Integral Equations and Fast Algorithms
Lecture 25: Singular quadrature, Intro Fast Algorithms

CS 598AK · November 19, 2013

Outline

Outline

Kussmaul/Martensen (aka "Kress")

Theorem (A special integral [Kress LIE Lemma 8.21])

$$\frac{1}{2\pi}\int_0^{2\pi} \log\left(4\sin^2\frac{t}{2}\right) e^{imt}\,dt = \begin{cases} 0 & m = 0, \\ -\frac{1}{|m|} & m = \pm 1, \pm 2 \ldots . \end{cases}$$

Kussmaul/Martensen (aka "Kress")

Theorem (A special integral [Kress LIE Lemma 8.21])

$$\frac{1}{2\pi}\int_0^{2\pi} \log\left(4\sin^2\frac{t}{2}\right)e^{imt}dt = \begin{cases} 0 & m = 0, \\ -\frac{1}{|m|} & m = \pm 1, \pm 2 \ldots. \end{cases}$$

Exciting?

KM quadrature demo

KM quadrature demo

DOF choice for KM. . . ?

KM quadrature demo

DOF choice for KM. . . ?

Describe scheme

KM quadrature demo

DOF choice for KM. . . ?

Describe scheme

Bigger idea hiding in KM. . . ?

KM quadrature demo

DOF choice for KM. . . ?

Describe scheme

Bigger idea hiding in KM. . . ?

What if you knew how to integrate Laplace and wanted to do Helmholtz?

Singularity subtraction

$$\int \langle \text{Thing } X \text{ you would like to integrate} \rangle$$
$$= \int \langle \text{Thing } Y \text{ you \textit{can} integrate} \rangle$$
$$+ \int \langle \text{Difference } X - Y \text{ which is easy to integrate (numerically)} \rangle$$

Singularity subtraction

$$\int \langle \text{Thing } X \text{ you would like to integrate} \rangle$$
$$= \int \langle \text{Thing } Y \text{ you } \textit{can} \text{ integrate} \rangle$$
$$+ \int \langle \text{Difference } X - Y \text{ which is easy to integrate (numerically)} \rangle$$

Drawback?

High-Order Corrected Trapezoidal Quadrature

- Conditions for new nodes, weights ($\rightarrow$ linear algebraic system, dep. on n) to integrate

$$\langle \text{smooth} \rangle \cdot \langle \text{singular} \rangle + \langle \text{smooth} \rangle$$

- Allowed singularities: $|x|^{\lambda}$ (for $|\lambda| < 1$), $\log |x|$
- Generic nodes and weights for log singularity
- Nodes and weights copy-and-pasteable from paper

[Kapur, Rokhlin '97]

High-Order Corrected Trapezoidal Quadrature

- Conditions for new nodes, weights
  ($\rightarrow$ linear algebraic system, dep. on n)
  to integrate

$\langle$sm

- Allowed singulariti
- Generic nodes and
- Nodes and weights

[Kapur, Rokhlin '97]

[Alpert '99] conceptually similar:

- Hybrid Gauss-Trapezoidal
- Positive weights
- Somewhat more accurate (empirically) than K-R
- Similar allowed singularities ($\lambda > -1$)
- Copy-paste weights

Generalized Gaussian

- "Gaussian":
  - Integrates $2n$ functions exactly with n nodes
  - Positive weights
- Clarify assumptions on system of functions ("Chebyshev system") for which Gaussian quadratures exist
- When do (left/right) singular vectors of integral operators give rise to Chebyshev systems?
  - In many practical cases!
- Find nodes/weights by Newton's method
  - With special starting point
- Very accurate
- Nodes and weights for download

[Yarvin/Rokhlin '98]

Generalized Gaussian

- "Gaussian":
  - Integrates $2n$ functions exactly with n nodes
  - Positive weights
- Clarify assumptions on system of functions ("Chebyshev system") for which Gaussian quadratures exist
- When do (left/right) singular vectors of integral operators give rise to Chebyshev systems?
  - In many practical cases!
- Find nodes/weights by Newton's method
  - With special starting point
- Very accurate
- Nodes and weights for download

[Yarvin/Rokhlin '98]

Generalizes to nD...

Generalized Gaussian

- "Gaussian":
  - Integrates $2n$ functions exactly with n nodes
  - Positive weights
- Clarify assumptions on system of functions ("Chebyshev system") for which Gaussian quadratures exist
- When do (left/right) singular vectors of integral operators give rise to Chebyshev systems?
  - In many practical cases!
- Find nodes/weights by Newton's method
  - With special starting point
- Very accurate
- Nodes and weights

[Yarvin/Rokhlin '98]

Generalizes to nD...

...if you know how to make Newton's method converge

Singularity cancellation: Polar coordinate transform

$$\iint_{\partial\Omega} K(\mathbf{x},\mathbf{y})\varphi(y)ds_y$$

$$=$$

$$\int_0^R \int_{\mathbf{x}+\mathbf{r}\in\partial\Omega\cap\partial B(\mathbf{x},r)} K(\mathbf{x},\mathbf{x}+\mathbf{r})\varphi(\mathbf{x}+\mathbf{r})d\langle\text{angles}\rangle\, r\, dr$$

$$=$$

$$\int_0^R \int_{\mathbf{x}+\mathbf{r}\in\partial\Omega\cap\partial B(\mathbf{x},r)} \frac{K_{\text{less singular}}(\mathbf{x},\mathbf{x}+\mathbf{r})}{r}\varphi(\mathbf{x}+\mathbf{r})d\langle\text{angles}\rangle \;\; r \;\; dr$$

where $K_{\text{less singular}} = K \cdot r$

Singularity cancellation: Polar coordinate transform

$$\iint_{\partial\Omega} K(\mathbf{x},\mathbf{y})\varphi(y)ds_y$$

$$=$$

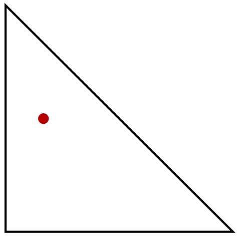
$$\int_0^R \int_{\mathbf{x}+\mathbf{r}\in\partial\Omega\cap\partial B(\mathbf{x},r)} K(\mathbf{x},\mathbf{x}+\mathbf{r})\varphi(\mathbf{x}+\mathbf{r})d\langle\text{angles}\rangle\, r\, dr$$

$$=$$

$$\int_0^R \int_{\mathbf{x}+\mathbf{r}\in\partial\Omega\cap\partial B(\mathbf{x},r)} \frac{K_{\text{less singular}}(\mathbf{x},\mathbf{x}+\mathbf{r})}{r}\varphi(\mathbf{x}+\mathbf{r})d\langle\text{angles}\rangle\, r\, dr$$

