A Privacy-preserving Method to Optimize Distributed Resource Allocation

Olivier Beaude, Pascal Benchimol, Stéphane Gaubert, Paulin Jacquot, Nadia Oudjane

A PRIVACY-PRESERVING METHOD TO OPTIMIZE DISTRIBUTED RESOURCE ALLOCATION *

OLIVIER BEAUDE †, PASCAL BENCHIMOL‡, STÉPHANE GAUBERT §, PAULIN JACQUOT†§, AND NADIA OUDJANE†

Abstract. We consider a resource allocation problem involving a large number of agents with individual constraints subject to privacy, and a central operator whose objective is to optimize a global, possibly nonconvex, cost while satisfying the agents' constraints, for instance an energy operator in charge of the management of energy consumption flexibilities of many individual consumers. We provide a privacy-preserving algorithm that does compute the optimal allocation of resources, avoiding each agent to reveal her private information (constraints and individual solution profile) neither to the central operator nor to a third party. Our method relies on an aggregation procedure: we compute iteratively a global allocation of resources, and gradually ensure existence of a disaggregation, that is individual profiles satisfying agents' private constraints, by a protocol involving the generation of polyhedral cuts and secure multiparty computations (SMC). To obtain these cuts, we use an alternate projection method, which is implemented locally by each agent, preserving her privacy needs. We address especially the case in which the local and global constraints define a transportation polytope. Then, we provide theoretical convergence estimates together with numerical results, showing that the algorithm can be effectively used to solve the allocation problem in high dimension, while addressing privacy issues.

1. Introduction.

1.1. Motivation. Consider an operator of an electricity microgrid optimizing the joint production schedules of renewable and thermal power plants in order to satisfy, at each time period, the consumption constraints of its consumers. To optimize power generation or market costs and the integration of renewable energies, this operator relies on demand response techniques, that is, taking advantage of the flexibilities of some of the consumers electric appliances—those which can be controlled without impacting the consumer's comfort, as electric vehicles or water heaters [18]. However, for privacy reasons, consumers are not willing to provide neither their consumption constraints nor their consumption profiles to a central operator or any third party, as this information could be used to infer private information such as their presence at home.

The *global problem* of the operator is to find an allocation of power (aggregate consumption) $\boldsymbol{p} = (p_t)_t$ at each time period (*resource*) $t \in \mathcal{T}$, such that $\boldsymbol{p} \in \mathcal{P}$ (feasibility constraints of power allocation, induced by the power plants constraints). Besides, this aggregate allocation has to match an individual consumption profile $\boldsymbol{x}_n = (x_{n,t})_{t\in\mathcal{T}}$ for each of the consumer (agent) $n \in \mathcal{N}$ considered. The problem can be written as follows:

$$\min_{\boldsymbol{x}\in\mathbb{R}^{N\times T},\ \boldsymbol{p}\in\mathcal{P}} f(\boldsymbol{p}) \tag{1.1a}$$

$$\boldsymbol{x}_n \in \mathcal{X}_n,\ \forall n \in \mathcal{N} \tag{1.1b}$$

$$\sum_{n\in\mathcal{N}} x_{n,t} = p_t,\ \forall t \in \mathcal{T}\ , \tag{1.1c}$$

The (aggregate) allocation $\boldsymbol{p}$ can be made *public*, that is, revealed to all agents. However, the individual constraint set $\mathcal{X}_n$ and individual profiles $\boldsymbol{x}_n$ constitute *private* information of agent n, and should not be revealed to the operator or any third party.

It will be helpful to think of problem (1.1) as the combination of two interdependent subproblems:

i) given an aggregate allocation $\boldsymbol{p}$, the *disaggregation problem* consists in finding, for each agent n, an individual profile $\boldsymbol{x}_n$ satisfying her individual constraint (1.1b), so that constraint (1.1c) is satisfied,

*Submitted to the editors August 12, 2019. Some of the present results have been announced in the conference proceeding [19].

†EDF R&D, OSIRIS, Palaiseau, France ([olivier.beaude,nadia.oudjane]@edf.fr)

‡CMAP, Ecole polytechnique, CNRS and Triscale Innov, Palaiseau, France (pascal.benchimol@polytechnique.edu).

§Inria and CMAP, Ecole polytechnique, CNRS, Palaiseau, France ([stephane.gaubert,paulin.jacquot]@inria.fr)

or equivalently, solve the problem:

$$\text{FIND } \boldsymbol{x} \in \mathcal{Y}_{\boldsymbol{p}} \cap \mathcal{X} \tag{1.2a}$$

$$\text{where } \mathcal{Y}_{\boldsymbol{p}} \stackrel{\text{def}}{=} \{\boldsymbol{y} \in \mathbb{R}^{NT} | \boldsymbol{y}^\top \mathbb{1}_N = \boldsymbol{p}\} \text{ and } \mathcal{X} \stackrel{\text{def}}{=} \prod_{n \in \mathcal{N}} \mathcal{X}_n \ . \tag{1.2b}$$

When (1.2) has a solution, we say that a *disaggregation* exists for $\boldsymbol{p}$;

ii) For a given subset $\mathcal{Q} \subset \mathcal{P}$, we define the *master problem*,

$$\min_{\boldsymbol{p} \in \mathcal{Q}} f(\boldsymbol{p}) \ . \tag{1.3}$$

When $\mathcal{Q}$ is precisely the set of aggregate allocations for which a disaggregation exists, the optimal solutions of the master problem correspond to the optimal solutions of (1.1).

Aside from the example above, *resource allocation problems* (optimizing common resources shared by multiple agents) with the same structure as (1.1), find many applications in energy [25, 18], logistics [22], distributed computing [24], health care [29] and telecommunications [37]. In these applications, several entities or agents (e.g. consumers, stores, tasks) share a common resource (energy, products, CPU time, broadband) which has a global cost for the system. For large systems composed of multiple agents, the dimension of the overall problem can be prohibitive: a solution is to rely on decomposition and distributed approaches [7, 28, 32]. Besides, agents' individual constraints are often subject to privacy issues [17]. These considerations have paved the way to the development of privacy-preserving, or non-intrusive methods and algorithms, e.g. [36, 20].

In this work, except in Section 4, we consider that each agent $n \in \mathcal{N}$ has a global demand constraint (e.g. energy demand or product quantity), which confers to the disaggregation problem the particular structure of a transportation polytope [8]: the sum over the agents is fixed by the aggregate solution $\boldsymbol{p}$, while the sum over the T resources are fixed by the agent global demand constraint. Besides, individual constraints can also include minimal and maximal levels on each resource, as for instance electricity consumers require, through their appliances, a minimal and maximal power at each time period.

1.2. Main Results. The main contribution of the paper is to provide a non-intrusive and distributed algorithm (Algorithm 3.4) that computes an aggregate resource allocation $\boldsymbol{p}$, optimal solution of the—possibly nonconvex—optimization problem (1.1), along with feasible individual profiles $\boldsymbol{x}$ for agents, without revealing the individual constraints of each agent to a third party, either another agent or a central operator. The algorithm solves iteratively instances of *master problems* $\min_{\boldsymbol{p} \in \mathcal{P}^{(s)}} f(\boldsymbol{p})$ by constructing successive approximations $\mathcal{P}^{(s)} \subset \mathcal{P}$ of the aggregate feasible set of (1.1) for which a disaggregation exists, by adding to the set $\mathcal{P}^{(s)}$ a new constraint on $\boldsymbol{p}$ (i.e. a cutting plane), before solving the next master problem. We shall see that this cutting plane can computed and added to the master problem without revealing any individual information on the agents.

More precisely, to identify whether or not disaggregation (1.2) is feasible and to add a new constraint in the latter case, our algorithm relies on the alternate projections method (APM) [31, 14] for finding a point in the intersection of convex sets. Here, we consider the two following sets: on the one hand, the affine space of profiles $\boldsymbol{x} \in \mathbb{R}^{NT}$ aggregating to a given resource allocation $\boldsymbol{p}$, and on the other hand, the set defined by all agents individual constraints (demands and bounds). As the latter is defined as a Cartesian product of each agent's feasibility set, APM can operate in a distributed fashion. The sequence constructed by the APM converges to a single point if the intersection of the convex sets is nonempty, and it converges to a periodic orbit of length 2 otherwise. If the APM converges to a periodic orbit, meaning that the disaggregation is not feasible, we construct from this orbit a polyhedral *cut*, i.e. a linear inequality satisfied by all feasible solutions $\boldsymbol{p}$ of the global problem (1.1), but violated from the current resource allocation (Theorem 3.3). Adding this cut to the *master problem* (1.3) by updating $\mathcal{Q}$ to a specific subset, we can recompute a new resource allocation and repeat this procedure until disaggregation is possible. At this stage, the use of a cryptographic protocol, secure multiparty computation, allows us to preserve the privacy of agents. Another main result stated in

this paper is the explicit upper bound on the convergence speed of APM in our framework (Theorem 3.2), which is obtained by spectral graph theory methods, exploiting also geometric properties of transportation polytopes. This explicit speed shows a linear impact of the number of agents, which is a strong argument for the applicability of the method in large distributed systems.

1.3. Related Work. A standard approach (e.g. [28, 33, 30]) to solve resource allocation problems in a distributed way is to rely on a a Lagrangian based decomposition technique: for instance dual subgradient methods [6, Ch. 6] or ADMM [13]. Such techniques are generally used to decompose a large problems into several subproblems of small dimension. However, those methods often require global convexity hypothesis, which are not satisfied in many practical problems (e.g. MILP). We refer the reader to [6, Chapter 6] for more background. On the contrary, our method can be used when the allocation problem (1.1) is not convex.

As developed in Section 4, the method proposed here can be related to Bender's decomposition [5]. The difference with Bender's approach is in the way of generating a new cut to add in the master problem: instead of solving linear programs, we use APM and our theoretical results, which provides a decentralized, privacy-preserving and scalable procedure. In contrast, at each stage, Benders' algorithm requires to solve a linear program requiring the knowledge of the private constraints of each individual agent (see Subsection 4.1 for more details).

The problem of the aggregation of constraints has been studied in the field of energy, in the framework of smart grids [25, 2]. In [25], the authors study the management of energy flexibilities and propose to approximate individual constraints by zonotopic sets to obtain an aggregate feasible set. A centralized aggregated problem is solved via a subgradient method, and a disaggregation procedure of a solution computes individual profiles. In [2], the authors propose to solve the economic power dispatch of a microgrid, subject to several agents private constraints, by using a Dantzig-Wolfe decomposition method.

The APM has been the subject of several works in itself [14, 3, 4]. The authors of [9] provide general results on the convergence rate of APM for semi-algebraic sets. They show that the convergence is geometric for polyhedra. However, it is generally hard to compute explicitly the geometric convergence rate of APM, as this requires to bound the singular values of certain matrices arising from the polyhedral constraints. A remarkable example where an explicit convergence rate for APM has been established is in [27], where the authors consider a different class of polyhedra arising in submodular optimization. A common point with our results is the use of spectral graph theory arguments to estimate singular values.

1.4. Structure. Section 2 describes the class of resource allocation problems we address in this paper, and formulate the idea of the decomposition with the *disaggregation* subproblems. In Section 3, we focus on APM, the subroutine used to solve the disaggregation subproblems. After stating results on the convergence of APM, In Subsection 3.1, we show the key result on which relies the proposed decomposition: how to generate a new cut to add in the master problem, from the output of APM. In Subsection 3.2, we show how to improve the privacy of the proposed procedure by using secure multiparty computation techniques. In Subsection 3.3 , we prove an explicit upper bound on the rate of convergence of APM in our case. In Section 4, we generalize part of our results and propose a modified algorithm in the case where agents constraints are polyhedral. Finally, in Section 5, we propose numerical examples of the method: Subsection 5.1 gives an illustrative toy example in dimension $T = 4$, while in Subsection 5.2, we consider a larger scale, nonconvex example, coming from the microgrid application exposed at the beginning of the introduction.

Notation. In the remaining of the paper, bold font $\boldsymbol{x}$ is used to denote a vector, while x refers to a scalar quantity. $\boldsymbol{v}^\top$ denotes the transpose of vector $\boldsymbol{v}$. Calligraphic letters such as $\mathcal{T}, \mathcal{N}, \mathcal{X}$ are used to denote sets, and if $\mathcal{T}_0 \subset \mathcal{T}$, the set $\mathcal{T}_0^c = \{t \in \mathcal{T} \setminus \mathcal{T}_0\}$ denotes the complementary set of $\mathcal{T}_0$. The notation $\mathcal{U}([a,b])$ stands for the uniform distribution on $[a,b]$. The notation $P_{\mathcal{C}}(.)$ refers to the Euclidean projection onto the convex set $\mathcal{C}$. For $d \in \mathbb{N}, \mathbb{1}_d$ denotes the vector of ones $(1 \dots 1)^\top \in \mathbb{R}^d$.

2. Resource Allocation and Transportation Structure.

2.1. A Decomposition based on Disaggregation. As stated in the introduction, we consider a centralized entity (e.g. an energy operator) interested in minimizing a possibly nonconvex cost function $\boldsymbol{p} \mapsto f(\boldsymbol{p})$, where $\boldsymbol{p} \in \mathbb{R}^T$ is the aggregate allocation of T dimensional resources (for example power production over T time periods). This resource allocation $\boldsymbol{p}$ is to be shared between a set $\mathcal{N}$ of N individual agents, each agent obtaining a part $\boldsymbol{x}_n \in \mathcal{X}_n$, where $\mathcal{X}_n$ denotes the individual feasibility set of agent n.

The global problem the operator wants to solve is described in (1.1). The idea behind the results of this paper is that, in problem (1.1), the constraints set $\mathcal{X}_n$ and individual profile $\boldsymbol{x}_n$ are confidential to agent n and should not be disclosed to the central operator or to another agent.

Let us define the set $\mathcal{P}_{\mathrm{D}}$ of feasible aggregate allocations that are disaggregable as:

$$\mathcal{P}_{\mathrm{D}} \stackrel{\text{def}}{=} \left\{ \boldsymbol{p} \in \mathcal{P} \mid \exists \boldsymbol{x} \in \mathcal{X} \ ; \ \boldsymbol{p} = \textstyle\sum_n \boldsymbol{x}_n \right\} . \tag{2.1}$$

Feasibility of problem (1.1) is equivalent to having $\mathcal{P}_{\mathrm{D}}$ not empty.

Constraints for each agent are composed of a global demand over the resources and lower and upper bounds over each resource, as given below:

Assumption 1. *For each $n \in \mathcal{N}$, there exists $E_n > 0$, $\overline{x}_n \in \mathbb{R}^T$, $\underline{x}_n \in \mathbb{R}^T$ such that :*

$$\mathcal{X}_n = \{\boldsymbol{x} \in \mathbb{R}^T : \textstyle\sum_{t \in \mathcal{T}} x_{n,t} = E_n \ \text{and} \ \underline{x}_{n,t} \leqslant x_{n,t} \leqslant \overline{x}_{n,t}\} \ \neq \emptyset. \tag{2.2}$$

In particular, $\mathcal{X}_n$ is convex and compact. Given an allocation $\boldsymbol{p}$, the structure obtained on the matrix $(\boldsymbol{x}_{n,t})_{n,t}$, where sums of coefficients along columns and along rows are fixed, is often referred to as transportation problem which has many applications (see e.g. [1, 26]). We focus on this case in Sections 2 and 3, while in Section 4, we shall give a generalization of some of our results in the general case where $\mathcal{X}_n$ is a polyhedron.

Given a particular allocation $\boldsymbol{p} \in \mathcal{P}$, the operator will be interested to know if this allocation is *disaggregable*, that is, if there exists individual profiles $(\boldsymbol{x}_n)_{n \in \mathcal{N}} \in \prod_n \mathcal{X}_n$ summing to $\boldsymbol{p}$, or equivalently if the *disaggregation problem* (1.2) has a solution.

Following (1.2), the *disaggregate* profile refers to $\boldsymbol{x}$, while the *aggregate* profile refers to the allocation $\boldsymbol{p}$. Problem (1.2) may not always be feasible. Some necessary conditions for a disaggregation to exist, obtained by summing the individual constraints on $\mathcal{N}$, are the following *aggregate* constraints:

$$\boldsymbol{p}^\top \mathbb{1}_T = \boldsymbol{E}^\top \mathbb{1}_N \tag{2.3a}$$

$$\text{and} \ \ \underline{\boldsymbol{x}}^\top \mathbb{1}_N \leqslant \boldsymbol{p} \leqslant \overline{\boldsymbol{x}}^\top \mathbb{1}_N \ . \tag{2.3b}$$

Those conditions are not sufficient in general, as explained in the the following section.

2.2. An equivalent flow problem and Hoffman conditions. The particular structure of the problem we consider implies that we can write it as a flow problem in a graph, as stated in Proposition 2.2. We refer the reader to the book [11, Chapter 3] for the terminology.

Definition 2.1. *Consider a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with vertices $\mathcal{V}$ and edges $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$, and demands $d : \mathcal{V} \to \mathbb{R}$ (where $d_v < 0$ means that v is a production node), edge lower capacities $\ell : \mathcal{E} \to \mathbb{R}_+$ and upper capacities $u : \mathcal{E} \to \mathbb{R}_+$. A flow on $\mathcal{G}$ is a function $\boldsymbol{x} : \mathcal{E} \to \mathbb{R}_+$ such that $\boldsymbol{x}$ satisfies the capacity constraints, that is $\forall e \in \mathcal{E}, \ell_e \leqslant x_e \leqslant u_e$, and Kirchoff's law, that is, $\forall v \in \mathcal{V}, \sum_{e \in \delta_v^+} x_e = d_v + \sum_{e \in \delta_v^-} x_e$, where δ_v^+ (resp. δ_v^-) is the set of edges ending at (resp. departing from) vertex v.*

The following proposition is immediate:

Proposition 2.2. *Consider the bipartite graph $\mathcal{G}$ with vertices $\mathcal{V} = \mathcal{T} \cup \mathcal{N}$ and edges $\mathcal{E} = \{(t,n)\}_{t \in \mathcal{T}, n \in \mathcal{N}}$. Define demands on nodes $\mathcal{T}$ by $d_t = -p_t$ and demands on nodes $\mathcal{N}$ by $d_n = E_n$. Assign to each edge (t,n) an upper capacity $u_{n,t} = \overline{x}_{n,t}$ and lower capacity $\ell_{n,t} = \underline{x}_{n,t}$. Then, finding a solution $\boldsymbol{x}$ to (1.2) is equivalent to finding a feasible flow in $\mathcal{G}$.*

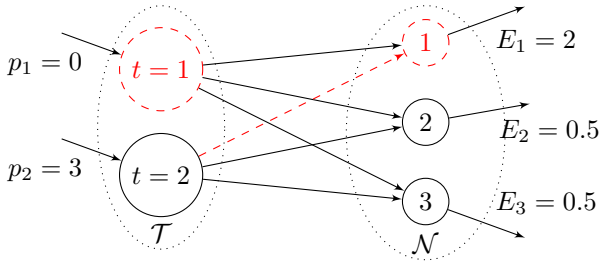


Figure 2.1: Example of a flow representation of the disaggregation problem ($T = 2$ and $N = 3$, $\underline{\boldsymbol{x}} = 0, \overline{\boldsymbol{x}} = 1$). *Here, the aggregate constraints* (2.3) *are verified, but condition* (2.4) *written with* $\mathcal{A} = \{t_1, n_1\}$ *(dashed nodes) does not hold.*

Hoffman [16] gave a necessary and sufficient condition for the flow problem to be feasible. This generalizes a result of Gale (1957). The stated condition is intuitive: there cannot be a subset of nodes whose demand exceeds its "import capacity".

