When to Stop Slowly Convergent Iteration?

§0. Abstract

As $k \to \infty$, iteration $x_{k+1} := Ç\cdot x_k + b$ is intended to converge to the fixed-point $z = (I - Ç)^{-1}\cdot b$, but rounding errors interfere by causing computed iterates x_k to dither rather than settle upon a computed limit. Dithering is worst when convergence would be very slow without roundoff.

> How should we decide to stop iterating before we have wasted too much time dithering, and yet not before our last iterate x_{n+1} approximates z as accurately as desired if not almost about as accurately as roundoff allows?

This question becomes most troublesome when the arithmetic's precision does not much exceed the accuracy desired in z, as occurs sometimes when a computation that used to be carried out in 8-byte floating-point arithmetic (53 sig.bits ≈ 16 sig.dec.) is attempted in 4-byte floating-point (24 sig.bits ≈ 7 sig.dec.) to increase speed and reduce energy dissipation.

§1. Introduction

While the *error* $e_k := x_k - z$ is unknown, some other byproduct of the iteration $x_{k+1} := Ç\cdot x_k + b$ must be gauged to determine when to stop. In the absence of anything better, we shall be compelled to stop when the *increment*

$$\Delta x_n := x_{n+1} - x_n = e_{n+1} - e_n \approx Ç\cdot\Delta x_{n-1}$$

is deemed negligible although, if convergence is very slow, the error e_{n+1} may still be far bigger than negligible. Why not deem Δx_n to be negligible if it is smaller or not much bigger than our least over-estimate of roundoff's contribution to it? That over-estimate turns out excessively big when convergence is slow, as we shall see in §2 where dithering's amplitude is over-estimated.

At the end of §3 a likelier estimate of that amplitude comes out of a probabilistic model of roundoff in the iteration. Though still generally unknowable, this liklier estimate suggests a plausible stopping criterion presented in §4 . An undesired by-product of this criterion is the amplification of a computed solution's uncertainty due to roundoff; slower convergence implies worse amplification to an extent intolerable unless arithmetic's precision exceeds adequately the accuracy desired in the computed solution x_{n+1} . An exception to this bleak analysis occupies §5.

§2. Unknowable Pessimistic Upper Bounds

Assume roundoff perturbs the pristine equation " $x_{k+1} = Ç\cdot x_k + b$ " to

$$x_{k+1} := Ç\cdot x_k + b + u_k$$

in which no more is known about roundoff's contribution u_k than an elementwise bound $\hat{u} > |u_k|$. Roundoff accumulates to inflate the error:

$$e_{k+1} = Ç\cdot e_k + u_k \quad \text{for every } k \ge 0$$
$$= Ç^{k+1}\cdot e_0 + \textstyle\sum_{0\le j\le k} Ç^j\cdot u_{k-j} \ .$$

Absent roundoff, e_k would converge to o from every e_0 ; therefore $Ç^k$ converges to O . Let ç barely exceed the biggest magnitude of the eigenvalues of Ç , so $0 \le ç < 1$. Some vector norm ‡...‡ must exist whose induced matrix norm $‡Ç‡ := \max_{v \ne o} ‡Ç·v‡/‡v‡ \le ç$; this unknown ‡Ç‡ exceeds the biggest | eigenvalue of Ç | by as little as one wishes.

Some ü with $|ü| \le û$ maximizes ‡ü‡ , and then $‡u_k‡ < ‡ü‡$ since each $|u_k| < û$. Consequently the formula above for e_{k+1} implies ultimately that

$$‡e_\infty‡ < \sum_{0 \le j \le \infty} ‡Ç‡^j·‡ü‡ \le ‡ü‡/(1 – ç) ;$$

and a conceivable but very unlikely malicious conspiracy among rounding errors could prevent every $‡e_k‡$ from falling much below that bound. It can be huge if convergence is slow because then ç is very nearly 1 ; this must happen when $‡(I – Ç)^{–1}‡$ is huge and may happen otherwise.