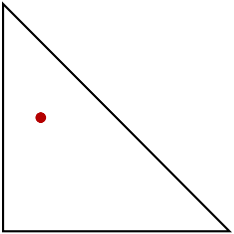
where $K_{\text{less singular}} = K \cdot r$

Quadrature on triangles



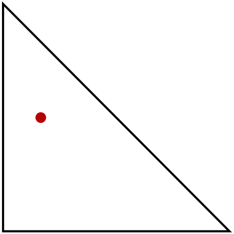
Problem: Singularity can sit *anywhere* in triangle

Quadrature on triangles



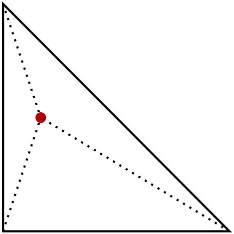
Problem: Singularity can sit *anywhere* in triangle
$\rightarrow$ need *lots* of quadrature rules (one per target)

Quadrature on triangles



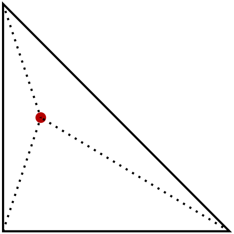
Problem: Singularity can sit *anywhere* in triangle
$\rightarrow$ need *lots* of quadrature rules (one per target) . . . !?

Quadrature on triangles



Problem: Singularity can sit *anywhere* in triangle
$\rightarrow$ need *lots* of quadrature rules (one per target) ...!?

Quadrature on triangles



Problem: Singularity can sit *anywhere* in triangle
$\rightarrow$ need *lots* of quadrature rules (one per target) ...!?

Possible issue?

Kernel Regularization

Singularity makes integration troublesome: *Get rid of it!*

$$\frac{\cdots}{\sqrt{(x-y)^2}} \quad \rightarrow \quad \frac{\cdots}{\sqrt{(x-y)^2+\varepsilon^2}}$$

Use Richardson extrapolation to recover limit as $\varepsilon \to 0$.

(May also use geometric motivation: limit along line towards singular point.)

Primary drawbacks:

- Low-order accurate
- Need to make ε smaller (i.e. kernel more singular) to get better accuracy

Kernel Regularization

Singularity makes integration troublesome: *Get rid of it!*

$$\frac{\cdots}{\sqrt{(x-y)^2}} \quad \to \quad \frac{\cdots}{\sqrt{(x-y)^2+\varepsilon^2}}$$

Use Richardson extrapolation to recover limit as $\varepsilon \to 0$.

(May also use geometric motivation: limit along line towards singular point.)

Primary drawbacks:

- Low-order accurate
- Need to make ε sm
better accuracy

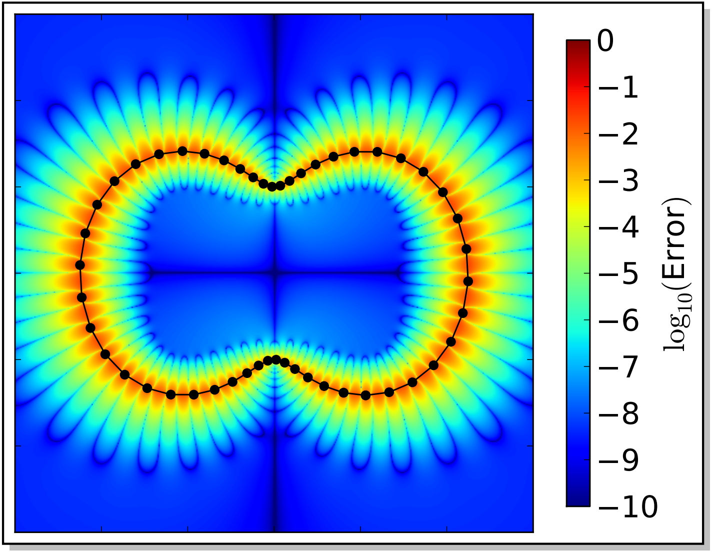
Can take many forms–for example:

- Convolve integrand to smooth it ($\to$ remove/weaken singularity)
- Extrapolate towards no smoothing

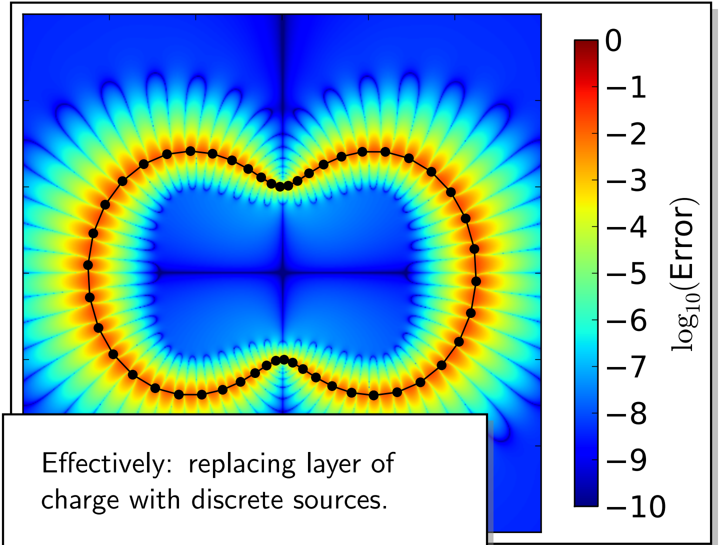
Related: [Beale/Lai '01]

