



Capacity acquisition for the single-item lot sizing problem under energy constraints

Christophe Rapine, Bernard Penz, Céline Gicquel, Ayse Akbalik

► To cite this version:

Christophe Rapine, Bernard Penz, Céline Gicquel, Ayse Akbalik. Capacity acquisition for the single-item lot sizing problem under energy constraints. *Omega*, 2018, 81, pp.112-122. 10.1016/j.omega.2017.10.004 . hal-01617045

HAL Id: hal-01617045

<https://hal.science/hal-01617045>

Submitted on 8 Jan 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Capacity acquisition for the single-item lot sizing problem under energy constraints

Christophe Rapine¹, Bernard Penz², Céline Gicquel³, Ayse Akbalik¹

¹ Université de Lorraine, Ile du Saulcy, Metz, F-57012, France

² Univ. Grenoble Alpes, CNRS, Grenoble INP, G-SCOP, F-38000 Grenoble, France

³ Université de Lorraine, Ile du Saulcy, Metz, F-57012, France

Abstract

We study a single-item lot sizing problem integrated with some energy constraints, called *energy-LSP* in the rest of the paper. We consider a production system composed of identical and capacitated machines in parallel. One has to decide how many machines to start and how much to produce to serve the demand and/or to replenish the inventory. In addition to the production capacity limit in each period, induced by the machines on, we have a limit on the energy that can be consumed by the start-ups of the machines and the production of units. We first provide a mixed integer programming formulation for the problem in the general case. We then develop efficient polynomial time algorithms running in $O(T \log T)$, with T the length of the planning horizon, for several sub-problems: (i) *energy-LSP* where all energy parameters are set to 0, (ii) *energy-LSP* where only the machine start-ups consume energy, (iii) *energy-LSP* where only the production of units consumes energy. We show that, in all these special cases, *energy-LSP* can be seen as an extension of the integrated capacity acquisition and lot sizing problem. This allows us, as by-product of our approach, to significantly improve the existing result proposed for the capacity acquisition problem without energy constraint nor subcontracting.

Keywords: Energy-efficient manufacturing ; lot sizing ; capacity acquisition ; energy constraints ; polynomial time algorithm.

1 Introduction

Due to the rarefaction of natural resources, the hardening of environmental public legislation and the growing customer awareness for green products, industrial companies are increasingly under pressure to reduce the environmental impact of their industrial operations. In particular, the availability and affordability of energy is becoming a critical parameter affecting the whole life cycle of an industrial product, including its production phase [Salonitis and Ball, 2013]. This translates into a growing attention of both researchers and practitioners on energy-efficient manufacturing.

Energy efficiency in manufacturing can be achieved by designing more eco-efficient products and by carrying out technological improvements of the production processes, but this usually requires large long-term financial investments. In

contrast, energy-efficient production planning, by adjusting managerial parameters of the existing production processes, can be seen as a relatively inexpensive organizational measure whose benefit can be visible in the short-term (see [Biel and Glock, 2016] and [Gahm et al., 2016]). Energy-efficient production planning is defined by Biel and Glock [Biel and Glock, 2016] as the computation of production plans which takes into account not only economic objectives (such as production costs or completion time minimization) but also energy-related objectives (e.g. energy cost minimization) or energy-related constraints (e.g. compliance with a maximum contracted consumption). It is particularly relevant for energy-intensive production processes as can be seen from the numerous case studies reported in the literature such as tile curing [Özdamar and Birbil, 1999], melting shops and hot rolling mills for steel production [Castro et al., 2013, Tang et al., 2012], pulp and paper mills [Santos and Almada-Lobo, 2012, Waldemarsson et al., 2013] or thermal cracking for ethylene production [Zhao et al., 2016]. Note that energy efficiency is also becoming a crucial element in the short-term planning of server farms activities [Chen et al., 2005, Mitrani, 2013].

Biel and Glock [Biel and Glock, 2016] provide a recent literature review on decision support models for energy-efficient production planning. They show that most papers focus either on long-term aggregate planning or on short-term scheduling problems but that energy-efficient dynamic lot-sizing problems have been scarcely studied. Moreover, most papers make use of energy-related objectives aiming at minimizing either the energy cost or the energy consumption but implicitly assume that the energy is available in an unlimited quantity. However, due to the development of renewable energy sources and the difficulty to store electric energy on a large scale, energy providers have to increasingly rely on measures such as demand side management programs to shift demand for energy to off-peak periods and balance the electric grid [Paulus and Borggreffe, 2011, Schultz et al., 2015, Shoreh et al., 2016]. For the manufacturers, this means in practice that electric energy cannot be considered anymore as a limitless resource and that they should seek not only to minimize their energy consumption but also to synchronize it with the available energy supply. Note that production scheduling under power availability restrictions has been investigated among others in [Artigues et al., 2013, Bruzzone et al., 2012, Mitra et al., 2012]. In this work, we consider production planning at a more aggregate decision level. We namely study a dynamic lot-sizing problem and introduce an energy constraint setting a limit on the total quantity of energy consumed in each time period rather than on the maximum instantaneous power demand.

More precisely, we investigate the single-item multi-resource Continuous Setup Lot sizing Problem (CSLP) with energy-related constraints. The CSLP is a small-bucket lot-sizing problem introduced by Karmakar and Schrage [Karmakar and Schrage, 1985]. In the CSLP, a production lot on a machine consists of a set of consecutive planning periods in which the item is produced, and start-up costs are incurred at the first period of a production lot to take into account the costs related to start-up operations. Note that the term "Continuous Setup" refers here to the fact that the quantity produced on a machine in a given period can take any value between 0 and the machine capacity. This contrasts with another small-bucket

lot-sizing problem, the Discrete Lot-sizing and Scheduling Problem (DLSP), where an all-or-nothing production policy enforcing that, in a given period, a machine is either idle or producing at full capacity is assumed. We consider a multi-resource extension of the CSLP involving a set of identical parallel capacitated machines. The optimization problem thus consists in deciding, for each period over a finite horizon, the number of machines to start-up, the quantity of the item to be produced and the quantity to be kept in inventory so as to satisfy the customer demand. In each period, these decisions have to satisfy two capacity constraints: one coming from the limited capacity of the production resources and one coming from the limited amount of energy. The objective is to minimize the start-up and inventory holding costs over the planning horizon. Similarly to [Karmarkar and Schrage, 1985], we assume null setup costs, i.e. the costs to maintain the resource in the producing state are assumed to be negligible. Note that, with positive setup costs, the same problem becomes NP-hard since it is a special case of the Capacitated Lot Sizing Problem (CLSP) with time-dependent capacities, which was shown to be NP-complete by Florian *et al.* [Florian et al., 1980] and Bitran and Yanasse [Bitran and Yanasse, 1982].

The main contributions of the paper are threefold. First, we extend the CSLP in order to explicitly take into account energy efficiency via a set of constraints limiting the energy consumption in each period. To the best of our knowledge, this is the first attempt at introducing energy-related constraints in a small-bucket lot-sizing model. Second, we propose an original combinatorial algorithm to solve very efficiently the lot-sizing problem with energy constraints under some rather general assumptions on the energy parameters and cost functions, when either the start-up or the unit production energy consumption is negligible. Third, as a by-product of our approach, we improve the complexity of the capacitated lot-sizing problem with capacity acquisition. For this problem, Atamtürk and Hochbaum [Atamtürk and Hochbaum, 2001] propose an $O(T^3)$ algorithm, allowing subcontracting, where T denotes the length of the planning horizon. Without subcontracting, we are able to solve the capacity acquisition problem in time $O(T \log T)$ for concave acquisition cost function.

This paper is organized as follows. A literature review is presented in Section 2. In Section 3, the lot-sizing problem with the general energy constraint is formulated as a mixed-integer program. In Section 4, an algorithm in $O(T^2)$ is proposed for a pure capacity acquisition problem. Although the energy parameters are not taken into account in this first model, it allows us to introduce the main concepts in a simple setting. An improvement of this algorithm is given in Appendix, with a time complexity in $O(T \log T)$. In Section 5, we extend our algorithm to deal with energy constraints. We consider two special cases of *energy-LSP*, where only the start-up of machines consumes energy, on one hand, and only production of units consumes energy, on the other hand. We show that in both cases we can adapt our algorithm, preserving its low time complexity of $O(T \log T)$. In Section 6, other extensions are given for more general start-up cost functions, namely convex or piecewise function, with a mix of convex and concave pieces. The overall time complexity of the algorithm is only slightly increased for these latter cases. The paper is concluded with some perspectives

in Section 7.

2 Literature review

The classical lot-sizing problem aims at deciding, for a finite horizon subdivided into discrete periods, how much to produce in each period in order to satisfy a deterministic time-varying demand, with the objective to reach the best possible trade-off between production costs and inventory holding costs. We refer the reader to the seminal paper of Wagner and Whitin [Wagner and Whitin, 1958] and to the survey of Brahimi *et al.* [Brahimi, 2007] for a general introduction to this field. This work is mostly related to a small-bucket variant of the lot-sizing problem known as the Continuous Setup Lot-sizing Problem (CSLP). Industrial applications of the CSLP to plan production on a set of unrelated parallel machines can be found in a variety of environments such as tile production [de Matta and Guignard, 1994], injection molding for healthcare products [Dastidar and Nagi, 2005], packaging lines for dairy products [Marinelli et al., 2007] or glass melting furnaces [Almada-Lobo et al., 2010]. In case of identical parallel machines, symmetry issues can arise during the solution phase due to the existence of many alternative optimal solutions obtained by renumbering the machines (see e.g. [Jans, 2009]). In order to avoid that, in small bucket models, it is useful to use an aggregate representation of the machines and to introduce integer rather than binary start-up variables. This has been exploited in a variety of real-world cases such as [Jans and Degraeve, 2004, Lasdon and Terjung, 1971] for tire manufacturing, [Sawik, 2009] for electronic devices production and [Kaczmarczyk, 2011] for mobile phone production. Note that the modeling of all the industrial applications mentioned in this paragraph does not include fixed setup costs to be incurred to keep a machine producing a given item once it has been started up for it, but only start-up costs to be incurred each time the setup state of the machine is changed from one item to the other.

As can be seen e.g. in the literature review provided by Jans and Degraeve [Jans and Degraeve, 2008], the research on lot-sizing has mainly focused on extending the basic problem in order to better model operational aspects or to integrate it as a substructure into larger tactical planning problems. However, it seems that only a limited effort has been done towards taking into account the growing pressure to establish sustainability in manufacturing and to comply with stricter environmental standards. In particular, as shown by the literature review provided by Biel and Glock [Biel and Glock, 2016], research on integrating energy efficiency in the lot-sizing problem appears to be only in its early stages.

