

Poetry & Algorithms

Michael Bartholomew-Biggs
Reader Emeritus in Computational Mathematics,
University of Hertfordshire, Hatfield, AL10 9AB, England.
matqmb@herts.ac.uk

Abstract,

This paper considers a supporting role for poetry in the statement of numerical algorithms and termination conditions. While this has certainly been touched on elsewhere, the present work takes nonlinear optimization as a case-study and shows what can be done with quasi-haiku formulations of familiar computational methods and convergence criteria.

Introduction

I am indebted to Dr Clemency Montelle of the University of Canterbury, New Zealand for pointing out that verse forms were often employed in the propositions and expositions of early Indian mathematics [1]. In such oral traditions, concise and memorable exposition of ideas was highly important [2]. On the other hand, since constraints of verse-forms might obscure some technical detail, it was usual for mathematical poetry to be accompanied by written commentary. In contemporary computational mathematics, the typical unit of concise exposition is the algorithm. It too is usually accompanied by a commentary which may include an intuitive justification for the proposed technique together with supporting convergence theorems. Is it possible that poetry could still have a role in enhancing bald algorithmic statements?

It is of course easy to debunk the notion of poetry as algorithm, perhaps by quoting the delightful imprecision of Carl Sandburg's 'Arithmetic' [3] which includes the stanza

*If you take a number and double it and double it again then double
it a few more times the number gets bigger and bigger and goes
higher and higher and only arithmetic can tell you what the number
is when you decide to quit doubling.*

or an anonymous limerick [4] that begins *There was an old man who said "Do / Tell me how I should add two and two?"* Nevertheless I intend to explore relationships between poetry and algorithm with particular reference to nonlinear optimization, which is a mathematical discipline concerned with how to "do one's best". It is assumed, of course, that *best* is measurable, as is it is when we want to do a task in minimum time or for minimum cost.

Optimization.

It has been claimed that *everything* is an optimization problem. Many systems in nature have evolved (or been created) to function with minimum expenditure of energy. Human beings often pursue a similar goal – although most of us operate by guesswork rather than by mathematics. And in so doing we will recognize that we are sometimes hindered by *constraints* imposed by nature and by other people in the form of physical laws, safety regulations or sheer human frailty.

Optimization comes into its own in science and engineering where well-founded mathematical models enable us to predict quite accurately what will happen if some variable is adjusted. Optimization can be less effective – to put it mildly – when human behaviour is involved. "Optimal" financial decisions can let us down badly when models of risk prove much less trustworthy than Newton's laws for describing flight dynamics. (See for instance [5],[6] for a general survey of optimization and various applications.)

Conventionally we pose optimization problems as *minimizations* – equivalent to seeking the lowest point on a hypersurface defined by some *performance function*. Iterative optimization algorithms approach a solution via a sequence of steadily improving approximations; and, if there are no constraints, then progress can be viewed as a systematic exploration of a mathematical landscape like the contour map in Figure 1 – where X marks the lowest point. This – already rather poetic – metaphor is now so embedded in the consciousness of optimization practitioners that it is hardly even thought of as imagery.

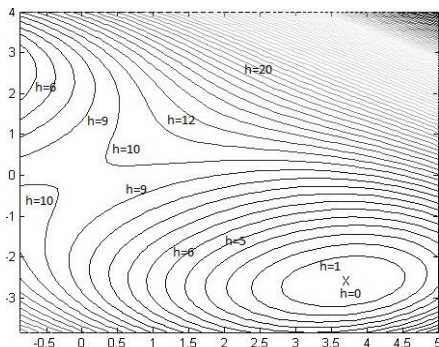


Figure 1

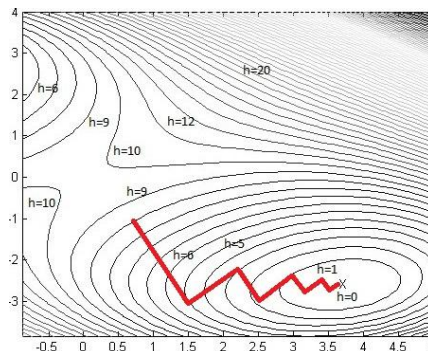


Figure 2

In reality we need to stretch our minds around a generalisation of Figure 1 into more than two dimensions. We let \mathbf{x} denote a point in such a space and $h(\mathbf{x})$ is then the corresponding terrain surface height. We want to find the \mathbf{x} that gives the lowest $h(\mathbf{x})$. Our main difficulty is that only *local* information is available – we do not have an area map. It is as if we are walking on a hillside in thick mist. All we can see is the ground at our feet. How will we find the bottom of the valley?

