A Distributed Prime Sieving Algorithm based on *Scheduling by Multiple Edge Reversal*

Gabriel Paillard, Christian Lavault

PARIS 13 NORD

Felipe França

UFRJ Universidade Federal do Rio de Janeiro

CAPES

Plan

I- Prime Sieving

II- Scheduling by Multiple Edge Reversal (SMER)

III- Semi-SMER

IV- Conclusions

I - Prime Sieving

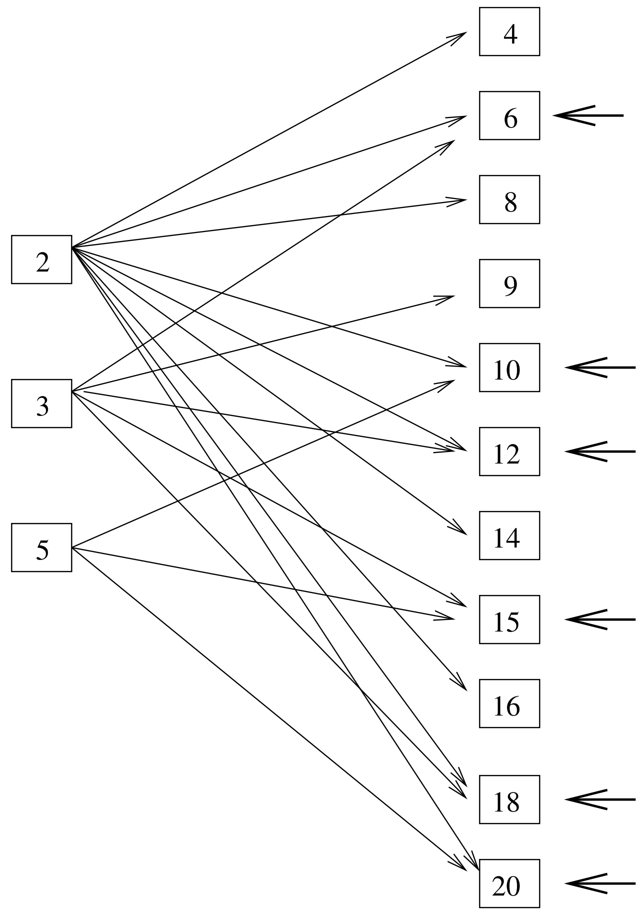
Some definitions :

- A prime number is a positive integer p having exactly two positive divisors, namely 1 and p.
- An integer n is composite if $n > 1$ and n is not prime.
- The number 1 is considered neither prime nor composite.
- The limit of generation is given by n.

The first sieve was conceived 2000 years ago.

Eratosthenes

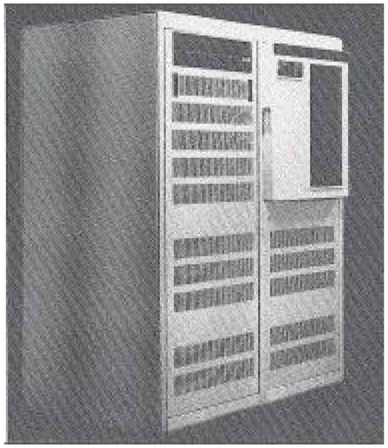
Sieve :



Eratosthenes'Sieve

Complexity : $O(n \log \log n)$

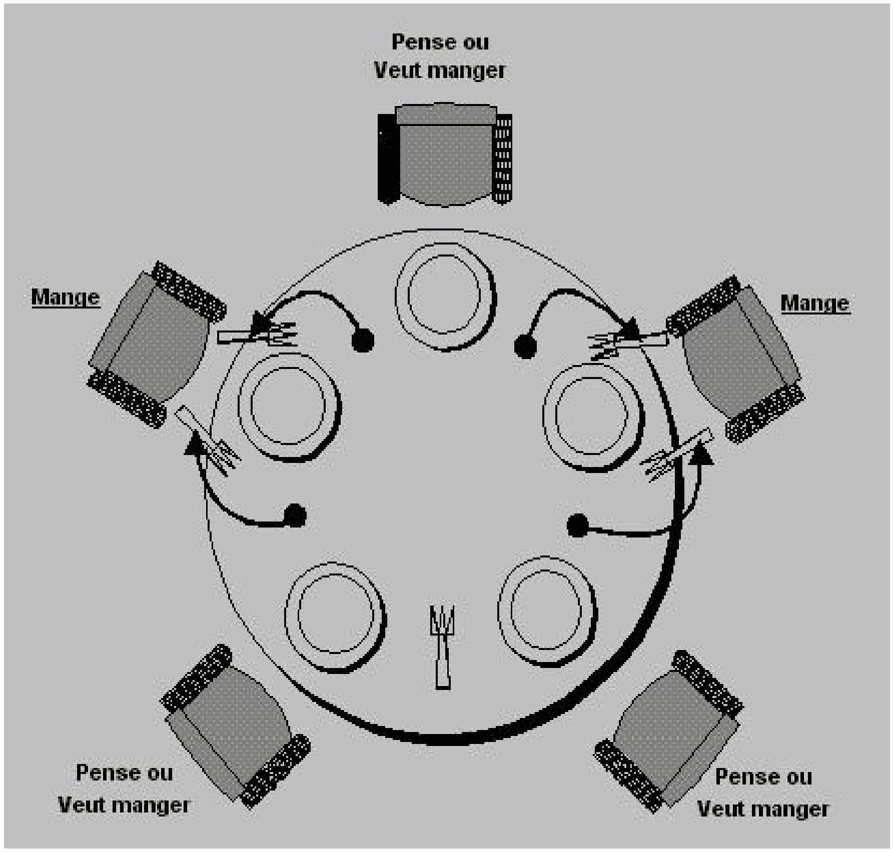
benchmarking of parallel architectures :



