CHERD-2458; No. of Pages 15
ARTICLE IN PRESS
chemical engineering research and design xxx (2016) xxx–xxx

Contents lists available at ScienceDirect

Chemical Engineering Research and Design

journal homepage: www.elsevier.com/locate/cherd

From rescheduling to online scheduling

Dhruv Gupta[a], *Christos T. Maravelias*[a,*], *John M. Wassick*[b]

[a] *Department of Chemical and Biological Engineering, University of Wisconsin-Madison, 1415 Engineering Drive, Madison, WI 53706, USA*
[b] *The Dow Chemical Company, Midland, MI 48764, USA*

ARTICLE INFO

Article history:
Received 4 August 2016
Received in revised form 18 October 2016
Accepted 19 October 2016
Available online xxx

Keywords:
Chemical production scheduling
Process uncertainty and disturbances
Re-optimization

ABSTRACT

We first review advances in rescheduling, traditionally viewed as an approach to tackle uncertainty, including methods that rely on recourse through feedback as well as methods that account for uncertainty a priori. Then, we show that traditional event-triggered rescheduling has some shortcomings which can be addressed if rescheduling is approached as an online problem. We review methods that consider aspects of this online problem and define notation and some key features of this problem. Furthermore, we propose a broad framework for the classification of online scheduling methods. Finally, we discuss a number of open research questions, including the generation of high quality of closed-loop (implemented) schedules through the selection of appropriate model, horizon length, time-step, objective function modifications, and constraint addition.

1. Introduction

Scheduling of chemical processes is an important problem that arises in many manufacturing sectors (Harjunkoski et al., 2014). Much work has been accomplished towards realistic modeling and effective solution methods for schedule generation (Méndez et al., 2006; Velez and Maravelias, 2014; Velez et al., 2015). Yet another aspect of scheduling, however, is rescheduling. A schedule can become suboptimal or even infeasible due to, for example, arrival of a rush order or a change in resource availability. The revision of an existing schedule in response to changes or disruptions is termed as reactive scheduling or rescheduling (Vieira et al., 2003; Li and Ierapetritou, 2008a; Ouelhadj et al., 2009).

The goal of this paper is twofold. The first goal is to provide a review on advances in rescheduling, clarify terminology, and examine limitations in the scope of the work till date. Importantly, the second goal is, as a way forward, to show how rescheduling is a special case of what in this paper we term "online scheduling", and emphasize some open problems to this end.

We would like to point out that the term rescheduling has also been used to denote solution methods, for building schedules by reordering tasks (Hasebe et al., 1991), for improving suboptimal schedules before execution (Roslöf et al., 2001; Méndez and Cerdá, 2003, 2004), or as a temporal decomposition heuristic technique to solve large-scale scheduling problems (Rodrigues et al., 1996; Dimitriadis et al., 1997; Li and Ierapetritou, 2010; Kopanos and Pistikopoulos, 2014). This use of the term rescheduling as a solution method is not the focus of our review.

The paper is structured as follows: In Section 2, through numerical examples, we motivate the need for rescheduling. In Section 3, we review previous work. In Section 4, we present examples to illustrate a limitation inherent in the current event-driven rescheduling approaches. In Section 5, we discuss online scheduling, review recent works in this direction, and propose a method classification. In Section 6, through a case study, we demonstrate how some modifications to the problem solved online can affect the quality of the implemented schedule. Finally, we summarize and provide future directions in Section 7.

* *Corresponding author.*
E-mail address: christos.maravelias@wisc.edu (C.T. Maravelias).
http://dx.doi.org/10.1016/j.cherd.2016.10.035
0263-8762/

Please cite this article in press as: Gupta, D., et al., From rescheduling to online scheduling. Chem. Eng. Res. Des. (2016), http://dx.doi.org/10.1016/j.cherd.2016.10.035

2. Motivating examples-I

2.1. *Event-triggered rescheduling*

To illustrate why rescheduling is necessary on occurrence of an event, we consider a problem, wherein for the network shown in Fig. 1, an order of 12 tons of products M2 and M3 is due at time $t = 13$. The scheduling model and the parameter values that we use are presented in Appendix A. We compute an initial schedule at $t = 0$ (shown in Fig. 2A), where it is seen that the order for 12 tons of both M2 and M3 is met (just) on time, i.e. at $t = 13$. However, through the execution of the schedule if an unexpected *event* happens, e.g. T3 gets delayed (say, due to electric-power loss) by 1 h, between $t = 3$ and $t = 4$, the natural reaction to this event on a typical shop-floor would be to delay (simply right-shift) the yet to be executed schedule ($t = 4$ onwards) by 1 h (Fig. 2B). This results in part of the order due at $t = 13$, specifically, 7 tons of M2, being met late. If instead, when the event is detected at $t = 4$, the remaining schedule is re-computed (as opposed to right-shifted), then the scheduled batch-sizes of T2 get revised to minimize the amount of order delayed. In this new schedule (Fig. 2C), although a total of 15 tons of M2 is produced, as opposed to the 12 tons needed due to the min–max batch-size (β^{MIN}/β^{MAX}) restrictions, only 2 tons of M2 is delayed with respect to meeting the order. This is a significant improvement over the simple right-shift policy. Re-computing the schedule helped reduce, at least partially, the overall effect of the delay. Hence re-computing the schedule is necessary on occurrence of an event (e.g. task delays, yield losses, unit breakdowns, demand changes, etc.).

