Interval Matrix Multiplication Algorithms Implementation and Accuracy

Philippe Théveny[1] Nathalie Revol[2]

[1]École Normale Supérieure de Lyon

[2]INRIA Grenoble – Rhône Alpes

SCAN 2012 – Novosibirsk

Content

Content

Infsup Arithmetic Formulas

Definition An interval $\boldsymbol{x}$ is a closed convex subset of $\mathbb{R}$.

Infsup Representation $\boldsymbol{x} = [\underline{x}, \overline{x}] = \{y \mid \underline{x} \le y \le \overline{x}\}$

- addition

$$\boldsymbol{x} + \boldsymbol{y} = [\underline{x} + \underline{y}, \overline{x} + \overline{y}]$$

- subtraction

$$\boldsymbol{x} - \boldsymbol{y} = [\underline{x} - \overline{y}, \overline{x} - \underline{y}]$$

- multiplication

$$\begin{aligned}\boldsymbol{x} \times \boldsymbol{y} = [\,&\min(\underline{x} \times \underline{y}, \underline{x} \times \overline{y}, \overline{x} \times \underline{y}, \overline{x} \times \overline{y}),\\ &\max(\underline{x} \times \underline{y}, \underline{x} \times \overline{y}, \overline{x} \times \underline{y}, \overline{x} \times \overline{y})]\end{aligned}$$

Classical Algorithm for Matrix Product

Input: $\boldsymbol{A} = [\underline{A}, \overline{A}] \in \mathbb{IF}^{m \times k}$, $\boldsymbol{B} = [\underline{B}, \overline{B}] \in \mathbb{IF}^{k \times n}$
Output: $\boldsymbol{C} \in \mathbb{IF}^{m \times n}$, $\boldsymbol{C} \supseteq \boldsymbol{A} \times \boldsymbol{B}$

```
for i = 1 to m do
    for j = 1 to n do
        C̲_ij ← 0; C̄_ij ← 0
        for l = 1 to k do
            C̲_ij ← fl_▽(C̲_ij + min{A̲_il × B̲_lj, A̲_il × B̄_lj, Ā_il × B̲_lj, Ā_il × B̄_lj})
            C̄_ij ← fl_△(C̄_ij + max{A̲_il × B̲_lj, A̲_il × B̄_lj, Ā_il × B̲_lj, Ā_il × B̄_lj})
        end for
    end for
end for
return [C̲, C̄]
```

$\underline{\boldsymbol{C}}_{ij} \leftarrow fl_{\nabla}(\underline{\boldsymbol{C}}_{ij} + \min\{\underline{A}_{il} \times \underline{B}_{lj}, \underline{A}_{il} \times \overline{B}_{lj}, \overline{A}_{il} \times \underline{B}_{lj}, \overline{A}_{il} \times \overline{B}_{lj}\})$

$\overline{\boldsymbol{C}}_{ij} \leftarrow fl_{\Delta}(\overline{\boldsymbol{C}}_{ij} + \max\{\underline{A}_{il} \times \underline{B}_{lj}, \underline{A}_{il} \times \overline{B}_{lj}, \overline{A}_{il} \times \underline{B}_{lj}, \overline{A}_{il} \times \overline{B}_{lj}\})$

Classical Algorithm for Matrix Product

Input: $\boldsymbol{A} = [\underline{A}, \overline{A}] \in \mathbb{IF}^{m \times k}$, $\boldsymbol{B} = [\underline{B}, \overline{B}] \in \mathbb{IF}^{k \times n}$
Output: $\boldsymbol{C} \in \mathbb{IF}^{m \times n}$, $\boldsymbol{C} \supseteq \boldsymbol{A} \times \boldsymbol{B}$

for $i = 1$ to m **do**
  for $j = 1$ to n **do**
    $\underline{C}_{ij} \leftarrow 0$; $\overline{C}_{ij} \leftarrow 0$
    for $l = 1$ to k **do**
      $\underline{C}_{ij} \leftarrow fl_{\nabla}(\underline{C}_{ij} + \min\{\underline{A}_{il} \times \underline{B}_{lj}, \underline{A}_{il} \times \overline{B}_{lj}, \overline{A}_{il} \times \underline{B}_{lj}, \overline{A}_{il} \times \overline{B}_{lj}\})$
      $\overline{C}_{ij} \leftarrow fl_{\Delta}(\overline{C}_{ij} + \max\{\underline{A}_{il} \times \underline{B}_{lj}, \underline{A}_{il} \times \overline{B}_{lj}, \overline{A}_{il} \times \underline{B}_{lj}, \overline{A}_{il} \times \overline{B}_{lj}\})$
    end for
  end for
end for
return $[\underline{C}, \overline{C}]$

Classical Algorithm for Matrix Product

Input: $\boldsymbol{A} = [\underline{A}, \overline{A}] \in \mathbb{IF}^{m\times k}$, $\boldsymbol{B} = [\underline{B}, \overline{B}] \in \mathbb{IF}^{k\times n}$
Output: $\boldsymbol{C} \in \mathbb{IF}^{m\times n}$, $\boldsymbol{C} \supseteq \boldsymbol{A} \times \boldsymbol{B}$

```
for i = 1 to m do
    for j = 1 to n do
        C̲_ij ← 0; C̄_ij ← 0
        for l = 1 to k do
            C̲_ij ← fl_∇(C̲_ij + min{A̲_il × B̲_lj, A̲_il × B̄_lj, Ā_il × B̲_lj, Ā_il × B̄_lj})
            C̄_ij ← fl_Δ(C̄_ij + max{A̲_il × B̲_lj, A̲_il × B̄_lj, Ā_il × B̲_lj, Ā_il × B̄_lj})
        end for
    end for
end for
return [C̲, C̄]
```

$\underline{C}_{ij} \leftarrow fl_{\nabla}(\underline{C}_{ij} + \min\{\underline{A}_{il} \times \underline{B}_{lj}, \underline{A}_{il} \times \overline{B}_{lj}, \overline{A}_{il} \times \underline{B}_{lj}, \overline{A}_{il} \times \overline{B}_{lj}\})$

$\overline{C}_{ij} \leftarrow fl_{\Delta}(\overline{C}_{ij} + \max\{\underline{A}_{il} \times \underline{B}_{lj}, \underline{A}_{il} \times \overline{B}_{lj}, \overline{A}_{il} \times \underline{B}_{lj}, \overline{A}_{il} \times \overline{B}_{lj}\})$

Midrad Addition/Subtraction Formulas

Neumaier, *Interval methods for systems of equations*, 1990

Midrad Representation $\boldsymbol{x} = \langle \mathrm{mid}\,\boldsymbol{x}, \mathrm{rad}\,\boldsymbol{x} \rangle = \left\langle \frac{\overline{x}+\underline{x}}{2}, \frac{\overline{x}-\underline{x}}{2} \right\rangle$

- addition

$$\boldsymbol{x} + \boldsymbol{y} = \langle \mathrm{mid}\,\boldsymbol{x} + \mathrm{mid}\,\boldsymbol{y}, \mathrm{rad}\,\boldsymbol{x} + \mathrm{rad}\,\boldsymbol{y} \rangle$$

- subtraction

$$\boldsymbol{x} - \boldsymbol{y} = \langle \mathrm{mid}\,\boldsymbol{x} - \mathrm{mid}\,\boldsymbol{y}, \mathrm{rad}\,\boldsymbol{x} + \mathrm{rad}\,\boldsymbol{y} \rangle$$

