Gestion hybride de la mémoire dans les systèmes Java temps-réel

Nicolas Berthier

Laboratoire VERIMAG

20 septembre 2007

Plan

1. **Contexte**
   - Contexte général
   - Approche de l'équipe
   - Problème à résoudre
2. **Contributions**
   - Approche proposée
   - Implantation
3. **Expérimentations**
   - Plus de programmes sont gérés...
   - ... mais certains restent incompatibles
4. **Conclusion**

Plan

1. **Contexte**
   - Contexte général
   - Approche de l'équipe
   - Problème à résoudre
2. **Contributions**
   - Approche proposée
   - Implantation
3. **Expérimentations**
   - Plus de programmes sont gérés...
   - ... mais certains restent incompatibles
4. **Conclusion**

Contexte général

Cible

- systèmes embarqués temps-réel
  $\Rightarrow$ temps d'exécution prévisible
- langage Java standard
  $\Rightarrow$ gestion automatique de la mémoire dynamique

Ramasse-miettes classique

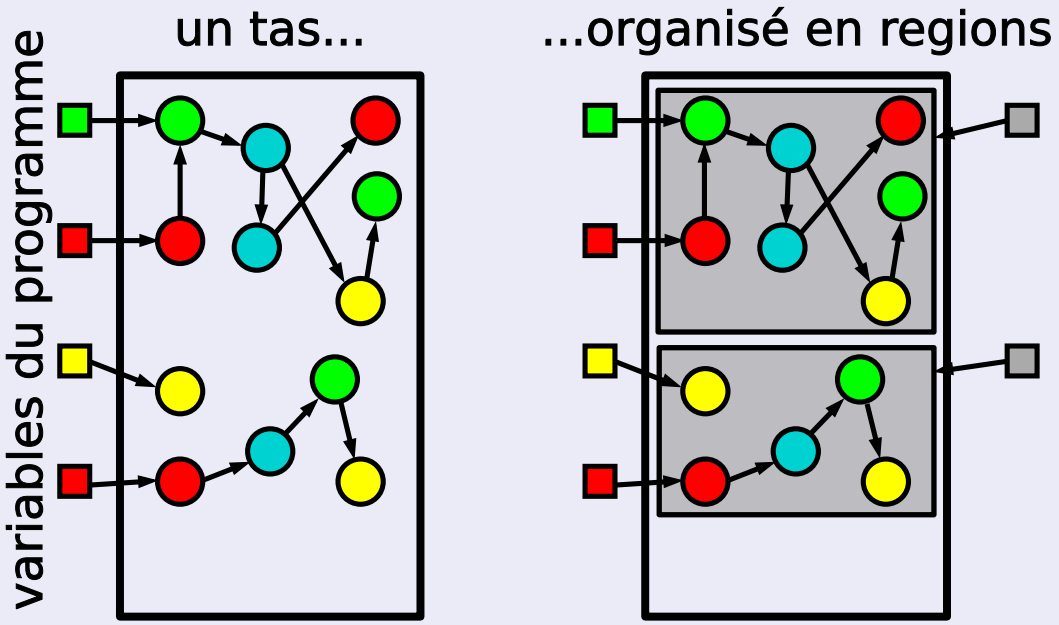
Pauses pendant l'exécution :

