19th European Symposium on Computer Aided Process Engineering – ESCAPE19
J. Jeżowski and J. Thullie (Editors)

Uncertainty, decomposition and feedback in batch production scheduling

Sebastian Engell

Process Dynamics and Operations Group, Department of Biochemical and Chemical Engineering, TU Dortmund, Emil-Figge-Str. 70, 44227 Dortmund, Germany
s.engell@bci.tu-dortmund.de

Abstract

In this paper, we consider production scheduling as a control problem under uncertainty. First the similarities and the differences between process control and online scheduling are discussed. A key issue in both contexts is the handling of uncertainty in feedback structures. It is advocated to use explicit models of the uncertainties in the form of scenario trees and to include the existence of feedback by solving two-stage models on rolling horizons. The interaction of planning and scheduling in the presence of uncertainty and feedback is discussed. The proposed model structure is a "telescopic" multi-scale model where the layers are coupled by targets for and feedback information about the states of the system.

Keywords: Online planning and scheduling, uncertainty models, multi-layer scheduling

1. Introduction

Only in rare cases, chemical production plants can be run at full capacity and under the same operating conditions, producing the same product over long periods of time. Usually, the throughput, the operating conditions, and also the products have to be adapted to changing demands, product prices, availability and prices of raw materials etc. The ability to react to these changing conditions and still to maintain a profitable operation is generally called flexibility. Flexibility comes at a price: a lower throughput increases the fixed costs, varying products cause changeover times, off-spec products, and a broad spectrum of products requires a broader set of equipment that is only partly used for each individual product. The more complex and specialized the products are and the smaller the volumes of each individual product, the higher the need for flexibility. The highest degree of flexibility is provided by multi-product plants, usually operated in batch or semi-batch mode. Recently, concepts for increased flexibility with respect to capacity and to the spectrum of products have also been proposed for continuous production, using modularized plants with intensified unit operations.

The economics of flexible plants are strongly affected by the efficient use of the available resources and by the ability to meet the demands of the customers with high-quality products that are delivered on time. The latter can always be achieved by either keeping a large stock of products or by providing large spare capacity, doing this exceedingly however contradicts the goal of profitable operations. Hence, as discussed in many papers on production planning and scheduling, the management or planning and scheduling of the available resources is a critical, but very complex task.

In recent years, remarkable progress has been made in the modeling of planning and scheduling problems by MILPs and their efficient solution (see e.g. Kallrath, 2002, Floudas and Lin, 2004, Castro and Grossmann, 2006, Castro et al. 2006, Ferrer-Nadal et al., 2007). Besides classical MILP techniques, other numerical approaches as e.g. evolutionary algorithms (Till et al., 2007, Sand et al., 2008) and automata-based methods (Panek et al., 2008) have also been proposed and demonstrated to be successful for small to medium-sized examples. These important developments provide the basis for the implementation of real production control systems, similar to the use of advanced continuous optimization techniques in RTO and model-predictive control, e.g. reported in (Janak et al., 2006a,b). When designing such systems, besides the representation of the problems at hand as static mixed-integer optimization problems and their solution, the problem formulation in a dynamic context must be addressed. This contribution tries to discuss communalities and differences of process control and online scheduling, in particular the representation of uncertainties, hierarchical decomposition, and rolling-horizon formulations in the context of feedback structures..

2. Process control vs. online scheduling – similarities and differences

Both conventional process control and online scheduling are reactive (and partly proactive) activities that have to cope with uncertainties and changes of operating conditions and targets and therefore employ feedback and feed-forward information structures. In (continuous) process control, the focus is on the control of *qualitative* properties of streams by changing the operating conditions of the plant. The resulting mass flows of raw materials or of products are usually prescribed externally. The challenge is to keep the quality parameters constant or to track time-varying set-points for these parameters, possibly under changing throughputs. The control of the mass flows as well as of the inventories is a less important problem that usually can be taken care of by low-level controls. Feedback control is mainly needed to handle the uncertainties involved, the uncertainty on the dependence of the process outcomes on the degrees of freedom that can be manipulated (inputs) – lack of models or *plant-model mismatch* – and the existence of external influences that influence the quality parameters and the economics of the process – summarized under the term *disturbances*. Process control is concerned with meeting the constraints on quality indicators, internal states (e.g. maximum pressures or temperature variations), and flows. Due to the size of plant-wide control problems, often a hierarchical decomposition as shown in Fig. 1 is employed.

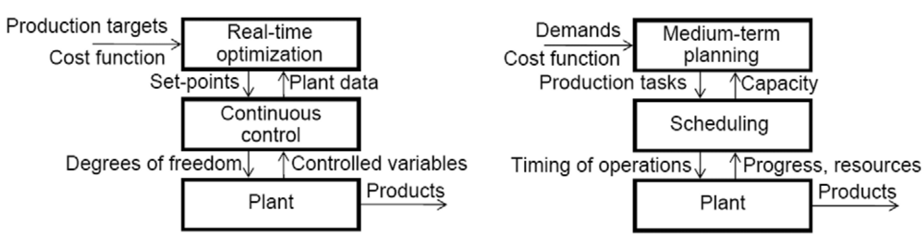

Figure 1: Hierarchical process control system and hierarchical planning and scheduling system

The so-called real-time operation (RTO) layer determines the set-points of the lower control layer by a steady-state optimization of the plant performance. The lower level implements these set-points in the presence of dynamic disturbances and deviations of the plant behavior from the model that is used on the RTO layer, to the best extent possible. The RTO layer performs its optimization based on accurate models of the steady-state behavior that are updated using measured data. On the lower level, usually linear MPC controllers or conventional PID controllers are used, and the discrepancy between the assumed linear models and the actual plant behavior is handled by feedback, an aspect that has been extensively discussed in the control literature under the term *robust control*. Recently, also nonlinear model-based controllers have been used on the lower level. As discussed in (Rolandi and Romagnoli, 2005, Engell, 2007), due to the advances in nonlinear dynamic optimization algorithms, in process control nowadays the option of a fusion of the two layers into one online optimizing feedback control layer is feasible and provides opportunities to improve the dynamic operation of the plant. This avoids several disadvantages of the layered approach, in particular a delayed reaction to disturbances due to the necessity to reach a new steady state before new set-points are computed and mismatch of the optimization criteria on both layers if optimization-based (MPC) control is employed on the lower layer.

In contrast to the situation in continuous control, the complexity of the problems does not admit a monolithic single-layer formulation of medium-term planning and scheduling for real-world problems.

In continuous process control, feedback is implemented based on measured or estimated values of (all or some) state variables of the process which are qualitative properties of the material along the flow through the plant. Inventory levels are allowed to vary to some extent in order to absorb some variability. In online scheduling, the quality indicators which are of predominant interest in continuous control are considered only in a discretized (abstracted) fashion: a batch production step is terminated successfully or not, in the latter case possibly causing the necessity of additional operations, it delivers certain amounts of products A, B, C, … and it requires certain resources for certain periods of time. The reference trajectories define mass flows of (discrete) products over time, averaged over certain intervals or as impulses at certain points in time (for fixed delivery dates), and the evolution of the inventories. Feedback and feed-forward control manipulate the timing of events – mainly the start times of production steps – and possibly the corresponding amounts of material (batch sizes). The state of the production process in scheduling is defined by the amounts of material stored, by the states of the resources (binary: operational or not, discrete: last operation was A, B, C, or continuous (rarely)) and by the progress of the running operations (discrete or continuous). From the scheduling point of view, the state (in the sense of process control) of the continuous parameters of the material and of the equipment only matters as far as it influences the durations of the operations and the amounts of material delivered upon termination. Consequently, the information obtained from the running processes in a feedback structure should be an accurate prediction of the expected finishing times and of the yields of the running batches.

Uncertainties in online scheduling are related to the availability of resources (breakdowns, lack of personnel), uncertain yields or unsuccessful production steps, uncertain durations of operations, and, often most importantly, dynamically changing demands or targets.