THEOREM 2.3 ([16]). *Given a digraph* $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ *with demand* $d \in \mathbb{R}^{\mathcal{V}}$ *such that* $d(\mathcal{V}) = 0$ *and capacities* $\ell \in (\mathbb{R} \cup \{-\infty\})^{\mathcal{E}}$ *and* $u \in (\mathbb{R} \cup \infty)^{\mathcal{E}}$ *with* $\ell \leqslant u$*, there exists a feasible flow* $\boldsymbol{x} \in \mathcal{E} \to \mathbb{R}_+$ *on* $\mathcal{G}$ *if and only if:*

$$\forall \mathcal{A} \subset \mathcal{V}, \quad \sum_{e \in \delta^+(\mathcal{A})} u_e \geqslant \sum_{v \in \mathcal{A}} d_v + \sum_{e \in \delta^+(\mathcal{A}^c)} \ell_e \ , \tag{2.4}$$

where $\delta_+(\mathcal{A}) \stackrel{\text{def}}{=} \{(u, v) \in \mathcal{E} | u \in \mathcal{A}^c, v \in \mathcal{A}\}$ *is the set of edges coming to set* $\mathcal{A}$ *and* $\mathcal{A}^c \stackrel{\text{def}}{=} \mathcal{V} \setminus \mathcal{A}$.

The following Proposition 2.4 translates Theorem 2.3 in our framework:

PROPOSITION 2.4. *Disaggregation is possible iff:*

$$\forall \mathcal{T}_0 \subset \mathcal{T}, \forall \mathcal{N}_0 \subset \mathcal{N}, \quad \sum_{t \in \mathcal{T}_0} p_t - \sum_{n \in \mathcal{N}_0} E_n + \sum_{t \notin \mathcal{T}_0, n \in \mathcal{N}_0} \underline{x}_{n,t} \leqslant \sum_{t \in \mathcal{T}_0, n \notin \mathcal{N}_0} \overline{x}_{n,t} \, . \tag{2.5}$$

Proof. We apply (2.4) with $\mathcal{A} \stackrel{\text{def}}{=} \mathcal{T}_0^c \cup \mathcal{N}_0^c$ and use the equality $d(\mathcal{V}) = 0 = \sum_{v \in} d_v + \sum_{v \in A^c} d_v$.□

From Theorem 2.3 or Proposition 2.4 above, one can see that the aggregate constraints (2.3) are in general not sufficient to ensure that the disaggregation problem has a solution.

For a given set $\mathcal{T}_0$, there is a choice of $\mathcal{N}_0$ which leads to the strongest inequality (2.5), namely:

$$\sum_{t \in \mathcal{T}_0} p_t \leqslant \min_{\mathcal{N}_0 \subset \mathcal{N}} \left\{ \sum_{n \in \mathcal{N}_0} E_n - \sum_{t \notin \mathcal{T}_0, n \in \mathcal{N}_0} \underline{x}_{n,t} + \sum_{t \in \mathcal{T}_0, n \notin \mathcal{N}_0} \overline{x}_{n,t} \right\}, \tag{2.6}$$

In this way, we get $2^T - 2$ inequalities corresponding to the proper subsets $\mathcal{T}_0 \subset \mathcal{T}$. Moreover, in general, these $2^T - 2$ inequalities are not redundant. Although this is not stated in [16], this is a classical result whose proof is elementary.

3. Disaggregation based on APM.

3.1. Generation of Hoffman's constraints with APM. In this section, we propose an algorithm that solves (1.1) while preserving the privacy of each agent constraints $\mathcal{X}_n$ and individual profile $\boldsymbol{x}_n \in \mathbb{R}^T$. To do this, the proposed algorithm is implemented in a decentralized fashion and relies on the method of alternate projections method (APM) to solve the disaggregation problem (1.2).

Let us consider the polyhedron enforcing the agents constraints:

$$\mathcal{X} \stackrel{\text{def}}{=} \mathcal{X}_1 \times \cdots \times \mathcal{X}_N \ \text{ where } \mathcal{X}_n \stackrel{\text{def}}{=} \left\{ \boldsymbol{x}_n \in \mathbb{R}_+^T \mid \textstyle\sum_{t \in \mathcal{T}} x_{n,t} = E_n \text{ and } \forall t, \ \ \underline{x}_{n,t} \leqslant x_{n,t} \leqslant \overline{x}_{n,t} \right\} \ .$$

Besides, given an allocation $\boldsymbol{p} \in \mathcal{P}$, we consider the set of profiles aggregating to $\boldsymbol{p}$:

$$\mathcal{Y}_{\boldsymbol{p}} \stackrel{\text{def}}{=} \left\{ \boldsymbol{x} \in \mathbb{R}^{NT} \mid \forall t \in \mathcal{T}, \sum_{n \in \mathcal{N}} x_{n,t} = p_t \right\} .$$

Note that $\mathcal{Y}_{\boldsymbol{p}}$ is an affine subspace of $\mathbb{R}^{NT}$ (to be distinguished from $\mathcal{P}$ which is a subset of $\mathbb{R}^T$), and that $\mathcal{Y}_{\boldsymbol{p}} \cap \mathcal{X}$ is empty iff $\boldsymbol{p} \notin \mathcal{P}_{\mathrm{D}}$, according to the definition of $\mathcal{P}_{\mathrm{D}}$ in (2.1). The idea of the proposed algorithm is to build a finite sequence of decreasing subsets $(\mathcal{P}^{(s)})_{0 \leqslant s \leqslant S}$ such that:

$$\mathcal{P} = \mathcal{P}^{(0)} \supset \mathcal{P}^{(1)} \supset \cdots \supset \mathcal{P}^{(S)} \supset \mathcal{P}_{\mathrm{D}} .$$

At each iteration, a new aggregate resource allocation $\boldsymbol{p}^{(s)}$ is obtained by solving an instance of the master problem introduced in (1.3) with $\mathcal{Q} = \mathcal{P}^{(s)}$:

$$\min_{\boldsymbol{p} \in \mathbb{R}^T} f(\boldsymbol{p}) \tag{3.1a}$$

$$\text{s.t. } \boldsymbol{p} \in \mathcal{P}^{(s)} . \tag{3.1b}$$

In the remaining of the paper, we will refer to (3.1) as an instance of *master problem*. Our procedure relies on the following immediate observation:

PROPOSITION 3.1. *If $\boldsymbol{p}^{(s)}$ is a solution of (3.1), and $\mathcal{Y}_{\boldsymbol{p}^{(s)}} \cap \mathcal{X} \neq \emptyset$ and $\boldsymbol{x} \in \mathcal{Y}_{\boldsymbol{p}^{(s)}} \cap \mathcal{X}$, then $(\boldsymbol{p}^{(s)}, \boldsymbol{x})$ is an optimal solution of the initial problem (1.1).*

Having in hands a solution $\boldsymbol{p}^{(s)}$, we can check if $\mathcal{Y}_{\boldsymbol{p}^{(s)}} \cap \mathcal{X} \neq \emptyset$ using APM on $\mathcal{X}$ and $\mathcal{Y}_{\boldsymbol{p}^{(s)}}$, as described in Algorithm 3.1 below (where $\mathcal{Y} = \mathcal{Y}_{\boldsymbol{p}}$).

Algorithm 3.1 Alternate Projections Method (APM)

Require: Start with $\boldsymbol{y}^{(0)}$, $k = 0$, $\varepsilon_{\mathrm{cvg}}$, a norm $\|.\|$ on $\mathbb{R}^{NT}$

1: **repeat**
2: $\quad \boldsymbol{x}^{(k+1)} \leftarrow P_{\mathcal{X}}(\boldsymbol{y}^{(k)})$
3: $\quad \boldsymbol{y}^{(k+1)} \leftarrow P_{\mathcal{Y}}(\boldsymbol{x}^{(k+1)})$
4: $\quad k \leftarrow k + 1$
5: **until** $\left\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\right\| < \varepsilon_{\mathrm{cvg}}$

The idea of using cyclic projections to compute a point in the intersection of two sets comes from von Neumann [31], where the idea was applied for affine subspaces. Convergence of APM is proved by Theorem 3.2:

THEOREM 3.2 ([14]). *Let $\mathcal{X}$ and $\mathcal{Y}$ be two closed convex sets with $\mathcal{X}$ bounded, and let $(\boldsymbol{x}^{(k)})_k$ and $(\boldsymbol{y}^{(k)})_k$ be the two infinite sequences generated by APM on $\mathcal{X}$ and $\mathcal{Y}$ (Algorithm 3.1) with $\varepsilon_{\mathrm{cvg}} = 0$. Then there exists $\boldsymbol{x}^\infty \in \mathcal{X}$ and $\boldsymbol{y}^\infty \in \mathcal{Y}$ such that:*

$$\boldsymbol{x}^{(k)} \underset{k \to \infty}{\longrightarrow} \boldsymbol{x}^\infty , \quad \boldsymbol{y}^{(k)} \underset{k \to \infty}{\longrightarrow} \boldsymbol{y}^\infty ; \tag{3.2a}$$

$$\|\boldsymbol{x}^\infty - \boldsymbol{y}^\infty\|_2 = \min_{\boldsymbol{x} \in \mathcal{X}, \boldsymbol{y} \in \mathcal{Y}} \|\boldsymbol{x} - \boldsymbol{y}\|_2 . \tag{3.2b}$$

In particular, if $\mathcal{X} \cap \mathcal{Y} \neq \emptyset$, then $(\boldsymbol{x}^{(k)})_k$ and $(\boldsymbol{y}^{(k)})_k$ converge to a same point $\boldsymbol{x}^\infty \in \mathcal{X} \cap \mathcal{Y}$.

The convergence theorem is illustrated in Figure 3.1 in the case where $\mathcal{X} \cap \mathcal{Y} = \emptyset$, that is, when the disaggregation problem (1.2) is not feasible. The idea of the algorithm proposed in this paper is, in the case where $\mathcal{Y}_{\boldsymbol{p}^{(s)}} \cap \mathcal{X} = \emptyset$, to use the resulting vectors $\boldsymbol{x}^\infty$ and $\boldsymbol{y}^\infty$ to construct a new subset $\mathcal{P}^{(s+1)}$ by adding a constraint of type (2.5) to $\mathcal{P}^{(s)}$: indeed, from Proposition 2.4, we know that, if $\mathcal{Y}_{\boldsymbol{p}^{(s)}} \cap \mathcal{X} = \emptyset$, there exists at least one inequality (2.5) violated.

The difficulty is to guess one of this violated inequality among the set of 2^T possible inequalities. It turns out that, using the output of APM, we can build such an inequality.

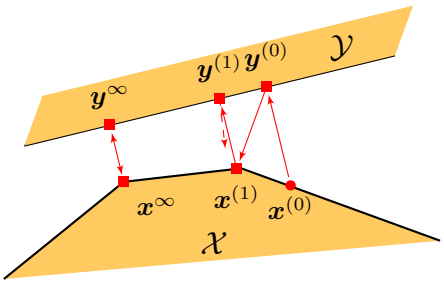


Figure 3.1: Alternate projections method (APM) on two sets $\mathcal{X}$ and $\mathcal{Y}$. *When $\mathcal{X} \cap \mathcal{Y} = \emptyset$, APM cycles over two points $\boldsymbol{x}^\infty$ and $\boldsymbol{y}^\infty$.*

Suppose that we obtain $\boldsymbol{x}^\infty \neq \boldsymbol{y}^\infty$ as defined in Theorem 3.2: we get a periodic cycle of the APM, that is, we have $\boldsymbol{x}^\infty = P_{\mathcal{X}}(\boldsymbol{y}^\infty)$ and $\boldsymbol{y}^\infty = P_{\mathcal{Y}}(\boldsymbol{x}^\infty)$, and the couple $(\boldsymbol{x}^\infty, \boldsymbol{y}^\infty)$ is the solution of the following optimization problem:

$$\min_{\boldsymbol{x},\boldsymbol{y}} \frac{1}{2} \|\boldsymbol{x} - \boldsymbol{y}\|_2^2 \tag{3.3a}$$

$$\forall n \in \mathcal{N},\ \sum_{t \in \mathcal{T}} x_{n,t} = E_n \qquad (\lambda_n) \tag{3.3b}$$

$$\forall n \in \mathcal{N}, \forall t \in \mathcal{T},\ \underline{x}_{n,t} \leqslant x_{n,t} \leqslant \overline{x}_{n,t} \qquad (\underline{\mu}_{n,t}, \overline{\mu}_{n,t}) \tag{3.3c}$$

$$\forall t \in \mathcal{T},\ \sum_{n \in \mathcal{N}} y_{n,t} = p_t \qquad (\nu_t)\ , \tag{3.3d}$$

where $\lambda_n \in \mathbb{R}$, $\underline{\mu}_{n,t}, \overline{\mu}_{n,t} \in \mathbb{R}_+$ and $\nu_t \in \mathbb{R}$ are the Lagrangian multipliers associated to the constraints (3.3b),(3.3c),(3.3d), with the associated Lagrangian function:

$$\mathcal{L}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \boldsymbol{\nu}) = \frac{1}{2} \|\boldsymbol{x} - \boldsymbol{y}\|_2^2 - \boldsymbol{\lambda}^\top \left(\textstyle\sum_t x_{n,t} - E_n\right)_n - \underline{\boldsymbol{\mu}}^\top (\boldsymbol{x} - \underline{\boldsymbol{x}}) - \overline{\boldsymbol{\mu}}^\top (\overline{\boldsymbol{x}} - \boldsymbol{x}) - \boldsymbol{\nu}^\top \left(\textstyle\sum_n \boldsymbol{y}_n - \boldsymbol{p}\right).$$

We notice that the stationarity condition of the Lagrangian with respect to the variable $y_{n,t}$ yields:

$$\forall n \in \mathcal{N}, \forall t \in \mathcal{T},\ \nu_t = x_{n,t} - y_{n,t}\ . \tag{3.4}$$

Let us consider the sets $\mathcal{T}_0 \subset \mathcal{T}$ and $\mathcal{N}_0 \subset \mathcal{N}$ defined from the output of APM on $\mathcal{X}$ and $\mathcal{Y}_{\boldsymbol{p}}$ as:

$$\mathcal{T}_0 \stackrel{\text{def}}{=} \left\{t \in \mathcal{T} \mid \exists n \in \mathcal{N},\ y_{n,t} > \overline{x}_{n,t}\right\} \text{ and } \mathcal{N}_0 \stackrel{\text{def}}{=} \left\{n \in \mathcal{N} \mid E_n - \textstyle\sum_{t \notin \mathcal{T}_0} \underline{x}_{n,t} - \sum_{t \in \mathcal{T}_0} \overline{x}_{n,t} < 0\right\}. \tag{3.5}$$

In Theorem 3.3 below, we show that applying the inequality (2.5) with the sets $\mathcal{T}_0$ and $\mathcal{N}_0$ defined in (3.5) defines a valid inequality for the disaggregation problem violated by the current allocation $\boldsymbol{p}$.

The intuition behind the definition of $\mathcal{T}_0$ and $\mathcal{N}_0$ in (3.5) is the following: $\mathcal{T}_0$ is the subset of resources for which there is an over supply (which overcomes the upper bound for at least one agent). Once $\mathcal{T}_0$ is defined, $\mathcal{N}_0$ is the associated subset of $\mathcal{N}$ minimizing the right hand side of (2.6). Indeed, (2.6) can be rewritten as:

$$\sum_{t \in \mathcal{T}_0} p_t \leqslant \min_{\mathcal{N}_0 \subset \mathcal{N}} \left\{ \sum_{n \in \mathcal{N}_0} \left(E_n - \textstyle\sum_{t \notin \mathcal{T}_0} \underline{x}_{n,t} - \sum_{t \in \mathcal{T}_0} \overline{x}_{n,t}\right) \right\} + \sum_{t \in \mathcal{T}_0, n \in \mathcal{N}} \overline{x}_{n,t}.$$

The following Theorem 3.3 is the key result on which relies the algorithm proposed in this paper.

THEOREM 3.3. *Consider the sequence of iterates $(\boldsymbol{x}^{(k)}, \boldsymbol{y}^{(k)})_{k \in \mathbb{N}}$ generated by the APM on $\mathcal{X}$ and $\mathcal{Y}_{\boldsymbol{p}}$ (see Algorithm 3.1). Then one of the following holds:*

(i) if $\mathcal{X} \cap \mathcal{Y}_{\boldsymbol{p}} \neq \emptyset$, then $\boldsymbol{x}^{(k)}, \boldsymbol{y}^{(k)} \underset{k \to \infty}{\longrightarrow} \boldsymbol{x}^\infty \in \mathcal{X} \cap \mathcal{Y}_{\boldsymbol{p}}$;

(ii) else, if $\mathcal{X} \cap \mathcal{Y}_{\boldsymbol{p}} = \emptyset$*, then* $\boldsymbol{x}^{(k)} \underset{k\to\infty}{\longrightarrow} \boldsymbol{x}^\infty \in \mathcal{X}$ *and* $\boldsymbol{y}^{(k)} \underset{k\to\infty}{\longrightarrow} \boldsymbol{y}^\infty \in \mathcal{Y}_{\boldsymbol{p}}$*. Then, considering the sets* $\mathcal{T}_0$ *and* $\mathcal{N}_0$ *as defined in* (3.5) *gives an inequality of Hoffman* (2.5) *violated by* $\boldsymbol{p}$*, that is:*

$$\sum_{n\in\mathcal{N}_0} E_n - \sum_{t\in\mathcal{T}_0} p_t + \sum_{t\in\mathcal{T}_0, n\notin\mathcal{N}_0} \overline{x}_{n,t} - \sum_{t\notin\mathcal{T}_0, n\in\mathcal{N}_0} \underline{x}_{n,t} < 0 \ . \tag{3.6}$$

Moreover, this Hoffman inequality can be written as a function of $\boldsymbol{x}^\infty$ *as:*

$$A_{\mathcal{T}_0}(\boldsymbol{x}^\infty) < \sum_{t\in\mathcal{T}_0} p_t \ \text{ with } A_{\mathcal{T}_0}(\boldsymbol{x}^\infty) \overset{\text{def}}{=} \sum_{t\in\mathcal{T}_0} \sum_{n\in\mathcal{N}} x^\infty_{n,t} \ . \tag{3.7}$$

Before giving the proof of Theorem 3.3, we need to show some technical properties on the sets $\mathcal{T}_0, \mathcal{N}_0$. For simplicity of notations, we use $\boldsymbol{x}$ and $\boldsymbol{y}$ to denote $\boldsymbol{x}^\infty$ and $\boldsymbol{y}^\infty$ in Proposition 3.4 and the proof of Theorem 3.3.

