Odd but Error-Free FastTwoSum

Sehyeok Park, Jay P. Lim, Santosh Nagarakatte

FastTwoSum

- **Error-free transformations (EFT)**: algorithms for computing rounding error
- FastTwoSum computes rounding error of floating-point (FP) addition
  - Preserves full information about real sum (EFT for FP addition)
  - Building block for accurate summation algorithms

Kahan Summation Algorithm with FastTwoSum

$$
\begin{aligned}
&\text{KahanSummation}(a):\\
&\quad s = 0,\ c = 0\\
&\quad \text{for } i \text{ in } 1 \ldots n:\\
&\qquad s, e = \text{FastTwoSum}(s, a[i])\\
&\qquad c = \circ(c + e)\\
&\quad s = \circ(s + c)
\end{aligned}
$$

- Computes $\Sigma_i a_i$ via FP addition
- FastTwoSum computes rounding errors
- s = FP sum, e = rounding error

Limitations of FastTwoSum

- Requires round-to-nearest (e.g., RNE)
- Other faithful rounding modes require additional conditions

Round-to-Odd (RO)

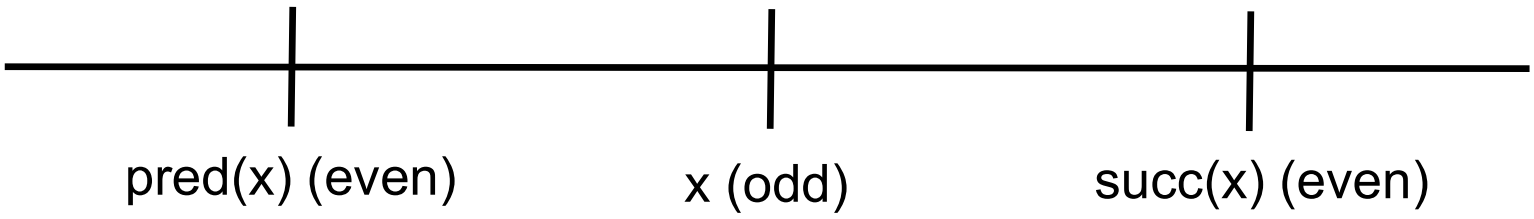
$$\mathrm{RO}(\mathrm{r}) = \begin{cases} \mathrm{r} & \text{if } \mathrm{r} \in \mathbb{F}, \\ \mathrm{succ}(\mathrm{r}) & \text{else if } \mathrm{succ}(\mathrm{r}) \text{ has an odd encoding}, \\ \mathrm{pred}(\mathrm{r}) & \text{otherwise} \end{cases}$$

- $\mathrm{r} \notin \mathbb{F} \rightarrow$ RO(r) is a faithful rounding of r with an *odd* significand

Round-to-Odd (RO)

$$\mathrm{RO}(\mathrm{r}) = \begin{cases} \mathrm{r} & \text{if } \mathrm{r} \in \mathbb{F}, \\ \mathrm{succ}(\mathrm{r}) & \text{else if } \mathrm{succ}(\mathrm{r}) \text{ has an odd encoding}, \\ \mathrm{pred}(\mathrm{r}) & \text{otherwise} \end{cases}$$

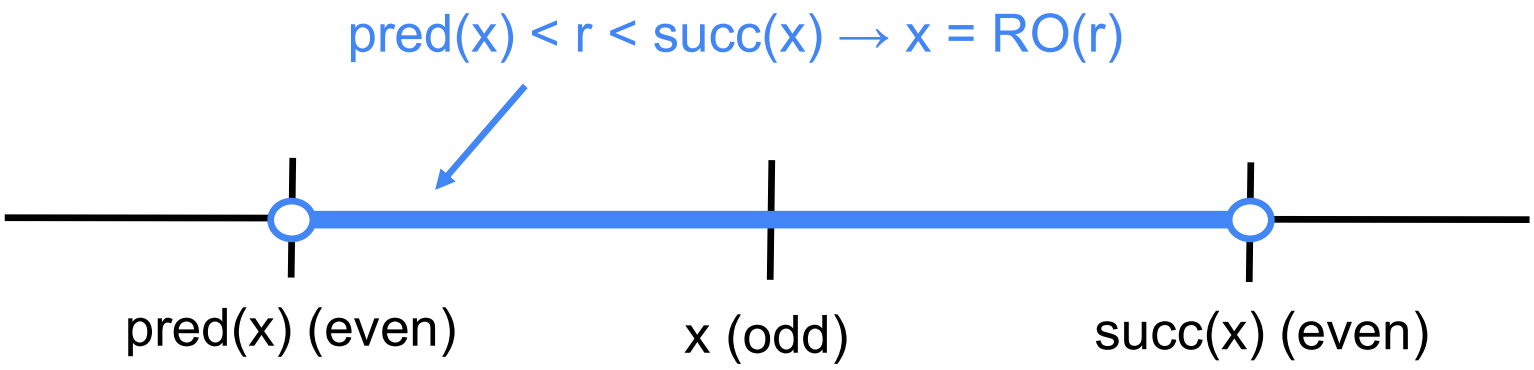
- $\mathrm{r} \notin \mathbb{F} \rightarrow$ RO(r) is a faithful rounding of r with an *odd* significand



Round-to-Odd (RO)

$$\mathrm{RO}(\mathrm{r}) = \begin{cases} \mathrm{r} & \text{if } \mathrm{r} \in \mathbb{F}, \\ \mathrm{succ}(\mathrm{r}) & \text{else if succ}(\mathrm{r}) \text{ has an odd encoding}, \\ \mathrm{pred}(\mathrm{r}) & \text{otherwise} \end{cases}$$

- $\mathrm{r} \notin \mathbb{F} \rightarrow$ RO(r) is a faithful rounding of r with an *odd* significand



Round-to-Odd (RO)

$$\mathrm{RO}(r) = \begin{cases} r & \text{if } r \in \mathbb{F}, \\ \mathrm{succ}(r) & \text{else if } \mathrm{succ}(r) \text{ has an odd encoding}, \\ \mathrm{pred}(r) & \text{otherwise} \end{cases}$$

- $r \notin \mathbb{F} \rightarrow$ RO(r) is a faithful rounding of r with an *odd* significand
- RO(r) has an *even* significand $\rightarrow r \in \mathbb{F}$
- If max real in $\mathbb{F}$ has odd significand, then RO saturates (prevents overflow)