Both in process control and in online scheduling, the difficulty of the control problem is due to the "inertia" of the processes – the energies and masses stored in the plant in the case of continuous control and the inventories and the irreversible allocations of resources to processing steps in batch plants. The lower this inertia, the faster the production process can be adapted to changing market conditions. Agility roughly is the available range of the degrees of freedom relative to the inertia of the system. The smaller the inertia and the larger the range of the inputs, the faster transitions can be implemented. Translating this into online scheduling, the inertia is determined by the available resources relative to the work in progress and the associated blocking of resources, plus the inertia of the procurement of raw materials. The inertia of the plant and of the procurement of raw materials and other resources causes the need for planning on longer horizons. Planning can be understood as the generation of the reference trajectories for the plant and for the procurement of material, similar to the computation of the optimal operating conditions in process control.

In the standard approach to plant-wide control shown in Fig. 1, the upper layer performs an infrequent adaptation of the reference values for the low-level control system, and the inertia of the plant that prohibits an abrupt change of the controlled variables is taken care of by the lower layer, inevitably causing a transient period where the variables deviate from the set-points. In continuous plants with frequent product changeovers, the sequences of the changes of the set-points as well as the trajectories between these set-points can be included in the optimization to control the effects of the transient periods better, see e.g. [Marquardt]. Interestingly, in process control the variation of the controlled variables around their set-points has mostly been considered as natural and inevitable, and the idea to constrain quality indicators directly in the control algorithm rather than only reducing the variation around the set-points by optimal tracking is a relatively recent idea (Toumi and Engell, 2004, Rolandi and Romagnoli, 2005). In contrast, in planning and scheduling, targets given to the lower level are often considered as rigid, and the inability to meet these is often assumed to imply the need of a revision of the targets.

In online production scheduling, the need for longer-range planning is mostly caused by the interaction with upstream or downstream units (or the respective markets), i.e. the necessary procurement of the raw materials and the request for the delivery of products to customers at fixed times, in contrast to a production from and into large storage tanks. The task of medium-term planning is to synchronize the operations in the various units, to procure the necessary materials in time and to filter the demand variations by deciding how much material is produced to store, to match the temporal spread of the demands and the production orders in campaign production and by adapting the promised delivery dates of the orders according to the available capacities so that the lower level targets are realistic and reliable delivery dates can be promised to the customers. Tightly coupled to this is the planning of the inventories both of raw and of finished materials which increase the flexibility of the overall system in several ways: the available resources can be used more efficiently if the timing of the production leaves room for optimization, unplanned demands can be covered, and breakdowns or other problems do not cause delivery problems. On the other hand, large stocks of material cause cost because of the capital that is not active (an issue that sometimes is overestimated because it is only the interest on this capital that can be saved by lower stock levels) and, more importantly, by products that ultimately cannot be sold or of raw materials that are not used because of lack of demand.

The medium-term planning layer itself is confronted with a continuous stream of demands, of information on the actual progress of the production steps and of the actual sales. Moreover, if has to take dynamically varying conditions on the production level in account where the capacities of the units may vary both in a planned fashion and by unplanned shutdown, maintenance etc. Both layers, medium-term planning and scheduling are highly reactive and face considerable uncertainties in their planning data.

Similar to the hierarchical decomposition of process control tasks, the hierarchical structure shown in Fig. 1 (right) results. The communication between the layers and the formulation of the corresponding optimization problems is discussed in more detail in section 4 below.

Both in continuous control and in online production scheduling, model abstractions are used on the higher level of the hierarchy. As mentioned above, in control, the abstraction employed usually consists of neglecting the dynamics or the inertia of the plant. This can be seen as an extension of the time horizon to infinity where the small transient periods can be ignored. On the other hand, the models on the upper layer can be more detailed, often rigorous models obtained from first principles, while on the lower layer linear approximations, often obtained from data, are used. In scheduling, abstraction is performed similarly by aggregating demands and production over time (slots, buckets) and by neglecting resources and production steps that are not critical for the actual production outcome. However, the more efficient the production system is the less easy it is to determine bottlenecks a priori and they may shift dynamically depending on the allocation of the jobs.

Summarizing, both in process control and in online scheduling, the task of the operational layer is to use the available degrees of freedom to reach the targets set by the upper layer in the presence of uncertainties and model inaccuracies. In both cases, feedback is applied to cope with the uncertainties, including the effects that are not modeled on the upper layer. In both cases, additional operational degrees of freedom are required to counteract the uncertainties and to cope with the aspects that are not included on the higher level: the dynamics of the plant in the case of process control and the unmodelled tasks and resources in the case of scheduling. Only if such additional degrees of freedom are available, a compensation of the effect of the uncertainties is possible. Control systems and online scheduling algorithms have to react to information on the actual situation that arrives iteratively. In model-based control and in model-based scheduling, the decisions are optimized over a forecast horizon in order to take longer term effects due to the inertia of the controlled system into account, but only a subset of "next" decisions or optimized variables have to be fixed and implemented based upon the available information. Moving horizon schemes were first employed in process control but are increasingly used in planning and scheduling, see e.g. (Sand et al., 2000), (Engell et al., 2001), (Kelly and Zygnier, 2008), (Pujgjaner and Lainez, 2008). On the other hand, the use of two-stage formulations to include uncertainties as well as the future reaction of the controller (resp. scheduler) to the realization of the uncertainty into account provides also a suitable non-conservative formulation for robust model-predictive control (Dadhe and Engell, 2008).

3. Uncertainty and feedback in a dynamic scheduling context

Uncertainties are a major element in any real-time decision problem where the information about the presence and the future unfolds iteratively. Besides the uncertainty on the future demands, in medium-term planning and in online scheduling, the information on the capacity and the constraints of the plant is to some extent uncertain, due to model abstraction on the higher level and to incomplete knowledge and stochastic events on the lower level.

In control problems, it is usually assumed that the reference trajectories are fixed and known over the horizon over which the online optimization problems are solved, but that the plant dynamics are not exactly represented by the model used. The usual approach in model-predictive control to handle these uncertainties is to modify the references and the constraints by the observed difference between the predicted and the observed outcome of the application of the control moves in order to obtain offset-free control. Due to plant-model mismatch, i.e. the gradients of the plant dynamics and of the constraints with respect to the free inputs are not known exactly, the optimization will usually provide a suboptimal, but feasible operating point (Forbes and Marlin, x). Techniques to modify the gradients based upon measured information have been proposed by (Tatjewski, 2002) and (Gao and Engell, 2005).A second element that introduces feedback in order to counteract the uncertainties is the estimation of the plant state from the available measurements and the update of the optimization problem using the new estimates of the state of the model.

3.1. Representation of uncertainty in scheduling

For the handling of the uncertainties in online scheduling, four basic approaches have been proposed:
- full event-driven or periodic rescheduling
- reactive scheduling
- robust scheduling
- multi-stage or two-stage stochastic scheduling.

The most straightforward approach to handle uncertainties is dynamic rescheduling. Taking into account the real progress of the operations and new information about the production targets, all decisions that can be modified are recomputed iteratively or if major events are encountered, using a fixed nominal model or including model adaptation. This is similar to the approach in model-predictive control when the optimization is iterated over a shifted horizon, the new (observed or estimated) plant state is taken into account and the targets are shifted by the difference of the planned and the real outcomes. The critical aspect of pure rescheduling is the issue of feasibility. There is no guarantee that the process will not run into deadlocks (or very unfavorable situations) because of the discrepancies between the assumed and the real evolution. A key role of humans in the production process is to foresee such situations and to correct the planned actions accordingly.

If the computation time required for full rescheduling is too long for a frequent online application, it can be approximated by reactive scheduling. Reactive scheduling modifies nominal schedules as a reaction to the occurrence of unexpected events (see e.g., Cott and Macchietto, 1989; Honkomp et. Al., 1997; Kanakamedala et al. 1994; Mendez and Cerda, 2004; Vin, and Ierapetritou, 2000, Janak et al., 2006a,b). The underlying models themselves usually do not incorporate information about the uncertainties. Reactive scheduling is present in any real production, often performed by the plant managers directly or induced by the process itself –e.g. when the preceding step is finished late, the start-time of the next one is adapted. The analogy to reactive scheduling is the use of nested low-level controllers to implement set-points in process control. Hereby, the effects of disturbances and plant uncertainty are reduced, variability is shifted from the controlled variables to the manipulated inputs. By using carefully chosen regulatory control loops, some of the potential of rigorous optimization can be recovered (Skogestad, 2000, Engell et al., 2005).

