۲.۳.۱ برخی از خواص نامتعارف حساب ممیز شناور

۱. جمع ممیزشناور، لزوما شرکت‌پذیر نیست یعنی

$$\exists a, b, c \in F_{\beta,p}^{L,U} \ \ s.t. \quad (a \oplus b) \oplus c \neq a \oplus (b \oplus c).$$

بعنوان مثال دستگاه بازیچه‌ی $F_{2,3}^{-1,2}$ را با سبک گردکردن به نزدیک‌ترین (زوج) در نظر بگیرید. داریم:

$$0.5 \oplus (2.5 \oplus 0.75) = 0.5 \oplus fl(3.25) = 0.5 \oplus 3 = fl(3.5) = 3.5,$$
$$(0.5 \oplus 2.5) \oplus 0.75 = fl(3) \oplus 0.75 = 3 \oplus 0.75 = fl(3.75) = 4,$$

در حالی که هیچ‌یک از این دو نیز جواب درست ریاضی نیستند!

۲. ضرب ممیزشناور، لزوما شرکت‌پذیر نیست یعنی

$$\exists a, b, c \in F_{\beta,p}^{L,U} \ \ s.t. \quad (a \otimes b) \otimes c \neq a \otimes (b \otimes c).$$

۳. ضرب ممیزشناور، لزوما بر جمع ممیزشناور پخش‌پذیر نیست یعنی

$$\exists a, b, c \in F_{\beta,p}^{L,U} \ \ s.t. \quad a \otimes (b \oplus c) \neq (a \otimes b) \oplus (a \otimes c).$$

۴. ترتیب انجام عملیات ممیزشناور، گاهی اوقات بر درستی نتیجه تاثیرگذار است.

۵. خاصیت حذفی عمل جمع (و همینطور عمل ضرب) لزوما برقرار نیست یعنی

$$\exists a, b, c \in F_{\beta,p}^{L,U} \ \ s.t. \quad a \oplus b = a \oplus c \ \ \& \ \ b \neq c.$$

$$\exists a, b, c \in F_{\beta,p}^{L,U} \ \ s.t. \quad a \otimes b = a \otimes c \ \ \& \ \ b \neq c.$$

۶. حاصل‌ضرب یک عدد ممیزشناور در معکوسش لزوما مساوی یک نیست.

همانگونه که قبلا گفتیم اگر بخواهیم خواص عملیات حسابی ممیزشناور را کنکاش کنیم نیاز چندانی

نیست که جزئیات غیرضروروی چگونگی انجام عملیاتی که واقعا در مبنای دو در ماشین صورت می پذیرد را بدانیم [۱] همانگونه که در تمرین زیر خواهیم دید، می‌توانیم از قضیه‌ی به صورت زیر استفاده کنیم: برای این‌که بدانیم حاصلِ $y \circledast x$ در ماشینی با دقت p چقدر خواهد شد، می‌توانیم $x * y$ را با دقت بینهایت رقم انجام دهیم و تنها پاسخ نهایی را یک بار به عددی متعلق به $F_{\beta,p}^{L,U}$ گرد کنیم:

تمرین ۲. سه عدد

$$a = +2.3371258 \times 10^{-5}$$

$$b = +3.3678429 \times 10^{+1}$$

$$c = -3.3677811 \times 10^{+1}$$

را در دستگاه $F_{10,8}^{-5,+5}$ در نظر گرفته و مقدار $(a \oplus b) \oplus c$ و $a \oplus (b \oplus c)$ را به‌دست آورده و مقایسه کنید. فرض کنید قواعد استاندارد آی-تریپل-ای در دستگاه $F_{10,8}^{-5,+5}$ برقرار است.

