# Global Convergence of a Curvilinear Search for Non-Convex Optimization

Michael Bartholomew-Biggs*, Salah Beddiaf and Bruce Christianson
University of Hertfordshire, Hatfield, UK

April 2022

**Abstract**

For a non-convex function $f : R^n \to R$ with gradient $g$ and Hessian $H$, define a step vector $p(\mu, x)$ as a function of scalar parameter $\mu$ and position vector $x$ by the equation $(H(x) + \mu I)p(\mu, x) = -g(x)$.

Under mild conditions on $f$, we construct criteria for selecting $\mu$ so as to ensure that the algorithm $x := x + p(\mu, x)$ descends to a second order stationary point of $f$, and avoids saddle points.

'Ah, you have said something true and so untidy,' complained Constantine, 'and what I said was not quite true, but so beautifully neat.'

*Black Lamb and Grey Falcon*
Rebecca West

**Keywords:** Nonlinear optimization; Newton-like methods; Non-convex functions.

## 1 Introduction and Context

In this paper we consider iterative methods for unconstrained minimization of a continuous $n$-variable function $f(x)$ whose first and second derivatives are available – that is, at any point of interest $x$ we can compute a gradient vector $g$ and a Hessian matrix $H$. In this situation the powerful Newton method can be used, which generates a new solution estimate $x + p$ using

$$p = -\alpha H^{-1} g \tag{1}$$

*matqmb@herts.ac.uk

where $\alpha$ is a positive scalar parameter determined by a line search to ensure that $f(x+p) < f(x)$. Essentially this amounts to taking a step along the direction towards the stationary point of a *local quadratic model* of $f$; and this works well when $H$ is positive definite and the stationary point of the local quadratic model is a minimum. In this situation the steplength choice $\alpha = 1$ is usually acceptable[1].

However if $H$ is not positive definite then the Newton step represents a move towards a saddle point or even a maximum; and this is unlikely to facilitate finding a minimum of $f(x)$, especially if the saddle point of the local quadratic model is close to a saddle point of $f$, and the algorithm becomes trapped there.

A strategy that has recently been discussed in [5], [3] is to use the iteration step obtained by solving

$$(H + \mu I)p = -g \tag{2}$$

for $p$ whenever $H$ is non-positive definite. In this scheme, the scalar $\mu > -\lambda$ is used as a search parameter, where $\lambda$ is the most negative eigenvalue of $H$. As $\mu$ varies, the new point $x + p(\mu)$ will typically lie on a curvilinear path.

As explained in [5], [3] the iteration given by solving (2) can be regarded as a *trust-region* approach [9] or as a *gradient-flow* method based on an approximation to a step along the continuous steepest descent path defined by the differential equation

$$\frac{dx}{dt} = -g.$$

Both these approaches have been extensively studied; but the work in [3] seems to be novel in the way it employs curvilinear searches based on the parameter $\mu$.

The use of equation (2) in non-convex optimization was popularized by Goldfeld, Quandt, and Trotter [11] in 1966. The first proposal to use $\mu$ as a search parameter, in order to avoid the need for a complete eigensolution of $H$, seems to have been made by Hebden [12] in 1973; more recent suggestions are due to Brown [7] in 1986, and Behrman [6] in 1998. A different, but related, idea is due to Higham [13], who describes a trust-region approach which is driven by adjustment of $\mu$ on each iteration instead of adjustment of a trust region radius.

It is worth mentioning that [3] describes an algorithm based on (2) in which the systematic adjustment of $\mu$ requires some knowledge of the eigenvalues of $H$. Full knowledge of the spectrum would require us to compute the rather costly eigensolution of the matrix $H$. However this potential drawback is alleviated in a working paper [2], which shows that we only need to determine the extreme eigenvalues of the Hessian. There are many ways to achieve this[2], the approach we use in [2] employs the power method.

Let $\Lambda$ denote the most positive (or least negative) eigenvalue of $H$, and $\lambda$ the most negative (or least positive). Applying the power method to $H$ gives the

[1] Usually, but not always: for example, the hyperbola given by $f(x) = (1 + x^2)^{1/2}$ with $x \in R^n$ is convex, but $H^{-1}g = (1 + x^2)x$, so taking $\alpha = 1$ oscillates for $\|x\| = 1$ and diverges for $\|x\| > 1$.

[2] See for example Chapter F02 of the NAG Toolbox at `www.nag.co.uk`

eigenvalue estimate $\hat{\lambda}$, corresponding to whichever of $\Lambda, \lambda$ has the larger absolute magnitude. Applying the power method to $H - \hat{\lambda} I$ now gives the eigenvalue estimate $\check{\lambda} - \hat{\lambda}$, where $\check{\lambda}$ is the other extreme eigenvalue of $H$.

The chief distinctive features of the approach proposed in [2] are the use of the power method mentioned above, and the particular way of adjusting $\mu$ to obtain trial points, which is based on controlling the condition number of the matrix $H + \mu I$.

Some quite extensive and encouraging computational experience with algorithms based on (2) has been given in [3] and [2]. The purpose of the present paper is to establish convergence properties of the iteration (2) when our algorithm is used to choose $\mu$. A good review of known convergence results for iterations of this form is given by [14].

Although the subsequent sections of this paper will be concerned purely with convergence properties of our algorithm, rather than performance, we conclude this introduction with three remarks of a more general nature.

(1) Cholesky factorization is often used both to determine if $H$ is positive definite and also – of necessity – to check the positive definiteness of $H + \mu I$ for an increasing sequence of $\mu$ values. The reader is entitled to wonder whether such a simple approach to adjusting $\mu$ directly is more or less computationally expensive than the method proposed in [3, 2], where estimating eigenvalues enables us to obtain a positive definite $H + \mu I$ immediately and avoid a trial and error process involving several attempted Cholesky factorizations. The present paper does not aim to shed any new light on this point; but the numerical experience reported in [3] shows that methods using an eigenvalue-based adjustment of $\mu$ typically perform better than a naive implementation of Higham's method [13], which simply uses constant factors to increase or decrease $\mu$.

(2) It is well-known that the power method for determining eigenvalues converges slowly when the dominant eigenvalue is close in magnitude to one or more other eigenvalues. It turns out that the algorithm which we analyse in the present paper does not rely on particularly accurate estimates of the extreme eigenvalues $\Lambda$ and $\lambda$, and so rough (within 10%) but sufficiently accurate estimates can therefore be obtained quite quickly using the power method. This point is explored in more detail towards the end of Section 4 below.

