

OBJECT-ORIENTED SIMULATION FOR THE CONTROL OF THE IRIS NUCLEAR POWER PLANT

A. Cammi*, F. Casella**, M. E. Ricotti*, F. Schiavo**, G. D. Storricks†

(* Politecnico di Milano, Dipartimento di Ingegneria Nucleare
(**) Politecnico di Milano, Dipartimento di Elettronica e Informazione
Piazza Leonardo da Vinci, 32
20133 Milano, ITALY

{antonio.cammi, francesco.casella, marco.ricotti, francesco.schiavo}@polimi.it

(†) Westinghouse Electric Company, LLC
Repair, Replacement and Automation Services
1740 Golden Mile Highway
Monroeville, PA, 15214, USA
{storrigd@notes.westinghouse.com}

Abstract: The case study of the dynamic modelling and simulation of the new IRIS power plant control system is presented. After reviewing the base concepts of object-oriented physical modelling, the principles upon which the component libraries are based, and the structure of the plant, the implementation of a system simulator, comprising the physical plant model and the control system model is described. In particular, it is shown how the object-oriented approach allows an efficient management of the different degrees of detail needed throughout the model life-cycle. A discussion on the simulator performance and some representative results are also included. *Copyright © 2005 IFAC*

Keywords: Nuclear power plant control, Object-oriented dynamic modelling, Simulation.

1. INTRODUCTION

Dynamic simulation is a very important tool in the design of power plant control systems, particularly when innovative plants are considered. Simulation can play a role from the initial design stages, when the control strategies and the required instrumentation are evaluated, to the validation of the controller tuning, up to the plant commissioning phase, not to mention personnel training. Dynamic modelling and transient analysis efforts can be very cost-effective, both during design, when it can catch flaws which would result in later costly interventions, and during the commissioning phase, when the savings in terms of reduced down-time can be huge.

Recent advances in object-oriented modelling of dynamical systems, and in particular the development of the Modelica language (Mattson, et al., 1998),

bring new possibilities in this field, allowing the fast development of system simulators which can be tailored to the different needs of the design process, while maximising the re-use of existing information and knowledge.

The case study of the control system for the IRIS PWR nuclear plant is presented here. In Section 2, the relevant features of object-oriented modelling are discussed; Section 3 briefly describes the component libraries used in this project; Section 4 introduces the IRIS reactor concept, highlighting its innovative features; in Section 5, the plant simulator is described, while Section 6 is devoted to a general discussion of the control system architecture. Selected simulation results are presented in Section 7, while Section 8 concludes the discussion.

2. OBJECT-ORIENTED MODELLING OF DYNAMICAL SYSTEMS

The Modelica language was introduced in 1997, as the product of an international cooperative effort to define an object-oriented language for the modelling of generic physical models, described by algebraic and differential equations. The features of the language which are relevant in the context of this paper are summarised here.

A-causal, declarative modelling. Each model is described by a set of algebraic, differential, and event-triggered difference equations; these describe how the modelled object behaves, rather than how the equations are to be numerically solved. The boundary conditions are not necessarily declared a-priori as input or outputs, which is essential to achieve truly object-oriented modelling of physical systems.

Code transparency. The declarative approach allows to write the model code in a way that tightly matches the way equations are written on the paper, which eases the model development, documentation, modification and reuse.

Encapsulation. The models of system components are connected through rigorously defined interfaces or connectors (e.g. fluid connectors with pressure, flow rate, and enthalpy, or heat connectors with temperatures and heat fluxes). Any two components with compatible connectors can be bound together, regardless of their internal details. This feature is essential to re-use models, and to easily replace subsystems with more or less detailed counterparts, without affecting the rest of the system.

Inheritance. Model libraries can be given a hierarchical structure, in which more complex models are obtained from basic models by adding specific variables, equations or even models. It is then possible to factor out the common behaviour of a family of components (e.g. valves, or pumps) in a parent model, and then to define child models which add their specific variables and equations. It is also possible to model objects with replaceable fluid models, by separating component model equations from the fluid model.

Multi-physics modelling. The Modelica language allows modelling of generic dynamical systems. It is then straightforward to combine physical models belonging to different domains with continuous- or discrete-time control systems models.

