

Round-off Error in Multiplication of an Integer by a Fixed-point Approximation of a Real Number.

Nikolai Golovchenko, 21-Mar-2004.

It is a common problem in computer arithmetic to multiply an integer (or fixed-point in general) number by a real number, which cannot be exactly represented by a fixed-point number. An example would be a real decimal number 0.1, which is an infinitely repeating binary fraction $0.0(0011)_2$. The question is then: how many fractional bits of the real number should be used to preserve the accuracy as much as possible? We will mathematically deduct an answer in this document. Other answers are possible if accuracy is defined differently.

1. Absolute Multiplication Error

Problem:

Given an integer

$$t = 0 \dots T,$$

a positive real number k , a fixed-point approximation of k with c fractional bits:

$$k' = \text{floor}(k * 2^c) / 2^c,$$

find the maximum absolute error of the product

$$p' = \text{floor}(t * k') = \text{floor}(t * \text{floor}(k * 2^c) / 2^c)$$

where:

p' – approximate product;

k' – approximated value of k using a fixed-point representation;

c – the number of preserved fractional bits of k ;

if the accurate product is defined as:

$$p = \text{floor}(t * k).$$

Solution:

The absolute error can be defined as:

$$e = \text{abs}(p' - p) = \text{abs}(\text{floor}(t * k') - \text{floor}(t * k))$$

Since we approximate k by truncating the least significant bits and k is always positive, the $\text{abs}()$ function is not necessary:

$$e = p - p' = \text{floor}(t * k) - \text{floor}(t * k')$$

The maximum absolute error is then expressed as:

$$e_{\max} = \max(\text{floor}(t*k) - \text{floor}(t * k'))$$

This expression can be simplified if we consider the approximation error of k . The absolute approximation error of k is:

$$e_k = k - k' = (k - \text{floor}(k * 2^c) / 2^c)$$

It can be between 0 (no approximation error) and 2^{-c} (all truncated bits of k at bit positions 2^{-c-1} to $2^{-\infty}$ are set to one). The limit of the maximum approximation error of k is then:

$$e_{k\max} = 2^{-c}$$

The maximum multiplication error can then be expressed as:

$$\begin{aligned} e_{\max} &= \max(\text{floor}(t * k) - \text{floor}(t * k')) = \\ &= \max(\text{floor}(t * k' + t * e_k) - \text{floor}(t * k')) = \\ &= \max(\text{ceil}(t * e_k)) \end{aligned}$$

It can be seen that as we increase t , the error increases as well. The maximum error occurs at $t=T$:

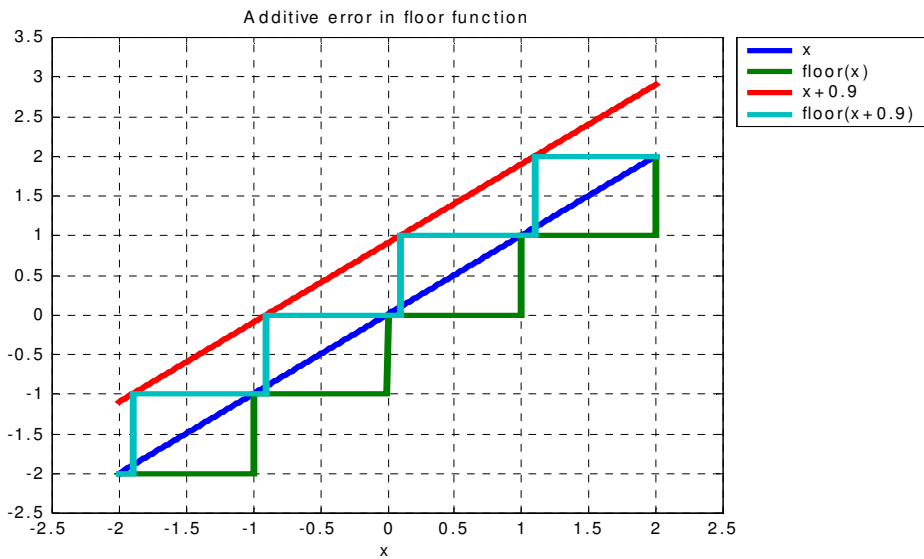
$$e_{\max} = \max(\text{ceil}(T * e_k))$$

Note: we just used the following identity:

$$\begin{aligned} \text{floor}(x+E) - \text{floor}(x) &= \text{floor}(E), & \text{if } \text{frac}(x) < 1-\text{frac}(E) \\ &= \text{ceil}(E), & \text{if } \text{frac}(x) \geq 1-\text{frac}(E) \end{aligned}$$

(Warning: this identity was only checked for positive x and positive E . $\text{frac}(x)$ is computed as $x - \text{floor}(x)$. The important point in our case is that in a general case, the largest error is produced when we use the $\text{ceil}()$ function.)