Flex/32 - Bokhari (1989)

II- Scheduling by Multiple Edge Reversal

SER - Scheduling by Edge Reverse

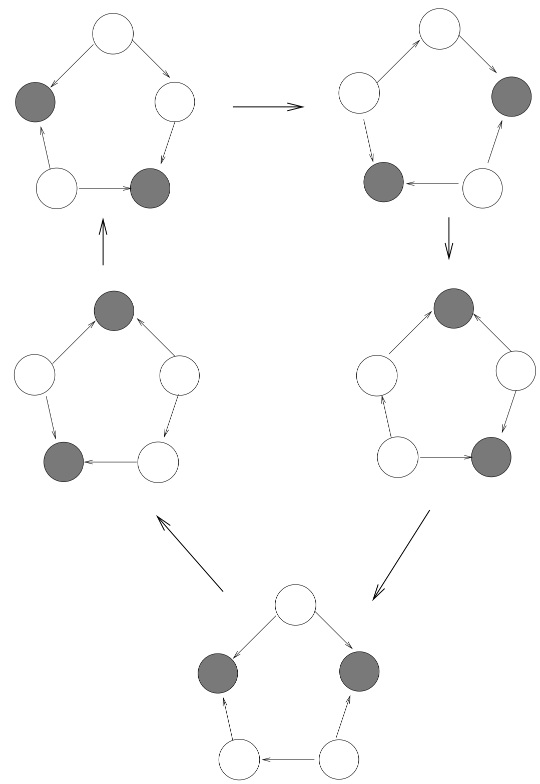


- Five philosophers in the original problem : three states ;

* Thinking
* Eating
* Hungry

- SER in a synchronous model : just to facilitate understanding ;
- Operations and message exchanging done at each step ;
- The SER starts from a initial acyclic and oriented graph ;
- Edge reversal is done via sending a simple message ;

SER dynamics for the Dinning Philosophers under heavy load



SMER - Scheduling by Multiple Edge Reversal

França (1993) França (1994) Barbosa, Benevides, and França (2001).

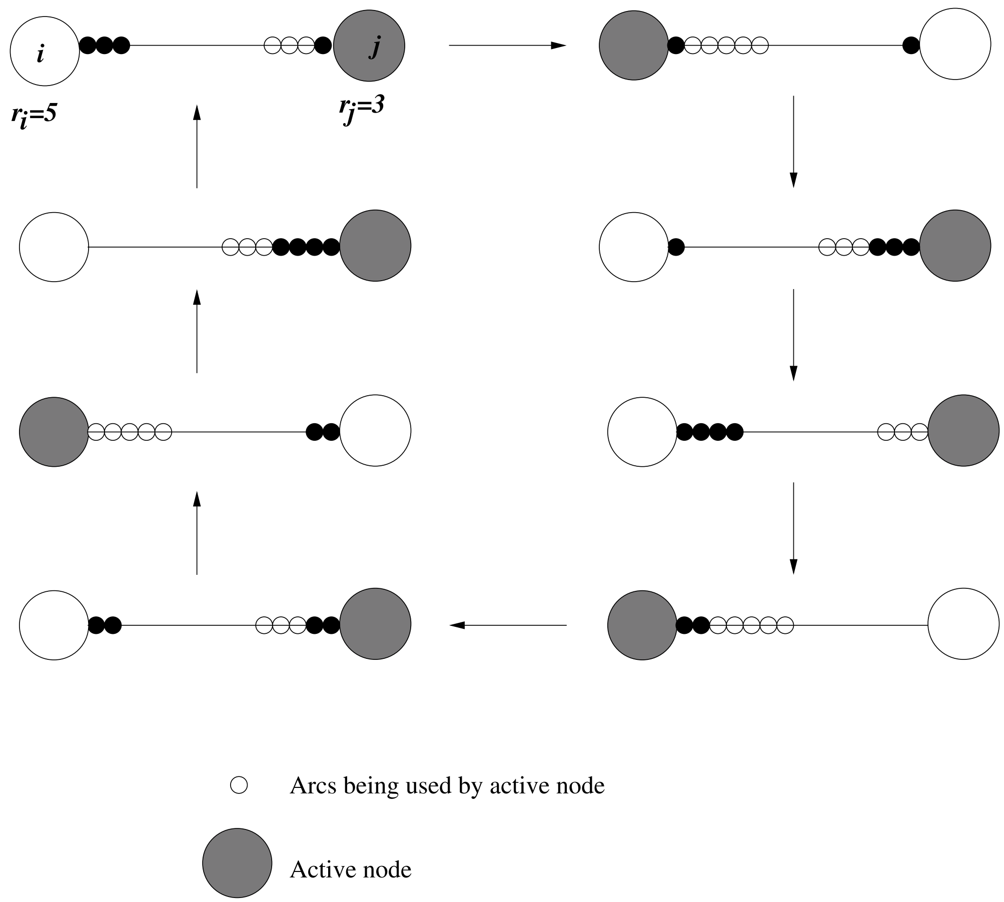
- SMER : generalization of SER allowing the co-existance of different *philosophic lines*, i.e. ;
- Possibility of specifying different access rates to shared resources ;
- Access rates are specified via the inverse of the variable (r_{node})

