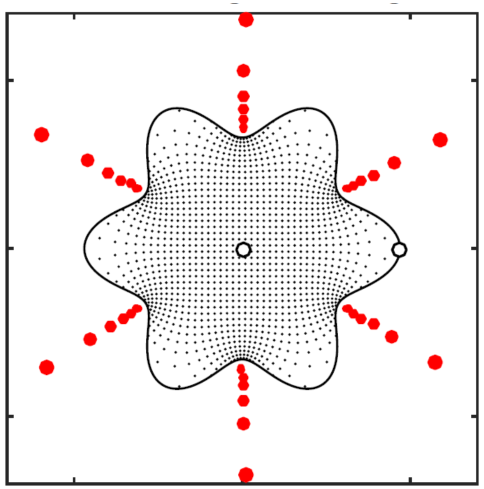
NUMERICAL COMPUTATION WITH RATIONAL FUNCTIONS

(scalars only)

Nick Trefethen, University of Oxford and ENS Lyon

+ thanks to Silviu Filip, Abi Gopal, Stefan Güttel, Yuji Nakatsukasa, and Olivier Sète

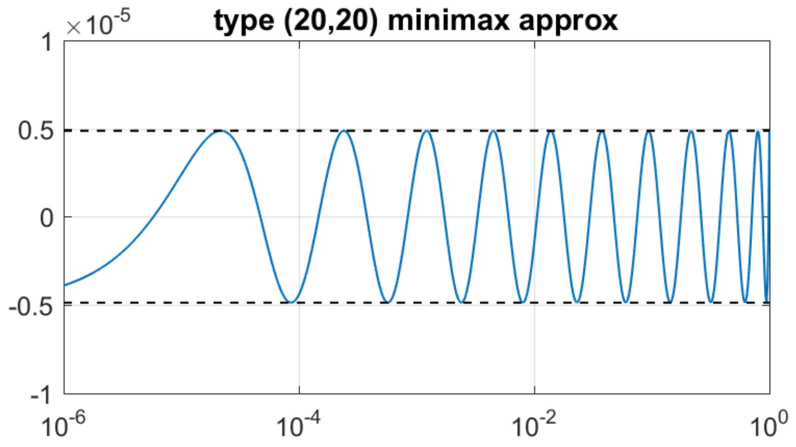


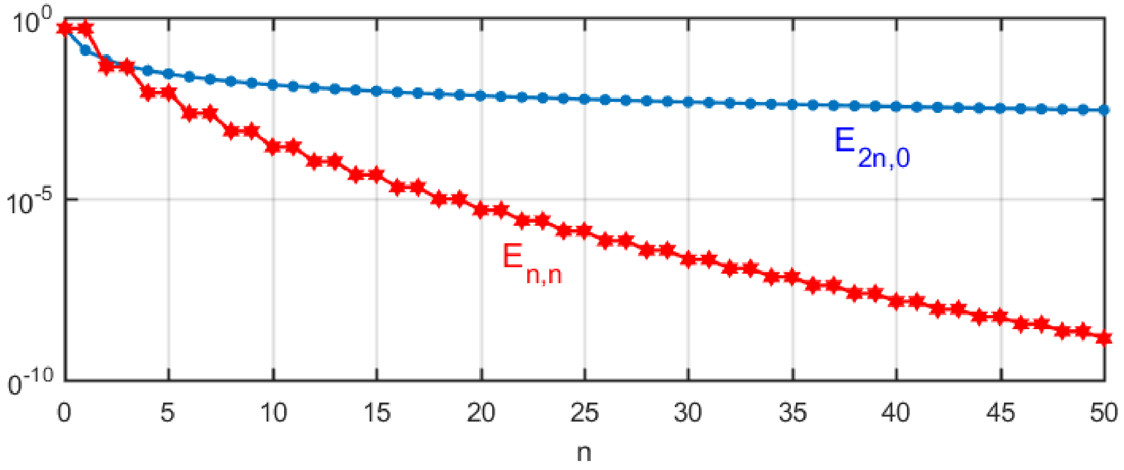
1. Polynomial vs. rational approximations
2. Four representations of rational functions
    2a. Quotient of polynomials
    2b. Partial fractions
    2c. Quotient of partial fractions (= barycentric)
    2d. Transfer function/matrix pencil
3. The AAA algorithm — with Nakatsukasa and Sète, to appear in *SISC*
4. Application: conformal maps — with Gopal, submitted to *Numer. Math.*
5. Application: minimax approximation — with Filip, Nakatsukasa, and Beckermann, to appear in *SISC*
6. Accuracy and noise

1. Polynomial vs. rational approximation

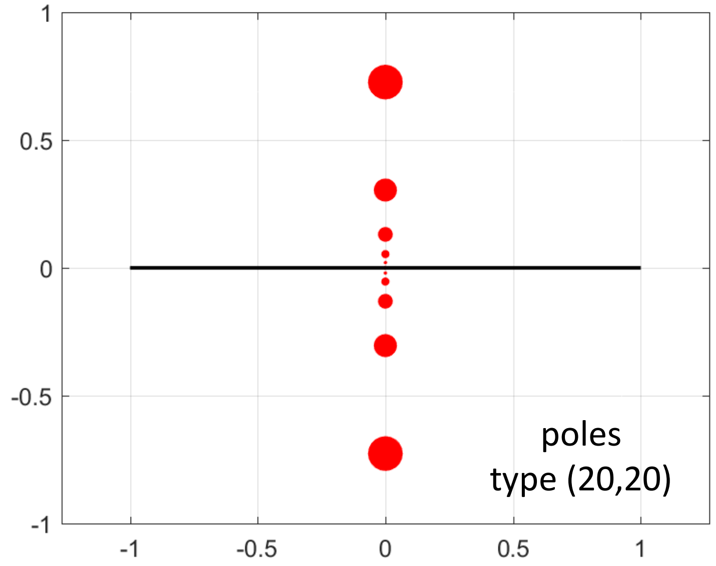
Newman, 1964: approximation of |x| on [−1,1]

$$E_{n0} \sim 0.2801\ldots/n\,, \qquad E_{nn} \sim 8e^{-\pi\sqrt{n}}$$





Poles and zeros of r:
exponentially clustered near $x=0$,
exponentially diminishing residues.



Rational approximation is nonlinear, so algorithms are nontrivial.
There may be nonuniqueness and local minima.

2. Four representations of rational functions

Quotient of polynomials	$p(z)/q(z)$	SK, IRF, AGH, ratdisk	Alpert, Carpenter, Coelho, Gonnet, Greengard, Hagstrom, Koerner, Levy, Pachón, Phillips, Ruttan, Sanathanen, Silantyev, Silveira, Varga, White,…
Partial fractions	$\sum \frac{a_k}{z - z_k}$	VF, exponential sums	Beylkin, Deschrijver, Dhaene, Drmač, Greengard, Gustavsen, Hochman, Mohlenkamp, Monzón, Semlyen,…
Quotient of partial fractions (= barycentric)	$n(z)/d(z)$	Floater-Hormann, AAA	Berrut, Filip, Floater, Gopal, Hochman, Hormann, Ionita, Klein, Mittelmann, Nakatsukasa, Salzer, Schneider, Sète, Trefethen, Werner,…
Transfer function/matrix pencil	$c^T(zB - A)^{-1}b$	IRKA, Loewner, RKFIT	Antoulas, Beattie, Beckermann, Berljafa, Druskin, Elsworth, Gugercin, Güttel, Knizhnerman, Meerbergen, Ruhe,…

Sometimes the boundaries are blurry!

2a. Quotient of polynomials

$r(z) = p(z)/q(z)$

IRF = Iterated rational fitting. Coelho-Phillips-Silveira 1999.
AGH. Alpert-Greengard-Hagstrom 2000.
ratdisk/ratinterp. Gonnet-Pachón-Trefethen 2011.

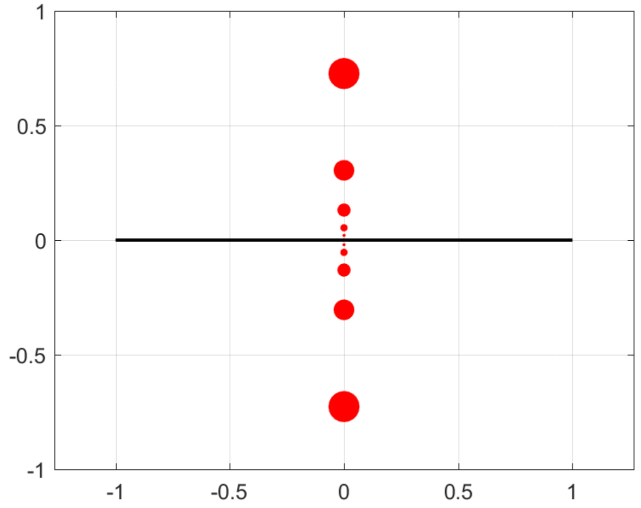
Need a good basis, such as —

disk:	monomials
interval:	Chebyshev
connected region:	Faber polynomials
disconnected region:	Faber-Walsh polynomials (Liesen & Sète)

The first two above construct a good basis along the way.

However, for problems with singularities, $r = p/q$ remains problematic regardless of the basis. (Carpenter-Ruttan-Varga 1993 used 200-digit arithmetic.)

