# Lecture 4

## Series, Function Approximation
## Chebyshev

# Summary lecture 3

- Integration
  - Open formulas don't use boundary point at end: 1 order lower precision in the stepsize h
  - Higher-order precision is obtained via reweighing the gridpoints: Simpsons rule, Bode rule
  - Romberg integration: use Euler-Maclaurin summation to cancel consecutive even orders in the power series expansion
    - Stepsize is halved in each step for closed formulas
    - Stepsize is tripled for open formulas – since the first point is ½ stepsize away from the boundary
  - Error estimation: from difference in intermediate results
  - Gaussian quadrature: use free choice of abscissa as well
    - Enough grid points are needed in the region where the function peaks
      - For strongly-peaked functions Romberg may be better
    - Periodic functions: Use Fourier-techniques instead

# Summary

- Integration to infinity : coordinate transformation (e.g $y=\tan(x)$, $x=\arctan(y)$)
- "smooth" functions : 10-100 steps (Gaussian quadrature) should suffice
- Multiple dimensions:
  - Try to reduce to lower dimensions
  - Reasonable maximum around 6 dimensions (20-point Gaussian quadrature – 30 million calculations)
  - Monte-Carlo techniques

# Function calculation

- Chapter 5.
  - Calculation of power series
  - acceleration
  - Continued fractions
    - E.g. to calculate pi to a million decimals
  - Recurrence
    - when you need to accelerate calculations involving series, study the book! I will only use the Chebyshev formalism in this course

# Power series

- analytic function: power series expansion
  - update terms (such as fac)
  - For the fastest code, do not use p(i)=c(i)*pow(x,i), or p(4)=c(4)*x*x*x*x.
  - Instead, one can calculate sum and derivative at once:

```
p=c[n];
dp = 0;
i=n;
while ( i>0) {
    i=i-1;
    dp =dp*x+p;
    p = p*x+c[i];
}
```

- convergence of power series (until first pole in complex plane) generally known.
- convergence can be slow (e.g. $\sin(x)=\sum_{n=0}^{\infty}\frac{(-1)^n x^{2n+1}}{(2n+1)!}$ )
- Numerically useless for x>10

# Power series

- analytic function: power series expansion
  - update terms (such as fac)
  - For the fastest code, do not use p(i)=c(i)*pow(x,i), or p(4)=c(4)*x*x*x*x.
  - Instead, one can calculate sum and derivative at once:

```
p=c[n];
dp = 0;
i=n;
while ( i>0) {
        i=i-1;
        dp =dp*x+p;
        p = p*x+c[i];
}
```

- convergence of power series (until first pole in complex plane) generally known.
- convergence can be slow (e.g. $\sin(x)=\sum_{n=0}^{\infty}\frac{(-1)^n x^{2n+1}}{(2n+1)!}$ )
- Numerically useless for x>10

# Accelerating convergence

- *geometric* series: Aitkens delta$^2$-process

$$S'_n = S_{n+1} - \frac{(S_{n+1} - S_n)^2}{S_{n+1} - 2S_n + S_{n-1}}$$

- $S_n$ = partial sum, up to term n
- $S'_n$ = new series, where partial sums are re-used and added with a different weight.
- also possible: n-1 and n+1 with n-p, n+p.
- example: $(½)^n$ – correct result for $S'_1$

# Accelerating convergence

- *Alternating* series: Eulers transformation
  - forward difference operator (widely used)

$$\sum_{s=0}^{\infty}(-1)^{n}u_{s}=u_{0}-u_{1}+u_{2}+..-u_{n-1}+\sum_{s=n}^{\infty}\frac{(-1)^{s}}{2^{s+1}}[\Delta^{s}u_{n}]$$

$$\Delta u_{n}\equiv u_{n+1}-u_{n}\quad ;\quad \Delta^{2}u_{n}\equiv u_{n+2}-2u_{n+1}+u_{n}\quad ;$$

$$\Delta^{3}u_{n}\equiv 1u_{n+3}-3u_{n+2}+3u_{n+1}-1u_{n},\quad \text{etc.}$$

- Eulers transformation converges more rapidly.
- Also possible : Levin transformation (see book)