۳.۳.۱ برخی فجایع و رخدادهای ناشی از استفاده‌ی نامناسب از حساب ممیزشناور

با اینکه خطاهای گردکردن معمولا کوچک هستند، وقتی در الگوریتم‌های طولانی و پیچیده چنین خطاهایی تکرار و انباشته می‌گردند، می‌توانند آثار فاجعه‌باری داشته باشند. در اینجا برخی از اتفاقاتی که در جهان واقعی بخاطر خطاهای محاسبات کامپیوتری رخ داده‌اند را مرور می‌کنیم:

۱. انفجار موشک آریان۵ در ۴ ژوئن ۱۹۹۶ در گینه‌ی فرانسه. این انفجار در اثر خطای سرریز در کامپیوتر تنظیم‌کننده‌ی مسیر حرکت موشک رخ داد. این موشک که توسط آژانس فضایی اروپا و با هزینه ۷ میلیون دلار ساخته شده بود، حدود ۴۰ ثانیه پس از پرتاب در ارتفاع ۳۷۰۰ متری منفجر شد. دو هفته بعد از این رخداد گروهی از بازرسان گزارش خود را از دلایل انفجار موشک ارائه کردند. در گزارش بیان شده که یک عدد ممیزشناور ۶۴بیتی مربوط به شتاب افقی موشک نسبت به سکوی پرتاب باید پس از یک تبدیل در محل یک عدد صحیح ۱۶ بیتی ذخیره می‌شد. این عدد، بزرگ‌تر از ۳۲۷۶۷ که بزرگ‌ترین عدد قابل ذخیره در۱۶بیت است، بوده و در نتیجه با خطای سرریز رخ داده است و انحراف موشک از مسیر و انفجار آن در نتیجه‌ی این ایراد نرم‌افزاری بوده است.

[۱] مثلا این‌که اگر در مراحل میانی محاسبه‌ای با دو عدد نرمال، عددی زیرنرمال ظاهر شد ماشین چه خواهد کرد؟ و یا اینکه استفاده از بیت نگهبان یا بیت گردکردن یا بیت چسبناک (که در تضمین خاصیت بیشترین کیفیت چهار عمل اصلی است) دقیقا به چه صورت می‌باشد؟

۲. شکست ماموریت موشک آمریکایی در جریان جنگ خلیج فارس در ۲۵ فوریه ۱۹۹۱. موشک آمریکایی پاتریوت که در واقع یک موشک ضدموشک است، قرار بود پس از پرتاب از ظهران عربستان، موشک اسکادی را که توسط ارتش عراق پرتاب شده بود ردگیری کند اما بواسطه‌ی اشتباه در محاسبه‌ی زمان نتوانست موشک اسکاد را مورد اصابت قرار دهد و در نتیجه 28 سرباز آمریکایی کشته شدند. در واقع زمان محاسبه‌شده توسط ساعت داخلی سیستم در واحد 10برابر یک ثانیه اندازه‌گیری شده و نهایتا در عدد $\frac{1}{10}$ ضرب می‌شده تا زمان بر حسب ثانیه بدست آید. هرچند بسط دهدهی عدد $\frac{1}{10}$ فقط یک رقم بامعنا دارد، بسط دودویی آن نامتناهی است:

$$\begin{aligned}(0.1)_{10} &= 2^{-4} + 2^{-5} + 2^{-8} + 2^{-9} + 2^{-12} + 2^{-13} + \cdots \\ &= (1.10011001100\cdots)_2 \times 2^{-4} = (1.100\overline{1100})_2 \times 2^{-4}.\end{aligned}$$

در سیستم موشک پاتریوت، عدد $\frac{1}{10}$ بعد از قطع‌شدن در یک ثبات ۲۴ بیتی ذخیره می‌شد. این خطای گردکردن البته کوچک بوده است. اما از آنجا که باتری موشک پاتریوت به مدت ۱۰۰ ساعت در وضعیت آماده‌باش بود، زمان پرتاب موشکِ پاتریوت باید در 10برابر تعداد ثانیه‌های موجود در ۱۰۰ ساعت (یعنی $3600000 = 10 \times 60 \times 60 \times 100$) ضرب می‌شده و ضرب در این عدد بزرگ باعث می‌شود که خطای کوچکِ گردکردن، بزرگ شده و نهایتا موشک پاتریوت با تاخیری 0.34 ثانیه‌ای پرتاب شود. از سوی دیگر موشک اسکاد در هر ثانیه تقریبا 1676 متر را می‌پیماید و در نتیجه در مدت زمان 0.34 ثانیه، بیش از 500 مترا طی می‌کند و این فاصله خارج از برد پوشش‌داده شده توسط یک موشک پاتریوت است. درنتیجه موشک اسکاد نهایتا به هدف اصابت می‌کند.