Reason: if poles and zeros are clustered, p and q are exponentially graded. So p/q will be inaccurate where p and q are small.
(p/q is an exponentially ill-conditioned function of p and q.)



2b. Partial fractions

$$r(z) = \sum \frac{a_k}{z - z_k}$$

VF = Vector fitting. Gustavsen-Semlyen 1999. 2390 citations at Google Scholar!

Exponential sums. Beylkin-Monzón, 2005.

Much better behaved than p/q. Ill-conditioning often exponential, yet it's not clear this hurts much. (Unanswered questions here…. related to frames?)

An advantage is that the poles z_k are explicitly present, which is often helpful if we want to manipulate them, e.g. to exclude them from a certain interval or region.

Note that partial fractions in this form cannot represent multiple poles.
Philosophically, this is perhaps analogous to breakdown of a Lanczos iteration
(which is related to degeneracies in the Padé table).

2c. Quotient of partial fractions (= barycentric)

$$r(z) = \sum \frac{a_k}{z-z_k} \Big/ \sum \frac{b_k}{z-z_k}$$

Salzer 1981, Schneider & Werner 1986, Antoulas & Anderson 1986, Berrut 1988

Floater-Hormann. F & H 2007.

AAA = adaptive Antoulas-Anderson. Nakatsukasa-Sète-Trefethen 2018.

Klein thesis 2012 (→ `equi` flag in Chebfun)

THEOREM 2.1 (Rational barycentric representations). *Let $z_1, \ldots, z_m$ be an arbitrary set of distinct complex numbers. As $f_1, \ldots, f_m$ range over all complex values and $w_1, \ldots, w_m$ range over all nonzero complex values, the functions*

$$r(z) = \frac{n(z)}{d(z)} = \sum_{j=1}^{m} \frac{w_j f_j}{z - z_j} \Bigg/ \sum_{j=1}^{m} \frac{w_j}{z - z_j} \qquad (2.5)$$

range over the set of all rational functions of type $(m-1, m-1)$ that have no poles at the points z_j. Moreover, $r(z_j) = f_j$ for each j.

(from the AAA paper)

$\{z_j\}$ are called the support points and can be chosen to enhance stability. They are *not* the poles!

2d. Transfer function/matrix pencil

$$r(z) = c^T(zB - A)^{-1}b$$

Ruhe, 1990s.

Loewner framework. Mayo-Antoulas 2007.

IRKA = Iterative rational Krylov. Antoulas-Beattie-Gugercin 2008.

RKFIT. Berljafa-Güttel 2015 (→ RKFUN).

These representations are fully imbedded in numerical linear algebra/model order reduction.
Scalar problems are just a special case where certain matrices are diagonal.
The poles of r are the eigenvalues of the pencil $[A, B]$.
The Loewner framework uses SVD to choose poles.
Rational Krylov constructs orthogonal bases iteratively.

3. The AAA algorithm

= "adaptive Antoulas-Anderson". Fall 2016.

Yuji Nakatsukasa

THE AAA ALGORITHM FOR RATIONAL APPROXIMATION

YUJI NAKATSUKASA*, OLIVIER SÈTE†, AND LLOYD N. TREFETHEN‡

For Jean-Paul Berrut, the pioneer of numerical algorithms based on rational barycentric representations, on his 65th birthday.

SISC, to appear

Olivier Sète

AAA Algorithm

Taking $m = 1,2,\dots,$ choose support points z_m one after another.
Next support point: point z_i where error $|f_i - r(z_i)|$ is largest.
Barycentric weights $\{w_j\}$ at each step:
chosen to minimize linearized least-squares error $||fd - n||$.

Comparied with other methods AAA is ***FAST!***

$$r(z) = \frac{n(z)}{d(z)} = \sum_{j=1}^{m} \frac{w_j f_j}{z - z_j} \Bigg/ \sum_{j=1}^{m} \frac{w_j}{z - z_j}$$

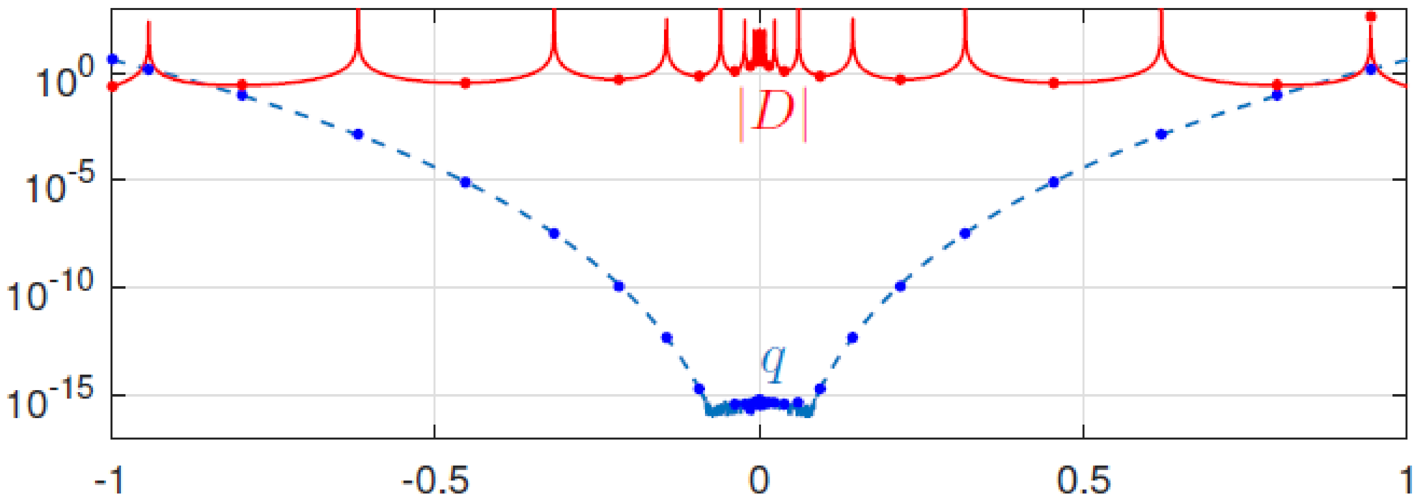
Generalizations also to type (μ, ν) with $\mu \neq \nu$.

cf. Berrut & Mittelmann 1997

Available (for $\mu = \nu$) as aaa in Chebfun.

```
for n = 0:nmax
  [~,j] = max(abs(F-R));        % select next support point
  z = [z; Z(j)];                % update set of support pts
  f = [f; F(j)];                % update set of data values
  J(J==j) = [];                 % update index vector
  C = [C 1./(Z-Z(j))];          % next column of Cauchy mat
  Sf = diag(f);                 % right scaling matrix
  A = SF*C - C*Sf;              % Loewner matrix
  [~,~,V] = svd(A(J,:),0);      % SVD
  w = V(:,end);                 % weight vec = min sing vec
  N = C*(w.*f); D = C*w;        % numerator and denominator
  R = F; R(J) = N(J)./D(J);     % rational approximation
  err = norm(F-R,inf);
end
```

Type $(20{,}20)$ approx of $|x|$ on $[-1{,}1]$



Size of denominator D for AAA representation N/D

Size of denominator q for polynomial representation p/q

AAA demos

```
ezplot(aaa(@gamma))
```

```
Z = randn(1000,1) + 1i*randn(1000,1);
plot(Z,'.k','markersize',4), axis equal, hold on
F = exp(Z)./sin(pi*Z);
tic, [r,pol] = aaa(F,Z); toc
norm(F-r(Z),inf)
plot(pol,'.r'), hold off
pol
```

4. Application: conformal maps

Representation of conformal maps by rational functions

Abinand Gopal · Lloyd N. Trefethen

Numerische Mathematik, submitted

Conformal maps

First, compute the conformal map f by standard methods. For polygons we use Driscoll's Schwarz-Christoffel Toolbox.

Then use AAA to represent the result:

Sample Z and F = f(Z) at a few thousand points on the boundary.

