REALISTIC RESCHEDULING: IS IT ACHIEVABLE?

Gabriela P. Henning
INTEC (Universidad Nacional del Litoral), Facultad de Ingeniería Química, UNL
Santa Fe, 3000, Argentina

Abstract

Predictive scheduling methodologies are aimed at addressing the generation of production agendas, while assuming that all activities and resources are given and there is no uncertainty in their behavior. However, experience demonstrates that there are intrinsically uncertain parameters and that industrial environments frequently face unexpected events that force schedulers to update agendas over time. After a brief review of the main proposals that have been developed to address the problem of uncertainty in production environments, the reasons why academic approaches are not implemented in practice are discussed. Then, an online scheduling framework is presented. It considers scheduling as an ongoing process in which evolving and changing circumstances continually force reconsideration and revision of pre-established plans with the goal of balancing stability and efficiency. Such framework is by no means complete and offers a roadmap of challenges and opportunities for future work.

Keywords

Rescheduling, Online scheduling, unforeseen events, Uncertainty.

Introduction

Scheduling is a very complex decision-making activity that takes place in process industries. It tries to make the best allocation of a set of limited resources to tasks over time, while optimizing an objective function or a combination of performance measures. In a batch industrial process, each processing task belonging to a product batch recipe requires certain amounts of specific resources during the associated processing time. Typical resources demanded by processing activities are equipment units, raw materials, intermediate products, manpower, and utilities, such as water, steam, electricity, etc. Similarly, storage tasks demand storage devices, material transfer activities require pumps, pipelines, manifolds, etc., and cleaning tasks need personnel, cleaning-in-place devices, etc. A great variety of aspects need to be considered when developing scheduling models, starting with an appropriate description of the production process and all the limited resources that would influence the agenda.

There is an overabundance of different methodologies that have appeared in the literature to address the scheduling problem. They have been recently reviewed by Harjunkoski et al. (2014). Most articles correspond to deterministic approaches in which all the parameters are regarded as known. In addition, all the operational features considered at the beginning of the scheduling horizon are not supposed to change during such horizon. However, in real plants, uncertainty is a very important concern that affects the scheduling process. In effect, many parameters, such as processing time, yields, stream quality, etc., which are employed to generate the agendas are inherently uncertain. In addition, production environments are dynamic; during schedule execution several unforeseen events, like equipment breakdowns, rush orders, late arrival of raw materials, lack of personnel, etc., may also affect the schedule. These uncertainties and disruptions generally cause the initial agenda to become infeasible and non-optimal. So, appropriate methodologies are needed to overcome the consequences of unexpected events and undetermined parameters. Despite the great practical interest of the problem not many proposals exist in the open literature. As a result, the dichotomy between the industrial and academic worlds that exists in the scheduling domain, is even stronger when rescheduling is focused. Following these introductory remarks, the next section is devoted to a brief literature review. The following one points out some of the reasons why the academic approaches have not made their way into the industrial domain. Finally, the last section presents an online scheduling framework that may help to make rescheduling activities attainable in practice. Within such framework there are many challenges and unsolved problems that are highlighted. They constitute an opportunity for researchers and industrial practitioners to work as partners.

Literature Review

According to Bonfill et al. (2008) two types of methodologies that explicitly deal with uncertainties in scheduling environments have been proposed: proactive and reactive scheduling. Proactive approaches incorporate the knowledge of uncertainty at the decision stage, prior to scheduling in order to generate resilient schedules that can absorb uncertainties. The schedules that are obtained exhibit an optimum expected performance, but they are not likely to be the optimum ones for the conditions that will actually take place. Li and Ierapetritou (2008a) provide a good review on scheduling under uncertainty that discusses proactive approaches in detail, starting with a description of various ways of representing uncertainty. The methods that have been proposed include robust optimization (Janak et al., 2007; Li and Ierapetritou, 2008b), stochastic optimization (Bonfill et al., 2005; Balasubramanian and Grossmann, 2004), fuzzy programming methods (Balasubramanian and Grossmann, 2003; Petrovic and Duenas, 2006) and multi-parametric programming (Ryu et al., 2007; Li and Ierapetritou, 2008c).