Round-to-Odd (RO)

- RO is an *unbiased* rounding mode
- RO is composable (e.g., $RO_{FP32}(RO_{FP64}(r))) = RO_{FP32}(r)$)

  → Correctly rounded RO results can be rounded across multiple formats

  → Useful for developing arbitrary precision math libraries

- RO will be supported by the upcoming P3109 standard for ML arithmetic

Motivation

- FastTwoSum-based algorithms are not directly applicable to RO
- Future hardware that supports RO requires custom algorithms
- Our paper explores

  a. More general EFT conditions for FastTwoSum under faithful rounding modes

  b. EFT conditions for FastTwoSum tailored to RO

  c. RO-based variant of the FP-splitting algorithm ExtractScalar

Floating-Point (FP) Format $\mathbb{F}$

- $\mathbb{F}$ is subset of $\mathbb{R} \cup \{-\infty, +\infty\}$ parameterized by
  - Radix $\beta = 2$
  - Precision $p \geq 2$, unit round-off $u = 2^{-p}$
  - Extremal exponents E_{min} and E_{max}
- Minimum non-zero magnitude $\alpha = 2^{Emin - p + 1}$
- Maximum finite magnitude $\Omega = 2^{Emax + 1} - 2^{Emax - p + 1}$ (odd significand)

Notations and Properties

- For $r \in \mathbb{R}$ we define the exponent, unit-in-first-place, and unit-in-last-place as

$$e_r = \begin{cases} -\infty & \text{if } r = 0, \\ \lfloor \log_2(|r|) \rfloor & \text{otherwise} \end{cases} \qquad \text{ufp}(r) = 2^{e_r}$$

$$\text{ulp}(r) = \begin{cases} \alpha & \text{if } |r| < 2^{E_{\min}}, \\ 2u \cdot \text{ufp}(r) & \text{otherwise} \end{cases}$$

- $a \in b\mathbb{Z}$ denotes that a is an integer multiple of b
- For all finite $x \in \mathbb{F}$, $x \in \text{ulp}(x)\mathbb{Z}$, $x \in 2u \cdot \text{ufp}(x)\mathbb{Z}$, and $x \in \alpha\mathbb{Z}$

FastTwoSum

$$\begin{aligned}&\mathrm{FastTwoSum}(a,b):\\ &\quad x = \circ_1(a+b)\\ &\quad z = \circ_2(x-a)\\ &\quad y = \circ_3(b-z)\end{aligned}$$

When o_1, o_2, o_3 perform faithful rounding, x + y = a + b if

FastTwoSum

$$\begin{aligned}&\mathrm{FastTwoSum}(a,b):\\ &\quad x = \circ_1(a+b)\\ &\quad z = \circ_2(x-a)\\ &\quad y = \circ_3(b-z)\end{aligned}$$

When o_1, o_2, o_3 perform faithful rounding, $x + y = a + b$ if

FastTwoSum

$$\begin{aligned}&\mathrm{FastTwoSum}(a,b):\\ &\quad x = \circ_1(a+b)\\ &\quad z = \circ_2(x-a)\\ &\quad y = \circ_3(b-z)\end{aligned}$$

When o_1, o_2, o_3 perform faithful rounding, x + y = a + b if

- $|a+b| \leq \Omega$
- $x - a \in \mathbb{F}$ (i.e., z = x - a)

FastTwoSum

$$\begin{aligned}&\mathrm{FastTwoSum}(a,b):\\ &\quad x = \circ_1(a+b)\\ &\quad z = \circ_2(x-a)\\ &\quad y = \circ_3(b-z)\end{aligned}$$

When o_1, o_2, o_3 perform faithful rounding, x + y = a + b if

- $|a+b| \leq \Omega$
- $x - a \in \mathbb{F}$ (i.e., z = x - a) ← *Guaranteed if $a \in \mathrm{ulp}(b)\mathbb{Z}$

*FastTwoSum Revisited. C.-P., Jeannerod and P., Zimmermann. ARITH 2025.

FastTwoSum

$$\begin{aligned}&\mathrm{FastTwoSum}(a,b):\\ &\quad x = \circ_1(a+b)\\ &\quad z = \circ_2(x-a)\\ &\quad y = \circ_3(b-z)\end{aligned}$$

When o_1, o_2, o_3 perform faithful rounding, x + y = a + b if

- $|a+b| \leq \Omega$
- $a \in \mathrm{ulp}(b)\mathbb{Z}$ (ensures z = x - a and y = o_3(a+b - x) = o_3(δ))
- $\delta \in \mathbb{F} \rightarrow$ y = δ and x + y = a + b

FastTwoSum

$$\begin{aligned}&\mathrm{FastTwoSum}(a,b):\\ &\quad x = \circ_1(a+b)\\ &\quad z = \circ_2(x-a)\\ &\quad y = \circ_3(b-z)\end{aligned}$$

When o_1, o_2, o_3 perform faithful rounding, x + y = a + b if

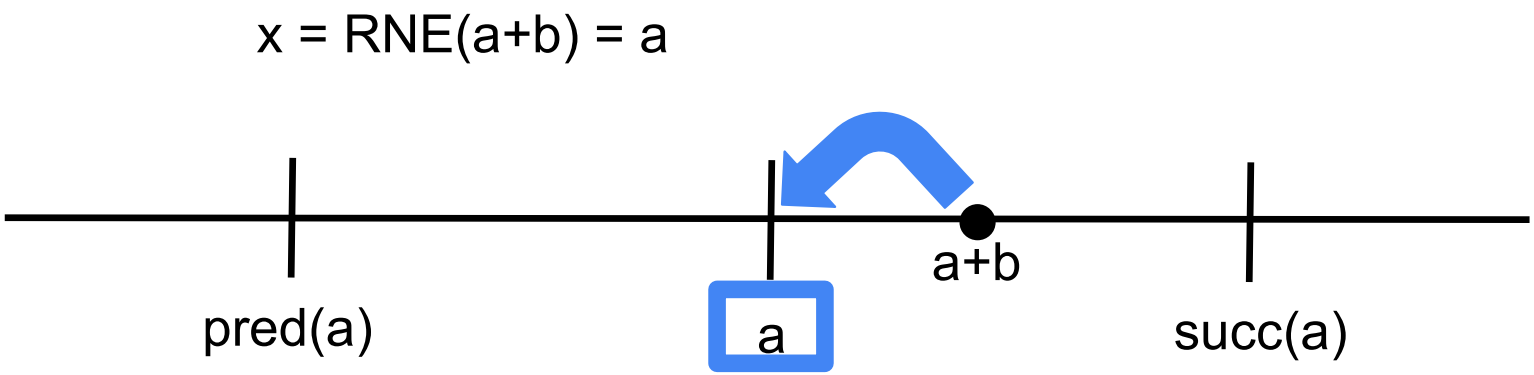
- $|a+b| \leq \Omega$
- $a \in \mathrm{ulp}(b)\mathbb{Z}$ (ensures z = x - a and y = o_3(a+b - x) = $o_3(\delta)$)
- $\delta \in \mathbb{F}$ ← Only guaranteed if x is a round-to-nearest sum

Question 1: When is FP addition error not representable?

