I&EC research
Industrial & Engineering Chemistry Research

Article

Cite This: *Ind. Eng. Chem. Res.* 2019, 58, 8172−8183

pubs.acs.org/IECR

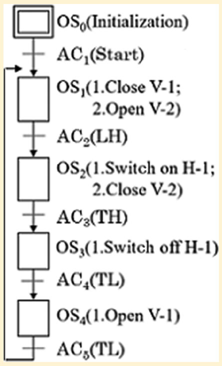
Design Approach To Synthesize, Validate, and Evaluate Operating Procedures Based on Untimed Automata and Dynamic Simulation

Tai-Yu Chen and Chuei-Tin Chang*

Department of Chemical Engineering, National Cheng Kung University, Tainan, Taiwan 70101

Supporting Information

ABSTRACT: A systematic design approach is proposed in this paper to synthesize, validate, and evaluate operating procedures for any given chemical process. To facilitate efficient procedure synthesis, the extended finite automata (EFA) are adopted in this work to model all components in a chemical process according to simple configuration rules. The intended operation is then divided into several stages on a case-by-case basis and each characterized with a unique set of attributes, e.g., stable operation, condition adjustment, phase change, reaction, material charging, and/or unloading. The control specifications of every stage should then be stipulated accordingly and described with automata. All observable event traces (OETs) are extracted from the system model assembled by synchronizing all aforementioned automata. The candidate operating procedures can be summarized with sequential function charts (SFCs) that mimic these OETs. The commercial package ASPEN PLUS DYNAMICS has been used to validate such SFCs in simulation studies. Since several candidates can usually be generated, it is also necessary to compare the simulation results so as to identify the most suitable procedure. Three realistic examples, i.e., the semibatch reaction process and the startup operations of flash drum and distillation column, are presented in this paper to demonstrate the merits of the proposed approach.



1. INTRODUCTION

Standard operating procedures (SOPs) are indispensable for running chemical plants. They are needed in performing a wide variety of essential tasks for continuous processes, such as the startup and shutdown operations of processing units, the emergency response actions under abnormal conditions, and equipment maintenance routines, and for virtually all production activities in batch processes. Therefore, other than the process flow diagram (PFD) and piping and instrumentation diagram (P&ID), the sequential function chart (SFC) of every SOP should also be documented carefully in process design. However, despite the fact that the modern plants are becoming more complex than they used to be, their operating procedures are still generated manually on the basis of the designer's experience in most cases. Manual synthesis of an operating procedure in a realistic system *can* be a very difficult undertaking since it is both time-consuming and error-prone. It is thus desirable to develop a systematic approach to automatically conjecture viable steps so as to achieve a specific production goal.[1]

Obviously, any operating procedure must be synthesized according to the initial system state and also the ultimate operational goal. To overcome the difficulties caused by combinatorial explosion of all possible operation pathways, many published studies have focused on issues concerning systematic procedure synthesis. The original problem formulation was first proposed by Rivas and Rudd,[2] and extensive works on the design and verification of procedural controllers were then carried out in later years. O'Shima[3] devised an algorithm to search for a series of valve operations that allow fluid flow between any two chosen points in a chemical plant. Foulkes et al.[4] constructed the so-called "condition lists" to describe all pipeline fragments and utilized AI-based search strategies to identify all possible routes between storage tanks for material transfer. Crooks and Macchietto[5] formulated a mixed-integer linear programming (MILP) model with embedded logic constraints to synthesize operating procedures for the batch processes. Uthgenannt[6] used digraphs to characterize the process networks and applied an existing search technique to uncover the material-transfer routes and the corresponding operating procedures. Yang et al.[7] made use of the symbolic model verifier to synthesize safe operating procedures and, furthermore, configured the safety interlocks accordingly. Ferrarini and Piroddi[8] suggested characterizing any given SFC with a Petri net to validate the corresponding operation schedule and to detect the presence of deadlock. Lai et al.[9] proposed to build binary integer programs (BIPs) based on Petri-net models for automatic generation of batch operating procedures. This approach was applied successfully to a beer filtration plant. Yeh and Chang[10,11] developed a systematic approach to generate procedures according to untimed automata under normal conditions and also for emergency response operations. Li et al.[12] developed an improved modeling strategy based on timed automata to create both the cyclic operation steps and the corresponding time schedules. Cochard et al.[13] presented a timed-automata based method to synthesize safe operation sequences.

Received: January 2, 2019
Revised: April 18, 2019
Accepted: April 19, 2019
Published: April 20, 2019

ACS Publications

DOI: 10.1021/acs.iecr.9b00015

Since all aforementioned studies emphasized only the procedure synthesis aspects, the resulting SFCs may not be readily acceptable in actual applications. To be specific, these SOPs were not validated either in simulation studies with credible software or in the pilot plant experiments. Furthermore, if several candidates can be generated, it is clearly necessary to evaluate them with a collection of different criteria so as to identify the most suitable procedure. Generally speaking, the previous works not only lacked efforts in verification and assessment of the synthesized procedures but also did not produce benchmark examples to establish their legitimacy for practical implementations. To fully address the above concerns, a comprehensive design approach is developed in the present study for synthesis, validation, and evaluation of alternative SFCs.

The untimed extended finite automata (EFA) are utilized in the present work for procedure synthesis. In particular, all components in a given system are first characterized with EFA according to simple modeling principles.[14,15] On the other hand, the intended operation is further divided into several distinguishable stages and the unique intrinsic features of each stage; e.g., stable operation, condition adjustment, phase change, reaction, and material charging and/or unloading, are identifiable. The so-called "control specifications" of every stage can then be described accordingly with automata so as to set the target state, to create different operation paths via state splitting, to limit feasible operations to those that follow only the designated partial sequences, and to avoid unsafe operations by stipulating illegal strings, etc. A system model and the corresponding observable event traces (OETs) can then be generated by synchronizing all aforementioned automata via the embedded functions of the free software SUPREMICA.[16] For any practical application, one or more operating procedures can be easily extracted from these traces and formally summarized with SFCs.

A popular commercial package, i.e., ASPEN PLUS DYNAMICS, is used next to validate these SFCs in simulation studies. Since more than one candidate may be generated, they are evaluated on the basis of several economic performance indices, e.g., the completion time, the energy consumption level, and the total amount of off-spec products. Finally, it is also possible to further fine-tune the best procedure by adjusting the critical parameters in SFCs. Three realistic examples, i.e., the semibatch reaction process and the startup operations of flash drum and distillation column, are presented to demonstrate the merits of the proposed design approach.

2. EXTENDED FINITE AUTOMATA

To facilitate a clear description of the proposed model construction method, a brief review of the automaton structure is needed here. Traditionally, a deterministic untimed automaton A can be viewed as a six-tuple:

$$A = (X, E, f, \Sigma, x_0, X_m) \tag{1}$$

where X is the set of system states; E is the event set; $f{:}X \times E \rightarrow X$ represents the state transition function; $\Sigma{:}X \rightarrow 2^E$ denotes the active event function and 2^E is the power set of E (i.e., the set of all possible subsets of E); $x_0 \in X$ is the initial state; $X_m \subset X$ is the set of marked states. The transition function $f(x,e) = x'$ means that a transition from state $x \in X$ to another state $x' \in X$ is caused by event $e \in E$, while the active event function $\Sigma(x)$ can be regarded as the set of active events at state x. The sketch of an example automaton can be found in Figure 1. The circles (S0, S1, and S2) are referred to as *places* and they are used to represent system states, while the directed arcs denote events (E1, E2, and E3). The initial state is indicated with an input arrow and the marked state is darkened.

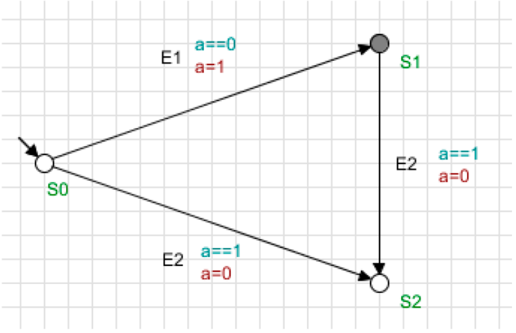


Figure 1. Graphic representation of an automaton.

The extended finite automaton (EFA) is an extended version of the above structure. Specifically, each event in EFA is equipped with two extra attributes, i.e., variable and guard, and they are further explained below:

- An integer variable (with user-specified upper and lower bounds) can be used to update the equipment state after completing an event-driven transition. An example is also shown in Figure 1, in which variable a is updated to 1 according to the equation "a = 1" via event E1.
- A guard is the sufficient condition of the corresponding state transition. Let us again consider Figure 1 as an example and assume that the initial value of variable a is 0. Therefore, only event E1 is permissible at the initial state S0 due to the logic constraint "a = = 0" and, when S1 is reached after state transition, this variable should be updated to 1.