An elementary tactic involves detecting the line of steepest slope and walking steadily in this direction until the ground flattens out. It is then time for a re-orientation to find and follow a new steepest downhill direction. This is essentially the method of *steepest descent*. If $\mathbf{g}(\mathbf{x})$ denotes the magnitude and direction of the surface gradient we may write the iterative scheme in mathematical notation as

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha \mathbf{g}(\mathbf{x}_k) \quad \text{where } \alpha \text{ is chosen so that } \mathbf{g}(\mathbf{x}_{k+1})^T \mathbf{g}(\mathbf{x}_k) = 0.$$

The orthogonality condition is explained in [5]. We might convey the same idea more vividly as

Steepest Descent

<i>Still the fog persists.</i>	<i>Keep taking footsteps</i>	<i>then turn left or right,</i>
<i>Let the incline have its way</i>	<i>until that first suspicion</i>	<i>just one of many zig-zags.</i>
<i>and set your compass.</i>	<i>of an uphill slope</i>	<i>Will it ever end?</i>

Note that the poetic statement is flexible and frank enough to acknowledge that the method often takes a zig-zag path (see Figure 2) that can be painfully slow to converge! To descend more rapidly on a curved path, we must move continuously at right angles to the contour lines and this involves solving a system of differential equations

$$d\mathbf{x} / dt = - \mathbf{g}(\mathbf{x})$$

with a high-order method or a very small step size. Both are computationally costly; and a compromise strategy is to restrict our radius of action and seek the direction that seems to promise the biggest descent. For this, we need information about the hillside’s curvature which is contained in $\mathbf{G}(\mathbf{x})$, the Hessian matrix of second partial derivatives of $h(\mathbf{x})$. We then seek $\delta \mathbf{x}_k$ to minimize $\delta \mathbf{x}_k^T \mathbf{g}(\mathbf{x}_k) + \delta \mathbf{x}_k^T \mathbf{G}(\mathbf{x}_k) \delta \mathbf{x}_k$ subject to an upper limit on $\|\delta \mathbf{x}_k\|_2$. This can be done by choosing a suitable δt and solving

$$(I + \delta t \mathbf{G}(\mathbf{x}_k)) \delta \mathbf{x}_k = - \delta t \mathbf{g}(\mathbf{x}_k).$$

(While using both Hessian matrix and gradient vector we may murmur lines from Stanislaus Lem’s ‘The Cyberiad’ [7]: *And every vector dreams of matrices./ Hark to the gentle gradient of the breeze.*)

A technique which searches within a steadily-expanding radius of exploration, as illustrated in Figures 3 and 4, is usually known as a “trust-region” method [8]. By invoking this non-numeric idea of “trust,” optimization practitioners have allowed another element of metaphor to creep into their discourse. (And this is not the only metaphor to be found in the literature. An algorithm devised by M.J.D. Powell [8] is called the “dog-leg” method because of its resemblance to a golfing strategy.)

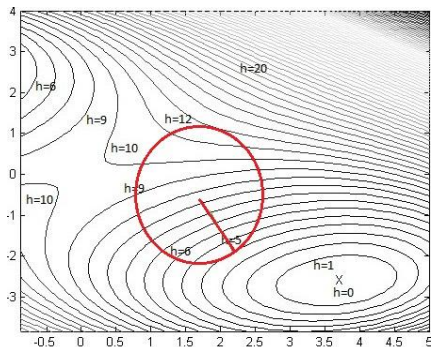


Figure 3

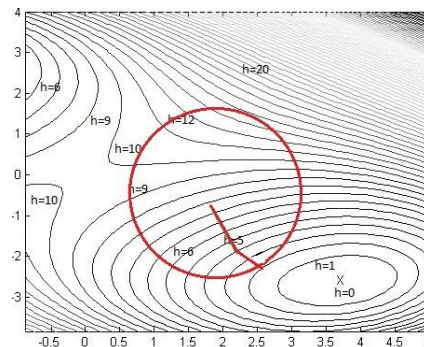


Figure 4

As noted already, trust region methods depend on curvature. This is much harder to deduce than slope which can be estimated by surface inspection. Curvature resembles an emotion hidden behind the face of someone with whom we are starting a romance. In a quest for more intimacy, we must allow for rapid changes in the loved one’s moods and advance cautiously so as not to lose ground already gained.

