Control and Optimization of Batch Processes

Dominique Bonvin

Laboratoire d'Automatique

École Polytechnique Fédérale de Lausanne (EPFL)

CH-1015 Lausanne, Switzerland

June 23, 2013

1 The Batch Chemical Industry

Both continuous and discontinuous operations are found in the chemical industry. In continuous operations, the process is maintained at an economically desirable operating point. However, since the design of a continuous chemical plant requires substantial engineering effort, continuous operation is rarely used for low-volume production. Discontinuous operations can be of the batch or semi-batch type. In batch operations, the reactants are loaded in a vessel and processed without material addition or removal. This operation permits more flexibility than continuous operation by allowing adjustment of the temperature profile and the final time. Even more flexibility is available in semi-batch operations, where reactants are continuously added by adjusting the feedrate profile. We use the term batch process to include semi-batch processes.

The operation of batch processes involves reaction and separation recipes developed in the laboratory. A sequence of operations is performed in a prespecified order in specialized process equipment, yielding a fixed amount of product. The sequence of tasks to be carried out on each piece of equipment, such as heating, cooling, reaction, distillation, crystallization, and drying, is predefined. The desired production volume is then achieved by repeating the processing steps on a predetermined schedule.

The main characteristics of batch process operations include the absence of steady state, the presence of constraints and the repetitive nature. These characteristics bring both challenges and opportunities to the operation of batch processes[1]. The challenges are related to the fact that the available models are often poor and incomplete, especially since models need to represent a wide range of operating conditions. Furthermore, although product quality must be controlled, this variable is not available online but is determined only at run end. Opportunities stem from the fact that industrial chemical processes are often slow, which facilitates larger sampling periods and extensive online computations. In addition, the repetitive nature of batch processes opens the way to run-to-run process improvement[2].

2 Control of Batch Processes

Control of batch processes differs from control of continuous processes in two main ways. First, since batch processes have no steady-state operating point, the setpoint and control signals correspond to time-varying profiles. Second, batch processes are repeated over time and are characterized by two independent variables, the run time t and the run counter k. The independent variable k provides additional degrees of freedom for meeting the control objectives when these objectives do not necessarily have to be completed in a single batch but can be distributed over several successive batches. This situation brings into focus an additional type of output that needs to be controlled, namely, run-end outputs that are available only at the completion of the batch. The most common run-end output is product quality. Consequently, the control of batch processes encompasses four different strategies (Figure 1):

1. *Online control of run-time outputs.* This control approach is similar to that used in continuous processing. However, although some controlled variables, such as temperature in isothermal operation, remain constant, the key process characteristics, such as process gain and time constants, can vary considerably because operation occurs along state trajectories rather than at a steady-state operating

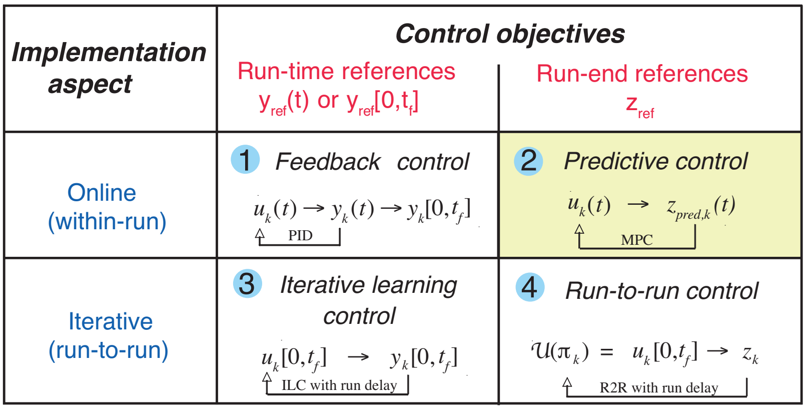

Figure 1: Control strategies for batch processes. The strategies are classified according to the control objectives (horizontal division) and the implementation aspect (vertical division). Each objective can be met either online or iteratively depending on the type of measurements available. u_k represents the input vector for the kth batch, $y_k(t)$ the run-time outputs measured online, and z_k the run-end outputs available at the final time.

point. Hence, adaptation in run time t is needed to handle the expected variations. Feedback control is implemented using PID techniques or more sophisticated alternatives[3].

2. *Online control of run-end outputs.* In this case it is necessary to predict the run-end outputs z based on measurements of the run-time outputs y. Model predictive control (MPC) is well suited to this task[4]. However, the process models available for prediction are often simplified and thus of limited accuracy.

