동적모사를 위한 공정모델링 도구
최근 연구동향 및 미래의 방향

오 민
공정부, LG 엔지니어링

A Process Modelling Tool for Dynamic Simulation
Recent Progress, Current Status & Future Directions

M. Oh
Process Engineering Department, LG Engineering Co. Ltd.

Introduction

Dynamic simulation is the activity of analysing and predicting the time transient behaviour of the physical process of interest which is usually described in terms of mathematical equations. Dynamic simulation serves an important role from the early stages of process design to plant commissioning and operation. Applications of dynamic simulation include the synthesis and analysis of control structures, startup and shutdown, safety assessments, optimisation and operator training facilities [1]. In order to achieve a certain goal from dynamic simulation, two activities are mainly involved, *e.g.* building a well-posed mathematical model and a solution method to deal with it. However this task is in many cases quite difficult or almost impossible to a non-expert and if not the case very time consuming. Process modelling tools provide a user with the high level declarative language to build mathematical models and support reliable solution method associated with them. The range of what is both desirable and practically feasible in process modelling has been expanding significantly in recent years. This trend is partly due to the realisation of the potential benefits of increased modelling realism and partly to rapid advances in computer hardware and numerical softwares.

This paper reviews some of the significant issues concerned with such modelling tools. This is then followed by the brief description of the background and functionalities of next generation modelling tool, *gPROMS* developed at Imperial College. Finally based on the discussion made here, some suggestions for future development are given.

Scope of Process Modelling & Recent Progress of Modelling Tools

It has long been recognise that a natural mathematical description of the transient behaviour of lumped parameter processes is in terms of mixed systems of ordinary differential and algebraic equations (DAEs). However, a significant number of unit operations in chemical and biochemical processes take place in distributed parameter systems in which properties vary with respect to one or more space dimensions as well as time. Examples include packed bed tubular reactors, packed bed absorption, adsorption and distillation columns *etc.* In other types of unit operations, some of the properties of the material are characterised by probability density functions instead of single scalar values. Examples include crystallisation units [2] and polymerisation reactors, in which the size of the crystals and the length of the polymer chains respectively are described in terms of distribution functions. The form of the latter may also vary with both time and spatial position. In fact, most complex processes typically involve a combination of both distributed and lumped parameter unit operations. However, currently available general-purpose softwares for the modelling and simulation of chemical and biochemical processes

are primarily intended for lumped parameter systems [3]. Several packages, such as *SpeedUp* [4], *DIVA* [5] and *ASCEND* [6] provide modelling languages that allow the transient behaviour of individual unit operations to be described in terms of a mixed set of DAEs. In such modelling packages, distributed processes are currently modelled by manual discretisation of the distributed dimensions, which reduces the mathematical system to a set of DAE with respect to time. This process is both difficult and error-prone, especially when advanced discretisation techniques are to be applied, and a high level of mathematical knowledge and skill is required to perform it in a satisfactory manner.

Although in their simplest form models for unit operations are described in terms of continuous equations, many such models also involve one or more discontinuities. These typically arise from thermodynamic (*e.g.* phase) or flow (*e.g.* from laminar to turbulent regime) transitions, or from irregularities in the geometry of process vessels (*e.g.* overflow pipes or weirs) [7]. Along with the intrinsic discrete characteristics of a processing system, most chemical unit operations experience external actions which lead to an introduction of discrete events. For instance, many computer control actions do indeed correspond to discrete changes in the values of the input variables. Further complication are introduced by the modelling of impulsive external actions, such as the instantaneous introduction of a quantity of a certain reactant in a reactor, or resetting the integral error of a proportional/integral controller to zero. Considering that one of the prime motivations for developing dynamics models and using dynamic simulation is the accurate analysis of the effects of external control actions and disturbances imposed on the physical system, we argue that modelling external forces should be an integral part of modelling activities of a physical system for dynamic simulation.

Mathematical Description of a Dynamic Model

The mathematical model of a physical process in which properties vary with time as well as one or more spatial dimensions is usually described by PDEs. If the independent variables (including both time and spatial position) are denoted by x over the solution domain Ω and its boundary $\delta\Omega$ and the dependent variable ϕ can be written as:

$$\phi = \phi(x)$$

where $\phi \in R^n, x \in R^m$. The partial derivatives are often denoted as follows:

$$(\phi_x)_{ij} = \frac{\partial \phi_i}{\partial x_j}, \qquad (\phi_{xx})_{ijk} = \frac{\partial^2 \phi_i}{\partial x_j \partial x_k}, \qquad etc.$$

As discussed by Marquardt [3], modelling of particulate system involving properties characterised by probability distributions, *e.g.* crystallisation and polymerisation, usually leads to a population balances which invariably introduce integral terms. Overall, the mathematical model of reasonably complicated chemical process can be described in terms of a mixed set of integro-partial differential algebraic equations (IPDAEs) over the bounded domain Ω:

$$f(\phi, \phi_x, \phi_{xx}, \ldots, \int \phi(x)dx, x) = 0, \qquad x \in \Omega$$

and auxiliary conditions over the boundary $\delta\Omega$:

$$g(\phi, \phi_x, \ldots, \int \phi(x)dx, x) = 0, \qquad x \in \delta\Omega$$

The range of boundary conditions encountered in typical process modelling applications is far wider than that usually studied in the mathematical literature. For instance, in addition

to the well-known Dirichlet, Neumann and Robin types of boundary conditions, one encounters cyclic or multipoint conditions that establish relationships between the values of the variables at two or more points in the domain of interest [8]. In some important cases, the boundary of the domain is not known in advance, but has to be determined as part of the solution. Moving boundaries are associated with time dependent problems, and the position of the boundary has to be determined as a function of time. Moving boundaries occur mostly in heat-flow problems with phase changes and in certain diffusion processes .

