Performance optimizations for the atmospheric tomography problem of extremely large telescopes on real-time hardware

Bernadett Stadler

joint work with Ronny Ramlau (JKU Linz) & Roberto Biasi (Microgate)

Reduced Order Modelling, Simulation and Optimization of Coupled Systems (ROMSOC)

Klosterneuburg, February 19, 2020

Funded by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Grant Agreement No. 765374.

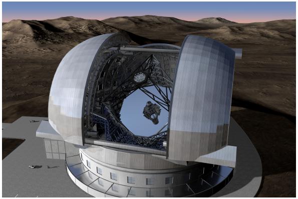
Outline

Extremely Large Telescope (ELT)

- planned to be the world's largest optical/near-infrared telescope
- built by the European Southern Observatory (ESO)
- in the Atacama desert, Chile
- with a primary mirror of 39 m

⇒ highly efficient algorithms to achieve an excellent image quality

- **adaptive optics:** compensate optical distortions in the atmosphere
- **atmospheric tomography:** reconstruction of the turbulent layers from sensor measurements



Credit: ESO

Atmospheric tomography

- **goal:** reconstruct turbulent layers ϕ from measurements s
- mathematical problem formulation:

$$s = A\phi$$

  inverse problem $\Rightarrow$ requires regularization
- **approach:** Bayesian framework and maximum a-posteriori estimate

$$(A^T C_\eta^{-1} A + C_\phi^{-1})\phi = A^T C_\eta^{-1} s$$

  where C_ϕ and C_η are the covariance matrices of layers ϕ and noise η

Challenges:

- computationally very expensive operations
- to be solved in real-time (500 Hz)

$\Rightarrow$ **need an efficient solver**

Standard approach: Matrix Vector Multiplication (MVM)

We want to solve:

$$(A^T C_\eta^{-1} A + C_\phi^{-1})\phi = A^T C_\eta^{-1} s$$

Soft Real Time (SRT):

Compute the reconstruction matrix by inverting the left-hand-side operator:

$$R := (A^T C_\eta^{-1} A + C_\phi^{-1})^{-1} A^T C_\eta^{-1}$$

Hard Real Time (HRT):

Apply matrix vector multiplication to get mirror commands:

$$\phi = R \cdot s$$

Finite Element Wavelet Hybrid Algorithm (FEWHA)

We want to solve:

$$(A^T C_\eta^{-1} A + C_\phi^{-1})\phi = A^T C_\eta^{-1} s$$

Use clever transformations - dual domain discretization:

$$(\boldsymbol{W}^{-T} \hat{A}^T C_\eta^{-1} \hat{A} \boldsymbol{W}^{-1} + \alpha D)c = \boldsymbol{W}^{-T} \hat{A}^T C_\eta^{-1} s$$

- $\boldsymbol{W}$ is a mapping between finite element and wavelet domain
- $\hat{A}$ discretized in the bilinear domain
- D is a diagonal approximation of C_ϕ^{-1} in the wavelet domain

Hard Real Time (HRT): iterative approach to compute c
$\Rightarrow$ Preconditioned Conjugate Gradient Method

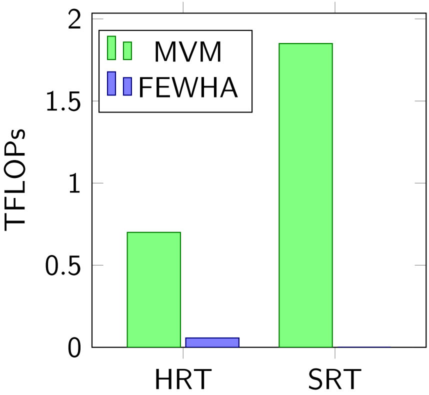
[2] *M. Yudytskiy, T. Helin, R. Ramlau, A finite element - wavelet hybrid algorithm for atmospheric tomography*

