



UNIUNEA EUROPEANĂ



GUVERNUL ROMÂNIEI



Instrumente Structurale
2007-2013



Platformă de e-learning și curriculum e-content
pentru învățământul superior tehnic

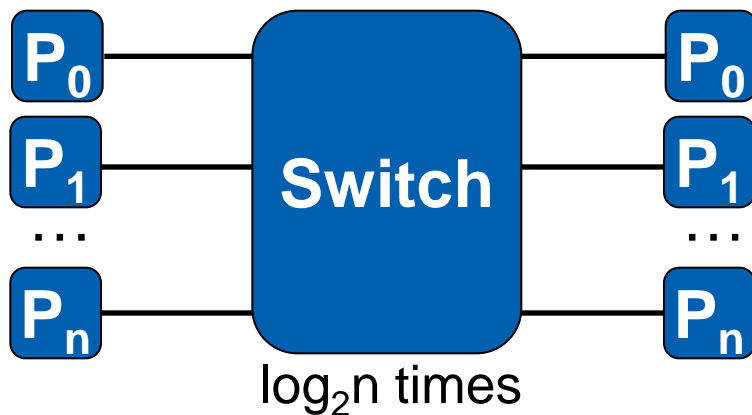
Arhitectura Sistemelor de Calcul

22. Interconectarea cu intercalare perfecta



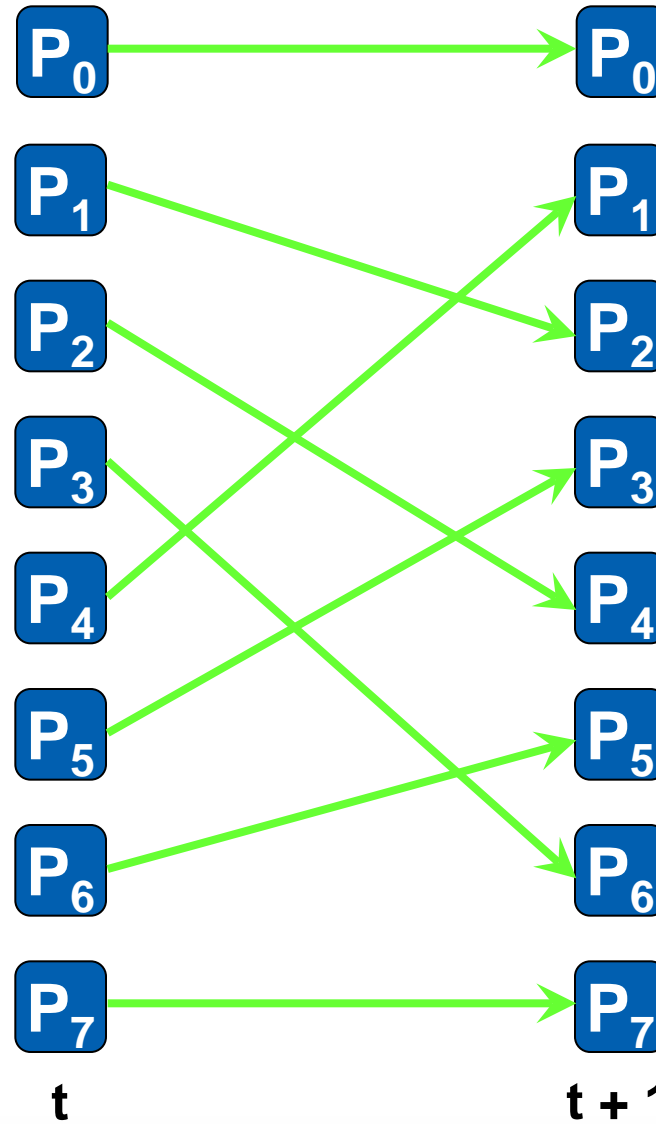
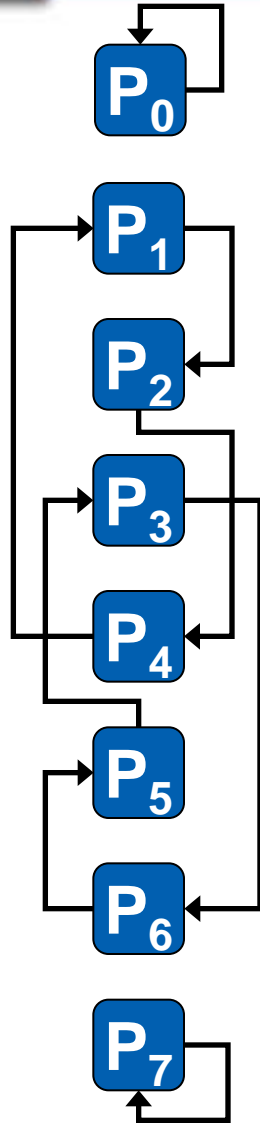
Intercalarea Perfecta – Shuffle

