Analysis of a Randomized Space Filling Algorithm and its Linear Program Formulation

Ka.Shrinivaasan (ka.shrinivaasan@gmail.com)

March 19, 2013

Abstract

In this article a grid gadget with binary contents is defined and a parallel algorithm to fill the grid coordinates with 1s is presented and its Linear Programming Formulation is analysed

1 Definition of a 2-dimensional Space Filling Grid Gadget

Let S be a finite 2 dimensional grid space with coordinates denoted by (x,y) to uniquely choose a point in the grid. Let the length of the finite space S be $2^{log(N)/2}$ for N > 0 and breadth be $2^{log(N)/2}$ making it a square grid. Thus the total number of points in the grid are $2^{log(N)/2} * 2^{log(N)/2} = N$.Thus any tuple that uniquely identifies a point in the grid ranges from $(0, 0)$ to $(2^{log(N)/2}, 2^{log(N)/2})$. Intially each point in the grid is set to 0. Goal is to find an efficient algorithm to set the maximum number of points in the grid to 1s(flipping 0 to 1 in that coordinate) or it can be above a threshold $K < N$. Some natural phenomena which use this grid filling algorithm:

1. Rainfall in which cloud "computes" the random coordinates on the earth to place the raindrops that too in parallel (this is an indefinitely parallel random coordinate computation which will be humongous task even for a supercomputer to simulate)

2. Randomly strewn sand with particles scattered at random points which again has a natural indefinite parallel computation of random coordinates.

2 Parallel Randomized Algorithm to Fill the Grid Gadget

Naive algorithm to fill the grid is to sequentially scan the rows and set the coordinates to 1 from 0 till the threshold is reached.This gives a polynomial or linear time algorithm.This can be parallelized with a parallel pseudorandom generator that outputs pseudorandom bits in parallel. There are pseudorandom generators that can output pseudorandom bits in parallel through multiple processors, for example PRG explained in [Tygar-Reif] which is a Randomized NC algorithm(RNC) and the Chaos theoretic Pseudorandom generator variant of Tygar-Reif which used Lehmer's generator in each processor.

Let the number of parallel processors be $k * logN$ in Tygar-Reif RNC algorithm.By executing this parallel PRG, $k*logN$ pseudorandom bits are generated in parallel. This $k*logN$ bit string can be split into k strings of logN bit length each.Each of the logN bit string can be split into ordered pair 2-tuples of the form $< size(logN/2), size(logN/2) >$. This tuple (x, y) denotes a coordinate point in the grid since grid is a square of side $logN/2$. Thus k ordered pairs are generated in parallel

by Tygar-Reif RNC algorithm. Thus k coordinates in the grid can be set to 1 in parallel. Thus instead of a linear time sequential algorithm ,a randomized NC parallel algorithm for grid filling is obtained.

3 Linear Programming and Grid Filling

Each coordinate in the grid can be denoted by a unique variable x_i where $1 < i < N$. Thus a linear program objective function can be defined as

$$Maximize(x_1 + x_2 + ... + x_N) \tag{1}$$

with N constraints being

1. $x_1 \leq 1$
2. $x_2 \leq 1$ ...
3. $x_N \leq 1$

Though simplistic the above is in standard linear program format with objective function and constraints. It is a known result that Linear Programming is P-Complete which implies LP is not parallelizable unless $NC = P$. The Simplex Algorithm uses Gaussian Elimination with Partial Pivoting(GEPP) which is a P-Complete problem with no known NC algorithm. Above LP formulation is a special case of linear program with only free variables in the constraints. The RNC algorithm to fill the grid coordinates with 1(s) in parallel is an indirect way to set the free variables in the Linear Program for the grid to 1 in parallel and at random. Thus there is an RNC algorithm for a special case linear programming with free variables in constraints. But this may not imply an RNC algorithm for the P-Complete problem of GEPP used by Simplex algorithm.

4 Acknolwdgement

I dedicate this article to God.

5 Bibliography

References

[1] RANDOM NUMBER GENERATORS ARE CHAOTIC -Charles Herring and Palmore,CACM, Vol. 38, Nu. 1, Technical Correspondence. (Appeared in ACM SIGPLAN Notices, 11, October 1989)

[2] Efficient Parallel Pseudorandom number generation - Tygar and Reif, SIAM journal of computing,Vol 17 No.2,1988

[3] Tygar-Reif variant with Lehmer's Chaotic Linear Congruence Generator replacing the bit computing function in each processor

[4] Various resources on Chaos theory,non-linear dynamics and attractors

[5] Various Lecture Notes and Resources on Simplex algorithm

[6] Various Lecture Notes and Resources on P-Complete problems

[7] Various Lecture Notes and Resources on Gaussian Elimination with Partial Pivoting or Gauss-Jordan algorithm