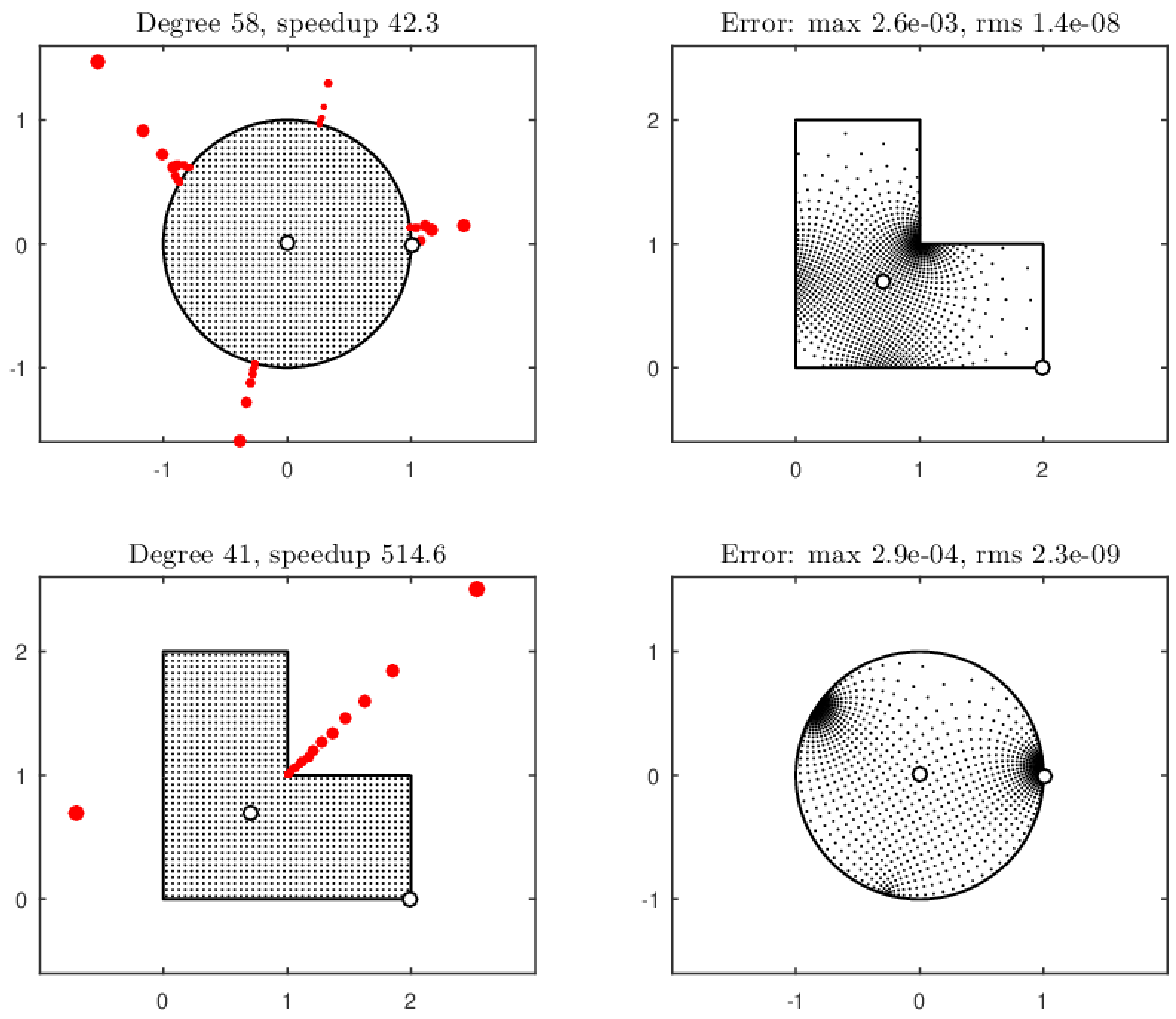
Forward map

```
[r, rpol, rres] = aaa(F, Z, 'tol', 1e-7)
```

Inverse map

```
[s, spol, sres] = aaa(Z, F, 'tol', 1e-7)
```

Note that the inverse map comes for free.
Speedups often by factors in the hundreds.



Conformal maps

First, compute the conformal map f by standard methods. For polygons we use Driscoll's Schwarz-Christoffel Toolbox.

Then use AAA to represent the result:

Sample Z and F $= f$(Z) at a few thousand points on the boundary.

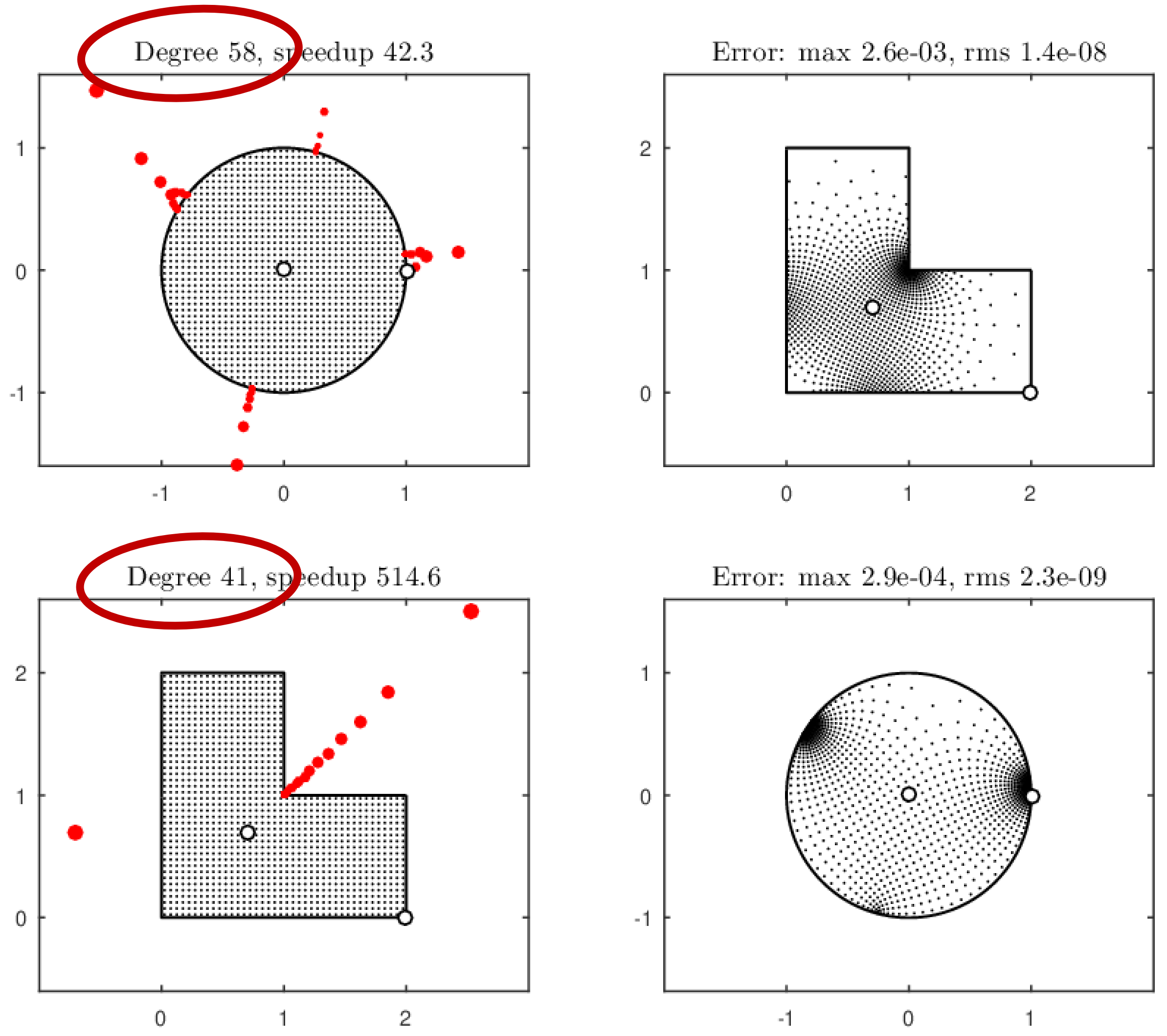
Forward map

```
[r, rpol, rres] = aaa(F, Z, 'tol', 1e-7)
```

Inverse map

```
[s, spol, sres] = aaa(Z, F, 'tol', 1e-7)
```

Note that the inverse map comes for free.
Speedups often by factors in the hundreds.



Conformal maps

First, compute the conformal map f by standard methods. For polygons we use Driscoll's Schwarz-Christoffel Toolbox.

Then use AAA to represent the result:

Sample Z and F $= f$(Z) at a few thousand points on the boundary.