3. PROCESS STRUCTURE

To facilitate clear illustration of the process structure, let us consider the startup operation of the continuous flash process in Figure 2 as an example. It is assumed that, at steady state, the feed is a mixture of 30 wt % water and 70 wt % methanol and its flow rate, temperature and pressure are kept at 26 000 kg/h, 20 °C, and 1.1 bar, respectively. The steady-state temperature and pressure in the flash drum are set at 75 °C and 1.01 bar, respectively, while the corresponding liquid level is 2.5 m. It is also required that the concentration of methanol in the top product should not be lower than 83 wt %. In this system, there are four proportional−integral−derivative (PID) controllers (FC01, TC01, PC01, and LC01) for controlling the feed rate, the temperature, the vapor pressure, and the liquid level in flash drum, respectively. The heating medium in the heater is assumed to be low pressure steam. The corresponding actuators are control valves, i.e., Vin, Vlps, Vvap, and Vliq. It is also assumed that, initially, all valves are closed, all controllers are on manual, and the flash drum is empty and at room temperature.

Basically every identifiable hardware item in the PFD is treated as a component in this study, and they are classified into a five-level hierarchy according to Figure 3. The top-level component is usually a programmable logic controller (PLC) or human operator. The PID controllers and their actuators (i.e., FC01/Vin, TC01/Vlps, PC01/Vvap, and LC01/Vliq) are classified as the second-level components. The material and energy flows surrounding each unit (i.e., INPUT, TOPPRO, BOTPRO, and the energy flow from heater to flash drum) in

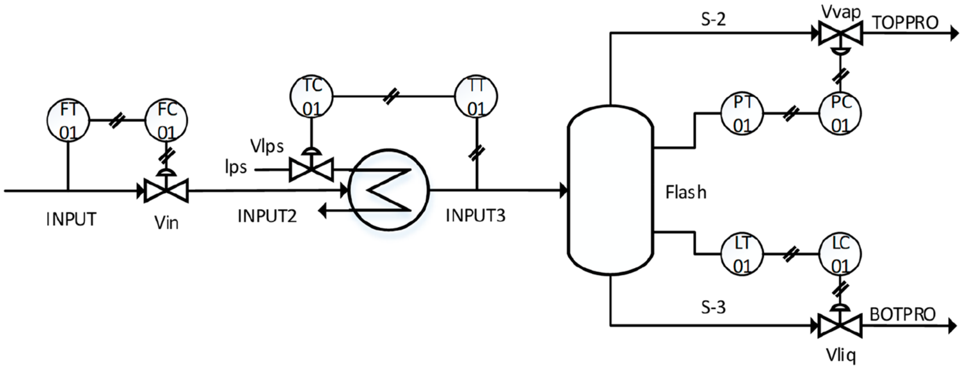


Figure 2. PFD of a continuous flash process.

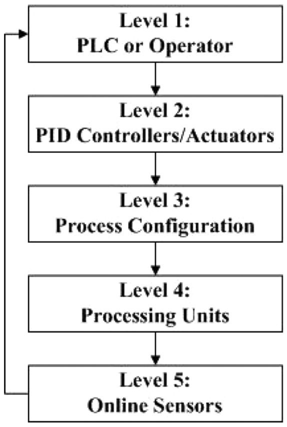


Figure 3. Hierarchical structure of a chemical process.

the subsequent level are viewed as the third-level components. Every processing unit, such as the flash drum in the present example, is regarded as a fourth-level component. All online sensors (i.e., FT01, TT01, PT01, and LT01) are grouped into level 5.

4. COMPONENT MODELS

Every component in a given process is modeled with an automaton in this study. To build this model for a component, all its normal conditions and failed states should be enumerated and represented with distinct places. The initial state should be indicated by pointing to the corresponding place with an arrow, but there is no need to assign the marked states in a component model. All events that facilitate state transitions should then be identified and each represented with a directed arc between its input and output places (states). Also, if necessary, the guard(s) of every event and the updated variable value(s) can be attached to the corresponding arc.

Note that there is no need to construct an automaton to describe a level-1 component, i.e., PLC or human operator, since the operating procedure is not available a priori. All component models in the other levels can be found in the Supporting Information. For illustration conciseness, let us consider in the present section only the level-2 components, i.e., the actuators and the PID controllers, as examples.

4.1. Actuator. To be specific, let us construct an automaton to characterize control valve Vin according to the model building principles presented above (see Figure 4). The places Vin_full_close and Vin_full_open respectively denote two extreme states of Vin, i.e., the fully closed and open positions, while the other three places between them are adopted to represent the partial openings of 25, 50, and 75%. It is also assumed that Vin is fully closed before the startup operation. Any valve state can be driven to another via a sequence of adjustment steps, i.e., the valve opening actions (oVin_0to1, oVin_1to2, oVin_2to3, and oVin_3to4) and the valve closing actions (cVin_4to3, cVin_3to2, cVin_2to1, and cVin_1to0). Two additional attributes of events, i.e., variable and guard, are also utilized on the corresponding arcs. An integer variable is used to update the component state after completing an event-driven transition, while the guard (guards) is (are) used to stipulate the sufficient condition(s) of a state transition. Let us consider event oVin_0to1 as an example. Its guards (prerequisites) are expressed as s_flow = = 1 &s_flow > A_Vin&A_Vin! = 4 and they can be interpreted as follows:

(i) s_flow = = 1: The controller output signal is at the qualitative value of 1.

(ii) s_flow > A_Vin: The output signal of flow controller FC01 is larger than the air pressure corresponding to the current position of valve Vin.

(iii) A_Vin! = 4: The current air pressure at valve Vin does not reach a maximum.

Note that the guards on other arcs between places Vin_full_close and Vin_full_open can be interpreted in a similar fashion. Two additional places, Vin_SC and Vin_SO, are included in this model to characterize the failures when Vin is stuck at the closed and open positions, respectively. These failed states can be reached via events f_VinSC and f_VinSO, and the resulting values of A_Vin should be fixed at 0 and 4, respectively. In the present study, all failed states are ignored because the objective here is to synthesize normal operating procedure. Since at least two places must be adopted to characterize a component condition, e.g., the positions of a control valve in the above automaton, it is clear that the above approach is only effective for modeling simple systems with relatively few state variables. To facilitate easy construction and a concise representation of automata, the component models have been "compressed" in this study. Specifically, the component Vin is modeled alternatively with an automaton using significantly fewer places and transitions (see Figure A1 in the Supporting Information).

4.2. PID Controller. Generally speaking, the roles of a PID controller in executing the steps in a SOP are defined according to its actuator in the following two scenarios:

(i) If the actuator is a hand valve or a solenoid valve, then the PID controller is obviously not needed.

(ii) If the actuator is a control valve, then the PID controller can be utilized to vary its opening in two alternative modes:

DOI: 10.1021/acs.iecr.9b00015
Ind. Eng. Chem. Res. 2019, 58, 8172−8183

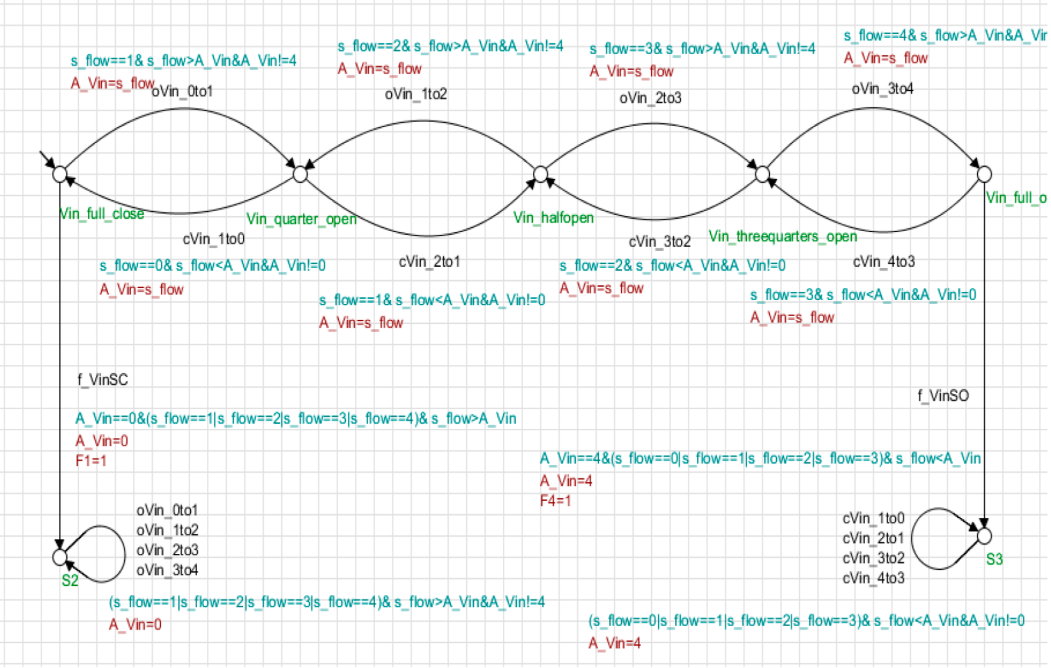


Figure 4. Traditional model of control valve Vin in flash startup example.