Midrad Multiplication Formulas

Neumaier 1990, Rump 1999, Nguyen 2011

- multiplication $\boldsymbol{x} \times \boldsymbol{y} = \boldsymbol{z} \subseteq \boldsymbol{z}_1 \subseteq \boldsymbol{z}_2$

	mid $\boldsymbol{z}$	rad $\boldsymbol{z}$
$\boldsymbol{z}$	$\alpha + \mathrm{sign}(\alpha) \times \min\{\beta, \gamma, \delta\}$	$\max\{\beta, \gamma\} + \max\{\min\{\beta, \gamma\}, \delta\}$
$\boldsymbol{z}_1$	$\alpha + \mathrm{sign}(\alpha) \times \min\{\alpha', \beta, \gamma, \delta\}$	$\beta + \gamma + \delta - \min\{\alpha', \beta, \gamma, \delta\}$
$\boldsymbol{z}_2$	α	$\beta + \gamma + \delta$

where

$$
\begin{aligned}
\alpha &= \mathrm{mid}\,\boldsymbol{x} \times \mathrm{mid}\,\boldsymbol{y} \\
\alpha' &= |\mathrm{mid}\,\boldsymbol{x}| \times |\mathrm{mid}\,\boldsymbol{y}| \\
\beta &= |\mathrm{mid}\,\boldsymbol{x}| \times \mathrm{rad}\,\boldsymbol{y} \\
\gamma &= \mathrm{rad}\,\boldsymbol{x} \times |\mathrm{mid}\,\boldsymbol{y}| \\
\delta &= \mathrm{rad}\,\boldsymbol{x} \times \mathrm{rad}\,\boldsymbol{y}
\end{aligned}
$$

Midrad Multiplication Formulas

Neumaier 1990, Rump 1999, Nguyen 2011

- multiplication $\boldsymbol{x} \times \boldsymbol{y} = \boldsymbol{z} \subseteq \boldsymbol{z}_1 \subseteq \boldsymbol{z}_2$

	$\mathrm{mid}\, \boldsymbol{z}$	$\mathrm{rad}\, \boldsymbol{z}$
$\boldsymbol{z}$	$\alpha + \mathrm{sign}(\alpha) \times \min\{\beta, \gamma, \delta\}$	$\max\{\beta, \gamma\} + \max\{\min\{\beta, \gamma\}, \delta\}$
$\boldsymbol{z}_1$	$\alpha + \mathrm{sign}(\alpha) \times \min\{\alpha', \beta, \gamma, \delta\}$	$\beta + \gamma + \delta - \min\{\alpha', \beta, \gamma, \delta\}$
$\boldsymbol{z}_2$	α	$\beta + \gamma + \delta$

where

$$
\begin{aligned}
\alpha &= \mathrm{mid}\, \boldsymbol{x} \times \mathrm{mid}\, \boldsymbol{y} \\
\alpha' &= |\mathrm{mid}\, \boldsymbol{x}| \times |\mathrm{mid}\, \boldsymbol{y}| \\
\beta &= |\mathrm{mid}\, \boldsymbol{x}| \times \mathrm{rad}\, \boldsymbol{y} \\
\gamma &= \mathrm{rad}\, \boldsymbol{x} \times |\mathrm{mid}\, \boldsymbol{y}| \\
\delta &= \mathrm{rad}\, \boldsymbol{x} \times \mathrm{rad}\, \boldsymbol{y}
\end{aligned}
$$

Midrad Multiplication Formulas

Neumaier 1990, Rump 1999, Nguyen 2011

- multiplication $\boldsymbol{x} \times \boldsymbol{y} = \boldsymbol{z} \subseteq \boldsymbol{z}_1 \subseteq \boldsymbol{z}_2$

	$\mathrm{mid}\, \boldsymbol{z}$	$\mathrm{rad}\, \boldsymbol{z}$
$\boldsymbol{z}$	$\alpha + \mathrm{sign}(\alpha) \times \min\{\beta, \gamma, \delta\}$	$\max\{\beta, \gamma\} + \max\{\min\{\beta, \gamma\}, \delta\}$
$\boldsymbol{z}_1$	$\alpha + \mathrm{sign}(\alpha) \times \min\{\alpha', \beta, \gamma, \delta\}$	$\beta + \gamma + \delta - \min\{\alpha', \beta, \gamma, \delta\}$
$\boldsymbol{z}_2$	α	$\beta + \gamma + \delta$

where

$$\begin{aligned}
\alpha &= \mathrm{mid}\, \boldsymbol{x} \times \mathrm{mid}\, \boldsymbol{y} \\
\alpha' &= |\mathrm{mid}\, \boldsymbol{x}| \times |\mathrm{mid}\, \boldsymbol{y}| \\
\beta &= |\mathrm{mid}\, \boldsymbol{x}| \times \mathrm{rad}\, \boldsymbol{y} \\
\gamma &= \mathrm{rad}\, \boldsymbol{x} \times |\mathrm{mid}\, \boldsymbol{y}| \\
\delta &= \mathrm{rad}\, \boldsymbol{x} \times \mathrm{rad}\, \boldsymbol{y}
\end{aligned}$$

Midrad Multiplication Formulas

Neumaier 1990, Rump 1999, Nguyen 2011

- multiplication $\boldsymbol{x} \times \boldsymbol{y} = \boldsymbol{z} \subseteq \boldsymbol{z}_1 \subseteq \boldsymbol{z}_2$

	mid $\boldsymbol{z}$	rad $\boldsymbol{z}$
$\boldsymbol{z}$	$\alpha + \mathrm{sign}(\alpha) \times \min\{\beta, \gamma, \delta\}$	$\max\{\beta, \gamma\} + \max\{\min\{\beta, \gamma\}, \delta\}$
$\boldsymbol{z}_1$	$\alpha + \mathrm{sign}(\alpha) \times \min\{\alpha', \beta, \gamma, \delta\}$	$\beta + \gamma + \delta - \min\{\alpha', \beta, \gamma, \delta\}$
$\boldsymbol{z}_2$	α	$\beta + \gamma + \delta$

where

$$\begin{aligned}
\alpha &= \mathrm{mid}\,\boldsymbol{x} \times \mathrm{mid}\,\boldsymbol{y} \\
\alpha' &= |\mathrm{mid}\,\boldsymbol{x}| \times |\mathrm{mid}\,\boldsymbol{y}| \\
\beta &= |\mathrm{mid}\,\boldsymbol{x}| \times \mathrm{rad}\,\boldsymbol{y} \\
\gamma &= \mathrm{rad}\,\boldsymbol{x} \times |\mathrm{mid}\,\boldsymbol{y}| \\
\delta &= \mathrm{rad}\,\boldsymbol{x} \times \mathrm{rad}\,\boldsymbol{y}
\end{aligned}$$

IIMul4 Algorithm

Rump, *Fast and Parallel Interval Arithmetic*, 1999

Input: $\boldsymbol{A} = [\underline{A}, \overline{A}]$, $\boldsymbol{B} = [\underline{B}, \overline{B}]$

Output: $\boldsymbol{C} \supseteq \boldsymbol{A} \times \boldsymbol{B}$

1: $\langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \leftarrow \text{InfsupToMidrad}(\boldsymbol{A})$

2: $\langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \leftarrow \text{InfsupToMidrad}(\boldsymbol{B})$

3: $R_{\boldsymbol{C}} \leftarrow fl_{\Delta}(|M_{\boldsymbol{A}}| \times R_{\boldsymbol{B}} + R_{\boldsymbol{A}} \times (|M_{\boldsymbol{B}}| + R_{\boldsymbol{B}}))$

4: $\overline{C} \leftarrow fl_{\Delta}(M_{\boldsymbol{A}} \times M_{\boldsymbol{B}} + R_{\boldsymbol{C}})$

5: $\underline{C} \leftarrow fl_{\nabla}(M_{\boldsymbol{A}} \times M_{\boldsymbol{B}} - R_{\boldsymbol{C}})$

6: **return** $[\underline{C}, \overline{C}]$

4 Matrix-Matrix Multiplications

IIMul4 Algorithm

Rump, *Fast and Parallel Interval Arithmetic*, 1999

Input: $\boldsymbol{A} = [\underline{A}, \overline{A}]$, $\boldsymbol{B} = [\underline{B}, \overline{B}]$

Output: $\boldsymbol{C} \supseteq \boldsymbol{A} \times \boldsymbol{B}$

1: $\langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \leftarrow \text{InfsupToMidrad}(\boldsymbol{A})$

2: $\langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \leftarrow \text{InfsupToMidrad}(\boldsymbol{B})$

3: $R_{\boldsymbol{C}} \leftarrow fl_{\Delta}(|M_{\boldsymbol{A}}| \times R_{\boldsymbol{B}} + R_{\boldsymbol{A}} \times (|M_{\boldsymbol{B}}| + R_{\boldsymbol{B}}))$

4: $\overline{C} \leftarrow fl_{\Delta}(M_{\boldsymbol{A}} \times M_{\boldsymbol{B}} + R_{\boldsymbol{C}})$