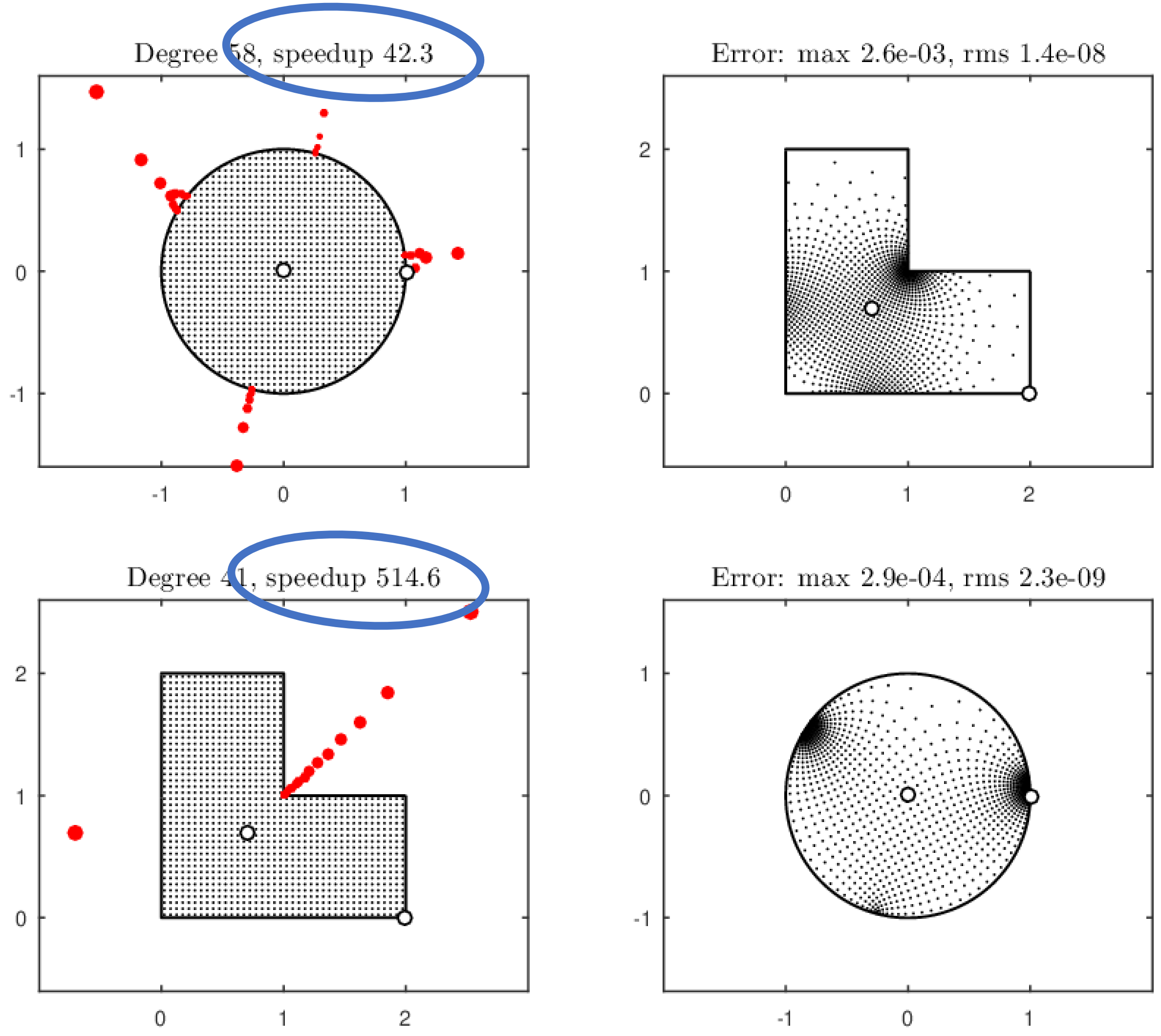
Forward map

```
[r, rpol, rres] = aaa(F, Z, 'tol', 1e-7)
```

Inverse map

```
[s, spol, sres] = aaa(Z, F, 'tol', 1e-7)
```

Note that the inverse map comes for free.
Speedups often by factors in the hundreds.



Conformal maps

First, compute the conformal map f by standard methods. For polygons we use Driscoll's Schwarz-Christoffel Toolbox.

Then use AAA to represent the result:

Sample Z and F $= f$(Z) at a few thousand points on the boundary.

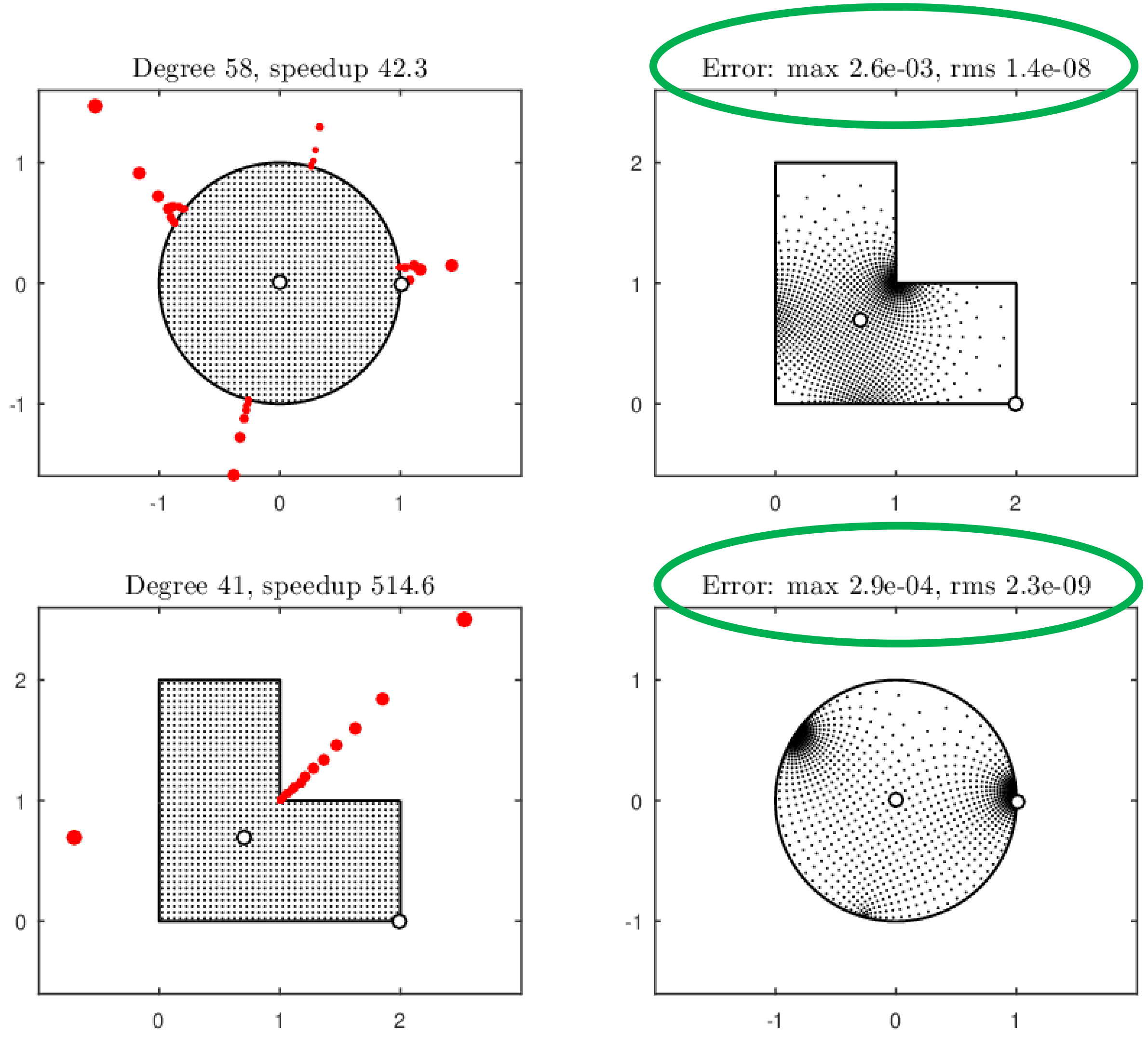
Forward map

```
[r, rpol, rres] = aaa(F, Z, 'tol', 1e-7)
```

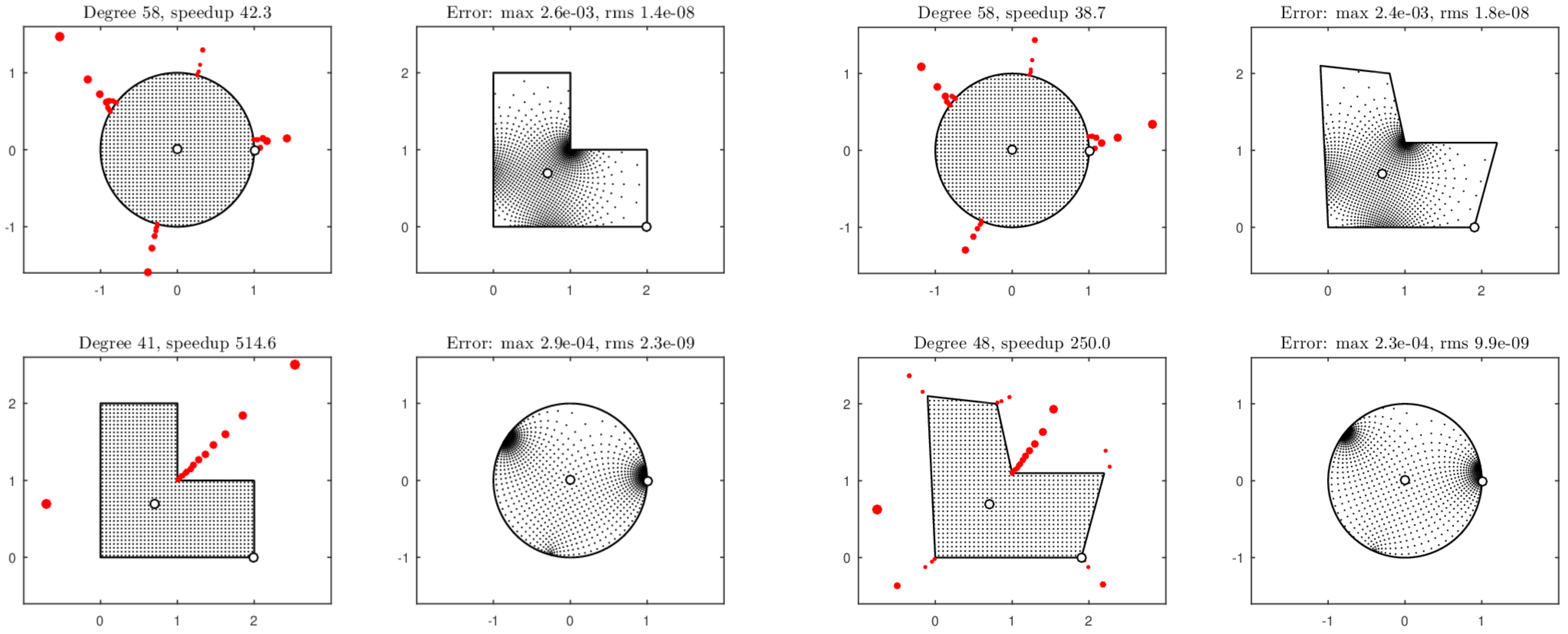
Inverse map

```
[s, spol, sres] = aaa(Z, F, 'tol', 1e-7)
```