Reactive scheduling methodologies are implemented at execution time in response to disruptions or changes once they have occurred. Gupta et al. (2016) have recently presented a comprehensive review of this domain, in which the various meanings of the rescheduling term are discussed. Most of the contributions have mainly focused on two types of problems: (a) machine breakdown or changes in their performance that affect the associated processing times, (b) order changes: new arrivals, cancellations and modifications of product demands and due-dates. The purpose of the approaches is to update the agenda being executed to provide an immediate response to the unforeseen event. The earliest proposals (Cott and Macchietto, 1989; Kanakamedala et al., 1994) were based on algorithmic and heuristic procedures that resorted to time shifting and/or unit reallocation. Huercio et al. (1995) also presented heuristic-based methodologies to tackle problems of variations in task processing times and equipment availability. Their strategies were based on task shifting and the reassignment of tasks to alternative units.

More recent works addressed the reactive scheduling problem through mathematical programming approaches that mostly rely on Mixed Integer Linear Programming (MILP) formulations. Vin and Ierapetritou (2000) considered the rescheduling of multiproduct batch plants in the event of two types of disturbances: equipment breakdown and rush order arrival. Roslöf et al. (2001) proposed an MILP-based heuristic approach that improves a non-optimal schedule or updates the in-progress schedule of a facility having a single/critical processing unit. Rescheduling is performed by iteratively releasing and reallocating a small number of jobs from the current schedule. Computational complexity is controlled by limiting the size of the set of jobs to be reallocated into the schedule.

Méndez and Cerdá (2003; 2004) presented a user controlled MILP-based approach that repairs schedules in an iterative fashion by means of the restricted reallocation and reordering of processing tasks. Both contributions rely on a predictive scheduling formulation that was previously proposed by the same authors. Janak et al. (2006) proposed an MILP-based method to provide responses to unexpected events, such as equipment breakdown and addition/ modification of orders. To avoid full rescheduling, the approach identifies tasks which are not affected by the unforeseen event and that can be carried out as originally scheduled. The resulting tasks, along with additional subsets of tasks, are then fixed and the rest of horizon is rescheduled using an efficient MILP mathematical framework. In this way the computational load is reduced and the shop floor nervousness is prevented. Novas and Henning (2010) presented a repair-based reactive scheduling approach based on an explicit object-oriented representation of the domain and a constraint programming (CP) model. More recently, Kopanos and Pistikopoulos (2014) addressed reactive scheduling by resorting to a state-space model and multi-parametric programming in a rolling horizon framework.

An analysis of the literature, which is not complete due to space limitations, confirms that scheduling in the presence of unforeseen events has received limited attention. Not many surveys have been done in this area and very few works have tried to understand and organize domain knowledge, as well as to characterize problem elements. In this direction, it is worth mentioning the recent contributions of Gupta and Maravelias (2016) and Gupta et al. (2016) in the PSE domain. Vieira et al. (2003), Aytug et al. (2005) and Ouelhadj and Petrovic (2009) have previously made an effort to systematize knowledge. Though more oriented towards the manufacturing domain, Vieira et al. (2003) presented definitions and concepts appropriate for most applications of reactive scheduling and also described a framework for understanding rescheduling strategies, policies and methods. Aytug et al. (2005) have attempted to classify uncertainties in terms of the following four dimensions: *Cause*, *Context*, *Impact* and *Inclusion*. The first three are considered to be on the problem side, while the fourth is on the problem-solving rim. Ouelhadj and Petrovic (2009), have classified dynamic scheduling – according to the authors scheduling in the presence of an ample variety of real-time events – under four categories: *completely reactive scheduling*, *predictive-reactive scheduling*, *robust predictive-reactive scheduling* and *robust pro-active scheduling*. In *completely reactive scheduling* no firm agenda is generated in advance, decisions are made locally in real-time, and priority dispatching rules are frequently used. It is worth mentioning that other authors refer to this as dynamic scheduling. *Predictive-reactive scheduling* is one of the most common approaches. It is a process under which already existing schedules are revised in response to real-time events, without taking care of stability issues. To cope with the stability problem, *robust predictive-reactive scheduling* focuses on building agendas that simultaneously take into account both shop efficiency and deviation from the original plan. Finally, *robust pro-active scheduling* approaches focus on creating predictive schedules that satisfy performance requirements predictably in a dynamic environment. As seen, robust scheduling denotes different things to distinct authors. This proves that this is still an inmature field in which there is no common vocabulary yet. Even more serious is the fact that, up to now, academic contributions have addressed bit and pieces of the problem, lacking a comprehensive view.

