Clean Techn Environ Policy (2010) 12:105–115
DOI 10.1007/s10098-009-0240-5

ORIGINAL PAPER

Using S-graph to address uncertainty in batch plants

José Miguel Laínez · Máté Hegyháti · Ferenc Friedler · Luis Puigjaner

Received: 2 June 2009 / Accepted: 7 June 2009 / Published online: 7 July 2009

Abstract Processes and markets uncertainties make batch plants a complex environment to manage production activities. Uncertainties may cause deviations and infeasibilities in predefined schedules; this may result in poor planning and inefficient utilization of materials. Consequently, the relevance of explicitly incorporating variability in the scheduling formulation in order to offer more efficient plans and robust decisions to changes has become recognized. This work addresses the batch plants scheduling under exogenous uncertainty. The most widely utilized approach to tackle this problem is stochastic programming; however its solution results in high computational expenses. From another standpoint S-graph, a graph-theoretic approach, has proved to be very efficient to deal with deterministic scheduling. In this work, the S-graph framework is enhanced so that stochastic scheduling problems can be handled. For this purpose, a LP model that is used as performance evaluator has been coupled with S-graph framework. One of the main advantages of the proposed approach is that the search space does not increase according to the number of scenarios considered in the problem. Finally, the potential of the proposed framework is highlighted through two illustrative examples.

J. M. Laínez · L. Puigjaner (✉)
Department of Chemical Engineering,
Universitat Politècnica de Catalunya, ETSEIB,
Avda. Diagonal, 647, 08028 Barcelona, Spain
e-mail: Luis.Puigjaner@upc.edu

M. Hegyháti · F. Friedler
Department of Computer Science,
Faculty of Information Technology,
University of Pannonia, Egyetem u. 10,
8200 Veszprém, Hungary

Keywords Scheduling · Uncertainty · S-graph

List of symbols

Indices

P Products
R Production routes
S Scenarios

Sets

R_p Routes that process product p

Parameters

$B_{p,r}^{\max}$ Maximum batch size for product P that route r can produce, kg
$\mathrm{Cost}_p^{\mathrm{over}}$ Overproduction cost for product P, c.u
$\mathrm{Cost}_p^{\mathrm{under}}$ Underproduction cost for product P, c.u
$\mathrm{dem}_{p,s}$ Demand for product P under scenario s, kg
p_s Scenario S probability of occurrence
price_p Market price for product P, c.u
$SF_{p,r}^{\min}$ Minimum batch size for product P allowed at route r expressed as a proportion of the maximum batch size

Discrete variable

N_{pr} Number of batches of product P that are processed using route r. (Please notice that this is a variable for the whole problem, however it is a given parameter for the LP-performance evaluator)

Continuous Variables

$E[\mathrm{profit}]$ Total expected profit, c.u
$o_{p,s}$ Overproduction of product P in scenario s, c.u
profit_s Profit accomplished under scenario s, c.u
$u_{p,s}$ Underproduction of product P in scenario s, c.u
$x_{p,r}$ Batch size for product P using route r as a proportion of the maximum batch size, c.u

Introduction

The scheduling of production facilities can be generally defined as a decision-making process that answers the questions of how, where, and when to produce a set of products in order to satisfy customer demand. How refers to the plant resources required (processing units, steam, electricity, raw materials, manpower, etc.); the question where is answered by allocating every operation to a specific unit; finally, when consists of predicting the start and end times for each operation (Pekny and Reklaitis 1998). It is evident that uncertainties in batch operations may arise from different sources (i.e. external demand, prices of raw and final products, processing times and equipment availability) causing previous schedules to become non-optimal and in some cases infeasible. Despite the uncertain nature of scheduling problems, research efforts over last decades have primarily focused on deterministic formulations which assume all parameters to be precisely known in advance.

One of the first contributions to this field is the work of Kondili et al. (1993). They developed the state-task-network (STN) representation in order to formulate the problem of production scheduling in multipurpose plants as an MILP. Later on, Pantelides (1994) presented the resource-task-network (RTN) representation which employs a uniform treatment for all available resources. Hence, RTN reduces the number of binary variables and equations when compared with the STN. Both, the STN and RTN formulation used a discrete time representation. Pinto and Grossmann (1995) extend the STN formulation to a continuous time representation. For this purpose, they proposed to use a set of global time slots with unknown duration for allocating units to tasks. Similarly, Castro et al. (2001) extend the RTN framework to a continuous time formulation. Lin et al. (2002) used the concept of event points. The global time point representation is relaxed by allowing different tasks to start at different moments in different units for the same event point. The state sequence network (SSN) formulation developed by Majozi and Zhu (2001) consists in a continuous formulation which eliminates the use of task and unit, thus reducing the number of binary variables compared to other continuous formulations. Finally, Cerda and co-workers (Cerda et al. 1997; Méndez and Cerda 2002) developed precedence based models which are suitable for cases where sequence-dependent changeovers are to be considered.

The approaches developed so far to address the problem of decision making under uncertainty can be generally classified into two groups, i.e. reactive and preventive procedures. Reactive scheduling attempts to modify a nominal schedule obtained by a deterministic formulation so as to adopt it to changes. Intelligent agents and dispatching rules are commonly used to perform the schedule modifications. On the other hand, preventive approaches explicitly take into account uncertainties into the problem formulation. Stochastic programming (SP) is the most commonly adopted approach in the literature for preventive scheduling. A solution with the maximum expected performance is obtained by including estimated scenarios in the formulation. These estimated scenarios are generated by representing uncertain parameters as random variables. Their goal is to find a solution that is feasible for all the possible data scenarios and which maximizes the expectation of a performance indicator. The most widely applied SP models are two-stage programs. In models of this type, the decision maker takes some actions in the first stage, after which a random event occurs and affects the outcome of those first-stage decisions. A recourse decision can then be made in the second stage that compensates for any negative effects that might have been experienced as a result of the first-stage decisions.

