What's the Frequency, Kenneth?: Sublinear Fourier Sampling Off the Grid

with application to bearing estimation

Petros Boufounos (MERL), Volkan Cevher (EPFL),
Anna Gilbert (UMich), **Yi Li** (UMich), Martin Strauss (UMich)

Intuitive problem description

Input: signal = linear combination of k ``frequencies''

Sample signal at roughly k positions in time

Output: k frequencies + coefficients in time comparable to # samples

Discrete setting: k frequencies in some finite group, usually $\mathbb{Z}_N$

History

Cooley-Tukey[1960s]: Fast Fourier Transform

Sublinear Fourier algorithms

- Kushilevitz-Mansour[1993] (Boolean cube; poly(k, $\log N$))
- Mansour[1995] ($\mathbb{Z}_p$, p prime; poly(k, $\log N$))
- Gilbert-Guha-Indyk-Muthukrishnan-Strauss[2002]
- Gilbert-Muthukrishnan-Strauss[2005] ($k \log k \log^c N$)
- Iwen[2010] (deterministic, $k^2 \log^c N$)
- Akavia[2010] (deterministic)
- Hassenieh-Indyk-Katabi-Price[2012] ($k \log \frac{N}{k} \log N$)

More precisely,

Randomized algorithm: succeed with constant probability over choice of samples

Hassenieh-Indyk-Katabi-Price[2012]

Return k coeffs c_λ and frequencies λ such that

$$\tilde{x} = \frac{1}{\sqrt{N}} \sum_{\lambda} c_\lambda e^{2\pi i \lambda / N}$$

and

$$\|x - \tilde{x}\|_2 \leq (1+\epsilon)\|x - x_k\|_2$$

in time $\frac{k}{\epsilon} \log \frac{N}{k} \log N$

Great success but…

Sparse Fourier sampling algorithms tremendously successful but address a certain discrete model problem