Industry-Academia Gap

Despite recent advances, the deployment of scheduling solutions into industrial environments is still preliminary and has many open demands (Harjunkoski, 2016). Within the realm of scheduling approaches, the academic proposals that attempt to address rescheduling are ones of the less mature, preventing their adoption by industry. There are a variety of reasons for this gap, including the following:

1. Most contributions envision reactive scheduling in an isolated fashion that is performed on an “as needed basis”. Thus, their focus is on schedule generation/ repair for a fixed horizon, resulting on a shrinking horizon problem (Gupta and Maravelias, 2016) instead of a moving horizon one, as employed in most industrial settings.
2. All academic proposals adopt an overly simplified vision of the problem, by assuming that disturbing events are properly identified and correctly informed in real time. However, this is not generally the case. The same happens with the status of the active schedule at the rescheduling point. If this information is taken into account, which is not always the case, it is presumed to be available online, in a format that can be directly fed into the solution methodology. These strong hypotheses derive from the supposition of having a seamless relation between the scheduling tool and the applications (DCS, MES, CPM, ERP, etc.) that host the relevant data required to solve the scheduling problems. As pointed out by Harjunkoski (2016), this interplay does not occur in full fashion yet. In addition to the complications of having different time scales and information granularities, there are serious problems associated with the different domain representations that are employed, and which do not interoperate (Vegetti and Henning, 2015). It should be noted that academia has traditionally adopted problem representations that are well suited to the computational efficiency point of view, but which have not been adopted by industry.
3. Many different events and situations can trigger the revision of an agenda. Most academic approaches have focused on a very limited set of unexpected events, mainly processing units’ breakdown and order arrival/cancelation/modification. In consequence, they have left aside other types of disruptions, such as changes in the availability of different renewable (energy, steam, cooling water, manpower, etc.) and non-renewable resources (changes in the amount/date of arrival of raw materials) that are also important.
4. A common simplifying assumption is that only one disruptive event occurs at a time. Even if this happens, it is possible that other minor changes have affected the active agenda (e.g., small variations in the processing times) and need to be considered when revising the ongoing schedule. This issue calls for updated and continuous production progress data.
5. The great majority of the proposals omit the consideration of the *context* – the environmental situation at the time when something happens – and *impact* associated with the unforeseen events. In fact, for a given type of disruptive event in a certain plant, the reaction should not be always the same. For instance, a unit breakdown needs to be differently tackled when it occurs at the beginning of the planning horizon or when it takes place at the end. Very few contributions have focused on these issues (Janak et al, 2006; Novas and Henning, 2010) and more work needs to be done.
6. During the process of manufacturing, the ongoing agenda faces fluctuations in various attributes due to uncertainties, being also affected by unpredicted events. In order to keep the feasibility and efficiency of the current schedule, rescheduling actions are taken. However, frequent agenda revisions will also disturb the smooth operation of the shop floor, leading to plant nervousness. Very few authors have focused on the when-to-revise policies (Novas and Henning, 2010; Suwa and Sandoh, 2013) in order to provide appropriate guidelines. In addition, almost no contributions have addressed the timings of the schedule inspection and its associated decisions: a) Monitoring of the current schedule, b) Decision to conduct a schedule revision, c) Review execution, and d) Release of new schedule (Suwa and Sandoh, 2013).
7. Objective functions need to take into account both shop efficiency and deviation from the original agenda (stability). Therefore, efforts are required in order to propose suitable performance measures (Novas and Henning, 2010) and solution methods (Rangsaritratsameea et al., 2004) that help avoiding nervousness.
8. In order to be well accepted by industry, any reactive scheduling methodology needs to provide timely responses. Most contributions focus on CPU solution time but omit all the work needed to prepare the necessary models and to select proper parameter values. These models, which are not standard ones, and are context dependent, are manually built and tuned. This way of working, quite common in academia is not accepted in industry. Hence, the automatic development of rescheduling models is another requirement to be addressed. In addition, optimal agendas should be avoided if they entail too much CPU time, looking just for solutions of good quality.
9. The scheduler, who is the one that is finally responsible for the schedule, has to be directly involved in the solution methodologies (Harjunkoski,

2016). Solution approaches should envision him/her as an important actor playing at the different decision-making points during the ongoing solution process. Scheduling systems, specially the ones with reactive capabilities, should start to be designed as mixed-initiative optimization (MIO) systems. These types of decision-making tools are based upon collaboration between the system and the user, taking into account that both possess complementary abilities, which can be applied to complex optimization problems such as scheduling (Kirkpatrick et al., 2005).

