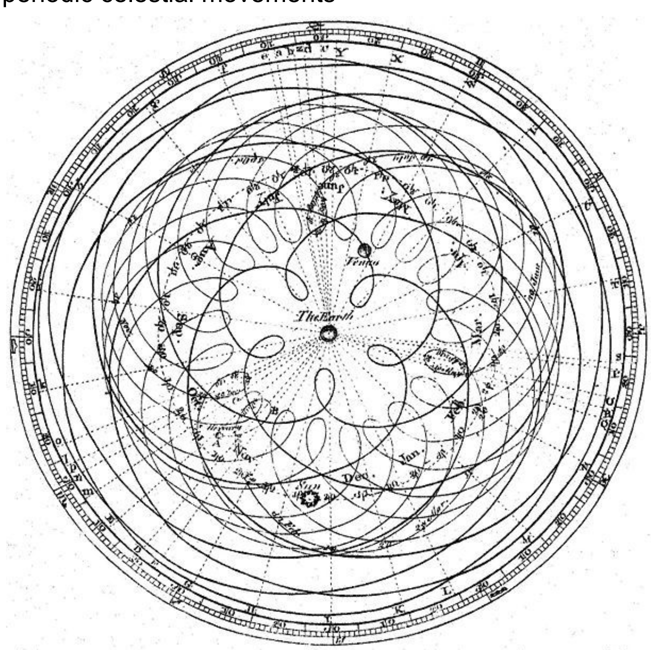
fast fourier transforms

colin shaw
02.25.2016

1. **introduction**
    a. welcome to computer science club
    b. thank you to revunit for sponsoring
    c. what is it about
        i. practice presenting
        ii. demonstrating competencies
        iii. promoting the community
    d. why present
        i. awesome feeling
        ii. forces you to prepare and be better
    e. mention that tyler is thinking about a small presentation
    f. general overview of presentation
        i. history
        ii. theory
        iii. code

2. **history**
    a. ancient greeks
        i. periodic celestial movements



b. gauss (1805)
    i. basic formulation
    ii. early notion of the fft
c. fourier (1822)
    i. studying heat distribution
    ii. assertion about convergence to any continuous function
d. dirichlet, gibbs
    i. convergence
e. nyquist
    i. sampling
f. cooley-tukey (1965)
    i. rediscovered fft

3. preliminary concepts

a. infinite support
    i. periodic functions
        1. $F(\omega) = \int_{-\infty}^{\infty} f(x)\, sin(\omega x)\, dx$
        2. $F(\omega) = \int_{-\infty}^{\infty} f(x)\, cos(\omega x)\, dx$
        3. functional commonality parameterized by ω

    ii. orthonormal basis
        1. $\int_{-\infty}^{\infty} sin(\omega x)\, sin(\psi\, x)\, dx = 0 \quad \forall\ \omega \neq \psi \quad \in R$
        2. $\int_{-\infty}^{\infty} sin(\omega x)\, sin(\psi\, x)\, dx = k \quad \forall\ \omega = \psi \quad \in R$
        3. $\int_{-\infty}^{\infty} cos(\omega x)\, cos(\psi\, x)\, dx = 0 \quad \forall\ \omega \neq \psi \quad \in R$
        4. $\int_{-\infty}^{\infty} cos(\omega x)\, cos(\psi\, x)\, dx = k \quad \forall\ \omega = \psi \quad \in R$
        5. $\int_{-\infty}^{\infty} sin(\omega x)\, cos(\psi\, x)\, dx = 0 \quad \forall\ \omega, \psi \quad \in R$

    iii. complex is more compact
        1. $$\begin{aligned} F(\omega) &= \int_{-\infty}^{\infty} f(x)\, e^{-2\pi i \omega x}\, dx \\ &= \int_{-\infty}^{\infty} f(x)\, [cos(-2\pi i \omega x) - i\, sin(-2\pi i \omega x)]\, dx \\ &= \int_{-\infty}^{\infty} f(x)\, cos(2\pi i \omega x)\, dx + i \int_{-\infty}^{\infty} f(x)\, sin(2\pi i \omega x)\, dx \end{aligned}$$

iv. inverse
   1. $f(x) = \int_{-\infty}^{\infty} F(\omega)\, e^{2\pi i \omega x}\, d\omega$
   2. exponential complex conjugate is the inverse

b. windowed support
   i. why
      1. finite problems domain
      2. simpler integration (no limits)
   ii. form
      1. $F(\omega) = \int_{-T/2}^{T/2} f(x)\, e^{-2\pi i \omega x/T}\, dx$
      2. $f(x) = \int_{-T/2}^{T/2} F(\omega)\, e^{2\pi i \omega x/T}\, da$
   iii. conditions
      1. $[-\frac{T}{2}, \frac{T}{2}]$ spans integer number of periods
      2. largest span is one cycle in T
      3. can be transformed to $[0, T]$

c. discrete case
   i. why
      1. discrete or sampled input data
         a. time series data
         b. pixelated images
      2. algebraic aspect of problem
   ii. basic form
      1. $X_\omega = \sum_{n=0}^{N-1} x_n\, e^{-2\pi i \omega n/N}$
      2. $x_n = \frac{1}{N} \sum_{\omega=0}^{N-1} X_\omega\, e^{2\pi i \omega n/N}$
      3. observations
         a. the $\frac{1}{N}$ is sometimes distributed as $\frac{1}{\sqrt{N}}$ for forward and reverse
         b. this is how the dft is most commonly computed
         c. the transform of a single element is itself (either direction)
   iii. roots of unit
      1. $\omega = e^{2\pi i/N}$ is a complex number on the unit circle
      2. expressions are powers of ω
      3. ω is just a function of N (constant for choice of problem)

