Exploiting System Level Heterogeneity to improve the performance of a GeoStatistics Multi-phase Task-based Application

50th International Conference on Parallel Processing

Lucas Leandro Nesi, Arnaud Legrand, Lucas Mello Schnorr

August 11, 2021

Institute of Informatics, Federal University of Rio Grande do Sul (UFRGS), Brazil
University Grenoble Alpes, CNRS, Inria, Grenoble INP, LIG, France

Heterogeneity in HPC – At System-Level

System-Level Heterogeneity: Nodes with Different Computational Power

Heterogeneity in HPC – At System-Level

System-Level Heterogeneity: Nodes with Different Computational Power

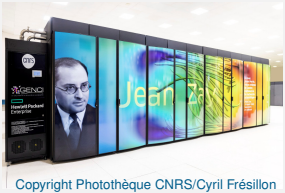
Santos Dummont

Copyright LNCC

Four Partitions/Systems:

- GPU: Xeon 2695 + K40
- Hybrid: Xeon 2695 + Xeon Phi
- CPU: Xeon 2695
- SDumont: Xeon 6252 + V100

Jean Zay



Copyright Photothèque CNRS/Cyril Frésillon

Four Partitions/Systems:

- GPU 1: Xeon 6248 + V100
- GPU 2: Xeon 6226 + V100
- GPU 3: Xeon 6240R + A100
- CPU Only: Xeon 6248

Heterogeneity in HPC – At System-Level

System-Level Heterogeneity: Nodes with Different Computational Power

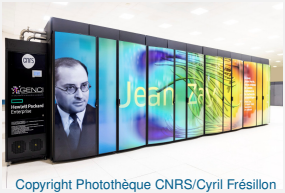
Santos Dummont

Copyright LNCC

Four Partitions/Systems:

- GPU: Xeon 2695 + K40
- Hybrid: Xeon 2695 + Xeon Phi
- CPU: Xeon 2695
- SDumont: Xeon 6252 + V100

Jean Zay

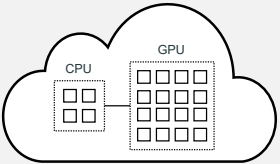


Copyright Photothèque CNRS/Cyril Frésillon

Four Partitions/Systems:

- GPU 1: Xeon 6248 + V100
- GPU 2: Xeon 6226 + V100
- GPU 3: Xeon 6240R + A100
- CPU Only: Xeon 6248

Cloud in HPC



Many Different Instances

Heterogeneity in HPC – At System-Level

System-Level Heterogeneity: Nodes with Different Computational Power

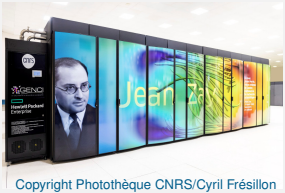
Santos Dummont

Copyright LNCC

Four Partitions/Systems:

- GPU: Xeon 2695 + K40
- Hybrid: Xeon 2695 + Xeon Phi
- CPU: Xeon 2695
- SDumont: Xeon 625

Jean Zay

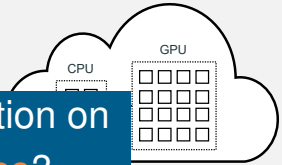


Copyright Photothèque CNRS/Cyril Frésillon

Four Partitions/Systems:

- GPU 1: Xeon 6248 + V100

Cloud in HPC

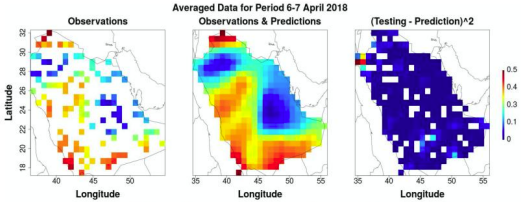


rent Instances

How to distribute an HPC application on these heterogeneous resources?