Robust scheduling and multi-stage scheduling are variants of stochastic scheduling. Here models are employed that take uncertainty explicitly into account. Stochastic models with recourse consider the corrective measures that can be taken after the realization of some uncertain parameters while in robust scheduling, this option is not included.

In robust scheduling, the parameters of the scheduling problem are considered as uncertain, usually varying in certain intervals, with or without knowledge of the distribution functions of the variation. Scheduling is then performed such that the best value of the cost function is achieved and all constraints are met in the worst possible situation. Jia and Ierapetritou (2004) investigated the use of MILP sensitivity analysis in robust short-term scheduling under demand uncertainty. Balasubramanian and Grossmann (2003) proposed the use of concepts from fuzzy set theory to describe imprecisions and uncertainty for the minimization of the makespan of flowshop scheduling with uncertain task duration. Lin et al. (2004) and Janak et al. (2007) proposed an efficient MILP optimization methodology for generating schedules that are robust in the sense that the solutions are feasible (with a prescribed infeasibility tolerance) in the presence of uncertainties in the inequality constraints for interval bounds on the parameters and, with a prescribed probability, for probabilistic uncertainties in the inequality constraints, and optimal with respect to the nominal cost function This methodology was applied to short-term scheduling with uncertainties in processing times, demands, and cost coefficients.

The problem of robust scheduling – and similarly of min-max formulations in model predictive control – is that the computed solution is feasible for the worst case but may fail to realize the potential of the real plant in all other situations. It corresponds to a pessimistic open-loop approach – all decisions are fixed and computed for the worst possible situation. Note that the concept of iterative optimization on a rolling horizon does not alleviate this problem – the decisions that are implemented are based on pessimistic (worst-case) assumptions and cannot be corrected while in the simple rescheduling approach they are based on average or optimistic assumptions. On the other hand, feasibility is always assured, given that the bounds include all possible situations that are met in the future. In the dynamic context, however, the future decisions can be adapted to the realization of the uncertainties, and this provides significant room for better choices of the decisions that have to be fixed here and now.

A multistage stochastic decision problem is characterized by a non-anticipative information structure. The problem description includes stochastic aspects modeled either by continuous probability distributions or by a finite number of scenarios (the latter case is usually considered because it is computationally better tractable). If the uncertainty is modeled by a scenario tree with N stages (see Fig. 2), then the decision process progresses along this scenario tree as well. In stage i, the decision is based on the certain information on the realization of a path in the tree up to this node whereas the future evolution is only known probabilistically, and is represented by the sub-tree that starts at the corresponding node of the tree (Till et al., 2008). To solve multi-stage problems, at each stage the reaction of the algorithm to the information obtained at later stages must be taken into account, leading to a complex nested structure.

Therefore multi-stage problems usually are approximated by two-stage problems, as shown in Fig. 2 (right). The decision variables are divided into the first and second-stage vectors $\boldsymbol{x}$ and $\boldsymbol{y}_\omega$, which belong to the sets X and Y, possibly with integrality requirements. The vector $\boldsymbol{x}$ represents “here and now”-decisions which are applied regardless of the future evolution and thus have to be identical for all scenarios. In contrast, the vectors $\boldsymbol{y}_\omega$ denote scenario-dependent recourses under the assumption that the respective scenario realizes.

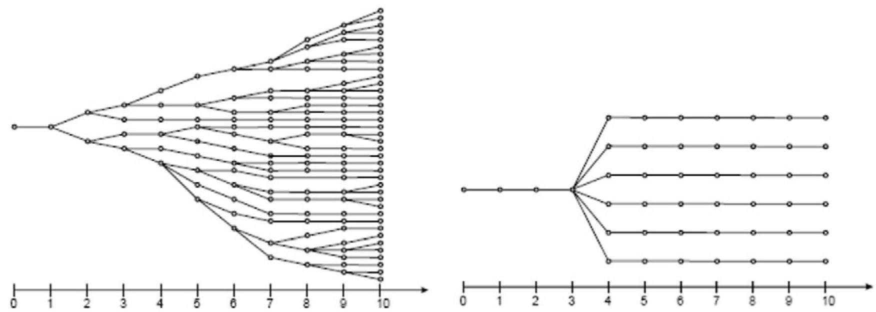

Figure 2: Multi-stage (left) and two-stage (right) stochastic optimization problems represented by scenario trees

The objective is to minimize the first-stage costs plus the expected second-stage costs calculated using the weighting-vectors $\boldsymbol{c}$ and $\boldsymbol{q}_\omega$. The uncertain parameters are represented by a finite number of realizations (scenarios) ω with corresponding probabilities π_ω. The objective is to minimize the first-stage costs plus the expected second-stage costs calculated using the weighting-vectors $\boldsymbol{c}$ and $\boldsymbol{q}_\omega$. In the mixed-linear case, this leads to an optimization problem of the form

$$\begin{aligned} &\min_{\boldsymbol{x},\boldsymbol{y}_\omega} \quad \boldsymbol{c}^T\boldsymbol{x} + \sum_{\omega=1}^{\Omega} \pi_\omega \boldsymbol{q}_\omega^T \boldsymbol{y}_\omega \quad s.t. \quad \boldsymbol{Ax} \le \boldsymbol{b}, \boldsymbol{T}_\omega \boldsymbol{x} + \boldsymbol{W}_\omega \boldsymbol{y}_\omega \le \boldsymbol{h}_\omega, \\ &\qquad \boldsymbol{x} \in X, \boldsymbol{y}_\omega \in Y, \omega = \{1,\ldots,\Omega\}, X \in IR^{n_1'} \times IN^{n_1''}, Y \in IR^{n_2'} \times IN^{n_2''}. \end{aligned} \tag{1}$$

The parameters of each realization enter into the matrices $\boldsymbol{T}_\omega$, $\boldsymbol{W}_\omega$ and the vectors $\boldsymbol{h}_\omega$, $\boldsymbol{q}_\omega$.

In Fig. 2b, the first stage comprises the variables associated with stages 0-3 whereas the second stage comprises the variables associated with stages 4-10.

The mathematical framework of two-stage stochastic programs provides a modular modeling concept for uncertainty conscious scheduling problems: In principle, any deterministic scheduling model can be extended to a stochastic model provided that the uncertainties affect only the parameters of the formulation. The extension requires the definitions of
(1) scenarios for the uncertain parameters along with their probabilities,
(2) first stage variables
(3) an appropriate objective (e.g. expected value, excess probability).

The approximation of a multi-stage problem by a two-stage problem corresponds to an optimistic assumption about the future: It is implicitly assumed that after the first stage decisions have been implemented, all uncertainties are revealed and the subsequent decisions are the optimal ones for the scenario that materializes. In fact, this optimal decision cannot be computed because of the sequential nature of the problem. However, the formulation guarantees that for each optimal first stage decision there is a feasible second-stage vector $\boldsymbol{y}_\omega$. for all scenarios $\omega = 1 \ldots \Omega$. The above formulation can be extended to include measures of risk either by adding constraints for risk-aversion or by setting up a two-criteria problem and computing Pareto-optimal solutions.

When integer requirements are present in the recourse, the resulting stochastic integer program is often of large scale and cannot be solved easily without incorporating decomposition based methods or problem-specific heuristics. Therefore solving stochastic integer programs is sometimes avoided by assuming that all integer decisions are first-stage variables.

3.2. Review of scenario-based stochastic formulations of scheduling problems

Sand et al. (2000) and (Engell et al. 2001) extended a multi-period model for the medium-term planning of an industrial multi-product batch plant for the production of expandable polystyrene (EPS) to a two-stage model. Interesting features of this example are that the production is strongly coupled, i.e. each batch yields several products and that the final continuous processing stage involves mixing of material from different batches. The model considers uncertainties in demands and in capacities. The objective is to maximize the expected profit including a measure of risk aversion. The first-stage decisions consist only of integer variables and the second-stage decisions consist of integer and continuous variables. Sand et al. (2004) applied the dual decomposition based algorithm of Carøe and Schultz (1999) to solve the resulting stochastic integer programs with approximately 600 continuous variables, 100 integers, and 500 constraints in each of 1024 scenarios. Solutions with a gap of less than 9% were found within 4 h of CPU-time. Their programs are based on discrete-time multi-period models where the stages correspond to the time periods.

