DELFT UNIVERSITY OF TECHNOLOGY

REPORT 10-08

On the convergence behaviour of IDR(s)

Peter Sonneveld

ISSN 1389-6520

Reports of the Department of Applied Mathematical Analysis

Delft 2010

On the convergence behaviour of IDR(s)

Peter Sonneveld*

March 10, 2010

Abstract

An explanation is given of the convergence behaviour of the IDR(s) methods. The convergence of the IDR(s) algorithms has two components. The first consists of damping properties of certain factors in the residual polynomials, which becomes less important for large values of s. The second component depends on the behaviour of quasi-Lanczos polynomials that occur in the theoretical description.

In this paper, this second component is analysed, the convergence behaviour of the methods is explained, and an expectation is given on the rate of convergence.

Keywords: Iterative methods, IDR, Krylov subspace methods, Bi-CGSTAB, nonsymmetric linear systems.

1 Introduction.

The IDR(s) method is a family of short recurrence Krylov subspace methods for solving a linear system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$, based on the following proposition [15]:

Proposition 1.1 *Let $\boldsymbol{A}$ be an $N \times N$ complex matrix, let $\boldsymbol{b}$ be a vector in $\mathbb{C}_N$, let $\mathcal{G}_0$ be the full Krylov subspace $\mathcal{K}(\boldsymbol{A}, \boldsymbol{b})$. let $\mathcal{S}$ be a proper subspace of $\mathcal{G}_0$ of codimension s, and let the sequence of spaces $\mathcal{G}_j$, $j = 1, 2, \ldots$ be defined recursively by*

$$\mathcal{G}_j = (\boldsymbol{I} - \omega_j \boldsymbol{A})(\mathcal{S} \cap \mathcal{G}_{j-1})$$

where ω_j are nonzero complex numbers, then the spaces are nested in the following way:

$$\mathcal{G}_j \subset \mathcal{G}_{j-1}$$

Moreover, apart from really exceptional circumstances, the dimensions of the spaces satisfy

$$dim(\mathcal{G}_j) = dim(\mathcal{G}_{j-1}) - s$$

This is the so-called dimension reduction phenomenon, and it is proved in two theorems in [15].
The IDR(s) algorithms make use of this property by constructing a sequence of residual vectors $\boldsymbol{r}_n = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_n$ that are forced to be in $\mathcal{G}_j$ for increasing values of j. The space $\mathcal{S}$ is chosen to be the null space of $\boldsymbol{P}^*$, where $\boldsymbol{P}$ is a randomly chosen $N \times s$ matrix. The construction principle is as follows: Suppose we have $s + 1$ independent residuals

*Delft University of Technology, Delft Institute of Applied Mathematics, Mekelweg 4, 2628 CD, The Netherlands. E-mail: p.sonneveld@ewi.tudelft.nl

$\boldsymbol{r}_{j1}, \boldsymbol{r}_{j2}, \ldots, \boldsymbol{r}_{j,s+1}$ in $\mathcal{G}_j$, we can make a nontrivial combination $\sum_{k=1}^{s+1} c_k \boldsymbol{r}_{jk}$ in $\mathcal{S} \cap \mathcal{G}_j$ by solving $c_1, c_2, \ldots, c_{s+1}$ from the homogeneous linear system

$$\boldsymbol{P}^* \left(\sum_{k=1}^{s+1} c_k \boldsymbol{r}_{jk} \right) = \boldsymbol{0}$$

By applying the mapping $\boldsymbol{I} - \omega_{j+1}\boldsymbol{A}$ to this combination, we obtain a vector in $\mathcal{G}_{j+1}$. By suitable scaling this vector can be made a residual $\boldsymbol{r}_{n+1}$, and also an update $\boldsymbol{x}_{n+1}$ for the solution can be found. This process can be repeated using also this new residual, because the $\mathcal{G}$-spaces are nested. After $s+1$ steps, we have found $s+1$ vectors in $\mathcal{G}_{j+1}$, and therefore we can enter the space $\mathcal{G}_{j+2}$ etc.
Every $s+1$ iterations, we enter a new space of which the dimension is s less than the former. Therefore in about $\frac{s+1}{s}N$ steps, the space $\mathcal{G}_j$ with $js \geq N$ has dimension zero, and hence the residuals in it are zero. In this sense, the IDR(s) method is finite.

This behaviour is illustrated in the upper plot of Figure 1, showing the convergence history of IDR(s) applied to a discretized one-dimensional diffusion equation, leading to 60 equations (Problem 1).
In the lower plot of Figure 1, the same graph is shown, but now with the horizontal axis stretched by a factor $\frac{s}{s+1}$. This plot confirms the $\frac{s+1}{s}N$ behaviour as predicted by the theory. Plots in which this scaling has been applied will be called *normalized* plots. In these plots, the horizontal axis shows no longer the number of matrix-vector multiplications but only $\frac{s}{s+1}$ times that number.

Now similarly as in the CG-algorithm, the method usually converges as an iterative method in a number of steps that is far below the theoretical bound $\frac{s+1}{s}N$. This is illustrated in the upper plot of Figure 2, showing the convergence history for a system of 3900 equations arising from a two-dimensional convection diffusion equation on a 60×65 grid, with mesh-Peclet numbers $[0.5, 0]$ (Problem 2). Here only about 200 iterations are required to gain 10 decimal digits.
If the IDR(s) methods are compared with full GMRES, the convergence characteristic seems to 'converge' at increasing s to a limiting curve close to the full GMRES curve for the same problem. Now GMRES is a 'lower bound' for all Krylov subspace methods, because it minimizes the residual norm $\|\boldsymbol{r}_n\|$ over the Krylov space $\text{span}\{\boldsymbol{r}_0, \boldsymbol{A}\boldsymbol{r}_0, \ldots, \boldsymbol{A}^n\boldsymbol{r}_0\}$, and the observed increasing similarity of GMRES and IDR(s) for increasing values of s is quite promising.
If we take a closer look to the upper plots of Figure 1 and Figure 2, it can be observed that the relative positioning of the convergence curves for the values $s = 1, 2, 8, 16$ in both plots are a bit similar. So apparently the 'speeds of convergence' for different s are more or less proportional to $\frac{s+1}{s}$.
Whether this is true can be verified by applying the same stretching, with $\frac{s}{s+1}$, of the horizontal axis as done in the left-lower plot. The result is shown in the lower plot of Figure 2.
We can make two observations in the right-lower plot.
1. The curves for $s = 1$ and $s = 2$ seem to show faster convergence than GMRES, which is not possible.
2. The curves for $s = 8$ and $s = 16$ are nearly covering the fast convergence part of the full GMRES curve.

The first observation has a simple explanation: For $s = 1$, only half of the work is shown, and for $s = 2$, only 66%.
The second observation will be analysed in the following sections.

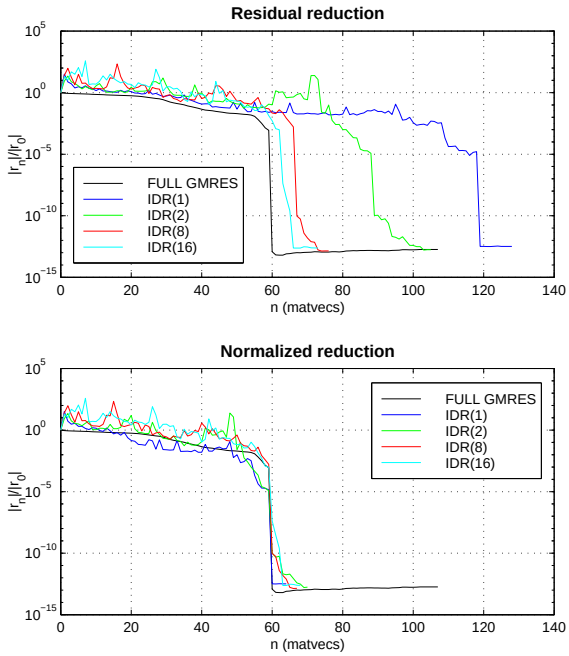

Figure 1: History for Problem 1

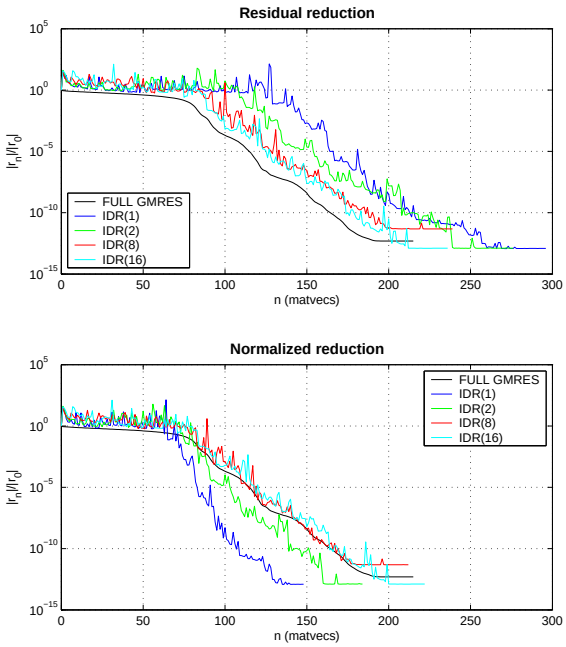

Figure 2: History for Problem 2

2 The polynomial background of the IDR(s) residuals.

2.1 Damping and Lanczos part of the convergence.

In each step of an IDR(s) algorithm a new residual is constructed, involving one matrix-vector multiplication. Therefore, as in most Krylov subspace methods, we can write

$$\boldsymbol{r}_n = \Phi_n(\boldsymbol{A})\boldsymbol{r}_0 \tag{1}$$

where Φ_n is an n-th degree polynomial, satisfying $\Phi_n(0) = 1$. We call this the *IDR-polynomial*

Furthermore, as described in [15], for residuals that are in $\mathcal{G}_j$, the IDR-polynomial Φ_n can be explicitly written as a product of two polynomials:

$$\Phi_n(\boldsymbol{A}) = \Omega_j(\boldsymbol{A})\Psi_{n-j}(\boldsymbol{A}) \tag{2}$$

where

$$\Omega_j(t) = \prod_{k=1}^{j}(1-\omega_k t), \quad \Psi_{n-j}(t) = 1 - \sum_{l=1}^{n-j} c_l t^l \tag{3}$$

The choice $c_0 = 1$ is required for the possibility to construct iterates $\boldsymbol{x}_n$ that satisfy $\boldsymbol{r}_n = \boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_n$.

We call the factors $\Omega_j(\boldsymbol{A})$ *damping factors* or *stabilization factors*, and the polynomials $\Psi_{n-j}(\boldsymbol{A})$ *Lanczos factors*.

The damping factors have their name because the coefficients ω_j are usually calculated with the purpose of minimizing the norm of $\boldsymbol{r}_{j(s+1)} = (\boldsymbol{I} - \omega_j \boldsymbol{A})\boldsymbol{s}$ for some vector $\boldsymbol{s}$ that arises in the algorithm at that stage.

