Schedule robustness analysis with the help of attainable sets in continuous flow problem under capacity disruptions

Dmitry Ivanov1* Alexandre Dolgui2 Boris Sokolov3,5 Frank Werner4

1* Berlin School of Economics and Law
Department of Business Administration; Chair of International Supply Chain Management
10825 Berlin, Germany
Phone: +49 30 85789155
E-Mail: divanov@hwr-berlin.de

2 Ecole Nationale Supérieure des Mines
FAYOL-EMSE, CNRS: UMR6158, LIMOS
158, Cours Fauriel, 42023 Saint-Etienne cedex 2, France
E-Mail: dolgui@emse.fr

3 Saint Petersburg Institute for Informatics and Automation of the RAS (SPIIRAS)
V.O. 14 line, 39 199178 St. Petersburg, Russia
E-Mail: sokol@iias.spb.su

4 Otto-von-Guericke University Magdeburg
Faculty of Mathematics
Postfach 4120,
D-39016 Magdeburg, Germany
E-Mail: frank.werner@ovgu.de

5 ITMO University, St. Petersburg, Russia
E-Mail: sokol@iias.spb.su

* Corresponding author
Dmitry Ivanov

Abstract

Continuous flow scheduling problems have their place in many industries such as gas, oil, chemicals, glass and fluids production as well as production of granular goods and steel details. The disruptions in processing capacities may result in schedule performance decrease. In this paper, we develop a new method for robustness analysis of those schedules that are formulated in continuous time in the state-space domain. The developed method is based on attainable sets that allow computing a form to represent the states and performance of schedules in regard to different capacity degradation levels. Having such a form, it becomes possible to estimate the schedule robustness. The technical development and approximation of attainable sets are presented. A robustness index is developed on the basis of the minimax regret approach, and it can be used for decision-makers regarding the trade-off “performance vs. robustness”. With the presented results, it becomes possible to obtain attainable sets for interval data with no a priori information about perturbation impacts, i.e., for non-stationary perturbations. Attainable sets permit to consider perturbations and schedule performances as time functions. Perturbation functions may be set up for different uncertainty scenarios, including interval perturbations.

Keywords: scheduling, robustness, continuous time systems, uncertainty modelling, system dynamics, attainable set, optimal program control
Introduction

Continuous flow scheduling problems have their place in many industries such as gas, oil, chemicals, glass and fluids production as well as production of granular goods and steel details. Since the manufacturing process typically has a multi-stage structure, the issue of capacity disruptions is crucial for the overall schedule performance. The disruptions in processing capacities may result in increase in flow times, makespan, tardiness and decrease in throughput, on-time delivery, and service level. In this setting, schedule robustness analysis becomes an important issue.

Two basic approaches on hedging schedules against the negative impacts of disruptions (e.g., machine breaks) – proactive and reactive scheduling have been recently developed to cope with uncertainty (Vieira et al. 2003, van de Vonder 2007, Dolgui and Proth, 2010). Reactive scheduling aims at adjusting schedules in the presence of unexpected events. Proactive scheduling creates a certain protection and takes into account possible perturbations while generating schedules. In this study, we concentrate on the proactive domain.

In proactive scheduling, one of the crucial issues is to develop measures for the assessment of the schedule protection (Aytug et al. 2005). One possible indicator to assess impacts of uncertainty on the schedule execution is robustness (Daniels and Kouvelis 1995, Wu et al. 1999, Sabuncuoglu and Gören 2009, Hazir et al. 2010, Feng et al. 2012). The details of the methodologies across the works on schedule robustness differ, but most of them consider under robustness that the schedule performance is insensitive to negative impacts of disruptions. A schedule that is able to achieve the planned performance in spite of disruptions is called robust (Sotskov et al. 2013, Sotskov and Werner 2013). Robustness analysis or robust optimization approaches for related problems in assembly line design and scheduling have been considered, for example, in the studies by (Sotskov et al., 2006, Dolgui and Kovalév, 2012a, Dolgui and Kovalév, 2012b, Hazir and Dolgui, 2013, Gurevsky et al., 2012, Gurevsky et al., 2013, Hazir and Dolgui, 2015).

We note that the method developed in this paper is complementary to robust discrete optimization.

In the analysis of schedule robustness, a number of particular features should be taken into account. In practice, decision makers may want to judge on the trade-off between robustness and such performance measures as makespan, flow time, or tardiness (Sotskov and Werner 2013). Frequently, it is desired to have a number of Pareto-optimal schedules in order to take decisions based on individual risk perceptions. Operation processes are subject to a number of variability sources which are frequently non-stationary and cannot always be described by deterministic models. Moreover, it is not always possible to describe the non-stationarity with the help of stochastic models. Frequently, there is no a priori information about many parameters. In other words, disruptions are possible but not predictable. At the same time, the performance of different schedules may be more or less sensitive to parameter variability at different instants of time. As the robustness reflects the dynamic changes in the system, the robustness analysis should also be approached from dynamic perspectives (Vieira et al. 2003).

Even though considerable advancements have been achieved in the analysis of schedule robustness, the studies on schedule robustness from the positions of dynamic system theory are quite rare. In this study,
we develop a new method for a robustness analysis of those schedules that are formulated in continuous
time in the state-space domain. Examples of recent works on application of state-space dynamic models
to scheduling include the studies by Khmelnitsky et al. (1997), Ivanov and Sokolov (2012), and Subramanian et al. (2013).
The developed method is based on attainable (reachable) sets (AS) (Chernousko 1994). ASs are used in
continuous optimization similar to robust programs in discrete optimization (Mulvey et al. 1995). ASs are
applied to those problems, where it is necessary to find a feasible solution over a restricted uncertainty
area for some uncertain parameters. The solution within an AS is guaranteed to be feasible and robust
over this uncertainty region. However, ASs can be applied to schedule robustness analysis if the sched-
ules are formulated in optimal program control (OPC) form. This will cause some differences to robust
optimization which we will consider in this paper. To the best of our knowledge, an attainable set-based
approach to schedule robustness analysis has never been documented in the literature so far.
The rest of this paper is organized as follows. In Section 2, a literature review is presented. In Section 3,
the methodology of AS with an application to scheduling is considered. Section 4 presents a technical
discussion on computation of AS. Section 5 is devoted to the application of AS to a schedule robustness
analysis. In Section 7, the application of AS to schedule robustness analysis is presented and exemplified
by a numerical example. Section 8 is devoted to the broader discussion of the schedule robustness analy-
sis with the help of AS. The paper is concluded by summarizing the main findings of the conducted re-
search.

2 State of the art in schedule robustness

In the literature, the studies on schedule robustness aim at closing the gap between theory and practice
regarding the uncertain nature of real environments for the schedule execution. Different approaches have
been proposed so far since robustness is really a multi-faceted issue in operational research (Roy 2010).
Although robustness is closely interconnected with uncertainty, we avoid in our literature review such
research streams like scheduling with unreliable machines (Leung and Pinedo 2004, Kasap 2006, Epstein
et al. 2012) and stochastic scheduling (Forst 1995, Chauhan et al. 2009, Cai et al. 2009), and concentrate
on robustness of schedules and its measures (indicators).