Trust Region

How far dare I go?

*I’ve a hunch what to expect
but I might be wrong.*

I want to confess

*where my fancy’s heading but
I must go gently*

for to cause alarm

*would risk ending the affair
before it’s started.*

The trust region method’s reliance on choosing apt values for the radius δt also echoes certain links between numerical scaling and emotion made by Auden in the poem ‘Numbers and Faces’ [9]:

*The Kingdom of Number is all boundaries
Which may be beautiful and must be true;
To ask if it is big or small proclaims one
The sort of lover who should stick to faces.*

Constrained optimization

We now turn to methods for handling constraints in an optimization problem. When such limitations exist, it is as if a capricious hyper-landowner has erected fences on our hyper-landscape to thwart our instincts about where to go, as in Figure 5.

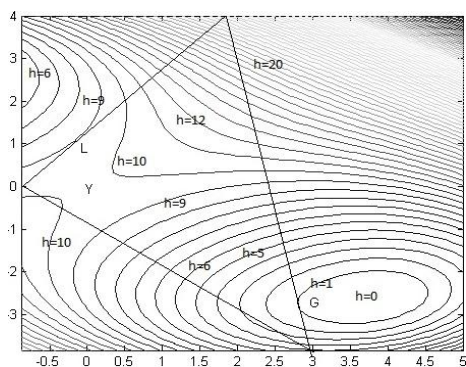


Figure 5

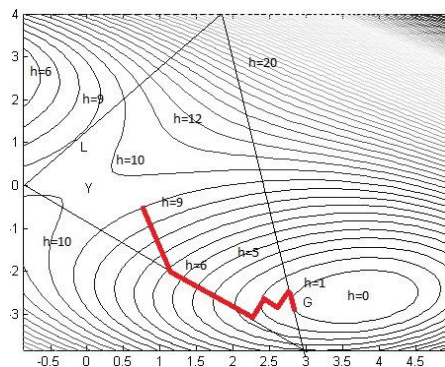


Figure 6

If, for instance, we can only explore the interior of the triangle in Figure 5 then we can no longer reach the valley floor. The best we can do is to peer over the fence at point G.

When we are confined within boundaries (not necessarily straight lines) defined by equations of the form $c_i(\mathbf{x}) = 0$ then the optimization problem can be written

$$\text{Minimize } h(\mathbf{x}) \text{ subject to } c_i(\mathbf{x}) \geq 0, \quad i=1, \dots, m.$$

Any downhill search direction determined by the slope of $h(\mathbf{x})$ must be prevented from leaving our so-called *feasible region*. If we regard each constraint as a barrier then a search direction must either *bounce back* from it or else *slide along* its length. The sideways sliding is called *projection*; and if \mathbf{a}_i is the normal vector to the constraint $c_i(\mathbf{x})$ then the projection of the gradient $\mathbf{g}(\mathbf{x})$ is $[I - \mathbf{a}_i(\mathbf{a}_i^T \mathbf{a}_i)^{-1} \mathbf{a}_i^T] \mathbf{g}$. A typical search using projected gradients is shown in Figure 6 and can be described poetically by:

Barrier

<i>You hit it, running.</i>	<i>till the hillside tilt</i>	<i>lurking in a dip,</i>
<i>The dry-stone wall deflects you</i>	<i>releases you – or else a</i>	<i>destroys more hope of getting</i>
<i>down its stubborn length</i>	<i>second obstacle,</i>	<i>where you want to be.</i>

