Annealing Meeting

Brendan Philbin, Chris Herdman

7/7/2020

* At this point, I have an operational implementation that simulates multiple seeds for spin configurations and MC sweeps
  + The simulation performs one full sweep on each configuration per iteration, printing results in real time and recording current and minimum energies in vectors
* Next goal seems to be simulated annealing, which would compare the configurations after a certain number of sweeps and copies/removes some depending on an algorithm
* Next: thermal annealing
  + Annealing flag that determines whether or not to enable
    - Change -a to be a boolean
  + Being with high temperature and increment downwards on each sweep until the goal temperature is reached
    - As last line of outermost MC loop, do something like: Beta -= Beta / M
  + This is done for all of the replicas individually, not implementing population annealing
  + Could do something like r=100 replicas
    - Plot histogram of minimum energy reached for all replicas to see what the distribution looks like
    - Get this distribution both with and without annealing
* The slower we anneal, the better the results we expect to get
  + Jumping out of local minimum is very difficult when not annealing
  + At high T (low B) we accept all updates, so it won’t get stuck
  + Allow system to fluctuate to prevent the system from freezing
* NEED TO CHANGE THE WAY YOU HANDLE MC\_SEEDS AND SPIN\_SEEDS
  + Have these seeds provide 1 but then set a vector of length (# of replicas) incrementing the initial seed by 1 on each element
* Double check the algorithm to make sure you haven’t done anything to encourage freezing
* Do trials with no annealing and 100, 200, 400 sweeps and then trials with annealing on and the same number of sweeps
  + Use r=100 replicas for each trial
  + This will give me 6 histograms
  + Use same Jij matrix for these runs
    - Try a few different ones to see if they provide different results
* MAKE ENERGY COMPUTATION MORE EFFICIENT
* Create an STL vector of RNGs that we seed individually and call them for each configuration
  + This way we don’t need to re-seed as I iterate through configurations
  + FIND A HEADER ONLY
* File opening: can have multiple files open at once – create an STL vector of filestreams
  + Opening and closing files can be very slow
  + Use “stream” operators instead of opening and closing
  + Open the files once and leave them opened to write into them throughout the loops, closing them at the end
* My current energies are doubles, be careful using “==”
  + Set a threshold value that is larger than machine precision
  + Define a Boolean “are these energies equal” function, compute absolute value of the difference in energies
* One option is to compute energies as integers