– The necessary number of arcs between two nodes (avoiding deadlock and assuring mutual exclusion) :

$$e_{i,j} = r_i + r_j - 1$$

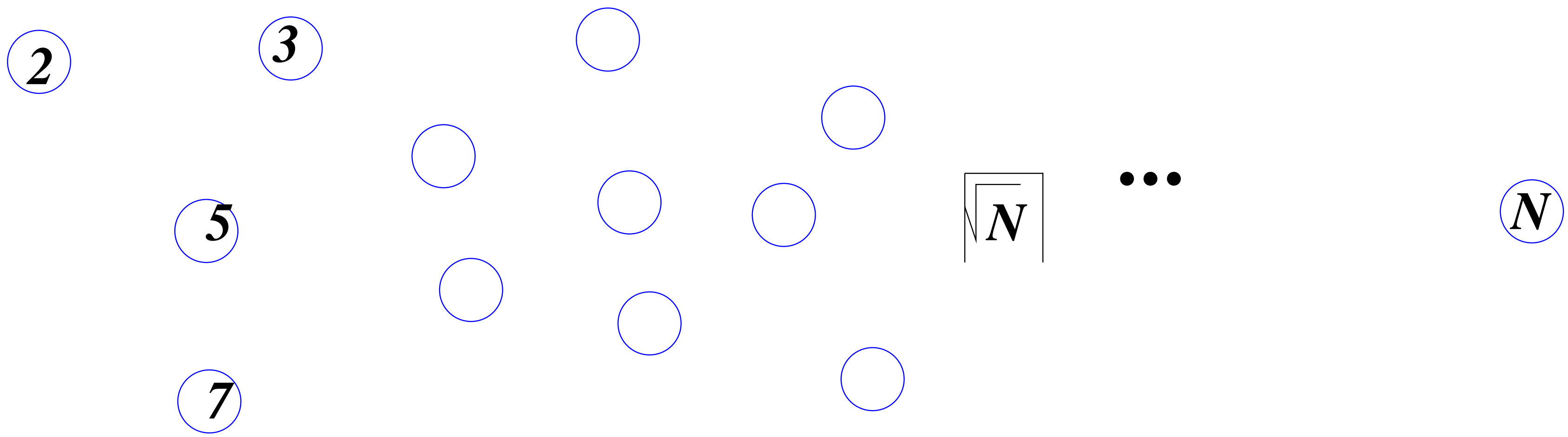
– Period length :

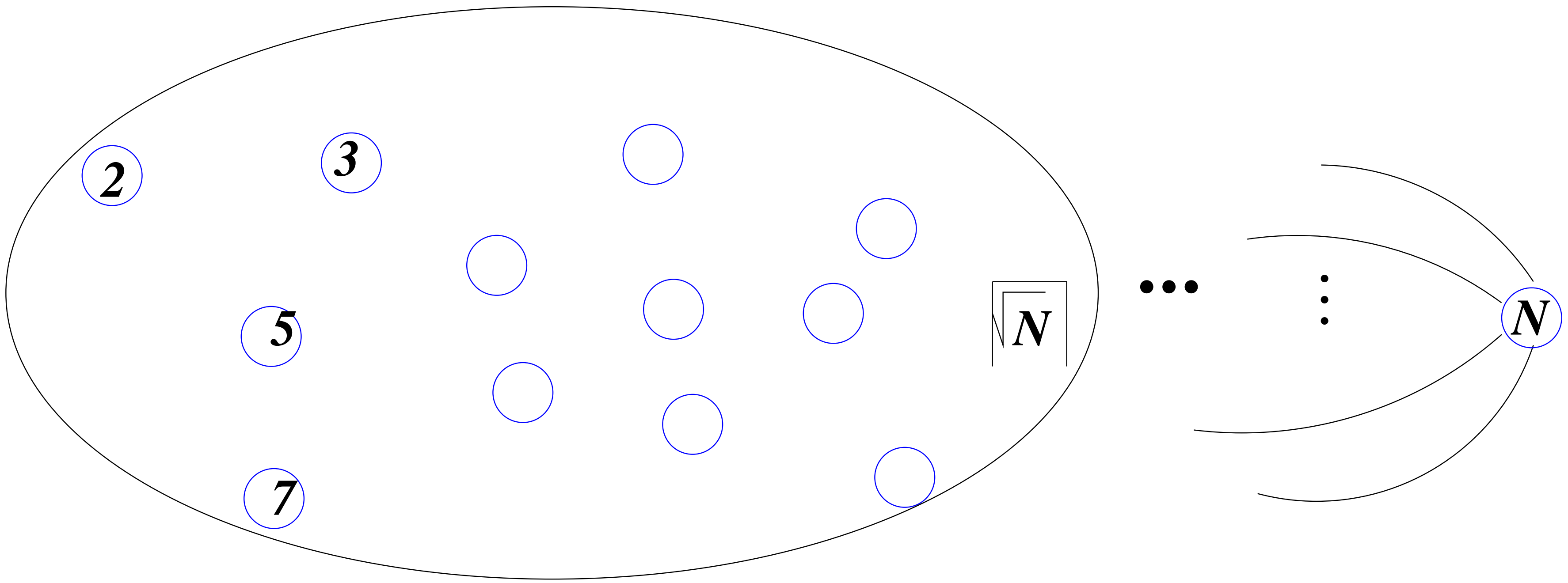
$$p_{i,j} = \frac{r_i + r_j}{gcd_{r_i, r_j}}$$



