Parallel Random Number Generation

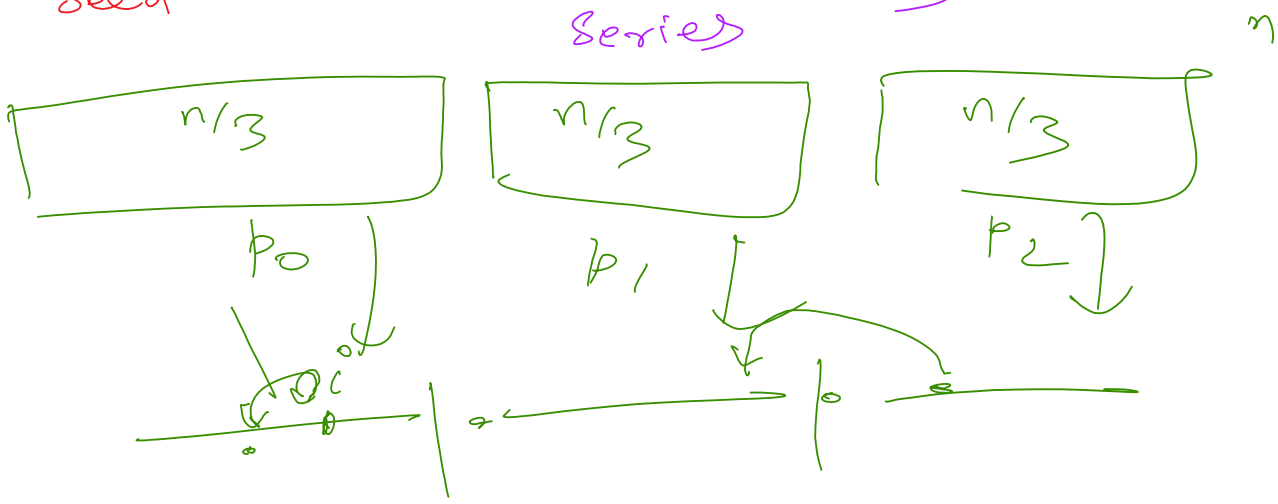
Tuesday, October 16, 2018 11:20 AM

Linear Congruential form :

Goal: x_0 x_1 - - - - x_i - - - - x_{n-1}

x_0 : Seed

Series



Desired Property

Serial output $\equiv$ parallel output

$\propto f(x_0)$ $\propto f(x_0)$

(independent of p)

Parallel Random Number

Tuesday, October 16, 2018 11:20 AM

index = $(A \times h(x) + B) \% \text{hash table size}(P)$

LC Generator:

$$x_i = (A\, x_{i-1} + B) \bmod P$$

Linear recurrence form:

$x_i = a\, x_{i-1} + b\, x_{i-2}$

Constants (user provided)

P: is Prime (big Prime)

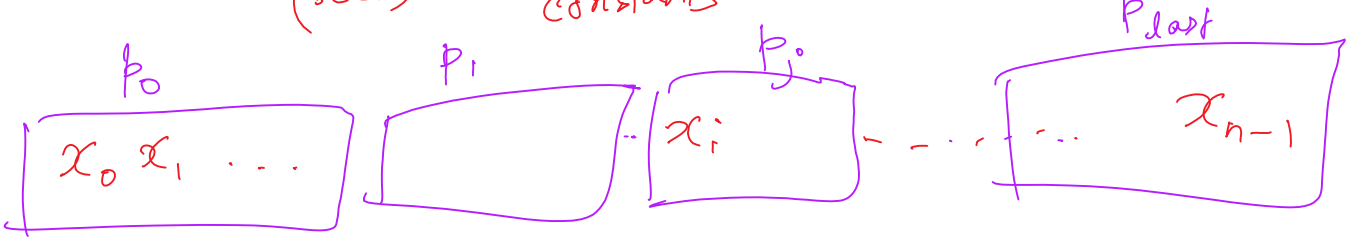
$\langle A_1, B_1 \rangle$
$\langle A_2, B_2 \rangle$
$\langle \vdots \rangle$
$\langle A_{10}, B_{10} \rangle$

Goal:

Using LC Generator, generate $x_0\ x_1 \ldots x_i \ldots x_{n-1}$ in parallel.