First, the literature suggests different approaches to generate robust schedules. The studies by Wu et al.
(1999) and Artigues et al. (2005) presented decomposition and grouping approaches to robust scheduling.
Wu et al. (1999) propose a graph-theoretic decomposition for the job shop scheduling problem to achieve
schedule robustness. The expected weighted tardiness is used as the robustness measure. Artigues et al.
(2005) propose to generate a family of schedules instead of a unique one to maintain schedule robustness
in a job shop environment.

Second, the studies on schedule robustness differ regarding the perturbation data. They may be expressed
in discrete, stochastic and interval form (i.e., in many practical situations, only a set of possible values is
known). In the stochastic domain (Leon et al. 1994), probability density functions of the model parame-
ters (e.g., processing time or activity durations, Al-Fawzani and Haouari 2005) or perturbations are used in
order to construct a robust schedule. For example, in the study by Al-Fawzan and Haouari (2005), schedule robustness is defined as the sum of the free slacks of the activities.

The interval data domain has been addressed first in the study by Daniels and Kouvelis (1995) subject to processing time variability in a single machine environment with the objective of minimizing total flow time. The authors propose a scenario-based representation and analysis of uncertainty in spite of using stochastic models. The aim is to find a schedule whose performance degradation in its worst-case scenario is the least among all feasible schedules (i.e., the minimax regret). Other examples can be found in the studies by Yang and Yu (2002) and Kasperski (2005).

Third, the multi-objective issue is addressed. In practice, scheduling quality is multi-dimensional (Kempf et al. 2000). Recent research addresses the issue that in many practical scheduling problems, the concerns of the decision-maker may not be all known in advance and therefore, may not be included in the initial problem definition as an objective function and/or as constraints. Gören and Pierreval (2013) show that in such a case, the usual techniques of multi-objective optimization may become inapplicable and demonstrate a new approach related to the ability of the schedules to absorb the negative effects due to random machine breakdowns.

Fourth, previous studies indicate that the quantitative approaches that use measures for robustness may sometimes be restrictive to compare schedules properly. The graphical representations of robustness may facilitate the understanding of potential behaviour of schedules, if a perturbation occurs (Ghezail et al. 2010). This approach is in line with the operational flexibility approach by Swaney and Grossman (1985) and operational envelopes approach developed by Samsatli et al. (2001).

Even though previous studies included a robustness objective into scheduling, they rarely considered multi-stage cases, alternative parallel machines, and flow shops. Robustness in continuous time domain has not been explored in the scheduling settings so far although it has been extensively investigated in system dynamics and control theory (Mayne et al. 2000, Ivanov and Sokolov 2013). In this study, we suggest to extend this schedule robustness analysis to the scheduling problems for multi-stage flow shops with alternative parallel machines working in continuous time. These problems will be formulated in the state-space domain with the help of the AS approach.

3 Methodology of attainable sets with an application to scheduling

In this section, first, we describe briefly a scheduling model for the continuous flow – flexible flow shop schedule (CF-FFS) problem. Second, the construction of AS for the robustness analysis of schedules described as optimal program control (OPC) is presented. We use the following list of notations:

Sets

\[ \mathbb{B} = \{ \bar{B}^{(i)}, i \in \overline{N}, \overline{N} = (1, \ldots, \overline{n}) \} \]  is the set of jobs

\[ D = \{ D^j, \mu \in \overline{S}, \overline{S} = (1, \ldots, \overline{s}) \} \]  is the set of operations

\[ M = \{ M^j, j \in N, N = (1, \ldots, n) \} \]  is the set of machines

\[ \Gamma \]  is the set of precedence operations

\[ Q(x(t)) \]  is the domain of feasible control inputs
\(Q(x(t))\) is the extended domain of feasible control inputs
\(\tilde{K}\) is the initial class of feasible control inputs
\(\tilde{K}\) is the extended class of feasible control inputs
\(K\) is a set of performance indicators \(J\)
\(D_x\) is an AS in the state space
\(D_f\) is an AS in the performance indicator space
\(\tilde{D}_f^{(2)}\) is an approximated AS under the disturbances
\(U\) is a set of feasible control inputs

**Indices**
- \(i\) is the job index
- \(j\) is the machine index
- \(\mu\) is the operation index (i.e., number of the operation in the job)
- \(l\) is the number of feasible schedules
- \(o\) is the index of parameters and variables in the model \(M_o\)
- \(r\) is the number of the iteration of the algorithm
- \(k\) is the number of execution scenarios (i.e., different perturbations impacts in regard to capacity decrease at different points of time)

**Parameters**
- \(a\) is the planned processing volume defined at the master planning level
- \(\tilde{R}\) is the total machine capacity
- \(T_0\) is the start instant of time of the scheduling horizon
- \(T_f\) is the end instant of time of the scheduling horizon
- \(c\) is the processing intensity
- \(\xi(t)\) is the vector of perturbation impacts
- \(\Delta t\) is the step length of integration for the main and the conjugate system.
- \(h_0^{(o)}, h_1^{(o)}\) are known differentiable functions that determine the end conditions of the vector
- \(t\) is the current time instant
- \(\sigma\) is the duration of the planning interval
- \(\varepsilon(t)\) is the given preset matrix time function of time-spatial constraints
- \(\alpha(\tau)\) is the penalty function in the mathematical model of the operation control processes
- \(q^{(1)}\) and \(q^{(2)}\) are vector-functions, defining the main spatio-temporal, economic, technical and technological conditions for the machine functioning process.
- \(\lambda\) is the vector of the weight coefficients of the performance indicators

**Decision variables**
- \(x^{(o)}\) is a variable characterizing the state of an operation, where \(^{(o)}\) indicates the relationship of the state variable \(x\) to the operation states.
- \(u_{ij}^{(o)}(t)\) is a control variable; we have \(u_{ij}^{(o)}(t) = 1\) at the time point \(t\), if the operation \(D_{ij}^{(3)}\) is assigned to the machine \(M^{(j)}\), and \(u_{ij}^{(o)}(t) = 0\) otherwise
- \(\tilde{u}(t)\) is the decision control action at the moment \(t\) in the extended class \(\tilde{K}\)
\( x(t) \) is a state variable characterizing the processed flow volume

\( u(t) \) is a feasible schedule

\( u^*(t) \) is an optimal schedule

### 3.1. Scheduling model for the CF-FFS problem

CF-FFS problem is a special case of the CF-FJS (continuous flow flexible job shop) scheduling problems (Bozek and Wysocki 2015). CF-FFS scheduling problem is characterized by the following aspects:

- There is a set of jobs and each job is defined by the same given sequence of operations
- For each operation, a set of alternative available machines is defined
- Processing speed of each machine is described as a function of time and is modelled by material flow functions (integrals of processing speed functions) and resulting processing time is, in general, position dependent
- Lot-sizes and release dates are known
- Temporary capacity unavailability is included both in planned and perturbed modes
- Capacity degradation is considered
- Impact of disturbances on performance is included in the analysis with the help of attainable sets

Consider the following concepts:

- “Job” comprises a set of operations needed to complete a customer order according to some criteria such as delivery date and ordered quantity
- „Operation“ is an action needed to complete a production process. Each operation is characterized by some parameters such as lead-time, processed quantity, resource consumption and material flow
- “Flow” is material flow characterizing by real and planned quantities, processing/transportation intensities and speed of flow volume change

The following performance indicators are considered:

- Throughput
- Total lateness

These criteria are typical for schedule optimization in continuous flow systems where the maximization of the total processing volume subject to specified delivery dates is crucial.