10. Most solution approaches lack an explicit representation of the schedule (Novas and Henning, 2010). Having such domain model would allow implementing explicit reasoning procedures to: a) compare the planned schedule with what actually occurred at the shop floor, b) analyze disturbance context and impact, c) automatically setup the models to be solved, d) enable scheduler-system interactions by means of high-quality graphical user interfaces, etc.
11. Along the planning horizon, the initial predictive schedule S_o suffers modifications. At time point t_i, S_o is changed to cope with uncertainties, giving rise to an adjusted agenda $S_{[i]}$, which can be later modified at time point t_j, originating another revised schedule $S_{[j]}$. Each of these agendas has a part that has already been completed $S_{[i]}^{A}$, and another portion that corresponds to the predictive schedule resulting from the revision $S_{[i]}^{P}$. The process continues until reaching the end of the horizon; generally, with no explicit tracking of the evolution. Practice shows that there is a need for a logical way to organize and control revisions, associating each version with a time stamp, accounting for what was changed between versions, allowing the gathering of historical data for future decision-making and in order to carry-out production control activities.

Online Scheduling Framework

This section focuses on online scheduling (OS) presenting a framework for its development. The notion of OS is not new in the PSE community (Sand et al., 2000; Gupta and Maravelias, 2016). The proposed framework does not encompass a detailed technical description, but a research roadmap that points out the main challenges to be faced if rescheduling is to become reality in industry. Thus, the objective of this section is to encourage efforts in tackling several defies. Online refers to the state of "being in production" of a manufacturing system having an environment that dynamically changes from moment to moment due to the influences of internal and external factors. Therefore, it demands the constant monitoring and control of the progress of the shop floor, contrasting what is actually executed with what was planned. In addition, it also requires continuous communication with the various informatic applications (ERP, CPM, MES, DCS, etc.) that would inform changes in the status/availability of resources or modifications in the demand (new/modified/cancelled orders). In view of the changes, the online scheduling system should have the capability of revising and/or adjusting the existing agenda in real time, automatically, and/or with the aid of the scheduler, in order to cope with disruptions, unforeseen events and variations that have been observed. In consequence, OS entails several complex decision-making activities that need to be carried out in an integrated form.

Figure 1 shows a conceptual sketch of this framework. It makes explicit the various decision making activities associated with OS, as well as the main information flows. Close to each of the activities, the difficulties that were listed in the previous section, and that need to be addressed, are highlighted with small squares having the number of the causative issue. The success of the OS system very much depends on the already pointed out problems being considered in a comprehensive fashion.

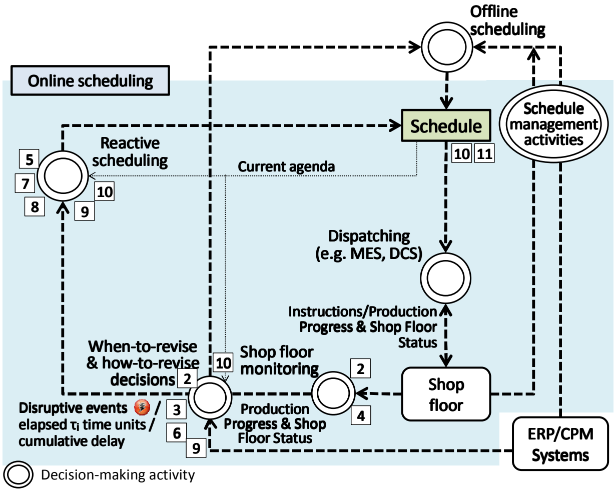

Figure 1:Decision-making activities and main information flows in an online scheduling framework

According to this framework, offline scheduling generates an agenda that is put into practice at the beginning of the planning horizon. To do so, the dispatching task takes care of sending detailed production instructions to the resources located at the shop floor in a timely fashion, based on the production progress and shop floor status. As uncertainties are materialized, the schedule revision must be carried out. Naturally, reactive scheduling is the main task of an online scheduler. However, when and how it takes place is by no means trivial. The when-to-revise and how-to-revise decisions depend very much on the type of manufacturing process, the features and context of the disruptions/ unforeseen events/variations that take place, as well as the characteristics of the current agenda. In consequence, these decisions cannot be taken a priori, and should be made online, on a case-by-case basis.

