



OpenMP PROFILING WITH ompP

ompP is a performance profiler for OpenMP applications. It supports the developers of OpenMP applications in quickly identifying the most time consuming parts of their code. In addition, ompP offers a number of enhanced productivity features, such as overhead analysis, scalability analysis, hardware counter evaluators, and performance properties. ompP supports OpenMP applications written in C/C++ and FORTRAN.

SUPPORTED PLATFORMS

COMPILERS

Pathscale
PGI
SUN Studio
Intel icc & ifort
IBM xlc & xlf
gcc 4.2.0 and up

OPERATING SYSTEMS

Linux/x86
x86_64
ia64Solaris/SPARC
x86
x86_64
AIX/Power

E-MAIL omp@cs.utk.edu

ompP IS A COLLABORATIVE RESEARCH PROJECT



INNOVATIVE COMPUTING LABORATORY



Technische Universität München

SPONSORED BY



EXAMPLE PROFILING DATA

(Execution with four threads):

	R00002	main.c (34-37)	(default)	CRITICAL		
	TID	execT	execC	bodyT	enterT	exitT
#pragma omp parallel	0	3.00	1	1.00	2.00	0.00
{						
#pragma omp critical	1	1.00	1	1.00	0.00	0.00
{						
sleep(1)	2	2.00	1	1.00	1.00	0.00
}	3	4.00	1	1.00	3.00	0.00
}						
	SUM	10.01	4	4.00	6.00	0.00

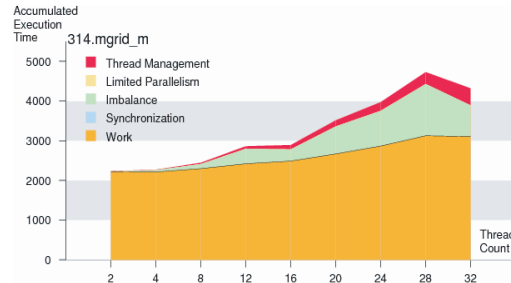
OVERHEAD CLASSIFICATION:

Quantify overheads of parallel execution into four classes

- Synchronization
- Imbalance
- Limited Parallelism
- Thread Management

SCALABILITY ANALYSIS:

See how overheads change with increasing number of threads

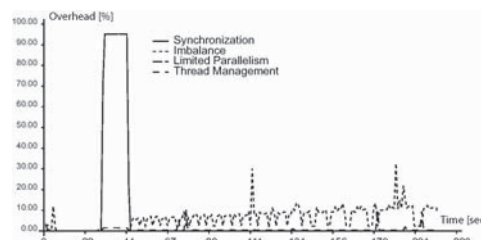


INCREMENTAL PROFILING:

- Capture profiles continuously (e.g., each second) while the program executes
- Analyze and display profiling data with temporal component

OVERHEADS OVER TIME

Application: 328.fma3d_m



PERFORMANCE COUNTER HEATMAPS