Besides pessimistic, that bound upon $‡e_\infty‡$ is useless for a stopping criterion since e_k , ç and ‡...‡ are so unlikely to be known during the iteration process. Otherwise iterating could be stopped as soon as roundoff caused at most a few violations of the expected inequality

$$“ ‡e_{n+1}‡ = ‡Ç·e_n‡ \le ç·‡e_n‡ < ‡e_n‡ ” .$$

Instead of e_k we can know every computed value of ...

$$\Delta x_k = x_{k+1} – x_k = e_{k+1} – e_k = Ç·\Delta x_{k-1} + u_k – u_{k-1}$$

which would obey a similar expected inequality “ $‡\Delta x_{n+1}‡ < ‡\Delta x_n‡$ ” in the absence of roundoff. If a few roundoff-induced departures from such monotonic convergence could be detected, they would supply ample incentive to stop iterating lest time be wasted dithering. But ‡...‡ is rarely known and, when known, the cost of computing $‡\Delta x_k‡$ is rarely affordable.

How small must $‡\Delta x_n‡$ be to violate that ideally expected inequality “ $‡\Delta x_{n+1}‡ < ‡\Delta x_n‡$ ” because of roundoff? That inequality cannot be violated until

$$‡\Delta x_n‡ \le ‡u_{n+1} – u_n‡/(1 – ç) \le 2‡ü‡/(1 – ç) .$$

This inequality is consistent with another obtained from the deduced formula

$$\Delta x_k = Ç^k·\Delta x_0 – Ç^{k–1}·u_0 + (Ç – I)·\sum_{1 \le j<k} Ç^{j–1}·u_{k-j} + u_k$$

which can be bounded, again too pessimistically, as before:

$$‡\Delta x_\infty‡ < ‡Ç – I‡·\sum_{1 \le j<\infty} ‡Ç‡^{j–1}·‡ü‡ + ‡ü‡ \le 2‡ü‡/(1 – ç) .$$

As before, this bound also is useless for a stopping criterion. But not entirely useless ...

Huge error-bounds $‡e_\infty‡ < ‡ü‡/(1 – ç)$ and $‡\Delta x_\infty‡ < 2‡ü‡/(1 – ç)$ above come from very small divisors (1 – ç) just when convergence is very slow, and then those huge bounds threaten the accuracy achievable by iteration. Inaccuracy is threatened by slow convergence regardless of how *Well-* or *Ill-Conditioned* the given equation “ $z = Ç·z + b$ ” may be. An example $Ç := –g·g'$ for any column g with $g'·g = 0.9999$ has a condition number $||I – Ç||·||(I – Ç)^{–1}|| < 2$ but $1/(1 – ç) = 10^4$, so iteration could conceivably lose 4 sig. dec. just because convergence is slow.

However, the threat of inaccuracy is mitigated by known gross pessimism in those error-bounds.

§3. Probabilistic Over-Estimates

When convergence is slow, pessimism is due mostly to the tiny divisor $(1 - \text{ç})$ in §2's bounds. To obtain a smaller estimate for the likeliest accumulation in Δx_k of all the contributions $u_{k\text{-}j}$ from roundoff, rounding errors shall be approximated by independent random variates that range between predictable bounds. We expect to obtain a bigger divisor.

First let us summarize probabilistic terminology. Let t be a *Random Variate* (not necessarily scalar) whose *Cumulative Distribution* $\mu(t)$ figures in the *Expected Value* $\text{Æ}f(t)$ of any function $f(t)$ thus:

$$\text{Æ}f(t) := \int_t f(t)\cdot d\mu(t) \quad \text{wherein} \quad d\mu(t) \ge 0 \quad \text{and} \quad \text{Æ}1 = 1\,.$$

(Our notation for the expected value, also called "*Mean Value*", must be confusing because $\text{Æ}f(t)$ depends not upon the value of t but upon its distribution μ. This notation is widespread among statisticians despite its confusion because it costs fewer names, — merely one name " t " per random variate instead of a pair "$\{t, \mu\}$" .)