۳. تغییر احزاب تشکیل دهنده‌ی پارلمان آلمان در سال ۱۹۹۲ در اثر خطای گردکردن. در سیستم پیچیده‌ی انتخابات آلمان اگر حزبی کمتر از پنج درصد آراء رابدست آورد، نمی‌تواند وارد پارلمان شده و کلیه آراء آن حزب حذف شده و کرسی‌های پارلمانی مربوطه بین سایر احزاب بصورت خاصی پخش می‌شود . پس از اعلام نتایج انتخابات ۵ آوریل ۱۹۹۲، اعلام می‌شود که حزب سبزها توانسته پنج درصد آراء را بدست آورد اما بعد از نیمه شب یکی از اعضای کمیته‌ی انتخابات متوجه شد که حزب سبزها در واقع توانسته بوده 4.97% آراء را بدست آورد، اما برنامه ای که محاسبات را انجام می‌داده، تنها یک رقم بعداز ممیز اعشار را پس از گرد کردن به سمت بالا نهایتا چاپ می‌کرده و به همین دلیل عدد 4.97% را به پنج درصد تبدیل کرده بود. پس از مشخص‌شدن این اشتباه حزب سبزها نتوانست وارد پارلمان شود و حزب SPD توانست اکثریت پارلمان را به خود اختصاص دهد.

۴.۱ عدد وضعیت مساله

می‌خواهیم بدانیم یک مساله‌ی ریاضی چقدر به خطاهای اندکِ گردکردن حساس است. برخی از پرسش‌های مرتبط و مهمی که مطرح می‌شوند عبارتند از:

- تا چه حد می‌توان روی میزان درستی جواب عددی یک مساله‌ی ریاضی حساب کرد؟
- آیا اصولا حل مساله‌ای که به خطاهای گردکردن حساسیت زیادی دارد با روش‌های عددی (که مرتبا درگیر خطاهای اندکِ گردکردن می‌باشند) ایده‌ی خوبی است؟

عدد وضعیت[۱] یک مساله، نشان می‌دهد که عدم اطمینان در داده‌های مساله تا چه حد می‌تواند جواب مساله را تغییر دهد، یعنی میزان حساسیت خروجی مساله را به تغییرات اندک در ورودی آن توصیف می‌کند. پس مفهوم عدد وضعیت برای یک مساله متناظر است با مفهوم پیوستگی برای یک تابع. (به زبانی نه کاملا دقیق) عدد وضعیت با تعیین نسبت

$$\frac{\text{اختلال نسبی خروجی}}{\text{اختلال نسبی ورودی}}$$

مشخص می‌شود.

بعنوان مثال، مساله‌ی تعیین مقدار تابع $f : \mathbb{R} \to \mathbb{R}$ در نقطه‌ی x را در نظر گرفته و فرض کنید $\tilde{x}$ با ایجاد اختلالی اندک در ورودی x حاصل شده باشد (مثلا به صورت $\tilde{x} = x + \Delta x$). واضح است که:

$$\frac{f(x) - f(\tilde{x})}{f(x)} = \frac{f(x) - f(\tilde{x})}{x - \tilde{x}} \times \frac{x}{f(x)} \times \frac{x - \tilde{x}}{x}$$

که در آن ارتباط بین اختلال ورودی و خروجی مشخص است:

$$\underbrace{\frac{f(x) - f(\tilde{x})}{f(x)}}_{\text{اختلال نسبی خروجی}} = \underbrace{\underbrace{\frac{f(x) - f(\tilde{x})}{x - \tilde{x}}}_{\text{تقریب مشتق تابع } f \text{ در نقطه‌ی } x} \times \frac{x}{f(x)}}_{\text{عدد وضعیت}} \times \underbrace{\frac{x - \tilde{x}}{x}}_{\text{اختلال نسبی ورودی}}$$

[۱] condition number

$\kappa_f(x)$ نماد مرسوم برای عدد وضعیت مساله‌ی ارزیابی تابع f در نقطه‌ی x است. با توجه به فرمول قبل داریم:

$$\frac{|f(x)-f(\tilde{x})|}{|f(x)|} \approx \kappa_f(x) \times \frac{|x-\tilde{x}|}{|x|} \tag{۸.۱}$$

