

Contents

1	Introduction	1
1.1	Basic goals of the book	1
1.2	What do I get for one Watt today ?	1
1.3	Main memory bottleneck	3
1.4	Optimize resource usage	3
1.5	Application design	4
1.6	Organization of the book	4
1.6.1	Historical aspects	4
1.6.2	Parameterization	5
1.6.3	Models	5
1.6.4	Core optimization	6
1.6.5	Node optimization	6
1.6.6	Cluster optimization	6
1.6.7	Grid-brokering to save energy	7
2	Historical highlights	9
2.1	Evolution of computing	9
2.2	The first computer companies	14
2.2.1	ERA, EMCC and Univac	14
2.2.2	Control Data Corporation, CDC	14
2.2.3	Cray Research	15
2.2.4	Thinking Machines Corporation	16
2.2.5	International Business Machines (IBM)	17
2.2.6	The ASCI effort	18
2.2.7	The Japanese efforts	19
2.3	The computer generations	20
2.4	The evolution in computing performance	20
2.5	Performance/price evolution	22
2.6	Evolution of basic software	22
2.7	Evolution of algorithmic complexity	23
2.8	The TOP500 list	25
2.8.1	Outlook with the TOP500 curves	27

2.8.2	The GREEN500 List	28
2.8.3	Proposal for a REAL500 list	30
3	Parameterization	31
3.1	Definitions	31
3.2	Parameterization of applications	35
3.2.1	Application parameter set	35
3.2.2	Parameterization of BLAS library routines	36
3.2.3	SMXV: Parameterization of sparse matrix*vector operation ..	38
3.3	Parameterization of a computational nodes $P_i \in r_i$	39
3.4	Parameterization of the interconnection networks	41
3.4.1	Types of networks	41
3.4.2	Parameterization of clusters and networks	42
3.5	Parameters related to running applications	44
3.6	Conclusion	47
4	Models	49
4.1	The performance prediction model	49
4.2	The execution time evaluation model (ETEM)	53
4.3	A network performance model	53
4.4	The extended $\Gamma - \kappa$ model	55
4.5	Validation of the models	56
4.5.1	Methodology	56
4.5.2	Example: The full matrix*matrix multiplication DGEMM ...	57
4.5.3	Example: Sparse matrix*vector multiplication SMXV	59
5	Core optimization	63
5.1	Some useful notions	63
5.1.1	Data hierarchy	63
5.1.2	Data representation	64
5.1.3	Floating point operations	67
5.1.4	Pipelining	68
5.2	Single core optimization	70
5.2.1	Single core architectures	70
5.2.2	Memory conflicts	70
5.2.3	Indirect addressing	74
5.2.4	Unrolling	75
5.2.5	Dependency	76
5.2.6	Inlining	78
5.2.7	If statement in a loop	78
5.2.8	Code porting aspects	79
5.2.9	How to develop application software	83

5.3	Application to plasma physics codes	84
5.3.1	Tokamaks and Stellarators	84
5.3.2	Optimization of VMEC	88
5.3.3	Optimization of TERPSICHORE	91
5.3.4	Conclusions for single core optimization	94
6	Node optimization	95
6.1	Shared memory computer architectures	95
6.1.1	SMP/NUMA architectures	95
6.1.2	The Cell	99
6.1.3	GPGPU for HPC	100
6.2	Node comparison and OpenMP	105
6.2.1	Race condition with OpenMP	109
6.3	Application optimization with OpenMP: the 3D Helmholtz solver	110
6.3.1	Fast Helmholtz solver for parallelepipedic geometries	111
6.3.2	NEC SX-5 reference benchmark	113
6.3.3	Single processor benchmarks	114
6.3.4	Parallelization with OpenMP	115
6.3.5	Parallelization with MPI	115
6.3.6	Conclusion	119
6.4	Application optimization with OpenMP: TERPSICHORE	119
7	Cluster optimization	121
7.1	Introduction on parallelization	121
7.2	Internode communication networks	121
7.2.1	Network architectures	121
7.2.2	Comparison between network architectures	129
7.3	Distributed memory parallel computer architectures	131
7.3.1	Integrated parallel computer architectures	131
7.3.2	Commodity cluster architectures	134
7.3.3	Energy consumption issues	136
7.3.4	The issue of resilience	137
7.4	Type of parallel applications	138
7.4.1	Embarrassingly parallel applications	138
7.4.2	Applications with point-to-point communications	138
7.4.3	Applications with multicast communication needs	139
7.4.4	Shared memory applications (OpenMP)	139
7.4.5	Components based applications	139
7.5	Domain decomposition techniques	139
7.5.1	Test example: The Gyrotron	140
7.5.2	The geometry and the mesh	142
7.5.3	Connectivity conditions	142

