LAPACK WORKING NOTE 194: A REFINED REPRESENTATION TREE FOR MRRR

CHRISTOF VÖMEL‡

Abstract. In order to compute orthogonal eigenvectors of a symmetric tridiagonal matrix without Gram-Schmidt orthogonalization, the MRRR algorithm finds a shifted LDL^T factorization (representation) for each eigenvalue such that the local eigenvalue is a singleton, that is defined to high relative accuracy and has a large relative gap.

MRRR's representation tree describes how, by successive shifting and refinement, each eigenvalue becomes relatively isolated. Its shape plays a crucial role for complexity: deeper trees are associated with more eigenvalue refinement to resolve clustering of eigenvalues.

Motivated by recently observed deteriorating complexity of the LAPACK 3.1 MRRR kernels for certain matrices of large dimension, we here re-examine and refine the representation tree concept.

We first describe the discovery of what we call a spectrum peeling problem: even though the matrix at hand might not have a spectrum with clusters within clusters, the representation tree might still contain a long chain of large nodes.

We then formulate a refined proposal for the representation tree that aims at avoiding the unwarranted work while preserving tight accuracy bounds where possible. The trade-off between performance and accuracy in our solution is discussed by practical examples.

AMS subject classifications. 65F15, 65Y15.

Key words. Multiple relatively robust representations, MRRR, complexity, representation tree, spectrum peeling, LAPACK, ScaLAPACK.

1. Introduction. In 2006, a technical report [23] comparing several parallel algorithms for the symmetric eigenvalue problem showed surprisingly poor performance of a ScaLAPACK prototype [2] of the algorithm of Multiple Relatively Robust Representations, MRRR or MR^3 [8, 21, 22, 9, 10, 12], on certain large matrices from electronic structure calculations. An additional remarkable finding was that an earlier implementation of parallel MRRR [3] did not exhibit the same kind of problem.

Interested in shedding light on this mystery, we investigated the behavior of the ScaLAPACK code on these matrices and were surprised to find the root cause for its poor performance to be beyond a parallelization issue. As we show in this paper, design decisions aimed at achieving greater accuracy of the LAPACK 3.1 version [17] of MRRR, which were adapted in the parallel code, can 'backfire' for certain large matrices and result in an artificially 'bloated' representation tree involving a huge amount of unwarranted computations.

In order to address this problem, we propose here a refined formulation of the representation tree that allows us to reap accuracy benefits without incurring high performance penalties. The key idea is to combine a strict requirement on the relative gap of singletons with an additional criterion for eigenvalue groups. This yields a refined representation tree that is beneficial for both the sequential and parallel MRRR algorithm.

The rest of this paper is organized as follows. Section 2 gives an overview of the MRRR algorithm with an emphasis on the role and construction of the representation tree. In Section 3, we present examples of representation trees associated with extreme amounts of computation, including the worst-case example from [23]. The

‡Institute of Computational Science, ETH Zürich, CAB, Universitätsstraße 6, CH-8092 Zürich, Switzerland, cvoemel@inf.ethz.ch

negative impact of such trees on computational complexity and also the parallelization of MRRR is discussed. Based on these observations, we propose in Section 4 the use of refined criteria to automatically build a preferable representation tree. The effect of our proposal is evaluated in Section 5. Section 6 summarizes our results and conclusions.

2. The role of the representation tree in the MRRR algorithm. This section gives a terse but self-contained description of the aspects of MRRR important for this paper. Further information on the representation tree can be found in particular in [9], but see also [8, 21, 22, 10, 12].

2.1. Basic description of the MRRR algorithm. Let the symmetric tridiagonal matrix T be represented in factored form

$$T - \sigma I = LDL^T. \tag{2.1}$$

Further, for an eigenvalue λ of LDL^T, let its approximation $\hat{\lambda}$ satisfy

$$\|\lambda - \hat{\lambda}\| = \mathcal{O}(\varepsilon|\lambda|). \tag{2.2}$$

Here, ε denotes the relative machine precision. To be able to compute $\hat{\lambda}$, we require that small relative changes in each entry of L and D should only cause small relative changes in λ. Such an LDL^T factorization is called a Relatively Robust Representation (RRR) for this eigenvalue.

At the heart of the MRRR algorithm lies the idea of being able to compute an eigenvector approximation $\hat{v}, \|\hat{v}\|_2 = 1$ to the the true eigenvector v that satisfies

$$\|(LDL^T - \hat{\lambda}I)\hat{v}\|_2 = \mathcal{O}(n\varepsilon|\hat{\lambda}|). \tag{2.3}$$

The reward for the residual norm being small relative to $|\hat{\lambda}|$ (and to $|\lambda|$) is revealed by the classical gap theorem [4, 5, 19]. In general, inverse iteration [16, 19] with T satisfies

$$|\sin\angle(v,\hat{v})| \leq \frac{\|(T - \hat{\lambda}I)\hat{v}\|_2}{\mathrm{gap}(\hat{\lambda})} = \frac{\mathcal{O}(n\varepsilon\|T\|_2)}{\mathrm{gap}(\hat{\lambda})} \tag{2.4}$$

where $\mathrm{gap}(\hat{\lambda}) = \min\left\{|\hat{\lambda} - \mu| : \lambda \neq \mu, \mu \in \mathrm{spectrum}(T)\right\}$. But with (2.3), one has

$$|\sin\angle(v,\hat{v})| \leq \frac{\|(LDL^T - \hat{\lambda}I)\hat{v}\|_2}{\mathrm{gap}(\hat{\lambda})} = \frac{\mathcal{O}(n\varepsilon)}{\mathrm{relgap}(\hat{\lambda})} \tag{2.5}$$

where

$$\mathrm{relgap}(\hat{\lambda}) := \frac{\mathrm{gap}(\hat{\lambda})}{|\hat{\lambda}|} \tag{2.6}$$

and $|\hat{\lambda}|$ could be much smaller than $\|T\|_2$.