Optimality conditions & termination

Rules about where to stop are just as important as rules about how to search. After roaming on a virtual hillside we need to know when we have reached our goal. The conditions at a constrained minimum mean that there must exist *Lagrange multipliers* λ_i , $i = 1, \dots, m$ such that the function

$$L(\mathbf{x}, \boldsymbol{\lambda}) = h(\mathbf{x}) - \boldsymbol{\lambda}^T \mathbf{c}(\mathbf{x})$$

has a *stationary point*. This implies a kind of Mexican stand-off between function and constraints. There exists no move which both reduces $h(\mathbf{x})$ and satisfies the constraints. Any downhill move also breaks a constraint; any move that stays within the constraints is an uphill one. Poetry can put this another way

Optimality

*Keep the rules and lose:
or win because you break them.
Checkmate or stalemate?*

The Lagrange multipliers – sometimes known as *shadow prices* or *dual variables* – are not simply mathematical conveniences. They tell how a solution might change if the constraints were relaxed. Specifically, if the i -th constraint boundary were “pushed outwards” to become $c_i(\mathbf{x}) \geq -\varepsilon$ then we can predict that the optimal value of h will decrease by approximately $\lambda_i \varepsilon$.

Lagrange multipliers

<i>These are our shadows,</i>	<i>the consequence of</i>
<i>dual personalities,</i>	<i>pushing our boundaries or</i>
<i>who know what we don't –</i>	<i>shrinking horizons.</i>

We can also introduce extra quantities called *slack variables*, s_i , which measure how far a solution is from the edges of the feasible region. We can then formulate a constrained optimization problem as

$$\text{Minimize } h(\mathbf{x}) \text{ subject to } c_i(\mathbf{x}) - s_i = 0, \quad s_i \geq 0, \quad i=1, \dots, m.$$

As a poetic definition of slack variables we can draw on a Biblical reference [10] and propose

Slack variables

<i>Like those labourers</i>	<i>They are positive:</i>
<i>outside the vineyard, they need</i>	<i>whatever the vacancy</i>
<i>opportunities.</i>	<i>they're equal to it.</i>

There is an intriguing symbiotic relationship between slack variables and Lagrange multipliers. For each constraint, either the corresponding slack variable must be zero or else the corresponding Lagrange multiplier must be zero. Mathematically we write this as $\lambda_i s_i = 0$; more poetically

Complementarity

*Face it: one must go.
This town just ain't big enough
for the both of us.*

Spurious convergence

Regrettably, the convergence tests above may not tell the whole story because they are only *first order* conditions. We neglect second derivatives at our peril for, without them, we cannot detect those false friends called *saddle points*. These can seem optimal when looked at in one way, yet they are unmasked as deceivers and tricksters when seen from another angle. One is lurking around point Y in Figure 5: the values on the contour lines show that, when approached from the north-east, Y looks like a low point of the terrain – but if we move away from Y to the north west or south-east the ground falls away again. We could express this by saying the Hessian matrix \mathbf{G} is non positive definite at Y – i.e. $\mathbf{z}^T \mathbf{G} \mathbf{z}$ is not positive for all \mathbf{z} . But perhaps the following is more compelling?

Saddle point

*You thought you'd arrived
till the valley arched its back
like a startled cat.*

There are other unsatisfactory places for an optimization algorithm to terminate. The point L in Figure 5 has all the characteristics of an optimum – i.e., no *locally* feasible move away from L can reduce the function. And yet, because it has a higher value of $h(\mathbf{x})$, it is inferior to point G which is the *global minimum*. The saddle point at Y is what separates the best point G from the merely rather good one at L. But we cannot tell, from information available at L, that the better point G even exists!

Global solution

*It's the only place
to be – if you can find it.
(If not, you won't know.)*

False convergence is a stressful possibility for numerical analysts and it has caused several of them to resort to poetry as a way of channelling the associated emotion. In their optimization text-book, Gill & Murray [11] offer an incomplete – and slightly flawed – Lennon-McCartney haiku [12] (rather like a Sappho fragment) as a poignant epigraph to their chapter on error detection: *You can get it wrong / and still you think that it's all right*. Strangely enough, however, they fail to quote the readily available, reassuring and syllabically correct concluding line *We can work it out*.

In their more specialised text, Broyden & Vespucci [13] borrow imaginatively from Eliot's 'The Hollow Men' [14] and distinguish iterations which are converging rapidly from those that are stagnating by assigning subscripts b ('bang') and w ('whimper').

