Exploiting multi-core parallelism on optimal decision trees

AUTHOR
Ayush Patandin
(S.A.S.patandin@student.tudelft.nl)

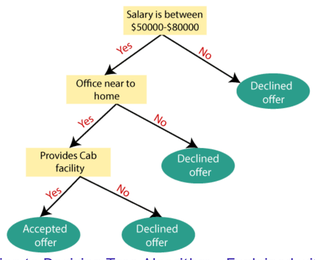
PROFESSOR/SUPERVISOR
Dr. Emir Demirović
(e.demirovic@tudelft.nl)

FACULTY
EEMCS TU Delft

COURSE
CSE3000 Research Project

1 INTRODUCTION

- Decision trees: popular for solving classification or regression problems
- Heuristic methods vs. optimal decision tree algorithms
- Multi-core parallelism: able to quickly produce optimal decision trees



Introduction to Decision Tree Algorithm - Explained with Examples (mygreatlearning.com)

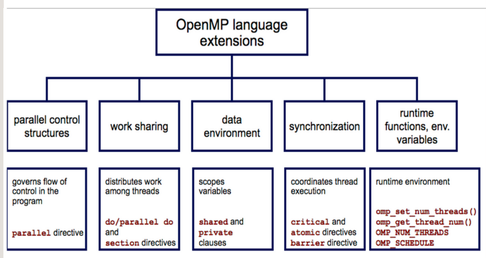
2 OBJECTIVE

Is it possible to exploit multi-core parallelism to produce optimal decision trees faster?

1. What are the possible trade-offs that need to be investigated when parallelizing parts of an optimal decision tree algorithm?
2. Which approach can be used to split the data and the tasks in parallel among the different workers in the decision tree?
3. How to avoid correctness and performance issues when integrating parallelism into a decision tree algorithm?

3 METHODOLOGY

- OpenMP: multi-threaded API
- Shared memory space
- Data race prevention
- Load Balancing



https://upload.wikimedia.org/wikipedia/commons/thumb/9/9b/OpenMP_language_extensions.svg/1024px-OpenMP_language_extensions.svg.png?1621512030122

4 PRELIMINARY WORK

- MurTree
- Specialized depth 2 algorithm
- General Depth algorithm
- Caching of optimal subtrees
- Performance metrics
  1. Parallel execution time
  2. Parallel speedup

$$S_p = \frac{T_{p=1}}{T_p}$$

  3. Program efficiency

$$E_p = \frac{S_p}{p} = \frac{T_{p=1}}{p \cdot T_p}$$

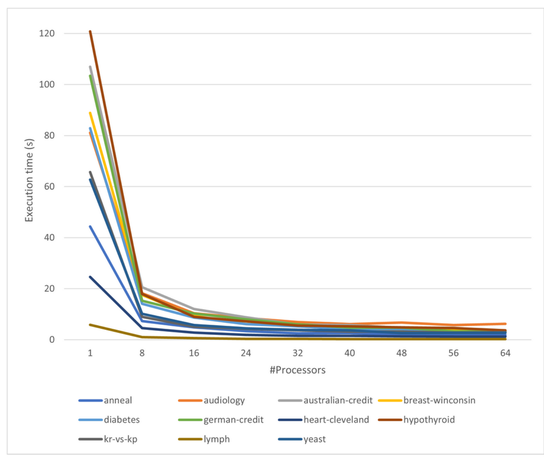
5 RESULTS

(1) Trained decision tree models with dataset properties and their misclassification score for different tree depths.
(2) Parallel execution time w.r.t. #processors for different benchmarks
(3) Parallel speedup w.r.t. #processors for different benchmarks
(4) Efficiency w.r.t. #processors for different benchmarks

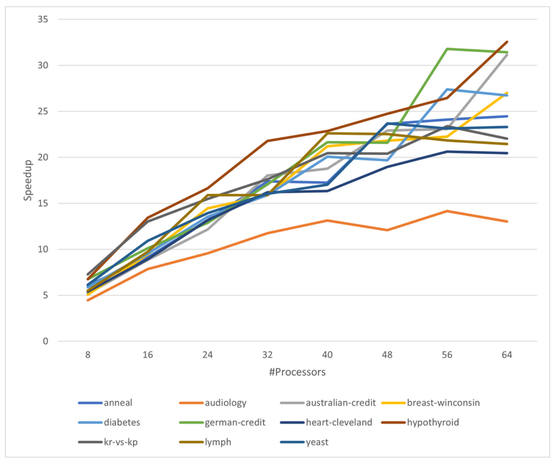
(1)

Name	|D|	|F|	$MS_{depth=1}$	$MS_{depth=2}$	$MS_{depth=3}$	$MS_{depth=4}$
anneal	812	93	151	137	112	91
audiology	216	148	29	10	5	1
australian-credit	653	125	89	87	73	56
breast-wisconsin	683	120	48	22	15	7
compas-binary	6907	12	2494	2333	2272	2250
diabetes	768	112	196	177	162	137
fico-binary	10459	17	3180	3019	2959	2894
german-credit	1000	112	290	267	236	204
heart-cleveland	296	95	69	60	41	25
hepatitis	137	68	19	16	10	3
hypothyroid	3247	88	118	70	61	53
ionosphere	351	445	59	32	22	7
kr-vs-kp	3196	73	1012	418	198	144
letter	20000	224	813	599	369	261
lymph	148	68	30	22	12	3
mushroom	8124	119	920	252	8	0
pendigits	7494	216	505	153	47	13
primary-tumor	336	31	70	58	46	34
segment	2310	235	41	9	0	0
soybean	630	50	92	55	29	14
splice-1	3190	287	575	508	224	141
tic-tac-toe	958	27	288	282	216	137
vehicle	846	252	189	75	26	12
vote	435	48	19	17	12	5
yeast	1484	89	442	437	403	366

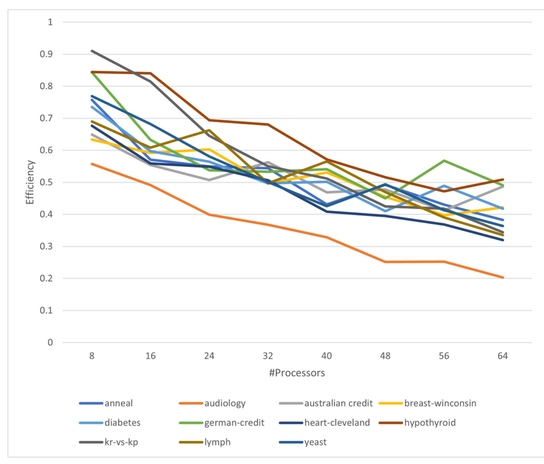
(2)



(3)



(4)



6 CONCLUSION

- Parallel MurTree algorithm: constructs accurate tree models on binarised datasets with incredible runtime improvements
- More CPUs -> better speedup and worse efficiency
- OpenMP constructs: partition and schedule chunks of computations to threads. Also prevent possible setbacks (data race, idle threads)
- Explore the work with Open MPI

RELATED LITERATURE

- E. Demirović, A. Lukina, E. Hébrard, J. Chan, J. Bailey, C. Leckie, K. Ramamohanarao, and P. J. Stuckey, Murtree: Optimal classification trees via dynamic programming and search,"ArXiv, vol. abs/2007.12652, 2020.
- T. Mattson, "An introduction to openmp," Feb. 2001, pp. 3–3,isbn: 0-7695-1010-8.doi:10.1109/CCGRID.2001.923161.
- T. Rauber and R. Gudula,Parallel Programming: for Multicore and Cluster Systems. Springer,2013.

TU Delft Delft University of Technology