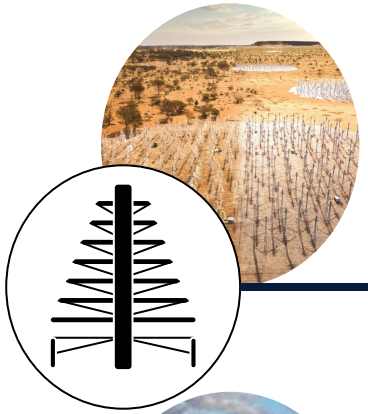
GT SKA

# SimSDP: Proof of Concept for Radio Astronomy Imaging on High-Performance Architectures

**O. Renaud***, M. Quinson, N. Gac
**ENS Rennes, IRISA, CNRS, SATIE Paris-Saclay, France*
*ophelie.renaud@ens-rennes.fr*

# Square Kilometre Array (SKA)



CSP

CSP

SDP

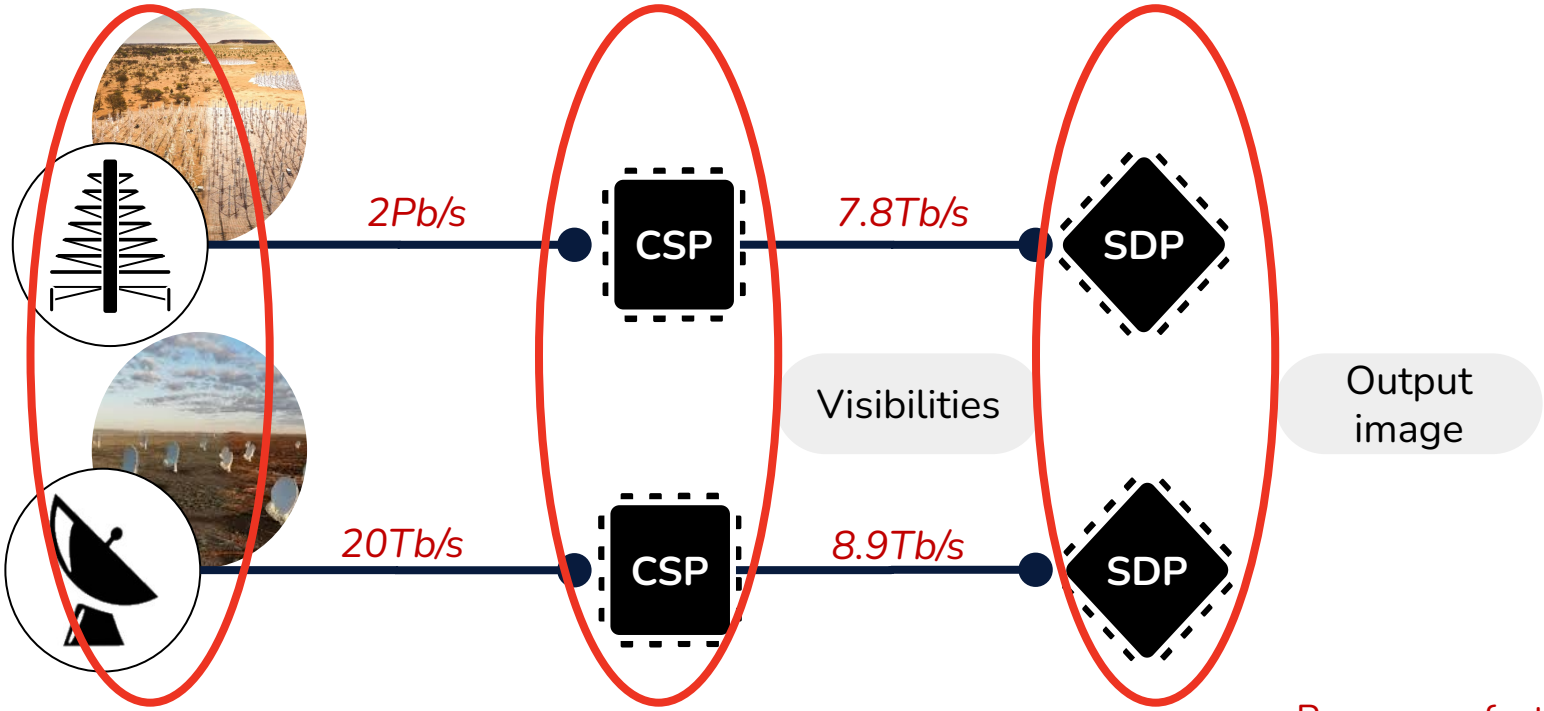
SDP

Visibilities

Output image

CSP: Central Signal Processor
SDP: Science Data Processor

# Square Kilometre Array (SKA)

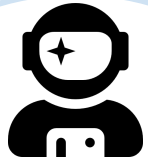



Huge collecting surface

Storage constraints

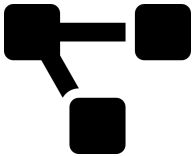
Process as fast as possible

# How to simulate SDP imaging pipelines on HPC systems?

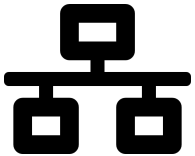


Not parallel programming expert

SKA objectives