که در آن

$$\boxed{\kappa_f(x) = \frac{|x|\ |f'(x)|}{|f(x)|}}$$

به طور کلی عدد وضعیت نسبی یک مساله را می‌توان از اندازه‌ی

$$\text{مشتق (خروجی بر حسب ورودی)} \times \frac{\text{ورودی}}{\text{خروجی}} \tag{۹.۱}$$

تخمین زد. مساله‌ای که عدد وضعیتش کوچک باشد را مساله‌ی خوش وضع می‌نامیم. به همین ترتیب یک مساله را بدوضع نامیم اگر عدد وضعیتش بزرگ باشد. اینکه چه عدد وضعیتی بزرگ محسوب می‌شود بسته به موقعیت، متفاوت است. در این مورد در ادامه بحث خواهیم کرد. بطور کلی اینکه ((چه عدد وضعیتی نگران‌کننده است؟)) مرتبط خواهد بود با این‌که چه میزان درستی از جواب مساله انتظار داریم و همچنین این‌که دقت دستگاه ممیزشناوری که از آن استفاده می‌کنیم چقدر است. توجه کنید که عدد وضعیت مساله موضوعی ریاضی است در بطن خود مساله و به خودی خود هیچ ارتباطی با الگوریتمی که بعدا برای حل آن مساله برخواهیم گزید ندارد.

از تعریف کلی‌ای که برای عدد وضعیت کردیم و یا همچنین از رابطه‌ی (۸.۱) می‌توان فهمید که عدد وضعیت، در واقع به طور تقریبی، میزان درشت‌شدنِ خطاهای گردکردن ورودی x را در خروجی $f(x)$ اندازه‌گیری می‌کند. چنانچه از طرفین رابطه‌ی (۸.۱) لگاریتم در مبنای ۱۰ بگیریم، خواهیم داشت:

$$-\log_{10}\left(\frac{|f(x)-f(\tilde{x})|}{|f(x)|}\right) \approx -\log_{10}\left(\frac{|x-\tilde{x}|}{|x|}\right) - \log_{10}\Big(\kappa_f(x)\Big) \tag{۱۰.۱}$$

در این رابطه نکات زیر حائز اهمیت هستند:

- $-\log_{10}\left(\frac{|f(x)-f(\tilde{x})|}{|f(x)|}\right)$ تقریبا برابر است با تعداد ارقام دهدهی یکسان $f(\tilde{x})$ و $f(x)$.

- همچنین $-\log_{10}\left(\frac{|x-\tilde{x}|}{|x|}\right)$ عبارت است از تعداد تقریبی ارقام دهدهی یکسان $\tilde{x}$ و x.

- از سوی دیگر طبق قضیه‌ی ۱.۲.۱ می‌دانیم که خطای نسبی حاصل از گردکردن هر عددی حقیقی به یک عدد ماشین با اپسیلون ماشین کراندار می‌شود.

- به یاد آورید که $-\log_{10}(\varepsilon_M)$ در قالب دوگانه‌ی آی-تریپل-ای تقریبا برابر است با ۱۶.

پس به عنوان یک قاعده‌ی سردستی، چنانچه عدد وضعیت یک مساله برابر با 10^k باشد آنگاه تعداد ارقام درستی که می‌توان (با دقت دوگانه) از خروجی مساله انتظار داشت تقریبا برابر خواهد بود با $16-k$. اگر عدد وضعیت مساله‌ای 10^8 باشد در بهترین حالت می‌توان حدود ۸ رقم درست از خروجی مساله انتظار داشت و یا اگر عدد وضعیت مساله‌ای 10^{16} باشد اصولا تلاش برای حل عددی آن مساله بیهوده است.

مثال ۶. در مورد میزان حساسیت مساله‌ی ارزیابی تابع $f(x)=\exp(-x)$ به خطاهای اندک (گردکردن) بحث کنید.

داریم

$$f'(x)=-\exp(-x)\to\kappa_f(x)=|\frac{x\exp(-x)}{\exp(-x)}|=|x|.$$

پس برای $x\approx 1$ مساله خوش وضع است. از سوی دیگر این مساله برای $x\approx 10^n$ وقتی n عددی بزرگ باشد بدوضع می‌باشد.