Outline

Using just the trapezoidal rule



Using just the trapezoidal rule

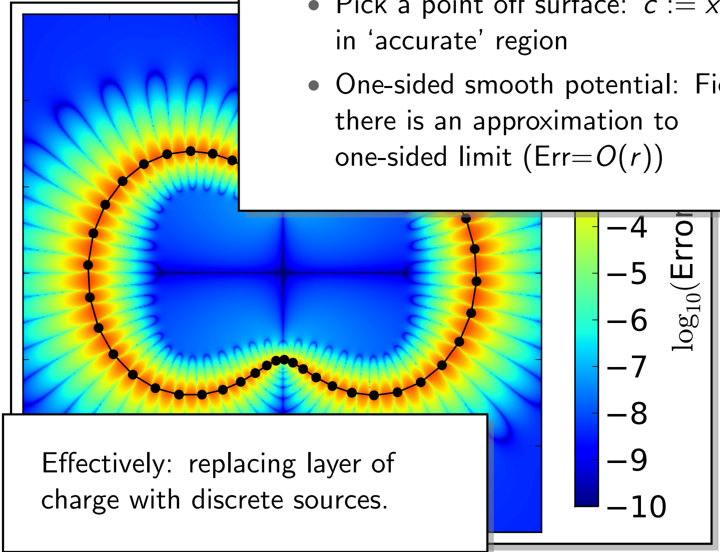


Effectively: replacing layer of charge with discrete sources.

Us

Idea:

- Pick a point off surface: $c := x + \hat{n}r$ in 'accurate' region
- One-sided smooth potential: Field value there is an approximation to one-sided limit (Err$=O(r)$)



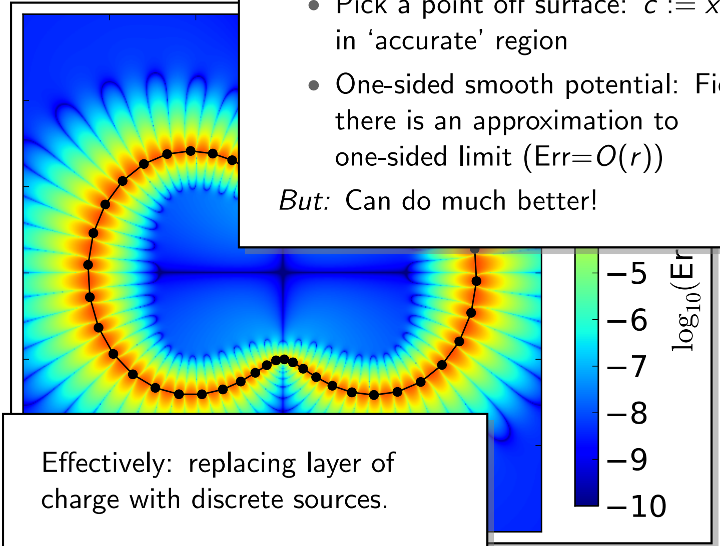
Effectively: replacing layer of charge with discrete sources.

Us

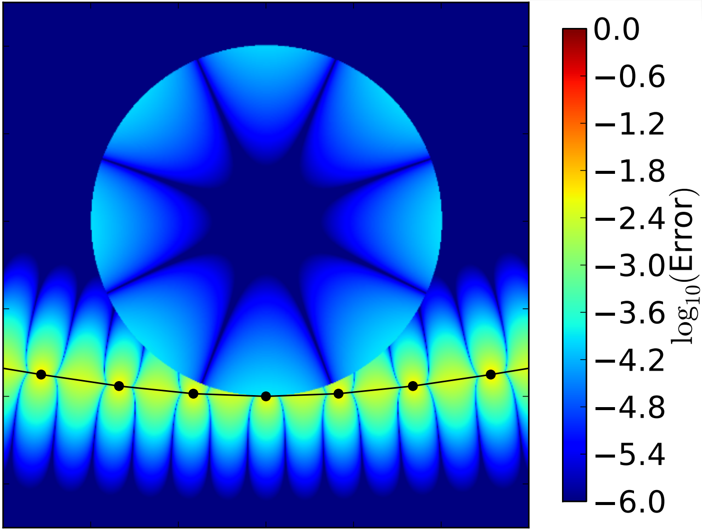
Idea:

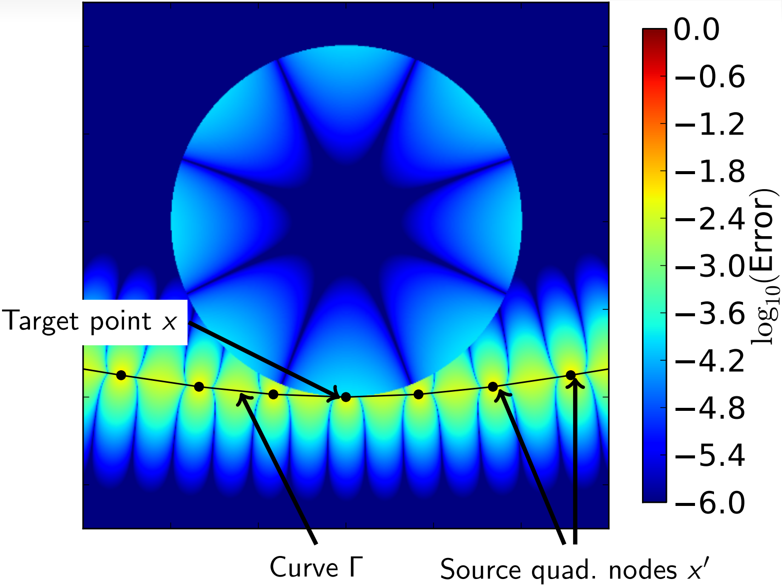
- Pick a point off surface: $c := x + \hat{n}r$ in 'accurate' region
- One-sided smooth potential: Field value there is an approximation to one-sided limit (Err$=O(r)$)

But: Can do much better!