Vin and Ierapetritou (2001) extended the event based continuous time model of Ierapetritou and Floudas (1998) to a two-stage stochastic program and minimized the expected makespan. They observed that using the stochastic model increased the robustness of the solution. Furthermore, they compared the use of deterministic and stochastic models in the reactive scheduling framework of Vin and Ierapetritou (2000) and found that the reactive scheduling performance was not necessarily improved by using the robust solutions obtained from stochastic models.

Bonfill et al. (2004) extended a continuous time batch-slot concept model to a two-stage stochastic program and optimized the weighted sum of the expected profit and a risk term. The first stage comprises all binary and integer decisions of the detailed schedule while the second-stage consists of the remaining continuous decisions as e.g. sales. They observed an improved performance of the stochastic scheduler for a flowshop example. Balasubramanian and Grossmann (2004) modeled integer recourse decisions explicitly in a multi-stage stochastic integer program. Each of the M stages corresponds to a time period and the decisions are assigned corresponding to the stages. An additional stage is added to consider the continuous variables for the amounts sold and lost, the costs, and the revenues after the end of the scheduling horizon. A shrinking-horizon approximation scheme for multi-stage stochastic integer programs was proposed based on the solution of a series of two-stage stochastic programs with continuous second-stage. At stage i, all decisions of stage i to M are taken as the first-stage variables, while the second-stage decisions consist of the continuous recourse decisions of the stage $M+1$. The variables that correspond to stage i are fixed, then the procedure is repeated for the remaining stages i+1 to M.

Gröwe-Kuska et al.(2005) reported the application of multi-stage programs with mixed-integer recourse for the short-term unit commitment of hydrothermal power systems under uncertainty. The dimension of the multi-stage problems ranges up to 200,000 binary and 250,000 continuous variables. The model was solved by a problem specific scheme based on Lagrangian relaxation that exploits the existence of loose couplings between the units. Two-stage stochastic programming has also been applied in production and supply chain planning (e.g. Ierapetritou et al. 1995, Clay and Grossmann, 1997; Gupta and Maranas, 2003). Goel and Grossmann (2004) applied stochastic programming to the planning of off-shore gas field developments where the uncertainty is partly endogenous, and Tarhan and Grossmann (2008) considered the optimization of investments into production plants under uncertainty modeled as a multistage stochastic program where the uncertainty is gradually removed.

Alonso-Ayuso et al. (2005) proposed two and multi-stage programs with mixed-binary recourse for production planning and scheduling problems under uncertainty. Guillen et al. (2006) used two-stage stochastic scheduling models on a shrinking horizon to evaluate the design of supply chains under demand uncertainty. Wu and Ierapetritou (2007) integrated a two-stage stochastic planning model on a moving horizon into a hierarchical planning and scheduling approach. Puigjaner and Lainez (2008) employed a two-stage stochastic formulation in supply chain optimization and also related this explicitly to the rolling horizon approach used in model predictive control. Their model integrates the financial situation of the supply chain.

In Cui and Engell (2009) the two-stage formulation is applied to the moving horizon medium-term planning problem of the EPS example mentioned above. In order to avoid the blow-up of the problem size with the length of the second stage, the more distant future is represented by only one deterministic scenario with the expected values of the parameters whereas the immediate future is represented by a tree of different scenarios of production capacity and demand evolutions.

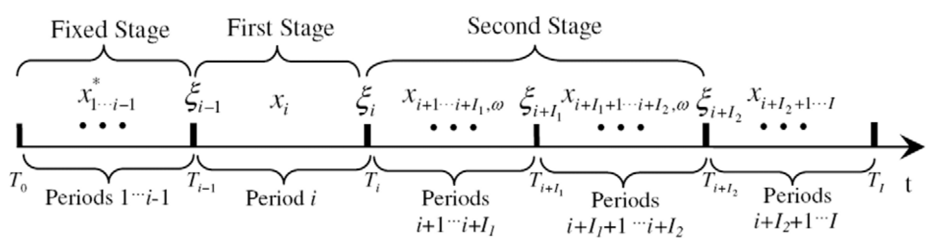

Figure 3: Moving horizon two-stage stochastic program in period i

The plant is scheduled under uncertainties $\xi = \{\xi_1, \xi_2, \ldots, \xi_{I-1}\}$ which are independent discrete random variables with discrete distributions $\psi(\xi_i)$ in a time horizon with I periods as shown in Fig. 3. $x_1, x_2, \ldots, x_I$ are the mixed-integer decision variables in each period. The near future within the next I_1 periods and represented by $f_2(\bullet)$ is modeled by a tree of scenarios in the combined sample space $\omega \in \Omega_i$ of the future demands whereas the more remote future within the following I_2-I_1 periods is represented by the expected values (EVs) of the stochastic variables (cost contribution $f_3(\bullet)$). As the inclusion of the more distant future predominantly has the purpose to rule out unrealistic solutions that maximize the benefit over a short horizon at the expense of the long-term performance, and realistic scenarios are difficult to generate for the distant future, this is a reasonable simplification. Both time horizons are rolling in the time coordinate i. In step i , as shown in Fig. 3, a 2SSIP model with the past optimal decisions $x^*_{1\cdots i-1}$, actual decision variables x_i, realized uncertainties ξ_{i-1}, additional uncertainties ξ_{i+I_1-1} and EVs for the distant future $\xi_{i+I_1\cdots i+I_2-1}$ is solved and the first stage optimal solutions x^*_i are implemented. Period i is the first stage and period i+1 to period $i+\boldsymbol{I_2}$ constitute the second stage in the 2SSIP setting.

The strength of the formulation of scheduling problems as two-stage stochastic decision problems is that it is perfectly adapted to a rolling-horizon approach where the decisions after the first period are re-computed based upon the new information obtained so that in the decision process, a reaction to the future developments is possible. In other words, the feedback that is present in the real decision structure by the update of the decision vector after each period is represented adequately, in a slightly optimistic fashion, because of the assumption that from some point on a clairvoyant scheduler is employed. In contrast, in a robust formulation, the scheduler bases its decisions on the assumption that the worst possible happens in the future and that no corrective action can be taken. The disadvantage of the approach is the high computational cost that is prohibitive in online applications. In particular, the time until a first feasible solution is obtained by an exact algorithm can be very large for large numbers of scenarios

In our recent work, hybrid algorithms were developed that employ stage decomposition. (Till et al., 2007, Sand et al., 2008, Tometzki and Engell, 2009).

3.3. Stage decomposition algorithm for the solution of two-stage programs

In a stage decomposition appraoch, the variables and the constraints of problem (1) are separated according to the stages into the master-problem (2) and Ω subproblems (3). Due to the integrality requirements in Y, the implicit function $Q_\omega(\boldsymbol{x})$ is in general non-convex.

$$\min_{\boldsymbol{x}} \quad f_1(\boldsymbol{x}) = \boldsymbol{c}^T\boldsymbol{x} + \sum_{\omega=1}^{\Omega} \pi_\omega Q_\omega(\boldsymbol{x}) \quad s.t. \quad \boldsymbol{Ax} \le \boldsymbol{b}, \boldsymbol{x} \in X, X \in IR^{n_1'} \times IN^{n_1''}, \tag{2}$$

$$Q_\omega(\boldsymbol{x}) = \min_{\boldsymbol{y}_\omega} \quad f_{2,\omega}(\boldsymbol{y}_\omega) = \boldsymbol{q}_\omega^T\boldsymbol{y}_\omega \quad s.t. \quad \boldsymbol{W}_\omega\boldsymbol{y}_\omega \le \boldsymbol{h}_\omega - \boldsymbol{T}_\omega\boldsymbol{x}, \boldsymbol{y}_\omega \in Y, Y \in IR^{n_2'} \times IN^{n_2''}. \tag{3}$$

The main idea of the hybrid algorithm is to use an evolution strategy (ES) (Beyer and Schwefel, 2002) to address the master-problem (2). An ES works on a population of individuals (pool of solution candidates). The ES interprets a certain instantiation of $\boldsymbol{x}$ as an individual and the corresponding $f_1(\boldsymbol{x})$ in (2) as the fitness of the individual. The fitness is evaluated by solving the independent subproblems (3) using a standard MILP solver. Infeasibilities are handled by penalty terms. Computational experiences show that the hybrid algorithm obtaines good feasible solutions for large numbers of scenarios considerably faster than the decomposition algorithm by Caroe and Schultz (1999) and straightforward application of a MILP solver (CPLEX). When CPLEX has found an admissible solution, however, it converges faster to optimal values of the cost function than the hybrid algorithm.

