Evaluation de Performance de Réseaux ATM: Etude et Perspectives en Utilisant la Simulation Distribuée

C.D. Pham et S. Fdida

Laboratoire MASI
Université Pierre et Marie Curie
4 place Jussieu 75252 Paris Cedex 05
{pham,fdida}@masi.ibp.fr

Mardi 15 octobre 1996
CFIP'96, Rabat, Maroc

Ce travail est en partie financé par la DRET

Evaluation de Performances des Réseaux

Nouvelles Technologies

- Problèmes nouveaux
- Systèmes de plus en plus complexes
- Taille de plus en plus grande

Besoin d'Outils Adaptés

- Méthodes analytiques trop limitées
- Simulations très consommatrices de puissance

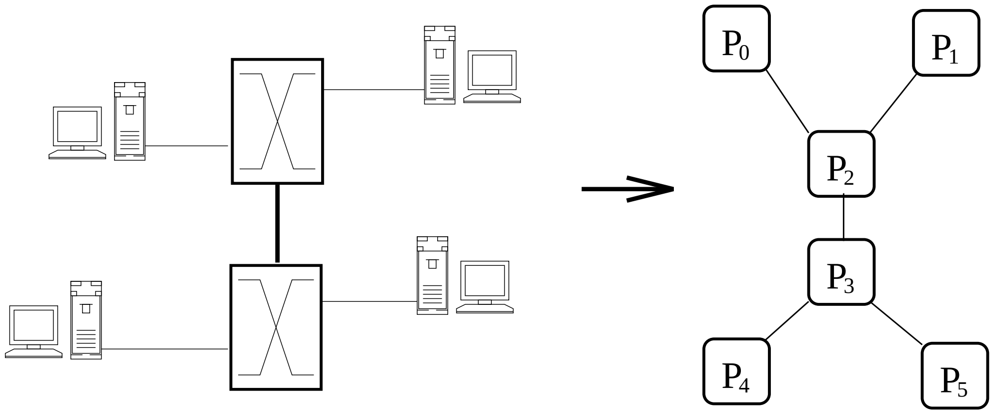
Situation Actuelle

- Simulations séquentielles trop limitées
- Puissance disponible sur architectures parallèles

☞ **Simulation distribuée**

Simulation Parallèle et le Problème de la Synchronisation

Processus Logique et Partitionnement Spacial



Les listes et les horloges sont distribuées.

⬇

La causalité est requise pour obtenir des résultats corrects.

⬇

Un algorithme de synchronisation est nécessaire.

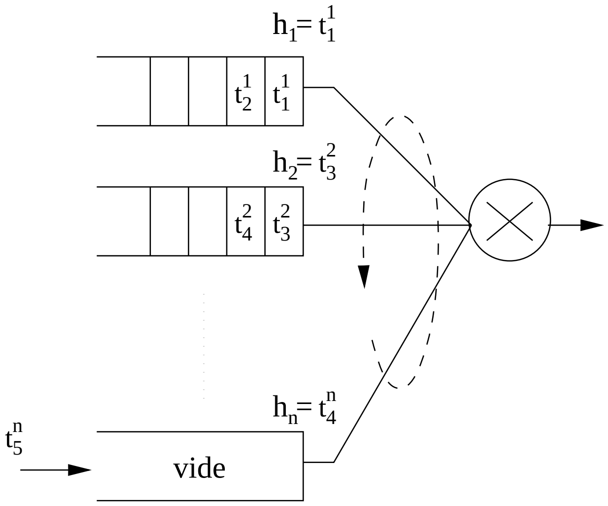
⬇

Ces mécanismes de synchonisation sont classés en 2 catégories: **<u>conservatrice</u>** et **<u>optimiste</u>**.

