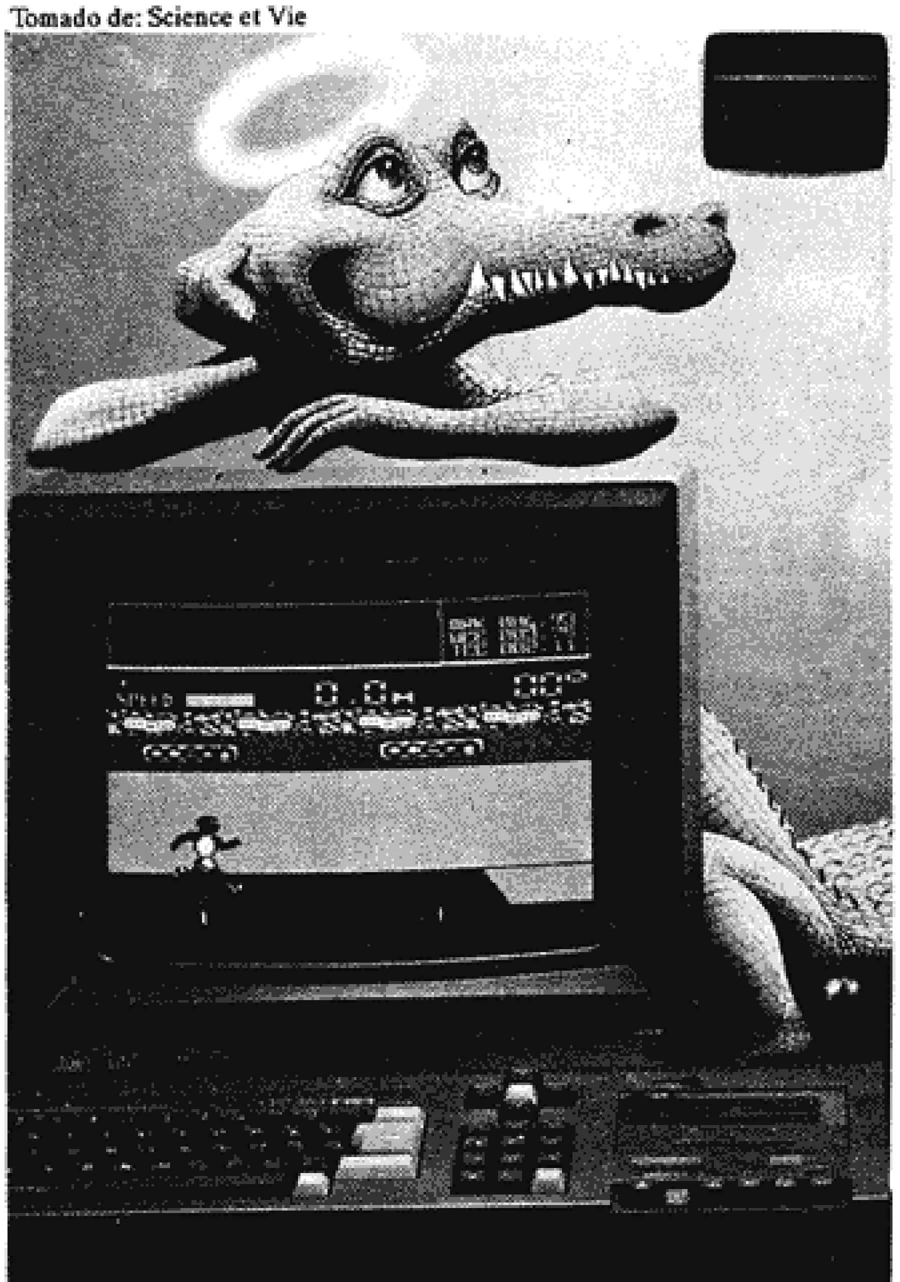
El usuario (casi) siempre escoge la formulación matemática menos estable para resolver su problema. Nash

¿Son incuestionables los resultados computacionales?

HUMBERTO MADRID V.
JOSE GUERRERO G.*

Tomado de: Science et Vie



PROLOGO

Nuestra intención al elaborar este trabajo es mostrar que si bien la computadora representa un instrumento de gran utilidad en la solución de problemas –llegando a ser indispensable en muchos casos–, su uso acrítico junto con la mitificación que este uso induce en una gran cantidad de usuarios, puede llevar a situaciones tan lamentables como el creer que un resultado obtenido computacionalmente es siempre correcto cuando en realidad en muchas ocasiones está muy lejos de serlo.