For the bound (2.5) on the angle between computed and true eigenvector to be small, $\mathrm{relgap}(\hat{\lambda})$ should be as large as possible. In practice, a threshold τ_{minrgp} is employed to decide when a relative gap is large enough. A (shifted) eigenvalue approximation satisfying (2.2) and

$$\mathrm{gap}(\hat{\lambda}) \geq \tau_{\mathrm{minrgp}} \cdot |\hat{\lambda}| \tag{2.7}$$

is called a *singleton*; by (2.5) MRRR can guarantee that the corresponding eigenvector satisfies

$$|\sin \angle(v, \hat{v})| = \mathcal{O}(n\varepsilon/\tau_{\text{minrgp}}). \tag{2.8}$$

2.2. Construction of the representation tree. The definition (2.6) of the relative gap shows that in theory, a relative gap can be made arbitrarily large by shifting close to the eigenvalue. Thus, for a matrix where several eigenvalues might agree to many or all their figures, one can compute a new RRR

$$L_+ D_+ L_+^T = LDL^T - \sigma I \tag{2.9}$$

for the eigenvalues in the cluster. This can be done stably by differential stationary qds (dstqds) factorization [13, 14, 18]. If one shifts close enough to the cluster, at least one of the eigenvalues of the new, shifted RRR becomes a singleton. If there are still clusters with respect to the new RRR, the procedure can be repeated for them. Note that because of the shifting, the computed eigenvalues lose some of their correct figures and need to be refined to become sufficiently accurate again. The idea of using various shifted RRRs to refine the spectrum, until each eigenvalue becomes a singleton, gives the algorithm its name, MRRR = multiple relatively robust representations.

The so-called representation tree is a graph that describes the application of MRRR. The root node is an RRR for all the desired eigenvalues. A descendant is an RRR for an eigenvalue subset, the edge between them a shift relation (2.9). The singletons are the leafs.

Choosing a good shift is of key importance. It has to be guaranteed that [9]

- the new factorization $L_+ D_+ L_+^T$ is an RRR for all eigenvalues of the group, and that
- at least one eigenvalue of the group, the one closest to the shift, becomes a singleton.

In order to guarantee that one finds at least one singleton as a child of the current RRR, one has to shift close enough so that subsequent eigenvalue refinement for the new RRR can reveal a large enough relative gap. Furthermore, one prefers to shift just outside of an eigenvalue group rather than inside as to reduce the risk of element growth in the factorization which could spoil the RRR property. One can shift at both ends of a cluster, and also back off slightly from the ends, to find a location with small element growth, see [12].

A second issue that is of particular importance for this paper is how to group those eigenvalues that are not singletons. The approach that is used in LAPACK 3.0 [1] and 3.1 [17] is to inspect one-sided gaps. Following [19], we number the eigenvalues of an unreduced tridiagonal in increasing order

$$\lambda_1 < \lambda_2 < \ldots < \lambda_n.$$

Then, define the right gap

$$\text{rgap}(\hat{\lambda}_i) := |\hat{\lambda}_{i+1} - \hat{\lambda}_i|, \tag{2.10}$$

and choose a threshold τ_{minrgp}. Then a boundary from one child to the next is defined through the separation criterion

$$\text{rgap}(\hat{\lambda}_k) \geq \tau_{\text{minrgp}} \cdot |\hat{\lambda}_k|. \tag{2.11}$$

If multiple eigenvalues are enclosed between rgaps satisfying (2.11), one has found a group for which a new RRR is computed. Otherwise one has found a singleton. Algorithm 1 gives the details.

Algorithm 1 Construction of the representation tree in the MRRR algorithm in LAPACK 3.0 and 3.1. See also Figure 2 in [9].

```
Let LDL^T be an RRR for λ̂_i, ..., λ̂_j.
Assume that λ̂_i, ..., λ̂_j have suitable accuracy (e.g. by bisection).
Initialize start index s = i of eigenvalue group.
for k = i : j do
  if rgap(λ̂_k) ≥ τ_minrgp · |λ̂_k| then
    A boundary between groups has been found.
    if start index s > k then
      An eigenvalue group has been found.
      Choose shift σ outside the group.
      Compute new RRR, L_+ D_+ L_+^T = LDL^T − σI.
    else
      A singleton has been found.
      The corresponding eigenvector can be computed.
    end if
    Set s = k + 1 to be the start index for the next group.
  end if
end for
```

2.3. The choice of $\tau_{\mathbf{minrgp}}$. A threshold of $\tau_{\text{minrgp}} = 1/n$, with n being the dimension of the unreduced tridiagonal matrix, was initially suggested in [8]. All experimental evidence available at the time indicated that residual norms and dot products between different computed vectors were bounded by multiples of $n\varepsilon$ rather than what a pessimistic interpretation of 'crude' bounds such as (2.8) would predict, see comments in [8]. Consequently, it was used for all matrices of size greater than 100 in the LAPACK 3.0 [1] implementation.

Subsequently, with more extensive testing on much larger matrices with more and more challenging eigenvalue distributions, it was felt that τ_{minrgp} should be bounded by a constant to better prevent deteriorating orthogonality with increasing matrix size observed in experiments. Secondly, the LAPACK 3.0 threshold cannot be chosen independently from the matrix size which makes it inflexible when more accuracy is desired. For these reasons, the current version of MRRR [12] in the latest LAPACK 3.1 release [17] uses, in DOUBLE PRECISION, a fixed threshold $\tau_{\text{minrgp}} = 10^{-3}$. This means that eigenvalues that agree to less than three digits are declared relatively isolated, independent of the matrix dimension.

A comparison [7] of LAPACK's tridiagonal eigensolvers using a comprehensive set of test problems [6] shows improvements in the accuracy of MRRR in LAPACK 3.1 compared to version 3.0. These were attributed in part to the modified threshold τ_{minrgp}.

