AMSC 660 / CMSC 660
Fall 2015

Final Exam. Due Friday, Dec. 18, 10 AM

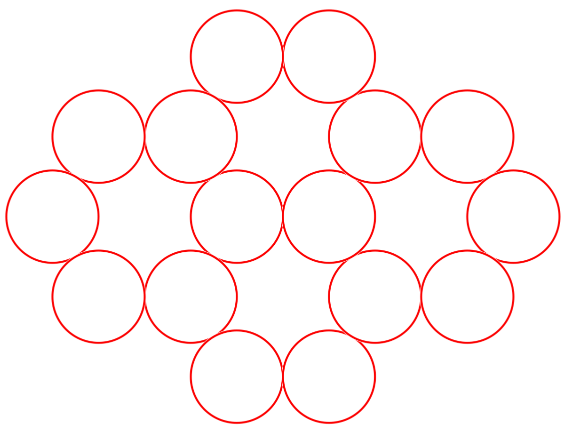
1. Consider 16 particles in 2D interacting according to the pair potential

$$V_{honey}(r) = \frac{5}{r^{12}} - \frac{6.50}{r^{10}} + 18.19e^{-2.21r} - 0.4e^{-40(r-1.755)^2}. \tag{1}$$

This potential was specially designed to favor the self-assembly of the honeycomb lattice [1]. The function to be minimized is the total potential of interaction of 16 particles

$$f(x_1, y_1, x_2, y_2, \ldots, x_{16}, y_{16}) = \sum_{i=1}^{15} \sum_{j=i+1}^{16} V_{honey}(r_{ij}),$$

where $r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$. Pick a suitable optimization method and set up a reasonable initial configuration to find the potential minimum shown in figure below.



Submit your code via ELMS.

Hint 1: A matlab routine `[f, g] = Honey(z)` *that computes the potential* $f(z)$ *and its gradient* $g(z)$ *is provided.* z *is a column vector with 32 entries, the first 16 entries are* $x_1, \ldots, x_{16}$*, the last 16 entries are* $y_1, \ldots, y_{16}$*. A matlab routine* `draw_configuration(z)` *is also provided.*

Hint 2: In order to set up a good initial configuration, plot the graph of $V_{honey}(r)$ *versus* r *and find its minima. (To find the minima, you can use matlab's* `fminsearch`*). The radii of the circles in the figure above are equal to one half of the first local minimizer* $r_1 \approx 1$*.*

$\frac{1}{2} - \frac{\sqrt{3}}{6}$	$\frac{1}{4}$	$\frac{1}{4} - \frac{\sqrt{3}}{6}$
$\frac{1}{2} + \frac{\sqrt{3}}{6}$	$\frac{1}{4} + \frac{\sqrt{3}}{6}$	$\frac{1}{4}$
	$\frac{1}{2}$	$\frac{1}{2}$

Table 1: Hammer-Hollingsworth, IRK, order 4

2. Integrate the canonical equations for the simple harmonic oscillator

$$\frac{dp}{dt} = -q, \quad \frac{dq}{dt} = p, \quad p(0) = 1, \quad q(0) = 0, \tag{2}$$

for the time interval $0 \le t \le T_{\max} = 32\pi$ using the Hammer-Hollingsworth method with the Butcher array given in Table 1. Do this using time steps $h = h_0 2^{-k}$, for $k = 1, 2, \ldots, 10$, $h_0 = \pi/2$. For each time step, compare your numerical solution at $T_{\max}$ with the exact one (find it analytically). Plot the graph of the error at $T_{\max}$ versus h in the log-log scale. Estimate the constant C and the power ρ in the error formula

$$Error \approx Ch^{\rho}.$$

Submit your code via ELMS.

Hint: You can use matlab's command `polyfit`.

3. (a) Read Chorin & Hald [2] about the Ising model and about the Markov chain Monte Carlo (the Metropolis algorithm): 2nd edition, pages 119 – 123, or 3rd edition, pages 150 – 152 and 157 – 161.

   (b) Write a matlab code to compute the mean magnetization m in the Ising model in 2 dimensions by the Metropolis algorithm (the Markov chain Monte Carlo algorithm presented in [2]), on a 30×30 lattice as a function of β. Make the boundary conditions periodic, i.e., the nearest neighbors of site (i, j) where $0 \le i, j \le 29$ are $(i \pm 1 \mod 30, j \pm 1 \mod 30)$ (see the matlab help for the command `mod`). Note that the analytic expression for the mean magnetization is

$$m(\beta) = \begin{cases} \left(1 - [\sinh(2\beta)]^{-4}\right)^{1/8}, & \beta > 1/T_c = 0.4408, \\ 0, & \beta < 1/T_c = 0.4408. \end{cases} \tag{3}$$

   (c) Calculate m for the set of values of $\beta =$ `0.2:0.01:1`. For each Monte Carlo run, make your program to plot the running mean of the magnetization, the running variance of the magnetization, and the running variance of the mean magnetization:

$$\bar{m}_k = \frac{1}{k}\sum_{i=1}^{k} m_i, \quad [\mathrm{Var}(m)]_k = \frac{1}{k-1}\sum_{i=1}^{k}(m_i - \bar{m}_i)^2,$$

$$[\mathrm{Var}(\bar{m})]_k = \frac{1}{k-1}\sum_{i=1}^{k}\left(\bar{m}_i - \frac{1}{i}\sum_{j=1}^{i}\bar{m}_j\right)^2.$$

Stop iterations as the running variance of the mean magnetization becomes less than some reasonable threshold. `Plot 1:` Plot the graph of the computed mean magnetization as a function of β and superimpose it with the graph of $m(\beta)$ given by Eq. (3). Plot the graph of the number of Monte Carlo iterations necessary to reach your stopping criterion vs β. For the case if there is some value of β for which convergence is not achieved in a reasonable time, restrict the maximal number of Monte Carlo steps by some `Nmax`. If for some values of β convergence is not achieved in `Nmax` steps, report about it in the comments to your program.

Submit your code via ELMS. Also submit a single pdf file with `Plot 1`.

References

[1] M. C. Rechtsman, F. H. Stillinger, and S. Torquato, Optimized interactions for targeted self-assembly: application to a honeycomb lattice, Phys. Rev. Lett. 95, 228301 (2005)

[2] A. Chorin and O. Hald, Stochastic Tools in Mathematics and Science, 2nd edition, Springer 2009