Les Cahiers du GERAD ISSN: 0711–2440

Task mapping strategies for electric power system simulations on heterogeneous clusters

J. Durette, G. Karabulut Kurt, A. Lesage-Landry

G–2025–32

April 2025

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2025
– Bibliothèque et Archives Canada, 2025

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Legal deposit – Bibliothèque et Archives nationales du Québec, 2025
– Library and Archives Canada, 2025

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

Task mapping strategies for electric power system simulations on heterogeneous clusters

Julie Durette [a, b, c]

Gunes Karabulut Kurt [a, d]

Antoine Lesage-Landry [a, b, c, d]

[a] *Department of Electrical Engineering, Polytechnique Montréal, Montréal (Qc), Canada, H3T 2A7*

[b] *GERAD, Montréal (Qc), Canada, H3T 1J4*

[c] *Mila, Montréal (Qc), Canada, H2S 3H1*

[d] *Poly-Grames Research Center, Montréal (Qc), Canada, H3T 1J4*

julie.durette@polymtl.ca
gunes.kurt@polymtl.ca
antoine.lesage-landry@polymtl.ca

April 2025
Les Cahiers du GERAD
G–2025–32

Abstract : In this work, we propose improved task mapping strategies for real-time electric power system simulations on heterogeneous computing clusters, considering both heterogeneous communication links and processing capacities, with a focus on bottleneck objectives. We approach the problem through two complementary models : the bottleneck quadratic semi-assignment problem (BQSAP), which optimizes task configuration for a fixed number of computing nodes while minimizing communication and computation costs ; and the variable-size bin packing problem with quadratic communication constraints (Q-VSBPP), which minimizes the required number of computing nodes, particularly valuable for resource provisioning scenarios. We extend the `PuLP` library to solve both problems with the explicit inclusion of communication costs and the processing constraints, and formalizing the nomenclature and definitions for bottleneck objectives in graph partitioning. This formalization fills a gap in the existing literature and provides a framework for the rigorous analysis and application of task mapping techniques to real-time electric power system simulation. Finally, we provide a quantitative study and benchmark the extended `PuLP` library with the `SCOTCH` partitioning library in the context of real-time electromagnetic transient (EMT) simulation task mapping.

Keywords : Task mapping, graph partitioning, real-time electromagnetic transient simulation, heterogeneous computing cluster

Résumé : Dans ce travail, nous proposons des stratégies améliorées de mappage de tâches pour les simulations de systèmes électriques en temps réel sur des grappes de calcul hétérogènes. L'approche considère l'hétérogénéité tant des liaisons de communication que des capacités de calcul et cible spécifiquement l'optimisation relative aux goulots d'étranglement. Nous abordons le problème à travers deux modèles complémentaires : le premier est le problème de semi-assignation quadratique avec goulot d'étranglement (*bottleneck quadratic semi-assignment problem*, BQSAP), qui optimise la configuration des tâches pour un nombre fixe de nœuds de calcul tout en minimisant les coûts de communication et de calcul. Le second modèle est le problème d'emballage optimal de taille variable avec des contraintes de communication quadratiques (*variable-size bin packing problem with quadratic communication constraints*, Q-VSBPP), qui minimise le nombre de nœuds de calcul requis, particulièrement utile pour les scénarios de provisionnement de ressources. Nous étendons la librairie `PuLP` pour résoudre les deux problèmes en incluant explicitement les coûts de communication et les contraintes de traitement, et en formalisant la nomenclature et les définitions des objectifs de goulot d'étranglement dans le partitionnement de graphes. Cette formalisation comble une lacune dans la littérature existante et fournit un cadre pour l'analyse rigoureuse et l'application des techniques de mappage de tâches à la simulation de systèmes électriques en temps réel. Enfin, nous fournissons une étude quantitative et comparons la librairie étendue `PuLP` avec la librairie de partitionnement `SCOTCH` dans le contexte du mappage de tâches de simulation transitoire électromagnétiques (EMT) en temps réel.

Mots clés : Mappage de tâches, partitionnement de graphes, simulation transitoire électromagnétique en temps réel, grappe hétérogène de calculs.

Acknowledgements: This work was funded by Mitacs Accelerate and OPAL-RT Technologies under the grant IT37717.

1 Introduction

Recent international standards, such as IEEE-2800 [1], mandate the use of electromagnetic transient (EMT) simulation tools for analyzing electric power system stability and the integration of renewable energy sources, driving their widespread adoption in the industry. To avoid reliance on large-scale supercomputers, task mapping strategies designed for clusters of modern multi-core computers allow to distribute power system simulations while maintaining real-time performance criteria [19]. To this end, we detail two optimization models and propose corresponding algorithms that improve the current state of EMT simulation task mapping. This is achieved by tailoring general-purpose partitioning algorithms to account for inter-process communication overhead and heterogeneous computing resources, leading to our two problem formulations : the bottleneck quadratic semi-assignment problem (BQSAP) and the variable-size bin packing problem with quadratic constraint for communication costs (Q-VSBPP).

