Centre Européen de Recherche et de Formation Avancée en Calcul Scientifique

Preconditioning and Globalizing Conjugate Gradients in Dual Space for Quadratically Penalized Nonlinear-Least Squares Problems

Serge Gratton, Selime Gürol, Philippe Toint

Technical Report TR/PA/12/136

Publications of the Parallel Algorithms Team

http://www.cerfacs.fr/algor/publications/

Preconditioning and Globalizing Conjugate Gradients in Dual Space for Quadratically Penalized Nonlinear-Least Squares Problems

Serge Gratton[*], Selime Gürol[†] and Philippe Toint[‡]

10 November 2010

Abstract

When solving nonlinear least-squares problems, it is often useful to regularize the problem using a quadratic term, a practice which is especially common in applications arising in inverse calculations. A solution method derived from a trust-region Gauss-Newton algorithm is analyzed for such applications, where, contrary to the standard algorithm, the least-squares subproblem solved at each iteration of the method is rewritten as a quadratic minimization subject to linear equality constraints. This allows the exploitation of duality properties of the associated linearized problems. This paper considers a recent conjugate-gradient-like method which performs the quadratic minimization in the dual space and produces, in exact arithmetic, the same iterates as those produced by a standard conjugate-gradients method in the primal space. This dual algorithm is computationally interesting whenever the dimension of the dual space is significantly smaller than that of the primal space, yielding gains in terms of both memory usage and computational cost. The relation between this dual space solver and PSAS (Physical-space Statistical Analysis System), another well-known dual space technique used in data assimilation problems, is explained. The use of an effective preconditioning technique is proposed and refined convergence bounds derived, which results in a practical solution method. Finally, stopping rules adequate for a trust-region solver are proposed in the dual space, providing iterates that are equivalent to those obtained with a Steihaug-Toint truncated conjugate-gradient method in the primal space.

Keywords: Data assimilation, dual-space minimization, preconditioning, conjugate-gradient methods, globalization, trust-region methods

1 Introduction

This paper investigates conjugate-gradients (CG) methods for the solution of under-determined nonlinear least-squares problems regularized by a quadratic penalty term. Such problems often result from a maximum likelihood approach, and involve a set of m physical observations and n

[*] ENSEEIHT, 2, rue Camichel, 3100 Toulouse, France. Email: serge.gratton@enseeiht.fr

[†] CERFACS, 42, avenue Coriolis, 31057 Toulouse, France. Email: selime.gurol@cerfacs.fr

[‡] Namur Centre for Complex Systems (NAXYS), FUNDP-University of Namur, 61, rue de Bruxelles, B-5000 Namur, Belgium. Email: philippe.toint@fundp.ac.be

unknowns which are estimated by a nonlinear regression. We suppose here that n is large compared to m. These problems are encountered for instance when tri-dimensional fields are reconstructed using physical observations, as is the case in data assimilation in Earth observation systems (Weaver, Vialard and Anderson 2003). In meteorological applications for example, the result of this minimization procedure is the initial state of a dynamical system, which is then integrated forward in time to produce a weather forecast.

A widely used algorithm in this context is the truncated Gauss-Newton (TGN) method, known in the Earth observation community under the name of *incremental four dimensional variational data assimilation* (Incremental 4D-Var) (Courtier, Thépaut and Hollingsworth 1994). The TGN method generates a sequence of iterates by solving linear least squares problems (Gratton, Lawless and Nichols 2007). However, it is well-known that this simple variant of the Gauss-Newton algorithm does not ensure a monotonic decrease of the objective function and that convergence is not guaranteed. Removing this difficulty is typically achieved by using a linesearch (Dennis and Schnabel 1983) or a trust-region (Conn, Gould and Toint 2000) strategy, which ensures global convergence to first order critical points under mild assumptions. We consider the second of these approaches in this paper. Moreover, taking into consideration the large-scale nature of the problem, we propose here to use a particular trust-region algorithm relying on the Steihaug-Toint truncated conjugate-gradients method for the approximate solution of the subproblem (Conn et al. 2000, pp. 133-139).

Solving this subproblem in the n-dimensional space (by CG) is referred to as *the primal approach*. Alternatively, a significant reduction in the computational cost is possible by rewriting the quadratic approximation in the m-dimensional space related to the observations. This is crucial for the large-scale applications such as those solved daily in weather prediction systems, where typically $n \sim 10^7$ and $m \sim 10^5$ (Bouttier and Courtier 1999). This approach, which performs minimization in the m-dimensional space using CG or variants thereof, is referred to as *the dual approach* for reasons that will appear clearly in Section 2.

The first proposed dual approach (Courtier 1997), known as the Physical-space Statistical Analysis System (PSAS) in the data assimilation community, starts by solving the corresponding dual objective in $\mathbb{R}^m$ by a standard preconditioned CG (PCG), and then recovers the step in $\mathbb{R}^n$ using a simple multiplication with an $n \times m$ matrix. Technically, the algorithm consists in recurrence formulas involving m-vectors instead of n-vectors. However, the use of PSAS can be unduly costly as it was noticed (Akkraoui, Gauthier, Pellerin and Buis 2008) that the linear least-squares cost function is not monotonically decreasing along the nonlinear iterations when applying standard termination criteria, and therefore that very conservative such criteria have to be used, resulting in many inner iterations.

Another dual approach has been proposed by (Gratton and Tshimanga 2009) and is known as the Restricted Preconditioned Conjugate Gradient (RPCG) method. It generates the same iterates in exact arithmetic as those generated by the primal approach, again using recursion formula involving m-vectors. The main interest of RPCG and PSAS is a reduction of both memory and computational costs per iteration. The convergence of RPCG is however the same as the primal approach (the two methods are algebraically equivalent) whereas PSAS may be erratic (Gratton and Tshimanga 2009). Unfortunately, the relation between these two dual approaches and the question of deriving efficient preconditioners – essential as soon large-scale problems are considered – was not addressed in (Gratton and Tshimanga 2009).

The main motivation for this paper is to address these open issues. In particular, we are interested in designing preconditioning techniques and a trust-region globalization which maintain the one-to-one correspondance between primal and dual iterates, thereby offering cost-effective computation in a globally convergent algorithm.

The outline of the paper is as follows. In Section 2, we present the dual approaches in a general framework and explore the connections between the PSAS and RPCG solvers. We also introduce practical preconditioners to accelerate the convergence of the latter by taking into account the fact that a sequence of slowly varying linear least-squares problems are solved in the Gauss-Newton process near convergence. In particular, a warm-start preconditioner derived from limited memory quasi-Newton updating formulas (Morales and Nocedal 2000) is proposed in the primal space, and its variational properties are recalled. A dual space counterpart to this preconditioner is then derived and its variational properties analyzed. An extension of the Steihaug-Toint truncated conjugate-gradient method to the dual space is then presented in Section 3. Finally, conclusions are drawn in Section 4, and perspectives are indicated.

2 Conjugate Gradients in Dual Space

2.1 Problem Formulation

The nonlinear least-squares problem arising in data assimilation problems is typically formulated as

$$\min_{x_0} \frac{1}{2}\|x_0 - x_b\|^2_{B^{-1}} + \frac{1}{2}\sum_{j=0}^{N_t} \|\mathcal{H}_j(x(t_j)) - y_j\|^2_{R_j^{-1}}, \tag{2.1}$$

where, for a symmetric positive definite matrix M, $\|x\|^2_M = x^T M x$, $x_0 = x(t_0)$ and x_b are n-dimensional vectors representing the initial state of the model at time t_0 and the background vector (an *a priori* information obtained from previous forecasts), respectively. The vector y_j is an m_j-dimensional vector of observations and $\mathcal{H}_j$ is the operator modeling the observation process. The matrices B and R_j are respectively $n\times n$ and $m_j \times m_j$ symmetric positive definite covariance matrices corresponding to the background and observation errors. The state $x(t_j)$ at time t_j is obtained by integrating an application-specific dynamical system. Therefore, the objective function (2.1) represents a trade-off between *a priori* background information and a misfit between predicted and observed quantities. This approach is motivated by statistical theory (Tarantola 1987, pp. 24-32) and corresponds to a maximum likelihood approach under a Gaussian assumption.

We consider a TGN algorithm for solving (2.1), where the nonlinear observation operator $\mathcal{H}_j(x(t_j))$ is linearized at step k in the neighbourhood of $x^k(t_j)$. At iteration k of this approach, a step δx_0^k from x_0^k is computed by minimizing the quadratic cost function which is the linearized least squares approximation to the nonlinear problem (2.1) (Gratton et al. 2007). This quadratic cost function is formulated as

$$\min_{\delta x_0^k} \frac{1}{2}\|x_0^k - x_b + \delta x_0^k\|^2_{B^{-1}} + \frac{1}{2}\|H^k \delta x_0^k - d^k\|^2_{R^{-1}}, \tag{2.2}$$

where H^k is a $m \times n$ matrix denoting the model (linearized at x_0^k) concatenated over time with $m = \sum_{j=0}^{N_t} m_j$, R is a $m \times m$ matrix concatenated over time of R_j and where d^k denotes the concatenated misfits of d_j over time with $d_j = y_j - \mathcal{H}_j(x^k(t_j))$. The initial state estimation is then updated according to

$$x_0^{k+1} = x_0^k + \delta x_0^k.$$

The main loop of TGN, which ranges over successive iterates $\{x_0^k\}$, is called the "outer-loop" (or nonlinear iteration), while the loop which is implemented whenever an iterative solver is used for the subproblem (2.2) is called the "inner-loop" (or linear iteration). The method we have just described is locally convergent on problems where the residual norm at the solution is small enough compared to the curvature of the nonlinear function [pp. 342-346](Björck 1996). As we mentioned earlier, this method can be made globally convergent by the introduction of a trust-region mechanism. However, we postpone the discussion of this feature to Section 3 for the sake of simplicity, and restrict our attention for now on the definition of the subproblem given by (2.2).