Reusability. A-causal modelling, encapsulation, and inheritance are a strong incentive towards reuse of modelling knowledge inside simulation projects. At the component level, it is often possible to re-use models provided by standard libraries, while developing a few specific components with ad-hoc (and possibly proprietary) modelling, wherever needed. At the system level, it is possible to easily manage a family of models with different accuracy

and simulation speeds; in most cases, this can be obtained by slight variations of a "reference" system model. This greatly helps to maintain the consistency of models as they evolve throughout the project life-cycle, as modifications and improvements applied to the reference model are automatically inherited by simplified ones.

3. THE MODEL LIBRARIES

The system simulator has been built by combining components from two different libraries, plus some custom-designed control system components.

3.1 The ThermoPower library

The ThermoPower library (Casella and Leva, 2003b) has been developed at the Dipartimento di Elettronica e Informazione as an open-source library for the modelling of thermal power plants at the system level, to support the design and validation of control systems. The library has been developed according to the following guidelines (Leva and Maffezzoni, 2003, Casella and Leva, 2003a):

- Models are derived from first principle equations or from acknowledged empirical correlations.
- The level of detail of the models is flexible, up to the maximum compatible with a system-level simulation.
- The structure of the model interfaces is independent of the modelling assumptions inside each model, to achieve full modularity.
- The library is structured with a limited use of the inheritance mechanism, to maximise the code readability and modifiability.
- The library is open-source software, available at <http://www.elet.polimi.it/upload/casella/thermopower/>.

The modelling principles of the components used in the simulator are summarised here: for more details see (Casella and Leva 2003b), and the library code.

3.1.1 Fluid Flow Models. The model of the fluid flow in a cylindrical conduit is based on 1-dimensional dynamic mass, energy, and momentum conservation equations. The partial differential equations are discretised by using either a finite-volume or a Petrov-Galerkin finite-element method. Flow reversal can be handled, while the pressure drop is computed according to well-established correlations. A homogeneous model is adopted for two-phase flow conditions. Heat transfer coefficients can be computed according to different correlations.

3.1.2 Heat Transfer Models. These models compute the heat flow between interacting 1-dimensional objects (usually, a fluid and a metal wall). The heat transfer coefficient can be assumed either as constant, or supplied by the fluid flow model.

3.1.3 Cylindrical Wall Model. A cylindrical metal wall is modelled by suitably discretising Fourier's equations.

3.1.4 Pressure Drop Models. A turbulent pressure drop, or simple linear pressure drop can be modelled.

3.1.5 Centrifugal pump model. The model is based on the characteristic curves of the pumps, including power consumption, to correctly represent the thermal effects on the fluid.

3.1.6 Header model. The model is based on a dynamic mass balance for the fluid, and dynamic energy balances for the fluid and metal wall.

3.1.7 Valve models. Different valve models are provided, from simplified linearised models, to more accurate models based on industrial standards.

3.1.8 Turbine models. Turbines are modelled by algebraic characteristic equations for the hydraulic behaviour, and by accounting for isentropic efficiency in the energy balance. Turbine models provide thermo-hydraulic connectors for the fluid inlet, as well as mechanical connectors to the electrical power generation equipment.

3.1.9 Power system component models. Idealised models of synchronous generators, power loads, and power grid, describing active power flows and frequency, are provided; they act as boundary conditions in transients involving the power grid.

3.2 The NUKOMP library

The NUKOMP library has been developed at the Dipartimento di Ingegneria Nucleare to provide components which are specific of the IRIS nuclear plant. The modelling principles are summarised here; for more details, see (Cammi et al, 2004).

3.2.1 Pressuriser model. This model accounts for separate mass and energy balances in the liquid and vapour volumes. A simple heater model, as well as steam relief valve and CVCS model are included.

3.2.3. Point Kinetics Neutronic model. The model is based on standard point kinetic dynamic balance equations, describing the evolution of the neutronic population and of the precursor concentration. Reactivity feedback from coolant density, fuel Doppler effect, and rod insertion are accounted for. The dynamic terms can be switched off, to obtain a simplified static model, neglecting the fast dynamics.

3.2.4 Fuel Model. The temperature dynamics is modelled by Fourier's equation, which is discretised radially in 5 zones, and longitudinally in N segments.

