Optimally managing chemical plant operations: an example oriented by Industry 4.0 paradigms [†]

Marco Vaccari,[‡] Riccardo Bacci di Capaci,[*,‡] Elisabetta Brunazzi,[*,‡] Leonardo Tognotti,[‡] Paolo Pierno,[¶] Roberto Vagheggi,[¶] and Gabriele Pannocchia[‡]

‡*Department of Civil and Industrial Engineering, University of Pisa, Italy*

¶*Altair Chimica SPA, Saline di Volterra (PI), Italy*

E-mail: riccardo.bacci@unipi.it; elisabetta.brunazzi@unipi.it

Abstract

Updating industrial facilities in order to increase the level of automation and digitalization to match Industry 4.0 paradigms has become essential for many companies. Following such trend, this paper presents a Real-Time Optimization (RTO) algorithm that plays a central role in a larger project framework devoted to highly interconnect different network components of an Italian chemical industrial site. The proposed methodology aims at best managing the production rates of various products in order to fulfill a sales plan organized to satisfy numerous client requests. The considered model takes into account both batch and continuous processes, as well as salable and non-storable products. The algorithm structure relies on the use of a non-linear optimization scheme and on the concepts of batch scheduling. Different features of the proposed methodology have been tested on real plant data showing how the predicted forecast always improved the initial operation plan, by considering both aspects of feasibility and

[†] A preliminary version of this work has been presented in Vaccari et al. [1].

economic nature. The use of the proposed algorithm assures the basis for fully integrating the control systems and the selling department of the facility in a more interactive and responsive manner.

1 Introduction

Within Industry 4.0 paradigms, both process simulation and simulation-based optimization have acquired a relevant role in the definition of the so-called *virtual twin* of the physical process[2]. In this context, mathematical modeling is not anymore dedicated to describe an industrial process, but also any product or a service on top of which specific analyses and/or suitable strategies have to be performed[3]. Another important aspect recently taken into consideration involves maintaining a reliable model by monitoring the process with the appropriate strategies of data collection[4,5]. Even though the Industry 4.0 paradigms have been formulated quite recently, the approach which deals with process simulations and optimization is nowadays well-establish and goes under the name of Real-Time Optimization (RTO)[6]. The RTO methods exploit process measurements to run an optimization framework that often, but non mandatorily, relies on a (possibly inaccurate) process model and data extrapolated from measurements. Due to their versatility, process industry applications of RTO strategies nowadays are multiple and can be found in different fields, as managing energy consumption efficiently[7] or optimizing batch and continuous operations[8].

The specific set of application of RTO methodologies oriented to optimally manage large and complex industrial facilities, takes the name of process scheduling. Finding the optimal production strategy to fulfill the sale requirements by solving scheduling problems is the typical objective. Such field of RTO finds applicability both on continuous and batch plants. Mixed Integer Linear Programming (MILP) models have been often used in scheduling problem of batch reactor plants[9], but also to deal with inventory management in refinery operations,[10] or with the organization of a petroleum transportation system[11]. For example, cyclic scheduling and operation of optimal multistage continuous plants are treated in Alle and Pinto[12]. A global optimization algorithm is here used to relax the nonconvexity present in the proposed mixed-integer nonlinear programming (MINLP) formulation. Scheduling has also been integrated with control via multiparametric programming by considering both continuous and binary decisions[13]. A surrogate model and offline maps of optimal scheduling are employed to operate the controller. Reactive scheduling has also been studied via MILP algorithms for short-term problems[14]. The algorithm robustness is tested against unit shutdown and orders modification on a large-scale industrial batch plant. Simultaneous batching and scheduling in complex multiproduct plants has also been addressed by Sundaramoorthy and Maravelias[15].

If merging RTO and control with supply chain higher level layers initially involved heavy computational costs due to millions of variables,[16] tremendous developments in efficient large-scale NonLinear Programming (NLP) algorithms have led to an increase of applications in the chemical industry[17]. Pontes et al.[18] described RTO strategies, both static and dynamic, to be implemented in industrial polymerization process. The authors showed how the proposed methodologies improve the process economic performance, rather than using traditional industrial practices. Krishnamoorthy et al.[19] proposed hybrid versions of RTO in order to overcome the limiting factor of its implementation in the industrial plants, that is, waiting for steady-state conditions. Moreover, RTO techniques have been seen as a key instrument to success in the increasing competition of refining industries, allowing one to optimize performance while fulfilling safety constraints[20]. RTO and predictive control have also been integrated. For example, this happened in a petrochemical plant to improve the automation level of the styrene production subject to disturbance and plant-model mismatch[21].

While petrochemical and refining industries have accepted RTO in the past few years, its application to different chemical processes is still limited. To overcome this, Hernandez et al.[22] proposed an RTO scheme applicable to a complex catalyzed process showing operational improvements despite modeling errors. A recent application on the bioethanol production showed that the use of closed-loop dynamic RTO in the ethanol distillation process can improve the profitability of this product as an environmentally friendly fuel[23]. Another RTO example managing the operability of hybrid energy systems to minimize operating costs while fulfilling all electrical and thermal load requirements can be found in Vaccari et al..[24] When planning to optimally manage a large chemical plant, many different aspects can be important to optimize. To this aim, Wang and Wang[25] proposed a multi-objective multi-factorial optimization model which takes account of product quality, production capacity and energy consumption.

Therefore, the main objective of the present work is to build an RTO scheme, according to the paradigms of Industry 4.0, to optimize a set of production rates of different products in a chemical plant facility. The minimization of an economic objective function is constrained by feasibility of product stocks and fulfillment of a complex and variable sales plan. It has to be noted that the current work is not only a mere extended version of what discussed in Vaccari et al.[1], but presents a more comprehensive formulation oriented to best fit company needs and constraints.

The rest of the paper is organized as follows. Section 2 presents the problem description, generalities and main components. A more detailed definition of variables, constraints, and a formalization of the proposed methodology is illustrated in Section 3. Description of a suitable preliminary scheduling for batch products, details about the optimization objective function and other algorithm implementation features can be found in this section. A real case study from an Italian inorganic chemical industry, together with results and discussion about the methodology test, is showed in Section 4. Section 5 then concludes the paper underlining the main achievements.

2 Problem definition

The problem considered in this work is to model and optimally schedule the production plan of an Italian industrial site of the inorganic chemical sector. The maximum horizon along which the optimization problem is developed is a week long, since after seven days it is neither safe nor convenient to forecast production. This work is part of a larger competitive project addressed to enhance the factory management of Altair Chimica SPA (later on cited as *Altair*), including aspects of automation, digitalization, machine learning and process computerization. The project aims at fully integrating the proposed RTO system with the Distributed Control System (DCS) and the local area network (LAN) of the industrial site through a specifically designed interface[26].

A block diagram of the project architecture that identifies the position of the developed RTO system among the other players of the industrial site is shown in Figure 1. The acquisition of production data takes place through a specifically developed dynamic connection between the DCS and the management system, in which client orders are entered. An additional connection, under definition, will allow the management system to automatically receive client orders, avoiding the manual entry phase, currently in place. The proposed RTO system can acquire input data from the DCS and give its outputs to DCS itself at fixed times; therefore, the optimization system occupies a hierarchically superior level to (basic and advanced) controllers and works as a fully automatic operator.

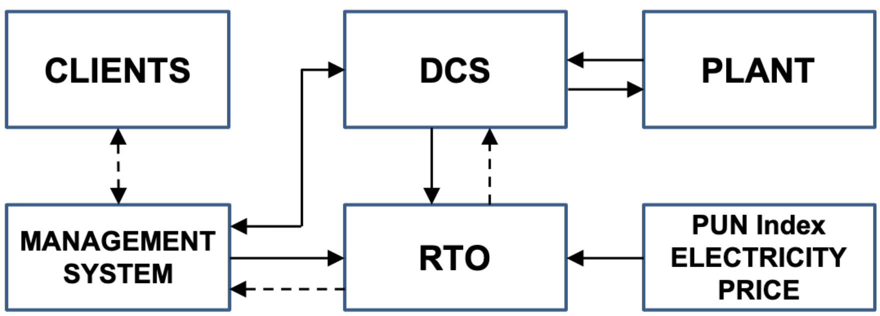

Figure 1: Block diagram of the local computer system. Dotted lines are for connections under definition.

Various (n_P) products of interest of the company are considered in this work. The starting modeling idea for the optimization problem is the weekly production plan designed by the operators of selling department on the basis of the various sales according to client requests. Let us introduce some notation and name x_j, with $j = 1, 2, \ldots$, the hourly production column vector of product j, i.e. $x_j = [x_j^0, \ldots, x_j^i, \ldots, x_j^{n_h-1}] \in \mathbb{R}^{n_h}$, where n_h is the total number of hours to be optimized, e.g., $n_h = 24 \times 7 = 168$ h is the optimization horizon length for a week ($n_d = 7$).

Sales plans of each product are input data obtained from the selling department of the company and used within the optimization problem as parameters. For each day considered in the optimization, let us define the *selling time* as τ_d, with $d = 1, 2, \ldots, n_d$, and let us establish that a sale is satisfied if and only if the stock of the considered product j contains enough material at time τ_d. From this definition, it follows that the sales vector of product j assumes the following form: $S_j = [S_j^0, \ldots, S_j^i, \ldots, S_j^{n_h-1}] \in \mathbb{R}^{n_h}$ in which the only non-zero components are the ones for $i = \tau_d$.

Stocks of each product are calculated within the optimization algorithm as functions of sales and production rates, and they are, as well, bounded by physical constraints. Analogously to production rates, let us define the initial stock of the product j as $\sigma_j^0 \in \mathbb{R}$ and its evolution over time is obtained by mass balance as follows:

$$\sigma_j^{i+1} = \sigma_j^i + \sum_{k=1}^{L_j} x_{j,k}^i - S_j^i - a_j(x)^i + E_j^i \quad \forall\, i = 0, \ldots, n_h \tag{1}$$

The stock σ_j depends linearly also on the function $a_j(x)$, named self-consumption, because some of the products are consumed within the industrial site to obtain other chemicals. Note that σ_j^0 is a parameter within the optimization problem, as it represents the initial stock value of product j before the optimization horizon. Its value is read directly from the DCS at the moment in which the optimization is intended to start. Moreover, some of the products can be obtained in multiple production lines, that is, L_j different production rates contribute to the same stock σ_j. Another possibility for storable products is to have external provision of raw materials (E_j), that are then transformed to final products of interest for the industrial site. From a modeling point of view, since the generic raw material comes from other suppliers and, therefore, its orders are still handled by the management system, it is convenient to represent its provision in a similar way as done for the sales plan.

On the other hand, some products cannot be stocked within the industrial site due to specific safety or logistic reasons. Since they may not be provisioned or sold either, they must be consumed within the facility. Hence, their material balance equation (1) reduces to:

$$0 = \sum_{k=1}^{L_j} x_{j,k}^i - a_j(x)^i \quad \forall\, i = 0, \ldots, n_h \tag{2}$$

Another important note is that some of the considered products are produced by means of batch reactors. This implies that the corresponding hourly production rate x_j can assume only a limited number of values. In particular, it is zero throughout most of the optimization period and then assumes a certain positive value for a few specific times. Let us identify the number of batch products as n_B, where $n_B < n_P$.