It is plausible to expect that the matrix polynomial $\Omega_j(\boldsymbol{A})$ will act as a contraction, but this is not allways the case. In many cases however, the damping-factors in the residuals

are at least partly responsible for the convergence. So we remain calling them 'damping factors' or 'stabilization factors', even if they do not damp or stabilize at all.
The name *Lanczos factors* for the polynomials $\Psi_{n-j}(\boldsymbol{A})$ is chosen because the polynomials Ψ_{n-j} have some similarity with the Lanczos polynomials.

We define the *reduced residuals* $\widetilde{\boldsymbol{r}}_{n-j}$ by

$$\widetilde{\boldsymbol{r}}_{n-j} = \Psi_{n-j}(\boldsymbol{A})\boldsymbol{r}_0 \tag{4}$$

Plots for the norms of these reduced residuals show the contribution of the Lanczos factors to the convergence of IDR(s). Unfortunately the reduced residuals are not explicitly present in the IDR(s) algorithm. However, in the variant of the algorithm that is presented in [7], it is possible to calculate them by applying a shadow process, using the calculated coefficients of the algorithm, but skipping the explicit multiplication with $\boldsymbol{I} - \omega_j \boldsymbol{A}$ at step $j(s+1)$.
Note that this skipping procedure leads to the same reduction by a factor $\frac{s}{s+1}$ as are applied in the normalization of convergence plots. This is quite natural after all, since for $n = j(s+1)$:

$$\frac{\text{degree of } \Phi_n}{\text{degree of } \Psi_{n-j}} = \frac{\text{degree of } \Phi_{j(s+1)}}{\text{degree of } \Psi_{js}} = \frac{s+1}{s}$$

2.2 Experiments on damping- and Lanczos parts.

We can view the contributions of the 'Lanczos part', and the 'damping part' of the residuals, by comparing the plots of the reduced residuals with the normalized plots of IDR(s) residuals.
In Figure 3 we compare the normalized IDR(s) with the reduced (= undamped) residuals. It is quite clear that IDR(1) has rather large profit from the damping part of the convergence. For larger s-values this effect becomes smaller.
Problem 2, on which IDR(s) is applied in Figure 3 represents a case in which the damping factors actually produce damping. For Figure 4, the same convection diffusion equation as in Problem 2 is used, but with a large mesh-Peclet numbers $[20, 0]$ (Problem 3). For classical iterative procedures this is disastrous, and the same effect can be concluded from Figure 4. Upper and lower plots in Figure 4 show no damping at all, and one would be better off in using the reduced residual. But producing this requires a lot of extra computational work.
The cause of the failing damping property is that the calculated real values for ω_j would be very small, causing *stagnation* of the procedure (as in Bi-CGSTAB). Therefore a modified calculation of ω_j is done, described in [13], which in these circumstances may cause *growth* instead of a damping.
So it is quite reasonable to look seriously for alternative criteria for choosing the parameters ω_j. For the case $s = 1$, which is equivalent to Bi-CGSTAB, Gutknecht in [6], and Sleijpen and Fokkema in [11] have developed look-ahead -like variants of the algorithm, in which the factors $\boldsymbol{I} - \omega_j \boldsymbol{A}$ are combined into higher degree polynomials, allowing for actual minimization in this kind of cases. Recently, Simoncini and Szyld in [10] developed alternative criteria for the ω-values, based on the field of values of $\boldsymbol{A}$.
One extremely easy way to handle the non-damping problem in IDR(s), is shown in [15]. By choosing the matrix $\boldsymbol{P}$ complex, the algorithm is forced to use complex arithmetic, and therefore may find better factors. The result is shown in the upper and lower plots of Figure 5. Apart from the increase of computational work caused by complex arithmetic, this seems to be a perfect remedy.

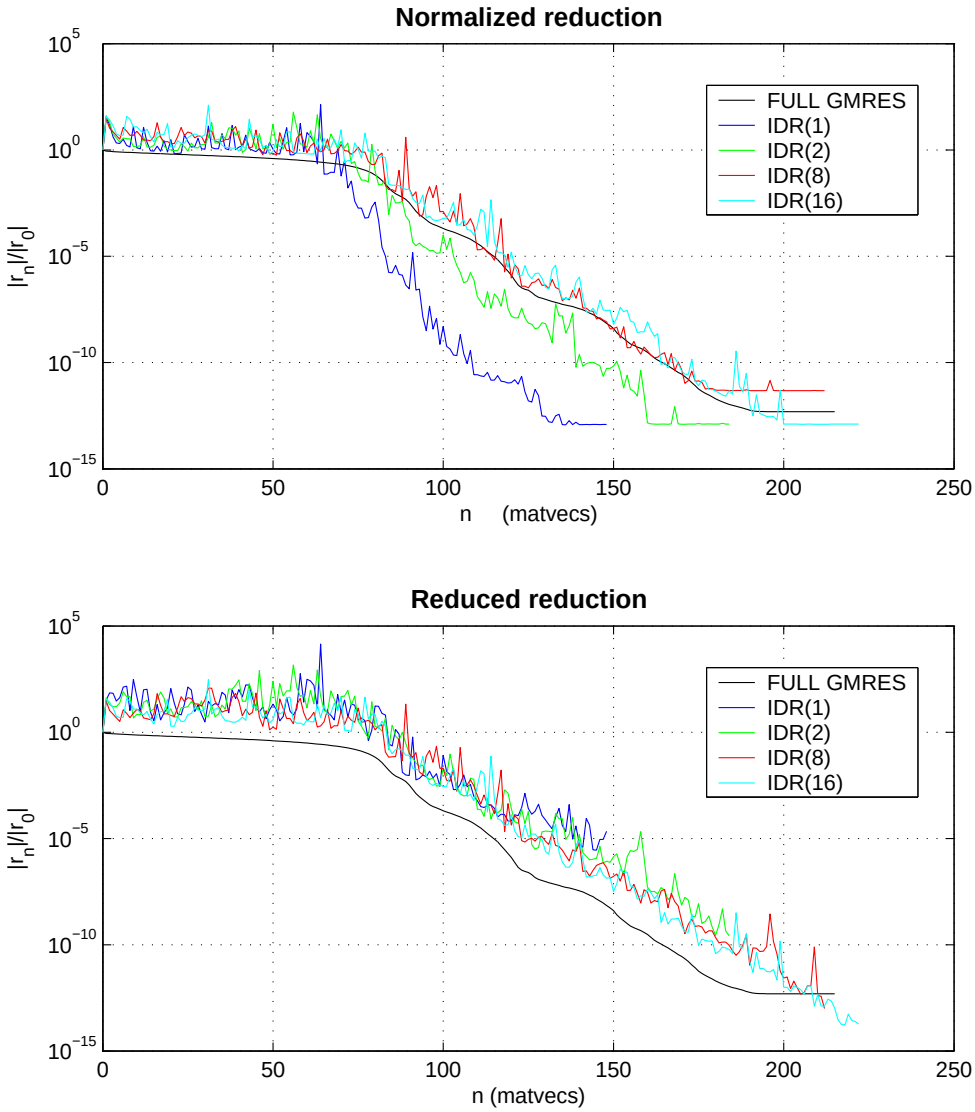

Figure 3: Normalized versus reduced residuals in Problem 2

The lower plot in Figure 3 and the lower plot in Figure 5 show an interesting property of the reduced residuals. They follow the GMRES-behaviour quite closely in the observed examples. This behaviour is a property of the Lanczos part of the residuals. This will be analysed in the next sections.

3 Analysis of the Lanczos factors.

It is shown in [15] that the polynomials Ψ_{n-j} satisfy relations of the following type:

$$\boldsymbol{p}_r^* \boldsymbol{A}^l \Psi_{n-j}(\boldsymbol{A}) \boldsymbol{r}_0 = 0, \;\; l = 0, 1, \ldots, j-1, \;\; r = 1, 2, \ldots, s \tag{5}$$

where s is the 'order' of the IDR(s) algorithm. Since Ψ_{n-j} has $n-j$ coefficients to be determined, the above relations can be consistent as long $n - j \geq s * j$. If the equality sign holds, the coefficients are determined uniquely, which is the case if $n = (s+1) * j$, corresponding exactly with the calculation of the very first residual in the space $\mathcal{G}_j$. In

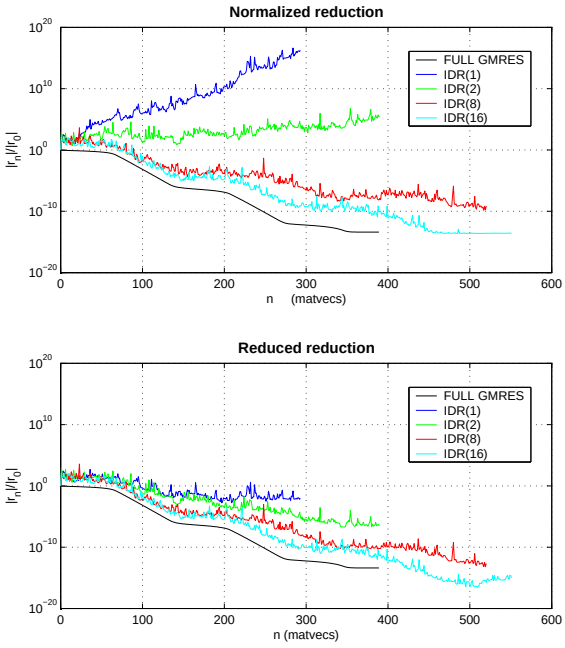

Figure 4: No damping in Problem 3.

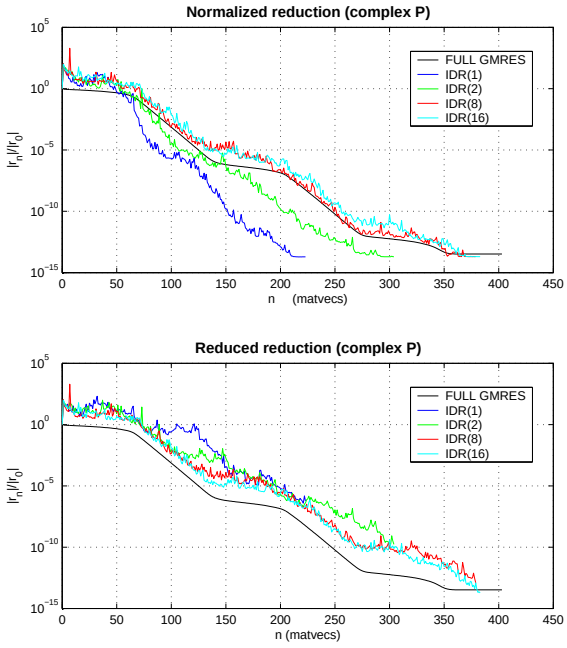

Figure 5: Remedy by complex $\boldsymbol{P}$

the calculatuions for the residuals $\boldsymbol{r}_n$ for intermediate values of n, there is freedom, which can be used for stability reasons.
One important conclusion can be drawn from the relations in (5): the Lanczos factor Ψ_{n-j} *is independent from the damping polynomial* Ω_j. So the Lanczos part of the convergence is not influenced by the damping strategy, at least as long finite precision issues are not taken into account.