Very specific specialized set-up

Not such a good approximation for analog (real) world

Discrete approximations can degrade the sparsity for sparse analog signals

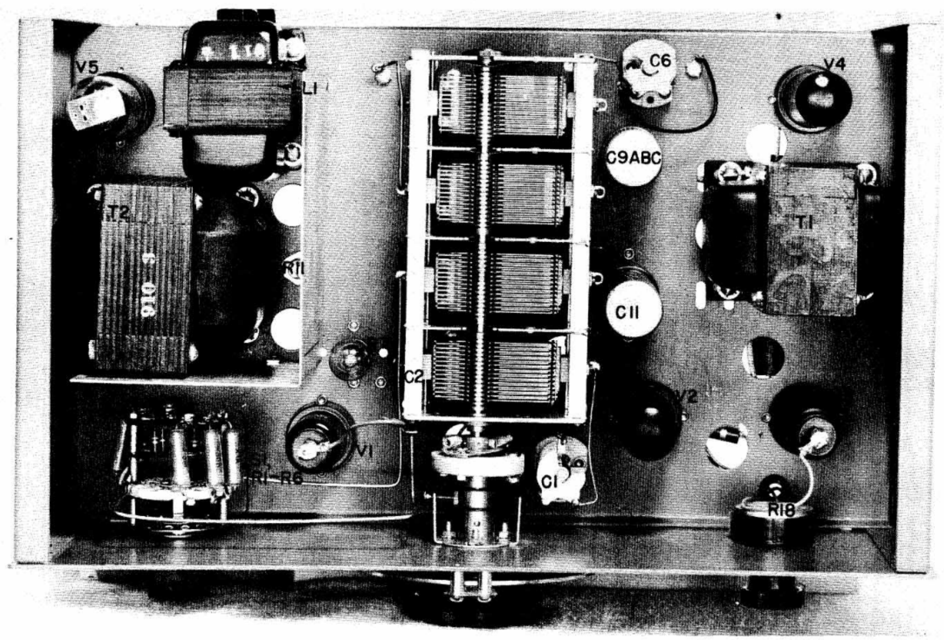
Examples of ℝ world signals

AM/FM radio signals

Musical instruments

Doppler radar

Analog signal generator

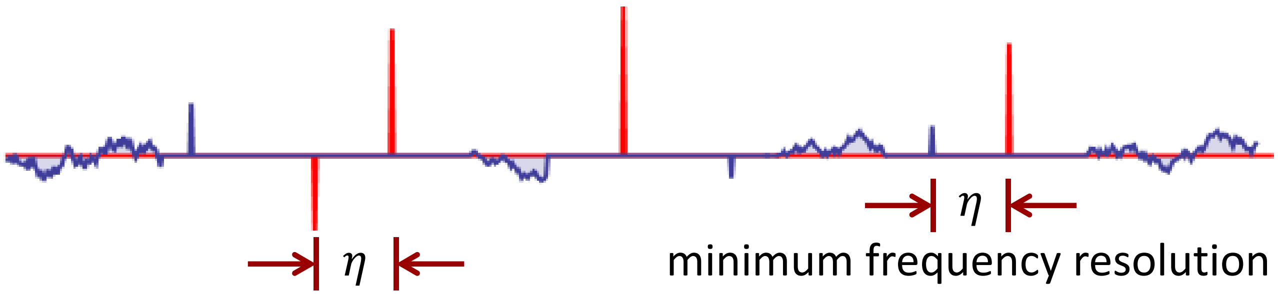


Model 200A. Top View
Cover Removed

New model

Input signal = exponential polynomial + noise

frequencies are contained in $[-\pi, -\eta] \cup [\eta, \pi]$

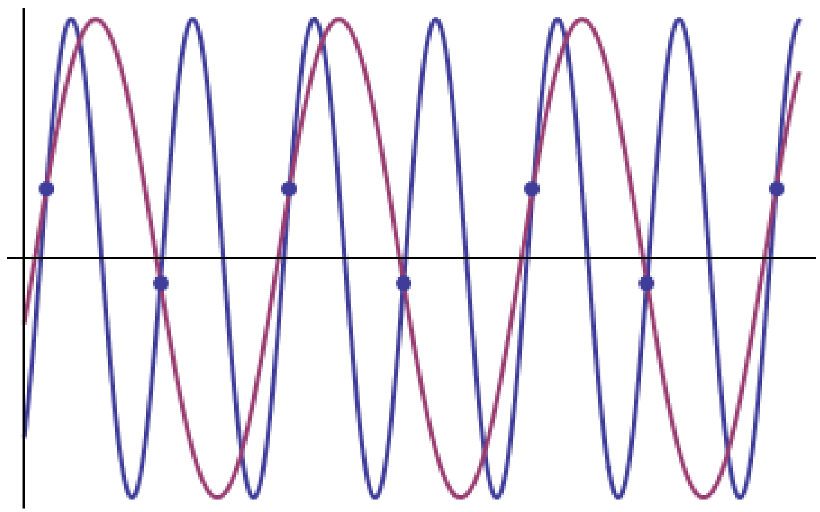


$$f(t) = \sum_{j=1}^{k} a_j \mathrm{e}^{i\omega_j t} + \int_{I_\nu} \nu(\omega) \mathrm{e}^{i\omega t} d\mu$$

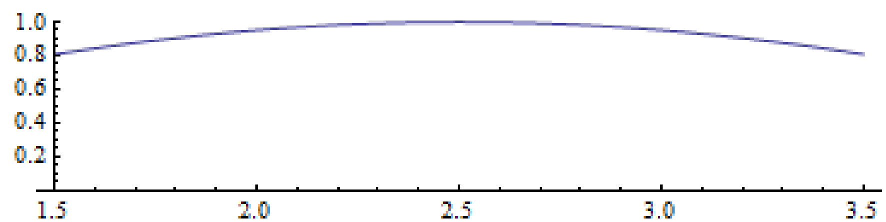
Find (a_j, ω_j) s.t. coefficients are significant:

$$|a_j| \geq \frac{1}{k} \int_{I_\nu} |\nu(\omega)| d\mu$$

Nyquist-Shannon Sampling



Sample interval has to be small to distinguish two high frequencies



Sample duration should be large to distinguish a low frequency from 0

$$N \simeq \frac{1}{\eta} \cdot bandwidth \simeq \frac{1}{\eta}$$

Main result

Theorem: there is a distribution on points (in time) s.t. w.h.p. for each input signal, return list $\{(a'_j, \omega'_j)\}$

