GRADIENT BASED METHODS: FUNCTIONAL VS PARAMETRIC FORMS

Tony J. Dodd, Sumitra Nair and Robert F. Harrison

Department of Automatic Control and Systems Engineering
University of Sheffield, Sheffield S1 3JD, UK
e-mail: {t.j.dodd, cop03sn, r.f.harrison}@shef.ac.uk

Abstract: Reproducing kernel Hilbert spaces (RKHS) provide a unified framework for the solution of a number of function approximation and signal estimation problems. A significant problem with RKHS methods for real applications is the poor scaling properties of the algorithms with the number of data. It is therefore often necessary to use iterative algorithms. Steepest descent and conjugate gradient solutions for approximation in RKHS are presented in this paper. Four different approaches are described and compared on a benchmark system identification problem.

Keywords: reproducing kernel, Hilbert spaces, system identification, function approximation, iterative methods, least-squares approximation, regularisation

1. INTRODUCTION

Reproducing kernel Hilbert spaces (RKHS) provide a unified framework for the solution of a number of function approximation, system identification and signal estimation problems. These include splines (Wahba 1990), support vector machines (Cristianini and Shawe-Taylor 2000), certain classes of neural networks (Poggio and Girosi 1990), finite and infinite degree Volterra series and bandlimited signal reconstruction (de Figueiredo 1983, Wan *et al.* 2003, Yao 1967).

A significant problem with RKHS methods for real applications is the linear scaling of the algorithms with the number of data which translates to a cubic scaling in terms of computation of the resulting matrix inverses. It is therefore often necessary to use iterative algorithms which can reduce the computational effort (Cristianini and Shawe-Taylor 2000). In the case of support vector machines, the natural sparsity of the solution allows for particularly efficient methods, for example the sequential minimal optimisation algorithm (Platt 1999). More generally, gradient methods provide a set of possible solutions which have been used with some success (Dodd and Harrison 2002*a*).

The main contribution of the paper is to present a detailed comparison of alternative formulations of steepest descent and conjugate gradient methods for the solution of RKHS approximation problems. In addition to a computable function-based approach three different parametric versions are described. It has been found that, whilst theoretically solving the same problem, these different approaches have significantly different numerical and convergence properties. Preliminary results on the application of these methods to a system identification problem are presented. These results provide initial guidance on which algorithms can be expected to perform best and also highlight a number of issues for further investigation.

2. PRELIMINARIES

We assume some unknown function, f, that we are able to observe at a finite number of points. f belongs to a RKHS, $\mathcal{F}$, defined on a parameter set, $\mathcal{X}$, that can be considered as an input set in the sense that, for any $x \in \mathcal{X}$, $f(x)$ represents the evaluation of f at x.

A finite set of (possibly noisy) observations, $\{z_i\}_{i=1}^N$, of the function is made corresponding to each $\{x_i\}_{i=1}^N$

$$z_i = L_i f + \epsilon_i \tag{1}$$

where $\{L_i\}_{i=1}^N$ is a set of linear evaluation functionals, defined on $\mathcal{F}$, which associate real numbers to the function, f and the ϵ_i are random noise. We can represent the set of observations $\{z_i\}_{i=1}^N$ thus

$$z = Lf + \epsilon = \sum_{i=1}^{N} (L_i f + \epsilon_i) e_i \tag{2}$$

where $e_i \in \mathbb{R}^N$ is the ith standard basis vector.

By assuming that $\mathcal{F}$ is a RKHS the L_i are continuous (hence bounded) (Aronszajn 1950). It follows from the Riesz representation theorem that we can express the evaluations as (Akhiezer and Glazman 1981)

$$L_i f = \langle f, k(x_i, \cdot)\rangle_{\mathcal{F}}, \quad i = 1, \ldots, N \tag{3}$$

where $\langle \cdot, \cdot \rangle_{\mathcal{F}}$ denotes the inner product in $\mathcal{F}$. The $\{k(x_i, \cdot)\}_{i=1}^N$ form a set of functions each belonging to $\mathcal{F}$ and uniquely determined by the functionals L_i.

