Interval and Affine Arithmetic

Bogdan Burlacu

October 1, 2020

1 Interval Arithmetic

1.1 Preliminaries

In interval arithmetic (IA), a *closed* interval $[a, b]$ is given by the set of real numbers:

$$[a, b] = \{x \in \mathbb{R} : a \leq x \leq b\} \tag{1}$$

Capital letters are commonly used to denote intervals:

$$X = [\underline{X}, \overline{X}] \tag{2}$$

where $\underline{X}$ and $\overline{X}$ are the *infimum* and *supremum*, respectively.

All elementary operations are well-defined for IA and produce bounds that are guaranteed to enclose the actual function bounds.

For a real valued monovariate function f, the range of values $f(x)$ for $x \in X$ (where X is an interval) is called the *image set* of f:

$$f(X) = \{f(x) : x \in X\} \tag{3}$$

In the multivariate case, this becomes :

$$f(X_1, ..., X_n) = \{f(x_1, ..., x_n) : x_1 \in X_1, ..., x_n \in X_n\} \tag{4}$$

1.1.1 United extension

We use the term *united extension* to describe the set images denoted in Equations (3) and (4).

More formally, let $g : M_1 \to M_2$ be a mapping between sets M_1 and M_2, and $S(M_1)$ and $S(M_2)$ the families of subsets of M_1 and M_2, respectively. The *united extension* of g is the set-value mapping $\bar{g} : S(M_1) \to S(M_2)$ such that:

$$\bar{g}(X) = \{g(x) : x \in X, X \in S(M_1)\} \tag{5}$$

Note that $\bar{g}(X)$ contains precisely the same elements as the set image of $g(X)$:

$$\bar{g}(X) = \cup_{x \in X}\{g(x)\} \tag{6}$$

1.2 Interval Extensions

Let $F(X)$ be the corresponding interval-valued function for $f(x)$. We say that F is an *interval extension* of f if for degenerate interval arguments F agrees with f:

$$F([x,x]) = f(x) \tag{7}$$

The interval extension maintains the same meaning and properties for multivariate functions.

1.2.1 Fundamental Theorem of Interval Analysis

From Equation (5) it results that $\bar{g}(X)$ has the following property, called the *subset property*:

$$X, Y \in S(M_1) \text{ with } X \subseteq Y \implies \bar{g}(X) \subseteq \bar{g}(Y) \tag{8}$$

We say that $F = F(X_1, ..., X_n)$ is an *inclusion isotonic* if

$$Y_i \subseteq X_i \text{ for } i = 1, ..., n \implies F(Y_1, ..., Y_n) \subseteq F(X_1, ..., X_n) \tag{9}$$

Then we note that united extensions, which all have the subset property, are inclusion isotonic. The set of operations of IA must sastisfy:

$$Y_1 \subseteq X_1, Y_2 \subseteq X_2 \implies Y_1 \odot Y_2 \subseteq X_1 \odot X_2 \tag{10}$$

We can now state the *fundamental theorem*:

If F is an inclusion isotonic interval extension of f, then

$$f(X_1, ..., X_n) \subseteq F(X_1, ..., X_n) \tag{11}$$

2 Affine Arithmetic

In affine arithmetic, a quantity x is represented as the following affine form:

$$x = x_0 + x_1\epsilon_1 + ... + x_n\epsilon_n \tag{12}$$

where $\epsilon_1, ..., \epsilon_n$ are symbolic real variables whose values are unknown but assumed to lie in $[-1, 1]$. Note that the number n changes during the calculation.

In the case of a multivariate function $f = (x_1, ..., x_m)$ the following affine forms are initialized:

$$x_1 = \frac{\overline{x}_1 + \underline{x}_1}{2} + \frac{\overline{x}_1 - \underline{x}_1}{2}\epsilon_1 \tag{13}$$

$$\vdots \tag{14}$$

$$x_m = \frac{\overline{x}_m + \underline{x}_m}{2} + \frac{\overline{x}_m - \underline{x}_m}{2}\epsilon_m \tag{15}$$

$$\tag{16}$$

where $[\underline{x}_k, \overline{x}_k]$ is the domain of variable x_k.