- WLOG, suppose |a| ≥ |b|
- δ (rounding error of FP addition) corresponds to the trailing bits of b
- If a and b overlap (i.e, |b| ≥ ulp(a)), then δ should be representable
- δ may require more than p-bits when there is **no overlap** (i.e, |b| < ulp(a))
- Without RN, faithful rounding won't ensure δ ∈ $\mathbb{F}$ if **|b| < ½ ulp(a)**

δ with Round-to-Nearest (RN)

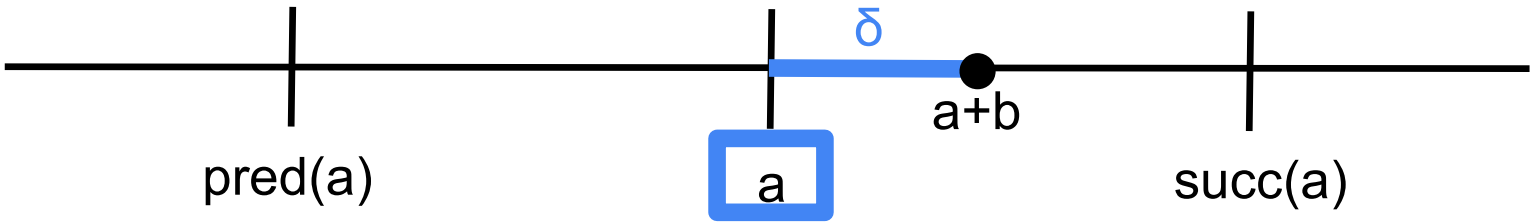
- Suppose |b| < ½ ulp(a) → pred(a) < a+b < succ(a)



δ with Round-to-Nearest (RN)

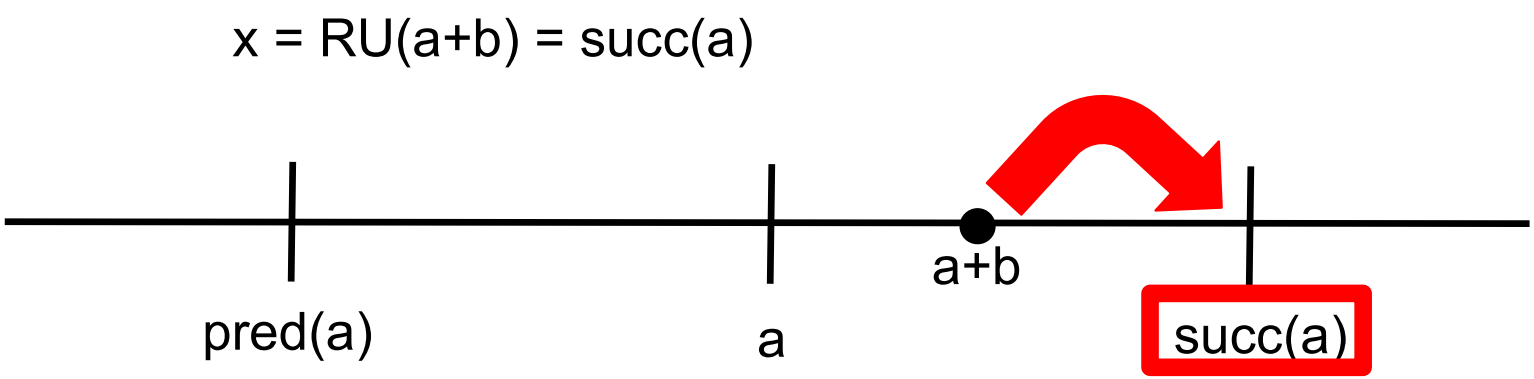
- Suppose $|b| < ½\ \text{ulp}(a) \rightarrow \text{pred}(a) < a+b < \text{succ}(a)$

$$x = \text{RNE}(a+b) = a \quad \rightarrow \quad \delta = a+b - x = b \in \mathbb{F}$$



δ without Round-to-Nearest (RN)

- Suppose |b| < ½ ulp(a) → pred(a) < a+b < succ(a)

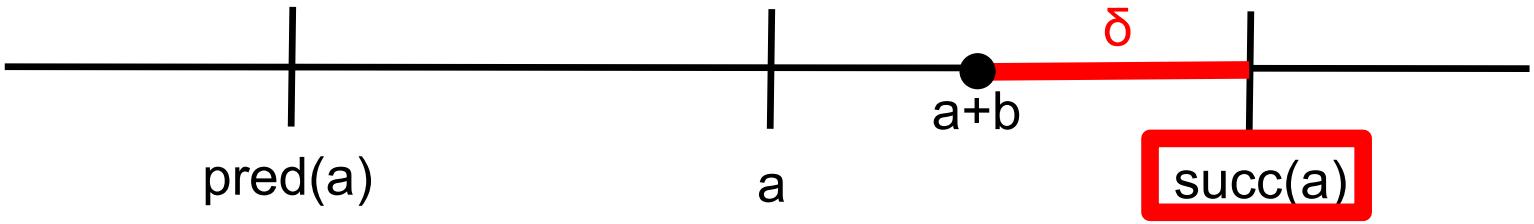


δ without Round-to-Nearest (RN)

- Suppose |b| < ½ ulp(a) → pred(a) < a+b < succ(a)

//May not be in $\mathbb{F}$ if |b| << |a|

x = RU(a+b) = succ(a) → δ = a+b - x = b - ulp(a)



