Chemical Engineering Science 68 (2012) 54–71

Contents lists available at SciVerse ScienceDirect

Chemical Engineering Science

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

A novel technique for prediction of time points for scheduling of multipurpose batch plants

Reshid Seid [a], Thokozani Majozi [a,b,*]

[a] Department of Chemical Engineering, University of Pretoria, Lynnwood Road, Pretoria 0002, South Africa
[b] Modelling and Digital Science, CSIR, Pretoria, South Africa

ARTICLE INFO

Article history:
Received 7 January 2011
Received in revised form
18 June 2011
Accepted 30 August 2011
Available online 22 September 2011

Keywords:
Scheduling
Multipurpose
Optimization
MILP
Optimal
Continuous-time
Time point

ABSTRACT

This paper presents a mathematical technique for prediction of the optimal number of time points in short-term scheduling of multipurpose batch plants. The mathematical formulation is based on state sequence network (SSN) representation. The developed method is based on the principle that the optimal number of time points depends on how frequent the critical unit is used throughout the time horizon. In the context of this work, a critical unit refers to a unit that is most frequently used and it is active for most of the time points when it is compared to other units. A linear model is used to predict how many times the critical unit is used. In conjunction with knowledge of recipe, this information is used to determine the optimal number of time points. The statistical R^2 value obtained between the predicted and actual number of optimal time points in all the problems considered was 0.998, which suggests that the developed method is accurate in determining optimal number of time points. Consequently this avoids costly computational times due to iterations. In the model by Majozi and Zhu (2001) the sequence constraint that pertains to tasks that consume and produce the same state, the starting time of the consuming task at time point p must be later than the finishing time of the producing task at the previous time point $p-1$. This constraint is relaxed by the proposed models if the state is not used at the current time point p. This relaxation gives a better objective value as compared to previous models. An added feature of the proposed models is their ability to exactly handle fixed intermediate storage (FIS) operational philosophy, which has proven to be a subtle drawback in published scheduling techniques.

1. Introduction

Batch processes differ from the continuous processes in many ways, the main of which is that time is inherent in batch processes. In batch process every task has a definite duration, with starting and finishing times, whereas in continuous processes time is important during non-steady state operation. As a result of this, the scheduling of batch processes is vital to the operation of any batch facility. In batch plants, detailed requirements for the various products may be specified on a day-to-day basis. A production schedule must indicate the sequence and manner in which the products are to be produced and specify the times at which the process operations are to be carried out. It is clear that the overall productivities and economic effectiveness of batch plants depend critically on the production schedule as it harmonizes the entire plant operation to attain production goals. While flexibility of batch plants improves productivity, it also makes plant scheduling a challenging task. Much research has gone in the past developing mathematical optimization models for scheduling of batch plants targeting in obtaining a better optimal objective value and computational time.

Méndez et al. (2006), Floudas and Lin (2004) and Shaik et al. (2006) presented excellent reviews of the current scheduling techniques based on different time representations and associated challenges. In their review the different models are classified as slot-based, event-based and precedence-based (sequence-based) time representation. In the slot-based models (Pinto and Grossmann, 1994, 1995; Lim and Karimi, 2003; Liu and Karimi, 2007a,b, 2008) the time horizon is divided into “nonuniform unknown slots” and “tasks start and finish in the same slot”. On the other hand, there exist slot models that use nonuniform unknown slots, where tasks are allowed to continue to the next slots (Sahinidis and Grossmann, 1991; Schilling and Pantelides, 1996; McDonald and Karimi, 1997; Karimi and McDonald, 1997; Lamba and Karimi, 2002a,b; Reddy et al., 2004; Sundaramoorthy and Karimi, 2005; Erdirik-Dogan and Grossmann, 2008).

The event-based models can also be categorized into those that use uniform unknown events, where the time associated with the events is common across all units (Maravelias and

* Corresponding author at: Department of Chemical Engineering, University of Pretoria, Lynnwood Road, Pretoria 0002, South Africa. Tel./fax: +27 12 420 4130.
E-mail address: thoko.majozi@up.ac.za (T. Majozi).

0009-2509/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ces.2011.08.048

Grossmann, 2003; Castro et al., 2004) and those that use unit-specific events, where the time associated to the events can be different across the units (Ierapetritou and Floudas, 1998a,b; Majozi and Zhu, 2001; Wu and Ierapetritou, 2004; Janak et al., 2004, 2005, 2006a,b, 2007; Janak and Floudas, 2008; Lin and Floudas, 2001; Lin et al., 2002, 2004; Shaik et al., 2006; Shaik and Floudas, 2008; Shaik and Floudas, 2009; Li et al., 2010). Shaik et al. (2006) and Shaik and Floudas (2007, 2008) concluded that due to heterogeneous location of events across the unit, the unit-specific event-based models require less events and perform better as compared to both the global event-based and slot-based models.

