Introduction

Overview

The standard bulge-chasing QR algorithm for a symmetric matrix A proceeds as follows:

① Reduce A to tridiagonal T by Householder similarity transformations.

② Choose a shift and apply a sweep of Givens similarity transformations within the active block,
chasing the temporary second-band bulge from top to bottom
and restoring tridiagonal form (Watkins [1]).

③ Between sweeps, after tridiagonal form has been restored,
scan the first off-diagonal band for conventional deflation opportunities (LAPACK [2]).
Whenever an off-diagonal entry becomes sufficiently small, typically near the end of the active block,
set it to zero and split the block at that position.

④ Repeat the process until every active block has size 1×1.

Why does QR iteration converge?

Within a fixed active block, let T_k denote the tridiagonal matrix after k sweeps.
First consider the unshifted iteration.
At step j, write $T_j = Q_j R_j$, where Q_j is orthogonal and R_j is upper triangular.
The next iterate is $T_{j+1} = R_j Q_j$.
Let $U_k = Q_0 Q_1 \cdots Q_{k-1}$ be the product of the orthogonal factors generated during the first k steps.
Repeated substitution gives
$T_0^k = U_k (R_{k-1} R_{k-2} \cdots R_0)$.
Thus U_k is the orthogonal factor obtained by repeatedly multiplying an orthogonal basis by T_0
and orthonormalizing it. This is simultaneous power iteration (Watkins [1]).

If $T_0 = V \Lambda V^T$, then

$T_0^k = V \Lambda^k V^T$.

Each eigenvector component is therefore multiplied by λ_j^k.

When the eigenvalue magnitudes are distinct, their relative weights separate, just as in the ordinary power method.

Repeated orthonormalization extracts the corresponding orthogonal invariant directions.

Consequently,

$$T_k = U_k^T T_0 U_k \to \Lambda,$$

so the off-diagonal entries tend to zero.

Why is Q_k formed from these rotations?

Bulge chasing computes Q_k implicitly as a product of adjacent Givens rotations.

The first rotation is determined by the first two entries of the shifted leading column of T_k.

Applying it as a similarity transformation creates one temporary second-band entry, called the bulge.

Each subsequent rotation removes the current bulge but creates a new bulge one position lower.

After the final rotation, the bulge leaves the bottom of the active block,

and the matrix is again symmetric and tridiagonal (Watkins [1]).

What does the shift add?

In practice, a shift μ_k is used to accelerate the underlying convergence process.

The usual Wilkinson shift is the eigenvalue of the trailing 2×2 block that is closer to the bottom-right diagonal entry.

The shift replaces the power-method factors λ_j by $\lambda_j - \mu_k$ during the current sweep.

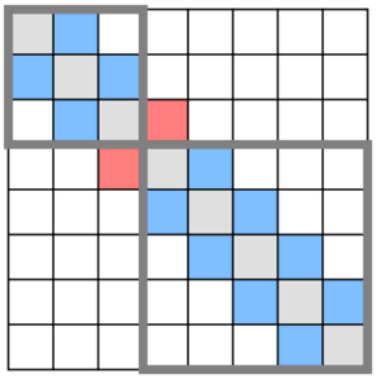
Because the Wilkinson shift lies close to a trailing eigenvalue,

the corresponding factor is small and the trailing eigendirection separates more rapidly.
Its remaining off-diagonal coupling therefore approaches zero faster,

permitting earlier deflation (Watkins [1]).

Convergence through splitting

The calculation is complete when recursive deflation has reduced every active block to size 1×1.
Splitting is therefore an essential mechanism of convergence.
When a first-band coupling is set to zero, one active eigenvalue problem becomes two smaller independent problems.
The algorithm then continues separately on those blocks until no nontrivial block remains.



Additional deflation opportunities

Conventional deflation requires the coupling at a proposed cut to be numerically negligible already.
We investigate whether some entries that fail this test can nevertheless be removed by one Jacobi rotation.
The rotation annihilates the selected first-band entry exactly but creates two temporary second-band entries.

If both new entries are below the accepted tolerance, they may be discarded and the matrix splits numerically.

We call this augmented bulge-chasing strategy Chase And Trim (CAT).
Testing every possible cut after every sweep
can impose a substantial cumulative cost.
We therefore use several split-selection and acceptance strategies:

- Accept every spontaneous conventional split, then test and apply only non-intersecting active splits
- Retain the local conventional-deflation tolerances τ for reuse in the active-split tests
- Reject candidates that cannot satisfy the acceptance inequality
- Construct and apply the Jacobi rotations only after the non-intersecting candidates have been accepted

Parallel implementation affords additional benefits:

- The cost of testing can be reduced
- Earlier splits create independent blocks sooner, exposing parallelism earlier

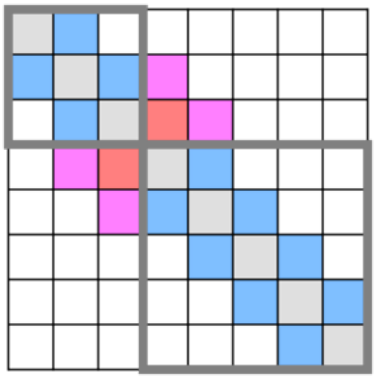
Jacobi rotation and fill-in

Consider a proposed cut between indices k and k + 1.
The local tridiagonal entries are

$$a = e_{k-1}, \quad b = e_k, \quad g = e_{k+1},$$

with diagonal entries d_1 and d_2 on the two sides of the cut.

A single Jacobi rotation acts on indices k and k + 1
and is chosen to annihilate the pivot b.
The highlighted first-band entry in the image is the pivot,
and the two highlighted second-band pairs are the possible fill-in.



The complete local similarity transformation is shown below.

$$T = \begin{bmatrix} \cdot & \bullet & 0 & 0 & 0 & 0 & 0 & 0 \\ \bullet & \cdot & a & 0 & 0 & 0 & 0 & 0 \\ 0 & a & d_1 & b & 0 & 0 & 0 & 0 \\ 0 & 0 & b & d_2 & g & 0 & 0 & 0 \\ 0 & 0 & 0 & g & \cdot & \bullet & 0 & 0 \\ 0 & 0 & 0 & 0 & \bullet & \cdot & \bullet & 0 \\ 0 & 0 & 0 & 0 & 0 & \bullet & \cdot & \bullet \\ 0 & 0 & 0 & 0 & 0 & 0 & \bullet & \cdot \end{bmatrix} \qquad G = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c & -s & 0 & 0 & 0 & 0 \\ 0 & 0 & s & c & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T' = G^{\intercal} T G = \begin{bmatrix} \cdot & \bullet & 0 & 0 & 0 & 0 & 0 & 0 \\ \bullet & \cdot & ca & -sa & 0 & 0 & 0 & 0 \\ 0 & ca & d_1' & 0 & sg & 0 & 0 & 0 \\ 0 & -sa & 0 & d_2' & cg & 0 & 0 & 0 \\ 0 & 0 & sg & cg & \cdot & \bullet & 0 & 0 \\ 0 & 0 & 0 & 0 & \bullet & \cdot & \bullet & 0 \\ 0 & 0 & 0 & 0 & 0 & \bullet & \cdot & \bullet \\ 0 & 0 & 0 & 0 & 0 & 0 & \bullet & \cdot \end{bmatrix}$$

The Jacobi rotation sets the pivot positions to zero.
The neighboring first-band entries become c a and c g,
while the only new second-band entries have magnitudes | s a | and | s g |.
At an endpoint of the active block, the absent neighboring coupling is taken as zero.

To construct the rotation, define $\Delta = d_1 - d_2$ and $t = \frac{s}{c}$

The condition that the rotated pivot vanish is
$b\,(1 - t^2) = \Delta t$.

Choosing the shorter rotation and Jacobi root with $|t| \le 1$ gives

$$t = \frac{2b}{\Delta + \operatorname{sign}(\Delta)\sqrt{\Delta^2 + 4b^2}}$$

with $|t| = 1$ when $\Delta = 0$ and $t = 0$ when $b = 0$.
The rotation coefficients are then

$$c = \frac{1}{\sqrt{1 + t^2}}, \quad s = t\,c.$$

The rotation produces a numerical split precisely when both fill-in magnitudes are no larger than the accepted tolerance τ:

$$|\,s\,a\,| \le \tau \quad \text{and} \quad |\,s\,g\,| \le \tau.$$

We use the same local threshold as the conventional deflation test:

$$\tau = \varepsilon\,(\,|\,d_1\,| + |\,d_2\,|\,),$$

where ε is the relative tolerance chosen for the calculation.