An affine form can be converted to an interval using the formula:

$$I(x) = [x_0 - \Delta, x_0 + \Delta] \qquad \text{where } \Delta = \sum_{i=1}^{n} |x_i| \tag{17}$$

2.1 Linear operations

For two affine forms, $x = x_0 + \sum_{i=1}^{n} x_i \epsilon_i$ and $y = y_0 + \sum_{i=1}^{n} y_i \epsilon_i$ the following linear operations are defined:

$$x \pm y = (x_0 \pm y_0) + \sum_{i=1}^{n} (x_i \pm y_i)\epsilon_i \tag{18}$$

$$x \pm \alpha = (x_0 \pm \alpha) + \sum_{i=1}^{n} x_i \epsilon_i \tag{19}$$

$$\alpha x = (\alpha x_0) + \sum_{i=1}^{n} (\alpha x_i) \tag{20}$$

A nonlinear function $f(x)$ of an affine form is generally not able to be represented directly as an affine form. We must therefore consider a linear approximation of f and a representation of the approximation error by introducing a new noise symbol ϵ_{n+1}.

Let $X = I(x)$ be the range of x. For a nonlinear function $f(x)$, a linear approximation in the form $ax + b$ will have a maximum approximation error δ:

$$\delta = \max_{x \in X} |f(x) - (ax + b)| \tag{21}$$

The result of the nonlinear operation can then be represented as follows:

$$f(x) = ax + b + \delta\epsilon_{n+1} \tag{22}$$

$$= a(x_0 + x_1\epsilon_1 + ... + x_n\epsilon_n) + b + \delta\epsilon_{n+1} \tag{23}$$

Nonlinear binomial operations are calculated similarly.

3 Minima and maxima of multivariate functions

We consider a multivariate nonlinear function

$$y = f(x_1, ..., x_m) \tag{24}$$

The domain of this function is the m-dimensional region (the box):

$$X^{(0)} = \left(X_1^{(0)}, ..., X_m^{(0)}\right) \tag{25}$$

$$= \left([\underline{X_1^{(0)}}, \overline{X_1^{(0)}}], ..., [\underline{X_m^{(0)}}, \overline{X_m^{(0)}}]\right) \tag{26}$$

One of the first methods to calculate the bounds of the codomain of f is Fujii's method, in which the maxima and minima are calculated with guaranteed accuracy by means of recursively dividing X into subregions and applying interval arithmetic (IA) to bound the range of f in each region. The method discards the subregions that are guaranteed not to contain the point corresponding to the minimum (maximum) value.

3.1 Miyajima and Kashiwagi's method

Without loss of generality, we consider finding maxima of a two-dimensional function $f(x_1, x_2)$ in the box $X^{(0)} = (X_1^{(0)}, X_2^{(0)}) = ([\underline{X_1^{(0)}}, \overline{X_1^{(0)}}], [\underline{X_2^{(0)}}, \overline{X_2^{(0)}}])$.

For an interval J, let the center and the width of J be $c(J)$ and $w(J)$, respectively.

For a box X, let $F_A(X)$ be the range boundary of f in X obtained by applying AA and let the upper bound of $I(F_A(X))$ be $\overline{F_A(X)}$.

Algorithm 1: Algorithm for computing maxima of multivariate function (part 1)

Data: $f(\mathbf{x})$, X (domain of f), stopping criteria ϵ_r, ϵ_b

Result: Maxima (minima) of f

`// Step 1`

Initialize lists $\mathcal{S}$ and $\mathcal{T}$ for storing boxes and range boundaries:

$\mathcal{S} \leftarrow \emptyset$;

$\mathcal{T} \leftarrow \emptyset$;