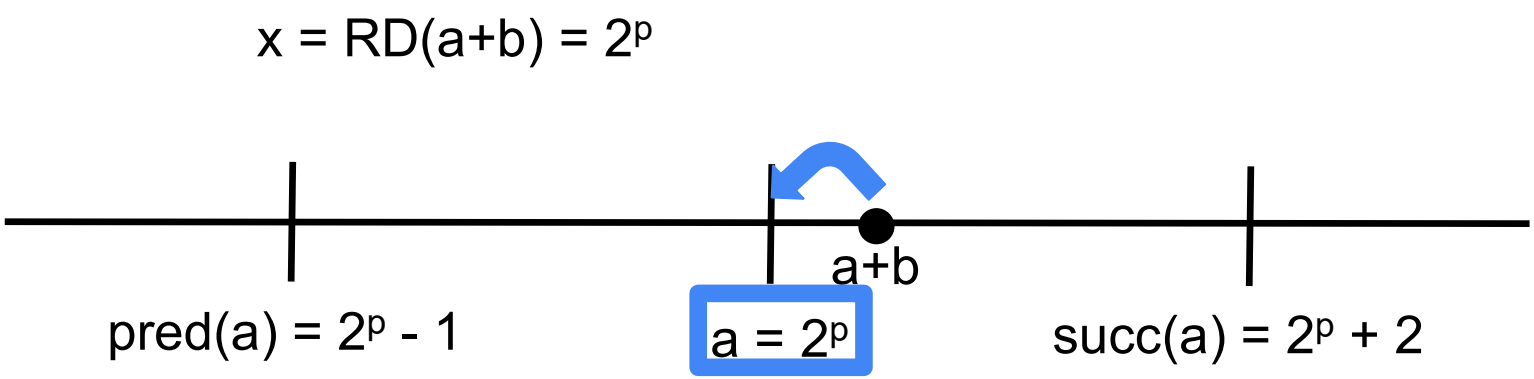
Question 1: When is FP addition error not representable?

- WLOG, suppose $|a| \geq |b|$
- δ (rounding error of FP addition) corresponds to the trailing bits of b
- If a and b overlap (i.e, $|b| \geq \text{ulp}(a)$), then δ should be representable
- δ may require more than p-bits when there is **no overlap** (i.e, $|b| < \text{ulp}(a)$)
- Without RN, faithful rounding won't ensure $\delta \in \mathbb{F}$ if **$|b| < \frac{1}{2}\,\text{ulp}(a)$**

→ Previous conditions for $\delta \in \mathbb{F}$ under faithful rounding require $e_a - e_b \leq p$

Counterexamples for $e_a - e_b \leq p$ under RD

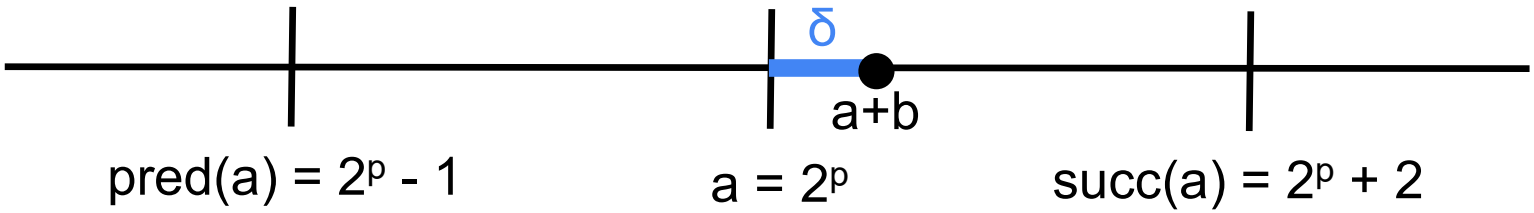
- Suppose $a = 2^p$ and $b = 2^{-1}$ (i.e., $e_a - e_b = p + 1$)



Counterexamples for $e_a - e_b \leq p$ under RD

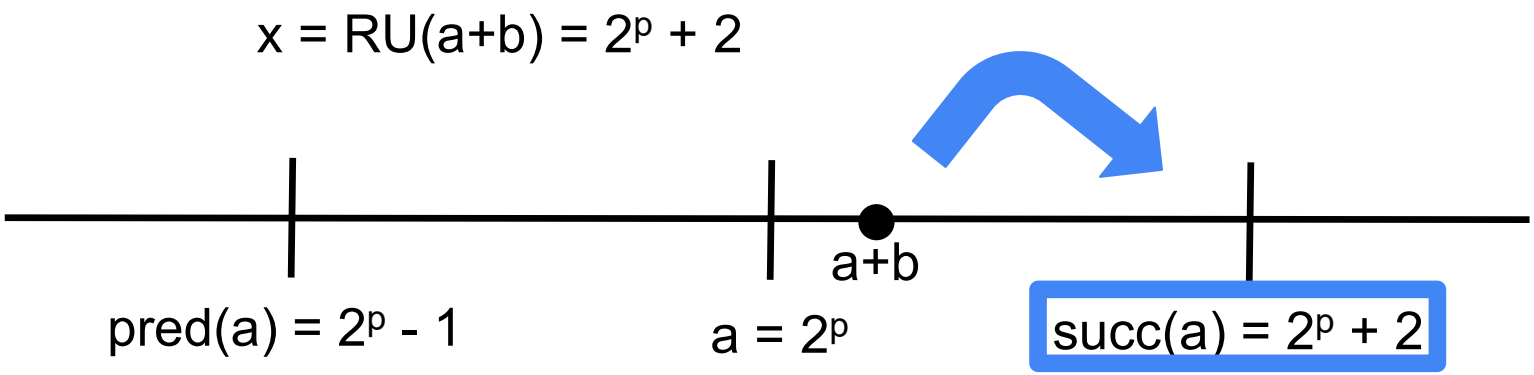
- Suppose $a = 2^p$ and $b = 2^{-1}$ (i.e., $e_a - e_b = p + 1$)

$x = RD(a+b) = 2^p \quad \rightarrow \quad \delta = a+b - x = b = 2^{-1} \in \mathbb{F}$



Counterexamples for $e_a - e_b \leq p$ under RU

- Suppose $a = 2^p$ and $b = 2^{-1}$ (i.e., $e_a - e_b = p + 1$)

