{a}$ **then**
                ▷ $\boldsymbol{k}_a$ contains a unique zero; add it to the array of roots
                $\boldsymbol{k}_a \to A$
            else
                ▷ $\boldsymbol{k}_a$ contains a possible zero; add it to the array of roots
                $\boldsymbol{k}_a \to A$
            end if
        else
            ▷ Bisect the current interval and apply the function recursively
            $b = bisect(a)$
            for $i = 1$ to 4 **do**
                $krawczyk2d(f, b_i)$
            end for
        end if
    end if
end function

Algorithm 5 Krawczyk algorithm in 2D with purity

```
A = [ ]                                              ▷ Empty array of roots
δ = 10^{-5}
function KRAWCZYK2D_PURITY(f, a)
    p = purity(f, a)
    if p ≠ −1 then
        if p == 1 then
            roots = krawczyk2d(f, a)
            if length(roots) > 0 then
                roots → A
            end if
        else if p == 0 then
            if max (diam a)_1 < δ and max (diam a)_2 < δ then
                                  ▷ Discard interval with the size below tolerance
            else
                b = bisect(a)
                for i = 1 to 4 do
                    krawczyk2d_purity(f, b_i)
                end for
            end if
        end if
    end if
end function
```

it is 0, we continue bisection and apply the Krawczyk method with purity recursively, and if it is -1, we discard it.