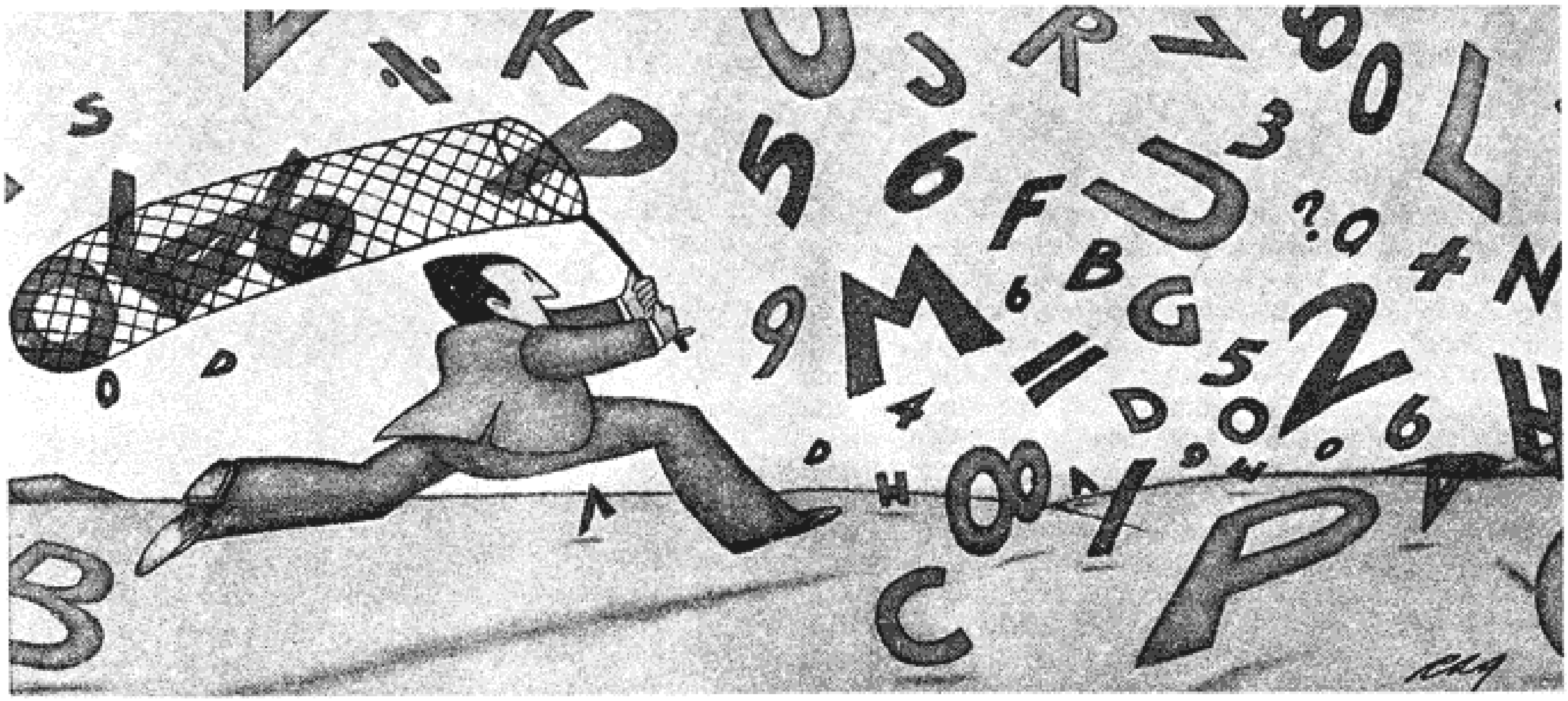
La presentación que se hace trata de ser lo menos técnica posible de tal forma que las ideas que se intentan transmitir puedan llegar a un público amplio.

I *Computación y análisis numérico*

Un hecho importante con relación a nuestro mundo actual, lo constituye el gran impacto que los avances de la computación han tenido sobre áreas cada vez más extensas de la actividad humana. Así, la investigación científica, la enseñanza, la industria, la administración y algunas manifestaciones artísticas contemporáneas se hallan estrechamente ligadas a la computación, y es éste un fenómeno que tiende a generalizarse cada vez más.

* *Profesores del Depto. de Matemáticas de la Facultad de Ciencias de la UNAM.*

Tomado de PC Magazine



Por lo que toca al campo de las matemáticas, la interacción matemáticas–computación ha sido extraordinariamente rica, siendo uno de sus principales frutos el surgimiento del moderno análisis numérico, cuyo desarrollo en las últimas décadas ha sido tan acelerado que sólo le son comparables la estadística, la investigación de operaciones y la computación misma. Un producto importante de este desarrollo son los programas para calcular soluciones numéricas de problemas matemáticos, presentes en cualesquiera de los centros de cómputo donde se hacen cálculos científicos.

II *Mételo a la computadora*

Una situación usual en la vida diaria de alumnos y profesores es la de enfrentarse a problemas cuya solución implica cálculos tediosos, y cuando esto sucede no falta una voz que sugiera: *mételo a la computadora*. Los resultados de optar por esta vía merecen algunas reflexiones, según se muestra en los siguientes ejemplos:

i) restas pánicas.

Problema: dados

$$x = .999995$$
$$y = .999994$$

calcular $x - y$.

El cálculo de la resta da como resultado

$$x - y = r = 10^{-6}$$

Consideremos ahora las siguientes aproximaciones de x, y:

$$\tilde{x} = .999997$$
$$\tilde{y} = .999992$$

De estos valores resulta

$$\tilde{r} = \tilde{x} - \tilde{y} = 5 * 10^{-6}$$

Por otra parte, los errores relativos cometidos en $\tilde{x}, \tilde{y}$ resultan ser

$$e_x = \left|\frac{x - \tilde{x}}{x}\right| \approx 2 * 10^{-6}$$
$$e_y = \left|\frac{y - \tilde{y}}{y}\right| \approx 2 * 10^{-6}$$

Así pues, podríamos considerar que las perturbaciones en x, y son razonablemente pequeñas, lo que podría llevarnos a pensar que el correspondiente e_r fuese también pequeño. Sin embargo resulta que

$$e_r = \left|\frac{r - \tilde{r}}{r}\right| \approx 4 * 10^{0}$$

Esto es, hemos llegado a que, errores del orden de 10^{-6} en los datos producen un error del orden de 10^{0} en el resultado.

Pero ¿y qué con todo esto?, podría preguntar algún lector.

Tratemos entonces de aclarar este drama numérico.