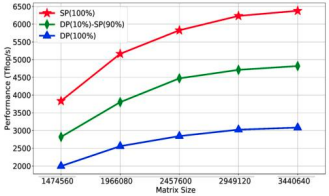
ExaGeoStat – Geostatistics on Manycore Systems

- Design at Kaust (Hatem Ltaief et al., 2018)
- Predict missing observations in climate/weather forecasting applications
- A Task-Based Application that uses Chameleon and StarPU
- Efficiency with Mixed-Precision shown with 4096 nodes



Prediction Accuracy using Space-Time model

Source: https://github.com/ecrc/exageostat



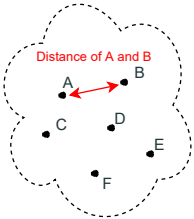
4096 **Shaheen-II** Cray XC40 Nodes

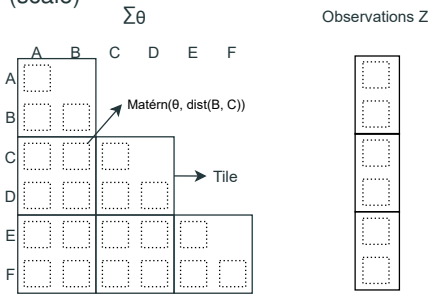
Source: https://github.com/ecrc/exageostat

ExaGeoStat – Fits the Gaussian Process

Their Gaussian Process uses:

- Matérn function, hyper-parameter θ (scale)



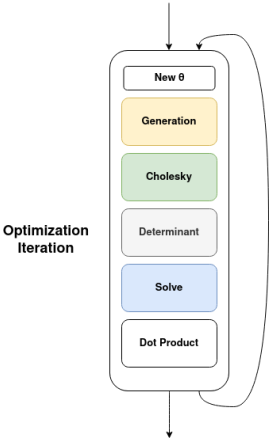


- Optimize θ to maximize Gaussian log-likelihood function:

$$l(\theta) = -\frac{n}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma_\theta| - \frac{1}{2}Z^T\Sigma_\theta^{-1}Z$$