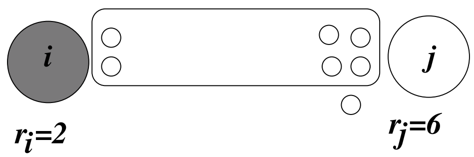
i
j
r_i=5
r_j=3
Arcs being used by active node
Active node

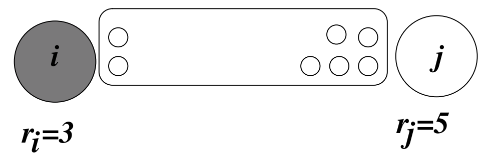
III- Semi-SMER


2
3
5
7
$\sqrt{N}$
•••
N


2
3
5
7
$\sqrt{N}$
...
N

$$f_{i,j} = r_i + r_j - gcd(r_i, r_j)$$

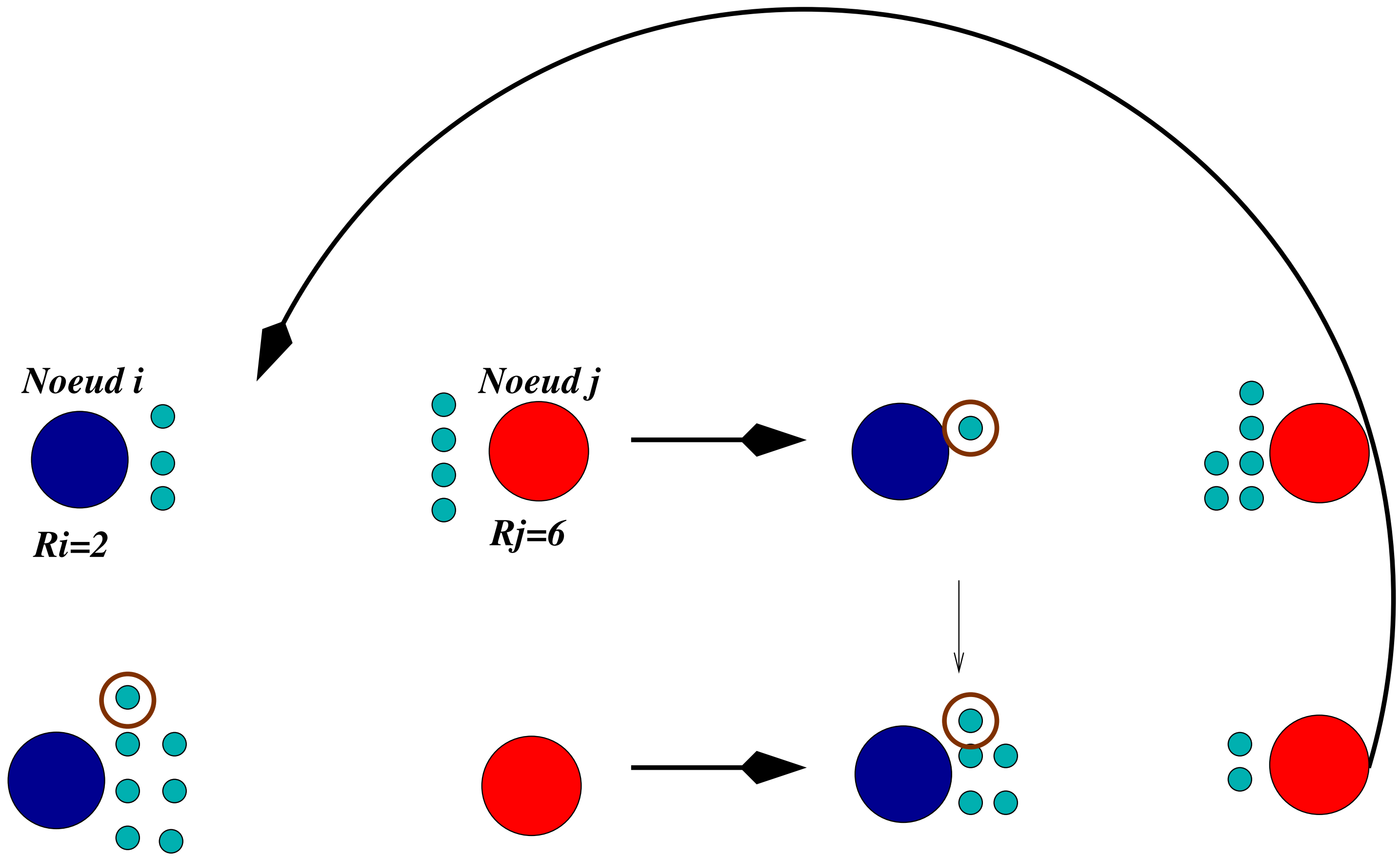




Theorem. Paillard (2004).