For the proposed split to be accepted, both new fill-in magnitudes must be no larger than τ:

$$|\,s\,a\,| \le \tau \quad \text{and} \quad |\,s\,g\,| \le \tau.$$

Defining $M = \max(\,|\,a\,|, |\,g\,|\,)$, these two requirements combine into the single condition

$$|\,s\,|\,M \le \tau.$$

Since any active symmetric 2×2 block is diagonalized directly by one Jacobi rotation,
the recursive stopping criterion becomes active-block size no larger than 2.

Below is example of a boundary case where d_1 is the first diagonal entry. The left neighboring coupling a is absent, so neither ca nor the fill-in −sa is produced.

$$T = \begin{bmatrix} d_1 & b & 0 & 0 & 0 & 0 & 0 & 0 \\ b & d_2 & g & 0 & 0 & 0 & 0 & 0 \\ 0 & g & \bullet & \bullet & 0 & 0 & 0 & 0 \\ 0 & 0 & \bullet & \bullet & \bullet & 0 & 0 & 0 \\ 0 & 0 & 0 & \bullet & \bullet & \bullet & 0 & 0 \\ 0 & 0 & 0 & 0 & \bullet & \bullet & \bullet & 0 \\ 0 & 0 & 0 & 0 & 0 & \bullet & \bullet & \bullet \\ 0 & 0 & 0 & 0 & 0 & 0 & \bullet & \bullet \end{bmatrix} \qquad G = \begin{bmatrix} c & -s & 0 & 0 & 0 & 0 & 0 & 0 \\ s & c & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T' = G^{\intercal} T G = \begin{bmatrix} d_1' & 0 & sg & 0 & 0 & 0 & 0 & 0 \\ 0 & d_2' & cg & 0 & 0 & 0 & 0 & 0 \\ sg & cg & \bullet & \bullet & 0 & 0 & 0 & 0 \\ 0 & 0 & \bullet & \bullet & \bullet & 0 & 0 & 0 \\ 0 & 0 & 0 & \bullet & \bullet & \bullet & 0 & 0 \\ 0 & 0 & 0 & 0 & \bullet & \bullet & \bullet & 0 \\ 0 & 0 & 0 & 0 & 0 & \bullet & \bullet & \bullet \\ 0 & 0 & 0 & 0 & 0 & 0 & \bullet & \bullet \end{bmatrix}$$

Only the right neighboring coupling remains,
producing cg and the single fill-in sg.

Why two fill-ins do not mean more accepted error

We compare the spectral norms of the perturbation matrices discarded by conventional and active splitting (Golub & Van Loan [12]).

This norm measures the largest factor by which a matrix can increase the Euclidean length of a vector.

After a Jacobi rotation, the local submatrix $T' =$

$$\begin{bmatrix} \bullet & ca & -sa & 0 \\ ca & d_1' & 0 & sg \\ -sa & 0 & d_2' & cg \\ 0 & sg & cg & \bullet \end{bmatrix}$$

An active split discards $-sa$ and sg, giving $\tilde{T} =$

$$\begin{bmatrix} \bullet & ca & 0 & 0 \\ ca & d_1' & 0 & 0 \\ 0 & 0 & d_2' & cg \\ 0 & 0 & cg & \bullet \end{bmatrix}$$

and $T' = \tilde{T} + E$

where $E = \begin{bmatrix} 0 & 0 & -sa & 0 \\ 0 & 0 & 0 & sg \\ -sa & 0 & 0 & 0 \\ 0 & sg & 0 & 0 \end{bmatrix}$ is the discarded perturbation

For any unit vector $\hat{v} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}$ $\quad E\,\hat{v} = \begin{bmatrix} -sa\,v_3 \\ sg\,v_4 \\ -sa\,v_1 \\ sg\,v_2 \end{bmatrix}$

Therefore

$$\| E\,\hat{v} \|_2^2 = (s\,a)^2 (v_1^2 + v_3^2) + (s\,g)^2 (v_2^2 + v_4^2).$$

Denote $m = \max(|\,s\,a\,|, |\,s\,g\,|)$

$$\downarrow$$

$$\| E\,\hat{v} \|_2^2 \le m^2 (v_1^2 + v_2^2 + v_3^2 + v_4^2) = m^2 \| \hat{v} \|_2^2$$

$$\downarrow$$

$$\| E \|_2 \le m.$$

The active-split acceptance conditions
$|\,s\,a\,| \le \tau$ and $|\,s\,g\,| \le \tau$
therefore imply $\| E \|_2 \le \tau$.

For comparison, conventional deflation discards a single entry b: (E passive) = $\begin{bmatrix} 0 & b \\ b & 0 \end{bmatrix}$

For $\hat{v} = [v_1, v_2]^T$, (E passive) $\hat{v} = [b\, v_2, b\, v_1]^T$,

↓

$\| \text{(E passive) } \hat{v} \|_2^2 = b^2 (v_1^2 + v_2^2) = b^2 \| \hat{v} \|_2^2$,

↓

$\| \text{E passive} \|_2 = | b |$.

Conventional deflation requires $| b | \leq \tau$,
so both rules impose a similar spectral-norm bound on E:

- conventional split: $\| \text{E passive} \|_2 = | b | \leq \tau$
- active split: $\| E \|_2 = \max(| s\, a |, | s\, g |) \leq \tau$.

Even though the active split discards two symmetric fill-in couplings rather than one,
fill-in occupies disjoint coordinate pairs,
and the spectral norm is determined by the larger quantity.

If we require only $| b | \leq \tau$, the discarded perturbation contains b
as well as the two fill-ins, so its spectral norm can exceed τ.
We tested this relaxed rule, and it produced very few additional split opportunities.

From the exact condition to an acceptance test

The exact numerical-split condition was derived to be
$| s | M \leq \tau$,

where $M = \max(|a|, |g|)$

τ is normally computed to evaluate conventional deflation opportunities.

Since the cost of computing t, s, and c is substantial, we look for acceptance inequalities to determine whether a rotation can produce a valid split.

Since $|s| \le 1$, any candidate with
$M \le \tau$
automatically satisfies $|s| M \le \tau$ and is accepted immediately.

The remaining candidates therefore satisfy $M > \tau$.

The Jacobi rotation equation is
$b(1 - t^2) = \Delta t$, where $\Delta = d_1 - d_2$.

Since $s = t c$,
$|\Delta| |s| = |\Delta t| c = |b| (1 - t^2) c$.

Since $|t| \le 1$ and $0 < c \le 1$, $|\Delta| |s| \le |b|$.

$$\text{For } |\Delta| > 0, |s| \le \frac{|b|}{|\Delta|}$$

Substitute the upper bound of $|s|$ into the exact numerical-split condition:

$$\text{if } \left(\frac{|b|}{|\Delta|} \right) M \le \tau, |s| M \le \tau,$$

and the rotation produces a valid split.

Equivalently,
$|b| M \le |\Delta| \tau$
is a sufficient acceptance condition for a valid split.

Candidates failing this acceptance condition may still satisfy the exact condition.

Summary: a one-rotation split is favored by
• a small pivot $|b|$, so only a small rotation is required
• small neighboring couplings $|a|$ and $|g|$, so the rotation creates only small fill
• a large local diagonal separation $|\Delta| = |d_1 - d_2|$

These opportunities tend to appear as ordinary bulge chasing progresses, allowing one-rotation deflation to stay ahead of conventional deflation.

Why were so few exact opportunities missed?

The algorithm accepts a candidate when the sufficient condition
$|b| M \le |\Delta| \tau$
holds.

The exact one-rotation split condition is
$|s| M \le \tau$,
where s is the sine of the Jacobi rotation and M is the larger neighboring coupling.

$$\text{Because } |s| \le \frac{|b|}{|\Delta|},$$

the CAT condition satisfies the exact inequality, but the converse need not hold.

We therefore expected the exact condition to identify a substantial number of

one-rotation splits.
When the diagnostic found only a few missed opportunities over the full test set, we initially searched for an error and then examined the gap analytically.

A false negative requires

$$|s| M \le \tau$$

and

$$\frac{|b| M}{|\Delta|} > \tau$$

Its possibility is therefore determined by the gap between $|s|$ and $\frac{|b|}{|\Delta|}$

With $t = \tan\theta$ and $|t| \le 1$,

$$|s| = \frac{|t|}{\sqrt{1+t^2}},$$

and the exact Jacobi rotation equation gives

$$\frac{|b|}{|\Delta|} = \frac{|t|}{1-t^2}$$

For small $|t|$, both quantities equal $|t|$ to first order,
and their difference is $O(t^3)$.

Thus, for small Jacobi rotations, the rigorous CAT upper bound on $|s|$ lies very close to the exact value of $|s|$.
When $M \gg \tau$, an admissible rotation must be small,
so the region satisfying the exact condition but rejected by the CAT condition is correspondingly narrow.

This explains the unexpectedly small number of false negatives found by the diagnostic.

Evaluating the acceptance inequality without forming products

The acceptance inequality is
$| b | M \leq | \Delta | \tau$.

