

**THÈSE PRÉSENTÉE A L'UNIVERSITÉ D'ORLÉANS  
POUR OBTENIR LE GRADE DE  
DOCTEUR DE L'UNIVERSITÉ D'ORLÉANS**

**PAR**

**Filip Arvid JAKOBSSON**

**ÉCOLE DOCTORALE Mathématiques, Informatique, Physique Théorique et Ingénierie des  
Systèmes**

**Discipline : Informatique**

**Analyse statique des programmes BSPlib**

Soutenue Publiquement

**10:00h le 28 juin 2019 à l'Université d'Orléans**

*Salle Soutenance de Thèse (Salle 101), 1er Etage au Bâtiment S (Faculté Des Sciences), Rue de  
Chartres, 45100 Orléans*

**MEMBRES DU JURY :**

- **Kuchen, Herbert – Professeur, WWU Münster**
- **Barthou, Denis – Professeur, Université de Bordeaux**
- **Bousdira-Semmar, Wadoud – Maître de conférence, Université d'Orléans**
- **Dabrowski, Frédéric – Maître de conférence, Université d'Orléans**
- **Hains, Gaétan – Chercheur-Ingénieur, Huawei Technologies**
- **Suijlen, Wijnand – Chercheur-Ingénieur, Huawei Technologies**
- **Loulergue, Frédéric – Professeur, Northern Arizona University et Université d'Orléans**
- **Chailloux, Emmanuel – Professeur, Sorbonne Université**

**RÉSUMÉ**

La programmation parallèle consiste à utiliser des architectures à multiples unités de traitement, de manière à ce que le temps de calcul soit inversement proportionnel au nombre d'unités matérielles. Le modèle de BSP (*Bulk Synchronous Parallel*) permet de rendre le temps de calcul prévisible. BSPlib est une bibliothèque pour la programmation BSP en langage C. En BSPlib on entrelace des instructions de contrôle de la structure parallèle globale, et des instructions locales pour chaque unité de traitement. Cela permet des optimisations fines de la synchronisation, mais permet aussi l'écriture de programmes dont les calculs locaux divergent et masquent ainsi l'évolution globale du calcul BSP.

Toutefois, les programmes BSPlib réalistes sont *syntactiquement alignés*, une propriété qui garantit la convergence du flot de contrôle parallèle. Dans ce mémoire nous étudions les trois dimensions principales des programmes BSPlib du point de vue de l'alignement syntaxique: la synchronisation, le temps de calcul et la communication. D'abord nous présentons une analyse statique qui identifie les instructions syntaxiquement alignées et les utilise pour vérifier la sûreté de la synchronisation globale. Cette analyse a été implémentée en Frama-C et certifiée en Coq. Ensuite nous utilisons l'alignement syntaxique comme base d'une analyse statique du temps de calcul. Elle est fondée sur une analyse classique du coût pour les programmes séquentiels. Enfin nous définissons une condition suffisante pour la sûreté de l'enregistrement des variables. L'enregistrement en BSPlib permet la communication par accès aléatoire à la mémoire distante (DRMA) mais est sujet à des erreurs de programmation. Notre développement technique est la base d'une future analyse statique de ce mécanisme.