Define the Krylov vectors $\boldsymbol{k}_l$, and the reduced Krylov matrices $\boldsymbol{K}_l$ by

$$\boldsymbol{k}_l = \boldsymbol{A}^l \boldsymbol{r}_0,\ l = 0, 1, 2, \ldots, \quad \boldsymbol{K}_l = (\boldsymbol{k}_1\ \boldsymbol{k}_2\ \ldots\ \boldsymbol{k}_l),\ l = 1, 2, \ldots \tag{6}$$

Then the equations for the coefficients c_l read explicitly:

$$\boldsymbol{p}_r^* \boldsymbol{A}^l [\boldsymbol{K}_{n-j} \boldsymbol{c} - \boldsymbol{r}_0] = 0,\ r = 1, 2, \ldots, s,\ l = 0, 2, \ldots, j-1$$

These relations can be written in the following form:

$$\boldsymbol{T}^* \boldsymbol{K}_{n-j} \boldsymbol{c} = \boldsymbol{T}^* \boldsymbol{r}_0 \tag{7}$$

where

$$\boldsymbol{T} = (\boldsymbol{t}_1\, \boldsymbol{t}_2\, \ldots\, \boldsymbol{t}_{sj}), \quad , \boldsymbol{t}_{ls+r} = (\boldsymbol{A}^*)^l \boldsymbol{p}_r,\ l = 0, 1, \ldots, j-1,\ r = 1, 2, \ldots, s$$

The vectors $\boldsymbol{t}_i$ can be considered as *test vectors*, and the matrix $\boldsymbol{T}$ as a *test matrix*, in a *Galerkin context*. In fact, the reduced residuals can be regarded as *Galerkin residuals*, produced by a Galerkin approximation of the overdetermined system

$$\widetilde{\boldsymbol{r}}_{n-j} = \boldsymbol{K}_{n-j} \boldsymbol{c},\ \text{with}\ \boldsymbol{K}_{n-j} \boldsymbol{c} = \boldsymbol{r}_0 \tag{8}$$

The interpretation described here is restricted to the case $n = (s+1) * j$, and is valid for every variant of IDR(s). In the variant described in [7], the inner freedom in choice for

$(s+1)j < n < (s+1)(j+1)$ is used in such a way that the Galerkin connection holds for all n.
The construction of the IDR(s) residual $\boldsymbol{r}_{j(s+1)+k}$ requires $j(s+1)+k$ matrix vector multiplications. Within the context of calculating a Galerkin approximation, with a 'Galerkin dimension' $n-j$, about j of these matrix vector products can be considered as 'overhead'. This overhead may be paid back partially by the (possible) contraction properties of the damping factors $\Omega_j(\boldsymbol{A})$.

We call the procedure defined by (6) and (8) *Krylov-Galerkin approximation*.

3.1 Galerkin versus least squares.

The following analysis applies to any Galerkin procedure for finite linear problems.
In classical least squares problems, the objective is to find a best fit of a certain linear 'model' to a set of measured data. Let there be given N measured data b_j, and let the model be represented by the $N \times k$ model matrix $\boldsymbol{M}$, then we have to minimize the 2-norm of $\boldsymbol{b} - \boldsymbol{Mc}$. The residual $\boldsymbol{r} = \boldsymbol{b} - \boldsymbol{Mc}$ is then called the model error for the least squares approximation.
The residual is minimized if it is orthogonal to the range of $\boldsymbol{M}$:

$$\|\boldsymbol{Mx} - \boldsymbol{b}\| \text{ is minimal} \Leftrightarrow \boldsymbol{M}^*(\boldsymbol{Mx} - \boldsymbol{b}) = \boldsymbol{0}$$

The same solution will be obtained by requiring $\widetilde{\boldsymbol{M}}^*(\boldsymbol{Mc} - \boldsymbol{b}) = \boldsymbol{0}$, for any $N \times k$ matrix $\widetilde{\boldsymbol{M}}$ sharing its range with $\boldsymbol{M}$. In general all matrices satisfying this requirement can be written as $\widetilde{\boldsymbol{M}} = \boldsymbol{MC}$, with $\boldsymbol{C}$ a $k \times k$ nonsingular matrix.
If we replace $\widetilde{\boldsymbol{M}}$ by a 'testmatrix' $\boldsymbol{T}$, of which the range differs from $\mathcal{R}(\boldsymbol{M})$, we call the approximation a *Galerkin approximaton*. It is obtained by solving $\boldsymbol{T}^*(\boldsymbol{Mc} - \boldsymbol{b}) = \boldsymbol{0}$. The Galerkin approximation equals the least squares solution if $\mathcal{R}(\boldsymbol{T}) = \mathcal{R}(\boldsymbol{M})$.

Write down the Galerkin approximation of the model coefficients explicitly:

$$\boldsymbol{c} = (\boldsymbol{T}^*\boldsymbol{M})^{-1}\boldsymbol{T}^*\boldsymbol{b}$$

then the model error satisfies

$$\boldsymbol{b} - \boldsymbol{Mc} = \boldsymbol{b} - \boldsymbol{M}(\boldsymbol{T}^*\boldsymbol{M})^{-1}\boldsymbol{T}^*\boldsymbol{b} = \left(\boldsymbol{I} - \boldsymbol{M}(\boldsymbol{T}^*\boldsymbol{M})^{-1}\boldsymbol{T}^*\right)\boldsymbol{b} = (\boldsymbol{I} - \boldsymbol{P})\boldsymbol{b}$$

where $\boldsymbol{P} = \boldsymbol{M}(\boldsymbol{T}^*\boldsymbol{M})^{-1}\boldsymbol{T}^*$ represents a, generally oblique, projection on $\mathcal{R}(\boldsymbol{M})$ (Obviously $\boldsymbol{P}^2 = \boldsymbol{P}$, so this is a projection allright).
The quality of a Galerkin approximation can be measured by the (norm of the) residual $\boldsymbol{r} = (\boldsymbol{I} - \boldsymbol{P})\boldsymbol{b}$. The matrix $\boldsymbol{I} - \boldsymbol{P}$ also represents a projection, and by inspection we find $\mathcal{R}(\boldsymbol{I} - \boldsymbol{P}) = \mathcal{N}(\boldsymbol{T}^*)$.

A priory statements on $\|\boldsymbol{r}\|$ in Galerkin approximations are rather restricted. In contrast with least squares approximations, there is no positive quantity to be minimized. Only if the test matrix has certain connections with the model matrix, there might be some possibility for an analysis. On the other hand, relations between the test matrix and the model matrix can also turn out to be disastrous for the quality of the approximation.
A simple example of a bad choice of test space is the case that $\boldsymbol{T}^*\boldsymbol{M}$ is (nearly) singular. If $\boldsymbol{M}$ is a proper model matrix, its columns are linearly independent (otherwise we would drop a dependent column, since it does not contribute to the model). So (near) singularity of the Galerkin equations are caused by a bad match between model space and test space.

Consider the case that $\boldsymbol{T}^*\boldsymbol{M}$ is singular. Let $\boldsymbol{y}$ be in the left null space of $\boldsymbol{T}^*\boldsymbol{M}$, then

$$\boldsymbol{y}^*\boldsymbol{T}^*\boldsymbol{M} = \boldsymbol{0}^* \Longrightarrow (\boldsymbol{T}\boldsymbol{y})^*(\boldsymbol{M}\boldsymbol{x}) = 0 \ \forall \boldsymbol{x} \Longrightarrow \boldsymbol{T}\boldsymbol{y} \perp \mathcal{R}(\boldsymbol{M})$$

So the test space contains a vector perpendicular to the whole model space.
Similarly, in the case that $\boldsymbol{T}^*\boldsymbol{M}$ has a bad condition (is nearly singular), it may be expected that the test space contains a vector that is *nearly* perpendicular to the model space.

These heuristic considerations will be quantified now. We compare the Galerkin approximation with the least squares method, by comparing the corresponding projection operators.
The following lemma reveals a remarkable property of projection operators with the same range.

Lemma 3.1 *Let $\boldsymbol{P}_1$ and $\boldsymbol{P}_2$ be projections, and let $\mathcal{R}(\boldsymbol{P}_1) = \mathcal{R}(\boldsymbol{P}_2)$. Then*
(i) $\boldsymbol{P}_1\boldsymbol{P}_2 = \boldsymbol{P}_2$.
(ii) $(\boldsymbol{I} - \boldsymbol{P}_1)(\boldsymbol{I} - \boldsymbol{P}_2) = \boldsymbol{I} - \boldsymbol{P}_1$

Proof: The proof is simple: Let $\boldsymbol{x}$ be arbitrary, let $\boldsymbol{y} = \boldsymbol{P}_2\boldsymbol{x}$, then $\boldsymbol{y} \in \mathcal{R}(\boldsymbol{P}_2)$, and since $\mathcal{R}(\boldsymbol{P}_1) = \mathcal{R}(\boldsymbol{P}_2)$, $\boldsymbol{y} = \boldsymbol{P}_1\widetilde{\boldsymbol{x}}$, for some $\widetilde{\boldsymbol{x}}$. Hence $(\boldsymbol{P}_1\boldsymbol{P}_2)\boldsymbol{x} = \boldsymbol{P}_1\boldsymbol{P}_1\widetilde{\boldsymbol{x}} = \boldsymbol{P}_1\widetilde{\boldsymbol{x}} = \boldsymbol{y} = \boldsymbol{P}_2\boldsymbol{x}$. This being true for every $\boldsymbol{x}$, property (i) follows. Property (ii) follows directly:

$$(\boldsymbol{I} - \boldsymbol{P}_1)(\boldsymbol{I} - \boldsymbol{P}_2) = \boldsymbol{I} - \boldsymbol{P}_1 - \boldsymbol{P}_2 + \boldsymbol{P}_1\boldsymbol{P}_2 = \boldsymbol{I} - \boldsymbol{P}_1$$