Before forming the products, we test whether the inequality can be decided from comparisons of its factors.

A candidate reaches this test only after conventional deflation has failed at b:
$| b | > \tau$.

It has also failed automatic acceptance:
$M > \tau$.

If $| \Delta | \leq | b |$, then
$| \Delta | \tau \leq | b | \tau < | b | M$,
so the acceptance inequality cannot hold.

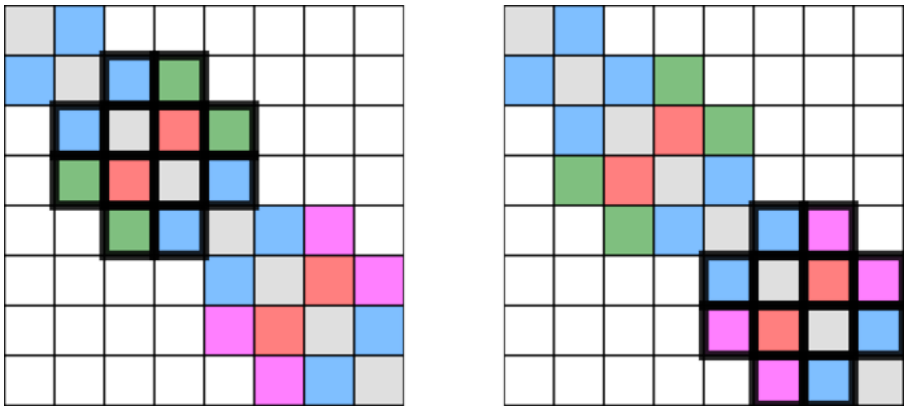
Similarly, if $| \Delta | \leq M$, then
$| \Delta | \tau \leq M \tau < | b | M$,
because $| b | > \tau$.

Therefore a necessary condition for the acceptance inequality to hold is
$| \Delta | > \max(| b |, M)$.

Define the diagonal-difference threshold
$T = \max(|b|, M)$.
Only candidates satisfying this necessary condition proceed to the product comparison.

How can we process active splits in parallel?

More than one split within a block can be accepted before any rotation is performed,
provided that the data required by their tests do not intersect.



Legend

- highlighted on left image: entries affected by rotation at cut 3
- highlighted on right image: entries affected by rotation at cut 6
- gray: diagonal entries
- blue: first off-diagonal band

• red at cut 3: accepted active split
• red at cut 6: additional active split being tested
• green: fill-in around the first split, already found negligible
• magenta: fill-in that must be tested for the additional split

A rotation at cut $k = 3$ changes the diagonal entries d_k and d_{k+1}

and the first-band couplings b_{k-1}, b_k, and b_{k+1}.

A candidate at cut $j = 6$ uses

d_j, d_{j+1}, b_{j-1}, b_j, and b_{j+1}.

Here $|j - k| = |6 - 3| = 3$.
Thus the two tests share none of these quantities.

If instead $|j - k| = 2$,
the two tests still share one first-band coupling.
Therefore simultaneous active splits must satisfy
$|j - k| \geq 3$.

The candidate cuts can therefore be partitioned into three non-intersecting rows:

row 1: i, i + 3, i + 6, ...
row 2: i + 1, i + 4, i + 7, ...
row 3: i + 2, i + 5, i + 8, ...

Within each row, the data used by the candidate tests do not intersect.
All candidates in that row can therefore be tested in parallel.

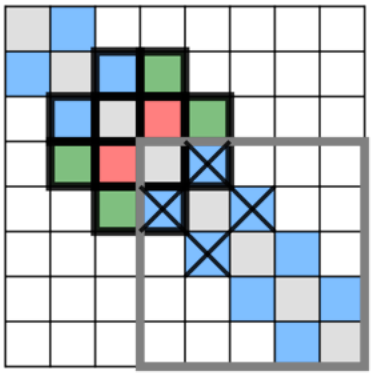
The successful rotations in the same row also have non-intersecting neighborhoods,
so they can be applied in parallel.

The three rows are processed successively.
After the successful rotations in one row are applied, the next row is tested using the updated matrix.
After the three rows have been processed,
the accepted splits divide the block into independent subblocks,
which can then continue independently and in parallel.

Rescanning newly created subblocks

① After an active split

- wide outline: the child block
- red pivot was zeroed by the rotation
- green is the negligible fill-in from the rotation
- blue crossed positions: the candidate cuts to be re-tested



Why two new candidates?

For an interior candidate at cut j, the test uses

d_j, d_{j+1}, b_{j-1}, b_j, and b_{j+1},

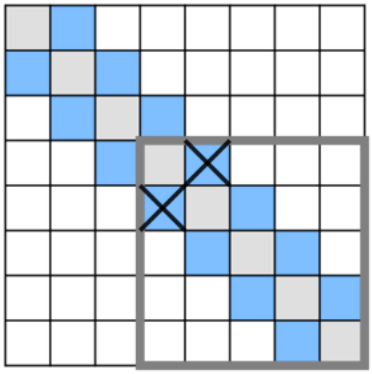
with a missing neighboring coupling taken as zero

- At the first cut, $j = k + 1$, the test uses d_{k+1} and b_{k+1}.
- At the second cut, $j = k + 2$, the test still uses b_{k+1}.

• For $j \geq k + 3$, the test uses none of the changed quantities.

• By symmetry, only the last two cuts of a left child need to be retested.
• A child bounded by active splits on both sides requires retesting the first two and the last two cuts.

② After a passive split no rotation is performed, so no interior entry is recolored
• wide outline: the child block.
• blue crossed position: the only candidate cut to be retested



Why only one new candidate?

For an interior candidate at cut j, the test uses

$$d_j,\ d_{j+1},\ b_{j-1},\ b_j, \text{ and } b_{j+1},$$

with a missing neighboring coupling taken as zero.

• For a right child, at $j = k + 1$, the quantity b_{j-1} lies outside the child and is now taken as zero.
• For $j \geq k + 2$, the test uses no changed quantity.

• By symmetry, only the last cut of a left child needs to be retested.
• A child bounded by passive splits on both sides requires retesting only the first and the last cuts.

For simplicity, the implementation uses the two-cut boundary window for both

active and passive cuts.
This adds a small amount of redundant work but does not increase the arithmetic span.

The recursive call therefore carries information on inspection scope:

- All: inspect every cut after a QR sweep
- None: no pre-sweep inspection is needed
- Left: inspect the first two cuts
- Right: inspect the last two cuts
- Left and right: inspect the first two and the last two cuts

Eigenvectors: work and span

If eigenvectors are required, every orthogonal rotation must also be accumulated.

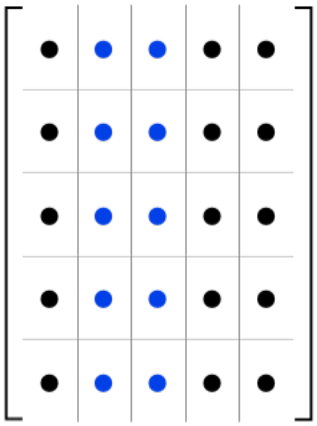
- conventional deflation sets an already negligible entry to zero and requires no eigenvector update
- active CAT deflation performs an additional rotation, which must be applied to the eigenvector matrix.

The additional work is substantial, but its contribution to dependency span is small.

If V is the accumulated eigenvector matrix, an additional CAT rotation G requires

$$V' = V\,G.$$

V G: update two columns



Right multiplication V G updates only columns k and k + 1.
For every row i,

- $v'_{ik} = c\, v_{ik} + s\, v_{ik+1}$

- $v'_{ik+1} = -s\, v_{ik} + c\, v_{ik+1}$.

First, all 4n multiplications required to update both columns can proceed concurrently.
Second, all 2n additions or subtractions can also be done concurrently.
Therefore the complete eigenvector update has arithmetic span 2, independent of n.

Parallelism can also be used at a higher level.
As shown above, the candidate cuts are partitioned into three non-intersecting candidate rows:

- candidate row 1: i, i + 3, i + 6, ...
- candidate row 2: i + 1, i + 4, i + 7, ...
- candidate row 3: i + 2, i + 5, i + 8, ...

All successful rotations whose cuts belong to the same candidate row act on disjoint pairs of eigenvector columns and can therefore be applied

simultaneously.

Therefore, if all three candidate rows contain successful rotations, their total eigenvector-accumulation span is
$2 + 2 + 2 = 6$.

Since the present study is already broad in scope, eigenvector accumulation is covered only conceptually.