3.2.5 Control rods drive model. The rod drive mechanism is modelled according to a simplified model, which describes the step-wise response of the rod position to the speed demand signal. A more detailed, physically-based model will be considered in the future, if needed for some analysis.

3.3. The Control Library

Digital PI/PID controller models, complete with tracking, anti-windup, and bumpless transfer features, have been developed, as well as a generic finite-state machine to describe the switching between different control modes by the supervisory controller.

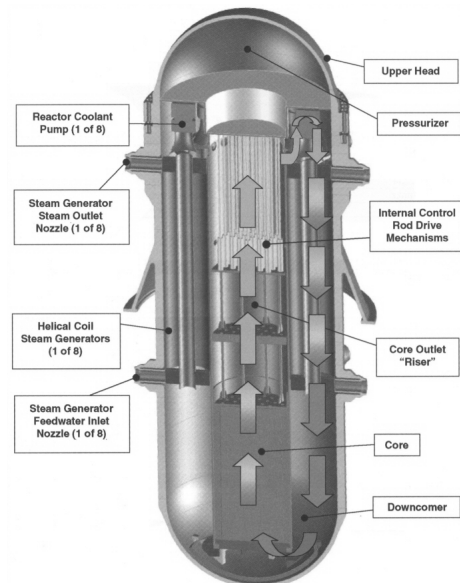


Fig. 1. The IRIS reactor

4. THE IRIS REACTOR

The IRIS project (Carelli et al., 2003) involves 21 organizations from 10 countries and refers to the design of an innovative, light water reactor with a modular, integral primary system configuration. The reactor pressure vessel houses the nuclear fuel, control rods and control rods drive mechanisms, but also all the major reactor coolant system components, including the coolant pumps, the steam generators and the pressurizer (Fig.1).

IRIS is basically a PWR (Pressurised Water Reactor): in the primary loop, liquid water is heated by the nuclear fuel rods in the core, and is then sent by the pumps to the primary side of heat exchanger; the secondary loop actually generates steam which is sent to turbines to produce power. Compared to conventional PWR plants, however, IRIS has a set of distinctive features, which directly affect the control system design:

- the integral configuration requires a large water inventory in the primary loop, whose residence time is much greater than usual;
- a helicoidal once-through steam generator is employed on the secondary side, which has a very short residence time, compared to the more widespread U-tube recirculating steam generators;
- sprayers are not available to reduce the pressure in the primary loop during fast transients.

The control strategy must take these facts into account, and a dynamic simulation tool is essential to ensure that the control objectives can be achieved.

5. MODELLING THE IRIS REACTOR FOR CONTROL SYSTEM PURPOSES

5.1 Control-Oriented Simulation of Nuclear Plants

Contrary to conventional power plants, where it is often deemed inessential, dynamic modelling is of paramount importance in nuclear plant engineering:

a plant can be actually built and operated only after an extensive licensing procedure, where the safety of the plant during accidental transients is demonstrated. Reliable, accurate and certified simulation packages are well established in this field (e.g. RELAP), and very detailed dynamic models of all nuclear plants, including IRIS, are always developed for new plants (Grgic et al., 2003). However, these simulators are unnecessarily over-detailed for control system studies, require very long times for the simulation of a single transient (hours, or even days), and offer little flexibility to integrate the plant model with the control system model, and other boundary conditions like a simplified power generator and grid.

Conversely, light and flexible simulation models are needed, which will gradually evolve during the control system design phases, and will eventually be required to perform up to ten thousands simulation runs in sensitivity studies for a new plant design. The aims of object-oriented modelling are thus:

- to obtain a reactor model which is accurate enough to reproduce the control-relevant dynamics during normal operation, validated against the more accurate, safety-oriented models;
- to integrate the reactor model with the control system model, and with other relevant sub-systems at its boundary;
- to provide a whole range of degrees of detail, so that the correct balance between accuracy and simulation speed can always be struck, depending on the specific simulation needs.

This task has been successfully accomplished by adopting the Modelica framework.