PROPOSITION 3.4. *With* $\boldsymbol{x} \neq \boldsymbol{y}$ *solutions of problem* (3.3) *(outputs of the APM on* $\mathcal{X}$ *and* $\mathcal{Y}_{\boldsymbol{p}}$ *with* $\varepsilon_{\text{cvg}} = 0$*),*

(i) $\forall t \in \mathcal{T}_0, \forall n \notin \mathcal{N}_0,\ y_{n,t} \geqslant \overline{x}_{n,t}$ *and* $x_{n,t} = \overline{x}_{n,t}$ *;*
(ii) $\mathcal{T}_0 = \{t \mid \nu_t > 0\} = \{t \mid p_t > \sum_n x_{n,t}\}$*, where* ν_t *is the optimal Lagrangian multiplier associated to* (3.3d)*;*
(iii) $\forall n \in \mathcal{N}_0, \lambda_n < 0$ *;*
(iv) $\forall t \notin \mathcal{T}_0, \forall n \in \mathcal{N}_0,\ x_{n,t} = \underline{x}_{n,t}$ *;*
(v) the sets $\mathcal{T}_0, \mathcal{T}_0^c, \mathcal{N}_0$ *and* $\mathcal{N}_0^c$ *are nonempty.*

The proof of Proposition 3.4 is technical and given in Appendix A. With Proposition 3.4, we are now ready to prove Theorem 3.3.

Proof of Theorem 3.3. We have:

$$\begin{aligned}
&\sum_{n\in\mathcal{N}_0} E_n + \sum_{t\in\mathcal{T}_0, n\notin\mathcal{N}_0} \overline{x}_{n,t} - \sum_{t\notin\mathcal{T}_0, n\in\mathcal{N}_0} \underline{x}_{n,t} - \sum_{t\in\mathcal{T}_0} p_t \\
&= \sum_{n\in\mathcal{N}_0} \sum_t x_{n,t} + \sum_{t\in\mathcal{T}_0, n\notin\mathcal{N}_0} \overline{x}_{n,t} - \sum_{t\notin\mathcal{T}_0, n\in\mathcal{N}_0} \underline{x}_{n,t} - \sum_{t\in\mathcal{T}_0} p_t \qquad \text{(from (3.3b))} \\
&= \sum_{n\in\mathcal{N}_0} \left(\sum_{t\notin\mathcal{T}_0} \underline{x}_{n,t} + \sum_{t\in\mathcal{T}_0} x_{n,t} \right) + \sum_{t\in\mathcal{T}_0, n\notin\mathcal{N}_0} x_{n,t} - \sum_{t\notin\mathcal{T}_0, n\in\mathcal{N}_0} \underline{x}_{n,t} - \sum_{t\in\mathcal{T}_0} p_t \qquad \text{(from Prop.3.4 (i) and (iv))} \\
&= \sum_{t\in\mathcal{T}_0} \left(\sum_{n\in\mathcal{N}_0} x_{n,t} + \sum_{n\notin\mathcal{N}_0} x_{n,t} \right) - \sum_{t\in\mathcal{T}_0} p_t = \sum_{t\in\mathcal{T}_0} \left(\sum_{n\in\mathcal{N}} x_{n,t} - \sum_{n\in\mathcal{N}} y_{n,t} \right) \\
&= \sum_{t\in\mathcal{T}_0} \left(\sum_{n\in\mathcal{N}} -\nu_t \right) = \sum_{t\in\mathcal{T}_0} \left(- \sum_{n\in\mathcal{N}} |x_{n,t} - y_{n,t}| \right)
\end{aligned}$$

using the stationarity conditions (3.4) and for all $t \in \mathcal{T}_0, \nu_t > 0$ by Prop.3.4 (ii). Moreover, using:

$$\sum_{t\in\mathcal{T}} \sum_{n\in\mathcal{N}} (x_{n,t} - y_{n,t}) = \sum_{n\in\mathcal{N}} E_n - \sum_{t\in\mathcal{T}} p_t = 0 \ , \tag{3.8}$$

we see that:

$$\sum_{t\in\mathcal{T}_0} \Big(- \sum_{n\in\mathcal{N}} |x_{n,t} - y_{n,t}| \Big) = -(\|\boldsymbol{x} - \boldsymbol{y}\|_1)/2 < 0 \ , \tag{3.9}$$

which shows (3.6). We now show that inequality (3.7) is obtained by a rewriting of (3.6), indeed:

$$\begin{aligned}
&\sum_{n\in\mathcal{N}_0} E_n + \sum_{t\in\mathcal{T}_0, n\notin\mathcal{N}_0} \overline{x}_{n,t} - \sum_{t\notin\mathcal{T}_0, n\in\mathcal{N}_0} \underline{x}_{n,t} \\
&= \sum_{n\in\mathcal{N}_0}\sum_{t\in\mathcal{T}} x_{n,t} + \sum_{t\in\mathcal{T}_0, n\notin\mathcal{N}_0} x_{n,t} - \sum_{t\notin\mathcal{T}_0, n\in\mathcal{N}_0} x_{n,t} \qquad \text{(from Prop.3.4 (i) and (iv))} \\
&= \sum_{t\in\mathcal{T}_0, n\in\mathcal{N}_0} x_{n,t} + \sum_{t\in\mathcal{T}_0, n\notin\mathcal{N}_0} x_{n,t} = \sum_{t\in\mathcal{T}_0}\sum_{n\in\mathcal{N}} x_{n,t} \;= A_{\mathcal{T}_0}(\boldsymbol{x}).
\end{aligned}$$

∎

Suppose, as before, that the two sequences generated by the APM on $\mathcal{X}$ and $\mathcal{Y}$ converge to two distinct points $\boldsymbol{x}^\infty$ and $\boldsymbol{y}^\infty$. Then, at each round k, we can define from (A.1) and considering any $n \in \mathcal{N}$, the multiplier $\boldsymbol{\nu}^{(k)} = \boldsymbol{y}_n^{(k)} - \boldsymbol{x}_n^{(k)}$ tends to $\boldsymbol{y}_n^\infty - \boldsymbol{x}_n^\infty \stackrel{\text{def}}{=} \boldsymbol{\nu}^\infty$. The set $\mathcal{T}_0$ of Theorem 3.3 is:

$$\mathcal{T}_0^\infty \stackrel{\text{def}}{=} \{t \in \mathcal{T} \mid 0 < \nu_t^\infty\}, \tag{3.10}$$

which raises an issue for practical computation, as $\boldsymbol{\nu}^\infty$ is only obtained *ultimately* by APM, possibly in infinite time. To have access to $\mathcal{T}_0^\infty$ *in finite time*, that is, from one of the iterates $(\boldsymbol{\nu}^{(k)})_k$, we consider the set:

$$\mathcal{T}_0^{(K)} \stackrel{\text{def}}{=} \{t \in \mathcal{T} \mid B\varepsilon_{\text{cvg}} < \nu_t^{(K)}\}, \tag{3.11}$$

where ε_{cvg} is the tolerance for convergence of APM as defined in Algorithm 3.1, $B > 0$ is a constant, and K (depending on ε_{cvg}) is the first integer such that $\left\|\boldsymbol{x}^{(K)} - \boldsymbol{x}^{(K-1)}\right\| < \varepsilon_{\text{cvg}}$.

We next show that we can choose B to ensure that $\mathcal{T}_0^{(K)} = \mathcal{T}_0^\infty$ for ε_{cvg} small enough. We rely on the geometric convergence rate of APM on polyhedra [9, 27]:

Proposition 3.5. *[27] If $\mathcal{X}$ and $\mathcal{Y}$ are polyhedra, there exists $\rho \in (0,1)$ such that the sequence $(\boldsymbol{x}^{(k)})_k$ and $(\boldsymbol{y}^{(k)})_k$ generated by APM verify for all $k \geqslant 1$:*

$$\left\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}\right\|_2 \leqslant \rho \left\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\right\|_2 \quad \text{and} \quad \left\|\boldsymbol{y}^{(k+1)} - \boldsymbol{y}^{(k)}\right\|_2 \leqslant \rho \left\|\boldsymbol{y}^{(k)} - \boldsymbol{y}^{(k-1)}\right\|_2 .$$

Proposition 3.5 applies to any polyhedra $\mathcal{X}$ and $\mathcal{Y}$. In Subsection 3.3 we shall give an explicit upper bound on the constant ρ in the specific transportation case given by (1.1c) and (2.2).

From the previous proposition, we can quantify the distance to the limits in terms of ρ:

Lemma 3.6. *Consider an integer K such that the sequence $(\boldsymbol{x}^{(k)})_{k\geqslant 0}$ generated by APM satisfies $\left\|\boldsymbol{x}^{(K)} - \boldsymbol{x}^{(K-1)}\right\| \leqslant \varepsilon_{\text{cvg}}$, then we have for any $K' \geqslant K-1$:*

$$\left\|\boldsymbol{x}^\infty - \boldsymbol{x}^{(K')}\right\| \leqslant \tfrac{\varepsilon_{\text{cvg}}}{1-\rho} .$$

Proof. From Proposition 3.5, we have for any $k \geqslant K$:

$$\left\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(K')}\right\| \leqslant \sum_{s=0}^{k-K'} \left\|\boldsymbol{x}^{(K+s+1)} - \boldsymbol{x}^{(K+s)}\right\| \leqslant \sum_{s=0}^{k-K'} \rho^s \left\|\boldsymbol{x}^{(K'+1)} - \boldsymbol{x}^{(K')}\right\| \leqslant \frac{1}{1-\rho}\varepsilon_{\text{cvg}} ,$$

so that, by taking the limit $k \to \infty$, one obtains $\left\|\boldsymbol{x}^\infty - \boldsymbol{x}^{(K')}\right\| \leqslant \frac{\varepsilon_{\text{cvg}}}{1-\rho}$. ∎

With this previous lemma, we can state the condition on B ensuring the desired property:

Proposition 3.7. *Define $\underline{\nu} \stackrel{\text{def}}{=} \min\{|\nu_t^\infty| > 0\}$ (least nonzero element of $\boldsymbol{\nu}^\infty$). If the constants B and $\varepsilon_{\text{cvg}} > 0$ are chosen such that $B > \frac{1}{1-\rho}$ and $\varepsilon_{\text{cvg}} \times 2B < \underline{\nu}$, and Algorithm 3.1 stops at iteration K, then we have:*

$$\mathcal{T}_0^{(K)} = \mathcal{T}_0^\infty .$$

Proof. Let $t \in \mathcal{T}_0^\infty$, that is $\nu_t^\infty > 0$ which is equivalent to $\nu_t^\infty \geqslant \underline{\nu}$ by definition of $\underline{\nu}$. We have:

$$\begin{aligned}\nu_t^{(K)} = \frac{1}{N}(p_t - \sum_n x_{n,t}^{(K)}) = \frac{1}{N}(p_t - \sum_n x_{n,t}^{\infty}) + \frac{1}{N}(\sum_n x_{n,t}^{\infty} - \sum_n x_{n,t}^{(K)}) \\ > \nu_t^\infty - \tfrac{\varepsilon_{\text{cvg}}}{1-\rho} \geqslant \underline{\nu} - \tfrac{\varepsilon_{\text{cvg}}}{1-\rho} > \varepsilon_{\text{cvg}}(2B - \tfrac{1}{1-\rho}) \ ,\end{aligned}$$

and this last quantity is greater than $B\varepsilon_{\text{cvg}}$ as soon as $B \geqslant \frac{1}{1-\rho}$, thus $t \in \mathcal{T}_0^{(K)}$.

Conversely, if $t \in \mathcal{T}_0^{(K)}$, then:

$$\begin{aligned}\nu_t^\infty = \frac{1}{N}(p_t - \sum_n x_{n,t}^{\infty}) = \frac{1}{N}(p_t - \sum_n x_{n,t}^{(K)}) - \frac{1}{N}(\sum_n x_{n,t}^{\infty} - \sum_n x_{n,t}^{(K)}) \\ \geqslant \nu_t^{(K)} - \tfrac{B}{1-\rho} \ > \ \nu_t^{(K)} - B\varepsilon_{\text{cvg}} \geqslant (B-B)\varepsilon_{\text{cvg}} \geqslant 0 \ ,\end{aligned}$$

so that $t \in \mathcal{T}_0^\infty$. Furthermore, the "approximated" cut $\sum_{t\in\mathcal{T}_0}\left(\sum_{n\in\mathcal{N}} x_{n,t}^{(K)} - p_t\right) \geqslant 0$ is violated by the current value of $\boldsymbol{p}$ (or $\boldsymbol{p}^{(s)}$ at iteration s) in the algorithm as:

$$\begin{aligned}\sum_{t\in\mathcal{T}_0}\left(\sum_{n\in\mathcal{N}} x_{n,t}^{(K)} - p_t\right) &\leqslant \sum_{t\in\mathcal{T}_0}\left(\sum_{n\in\mathcal{N}} x_{n,t}^{(K)} - x_{n,t}^{\infty}\right) + \sum_{t\in\mathcal{T}_0}\left(\sum_{n\in\mathcal{N}} x_{n,t}^{\infty} - p_t\right) \\ &\leqslant \left\|\boldsymbol{x}^{(K)} - \boldsymbol{x}^\infty\right\|_1 - \frac{1}{2}\left\|\boldsymbol{x}^\infty - \boldsymbol{y}^\infty\right\|_1\end{aligned}$$

using (3.8) and (3.9). This last quantity is negative as soon as $\left\|\boldsymbol{x}^{(K)} - \boldsymbol{x}^\infty\right\|_1 < \frac{1}{2}\left\|\boldsymbol{x}^\infty - \boldsymbol{y}^\infty\right\|_1$, which holds in particular if $B\varepsilon_{\text{cvg}} < \frac{1}{2}\left\|\boldsymbol{x}^\infty - \boldsymbol{y}^\infty\right\|_1$. ∎

This second proposition shows a surprising result: even if we do not have access to the limit $\boldsymbol{x}^\infty$, we can compute *in finite time* the *exact* left hand side term $A_{\mathcal{T}_0}(\boldsymbol{x}^\infty)$ of the cut (3.7):

Proposition 3.8. *Under the hypotheses of Proposition 3.7, we have:*

$$A_{\mathcal{T}_0}(\boldsymbol{x}^{(K)}) = \sum_{t\in\mathcal{T}_0}\sum_{n\in\mathcal{N}} x_{n,t}^{(K)} = A_{\mathcal{T}_0}(\boldsymbol{x}^\infty) \ .$$

Proof. We start by showing some technical properties similar to Proposition 3.4:

Lemma 3.9. *The iterate $\boldsymbol{x}^{(K)}$ satisfies the following properties:*
(i) $\forall t \in \mathcal{T}_0, \forall n \notin \mathcal{N}_0, x_{n,t}^{(K)} = x_{n,t}^\infty = \overline{x}_{n,t}$;
(ii) $\forall t \notin \mathcal{T}_0, \forall n \in \mathcal{N}_0, \ x_{n,t}^{(K)} = x_{n,t}^\infty = \underline{x}_{n,t}$.

The proof of Lemma 3.9 is similar to Proposition 3.4 and is given in Appendix B. Then, having in mind that $\mathcal{T}_0^{(K)} = \mathcal{T}_0^\infty$ from Proposition 3.7, and $\mathcal{N}_0$ is obtained from $\mathcal{T}_0^\infty$ by (3.5), we obtain:

$$\begin{aligned}A_{\mathcal{T}_0}(\boldsymbol{x}^{(K)}) &= \sum_{n\in\mathcal{N}_0}\Big(\sum_{t\notin\mathcal{T}_0}\underline{x}_{n,t} + \sum_{t\in\mathcal{T}_0} x_{n,t}^{(K)}\Big) - \sum_{t\notin\mathcal{T}_0, n\in\mathcal{N}_0}\underline{x}_{n,t} + \sum_{t\in\mathcal{T}_0, n\notin\mathcal{N}_0} x_{n,t}^{(K)} \\ &\leqslant \sum_{n\in\mathcal{N}_0}\sum_{t\in\mathcal{T}} x_{n,t}^{(K)} - \sum_{t\notin\mathcal{T}_0, n\in\mathcal{N}_0}\underline{x}_{n,t} + \sum_{t\in\mathcal{T}_0, n\notin\mathcal{N}_0}\overline{x}_{n,t} \quad \text{(from Lemma 3.9)}\end{aligned}$$

which equals to $A_{\mathcal{T}_0}(\boldsymbol{x}^\infty)$ as we have $\sum_{t\in\mathcal{T}} x_{n,t}^{(K)} = E_n$ for each $n \in \mathcal{N}$. ∎

Before presenting our algorithm using this last result, we focus on the technique of multiparty secure computation (SMC) which will be used here to ensure the privacy of agent's constraints and profiles while running the APM.

3.2. Privacy-preserving Projections through SMC . APM, as described in Algorithm 3.1, enables a distributed implementation in our context, by the structure of the algorithm itself: the operator computes the projection on $\mathcal{Y}_{\boldsymbol{p}}$ while each agent n can compute, possibly in parallel, the projection on $\mathcal{X}_n$ of the new profile transmitted by the operator. This enables each agent (as well as the operator) to keep her individual constraint and not reveal it to the operator or other agents. However, each agent has to transmit back her newly computed individual profile to the operator for the next iteration.

Using a secure multi-party computation (SMC) protocol as introduced by [34], we can avoid this communication of individual profiles and perform APM without revealing the sequence of agent profiles $\boldsymbol{x}$ to the aggregator.

For this, we use the fact that $\mathcal{Y}_{\boldsymbol{p}}$ is an affine subspace and thus the projection on $\mathcal{Y}_{\boldsymbol{p}}$ can be obtained explicitly component-wise. Indeed, summing (3.4) on $\mathcal{T}$, we immediately obtain:

$$\forall n \in \mathcal{N},\ [\mathcal{P}_{\mathcal{Y}_{\boldsymbol{p}}}(\boldsymbol{x})]_n = \boldsymbol{x}_n + \tfrac{1}{N}(\boldsymbol{p} - \textstyle\sum_{m\in\mathcal{N}} \boldsymbol{x}_m)\ . \tag{3.12}$$

Thus, having access to the *aggregate* profile $\boldsymbol{S} \overset{\text{def}}{=} \sum_{n\in\mathcal{N}} \boldsymbol{x}_n$, each agent can compute locally the component of the projection on $\mathcal{Y}_{\boldsymbol{p}}$ of her profile, instead of transmitting the profile to the operator for computing the projection in a centralized way.