For each sign. coefficient , $|\omega_j - \omega'_j| \leq \eta/k$ and $|a'_j - a_j| \leq \frac{\|\nu\|_1}{k}$

samples, running time =

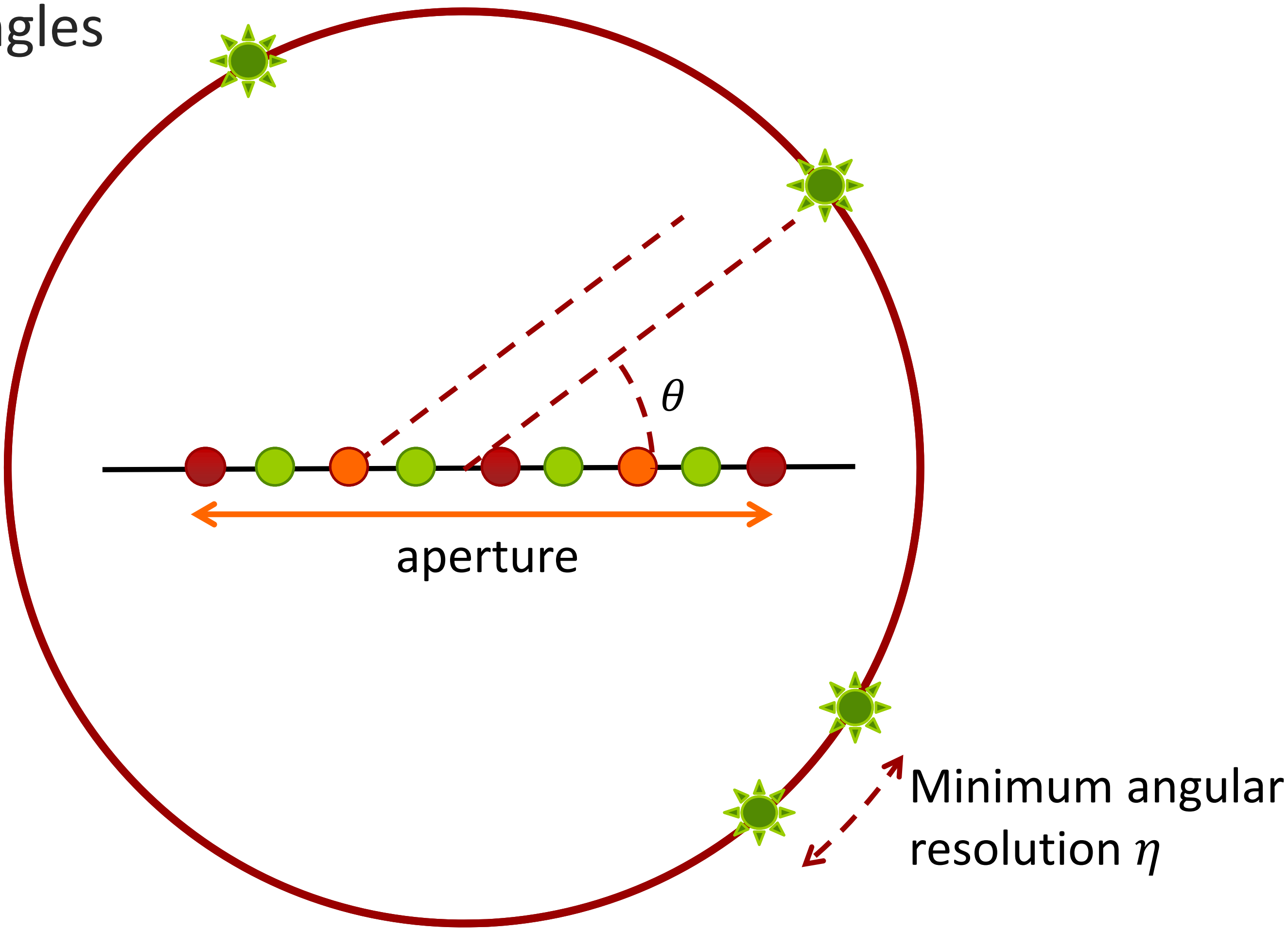
$$O(k \log k \log(1/\eta)(\log k + \log(\|a\|_1/\|\nu\|_1)))$$

Sample duration/extent =

$$O(k/\eta(\log k + \log(\|a\|_1/\|\nu\|_1)))$$

Application: bearing of sources

Determine angles of sources transmitting sound



Application: bearing of sources

Receivers on the x-axis: $\omega_i = \omega \cdot \cos\theta_i$

Find sources with angles in $[\frac{\pi}{8}, \frac{3\pi}{8}]$

- For $\{\theta_i\}$ in this range, $\{\cos\theta_i\}$ has a minimum separation of $\Theta(\eta)$