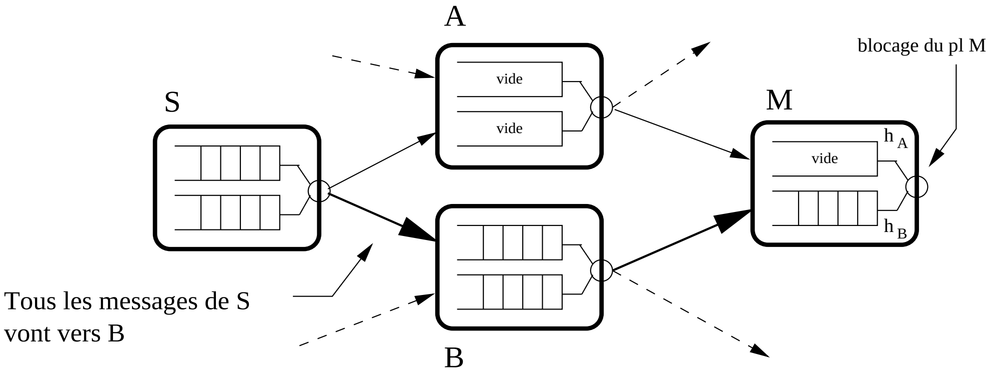
Approches Conservatrices

Une approche conservatrice (Chandy-Misra-Bryant) évite les fautes en bloquant les PLs.

Algorithme conservateur

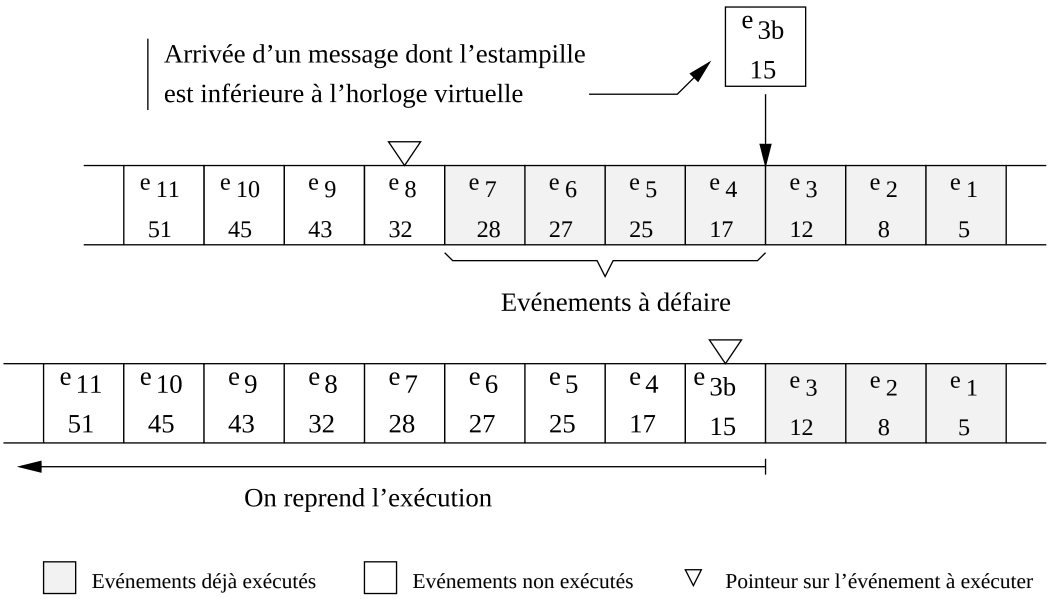


Inter-blocage $\rightarrow$ *null-messages* et *lookahead*



Approches Optimistes

Les méthodes optimistes (Time Warp) utilisent un mécanisme de *rollback*.



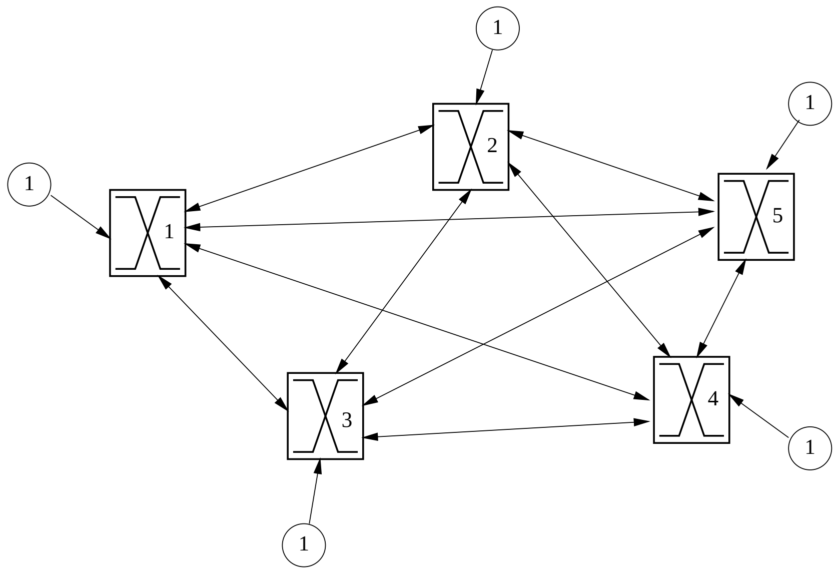
Besoin de sauvegarder l'état et les *anti-messages* pour:

- Restaurer l'état du PL,
- Envoyer les anti-messages,
- Exécuter de nouveau les événements dans l'ordre.

Modèle de réseaux

Réseau complètement maillé

Simulation au niveau cellule

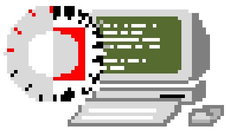


Configuration pour $N_s = 5$

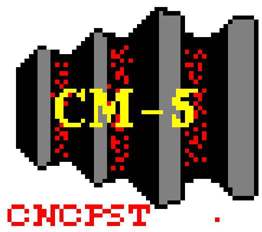
- Grande Localité
- Densité de messages $= n_c$
- Faible granularité

Environnement de test

2 CONFIGURATIONS SONT UTILISÉES.

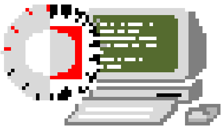


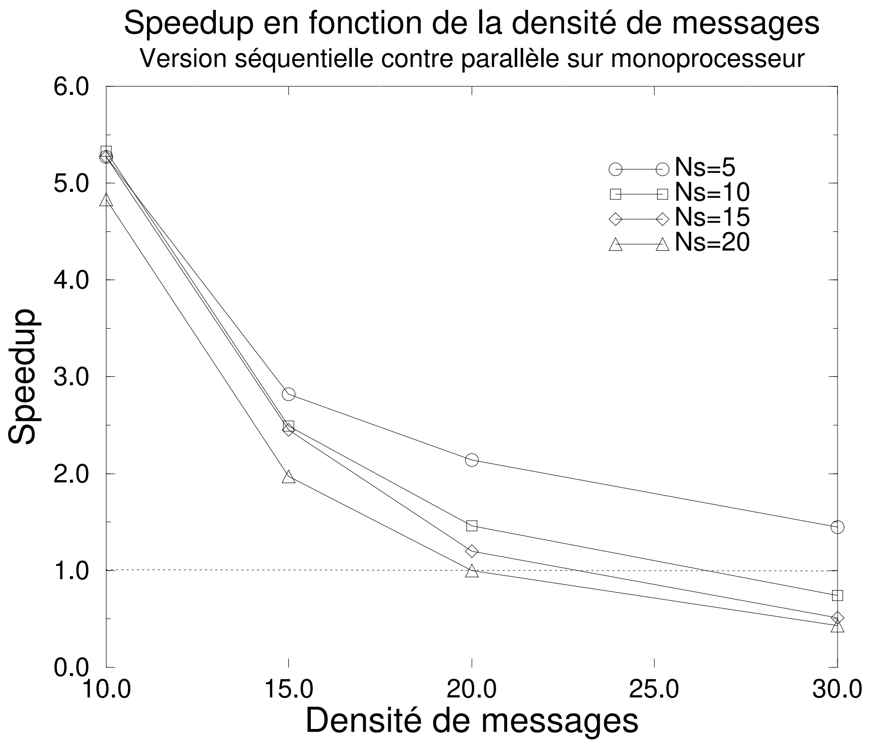
SS20 avec 64MB de mémoire - librairie PVM



