Available online at www.sciencedirect.com
ScienceDirect
IFAC PapersOnLine 53-2 (2020) 11545–11550

Implementation of an Industry 4.0 system to optimally manage chemical plant operation

Marco Vaccari * Riccardo Bacci di Capaci * Elisabetta Brunazzi *
Leonardo Tognotti * Paolo Pierno ** Roberto Vagheggi **
Gabriele Pannocchia *,***

* Department of Civil and Industrial Engineering, University of Pisa, Italy
** Altair Chimica SPA, Saline di Volterra (PI), Italy
*** Corresponding author: gabriele.pannocchia@unipi.it

Abstract: The evolution of the process industry in the direction of automation and digitalization is nowadays a consolidated phenomenon. In this direction, Industry 4.0 paradigms are leading many industrial companies to significantly update their facilities. This paper presents a scheduling algorithm that takes the role of a real-time optimization (RTO) element in a larger project framework where the various network components are aimed to be all highly interconnected. The proposed methodology is applied to an Italian chemical industrial site, in order to best manage the production rates of the various products and the sales plan for the different clients. Numerous plants and processes are considered into the model: batch and continuous production lines, saleable and non-storable products. Concepts of linear optimization and batch operation scheduling are used in the algorithm construction. This whole structure lays the foundation for a full integration between different elements of the facility, that is, the control systems and the selling department.

Keywords: Process modeling; process optimization; scheduling; RTO; industry 4.0

1. INTRODUCTION

Process simulation and simulation-based optimization, in particular, play a key role within Industry 4.0 paradigms (Alrabghi, 2018), as they contribute to create what is called *virtual twin* of the physical process, i.e. a mathematical model able to accurately describe a process, a product or a service in order to perform specific analyses and apply suitable strategies (Uhlemann et al., 2017). Nevertheless, it should be noted that a simulation-based optimization approach is well-established in the academic and industrial worlds since last 30 years, under the name of Real-Time Optimization (RTO) (Cutler and Perry, 1983). This family of methods integrates process measurements into the optimization framework and does not rely exclusively on a (possibly inaccurate) process model, but also on process information obtained from measurements. Practical applications of RTO cover various fields of the process industry, as refineries, well networks, energy systems, combustion, batch operations (Bonvin, 2017).

Process scheduling can be considered a specific field of application of RTO methodologies, particularly devoted to optimization of large and complex industrial facilities. The typical objective is to solve short term scheduling problems of continuous and/or batch plants, that is, finding the optimal production policy to satisfy sales demands at a given time. Pinto and Grossmann (1995) treated batch plants, by applying a Mixed Integer Linear Programming (MILP) model over an LP-based branch-and-bound method to deal with the large scale problem. Also in the field of refinery operations, optimization-based algorithms and scheduling operation have been used to increase the annual profitability. Applied to a real case in Brazil, Pinto et al. (2000) firstly develop a model able to represent a general refinery topology, and define new operating points, more economically profitable. Then, a scheduling procedure is implemented using a MILP to deal with crude oil inventory management problem.

The computational problem of merging RTO and control with higher level decision-making has been considered by Biegler and Zavala (2009). The authors discuss about the possibility to realistically solve NonLinear Programming (NLP) problems on the order of a million variables. An example of RTO applied to operational optimization of energy systems has been presented by Vaccari et al. (2019). In this case, a Sequential Linear Programming (SLP) algorithm is used to generate an operating plan for each device in an energy system over a specified time horizon. The goal is to satisfy all electrical and thermal load requirements with possibly minimum operating costs.

The main objective of the present work is to build an RTO tool according to the paradigms of Industry 4.0. In particular, the developed algorithm aims at optimizing a set of production rates in order to minimize an economic objective function involving product stocks and to satisfy a complex sales plan.

The remainder of the paper is organized as follows. The problem definition is stated in Section 2; while the proposed optimization methodology is then formulated and illustrated in Section 3. All data, variables and constraints are defined, and a suitable preliminary scheduling for batch products is illustrated in this section. The algorithm is then tested over a real case study from an Italian inorganic chemical industry in Section 4. Results and discussions are here reported. Finally, Section 5 summarizes main achievements.