(a) If the controller is set at AUTO mode, it is necessary for the operator or PLC to alter the set point so as to adjust the controller output signal indirectly.

(b) If the controller is set at MANUAL mode, the operator or PLC should be able to directly adjust the controller output signal.

The compressed component model of the PID controller FC01 can be found in Figure A2 in the Supporting Information.

5. INTRINSIC STAGES AND THEIR CONTROL SPECIFICATIONS

Intuitively, the system model can be constructed by synchronizing all aforementioned component models with an automaton that specifies the final target of operation. This target-setting automaton for the flash startup operation is given in Figure 5.

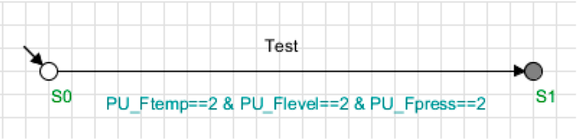


Figure 5. Ultimate target of flash startup operation.

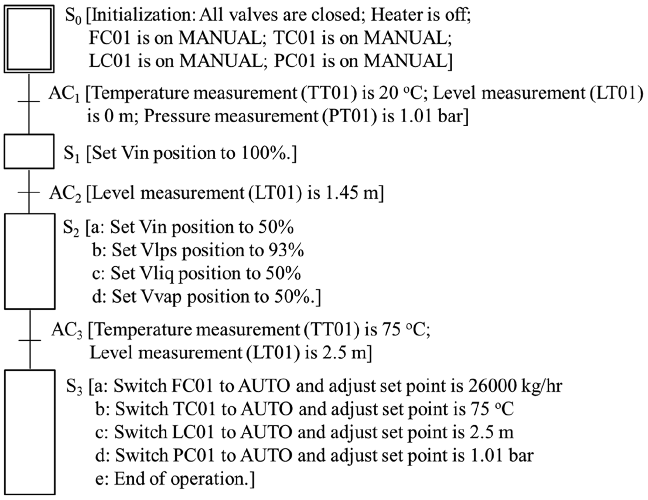
Since only generic engineering knowledge is utilized to build the automata in the Supporting Information (part A), an overwhelmingly large number of operation pathways may be extracted from this integrated system even when its dynamics is moderately complex.

More specifically, the synchronization operation in SUPREMICA[16] yielded a complicated and unmanageable pathway network for the flash startup example. Since the dynamic behavior of a multi-input–multi-output (MIMO) system cannot be adequately described with the untimed automata developed on the basis of qualitative information only, this network consists of not only multiple feasible routes but also an extremely large number of unnecessary and impractical scenarios.

Although the final goal of a specific operation can be unambiguously given (e.g., Figure 5), it may only be approached properly via a series of stages with interim goals which are often not explicitly stipulated a priori. It is thus important to uncover these embedded subtasks and identify their unique features in advance. These features may be broadly described as (1) material charging, (2) material unloading, (3) reaction, (4) state adjustment, (5) phase change, and (6) stable operation. For illustration purposes, let us revisit the flash startup process. Based on engineering knowledge and operational experience, it is clearly necessary to place a small quantity of raw material in the flash drum in the initial stage and allow the liquid level to reach a height which is safe for intense heating. In the next stage, the temperature and pressure in the drum should be transferred to the set points and the input and output flow rates be raised to the steady-state levels. Notice also that, whenever such adjustments in operating conditions are called for, it is always beneficial to assess the pros and cons of alternative pathways that facilitate the required state transitions. Finally, the stable operating conditions should be maintained for a long period with the PID controllers. Thus, the feature sets of the above three stages may be described as follows: (1) state transfer and material charging; (2) state transfer, phase change, and material charging and unloading; and (3) stable operation.

All features in a stage should be expressed first as the "control specifications" in natural language and then represented with automata accordingly. Cassandras and Lafortune[17] suggested that five different types of automata may be constructed for use to set the target state (type A), to perform state splitting (type B), to impose a partial sequence (type C), to suppress an illegal substring (type D), and to ensure alternation between two particular events (type E). Let us again use the flash startup process as an example for illustration. For the sake of conciseness, the control specifications

DOI: 10.1021/acs.iecr.9b00015
Ind. Eng. Chem. Res. 2019, 58, 8172−8183


S_0 [Initialization: All valves are closed; Heater is off;
FC01 is on MANUAL; TC01 is on MANUAL;
LC01 is on MANUAL; PC01 is on MANUAL]

AC_1 [Temperature measurement (TT01) is 20 °C; Level measurement (LT01) is 0 m; Pressure measurement (PT01) is 1.01 bar]

S_1 [Set Vin position to 100%.]

AC_2 [Level measurement (LT01) is 1.45 m]

S_2 [a: Set Vin position to 50%
b: Set Vlps position to 93%
c: Set Vliq position to 50%
d: Set Vvap position to 50%.]

AC_3 [Temperature measurement (TT01) is 75 °C;
Level measurement (LT01) is 2.5 m]

S_3 [a: Switch FC01 to AUTO and adjust set point is 26000 kg/hr
b: Switch TC01 to AUTO and adjust set point is 75 °C
c: Switch LC01 to AUTO and adjust set point is 2.5 m
d: Switch PC01 to AUTO and adjust set point is 1.01 bar
e: End of operation.]

Figure 6. SFC-1 obtained in flash startup example.

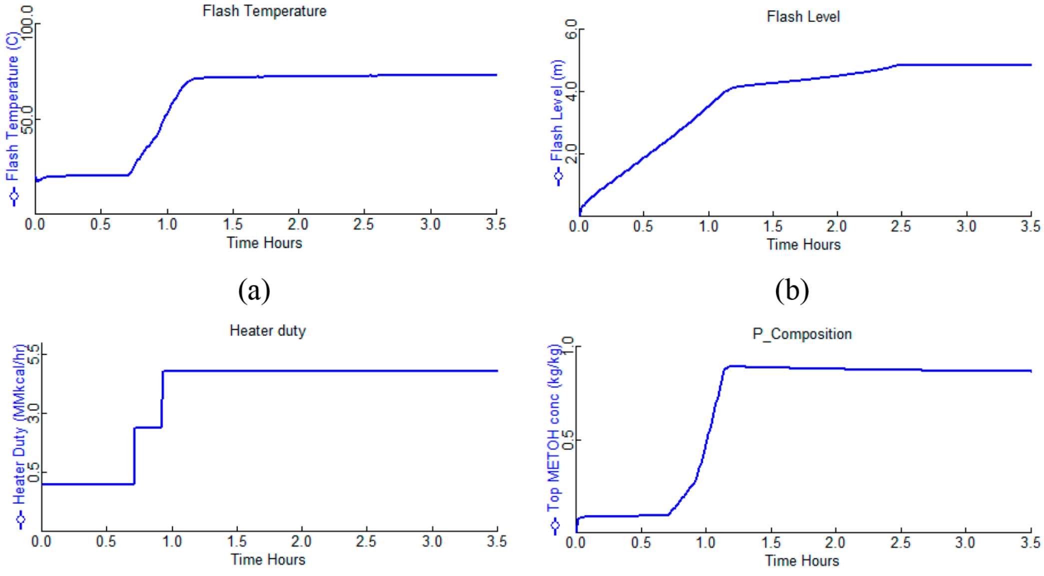
in all stages are detailed in the Supporting Information (part B).