(3) Some might object that any optimization method requiring explicit second derivatives is unsuitable in practice for problems involving large numbers of variables, and to add an eigenvalue calculation into the process is simply to make a bad situation worse. While this point – like the previous two – is peripheral to an investigation of the convergence properties of our algorithm, advances in Automatic Differentiation [4] have considerably changed the landscape around the issue, and we shall give this some attention in the concluding section of this paper.

The remainder of the paper is organized as follows. In the next section we establish some notation, motivate our algorithm by relating it to a trust-region approach, and sketch out how our convergence proof will proceed. In Section 3 we establish a number of technical lemmas giving conditions which ensure "sufficient" descent at each step. In Section 4 we set out our main algorithm in

detail and discuss its properties, including its tolerance for inaccurate eigenvalue estimates. In Section 5 we augment the algorithm to guarantee that it does not stall even if by chance it lands on a saddle point, and then set out our main convergence results. In the final section we review some possibilities to exploit further the techniques we have developed, particularly in the practical context of large-dimensional spaces.

### Useful Constants

A number of symbols make multiple appearances in the sequel. For convenience of reference we give a list of the main ones here, together with pointers to where they are defined:

| | |
|---|---|
| $\alpha_0$ | Lemma 3.2 |
| $\beta$ | Algorithm 4.1 |
| $\Delta f, \Delta q$ | Definition 2.4 |
| $e_\lambda$ | Lemma 5.2 |
| $\eta_1, \eta_2$ | Definition 2.4 |
| $\kappa_0$ | Algorithm 4.1 |
| $\kappa_c$ | Lemma 3.5 |
| $\kappa_{max}$ | Definition 2.3 |
| $K$ | after Definition 2.4 |
| $\lambda, \Lambda$ | Definition 3.4 |
| $L$ | Lemma 5.2 |
| $M$ | Lemma 3.1 |
| $q$ | Lemma 2.2 |

## 2 Preliminaries

Let $f$ be a function from $R^n$ to $R$, and let $x_0$ be a point in $R^n$. We assume that $f$ is bounded below, so that the set $\{f(y) : y \in R^n\}$ has a lower bound. From the given starting point $x_0$, we seek to construct a sequence of points $x_k = x_{k-1} + p_k$ which converges to a local minimum of $f$.

The *basin of descent* from $x_0$ is the set $B = \{y : f(y) \leq f(x_0)\}$. The *convex hull* of $B$, denoted $cxB$, is the set of all convex combinations of points in $B$.

We assume that $f$ is twice-differentiable on $cxB$, with gradient $g$ and Hessian $H$, and that there exists a global bound $M$ for the largest (in absolute value) eigenvalue of $H$ on $cxB$. The assumption of a global bound $M$ is satisfied in particular if $cxB$ is bounded and $H$ is Lipschitz-continuous on it.

At the $k$-th stage of our algorithm, by an abuse of notation, we write $x, f, g, H$ for $x_{k-1}, f(x_{k-1}), g(x_{k-1})$ and $H(x_{k-1})$. For any vector $s$ we write $\|s\|$ to denote the 2-norm defined by $\|s\|^2 = s \cdot s = s^2$

**Definition 2.1** *For a scalar $\mu$ such that $H + \mu I$ is positive definite, define $p$ as a function of $\mu$ by*

$$(H + \mu I)p = -g$$

Our approach will be to search along the curve $p(\mu)$ for a suitable value of $p$ to use as $p_k$. One motivation for doing this is that it is dual to a trust-region approach, in the sense made precise by the following Lemma.

**Lemma 2.2** *Let $q$ be the quadratic function defined by*

$$q(s) = g \cdot s + \frac{1}{2} s^T H s$$

*so that $f + q(s)$ is the quadratic model for $f(x + s)$ at $x$. Note that $q(0) = 0$.*

*If $\mu = 0$ then $p$ is the global minimizer for $q$. If $\mu > 0$ then $q(p)$ is the minimum value of $q(s)$ in the region $\{s : \|s\| \leq \|p\|\}$.*

*Proof:* $q(p)$ is the minimum value of $q(s)$ with $\|s\| = \|p\|$. This follows from the standard Lagrangean formulation $L(s, \mu) = q(s) + \mu(s^2 - r^2)/2$ by choosing $r = \|p\|$. Inclusion of the interior follows from the fact that as $\mu$ increases, $\|p\|$ decreases and $q(p)$ increases:

$$\frac{d}{d\mu}(\frac{1}{2}p^2) = p \cdot p' = -p(H + \mu I)^{-1} p < 0$$

since $Hp' + \mu p' + p = 0, p' = -(H + \mu I)^{-1} p$ ; and so

$$\frac{d\ q(p)}{d\mu} = g \cdot p' + p^T H p' = -\mu p \cdot p' > 0$$

*qed.*

In the trust-region approach both $\mu$ and $p$ are dependent variables and the independent variable is $r$, the radius of the trust region. In our curvilinear approach, we follow [13] and take $\mu$ as the independent variable. We shall show how to choose $\mu$ at each stage so that the algorithm $x_k = x_{k-1} + p(\mu)$ is globally convergent.

The basic idea is that if $H$ is very different to the value of the Hessian at the optimum, which is the case in particular if $H$ has directions of negative curvature, then we want $p$ to go far enough that $q$ is no longer a particularly good approximation to $f$, but not so far that a decrease in $q$ ceases to predict descent for $f$.

**Definition 2.3** *We say that $H + \mu I$ is well-conditioned if the condition number of $H + \mu I$ is less than some global parameter $\kappa_{max}$.*

**Definition 2.4** *Let $\eta_1, \eta_2$ be global parameters with $1 > \eta_1 > \eta_2 > 0$. Define*

$$\Delta q = q(0) - q(p), \qquad \Delta f = f(x) - f(x + p).$$

*We say that $\mu$ is* big enough *if $H + \mu I$ is positive definite and well-conditioned, and*

$$\Delta f \geq \eta_2 \Delta q.$$

*Similarly, we say that $\mu$ is* small enough *if $H + \mu I$ is positive definite and well-conditioned, and*

$$\Delta f \leq \eta_1 \Delta q$$

Typically we pick values such as $\eta_1 = 0.9, \eta_2 = 0.1$

We shall show that, under suitable conditions on $\mu$, we have $\Delta f \geq K\|g\|^2$ for some global constant $K$. From this and the assumption that $f$ is bounded below it follows that, for any $\varepsilon$, repeatedly replacing $x$ by $x+p$ eventually gives $\|g\| < \varepsilon$.