From the optimality condition (Nocedal and Wright 1999, pp. 14-17), the solution of the subproblem (2.2) is given by

$$\delta x_0 = x_b - x_0 + (B^{-1} + H^T R^{-1} H)^{-1} H^T R^{-1}(d - H(x_b - x_0)), \tag{2.3}$$

where we have dropped the outer-loop index k for simplicity. We now reformulate (2.2) as a convex quadratic problem with linear equality constraints given by

$$\min_{\delta x_0, v} \frac{1}{2}\|x_0 - x_b + \delta x_0\|^2_{B^{-1}} + \frac{1}{2}\|v\|^2_{R^{-1}} \tag{2.4}$$

subject to

$$v = H\delta x_0 - d,$$

which can, in turn, be solved using duality theory. The dual objective for (2.4) (Nocedal and Wright 1999, p. 349) is given by

$$q(\lambda) = \inf_{\delta x_0, v} \mathcal{L}(\delta x_0, v, \lambda) \overset{\text{def}}{=} \inf_{\delta x_0, v} \frac{1}{2}\|x_0 - x_b + \delta x_0\|^2_{B^{-1}} + \frac{1}{2}\|v\|^2_{R^{-1}} - \lambda^T(H\delta x_0 - v - d), \tag{2.5}$$

where λ is a Lagrange multiplier. The infimum is achieved when

$$\nabla_{\delta x_0}\mathcal{L}(\delta x_0, v, \lambda) = B^{-1}(x_0 + \delta x_0 - x_b) - H^T\lambda = 0 \tag{2.6}$$

$$\nabla_v \mathcal{L}(\delta x_0, v, \lambda) = R^{-1}v + \lambda = 0 \tag{2.7}$$

which yields that

$$\delta x_0 = x_b - x_0 + BH^T\lambda \tag{2.8}$$

$$v = -R\lambda. \tag{2.9}$$

We may therefore substitute δx_0 and v in the expression (2.5) and obtain the dual objective explicitly as follows:

$$q(\lambda) = -\frac{1}{2}\lambda^T(HBH^T + R)\lambda + \lambda^T(d - H(x_b - x_0)), \tag{2.10}$$

which is maximized for

$$\lambda = (HBH^T + R)^{-1}(d - H(x_b - x_0)). \tag{2.11}$$

Therefore, from (2.11) and (2.8), the solution of the subproblem (2.2) may also be written as

$$\delta x_0 = x_b - x_0 + BH^T(HBH^T + R)^{-1}(d - H(x_b - x_0)). \tag{2.12}$$

Note that this solution may be obtained directly from the solution (2.3) by using the Sherman-Morrison-Woodbury formula (Nocedal and Wright 1999, pp. 612-613), as done in the original derivation of the PSAS algorithm.

In the context of interest in this paper, the matrices H, B and R are so large that they can not be stored explicitly, and the information they contain is only available through matrix-vector products. Similarly, the matrix-vector products by H^T are assumed to be performed by using the adjoint computation (Giering and Kaminski 1998). It is therefore natural to solve for the linear system in (2.3) using conjugate-gradients, whose sole access to the system matrix is via such products. We therefore apply this method to the system

$$(B^{-1} + H^T R^{-1} H)\delta v_0 = H^T R^{-1}(d - H(x_b - x_0)), \tag{2.13}$$

where $B^{-1} + H^T R^{-1} H$ matrix is symmetric and positive definite and find δx_0 from the relation

$$\delta x_0 = x_b - x_0 + \delta v_0. \tag{2.14}$$

2.2 Solving the linearized problem

We now consider the PCG algorithm for solving the system (2.13) with a preconditioner which is an approximation of the matrix A^{-1} where $A = B^{-1} + H^T R^{-1} H$. For this algorithm we examine a stopping criterion based on the energy norm of error within the standard unpreconditioned CG. The energy norm of error in primal space can be written as

$$\|\delta v_0 - \delta v_i\|_A \leq \eta \|\delta v_0\|_A, \tag{2.15}$$

where δv_i is the ith inner loop solution and $\eta < 1$. Equivalently, in terms of dual norm (Arioli 2004), as

$$\|r_i\|_{A^{-1}} \leq \eta \|r_0\|_{A^{-1}}, \tag{2.16}$$

where r_i is the residual of the linear system (2.13). Since the preconditioner P is an approximation of A^{-1}, (2.16) can be approximated by the condition

$$\|r_i\|_P \leq \eta \|r_0\|_P. \tag{2.17}$$

Note that $\|r_i\|_P$ contrary to $\|r_i\|_{A^{-1}}$ may not decrease along the iterations especially when P is a poor approximation of A^{-1}. Including this termination test in the standard PCG gives Algorithm 2.1 on the following page.

An iterative technique can also be applied in the dual approach, by applying CG in (2.11), which yields the linear system

$$(HBH^T + R)\lambda = d - H(x_b - x_0), \tag{2.18}$$

and then using the expression (2.8) to recover δx_0.

A first alternative for solving the system (2.18) is the PSAS method, which uses PCG with the canonical inner product in $\mathbb{R}^m$ and R^{-1} as a preconditioner. This gives the preconditioned system

$$R^{-1/2}(HBH^T + R)R^{-1/2}(R^{1/2}\lambda) = R^{-1/2}(d - H(x_b - x_0)). \tag{2.19}$$

It is known (Akkraoui et al. 2008), (Gratton and Tshimanga 2009) that the PSAS algorithm produces iterates in the space of Lagrange multipliers, and that their corresponding primal-space

Algorithm 2.1: PCG Algorithm in R^n

```
$\delta v_0 = 0$;
$r_0 = H^T R^{-1}(d - H(x_b - x_0))$;
$z_0 = P r_0$;
$p_0 = z_0$;
for $i = 0, 1, ...$ do
    $q_i = (B^{-1} + H^T R^{-1} H) p_i$;
    $\alpha_i = < r_i, z_i > / < q_i, p_i >$;
    $\delta v_{i+1} = \delta v_i + \alpha_i p_i$;
    $r_{i+1} = r_i - \alpha_i q_i$;
    $z_{i+1} = P r_{i+1}$;
    if $r_{i+1}^T z_{i+1} \leq \eta r_0^T z_0$ then
        exit from the loop
    end
    $\beta_i = < r_{i+1}, z_{i+1} > / < r_i, z_i >$;
    $p_{i+1} = z_{i+1} + \beta_i p_i$;
end
$\delta x_0 = x_b - x_0 + \delta v_{i+1}$;
```

counterparts (given by (2.8)) do not ensure monotonic decrease of the quadratic function (2.2) along the inner-iterations. Indeed, it turns out that this quadratic cost has an erratic behaviour, even on simple examples. Thus, if the iterations are stopped before exact optimality is attained, there is no guarantee that the value of the quadratic cost has decreased, which may then negatively affect the convergence of the TGN algorithm.

A better alternative is as follows. The linear system (2.19) can also be rewritten as

$$\widehat{A}\,\lambda = R^{-1}(d - H(x_b - x_0)), \tag{2.20}$$

where I_m is the identity matrix in dual space and

$$\widehat{A} = R^{-1} H B H^T + I_m. \tag{2.21}$$

This non-symmetric formulation can be solved by PCG with a non-standard inner product in which $R^{-1}HBH^T + I_m$ becomes symmetric. Such an approach has already been used by several authors, see for instance (Stoll and Wathen 2008) and (Ashby, Holst, Manteuffel and Saylor 2001). This strategy is also the main idea behind the RPCG method that can be interpreted as solving the linear system (2.20) using the (possibly semi-)definite HBH^T-(semi) inner product in which $R^{-1}HBH^T + I_m$ is symmetric.

A suitable algorithm for RPCG can be deduced from Algorithm 2.1 by using the "hatted" vectors defined by

$$r = H^T \widehat{r}, \quad p = BH^T \widehat{p}, \quad z = BH^T \widehat{z} \quad \text{and} \quad q = H^T \widehat{q}. \tag{2.22}$$

This derivation is detailed in (Gratton and Tshimanga 2009) and uses the preconditioning matrix G, dual of the primal preconditioner P, such that

$$PH^T = BH^T G. \tag{2.23}$$

This gives the first version of the RPCG algorithm stated as Algorithm 2.2 on the current page, which provides mathematically equivalent iterations to those of Algorithm 2.1.

Algorithm 2.2: PCG Algorithm in R^m (RPCG, version 1)

```
1  $\lambda_0 = 0$;
2  $\widehat{r}_0 = R^{-1}(d - H(x_b - x_0))$;
3  $\widehat{z}_0 = G\widehat{r}_0$;
4  $\widehat{p}_0 = \widehat{z}_0$;
5  for $i = 0, 1, ...$ do
6      $\widehat{q}_i = (I_m + R^{-1}HBH^T)\widehat{p}_i$;
7      $\alpha_i =< \widehat{r}_i, \widehat{z}_i >_{HBH^T} \ / < \widehat{q}_i, \widehat{p}_i >_{HBH^T}$;
8      $\lambda_{i+1} = \lambda_i + \alpha_i \widehat{p}_i$;
9      $\widehat{r}_{i+1} = \widehat{r}_i - \alpha_i \widehat{q}_i$;
10     $\widehat{z}_{i+1} = G\widehat{r}_{i+1}$;
11     if $\|\widehat{r}_{i+1}\|_{HBH^TG} \leq \eta \|\widehat{r}_0\|_{HBH^TG}$ then
12         exit from the loop
13     end
14     $\beta_i =< \widehat{r}_{i+1}, \widehat{z}_{i+1} >_{HBH^T} \ / < \widehat{r}_i, \widehat{z}_i >_{HBH^T}$;
15     $\widehat{p}_{i+1} = \widehat{z}_{i+1} + \beta_i \widehat{p}_i$;
16 end
17 The solution is recovered from $\delta x_0 = x_b - x_0 + BH^T \lambda_{i+1}$;
```

At first sight, assumption (2.23) may appear restrictive, because such a preconditioner G may not exist, in particular if, for some P, PH^T is not included in range of BH^T. However, we show in this paper that the widespread warm-start preconditioning techniques based on limited memory methods (Tshimanga, Gratton, Weaver and Sartenaer 2008), (Morales and Nocedal 2000) do satisfy this condition. Also note that the stopping criterion used in step 11 of Algorithm (2.2) is the translation of (2.17).