$\triangle$

We apply this to the expression for the residual of a Galerkin approximation. Let $\boldsymbol{P}$ be the Galerkin projection for the model matrix $\boldsymbol{M}$ and the test matrix $\boldsymbol{T}$, and let $\widehat{\boldsymbol{P}}$ denote the least squares projection for the model $\boldsymbol{M}$. Then $\boldsymbol{P}$ and $\widehat{\boldsymbol{P}}$ share the column space $\mathcal{R}(\boldsymbol{M})$. Hence $(\boldsymbol{I} - \boldsymbol{P})(\boldsymbol{I} - \widehat{\boldsymbol{P}}) = \boldsymbol{I} - \boldsymbol{P}$, and we may write

$$\boldsymbol{r} = (\boldsymbol{I} - \boldsymbol{P})\boldsymbol{b} = (\boldsymbol{I} - \boldsymbol{P})(\boldsymbol{I} - \widehat{\boldsymbol{P}})\boldsymbol{b} = (\boldsymbol{I} - \boldsymbol{P})\widehat{\boldsymbol{r}} = \widehat{\boldsymbol{r}} - \boldsymbol{P}\widehat{\boldsymbol{r}}$$

where $\widehat{\boldsymbol{r}}$ is the least squares residual. Now $\widehat{\boldsymbol{r}} \in \mathcal{R}(\boldsymbol{M})^\perp$, and $\boldsymbol{P}\widehat{\boldsymbol{r}} \in \mathcal{R}(\boldsymbol{P}) = \mathcal{R}(\boldsymbol{M})$, so the Galerkin residual is divided into two mutually orthogonal components $\widehat{\boldsymbol{r}}$ and $\boldsymbol{P}\widehat{\boldsymbol{r}}$. For convenience we analyse the *residual surplus*

$$d\boldsymbol{r} = \boldsymbol{r} - \widehat{\boldsymbol{r}} = \boldsymbol{P}\widehat{\boldsymbol{r}} \Longrightarrow \|\boldsymbol{r}\| = \sqrt{\|\widehat{\boldsymbol{r}}\|^2 + \|d\boldsymbol{r}\|^2} \tag{9}$$

Let $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ be matrices of which the columns build orthonormal bases for $\mathcal{R}(\boldsymbol{M})$ and $\mathcal{R}(\boldsymbol{M})^\perp$ respectively. Then $\boldsymbol{P} = \boldsymbol{Q}_1(\boldsymbol{T}^*\boldsymbol{Q}_1)\boldsymbol{T}^*$, and the least squares residual can be written as $\widehat{\boldsymbol{r}} = \boldsymbol{Q}_2\boldsymbol{s}$, where $\boldsymbol{s}$ may be any vector in $\mathbb{R}^{N-k}$ satisfying $\|\boldsymbol{s}\| = \|\widehat{\boldsymbol{r}}\|$. Then the residual surplus can be written as

$$d\boldsymbol{r} = \boldsymbol{Q}_1(\boldsymbol{T}^*\boldsymbol{Q}_1)^{-1}\boldsymbol{T}^*\boldsymbol{Q}_2\boldsymbol{s} \tag{10}$$

from which follows, since $\|\boldsymbol{Q}_1\boldsymbol{v}\| = \|\boldsymbol{v}\|$ for every $\boldsymbol{v}$:

$$\|d\boldsymbol{r}\| = \|(\boldsymbol{T}^*\boldsymbol{Q}_1)^{-1}\boldsymbol{T}^*\boldsymbol{Q}_2\boldsymbol{s}\| \leq \|(\boldsymbol{T}^*\boldsymbol{Q}_1)^{-1}\|.\|\boldsymbol{T}^*\boldsymbol{Q}_2\|.\|\widehat{\boldsymbol{r}}\| \tag{11}$$

We can also describe the test space $\mathcal{R}(\boldsymbol{T})$ by an orthonormal basis, for instance by making a QR-factorisation of $\boldsymbol{T}$: $\boldsymbol{T} = \widetilde{\boldsymbol{Q}}\boldsymbol{R}$, with $\widetilde{\boldsymbol{Q}}^*\widetilde{\boldsymbol{Q}} = \boldsymbol{I}$. Then (11) turns over into

$$\|d\boldsymbol{r}\| \leq \|(\widetilde{\boldsymbol{Q}}^*\boldsymbol{Q}_1)^{-1}\|.\|\widetilde{\boldsymbol{Q}}^*\boldsymbol{Q}_2\|.\|\widehat{\boldsymbol{r}}\| \tag{12}$$

Now the ranges of $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ together span the complete space, hence the matrix $\boldsymbol{Q} = (\boldsymbol{Q}_1\ \boldsymbol{Q}_2)$ is unitary. Therefore

$$\boldsymbol{Q}_1\boldsymbol{Q}_1^* + \boldsymbol{Q}_2\boldsymbol{Q}_2^* = \boldsymbol{I}$$

Define

$$\boldsymbol{B}_1 = \widetilde{\boldsymbol{Q}}^*\boldsymbol{Q}_1, \quad \boldsymbol{B}_2 = \widetilde{\boldsymbol{Q}}^*\boldsymbol{Q}_2 \quad \boldsymbol{B} = (\boldsymbol{B}_1\ \boldsymbol{B}_2) = \widetilde{\boldsymbol{Q}}^*\boldsymbol{Q}$$

then

$$\boldsymbol{B}_1\boldsymbol{B}_1^* + \boldsymbol{B}_2\boldsymbol{B}_2^* = \boldsymbol{B}\boldsymbol{B}^* = (\widetilde{\boldsymbol{Q}}^*\boldsymbol{Q})(\boldsymbol{Q}^*\widetilde{\boldsymbol{Q}}) = \widetilde{\boldsymbol{Q}}^*\widetilde{\boldsymbol{Q}} = \boldsymbol{I} \tag{13}$$

The estimate (12) for $\|d\boldsymbol{r}\|$ can be written in the following simple form:

$$\|d\boldsymbol{r}\| \leq \|\boldsymbol{B}_1^{-1}\|.\|\boldsymbol{B}_2\|.\|\widehat{\boldsymbol{r}}\| \tag{14}$$

Let $\boldsymbol{C}$ be any matrix, then

$$\|\boldsymbol{C}\| = \sqrt{\lambda_{\max}(\boldsymbol{C}\boldsymbol{C}^*)}, \quad \text{and when } \boldsymbol{C} \text{ is square: } \|\boldsymbol{C}^{-1}\| = \frac{1}{\sqrt{\lambda_{\min}(\boldsymbol{C}\boldsymbol{C}^*)}}$$

Apply this to $\boldsymbol{B}_1$ and $\boldsymbol{B}_2$, and bearing in mind that $\boldsymbol{B}_2\boldsymbol{B}_2^* = \boldsymbol{I} - \boldsymbol{B}_1\boldsymbol{B}_1^*$, we get

$$\|\boldsymbol{B}_1^{-1}\|^2 = \lambda_{\min}(\boldsymbol{B}_1\boldsymbol{B}_1^*)^{-1}, \quad \|\boldsymbol{B}_2\|^2 = \lambda_{\max}(\boldsymbol{B}_2\boldsymbol{B}_2^*) = 1 - \lambda_{\min}(\boldsymbol{B}_1\boldsymbol{B}_1^*)$$

From which follows

$$\|d\boldsymbol{r}\| \leq \|\boldsymbol{B}_1^{-1}\|.\|\boldsymbol{B}_2\|.\|\widehat{\boldsymbol{r}}\| = \sqrt{\frac{1-c}{c}}\|\widehat{\boldsymbol{r}}\| \tag{15}$$

where $c = \lambda_{\min}(\boldsymbol{B}_1\boldsymbol{B}_1^*)$.
The eigenvalues of $\boldsymbol{B}_1\boldsymbol{B}_1^*$ are the squares of the singular values σ_j of $\boldsymbol{B}_1$. Let $\sigma_{\min}$ be the least singular value, then $\sigma_{\min} = \sqrt{\lambda_{\min}(\boldsymbol{B}_1\boldsymbol{B}_1^*)}$, and (15) can be written as

$$\|d\boldsymbol{r}\| \leq \frac{\sqrt{1-{\sigma_{\min}}^2}}{\sigma_{\min}}\|\widehat{\boldsymbol{r}}\|$$

This last formula suggests the use of trigonnometric functions. We can define a positive angle ϑ, such that $\sigma_{\min} = \cos(\vartheta)$. Then

$$\|d\boldsymbol{r}\| \leq \tan(\vartheta)\|\widehat{\boldsymbol{r}}\| \tag{16}$$

The angle ϑ has a geometrical interpretation with respect to the model space and the test space. We give a short description.
The angle $\angle(\boldsymbol{x}, \mathcal{V})$ between a vector $\boldsymbol{x}$ and a subspace $\mathcal{V}$ can be defined as the *minimum positive angle* between $\boldsymbol{x}$ and any nonzero $\boldsymbol{y} \in \mathcal{V}$:

$$\angle(\boldsymbol{x}, \mathcal{V}) = \min_{yv \in \mathcal{V}} \angle(\boldsymbol{x}, \boldsymbol{y})$$

where the angle between two vectors has the usual meaning.
Similarly the angle $\angle(\mathcal{U}, \mathcal{V})$ between two subspaces $\mathcal{U}$ and $\mathcal{V}$ can be defined as the *maximum angle between* $\boldsymbol{x}$ *and* $\mathcal{V}$ for any nonzero $\boldsymbol{x} \in \mathcal{U}$:

$$\angle(\mathcal{U}, \mathcal{V}) = \max_{\boldsymbol{x} \in \mathcal{U}} \angle(\boldsymbol{x}, \mathcal{V})$$

Using cosines instead of angles for computational convenience, and bearing in mind that the cosine function is decreasing on $(0, \frac{1}{2}\pi)$, this can be expressed by

$$\cos(\angle(\mathcal{U},\mathcal{V})) = \min_{\boldsymbol{x}\in\mathcal{U}} \left(\max_{\boldsymbol{y}\in\mathcal{V}} \frac{\boldsymbol{x}^*\boldsymbol{y}}{\|\boldsymbol{x}\|.\|\boldsymbol{y}\|} \right)$$

Let $\mathcal{U}$ and $\mathcal{V}$ have orthonormal bases, defined by matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ respectively, and let $\boldsymbol{x} = \boldsymbol{U}\widetilde{\boldsymbol{x}}$, and $\boldsymbol{y} = \boldsymbol{V}\widetilde{\boldsymbol{y}}$, with $\|\widetilde{\boldsymbol{x}}\| = \|\boldsymbol{x}\|$ and $\|\widetilde{\boldsymbol{y}}\| = \|\boldsymbol{y}\|$. Then

$$\cos(\angle(\mathcal{U},\mathcal{V})) = \min_{\widetilde{\boldsymbol{x}}} \left(\max_{\widetilde{\boldsymbol{y}}} \frac{\widetilde{\boldsymbol{x}}^*\boldsymbol{U}^*\boldsymbol{V}\widetilde{\boldsymbol{y}}}{\|\widetilde{\boldsymbol{x}}\|.\|\widetilde{\boldsymbol{y}}\|} \right)$$

Now according to the Cauchy Schwartz inequality we have

$$\frac{|\widetilde{\boldsymbol{x}}^*\boldsymbol{U}^*\boldsymbol{V}\widetilde{\boldsymbol{y}}|}{\|\widetilde{\boldsymbol{x}}\|.\|\widetilde{\boldsymbol{y}}\|} \leq \frac{\|\boldsymbol{V}^*\boldsymbol{U}\widetilde{\boldsymbol{x}}\|.\|\widetilde{\boldsymbol{y}}\|}{\|\widetilde{\boldsymbol{x}}\|.\|\widetilde{\boldsymbol{y}}\|} = \frac{\|\boldsymbol{V}^*\boldsymbol{U}\widetilde{\boldsymbol{x}}\|}{\|\widetilde{\boldsymbol{x}}\|}$$

where the equality sign holds for $\widetilde{\boldsymbol{y}} = \boldsymbol{V}^*\boldsymbol{U}\widetilde{\boldsymbol{x}}$. Hence we have

$$\cos(\angle(\mathcal{U},\mathcal{V})) = \min_{\widetilde{\boldsymbol{x}}} \frac{\|\boldsymbol{V}^*\boldsymbol{U}\widetilde{\boldsymbol{x}}\|}{\|\widetilde{\boldsymbol{x}}\|} = \frac{1}{\|(\boldsymbol{V}^*\boldsymbol{U})^{-1}\|} = \sigma_{\min}(\boldsymbol{V}^*\boldsymbol{U})$$

In estimate (16), we have $\cos(\vartheta) = \sigma_{\min}(\boldsymbol{B}_1) = \sigma_{\min}(\widetilde{\boldsymbol{Q}}^*\boldsymbol{Q}_1)$. *Therefore ϑ in (16) represents the angle between the model space and the test space.*

Obviously, Galerkin can be in trouble if ϑ is close to $\pi/2$, i.e. if some testvectors are nearly perpendicular to the model space. On the other hand, if ϑ is small, the estimate (16) garantees that $\|\boldsymbol{r} - \widehat{\boldsymbol{r}}\|$ is small compared to $\|\widehat{\boldsymbol{r}}\|$.