We prove this condition on $\Delta f$ in two pieces: we show how to choose $\mu$ to (i) bound $\Delta q$ below by a fixed multiple of $g^2$, and to (ii) ensure in addition that $\Delta f$ is bounded below by a fixed multiple of $\Delta q$.

Achieving the second condition (ii) is a straightforward matter of increasing $\mu$. Increasing $\mu$ decreases $\|p\|$ arbitrarily, and for short enough $p$ the quadratic model is (by definition) accurate to $o(p^2)$. If $H + \mu I$ is not well-conditioned, or if $H + \mu I$ is not positive definite, the value of $\mu$ can also be increased until it is.

However ensuring the first condition (i) under the constraint of the second is more delicate, and in the next section we shall explore some suitable approaches for achieving this.

# 3 Bounding $\Delta q$ below by $Kg^2$

Now we look at some conditions under which $\Delta q$ is bounded below by a global multiple of $g^2$.

**Lemma 3.1** *Suppose $H$ is positive definite and well-conditioned, and $\mu = 0$. Then*

$$\Delta q \geq \frac{g^2}{2M}$$

*where $M$ is a global bound for the largest (in absolute value) eigenvalue of $H(y)$ on $cxB$, the convex hull of the basin of descent from $x_0$.*

*Proof:* $p$ is the global minimizer of $q$, so $q(p) \leq q(-\alpha g)$ for any $\alpha$. But $q(-\alpha g)$ is a parabola in $\alpha$, with

$$q(-\alpha g) = -\alpha g^2 + \frac{1}{2}\alpha^2 g^T H g = \frac{1}{2} g^T H g \left[ \left( \alpha - \frac{g^2}{g^T H g} \right)^2 - \left( \frac{g^2}{g^T H g} \right)^2 \right]$$

$$= \frac{1}{2} g^T H g \left[ (\alpha - \alpha_0)^2 - \alpha_0^2 \right]$$

where $\alpha_0 = g^2 / g^T H g$. In particular putting $\alpha = \alpha_0$ gives the *perfect steepest descent* step $-\alpha_0 g$, and we have

$$\Delta q \geq -q(-\alpha_0 g) = \frac{1}{2}\alpha_0^2 g^T H g = \frac{g^4}{2 g^T H g} \geq \frac{g^2}{2M}$$

since $0 < g^T H g \leq M\|g\|^2$.
*qed.*

**Lemma 3.2** *Suppose $H$ has negative eigenvalues or is ill-conditioned, but that $H + \mu I$ is positive definite and well-conditioned.*

*Suppose that $g^T H g \neq 0$, and define $\alpha_0 = g^2/g^T H g$. Suppose that $p$ is at least as long as the perfect steepest descent step, i.e. that $\|p\| \geq |\alpha_0| \|g\|$.*

*Then $\Delta q$ is bounded below just as in Lemma 3.1.*

*Proof:* By assumption, $-|\alpha_0|g$ lies in the region for which $p$ minimizes $q$, so $\Delta q \geq -q(-|\alpha_0|g)$. If $g^T H g > 0$ then we can proceed exactly as in Lemma 1. If $g^T H g < 0$ then we have

$$q(-|\alpha_0|g) = \frac{1}{2} g^T H g \left[ (|\alpha_0| - \alpha_0)^2 - \alpha_0^2 \right] = \frac{3}{2} (g^T H g) \alpha_0^2 = \frac{-3g^4}{2|g^T H g|.}$$

But $|g^T H g| \leq M g^2$ and so

$$\Delta q \geq -q(-|\alpha_0|g) = \frac{3g^2}{2} \cdot \frac{g^2}{|g^T H g|} > \frac{g^2}{2M}$$

as required.
*qed.*

**Lemma 3.3** *Suppose (as in Lemma 3.2) that $H$ has negative eigenvalues or is ill-conditioned, but $H + \mu I$ is positive definite and well-conditioned.*

*Suppose that either $g^T H g = 0$, or else that $g^T H g \neq 0$ and $\|p\| < |\alpha_0| \|g\|$ where (as usual) $\alpha_0 = g^2/g^T H g$.*

*Then*

$$\Delta q \geq \frac{1}{2} \|g\| \|p\|$$

*Proof:* Let $\zeta = \|p\|/\|g\|$. If $g^T H g = 0$ then since $p$ minimizes $q(s)$ for $\|s\| = \|p\|$ we have $\Delta q \geq -q(-\zeta g) = \zeta g^2 = \|p\| \|g\|$.

If $g^T H g < 0$ then $\Delta q \geq -q(-\zeta g) = \zeta g^2 - \zeta^2 g^T H g/2 \geq \zeta g^2 = \|p\| \|g\|$.

If $g^T H g > 0$ then $q(-\alpha g)$ is convex in $\alpha$, and $0 < \zeta < \alpha_0$, so $q(-\zeta g)$ lies below the straight line interpolating between $q(0)$ and $q(-\alpha_0 g)$. This means that

$$q(-\zeta g) \leq \frac{\zeta}{\alpha_0} q(-\alpha_0 g)$$

To see this formally, observe that $\zeta < \alpha_0$ by assumption on $p$, and $g^T H g > 0$, so

$$\alpha_0 q(-\zeta g) = \alpha_0(-\zeta g^2 + \frac{1}{2}\zeta^2 g^T H g) = -\alpha_0 \zeta g^2 + \frac{1}{2}\alpha_0 \zeta^2 g^T H g$$

$$< -\alpha_0 \zeta g^2 + \frac{1}{2}\alpha_0^2 \zeta g^T H g = \zeta(-\alpha_0 g^2 + \frac{1}{2}\alpha_0^2 g^T H g) = \zeta q(-\alpha_0 g).$$

Now $\alpha_0 = g^2/g^T H g$ and $q(-\alpha_0 g) = -g^4/(2 g^T H g) = -\alpha_0 g^2/2$ (from the proof of Lemma 3.1), so

$$\frac{\zeta}{\alpha_0} q(-\alpha_0 g) = -\frac{1}{2}\zeta g^2 = -\frac{1}{2}\|p\| \|g\|,$$

so $\Delta q = -q(p) \geq -q(-\zeta g) \geq \frac{1}{2}\|p\| \|g\|$.
*qed.*

**Definition 3.4** *Let $\Lambda$ be the most positive (or least negative) eigenvalue of $H$, and let $\lambda$ be the most negative (or least positive). Then for $\mu > -\lambda$ we have*

$$\frac{\Lambda + \mu}{\lambda + \mu} = \kappa$$

*where $\kappa$ is the condition number for $H + \mu I$.*

**Lemma 3.5** *Suppose (as previously) that $H$ has negative eigenvalues or is ill-conditioned, but $H + \mu I$ is positive definite and well-conditioned.*