1.1 Related works

Extensive research on the assignment problem and the bin packing problem has been conducted, resulting in a considerable variety of formulations. The fundamental formulation on which we build on for this work is laid out in [18]. The quadratic assignment problem (QAP), which is formulated for facility location optimization, assigns facilities to locations and minimizes both location cost and road distance between facilities. This model adapts well to distributed task assignment because it takes into account both the partition and the data transfer costs. As for the bin packing problem, early work addresses the problem of cutting stock, focusing on minimizing waste when cutting materials from larger rolls [11]. The important aspect of bin packing is that the number of bins, or partitions, also has to be minimized, compared to assignment where the number of partitions is fixed.

Greenberg [17] extends the QAP to the quadratic semi-assignment problem (QSAP), allowing multiple facilities at a single location. This formulation is similar to our problem of assigning multiple tasks to each partition. However, the objective in [17] is the sum of the assignment costs, not the bottleneck costs, as our problem asks for. For extensive reviews of existing variations on the assignment problems, [8] and [10] explore various cost functions, constraints, and solution approaches, highlighting trade-offs between optimality and computational tractability. As for the bin packing problem, [11] synthesizes multiple surveys presenting a large number of variations.

In distributed systems, [4] introduces models for task assignment in the context of parallel computing, taking into account limited computing resources by using bottleneck constraints. This insight is what also drives our formulation. Moreover, recent developments in cloud computing have led to new research directions and monographs [2, 23, 28]. Recent reviews examine assignment problem applications in cloud computing [13] and quadratic bin packing for cloud task partitioning is also explored in [22], though without considering bottleneck objectives.

1.1.1 Algorithmic approaches

Task assignment and bin packing for numerical simulations on cluster computers can be addressed with several algorithmic approaches. Exact methods, such as those explored in [24] provide optimal solutions but are computationally expensive for large problems. For the size of our networks, the resolution time of exact methods is prohibitively high, hence our focus on heuristic algorithms for this study. Metaheuristics offer a compromise between solution quality and runtime and an approach like Silva's tabu search [29] has been applied to several quadratic assignment problem variants, though not BQSAP. Deep learning methods are also currently being explored in the literature, such as end-to-end learning of assignments [16, 31]. However, deep learning requires training data and incur computational costs for both training and inference.

Dedicated graph algorithms are another option for solving task mapping problems, which, compared to metaheuristics, offer an advantage regarding resolution speed. Because they are tailored to work on graphs, they can exploit task dependencies and connectivity patterns [3, 14]. Literature surveys [7, 9] identify persistent challenges in graph partitioning, notably the handling of vastly different node weights, which shifts the problem closer to bin packing, and the need to optimize for bottleneck performance rather than solely edge cut minimization. For the problem at hand, the `SCOTCH` library [26] provides relevant tools to solve both BQSAP and Q-VSBPP. For the BQSAP, it combines multiple algorithms in sequence, including the Fiduccia–Mattheyses algorithm [15]. For the Q-VSBPP, `SCOTCH` uses a dual recursive bipartitioning algorithm [26]. Alternatively, the `PuLP` graph partitioning library [30] offers a flexible framework for scalable multi-objective and multi-criteria optimization that also allows to solve generic partitioning problems on graphs. The `PuLP` library exploits small-world graphs properties [32] for refining the modelling of inter-task communications.

1.1.2 Previous work on EMT simulation task mapping

In prior works, the author of [33] provides a formulation for EMT simulations task partitioning in homogeneous computing environments. References [5, 6] suggest the use of `SCOTCH` partitioning tools for performance improvement over `A-star`-based partitioning. While improving solving times, the task mapping settings only consider fully connected homogeneous computing cluster as tested in [19, 20].

1.2 Contributions

Our work introduces two new formulations : the bottleneck quadratic semi-assignment problem (BQSAP) and the variable-size bin packing problem with bottleneck quadratic constraint for communication costs (Q-VSBPP). Our models improve previous approaches by incorporating bottleneck constraints for heterogeneous clusters, creating a more accurate representation of real-world task assignment scenarios where communication delays impact performance [19].

Our contributions extend the current literature in two main areas.

Modelling

- We formulate the BQSAP which establishes an explicit connection between graph partitioning and QSAP with a specific bottleneck objective structure not previously documented in the existing literature.
- We introduce the Q-VSBPP to focus on minimizing the longest communication delay rather than the sum of communication costs, which more directly impacts real-time performance of EMT simulations.
- We establish a correspondence between small-world graph properties and the task structure of power grid simulations and provide a framework for effective task partitioning strategies.

Algorithmic

- We adapt a label propagation algorithm (`PuLP`) to solve the BQSAP formulation by taking into account communication characteristics between partitions and partition capacities.
- We extend `PuLP` for Q-VSBPP by removing the constraints that fixed the number of partitions, allowing the algorithm to minimize the partition count.
- We benchmark `SCOTCH` on both the BQSAP for EMT simulation task mapping on heterogeneous clusters and the Q-VSBPP for homogeneous fully connected clusters.
- We experimentally show that `SCOTCH` solves the task mapping problem faster than `PuLP`. While `PuLP` has slower resolution time, its flexibility allows us to easily extend the functionalities and solve the heterogeneous problem as well. The mapping solution quality is comparable for problems solved with both solvers.

Next, we present the context of task mapping of EMT simulations (Section 2) and detail the adaptation of graph partitioning techniques into BQSAP and Q-VSBPP (Section 3.1). We then describe the specific modifications made (Section 3.2) and compare the performance of our approach with conventional methods (Section 4). Our contributions aim to extend current practices regarding graph partitioning while providing a more precise mathematical model that represents accurately the task mapping of real-time electric power system simulations.