6. PROCEDURE SYNTHESIS

The operating procedures of every stage in an operation can be produced by synchronizing all component models and the corresponding control specifications with SUPREMICA.[16] A sequential function chart (SFC) is used in this work to formally summarize the overall procedure obtained by piecing together the steps in all stages of the given operation. A total of four SFCs have been generated in the flash startup example. Specifically, the control actions and their responses in the three stages of all SFCs are listed as follows:

- SFC-1. (1) Raise the liquid level in the flash drum to 1.45 m by opening the inlet valve (Vin) fully. (2) Raise the temperature and level directly to their set points (i.e., 75 °C and 2.5 m) by adjusting the inlet valve (Vin), the outlet valves (Vvap and Vliq), and the steam valve (Vlps). (3) Switch all PID controllers from MANUAL to AUTO modes and maintain stable operation at the targeted set points.
- SFC-2. (1) Raise the liquid level in the flash drum to 1.45 m by opening the inlet valve (Vin) fully. (2) Raise the temperature to 40 °C and then to 75 °C in two consecutive steps and simultaneously raise the level to 2.5 m in one step by adjusting the inlet valve (Vin), the outlet valves (Vvap and Vliq), and the steam valve (Vlps). (3) Switch all PID controllers from MANUAL to AUTO modes and maintain stable operation at the targeted set points.
- SFC-3. (1) Raise the liquid level in the flash drum to 2.5 m by opening the inlet valve (Vin) fully. (2) Raise the temperature directly to 75 °C by adjusting the inlet valve (Vin), the outlet valves (Vvap and Vliq), and the steam valve (Vlps). (3) Switch all PID controllers from MANUAL to AUTO modes and maintain stable operation at the targeted set points.
- SFC-4. (1) Raise the liquid level in the flash drum to 2.5 m by opening the inlet valve (Vin) fully. (2) Raise the temperature to 40 °C and then to 75 °C in two consecutive steps by adjusting the inlet valve (Vin), the outlet valves (Vvap and Vliq), and the steam valve (Vlps). (3) Switch all PID controllers from MANUAL to AUTO modes and maintain stable operation at the targeted set points.

As an example for illustration, let us take a closer look at SFC-1 in Figure 6. The initial settings of PID controllers and actuators in this procedure are implemented according to S_0. The first activation conditions in SFC-1 are basically the initial sensor readings specified in AC_1. After AC_1 is verified, valve Vin is supposed to be opened fully as required in S_1 to raise the liquid level in the flash drum as quickly as possible. Upon observing the level reading of 1.45 m, i.e., AC_2, utility heating should be applied and, at the same time, the inlet and outlet flows also begin via the operation steps specified in S_2. The subsequent activation conditions in AC_3 are essentially sensor readings at the set points of temperature (75 °C) and level (2.5 m) for the continuous flash operation at steady state. The final steps of the startup operation, i.e., S_3, are operator (or PLC) actions to switch all PID controllers from MANUAL to AUTO modes and adjust their set points to the intended steady-state values, respectively. Finally, it should be noted that the valve opening of Vlps (93%) in this SFC is only an



(a) (b)

(c) (d)

Figure 7. Simulation results of the flash startup process driven by SFC-4: (a) temperature; (b) level; (c) heater duty; (d) product concentration.

DOI: 10.1021/acs.iecr.9b00015
Ind. Eng. Chem. Res. 2019, 58, 8172−8183

Table 1. Performance Indices of Flash Startup Processes Driven by SFC-1 and SFC-2

	total amt off-spec product (kg)	total amt energy consumed (MMkcal)	total operation time (h)
SFC-1	150.20	0.6688	0.51
SFC-2	175.31	0.7882	0.59

approximated (interpolated) value. This is due to the fact that, in ASPEN PLUS DYNAMICS, only the heater duty can be adjusted. If SFC-1 is to be implemented in practical application, the exact opening of the steam valve must be determined experimentally.

7. DYNAMIC SIMULATION

The candidate procedures generated with the aforementioned approach can be validated in either simulation or pilot studies. For illustration purposes, the above four SFC-driven flash startup processes have all been simulated with ASPEN PLUS DYNAMICS. The simulation results can then be compared on the basis of economic performance indices and the best one then selected accordingly.

First of all, it should be noted that the simulation studies reveal that the last two SFCs, i.e., SCF-3 and SFC-4, may be unsafe in practical applications due to delayed heating in startup operation. Specifically, the heater is turned on in these two cases only after the liquid level reaches 2.5 m (instead of 1.45 m in SFC-1 and SFC-2). Let us use SFC-4 as an example for illustration. The simulated time profiles of temperature and liquid level in the flash drum, the heater duty, and the concentration of methanol in overhead product are presented in Figure 7. It can be observed that the heater duty is adjusted first around 0.7 h when the liquid level reaches 2.5 m and then around 0.92 h when the temperature is 40 °C. Notice also that the level rises continuously to 5 m (which is the height of flash drum) around 2.49 h and stays unchanged afterward. Based on

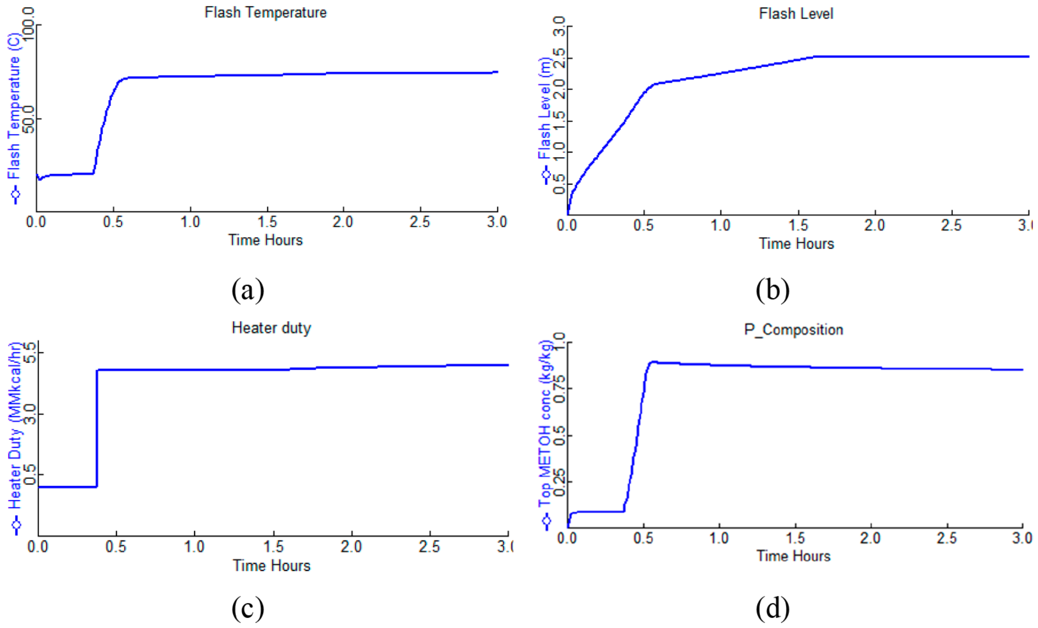


Figure 8. Simulation results of the flash startup process driven by SFC-1: (a) temperature; (b) level; (c) heater duty; (d) product concentration.

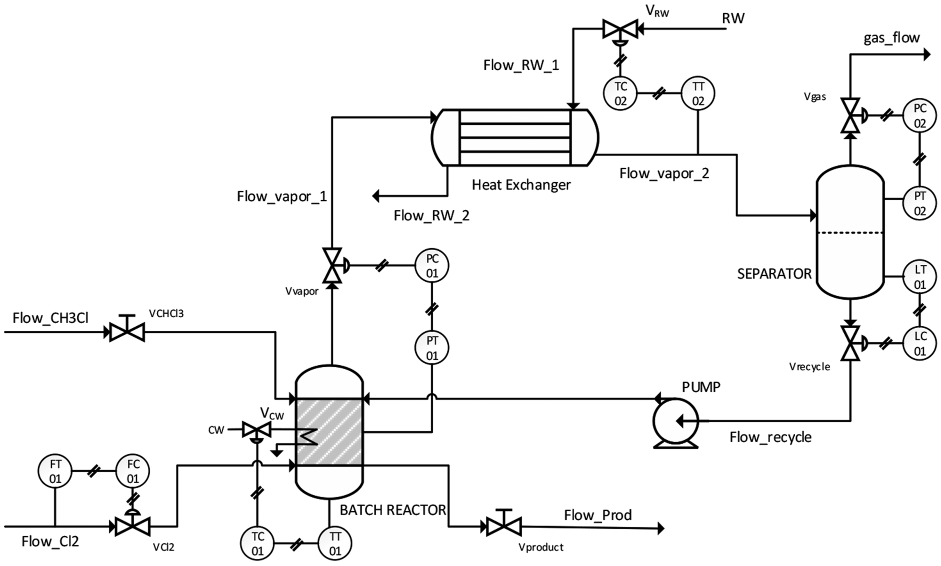


Figure 9. PFD of a batch reaction process.