*Suppose (as in Lemma 3.3) that $g^T H g = 0$ or that $\|p\| < |\alpha_0| \|g\|$.*

*Suppose that $\kappa \geq \kappa_c$, where $\kappa_c$ is a global parameter with $1 < \kappa_c < \kappa_{max}$.*

*In this case we have*

$$\Delta q \geq \frac{\kappa_c - 1}{\kappa_c} \cdot \frac{g^2}{4M}$$

*Proof:* Definition 3.4 gives

$$\kappa - 1 = \frac{\Lambda - \lambda}{\lambda + \mu}; \qquad \mu = -\lambda + \frac{\Lambda - \lambda}{\kappa - 1}; \qquad \Lambda + \mu = \frac{\kappa}{\kappa - 1}(\Lambda - \lambda)$$

and $p = -(H + \mu I)^{-1} g$, so

$$\|p\| \geq \frac{\|g\|}{\Lambda + \mu} = \frac{\kappa - 1}{\kappa} \cdot \frac{\|g\|}{\Lambda - \lambda}$$

now $\Lambda - \lambda \leq 2M$, thus by Lemma 3.3

$$\Delta q \geq \frac{1}{2} \|p\| \|g\| \geq \frac{\kappa - 1}{4\kappa M} g^2.$$

We have $\kappa \geq \kappa_c$, so $1 - 1/\kappa \geq 1 - 1/\kappa_c$
*q.e.d.*

For example, if we take $\kappa_c = 2$ then $(\kappa_c - 1)/\kappa_c = 1/2$ and $\Delta q \geq g^2/8M$.

For our final result in this section, we show that the quadratic model for $f$ only starts to break down when $\Delta q$ is sufficiently large in terms of $g^2$.

**Lemma 3.6** *Suppose (once again) that $H$ has negative eigenvalues or is ill-conditioned, but $H + \mu I$ is positive definite and well-conditioned.*

*Suppose (as in Lemmas 3.3 and 3.5) that $g^T H g = 0$ or that $\|p\| < |\alpha_0| \|g\|$.*

*Suppose that $\mu$ is small enough, so that $\Delta f \leq \eta_1 \Delta q$, where $0 < \eta_1 < 1$ is a global constant.*

*Then we have*

$$\Delta q \geq \frac{1 - \eta_1}{4M} \|g\|^2.$$

*Proof:* $\Delta f \leq \eta_1 \Delta q$ implies

$$(1 - \eta_1) \Delta q \leq \Delta q - \Delta f,$$

and by the second mean value theorem, for some $0 < \theta < 1$,

$$-\Delta f = f(x+p) - f(x) = p \cdot g + \frac{1}{2} p^T H^\theta p$$

where $H^\theta = H(x + \theta p)$. This gives

$$(1-\eta_1)\Delta q \leq \Delta q - \Delta f = p \cdot g + \frac{1}{2} p^T H^\theta p - p \cdot g - \frac{1}{2} p^T H p = \frac{1}{2} p^T (H^\theta - H) p \leq M p^2$$

whence

$$\frac{1}{1-\eta_1} M p^2 \geq \Delta q \geq \frac{1}{2} \|p\| \|g\|$$

by Lemma 3.3. It follows that $\|p\| \geq \|g\|(1-\eta_1)/2M$ and so

$$\Delta q \geq \frac{1}{2} \|p\| \|g\| \geq \frac{1-\eta_1}{4M} \|g\|^2$$

*qed.*

# 4 Choosing $\mu$

Now we describe our strategy for choosing $\mu$ along the lines laid out in [2], and show that this strategy ensures global convergence.

We begin by setting out the algorithm, and then walk through its properties. In the algorithm, $\kappa_0$ is an initial trial value for the condition number of $H + \mu I$, with $\kappa_c \leq \kappa_0 < \kappa_{max}$ ; $\beta$ is a (fixed) interpolation constant with $0 < \beta < 1$ ; and whenever we set a new value for $\mu$, we calculate the corresponding values for $p, \Delta q$, and $\Delta f$.

**Algorithm 4.1** *Choosing $\mu$.*

```
if H is positive definite
    if Λ/λ > κ_max
        set μ_b := −λ + (Λ − λ)/(κ_max − 1)
    else
        set μ_b := 0
    endif
    set oktoreducemu := false
else
    if Λ > λ
        set μ_b := −λ + (Λ − λ)/(κ_0 − 1)
    else
        set μ_b := −λ + 1
    endif
    set oktoreducemu := true
endif
```

```
if μ_b is big enough and oktoreducemu
     set κ_b := (Λ + μ_b)/(λ + μ_b)
     while κ_b ≤ κ_max and μ_b is not small enough
          set μ_a := μ_b
          set μ_b := β(μ_a + λ) − λ
          set κ_b := (Λ + μ_b)/(λ + μ_b)
     endwhile
     if κ_b > κ_max or μ_b is not big enough
          accept μ_a
     else
          accept μ_b
     endif
else
     while μ_b is not big enough
          set μ_a := μ_b
          set μ_b := (μ_a + λ)/β − λ
     endwhile
     accept μ_b
endif
```

If $H$ is positive definite and well conditioned, and $\mu = 0$ is big enough, then we take $p$ to be the Newton step. In this case $\Delta f$ meets the termination condition $\Delta f \geq Kg^2$ for a global constant $K$, by Lemma 3.1 and the definition of big enough. Otherwise, we have the more problematic case that $H$ is not positive definite, or is not well-conditioned, or the Newton step is too long to give quadratic descent. In all these cases, the remedy is to choose a suitable positive value of $\mu$. Apart from one edge case, our algorithm for adjusting $\mu$ can be interpreted as systematically inflating or deflating the condition number $\kappa$ of $H + \mu I$.

This (slightly annoying) edge case occurs when $\lambda = \Lambda$, all the eigenvalues of $H$ are equal, $\kappa = 1$ for all candidate values of $\mu$, $H$ is a multiple of the identity matrix, and $p$, which minimizes $q(s)$, is a vector in the direction $-g$. Our algorithm still works perfectly well in this case, but the "curvilinear search" degenerates to gradient descent.