# Eulers transformation

- Eulers transformation:
  - sometimes even OK for non-convergent series!
  - typically used for asymptotic series
  - Van Wijngaarden method: eulsum in NR2, series.h in NR3
    - numerical algorithm for Eulers transformation
    - calculates itself whether to increase terms before Euler summation or update Euler terms
  - Euler converges rapidly! Sometimes, convert series with only single-sign terms into alternating series just to use Eulers transformation afterwards

# Converting series:

- replace sum $\sum_r v_r \Rightarrow \sum_r (-)^r w_r$

$$w_r = v_r + 2v_{2r} + 4v_{4r} + 8v_{8r} + \ldots.$$

- replaces sum over v by double sum, sum over w which itself contains a sum over v
- $w_r$ converges rapidly, since index grows so fast.
- can only be used if random values of $v_r$ can be easily calculated.
- Eulers transform: special case of general power transformation: $g(z) = \sum_n b_n z^n; \quad known$

$$f(z) = \sum_n c_n b_n z^n; \quad f(z) = \sum_n [\Delta^n c_0] \frac{g^{(n)}}{n!} z^n$$

# Euler Transformation

$$g(z) = \sum_n b_n z^n; \quad known$$

$$f(z) = \sum_n c_n b_n z^n; \quad f(z) = \sum_n [\Delta^n c_0] \frac{g^{(n)}}{n!} z^n$$

- Usual Euler transformation:

$$g(z) = \frac{1}{1+z} = 1 - z + z^2 - z^3 + \ldots$$

- Note, these series techniques are frequently used in Fourier analysis (later this course). Also for Chebyshev.

# Clenshaws recurrence formula

- Make use of recurrence relations; e.g. Legendre polynomials, Bessel, etc.

$$(n+1)P_{n+1}(x) = (2n+1)xP_n(x) - nP_{n-1}(x)$$

$$J_{n+1}(x) = \frac{2n}{x} J_n(x) - J_{n-1}(x)$$

$$\cos(n\theta) = 2\cos(\theta)\cos((n-1)\theta) - \cos(n-2)\theta$$

# recurrence

- recurrence relations; e.g. Legendre polynomials, Bessel, etc.
  - Stability!
  - $y_{n+1} + a y_n + b y_{n-1} = 0$ : 2 solutions, $f_n$ and $g_n$
  - maybe $f_n$ wanted. $g_n$ can be exponentially stable, exp. damped, or exp. growing (e.g. Bessel, growing n)
  - $f_n/g_n$ -> 0 for n->inf : $f_n$ is minimal solution
  - $g_n$ dominant solution
  - minimal solution is unique, dominant not
  - How to test?
    - start with 0 and 1 and 1 and 0
    - evolve 20 terms and see difference.
- (stability is a property of the recurrence relation, not of the function)

# stability

- Alternatively, replace recurrence relation with linear one with constant coefficients

$$y_{n+1} - \frac{2n}{x} y_n + y_{n-1} \Rightarrow y_{n+1} - 2\gamma y_n + y_{n-1}$$

$$Try \quad y_n = a^n$$

$$a^2 - 2\gamma a + 1 = 0 \quad a = \gamma \pm \sqrt{\gamma^2 - 1}$$

$$\text{Stable if } |a| \le 1 \quad \rightarrow \quad \gamma \le 1, \quad n < x$$

# recurrence

- how to proceed if recurrence is non-stable?
- start in other direction with arbitrary seeds. Solution is correct times a normalization constant. E.g in the case of Bessel functions:
  - start with large n (eg $\sqrt{100N}$ ) for 10 significant digits. set
    $$j_n = 1, \quad j_{n-1} = 0$$
  - go down to $J_0$, $J_1$ and normalize, to the calculated value of $J_0(x)$.
  - ( or alternatively e.g. with a normalization rule like $1=J_0+2J_2+2J_4+2J_6+..+2J_n(x)$ )