Using SMC, the sum $\boldsymbol{S}$ can be computed in a non-intrusive manner and by several communications between agents and the operator, as described in Algorithm 3.2. The main idea of SMC is that, instead of sending her profile $\boldsymbol{x}_n$, agent n *splits* $x_{n,t}$ for each t into N random parts $(s_{n,t,m})_m$, according to an uniform distribution and summing to $\boldsymbol{x}_{n,t}$ (Lines 2-3). Thus, each part $s_{n,t,m}$ taken individually does not reveal any information on $\boldsymbol{x}_n$ nor on $\mathcal{X}_n$, and can be sent to agent m. Once all exchanges of parts are completed (Line 5), and n has herself received the parts from other agents, agent n computes a new aggregate quantity $\boldsymbol{\sigma}_n$ (Line 7), which does not contain either any information about any of the agents, and sends it to the operator (Line 8). The operator can finally compute the quantity $\boldsymbol{S} = \boldsymbol{x}^\top \mathbb{1}_N = \boldsymbol{\sigma}^\top \mathbb{1}_N$.

Algorithm 3.2 SMC of Aggregate (SMCA) $\sum_{n\in\mathcal{N}} \boldsymbol{x}_n$

Require: A profile $\boldsymbol{x}_n$ for each agent $n \in \mathcal{N}$
1: **for** each agent $n \in \mathcal{N}$ **do**
2: Draw $\forall t, (s_{n,t,m})_{m=1}^{N-1} \in \mathcal{U}([0,A]^{N-1})$
3: and set $\forall t, s_{n,t,N} \overset{\text{def}}{=} x_{n,t} - \sum_{m=1}^{N-1} s_{n,t,m}$
4: Send $(s_{n,t,m})_{t\in\mathcal{T}}$ to agent $m \in \mathcal{N}$
5: **done**
6: **for** each agent $n \in \mathcal{N}$ **do**
7: Compute $\forall t, \sigma_{n,t} = \sum_{m\in\mathcal{N}} s_{m,t,n}$
8: Send $(\sigma_{n,t})_{t\in\mathcal{T}}$ to operator
9: **done**
10: Operator computes $\boldsymbol{S} = \sum_{n\in\mathcal{N}} \boldsymbol{\sigma}_n$ (and broadcasts it to agents)

Remark 1. *As $\boldsymbol{\sigma}_n$, and $\boldsymbol{s}_n$ are random by construction, an eavesdropper aiming to learn the profile $\boldsymbol{x}_n$ of n has no choice but to intercept all the communications of n to all other agents (to learn $(s_{n,t,m})_{m\neq n}$ and $(s_{m,t,n})_{m\neq n}$) and to the operator (to learn σ_n).*

We sum up in Algorithm 3.3 below the procedure of generating a new constraint as stated in Theorem 3.3 from the output of APM in finite time (see Proposition 3.7) and in a privacy-preserving way using SMC.

To choose B and ε_{cvg} satisfying the conditions of Proposition 3.7 *a priori*, one has to know the value of $\underline{\nu}$. Although a conservative lower bound could be obtained by Diophantine arguments if we consider rationals as inputs of the algorithm, in practice it is easier and more efficient to proceed in an iterative manner for the value of ε_{cvg}. Indeed, one can start with ε_{cvg} arbitrary large so that

Algorithm 3.3 Non-intrusive APM (NI-APM)

Require: Start with $\boldsymbol{y}^{(0)}$, $k=1$, $\varepsilon_{\text{cvg}}, \varepsilon_{\text{dis}}$, norm $\|.\|$ on $\mathbb{R}^{NT}$

1: **repeat**
2: **for** each agent $n \in \mathcal{N}$ **do**
3: $\boldsymbol{x}_n^{(k)} \leftarrow P_{\mathcal{X}_n}(\boldsymbol{y}_n^{(k-1)})$
4: **done**
5: Operator obtains $\boldsymbol{S}^{(k)} \leftarrow$SMCA$(\boldsymbol{x}^{(k)})$ (*cf* Algo. 3.2)
6: and sends $\boldsymbol{\nu}^{(k)} \stackrel{\text{def}}{=} \frac{1}{N}(\boldsymbol{p} - \boldsymbol{S}^{(k)}) \in \mathbb{R}^T$ to agents $\mathcal{N}$
7: **for** each agent $n \in \mathcal{N}$ **do**
8: Compute $\boldsymbol{y}_n^{(k)} \leftarrow \boldsymbol{x}_n^{(k)} + \boldsymbol{\nu}^{(k)}$ ▷ *from* (3.4) *and* (3.12), $\boldsymbol{y}^{(k)} = P_{\mathcal{Y}_{\boldsymbol{p}}}(\boldsymbol{x}^{(k)})$
9: **done**
10: $k \leftarrow k+1$
11: **until** $\left\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\right\| < \varepsilon_{\text{cvg}}$
12: **if** $\left\|\boldsymbol{x}^{(k)} - \boldsymbol{y}^{(k)}\right\| \leqslant \varepsilon_{\text{dis}}$ **then** ▷ *found a* ε_{dis}*-solution of the disaggregation problem*
13: Each agent adopts profile $\boldsymbol{x}_n^{(k)}$
14: **return** DISAG $\leftarrow$ TRUE
15: **else** ▷ *have to find a valid constraint violated by* $\boldsymbol{p}$
16: Operator computes $\mathcal{T}_0 \leftarrow \{t \in \mathcal{T} \mid B\varepsilon_{\text{cvg}} < \nu_t^{(k)}\}$
17: Operator computes $A_{\mathcal{T}_0} \leftarrow$ SMCA$(\ (\boldsymbol{x}_t^{(k)})_{t\in\mathcal{T}_0})$
18: **if** $A_{\mathcal{T}_0} - \sum_{t\in\mathcal{T}_0} \boldsymbol{p}_t < 0$ **then**
19: **return** DISAG $\leftarrow$ FALSE, $\mathcal{T}_0$, $A_{\mathcal{T}_0}$
20: **else** ▷ *need to run APM with higher precision*
21: Return to Line 1 with $\varepsilon_{\text{cvg}} \leftarrow \varepsilon_{\text{cvg}}/2$
22: **end**
23: **end**

APM will converge quickly, and then check if the cut obtained is violated by the current value of $\boldsymbol{p}$ (Subsection 3.2): if it is not the case, we can continue the iterations of APM with convergence precision improved to $\varepsilon_{\text{cvg}}/2$ (Subsection 3.2). Proposition 3.7 ensures that this loop terminates in finite time.

The parameter $\varepsilon_{\text{dis}} > 0$ (Line 12 of Algorithm 3.3) has to be chosen a priori by the operator, depending on the precision required. In general in APM, $\boldsymbol{x}^\infty = \boldsymbol{y}^\infty$ will only be achieved in infinite time, so choosing ε_{dis} strictly positive is required.

We end this section by summarizing in Algorithm 3.4 the global iterative procedure to compute an optimal and disaggregable resource allocation $\boldsymbol{p}$, solution of the initial problem (1.1), using iteratively NI-APM (Algorithm 3.3) and adding constraints as stated in Theorem 3.3.

Algorithm 3.4 iteratively calls NI-APM (Algorithm 3.3) and in case disaggregation is not possible (Line 11), a new constraint is added (Line 13), obtained from the quantity $A_{\mathcal{T}_0}$ defined in (3.7), to the feasible set of resource allocations $\mathcal{P}^{(s)}$ in problem (3.1). This constraint is an inequality on $\boldsymbol{p}$ and thus does not reveal significant individual information to the operator. The algorithm stops when disaggregation is possible (Line 9). The termination of Algorithm 3.4 is ensured by the following property and the form of the constraints added (3.6):

PROPOSITION 3.10. *Algorithm 3.4 stops after a finite number of iterations, as at most* $2^T - 2$ *constraints (Line 13) can be added to the master problem (Line 2).*

The following Proposition 3.11 shows the correctness of our Algorithm 3.4.

PROPOSITION 3.11. *Let* B *and* ε_{cvg} *satisfy the conditions of Proposition 3.7 and 3.8. Then:*

- *if the problem (1.1) has no solution, Algorithm 3.4 exits at Line 4 after at most* $2^T - 2$ *iterations;*
- *else, Algorithm 3.4 computes, after at most* $s \leqslant 2^T - 2$ *iterations, an aggregate solution* $\boldsymbol{p}^{(s)} \in \mathcal{P}$,

Algorithm 3.4 Non-intrusive Optimal Disaggregation

Require: $s = 0$, $\mathcal{P}^{(0)} = \mathcal{P}$; DISAG= FALSE

while Not DISAG **do**
  Solve $\min_{\boldsymbol{p}\in\mathcal{P}^{(s)}} f(\boldsymbol{p})$
  if problem infeasible **then**
    Exit
  else
    Compute $\boldsymbol{p}^{(s)} = \arg\min_{\boldsymbol{p}\in\mathcal{P}^{(s)}} f(\boldsymbol{p})$
  end
  DISAG← NI-APM($\boldsymbol{p}^{(s)}$) (Algo. 3.3)
  if DISAG **then**
    Operator adopts $\boldsymbol{p}^{(s)}$
  else
    Obtain $\mathcal{T}_0^{(s)}, A_{\mathcal{T}_0}^{(s)}$ from NI-APM($\boldsymbol{p}^{(s)}$)
    $\mathcal{P}^{(s+1)} \leftarrow \mathcal{P}^{(s)} \cap \{\boldsymbol{p} | \sum_{t\in\mathcal{T}_0^{(s)}} p_t \leqslant A_{\mathcal{T}_0}^{(s)}\}$
  end
  $s \leftarrow s + 1$
done

associated to individual profiles $(\boldsymbol{x}^*)_n$ = *NI-APM*($\boldsymbol{p}^{(s)}$) *such that:*

$$\boldsymbol{p}^{(s)} \in \mathcal{P}, \quad , \forall n \in \mathcal{N}, \boldsymbol{x}_n^* \in \mathcal{X}_n, \quad \left\| \sum_{n\in\mathcal{N}} \boldsymbol{x}_n^* - \boldsymbol{p}^{(s)} \right\| \leqslant \varepsilon_{\text{dis}} \quad \text{and } f(\boldsymbol{p}^{(s)}) \leqslant f^* \ ,$$

where f^* *is the optimal value of problem* (1.1).

Proof. The proof is immediate from Theorem 3.3, Proposition 3.7 and Proposition 3.8. ∎

REMARK 2. *The upper bound on the number of constraints added has no dependence on* N *because, as stated in* (2.6)*, once a subset of* $\mathcal{T}$ *is chosen, the constraint we add in the algorithm is found by taking the minimum by taking a minimum over the subsets of* $\mathcal{N}$*.*

Although there exist some instances with an exponential number of independent constraints, this does not jeopardize the proposed method: in practice, the algorithm stops after a very small number of constraints added. Intuitively, we will only add constraints "supporting" the optimal allocation $\boldsymbol{p}$. Thus, Algorithm 3.4 is a method which enables the operator to compute a resource allocation $\boldsymbol{p}$ and the N agents to adopt profiles $(\boldsymbol{x}_n)_n$, such that $(\boldsymbol{x}, \boldsymbol{p})$ solves the global problem (1.1), and the method ensures that both agent constraints (upper bounds $\overline{\boldsymbol{x}}_n$, lower bounds $\underline{\boldsymbol{x}}_n$, demand E_n); and disaggregate (individual) profile $\boldsymbol{x}_n$ (as well as the iterates $(\boldsymbol{x}^{(k)})_k$ and $(\boldsymbol{y}^{(k)})_k$ in NI-APM) are kept confidential by agent n and can not be induced by a third party (either the operator or any other agent $m \neq n$).

REMARK 3. *A natural approach to address problem* (1.1) *in a distributed way, assuming that both the cost function* $\boldsymbol{p} \mapsto f(\boldsymbol{p})$ *and the feasibility set* $\mathcal{P}$ *are convex, is to rely on Lagrangian based decomposition techniques. Examples of such methods are Dual subgradient methods [6, Chapter 6], auxiliary problem principle method [10], ADMM [13],[35] or bundle methods [23].*

One can think of a privacy-preserving implementation of those techniques, where Lagrangian multipliers associated to the (relaxed) aggregation constraint $\sum_n \boldsymbol{x}_n = \boldsymbol{p}$ *would be updated using the SMC technique as described in Algorithm* 3.2*. However, those techniques usually ask for strong convexity hypothesis: for instance, in ADMM, in order to keep the decomposition structure in agent by agent, a possibility is to use multi-blocs ADMM with* $N+1$ *blocs (*N *agents and the operator), which is known to converge in the condition that strong convexity of the cost function in at least* N *of the* $N+1$ *variables holds [12]. The complete study of privacy-preserving implementations of Lagrangian decomposition methods is left for further work.*

The advantage of Algorithm 3.4 proposed in this paper is that convergence is ensured (see Proposition 3.10) even if the cost function $\boldsymbol{p} \mapsto f(\boldsymbol{p})$ and the feasibility set $\mathcal{P}$ are not convex, which is the case in many practical situations (see Subsection 5.2).

REMARK 4. *Algorithm 3.4 solves problem (1.1) in a privacy-preserving manner. For this, we use both the results stated in Theorem 3.3 and SMC to* securely *transmit the aggregate profile to the operator at each step. For the latter point, other techniques could be used instead of SMC such as consensus-based aggregation algorithms [15]. A comparison of the different possible techniques, relying on quantitative privacy indicators, would be interesting and is an avenue for further work.*

In the next section, we focus on the convergence rate of APM in the particular case of transportation constraints, precising the geometric rate stated in Theorem 3.2.

3.3. Complexity Analysis of APM in the Transportation Case. In this section we analyze the speed of convergence of the alternate projections method (APM) described in Algorithm 3.1 on the sets $\mathcal{X}$ and $\mathcal{Y}_{\boldsymbol{p}}$ defined in Section 2.

A general result in [9] gives an upper bound of the sequences generated by APM on $\mathcal{X}$ and $\mathcal{Y}$ if these two sets are semi-algebraic. In particular, it establishes the geometric convergence for polyhedral sets. However, as stated in [27], given two particular polyhedral sets $\mathcal{X}$ and $\mathcal{Y}$, it is not straightforward to deduce an explicit rate of convergence from their result.

The authors in [27] established in a particular case a geometric convergence with an explicit upper bound on the convergence rate. They consider APM on two sets P and Q, where P is a linear subspace and Q is a product of base polytopes of submodular functions.

In this section, we also establish an explicit upper bound on the convergence rate of APM in the transportation case, that is with $\mathcal{X}$ and $\mathcal{Y}_{\boldsymbol{p}}$ defined in (2.2) and (1.2b):

THEOREM 3.12. *For the two sets $\mathcal{X}$ and $\mathcal{Y}_{\boldsymbol{p}}$, the sequence of alternate projections converges to $\boldsymbol{x}^* \in \mathcal{X}$, $\boldsymbol{y}^* \in \mathcal{X}^P$ satisfying $\|\boldsymbol{x}^* - \boldsymbol{y}^*\| = \inf_{\boldsymbol{x}\in\mathcal{X},\boldsymbol{y}\in\mathcal{Y}_{\boldsymbol{p}}} \|\boldsymbol{x} - \boldsymbol{y}\|$, at the geometrical rate:*

$$\left\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^*\right\| \leqslant 2\left\|\boldsymbol{x}^{(0)} - \boldsymbol{x}^*\right\| \times \left(1 - \tfrac{4}{N(T+1)^2(T-1)}\right)^k,$$

and the symmetric inequalities hold for $(\boldsymbol{y}^{(k)})_k$.

For the remaining of this section, we will just use $\mathcal{Y}$ to denote $\mathcal{Y}_{\boldsymbol{p}}$, as $\boldsymbol{p}$ remains fixed during APM. For the result stated in Theorem 3.12 above, we use several partial results of [27].

Proof. First, we use the fact stated in [27] that APM on subspaces U and V converge with geometric rate $c_F(U,V)^2$, where the rate is given by the square of the cosine of the Friedrichs angle between U and V, given by:

$$c_F(U,V) = \sup\{u^T v \mid u \in U \cap (U\cap V)^\perp, v \in V \cap (U\cap V)^\perp, \|u\| \leqslant 1, \|v\| \leqslant 1\}.$$

An intuitive generalization of this result for polyhedra $\mathcal{X}$ and $\mathcal{Y}$, considering all affine subspaces supporting the faces of $\mathcal{X}$ and $\mathcal{Y}$ is given in [27]:

LEMMA 3.13 ([27]). *For APM on polyhedra $\mathcal{X}$ and $\mathcal{Y}$ in $\mathbb{R}^D$, the convergence is geometric with rate bounded by the square of the maximal cosine of Friedrichs angle between subspaces supporting faces of $\mathcal{X}$ and $\mathcal{Y}$:*

$$\max_{\boldsymbol{x},\boldsymbol{y}} c_F\big(\operatorname{aff}_0(\mathcal{X}_{\boldsymbol{x}}), \operatorname{aff}_0(\mathcal{Y}_{\boldsymbol{y}})\big), \tag{3.13}$$

where, for any $\boldsymbol{x} \in \mathbb{R}^D$, $\mathcal{X}_{\boldsymbol{x}} \overset{\text{def}}{=} \arg\max_{\boldsymbol{v}\in\mathcal{X}} \boldsymbol{x}^\top \boldsymbol{v}$ is the face of $\mathcal{X}$ generated by direction $\boldsymbol{x}$ and $\operatorname{aff}_0(C) = \operatorname{aff}(C) - \boldsymbol{c}$ for some $\boldsymbol{c} \in C$ denotes the subspace supporting the affine hull of C, for $C = \mathcal{X}_{\boldsymbol{x}}$ or $C = \mathcal{Y}_{\boldsymbol{y}}$.

In the remaining of the proof, we bound the quantity (3.13) for our polyhedra $\mathcal{X}$ and $\mathcal{Y}$.

For this, we use the space $\mathbb{R}^{NT} = \mathbb{R}^T \times \cdots \times \mathbb{R}^T$, where the $(n-1)T+1$ to nT entries correspond to the profile of agent n, for $1 \leqslant n \leqslant N$. As in [27], we use a result connecting angles between subspaces and the eigenvalues of matrices giving the directions of these spaces:

LEMMA 3.14 ([27]). *If A and B are matrices with orthonormal rows with same number of columns, then:*

- *if all singular values of $AB^\top$ are equal to one, then $c_F\big(\operatorname{Ker} A,\ \operatorname{Ker} B\big) = 0$;*
- *else, $c_F\big(\operatorname{Ker} A,\ \operatorname{Ker} B\big)$ is equal to the largest singular value of $AB^\top$ among those that are smaller than one.*

We are left with finding a matricial representation of the faces of polyhedra $\mathcal{X}$ and $\mathcal{Y}$ and, then, bounding the corresponding singular values.