Therefore, the scope of the presented methodology is to find the best production schedule for all the n_P products, by minimizing operating costs and the summation of stocks of certain products while fulfilling all the various constraints. In the process control field, this indeed represents an RTO-level decision, since its main purpose is to communicate the various set-points to be used in the control layer, e.g. DCS.

3 Proposed methodology

In this section, the various features of our RTO scheme for optimizing the production plan and based on algorithms developed in Python, are presented and detailed.

3.1 Data, variables and constraints

The hourly production rates of the various products are treated as optimization variables subject to different bound constraints. Let us identify the optimization variable vector with $x = [{x_1}^T, \dots x_j^T, \dots, x_{n_P - n_B}^T]^T \in \mathbb{R}^{n_x}$, where $n_x = (n_P - n_B) n_h$.

Input data and parameters of the problem are sale vector S_j and initial stock value σ_j^0 of each product. These quantities are used, in particular, to build the material balances of all the chemicals treated in the facility and hence involved in the algorithm. Additional both linear and non-linear relations implying different components of x, and safety considerations represent the problem constraints. Minimum and maximum values for bound and process constraints have been set as constant. Initialization values for the optimization variables are taken from the weekly production plan designed by hand by the selling department.

3.2 Scheduling procedure for batch products

As anticipated in Section 2, the company produces also n_B different products in batch reactors. In Section 1, it has been underlined how batch scheduling is a necessary step when dealing with chemical plant optimization[27,28]. A comprehensive review on batch process scheduling can be found in Méndez et al.[29]. The different typologies of batch products considered here are named B, i.e. $B_1, \ldots, B_l, \ldots, B_{n_B}$. Despite usually batch products result from multiple batch operations, we underline that each considered batch products B_l is here obtained via a single reaction operation. The correlated service operations are not here considered and, for this reason, the associated specific reaction time t_{B_l} is comprehensive of service time ($t_{B_l}^{serv}$). We assume that each reactor produces an amount W_{B_l} that depends on the type of B_l, so that the corresponding "hourly production rate" can be calculated as follows: $x_{B_l} = \frac{W_{B_l}}{t_{B_l}}$ with $l = 1, \ldots, n_B$. Note that these hourly production rates are not considered as optimization variables in order to avoid dealing with a mixed-integer problem, where batch and continuous productions are simultaneously optimized. Therefore, a specific, preliminary optimization procedure for batch products has been implemented inspired by the General Precedence (GP) notion[29].

The n_r batch reactors available at the plant facility in which product B_l can be produced are named $R_1, \ldots, R_r, \ldots, R_{n_r}$. Since they can be employed simultaneously and at any time during the day, a criterion for scheduling their operation is needed. The criterion chosen is rather simple, yet effective, for the company needs and it is based on the sales plan of each B_l. The underlying idea can be expressed by the sentence *the first needed is the first to be produced*. Practically, the procedure scans every selling times τ_d of each B_l and registers the corresponding sale. Then, depending on the current stock value recalculated at each iteration, the production of product B_l related to its sale request is scheduled or not. In order to make a comparison with the selling times τ_d, another time variable is defined: the reactor time ($T_{R_1}, \ldots, T_{R_r}, \ldots, T_{R_{n_r}}$), which is linked to the reactor employment and, indicating the last time instant a reactor is used, allows us to track the production assignment. The reactor time starts from zero for an unemployed reactor, and grows depending on the production schedule for the considered reactor, that is, the more a reactor

Table 1: Sales plan example for batch products. $S_{B_l}^{\tau_d}$ are tons of B_l requested by the client on day d.

	τ_1	τ_2	τ_3	τ_4	τ_5
B_1	0	$S_{B_1}^{\tau_2}$	0	0	0
B_2	$S_{B_2}^{\tau_1}$	0	0	0	$S_{B_2}^{\tau_5}$
B_3	0	$S_{B_3}^{\tau_2}$	$S_{B_3}^{\tau_3}$	0	$S_{B_3}^{\tau_5}$

is employed the greater is its T_{R_r}. If a reactor is in use and not available since the start of the optimization horizon, the corresponding future T_{R_r} and the product B_l in production have to be known. Even though this second information is not directly needed for the scheduling procedure, it will be used later on in the optimization problem. The scheduled batch is placed always in the reactor that has the lowest T_{R_r}. When for the product B_l is required more than one batch to cover the sale, the sequencing procedure schedules the first batch in the selected reactor, and then scans all the reactor times to see which one is the smallest. Therefore, with this logic, each reactor schedule is filled with production stages in a homogeneous way, employing all the reactors, possibly, at the same time. When more than one product B_l is required on a single τ_d, the sale of the one with the longest reaction time t_{B_l} is the first to be addressed. Only after its fulfillment, the sales of the products with smaller t_{B_l} are tackled.

Here follows a simple example to better clarify the implemented procedure. Let us consider the 5 day sales plan for three types of products as reported in Table 1. We first scan the actual reactors activity and check which ones are available and which ones are operating. Let us assume that both R_1 and R_2 are currently busy in producing B_2 and B_3, respectively. Hence, the initial situation can be represented as $T_{R_1} > 0$, $T_{R_2} > 0$, $T_{R_3} = 0$. Depending on when the production in R_1 and R_2 is scheduled to finish, the corresponding stocks of B_2 and B_3 are updated. In this way we can consider the proper stocked amount of each product when checking for covering the sales. The first sale concerns B_2 on day 1: the production in R_1 is assumed to finish on day 1, so that the updated stock of B_2 is sufficiently high to cover $S_{B_2}^{\tau_1}$ and no new batch is scheduled. Referring to the example of Table 1, Figure 2 shows a simple diagram furtherly explaining the scheduling criterion. For the

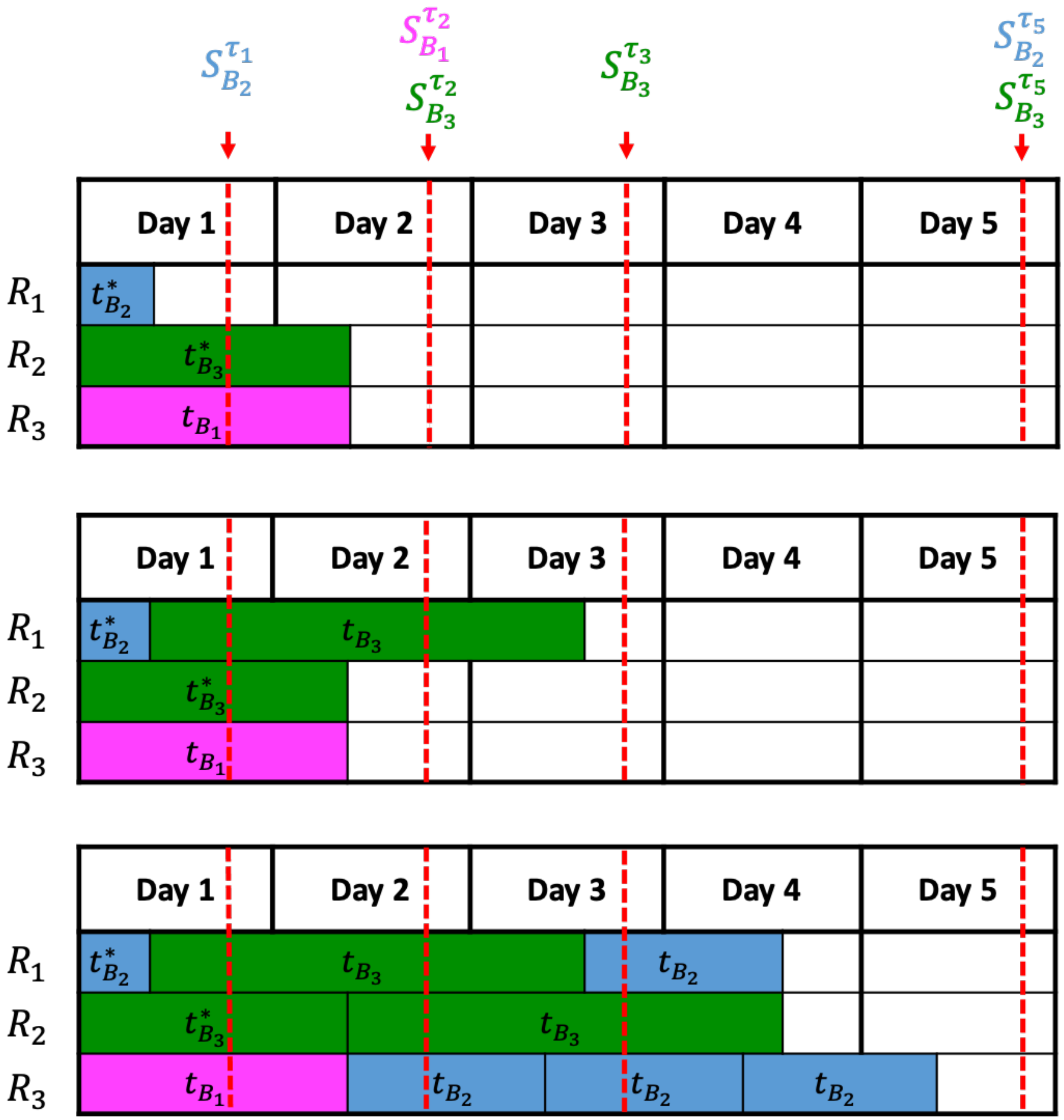

Figure 2: Scheme for reactor scheduling criterion. Times T_{R_r} are represented by the end of the solid box. The daily selling times τ_d are indicated by vertical red dashed lines on top of which the sales to be matched are evidenced. The asterisk on t_{B_l} identifies an ongoing production not yet finished when the optimization began.

sake of simplicity, only three reactors are here considered, i.e. $n_r = 3$. Moving to day 2 of the sales plan, two products are requested, B_1 and B_3, and, since $t_{B_3} > t_{B_1}$, we start from B_3. After checking its stocks updated with the production in R_2 finished on day 2, we assume that the product amount is enough to cover for $S_{B_3}^{\tau_2}$. Moving on to analyze B_1 stock, we assume that $S_{B_1}^{\tau_2} - \sigma_{B_1}^0 \leq W_{B_1}$ so that only 1 batch to produce B_1 is needed. In order to place the B_1 production, we check which reactor has the lowest reactor time t_R, hence we employ R_3 and update its time: $T_{R_3} = 0 + t_{B_1}$. On day 3 of the sales plan, we need (after checking its stock) to schedule another production of B_3 in order to cover $S_{B_3}^{\tau_3}$. As previously seen for B_1, we start by analyzing the reactor times and, consequently, placing the production in R_1, as shown in middle panel of Figure 2. Hence, the updated reactor times are: $T_{R_1} = t_{B_2}^* + t_{B_3}, T_{R_2} = t_{B_3}^*, T_{R_3} = t_{B_1}$. The last sale on day 5 requests more than one product, specifically B_2 and B_3. From stock calculation, four batch productions of B_2 and one of B_3 are needed. As done in day 2, we start by placing the one batch of B_3 (since $t_{B_3} > t_{B_1}$) in R_2, and update the reactor times. Only then, we place the first two batches of B_2 in R_3, the third one in R_1 and the fourth one again in R_3. Note that every time a batch is scheduled the reactor times t_R are updated and the procedure looks always for the smallest one. This is why the four batches of B_2 are scheduled in such alternated way (see bottom panel of Figure 2).