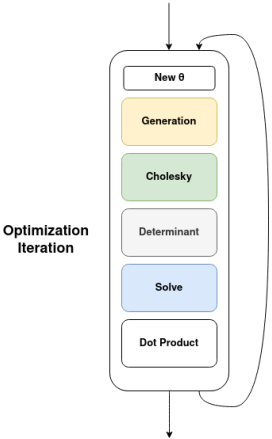
ExaGeoStat – Application DAG Structure

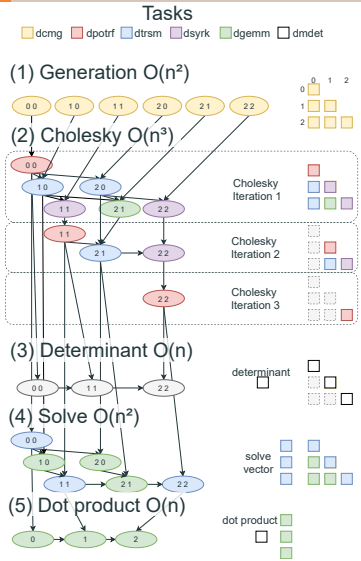
Multiple Phases



ExaGeoStat – Application DAG Structure

Multiple Phases

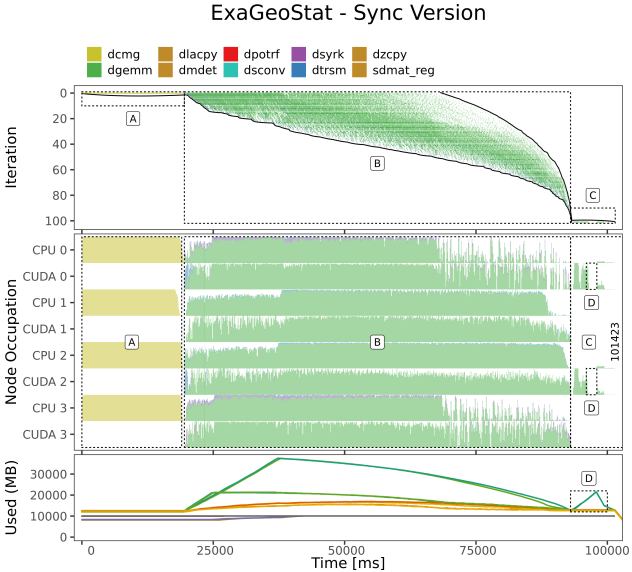




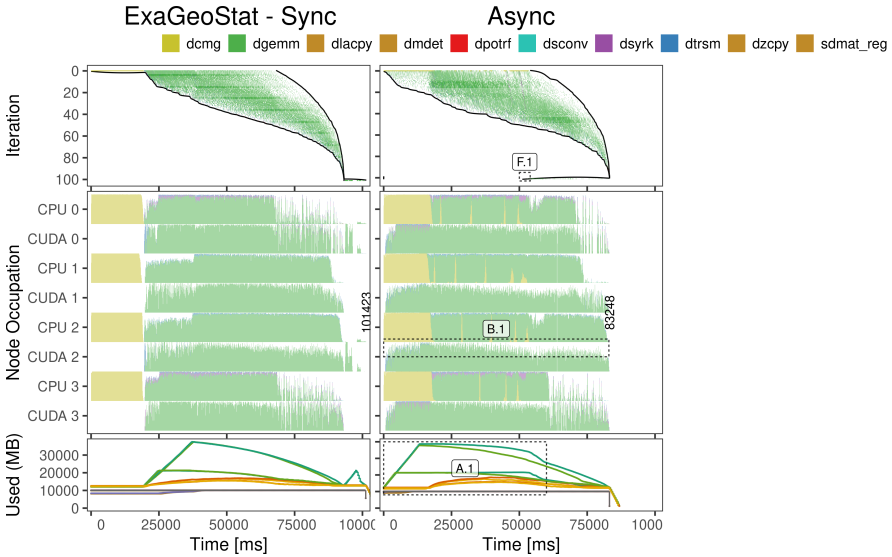
ExaGeoStat - Synchronous Version

Three Major phases

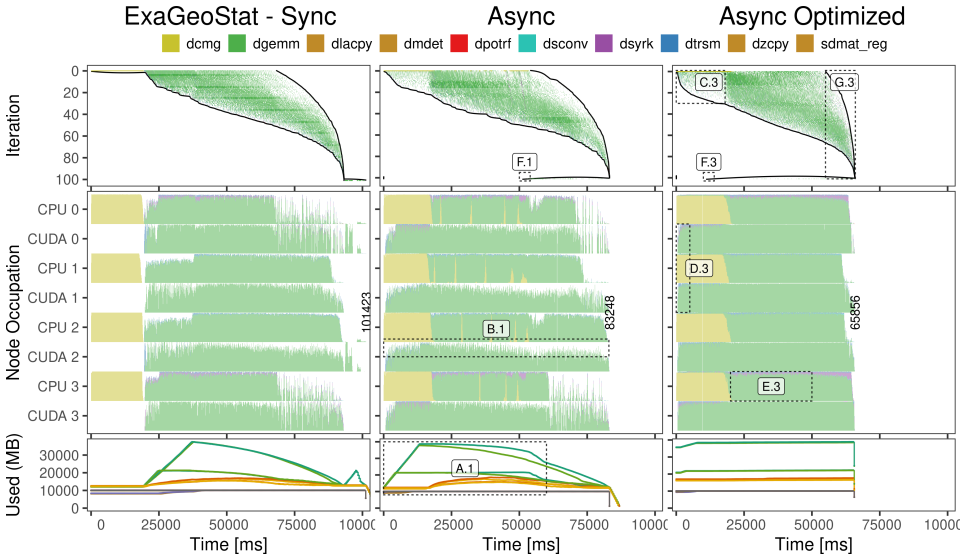
- A: Generation – Matérn
- B: Cholesky
- C: Solve, Determinant, Dot Product