Similar results can be found in [3], in which comparisons of *orthogonal residual methods* with *minimal residual methods* are made. In [8], a similar analysis is done to find estimates for the convergence of BiCG, QMR, FOM, and GMRES.

We shall apply the framework differently, because the test matrix in the Krylov Galerkin procedure inside the IDR(s) algorithm has a completely different background. Instead of being built up from images $(\boldsymbol{A}^*)^n\widetilde{\boldsymbol{r}}_0$ of one shadow residual, the sequence of test vectors is defined by $\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_s, \boldsymbol{A}^*\boldsymbol{p}_1, \boldsymbol{A}^*\boldsymbol{p}_2, \ldots, \boldsymbol{A}^*\boldsymbol{p}_s, (\boldsymbol{A}^*)^2\boldsymbol{p}_1, \ldots$. So the amount of $\boldsymbol{A}^*$-influence in the test matrices is far less than in Bi-CGSTAB-like methods.

4 Random testvectors.

In the case of Krylov-based iterative methods, the model matrix in the Galerkin interpretation is $\boldsymbol{AK}$, where the columns of $\boldsymbol{K}$ contain a basis for a Krylov subspace. The choice $\boldsymbol{T} = \boldsymbol{M}$, for which choice the GMRES method is a very efficient and stable implementation, is the best choice with respect to the residual norm. The only disadvantage is the fact that the recurrence relations in the algorithm grow in depth with the iteration count. So after numerous steps, the overhead in linear combinations and inner product calculations becomes a bottleneck with respect to both storage and computing time.
In de IDR(s) algorithms as presented in [15], the s shadow vectors are chosen randomly. In the prototype code, every component of every vector is chosen independently uniformly distributed in the interval $[0, 1]$. This choice however does not provide true random vectors in $\boldsymbol{C}^N$, because the density of vectors in directions like $[1, 1, \ldots, 1]^T$ is considerably higher that in directions like $[1, 0, 0 \ldots, 0]^T$

Since the algorithm is invariant for transformations like $\widehat{\boldsymbol{P}} = \boldsymbol{PC}$ for any invertible $s \times s$ matrix $\boldsymbol{C}$, the length of the vectors $\boldsymbol{p}_k$ is not of importance. Now we want the vectors uniformly distributed over the *directions*, so if we normalize them, we want the results to be uniformly distributed over the unit sphere.
An excellent way to do this is: Choose vectors of which all components are independent identically-distributed normally with zero mean and unit variance:

$$\boldsymbol{P} \overset{\text{iid}}{\sim} N(0,1)^{N\times s}$$

Since the simultaneous distribution for each random vector is invariant for unitary transformations, the normed vectors are uniformly distributed over the unit sphere. Since the scale of vectors is not relevant, we have chosen unit variance.
For simplicity and readability, we define the class $\mathcal{N}$ as the set of stochastic variables normally distributed with zero mean and unit variance.

$$x \in \mathcal{N} \mapsto x \overset{\text{iid}}{\sim} N(0,1)$$

Then by $\mathcal{N}^k$ and $\mathcal{N}^{k\times l}$ we mean classes of vectors resp. matrices, of wich all entries are in $\mathcal{N}$, and are mutually stochastically independent.
In the practical IDR(s) algorithms, the Galerking testmatrices are build from the columns

$$\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_s, \boldsymbol{A}^*\boldsymbol{p}_1, \boldsymbol{A}^*\boldsymbol{p}_2, \ldots, (\boldsymbol{A}^*)^j\boldsymbol{p}_s$$

These columns are not stochastically independent for $k > s$, and therefore not easily to be analysed. In the numerical tests of IDR(s), the convergence curves were shifted to the left at increasing s, with a virtual limiting position close to the full-GMRES curve, which is the optimal curve from the 'number of matrix-vector operation' point of view.
As a first attempt for analysing the role of random shadow vectors we choose a case where s is greater than the number of iterations. In this case we might expect to get a convergence curve that is to the left of all practical IDR(s) curves. Indeed, some experiments show such behaviour. We refer to this – highly unpractical – method as 'Full Random Galerkin Krylov' algorithm.

So we next assume $\boldsymbol{T} \in \mathcal{N}^{N\times s}$. Then $d\boldsymbol{r} = \boldsymbol{r} - \widehat{\boldsymbol{r}}$ is a stochastic vector, and its norm is a stochastic variable $v = \|\boldsymbol{r} - \widehat{\boldsymbol{r}}\|$. We try to figure out properties for v. According to (10), v satisfies

$$v = \|(\boldsymbol{T}^*\boldsymbol{Q}_1)^{-1}\boldsymbol{T}^*\boldsymbol{Q}_2\boldsymbol{s}\|$$

where $\boldsymbol{s}$ is a vector which is completely determined by the least squares approximation, and satisfies $\|\boldsymbol{s}\| = \|\widehat{\boldsymbol{r}}\|$.
If we would replace the random testmatrix $\boldsymbol{T}$ by the '$\widetilde{\boldsymbol{Q}}$' from its QR factorisation, the matrix $\widetilde{\boldsymbol{Q}}$ is not in $\mathcal{N}^{N\times s}$, and therefore cannot easily to be analysed. For this reason the analysis as done in Section 3.1, with angles between subspaces, is not appropriate at first sight. However, we can base an analysis on a property of $\mathcal{N}$-distributed vectors.
Let $\boldsymbol{Q}$ be a unitary $N \times N$ matrix, let $\boldsymbol{v} \in \mathcal{N}^N$, and let $\widetilde{\boldsymbol{v}} = \boldsymbol{Q}^*\boldsymbol{v}$, then according to an elementary law of statistics $\widetilde{\boldsymbol{v}} \in \mathcal{N}^N$ as well.
Applying $\boldsymbol{Q}$ on $\boldsymbol{T}$, all columns of $\boldsymbol{Q}^*\boldsymbol{T}$ are in $\mathcal{N}^N$, and since the columns of $\boldsymbol{T}$ are stochastically independent, also the images are, hence

$$\widetilde{\boldsymbol{T}} = \boldsymbol{Q}^*\boldsymbol{T} \in \mathcal{N}^{N\times k}$$

Now write $\widetilde{\boldsymbol{T}}_1 = \boldsymbol{Q}_1^* \boldsymbol{T}$, $\widetilde{\boldsymbol{T}}_2 = \boldsymbol{Q}_2^* \boldsymbol{T}$, then

$$\widetilde{\boldsymbol{T}}^* = \boldsymbol{T}^* \boldsymbol{Q} = \boldsymbol{T}^* [\boldsymbol{Q}_1 \,|\, \boldsymbol{Q}_2] = [\widetilde{\boldsymbol{T}}_1^* \,|\, \widetilde{\boldsymbol{T}}_2^*]$$

Now the stochastic variable v satisfies

$$v = \|(\widetilde{\boldsymbol{T}}_1^*)^{-1} \widetilde{\boldsymbol{T}}_2^* \boldsymbol{s}\|$$

where $\widetilde{\boldsymbol{T}}_1 \in \mathcal{N}^{k \times k}$, and $\widetilde{\boldsymbol{T}}_2^* \in \mathcal{N}^{k \times (N-k)}$.

Let $\boldsymbol{y} = \widetilde{\boldsymbol{T}}_2^* \boldsymbol{s}$, then

$$y_j = \sum_{l=1}^{N-k} \overline{\widetilde{t}_{l,j}} s_l$$

This is a stochastic variable, distributed $N(0, \|\boldsymbol{s}\|)$, since the entries of j-th column of $\widetilde{\boldsymbol{T}}_2$ are stochastically independent standard normally distributed variables. This holds for all $j = 1, 2, \ldots, k$, and since *all k columns of $\widetilde{\boldsymbol{T}}_2$ are stochastically independent*, the variables y_j are also stochastically independent. Hence the vector $\boldsymbol{y}$ is a stochastic vector, distributed $\overset{\text{iid}}{\sim} \mathcal{N}(0, \|\boldsymbol{s}\|)^k$, and we may write:

$$\boldsymbol{y} = \|\boldsymbol{s}\| \boldsymbol{z}, \text{ with } \boldsymbol{z} \in \mathcal{N}^k$$

Therefore $d\boldsymbol{r} = \boldsymbol{r} - \widehat{\boldsymbol{r}}$ can be written as

$$d\boldsymbol{r} = \widetilde{\boldsymbol{T}}_1^{-H} \boldsymbol{y} = \|\widehat{\boldsymbol{r}}\| . \widetilde{\boldsymbol{T}}_1^{-1} \boldsymbol{z} \Longrightarrow v = \|\widehat{\boldsymbol{r}}\| . \|\widetilde{\boldsymbol{T}}_1^{-1} \boldsymbol{z}\| \tag{17}$$

with $\widetilde{\boldsymbol{T}}_1 \in \mathcal{N}^{k \times k}$ and $\boldsymbol{z} \in \mathcal{N}^k$.

Usually in numerical mathematics, one is interested in the worst case that can happen. In the case of (17), this is not a trivial thing since $\|\boldsymbol{T}_1^{-1}\|$ is a stochastic variable with unbounded range. So we must apply statistical concepts like expectations, and standard deviations.