5.2 Structure of the IRIS simulator

The system model results from the aggregation of 4 different subsystems:

- the Control System (CS);
- the Supervisory System (SS);
- the Nuclear Steam Supply System (NSSS), including the core, primary loop, and once-through evaporators;

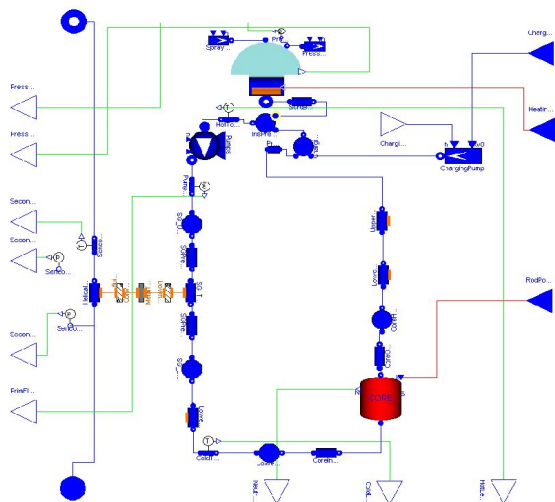


Fig. 2. Diagram of the NSSS simulator

- the Turbine, Generator and Feedwater System (TGFS), including turbine admission valves, steam turbines, electric generator and grid connection, feedwater pumps and condensate train.

Currently, the feedwater system is modelled by an ideal flow source with prescribed temperature, and the turbine model is represented by a simplified linear model. Since the plant efficiency and the turbine response heavily depend on the regenerative system, using bled steam to preheat the feedwater, a more detailed model of the TGFS is planned for the near future.

The CS and SS models will be discussed in more detail in the next section. It is worth mentioning that also the CS will evolve during the project lifetime, as more functionalities and details are progressively added.

The NSSS model (Fig. 2) is now briefly reviewed. Starting from the top of the diagram, counterclockwise, the primary loop is composed by the pressuriser model, a mixing volume, the pumps, a header, the primary side of the steam generator, the bottom plenum, the core, the risers, the charging system, and an upper plenum.

The steam generator is completed by the secondary side fluid model, heat transfer models, and helicoidal tube models (left of the diagram). The core, in turn, contains a standard fluid model for the moderator, connected to the rod and point kinetics models.

The NSSS model is connected to the outside by input connectors (corresponding actuator signals going to the actuators), output connectors (corresponding to the sensor signals), and two thermo-hydraulic connectors (bottom left, top left) corresponding to the feedwater inlet and the steam outlet.

5.3 Model flexibility

As already anticipated in Section 2.6, by slightly changing the model code it is possible to vary the degree of detail of the reference model; for specific simulations, the accuracy can be only marginally affected, while the simulation speed can improve tenfold or even more. Some examples are now given.

5.3.1 Fluid models. The standard water/steam model employed in the simulator is the very accurate IAPWS 97 model. However, the thermodynamic conditions in the primary loop vary in a quite narrow range, so that it is possible to use approximated (e.g. linear or table-based) water models, where temperature and density are only a function of the temperature. Besides being much faster to compute, the resulting system lack the fast pressure states originated by the small compressibility and small hydraulic resistances along the primary loop. Note that the code of the models is unaffected: for example, the dynamic model of a plenum is always

```
fluid.p = p; // Set the fluid state
fluid.h = h; // Set the fluid state
M = fluid.d*v; // Fluid mass
```

```

E = fluid.u*M // Fluid internal energy
// Mass & energy dynamic balance equations
der(M) = inlet.w + outlet.w;
der(E) = inlet.w*h_in+outlet.w*h_out;

```

regardless of the fluid model. The symbolic manipulation algorithm of the Modelica compiler can determine whether the pressure is a state variable or not, depending on the structure of the `fluid` object.

5.3.2. Steam generator model. The number of nodes in the steam generator can be reduced from 15 to 4 or 5; besides that, it is possible to replace the fluid flow model on the secondary side with a simplified, moving boundary model, having the same interface.

5.3.3. Heat transfer coefficients. When working around a certain operating point, Dittus-Boelter and Chen's correlations can be substituted by simpler ad-hoc interpolations, or even by constant coefficients.

5.4. Simulator performance

Open loop plant simulations (e.g. step responses) with the full model can be performed efficiently using variable stepsize algorithms such as DASSL; however, the digital control systems may force one to employ fixed time step algorithms instead.