2.4. The representation tree for W_{21}^+. As an example, we consider the application of Algorithm 1 to a Wilkinson matrix [24]

$$W_{2m+1}^+ = \text{tridiag}\begin{pmatrix} 1 & 1 & \dots & 1 & 1 \\ m & m-1 & \dots\ 0\ \dots & m-1 & m \\ 1 & 1 & \dots & 1 & 1 \end{pmatrix}. \tag{2.12}$$

What follows is a summary of the representation tree construction for W_{21}^+, a detailed exposition is given in [11] using the same threshold $\tau_{\text{minrgp}} = 10^{-3}$. (That is, eigenvalues that agree to fewer than three digits are declared relatively isolated.) The large eigenvalues of Wilkinson matrices of odd order come in very close pairs which are separated by large gaps. This is reflected in the representation tree.

MRRR first computes an RRR for all eigenvalues, shifting to the outside of the rightmost eigenvalue. The eigenvalues of this root RRR are inspected. The leftmost nine eigenvalues are declared singletons. The remaining ones come in close pairs (close with respect to τ_{minrgp}) for each of which a new RRR is computed. At the next level of the tree, once the clustered eigenvalues have been refined, their relative gaps are found to be larger than τ_{minrgp}. This concludes the construction. of the representation tree which is depicted in Figure 2.1.

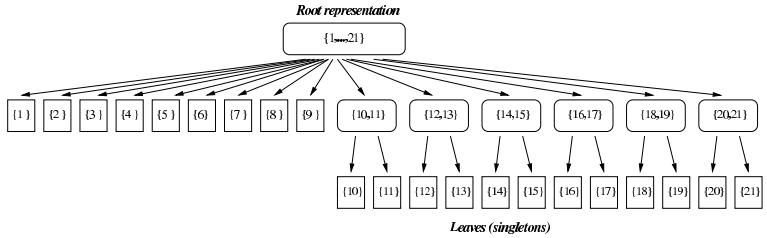

FIG. 2.1. *The representation tree for W_{21}^+. Square boxes correspond to singletons. Boxes with round corners correspond to eigenvalue groups for which an individual RRR is computed whose local eigenvalues are subsequently refined.*

3. The eigenvalue peeling problem. We now return to the starting point of our investigation. The introduction, Section 1, mentioned severe performance differences between different versions of parallel MRRR on certain large matrices. As we will see shortly, the cause of this problem is the definition of separation between eigenvalue groups as in (2.11) and the resulting representation tree produced by Algorithm 1.

3.1. The peeling problem in LAPACK 3.1. For sake of illustration, let us consider a large Wilkinson matrix W_{2m+1}^+ from (2.12) where m is extremely large, say W_{20001}^+. It can be verified that all eigenvalues are contained in an interval $[-2, m+1]$ and that the larger eigenvalues of Wilkinson matrices of odd order come in very close pairs which are separated from each other by absolute gaps of size about one. Further, to simplify the discussion, we assume that MRRR at the beginning shifts to the left so that the root representation is positive definite.

With the LAPACK 3.1 threshold $\tau_{\text{minrgp}} = 10^{-3}$, see Section 2.3, criterion (2.11) implies that the separation between eigenvalue groups needs to satisfy

$$\text{rgap}(\hat{\lambda}_k) \geq 10^{-3}|\hat{\lambda}_k|. \tag{3.1}$$

We then observe that for $\hat{\lambda}_k$ larger than $\tau_{\text{minrgp}}^{-1} = 1000$, an absolute gap of size about one will not pass the test (3.1). As a consequence, all these eigenvalues, in our example about 18,000, are declared to be part of a giant 'group.' On the next representation tree level, the process repeats itself: about one thousand eigenvalue pairs each become a child, all others are declared grouped together. Thus, at each level of the tree, the algorithm cumbersomely 'peels off' a fraction of the spectrum.

This situation is unsatisfying for the following reasons:

1. The human eye immediately recognizes that an RRR is needed for each pair of close eigenvalues, whereas the automatic criterion produces an 'artificially long' chain in the representation tree. The computer suggests that the spectrum consists of clusters within clusters of eigenvalues where in reality, it does not.

2. From the efficiency point of view, the repeated refinement of eigenvalues for each RRR with a deep representation tree is hugely expensive. Plus the potential for rounding errors spoiling the results increases.

The example of a large Wilkinson matrix in the previous exposition was chosen in retrospect, to give a simple illustration of a real problem. We now turn to our original motivation of understanding the results in [23]. It turns out that the same peeling phenomenon can be observed. An analysis of the largest matrix, of dimension 63504 and arising from Hubbard models in electronic structure calculations, reveals a group of size 61735, spanning eigenvalues 1115 to 62849, that contains a group of eigenvalues 1115 to 59566, in which eigenvalues 4226 to 59566 again forms a group, and so forth. The maximum depth of the representation tree is 17.

3.2. The representation tree in LAPACK 3.0. For the LAPACK 3.0 threshold $\tau_{\text{minrgp}} = 1/n$, with n being the dimension of the unreduced tridiagonal matrix, criterion (2.11) implies that the separation between eigenvalue groups needs to satisfy

$$\text{rgap}(\hat{\lambda}_k) \geq \frac{|\hat{\lambda}_k|}{n}. \tag{3.2}$$

One can see that this criterion leads to a weaker bound (2.8) than (3.1) which was the motivation to change it in LAPACK 3.1. Nevertheless, let us investigate the representation trees for the examples from Section 3.1 with this criterion.

For the Wilkinson matrix, because the width of the spectrum is less than $m+3 \approx N/2$, each gap of size about one is large enough to pass the threshold test (3.2). As a result, the children of the root representation are either singletons or pairs of eigenvalues. In this case, the automatic criterion produces exactly what would be decided by a human: compute an RRR for each of the close pairs and use the new representation for the eigenvector computation. The resulting representation tree is shallow.

In the case of the Hubbard matrix, one finds that the largest group in the representation tree with criterion (3.2). consists of 16 eigenvalues. Again, the tree is shallow with a maximum depth of two.

3.3. Summary and impact of the peeling problem. Table 3.1 summarizes the characteristics of the different representation trees discussed in Sections 3.1 and 3.2.