Consider a multi-stage flexible flow shop. At each \( l \)-stage, there are some uniform alternative machines \( M^{(j)} \) \((j = 1, \ldots, n)\). Consider the operations \( D^{(i)}_{\mu} \) \((\mu = 1, \ldots, s_i)\), each of which belongs to a job \( B^{(i)} \) from the set \( B = \{B^{(i)} : i \in \overline{N}, N = \{1, \ldots, n\}\} \). All jobs are assumed to be available for processing at time 0. Each machine \( M^{(j,i)} \) is capable of producing all the operations at the \( l \)-stage, but it can handle only one job at a time. Note that for simplification, the stage index \( l \) is omitted in the further progress of this paper, and it is assumed to be considered in the machine indexes \( j \) subject to the non-stationary machine availability at each stage (i.e., “availability windows”), which is expressed in the preset matrix time function
At each of the stages, each machine $M^{(j)}$ has a speed (i.e., an effective processing intensity) $c_{ij}$ that is subject to the total machine capacity $\tilde{R}_j$. The impact of the processing intensity $c_{ij}(t)$ is that the machine $M^{(j)}$ can process $a_{ij}$ units subject to the planned processing volume $a_{ij}$ and $c_{ij}(t)$. An operation $D^{(i)}_{\mu}$ may start only after the previous operation $D^{(i)}_{\mu-1}$ has been completed. All jobs have to be completed by time $T_f$.

The problem consists of scheduling the operations taking into account flow dynamics control, and two objectives are considered: $J_1$ – throughput maximization of the volume of the fully completed jobs (subject to $a_{ij}$ and $c_{ij}$, i.e., in an ideal case $a_{ij} = c_{ij}$ for all jobs subject to $c_{ij}(t)$ and $c_{ij}(t)$; $J_2$ – minimization of total lateness (subject to $T_f$) (a strong requirement on the full completion of all jobs by time $T_f$ may also be included).

Let us consider the mathematical model of processing the operation $D^{(i)}_{\mu}$ in the job $B^{(i)}$. The following notations can be introduced:

- $x_{ij}^{(o)}$ is a variable characterizing the state of the operation $D^{(i)}_{\mu}$, where $^{(o)}$ indicates the relationship of the state variable $x$ to the operation states;
- $\varepsilon_{ij}(t)$ is a given preset matrix time function of time-space constraints;
- $u_{ij}(t)$ is a decision control action for the assignment of an operation to a machine at the moment $t$.

The dynamics of the operation $D^{(i)}_{\mu}$ can be expressed as follows:

$$\frac{dx_{ij}^{(o)}}{dt} = \dot{x}_{ij}^{(o)} = \sum_{j=1}^{n} \varepsilon_{ij}(t)u_{ij}(t). \tag{1}$$

Eq. (1) represents the operation execution dynamics in which the non-stationarity of the execution of the operations is reflected (e.g., a constraint on the breaks in a production shift) (see Figs 1-2). We have $\varepsilon_{ij}(t) = 1$, if machine $M^{(j)}$ is available, and $\varepsilon_{ij}(t) = 0$, otherwise. $u_{ij}(t)$ is a decision variable. We have $u_{ij}(t) = 1$ at the time point $t$, if the operation $D^{(i)}_{\mu}$ is assigned to the machine $M^{(j)}$, and $u_{ij}(t) = 0$ otherwise.
Fig. 1. Dynamics of job execution

\[ T_0 = 1 \text{ day} \]
\[ T_f = 12.75 \text{ days} \]

\[ T_0 = 1 \text{ day} \]
\[ T_f = 12.75 \text{ days} \]

Fig. 2. Dynamics of operation execution

\[ \dot{x}^{(o,1)}_{ij} = \sum_{j=1}^{m} y^{(o,1)}_{ij} \] means that at each point of time where if \( u^{(o,1)}_{ij}(t) = 1 \), the operation is being processed and its processed quantity increases. \[ \dot{x}^{(o,2)}_{ij} = \sum_{j=1}^{m} \varepsilon^{(o,2)}_{ij} u^{(o,2)}_{ij} \] means that the processing is possible subject to feasible capacity time windows and \[ \dot{x}^{(o,3)}_{ij} = u^{(o,3)}_{ij} \] means that if the job is completed, the time to the end of the planning horizon elapses (i.e., earliness of the job completion subject to slack time).
In the example given in Fig. 1, it can be observed that for a planned batch of 29 units, the flow time of the job is 3 days (starting at t=1 and ending at t=4), slack time is 8.75 days (subject to the due date 12.75 days). Fig. 2 depicts execution dynamics of two subsequent operations at the μ-machine subject to three capacity availability time windows (εμ=1). From Fig. 2, processing times, idle times, completion quantities and times can be observed.

The control actions are *constrained* as follows:

\[
\sum_{i=1}^{s_i} \sum_{\mu=1}^{n} u^{(\mu)}_{ij}(t) \leq 1, \quad j = 1, \ldots, n; \quad \sum_{i=1}^{s_i} u^{(\mu)}_{ij}(t) \leq 1, \quad i = 1, \ldots, n, \mu = 1, \ldots, s_i, \quad \text{(2)}
\]

\[
\sum_{j=1}^{n} u^{(0)}_{ij} \left[ \sum_{a \in \Gamma_{i\mu}} (a^{(\mu)} - x^{(\mu)}_{ja}) + \prod_{\beta \in \Gamma_{j\mu}} (a^{(\mu)}_{\beta} - x^{(\mu)}_{\beta}) \right] = 0, \quad \text{(3)}
\]

\[
0 \leq u^{(0)}_{ij}(t) \leq c^{(f)}_{ij} \cdot u^{(0)}_{ij}, \quad \text{(4)}
\]

\[
\sum_{i=1}^{s_i} \sum_{\mu=1}^{n} u^{(f)}_{ij}(t) \leq \tilde{R}^{(f)}_{j}(t) \cdot \sigma^{(f)}(t), \quad \text{(5)}
\]

where conditions (3) - (5) hold for \( i = 1, \ldots, n, \mu = 1, \ldots, s_i, j = 1, \ldots, n; \Gamma_{i\mu}, \Gamma_{j\mu} \) are the sets of operations which immediately precede the operation \( D^{(i)}_{\mu} \), \( \sum_{j=1}^{n} u^{(0)}_{ij} \sum_{a \in \Gamma_{i\mu}} (a^{(\mu)} - x^{(\mu)}_{ja}(t)) = 0 \) is an “and” constraint, which denotes the condition of the total processing of *all* the predecessor operations, \( \sum_{j=1}^{n} u^{(0)}_{ij} \prod_{\beta \in \Gamma_{j\mu}} (a^{(\mu)}_{\beta} - x^{(\mu)}_{\beta}) = 0 \) is an “or” constraint, which denotes the condition of the processing of *at least one* of the predecessor operations.