# Clenshaw´s recurrence formula

- Coefficients x functions that obey recurrence

$$f(\theta)=\sum_k c_k\cos(k\theta);\quad f(x)=\sum_k c_k P_k(x);$$

$$f(x)=\sum_k c_k F_k(x);\quad F_{n+1}(x)=\alpha(n,x)F_n(x)+\beta(n,x)F_{n-1}(x)$$

$$define\quad y_{n+2}=y_{n+1}=0;\quad y_k=\alpha(k,x)y_{k+1}+\beta(k+1,x)y_{k+2}+c_k$$

solve for $c_k$:

$$\begin{aligned}
f(x) &= \ldots + \\
&+[y_4-\alpha(4,x)y_5-\beta(4,x)y_6]F_4(x) \\
&+[y_3-\alpha(3,x)y_4-\beta(3,x)y_5]F_3(x) \\
&+[y_2-\alpha(2,x)y_3-\beta(2,x)y_4]F_2(x) \\
&+[y_1-\alpha(1,x)y_2-\beta(1,x)y_3]F_1(x) \\
&+[c_0+\beta(1,x)y_2-\beta(1,x)y_2]F_0(x)
\end{aligned}$$

# Clenshaw Recurrence

- Terms sum to zero up till $y_2$
- only surviving term:

$$f(x) = \beta(1,x)F_0(x)y_2 + F_1(x)y_1 + F_0(x)c_0$$

- make one pass through $y_k$'s
- apply above formula
- almost always stable, only not when $F_k$ is small for large k and $c_k$ small for small k, when there is delicate cancellation: first terms in above eq. cancel each other: use upwards recurrence (see book). This can be detected:

$$\beta(1,x)F_0(x)y_2 \approx -F_1(x)y_1$$

# Chebyshev Approximation

- Chebyshev polynomials:

$$T_n(x) = \cos(n \arccos(x))$$

$$T_0 = 1$$

$$T_1(x) = x$$

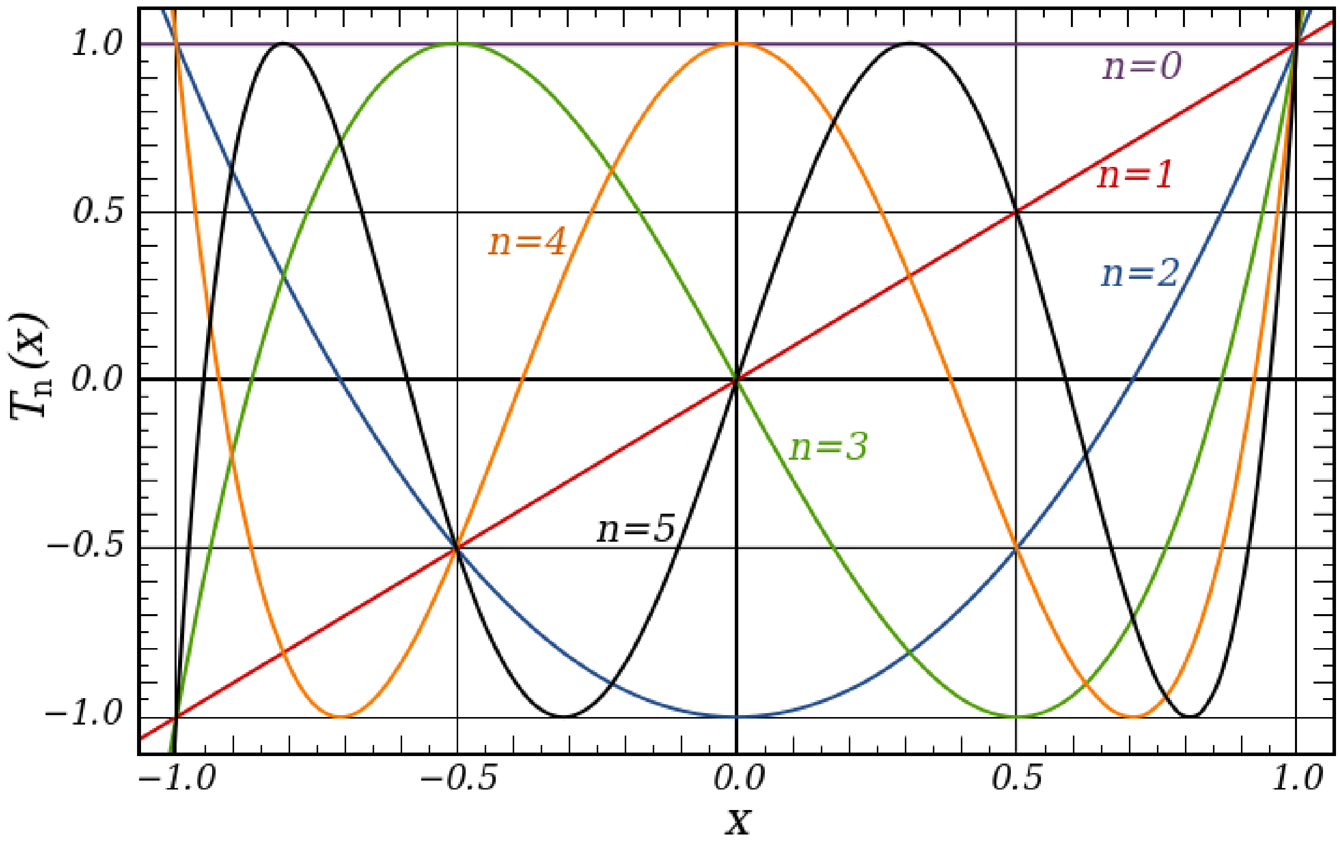
$$T_2(x) = 2x^2 - 1$$

$$T_3(x) = 4x^3 - 3x$$

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$$

- $T_n$ has n zeros, at cos(π(k+1/2)/n)
- $T_n$ has n+1 extrema, located at cos(πk/n)
- All maxima equal +1, all minima –1

Chebyshev Polynomials



# Chebyshev

- orthogonal over weight 1/sqrt(1-x$^2$)

$$\int_{-1}^{1} \frac{T_i(x)T_j(x)}{\sqrt{1-x^2}} = \begin{array}{l} 0 \rightarrow i \neq j \\ \pi/2 \rightarrow i = j \neq 0 \\ \pi \rightarrow i = j = 0 \end{array}$$

- discrete relation for the m zeros $x_k$: (i,j<m)

$$\sum_{k=0}^{m-1} T_i(x_k)T_j(x_k) = \begin{array}{ll} 0 & i \neq j \\ m/2 & i = j \neq 0 \\ m & i = j = 0 \end{array}$$

# Chebyshev approximation

- We can make use of these nice orthogonality relations to make an approximation of an arbitrary function of x in the interval [-1,1] by calculating the coefficients $c_j$ at the N zero's $x_k$ of the N-th Chebyshev polynomial:

$$c_j = \frac{2}{N} \sum_{k=0}^{N-1} f(x_k) T_j(x_k)$$

$$c_j = \frac{2}{N} \sum_{k=0}^{N-1} f\left(\cos\left(\frac{\pi(k+1/2)}{N}\right)\right) \cos\left(\frac{\pi j(k+1/2)}{N}\right)$$

Then, the function is represented exactly at those N values of x and approximated at other x by

$$f(x) \approx \left[\sum_{k=1}^{N-1} c_k T_k(x)\right] - \frac{1}{2} c_0$$

# Chebyshev truncation

$$f(x) \approx \sum_{k=1}^{m} c_k T_k(x) \quad - \frac{1}{2} c_0$$

: difference is smaller than the sum of all neglected $c_k$

- typically, $c_k$ rapidly decreasing (exponentially)
- $c_m T_m$ : rapidly oscillating function bounded between +/- 1
- Chebyshev polynomial is very close to the minimax polynomial; the polynomial of degree m that has the smallest distance to the function for all x.
- change of variables: eg.

$$x \to \infty \quad y = \frac{1}{x} \quad , \quad y = \arctan\left(\frac{\pi x}{2}\right)$$

$$x \in [a,b] \quad : \quad y = \frac{x - \frac{b+a}{2}}{\frac{1}{2}(b-a)}$$

Computational meth

# Chebyshev approximation:

- calculate the Chebyshev coeff's to order N ( N function calls); $N^2$ cosines once.
- store these coefficients
- How to calculate f(x)?
- Clenshaw´s recurrence : 2m sums and multplications

$$d_{m+1} = d_m = 0$$

$$d_j = 2xd_{j+1} - d_{j+2} + c_j \quad for \ j = M-1 \ldots 1$$

$$f(x) = d_0 = xd_1 - d_2 + \frac{1}{2}c_0$$

# Chebyshev:

- Even function: all odd coeffs are zero
- better to use $T_{2n}(x)=T_n(2x^2-1)$ from $\cos(2x)=2\cos^2 x -1$
- call chebev with the argument x replaced by $2x^2-1$
- Odd function : calculate f(x)/x; this will give accurate results close to x=0.
- alternatively, again use $y= 2x^2-1$, but now (since $c_0$ is not used) the last line of the code chebev needs to be changed:
- $f(x)=x[(2y-1)d_1-d_2+c_0]$