For all pair of neighboring nodes i and j, connected by $e_{ij} = r_i + r_j - 1$ arcs, of a given arbitrary multigraph $\mathcal{M}$ under SMER, such that $\gcd(r_i, r_j) > 1$, at least one of such e_{ij} arcs remains static after a period of length $P_{i,j}$ is reached. Otherwise ($\gcd(r_i, r_j) = 1$), no static arcs are observed.

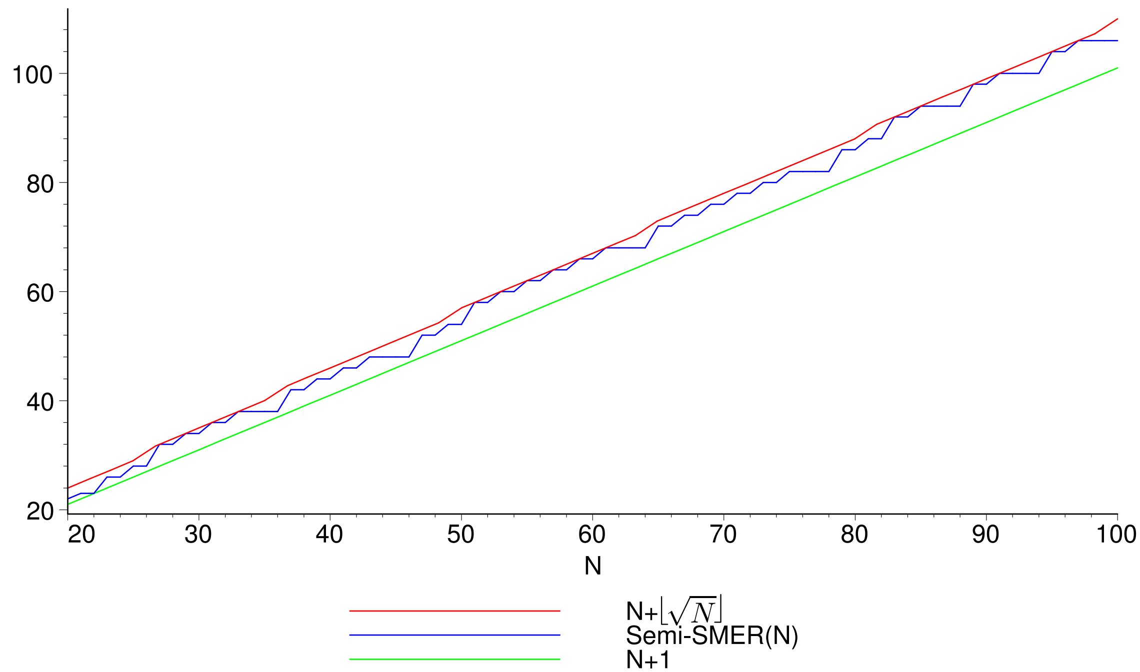
Proof : Consider two nodes i and j of the multigraph $\mathcal{M} = (V, \mathcal{E})$, with their corresponding "reversibilities" r_i and r_j. Two cases may arise. First, if r_i and r_j are coprime, then $f_{i,j} = r_i + r_j - 1 = e_{i,j}$. By contrast, when r_i and r_j have at least a proper common divisor, $\gcd(r_i, r_j) \geq 2$, and then $f_{i,j} \neq e_{i,j}$. Note that, conversely, whenever all arcs are reversed within the SMER period of $\mathcal{M}$, then $f_{i,j} = e_{i,j}$ and thus, there exists a pair of nodes (i, j), in $\mathcal{M}$, such that $(r_i, r_j) = 1$. ■