Asynchronization Optimizations Results



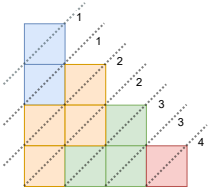
Asynchronization Optimizations Results



Computing Power for Multi-Phase Balancing

Notation

- t – a task from the $\mathcal{T}$ set
- r – a resource from the $\mathcal{R}$ set
- s – a step from the $\mathcal{S}$ set
- G_s – Generation step s end time
- F_s – Factorization step s end time
- $Q_{s,t}$ – Number of tasks t of step s
- $w_{t,r}$ – Duration of task t on resource r
- $\alpha_{s,t,r}$ – The tasks t of step s to be placement on r (Optimized Variable)



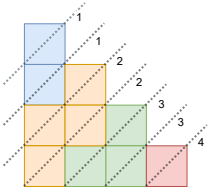
Computing Power for Multi-Phase Balancing

Linear Program

Notation

- t – a task from the $\mathcal{T}$ set
- r – a resource from the $\mathcal{R}$ set
- s – a step from the $\mathcal{S}$ set
- G_s – Generation step s end time
- F_s – Factorization step s end time
- $Q_{s,t}$ – Number of tasks t of step s
- $w_{t,r}$ – Duration of task t on resource r
- $\alpha_{s,t,r}$ – The tasks t of step s to be placement on r (Optimized Variable)

$$\text{Minimize} \sum_{s \in \mathcal{S}} (G_s + F_s) \; s.t. :$$



Computing Power for Multi-Phase Balancing

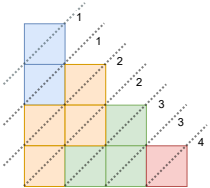
Linear Program

Notation

- t – a task from the $\mathcal{T}$ set
- r – a resource from the $\mathcal{R}$ set
- s – a step from the $\mathcal{S}$ set
- G_s – Generation step s end time
- F_s – Factorization step s end time
- $Q_{s,t}$ – Number of tasks t of step s
- $w_{t,r}$ – Duration of task t on resource r
- $\alpha_{s,t,r}$ – The tasks t of step s to be placement on r (Optimized Variable)

$$\text{Minimize} \sum_{s \in \mathcal{S}} (G_s + F_s) \; s.t. :$$

$$\forall t \in \mathcal{T}, \forall s \in \mathcal{S} : \sum_{r \in \mathcal{R}} \alpha_{s,t,r} = Q_{s,t}$$



Computing Power for Multi-Phase Balancing

Linear Program

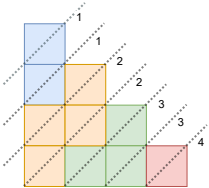
Notation

- t – a task from the $\mathcal{T}$ set
- r – a resource from the $\mathcal{R}$ set
- s – a step from the $\mathcal{S}$ set
- G_s – Generation step s end time
- F_s – Factorization step s end time
- $Q_{s,t}$ – Number of tasks t of step s
- $w_{t,r}$ – Duration of task t on resource r
- $\alpha_{s,t,r}$ – The tasks t of step s to be placement on r (Optimized Variable)

$$\text{Minimize} \sum_{s \in \mathcal{S}} (G_s + F_s) \; s.t. :$$

$$\forall t \in \mathcal{T}, \forall s \in \mathcal{S} : \sum_{r \in \mathcal{R}} \alpha_{s,t,r} = Q_{s,t}$$

$$\forall s > 1, \forall r \in \mathcal{R} : G_{s-1} + \alpha_{s,\text{dcmg},r} w_{\text{dcmg},r} \leq G_s$$