# Chebyshev polynomials:

- If you have the Chebyshev coeffs, the coeffs for the derivative and integral of f are given by:

$$C_i = \frac{c_{i-1} + c_{i+1}}{2i} \quad (i > 0) \rightarrow \quad \int f(x)$$

$$c'_{i-1} = c'_{i+1} + 2ic_i \quad (i = m-1....1) \rightarrow \frac{df}{dx}$$

- integral: arbitrary $C_0$
- derivative: no info on m+1

# Chebyshev example from book, 5.13

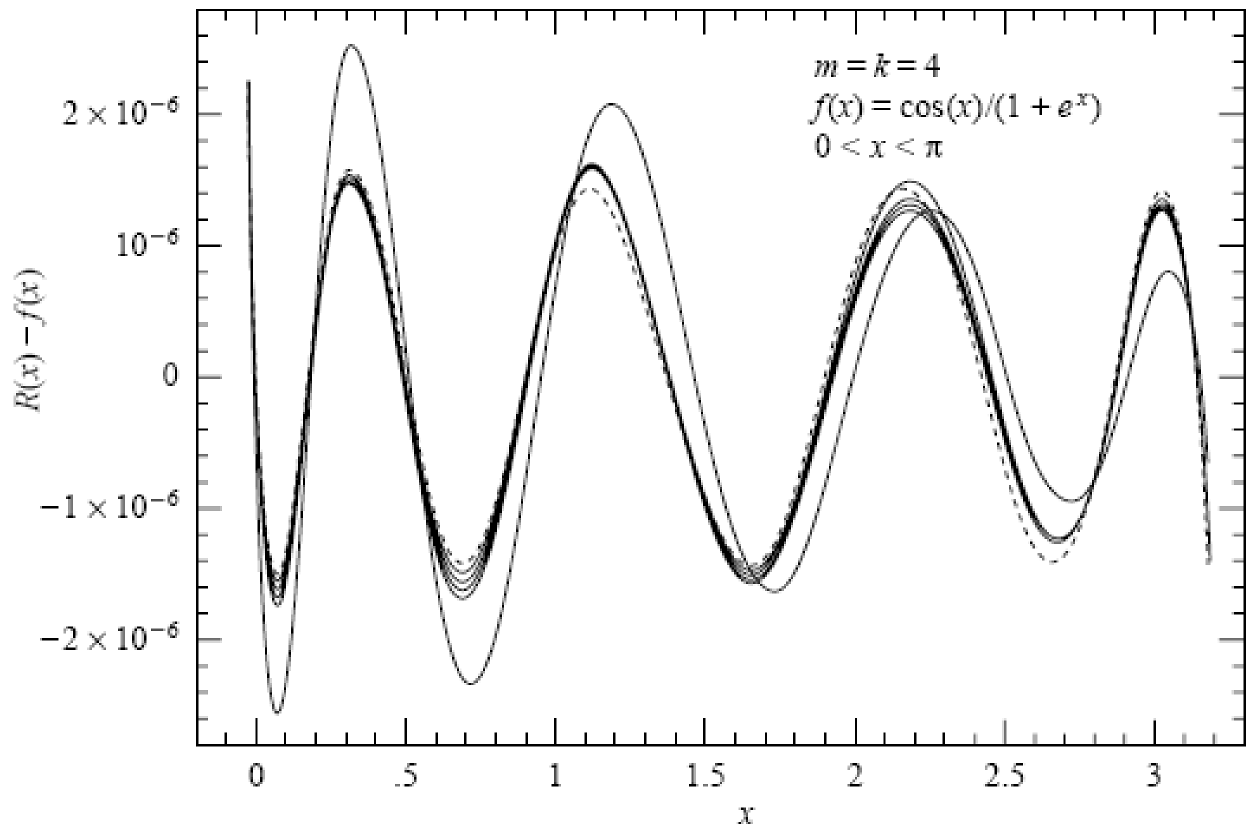



Figure 5.13.1. Solid curves show deviations $r(x)$ for five successive iterations of the routine `ratlsq` for an arbitrary test problem. The algorithm does not converge to exactly the minimax solution (shown as the dotted curve). But, after one iteration, the discrepancy is a small fraction of the last significant bit of accuracy.