Scheduling problems are highly complex problems. Due to the discrete decisions involved (e.g., equipment assignment, task allocation over time) these problems are inherently combinatorial in nature, and hence very challenging from the computational complexity point of view (Pekny and Reklaitis 1998). Therefore, a modest growth in problem size can lead to a significant increase in the computational requirements (Lin and Floudas 2004). Furthermore, stochastic programs become deterministic equivalent programs with the utilization of scenarios or scenario tree. The size of the deterministic scheduling formulation can easily grow out of hand for a large number of scenarios, which renders the direct solution approaches numerically intractable and thus necessitates special methods, such as decomposition and aggregation (Cheng et al. 2004). Hence, it turns out that one of the major challenges in the area of scheduling under uncertainty is to reduce the computational cost required to solve this kind of problems (NP complete problems which are complicated by the consideration of uncertainty). It is noteworthy that solution procedures based on knowledge of the specific problem have been recognized to exhibit a good potential in providing advances in this direction (Li and Ierapetritou 2008).

S-graph is a scheduling approach that has proven to significantly reduce the computational effort compared to mathematical programming techniques. S-graph is a representation that takes into consideration the specific characteristics of chemical processes in scheduling. It allows for the formulation of scheduling problems using similar graph representations as those used to solve the job-shop problem but contemplating the higher complexity of the chemical multipurpose batch scheduling (Sanmartí et al. 2002). Moreover, one of its important capabilities is that it offers a strictly continuous time formulation. Initially, this approach was only applied to problems in which the objective was to minimize makespan. The problem was solved using a Branch and Bound and an efficient graph algorithm to evaluate the makespan. Recently, S-graph has been extended to be an effective search algorithm for determining schedules that optimize throughput, revenue, or profit over a predefined time horizon in multipurpose batch plants (Majozi and Friedler 2006).

The work presented in this paper starts from the above mentioned latest framework. A new extension of S-graph that allows tackling scheduling problems under external uncertainty (demand and prices) is shown. The resulting schedule is equivalent to the one that can be obtained using two-stage stochastic programming techniques. We demonstrate the capabilities of the proposed approach in addressing stochastic scheduling problems by solving two illustrative examples.

The paper is organized as follows. In "Problem statement" a formal definition of the problem of interest is given. A brief introduction to S-graph presents a short overview to S-graph representation. In Enhancing S-graph framework to approach scheduling under uncertainty the specific algorithm used to deal with scheduling under uncertainty is derived and its performance is highlighted through illustrative examples in Literature examples. Finally, some conclusions about this work are drawn in Conclusions.

Problem statement

In last decade many authors have recognized that it is unlikely to apply deterministic schedules in real scenarios without decreasing considerably their performance, and have made efforts to extend deterministic approaches to situations with some type of uncertainty, so as to obtain better results when their solutions are deployed in real scenarios. Here, we extend the S-graph deterministic framework for solving scheduling problems under uncertainty in demand.

Generally speaking, the scheduling problem seeks the best way of allocating and timing the different production tasks to the available resources. In this work, the scheduling problem aims at maximizing the business profit. The problem input data can be classified into three groups: (1) resources related data, (2) process related data and (3) economic data.

The first group describes all the enterprise available equipment such as processing and storage units. Data regarding its maximum capacity and minimum working capacity are usually required. Moreover, maximum availability of raw materials may be relevant depending on the problem scope. The process related data describes the different recipes that may be followed in order to obtain the final products. Here, the different tasks required to produce an intermediate or final product are determined. The input and output materials for each task are stipulated as well as their respective mass proportions and processing times. Additionally, the equipment that is suitable to perform each task is identified. Other relevant data may be the energy, vapor or any other utility consumption for each task.

Finally, the last group is concerning all the data required to quantify the expected net benefits due to production operations. On the outcome side, raw materials and operation costs are included. The revenues are generated by selling the final products to the target marketplace, thus two important input parameters to define are the market price and demand of each product. It is important to point out that given the exogenous stochastic nature of the problem tackled in this paper, demand is considered as uncertain. Commonly, random parameters are described by using a probability distribution function. Instead, the scenario approach is adopted in this work so that the stochastic scheduling problem can be formulated using a deterministic equivalent approach. In case that demand probability distribution function are available, a Monte Carlo sampling can be executed in order to obtain satisfactory equally probable scenarios of demand.

The proposed stochastic S-graph framework is intended to support plant managers on the decision making about the timing of tasks to be performed in each processing unit, the amount of material being processed at each time in each unit, and the amount of final products to be sold in each demand scenario. These decisions will be taken such that the expected profit evaluated at the end of a predefined planning horizon is maximized.

A brief introduction to S-graph

A detailed description of the S-graph framework has been presented in the works of Sanmartí et al. (2002) and Romero et al. (2004). Additionally, the reader is referred to the web site http://www.s-graph.com (University of Pannonia, Faculty of Information Technology, last visited 10 April 2009) for further information. For comprehensiveness a short description is given here.

Graph representation of scheduling problems

The S-graph framework consists of a sophisticated graph theoretic model developed to address the deterministic scheduling problem in multipurpose batch plants. S-graph was originally designed for makespan minimization problems assuming Non Intermediate Storage (NIS) policy. Later works have extended the framework so that other operational policies (common intermediate storage, fixed intermediate storage, zero wait) can be considered.

In an S-graph, the nodes correspond to production tasks except terminal nodes which are to denote the final products. The S-graph arcs are classified into two classes; the so-called recipe arcs and schedule arcs. It is noteworthy that recipe arcs are an input to the scheduling problem, while schedule arcs result from the S-graph algorithm solution.

Recipe arcs represent the preceding relationship among tasks. If a recipe arc leads from task k_1 to task k_2 means that task k_2 execution must start at least $c(k_1, k_2)$ time units later than task k_1 execution. Here, $c(k_1, k_2)$ is the weight of the recipe arc (k_1, k_2). In case of problem initialization and more than one equipment is suitable to perform a recipe arc (k_1, k_2) (i.e. execution of task represented by the origin node k_1), the arc weight is the minimum processing time for task k_1 among the suitable equipment units.