In our case, the polyhedra $\mathcal{Y}$ is an affine subspace $\mathcal{Y} = \{\boldsymbol{x} \in \mathbb{R}^{NT} \ | A\boldsymbol{x} = \sqrt{N}^{-1}\boldsymbol{p}\}$ where:

$$A \stackrel{\text{def}}{=} \sqrt{N}^{-1} J_{1,N} \otimes I_T,$$

where $\otimes$ denotes the Kronecker product. The matrix A has orthonormal rows and the linear subspace associated to $\mathcal{Y}$ is equal to $\operatorname{Ker}(A)$.

Obtaining a matricial representation of the faces of $\mathcal{X}$ is more complex. The faces of $\mathcal{X}$ are obtained by considering, for each $n \in \mathcal{N}$, subsets of the time periods that are at lower or upper bound (respectively $\underline{\mathcal{T}}_n$ and $\overline{\mathcal{T}}_n$, with $\underline{\mathcal{T}}_n \cap \overline{\mathcal{T}}_n = \emptyset$). Considering a collection of such subsets, a face of $\mathcal{X}$ can be written as:

$$\mathcal{A}_{(\overline{\mathcal{T}}_n, \underline{\mathcal{T}}_n)_n} \stackrel{\text{def}}{=} \left\{ (\boldsymbol{x})_{n,t} \ | \forall n,\ \textstyle\sum_t x_{n,t} = E_n \text{ and } \forall t \in \underline{\mathcal{T}}_n, x_{n,t} = \underline{x}_{n,t},\ \text{and } \forall t \in \overline{\mathcal{T}}_n, x_{n,t} = \overline{x}_{n,t} \right\}.$$

For some particular collection of subsets $(\overline{\mathcal{T}}_n, \underline{\mathcal{T}}_n)_n$, the set $\mathcal{A}_{(\overline{\mathcal{T}}_n, \underline{\mathcal{T}}_n)_n}$ might be empty. The linear subspace associated to $\mathcal{A}_{(\overline{\mathcal{T}}_n, \underline{\mathcal{T}}_n)_n}$ is given by $\{\boldsymbol{x} \in \mathbb{R}^{NT} | \ B\boldsymbol{x} = 0\} = \operatorname{Ker}(B)$, where the N first rows of B, corresponding to the constraints $\sum_t x_{n,t} = E_n$, are given before orthonormalization by:

$$\sqrt{T}^{-1} I_N \otimes J_{1,T}\ ,$$

and the matrix B has $b \stackrel{\text{def}}{=} \sum_n |\mathcal{T}_n|$ more rows, where $\mathcal{T}_n \stackrel{\text{def}}{=} \underline{\mathcal{T}}_n \cup \overline{\mathcal{T}}_n$, corresponding to the saturated bounds. Each of this row is given by the unit vector $\boldsymbol{e}_{n,t}^\top \in \mathbb{R}^{NT}$ for $n \in \mathcal{N}$, $t \in \mathcal{T}_n$, which gives already an orthonormalized family of (unit) vectors. Therefore, a simple orthonormalized matrix B giving the direction of $\mathcal{A}_{(\overline{\mathcal{T}}_n, \underline{\mathcal{T}}_n)_n}$ is given by:

$$B \stackrel{\text{def}}{=} \left(\operatorname{diag}\left(\sqrt{T - |\mathcal{T}_1|}^{-1} \mathbf{1}_{\mathcal{T}_1^c}^\top, \ldots, \sqrt{T - |\mathcal{T}_N|}^{-1} \mathbf{1}_{\mathcal{T}_N^c}^\top \right) \quad | \quad \operatorname{diag}\left(B_{\mathcal{T}_1}, \ldots, B_{\mathcal{T}_N}\right) \right)^\top \in \mathcal{M}_{N+b,NT}(\mathbb{R})\ ,$$

where $\mathbf{1}_{\mathcal{T}_n^c} \in \mathbb{R}^T$ is the vector where the indices in $\mathcal{T}_n^c$ are equal to 1 and 0 otherwise, and $B_{\mathcal{T}_n} \stackrel{\text{def}}{=} \sum_{\substack{1 \leqslant k \leqslant |\mathcal{T}_n| \\ \mathcal{T}_n = \{t_1, \ldots, t_{|\mathcal{T}_n|}\}}} E_{k t_k}$ is the matrix $|\mathcal{T}_n| \times T$ with indices of $\mathcal{T}_n$. We obtain the double product:

$$\begin{aligned}
(AB^\top)(A^\top B) &= \frac{1}{N} \left(\sum_n \frac{\mathbb{1}_{k \notin \mathcal{T}_n \wedge \ell \notin \mathcal{T}_n}}{T - |\mathcal{T}_n|} \right)_{1 \leqslant k, \ell \leqslant T} + \frac{1}{N} \sum_n B_{\mathcal{T}_n}^\top B_{\mathcal{T}_n} \\
&= \frac{1}{N} \left(\sum_n \frac{\mathbb{1}_{\{k,\ell\} \subset \mathcal{T}_n^c}}{T - |\mathcal{T}_n|} \right)_{1 \leqslant k, \ell \leqslant T} + \frac{1}{N} \sum_{1 \leqslant t \leqslant T} \left(\sum_n \mathbb{1}_{t \in \mathcal{T}_n} \right) E_{t,t}\ .
\end{aligned}$$

We observe that:

- if $t_0 \in \bigcap_{i=1}^N \mathcal{T}_n$, then $\boldsymbol{e}_{t_0}$ is an eigenvector associated to eigenvalue $\lambda_{t_0} = 1$;
- the vector $\mathbf{1}_{\bar{\mathcal{T}}} \stackrel{\text{def}}{=} (\mathbb{1}_{t \notin \cap_n \mathcal{T}_n})_{t \in \mathcal{T}} \in \mathbb{R}^T$, where $\bar{\mathcal{T}} \stackrel{\text{def}}{=} \cup_n \mathcal{T}_n^c$, is an eigenvector associated to eigenvalue

$\lambda = 1$. Indeed, if we denote by $\mathcal{N}_\theta = \{n \in \mathcal{N} | \theta \in \mathcal{T}_n\}$, then $[\mathbf{1}_{\bar{\mathcal{T}}}]_\theta = 1 \Leftrightarrow \mathcal{N}_\theta^c \neq \emptyset$, and for each $\theta \in \bar{\mathcal{T}}$:

$$[(AB^\top)(A^\top B)]_\theta \mathbf{1}_{\bar{\mathcal{T}}} = \frac{1}{N}\left(\sum_{i \in \mathcal{N}_\theta^c} \sum_t \frac{\mathbb{1}_{t \notin \mathcal{T}_n}}{T - |\mathcal{T}_n|}[\mathbf{1}_{\bar{\mathcal{T}}}]_t + \sum_n \mathbb{1}_{\theta \in \mathcal{T}_n}[\mathbf{1}_{\bar{\mathcal{T}}}]_\theta\right)$$
$$= \frac{1}{N}\left(\sum_{i \in \mathcal{N}_\theta^c} \frac{T - |\mathcal{T}_n|}{T - |\mathcal{T}_n|} 1 + \sum_{i \in \mathcal{N}_\theta} 1 \times [\mathbf{1}_{\bar{\mathcal{T}}}]_\theta\right) = \frac{|\mathcal{N}_\theta^c| + |\mathcal{N}_\theta|[\mathbf{1}_{\bar{\mathcal{T}}}]_\theta}{N} = [\mathbf{1}_{\bar{\mathcal{T}}}]_\theta \ .$$

To bound the other eigenvalues of the matrix $(AB^\top)(A^\top B)$, we rely on spectral graph theory arguments. Consider the weighted graph $\mathcal{G} = (\mathcal{T}, \mathcal{E})$ whose vertices are the time periods $\mathcal{T}$ and each edge $(k, \ell) \in \mathcal{T} \times \mathcal{T}$ with $k \neq \ell$ has weight $S_{k,\ell} = \frac{1}{N}\sum_n \frac{\mathbb{1}_{\{k,\ell\} \subset \mathcal{T}_n^c}}{T - |\mathcal{T}_n|}$ (if this quantity is zero, then there is no edge between k and ℓ).

The matrix $P \stackrel{\text{def}}{=} I_T - (AB^\top)(A^\top B)$ verifies for each $k \in \mathcal{T}$:

$$\sum_{\ell \neq k} -P_{k,\ell} = \sum_{\ell \neq k} \tfrac{1}{N} \sum_n \frac{\mathbb{1}_{\{k,\ell\} \subset \mathcal{T}_n^c}}{T - |\mathcal{T}_n|} = \tfrac{1}{N} \sum_n \frac{\mathbb{1}_{k \in \mathcal{T}_n^c}(T - |\mathcal{T}_n| - 1)}{T - |\mathcal{T}_n|} \tag{3.14}$$

$$= \tfrac{1}{N}\sum_n (1 - \mathbb{1}_{k \in \mathcal{T}_n}) - \tfrac{1}{N}\sum_n \frac{\mathbb{1}_{k \in \mathcal{T}_n^c}}{T - |\mathcal{T}_n|} = P_{kk} \ , \tag{3.15}$$

which shows that P is the Laplacian matrix of graph $\mathcal{G}$. As $\mathrm{Sp}(AB^\top A^\top B) = 1 - \mathrm{Sp}(P)$, we want to have a lower bound on the least eigenvalue of P greater than 0, that we denote by λ_1.

By rearranging the indices of $\mathcal{T}$ in two blocs $\bar{\mathcal{T}}$ and $\bar{\mathcal{T}}^c$, we observe that P can be written as a block diagonal matrix $P = \mathrm{diag}(P_{\bar{\mathcal{T}}}, 0_{\bar{\mathcal{T}}^c})$. As we are only interested in the positive eigenvalues of $\mathcal{P}$, we can therefore study the linear application associated to P restricted to the subspace $\mathrm{Vect}(e_t)_{t \in \bar{\mathcal{T}}}$.

As $\mathbf{1}_{\bar{\mathcal{T}}}$ is an eigenvector of P associated to $\lambda_0 = 0$, from the minmax theorem, we have:

$$\lambda_1 = \min_{u \perp \mathbf{1}_{\bar{\mathcal{T}}}, u \neq 0} \frac{u^\top P u}{u^\top u} \ . \tag{3.16}$$

Let us consider an eigenvector u realizing (3.16). Let $u_{t^*} \stackrel{\text{def}}{=} \max_t u_t$ and $u_{s^*} \stackrel{\text{def}}{=} \min_t u_t$ and let d_{s^*,t^*} be the distance between s^* and t^* in $\mathcal{G}$, and let (s^*-t^*) denote a shortest path from s^* to t^* in $\mathcal{G}$. As P is a Laplacian matrix, we have:

$$u^\top P u = \frac{1}{2}\sum_{k,\ell \in \bar{\mathcal{T}}} -P_{k,\ell}(u_k - u_\ell)^2 \geqslant \frac{1}{2}\sum_{\{k,\ell\} \in (s^*\text{-}t^*)} -P_{k,\ell}(u_k - u_\ell)^2 \geqslant \min_{k,\ell \in (s^*\text{-}t^*)}(-P_{k,\ell})\frac{(u_{t^*} - u_{s^*})^2}{d_{s^*,t^*}} \ , \tag{3.17}$$

where the last inequality is obtained from Cauchy-Schwartz inequality.

Let us write the path $(s^*\text{-}t^*) = (t_0, t_1, \ldots, t_d)$. As ($s^*$-$t^*$) is a shortest path, for each $k \in \{0, d-1\}$, the edge (t_k, t_{k+1}) exists so there exists $n \in \mathcal{N}$ such that $\{t_k, t_{k+1}\} \subset \mathcal{T}_n^c$. Moreover, for each n, we have $\mathcal{T}_n^c \cap \{t_0, \ldots, t_{k-1}, t_{k+2}, \ldots, t_d\} = \emptyset$, otherwise we could "shortcut" the path (s^*-t^*), thus we have $|\mathcal{T}_n| \geqslant d - 1$. We obtain:

$$-P_{t_k,t_{k+1}} = \frac{1}{N}\sum_n \frac{\mathbb{1}_{\{t_k,t_{k+1}\} \subset \mathcal{T}_n^c}}{T - |\mathcal{T}_n|} \geqslant \frac{1}{N(T - d + 1)} \ .$$

On the other hand, we have $(u_{t^*} - u_{s^*}) \geqslant u_{t^*} + \frac{u_{t^*}}{T-1} = \frac{T}{T-1}u_{t^*} \geqslant \frac{T}{(T-1)\sqrt{T}}\|u\|_2$.

Using these bounds and (3.17), we obtain:

$$u^\top P u \geqslant \frac{(u_{t^*} - u_{s^*})^2}{N(T - d_{s^*,t^*} + 1)d_{s^*,t^*}} \geqslant \frac{4T}{N(T+1)^2(T-1)^2}\|u\|_2^2 \geqslant \frac{4}{N(T+1)^2(T-1)}\|u\|_2^2 .$$

Therefore, $\lambda_1 \geqslant \frac{4}{N(T+1)^2(T-1)} \stackrel{\text{def}}{=} \kappa_{N,T}$ and the greatest singular value lower than one of $(AB^\top)(A^\top B)$ is $1 - \kappa_{N,T}$. We conclude by applying successively Lemma 3.14 and Lemma 3.13, to obtain the convergence rate stated in Theorem 3.12. ∎

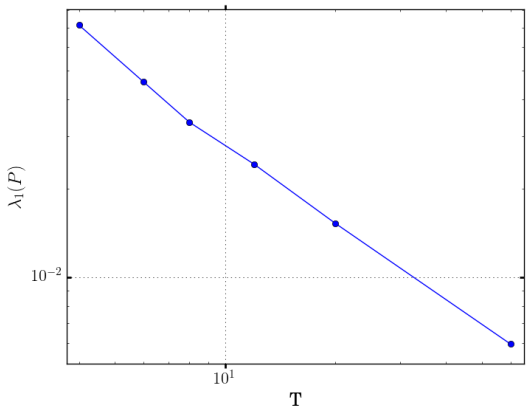


Figure 3.2: Evolution of the convergence rate, given as $\lambda_1(P)$ (lowest nonzero eigenvalue of P), with $N = 6$ and $T \in \{4, 6, 8, 12, 20, 60\}$. *The worst convergence rate is evaluated by taking* $100 \times T$ *random draws of the sets* $\mathcal{T}_n \subset \mathcal{T}$ *for each* n*, and evaluating the eigenvalue of the matrix. The slope is around -0.93, which indicates that in practice the convergence rate is* $\mathcal{O}(T^{-1})$*, faster than the upper bound in* $\mathcal{O}(T^{-3})$ *established in Theorem 3.12.*

4. Generalization to Polyhedral Agents Constraints. In this section, we extend our results to a more general framework where for each $n \in \mathcal{N}$, $\mathcal{X}_n$ is an arbitrary polyhedron, instead of having the particular structure given in (2.2). Let us now consider that $(\mathcal{X}_n)_n$ are polyhedra with, for each n:

$$\mathcal{X}_n = \{\boldsymbol{x}_n \in \mathbb{R}^T | A_n \boldsymbol{x}_n \leqslant \boldsymbol{b}_n\} \ , \tag{4.1}$$

with $A_n \in \mathcal{M}_{T,k_n}(\mathbb{R})$ with $k_n \in \mathbb{N}$. The disaggregation problem (1.2), *with* $\boldsymbol{p} \in \mathcal{P}$ *fixed*, writes:

$$\min_{\boldsymbol{x} \in \mathbb{R}^{NT}} 0 \tag{4.2a}$$

$$\text{s.t. } A_0 \boldsymbol{x} = B\boldsymbol{p} \quad (\boldsymbol{\lambda}_0) \tag{4.2b}$$

$$A_n \boldsymbol{x}_n \leqslant \boldsymbol{b}_n, \ \forall n \in \mathcal{N} \quad (\boldsymbol{\lambda}_n) \ . \tag{4.2c}$$

where $A_0 = J_{1,N} \otimes I_T$, $B = I_T$, (such that (4.2b) corresponds to the aggregation constraint $\sum_n \boldsymbol{x}_n = \boldsymbol{p}$) and $\boldsymbol{\lambda}_0 \in \mathbb{R}^T_+$, $(\boldsymbol{\lambda}_n)_{n \in \mathcal{N}} \in \mathbb{R}^{\sum_n k_n}$ are the Lagrangian multipliers associated to (4.2b) and (4.2c).

With the polyhedral constraints (4.1), the graph representation of the disaggregation problem, as illustrated in Figure 2.1 is no longer valid. Consequently, one can not directly apply Hoffman's theorem (Theorem 2.3) to obtain a characterization of disaggregation feasibility by inequalities on $\boldsymbol{p}$. However, using duality theory, Proposition 4.1 below also gives a characterization of disaggregation:

PROPOSITION 4.1. *A profile* $\boldsymbol{p} \in \mathcal{P}$ *is disaggregable iff:*

$$\forall (\boldsymbol{\lambda}_0, \boldsymbol{\lambda}_1, \dots \boldsymbol{\lambda}_N) \in \Lambda, \quad \boldsymbol{\lambda}_0^\top B\boldsymbol{p} + \textstyle\sum_{n \in \mathcal{N}} \boldsymbol{\lambda}_n^\top \boldsymbol{b}_n \geqslant 0 \ , \tag{4.3}$$

where $\Lambda \stackrel{\text{def}}{=} \{\boldsymbol{\lambda}_0 \in \mathbb{R}^{k_0}, \forall n \in \mathcal{N}, \boldsymbol{\lambda}_n \in \mathbb{R}^{k_n}_+ \mid A_0^\top \boldsymbol{\lambda}_0 A_0 + A^\top (\boldsymbol{\lambda}_n)_n = 0\}$*, with* $A \stackrel{\text{def}}{=} diag(A_n)_{n \in \mathcal{N}}$.

