Math. What good is it?

Modern society cannot exist without sophisticated mathematics

Chris Lomont
April 6, 2011, EMU
Chris Lomont

• Research Engineer at Cybernet Systems
  – Ann Arbor
  – Uses math from arithmetic level through PhD coursework every day
    • also computer science, physics
  – Hired originally to do quantum computing
    • where all three tie together very intricately
  – Works on algorithms, security, robotics, image processing, edge of technology ideas
Introduction

• Stereotypical question from beginning math students: “when will I ever need this?”

• Take a common (yet sophisticated) piece of modern technology: iPhone
  – Analyze places math is required to make it
  – Math subjects colored in bold red

• Can do the same thing for all parts of modern life
  – technology, economics, agriculture, medicine, politics
Sound to MP3s

• What is sound?
  – Mechanics of sound

• How does the human ear work?
  – Human range 20 - 20,000 Hz
  – Decibels – measured energy
Recording Sound

• Need 20-20,000 Hz samples.
  – Nyquist-Shannon theorem:
    • Need to sample at twice the rate
    • Information Theory (founded by Shannon)
  – Thus need at least 40,000 samples per second
    • 44.1 kHz for CDs, gives 22,050 top frequency.
How to store samples?

- **Mechanism measures “back and forth”**
  - **Bit:** Binary digit stores a 0 or 1 in a base 2 number.
  - Digitized to 16 bits, represent $2^{16} = 65536$ values from -32768 to 32767, called **Pulse Code Modulation (PCM)**, invented 1937.
  - Sometimes sampled at 20 bits.
- **Playback is reverse mechanism.**
- **Now – how many bits to store a 4 minute song in stereo?**
  - $44,100 \times 2 \times 16 \times 4 \times 60 = 338,688,000$ bits (arithmetic)
What does it mean?

• 8 bits in a byte, so need $338688000/8=42$ million bytes, called 42 megabytes.

• CD holds 700 megabytes, which means $\frac{700}{42} \approx 17$ four minute songs, or a little more than 1 hour of audio.
Compression

• At 42 MB per song, on your 16GB iPhone you could only get $\frac{16000000000}{42000000}=380$ songs. What gives?

  – Model human ear
    • Some sounds cannot hear
    • Some sounds easier to hear than others
    • Two sounds at once, often cannot hear softer one.
    • Ears ringing from previous sound, can ignore later ones

  – Throw most audio out

• How do we get frequency information from audio?
Fourier Transforms

• Decompose a wave into frequency bands:

\[ f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos(nx) + b_n \sin(nx) \]

  – Note the \( a_n \) and \( b_n \) tell how much at frequency “n”. **Fourier Analysis, Trigonometry, Analysis**

• Allows removing pieces we don’t want.

• **Compressed to 128 kb/s gives 22 fold improvement.**
Computer Science
(Math Aside #1)

• Lambda calculus: evolved out of Leibniz and Hilbert questions (1930’s) : what is computable?
• Places fundamental limits on “knowledge”
  – Gödel Incompleteness Theorems (Logic)
  – Halting Problem (Turing, 1936).
• Everything base 2, all “zeroes and ones”
  – Binary Digit “bit” – everything is 0’s and 1’s
• **Algorithms** (Discrete Math)
  – P=NP worth $1,000,000, finding a way to do NP problems in P time worth billions of $ in applications.
  – Knuth books – created TeX to format his computer books, 3168 pages so far.
Floating Point Numbers

- Computers work with approximations to real numbers, usually called floating-point numbers
- Format: sign bit, exponent, mantissa
- Value is \((-1)^{\text{sign}} \times 2^{e-bias} \times 1.\text{mantissa}\)

- Not quite like real numbers
Floating-point error

• Many common rules fail:
  – Associativity can fail: $a + (b + c) \neq (a + b) + c$
  – TI-83 says:

<table>
<thead>
<tr>
<th>Expression</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>$1 + 10^{-12} - 1$</td>
<td>$10^{-12}$</td>
</tr>
<tr>
<td>$1 + 10^{-13} - 1$</td>
<td>$0$</td>
</tr>
<tr>
<td>$\sqrt{10^{-40}} = 10^{-20}$</td>
<td></td>
</tr>
<tr>
<td>$1 - 1 + 10^{-13}$</td>
<td>$10^{-13}$</td>
</tr>
</tbody>
</table>

• Must understand how stored and what guarantees are given by your platform to make programs that don’t fail in weird ways.
  – Numerical Analysis
Discrete Math

• Study of mathematical structures that are fundamentally discrete, rather than continuous.

• Includes or overlaps set theory, logic, combinatorics, graph theory, probability, number theory, discrete calculus, geometry, topology, game theory.

• Uses
  – Algorithms, programming languages, crypto, networking
Storage

• Memory sizes limited by physics and cost – what are they?
• Quantum mechanics (and special relativity) underlies all solid state electronics and modern technology
  – Hilbert Spaces, Operators (Functional Analysis)
• Running up against physical limits
• Stores numbers, ‘0’ and ‘1’, bytes.
iPhone

- Comes in 8GB, 16GB, 32GB
- 4.5 x 2.3 x 0.37 inches
- 960 x 640 pixels
- 137 grams
- Audio: MP3 (8-320 Kbps), other formats
- Video: H.264 and others
- Camera: 5 million pixels, JPEG, 30 fps video
- Global Positioning System (GPS)
- 3-axis gyroscope, accelerometer, digital compass
- Battery 7-14 hours
- 802.11b/g/n, Bluetooth
- Oh yeah, it is also a phone
Pictures and Video

• We did audio, now how about imagery?

• Same approach
  1. Determine how sound **light** works,
  2. Determine how **ear** **eye** works,
  3. Determine how to capture **audio images**,
  4. Determine how to store,
  5. Determine how to playback,
  6. Determine how to compress for efficiency.
Light

- Visible part of electromagnetic spectrum
  - Part of radio waves, microwaves, x-rays, all same.

- Maxwell’s laws (all vector calculus)

- Behaves as wave
  - For our purposes 😊
  - Wavelength 400-700 nm
  - White light is mix of many colors
  - Newton pressed on his eye
  - Prism

\[
\nabla \cdot \vec{E} = \frac{\rho}{\varepsilon_0} \\
\n\nabla \cdot \vec{B} = 0 \\
\n\n\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t} \\
\n\n\n\nabla \times \vec{B} = \mu_0 \vec{J} + \frac{\partial \vec{E}}{\partial t}
\]
Optics

- To focus images, must bend light
- Lenses, mirrors
  - Newton wrote Opticks, 1704
  - Geometry, calculus
- Aberrations
  - Coma, curvature, axis, more
  - Can prove no perfect lens possible
  - Often fixed in software (more math!)
Human Eye

- Rod – can detect a single photon
- Cone – three kinds, peak at

<table>
<thead>
<tr>
<th>Cone type</th>
<th>Name</th>
<th>Range</th>
<th>Peak wavelength</th>
</tr>
</thead>
<tbody>
<tr>
<td>Short</td>
<td>β</td>
<td>400–500 nm</td>
<td>420–440 nm</td>
</tr>
<tr>
<td>Medium</td>
<td>γ</td>
<td>450–630 nm</td>
<td>534–555 nm</td>
</tr>
<tr>
<td>Long</td>
<td>ρ</td>
<td>500–700 nm</td>
<td>564–580 nm</td>
</tr>
</tbody>
</table>

- Lens focuses
Camera

• Same idea: capture light in color grid
• How to capture:
  – Bayer Filter (patent, 1976)
• Final samples interpolated
  – gives each pixel Red, Green, Blue component
  – Lanczos Filter (filtering theory)

\[
L(x) = \begin{cases} 
\frac{sinc(x)}{sinc\left(\frac{x}{a}\right)} & -a < x < a, x \neq 0 \\
1 & x = 0 \\
0 & \text{otherwise}
\end{cases}
\]
Camera