CM-5 avec 32 processeurs - librairie CM/PVM

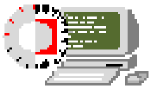
Implantation conservatrice

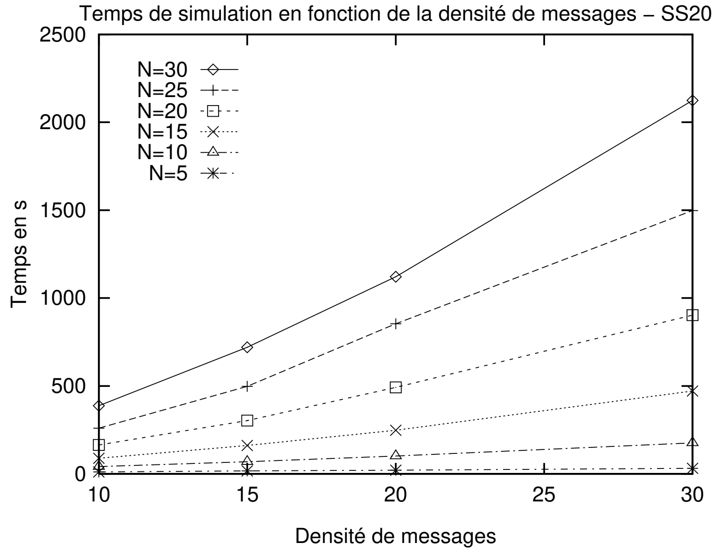


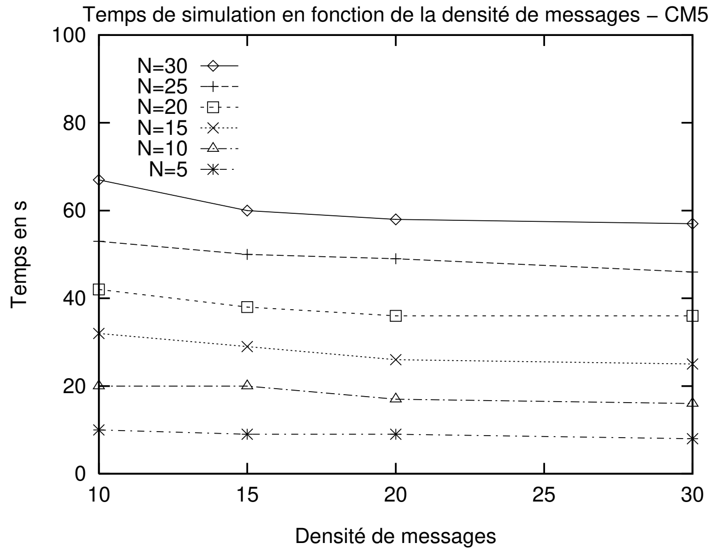


Séquentielle vs Parallèle monoprocesseur

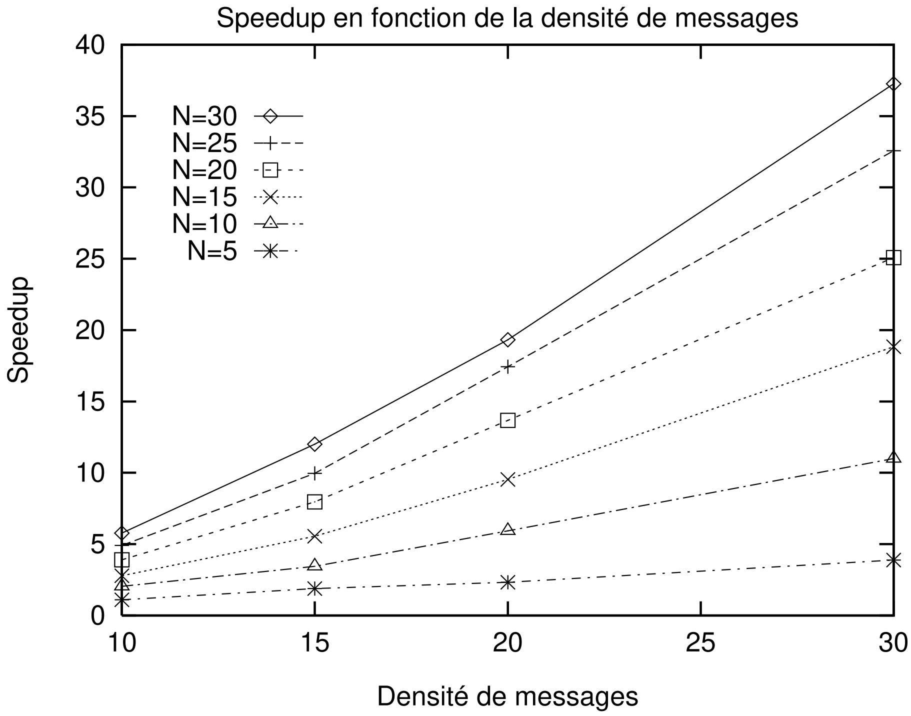
Temps de simulation







Speedup



Speedup en fonction de la densité de messages

Approche optimiste pure

- Grande localité
  - Les fautes impliquent beaucoup de PLs
- Grande densité de messages
  - Synchronisation fréquente
  - Grand nombre de fautes temporelles
- Faible granularité
  - Les fautes se propagent très rapidement
  - La "profondeur" des fautes peut être grande

☞ **Peu performante en raison du coût des *rollbacks***

Relâchement des contraintes

Maintenir la causalité est très coûteux

Les approches optimistes:

- sont très limitées par le coût du *rollback*,
- peuvent présenter des phénomènes de *rollbacks* en cascade.

Si le coût du *rollback* peut être fortement réduit, alors une approche optimiste est toujours optimale.

