Continuous analogues of QR, SVD, and LU

Nick Trefethen, Oxford

with thanks to Ricardo Pachón, Rodrigo Platte & Toby Driscoll

Fix an interval [a,b], and let A be an $\infty \times n$ quasimatrix on [a,b].

This means A has n "columns", each a function on [a,b].

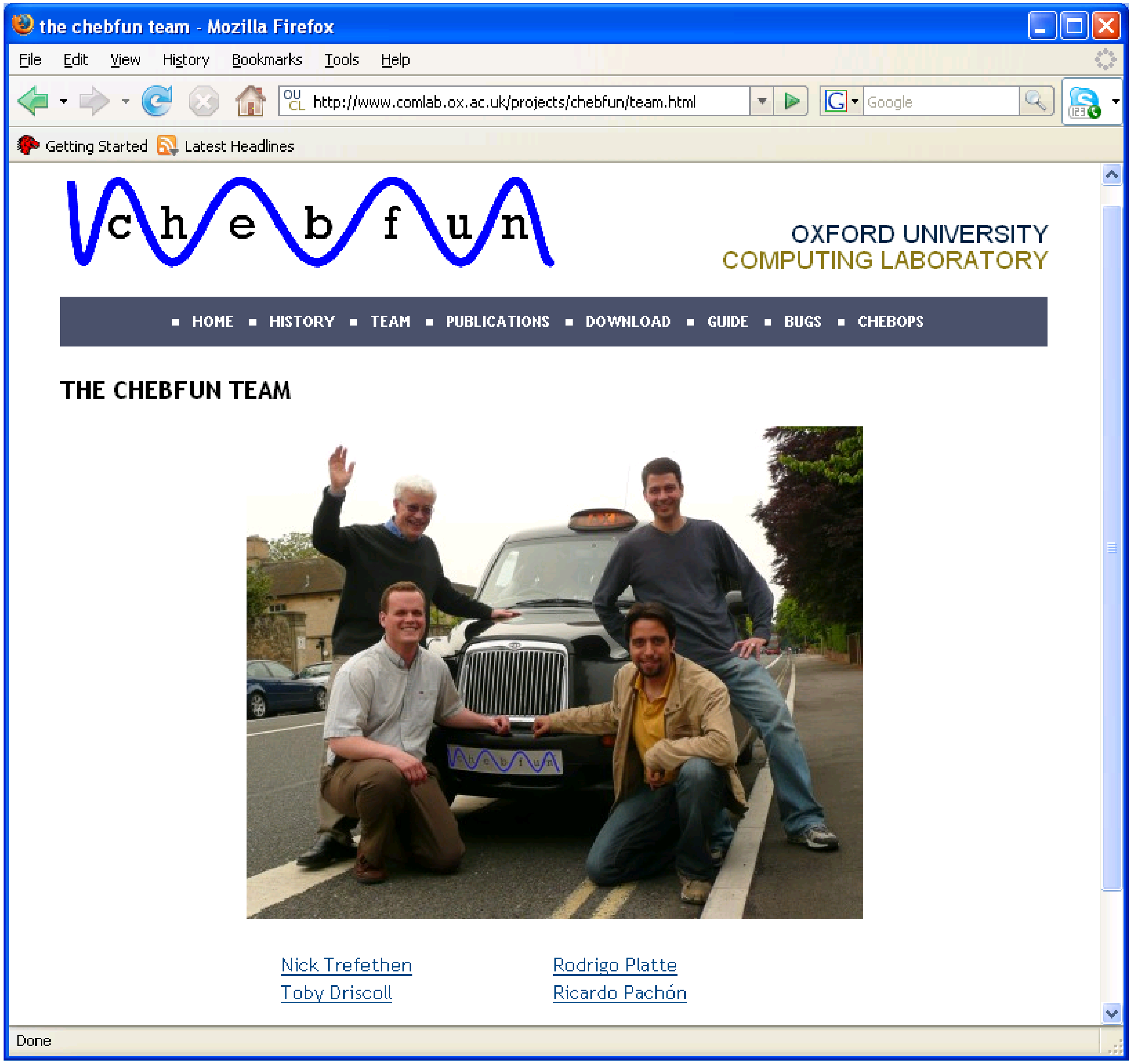
What can we say about familiar operations applied to A?

