Chapter 6

Homework 21935004 谭焱

6.1 Problem

Problem 6.1. Convert the decimal integer 477 to a nomalized FPN with $\beta = 2$.

Solution. $477 = (111011101)_2 = 1.11011101 \times 2^8$.

Problem 6.2. Convert the decimal fraction 3/5 to a nomalized FPN with $\beta = 2$.

Solution. $3/5 = (0.100110011001\cdots)_2 = 1.00110011001100110011001\cdots \times 2^{-1}$

Problem 6.3. Let $x = \beta^e, e \in \mathbb{Z}, L < e < U$ be a normalized FPN in $\mathbb{F}$ and $x_L, x_R \in \mathbb{F}$ the two nomalized FPNs adjacent to x such that $x_L < x < x_R$. Prove $x_R - x = \beta(x - x_L)$.

Solution. As FPN form, $x = 1 \times \beta^e$. So we know that $x_L = ((\beta - 1) + \frac{\beta-1}{\beta} + \cdots + \frac{\beta-1}{\beta^{p-1}}) \times \beta^{e-1} = 1 \times \beta^e - 1 \times \beta^{e-p}, x_R = 1 \times \beta^e + 1 \times \beta^{e-p+1}$. It's equal to $x_R - x = \beta(x - x_L)$.

Problem 6.4. By reusing your result of (Problem 6.2), find out the two normalized FPNs adjacent to $x = 3/5$ under the IEEE 745 single-precision protocol. What is $\mathrm{fl}(x)$ and the relative roundoff error?

Solution. By (Problem 6.2), get $x_L = 1.00110011001100110011000 \times 2^{-1}, x_R = 1.00110011001100110011010 \times 2^{-1}$. The $\mathrm{fl}(x) = x_R$ and the relative roundoff error is about 2^{-25}

Problem 6.5. If the IEEE 754 single-precision protocol did not roundoff numbers to the nearst, but simply dropped excess bit, what would the unit roundoff be?

Solution. This situation, the unit roundoff would be $\epsilon_u := \epsilon_M = \beta^{1-p} = 2^{-23}$.

Problem 6.6. How many bits of precision are lost in the subtraction $1 - \cos x$ when $x = \frac{1}{4}$?

Solution. $1 - \cos\frac{1}{4} = 1 \times 2^0 - 1.11110000000101010100100 \times 2^{-1} = (0.000001111111010101011100)_2 = 1.111111010101011100 \times 2^{-6}$. So lost 5 bits of precision.

Problem 6.7. Suggest at least two ways to compute $1 - \cos x$ to avoid catastrophic cancellation caused by subtraction.

Solution. Replace $1 - \cos x$ with $2\sin^2\frac{x}{2}$, or from the Taylor expansion $\frac{x^2}{2!} + \frac{x^4}{4!} + \cdots$ computing $1 - \cos x$ to avoid catastrophic cancellation.

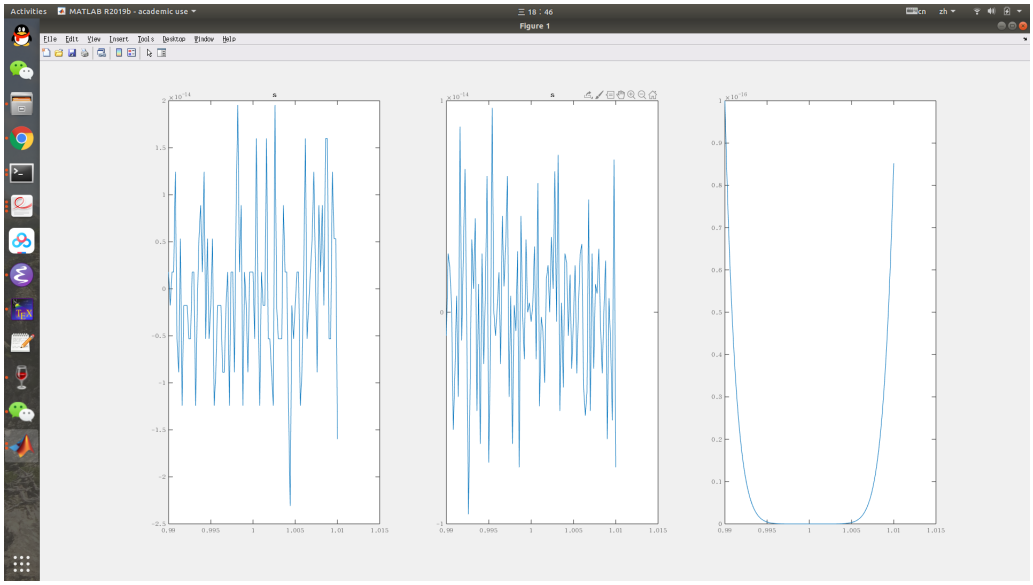

图 6.1: fgh-values

6.2 Program

(A) By programming in **C++**, print values of the functions in (1) at 101 equally spaced points covering the interval $[0.99, 1.01]$. Calculate each function in a straightforward way without rearranging or factoring. Note that the three functions are theoretically the smae, but the computed values might be very different. Plot these functions near 1.0 using a magnified scaled for the function values to see the variations involved. Discuss what you see. Which one is the most accurate? Why?

$$f(x) = x^8 - 8x^7 + 28x^6 - 56x^5 + 70x^4 - 56x^3 + 28x^2 - 8x + 1 \tag{1a}$$

$$g(x) = (((((((x-8)x+28)x-56)x+70)x-56)x+28)x-8)x+1 \tag{1b}$$

$$h(x) = (x-1)^8 \tag{1c}$$

Solution. *make machinecompute* will plot the figure. The figure as below (figure 6.1), and its natural to consider that $(x-1)^8$ is the most accurate. Since this function won't have catastrophic cancellation.

(B) Consider a normalized FPN system $\mathbb{F}$ with the characterization $\beta = 2, p = 3, L = -1, U = +1$. Answer the following by *programming* in **C++**

- compute UFL($\mathbb{F}$) and OFL($\mathbb{F}$) and output them as decimal numbers;
- enumerate all numbers in $\mathbb{F}$ and verify the corollary on the cardinality of $\mathbb{F}$ in the summary handout;
- plot $\mathbb{F}$ on the real axis;
- enumerate all the subnormal numbers of $\mathbb{F}$;
- plot the *extended* $\mathbb{F}$ on the real axis.

Solution. *make FPN* will output every thing, enumerate numbers in $\mathbb{F}$ is 25, which is equal to $2^3(1-(-1)+1)+1$. And the numbers of the subnormal numbers of $\mathbb{F}$ is 6.