This mathematical equivalence of Algorithms PCG and RPCG (under (2.23)) ensures the monotonically decreasing nature of the quadratic cost (2.2) along the RPCG inner iterations, and is thus ideal for applying a termination criterion allowing for approximate solutions. This feature makes RPCG preferable to PSAS in our framework.

However, the first version of the RPCG Algorithm (stated as Algorithm 2.2) is expensive since it requires five matrix vector products (seven matrix vector products counting the stopping criteria) involving HBH^T for each inner loop. Fortunately, it can be rewritten in a much cheaper form by introducing additional dual-space vectors, reducing its cost per loop to a single matrix-vector product with HBH^T. More precisely, consider w and t defined by

$$w_i = HBH^T \widehat{z}_i \text{ and } t_i = HBH^T \widehat{p}_i. \tag{2.24}$$

where $\widehat{z}_i$ and $\widehat{p}_i$ are defined in Algorithm 2.2. If we multiply lines 4 and 15 of Algorithm 2.2 by HBH^T, we obtain that

$$t_i = \begin{cases} w_0 & \text{if } i = 0 \\ w_i + \beta_{i-1} t_{i-1} & \text{if } i > 0, \end{cases}$$

which yields the final version of RPCG (Algorithm 2.3) on the following page.

Algorithm 2.3: RPCG Algorithm

```
$\lambda_0 = 0$;
$\widehat{r}_0 = R^{-1}(d - H(x_b - x_0))$;
$\widehat{z}_0 = G\widehat{r}_0$;
$\widehat{p}_0 = \widehat{z}_0$;
$w_0 = HBH^T\widehat{z}_0$;
$t_0 = w_0$;
for $i = 0, 1, ...$ do
    $\widehat{q}_i = R^{-1}t_i + \widehat{p}_i$;
    $\alpha_i = w_i^T\widehat{r}_i / \widehat{q}_i^T t_i$;
    $\lambda_{i+1} = \lambda_i + \alpha_i\widehat{p}_i$;
    $\widehat{r}_{i+1} = \widehat{r}_i - \alpha_i\widehat{q}_i$;
    $\widehat{z}_{i+1} = G\widehat{r}_{i+1}$;
    $w_{i+1} = HBH^T\widehat{z}_{i+1}$;
    if $\widehat{r}_{i+1}^T\widehat{w}_{i+1} \leq \eta\,\widehat{r}_0^T\widehat{w}_0$ then
        exit from the loop
    end
    $\beta_i = w_{i+1}^T\widehat{r}_{i+1} / w_i^T\widehat{r}_i$;
    $\widehat{p}_{i+1} = \widehat{z}_{i+1} + \beta_i\widehat{p}_i$;
    $t_{i+1} = w_{i+1} + \beta_i t_i$;
end
The solution is recovered from $\delta x_0 = x_b - x_0 + BH^T\lambda_{i+1}$
```

We conclude this presentation of conjugate-gradient-like methods in the dual space by considering the effect of round-off errors. It is known that round-off errors typically cause loss of orthogonality between successive residuals (Fisher, Nocedal, Tremolet and Wright 2009), a central property ensuring fast convergence of the conjugate-gradient methods in exact arithmetic. As a result the rate of convergence might be substantially deteriorated. A possible cure for this problem is to consider reorthogonalization of the residuals, either explicitly (Roux 1989) or in the form of the mathematically equivalent Full-Orthogonalization-Method (FOM) (van der Vorst 2003). It is remarkable that both these strategies, which may be considered as costly in both space and time in the primal setting, turn out to be much cheaper on both counts in the dual framework. This difference is caused by the (typically much) smaller dimension of the residual vectors which need to be stored and reorthogonalized. As a result, complete reorthogonalization may often be considered as a viable computational strategy in conjunction with dual-space solvers.

2.3 Quasi Newton Limited Memory Preconditioners

As mentioned in previous section, we are interested in solving the quadratic problem (2.2) using a variant of conjugate gradients. In practice, this class of method is always associated with suitable preconditioning techniques in order to improve its convergence rate. Finding a good preconditioner and computing an approximation to the inverse Hessian are tightly related problems. We consider here the use of quasi-Newton limited-memory preconditioners (LMPs) (Tschimanga 2007) which are derived from the inverse Hessian approximations using the Limited Memory BFGS (LBFGS) updating formula (Morales and Nocedal 2000). Such techniques are able to improve a preconditioner at each outer loop by using directions generated during the preceeding outer loop, for instance inside the CG Algorithm 2.1. Let p_i, $i = 0, 1, ..., j$ be such directions. We define the preconditioner P_{j+1} for $B^{-1} + H^T R^{-1} H$ in the form of the approximate inverse which is given by

$$P_{j+1} = (I_n - \tau_j p_j q_j^T) P_j (I_n - \tau_j q_j p_j^T) + \tau_j p_j p_j^T, \tag{2.25}$$

where $P_0 = B$, (which exploits the fact that the matrix in (2.2) is a rank m modification of B), $\tau_j = 1/(q_j^T p_j)$ and $q_j = (B^{-1} + H^T R^{-1} H) p_j$.

Several strategies have been proposed to select, within the CG iterations, the l "secant pairs" consisting of descent directions p_j and associated changes in gradient q_j. For instance, the use of the l last pairs is proposed in (Nocedal and Wright 1999, p. 177), while a uniform sampling accross all generated pairs is proposed in (Morales and Nocedal 2000). A remarkable feature of the update (2.25) is that the matrix ΔP_j defined by $\Delta P_j = P_{j+1} - P_j$ is the solution to the following minimization problem (Nocedal and Wright 1999, pp. 139-140):

$$\min_{\Delta P_j} \| W^{1/2} \Delta P_j W^{1/2} \|_F \tag{2.26}$$

$$\text{subject to} \;\; \Delta P_j = \Delta P_j^T, \quad P_{j+1} q_j = p_j, \tag{2.27}$$

where W is any symmetric positive definite matrix satisfying $W p_j = q_j$.

Note that $\| W^{1/2} \Delta P_j W^{1/2} \|_F = \| \Delta P_j \|_W$, where $\| \cdot \|_W$ is a weighted Frobenius norm of weight W. The solution of problem (2.26)-(2.27) can be computed in close form and is given by

$$\Delta P_j = \frac{W^{-1} q_j (p_j - P_j q_j)^T + (p_j - P_j q_j) q_j^T W^{-1}}{q_j^T W^{-1} q_j} - \frac{q_j^T (p_j - P_j q_j) W^{-1} q_j q_j^T W^{-1}}{(q_j^T W^{-1} q_j)^2}. \tag{2.28}$$

Substituting the expression $W p_j = q_j$ into (2.28), it can easily be seen that $P_j + \Delta P_j$ gives (2.25).

May we follow the by now familiar pattern of deriving an equivalent preconditioner in the dual space? We now show that this is indeed possible and that the resulting formula satisfies a variational property similar to that described by (2.26)-(2.27). We first focus on deriving a dual-space preconditioner satisfying (2.23).

Lemma 2.1 *Suppose that $HBH^T G_0 = G_0^T HBH^T$ and that $\widehat{p}_i$ are any linearly independent vectors for $i = 0, 1, ..., j$ and j being the number of stored vectors. Let $\widehat{q}_j$ and $\widehat{\tau}_j$ defined by $\widehat{q}_j = (I_m + R^{-1} M) \widehat{p}_j$, $\widehat{\tau}_j = 1/(\widehat{q}_j^T M \widehat{p}_j)$ with $M = HBH^T$. Then the matrices G_{j+1} defined by*

$$G_{j+1} = (I_m - \widehat{\tau}_j \widehat{p}_j (M \widehat{q}_j)^T) G_j (I_m - \widehat{\tau}_j \widehat{q}_j \widehat{p}_j^T M) + \widehat{\tau}_j \widehat{p}_j \widehat{p}_j^T M, \tag{2.29}$$

satisfies $HBH^T G_{j+1} = G_{j+1}^T HBH^T$.

Suppose also that $P_0 H^T = BH^T G_0$, for instance $P_0 = B$ and $G_0 = I_m$. If we denote $p_j = BH^T \widehat{p}_j$ and $q_j = H^T \widehat{q}_j$, then the sequence P_{j+1} defined by (2.25) and the sequence G_{j+1} defined by (2.29) satisfies $P_{j+1} H^T = BH^T G_{j+1}$.