2. PROBLEM DEFINITION

The problem considered in this work is to model and optimally schedule the weekly production plan of an Italian industrial

10.1016/j.ifacol.2020.12.631

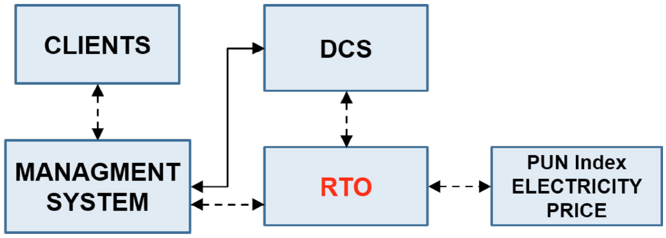

Fig. 1. Block diagram of the local computer system.

site of the inorganic chemical sector. This work is part of a larger 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.

The acquisition of production data will take place through a specifically developed dynamic connection between the DCS and the management system. An additional connection under definition will allow the management system to automatically receive customer orders, avoiding the manual entry phase, currently in place. The proposed RTO system will exchange input and output data with the DCS 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 shows a block diagram of the project architecture and identifies the position of the developed RTO system.

Various n_p products of interest of the company, named $A, B, \ldots$, etc., are considered in this work. The optimization problem starts from the weekly production plan designed by the operators of selling department on the basis of client requests. Let us name x_j with $j = A, B, \ldots$ the hourly production vector of product j along the week, i.e. $x_j = [x_j^0, \ldots, x_j^i, \ldots, x_j^{n_h-1}]$, where n_h is the total number of hours in a week, that is, $n_h = 24 \times 7 = 168$ h is the optimization horizon length.

Sales plans of each product are input data obtained from the selling department of the company and used within the problem as parameters. Defining the selling times of each day as $t_{s,d}$, with $d = 1, \ldots, 7$, we establish that a sale is satisfied if and only if the stock of the considered product j contains enough material at time $t_{s,d}$. Hence, the sales vector assumes this 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 $i = t_{s,d}$.

Stocks of each product are calculated 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 substance j as σ_j^0 and its evolution over time is obtained by mass balance as follows:

$$\sigma_j^{i+1} = \sigma_j^i + x_j^i - S_j^i - a_j(x)^i \quad \forall\, i = 0, \ldots, n_h \tag{1}$$

The quantity $\sigma_j^i \in \mathbb{R}$ depends linearly also on the function $a_j(x)$, named self-consumption, that is, some of the products are consumed within the industrial site to obtain other chemicals.

It has to be noted that some products cannot be stocked within the industrial site due to specific safety or logistic reasons. Since they cannot be sold either, they have to be consumed within the facility. Hence, their material balance (1) reduces to:

$$0 = x_j^i - a_j(x)^i \quad \forall\, i = 0, \ldots, n_h \tag{2}$$

Another important note is that some of the considered products are produced by means of batch reactors. This implies that the corresponding x_j can assume only a limited number of values. In particular, it is zero throughout most of the week and then assumes a certain positive value for a defined period of time. 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 the summation of stocks of certain ones while fulfilling all the various constraints. In the process control field, this would represent an RTO-level decision, since its main address is to communicate to operators the various set-points to be used in the advanced control layer, e.g. DCS.

3. PROPOSED METHODOLOGY

In this section, the various features of the method developed for optimizing the production plan are presented.

3.1 Data, variables and constraints

The hourly production rates of the various products are treated as optimization variables subject to bound constraints. Let optimization variable vector be $x = [x_A^T, \ldots x_j^T, \ldots, x_{n_p-n_b}^T]^T \in \mathbb{R}^{n_x}$, 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. Material balances (1) and (2), additional 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 of 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 said in Section 2, the company produces n_b different products in batch reactors. Let us name the different typologies of batch products as P, i.e. $P_1, \ldots, P_l, \ldots, P_{n_b}$. For each product P_l, a specific reaction time $(t_{P_1}, \ldots, t_{P_l}, \ldots, t_{P_{n_b}})$, comprehensive of service time $t_{P_l}^{serv}$, is considered. Each reactor produces the same amount W_P of P_l, that is, the corresponding "hourly production rate" can be calculated as follows: $x_{P_l} = \frac{W_P}{t_{P_l}}$ with $l = 1, \ldots, n_b$. Note that these hourly production rates are not considered as optimization variables in order to avoid to build a mixed-integer problem, where batch and continuous productions are simultaneously optimized. Therefore, a specific preliminary scheduling procedure for batch products has been developed.