If $H$ is positive definite but not well-conditioned, we choose $\mu$ so as to reduce the condition number of $H + \mu I$ to $\kappa_{max}$. Here we use the fact, given at the start of the proof of Lemma 3.5, that

$$\mu + \lambda = \frac{\Lambda - \lambda}{\kappa - 1}.$$

If $H$ is not positive definite, we use the same trick to choose $\mu$ so that $H + \mu I$ has condition number $\kappa_0$, where $\kappa_0$ is a trial value with $\kappa_c \leq \kappa_0 < \kappa_{max}$. (In the edge case, we rather arbitrarily set $H + \mu I$ to be the identity.) Ideally, in the non-convex case, we seek a value of $\mu$ for which

$$\eta_1 \Delta q \geq \Delta f \geq \eta_2 \Delta q.$$

The first condition with $\eta_1$ says that $\mu$ is small enough (and $p$ is long enough) that the quadratic model is starting to break down, and the second condition with $\eta_2$ says that $\mu$ is big enough (and $p$ is short enough) to ensure that there is sufficient descent for $f$ relative to that of the quadratic model $q$.

If both conditions are met then the step $\Delta f$ meets the termination condition $\Delta f \geq Kg^2$ either by Lemma 3.2 or by Lemma 3.6. It's not an issue for termination if $\mu$ is not small enough, as any step with $\kappa \geq \kappa_c$ for which $\mu$ is big enough will suffice, by Lemmas 3.2 and 3.5, and in the edge case we have $\Delta q \geq g^2$ anyway.

However, when $H$ is not positive definite, and $\mu$ is big enough but not small enough, it is worth *extrapolating*, i.e. seeing if we can get a longer step with better descent by choosing a smaller value of $\mu$, which corresponds to increasing $\kappa$. We extrapolate by multiplying $\mu + \lambda$ by $\beta$ for some global constant $\beta$ with $0 < \beta < 1$. From Definition 3.4 and the proof of Lemma 3.5, this corresponds to dividing $\kappa - 1$ by $\beta$, because

$$\kappa_b - 1 = \frac{\Lambda - \lambda}{\mu_b + \lambda} = \frac{\Lambda - \lambda}{\beta(\mu_a + \lambda)} = \frac{\kappa_a - 1}{\beta}.$$

Eventually one of two things will happen: either we reach a value of $\mu$ that is small enough, or we reach the conditioning limit $\kappa_{max}$. In the second case we stop and accept the previous step; in the first case we accept the most recent value of $\mu$ that was big enough, which will always be one of the final two values. This is because any value of $\mu$ for which $H + \mu I$ is positive definite and well-conditioned is either big enough, or small enough, or both.

The other case in which we need to adjust $\mu$ is where the initial value for $\mu$ is not big enough. In this case we *interpolate* in order to increase $\mu$ and obtain a shorter step. We don't extrapolate when $H$ is positive definite, as that would result in negative values for $\mu$, but we interpolate whenever $\mu$ is not big enough, regardless of whether or not $H$ is positive definite. We interpolate by dividing $\mu + \lambda$ by $\beta$, which corresponds to multiplying $\kappa - 1$ by $\beta$, and we repeat this until $\mu_b$ is big enough.

However, once $\kappa < \kappa_c$ and $p$ is shorter than the perfect steepest descent step, we are relying on Lemma 3.6 for the termination condition, which means that we need a value of $\mu$ that is small enough as well as one that is big enough. What should we do if $\mu_a$ is not big enough, but $\mu_b$ is not small enough? This happens when

$$\Delta f_a < \eta_2 \Delta q_a \quad \text{but} \quad \Delta f_b > \eta_1 \Delta q_b.$$

By continuity there must be a value $\mu$ between $\mu_a$ and $\mu_b$ with

$$\eta_1 \Delta q \geq \Delta f \geq \eta_2 \Delta q$$

but we do not need to find it: instead we can just take $x + p_b$ as the new point. We prove this next.

**Lemma 4.2** *If $\mu_a$ is not big enough and $\mu_b$ is not small enough, then we can accept the step $p_b$.*

*Proof:* $H + \mu_b I$ is certainly well-conditioned and positive definite. If $g^T H g \neq 0$ and $\|p_b\| > |\alpha_0|\|g\|$, the length of the perfect steepest descent step, then we could just accept $p_b$ anyway, by Lemma 3.2. Similarly, if the condition number $\kappa_b$ of $H + \mu_b I$ were greater than $\kappa_c$ then we could accept $p_b$ by Lemma 3.5. So we can assume $\mu_b + \Lambda < \kappa_c(\mu_b + \lambda)$.

Now we have:

$$q(p) = g \cdot p + \frac{1}{2} p^T H p = \frac{1}{2} g \cdot p - \frac{1}{2} \mu p^2$$

since $g + (H + \mu I)p = 0$ so $g \cdot p + p^T H p + \mu p^2 = 0$. Thus

$$\Delta q_a = \frac{1}{2} g^T (H + \mu_a I)^{-1} g + \frac{1}{2} \mu_a g^T (H + \mu_a I)^{-2} g$$

$$\leq \frac{1}{2} \frac{g^2}{\mu_a + \lambda} + \frac{\mu_a}{2} \frac{g^2}{(\mu_a + \lambda)^2} = \frac{g^2}{2} \cdot \frac{2\mu_a + \lambda}{(\mu_a + \lambda)^2}$$

similarly

$$\Delta q_b = \frac{1}{2} g^T (H + \mu_b I)^{-1} g + \frac{1}{2} \mu_b g^T (H + \mu_b I)^{-2} g \geq \frac{g^2}{2} \cdot \frac{2\mu_b + \Lambda}{(\mu_b + \Lambda)^2}$$

so

$$\frac{\Delta q_b}{\Delta q_a} \geq \frac{2\mu_b + \Lambda}{2\mu_a + \lambda} \cdot \left[ \frac{\mu_a + \lambda}{\mu_b + \Lambda} \right]^2 \geq \frac{\beta^2}{\kappa_c^2}$$

because $\mu_b > \mu_a; \Lambda \geq \lambda; \mu_a + \lambda = \beta(\mu_b + \lambda)$; and $\mu_b + \Lambda \leq \kappa_c(\mu_b + \lambda)$. Since $\Delta f_b > \eta_1 \Delta q_b$ by assumption, we have