The control system has to be modelled as a digital algorithm (i.e., a discrete-time sampled system), in order to be able to model the bumpless switching between the different control modes, and, later, to include the actual control code in the simulation model to check its correctness. As a consequence, events and control signal discontinuities happen every 100 ms in the system model; this makes variable stepsize algorithms very inefficient, because the integration time steps are frequently shortened to accurately reproduce the fast dynamics excited by the steps, which is inessential to the control system performance. The stepwise nature of the control rod mechanism, which must be included in the simulation model to correctly estimate the safety margins of some controlled variables, also causes similar effects.

In order to be able to run batches of thousands of transients to thoroughly validate the control system, explicit fixed-step algorithms have been tried; this in turn required to avoid including any fast dynamics with a time constant smaller than 50 ms, i.e. using an incompressible fluid model in the primary loop, neglecting the dynamic terms in the momentum balances, and all the point kinetics dynamics. By also using table-based fluid models and simplified heat transfer coefficient interpolations, simulations were run more than 10 times faster than the real time on a standard PC. These simulations can easily be checked against the results of the full-fledged model.

6. THE CONTROL SYSTEM: ARCHITECTURE AND ISSUES

Broadly speaking, the task of the control system is to bring the plant from the so-called hot-zero-power state (i.e. turbine tripped, hot primary loop, reactor critical but with negligible power output) to full

power and back, and to follow the desired load profiles, while ensuring that no controlled variable crosses the safety threshold, thus avoiding the need for the protection system to shut down the plant.

On one hand, nuclear plants are generally designed to be stable and easily controllable, and consolidated engineering experience with similar plants gives the basic guidelines on how to control them. On the other hand, the specific features of the IRIS plant make it necessary to check that no problems arise on some particular transient, e.g. during the startup sequence.

The control system architecture is basically a coordinated control scheme. A supervisory controller switches between different control modes (i.e. a full power mode, plus other modes which are needed to start up the plant, or to manage special conditions such as a turbine trip or islanding), and sends appropriate setpoints and feedforward terms to five lower level controllers. These in turn act on the five plant control variables: control rod insertion speed, feedwater flow, turbine admission valves, steam dump valves, and heater power in the pressuriser.

The actual measured and controlled variables of each low-level controller depend on the control modes: for example, in full power mode, the rods control the average temperature in the primary loop to a specified value (a function of the power load), the turbine valves control the boiler pressure (with the steam dump valves as backup), the feedwater flow controls the turbine power, and the heaters control the primary loop pressure. In other control modes, the variable pairings and the controlled variables are different; bumpless transfer must be guaranteed when switching modes. The lower level controller are built by a combination of PI/PID blocks, feedforward terms (possibly with some dynamic compensation), variable gains. Dead bands are used in the rod controller, to avoid persistent oscillations at steady-state, due to the inherent quantisation of the actuator.

Currently, only the full power control mode has been implemented, to get a preliminary assessment of the control system performance. In the near future it is planned to include other control modes, for which a conceptual design has already been carried out, and to check the transitions between them in during plant startup, shutdown, turbine trip, and islanding.

7. SIMULATION RESULTS

Selected simulation results are now presented. Fig. 3 shows the response to a 10% decrease of the power setpoint from full power, and to a subsequent step increase back to full power (all the variables are normalised to the full power value). Although this transient will never be performed in practice, it gives a good insight into the control system performance. The system response is generally well-damped and with acceptable overshoot. The electrical power response shows some nonminimum phase behaviour, partly due to over-simplified modelling of the turbine system. Fast, small-amplitude oscillations due to the rod stepwise control can also be observed.

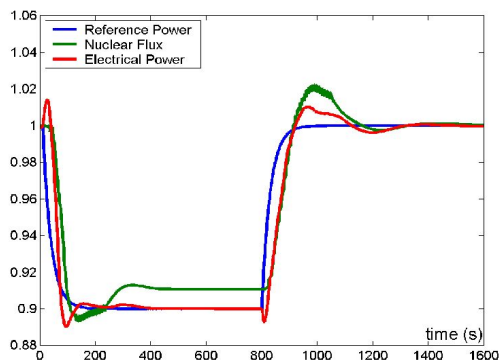


Fig. 3. Power transients (per unit)