Finally, all the sales are satisfied if $T_{R_r} \leq \tau_d \; \forall \, d = 1, \ldots, n_d \wedge r = 1, \ldots, n_r$, otherwise an automatic message to the operator is sent. With the n_r reactor schedules completed, it is possible to calculate the "hourly production rate" of batch products x_P, and, consequently, evaluate their contribution to the hourly self-consumption function $a_j(\cdot)$ of other substances for the whole optimization horizon. This let us define $n_B n_h$ parameters used within the constraint set of the optimization problem.

3.3 "ON-OFF" switching procedure for production lines

In the real plant operation, each production line has evidently two operability modes: "ON", that is, the line is running within a range, between a minimum and a maximum production capacity, and "OFF" when the line is shut down for logistic or safety reasons. In such a framework, there are different ordinary and/or abnormal situations for which a line could be switched off. Obviously, for the production lines currently under maintenance, the problem to deal with is much simpler as it is only required to collapse the production capacity range to zero. Conversely, safety reasons are not to be predicted most of the time, that is, avoiding overfilling product storage tanks may be obtained in practice by switching off the corresponding line. In addition, in normal operations of the plant, there are periods along the year when the general production has to be reduced as a direct consequence of a lower market demand. However, in our problem formulation the production rates are bottom limited by a minimum value that for most of the products is strictly greater than zero. Hence, when a scenario in which the algorithm decides to impose minimum rates occurs, the stocks could still be overfilled due to the lack of sales. This is particularly crucial when the initial stocks σ_j^0 are quite high.

Hence, in order to represent the production lines behavior at best, several optimization variables should be in principle binary and not continuous within the optimization space. Moreover, since in this work we aimed at developing a tool able to handle quite large problems, this would have implied MINLP problems in several hundreds/thousands of variables, which cannot be efficiently tackled by off-the-shelf solvers. Therefore, in order to avoid such algorithm structure, a specific procedure to check whether some production line is to be shut down or not has been formulated as in Procedure 1.

For each product j, this procedure firstly acquires all the information about the sales plan vector (S_j), the initial stock (σ_j^0), the lower and upper stock bounds ($\sigma_{\min,j}$, $\sigma_{\max,j}$) and the lower bound for the production rate ($x_{\min,j} = [x_{\min,j}^0, \ldots, x_{\min,j}^i, \ldots, x_{\min,j}^{n_h-1}]$). Hence, the stock profile along the simulation horizon σ_j is calculated with the given $x_{\min,j}$. Then, if there is at least one time instant i^+ in which the stock exceeds its maximum bound (line 4 in Procedure 1) both the lower and the upper bounds on the production rate are set to zero from there to the end, as displayed in lines 5-6 of Procedure 1. This allows the production rate to be at zero and avoids an overload of the storage tanks. The reason why also the production rate upper bound ($x_{\max,j}$) is set to zero is twofold: firstly, we need to simulate a switched-off line, and secondly to avoid a non-zero production rate

Procedure 1 “ON-OFF" switch for production lines

Require: $S_j, \sigma_j^0, \sigma_{\min,j}, \sigma_{\max,j}, x_{\min,j} = [x_{\min,j}^0, \ldots, x_{\min,j}^{n_h-1}]$, $x_{\max,j} = [x_{\max,j}^0, \ldots, x_{\max,j}^{n_h-1}]$

Evaluate $\sigma_j(x_{\min,j})$
while not $\sigma_{\min,j} \le \sigma_j^i \le \sigma_{\max,j} \quad \forall i = 0, \ldots, n_h$ **do**
    for $i^+ = 0$ **to** n_h **do**
        if $\sigma_j^{i^+} > \sigma_{\max,j}^{i^+}$ **then**
            $x_{\min,j} = [x_{\min,j}^0, \ldots, x_{\min,j}^{i^+-1}, 0, 0 \ldots, 0]$
            $x_{\max,j} = [x_{\max,j}^0, \ldots, x_{\max,j}^{i^+-1}, 0, 0 \ldots, 0]$
        end if
        break
    end for
    Evaluate $\sigma_j(x_{\min,j})$
    for $i^* = 0$ **to** n_h **do**
        if $\sigma_j^{i^*} < \sigma_{\min,j}^{i^*}$ **then**
            Calculate $H_{rec} = \left\lceil \frac{\sigma_{\min,j}^{i^*} - \sigma_j^{i^*}}{x_{\min,j}^0} \right\rceil$
            $x_{\min,j} = [x_{\min,j}^0, \ldots, x_{\min,j}^{i^+-1}, 0, \ldots, 0, x_{\min,j}^{i^*-H_{rec}} \ldots, x_{\min,j}^{n_h-1}]$
            $x_{\max,j} = [x_{\max,j}^0, \ldots, x_{\max,j}^{i^+-1}, 0, \ldots, 0, x_{\max,j}^{i^*-H_{rec}} \ldots, x_{\max,j}^{n_h-1}]$
        end if
        break
    end for
    Evaluate $\sigma_j(x_{\min,j})$
end while
return $x_{\min,j}, x_{\max,j}$

lower than the original minimum. Once $x_{\min,j}$ is updated, it is applied to recalculate σ_j. This time we check if there is at least one time instant i^* in which the stock goes below its minimum bound (line 12). If this is case, from the time instant i^* the production line has to be switched-on again and original $x_{\min,j}$ and $x_{\max,j}$ have to be reinstated somehow. At this point, line 13 shows the calculation of the maximum number of hours H_{rec} needed to recover the missing stock $\sigma_{\min,j} - \sigma_j^{i^*}$, in which $\lceil z \rceil$ represents the ceiling operator applied to a real number z.

Therefore, the recalculated lower and upper bounds for the production rate are displayed in the lines 14-15. The procedure is iterative and stops when no issues about σ_j are found, i.e. $\sigma_{\min,j} \leq \sigma_j^i \leq \sigma_{\max,j} \; \forall i = 0, \ldots, n_h$. If no feasible configuration is found, an error is risen and the sales plan has to be reformulated. Clearly, this procedure actually applies only to those products with a non-zero $x^i_{\min,j}$. The final (eventually) recalculated $x_{\min}$ and $x_{\max}$ then enter into the optimization problem as decision variable bounds as illustrated in Section 3.4. Finally, note that Procedure 1 and the scheduling procedure for batch operations, described in Section 3.2, let us avoid binary variables that would have required a mixed-integer formulation of the optimization problem.

3.4 Optimization problem

The problem to be solved is a NonLinear Program (NLP) with the following general structure:

$$\min_{x} \; f(x) \tag{3a}$$

subject to:

$$x_{\min} \leq x \leq x_{\max} \tag{3b}$$

$$c_{\min} \leq c(x) \leq c_{\max} \tag{3c}$$

$$c_{eq}(x) = 0 \tag{3d}$$

in which $x \in \mathbb{R}^{n_x}$, $c_{eq}(x)$ refers to the material balance of n_{ns} non-storable products and to further nonlinear constraints that are better explained below, while $c(x)$ refers to bound constraints on stocks plus other process constraints. Non-linearity of the optimization problem derives from modeling refinements of some peculiar process dynamics. In particular, in order to avoid the case in which one piece of equipment is used to synthesize simultaneously two different products (j_1, j_2), an exclusivity constraint between two optimization variables is introduced:

$$0 = x^i_{j_1} x^i_{j_2} \quad \forall\, i = 0, \ldots, n_h \tag{4}$$

We underline that, as written, equality (4) violates constraints qualifications[30]. To this aim, the actual implementation of the exclusivity constraint is described by (5):

$$x^i_{j_1} x^i_{j_2} \leq \varepsilon \quad \forall\, i = 0, \ldots, n_h \tag{5}$$

where ε is a small real number (magnitude 10^{-7}).

Moreover, we also note that equality (4) can be reformulated into binary variables by introducing at least n_h additional variables, hence defining a MILP problem. Despite this enlarges the possibilities to explore for the future research by exploiting well-known solvers (eg. GuRoBi, CPLEX), this is actually out of the scope of the current work, that is, we maintain a NLP formulation.

The objective function $f(x)$ to minimize is continuous, linear in x and is defined according to the company needs, as detailed in Section 3.5.

Since sales misplacement can generate infeasible solutions, a smooth replacement for $f(x)$ in (3) is considered:

$$\min_{\xi} \; f(x) + \mu \left(\sum_i \overline{s}_i + \sum_i \underline{s}_i + \sum_i \overline{s}_{eq,i} + \sum_i \underline{s}_{eq,i} \right) \tag{6a}$$

subject to:

$$\xi_{\min} \leq \xi \leq \xi_{\max} \tag{6b}$$

$$c_{\min} - c(x) - \underline{s} \leq 0 \tag{6c}$$

$$c(x) - c_{\max} - \overline{s} \leq 0 \tag{6d}$$

$$-c_{eq}(x) - \underline{s}_{eq} \leq 0 \tag{6e}$$

$$c_{eq}(x) - \overline{s}_{eq} \leq 0 \tag{6f}$$

$$\overline{s}, \underline{s}, \overline{s}_{eq}, \underline{s}_{eq} \geq 0 \tag{6g}$$

in which

$$\xi = [x^T, \overline{s}^T, \underline{s}^T, \overline{s}_{eq}^T, \underline{s}_{eq}^T]^T, \quad \xi_{\min} = [x_{\min}^T, 0^T, 0^T, 0^T, 0^T]^T, \quad \xi_{\max} = [x_{\max}^T, \infty^T, \infty^T, \infty^T, \infty^T]^T \tag{7}$$

where ξ is the augmented decision variable; μ is a positive scalar penalty factor for the slack variables, assumed the same for all, for the sake of simplicity; ∞ is a vector of "infinity" and 0 is a vector of zeros. The slack variables $\overline{s}, \underline{s}, \overline{s}_{eq}, \underline{s}_{eq}$ are defined by the maximum deviation from the corresponding imposed constraint over the time horizon. Their dimensions are: $\overline{s}, \underline{s} \in \mathbb{R}^{n_P - n_B + n_{oc}}, \overline{s}_{eq}, \underline{s}_{eq} \in \mathbb{R}^{n_{ns}}$, where n_{oc} is the number of further process constraints. Thus, problem (6) is the one actually solved within the algorithm, and by construction it admits always a feasible solution. Furthermore, an initialization procedure for the *slack* variables has also been finalized in order to make the starting point always numerically feasible. This approach helps also in terms of reduction of computational costs. For this reason, a post-processing analysis of the optimization result is needed in order to verify if all the hard constraints are fulfilled, as detailed in Section 3.6.

Since the horizon length n_h is not a fixed parameter, it can be set also shorter in order to rerun a forecast that ends on the same day, but using updated parameters data. This is the case, for example, when the sales plan is changed along a week due to new client requests or sudden offer withdrawals. It can also happen that due to unexpected plant operation variations, some product stock values face significant changes that could not been taken into account during the forecast. For this reason, it is suitable to rerun the algorithm in order to obtain an updated optimal operation indication on a shorter horizon. In this way, a closed-loop like behavior of the algorithm can be tested offline at first, and then online directly via the DCS in the final phase of the project.