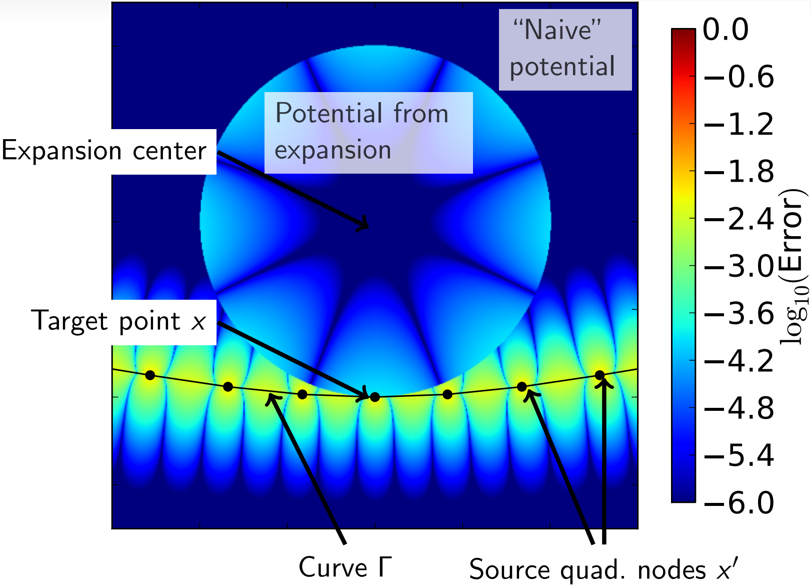


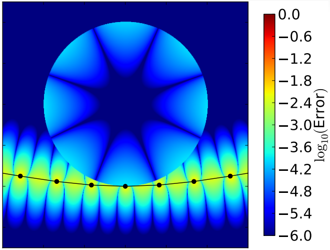
Effectively: replacing layer of charge with discrete sources.


0.0
−0.6
−1.2
−1.8
−2.4
−3.0
−3.6
−4.2
−4.8
−5.4
−6.0
$\log_{10}$(Error)

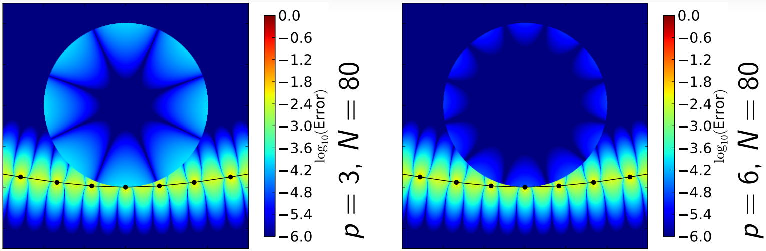


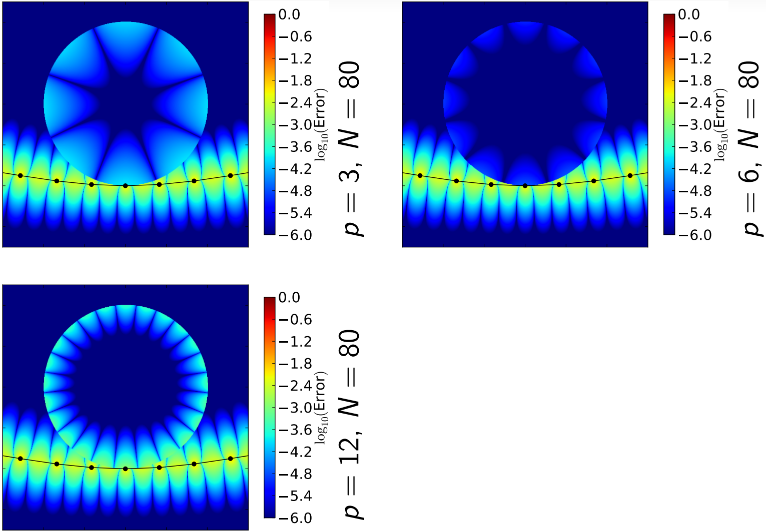
0.0
−0.6
−1.2
−1.8
−2.4
−3.0
−3.6
−4.2
−4.8
−5.4
−6.0
$\log_{10}$(Error)
Target point x
Curve Γ
Source quad. nodes x'

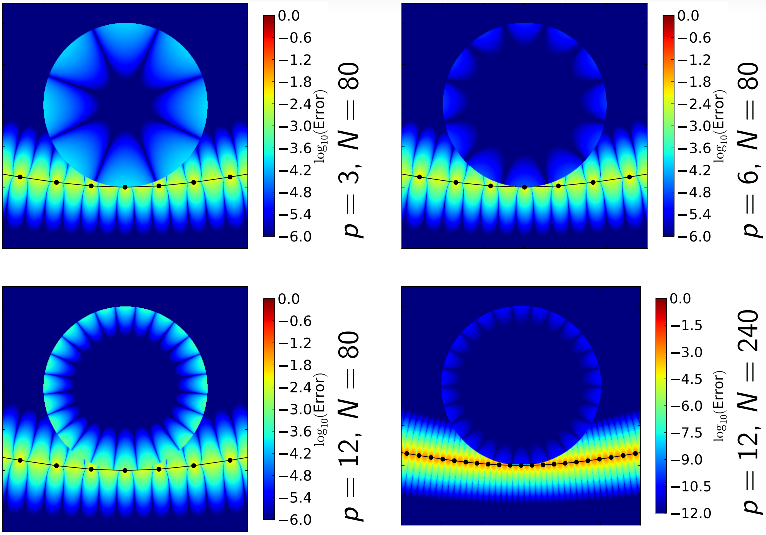

"Naive" potential
Potential from expansion
Expansion center
Target point x
Curve Γ
Source quad. nodes x'
0.0
−0.6
−1.2
−1.8
−2.4
−3.0
−3.6
−4.2
−4.8
−5.4
−6.0
$\log_{10}(\mathrm{Error})$