Since there are n_r reactors, named $R_1, \ldots, R_r, \ldots, R_{n_r}$, in which product P_l can be produced, a criterion for scheduling is needed. The chosen one is based on the weekly sales of each P_l, as a sort of *the first needed is the first to be produced*. The procedure scans every selling times $t_{s,d}$ and registers each sale, then, depending on the stock value, it decides whether to produce the product P_l related to sale or not.

As example, let us consider the sales plan reported in Table 1 with three types of products. Since the first sale is on day 2 and P_1 is requested, we first check whether the current stock of P_1 is

Table 1. Example of sales plan for batch products. $S_{P_l}^{t_{s,d}}$ are tons of P_l requested by the client on day d.

	$t_{s,1}$	$t_{s,2}$	$t_{s,3}$	$t_{s,4}$	$t_{s,5}$	$t_{s,6}$	$t_{s,7}$
P_1	0	$S_{P_1}^{t_{s,2}}$	0	$S_{P_1}^{t_{s,4}}$	0	0	0
P_2	0	0	0	$S_{P_2}^{t_{s,4}}$	0	0	0
P_3	0	0	0	0	$S_{P_3}^{t_{s,5}}$	0	0

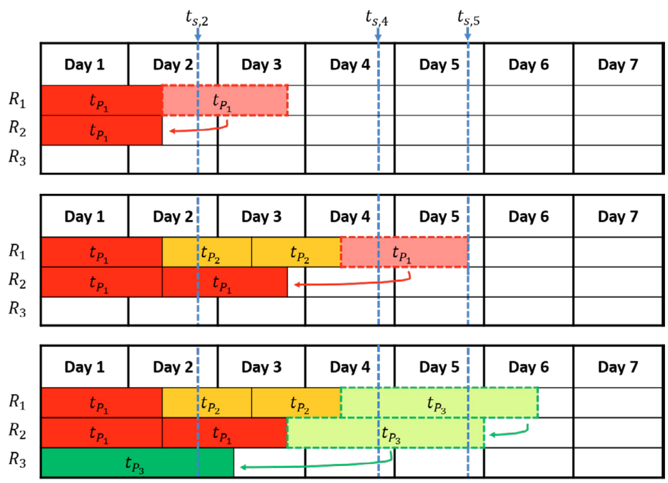

Fig. 2. Scheme for reactor scheduling criterion. The end of the dashed box indicates $t_{R_r,h}$, while the end of the solid box is t_{R_r}. The vertical blue dashed lines correspond to daily selling times $t_{s,d}$.

sufficiently high to cover for the sale. Otherwise, the production of P_1 is scheduled in the first reactor R_1. When the sale $S_{P_1}^{t_{s,2}}$ is greater than the quantity W_P produced in each reactor, multiple batches of P_1 need to be scheduled. In this case, there is a choice of setting a second production into R_1 or employing R_2. This decision implies to define another time variable, the reactor time $(t_{R_1}, \ldots, t_{R_r}, \ldots, t_{R_{n_r}})$, that is, the last time instant a reactor is used.

A diagram explaining the scheduling criterion is depicted in Figure 2. Note that Figure 2 is referring to the example of Table 1 and dealing, for the sake of simplicity, with only three reactors, i.e. $n_r = 3$. The reactor time starts from zero and grows depending on the production schedule for the considered reactor. Since the first production of P_1 is scheduled in R_1, the reactor time changes, i.e. $t_{R_1} = 0 + t_{P_1}$. When a second production of P_1 needs to be scheduled, the hypothetical reactor time $t_{R_1,h}$ is checked to be less than the sale time, i.e. $t_{R_1,h} = 2t_{P_1} \leq t_{s,2}$. If this holds true, than the new reactor time is $t_{R_1} = t_{R_1,h}$, otherwise another reactor is deployed, i.e. $t_{R_1} = t_{P_1}, t_{R_2} = t_{P_1}$, as shown in top panel of Figure 2. With this logic, each reactor schedule is filled with production stages until its reactor time exceeds a sale time. This approach allows one to operate always the same reactor, or reactors, leaving possibly the others unused.