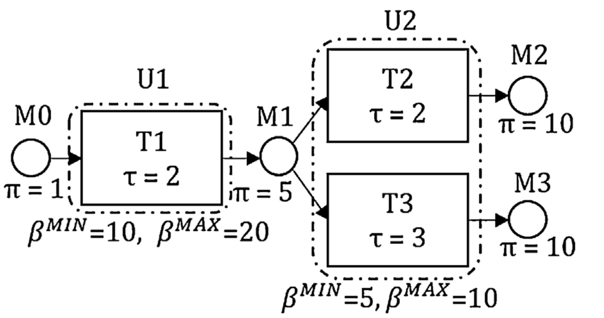

Fig. 1 – State task network (STN) (Kondili et al., 1993) representation of two units (U1–U2), three tasks (T1–T3), and four materials (M0–M3). Tasks (rectangular nodes) are connected by arcs to corresponding input/output materials (circular nodes). Material prices are denoted by π, task processing times by τ, and task-batch-size lower/upper bounds by β^{MIN}/β^{MAX}. T1 can be carried out in U1, while T2 and T3 can be carried out on U2.

2.2. *Accounting for uncertainty a priori*

It can be seen in the above example, that the original schedule (Fig. 2A) had no "slack" built in to absorb the effect of unexpected delays. If the scheduler knew that up to 1 h delay (at any point of time) was expected, the original schedule could have been, to begin with, left-shifted to start at $t = 0$ (Fig. 3A). This would have allowed the order to be met on time using the right-shift policy (Fig. 3B). In fact, it happens to be so in this example, that even if the remaining schedule, from $t = 4$ onwards was re-computed, it would result in an identical schedule as that from the right-shift policy. This shows that accounting for uncertainty appropriately when generating a schedule can be beneficial, as it can partially or fully mitigate the effect of uncertainty. However, generating high quality closed-loop schedules while, at the same time, accounting for uncertainty remains an open challenge. To illustrate, in the previous example, the original schedule was computed to guard against the scenario where a 1 h delay is observed. If the delay is not observed, which is a more likely outcome, the generated schedule would be suboptimal.

3. Rescheduling due to uncertainty

3.1. *Preliminaries*

Karimi and Reklaitis (1985) suggested changes in operator response time and raw material quality, fluctuations in utility availability, minor equipment malfunctions, etc. as possible causes for batch process variability. They carried out a variability analysis for intermediate storage units in a batch processing plant, and emphasized how even small variations can propagate and affect all scheduled operations.

Traditionally, rescheduling has been thought to be needed as a reaction to an event, thus termed reactive scheduling. Early approaches focused on creating a schedule using nominal values of the scheduling parameters (processing times, yields, etc.) to create a schedule before the uncertainty is observed, and then resorted to reactive scheduling to update the remaining (un-executed) part of this schedule in order to provide an immediate response to an unexpected event. This update or repair of the schedule can be achieved either through heuristics (task-time shifting, mixing-splitting of batches, etc.), or through regeneration of the yet-to-be-executed part of the schedule using the nominal model

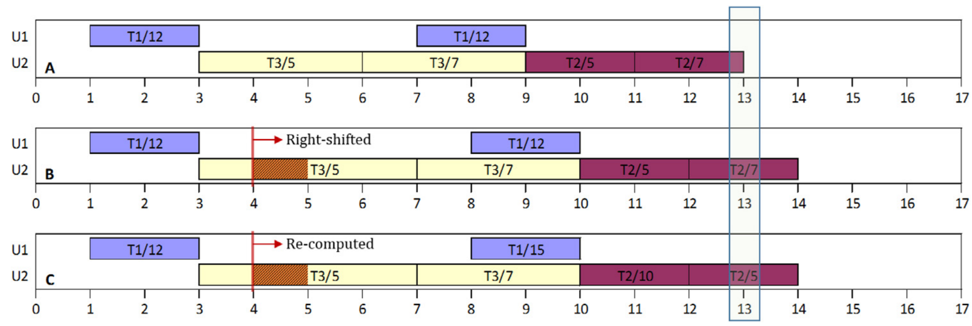

Fig. 2 – (A) Schedule to meet an order for 12 tons of M2 and M3 each due at $t = 13$. (B) A delay in T3 (due to electric-power loss) between $t = 3$ and $t = 4$, requires right-shifting of the schedule. This results in part of the order (7 tons) delayed. (C) Instead of simple right-shift, the schedule is re-computed at $t = 4$, thereby, reducing the part of the order delayed (to 2 tons).

Please cite this article in press as: Gupta, D., et al., From rescheduling to online scheduling. Chem. Eng. Res. Des. (2016), http://dx.doi.org/10.1016/j.cherd.2016.10.035