

Artefact* : Modeling Data Partitioning in HPC

Alix Peigue

Résumé

Ce poster présente une approche visant à décrire une donnée d'une manière indépendante du modèle de programmation utilisée. Le but des artefacts utilisés dans le papier est de montrer que cette approche est viable tant à un niveau d'architecture logicielle que de performances.

Mots-clés : HPC, StarPU, COMET, données partitionnées

1. Vue générale

1.1. Contributions

C₁ Ce poster démontre la viabilité d'une approche de description de données partitionnées unifiée sur nos cas d'évaluation

1.2. Artefacts

Artefact	Contributions	Éléments du papier	URL
A ₁	C ₁	CPU Heat Diffusion	https://gitlab.inria.fr/apeigue/prism_reprohackaton

2. Détails des artefacts

2.1. Artefact : A₁

Lien avec les contributions

Cet artefact démontre l'absence d'overhead de l'utilisation du modèle de partitionnement de données PRISM sur un cas simple (diffusion thermique sur un vecteur uniquement sur CPU)

Résultats attendus

L'overhead relatif (différence entre le temps d'exécution avec PRISM et le temps d'exécution de référence) est proche de 0%.

Temps de reproduction en minutes

Temps humain : 30 minutes, Temps machine : 30 minutes

*. Format inspiré de SC : <https://github.com/jennfshr/sc26-repro/>

Setup de l'artefact

Dépendances logicielles

- StarPU 1.4 : StarPU 1.4 doit être installé sur Grid'5000 (`pkg-config --exists starpu-1.4`).
- `g5k_bench` : `g5k_bench` est utilisé pour décrire les paramètres et étapes du benchmark (https://gitlab.inria.fr/apeigue/g5k_bench)
 - Si `uv` : installation des dépendances automatiques
 - Si `pip` : installer `python-grid5000`, `PyYAML` et `typer`
- Pour le script pour tracer les courbes
 - Si `uv` : installation des dépendances automatiques
 - Si `pip` : installer `pandas`, `matplotlib`, `numpy`

Dépendances matérielles

- Nécessite une machine sur `g5k` avec un nombre important de coeur (recommandé 32 ou plus) pour bien observer le scaling.

Exécution de l'artefact

E₁ Compiler/installer StarPU sur Grid'5000

E₂ Mettre en place les informations de connexion pour `g5k_bench` (voir README)

E₃ Lancer les benchmarks

`benchmarks/starpu_cpu_model_vector_string_scaling.yaml`

`benchmarks/starpu_cpu_ref_vector_string_scaling.yaml`

avec `g5k_bench` (les deux peuvent être faits en parallèle sur 2 noeuds du même cluster).

Le site et cluster sur lequel le benchmark est exécuté peut être réglé dans ces fichiers YAML. De même, si aucune clé SSH n'est configurée avec le Gitlab Inria sur Grid'5000, HTTPS peut être utilisé au lieu de SSH pour l'adresse du dépôt git dans les configurations des benchmarks.

E₄ Tracer les courbes avec `benchmarks/plot\_starpu\_vector.py`

E₁ → E₃ → E₄, E₁ → E₃

Commentaires sur l'analyse des résultats

Le graphe montre le résultat attendu d'un strong scaling pour la référence et le modèle et ne montre pas d'overhead significatif. La figure 1 montre le type de courbe attendue.

Annexes

A. Installation de StarPU sur Grid'5000

A.1. Installation manuelle (recommandé)

- Recupérer le code source de StarPU 1.4 en le téléchargeant depuis les releases plutôt que de clone le dépôt git, cela évite d'avoir à utiliser autotools
- Configurer la compilation avec `./configure`
 - `--disable-build-doc`
 - `--disable-build-examples`
 - `--disable-build-tests`
 - `--enable-fast`
- Configurer le nombre max de coeurs et de noeuds NUMA pour correspondre au

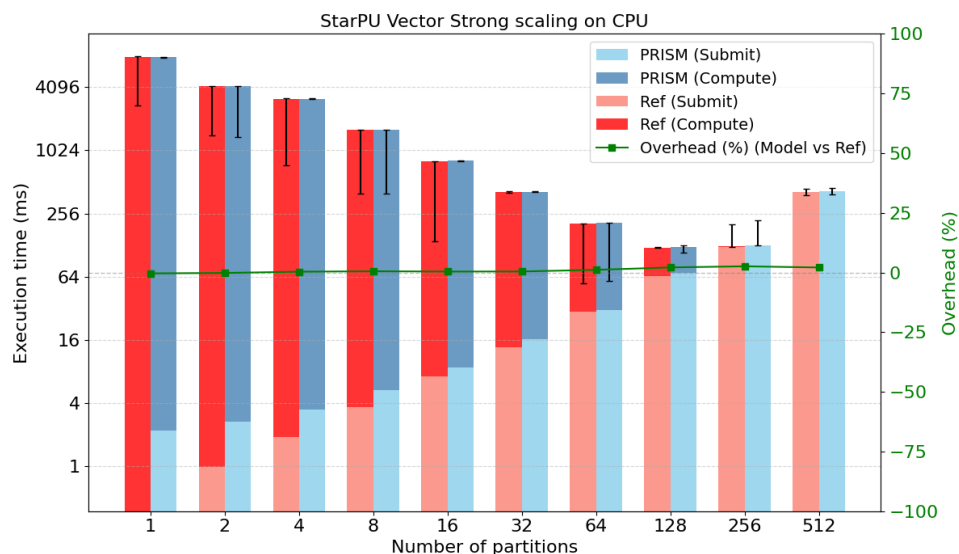


FIGURE 1 – Courbe obtenue sur un noeud chiffnot (site de lille)

moins aux spécifications du cluster sur lequel le benchmark sera effectué (par exemple pour le cluster paradoxe `--enable-maxcpus=52` et `--enable-maxnumanodes=2`)

— Configurer le dossier d'installation (recommandé `-prefix=$HOME/.local`)

— Compiler et installer `make -j install`

Si StarPU est installé dans `$HOME/.local`, il faut penser à mettre à jour les variables d'environnement (par exemple dans le `.bashrc`)

```
export LD_LIBRARY_PATH=$HOME/.local/lib:$LD_LIBRARY_PATH
export PKG_CONFIG_PATH=$HOME/.local/lib/pkgconfig:$PKG_CONFIG_PATH
```

A.2. Installation via un gestionnaire de paquets

La documentation de StarPU donne tous les gestionnaires de paquets qui peuvent être utilisés pour installer StarPU (Guix, Spack,...). Cependant, cette méthode ne permet pas forcément de configurer correctement StarPU. Dans tous les cas, ne pas utiliser le module sur Grid'5000 ou le package apt sur l'environnement de base car ces versions ne sont pas à jour.