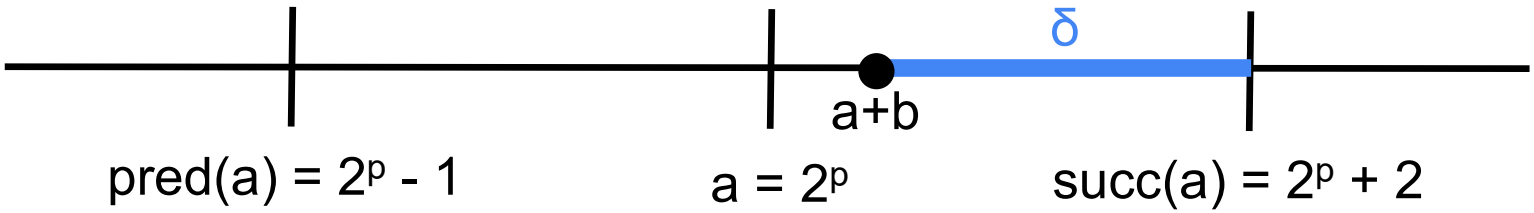


Counterexamples for $e_a - e_b \leq p$ under RU

- Suppose $a = 2^p$ and $b = 2^{-1}$ (i.e., $e_a - e_b = p + 1$)

//If $p \geq 2$, then $\delta \in \mathbb{F}$

$x = RU(a+b) = 2^p + 2 \quad \rightarrow \quad \delta = a+b - x = 2^{-1} - 2^1 = -1.1_2$



Counterexamples for $e_a - e_b \leq p$

- Let $a = 2^p$ and $b = 2^{-1}$ (i.e., $e_a - e_b = p + 1$)

  $\rightarrow \delta = a + b - x$ is representable under all faithful rounding modes

- $e_a - e_b \leq p$ is *sufficient* but **not necessary**

- More precise conditions are possible

Question 2: When is a real representable?

- Let $r \in \mathbb{R}$ and $k \in \mathbb{Z}$. If $r \in \max(\alpha, 2^k)\mathbb{Z}$ and $|r| \leq \min(\Omega, u^{-1} \cdot 2^k)$, then $r \in \mathbb{F}$.

- $r \in \alpha\mathbb{Z}$ and $|r| \leq \Omega$ (r is entirely within dynamic range)

Question 2: When is a real exactly representable in $\mathbb{F}$?

- Let $r \in \mathbb{R}$ and $k \in \mathbb{Z}$. If $r \in \max(\alpha, 2^k)\mathbb{Z}$ and $|r| \leq \min(\Omega, u^{-1} \cdot 2^k)$, then $r \in \mathbb{F}$.

- $r \in \alpha\mathbb{Z}$ and $|r| \leq \Omega$ (r is entirely within dynamic range)

- There exists some power of 2 (e.g. 2^k) such that

  (i) $r \in 2^k\mathbb{Z}$

  (ii) $|r/2^k| \leq u^{-1} = 2^p$ (i.e., magnitude of integer quotient is no greater than 2^p)

  → For some binary exponent, r's significand is **representable in p-bits**

Question 2: When is a real exactly representable in $\mathbb{F}$?

- Let $r \in \mathbb{R}$ and $k \in \mathbb{Z}$. If $r \in \max(\alpha, 2^k)\mathbb{Z}$ and $|r| \leq \min(\Omega, u^{-1} \cdot 2^k)$, then $r \in \mathbb{F}$.

- $r \in \alpha\mathbb{Z}$ and $|r| \leq \Omega$ (r is entirely within dynamic range)

- There exists some power of 2 (e.g. 2^k) such that

  (i) $|r| \leq 2^k$

  (ii) $r \in u \cdot 2^k\mathbb{Z}$ (i.e., $r \in 2^{k-p}\mathbb{Z}$)

  → For some binary exponent, r's significand is **representable in p-bits**

Question 3: When is δ representable?

Suppose $|b| < \frac{1}{2}\,\mathrm{ulp}(a)$. Let $x = o(a+b)$ for faithful o and $\delta = a + b - x$.

- $\delta \in \alpha\mathbb{Z}$ and $|\delta| \leq \Omega$
- $|\delta| < 2u \cdot \mathrm{ufp}(a)$
- $\sigma = 2u^2 \cdot \mathrm{ufp}(a)$ and $\delta \in \sigma\mathbb{Z} \rightarrow \delta \in \mathbb{F}$

Question 3: When is δ representable?

Suppose $|b| < ½\ \mathrm{ulp}(a)$. Let $x = o(a+b)$ for faithful o and $\delta = a + b - x$.

- $\delta \in \alpha\mathbb{Z}$ and $|\delta| \leq \Omega$
- $|\delta| < 2u \cdot \mathrm{ufp}(a)$
- $\sigma = 2u^2 \cdot \mathrm{ufp}(a)$ and $\delta \in \sigma\mathbb{Z} \rightarrow \delta \in \mathbb{F}$
- $\delta \in \gcd(a, b)\mathbb{Z}$ //Addition/subtraction preserves common binary factors
- $a \in \sigma\mathbb{Z}$
- $b \in \sigma\mathbb{Z} \rightarrow \delta \in \sigma\mathbb{Z} \rightarrow \delta \in \mathbb{F}$

New Conditions for $\delta \in \mathbb{F}$

Let $a, b \in \mathbb{F}/\{\pm\infty\}$. Let $x = o(a+b)$ for a faithful o and $\delta = a+b - x$. If

- $|a+b| \leq \Omega$
- $a \in 2u^2 \cdot ufp(b)\mathbb{Z}$
- $b \in 2u^2 \cdot ufp(a)\mathbb{Z}$

If $|a| \geq |b|$ then $ufp(a) \geq ufp(b)$ and v.v.

One of these will always hold by default.

Then, $\delta \in \mathbb{F}$.