Proof. The result is proved in two parts by induction, whose inital steps are true by assumption. Suppose G_j is such that $HBH^T G_j = G_j^T HBH^T$. Multiplying G_{j+1} on the left by HBH^T gives

$$M G_{j+1} = (I_m - \widehat{\tau}_j M \widehat{p}_j \widehat{q}_j^T) M G_j (I_m - \widehat{\tau}_j \widehat{q}_j \widehat{p}_j^T M) + \widehat{\tau}_j M \widehat{p}_j \widehat{p}_j^T M. \tag{2.30}$$

Using the symmetry property of G_j in M, we deduce that

$$\begin{aligned} MG_{j+1} &= (I_m - \widehat{\tau}_j M\widehat{p}_j\widehat{q}_j^T)G_j^T M(I_m - \widehat{\tau}_j\widehat{q}_j\widehat{p}_j^T M) + \widehat{\tau}_j M\widehat{p}_j\widehat{p}_j^T M, && (2.31)\\ &= [(I_m - \widehat{\tau}_j M\widehat{p}_j\widehat{q}_j^T)G_j^T(I_m - \widehat{\tau}_j M\widehat{q}_j\widehat{p}_j^T) + \widehat{\tau}_j M\widehat{p}_j\widehat{p}_j^T]M, && (2.32)\\ &= G_{j+1}^T M. && (2.33)\end{aligned}$$

To prove the second part, suppose that G_j is such that $P_jH^T = BH^TG_j$. Using the relations

$$\begin{aligned} p_j &= BH^T\widehat{p}_j, && (2.34)\\ q_j &= H^T\widehat{q}_j, && (2.35)\end{aligned}$$

from the assumption, formula (2.25) can be rewritten in terms of the vectors $\widehat{p}_j$ and $\widehat{q}_j$ in dual space as follows.

$$P_{j+1} = (I_n - \widehat{\tau}_j BH^T\widehat{p}_j\widehat{q}_j^T H)P_j(I_n - \widehat{\tau}_j H^T\widehat{q}_j\widehat{p}_j^T HB) + \widehat{\tau}_j BH^T\widehat{p}_j\widehat{p}_j^T HB,$$

where $\widehat{\tau}_j = 1/(\widehat{q}_j^T HBH^T\widehat{p}_j)$. Multiplying both sides on the right by H^T gives that

$$\begin{aligned} P_{j+1}H^T &= (I_n - \widehat{\tau}_j BH^T\widehat{p}_j\widehat{q}_j^T H)P_j(H^T - \widehat{\tau}_j H^T\widehat{q}_j\widehat{p}_j^T HBH^T) + \widehat{\tau}_j BH^T\widehat{p}_j\widehat{p}_j^T HBH^T\\ &= (I_n - \widehat{\tau}_j BH^T\widehat{p}_j\widehat{q}_j^T H)P_jH^T(I_m - \widehat{\tau}_j\widehat{q}_j\widehat{p}_j^T HBH^T) + \widehat{\tau}_j BH^T\widehat{p}_j\widehat{p}_j^T HBH^T.\end{aligned}$$

Using the relation $P_jH^T = BH^TG_j$, we deduce that

$$\begin{aligned} P_{j+1}H^T &= (I_n - \widehat{\tau}_j BH^T\widehat{p}_j\widehat{q}_j^T H)BH^TG_j(I_m - \widehat{\tau}_j\widehat{q}_j\widehat{p}_j^T HBH^T) + \widehat{\tau}_j BH^T\widehat{p}_j\widehat{p}_j^T HBH^T\\ &= (BH^T - \widehat{\tau}_j BH^T\widehat{p}_j\widehat{q}_j^T HBH^T)G_j(I_m - \widehat{\tau}_j\widehat{q}_j\widehat{p}_j^T HBH^T) + \widehat{\tau}_j BH^T\widehat{p}_j\widehat{p}_j^T HBH^T\end{aligned}$$

and factoring BH^T on the left of this expression yields that

$$P_{j+1}H^T = BH^T[(I_m - \widehat{\tau}_j\widehat{p}_j\widehat{q}_j^T HBH^T)G_j(I_m - \widehat{\tau}_j\widehat{q}_j\widehat{p}_j^T HBH^T) + \widehat{\tau}_j\widehat{p}_j\widehat{p}_j^T HBH^T],$$

from which it can be seen that the formula for G_{j+1} given by (2.29) satisfies $P_{j+1}H^T = BH^TG_{j+1}$. □

Note that, in this Lemma if the vectors p_j and $\widehat{p}_j$ are chosen as the search directions from CG in primal space and accordingly that of from Algorithm (2.3), the relations (2.34) and (2.35) naturally hold (Gratton and Tshimanga 2009). Also, since P is a symmetric positive definite preconditioner and $HPH^T = HBH^TG$, we have that when H has full row rank, the matrix HBH^TG is not only symmetric but also positive definite. Note also that the preconditioner built by (2.29) is a preconditioner for $R^{-1}HBH^T + I_m$.

We now show that the preconditioner G obtained from Lemma 2.1 also satisfies variational properties in the dual space.

Lemma 2.2 *Let $M = HBH^T$ and suppose that M is non-singular. Then the matrix ΔG_j defined by $\Delta G_j = G_{j+1} - G_j$ where G_{j+1} defined in (2.29) is the solution of*

$$\min_{\Delta G_j} \left\| W^{1/2}M^{1/2}\Delta G_j M^{-1/2}W^{1/2}\right\|_F \qquad (2.36)$$

$$\textit{subject to}\;\; M\Delta G_j = \Delta G_j^T M,\; G_{j+1}\widehat{q}_j = \widehat{p}_j,$$

where W is any symmetric positive definite matrix satisfying $WM^{1/2}\widehat{p}_j = M^{1/2}\widehat{q}_j$.

Proof. Using the change of variables

$$\Delta X_j = M^{1/2}\Delta G_j M^{-1/2},\ \widehat{p}_j = M^{-1/2}s_j \text{ and } \ \widehat{q}_j = M^{-1/2}y_j \tag{2.37}$$

we can rewrite problem (2.36) as

$$\min_{\Delta X_j} \left\|W^{1/2}\Delta X_j W^{1/2}\right\|_F$$

$$\text{subject to } \ \Delta X_j = \Delta X_j^T, \quad X_{j+1}y_j = s_j,$$

which is structurally identical to problem (2.26). Using now (2.28) in this context yields that

$$\Delta X_j = \frac{W^{-1}y_j(s_j - X_jy_j)^T + (s_j - X_jy_j)y_j^TW^{-1}}{y_j^TW^{-1}y_j} - \frac{y_j^T(s_j - X_jy_j)W^{-1}y_jy_j^TW^{-1}}{(y_j^TW^{-1}y_j)^2}. \tag{2.38}$$

Substituting (2.37) into this solution and multiplying by $M^{1/2}$ on the right and $M^{-1/2}$ on the left gives that

$$\begin{aligned}\Delta G_j = {} & \frac{M^{-1/2}W^{-1}M^{1/2}\widehat{q}_j(M^{1/2}\widehat{p}_j - M^{1/2}G_j\widehat{q}_j)^TM^{1/2} + (\widehat{p}_j - G_j\widehat{q}_j)\widehat{q}_j^TM^{1/2}W^{-1}M^{1/2}}{\widehat{q}_j^TM^{1/2}W^{-1}M^{1/2}\widehat{q}_j} \\ & - \frac{M^{-1/2}\widehat{q}_j^TM^{1/2}(M^{1/2}\widehat{p}_j - M^{1/2}G_j\widehat{q}_j)W^{-1}M^{1/2}\widehat{q}_j\widehat{q}_j^TM^{1/2}W^{-1}M^{1/2}}{(\widehat{q}_j^TM^{1/2}W^{-1}M^{1/2}\widehat{q}_j)^2}.\end{aligned}$$

From the relation $WM^{1/2}\widehat{p}_j = M^{1/2}\widehat{q}_j$, we deduce that $M^{1/2}\widehat{p}_j = W^{-1}M^{1/2}\widehat{q}_j$. Substituting this expression in the solution gives

$$\Delta G_j = \frac{\widehat{p}_j(\widehat{p}_j - G_j\widehat{q}_j)^TM + (\widehat{p}_j - G_j\widehat{q}_j)\widehat{p}_j^TM}{\widehat{q}_j^TM\widehat{p}_j} - \frac{\widehat{q}_j^TM(\widehat{p}_j - G_j\widehat{q}_j)p_j\widehat{p}_j^TM}{(\widehat{q}_j^TM\widehat{p}_j)^2}. \tag{2.39}$$

On the other hand, we can reformulate (2.29) as:

$$\begin{aligned}G_{j+1} &= G_j - \frac{G_j\widehat{q}_j\widehat{p}_j^TM}{\widehat{q}_j^TM\widehat{p}_j} - \frac{\widehat{p}_j\widehat{q}_j^TMG_j}{\widehat{q}_j^TM\widehat{p}_j} + \frac{\widehat{p}_j\widehat{q}_j^TMG_j\widehat{q}_j\widehat{p}_j^TM}{(\widehat{q}_j^TM\widehat{p}_j)^2} + \frac{\widehat{p}_j\widehat{p}_j^TM}{\widehat{q}_j^TM\widehat{p}_j} & (2.40)\\ &= G_j + \frac{(\widehat{p}_j - G_j\widehat{q}_j)\widehat{p}_j^TM}{\widehat{q}_j^TM\widehat{p}_j} - \frac{\widehat{p}_j\widehat{q}_j^TMG_j}{\widehat{q}_j^TM\widehat{p}_j} + \frac{\widehat{p}_j\widehat{q}_j^TMG_j\widehat{q}_j\widehat{p}_j^TM}{(\widehat{q}_j^TM\widehat{p}_j)^2}. & (2.41)\end{aligned}$$

Adding and substracting the term $\frac{\widehat{p}_j\widehat{p}_j^TM}{\widehat{q}_j^TM\widehat{p}_j}$ gives that,

$$G_{j+1} = G_j + \frac{(\widehat{p}_j - G_j\widehat{q}_j)\widehat{p}_j^TM}{\widehat{q}_j^TM\widehat{p}_j} + \frac{\widehat{p}_j\widehat{p}_j^TM}{\widehat{q}_j^TM\widehat{p}_j} - \frac{\widehat{p}_j\widehat{p}_j^TM}{\widehat{q}_j^TM\widehat{p}_j} - \frac{\widehat{p}_j\widehat{q}_j^TMG_j}{\widehat{q}_j^TM\widehat{p}_j} + \frac{\widehat{p}_j\widehat{q}_j^TMG_j\widehat{q}_j\widehat{p}_j^TM}{(\widehat{q}_j^TM\widehat{p}_j)^2}, \tag{2.42}$$

which can be reorganized as

$$G_{j+1} = G_j + \frac{(\widehat{p}_j - G_j\widehat{q}_j)\widehat{p}_j^TM + \widehat{p}_j(M\widehat{p}_j - G_j^TM\widehat{q}_j)^T}{\widehat{q}_j^TM\widehat{p}_j} - \frac{\widehat{q}_j^TM(\widehat{p}_j - G_j\widehat{q}_j)\widehat{p}_j\widehat{p}_j^TM}{(\widehat{q}_j^TM\widehat{p}_j)^2}. \tag{2.43}$$