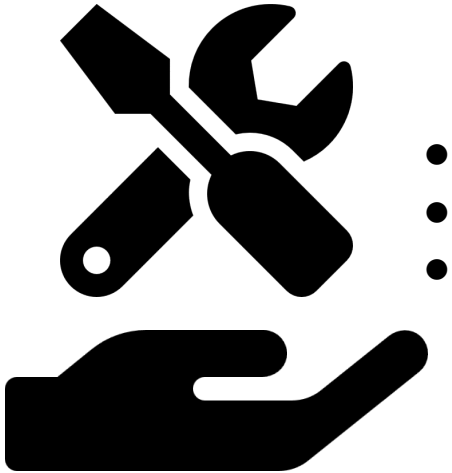


Algorithm in development

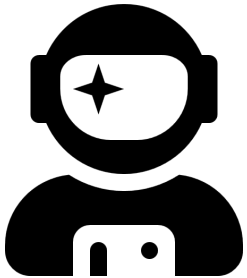


Not yet built HPC System

# Objectives



- Optimize
- Allocate
- Analyse

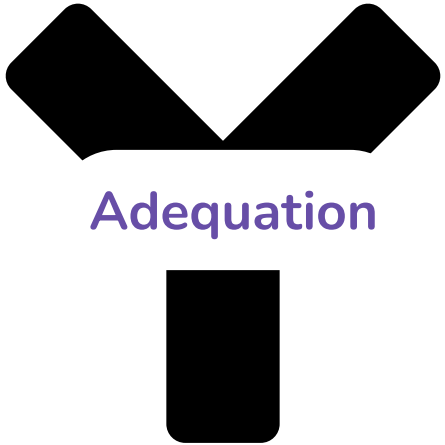



Resource
Allocation process
on HPC systems

# Graph based Algorithm-Architecture Adequation (AAA)

Model of Computation (MoC)

Model of Architecture (MoA)



Adequation

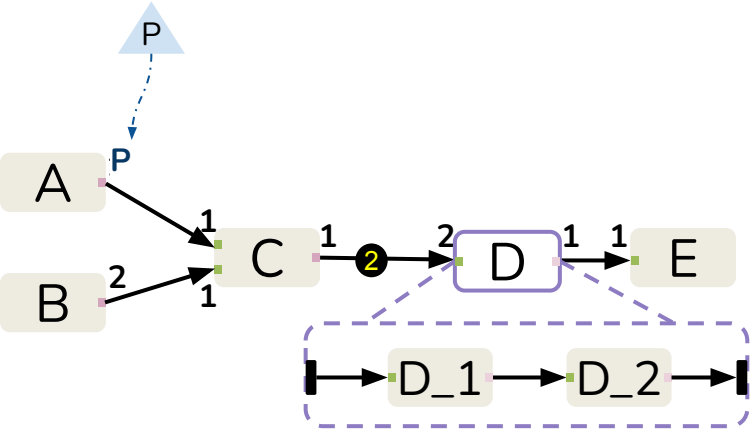
Code generation

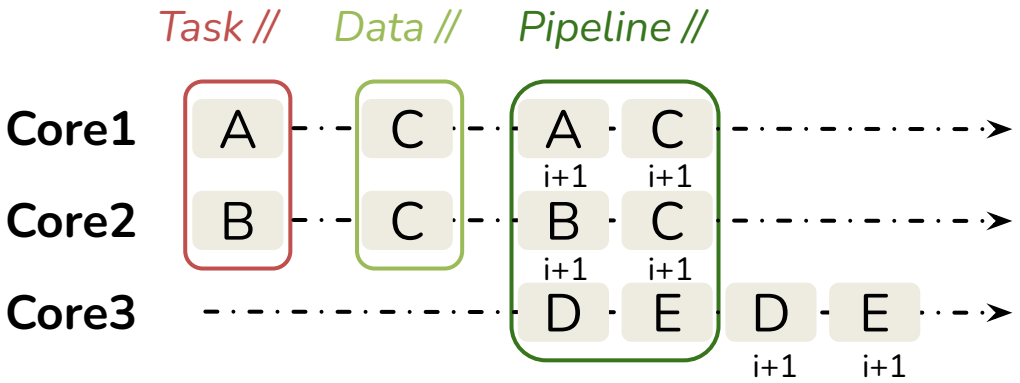


[r] C. Erbas, et. al, « Multi objective optimization and evolutionary algorithms for the application mapping problem in multiprocessor system-on-chip design »

# Dataflow MoC

✓ Expression of several types of parallelism



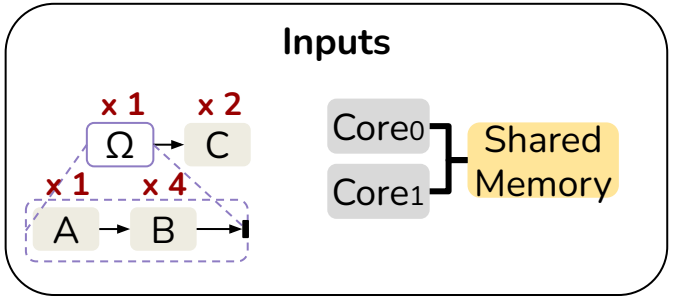

✓ Ensure **consistency,** prevents manual mistakes.