2 Background

EMT simulations provide high-resolution temporal modelling of electric power systems and their dynamic behaviours [21]. It captures rapid switching events, wave propagation phenomena, and transient responses for the analysis of the system stability, protection requirements, and the equipment performance under various operating conditions. EMT simulation's ability to model electromagnetic phenomena at a micro- to nanosecond resolution enables power engineers to analyze interactions between power electronic devices, control systems, and power system components, for example.

2.1 EMT simulation parallelization

The parallelization of EMT simulations can be achieved through various techniques. In this work, we focus on the approach utilizing the physical propagation delays inherent to transmission lines as its decomposition strategy [12]. When transmission line propagation delays exceed the simulation time step, they create natural boundaries for parallel computations, enabling a distributed processing approach where different network segments can be computed independently while maintaining the solution accuracy through synchronized data transfer between adjacent tasks. The simulation proceeds through discrete time steps, where each time step consists of two distinct phases : a computation phase followed by a data communication phase. In the computation phase, each processor independently solves its assigned network segment using local data. A synchronization barrier ensures that all processes complete this phase before proceeding. The communication phase then enables the exchange of boundary values between adjacent network segments, maintaining solution continuity across the entire system. This sequence of computation-communication repeats for each time step, advancing the simulation forward in time while maintaining synchronization between all parallel processes, as illustrated on Figure 1.

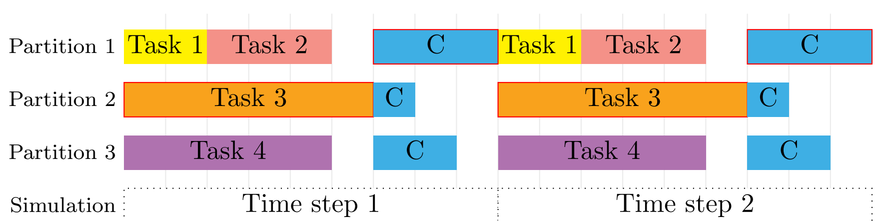


Figure 1 – Gantt chart of a task mapping example, with bottlenecks on communication delay (C) on Partition 1 and on task computation time on Partition 2

2.2 Small-world task graph

The graph representing data transfer between EMT simulation tasks exhibit characteristics of small-world networks, as analyzed by [32]. The graph structure, characterized by a combination of short-range and occasional long-range connections, is similar to the sparse connection pattern observed in typical electric power networks. Graph partitioning algorithms such as the label-propagation algorithm `PuLP` [30] is designed to take advantage of the small-world structure of the graph to identify highly connected nodes and local clusters.

2.3 Notation

We now introduce our formulation. Let $\mathcal{T} = \{1, ..., t\}$ be the set of tasks and $\mathcal{P} = \{1, ..., p\}$ be the set of partitions or the nodes of the cluster, with $t, p \in \mathbb{N}$ the number of tasks and the number of partitions, respectively. Each task $i \in \mathcal{T}$ is characterized by its estimated computation time $e_i > 0$ on a reference node and the volume of data $a_{ij} > 0$ transmitted between tasks i and j. The maximum computation time across all the tasks is denoted by $e_{\max} = \max_{i \in \mathcal{T}} e_i$. The total number of communicating task pairs is given by the ℓ_0 pseudo-norm, $\|A\|_0$, that is the number of non-zero elements in the matrix $A = (a_{ij})_{i,j \in \mathcal{T}}$. The cluster of computers is characterized by a communication delay $b_{kl} > 0$ and a latency $c_{kl} > 0$ between the nodes k and l. Each node $k \in \mathcal{P}$ has a capacity factor $\rho_k > 0$ relative to the reference node that reflects its computational capabilities such as the frequency of the processor clock. Finally, the assignment of a task $i \in \mathcal{T}$ on a node $k \in \mathcal{P}$ is represented by a binary variable $x_{ik} \in \{0, 1\}$. The following section describes the proposed models using this notation.

3 Methodology

We present in this section our mathematical models for task mapping of EMT parallel simulations and the modified algorithm.

3.1 Task mapping optimization models

Using the notation from the Section 2.3, we present below our task mapping models.

3.1.1 Bottleneck quadratic semi-assignment problem (BQSAP) model

Let $y > 0$ be the bottleneck on computing time, and $z > 0$ be the bottleneck on communication time. The BQSAP optimization problem formulation is :

$$\min_{y,z,x_{ik}} \quad y + z \tag{1a}$$

$$\text{s.t.} \quad y \geq \rho_k \sum_{i \in \mathcal{T}} e_i \, x_{ik} \qquad k \in \mathcal{P}, \tag{1b}$$

$$z \geq b_{kl} \sum_{i \in \mathcal{T}} \sum_{j \in \mathcal{T}} a_{ij} \, x_{ik} x_{jl} + c_{kl}, \qquad \forall k, l \in \mathcal{P}, \tag{1c}$$

$$\sum_{k \in \mathcal{P}} x_{ik} = 1, \qquad i \in \mathcal{T}, \tag{1d}$$

$$x_{ik} \in \{0, 1\}, \qquad i \in \mathcal{T}, k \in \mathcal{P}, \tag{1e}$$

where (1a) aims to minimize the total bottleneck time, combining both computation and communication delays. The computing bottleneck (1b) ensures that the minimum time y accounts for the total computational load of each node j considering the node's capacity factor ρ_j. The communication bottleneck (1c) calculates the total communication delay based on the volume of data transmitted and latencies between nodes where the tasks are assigned. Finally, (1d) ensures each task is assigned to exactly one partition and (1e) constrains task assignments to be binary (either assigned or not assigned), maintaining the integrity of the mapping.