Matrix	Size	Tree depth		Max. group size	
		(3.2)	(3.1)	(3.2)	(3.1)
Wilkinson	20001	1	11	2	18003
Hubbard	63504	2	17	16	61735

TABLE 3.1

Comparison of representation tree properties when using criteria (3.2) and (3.1). Reported are the maximum tree depth and the size of the largest eigenvalue group in an RRR except for the root.

What we call the eigenvalue peeling problem manifests itself as a long 'artificial' chain of fat nodes in the representation tree from criterion (3.1): the representation tree can suggest that the spectrum consists of clusters within clusters of eigenvalues where in reality, it might not. In the example of the Wilkinson matrix, the representation tree merely gives the illusion of a difficult matrix where the human eye

immediately recognizes that one representation is needed for each pair of close eigenvalues. (And incidentally in this case, the automatic criterion (3.2) does produce such a tree.) To our surprise we learn, compared to [9], that grouping eigenvalues by relative gaps might become less reliable a criterion when the matrix size increases.

In the case of the Hubbard matrix, we also note the spectrum peeling phenomenon with criterion (3.1). Whether this is indeed a difficult matrix or whether the representation tree complexity is unwarranted, will be discussed later, in Section 5. Here, we discuss the impact of spectrum peeling on complexity and also parallelization of MRRR.

First and foremost, each additional level of RRRs in the representation tree is associated to refinement of eigenvalues. In the Wilkinson example from Section 3.1, one first needs to refine 18,000 eigenvalues, then 16,000 of those, then again 14,000, and so forth. This makes a deep representation tree hugely expensive. Plus the potential for rounding errors spoiling the results increases. When the tree is shallow as in Section 3.2, this overhead is not present.

We also mention that when MRRR is parallelized, such trees can cause additional scalability issues: when the eigenvector computation is equally distributed among processors, the work load is highly unbalanced between a processor working on a shallow part of the representation tree and one assigned to a deep part arising from peeling. We note that the eigenvector computation in MRRR is only embarrassingly parallel if the root representation consists of singletons.

The observations in this section explain the results in [23], in particular the most extreme case of the Hubbard matrix considered here. The ScaLAPACK prototype [2] of MRRR used the newer LAPACK 3.1 criterion (3.1) which has the peeling problem. The earlier implementation of parallel MRRR [3] uses the original LAPACK 3.0 criterion (3.2) which can produce shallow and skinnier representation trees.

4. Refining the definition of the representation tree. In this section, we discuss possible remedies for the peeling issue with some representation trees generated by MRRR in LAPACK 3.1.

4.1. Necessary stability of invariant subspaces. Before we start introducing modifications, it is instructive to revisit the foundations of the MRRR algorithm.

For an LDL^T factorization, consider a set of its eigenvalues $\lambda_i, \ldots, \lambda_j$ with eigenvectors $v_i, \ldots, v_j$. Let $\Gamma = \{i, \ldots, j\}$ denote the corresponding index set. Assume that MRRR decides, by some criterion, that one should work with a new shifted factorization $L_+D_+L_+^T = LDL^T - \sigma I$, see (2.9), to compute eigenvector approximations to $v_i, \ldots, v_j$.

For the moment, we are not yet interested in the computation itself. Instead, let us rather ask what is required from the true eigenpairs of $L_+D_+L_+^T$ in question. For $k \in \Gamma$, denote those eigenvalues by μ_k, and the corresponding eigenvectors by w_k. Without rounding errors in the computation of $L_+D_+L_+^T$, one would have exact relations

$$\lambda_k - \sigma = \mu_k, \; v_k = w_k, \; k \in \Gamma. \tag{4.1}$$

In finite precision, the computation of $L_+D_+L_+^T$ involves rounding errors so (4.1) holds only in some approximate way.

Nevertheless, MRRR has decided to treat the group and its invariant subspace independent from the rest of the spectrum and the associated vectors. If

$$\mathcal{V}_\Gamma = \mathrm{span}\{v_k : k \in \Gamma\} \tag{4.2}$$

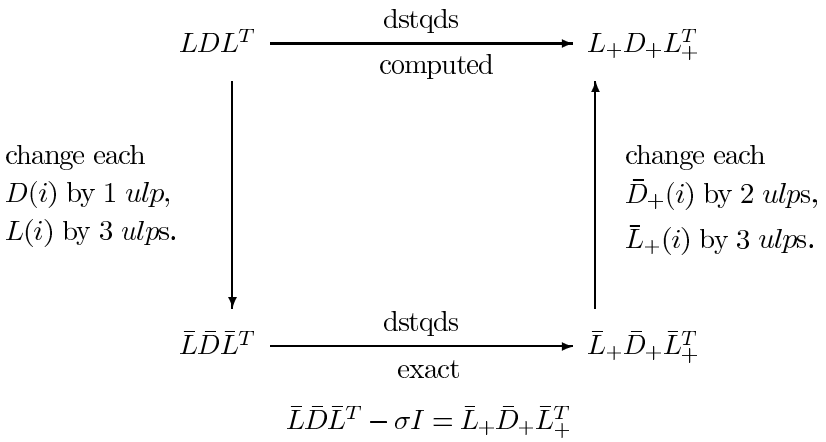

FIG. 4.1. *Commutative Diagram illustrating the mixed relative stability of the differential stationary qds (dstqds) algorithm in finite precision [8, 9, 10].*

and similarly

$$\mathcal{W}_\Gamma = \text{span}\{w_k : k \in \Gamma\}, \tag{4.3}$$

then a necessary condition for correctness in the sense of numerically orthogonal vectors as promised in (2.8) is for the subspace distance (see Section 2.6.3 of [15]) to fulfill

$$\text{dist}\,(\mathcal{V}_\Gamma, \mathcal{W}_\Gamma) = \max_{v \in \mathcal{V}_\Gamma^\perp, \|v\|_2 = 1} \; \max_{w \in \mathcal{W}_\Gamma, \|w\|_2 = 1} |v^T w| = \mathcal{O}(n\varepsilon). \tag{4.4}$$

Another interpretation of (4.4) is that the sine of the largest principal angle between $\mathcal{V}_\Gamma$ and $\mathcal{W}_\Gamma$ satisfies

$$\text{dist}\,(\mathcal{V}_\Gamma, \mathcal{W}_\Gamma) = \sin \angle\,(\mathcal{V}_\Gamma, \mathcal{W}_\Gamma) = \mathcal{O}(n\varepsilon) \tag{4.5}$$

see Section 12.4.3 of [15].