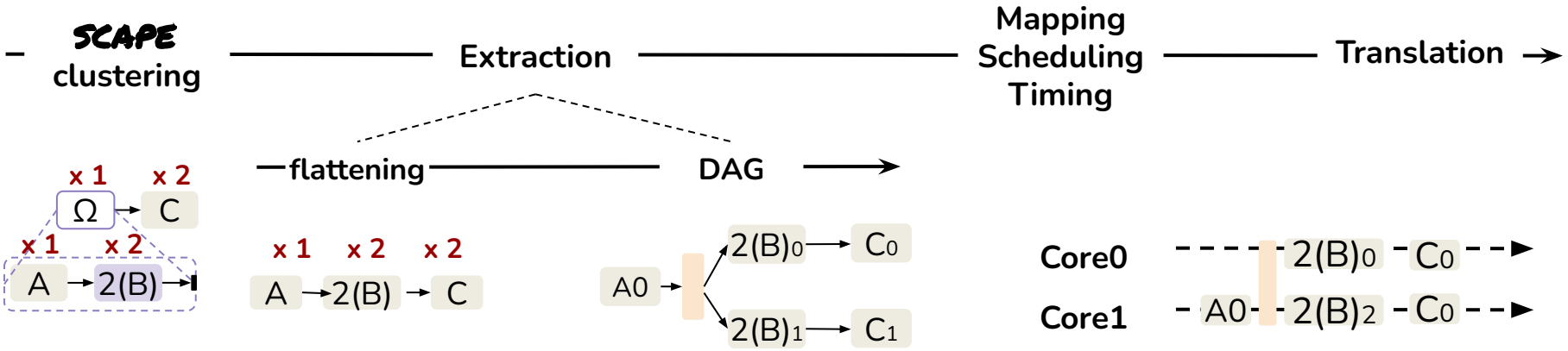
✓ High **predictability**.

✓ Allow automatic resource allocation.

*SDF: Synchronous Dataflow*
*PiSDF: Parameterized and Interfaced SDF*
*[r] E. Lee and D. Messerschmitt. "Static scheduling of synchronous data flow programs for digital signal processing".*
*[r] K. Desnos et. al. "Pimm : Parameterized and interfaced dataflow meta-model for mpsocs runtime reconfiguration"*

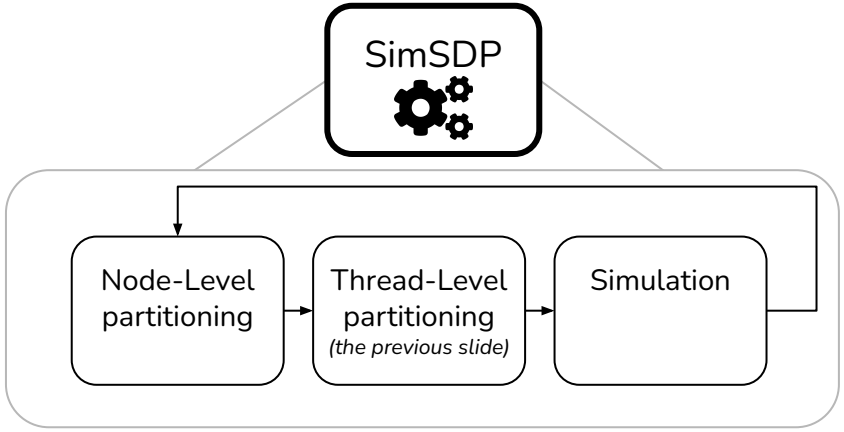
# Resource allocation on standard system




[1] **O. Renaud**, D. Gageot, K. Desnos, J.-F. Nezan. SCAPE: HW-Aware Clustering of Dataflow Actors for Tunable Scheduling Complexity, DASIP, 2023
[2] **O. Renaud**, N. Haggui, K. Desnos, J.-F. Nezan. Automated Clustering and Pipelining of Dataflow Actors for Controlled Scheduling Complexity, EUSIPCO, 2023
[3] **O. Renaud**, H. Miomandre, K. Desnos, J.-F. Nezan. Automated Level-Based Clustering of Dataflow Actors for Controlled Scheduling Complexity, JSA, 2024

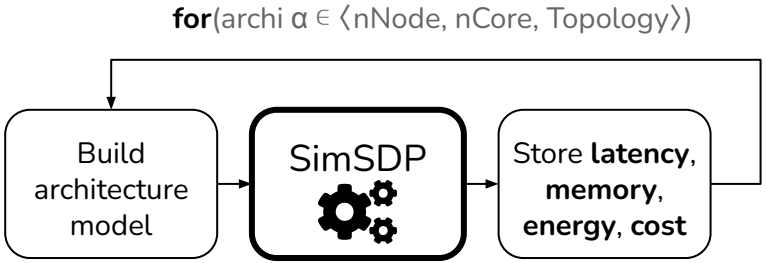
# SimSDP - Resource allocation on HPC system