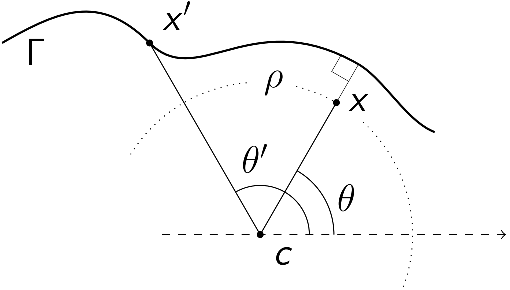
$p = 3,\ N = 80$


0.0
−0.6
−1.2
−1.8
−2.4
−3.0
−3.6
−4.2
−4.8
−5.4
−6.0
log10(Error)
p = 3, N = 80
p = 6, N = 80


0.0
−0.6
−1.2
−1.8
−2.4
−3.0
−3.6
−4.2
−4.8
−5.4
−6.0
$\log_{10}$(Error)
$p = 3,\ N = 80$
$p = 6,\ N = 80$
$p = 12,\ N = 80$

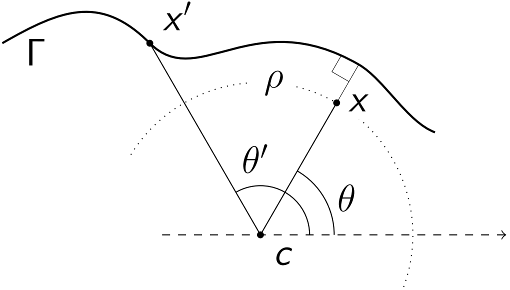

0.0
−0.6
−1.2
−1.8
−2.4
−3.0
−3.6
−4.2
−4.8
−5.4
−6.0
log10(Error)
p = 3, N = 80
p = 6, N = 80
p = 12, N = 80
−1.5
−3.0
−4.5
−6.0
−7.5
−9.0
−10.5
−12.0
p = 12, N = 240

Graf's addition theorem



QBX in formulas: Notation, Basics

Graf's addition theorem

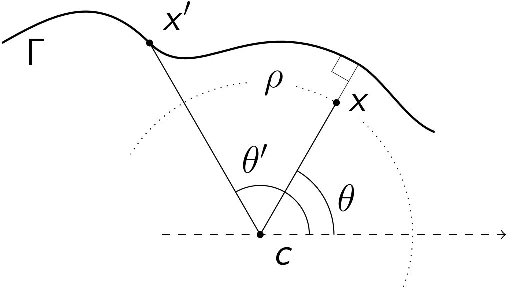


$$H_0^{(1)}(k|x-x'|) = \sum_{l=-\infty}^{\infty} H_l^{(1)}(k|x'-c|)e^{il\theta'} J_l(k|x-c|)e^{-il\theta}$$

QBX

Requires: $|x - c| < |x' - c|$ ("local expansion")

Graf's addition theorem



$$H_0^{(1)}(k|x - x'|) = \sum_{l=-\infty}^{\infty} H_l^{(1)}(k|x' - c|)e^{il\theta'} J_l(k|x - c|)e^{-il\theta}$$

QBX in formulas: Formulation, discretization

Compute layer potential on the disk as

$$S_k\sigma(x) = \sum_{l=-\infty}^{\infty} \alpha_l J_l(k\rho)e^{-il\theta}$$

with

$$\alpha_l = \frac{i}{4}\int_\Gamma H_l^{(1)}(k|x'-c|)e^{il\theta'}\sigma(x')\,dx' \quad (l=-\infty,\ldots,\infty)$$

$S\sigma$ is a smooth function *up to* Γ.

QBX in formulas: Formulation, dis

Now discretize.

Compute layer potential on the disk as

$$S_k\sigma(x) = \sum_{l=-\infty}^{\infty} \alpha_l J_l(k\rho) e^{-il\theta}$$

with

$$\alpha_l = \frac{i}{4} \int_\Gamma H_l^{(1)}(k|x' - c|) e^{il\theta'} \sigma(x')\, dx' \quad (l = -\infty, \ldots, \infty)$$

$S\sigma$ is a smooth function *up to* Γ.

QBX in formulas: Formulation, dis

Now discretize.

Compute layer potential on the disk as

$$S_k\sigma(x) = \sum_{l=-p}^{p} \alpha_l J_l(k\rho) e^{-il\theta}$$

with

$$\alpha_l = \frac{i}{4} \int_\Gamma H_l^{(1)}(k|x' - c|) e^{il\theta'} \sigma(x')\, dx' \quad (l = -\infty, \ldots, \infty)$$

$S\sigma$ is a smooth function *up to* Γ.

QBX in formulas: Formulation, dis

Now discretize.

Compute layer potential on the disk as

$$S_k\sigma(x) = \sum_{l=-p}^{p} \alpha_l J_l(k\rho) e^{-il\theta}$$

with

$$\alpha_l = \frac{i}{4}\, T_N\left(\int_\Gamma H_l^{(1)}(k|x' - c|) e^{il\theta'} \sigma(x')\, dx' \right) \quad (l = -\infty, \dots, \infty)$$

$S\sigma$ is a smooth function *up to* Γ.

QBX in formulas: Formulation, dis

Now discretize.

Compute layer potential on the disk as

$$S_k\sigma(x) = \sum_{l=-p}^{p} \alpha_l J_l(k\rho) e^{-il\theta}$$

with

$$\alpha_l = \frac{i}{4}\, T_N\left(\int_\Gamma H_l^{(1)}(k|x'-c|) e^{il\theta'} \sigma(x')\, dx' \right) \quad (l = -\infty, \ldots, \infty)$$

$S\sigma$ is a smooth function *up to* Γ.

Two limits ($p, N \to \infty$)! Experiment showed: *order matters!*

QBX in formulas: Formulation, dis

Now discretize.