مثال ۷. در مورد میزان حساسیت مساله‌ی ارزیابی تابع $f(x)=\log(x)$ به اختلالی کوچک در نزدیکی $x\approx 1$ چه می‌توان گفت؟

داریم

$$f'(x)=\frac{1}{x}\to\kappa_f(x)=|\frac{xf'(x)}{f(x)}|=|\frac{x\frac{1}{x}}{\log(x)}|=|\frac{1}{\log(x)}|.$$

پس برای $x\approx 1$ چون $\log(x)$ عددی کوچک خواهد بود، مساله بدوضع است.

نکته‌ی ۴. معمولا هرگاه از عدد وضعیت صحبت می‌شود منظور عدد وضعیت نسبی است. با این حال عدد وضعیت مطلق که معمولا با نماد $\hat{\kappa}_f(x)$ نشان داده می‌شود نیز گاها مفید است (مثلا آنچنان که در مساله‌ی ریشه یابی در فصل بعد خواهیم دید).

$$\underbrace{f(x)-f(\tilde{x})}_{\text{اختلال مطلق خروجی}} = \underbrace{\frac{f(x)-f(\tilde{x})}{x-\tilde{x}}}_{\text{تقریب مشتق تابع } f \text{ در نقطه‌ی } x} \times \underbrace{x-\tilde{x}}_{\text{اختلال مطلق ورودی}}.$$

پس عدد وضعیت مطلق مساله‌ی ارزیابی تابع f در نقطه‌ی x برابر است با

$$\hat{\kappa}_f(x) = |f'(x)|.$$

۱.۴.۱ پدیده‌ی حذف

منظور از حذف[۱]، خنثی‌شدن ارقام بامعنا در حساب با دقت متناهی است وقتی حاصل تفریق دو عدد هم‌علامتِ نزدیک به هم (اعداد با ارقام پیشروی یکسان) محاسبه می‌شود. همین اتفاق در جمع با دقت متناهی دو عدد با علامت‌های متضاد که از نظر اندازه به هم نزدیک هستند نیز رخ می‌دهد. فرض کنید k رقم پیشرو از ارقام بامعنای دو عدد ماشین x و y یکسان باشند و بخواهیم حاصل تفریق آنها را در حساب ممیز شناور بیابیم. داریم:

$$\begin{array}{l} \quad (d_0.d_1d_2\cdots d_{k-1}\ d_k\ d_{k+1}\cdots d_{p-1})_\beta \times \beta^e \\ \ominus\ (d_0.d_1d_2\cdots d_{k-1}\ c_k\ c_{k+1}\cdots c_{p-1})_\beta \times \beta^e \\ \hline =\ (0\ .\ 0\ 0\ \cdots\ 0\ \ f_k\ f_{k+1}\cdots f_{p-1})_\beta \times \beta^e \end{array}$$

که وقتی نرمال می‌شود برابر خواهد بود با

$$(f_k.f_{k+1}\cdots f_{p-1}??\cdots ?)_\beta \times \beta^{e-k}$$

پس k رقم داریم که بجایشان هرچه بگذاریم نادرست خواهند بود. پس دو عدد x و y دارای p رقم بامعنا بودند ولی حاصل تفریق آن‌ها تنها دارای $p-k$ رقم بامعنای درست است و k رقم پایانی آن صفرهایی خواهند بود که به صورت مصنوعی جلوی مانتیس قرار گرفته‌اند.

در زنجیره‌ای از محاسبات علمی با حساب ممیز شناور قویا ممکن است x و y خود تقریب‌هایی از اعدادی دیگر بوده باشند. در چنین موقعیتی قاعدتا امکان بیشتری وجود دارد که ارقام پیشروی x و y درست بوده باشند و ارقام پایانی آن‌ها از اعداد اصلی متفاوت و در نتیجه غلط بوده باشند چرا که ارقام پایانی یک عدد در محاسبات ممیزشناور بیشترین تاثیر را از خطاهای گردکردن می‌پذیرند. آنچه نگرانی از پدیده‌ی حذف را بیش از پیش تشدید می‌کند این است که $p-k$ رقم بامعنایی که واقعا حاصل اجرای یک

[۱] cancellation

محاسبه بودند، دقیقا از همان ارقامی از x و y به دست آمده‌اند که احتمال نادرست‌بودنشان بیشتر بوده است. پس حتی به درستی آن رقم‌های غیرمصنوعی نیز چندان نمی‌توان خوشبین بود! در واقع گاهی بجای حذف(خنثی‌شدن) از نام پدیده‌ی ((حذف فاجعه‌بار[۱])) استفاده می‌شود.