Figure 2 conceptually shows three of the most common when-to-revise policies. Under the *periodic policy*, schedule revisions are to be carried out on a regular basis, where the time τ between two revisions is pre-determined. Such parameter depends on the average processing times, the inherent uncertainty of the manufacturing environment, the length of the planning horizon, among other issues. Therefore, it needs to be properly chosen for each particular case and eventually modified, based on historical data. In the *event-driven* *policy*, the floor schedule is revised when a new critical event occurs. This policy requires the ability to automatically classify events into critical (e.g. a unit breakdown requiring two shifts to restore or an urgent job with a strict and close due date) and non-critical ones (e.g. a slight fluctuation in a changeover time or a small delay in the processing time); otherwise the agenda would be continuously modified, leading to shop nervousness. It can be seen that the event-driven revision policy can deal with emergencies as promptly as possible at the expense of managerial simplicity, compared with the periodic policy (Suwa and Sandoh, 2013). In real settings, schedule revisions are performed on a periodic basis, but in many situations the event-driven rescheduling is also triggered by certain emergent events, leading to hybrid policies, such as the *enhanced event-driven schedule revision* one (Suwa and Sandoh, 2013), shown at the bottom of Figure 2. This policy conducts a schedule revision immediately when a critical event occurs or when the elapsed time since the most recent revision reaches τ, whichever occurs first. Thus, it suits better the industrial needs but requires the automatic classification of critical and non-critical events and the adoption of a proper τ parameter value.

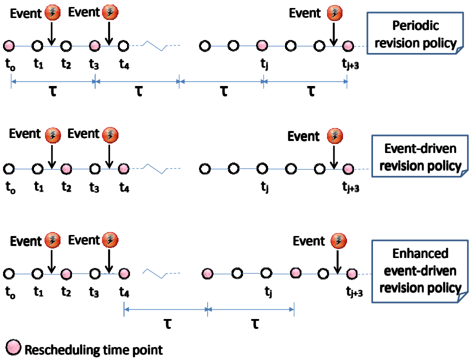

Figure 2: Three common when-to-revise policies

In addition to the previous policies Suwa and Sandoh (2013) have proposed the cumulative delay-based revision policy, which is conceptually shown in Figure 3. It focuses on the cumulative delays of the schedule as a criterion for the purpose of making a decision whether or not a schedule revision should be done. A Cumulative delay is considered consolidated information to assess if the current schedule is making smooth progress or not. If it exceeds a threshold *D**, it can be viewed as a sort of event triggering the revision. However, the cumulative delay estimation needs frequent monitoring, as well as a calculation requesting the comparison of the planned schedule with the actual one. Figure 3.a depicts a very simplified case of a one-stage process, in which it is supposed that at the current inspection time t_i tasks 6, and 9 were delayed due to the urgent task 12, which is the only unexpected event. Under such a situation, the total delay D_i over the $(t_{i-1}, t_i]$ period, being t_{i-1} the last inspection point, can be measured as $\delta_6 + \delta_9$, because these two tasks have finished during $(t_{i-1}, t_i]$. This example shows that only the tasks completing during the period were considered to count up their delays. From a managerial point of view, this approach is advantageous due to its simplicity. Nevertheless, for very time-consuming processing tasks, such as 10 in Figure 3.b, their delays would neither be measured nor be counted up for a long period of time. This simple example shows that if a cumulative delay policy is adopted for when-to-revise decisions, a more elaborated delay calculation method may be required. In addition, Suwa and Sandoh (2013) did not address delay calculations in complex manufacturing settings, such as multiproduct multistage or multipurpose batch plants with various interstage storage and wait policies, being this an open issue yet.