The norm of the inverse of a matrix is the reciprocal of the smallest singular value of the matrix. Edelman, in [2] has derived an asymptotic probability density for the smallest sigular value of a matrix in $\mathcal{N}^{k \times k}$. Denote this singular value by σ_1, then the probability density for $\sigma_1 \sqrt{k}$ reads

$$f(\sigma) = (1 + \sigma) e^{-\sigma^2/2 - \sigma} \tag{18}$$

This means that $\sigma_1 \sqrt{k} \geq C$ for moderate C, and consequently $\|\boldsymbol{T}_2^{-1}\| \leq \widetilde{C} \sqrt{k}$ with probability close to 1, for moderate $\widetilde{C}$.

For the random vector $\boldsymbol{z}$ in (17), we may expect, by elementary statistics, that $\|\boldsymbol{z}\| \leq D\sqrt{k}$ with probability close to 1, again for moderate D. Therefore, since $\boldsymbol{T}_1$ and $\boldsymbol{z}$ are stochastically independent, we may expect

$$\|d\boldsymbol{r}\| \leq \widehat{C} . k . \|\widehat{\boldsymbol{r}}\|$$

with probability close to 1, and $\widehat{C} = \widetilde{C} D$ a moderate constant.

4.1 Completely random linear systems.

We do this analysis in order to examine the *convergence behaviour* of the IDR(s) method, rather than upper bounds. So we really are interested in *expectations* . For the expectation of $\|d\boldsymbol{r}\|$, we may expect a behaviour like $C\sqrt{k}$ instead of Ck. In fact, according to (17), the quantity $\|d\boldsymbol{r}_k\|/\|\widehat{\boldsymbol{r}}_k\|$ behaves like the stochastic variable $\xi = \|\boldsymbol{x}\|$, in which $\boldsymbol{x}$ is the solution of a system $\boldsymbol{B}\boldsymbol{x} = \boldsymbol{b}$, with $\boldsymbol{B} \in \mathcal{N}^{k\times k}$ and $\boldsymbol{b} \in \mathcal{N}^k$. Such a system we call a *completely random linear system*, and we give a description in this section.

Definition 1 *1. A $k\times k$ linear system $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ is called a completely random real system if all entries of $\boldsymbol{A}$ and $\boldsymbol{b}$ are independent standard, normally distributed stochastic variables. 2. The class of solutions $\boldsymbol{x}$ of a completely random $k \times k$ system is denoted by $\mathcal{Q}^k$ if the matrix and the righthand side are real, and by $\widehat{\mathcal{Q}}^k$ if they are complex.*

We start with constructing an *experimental probability density function* (*E-Pdf*) of $\|\boldsymbol{x}\|$, for $\boldsymbol{x} \in \mathcal{Q}^k$, by computing 500 samples of this stochastic variable, for sizes $k = 25, 50, 100, 200$. We actually plotted the histograms of $u = \log_{10}(\|\boldsymbol{x}\|)$ instead of $\|\boldsymbol{x}\|$, since they represent *E-Pdf* 's for the number of decimal digits that Galerkin will 'be behind' the least squares method. The results are shown in Figure 6. The histograms for each k are plotted in different colors. A slight shift to the right can be seen for increasing size. The heuristically expected behaviour $\|\boldsymbol{x}\| \approx C\sqrt{k}$ for a $k \times k$ completely random system, would require the shift to be $\log_{10}(k)/2$.
Therefore we also plot the histograms shifted to the left, with an amount of $\log_{10}(k)/2$ for the case corresponding to size k. The result is shown in Figure 7.

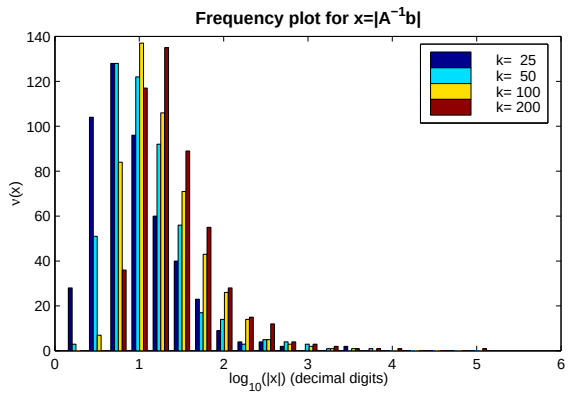

Figure 6: *E-Pdf* of $\log_{10}(\|\boldsymbol{x}\|)$, $\boldsymbol{x} \in \mathcal{Q}^N$

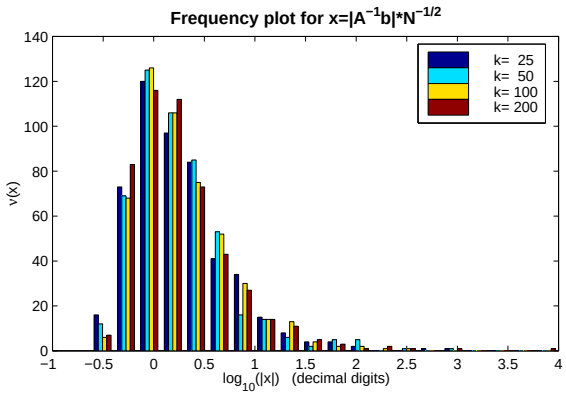

Figure 7: *E-Pdf* of $\log_{10}(\|\boldsymbol{x}\|/\sqrt{k})$, $\boldsymbol{x} \in \mathcal{Q}^k$

The calculated means and variances are given in Table 1. For the classes $\widehat{\mathcal{Q}}^k$, the complex completely random sytems, similar histograms can be made; these can be found in [14]. The relevant statistical quantities are present in Table 2.

k	mean	shifted	var	stdd
25	0.953	0.254	0.239	0.488
50	1.099	0.249	0.226	0.476
100	1.270	0.270	0.211	0.459
200	1.413	0.263	0.264	0.514

Table 1: Params. for $\log_{10}(\|\boldsymbol{x}\|)$, $\boldsymbol{x} \in \mathcal{Q}^k$

k	mean	shifted	var	stdd
25	0.816	0.117	0.080	0.283
50	0.996	0.146	0.083	0.288
100	1.097	0.097	0.064	0.253
200	1.270	0.120	0.068	0.261

Table 2: Params. for $\log_{10}(\|\boldsymbol{x}\|)$, $\boldsymbol{x} \in \widehat{\mathcal{Q}}^k$

The calculated means as well as the histograms indicate that the variable $\log_{10}(\|\boldsymbol{x}\|/\sqrt{k})$ has a distribution function that is nearly independent of k.

Recent developments in the theory of completely random systems have produced analytic density functions for stochastic vectors in $\mathcal{Q}^k$ and $\widehat{\mathcal{Q}}^k$ and their norms, described in [14]. These are given in the following theorem:

Theorem 4.1 *Let $\boldsymbol{x}$ belong to $\mathcal{Q}^k$, and let the stochastic variable $x = \|\boldsymbol{x}\|$ have the probability density function f_k, then*

$$f_k(x) = C\frac{x^{k-1}}{(1+x^2)^{(k+1)/2}}, \quad \text{with } C = \frac{2}{\sqrt{\pi}}\frac{\Gamma(\frac{k+1}{2})}{\Gamma(\frac{k}{2})} \tag{19}$$

If $\boldsymbol{x}$ is in $\widehat{\mathcal{Q}}^N$, the density function for $\|\boldsymbol{x}\|$ reads

$$\widehat{f}_k(x) = \frac{2kx^{2k-1}}{(1+x^2)^{k+1}} \tag{20}$$

In Figure 8 the densities f_k for the real case are plotted against $\log_{10}(x)$ for $k = 25, 50, 100, 200$. $\log_{10}(x)$.

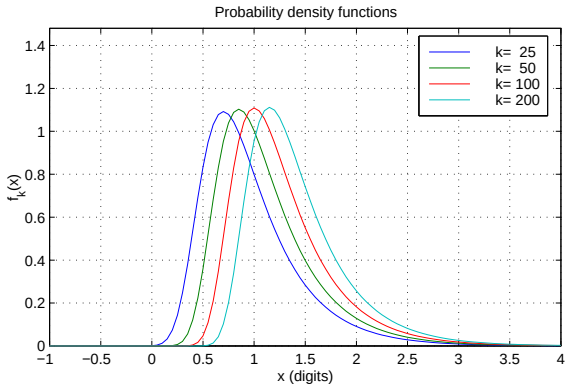

Figure 8: *Pdf* 's for $\log 10(\|\boldsymbol{x}\|)$, $\boldsymbol{x} \in \mathcal{Q}^k$

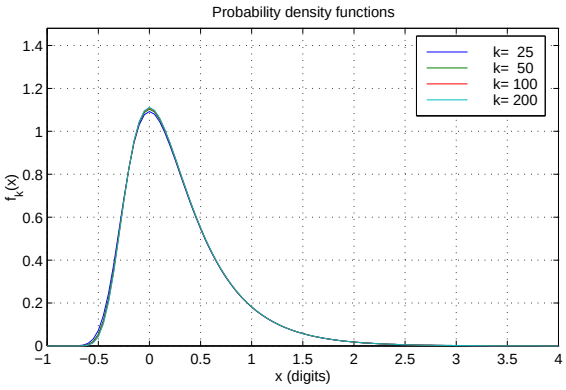

Figure 9: *Pdf* 's for $\log 10(\|\boldsymbol{x}\|/\sqrt{k})$, $\boldsymbol{x} \in \mathcal{Q}^k$

For this family *Pdf* 's, the quantities $\mu_k = E(\log_{10}(x))$ and $\sigma_k^2 = \sigma^2(\log_{10}(x))$ can be derived analytically. We give the asymptotic behaviour for large k:

$$\mu_k \approx \frac{1}{2}\log_{10}(k) + \frac{\gamma + \log(2)}{2\log(10)} \approx \frac{1}{2}\log_{10}(k) + 0.27586 \qquad \text{(real case)} \tag{21}$$

$$\sigma_k^2 \approx \frac{\pi^2}{8\log^2(10)} \approx 0.23269 \qquad \text{(real case)} \tag{22}$$

$$\mu_k \approx \frac{1}{2}\log_{10}(k) + \frac{\gamma}{2\log(10)} \approx \frac{1}{2}\log_{10}(k) + 0.12534 \qquad \text{(complex case)} \tag{23}$$

$$\sigma_k^2 \approx \frac{\pi^2}{24\log^2(10)} + \approx 0.077563 \qquad \text{(complex case)} \tag{24}$$

where γ is Eulers constant:

$$\gamma = \lim_{k\to\infty}\left(\sum_{j=1}^{k}\frac{1}{j} - \log(k)\right)$$

For practical use, it is interesting for which values of $\|\boldsymbol{x}\|$, the probability is less than, say 10^{-j}. It is easy to give estimates, that are quite sharp for $j \geq 2$.
For $\boldsymbol{x} \in \mathcal{Q}^k$:

$$\mathcal{P}\left(\log_{10}(x) > \frac{1}{2}\log_{10}(\frac{2k}{\pi}) + j\right) < 10^{-j} \tag{25}$$

For $\boldsymbol{x} \in \widehat{\mathcal{Q}}^k$:

$$\mathcal{P}\left(\log_{10}(x) > \frac{1}{2}\log_{10}(k) + j/2\right) < 10^{-j} \tag{26}$$

The graphs in Figure 9 are nearly identical, and this can be explained by an asymptotic approximation. Let the stochastic variable y be defined by $x = y\sqrt{k}$, then the probability density of y satisfies

$$g_k(y)dy = f_k(x)dx = f_k(\sqrt{k}y)\sqrt{k}dy \Longrightarrow g_k(y) = C\frac{x^{k-1}\sqrt{k}}{(1+x^2)^{\frac{1}{2}(k+1)}}$$

with C as defined in (19).
With help of Stirlings formula we find $C \approx \sqrt{\frac{2k}{\pi}}$. Then by applying the elementary approximation $(1+a)^b \approx e^{ab}(1+O(a^2b))$ for $|a| < 1$, to $[1+\frac{1}{ny^2}]^{-\frac{1}{2}(n+1)}$, we arrive at

$$g_k(y) \approx \sqrt{\frac{2}{\pi}}\frac{\exp(-\frac{1}{2y^2})}{y^2}$$

For $y \to 0$ and k fixed, $g_k(y) = O(y^{k-1})$, so it tends to zero rapidly. The asymptotic approximation tends to zero expremely fast as y^2 becomes small. Although there is a difference in behaviour, this difference is hardly visible in practice. Therefore the asymptotic formula might replace the original distribution perfectly well if only k is not too small, say $k > 20$.
This analysis explains the coinciding graphs in Figure 9, and the suggestion raised by Figure 7.