• CCD converts light at each sensor point to a value in 0-255 (8 bits).
• Demosaicing converts Bayer pattern to grid of red, green, blue values
Photo Size

• 8 bits each for red, green, blue:
  – Gives $2^8 = 256$ levels each of red, green, blue
  – Total $8 \times 3 = 24$ bits per pixel
  – $2^{24} = 16,777,216$ possible colors

• Storage for 12 megapixel image = 4000x3000

\[
12,000,000 \text{ pixels} \times \frac{24 \text{ bits}}{\text{pixel}} \times \frac{1 \text{ byte}}{8 \text{ bits}} = 36 \text{MB per image}
\]

• Could only fit 400 pictures on 16GB iPhone.
Display Technology

• iPhone 960x640 “pixel” display.
• Each pixel has red, green, and blue components to match the human eye.
• Think Christmas lights
  – Basics of solid state physics is another talk!
Human eye more sensitive to brightness than color, especially at high frequencies.

Convert RGB to YCrCb

\[
Y' = \frac{R_D' + 0.299 \cdot R_D' + 0.587 \cdot G_D' + 0.114 \cdot B_D'}{255}
\]

\[
C_B = 128 - (0.168736 \cdot R_D') - (0.331264 \cdot G_D') + 0.5 \cdot B_D'
\]

\[
C_R = 128 + (0.5 \cdot R_D') - (0.418688 \cdot G_D') - (0.081312 \cdot B_D')
\]

- Linear Algebra

Split image into 8x8 blocks

Average colors 2x2 blocks (loses data)

Perform Discrete Cosine Transform (DCT)
DCT

• Similar to Fourier Transform (Analysis)

\[ G_{u,v} = \sum_{x=0}^{7} \sum_{y=0}^{7} \alpha(u) \alpha(v) g_{x,y} \cos \left[ \frac{\pi}{8} \left( x + \frac{1}{2} \right) u \right] \cos \left[ \frac{\pi}{8} \left( y + \frac{1}{2} \right) v \right] \]
Video

- HDTV is 1920x1080 pixels, 30 fps ≈ 187MB/s
- Use eye models, add motion information
- Intra frame motion prediction
  - statistics, optimization, signal analysis
Applications

- Games
  - **Physics** for lighting, motion, particle effects, cars
  - Numerical integration (**analysis**, **calculus**)
  - **Differential Equations** for motion, lighting, collisions
  - World all done through **linear algebra**
  - Quaternions for rotations (**group theory**)
  - Probabilistic methods for input (touch, motion)

- Streaming – Quality of Service (QoS) Analysis
Example – Angry Birds

• Basic 2D physics simulation
  – Motion, linear and angular momentum
  – Hindered by discrete time and floating-point math
  – Geometry, calculus, differential equations, linear algebra
Angry Birds

- Collision detection
  - significant numerical challenges to be robust
- 2D physics $x = x_0 + v_0 t + \frac{1}{2} at^2$
  - Linear, angular momentum
  - Numerical integration likely used
  - Basic Euler integration insufficient
    - $x_{i+1} = x_i + v_i \Delta t$
    - $v_{i+1} = v_i + a \Delta t$
  - Runge Kutta Integration RK4 often used
    - Higher order approximation than Euler
  - Verlet integration also used
    - Based on Taylor expansions going forward and backward in time
- Uses **Calculus** and **Differential Equations**
Font Rendering

• Curves defined as polynomials
  – Hinting for small parts
  – Each font is a little program
• Final rasterized to pixel grid
Cubic Bezier

• Four points $P_0, P_1, P_2, P_3$

• Parametric for $t \in [0,1]$ (analysis)
  $B(t) = (1 - t)^3 P_0 + 3(1 - t)^2 t P_1 + 3(1 - t)t^2 P_2 + t^3 P_3$