A first situation in which energy should be explicitly taken into account is found when the energy needed by the production system is not only obtained from an external energy provider but is also produced internally, for instance by converting by-products of the process into steam or electrical energy. In this case, one has to simultaneously plan the activities of the pro-

duction system with the activities of the energy system in order to coordinate them. Such problems are considered among others by Santos and Almada-Lobo [Santos and Almada-Lobo, 2012], Waldemarsson *et al.* [Waldemarsson et al., 2013] and Zhao *et al.* [Zhao et al., 2016]. However, most often, the energy needed for production is obtained entirely from an external provider. In this case, a possible way to integrate energy in the lot-sizing problem consists in taking explicitly into account its cost in the objective function: see e.g. Özdamar and Birbil [Özdamar and Birbil, 1999], Uzel [Uzel, 2004], Heck and Schmidt [Heck and Schmidt, 2010] and Tang *et al.* [Tang et al., 2012]. Note that these approaches all assume that the energy is available in an unlimited quantity. We found a single approach integrating energy-related constraints in a lot-sizing problem. Masmoudi [Masmoudi et al., 2017] consider a single-item multi-stage multi-resource lot-sizing problem and introduce constraints setting a strict limit on the maximum instantaneous power demand used in each period. They propose two heuristic solution approaches for their problem and compare their performance to the one of the mathematical programming solver CPLEX. In contrast, we focus on providing structural results of the optimal solutions and on developing exact polynomial time algorithms.

A simplified version of the *energy-LSP* problem is very similar to the single-item lot sizing problem with capacity acquisition and subcontracting studied by Atamtürk and Hochbaum [Atamtürk and Hochbaum, 2001]. The authors propose an exact algorithm running in $O(T^3)$ time for the case with nonspeculative linear production costs, nonspeculative linear subcontracting costs and concave capacity acquisition costs. We show (see Section 4 and Appendix) that the same problem can be solved with a reduced time complexity of $O(T \log T)$ in case of infinite subcontracting costs. Note that Li and Meissner [Li and Meissner, 2011] also study an integrated lot sizing and capacity acquisition problem in which subcontracting is not allowed but, in contrast to our work, they assume time-independent fixed-charge production costs and focus on developing a heuristic polynomial time algorithm rather than an exact polynomial time algorithm. In the literature one can find many papers dealing with capacity planning in supply chains, but most of them focus on strategic decisions such as integrated facility location and capacity planning. For more details on the mathematical models proposed in capacity planning in manufacturing, see the review from Martínez-Costa *et al.* [Martínez-Costa et al., 2014]. To the best of our knowledge, in the literature, such a capacity acquisition problem has not been yet studied under energy constraints.

3 Problem formulation

The *energy-LSP* consists in planning the production of a single-item, assuming a deterministic demand (d_t in each period t) over a finite horizon of T periods. The production takes place on parallel and identical machines, each one with a maximum capacity U . The energy available in period t , E_t , is used both for the production of units and for the start-up of the machines. We denote p_t

the amount of energy needed to produce one unit of product in period t and w_t the amount of energy needed to start-up one machine in period t . The cost to start m machines in a period is $f(m)$. The start-up cost functions $f()$ are assumed to be concave and stationary. The concavity of $f()$ allows us to capture in our model economies of scale that can be encountered in industry. We also assume that $f(m)$ can be evaluated in constant time $O(1)$ for any value of m . The production and inventory holding costs are linear: we denote c_t the unit production cost and h_t the unit holding cost for storing one unit between periods t and $t + 1$. Notice that we have no classical setup cost (paid in each period with a positive production amount), but only start-up costs to be incurred when one or several machines are switched on in compliance with small bucket multi-resource CSLP we consider. We summarize below the different parameters of the problem:

c_t	unit production cost in period t
h_t	unit holding cost from period t to $t + 1$
$f(m)$	cost of turning on m machines
p_t	unit energy consumption in period t
w_t	energy consumption to start one machine in period t
E_t	amount of energy available in period t
d_t	demand in period t
U	capacity of a machine

The decision variables are given as follows:

m_t	the number of machines on in period t
m_t^+	the number of machines started-up at the beginning of period t
s_t	stock level between t and $t + 1$
x_t	production quantity in t

The *energy-LSP* is formulated as the following mixed-integer program:

$$\begin{aligned}
\min \quad & (\sum_{t=1}^T f(m_t^+) + \sum_{t=1}^T c_t x_t + \sum_{t=1}^T h_t s_t) \\
s.t. \quad & s_{t-1} + x_t = s_t + d_t & \forall t \in \{1..T\} \quad (1) \\
& x_t \leq U m_t & \forall t \in \{1..T\} \quad (2) \\
& p_t x_t + w_t m_t^+ \leq E_t & \forall t \in \{1..T\} \quad (3) \\
& m_t^+ \geq m_t - m_{t-1} & \forall t \in \{1..T\} \quad (4) \\
& s_t \geq 0, x_t \geq 0, m_t \in \mathbb{Z}^+, m_t^+ \in \mathbb{Z}^+ \in \mathbb{Z}^+ & \forall t \in \{1..T\} \quad (5)
\end{aligned}$$

The objective function aims at minimizing the overall cost of machine start-up, production and inventory. Constraints (1) are the balancing constraints linking inventory, ordering quantities and demand over T periods. Constraints (2) ensure that, in each period, the production quantity stays below the total production capacity which depends on the number of machines on. Constraints (3) guarantee that the energy consumed by both production and starting-up activities is limited by the available energy in each period. Constraints (4) link

the number of machines started-up at the beginning of period t to the number of machines on in periods t and $t - 1$, thus ensuring that the start-up costs $f(m_t^+)$ are correctly incurred in the objective function. The remaining constraints (5) define the feasibility domain of each decision variable.

In the following of the paper we assume that production costs follow Wagner-Whitin cost structure, also called non-speculative motives. It implies that for any period t of the time horizon, we have $c_t + h_t \geq c_{t+1}$. Notice that, without loss of generality, we can consider that the holding costs are null, by substituting unit production costs c_t with $\tilde{c}_t = c_t + \sum_{u=t}^T h_u$, see for instance [Pochet and Wolsey, 2006]. The non-speculative production cost assumption implies that $\tilde{c}_t \geq \tilde{c}_{t+1}$ holds for all periods, that is, modified unit production costs are non-increasing over time. Finally, recall that we focus on the case of stationary and concave start-up cost function $f(\cdot)$. In the next section, we relax the energy constraints (3) of the problem to focus on what we call the *pure capacity acquisition problem*. We develop a very efficient algorithm for this case in Section 4, and we extend it in Section 5 to two cases with only one non-null energy parameter at a time (either start-up or production consumption).

4 Pure capacity acquisition problem

We begin with the study of a simplified version of the *energy-LSP* in which all energy consumption parameters are assumed to be null, that is, $w_t = p_t = 0$ for all periods t , and propose an algorithm in $O(T^2)$ for this special case. This algorithm will be generalized to deal with positive energy parameters in Section 5. In Appendix, we also show that it can be implemented to run in time complexity $O(T \log T)$.

4.1 Preliminaries

When all energy consumption parameters are null, *energy-LSP* reduces to a pure lot-sizing problem with capacity acquisition. Namely, the machines on do not incur any cost in a given period if they do not produce at that period. Thus, it is dominant to start-up all the necessary machines at the beginning of the planning horizon and to keep them on till the end. As a consequence, one simply has to decide the optimal value of m_1^+ and set $m_t = m_1^+, \forall t, m_t^+ = 0, \forall t = 2 \dots T$. The main decision thus consists in deciding how many machines to start-up at the beginning of the planning horizon, i.e. in determining the capacity to be acquired at the beginning of the planning horizon.

The *energy-LSP* problem with $w = p = 0$ is a special case of the problem investigated by Atamtürk and Hochbaum [Atamtürk and Hochbaum, 2001] in which there is no subcontracting option. Atamtürk and Hochbaum [Atamtürk and Hochbaum, 2001] consider a continuous capacity model in which capacity can be acquired at any non-negative level. They show that, under the assumptions of non-speculative linear production and subcontracting costs and of concave capacity acquisition

costs, the problem is polynomially solvable in time $O(T^3)$. In this section, we show that there exists an efficient algorithm in $O(T^2)$ to solve the special case with no subcontracting option under the same assumptions. The time complexity of our algorithm can even be improved into an $O(T \log T)$ algorithm, using more sophisticated data structures (see Appendix).

We thus consider in what follows *energy-LSP* with $w = p = 0$. The problem is a capacitated lot-sizing problem: in each period, the amount produced cannot exceed the production capacity of the system. However, the production capacity $C = Um_1^+$ is not a given parameter but is itself a decision variable of the problem. It means that we have to simultaneously determine the capacity C to be acquired (i.e. the number of machines m_1^+ to start at the beginning of the planning horizon) and the production plan which should be feasible with respect to this installed capacity C . For the sake of simplicity, we consider in the description of the proposed algorithm a continuous capacity C and concave acquisition cost $f(C)$. As explained at the end of Subsection 4.2, our results can be applied without any difficulty to the discrete capacity problem in which the value of C would be restricted to the set of integer multiples of U . Note that the problem may become infeasible if the production capacity C chosen is too small. We denote by C_{\min} the smallest capacity value for which the problem is feasible. Clearly, there exists a feasible planning for any given capacity $C \geq C_{\min}$. Though the value of C_{\min} can be easily determined, the algorithm we propose does not need to compute it explicitly.

We now provide some insights of the ideas underlying the proposed algorithm.

- Since there are no setup costs and production costs are assumed to be non-speculative, one should seek to plan production using a lot-for-lot strategy in order to reduce the inventory holding costs. However, as the production level is limited by the installed capacity C (which makes part of the decision), the demands exceeding C should be anticipated and smoothed to the previous periods. The optimal production plans have a particular structure: in all the periods of a subplan, except the first one, production occurs at full capacity, that is, $x_t = C$. For a given value of the installed capacity C , determining the optimal production plan can be done in linear time $O(T)$, see Atamtürk and Hochbaum [Atamtürk and Hochbaum, 2001].
- Searching for the optimal solution can thus be reduced to the search of the optimal value of the installed capacity C . In their approach, Atamtürk and Hochbaum studied the polyhedral structure of the problem, and established that the number of extreme points can be bounded by $O(T^2)$. Since each extreme point can be evaluated in linear time, it leads to an $O(T^3)$ algorithm. In our approach, we show that among the $O(T^2)$ extreme points, only a small number needs to be considered to find an optimal solution. Specifically, we focus on the function $OPT(C)$ representing the cost of an optimal solution for an installed capacity C , and show that $OPT(C)$ is a

piecewise concave function comprising at most T breakpoints. Hence, the number of values of C to be considered for finding the minimum value of $OPT(C)$ and the corresponding optimal solution is bounded by T , leading to an $O(T^2)$ time algorithm.