When, instead, more than one P_l is requested on the same day, as in day 4 of Table 1, the first sale to be addressed is the biggest one, e.g., if $S_{P_2}^{t_{s,4}} > S_{P_1}^{t_{s,4}}$, than we start from $t_{R_1,h} = t_{R_1} + t_{P_2}$ and so on. Once all the n_r reactor schedules have been completed, the "hourly production rate" of batch products x_P is calculated, and then the corresponding contribution to the hourly self-consumption function $a_j(\cdot)$ of other substances for the whole week is evaluated. This fact is important as it counts for $n_b n_h$ parameters to be used within constraint set for the optimization problem. Finally, if the sales cannot be satisfied, i.e. $t_{R_1,h} > t_{s,d} \wedge \ldots t_{R_r,h} > t_{s,d} \ldots \wedge t_{R_{n_r},h} > t_{s,d}$, an automatic message to the operator is sent. In this case, in order to be conservative, the contribution to self-consumption of other substances is set at its maximum value for the all batch products.

3.3 Optimization problem

The problem to be solved is a Linear Programming (LP) problem and has the following general structure:

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

subject to:

$$x_{\min} \leq x \leq x_{\max} \tag{3b}$$
$$c_{\min} \leq c(x) \leq c_{\max} \tag{3c}$$
$$c_{eq}(x) = \mathbf{0} \tag{3d}$$

in which $x \in \mathbb{R}^{n_x}$, $c_{eq}(x)$ refers to the material balance of n_{ns} non-storable products, while $c(x)$ refers to bound constraints on stocks plus other process constraints. The objective function $f(x)$ is continuous, linear in x and is defined according to the company needs, e.g. to minimize the summation of stocks of certain products. It has to be noted that problem (3) can be easily rewritten in a standard LP formulation.

In order to better manage possibly infeasible solutions, i.e. due to sales misplacement, the following smooth replacement for $f(x)$ in (3) is considered:

$$\min_\xi \; f(x) + \mu \sum_i \bar{s}_i + \mu \sum_i \underline{s}_i + \mu \sum_i \bar{s}_{eq,i} + \mu \sum_i \underline{s}_{eq,i} \tag{4a}$$

subject to:

$$\xi_{\min} \leq \xi \leq \xi_{\max} \tag{4b}$$
$$c_{\min} - c(x) - \underline{s} \leq \mathbf{0} \tag{4c}$$
$$c(x) - c_{\max} - \bar{s} \leq \mathbf{0} \tag{4d}$$
$$-c_{eq}(x) - \underline{s}_{eq} \leq \mathbf{0} \tag{4e}$$
$$c_{eq}(x) - \bar{s}_{eq} \leq \mathbf{0} \tag{4f}$$
$$\bar{s}, \underline{s}, \bar{s}_{eq}, \underline{s}_{eq} \geq \mathbf{0} \tag{4g}$$

in which

$$\xi = \begin{bmatrix} x \\ \bar{s} \\ \underline{s} \\ \bar{s}_{eq} \\ \underline{s}_{eq} \end{bmatrix}, \quad \xi_{\min} = \begin{bmatrix} x_{\min} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \quad \xi_{\max} = \begin{bmatrix} x_{\max} \\ \boldsymbol{\infty} \\ \boldsymbol{\infty} \\ \boldsymbol{\infty} \\ \boldsymbol{\infty} \end{bmatrix} \tag{5}$$

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; $\boldsymbol{\infty}$ is a vector of "infinity" and $\mathbf{0}$ is vector/matrix of zeros. The slack variables $\bar{s}, \underline{s}, \bar{s}_{eq}, \underline{s}_{eq}$ are defined by the maximum deviation from the corresponding imposed constraint over the time horizon. Their dimensions are: $\bar{s}, \underline{s} \in \mathbb{R}^{n_p - n_b + n_{oc}}, \bar{s}_{eq}, \underline{s}_{eq} \in \mathbb{R}^{n_{ns}}$, where n_{oc} is the number of further process and safety constraints. Thus, problem (4) is the one actually solved within the algorithm, and by construction it admits always a feasible solution. For this reason, a post-processing analysis of the optimization result is needed in order to verify if all the hard constraints are fulfilled.

3.4 Result analysis

Once a solution is calculated, the values of the slack variables must be null for the problem (3) to be feasible. If, at least, one component of the slack variables is positive, one or more constraints along the weekly horizon is violated. We distinguish 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 or complaints from clients. Thus, a warning is sent to the operator, but the sales can be left as planned.

Inadmissible violations are related to *hard* constraints and to physical impossibilities or unsafe operations. The simpler example is the one referring to stocks reaching and overpassing the maximum limit, which means containers spilling materials. Another scenario is when electrical devices cannot work in different voltage ranges with respect to the one 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 weekly sales plan.