En primer lugar obsérvese que, tanto para $x, \bar{x}$ como para $y, \bar{y}$ hemos calculado los llamados errores relativos e_x, e_y respectivamente. Estos valores, además de tener ciertas propiedades interesantes, nos permiten saber qué tan buenas aproximaciones de x, y son $\bar{x}, \bar{y}$ en un sentido muy preciso.

En efecto, obsérvese que

$$e_x \leq 10^{-5}$$
$$e_y \leq 10^{-5}$$

y en ambos casos $\bar{x}, \bar{y}$ coinciden con x, y en los primeros 5 dígitos; esto es, ambas aproximaciones tienen 5 cifras correctas de x, y respectivamente. Siguiendo este esquema es directo concluir que en r no tenemos ninguna cifra correcta de r, cosa que en efecto sucede. Al fenómeno antes ejemplificado se le conoce con el nombre de cancelación catastrófica y se presenta cuando (como en nuestro caso) se restan dos cantidades muy parecidas. Un caso típico en el que este fenómeno se presenta es cuando se intenta calcular numéricamente las raíces de una cuadrática

$$F(x) = ax^2 + bx + c$$

mediante el conocido algoritmo

$$x_i = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

El lector interesado puede recurrir a la bibliografía citada al final para mayor información sobre este punto.

ii) No todos los caminos conducen a Roma

Presentamos ahora un problema cuya formulación es una sucesión numérica de las llamadas recurrentes. Aunque el ejemplo aquí presentado es en términos de integrales, al lector no familiarizado con este tema le podemos decir que existen muchos problemas cuya formulación matemática es similar a la aquí presentada. A estos lectores les pedimos que concentren su atención en la fórmula (1). Vayamos pues a lo que nos interesa, que en este caso es calcular los valores de la integral

$$y_n = \int_0^1 \frac{x^n}{x+5} dx, \qquad n \geq 0$$

para diversos valores de n. Por supuesto, podríamos intentar el cálculo directamente o bien usar algún método numérico para conseguir una aproximación de nuestra integral, sin embargo el problema resulta más sencillo si notamos que

$$\begin{aligned} y_n + 5y_{n-1} &= \int_0^1 \left(\frac{x^n}{x+5} + \frac{5x^{n-1}}{x+5} \right) dx \\ &= \int_0^1 \frac{x^{n-1}(x+5)}{x+5} dx \\ &= \int_0^1 x^{n-1} dx = \frac{x^n}{n} \Big|_0^1 = \frac{1}{n} \end{aligned}$$

Lo realmente importante del desarrollo anterior es la relación

$$y_n + 5y_{n-1} = \frac{1}{n} \qquad (1)$$

Tomado de: Omni, 1981

Esta relación nos indica que si conocemos el valor de alguna y_n para alguna n particular, podemos entonces calcular los valores de las integrales que le siguen o que le anteceden. Por ejemplo, si conocemos y_{20} podemos conocer y_{21} usando nuestra fórmula (1) que para este caso nos dice que

$$y_{21} = \frac{1}{20} - 5y_{20}$$

En forma análoga podemos conocer y_{22}, y_{23}, etc. Pero también partiendo de nuestra relación (1) podemos conocer y_{19} dado que

$$y_{20} + 5y_{19} = \frac{1}{20}$$

de donde resulta

$$y_{19} = \frac{1}{5}\left(\frac{1}{20} - y_{20}\right)$$

y una vez tenido esto, podemos proceder a calcular recursivamente y_{18}, y_{17}, etc. Ahora bien, todo esto partir del supuesto de que conocemos y_{20}. Sin embargo, conocerla en la práctica no resulta tan fácil. Pero lo que sí es fácil de calcular es y_0, ya que

$$\begin{aligned} y_0 = \int_0^1 \frac{1}{x+5} dx &= ln(x+5) \Big|_0^1 \\ &= ln(6) - ln(5) \\ &= ln\left(\frac{6}{5}\right) \approx 0.18232156 \end{aligned}$$

Usando este valor inicial podemos entonces calcular los valores $y_1, y_2, y_3, \cdots$ usando la relación

$$y_n = \frac{1}{n} - 5y_{n-1} \qquad (2)$$

Una posibilidad para hacer esto sería realizar los cálculos auxiliándonos de papel y lápiz. Otra, mucho más veloz es *¡pasarle el problema a la micro!*. Por nuestra parte elaboramos un programa que fue ejecutado en una microcomputadora PC-compatible, obteniendo los siguientes resultados:

n	y_n	n	y_n
1	0.08839222	11	0.01408789
2	0.05803892	12	0.01289387
3	0.04313873	13	0.01246373
4	0.03430633	14	0.00915994
5	0.02846836	15	0.02086697
6	0.02432490	16	-0.04183485
7	0.02123264	17	0.26799777
8	0.01883679	18	-1.28443329
9	0.01692715	19	6.47479804
10	0.01536424	20	-32.32399022