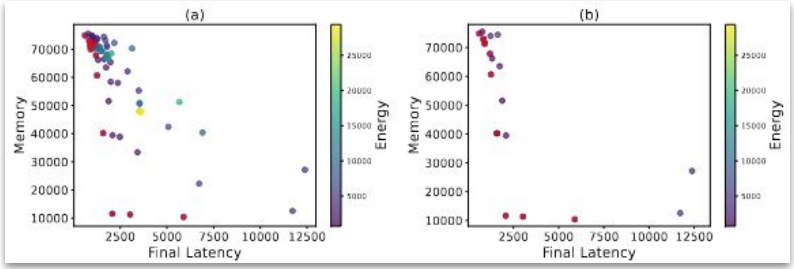



[4] **O. Renaud**, A. Gougeon, K. Desnos, C. Phillips, J. Tuthill, M. Quinson, J.-F. Nezan. SimSDP: Automatic Workload-Balancing on Multi-Node & Multi-Core HPC Architectures based on dataflow models, TPDS.

# Co-designing HPC HW/SW with SimSDP




Stop ⟶ ∃ i ≥ δα : Lfinal(α, i) ≤ Lfinal(Smax)




| | Sobel | RFI | Squeezenet |
|---|---|---|---|
| **Final latency** | 0,52 % | 1,06 % | 0,98 % |
| **Memory** | 3,15 % | 6,52 % | 0,02 % |

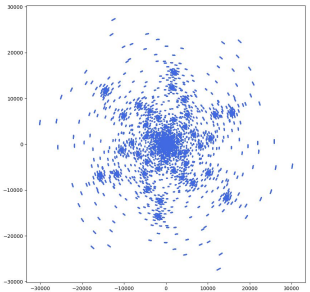
TABLE I
IN VIVO VS. IN SILICO ERROR ANALYSIS: DEPLOYING SOBEL, RFI FILTER, AND SQUEEZENET ON 5 ARCHITECTURES

This prove the reliability of the SimSDP and its exploitability in HPC DSE

[5] Ophélie Renaud, Karol Desnos, Erwan Raffin, Jean-François, "Multicore and Network Topology Codesign for Pareto-Optimal Multinode Architecture", EUSIPCO, 2024