4. Hierarchical decision structures

The motivation of hierarchical decision structures is to reduce the complexity of the problem at hand to make it tractable by humans or algorithms and also to maintain transparency of the solutions. While in most problems all decision variables interact in their influence on the overall performance, this interaction can and should be neglected to some extent because of the presence of uncertainty and of feedback. The presence of uncertainty leads to the situation that detailed decisions beyond a certain horizon are obsolete soon after they have been computed, while feedback remedies the propagation of uncertainty. The decomposition in layers corresponds to time-scale decomposition:

long-term decisions are optimized over a long horizon using simplified (averaged) models, short-term decisions are based upon very detailed models taking into account all available information. The temporal resolution as well as the decisions variables included on each level should reflect the precision of the information that is available as well as the duration of the impact of the decisions. Typically, a hierarchical decomposition for the solution of industrial-size problems will include (at least) the layers of long-term planning, mid-term planning and (short term) scheduling.

In long-term planning, decisions on the production capacities, on the distribution of production tasks among sites and on the production volumes are taken using aggregate, and usually predicted, hence relatively uncertain data on market demand, cost structures, raw material prices etc. Usually coarse models of the required and of the available resources are used. The accuracy of long-term planning can be improved by performing detailed planning over the full planning horizon using semi-heuristic algorithms (Plapp et al., 2008), however this requires a large effort and the result can then only be used in "what ... if" investigations. The implications of the decisions on this level concern mostly procurement: ordering of material, addition or reduction of production capacities, long-term staff planning. The precision and the temporal resolution of the data usually is low, a resolution finer than a week will usually by not justified, often planning intervals will be months. The update interval may be (considerably) longer than the resolution, so this is an off-line activity with only weak feedback from the daily operations.

The medium-term planning and the scheduling layer correspond to the scope of APS (advanced planning and scheduling) systems (Goebelt et al., 2008, Jaenicke and Seeger, 2008). On these layers, all orders that are considered (customer orders or production to stock) have delivery dates and the assignment of production capacities to the tasks is performed based upon a detailed model of the required resources. The temporal resolution Δ_M on the medium-term planning layer usually will be days or shifts. Short-term scheduling is the exact temporal assignment of all required resources to production tasks. The temporal resolution will be hours or even minutes.

4.1. Review of hierarchical solution approaches

Hierarchical algorithms were first proposed for the solution of large scheduling problems that cannot be solved directly with the goal to obtain a near-optimum schedule over the full horizon. (Bassett et al., 1994/1996) proposed to employ different discretizations of time. On the upper layer, aggregated problems are formulated to allocate production tasks to time slots for which a detailed schedule is computed afterwards one-by-one. The automatic aggregation of detailed models was refined in (Wilkinson et al., 1995) with special attention to the couplings between the intervals. Dimitrades st al (1997) applied a non-uniform discretization of the time axis where the initial intervals are modeled on a finer grid than the subsequent ones. In each step, the problem is solved over the full horizon, the initial decisions are frozen and the solution process proceeds in a shrinking horizon fashion until a schedule over the full horizon has been computed. Carryovers from one period to the next are handled by extending the detailed model into the next period until all tasks considered have been finished.

The idea that only the immediate future needs to be computed in a detailed fashion goes back to Zentner et al. (1996). Papageorgiou and Pantelides (2000) proposed a hierarchical approach to campaign planning. Harjunkoski and Grossmann (2001) used a bi-level decomposition strategy for the scheduling of a steel plant. The products are grouped into blocks, and the sequences within these blocks are determined separately, following by an optimization of the sequence of the blocks. Van den Heever and Grossmann (2003) used a rolling horizon approach and a Lagrangean decomposition strategy for production planning and scheduling of a hydrogen plant. Mendes and Cerda (2003) discussed problem formulations and solution algorithms for dynamic scheduling problems where new data arrives iteratively.

A hierarchical decision structure that includes explicit models of uncertainty and a hierarchical decomposition was proposed in (Sand et al., 2000) and (Engell et al., 2001) based on the analysis of the scheduling problem for the EPS plant. This plant presents a typical example were on the one hand side decisions have to be taken on a very fine time scale using detailed models but on the other hand have long-term effects that go beyond the horizon that can be dealt with in detailed scheduling. Moreover, the problem data is uncertain, concerning demands, yields, and capacities. The time horizon needed for the decision on the choice of the recipes and the number of batches exceeds the period over which a detailed schedule can be computed. It was proposed to combine a two-stage stochastic medium-term planning model on a coarse time grid with fixed period lengths with a deterministic short-term scheduling model formulated in continuous time that is solved in an event-driven fashion (see Fig. 4). The upper layer was assumed to provide guidelines (numbers of batches for each recipe that are started in a two-day period and states of the finishing lines (on or off)). Feedback is provided by reporting the batches that have been started or are scheduled to start before the end of the computation period back to the medium-term planning layer. On the upper layer, linearized models are used while for scheduling, nonlinear models of the mixers and a partly heuristic MINLP algorithm are employed (Schulz et al., 1998, Engell et al., 2000).

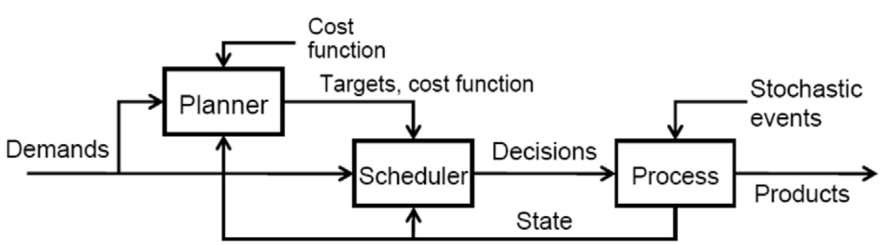

Figure 4: Two-level online decision structure proposed for the EPS process in (Sand et al., 2000)

In this scheme, the medium-term planning problem with uncertainties in demands and yields was formulated as a two-stage stochastic planning problem using aggregate models (Sand et al., 2000, Sand and Engell, 2004) and the problem was solved using the decomposition strategy by Carøe and Schultz (1999).

Stefansson et al (2006) proposed a multi-layer architecture for campaign planning and detailed scheduling in the pharmaceutical industry. They propose a three-layer architecture in which on the upper layer, campaign planning is performed over a horizon of 12 months, on the middle layer, a detailed order allocation is performed over a horizon of three months with the goal to confirm delivery dates, and on the lowest level detailed scheduling is performed based upon confirmed customer orders using a continuous-time model. They assume a one-directional interaction between the layers, i.e. there is no feedback from the lower layers back to the upper ones which implies that all proposed decisions on the upper layers can be implemented on the lower layers.

Janak et al (2006a,b) applied a temporal decomposition approach to an industrial batch scheduling problem. The decomposition idea is to first determine the time period and the products that are considered in short-term scheduling, and then to solve the scheduling model. The horizon length is maximized under the constraint that the complexity of the resulting problem is bounded, thus enabling reasonable computing times. In the same spirit, Shaik et al. (2008) solve a scheduling problem for a large industrial continuous plant by two-level decomposition. On the upper (medium-term) level, demands are allocated to time slots (sub-horizons), and on the lower level the sequencing of the next operations in the sub-horizon is performed in a rolling horizon fashion. Verderame and Floudas (2008) define a relaxed planning model for large scale problems on the upper layer of a two-level formulation similar to Janak et al. (2006a) where the resources considered are only the bottleneck resources and use this model to generate daily production profiles (targets) for the scheduling layer.