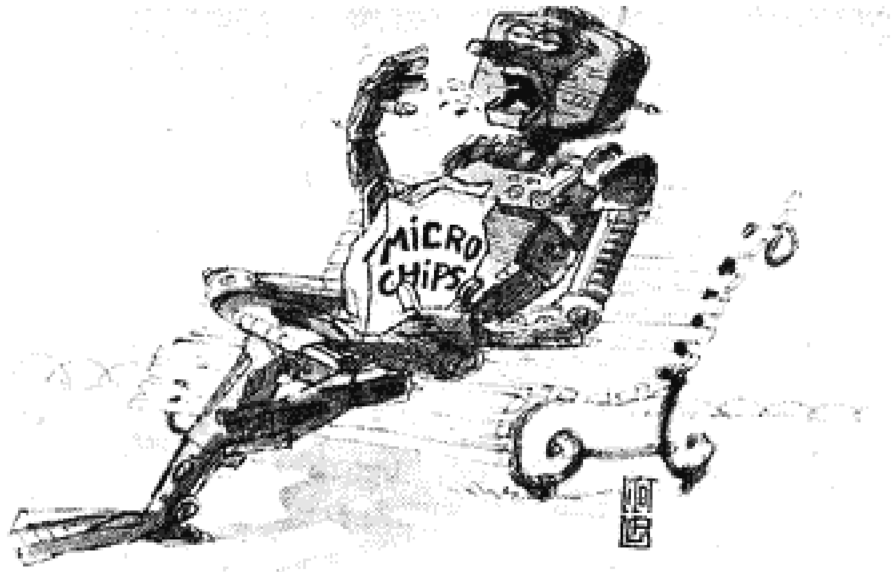
¿Y bien? ¿Podemos irnos felices a casa con nuestros resultados? ¡Nada de eso!, porque resulta que y_n debe ser positiva para todo valor de $n \geq 0$, dado que en el intervalo [0,1] el integrando

$$\frac{x^n}{x+5}$$

es una función continua positiva, y debido a ello es que y_n debe ser positiva. ¿Cuál es entonces la razón de estos resultados absurdos? ¿Será que nuestro programa está mal escrito? A continuación lo presentamos:

```
program recursion (input, output)

var s:real;

var n, nmax:integer;

    begin

    writeln("NUMERO DE TERMINOS A CALCULAR");

    readln(nmax);

    s:= ln(6/5); {valor inicial}

        writeln("VALOR INICIAL", s:14:8);

            for n:=1 to nmax do

                begin

                s:=1/n - 5*s;

                writeln(n:3, s:14:8);

            end;

end.
```

El lector que conozca un poco de programación podrá darse cuenta que el programa es correcto. Al lector perspicaz le podemos asegurar que las fórmulas (1) y (2) son también correctas. ¿Cuál es entonces la bronca?



Tomado de: Omni, 1982

El problema radica en la fórmula (2). Pero no piense usted mal, estimado lector. No se trata de un timo; dicha fórmula es impecable desde el punto de vista matemático, pero resulta muy inadecuada desde el punto de vista numérico. Veamos por qué. El valor y_0, por ser un número irracional, no es representable en forma exacta en una computadora, por lo que de hecho ésta trabaja internamente con una aproximación —llamémosla $\bar{y}_0$—. Sea ahora $\epsilon = y_0 - \bar{y}_0$ la diferencia entre el valor exacto y su aproximación numérica (ϵ es llamado el error de aproximación inicial). Supongamos además, con el afán de ser lo menos complicados posible, que en los cálculos posteriores no se comete ningún error de aproximación. Denotemos ahora por $\bar{y}_1, \bar{y}_2 \cdots$ etcétera a las aproximaciones de $y_1, y_2, \cdots$ calculadas por la computadora. Tenemos entonces que $\bar{y}_1 = 1 - 5\bar{y}_0 = 1 - 5(y_0 - \epsilon) = 1 - 5y_0 + 5\epsilon = y_1 + 5\epsilon$, esto es, resulta que en el cómputo de y_1 a partir de y_0 se comete un error de 5ϵ. Procediendo en forma similar para y_2, y_3, etc. resulta que

$$\bar{y}_n = y_n + (-1)^{n+1}5^n\epsilon$$

Lo que esta fórmula nos dice es que $|y_n - \bar{y}_n| = 5^n|\epsilon|$; o bien, dicho en palabras, que el error en el paso n es 5^n veces el error inicial. Así pues, a medida que crece n, el error de aproximación inicial se propaga en forma exponencial, por lo que, en unos cuantos pasos el error correspondiente deja de ser despreciable y esto a la larga conduce a resultados cada vez más incorrectos. El fenómeno descrito con este ejemplo se conoce como inestabilidad algorítmica, y a él nos referiremos más adelante.

Y ahora...... ¿quién podrá ayudarnos?

En nuestro caso, el problema se debe a que en cada paso el error de aproximación inicial se multiplica por un factor de 5, por lo que, aunque el valor de ϵ sea pequeño, a la larga el error de aproximación será grande; y todo ello es debido a la forma como hemos calculado los valores de y_1, y_2, etc.; esto es, la raíz de nuestro problema está en la fórmula (2). Sería conveniente tratar de ver si es posible plantearse los cálculos de otra forma. Y en efecto, resulta que nuestro ejemplo acepta un tratamiento que nos permite salvar estos problemas. Para ello obsérvese que la fórmula (2) puede ser reescrita como

$$y_{n-1} = \frac{1}{5n} - \frac{1}{5}y_n \qquad (3)$$

Si ahora aplicamos esta nueva fórmula para el cálculo de nuevos valores, tendremos un fenómeno totalmente opuesto al anterior; es decir, el error inicial de aproximación disminuirá en cada paso en vez de crecer. Para mostrar esto, supongamos que conocemos, por ejemplo, una aproximación de y_{20}, que llamaremos $\bar{y}_{20}$. Tendremos entonces como error de aproximación $\epsilon = y_{20} - \bar{y}_{20}$. Usando ahora la fórmula (3) tendremos

$$\begin{aligned}\bar{y}_{19} &= \frac{1}{5(20)} - \frac{1}{5}\bar{y}_{20} \\ &= \frac{1}{5(20)} - \frac{1}{5}(y_{20} - \epsilon) \\ &= \frac{1}{5(20)} - \frac{1}{5}y_{20} + \frac{1}{5}\epsilon \\ &= \frac{1}{5(19)} + \frac{1}{5}\epsilon,\end{aligned}$$