On the other hand, schedule-arcs denote the sequencing of tasks assigned to the same equipment unit. Assume that according to the scheduling, task k_1 and k_2 are assigned to equipment unit E_1 and additionally, these tasks will be performed in the sequence k_1–k_2. Then, a zero-weighted schedule arc (or an arc with the length of change over time if appropriate) is added from all immediately subsequent tasks of k_1 in the recipe to task k_2. A graph without any schedule-arc is called recipe-graph, otherwise it is termed schedule-graph. When all tasks have been sequenced for all units, a complete schedule-graph have been generated. Note that one schedule graph exists for each feasible schedule. Therefore, an S-graph is given in the mathematical form $G(N,A_1,A_2)$, where N, A_1 and A_2 denote the sets of nodes, recipe arcs, and schedule arcs, respectively. In Fig. 1 is shown a recipe-graph, while Fig. 2 depicts a complete schedule-graph.

One of the special features of S-graph is that feasible schedules can be straightly identified. Loops must not appear in an S-graph corresponding to a feasible schedule. Following an appropriate Branch and Bound search strategy, the S-graph of the global optimal schedule can be

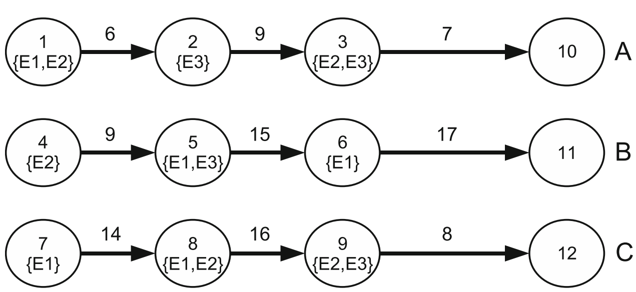

Fig. 1 A recipe graph

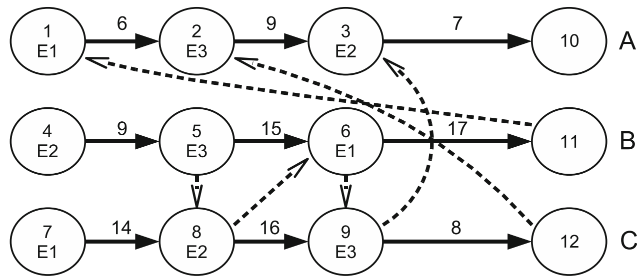

Fig. 2 A schedule graph

efficaciously found. Please refer to Sanmartí et al. (2002) for details regarding the search strategy.

Throughput maximization using S-graph

Majozi and Friedler (2006) had recently extended the S-graph framework so that problems that involve economic performance indicators can be tackled. Specifically, they addressed the throughput maximization problem during a fixed time horizon, but their approach can be certainly extended to consider other indicators such as cost and profit.

The optimization strategy they proposed can be understood as comprised of two components: an optimality search algorithm and a feasibility test.

Feasibility test

Having in mind that a node (P_i) in the search space corresponds to a discrete combination of batches of products, the feasibility test of a node basically consists in: (1) finding the minimum makespan schedule graph for the specific combination of batches of products and (2) a comparison between the minimum makespan obtained and the fixed time horizon. Clearly, the schedule graph is feasible if the minimum makespan obtained is less or equal than the time horizon length.

Next, the optimality search algorithm is briefly explained. The algorithm roughly consists in rules that allow reducing the search space without losing optimality. Given a set of products p, the number of product p batches associated with node i is represented by N_p. Using the feasibility test it can be found the maximum number of batches of each of the products that can be processed over the time horizon of interest (N_p^u). Once the infeasibility of a node P_i belonging to this new reduced search region has been proved, any other node $P_{i'}$ that accomplished that $N_p{}' \geq N_p$ for all products P is infeasible as well (Fig. 3). Here $N_p{}'$ is the number of batches of product P at node P_i and N_p is the number of batches of product P at node P_i. It can be noticed that the efficiency of the search results from the elimination of redundancy since at each search point a

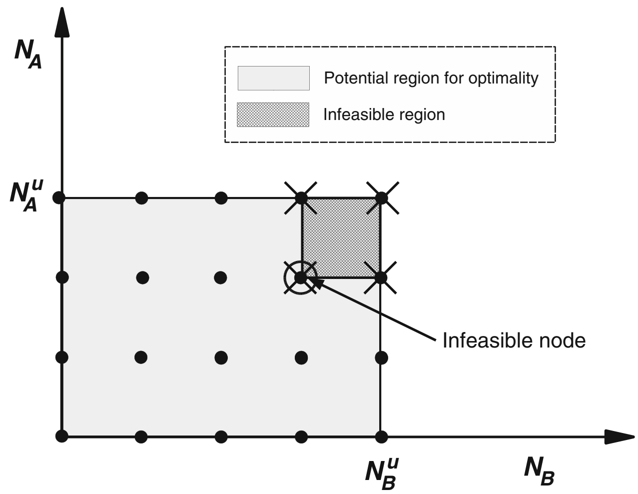

Fig. 3 Procedure to reduce search region when it is found a unfeasible node

node with a unique combination of batches of products is explored. Furthermore, regions with no opportunity for optimality are identified and eliminated a priori from the search algorithm.

Enhancing S-graph framework to approach scheduling under uncertainty

In this section we describe the framework proposed for solving stochastic scheduling problems which basically consists in a systematic search strategy based on: the schedule generator (S-graph) and the expected performance evaluator (LP Model). The algorithm flowsheet is presented in Fig. 4.

The first step of the algorithm is to define the search space. This consists in the set of nodes corresponding to different combination of production routes for final products. Routes are those different realistic ways to process a product or combination of products. Then, a node **N** could be described by an $|R||P|$ dimensional integer matrix, where each component $N_{p,r}$ represents the number of batches of product p to produce from route r. The procedure that is followed to define the search space is the same described in Throughput maximization using S-graph. For a general case, this procedure finds the maximum number of batches for each of product p that route r can process over the time horizon of interest ($N^u_{p,r}$) (see section Feasibility test). Here, **toTest** is defined as the set of nodes that have not been tested yet but still have an opportunity to result in a higher expected profit. The initial search space is used to initialize **toTest** .