Compute layer potential on the disk as

$$S_k\sigma(x) = \sum_{l=-p}^{p} \alpha_l J_l(k\rho)e^{-il\theta}$$

with

$$\alpha_l = \frac{i}{4}\, T_N\left(\int_\Gamma H_l^{(1)}(k|x'-c|)e^{il\theta'}\sigma(x')\,dx' \right) \quad (l = -\infty, \ldots, \infty)$$

$S\sigma$ is a smooth function *up to* Γ.

Two limits ($p, N \to \infty$)! Experiment showed: *order matters!*

And: failure and repair not actually surprising.

Error result

$$\left| S\sigma(x) - \sum_{l=-p}^{p} \alpha_l^{\mathrm{QBX}} J_l(k|x-c|) e^{-il\theta_{cx}} \right|$$

$$\leq \left(\underbrace{C_{p,\beta}\ r^{p+1} \|\sigma\|_{\mathcal{C}^{p,\beta}(\Gamma)}}_{\text{Truncation error}} + \underbrace{\tilde{C}_{p,2q,\beta} \left(\frac{h}{4r}\right)^{2q} \|\sigma\|_{\mathcal{C}^{2q,\beta}(\Gamma)}}_{\text{Quadrature error}} \right)$$

Proof sketch:

1. First, assume exact calculation of coefficients
2. Estimate tail of expansion
3. Estimate quadrature error in coefficients (derivatives/...)
4. Sum quadrature errors in truncated expansion

[K, Barnett, Greengard, O'Neil '12 (submitted)]

Outline

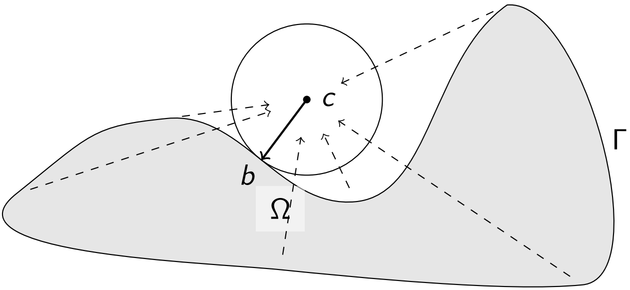
Achieving high order

$$\text{Error} \le \left(C \underbrace{r^{p+1}}_{\text{Truncation error}} + C \underbrace{\left(\frac{h}{r}\right)^q}_{\text{Quadrature error}} \right) \|\sigma\|$$

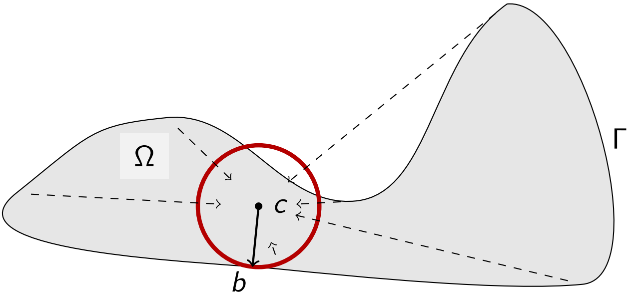
Two approaches:

- *Asymptotically convergent*: $r = \sqrt{h}$
  - ⊕ Error $\to 0$ as $h \to 0$
  - ⊖ Low order: $h^{(p+1)/2}$
- *Convergent with controlled precision*: $r = 5h$
  - ⊖ Error $\not\to 0$ as $h \to 0$
  - ⊕ High order: h^{p+1}
    to controlled precision $\varepsilon := (1/5)^q$

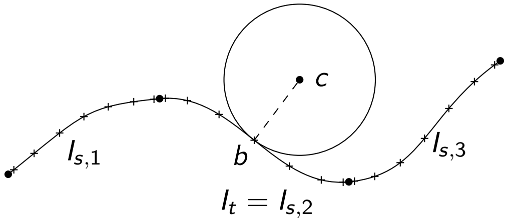
"Global" QBX: Dealing with geometry



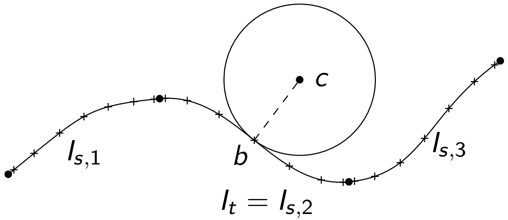
"Global" QBX, part II



"Local" QBX

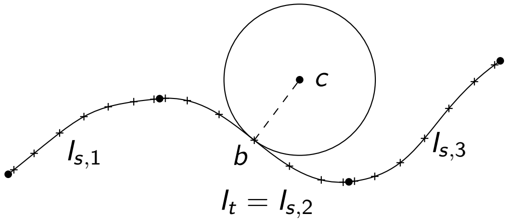


"Local" QBX



Makes geometry processing much simpler

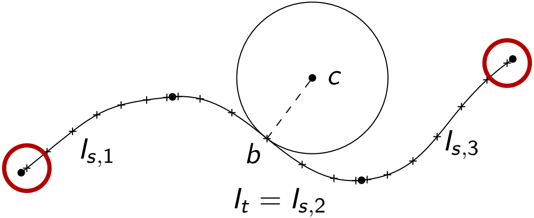
“Local” QBX



kes geometry process-
much simpler

Problem: Expanded field becomes non-smooth (because of end singularities)

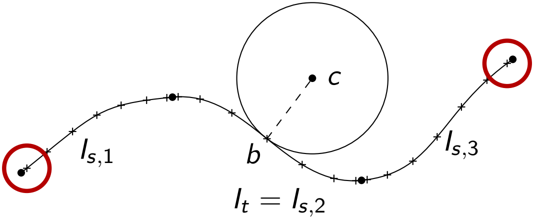
“Local” QBX



...kes geometry process-
...much simpler

Problem: Expanded field becomes non-smooth (because of end singularities)

“Local” QBX



Problem: Expanded field becomes non-smooth (because of end singularities)

Idea: Manage as additional, finite error contribution (using p, $h \propto r$)

kes geometry process-
much simpler

Other layer potentials

Can't just do single-layer potentials:

$$\alpha_l^D = \frac{i}{4} \int_\Gamma \frac{\partial}{\partial \hat{n}_{x'}} H_l^{(1)}(k|x' - c|) e^{il\theta'} \mu(x')\, dx'.$$

Even easier for target derivatives (S' et al.):

Take derivative of local expansion.