The expected value of a sum is the sum of expected values; $\text{Æ}(f(t_1) + g(t_2)) = \text{Æ}f(t_1) + \text{Æ}g(t_2)$ regardless of whether t_1 and t_2 are the same or different random variates, correlated or not. If s and t are *Independent* random variates then the expected value of a product is the product of expected values: $\text{Æ}(f(t)\cdot g(s)) = \text{Æ}f(t)\cdot\text{Æ}g(s)$.

The *Variance* $\sigma^2 f(t) := \text{Æ}(\|f(t) - \text{Æ}f(t)\|^2)$ is the square of a gauge, the *Standard Deviation* $\sigma f(t) = \sqrt{(\sigma^2 f(t))}$, of the likely departure of $f(t)$ from its mean value. Here $\|v\| := \sqrt{(v'\cdot v)}$ is the *Euclidean* norm. $\text{Æ}((f(t) - \text{Æ}f(t))\cdot(f(t) - \text{Æ}f(t))')$ is the *Covariance Matrix* of a column $f(t)$. If its elements are independent the matrix is diagonal. The variance of the sum of independent variates is their variances' sum: $\sigma^2(f(t) + g(s)) = \sigma^2 f(t) + \sigma^2 g(s)$. Finally, if a scalar variate $f(t)$ has mean $\text{Æ}f(t) = 0$ then $\text{Æ}|f(t)| \le \sigma f(t)$; the magnitude's expected value cannot exceed the standard deviation. This will help attenuate the pessimism of §2's over-estimate of roundoff's accumulation in Δx_∞ .

IEEE Standard 754 arithmetic's default rounding is unbiased; this means that each rounding error can be approximated by an independent bounded random variate with mean zero. Such a variate's standard deviation must be smaller than its bound, usually a small fraction of it. Each element of $(\text{Ç} - I)\cdot\text{Ç}^{j-1}\cdot u_{k\text{-}j}$ is such a variate; its standard deviation is some unknown fraction of unknowable bound $(\text{ç}+1)\cdot\text{ç}^{j-1}\cdot\text{‡ü‡}$. Therefore the standard deviation of each element of Δx_∞ turns out to be some unknown but probably small fraction of

$$\text{‡ü‡}\cdot\sqrt{\left((\text{ç}+1)^2\cdot\textstyle\sum_{1\le j<\infty} \text{ç}^{2j-2} + 1\right)} = \text{‡ü‡}\cdot\sqrt{\left(2/(1 - \text{ç})\right)}\,,$$

which is rather smaller than §2's bound $2\text{‡ü‡}/(1 - \text{ç})$ though still unknowable.

§4. Estimating the Unknowable

Estimates of ‡ü‡ and ç will be needed if iterating is to be stopped after every element of Δx_n is not much bigger than ‡ü‡$\cdot\sqrt{(2/(1-\text{ç}))}$, which is our estimate in §3 of dithering's expected amplitude. To guess at the unknowable seems better than to stay paralyzed by it.

In the absence of roundoff, for almost every Δx_0,

$$\text{ç} \approx \mathrm{Lim}_{k\to\infty}(\|\Delta x_k\|/\|\Delta x_0\|)^{1/k}$$

regardless of the choice of norm ||...|| . This suggests approximating ç by $(\|\Delta x_k\|/\|\Delta x_0\|)^{1/k}$ for some big iteration-counts k not yet so big that Δx_k is contaminated too badly by roundoff. Such approximations more often under-estimate ç than over-estimate it until after k gets big enough to incur roundoff-induced dithering.

Unknowable ‡ü‡ came in §2 from û , the result of a rounding-error-analysis of the programmed formula " $x_{k+1} := \text{Ç}\cdot x_k + b$ " that produced $\hat{u} > |\text{Ç}\cdot x_k + b - x_{k+1}|$ elementwise for §2 . Since ‡...‡ is usually too big as well as unknown, little is lost by the adoption of $\|...\|_\infty$, the biggest-element vector norm, to measure the magnitudes of ü , û and Δx_k. As norms go, $\|...\|_\infty$ is among the smaller ones, especially when dimensions are large. By its adoption we choose to approximate ç by

$$\text{ç} \approx (\|\Delta x_k\|_\infty/\|\Delta x_0\|_\infty)^{1/k}$$

for sufficiently (but not too) big counts k . Now let us choose to stop iterating as soon as, say,

$$\|\Delta x_n\|_\infty \le 3\cdot\|\hat{u}\|_\infty\cdot\sqrt{(2/(1-\text{ç}))} \qquad \textbf{\textit{STOPPING CRITERION}}$$

for at most a few consecutive increments Δx_n, if not sooner. The " 3 " is another guess.