Note that this discussion does not affect singletons. Here, one can continue working with LDL^T and only the approximation errors in the computed $\hat{v}$ matter, which are controlled by (2.7) to yield (2.8).

4.2. Necessary RRR property. As mentioned in Section 2.2, the differential stationary qds (dstqds) factorization is the key tool for computing $L_+D_+L_+^T = LDL^T - \sigma I$ in MRRR and hence, we need to assess what errors are introduced in finite precision. This is where relative perturbation theory enters the scene [10, 9, 20].

Figure 4.1 gives the main result of the mixed relative stability of dstqds, see also [8, 9, 10]. $D(i)$ and $L(i)$ denote the i-th diagonal and sub-diagonal entry of D and L, respectively. First, $D(i)$ and $L(i)$ are changed by one and three relative Units in the Last Place (ulps), to obtain the respective $\bar{D}(i)$ and $\bar{L}(i)$. Second, an exact dstqds computation yields the $\bar{D}_+(i)$ and $\bar{L}_+(i)$. Last, another component-wise small relative perturbation yields $D_+(i)$ and $L_+(i)$, the results of the finite precision computation.

Like in Section 4.1, let (λ_k, v_k) and (μ_k, w_k) be the exact eigenpairs of LDL^T and computed $L_+D_+L_+^T$ respectively, where $k \in \Gamma = \{i, \ldots, j\}$. Denote by $\bar{\lambda}_i, \ldots, \bar{\lambda}_j$ and $\bar{\mu}_i, \ldots, \bar{\mu}_j$ the relevant eigenvalues of $\bar{L}\bar{D}\bar{L}^T$ and $\bar{L}_+\bar{D}_+\bar{L}_+^T$.

First, we discuss the impact of the commutative diagram on the eigenvalues. Because LDL^T is an RRR for $\lambda_i, \ldots, \lambda_j$, we can expect that

$$|\lambda_k - \bar{\lambda}_k| = \mathcal{O}(n\varepsilon|\lambda_k|), \quad k \in \Gamma. \tag{4.6}$$

Likewise, the RRR property of $L_+D_+L_+^T$ yields

$$|\mu_k - \bar{\mu}_k| = \mathcal{O}(n\varepsilon|\mu_k|), \quad k \in \Gamma, \tag{4.7}$$

where the $\bar{\mu}_k$ are the eigenvalues of $\bar{L}_+\bar{D}_+\bar{L}_+^T$. Furthermore,

$$\bar{\lambda}_k - \bar{\mu}_k = \sigma. \tag{4.8}$$

This shows that

$$|\lambda_k - (\sigma + \mu_k)| \leq |\lambda_k - \bar{\lambda}_k| + |\mu_k - \bar{\mu}_k| = \mathcal{O}\left(n\varepsilon[|\mu_k| + |\lambda_k|]\right). \tag{4.9}$$

We now proceed to investigating how the commutative diagram gives robustness of the invariant subspace in the sense of (4.4). Let the gap between the eigenvalues of LDL^t that are in Γ and those that are not be

$$\operatorname{gap}\left(\Gamma, LDL^t\right) := \min_{k\in\Gamma, l\notin\Gamma} |\lambda_k - \lambda_l|. \tag{4.10}$$

Further, in accordance to Section 3 of [9], let the subspace relative gap

$$\operatorname{relgap}\left(\Gamma, LDL^t\right) := \min_{k\in\Gamma, l\notin\Gamma} \frac{|\lambda_k - \lambda_l|}{|\lambda_k|} = \frac{\operatorname{gap}\left(\Gamma, LDL^t\right)}{|\lambda_k|}. \tag{4.11}$$

Last define, in analogy to (4.2) and (4.3), the invariant subspaces $\bar{\mathcal{V}}_\Gamma$ and $\bar{\mathcal{W}}_\Gamma$ of $\bar{L}\bar{D}\bar{L}^T$ and and $\bar{L}_+\bar{D}_+\bar{L}_+^T$, respectively.

Then Section 4 of the same paper shows that with

$$\operatorname{dist}\left(\mathcal{V}_\Gamma, \bar{\mathcal{V}}_\Gamma\right) = \mathcal{O}\left(\frac{n\varepsilon}{\operatorname{relgap}\left(\Gamma, LDL^t\right)}\right), \operatorname{dist}\left(\mathcal{W}_\Gamma, \bar{\mathcal{W}}_\Gamma\right) = \mathcal{O}\left(\frac{n\varepsilon}{\operatorname{relgap}\left(\Gamma, L_+D_+L_+^t\right)}\right), \tag{4.12}$$

and

$$\operatorname{relgap}\left(\Gamma, LDL^t\right) > \tau_{\text{minrgp}}, \ \operatorname{relgap}\left(\Gamma, L_+D_+L_+^t\right) > \tau_{\text{minrgp}}, \tag{4.13}$$

one has

$$\sin\angle\left(\mathcal{V}_\Gamma, \mathcal{W}_\Gamma\right) \leq \frac{Kn\varepsilon}{\tau_{\text{minrgp}}}. \tag{4.14}$$

The proof is based on the exact shift relation $\bar{L}\bar{D}\bar{L}^T - \sigma I = \bar{L}_+\bar{D}_+\bar{L}_+^T$ at the bottom of the commutative diagram in Figure 4.1. It implies that $\bar{L}\bar{D}\bar{L}^T$ and $\bar{L}_+\bar{D}_+\bar{L}_+^T$ have the same eigenvectors.