New EFT Conditions for FastTwoSum

$$\begin{aligned}&\mathrm{FastTwoSum}(a,b):\\ &x = \circ_1(a+b)\\ &z = \circ_2(x-a)\\ &y = \circ_3(b-z)\end{aligned}$$

When o_1, o_2, o_3 perform faithful rounding, x + y = a + b if

- $|a+b| \leq \Omega$
- $a \in \mathrm{ulp}(b)\mathbb{Z}$ (ensures $x - a \in \mathbb{F}$ and $y = o_3(\delta)$)
- $b \in 2u^2 \cdot \mathrm{ufp}(a)$ (ensures $\delta \in \mathbb{F}$) **//Lower bound on |b|**

//Exponent diff. ≤ 2p - 1

What about RO?

$$\begin{aligned}&\mathrm{FastTwoSum}(a,b):\\ &\quad x = \circ_1(a+b)\\ &\quad z = \circ_2(x-a)\\ &\quad y = \circ_3(b-z)\end{aligned}$$

When o_1 = RO and o_2, o_3 perform faithful rounding, x + y = a + b if

- $|a+b| \leq \Omega$
- $a \in \mathrm{ulp}(b)\mathbb{Z}$ (ensures $x - a \in \mathbb{F}$ and $y = o_3(\delta)$)
- $b \in 2u^2 \cdot \mathrm{ufp}(a)$ (ensures $\delta \in \mathbb{F}$)

What about RO?

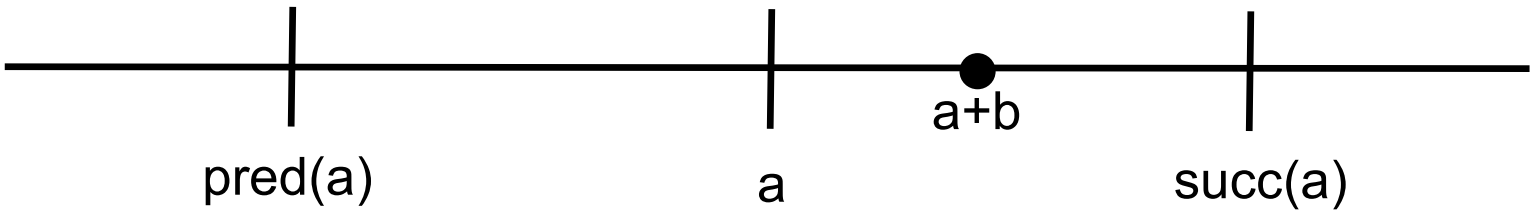
$$\begin{aligned}&\mathrm{FastTwoSum}(a,b):\\ &\quad x = \circ_1(a+b)\\ &\quad z = \circ_2(x-a)\\ &\quad y = \circ_3(b-z)\end{aligned}$$

When o_1 = RO and o_2, o_3 perform faithful rounding, x + y = a + b if

- ~~|a+b| ≤ Ω~~ (|x| = |RO(a+b)| ≤ Ω due to saturation)
- a ∈ ulp(b)$\mathbb{Z}$ (ensures x - a ∈ $\mathbb{F}$ and y = o_3(δ))
- b ∈ $2u^2$ · ufp(a) (ensures δ ∈ $\mathbb{F}$)

Properties of RO addition

- Suppose |b| < ½ ulp(a) → pred(a) < a+b < succ(a)

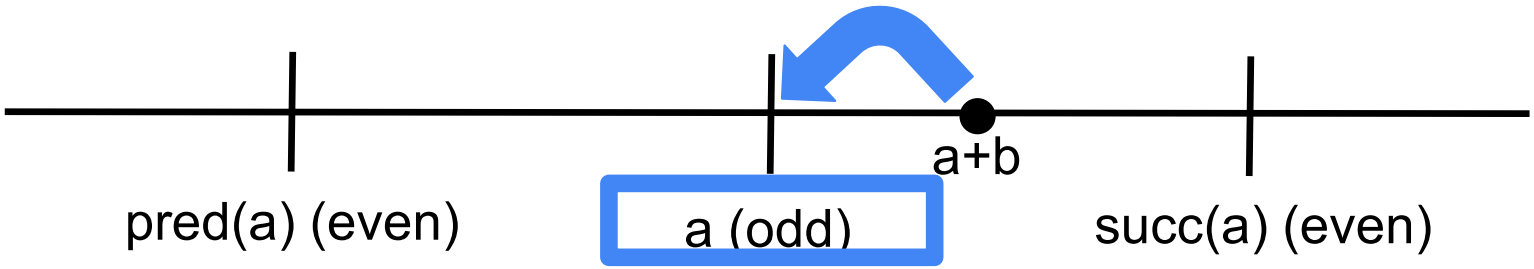


Properties of RO addition

- Suppose |b| < ½ ulp(a) → pred(a) < a+b < succ(a)

If *a* has an *odd significand*

x = RO(a+b) = a

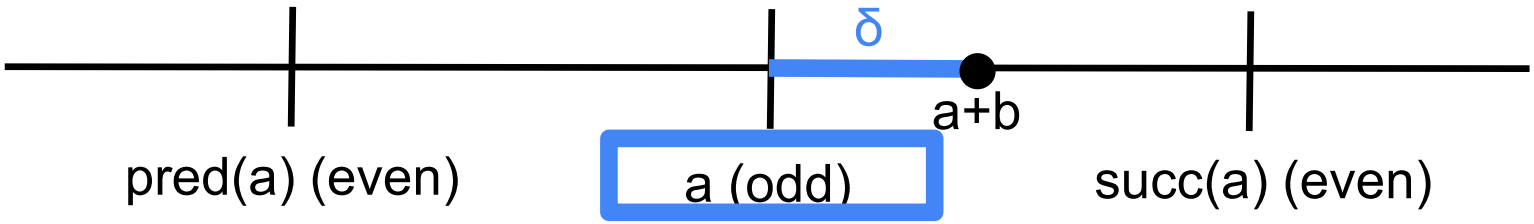


Properties of RO addition

- Suppose $|b| < \frac{1}{2}\,\mathrm{ulp}(a) \rightarrow \mathrm{pred}(a) < a+b < \mathrm{succ}(a)$

If *a* has an *odd significand*

$x = RO(a+b) = a \quad \rightarrow \quad \delta = a+b - x = b \in \mathbb{F}$



New Conditions for δ ∈ 𝔽 under RO

Let a, b ∈ 𝔽/{±∞}. Let x = RO(a+b) and δ = a+b - x.

If max(|a|, |b|) has an *odd* significand, then δ ∈ 𝔽.