de donde resulta

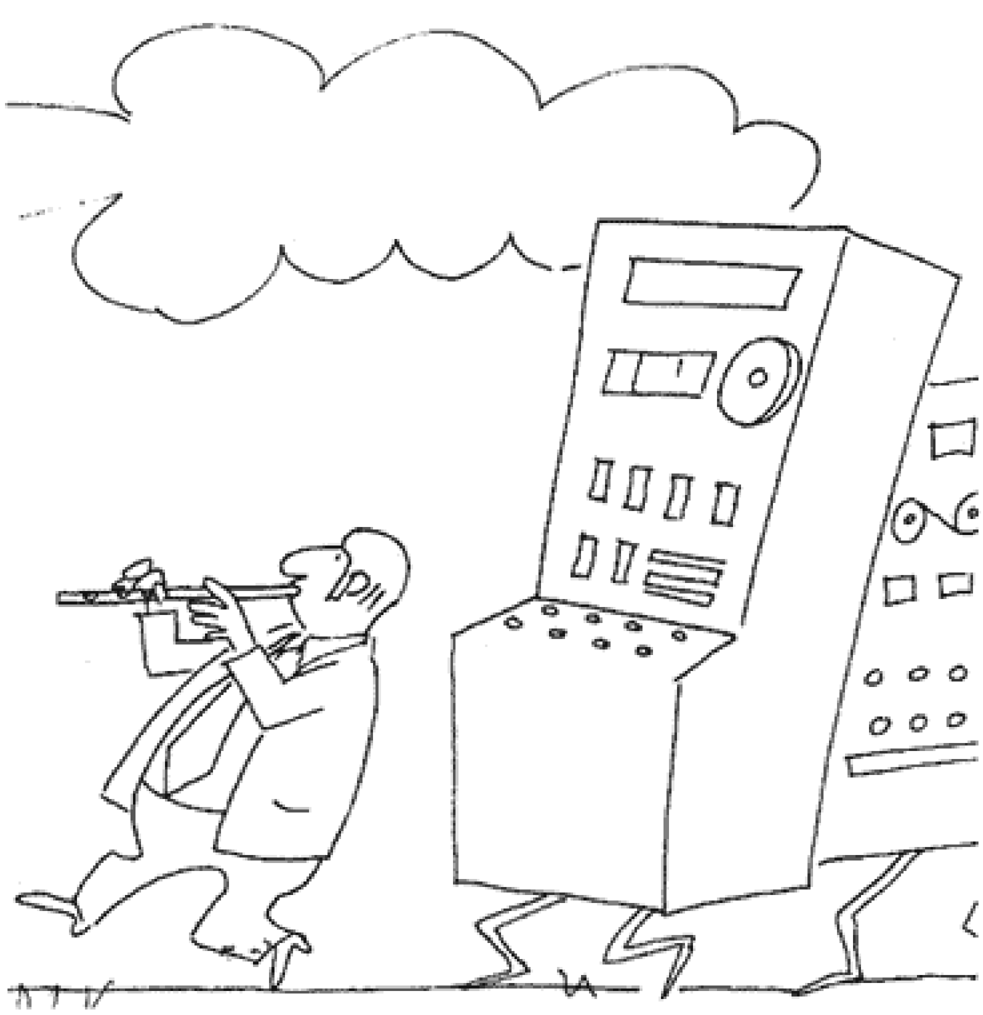
$$\bar{y}_{19} - y_{19} = \frac{1}{5}\epsilon$$

Esto es, el error de aproximación disminuye en 1/5 . Si se prosigue en esta forma el error disminuye 1/5 en cada paso. En general, si conocemos una aproximación $\bar{y}_n$ de y_n, con un error de aproximación $\epsilon = \bar{y}_n - y_n$, tendremos que

$$\bar{y}_{n-k} = y_{n-k} + \frac{(-1)^k}{5^k}\epsilon$$

es decir,

$$|\bar{y}_{n-k} - y_{n-k}| = \frac{1}{5^k}|\epsilon|$$



Esto significa que, a medida que k crece, los valores calculados computacionalmente son cada vez más exactos. El nuevo camino que hemos propuesto vía la fórmula (3) es satisfactorio, según se ha mostrado teóricamente. Sin embargo persiste un problema. ¿Cómo encontrar un valor aproximado de y que nos permita inicializar nuestro proceso?. Recurriendo nuevamente a nuestro problema original, no es difícil demostrar que $0 < y_n < \frac{1}{n}$ (esto se puede deducir partiendo de que en $[0, 1]$, $x + 5 \geq x$, y de las propiedades básicas de la integral). De acuerdo a lo anterior $0 < y_{20} < 1/20$. En principio cualquier número entre 0 y 1/20 puede ser usado como una razonable aproximación. A continuación se presentan algunos resultados computacionales obtenidos usando la fórmula (3) y dos valores iniciales de y_{20}.