- d'occurences non prévisibles
- de durée non prévisible

$\Rightarrow$ inadapté aux contraintes temps-réel

Analyse statique et gestion en régions

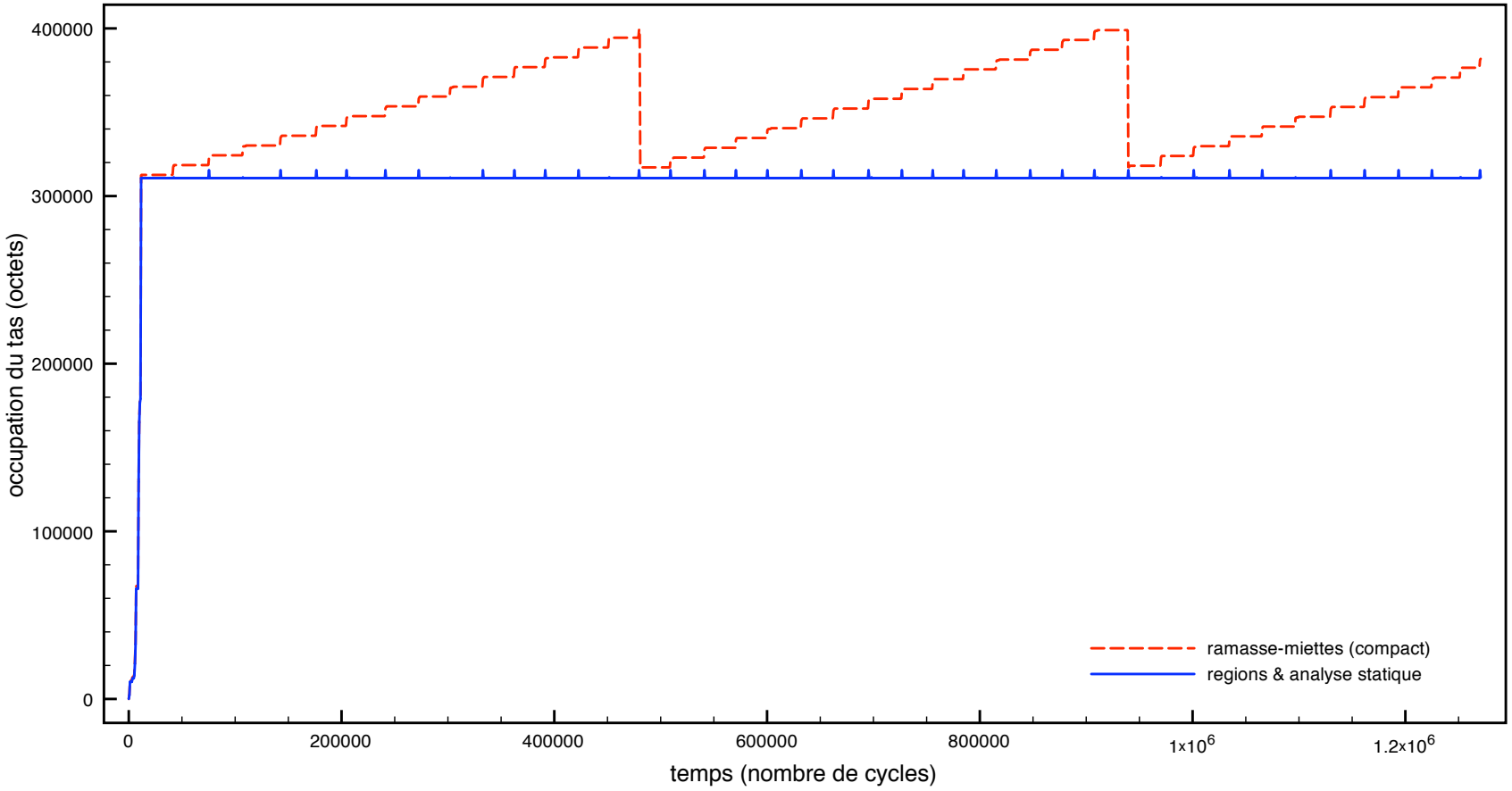
Regroupement des objets



- désallocation d'une région complète en une seule fois

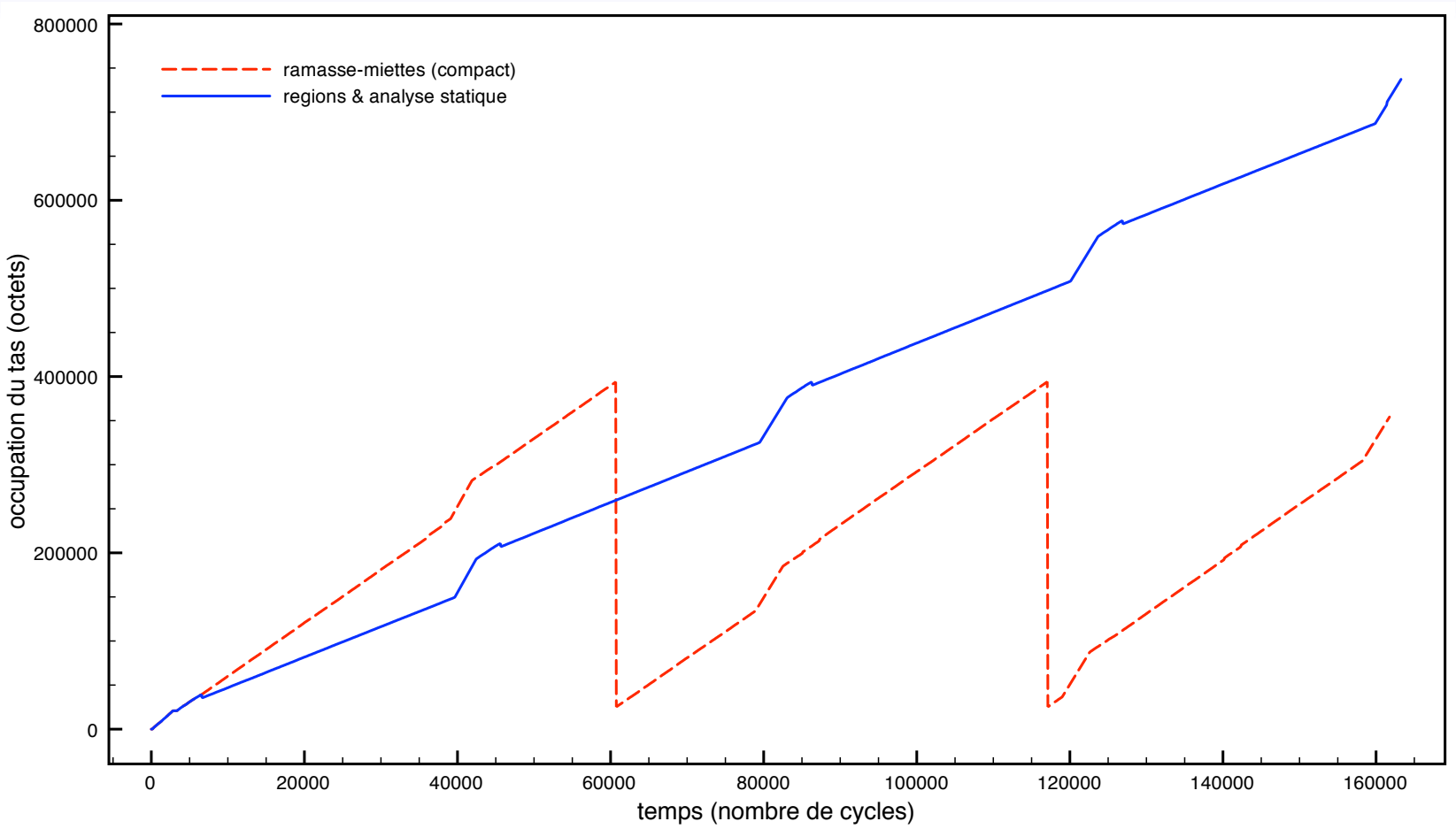
$\Rightarrow$ temps prévisibles

Problème à résoudre