Proof. From strong duality, we have:

$$\min_{\boldsymbol{x} \in \mathbb{R}^{NT}} \max_{\boldsymbol{\lambda}_0 \in \mathbb{R}^{k_0}, \boldsymbol{\lambda}_n \in \mathbb{R}^{k_n}_+} \boldsymbol{\lambda}_0^\top (A_0 \boldsymbol{x} - B\boldsymbol{p}) + \sum_n \boldsymbol{\lambda}_n^\top (A_n \boldsymbol{x}_n - \boldsymbol{b}_n) \tag{4.4}$$

$$= \max_{\substack{\boldsymbol{\lambda}_0 \in \mathbb{R}^{k_0}, \boldsymbol{\lambda}_n \in \mathbb{R}^{k_n}_+ \\ \text{s.t. } \boldsymbol{\lambda}_0^\top A_0 + (\boldsymbol{\lambda}_n)_n^\top A = 0}} -\boldsymbol{\lambda}_0^\top B\boldsymbol{p} - \sum_n \boldsymbol{\lambda}_n^\top \boldsymbol{b}_n \ . \tag{4.5}$$

If the polytope $\mathcal{Y}_{\boldsymbol{p}} \cap \mathcal{X}$ given by the constraints of (4.2) is empty, then there is an infeasibility certificate $\boldsymbol{\lambda}^\top = (\boldsymbol{\lambda}_0^\top \ \boldsymbol{\lambda}_1^\top \dots \boldsymbol{\lambda}_N^\top) \in \mathbb{R}^T \times \prod_n \mathbb{R}_+^{k_n}$ such that:

$$\boldsymbol{\lambda}_0^\top A_0 + (\boldsymbol{\lambda}_n^\top)_n A = 0 \text{ and } \boldsymbol{\lambda}_0^\top B\boldsymbol{p} + (\boldsymbol{\lambda}_n)_n^\top \boldsymbol{b} < 0 \ . \tag{4.6}$$

On the other hand, if $\mathcal{Y}_{\boldsymbol{p}} \cap \mathcal{X}$ is nonempty, then a solution to the dual problem (4.5) is bounded, which implies that $\forall \boldsymbol{\lambda} \stackrel{\text{def}}{=} (\boldsymbol{\lambda}_0, (\boldsymbol{\lambda}_n)_n) \in \Lambda, \ \ \boldsymbol{\lambda}_0^\top B\boldsymbol{p} + \sum_n \boldsymbol{\lambda}_n^\top \boldsymbol{b}_n \geqslant 0$.

As opposed to Hoffman circulation's theorem where disaggregation is characterized by a finite number of inequalities, Proposition 4.1 involves a priori an infinite number of inequalities.

However, we know that the polyhedral cone Λ can be represented by a finite number of generators (edges), that is, there exists $\Lambda^* \stackrel{\text{def}}{=} \{\boldsymbol{\lambda}^{*(1)}, \dots, \boldsymbol{\lambda}^{*(d)}\}$ such that:

$$\Lambda = \left\{ \textstyle\sum_{1 \leqslant i \leqslant d} \alpha_i \boldsymbol{\lambda}^{*(i)} \mid (\alpha_i)_i \in \mathbb{R}_+^d \right\} . \tag{4.7}$$

Thus, we obtain the following corollary to Proposition 4.1:

Corollary 4.2. *There exists a finite set $\Lambda^* \subset \Lambda$ such that, for any $\boldsymbol{p} \in \mathcal{P}$, $\boldsymbol{p}$ is disaggregable iff:*

$$\forall (\boldsymbol{\lambda}_0, (\boldsymbol{\lambda}_n)_n) \in \Lambda^*, \ \ \boldsymbol{\lambda}_0 B\boldsymbol{p} + \textstyle\sum_n \boldsymbol{\lambda}_n \boldsymbol{b}_n \geqslant 0 \ . \tag{4.8}$$

Remark 5. *In the transportation case (2.2), we can write each agent constraints in the form $A_n \boldsymbol{x}_n \leqslant \boldsymbol{b}_n$ (writing the equality $\sum_t x_{n,t} = E_n$ is written as two inequalities), and Hoffman conditions (2.5) can be written in the form (8). Moreover, Theorem 3.3 ensures that one possibility for Λ^* of Corollary 4.2 is to consider the collection of 2^T multipliers corresponding to the subsets $\mathcal{T}_0 \subset \mathcal{T}$ and $\mathcal{N}_0$ minimizing (2.6). We skip the details here for brevity.*

As in the first part of the paper, we want to use APM to decompose problem (1.1) and, in the case where disaggregation is not possible, use the result of APM to generate an inequality (4.3) violated by the current profile $\boldsymbol{p}$.

In the case of impossible disaggregation, the APM converges to the orbit $(\boldsymbol{y}^\infty, \boldsymbol{x}^\infty)$, and $\boldsymbol{\mu} \stackrel{\text{def}}{=} \boldsymbol{y}^\infty - \boldsymbol{x}^\infty$ defines a separating hyperplan $\bar{\boldsymbol{x}} + \boldsymbol{\mu}^\perp$, where $\bar{\boldsymbol{x}} = \frac{\boldsymbol{y}^\infty + \boldsymbol{x}^\infty}{2}$, that satisfies, with $a \stackrel{\text{def}}{=} \bar{\boldsymbol{x}}.\boldsymbol{\mu}$ (note that $\bar{\boldsymbol{x}}$ can be replaced by any $\boldsymbol{y} \in [\boldsymbol{y}^\infty, \boldsymbol{x}^\infty]$):

$$\forall \boldsymbol{x} \in \mathcal{Y}_{\boldsymbol{p}}; \ \boldsymbol{\mu}^\top \boldsymbol{x} > a \qquad\qquad \forall \boldsymbol{x} \in \mathcal{X}; \ a > \boldsymbol{\mu}^\top \boldsymbol{x},$$

which give lower bounds on the linear problems (the second one is decomposed because A is a block-diagonal matrix, but it can also be written in one problem):

$$\begin{array}{ll} \min\limits_{\boldsymbol{x} \in \mathbb{R}^{NT}} \boldsymbol{\mu}^\top \boldsymbol{x} & \max\limits_{\boldsymbol{\lambda}_0 \in \mathbb{R}^{k_0}} -\boldsymbol{\lambda}_0^\top B\boldsymbol{p} \\ A_0 \boldsymbol{x} = B\boldsymbol{p} \ (\boldsymbol{\lambda}_0) & \boldsymbol{\mu} = -A_0^\top \boldsymbol{\lambda}_0 \end{array} \tag{4.9}$$

$$\text{and} \quad \forall n \in \mathcal{N}, \quad \begin{array}{ll} \max\limits_{\boldsymbol{x} \in \mathbb{R}^{NT}} \boldsymbol{\mu}_n \boldsymbol{x}_n & \min_{\boldsymbol{\lambda}_n \in \mathbb{R}_+^{k_n}} \boldsymbol{b}_n^\top \boldsymbol{\lambda}_n \\ A_n \boldsymbol{x}_n \leqslant \boldsymbol{b}_n \ (\boldsymbol{\lambda}_n) & \boldsymbol{\mu}_n = \boldsymbol{\lambda}_n^\top A_n \end{array} \ . \tag{4.10}$$

Strong duality on these problems implies that there exist $\boldsymbol{\lambda}_0$ and $\boldsymbol{\lambda}$ such that:

$$\boldsymbol{\mu} = -A_0^\top \boldsymbol{\lambda}_0 \text{ and } a < -\boldsymbol{\lambda}_0^\top B\boldsymbol{p} \qquad\qquad \boldsymbol{\mu} = \boldsymbol{\lambda}^\top A \text{ and } a > \boldsymbol{b}^\top \boldsymbol{\lambda} \ . \tag{4.11}$$

Thus, we obtain $(\boldsymbol{\lambda}_0, \boldsymbol{\lambda}) \in \Lambda$ satisfying (8), that is, $\boldsymbol{\lambda}_0^\top B\boldsymbol{p} + \boldsymbol{b}^\top \boldsymbol{\lambda} < 0$, and we can use this to add a new valid additional inequality on $\boldsymbol{p}$ of form (4.3), that will change the current profile $\boldsymbol{p}$:

$$\boldsymbol{\lambda}_0^\top B\boldsymbol{p} + \boldsymbol{\lambda}^\top \boldsymbol{b} \geqslant 0 \tag{4.12}$$

In Algorithm 4.1 below, we summarize the proposed decomposition of problem Equation (1.1). This is a generalization of the decomposition principle used for Algorithm 3.4.

Algorithm 4.1 Non-intrusive optimal disaggregation with polyhedral constraints

Require: Start with $\Lambda^{(0)} = \{\}$, $k = 0$, DISAG= *false*

while not DISAG **do**
  get solution $\boldsymbol{p}^{(k)}$ of problem $\min_{\boldsymbol{p}\in\mathcal{P}}\{f(\boldsymbol{p}) \mid \boldsymbol{\lambda}_0^\top B\boldsymbol{p} + \boldsymbol{\lambda}^\top \boldsymbol{b} \geqslant 0,\ \forall \boldsymbol{\lambda} \in \Lambda^{(k)}\}$
  get $\boldsymbol{\mu}^{(k)} = \boldsymbol{y}^\infty - \boldsymbol{x}^\infty \leftarrow \mathrm{APM}(\mathcal{Y}_{\boldsymbol{p}^{(k)}}, \mathcal{X})$
  if $\boldsymbol{\mu}^{(k)} \neq 0$ **then**
    obtain $\boldsymbol{\lambda}_0^{(k)} \leftarrow \max_{\boldsymbol{\lambda}_0\in\mathbb{R}^{k_0}}\{-\boldsymbol{\lambda}_0^\top B\boldsymbol{p}^{(k)} \mid \boldsymbol{\mu}^{(k)} = -A_0^\top \boldsymbol{\lambda}_0\}$
    obtain for each n, $\boldsymbol{\lambda}_n^{(k)} \leftarrow \max_{\boldsymbol{\lambda}_n\geqslant 0}\{\boldsymbol{b}_n^\top \boldsymbol{\lambda}_n \mid \boldsymbol{\mu}_n^{(k)} = \boldsymbol{\lambda}_n^\top A_n\}$
    add $\Lambda^{(k+1)} = \Lambda^{(k)} \cup \{(\boldsymbol{\lambda}_0^{(k)}, \boldsymbol{\lambda}^{(k)})\}$
  else
    Return DISAG = *true*, $\boldsymbol{p}^{(k)}$ as optimal solution
  end
  $k \leftarrow k + 1$
done

REMARK 6. *We use the fact that $\boldsymbol{\mu} = \boldsymbol{y}^\infty - \boldsymbol{x}^\infty$ although, as before, we only have an* approximation *of this quantity. The approximation has to be precise enough to ensure that the solution obtained verifies $\boldsymbol{\lambda}_0^\top B\boldsymbol{p} + \boldsymbol{b}^\top \boldsymbol{\lambda} < 0$. In practice, one can proceed as in the transportation case and Algorithm 3.3 use a large $\varepsilon_{\mathrm{cvg}}$ as stopping criteria in APM, then compute $(\boldsymbol{\lambda}_0, \boldsymbol{\lambda}) \in \Lambda$ and check if $\boldsymbol{\lambda}_0^\top B\boldsymbol{p} + \boldsymbol{b}^\top \boldsymbol{\lambda} < 0$. If this is not the case, restart with $\varepsilon_{\mathrm{cvg}} = \varepsilon_{\mathrm{cvg}}/2$.*

REMARK 7. *When $\mathcal{Y}_{\boldsymbol{p}} = \{\boldsymbol{x} \in \mathbb{R}^{NT} | A_0\boldsymbol{x} = B_0\boldsymbol{p}\} = \{\boldsymbol{x} | \sum_n \boldsymbol{x}_n = \boldsymbol{p}\}$, we can obtain a non-intrusive version of APM on $\mathcal{Y}_{\boldsymbol{p}}$ and $\mathcal{X}$, similar to Algorithm 3.3. In this case, (4.11) ensures that we have $\boldsymbol{\mu}_{n,t} = -[\boldsymbol{\lambda}_0]_t$ for each $n \in \mathcal{N}$, and $\boldsymbol{\lambda}_0$ is fixed by $\boldsymbol{\mu}$). The only difference with the transportation case for a non-intrusive APM in the general polyhedral case, is in the way of computing the valid constraint violated by $\boldsymbol{p}$. Thus, Lines 16 to 19 of Algorithm 3.3 have to be replaced by Algorithm 4.2.*

Algorithm 4.2 Modification of Lines 16-19 of Algorithm 3.3 for NI-APM with polyhedral constraints

for each agent $n \in \mathcal{N}$ **do**
  compute M_n optimal value of (4.10).
done
Operator computes $M \leftarrow \mathrm{SMCA}((M_n)_n)$
if $-\boldsymbol{\nu}.\boldsymbol{p} + M < 0$ **then**
  return DISAG $\leftarrow$ FALSE, $-\boldsymbol{\nu}, M$

4.1. Link with Benders' decomposition. In this generalized case, we obtain an algorithm related to Benders' decomposition [5] (recall that in our specific case (4.2), the cost function does not involve the variable $\boldsymbol{x}$ but only variable $\boldsymbol{p}$).

The difference between the proposed Algorithm 4.1 and Benders' decomposition is on the way of generating the new cut. Benders' decomposition would directly solve the dual problem (4.5): $\max_{\boldsymbol{\lambda}}\{-\boldsymbol{\lambda}_0^\top B_0\boldsymbol{p} - (\boldsymbol{\lambda}_n)_n^\top \boldsymbol{b} \mid \boldsymbol{\lambda}_0 A_0 + (\boldsymbol{\lambda}_n)_n A = 0\}$ and obtain a cut if it is unbounded. However, this problem involves the constraints of all users (through A and $\boldsymbol{b}$), and is it not straightforward to obtain a method to solve this problem in a decentralized and efficient way.

5. Numerical examples.

5.1. An illustrative example with T=4. In this section we illustrate the iterations of the method proposed in this paper on an example with $T = 4$ and $N = 3$. Assuming that we have to satisfy the aggregate constraint $\sum_t p_t = \sum_n E_n$, we can use the projections on this affine space of solutions of master problems $(\boldsymbol{p}^{(s)})_s$ to visualize them in dimension 3.

One can wonder if, in the transportation case, applying Algorithm 3.4 or Algorithm 4.1 will always

lead to the same cuts and solutions: the answer is no, as shown by the instance considered in this section, for which Algorithm 3.4 converges in 3 iterations and Algorithm 4.1 needs 4 iterations.

We consider the problem (1.1) with agents constraints (2.2) with parameters $\underline{\boldsymbol{x}} \overset{\text{def}}{=} 0$ and:

$$\overline{\boldsymbol{x}} \overset{\text{def}}{=} \begin{matrix}[0.8, 0.2, 0.7, 0.1] \\ [0.5, 0.1, 0.3, 0.6] \\ [0.1, 0.1, 0.7, 0.2]\end{matrix} \quad , \quad \begin{matrix}E_1 = 1.8 \\ E_2 = 0.4 \\ E_3 = 1.1\end{matrix} \quad , \qquad \forall \boldsymbol{p} \in \mathbb{R}^4, f(\boldsymbol{p}) \overset{\text{def}}{=} \sum_{1 \leqslant t \leqslant 4} 0.8 \times p_t + 0.1 \times p_t^2 \ . \tag{5.1}$$

Considering the aggregate equality constraint $\sum_{1 \leqslant t \leqslant 4} p_t = \sum_{1 \leqslant n \leqslant 3} E_n = 3.3$, we use the canonical projection of 4 dimensional vectors into the 3 dimensional space (p_1, p_2, p_3) to visualize the cuts and solutions. In this example, there exist $2^T - 2 = 14$ nontrivial Hoffman inequalities characterizing disaggregation from Theorem 2.3. The projection of the obtained polytope $\mathcal{P}_{\text{D}}$, as defined in (2.1), is represented in Figure 5.1a. One can remark that this polytope has only 6 facets. Our Algorithm 3.4 applied on this instance with $\varepsilon_{\text{dis}} = 10^{-3}$ and $\varepsilon_{\text{cvg}} = 10^{-5}$ converges in 3 iterations, with successive solutions of the master problem (3.1) and cuts added:

$$\begin{aligned}
\boldsymbol{p}^{(1)} &= [1., 0.4, 1., 0.9] && \overset{\text{cut}}{\longrightarrow} && p_1 + p_2 + p_4 \leqslant 1.9 \\
\boldsymbol{p}^{(2)} &= [0.75, 0.4, 1.4, 0.75] && \overset{\text{cut}}{\longrightarrow} && p_2 + p_3 + p_4 \leqslant 2.4 \\
\boldsymbol{p}^{(3)} &= [0.9, 0.4, 1.4, 0.6] \ .
\end{aligned}$$

Figure 5.1b represents in the projection space the three successive solutions and the two generated cuts (in red for each iteration).

On the other hand, applying Algorithm 4.1 with the same precision parameters$(\varepsilon_{\text{dis}}, \varepsilon_{\text{cvg}})$, there are 3 cuts generated and 4 resolutions of master problem needed for convergence, given by (we refer the reader to Remark 6 for the numerical precision obtained in the values):

$$\begin{aligned}
\hat{\boldsymbol{p}}^{(1)} &= [1., 0.4, 1., 0.9] && \overset{\text{cut}}{\longrightarrow} && -0.25p_1 - 0.25p_2 + 1.0p_3 - 0.5p_4 \geqslant 0.75 \\
\hat{\boldsymbol{p}}^{(2)} &= [0.8097, 0.4, 1.3984, 0.6919] && \overset{\text{cut}}{\longrightarrow} && 1.0p_1 - 0.509p_2 + 0.018p_3 - 0.509p_4 \geqslant 0.4161 \\
\hat{\boldsymbol{p}}^{(3)} &= [0.9062, 0.4, 1.3823, 0.6115] && \overset{\text{cut}}{\longrightarrow} && -0.333p_1 - 0.333p_2 + 1.0p_3 - 0.333p_4 \geqslant 0.7666 \\
\hat{\boldsymbol{p}}^{(4)} &= [0.9, 0.4, 1.4, 0.6] \ .
\end{aligned}$$

The 4 successive solutions and the 3 added cuts are represented in the three dimensional space on Figure 5.1c: we observe that the last cut needed to obtain the convergence of Algorithm 4.1, corresponds to the first one added with Algorithm 3.4.

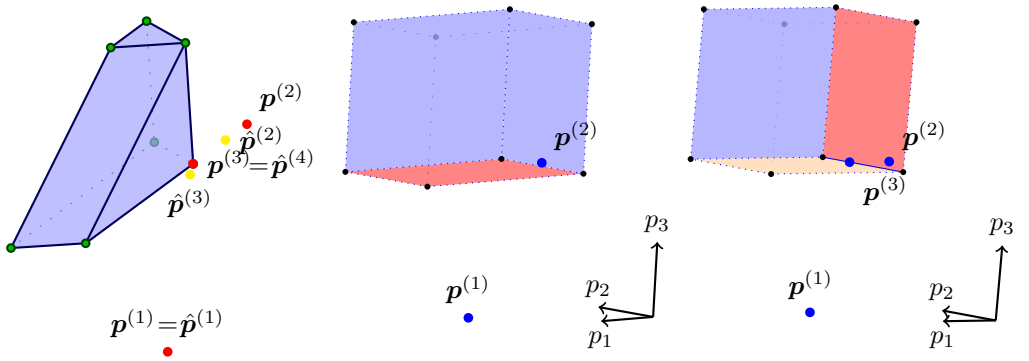
Due to the strict convexity of the cost function $\boldsymbol{p} \mapsto f(\boldsymbol{p})$, the final solution obtained is the same, unique aggregated optimal solution of (1.1). The 4 successive solutions and the 3 added cuts are represented in the three dimensional space on Figure 5.1c: we observe that the last cut needed to obtain the convergence of Algorithm 4.1, corresponds to the first one added with Algorithm 3.4.

