Sieve of Eratosthenes

Parallelized with DPS

Sieve of Eratosthenes

Simple prime number sieve for finding all prime numbers up to any given limit.

Serial algorithm

In a list [2,L] of initially unmarked integers (1) find the first next unmarked number (2) mark all multiples of that number in the list (3) repeat until the square root of L has been reached. Remaining unmarked numbers are prime.

Amdahl's Law

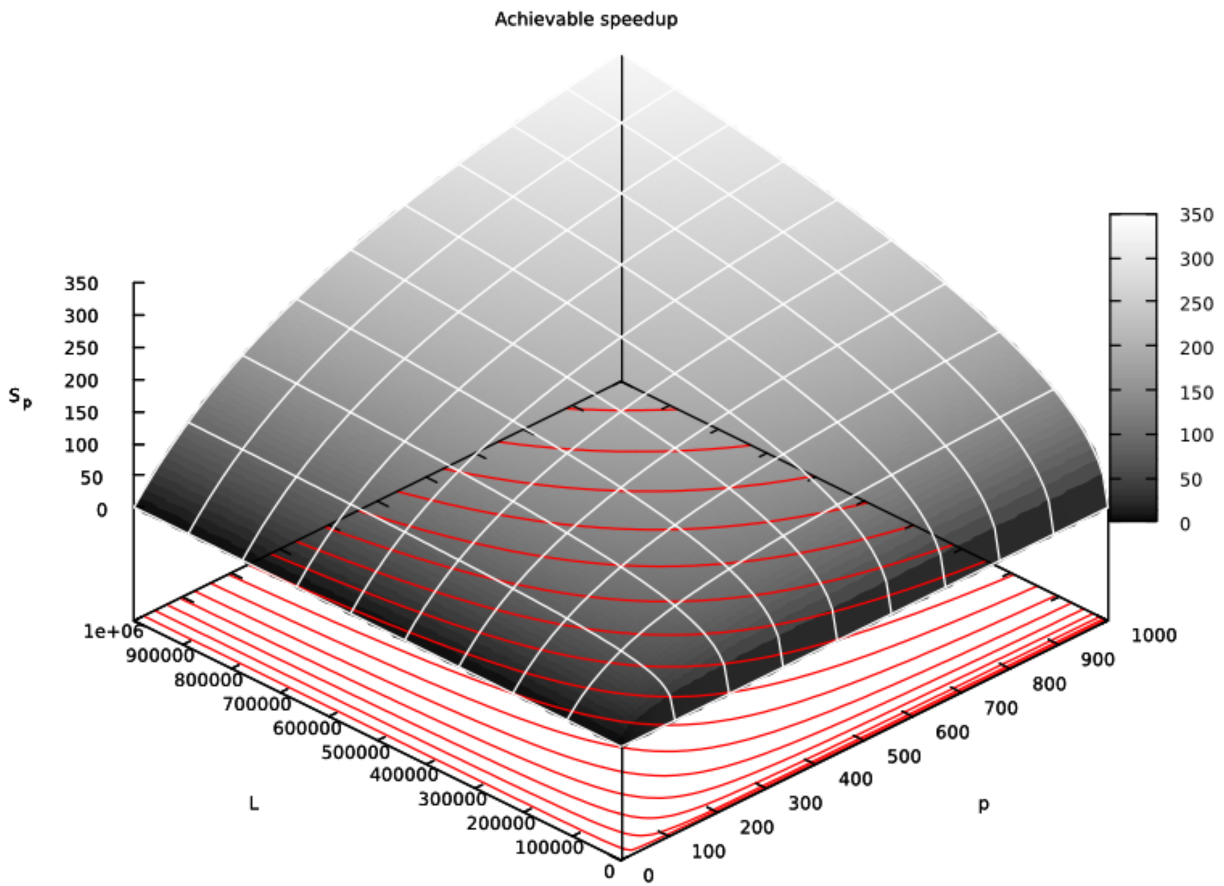
Serial part

- One loop to initialize a list of L numbers
- One loop to retrieve prime (unmarked) numbers in the sieved list

Parallelizable part: The actual sieving is considered fully parallelizable, with sqrt(L) loops over the list of numbers → $f = 2 / \sqrt{L}$

$$Sp = (p * \sqrt{L}) / (2p + \sqrt{L} - 2)$$

Amdahl’s Law



Achievable speedup
350
300
250
200
150
100
50
0
S_p
1e+06
900000
800000
700000
600000
500000
400000
300000
200000
100000
0
L
0
100
200
300
400
500
600
700
800
900
1000
p

Parallelization strategy

Pitfall: Most parallelization strategies highly increase communications.

Goal: Avoid communications.