The schedule resulting with the higher expected profit is selected from the nodes found during the definition of the search space ($N^u_{p,r}$). Such schedule is used to initialize **cb_value** and **cb_schedule** which represent the best expected profit currently found and its corresponding schedule, respectively.

The iterative part of the algorithm is explained next. While **toTest** is not empty, a node is chosen and saved in **cnode**. That node is then deleted from **toTest**. It is noteworthy that the algorithm may be accelerated by defining properly a strategy for choosing this node. Afterwards, the expected profit is computed for **cnode**. This is possible by solving the LP problem described in subsection Expected performance evaluator: an LP. If **cnode** expected profit is less than the current best expected profit (**cb_value**), then this node is not the optimal and it is not tested for feasibility. Otherwise, **cnode** is tested for feasibility; if it has a feasible schedule within the time horizon, such schedule will be the new value for **cb_schedule**, and its corresponding expected profit will be the new value for **cb_value**. In case the **cnode** is infeasible, any other node **N'** that accomplishes that $N'_{p,r} \geq N^{\text{cnode}}_{p,r}$ for all routes r and products p is infeasible as well, so it should be also removed from **toTest** as described in section 3.2.

If **toTest** is empty, **cb_value** is the expected profit corresponding to the optimal solution stored in **cb_schedule**. Otherwise, the iterative part of the algorithm is repeated as above described.

Expected performance evaluator: an LP

In this section the Linear Program for node expected profit evaluation is described. Each node corresponds to a given number of batches for each product-route. Since overproduction has as penalty the carrying inventory cost, it is not always worth to work at full capacity. The LP model allows determining the batch sizes for each route that maximize the expected profit at each evaluated node. The formal mathematical description is stated as follows.

Given:

- Market related inputs

P	Set of products
Price_p	Product price
$\text{Cost}^{\text{over}}_p$	Overproduction cost for each product
$\text{Cost}^{\text{under}}_p$	Underproduction cost for each product

- Recipe related inputs

R	Set of routes
$B^{\max}_{p,r}$	The maximum batch size for product p that route r can produce
$SF^{\min}_{p,r}$	The minimum batch size allowed for product p at route r expressed as a proportion of the maximum batch size

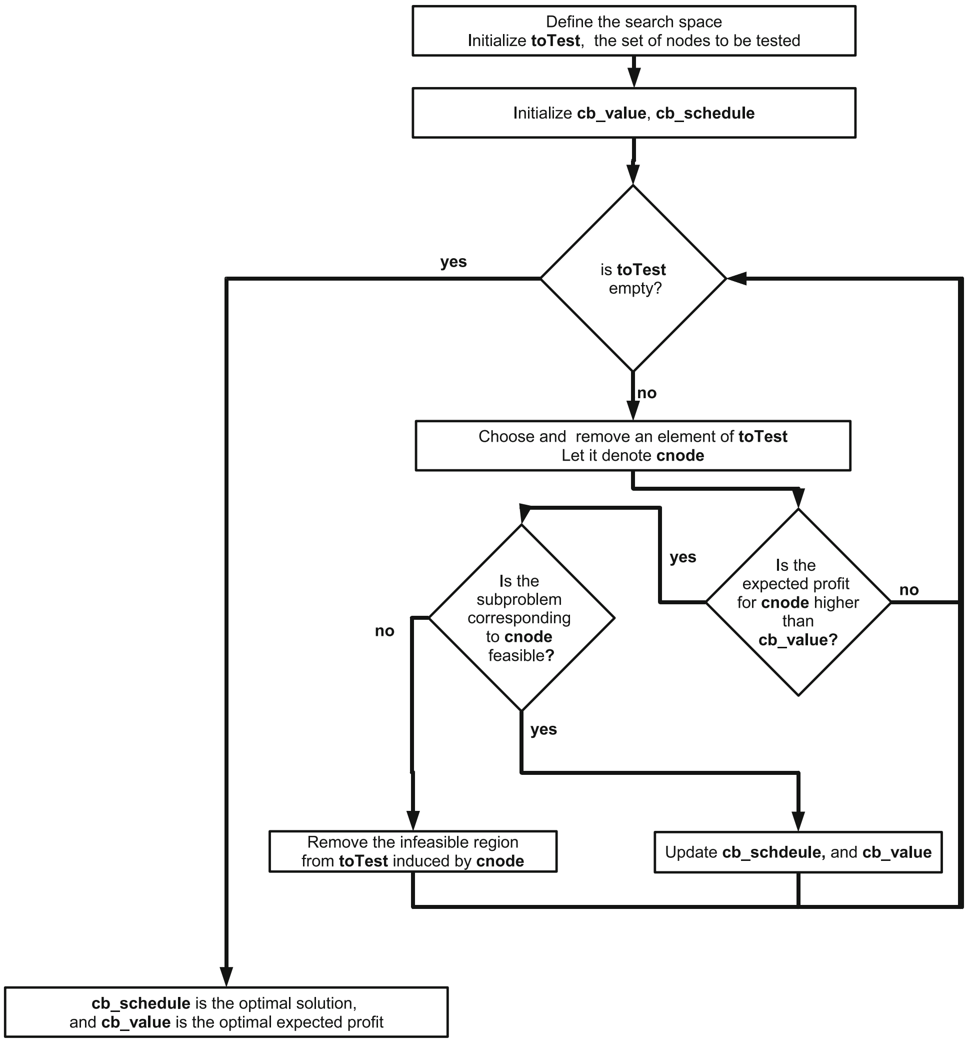

Fig. 4 Flowsheet of the proposed algorithm

- Scenario related inputs

S	Set of scenarios
P_s	Scenario probability
$\mathrm{dem}_{p,s}$	The product p demand for scenario s

- Node data

$N_{p,r}$	Number of batches produced using route r for the node being evaluated

The goal is to determine:

$x_{p,r}$	Batch size as a proportion of the maximum batch size
$u_{p,s}$	Underproduction of product P in scenario s
$o_{p,s}$	Overproduction of product P in scenario s

Such that the expected profit is maximized.