`// Step 2: divide` $X^{(0)}$ `into subregions` $X^{(1)}$ `and` $X^{(2)}$

if $w(X_1^{(0)}) < w(X_2^{(0)})$ **then**

$$X^{(1)} = ([\underline{X_1^{(0)}}, \overline{X_1^{(0)}}], [\underline{X_2^{(0)}}, c(X_2^{(0)})])$$
$$X^{(2)} = ([\underline{X_1^{(0)}}, \overline{X_1^{(0)}}], [c(X_2^{(0)}), \overline{X_2^{(0)}}])$$

else

$$X^{(1)} = ([\underline{X_1^{(0)}}, c(X_1^{(0)})], [\underline{X_2^{(0)}}, \overline{X_2^{(0)}}])$$
$$X^{(2)} = ([c(X_1^{(0)}), \overline{X_1^{(0)}}], [\underline{X_2^{(0)}}, \overline{X_2^{(0)}}])$$

`// Step 3`

Calculate $F_A(X^{(1)})$ and $F_A(X^{(2)})$, then calculate $\underline{f_{\max}^{(1)}}$ and $\underline{f_{\max}^{(2)}}$ (use algorithm 3). The lower bound of the maxima is then given as $\underline{f_{\max}} = \max(\underline{f_{\max}^{(1)}}, \underline{f_{\max}^{(2)}})$.

`// Step 4`

if $\overline{F_A(X^{(1)})} < \underline{f_{\max}}$ **then**

  Insert $X^{(2)}$ and $F_A(X^{(2)})$ into $\mathcal{S}$ and discard $X^{(1)}$.

else if $\overline{F_A(X^{(2)})} < \underline{f_{\max}}$ **then**

  Insert $X^{(1)}$ and $F_A(X^{(1)})$ into $\mathcal{S}$ and discard $X^{(2)}$.

else

  Insert $X^{(1)}$, $F_A(X^{(1)})$, $X^{(2)}$, $F_A(X^{(2)})$ into $\mathcal{S}$.

Algorithm 2: Algorithm for computing maxima of multivariate function (part 2)

Data: $f(\mathbf{x})$, X (domain of f), stopping criteria ϵ_r, ϵ_b

Result: Maxima (minima) of f

// Step 5

while $\mathcal{S} \neq \emptyset$ **do**

Find the box $X^{(i)} \in \mathcal{S}$ for which $F_A(X^{(i)})$ is largest.

$$X^{(i)} = \arg\max_i \left(F_A(X^{(i)})\right)$$

Remove $X^{(i)}$ from $\mathcal{S}$.

Select $X^{(i)}$ and $F_A(X^{(i)})$ as the box and range to be processed.

Calculate $\underline{f_{\max}^{(i)}}$ (the candidates of $\underline{f_{\max}}$) by utilizing $X^{(i)}$ and $F_A(X^{(i)})$ and by applying algorithm 3. Update $\underline{f_{\max}} = \max\{\underline{f_{\max}^{(i)}}\}$.

Discard any box X and range boundary $F_A(X)$ from $\mathcal{S}$ and $\mathcal{T}$ for which $\overline{F_A(X)} < \underline{f_{\max}}$.

Narrow $X^{(i)}$ down by utilizing $X^{(i)}$, $F_A(X^{(i)})$ and $\underline{f_{\max}}$ using algorithm 4.

Divide $X^{(i)}$ into $X^{(j)}$ and $X^{(k)}$.

if $w(X_1^{(i)}) < w(X_2^{(i)})$ **then**

$$X^{(j)} = ([\underline{X_1^{(i)}}, \overline{X_1^{(i)}}], [\underline{X_2^{(i)}}, c(X_2^{(i)})])$$
$$X^{(k)} = ([\underline{X_1^{(i)}}, \overline{X_1^{(i)}}], [c(X_2^{(i)}), \overline{X_2^{(i)}}])$$

else

$$X^{(j)} = ([\underline{X_1^{(i)}}, c(X_1^{(i)})], [\underline{X_2^{(i)}}, \overline{X_2^{(i)}}])$$
$$X^{(k)} = ([c(X_1^{(i)}), \overline{X_1^{(i)}}], [\underline{X_2^{(i)}}, \overline{X_2^{(i)}}])$$