This formulation creates an optimization problem that considers both computation and communication bottlenecks while ensuring valid task assignments across the distributed system.

3.1.2 Variable-size bin packing with quadratic constraint for communication costs (Q-VSBPP) model

Let $Y_{\max} > 0$ be the maximum computing time allowed of a partition and $z > 0$ be the bottleneck on communication time. The Q-VSBPP optimization problem formulation is :

$$\min_{z, x_{ik}} \; z \tag{2a}$$

$$\text{s.t.} \quad Y_{\max} \geq \rho_k \sum_{i \in \mathcal{T}} e_i x_{ik}, \qquad k \in \mathcal{P}, \tag{2b}$$

$$z \geq b_{kl} \sum_{i \in \mathcal{T}} \sum_{j \in \mathcal{T}} a_{ij}\, x_{ik} x_{jl} + c_{kl}, \qquad \forall k, l \in \mathcal{P}, \tag{2c}$$

$$\sum_{k \in \mathcal{P}} x_{ik} = 1, \qquad i \in \mathcal{T}, \tag{2d}$$

$$x_{ik} \in \{0, 1\}, \qquad \forall i \in \mathcal{T}, k \in \mathcal{P}, \tag{2e}$$

where (2a) aims to minimize the communication bottleneck time z ; (2b) defines the maximum computing time considering node capacity factors ρ_j ; (2c) models the communication delay bottleneck ; (2d) ensures each task is assigned to exactly one partition ; and (2e) ensures that task assignments are binary.

This model represents Q-VSBPP task assignment problem where the goal is to find the optimal distribution of computing tasks across partitions while respecting a maximum computing time per partition and minimizing the communication overhead.

3.2 Task mapping algorithms

We evaluate the performance of both `PuLP` and `SCOTCH` algorithms. They represent examples of two common paradigms for graph partitioning techniques as identified in [9], respectively, the label propagation and the multilevel paradigms. This allows us to evaluate both approaches for the specific requirements of our optimization objectives and constraints, taking into account the characteristics of the EMT simulation task graph and the architecture of the computing cluster.

3.2.1 `PuLP` for BQSAP and Q-VSBPP

We extend the original `PuLP` label propagation partitioning algorithm to account for the specific needs of BQSAP and Q-VSBPP. First, a constraint is added during vertex and edge balancing to prevent the creation of empty partitions, a situation observed when dealing with smaller graphs than those used in the original work [30] and relevant to solving an assignment problem. Second, the algorithm is modified to incorporate the communication weights a_{ij} and the capacity factor ρ_j for each partition, rather than assuming uniform communication costs and partition capacities. This allows `PuLP` to accurately represent the heterogeneous communication capacities of the cluster and solve the quadratic bottleneck assignment problem. Third, for the bin packing problem, a maximum capacity constraint is introduced, limiting the total weight assigned to any single partition.

Despite this extension, `PuLP` retains certain limitations. Label exchange is restricted to neighbouring partitions, which constrains the achievable balance quality. Additionally, the breadth-first search initialization begins at a randomly selected vertex, rather than the vertex with the highest degree of connectivity, which could potentially lead to a more optimal starting point. Also, while the original `Pulp` algorithm includes initialization, vertex balancing, and edge-balancing phases, the scope of the current implementation was focused on the first two. Assessing the impact of this third phase is left for future investigation.

3.2.2 SCOTCH for BQSAP and Q-VSBPP

For BQSAP, we use the `gmap` program from `SCOTCH` library with the options `-cbq -b0` to enable the balancing of both task computation time and the communication delay [26, 27]. For Q-VSBPP, we use the `gmap` program with the options `-q` for the maximum number of partitions and `-cr` for the algorithm of bin packing as described in [26].

4 Numerical results

We present in this section the results of numerical examples based on the models described above.

4.1 Numerical setting

To evaluate task assignment strategies, we used the following setting. For reproducibility, we provide our modification of the `PuLP` library, [1] and the `SCOTCH` library source code is provided by the INRIA laboratory. [2] The `PuLP` execution is set to use four cores and the `SCOTCH` execution is sequential. The numerical computations are executed of a `i7-2.30GHz-16GB` laptop computer.

4.1.1 Cluster characteristics

The clusters considered for the numerical examples have the following characteristics :

— A 30-node heterogeneous cluster is considered for BQSAP. It consists of a 18-node and a 12-node server, where the former has twice the computing capacity, e.g., processor clock frequency, of the latter.

— A 72-node homogeneous cluster is considered for Q-VSBPP. It represents fully connected nodes with similar computing capacity.

— A 72-node heterogeneous cluster is finally considered for Q-VSBPP. It consists of a cluster of two 18-node and three 12-node servers, for which the two former have twice the computing capacity, e.g., processor clock frequency, of the three latter.