5: $\underline{C} \leftarrow fl_{\nabla}(M_{\boldsymbol{A}} \times M_{\boldsymbol{B}} - R_{\boldsymbol{C}})$

6: **return** $[\underline{C}, \overline{C}]$

4 Matrix-Matrix Multiplications

`IIMul4` Algorithm

Rump, *Fast and Parallel Interval Arithmetic*, 1999

Input: $\boldsymbol{A} = [\underline{A}, \overline{A}]$, $\boldsymbol{B} = [\underline{B}, \overline{B}]$

Output: $\boldsymbol{C} \supseteq \boldsymbol{A} \times \boldsymbol{B}$

1: $\langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \leftarrow \text{InfsupToMidrad}(\boldsymbol{A})$

2: $\langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \leftarrow \text{InfsupToMidrad}(\boldsymbol{B})$

3: $R_{\boldsymbol{C}} \leftarrow fl_{\triangle}(|M_{\boldsymbol{A}}| \times R_{\boldsymbol{B}} + R_{\boldsymbol{A}} \times (|M_{\boldsymbol{B}}| + R_{\boldsymbol{B}}))$

4: $\overline{C} \leftarrow fl_{\triangle}(M_{\boldsymbol{A}} \times M_{\boldsymbol{B}} + R_{\boldsymbol{C}})$

5: $\underline{C} \leftarrow fl_{\nabla}(M_{\boldsymbol{A}} \times M_{\boldsymbol{B}} - R_{\boldsymbol{C}})$

6: **return** $[\underline{C}, \overline{C}]$

4 Matrix-Matrix Multiplications

IIMul7 Algorithm

Nguyen, *Efficient algorithms for verified scientific computing: numerical linear algebra using interval arithmetic*, 2011

Input: $\mathbf{A} = [\underline{A}, \overline{A}]$, $\mathbf{B} = [\underline{B}, \overline{B}]$
Output: $\mathbf{C} \supseteq \mathbf{A} \times \mathbf{B}$
1: $\langle M_{\mathbf{A}}, R_{\mathbf{A}} \rangle \leftarrow \text{InfsupToMidrad}(\mathbf{A})$
2: $\langle M_{\mathbf{B}}, R_{\mathbf{B}} \rangle \leftarrow \text{InfsupToMidrad}(\mathbf{B})$
3: $\rho_{\mathbf{A}} \leftarrow \mathrm{sign}(M_{\mathbf{A}}) .* \min(|M_{\mathbf{A}}|, R_{\mathbf{A}})$
4: $\rho_{\mathbf{B}} \leftarrow \mathrm{sign}(M_{\mathbf{B}}) .* \min(|M_{\mathbf{B}}|, R_{\mathbf{B}})$
5: $R_{\mathbf{C}} \leftarrow$
$fl_{\Delta}\left(|M_{\mathbf{A}}| \times R_{\mathbf{B}} + R_{\mathbf{A}} \times (|M_{\mathbf{B}}| + R_{\mathbf{B}}) + (-|\rho_{\mathbf{A}}|) \times |\rho_{\mathbf{B}}|\right)$
6: $\overline{\mathbf{C}} \leftarrow fl_{\Delta}\left(M_{\mathbf{A}} \times M_{\mathbf{B}} + \rho_{\mathbf{A}} \times \rho_{\mathbf{B}} + R_{\mathbf{C}}\right)$
7: $\underline{\mathbf{C}} \leftarrow fl_{\nabla}\left(M_{\mathbf{A}} \times M_{\mathbf{B}} + \rho_{\mathbf{A}} \times \rho_{\mathbf{B}} - R_{\mathbf{C}}\right)$
8: **return** $[\underline{\mathbf{C}}, \overline{\mathbf{C}}]$

7 Matrix-Matrix Multiplications

`IIMul7` Algorithm

Nguyen, *Efficient algorithms for verified scientific computing: numerical linear algebra using interval arithmetic*, 2011

Input: $\mathbf{A} = [\underline{A}, \overline{A}]$, $\mathbf{B} = [\underline{B}, \overline{B}]$
Output: $\mathbf{C} \supseteq \mathbf{A} \times \mathbf{B}$

1: $\langle M_{\mathbf{A}}, R_{\mathbf{A}} \rangle \leftarrow \mathrm{InfsupToMidrad}(\mathbf{A})$
2: $\langle M_{\mathbf{B}}, R_{\mathbf{B}} \rangle \leftarrow \mathrm{InfsupToMidrad}(\mathbf{B})$
3: $\rho_{\mathbf{A}} \leftarrow \mathrm{sign}(M_{\mathbf{A}}) .* \min(|M_{\mathbf{A}}|, R_{\mathbf{A}})$
4: $\rho_{\mathbf{B}} \leftarrow \mathrm{sign}(M_{\mathbf{B}}) .* \min(|M_{\mathbf{B}}|, R_{\mathbf{B}})$
5: $R_{\mathbf{C}} \leftarrow$ $fl_{\triangle}\left(|M_{\mathbf{A}}| \times R_{\mathbf{B}} + R_{\mathbf{A}} \times (|M_{\mathbf{B}}| + R_{\mathbf{B}}) + (-|\rho_{\mathbf{A}}|) \times |\rho_{\mathbf{B}}|\right)$
6: $\overline{\mathbf{C}} \leftarrow fl_{\triangle}\left(M_{\mathbf{A}} \times M_{\mathbf{B}} + \rho_{\mathbf{A}} \times \rho_{\mathbf{B}} + R_{\mathbf{C}}\right)$
7: $\underline{\mathbf{C}} \leftarrow fl_{\triangledown}\left(M_{\mathbf{A}} \times M_{\mathbf{B}} + \rho_{\mathbf{A}} \times \rho_{\mathbf{B}} - R_{\mathbf{C}}\right)$
8: **return** $[\underline{\mathbf{C}}, \overline{\mathbf{C}}]$

7 Matrix-Matrix Multiplications

IIMul7 Algorithm

Nguyen, *Efficient algorithms for verified scientific computing: numerical linear algebra using interval arithmetic*, 2011

Input: $\mathbf{A} = [\underline{A}, \overline{A}]$, $\mathbf{B} = [\underline{B}, \overline{B}]$

Output: $\mathbf{C} \supseteq \mathbf{A} \times \mathbf{B}$

$\langle M_{\mathbf{A}}, R_{\mathbf{A}} \rangle \leftarrow \text{InfsupToMidrad}(\mathbf{A})$

$\langle M_{\mathbf{B}}, R_{\mathbf{B}} \rangle \leftarrow \text{InfsupToMidrad}(\mathbf{B})$

$\rho_{\mathbf{A}} \leftarrow \text{sign}(M_{\mathbf{A}}).* \min(|M_{\mathbf{A}}|, R_{\mathbf{A}})$

$\rho_{\mathbf{B}} \leftarrow \text{sign}(M_{\mathbf{B}}).* \min(|M_{\mathbf{B}}|, R_{\mathbf{B}})$

$R_{\mathbf{C}} \leftarrow$
$fl_{\Delta}\left(|M_{\mathbf{A}}| \times R_{\mathbf{B}} + R_{\mathbf{A}} \times (|M_{\mathbf{B}}| + R_{\mathbf{B}}) + (-|\rho_{\mathbf{A}}|) \times |\rho_{\mathbf{B}}|\right)$

$\overline{\mathbf{C}} \leftarrow fl_{\Delta}\left(M_{\mathbf{A}} \times M_{\mathbf{B}} + \rho_{\mathbf{A}} \times \rho_{\mathbf{B}} + R_{\mathbf{C}}\right)$

$\underline{\mathbf{C}} \leftarrow fl_{\nabla}\left(M_{\mathbf{A}} \times M_{\mathbf{B}} + \rho_{\mathbf{A}} \times \rho_{\mathbf{B}} - R_{\mathbf{C}}\right)$

return $[\underline{\mathbf{C}}, \overline{\mathbf{C}}]$

7 Matrix-Matrix Multiplications

IIMul7 Algorithm

Nguyen, *Efficient algorithms for verified scientific computing: numerical linear algebra using interval arithmetic*, 2011

Input: $\boldsymbol{A} = [\underline{A}, \overline{A}]$, $\boldsymbol{B} = [\underline{B}, \overline{B}]$