Radio astronomy
imaging
algorithms

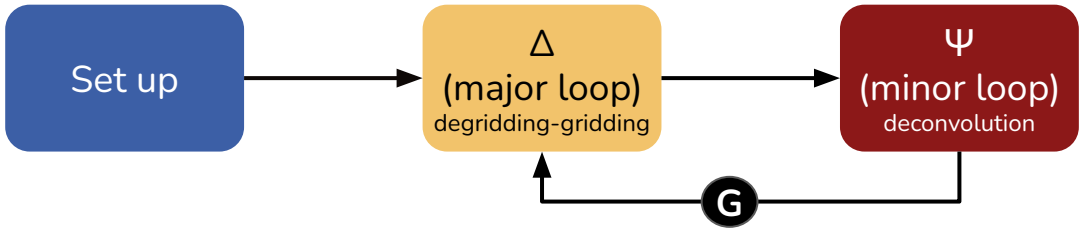
# Radio astronomy imaging principle



CSP visibilities

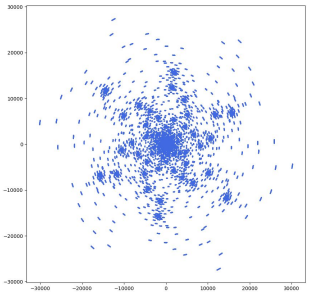
↑

correlation point of a pair of antenna

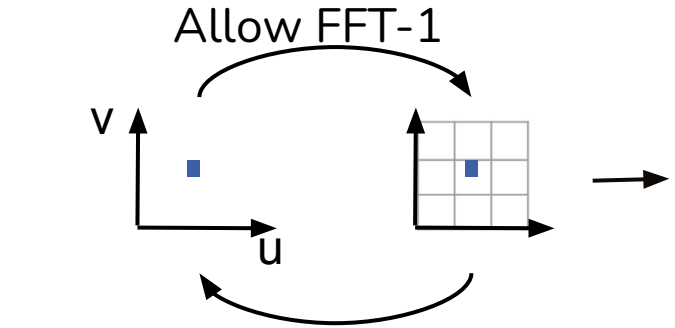


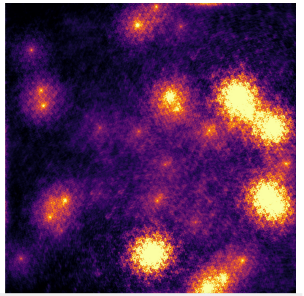

Output image

# Radio astronomy imaging principle

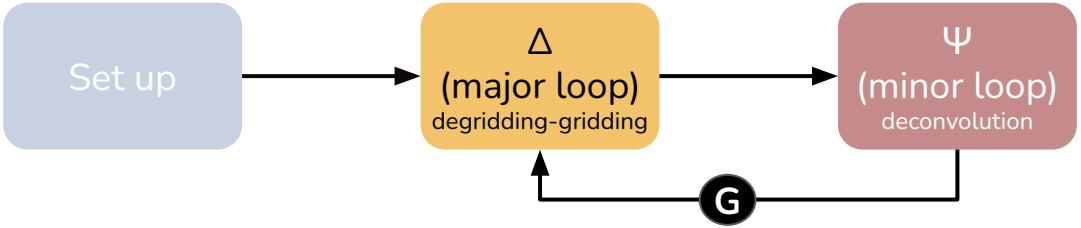


CSP visibilities

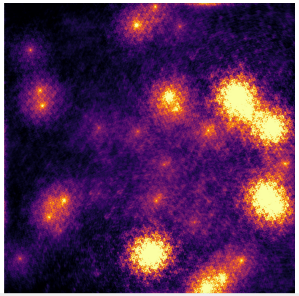


Dirty image

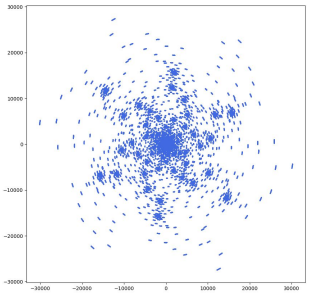


# Radio astronomy imaging principle

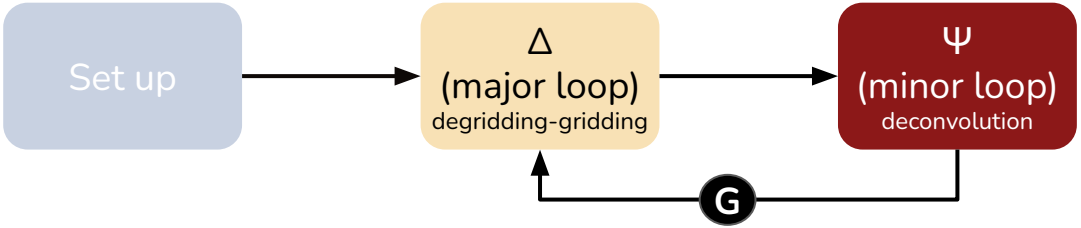


Dirty image

= Sky Image * PSF

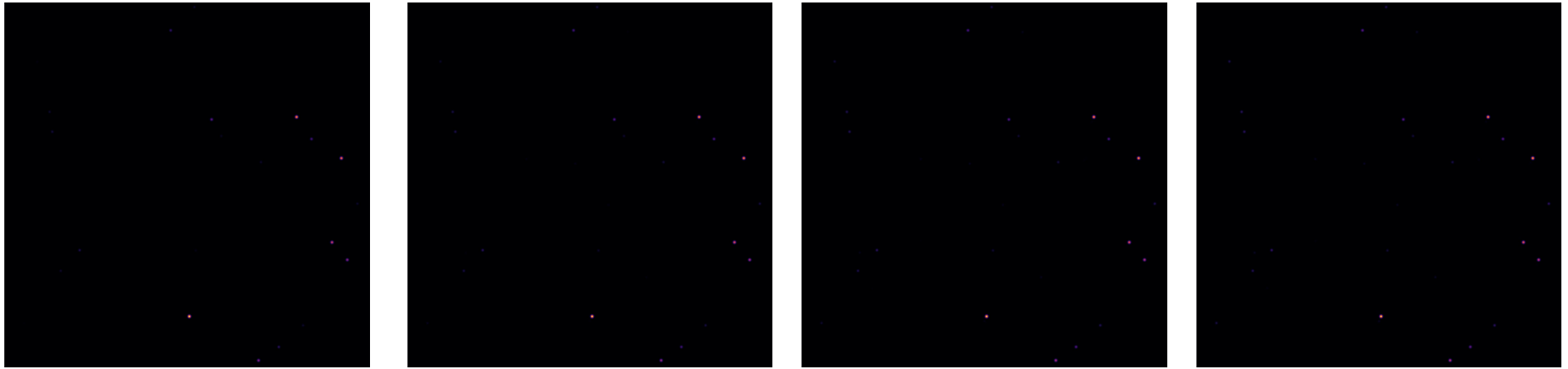


CSP visibilities

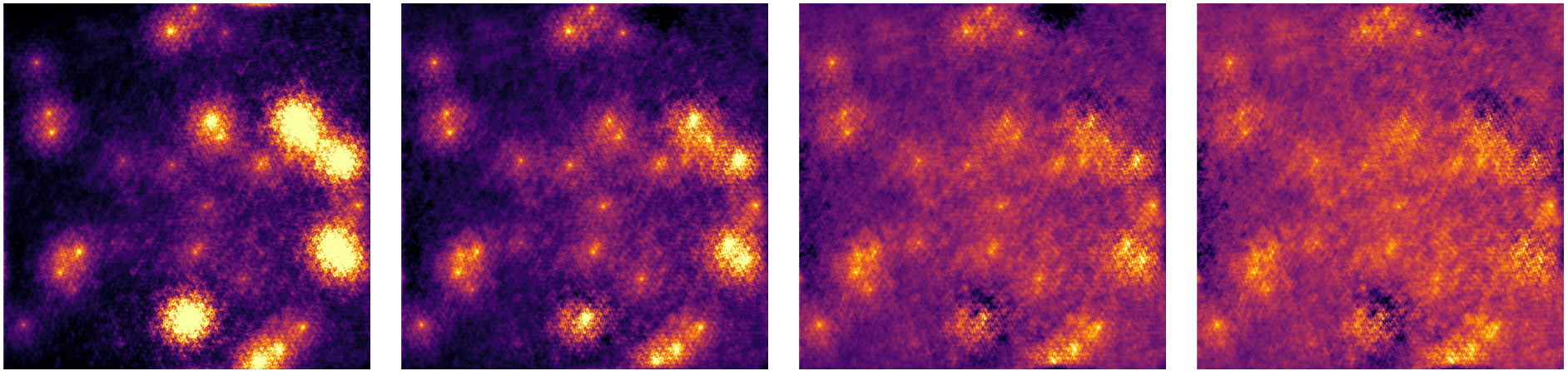


Deconvoluted image

Existing work

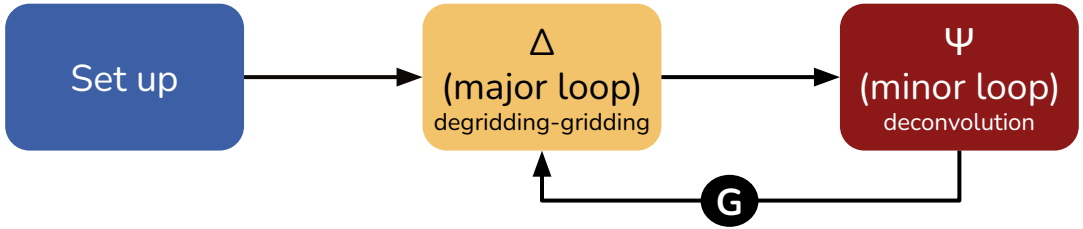


Dirty Image



Cycle

# Generic imaging pipeline




Selection of algorithms specifying major and minor loops

Direct Fourier Transform (DFT)
*(simple but long)*

Fast Fourier Transform (FFT)