$\Rightarrow$ ce modèle est efficace

Problème à résoudre



$\Rightarrow$ certains programmes posent problème

Plan

1 Contexte
- Contexte général
- Approche de l'équipe
- Problème à résoudre

2 Contributions
- Approche proposée
- Implantation

3 Expérimentations
- Plus de programmes sont gérés...
- ... mais certains restent incompatibles

4 Conclusion

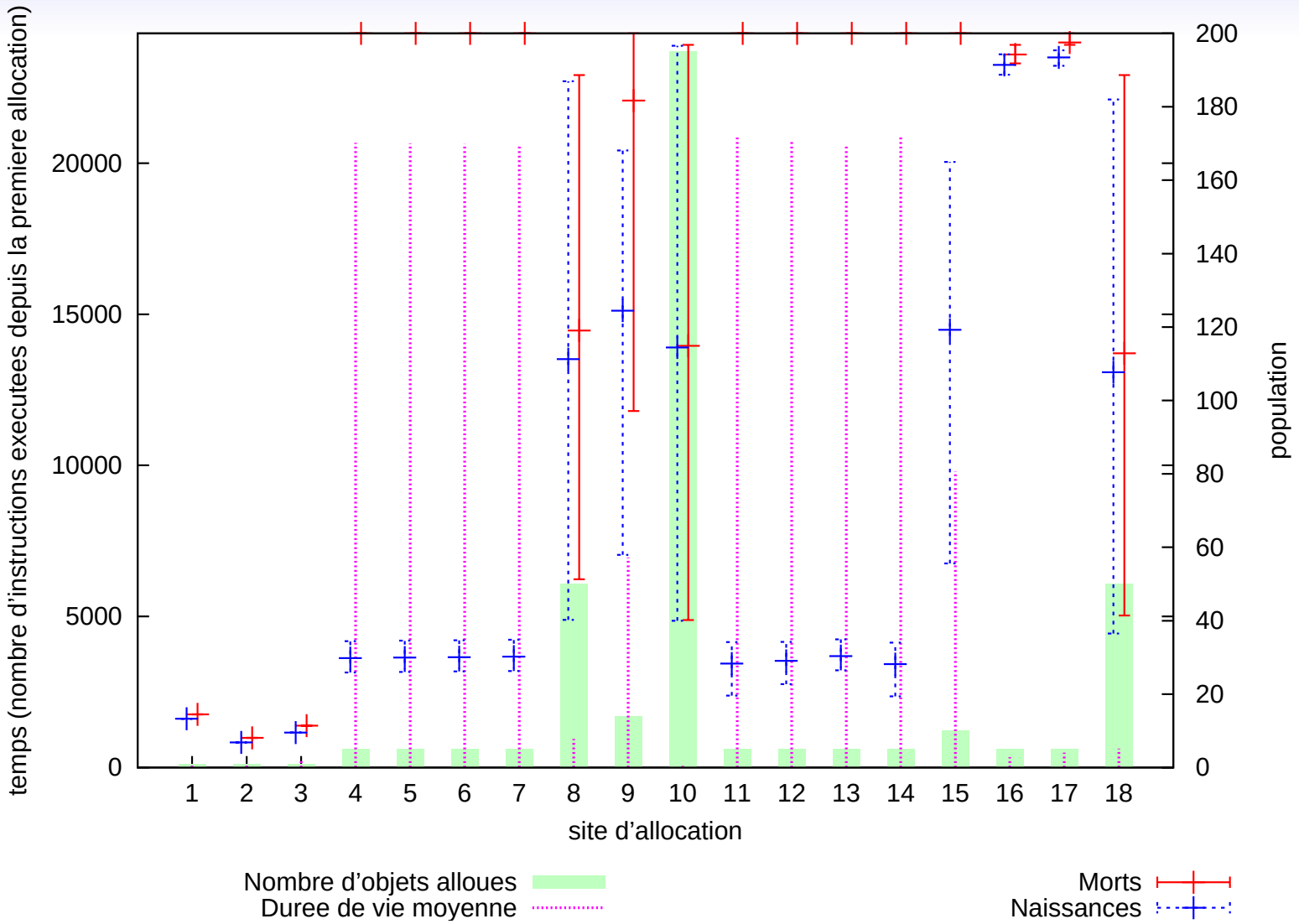
Approche proposée

Système hybride

gestion en régions secondée d'un ramasse-miettes par comptage de références

- respecte les contraintes temps-réel
- relativement simple à implémenter

Étude de la démographie des objets



Ramasse-miettes par comptage de références

Implantation

- dans la machine virtuelle JITS
- barrières en écriture : gestion d'un compteur ajouté aux objets