1) Valor inicial $y_{20} = 0.05$		2) Valor inicial $y_{20} = 0.00$	
n	y_n	n	y_n
19	0.00000000	19	0.01000000
18	0.01052632	18	0.00852632
17	0.00900585	17	0.00940585
16	0.00996354	16	0.00988354
15	0.01050729	15	0.01052329
14	0.01123187	14	0.01122867
13	0.01203934	13	0.01203998
12	0.01297675	12	0.01297662
11	0.01407132	11	0.01407134
10	0.01536755	10	0.01536755
9	0.01692649	9	0.01692649
8	0.01883692	8	0.01883692
7	0.02123262	7	0.02123262
6	0.02432491	6	0.02433491
5	0.02846835	5	0.02846835
4	0.03430633	4	0.03430633
3	0.04313873	3	0.04313873
2	0.05803892	2	0.05803892
1	0.08839222	1	0.08839222
0	0.18232156	0	0.18232156

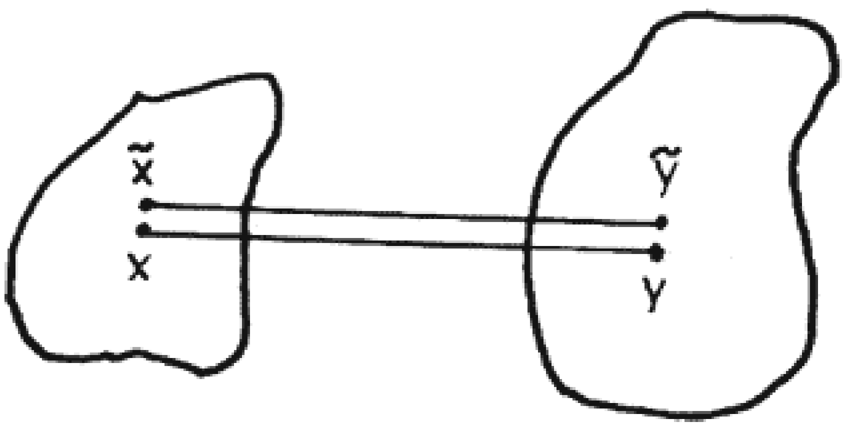
En ambos casos se obtiene para y_0 un valor con ocho dígitos correctos.

III La teoría sirve, aún después de ahogado el niño

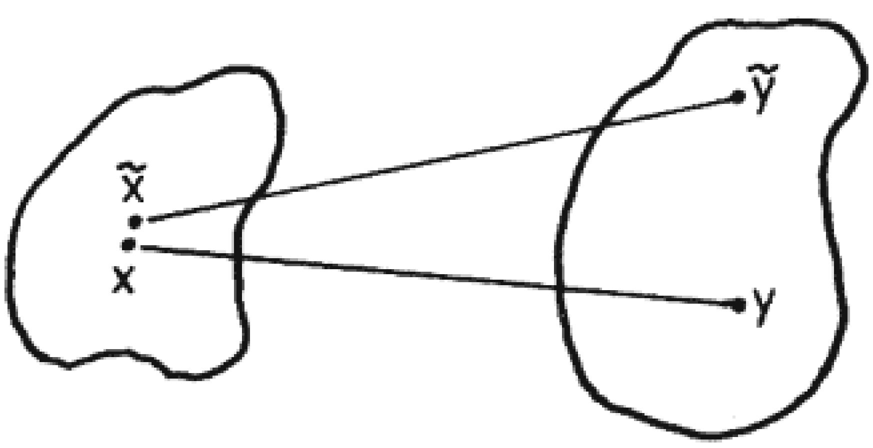
En general, en un problema numérico, se tiene como punto de partida una colección $(x_1, x_2, \cdots, x_m)$ de datos relativos al problema, y si éste tiene solución única entonces dichos datos deberán producir una colección $(y_1, y_2, \cdots, y_n)$ de resultados. Esto es, se puede decir que existe una relación "funcional" entre datos y resultados; lo que puede simbolizarse como

$$x = (x_1, \cdots, x_m) \xrightarrow{\quad f \quad} (y_1, \cdots, y_n) = y$$

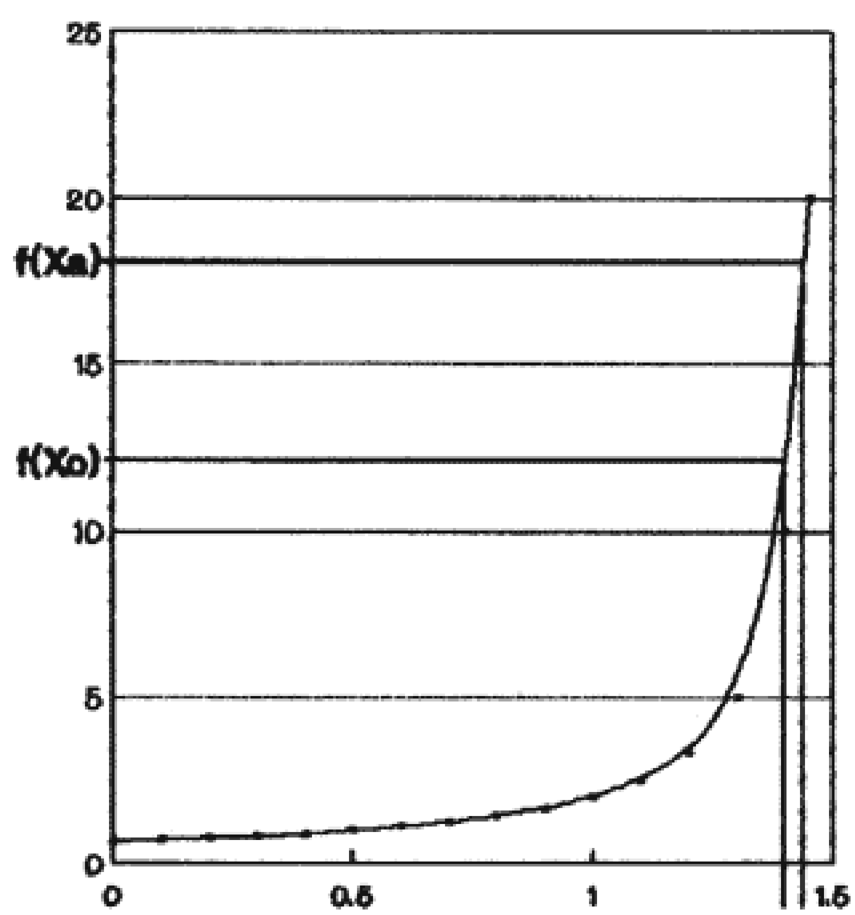
El interés consiste en calcular una "buena" aproximación de $(y_1, \cdots, y_n)$ a partir de los datos $(x_1, \cdots, x_m)$. Y es aquí justamente donde comienzan los problemas, ya que en general lo que se tiene no es exactamente x sino una —digamos "buena"— aproximación, a la que llamaremos $\bar{x}$. Esto establece de inmediato la siguiente situación: si a partir de x el resultado es y, entonces si partimos de $\tilde{x}$ deberemos llegar a una solución $\tilde{y}$. Ahora bien, nosotros desearíamos que se satisficiera algo del siguiente estilo