What can go wrong? If that stopping threshold is too big, iterating will stop too soon, before x_{n+1} has come as close to the desired z as it would come after more iterations. More likely is a threshold too small; then iterations will dither while recomputed estimates of ç increase, thus increasing the threshold until it stops the iteration.

Something else can go wrong. Since $e_k = (\text{Ç} - I)^{-1}\cdot(u_k + \Delta x_k)$,

$$\|e_n\|_\infty \le \|(\text{Ç} - I)^{-1}\|_\infty\cdot\|\hat{u}\|_\infty\cdot(1 + 3\cdot\sqrt{(2/(1-\text{ç}))})$$

when iterating is stopped. The factor $(1 + 3\cdot\sqrt{(2/(1-\text{ç}))})$ amplifies roundoff's unavoidable contribution to the uncertainty $\|(\text{Ç} - I)^{-1}\|_\infty\cdot\|\hat{u}\|_\infty$ in every computed solution $x = \text{Ç}\cdot x + b + u$ wherein roundoff injects a term u bounded by $|u| < \hat{u}$. That amplification is due solely to the choice of a slowly convergent iteration to solve the given fixed-point problem, which may be well-conditioned in so far as $\|(\text{Ç} - I)^{-1}\|_\infty$ is not very big. Otherwise, when $\|(\text{Ç} - I)^{-1}\|_\infty$ is big, that amplification may subtract extra accuracy intolerably from the arithmetic's precision,

When iteration is performed in arithmetic carrying about 16 sig.dec., a loss to slow convergence of a few digits beyond the several digits lost to ill-condition goes unnoticed. But when arithmetic carries fewer than 8 sig.dec. the loss of a few extra digits comes as an unwelcome surprise.

How often does that extra loss occur? It is not often reported, if at all, but it can occur; here is a didactic example:

Example:
All the computations on this page were performed by PC-MATLAB 3.5 on a 386/387-based Intel 302 taking advantage of the 387's capability to control its arithmetic precision.

All input data Ç and b were stored as 4-byte wide floating-point variables with 24 sig.bits, worth about 7 - 8 sig.dec. Arithmetic of this same precision was used to compute all the iterates $x_{k+1} := Ç \cdot x_k + b$. The desired fixed-point $z := (I - Ç)^{-1} \cdot b$ and $-ç$, the biggest in magnitude among the eigenvalues of Ç , were computed more accurately in arithmetic with at least 53 sig.bits, worth at least 15 sig.dec.

	814992	555704	5341046	895145	354535	
	2494324	2994674	5538020	1198341	5070571	
Ç := –	124469	5863168	5195799	4248937	4634218	$/2^{24}$
	3068273	3969477	2396288	4876337	2977128	
	2752950	3447374	1868283	2882239	5928951	

	-0.048577309	-0.033122540	-0.31835115	-0.053354800	-0.021131933
	-0.14867330	-0.17849648	-0.33009171	-0.071426690	-0.30222958
≈	-0.0074189305	-0.34947205	-0.30969375	-0.25325638	-0.27622092
	-0.18288332	-0.23659927	-0.14282990	-0.29065233	-0.17745066
	-0.16408861	-0.20547950	-0.11135834	-0.17179483	-0.35339302

and $-ç \approx -0.9998912395141 > -1$ is its eigenvalue of biggest magnitude, though every familiar $||Ç|| > 1$. Consequently the iteration converges to z extremely slowly in the absence of roundoff. $||(I - Ç)^{-1}|| < 1.43$, so the equation " $z = Ç \cdot z + b$ " is quite well-conditioned and determines z within a few units in the arithmetic's last digit. Tabulated here are b , z and the dithering iterates x_n and $x_{n\pm1}$ obtained several iterations after starting from $x_0 := b/2$:

b' =	8195868	15879334	19379998	14924160	14431199
x_n' =	4098***027.75***	7939***849.0***	9690***221.***	7462***251.0***	7215***764.5***
z' ≈	4098156.9237	7940098.5675	9690526.1056	7462485.7067	7215992.0155
$x_{n\pm1}$' =	4098***286.0***	7940***348.5***	9690***831.***	7462***720.5***	7216***219.0***