Constraints (2) define the 1x1 assignment problem. Constraints (3) bring the natural time logic into the model and determine the precedence relations by blocking operation \( D^{(i)}_{\mu} \) until the previous operations \( D^{(i)}_{\alpha}, D^{(i)}_{\beta} \) have been completed. Inequalities (4) use the assignment decisions \( u^{(0)}_{ij}(t) \) from the model \( M_a \) and the processing speed \( c^{(f)}_{ij}(t) \) of the machines \( M_{(j)} \). These constraints can be classified into two groups: \( q^{(1)}(x, u) = 0 \) (constraint (3)) and \( q^{(2)}(x, u) \leq 0 \) (constraints (2), (4), and (5)).

In order to assess the results of operation execution, we define the following start and end conditions:

\[
h^{(o)}_0(x^{(o)}(T_0)) \leq 0; \quad h^{(o)}_1(x^{(o)}(T_f)) \leq 0, \quad \text{(6)}
\]

where \( h^{(o)}_0, h^{(o)}_1 \) are known differentiable functions that determine the start and end conditions of the vector

\[
x^{(o)} = (x^{(o)}_{i1}, \ldots, x^{(o)}_{i\mu})^T. \quad \text{(7)}
\]

The initial and end conditions (8) and (9) specify the values of the variables at the beginning and end of the planning period, namely:
at the moment $t = T_0: x_{\mu}^{(o)}(T_0) = 0$.  
\hspace{1cm} (8)

at the moment $t = T_f: x_{\mu}^{(o)}(T_f) = d_{\mu}^{(o)}$.  
\hspace{1cm} (9)

The OPC $\mathbf{u}(t)$ and the state trajectory $\mathbf{x} = f(\mathbf{x}, \mathbf{u}, t)$ should be determined so that the constraints (8) and (9) are met; in other words, the desired values of the performance indicators should be achieved as an analogy to goal programming.

Constraint (8) reflects that, at the beginning, the volume of the executed operations is equal to zero (in the case that a certain volume of the orders is to be transferred from the previous planning period to the beginning of the current planning period, this should be reflected in (8)). Condition (9) reflects the desired end state, i.e., the completion of the operations by the time $T_f$.

According to the problem statement, let us introduce the following performance indicators (objectives):

$$J_1^{(o)} = \frac{1}{2} \sum_{i=1}^{n} \sum_{\mu=1}^{m} (a_{\mu}^{(o)} - x_{\mu}^{(o)}(T_f))^2,$$

$$J_2^{(o)} = \sum_{i=1}^{n} \sum_{\mu=1}^{m} \int_{T_0}^{T_f} a_{\mu}^{(o)}(\tau) u_{i\mu}^{(o)}(\tau) d\tau.$$  
\hspace{1cm} (10)

\hspace{1cm} (11)

The performance indicator $J_1^{(o)}$ (function (10)) characterizes the accuracy of the accomplishment of the end conditions, i.e., the throughput by the time $T_f$. This can also express the extent of losses caused by a non-fulfillment of the end conditions. The objective function (11) minimizes total maximum lateness using penalties. The function $a_{\mu}^{(o)}(\tau)$ is assumed to be known for each operation.

Now the scheduling problem can be formulated as the following problem of dynamic system control, which is used to find a feasible control $\mathbf{u}(t), \quad t \in [T_0, T_f]$: meeting the constraints $\mathbf{q}^{(1)}(x, u) = 0, \mathbf{q}^{(2)}(x, u) \leq 0$ (2)–(5), the model (1) guides the integrated dynamic system (i.e., a flexible flow shop) $\mathbf{x} = f(\mathbf{x}, \mathbf{u}, t)$ from the initial state $\mathbf{h}_0$ to the specified final state $\mathbf{h}_1$. If there are several feasible OPC (schedules), then the best (optimal) one should be selected in order to minimize a general performance indicator $J_G$ taking into account the priority coefficients $\lambda_k, k = \overline{1, K}$ of the normalized performance indicators $J = \{J_1, ..., J_K\}$ (e.g., $J_1$ is the total number of delayed jobs and $J_2$ is the total number of on-time completed jobs):

$$J_G = \sum_{k=1}^{K} \lambda_k J_k; \quad \lambda_k > 0; \quad \sum_{k=1}^{K} \lambda_k = 1$$

\hspace{1cm} (12)

The formulated scheduling model satisfies the conditions of the existence theorem in Lee and Markus (1967, Theorem 4, Corollary 2), which allows us to assert the existence of an optimal solution in the appropriate class of admissible controls. In the study by Ivanov et al. (2015), the existence of an OPC as well as optimality and sufficiency conditions have been proved, and an algorithm for the calculation of an OPC has been presented and analyzed. The optimality properties have been proved theoretically and ver-
fied experimentally. Since the same model and algorithm can be applied to the model presented here, we avoid their representation in this paper and concentrate only on the robustness assessment.

3.2. Construction of an AS for a robustness analysis

Consider a decision maker who seeks to find a robust schedule in a flexible flow-shop with alternative parallel machines subject to some performance indicators (e.g., tardiness and throughput) and different uncertainties with regard to processing capacity disruptions. After having calculated an optimal schedule (or a number of feasible schedules), a schedule as an OPC may be calculated for different perturbations (e.g., machine capacity fluctuations). That is why we need a form to represent the states and performance of schedules in regard to different capacity degradation levels. Having such a form, it becomes possible to develop a method to estimate the schedule robustness as the achievement of the planned performance in the presence of disruptions. We suggest using ASs for these tasks.

Let us introduce the notation for an AS. $D_J(t, T_0, x(T_0), U(x(T_0)))$ is an AS in the state space, $D_J(t, T_0, x(T_0), U(x(T_0)))$ is an AS in the performance indicators’ space, and $D_J^J(t, T_0, x(T_0), U(x(T_0)))$ is an approximated AS under the disturbances at the moment $t_1$.

An AS is a fundamental characteristic of any dynamic system in the interval $(x(T_0), x(T_f))$. The AS approach determines the range of execution policies in the presence of disturbances over which the system can be guaranteed to meet certain goals. The union of these execution policies (i.e., feasible schedules) is called an AS in the state space. The union of possible performance outcomes from the given execution policies is called an AS in the performance space. The AS in the state space depicts the possible states of a schedule subject to variations of the parameters (both planned and perturbation-driven) in the nodes and channels (e.g., different capacities, lot-sizes, etc.). In order to interconnect the schedule execution and the performance analysis to the AS in the state space, an AS in the performance space can be brought into correspondence (see Fig. 3).