Output: $\boldsymbol{C} \supseteq \boldsymbol{A} \times \boldsymbol{B}$

1: $\langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \leftarrow \mathrm{InfsupToMidrad}(\boldsymbol{A})$

2: $\langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \leftarrow \mathrm{InfsupToMidrad}(\boldsymbol{B})$

3: $\rho_{\boldsymbol{A}} \leftarrow \mathrm{sign}(M_{\boldsymbol{A}}) .* \min(|M_{\boldsymbol{A}}|, R_{\boldsymbol{A}})$

4: $\rho_{\boldsymbol{B}} \leftarrow \mathrm{sign}(M_{\boldsymbol{B}}) .* \min(|M_{\boldsymbol{B}}|, R_{\boldsymbol{B}})$

5: $R_{\boldsymbol{C}} \leftarrow$
$fl_{\triangle}\left(|M_{\boldsymbol{A}}| \times R_{\boldsymbol{B}} + R_{\boldsymbol{A}} \times (|M_{\boldsymbol{B}}| + R_{\boldsymbol{B}}) + (-|\rho_{\boldsymbol{A}}|) \times |\rho_{\boldsymbol{B}}|\right)$

6: $\overline{C} \leftarrow fl_{\triangle}\left(M_{\boldsymbol{A}} \times M_{\boldsymbol{B}} + \rho_{\boldsymbol{A}} \times \rho_{\boldsymbol{B}} + R_{\boldsymbol{C}}\right)$

7: $\underline{C} \leftarrow fl_{\nabla}\left(M_{\boldsymbol{A}} \times M_{\boldsymbol{B}} + \rho_{\boldsymbol{A}} \times \rho_{\boldsymbol{B}} - R_{\boldsymbol{C}}\right)$

8: **return** $[\underline{C}, \overline{C}]$

The same operations are computed twice but with different rounding modes.

Bound on the Error of a Matrix Product in Rounding to Nearest

Rump, *Error estimation of floating-point summation and dot product*, 2011

Theorem

Let $A \in \mathbb{F}^{m \times k}$ *and* $B \in \mathbb{F}^{k \times n}$ *with* $2(k+2)u \leq 1$ *be given, and let* $C = fl_{\square}(A \times B)$ *and* $\Gamma = fl_{\square}(|A| \times |B|)$*. Here* C *may be computed in any order, and we assume that* Γ *is computed in the same order. Then*

$$|fl_{\square}(A \times B) - A \times B| \leq fl_{\square}\left(\frac{k+2}{2}\mathrm{ulp}(\Gamma) + \frac{1}{2}u^{-1}\eta\right)$$

Bound on the Error of a Matrix Product in Rounding to Nearest

Rump, *Error estimation of floating-point summation and dot product*, 2011

Theorem

Let $A \in \mathbb{F}^{m\times k}$ and $B \in \mathbb{F}^{k\times n}$ with $2(k+2)u \le 1$ be given, and let $C = fl_{\square}(A \times B)$ and $\Gamma = fl_{\square}(|A| \times |B|)$. Here C may be computed in any order, and we assume that Γ is computed in the same order. Then

$$|fl_{\square}(A \times B) - A \times B| \le fl_{\square}\left(\frac{k+2}{2}\mathrm{ulp}(\Gamma) + \frac{1}{2}u^{-1}\eta\right)$$

`MMMul3` Algorithm

Rump, *Fast Interval Matrix Multiplication*, 2011

Input: $\boldsymbol{A} = \langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \in \mathbb{IF}^{m \times k}$, $\boldsymbol{B} = \langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \in \mathbb{IF}^{k \times n}$

Output: $\boldsymbol{C} \supseteq \boldsymbol{A} \times \boldsymbol{B}$

1: $M_{\boldsymbol{C}} \leftarrow fl_{\square}(M_{\boldsymbol{A}} \times M_{\boldsymbol{B}})$

2: $R'_{\boldsymbol{B}} \leftarrow fl_{\Delta}((k+2)\mathtt{u}|M_{\boldsymbol{B}}| + R_{\boldsymbol{B}})$

3: $R_{\boldsymbol{C}} \leftarrow fl_{\Delta}(|M_{\boldsymbol{A}}| \times R'_{\boldsymbol{B}} + \mathtt{u}^{-1}\eta + R_{\boldsymbol{A}} \times (|M_{\boldsymbol{B}}| + R_{\boldsymbol{B}}))$

4: **return** $\langle M_{\boldsymbol{C}}, R_{\boldsymbol{C}} \rangle$

3 Matrix-Matrix Multiplications

MMMul3 Algorithm

Rump, *Fast Interval Matrix Multiplication*, 2011

Input: $\boldsymbol{A} = \langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \in \mathbb{IF}^{m \times k}$, $\boldsymbol{B} = \langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \in \mathbb{IF}^{k \times n}$
Output: $\boldsymbol{C} \supseteq \boldsymbol{A} \times \boldsymbol{B}$

1: $M_{\boldsymbol{C}} \leftarrow fl_{\square}(M_{\boldsymbol{A}} \times M_{\boldsymbol{B}})$
2: $R'_{\boldsymbol{B}} \leftarrow fl_{\Delta}((k+2)\mathrm{u}|M_{\boldsymbol{B}}| + R_{\boldsymbol{B}})$
3: $R_{\boldsymbol{C}} \leftarrow fl_{\Delta}(|M_{\boldsymbol{A}}| \times R'_{\boldsymbol{B}} + \mathrm{u}^{-1}\eta + R_{\boldsymbol{A}} \times (|M_{\boldsymbol{B}}| + R_{\boldsymbol{B}}))$
4: **return** $\langle M_{\boldsymbol{C}}, R_{\boldsymbol{C}} \rangle$

3 Matrix-Matrix Multiplications

MMMul5 Algorithm

Rump, *Fast Interval Matrix Multiplication*, 2011

Input: $\boldsymbol{A} = \langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \in \mathbb{IF}^{m\times k}$, $\boldsymbol{B} = \langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \in \mathbb{IF}^{k\times n}$

Output: $\boldsymbol{C} \supseteq \boldsymbol{A} \times \boldsymbol{B}$

1: $\rho_{\boldsymbol{A}} \leftarrow \mathrm{sign}(M_{\boldsymbol{A}}).*\min(|M_{\boldsymbol{A}}|, R_{\boldsymbol{A}})$

2: $\rho_{\boldsymbol{B}} \leftarrow \mathrm{sign}(M_{\boldsymbol{B}}).*\min(|M_{\boldsymbol{B}}|, R_{\boldsymbol{B}})$

3: $M_{\boldsymbol{C}} \leftarrow fl_{\square}\left(M_{\boldsymbol{A}} \times M_{\boldsymbol{B}} + \rho_{\boldsymbol{A}} \times \rho_{\boldsymbol{B}}\right)$

4: $\Gamma \leftarrow fl_{\square}\left(|M_{\boldsymbol{A}}| \times |M_{\boldsymbol{B}}| + |\rho_{\boldsymbol{A}}| \times |\rho_{\boldsymbol{B}}|\right)$

5: $\gamma \leftarrow fl_{\Delta}\left((k+1)\mathrm{ulp}(\Gamma) + \frac{1}{2}\mathrm{u}^{-1}\eta\right)$

6: $R_{\boldsymbol{C}} \leftarrow fl_{\Delta}\left(((|M_{\boldsymbol{A}}| + R_{\boldsymbol{A}}) \times (|M_{\boldsymbol{B}}| + R_{\boldsymbol{B}}) - \Gamma + 2\gamma\right)$

7: **return** $\langle M_{\boldsymbol{C}}, R_{\boldsymbol{C}} \rangle$

5 Matrix-Matrix Multiplications

MMMul5 Algorithm

Rump, *Fast Interval Matrix Multiplication*, 2011

Input: $\boldsymbol{A} = \langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \in \mathbb{IF}^{m \times k}$, $\boldsymbol{B} = \langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \in \mathbb{IF}^{k \times n}$
Output: $\boldsymbol{C} \supseteq \boldsymbol{A} \times \boldsymbol{B}$