4.2 Structure of an optimal solution

Clearly, a solution is entirely specified by its vector $x = (x_1, \dots, x_T)$ of production. For the ease of the presentation, we also use the associated vector s of inventory levels, where

$s_t \equiv (x_1 + \dots + x_t) - (d_1 + \dots + d_t)$ represents the amount of products in stock at the end of period t . We focus on solutions with a particular structure. For a given capacity C , we call a period t a *full production period* if $x_t = C$. A period t with a null entering stock ($s_{t-1} = 0$) is called in the literature a *regeneration period*. We introduce the following definition :

Definition 1 *We say that a solution x is no-slack if and only if $s_{t-1}(C - x_t) = 0$ holds for each period $t = 1, \dots, T$. That is, each period of the planning horizon is either a full production period or a regeneration period.*

In addition, we say that solution x is dominant if it is feasible and $s_T = 0$, that is, its final inventory is null.

Such a no-slack solution x defines in particular a subset $\mathcal{B} = \{t | x_t = C\}$ of periods, corresponding to the periods where the capacity constraint is saturated. We will establish that conversely, each subset \mathcal{B} defines a no-slack solution, and that only a small number of subsets need to be considered to find an optimal solution.

For a set \mathcal{B} of periods, let us define $\mathcal{A} = \{t | t \notin \mathcal{B} \text{ and } (t+1) \in \mathcal{B}\}$. Set \mathcal{A} is thus the set of periods immediately preceding a sequence of saturated periods. If a period t belongs to \mathcal{A} , by definition period $(t+1)$ belongs to set \mathcal{B} . Hence we can define $v(t)$ as the largest index such that $t+1, \dots, v(t)$ belongs to \mathcal{B} . We call set $\{t, \dots, v(t)\}$ the *block* associated with a period $t \in \mathcal{A}$. We denote by $D_{i,j} \equiv \sum_{t=i}^j d_t$ the demands to satisfy over the periods i upto j . For a given set \mathcal{B} of periods, we define the associated no-slack solution $x(\mathcal{B}, C)$ as follows:

$$x_t(\mathcal{B}, C) = \begin{cases} C & \text{if } t \in \mathcal{B} \\ D_{t, v(t)} - (v(t) - t)C & \text{if } t \in \mathcal{A} \\ d_t & \text{otherwise} \end{cases}$$

We illustrate the definition of a no-slack solution on the following example. We consider a planning horizon of 6 periods with demand $d = (10, 12, 6, 20, 8, 14)$ to satisfy. The no-slack solution $x(\mathcal{B}, C)$ associated with capacity $C = 13.5$ and set $\mathcal{B} = \{4, 6\}$ is represented in Figure 1a. The full production periods (set \mathcal{B}) are represented in red and periods in set \mathcal{A} are represented in grey. The network flow representation of this solution is given in Figure 1b, where the saturated arcs are represented by bold red arrows. In accordance with the definition of

$x(\mathcal{B}, C)$, there is no entering stock in periods $\{1, 2, 3, 5\}$. It is easy to see that this solution is feasible and has a null ending stock level : It is also a dominant solution.

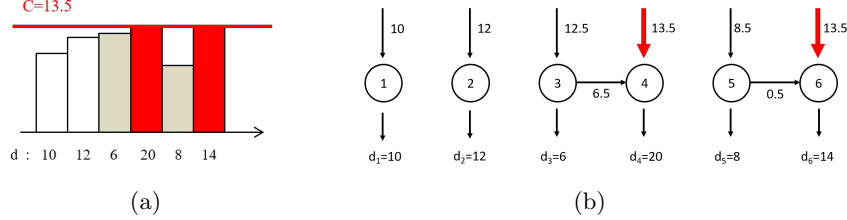


Figure 1: (a) The no-slack solution $x(\mathcal{B}, C)$ associated with capacity $C = 13.5$ and set $\mathcal{B} = \{4, 6\}$ and (b) Its network flow representation.

Notice that solution $x(\mathcal{B}, C)$ may be infeasible, since the production in a period $t \notin \mathcal{B}$ may exceed capacity C . It may also be feasible but not dominant if it produces too much, for instance if C is a large value and set $\mathcal{B} = \{1, \dots, T\}$. However, if $x(\mathcal{B}, C)$ is dominant, that is, is feasible and produces exactly the demand, we have the following property:

Property 1 *If solution $x(\mathcal{B}, C)$ is a dominant no-slack solution, then it is an optimal solution for the problem with capacity C .*

Proof. Recall that \tilde{c}_t represents the modified unit production cost after the addition of the linear holding cost. Since unit production costs \tilde{c}_t are non-increasing, a solution producing “at latest” is clearly optimal. For short, let us denote simply by x the solution $x(\mathcal{B}, C)$ and let $X_t = x_1 + \dots + x_{t-1}$ denote its cumulative production up to period $t - 1$. It is sufficient to prove that for any feasible solution z , the cumulative production Z_t is greater than or equal to X_t for all periods. For a period t that does not belong to set \mathcal{B} , this is immediate, since $s_{t-1} = 0$ implies that $X_t = D_{1,t-1}$. For a period $t \in \mathcal{B}$, let v be the last period of its block. By construction, if $v < T$, we have $s_v = 0$ since $(v + 1)$ does not belong to set \mathcal{B} . Notice that $s_v = 0$ also holds if $v = T$, since we assume that x is dominant. Hence, we have $X_t + (v - t + 1)C = D_{1,v}$. Flow conservation imposes for any feasible solution that $D_{1,v} \leq Z_t + z_t + \dots + z_v \leq Z_t + (v - t + 1)C$. The result follows. \square

Recall that we denote by $\text{OPT}(C)$ the cost of an optimal solution relatively to an installed capacity C . For a set \mathcal{B} of periods, Property 1 shows that $\text{OPT}(C)$ is realized by solution $x(\mathcal{B}, C)$ if it is dominant. Next property will allow us to establish that $\text{OPT}(C)$ is a piecewise concave function:

Property 2 *Let \mathcal{B} be a set of periods and $[a, b]$ a real interval. If for all capacities C in $[a, b]$ the solution $x(\mathcal{B}, C)$ is dominant, then $\text{OPT}(C)$ is concave with respect to C over $[a, b]$. As a consequence, its minimum over $[a, b]$ is reached at one of the extremities of the interval.*

Proof. Consider C and C' two capacities of interval $[a, b]$. Let $x = x(\mathcal{B}, C)$ and $x' = x(\mathcal{B}, C')$ be the solutions associated to C and C' , respectively. Notice that x and x' differ only by their production in periods of set $\mathcal{A} \cup \mathcal{B}$. Precisely, we have:

$$\begin{aligned} \text{OPT}(C') - \text{OPT}(C) &= f(C') - f(C) + \sum_{t \in \mathcal{A}} \tilde{c}_t(x'_t - x_t) + \sum_{t \in \mathcal{B}} \tilde{c}_t(C' - C) \\ &= f(C') - f(C) + \sum_{t \in \mathcal{A}} \left(\tilde{c}_t(v(t) - t)(C - C') + \sum_{u=t+1}^{v(t)} \tilde{c}_u(C' - C) \right) \\ &= f(C') - f(C) - (C' - C) \sum_{t \in \mathcal{A}} \sum_{u=t+1}^{v(t)} (\tilde{c}_t - \tilde{c}_u) \end{aligned}$$

The term $\sum_{t \in \mathcal{A}} \sum_{u=t+1}^{v(t)} (\tilde{c}_t - \tilde{c}_u)$ is fixed for a given set \mathcal{B} , which means that the modified production costs vary linearly with C over $[a, b]$. Writing the expression $\text{OPT}(C') - \text{OPT}(C)$ for the particular value $C = a$, we obtain :

$$\text{OPT}(C') = \text{OPT}(a) - f(a) + f(C') - (C' - a) \sum_{t \in \mathcal{A}} \sum_{u=t+1}^{v(t)} (\tilde{c}_t - \tilde{c}_u)$$

The quantity $\text{OPT}(a) - f(a)$ being fixed, OPT is the sum of a concave function and of a linear function and thus is itself a concave function. The result follows. \square

Next section describes how the capacity breakpoints of $\text{OPT}(C)$ can be found very efficiently.

4.3 Description of the algorithm

Our algorithm constructs a series of capacities $C^{(1)} \geq C^{(2)} \geq \dots \geq C^{(k)}$ and a series of subsets $\mathcal{B}^{(1)} \subset \mathcal{B}^{(2)} \subset \dots \subset \mathcal{B}^{(k)}$ such that the no-slack solution $x(\mathcal{B}^{(i)}, C^{(i)})$ is dominant for each capacity value in $[C^{(i+1)}, C^{(i)}]$. The algorithm stops when the first period is a full production period, that is, $1 \in \mathcal{B}^{(k)}$. We claim that the capacity $C^{(k)}$ is then equal to the smallest capacity value, C_{\min} , such that the problem is feasible. Indeed, a necessary and sufficient condition for the problem with a capacity C to be feasible is that $D_{1,t} \leq tC$ for all periods $t = 1, \dots, T$. If a subset \mathcal{B} contains the first period and solution $x(\mathcal{B}, C)$ is feasible, then we have $D_{1,v} = vC$ for $v = v(1)$. This implies that $C = C_{\min}$. As a consequence, any capacity C admitting a feasible plan belongs to (at least) one of the intervals $[C^{(i+1)}, C^{(i)}]$. We will demonstrate that at most T different capacities $C^{(i)}$ need to be considered, that is, k is bounded by T .