Combinaison avec les régions

- gère les objets potentiellement problématiques

Outil de photographie du tas

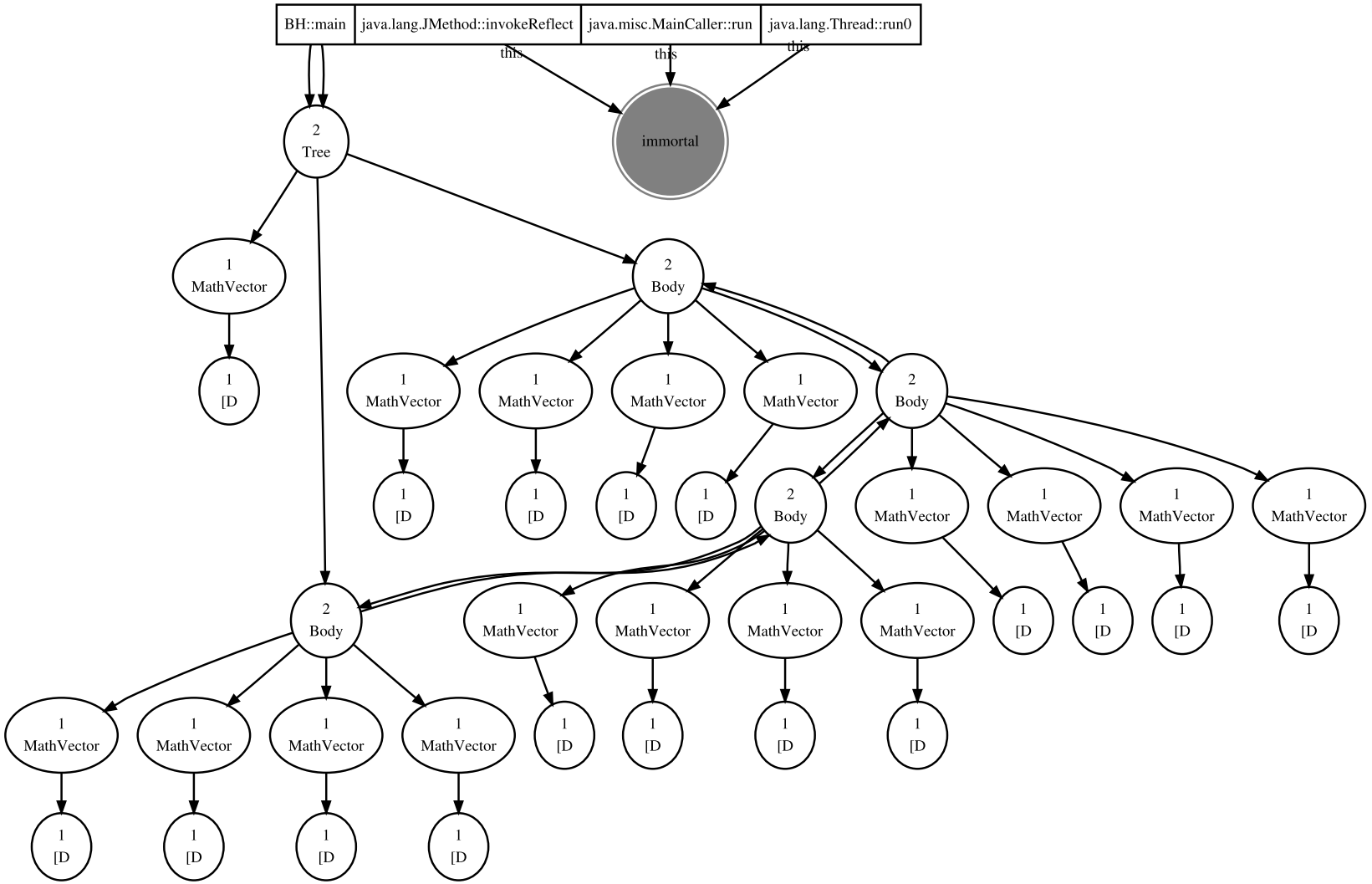
Représente

- l'état des compteurs
- les références

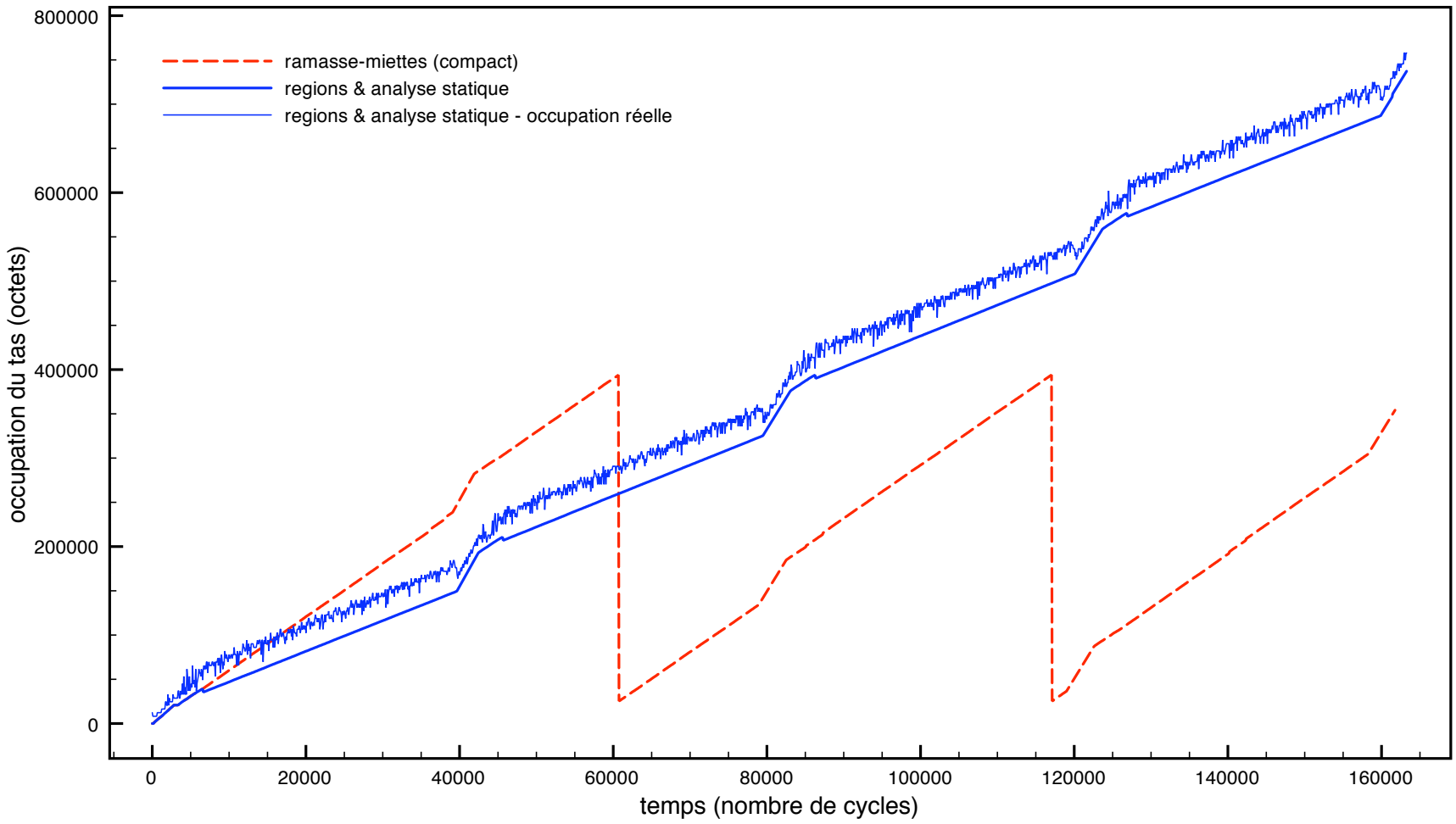
Utilité

- aide à l'implémentation et à la vérification du ramasse-miettes