Figure 3. Representation of dynamic changes in the schedule performance by attainable sets

In projecting these two ASs on each other, a certain range of the schedule execution policies and the corresponding variation of the performance indicators can be determined. A continuous time representation allows to investigate the changes in the execution at each time point. Therefore, at each time point, an AS can be calculated and related to the output performance. The justification of the choice of the AS method
is related first of all to its dynamic nature (Gubarev et al. 1988, Chernousko 1994, Clarke et al. 1995). An AS may be favourable to obtain estimations of performance attainability and to consider perturbations and attainability abilities of the schedules as time functions. The perturbation functions may be set up for different data, including interval perturbations [0;1].

Consider the model (1)-(9) under the disturbances $\xi(t)$

$$M_\xi = \left\{ u(t) \mid \dot{x} = f(x,u,t), \right. $$

$$h_0(x(T_0)) \leq 0, h_i(x(T_f)) \leq 0, $$

$$q_0(x,u,\xi) = 0, q_2(x,u,\xi) \leq 0 \right\} \quad (13)$$

Assume that a certain initial state of $x(T_0)$ is known and that a schedule $u^*(t)$ has been calculated. Then, an AS can be brought into correspondence with the vectors $x(T_0)$, $u(t)$, and $\xi(t)$. This AS represents the set of all possible execution scenarios which may occur in the schedule execution after the perturbations. We propose this area to be named as the AS in the state space under disturbances defined as follows:

$$D_x^{(2)}(T_f, T_0, X_0, \Xi, u_f) \quad (14)$$

Let the admissible limits of the oscillations be as follows:

$$J_1 \leq J_{sl}, J_2 \leq J_{bl}. \quad (15)$$

They construct a special area $P_J$ in the performance indicator space. To the given schedule $u^*(t)$ under the non-stationary perturbations $0 \leq \xi(t) \leq 1$, an AS can be calculated (see Fig. 4).

Figure 4. An attainable set in the performance space and the area of admissible performance deviation

4. Computation of attainable sets

The AS is calculated from the main OPC vector by varying the perturbation impacts as control variables. Therefore, the AS can be calculated for the perturbed schedules. In varying these perturbations at each instant of time over the schedule within the time interval and setting these variations into the initial differential system, a set of points where the schedule performance can be steered to is generated (i.e., the AS $D_x^{(2)}(T_f, T_0, \Xi, u_f)$).
Assume that at the moment \( t = T_0 \) we have some start conditions \( h_0(x(t)) \in \mathbb{R}^{n_0} \) and it becomes possible to find both OPC vector \( u^*(t) \) and state vector \( x^* \) at \( t = T_0 \). Assuming that partial objective functions can be converted to a general performance indicator \( J_G \), we have

\[
J_G = \lambda_1 \varphi_G(x(T_f)) + \lambda_2 \int_{T_0}^{T_f} f_G^{(1)}(x(\tau))d\tau + \lambda_3 \int_{T_0}^{T_f} f_G^{(2)}(u(\tau))d\tau ,
\]

where \( \lambda_1 > 0, \lambda_2 > 0, \lambda_3 > 0, \sum_{i=1}^{3} \lambda_i = 1 \) are given weight coefficients, \( \varphi_G, f_G^{(1)}, f_G^{(2)} \) are given functions at \( X \subseteq \mathbb{R}^{n_0}, \mathbb{R}^{d^G} \times T, U \subseteq \mathbb{R}^{nU} \) \( (T = \mathbb{R}^+ \) is the set of time moments, \( \mathbb{R}^+ \) is a set of real numbers, \( X, U \) are sets for \( x, u \). Assume that \( \varphi_G(x) \) has no interruptions for \( X \), \( f_G^{(1)}(x(\tau), \tau) \) along with its derivative for \( x \) at each \( \tau \in (T_0, T_f) \) has no interruptions for \( x \) and is piecewise-continuous at \( x \in X \) for \( \tau \); \( f_G^{(2)} \) for each \( \tau \) is convex regarding \( u \) and for each \( u \in U \) it is bounded and piecewise-continuous for \( t \). In the model \( M \) we consider along with the feasible control class \( \tilde{K} \) and extended feasible control class \( \tilde{K}^* \), where the relay condition

\[
u_{ig}^{(o)}(t) \in [0,1]
\]

is replaced by a less strict one

\[
u_{ig}^{(o)}(t) \in [0,1]
\]

In this case, an extended domain \( \tilde{Q}(x(t)) \) of feasible control inputs may be formed by means of special transformations ensuring the convexity and the compactness of \( Q(x(t)) \) (Moiseev, 1974). The theorem of Lee and Markus (1967) confirms that all the conditions for the existence of an optimal control for the extended control class \( \tilde{K} \) are valid. If in a given class of feasible control actions \( \tilde{K} \), an optimal control \( \tilde{u}(t) \) exists, then, as arises from the local cut method, the control \( \tilde{u}(t) \) returns at each time instant \( t \in (T_0, T_f) \) at the set \( \tilde{Q}(x(t)) \) the maximum to the Hamiltonian function (Ivanov et al. 2015).

For a simplification, it is assumed that the transition from the vector form \( J \) to a scalar form \( J_G \) has been performed on the basis of the weight coefficients. Now the scheduling problem can be re-formulated as the following problem of dynamic system control. The task is to find a feasible control \( u(t), [T_0, T_f] \) which ensures that the dynamic control model meets the constraint functions and guides the dynamic system (i.e., the schedule) \( \dot{x} = f(t, x, u) \) from the initial state to the specified final state subject to given end conditions and the uncertainty area under the disturbances \( \xi(t) \). If there are several feasible controls (schedules), then the best one (optimal) should be selected in order to maximize (minimize) the components of \( J_G \).
Introduction of disturbance (perturbation) functions in constraints of the model (1)-(9) allows computing
AS of different feasible schedules, i.e., all possible results (i.e., throughout and tardiness) of the schedule
execution subject to different variations of the parameters (e.g., the capacity disruptions). In other words,
the introduction of AS allows to analyse the performance variations in feasible schedule execution policies
under conditions of non-stationary perturbations. If so, an AS can be used to analyze the schedule
robustness, i.e., an ability to achieve the specified schedule objectives in the presence of perturbations.

Assume that a certain initial state of \( x(T_0) \) is known and that a schedule \( u^*(t) \) has been calculated. Then,
an AS can be brought into correspondence with the vectors \( x(T_0), u(t), \) and \( \xi(t). \) \( \xi(t) \) is a perturbation
vector at the moment \( t, \) \( \Xi \) is a set of allowable perturbations \( \xi_1(t) \leq \xi(t) \leq \xi_2(t). \) \( \xi_1(t), \xi_2(t) \) are pre-
scribed vector functions, which define minimum and maximum values of perturbation effects on the realiza-
tion stage for each fixed schedule.

Using AS, it becomes possible to formulate a dual problem

\[
J_G^*(x(\cdot)) \to \min_{x(\cdot) \in D_i(T_f, T_0, x(T_0))},
\]

where \( D_i(T_f, T_0, x(T_0)) \) is AS of \( M_i \) (\( i = 1,2,3 \)); \( J_G^*(x(\cdot)) \) the terminal functional subject to Eq. (12).

