



université PARIS-SACLAY

Rapport de projet Numérique

Domaine : Calcul Haute Performance, Simulation
Filière : Calcul Haute Performance, Simulation

Thème

**Utilisation et analyse critique des rapports MAQAO
pour optimiser des mini-apps**

Présenté par :

M^r Anis MEHIDI
M^r Arezki Takfarines HAMIDANI
M^{me} Katia MOALI
M^r Madjid BOUZOURENE
M^{me} Sylia BENBACHIR

Proposé et encadré par :

M^r Cedric VALENCI
M^r Emmanuel OSERET

Janvier 2022

Table des matières

1	Introduction	3
2	Outil d'Analyse MAQAO	4
2.1	Definition	4
2.2	Les modules de base	4
2.2.1	LProf	4
2.2.2	CQA	4
2.2.3	ONE View	5

Table des figures

1 Introduction

Dans le monde de la programmation informatique, l'optimisation est le processus qui consiste à réduire le temps d'exécution d'une fonction, l'espace occupé par les données et le programme, ou la consommation d'énergie.

En règle générale, l'optimisation doit s'effectuer une fois que le programme est fonctionnel et qu'il réponde aux spécificités attendues.

Avant de commencer l'optimisation, pour cela, il existe plusieurs approches d'optimisation l'une plus complexe que l'autre, on peut citer quelques unes :

- au niveau algorithmique, en choisissant un algorithme de complexité inférieure (au sens mathématique) et des structures de données adaptées,
- au niveau du langage de développement, en ordonnant au mieux les instructions et en utilisant les bibliothèques disponibles,
- en utilisant localement un langage de bas niveau, qui peut être le langage C ou, pour les besoins les plus critiques, le langage assembleur.

Comme on peut le voir tout cela demande énormément de temps et de travail. C'est là qu'intervient maqao (Modular Assembly Quality Analyzer and Optimizer) qui est de façon générale un outil d'analyse et d'optimisation des performances.

Dans le premier chapitre ci-dessous, nous allons définir plus en détails ce qu'est MAQAO.

2 Outil d'Analyse MAQAO

2.1 Definition

MAQAO (Modular Assembly Quality Analyzer and Optimizer) est un framework qui permet d'analyser et d'optimiser les performances d'un programme grâce à un ensemble de modules (CQA, LPROF, ONEVIEW). MAQAO effectue une analyse dynamique et statique du code et ainsi détermine les éléments limitant la performance d'une zone de l'application, il travaille au niveau binaire il n'a pas de restriction par rapport au langage utilisé dans le code source. l'objectif principal de MAQAO est de guider les développeurs d'applications tout au long du processus d'optimisation grâce à des rapports synthétiques et des astuces.[w1]

2.2 Les modules de base

Les principaux modules de base qu'utilise MAQAO pour la génération de rapports d'analyse de performance sont :

2.2.1 LProf

LPROF (Lightweight Profiler) est un module de MAQAO, c'est un profiler qui vérifie l'état d'un programme en effectuant un échantillonnage (200/s par défaut). Ainsi il permet de détecter le point chaud du programme. Son utilisation est relativement simple elle consiste en une ligne de commande composée de différents champs[w1] :

- Nom de l'application (ou chemin sinon situé dans le répertoire courant).
- Les arguments de l'application, le cas échéant.

Il existe plusieurs autres options pour LPROF :

```
$ maqao lprof - help
$ maqao lprof - <application> [arg1 arg2 ...]
```

2.2.2 CQA

CQA (Code Quality Analyzer) est un module de MAQAO, Le module CQA analyse le code généré statiquement et évalue son efficacité sur la base de l'architecture de la machine cible et des registres et instructions utilisées par le programme [w1]. Son utilisation est relativement simple elle consiste en une ligne de commande composée de différents champs :
\$ maqao cqa <BINARY_FILE> fct-loops=<FUNCTION> uarch=<MICRO-ARCHITECTURE>

- Nom du fichier binaire à analyser (ou chemin si non présent dans le répertoire courant).
- Nom des fonctions à analyser. Vous pouvez donner une liste d'expressions rationnelles : foo,bar\$ correspondra à foo29, my_foo et bar mais pas my_bar.

- Micro-architectures prises en charge (CORE265 and CORE2_45 (Core 2 65/45 nm), NEHALEM (Nehalem and Westmere), SANDY_BRIDGE IVY_BRIDGE, HASWELL BROADWELL, SKYLAKE (incl. Skylake SP), KABY/COMET_LAKE, ICELAKE & ICELAKE.SP, KNL or KNIGHTS_LANDING, AMD Zen/Zen+/Zen 2/Zen 3).

2.2.3 ONE View

Generation de rapport avec Oneview :

Oneview est un module délivré par MAQAO, il permet de générer des rapport d'analyse de codes source compilés en se basant sur différent modules. LPROF un profiler basé sur l'échantillonnage, et CQA un analyseur statique qui évalue la qualité du code généré par le compilateur. Oneview propose deux format de rapport HTML(defaut), XSLX ou text[w1]. Son utilisation est relativement simple elle consiste en une ligne de commande composée de différents champs :

```
$ maqao oneview -create-report=<report> -c=<config> [-xp=<dir>] [-of=<format>]
[-with-scalability]
```

- -create-report=*jreport*, permet de réaliser un rapport et exécutez toutes les étapes nécessaires les valeurs disponibles sont : one.
- -c=<config> spécifie le chemin d'un fichier de configuration.
- -xp=<dir> spécifie le chemin d'un répertoire d'expérimentation.
- -of=<format> spécifie le format du rapport généré (HTML pas défaut).
- -with-scalability active l'analyse d'évolutivité.

*Annexe