Strategy

- Each node performs the sieve up to the square root of L.
- Each node is responsible of marking only a portion of the remaining list.

Computational complexity

Prime counting function $\pi \rightarrow k = \pi(\text{sqrt}(L))$

Computations: Each node processes

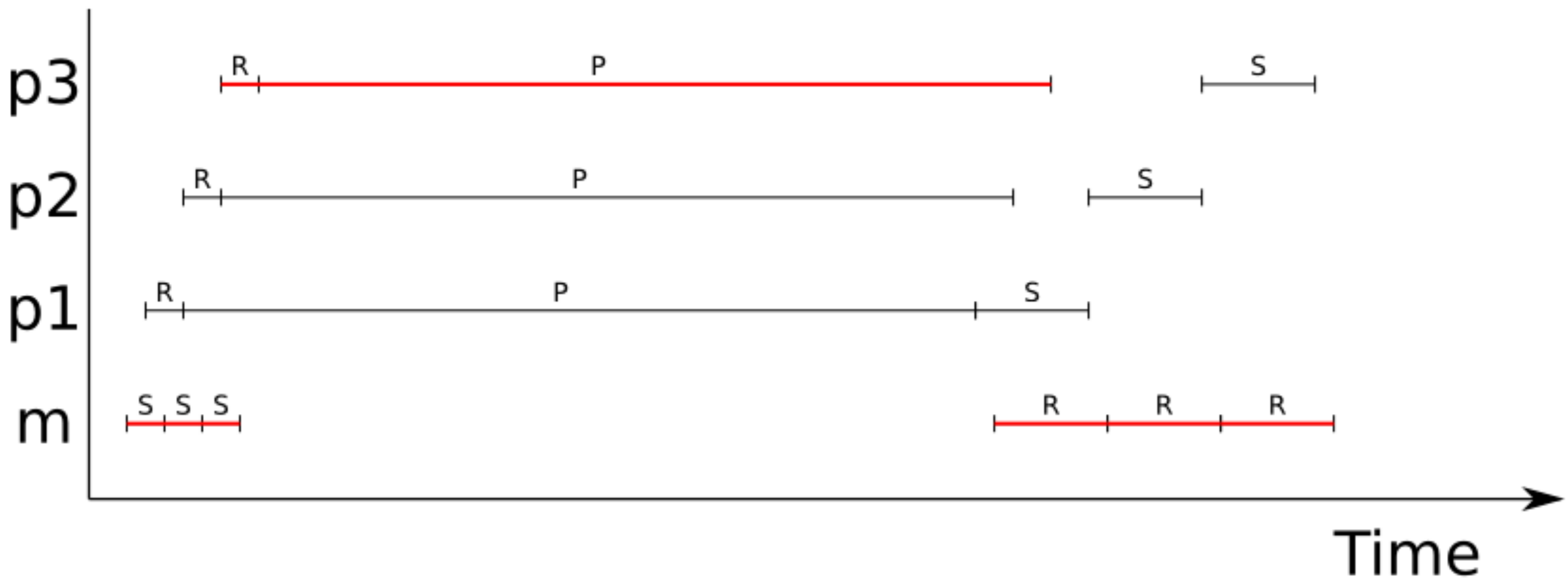
- numbers up to sqrt(L)
- k loops over own portion from sqrt(L) to L

Communications: Each node sends 1 message back to the master when finished.

C/C ratio: $p \ll \text{sqrt}(L) \rightarrow O(L^{1.5} / p^2)$

Timing diagram

(1) Master starts slaves (2) Slaves perform the sieve (3) Slaves send found primes to master.



Theoretical speedup

t_p: Total processing time
t_c: Total communication time
t_l: Network latency
t_r: Time to transmit all primes once