BACKSLASH	CHOL	COND	EIG
EIGS	GMRES	GSVD	HESS
LU	NORM	NULL	ORTH
PINV	QR	QZ	SCHUR
SVD	SVDS	... ?	

"Core MATLAB contains hundreds of functions. We have found that this collection has an extraordinary power to focus the imagination. We simply asked ourselves, for one MATLAB operation after another, what is the 'right' analogue of this operation in the continuous case? The question comes in two parts. What should each operation mean? And how should one compute it?"

- Battles & T, *SISC* 2004.

In the chebfun system, released yesterday, we can compute such things.



the chebfun team - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://www.comlab.ox.ac.uk/projects/chebfun/team.html

Getting Started Latest Headlines

chebfun

OXFORD UNIVERSITY
COMPUTING LABORATORY

▪ HOME ▪ HISTORY ▪ TEAM ▪ PUBLICATIONS ▪ DOWNLOAD ▪ GUIDE ▪ BUGS ▪ CHEBOPS

THE CHEBFUN TEAM

Nick Trefethen
Toby Driscoll
Rodrigo Platte
Ricardo Pachón

Done

Demo 1: chebfuns

QUASIMATRICES IN PRINT

de Boor 1991, *LAA* – "column map"

T. & Bau 1997, *Numerical Linear Algebra* – "matrix with continuous columns"

Stewart 1998, *Afternotes Goes to Graduate School* – "quasimatrix"

Battles & T. 2004, *SISC*

Battles thesis 2006, *Numerical Linear Algebra for Continuous Functions*

In this talk I'll focus on:

(1) QR (which gives us BACKSLASH)

(2) SVD (which gives us COND, NORM, NULL, ORTH, PINV, RANK)

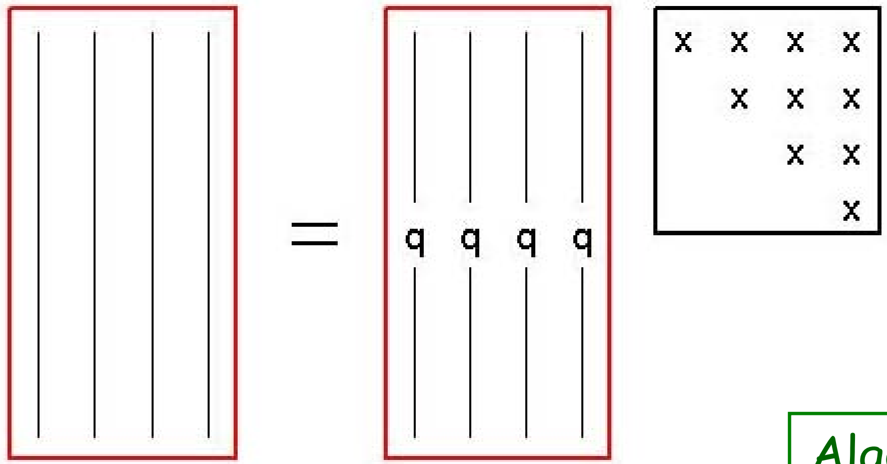
(3) LU

...followed by an Epilogue.

QR factorization ("reduced" or "skinny")

$$\begin{bmatrix} x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \end{bmatrix} = \begin{bmatrix} q & q & q & q \\ q & q & q & q \\ q & q & q & q \\ q & q & q & q \\ q & q & q & q \\ q & q & q & q \end{bmatrix} \begin{bmatrix} x & x & x & x \\ & x & x & x \\ & & x & x \\ & & & x \end{bmatrix}$$