Computing Power for Multi-Phase Balancing

Linear Program

Notation

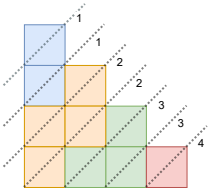
- t – a task from the $\mathcal{T}$ set
- r – a resource from the $\mathcal{R}$ set
- s – a step from the $\mathcal{S}$ set
- G_s – Generation step s end time
- F_s – Factorization step s end time
- $Q_{s,t}$ – Number of tasks t of step s
- $w_{t,r}$ – Duration of task t on resource r
- $\alpha_{s,t,r}$ – The tasks t of step s to be placement on r (Optimized Variable)

$$\text{Minimize} \sum_{s \in \mathcal{S}} (G_s + F_s) \; s.t. :$$

$$\forall t \in \mathcal{T}, \forall s \in \mathcal{S} : \sum_{r \in \mathcal{R}} \alpha_{s,t,r} = Q_{s,t}$$

$$\forall s > 1, \forall r \in \mathcal{R} : G_{s-1} + \alpha_{s,\text{dcmg},r} w_{\text{dcmg},r} \leq G_s$$

$$\forall s \in \mathcal{S}, \forall r \in \mathcal{R} : G_s + \sum_{t \neq \text{dcmg}} \alpha_{s,t,r} w_{t,r} \leq F_s$$



Computing Power for Multi-Phase Balancing

Linear Program

Notation

- t – a task from the $\mathcal{T}$ set
- r – a resource from the $\mathcal{R}$ set
- s – a step from the $\mathcal{S}$ set
- G_s – Generation step s end time
- F_s – Factorization step s end time
- $Q_{s,t}$ – Number of tasks t of step s
- $w_{t,r}$ – Duration of task t on resource r
- $\alpha_{s,t,r}$ – The tasks t of step s to be placement on r (Optimized Variable)

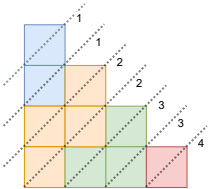
$$\text{Minimize} \sum_{s \in \mathcal{S}} (G_s + F_s) \; s.t. :$$

$$\forall t \in \mathcal{T}, \forall s \in \mathcal{S} : \sum_{r \in \mathcal{R}} \alpha_{s,t,r} = Q_{s,t}$$

$$\forall s > 1, \forall r \in \mathcal{R} : G_{s-1} + \alpha_{s,\text{dcmg},r} w_{\text{dcmg},r} \leq G_s$$

$$\forall s \in \mathcal{S}, \forall r \in \mathcal{R} : G_s + \sum_{t \neq \text{dcmg}} \alpha_{s,t,r} w_{t,r} \leq F_s$$

$$\forall s > 1, \forall r \in \mathcal{R} : F_{s-1} + \sum_{t \neq \text{dcmg}} \alpha_{s,t,r} w_{t,r} \leq F_s$$



Computing Power for Multi-Phase Balancing

Linear Program

Notation

- t – a task from the $\mathcal{T}$ set
- r – a resource from the $\mathcal{R}$ set
- s – a step from the $\mathcal{S}$ set
- G_s – Generation step s end time
- F_s – Factorization step s end time
- $Q_{s,t}$ – Number of tasks t of step s
- $w_{t,r}$ – Duration of task t on resource r
- $\alpha_{s,t,r}$ – The tasks t of step s to be placement on r (Optimized Variable)

$$\text{Minimize} \sum_{s \in \mathcal{S}} (G_s + F_s) \; s.t. :$$

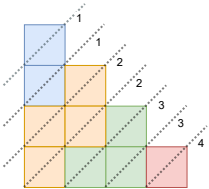
$$\forall t \in \mathcal{T}, \forall s \in \mathcal{S} : \sum_{r \in \mathcal{R}} \alpha_{s,t,r} = Q_{s,t}$$

$$\forall s > 1, \forall r \in \mathcal{R} : G_{s-1} + \alpha_{s,\text{dcmg},r} w_{\text{dcmg},r} \leq G_s$$

$$\forall s \in \mathcal{S}, \forall r \in \mathcal{R} : G_s + \sum_{t \neq \text{dcmg}} \alpha_{s,t,r} w_{t,r} \leq F_s$$

$$\forall s > 1, \forall r \in \mathcal{R} : F_{s-1} + \sum_{t \neq \text{dcmg}} \alpha_{s,t,r} w_{t,r} \leq F_s$$

$$\forall r \in \mathcal{R}, \forall s \in \mathcal{S} : \sum_{z \leq s, t \in T} \alpha_{z,t,r} w_{t,r} \leq F_s$$