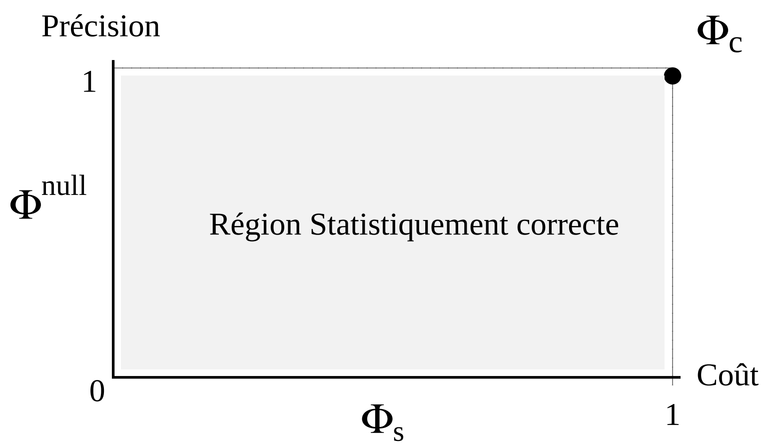
L'approche statistiquement fiable relâche certaines contraintes de synchronisation.

Relâchement suivant une fonction de contrôle

Une fonction de contrôle decide quand le mécanisme du *rollback* doit être utilisé.

⬇

Il est possible de couvrir un continuum de précision selon le coût du contrôle



- Pas de contrôle $\rightarrow$ coût nul.
- Contrôle basé sur la distance $\rightarrow$ coût dépendant du seuil.

Estimations locales

CAS: pas de contrôle

- Estimations locales comme substitut au *rollback*.
- Des erreurs sont acceptées pour obtenir des bons speedups.
- Les valeurs moyennes devraient être préservées.

- Inefficace si le nombre de fautes est trop élevé.

- Il faut réduire le nombre de fautes.

Calcul des estimations locales

- Soit L_s la liste des états sauvegardés
- Soit $S_i = (o_1^i, o_2^i, last_1^i, last_2^i, n_r^i, n_s^i)$ le $i$ème état sauvegardé. S_i a une estampille T_i.