- évaluation qualitative du tas (*e.g.* structure des données)

Outil de photographie du tas



Évaluation de la fragmentation interne



Plan

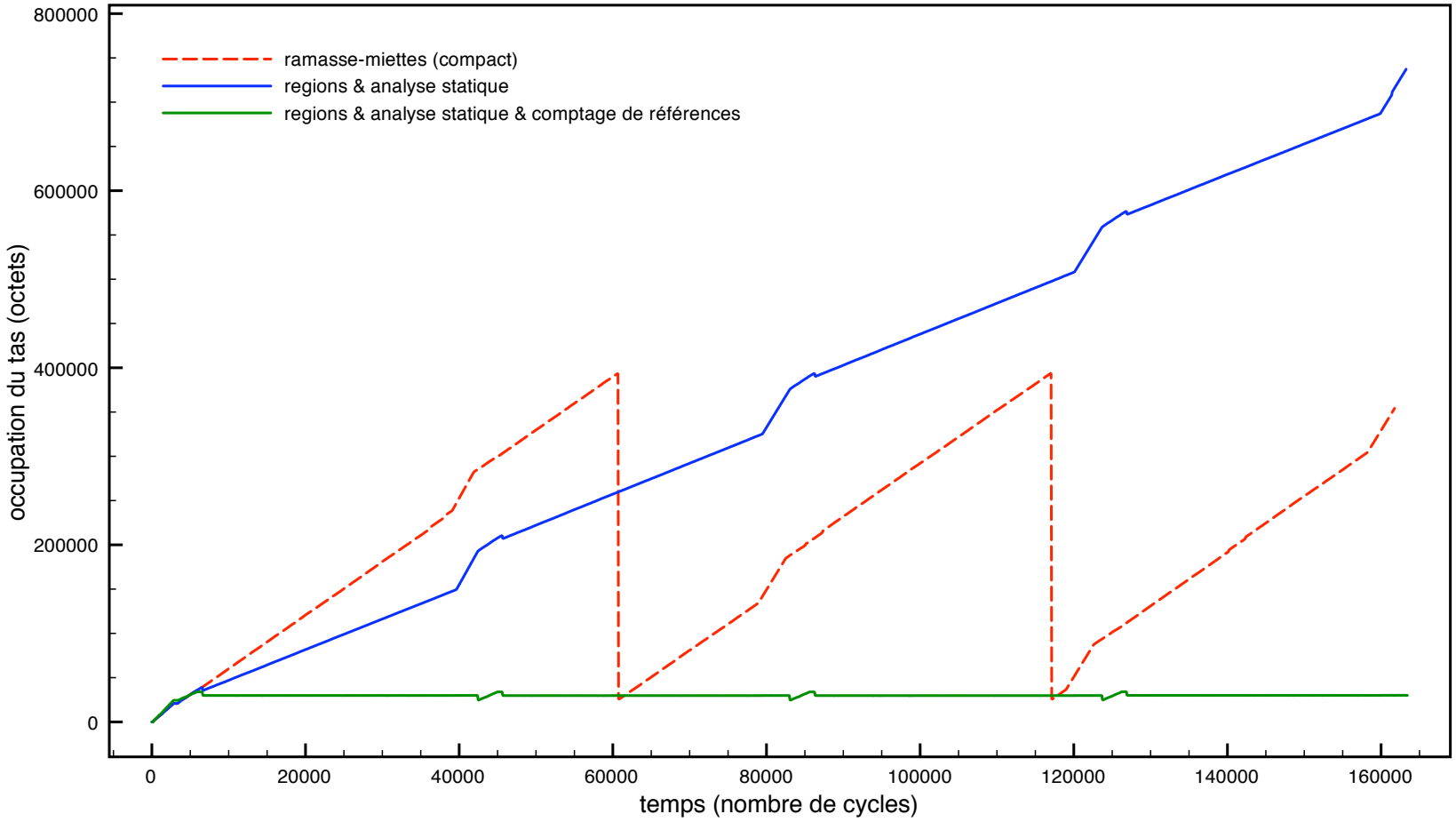
1 Contexte
- Contexte général
- Approche de l'équipe
- Problème à résoudre