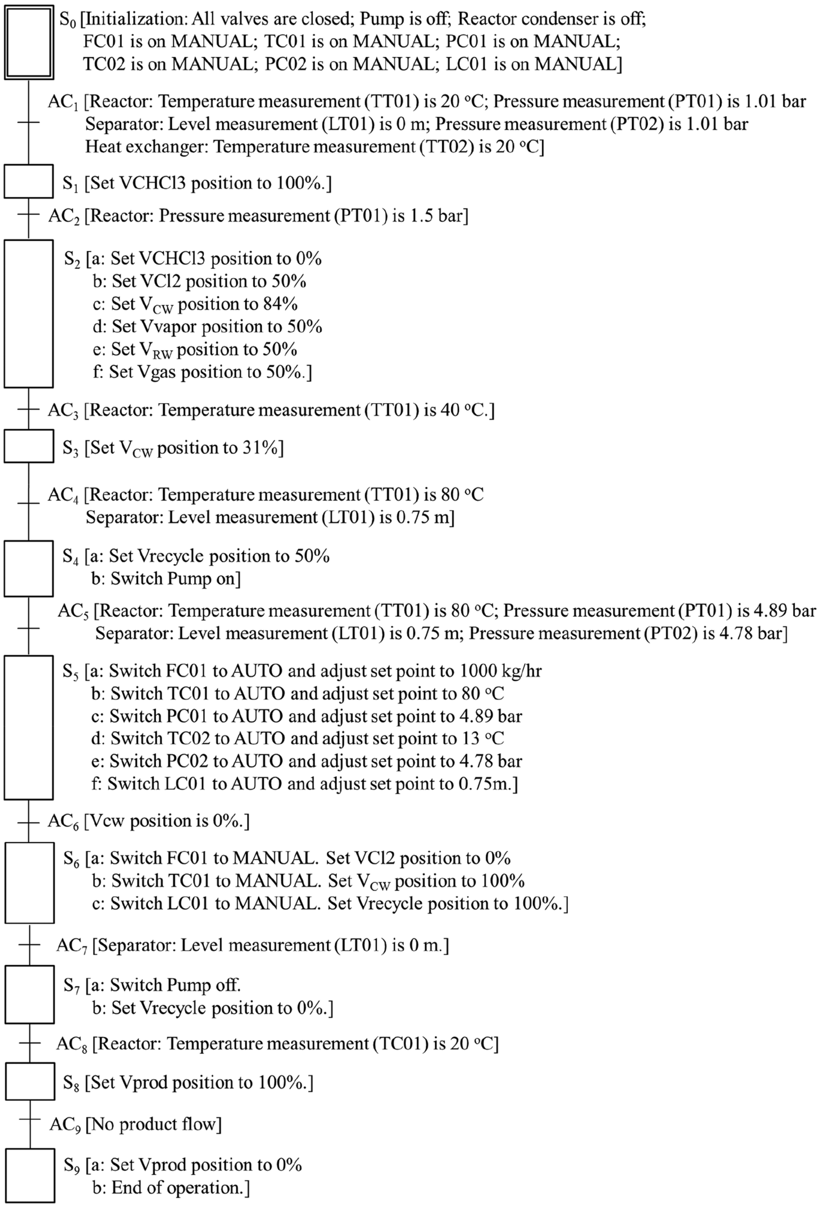


Figure 10. SFC-5 obtained in semibatch reaction example.

the observations concerning the overly high liquid level in the simulation results of SFC-3 and SFC-4, only SFC-1 and SFC-2 are compared in Table 1 according to three performance indices, i.e., the total amount of off-spec product, the total amount of energy consumed, and the total operation time. It can be found that SFC-1 outperforms SFC-2 essentially in every aspect.

On the other hand, the ASPEN simulation results in the flash startup process driven by SFC-1 are presented in Figure 8. It can be clearly verified that the concentration specification (83 wt % methanol) of the overhead product is reached at 0.51 h in Figure 8d and the corresponding temperature stabilizes around 75 °C at about the same time in Figure 8a. It can also be confirmed from Figure 8c that the liquid level eventually approaches 2.5 m, which is well below the height of flash drum (5 m).

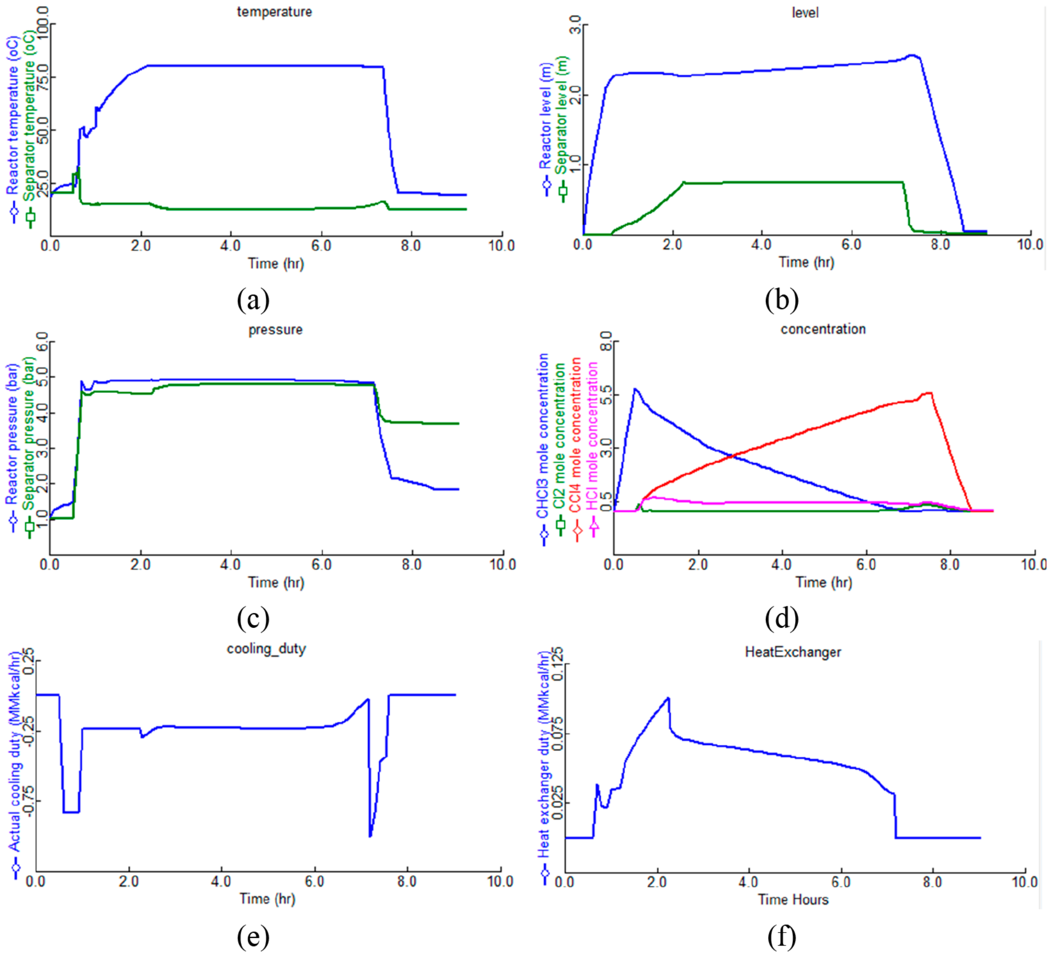
8. ADDITIONAL CASE STUDIES

To demonstrate the feasibility of proposed approach in more practical applications, two realistic examples are presented in the following.

8.1. Semibatch Reaction.

The chemical reaction considered in this example is $CHCl_3 + Cl_2 \rightarrow CCl_4 + HCl$, and the corresponding PFD can be found in Figure 9. To meet the product demand and ensure operational safety, the entire batch of 13 800 kg of one of the reactants, i.e., chloroform, is first transported into the reactor and then the more toxic and corrosive chlorine is fed at a relatively low flow rate to ensure its quick consumption. A period of stable operation can be maintained as the steady transferring process of Cl_2 begins. The reactor temperature and pressure are respectively kept steady at 80 °C and 4.89 bar in this period, while the liquid level, temperature, and pressure in the separator are at 0.75 m, 13 °C, and 4.78 bar, respectively. The concentrations of the reactant ($CHCl_3$) and the product (CCl_4) in the reactor and in the separator, however, should vary significantly with time during this so-called stable period. Finally, it should be noted that the present operation is essentially the modified version of a built-in example of ASPEN PLUS DYNAMICS. Modifications have been introduced since the existing example did not

DOI: 10.1021/acs.iecr.9b00015
Ind. Eng. Chem. Res. 2019, 58, 8172−8183



(a) (b) (c) (d) (e) (f)

Figure 11. Simulation results of the batch reaction process driven by SFC-5: (a) temperature; (b) level; (c) pressure; (d) concentration; (e) reactor cooling duty; (f) heat exchanger cooling duty.

incorporate the product discharge stage in operation and, also, adopted only the simple flow-driven mode in dynamic simulation.

8.1.1. Identification of Intrinsic Stages. As mentioned before, the two reactants in the present example are supposed to be charged one-at-a-time in sequence. The entire amount of the first reactant (chloroform) is placed into the reactor first, while the other (chlorine) is then fed at a relatively low flow rate. Since the reaction is exothermic, the operating temperature and pressure can be raised by reaction heat to their anticipated set points and, at which instance, all controller modes in this system must be switched from MANUAL to AUTO. The resulting stable operating conditions are supposed to last until some observable signs of the end of reaction are detected online. Subsequently, the product in the reactor should be cooled to room temperature and then discharged.