5 Experimental verification.

5.1 Irregularities, residual replacents, and residual smoothing.

As many other short recurrence Krylov subspace solvers, the behaviour of the residuals can be very irregular. These irregularities can influence the convergence dramatically, and although this problem is not the subject of this paper, we must deal with it properly.
First, in the basic form of the IDR(s) method, it can be observed that the occurrence of residuals that are some orders of magnitude larger than the initial residual, lead to a final (stagnation) level that is similarly higher than the theoretical possibility. Roughly speaking: Occurrence of a residual $\|\boldsymbol{r}_k\| > 10^d\|\boldsymbol{r}_0\|$ leads to a loss of d digits in the end. This is explainable on very elementary grounds.
Backgrounds of these phenomena can be found in [5], and remedies are proposed in [18]. Most remedies are based on a careful application of *residual replacement*. This means that at a certain stage the recursively calculated residual is replaced by the actual one:

$\boldsymbol{b} - \boldsymbol{A}\boldsymbol{x}_k$, at the cost of one extra matrix vector product. This may not be done near the end of the process, since then the convergence will detoriate by an other cause.
In the experiments, we used a very primitive variant of this principle: As soon as some residual is higher than the starting residual, (which is practically always the case in the beginning of the process), a repair flag is set. As soon as a residual arises which is a factor 10 smaller than the initial level, and the repair flag is set, the current residual is replaced by the true residual, and the repair flag is reset.
In many experiments, the stagnation level turned out to be comparable to the stagnation level of full GMRES. The number of extra matrix vector products never exceeded 3, so it is done at marginal cost.
In some rare cases, the primitive strategy did not help, although it did not harm either. But it is highly probable that a strategy based on [18] will work in all circumstances, at an acceptable price.

In testing the convergence theory for the Lanczos part of the residuals, we can accept the irregularities as a consequence of the stochastic basis of the theory, and as such, it can serve as a possible confirmation of the theory. But in order to visualize what the theory means for practical use, we may drop all irregularities, and apply some kind of *residual smoothing*. Such a technique does not influence the algorithm as such, but makes a more reliable use of the algorithm's results. In this paper, we use an extremely primitive variant, and call it *lower bound smoothing*. Here we define the smoothed residual $\overline{\boldsymbol{r}}_k$ as follows

$$\overline{\boldsymbol{r}}_0 = \boldsymbol{r}_0, \quad \overline{\boldsymbol{r}}_k = \min\{\overline{\boldsymbol{r}}_{k-1}, \boldsymbol{r}_k\}, \ k \geq 1$$

In practice, one should only use $\boldsymbol{x}_k$ as approximation of the solution $\boldsymbol{x}$, if $\overline{\boldsymbol{r}}_k = \boldsymbol{r}_k$, but this will automatically be the case in most stop criteria.

5.2 Test examples.

For the test of the stochastic convergence theory, we need a sequence of reduced residuals $\widetilde{\boldsymbol{r}}_k = \Psi_k(\boldsymbol{A})\boldsymbol{r}_0$, as defined in (4). These reduced residuals satisfy the same recurrence relations as the ordinary residuals, except at the steps $k = j(s+1)$, in which the explicit factor $(\boldsymbol{I} - \omega_j \boldsymbol{A})$ that occurs in the computation of $\boldsymbol{r}_k$, is not applied in the calculation of $\widetilde{\boldsymbol{r}}_{k-j}$.
Unfortunately, we have no $\boldsymbol{x}$-recurrence for these reduced residuals, hence we only have the recursively calculated residuals. We assume that these are reliable as long as the *true IDR(s) residuals* are reliable, i.e. are above their stagnation level.
According to the theory described in Section 3, the reduced residuals can be regarded as Krylov Galerkin residuals, with increasing Galerkin dimension k. The GMRES residuals are considered as least squares residuals, and denoted by $\widehat{\boldsymbol{r}}_k$. The residual surplus, as defined in (9), is then $d\boldsymbol{r}_k = \widetilde{\boldsymbol{r}}_k - \widehat{\boldsymbol{r}}_k$, and its norm is calculated as

$$\|d\boldsymbol{r}_k\| = \sqrt{\|\widetilde{\boldsymbol{r}}_k\|^2 - \|\widehat{\boldsymbol{r}}_k\|^2}$$

The k-th iteration step provides a realization of a stochastic variable $\log_{10}(\|d\boldsymbol{r}_k\|) = \log_{10}(\|\widehat{\boldsymbol{r}}_k\|) + \log_{10}(\|\boldsymbol{z}\|)$, with $\boldsymbol{z} \in \mathcal{Q}^k$ (or $\widehat{\mathcal{Q}}^k$ in the case of complex $\boldsymbol{P}$).

The test problems are Problem 2 (convection diffusion with mild Peclet numbers) and Problem 3 (similar, but with high Peclet numbers), as described in Section 1, and Section 2.1.

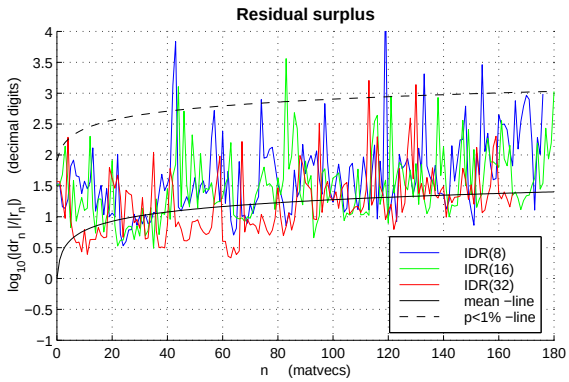

Figure 10: $\log_{10}(\|d\widetilde{\boldsymbol{r}}\|/\|\widehat{\boldsymbol{r}}\|)$, Prob. 2, $\boldsymbol{P} \in \mathbb{R}^{N\times s}$

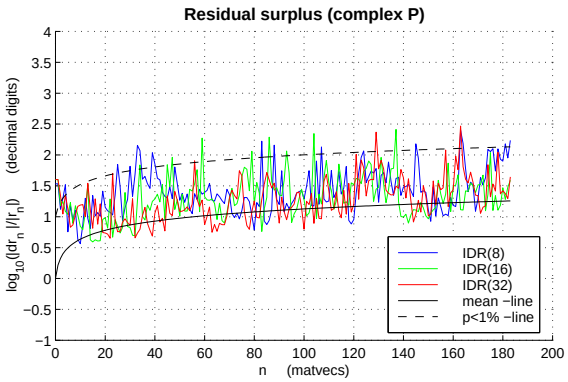

Figure 11: $\log_{10}(\|d\widetilde{\boldsymbol{r}}\|/\|\widehat{\boldsymbol{r}}\|)$, Prob. 2, $\boldsymbol{P} \in \mathbb{C}^{N\times s}$

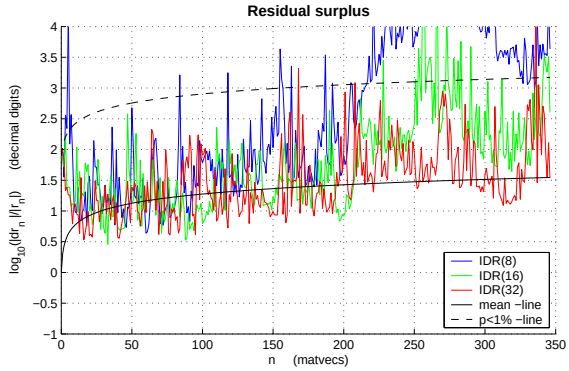

Figure 12: $\log_{10}(\|d\widetilde{\boldsymbol{r}}\|/\|\widehat{\boldsymbol{r}}\|)$, Prob. 3, $\boldsymbol{P} \in \mathbb{R}^{N\times s}$

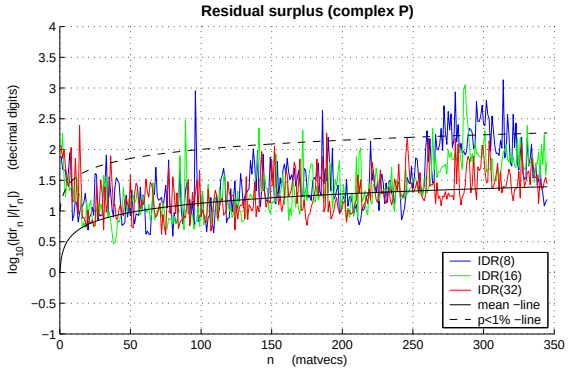

Figure 13: $\log_{10}(\|d\widetilde{\boldsymbol{r}}\|/\|\widehat{\boldsymbol{r}}\|)$, Prob.3, $\boldsymbol{P} \in \mathbb{C}^{N\times s}$

In Figure 10–13, a solid black curve denotes the expectation of this stochastic variable; a dashed black curve depicts the number of digits that the reduced residual is behind the GMRES curve with a probability of at most 1 %.
The theoretical formulae are designed for $s = \infty$. For finite s, they can be considered as valid as long as $k < s$. From the pictures we can see wether the theory remains valid for iteration steps beyond this bound.
We run IDR(s) for various s-values, and with real $\boldsymbol{P}$ as well as with complex $\boldsymbol{P}$. The results for real $\boldsymbol{P}$ are shown in Figure 10 and Figure 12. Figure 11 and Figure 13 show the results for complex $\boldsymbol{P}$.
In order to get a view on the expectation for practical use of the method, we added the plots based on lower bound smoothed residuals in Figure 14–17.