In any case, the optimal solution is communicated, together with the stocks and the error/warning messages, in order to be analyzed by the operator. As a matter of fact, in this first 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. APPLICATION TO INDUSTRIAL CASE STUDY

In this section, we present an application example to real data and sales plan from Altair. The products considered are:

- 12 continuous-time products: $HCl^{(a)}$, $HCl^{(b)}$, $HCl^{(c)}$, $FeCl_3^{(a)}$, $FeCl_3^{(b)}$, NaClO, $NaOH^{(a)}$, $NaOH^{(b)}$, $KOH^{(aq)}$, $KOH^{(s)}$, $K_2CO_3^{(aq)}$, $K_2CO_3^{(s)}$;
- 1 non-storable and non salable product: Cl_2;
- 3 batch chloroparaffins products: Cl-Par$^{(a)}$, Cl-Par$^{(b)}$, Cl-Par$^{(c)}$.

A simplified process scheme is shown in Figure 4. Note that some electrolysis reactants (NaCl, KCl, H_2O) and product H_2 are not directly included into the optimization problem, since not relevant for the company's purposes in this phase of the project. According to our notation, $n_p = 16$, $n_{ns} = 1$, $n_b = 3$; the number of reactors, identical in mass and productivity ($W_P =$ 12 t), available for the batch products is three, i.e. $n_r = 3$. The reaction times for three chloroparaffins is considered the same, i.e. $t_{Cl\text{-}Par^{(a)}} = t_{Cl\text{-}Par^{(b)}} = t_{Cl\text{-}Par^{(c)}} = 31$h.

Therefore, the overall dimension of the optimization problem is not negligible at all, being $n_x = 13 \times 168 = 2184$. The number of further process and safety constraints is $n_{oc} = 5$, in addition to the $n_p - n_b - n_s = 12$ constraints on products stocks. Hence, the total number of constraints along the optimization horizon is over 3000. For this reason, a solver widely used and validated in the literature for large linear and non-linear programming problems is adopted within the optimization algorithm: IPOPT (Wächter and Biegler, 2006). The selected objective function $f(x)$ is defined as the total stock of products $HCl^{(a)}$, $HCl^{(b)}$, $HCl^{(c)}$ on the last day of the week:

Table 2. Initial stock and sales plan for the week for each product; σ_j^0 and S_j are expressed in tons.

Product	σ_j^0	$S_j^{t_{s,1}}$	$S_j^{t_{s,2}}$	$S_j^{t_{s,3}}$	$S_j^{t_{s,4}}$	$S_j^{t_{s,5}}$	$S_j^{t_{s,6}}$	$S_j^{t_{s,7}}$
$HCl^{(a)}$	7	56	0	266	168	112	112	84
$HCl^{(b)}$	7	28	0	195	112	140	140	112
$HCl^{(c)}$	7	56	0	113	84	140	115	84
$FeCl_3^{(a)}$	60	0	150	84	140	84	150	0
$FeCl_3^{(b)}$	180	0	80	56	55	50	60	0
NaClO	50	0	0	0	28	28	0	28
$NaOH^{(a)}$	2	0	0	0	0	0	0	0
$NaOH^{(b)}$	20	0	0	40	45	45	45	0
$KOH^{(aq)}$	360	0	0	208	237	182	195	143
$KOH^{(s)}$	0	0	0	20	20	0	0	0
$K_2CO_3^{(aq)}$	87	0	0	12	0	12	12	0
$K_2CO_3^{(s)}$	0	0	0	40	40	0	0	0
Cl-Par$^{(a)}$	49	0	0	51	0	0	12	0
Cl-Par$^{(b)}$	20	0	0	31	0	13	0	0
Cl-Par$^{(c)}$	12	0	0	0	23	0	0	24

$$f(x) = \sigma^{n_h}_{HCl^{(a)}} + \sigma^{n_h}_{HCl^{(b)}} + \sigma^{n_h}_{HCl^{(c)}} \tag{6}$$

The definition of function (6) is linked to a specific profit strategy defined by Altair; hence, explicit economic factor are not included. Table 2 shows initial values of stocks and the sales plan for all the products to be satisfied during the analyzed week, i.e. all the optimization parameters used in Problem (4).