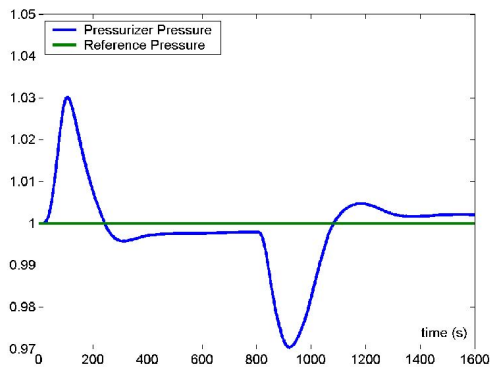


Fig. 4. Primary pressure transient (per unit)

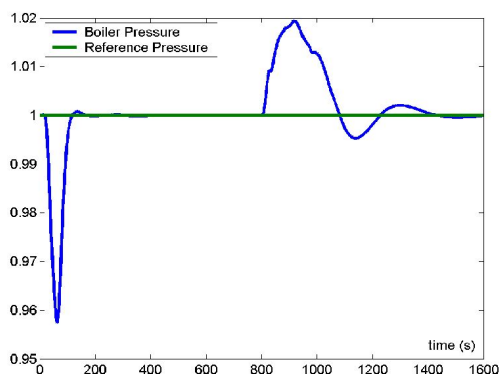


Fig. 5. Boiler outlet pressure transient (per unit)

Fig. 4 shows the corresponding pressuriser pressure transient, while Fig. 5 shows the pressure transient at the boiler output. Both transients remain within acceptable limits.

8. CONCLUSIONS

The application of object-oriented modelling to the study of the control system of the innovative IRIS nuclear has been demonstrated. Compared to the ACSL-language-based simulators previously used by Westinghouse, the resulting simulation tool was much more accurate, required a smaller development effort, while providing adequate simulation speed. The plant model can be used for conventional, "routine" engineering activities, as well as a support

tool for more advanced studies, e.g. model-based supervisory control. Since the IRIS project is supported by a wide, multinational consortium, the availability of a reference dynamic model which is much lighter and offers a greater flexibility than the high-accuracy RELAP code is a definite benefit.

Finally, the object-oriented approach strongly encourages the re-use of modelling and control knowledge, for the following reasons:

- the model code is highly structured, so that it can be better understood and re-used by people different from the original developers;
- the description of the physical models is much closer to the physical equations and to the physical structure of the plant than that of more traditional simulation approaches;
- it is possible to build a "reference" model, evolving throughout the project lifetime, from which simpler and faster models are obtained through a few modifications, thus facilitating consistent maintenance and documentation;
- most of the modelling effort could be easily re-used for similar projects (e.g. any PWR plant control system), at a fraction of the effort of developing this first instance.

REFERENCES

- Cammi, A., F. Casella, M.E. Ricotti and F. Schiavo (2004). Object-oriented modelling for integral nuclear reactors dynamic simulation. In: *Proc. International Conference on Integrated Modeling & Analysis in Applied Control & Automation*, Genoa, Italy.
- Carelli, M.D. et al. The Design and Safety Features of the IRIS Reactor. In: *Proc. ICONE11 Int. Conf. On Nuclear Engineering, paper 36564*, Tokyo, Japan.
- Casella, F. and A. Leva (2003a): Modelling of Distributed Thermo-Hydraulic Processes using Modelica. In: *Proceedings of the MathMod Conference*, Vienna, Austria, 2003. To appear in *Mathematical and Computer Modelling of Dynamical Systems*.
- Casella, F. and A. Leva (2003b). Modelica open library for power plant simulation: design and experimental validation. In: *Proc. Modelica Conference 2003*, Linköping, Sweden. http://www.modelica.org/Conference2003/papers/h08_Leva.pdf
- Grgic, D., T. Bajs and L. Oriani (2003). Development of RELAP5 Nodalization for IRIS Non-LOCA Transient Analyses, *American Nuclear Society Topical Meeting in Mathematics & Computations (M&C)*, Gatlinburg, USA.
- Leva, A. and C. Maffezzoni (2003). Modelling of Power Plants, in D. Flynn (Ed.), *Thermal Power Plant Simulation and Control*, IEE, London, pp. 17-60.
- Mattsson, S.E., H. Elmqvist and M. Otter (1998). Physical system modeling with Modelica, *Control Engineering Practice*, 6, pp. 501-510.