6 Conclusions

Convergence mechanisms. We have shown that the convergence of IDR(s) depends on two mutually independent regimes. The Lanczos regime is determined completely by the choice of $\boldsymbol{P}$, whereas the damping regime depends also on the choice of the ω-parameters. The example shown in Figure 4 shows that the polynomials $\Omega_j(\boldsymbol{A})$ are not damping at all for that problem. Only for high values of s, the Lanczos component of the convergence is stronger, because the relatively low degree of the $\Omega_j(\boldsymbol{A})$ polynomials in these circumstances.

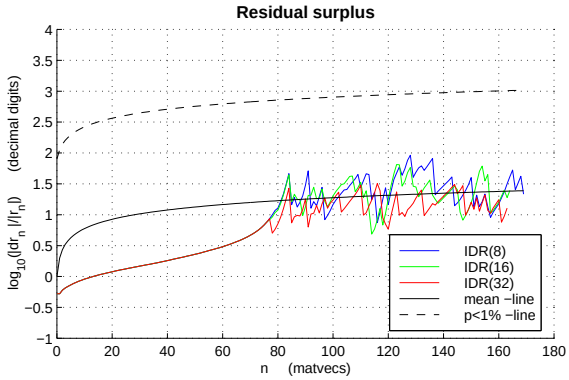

Figure 14: $\log_{10}(\|d\widetilde{\boldsymbol{r}}\|/\|\widehat{\boldsymbol{r}}\|)$, Prob. 2, $\boldsymbol{P} \in \mathbb{R}^{N\times s}$

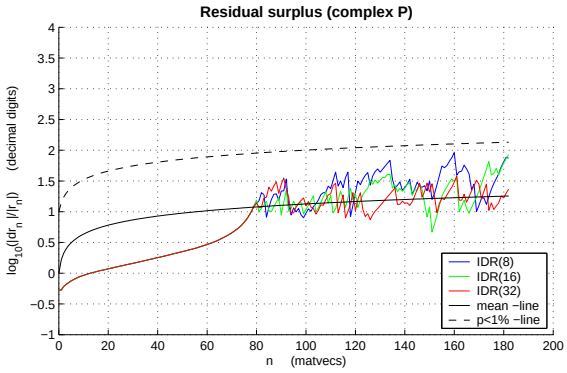

Figure 15: $\log_{10}(\|d\widetilde{\boldsymbol{r}}\|/\|\widehat{\boldsymbol{r}}\|)$, Prob. 2, $\boldsymbol{P} \in \mathbb{C}^{N\times s}$

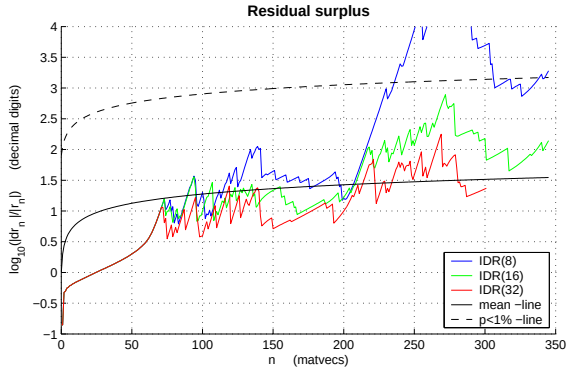

Figure 16: $\log_{10}(\|d\widetilde{\boldsymbol{r}}\|/\|\widehat{\boldsymbol{r}}\|)$, Prob. 3, $\boldsymbol{P} \in \mathbb{R}^{N\times s}$

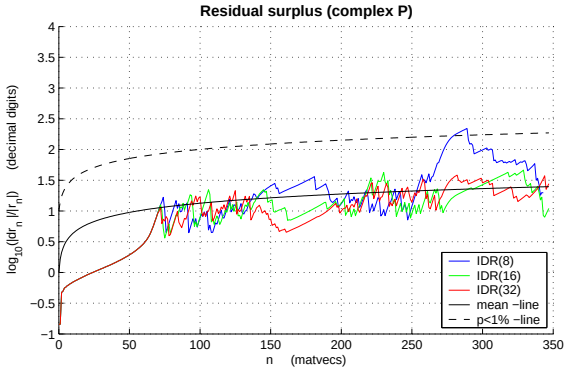

Figure 17: $\log_{10}(\|d\widetilde{\boldsymbol{r}}\|/\|\widehat{\boldsymbol{r}}\|)$, Prob.3, $\boldsymbol{P} \in \mathbb{C}^{N\times s}$

Real versus complex $\boldsymbol{P}$. The results shown in Figure 5, Figure 11 and Figure 13 justify the use of complex $\boldsymbol{P}$, in any case for complex problems. The last two examples also show a significantly lower level of the residual surplus. This is in agreement with the theory (formulae (21), (23), (25) and (26)).
For real problems, the complex choice of $\boldsymbol{P}$ is a bit expensive. The work load for inner product calculations and linear combinations is about four times as high compared to the real choice of $\boldsymbol{P}$. Only the matrix vector multiplications can be carried out in only two times the work for the real case. This can be done by calculating $\Re(\boldsymbol{y}) = \boldsymbol{A}\Re(\boldsymbol{x})$, and $\Im(\boldsymbol{y}) = \boldsymbol{A}\Im(\boldsymbol{x})$ instead of simply $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$. But besides that, contributions as in [6], [11], [10], and recently in [12] are important for mastering the stabilization/damping issue completely.
Bad damping by the $\Omega_j(\boldsymbol{A})$ factors also occur in cases where $\boldsymbol{A}$ has eigenvalues to both sides of the imaginary axis, as in Helmholtz equation. In [15] is shown that IDR(s) converges very well in these cases. Unfortunately we could not test the theory on this kind of problems, since it appeared to be impossible to obtain reliable reduced residuals.

Lanczos part of the convergence. The stochastic theory developed in Section 4 appears to be confirmed rather well by Figure 10–13. So the theory is a plausible explanation of the observed limiting behaviour of the IDR(s) residual curves for increasing s. But although the use of a normally distributed stochastic matrix $\boldsymbol{P}$ is essential for the proof of this limiting behaviour, other choices for $\boldsymbol{P}$ may work equally good. It appears that higher values for s are only necessary for problems where methods like Bi-CGSTAB per-

form poorly. These are the problems for which the angle between the Krylov subspaces belonging to $\boldsymbol{A}$ and $\boldsymbol{A}^*$ respectively is close to $\frac{1}{2}\pi$. A high value of s keeps the 'bad influence' of the adjoint Krylov space down.

Practical expectations. Concluding from Figure 14–17, it appears that for a wide range of problems and Krylov dimensions, $\|\boldsymbol{r}_k^{\mathrm{IDR(s)}}\| \lesssim 10\|\boldsymbol{r}_k^{\mathrm{GMRES}}\|$, as long as the 'damping factors' are not too expanding. If complex $\boldsymbol{P}$ is chosen, the damping problem appears to be of no importance.

If IDR(s) is compared to full GMRES when the latter is in its fast convergence regime, the factor 10 is visible as only a slight shift to the right, meaning IDR(s) is only a few iteration steps behind.
The IDR(s) method therefore provides a relatively cheap and reliable alternative for full GMRES.

Acknowledgements: The author wants to thank Jos van Kan, Martin van Gijzen, Tijmen Collignon and Kees Vuik for many fruitful discussions, and Kees Vuik for careful reading the manuscript and for his many useful hints.

References

[1] Z. Chen and J. J. Dongarra: *Condition numbers of Gaussian Random matrices;* SIAM Journal on Matrix Analysis and Applications **27(3)**: pp. 603-620, (2005)

[2] A. Edelman: *Eigenvalues and Condition numbers of Random matrices;* SIAM Journal on Matrix Analysis and Applications **9(4)**: pp. 543-560, (1988)

[3] M. Eierman and O. G. Ernst: *Geometric aspects of the theory of Krylov subspace methods;* Acta Numerica : pp. 251-312, (2001)

[4] M. Ghosh and B. K. Sinha: *A simple Derivation of the Wishart Distribution;* The American Statistician, May 2002 **56(2)**: pp. 100-101, (2002)

[5] A. Greenbaum: *Estimating the attainable accuracy of recursively computed residual methods;* SIAM J. Matrix Anal. Appl. **18**: pp. 535-551, (1997)

[6] M.H. Gutknecht: *Variants of BICGSTAB for Matrices with Complex Spectrum;* SIAM J. Sci. Comp. **14(5)**: pp. 1020-1033, (1993)

[7] M. B. van Gijzen and P. Sonneveld: *An elegant IDR(s) variant that efficiently exploits bi-orthogonality properties;* Delft University of Technology, Reports of the Department of Applied Mathematical Analysis **08-21**: (2008)

[8] M. Hochbruck and C. Lubich: *Error Analysis of Krylov Methods in a Nutshell;* SIAM J. Sci. Comp. **19(2)**: pp. 695-701, (1998)

[9] Y. Saad and M.H. Schultz: *GMRES: A generalized minimum residual algorithm for solving nonsymmetric linear systems;* SIAM J. Sci. Statist. Comput. **7**: pp. 856-869, (1986)

[10] V. Simoncini and D. B. Szyld : *Interpreting IDR as a Petrov-Galerkin method ;* Research Report 9-10-22, Department of Mathematics, Temple University : (2009)

[11] G.L.G. Sleijpen and D.R. Fokkema: *BiCGstab(ℓ) for linear equations involving matrices with complex spectrum;* ETNA **1**: pp. 11-32, (1994)

[12] G.L.G. Sleijpen and M.B. van Gijzen: *Exploiting BiCGstab(l) strategies to induce dimension reduction;* Delft University of Technology, Reports of the Department of Applied Mathematical Analysis **09-02**: (2009)

[13] G.L.G. Sleijpen and H.A. van der Vorst: *Maintaining convergence properties of BiCGstab methods in finite precision arithmetic;* Numerical Algorithms **10**: pp. 203-223, (1995)

[14] P. Sonneveld: *On the statistical properties of solutions of completely random linear systems.;* Delft University of Technology, Reports of the Department of Applied Mathematical Analysis **10-09**: (2010)

[15] P. Sonneveld and M. B. van Gijzen: *IDR(s): a family of simple and fast algorithms for solving large nonsymmetric systems of linear equations ;* SIAM J. Sci. and Statist. Comput. **31:2**: pp. 1035-1062, (2008)

[16] S. J. Szarek: *Condition numbers of random matrices;* Journal of Complexity **7(2)**: pp. 131-149, (1991)

[17] H. A. van der Vorst: *Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems;* SIAM J. Sci. Comp. **13**: pp. 631-644, (1992) .

[18] H. A. van der Vorst, Q. Ye: *Residual replacement strategies for Krylov subspace iterative methods for the convergence of true residuals;* SIAM J. Sci. Comp. **22**: pp. 835-852, (2001) .