The approximation problem can be formulated as follows (Bertero *et al.* 1985): given the RKHS of functions, $\mathcal{F}$, the set of functions, $\{k(x_i, \cdot)\}_{i=1}^N \subset \mathcal{F}$, and the observations, $\{z_i\}_{i=1}^N$, find a function $f \in \mathcal{F}$ such that (3) is satisfied.

The functions, $k(x_i, \cdot)$, are positive-definite and are known as the reproducing kernels of the RKHS. Further, for every $x, x' \in \mathcal{X}$ (where $k(\cdot, x')$ is the function defined on $\mathcal{X}$, with value at x in $\mathcal{X}$ equal to $k(x, x')$):

(1) $k(\cdot, x') \in \mathcal{F}$; and
(2) $\langle f, k(\cdot, x')\rangle_{\mathcal{F}} = f(x')$

for every f in $\mathcal{F}$.

We now seek the regularised solution $f_{reg} \in \mathcal{F}$ to (2) which minimises

$$g_{reg}(f) = \frac{1}{2}\|Lf - z\|^2 + \frac{\rho}{2}\|f\|^2 \tag{4}$$

where $\rho \geq 0$ is known as the regularisation parameter. The unique minimiser of (4) satisfies

$$f_{reg}(\cdot) = (\rho I + L^* L)^{-1} L^* z \tag{5}$$

or, equivalently

$$f_{reg}(\cdot) = L^* (\rho I + L L^*)^{-1} z \tag{6}$$

where I is the appropriate identity operator. To compute the prediction at some new point x we use

$$\begin{aligned} f_{reg}(x) &= \langle f_{reg}(\cdot), k(x, \cdot)\rangle \\ &= \langle L^*(\rho I + LL^*)^{-1} z, k(x, \cdot)\rangle. \end{aligned}$$

Equivalently

$$f_{reg}(x) = \langle Lk(x, \cdot), (\rho I + LL^*)^{-1} z\rangle$$

In the case of finite dimensional RKHS we have (Dodd and Harrison 2002*b*)

$$L^* c = \sum_{i=1}^{p} k(x_i, \cdot) c_i, \quad LL^* = \sum_{j=1}^{p}\sum_{i=1}^{p} k(x_i, x_j) e_j e_i^T \tag{7}$$

for any $c \in \mathbb{R}^N$. The expression for LL^* is equivalent to the kernel (Gram) matrix K where $[K]_{ij} = k(x_i, x_j)$. The expression $L^* c$ represents a general function in the finite dimensional RKHS spanned by the $k(x_i, \cdot)$.

Therefore

$$f_{reg}(x) = k^T (\rho I + K)^{-1} z \tag{8}$$

where k is the vector

$$k = Lk(x, \cdot) = [k(x_1, x), \ldots, k(x_p, x)]^T. \tag{9}$$

3. ITERATIVE METHODS: LINEAR OPERATOR EQUATIONS

We consider first the general formulation of gradient-based iterative methods for solving linear operator equations in Hilbert spaces. This general formulation will then be specialised to RKHS in subsequent sections. Consider

$$Au = b \tag{10}$$

where $A : \mathcal{U} \to \mathcal{B}$ is a bounded linear operator, $u \in \mathcal{U}$, $b \in \mathcal{B}$ and $\mathcal{U}, \mathcal{B}$ are Hilbert spaces defined on a common field, $\mathcal{X}$.

We seek the solution which minimises the regularised least squares cost function

$$J_{reg}(u) = \frac{1}{2}\|Au - b\|^2 + \frac{\rho}{2}\|u\|^2. \tag{11}$$

Now $J_{reg}(u)$ is Fréchet differentiable and we can therefore find the gradient

$$\nabla J_{reg}(u) = A^* A u - A^* b + \rho u = A^*(Au - b) + \rho u.$$

which we subsequently denote $\tilde{u}^{reg}$.