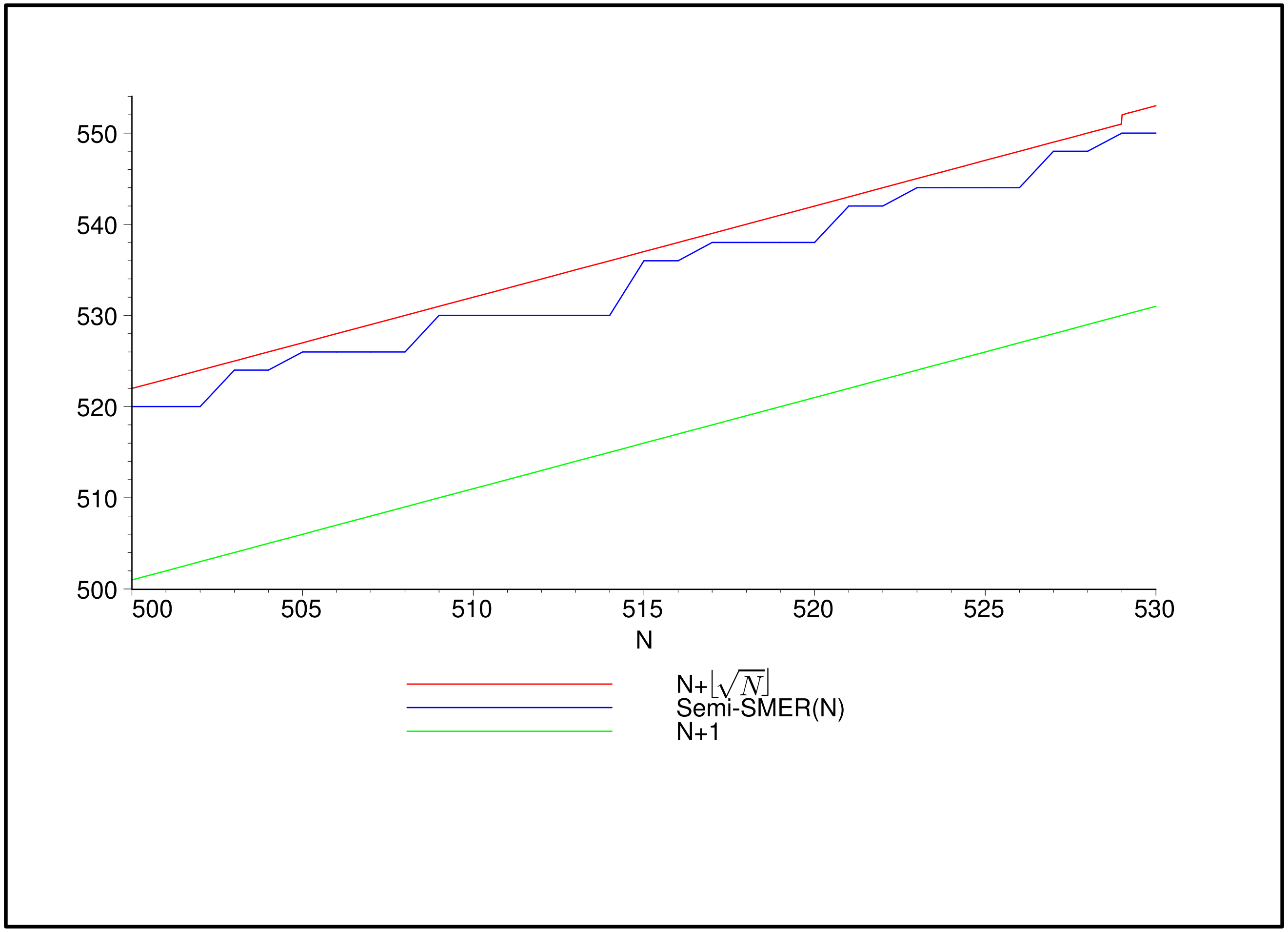

Noeud i
Ri=2
Noeud j
Rj=6

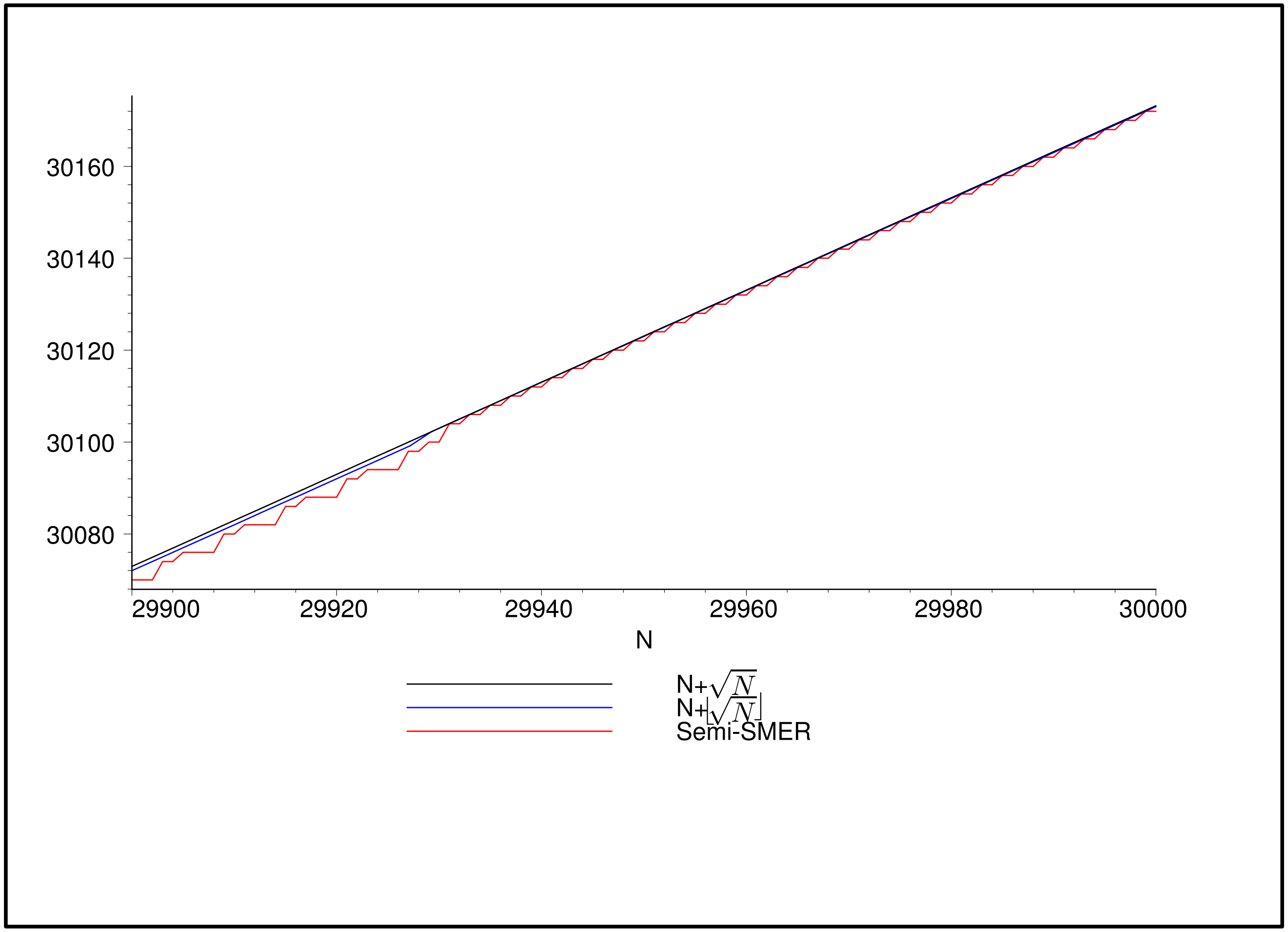
Algorithm analysis :