- Este adecvata pentru:
 - Transformata Fourier (FFT – Fast Fourier Transformation)
 - Algoritmi de sortare
 - Transpuneri de matrice
- Asigura conectarea intre procesoarele unei structuri SIMD de n procesoare cu:
 - $n-2$ comunicatii simultane
 - dupa $\log_2 n$ iteratii datele ajung pe procesorul sursa



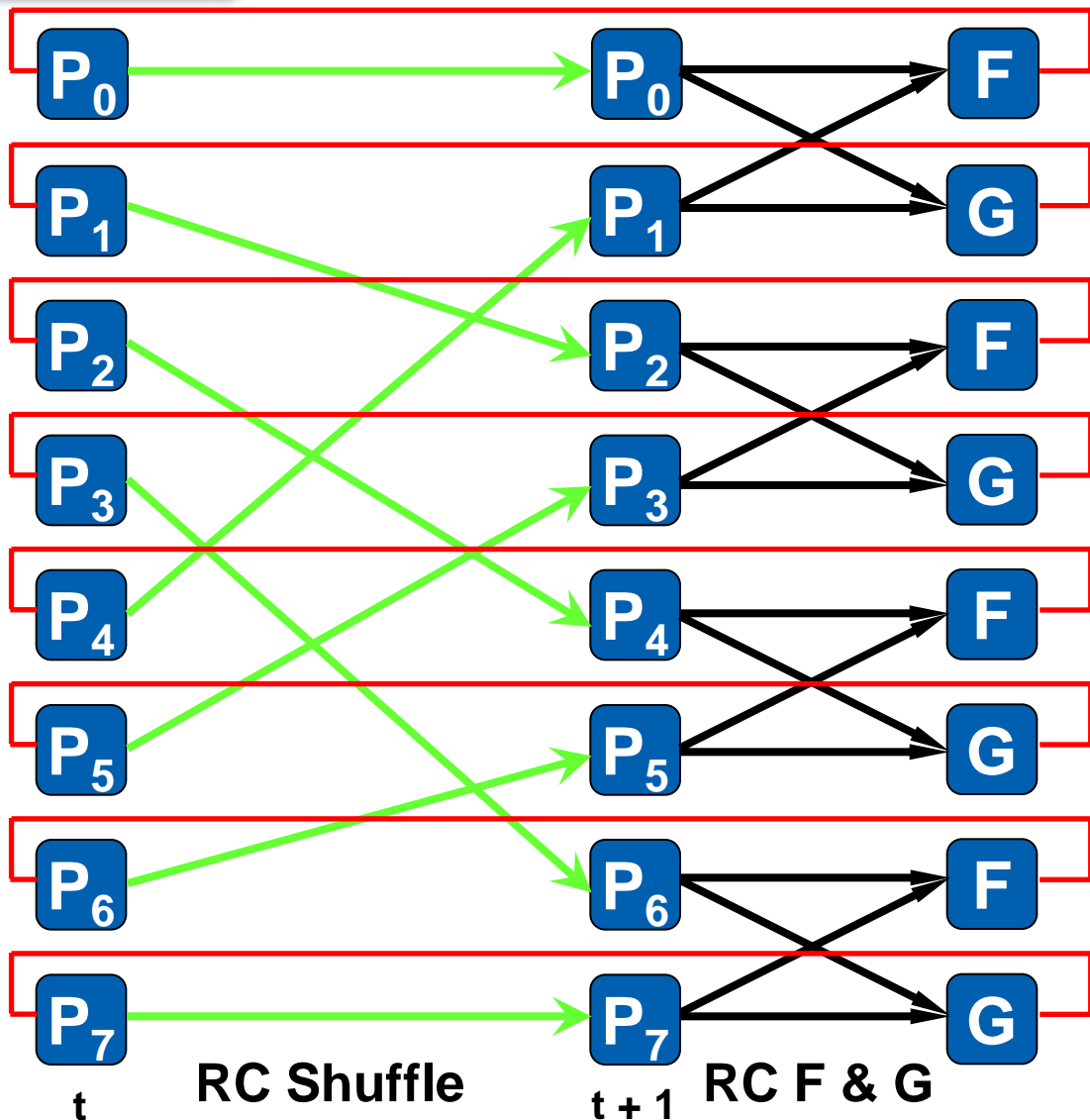


Shuffle – 8 Procesoare





Shuffle – 8 Procesoare



- Doua RC:
 - Shuffle
 - Adiacenta (pt F & G):
 - F e pe $Y_i = \text{pare}$
 - G e pe $Y_{i+1} = \text{impare}$
 - $\log_2 n$ iteratii
- ```
for i = 1 to log2n
 shuffle[y]
 Yi = F[Yi, Yi+1]
 Yi+1 = G[Yi, Yi+1]
end i loop
```