Current State of the Art of a Modelling System - *gPROMS* system

Considering rapid changes of working environments in process industries and new requirements discussed above, *gPROMS* package has been developed at Imperial College since 1988 [7, 8]. The development of the general purpose modelling tool raises a number of issues. One of them is the need for extensive support for model construction activities. In attempting to satisfy the requirements during modelling building and simulation, the package separate the description of the physical system from the mathematical solution method. The former then is provided by the model developer, while the system takes responsibility of the latter. The following describes the basic building block and the capabilities of the package:

Build blocks: The basic building blocks of the package consist of *MODEL* and *TASK* entities. *MODEL*s describe the physical, chemical and biological laws that govern the intrinsic behaviour of a system, which *TASK*s are description of the external actions and disturbances imposed on it. A third type of entity, the *PROCESS*, is formed by a *TASK* driving a *MODEL* with some additional information, such as the initial conditions and the time variation of the input variables. As a result, a simulation is defined as the execution of this process.

Hierarchical submodel decomposition: Due to the complexity of chemical processes, it is often difficult to handle the whole modelling task simultaneously. It is also of paramount significance to re-use any models which are developed and proven at substantial cost. Considering these issues, hierarchical mechanisms which enable the user to construct the complex model from simpler components are introduced. The basic principle of these mechanism, called hierarchical submodel decomposition, is repeatedly to subdivide the modelling problem of interest until a sufficiently simple level of model is reached. In the chemical industry, such models usually correspond to elementary equipment items for unit operation, (*e.g.* pump, value) or parts thereof (*e.g.* distillation column trays). In *gPROMS* language, the user can construct a complex model as a combination of submodels possibly by streams. Each of these submodels is an instance of another model which may, in turn, have its own submodels. In this way, a hierarchy of arbitrary depth can be created.

Hierarchical subtask decomposition: As discussed above, the capability of handling external forces in the simulation package is one of the most important issue for combined discrete/continuous process simulation. For all but the most trivial simulation, the complexity of these external actions may grow just as rapidly as that of the model of the physical behaviour. With the similarity to hierarchical submodel decomposition, complex tasks are decomposed into lower level tasks until they end up with a set of elementary tasks. This mechanism called hierarchical subtask decomposition allows a user to deal with a complex task on one hand and provides a way to re-use an elementary task on the other.

Capabilities: As a result of substantial amount of efforts, *gPROMS* package can effectively deal with a wide spectrum of simulation problems. The main capabilities can be summerised as follows:

- Modelling and simulation of combined discrete/continuos processes
- Modelling and simulation of combined lumped/distributed processes
- Modelling and simulation of combined deterministic/stochastic processes

Future Directions & Concluding Remarks

Recent progress of modelling tools allows a user to tackle considerable amount of compliecated modelling and simulation problems. Despite the current situations, there are many outstanding problems to be resolved. Among them, this section concentrates on two problems which, we believe, are of more strategic and fundamental nature.

Multi purpose process modelling: Although the discussion made in this paper is mainly devoted to the issues associated with dynamic simulation, well-posed mathematical models can be utilised in many aspects. Those include steady state/dynamic optimisation, parameter estimation, data reconciliation and controllability analysis for control system design. This requirement therefore leads to a conclusion that future modelling tools should support all such activities within a unified frame work of modelling tools. This approach guarantees the re-use of proven mathematical models through simulation and consequently eliminates possible errors while new mathematical models are built according to the applications. However, in order to achieve this goal, two problems are to be resolved. First, a natural high-level language to represent the application problem (*e.g.* dynamic optimisation) must be devised. Secondly, general numerical algorithms and codes for the solution of large-scale problem in the application are essential.

Problem detection and diagnosis: Process modelling tools afford the user considerable flexibility both in model construction and in the specification of degrees of freedom and initial conditions. The possibility therefore exists for the definition of problems which are either badly posed (in the sense that they do not possess a well-defined mathematical solution), or impossible or difficult to solve using the current state of solution techniques. The detection and diagnosis of badly posed problems is already a difficult task for systems of nonlinear algebraic equations and DAEs. For the IPDAE systems, this task is much more complex. Given the lack of the necessary mathematical framework, we probably have to rely mainly on analysis performed at the level of the DAE system resulting from discretisation, checking for instance, the well-posedness of the system and its initial conditions, and the index of the DAE system. An additional complication in this context is that, since the discretisation is carried out automatically, the user is not necessarily aware of the form of the discretised equations. The results of any such analysis would therefore have to be expressed in terms of the original IPDAE system for presentation to the user.

References

1. Wozny, G and Jeromin, L.: *Intl. chem. Engng*, **34**, 159(1994).
2. Pantelides, C.C. and Oh, M.: *J. Powder Technol*, in press(1996).
3. Marquardt, W.:"*Dynamic process simulation - recent progress and future challenges.*" In proceedings *CPC IV*, Texas, USA(1992).
4. Aspen Technology: "*SpeedUp User Manual.*", Cambridge, United Kingdom(1991).
5. Kroner, A., Holl, P., Marquardt, W. and Gilles, E.D.: *Computers chem. Engng*, **14**, 1289(1990).
6. Piela., P.C., Epperly, T.G., Westerberg. K.M. and Westerberg, A.W.: *Computers chem. Engng*, **15**, 53(1991).
7. Pantelides, C.C. and Barton, P.I.: *Computers chem. Engng*, **17S**, 263(1993).
8. Oh, M. and Pantelides, C.C.: *Computers chem. Engng*, in press(1996).