A simple illustration of the floor function with an offset of $E=0.9$:



We can minimize the maximum multiplication error by adjusting two variables - T and c:

Maximum absolute multiplication error:

$$e_{\max 1} = \text{ceil}(T * e_k) \quad (\text{maximized for a known } e_k)$$

$$e_{\max 2} = \text{ceil}(T * 2^{-c}) \quad (\text{maximized for the worst case } e_k)$$

Absolute approximation error in k:

$$e_k = k - k' = (k - \text{floor}(k * 2^c) / 2^c)$$

Example:

t=0,1,..255 (all possible values of an 8-bit integer);

T=255;

k=0.1;

c=6 (k' is approximated to 6 fractional bits);

k' = 0.09375;

The maximum absolute error is then:

$$\begin{aligned} e_{\max 1} &= \text{ceil}(T * e_k) = \text{ceil}(T * (k - k')) = \\ &= \text{ceil}(255 * (0.1 - 0.09375)) = 2 \end{aligned}$$

for the known k.

For an unknown k with six preserved fractional bits:

$$e_{\max 2} = \text{ceil}(T * 2^{-c}) = \text{ceil}(255 * 2^{-6}) = 4$$

2. The Minimum Possible Absolute Error

Problem:

Given an integer

$$t = 0 \dots T,$$

a positive real number k , a fixed-point approximation of k with c fractional bits:

$$k' = \text{floor}(k * 2^c) / 2^c,$$

find the minimum number of fractional bits that are sufficient to achieve the minimum possible absolute error of the product:

$$p' = \text{floor}(t * k') = \text{floor}(t * \text{floor}(k * 2^c) / 2^c),$$

where:

p' – approximate product;

k' – approximated value of k using a fixed-point representation;

c – the number of preserved fractional bits of k ;

if the accurate product is defined as:

$$p = \text{floor}(t * k).$$

Solution:

From the previous section, the absolute error expressions were deduced:

$$e_k = k - k' = (k - \text{floor}(k * 2^c) / 2^c) - \text{absolute error in approximated } k.$$

$$e_{\max 1} = \text{ceil}(T * e_k) \quad (\text{maximized for a known } e_k)$$

$$e_{\max 2} = \text{ceil}(T * 2^{-c}) \quad (\text{maximized for the worst case } e_k)$$

They show that if there is an approximation error in k , the absolute multiplication error e_{\max} will equal at least one. This is the minimum possible error. Let's find the minimum number of fractional bits c in the approximated k that produces the same maximum error. The condition can be expressed as:

$$e_{\max} < 2$$

First, the case of a known e_k :

$$\begin{aligned}e_{\max 1} &< 2 \\ \text{ceil}(T * e_k) &< 2 \\ T * e_k &\leq 1 \\ e_k &\leq 1/T\end{aligned}$$

Second, in case of unknown e_k (e.g., k is a variable):

$$\begin{aligned}e_{\max 2} &< 2 \\ \text{ceil}(T * 2^{-c}) &< 2 \\ T * 2^{-c} &\leq 1 \\ T &\leq 2^c \\ c &\geq \log_2 T\end{aligned}$$

Minimized multiplication error conditions:

$$\begin{aligned}e_k &\leq 1/T \quad (\text{for a known } e_k) \\ c &\geq \log_2 T \quad (\text{for the worst case } e_k)\end{aligned}$$

Absolute approximation error in k :

$$e_k = k - k' = (k - \text{floor}(k * 2^c) / 2^c)$$

Example:

$t=0,1,..255$ (all possible values of an 8-bit integer);

$T=255$;

$k=0.1$;

In case of known k :

$$e_k \leq 1/T$$

$$1/T = 1/255$$

$$k = 1/16 + 1/32 + 1/256 + 1/512 + 1/4096 + 1/8192 \dots$$

$$\text{Try 1: } e_k = 0.1 - 1/16 - 1/32 - 1/256 < 1/255$$

$$\text{Try 2: } e_k = 0.1 - 1/16 - 1/32 > 1/255$$

$$k' = 1/16 + 1/32 + 1/256.$$

In case of unknown k:

$$c \geq \log_2 T$$

$$c \geq \log_2(255)$$

$$c \geq 7.99$$

$$c = 8$$

$$k' = \text{floor}(k * 2^c) / 2^c = \text{floor}(0.1 * 2^8) / 2^8 = 1/16 + 1/32 + 1/256$$

Check the maximum absolute errors:

$$e_{\max 1} = \text{ceil}(T * e_k) = \text{ceil}(255 * (0.1 - 1/16 - 1/32 - 1/256)) = 1$$

$$e_{\max 2} = \text{ceil}(T * 2^{-c}) = \text{ceil}(255 * 2^{-8}) = 1$$