Computing Power for Multi-Phase Balancing

Linear Program

Notation

- t – a task from the $\mathcal{T}$ set
- r – a resource from the $\mathcal{R}$ set
- s – a step from the $\mathcal{S}$ set
- G_s – Generation step s end time
- F_s – Factorization step s end time
- $Q_{s,t}$ – Number of tasks t of step s
- $w_{t,r}$ – Duration of task t on resource r
- $\alpha_{s,t,r}$ – The tasks t of step s to be placement on r (Optimized Variable)

$$\text{Minimize} \sum_{s \in \mathcal{S}} (G_s + F_s) \; s.t. :$$

$$\forall t \in \mathcal{T}, \forall s \in \mathcal{S} : \sum_{r \in \mathcal{R}} \alpha_{s,t,r} = Q_{s,t}$$

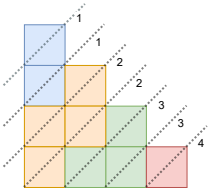
$$\forall s > 1, \forall r \in \mathcal{R} : G_{s-1} + \alpha_{s,\text{dcmg},r} w_{\text{dcmg},r} \leq G_s$$

$$\forall s \in \mathcal{S}, \forall r \in \mathcal{R} : G_s + \sum_{t \neq \text{dcmg}} \alpha_{s,t,r} w_{t,r} \leq F_s$$

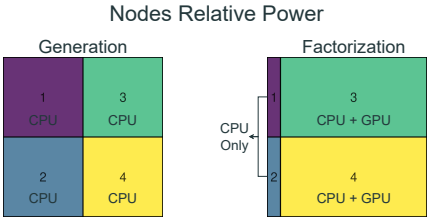
$$\forall s > 1, \forall r \in \mathcal{R} : F_{s-1} + \sum_{t \neq \text{dcmg}} \alpha_{s,t,r} w_{t,r} \leq F_s$$

$$\forall r \in \mathcal{R}, \forall s \in \mathcal{S} : \sum_{z \leq s, t \in T} \alpha_{z,t,r} w_{t,r} \leq F_s$$

$$\min_{r \in \mathcal{R}} (w_{\text{dcmg},r}) \leq G_1$$



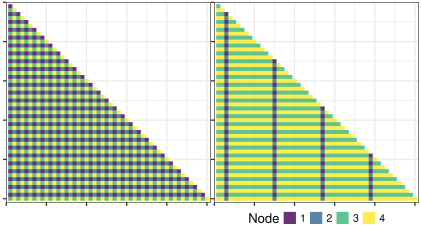
Multi Distributions: Communication Issue



Multi Distributions: Communication Issue

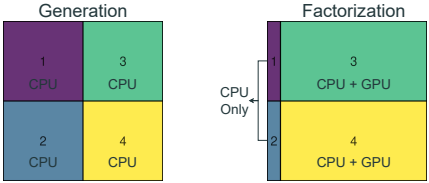
Block-Cyclic independent Generation distribution