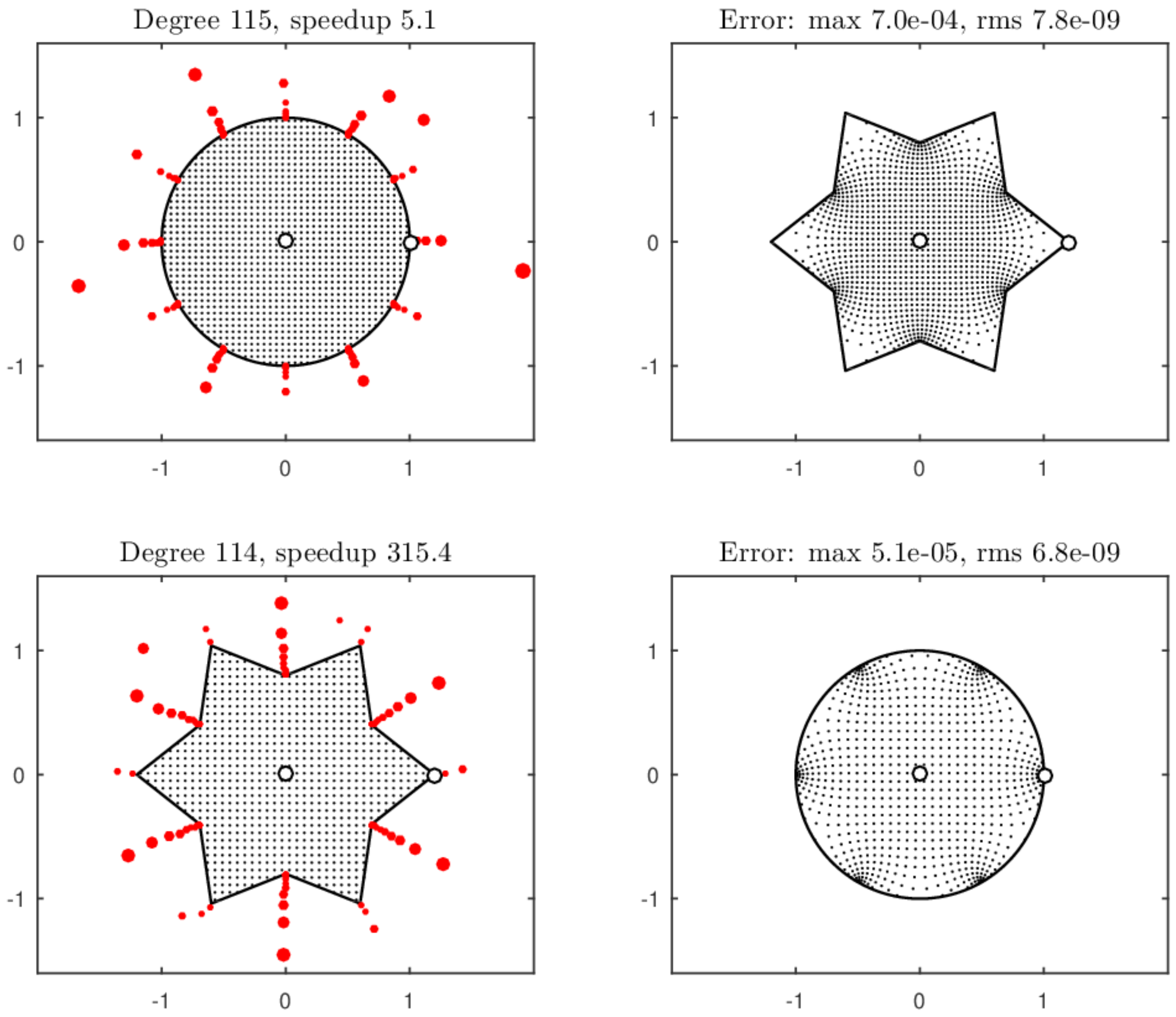
Note that the inverse map comes for free.
Speedups often by factors in the hundreds.



Perturbed L shape

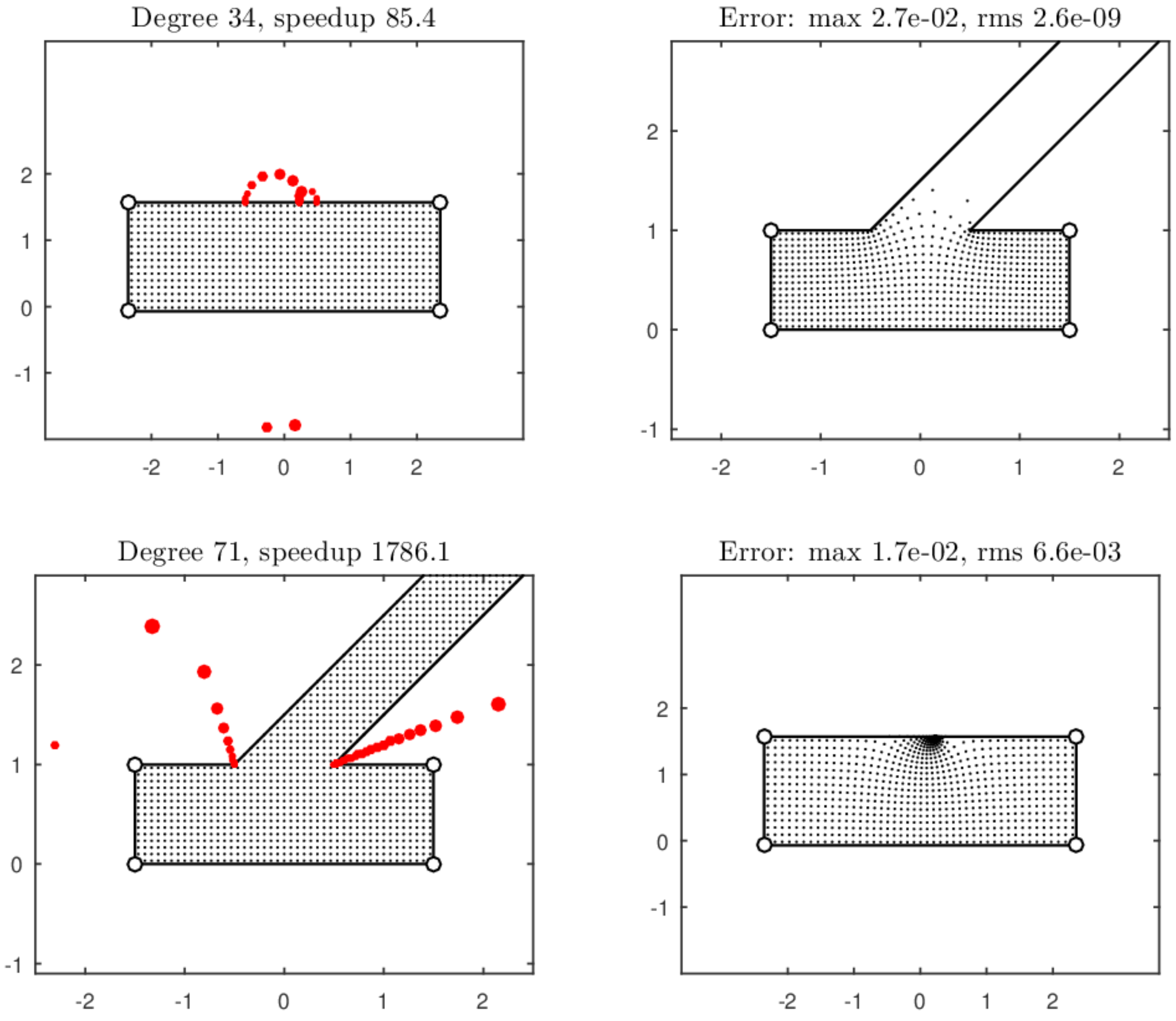


Snowflake



Degree 115, speedup 5.1
Error: max 7.0e-04, rms 7.8e-09
Degree 114, speedup 315.4
Error: max 5.1e-05, rms 6.8e-09
1
0
-1
-1
0
1

Unbounded domain



Root-exponential convergence

The classic problem concerns $|x|^{\alpha}$. (Zolotarev, Newman, Vyacheslavov, Stahl,...)

For conformal maps of regions with corners we need the complex function x^{α}. This is different, and not a corollary.