بعنوان یک مثال عددی به یاد آورید که در قسمتی از حل تمرین ۲ قرار بود دو عدد

$$b = +3.3678429 \times 10^{+1}$$
$$c = -3.3677811 \times 10^{+1}$$

را در دستگاه $F_{10,8}^{-5,+5}$ با هم جمع کنیم. داریم:

$$\begin{array}{r} 3.3678429 \times 10^{+1} \\ \ominus\ 3.3677811 \times 10^{+1} \\ \hline = 0.0000618 \times 10^{+1} \end{array}$$

و با نرمال کردن آن، پاسخِ 6.1800000×10^{-4} به دست می‌آید[۲]. همانطور که می‌بینیم b و c دارای هشت رقم بامعنا هستند ولی حاصل‌جمع، قبل از نرمال‌سازی تنها دارای سه رقم بامعناست چرا که پنج رقم پیشروی b و c با هم خنثی شده‌اند. پس دو عددی که هشت رقم بامعنا دارند را با هم جمع کرده و نهایتا بعد از نرمال‌سازی پاسخی به دست آورده‌ایم که تنها سه رقمش درست است.

قبل از پایان بحث پدیده‌ی حذف، خوب است ارتباط احتمالی بین خطرناک‌بودن پدیده‌ی حذف با عدد وضعیت مساله‌ی ریاضیِ تفریق دو عدد را بررسی کنیم. مساله‌ی محاسبه‌ی مقدار تابع $f(x) = x - c$ که

---

[۱] catastrophic cancellation

[۲] همین پاسخ را می‌شد از راه زیر نیز به دست آورد:

$$\begin{aligned} b \oplus c &= 3.3678429 \times 10^{+1} \ominus 3.3677811 \times 10^{+1} \\ &= fl(6.18000000028940 \times 10^{-04}) = 6.1800000 \times 10^{-04}. \end{aligned}$$

در آن c عددی ثابت است را درنظر بگیرید. داریم:

$$\kappa_f(x) = \frac{|x|(1)}{|x-c|} = |\frac{x}{x-c}|$$

اگر x و c خیلی به هم نزدیک باشند، مخرج کسر بالا خیلی به صفر نزدیک شده و در نتیجه عدد وضعیت، خیلی بزرگ خواهد شد. در غیر این‌صورت مساله‌ی تفریق دو عدد، یک مساله‌ی بدوضع نخواهد بود. پس ناخوشایند بودن پدیده‌ی حذف شگفت‌آور نیست: این متناظر با مساله‌ای بدوضع است.

تمرین ۳. بار دیگر تمرین ۲ را مرور کنید. هم در محاسبه‌ی $(a \oplus b) \oplus c$ و هم در محاسبه‌ی $a \oplus (b \oplus c)$ پدیده‌ی حذف رخ می‌دهد. در توضیحات قبل مشخص کردیم که در کدام قسمت از فرمول $a \oplus (b \oplus c)$ پدیده‌ی حذف رخ می‌دهد.

۱. درکدام قسمت از فرمول دیگر یعنی $(a \oplus b) \oplus c$ پدیده‌ی حذف رخ می‌دهد؟

۲. کدام‌یک از این دو پدیده‌ی حذف ذکرشده مخرب‌تر است؟ (مثلا با تعیین عدد وضعیت هر یک از دو رخداد حذف)

این امکان وجود دارد که پدیده‌ی حذف تاثیر مخرب زیادی نداشته باشد. ممکن است بتوان به سادگی آنرا از بین برد و یا ممکن است به راحتی قابل برطرف‌کردن نباشد. اما باتوجه به فراوانیِ عمل تفریق در مسائل و الگوریتم‌های محاسبات علمی، بهتر است در زمان طراحی الگوریتم‌های عددی مواظب رخداد آن بوده و حتی الامکان از آن پیشگیری کنیم. بعدا در بحث پایداری عددی الگوریتم‌ها، پدیده‌ی حذف را در فرمول دبیرستانی دلتا برای یافتن ریشه‌ی چندجمله‌ای‌های درجه دو شناسایی و راهکار جلوگیری از آن را خواهیم دید.