Graphical User Interface Development for the MARS Code

J.-J. Jeong*, M. Hwang, Y. J. Lee, K. D. Kim, B. D. Chung
Korea Atomic Energy Research Institute (KAERI)
Dukjin 150, Yuseong, 305-600 Daejeon, South Korea

KAERI has developed the best-estimate thermal-hydraulic system code MARS using the RELAP5/MOD3 and COBRA-TF codes. To exploit the excellent features of the two codes, we consolidated the two codes. Then, to improve the readability, maintainability, and portability of the consolidated code, all the subroutines were completely restructured by employing a modular data structure. At present, a major part of the MARS code development program is underway to improve the existing capabilities. The code couplings with three-dimensional neutron kinetics, containment analysis, and transient critical heat flux calculations have also been carried out. At the same time, graphical user interface (GUI) tools have been developed for user friendliness.

This paper presents the main features of the MARS GUI. The primary objective of the GUI development was to provide a valuable aid for all levels of MARS users in their output interpretation and interactive controls. Especially, an interactive control function was designed to allow operator actions during simulation so that users can utilize the MARS code like conventional nuclear plant analyzers (NPAs).

KEYWORDS: Best-estimate System Codes, MARS, GUI, ViSA, Nuclear Plant Analyzer

I. Introduction

KAERI has developed the best-estimate thermal-hydraulic system code MARS. The backbones of the MARS code are the RELAP5/MOD3 and COBRA-TF codes. The RELAP5 code is a versatile and robust system code based on a one-dimensional two-fluid model for two-phase flows. The COBRA-TF code employs a three-dimensional two-fluid, three-field model. To exploit the excellent features of the two codes, we consolidated the two codes into a single code. As a result, the COBRA-TF code functions as a three-dimensional thermal-hydraulic component in the MARS code.

The second step of the MARS development was the restructuring of the consolidated code. To improve the readability, maintainability, and portability of the consolidated code, all the routines were converted into a standard Fortran 90, and then were restructured by employing modular data structures based on the “derived type variables” and “dynamic memory allocation” features. The resulting source programs are now easy to understand and modify. The first two steps were completed in 1998.

The third step, the major part of the MARS code development program currently in progress, aims at the improvement of the existing capabilities. The code couplings with three-dimensional neutron kinetics, containment analysis, and transient critical heat flux calculations have also been carried out. At the same time, graphical user interface (GUI) tools have been developed for user friendliness.

This paper presents the main features of the MARS GUI. The basis of the MARS GUI was originally developed for RETRAN users. RETRAN has been used in various plant support activities such as licensing calculations for plant design changes, emergency operation procedure (EOP) validations, and training in Korea. The use of RETRAN, however, has been limited because of the difficulties involved in its usage. To assist ordinary users in their input preparations and output processing, Kim et al. developed a GUI-based RETRAN running environment, which was later named ViSA (Visual System Analysis). ViSA allowed for a more easy and efficient use of RETRAN. This motivated the MARS GUI development and, thereafter, the ViSA program was extended to MARS.

At present, ViSA can be used for both RETRAN and MARS with different interface routines. The original version of ViSA has the pre-processor function for input generation/modification as well as output processing and interactive control functions. However, the former was removed (changed into a separate program) for simplicity and the interactive control function has been further improved in recent versions. It is noted that the ViSA program is also being developed for the three-dimensional thermal-hydraulics module of MARS, but the relevant descriptions are not given in this paper.

As a target platform of MARS, a personal computer (PC; Windows 98, 2000, XP, and NT) was selected because of its high accessibility and portability. Thus, the ViSA program was also developed on Windows platforms.

Similar efforts for user interface improvement have been made, such as SNAP and PEGASYS. Compared with these GUI tools, the MARS GUI is easier-to-use and more efficient for enhancing understandings of the transient phenomena. Especially, the interactive control function of ViSA allows operator actions during simulation so that users can utilize the MARS code like conventional nuclear plant
analyzers (NPAs).\textsuperscript{11,12}

II. ViSA Structure and Programming

The ViSA is developed to provide the code (MARS and RETRAN) users with efficient output processing and interactive control functions. The key features of ViSA can be summarized as follows:

- The current MARS (and RETRAN) input features are maintained so that the existing input data can be fully utilized.
- Visual displays can be easily created by drag-and-drop of elementary graphic display units, such as digital meters, level gauges, dial meters, LED indicators, X-Y plots, etc.
- Interactive control functions are provided so that users can perform operator actions during the calculation.
- ViSA can run by itself without any additional graphics software on personal computers.
- The ViSA program is generalized so that it can be extended to other codes and/or time-dependent analysis tools.