We've proved root-exponential convergence via a trapezoidal rule estimate adapted from *ATAP,* pp. 221-212. (Gopal & T., submitted to *Numer. Math.*)

$$x^{\alpha} = C\int_0^{\infty} \frac{x\,dt}{t^{1/\alpha}+x}, \qquad C = \frac{\sin(\alpha\pi)}{\alpha\pi} \qquad t = e^{\alpha\pi i/2+s}$$

$$x^{\alpha} = C\int_{-\infty}^{\infty} \frac{x\,e^{\alpha\pi i/2+s}ds}{e^{\pi i/2+s/\alpha}+x}$$

$$r(x) = hC\sum_{k=-(n-1)/2}^{(n-1)/2} \frac{x\,e^{\alpha\pi i/2+kh}}{e^{\pi i/2+kh/\alpha}+x}$$

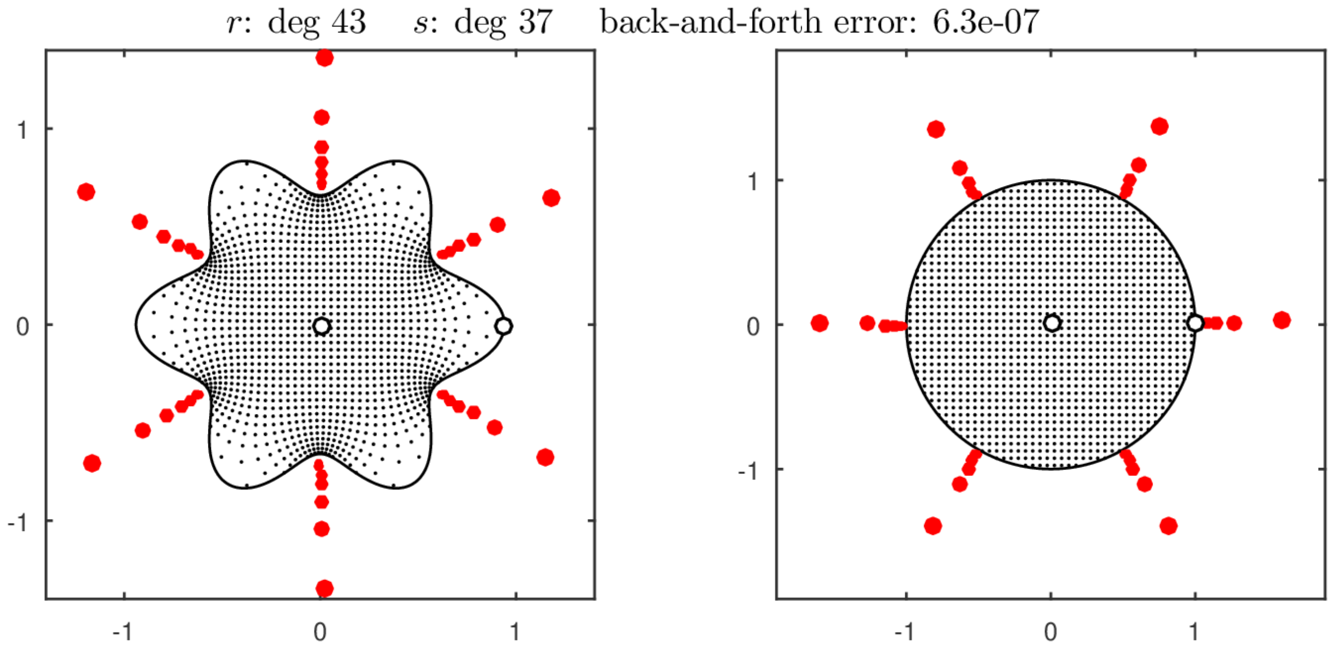
$$h = \pi\sqrt{2\alpha/n} \qquad \|r_n(x) - x^{\alpha}\|_H \lesssim \exp(-\pi\sqrt{\alpha n/2})$$

Smooth domains

Following Caldwell, Li, and Greenbaum, we use the Kerzman-Stein integral equation as discretized by Kerzman and Trummer (taking 800 points on the boundary).

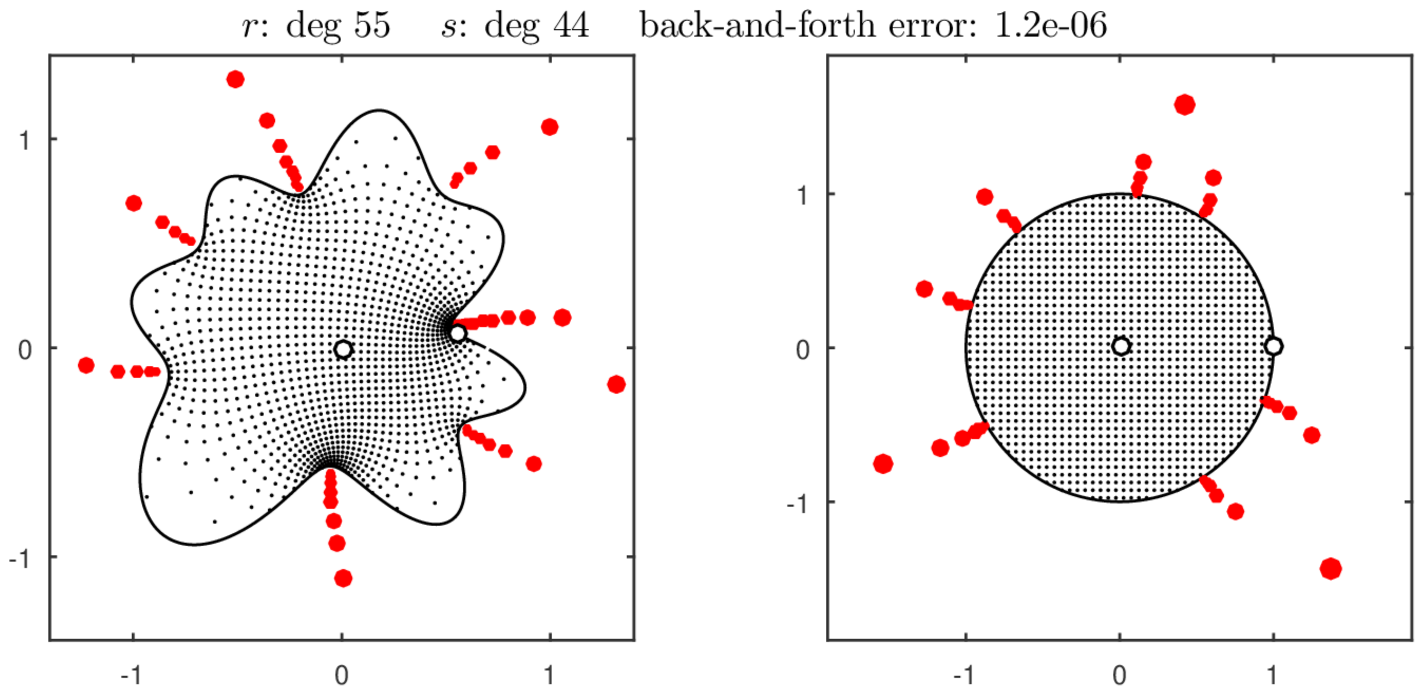
Then AAA represents the map and its inverse:

```
[r, rpol, rres] = aaa(F, Z, 'tol', 1e-6)
[s, spol, sres] = aaa(Z, F, 'tol', 1e-6)
```



Random boundary

(defined by Chebfun `randnfun`)



Note that maps involving analytic boundaries
may have singularities exponentially close.
Polynomial approximations would be unworkable.

5. Application: minimax approximation

RATIONAL MINIMAX APPROXIMATION VIA ADAPTIVE BARYCENTRIC REPRESENTATIONS

SILVIU-IOAN FILIP*, YUJI NAKATSUKASA†, LLOYD N. TREFETHEN†, AND BERNHARD BECKERMANN‡

SISC, to appear

a loyal friend from Rennes

a man of steel from Lille

Minimax approximation on a real interval

Previous work: Ioniţă 2013
Rice U. / MathWorks

Before 2017, Chebfun failed on type $(10,10)$ approx to $|x|$ (see *ATAP,* p. 192).
Varga-Ruttan-Carpenter 1993 got to type $(80,80)$, but using 200-digit arithmetic.

Chebfun's new code `minimax` can do type $(80,80)$ in 16-digit arithmetic!
Types (m, n) with $m \neq n$ are also allowed.

Key advance: barycentric representation.

This was successful for three different minimax methods:

(1) Remez algorithm (Werner 1962, Maehly 1963, Curtis & Osborne 1966)
(2) "AAA-Lawson" algorithm (AAA in noninterpolatory mode, iterative reweighting) (cf. Lawson 1961)
(3) differential correction algorithm (making key use of linear programming) (Cheney and Loeb 1961)

Remez can still be difficult because of the initialization problem. `minimax` uses, as necessary:

- CF approximation (SVD of Hankel matrix of Chebyshev coefficients)
- AAA-Lawson
- stepping up from smaller types (m, n)

minimax demos

```
f = @(x) sqrt(abs(x+.5)) + abs(x-.5);
xx = linspace(-1,1,20000);

[p,q,r] = minimax(f,100,0); plot(xx, f(xx)-r(xx)), grid on
                    0,100

[p,q,r] = minimax(f,10,10); plot(xx, f(xx)-r(xx)), grid on
                    20,20    40,40    80,80

plot(xx,r(xx))
```

6. Accuracy and noise

Q1: Are they capable of representing difficult rational functions?