1: $\rho_{\boldsymbol{A}} \leftarrow \mathrm{sign}(M_{\boldsymbol{A}}) .* \min(|M_{\boldsymbol{A}}|, R_{\boldsymbol{A}})$
2: $\rho_{\boldsymbol{B}} \leftarrow \mathrm{sign}(M_{\boldsymbol{B}}) .* \min(|M_{\boldsymbol{B}}|, R_{\boldsymbol{B}})$
3: $M_{\boldsymbol{C}} \leftarrow fl_{\square}\left(M_{\boldsymbol{A}} \times M_{\boldsymbol{B}} + \rho_{\boldsymbol{A}} \times \rho_{\boldsymbol{B}}\right)$
4: $\Gamma \leftarrow fl_{\square}\left(|M_{\boldsymbol{A}}| \times |M_{\boldsymbol{B}}| + |\rho_{\boldsymbol{A}}| \times |\rho_{\boldsymbol{B}}|\right)$
5: $\gamma \leftarrow fl_{\Delta}\left((k+1)\mathrm{ulp}(\Gamma) + \frac{1}{2}\mathrm{u}^{-1}\eta\right)$
6: $R_{\boldsymbol{C}} \leftarrow fl_{\Delta}\left((|M_{\boldsymbol{A}}| + R_{\boldsymbol{A}}) \times (|M_{\boldsymbol{B}}| + R_{\boldsymbol{B}}) - \Gamma + 2\gamma\right)$
7: **return** $\langle M_{\boldsymbol{C}}, R_{\boldsymbol{C}} \rangle$

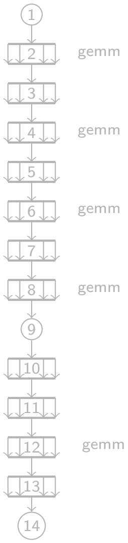
5 Matrix-Matrix Multiplications

Content

`MMMul5` Implementation with Level-3 BLAS

Input: $\boldsymbol{A} = \langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \in \mathbb{IF}^{m \times k}$, $\boldsymbol{B} = \langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \in \mathbb{IF}^{k \times n}$

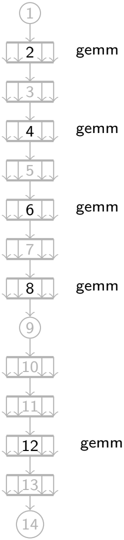
1: set rounding mode to nearest
2: $M_{\boldsymbol{C}} \leftarrow M_{\boldsymbol{A}} \times M_{\boldsymbol{B}}$
3: $T_1 \leftarrow |M_{\boldsymbol{A}}|$; $T_2 \leftarrow |M_{\boldsymbol{B}}|$
4: $R_{\boldsymbol{C}} \leftarrow T_1 \times T_2$
5: $T_3 \leftarrow Compute(\rho_{\boldsymbol{A}})$; $T_4 \leftarrow Compute(\rho_{\boldsymbol{B}})$
6: $M_{\boldsymbol{C}} \leftarrow T_3 \times T_4 + M_{\boldsymbol{C}}$
7: $T_5 \leftarrow |T_3|$; $T_6 \leftarrow |T_4|$
8: $R_{\boldsymbol{C}} \leftarrow T_5 \times T_6 + R_{\boldsymbol{C}}$
9: set rounding mode towards $+\infty$
10: $T_7 \leftarrow (k+1)\mathrm{ulp}(R_{\boldsymbol{C}}) + 0.5\mathrm{u}^{-1}\eta$
11: $T_8 \leftarrow T_1 + R_{\boldsymbol{A}}$; $T_9 \leftarrow T_2 + R_{\boldsymbol{B}}$
12: $R_{\boldsymbol{C}} \leftarrow T_8 \times T_9 - R_{\boldsymbol{C}}$
13: $R_{\boldsymbol{C}} \leftarrow R_{\boldsymbol{C}} + 2T_7$
14: **return** $\langle M_{\boldsymbol{C}}, R_{\boldsymbol{C}} \rangle$



MMMul5 Implementation with Level-3 BLAS

Input: $\boldsymbol{A} = \langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \in \mathbb{IF}^{m \times k}$, $\boldsymbol{B} = \langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \in \mathbb{IF}^{k \times n}$

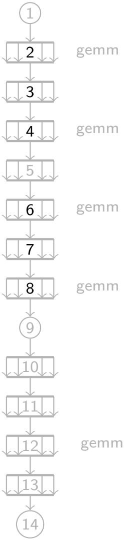
set rounding mode to nearest
$M_{\boldsymbol{C}} \leftarrow M_{\boldsymbol{A}} \times M_{\boldsymbol{B}}$
$T_1 \leftarrow |M_{\boldsymbol{A}}|$; $T_2 \leftarrow |M_{\boldsymbol{B}}|$
$R_{\boldsymbol{C}} \leftarrow T_1 \times T_2$
$T_3 \leftarrow Compute(\rho_{\boldsymbol{A}})$; $T_4 \leftarrow Compute(\rho_{\boldsymbol{B}})$
$M_{\boldsymbol{C}} \leftarrow T_3 \times T_4 + M_{\boldsymbol{C}}$
$T_5 \leftarrow |T_3|$; $T_6 \leftarrow |T_4|$
$R_{\boldsymbol{C}} \leftarrow T_5 \times T_6 + R_{\boldsymbol{C}}$
set rounding mode towards $+\infty$
$T_7 \leftarrow (k+1)\mathrm{ulp}(R_{\boldsymbol{C}}) + 0.5\mathrm{u}^{-1}\eta$
$T_8 \leftarrow T_1 + R_{\boldsymbol{A}}$; $T_9 \leftarrow T_2 + R_{\boldsymbol{B}}$
$R_{\boldsymbol{C}} \leftarrow T_8 \times T_9 - R_{\boldsymbol{C}}$
$R_{\boldsymbol{C}} \leftarrow R_{\boldsymbol{C}} + 2T_7$
return $\langle M_{\boldsymbol{C}}, R_{\boldsymbol{C}} \rangle$



MMMul5 Implementation with Level-3 BLAS

Input: $A = \langle M_A, R_A \rangle \in \mathbb{IF}^{m\times k}$, $B = \langle M_B, R_B \rangle \in \mathbb{IF}^{k\times n}$

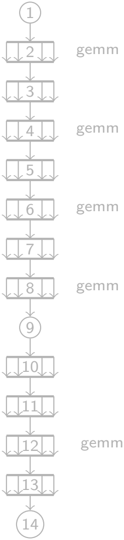
1: set rounding mode to nearest
2: $M_C \leftarrow M_A \times M_B$
3: $T_1 \leftarrow |M_A|$; $T_2 \leftarrow |M_B|$
4: $R_C \leftarrow T_1 \times T_2$
5: $T_3 \leftarrow Compute(\rho_A)$; $T_4 \leftarrow Compute(\rho_B)$
6: $M_C \leftarrow T_3 \times T_4 + M_C$
7: $T_5 \leftarrow |T_3|$; $T_6 \leftarrow |T_4|$
8: $R_C \leftarrow T_5 \times T_6 + R_C$
9: set rounding mode towards $+\infty$
10: $T_7 \leftarrow (k+1)\mathrm{ulp}(R_C) + 0.5\mathrm{u}^{-1}\eta$
11: $T_8 \leftarrow T_1 + R_A$; $T_9 \leftarrow T_2 + R_B$
12: $R_C \leftarrow T_8 \times T_9 - R_C$
13: $R_C \leftarrow R_C + 2T_7$
14: **return** $\langle M_C, R_C \rangle$



MMMul5 Implementation with Level-3 BLAS

Input: $\boldsymbol{A} = \langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \in \mathbb{IF}^{m\times k}$, $\boldsymbol{B} = \langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \in \mathbb{IF}^{k\times n}$