4.3. Addressing the peeling problem. In this section, we study one remedy for the peeling problem with the strict LAPACK 3.1 threshold. The idea is to continue using the threshold $\tau_{\text{minrgp}} = 10^{-3}$ where possible for stricter accuracy bounds. On the other hand, by adding an additional, compatible criterion, we want to reduce the size of large groups at the root level of the representation tree without falling back

to lower accuracy. The criterion should be economical and efficient at the same time: without requiring too much accuracy for the local eigenvalues, it should recognize as much in advance as possible what RRRs are needed to limit possibly redundant eigenvalue refinement.

In this regard, the alternative to a smaller threshold τ_{minrgp} is to choose a shift-invariant criterion: from standard inverse iteration [16, 19], it is known that the absolute distance between eigenvalues can be used as indicator of whether Gram-Schmidt orthogonalization is necessary. If an eigenvalue, or a cluster of eigenvalues, is well separated from its neighbors, the associated subspace will be naturally orthogonal. This can also be seen from the original Davis-Kahan bound (2.4). Second, a shift invariant criterion reflects what a human would understand by a cluster: arguably, one would group eigenvalues based on their closeness, independent of the position of the eigenvalues relative to zero. Yet, this idea has not yet been exploited for MRRR.

A natural measure of what gaps one can consider large is the following. Define the spectral diameter of an unreduced tridiagonal by

$$\text{spdiam} := \lambda_n - \lambda_1, \tag{4.15}$$

then the average spectral gap is

$$\text{avgap} := \frac{\text{spdiam}}{n-1}. \tag{4.16}$$

This yields the following grouping criterion: the non-singletons $\hat{\lambda}i, \hat{\lambda}_{i+1}$ belong to two different groups if

$$\text{rgap}(\hat{\lambda}_i) = |\hat{\lambda}_{i+1} - \hat{\lambda}_i| \geq \text{avgap}; \quad \hat{\lambda}_i, \hat{\lambda}_{i+1} \text{ no singletons.} \tag{4.17}$$

The criterion is shift-invariant as desired. It is solely aimed at creating smaller groups in the root representation. Since the gaps don't change noticeably when computing an RRR of a child, see (4.9), the criterion will not be applied again.

What can we say about eigenvalue distances passing criterion (4.17) but not the associated relative gap test (2.11)? From $\text{rgap}(\hat{\lambda}_i) < \tau_{\text{minrgp}} \cdot |\hat{\lambda}_i|$, we conclude that

$$|\lambda_i| > \frac{\text{spdiam}}{(n-1)\tau_{\text{minrgp}}}. \tag{4.18}$$

Thus the criterion effects the eigenvalues of larger magnitude, precisely those responsible for the artificial peeling problem.

Plugging LAPACK 3.0's $\tau_{\text{minrgp}} = 1/n$ into (4.18), we see that a gap larger than average, (4.17), implies (2.11) for this threshold. Thus, (4.17) delivers a tighter accuracy bound (4.5) than LAPACK 3.0. For this reason, the analysis of Section 4.2 guarantees correctness of the proposed modification. [1]

We do note that the average gap criterion only supplements the LAPACK 3.1 threshold (3.1) for groups of non-singletons. For a singleton, (3.1) stays in effect as sole criterion.

[1] When $\text{gap}\left(\Gamma, LDL^t\right) > \text{avgap}$, with the gap of the invariant subspace belonging to Γ from (4.10), one can also show the necessary robustness (4.4) by the analogous Davis-Kahan gap theorem for subspaces, see [5] and also Section 11.7.1 in [19].

5. The new representation tree in practice. We consider some benchmark problems to compare the shape of the representation trees resulting from application of the various criteria. To simplify the presentation, we name the trees with criteria (3.2) and (3.1) according to the respective LAPACK versions 3.0 and 3.1. As we anticipate that the refined tree construction from Section 4.3 will be adopted in the next LAPACK release, we call it 3.2.

As evaluation parameters, we consider the depth of the tree and the size of its largest child (apart from the root), the results are shown in Table 5.1. Note that we consider quite large matrices to illustrate notable differences between the criteria 3.0 and 3.1. A second purpose is to show the potential impact on the representation tree of parallel MRRR where one can expect to encounter eigenproblems of that size. We also show the numerical orthogonality of the eigenvectors computed by the ScaLAPACK prototype with each of the criteria [2]. Conform to what can be expected from MRRR by (2.8), we report the maximum deviation from numerical orthogonality, that is the magnitude of the largest entry of $\hat{V}^t V - I$, measured in units of $n\varepsilon$.

Matrix	Size n	Tree depth			Max. group size			Num. Orthog. $[n\varepsilon]$		
		3.0	3.1	3.2	3.0	3.1	3.2	3.0	3.1	3.2
1-2-1	10001	1	7	2	1963	8065	2198	0.87	0.94	3.18
	20001	1	17	4	3924	18019	4394	0.88	1.15	0.82
	30001	1	26	6	5886	28010	6591	0.82	1.47	2.32
	40001	1	36	8	7848	38005	8789	1.03	3.14	5.92
Wilkinson	10001	2	6	2	2	8003	3	0.34	0.25	0.34
	20001	1	11	2	2	18003	3	0.41	0.33	0.41
	30001	1	16	2	2	28003	3	0.28	0.06	0.28
	40001	2	21	2	2	38003	3	0.49	0.24	0.40
Poly	8000	0	4	4	1	3409	3409	0.32	1.33	1.33
	16000	0	9	9	1	11909	11909	0.30	12.41	12.41
LAPW	22908	3	7	6	155	19898	18843	78.62	49.99	28.69
Hubbard	63504	2	17	2	16	61735	274	749.85	265.36	130.12

TABLE 5.1

Tridiagonal test matrices and characteristics of different representation trees. '1-2-1' refers to symmetric tridiagonal Toeplitz matrices with diagonal 2. 'Wilkinson' denotes W^+_{2m+1} from (2.12). 'Hubbard', see Section 3.1, and 'Poly' are tridiagonal matrices from [23]. 'LAPW' is also from applications.