# Using Chebyshev

- NR version3:
  - chebyshev.h
  - Create Chebyshev struct:

    Chebyshev cheb(func,xlow,xhi,order);
  - Calculate a function value:
    - cheb.eval(x,order);
  - Create the structs to calculate a derivative or the primitive of the function:
    - Chebyshev der = cheb.derivative();
    - Chebyshev integ = cheb.integral()

# Exercise 5: pdf of minimum ionizing particles

- Straggling losses of a minimum-ionizing particle may be approximated by a Landau distribution

$$pdf(E) = N\int_0^{\infty} e^{-x\ln x-(E-E_p)x}\sin(\pi x)dx$$

- With N a normalization constant and $E_p$ the most probable energy loss.
- Suppose we read out a scintillator with a photomultiplier and want to set a threshold on the signal. For this threshold, we want to calculate the fraction of minimum-ionized particles that will be missed, and the false alarm rate: the probability that noise exceeds the threshold in the absence of a particle. We want to distinguish also between the passage of 1 particle and of 2 or more particles.
- In order to calculate these probabilities we will use the Chebyshev formalism to store the pdf and cumulative pdf for signals from 1 and 2 particles passing the scintillator

# Landau straggling

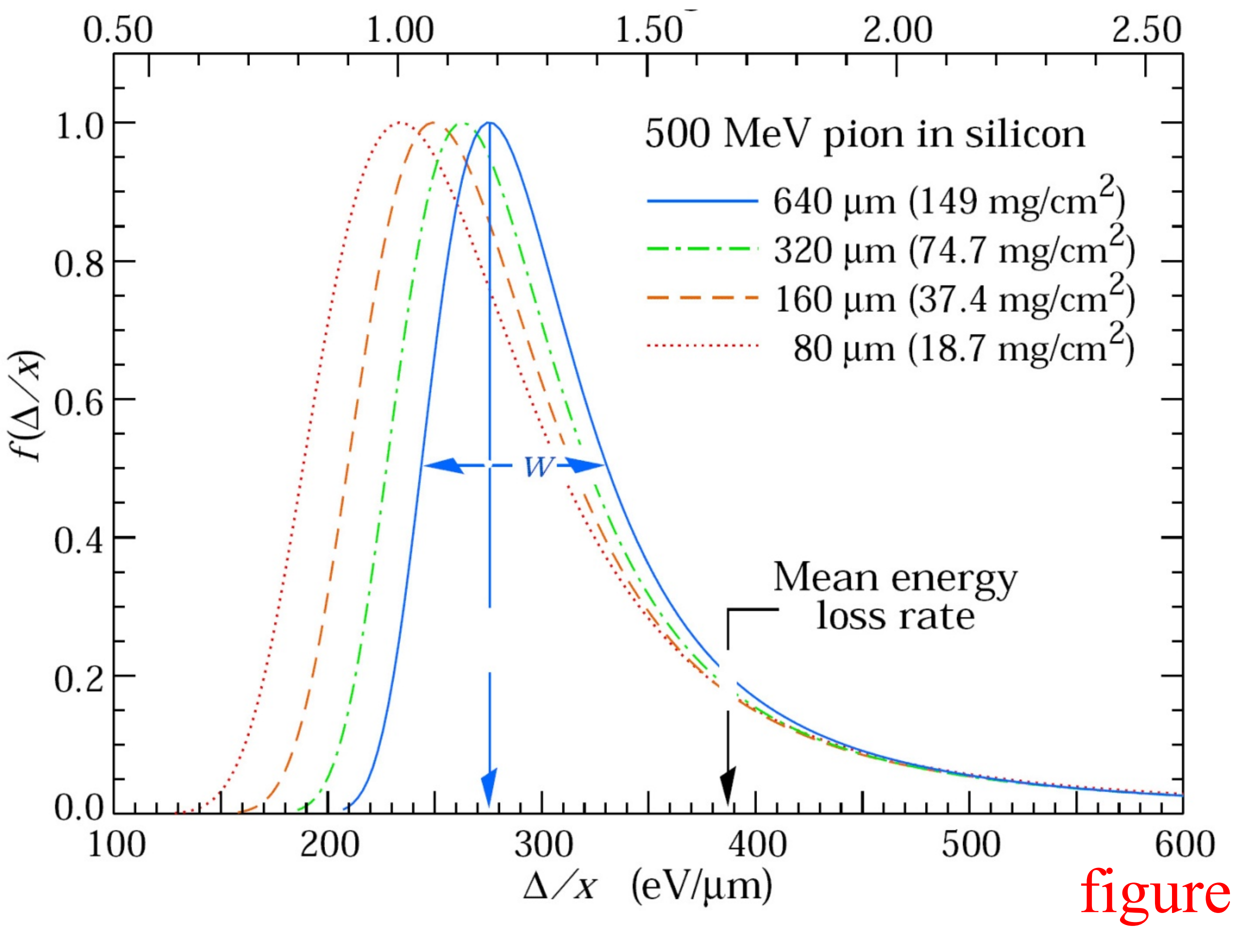



figure from
particle data group
pdg.web.cern.ch/pdg

**Figure 26.6:** Straggling functions in silicon for 500 MeV pions, normalized to unity at the most probable value $\Delta_p/x$. The width $w$ is the full width at half maximum.

# Exercise 5

- Scintillator response:
  - Noise, Gaussian distributed, x=signal strength in the detector. The pdf(x) follows the normal distribution, the cumulative pdf can be calculated from the error function:

$$\frac{1}{\sqrt{2\pi\sigma^2}}\int_{-\infty}^{x} e^{-\frac{x^2}{2\sigma^2}} = \frac{1+erf(x/\sqrt{2\sigma^2})}{2}$$

  - Signal (energy deposit): the pdf follows the Landau distribution from the previous slide, with $E_p$ = 2.4.
  - Detector response: Noise+signal, for the noise take sigma = 0.25.