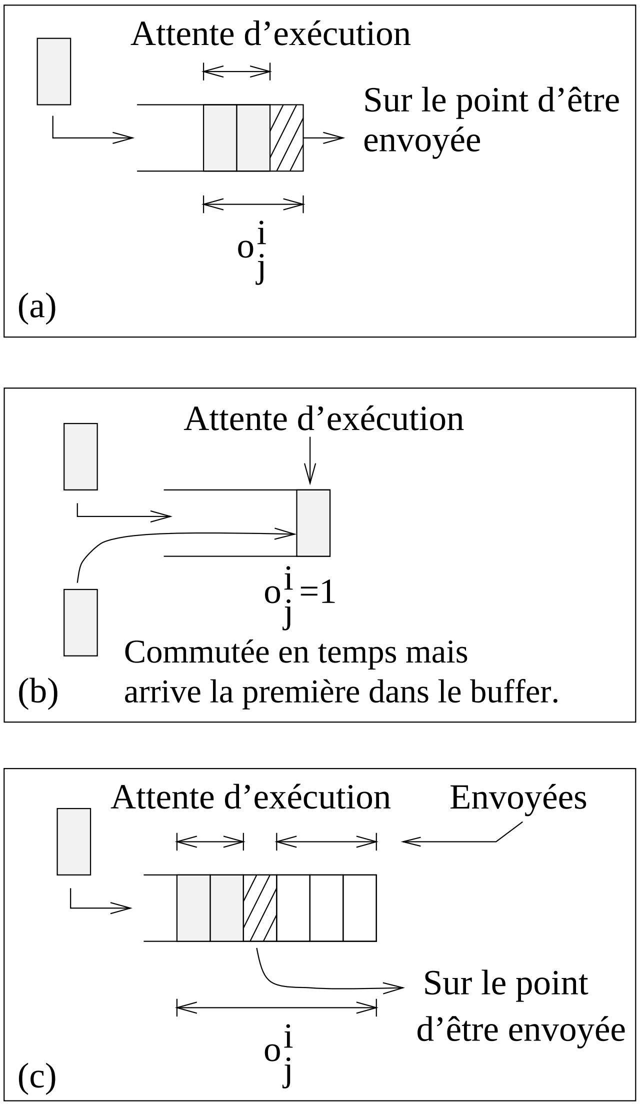
Algorithme d'estimation

⬇

1. Récupère le premier état avant t de la liste L_s.
2. Détermine la sortie j, calcule la date de départ t_d de la nouvelle cellule à partir du temps d'attente puis programme un événement $< SEND, t_d >$ pour lui-même.
3. Insert le nouvel état dans la liste juste avant l'état S_i.
4. Réévalue tous les états S_k avec $k > i + 1$.

Calcul de t_d

- Etape 2 $\begin{cases} \text{if } t = last^i_j \text{ then } t_w = max\{o^i_j - 1, 1\} \\ \text{else } t_w = max\{o^i_j - 1 - (t - T_i), 0\} \\ t_d = t + t_w + 1 + 1 \end{cases}$



Nouvel état et réévaluation

Nouvel état S_{i+1}

- Etape 3 $\left\{\begin{array}{l} o_j^{i+1} = max\{o_j^i - (t - T_i), 0\} + 1 \\ last_j^{i+1} = t \\ n_r^{i+1} = n_r^i + 1 \\ n_s^{i+1} = n_s^i + min\{o_j^i, t - T_i\} \end{array}\right.$

Réévaluation des états S_k avec $k > i+1$

- Etape 4 $\left\{\begin{array}{l} \text{if } T_k - t < o_j^{i+1} \text{ then } o_j^k = o_j^k + 1 \\ \quad \text{else } o_j^k \text{ is unchanged} \\ last_j^k \text{ is unchanged} \\ n_r^k = n_r^k + 1 \\ \text{if } t_d < T_k \text{ then } n_s^k = n_s^k + 1 \\ \quad \text{else } n_s^k \text{ is unchanged} \end{array}\right.$

Stratégies de blocage

- ☐ Blocage: un moyen pour réduire la probabilité de fautes.
- ☐ Deux classes de stratégies: l'une basée sur le temps et l'autre basée sur la longueur.