*(faster on big grid)*

Grid to Grid (G2G)
*(faster, same O)*
N. Monnier

Högbom CLEAN
*(simple but long)*

This pipeline is a very good entry point for comparing the performance of algorithms on **1 CPU** architecture node



[r] S. Wang, N. Gac, H. Miomandre, J.-F. Nezan, K. Desnos, F. Orieux « An Initial Framework for Prototyping Radio-Interferometric Imaging Pipelines»

# Polynomial fit function for timing simulation

The goal is to facilitate pipeline comparison varying parameters

actor_timing(target, param1,param2) = polynomial(param1,param2), where param ∈ **NUM_VIS**, **GRID_SIZE**, **NUM_MINOR_CYCLE**

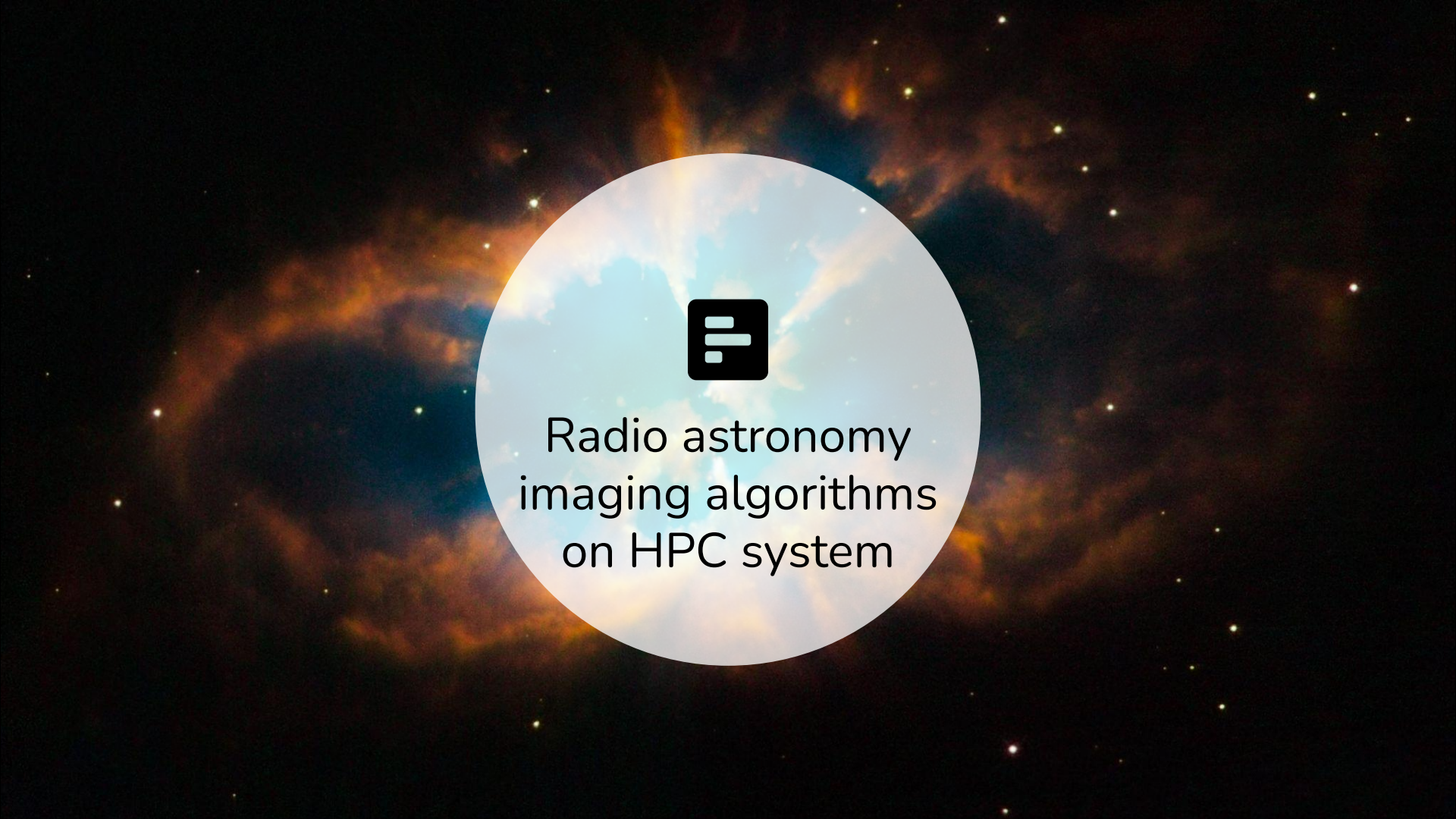
```
# Building the scripted benchmark
 FOR each param1 ∈ PARAM1 DO:
  FOR each param2 ∈ PARAM2 DO:
   EXECUTE Instrumented_code(param1,param2)
   SAVE {timing,config} → actor_timing.csv

# Calculating the polynomial and RMSE
FOR each actor_timing.csv ∈ CSV
 FOR each dof ∈ DOF
  COMPUTE polynomial(dof)
  COMPUTE RMSE(measure, polynomial)
```

