National University of Singapore

# PC5215 – NUMERICAL RECIPES WITH APPLICATIONS

(Semester I: AY 2010-11)

Time Allowed: 2 hours

---

Instructions to Candidates:

1. This examination paper contains SIX questions and comprises TWO printed pages.
2. This is a closed book examination.
3. Questions carry equal marks.
4. Answer all SIX questions.
5. Non-programmable calculators are allowed.

1. Consider the calculation of two equivalent expressions
   a) $\sqrt{b^2 + a} - b$
   b) $\dfrac{a}{\sqrt{b^2+a} + b}$

   in 4-digit decimal precision (four significant figures) for the intermediate steps as well as the final results, with $a$ = 0.001000, $b$ = 8.000. Perform the computation and comment on errors of floating-point number arithmetic.

   *$b^2$= 64.00, $b^2$+a = 64.00 + 0.001000 = 64.001000 ≈ 64.00, so $\sqrt{b^2 + a}$ =8.000. For part a) we get 8.000 – 8.000 = 0.000. For part b) we get 0.001000/(8.000 + 8.000) = 0.00006250 =$6.250\times10^{-5}$ . This should be compared with a more exact value $6.24997558613\times10^{-5}$.*

   *Four significant figures mean 4 digits for the mantissa if written in scientific notation.*

   *Method a) causes a catastrophic cancellation when we do subtraction, thus loss much accuracy in floating point calculation.*

2. Consider the calculation of determinant of a matrix $A$ of size $n$ by $n$.
   a) Describe two different algorithms and state their computational complexities.
   b) Suppose det($A$) is already computed, now one of the elements, say $a_{11}$, is changed. Can we compute the new determinant faster than a re-computation of the determinant of the new matrix? Give a procedure to do so if you can.

   *a ) we can use (1) LU decomposition. let A = LU, det(A) = product of the diagonal values in matrix U (known as $\beta_{jj}$); computational complexity is $O(N^3)$ for an N×N matrix A. (2) we can use Laplace minor expansion recursively, this take O(N!), which is much slower.*

   *b ) Yes. Let assume $a_{11}$ is change to $a_{11}+\delta$, and matrix A becomes A'. Then det(A') = det(A) + δ det($A_{11}$), we have used the property that determinant is a linear function with respect to one of the column vector, and $A_{11}$ is the matrix with the first row and first column deleted. Let $A^{-1}$=C, then $C_{11}$=$A_{11}$/det(A) (cramer's rule). If we already have LU decomposition, $C_{11}$ can be computed in $O(N^2)$by forward/backward substitution, and new determinant can be computed also in $O(N^2)$steps by det(A') = det(A)[ 1 + δ $C_{11}$]. Note that this worked for any elements not limited to the (1,1)* *element. Also note when $a_{11}$ is changed, all the other $\beta_{jj}$ also change we cannot reuse them correctly.*

3. Design an efficient Monte Carlo algorithm and present a pseudo-code to compute approximately the following two-dimensional integral:

$$\int_0^1 dx \int_0^x dy \cos(x^2\, y).$$

*s=0;*

*do i = 1, N*

*x = drand48();*

*y = drand64();*

*if (x > y) {*

*s += cos($x^2$y);*

*}*

*end do;*

*S=s/N*

*Metropolis algorithm will not work if we use the cos($x^2$y) as a probability distribution as we have unknown constant to determine (which is equivalent to find the value of the integral).*

4. Given the following quadratic form, $f(x,y) = x^2 + 2y^2 - x$, starting from the position $(x,y) = (0,1)$, and following the steps of the conjugate gradient method, find the set of values of $(x,y)$ such that the function reaches the minimum.

*Following the recipe of conjugate gradient method, we start at point $(x_0,y_0)=(0,1)$, the negative gradient is $n_0= g_0= -f' = (-2x+1, -4y) = (1,-4)$. First search line is x=t, y = 1 - 4t. This gives $f(x(t),y(t))=f(t)=33t^2-17t+2$. Minimum is reached at t=17/66, given*

*$(x_1,y_1)$=(17/66,-1/33)=(0.257575,-0.030303). For the second step the new gradient is $g_1$=(0.4848,0.1212). This gives $\gamma=|g_1|^2/|g_0|^2=(4/33)^2$=0.01469. The new search direction is $n_1=g_1+\gamma n_0$ = (0.499541, 0.0624426). With the new search line x=0.257575 + 0.499541 t, y = -0.03030 + 0.06244 t, we can locate t = 33/68 = 0.485294. This gives the final minimum position at $(x_2,y_2)$=(0.5, 0).*

5. Consider a least-squares fit to a parabola in the form: $y(x) = a + bx^2$, given the data points $(x_i,y_i)$, i = 1, 2, …*n*. Assuming that the standard deviations σ in *y* are all equal (but unknown) derive formulas that determine the coefficient *a* and *b*.

   *Same as the standard straight-line-fit formula if we replace x by $x^2$ and set the standard deviations $\sigma_j \equiv 1$. Steps skipped.*

6. Consider the following second-order ordinary differential equation $y''(x) + xy'(x) - y(x) = 0$. Give a discretization scheme accurate to third order in step size *h*. Let *q* = *y*, *p* = $y'$, and *x* as time *t*. Can we construct a symplectic algorithm for the equation?

   *Using central differences for second and first derivatives:*

   *y''(x)=[y(x+h)-2y(x)+y(x-h)]/$h^2$+O($h^2$),*

   *y'(x)=[(y(x-y)-y(x-h)]/h + O($h^2$),*

   *Put into the differential equation, we have*

   *(1+xh)y(x+h) – (2+$h^2$)y(x) + (1-xh) y(x-h) + O($h^4$) = 0.*

   *This means the method is accurate to $3^{rd}$ order and errors are in the 4-th order.*

   *We cannot construct a symplectic algorithm as the equation explicitly depends on time t, and the first derivative y' term represents a damping (fractional force) and the system cannot be written as a conserved Hamiltonian dynamics.*

-- the end --

[WJS]