Basée sur le temps

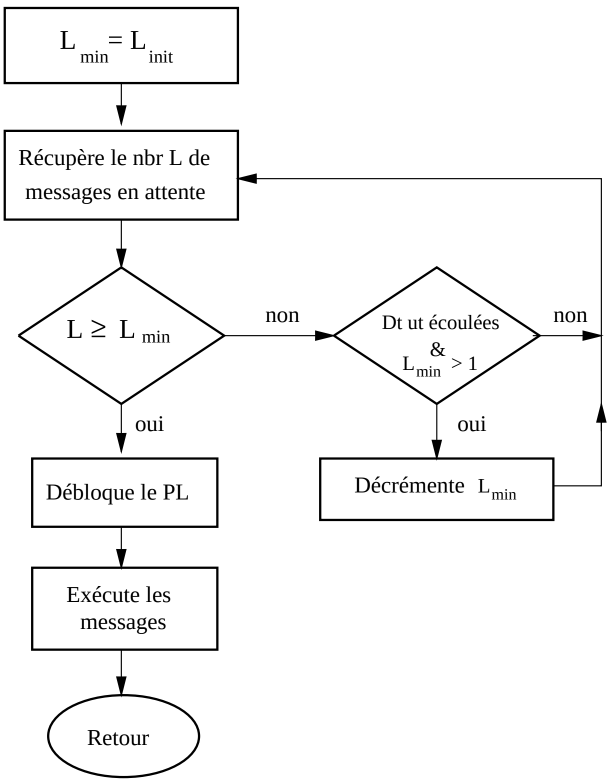
- paramétrage continu.
- réduction du problème de mauvaise répartition des charges.

Basée sur la longueur

- approche plus naturelle.
- le coût du blocage peut être réduit.

Blocage basé sur la longueur de file

- Approche préventive basée sur la longueur de la file d'attente.
- Un seuil L_{min} est defini.



Un blocage peut se produire.

L_{min} est décrémenté à chaque D_t ut.

Valeurs pour L_{init} et D_t

2 SOLUTIONS

L_{init} petit and D_t grand

- si les messages arrivent rapidement
  → la condition est satisfaite rapidement.

- si les messages arrivent lentement
  → beaucoup de temps perdu.

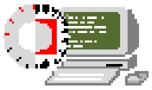
L_{init} grand and D_t petit

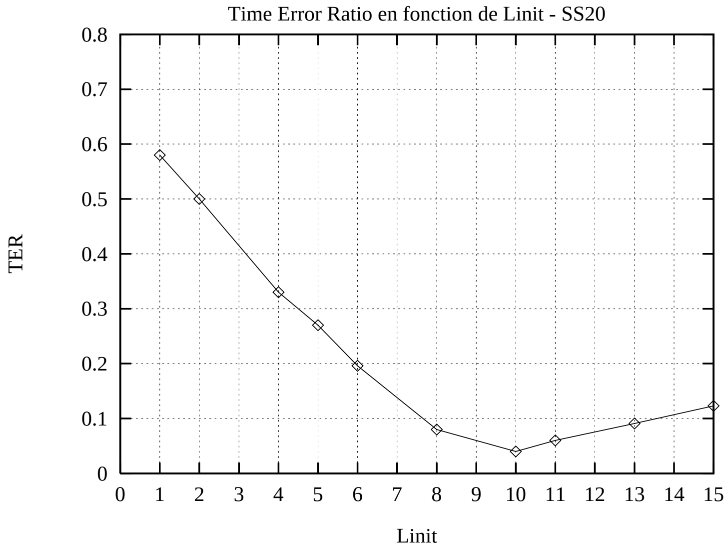
- si les messages arrivent rapidement
  → la condition est satisfaite rapidement.

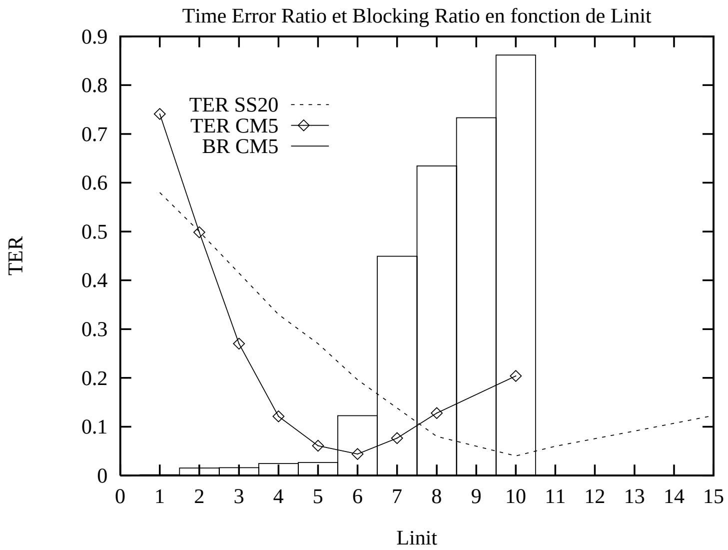
- si les messages arrivent lentement
  → L_{init} décroît rapidement.