3.5 A multi-purpose objective function

As explained in Section 3.4, the objective function of problem (6) is based on the company needs, optimal practices and economic goals. To this aim, defining a single-purpose function would mean disregarding some key concepts. Hence, a multi-purpose objective function is considered. In particular, the different components of $f(x)$ are grouped into two main parts. The first one ($f_\sigma(x)$) is the summation of stocks of certain products at the end of the optimization horizon. This takes account of a specific plant strategy, that is, to have the minimum amount of certain key products in a specific period of the week, month or year. The second part ($f_{eco}(x)$) represents the economic expenses linked to the electrical energy consumption of the facility. Since the considered chemical processes are great consumers of electrical energy, analyzing the energy price variation along time allows one to encourage production at lower costs. The electrical energy prices, in Italy, can be found in the so-called National Unit Price (PUN) Index that gives hourly prices for the current day. Nevertheless, the optimization based on the PUN can only be performed on the first day considered, given the daily variability of the data and the impossibility of a reliable forecast for future days. In addition, an *ad hoc* procedure was also set up to pre-process the raw PUN data. According to the company specifications, the hourly PUN data have been divided into three groups of at least six hours each in order to limit the operational variation. This makes possible to identify three daily bands, each characterized by an average energy price and therefore to weigh accordingly the production of certain products during the day within the objective function. The procedure for identifying the three bands is automated by choosing as a criterion the minimization of the sum of the three variances. Nevertheless, it should be noted how the proposed optimization problem is still able to account for the hourly energy data, that is, the daily bands identification reflects a simplified approach of the main general one.

In this way, the formulated objective function can evaluate both the economic and practical feasibility aspects of the plant operations. To avoid problems due to the non-uniformity of the units of measure involved in the objective function, as the stock term is measured in tons, while the energy term (PUN) is usually expressed in €/MWh, a normalization is applied. Therefore, the final objective function is expressed as:

$$f(x) = \frac{(1-\alpha)}{\alpha_1} f_\sigma(x) + \frac{\alpha}{\alpha_2} f_{eco}(x) \tag{8}$$

where $\alpha \in [0,1]$ is a weight, chosen by the operator, that shifts the focus of the function to be more “energy-oriented" ($\alpha \to 1$) or more “storage-oriented" ($\alpha \to 0$), while α_1 and α_2 are the two suitable scaling factors.

3.6 Post-processing analysis

Given ξ^* the optimal solution of Problem (6), we need to check whether the values of the slack variables $(\bar{s}, \underline{s}, \bar{s}_{eq}, \underline{s}_{eq})$ are null or not. If, at least, one component of the slack variables is positive, one or more constraints along the weekly horizon is violated, that is, Problem (3) is not feasible. In this work, we consider two types of constraint violations: admissible or inadmissible.

The first category identifies the so-called *soft* constraints, the ones that when violated do not imply issues of safety or physical infeasibility. This is the case of non-critical products which, when missing, can be replaced by others without particular problems (e.g., by dilution or mixing of available products) or complaints from clients. The drawback of such product replacement can be a small economic loss, hence, even if these constrain violations are not harmful, they should be avoided or limited as much as possible. This is the reason why there is no distinction in constraints treatment in the algorithm itself, but just on the post-processing analysis of the optimization results. Therefore, when *soft* constraints are violated the operator still receives a warning as output message, but the sales plan can be left unchanged and the solution accepted.

Whereas, *hard* constraints are related to physical impossibilities or unsafe operations, that is, their violations are inadmissible. The levels of storage tanks represent the simpler example of this type of constraints. When the stock value overpasses the maximum limit, containers are spilling materials, that is, for sure a dangerous scenario. On the other hand, negative values of stocks simply are not picturing a real situation. Despite that, an additional threshold of 1 ton has been considered for violations of storage tank bounds, in order to avoid generating messages (flooding alarms) perceived as false alert state by the operator. Another alert scenario is when electrical devices are not working in the voltage ranges imposed by ordinary factory configuration. In these cases, the operator receives an error message, indicating which constraint(s) is (are) violated, and suggesting a change in the sales plan in order to obtain an acceptable solution.

Independently from the post-processing analysis, the final output communicated to the operator is threefold: the optimal solution of Problem (6), the stocks forecast along the optimization horizon, and, if present, the error/warning messages. As a matter of fact, in the current phase of the project, the algorithm is intended to work as a decision supporting tool in background mode, that is, the company operators always take the final decisions.

4 Industrial case study

An application example to real data and sales plan from Altair is now presented and discussed. Altair offers products and services for the inorganic chemistry and oenology industry, always taking into account process efficiency, energy saving, environmental sustainability and its renewability. A simplified process scheme is shown in Figure 3.

4.1 Case study description

The considered products, divided by category, are:

- 13 continuous products: $HCl^{(a)}$, $HCl^{(b)}$, $HCl^{(c)}$, $FeCl_3^{(a)}$, $FeCl_3^{(b)}$, NaClO, $NaOH^{(a)}$, $NaOH^{(b)}$, $KOH^{(a)}$, $KOH^{(b)}$, $KOH^{(s)}$, $K_2CO_3^{(aq)}$, $K_2CO_3^{(s)}$;

- 1 non-storable and non salable product: Cl_2;
- 3 batch products (chloroparaffins): Cl-Par$^{(a)}$, Cl-Par$^{(b)}$, Cl-Par$^{(c)}$.

Among the continuous-time productions there are some peculiarities. Three products, HCl$^{(a)}$, $FeCl_3^{(b)}$ and KOH$^{(b)}$, consist of two production lines each, i.e. $L_{\text{HCl}^{(a)}} = L_{\text{FeCl}_3^{(b)}} = L_{\text{KOH}^{(b)}} = 2$, where both lines contribute to the same storage tanks. Consequently, this implies two sets of n_h optimization variables for this kind of products. Moreover, the second line of $FeCl_3^{(b)}$ needs an external raw material to operate, that is, exhausted chloridric acid, E-HCl; therefore, the optimization variable $x_{\text{FeCl}_3^{(b)},2}$ has to satisfy the balance equation (1) with the provision of E-HCl as constraint. In addition, since E-HCl does not have a production rate associated, it is not part of the decision variables, but it is a parameter in Problem (6) and its stock values are included as inequality constraints. A proportionality factor (0.783) links $x_{\text{FeCl}_3^{(b)},2}$ with the consumption rate of E-HCl. In addition, the second production line of KOH$^{(b)}$ and the one of NaOH$^{(b)}$ use the same piece of equipment, an evaporator; therefore, these products cannot be obtained simultaneously as they need to fulfill the exclusivity constraint (4). This is the main reason for the NLP nature of Problem (6). Moreover, some chemicals shown in Figure 3 are not included in the optimization problem, since their consumption (NaCl, KCl, H_2O) or production (H_2) can be derived from the other substances considered. According to our notation, the number of variables we take into account is: $n_P = 20$, $n_{ns} = 1$, $n_B = 3$. Moreover, the reactors available for the batch products, are three, i.e. $n_r = 3$. The reaction times for the three chloroparaffins is the same, i.e. $t_{Cl\text{-}Par^{(a)}} = t_{Cl\text{-}Par^{(b)}} = t_{Cl\text{-}Par^{(c)}} = 31$h, and so it is their productivity per batch ($W_{Cl\text{-}Par^{(a)}} = W_{Cl\text{-}Par^{(b)}} = W_{Cl\text{-}Par^{(c)}} = 12$ t).

Given the structure of the optimization problem, it is clear how its overall dimension depends on the horizon length, from $n_x = 17 \times 168 = 2856$ for a 7 day optimization, to $n_x = 17 \times 24 = 408$ for 1 day forecast. In addition to the $n_P - n_B - n_{ns} = 16$ constraints on product stocks, further process and safety constraints ($n_{oc} = 7$) are to be considered. Hence, the total number of constraints along the optimization horizon ranges from over 1000 for a 1 day simulation to over 8000 for a week long forecast. Clearly, the computational cost of the optimization is also greatly dependent on the selected horizon length: from a couple of seconds for a 1 day simulation to 10-15 minutes

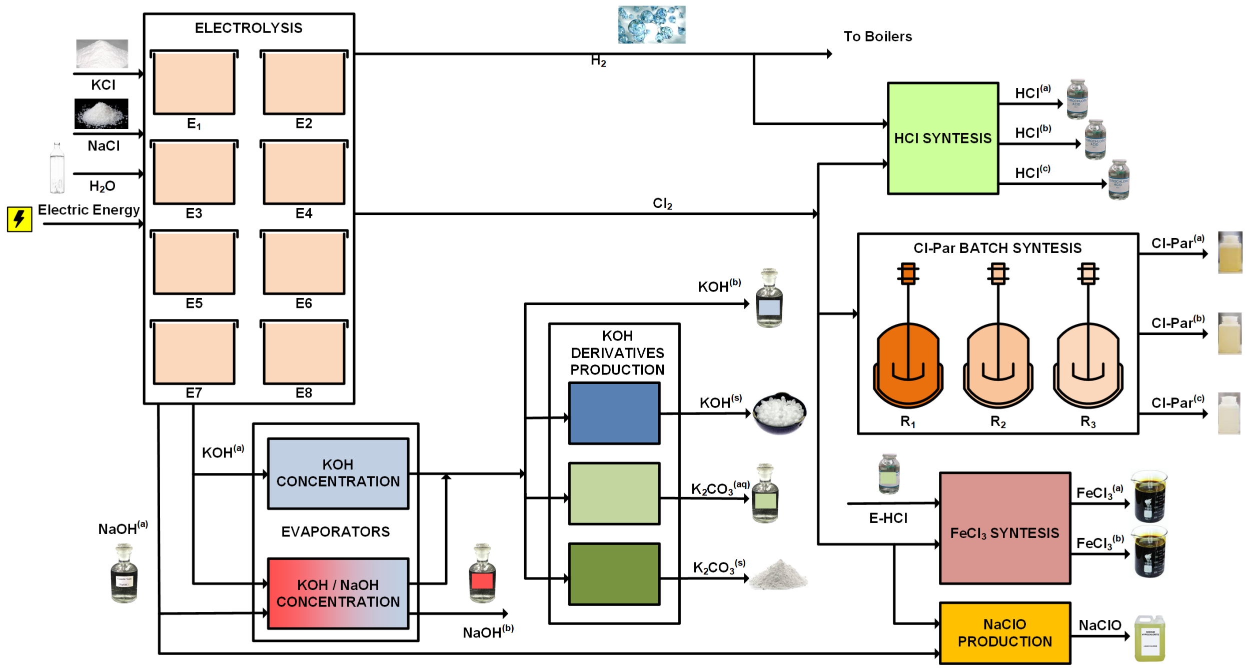

Figure 3: Simplified process scheme of the Altair case study.

for a week long forecast. More detailed examples can be found in Sections 4.2.2 and 4.3.2.