The data communication delay between servers is based on experimental tests from [25]. It defines the linear relation of (1c) and (2c) with a delay of $b_{kl} = 0.0056$ microsecond per quantity of data in `Doubles` transferred between the partitions k and l and a latency $c_{kl} = 2.7$ microseconds. For partitions located within a server, the delay and latency for data communication is set to a hundredth of the ones between servers.

4.1.2 Tasks characteristics

The computation tasks used for the numerical examples are taken from real-world configurations of EMT simulations representing five large-scale electric power systems, as described previously in Section 2.1. Key characteristics of these tasks are detailed in Table 1. Tasks without any communications were removed from the list during the preprocessing phase because our models focus on the integration of communications.

4.2 BQSAP on a heterogeneous cluster

We now present the results obtained for the BQSAP on a heterogeneous cluster, utilizing both `PuLP` and `SCOTCH` solvers. Table 2 shows that the task computation time bottlenecks exceed the longest task's computation time for Networks B, D, and E shown in Table 1. This outcome is attributed to multiple tasks being assigned to the same processors, as per the definition the problem as one of semi-assignment.

1. https://github.com/julie9/PuLP-QSAP
2. https://gitlab.inria.fr/SCOTCH/SCOTCH

Table 1 – Characteristics of the computation tasks of EMT simulations of five electric power systems

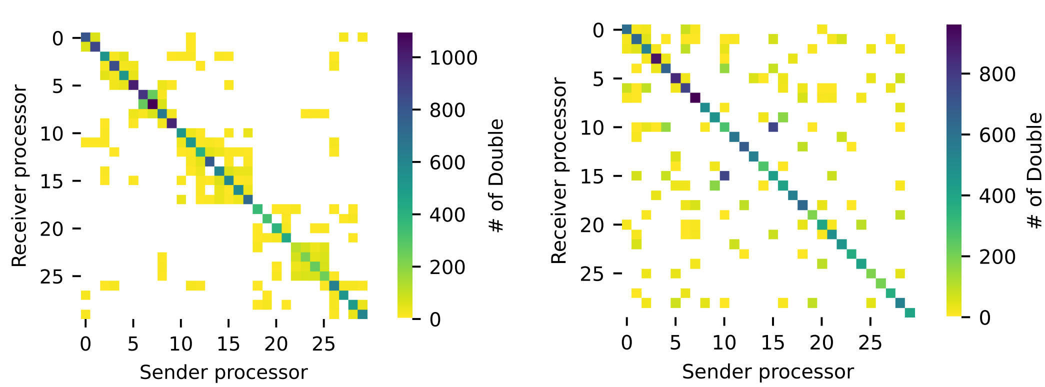
Network	t	$\|A\|_0$	$e_{\max}$ [µs]
A	553	747	17.3
B	435	545	27.4
C	46	53	140.0
D	1106	1295	17.0
E	457	539	102.3

However, we observe that PuLP's solutions exhibit total simulation time steps that are 18% to 108% longer compared to the ones with SCOTCH. In term of solver performance, PuLP requires between 0.5 and 9.5 seconds to compute a solution, while SCOTCH consistently solves the problem in under 0.1 second.

Finally, Figure 2 compares the behaviour of the solver on Network D in terms of communication patterns. SCOTCH demonstrates a tendency to group communications more efficiently on adjacent partitions. This leads to a lower communication delay overall, as noted in Table 2. While PuLP results exhibit less tightly grouped communications, a discernible pattern of grouping communications on faster links is still observed.

Table 2 – BQSAP on a 30-node heterogeneous cluster

Network	Solver	Solver time [s]	Bottleneck		Simulation time step [µs]
			Task computation time [µs]	Communication delay [µs]	
A	SCOTCH	0.05046	17.32	19.79	37.11
	PuLP	1.57716	17.32	59.70	77.02
B	SCOTCH	0.02306	52.68	47.34	100.02
	PuLP	1.83637	77.50	41.44	118.94
C	SCOTCH	0.00071	140.00	14.00	154.00
	PuLP	0.41699	160.34	22.14	182.48
D	SCOTCH	0.10255	41.01	19.41	60.42
	PuLP	9.56950	45.63	31.13	76.76
E	SCOTCH	0.02227	151.59	19.77	171.36
	PuLP	4.53063	242.17	16.97	259.14



(a) SCOTCH (b) PuLP

Figure 2 – Communication volume, for BQSAP of Network D on a heterogeneous cluster

4.3 Q-VSBPP on a homogeneous cluster

Next, we compare the results obtained for Q-VSBPP on a homogeneous cluster using `SCOTCH` and `PuLP`. Q-VSBPP seeks to minimize the number of partitions required. The maximal computing time $Y_{\max}$ is fixed by the computation time of the longest task as presented in Table 1.

Table 3 illustrates that the total simulation time step, calculated as the sum of the communication delay and task computation time bottlenecks, is similar for both solvers, with differences ranging from 0% to 2%. The distribution of task computing time across partitions is visualized in Figure 3 for Network D. The number of partitions found by each solver is generally comparable, within a range of ±13%, with a notable exception on Network A, where `PuLP` identifies nearly twice as many partitions. This is potentially due to the limitation in `PuLP`'s algorithm, identified in Section 3.2.1 : its inability to exchange labels that are not neighbours, which may hinder the minimization of partitions. In terms of solver resolution time, a significant difference is observed : `PuLP` requires between 0.4 and 9.6 seconds, whereas `SCOTCH` consistently solves the problem in under 0.1 second.