We now discuss the results in Table 5.1.

The 3.0 criterion consistently produces shallow trees. It is also remarkable that for both the 1-2-1 matrices and Wilkinson matrices, the depth of the tree stays constant even when the matrix dimension growths considerably. The Poly matrices are interesting because every eigenvalue is treated as a singleton with respect to the root representation. The infamous Hubbard matrix exhibits the worst orthogonality observed in all tests.

Criterion 3.1 exhibits the peeling problem for both Wilkinson and 1-2-1 matrices: the depth of the tree and the maximum eigenvalue group size increase substantially with increasing matrix dimension. The problem with the Hubbard matrix has already been discussed in Section 3.1, and one can defer its presence in the LAPW and Poly matrices, too, when looking at maximum group size and tree depth. The orthogonality result for the Hubbard matrix is better than for criterion 3.0. However, it is worth

pointing out that the different criteria merely influence the worst-case orthogonality *bounds*.

For Wilkinson matrices, criterion 3.2 manages to produce representation trees of identical quality as LAPACK 3.0 criterion. Furthermore, in the case of the 1-2-1 matrices, the maximum group size stays comparable and the tree depth increases much more insignificantly than when using criterion 3.1. We also note that the representation tree for the Hubbard matrix becomes benign, with a depth of two and a small maximum cluster size. The orthogonality is improved again. This gives an answer to the question raised in Section 3.3: the Hubbard matrix may be considered as not such a difficult matrix, the complexity of the representation tree with criterion 3.1 is not warranted.

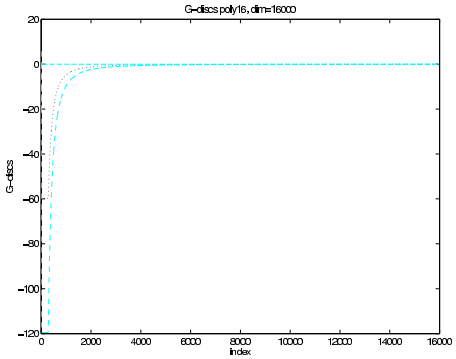

FIG. 5.1. *Gersgorin discs of Poly matrix, dimension = 16,000. The circle centers (diagonal entries) are in black, the circle radii are in cyan/grey.*

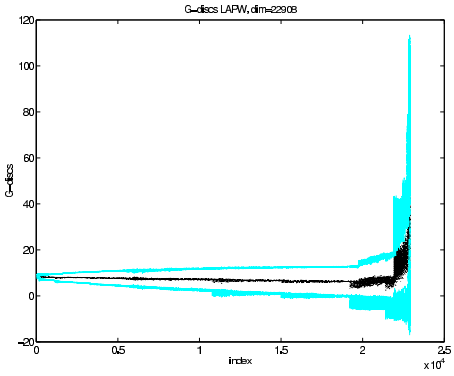

FIG. 5.2. *Gersgorin discs of LAPW matrix, dimension = 22,908. The circle centers (diagonal entries) are in black, the circle radii are in cyan/grey.*

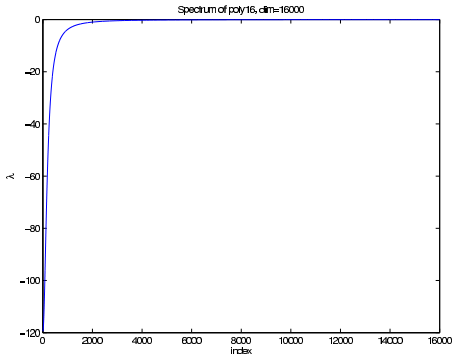

FIG. 5.3. *Spectrum of Poly matrix, dimension = 16,000.*

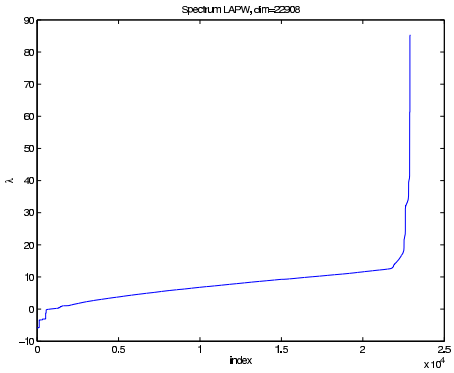

FIG. 5.4. *Spectrum of LAPW matrix, dimension = 22,908.*

The LAPW and Poly matrices are interesting as they shed light on the limits of the approach. Both criteria 3.1 and 3.2 produce a deeper tree than what criterion 3.0 deems necessary. The shape of the 3.1 and 3.2 trees can be understood from the Gersgorin discs (Figures 5.1 and 5.2) and the spectra (Figures 5.3 and 5.4). Indeed, in both cases, it is hard to define what should be a cluster. The matrix entries show

little variation in a large part of the matrix but do vary drastically in one smaller section. This is also reflected in the eigenvalues.

Do the matrices warrant the more conservative representation trees of criteria 3.1 and 3.2? The answer depends on the point of view. For example, take the Poly matrix of size 16,000. In retrospect and in the light of the orthogonality result for the 3.0 criterion, it turns out to be acceptable to allow relative gaps to be as small as $1/16000 = 6.25 \cdot 10^{-5}$. Prior to the computation, and only equipped with information on the spectrum, one might agree that 11,909 eigenvalues in the middle of the spectrum contained in an interval of width less than .75% of the spectral diameter do form a cluster that needs to be refined gradually. This example illustrate the difficulty of choosing the 'right' criterion, not too conservative but accurate enough, for all cases.

6. Summary and conclusions. MRRR in LAPACK 3.0 used a representation tree threshold inverse-proportional to the dimension of the unreduced tridiagonal. In order to better prevent deteriorating orthogonality with increasing matrix size, and also to allow a more flexible, independent setting, LAPACK 3.1 replaced the threshold by a fixed number.

Prompted by a performance comparison of several parallel eigensolvers [23], we investigate in this paper how the complexity of MRRR depends on the threshold choice and the associated representation tree. As a first contribution, we show that the representation tree construction in LAPACK 3.1 is prone to producing trees with an artificial peeling problem.