Figure 1 shows a schematic of the ViSA package. It consists of three functional modules; (i) project, (ii) visual analyzer for output processing, and (iii) interactive control. Each of the functional modules is introduced in the next section. ViSA loads the MARS (or RETRAN) code as a dynamic link library (DLL). A code-specific interface routine connects the code with the visual analyzer and interactive control module.

ViSA takes three input data: a normal MARS input deck and two graphic files containing system mimic and nodalization diagrams produced by any drawing software. The latter is not obligatory and is used for the dynamic display of the system’s operation status. It is noted that MARS can use the RELAP5 input data without any changes. However, for interactive control functions, additional input cards are needed (See the next section).

The ViSA package is programmed using DELPHI 5.\textsuperscript{13} DELPHI is an object-oriented visual programming environment for Pascal language. Each component was created in modular type with a well-defined interface and then the developed components were simply assembled later. This component-based approach could reduce the complexity of the ViSA design. The multi-thread technique was also used for CPU sharing between MARS and ViSA. The technique ensures a prompt response to a user’s request.

![Fig. 1 The ViSA package structure](image-url)
III. Main Features of ViSA

Figure 2 shows the main window of ViSA. It consists of a main tool bar and five tab sheets. User controllable pop up windows including “trend windows” are also available. Users can control the execution of MARS through the main tool bar, where buttons for run, stop, pause, resume, and replay are installed.

The tab sheets include the “project”, “calculation data”, “system mimic”, “nodalization” and “trip messages / interactive control”. In Fig. 2, “project” tab sheet is shown. The function of the “project” module is to control and save the MARS execution environment, which includes the selection of input data, output unit, mimic diagram, and nodalization. Other tab sheets with four pop up windows are used for a visual analyzer and interactive control.

1. Visual Analyzer Module

MARS (or RETRAN) produces a large amount of text-based output “after” transient simulation. From this kind of text-based output, it is not easy to obtain a clear understanding of the physical phenomena during the transient. ViSA provides various graphical displays “during” and “after” the transient simulation.

To provide a visual analyzer function, the tab sheets for “calculation data”, “system mimic”, “nodalization”, “trip messages / interactive control” are designed. Trend curves for user-selected variables are given by pop up windows.

“Calculation Data” Tab Sheet

Figure 3 shows the “calculation data” tab sheet, which is designed to show the text-type output for minor edit variables, volume data, and junction data. Users can select any output variable from the list box (tree-view type) in the left column of Fig. 3. The output variables are updated at user-specified time intervals.

“System Mimic” Tab Sheet

The system mimic tab sheet shows the major output variables through user-selected indicators as shown in Fig. 4. The tab sheet is designed to provide an overview of the system transient. ViSA allows easy construction of this tab sheet so that a user- or problem-specific system mimic can be made. The use of this tab sheet is optional.

The background image that shows the schematic of the simulated system is generated by the user using other graphic software, saved as a bit map file, and then imported to the “system mimic” tab sheet. ViSA provides various types of indicators, such as digital meters, level gauges, dial meters, LED indicator, etc. User can add these indicators into the system mimic by drag-and-drop operations. The indicators are mapped with specific output variables that are listed in a pop up window for “variable selection”. This feature allows for a user-specific “system mimic” display.
Fig. 3  “Calculation Data” tab sheet

Fig. 4  “System mimic” tab sheet
**“Nodalization” Tab Sheet**

The nodalization tab sheet is designed to show the void fraction or temperature distributions of the whole volumes in a color spectrum (See Fig. 5). Moving the cursor into a volume of the nodalization, exact value of the void fraction or temperature in the volume is displayed in the status bar.

To construct this tab sheet, a nodalization diagram should be prepared, saved as a bit map file, and then imported to this tab sheet. Then, each of the hydrodynamic volumes has to be mapped with a corresponding volume. A list of volumes is given in a pop up window upon the user’s request, which is transferred from the MARS input processing routine. The use of this tab sheet is optional.

**“Trend” Pop Up Window**

The trend windows show on-line time plots for user-selected variables. The trend windows can be activated in two ways.

First, the user can specify variables in the code input data. In this case, the trend windows for the specified variables are automatically generated. Second, the user can interactively generate the windows by clicking the right button of the mouse on the “system mimic” or “nodalization” tab sheet. The seventh button in the main tool bar also activates the trend window. In this case, a tree-view list box displays minor edit variables and major volume and junction data in a pop up window. Then, the user can select the variables that he wants to see (See the left figure in Fig. 6). Multiple variables can be drawn in a trend window, and the number of trend windows is not limited. The scales of X- and Y-axis are automatically given, but the user can specify them. “Print”, “zoom”, and “save” functions are also available (See the right figure in Fig. 6).