The general form of a gradient iteration is then given by

$$u_0 \in \mathbb{R}(A^*), \quad u_{n+1} = u_n - \eta_n \tilde{p}_n \tag{12}$$

where $\tilde{p}_n$ is related to the gradient $\nabla J_{reg}(u)$. We consider two specific cases:

Steepest Descent

$$\tilde{p}_n = \tilde{u}_n^{reg} \tag{13}$$

$$\eta_n = \frac{\|\tilde{p}_n\|^2}{\|A\tilde{p}_n\|^2 + \rho\|\tilde{p}_n\|^2}. \tag{14}$$

Conjugate Gradient

$$\tilde{p}_n = \tilde{u}_n^{reg} + \delta_{n-1}\tilde{p}_{n-1}, \quad \tilde{p}_0 = \tilde{u}_0^{reg} \tag{15}$$

$$\delta_{n-1} = \frac{\|\tilde{u}_n^{reg}\|^2}{\|\tilde{u}_{n-1}^{reg}\|^2} \tag{16}$$

$$\eta_n = \frac{\|\tilde{u}_n^{reg}\|^2}{\|A\tilde{p}_n\|^2 + \rho\|\tilde{p}_n\|^2}. \tag{17}$$

4. ITERATIVE METHODS: RKHS

We now specialise the methods of steepest descent and the conjugate gradient to RKHS.

4.1 Steepest Descent

4.1.1. Function Form (SDF) Setting $A = L, u = f, b = z$ we have

$$\nabla J_{reg}(f) = L^*(Lf - z) + \rho f \tag{18}$$

and, therefore,

$$f_0 \in R(L^*), \quad f_{n+1} = f_n - \eta_n[L^*(Lf_n - z) + \rho f_n].$$

Since $f_n = L^* c_n$ we can re-write this as

$$f_{n+1} = L^* c_n - \eta_n[L^*(LL^* c_n - z) + \rho L^* c_n]$$

and letting

$$c_{n+1} = c_n - \eta_n[(LL^* c_n - z) + \rho c_n]$$

we have $f_{n+1} = L^* c_{n+1}$. Computationally

$$c_0 \in \mathbb{R}^N, \quad c_{n+1} = c_n - \eta_n[(Kc_n - z) + \rho c_n]. \tag{19}$$

Denoting $\tilde{f}_n^{reg} = \nabla J_{reg}(f_n)$,

$$\eta_n = \frac{\|\tilde{f}_n^{reg}\|^2}{\|L\tilde{f}_n^{reg}\|^2 + \rho\|\tilde{f}_n^{reg}\|^2}.$$

Therefore, using the fact that $f_n = L^* c_n$ and defining $\bar{c}_n = LL^* c_n - z + \rho c_n$,

$$\eta_n = \frac{\|L^*\bar{c}_n\|^2}{\|LL^*\bar{c}_n\|^2 + \rho\|L^*\bar{c}_n\|^2} = \frac{\langle LL^*\bar{c}_n, \bar{c}_n\rangle}{\|LL^*\bar{c}_n\|^2 + \rho\langle LL^*\bar{c}_n, \bar{c}_n\rangle}.$$

This can be written in terms of $K = LL^*$ as

$$\eta_n = \frac{\bar{c}_n^T K \bar{c}_n}{\bar{c}_n^T K^2 \bar{c}_n + \rho \bar{c}_n^T K \bar{c}_n} \tag{20}$$

where, in practice, $\bar{c}_n = Kc_n - z + \rho c_n$.

4.1.2. Parameter Form I (SDPI) Using $A = LL^*, u = c, b = z$ we have

$$\nabla J_{reg}(c) = LL^*(LL^* c - z) + \rho c. \tag{21}$$

The general iteration is therefore

$$c_0 \in \mathbb{R}^N, \quad c_{n+1} = c_n - \eta_n[LL^*(LL^* c_n - z) + \rho c_n] \tag{22}$$

which can be written in terms of $K = LL^*$ as

$$c_{n+1} = c_n - \eta_n[K(Kc_n - z) + \rho c_n]. \tag{23}$$

Now

$$\eta_n = \frac{\|\tilde{c}_n^{reg}\|^2}{\|LL^*\tilde{c}_n^{reg}\|^2 + \rho\|\tilde{c}_n^{reg}\|^2}$$

where $\tilde{c}_n^{reg} = LL^*(LL^* c_n - z) + \rho c_n$, which we subsequently denote by $\check{c}_n$. Therefore

$$\eta_n = \frac{\check{c}_n^T \check{c}_n}{\check{c}_n^T K^2 \check{c}_n + \rho \check{c}_n^T \check{c}_n}. \tag{24}$$

Computationally we use $\check{c}_n = K(Kc_n - z) + \rho c_n$.