Quotient of polynomials	$p(z)/q(z)$	*NO*
Partial fractions	$\sum \frac{a_k}{z - z_k}$	*COMPLICATED !*
Quotient of partial fractions (= barycentric)	$n(z)/d(z)$	*YES*
Transfer function/matrix pencil	$c^T(zB - A)^{-1}b$	*YES*

The matrices usually have awful condition numbers, like 10^8 or worse. Yet often the fits are good anyway.

aaacompare demos

(AAA vs. partial fractions least-squares fit using the same AAA poles)

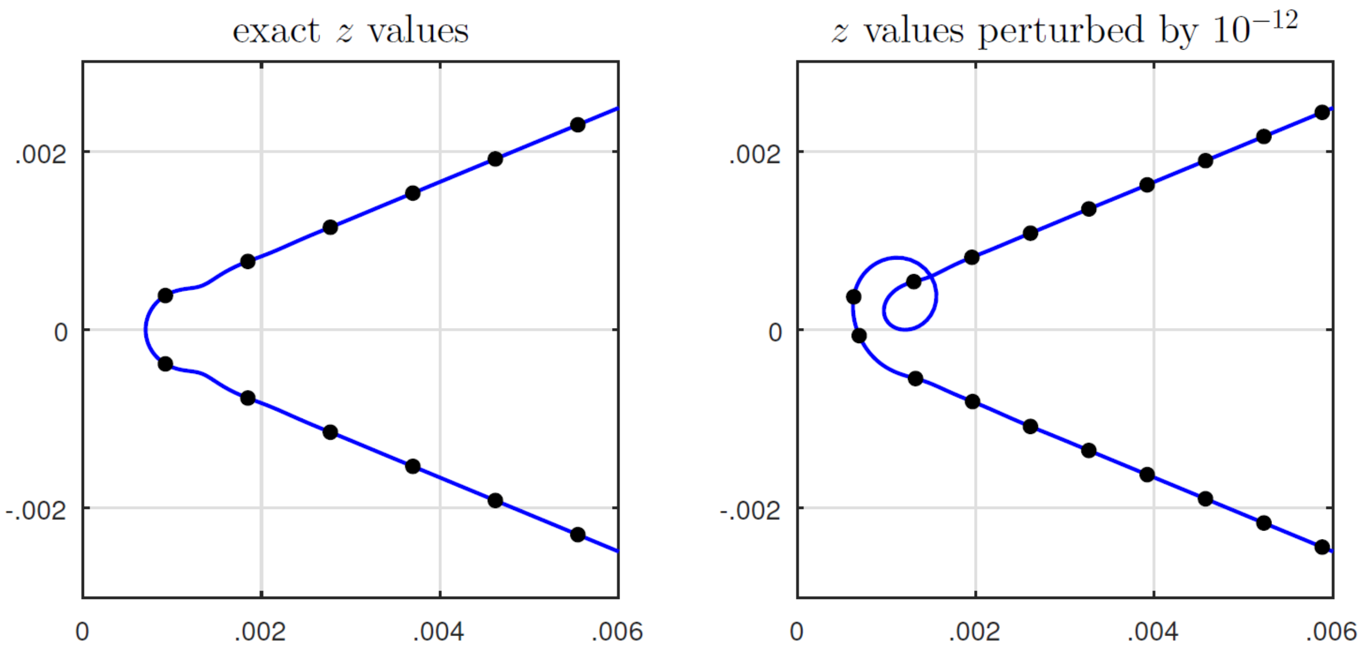
```
exp(x)

abs(x-0.1i)

sin(1/(1.1-x))
```

What happens with noisy data?

Effect on AAA of small errors in a fit of $x^{1/4}$

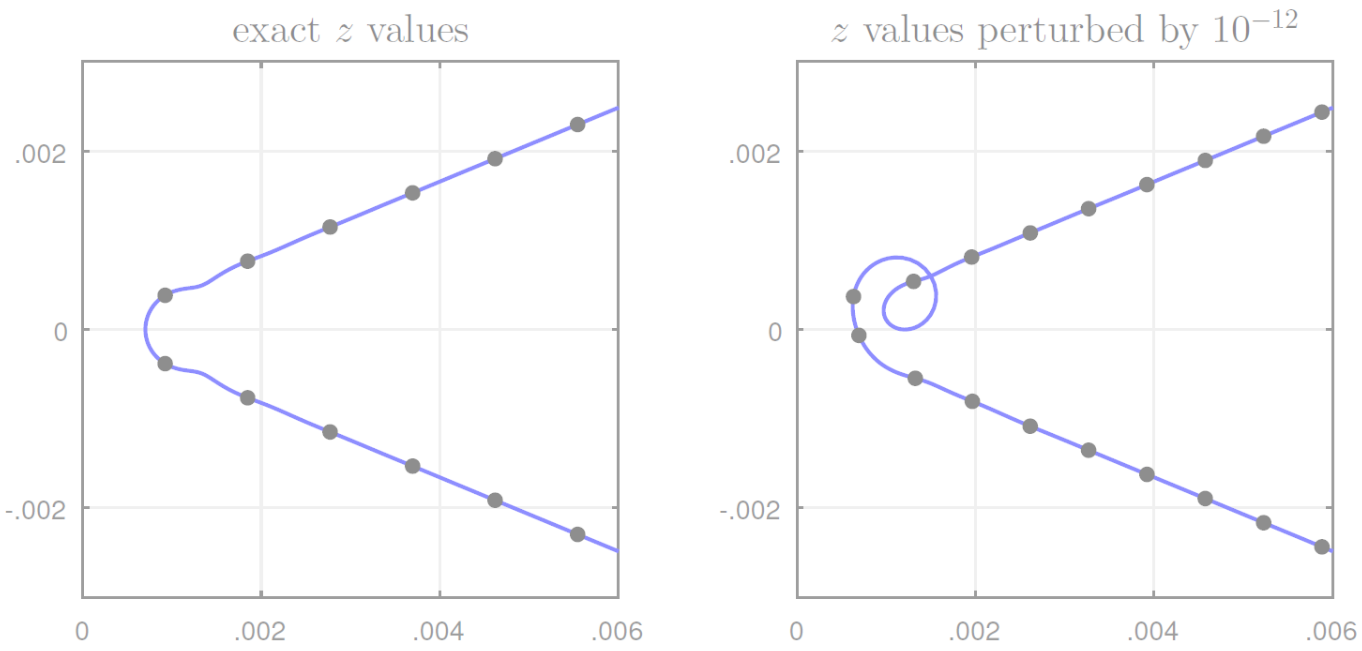


The noise has led to an unwanted pole-zero pair — a Froissart doublet.

Currently AAA seems more fragile than RKFIT, and we are investigating.

What happens with noisy data?