# Exercise 5

- Make Chebyshev objects for the probability density function of the Landau distribution (needed for E from 0 to infinity: make a coordinate transformation), for the detector response A in case of a single particle passing the scintillator and for the case that 2 minimum ionizing particles are passing (from A = -5 to infinity)
- The detector response is given by the sum of the noise and the signal(s)
- The cumulative pdfs are given by the integrals of the pdfs (make Chebyshev approximations for that too).
- In order to calculate the pdf for the sum of noise and signal, one needs to convolve:

$$pdf_{detresponse}(A) = \int pdf_{signal}(A - x)\, pdf_{noise}(x)\, dx$$

# Exercise 5

- In order to calculate the pdf for the detector in case 2 minimum-ionizing particles pass, one needs to convolve the Landau distribution with the pdf of the detector for 1 passing particle (the sum of noise and signal).
- Assume that 40 million measurements are made per second, and that one requires a false-alarm rate of less then 10 Hz. Determine the threshold for this case (i.e. the value that results in less than 10 Hz noise triggers). Determine the false rejection rate for this threshold: the probability that a minimizing ionizing particle did not trigger the detector (sum of noise and signal fall below the threshold).
- Determine with an accuracy of 0.001 for which signal strength the pdfs of one and two minimum-ionizing particles is the same (this somewhere between the maxima of the single and double-particle distributions). For this value, determine the cpdfs. Give  the probability that a single particle gives a larger signal than this threshold and the probability that two simultaneously passing particles give a smaller signal than this threshold.
- From these pdfs, one can calculate likelihoods that no particle, one particle, or two particles were present. Usually one takes the logarithm of the pdfs of each hypothesis and compares them. From a set of measurements one can then determine what the relative likelihoods are. In this example, the distinguishing power is poor: it is difficult to separate single and double particle hits from a single measurement.