Wu and Ierapetritou (2007) proposed a scheme which implements a similar concept as shown in Fig. 4. The planning horizon in their work consists of three stages of different lengths. In the short first stage, the data is assumed to be deterministic and detailed scheduling problems are solved. In the two other stages of increasing lengths that represent the immediate and the remote future, the demands and the production capacity are aggregated. The planning problems are solved using simplified models in which the unmodeled parts of the scheduling problem are represented by a sequence factor that describes how much less products can be obtained due to the additional constraints in the scheduling model compared to the planning model. The planning algorithm determines the amount of product that is required after the first stage, taking into account different scenarios for the future demands with associated probabilities. So the planning model is a two-stage problem with the requested production in the first stage being the here-and-now variables. Depending on whether the scheduling problem is feasible or not, the sequence factor is updated, introducing feedback from the scheduling layer to the planning layer. The results for a case study show the superiority of the approach over pure rolling horizon scheduling approaches with look-ahead of one or two periods, compared to a horizon of 9 periods considered in the planning model, for a situation where the backorder cost is high.

The interaction of long-term planning by ERP systems and APS systems is discussed in (Goebelt et al., 2008 and in (Jaenicke and Seeger, 2008) from an industrial point of view. Goebelt et al. assume different models and algorithms on the two layers that are reconciled by overlapping horizons whereas Jaenicke and Seeger propose a top-down refinement approach.

(Sousa et al., 2008) discuss a hierarchical multilevel approach to supply chain optimization which is nicely illustrated by an example of active ingredients manufacturing. The proposed approach is to solve a planning problem with aggregated capacities over a cyclic horizon of one year, discretized into 12 planning periods. This provides inputs to the lower scheduling layer, inventory levels of products at locations at the start and at the end of a month, product flows to formulation sites (integrated) per period, product flows to customers (integrated) per period, site opening, allocations of products to manufacturing sites, product supplies to specific customers from specific sites.

Based upon this information, schedules are computed at the lower level for each month using detailed recipes and changeover times. On the lower level, the demands are distributed over time whereas on the upper level, they are aggregated to the ends of the months. The inventory levels are considered as constraints whereas the demand satisfaction is reflected in the cost (penalty terms for late delivery). It is observed that without feedback from the second stage to the first stage, there is a significant mismatch between demands and delivery. In order to solve this problem, capacity correction factors that depend on the number of campaigns planned are introduced on the planning layer and additional constraints are formulated. These measures achieve the goal of reducing the gap between the layers without sacrificing overall production, after some manual adaptations. They increase the inventories computed on the planning level. A direct manipulation of inventory levels did not provide consistently good solutions.

4.2. Considerations on the interaction between medium-term planning and scheduling

The relation between hierarchical levels is one of refinement and abstraction. The lower levels include more details on the timing of the production steps and on the resources needed, hence more constraints and more degrees of freedom are present. In the ideal case, these degrees of freedom are sufficient to satisfy the additional constraints that have to be taken into account such that the results of the optimization on the upper layer can be implemented on the lower layer without any violation of constraints. In reality, this will rarely be the case. If, e.g., the production capacity is assumed to be known on a weekly basis on the upper layer and is fully used, it is unrealistic that exactly this amount of products can be produced if more detailed constraints are considered. The simplest way to avoid this situation is to systematically underestimate the capacities when reporting to the higher layer, a strategy that is not uncommon in practice. In general, it cannot be assumed that the implementation of the upper layer decisions is always feasible, hence feedback mechanisms are necessary and it is preferable to provide targets and a cost function that measures the satisfaction of the targets in contrast to restricting the lower layer by hard constraints. Within the horizon under control of the lower layer, as few constraints as possible should be formulated to provide maximum flexibility to the lower layer to cope with unmodelled effects and uncertain parameters and events.

The inclusion of explicit descriptions of uncertainty into the models requires a careful design of the decomposition and of the coupling of the layers. Assume e.g. that a solution was obtained on the medium-term planning layer that assigns certain production tasks within the first k_M intervals of the planning horizon, taken as first-stage decisions, to the scheduling layer. Then this assignment is based on the assumption of a certain estimated average capacity during the first interval, determined from available information, e.g. on planned maintenance. If the available capacity in the first scheduling interval is higher than this average value, how should the scheduler react? Intuitively, it might be wise to use the higher-than-average capacity in order to account for possible breakdowns in the future. But this would be penalized if the first stage variables from the medium-term planning are used as targets for the scheduler in the respective intervals. In addition, updated information may be available during the scheduling process on the amounts to be produced and on the due dates. Generally speaking, it is not a good idea to fix variables inside the horizon of the lower layer unless these variables really have to be set and are beyond the control of the scheduler. Maximum flexibility should be given to the lower level algorithms to use the information and the degrees of freedom that are available in order to maximize the overall goals in the presence of uncertain parameters, stochastic events and restrictions that are unknown to the upper layer. Conversely, only the information needed to perform the next update on the upper layer should be fed back from the lower layer, e.g. updates on yields, expected capacities, planned resource utilization etc.

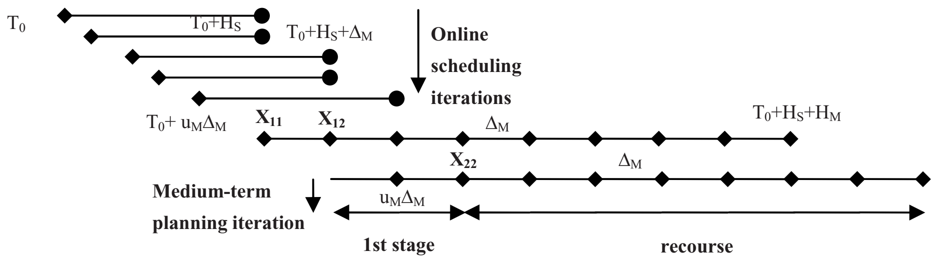

Figure 5: Telescopic two-layer moving horizon scheme

In Fig. 5, a telescopic two-layer moving horizon scheme is sketched where the medium-term planning layer employs a two-stage formulation to include uncertain future evolutions. The horizon on the scheduling level is $\Delta_S\ H_S = k_{M.}\ \Delta_{M,}\ k_M \geq 1$ where Δ_M is the length of the time slots on the medium-term planning layer. On the planning layer, a two-stage formulation is used where the length of the first stage is $u_M\ \Delta_M$ The key idea of the scheme depicted in Fig. 5 is to avoid double planning and to link the layers only via target values or predicted values of the "states" of the system. These are the amounts of materials considered in medium-term planning and scheduling at time t, **X(**t) and the planned resource utilizations after time t, **U**(t). From the point of view of the lower layer, a target is required that reflects the look-ahead-capability of the upper layer but does not restrict the decisions of the lower layer more than necessary. From the point of view of the upper layer, the predictions of the materials produced or consumed at the beginning of the medium-term planning horizon and of the resource utilization after this time by the scheduling horizon provide the necessary information about the past and the inertia of the system. Iterative medium-term re-planning is not performed over the full scheduling horizon but only over that part of it that lies in the next planning horizon.

At time T_0 the scheduling algorithm performs an optimization of the resource allocation over the scheduling horizon of length H_S taking the actual data on the available resources and the demands and possibly also scenarios of the future resources and demands into account plus targets for the state $\mathbf{X}(T_0+H_S)$ at the end of the scheduling horizon which is provided by the medium-term planning algorithm ($\mathbf{X}_{11}$ in Fig. 5). The end of the scheduling horizon should coincide with the endpoint of a medium-term planning interval, $T_0+H_S = p_0\Delta_M$, p_0 integer. The scheduling algorithm is iterated either after fixed periods of time or when new data is available with the same end point T_0+H_S as before and the same target state, until the length of the scheduling horizon is less than or equal to H_S-Δ_M. At this point, the endpoint is shifted to $(p_0+1)\ \Delta_M$ and the target is the state at $T_0+H_S+\Delta_M$ as computed by the medium-term planning algorithm in the previous iteration ($\mathbf{X}_{12}$ in Fig. 5). This is iterated until the endpoint has been shifted by the update interval on the medium-term layer relative to the first endpoint. At this point, the medium-term planning algorithm is run to compute an update of the targets. The planning horizon of the medium-term planning algorithm is shifted by $u_M\ \Delta_M$. The horizon that is now considered by the medium-term planning algorithm is $((p_0+u_{M)}\ \Delta_M,\ p_0+u_M\Delta_M+H_M]$ and the decisions between $(p_0+u_M)\ \Delta_M$ and $(p_0+2u_M)\ \Delta_M$ are taken as first-stage variables while the remaining ones are recourse variables. After the computation, the decision variables of the first stage are not given to the scheduler but only the resulting values of the planned materials at times $(p_0+1+u_M)\ \Delta_M$ … (p_0+2u_M)□$_M$ ($\mathbf{X}_{21}$ and $\mathbf{X}_{22}$ in Fig. 5). At time $T_0 + u_M\ \Delta_M$ the best possible prediction of the state of the system at the initial time $(p_0+u_M)\ \Delta_M$ is the (expected) value that is computed by the scheduling algorithm for

$\mathbf{X}((p_0+u_M)\,\Delta_M)$ and $\mathbf{U}((p_0+u_M)\,\Delta_M)$. Therefore this state is used as the initial condition of the medium-term planning algorithm. This scheme is iterated every $u_M\,\Delta_M$ units of time.