Table 3 – Q-VSBPP on a 72-node homogeneous cluster

Network	Solver	Solver time [s]	Number of partitions found	Bottleneck: Task computation time [µs]	Bottleneck: Communication delay [µs]	Simulation time step [µs]
A	`SCOTCH`	0.00994	19	17.30	0.38	17.68
	`PuLP`	2.69453	55	17.30	0.79	18.09
B	`SCOTCH`	0.00980	72	27.40	0.74	28.14
	`PuLP`	5.47200	63	27.40	0.69	28.09
C	`SCOTCH`	0.00109	10	140.00	0.17	140.17
	`PuLP`	0.52730	14	140.00	0.22	140.22
D	`SCOTCH`	0.03035	67	16.98	0.20	17.18
	`PuLP`	2.53274	72	16.98	0.42	17.40
E	`SCOTCH`	0.01179	57	102.30	0.20	102.50
	`PuLP`	3.04640	58	102.30	0.31	102.61

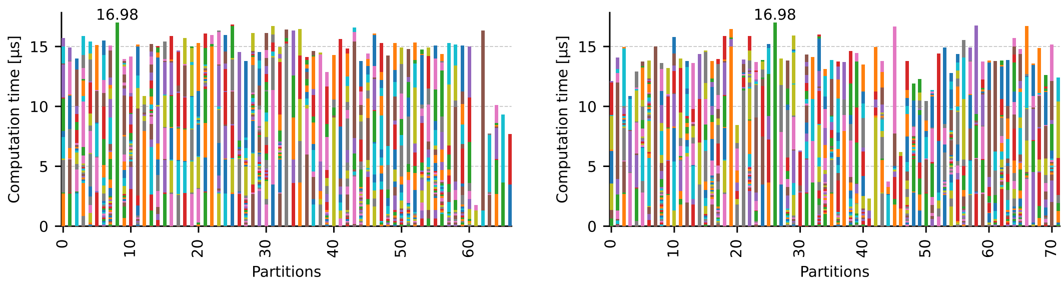
Figures 3 and 4 depict the communication patterns for both solvers on Network D. Because the experimental setup employed a homogeneous communication pattern between processing units, a relatively uniform distribution of communication between the partitions is observed, as shown in Figure 4a. Consistent with the results of the previous section, `PuLP` exhibits less compact communication patterns when compared to `SCOTCH`.

4.4 Q-VSBPP on a heterogeneous cluster

Lastly, we present results for Q-VSBPP on a 72-node heterogeneous cluster obtained using `PuLP`. At the time of this analysis, `SCOTCH`'s support for bin packing on heterogeneous clusters is limited, and therefore could not be used for this study. For the maximal computing time $Y_{\max}$, it is fixed to the longest task's computation time as presented in Table 1.

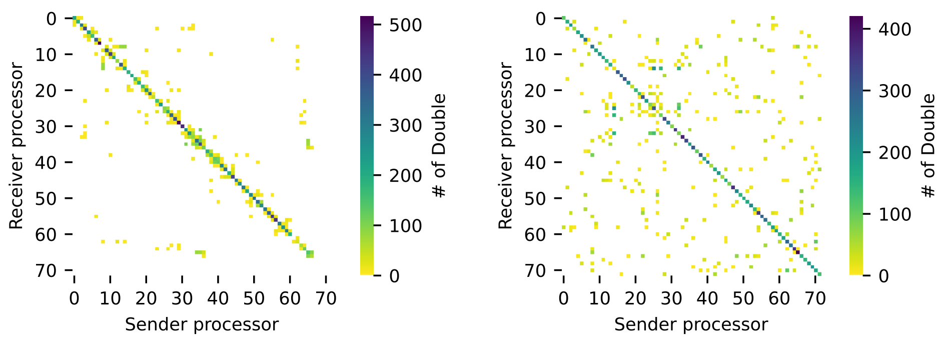
Table 4 illustrates that solver resolution times are consistently around 0.3 second. The bin packing solutions utilize between 55 and 68 nodes out of the 72 available ones. The communication delay constitutes a more significant portion of the total simulation time step, e.g., up to 79% on Network A, due to the heterogeneous communication characteristics of the cluster. Namely, data communication between more distant nodes incurs longer delays.

Figure 5 shows the task mapping for Network D on 68 partitions, where the 36 first partitions have larger computing capacity and get assigned a larger quantity of computing tasks. For the communication volume, while less concentrated then on previous `SCOTCH` results, we observe that the mapping mainly groups communicating tasks on the same partition, thus reducing communication load in the cluster.