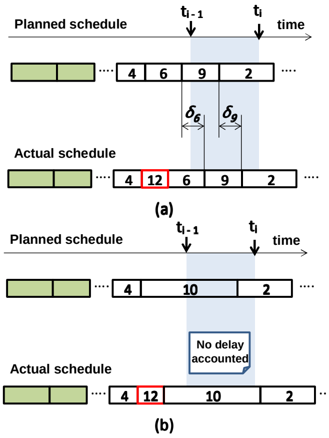

Figure 3: Cumulative delay-based revision policy. Examples of delay calculations

As for the how-to-revise policy, Figure 1 conceptually shows that if the unexpected event is a very disruptive one and it has a strong impact on the current agenda, the system and/or the scheduler can decide to revise the whole plan using any predictive scheduling performance measure. On the contrary, if a partial revision is considered, stability aspects would also guide the agenda amendment. Thus, the reactive scheduling activity would have to take into account that those tasks that are close to the revision point should not be modified unless it is strictly necessary. The previous paragraphs demonstrate that both the how-to-revise and when-to-revise policies call for an explicit representation of the planned and actual schedule, tracking also their continuous updates. Similarly, these decisions require the proper identification of critical events and the assessment of their predicted impact on the agenda.

Another open issue is concerned with the timing of the rescheduling activity, which is wrongly assumed to be instantaneous. As seen in Figure 4, once the current agenda is reviewed (schedule inspection point, triggered by an event or elapsed τ time units), the decision about whether or not to conduct a schedule revision based on the results of the review has to be made. As soon as it is decided to execute an agenda adjustment, there is an interval to gather information and to generate the new schedule. In addition, the revised agenda is not released to the shop floor immediately after it is obtained. As a result, there is a gap between the inspection and the release time points that needs to be considered in any solution methodology.

This contribution has shown that OS is a puzzling process with many intertwined decision making activities. Despite not presenting a complete catalogue of challenges (scheduling models were not addressed due to lack of space), several open issues that need industry-academia collaboration to be dealt with, have been emphasized.

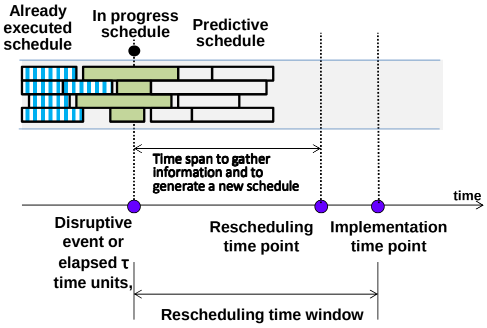

Figure 4: Schedule revision and its implementation point

Acknowledgments

The author wish to acknowledge the financial support received from CONICET (PIP 112 20110101145) and UNL (CAI+D 2011 50120110100473LI).

References

Aytug, H., Lawley, M. A., McKay, K., Mohan, S., Uzsoy, R. (2005). Executing production schedules in the face of uncertainties: A review and some future directions. *Eur J Oper Res*, 161, 86-119.

Balasubramanian, J., Grossmann, I.E. (2003). Scheduling optimization under uncertainty - An alternative approach. *Comput. Chem. Eng.*, 27, 469–490.

Balasubramanian, J., Grossmann, I. E. (2004). Approximation to multistage stochastic optimization in multiperiod batch plant scheduling under demand uncertainty. *Ind. Eng. Chem. Res.*, 43, 3695–3713.

Bonfill, A., Espuña, A., & Puigjaner, L. (2005). Addressing robustness in scheduling batch processes with uncertain operation times. *Ind. Eng. Chem. Res.*, 44, 1524–1534.

Bonfill, A., Espuña, A., Puigjaner, L. (2008). Proactive approach to address the uncertainty in short-term scheduling. *Comput. Chem. Eng.*, *32*, 1689–1706.

Cott, B.J., Macchietto, S. (1989). Minimizing the effects of batch process variability using online schedule modification. *Comp. Chem. Eng.*, 13, 105-113.

Gupta, D., Maravelias, C. (2016) On deterministic online scheduling: Major considerations, paradoxes and remedies. *Comput. Chem. Eng.*, 94, 312–330.

Gupta, D., Maravelias, C., Wassick, J. (2016) From Rescheduling to Online Scheduling. *Chem. Eng. Res. Des*. In Press.

Harjunkoski, I., Maravelias C., Bongers, P., Castro, P., Engell, S., Grossmann, I., Hooker, J., Méndez, C., Sand, G., Wassick, J. (2014). Scope for Industrial Applications of Production Scheduling Models and Solution Methods. *Comput. Chem. Eng.*, 62, 161-193.

Harjunkoski, I. (2016) Deploying scheduling solutions in an industrial environment. *Comput. Chem. Eng.*, 91, 127–135.

Huercio, A., Espuña A., Puigjaner, L. (1995) Incorporating on-line scheduling strategies in integrated batch production control. *Comp. Chem. Eng.*, 19, S609-S615.