The amplitude $||\Delta x_\infty||_\infty = 610$ of dithering is moderately bigger than the product of the expected amplification factor $\sqrt{(2/(1 - ç))} \approx 135.6$ and a roundoff bound $||\hat{u}||_\infty \approx 2$. Because dithering began after several iterations, ç was soon over-estimated and then the ***STOPPING CRITERION*** above stopped further iterating and left the last few sig.dec. of x_n wrong, as predicted above, though nothing about the rounding errors was *random*. "Accidental" describes them better.

x_n is wrong solely because slow convergence exacerbated roundoff's contribution.
It could have been worse.

But roundoff's contribution from 8-byte floating-point carrying 53 sig.bits would go unnoticed.

§5. The Exceptional SOR Iteration

The iteration to be discussed next continues to converge despite dithering until its amplitude dies down to the last few bits of the iterates x_k no matter how slowly they converge. The criterion for stopping this iteration can ignore the rate of convergence; no accuracy need be lost by stopping before Δx_n is deemed negligible, though ill-condition may allow the final error $e_n = x_n - z$ to be vastly bigger than negligible. However, this SOR iteration applies only to a special case:

We seek the solution z of $A \cdot z = b$ given b and a symmetric positive definite matrix $A = A'$. Its diagonal elements must be positive and somewhat bigger than its off-diagonal elements in so far as every $a_{ij}^2 < a_{ii} \cdot a_{jj}$ if $i \neq j$. No generality is lost but simplicity is gained by assuming every $a_{jj} = 1$; this can be arranged in either of two ways. The first way divides each row of $[A, b]$ by its diagonal element, thus simplifying the iteration's arithmetic. The second way divides each row by the square root of its diagonal element and does likewise to each column of A, thus imposing upon z a change that must be undone after z has been computed by iteration. Neither way alters the fact nor the speed of convergence, but the second way simplifies the exposition, so it shall be adopted. Then $A = I - L - L'$ for some strictly lower-triangular matrix L, and the SOR iteration's formula becomes

$$x_{k+1} := x_k + \Omega \cdot (b + L \cdot x_{k+1} - x_k + L' \cdot x_k) = (I - \Omega \cdot L)^{-1} \cdot \big(\Omega \cdot b + (I - \Omega + \Omega \cdot L') \cdot x_k \big)$$

in which Ω is a diagonal matrix whose diagonal elements, all strictly between 0 and 2, are chosen to boost the rate of convergence. The formula seems at first to determine x_{k+1} implicitly but actually determines its elements explicitly and strictly sequentially from top to bottom.

"SOR" stands for "Successive Over-Relaxation". It could also be called "Extrapolated Gauss-Seidel Iteration". The unextrapolated version with $\Omega = I$ was invented by Gauss for a geodetic survey and later deprecated by Seidel, an astronomer, in the 19th century. In the early 20th Southwell revived and generalized the iteration for loaded elastic structures, whence the name "Relaxation". Similar iterations were applied to passive electric circuits. In the 1940s and early 1950s SOR was the easiest way to solve big discretized elliptic boundary-value problems on computers with memories infinitesimal by today's standards. Over-Relaxation occurred for diagonal elements of Ω strictly between 1 and 2; Under-Relaxation … 0 and 1. For more details see chapters 3 and 4 of the book by Varga [1962].

