Banded Householder representation of linear subspaces

G. Irving

Weta Digital, 9-11 Manuka St., Miramar, Wellington, New Zealand

Abstract

We show how to compactly represent any n-dimensional subspace of $\mathbf{R}^m$ as a banded product of Householder reflections using $n(m-n)$ floating point numbers. This is optimal since these subspaces form a Grassmannian space $\mathrm{Gr}_n(m)$ of dimension $n(m-n)$. The representation is stable and easy to compute: any matrix can be factored into the product of a banded Householder matrix and a square matrix using two to three QR decompositions.

Keywords: banded, linear subspace, orthogonal matrix, Householder reflection

If $m \geq n$, the Householder QR algorithm represents an $m \times n$ orthogonal matrix U as a product of n Householder reflections using a total of $n(m-(n+1)/2)$ floating point numbers [1, chap. 5]. However, in some applications only the range of U is important; any other orthogonal matrix with the same range is equivalent. One example is the hierarchically semiseparable representations of [2], where a tree of orthogonal matrices is used to compress matrices with significant offdiagonal structure. Since the n-dimensional subspaces of $\mathbf{R}^m$ form a Grassmannian manifold $\mathrm{Gr}_n(m)$ of dimension $n(m-n)$, we expect that *some* orthogonal matrix with the correct range can be represented with $n(m-n)$ floats. The following theorem provides such a representation:

Theorem 1. *If $m \geq n$, any $A \in \mathbf{R}^{m\times n}$ can be factored as*

$$A = G \begin{pmatrix} B \\ 0 \end{pmatrix}$$

where $B \in \mathbf{R}^{n\times n}$ is square and G is a product of n Householder reflections with

Email address: irving@naml.us (G. Irving) *banded structure:*

$$
\begin{aligned}
G &= H_1 H_2 \cdots H_n \\
H_i &= I - \frac{2 v_i v_i^T}{v_i^T v_i} \\
v_i &= \left(\underbrace{0, 0, \cdots, 0}_{i-1}, 1, \underbrace{v_{i,i+1}, \cdots, v_{i,i+m-n}}_{m-n}, \underbrace{0, 0, \cdots, 0}_{n-i} \right)^T
\end{aligned}
\tag{1}
$$

Since each v_i has $m - n + 1$ nonzero components, the first of which is 1, G can be stored in $n(m - n)$ floats. If A is full rank, the matrices G and B are unique, although the Householder vectors v_i may not be.

Proof. Observe that (1) is exactly the factored form produced by standard Householder QR except for the trailing zeroes in each v_i, which correspond to the extreme lower triangle $i > j + m - n$. To introduce these zeroes, define $A^\diamondsuit$ as A rotated by 180° (pronounced "flip A"),

$$A^\diamondsuit_{ij} = A_{m-i+1,n-i+1}$$

and perform an LQ decomposition of $A^\diamondsuit$:

$$
\begin{aligned}
A^\diamondsuit &= L \;\, Q \\
A \;\; &= L^\diamondsuit Q^\diamondsuit
\end{aligned}
$$

Since $L_{ij} = 0$ for $i < j$, $L^\diamondsuit_{ij} = 0$ for $i > j + m - n$. The Householder QR algorithm constructs v_i as a linear combination of e_i and the ith column of the matrix (after rotation by the previous Householder reflections), and the first component of each vector can be chosen to be 1 [1, chap. 5]. Therefore, a Householder QR decomposition

$$L^\diamondsuit = G \begin{pmatrix} R \\ 0 \end{pmatrix}$$

will produce G with the correct banded structure. Our final factorization is

$$A = G \begin{pmatrix} R \\ 0 \end{pmatrix} Q^\diamondsuit = G \begin{pmatrix} B \\ 0 \end{pmatrix}$$

The steps are visualized in Figure 1. Uniqueness of G and B follows from the uniqueness of the QR decomposition when A is full rank [1, chap. 5]. Note that B is orthogonal whenever A is orthogonal. □

Since the construction uses only matrix multiply and Householder QR decomposition as primitives, the computation of G is stable and requires $O(mn^2)$ flops. Normally the scalar factors $\beta_i = 2/(v_i^T v_i)$ will be precomputed and stored for an additional n floats of storage. Once β_i is available, a single Householder

$$\underset{A}{\begin{pmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{pmatrix}} \Rightarrow \underset{L = A^{\Diamond}Q^T}{\begin{pmatrix} \times & 0 & 0 & 0 \\ \times & \times & 0 & 0 \\ \times & \times & \times & 0 \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{pmatrix}} \Rightarrow \underset{L^{\Diamond}}{\begin{pmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times \end{pmatrix}} \Rightarrow \underset{\begin{pmatrix} v_1 & \cdots & v_n \end{pmatrix}}{\begin{pmatrix} 1 & 0 & 0 & 0 \\ \times & 1 & 0 & 0 \\ \times & \times & 1 & 0 \\ \times & \times & \times & 1 \\ \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times \end{pmatrix}}$$

Figure 1: To compute a banded Householder factorization of a matrix A, we perform an LQ factorization of A rotated by 180° (denoted $A^{\Diamond}$) to zero the extreme lower triangle of A, then perform a QR decomposition to zero the upper triangle and construct the banded Householder vectors v_i.

reflection requires $4(m-n)+2$ flops and matrix vector products Gx or G^Ty can be computed in $4n(m-n)+2n$ flops. If blocking is desired, we can represent products of b Householder reflections where b is the block size using the $I-VTV^T$ representation [3], which involves a relatively small increase in storage and flops if $b \ll m-n$.