2 Contributions
- Approche proposée
- Implantation

3 Expérimentations
- Plus de programmes sont gérés...
- ... mais certains restent incompatibles

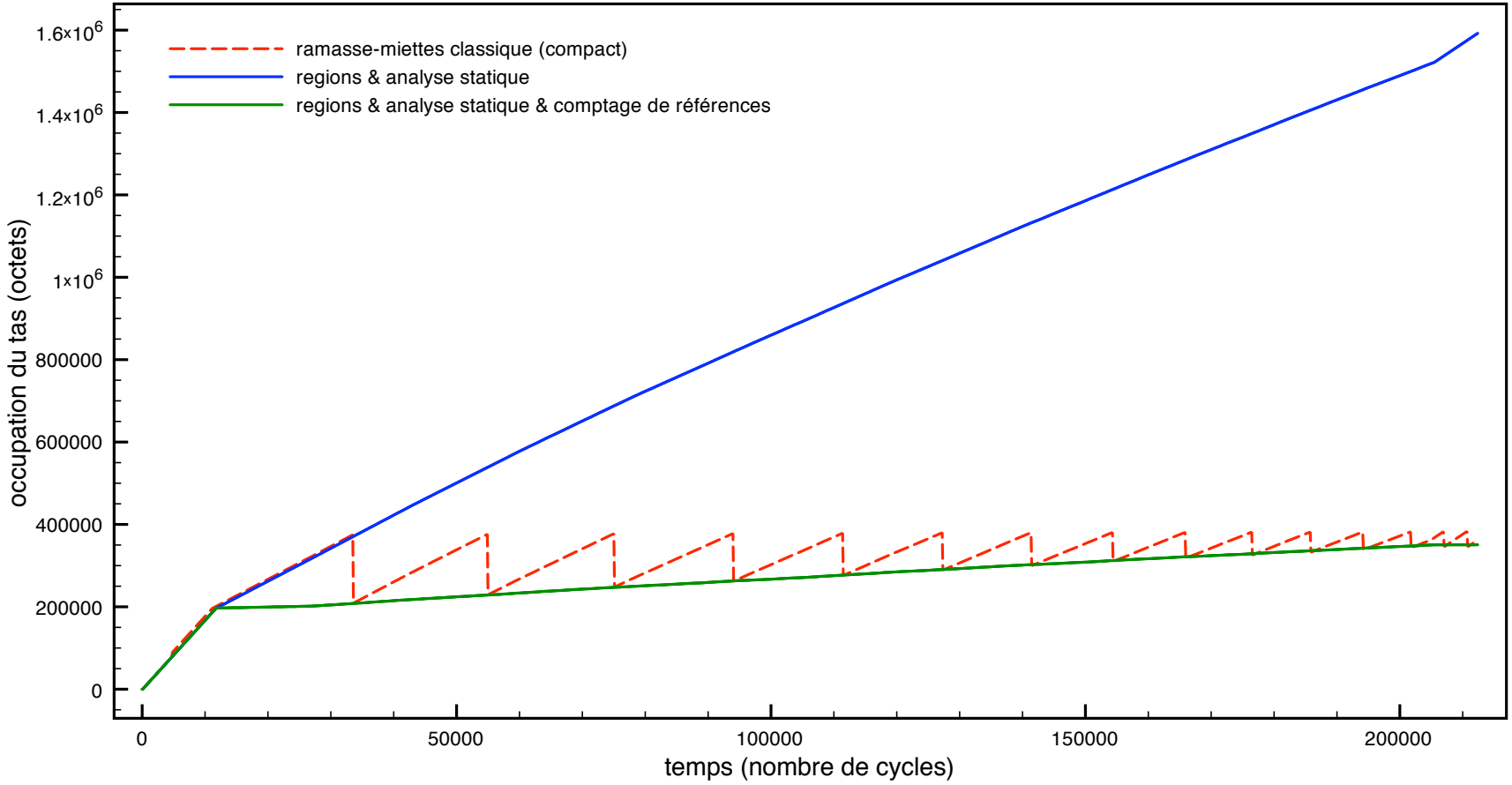
4 Conclusion

Plus de programmes sont gérés...