Matrix families used for testing

Matrix family	Construction for the experiment	Typical setting	Source if applicable
General dense symmetric No imposed structure	• Diagonal entries sampled uniformly from [−2.0, 2.0] • Off-diagonal entries sampled uniformly from [−2.0, 2.0], mirrored across the diagonal	Generic baseline	
Dense correlation Positive-definite with unit diagonal	• Eigenvalues sampled uniformly from [0.5, 1.5] • A is formed as $Q\,D\,Q^T$, where D is the diagonal eigenvalue matrix and Q is generated as a product of randomized plane rotations • Rows and columns rescaled so that all diagonal entries equal 1	Correlation analysis for standardized multivariate data	(Davies & Higham [5])
Dense clustered spectrum Eigenvalues concentrated in 4 narrow clusters	• Eigenvalues generated in 4 clusters centered at -1.5, -0.5, 0.5, 1.5 with independent jitter in [−0.001, 0.001] • A is formed as $Q\,D\,Q^T$, where D is the diagonal eigenvalue matrix and Q is generated as a product of randomized plane rotations	Closely spaced vibration modes or energy levels	(LAPACK [4])
Dense graded spectrum Positive-definite with a geometrically graded spectrum	• Eigenvalue exponents evenly spaced from -3.0 to 3.0 ↓ Eigenvalues range from 0.001 to 1000.0 with condition number 1,000,000 • A is formed as $Q\,D\,Q^T$, where D is the diagonal eigenvalue matrix and Q is generated as a product of randomized plane rotations	Ill-conditioned problems with widely separated spectral scales	(LAPACK [4])
General symmetric tridiagonal No imposed weak links	• Diagonal entries sampled uniformly from [−2.0, 2.0] • First off-diagonal entries sampled uniformly from [−2.0, 2.0] and mirrored	Generic tridiagonal baseline	
Weak-link symmetric tridiagonal Deliberately inserted weak couplings	• Diagonal entries sampled uniformly from [−2.0, 2.0] • Ordinary first-band entries sampled uniformly from [−2.0, 2.0] • Every 8th coupling has random sign and magnitude in [2e-11, 1e-10] with the position shifted between samples • Conventional deflation uses $\tau = \varepsilon(\lvert d_1\rvert + \lvert d_2\rvert)$ • Initially $\tau < 4e\text{-}12$	Controlled near-split stress test with weak links still above conventional deflation tolerance	(Demmel et al. [14])

	so even the smallest inserted weak link is at least 5.0 times the largest possible initial τ		
Wilkinson tridiagonal	• For $n = 2m + 1$ • diagonal = $m, m-1, \ldots, 1, 0, 1, \ldots, m-1, m$ • First off-diagonal entries equal 1	Classical symmetric tridiagonal eigensolver test matrix	(Wilkinson [13])
Unequal variance-covariance Positive-definite with strongly unequal variances	• Begin with a dense correlation matrix R • Choose standard deviations s_i log-uniformly from [1.0, 10.0] • Form $S = \mathrm{diag}(s_1, \ldots, s_n)$ ↓ diagonal variances range from 1.0 to 100.0 • $A = S\,R\,S$	Studies for continuity with the earlier Householder–Jacobi study	(Davies & Higham [5])

Dense inputs are reduced to symmetric tridiagonal form by Householder similarity transformations before QR iteration.
The same reduction is used by the baseline and candidate algorithms, so its arithmetic cost is excluded from the comparison.

The tridiagonal families are generated directly in tridiagonal form.

For each matrix family, 3 deterministic matrices are tested at n = 63, 255, 511, 1023.
For the Wilkinson family one matrix is tested at each size.

Methods

①

Preprocessing

For dense inputs, both algorithms begin with Householder tridiagonalization. Because this reduction is common to both, its arithmetic cost is excluded from the comparison.

②

Convergence criteria

- A 1 x 1 block is complete
- A 2 x 2 block is diagonalized directly by a single Jacobi rotation

The same exit conditions are used in both algorithms.

- A passive split is accepted if the subdiagonal coupling satisfies $|b| \le \tau$
- An active split is accepted if both Jacobi rotation fill-ins satisfy $|s\,a| \le \tau$ and $|s\,g| \le \tau$

where $\tau = \varepsilon\,(|d_1| + |d_2|)$, with ε = 1e-12.

③

Relative weights of arithmetic operations

Operation	Weight	Source if available
floating-point addition	1	(Fog [3])
floating-point subtraction	1	(Fog [3])
floating-point multiplication	1	(Fog [3])
floating-point division	4	(Fog [3])
$\text{hypot}(x, y) = \sqrt{x^2 + y^2}$	8	hardware-optimized 8 is our estimate

④

Arithmetic span concept and use

• If operations can proceed independently, span contributions are combined by taking the maximum
• If operations must proceed successively, span contributions are added

(Brent [11])

All reported arithmetic spans are conservatively modeled at ceiling.
For instance, every evaluated active-split candidate is assigned the full test span of 2.

The span model exploits parallelism aggressively, but even the widest stages in the tested sizes
require only a few thousand simultaneous arithmetic operations,
so processor availability is unlikely to be the limiting factor on massively parallel hardware.

⑤

For hypot function hypot(x, y) = $\sqrt{x^2 + y^2}$:

• work weight is 8
• span weight is 6

In this illustrative span model, the two squarings can proceed concurrently,
followed by one addition and a square root of weight 4,
so the modeled hypot span is 1 + 1 + 4 = 6.

⑥

For a bulge-chasing sweep on an m×m block

For a natural QR iteration on a symmetric tridiagonal matrix, our source gives

4m additions, 10m multiplications, 2m divisions, and 1m square-root evaluations (Reinsch [10]).

The implementation uses hypot for the corresponding normalization.
Counting every arithmetic operation equally gives 17m unit-weight operations.

- Using the span weights defined above gives 28m weighted span.
- For the span model used here, each QR sweep is treated as a sequential black box,

so its assigned span is 28m.

The same sweep model is used for the baseline and the candidate algorithms.

⑦

Concurrency after splits

For independent blocks, arithmetic span is the maximum block span.

⑧

Concurrency during active-split evaluation and execution is described in the next section.

Arithmetic span during active-split evaluation and execution

Candidate cuts are partitioned into three rows:

row 1: i, i + 3, i + 6, ...
row 2: i + 1, i + 4, i + 7, ...

row 3: i + 2, i + 5, i + 8, ...

Cuts within one row have non-intersecting local neighborhoods.
The three rows are processed successively because each row is evaluated using the matrix updated by the preceding rows.

For one row, define

$$R = T + Q$$

where
T = conservative candidate-test span in the row
Q = conservative successful-rotation span in the row.

The complete active-split span is therefore

$$\text{Sum} = R_1 + R_2 + R_3.$$

For a candidate that reaches the full acceptance test, the arithmetic work is

- 1 subtraction: $\Delta = d_1 - d_2$
- 2 multiplications: $|b| M$ and $|\Delta| \tau$.

This is a conservative count: many candidates are accepted or rejected by comparisons among the factors
before either product is formed, so they incur less arithmetic than this full-test cost.

The two multiplications are independent once Δ is available, so the dependency span is 2.

Work of a successful rotation

For an accepted split, the fill-ins −s a and s g have already been certified

negligible,
so their matrix entries are written directly as zero.
Denote k as the subdiagonal cut index (column index) of the pivot b:

- $A[k-1,k] = A[k,k-1] = c\ a$
- $A[k+1,k+2] = A[k+2,k+1] = c\ g$

are still computed when they exist.

The resulting application work is

- isolated 2 x 2 block: 24 operations
- endpoint split: 25 operations
- interior split: 26 operations.

Construction of c and s is counted separately:

- 7 operations when Δ is reused from the acceptance test
- 8 operations when Δ must be formed during rotation construction.

Estimation of span Q

Algebraically, a successful one-rotation deflation is the local similarity transformation shown earlier.
In the implementation, the fill-ins s a and s g are written as zero.

$$T = \begin{bmatrix} \cdot & \bullet & 0 & 0 & 0 & 0 & 0 & 0 \\ \bullet & \cdot & a & 0 & 0 & 0 & 0 & 0 \\ 0 & a & d_1 & b & 0 & 0 & 0 & 0 \\ 0 & 0 & b & d_2 & g & 0 & 0 & 0 \\ 0 & 0 & 0 & g & \cdot & \bullet & 0 & 0 \\ 0 & 0 & 0 & 0 & \bullet & \cdot & \bullet & 0 \\ 0 & 0 & 0 & 0 & 0 & \bullet & \cdot & \bullet \\ 0 & 0 & 0 & 0 & 0 & 0 & \bullet & \cdot \end{bmatrix} \qquad G = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c & -s & 0 & 0 & 0 & 0 \\ 0 & 0 & s & c & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T' = G^{\intercal} T G = \begin{bmatrix} \cdot & \bullet & 0 & 0 & 0 & 0 & 0 & 0 \\ \bullet & \cdot & ca & -sa & 0 & 0 & 0 & 0 \\ 0 & ca & d_1' & 0 & sg & 0 & 0 & 0 \\ 0 & -sa & 0 & d_2' & cg & 0 & 0 & 0 \\ 0 & 0 & sg & cg & \cdot & \bullet & 0 & 0 \\ 0 & 0 & 0 & 0 & \bullet & \cdot & \bullet & 0 \\ 0 & 0 & 0 & 0 & 0 & \bullet & \cdot & \bullet \\ 0 & 0 & 0 & 0 & 0 & 0 & \bullet & \cdot \end{bmatrix}$$

The span Q has three successive parts:

① Construct the Jacobi rotation G

For the conservative case, Δ has not already been computed by the acceptance test.
The dependency chain used to obtain c and s is shown below.