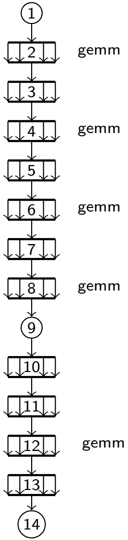
1: set rounding mode to nearest
2: $M_{\boldsymbol{C}} \leftarrow M_{\boldsymbol{A}} \times M_{\boldsymbol{B}}$
3: $T_1 \leftarrow |M_{\boldsymbol{A}}|$; $T_2 \leftarrow |M_{\boldsymbol{B}}|$
4: $R_{\boldsymbol{C}} \leftarrow T_1 \times T_2$
5: $T_3 \leftarrow Compute(\rho_{\boldsymbol{A}})$; $T_4 \leftarrow Compute(\rho_{\boldsymbol{B}})$
6: $M_{\boldsymbol{C}} \leftarrow T_3 \times T_4 + M_{\boldsymbol{C}}$
7: $T_5 \leftarrow |T_3|$; $T_6 \leftarrow |T_4|$
8: $R_{\boldsymbol{C}} \leftarrow T_5 \times T_6 + R_{\boldsymbol{C}}$
9: set rounding mode towards $+\infty$
10: $T_7 \leftarrow (k+1)\mathrm{ulp}(R_{\boldsymbol{C}}) + 0.5\mathrm{u}^{-1}\eta$
11: $T_8 \leftarrow T_1 + R_{\boldsymbol{A}}$; $T_9 \leftarrow T_2 + R_{\boldsymbol{B}}$
12: $R_{\boldsymbol{C}} \leftarrow T_8 \times T_9 - R_{\boldsymbol{C}}$
13: $R_{\boldsymbol{C}} \leftarrow R_{\boldsymbol{C}} + 2T_7$
14: **return** $\langle M_{\boldsymbol{C}}, R_{\boldsymbol{C}} \rangle$



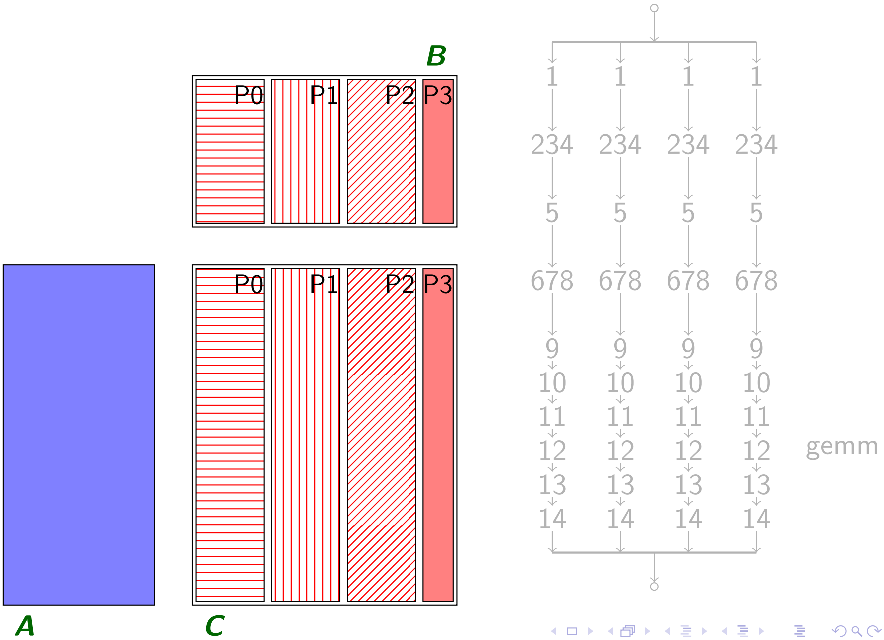
`MMMul5` Implementation with Level-3 BLAS

Input: $\boldsymbol{A} = \langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \in \mathbb{IF}^{m \times k}$, $\boldsymbol{B} = \langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \in \mathbb{IF}^{k \times n}$

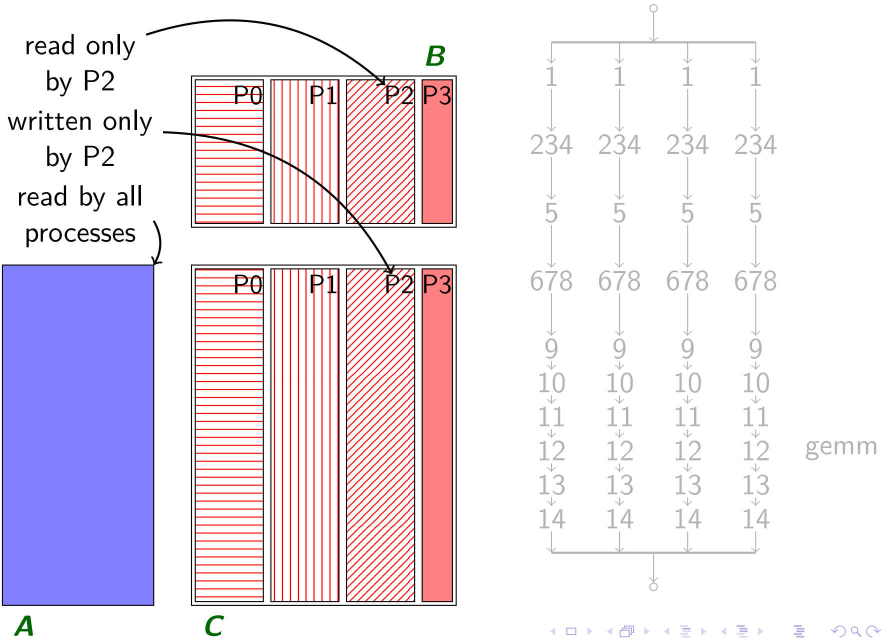
set rounding mode to nearest
$M_{\boldsymbol{C}} \leftarrow M_{\boldsymbol{A}} \times M_{\boldsymbol{B}}$
$T_1 \leftarrow |M_{\boldsymbol{A}}|; \; T_2 \leftarrow |M_{\boldsymbol{B}}|$
$R_{\boldsymbol{C}} \leftarrow T_1 \times T_2$
$T_3 \leftarrow Compute(\rho_{\boldsymbol{A}}); \; T_4 \leftarrow Compute(\rho_{\boldsymbol{B}})$
$M_{\boldsymbol{C}} \leftarrow T_3 \times T_4 + M_{\boldsymbol{C}}$
$T_5 \leftarrow |T_3| \, ; T_6 \leftarrow |T_4|$
$R_{\boldsymbol{C}} \leftarrow T_5 \times T_6 + R_{\boldsymbol{C}}$
set rounding mode towards $+\infty$
$T_7 \leftarrow (k+1)\mathrm{ulp}(R_{\boldsymbol{C}}) + 0.5\mathrm{u}^{-1}\eta$
$T_8 \leftarrow T_1 + R_{\boldsymbol{A}}; \; T_9 \leftarrow T_2 + R_{\boldsymbol{B}}$
$R_{\boldsymbol{C}} \leftarrow T_8 \times T_9 - R_{\boldsymbol{C}}$
$R_{\boldsymbol{C}} \leftarrow R_{\boldsymbol{C}} + 2T_7$
return $\langle M_{\boldsymbol{C}}, R_{\boldsymbol{C}} \rangle$