4.1.3. Parameter Form II (SDPII) The previous parametric form does not exactly correspond to the functional form as it utilises a different cost functional. We have actually solved

$$J_{reg}(c) = \frac{1}{2}\|LL^* c - z\|^2 + \frac{\rho}{2}\|c\|^2 \tag{25}$$

whereas we are really interested in

$$J_{reg}(c) = \frac{1}{2}\|LL^* c - z\|^2 + \frac{\rho}{2}\|L^* c\|^2. \tag{26}$$

In the former the regulariser is proportional to $\|c\|^2$ whereas it should in fact be proportional to $\|L^* c\|^2 = \|f\|^2$.

Rearranging (26)

$$J_{reg}(c) = \frac{1}{2}\|LL^* c - z\|^2 + \frac{\rho}{2}\langle c, LL^* c\rangle. \tag{27}$$

The gradient is now given by

$$\nabla J_{reg}(c) = LL^* LL^* c - LL^* z + \rho LL^* c. \tag{28}$$

The steepest descent iteration is

$$c_0 \in \mathbb{R}^N, \quad c_{n+1} = c_n - \eta_n LL^*(LL^* c_n - z + \rho c_n)$$

where

$$\eta_n = \frac{\|\tilde{c}_n^{reg}\|^2}{\|LL^*\tilde{c}_n^{reg}\|^2 + \rho\|L^*\tilde{c}_n^{reg}\|^2} \tag{29}$$

which makes use of the definition $\tilde{c}_n^{reg} = \nabla J_{reg}(c_n)$.

Computationally we have

$$c_{n+1} = c_n - \eta_n K(Kc_n - z + \rho c_n). \tag{30}$$

Writing (29) in terms of $\bar{c}_n$

$$\eta_n = \frac{\|LL^*\bar{c}_n\|^2}{\|LL^* LL^*\bar{c}_n\|^2 + \rho\|L^* LL^*\bar{c}_n\|^2}$$

or

$$\eta_n = \frac{\|LL^*\bar{c}_n\|^2}{\|LL^* LL^*\bar{c}_n\|^2 + \rho\langle LL^* LL^*\bar{c}_n, LL^*\bar{c}_n\rangle}.$$

In terms of $K = LL^*$

$$\eta_n = \frac{\bar{c}_n^T K^2 \bar{c}_n}{\bar{c}_n^T K^4 \bar{c}_n + \rho \bar{c}_n^T K^3 \bar{c}_n}. \tag{31}$$

4.2 Conjugate Gradient

4.2.1. Function Form (CGF) Setting $A = L$, $u = f$, $b = z$, the iteration is given by

$$f_0 \in R(L^*), \quad f_{n+1} = f_n - \eta_n \tilde{p}_n \tag{32}$$

where

$$\tilde{p}_0 = \tilde{f}_0^{reg}, \quad \tilde{p}_n = \tilde{f}_n^{reg} + \delta_{n-1}\tilde{p}_{n-1} \tag{33}$$

and

$$\tilde{f}_n^{reg} = L^*(Lf_n - z) + \rho f_n. \tag{34}$$

The associated learning rates are

$$\delta_{n-1} = \frac{\|\tilde{f}_n^{reg}\|^2}{\|\tilde{f}_{n-1}^{reg}\|^2}, \quad \eta_n = \frac{\|\tilde{f}_n^{reg}\|^2}{\|L\tilde{p}_n\|^2 + \rho\|\tilde{p}_n\|^2}. \tag{35}$$

Now let

$$f_n = L^* c_n, \quad \tilde{p}_{n-1} = L^* b_{n-1}. \tag{36}$$

Then

$$\tilde{p}_n = L^*(LL^*c_n - z) + \rho L^* c_n + \delta_{n-1} L^* b_{n-1}. \tag{37}$$