(a) SCOTCH (b) PuLP

Figure 3 – Task computing time, for Q-VSBBP of Network D on a homogeneous cluster

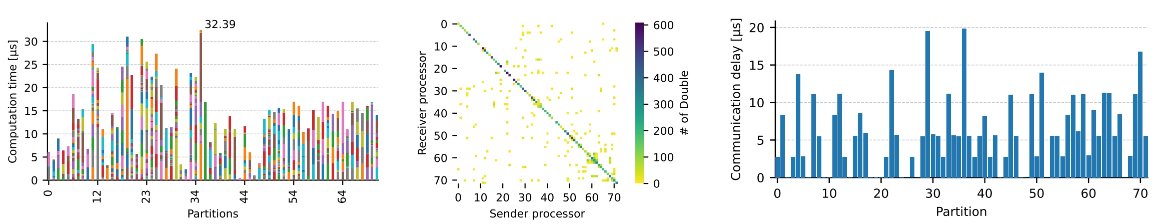


(a) SCOTCH (b) PuLP

Figure 4 – Communication volume, for Q-VSBBP of Network D on a homogeneous cluster

Table 4 – Q-VSBPP on a 72-node heterogeneous cluster using PuLP

Network	Solver Time [s]	Number of partitions found	Bottleneck Task computation time [µs]	Bottleneck Communication delay [µs]	Simulation time step [µs]
A	0.3271	55	17.35	64.74	82.09
B	0.2815	58	26.12	57.77	83.88
C	0.1884	7	207.69	0.08	207.77
D	0.3133	68	16.20	28.11	44.30
E	0.3117	60	98.74	25.14	123.88



(a) Task computing time (b) Communication volume (c) Total communication time delay

Figure 5 – Q-VSBPP of the Network D on a heterogeneous cluster using PuLP

5 Discussion

The results of our research demonstrate several meaningful contributions to the field of task assignment in heterogeneous computing environments. By bridging the gap between graph partitioning algorithms and operations research models, we show empirically that existing tools can be effectively adapted to address specific challenges in electric power system simulations.

Our adaptation of graph partitioning algorithms to solve BQSAP and Q-VSBPP problems puts forward the versatility of these approaches when properly customized. The performance improvements observed in our experiments suggest that graph-based partitioning methods, when combined with accurate models, can offer practical advantages over traditional operational research techniques such as exact algorithms and metaheuristics. This is particularly evident in the context of large scale simulations, where quick resolution times are crucial.

The introduction of the bottleneck constraint in both BQSAP and Q-VSBPP formulations addresses a significant practical concern in heterogeneous computing environments. Our results indicate that this constraint effectively captures the performance limitations inherent in real-world clusters, leading to task assignments that take into account bottlenecks on capacity and communication costs. The ability to account for worst-case communication delays represents a meaningful step forward in ensuring consistent performance of real-time simulations.

The computational efficiency achieved through our approach has practical implications for the EMT simulations. The faster resolution times enable more frequent reconfigurations of task assignments, which is particularly valuable when dealing with electric power system changes during the design phase. This capability allows to test efficiently multiple configurations of the power grid.

6 Conclusion

In this work, we experimentally illustrate the effectiveness of graph partitioning algorithms, specifically for the bottleneck quadratic semi-assignment problem (BQSAP) and the variable-size bin packing problem with quadratic communication constraints (Q-VSBBP), for the task-mapping of electromagnetic transient (EMT) simulations. The results obtained with the extension of `PuLP` highlight its flexibility regarding cluster architecture as well as the problem formulation, which allows its adaptation to both the assignment and the bin packing problems. However, compared to the results with `SCOTCH`, the partitioning generated by `PuLP` resulted in higher communication delay. Future works include extending the capabilities of `SCOTCH` to include variable-size bin packing with support for heterogeneous clusters. While the experimental results presented are conclusive for our application, further analysis of the optimality gap needs to be explored for problems requiring even higher precision. Finally, while the current speed of the algorithms is sufficient for the size of networks studied, for larger networks, the remapping tools available within the `SCOTCH` library, designed to minimize modifications to existing partitioning, could be investigated and applied to larger EMT simulation problems.

Références

[1] IEEE-2800-2022 Standard for Interconnection and Interoperability of Inverter-Based Resources (IBRs) Interconnecting with Associated Transmission Electric Power Systems, April 2022.

[2] Aaron Becker, Gengbin Zheng, and Laxmikant V. Kalé. Load Balancing, Distributed Memory. In Encyclopedia of Parallel Computing, pages 1043–1051. Springer US, Boston, MA, 2011.

[3] Charles-Edmond Bichot and Patrick Siarry, editors. Graph Partitioning. Wiley, 1 edition, February 2013.

[4] Shahid H. Bokhari. Assignment Problems in Parallel and Distributed Computing, volume 32 of The Kluwer International Series in Engineering and Computer Science. Springer US, Boston, MA, 1987.

[5] Boris Bruned. Amélioration de la vitesse de calcul et de la précision des outils de simulation des phénomènes transitoires électromagnétiques sur les réseaux électriques. These de doctorat, Nantes Université, January 2023.

[6] Boris Bruned, Pierre Rault, Sébastien Dennetière, and Ian Menezes Martins. Use of efficient task allocation algorithm for parallel real-time EMT simulation. Electric Power Systems Research, 189 :106604, December 2020.

[7] Aydın Buluç, Henning Meyerhenke, Ilya Safro, Peter Sanders, and Christian Schulz. Recent Advances in Graph Partitioning. In Algorithm Engineering : Selected Results and Surveys, pages 117–158. Springer International Publishing, Cham, 2016.

[8] Rainer Burkard, Mauro Dell'Amico, and Silvano Martello. Assignment Problems. Other Titles in Applied Mathematics. Society for Industrial and Applied Mathematics, January 2012.

