Extension of Estelle formalism for performance evaluation: SSCOP protocol performance study for ATM high speed network *

Delphine GAZAL **
Yves RAYNAUD **

Abstract

This article presents an approach used to obtain performance measures from an iso-standardized formal description technique, Estelle. The underlying aim of this approach is not to modify Estelle syntax so as to maintain the advantages provided by conformance to the iso standard. To achieve this, annotations relating to the quantitative aspect have been added to the specification. This annotated specification is then translated into a simulation language, Modsim, to achieve a simulation capable of supplying performance results. At the end of the paper, this approach is illustrated by evaluating the sscop protocol of the AAL layer, the results obtained being in compliance with those obtained using other evaluation methods.

Key words : Formal description technique, Performance evaluation, Transmission protocol, High rate, ATM, ATM adaptation layer, Simulation language.

I. Introduction.

The increasing complexity of cooperative systems in general, and that of real time and embedded distributed systems requires from the designers to validate their systems with respect to both functional and non-functional requirements. In order to avoid costly redesigns and to reduce the complexity of the testing phase, the desi-

* This study has been conducted inside the CNERT-CNRS Cesame project on the design of high speed multimedia cooperative distributed systems, under grant 92 1B 178 from CNERT France Télécom.
** Laboratoire IRIT/ES, Université Paul Sabatier, 118, route de Narbonne, F-31062 Toulouse Cedex, France.
designer should try quantitative and qualitative validations. Concerning the qualitative validation, we propose the use of a formal description technique which allows an unambiguous and precise description of a system. On the other hand, quantitative evaluation is based on performance modelling.

Up to now, the architectural design and the performance modelling activities are disconnected but the latter might influence the design. Our purpose is to provide the designer with a methodology of tackling the complete validation of its system before the implementation phase.

Furthermore, all the work held during the design activity with the construction of the functional model by means of a formal description technique should be capitalised to optimise the construction of performance models by means of a simulation language. After the functional validation of the design model the designer can add performance aspects and then derive a simulation model. So, this brings out two advantages: the first one is that unambiguous and validated system’s descriptions reduce the risk to build inappropriate performance models. The second advantage is the use of well adapted tools: a specific tool for specification and verification and a specific, powerful simulation tool for performance evaluation.

The formal description technique we propose to use is Estelle [1], an ISO standardised formal description technique generally used for specifying protocols. However, it can be used to describe other types of systems. Well defined and validated Estelle specifications must evolve to a realistic implementation under performance constraints. However, Estelle doesn’t care about neither time nor probability means to perform performance evaluation. The idea is then to annotate an Estelle design with quantitative values in order to run the derived simulation model. With these annotations, the Estelle semantics remain unchanged avoiding standardised language extensions and therefore new tools development.

The simulation language we use is Modsim [2], an object oriented simulation language, used to obtain system performance measures. In order to connect these activities, a set of rules is defined that translates the annotated Estelle specification into the Modsim language whose object orientation squares with many Estelle concepts.

II. THE ESTELLE MODEL

II.1. Description.

Estelle is a formal description technique standardized by the ISO for the specification of protocols [1]. Its role is to describe the logic of the component parts of a system running a communications protocol. The main aim is to ensure that the protocol specification is correct so that it can be successfully implemented. It can be used to design any type of communications system.

The formal nature of Estelle specifications makes it possible to apply semi-automatic methods for:

- the validation of specifications;
- the implementation process;
- systematic testing of implementation results.

Estelle is based on interconnected finite-state machines. It also incorporates input-output parameters, data types, procedures and functions, and local variables in addition to the state variable that is a feature of all finite-state machines.

Estelle describes a system as a hierarchy of modules able to communicate with each other. These modules send messages (interactions) to each other via interaction points. The interaction points of 2 modules wishing to communicate are linked via a two-way channel that defines all authorized types of interaction. Each interaction point is associated with a fifo queue where interactions received are held.

The instruction to transmit an interaction is Output $lp$.int, where $lp$ is the interaction point and int the interaction itself.

The behaviour of each module is represented by a set of transitions. Attributes attached to each module define the degree of parallelism between them as well as rules for the nesting of modules in the hierarchy. These attributes are SystemActivity, SystemProcess, Activity and Process.

The structure of a module can be divided into sub-modules. We then obtain a parent-child hierarchy of modules under one condition: System modules (attribute SystemProcess or SystemActivity) are always highest up in the parent-child hierarchy.

The Estelle specification is therefore able to describe an architecture represented by a set of trees making up the module hierarchies, where the root of the tree has to be a module of type System. The process for executing the specification breaks down into two phases: a system evaluation phase and an execution phase. A process of this type is initialized for each System in the Estelle specification and repeats while no blockage is encountered. The system evaluation phase consists in:

- testing the executable transitions among the System module transitions and its descendants;
- selecting the transition(s) to be executed.

The execution of a transition is subjected to a set of rules. These rules are determined by the hierarchy, the attribute attached to each module, the scheduling priority for executing transitions of the parent with respect to its children, and the execution clause attached to each transition. Whatever the type of module, only one of its transitions may be executed at a given instant.

In the case of a SystemProcess or Process module, its children behave in a parallel fashion, i.e. if more than one child has an executable transition, they are all