Janak, S. L., Floudas, C. A., Kallrath, J., Vormbrock, N. (2006) Production scheduling of a large-scale industrial batch plant. II. Reactive scheduling. *Ind. Eng. Chem. Res. 45*, 8253-8269.

Janak, S. L., Lin, X., Floudas, C. A. (2007). A new robust optimization approach for scheduling under uncertainty. II. Uncertainty with known probability distribution. *Comput. Chem. Eng.*, 31, 171–195.

Kanakamedala, K.B., Reklaitis, G.V., Venkatasubramanian, V. (1994). Reactive scheduling modification in multipurpose batch chemical plants. *Ind. Eng. Chem. Res.*, 33, 77-90.

Kirkpatrick, A.E., Dilkina, B., Havens, W. (2005) A framework for designing and evaluating mixed-initiative optimization systems. Proceedings of WS3, Workshop on Mixed-Initiative Planning and Scheduling, In: ICAPS05, International Conference on Automated Planning and Scheduling, Monterrey, California, 5-11.

Kopanos, G.M., Pistikopoulos, E.N. (2014) Reactive scheduling by a multiparametric programming rolling horizon framework: a case of a network of combined heat and power units. *Ind. Eng. Chem. Res. 53*, 4366–4386.

Méndez, C.A., Cerdá, J. (2003). Dynamic scheduling in multiproduct batch plants. *Comput. Chem. Eng., 27*, 1247-1259.

Méndez, C.A., Cerdá, J. (2004). An MILP framework for batch reactive scheduling. *Comput. Chem. Eng.*, *28*, 1059-1068.

Li, Z., Ierapetritou, M.G. (2008a). Process scheduling under uncertainty: Review and challenges. *Comput. Chem. Eng.*, *32*, 715–727.

Li, Z., Ierapetritou, M.G. (2008b). Robust optimization for process scheduling under uncertainty. *Ind. Eng. Chem. Res.*, *47*, 4148–4157.

Li, Z., Ierapetritou, M.G., (2008c). Reactive scheduling using parametric programming. *AIChE J.*, 54, 2610–2623.

Novas, J.M., Henning, G.P. (2010), A Reactive Scheduling Framework Based-on Domain Knowledge and Constraint Programming",. *Comput. Chem. Eng*, 34, 2129-2148.

Ouelhadj, D., Petrovic, S. (2009) A survey of dynamic scheduling in manufacturing systems. *Journal of Scheduling*, 12, 417-431.

Petrovic, D., Duenas, A. (2006). A fuzzy logic based production scheduling/rescheduling in the presence of uncertain disruptions. *Fuzzy Sets and Systems*, 157, 2273–2285.

Rangsaritratsameea, R. Ferrell Jr., W.G., Kurz, M.B. (2004) Dynamic rescheduling that simultaneously considers efficiency and stability. *Comput. Ind. Eng.*, 46, 1–15.

Ryu, J.-H., Dua, V., Pistikopoulos, E. N. (2004). A bilevel programming framework for enterprise-wide process networks under uncertainty. *Comput. Chem. Eng.*, 28, 1121–1129.

Roslöf, J., Harjunkoski, I., Björkqvist, J., Karlsson, S., Westerlund, T. (2001) An MILP-based reordering algorithm for complex industrial scheduling and rescheduling. Comp. Chem. Eng. *25*, 821-828.

Sand, G., Engell, S., Miirkert, A., Schultz, R., Schulz, C. (2000) Approximation of an ideal online scheduler for a multiproduct batch plant. *Comput. Chem. Eng.*, 24, 361-367.

Suwa, H., Sandoh, H., (2013) Online scheduling in manufacturing: a cumulative delay approach. *Springer-Verlag*, London, UK.

Vieira, G., Herrmann, J., Lin, E. (2003). Rescheduling manufacturing systems: A framework of strategies, policies, and methods. *Journal of Scheduling*, 6, 39-62.

Vegetti, M., Henning, G.P. (2015) An Ontological Approach to Integration of Planning and Scheduling Activities in Batch Process Industries. *Comp. Aid. Chem. Eng., 37*, 995-1000.

Vin, J.P., Ierapetritou, M. (2000). A new approach for efficient rescheduling of multiproduct batch plants. *Ind. Eng. Chem. Res.*, *39*, 4228-4238.