5. Summary and Future Research

In this paper, we have discussed the design of medium-term planning and scheduling algorithms from a dynamic systems point of view and drawn parallel between process control and planning and scheduling. In summary, medium-term planning and scheduling are reactive activities where decisions have to be made under uncertainty and in real time. Moving horizon formulations with recourse (two-stage or multi-stage stochastic programs) reflect the dynamic nature, the presence of uncertainty and the potential of reacting to the realization of the uncertainties adequately. In addition, measures of risk can be included in the optimization. A drawback is the computational effort which limits the horizon of the second stage and the number of scenarios. The solution of real-world problems requires a hierarchical decomposition, as advocated by several authors recently. In such a hierarchical decomposition, the coupling of the layers is critical. The task of the lower layer should be formulated such that maximum freedom is left for the adaptation to the constraints and the reaction to uncertainties while the overall goal is properly reflected in the cost function. In a hierarchical decomposition approach, the issue of distributed solutions, in particular on the lower layer arises naturally, as this enables the parallel solution of smaller sub-problems. An initial discussion of the coordination between different layers and different decision makers on these layers can be found in (Kelly and Zygnier, 2008). Distributed decision making for more than very few units however has up to now only rarely been investigated in detail in the literature on batch scheduling and is a promising area for future research.

At the interface of scheduling and process control for batch processes, there is a potential for a tighter coupling of the optimization of the batch runs which is a prominent topic in process control (see e.g. Srinivasan et al., 2002a,b) and medium-term planning and scheduling. Depending on the schedule, the optimization goals on the control layer can be modified: if the unit is in high demand, a reduction of the batch time by time-optimal feeding and cooling policies is indicated whereas in a situation where the schedule is not tight, energy could be saved by a different operation policy or the yield could be increased. In the opposite direction, the scheduling layer could be supported by a precise prediction of the expected batch times and yields before the termination of a batch.

Finally, the practical success of any sophisticated solution, in control as in scheduling, depends on a good interface to the operators. Optimal policies must be visualized adequately so that they can be understood by the operators. E.g. for well qualified and responsible operators, not only one single result of an optimization run should be provided but a set of several "best" solutions such that they can trade the loss of optimality with other criteria that are not represented in the optimization and gain confidence in the proposed solutions.

6. References

Alonso-Ayuso, A., L. F. Escudero, and M.T. Ortuno (2005). Modeling production planning and scheduling under uncertainty. In S. W. Wallace, and W. T. Ziemba (Eds.), Applications of stochastic programming. MPS-SIAM Series in Optimization, 217–252.

Balasubramanian, J., and I.E. Grossmann (2003). Scheduling optimization under uncertainty - An alternative approach. Comp. and Chem. Engg. 27 (4), 469–490.

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

Bassett, M.H., F.J. Doyle III, G.K. Kudva, G.V. Reklaitis, S. Subrahmanyam, M.G. Zentner, and D.L. Miller (1994/1996). Perspectives on model-based integration of process operations. Proc. 5th Int. Symposium on Process Systems Engineering, Kyongju, Korea and Comp. Chem. Engg. 20, 821.

Beyer, H., and H. Schwefel (2002). Evolution strategies. Natural Computing 1, 3-52.

Bonfill, A., M. Bagajewicz, A. Espuna, and L. Puigjaner (2004). Risk management in the scheduling of batch plants under uncertain market demand.. Ind. Eng. Chem. Res. 43, 741–750.

Carøe, C., and R. Schultz (1999). Dual decomposition in stochastic integer programming. Operations Research Letters 24, 37-45.

Castro, P., and I.E. Grossmann (2006). Multiple Time Grid Continuous-Time Formulation for the Short Term Scheduling of Multiproduct Batch Plants. Proc. 2006 ESCAPE/PSE, Elsevier, 2093-2098.

Castro, P., C. A. Mendez, I.E. Grossmann, I. Harjunkoski, and M. Fahl (2006). Efficient MILP-based solution strategies for large-scale industrial batch scheduling problems. Proc. 2006 ESCAPE/PSE, Elsevier, 2231-2236.

Clay, R., and I.E. Grossmann. (1997). A disaggregation algorithm for the optimization of stochastic planning models. Comp. Chem. Engg.21, 751–774.

Cott, B., and S. Macchietto (1989). Minimizing the effects of batch process variability using online schedule modification. Comp. Chem. Engg. 13, 105–113.

Cui, J. and S. Engell (2009). Scheduling of a multiproduct batch plant under multiperiod demand uncertainties by means of a rolling horizon strategy. Proc. 19th European Symposium on Computer Aided Process Engineering.

Dadhe, K. and S. Engell (2008). Robust nonlinear model predictive control: A multi-model non-conservative approach. Presented at the 2008 Workshop on the Assessment and Future Directions of Nonlinear Model-predictive Control, Pavia.

Dimitrades, A.D., N. Shah, and C.C. Pantelides (1997). RTN-based rolling horizon algorithms for medium-term scheduling of multipurpose plants. Comp. Chem. Engg. 21, 1061-1066.

Engell, S. (2007). Feedback Control for Optimal Process Operation. J. Proc. Control 17, 203-219.

Engell, S., A. Maerkert, G. Sand, R. Schultz, and C. Schulz (2001). Online Scheduling of Multiproduct Batch Plants under Uncertainty. In: M Groetschel, S.O. Krumke, J. Rambau (Eds.): Online Optimization of Large Scale Systems, Springer, 2001, 649-676.

Engell, S., S. Kowalewski, C. Schulz, and O. Stursberg (2000). Continuous-Discrete Interactions in Chemical Processing Plants. Proceedings of the IEEE 88, 1050-1068.

Engell, S., T. Scharf, and M. Völker (2005). A methodology for control structure selection based on rigorous process models. Proc. 16th IFAC World Congress, Prague, Paper Code Tu-E14-TO/6.

Ferrer-Nadal, S., C.A. Mendez, M. Graells, and L. Puigjaner (2007). A novel continuous-time MILP approach for short-term scheduling of multipurpose pipeless batch plants. Proc. ESCAPE 17, Elsevier, 595-601.

Floudas, C. A., and X. Lin (2004). Continuous-time versus discrete-time appraoches for scheduling of chemical processes: a review. Comp. and Chem. Engg. 28, 2109-2129.

Forbes, J.F. and T.E. Marlin (1995). Model accuracy for ecconomic optimizing controllers: The bias update case. Ind. Eng. Chem. Fund. 33, 1919–1929.

Gao, W. and S. Engell (2005). Iterative set-point optimisation of batch chromatography. Comp. Chem. Eng. 29, 1401–1410.

Göbelt, M., T. Kasper, and C. Sürie (2008). Integrated Short and Midterm Scheduling of Chemical Production Processes – A Case Study. In: Engell, S. (Ed.): Logistics of Chemical Production Processes, Wiley-VCH, 239-261.

Goel, V., and I.E. Grossmann (2004). A stochastic programming approach to planning of offshore gas field developments under uncertainty in reserves. Comp. Chem. Engg. 28, 1409–1429.

Gröwe-Kuska, N., W. Römisch, and T.M. Ortuno. (2005). Stochastic unit commitment in hydrothermal power production planning. In S. W. Wallace and W. T. Ziemba (Eds.), Applications of stochastic programming. MPS-SIAM Series in Optimization). Philadelphia, SIAM, 633-653.

Guillen, G., F. D. Mele, A. Espuna, and L. Puigjaner (2006). Adressing the design of chemical supply chains under demand uncertainty. Proc. 2006 ESCAPE/PSE, Elsevier, 1095-1100.

Gupta, A., and C.D. Maranas (2003). Managing demand uncertainty in supply chain planning. Comp. Chem. Engg. 27, 1219-1227.