Using the property that $MG_j = G_j^T M$ (see Lemma 2.1), we write that

$$G_{j+1} = G_j + \frac{(\widehat{p}_j - G_j\widehat{q}_j)\widehat{p}_j^T M + \widehat{p}_j(\widehat{p}_j - G_j\widehat{q}_j)^T M}{\widehat{q}_j^T M\widehat{p}_j} - \frac{\widehat{q}_j^T M(\widehat{p}_j - G_j\widehat{q}_j)\widehat{p}_j\widehat{p}_j^T M}{(\widehat{q}_j^T M\widehat{p}_j)^2}, \tag{2.44}$$

which is equivalent to $G_j + \Delta G_j$ where ΔG_j is given by the formula (2.39). □

Having found a suitable preconditioner (in the sense that it satisfies (2.23)) and having verified that it shares desirable variational properties with its primal equivalent, we are left with the task of integrating it into the RPCG algorithm. From formula (2.29), we need to store the sequences of $\widehat{q}$, $\widehat{p}$, $M\widehat{p}$, and $M\widehat{q}$ to obtain the successive preconditioner updates. Storing $\widehat{q}$, $\widehat{p}$ and $M\widehat{p}$ does not require additional cost since they are already available from a run of Algorithm 2.3 with same H, R and B. On the other hand, the quantity $M\widehat{q}$ is not a by-product of the algorithm, and seems, at first sight, to require an additional matrix vector product, which may appear computationally costly. Fortunately, under the same assumption on H, R and B, we can rewrite Algorithm 2.3 in a more computationally effective way by introducing a vector l defined by

$$l_i = HBH^T\widehat{r}_i.$$

Since $\widehat{z}_i = G\widehat{r}_i$ and HBH^TG is symmetric from Lemma 2.1, we may therefore write that

$$w_i = HBH^TG\widehat{r}_i = G^THBH^T\widehat{r}_i = G^Tl_i. \tag{2.45}$$

Moreover, multiplying line 11 of Algorithm 2.3 by HBH^T gives that

$$HBH^T\widehat{q}_i = (l_i - l_{i+1})/\alpha_i$$

which is the matrix vector product $M\widehat{q}$ that we need to store. Using all these relations, we can transform Algorithm 2.3 into Algorithm 2.4 on the next page. Note that the algorithm requires two matrix products in the preconditioning phase, one with G and one with G^T.

2.4 Convergence Properties

After defining our dual-space preconditioner, we now consider bounds on its efficiency and compare it to those that can be derived for its primal-space equivalent. For this purpose, we start by recalling known properties of the preconditioned conjugate-gradient method. This method implicitly computes the coefficients of the polynomial $\mathcal{P}_k^*(PA)$ that solves the minimization problem (Axelsson 1996, p. 560)

$$\min_{\mathcal{P}_k} \|(\mathcal{P}_k(PA)PA + I_n)\delta v_0\|_A^2, \tag{2.46}$$

where $A = B^{-1} + H^TR^{-1}H$ is a symmetric positive definite matrix, P is a symmetric positive definite preconditioner, δv_0 is the solution of the linear system (2.13) in primal space and $\mathcal{P}_k$ is a polynomial defined by

$$\mathcal{P}_k(PA) = a_0I + a_1PA + ... + a_k(PA)^k.$$

If PA has eigenvalues $\mu_1 \leq \mu_2 \leq ... \leq \mu_n$, the PCG algorithm (Golub and Van Loan 1989, p. 534) with zero initial starting vector ensures the inequality

$$\|\delta v_{k+1} - \delta v_0\|_A \leq 2\left(\frac{\sqrt{\mu_n} - \sqrt{\mu_1}}{\sqrt{\mu_n} + \sqrt{\mu_1}}\right)^k \|\delta v_0\|_A \tag{2.47}$$

Algorithm 2.4: RPCG Algorithm with quasi-Newton Preconditioner

```
$\lambda_0 = 0$;
$\widehat{r}_0 = R^{-1}(d - H(x_b - x_0))$;
$l_0 = HBH^T\widehat{r}_0$;
$\widehat{z}_0 = G\widehat{r}_0$;
$\widehat{p}_0 = \widehat{z}_0$;
$w_0 = G^T l_0$;
$t_0 = w_0$;
for $i = 0, 1, ...$ do
    $\widehat{q}_i = R^{-1}t_i + \widehat{p}_i$;
    $\alpha_i = w_i^T\widehat{r}_i / \widehat{q}_i^T t_i$;
    $\lambda_{i+1} = \lambda_i + \alpha_i\widehat{p}_i$;
    $\widehat{r}_{i+1} = \widehat{r}_i - \alpha_i\widehat{q}_i$;
    $l_{i+1} = HBH^T\widehat{r}_{i+1}$;
    $\varrho_i = (l_i - l_{i+1})/\alpha_i$;
    $\widehat{z}_{i+1} = G\widehat{r}_{i+1}$;
    $w_{i+1} = G^T l_{i+1}$;
    if $\widehat{r}_{i+1}^T\widehat{w}_{i+1} \leq \eta\widehat{r}_0^T\widehat{w}_0$ then
        exit from the loop
    end
    $\beta_i = w_{i+1}^T\widehat{r}_{i+1} / w_i^T\widehat{r}_i$;
    $\widehat{p}_{i+1} = \widehat{z}_{i+1} + \beta_i\widehat{p}_i$;
    $t_{i+1} = w_{i+1} + \beta_i t_i$;
end
The solution is recovered from $\delta x_0 = x_b - x_0 + BH^T\lambda_{i+1}$
```

(see (Conn et al. 2000, p. 89), for instance). Note that, when RPCG is used, the iterates all belong to the affine subspace IM(BH^T). This information can be taken into account to obtain a better bound on the convergence rate of both RPCG and its PCG primal equivalent, as shown in the next lemma.

Lemma 2.3 *Suppose that HBH^T is non-singular, and that G is a preconditioner satisfying (2.23). If $G(I_m + R^{-1}HBH^T)$ has eigenvalues $\nu_1 \leq \nu_2 \leq ... \leq \nu_m$, then the RPCG Algorithm 2.4 and its primal equivalent ensure the inequality*

$$\|\delta v_{k+1} - \delta v_0\|_A \leq 2\left(\frac{\sqrt{\nu_m} - \sqrt{\nu_1}}{\sqrt{\nu_m} + \sqrt{\nu_1}}\right)^k \|\delta v_0\|_A \leq 2\left(\frac{\sqrt{\mu_n} - \sqrt{\mu_1}}{\sqrt{\mu_n} + \sqrt{\mu_1}}\right)^k \|\delta v_0\|_A . \tag{2.48}$$

Proof. From (2.14) and (2.8) the solution of the linear system (2.13) can be written as $\delta v_0 = BH^T\lambda$, where λ is the solution of the linear system (2.18) given by (2.11). Substituting this form for the solution in the objective function of (2.46) then yields the new form

$$\left\|\left(\sum_{i=0}^{k} a_i\,(PA)^{i+1}BH^T + BH^T\right)\lambda\right\|_A^2$$

and this objective is minimized over all choices of the coefficients $\{a_i\}_{i=0}^k$. Using the fact that $(B^{-1} + H^T R^{-1} H) BH^T = H^T (I_m + R^{-1} HBH^T)$, which we can simply write as $ABH^T = H^T \widehat{A}$, we obtain that our objective may now be written as

$$\left\| \left(\sum_{i=0}^{k} a_i \, (PA)^i PH^T \widehat{A} + BH^T \right) \lambda \right\|_A^2$$

Using the equality (2.23), we obtain the further form

$$\left\| \left(\sum_{i=0}^{k} a_i \, (PA)^i BH^T G \widehat{A} + BH^T \right) \lambda \right\|_A^2 .$$

Substituting the term ABH^T with $H^T \widehat{A}$ and using (2.23) then yields an objective of the form

$$\begin{aligned} \left\| \left(BH^T \sum_{i=0}^{k} a_i \, (G\widehat{A})^{i+1} + I_m \right) \lambda \right\|_A^2 &= \left\| BH^T (\mathcal{P}_k(G\widehat{A}) G\widehat{A} + I_m) \lambda \right\|_A^2 \\ &= \left\| (\mathcal{P}_k(G\widehat{A}) G\widehat{A} + I_m) \lambda \right\|_{HBABH^T}^2 \\ &= \left\| (\mathcal{P}_k(G\widehat{A}) G\widehat{A} + I_m) \lambda \right\|_{HBH^T \widehat{A}}^2 \qquad (2.49) \end{aligned}$$

Performing the change of variables $\widetilde{A} = HBH^T \widehat{A}$ and $\widetilde{P} = G(HBH^T)^{-1}$ in (2.49), we may write the minimization problem in dual space as:

$$\min_{\mathcal{P}_k} \left\| (\mathcal{P}_k(\widetilde{P}\widetilde{A}) \widetilde{P}\widetilde{A} + I_m) \lambda \right\|_{\widetilde{A}}^2 \qquad (2.50)$$

Using the relation (2.23), we then write $(HBH^T)^{-1} HPH^T (HBH^T)^{-1} = G(HBH^T)^{-1}$ which shows that the matrix $\widetilde{P}$ is symmetric positive definite. On the other hand, $\widetilde{A} = HBH^T \widehat{A} = HBH^T + HBH^T R^{-1} HBH^T$ is also a symmetric positive definite matrix. Therefore, from (2.46) and (2.47), if $\widetilde{P}\widetilde{A}$ has eigenvalues $\nu_1 \le \nu_2 \le ... \le \nu_m$, the RPCG Algorithm 2.4 ensures the inequality

$$\| \lambda_{k+1} - \lambda \|_{\widetilde{A}} \le 2 \left(\frac{\sqrt{\nu_m} - \sqrt{\nu_1}}{\sqrt{\nu_m} + \sqrt{\nu_1}} \right)^k \| \lambda \|_{\widetilde{A}} . \qquad (2.51)$$

One also has that

$$\| \lambda \|_{HBH^T \widehat{A}} = \| \lambda \|_{HBABH^T} = \left\| BH^T \lambda \right\|_A = \| \delta v_0 \|_A . \qquad (2.52)$$

Finally, substituting $\widetilde{A}$ with $HBH^T \widehat{A}$ and $\widetilde{P}$ with $G(HBH^T)^{-1}$ in $\widetilde{P}\widetilde{A}$, and then this quantity in (2.51) and using the relation (2.52) proves the first part of the inequality (2.48).