This method is currently manual, so that limits it:

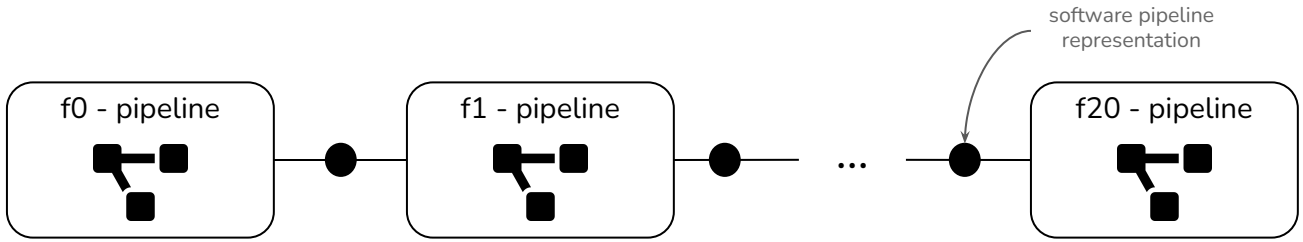
- in sampling (up to 8 samples per parameter)
- in the degree of polynomial evaluated (limited to 2)



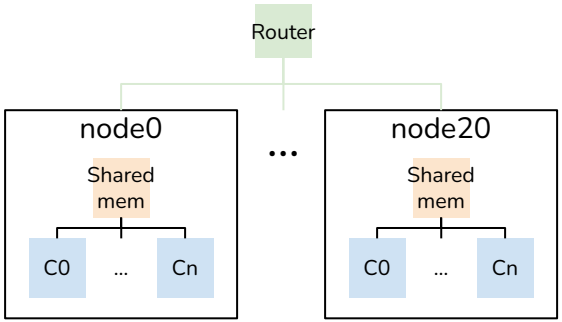
*[r] S. Wang, N. Gac, H. Miomandre, J.-F. Nezan, K. Desnos, F. Orieux « An Initial Framework for Prototyping Radio-Interferometric Imaging Pipelines»*