Rotate the receiver array

Ambiguities

Algorithm

Identify

Isolate frequencies by hashing => multiply samples by filter weights

Read off bits (up to desired resolution) by dilation + hashing

Generate list of candidate frequencies

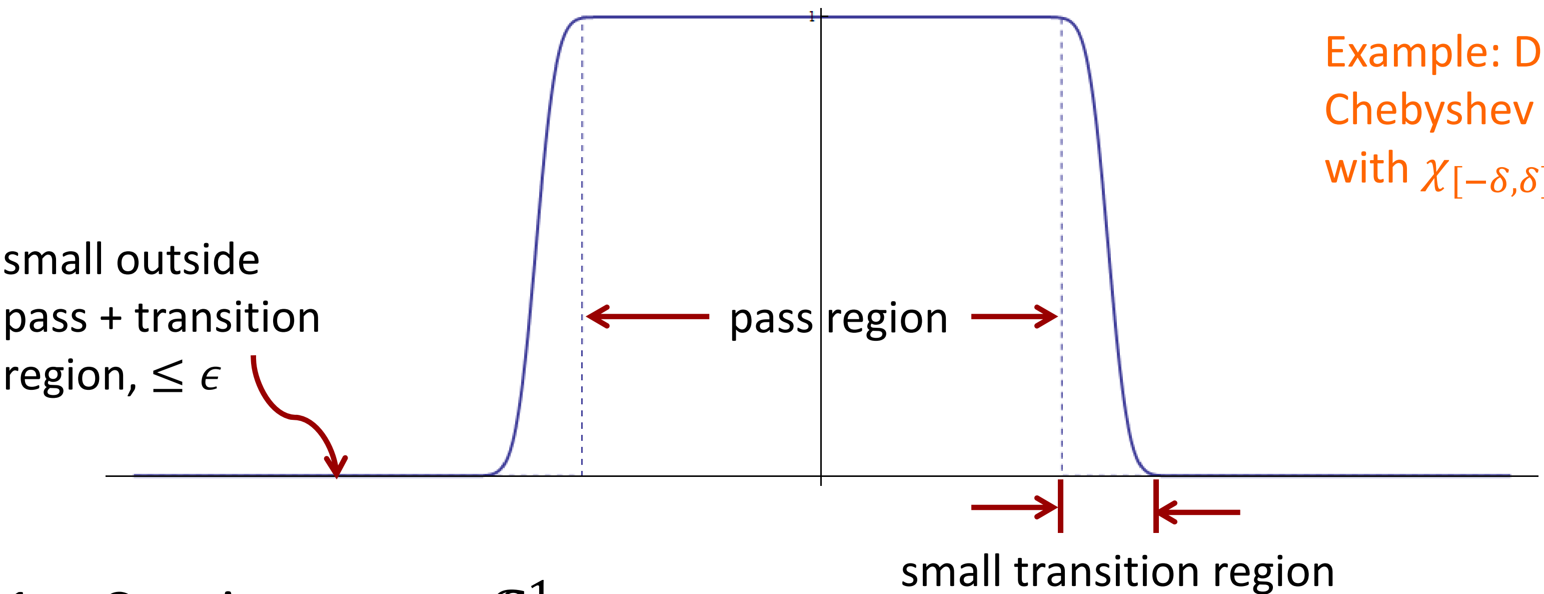
Estimate

Median of values in hashed buckets for specific freqs. in list

1. Need extremely good filter for hashing
2. Non-iterative

Similar to [HIKP 2012a], but with simple bit-testing

A good filter/hash



Example: Dolph-Chebyshev convolved with $\chi_{[-\delta,\delta]}$

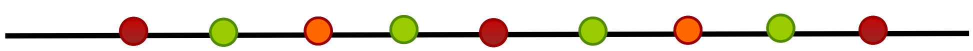
1. Continuous on $\mathbb{S}^1$
2. Fourier transform has finite support on $\mathbb{Z}$
3. Approximates $[-\frac{\pi}{k}, \frac{\pi}{k}]$ well for parameter m

$O\left(k \log \frac{1}{\epsilon}\right)$

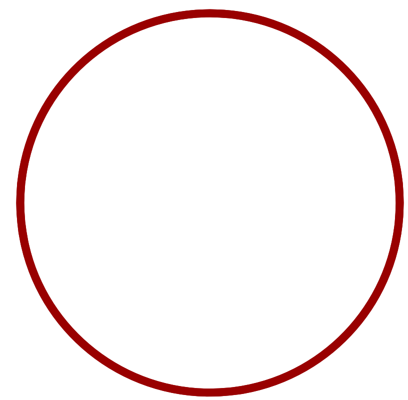
Groups and dual groups

Sample at equidistant discrete points $\leftrightarrow$ Frequencies in $\mathbb{S}^1$