New EFT Conditions for FastTwoSum under RO

$$\begin{aligned}&\mathrm{FastTwoSum}(a,b):\\ &\quad x=\circ_1(a+b)\\ &\quad z=\circ_2(x-a)\\ &\quad y=\circ_3(b-z)\end{aligned}$$

When o_1 = RO and o_2, o_3 perform faithful rounding, x + y = a + b if

- a ∈ ulp(b)$\mathbb{Z}$ (ensures x - a ∈ $\mathbb{F}$ and y = o_3(δ))
- a has an odd significand (ensures δ ∈ $\mathbb{F}$)

  * No lower bound on |b|

FP Splitting via FastTwoSum

$$\mathrm{FastTwoSum}(a,b):$$
$$x = \circ_1(a+b)$$
$$z = \circ_2(x-a)$$
$$y = \circ_3(b-z)$$

- Under EFT conditions, b = z+y
- z corresponds to higher order bits of b
- y corresponds to lower order bits of b

→ FastTwoSum enables FP splitting

ExtractScalar [Rump et al., 2008]

$$\text{ExtractScalar}(\sigma, x):$$
$$\sigma' = \circ_1(\sigma + x)$$
$$x_h = \circ_2(\sigma' - \sigma)$$
$$x_\ell = \circ_3(x - x_h)$$

*Accurate Floating-Point Summation Part 1: Faithful Rounding. Rump et al. SIAM Vol 31. No.1. 2008.

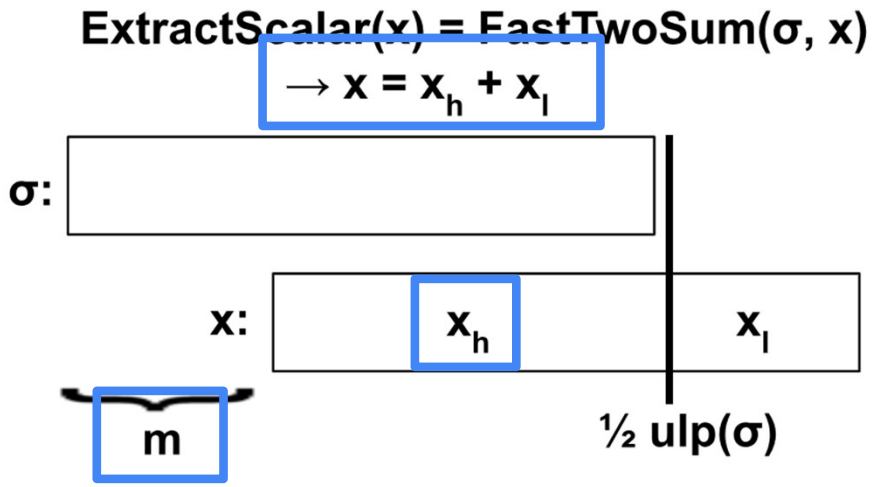
ExtractScalar [Rump et al., 2008]

$$\begin{aligned}&\text{ExtractScalar}(\sigma, x):\\ &\sigma' = \circ_1(\sigma + x)\\ &x_h = \circ_2(\sigma' - \sigma)\\ &x_\ell = \circ_3(x - x_h)\end{aligned}$$

- σ is a power of two s.t. σ ≥ |x|
- Performs FastTwoSum with σ and x
- 1st op. rounds away trailing bits of x
- 2nd op. obtains non-rounded portion
- 3rd op. restores rounded portion

*Accurate Floating-Point Summation Part 1: Faithful Rounding. Rump et al. SIAM Vol 31. No.1. 2008.

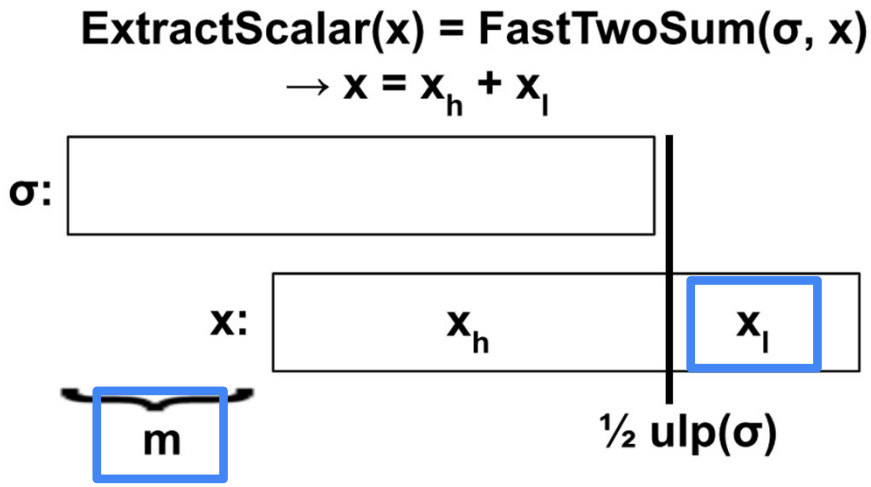
ExtractScalar [Rump et al., 2008]



- σ is a power of two s.t. σ ≥ |x|
- Performs FastTwoSum with σ and x
- Extracts higher order bits of x into x_h
- m controls no. of effective bits in x_h
- Ensures an EFT: $x = x_h + x_l$

*Accurate Floating-Point Summation Part 1: Faithful Rounding. Rump et al. SIAM Vol 31. No.1. 2008.