$$t_c = (n + 1) * t_l + t_r$$

$$Sp = 1 / (t_c + (n - 1) / (n * sqrt(t_p)) + 1 / n)$$

Theoretical speedup

Processing time: 6.4e-6 ms / number

→ approx. one prime found each 1.1e-4 ms

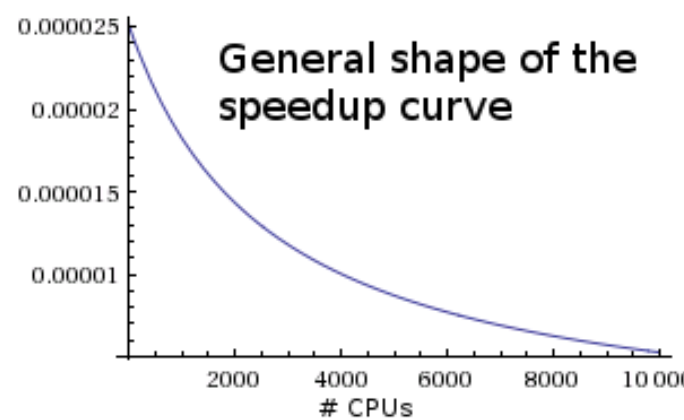
Communication time: 5.0e-4 ms / prime

Communication latency: 15 ms

Speedup

L = 1.6e9, n = 2..12

Sp → ~ 2.5e-5

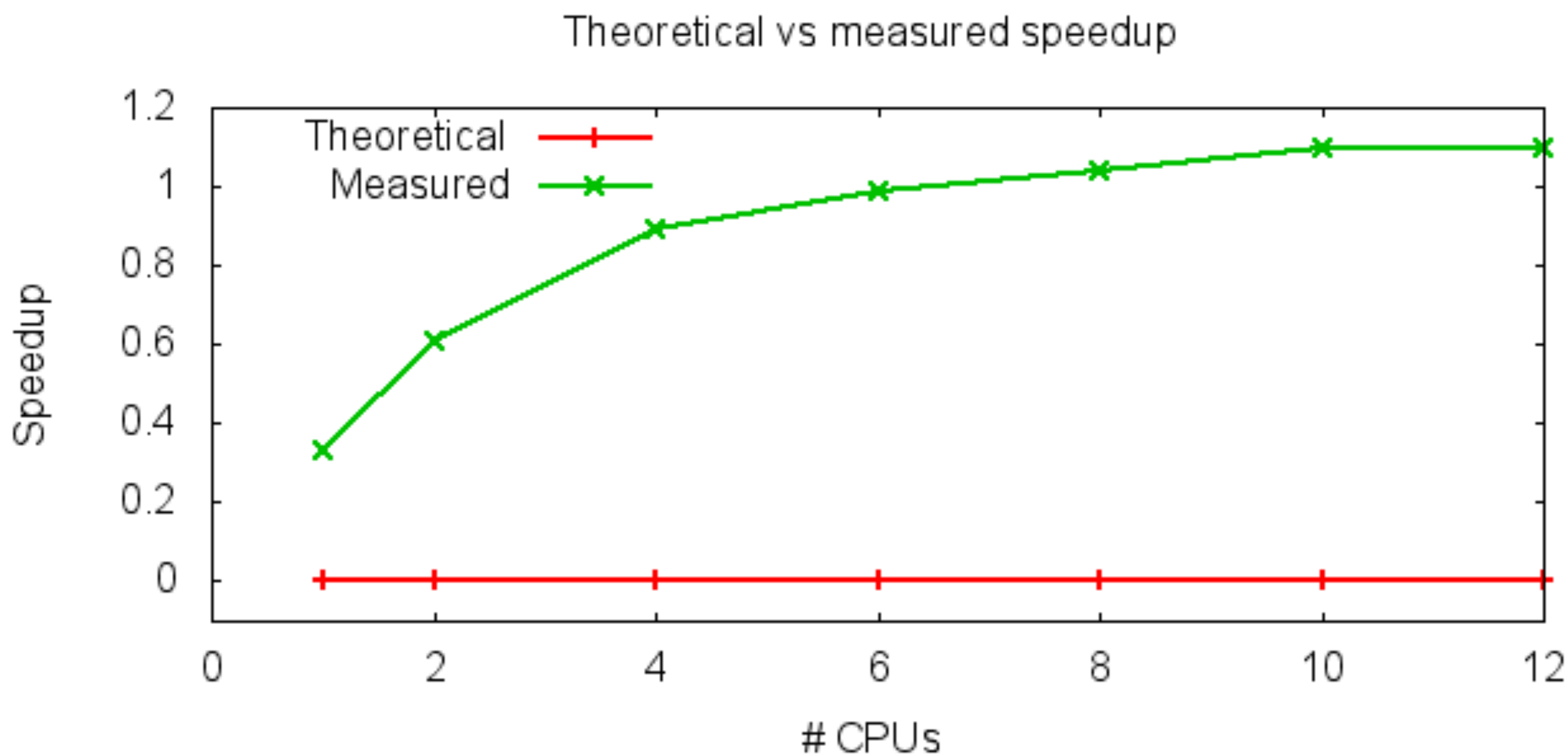


Execution times and Effective Speedup

L = 1’600’000’000; π(1.6e9) = 79’451’833

n	Execution time [ms]	Speedup
1 (Serial)	11905.4	-
1 (Parallel)	35996.38	0.33
2	19515.78	0.61
4	13327.40	0.89
6	12057.18	0.99
8	11454.46	1.04
10	10845.70	1.10
12	10828.62	1.10

Speedup vs speedup



Conclusion

Future plans

- Deeper investigation of network transfer times to try to explain the “good” measured speedups vs the poor expected ones.
- Try to reduce some memory limitations.