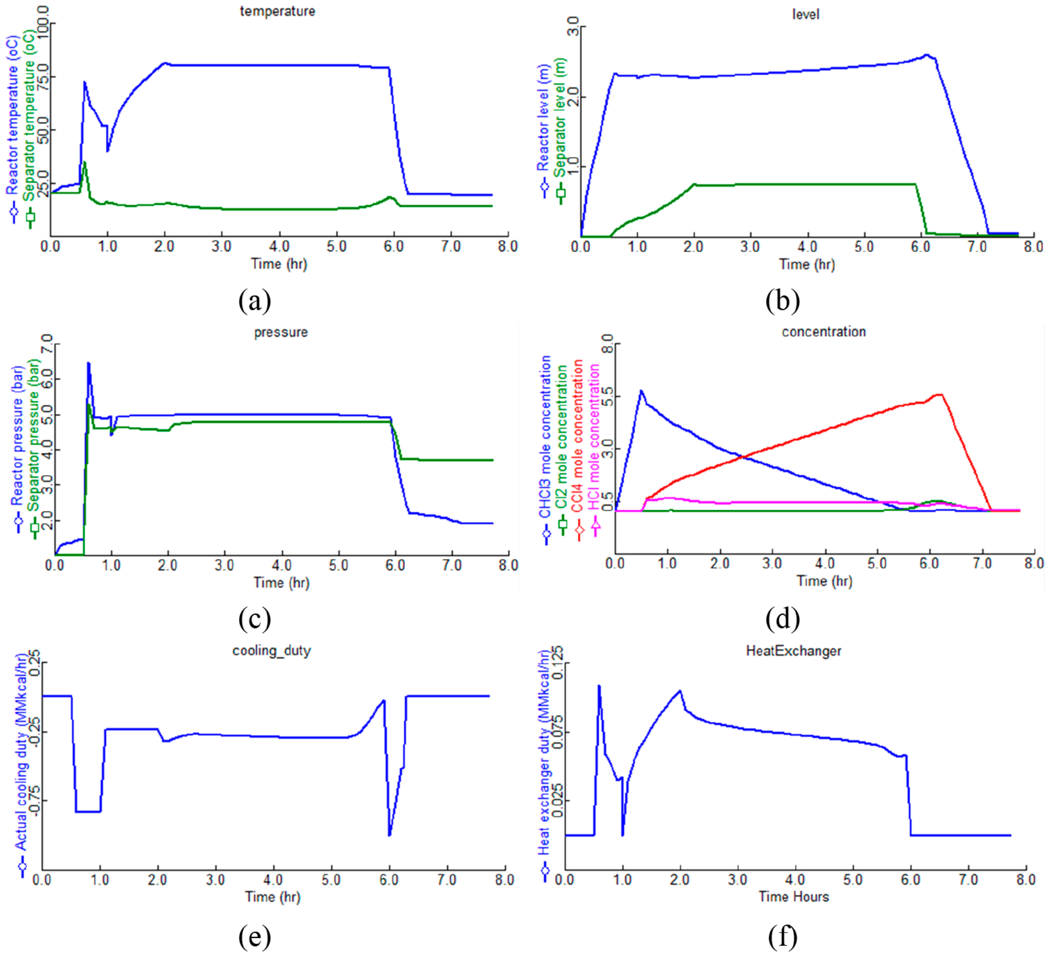
Based upon the aforementioned insights, the entire operation may be divided into five consecutive stages and their respective features are the following: (1) charging chloroform into the reactor that results in a rise in liquid level; (2) activating the exothermic reaction by charging the chlorine gas, and also removing the reaction heat with cooling water after the set points are reached; (3) stable operation; (4) lowering temperature and pressure to the safe conditions; (5) unloading product.

8.1.2. Synthesis of Operating Procedures. Four SFCs, labeled respectively from SFC-5 to SFC-8, have been generated in the semibatch reaction example. Basically only the steps in the aforementioned second stage of each SFC are unique. These steps are outlined below:

- SFC-5. Set flow controller FC01 on MANUAL and fix the chlorine flow rate to 1000 kg/h by adjusting valve VCl2. Set temperature controller TC01 on MANUAL and allow the reactor temperature to rise first to 40 °C and then to 80 °C in two steps by adjusting valve Vcw manually.
- SFC-6. Set flow controller FC01 on MANUAL and fix the chlorine flow rate to 1500 kg/h by adjusting valve VCl2. Set temperature controller TC01 on MANUAL and allow the reactor temperature to rise first to 40 °C and then to 80 °C in two steps by adjusting valve Vcw manually.
- SFC-7. Set flow controller FC01 on MANUAL and fix the chlorine flow rate to 1000 kg/h by adjusting valve VCl2. Set temperature controller TC01 on MANUAL and allow the reactor temperature to rise first to 60 °C and then to 80 °C in two steps by adjusting valve Vcw manually.
- SFC-8. Set flow controller FC01 on AUTO and fix the chlorine flow rate to 1500 kg/h by adjusting valve VCl2. Set temperature controller TC01 on MANUAL and allow the reactor temperature to rise first to 60 °C and then to 80 °C in two steps by adjusting valve Vcw manually.

For the sake of brevity, only SFC-5 is presented in Figure 10 in detail as an illustration example

8.1.3. Simulation, Validation, and Performance Assessment. All SFC-driven semibatch reaction processes have been



(a) (b) (c) (d) (e) (f)

Figure 12. Simulation results of the batch reaction process driven by SFC-6: (a) temperature; (b) level; (c) pressure; (d) concentration; (e) reactor cooling duty; (f) heat exchanger cooling duty.

simulated in this work with ASPEN PLUS DYNAMICS. For the sake of brevity, only the simulation results generated according to SFC-5 and SFC-6 are presented in Figures 11 and 12, respectively. Since in the latter case a higher chlorine flow rate results in a spike in reactor pressure (see Figure 12c) and more violent fluctuation in reactor temperature (see Figure 12a), SFC-5 should be viewed as a safer and more operable procedure.

It should be noted that the simulation results for SFC-7 and SFC-8 are quite similar to those for SFC-5 and SFC-6, respectively. Therefore, only the performance indices of the more feasible SFC-5 and SFC-7 are compared in Table 2. Notice that, since the generation rate of reaction heat is primarily governed by the chlorine feed rate, the outcomes of these two procedures are essentially the same and, thus, both are acceptable for practical applications.

Table 2. Performance Indices of Flash Startup Processes Driven by SFC-5 and SFC-7

	total cooling duty (MMkcal)	total yield CCl_4 (kg)	total operation time (h)
SCF-5	2.3604	17913	9.25
SFC-7	2.3528	17921	9.30

8.2. Distillation Startup. In this last case study, let us consider the startup operation of the continuous distillation process described in Figure 13. It is assumed that, at steady state, the feed is a mixture of 6 wt % CH_2Cl_2, 54 wt % $CHCl_3$, and 40 wt % CCl_4 and its flow rate, temperature, and pressure are kept at 10 000 kg/h, 20 °C, and 6 bar, respectively. The total number of plates in the distillation column is 20, while the feed is directed toward the tenth plate. The steady-state set-point temperatures at the seventh plate and 16th plate are set at 87.4 and 101.5 °C, respectively. The column is equipped with a condenser at the top and a reboiler at the bottom. The steady-state reflux ratio is approximately 5 mol/mol. The steady-state pressure settings at plate 1/condenser and plate 2

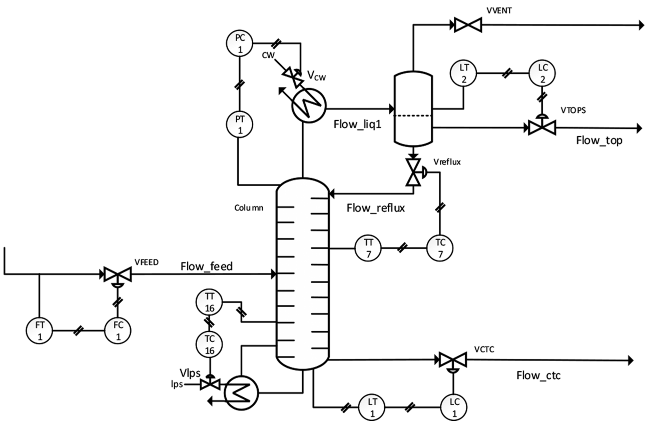


Figure 13. PFD of a continuous distillation process.

