CS 6868: Concurrent Programming

Programming Assignment 02

Mahesh Gupta [CS12M018]

08th March 2013

---

1. Random Matrices

   Strategy:

   The matrix is initialized each entry with rand_r method.

2. Shuffle

   Strategy:

   The strategy is based on the fact that given a set of row and column transformations to be applied on a matrix, it can be converted into series of row transformations followed by series of column transformation so that final output remains the same unless two row or column transformation order is not interchanged.

   From the given list of transformations to be applied, we separate them into a list of row and a list of column transformation.

   To do the row transformation, we do first transformation followed by next non conflicting transformation and so on, all in parallel. Then we wait for all these operations to be completed. Now from the remaining list of transformations to be applied we do the first transformation followed by next set of transformation and so on until all row transformations are done. We do the same for column transformation as well.

   Given two transformations to be applied on a matrix, they conflict each other if one of the row or column is common between them. Such transformations should not be executed in parallel.

3. Threshold:

   Strategy:

   To do the thresholding, we first need to find the threshold intensity value below which all values should be set of 0. To find the threshold value, we have to calculate the distribution of intensity along each entry of the matrix. Once the statistics is calculated, the threshold can be calculated easily.

   From the given matrix, to calculate the threshold, for each row we count the statistical distribution of data and store in a separate matrix. Once these values are calculated, we merge them to find the total statistics. Now using algebraic formula we can find the threshold intensity value.

Once the threshold intensity value is calculated, we can again execute the code in parallel on each row to get the binary image and set each of its pixel value.

4. Game of Life:

Strategy:

For implementation, two matrices have been used one act as the original matrix and another as buffer matrix used to store the next step from current configuration. Each cell is processed concurrently using cilk_for construct to generate the next configuration. Now the next configuration matrix acts as the current step and program proceeds for no of steps.

Analysis

Problem name	Lines of code	Sequential run time (1000*1000 matrix) (seconds)	Parallel run time (1000*1000 matrix) (seconds)	Time to develop	Peak memory used in parallel phase (1000*1000 matrix)
randMatrices	81	5.037	4.111	1.15 hours	4 MB
Shuffle	228	5.117	3.280	1 hour	4 MB
threshold	150	3.382	2.932	1 hour	8 MB
Gameoflife	170	520.241	353.235	15 minutes	8 MB