In order to handle the dimensionality and nonlinearity of the problem, the optimization algorithm has been equipped with a solver widely used and validated in the literature for large linear and non-linear programming problems, IPOPT[31], and a symbolic framework offered by CasADi[32]. The two parts of the selected objective function $f(x)$ are defined as follows:

$$f_\sigma(x) = \sigma^{n_h}_{\mathrm{HCl}^{(a)}} + \sigma^{n_h}_{\mathrm{HCl}^{(b)}} + \sigma^{n_h}_{\mathrm{HCl}^{(c)}} + \sigma^{n_h}_{\mathrm{NaClO}} + \sigma^{n_h}_{\text{E-HCl}} \tag{9}$$

$$f_{eco}(x) = \gamma \left(\overline{\mathrm{PUN}}^{I} \sum_{0}^{i_I} x^{(i)}_{\mathrm{KOH}^{(a)}} + \overline{\mathrm{PUN}}^{II} \sum_{i_I+1}^{i_{II}} x^{(i)}_{\mathrm{KOH}^{(a)}} + \overline{\mathrm{PUN}}^{III} \sum_{i_{II}+1}^{24} x^{(i)}_{\mathrm{KOH}^{(a)}} \right) \tag{10}$$

in which, the stock-oriented term $f_\sigma(x)$ includes the stocks of $HCl^{(a)}$, $HCl^{(b)}$, $HCl^{(c)}$, NaClO and E-HCl on the last day of the optimization horizon; while the economical term $f_{eco}(x)$ is the sum of three different sets of hourly production rates of $KOH^{(a)}$ on the first day of optimization, each weighted by the corresponding mean band price of PUN. The indices "*I*, *II*, *III*" in (10) represent the three periods of the day in which the PUN index is divided according to the procedure described in Section 3.5, $\overline{\text{PUN}}^z$ with $z = 1,2,3$ is the corresponding mean energy price, and, finally, γ is a conversion factor with dimensions $\frac{\text{MWh}}{\text{ton/h}}$. Let us underline how the definition of functions (9) and (10) is linked to a specific profit strategy defined by Altair on the basis of the last three years of productivity, inventory management and client order dispatch organization. Hence, other economic factors, as chemical prices or conversion factors, are not explicitly included.

In order to fully understand the complexity of the problem, some aspects need to be clarified. Chlorine Cl_2 is non-storable, thus non salable, albeit produced by some products and consumed by others, i.e. Eq. (2) becomes $x^i_{Cl_2} = a_{Cl_2}(x)^i$, $\forall\, i = 0,\ldots,n_h$. Its self-consumption function, $a_{Cl_2}(\cdot)$, has positive terms corresponding to those products that generates Cl_2 and negative ones for the chemicals which consume it. Mass balances and reaction stoichiometry allow one to calculate the specific constants used to link each term of $a_{Cl_2}(\cdot)$ to Cl_2 production rate.

As explained in Section 3.2, the batch products (Cl-Par) do not enter directly in the optimization problem. Nevertheless, since they are chlorine consumers, their contribution to $a_{Cl_2}(\cdot)$ needs to be calculated. The preliminary chloroparaffins production schedule, once defined, gives the number of reaction batches needed to satisfy the sales plan. This information, together with the known reaction time $t_{\text{Cl-Par}}$ required per each batch, allows the calculation of the chlorine requests schedule along all the optimization horizon. Therefore, taking into consideration only the effective reaction time, i.e., $t_{\text{Cl-Par}} - t^{serv}_{\text{Cl-Par}}$, the hourly consumption of Cl_2 is computed from mass balances.

Sodium and potassium hydroxide solutions ($NaOH^{(b)}$, $KOH^{(b)}$) are obtained by concentration from $NaOH^{(a)}$ and $KOH^{(a)}$, respectively; hence the self-consumption function plays also an important role into the mass balance equations of these products. The four products are still considered, stored and sold separately with different destinations, but their stock values are linked through function $a_j(\cdot)$.

In addition, the considered problem has three soft constraints: sales for missing $HCl^{(a)}$ can be covered by both $HCl^{(b)}$ and $HCl^{(c)}$ after dilution, while sales for missing $HCl^{(b)}$ can be covered only by $HCl^{(c)}$, still after dilution; a similar logic lets $FeCl_3^{(b)}$ (high-purity) to be sold directly as

$FeCl_3^{(a)}$ (low-purity) with a little profit loss. Apart from stock bounds and nonlinear exclusivity constraint, many other hard constraints are to be satisfied for these replacements to be feasible: sum of stocks of three concentration levels of HCl, sum of stocks of the two higher concentrated HCl ($^{(b)}$ and $^{(c)}$), sum of stocks of two qualities of $FeCl_3$. In addition, since $NaOH^{(a)}$ and $KOH^{(a)}$ are produced in electrolysis cell from NaCl and KCl, respectively, electrical bounds on working conditions have to be considered as well.

4.2 Case 1: receding horizon optimization

4.2.1 Case description

Since sales plan updates or unpredictable (eventually emergency) situations may happen and affect stocks level of certain products, it is a good practice to perform receding horizon optimization in order to follow the evolution of the plant conditions and apply more suitable control actions. In the considered example, one week is firstly optimized; then, from a 7 day prediction, the horizon is reduced to reach a one day ahead forecast, by moving ahead the starting time and keeping fixed the final one. As example and for synthesis purposes, Table 2 shows the initial stock values and the sales plans for the product $KOH^{(b)}$ for all the optimization horizons taken into consideration. Note that those in Table 2 are just part of all the parameters used in the optimization Problem (6) for each simulation.

Table 2: Initial stock and sales plans for the product $KOH^{(b)}$ along the week for different horizon lengths; $\sigma^0_{KOH^{(b)}}$ and $S_{KOH^{(b)}}$ are expressed in tons.

Horizon length	$\sigma^0_{KOH^{(b)}}$	$S^{\tau_1}_{KOH^{(b)}}$	$S^{\tau_2}_{KOH^{(b)}}$	$S^{\tau_3}_{KOH^{(b)}}$	$S^{\tau_4}_{KOH^{(b)}}$	$S^{\tau_5}_{KOH^{(b)}}$	$S^{\tau_6}_{KOH^{(b)}}$	$S^{\tau_7}_{KOH^{(b)}}$
7 days	630	13	0	299	182	143	104	130
4 days	700	–	–	–	182	143	104	130
3 days	600	–	–	–	–	143	130	130
2 days	570	–	–	–	–	–	104	143
1 day	540	–	–	–	–	–	–	130

The sales of $KOH^{(b)}$ are a typical example of how the sales plan can change during the same week. In order to be more visually clear, the sales in Table 2 span from day 1 to day 7 for all the optimization horizons. The horizon gets shorter while going down the table rows and each optimization uses $S_j^{\tau_1}$ as parameter, that is, the first day of the 1 day optimization corresponds to the seventh day of 7 day one. As the objective of this first example is to stress the receding horizon feature of the proposed algorithm, the chosen objective function to be minimized is fully stock-oriented, that is, $f(x) = f_\sigma(x)$ as $\alpha = 0$.

4.2.2 Results

Table 3: Optimization results. Initial and final values of objectives function, number of violated constraints ($n_{g,viol}(x)$) and computational times; *in* and *opt* represent the initial condition and optimal solution found by Problem (6).

	7 days		4 days		3 days		2 days		1 day	
	in	*opt*	*in*	*opt*	*in*	*opt*	*in*	*opt*	*in*	*opt*
$\Phi(\xi)$[ton]	1.7×10^6	102.9	1.8×10^6	117.5	1.7×10^6	117.3	1.6×10^6	397.5	2.8×10^5	109.4
$f(x)$ [ton]	43.1	96.16	15.04	96.16	132.1	98.2	206	107.3	153	109.4
$n_{g,viol}(x)$	430	27	315	26	177	19	104	**25**	31	0
t_s[s]	617		56.4		97.5		7.2		2.4	

The optimization results for the receding horizon example are summarized in Table 3. The main indices adopted to evaluate the performance of the proposed methodology are here illustrated. The first index $\Phi(\xi)$ represents the augmented objective function in (6a). Its initial values (*in*) are so high because of the initial values of slack variables that compensate for the different constraint violations. In general, also the optimized value (*opt*) of $\Phi(\xi)$ is not so small due to some residual, usually *soft*, constraint violations. As a matter of fact, it can be seen how the initial very large values of $\Phi(\xi)$ most of the times reflect into very small values of $f(x)$. On the contrary, the optimized value of $f(x)$ is more or less the same for the first three optimizations, while it is increasing for shorter horizons. This is mainly due to the fact that the objective function (9) consists of minimizing the stocks of specific products at the end of the horizon. Therefore, when a change in the sales plan of the products directly involved in (9) occurs, it may result into an initial stock higher than the one forecasted in the previous optimizations with longer horizons. In this case study, the product NaClO shows a sale reduction along the week, which implies increased residual stock values at the end of the horizon for the 2 day and 1 day optimizations.

The results obtained by each optimization are always numerically feasible due to Problem (6) definition, but different messages are produced. Only the 1 day optimization achieves a solution that is feasible also for Problem (3). As a matter of fact, $n_{g,viol}(x)$ is the total number of constraints violated for Problem (3) ($c(x)$ and $c_{eq}(x)$), and, as explained in Section 3.6, this number accounts for both *soft* and *hard* constraints. In particular, for all the optimizations but the 1 day one, there are two kinds of *soft* constraint violations: the first one signals that the stock of $HCl^{(a)}$ is under the minimum bound considered, while the second one alerts that also the stock of $HCl^{(b)}$ is under the same circumstances. However, the values of total stocks of HCl and sum of stocks of $HCl^{(b)}$ and $HCl^{(c)}$ are always acceptable. Only in one occasion, the 2 day optimization (in bold in Table 3), the lack of $HCl^{(a)}$ and $HCl^{(b)}$ cannot be recovered by dilution of $HCl^{(c)}$ and thus two *hard* constraints are not fulfilled. In order to better understand, the time trends of the stocks for the three dilution levels of HCl, the total stock and the sum of types $^{(b)}$ and $^{(c)}$ are shown in Figure 4. It can be seen how both $HCl^{(a)}$ and $HCl^{(b)}$ are missing at different hours, but only with the 2 day optimization (red line in Figure 4) the total stock and the cumulative stock $\sigma_{HCl^{(b)}} + \sigma_{HCl^{(c)}}$ are going under their minimum bound at the 144th hour (that is, the 21st hour in the 2 day optimization). In this case, being the initial stocks read directly by the DCS, the operator can communicate the algorithm result to the selling department and then request for a possible sale reorganization of $HCl^{(a)}$, $HCl^{(b)}$ or $HCl^{(c)}$ in order to have a feasible solution also for the 2 day optimization.

The last row of Table 3 includes the computation time t_s, comprehensive of the batch scheduling and optimization stages. Simulations are performed on a macOS, CPU 2.6 GHz Core i5 (I5-4278U), 8GB DDR3. It can be noted how the computation time drastically decreases by shortening the optimization horizon, spanning between 10 min for a 7 day optimization to less than 3 s when dealing with 1 day forecast. This is mainly due to the dimension variability of the problem and especially to the increase/decrease of the nonlinear constraints. Since a possible re-run of the algorithm may be necessary due to sales plan updates, it is important that, especially the short horizon optimizations, can be executed fairly quickly.