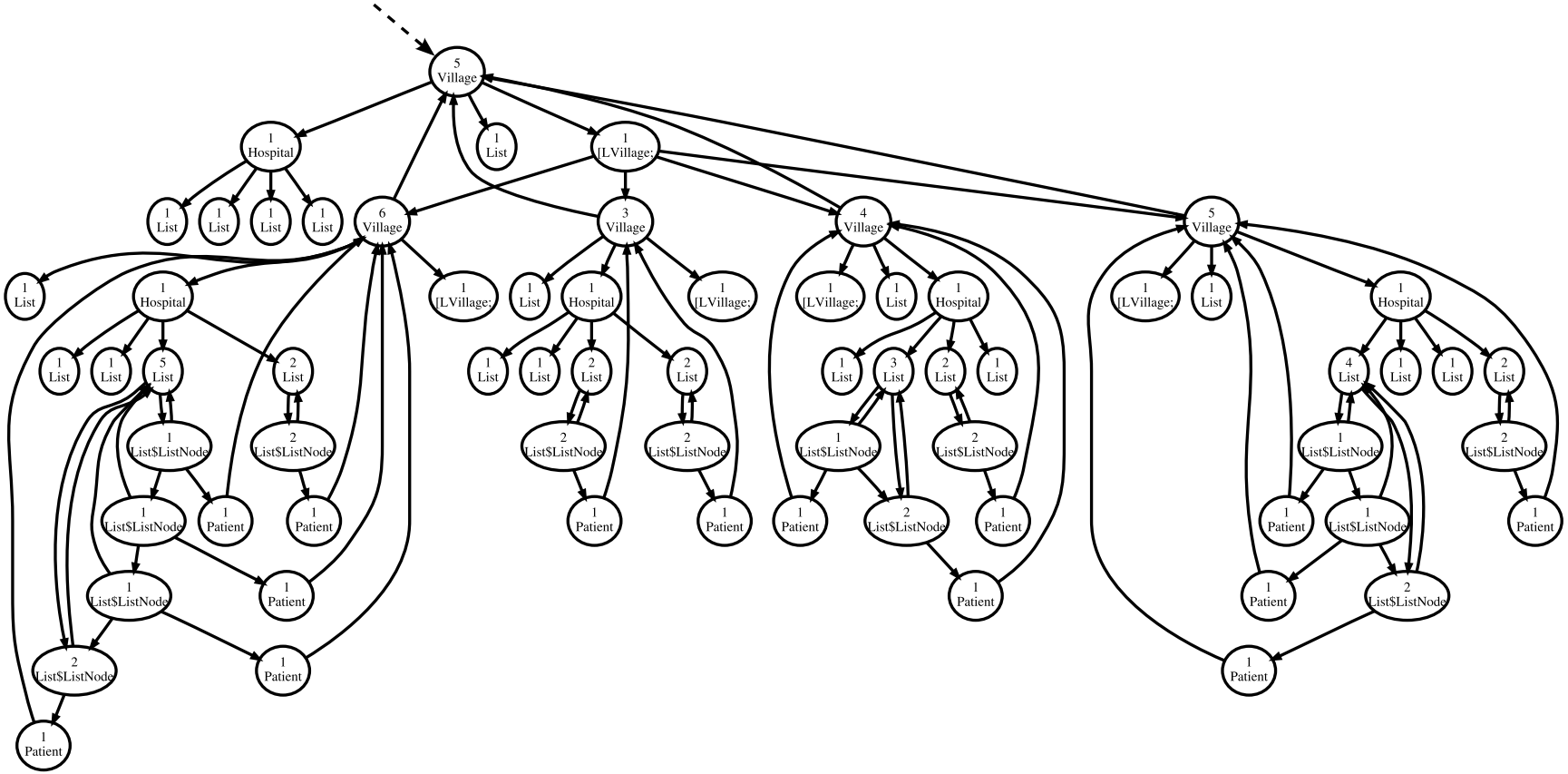
l'occupation de la mémoire reste globalement constante

... mais d'autres sont jugés incompatibles



l'occupation de la mémoire augmente encore

... mais d'autres sont jugés incompatibles



structures de données trop complexes (*e.g.* graphes irréguliers)

Plan

1 Contexte
- Contexte général
- Approche de l'équipe
- Problème à résoudre

2 Contributions
- Approche proposée
- Implantation

3 Expérimentations
- Plus de programmes sont gérés...
- ... mais certains restent incompatibles

4 Conclusion

Conclusion

Résultats

- extension de l'approche à une nouvelle classe de programmes
- identification des cas incompatibles
  - manipulation de structures trop complexes
  - usage mémoire irrégulier

Perspectives

- mécanisme de comptage de références secondé des régions pour gérer les cycles
- analyse statique pour détecter les cycles potentiels

Merci de votre attention

Des questions ?