In order to fully understand the complexity of the problem, few 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) reads $x^i_{Cl_2} = a_{Cl_2}(x)^i$, $\forall\ i = 0, \ldots, n_h$. The self-consumption function $a_{Cl_2}(\cdot)$ has positive (negative) terms corresponding to those products that generate (consume) Cl_2. Each of these terms is linked to Cl_2 production rate by specific constants derived from mass balances and reaction stoichiometry.

The batch products (Cl-Par) do not enter directly in the optimization problem, as explained in Section 3.2, but since they are consumers of chlorine, their consumption has to be calculated. Once the preliminary scheduling procedure is completed, the number of reaction batches for each type of chloroparaffin, the time required for each reaction $t_{Cl\text{-}Par}$, and the total reaction time along the week, that is, the chlorine requests schedule, are known information. Then, the hourly consumption of Cl_2 can be calculated from mass balances, taking into account only the effective reaction time, i.e., $t_{Cl\text{-}Par} - t^{serv}_{Cl\text{-}Par}$.

The self-consumption function is important also for the relation between the two solutions of sodium hydroxide, $NaOH^{(a)}$ and $NaOH^{(b)}$, since the latter is obtained by concentration from the former one. These two products are considered, stored and sold separately with different destinations, but their stocks values are linked through a mass balance equation expressed within $a_j(\cdot)$.

As introduced before, there are two kinds of constraints, *soft* and *hard* ones. For this problem, we define three soft constraints as follows: $HCl^{(b)}$ can be sold after dilution to cover sales for missing $HCl^{(a)}$, $HCl^{(c)}$ can be sold after dilution to cover sales for both $HCl^{(a)}$ and $HCl^{(b)}$; $FeCl_3^{(b)}$ can be sold directly as $FeCl_3^{(a)}$ with a little profit loss. Note that many other hard constraints need to be fulfilled, as stock bounds, sum of stocks of three concentration levels of HCl, and of two qualities of

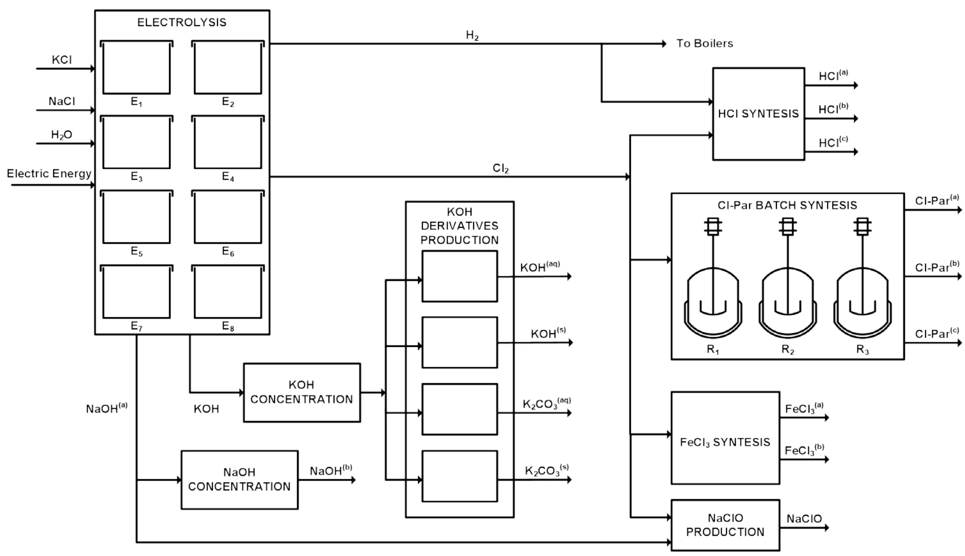

Fig. 3. Simplified process scheme of the Altair case study.

$FeCl_3$, as well as electrical bounds for production of $NaOH^{(a)}$ and KOH by electrolysis from NaCl and KCl, respectively.

The initial conditions for the different production rates give $f(x_0) = 18$ as initial value of the objective function, but two types of hard constraints are violated, i.e. the sum of stocks for HCl and $FeCl_3$ are negative at a certain time during the week.