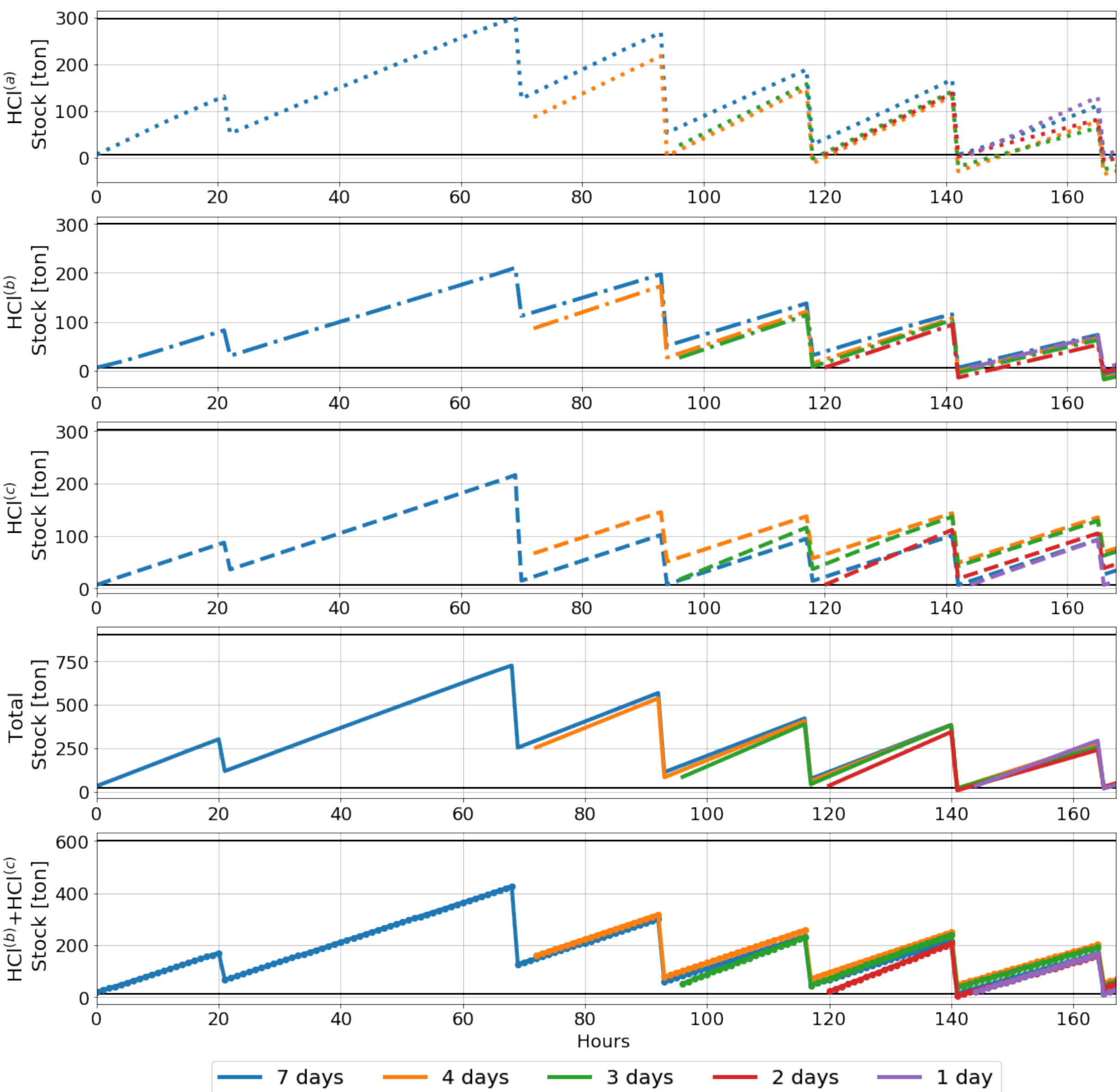

Figure 4: Stocks behavior for the three dilutions of HCl (first three panels) and their sum (last two panels).

Finally, Figure 5 shows the optimal trends, production rate and stock, for the product $KOH^{(b)}$. The corresponding initial conditions and sales plans are reported in Table 2. It can be observed

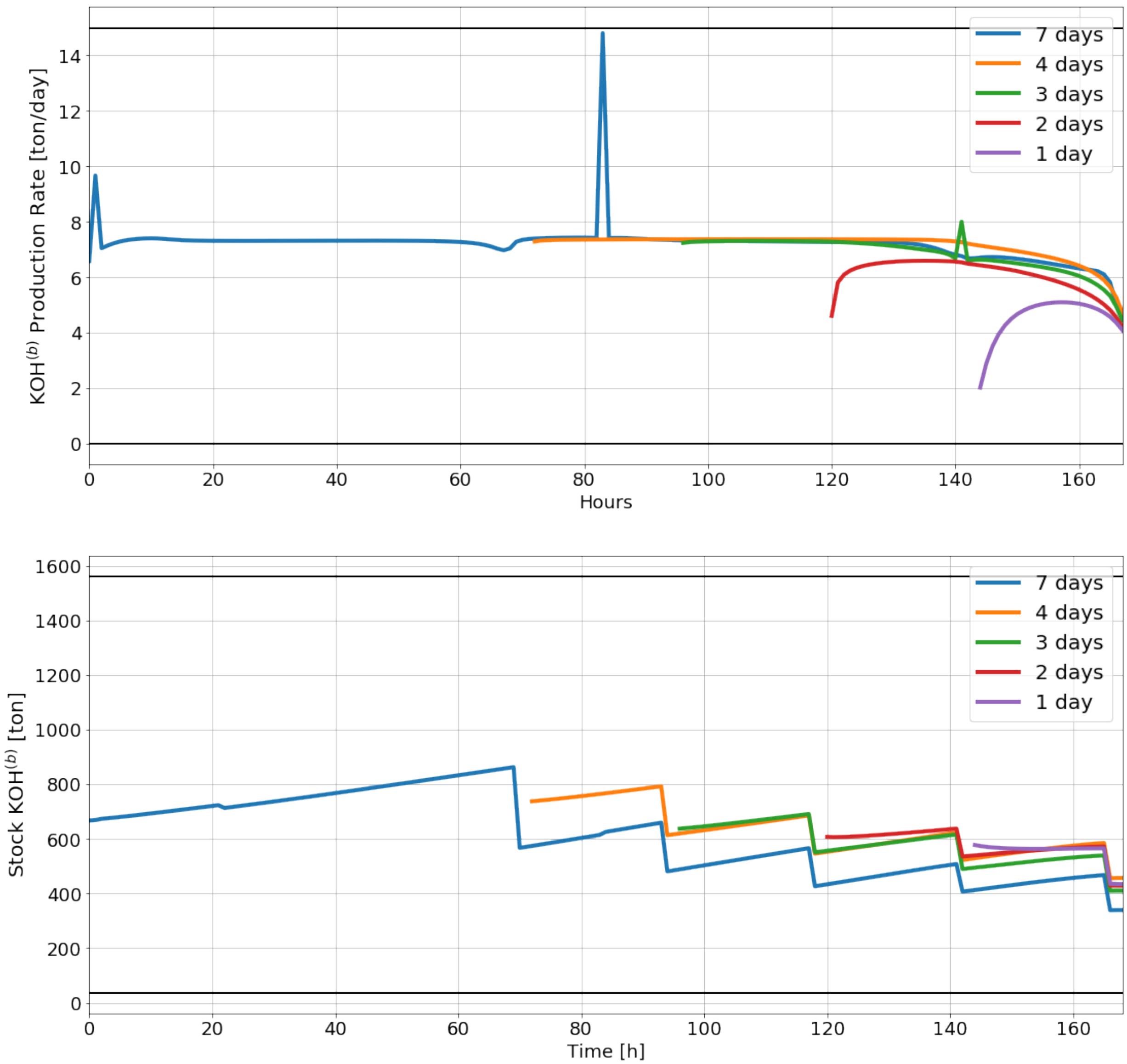

Figure 5: Production rate (top) and stock behavior (bottom) for the product $KOH^{(b)}$.

that a 3 day jump occurs between the 7 day and the 4 day optimizations, as the stock forecast at the hour 72 with the 7 day optimization is quite far from the actual initial stock of the 4 day one. The production rate is more or less constant for the 7 day optimization, while the 4 day ones has higher values due to the second line activity increase. Note that the spikes within the trends of production rates are due to numerical problems due to the exclusivity constraint on $x^{(i)}_{\text{NaOH}^{(b)}}$ and $x^{(i)}_{\text{KOH}^{(b)},2}$. Anyway, these are still not significantly impacting the stock value behavior as it is usually characterized by a saw-tooth shape, that is, stock time trend shows a cyclic behavior with a linear slow increase and then a sudden decrease as a corresponding sell occurs. Another aspect to note for the 1 day and 2 day optimizations, is a rounded profile of the production rate. This can be explained considering the actual initial stock for the considered two cases that is higher than the corresponding one calculated for longer horizons. This lets the production rate to be lower, so that the stock profile slowly increases until reaching the value needed to fulfill the last sale. In order to

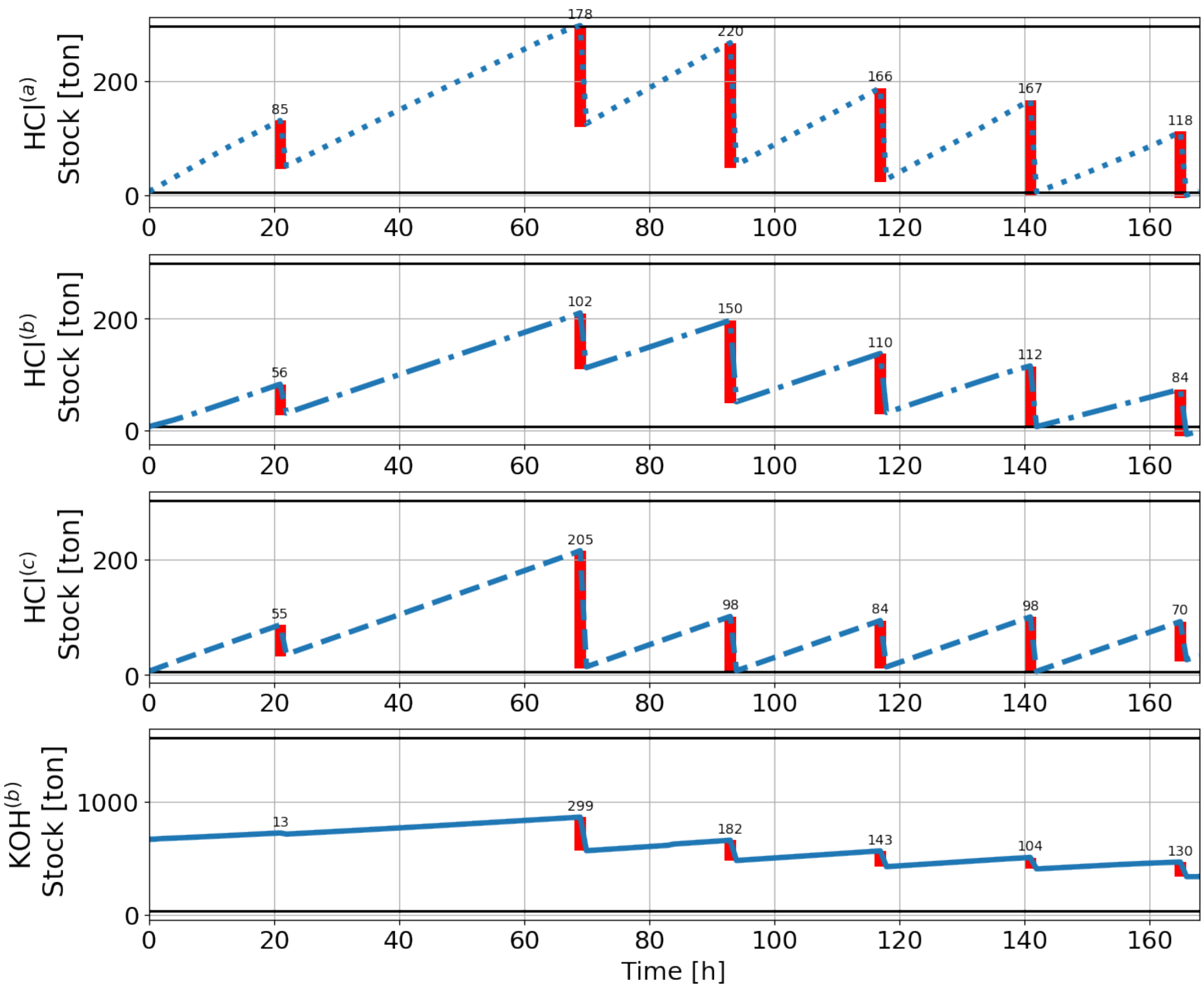

Figure 6: 7 day stock profiles from Figures 4 and 5 with red bars indicating the product demand per each sale. All the values are in tons.

make more clear the link between the sales plan and the stock profiles, Figure 6 shows the stock profiles seen in Figures 4 and 5 with the addition of red bars indicating the product demand per each sale. For the sake of clearness, only the 7 day results are plotted. When looking at Figure 6, the relation between the saw-thoot profile of the stock and the sale modeled to be accounted only on the selling time τ_d appears clear.