For the second part of the inequality, we start from the equality $ABH^T = H^T \widehat{A}$. If we multiply both sides of this equality from the left by P, we obtain $PABH^T = PH^T \widehat{A}$ and using the equality (2.23), we deduce that $(PA)BH^T = BH^T (G\widehat{A})$. This equality tells us that BH^T spans an invariant subspace of PA, from which we may deduce that every eigenvalue of $G\widehat{A}$ is an eigenvalue of PA. So, $\mu_1 \leq \nu_1$ and $\mu_n \geqslant \nu_m$. From these two inequalities we obtain that

$$\frac{\sqrt{\mu_1}}{\sqrt{\mu_n}} \leq \frac{\sqrt{\nu_1}}{\sqrt{\nu_m}}.$$

Using the fact that the function $\frac{1-x}{1+x}$ is decreasing for $x \geqslant 0$, we then deduce the desired result and complete the proof. □

This lemma shows that the condition number of PA is generally worse than that of $G\widetilde{A}$ and we now show that it can be arbitrarily worse. For example, taking B as the identity matrix, R is a diagonal matrix, $H^T = [I \ 0]$, $P = [G \ 0; \ 0 \ \mathrm{diag}(\xi, 1)]$ where $\mathrm{diag}(\xi, 1)$ is the 2×2 diagonal matrix with the diagonal entries ξ and 1 and $G = (I_m + R^{-1}HH^T)^{-1}$ we easily verify that (2.23) holds. Then the diagonal preconditioned system matrices are

$$\begin{aligned} PA &= \mathrm{diag}(d_i) \text{ where } d_i = 1 \text{ for } 1 \leq i \leq m \text{ and } d_i = \xi \text{ for } i > m, \\ G\widehat{A} &= \mathrm{diag}(d_i) \text{ where } d_i = 1 \text{ for } 1 \leq i \leq m. \end{aligned}$$

We conclude that PA is ill-conditioned with a condition number of $1/\xi$ whereas $G\widehat{A}$ has a condition number of 1, meaning that the convergence takes place in one iteration both in dual and primal spaces. Therefore, for a given preconditioner G in dual space, we can find a preconditioner P in primal space satisfying the relation (2.23) that is arbitrarily ill-conditioned, showing the relevance of the bound (2.48) in terms of the ν's.

Note that we assumed HBH^T to be non-singular for Lemma 2.3, an assumption which has been used to solve the minimization problem (2.50) on the space of polynomials. Lemma 2.3 can nevertheless be generalized to the case where HBH^T is singular, which results in the following improvement on Lemma 2.3.

Lemma 2.4 *Suppose that HBH^T is singular, and that G is a preconditioner satisfying (2.23). Then, it is possible to find a $r \times n$ matrix of rank r $\check{H}$ and a $r \times r$ matrix of rank r $\check{R}^{-1}$ satisfying*

$$B^{-1} + H^T R^{-1} H = B^{-1} + \check{H}^T \check{R}^{-1} \check{H} \tag{2.53}$$

where $r < m$, and a $r \times r$ matrix $\check{G}$ satisfying

$$F\check{H}^T = B\check{H}^T\check{G}. \tag{2.54}$$

Moreover, the RPCG Algorithm 2.4 and its primal equivalent ensure the inequality

$$\|\delta v_{k+1} - \delta v_0\|_A \leq 2\left(\frac{\sqrt{\nu_r} - \sqrt{\nu_1}}{\sqrt{\nu_r} + \sqrt{\nu_1}}\right)^k \|\delta v_0\|_A \leq 2\left(\frac{\sqrt{\mu_n} - \sqrt{\mu_1}}{\sqrt{\mu_n} + \sqrt{\mu_1}}\right)^k \|\delta v_0\|_A, \tag{2.55}$$

where $\nu_1 \leq \nu_2 \leq ... \leq \nu_r$ are the eigenvalues of the full rank matrix $\check{G}(I_r + \check{R}^{-1}\check{H}B\check{H}^T)$.

We refer to the appendix for the proof of this result.

3 The Steihaug-Toint Truncated Conjugate Gradient Method in Dual Space

When using the simple Gauss-Newton approach described at the beginning of Section 2 for more than mildly non-linear cost functions, the iterations can unfortunately diverge, and the function value can increase with the Gauss-Newton step computed from (2.2), see for instance (Kelley 1999, p. 39). This problem is not purely theoretical, and is also discussed in a real life problem in (Tshimanga et al. 2008), where the necessity for global minimization is emphasized. As indicated above, global convergence can be ensured by inserting the Gauss-Newton strategy in a trust-region framework. For data assimilation problem, trust-region methods amount to solving approximately a sequence of quadratic problems

$$\min_{\delta x_0^k} \frac{1}{2} \left\| \delta x_0^k + x_0 - x_b \right\|_{B^{-1}}^2 + \frac{1}{2} \left\| H_k \delta x_0^k - d \right\|_{R^{-1}}^2 \tag{3.1}$$

$$\text{subject to } \|\delta x_0^k\|_{P_k^{-1}} \leq \Delta_k, \tag{3.2}$$

where Δ_k is the radius of the "trust region", which is the region where we believe that the objective function (2.1) of our nonlinear problem is adequately approximated by that of (3.1). It is important to note that preconditioning appears in this problem as the norm $\| \cdot \|_{P_k^{-1}}$ used in (3.2).

After solving this subproblem, the step δx_0^k is accepted or rejected and the trust region radius is updated accordingly. The acceptance of the trial point and trust region radius update are decided by considering the ratio

$$\rho_k = \frac{f(x_0^k) - f(x_0^k + \delta x_0^k)}{m_k(x_0^k) - m_k(x_0^k + \delta x_0^k)}$$

where f is the objective function (2.1) and m_k is its quadratic approximation (3.1). This ratio of achieved to predicted reductions gives an indication of the model's quality. If it is larger than some small constant, the step is accepted and the trust-region radius possibly enlarged, while, if it is too small or negative, the step is rejected and the trust-region radius decreased. We refer the reader to (Conn et al. 2000, p. 116) for a more complete description. For large scale instances, the subproblem (3.1)-(3.2) can be solved approximately using iterative techniques, for instance the Large-Scale Trust-Region Subproblem (LSTRS) algorithm (Lampe, Rojas, Sorensen and Voss 2011), (Rojas, Santos and Sorensen 2008) or the Steihaug-Toint truncated conjugate-gradient method (Conn et al. 2000, p. 205). We consider below an efficient implementation of the Steihaug-Toint truncated conjugate gradient method. A similar adaptation of LSTRS is out of the scope of the present paper and might be considered for future work.

The Steihaug-Toint truncated conjugate-gradient technique, where the model (3.1) is approximately minimized using PCG until the boundary of the trust region (3.2) is encountered. More specifically, (dropping again the outer-iterations index k for simplicity) three different cases may occur when applying PCG to (3.1) (Conn et al. 2000, pp. 202-204):

1. the curvature $\langle p_i, Ap_i \rangle$ remains positive at each inner iteration, and the PCG iterates remain inside the trust region (the standard PCG stopping rule (2.17) then applies);

2. the curvature $\langle p_i, Ap_i \rangle$ remains positive at each inner iteration, and the PCG iterates leave the trust region, in which case the iterates are stopped when the trust region boundary is met;

3. the curvature $\langle p_i, Ap_i \rangle$ is negative at some PCG step, in which case, the associated descent direction is followed until the trust region boundary is met.

This strategy can be shown to yield a sufficient decrease condition (Nocedal and Wright 1999, p. 33) which guarantees global convergence of the iterates. Note that, since the curvature is always positive in our study, we consider the first and second situations only.

Again the same question arises: may we derive an equivalent dual-space version of this method? In particular, how easy is it to compute a final iterate on the boundary of the trust region following a descent direction form a given inner iterate? For answering these questions, we start by rewriting the Steihaug-Toint algorithm described in (Conn et al. 2000, p. 205) in terms of the vectors in dual space, using the relations (Gratton and Tshimanga 2009) $r_i = H^T \widehat{r}_i$, $p_i = BH^T \widehat{p}_i$, $\delta v_i = BH^T \lambda_i$, $z_i = BH^T \widehat{z}_i$, $q_i = H^T \widehat{q}_i$ and the equality $PH^T = BH^T G$ where G is the preconditioner in dual space. This gives a first version of the Steihaug-Toint truncated conjugate gradient algorithm (Algorithm 3.1) in dual space.