Operation	Arithmetic span	Weighted span
$\Delta = d_1 - d_2$	1	1
HalfDelta = 0.5 Δ	1	1
Radius = hypot(HalfDelta, b)	1	6
Denominator = HalfDelta ± Radius	1	1
Norm = hypot(Denominator, b)	1	6
$\text{InverseNorm} = \frac{1}{\text{Norm}}$	1	4
• c = | Denominator | InverseNorm • s = ±b InverseNorm (computed in parallel)	1	1
Total	7	20

Each row depends on the preceding row; c and s in the final row are formed concurrently.
So construction of G has arithmetic span 7 and weighted span 20.

② Apply T G from the right

For every affected row, the two new entries have the form

$$x' = c\,x + s\,y$$

$$y' = -s\,x + c\,y.$$

• the four multiplications can proceed concurrently

• after the multiplications, the two additions or subtractions can proceed concurrently

↓

Thus the right application has arithmetic span 2 and weighted span 2.

③ Apply G^T from the left

The left application has the same two-level multiply-then-add structure, so it also has arithmetic span 2 and weighted span 2.
It must follow the right application because it uses the matrix produced by T G.

Therefore

• construction of G: 7, weighted 20
• right application: 2, weighted 2
• left application: 2, weighted 2

Q = 11, weighted Q = 24.

Successful rotations within one candidate row have non-intersecting neighborhoods.
Their rotation constructions and matrix updates can therefore proceed concurrently.
Thus a candidate row containing one or more successful rotations contributes one Q, not one Q per rotation.

Returning to the three candidate rows,

$$\text{Sum} = R_1 + R_2 + R_3 =$$
$$(T_1 + Q_1) + (T_2 + Q_2) + (T_3 + Q_3).$$

For every evaluated candidate row,
T = 2.

For each candidate row, the acceptance tests determine a fixed set of successful cuts.
If that set is empty, the row contributes no rotation span.
If it is nonempty, all successful rotations in the row proceed concurrently, so the row contributes one weighted rotation span of 24.

Thus each candidate row contributes either
2
or
2 + 24
to the weighted arithmetic span.

Reported quantities

Ⓐ
For sweep positions, weighted arithmetic, and weighted arithmetic span

- absolute reduction = baseline − candidate
- relative reduction = $\dfrac{\text{baseline} - \text{candidate}}{\text{baseline}}$

Ⓑ
For eigenvalue reference and error sources

- Reference eigenvalues are computed by LAPACK DSYEVD (LAPACK [8]).

• Errors are computed as the absolute difference from the LAPACK reference eigenvalue
and reported side by side for the candidate and the standard implementation

Measure	Source
Scaled eigenvalue error vs LAPACK	(DLMF [6]) (Jiang, Cui & Dai [7])
Matrix-scaled eigenvalue error vs LAPACK	(LAPACK [9])

①
Sweep positions

For an active block of size m, one complete QR sweep contains m − 1 sweep positions.
Total sweep positions are the sum of m − 1 over all active blocks and all sweeps.

②
Weighted arithmetic work W

The sum of the operations multiplied by the assigned operation weights.

③
Weighted arithmetic span S

Weighted cost along the longest arithmetic dependency path.

④
Conventional and active deflations
• All counts refer to the candidate method
• Counts are summed over all tested samples at each matrix size

- Conventional deflations are ordinary QR deflations
- Active CAT deflations are one-rotation splits of active blocks
- Terminal 2 x 2 diagonalizations are excluded from the active CAT count

$$\text{Active CAT fraction} = \frac{\text{active CAT deflations}}{\text{active CAT deflations + conventional deflations}}$$

⑤

$$\text{Mean} \quad \frac{\text{smaller child block}}{\text{parent-block}} \quad \text{ratio}$$

For a parent block of size $m > 2$, let n_l and n_r be the child sizes.

$$Q = \frac{\min\left(n_l, n_r\right)}{m}$$

$$\frac{1}{m} \leq Q \leq \frac{1}{2}$$

Direct diagonalization of 2 x 2 blocks is excluded because it is the endpoint of the iteration

Mean of all Q values for each matrix size and family is reported for
- active splits, candidate
- passive splits, candidate
- passive splits, standard

⑥

CAT weighted work and span decomposition

Decomposition of the candidate method into QR iteration, CAT tests, and CAT rotations.

- W is each component's percentage of total weighted arithmetic work.
- S is each component's percentage of total weighted arithmetic span.

⑦

Scaled eigenvalue error vs LAPACK

Mollified eigenvalue error: absolute scaling for $|\lambda| < 1$ and relative scaling for $|\lambda| > 1$.

⑧

Matrix-scaled eigenvalue error vs LAPACK

Eigenvalue error normalized by the largest $|\lambda|$.

General dense symmetric

Sweep positions

n	baseline	candidate	relative reduction	absolute reduction
63	4,301.67	4,155.67	3.39%	146
255	65,113.67	63,012.33	3.23%	2,101.33
511	252,460.33	242,201.67	4.06%	10,258.67
1023	975,629	938,782	3.78%	36,847

Weighted arithmetic work W

n	baseline	candidate	relative reduction	absolute reduction
63	275,229	275,542.33	-0.11%	-313.33
255	4,166,884.33	4,178,575	-0.28%	-11,690.67
511	16,156,702	16,060,837.33	0.59%	95,864.67
1023	62,438,857.33	62,256,039	0.29%	182,818.33

Weighted arithmetic span S

n	baseline	candidate	relative reduction	absolute reduction
63	124,232.67	120,938	2.65%	3,294.67
255	1,837,418.67	1,781,598	3.04%	55,820.67
511	7,096,585.33	6,814,883.33	3.97%	281,702
1023	27,370,791.33	26,350,107.33	3.73%	1,020,684

Conventional and active deflations

- All counts refer to the candidate method
- Counts are summed over all tested samples at each matrix size

n	conventional	active CAT	$\frac{\text{active CAT}}{\text{all deflations}}$
63	158	25	13.66%
255	668	91	11.99%
511	1,359	167	10.94%
1023	2,715	338	11.07%

$$\text{Mean } \frac{\text{smaller child block}}{\text{parent-block}} \text{ ratio}$$

n	active candidate	passive candidate	passive standard
63	6.37%	5.12%	5.29%
255	2.32%	1.76%	1.82%
511	1.2%	1.07%	1.04%
1023	1.35%	0.61%	0.68%

CAT weighted work and span decomposition

n	component	W	S
63	QR iteration	96.49%	99.2%
	CAT tests	3.34%	0.62%
	CAT rotations	0.17%	0.19%
255	QR iteration	96.5%	99.8%
	CAT tests	3.46%	0.16%
	CAT rotations	0.04%	0.04%
511	QR iteration	96.51%	99.9%
	CAT tests	3.47%	0.08%
	CAT rotations	0.02%	0.02%
1023	QR iteration	96.51%	99.95%
	CAT tests	3.48%	0.04%
	CAT rotations	0.01%	0.01%

Scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	7.67e-15	5.55e-15
255	2.10e-14	2.05e-14
511	1.86e-14	2.03e-14
1023	4.11e-14	4.11e-14

Matrix-scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	6.38e-15	4.73e-15
255	1.08e-14	1.57e-14
511	1.65e-14	1.80e-14
1023	2.62e-14	3.28e-14

Dense correlation

Sweep positions

n	baseline	candidate	relative reduction	absolute reduction
63	4,048.67	3,927.33	3%	121.33
255	62,824.33	59,793	4.83%	3,031.33
511	240,026.67	230,220	4.09%	9,806.67
1023	919,835	880,675	4.26%	39,160

Weighted arithmetic work W

n	baseline	candidate	relative reduction	absolute reduction
63	259,054.33	259,445.67	-0.15%	-391.33
255	4,020,390.67	3,957,900	1.55%	62,490.67
511	15,361,009.67	15,228,558.33	0.86%	132,451.33
1023	58,868,077.33	58,289,429	0.98%	578,648.33

Weighted arithmetic span S

n	baseline	candidate	relative reduction	absolute reduction
63	116,888.67	114,186.67	2.31%	2,702
255	1,773,178.67	1,691,079.33	4.63%	82,099.33
511	6,747,724	6,478,636	3.99%	269,088
1023	25,807,486	24,722,187.33	4.21%	1,085,298.67