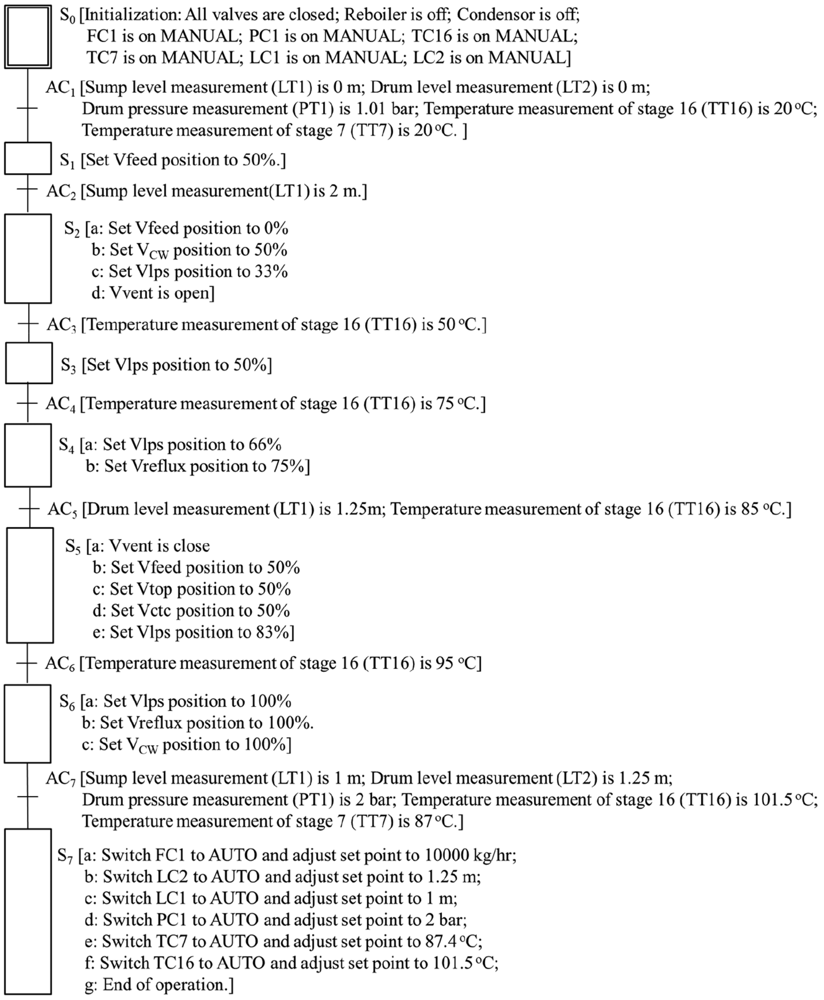


Figure 14. SFC-9 obtained in distillation startup example.

are chosen to be 2.00 and 2.02 bar, respectively, while the column pressure drop from the bottom is 0.235 bar. At steady state, the heights of liquid levels in the reflux drum and in the column sump are controlled at 1.25 and 1.0 m, respectively. It is also required that the concentration of light key ($CHCl_3$) in the top product should be greater than 81 mol % and that of heavy key (CCl_4) in the bottom product should not be lower than 97 mol %. In this system, there are six PID controllers (FC01, TC07, TC16, PC01, LC01, and LC02) for controlling the feed rate, the temperatures on the seventh and 16th plates, the condenser pressure, and the liquid levels in the column sump and reflux drum, respectively. The corresponding control valves are VFEED, Vreflux, Vlps, Vcw, VCTC, and VTOPS. It is assumed that, before the startup operation, all valves are closed, all controllers are on MANUAL, and the reflux drum and column sump are both empty and at room temperature. Finally, it should be noted that startup operation of this system is also included in ASPEN PLUS DYNAMICS as a built-in illustrative example. The objective of the present case study is to generate additional procedures with the proposed modeling approach and then compare them with the existing one.

8.2.1. Identification of Intrinsic Stages. Similar to the flash startup operation, it is also necessary to put enough feedstock into the column sump first so as to allow the liquid level to reach a reasonable height which is safe for heating. Since the present example is concerned with startup, the desired steady-state conditions should eventually be maintained with the PID controllers. Therefore, the operating conditions of the distillation column must be transferred from those at the end of initial stage to their targeted set points in the final stage and the input and output flow rates be raised to the steady-state levels. This transition process is supposed to take place in two intermediate stages, i.e., stage 2 and stage 3. First, in order to establish good contact between liquid and vapor phases in the entire distillation column, the adjustments of operating conditions are carried out in total-reflux mode in stage 2. A total-reflux configuration can be realized by keeping input and output valves (i.e., VFEED, VTOPS, and VCTC) fully closed and the reflux valve (i.e., Vreflux) and vent valve (VVENT) open, while the operating conditions are adjusted simultaneously by manipulating the cooling and heating utilities via Vcw and Vlps. Next in stage 3, when the designated intermediate system state is reached, the input and output valves should be opened and the vent valve closed to allow the column to run continuously. The steam and cooling water flows can then be further increased to drive the system toward the final steady state. Based on the above analysis, the four stages of the distillation startup operation may be characterized as follows: (1) state transfer and material charging; (2) state transfer and phase change; (3) state transfer, phase change, material charging and discharging; (4) stable operation.

8.2.2. Synthesis of Operating Procedures. In either stage 2 or stage 3, the system state is supposed to be transformed from

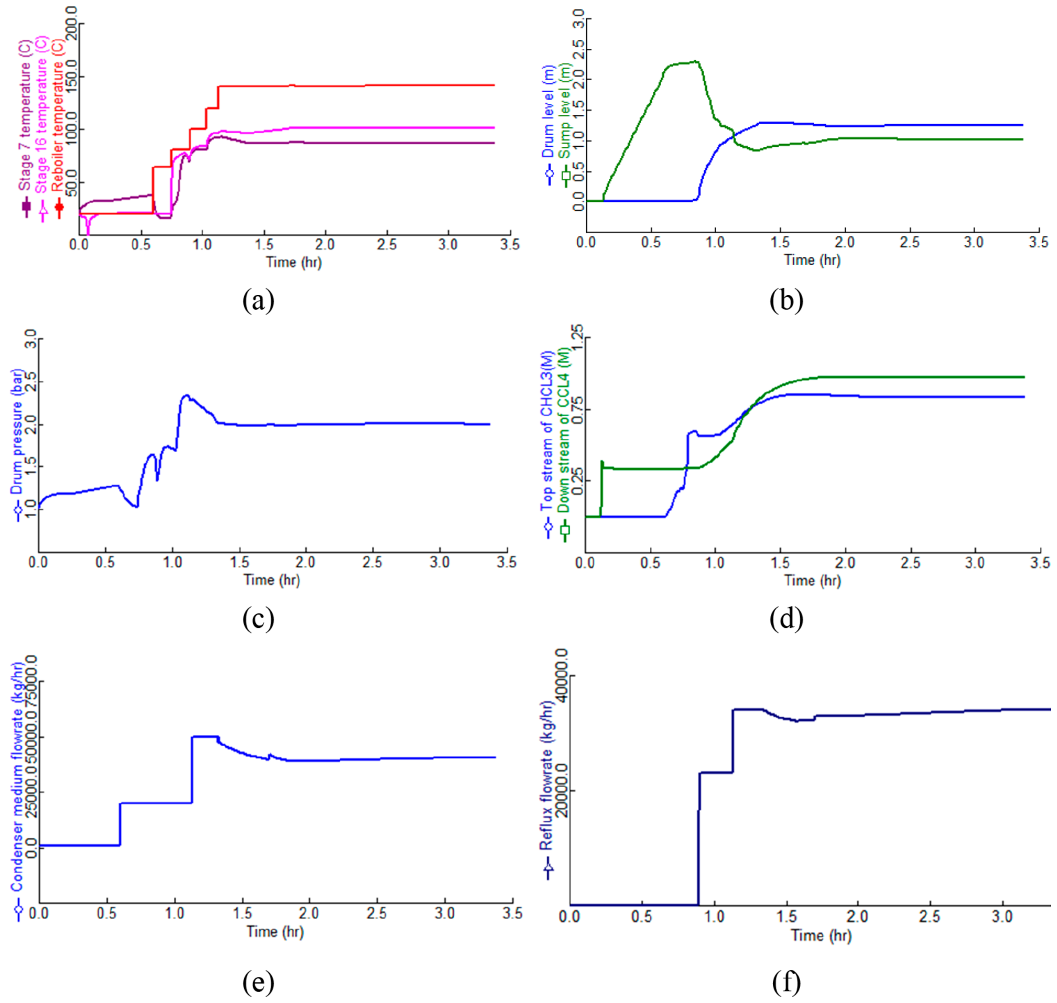


Figure 15. Simulation results of the distillation startup process driven by SFC-9: (a) temperatures of stage 7, stage 16, and reboiler heating medium; (b) liquid levels in reflux drum and column sump; (c) pressure in reflux drum; (d) overhead and bottom concentrations; (e) flow rate of condenser cooling medium; (f) reflux flow rate.

one to another. As also mentioned previously, whenever such adjustments in operating conditions are called for, it is beneficial to follow pathways that facilitate smooth transitions. Two SFCs, labeled respectively as SFC-9 and SFC-10, have been generated according to this principle. Basically, the temperature at plate 16 is manipulated in three steps in the second stage of each SFC by adjusting the reboiler duty, i.e., from 20 to 50 °C, next to 75 °C, and finally to 85 °C. In stage 3, this temperature is further altered in SFC-9 and SFC-10 respectively as follows: (1) from 85 to 95 °C and then to the target set point 101.5 °C in two steps; (2) from 85 °C directly to 101.5 °C in one step. For the sake of brevity, again only the former case (i.e., SFC-9) is presented in detail in Figure 14 for use as an illustration example.