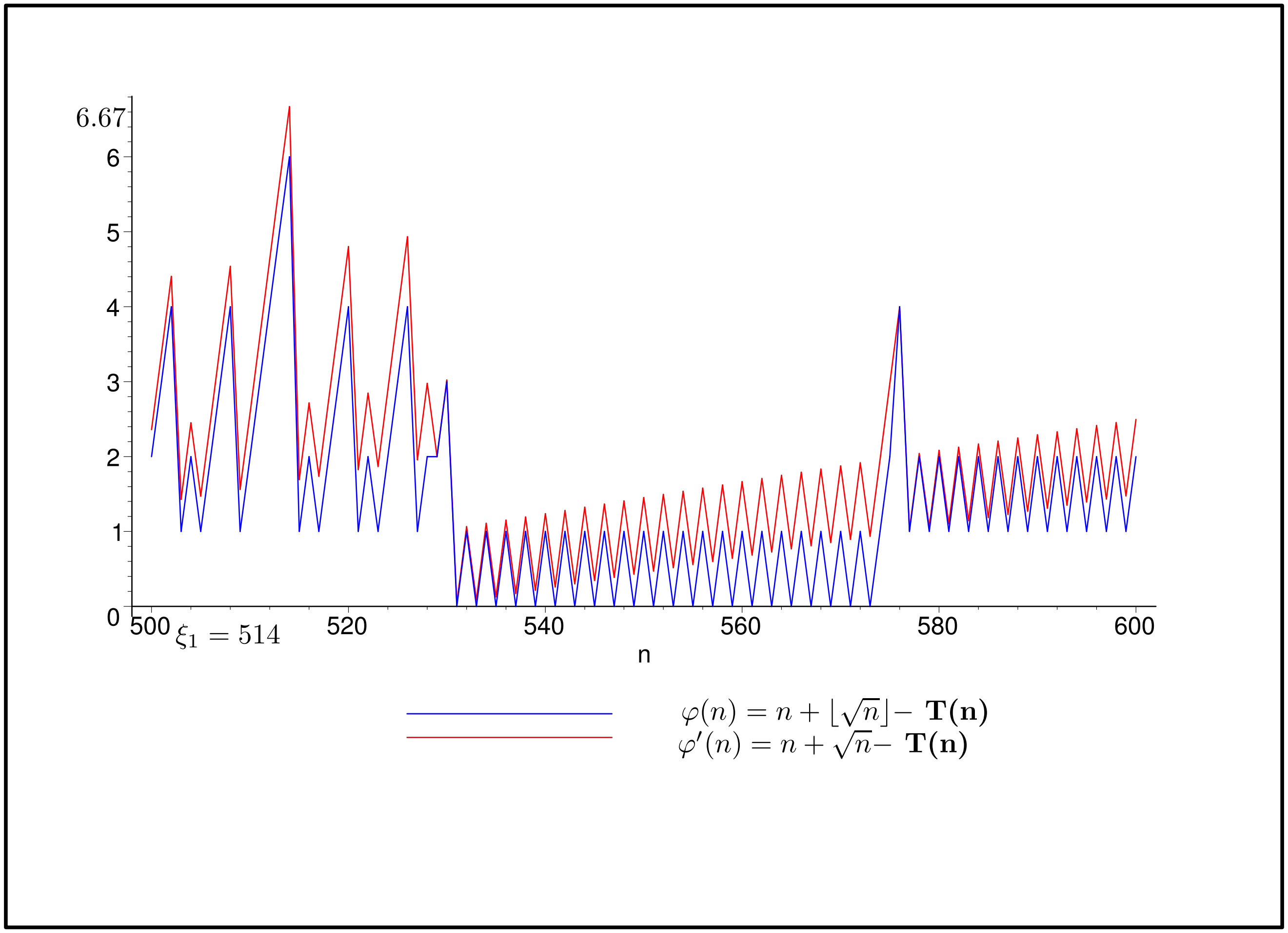
Theorem Paillard *et al.* (2005).