Harjunkoski, I.,. and I.E. Grossmann(2001). A decomposition approach for the scheduling of a steel plant production. Comp. Chem. Engg. 25, 1647-1660.

Honkomp, S. J., L. Mockus, and G.V. Reklaitis (1997). Robust scheduling with processing time uncertainty. Comp. Chem. Engg. 21, S1055-S1060.

Ierapetritou, M., and C.A. Floudas (1998). Effective continuous-time formulation for short-term scheduling. 1. Multipurpose batch processes. Ind. Eng. Chem. Res. 37, 4341–4359.

Ierapetritou, M., E. Pistikopoulos, and C.A. Floudas (1995). Operational planning under uncertainty. Comp. Chem. Engg. 20, 1499-1516.

Jaenicke, W., and R. Seeger (2008). Integration of Scheduling with ERP Systems. In: Engell, S. (Ed.): Logistics of Chemical Production Processes, Wiley-VCH, 2008, 263-277.

Janak, S. L., X. Lin, and C.A. Floudas(2007). A new robust optimization approach for scheduling under uncertainty. II. Uncertainty with known probability distribution. Comp. Chem Engg. 31, 171-195.

Janak, S.L., C.A. Floudas, J. Kallrath and N. Vormbrock (2006a). Production scheduling of a large-scale industrial batch plant. I. Short-term and medium-term scheduling.. Ind. Eng. Chem. Res. 45, 8234-8252.

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

Jia, Z., and M. G. Ierapetritou (2004). Short-term Scheduling under Uncertainty Using MILP Sensitivity Analysis. Ind. Eng. Chem. Res. 43, 3782-3791.

Kallrath, J. (2002). Planning and Scheduling in the Process Industry. OR Spectrum 24, 219-250.

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

Kelly, J.D., and D. Zygnier (2008). Hierarchical decomposition heuristic for scheduling: Coordinated reasoning for decentralized and distributed decision-making problems. Comp Chem. Engg. 32 (2008), 2684-2705.

Lin, X., S.L. Janak, and C.A. Floudas(2004). A new robust optimization approach for scheduling under uncertainty: I. Bounded uncertainty. Comp. Chem. Engg. 28, 1069-1085.

Mendez, C. A., and J. Cerda (2004). A MILP Framework for batch reactive scheduling with limited discrete resources. Dynamic scheduling in multiproduct batch plants. *Comp. and Chem. Engg.* 28, 1059-1068.

Mendez, C. A., and J. Cerda (2003). Dynamic scheduling in multiproduct batch plants. Comp. Chem. Engg. 27, 1247-1259.

Panek, S., S. Engell, S. Subbiah, and O. Stursberg (2008). Scheduling of multi-product batch plants based upon timed automata models. Comp. Chem. Engg. 32, 275-291.

Papageorgiou, L.G., and C.C.Pantelides(2000). A hierarchical approach for campaign planning of multipurpose batch plants. Comp. Chem. Engg. 17, 27-32.

Plapp, C., D. Surholt, and D. Syring (2008). Planning large supply chain scenarios with "quant-based combinatorial optimization". In: Engell, S. (Ed.): Logistics of Chemical Production Processes, Wiley-VCH, 2008, 59-91.

Puigjaner, L., and J.M. Lainez (2008). Capturing dynamics in integrated supply chain management. Comp. Chem. Engg. 32, 2582-2605.

Rolandi, P.A. and J.A. Romagnoli (2005). A framework for online full optimizing control of chemical processes. Proc. ESCAPE 15, Elsevier, 1315–1320.

Sand, G., and S. Engell (2004). Modelling and solving real-time scheduling problems by stochastic integer programming. Comp. and Chem. Engg. 28, 1087-1103.

Sand, G., J. Till, T. Tometzki, M. Urselmann, S. Engell, and M. Emmerich (2008). Engineered vs. Standard Evolutionary Algorithms: A Case Study in Batch Scheduling with Recourse. Comp. Chem. Engg. 32, 2706-2722.
Sand, G., S. Engell, C. Schulz, and R. Schultz (2000). Approximation of an Ideal Online Scheduler for a Multiproduct Batch Plant. Comp. and Chem. Engg. 24, 361-367.
Schulz, C., S. Engell, and R. Rudolf (1998). Scheduling of a multi-product polymer batch plant. Preprints FOCAPO, CACHE Publications, 75-90.
Shaik, M.A., C.A. Floudas, J. Kallrath, and H.-J. Pitz (2008). Production scheduling of a large-scale industrial continuous plant: Short-term and medium-term scheduling. Comp. Chem. Engg. 33, 670-686.
Skogestad, S. (2000). Plantwide control: the search for the self-optimizing control structure, J. Process Control 10, 487–507.
Sousa, R., N. Shah and L.G. Papageorgiou: Supply chain design and multilevel planning – An industrial case. Comp. Chem. Engg 32, 2642-2663
Srinivasan, B., D. Bonvin, E. Visser and S. Palanki (2002a). Dynamic optimization of batch processes. I. Characterization of the nominal solution. Comp. Chem. Engg. 27, 1-26.
Srinivasan, B., D. Bonvin, E. Visser and S. Palanki (2002b). Dynamic optimization of batch processes. II. Role of measurements in handling uncertainty. Comp. Chem. Engg**.** 27**,** 27–44.
Stefansson, H., N. Shah, and P. Jensson (2006). Multiscale Planning and Scheduling in the Secondary Pharmaceutical Industry. AIChE Journal 52, 4133-4149.
Subrahmanyam, S., J.F. Pekny, and G.V. Reklaites (1996). Decomposition approaches to batch plant design and planning. Ind. Eng. Chem. Res. 35, 1866-1876.
Sung, C., and C.T. Maravelias, C. T. (2006). An attainable region approach for effective production planning. Proc. 2006 ESCAPE/PSE, Elsevier, 1893-1898.
Tarhan, B. and I.E. Grossmann (2008). A multistage stochastic programming approach with strategies for uncertainty reduction in the synthesis of process networks with uncertain yields. Comp. Chem. Engg 32, 766-788.
Tatjewski, P. (2002). Iterative optimizing set-point control – the basic principle redesigned, Proc. 15th IFAC World Congress, Barcelona, Paper T-Th-E16-3.
Till, J., G. Sand, M. Urselmann, and S. Engell (2007). A hybrid evolutionary algorithm for solving two-stage stochastic integer programs in chemical batch scheduling. Comp. Chem. Engg. 31, 630-647.
Tometzki, T., and S. Engell (2009). A hybrid multiple populations evolutionary algorithm for two stage scheduling problems with disjunctive decisions. Proc. 19th European Symposium on Computer Aided Process Engineering.
Toumi, A., and S. Engell (2004). Optimization-based Control of a Reactive Simulated Moving Bed Process for Glucose Isomerization. Chem. Eng. Sci. 59, 3777-3792.
Van den Heever, S., and I.E. Grossmann (2003). A strategy for the integration of production planning and reactive scheduling in the optimization of a hydrogen supply chain network. Comp. Chem. Engg. 27, 1813–1839.
Verderame, P.M., and C.A. Floudas (2008). Integrated Operational Planning and Medium-Term Scheduling of a Large-Scale Industrial Batch Plants. Ind. Eng. Chem. Res. 47, 4845-4860.
Vin, J. P., and M. Ierapetritou (2000). A new approach for efficient rescheduling of multiproduct batch plants. Ind. Eng. Chem. Res. 39, 4228- 4283.
Vin, J. P., and M. Ierapetritou (2001). Robust short-term scheduling of multiproduct batch plants under demand uncertainty. Ind. Eng. Chem. Res. 40, 4543–4554.
Wilkinson, S.J., N. Shah, and C.C. Pantelides (1995). Aggregate modeling of multipurpose plant operation. Comp. Chem. Engg. 19, 583-588.
Wu, D. and M. Ierapetritou (2007). Hierarchical approach for production planning and scheduling under unvertainty. Chem. Eng. Proc. 46, 1129-1140.
Zentner, M.G., J.F. Pekny, G.V. Reklaites, and J.N.D. Gupta (1996). Practical considerations in using model-based optimization for the scheduling and planning of batch/semicontinuous processes. J. Proc. Control 35, 259-280.