Es decir, nos gustaría que si $\tilde{x}$ está próxima a x, entonces la correspondiente $\tilde{y}$ estuviera igualmente próxima a y. Sin embargo resulta que en general esto no es cierto, pues bien puede darse la siguiente situación



y esto puede ser así, independientemente del camino que se siga para calcular $\tilde{y}$. Presentamos enseguida un ejemplo al respecto. Considérese el problema de evaluar la función $f(x)$ en el punto x_0. Gráficamente se tiene

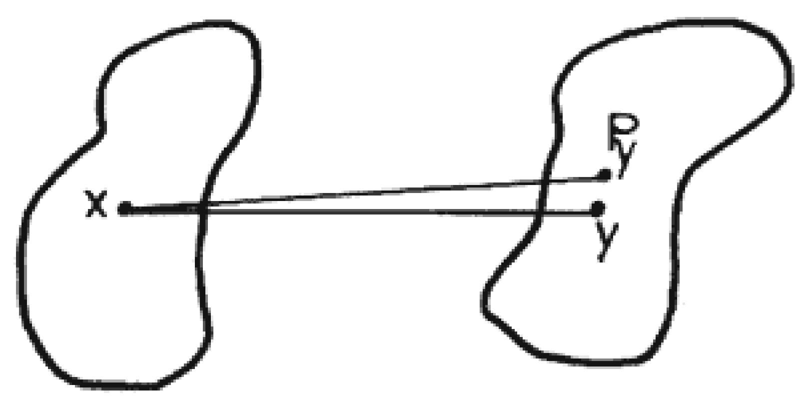


como se observa, aún a pesar de que la evaluación en x_a se realice en forma exacta, puede ser que la diferencia entre $f(x_0)$ y $f(x_a)$ sea mucho mayor que la diferencia entre x_0 y x_a. Dicho en otras palabras: dependiendo de f y x_0, puede suceder que un error "pequeño" en los datos nos produzca un error "grande" en los resultados, y ello sin importar el camino seguido en el cálculo de la solución.

El fenómeno antes descrito es de *ocurrencia generalizada* en Análisis Numérico y se le conoce como problema de *mal condicionamiento*; e igual puede presentarse al evaluar una función como al resolver un sistema de ecuaciones o calcular las raíces de un pólinomio. El efecto de cancelación catastrófica mostrado en el ejemplo de la resta de cantidades parecidas es un caso particular de mal condicionamiento.

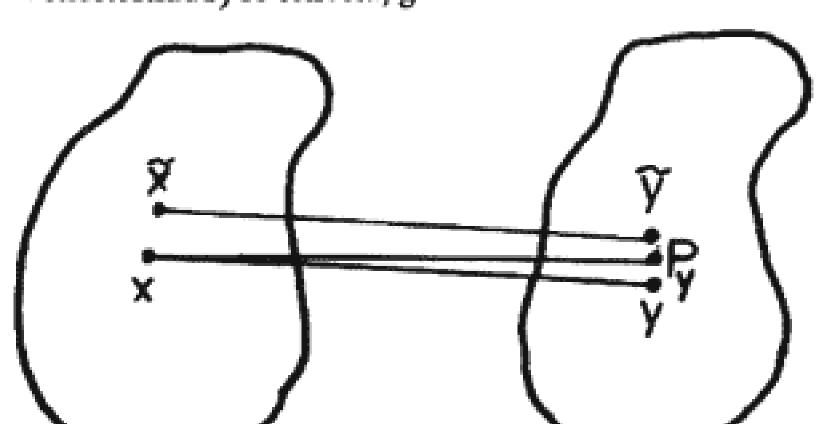
No es nuestro interés discutir en este trabajo cada uno de los casos en los que éste fenómeno se presenta —así como sus posibles curas—, sino sólo mostrarlo como una fuente importante de resultados erróneos cuando se intenta resolver un problema numérico a tontas y a locas; y esto con el fin de que el lector reflexione un poco cuando se enfrenta a un problema numérico, dado que —como hemos visto—, el drama puede empezar con el problema mismo.

