A butterfly algorithm

for the geometric nonuniform FFT

John Strain
Mathematics Department
UC Berkeley
February 2013
FAST FOURIER TRANSFORMS

Standard pointwise FFT evaluates

\[ \hat{f}(k) = \sum_{j=1}^{n} e^{2\pi i k j / n} f_j \quad |k| \leq n/2 \quad O(n^{d+1}) \rightarrow O(n^d \log n) \]

with rigid algebraic recursion for equidistant points

Multidimensional pointwise nonuniform NUFFT evaluates

\[ \hat{f}(t_k) = \sum_{j=1}^{N} e^{i T_k s_j} f_j \quad 1 \leq k \leq N \quad O(N^2) \rightarrow O(N \log N) \]

via low-rank expansion and butterfly recursion

Geometric GNUFFT adds dimensional recursion for

\[ \langle g_k \chi_{T_k}, \hat{f} \rangle = \int_{T_k} g_k(t) \sum_{j=1}^{N} \int_{S_j} e^{i T s} f_j(s) \, ds \, dt \]

with \( N \) polynomials \( g_k, f_j \) on simplices \( T_k, S_j \) in \( R^D \)
OUTLINE

Low-rank expansions separate variables
– enable fast local interactions

Butterfly algorithm propagates information between scales
– simultaneously merge sources and focus targets

New dimensional recursion simplifies matrix elements $\hat{f}_{kj}$
– exactly evaluated by fast low-rank expansion
– gives direct algorithms as well as fast algorithms
LOW-RANK EXPANSION

Complex exponential Taylor series

\[ e^z = \sum_{\alpha=0}^{m} \frac{z^\alpha}{\alpha!} + E_m, \quad |E_m| \leq \left( \frac{|z|e}{m} \right)^m \]

Approximates \( D \)-dimensional clustered nonuniform FFT

\[ \hat{f}(t_k) = \sum_{j=1}^{N} e^{it_k^T s_j} f_j = \sum_{|\alpha| \leq m} t_k^\alpha \left( \frac{i^\alpha}{\alpha!} \sum_{j=1}^{N} f_j s_j^\alpha \right) + E_m = \sum_{|\alpha| \leq m} C_\alpha t_k^\alpha + E_m \]

\[ |E_m| \leq FD \left( \frac{Re}{m} \right)^m \quad F = \sum_j |f_j| \quad |t_k^T s_j| \leq R \]

Fast algorithm for clustered interactions:
- form \( O(m^D) \) moments \( C_\alpha \) of \( N \) sources \( s_j \)
- evaluate \( O(m^D) \)-term series \( \hat{f}(t_k) \) at \( N \) targets \( t_k \)
- total cost \( O(N m^D) = O(N \log \epsilon) \) for accuracy \( \epsilon \)
- assuming all sources and targets have \( |t_k^T s_j| \leq R = O(1) \)
LOCALIZE

Usually $|t_k^T s_j| = O(N)$ is not bounded by a constant $R$

Instead $|s_j| = O(1)$ and $|t_k| = O(N)$ or vice versa so $R = O(N)$

Shift to centers $\tau$ and $\sigma$ of intervals $T$ and $S$

$$e^{i t_k^T s_j} = e^{i \tau T \sigma} e^{i (t_k - \tau)^T \sigma} e^{i (t_k - \tau)^T (s_j - \sigma)} e^{i \tau T (s_j - \sigma)}$$

$$= e^{i \tau T \sigma} e^{i (t_k - \tau)^T \sigma} \sum_{|\alpha| \leq m} \frac{i^\alpha}{\alpha!} (t_k - \tau)^\alpha (s_j - \sigma)^\alpha e^{i \tau T (s_j - \sigma)} + E_m$$

Accurate in Heisenberg pairs $(T, S)$ where

$$(t_k \in T, s_j \in S) \quad \rightarrow \quad |(t_k - \tau)^T (s_j - \sigma)| \leq R \quad \rightarrow \quad |E_m| \leq \epsilon$$
BUTTERFLY ALGORITHM

Sort $N$ sources into $O(N)$ cells

Build $N$ partial expansions, each converging at all $N$ targets

Repeatedly split and merge shifted expansions
   – split each target expansion into $2^D$ adjacent children
   – merge $2^D$ adjacent source children into parent expansion
   – until . . .