One possible method for construction \( D(T_f, T_0, x(T_0)) \) is multiple solution of the problem (20)

\[
J_G^*(x(\cdot)) = c^T x(T_f) \to \min_{x \in \Omega(x)},
\]

The boundary points of the set \( D_i(T_f, T_0, x(T_0), U(x(T_0))) \) are obtained as the solutions to Eq. (20),
where \( c \) is a intended vector such that \( \|c\| = 1. \) For each vector \( c \) we obtain the OPC \( u^*(t), \) state
\( x^*(T_f) \) as one boundary point of \( D_i(t, T_0, x(T_0), U(x(T_0))) \), and the hyperplane \( c^T x^*(T_f) \). Let \( \Delta_x \) be
the number of different vectors \( c_\alpha, \alpha = 1,...,\Delta_x, \) then the external approximation
\( \overline{D}_i^+(T_f, T_0, x(T_0), U(x(T_0))) \) of the set \( D_i(T_f, T_0, x(T_0), U(x(T_0))) \) is a polyhedron whose faces lie on
the corresponding hyperplanes, and the internal approximation \( \overline{D}_i^-(T_f, T_0, x(T_0), U(x(T_0))) \) is a polyhe-
dron whose vertices are the points

\[
x^*_\alpha(T_f), \alpha = 1,...,\Delta_x, \text{ i.e. } D^-(T_f, T_0, x(T_0)) = C_0(x_1(T_f),...,x_{\Delta_x}(T_f)).
\]

In general, an AS can be non-convex. In this study, an AS is calculated upon the scheduling model (1)-
(10).

**Corollary 1.** The scheduling model (1)-(10) is a linear non-stationary finite-dimensional controlled differen-
tial system with a convex area of admissible control. According to Lee and Markus (1967, Theorem 4,
Corollary 2), \( U(x(T_0)) \) is convex, and the AS is bounded, closed and convex.

If, for example, the desired performance is bounded by the values \( J_{a1} \) and \( J_{b1}, \) the AS from Fig. 3 shows
the attainability of the desired performance for all considered execution scenarios and perturbations. All
the points in the AS (i.e., the performance value subject to different perturbations) lie outside the rectan-
gle $P_j$ created by the limits on the minimal desired performance. However, at the same time, it may be observed that in many cases, the performance may significantly exceed the desired values. This may be an indication for too high robustness reserves. In this case, it has to be checked whether some unnecessary structural elements are available which create costs but do not add a value. Another option could be to increase the desired performance to the points $J_{1l}$ and $J_{2l}$. However, in this case, the risks increase. There is a certain portion of possible outcomes where the planned performance may not be achieved (i.e., the area of the intersection of $P_j$ and the $D_j^\xi$).

As the dimensionality of the AS may be high, intervals, boxes, and ellipsoids can be used to define AS for discrete cases. We suggest a rectangular approximation of the AS based on four points (a schedule with the best value of $J_1$, a schedule with the best value of $J_2$, a schedule with the worst value of $J_1$, and a schedule with the worst value of $J_2$). Then for an AS approximation, the task is to solve problem (16):

$$\min \{J_1^{\xi}, J_2^{\xi}(t) \} \in \Xi$$

$$\min \{J_3^{\xi}, J_4^{\xi}(t) \} \in \Xi$$

In (16), the components of the objective function are presented subject to Eq. (10) and (11) and model (1)-(11). In problem (16) we have four sub-problems each with a single criterion. The meaning of problem (16) is that from the set of feasible schedules for different data scenarios, four points are selected which correspond to the best and worst values for $J_1$ and $J_2$, respectively. The result of solving this problem are the coordinates of the points $J_1^\star, J_2^\star, J_3^\star, J_4^\star$, which makes it possible to construct an AS approximation (see Figure 5).

![Figure 5. Approximated attainable set](image-url)

Another possibility for the approximation is simpler where the points would correspond to the best case for all $J$, the worst case for all $J$, and the best cases for only one $J$. The meaning of such an AS approximation is that not all possible scenarios but only some of them are considered regarding the best case, the worst case, and some intermediate scenarios. Such an approximation, e.g., as a rectangular one, makes it possible to represent the AS $D_j^\xi$ also as a rectangular one similar to $P_j$. 
5. Computation of schedule robustness

Consider a set of performance indicator variations defined as \( D_j^{(s)}(T_f, T_0, X_0, \Xi, u) \)

**Corollary 2.** If, for an schedule \( u_j(t) \), (\( l = 1, ..., L \)) under perturbations \( \xi_j(t) \), the requirement \( \overline{D}_j^{(s)}(T_f, T_0, X_0, \Xi, u_j) \subset P_j \) is fulfilled, the schedule \( u_j(t) \) is considered to be robust under the perturbations \( \xi_j(t) \). In other words, the admissible deviations of \( J_1 \) and \( J_2 \) are considered to be acceptable.

The points within the area of the intersection of the two rectangles \( D_j^{(s)} \) and \( P_j \) include the possible outcomes for a non-achievement of the planned performance in the presence of perturbations. This area explicitly reflects the robustness and can be easily quantified due to a rectangle resulting from four-point approximation. A robustness assessment of the schedules comes down to the calculation and analysis of a robustness index for different uncertainty scenarios. The most robust schedule is selected according to condition (21)

\[
S_k(u_j(t)) = \min_{t \in S_k} \max_{l \in S_k} S_k(u_j(t)) \tag{21}
\]

where \( S_k(u_j(t)) \) is the intersection area of the sets \( \overline{D}_j^{(s)}(T_f, T_0, X_0, \Xi, u_j) \) and \( P_j \).

The condition (21) is based on the estimation of four points of the AS corresponding to the point of the best performance for both the performance indicators, the two points for the best performance for one of the performance indicators and the worst-case point. These points result from the rectangular approximation (compared with problem (16)).

The larger the intersection area of the two rectangles, the less robust is the schedule. Therefore, in our approach, to each of the alternative schedules, performance indicators with different values (as declined by perturbations) and the corresponding robustness for different capacity disruptions can be calculated. In the ideal case (i.e., no disruptions), there should be no intersection and the robustness index is equal to zero, i.e., the considered perturbations and resulting variations of the control variables do not affect the performance. It is now the task of the decision-maker to take the final decision on the schedule selection based on the performance and robustness estimations subject to his/her own risk perceptions.

If none of the generated schedules provides a satisfactory level of performance and robustness, the parameters of the master plans (e.g., resource capacities, inventories, lot-sizes, delivery data, etc.) can be tuned. Besides, such an analysis can reveal that a very cost-intensive master plan attains the same schedule robustness as a more cost-efficient master plan. Based on the AS-supported robustness analysis, further analysis of master plans and schedules can take place and bottlenecks can be revealed. They can be, e.g., a low service level, permanent disruptions in particular flows, deviations from forecasts and a low adaptability to these changes, etc. Then, changes in the parameters of the master plan can be taken to increase the schedule robustness.
6. Algorithmic realisation

Let us consider the algorithmic realization of our approach. At the first stage, the OPC vector $u^*(t)$ and the state trajectory $x^*(t)$ are obtained for the model (1)-(11). The OPC vector at time $t = T_0$ and for the given value of $\psi(t)$ should return the maximum to (10)–(11), which have been transformed to a general performance index and expressed in scalar form $J_G$. At this stage, the schedule feasibility is proved.