Analysis says: Will lose an order.

Other layer potentials

Can't just do single-layer potentials:

$$\alpha_l^D = \frac{i}{4} \int_\Gamma \frac{\partial}{\partial \hat{n}_{x'}} H_l^{(1)}(k|x' - c|) e^{il\theta'} \mu(x')\, dx'.$$

Even easier for target derivatives (S' et al.):

Take derivative of local expansion.

Analysis says: Will lose an order.

Slight issue: QBX computes one-sided limits.

Fortunately: Jump relations are known–e.g.

$$(PV) D^* \mu(x)|_\Gamma = \lim_{x^\pm \to x} D\mu(x^\pm) \mp \frac{1}{2}\mu(x).$$

Other layer potentials

Can't just do single-layer potentials:

$$\alpha_l^D = \frac{i}{4} \int_\Gamma \frac{\partial}{\partial \hat{n}_{x'}} H_l^{(1)}(k|x'-c|) e^{il\theta'} \mu(x')\, dx'.$$

Even easier for target derivatives (S' et al.):

Take derivative of local expansion.

Analysis says: Will lose an order.

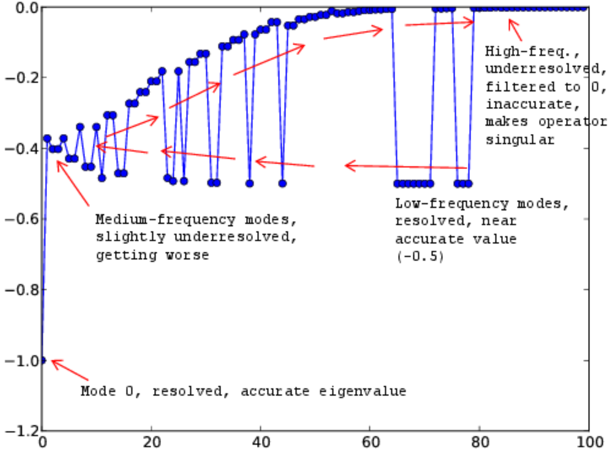
Slight issue: QBX computes one-sided limits.

Fortunately: Jump relations are known–e.g.

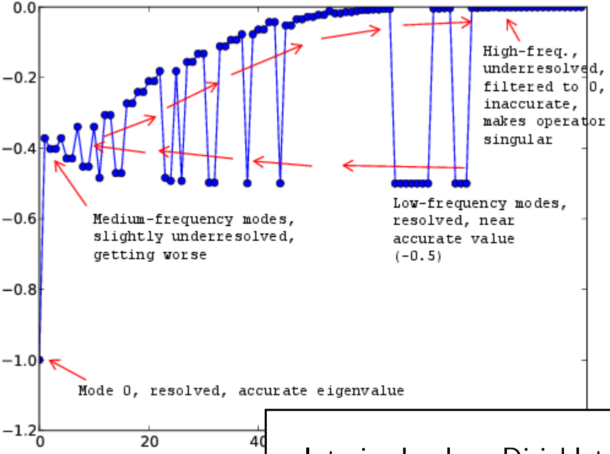
$$(PV)D^*\mu(x)|_\Gamma = \lim D\mu(x^\pm) \mp \frac{1}{\;}\mu(x)$$

Alternative: Two-sided average

Spectral behavior



Spectral behavior



Interior Laplace Dirichlet problem would try to invert this operator.

Spectral behavior, part II

- QBX *wants* to approximate a compact operator–let it:

$$D\mu(x) = \frac{1}{2}\left(\lim_{x^+\to x} D\mu(x^+) + \lim_{x^-\to x} D\mu(x^-)\right).$$

  Simply use two QBX applications.

- *Predictably benign spectral behavior* at high frequencies.

  Important for iterative solvers (e.g. GMRES)

  Not many competing schemes have that!

Outline

Singular quadrature

Fast Algorithms

Integral equations + computers

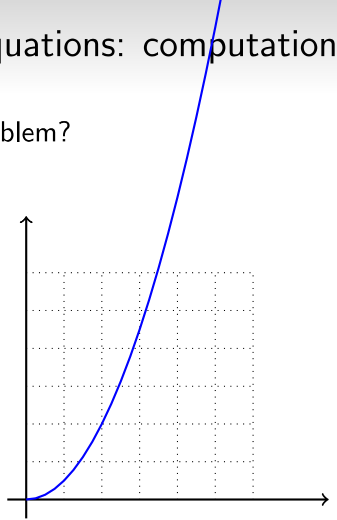
$$O(n^2)$$

Integral equations: computational expense

Why is $O(n^2)$ a problem?

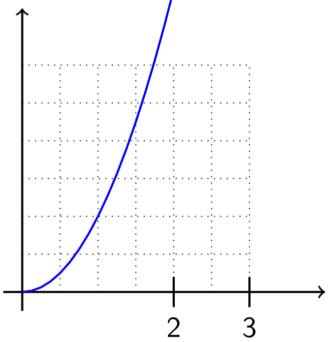
Integral equations: computational expense

Why is $O(n^2)$ a problem?



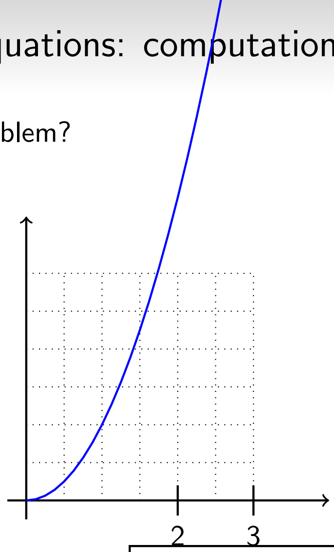
Integral equations: computational expense

Why is $O(n^2)$ a problem?