Algorithm 3.1: The Steihaug-Toint truncated CG method in dual space (version 1)

```
$\lambda_0 = 0$;
$\widehat{r}_0 = R^{-1}(d - H(x_b - x_0))$;
$\widehat{z}_0 = G\widehat{r}_0$;
$\widehat{p}_0 = \widehat{z}_0$;
for $i = 0, 1, ...$ do
    $\widehat{q}_i = (R^{-1}HBH^T + I_m)\widehat{p}_i$;
    $\alpha_i = \langle \widehat{r}_i, \widehat{z}_i \rangle_{HBH^T} / \langle \widehat{p}_i, \widehat{q}_i \rangle_{HBH^T}$;
    if $\|\lambda_i + \alpha_i \widehat{p}_i\|_{HBH^T G^{-1}} \geqslant \Delta$ then
        compute $\alpha_i$ as the positive root of $\|\lambda_i + \alpha_i \widehat{p}_i\|_{HBH^T G^{-1}} = \Delta$;
        $\lambda_{i+1} = \lambda_i + \alpha_i \widehat{p}_i$ ;
        return;
    end
    $\lambda_{i+1} = \lambda_i + \alpha_i \widehat{p}_i$;
    $\widehat{r}_{i+1} = \widehat{r}_i - \alpha_i \widehat{q}_i$;
    $\widehat{z}_{i+1} = G\widehat{r}_{i+1}$;
    if $\|\widehat{r}_{i+1}\|_{HBH^T G} \leq \eta \|\widehat{r}_0\|_{HBH^T G}$ then
        exit from the loop
    end
    $\beta_i = \langle \widehat{r}_{i+1}, \widehat{z}_{i+1} \rangle_{HBH^T} / \langle \widehat{r}_i, \widehat{z}_i \rangle_{HBH^T}$;
    $\widehat{p}_{i+1} = \widehat{z}_{i+1} + \beta_i \widehat{p}_i$;
end
The solution is recovered from $\delta x_0 = x_b - x_0 + BH^T \lambda_{i+1}$
```

As before, this version of the algorithm turns out to be very expensive in terms of HBH^T matrix-vector products, and we introduce new vectors to transform it into a computationally efficient method. From

$$\|\lambda_i + \alpha_i \widehat{p}_i\|^2_{HBH^T G^{-1}} = \|\lambda_i\|^2_{HBH^T G^{-1}} + 2\alpha_i \langle \lambda_i, HBH^T G^{-1} \widehat{p}_i \rangle + \alpha_i^2 \|\widehat{p}_i\|^2_{HBH^T G^{-1}},$$

the positive root of $\|\lambda_i + \alpha_i \widehat{p}_i\|^2_{HBH^TG^{-1}} = \Delta$ is given by

$$\alpha_i = \frac{-\langle \lambda_i, HBH^TG^{-1}\widehat{p}_i\rangle + \sqrt{\langle \lambda_i, HBH^TG^{-1}\widehat{p}_i\rangle^2 + \|\widehat{p}_i\|^2_{HBH^TG^{-1}}\left(\Delta^2 - \|\lambda_i\|^2_{HBH^TG^{-1}}\right)}}{\|\widehat{p}_i\|^2_{HBH^TG^{-1}}}$$

Consider now the vectors y_i, v_i and s_i defined by

$$y_i = HBH^T\lambda_i,\ v_i = G^{-1}\lambda_i \text{ and } s_i = G^{-1}\widehat{p}_i.$$

If we multiply line 4 and 20 of the Algorithm 3.1 by G^{-1} we obtain that

$$s_i = \begin{cases} \widehat{r}_0 & \text{if } i = 0 \\ \widehat{r}_i + \beta_{i-1}s_{i-1} & \text{if } i > 0, \end{cases}$$

and line 1 and 13 by HBH^T and G^{-1} we obtain that

$$y_i = \begin{cases} 0 & \text{if } i = 0 \\ y_{i-1} + \alpha_{i-1}t_{i-1} & \text{if } i > 0, \end{cases}$$

$$v_i = \begin{cases} 0 & \text{if } i = 0 \\ v_{i-1} + \alpha_{i-1}s_{i-1} & \text{if } i > 0, \end{cases}$$

where t_{i-1} is given by (2.24). We may now use these new vectors to calculate α_{i-1} , which yields

$$\alpha_i = \frac{-y_i^Ts_i + \sqrt{(y_i^Ts_i)^2 + t_i^Ts_i(\Delta^2 - y_i^Tv_i)}}{t_i^Ts_i}. \tag{3.3}$$

Introducing this change, we now obtain Algorithm 3.2 on page 24. This last version requires a single HBH^T matrix-vector product and products with G and G^T in each inner loop.

This is the algorithm which we recommend for solving truly nonlinear instances of our original problem (2.1) when $m \ll n$.

4 Conclusions

Inverse problems where an n-vector is estimated using physical observations are very common in the simulation of complex systems. Important applications abound in environmental sciences like meteorology or oceanography, where the estimated vector is the initial state of a dynamical system which is integrated in time to produce a forecast. Our work concentrates on the situation where the estimation process results in a nonlinear least squares problem, in which there are much fewer observations than variables to be estimated, and where a quadratic regularization term has therefore to be introduced in order to guarantee uniqueness of the solution. We consider a solution method based on a truncated Gauss-Newton technique, made globally convergent with a trust-region strategy. The sequence of linear least-squares problems involved in the method is iteratively solved by a conjugate-gradients method, appropriately truncated by the Steihaug-Toint strategy, and which is accelerated by limited memory preconditioners.

It has been recently shown that it is possible to use duality theory and rewrite the linear least-squares solver into an equivalent algorithm (in exact arithmetic) in which all vectors of the short-term recurrences are represented by vectors of dimension m, m being the number of physical observations. Two proposed dual approaches, called PSAS and RPCG, are shown to differ in the way they define the scalar product in the dual space. It is also argued that the RPCG method is preferable to PSAS because it maintains the convergence properties of the initial Gauss-Newton process.

In this paper, we take a further step in making the RPCG dual solver relevant to practice for large scale, nonlinear problems. This is done by introducing an adequate preconditioner and an efficient implementation of the Steihaug-Toint truncation of CG in the dual space. All these techniques are implemented in such a way that RPCG and the primal approach generate the same sequence of iterates and keep the number of matrix-vector products in the CG algorithm constant. A further advantage of the proposed dual approach in the common situation where $m \ll n$, is that storing vectors for the preconditioner or performing re-orthogonalization is computationally much cheaper than with standard primal algorithms, making any of these techniques applicable in realistic cases.

There are many open issues worth further exploration. A first important issue is the development of other preconditioners in the dual space, like the Ritz preconditioner (Tshimanga et al. 2008) which proved efficient in data assimilation for oceanography. It could also be interesting to further explore globalization strategies by developping algorithms that are similar in spirit to the Moré-Sorensen (Conn et al. 2000, Section 7.3) approach, or by considering techniques based on cubic regularization (Cartis, Gould and Toint 2009).

Acknowledgments

The third author gratefully acknowledges partial support from CERFACS.

Appendix: Proof for the Lemma 2.4

Proof. If the singular value decomposition for H is given by

$$H = [U_1 \;\; U_2] \begin{bmatrix} \Sigma_r & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}, \tag{4.1}$$

a possible theoretical choice for $\check{H}$ could be

$$\check{H} = \Sigma_r V_1^T.$$

Denoting $\check{R} = \left(U_1^T R^{-1} U_1\right)^{-1}$, direct computations show that

$$B^{-1} + H^T R^{-1} H = B^{-1} + \check{H}^T \check{R}^{-1} \check{H}.$$

Using the assumption (2.23) and denoting $\check{G} = U_1^T G U_1$, we also obtain that $F\check{H}^T = B\check{H}^T\check{G}$.

The matrix $\check{H}$ is now a $r \times n$ matrix of rank r and Lemma 2.3 can then be applied using r, $\check{R}$, $\check{H}$ and $\check{G}$ instead of m, R, H and G, yielding the desired result where $\nu_1, ..., \nu_r$ the eigenvalues of $\check{G}(I_r + \check{R}^{-1}\check{H}B\check{H}^T)$, replace those of $G(I_m + R^{-1}HBH^T)$ in (2.48). We next investigate the relations between these two sets of eigenvalues.

Using the relation on $\check{H}$, $\check{R}$ and $\check{G}$, it can be shown that

$$[U_1 U_1^T G \widehat{A}] U_1 = U_1 [\check{G}\check{A}] \tag{4.2}$$

where $\widehat{A}$ is defined in (2.21). This says that U_1 is an invariant subspace of $U_1U_1^TG\widehat{A}$ and every eigenvalue of $\check{G}\breve{A}$ is an eigenvalue of $U_1U_1^TG\widehat{A}$. Therefore, the nonzero eigenvalues of $U_1U_1^TG\widehat{A}$ are equal to the eigenvalues of $\check{G}\breve{A}$ using the fact that $U_1U_1^TG\widehat{A}$ has $(m-r)$ null eigenvalues. We now consider the relations between the eigenvalues of $U_1U_1^TG\widehat{A}$ and $G\widehat{A}$, and start by rewriting these matrices blockwise.

Using the relation (4.1), it can be shown that

$$HBH^T = U_1\Sigma_rV_1^TBV_1\Sigma_rU_1^T. \tag{4.3}$$

Defining

$$U_1 = \begin{bmatrix} U_r \\ 0 \end{bmatrix}, \tag{4.4}$$

HBH^T can thus be rewritten in a block matrix form as

$$HBH^T = \begin{bmatrix} U_r\Sigma_rV_1^T \\ 0 \end{bmatrix} B \begin{bmatrix} V_1\Sigma_rU_r^T & 0 \end{bmatrix} = \begin{bmatrix} M_r & 0 \\ 0 & 0 \end{bmatrix}, \tag{4.5}$$

where $M_r = U_r\Sigma_rV_1^TBV_1\Sigma_rU_r^T$ has full rank r. Using the equality (2.23), we can write that

$$HFH^T = HBH^TG. \tag{4.6}$$

Hence, HBH^TG is symmetric due to the symmetry of HFH^T. Using the relation (4.5) and defining

$$G = \begin{bmatrix} G_r & G_2 \\ G_3 & G_{m-r} \end{bmatrix}, \tag{4.7}$$

where G_r is $r\times r$ matrix and G_{m-r} is a $m-r\times m-r$ matrix, we can write HBH^TG as

$$HBH^TG = \begin{bmatrix} M_rG_r & M_rG_2 \\ 0 & 0 \end{bmatrix}. \tag{4.8}$$

From the symmetry of HBH^TG given by (4.8), $M_rG_2 = 0$ which implies that $G_2 = 0$ since M_r is a full rank matrix. Thus, G has the form

$$G = \begin{bmatrix} G_r & 0 \\ G_3 & G_{m-r} \end{bmatrix}. \tag{4.9}$$

We next derive a block matrix form of $\widehat{A}$. Defining R^{-1} as

$$R^{-1} = \begin{bmatrix} R_r^{-1} & R_2^{-1} \\ R_3^{-1} & R_{m-r}^{-1} \end{bmatrix}, \tag{4.10}$$

and using (4.5) and the definition of $\widehat{A}$, we can write $\widehat{A}$ as

$$\widehat{A} = I + \begin{bmatrix} R_r^{-1} & R_2^{-1} \\ R_3^{-1} & R_{m-r}^{-1} \end{bmatrix} \begin{bmatrix} M_r & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \widehat{A}_r & 0 \\ 0 & I \end{bmatrix}, \tag{4.11}$$

where $\widehat{A}_r = I + R_r^{-1} M_r$. From (4.4), (4.9) and (4.11), we deduce that

$$G\widehat{A} = \begin{bmatrix} G_r \widehat{A}_r & 0 \\ G_3 \widehat{A}_r & G_{m-r} \end{bmatrix} \text{ and } U_1 U_1^T G\widehat{A} = \begin{bmatrix} G_r \widehat{A}_r & 0 \\ 0 & 0 \end{bmatrix}. \tag{4.12}$$