$$\Delta f_b > \frac{\eta_1 \beta^2}{\kappa_c^2} \Delta q_a$$

Now if $g^T H g \neq 0$ and $p_a$ is longer than the perfect steepest descent step, then we have $\Delta q_a > g^2/2M$ just as in Lemma 3.2. Otherwise by Lemma 3.3 we have $\Delta q_a > \frac{1}{2}\|p_a\|\|g\|$, and since $\Delta f_a < \eta_2 \Delta q_a$ by assumption, the argument of Lemma 3.6 with $\eta_2$ in place of $\eta_1$ gives

$$\frac{1}{1 - \eta_2} M p_a^2 \geq \Delta q_a \geq \frac{1}{2} \|p_a\| \|g\|.$$

It follows that $\|p_a\| \geq \|g\|(1 - \eta_2)/2M$ and so

$$\Delta q_a \geq \frac{1}{2} \|p_a\| \|g\| \geq \frac{1 - \eta_2}{4M} \|g\|^2$$

In either case the previous inequality for $\Delta f_b$ in terms of $\Delta q_a$ now gives

$$\Delta f_b > \frac{\eta_1 \beta^2}{\kappa_c^2} \cdot \frac{1 - \eta_2}{4M} \|g\|^2$$

so $\Delta f_b$ is bounded below by $Kg^2$ for a suitable global constant $K$.

*qed.*

Algorithm 4.1 requires values to be provided for $\Lambda$ and $\lambda$. Fortunately it turns out, as far as our proof of convergence is concerned, that even very rough approximations to the extreme eigenvalues suffice, and we end this section with a discussion showing why this is so. We start with a Lemma.

**Lemma 4.3** *Let $\tilde{\Lambda}, \tilde{\lambda}$ be estimates of $\Lambda, \lambda$ satisfying*

$$0 < \Lambda - \lambda \leq \tilde{\Lambda} - \tilde{\lambda} \leq 1.33(\Lambda - \lambda) \; ; \quad 0 \leq \lambda - \tilde{\lambda} \leq 0.33(\Lambda - \lambda).$$

*Assume that $\kappa_0 \geq 3$ and define $\mu$ by*

$$\mu = -\tilde{\lambda} + \frac{\tilde{\Lambda} - \tilde{\lambda}}{\kappa_0 - 1}.$$

*Then $H + \mu I$ is positive definite and well-conditioned, with condition number $\kappa$ satisfying $2 \leq \kappa \leq \kappa_0$.*

*Proof:* $\lambda \geq \tilde{\lambda}$ so (unless we are in the edge case $\Lambda = \lambda$), $\mu + \lambda \geq \mu + \tilde{\lambda} = (\tilde{\Lambda} - \tilde{\lambda})/(\kappa_0 - 1) > 0$, so $H + \mu I$ is certainly positive definite, and

$$\kappa - 1 = \frac{\Lambda - \lambda}{\mu + \lambda} \leq \frac{\tilde{\Lambda} - \tilde{\lambda}}{\mu + \tilde{\lambda}} = \kappa_0 - 1$$

using Definition 3.4 for the first equality. It remains to show that $\mu + \lambda \leq \Lambda - \lambda$. But

$$\mu + \lambda = \mu + \tilde{\lambda} + (\lambda - \tilde{\lambda}) = \frac{\tilde{\Lambda} - \tilde{\lambda}}{\kappa_0 - 1} + (\lambda - \tilde{\lambda}) \leq (\Lambda - \lambda) \left( \frac{1.33}{\kappa_0 - 1} + 0.33 \right)$$

and $\kappa_0 \geq 3$.
*qed.*

Algorithm 4.1 does not *require* extrapolation: for correctness it is enough that our initial trial value for $\mu$ give a condition number $\kappa$ for $H + \mu I$ that lies between $\kappa_c$ and $\kappa_{max}$, and (as remarked after Lemma 3.5) we may take $\kappa_c = 2$. It follows that eigenvalue estimates that satisfy the conditions of Lemma 4.3 suffice to ensure convergence for Algorithm 4.1.

In the approach described in [2], the power method is first applied to $H$ to estimate whichever of $\Lambda$ and $\lambda$ has the larger absolute value, and then to $H - \Lambda I$ (or $H - \lambda I$) to estimate $\Lambda - \lambda$ and thence the smaller eigenvalue. Let us suppose that we converge our power estimates until they are at least 91% of the quantity being estimated, and then round them up, i.e. away from zero, by ten percent.

If $H$ is indefinite, then $\Lambda - \lambda > \max |\Lambda|, |\lambda|$. In this case our estimates satisfy the conditions of Lemma 4.3, with $\tilde{\Lambda} - \tilde{\lambda} \leq 1.21 \; (\Lambda - \lambda) \; ; \lambda - \tilde{\lambda} \leq 0.11 \; (\Lambda - \lambda)$. (The extra 1% is due to a compound error in the estimated value of an endpoint.)

If $H$ is negative definite and has a condition number greater than 2.0, then $2\Lambda > \lambda$ so $2(\Lambda - \lambda) > -\lambda$. In this case the same argument avails, with coefficients of 1.32 and 0.20 respectively.

We can reduce the remaining case, where $|\Lambda| > \Lambda - \lambda$, to the indefinite case by adding a suitable multiple of the identity to $H$, and starting the power method again. But here $H$ is very well conditioned (condition number less than 2), so as observed earlier we lose little by taking $p = -g$ as our initial guess, and this alternative approach also allows our convergence proof to go through.

The power method is quite efficient at obtaining estimates of a dominant eigenvalue to within ten percent: eigencomponents for eigenvalues less than 91% of the dominant value die rapidly away, and eigenvalues within 10% of the dominant value are simply rounded up into the estimate.

Of course, we may desire to exploit more accurate eigenvalue estimates for performance reasons, and apply techniques such as Aitken acceleration [1, 10]; but for the purpose of establishing convergence, which is our concern in this paper, the crude estimates discussed here suffice.

# 5 Avoiding Saddle Points

Algorithm 4.1 tends to keep away from saddle points, because it includes downhill components along directions of negative curvature. But we are deliberately trying to choose a step $p$ that is long enough for the Hessian to change dramatically, and so we may find ourselves moving onto a saddle point by sheer bad luck.