• Utilizare: sortari, FFT, transpuneri, functii cu operatii recurente



# Shuffle – Aplicatii

5

- Pentru sortari:
  - F calculeaza maximul celor doi operanzi
  - G calculeaza minimul celor doi operanzi
- Pentru FFT:
  - F produce suma ponderata
  - G produce diferenta ponderata
  - Ponderile sunt calculate dinamic la fiecare pas al iteratiei
- Pentru transpunerea matricelor:
  - Organizarea directa a matricei
  - $F = Y_i$
  - $G = Y_{i+1}$

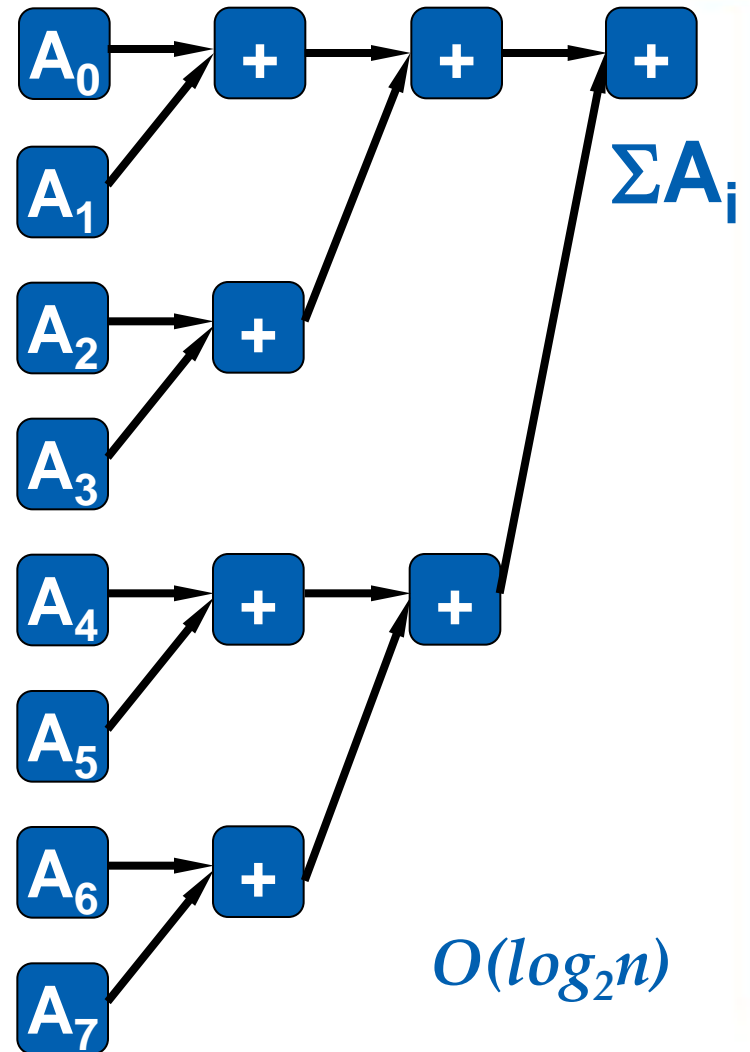


# Shuffle – Aplicatii Iterative

- Suma elementelor unui vector
- Daca vectorul e intr-o singura memorie → secvential:

```
sum = 0
for i = 0 to n-1
 sum += ai
end i loop $O(n)$
```

- Se poate reduce  $O(n)$  pe SIMD?
- Distribuim vectorul intre memorii
- Probleme:
  - Incarcarea neechilibrata a procs
  - La fiecare pas trebuie schimbata RC in mod dinamic





# Shuffle – Aplicatii Iterative (cont)

- Sunt necesare
  - Comunicare liniara
  - RC statica
- Reconfigurarea conexiunilor e mai simpla
- Dezavantaj: este in continuare necesara reconfigurarea la **fiecare** pas!
- Complexitate:  $O(\log_2 n)$