4.3 Case 2: multi-purpose objective function

4.3.1 Case description

The purpose of this second section is to better analyze what happens when a multi-purpose objective function is considered, that is, not only the stocks of certain products are minimized, but also the electrical energy costs due to the electrolysis reactions are taken into account. The horizon length optimization is now fixed and we intend to study how the different objective functions influence the algorithm outcome and performance. For the sake of simplicity and clearness, our focus is on the 3 day optimization. The PUN index considered and its division into three groups made by the automatic procedure explained in Section 3.5, are shown in Table 4. The three periods of the day have been identified in order to have the most distance between the mean prices of each group. The minimum number of hours per each group is six and, according to the values of the PUN given, the procedure has calculated to expand the central group to its maximum: 12 hours. Once three mean-levels of PUN are obtained, the multi-purpose function is defined as in (8), (9) and (10), and the scaling factors are set as $\alpha_1 = 100$ ton, $\frac{\alpha_2}{\gamma} = 7000$ €$\frac{\text{ton}}{\text{MW}}$. Different optimizations are thus performed varying the parameter α.

4.3.2 Results

Three values of α are here considered: 0, 0.5, 1. The optimization results for $\alpha = 0.5$ and $\alpha = 1$ are reported in Table 5. For the sake of comparison, the values for $\alpha = 0$ are taken from the 3 day column of Table 3. Note that the value of $f(x)$ for non-zero α is of two order of magnitudes

Table 4: PUN index values, division into groups and mean prices for each group.

Hour	PUN [€/MWh]	
12:00 AM	30.00	
1:00 AM	27.12	
2:00 AM	25.88	
3:00 AM	24.40	$\overline{\text{PUN}}^{I} = 26.48$ €/MWh
4:00 AM	24.43	
5:00 AM	27.04	
6:00 AM	30.00	
7:00 AM	32.91	
8:00 AM	32.94	
9:00 AM	32.98	
10:00 AM	30.43	
11:00 AM	28.46	
12:00 PM	27.65	$\overline{\text{PUN}}^{II} = 29.71$ €/MWh
1:00 PM	26.58	
2:00 PM	26.33	
3:00 PM	27.34	
4:00 PM	28.77	
5:00 PM	32.11	
6:00 PM	37.16	
7:00 PM	43.63	
8:00 PM	42.78	$\overline{\text{PUN}}^{III} = 39.87$ €/MWh
9:00 PM	42.00	
10:00 PM	39.82	
11:00 PM	33.82	

lower; this is due to the normalization and to the factors α_1, α_2. In addition, the number of violated constraints for the optimal solution is reduced with respect to the case for $\alpha = 0$. This is still due to the objective function composition and order of magnitude. However, all the constraints violated for both $\alpha = 0.5$ and $\alpha = 1$ optimizations are still the *soft* ones and do not impact the sales plan. Moreover, the final objective function value is always less than the starting one. This is because the economic part of the function is significantly decreased by the $KOH^{(a)}$ production rate update on the first day of optimization.

Table 5: Optimization results for $\alpha = 0, 0.5, 1$; *in* and *opt* represent the initial condition and optimal solution found by Problem (6).

	$\alpha = 0$		$\alpha = 0.5$		$\alpha = 1$	
	in	*opt*	*in*	*opt*	*in*	*opt*
$\Phi(\xi)$[-]	1.7×10^6	117.3	1.7×10^6	13.11	1.7×10^6	12.77
$f(x)$ [-]	132.1	98.2	0.99	0.84	0.65	0.50
$n_{g,viol}(x)$	177	19	177	18	177	13
t_s[s]	97.5		56.6		77.4	

In order to better understand this behavior, the trends of the production rates of $KOH^{(a)}$ are showed in Figure 7. It can be immediately seen how a non-zero α value is affecting the production rate of $KOH^{(a)}$, that is, the primary variable linked to the actual consumption of energy in the facility. Two optimizations with non-zero alpha have quite the same behavior in the first 24 hours and present a peculiar profile in this time lapse. As one would expect a lower production rate when a higher price is in force, the algorithm decides to do the opposite, by setting the rate at a value even higher than the one with “non-economic" function. As a fact, staying for 18 hours at a rate in the range $6.5-7$ ton/day and only for the remaining 6 hours around 10 ton/day gives a total cost of 3.5€ against 1.5€. One more explanation can be given looking at the production of chlorine in the first 24 hours. The zero value gives approximatively 136 ton of Cl_2 against around 115 ton and 113 ton for $\alpha = 0.5$ and $\alpha = 1$ cases, respectively. This has an influence on all the other products rates, but, since the material balance (2) for Cl_2 is always satisfied and no other constraints are violated, the algorithm behavior appears reasonable. Another peculiarity that can be observed in Figure 7 affects the second part of the plot. As a matter of fact, around from the 45th hour, the

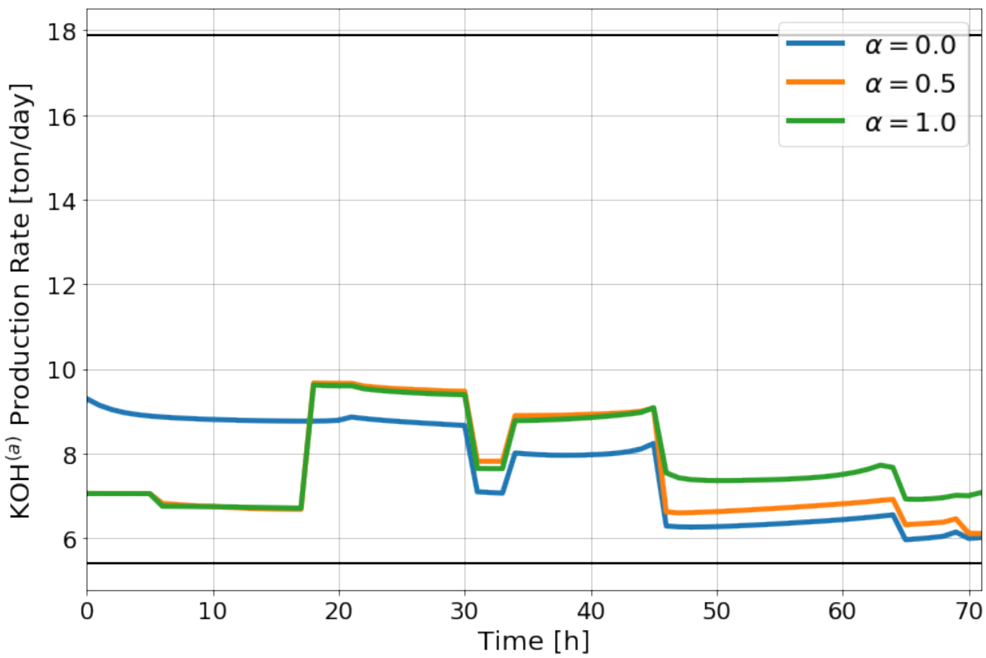

Figure 7: Production rate behavior for the product KOH$^{(a)}$ with different values of α. Lower and upper bounds are equal to 5 and 18, respectively.

trend for the $\alpha = 0.5$ optimization tends to the one with zero α. This is because, despite the interest in minimizing the electrical energy costs, the $\alpha = 0.5$ optimization still wants to tackle the stock minimization for the end of the horizon. As one should expect, a middle value of α reflects on an average behavior between the two extremes. This offers for sure an advantage when forecasting on short horizons, but it still can be useful for week-long prediction to understand how the plant would behave. Table 5 shows how this flexibility in the objective function does not reflect into an increase in the computational cost. On the contrary, the time employed for $\alpha = 0.5$ is almost half with respect to a “non-economic" function and one third less than the “pure-economic" one.

In order to better analyze the variability of the optimal objective function value, the Pareto distribution of the objective function for different values of α is presented in Figure 8. From the right panel of Figure 8, we can see how the the two parts of the objective function $f_\sigma(x_{opt})$ and $f_{eco}(x_{opt})$ are quite constant when varying α from 0 to 1. Hence, once a sales plan, initial stocks and a PUN vector are given, α itself represents the main contributor to $f(x_{opt})$ value. As a matter of fact, as evidenced in the left panel of Figure 8, the relation between $f(x_{opt})$ and α is an almost

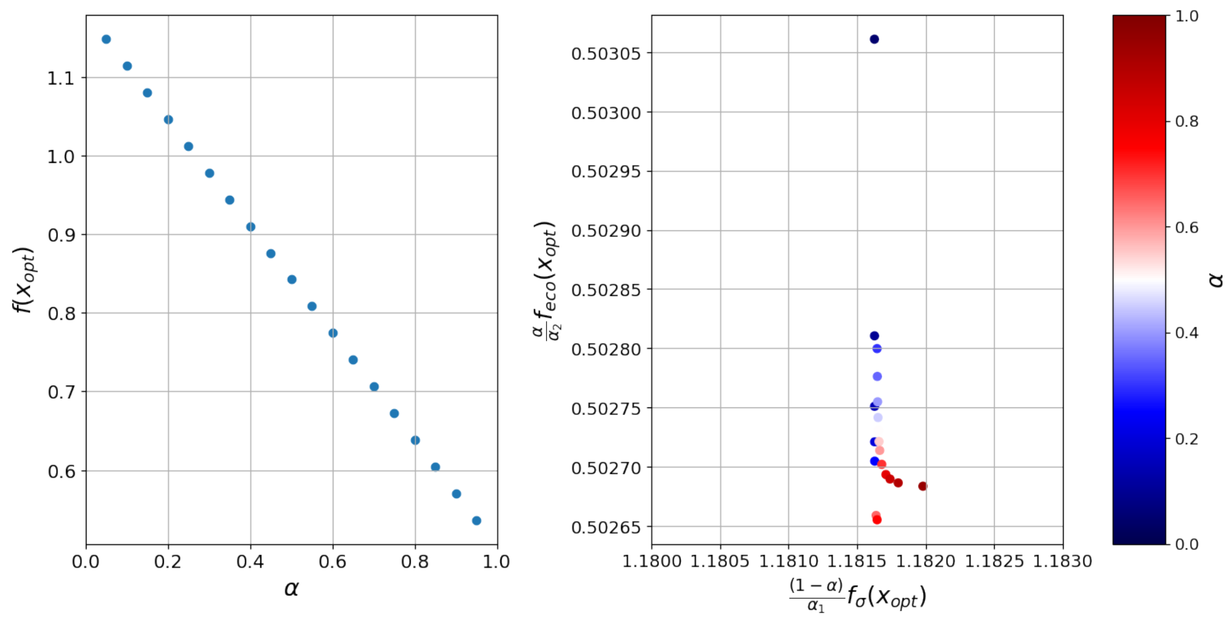

Figure 8: Pareto distribution of the objective function $f(x_{opt})$ (on the left) and its two parts $f_\sigma(x_{opt})$ and $f_{eco}(x_{opt})$ (on the right) varying α value from 0 to 1.

perfect negative linearity. As already explained above referring to Figure 7, for $\alpha \neq 1$, $f_\sigma(x_{opt})$ is the dominant component of $f(x)$ at the end of the optimization horizon, that is, just when the stock of the products involved in its formulation are computed. On the contrary, $f_{eco}(x_{opt})$ involves only the PUN prices relative to the first day of the horizon length. This fact implies very similar stock values at the end of the horizon for all $\alpha \neq 1$, which leads to an almost constant value of $f_\sigma(x_{opt})$. Nevertheless, as already shown by Figure 7 for $\alpha = 0$ and $\alpha = 0.5$, this does not mean an overlapping behavior of the production rate along all the horizon length, but only at its end.