Effect on AAA of small errors in a fit of $x^{1/4}$

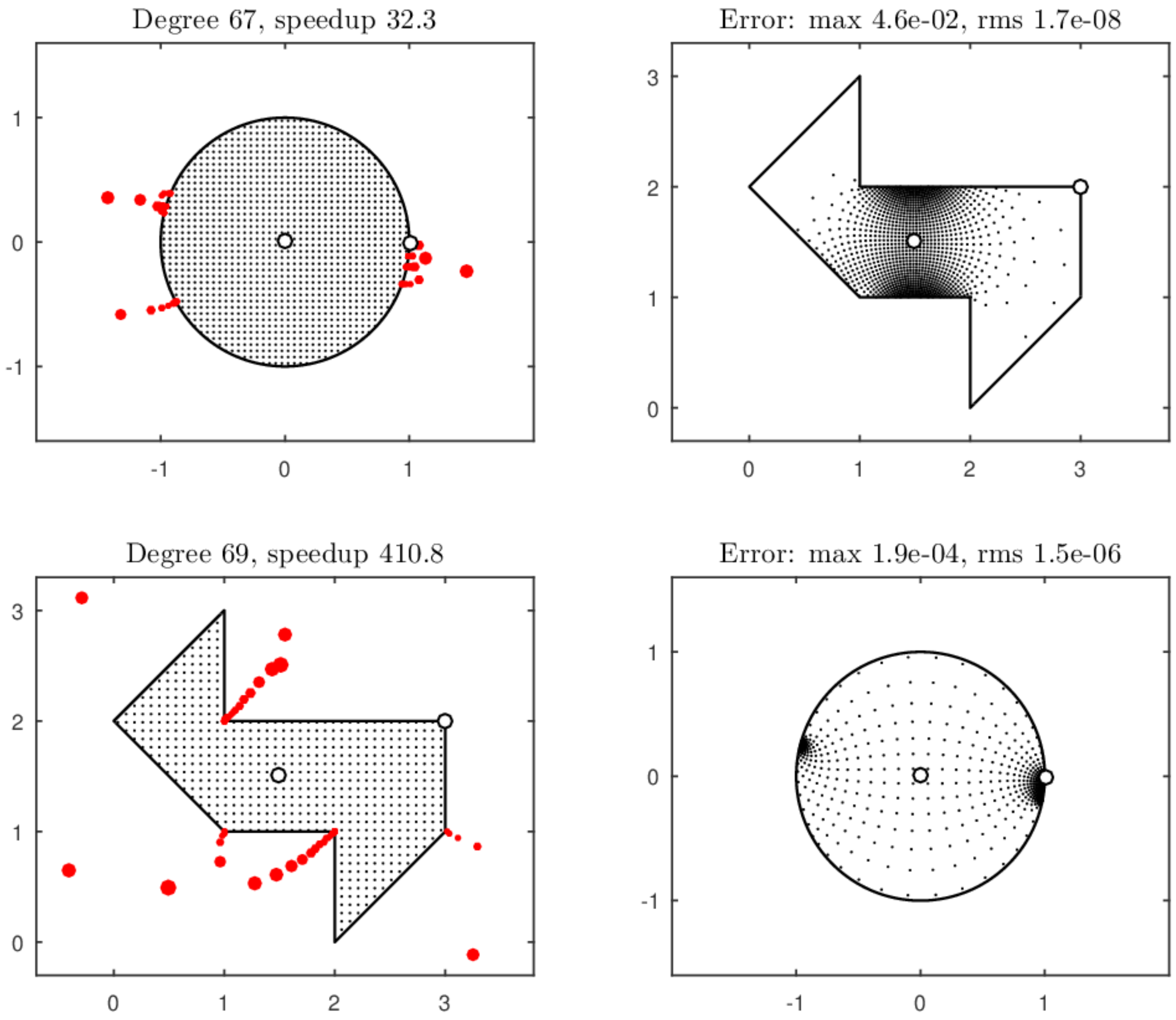


The noise has led to an unwanted pole-zero pair — a Froissart doublet.

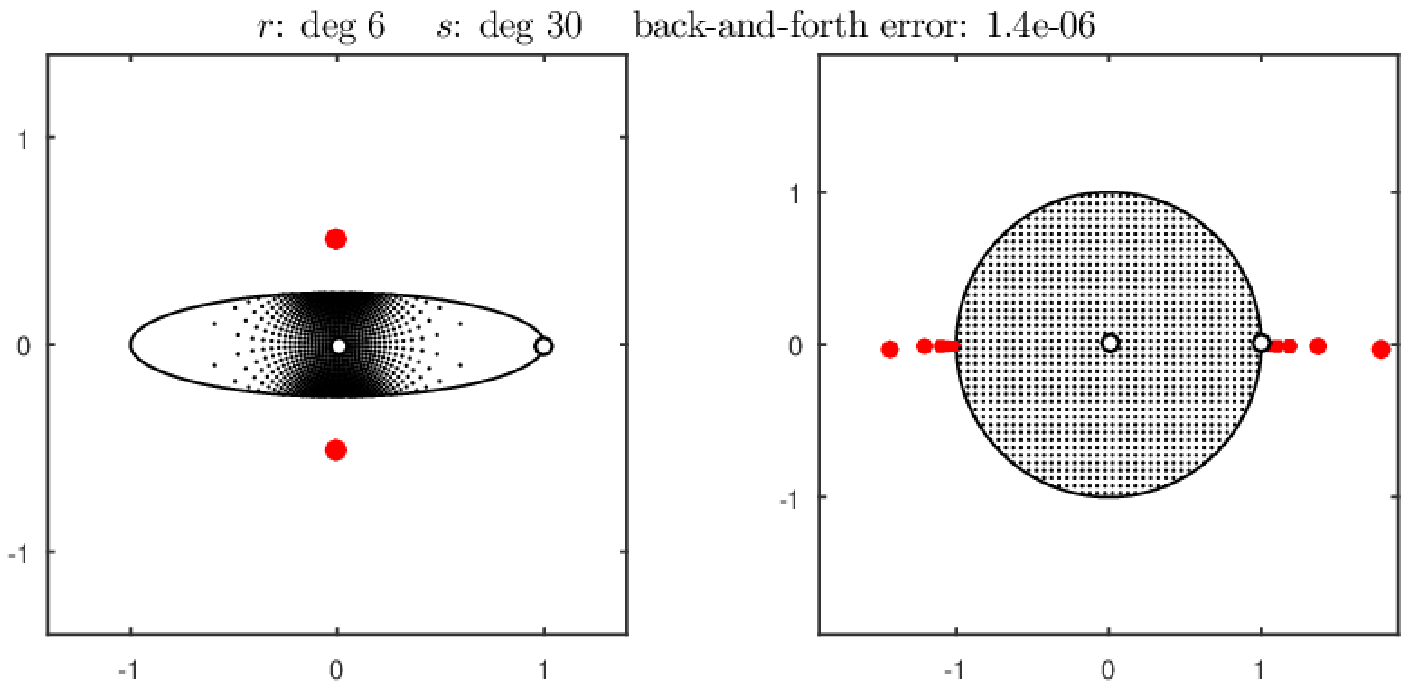
Currently AAA seems more fragile than RKFIT, and we are investigating.

Thank you Ana, Karl, Laurent and Bernd!

Isospectral drum

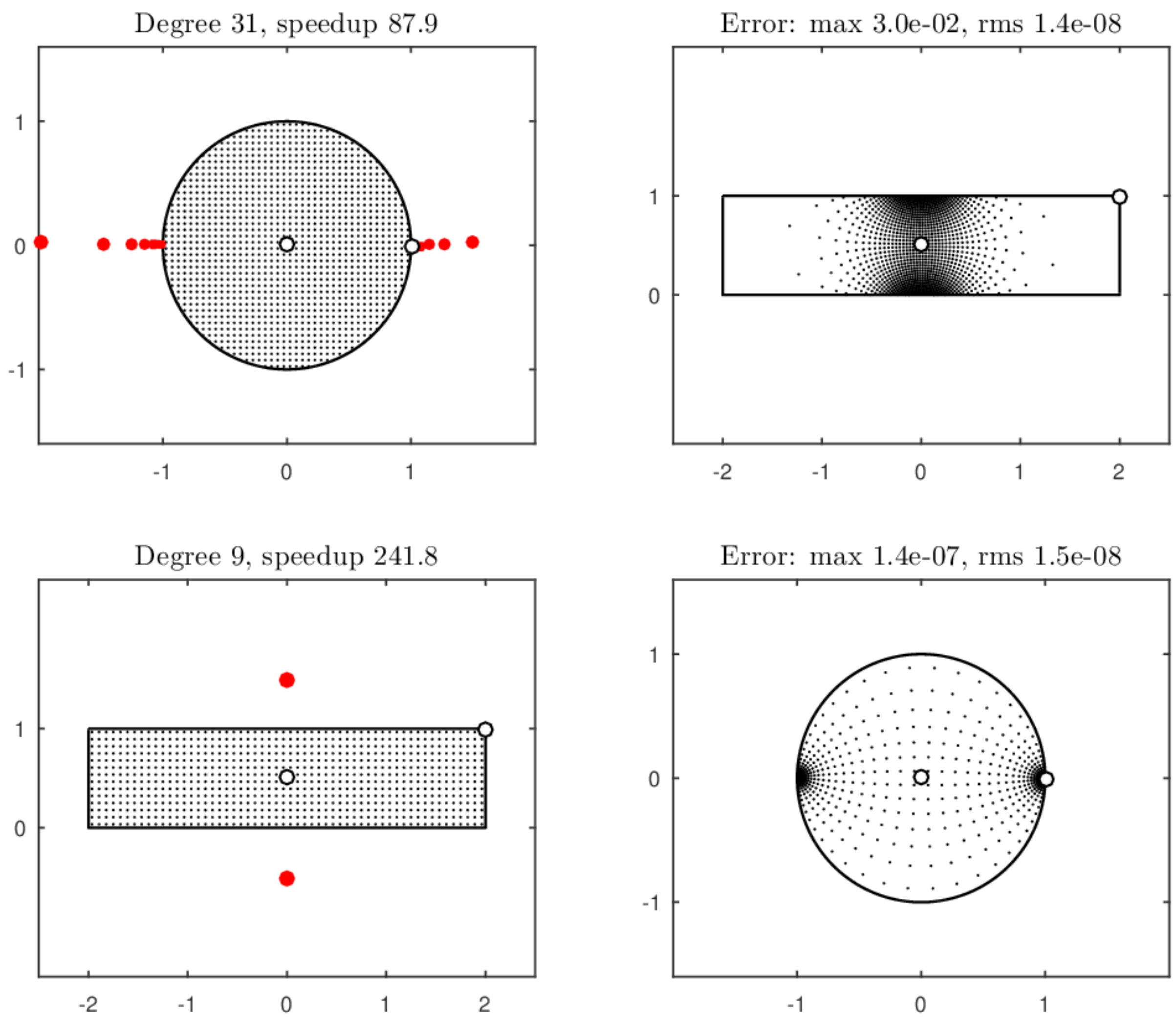


Ellipse



Note that maps involving analytic boundaries may have singularities exponentially close (here, 3×10^{-5}). Polynomial approximations would be unworkable.

Rectangle



Degree 31, speedup 87.9
Error: max 3.0e-02, rms 1.4e-08
Degree 9, speedup 241.8
Error: max 1.4e-07, rms 1.5e-08

Illustration of possible ill-conditioning of partial fractions representation

Approximate e^{-x^2} on $[-1,1]$

AAA represents the function to 15 digits with 10 poles.

Here we make a 10000×10 matrix from these poles and do a least-squares fit. 7 digits are lost.