Evaluate 1 total expansion at targets in each target cell
SHIFT SOURCE MOMENTS

For targets in cell \( T \) near \( \tau \) and sources in cell \( S \) near \( \sigma \)

\[
\sum_{s_j \in S} e^{it_k^T s_j f_j} = e^{i(t_k - \tau)^T \sigma} \sum_{\alpha} (t_k - \tau)^\alpha C_\alpha(\sigma, \tau)
\]

\[
C_\alpha(\sigma, \tau) = e^{i\tau^T \sigma} \frac{i^\alpha}{\alpha!} \sum_{s_j \in S} (s_j - \sigma)^\alpha e^{i\tau^T (s_j - \sigma)} f_j
\]

Exponential expansion shifts \( \tau \) to target child cell centers \( \{\xi\} \)

\[
C_\alpha(\sigma, \xi) = e^{i(\xi - \tau)^T \sigma} \sum_\beta \binom{\beta + \alpha}{\beta} (\xi - \tau)^\beta C_{\beta + \alpha}(\sigma, \tau)
\]

Binomial theorem shifts \( \{\sigma\} \) to source parent cell center \( \rho \)

\[
C_\alpha(\rho, \xi) = \sum_\sigma \sum_{\beta \leq \alpha} \frac{i^{\alpha - \beta}}{(\alpha - \beta)!} (\sigma - \rho)^{\alpha - \beta} C_\beta(\sigma, \xi)
\]

Step \((S, T) \rightarrow \) (parent of \( S \), children of \( T \)) preserves \( R \)
**D-DIMENSIONAL POINTWISE NUFFT**

0. Organize source and target points
   – into $D$-dimensional $L$-level quadtrees
   – with $2^{-L}R_SR_T \leq R$ so $D(Re/m)^m \leq \epsilon$

1. Build coefficients $C_\alpha(\sigma_L, \tau_0)$
   – for leaf source cells $\sigma_L$ and root target cells $\tau_0$

2. For $l = 1 \ldots L$
   Recursively shift and merge coefficients to
   – each child $\tau_l$ of target cell $\tau_{l-1}$, yielding $C_\alpha(\sigma_{L-l+1}, \tau_l)$
   – parent $\sigma_{L-l}$ of each source cell $\sigma_{L-l+1}$, summing to $C_\alpha(\sigma_{L-l}, \tau_l)$

3. Evaluate expansion with coefficients $C_\alpha(\sigma_0, \tau_L)$
   – for root source cells $\sigma_0$ and leaf target cells $\tau_L$
POINTWISE COMPUTATIONAL KERNEL

One computational kernel

\[ T_\alpha \leftarrow T_\alpha + \frac{i^\alpha}{\alpha!} (s - \sigma)^\alpha e^{i(t-\tau)T(s-\sigma)} \sum_{|\beta| \leq n_T} (t - \tau)^\beta S_\beta \quad |\alpha| \leq n_S \]

does all

– direct evaluation with \( n_S = n_T = 0 \)
– coefficient building with \( n_S > n_T = 0 \)
– expansion evaluation with \( n_T > n_S = 0 \)

Key observation: either \( n_S \) or \( n_T \) is zero

Generalize sums over points \( s \) or \( t \) to integrals over \( s \) or \( t \)
GEOMETRIC SOURCES AND TARGETS

Points $\rightarrow$ sources, targets and densities with geometry

Points $s_j, t_k \rightarrow d$-dimensional simplices $S_j, T_k$ in $R^D$ — points, line segments, triangles, tetrahedra, . . .