Integral equations: computational expense

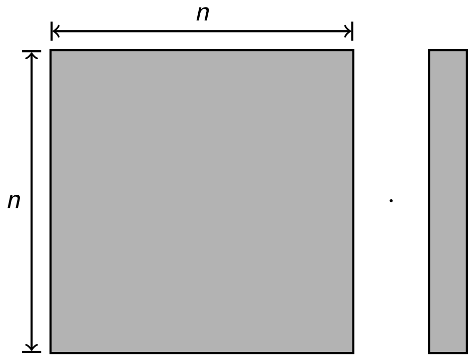
Why is $O(n^2)$ a problem?



Why is $O(n^2)$ *storage complexity* a problem specifically?

Integral equations: computational expense

$O(n^2)$ not in principle incorrect:
Natural complexity of a dense mat-vec.

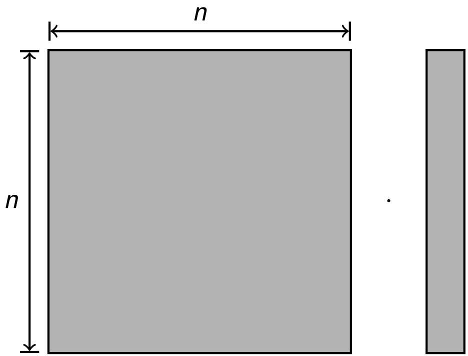


Or, to be more precise:

$$O(\langle \#\text{sources} \rangle \cdot \langle \#\text{targets} \rangle)$$

Integral equations: computational expense

$O(n^2)$ not in principle incorrect:
Natural complexity of a dense mat-vec.

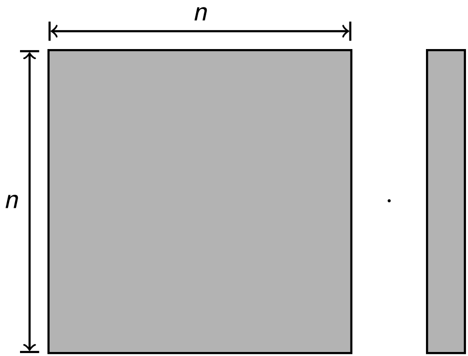


Or, to be more precise:

$$O(\langle\#\text{sources}\rangle \cdot \langle\#\text{targets}\rangle) = O(\langle\#\text{rows}\rangle \cdot \langle\#\text{columns}\rangle)$$

Integral equations: computational expense

$O(n^2)$ not in principle incorrect:
Natural complexity of a dense mat-vec.



Or, to be more precise:

$O(\langle\#\text{sources}\rangle \cdot \langle\#\text{targets}$

Special dense matrices with faster mat-vecs?

Faster dense mat-vecs

If $A = uv^T$, then

$$Ax = \underbrace{\left(uv^T\right)}_{O(n^2)} x$$

Faster dense mat-vecs

If $A = uv^T$, then

$$Ax = \underbrace{(uv^T)}_{O(n^2)} x = \underbrace{u \underbrace{(v^T x)}_{O(n)}}_{O(n)}$$

Faster dense mat-vecs

If $A = uv^T$, then

$$Ax = \underbrace{(uv^T)}_{O(n^2)}x = \underbrace{u\underbrace{(v^Tx)}_{O(n)}}_{O(n)}$$

If $A = u_1v_1^T + u_2v_2^T + \cdot + u_kv_k^T$, then

$$A = ?$$

Faster dense mat-vecs

If $A = uv^T$, then

$$Ax = \underbrace{(uv^T)}_{O(n^2)} x = \underbrace{u \underbrace{(v^T x)}_{O(n)}}_{O(n)}$$

If $A = u_1 v_1^T + u_2 v_2^T + \cdot + u_k v_k^T$, then

$$A = ?$$

Computational cost?

Faster dense mat-vecs

If $A = uv^T$, then

$$Ax = \underbrace{(uv^T)}_{O(n^2)} x = \underbrace{u \underbrace{(v^T x)}_{O(n)}}_{O(n)}$$

If $A = u_1 v_1^T + u_2 v_2^T + \cdot + u_k v_k^T$, then

$$A = ?$$

Computational cost?

Relation to 1D fast algorithm (HW 4)?

Faster dense mat-vecs

If $A = uv^T$, then

$$Ax = \underbrace{(uv^T)}_{O(n^2)} x = \underbrace{u \underbrace{(v^T x)}_{O(n)}}_{O(n)}$$

If $A = u_1 v_1^T + u_2 v_2^T + \cdot + u_k v$

Computational cost?

Relation to 1D fast algorithm (HW 4)?

Relation to separation-of-variables?

Faster dense mat-vecs

If $A = uv^T$, then

$$Ax = \underbrace{(uv^T)}_{O(} x = u \underbrace{(v^T x)}$$

If $A = u_1 v_1^T + u_2 v_2^T + \cdot + u_k v$

Computational cost?

Relation to 1D fast algorithm (HW 4)?

Relation to separation-of-variables?

Can any matrix be written like this?

Faster dense mat-vecs

If $A = uv^T$, then

$$Ax = \underbrace{(uv$$

$O($

If $A = u_1 v_1^T + u_2 v_2^T + \cdot + u_k v$

Computational cost?

Relation to 1D fast algorithm (HW 4)?

Relation to separation-of-variables?

Can any matrix be written like this?

Demo

Faster dense mat-vecs

If $A = uv^T$, then

$$Ax = \underbrace{(uv}_{O(}$$

If $A = u_1 v_1^T + u_2 v_2^T + \cdot + u_k v$

Computational cost?

Relation to 1D fast algorithm (HW 4)?

Relation to separation-of-variables?

Can any matrix be written like this?

Demo

Define numerical rank

Questions?

?