8.2.3. Simulation, Validation, and Performance Assessment. The distillation startup process has been simulated in this work with ASPEN PLUS DYNAMICS according to SFC-9 and SFC-10. Only the simulation results generated in the former case are presented in Figure 15 for illustration brevity, while a comparison of various performance indices of the two procedures is given in Table 3.

First of all, it can be observed from Figure 15d that, for SFC-9, the steady-state concentrations of overhead and bottom products are both on spec (which are stipulated in the beginning of section 8.2). Notice also that the same trends in product concentrations also appear in the simulation results generated according to SFC-10. Although the ultimate goals of startup operation can be achieved with the above two procedures, it can be observed from Table 3 that SFC-9 essentially outperforms SFC-10 in every aspect.

Table 3. Performance Indices of Distillation Startup Processes Driven by SFC-9 and SFC-10

	total amt off-spec overhead product (kg)	total amt off-spec bottom product (kg)	total amt heating energy (MMkcal)	total amt cooling medium (kg)	total operation time (h)
SFC-9	5630	1681	23.3437	107950	1.78
SFC-10	7044	3457	25.7482	119650	2.06

8.2.4. Tuning of Total Reflux End Point. As mentioned previously in SFC-9 (see Figure 14), the plate 16 temperature reached at the end of total reflux operation is 85 °C. In order to search for an improved end point of stage 2, two additional simulation studies have been repeated for temperatures in its neighborhood, e.g., 80 and 90 °C. A comparison of these two scenarios is given in Table 4 and they are referred to as SFC-11 and SFC-12, respectively. It can be clearly observed from Tables 3 and 4 that it is beneficial to slightly raise the end-point temperature.

Table 4. Performance Indices of Distillation Startup Processes Driven by SFC-11, SFC-12, and ASPEN Built-In Procedure

	total amt off-spec overhead product (kg)	total amt off-spec bottom product (kg)	total amt heating energy (MMkcal)	total amt cooling medium (kg)	total operation time (h)
SFC-11	7617	3867	28.5956	132710	2.20
SFC-12	4632	203	21.7141	99864	1.54
ASPEN	6105	897	28.6867	121640	2.16

8.2.5. Comparison with Existing Procedure. Notice that the performance indices of SFC-11 and SFC-12 in Table 4 are also compared with those resulting from the ASPEN built-in procedure (see part C in the Supporting Information). It can be observed from Table 4 that the operation time of SFC-12 is significantly shorter than that of the ASPEN built-in procedure. As a result, the total amounts of off-spec products and the total heating and cooling duties of the former operation are all smaller than those of the latter. From the SFC presented in part C of the Supporting Information, it may be deduced that the longer operation time of the ASPEN built-in procedure is probably due to the more conservative startup practices. First of all, the liquid levels of the reflux drum and column sump are both brought back to 2 m during total reflux operation (stage 2) to ensure that there are enough inventories for the subsequent continuous operation. Second, in each reboiler heating step it is required to wait for 0.1 or 0.15 h, while the corresponding activation conditions in SFC-12 are simply the designated online measurement values of temperatures at plate 16. In addition, the operating policy of SFC-12 concerning the switching actions between MANUAL and AUTO modes of the PID controllers is more straightforward than that of the ASPEN built-in procedure. In the former case, MANUAL actions are always adopted to manipulate the actuators and the AUTO modes can only be activated when the designated set-point conditions are reached. On the other hand, the controller settings are switched from MANUAL to AUTO and vice versa throughout the startup process in the latter case and this practice inevitably prolongs the time to reach final steady state.

9. CONCLUSIONS

A generic approach has been developed in this work for systematically creating operating procedures based on untimed automata. The proposed procedure-synthesis steps include the following: (1) constructing an automaton model for each component in a given PFD; (2) dividing operation into stages and developing automata to represent the control specifications of every stage; (3) assembling the system model of each stage and consolidating them into a single SFC. The commercial software ASPEN PLUS DYNAMICS was used to validate and evaluate the candidate SFCs. Finally, this approach has been tested extensively and successfully on realistic cases.

■ ASSOCIATED CONTENT

Ⓢ Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.iecr.9b00015.

> Component models and control specifications used in flash startup example; ASPEN built-in procedure in the distillation startup example (PDF)

■ AUTHOR INFORMATION

Corresponding Author

*E-mail: ctchang@mail.ncku.edu.tw.

Notes

The authors declare no competing financial interest.

■ REFERENCES

(1) Lu, Y.-C.; Chen, Z.-L.; Lee, H.-Y. Optimal start-up strategies for a conventional distillation column using simulated annealing. *Chem. Eng. Trans.* **2017**, *61*, 901−906.

(2) Rivas, J. R.; Rudd, D. F. Synthesis of failure-safe operation. *AIChE J.* **1974**, *20*, 320−325.

(3) O'Shima, E. Safety supervision of valve operation. *J. Chem. Eng. Jpn.* **1978**, *11*, 390−395.

(4) Foulkes, N. R.; Walton, M. J.; Andow, P. K.; Galluzzo, M. Computer-aided synthesis of complex pump and valve operations. *Comput. Chem. Eng.* **1988**, *12*, 1035−1044.

(5) Crooks, C. A.; Macchietto, S. A. A combined MILP and logic-based approach to the synthesis of operating proceduresfor batch plants. *Chem. Eng. Commun.* **1992**, *114*, 117−144.

(6) Uthgenannt, J. A. Path and equipment allocation for multiple, concurrent process on networked process plant units. *Comput. Chem. Eng.* **1996**, *20*, 1081−1087.

(7) Yang, S. H.; Tan, L. S.; He, C. H. Automatic verification of safety interlock systems for industrial processes. *J. Loss Prev. Process Ind.* **2001**, *14*, 379−386.

(8) Ferrarini, L.; Piroddi, L. Modular design and implementation of a logic control system for a batch process. *Comput. Chem. Eng.* **2003**, *27*, 983−996.

(9) Lai, J. W.; Chang, C. T.; Hwang, S. H. Petri-net based binary integer programs for automatic synthesis of batch operating procedures. *Ind. Eng. Chem. Res.* **2007**, *46* (9), 2797−2813.

(10) Yeh, M. L.; Chang, C. T. An automata-based approach to synthesis untimed operating procedures in batch chemical processes. *Korean J. Chem. Eng.* **2012**, *29* (5), 583−594.

(11) Yeh, M. L.; Chang, C. T. An automata based method for online synthesis of emergency response procedures in batch processes. *Comput. Chem. Eng.* **2012**, *38*, 151−170.

(12) Li, J. H.; Chang, C. T.; Jiang, D. Systematic generation of cyclic operating procedures basedon timed automata. *Chem. Eng. Res. Des.* **2014**, *92*, 139−155.

(13) Cochard, T.; Gouyon, D.; Ṕetin, J.-F. Safe operation sequences: a generation approach based on iterative refinements and abstractions of timed automata. *IFAC PapersOnLine* **2017**, *50* (1), 6952−6957.

(14) Kang, A.; Chang, C. T. Automata generated test plans for fault diagnosis insequential material- and energy-transfer operations. *Chem. Eng. Sci.* **2014**, *113*, 101−115.

(15) Wang, C. J.; Chen, Y. C.; Feng, S. T.; Chang, C. T. Automata-based operating procedure for abnormal situation management in batch processes. *Comput. Chem. Eng.* **2017**, *97*, 220−241.

(16) Åkesson, K.; Fabian, M.; Malik, R.; Flordal, H. Supremica − An integrated environment for verification, synthesis and simulation of discrete event systems. *Proceedings of the 8th IEEE International Workshop on Discrete Event Systems*; IEEE: 2006; pp 384−385..

(17) Cassandras, C. G.; Lafortune, S. *Introduction to Discrete Event Systems*, 2nd ed.; Springer Science+Business Media: New York, 2008.