7.5.4	Parallel matrix solver	143
7.5.5	The electrostatic precipitator	145
7.6	Scheduling of parallel applications	146
7.6.1	Static scheduling	146
7.6.2	Dynamic scheduling	146
7.7	SpecuLOOS	147
7.7.1	Introduction	147
7.7.2	Test case description	147
7.7.3	Complexity on one node	149
7.7.4	Wrong complexity on the Blue Gene/L	150
7.7.5	Fine results on the Blue Gene/L	151
7.7.6	Conclusions	151
7.8	TERPSICHORE	153
7.9	Parallelization of the LEMan code with MPI and OpenMP	154
7.9.1	Introduction	154
7.9.2	Parallelization	154
7.9.3	CPU time results	156
7.10	Conclusions	159
8	Grid-level Brokering to save energy	161
8.1	About Grid resource brokering	161
8.2	An Introduction to ianos	162
8.2.1	Job Submission Scenario	164
8.3	The cost model	165
8.3.1	Mathematical formulation	165
8.3.2	CPU costs K_e	167
8.3.3	License fees K_ℓ	169
8.3.4	Costs due to waiting time K_w	169
8.3.5	Energy costs K_{eco}	169
8.3.6	Data transfer costs K_d	171
8.3.7	Example: The Pleiades clusters CPU cost per hour	171
8.3.8	Different currencies in a Grid environment	173
8.4	The implementation	173
8.4.1	Architecture & Design	174
8.4.2	The Grid Adapter	174
8.4.3	The Meta Scheduling Service (MSS)	175
8.4.4	The Resource Broker	176
8.4.5	The System Information	177
8.4.6	The Data Warehouse	177
8.4.7	The Monitoring Service	177
8.4.8	The Monitoring Module VAMOS	178
8.4.9	Integration with UNICORE Grid System	179
8.4.10	Scheduling algorithm	179
8.4.11	User Interfaces to the ianos framework	181

- 8.5 DVS-able processors 182
 - 8.5.1 Power consumption of a CPU 183
 - 8.5.2 An algorithm to save energy 184
 - 8.5.3 First results with SMXV 185
 - 8.5.4 A first implementation 186
- 8.6 Conclusions 188
- 9 Recommendations 189**
 - 9.1 Application oriented recommendations 189
 - 9.1.1 Code development 189
 - 9.1.2 Code validation 189
 - 9.1.3 Porting codes 190
 - 9.1.4 Optimizing parallelized applications 190
 - 9.1.5 Race condition 190
 - 9.2 Hardware and basic software aspects 191
 - 9.2.1 Basic software 191
 - 9.2.2 Choice of system software 192
 - 9.3 Energy reduction 192
 - 9.3.1 Processor frequency adaptation 192
 - 9.3.2 Improved cooling 193
 - 9.3.3 Choice of optimal resources 193
 - 9.3.4 Best choice of new computer 193
 - 9.3.5 Last but not least 194
 - 9.4 Miscellaneous 194
 - 9.4.1 Course material 194
 - 9.4.2 A new REAL500 List 194
- Glossary 197**
- References 205**
- About the authors 213**
- Index 215**