Let us define capacity $C^{(1)} = \max_t \{d_t\}$ and $\mathcal{B}^{(1)} = \{t | d_t = C^{(1)}\}$. Clearly the no-slack solution associated with $\mathcal{B}^{(1)}$ is dominant, and thus realizes $\text{OPT}(C^{(1)})$. Consider now that we have determined a capacity $C^{(i)}$ and a subset $\mathcal{B}^{(i)}$ such that $x(\mathcal{B}^{(i)}, C^{(i)})$ is dominant, and that $1 \notin \mathcal{B}^{(i)}$ (otherwise the algorithm stops). With immediate notation, we denote by $\mathcal{A}^{(i)}$ the set of \mathcal{A} periods associated with $\mathcal{B}^{(i)}$. We want to determine the smallest value $C \leq C^{(i)}$ for which $x(\mathcal{B}^{(i)}, C)$ is dominant. Consider a capacity $C \leq C^{(i)}$. Using its definition,

solution $x(\mathcal{B}^{(i)}, C)$ is dominant if and only if the two following conditions hold:

$$\begin{aligned} (i) \quad & \forall t \notin \mathcal{A}^{(i)} \cup \mathcal{B}^{(i)}, \quad C \geq d_t \\ (ii) \quad & \forall t \in \mathcal{A}^{(i)}, \quad C \geq D_{t,v(t)} / |\{t, \dots, v(t)\}| \end{aligned}$$

We define

$$C^{(i+1)} = \max \left\{ \max_{t \notin \mathcal{A}^{(i)} \cup \mathcal{B}^{(i)}} d_t, \max_{t \in \mathcal{A}^{(i)}} \frac{D_{t,v(t)}}{|\{t, \dots, v(t)\}|} \right\} \quad (6)$$

Notice that by construction, $x(\mathcal{B}^{(i)}, C^{(i+1)})$ is dominant. However, in this solution, at least one period is both a full production period and a regeneration period. That is, there exists a full production period not belonging to $\mathcal{B}^{(i)}$. We define:

$$\mathcal{B}^{(i+1)} = \{t \mid x(\mathcal{B}^{(i)}, C^{(i+1)}) = C^{(i+1)}\}$$

By construction we have $\mathcal{B}^{(i)} \subset \mathcal{B}^{(i+1)}$. As a consequence, set $\mathcal{B}^{(i)}$ contains at least i periods, and thus the algorithm stops after at most T steps. Since each step can be performed in time $O(T)$, determining the $C^{(i)}$'s and the $\mathcal{B}^{(i)}$'s takes $O(T^2)$ time.

To output the optimal solution, we use Property 2. Since $\text{OPT}(C)$ is concave over each interval $[C^{(i+1)}, C^{(i)}]$, the optimal value is reached at one of the capacities $\{C^{(1)}, \dots, C^{(k)}\}$. Notice that determining the optimal cost $\text{OPT}(C^{(i)})$ for a given capacity can be performed in linear time $O(T)$. Hence, the algorithm runs in time complexity $O(T^2)$. We can state the following theorem:

Theorem 1 *An optimal solution OPT for the pure capacity acquisition problem can be found in time $O(T^2)$.*

In case of a discrete capacity problem, where C is restricted to be a multiple of U , we consider for each index $i = 1, \dots, k$ the capacities $\lfloor C^{(i)}/U \rfloor U$ and $\lceil C^{(i)}/U \rceil U$. It represents at most $2k$ different values. Again, due to Property 2, we can assert that the discrete capacity minimizing $\text{OPT}(C)$ belongs to this set. Evaluating all these $2k$ values to find the minimal value can be performed in time complexity $O(T^2)$.

In the next section we illustrate the idea of the algorithm on a simple example.

4.4 Illustrative example

We consider the same instance as in Figures 1a and 1b : we have 6 periods with demand $d = (10, 12, 6, 20, 8, 14)$ to satisfy. In the Figures 2 to 6 below, we have represented the optimal policy related with the current capacity C . We used different colors to distinguish periods belonging to \mathcal{A} or \mathcal{B} : A red rectangle corresponds to a period in \mathcal{B} (saturated periods), a grey rectangle corresponds to a period in \mathcal{A} . Other periods are represented by white rectangles. Notice that for these periods the production is equal to the demand. Finally, the red circles indicate the periods that are saturated but not yet in the current set \mathcal{B} . Note that those periods will enter set \mathcal{B} in the next step.

First step ($i = 1$)

The algorithm starts with an empty set \mathcal{B} and finds the first capacity $C^{(1)}$ to consider, which corresponds to the highest demand in period 4, here $C^{(1)} = \max_t \{d_t\} = 20$. By construction, period 4 is saturated (red circle in Figure 2) and it enters set \mathcal{B} . Thus, we have $\mathcal{B}^{(1)} = \{4\}$. Clearly $x(\mathcal{B}^{(1)}, C^{(1)})$ is dominant (hence optimal for $C^{(1)}$).

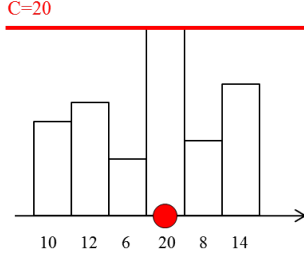


Figure 2: First step ($i=1$): Search of $C^{(1)}$ and $\mathcal{B}^{(1)}$

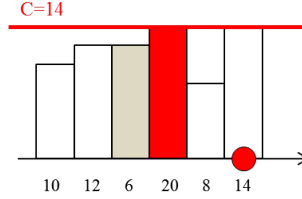


Figure 3: Second step ($i=2$): Search of $C^{(2)}$ and $\mathcal{B}^{(2)}$

Second step ($i = 2$)

We search now the lowest capacity $C^{(2)} \leq C^{(1)}$ for which solution $x(\mathcal{B}^{(1)}, C^{(2)})$ is dominant. We compute $C^{(2)}$ using Equation (6). Note that $\mathcal{A}^{(1)} = \{3\}$. We obtain:

$$C^{(2)} = \max \left\{ \max_{t \notin \{3\} \cup \{4\}} d_t, \frac{D_{3,4}}{4 - 3 + 1} \right\} = \max \left\{ 14, \frac{26}{2} \right\} = 14$$

This value corresponds to the demand of period 6, hence, the new capacity $C^{(2)}$ is obtained by hitting the second highest demand. See Figure 3. The new $\mathcal{B}^{(2)}$ is thus $\mathcal{B}^{(2)} = \mathcal{B}^{(1)} \cup \{6\} = \{4, 6\}$.

Third step ($i = 3$)

Again, using Equation (6), we find $C^{(3)} = \max \{12, \max \{\frac{26}{2}, \frac{22}{2}\}\} = 13$. This time, a period of set $\mathcal{A}^{(2)}$ becomes saturated, here, period 3 (red circle in Figure 4). Hence, $\mathcal{B}^{(3)} = \mathcal{B}^{(2)} \cup \{3\} = \{3, 4, 6\}$.

Fourth step ($i = 4$)

$C^{(4)} = \max \{12, \max \{\frac{38}{3}, \frac{22}{2}\}\} = 12.66$. Period 2 in set $\mathcal{A}^{(3)}$ becomes saturated (red circle in Figure 5). Thus, $\mathcal{B}^{(4)} = \mathcal{B}^{(3)} \cup \{2\} = \{2, 3, 4, 6\}$.

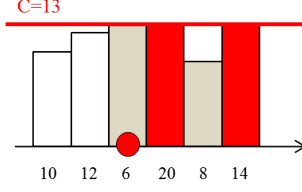


Figure 4: Third step ($i=3$): Search of $C^{(3)}$ and $\mathcal{B}^{(3)}$

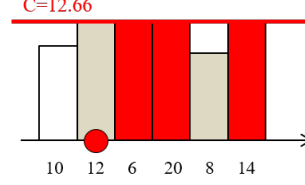


Figure 5: Fourth step ($i=4$): Search of $C^{(4)}$ and $\mathcal{B}^{(4)}$

Fifth step ($i = 5$)

$C^{(5)} = \max \{10, \max \{\frac{48}{4}, \frac{22}{2}\}\} = 12$ and $\mathcal{B}^{(5)} = \mathcal{B}^{(4)} \cup \{1\} = \{1, 2, 3, 4, 6\}$. Notice that period $\{1\}$ enters set \mathcal{B} , which stops the algorithm. No feasible solution exists for a lower capacity. See Figure 6. The optimal value is thus reached for one of the following capacities $C \in \{12, 12.66, 13, 14, 20\}$.

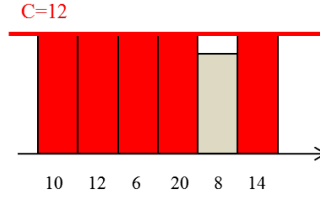


Figure 6: Fifth step ($i=5$): Search of $C^{(5)}$ and $\mathcal{B}^{(5)}$

In the next section, we generalize this algorithm to *energy-LSP* when only one of the energy consumption parameter is null. We also show that it can be implemented to run in time complexity $O(T \log T)$.

5 Polynomial time algorithms for *energy-LSP*

In this section we extend the algorithm presented for the pure capacity acquisition problem and give two polynomial time algorithms, for the cases when one of the consumption energy parameters of our model, either start-ups or unit production, is time-varying and the other is negligible. In both cases, we show that the problem is equivalent to a capacity acquisition problem with additional production restrictions. That is, in addition to the nominal capacity C to be acquired, in each period t we also have a restriction on the production quantity, due to the limited energy availability. We will call this limitation F_t . Hence, in each period t , the production is limited by the minimum between C and F_t . We show that with null setup costs and under non-speculative motive, we can find the nominal capacity C to install and the optimal production plan in time

$O(T^2)$, using the same approach as the one presented in the previous section for the pure capacity acquisition problem.

5.1 Case with $w = 0$, $p_t > 0$, $E_t > 0$: Production energy consumption

We consider that the unit energy consumption p_t and the available energy E_t may vary from one period to another and that the energy needed to start a machine is negligible, that is, $w_t = 0$ for all periods t . Hence, whatever the installed capacity C , i.e. whatever the number of machines started up in period 1, the production quantity x_t in period t is limited by E_t/p_t . We thus have to decide on a nominal capacity C for the system, that is on the number of machines to start at period 1, and on a production plan where the production quantities comply with the restrictions imposed by both the installed capacity C and the energy availability E_t/p_t . In case the acquisition cost function f is concave in C , the resulting problem can be seen as a capacity acquisition problem with additional production restrictions $F_t \equiv \frac{E_t}{p_t}$.

5.2 Case with $p = 0$, $w_t > 0$, $E_t > 0$: Start-up energy consumption

We now assume that the start-up of a machine is the operation requiring the greatest amount of energy (think for instance of furnaces), and that unit production energy consumption is negligible compared to it. More precisely, we consider that the start-up energy consumption w_t and the available amount of energy E_t are time-varying, and that the unit production energy consumption is null for all periods t . This problem seems to significantly differ from problems described in Sections 4 and 5.1 since it may not be possible to install the nominal capacity C , i.e. to start all the machines needed, in the first period of the planning horizon due to the energy constraints.