1D-1D Factorization distribution



Total Communication Phases: 890

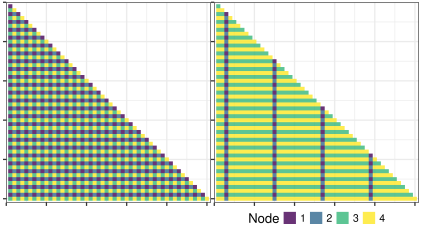
Nodes Relative Power



Multi Distributions: Communication Issue

Block-Cyclic independent Generation distribution

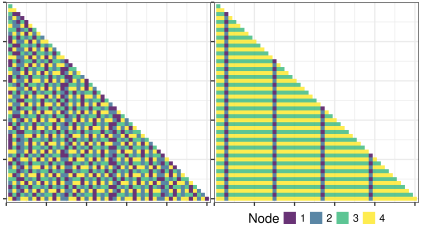
1D-1D Factorization distribution



Total Communication Phases: 890

Improved Generation distribution

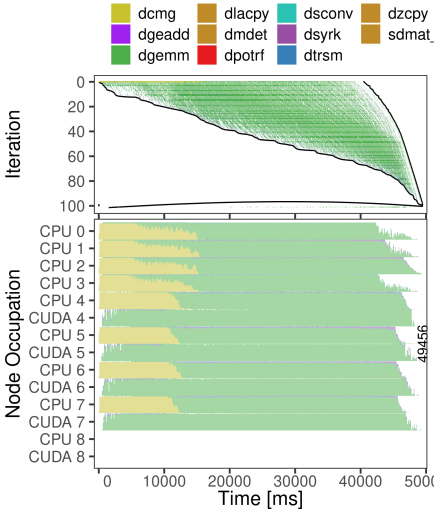
1D-1D Factorization distribution



Total Communication Phases: 517 (-41.9%)