5.2. A nonconvex example: management of a microgrid. In this section, we illustrate the proposed method on a larger scale practical example from energy. We consider an electricity microgrid [21] composed of N electricity consumers with flexible appliances (such as electric vehicles or water heaters), a photovoltaic (PV) power plant and a conventional generator. The operator of the microgrid aims at satisfying the demand constraints of consumers over a set of time periods $\mathcal{T} = \{1, \dots, T\}$, while minimizing the energy cost for the community. We have the following characteristics:

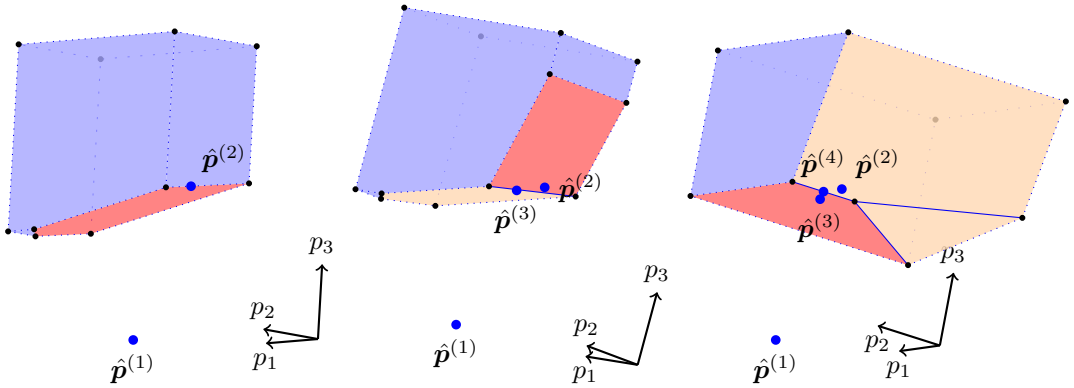
- the PV plant generates a nondispatchable power profile $(p_t^{\text{PV}})_{t \in \mathcal{T}}$ at marginal cost zero;
- the conventional generator has a starting cost C^{ST}, minimal and maximal power production $\underline{p}^g, \overline{p}^g$, and piecewise-linear and continuous generation cost function $p^g \mapsto f(p^g)$:

$$f(p^g) = \alpha_k + c_k p^g, \text{ if } p^g \in \mathcal{I}_k \overset{\text{def}}{=} [\theta_{k-1}, \ \theta_k[, \ k = 1 \dots K,$$

where $\theta_0 \overset{\text{def}}{=} 0$ and $\theta_K \overset{\text{def}}{=} \overline{p}^g$;



(a) Projected Polyhedron $\mathcal{P}_{\mathrm{D}}$ (b) Solutions and cuts (projected) obtained with Algorithm 3.4



(c) Solutions and cuts (projected) obtained with Algorithm 4.1

Figure 5.1: Illustration of the iterations of the proposed decomposition method. *The cut* $p_3 \geqslant 1.4$, *which is added at first for Algorithm 3.4 , is only added at the third iteration of Algorithm 4.1*

- each agent $n \in \mathcal{N}$ has some flexible appliances which require a global energy demand E_n on $\mathcal{T}$, and has consumption constraints on the total household consumption, on each time period $t \in \mathcal{T}$, that are formulated with $\underline{\boldsymbol{x}}_n, \overline{\boldsymbol{x}}_n$. These parameters are confidential because they could for instance contain some information on agent n habits.

The master problem (3.1) can be written as the following MILP (5.2):

$$\min_{\boldsymbol{p},\boldsymbol{p}^g,(\boldsymbol{p}_k^g),(\boldsymbol{b}_k),\boldsymbol{b}^{\mathrm{ON}},\boldsymbol{b}^{\mathrm{ST}}} \sum_{t\in\mathcal{T}} \left(\alpha_1 b_t^{\mathrm{ON}} + \sum_k c_k p_{kt}^g + C^{\mathrm{ST}} b_t^{\mathrm{ST}}\right) \tag{5.2a}$$

$$p_t^g = \textstyle\sum_{k=1}^K p_{k,t}^g, \ \forall t \in \mathcal{T} \tag{5.2b}$$

$$b_{k,t}(\theta_k - \theta_{k-1}) \leqslant p_{k,t}^g \leqslant b_{k-1,t}(\theta_k - \theta_{k-1}), \ \forall 1 \leqslant k \leqslant K, \ \forall t \in \mathcal{T} \tag{5.2c}$$

$$b_t^{\mathrm{ST}} \geqslant b_t^{\mathrm{ON}} - b_{t-1}^{\mathrm{ON}}, \ \forall t \in \{2,\dots,T\} \tag{5.2d}$$

$$\underline{p}^g b_t^{\mathrm{ON}} \leqslant p_t^g \leqslant \overline{p}^g b_t^{\mathrm{ON}}, \ \forall t \in \mathcal{T} \tag{5.2e}$$

$$b_t^{\mathrm{ON}}, b_t^{\mathrm{ST}}, b_{1,t}, \dots, b_{K-1,t} \in \{0,1\}, \ \forall t \in \mathcal{T} \tag{5.2f}$$

$$\boldsymbol{p} \leqslant \boldsymbol{p}^{\mathrm{PV}} + \boldsymbol{p}^g \tag{5.2g}$$

$$\boldsymbol{p}^\top \mathbb{1}_T = \boldsymbol{E}^\top \mathbb{1}_N \tag{5.2h}$$

$$\underline{\boldsymbol{x}}^\top \mathbb{1}_N \leqslant \boldsymbol{p} \leqslant \overline{\boldsymbol{x}}^\top \mathbb{1}_N \ . \tag{5.2i}$$

In this formulation (5.2b-5.2c), where $b_{0,t} \overset{\text{def}}{=} 1$ and $b_{K,t} \overset{\text{def}}{=} 0$, are a mixed integer formulation of the

generation cost function f. One can show that the Boolean variable $b_{k,t}$ is equal to one *iff* $p_t^g \geqslant \theta_k$ for each $k \in \{1, \dots, K-1\}$. Note that only α_1 appears in (5.2a) because of the continuity of f.

Constraints (5.2d-5.2e) ensure the on/off and starting constraints of the power plant, (5.2g) ensures that the power allocated to consumption is not above the total production, and (5.2h-5.2i) are the aggregated feasibility conditions already referred to in (2.3). The nonconvexity of (5.2) comes from the existence of starting costs and constraints of minimal power, which makes necessary to use Boolean state variables $b^{\text{ST}}, b^{\text{ON}}$.

We simulate the problem described above for different values of $N \in \{2^4, 2^5, 2^6, 2^7, 2^8\}$ and one hundred instances with random parameters for each value of N. A scaling factor $\kappa_N = N/20$ is applied on parameters to ensure that production capacity is large enough to meet consumers demand. The parameters are chosen as follows:

- $T = 24$ (hours of a day);
- production costs: $K = 3$, $\theta = [0, 70, 100, 300]\kappa_N, \boldsymbol{c} = [0.2, 0.4, 0.5]$, $\underline{p}^g = 50\kappa_N, \overline{p}^g = 300\kappa_N$, $\alpha_1 = 4$ and $C^{\text{ST}} = 15$;
- photovoltaic: $p_t^{\text{PV}} = \left[50(1-\cos(\frac{(t-6)2\pi}{16}) + \mathcal{U}([0,10])\right]\kappa_N$ for $t \in \{6, \dots, 20\}$, $p_t^{\text{PV}} = 0$ otherwise;
- consumption parameters are drawn randomly with: $\underline{x}_{n,t} \sim \mathcal{U}([0,10])$, $\overline{x}_{n,t} \sim \mathcal{U}([0,5]) + \underline{x}_{n,t}$ and $E_n \sim \mathcal{U}([\mathbb{1}_T^\top \underline{\boldsymbol{x}}_n, \mathbb{1}_T^\top \overline{\boldsymbol{x}}_n])$, so that individual feasibility ($\mathcal{X}_n \neq \emptyset$) is ensured.

$N =$	2^4	2^5	2^6	2^7	2^8
# master	193.6	194.1	225.5	210.9	194.0
# projs.	9507	15367	24319	26538	26646

Table 5.1: number of subproblems solved (average on 100 instances)

We implement Algorithm 3.4 using Python 3.5. The MILP (5.2) is solved using Cplex Studio 12.6 and Pyomo interface. Simulations are run on a single core of a cluster at 3GHz. For the convergence criteria (see Lines 11 and 12 of Algorithm 3.3), we use $\varepsilon_{\text{dis}} = 0.01$ with the operator norm defined by $|||\boldsymbol{x}||| = \max_{n \in \mathcal{N}} \sum_t |x_{n,t}|$ (to avoid the $\sqrt{N}$ factor in the convergence criteria appearing with $\|.\|_2$), and starts with $\varepsilon_{\text{cvg}} = 0.1$. The largest instances took around 10 minutes to be solved in this configuration and without parallel implementation. As the CPU time needed depends on the cluster load, it is not a reliable indicator of the influence on N on the complexity of the problems. Moreover, one advantage of the proposed method is that the projections in APM can be computed locally by each agent in parallel, which could not be implemented here for practical reasons.

Table 5.1 gives the number of master problems solved and the total number of projections computed, on average over the hundred instance for each value of N.

One observes that the number of master problems (5.2) solved (number of "cuts" added), remains almost constant when N increases. In all instances, this number is way below the upper bound of $2^{24} > 1,6 \times 10^7$ possible constraints (see Proposition 3.10), which suggests that only a limited number of constraints are added in practice. The average total number of projections computed for each instance (total number of iterations of the **while** loop of Algorithm 3.3, Line 1 over all calls of APM in the instance) increases in a sublinear way which is even better that one could expect from the upper bound given in Theorem 3.12.

6. Conclusion. We provided a non-intrusive algorithm that enables to compute an optimal resource allocation, solution of a–possibly nonconvex–optimization problem, and affect to each agent an individual profile satisfying a global demand and lower and upper bounds constraints. Our method uses local projections and works in a distributed fashion. Hence, the resolution of the problem is still efficient even in the case of a very large number of agents. The method is also privacy-preserving, as agents do not need to reveal any information on their constraints or their individual profile to a third party.

Several extensions and generalizations can be considered for this work. Section 4 generalizes the procedure to arbitrary polyhedral constraints for agents. However, the number of constraints (cuts)

added to the master problem is not proved to be finite as done in the transportation case. Proving that only a finite number of constraints can be added (maybe up to a refinement procedure of the current constraint obtained) will enable to have a termination result for the algorithm in the general polyhedral case. In the transportation case, we showed the geometric convergence of APM with a rate linear in the number of agents. Moreover, the number of cuts added in the procedure is finite but the upper bound that we have remains exponential. In practice however, the number of constraints to consider remains small, as seen in Section 5. A thinner upper bound on the number of cuts added in the algorithm in this case would constitute an interesting result.

Appendix A. Proof of Proposition 3.4.

Proof of Item (i). Let us write the stationarity conditions associated to problem (3.3):

$$\forall n \in \mathcal{N}, \forall t \in \mathcal{T}, \quad 0 = (x_{n,t} - y_{n,t}) - \lambda_n - \underline{\mu}_{n,t} + \overline{\mu}_{n,t} \quad \text{and} \quad y_{n,t} = x_{n,t} + \nu_t \ . \tag{A.1}$$

By summing the preceding equalities on $\mathcal{T}$ and $\mathcal{N}$, we obtain the three equalities:

$$\sum_t \nu_t = \sum_t y_{n,t} - E_n, \ \forall n \in \mathcal{N} \qquad p_t = \sum_n x_{n,t} + N\nu_t, \ \forall t \in \mathcal{T} \tag{A.2}$$

$$|\mathcal{T}_n^\circ|\lambda_n = E_n - \sum_{t \in \underline{\mathcal{T}}_n} \underline{x}_{n,t} - \sum_{t \in \mathcal{T}_n^\circ} y_{n,t} - \sum_{t \in \overline{\mathcal{T}}_n} \overline{x}_{n,t}, \forall n \in \mathcal{N} \ , \tag{A.3}$$

where we define for each $n \in \mathcal{N}$:

$$\mathcal{T}_n^\circ \stackrel{\text{def}}{=} \{t \mid \underline{x}_{n,t} < x_{n,t} < \overline{x}_{n,t}\}, \quad \underline{\mathcal{T}}_n = \{t \mid x_{n,t} = \underline{x}_{n,t}\} \ \text{ and } \ \overline{\mathcal{T}}_n = \{t \mid x_{n,t} = \overline{x}_{n,t}\} \ .$$

From (A.2) and the aggregate equality $\sum_n E_n = \sum_t p_t$ we obtain: $\sum_t \nu_t = 0$ and:

$$\forall n \in \mathcal{N}, \ \textstyle\sum_{t \in \mathcal{T}} y_{n,t} = E_n. \tag{A.4}$$

Suppose that Item (i) is false: there exists $n \notin \mathcal{N}_0$ and $\hat{t} \in \mathcal{T}_0$ such that $x_{n,\hat{t}} < \overline{x}_{n,\hat{t}}$. We have:

$$x_{n,\hat{t}} \geqslant y_{n,\hat{t}} + \lambda_n = x_{n,\hat{t}} + \nu_{\hat{t}} + \lambda_n \implies \lambda_n < 0 \ .$$

Immediately, we have $\overline{\mathcal{T}}_n \subset \mathcal{T}_0$: indeed, for $t \in \overline{\mathcal{T}}_n$, we have:

$$y_{n,t} + \lambda_n \geqslant \overline{x}_{n,t} \implies y_{n,t} \geqslant \overline{x}_{n,t} - \lambda_n > \overline{x}_{n,t} \implies t \in \mathcal{T}_0 \ .$$

From the condition (A.4) and from $\nu_t > 0$ for each $t \in \overline{\mathcal{T}}_n$ because $\overline{\mathcal{T}}_n \subset \mathcal{T}_0$, we get:

$$0 = \sum_{t \in \mathcal{T}} (y_{n,t} - x_{n,t}) = \sum_{t \in \underline{\mathcal{T}}_n} (y_{n,t} - \underline{x}_{n,t}) + \sum_{t \in \mathcal{T}_n^\circ} (-\lambda_n) + \sum_{t \in \overline{\mathcal{T}}_n} \nu_t \iff \sum_{t \in \underline{\mathcal{T}}_n} (y_{n,t} - \underline{x}_{n,t}) = \sum_{t \in \mathcal{T}_n^\circ} \lambda_n - \sum_{t \in \overline{\mathcal{T}}_n} \nu_t,$$

which is strictly negative: this implies that there exists $t' \in \underline{\mathcal{T}}_n$ such that $y_{n,t'} < \underline{x}_{n,t'}$. Necessarily, $t' \notin \mathcal{T}_0$ because $\nu_{t'} = y_{n,t'} - x_{n,t'} < \underline{x}_{n,t'} - \underline{x}_{n,t'} = 0$. Then, as we have $\sum_{m \in \mathcal{N}} y_{m,t'} = p_{t'} \geqslant \sum_m \underline{x}_{m,t'}$, there exists $m \in \mathcal{N}$ such that $y_{m,t'} > \underline{x}_{m,t'}$. If $\lambda_m \leqslant 0$, and as $x_{m,t'} = y_{m,t'} - \nu_{t'} > \underline{x}_{m,t'}$, we get:

$$x_{m,t'} = \min(\overline{x}_{m,t'}, y_{m,t'} + \lambda_m) \leqslant y_{m,t'} + \lambda_m \leqslant y_{m,t'} = x_{m,t'} + \nu_{t'} < x_{m,t'} \ ,$$

which is impossible, thus $\lambda_m > 0$. Now, we observe that $\mathcal{T}_n^\circ \subset \mathcal{T}_0$. Indeed, otherwise, if $t" \in \mathcal{T}_n^\circ \cap \mathcal{T}_0^c$, we have $\nu_{t"} = -\lambda_n > 0$ and $x_{m,t"} = y_{m,t"} - \nu_{t"} < y_{m,t"} < \overline{x}_{m,t"}$, thus we get:

$$x_{m,t"} = \max(\underline{x}_{m,t"}, y_{m,t"} + \lambda_m) \geqslant y_{m,t"} + \lambda_m > y_{m,t"} = x_{m,t"} + \nu_{t"} + \lambda_m > x_{m,t"} \ ,$$

which is impossible, thus $\mathcal{T}_n^\circ \subset \mathcal{T}_0$.