In this case $g = 0$, and the constrained minimum of $q(s) = s^T Hs/2$ subject to $\|s\| = r$ is $p = re_\lambda$, where $e_\lambda$ is a unit eigenvector for the eigenvalue $\lambda$. In terms of a solution to $(H + \mu I)p = -g$ this corresponds to the positive semi-definite case $(H - \lambda I)p = 0$ with $\mu = -\lambda$ and $g = 0$.

In this case we can perform a more conventional line search along the line $re_\lambda$ in order to obtain sufficient descent to ensure eventual convergence to a local minimum. We next give the algorithm for this line search, and then prove that it has the required properties.

In the algorithm below, whenever we set a new value for $r$, we put $p = re_\lambda$ and calculate the corresponding values for $\Delta q$ and $\Delta f$.

**Algorithm 5.1** *Choosing $r$.*

```
set  r_a := 1, r_b := 1
while  η1 Δq_b < Δf_b
     set  r_a := r_b
     set  r_b := r_a/β
endwhile
while  Δf_a < η2 Δq_a
     set  r_b := r_a
     set  r_a := β r_b
endwhile
accept  p_a
```

The analysis of this algorithm is similar to that of Algorithm 3 in [8]. The first while loop must terminate because $f$ is bounded below and $\lambda$ is negative. The second must terminate because $f$ is twice-differentiable. If the first precondition is true then the second is false, so we have the post-condition that $r_b$ satisfies the condition with $\eta_1$ and $r_a$ that with $\eta_2$, and either $r_b = r_a$ or $\beta r_b = r_a$. Just as in Algorithm 4.1, we do not insist on finding a single step that satisfies both conditions, and at most one of the while loops will be performed.

**Lemma 5.2** *Assume that $\lambda < 0$ and that $e_\lambda$ is a unit eigenvector of $H$ corresponding to $\lambda$, with $g \cdot e_\lambda \leq 0$. (If $g \cdot e_\lambda > 0$ then swap $e_\lambda$ for $-e_\lambda$.)*

*Assume that $H$ is Lipschitz-continuous on the convex hull of the basin of descent, with Lipschitz constant $L$.*

*Then at the end of Algorithm 5.1 we have $\Delta f_a \geq K_2(-\lambda)^3$ for some global constant $K_2$.*

*Proof:* Define $d(r) = f(x + re_\lambda) - f(x)$. Then $d'(r) = e_\lambda \cdot g(x + re_\lambda)$, and

$$d''(r) = e_\lambda^T H(x + re_\lambda)e_\lambda = e_\lambda^T He_\lambda + e_\lambda^T(H(x + re_\lambda) - H)e_\lambda \leq \lambda + Lr$$

Hence, writing $d_1$ for $g \cdot e_\lambda$, we have

$$d'(r) \leq d_1 + \lambda r + Lr^2/2, \qquad \text{so} \quad d(r) \leq d_1 r + \lambda r^2/2 + Lr^3/6.$$

But $\eta_1 \Delta q_b \geq \Delta f_b$, and $\Delta q_b = -d_1 r_b - \lambda r_b^2/2; \Delta f_b = -d(r_b) \geq \Delta q_b - Lr_b^3/6$, so

$$(1 - \eta_1)\Delta q_b \leq \Delta q_b - \Delta f_b \leq \frac{1}{6} Lr_b^3$$

and $d_1 \leq 0$ so

$$(-\lambda)r_b^2/2 \leq -d_1 r_b + (-\lambda)r_b^2/2 = \Delta q_b$$

whence

$$\frac{1}{6} Lr_b^3 \geq \frac{1}{2}(1 - \eta_1)(-\lambda)r_b^2$$

and $r_b > 0$ so $r_b \geq 3(1 - \eta_1)(-\lambda)/L$, whence

$$r_a \geq 3\beta \frac{(1 - \eta_1)(-\lambda)}{L}$$

Since $\Delta f_a \geq \eta_2 \Delta q_a$ and $\Delta q_a \geq (-\lambda)r_a^2/2$, we have $\Delta f_a \geq \eta_2(-\lambda)r_a^2/2$, whence

$$\Delta f_a \geq \frac{9\beta^2 \eta_2 (1 - \eta_1)^2}{2L^2}(-\lambda)^3.$$

*qed.*

Putting all these pieces together gives the following:

**Theorem 5.3** *Let $f$ be twice-differentiable with uniformly bounded Hessian on the convex hull of the basin of descent from $x_0$, and choose the sequence $x_k$*

*using Algorithm 4.1. Then for any $\varepsilon > 0$, we have that $\|g_k\| \geq \varepsilon$ for only a finite number of values of $k$.*

*If the Hessian is Lipschitz continuous on the convex hull of the basin of descent, and we use Algorithm 5.1 to choose the next point whenever $\|g_k\| < \varepsilon$, then $\lambda_k \leq -\varepsilon$ for only a finite number of values of $k$. Therefore eventually we will come to an $k$ with $\|g_k\| < \varepsilon$ and $H_k + \varepsilon I$ positive definite.*

**Theorem 5.4** *Let $f$ be twice-differentiable with Lipschitz continuous Hessian on the convex hull of the basin of descent from $x_0$, pick $\varepsilon > 0$, and define the sequence $x_k$ as follows:*
*if $\|g_{k-1}\| \geq \varepsilon$ use Algorithm 4.1 to choose $x_k$;*
*if $\|g_{k-1}\| < \varepsilon$ and $\lambda_{k-1} \leq -\varepsilon$ use Algorithm 5.1 to choose $x_k$;*
*if $\|g_{k-1}\| < \varepsilon$ and $\lambda_{k-1} > -\varepsilon$ replace $\varepsilon$ by $\varepsilon/2$ and test again.*

*If the basin of descent is bounded then, by compactness and the previous Theorem, a subsequence of $x_k$ converges to a second order stationary point $x_*$ for $f$, ie a point where $\|g_*\| = 0$ and $H_*$ is at least positive semidefinite. (Alternatively, the sequence $x_k$ may terminate at such a point after a finite number of steps.) If $H_*$ is positive definite, then the entire sequence $x_k$ converges to $x_*$, and $x_*$ is a local minimum.*

This is about the best that we can hope for from a purely local second-order method: for example, if $f(x) = x^4 + x^3$ and $x_0 = 1.5$, then the Newton method converges to the point of inflection at $x = 0$, rather than to the second order minimum at $x = -0.75$.

# 6 Further Developments

The positive semidefinite case $(H - \lambda I)p = -g$ is also of potential interest, primarily from a performance point of view, when we find ourselves on a ridge leading down to a saddle point. This can occur when $g$ is significantly non-zero, but $g \cdot e_\lambda \approx 0$.