Letting

$$b_n = LL^* c_n - z + \rho c_n + \delta_{n-1} b_{n-1} \tag{38}$$

then

$$\tilde{p}_n = L^* b_n. \tag{39}$$

Note that, in practice, we have

$$b_0 = \bar{c}_0, \quad b_n = \bar{c}_n + \delta_{n-1} b_{n-1}. \tag{40}$$

Further, if we define

$$c_0 \in \mathbb{R}^N, \quad c_{n+1} = c_n - \eta_n b_n \tag{41}$$

then

$$f_{n+1} = L^* c_{n+1}. \tag{42}$$

Now

$$\delta_{n-1} = \frac{\|L^*(Lf_n - z) + \rho f_n\|^2}{\|L^*(Lf_{n-1} - z) + \rho f_{n-1}\|^2}. \tag{43}$$

In terms of c_n this becomes

$$\delta_{n-1} = \frac{\|L^*\bar{c}_n\|^2}{\|L^*\bar{c}_{n-1}\|^2} = \frac{\langle LL^*\bar{c}_n, \bar{c}_n\rangle}{\langle LL^*\bar{c}_{n-1}, \bar{c}_{n-1}\rangle}$$

which is equivalent to

$$\delta_{n-1} = \frac{\bar{c}_n^T K \bar{c}_n}{\bar{c}_{n-1}^T K \bar{c}_{n-1}}. \tag{44}$$

We also have

$$\eta_n = \frac{\|L^*(Lf_n - z) + \rho f_n\|^2}{\|L(\tilde{f}_n^{reg} + \delta_{n-1}\tilde{p}_{n-1})\|^2 + \rho\|\tilde{f}_n^{reg} + \delta_{n-1}\tilde{p}_{n-1}\|^2}.$$

Writing this in terms of $\bar{c}_n$ and b_n

$$\eta_n = \frac{\|L^*\bar{c}_n\|^2}{\|LL^*b_n\|^2 + \rho\|L^*b_n\|^2}. \tag{45}$$

or

$$\eta_n = \frac{\bar{c}_n^T K \bar{c}_n}{b_n^T K^2 b_n + \rho b_n^T K b_n}. \tag{46}$$

4.2.2. Parameter Form I (CGPI) We use $A = LL^*, x = c, b = z$ and the basic iteration is given by

$$c_0 \in \mathbb{R}^N, \quad c_{n+1} = c_n - \eta_n \tilde{p}_n \tag{47}$$

where

$$\tilde{p}_n = \tilde{c}_n^{reg} + \delta_{n-1}\tilde{p}_{n-1} \tag{48}$$

and

$$\tilde{c}_n^{reg} = LL^*(LL^*c_n - z) + \rho c_n. \tag{49}$$

Defining $b_n = \tilde{p}_n$ we have

$$c_{n+1} = c_n - \eta_n b_n \tag{50}$$

and

$$\begin{aligned} b_n &= LL^*(LL^*c_n - z) + \rho c_n + \delta_{n-1} b_{n-1} && (51)\\ &= K(Kc_n - z) + \rho c_n + \delta_{n-1} b_{n-1}. && (52) \end{aligned}$$

Also

$$\delta_{n-1} = \frac{\|\tilde{c}_n^{reg}\|^2}{\|\tilde{c}_{n-1}^{reg}\|^2} = \frac{\check{c}_n^T \check{c}_n}{\check{c}_{n-1}^T \check{c}_{n-1}}. \tag{53}$$

For η_n,

$$\eta_n = \frac{\|\tilde{c}_n^{reg}\|^2}{\|LL^*\tilde{p}_n\|^2 + \rho\|\tilde{p}_n\|^2} = \frac{\bar{c}_n^T \bar{c}_n}{b_n^T K^2 b_n + \rho b_n^T b_n}. \tag{54}$$

4.2.3. Parameter Form II (CGPII) Again, using the modified loss function, (26), the iteration is now given by

$$c_0 \in \mathbb{R}^N, \quad c_{n+1} = c_n - \eta_n \tilde{p}_n \tag{55}$$

where

$$\tilde{p}_n = \tilde{c}_n^{reg} + \delta_{n-1}\tilde{p}_{n-1} \tag{56}$$

and, in this case,

$$\tilde{c}_n^{reg} = LL^*(LL^*c_n - z + \rho c_n). \tag{57}$$

As usual we use $b_n = \tilde{p}_n$ and therefore

$$c_{n+1} = c_n - \eta_n b_n \tag{58}$$

and

$$\begin{aligned} b_n &= LL^*(LL^*c_n - z + \rho c_n) + \delta_{n-1} b_{n-1} && (59)\\ &= K(Kc_n - z + \rho c_n) + \delta_{n-1} b_{n-1}. && (60) \end{aligned}$$