Unfortunately, the banded Householder representation in Theorem 1 is inefficient for large subspaces of $\mathbf{R}^m$, since Gx involves a large number of small level 1 BLAS operations if $m-n$ is small. To remedy this problem, we can represent a large n-dimensional subspace in terms of its small $(m-n)$-dimensional orthogonal complement as follows:

Theorem 2. *If $m \geq n$, any $A \in \mathbf{R}^{m\times n}$ can be factored as*

$$A = G\begin{pmatrix} 0 \\ B \end{pmatrix}$$

where $B \in \mathbf{R}^{n\times n}$ is square and G is a banded product of $m-n$ Householder reflections with $n+1$ nonzero components per vector, the first of which is 1 (equivalent to (1) with $m-n$ and n swapped). In particular, G can also be stored in $n(m-n)$ floats.

Proof. Perform a QR decomposition of A to get

$$A = \begin{pmatrix} U_1 & U_2 \end{pmatrix}\begin{pmatrix} R \\ 0 \end{pmatrix} = U_1R$$

Here the column span of $U_1 \in \mathbf{R}^{m\times n}$ contains the range of A, and the span of $U_2 \in \mathbf{R}^{m\times(m-n)}$ is contained inside the nullspace of A^T. Banded Householder factorization of U_2 gives

$$U_2 = G\begin{pmatrix} Q \\ 0 \end{pmatrix} = \begin{pmatrix} G_1 & G_2 \end{pmatrix}\begin{pmatrix} Q \\ 0 \end{pmatrix} = G_1Q$$

Figure 2: Frames from an animation of a digital character using blend shapes stored in a hierarchically semiseparable representation (HSS). Using banded Householder form for the rotations in the HSS tree reduces the storage costs by 45.7% over dense storage, or 29.5% of Householder storage.

whence $G_1 = U_2 Q^T$ and

$$G^T A = \begin{pmatrix} G_1^T \\ G_2^T \end{pmatrix} A = \begin{pmatrix} Q U_2^T A \\ G_2^T A \end{pmatrix} = \begin{pmatrix} 0 \\ G_2^T A \end{pmatrix}$$

since $U_2^T A = U_2^T U_1 R = 0$. Our final decomposition is

$$A = G \begin{pmatrix} 0 \\ G_2^T A \end{pmatrix} = G \begin{pmatrix} 0 \\ B \end{pmatrix}$$

□

Using Theorem 1 for $m - n \geq n$ and Theorem 2 for $m - n < n$, the resulting G consists of at most $m/2$ Householder vectors each with at least $m/2 + 1$ nonzero components. In particular, the blocked $I - VTV^T$ form is efficient whenever $b \ll m, n$, regardless of the value of $m - n$.

1. Application

Our motivating application for the banded Householder decomposition is the compression of blend shape matrices for digital characters. We start with a large, mostly dense matrix where each column represents a pose of the digital character mesh. An example face with 42391 vertices and 730 blend shapes is shown in Figure 2. The original matrix consumes 348 MB of storage in single precision. To reduce this, we compute a lossy hierarchically semiseparable (HSS) representation for the matrix [2], which represents a matrix as a tree of rotations and dense blocks. If the rotations are stored in dense form, the HSS representation requires 46.8 MB of storage. Using Householder form reduces the storage cost to 36.0 MB (77.7% of dense), and banded Householder form reduces the cost further to 25.4 MB (54.3% of dense). On an 8 core Intel Xeon 2.8 GHz machine, the cost to multiply the HSS representation with a vector is 11.2

ms using dense storage with optimized BLAS and 10.7 ms using banded Householder storage with handwritten, unvectorized C. Since the required memory traffic in the banded Householder case is roughly half that of the dense case, we expect this comparison would improve significantly if the banded Householder code were appropriately vectorized.

2. Acknowledgements

I am grateful to Tamar Shinar, J.P. Lewis, and Jaewoo Seo for helpful discussions and comments on the paper.

References

[1] G. Golub, C. Van Loan, Matrix computations, Johns Hopkins Univ. Pr., 1996.

[2] S. Chandrasekaran, M. Gu, T. Pals, Fast and stable algorithms for hierarchically semi-separable representations, submitted for publication (2004).

[3] R. Schreiber, C. Van Loan, A storage-efficient WY representation for products of Householder transformations, SIAM J. Sci. Stat. Comput. 10 (1989) 53–57.