The linear problem equations can be classified in three groups, namely (1) batch size equations, (2) demand satisfaction, and (3) the objective function.

Batch size equations

Equation 1 states that product p batch size for each route ($x_{p,r}$) is bounded in the range of $\left[SF_{p,r}^{\min}, 1\right]$ which represents the interval where it must fall.

$$SF_{p,r}^{\min} \leq x_{p,r} \leq 1 \quad \forall p, r \in R_p \tag{1}$$

To avoid overlapping among search regions, $x_{p,r}$ is forced to be greater or equal than $\frac{N_{p,r}-1}{N_{p,r}}$ if $N_{p,r}$ is not equal to zero. The following constraint expresses this requirement:

$$N_{p,r} - 1 \leq N_{p,r} x_{p,r} \quad \forall p \in P, r \in R_p \tag{2}$$

Demand satisfaction

Equation 3 expresses that market sales must be less than or equal to demand ($\mathrm{dem}_{p,s}$). Here, $u_{p,s}$ and $o_{p,s}$ represent the shortage and excess of product P over its corresponding demand in scenario s, respectively. This equation states that the total product demand must be equal to the amount of product P processed by this node $\left(\sum_{r\in R_p} x_{p,r}B^{\max}_{p,r}N_{p,r}\right)$ plus the shortage, minus the overproduction. Since demand depends on the disclosed scenario, we have $|P||S|$ equations.

$$\sum_{r\in R_p} x_{p,r}B^{\max}_{p,r}N_{p,r} + u_{p,s} - o_{p,s} = \mathrm{dem}_{p,s} \quad \forall p\in P, s\in S \tag{3}$$

Objective function: expected profit

Equation 4 is to calculate the profit associated to each scenario s. The objective function summarizes the revenues associated to sales and the costs of not exactly meeting the demand (shortages and overproduction). As Eq. 4 states, product p market sales is equal to the amount produced $\left(\sum_{r\in R_p} x_{p,r}B^{\max}_{p,r}N_{p,r}\right)$ minus the overproduced amount ($o_{p,s}$). Here, it is noteworthy that overproduction is equivalent to the product inventory held at the end of the scheduling horizon. Hence, the unitary overproduction cost ($\mathrm{Cost}^{\mathrm{over}}_p$) can represent the carrying cost of inventory associated with a specific product p. On the other hand, underproduction cost is related to those costs incurred when an item is out of stock (shortage or backorders). Businesses usually quantify these costs including the lost contribution margin on sales plus lost customer goodwill.

$$\mathrm{profit}_s = \mathrm{price}_p \sum_{p\in P}\left(\sum_{r\in R_p} x_{p,r}B^{\max}_{p,r}N_{p,r} - o_{p,s}\right) - \left(\mathrm{Cost}^{\mathrm{over}}_p o_{p,s} + \mathrm{Cost}^{\mathrm{under}}_p u_{p,s}\right) \quad \forall s\in S \tag{4}$$

Once the scheduling decisions have been assessed in each possible scenario by Eq. 4, Eq. 5 calculates the expected profit by considering the scenarios probability of occurrence (P_s). The expected profit is the LP objective function under the assumption that the decision maker is neutral about risk.

$$E[\mathrm{profit}] = \sum_{s\in S} p_s \mathrm{profit}_s \tag{5}$$

The LP problem for evaluating the expected profit can be then mathematically posed as follows:

$$\max_{x_{p,r},o_{p,s},u_{p,s}} \quad E[\mathrm{profit}]$$

subject to
Equations 1–5

$$x_{p,r}, o_{p,s}, u_{p,s} \in \mathbb{R}^+$$

Extension to other uncertain parameters Market uncertainty usually escalates not merely on product demands, but also on product prices. In the strict mathematical formulation that means, that instead of just having $\mathrm{dem}_{p,s}$ as uncertain parameter; the input for the algorithm would consider $\mathrm{price}_{p,s}$ as random parameter as well. However, it is important to notice that such extension does not cause any alteration on the LP model. The exploration of the search space, the feasibility testing, the variables, and constraints in the LP remain the same. The profit equation is the only change required in order to generalize the problem in this manner:

$$\mathrm{profit}_s = \mathrm{price}_{p,s} \sum_{p\in P}\left(\sum_{r\in R} x_{p,r}B^{\max}_{p,r}N_{p,r} - o_{p,s}\right) - \left(\mathrm{Cost}^{\mathrm{over}}_{p,s} o_{p,s} + \mathrm{Cost}^{\mathrm{under}}_{p,s} u_{p,s}\right) \quad \forall s\in S \tag{6}$$

Since the proposed algorithm is based on the Throughput Maximization method published by Majozi and Friedler (2006), it renders the advantages of that algorithm and the S-graph framework. Examples of those advantages are (1) globally optimal solutions are obtained, (2) no infeasible solutions are found in terms of cross-transfers (Friedler et al. 2008), (3) search space significant reduction, and (4) it consists in a continuous formulation without the necessity of determining the so-called time points.

Usually, stochastic MILP models are very sensitivity to the number of considered scenarios. Indeed, the size of MILP model increases dramatically by increasing the number of scenarios and accordingly the needed computational effort. By using the proposed algorithm, the computational time for the expected profit calculation will be increased; nevertheless the search space does not grow by increasing the number of scenarios. Notice that the search space size merely depends on route combinations. As a result, the computational burden required to solve industrial stochastic scheduling problems can be reduced significantly by using the proposed algorithm.

Literature examples

The capabilities of the proposed framework are illustrated by solving the next two illustrative examples.

Example 1

Consider the following example introduced by Majozi and Friedler (2006) in which two products (product A and B) are to be produced, according to the recipes given in Figs. 5, 6. Five different equipment units are available. The suitability of equipment units are shown in the recipe figures. Each unit of product A has a market price of 30 c.u, whereas product B has a price of 10 c.u. It is assumed that overproduction and under-production cost are equal to 15 and 25% of the market price, respectively. Three scenarios are considered for this example. The data related to product demands in each scenario is presented in Table 1. In this case the objective is to maximize expected profit over a time horizon of 60 h under Non Intermediate Storage (NIS) policy.