Pero lo anterior no es todo. Resulta que una mala solución puede deberse también al procedimiento seguido para calcularla. En principio uno esperaría que al ejecutar un proceso para calcular la solución de un problema, el resultado fuera suficientemente cercano a la solución teórica (ver figura)

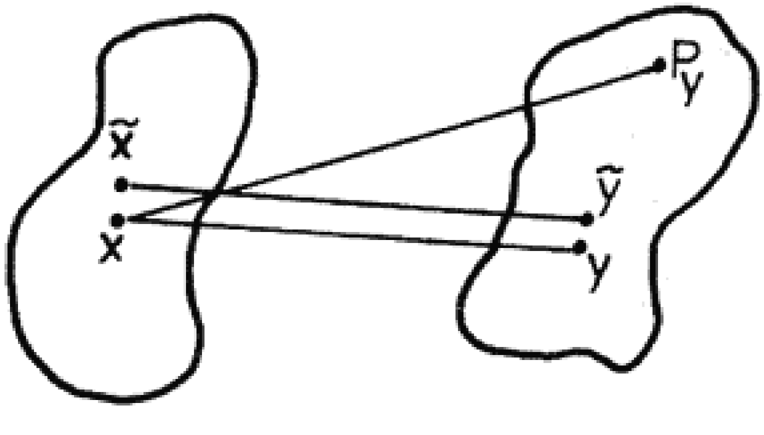


Esta situación es plenamente razonable dado que los cálculos involucrados en el proceso de solución pueden inducir un resultado que difiera de la solución exacta debido a que en general la computadora opera con aproximaciones de números reales. De aquí se desprenden distintas posibilidades:

i) Si el problema a resolver es bien condicionado (si no es mal condicionado) se tendría, gráficamente



El procedimiento P produce un resultado P_y "próximo" a $\tilde{y}$, que a su vez es resultado exacto para los datos $\tilde{x}$. Todo procedimiento que satisfaga estas propiedades se conoce como *algoritmo numéricamente estable*; y por el contrario, un algoritmo será inestable si no produce un resultado cercano a la solución exacta del problema con datos "cercanos". (ver figura).

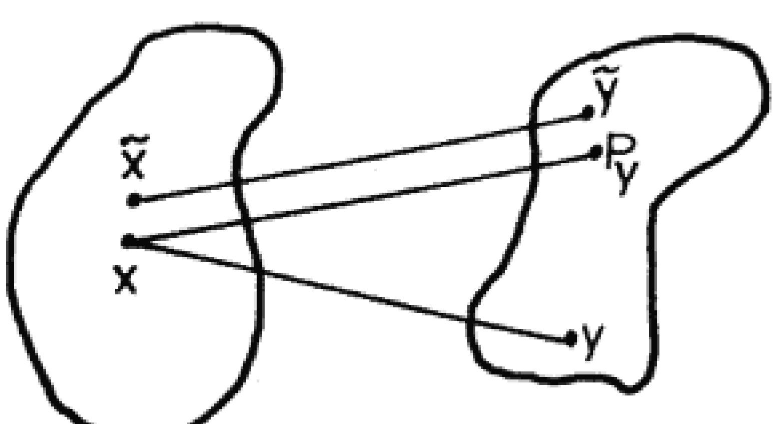


Podríamos resumir lo anterior como sigue:

a) Si estamos ante un problema bien condicionado y aplicamos un algoritmo numéricamente estable, es de esperar una buena solución.

b) Si el problema en cuestión es bien condicionado y aplicamos un algoritmo inestable, en general la solución será mala; y esta situación puede agravarse cada vez más dependiendo de las dimensiones del problema. En el ejemplo relativo al cálculo de la integral visto en la sección anterior se tiene el caso de un problema bien condicionado cuya solución puede resultar "buena" o "mala" dependiendo de la utilización de un algoritmo estable o uno inestable.

ii) Si el problema a resolver es mal condicionado y se aplica un algoritmo estable, la situación —gráficamente— sería



aquí se muestra que puede tenerse una solución "mala" a pesar de haber sido bien calculada. El caso de atacar un problema mal condicionado con un algoritmo inestable se deja a la imaginación del lector.

IV A manera de despedida

Es alarmante el número de personas que usan la computadora en forma irreflexiva, pensando que por el hecho de programar una fórmula encontrarán la solución correcta a su problema. Estas personas sólo validan de forma contundente el axioma de Nash citado al inicio de este trabajo. Es por ello que resulta recomendable el tener ciertos cuidados al escribir un programa o usar un paquete. Existen paquetes en los que se recoge la experiencia de gentes que durante años han trabajado en problemas de estabilidad y mal condicionamiento. Hay que conocerlos. Pero igualmente importante resulta el que el usuario reflexione sobre la naturaleza de su problema y se informe sobre las características numéricas de los algoritmos que piense usar, antes de confiar ciegamente en los resultados. Algunas recomendaciones finales: no calcular determinantes para resolver sistemas de ecuaciones lineales; no calcular los valores propios de una matriz evaluando directamente su polinomio característico; no usar la fórmula por radicales para calcular las raíces de un polinomio de tercer grado. Si tiene problemas consulte con su analista numérico más cercano.

El lector interesado podrá ampliar su visión sobre todo lo antes tratado recurriendo a los trabajos citados en la bibliografía.

BIBLIOGRAFIA

1. Dahlquist, G., and A., Björck, **Numerical Methods**, Prentice Hall, 1977.

2. Forsythe, G. E., Pitfalls in Computation, or why a Math. Book isn't enough, **Amer. Math. Monthly**, 77, 1970.

3. Forsythe, G. E., M., A. Malcolm and C., Moler, **Computer Methods for Math**. Computations, Prentice-Hall, 1977.

4. Miller, W., **The Engineering of Numerical Software**, Prentice-Hall, 1984.

5. Stewart, G. W., **Introduction to Matrix Computations,** Academic Press, 1973.