Radio astronomy
imaging algorithms
on HPC system

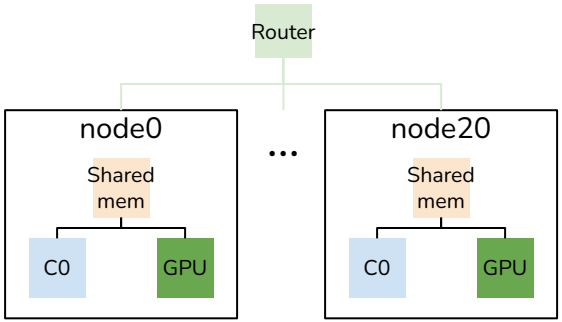
# Target architectures



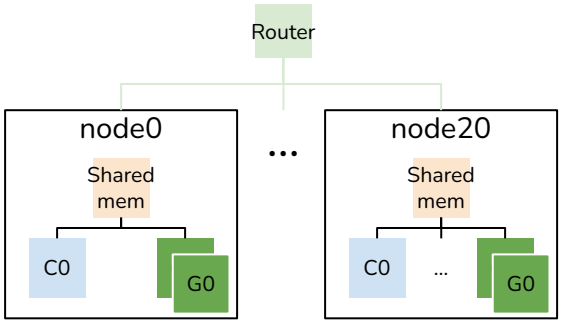

Multinode - multicore



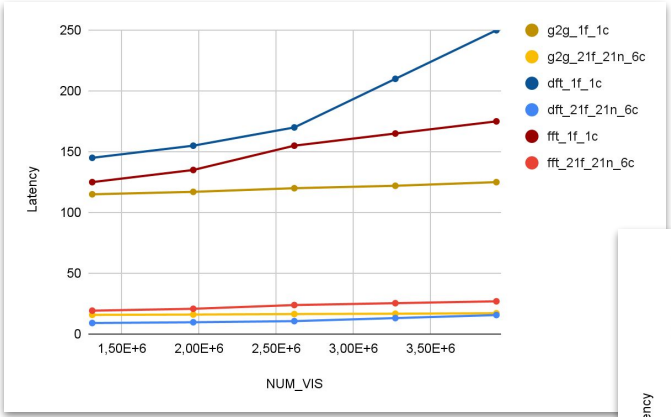

Multinode - monoGPU

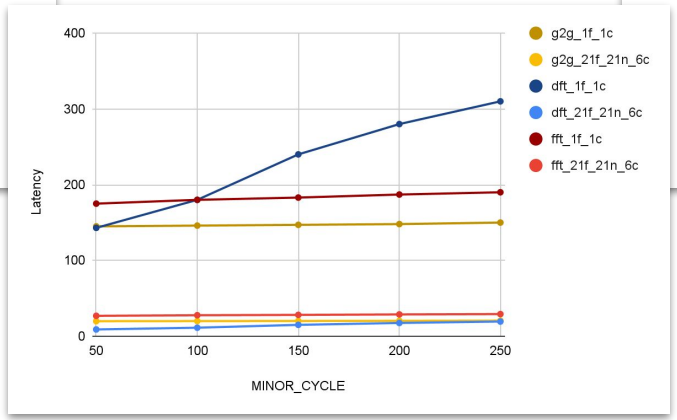


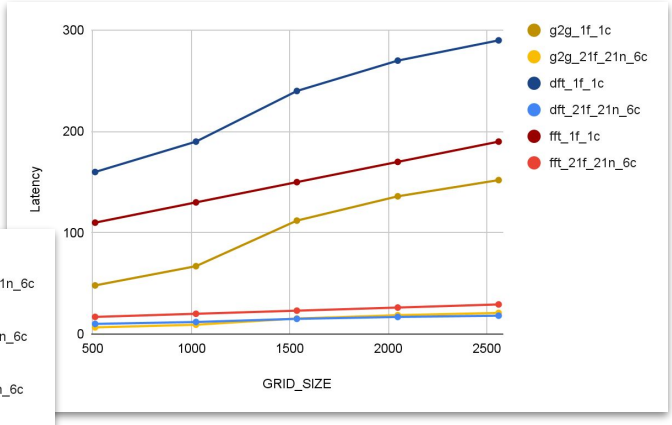

Multinode - multiGPU



# Simulating generic imaging pipelines - 21 freq - CPU frequency based node partitioning




This corresponds to the Sunrise benchmark [DASIP] applied to HPC (the comparison with the measurement is missing).

# Polynomial fit function for timing simulation

The goal is to facilitate pipeline comparison varying parameters

actor_timing(target, param1,param2) = polynomial(param1,param2), where param ∈ **NUM_VIS**, **GRID_SIZE**, **NUM_MINOR_CYCLE**

```
# Building the scripted benchmark
FOR each target ∈ [CPU ∨ GPU]  DO:
  FOR each param1 ∈ PARAM1 DO:
    FOR each param2 ∈ PARAM2 DO:
      EXECUTE Instrumented_code(param1,param2)
      SAVE {timing,config} → actor_timing.csv

# Calculating the polynomial with the degree offering the best RMSE
FOR each actor_timing.csv ∈ CSV
  FOR each dof ∈ DOF
    COMPUTE polynomial(dof)
    COMPUTE RMSE(measure, polynomial)
    SAVE best_RMSE_config → parameterized_actor_timing.csv
```

This will reduce the gap between estimated timing and measured value



[r] S. Wang, N. Gac, H. Miomandre, J.-F. Nezan, K. Desnos, F. Orieux « An Initial Framework for Prototyping Radio-Interferometric Imaging Pipelines»

# Summary conclusion

- On going work:
  - 🔜 Automated radio astronomy imaging benchmarking on HPC systems.
  - 🔜 Dataflow methodology available on Github.
  - 🔜 Comparison with manual implementation [N. Monnier g2g]
  - 🔜 validation on Ruche and Grid5000

  - 🌐 SimSDP Tutorial available on the PREESM website: SimSDP: Multinode Design Space Exploration - Preesm
  - 🌐 Radio astronomy imaging benchmark available on central supelec gitlab: SIMSDP - Generic Imaging Pipeline

- Future work:
  - 🚀 Enhancing SimSDP reliability automating fine-grained description