Recalling the algorithm of Enhancing S-graph framework to approach scheduling under uncertainty, the search region contains 15 nodes. The search region is illustrated in Fig. 7. As shown in this Figure, five nodes are tested for feasibility. The optimal solution is found in node (2, 2) which exhibits an expected profit of 3,165.20 c.u. The optimal solution comprises two batches of product A as well as two batches of product B. The corresponding batch sizes are shown in Table 2. In Fig. 8 the optimal schedule obtained for this example is depicted.

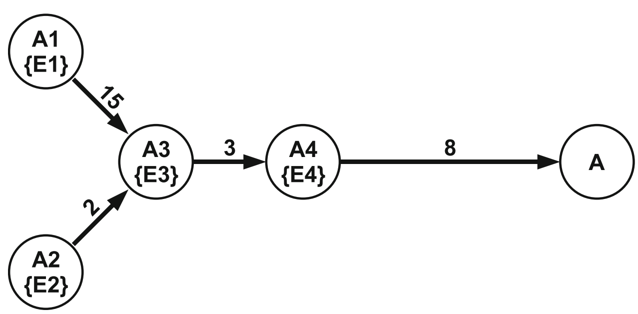

Fig. 5 Route product A for example 1

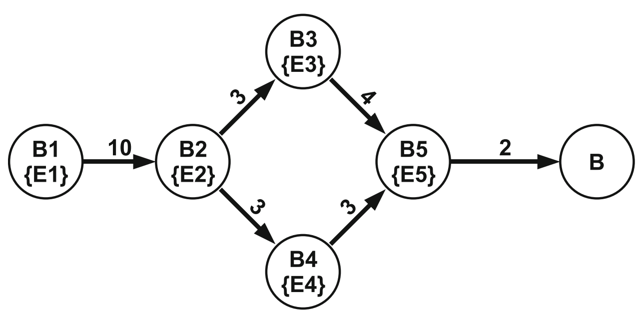

Fig. 6 Route product B for example 1

Table 1 Scenario data for illustrative example 1

Scenario	Demand (kg)		Probability
	*P*1	*P*2	
I	58	64	0.30
II	100	92	0.40
III	148	62	0.30

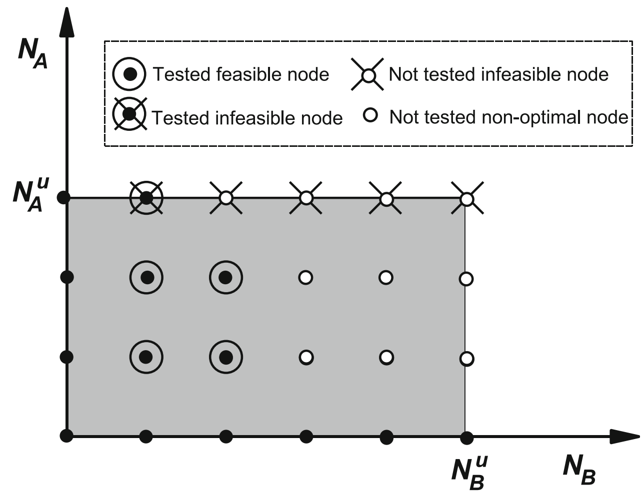

Fig. 7 Search space for example 1

Table 2 Optimal batch quantity and sizes for illustrative example 1

Route	Quantity	Batch size proportion
A	2	1.00
B	2	0.92

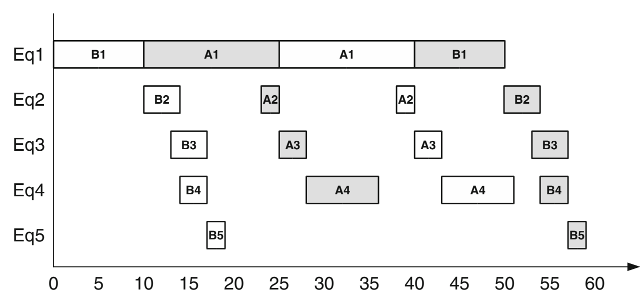

Fig. 8 Optimal schedule for example 1

Example 2

This example was first presented by Kondili et al. (1993). Two products are produced from three feeds according to the STN shown in Fig. 9. The STN utilizes five tasks which can be performed in four different units. The corresponding operational data for the example including units, tasks, and materials is given in Tables 3, 4, and 5. Six scenarios are considered in this problem. Table 6 shows scenario product demands and the probability corresponding to each of them. The objective is to maximize the expected profit within a time horizon of 18 h following NIS policy.