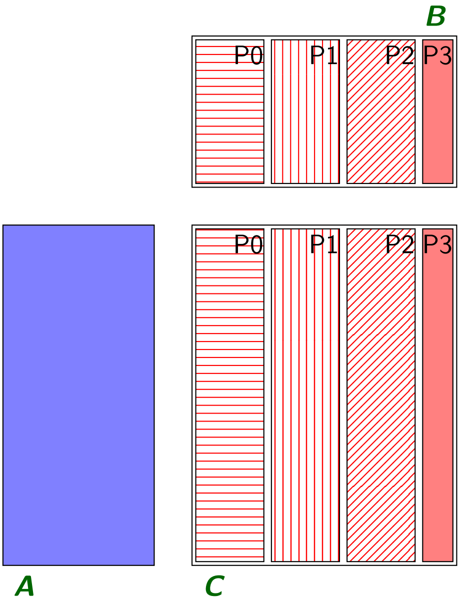
MMMul5: Blocked Version

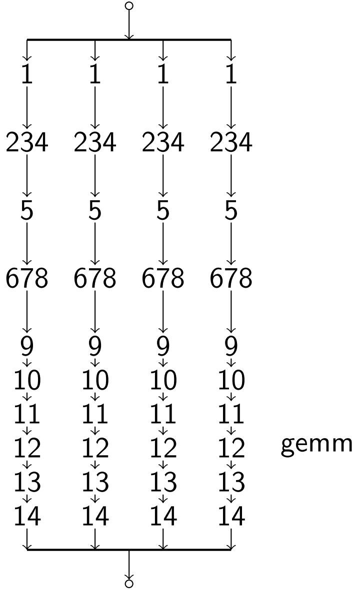


MMMul5: Blocked Version

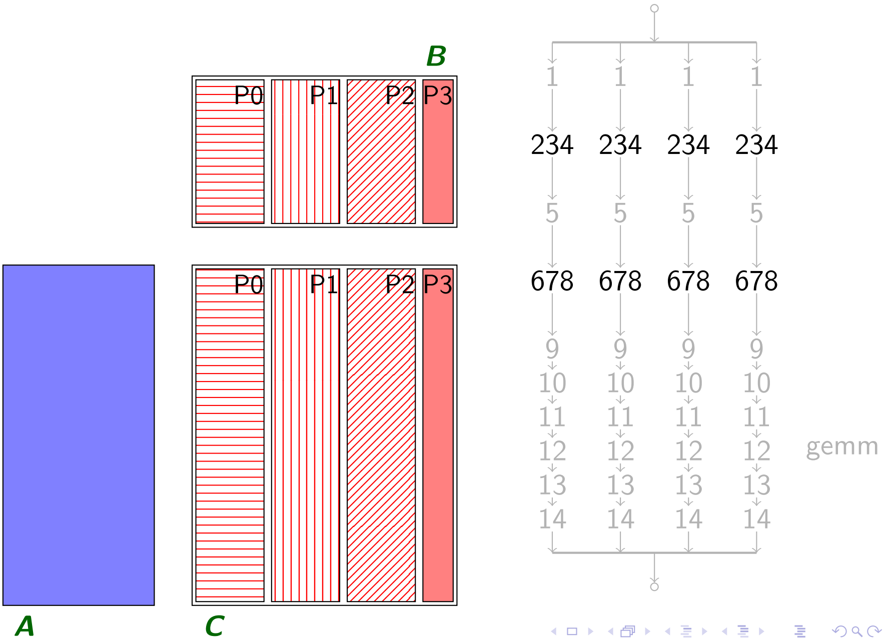


`MMMul5`: Blocked Version

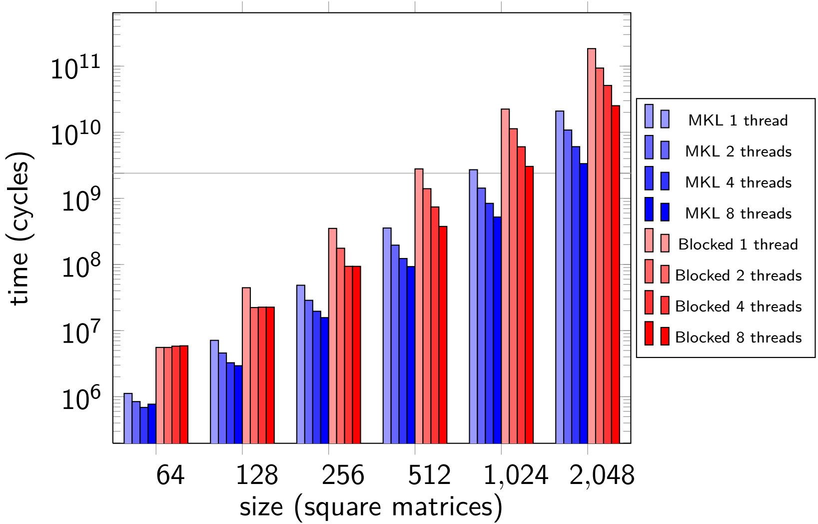




MMMul5: Blocked Version



Execution Time – `MMMul5` BLAS vs blocked versions



Minimum time for product of two square matrices – Intel Bi-Xeon E5620, 2 x Quad-cores 2,40 GHz – Intel Math Kernel Library version 10.3 – gcc version 4.7.1

Content

Notations

Let $\boldsymbol{A} = \langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \in \mathbb{IR}^{m \times k}$, and $B = \langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \in \mathbb{IR}^{k \times n}$ two interval matrices.

We note

- the exact $\boldsymbol{A} \times \boldsymbol{B}$: $\boldsymbol{C} = \langle M_{\boldsymbol{C}}, R_{\boldsymbol{C}} \rangle$
- $\boldsymbol{A} \times \boldsymbol{B}$ computed with `MMMul5`: $\boldsymbol{C}_5 = \langle M_{\boldsymbol{C}_5}, R_{\boldsymbol{C}_5} \rangle$
- $\boldsymbol{A} \times \boldsymbol{B}$ computed with `MMMul3`: $\boldsymbol{C}_3 = \langle M_{\boldsymbol{C}_3}, R_{\boldsymbol{C}_3} \rangle$

As $\boldsymbol{C}_5 \supseteq \boldsymbol{C}$, we have

$$\left| M_{\boldsymbol{C}_5} - M_{\boldsymbol{C}} \right| \leq R_{\boldsymbol{C}_5} - R_{\boldsymbol{C}}.$$

Thus, the radius error accounts for the midpoint drift error.

Notations

Let $\boldsymbol{A} = \langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \in \mathbb{IR}^{m \times k}$, and $B = \langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \in \mathbb{IR}^{k \times n}$ two interval matrices.
We note

- the exact $\boldsymbol{A} \times \boldsymbol{B}$: $\boldsymbol{C} = \langle M_{\boldsymbol{C}}, R_{\boldsymbol{C}} \rangle$
- $\boldsymbol{A} \times \boldsymbol{B}$ computed with `MMMul5`: $\boldsymbol{C}_5 = \langle M_{\boldsymbol{C}_5}, R_{\boldsymbol{C}_5} \rangle$
- $\boldsymbol{A} \times \boldsymbol{B}$ computed with `MMMul3`: $\boldsymbol{C}_3 = \langle M_{\boldsymbol{C}_3}, R_{\boldsymbol{C}_3} \rangle$

As $\boldsymbol{C}_5 \supseteq \boldsymbol{C}$, we have

$$\left| M_{\boldsymbol{C}_5} - M_{\boldsymbol{C}} \right| \leq R_{\boldsymbol{C}_5} - R_{\boldsymbol{C}}.$$

Thus, the radius error accounts for the midpoint drift error.

Notations

Let $\boldsymbol{A} = \langle M_{\boldsymbol{A}}, R_{\boldsymbol{A}} \rangle \in \mathbb{IR}^{m\times k}$, and $B = \langle M_{\boldsymbol{B}}, R_{\boldsymbol{B}} \rangle \in \mathbb{IR}^{k\times n}$ two interval matrices.
We note

- the exact $\boldsymbol{A} \times \boldsymbol{B}$: $\boldsymbol{C} = \langle M_{\boldsymbol{C}}, R_{\boldsymbol{C}} \rangle$
- $\boldsymbol{A} \times \boldsymbol{B}$ computed with `MMMul5`: $\boldsymbol{C}_5 = \langle M_{\boldsymbol{C}_5}, R_{\boldsymbol{C}_5} \rangle$
- $\boldsymbol{A} \times \boldsymbol{B}$ computed with `MMMul3`: $\boldsymbol{C}_3 = \langle M_{\boldsymbol{C}_3}, R_{\boldsymbol{C}_3} \rangle$

As $\boldsymbol{C}_5 \supseteq \boldsymbol{C}$, we have

$$\left| M_{\boldsymbol{C}_5} - M_{\boldsymbol{C}} \right| \leq R_{\boldsymbol{C}_5} - R_{\boldsymbol{C}}.$$

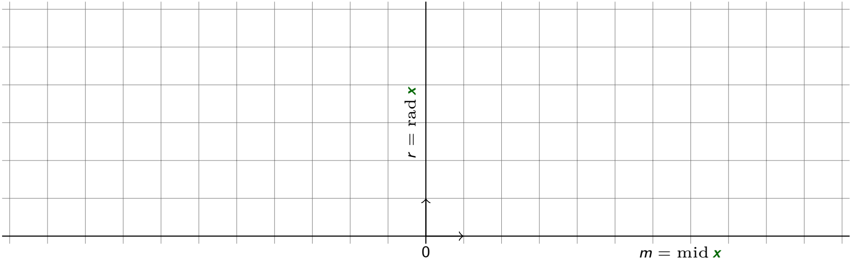
Thus, the radius error accounts for the midpoint drift error.