SOR seems at first too sequential to exploit concurrency on computers nowadays. However, if the bandwidth of A is small compared with its dimension, a peristaltic (often wrongly called "systolic") process can begin the computation of x_{k+2} well before x_{k+1} is finished, and x_{k+3} before x_{k+2} is finished, and so on. Their computations' concurrency requires synchronizations that may be complicated to program. To reduce the cost of communications when x_k must be spread out among the distributed memories of many processors, SOR may be supplanted by a "Chaotic Iteration" that will not be considered here; it requires constraints like $\| |L + L'| \| < 1$ among others according to D. Chazan and W.L. Miranker [1969], who disregarded roundoff.

SOR deserves to be considered here because, as iterations go, it is comparatively indifferent to roundoff. To prove this, we shall replace Ω by Ω_k and vary it from iteration to iteration. Doing so permits the subscript k to be dropped, further simplifying a complicated convergence proof.

SOR converges because each iteration reduces the error $e := x - z$ when measured in a norm $\text{«}e\text{»} := \sqrt{(e'\cdot A\cdot e)}$ that Southwell related to elastic energy of deformation. This «…» is a norm for vectors since A is positive definite; but «…» generally differs from the unknowable norm ‡…‡ used above. Here follows a proof that, absent roundoff, $\text{«}e+\Delta x\text{»} < \text{«}e\text{»}$ while $e \neq o$:

Each SOR iteration replaces x by $x + \Delta x := x + \Omega\cdot\big(b + L\cdot(x+\Delta x) - x + L'\cdot x\big)$, changing its error e to $e + \Delta x = e + \Omega\cdot(L\cdot(e+\Delta x) - e + L'\cdot e) = e + \Omega\cdot L\cdot\Delta x - \Omega\cdot A\cdot e$. Increment Δx appears to come from something resembling x's *Residual* $r := b - A\cdot x = -A\cdot e$ but different; $\Delta x = \Omega\cdot c$ where the elementwise *Current Residual* is

$$c := b + L\cdot(x+\Delta x) - x + L'\cdot x \;=\; L\cdot\Delta x - A\cdot e \;=\; (I - L')\cdot\Delta x - A\cdot(e+\Delta x)\ .$$

Now we find $c = -(I - L\cdot\Omega)^{-1}\cdot A\cdot e \neq o$ while $e \neq o$, and then

$$\begin{aligned}\text{«}e\text{»}^2 - \text{«}e+\Delta x\text{»}^2 &= (e - e - \Delta x)'\cdot A\cdot(e + e + \Delta x)\\ &= \Delta x'\cdot\big(\,c - L\cdot\Delta x + c - (I - L')\cdot\Delta x\,\big) \quad \text{from the previous equations}\\ &= c'\cdot\Omega\cdot(2I - \Omega)\cdot c \quad \text{because } \Delta x'\cdot(L' - L)\cdot\Delta x = 0\\ &> 0 \quad \text{since diagonal } \Omega \text{ lies strictly between } 0 \text{ and } 2\,. \qquad [\,]\end{aligned}$$

Now restore subscript k to iterate x_k, increment $\Delta x_k = x_{k+1} - x_k = e_{k+1} - e_k$, current residual c_k and diagonal Ω_k to infer that $c_k = -(I - L\cdot\Omega_k)^{-1}\cdot A\cdot e_k \neq o$ while $e_k \neq o$, and that then

$$\text{«}e_k\text{»}^2 - \text{«}e_{k+1}\text{»}^2 = c_k'\cdot(2I - \Omega_k)\cdot\Omega_k\cdot c_k > 0\ .$$

This inequality is crucial. It is valid while $\Delta x_k = \Omega_k\cdot c_k \neq o$ and every element of diagonal Ω_k is nonnegative and strictly less than 2, even if some diagonal elements and their elements of Δx_k vanish. It forces the values $\text{«}e_k\text{»}$ to form a strictly descending sequence and therefore converge.

Must $\text{«}e_k\text{»} \to 0$?

Not necessarily. Not without some further constraint upon Ω_k .