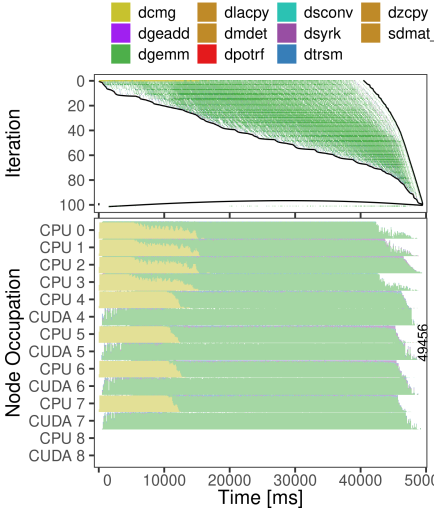
Performance Evaluation

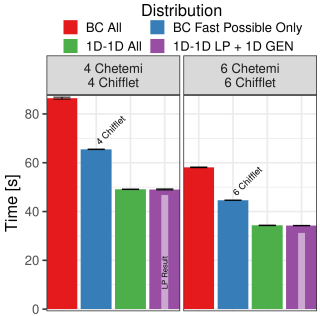
4 Chetemi + 4 Chifflet



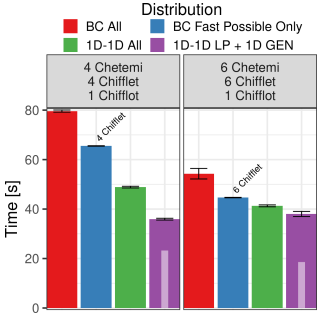
Performance Evaluation

4 Chetemi + 4 Chifflet

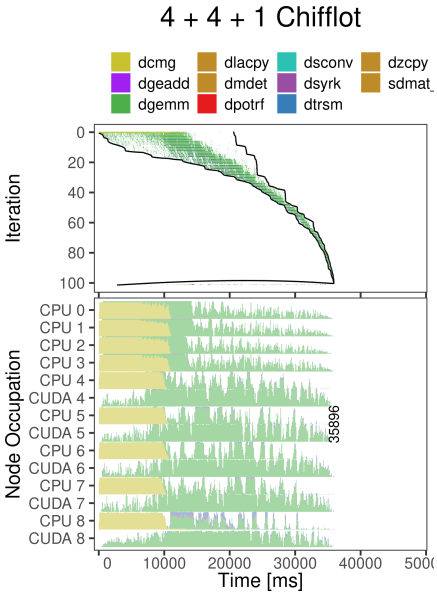


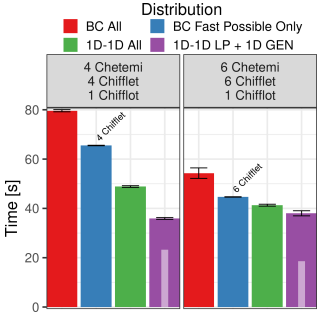


Performance Evaluation

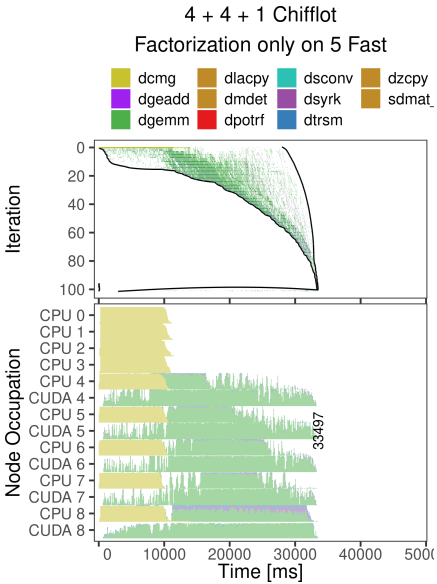


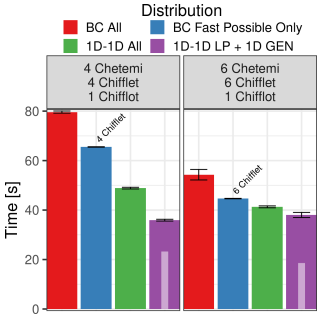
Performance Evaluation





Performance Evaluation





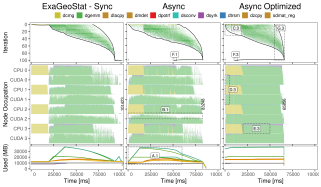
Conclusion

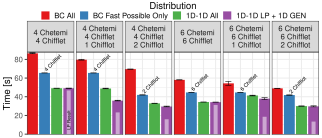
How to distribute an HPC application on these heterogeneous resources?

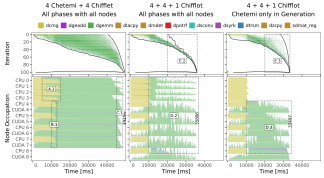
Conclusion

How to distribute an HPC application on these heterogeneous resources?

- Asynchronous Phase Execution with correct Overlapping
- Multi-Phase Distributions is not a individual Task





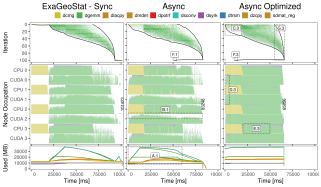


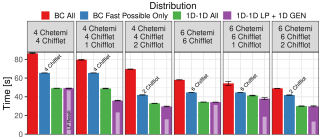
Conclusion

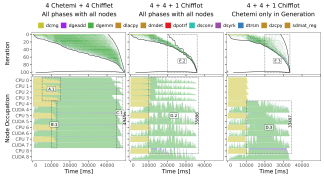
How to distribute an HPC application on these heterogeneous resources?

- Asynchronous Phase Execution with correct Overlapping
- Multi-Phase Distributions is not a individual Task

Correctaly using Node Heterogeneity improves performance







Acknowledgments

This study was financed in part by the “Coordenação de Aperfeiçoamento de Pessoal de Nível Superior” - Brasil (CAPES) - Finance Code 001, the National Council for Scientific and Technological Development (CNPq), grant no 141971/2020-7 to the first author, and the projects:

- FAPERGS (Data Science – 19/711-6, MultiGPU 16/354-8, and GreenCloud – 16/488-9)
- CNPq (447311/2014-0)
- CAPES (Brafitec 182/15, and Cofecub 899/18)

Experiments were carried out using Grid’5000, supported by a scientific interest group hosted by Inria and including CNRS, RENATER and several Universities as well as other organizations (see `https://www.grid5000.fr`).

Thank you!

Questions?

Contact: lucas.nesi@inf.ufrgs.br

Lucas Leandro Nesi, Arnaud Legrand, Lucas Mello Schnorr

August 11, 2021

Institute of Informatics, Federal University of Rio Grande do Sul (UFRGS), Brazil
University Grenoble Alpes, CNRS, Inria, Grenoble INP, LIG, France