However, under the assumption of a *linear* acquisition cost f , that is, if $f(C) = fC$, the problem can be seen as a special case of the problem described in 5.1. For the ease of the discussion, we focus on the discrete capacity case, that is, when we have to choose the number m_t^+ of machines to start in each period. Since the production energy consumption is null and the acquisition cost f is linear and stationary, it is dominant to start as many machines as possible in each period, till the desired nominal capacity of the system is obtained. In a period t , we can start at most $\lfloor E_t/w_t \rfloor$ machines. We define:

$$N_t = \sum_{u=1}^t \lfloor E_u/w_u \rfloor$$

Clearly, in any solution, at most N_t machines can be available for production in period t . Hence $x_t \leq UN_t$ is a dominant inequality for the problem, that is, production in period t is limited by $F_t = UN_t$. Again, we have a capacity acquisition problem with additional production restrictions F_t .

5.3 Extension of the pure capacity acquisition algorithm

We consider the capacity acquisition problem with additional production capacities F_t . These additional capacities are imposed by the energy parameters, as explained in 5.1 and 5.2. As a consequence, in each period t , the production is limited by $C_t \equiv \min\{C, F_t\}$. Notice that this production capacity is time-varying. We show how the algorithm described in Section 4 can be adapted to deal with these additional constraints. As in the previous section, we assume null setup costs, non-speculative motive, and a stationary and concave acquisition cost function $f(\cdot)$. Recall that for start-up energy consumption parameters, the acquisition function is restricted to be linear, see Section 5.2.

Again, we start by considering the case of a continuous nominal capacity: C can be any real positive value. To deal with the time-dependent capacities $C_t = \min\{C, F_t\}$, we extend the definition of the no-slack solution $x(\mathcal{B}, C)$ given in the previous section as follows:

$$x_t(\mathcal{B}, C) = \begin{cases} C_t & \text{if } t \in \mathcal{B} \\ D_{t,v(t)} - \sum_{u=t+1}^{v(t)} C_u & \text{if } t \in \mathcal{A} \\ d_t & \text{otherwise.} \end{cases}$$

It is easy to check that Property 1 still holds with this definition for our case of *energy-LSP*. To extend Property 2 to Property 3 below, we need the notion of F -interval. For a given nominal capacity C , we define $\mathcal{F}(C) = \{t | F_t < C\}$ as the set of periods whose capacity is limited by F_t . Let us denote by $F^{(1)} \leq \dots \leq F^{(l)}$ the $l \leq T$ distinct values appearing in set F . We also add the dummy value $F^{(0)} = 0$. We call an F -interval a real interval of the form $[F^{(j)}, F^{(j+1)}]$. Notice that set $\mathcal{F}(C)$ does not change inside an F -interval. We have the following property:

Property 3 *Let $[a, b]$ be an interval included in an F -interval. If there exists a set \mathcal{B} of periods such that $x(\mathcal{B}, C)$ is dominant for all capacities C in $[a, b]$, then $\text{OPT}(C)$ is concave with C over $[a, b]$. As a consequence, the minimum of $\text{OPT}(C)$ over $[a, b]$ is reached at one of the extremities of the interval.*

Proof. Consider C and C' two capacities of interval $[a, b]$. Let $x = x(\mathcal{B}, C)$ and $x' = x(\mathcal{B}, C')$ be the solutions associated to C and C' , respectively. Notice that x and x' differ only by their production in periods of sets \mathcal{A} and \mathcal{B} . Precisely we have:

$$\begin{aligned} \text{OPT}(C') - \text{OPT}(C) &= f(C') - f(C) + \sum_{t \in \mathcal{A}} \tilde{c}_t(x'_t - x_t) + \sum_{t \in \mathcal{B}} \tilde{c}_t(C'_t - C_t) \\ &= f(C') - f(C) + \sum_{t \in \mathcal{A}} \left(\tilde{c}_t \sum_{u=t+1}^{v(t)} (C_u - C'_u) + \sum_{u=t+1}^{v(t)} \tilde{c}_u (C'_u - C_u) \right) \\ &= f(C') - f(C) + \sum_{t \in \mathcal{A}} \sum_{u=t+1}^{v(t)} (\tilde{c}_t - \tilde{c}_u) (C_u - C'_u) \\ &= f(C') - f(C) + \sum_{t \in \mathcal{A}} \sum_{u=t+1}^{v(t)} (\tilde{c}_t - \tilde{c}_u) (\min\{C, F_u\} - \min\{C', F_u\}) \end{aligned}$$

Now, since $[a, b]$ is included in an F -interval, we have the same set of periods where the capacity is limited by F_u for C and C' . More precisely,

$C_u \equiv \min\{C, F_u\}$ is equal to F_u if period u belongs to $\mathcal{F}(b)$, and to C otherwise. Hence, we can write that:

$$\text{OPT}(C') - \text{OPT}(C) = f(C') - f(C) - (C' - C) \sum_{t \in \mathcal{A}} \sum_{u \in \{t, \dots, v(t)\} \setminus \mathcal{F}(b)} (\tilde{c}_t - \tilde{c}_u)$$

Hence, OPT is the sum of a concave function and of a term which varies linearly with C . It is thus a concave function with respect to the capacities over the interval $[a, b]$. \square

In the same way as in Section 4, our algorithm constructs a series of capacities $C^{(1)} > C^{(2)} > \dots > C^{(k)}$ and a series of subsets $\mathcal{B}^{(1)} \subseteq \mathcal{B}^{(2)} \subseteq \dots \subseteq \mathcal{B}^{(k)}$ such that the no-slack solution associated with $\mathcal{B}^{(i)}$ is dominant for any capacity in $[C^{(i+1)}, C^{(i)}]$. However, contrary to the algorithm for the pure acquisition problem, we can not assert that $\mathcal{B}^{(i)}$ is a strict subset of $\mathcal{B}^{(i+1)}$. But we will prove the invariant that, either $\mathcal{B}^{(i)} \subset \mathcal{B}^{(i+1)}$, or $C^{(i+1)}$ does not belong to the same F -interval as $C^{(i)}$. It will allow us to establish that $k \leq 2T$, that is, at most $2T$ different capacities $C^{(i)}$ need to be considered.

To avoid straightforward cases, we assume that a feasible solution exists if $C \geq \max_t \{F_t\}$. We can compute an initial no-slack optimal solution $x^{(0)}$ for capacities $(F_t)_{t=1, \dots, T}$ in time $O(T)$ using a simple backward algorithm to shift the demands. If S_t denotes the amount of demand which has to be shifted backward (i.e. whose production should be anticipated) due to the limited production capacity, then for all period $t = 1, \dots, T$, we have $x_t^{(0)} = \min(S_{t+1} + d_t; F_t)$ and $S_t = S_{t+1} + d_t - x_t$, initializing the backward computation with $S_{T+1} = 0$. Let us define capacity $C^{(1)} = \max_t \{x_t^{(0)}\}$ and $\mathcal{B}^{(1)} = \{t | x_t^{(0)} = \min\{F_t, C^{(1)}\}\}$. Hence $x^{(0)}$ is the no-slack solution associated with set $\mathcal{B}^{(1)}$, and by construction it is optimal for capacity $C = C^{(1)}$ and its value is equal to $\text{OPT}(C^{(1)})$.

Consider now that we have determined a capacity $C^{(i)}$ and a subset $\mathcal{B}^{(i)}$ such that $x(\mathcal{B}^{(i)}, C^{(i)})$ is dominant, and that $1 \notin \mathcal{B}^{(i)}$. Using the same approach as in Section 4, we want to determine the smallest value $C \leq C^{(i)}$ for which $x(\mathcal{B}^{(i)}, C)$ is dominant. For a capacity $C \leq C^{(i)}$, solution $x(\mathcal{B}^{(i)}, C)$ is dominant if and only if the two following conditions hold:

$$\begin{aligned} (i) \quad & \forall t \notin \mathcal{A}^{(i)} \cup \mathcal{B}^{(i)} \quad \min\{C, F_t\} \geq d_t \\ (ii) \quad & \forall t \in \mathcal{A}^{(i)} \quad \min\{C, F_t\} \geq D_{t, v(t)} - \sum_{u=t+1}^{v(t)} \min\{C, F_u\} \end{aligned}$$

To determine C , we strengthen our requirement by imposing that C must be in the same F -interval as $C^{(i)}$. Let us define $\hat{F}^{(i)} = \max_t \{F_t | F_t < C^{(i)}\}$. By convention, we let $\hat{F}^{(i)} = 0$ if all the additional capacities F_t are greater than or equal to $C^{(i)}$. Clearly, interval $[\hat{F}^{(i)}, C^{(i)}]$ is included in an F -interval. Inside this interval, solution $x(\mathcal{B}^{(i)}, C)$ remains dominant if and only if:

$$\begin{aligned} (C1) \quad & \forall t \notin \mathcal{A}^{(i)} \cup \mathcal{B}^{(i)} \quad C \geq d_t \\ (C2) \quad & \forall t \in \mathcal{A}^{(i)} \quad \sum_{u=t}^{v(t)} \min\{C, F_u\} \geq D_{t, v(t)} \\ (C3) \quad & C \geq \hat{F}^{(i)} \end{aligned}$$

Namely, note that, as $x(\mathcal{B}^{(i)}, C^{(i)})$ is a feasible solution, we have: $d_t \leq F_t \leq C^{(i)}$ for all periods t s.t. $t \notin \mathcal{A}^{(i)} \cup \mathcal{B}^{(i)}$ and $t \in \mathcal{F}(C^{(i)})$. By construction, C is restricted to stay within the same F -interval as $C^{(i)}$ so that $\mathcal{F}(C) = \mathcal{F}(C^{(i)})$. We thus have $d_t \leq F_t \leq C$ for all periods t s.t. $t \notin \mathcal{A}^{(i)} \cup \mathcal{B}^{(i)}$ and $t \in \mathcal{F}(C)$. Hence, whatever the value of $C \geq \hat{F}^{(i)}$, condition (i) holds for the periods $t \notin \mathcal{A}^{(i)} \cup \mathcal{B}^{(i)}$ s.t. $\min(C, F_t) = F_t$. It can thus be simplified into condition (C1).