4.4 Summary and outlook

Given the results obtained and discussion presented, it is important to remark that, the objective function is easily customizable to any new company request or interest and the output of the algorithm can be directly implemented into IT systems of the industrial site. Hence, even though the algorithm inputs arrive automatically from the DCS and the management system through a collection and store data framework, the final output of the RTO system is still analyzed offline by an operator. The two options available are: accepting the proposed solution and passing it to the control room; or communicating the algorithm result to the selling department in order to proceed for a possible sale reorganization. It should be noted that, in a future release, the RTO algorithm will be able to directly communicate with the management system and selling department. Nonetheless, field measurements and other process variables and performance indices that are currently been imported directly from DCS and stored in the above mentioned framework, are to be used not only to run the optimization algorithm, but also to modify and possibly adaptively correct the underlying model [26]. Finally, it is important to underline that the objective function is easily customizable to any new company request or interest.

5 Conclusions

A real-time optimization algorithm to best manage production rates based on the sales plan has been presented. This work is part of a larger project involving the integrated digitalization of an Italian industrial site according to Industry 4.0 paradigms.

The products considered are produced continuously or in batch reactors, can be stored and sold to clients or must be consumed in real-time by other processes. The proposed algorithm implements a preliminary scheduling procedure to deal with batch productions, so to avoid a mixed-integer optimization problem. A preliminary suitable scheduling criterion is defined and a corresponding procedure is developed. Once the best configuration is found, the batch production schedule is passed to the optimization algorithm as a parameter. Another preliminary procedure for setting the production lines switch-off is implemented and its mechanism illustrated. This allows one to avoid the use of binary variables inside the optimization problem. In order to obtain always a numerically feasible solution, a smooth version of a nonlinear problem has been formulated. For this reason, a general and widely used NLP solver is adopted. A multi-purpose function is implemented in the NLP in order to consider both best facility practices and energy costs linked to operations. A post-processing analysis of the optimal solution gives a feedback to the operator who can accept or reject the suggested decision plan.

The algorithm has been successfully tested over real data of Altair, an Italian inorganic chemical company. It has been shown how the possibility of applying a receding horizon approach and/or a multi-purpose optimization, give significant enhancements to the production scheduling and sales fulfillment. In this way, operators are helped in a demanding task otherwise manual, time-consuming and highly subject to errors, and process managers are helped in better planning plant operations. Nonetheless, the currently ongoing project, devoted to a full computerization and digitalization of the facility, finds its kernel in the presented RTO system, taking advantage of its high versatility and suitability to different plant conditions.

Acknowledgement

This work has been supported by Sviluppo Regione Toscana through the POR CREO FESR 2014 - 2020 funding program (GREEN FIELD PEAS, sub-project SODA4.0).

References

(1) Vaccari, M.; Bacci di Capaci, R.; Brunazzi, E.; Tognotti, L.; Pierno, P.; Vagheggi, R.; Pannocchia, G. Implementation of an Industry 4.0 system to optimally manage chemical plant operation. *IFAC-PapersOnLine* **2020**,

(2) Alrabghi, A. Simulation based optimization frameworks as key enablers for the transformation to industry 4.0. Proceedings of International Conference on Computers and Industrial Engineering, CIE. 2018.

(3) Uhlemann, T. H.-J.; Lehmann, C.; Steinhilper, R. The digital twin: Realizing the cyber-physical production system for industry 4.0. *Procedia Cirp* **2017**, *61*, 335–340.

(4) Udugama, I. A.; Gargalo, C. L.; Yamashita, Y.; Taube, M. A.; Palazoglu, A.; Young, B. R.;

Gernaey, K. V.; Kulahci, M.; Bayer, C. The role of big data in industrial (bio) chemical process operations. *Industrial & Engineering Chemistry Research* **2020**, *59*, 15283–15297.

(5) Bacci di Capaci, R.; Scali, C. A Cloud-Based Monitoring System for Performance Assessment of Industrial Plants. *Industrial & Engineering Chemistry Research* **2020**, *59*, 2341–2352.

(6) Cutler, C. R.; Perry, R. Real time optimization with multivariable control is required to maximize profits. *Computers & Chemical Engineering* **1983**, *7*, 663–667.

(7) Zavala, V. M. Real-time optimization strategies for building systems. *Industrial & Engineering Chemistry Research* **2013**, *52*, 3137–3150.

(8) Bonvin, D. Special issue "Real-Time Optimization" of processes. 2017; *Processes*.

(9) Pinto, J.; Grossmann, I. A Continuous Time Mixed Integer Linear Programming Model for Short Term Scheduling of Multistage Batch Plants. *Industrial & Engineering Chemistry Research* **1995**, *34*, 3037–3051.

(10) Pinto, J.; Joly, M.; Moro, L. Planning and scheduling models for refinery operations. *Computers & Chemical Engineering* **2000**, *24*, 2259–2276.

(11) Rejowski Jr., R.; Pinto, J. Scheduling of a multiproduct pipeline system. *Computers & Chemical Engineering* **2003**, *27*, 1229–1246.

(12) Alle, A.; Pinto, J. Global optimization for the cyclic scheduling and operation of multistage continuous plants. *Industrial & Engineering Chemistry Research* **2004**, *43*, 1485–1498.

(13) Burnak, B.; Katz, J.; Diangelakis, N. A.; Pistikopoulos, E. N. Simultaneous process scheduling and control: a multiparametric programming-based approach. *Industrial & Engineering Chemistry Research* **2018**, *57*, 3963–3976.

(14) Janak, S. L.; Floudas, C. A.; Kallrath, J.; Vormbrock, N. Production scheduling of a large-scale industrial batch plant. II. Reactive scheduling. *Industrial & Engineering Chemistry Research* **2006**, *45*, 8253–8269.

(15) Sundaramoorthy, A.; Maravelias, C. T. Modeling of storage in batching and scheduling of multistage processes. *Industrial & Engineering Chemistry Research* **2008**, *47*, 6648–6660.

(16) Biegler, L.; Zavala, V. Large-scale nonlinear programming using IPOPT: An integrating framework for enterprise-wide dynamic optimization. *Computers & Chemical Engineering* **2009**, *33*, 575–582.

(17) Biegler, L.; Yang, X.; Fischer, G. Advances in sensitivity-based nonlinear model predictive control and dynamic real-time optimization. *Journal of Process Control* **2015**, *30*, 104–116.

(18) Pontes, K. V.; Wolf, I. J.; EmbiruÃğu, M.; Marquardt, W. Dynamic real-time optimization of industrial polymerization processes with fast dynamics. *Industrial & Engineering Chemistry Research* **2015**, *54*, 11881–11893.

(19) Krishnamoorthy, D.; Foss, B.; Skogestad, S. Steady-state real-time optimization using transient measurements. *Computers & Chemical Engineering* **2018**, *115*, 34–45.

(20) Shokri, S.; Hayati, R.; Hayati, M. A.; Ayazi, M.; Ganji, H. Real time optimization as a tool for increasing petroleum refineries profits. *Petroleum & coal* **2009**, *51*, 110–114.

(21) Wang, X.; Mahalec, V.; Li, Z.; Qian, F. Real-time optimization and control of an industrial ethylbenzene dehydrogenation process. *Chemical Engineering Transactions* **2017**, *61*, 331–336.

(22) Hernandez, R.; Dreimann, J.; Vorholt, A.; Behr, A.; Engell, S. Iterative real-time optimization scheme for optimal operation of chemical processes under uncertainty: Proof of concept in a miniplant. *Industrial & Engineering Chemistry Research* **2018**, *57*, 8750–8770.

(23) Pataro, I. M.; da Costa, M. V. A.; Joseph, B. Closed-Loop Dynamic Real-Time Optimization (CL-DRTO) of a Bioethanol Distillation Process Using an Advanced Multilayer Control Architecture. *Computers & Chemical Engineering* **2020**, *143*, 107075.

(24) Vaccari, M.; Mancuso, G. M.; Riccardi, J.; Cantú, M.; Pannocchia, G. A Sequential Linear Programming algorithm for economic optimization of Hybrid Renewable Energy Systems. *Journal of Process Control* **2019**, *74*, 189–201.

(25) Wang, Z.; Wang, X. Multiobjective multifactorial operation optimization for continuous annealing production process. *Industrial & Engineering Chemistry Research* **2019**, *58*, 19166–19178.

(26) Badii, C.; Bellini, P.; Cenni, D.; Mitolo, N.; Nesi, P.; Pantaleo, G.; Soderi, M. Industry 4.0 Synoptics Controlled by IoT Applications in Node-RED. 2020 International Conferences on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData) and IEEE Congress on Cybermatics (Cybermatics). 2020; pp 54–61.

(27) Castro, P. M.; Erdirik-Dogan, M.; Grossmann, I. E. Simultaneous Batching and Scheduling of Single Stage Batch Plants with Parallel Units. *AICHE Journal* **2007**, *54*, 183–193.

(28) Nie, Y.; Biegler, L. T.; Wassick, J. M.; Villa, C. M. Extended discrete-time resource task network formulation for the reactive scheduling of a mixed batch/continuous process. *Industrial & Engineering Chemistry Research* **2014**, *53*, 17112–17123.

(29) Méndez, C. A.; Cerdá, J.; Grossmann, I. E.; Harjunkoski, I.; Fahl, M. State-of-the-art review of optimization methods for short-term scheduling of batch processes. *Computers & Chemical Engineering* **2006**, *30*, 913–946.

(30) Baumrucker, B.; Renfro, J. G.; Biegler, L. T. MPEC problem formulations and solution strategies with chemical engineering applications. *Computers & Chemical Engineering* **2008**, *32*, 2903–2913.

(31) Wächter, A.; Biegler, L. T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming* **2006**, *106*, 25–57.

(32) Andersson, J. A.; Gillis, J.; Horn, G.; Rawlings, J. B.; Diehl, M. CasADi: a software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation* **2019**, *11*, 1–36.

Graphical TOC Entry