From (4.12), the eigenvalues of $G\widehat{A}$ are the eigenvalues of $G_r A_r$ and the eigenvalues of G_{m-r}. Also, the eigenvalues of $U_1 U_1^T G\widehat{A}$ are the eigenvalues of $G_r A_r$ and $(m-r)$ null eigenvalues. Therefore, the nonzero eigenvalues of $U_1 U_1^T G\widehat{A}$ which are equal to the eigenvalues of $\check{G}\check{A}$ form a subset of the eigenvalues of $G\widehat{A}$. As a result, the eigenvalues of $\check{G}(I_r + \check{R}^{-1}\check{H}B\check{H}^T)$ can be used in (2.47) instead of those $G(I_m + R^{-1}HBH^T)$, which completes the proof.

References

A. E. Akkraoui, P. Gauthier, S. Pellerin, and S. Buis. Intercomparison of the primal and dual formulations of variational data assimilation. *Quarterly Journal of the Royal Meteorological Society*, **134**, 1015–1025, 2008.

M. Arioli. A stopping criterion for the conjugate gradient algorithm in a finite element framework. *Numerische Mathematik*, **97**, 1–24, 2004.

S. F. Ashby, M. J. Holst, T. A. Manteuffel, and P. E. Saylor. The role of the inner product in stopping criteria for conjugate gradient iterations. *BIT*, **41**(1), 026–052, 2001.

O. Axelsson. *Iterative Solution Methods*. Cambridge University Press, Cambridge, England, 1996.

Å. Björck. *Numerical Methods for Least Squares Problems*. SIAM, Philadelphia, USA, 1996.

F. Bouttier and P. Courtier. Data assimilation concepts and methods. Technical report, ECMWF, Reading, England, 1999.

C. Cartis, N. I. M. Gould, and Ph. L. Toint. Adaptive cubic overestimation methods for unconstrained optimization. Part I: motivation, convergence and numerical results. *Mathematical Programming, Series A*, 2009. DOI: 10.1007/s10107-009-0286-5, 51 pages.

A. R. Conn, N. I. M. Gould, and Ph. L. Toint. *Trust-Region Methods*. Number 01 *in* 'MPS-SIAM Series on Optimization'. SIAM, Philadelphia, USA, 2000.

Ph. Courtier. Dual formulation of four-dimensional variational assimilation. *Quarterly Journal of the Royal Meteorological Society*, **123**, 2449–2461, 1997.

Ph. Courtier, J.-N. Thépaut, and A. Hollingsworth. A strategy for operational implementation of 4D-Var using an incremental approach. *Quarterly Journal of the Royal Meteorological Society*, **120**, 1367–1388, 1994.

J. E. Dennis and R. B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall, Englewood Cliffs, NJ, USA, 1983. Reprinted as *Classics in Applied Mathematics 16*, SIAM, Philadelphia, USA, 1996.

M. Fisher, J. Nocedal, Y. Tremolet, and S. J. Wright. Data assimilation in weather forecasting: A case study in PDE-constrained optimization. *Optimization and Engineering*, **10**, 409–426, 2009.

R. Giering and T. Kaminski. Recipes for adjoint code construction. *ACM Trans. Math. Softw.*, **24**(4), 437–474, 1998.

G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, second edn, 1989.

S. Gratton and J. Tshimanga. An observation-space formulation of variational assimilation using a restricted preconditioned conjugate-gradient algorithm. *Quarterly Journal of the Royal Meteorological Society*, **135**, 1573–1585, 2009.

S. Gratton, A. Lawless, and N. K. Nichols. Approximate Gauss-Newton methods for nonlinear least-squares problems. *SIAM Journal on Optimization*, **18**, 106–132, 2007.

C. T. Kelley. *Iterative Methods for Optimization*. Frontiers in Applied Mathematics. SIAM, Philadelphia, USA, 1999.

J. Lampe, M. Rojas, D. C. Sorensen, and H. Voss. Accelerating the LSTRS algorithm. *SIAM Journal on Scientific Computing*, **33**(1), 175–194, 2011.

J. L. Morales and J. Nocedal. Automatic preconditioning by limited-memory quasi-Newton updating. *SIAM Journal on Optimization*, **10**, 1079–1096, 2000.

J. Nocedal and S. J. Wright. *Numerical Optimization*. Series in Operations Research. Springer Verlag, Heidelberg, Berlin, New York, 1999.

M. Rojas, S. A. Santos, and D. C. Sorensen. Algorithm 873: LSTRS: MATLAB software for large-scale trust-region subproblems and regularization. *ACM Transactions on Mathematical Software*, **34**(2), 11, 2008.

F. X. Roux. Acceleration of the outer conjugate gradient by reorthogonalization for a domain decomposition method for structural analysis problems. *in* 'Proceedings of the 3rd international conference on Supercomputing', ICS '89, pp. 471–476, New York, NY, USA, 1989. ACM.

M. Stoll and A. Wathen. Combination preconditioning and the Bramble-Pasciak$^+$ preconditioner. *SIAM Journal on Matrix Analysis and Applications*, **30**(2), 582–608, 2008.

A. Tarantola. *Inverse problem theory. Methods for data fitting and model parameter estimation*. Elsevier, Amsterdam, The Netherlands, 1987.

Jean Tschimanga. *On a Class of Limited Memory Preconditioners for Large-Scale Nonlinear Least-Squares Problems (with Application to Variational Ocean Data Assimilation)*. PhD thesis, Department of Mathematics, FUNDP - University of Namur, Namur, Belgium, 2007.

J. Tshimanga, S. Gratton, A. Weaver, and A. Sartenaer. Limited-memory preconditioners with application to incremental four-dimensional variational data assimilation. *Quarterly Journal of the Royal Meteorological Society*, **134**, 751–769, 2008.

H. A. van der Vorst. *Iterative Krylov methods for large linear systems.* Cambridge University Press, 2003.

A. T. Weaver, J. Vialard, and D. L. T. Anderson. Three- and four-dimensional variational assimilation with a general circulation model of the tropical Pacific ocean, part 1 : formulation, internal diagnostics and consistency checks. *Monthly Weather Review*, **131**, 1360–1378, 2003.

Algorithm 3.2: The Steihaug-Toint truncated CG method in dual space

```
$\lambda_0 = 0$;
$\widehat{r}_0 = R^{-1}(d - H(x_b - x_0))$;
$l_0 = HBH^T\widehat{r}_0$;
$\widehat{z}_0 = G\widehat{r}_0$;
$p_0 = z_0$;
$w_0 = G^T l_0$;
$t_0 = w_0$;
$s_0 = \widehat{r}_0$;
$y_0 = 0$;
$v_0 = 0$;
for $i = 0, 1, ...$ do
    $\widehat{q}_i = R^{-1}t_i + \widehat{p}_i$;
    $\alpha_i = w_i^T\widehat{r}_i / t_i^T\widehat{q}_i$;
    $\gamma = \sqrt{y_i^T v_i + 2\alpha_i y_i^T s_i + \alpha_i^2 t_i^T s_i}$;
    if $\gamma \geqslant \Delta$ then
        Calculate $\alpha_i$ from the formula 3.3;
        $\lambda_{i+1} = \lambda_i + \alpha_i\widehat{p}_i$ ;
        return ;
    end
    $\lambda_{i+1} = \lambda_i + \alpha_i\widehat{p}_i$;
    $\widehat{r}_{i+1} = \widehat{r}_i - \alpha_i\widehat{q}_i$;
    $l_{i+1} = HBH^T\widehat{r}_{i+1}$;
    $\varrho_i = (l_i - l_{i+1})/\alpha_i$;
    $\widehat{z}_{i+1} = G\widehat{r}_{i+1}$;
    $w_{i+1} = G^T l_{i+1}$;
    if $\widehat{r}_{i+1}^T\widehat{w}_{i+1} \leq \eta\widehat{r}_0^T\widehat{w}_0$ then
        exit from the loop
    end
    $\beta_i = w_{i+1}^T\widehat{r}_{i+1} / w_i^T\widehat{r}_i$;
    $y_{i+1} = y_i + \alpha_i t_i$;
    $\widehat{p}_{i+1} = \widehat{z}_{i+1} + \beta_i\widehat{p}_i$;
    $t_{i+1} = w_{i+1} + \beta_i t_i$;
    $v_{i+1} = v_i + \alpha_i s_i$;
    $s_{i+1} = \widehat{r}_{i+1} + \beta_i s_i$;
end
The solution is recovered from $\delta x_0 = x_b - x_0 + BH^T\lambda_{i+1}$
```