3. *Iterative control of run-time outputs.* The manipulated variable profiles can be generated using iterative learning control (ILC), which exploits information from previous runs[5]. This strategy exhibits the limitations of open-loop control with respect to the current run, in particular, the fact that there is no feedback correction for run-time disturbances. Nevertheless, this scheme is useful for generating a time-varying feedforward input term.

4. *Iterative control of run-end outputs.* In this case the input profiles are parameterized as $u_k[0, t_f] = \mathcal{U}(\pi_k)$ using the input parameters π_k. The batch process is thus seen as a static map between the input

parameters π_k and the run-end outputs z_k[6].

It is also possible to combine online and run-to-run control for both y and z. However, in such a combined scheme, care must be taken so that the online and run-to-run corrective actions do not oppose each other. Stability during run time and convergence in run index must be guaranteed[7].

3 Optimization of Batch Processes

The process variables undergo significant changes during batch operation. Hence, the major objective in batch operations is not to keep the system at an optimal constant setpoint, but rather to determine an input profile that optimizes an objective function expressing the system performance.

3.1 Problem formulation

A typical optimization problem in the context of batch processes is

$$\min_{u_k[0,t_f]} \qquad J_k = \phi\big(x_k(t_f)\big) + \int_0^{t_f} L\big(x_k(t), u_k(t), t\big)\, dt \tag{1}$$

subject to

$$\dot{x}_k(t) = F\big(x_k(t), u_k(t)\big), \quad x_k(0) = x_{k,0} \tag{2}$$

$$S\big(x_k(t), u_k(t)\big) \leq 0, \qquad T\big(x_k(t_f)\big) \leq 0\,, \tag{3}$$

where x represents the state vector, J the scalar cost to be minimized, S the run-time constraints, T the run-end constraints and t_f the final time.

In constrained optimal control problems, the solution often lies on the boundary of the feasible region. Batch processes involve run-time constraints on inputs and states as well as run-end constraints.

3.2 Optimization strategies

As can be seen from the cost objective (1), optimization requires information about the complete run and thus cannot be implemented in real time using only online measurements. Some information regarding the future of the run is needed in the form of either a process model capable of prediction or measurements from previous runs. Accordingly, measurement-based optimization methods can be classified depending on whether or not a process model is used explicitly for implementation, as illustrated in Figure 2 and discussed next:

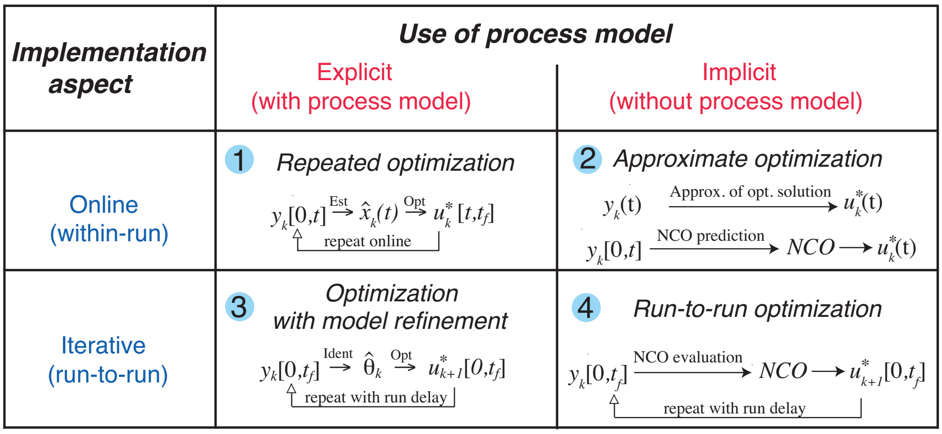

Figure 2: Optimization strategies for batch processes. The strategies are classified according to whether or not a process model is used for implementation (horizontal division). Furthermore, each class can be implemented either online or iteratively (vertical division).

1. *Online explicit optimization.* This approach is similar to model predictive control[4]. Optimization uses a process model explicitly and is repeated whenever a new set of measurements becomes available. This scheme involves two steps, namely, updating the initial conditions for the subsequent optimization (and optionally the parameters of the process model), and numerical optimization based on the updated process model[8]. Since both steps are repeated as measurements become available, the procedure is also referred to as repeated optimization. The weakness of this method is its reliance on the model; if the model is not updated, its accuracy plays a crucial role. However, when the model is updated, there is a

conflict between parameter estimation and optimization since parameter estimation requires persistency of excitation, that is, the inputs must be sufficiently varied to uncover the unknown parameters, a condition that is usually not satisfied when near-optimal inputs are applied.