4.1 Optimization Results

The computation time is around 80 s, comprehensive of the batch scheduling and optimization stages; simulations are performed on a macOS, CPU 2.6 GHz i5, 8GB DDR3. By definition of Problem (4), the result obtained is numerically feasible, but two messages are produced. The first one signals that the stock of $HCl^{(a)}$ is under the minimum bound considered, although the value of total stocks of HCl is always acceptable. Figure 4 shows the time trends of the stocks for the three dilutions of HCl and the total stock. It can be observed that, during the week, $HCl^{(a)}$ is missing for many hours, associated with four violations of the lower bound, but, in the meantime, $HCl^{(b)}$ and $HCl^{(c)}$ can overcome this lack by dilution. In other words, the first plot in Figure 4 shows how one of the soft constraints is violated, and the bottom panel shows how the corresponding hard constraint is still satisfied.

A similar scenario can be observed for the second warning emitted by the algorithm. The time trend of stocks of two qualities of $FeCl_3$ is shown in Figure 5. As a matter of fact, also $FeCl_3^{(a)}$ is missing in the last part of the week and, therefore, $FeCl_3^{(b)}$ has to be sold at its place. Note that the final value of the objective function is $f(x) = 39.1$. It is not surprising that the optimal value of the objective function is higher than the initial one, since initial conditions were infeasible.

As said before, in this first part of the project, only a background mode for the algorithm is considered. Hence, the final output of the RTO system is analyzed by an operator who has basically two options. He/she can accept the proposed solution, pass it to the control room and apply the weekly production plan as calculated. Otherwise, he/she can communicate the algorithm result to the selling department and then request for a possible

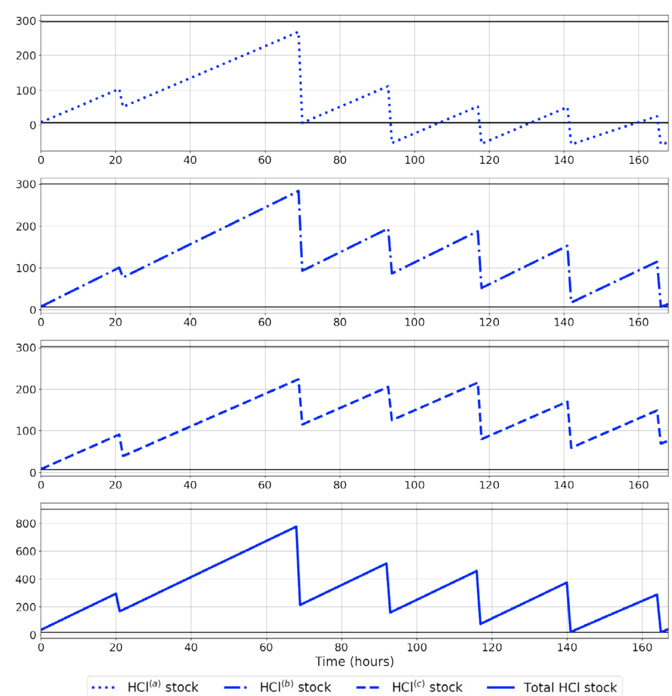

Fig. 4. Stocks behavior for the three dilutions of HCl (top and middle panels) and their sum (bottom).

sale reorganization, in the specific case, of $HCl^{(a)}$ and $FeCl_3^{(a)}$. For the previous example, as shown in Figure 5, a positive value for the stock of $FeCl_3^{(a)}$ at the end of the week, after a violation of the lower bound, is indication of an ill-positioning of the sales. If the sales plan is changed, the operator needs only to implement new parameters in a list as Table 2 and then re-run the algorithm. 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, and also read automatically the first-guess sales plan. Nonetheless, field measurements and other process variables and performance indices will be imported directly from DCS to run optimization algorithm and possibly correct adaptively the model. Finally, is important to underline that the objective function selected for this application can be changed and adapted to any company