Finally, since $\overline{\mathcal{T}}_n^c \neq \emptyset$, consider $t_0 \in \arg\min_{t \notin \overline{\mathcal{T}}_n} \{\overline{x}_{n,t} - y_{n,t}\}$. By (A.3), we obtain:

$$y_{n,t_0} + \lambda_n < \overline{x}_{n,t_0} \iff E_n - \sum_{t \in \underline{\mathcal{I}}_n} \underline{x}_{n,t} - \sum_{t \in \mathcal{T}_n^\circ} y_{n,t} - \sum_{t \in \overline{\mathcal{T}}_n} \overline{x}_{n,t} < |\mathcal{T}_n^\circ|(\overline{x}_{n,t_0} - y_{n,t_0}) \tag{A.5}$$

and thus:

$$\begin{aligned} E_n - \sum_{t \in \mathcal{T}_0^c} \underline{x}_{n,t} - \sum_{t \in \mathcal{T}_0} \overline{x}_{n,t} &= E_n - \sum_{t \in \underline{\mathcal{I}}_n} \underline{x}_{n,t} + \sum_{t \in \underline{\mathcal{I}}_n \cap \mathcal{T}_0} \underline{x}_{n,t} - \sum_{t \in \overline{\mathcal{T}}_n \cup \mathcal{T}_n^\circ} \overline{x}_{n,t} - \sum_{t \in \underline{\mathcal{I}}_n \cap \mathcal{T}_0} \overline{x}_{n,t} \quad (\text{as } \overline{\mathcal{T}}_n \cup \mathcal{T}_n^\circ \subset \mathcal{T}_0) \\ &< \sum_{t \in \mathcal{T}_n^\circ} (\overline{x}_{n,t_0} - y_{n,t_0}) - (\overline{x}_{n,t} - y_{n,t}) + \sum_{\underline{\mathcal{I}}_n \cap \mathcal{T}_0} (\underline{x}_{n,t} - \overline{x}_{n,t}) \quad (\text{from (A.5)}) \qquad \text{(A.6)} \\ &\leqslant 0 \quad (\text{from the definition of } t_0 \text{ and } \underline{x}_{n,t} \leqslant \overline{x}_{n,t}), \qquad \text{(A.7)} \end{aligned}$$

which contradicts $n \notin \mathcal{N}_0$ and terminates the proof for Item (i). ∎

Proof of Item (ii). To prove (ii), we see that if t is such that $\nu_t > 0$, then all the facts said before are true for $n \notin \mathcal{N}_0$ if we consider $\mathcal{T}_0' \stackrel{\text{def}}{=} \{t | \nu_t > 0\}$ instead of $\mathcal{T}_0$. In that case we will have $\lambda_n < 0$. However we cannot have $t" \in \mathcal{T}_n^\circ \cap \mathcal{T}_0'^c$ because this would mean $\nu_{t"} = -\lambda_n > 0$ but we have $\nu_{t"} \leqslant 0$ because $t" \notin \mathcal{T}_0'$. Thus $\mathcal{T}_n^\circ$ is necessarily empty, and if there is $t \in \underline{\mathcal{I}}_n \cap \mathcal{T}_0'$, the same sequence of inequalities as (A.6-A.7) show a contradiction. Consequently, for each $t \in \mathcal{T}_0'$, $x_{n,t} = \overline{x}_{n,t}$ and $y_{n,t} = x_{n,t} + \nu_t > \overline{x}_{n,t}$, thus $t \in \mathcal{T}_0$ and $\mathcal{T}_0' \subset \mathcal{T}_0$. The other inclusion is immediate. ∎

Proof of Item (iii). Suppose on the contrary that there exists $n \in \mathcal{N}_0$ such that $\lambda_n \geqslant 0$. For $t \in \mathcal{T}_n^\circ$, we have $\nu_t = -\lambda_n \leqslant 0$, thus, $\mathcal{T}_n^\circ \subset \mathcal{T}_0^c$. Then, if $t \in \mathcal{T}_0$ and if $x_{n,t} < \overline{x}_{n,t}$, we would have:

$$x_{n,t} = \max(\underline{x}_{n,t}, y_{n,t} + \lambda_n) \geqslant x_{n,t} + 0 + \nu_t > x_{n,t} \ ,$$

which is impossible, thus $x_{n,t} = \overline{x}_{n,t}$, and $\mathcal{T}_0 \subset \overline{\mathcal{T}}_n$. As we show independently in Item (v) that $\mathcal{T}_0 \neq \emptyset$, we know $\overline{\mathcal{T}}_n \neq \emptyset$. Let us consider $t_0 \in \arg\min_{t \notin \underline{\mathcal{I}}_n} \{y_{n,t} - \underline{x}_{n,t}\}$. By (A.3), we obtain:

$$y_{n,t_0} + \lambda_n > \underline{x}_{n,t_0} \iff E_n - \sum_{t \in \underline{\mathcal{I}}_n} \underline{x}_{n,t} - \sum_{t \in \mathcal{T}_n^\circ} y_{n,t} - \sum_{t \in \overline{\mathcal{T}}_n} \overline{x}_{n,t} > |\mathcal{T}_n^\circ|(\underline{x}_{n,t_0} - y_{n,t_0}) \tag{A.8}$$

and thus:

$$\begin{aligned} E_n - \sum_{t \in \mathcal{T}_0^c} \underline{x}_{n,t} - \sum_{\mathcal{T}_0} \overline{x}_{n,t} &= E_n - \sum_{t \in \underline{\mathcal{I}}_n} \underline{x}_{n,t} - \sum_{t \in \mathcal{T}_0^c \cap \overline{\mathcal{T}}_n} \underline{x}_{n,t} - \sum_{t \in \mathcal{T}_n^\circ} \underline{x}_{n,t} - \sum_{t \in \overline{\mathcal{T}}_n} \overline{x}_{n,t} + \sum_{t \in \mathcal{T}_0^c \cap \overline{\mathcal{T}}_n} \overline{x}_{n,t} \quad (\text{as } \mathcal{T}_0 \subset \overline{\mathcal{T}}_n) \\ &> \sum_{t \in \mathcal{T}_n^\circ} \big((y_{n,t} - \underline{x}_{n,t}) - (y_{n,t_0} - \underline{x}_{n,t_0})\big) + \sum_{t \in \mathcal{T}_0^c \cap \overline{\mathcal{T}}_n} \overline{x}_{n,t} - \underline{x}_{n,t} \quad (\text{from (A.8)}) \qquad \text{(A.9)} \\ &\geqslant 0 \qquad (\text{from the definition of } t_0 \text{ and } \underline{x}_{n,t} \leqslant \overline{x}_{n,t}), \qquad \text{(A.10)} \end{aligned}$$

which contradicts $n \in \mathcal{N}_0$ and terminates the proof of Item (iii). ∎

Proof of Item (iv). From (ii), we know that $\mathcal{T}_0^c = \{t | \nu_t \leqslant 0\}$, thus, if $t \notin \mathcal{T}_0$ and $n \in \mathcal{N}_0$, if $x_{n,t} > \underline{x}_{n,t}$ then we would have $x_{n,t} \leqslant y_{n,t} + \lambda_n = x_{n,t} + \nu_t + \lambda_n < x_{n,t}$, which is a contradiction. ∎

Proof of Item (v). From $\sum_t \nu_t = 0$, we see that if $\mathcal{T}_0 = \emptyset$, then this means that $\nu_t = 0$ for all $t \in \mathcal{T}$, and thus $\boldsymbol{y} = \boldsymbol{x}$ which is a contradiction. Thus there exists t_0 such that $\nu_{t_0} > 0$ and for the same reason, there exists t_0' such that $\nu_{t_0'} < 0$.

If $\mathcal{N}_0 = \emptyset$, then using (i), we would have for all n, $y_{n,t_0} > \overline{x}_{n,t_0}$ and thus $p_{t_0} > \sum_{n \in \mathcal{N}} \overline{x}_{n,t_0}$, which contradicts the aggregate upper bound constraint $\forall t,\ p_t \leqslant \sum_{n \in \mathcal{N}} \overline{x}_{n,t}$.

If $\mathcal{N}_0^c = \emptyset$, then using (iv), we would have for all n, $y_{n,t_0'} < \underline{x}_{n,t_0'}$ and thus $p_{t_0'} < \sum_{n \in \mathcal{N}} \underline{x}_{n,t_0'}$, which contradicts the aggregate lower bound constraint $\forall t,\ p_t \geqslant \sum_{n \in \mathcal{N}} \underline{x}_{n,t}$. ∎

Appendix B. Proof of Lemma 3.9.

Proof of Item (i). From $\boldsymbol{x}^{(K)} = P_{\mathcal{X}}(\boldsymbol{y}^{(K-1)})$ and $\boldsymbol{y}^{(K)} = P_{\mathcal{Y}}(\boldsymbol{x}^{(K)})$, we obtain, similarly to (A.1):

$$\text{(B.1)} \quad \forall n \in \mathcal{N}, \forall t \in \mathcal{T}, \quad 0 = (x_{n,t}^{(K)} - y_{n,t}^{(K-1)}) - \lambda_n^{(K)} - \underline{\mu}_{n,t}^{(K)} + \overline{\mu}_{n,t}^{(K)} \quad \text{and} \quad y_{n,t}^{(K)} = x_{n,t}^{(K)} + \nu_t^{(K)} \ .$$

where the Lagrangian multipliers $\lambda_n^{(K)}, \underline{\mu}_{n,t}^{(K)}, \overline{\mu}_{n,t}^{(K)}$ (resp. $\nu_t^{(K)}$) are associated to the quadratic problem characterizing the projections $P_{\mathcal{X}}(\boldsymbol{y}^{(K-1)})$ (resp. $\boldsymbol{y}^{(K)} = P_{\mathcal{Y}}(\boldsymbol{x}^{(K)})$). We obtain equalities similar to (A.2, A.3). We proceed as for Proposition 3.4(i) and suppose that there exists $n \notin \mathcal{N}_0$ and $\hat{t} \in \mathcal{T}_0$ such that $x_{n,\hat{t}} < \overline{x}_{n,\hat{t}}$. Then, as $\left\|\boldsymbol{y}^{(K)} - \boldsymbol{y}^{\infty}\right\| \leqslant \frac{\varepsilon_{\text{cvg}}}{1-\rho}$ and $\sum_{t\in\mathcal{T}} \boldsymbol{y}_t^{(K)} = \sum_{t\in\mathcal{T}} \boldsymbol{y}_t^{\infty}$, we have for each $n \in \mathcal{N}, t \in \mathcal{T}$, $|y_{n,t}^{(K)} - y_{n,t}^{\infty}| \leqslant \frac{\varepsilon_{\text{cvg}}}{2(1-\rho)}$, and thus we get:

$$\begin{aligned} &\overline{x}_{n,\hat{t}} \geqslant x_{n,\hat{t}}^{(K)} \geqslant y_{n,\hat{t}}^{(K-1)} + \lambda_n^{(K)} \geqslant y_{n,\hat{t}}^{\infty} - \tfrac{\varepsilon_{\text{cvg}}}{2(1-\rho)} + \lambda_n^{(K)} = x_{n,\hat{t}}^{\infty} + \nu_{\hat{t}}^{\infty} - \tfrac{\varepsilon_{\text{cvg}}}{2(1-\rho)} + \lambda_n^{(K)} \\ &\implies \lambda_n^{(K)} < \tfrac{\varepsilon_{\text{cvg}}}{2(1-\rho)} v - \nu_{\hat{t}}^{\infty} < \tfrac{B\varepsilon_{\text{cvg}}}{2} - 2B\varepsilon_{\text{cvg}} = -\tfrac{3}{2}B\varepsilon_{\text{cvg}} \ , \end{aligned} \tag{B.2}$$

as $\nu_{\hat{t}}^{\infty} \geqslant \underline{\nu} > 2B\varepsilon_{\text{cvg}}$. Let us now consider $t' \in \mathcal{T}_n^{\circ(K)} \cup \overline{\mathcal{T}}_n^{(K)}$, then:

$$\text{(B.3)} \quad \nu_{t'}^{(K)} = y_{n,t'}^{(K)} - x_{n,t'}^{(K)} \geqslant y_{n,t'}^{(K)} - y_{n,t'}^{(K)} - \lambda_n^{(K)} > -\tfrac{\varepsilon_{\text{cvg}}}{2} + \tfrac{3}{2}B\varepsilon_{\text{cvg}} > B\varepsilon_{\text{cvg}} + \tfrac{\varepsilon_{\text{cvg}}}{2}(B-1) \geqslant B\varepsilon_{\text{cvg}} \ ,$$

which shows that $t' \in \mathcal{T}_0^{(K)} = \mathcal{T}_0^{\infty}$ and thus $\mathcal{T}_n^{\circ(K)} \cup \overline{\mathcal{T}}_n^{(K)} \subset \mathcal{T}_0$. Then, the same sequence of inequalities as (A.5, A.6, A.7) applied to $\boldsymbol{y}^{(K-1)}$ gives a contradiction to $n \notin \mathcal{N}_0$. ∎

Proof of Item (ii). The proof of Item (ii) is symmetric to the one of Item (i): if we suppose that there exists $n \in \mathcal{N}_0$ and $\hat{t} \notin \mathcal{T}_0$ such that $x_{n,\hat{t}}^{(K)} > \underline{x}_{n,\hat{t}}$, we obtain, symmetrically to (B.2), that $\lambda_n^{(K)} \geqslant -\frac{\varepsilon_{\text{cvg}}}{2(1-\rho)}$. Then, considering $t' \in \underline{\mathcal{T}}_n^{(K)} \cup \mathcal{T}_n^{\circ(K)}$, we show, symmetrically to (B.3), that $\nu_{t'}^{(K)} < B\varepsilon_{\text{cvg}}$ i.e. $t' \notin \mathcal{T}_0$ and thus $\underline{\mathcal{T}}_n^{(K)} \cup \mathcal{T}_n^{\circ(K)} \subset \mathcal{T}_0^c$. We conclude by obtaining a contradiction to $n \in \mathcal{N}_0$ by the same sequence of inequalities as (A.8, A.9, A.10). ∎

References.

[1] Y. P. Aneja and K. P. Nair, *Bicriteria transportation problem*, Management Science, 25 (1979), pp. 73–78.

[2] M. F. Anjos, A. Lodi, and M. Tanneau, *A decentralized framework for the optimal coordination of distributed energy resources*, IEEE Trans. Pow. Sys., 34 (2018), pp. 349–359.

[3] H. H. Bauschke and J. M. Borwein, *On the convergence of Von Neumann's alternating projection algorithm for two sets*, Set-Valued Analysis, 1 (1993), pp. 185–212.

[4] H. H. Bauschke, J. Chen, and X. Wang, *A Bregman projection method for approximating fixed points of quasi-Bregman nonexpansive mappings*, Applicable Analysis, 94 (2015), pp. 75–84.

[5] J. F. Benders, *Partitioning procedures for solving mixed-variables programming problems*, Numerische mathematik, 4 (1962), pp. 238–252.

[6] D. P. Bertsekas, *Nonlinear programming*, Athena Scientific, 1999.

[7] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and distributed computation: numerical methods*, vol. 23, Prentice hall Englewood Cliffs, NJ, 1989.

[8] E. D. Bolker, *Transportation polytopes*, Journal of Combinatorial Theory, Series B, 13 (1972), pp. 251–262.

[9] J. M. Borwein, G. Li, and L. Yao, *Analysis of the convergence rate for the cyclic projection algorithm applied to basic semialgebraic convex sets*, SIAM J. Optim., 24 (2014), pp. 498–527.

[10] G. Cohen and D. L. Zhu, *Decomposition and coordination methods in large scale optimization problems: The nondifferentiable case and the use of augmented lagrangians*, Adv. in Large Scale Systems 1, (1984), pp. 203–266.

[11] W. J. Cook, W. Cunningham, W. Pulleyblank, and A. Schrijver, *Combinatorial optimization*, Springer, 2009.

[12] W. Deng, M.-J. Lai, Z. Peng, and W. Yin, *Parallel multi-block admm with o (1/k) convergence*, Journal of Scientific Computing, 71 (2017), pp. 712–736.

[13] R. Glowinski and A. Marroco, *Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité d'une classe de problèmes de dirichlet non linéaires*, ESAIM, 9 (1975), pp. 41–76.

[14] L. Gubin, B. Polyak, and E. Raik, *The method of projections for finding the common point of convex sets*, USSR Comput. Math. & Math. Phys., 7 (1967), pp. 1 – 24.
[15] J. He, L. Cai, P. Cheng, J. Pan, and L. Shi, *Consensus-based data-privacy preserving data aggregation*, IEEE Transactions on Automatic Control, (2019), pp. 1–1.
[16] A. J. Hoffman, *Some recent applications of the theory of linear inequalities to extremal combinatorial analysis*, in Proc. of Symposia on Applied Mathematics, 1960, pp. 113–127.
[17] B. A. Huberman, E. Adar, and L. R. Fine, *Valuating privacy*, IEEE security & privacy, 3 (2005), pp. 22–25.
[18] P. Jacquot, O. Beaude, S. Gaubert, and N. Oudjane, *Analysis and implementation of an hourly billing mechanism for demand response management*, IEEE Trans. Smart Grid, 10 (2019), pp. 4265–4278.
[19] P. Jacquot, O. Beaude, P.Benchimol, S. Gaubert, and N. Oudjane, *A privacy-preserving dis-aggregation algorithm for non-intrusive management of flexible energy*, in Decision and Control (CDC), 2019 IEEE 58th Annual Conference on, IEEE, 2019.
[20] G. Jagannathan, K. Pillaipakkamnatt, and R. N. Wright, *A new privacy-preserving distributed k-clustering algorithm*, in Proc. of the 2006 SIAM Int. Conf. on Data Mining, SIAM, 2006, pp. 494–498.
[21] F. Katiraei, R. Iravani, N. Hatziargyriou, and A. Dimeas, *Microgrids management*, IEEE power and energy magazine, 6 (2008).
[22] K. K. Lai, K. Lam, and W. K. Chan, *Shipping container logistics and allocation*, J. Oper. Res. Soc., 46 (1995), pp. 687–697.
[23] C. Lemaréchal, A. Nemirovskii, and Y. Nesterov, *New variants of bundle methods*, Math. Program., 69 (1995), pp. 111–147.
[24] P.-Y. R. Ma et al., *A task allocation model for distributed computing systems*, IEEE Trans. Computers, 100 (1982), pp. 41–47.
[25] F. L. Müller, J. Szabó, O. Sundström, and J. Lygeros, *Aggregation and disaggregation of energetic flexibility from distributed energy resources*, IEEE Trans. Smart Grid, (2017).
[26] J. Munkres, *Algorithms for the assignment and transportation problems*, SIAM J. App. Math., 5 (1957), pp. 32–38.
[27] R. Nishihara, S. Jegelka, and M. I. Jordan, *On the convergence rate of decomposable submodular function minimization*, in NIPS, 2014, pp. 640–648.
[28] D. P. Palomar and M. Chiang, *A tutorial on decomposition methods for network utility maximization*, IEEE J. Sel. Areas Commun., 24 (2006), pp. 1439–1451.
[29] A. Rais and A. Viana, *Operations research in healthcare: a survey*, Int. Trans. Oper. Res., 18 (2011), pp. 1–31.
[30] K. Seong, M. Mohseni, and J. M. Cioffi, *Optimal resource allocation for ofdma downlink systems*, in Information Theory, 2006 IEEE Int. Sym., IEEE, 2006, pp. 1394–1398.
[31] J. Von Neumann, *Functional operators: Measures and integrals*, vol. 1, Princeton University Press, 1950.
[32] L. Xiao and S. Boyd, *Optimal scaling of a gradient method for distributed resource allocation*, J. Optim. Theory. Appl., 129 (2006), pp. 469–488.
[33] L. Xiao, M. Johansson, and S. P. Boyd, *Simultaneous routing and resource allocation via dual decomposition*, IEEE Trans. Comm., 52 (2004), pp. 1136–1144.
[34] A. C. Yao, *How to generate and exchange secrets*, in 27th Annual Symp. Found. of Comp. Sci. (SFCS), 1986, pp. 162–167.
[35] H. Yu and M. J. Neely, *A simple parallel algorithm with an o(1/t) convergence rate for general convex programs*, SIAM J. Optim., 27 (2017), pp. 759–783.
[36] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, *Non-intrusive load monitoring approaches for disaggregated energy sensing: A survey*, Sensors, 12 (2012), pp. 16838–16866.
[37] M. Zulhasnine, C. Huang, and A. Srinivasan, *Efficient resource allocation for device-to-device communication underlaying lte network*, in WiMob, 2010 IEEE 6th Int. Conference, IEEE, 2010, pp. 368–375.