iv. rephrasing as an algebra problem

1. consider matrix equation $X = Mx$
2. where M is a vandermonde matrix of coefficients

$$\begin{bmatrix} \omega_N^{0\cdot 0} & \omega_N^{0\cdot 1} & \ldots & \omega_N^{0\cdot (N-1)} \\ \omega_N^{1\cdot 0} & \omega_N^{1\cdot 1} & \ldots & \omega_N^{1\cdot (N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_N^{(N-1)\cdot 0} & \omega_N^{(N-1)\cdot 1} & \ldots & \omega_N^{(N-1)\cdot (N-1)} \end{bmatrix}$$

3. compute and reduce (mod N) the powers

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & \omega & \omega^2 & \omega^3 & \omega^4 & \omega^5 & \omega^6 & \omega^7 \\ 1 & \omega^2 & \omega^4 & \omega^6 & \omega^8 & \omega^{10} & \omega^{12} & \omega^{14} \\ 1 & \omega^3 & \omega^6 & \omega^9 & \omega^{12} & \omega^{15} & \omega^{18} & \omega^{21} \\ 1 & \omega^4 & \omega^8 & \omega^{12} & \omega^{16} & \omega^{20} & \omega^{24} & \omega^{28} \\ 1 & \omega^5 & \omega^{10} & \omega^{15} & \omega^{20} & \omega^{25} & \omega^{30} & \omega^{35} \\ 1 & \omega^6 & \omega^{12} & \omega^{18} & \omega^{24} & \omega^{30} & \omega^{36} & \omega^{42} \\ 1 & \omega^7 & \omega^{14} & \omega^{21} & \omega^{28} & \omega^{35} & \omega^{42} & \omega^{49} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & \omega & \omega^2 & \omega^3 & \omega^4 & \omega^5 & \omega^6 & \omega^7 \\ 1 & \omega^2 & \omega^4 & \omega^6 & 1 & \omega^2 & \omega^4 & \omega^6 \\ 1 & \omega^3 & \omega^6 & \omega & \omega^4 & \omega^7 & \omega^2 & \omega^5 \\ 1 & \omega^4 & 1 & \omega^4 & 1 & \omega^4 & 1 & \omega^4 \\ 1 & \omega^5 & \omega^2 & \omega^7 & \omega^4 & \omega & \omega^6 & \omega^3 \\ 1 & \omega^6 & \omega^4 & \omega^2 & 1 & \omega^6 & \omega^4 & \omega^2 \\ 1 & \omega^7 & \omega^6 & \omega^5 & \omega^4 & \omega^3 & \omega^2 & \omega \end{bmatrix}$$

4. observations
   a. the inverse is negative powers of ω (recompute powers mod N)
   b. the fft is a means of taking advantage of this symmetry
   c. factorization into product of sparse matrices

v. decimation algorithm

1. consider a two-point fft (we know a single-point fft is just itself)

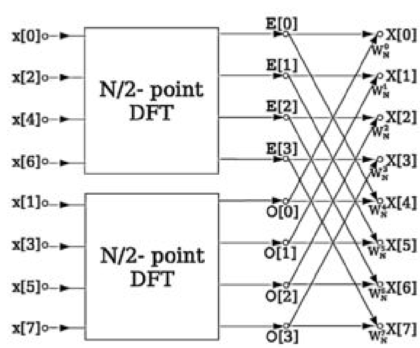
$$X_0 = x_0 + x_1$$
$$X_1 = x_0 - x_1$$

2. which generalizes to the sub-transform

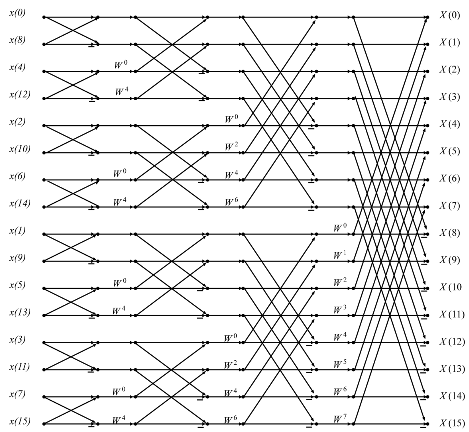
$$X_e = x_e + x_o\,\omega^{j/k}$$
$$X_o = x_e - x_o\,\omega^{j/k}$$

   for even tems x_e and X_e, odd terms x_o and X_o,
   and properly selected sub-transform indices j and k

3. a more general diagram for an n-point dft



4. expanded depth



5. looks like a textbook case of recursion!

4. look at code

a. waveform
b. omega (memoization)
c. dft
   i. procedural implementation
   ii. $O(n^2)$ time complexity
   iii. $O(n)$ space complexity
d. fft
   i. recursive implementation (naive but simple)
   ii. $O(n\ log\ n)$ time complexity
   iii. $O(n\ log\ n)$ space complexity (naive recursive)
   iv. $O(n)$ space complexity (iterative in-place)

5. observations

a. relates to prime decomposition
b. powers of two by far most common
c. dft is worst-case computation
d. pre-computed prime decomposition possible
e. iterate rather than recurse to save memory

6. run examples

7. questions