Relative Precision

Definition An interval $\boldsymbol{x} = \langle m, r \rangle$ is said to be of *relative precision* e if $r \leq e \cdot |m|$.

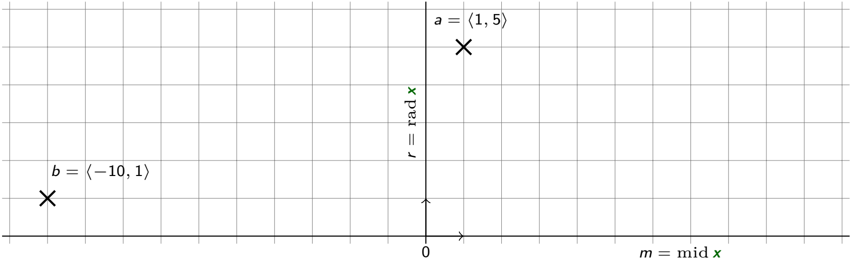
Relative Precision

Definition An interval $\boldsymbol{x} = \langle m, r \rangle$ is said to be of *relative precision* e if $r \le e \cdot |m|$.



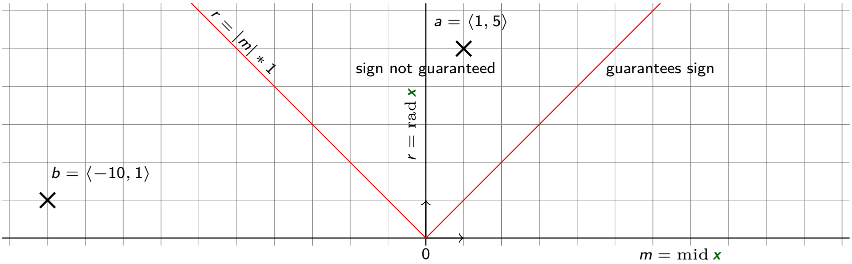
Relative Precision

Definition An interval $\boldsymbol{x} = \langle m, r \rangle$ is said to be of *relative precision* e if $r \leq e \cdot |m|$.



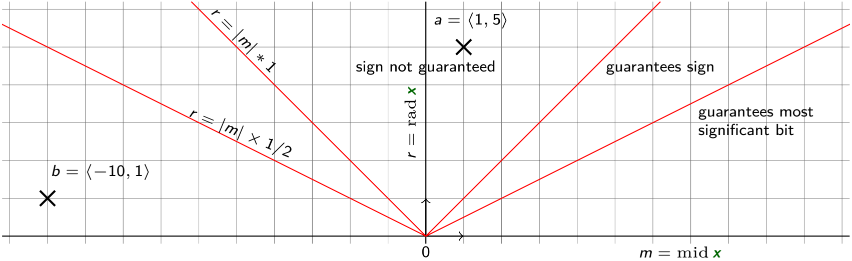
Relative Precision

Definition An interval $\boldsymbol{x} = \langle m, r \rangle$ is said to be of *relative precision* e if $r \leq e \cdot |m|$.



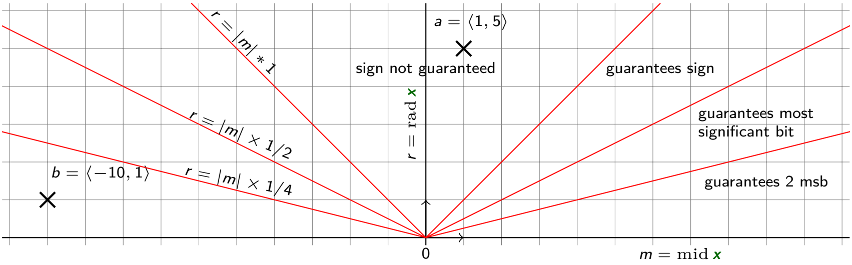
Relative Precision

Definition An interval $\boldsymbol{x} = \langle m, r \rangle$ is said to be of *relative precision* e if $r \leq e \cdot |m|$.



Relative Precision

Definition An interval $\boldsymbol{x} = \langle m, r \rangle$ is said to be of *relative precision* e if $r \leq e \cdot |m|$.



Radius of the Product of Matrices of Fixed Relative Precisions

Proposition

Let $\boldsymbol{A}$ *with fixed relative precision* e *and* $\boldsymbol{B}$ *with fixed relative precision* f.
Then the radii of the approximations of the product $\boldsymbol{A} \times \boldsymbol{B}$ *verify*

$$\begin{aligned} R_{\boldsymbol{C}} &= (\max_1 + \max_2) & |M_{\boldsymbol{A}}| \times |M_{\boldsymbol{B}}| \\ R_{\boldsymbol{C}_5} &= (\max_1 + \max_2 + \min_3 - \min_4) & |M_{\boldsymbol{A}}| \times |M_{\boldsymbol{B}}| \\ R_{\boldsymbol{C}_3} &= (\max_1 + \max_2 + \min_3) & |M_{\boldsymbol{A}}| \times |M_{\boldsymbol{B}}| \end{aligned}$$

where

$$\begin{aligned} \max_1 &= \max\{e, f\} \\ \max_2 &= \max\{\min\{e, f\}, ef\} \\ \min_3 &= \min\{e, f, ef\} \\ \min_4 &= \min\{1, e, f, ef\} \end{aligned}$$

Relative Radius Errors

The relative radius error for C_5 is

$$E_5 = \frac{R_{C_5} - R_C}{R_C} = \begin{cases} 0 & \text{if } e \leq 1 \text{ or } f \leq 1 \\ \frac{\min\{e,f\}-1}{\max\{e,f\}+ef} & \text{otherwise} \end{cases}$$

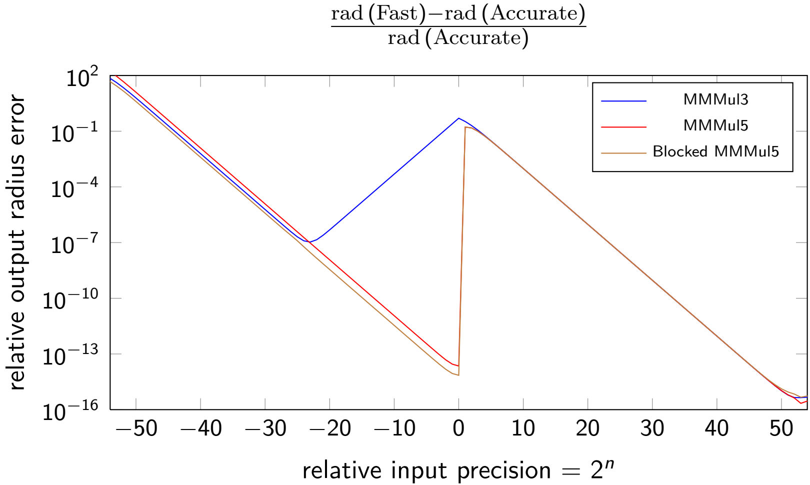
So, $E_5 \leq 3 - 2\sqrt{2} \leq 0.18$ and maximum is reached for $e = f = 1 + \sqrt{2}$.

The relative radius error for C_3 is

$$E_3 = \frac{R_{C_3} - R_C}{R_C} = \frac{\mathrm{min}_3}{\mathrm{max}_1 + \mathrm{max}_2}$$

So, $E_3 \leq 0.5$ and maximum is reached for $e = f = 1$.

Relative Radius Error – Numerical Experiment



Median relative precision of 10 products of 100×100 matrices with random midpoint (standard normal distribution) and fixed relative precision 2^{-k}. Floating-point number are in double precision.

Content

Conclusion: trading off efficiency for accuracy, still being efficient

A panel of formulas exist for operations in interval arithmetic using mid-rad representation:

- trade off accuracy for efficiency, but not always;
- reduce drastically the number of rounding mode changes;
- allow to resort to Level-3 BLAS.

Accuracy:

Floating-point error dominates for input intervals with small relative precision.
Rounding modes: beware your BLAS library, beware your compiler.

Efficiency:

Parallelization of algorithms may invalidate some hypotheses.
Are blocked algorithms the best choice to exploit multi-cores (more than 10 cores)?