At the second stage, model (13) is used. The basic peculiarity of the considered boundary problem is that the initial conditions for the conjunctive variables are not given. At the same time, an OPC should be calculated subject to the end conditions (8)–(9). That is why the initial schedule $u^*(t)$ is used as an initial solution. The AS is now calculated from the main OPC vector by varying the perturbation impacts as control variables. Therefore, the AS can be calculated for the perturbed schedules. In varying these perturbations at each instant of time over the schedule within the time interval and setting these variations into the initial system of differential equations, a set of points where the schedule performance can be steered to is generated (i.e., the AS $D_j^{(r)}(T_f, T_0, X_0, \Xi, u_j)$). At this stage, schedule robustness and feasibility are proved.

We suggest the following algorithm based on maximum principle:

**Step 1** An initial solution $\tilde{u}(t), t \in (T_0, T_f]$ (a feasible control, in other words, a feasible schedule) is selected and $r = 0$.

**Step 2** As a result of the dynamic model run, $x^{(r)}(t)$ is received. Besides, if $t = T_f$, then the record value $J_G = J_G^{(r)}$ can be calculated. Then, the transversality conditions are evaluated.

Note. A methodical challenge in applying the maximum principle is to find the coefficients of the conjunctive system which change in dynamics. In the study (Ivanov et al. 2015) it has been shown that these coefficients can be found analytically. The coefficients of the conjunctive system play the role of the dynamical Lagrange multipliers as compared with discrete optimization dual formulations. Thus the transversality conditions establish the connection between the dual and direct problems.

**Step 3** The conjugate system is integrated subject to $u(t) = \tilde{u}(t)$ and over the interval from $t = T_f$ to $t = T_0$. For the time moment $t = T_0$, the first approximation is received as a result. Here, the iteration with the number $r = 0$ is completed.

**Step 4** From the time point $t = T_0$ onwards, the control is determined ($r = 0, 1, 2, ...$ denotes the number of the iteration). In parallel with the maximization of the Hamiltonian, the main system of equations and the conjugate one are integrated. The maximization involves the solution of several mathematical programming problems at each time point.

The assignments (i.e., the control variables $u^{(s)}_{ijg}$) from constraints (2) and (3) are used in constraints (4) and (5). For the time, constraints (4) and (5) influence constraints (2) and (3) through the transversality
conditions, the conjunctive system, and the Hamiltonian function. The iterative process of the search for an optimal schedule is terminated under the following circumstances: either the allowable solution to the problem is determined during the solution of a relaxed problem, or at the fourth step of the algorithm after the integration we obtain:

\[
\begin{align*}
J^{(r+1)}_G - J^{(r)}_G &< \varepsilon_1, \\
\|u^{(r+1)} - u^{(r)}\| &< \varepsilon_2
\end{align*}
\]  

(18) (19)

where \(\varepsilon_1, \varepsilon_2\) are given small values, \(r = 0, 1, 2, \ldots\). If conditions (18)–(19) are not satisfied, then the third step is repeated, etc.

In the result, the definition of the starting and end time for processing an operation on a machine results automatically from the \(\text{OPC v vector}\) subject to the assignment control variables \(u^{(o)}_{ij}\) (cf. Figs 1 and 2).

Since a number of feasible schedules can be used as \(\bar{u}(t), t \in [T_0, T_f]\), their ASs may differ from each other and the schedule robustness can be compared based on index (19). The usage of continuous perturbation functions allows a non-stationary analysis of the schedule robustness (e.g., the perturbations between 8:00 and 10:00 am, or between 9:00 and 11:00 am may have a different impact on the performance). For this reason, the AS analysis will select a schedule as a more robust if we have assignments and processing at the time intervals without critical perturbations.

7 Example

For numerical tests, a software prototype has been developed with C++ and XML. The performance indicators may be weighted regarding their priority. Schedules are obtained as OPC, and different schedules are analysed on robustness subject to three uncertainty scenarios. For each schedule and scenario, the performance indicators are calculated and represented graphically. The intersection area (i.e., the robustness index) is depicted explicitly.

Let us provide an example. The analysis has been performed subject to \(J_1\) as the throughout, \(J_2\) as the total tardiness, and the robustness (i.e., the ability to attain the planned performance in the presence of perturbations given by intervals 0.5-1.0, 0.7-1.0, and 0.9-1.0). For example, the interval 0.5-1.0 means that the capacity availability will vary during the execution from 50% to 100% as a time function \(\xi(t)\). The admissible values of the performance indicators are set up as \(J_1 > 30\) and \(J_2 > 8\). Different perturbation impacts can be modelled through the Eq. (13).

Note. A number of modelled scenarios in model (13) can be unlimited. Since we are interested not in the exact computation of the AS but rather in using AS for comparing the robustness of different schedules, we consider for AS construction only the performance results of four scenarios subject to the rectangle AS approximation.

Consider three alternative schedules (subject to alternative plans) and three scenarios. The analysis has been performed subject to service level (SL), delivery reliability (DR) and the robustness (i.e., the ability
to attain the planned performance in the presence of perturbations given by intervals 0.5-0.7, 0.7-0.9, and 0.9-1.0). The considered plans differ regarding machine intensity, puffer times, job volumes, reserves of resource capacities and alternative transportation routes.

Running the scheduling algorithm, the values of the performance indicators and the robustness index given in Table 1 have been obtained.

**Table 1 Results of the performance analysis**

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Schedule 1 $u_1(t)$</th>
<th>Schedule 2 $u_2(t)$</th>
<th>Schedule 3 $u_3(t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$J_1$</td>
<td>$J_2$</td>
<td>RIX</td>
</tr>
<tr>
<td>Ideal scenario</td>
<td>42</td>
<td>11</td>
<td>0</td>
</tr>
<tr>
<td>Scenario 1</td>
<td>40</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>Scenario 2</td>
<td>38</td>
<td>7</td>
<td>0</td>
</tr>
<tr>
<td>Scenario 3</td>
<td>27</td>
<td>5</td>
<td>9</td>
</tr>
</tbody>
</table>

The robustness index can be calculated as the area of intersection of two rectangles defined by the AS and the limits on its performance (see Figure 6).

![Figure 6](image)

Figure 6. Different robustness variants of schedules: (a) a more robust case, and (b) a less robust case.

The first (blank) rectangle is constructed on the basis of the above-mentioned admissible values of the performance indicators. After running the scheduling model subject to different execution scenarios, different ASs (i.e., the shaded rectangles) have been generated (two of them are shown in Figure 3; they display the results for scenario 3 and schedules #1 and #3, respectively). Recall that in an AS, a certain range of schedule executions and the corresponding variation of the performance indicators are determined. This means that the more the intersection region of the two rectangles is, the less robust is the schedule.
It is now the task of a decision-maker to take the final decision on the schedule selection based on the performance and robustness estimations subject to his/her own risk perceptions. If the values of the performance indicators lie outside the AS (i.e., those cases having a zero value of the robustness), this means that the desired performance can be \textit{attained} under any disturbances from the considered execution scenario. In the other cases, there are some disturbances which may decrease the performance to an undesirable level. In both cases from Figure 3, the worst-case performance lies within the performance rectangles and creates, therefore, the intersection areas of 9 in the case (a) and 36 in the case (b). If none of the generated schedules provides a satisfactory level of performance and robustness, the parameters of the master plans (e.g., resource capacities, inventories, lot-sizes, delivery data, etc.) can be tuned.