Conventional and active deflations

- All counts refer to the candidate method
- Counts are summed over all tested samples at each matrix size

n	conventional	active CAT	active CAT / all deflations
63	165	18	9.84%
255	662	96	12.66%
511	1,359	165	10.83%
1023	2,685	371	12.14%

Mean $\frac{\text{smaller child block}}{\text{parent-block}}$ ratio

n	active candidate	passive candidate	passive standard
63	7.26%	5.08%	5.29%
255	3.48%	1.66%	1.86%
511	1.18%	1.09%	1.07%
1023	0.67%	0.6%	0.58%

CAT weighted work and span decomposition

n	component	W	S
63	QR iteration	96.85%	99.25%
	CAT tests	3.02%	0.6%
	CAT rotations	0.13%	0.15%
255	QR iteration	96.68%	99.79%
	CAT tests	3.28%	0.16%
	CAT rotations	0.04%	0.05%
511	QR iteration	96.75%	99.9%
	CAT tests	3.23%	0.08%
	CAT rotations	0.02%	0.02%
1023	QR iteration	96.69%	99.95%
	CAT tests	3.3%	0.04%
	CAT rotations	0.01%	0.01%

Scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	3.22e-15	4.28e-15
255	6.48e-15	9.53e-15
511	1.97e-14	9.53e-15
1023	1.75e-14	2.18e-14

Matrix-scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	2.74e-15	3.90e-15
255	6.46e-15	9.30e-15
511	1.96e-14	9.31e-15
1023	1.75e-14	2.18e-14

Dense clustered spectrum

Sweep positions

n	baseline	candidate	relative reduction	absolute reduction
63	2,977	2,785.67	6.43%	191.33
255	43,473.33	42,286.67	2.73%	1,186.67
511	169,723.67	165,605.33	2.43%	4,118.33
1023	663,385.67	649,763.33	2.05%	13,622.33

Weighted arithmetic work W

n	baseline	candidate	relative reduction	absolute reduction
63	190,634.67	185,821	2.53%	4,813.67
255	2,782,137.33	2,808,710.33	-0.96%	-26,573
511	10,861,880	10,994,185	-1.22%	-132,305
1023	42,455,867.67	43,144,592.67	-1.62%	-688,725

Weighted arithmetic span S

n	baseline	candidate	relative reduction	absolute reduction
63	84,374.67	79,756	5.47%	4,618.67
255	1,182,419.33	1,157,694	2.09%	24,725.33
511	4,606,706.67	4,526,251.33	1.75%	80,455.33
1023	18,059,965.33	17,794,909.33	1.47%	265,056

Conventional and active deflations

- All counts refer to the candidate method
- Counts are summed over all tested samples at each matrix size

n	conventional	active CAT	active CAT / all deflations
63	139	35	20.11%
255	636	109	14.63%
511	1,299	205	13.63%
1023	2,565	443	14.73%

Mean $\frac{\text{smaller child block}}{\text{parent-block}}$ ratio

n	active candidate	passive candidate	passive standard
63	13.54%	8.87%	9.74%
255	7.75%	4.23%	4.73%
511	5.36%	2.6%	2.99%
1023	3.56%	1.66%	1.86%

CAT weighted work and span decomposition

n	component	W	S
63	QR iteration	95.93%	99%
	CAT tests	3.65%	0.67%
	CAT rotations	0.41%	0.32%
255	QR iteration	96.35%	99.77%
	CAT tests	3.58%	0.17%
	CAT rotations	0.07%	0.05%
511	QR iteration	96.4%	99.89%
	CAT tests	3.57%	0.09%
	CAT rotations	0.03%	0.03%
1023	QR iteration	96.38%	99.94%
	CAT tests	3.6%	0.04%
	CAT rotations	0.02%	0.01%

Scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	4.00e-15	3.11e-15
255	8.29e-15	1.01e-14
511	9.18e-15	7.55e-15
1023	2.83e-14	2.83e-14

Matrix-scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	3.99e-15	3.11e-15
255	8.28e-15	1.01e-14
511	9.17e-15	7.54e-15
1023	2.22e-14	2.35e-14

Dense graded spectrum

Sweep positions

n	baseline	candidate	relative reduction	absolute reduction
63	2,668	2,474.67	7.25%	193.33
255	42,181.67	39,417.33	6.55%	2,764.33
511	166,755.67	156,003.33	6.45%	10,752.33
1023	657,693.33	616,742	6.23%	40,951.33

Weighted arithmetic work W

n	baseline	candidate	relative reduction	absolute reduction
63	170,778.33	165,556	3.06%	5,222.33
255	2,699,594.33	2,616,913	3.06%	82,681.33
511	10,672,258.33	10,329,135.33	3.22%	343,123
1023	42,092,139	40,738,760	3.22%	1,353,379

Weighted arithmetic span S

n	baseline	candidate	relative reduction	absolute reduction
63	76,930	71,726	6.76%	5,204
255	1,189,952.67	1,112,946	6.47%	77,006.67
511	4,686,784.67	4,381,958	6.5%	304,826.67
1023	18,450,349.33	17,302,788	6.22%	1,147,561.33

Conventional and active deflations

- All counts refer to the candidate method
- Counts are summed over all tested samples at each matrix size

n	conventional	active CAT	active CAT / all deflations
63	145	30	17.14%
255	616	122	16.53%
511	1,249	242	16.23%
1023	2,460	528	17.67%

Mean $\frac{\text{smaller child block}}{\text{parent-block}}$ ratio

n	active candidate	passive candidate	passive standard
63	11.79%	4.85%	5.29%
255	6.3%	2.32%	1.83%
511	5.32%	1.55%	1.04%
1023	3.45%	1.14%	0.59%

CAT weighted work and span decomposition

n	component	W	S
63	QR iteration	95.69%	99.09%
	CAT tests	3.91%	0.56%
	CAT rotations	0.41%	0.35%
255	QR iteration	96.41%	99.77%
	CAT tests	3.5%	0.15%
	CAT rotations	0.09%	0.08%
511	QR iteration	96.66%	99.89%
	CAT tests	3.29%	0.07%
	CAT rotations	0.04%	0.04%
1023	QR iteration	96.89%	99.94%
	CAT tests	3.08%	0.04%
	CAT rotations	0.02%	0.02%

Scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	5.63e-14	5.63e-14
255	5.80e-14	5.78e-14
511	4.82e-14	4.83e-14
1023	7.06e-14	7.05e-14

Matrix-scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	1.36e-15	1.93e-15
255	7.84e-15	7.28e-15
511	1.94e-14	9.78e-15
1023	3.49e-14	1.52e-14

General symmetric tridiagonal

Sweep positions

n	baseline	candidate	relative reduction	absolute reduction
63	4,411.67	4,276	3.08%	135.67
255	72,141.67	68,094.33	5.61%	4,047.33
511	282,923.67	273,051.67	3.49%	9,872
1023	1,045,347.67	1,017,597.33	2.65%	27,750.33

Weighted arithmetic work W

n	baseline	candidate	relative reduction	absolute reduction
63	282,295	283,102.33	-0.29%	-807.33
255	4,616,724.33	4,507,037.33	2.38%	109,687
511	18,106,488.67	18,077,006.67	0.16%	29,482
1023	66,900,839.33	67,340,328	-0.66%	-439,488.67

Weighted arithmetic span S

n	baseline	candidate	relative reduction	absolute reduction
63	127,412.67	124,408	2.36%	3,004.67
255	2,031,279.33	1,918,522.67	5.55%	112,756.67
511	7,933,989.33	7,650,327.33	3.58%	283,662
1023	28,949,187.33	28,396,581.33	1.91%	552,606

Conventional and active deflations

- All counts refer to the candidate method
- Counts are summed over all tested samples at each matrix size

n	conventional	active CAT	active CAT / all deflations
63	165	17	9.34%
255	669	80	10.68%
511	1,330	176	11.69%
1023	2,690	329	10.9%

Mean $\frac{\text{smaller child block}}{\text{parent-block}}$ ratio

n	active candidate	passive candidate	passive standard
63	4.77%	5.38%	5.34%
255	5.53%	2.92%	2.86%
511	5.06%	2.17%	2.36%
1023	3.41%	2.22%	2.16%

CAT weighted work and span decomposition

n	component	W	S
63	QR iteration	96.63%	99.26%
	CAT tests	3.25%	0.62%
	CAT rotations	0.12%	0.13%
255	QR iteration	96.69%	99.81%
	CAT tests	3.28%	0.16%
	CAT rotations	0.03%	0.03%
511	QR iteration	96.67%	99.91%
	CAT tests	3.31%	0.08%
	CAT rotations	0.02%	0.02%
1023	QR iteration	96.71%	99.95%
	CAT tests	3.28%	0.04%
	CAT rotations	0.01%	0.01%

Scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	8.44e-15	6.67e-15
255	1.72e-14	1.45e-14
511	2.39e-14	2.53e-14
1023	3.58e-14	3.32e-14

Matrix-scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	6.53e-15	5.35e-15
255	1.26e-14	9.88e-15
511	1.31e-14	2.51e-14
1023	2.35e-14	2.31e-14

Weak-link symmetric tridiagonal

Sweep positions

n	baseline	candidate	relative reduction	absolute reduction
63	4,065.67	3,668	9.78%	397.67
255	54,735	52,099.33	4.82%	2,635.67
511	153,513.33	121,477	20.87%	32,036.33
1023	436,661	345,006.67	20.99%	91,654.33

Weighted arithmetic work W

n	baseline	candidate	relative reduction	absolute reduction
63	260,133.67	244,366.33	6.06%	15,767.33
255	3,502,802.33	3,468,127.33	0.99%	34,675
511	9,824,424	8,085,978.67	17.7%	1,738,445.33
1023	27,945,338	22,960,585.67	17.84%	4,984,752.33

Weighted arithmetic span S

n	baseline	candidate	relative reduction	absolute reduction
63	117,494.67	106,834.67	9.07%	10,660
255	1,513,199.33	1,450,287.33	4.16%	62,912
511	3,929,238	2,845,680	27.58%	1,083,558
1023	9,232,508.67	7,059,254	23.54%	2,173,254.67

Conventional and active deflations

- All counts refer to the candidate method
- Counts are summed over all tested samples at each matrix size

n	conventional	active CAT	$\frac{\text{active CAT}}{\text{all deflations}}$
63	161	22	12.02%
255	636	113	15.09%
511	1,267	234	15.59%
1023	2,576	439	14.56%

$$\text{Mean } \frac{\text{smaller child block}}{\text{parent-block}} \text{ ratio}$$

n	active candidate	passive candidate	passive standard
63	4.62%	5.38%	5.29%
255	3.94%	3.04%	3.6%
511	5.05%	3.27%	3.39%
1023	4.73%	3.31%	3.19%

CAT weighted work and span decomposition

n	component	W	S
63	QR iteration	96.04%	99.18%
	CAT tests	3.79%	0.63%
	CAT rotations	0.17%	0.19%
255	QR iteration	96.14%	99.77%
	CAT tests	3.8%	0.17%
	CAT rotations	0.06%	0.06%
511	QR iteration	96.14%	99.85%
	CAT tests	3.8%	0.11%
	CAT rotations	0.05%	0.04%
1023	QR iteration	96.16%	99.92%
	CAT tests	3.8%	0.06%
	CAT rotations	0.03%	0.02%

Scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	7.39e-15	6.63e-15
255	1.62e-14	1.59e-14
511	2.18e-14	2.05e-14
1023	3.08e-14	3.06e-14

Matrix-scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	5.44e-15	5.57e-15
255	8.97e-15	9.48e-15
511	1.14e-14	1.34e-14
1023	1.53e-14	1.99e-14

Wilkinson tridiagonal

Sweep positions

n	baseline	candidate	relative reduction	absolute reduction
63	3,939	3,670	6.83%	269
255	44,224	40,330	8.81%	3,894
511	154,290	148,505	3.75%	5,785
1023	580,006	568,819	1.93%	11,187

Weighted arithmetic work W

n	baseline	candidate	relative reduction	absolute reduction
63	252,037	243,301	3.47%	8,736
255	2,830,207	2,679,815	5.31%	150,392
511	9,874,490	9,852,404	0.22%	22,086
1023	37,120,286	37,710,968	-1.59%	-590,682

Weighted arithmetic span S

n	baseline	candidate	relative reduction	absolute reduction
63	113,798	107,082	5.9%	6,716
255	1,248,588	1,143,906	8.38%	104,682
511	4,337,966	4,181,618	3.6%	156,348
1023	16,273,794	15,969,696	1.87%	304,098

Conventional and active deflations

- All counts refer to the candidate method
- Counts are summed over all tested samples at each matrix size

n	conventional	active CAT	$\frac{\text{active CAT}}{\text{all deflations}}$
63	50	9	15.25%
255	138	115	45.45%
511	416	93	18.27%
1023	920	100	9.8%

$$\text{Mean } \frac{\text{smaller child block}}{\text{parent-block}} \text{ ratio}$$

n	active candidate	passive candidate	passive standard
63	5.47%	5.62%	5.57%
255	0.78%	2.72%	1.84%
511	0.56%	1.16%	1.08%
1023	0.37%	0.61%	0.59%

CAT weighted work and span decomposition

n	component	W	S
63	QR iteration	96.51%	99.12%
	CAT tests	3.25%	0.66%
	CAT rotations	0.24%	0.22%
255	QR iteration	96.33%	99.55%
	CAT tests	3.46%	0.2%
	CAT rotations	0.21%	0.24%
511	QR iteration	96.47%	99.85%
	CAT tests	3.48%	0.09%
	CAT rotations	0.05%	0.05%
1023	QR iteration	96.54%	99.94%
	CAT tests	3.45%	0.04%
	CAT rotations	0.01%	0.02%

Scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	6.00e-14	5.97e-14
255	4.19e-14	3.64e-14
511	3.76e-14	4.60e-14
1023	1.73e-13	2.32e-13

Matrix-scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	1.89e-14	1.88e-14
255	6.45e-15	5.23e-15
511	2.40e-14	1.62e-14
1023	5.59e-14	2.21e-14

Unequal variance-covariance

Sweep positions

n	baseline	candidate	relative reduction	absolute reduction
63	3,508.67	3,326.33	5.2%	182.33
255	55,078	52,323.33	5%	2,754.67
511	213,323.67	203,433.33	4.64%	9,890.33
1023	824,736	787,883.33	4.47%	36,852.67

Weighted arithmetic work W

n	baseline	candidate	relative reduction	absolute reduction
63	224,546.67	220,159	1.95%	4,387.67
255	3,524,831.67	3,443,678.33	2.3%	81,153.33
511	13,652,352.67	13,341,608.67	2.28%	310,744
1023	52,782,442	51,555,290.33	2.32%	1,227,151.67

Weighted arithmetic span S

n	baseline	candidate	relative reduction	absolute reduction
63	101,228.67	96,657.33	4.52%	4,571.33
255	1,552,508.67	1,477,498.67	4.83%	75,010
511	5,994,090	5,721,591.33	4.55%	272,498.67
1023	23,134,542.67	22,110,908	4.42%	1,023,634.67

Conventional and active deflations

- All counts refer to the candidate method
- Counts are summed over all tested samples at each matrix size

n	conventional	active CAT	active CAT / all deflations
63	155	27	14.84%
255	650	103	13.68%
511	1,296	224	14.74%
1023	2,628	414	13.61%

Mean $\frac{\text{smaller child block}}{\text{parent-block}}$ ratio

n	active candidate	passive candidate	passive standard
63	5.63%	5.76%	5.34%
255	3.67%	2.28%	2.08%
511	2.1%	1.25%	1.25%
1023	1.62%	0.8%	0.7%

CAT weighted work and span decomposition

n	component	W	S
63	QR iteration	96.68%	99.15%
	CAT tests	3.08%	0.6%
	CAT rotations	0.23%	0.25%
255	QR iteration	97.24%	99.8%
	CAT tests	2.71%	0.15%
	CAT rotations	0.05%	0.06%
511	QR iteration	97.59%	99.89%
	CAT tests	2.38%	0.08%
	CAT rotations	0.03%	0.03%
1023	QR iteration	97.81%	99.95%
	CAT tests	2.18%	0.04%
	CAT rotations	0.01%	0.01%

Scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	1.25e-14	1.25e-14
255	1.85e-14	2.15e-14
511	1.85e-14	1.67e-14
1023	3.23e-14	2.33e-14

Matrix-scaled eigenvalue error vs LAPACK

n	baseline	candidate
63	4.43e-15	2.63e-15
255	1.08e-14	1.34e-14
511	1.47e-14	9.20e-15
1023	3.05e-14	1.84e-14

Conclusions

Chase And Trim is a method for creating valid matrix splits earlier than conventional QR deflation, which

- reduces the amount of subsequent QR iteration on the larger child block
- allows the smaller child block to be processed independently, without adding to the dependency path

①

Structural effect

⊙ The algorithm did reduce the number of QR sweep positions, defined as the sum of m − 1 over all QR sweeps on all active blocks of size m,

in 32 of 32 tested matrix-family/matrix-size combinations.
The relative reduction ranged
• from 1.93% for Wilkinson tridiagonal, n = 1023,
• to 20.99% for Weak-link symmetric tridiagonal, n = 1023.