Input: x_0 (seed), $\{A, B, P\}$ constants (P: prime)

output



Parallel Random Number

Tuesday, October 16, 2018 11:20 AM

$x_i = (A x_i + B) \bmod P$

Algorithm:

same form

$$[\, x_i \quad 1 \,] = [\, x_{i-1} \quad 1 \,] \otimes \begin{bmatrix} A & 0 \\ B & 1 \end{bmatrix}$$

$\otimes$ operator:

$x_i = (x_{i-1} \times A + 1 \times B) \bmod P$

$$= [\, x_{i-2} \quad 1 \,] \otimes \begin{bmatrix} A & 0 \\ B & 1 \end{bmatrix} \otimes \begin{bmatrix} A & 0 \\ B & 1 \end{bmatrix}$$

$$= [\, x_{i-2} \quad 1 \,] \otimes M^2$$

$M = \begin{bmatrix} A & 0 \\ B & 1 \end{bmatrix}$

for i = 1 to n-1

$$[\, x_i \quad 1 \,] = [\, x_0 \quad 1 \,] \otimes M^i$$

$i - i$

for Project #4:

Serial implementations:

does this

serial_baseline(n) {

x_0 // init seed

for (i = 1 to n-1)

$x_i = (A x_{i-1} + B) \bmod P$

output $[x_0 \; x_1 \; \ldots \; x_{n-1}]$

}

check

serial_matrix(n) {

Init: $x_0 \leftarrow$ seed, $M \leftarrow \begin{bmatrix} A & 0 \\ B & 1 \end{bmatrix}$

$M_{next} \leftarrow M$

for (i = 1 to n-1)

$[\, x_i \quad 1 \,] = [\, x_0 \quad 1 \,] \otimes M_{next}$

$M_{next} \leftarrow M_{next} \otimes M$

output $[x_0 \; x_1 \; \ldots \; x_{n-1}]$

}

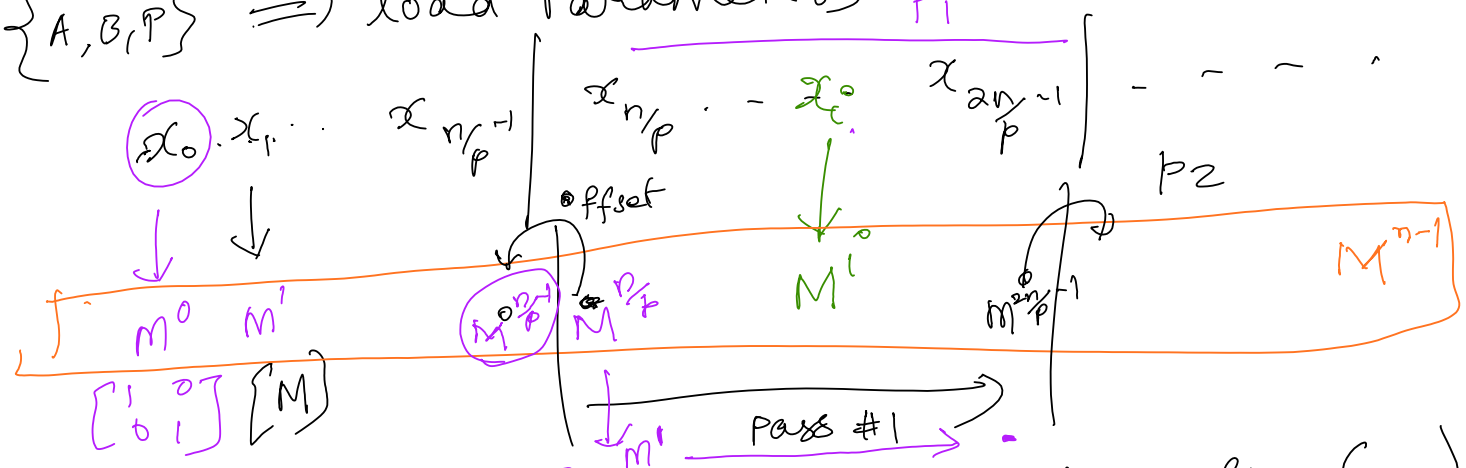
Parallel Random Number

Tuesday, October 16, 2018 11:20 AM

Parallel Implementation:

P_0 P_1 . . . P_j - - - P_{last}

$\{x_0, A, B, P\}$ ⟹ load Parameters P_1



n-element parallel prefix ( )

calls

p-element parallel prefix ( )

(M^i) offset

postprocess ( )

updates the local array (using offset)