Given an oriented arbitrary multigraph $\mathcal{M} = (V, \mathcal{E})$ having N vertices, the maximum number of steps required by the algorithm Semi-SMER is at most $n + \lfloor \sqrt{n} \rfloor$. The message complexity of the algorithm achieves at most $n\Delta_N + \lfloor \sqrt{n} \rfloor \Delta_N$, where Δ_N denotes the maximum "multidegree" of $\mathcal{M}$. The maximum space required per process is linear.

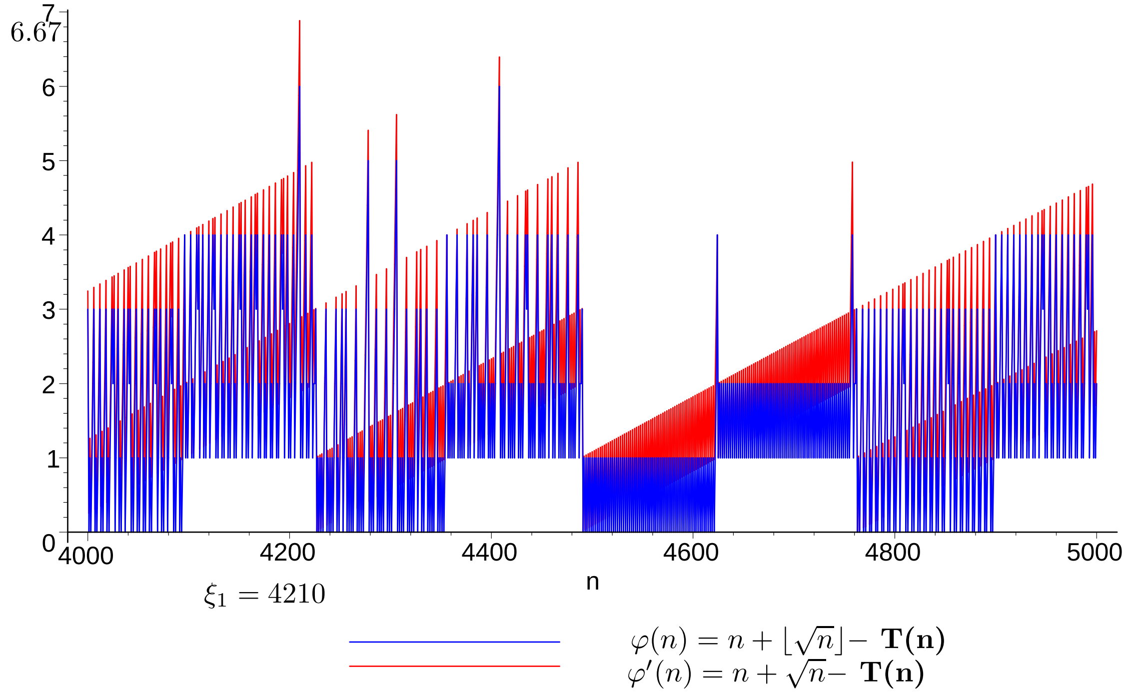

100
80
60
40
20
20
30
40
50
60
70
80
90
100
N
N+⌊√N⌋
Semi-SMER(N)
N+1


550
540
530
520
510
500
500
505
510
515
520
525
530
N
N+$\lfloor\sqrt{N}\rfloor$
Semi-SMER(N)
N+1


30160
30140
30120
30100
30080
29900
29920
29940
29960
29980
30000
N
N+√N
N+⌊√N⌋
Semi-SMER



6.67
6
5
4
3
2
1
0
500
$\xi_1 = 514$
520
540
560
580
600
n
$\varphi(n) = n + \lfloor\sqrt{n}\rfloor - \mathbf{T(n)}$
$\varphi'(n) = n + \sqrt{n} - \mathbf{T(n)}$



7
6.67
6
5
4
3
2
1
0
4000
4200
4400
4600
4800
5000
n
$\xi_1 = 4210$
$\varphi(n) = n + \lfloor\sqrt{n}\rfloor - \mathbf{T(n)}$
$\varphi'(n) = n + \sqrt{n} - \mathbf{T(n)}$

IV - Conclusions

– New SMER-based prime generation was introduced (fundamental operation : comparison) ;

– No gcd computation !

– No precomputation assumed in the complexity analysis (precomputation takes $O(n \log \log \Pi_k)$, Π_k being the multiplication of the first k prime numbers, in the wheel sieve) ;