Calculate $F_A(X^{(j)})$ and $F_A(X^{(k)})$.

if $\max_{1\leq h\leq m} w(X_h^{(j)}) < \epsilon_r$ ***and*** $w(I(F_A(X^{(j)}))) < \epsilon_b$ **then**

Insert $X^{(j)}$ and $F_A(X^{(j)})$ into $\mathcal{T}$.

else

Insert $X^{(j)}$ and $F_A(X^{(j)})$ into $\mathcal{S}$.

if $\max_{1\leq h\leq m} w(X_h^{(k)}) < \epsilon_r$ ***and*** $w(I(F_A(X^{(k)}))) < \epsilon_b$ **then**

Insert $X^{(k)}$ and $F_A(X^{(k)})$ into $\mathcal{T}$.

else

Insert $X^{(k)}$ and $F_A(X^{(k)})$ into $\mathcal{S}$.

// Step 6

Group together boxes in $\mathcal{T}$ that share a common point. Let $Y^{(1)}, ..., Y^{(l)}$ be one such group. Then, the maxima is given by $\cup_{h=1}^{l} I(F_A(Y^{(h)}))$, with corresponding point $\cup_{h=1}^{l} Y^{(h)}$. Repeat for all groups.

Algorithm 3: Algorithm 1

```
// Compared to Fujii's method, this algorithm is able to calculate candidates
   bounding f_max more closely, therefore this allows to discard more subregions
   (boxes) in the initial stage.
```

Suppose $F_A(X)$ is calculated as follows:

$$F_A(X) = a_0 + a_1\epsilon_1 + \ldots + a_m + a_{m+1} + \ldots + a_n\epsilon_n \tag{27}$$

Let the point (vector) $y = (y_1, \ldots, y_m)$ be as follows:

$$y_i = \begin{cases} \overline{X_i} & 0 < a_i \\ \underline{X_i} & a_i < 0 \qquad (i = 1, \ldots, m) \\ c(X_i) & \text{otherwise.} \end{cases} \tag{28}$$

$$\tag{29}$$

Then, the candidate for $\underline{f_{\max}}$ is calculated as $f(y)$.

Algorithm 4: Algorithm 2

Calculate $F_A(X)$ using Equation (27).

Calculate

$$\alpha = \sum_{i=m+1}^{n} |a_i| \tag{30}$$

forall $i = 1, \ldots, m$ **do**

  if $a_i \neq 0$ **then**

    Apply IA (interval arithmetic) as follows:

$$\varepsilon_i^* = \frac{1}{a_i}\left(f_{\max} - a_0 - \alpha - \sum_{j=1, j\neq i}^{m} (a_j \times [-1, 1])\right) \tag{31}$$

  else

    Let $\varepsilon_i^* = [-1, 1]$.

  Narrow X_i down as follows: **if** $\varepsilon_i^* \in [-1, 1]$ **then**

$$X_i = \begin{cases} [\underline{X_i} + r(X_i)(\underline{\varepsilon_i^*} + 1), \overline{X_i}] & 0 < a_i \\ [\underline{X_i}, \overline{X_i} - r(X_i)(1 - \overline{\varepsilon_i^*})] & a_i < 0 \end{cases} \text{ where } r(X_i) = \frac{\overline{X_i} - \underline{X_i}}{2}$$

  else if $\underline{\varepsilon_i^*} \leq -1$ ***and*** $\overline{\varepsilon_i^*} \in [-1, 1)$ ***and*** $a_i < 0$ **then**

$$X_i = [\underline{X_i}, \overline{X_i} - r(X_i)(1 - \overline{\varepsilon_i^*})]$$

  else if $\underline{\varepsilon_i^*} \in (-1, 1]$ ***and*** $1 \leq \overline{\varepsilon_i^*}$ ***and*** $0 < a_i$ **then**

$$X_i = [\underline{X_i} + r(X_i)(\underline{\varepsilon_i^*} + 1), \overline{X_i}]$$

  else

    We are not able to narrow X_i down.