

Study and Optimization of a Finite Volume's Method Application

José Alves, Rui Brito

Universidade do Minho

Braga, November 2012

Index

- 1 Case Study
- 2 Test Methodology
- 3 Sequential version
- 4 Roofline
- 5 PAPI Analysis
- 6 Shared-Memory Parallel Optimization
- 7 Conclusion
- 8 Questions

- What?** Simulates the way heat is transfered in a fluid;
- How?** Uses finite-volumes method;
- Why?** Represents surface as a mesh, making each cell only dependent of its neighbours;

makeFlux Calculates the contribution from each edge <- VER ISTO;

makeResidual Calculates the ϕ vector, adding the flux for each cell, from each contribution <- VER ISTO;

LUFactorize Calculate a Gauss elimination <- VER ISTO;

Test Machines

	compute-511-2@search AMD Opt 6174	compute-611-1 Xeon X5650	MacBookPro Intel Ivy-Bridge i7
# processors	2	2	1
# cores per processor	12	6	4
hyperthreading	-	yes	yes
clock frequency(GHz)	2.2	2.66	2.3
L1 capacity	128KB	128KB	64KB
L2 capacity	512KB	256KB	256KB
L3 capacity	12MB	12MB	6MB
RAM capacity	64GB	48GB	16GB

Table: Test cases

Test Parameters

- 1 Best of 3 executions;
- 2 Test for different number of threads;
- 3 CENAS;

Original version

For each edge:

- 1 Calculate edge velocity;
- 2 Calculate flux;

For each cell:

- 1 Compute all contributions;

Compute vector ϕ ; Compute a Gauss elimination; Compute the error;

Optimized version

- 1 Reduce number of loads;
- 2 Change some variable definitions to *const*;
- 3 Usage of a recent compiler;

Counters Used

Used counters gathered by PAPI:

PAPI_TOT_CYC Total cycles;

PAPI_TOT_INS Total instructions

PAPI_LD_INS Load Instructions

PAPI_SR_INS Store Instructions

PAPI_FML_INS Multiply instructions

PAPI_FDV_INS Division instructions

PAPI_VEC_INS Vector Instructions

PAPI_FP_OPS Floating point operations

PAPI_L1_DCA L1 data cache accesses

PAPI_L1_DCM L1 data cache misses

PAPI_L2_DCA L2 data cache accesses

PAPI_L2_DCM L2 data cache misses

PAPI comparison

OpenMP Objectives

- 1 Parallelize application;
- 2 Decrease runtime;

Amdahl's Law

$$S_N = \frac{1}{(1-P) + P/N}$$

Parallel Portion	# Cores	Expected Speedup
CENAS	1	CENAS
CENAS	2	CENAS
CENAS	4	CENAS
CENAS	8	CENAS
CENAS	12	CENAS
CENAS	16	CENAS
CENAS	24	CENAS

Table: Test cases

Achieved Results

Conclusion

- Some difficulties measuring memory ceilings;
- Lack of analysis of output from PAPI;

Study and Optimization of a Finite Volume's Method Application

José Alves, Rui Brito

Universidade do Minho

Braga, November 2012