[9] Ümit Çatalyürek, Karen Devine, Marcelo Faraj, Lars Gottesbüren, Tobias Heuer, Henning Meyerhenke, Peter Sanders, Sebastian Schlag, Christian Schulz, Daniel Seemaier, and Dorothea Wagner. More Recent Advances in (Hyper)Graph Partitioning. ACM Comput. Surv., 55(12) :253 :1–253 :38, March 2023.

[10] Eranda Çela. The Quadratic Assignment Problem, volume 1 of *Combinatorial Optimization*. Springer US, Boston, MA, 1998.

[11] Maxence Delorme, Manuel Iori, and Silvano Martello. Bin packing and cutting stock problems : Mathematical models and exact algorithms. European Journal of Operational Research, 255(1) :1–20, November 2016.

[12] S. Dennetiere, B. Bruned, H. Saad, and E. Lemieux. Task Separation for Real-Time Simulation of the CIGRE DC Grid Benchmark. In 2018 Power Systems Computation Conference (PSCC), pages 1–7, June 2018.

[13] Nisha Devi, Sandeep Dalal, Kamna Solanki, Surjeet Dalal, Umesh Kumar Lilhore, Sarita Simaiya, and Nasratullah Nuristani. A systematic literature review for load balancing and task scheduling techniques in cloud computing. Artificial Intelligence Review, 57(10) :276, September 2024.

[14] K Erciyes. Guide to Graph Algorithms. Texts in Computer Science. Springer International Publishing, Cham, 2018.

[15] C.M. Fiduccia and R.M. Mattheyses. A Linear-Time Heuristic for Improving Network Partitions. In 19th Design Automation Conference, pages 175–181, Las Vegas, NV, USA, 1982. IEEE.

[16] Alice Gatti, Zhixiong Hu, Tess Smidt, Esmond G. Ng, and Pieter Ghysels. Deep Learning and Spectral Embedding for Graph Partitioning. In Proceedings of the 2022 SIAM Conference on Parallel Processing for Scientific Computing. arXiv, December 2021.

[17] Harold Greenberg. A quadratic assignment problem without column constraints. Naval Research Logistics Quarterly, 16(3) :417–421, 1969.

[18] Eugene L. Lawler. The Quadratic Assignment Problem. Management Science, 9(4) :586–599, July 1963.

[19] P. Le-Huy, S. Guérette, and F. Guay. Performance evaluation of communication fabrics for offline parallel electromagnetic transient simulation based on MPI. Electric Power Systems Research, 223 :109629, October 2023.

[20] P Le-Huy, M Woodacre, S Guérette, and É Lemieux. Massively Parallel Real-Time Simulation of Very-Large-Scale Power Systems. In International Conference on Power Systems Transients, Seoul, South Korea, June 2017.

[21] Jan Machowski, Janusz W. Bialek, and J. R. Bumby. Power System Dynamics : Stability and Control. Wiley, Chichester, U.K., 2nd ed edition, 2008.

[22] Fanchao Meng, Bo Cao, Dianhui Chu, Qingran Ji, and Xuequan Zhou. Variable neighborhood search for quadratic multiple constraint variable sized bin-packing problem. Computers & Operations Research, 143 :105803, July 2022.

[23] Anwesha Mukherjee, Debashis De, and Rajkumar Buyya, editors. Resource Management in Distributed Systems, volume 151 of Studies in Big Data. Springer Nature Singapore, Singapore, 2024.

[24] Tony Nowatzki, Michael C. Ferris, Karthikeyan Sankaralingam, Cristian Estan, Nilay Vaish, and David Allen Wood. Optimization and Mathematical Modeling in Computer Architecture. Springer, Cham, Switzerland, 2014.

[25] OPAL-RT TECHNOLOGIES. Dolphin with Orchestra Performance Testing. Data and report PXH830_Dolphin_Bench_Tool, Montreal.

[26] Francois Pellegrini. Scotch and libScotch 7.0 User's Guide, August 2024.

[27] François Pellegrini and Cédric Lachat. Process Mapping onto Complex Architectures and Partitions Thereof. Report, Inria Bordeaux Sud-Ouest, December 2017.

[28] Thomas Rauber and Gudula Rünger. Parallel Programming : For Multicore and Cluster Systems. Springer International Publishing, Cham, 2023.

[29] Allyson Silva, Leandro C. Coelho, and Maryam Darvish. Quadratic assignment problem variants : A survey and an effective parallel memetic iterated tabu search. European Journal of Operational Research, 292(3) :1066–1084, August 2021.

[30] George M. Slota, Kamesh Madduri, and Sivasankaran Rajamanickam. PuLP : Scalable multi-objective multi-constraint partitioning for small-world networks. In 2014 IEEE International Conference on Big Data (Big Data), pages 481–490, October 2014.

[31] Zhentao Tan and Yadong Mu. Learning solution-aware transformers for efficiently solving quadratic assignment problem. In Proceedings of the 41st International Conference on Machine Learning, volume 235 of ICML'24, pages 47627–47648, July 2024.

[32] Duncan J. Watts and Steven H. Strogatz. Collective dynamics of 'small-world' networks. Nature, 393(6684) :440–442, June 1998.

[33] Tony Wong. Répartition automatique des tâches parallèles : Application dans la simulation de réseaux électriques en temps réel. PhD thesis, Ecole Polytechnique, Montreal (Canada), 1999.