Moreover, observe that $\min\{C, F_u\}$ is equal to F_u if $u \in \mathcal{F}(C^{(i)})$ and to C otherwise. Hence condition (C2) can be rewritten for a period $t \in \mathcal{A}^{(i)}$ as:

$$\sum_{u \in \{t, \dots, v(t)\} \setminus \mathcal{F}(C^{(i)})} C \geq D_{t, v(t)} - \sum_{u \in \{t, \dots, v(t)\} \cap \mathcal{F}(C^{(i)})} F_u$$

Notice that if $F_u < C^{(i)}$ for all the periods u of a block $\{t, \dots, v(t)\}$, certainly solution $x(\mathcal{B}^{(i)}, C)$ remains dominant for any capacity $C \geq \hat{F}^{(i)}$. Also, the right hand side of the previous inequality must be negative in this case. With the convention that the ratio of a negative number divided by 0 is equal to $-\infty$, we define:

$$C^{(i+1)} = \max \left\{ \max_{t \notin \mathcal{A} \cup \mathcal{B}} d_t, \max_{t \in \mathcal{A}} \frac{D_{t, v(t)} - \sum_{u \in \{t, \dots, v(t)\} : F_t < C^{(i)}} F_u}{|\{u \in \{t, \dots, v(t)\} : C^{(i)} \leq F_t\}|}, \max_t \{F_t \mid F_t < C^{(i)}\} \right\} \quad (7)$$

By construction, $x(\mathcal{B}^{(i)}, C^{(i+1)})$ is dominant. We define:

$$\mathcal{B}^{(i+1)} = \{t \mid x_t(\mathcal{B}_i, C^{(i+1)}) = \min(F_t, C^{(i+1)})\}$$

That is, $\mathcal{B}^{(i+1)}$ is constituted of the periods producing at full capacity in solution $x(\mathcal{B}^{(i)}, C^{(i+1)})$. By construction we have $\mathcal{B}_i \subseteq \mathcal{B}_{i+1}$. We have two possible situations, depending on which inequality (C1), (C2) or (C3) is saturated by $C^{(i+1)}$:

- Either $C^{(i+1)} = \hat{F}^{(i)}$. It implies that $\hat{F}^{(i+1)} < \hat{F}^{(i)}$.
- Otherwise, in solution $x(\mathcal{B}^{(i)}, C^{(i+1)})$, at least one period is both a full production period and a regeneration period. That is, there exists a full production period not belonging to $\mathcal{B}^{(i)}$. It implies that $|\mathcal{B}^{(i+1)}| > |\mathcal{B}^{(i)}|$.

Since a set \mathcal{B} can contain at most T periods and there are at most T distinct values F_t , the algorithm stops after at most $2T$ steps. Since each step can be performed in time $O(T)$, determining the $C^{(i)}$'s and the $\mathcal{B}^{(i)}$'s takes $O(T^2)$ time.

To output the optimal solution, we use Property 3. By construction, each capacity interval $[C^{(i+1)}, C^{(i)}]$ is included in an F -interval, and $x(\mathcal{B}^{(i)}, C)$ is dominant for any capacity $C \in [C^{(i+1)}, C^{(i)}]$. As a consequence, for a continuous nominal capacity, the optimal value is reached at one of the capacities $\{C^{(1)}, \dots, C^{(k)}\}$. Notice that determining the optimal cost $\text{OPT}(C^{(i)})$ for a given capacity can be performed in linear time $O(T)$. Hence, the algorithm

runs in time complexity $O(T^2)$. In Appendix, we show that this algorithm can be implemented to run in time complexity $O(T \log T)$ using efficient data structures. We have the following theorem:

Theorem 2 *An optimal solution can be found in time $O(T \log T)$ for energy-LSP with either null start-up energy consumption and concave start-up costs $f()$ or null unit production energy consumption and linear start-up costs $f()$. The algorithm requires at most $O(T)$ evaluations of function $f()$.*

In case of a discrete capacity problem, where C is restricted to be a multiple of U , we consider for each index $i = 1, \dots, k$ the capacities $\lfloor C^{(i)}/U \rfloor U$ and $\lceil C^{(i)}/U \rceil U$. It represents at most $4T$ different values. Again, due to Property 3, we can assert that the discrete capacity minimizing $\text{OPT}(C)$ belongs to this set. Evaluating all these $2k$ values to find the minimal value can be performed in time complexity $O(T)$ as all the breakpoints of $\text{OPT}(C)$ are already computed. Hence, the complexity of the algorithm is not modified in the discrete capacity case.

6 Extension to more general start-up cost functions

In this section, we show that the algorithms presented in this paper can be easily adapted, by slightly increasing their time complexity, to deal with the case where the start-up (capacity acquisition) cost function f is convex, or more generally, is a piecewise function, with a mix of convex and concave pieces. We restrict this discussion to the case of discrete capacities, that is, the admissible values of C belong to set $\{0, U, 2U, \dots, MU\}$, where M is an input and represents the maximal number of available machines. Several real applications can motivate the use of convex start-up cost function. For instance, when the existing capacity in a shop floor is not enough to cover the excess demand, it may be necessary to call for external or remote resources. Those resources are typically more costly due to the supplementary costs of transportation, material handling, and the spot market costs. Another possible application corresponds to capacity reservation contracts, where a certain capacity is reserved a priori at the supplier to obtain a more advantageous price. Once this level is exceeded, the cost to acquire additional capacities become more expensive.

Consider first the case where the start-up cost function f is a convex function of C . In the previous sections, f was assumed to be concave, and, as a consequence, the optimal cost $\text{OPT}(C)$ was a piecewise concave function of the installed capacity C . Looking at the proof of Property 3, we know that on each interval $[C^{(i+1)}, C^{(i)}]$, the optimum cost $\text{OPT}(C)$ is of the form $f(C) + (\alpha C + \beta)$, that is, we add an affine term to function f . Since the sum of two convex functions is a convex function, we have now that $\text{OPT}(C)$ is a piecewise convex function, whose breakpoints are the capacities $C^{(i)}$ determined by our algorithm. The only difference is that we can not assert that the optimum of

$\text{OPT}(C)$ is realized at one of its breakpoints. However, since a convex function is in particular unimodal, we can use a Ternary Search algorithm (see [?]) to find the minimum value of $\text{OPT}(C)$ on each interval $[C^{(i+1)}, C^{(i)}]$. Ternary Search algorithm is a divide & conquer algorithm, similar to a Binary Search for monotonic functions, except that the search domain is divided into three pieces, and at each step the domain is reduced by at least one third. Note that the running time of a Ternary Search algorithm when restricted to integer values is in $O(\log l + 1)$ on an interval containing l integers. Hence, $\text{OPT}(C)$ can be determined on each interval $[C^{(i+1)}, C^{(i)}]$ in $O(\log M)$ operations. As a consequence, the overall time complexity of the algorithm is in $O(T(\log T + \log M))$, where the first term $O(T \log T)$ corresponds to the determination of the breakpoints $C^{(i)}$, and the second term $O(T \log M)$ corresponds to finding the optimum value OPT . Hence the time complexity is only slightly increased. However, notice that the number of evaluations of function f is now in $O(T \log M)$, while the number of evaluations is linear for concave functions, see Theorem 2.

Now, let us consider a more general case where f is a piecewise function of C , where each piece is either concave or convex. Let R be the number of pieces and $\{b_0, b_1, \dots, b_R\}$ be the set of corresponding breakpoints. We assume in the following that the list $b_1 \leq \dots \leq b_R$ of its breakpoints are given in the inputs in the description of f . We also assume that on each interval, f can be evaluated, possibly through an oracle, in constant time $O(1)$. As already discussed in Section 5, $\text{OPT}(C)$ is the sum of the start-up costs given by the piecewise function f plus the production and inventory costs, which are a piecewise linear function of the installed capacity. This can be exploited in order to find the optimal solution in polynomial time thanks to the following algorithm:

- Step 1. Use the algorithm described in Section 5 to identify the set of capacities $C^{(1)}, \dots, C^{(k)}$ where the block structure of the production plan changes, that is, the set of breakpoints of the production/inventory holding cost function, together with the values $\text{OPT}(C^{(1)}), \dots, \text{OPT}(C^{(k)})$. This step can be performed in time $O(T \log T)$, and is independent of the start-up cost function f .
- Step 2. Merge all the breakpoints $\{C^{(1)}, \dots, C^{(k)}\}$ and $\{b_1, \dots, b_R\}$ to obtain an ordered list $a_0, \dots, a_r, \dots, a_l$. Since we assume that both the $C^{(i)}$'s and the b_j 's are indexed in non-decreasing order, this can be done in linear time $O(T + R)$.
- Step 3. On each interval $[a_j, a_{j+1}]$, $\text{OPT}(C)$ is either convex or concave (depending if f itself is either convex or concave). Since each interval $[a_j, a_{j+1}]$ is included by construction in an interval $[C^{(i)}, C^{(i+1)}]$ for some index i , we can evaluate $\text{OPT}(C)$ in constant time using the expression :

$$\text{OPT}(C) = f(C) + \text{OPT}(C^{(i)}) - f(C^{(i)}) + \frac{\text{OPT}(C^{(i+1)}) - \text{OPT}(C^{(i)})}{C^{(i+1)} - C^{(i)}}(C - C^{(i)})$$

Hence the optimum on each interval can be determined in time complexity $O(1)$ if f is concave on the interval, and in time complexity at most $O(\log M)$ if f is convex, using a Ternary Search algorithm.

- Step 4. Output the best value found in the course of the algorithm.

The overall time complexity of the algorithm is dominated by Steps 1 and 3 and is in $O(T \log T + l \log M)$, with l the number of breakpoints to consider, which is bounded by $(T + R)$, and by M , since intervals that do not contain an integer can be ignored. Hence, if $M \geq T + R$, the time complexity is in $O((T + R) \log M)$. Conversely, if $M \leq T + R$, the time complexity is in $O(T \log T + R \log M)$. Notice that if f is piecewise concave, that is, without convex pieces, then the complexity of the algorithm is reduced to $O(T \log T + R)$.

7 Conclusion and perspectives

In this paper we studied a single-item lot sizing problem integrated with energy consumption constraints. We showed that, without energy parameters, the associated problem becomes a pure capacity acquisition problem, as stated in Atamtürk and Hochbaum [Atamtürk and Hochbaum, 2001], but without the subcontracting option. We developed an original solution approach and proposed an $O(T \log T)$ time algorithm to solve this problem. The same algorithm can be easily extended to some special cases of *energy-LSP*, with only one non null energy parameter, without altering its time complexity. This algorithm allows to tackle with quite general acquisition cost functions as well, such as concave, convex, or piecewise functions with a mix of concave and convex pieces, by slightly modifying its computational complexity.