Test setting - MAORY

- first light instrument of the ELT
- collaboration with Microgate (responsible for RTC design)

number of layers:	6
number of wavefront sensors:	6
number of measurements per sensor:	9 600
number of modes per layer:	5 000
number of wavelets scales per layer:	128x128 or 64x64
number of PCG iterations for FEWHA:	4
hard real time:	2 ms
soft real time:	360 s
size of MVM matrix:	30 000 x 57 600
size of FEWHA matrix:	32 768 x 57 600

Theoretical performance analysis



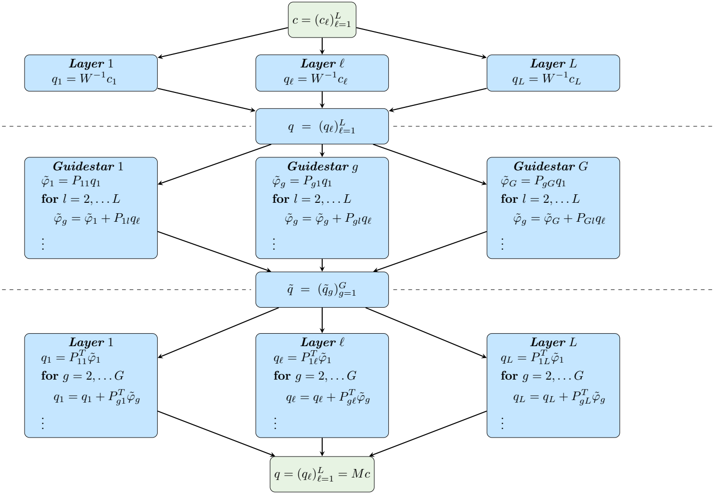
Memory Usage:

	MVM	**FEWHA**
HRT	3GB	15MB
SRT	50GB	-
	53GB	**15MB**

single precision floating point numbers

⇒ FEWHA **12 times faster** than the MVM for hard real time
⇒ FEWHA has **no additional soft real time** computational load
⇒ matrix free → saves a lot of memory and FLOPs

Global parallelization of operator $(W^{-T}\hat{A}^T C_\eta^{-1}\hat{A}W^{-1} + \alpha D)$



[3] *B.Stadler, R. Biasi, R. Ramlau, Feasibility of standard a novel solvers for atmospheric tomography*

Implementation on RTC hardware

One node of Radon1 HPC cluster of RICAM

- equipped with two 8-core Xeon E5-2630v3, 2.4Ghz
- **parallelization:**
  OpenMP, Intel TBB and SSE
- **optimization:**
  - number of cores/threads (6 / 12)
  - vectorization
  - memory management

NVIDIA Tesla V100 GPU

- 5120 CUDA Cores, 1.2 GHz
  max. blocks per SM: 32
  max. threads per block: 1024
- **parallelization:**
  CUDA 10.1
- **optimization:**
  - number of threads/blocks
  - memory management
  - concurrent kernel execution
  - avoid transfer GPU - CPU

$\Rightarrow$ **2.3ms for FEWHA** $\Rightarrow$ **10ms for FEWHA**

Interpretation

⇒ for FEWHA the CPU is a lot faster than the GPU
⇒ but for MVM the GPU is faster

Possible reasons:

- CPU has a faster clock (2.4 GHz vs. 1.2 GHz)
- memory bounded algorithm → faster memory access on CPU (Caches)
- dimension of the problem is too small
  - local parallelization of operators:
    matrix-free implementation → over max. 128x128
  - global parallelization:
    number of 6 layers or 6 wavefront sensors
- maybe other...

Thank you!

[1] *B. L. Ellerbroek and C. R. Vogel, Inverse problems in astronomical adaptive optics*.

[2] *M. Yudytskiy and T. Helin and R. Ramlau, Finite element-wavelet hybrid algorithm for atmospheric tomography*.

[3] *B.Stadler, R. Biasi, R. Ramlau, Feasibility of standard a novel solvers for atmospheric tomography*.

Funded by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie Grant Agreement No. 765374.

Performance of FEWHA Operators

	GPU	CPU
Discrete Wavelet Transform	0.15	0.08
Bilinear Interpolation	0.08	0.015
Shack-Hartmann Operator	0.02	0.004
Diagonal Matrix	0.08	0.02