A = QR

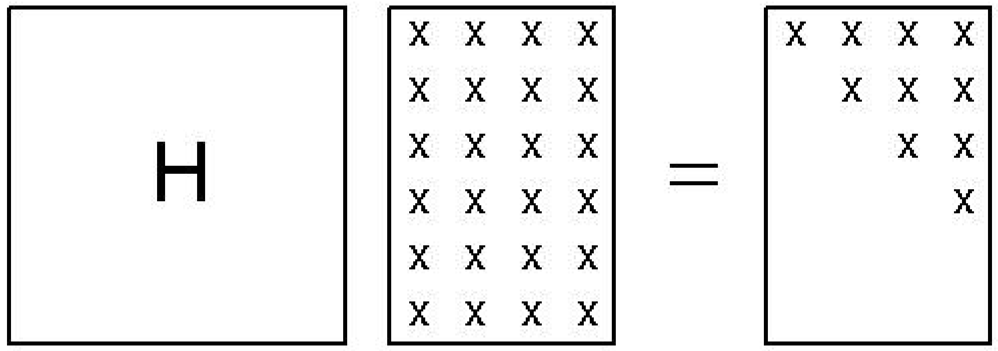


A = QR

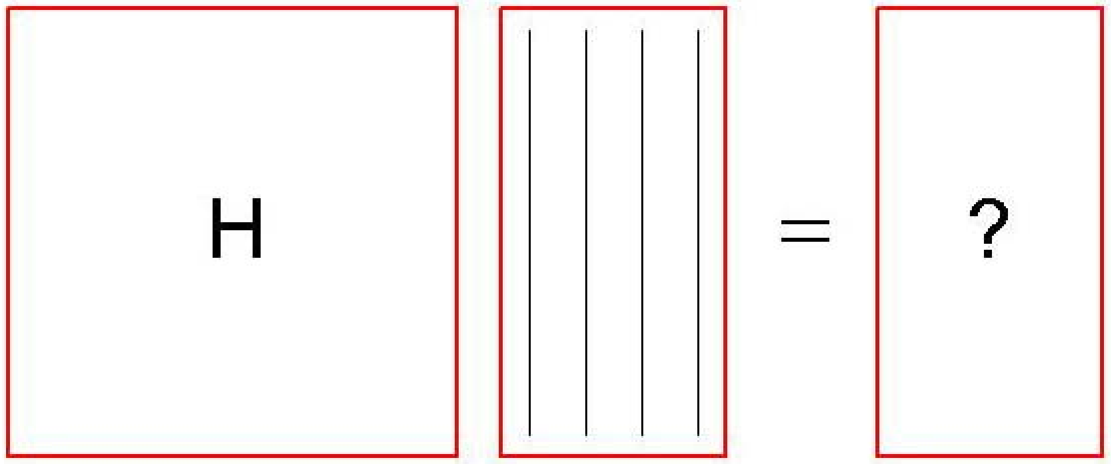
Algorithm: modified Gram-Schmidt

Demo 2: QR

But this is a Householder Symposium!
Can we use Householder reflections to compute quasimatrix QR?