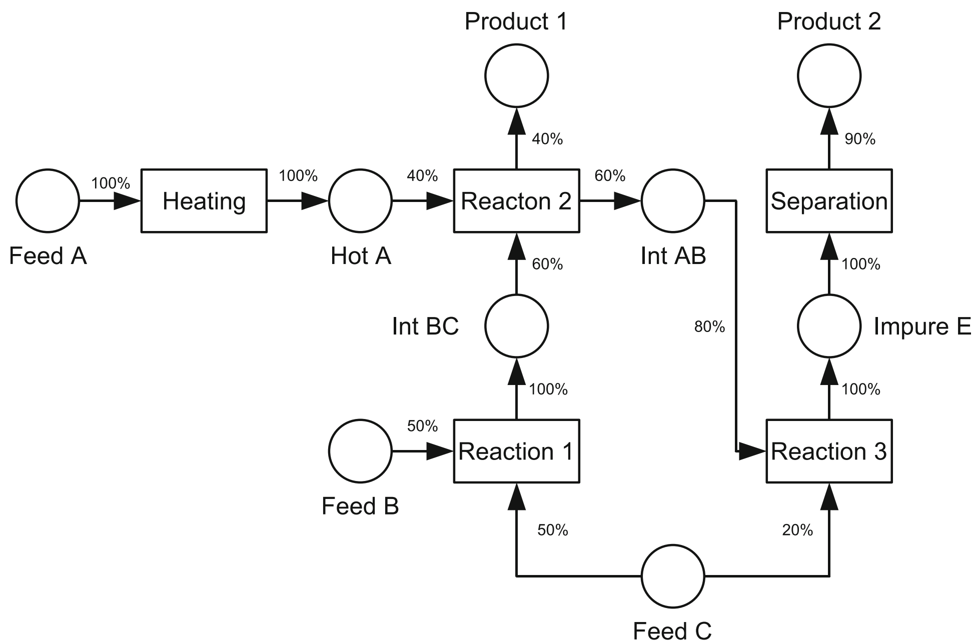

Fig. 9 State-task network of example 2

Table 3 Unit data for illustrative example 2

Unit	Maximum capacity (kg)	Suitable for task
Heater	100	Heating
Reactor 1	50	Reaction 1, 2, 3
Reactor 2	80	Reaction 1, 2, 3
Separator	200	Separation

Table 4 Material data for illustrative example

States	Storage Capacity (kg)	Market Price (c.u.)	Overproduction Cost (c.u.)	Underproduction Cost (c.u.)
Feed A	Unlimited	0.00	0.00	0.00
Feed B	Unlimited	0.00	0.00	0.00
Feed C	Unlimited	0.00	0.00	0.00
Hot A	0.00	0.00	0.00	0.00
IntAB	0.00	0.00	0.00	0.00
IntBC	0.00	0.00	0.00	0.00
Impure E	0.00	0.00	0.00	0.00
Product 1	Unlimited	10.00	2.50	1.50
Product 2	Unlimited	10.00	2.50	1.50

For this example six different routes to produce final products exist which are depicted in Figs. 10, 11, 12, 13, 14, 15. The aforementioned fact leads to a six-dimension search region that includes 6,400 nodes. Before finding the optimal solution 318 nodes need to be tested for feasibility. The optimal solution has an expected profit of 2,475.31 c.u. The optimal schedule is shown in Fig. 16 and the corresponding batch sizes are given in Table 7.

Conclusions

A new approach for solving scheduling problems under exogenous uncertainty is presented. The approach is based on the S-graph framework which has proven to be a rigorous and efficient tool for solving deterministic scheduling problems.

The proposed framework does not only inherit the advantages of S-graph, but it also has an advantage against stochastic programming techniques; namely the computational effort needed to solve the problem does not increase by increasing the number of scenarios. Such convenience relies on the fact that the search space size is independent on the number of considered scenarios. The size is uniquely dependent on the route-product batches combination. As the number of scenarios increase a larger LP is to be solved but still due to its nature the computational times are very small. Therefore, the presented framework has a great potential to solve industrial scale problems of scheduling under uncertainty.

Finally, the authors would like to highlight the contribution of this work to the clean technologies field. Cleaner production is proposed as an integral and preventive strategy that aims at minimizing the environmental impacts from industrial products and services. One of its main activities is to identify options to minimize waste and emissions out of industrial processes. This does include

Table 5 Task data for illustrative example 2

Task	Processing time (h)
Heating	1.0
Reaction 1	2.0
Reaction 2	2.0
Reaction 3	1.0
Separation	2.0

Table 6 Scenario data for illustrative example 2

Scenario	Demand (kg)		Probability
	*P*1	*P*2	
I	102.3	174.8	0.167
II	148.8	344.2	0.167
III	158.6	128.2	0.167
IV	0.0	225.1	0.167
V	72.0	109.1	0.167
VI	54.6	268.8	0.167