☞ La seconde solution apparaît plus adaptative.

TER en fonction de L_{init}







Limitation de l'optimisme et Estimations locales

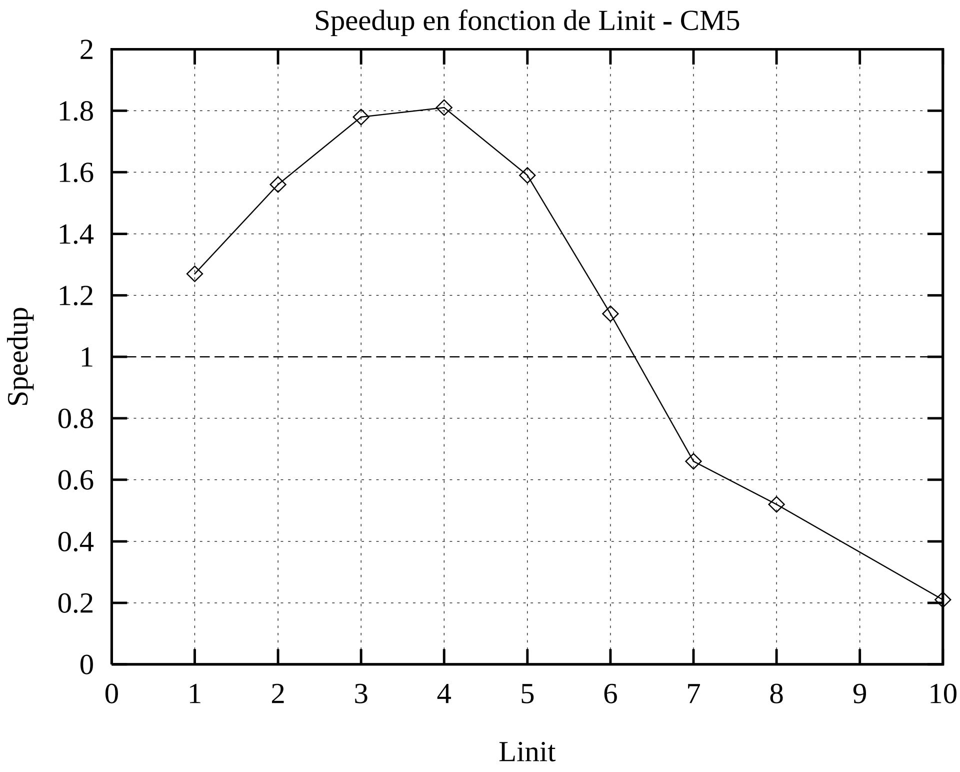
MOINS DE FAUTES POUR UNE MEILLEURE PRÉCISION

$\Downarrow$

L_{init}	TER	L	% d'erreur
6	0.044	1.27	+1.56%
CMB	0	1.25	0%
5	0.061	1.22	-1.69%
4	0.121	1.20	-3.72%
3	0.270	1.07	-13.6%
2	0.498	0.93	-25.6%
1	0.740	0.77	-33.88%

% d'erreur et TER en fonction de L_{init}

Speedup par rapport à une méthode conservatrice



Conclusions

- La simulation distribuée permet de réduire considérablement le temps de simulation.
- L'utilisation des estimations locales peut être inefficace si le nombre de fautes est important.
- Une stratégie de blocage basée sur la longueur de la file réduit considérablement le taux de fautes.
- Cette stratégie donne de meilleurs résultats lorsque les messages arrivent rapidement.
- En combinant blocage et estimations, on peut obtenir à la fois une bonne précision et un speedup accru.

Perspectives

- Améliorer la précision des estimations.
- Etudier l'influence de la topologie sur le speedup.
- Approfondir les études sur la stratégie de blocage.
- Extension à des cas réels (routage).