\textit{Remark.} Note that with the help of an AS, the robustness analysis is performed not by simulation but on the basis of an optimization-based model and for intervals of non-stationary perturbation data. In this case study, uncertainty has been previously considered in a subjective way during a “risk analysis” meeting only on the basis of the expert knowledge. With our approach, the sophisticated scientific methodology, i.e., the AS, can be used for a decision-making support.

\section*{8. Application discussion}

The results of this research can be of interest not only for control engineers but also for a large production research community. In essence, the developed method allows estimating robustness of any control policy subject to some changes in system parameters (e.g., capacities, inventory, material supply, etc.). Such problems can be formulated for different systems, including supply chains.

The specialists in scheduling (discrete optimization) can also integrate the developed control method in their approaches. First, end conditions (e.g., Eqs (6)-(9)) in OPC models play the role of demand variables in discrete models. The right parts of Eqs. (6)-(9) are predetermined at the master planning stage subject to the planned demand for each job. Second, the constraints (2)-(5) are identical to demand and capacity constraints in discrete models. However, at each $t$-point of time, the number of variables is determined only by the operations which are currently in the “scheduling window”. Therefore, the tendency for OPC computation will be to have small-size instances and to apply known methods for the solution of discrete models (e.g., Hungarian or Branch & Bound methods). Finally, the coefficients of the conjunctive system play the role of the dynamical Lagrange multipliers as compared with discrete optimization dual formulations. Thus the transversality conditions establish the connection between the dual and direct problems, similar to shadow prices in inverse linear optimization models.

The developed model can be used by scheduling specialists to adjust mitigation and recovery policies with regard to continuous multi-stage scheduling systems. The model helps to identify whether the existing schedule is robust for different disruption scenarios. The schedule robustness index can be used to compare different schedules regarding the trade-off “Performance vs robustness”. If none of the generated schedules provides a satisfactory level of performance and resilience, the parameters of the master plans (e.g., resource capacities, inventories, lot-sizes, delivery data, etc.) can be tuned. Besides, such an analysis
can reveal that a very cost-intensive master plan attains the same schedule robustness as a more cost-efficient master plan.

Based on the AS-supported robustness analysis, further analysis of plans and schedules can take place and bottlenecks can be revealed. They can be, e.g., a low service level, permanent disruptions in particular flows, deviations from forecasts and a low adaptability to these changes, etc. Therefore, an AS can be applied to estimate the possible perturbation impacts on the schedule performance. An AS in the state space depicts the possible states of a schedule subject to variations of the model parameters (e.g., different capacities and processing times (Sotskov et al. 2013)). In order to interconnect the schedule execution and the performance, an AS in the performance space can be brought into correspondence to an AS in the state space. In projecting these two ASs on each other, a certain range of the schedule execution policies and the corresponding variation of the performance indicators can be determined. A continuous time representation allows to investigate the changes in the execution at each time point. Therefore, at each time point, an AS can be calculated and related to the schedule performance.

If, for example, the desired performance is bounded by the values $J_{a1}$ and $J_{b1}$, the AS from Fig. 1 shows the attainability of the desired performance for all considered execution scenarios and perturbations. All the points in the AS (i.e., the performance value subject to different perturbations) lie outside the rectangle $P_J$ created by the limits on the minimal desired performance. However, at the same time, it may be observed that in many cases, the performance may significantly exceed the desired values. This may be an indication for too high robustness reserves. In this case, it has to be checked whether some unnecessary structural elements are available which create costs but do not add a value. Another option could be to increase the desired performance to the points $J_{1l}$ and $J_{2l}$. However, in this case, the risks increase. There is a certain portion of possible outcomes where the planned performance may not be achieved (i.e., the area of the intersection of $P_J$ and the $D_J$).

An AS may be favourable to obtain estimations of performance attainability and to consider perturbations and attainability abilities of the schedules as time functions. The perturbation functions may be set up for different data, including interval perturbations $[0;1]$ with fully unpredictable disruptions. An AS allows multi-objective considerations. Besides, the application of an AS allows a direct interconnection with the adaptation models which are widely described in control terms in order to design rescheduling algorithms in the state-space form with desired closed-loop properties (Ivanov and Sokolov 2012, Subramanian et al. 2013).

Besides, if an AS is known, it becomes possible to analyze the dependence between the scheduling results subject to the schedule performance and the start and end states. In other words, it becomes possible to define the area in which permissible solutions (e.g., schedules) are included. On the other hand, an AS analysis may show that, with the given resources and at the given time horizon, it is impossible to achieve the required output performance; hence, additional resources should be introduced or the supply cycle shall be expanded (here, the AS approach is similar to goal programming). Limitations of using AS are
related to their dimensionality. However, in most cases, it is possible to approximate AS, e.g., to a rectangular form while estimating the outcomes at four points of an AS.

9. Conclusions
The issue of an integrated analysis of schedule execution dynamics and the achievement of the planned performance in a real uncertain and perturbed environment has been faced in this paper from the position of proactive scheduling. This study is the first one that addresses this issue with the help of ASs. It is shown how ASs can be used to obtain estimations of performance attainability and consider perturbations in continuous time and as interval functions.

In this study, we developed a new method for a robustness analysis of those schedules that are formulated in continuous time in the state-space domain. The technical development and the approximation of attainable sets have been presented. The proposed robustness index can be used as an additional indicator for decision-makers regarding the trade-off “efficiency vs. robustness”. In particular, this metric can be used for ranging alternative schedules regarding different individual risk perceptions on the basis of the minimax regret approach.

With the presented results, it becomes possible to obtain ASs for interval data with no a priori information about perturbation impacts, i.e., for non-stationary perturbations. We conclude that the ASs may be favourable to obtain estimations of the performance attainability and to consider perturbations and attainability abilities of the schedules as time functions. Perturbation functions may be set up for different data, including interval perturbations with fully unpredictable disruptions. An AS allows multi-objective considerations. Besides, the application of an AS allows a direct interconnection with the adaptation models which are widely described in control terms in order to design rescheduling algorithms in the state-space form with desired closed-loop properties.

Limitations of using an AS are related to their dimensionality. However, in most cases, it is possible to approximate an AS, e.g., to a rectangular form while estimating outcomes at four points of the AS. This is enough for ranking different alternative schedules regarding robustness. For a more detailed analysis, other forms of approximations can be considered, also for non-convex cases. An analysis of “positive” cases, i.e., non-intersection cases is also required, e.g., if the two rectangles are located too far from each other, it could point out too high excessiveness of the capacities and potentials for the master plan optimization. In addition, a more detailed analysis of the intersection region can reveal some observations on the individual performance indicators and the robustness. Finally, an investigation can be performed subject to different problems for single and multiple machines in job and flow shop scheduling.

References