Dilation and translation



$\mathbb{Z}$

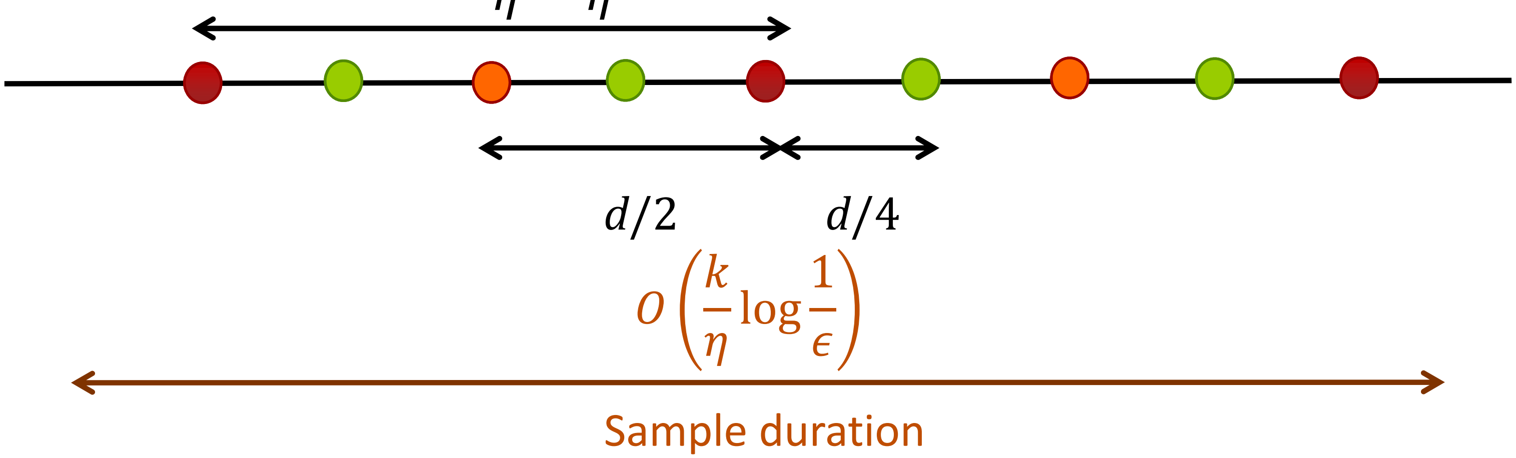


$\mathbb{S}^1$

Distribution = random uniform spacing + bit testers

Random uniform spacing

$$d \in [\frac{2\pi}{\eta}, \frac{4\pi}{\eta}]$$



Total # samples = $O(k \log\frac{1}{\epsilon} \cdot \log\frac{1}{\eta} \cdot \log k)$ $\qquad \epsilon = \frac{\|\nu\|_1}{k\|a\|_1}$

kernel's spectrum size

bits

repetitions

Whither iterative algorithm(s)?

[GMS 2005]: lousy filter but same # buckets in each iteration, # iterations depends on dynamic range of signal, improve est. each iteration

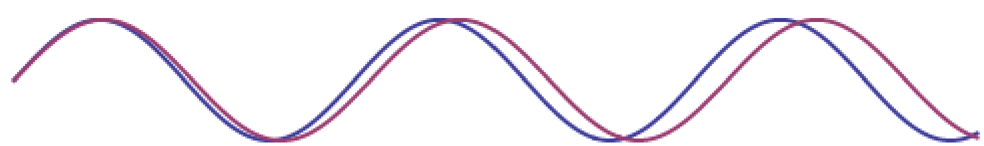
[GLPS 2010, HIKP 2012b]: iterative, # buckets decreases in each iteration

~~Wider bu… …olution~~

Would need k * longer duration

Can't subtract recovered frequencies (easily)

$$\sum_t \left|\mathrm{e}^{i\omega t} - \mathrm{e}^{i\omega' t}\right|$$



Open problems

Lower bounds

- Sample duration (aperture size)
- Number of samples

Iterative vs. non-iterative algorithm

Simple discretization

Error Metric(s)

Thank you!