• De Casteljau Algorithm
  – Numerically stable
  – Subdivision
Probabilistic Methods

(Math Aside #2)

• Markov Models
  – statistics, graph theory, matrices, linear algebra, Mathematical modeling, stochastic modeling

• Hidden Markov Models
  – Allows learning the “hidden” state
Security

• Encryption (cryptology)
  – AES – based on finite fields
  – RSA – based on number theory
  – ECC – Elliptic Curves over finite fields
    \[ y^2 = x^3 + ax + b \]
  – Hits most of abstract algebra and some algebraic geometry, ring theory.
Global Positioning System

• Each sends time, orbital info, system health
  – Per satellite atomic clock, 14 ns accuracy
  – 50 bits per second, each frame 30 seconds
  – Uses CDMA encodings

• Receiver computes distance to each satellite
  – Needs 3 naively, but too much error
  – 4+ enough.

• Needs relativity
  – Differential Geometry
GPS

• Two spheres give circle
• Circle and 3\textsuperscript{rd} sphere give 2 points
• 4\textsuperscript{th} gives which point and allows error correction.
Miscellaneous Features

• Speech Recognition, Language conversion, Touchscreen intent, AutoCorrect
  – Markov Models
  – Bayesian Belief
    • Predictor models
  – Learning

• Sensors
  – Digital compass
  – Accelerometer
  – 3-axis gyro
  – merged using Kalman Filtering to get knowledge
Physical

• Aesthetics
  – Outer shell case
  – Spline surfaces, subdivision, NURBS
    • Surface version of splines used for font outlines
  – **Topology, Analysis, Differential Geometry**

• Materials
  – Chemistry
    • Gorilla glass
    • Battery - lithium chemistry
    • Quantized orbitals – drive all of life
  – Semiconductors

\[ Y_{l}^{m}(\theta, \varphi) = Ne^{im\varphi}P_{l}^{m}(\cos \theta) \]
\[ P_{l}^{m}(x) = \frac{(-1)^{m}}{2^{l}l!} (1 - x^{2})^{\frac{m}{2}} \frac{d^{l+m}}{dx^{l+m}} (x^{2} - 1)^{l} \]
<table>
<thead>
<tr>
<th>Math Topics Employed</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Arithmetic</td>
<td>Differential Equations</td>
</tr>
<tr>
<td>Basic Algebra</td>
<td>Algorithms</td>
</tr>
<tr>
<td>Trigonometry</td>
<td>Number Theory</td>
</tr>
<tr>
<td>Basic and Advanced Probability</td>
<td>Cryptology</td>
</tr>
<tr>
<td>Calculus I,II,III</td>
<td>Abstract Algebra</td>
</tr>
<tr>
<td>Linear Algebra</td>
<td>Ring Theory</td>
</tr>
<tr>
<td>Statistics</td>
<td>Stochastic Modeling</td>
</tr>
<tr>
<td>Discrete Math (Math for CS)</td>
<td>Real and complex analysis</td>
</tr>
<tr>
<td>Functional Analysis (Hilbert)</td>
<td>Numerical Analysis</td>
</tr>
<tr>
<td>Geometry (analytic, 2D, 3D)</td>
<td>Differential Geometry (and relativity)</td>
</tr>
<tr>
<td>Mathematical Modeling</td>
<td>Mathematical Logic</td>
</tr>
<tr>
<td>Fourier Analysis</td>
<td>Group Theory (Symmetric, permutation)</td>
</tr>
<tr>
<td>Information, Filtering Theory</td>
<td>Graph Theory</td>
</tr>
<tr>
<td>Topology</td>
<td>Optimization Theory</td>
</tr>
<tr>
<td>Algebraic Geometry</td>
<td>Set Theory</td>
</tr>
<tr>
<td>Game Theory</td>
<td>Combinatorics</td>
</tr>
</tbody>
</table>
THE END