For example, if $q(u, v) = u^2 - 2u - v^2, g = (-2, 0), H = ((2, 0), (0, -2))$, then $q$ has a saddle point at $(1, 0)$ but for any $\mu > 2$ we have $\|p\| < 1/4$. However, adding a multiple of $e_\lambda$ to such a value of $p$ allows us to fall off the ridge, and potentially move further at the next step.

It is therefore worth pointing out that our termination proof does not require us to accept the step $p$. Provided $\mu$ is chosen so that $\Delta q$ is bounded below by a global multiple of $g^2$, it suffices for termination to accept any new value $x_k = x + s$ for $x$ for which $f(x) - f(x + s) > \eta_2 \Delta q$.

In the case where $g$ is significantly non-zero, but $e_\lambda$ is nearly orthogonal to the solution $p$ given by Algorithm 4.1, we can improve performance without affecting termination by adding to $p$ a component in the direction of $e_\lambda$, so as to further reduce $f$.

We have shown that under mild conditions, an algorithm using the iteration $x := x + p$ with $p$ satisfying $(H + \mu I)p = -g$ will converge for suitable values of $\mu$. It is worth adding the comment that in the case where the algorithm

converges to a local minimum $x_*$ of $f(x)$ at which $H(x_*)$ is positive definite, there will be a convex region around $x_*$ in which the choice $\mu = 0$ will be made, and then the algorithm becomes the classic Newton step, given by $Hp = -g$. It is well-known that Newton's method has an ultimately quadratic convergence rate which is therefore inherited by our approach. One could propose a hybrid algorithm which only uses $\mu > 0$ when $H$ is non-positive-definite, and reverts to the Newton step with a classical line-search otherwise. In this hybrid algorithm the (more expensive) curvilinear search is only employed in regions where $f(x)$ is non-convex. Numerical experiments in [2] show that this hybrid approach can be computationally more efficient. Proof of convergence of the hybrid approach is a straightforward combination of the results in the previous sections with the well-known properties of conventional Newton.

In this paper we are completely agnostic about how the linear equations $(H+\mu I)p = -g$ are solved. For small to medium-dimension problems, a popular strategy is to use the Cholesky factorization to solve the equations for $p$, and at the same time to verify that $H + \mu I$ really is positive definite. When $n$ is very large, we advocate an approach using Truncated Newton (see below). Whatever method is used to solve the linear equations, Algorithm 4.1 generates an appropriate sequence of trial points, and specifies the relevant acceptance criterion.

An objection can be made that methods using explicit second derivatives are inappropriate in practice for problems with very large numbers of independent variables. We claim that the approach analysed here is nevertheless of general value, since a closer understanding of how best to transit through regions of non-convexity when the Hessian is available may also offer ideas on how to proceed in such regions when using other methods.

However it is also worth making the point that our approach does not require an explicit decomposition, or even representation, of the full Hessian $H$: we require values only for the extreme eigenvalues $\Lambda$ and $\lambda$ and, as we have seen, even very rough estimates of them suffice.

The techniques of Automatic Differentiation [4] allow complete directional second derivatives of the form $Hs$ to be evaluated, for arbitrary $s$, at the computational cost of a few evaluations of $f$. This allows $\lambda$ and $\Lambda$ to be obtained cheaply, to adequate accuracy, by using the power method, and thus for $\mu$ to be chosen so as to ensure that $H+\mu I$ is well-conditioned. This conditioning in turn allows rapid approximation to the solution $p$ of the equation $(H + \mu I)p = -g$ using the Truncated Newton method, which requires only vector-Hessian products of the form $Hs$. Algorithm 4.1 can thus be applied at a low computational cost even when $x$ has very large dimension.

## Conflict of Interest

The authors declare that they have no conflict of interest.

## Data Sharing

Data sharing is not applicable to this article, as no datasets were generated or analysed during the current study.

# References


[1] A. C. Aitken, Studies in Practical Mathematics II: The Evaluation of Latent Roots and Latent Vectors of a Matrix. Proceedings of the Royal Society of Edinburgh, 57, 269–304, 1937.

[2] Michael Bartholomew-Biggs, Salah Beddiaf and Bruce Christianson. Further developments of methods for traversing regions of non-convexity in optimization problems. Optimization Online 8306, 2021.

[3] Michael Bartholomew-Biggs, Salah Beddiaf, and Bruce Christianson. Comparison of methods for traversing regions of non-convexity in optimization problems. Numerical Algorithms 85(2), 1–23, 2019.

[4] M. Bartholomew-Biggs, S. Brown, B. Christianson, and L. Dixon. Automatic Differentiation of algorithms. Journal of Computational and Applied Mathematics, 124 (1-2), 171–190, 2000.

[5] Salah Beddiaf. Continuous Steepest Descent Path for Traversing Non-Convex Regions. PhD thesis, University of Hertfordshire UK, 2016.

[6] W. Behrman, An Efficient Gradient Flow Method for Unconstrained Optimization, PhD Thesis, Stanford University, 1998.

[7] A.A. Brown, Optimization Methods involving the Solution of Ordinary Differential equations, PhD Thesis, Hatfield Polytechnic, 1986.

[8] Bruce Christianson. Global Convergence using De-linked Goldstein or Wolfe Linesearch Conditions. Advanced Modeling and Optimization, 11(1), 25–31, 2009.

[9] A.R. Conn, N.I.M. Gould, and P.T. Toint. Trust Region Methods. In: MPS-SIAM Series on Optimization, Philadelphia(2000).

[10] I.J.D. Craig and A.D. Sneyd, The Acceleration of Matrix Power Methods by Cyclic Variations of the Shift Parameter, Computers & Mathematics with Applications, 17(7), 1149–1159, 1989.

[11] S.M. Goldfeld, R.E. Quandt, and H.F. Trotter, Maximization by Quadratc Hill Climbing, Econometrica, 34, 541–551, 1966.

[12] M. D. Hebden, An algorithm for minimization using exact second derivatives, Atomic Energy Research Establishment report TP515, Harwell, England, 1973.

[13] D.J. Higham, Trust-region Algorithms and Time step selection, SIAM Journal on Numerical Analysis, 37(1), 194–210, 1999.

[14] Mohammadreza Samadi. Efficient Trust Region Methods for Nonconvex Optimization. PhD thesis, Lehigh University, Bethlehem Pennsylvania, 2019.