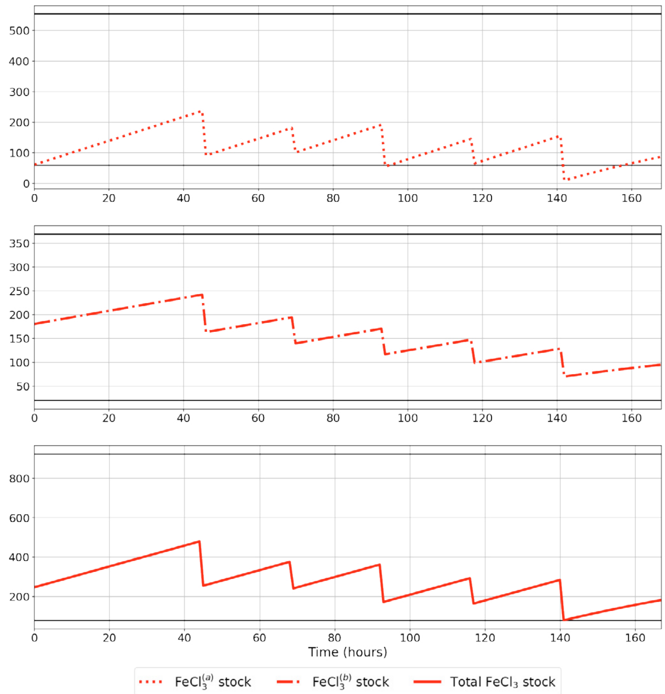

Fig. 5. Stocks behavior for the two qualities of $FeCl_3$ (top and middle panels) and their sum (bottom).

request. In particular, explicit economic factors can be easily implemented once hourly production rates are linked to a positive or negative cost term.

5. CONCLUSIONS

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

The considered products are of different nature, both continuous and batch, some storable to be sold and some others to be produced and consumed in real-time within the industrial site. To avoid a mixed-integer optimization problem, a preliminary scheduling procedure has been implemented for batch productions. Different batch products and reactors have to be considered, hence a specific scheduling criterion has been established and explained. The best reactor configuration gives parameters used into the optimization problem. A smooth version of a linear problem has been formulated in order to obtain always a numerically feasible solution. A general NLP solver is adopted since nonlinear features are to be incorporated in the next future. A post-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. In particular, it proves to give significant enhancements to the production scheduling and sales fulfillment, so that operators are helped in a demanding task otherwise manual, time-consuming and highly subject to errors. Nonetheless, its features of versatility and suitability to different plant conditions make the algorithm a key instrument in a full computerization and digitalization project of the company that is currently ongoing.

Possible future updates of the proposed algorithm could include a better performing approach for batch scheduling in order to avoid any backlog (Castro et al., 2007; Sundaramoorthy and Maravelias, 2008); possible integration between batching and continuous variables can be taken into consideration (Nie et al., 2014) as well as the analysis of the RTO behavior when subject to a rolling horizon in a closed-loop fashion (Subramanian et al., 2014).

ACKNOWLEDGMENT

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

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

Biegler, L. and Zavala, V. (2009). Large-scale nonlinear programming using ipopt: An integrating framework for enterprise-wide dynamic optimization. *Computers and Chemical Engineering*, 33(3), 575–582.

Bonvin, D. (2017). Special issue “Real-Time Optimization” of processes. *Processes*.

Castro, P.M., Erdirik-Dogan, M., and Grossmann, I.E. (2007). Simultaneous batching and scheduling of single stage batch plants with parallel units. *AICHE Journal*, 54(1), 183–193.

Cutler, C.R. and Perry, R. (1983). Real time optimization with multivariable control is required to maximize profits. *Computers & Chemical Engineering*, 7(5), 663–667.

Nie, Y., Biegler, L.T., Wassick, J.M., and Villa, C.M. (2014). Extended discrete-time resource task network formulation for the reactive scheduling of a mixed batch/continuous process. *Industrial and Engineering Chemistry Research*, 53(44), 17112–17123.

Pinto, J. and Grossmann, I. (1995). A continuous time mixed integer linear programming model for short term scheduling of multistage batch plants. *Industrial and Engineering Chemistry Research*, 34(9), 3037–3051.

Pinto, J., Joly, M., and Moro, L. (2000). Planning and scheduling models for refinery operations. *Computers and Chemical Engineering*, 24(9-10), 2259–2276.

Subramanian, K., Rawlings, J.B., and Maravelias, C.T. (2014). Economic model predictive control for inventory management in supply chains. *Computers & Chemical Engineering*, 64, 71–80.

Sundaramoorthy, A. and Maravelias, C.T. (2008). Modeling of storage in batching and scheduling of multistage processes. *Industrial and Engineering Chemistry Research*, 47(17), 6648–6660.

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

Vaccari, M., Mancuso, G.M., Riccardi, J., Cantù, M., and Pannocchia, G. (2019). A sequential linear programming algorithm for economic optimization of hybrid renewable energy systems. *Journal of Process Control*, 74, 189–201.

Wächter, A. and Biegler, L.T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1), 25–57.