Questions?
Removed Slides

• Coming next for more discussion 😊
RSA

• Ron Rivest, Adi Shamir, Leonard Adelman (1978)

1. Generate two large primes $p$ and $q$ such that $n = pq$ is large enough (4096 bits). Set $\phi = (p - 1)(q - 1)$

2. Choose $1 < e < \phi$ with $(e, \phi) = 1$.

3. Compute $1 < d < \phi$ with $ed \equiv 1 \pmod{\phi}$.

4. $(n, e)$ is public key, $(n, d)$ is private key

5. To send message $m$, send ciphertext $c \equiv m^e \pmod{n}$

6. To decode, compute

$$c^d \equiv m^{de} \equiv m^{1+k\phi} \equiv m(m^\phi)^k \equiv m(1)^k \equiv m \pmod{n}$$

– Last step works by Euler’s theorem
CDMA

• Each user has different random code – chosen very carefully.
• Exploits **linear algebra** to find orthogonal vectors representing data strings.
• Each code orthogonal to all others.
• Addition of signals is decode-able using clever mathematics
• Utilized broad spectrum for more room
Probabilistic Methods

Position Uncertainty → Worse over time + Combine multiple readings = Increased Accuracy
GPS

• Relativity analysis
  – Need time accuracy at receiver or 20-30 ns
  – 20,000 km orbit
  – move at 14,000 km/hr relative to ground
  – Lose 7 \( \mu s \) a day relative to Earth due to slower tick rates from Earth viewpoint
  – Gain 45 \( \mu s \) a day from curvature of space due to Earth mass slowing down clocks on Earth surface
  – Result 38 \( \mu s \) per day = 38,000 ns, huge error if not corrected

• General relativity – need **differential geometry**
  – 150 years ago **differential geometry** was abstract cutting edge pure mathematics! Now we use it in our toys. (Large parts are still cutting edge mathematics research topics)
• Zero out enough to reach the compression level you want
• Store in zigzag — most important first
• Note the blockiness of over compressed JPEG is an artifact of the 8x8 pixel blocking:
Model of “Hey Jude”

Breakdown of Lyrics to "Hey Jude"

- Na Na Na Na
- Hey Jude
- Other Words
Wireless

• 802.11 b/g/n - TODO
Wireless Phone Technology

• Cell-phone carrier gets 832 frequencies.
  • Two frequencies per call -- a duplex channel
    – typically 395 voice channels per carrier
    – 42 frequencies used for control channels

• Cells in hex grid
  – each has 6 neighbors
  – uses 1/7 of channels per cell

• Gives 56 channels per cell, so **56 users at a time per cell.**
FDMA

• Frequency Division Multiple Access
TDMA

• Time Division Multiple Access
CDMA

• Code Division Multiple Access

Transmitted signal: Data Signal XOR with the Pseudorandom
Font Rasterization
Old Fonts

- Roman capitals
  - Defined during Italian Renaissance
  - Albrecht Dürer, 1525, from four volume series on geometry.
Error Correction

• Problem: how to deal with inevitable errors?
• Bits sometimes get flipped during transmission $0 \leftrightarrow 1$

• Simple idea: repeat each bit three times and take majority vote:

$$0 \rightarrow 000 \quad \text{error!} \quad 001 \quad \text{majority vote} \rightarrow 0$$

• Corrects 1 bit errors, but at cost of tripling data requirement.
Error Correction

• Smarter: take any three bits $abc$ and append $a \oplus b$, $a \oplus c$, $b \oplus c$ where $a, b, c \in \{0,1\}$ and $\oplus$ is addition mod 2
  
  $0 \oplus 0 = 0$
  $0 \oplus 1 = 1$
  $1 \oplus 0 = 1$
  $1 \oplus 1 = 0$

• Then only doubled number of bits (three to six) but can recover any single bit flipping error (you must check).

• Question – how good can errors be fixed? Answer – very good, math is quite deep.
  – Many open problems