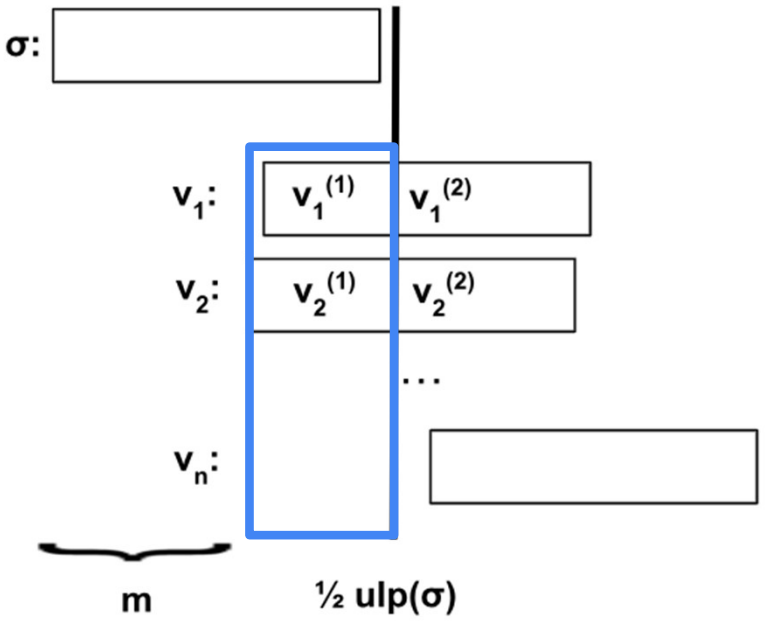
ExtractScalar without RN



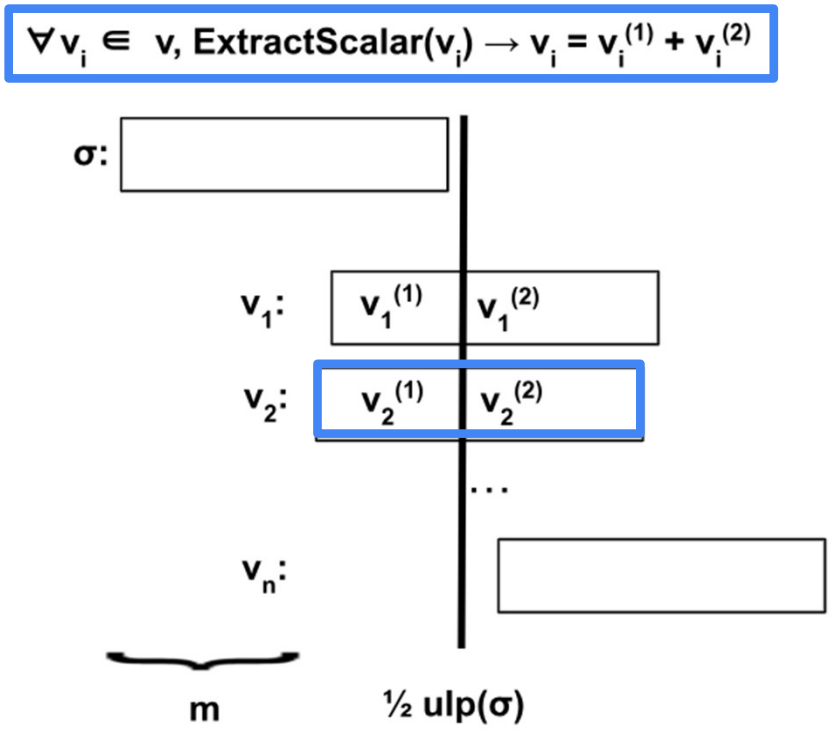
- x_l captures rounding error of o(σ + x)
- No upper bound on m
- δ = σ + x - o(σ + x) ∈ 𝔽 not ensured
- x = x_h + x_l not ensured

ExtractScalar on Multiple FP Numbers

$\forall v_i \in v$, ExtractScalar(v_i) $\rightarrow v_i = v_i^{(1)} + v_i^{(2)}$

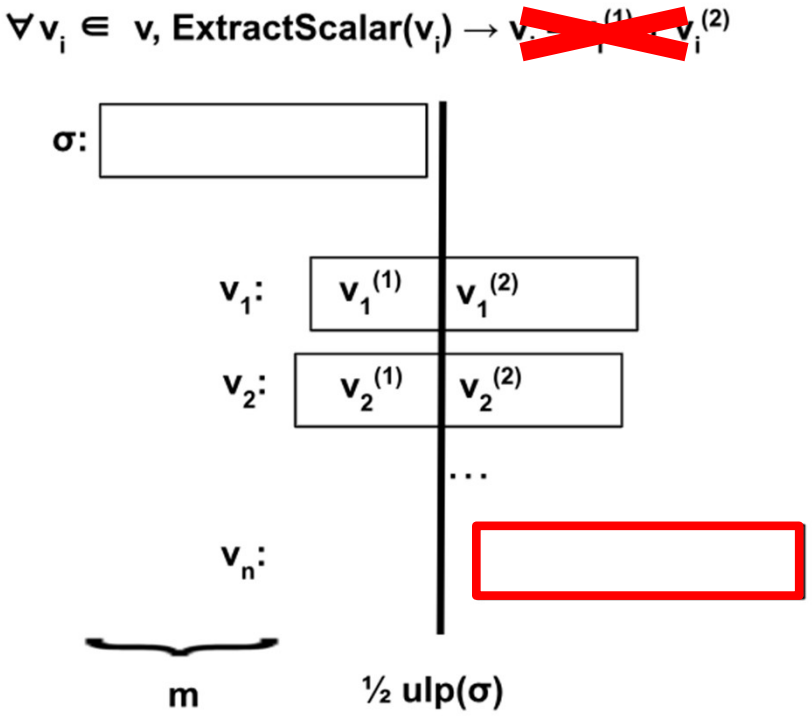


ExtractScalar on Multiple FP Numbers



- Set σ relative to $\max_i |v_i|$
- Perform ExtractScalar element-wise

ExtractScalar on Multiple FP Numbers with RO



- Set σ relative to $\max_i|v_i|$
- Perform ExtractScalar element-wise
- If $|v_i| << \sigma$, EFT not ensured
- RO can't handle $\max_i|v_i| >> \min_i|v_i|$

Properties of ExtractScalar

- ExtractScalar's properties depend on the *exponent difference* of $\boldsymbol{\sigma}$ and x
- The *significand* of $\boldsymbol{\sigma}$ does not matter
- Powers of 2 (2^k for an integer k) have even significands

  → succ(2^k) has same exponent but an *odd significand*

  → Make $\boldsymbol{\sigma}$ = succ(2^k)!

New EFT Conditions for ExtractScalar under RO

$$\begin{aligned}&\mathrm{ExtractScalar}(\sigma, x):\\ &\sigma' = \circ_1(\sigma + x)\\ &x_h = \circ_2(\sigma' - \sigma)\\ &x_\ell = \circ_3(x - x_h)\end{aligned}$$

When o_1 = RO and o_2, o_3 perform faithful rounding, if

- $|x| \leq 2^k$ for $k \in \mathbb{Z}$
- $\sigma = \mathrm{succ}(2^k)$

Then, $x_h \in \mathrm{ulp}(\sigma)\mathbb{Z}$ and $x = x_h + x_l$

Conclusion

- Identified more general EFT conditions for FastTwoSum with faithful rounding
- Identified EFT conditions for FastTwoSum tailored to RO
- Developed a RO-based variant of the FP splitting algorithm ExtractScalar