We note that the same integrated energy & lot sizing problem can also be seen in the context of a carbon cap to not exceed in each period, and the start-up & production activities that emit carbon units. Another possible application is a certain limited manpower to not exceed in each period, together with a capacity limitation of parallel resources. Notice that the common property of those systems is the existence of a non-storable and periodical capacitated resources.

Among the possible research perspectives of this work, one can investigate whether our approach can be successfully extended to the capacity acquisition problem with subcontracting option or to more general cost structure, keeping a low resolution time complexity. Regarding the *energy-LSP* problem, one open question is the existence and the design of a polynomial time algorithm for the case with both positive start-up and production energy consumption. Another extension of the model would be to consider that a started machine consumes energy not only while actively producing but also while being idle.

Acknowledgments.

This project was supported by Conseil Régional de Lorraine (2015-2016). We are grateful to the anonymous referees, whose comments helped to improve the presentation of this paper.

References

- [Almada-Lobo et al., 2010] Almada-Lobo, B., Klabjan, D., Carravilla, M. A., and Oliveira, J. F. (2010). Multiple machine continuous setup lotsizing with sequence-dependent setups. *Computational Optimization and Applications*, 47(3):529–552.
- [Artigues et al., 2013] Artigues, C., Lopez, P., and Haït, A. (2013). The energy scheduling problem: Industrial case-study and constraint propagation techniques. *International Journal of Production Economics*, 143(1):13 – 23.
- [Atamtürk and Hochbaum, 2001] Atamtürk, A. and Hochbaum, D. S. (2001). Capacity acquisition, subcontracting, and lot sizing. *Management Science*, 47(8):1081–1100.
- [Biel and Glock, 2016] Biel, K. and Glock, C. H. (2016). Systematic literature review of decision support models for energy-efficient production planning. *Computers and Industrial Engineering*, 101:243 – 259.
- [Bitran and Yanasse, 1982] Bitran, G. R. and Yanasse, H. H. (1982). Computational complexity of the capacitated lot size problem. *Management Science*, 28(10):1174–1186.
- [Bruzzone et al., 2012] Bruzzone, A., Anghinolfi, D., Paolucci, M., and Tonelli, F. (2012). Energy-aware scheduling for improving manufacturing process sustainability: A mathematical model for flexible flow shops. *{CIRP} Annals - Manufacturing Technology*, 61(1):459 – 462.
- [Castro et al., 2013] Castro, P. M., Sun, L., and Harjunkski, I. (2013). Resource-task network formulations for industrial demand side management of a steel plant. *Industrial & Engineering Chemistry Research*, 52(36):13046–13058.
- [Chen et al., 2005] Chen, Y., Das, A., Qin, W., Sivasubramaniam, A., Wang, Q., and Gautam, N. (2005). Managing server energy and operational costs in hosting centers. In *Proceedings of the 2005 ACM SIGMETRICS International Conference on Measurement and Modeling of Computers Systems*, pages 303–314, Banff, Alberta, Canada.
- [Dastidar and Nagi, 2005] Dastidar, S. G. and Nagi, R. (2005). Scheduling injection molding operations with multiple resource constraints and sequence dependent setup times and costs. *Computers & Operations Research*, 32(11):2987 – 3005.

- [de Matta and Guignard, 1994] de Matta, R. and Guignard, M. (1994). Dynamic production scheduling for a process industry. *Operations Research*, 42(3):492–503.
- [Florian et al., 1980] Florian, M., Lenstra, J. K., and Rinnooy Kan, A. H. G. (1980). Deterministic production planning: Algorithms and complexity. *Management Science*, 26(7):669–679.
- [Gahm et al., 2016] Gahm, C., Denz, F., Dirr, M., and Tuma, A. (2016). Energy-efficient scheduling in manufacturing companies: A review and research framework. *European Journal of Operational Research*, 248(3):744 – 757.
- [Heck and Schmidt, 2010] Heck, M. and Schmidt, G. (2010). *Lot-Size Planning with Non-linear Cost Functions Supporting Environmental Sustainability*, pages 1–6. Springer Berlin Heidelberg, Berlin, Heidelberg.
- [Jans, 2009] Jans, R. (2009). Solving lot-sizing problems on parallel identical machines using symmetry-breaking constraints. *INFORMS Journal on Computing*, 21(1):123–136.
- [Jans and Degraeve, 2004] Jans, R. and Degraeve, Z. (2004). An industrial extension of the discrete lot-sizing and scheduling problem. *IIE Transactions*, 36(1):47–58.
- [Jans and Degraeve, 2008] Jans, R. and Degraeve, Z. (2008). Modeling industrial lot sizing problems: a review. *International Journal of Production Research*, 46(6):1619–1643.
- [Kaczmarczyk, 2011] Kaczmarczyk, W. (2011). Proportional lot-sizing and scheduling problem with identical parallel machines. *International Journal of Production Research*, 49(9):2605–2623.
- [Karmarkar and Schrage, 1985] Karmarkar, U. S. and Schrage, L. (1985). The deterministic dynamic product cycling problem. *Operations Research*, 33(2):326 – 345.
- [Lasdon and Terjung, 1971] Lasdon, L. S. and Terjung, R. C. (1971). An efficient algorithm for multi-item scheduling. *Operations Research*, 19(4):946–969.
- [Li and Meissner, 2011] Li, H. and Meissner, J. (2011). Capacitated dynamic lot sizing with capacity acquisition. *International Journal of Production Research*, 49(16):4945–4963.
- [Marinelli et al., 2007] Marinelli, F., Nenni, M. E., and Sforza, A. (2007). Capacitated lot sizing and scheduling with parallel machines and shared buffers: A case study in a packaging company. *Annals of Operations Research*, 150(1):177–192.

- [Martínez-Costa et al., 2014] Martínez-Costa, C., Mas-Machuca, M., Benedito, E., and Corominas, A. (2014). A review of mathematical programming models for strategic capacity planning in manufacturing. *International Journal of Production Economics*, 153:66 – 85.
- [Masmoudi et al., 2017] Masmoudi, O., Yalaoui, A., Ouazene, Y., and Chehade, H. (2017). Lot-sizing in a multi-stage flow line production system with energy consideration. *International Journal of Production Research*, 55(6):1640–1663.
- [Mitra et al., 2012] Mitra, S., Grossmann, I. E., Pinto, J. M., and Arora, N. (2012). Optimal production planning under time-sensitive electricity prices for continuous power-intensive processes. *Computers and Chemical Engineering*, 38:171 – 184.
- [Mitrani, 2013] Mitrani, I. (2013). Managing performance and power consumption in a server farm. *Annals of Operations Research*, 202:121–134.
- [Özdamar and Birbil, 1999] Özdamar, L. and Birbil, S. I. (1999). A hierarchical planning system for energy intensive production environments. *International Journal of Production Economics*, 58(2):115 – 129.
- [Paulus and Borggrefe, 2011] Paulus, M. and Borggrefe, F. (2011). The potential of demand-side management in energy-intensive industries for electricity markets in germany. *Applied Energy*, 88(2):432 – 441. The 5th Dubrovnik Conference on Sustainable Development of Energy, Water and Environment Systems, held in Dubrovnik September/October 2009.
- [Pochet and Wolsey, 2006] Pochet, Y. and Wolsey, L. A. (2006). *Production Planning by Mixed Integer Programming*. Springer.
- [Salonitis and Ball, 2013] Salonitis, K. and Ball, P. (2013). Energy efficient manufacturing from machine tools to manufacturing systems. *Procedia CIRP*, 7:634 – 639.
- [Santos and Almada-Lobo, 2012] Santos, M. O. and Almada-Lobo, B. (2012). Integrated pulp and paper mill planning and scheduling. *Computers & Industrial Engineering*, 63(1):1 – 12.
- [Sawik, 2009] Sawik, T. (2009). Coordinated supply chain scheduling. *International Journal of Production Economics*, 120(2):437 – 451. Special Issue on Introduction to Design and Analysis of Production Systems.
- [Schultz et al., 2015] Schultz, C., Sellmaier, P., and Reinhart, G. (2015). An approach for energy-oriented production control using energy flexibility. *Procedia {CIRP}*, 29:197 – 202. The 22nd {CIRP} Conference on Life Cycle Engineering.

- [Shoreh et al., 2016] Shoreh, M. H., Siano, P., Shafie-khah, M., Loia, V., and Catal  o, J. P. (2016). A survey of industrial applications of demand response. *Electric Power Systems Research*, 141:31 – 49.
- [Tang et al., 2012] Tang, L., Che, P., and Liu, J. (2012). A stochastic production planning problem with nonlinear cost. *Computers & Operations Research*, 39(9):1977 – 1987.
- [Uzel, 2004] Uzel, E. (2004). A mathematical modeling approach to energy cost saving in a manufacturing plant. Master’s thesis, Izmir Institute of Technology.
- [Wagner and Whitin, 1958] Wagner, H. M. and Whitin, T. M. (1958). Dynamic version of the economic lot size model. *Management Science*, 5(1):89–96.
- [Waldemarsson et al., 2013] Waldemarsson, M., Lidestam, H., and Rudberg, M. (2013). Including energy in supply chain planning at a pulp company. *Applied Energy*, 112:1056 – 1065.
- [Zhao et al., 2016] Zhao, H., Ierapetritou, M. G., and Rong, G. (2016). Production planning optimization of an ethylene plant considering process operation and energy utilization. *Computers & Chemical Engineering*, 87:1 – 12.

Appendix: Improvement of the time complexity

We show in this appendix that the algorithm described in Section 4 and 5 can be implemented to run in time complexity $O(T \log T)$, as stated in Theorem 2.

