Analyse de performance et optimisation de code

Pierre AYOUB

25 janvier 2019

INSTITUT DES SCIENCES ET
TECHNIQUES DES YVELINES

Résumé

Table des matières

1 Introduction 4

2 Analyse du code 4

3 Protocole expérimental 6

3.1 Théorie 6

3.2 Pratique 6

4 Optimisations 6

4.1 Déroulage de boucle 6

4.2 Vectorisation 6

4.3 Inlining 6

5 Conclusion 6

1 Introduction

2 Analyse du code

fluid.cflow

```
+-advect <void  at fluid.c:183>
  +-build_index <int  at fluid.c:10>
  \-setBoundry <void  at fluid.c:52>
    \-build_index <int  at fluid.c:10>
+-buoyancy <float  at fluid.c:143>
  \-build_index <int  at fluid.c:10>
+-c_densitySolver <void  at fluid.c:319>
  +-addSource <void  at fluid.c:37>
  +-swap <void  at fluid.c:20>
  +-diffuse <void  at fluid.c:117>
   \-linearSolver <void  at fluid.c:92>
     +-build_index <int  at fluid.c:10>
     \-setBoundry <void  at fluid.c:52>
       \-build_index <int  at fluid.c:10>
  \-advect <void  at fluid.c:183>
    +-build_index <int  at fluid.c:10>
    \-setBoundry <void  at fluid.c:52>
      \-build_index <int  at fluid.c:10>
+-c_velocitySolver <void  at fluid.c:341>
  +-addSource <void  at fluid.c:37>
  +-vorticityConfinement <void  at fluid.c:232>
   +-build_index <int  at fluid.c:10>
   +-abs
   +-calculate_curl <float  at fluid.c:132>
    \-build_index <int  at fluid.c:10>
   \-sqrt
  +-buoyancy <float  at fluid.c:143>
   \-build_index <int  at fluid.c:10>
  +-swap <void  at fluid.c:20>
  +-diffuse <void  at fluid.c:117>
   \-linearSolver <void  at fluid.c:92>
     +-build_index <int  at fluid.c:10>
     \-setBoundry <void  at fluid.c:52>
       \-build_index <int  at fluid.c:10>
  +-project <void  at fluid.c:288>
   +-build_index <int  at fluid.c:10>
   +-setBoundry <void  at fluid.c:52>
    \-build_index <int  at fluid.c:10>
   \-linearSolver <void  at fluid.c:92>
     +-build_index <int  at fluid.c:10>
     \-setBoundry <void  at fluid.c:52>
       \-build_index <int  at fluid.c:10>
  \-advect <void  at fluid.c:183>
    +-build_index <int  at fluid.c:10>
    \-setBoundry <void  at fluid.c:52>
      \-build_index <int  at fluid.c:10>
+-calculate_curl <float  at fluid.c:132>
  \-build_index <int  at fluid.c:10>
```

```
+-diffuse <void  at fluid.c:117>
  \-linearSolver <void  at fluid.c:92>
    +-build_index <int  at fluid.c:10>
    \-setBoundry <void  at fluid.c:52>
      \-build_index <int  at fluid.c:10>
+-linearSolver <void  at fluid.c:92>
  +-build_index <int  at fluid.c:10>
  \-setBoundry <void  at fluid.c:52>
    \-build_index <int  at fluid.c:10>
+-project <void  at fluid.c:288>
  +-build_index <int  at fluid.c:10>
  +-setBoundry <void  at fluid.c:52>
   \-build_index <int  at fluid.c:10>
  \-linearSolver <void  at fluid.c:92>
    +-build_index <int  at fluid.c:10>
    \-setBoundry <void  at fluid.c:52>
      \-build_index <int  at fluid.c:10>
+-setBoundry <void  at fluid.c:52>
  \-build_index <int  at fluid.c:10>
+-vorticityConfinement <void  at fluid.c:232>
  +-build_index <int  at fluid.c:10>
  +-abs
  +-calculate_curl <float  at fluid.c:132>
   \-build_index <int  at fluid.c:10>
  \-sqrt
```

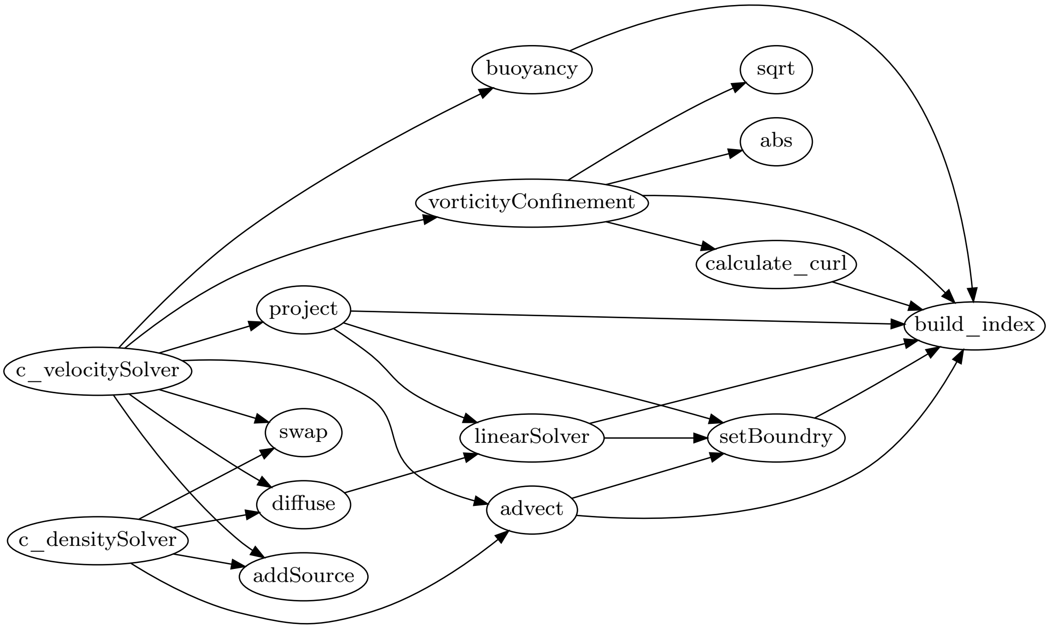

Figure 1 – Graphe d'appel du fichier *fluid.c*

3 Protocole expérimental

3.1 Théorie

3.2 Pratique

4 Optimisations

4.1 Déroulage de boucle

4.2 Vectorisation

4.3 Inlining

5 Conclusion

CPU Central Processing Unit, processeur central de l'ordinateur

RAM Random Access Memory

HT Hyper-Threading