⊙ Active one-rotation splits as fraction of all deflations
• 9.34%, General symmetric tridiagonal, n = 63 (minimum)
• 45.45%, Wilkinson tridiagonal, n = 255 (maximum)

⊙ As a secondary observation, active CAT splits tended to be somewhat more interior
than conventional deflations in the standard method.

The mean $\frac{\text{smaller child size}}{\text{parent size}}$ ratio was larger in 26 of 32 tested matrix-family/matrix-size combinations.

②
Main objective: weighted arithmetic span
This is the most consequential metric for a parallel implementation

⊙ Weighted arithmetic span was reduced in every tested matrix-family/matrix-size combination.

• Dense clustered spectrum, n = 1023: about 1.47% (smallest reduction).
• Weak-link symmetric tridiagonal, n = 511: about 27.58% (largest reduction).

③
Secondary objective: weighted arithmetic work

⊙ CAT reduced weighted arithmetic work in 23 of 32 tested matrix-family/matrix-size combinations.
In the remaining 9 combinations, testing overhead met or exceeded the arithmetic eliminated by earlier splitting.

④

Candidate algorithm overhead

CAT overhead was small and explicitly accounted for.

The additional computation consists of:

- testing candidate cuts
- constructing and applying successful one-rotation splits

We report the absolute weighted work W and weighted span S contributed by CAT testing and CAT rotations separately.

The acceptance test was reduced algebraically to the minimum practical sequence of comparisons and arithmetic operations described above, avoiding construction of the Jacobi rotation unless a candidate is accepted.

These quantities separate the cost of finding earlier splits from the QR work and dependency path eliminated by those splits.

Across the tested cases, the largest combined CAT cost was

- weighted work: 4.31%
- weighted span: 1.00%

⑤

Numerical accuracy

A successful CAT rotation creates only two cross-cut fill entries. The acceptance condition guarantees algebraically that both are no larger than the same local tolerance used for deflation, justifying their removal and the resulting matrix split.

Candidate and baseline eigenvalue errors relative to LAPACK remained of the same order across the tested matrix families and sizes.

Maximum scaled eigenvalue error versus LAPACK:
• standard: 1.73e-13
• candidate: 2.32e-13

Maximum matrix-scaled eigenvalue error versus LAPACK:
• standard: 5.59e-14
• candidate: 3.28e-14

Thus the reductions in sweep positions and weighted arithmetic span
were obtained without relaxing the numerical accuracy criterion.

⑥

Local exact annihilation

For a chosen adjacent pair, any orthogonal similarity restricted to those two coordinates
that annihilates the central off-diagonal entry is, up to equivalent sign and ordering choices,
a Jacobi diagonalizing rotation.

The smaller-angle branch minimizes | s | and therefore minimizes
the two cross-cut fill magnitudes | s a | and | s g |.

If the smaller-angle rotation fails the split criterion,
the larger-angle branch cannot succeed.

Thus, within local two-coordinate orthogonal similarities,
the remaining choices are to discard the Jacobi fill when it is within tolerance
or to perform additional work to remove or control it.

Limitations

• Eigenvalues only.
The present study does not include eigenvector accumulation.

• No wall-clock performance claim.
The comparison is algorithmic and reports sweep positions,
weighted arithmetic work W, and weighted arithmetic span S.
Actual runtime depends on implementation, hardware, scheduling,
memory behavior, and available parallel resources.

• The implementation is deliberately transparent rather than
production-optimized.
Baseline and candidate use the same QR implementation;
the candidate differs by adding CAT.
Thus the experiment is designed to isolate the incremental effect of CAT,
not to compare this QR implementation with an optimized production
eigensolver.

• Weighted span is an algorithmic dependency-path model.
It does not include communication, synchronization, thread scheduling,
cache effects, or other machine-dependent costs.

• The conclusions are limited to the tested matrix families,
matrix sizes, and tolerance.
The observed dependence on matrix structure should not be assumed
to extend unchanged to arbitrary symmetric matrices.

Future directions

The local exact-zero question appears largely closed:
within a two-coordinate orthogonal similarity, exact annihilation returns to Jacobi,
and the smaller-angle branch already minimizes the resulting fill.

One remaining local possibility is additional cleanup of the fill created by the first
rotation.
Our diagnostic tests found that this was rarely useful overall,
although graded-spectrum matrices behaved differently enough to merit separate

study.

A broader direction is suggested by the contrast with our separate Householder–Jacobi study.

Small local diagonal separation relative to coupling can favor Jacobi-oriented preprocessing,
whereas large local $|d_1 - d_2|$ directly favors the CAT acceptance condition.

This suggests a possible triage principle based on diagonal spread and coupling:
large diagonal spread may favor QR-based methods,
while smaller diagonal spread may define a Jacobi-favorable regime.

A natural next step is therefore to test whether inexpensive structural measures of diagonal spread and coupling can predict which eigensolver family is preferable,
using common work and span accounting across controlled matrix families.

References

① D. S. Watkins
Fundamentals of Matrix Computations, 3rd edition, Section 5.6: Francis's Algorithm
John Wiley & Sons, 2010.

② LAPACK
DSTEQR: Symmetric Tridiagonal Eigensolver Using Implicit QL or QR Iteration

LAPACK 3.12.1 documentation, Netlib.
https://www.netlib.org/lapack/explore-html/d3/d16/group__steqr.html

③ Agner Fog
Instruction Tables: Lists of Instruction Latencies, Throughputs and Micro-operation Breakdowns for Intel, AMD and VIA CPUs
https://www.agner.org/optimize/instruction_tables.pdf

④ LAPACK/ScaLAPACK testing literature
Test matrices for symmetric eigenvalue problems include prescribed spectra with arithmetic, geometric, and clustered eigenvalue distributions, combined with orthogonal similarity transformations.
https://www.netlib.org/lapack/lawnspdf/lawn18.pdf

⑤ Philip I. Davies and Nicholas J. Higham
Numerically Stable Generation of Correlation Matrices and Their Factors
BIT Numerical Mathematics, 40(4), 640–651, 2000.
https://eprints.maths.manchester.ac.uk/323/

⑥ NIST Digital Library of Mathematical Functions
Section 3.1(v): Error Measures
Defines the mollified error $\varepsilon_m = |x^* - x|/\max(|x|, 1)$.
https://dlmf.nist.gov/3.1#v

⑦ Bo Jiang, Chunfeng Cui, and Yu-Hong Dai
Unconstrained Optimization Models for Computing Several Extreme Eigenpairs of Real Symmetric Matrices
Pacific Journal of Optimization, 10(1), 55–71, 2014.
Uses $|\hat{\lambda}_i - \lambda_i|/\max(1, |\lambda_i|)$ to report eigenvalue error.
https://lsec.cc.ac.cn/~dyh/publication.html

⑧ LAPACK
DSYEVD: Symmetric Eigensolver
Computes all eigenvalues and, optionally, eigenvectors of a real symmetric matrix.

https://www.netlib.org/lapack/explore-html/d8/d30/group__heevd_ga25b71a69f9921df0a6050aa5883f54f4.html

⑨ LAPACK Users' Guide
Further Details: Error Bounds for the Symmetric Eigenproblem
For a symmetric matrix, standard eigenvalue error bounds scale with $\varepsilon \|A\|_2$, where $\|A\|_2 = \max_i |\lambda_i|$.
https://www.netlib.org/lapack/lug/node90.html

⑩ Christian H. Reinsch
A Stable, Rational QR Algorithm for the Computation of the Eigenvalues of an Hermitian, Tridiagonal Matrix
Mathematics of Computation, 25(115), 591–597, 1971.
For a natural QR iteration on a symmetric tridiagonal matrix, gives
4n additions, 10n multiplications, 2n divisions, and n square roots per iteration.
DOI: 10.1090/S0025-5718-1971-0295555-4

⑪ Richard P. Brent
The Parallel Evaluation of General Arithmetic Expressions
Journal of the ACM, 21(2), 201–206, 1974.
https://doi.org/10.1145/321812.321815

⑫ Gene H. Golub and Charles F. Van Loan
Matrix Computations, 4th edition
Johns Hopkins University Press, 2013.
Standard reference for induced matrix norms, including the spectral norm $\|A\|_2$.

⑬ J. H. Wilkinson
The Algebraic Eigenvalue Problem
Clarendon Press, Oxford, 1965.

⑭ James W. Demmel, Osni A. Marques, Beresford N. Parlett, and Christof Vömel
A Testing Infrastructure for LAPACK's Symmetric Eigensolvers
LAPACK Working Note 182, April 2007.

Defines general glued symmetric tridiagonal matrices as tridiagonal blocks connected by interface glue factors.
https://www.netlib.org/lapack/lawnspdf/lawn182.pdf