This raises the issue of finding an economical and efficient criterion for discovering when a new RRR can or should be used. This is not an easy task as depending on the matrix at hand, it might be difficult to define what eigenvalue groups should be considered clustered.

As a remedy, we propose extending the LAPACK 3.1 criterion to recognize when groups are separated by a gap that is larger than average. It is shown that this modification fits in the frame-work of MRRR theory. In particular, the invariant subspace of a group that is separated in this way can be computed independently.

We present matrices that show the limitations of the average gap approach: complicated representation trees are not always avoided. Nevertheless, experimental data shows that our proposed solution often produces shallow trees and small clusters, favorable qualities that are important for guaranteeing good complexity of both sequential and parallel MRRR. At the same time, numerical orthogonality is similar to or better than for LAPACK 3.1. For these reasons, we plan on adapting our new criterion in LAPACK as well as in our parallel MRRR algorithm for ScaLAPACK.

Last but not least, the poor complexity of the ScaLAPACK MRRR prototype on the infamous Hubbard example reported in [23] is explained. The proposed refined representation tree resolves the artifical peeling problem without sacrificing accuracy.

Acknowledgment. Thanks to Bob Ward and Yihua Bai for providing the interesting industrial example matrices from [23] in tridiagonal form, to Paul Willems for comments on a first version of this paper, and to Beresford Parlett for many discussions and remarks.

REFERENCES

[1] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, and D. Sorensen. *LAPACK User's Guide*. SIAM, Philadelphia, 3. edition, 1999.

[2] D. Antonelli and C. Vömel. LAPACK working note 168: PDSYEVR. ScaLAPACK's parallel MRRR algorithm for the symmetric eigenvalue problem. Technical Report UCBCSD-05-1399, University of California, Berkeley, 2005.

[3] P. Bientinesi, I. S. Dhillon, and R. A. van de Geijn. A Parallel Eigensolver for Dense Symmetric Matrices based on Multiple Relatively Robust Representations. *SIAM J. Sci. Comput.*, 27(1):43–66, 2005.

[4] C. Davis and W. Kahan. Some new bounds on perturbation of subspaces. *Bull. Amer. Math. Soc.*, 75:863–868, 1969.

[5] C. Davis and W. Kahan. The rotation of eigenvectors by a perturbation. III. *SIAM J. Numer. Anal.*, 7(1):1–47, 1970.

[6] J. W. Demmel, O. A. Marques, B. N. Parlett, and C. Vömel. A Testing Infrastructure for Symmetric Tridiagonal Eigensolvers. Technical report LBNL-61831, Lawrence Berkeley National Laboratory, 2006.

[7] J. W. Demmel, O. A. Marques, B. N. Parlett, and C. Vömel. Lapack Working Note 183: Performance and Accuracy of LAPACK's Symmetric Tridiagonal Eigensolvers. Technical report, UC Berkeley, 2006.

[8] I. S. Dhillon. *A New $O(n^2)$ Algorithm for the Symmetric Tridiagonal Eigenvalue/Eigenvector Problem*. PhD thesis, University of California, Berkeley, California, 1997.

[9] I. S. Dhillon and B. N. Parlett. Multiple representations to compute orthogonal eigenvectors of symmetric tridiagonal matrices. *Linear Algebra and Appl.*, 387:1–28, 2004.

[10] I. S. Dhillon and B. N. Parlett. Orthogonal eigenvectors and relative gaps. *SIAM J. Matrix Anal. Appl.*, 25(3):858–899, 2004.

[11] I. S. Dhillon, B. N. Parlett, and C. Vömel. Glued matrices and the MRRR algorithm. *SIAM J. Sci. Comput.*, 27(2):496–510, 2005. Revised version of LAPACK Working Note 163.

[12] I. S. Dhillon, B. N. Parlett, and C. Vömel. The design and implementation of the MRRR algorithm. *ACM Trans. Math. Software*, 32(4):533–560, 2006.

[13] K. V. Fernando and B. N. Parlett. Accurate singular values and differential *qd* algorithms. *Numerische Mathematik*, 67:191–229, 1994.

[14] K. V. Fernando and B. N. Parlett. Implicit Cholesky algorithms for singular values and vectors of triangular matrices. *Numerical Linear Algebra with Applications*, 2(6):507–531, 1995.

[15] G. H. Golub and C. van Loan. *Matrix Computations*. The John Hopkins University Press, Baltimore, Maryland, 3. edition, 1996.

[16] I. C. F. Ipsen. Computing an eigenvector with inverse iteration. *SIAM Review*, 39(2):254–291, 1997.

[17] Lapack 3.1. http://www.netlib.org/lapack/lapack-3.1.0.changes, 2006.

[18] B. N. Parlett. *Acta Numerica*, chapter The new qd algorithms, pages 459–491. Cambridge University Press, Cambridge, UK, 1995.

[19] B. N. Parlett. *The Symmetric Eigenvalue Problem*. SIAM Press, Philadelphia, PA, 1998.

[20] B. N. Parlett. A bidiagonal matrix determines its hyperbolic SVD to varied relative accuracy. *SIAM J. Matrix Anal. Appl.*, 26(4):1022–1057, 2005.

[21] B. N. Parlett and I. S. Dhillon. Fernando's solution to Wilkinson's problem: an application of double factorization. *Linear Algebra and Appl.*, 267:247–279, 1997.

[22] B. N. Parlett and I. S. Dhillon. Relatively robust representations of symmetric tridiagonals. *Linear Algebra and Appl.*, 309(1-3):121–151, 2000.

[23] R. C. Ward and Y. Bai. Performance of Parallel Eigensolvers on Electronics Structure Calculations II. Technical Report UT-CS-06-572, University of Tennessee, Knoxville, TN, USA, 2006.

[24] J. H. Wilkinson. *The Algebraic Eigenvalue Problem*. Oxford University Press, Oxford, 1965.