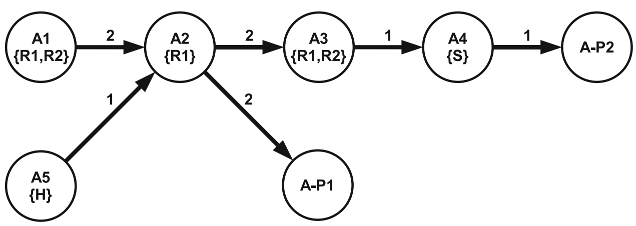

Fig. 10 Route A for example 2

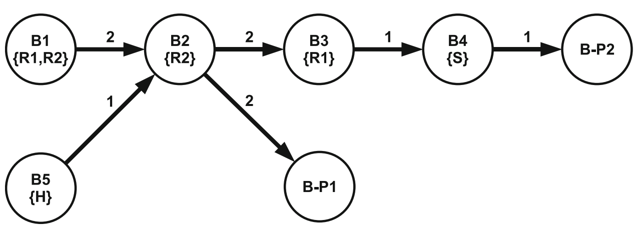

Fig. 11 Route B for example 2

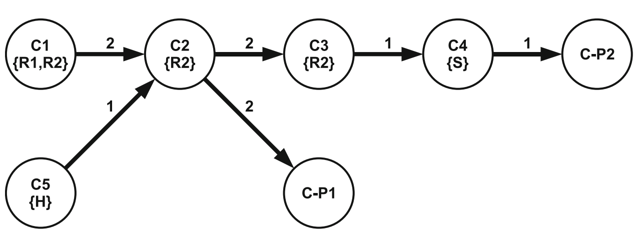

Fig. 12 Route C for example 2

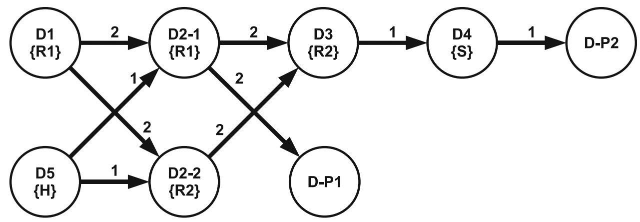

Fig. 13 Route D for example 2

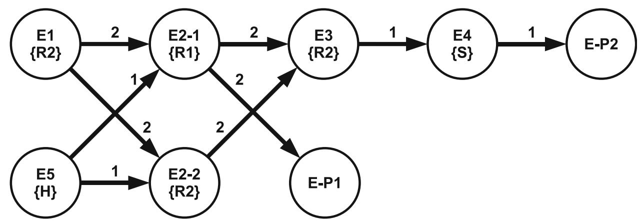

Fig. 14 Route E for example 2

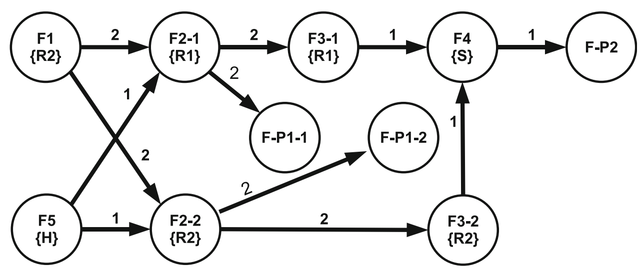

Fig. 15 Route F for example 2

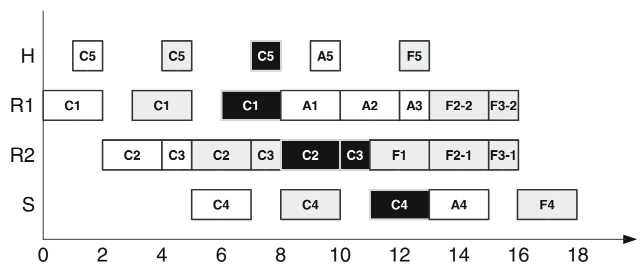

Fig. 16 Optimal schedule for example 2

Table 7 Optimal batch quantity and sizes for illustrative example 2

Route	Quantity	Batch size proportion
A	1	1.00
B	0	0.00
C	3	0.91
D	0	0.00
E	0	0.00
F	1	1.00

identifying losses from poor planning and suggesting better choices in the utilization of materials. The S-graph approach presented in this work supports managers on deciding how to better use materials and allocate equipment to the production of final products in the face of uncertainty, hence collaborating with the cleaner production purposes. What's more, cleaner production is regarded as a proactive philosophy as a key difference from pollution control. The proposed framework follows such philosophy; it prevents the accumulation of inventory and assists to efficiently maximize the incomes due to production outputs by anticipating and taking into account different probable demand scenarios.

Current work is going towards converting this approach into an exact one by considering explicitly the probability distribution of product demands in the LP model and not just a discrete number of demand scenarios. Research efforts are also devoted to develop accelerating algorithms for this stochastic S-graph framework.

Acknowledgments Financial support received from "Generalitat de Catalunya" (FI programs) is acknowledged. Also support received from the Hungary-Spain Integrated Action (HH2004-0025) is thankfully acknowledged. Besides, financial support from projects Xartap (I0898) and ToleranT (DPI2006-05673) is fully appreciated. Valuable suggestions of Tibor Holczinger, Aaron Bojarski and Thokozani Majozi have enriched this work, for which the authors express their gratitude.

References

Castro P, Barbosa-Pavoa A, Matos H (2001) An improved RTN continuous-time formulation for the short-term scheduling of multipurpose batch plants. Ind Eng Chem Res 40:2059–2068

Cerda J, Henning GP, Grossmann IE (1997) A mixed-integer linear programming model for short-term scheduling of single-stage multiproduct batch plants with parallel lines. Ind Eng Chem Res 36:1695–1707

Cheng L, Subrahmanian E, Westerberg AW (2004) A comparison of optimal control and stochastic programming from a formulation and computation perspective. Comput Chem Eng 29:149–164

Friedler F, Majozi T, Holczinger T (2008) Implications of cross-transfer in batch plants with complex recipes: S-graph vs MILP methods. In: Proceedings of 11th conference on process integration, modelling and optimisation for energy saving and pollution reduction—PRES 2008 P5.129, Prague: Orgit Ltd publicatios, ISBN: 978-80-02-02047-9 (CD-ROM)

Kondili E, Pantelides CC, Sargent R (1993) A general algorithm for short-term scheduling of batch operations. Part 1. MILP formulation. Comput Chem Eng 17:211–227

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

Lin X, Floudas CA (2004) Continuous-time versus discrete-time approaches for scheduling of chemical processes: a review. Comput Chem Eng 28:2109–2129

Lin X, Floudas CA, Modi S, Juhasz NM (2002) Continuous-time optimization approach for medium-range production scheduling of a multiproduct batch plant. Ind Eng Chem Res 41:3884–3906

Majozi T, Friedler F (2006) Maximization of throughput in a multipurpose batch plant under a fixed time horizon: S-graph approach. Ind Eng Chem Res 45:6713–6720

Majozi T, Zhu X (2001) A novel continuous-time milp formulation for multipurpose batch plants. Part 1. Short-term scheduling. Ind Eng Chem Res 40:5935–5949

Méndez CA, Cerda J (2002) An MILP framework for short-term scheduling of single-stage batch plants with limited discrete resources. In: Computer-aided chemical engineering, vol 12. Elsevier Science Ltd, Amsterdam, pp 721–726

Pantelides C (1994) Unified frameworks for optimal process planning and scheduling. In: Rippin DWT, Hale JC, Davis J (eds) Proceedings of the second international conference on foundations of computer-aided process operations. Cache publications, New York, pp 253–274

Pekny JF, Reklaitis GV (1998) Towards the convergence of theory and practice: A technology guide for scheduling/planning methodology. In: Pekny Blau (ed) Proceedings of the third international conference on foundations of computer-aided process operations. Cache publications, New York, pp 91–111

Pinto JM, Grossmann IE (1995) A continuous time mixed integer linear programming model for short-term scheduling of multistage batch plants. Ind Eng Chem Res 34:3037–3051

Romero J, Puigjaner L, Holczinger T, Friedler F (2004) Scheduling intermediate storage multipurpose batch plants using the S-graph. AIChE J 50:403–417

Sanmartí E, Puigjaner L, Holczinger T, Friedler F (2002) Combinatorial framework for effective scheduling of multipurpose batch plants. AIChE J 48:2557–2570