Now

$$\delta_{n-1} = \frac{\|\tilde{c}_n^{reg}\|^2}{\|\tilde{c}_{n-1}^{reg}\|^2} = \frac{\|LL^*(LL^*c_n - z + \rho c_n)\|^2}{\|LL^*(LL^*c_{n-1} - z + \rho c_{n-1})\|^2}.$$

In terms of K this becomes

$$\delta_{n-1} = \frac{\bar{c}_n^T K^2 \bar{c}_n^T}{\bar{c}_{n-1}^T K^2 \bar{c}_{n-1}^T} \tag{61}$$

For η_n we have the following

$$\eta_n = \frac{\|\tilde{c}_n^{reg}\|^2}{\|LL^*\tilde{p}_n\|^2 + \rho\|L^*\tilde{p}_n\|^2} = \frac{\bar{c}_n^T K^2 \bar{c}_n}{b_n^T K^2 b_n + \rho b_n^T K b_n}. \tag{62}$$

5. SELF-ADJOINT, POSITIVE DEFINITE A

Consider now the case where A is self-adjoint and postive definite. We can then minimise

$$J_{reg'}(u) = \frac{1}{2}\langle Au, u\rangle - \langle u, b\rangle + \frac{\rho}{2}\|u\|^2. \quad (63)$$

The gradient is then given by

$$\nabla J_{reg'}(u) = Au - b + \rho u. \quad (64)$$

We define $\tilde{u}^{reg'} = \nabla J_{reg'}(u)$ and the general iteration is given by

$$u_0 \text{ arbitrary }, \quad u_{n+1} = u_n - \eta_n' \tilde{p}_n'. \quad (65)$$

Steepest Descent

$$\tilde{p}_n' = \tilde{u}^{reg'} \quad (66)$$

$$\eta_n' = \frac{\|\tilde{p}_n'\|^2}{\langle A\tilde{p}_n', \tilde{p}_n'\rangle + \rho\|\tilde{p}_n'\|^2} \quad (67)$$

Conjugate Gradient

$$\tilde{p}_n' = \tilde{u}_n^{reg'} + \delta_{n-1}\tilde{p}_{n-1}', \quad \tilde{p}_0' = \tilde{u}_0^{reg'} \quad (68)$$

$$\delta_{n-1} = \frac{\|\tilde{u}_n^{reg'}\|^2}{\|\tilde{u}_{n-1}^{reg'}\|^2} \quad \eta_n' = \frac{\|\tilde{u}_n^{reg'}\|^2}{\langle A\tilde{p}_n', \tilde{p}_n'\rangle + \rho\|\tilde{p}_n'\|^2}. \quad (69)$$

5.1 Steepest Descent (SDPIII)

We are restricted to the parametric case with $A = LL^*, u = c, b = z$ and therefore

$$\tilde{p}_n' = \nabla J_{reg'}(c_n) = LL^* - z + \rho c_n. \quad (70)$$

The general iteration is given by

$$c_0 \in \mathbb{R}^N, \quad c_{n+1} = c_n - \eta_n'(LL^*c_n - z + \rho c_n) \quad (71)$$

which can be computed as

$$c_{n+1} = c_n - \eta_n'(Kc_n - z + \rho c_n). \quad (72)$$

In the steepest descent case

$$\eta_n' = \frac{\|\tilde{c}_n^{reg'}\|^2}{\langle LL^*\tilde{c}_n^{reg'}, \tilde{c}_n^{reg'}\rangle + \rho\|\tilde{c}_n^{reg'}\|^2} \quad (73)$$

where $\tilde{c}_n^{reg'} = Kc_n - z + \rho c_n$. We then have

$$\eta_n' = \frac{\bar{c}_n^T \bar{c}_n}{\bar{c}_n^T K \bar{c}_n + \rho \bar{c}_n^T \bar{c}_n}. \quad (74)$$