The sequence-based or precedence-based representation uses either direct precedence (Méndez and Cerdá, 2000; Hui and Gupta, 2000; Gupta and Karimi, 2003; Liu and Karimi, 2007a,b) or indirect precedence sequencing of pairs of tasks on units (Méndez et al., 2000, 2001; Méndez and Cerdá, 2003, 2004; Ferrer-Nadal et al., 2008). The models do not require pre-postulation of events and slots. Shaik and Floudas (2009) proposed an MILP formulation that used three index binary variables. Their formulation used unit-specific event-based time representation, where tasks were allowed to continue processing on multiple event points. Susarla et al. (2010) presented models that use unit-specific slots that allowed tasks to span over multiple slots. Their models also allow nonsimultaneous transfer of material into a unit to get a better schedule.

Li and Floudas (2010) presented optimal time point determination based on the model of Shaik and Floudas (2009). They used an iterative procedure to obtain the maximum number of event points and determine the critical intermediate states. The minimum number of event points is determined by the approach proposed by Janak and Floudas (2008). A branch and bound strategy is then used based on the minimum and maximum number of event points to determine the optimal number of event points. In all continuous-time representation methods, the optimal number of event points is obtained by iteration. As the time horizon increases, both the iteration and binary variables required increases, which drastically increases the time required to get the optimal schedule.

In this paper, the scheduling techniques are formulated based on the SSN representation that results in a MILP problem. The proposed models address the suboptimality by the model of Majozi and Zhu (2001) because of the sequence constraint associated with different tasks in different units. This suboptimality is addressed without spanning a task over multiple time points unlike the model by Shaik and Floudas (2009), where a task spans over multiple time points. Let us examine the sequence constraint of different tasks in different units in the model as presented by Majozi and Zhu (2001) and Shaik and Floudas (2008):

$$t_u(s_{in,j},p) \geq t_p(s_{out,j'},p), \quad \forall j,j' \in J, p \in P, s_{out,j'} = s_{in,j} \quad \text{Majozi and Zhu (2001)}$$

$$T^s(i,n+1) \geq T^s(i',n) + \alpha_{i'} w(i',n) + \beta_{i'} b(i',n) - H(1-w(i',n)),$$
$$\forall s \in S, i' \in I_s^p, i \in I_s^c, i \in I_j, i' \in I_{j'},$$
$$j,j' \in J, j' \neq j, n \in N, n < N \quad \text{Shaik and Floudas (2008)}$$

The above constraints state that the starting time of the consuming task at the current event point n should be after the end time of the producing task at the previous event $n-1$, which need not be true if there is sufficient material for the consuming task to start production, as a result the above constraints lead to suboptimal results. Moreover, new constraints are developed to handle FIS operational philosophy, which has been inadvertently violated by the unit-specific time point-based models of Ierapetritou and Floudas (1998a,b) and Majozi and Zhu (2001). The models use the continuous-time representation based on unit-specific time points. The developed method to determine the optimal number of time points is based on the branch and bound strategy on the predicted optimal number of time points obtained by solving linear programming maximization (LP MAX) and linear programming minimization (LP MIN) of the scheduling problem. LP MAX and LP MIN are solved only once to predict the optimal number of time points.

2. Motivation

In the continuous-time representation the optimal number of time points that gives the optimal objective value is found through iteration. This is done by increasing the number of time points at each iteration by one until the objective value converges. The objective value may not change with an increment of one additional time point, but may change with an increment of two or more. For example in Case I in this paper, where duration constraints depend on batch size, for a time horizon of 36 h, the objective value for both time points 11 and 12 is 445.5. The iteration is stopped at this point giving an optimal objective value of 445.5. However, if the time points is increased by one (which is 13), a better objective value of 447 is obtained, so that the optimal number of time points is not 11 but 13. This indicates that the iteration method of obtaining the optimal number of time points (where the criteria is to stop when the solution does not improve by adding one time point to the previous one) is subject to a suboptimal solution, unless it is checked further with the addition of two or more time points.

Again for Case I where duration constraints is fixed (not dependent on batch size), for a time horizon of 168 h the objective value is 3525 at time points 73 and requires more than 40,000 s of CPU time. At time points 74 the objective value is 3350 and requires a CPU time of more than 40,000 s. At time points 75 the objective value is 3350. Increasing the number of time points beyond time points 74 does not improve the objective value. As a result, the optimal number of time points is 74. A time horizon of 168 h needs 74 time points and only iteration 73 require a specified CPU time of about 11 h. In the iteration method of getting the optimal objective value, the CPU time required is the sum of the CPU times of each iteration. This becomes computationally costly as the time horizon increases. For complicated problems where each iteration takes a day, a number of days will be required to obtain the objective value, which is not desirable for batch plants where it is usually a norm to schedule on a daily or weekly basis. Moreover, process shifts might necessitate a schedule revisit in the order of hours, thereby militating against this iterative procedure.

3. Problem statement

In the scheduling of multipurpose batch plants, the following are given: (i) the production recipe that indicates the sequence of unit processes whereby the raw materials are changed into products, (ii) the capacity of a unit and the type of tasks the unit can perform, (iii) the maximum storage capacity for each material and (iv) the time horizon of interest.

Using the given data, it is required to determine (i) the maximum achievable profit of the plant, (ii) the minimum makespan if throughput is given and (iii) a production schedule related to the optimal resource utilization.

4. Mathematical model using state sequence network (SSN)

It is important to explain the effective state in the SSN representation because it renders the opportunity to reduce the