We focus on how determining $\text{OPT}(C)$. Recall that $\text{OPT}(C)$ is a (continuous) piecewise concave function with $k \leq 2T$ breakpoints, $C^{(1)} \geq C^{(2)} \geq \dots \geq C^{(k)}$. As a consequence, the optimum is realized at one of the breakpoints, and thus it suffices to evaluate $\text{OPT}(C^{(i)})$ for $i = 1, \dots, k$, where k is the number of step of the algorithm. We will establish that the breakpoints $C^{(i)}$ ’s can be determined in overall complexity $O(T \log T)$ using efficient data structures. Notice that evaluating an optimal solution for a given capacity C can be performed in linear time. However, as we may have up to $2T$ values to evaluate in the course of the algorithm, it is still a too large computational effort. Instead, we compute the optimal cost $\text{OPT}(C^{(i+1)})$ incrementally from the value of $\text{OPT}(C^{(i)})$. As established in Property 3, the optimum value $\text{OPT}(C)$ is concave on each interval $[C^{(i+1)}, C^{(i)}]$. More precisely, let us denote by $\alpha^{(i)}$ the quantity:

$$\alpha^{(i)} = \sum_{t \in \mathcal{A}^{(i)}} \sum_{u \in \{t, \dots, v(t)\}: C^{(i)} \leq F_u} (\tilde{c}_t - \tilde{c}_u)$$

Then, following the proof of Property 3, we have:

$$\text{OPT}(C^{(i+1)}) = \text{OPT}(C^{(i)}) + f(C^{(i+1)}) - f(C^{(i)}) + \alpha^{(i)}(C^{(i+1)} - C^{(i)})$$

As a consequence, value $\text{OPT}(C^{(i+1)})$ can be computed in constant time, if $\text{OPT}(C^{(i)})$ and the slope $\alpha^{(i)}$ are known.

We now show how these different values can be updated at each step of the algorithm. For a given set \mathcal{B} , recall that a *block* is a set of consecutive periods $\{t, \dots, v\}$ such that $t \in \mathcal{A}$ and $v = v(t)$. This corresponds to a subplan of the associated no-slack solution. We maintain in an array all the blocks of the current set \mathcal{B} . Initially, there is one block for each single period. In addition, each period refers to the block where it belongs to, simply by using an index. For a current capacity C and a current set \mathcal{B} , each block B contains the following data:

- Its first and last periods, $t(B)$ and $v(B)$, respectively.
- $D(B)$, the total demand of the periods of the block, that is, $\sum_{t \in B} d_t$.
- $F(B)$, defined as $\sum_{u \in B: F_u < C} F_u$ if set $\{u \in B : F_u < C\}$ is not empty, and 0 otherwise.
- Its pseudo-cost $\tilde{c}(B)$, defined as $\sum_{u \in B: C \leq F_u} \tilde{c}_u$ if set $\{u \in B : C \leq F_u\}$ is not empty, and 0 otherwise.
- Its pseudo-size $n(B)$, defined as $|\{u \in B : C \leq F_u\}|$, that is, the number of periods of the block with an additional capacity F_t being greater than or equal to the nominal capacity C . By convention, $|\emptyset| = 0$.

We need these different quantities in order to be able to update efficiently the slope $\alpha^{(i)}$ of the current no-slack solution. Consider a block B and let $t = t(B)$ be its first period. If period $t \in \mathcal{A}^{(i)}$, then $B = \{t, \dots, v(t)\}$ and we have:

$$\sum_{u \in \{t, \dots, v(t)\}: C^{(i)} \leq F_u} (\tilde{c}_t - \tilde{c}_u) = n(B)\tilde{c}_t - \tilde{c}(B)$$

If period t does not belong to set $\mathcal{A}^{(i)}$, then $B = \{t\}$, that is, t is a lot for lot period where demand d_t is produced. In this case, we clearly have $n(B)\tilde{c}_t - \tilde{c}(B) = 0$. Hence, we can rewrite the slope $\alpha^{(i)}$ as:

$$\alpha^{(i)} = \sum_{B: B \text{ a block of } \mathcal{B}^{(i)}} (n(B)\tilde{c}_{t(B)} - \tilde{c}(B))$$

Now, consider a step of the algorithm. Suppose that we have found the current breakpoint $C^{(i)}$. Finding the next breakpoint consists in:

- (i) Determining $C^{(i+1)}$ according to Equation (7)
- (ii) Updating the blocks of the new set $\mathcal{B}^{(i+1)}$
- (iii) Evaluating $\text{OPT}(C^{(i+1)})$

As step (iii) can be computed in constant time, we focus on the two first steps.

For the first step (i), we need to find the largest value of C saturating one of the constraints (C1), (C2) or (C3), see Equation (7). For constraints (C1) and (C3), we can sort via a precomputation phase all the F_t and d_t values in non-increasing order. This precomputation phase can be performed in time $O(T \log T)$. During the execution of the algorithm, we maintain an index of the current position in each of the arrays. Clearly, at each step, we can find the next F_t value lower than $C^{(i)}$ in constant time. Recall that each period stores the block it belongs to, and hence can be determined in constant time if it belongs to the current set $\mathcal{A}^{(i)}$ or $\mathcal{B}^{(i)}$ (by comparing its index with the values $t(B)$ and $v(B)$ of its block). From the current position in the sorted array of d_t values, we scan the array until we encounter a period not in the current set $\mathcal{A}^{(i)}$ or $\mathcal{B}^{(i)}$. Notice that it may require up to $O(T)$ operations. Nevertheless, since a period entering $\mathcal{A} \cup \mathcal{B}$ does not leave this set in the following steps of the algorithm, each period is scanned only once. Hence the overall time complexity is in $O(T)$. Finally, if $C^{(i+1)}$ saturates constraint (C2), we need to determine the largest value of:

$$e_t \equiv \frac{D_{t,v(t)} - \sum_{u \in \{t, \dots, v(t)\} : F_t < C^{(i)}} F_u}{|\{u \in \{t, \dots, v(t)\} : C^{(i)} \leq F_t\}|}$$

over all the periods $t \in \mathcal{A}^{(i)}$. Notice that e_t is also equal, with our notations, to:

$$e_t = \frac{D(B) - F(B)}{n(B)}$$

with $B = \{t, \dots, v(t)\}$ being the block beginning at period t . We note that for a given block B , value e_t does not change if capacity C remains in the same F -interval. Hence, at each step, we need to retrieve the largest value e_t and to update (delete and re-insert) at most one e_t , corresponding to the block updated in the current set \mathcal{B} . In order to maintain the e_t values of the blocks in a data structure H , we can use for instance a heap or a balanced binary tree such that these operations can be performed in logarithmic time in the number of elements in the structure. Notice that the number of periods of \mathcal{A} is at most T , and thus each operation can be performed in time complexity $O(\log T)$. In conclusion, step (i) of the algorithm requires $O(T \log T)$ operations in total.

For the second step (ii), we need to update the blocks of the current set $\mathcal{B}^{(i+1)}$. We distinguish 3 cases depending which constraint (C1), (C2) or (C3) has been saturated by $C^{(i+1)}$:

Case 1. Constraint (C1) is saturated. Let t be a period not in $\mathcal{A}^{(i)} \cup \mathcal{B}^{(i)}$ such that $d_t = C^{(i+1)}$. In this case period t belongs to the current solution $\mathcal{B}^{(i+1)}$. This corresponds to the merge of the block $B = \{t\}$ with its preceding block $B' = \{t(B'), \dots, t-1\}$. Block B' can be found in constant time, since period $(t-1)$ keeps track of its index. Merging B and B' into a new

block $B'' = \{t(B'), \dots, v(B)\}$ can be done in constant time, since each attribute $(F(B''), n(B''), \dots)$ can be easily deduced from the attributes of blocks B and B' . In order to update the new belonging of each period of B'' to this block, we chose to index B'' with the index of B' , such that only period t needs to be updated.

Case 2. Constraint (C2) is saturated. Let t be a period in $\mathcal{A}^{(i)}$ such that $e_t = C^{(i+1)}$. We are in a situation similar to the previous case when constraint (C1) was saturated: this corresponds to the merge of the block $B = \{t, \dots, v(t)\}$ with its preceding block $B' = \{t(B'), \dots, t-1\}$. We have the same operations to perform. However, updating the index of its block for each period of B'' may now be the most costly part, since both B and B' may be large sets of periods. For that, we choose for the index of B'' the index of the largest block between B and B' , such that we need to reindex only the periods of the smallest block. Of course, we may still have up to $O(T)$ periods to reindex. However, we claim that the overall complexity for the algorithm is bounded by $O(T \log T)$. To prove that, let $g(m)$ be the total number of reindexations incurred by the algorithm from the beginning to obtain a block of size m . That is, for each period of the block, we count for the number of times it has been reindexed by the algorithm. We prove by induction that $g(m) \leq m \log m$. For $m = 1$, this is true since no merging is performed for a block of a single period. Consider now a block of m periods, requiring $g(m)$ reindexation, obtained by merging two blocks A and B . Without loss of generality, assume that $|A| \leq |B|$. We have:

$$\begin{aligned} g(n) &\leq |A| + g(|A|) + g(|B|) \\ &\leq |A| + |A| \log |A| + |B| \log |B| \\ &\leq |A| + |A|(\log m - 1) + |B| \log m \\ &\leq (|A| + |B|) \log m = m \log m \end{aligned}$$

The first inequality is due to the fact that the algorithm reindexes only the smallest block, A , when merging A and B . The induction hypothesis implies the second inequality. The third inequality simply uses that $|A| \leq |B|$ and hence $|A| \leq m/2$. This allows us to establish that at most $T \log T$ reindexations are performed by the algorithm.

In both cases, the values e eventually associated with blocks B and B' (if they are not reduced to a single period) are removed from data structure H , and the new value e associated with B'' is inserted, which can be done in time $O(\log T)$. Finally, we update the slope by setting $\alpha^{(i+1)} = \alpha^{(i)} + (n(B'')\tilde{c}_{t(B'')} - \tilde{c}(B'')) - (n(B')\tilde{c}_{t(B')} - \tilde{c}(B')) - (n(B)\tilde{c}_{t(B)} - \tilde{c}(B))$.

Case 3. Constraint (C3) is saturated. Let t be a period such that $C^{(i+1)} = F_t$. In this case, the structure of the blocks is not modified, that is, $\mathcal{B}^{(i+1)} = \mathcal{B}^{(i)}$. However, the slope $\alpha^{(i+1)}$ is modified. More precisely, let B be the

block where period t belongs. We update the attributes of block B by setting $F(B) \leftarrow F(B) - F_t$, $\tilde{c}(B) \leftarrow \tilde{c}(B) + \tilde{c}_t$ and $n(B) = n(B) + 1$. The slope $\alpha^{(i+1)}$ is then equal to $\alpha^{(i+1)} = \alpha^{(i)} + (\tilde{c}_{t(B)} - \tilde{c}_t)$. In this case, we update the e value associated with block B by removing it from H and by inserting its new value.

The time complexity of the algorithm is thus dominated by the precomputation phase, the updating of the e values in the data structure H , and the reindexing of the periods to keep track of their blocks, each one requiring $O(T \log T)$ operations. Hence the algorithm can be implemented in overall time complexity $O(T \log T)$.