5.2 Conjugate Gradient (CGPIII)

The general iteration is given by

$$c_0 \in \mathbb{R}^N, \quad c_{n+1} = c_n - \eta_n' \tilde{p}_n' \quad (75)$$

where

$$\tilde{p}_n' = \tilde{c}_n^{reg'} + \delta_{n-1}\tilde{p}_{n-1}' \quad (76)$$
$$= LL^*c_n - z + \rho c_n + \delta_{n-1}\tilde{p}_{n-1}'. \quad (77)$$

$$\eta_n' = \frac{\|\tilde{c}_n^{reg'}\|^2}{\langle LL^*\tilde{p}_{n-1}', \tilde{p}_{n-1}'\rangle + \rho\|\tilde{p}_{n-1}'\|^2}. \quad (78)$$

Defining $b_n = \tilde{p}_n'$

$$c_{n+1} = c_n - \eta_n' b_n \quad (79)$$

where

$$b_n = LL^*c_n - z + \rho c_n + \delta_{n-1}b_{n-1} \quad (80)$$
$$= Kc_n - z + \rho c_n + \delta_{n-1}b_{n-1}. \quad (81)$$

Now

$$\delta_{n-1} = \frac{\|\tilde{c}_n^{reg'}\|^2}{\|\tilde{c}_{n-1}^{reg'}\|^2} = \frac{\bar{c}_n^T \bar{c}_n}{\bar{c}_{n-1}^T \bar{c}_{n-1}} \quad (82)$$

Also

$$\eta_n' = \frac{\|\bar{c}_n\|^2}{\langle LL^*b_n, b_n\rangle + \rho\|b_n\|^2} = \frac{\bar{c}_n^T \bar{c}_n}{b_n^T K b_n + \rho b_n^T b_n}. \quad (83)$$

6. RESULTS

We now investigate, empirically, the convergence properties and numerical sensitivity of the gradient methods described above. For brevity we restrict our attention to the conjugate gradient methods. For these we expect, theoretically, convergence in, at most, N iterations. The following nonlinear dynamical system was simulated in Matlab

$$z(t) = 0.5y(t-1) + 0.3y(t-1)u(t-1) + 0.2u(t-1)$$
$$+0.05y^2(t-1) + 0.6u^2(t-1) + \epsilon(t)$$

where $\epsilon(t) \sim N(0, 0.001)$, $y(0) = 0.1$ and $u(t) \sim N(0.2, 0.1)$. Results of modelling the system with a Gaussian kernel ($k(x, x') = \exp(-\beta\|x - x'\|)$), based on 25 data points and averaged over 50 realisations of the data are shown in Figures 1, 2.

Theoretically, the method of conjugate gradients for each of the cases above should converge in, at most, N iterations, i.e. the norm difference between the true and iterated parameters should be zero. However, round-off errors make this impractical and can reduce the rate of convergence. In general, all the algorithms converged to an acceptable error within N iterations with the exception of CGPII, for which the error was two orders of magnitude greater. In many cases CGPII converged only after a significantly higher number of iterations - in Figure 1 even after 500 iterations the error has only just reduced to the level after 25 iterations for the other algorithms.

The actual rate of convergence was found to degrade with increasing regularisation for all algorithms, although most notably for PII. This is

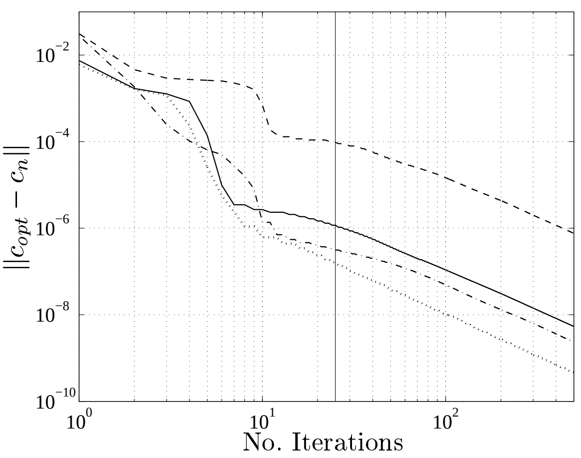

Fig. 1. Norm difference between true, c_{opt}, and iterated, c_n, parameters over 500 iterations for $\rho = 0.1$, $\beta = 100$, $N = 25$. Shown are CGF ('–'), CGPI ('$- \cdot -$'), CGPII ('- -'), CGPIII ('$\cdots$'). The vertical line indicates 25 iterations.