Poets who are not mathematicians seem instinctively to understand the anguish caused by spurious solutions as well as the pleasure given by correct ones. Here is Carl Sandburg again from 'Arithmetic' [3]

*Arithmetic is where the answer is right and everything is nice and
you can look out of the window and see blue sky – or the answer is
wrong and you have to start all over and try again and see how it
comes out this time.*

Concluding reflections

Writers of optimization algorithms aspire not only to *finding* solutions but to doing so *quickly and efficiently*. Hence they become quite competitive and are often “done-to as they do,” being under continual scrutiny regarding whose optimization algorithm is currently the best. Sections of the literature resemble arenas for numerical duels to determine whose algorithm can solve the hardest problems fastest. Rivalries emerge as intense as that between Newton and Leibniz, commemorated in the Sarah Glaz poem ‘Calculus’ [15], over whether a derivative should be *evaluated by Newton's fluxions method, $\Delta y/\Delta x$; / or by a formal quotient of differentials dy/dx*

Poets, of course, are subject to similar evaluation, both formally through major international competitions, and informally in the gossip about who’s ‘in’ and who’s ‘out.’ But while rivalries over algorithm performance and reputation may be as keen as any which exist between parallel contenders for the T.S. Eliot prize, optimizers and poets may also have something in common within their nobler natures – namely a wish to express their deep belief that the world can be improved. If this is so then they will also share a wish for the world to take some notice of them! For poets at least, this wish is commonly ungranted; hence poets may remain unacknowledged optimizers of the world; and – unless algorithmic verse becomes more popular – optimizers will be among the world’s unacknowledged poets.

References

- [1] C. Montelle, *Chasing Shadows: Mathematics, Astronomy, and the Early History of Eclipse Reckoning*, Johns Hopkins Studies in the History of Mathematics, 2011
- [2] The British Library, *Mughal India: Art, Culture and Empire*, January 2013
- [3] C. Sandburg, Arithmetic, in H. Plotz (ed) *Imagination's Other Place, Poems of Science and Mathematics*, Thomas Y Crowell, 1955.
- [4] Anon, There was an old man who said “Do, in H. Plotz (ed) *Imagination's Other Place, Poems of Science and Mathematics*, Thomas Y Crowell, 1955.
- [5] M.C. Bartholomew-Biggs, *Nonlinear Optimization with Financial Applications*, Kluwer, 2005
- [6] M.C. Bartholomew-Biggs, *Nonlinear Optimization with Engineering Applications*, Springer, 2008
- [7] Stanislaus Lem, The Cyberiad, in Sarah Glaz & JoAnne Growney (eds) *Strange Attractors: Poems of Love and Mathematics*, A.K. Peters, 2008.
http://www.math.uconn.edu/~glaz/Strange_Attractors/Cyberiad_by_Stanislaw_Lem.html
- [8] A.R. Conn, N.I.M. Gould & Ph. L. Toint, *Trust Region Methods*, MPS-SIAM Series on Optimization, 2000
- [9] W.H. Auden, Numbers and Faces, in JoAnne Growney, (ed), *Numbers and Faces: A Collection of Poems with Mathematical Imagery*, Humanistic Mathematics Network, 2001.
<http://joannegrowney.com/numbersandfaces.pdf>
- [10] Gospel of St Matthew, Chapter 20:,vv 1-16
- [11] P.E. Gill, W. Murray & M.H. Wright, *Practical Optimization*, Academic Press, 1982
- [12] J. Lennon & P. McCartney, *We Can Work it Out*, Apple Music, 1965
- [13] C.G. Broyden & M.T. Vespucchi, *Krylov Solvers for Linear Algebraic Systems*, Studies in Computational Mathematics **11**, Elsevier, 2004
- [14] T.S. Eliot, The Hollow Men, *Collected Poems 1909-1962*, Faber & Faber, 1963
- [15] Sarah Glaz, Calculus, in Sarah Glaz & JoAnne Growney (eds) *Strange Attractors: Poems of Love and Mathematics*, A.K. Peters, 2008.
http://www.math.uconn.edu/~glaz/Strange_Attractors/Calculus_by_Sarah_Glaz.html