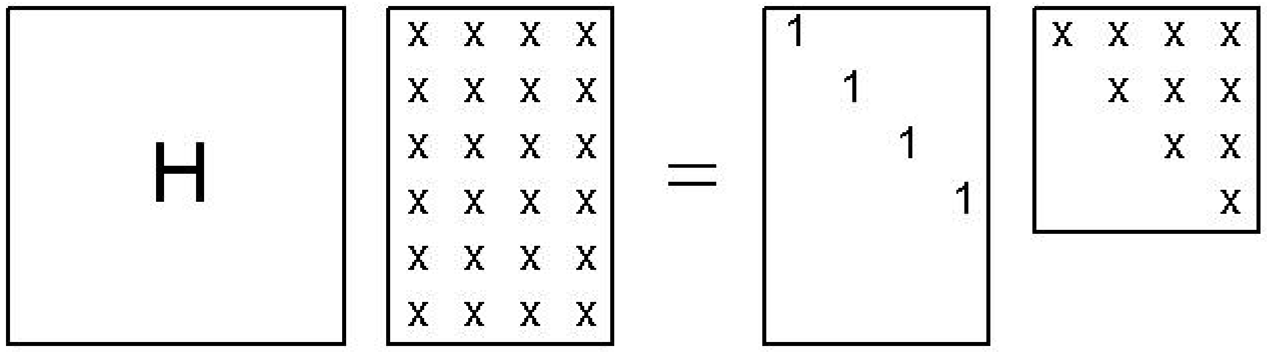


$HA = R$



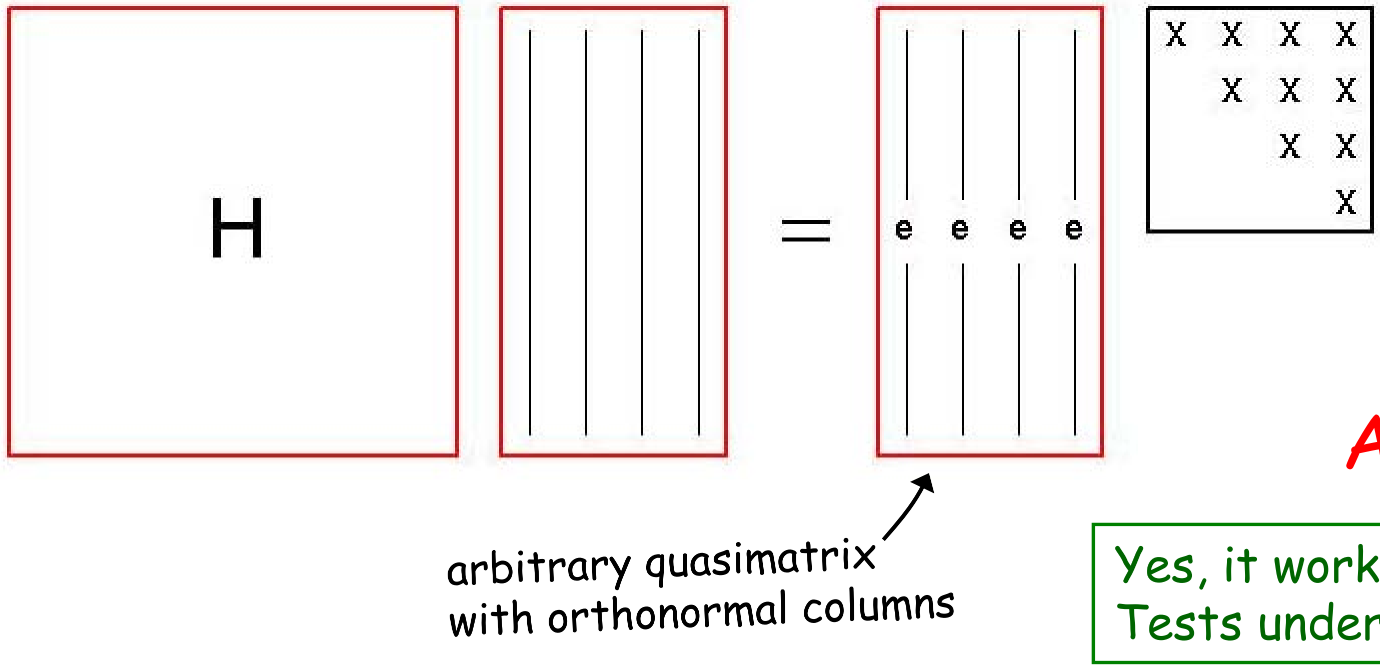
$HA = ?$

To link the continuous and the discrete, we can insert another matrix:



$HA = ER$

$A = (H^*E)R$

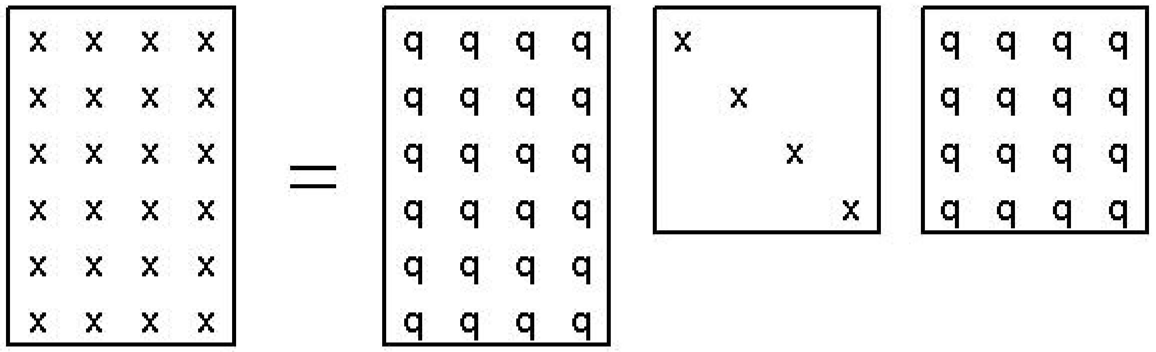


$HA = ER$

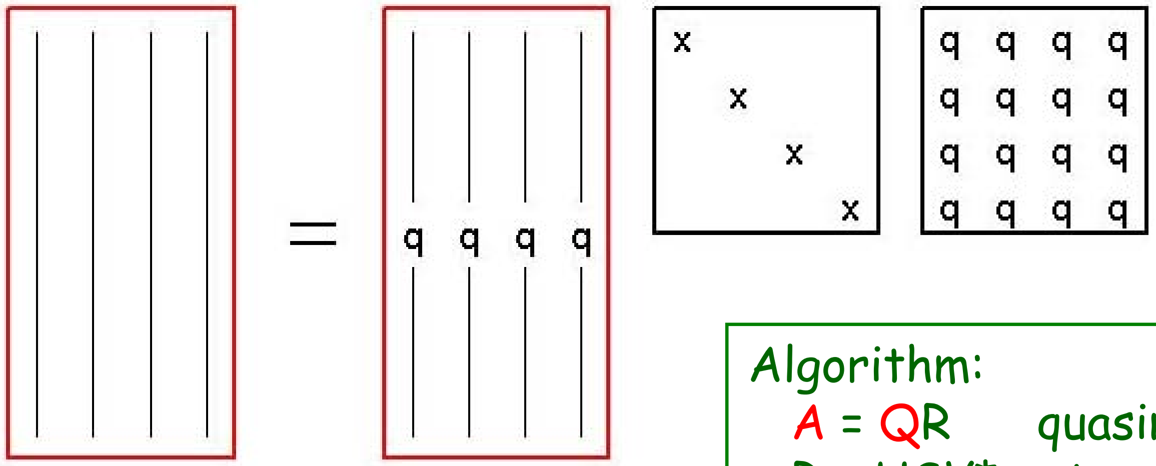
$A = (H^*E)R$

Yes, it works.
Tests underway.

SVD ("reduced" or "skinny")



$A = USV^*$



$A = USV^*$

Algorithm:

$A = QR$ quasimatrix QR factorization

$R = USV^*$ standard SVD

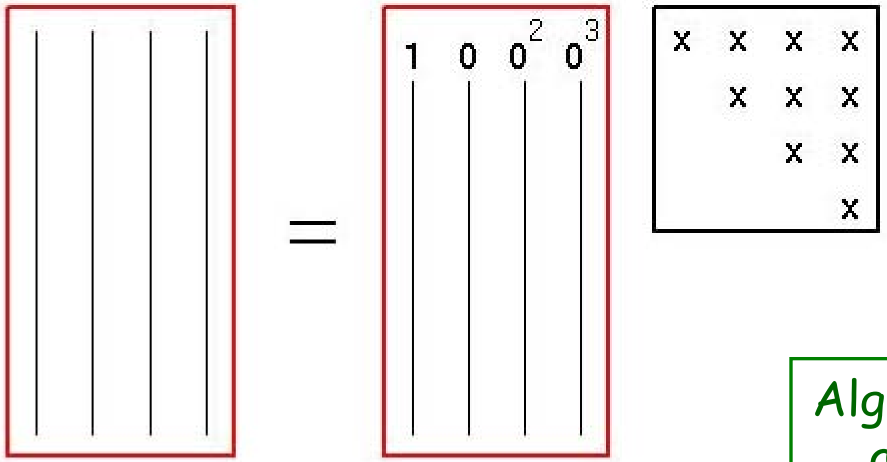
$A = (QU)SV^*$

Demo 3: SVD

LU (without pivoting)

$$\begin{bmatrix} x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \\ x & x & x & x \end{bmatrix} = \begin{bmatrix} 1 & & & \\ x & 1 & & \\ x & x & 1 & \\ x & x & x & 1 \\ x & x & x & x \\ x & x & x & x \end{bmatrix} \begin{bmatrix} x & x & x & x \\ & x & x & x \\ & & x & x \\ & & & x \end{bmatrix}$$

$A = LU$



$A = LU$

Algorithm: column operations,
analogous to modified Gram-Schmidt

And what does this mean?? Well if x is a vector, f = Ax is a function, and we could work with f(a), f'(a), f"(a), … via the LU decomposition.

Demo 4: LU

EPILOGUE

Two other chebfun developments linking continuous and discrete:

KRYLOV SUBSPACE ITERATIONS   Toby Driscoll + Sheehan Olver

GMRES for differential and integral operators.

Theorem by Olver concerning $f \in \mathrm{span}(f', f'', \dots)$ (*SINUM*, submitted)

"CHEBOPS"   Folkmar Bornemann + Toby Driscoll

Adaptive spectral realization of operators in chebfun system.

Example:

```
[d,x] = domain(-10,10);
L = diff(d,2) + diag(100*sin(x));
L.bc = 'dirichlet';
u = L\1;
plot(u)
eigs(u)
```