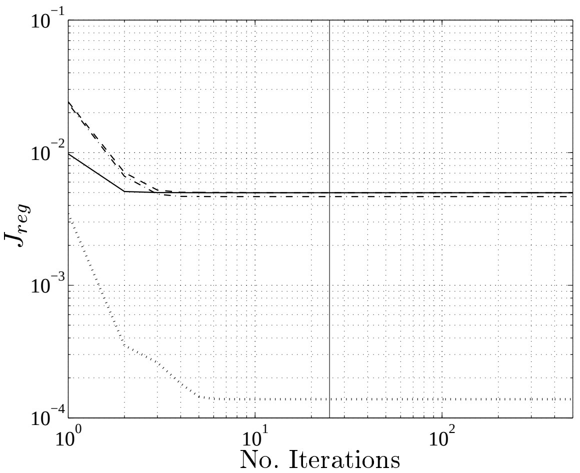

Fig. 2. Cost function values corresponding to Figure 1.

counter to the normal view that regularisation improves numerical sensitivity. The authors believe this is probably due to the multiplication of round-off errors by the regularisation parameter in the denominator of the learning rate equations.

In contrast, the values of the associated cost functions all converged well within N iterations, Figure 2. This result was not affected by the amount of regularisation. In terms of the error in the parameters, algorithm CGPIII tends to converge earliest and to a lower norm error. However, if early stopping is to be used then algorithm CGPI may be preferred as it achieves better errors for iterations 2-5. Computationally, CGPIII is also the most efficient algorithm and CGPII the least.

These results are preliminary and a more detailed investigation is currently underway. The numerical sensitivity of the algorithms will be compared using different machine precisions. Convergence will be assessed on a number of problems with varying amounts of regularisation and using Monte Carlo simulations to assess the variability of the results. The theoretical convergence rates of the different algorithms will also be studied to provide further guidance on the expected convergence rates of the algorithms. In particular these will be compared with the computational overhead in terms of floating point operations.

REFERENCES

Akhiezer, N.I. and I.M. Glazman (1981). *Theory of Linear Operators in Hilbert Space*. Vol. I. Pitman.

Aronszajn, N. (1950). Theory of reproducing kernels. *Transactions of the American Mathematical Society* **68**, 337–404.

Bertero, M., C. De Mol and E.R. Pike (1985). Linear inverse problems with discrete data. I: General formulation and singular system analysis. *Inverse Problems* **1**, 301–330.

Cristianini, N. and J. Shawe-Taylor (2000). *An Introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press.

de Figueiredo, Rui J.P. (1983). A generalized Fock space framework for nonlinear system and signal analysis. *IEEE Transactions on Circuits and Systems* **CAS-30**, 637–647.

Dodd, T.J. and R.F. Harrison (2002*a*). Iterative solution to approximation in reproducing kernel Hilbert spaces. In: *CD-ROM Proceedings of the 15th IFAC World Congress*.

Dodd, T.J. and R.F. Harrison (2002*b*). Some lemmas on reproducing kernel Hilbert spaces. Technical Report 819. Department of Automatic Control and Systems Engineering, University of Sheffield, UK.

Platt, J. (1999). Fast training of support vector machines using sequential minimal optimization. In: *Advances in Kernel Methods - Support Vector Learning* (B. Schölkopf, C.J.C. Burges and A.J. Smola, Eds.). MIT Press. pp. 185–208.

Poggio, T. and F. Girosi (1990). Networks for approximation and learning. *Proceedings of the IEEE* **78**(9), 1481–1497.

Wahba, G. (1990). *Spline Models for Observational Data*. Vol. 50 of *Series in Applied Mathematics*. SIAM. Philadelphia.

Wan, Y., T.J. Dodd and R.F. Harrison (2003). Infinite degree Volterra series estimation. In: *Proceedings of The 2nd International Conference on Computational Intelligence, Robotics and Autonomous Systems, Singapore*.

Yao, K. (1967). Applications of reproducing kernel Hilbert spaces - bandlimited signal models. *Information and Control* **11**, 429–444.