**“Trip Messages / Interactive Control” Tab Sheet**

The functions of this tab sheet are (i) to display trip messages and (ii) to perform the interactive control function. The latter will be described in the next section.

The left half of this tab sheet (see Fig. 7) shows trip messages. Whenever a change occurs in the status of the trip cards, the trip time, condition, and description are displayed. This feature is useful for understanding the evolution of the transient and for tracking the causes of discontinuous change during the transient. For user’s convenience, two different colors are used for the trip message in order to distinguish between the normal trip and reset trip.
Fig. 6  An example of “Trend” window

Fig. 7  “Trip messages / Interactive Control” tab sheet
2. Interactive Control Module

The interactive control function of ViSA allows for operator actions so that users can utilize the MARS code like conventional NPAs. For evaluating the emergency operation procedure of a power plant, a number of transient simulations should be performed with different operator actions, which depend on the transient progress. Input preparations for these calculations are very complicated and time consuming. If operator actions can be conducted during the code simulations, these difficulties may be resolved. In this regard, the interactive control function was designed.

The operator actions are divided into four functions in ViSA; that is, manual control for trip (on/off), valve area, mass flow rate, and heater power. Controls for pump speed and boron concentration are also being considered. Using these functions, most operator actions can be simulated.

To design these functions, the existing functions in MARS are modified:
- For on/off control, the “trip” component is modified.
- For valve area control, the “servo valve” component is modified so that the normalized valve area can be manually changed.
- For mass flow control, the “time dependent junction” component is modified.
- For heater power control, the “general table” component for power is modified.

These components are modified so that user’s actions can override the code input. Therefore, to use the interactive control function, relevant control input cards should be prepared first and, then, additional input cards are needed to specify the interactive controls.

An example of interactive input cards are given below:

```
*interactive control input (Card 801-999)

801  trip  531 "Manual reactor trip"
802  trip  540 "Manual turbine trip"

806  vlvarea  283 "PRZ spray valve 1"
807  vlvarea  289 "PRZ spray valve 2"
808  vlvarea  951 "PRZ PORV 1"
809  vlvarea  953 "PRZ PORV 2"

811  trip  511 "RCP 1 trip"
812  trip  512 "RCP 2 trip"
813  trip  513 "RCP 3 trip"

818  mflowj  393 "Charging flow"
819  mflowf  597 "Letdown flow"

821  mflowf  375 "LPSI flow, loop 1"
822  mflowf  377 "HPSI flow, loop 1"
823  mflowf  475 "LPSI flow, loop 2"
824  mflowf  477 "HPSI flow, loop 2"
825  mflowf  575 "LPSI flow, loop 3"
826  mflowf  577 "HPSI flow, loop 3"
```

In the above input data, the number 531 in card 801 is a trip card number, the number 283 in card 806 is a servo valve component number, and the number 393 in card 818 is a time-dependent junction number. The “interactive control” tab sheet for these input is shown in the right half of Fig. 7. The tab sheet displays the list of interactive controls, which includes the description, current status, control mode (automatic or interactive) check box, and a toggle switch or edit boxes.

Through the tab sheet, the user can select operator actions. In the case of trip control, a toggle switch is used. In other cases, target value and rate can be entered through the edit boxes. For example, let’s consider the situation that a pressurizer PORV (power-operated relief valve) is closed and the user wants to fully open the valve in 5 seconds. In this case, the target value and the rate are entered as 100 % and 20 %/s, respectively. This situation is given in Fig. 7. When the user clicks the “OK” button, the interactive user input is transferred to the system code.

IV. Concluding Remarks

KAERI has developed the best-estimate thermal-hydraulic system code MARS. To improve the user friendliness of MARS, the ViSA package, which was initially designed for the RETRAN code, has been extended to the MARS code.

The major advantages of ViSA can be summarized as follows:
- Various graphical representations of the output are available.
- Graphical display systems are easily constructed.
- The interactive control function is very simple and easy-to-use.
- Because of the component-based design and general interface structure, VISA is easily portable to any other system code or transient analysis code.

The first version of MARS ViSA has been released and user’s feedback is currently being taken into account for further improvement.

Acknowledgment

This work has been performed as a part of the Nuclear R&D Program supported by Ministry of Science and Technology (MOST) of the Republic of Korea.

References