2. *Online implicit optimization.* In this scenario, measurements are used to update the inputs directly, that is, without the intermediary of a process model. Two classes of techniques can be identified. In the first class, an update law that approximates the optimal solution is sought. For example, a neural network is trained with data corresponding to optimal behavior for various uncertainty realizations[9]. The second class of techniques relies on transforming the optimization problem into a control problem that enforces the necessary conditions of optimality (NCO)[10]. The NCO involve constraints that need to be made active and sensitivities that need to be pushed to zero. Since some of these NCO are evaluated at run time and others at run end, the control problem involves both run-time and run-end outputs. The main issue is the measurement or estimation of the controlled variables, that is, the constraints and sensitivities that constitute the NCO.

3. *Iterative explicit optimization.* The steps followed in run-to-run explicit optimization are the same as in online explicit optimization. However, there is substantially more data available at the end of the run as well as sufficient computational time to refine the model by updating its parameters and, if needed, its structure. Furthermore, data from previous runs can be collected for model update[11]. As with online explicit optimization, this approach suffers from the conflict between estimation and optimization.

4. *Iterative implicit optimization.* In this scenario, the optimization problem is transformed into a control problem, for which the control approaches in the second row of Figure 1 are used to meet the run-time and run-end objectives[6]. The approach, which is conceptually simple, might be experimentally expensive since it relies more on data.

These complementary measurement-based optimization strategies can be combined by implementing some aspects of the optimization online and others on a run-to-run basis. For instance, in explicit schemes, the states can be estimated online, while the model parameters can be estimated on a run-to-run basis. Similarly, in implicit optimization, approximate update laws can be implemented online, leaving the responsibility for satisfying terminal constraints and sensitivities to run-to-run controllers.

4 Future Challenges

Batch processing presents several challenges. Since there is little time for developing appropriate dynamic models, there is a need for improved data-driven control and optimization approaches. These approaches require the availability of online concentration-specific measurements such as chromatographic and spectroscopic sensors, which are not yet readily available in production.

Technically, the main operational difficulty in batch-process improvement lies in the presence of run-end outputs such as final quality, which cannot be measured during the run. Although model-based solutions are available, process models in the batch area tend to be poor. On the other hand, measurement-based optimization for a given batch faces the challenge of having to know about the future to act during the batch. Consequently, the main research push is in the area of measurement-based optimization and the use of data from both the current and previous batches for control and optimization purposes.

References

1. D. Bonvin. Optimal operation of batch reactors - A personal view. *J. Process Contr.*, 8(5–6):355–368, 1998.

2. Bonvin D., B. Srinivasan, and D. Hunkeler. Control and optimization of batch processes: Improvement of process operation in the production of specialty chemicals. *IEEE Contr. Syst. Mag.*, 26(6):34–45, 2006.

3. D.E. Seborg, T.F. Edgar, and D.A. Mellichamp. *Process Dynamics and Control.* John Wiley, New York, 2004.

4. Z.K. Nagy and R.D. Braatz. Robust nonlinear model predictive control of batch processes. *AIChE Journal*, 49(7):1776–1786, 2003.

5. K.L. Moore. *Iterative Learning Control for Deterministic Systems.* Springer-Verlag, Advances in Industrial Control, London, 1993.

6. G. Francois, B. Srinivasan, and D. Bonvin. Use of measurements for enforcing the necessary conditions of optimality in the presence of constraints and uncertainty. *J. Process Contr.*, 15(6):701–712, 2005.

7. B. Srinivasan and D. Bonvin. Controllability and stability of repetitive batch processes. *J. Process Contr.*, 17(3):285–295, 2007.

8. O. Abel, A. Helbig, W. Marquardt, H. Zwick, and T. Daszkowski. Productivity optimization of an industrial semi-batch polymerization reactor under safety constraints. *J. Process Contr.*, 10(4):351–362, 2000.

9. S. Rahman and S. Palanki. State feedback synthesis for on-line optimization in the presence of measurable disturbances. *AIChE J.*, 42:2869–2882, 1996.

10. B. Srinivasan and D. Bonvin. Real-time optimization of batch processes by tracking the necessary conditions of optimality. *Ind. Eng. Chem. Res.*, 46(2):492–504, 2007.

11. A. Rastogi, J. Fotopoulos, C. Georgakis, and H.G. Stenger. The identification of kinetic expressions and the evolutionary optimization of specialty chemical batch reactors using tendency models. *Chem. Eng. Sci.*, 47(9-11):2487–2492, 1992.