Densities $f_j \rightarrow$ polynomials $f_j(s), g_k(t)$ on simplices

Matrix elements $e^{it^T s_j} \rightarrow$ integrals

$$\hat{f}_{kj} = \int_{T_k} g_k(t) \int_{S_j} e^{it^T s} f_j(s) \, ds \, dt$$

Fourier transform $\rightarrow$ sum of integrals

$$\hat{f}(k) = \sum_{j=1}^{N} \hat{f}_{kj} = \int_{T_k} g_k(t) \sum_{j=1}^{N} \int_{S_j} e^{it^T s} f_j(s) \, ds \, dt$$
GNUFFT INITIALIZATION

0. Organize source and target simplices
   – into $D$-dimensional $L$-level quadtrees
   – where $2^{-L}R_SR_T \leq R$ and $D(Re/m)^m \leq \epsilon$

Approximate inclusion is enough

But some simplices are left behind in non-leaf cells
GNUFFT COEFFICIENTS

1. Build geometric integrated coefficients in expansion

\[
\sum_{S_j \subset \sigma} \int_{S_j} e^{i t^T s} f_j(s) \, ds = e^{i (t-\tau)^T \sigma} \sum_{\alpha} (t - \tau)^\alpha C_\alpha(\sigma, \tau)
\]

\[
C_\alpha(\sigma, \tau) = e^{i \tau^T \sigma} \frac{i^\alpha}{\alpha!} \sum_{S_j \subset S} \int_{S_j} (s - \sigma)^\alpha e^{i \tau^T (s-\sigma)} f_j(s) \, ds
\]

Oscillatory integrands challenge quadrature schemes

Use dimensional recursion (later) to obtain exact coefficients
GNUFFT BUTTERFLY RECURSION

Almost identical to pointwise NUFFT!

2. For $l = 1 \ldots L$

   a. Recursively shift and merge coefficients to
      – each child $\tau_l$ of target cell $\tau_{l-1}$, yielding $C_\alpha(\sigma_{L-l+1}, \tau_l)$
      – parent $\sigma_{L-l}$ of each source cell $\sigma_{L-l+1}$, yielding $C_\alpha(\sigma_{L-l}, \tau_l)$

   b. Add source simplices left behind on level $L - l$
3. Integrate expansion \((\sigma_0, \tau_L)\) over target simplices – in each level-\(L\) target cell \(\tau_L\)

Dual to coefficient initialization

Use same dimensional recursion to obtain exact integrals
DIMENSIONAL RECURSION

On $d$-dimensional simplex $S$ carrying polynomial $p$

$$F(t, d, S, p, \alpha, \sigma) = \int_S e^{it^T s} (s - \sigma)^\alpha p(s) \, ds$$

Gauss theorem integrates by parts parallel to $S$

$$\int_S q^T \nabla f(s) \, ds = \int_{\partial S} q^T n_f(\sigma) \, d\sigma$$

Average over parallel target components $t_S \parallel S$ to get

$$F(t, d, S, p, \alpha, \sigma) = -\frac{i}{\|t_S\|^2} \left( F(t_S, d, S, t_S^T \nabla p, \alpha, \sigma) \right. \right.$$ 

$$\left. - \sum_{j=1}^{D} \alpha_j F(t_S, d, S, p, \alpha - e_j, \sigma) \right.$$ 

$$\left. - \sum_{f=0}^{d} t_S^T n_f F(t_S, d - 1, \partial_f S, p, \alpha, \sigma) \right)$$
CONCLUSIONS

Geometric extension of butterfly algorithm
– exact Fourier transform of simplicial polynomials
– arbitrary dimension and codimension
– $O(N \log N \log \epsilon)$ work with accuracy $\epsilon$

Simple speedups:
– Tabulate or approximate shift/merge operators
– Optimized compressed dimensional recursion
– Taylor expansion $\rightarrow$ Chebyshev approximation

Taylor expansion accurate on disk in complex plane
– GNUFFT evaluates geometric Laplace and Gauss transforms with single parametrized code