The simplest further constraint is constancy. If the one diagonal $\Omega_k := \Omega$ has diagonal elements strictly between 0 and 2, then the convergence of $\text{«}e_k\text{»}$ forces $c_k'\cdot(2I - \Omega)\cdot\Omega\cdot c_k \to 0$, whence follows $c_k \to o$ and then $e_k = -A^{-1}\cdot(I - L\cdot\Omega)\cdot c_k \to o$ too as desired. Moreover the sequence of values $\text{«}\Delta x_k\text{»}$ can be shown to descend to 0 monotonically too. But roundoff spoils all that.

Because roundoff can prevent the convergence of e_k to o, some other goal for SOR must be accepted. Our chosen goal is to continue iterating no longer than while a diagonal Ω_k exists in $O \le \Omega_k < 2I$ such that the computed nonzero $\Delta x_k := x_{k+1} - x_k$ and the true but not quite known current residual c_k satisfy $\Delta x_k = \Omega_k\cdot c_k$ exactly, thus ensuring that $\text{«}e_{k+1}\text{»} = \text{«}e_k+\Delta x_k\text{»} < \text{«}e_k\text{»}$.

Drop subscript k again. Let u bound roundoff elementwise in the computed value c so that the true current residual surely lies within $c \pm u$; for instance, while $|\Delta x| \le |x|$ we may set, say, $u := (\,|b| + 2\cdot|A|\cdot|x|)\cdot\varepsilon$ where ε is the arithmetic's roundoff threshold. Let v bound roundoff in the computed Δx so that the computed new $x + \Delta x$ will surely lie within $x + \Omega\cdot c \pm v$; for instance, we may set $v := (\,|x| + 2\Omega\cdot|c|\,)\cdot\varepsilon$. The columns u and v of overestimates can be computed simultaneously with c and Δx, or else just once after x stops changing much.

Here are two ways to decide when to stop SOR iterating:

As the elements of the new $x + \Delta x$ are being computed in turn from top to bottom, leave a new element unchanged (thus treating the corresponding element of Δx as 0) if the corresponding element of $|c| \le u$. Otherwise Δx will have the correct sign but, to keep Δx from getting too big, reduce the corresponding diagonal element of Ω if necessary to keep it between 0 and $\max\{0,\ 2 - (2u+v)./|c|\}$. If $2I - \Omega$ is not too small, a simpler alternative leaves unaltered any element of x corresponding to an element of $(2I - \Omega)\cdot|c| \le 2u + v$. Either way, whenever a computed (rounded) element of the new $x + \Delta x$ matches the old x , regard that element of Δx as 0 ; otherwise the new $x + \Delta x$ is better than the old x because $«e + \Delta x» < «e»$.

STOP ITERATING as soon as all of $\Delta x = o$, if not sooner.

This stopping criterion takes no notice of the rate of convergence, which can be arbitrarily slow even if Ω was chosen to maximize it. Convergence must be slow if A is nearly singular, and may be slow otherwise too, but continues without dithering until the residual $r := b - A\cdot x$ is not much bigger than its uncertainty $\pm u$ due to roundoff in its own computation. After that the error $e = x - z = -A^{-1}\cdot r$ cannot much exceed the smallest uncertainty that the condition of A allows.

Choosing Ω optimally is a hard problem solved in the 1950s only partially for some special but common cases. Under-Relaxation ($\Omega \le I$) has accelerated convergence for a few matrices A , but Over-Relaxation ($\Omega \ge I$) has been found better for most others. In most of these other cases $2I - \Omega \approx 1 - ç$ is too small for the simpler alternative above; stopping when $(2I - \Omega)\cdot|c| \le 2u + v$ then stops too soon, sooner than signalled by §4's probabilistic stopping criterion.

§6. Conclusion

If any further incentive were needed to avoid slowly convergent iterations, it would be supplied by questions about stopping. They didn't matter while arithmetic's precision amply exceeded the desired accuracy, but become troublesome when precision barely exceeds the accuracy desired.

§7. Citations

D. Chazan & W.L. Miranker [1969] "Chaotic Relaxation" in pp. 199-222 of *Linear Algebra & its Applications* **2** (American Elsevier)

Richard S. Varga [1962] *Matrix Iterative Analysis* (Prentice-Hall, New Jersey)