The Tactical Radar and InfraRed Engagement Modelling Environment (TRIREME)

Byron Hatfield
Electronic Warfare & Radar Division, Defence Science and Technology Organisation
Byron.Hatfield@dsto.defence.gov.au

Simon Reynolds
Avalon Systems
Simon.Reynolds@dsto.defence.gov.au

Belinda J. Hermans
Electronic Warfare & Radar Division, Defence Science and Technology Organisation
Belinda.Hermans@dsto.defence.gov.au

Abstract. The Tactical Radar and InfraRed Engagement Modelling Environment (TRIREME) is a flexible simulation architecture for the building and running of military engagement scenarios in Simulink. TRIREME consists of a set of rules (e.g. interface specifications) for developing engagement models for insertion into the environment as well as a toolbox of custom Simulink blocks. These rules and tools greatly simplify the building and modifying of engagement simulations, eliminating the large amount of work normally required to manually change signal and bus configurations. A library of system models can be developed that can readily be added to (or deleted from) a TRIREME configuration and the user can vary the number of objects in a simulation without having to make any changes to the configuration itself or the models involved. TRIREME currently allows for the simulation of a variety of one-on-few scenarios between a guided weapon and a number of aircraft employing a range of electronic countermeasures and other electronic warfare systems, but given the flexible nature of the architecture this could be readily extended to include land and maritime environments as well as multiple threats and a variety of platforms.

1. INTRODUCTION

The ability to assess the effectiveness of electronic countermeasures and/or countermeasure techniques for the protection of Australian Defence Force (ADF) assets against various threat systems through the use of simulation is becoming increasingly important, particularly where field trials are not an option due to resource limitations or equipment availability. In those cases where trials can be carried out, engagement modelling and simulation also provides an increased capability for pre- and post-trials analysis. A range of different scenarios can be modelled prior to the trial, which may offer some guidance regarding the sorties that should be undertaken in order to get the most out of the trial and to achieve the most effective results.

Such a simulation tool requires a software environment in which realistic models of threat missile systems, ADF platforms, environmental effects, countermeasure systems and techniques and other system models can be brought together and run interactively. Staff of the Electronic Warfare and Radar Division at the Defence Science and Technology Organisation (DSTO), with contractor support from Avalon Systems, have been attempting to meet the requirement for such a tool through the development of TRIREME, a flexible Simulink-based simulation environment.

The work is being carried out under the auspices of Project Arrangement 10 (PA10), a Research, Development and Engineering agreement on next generation aircraft survivability technologies between the ADF and the United States Army.

2. SIMULINK-BASED SIMULATION

Simulink [1] is a software package that allows the user to graphically model, simulate and analyse dynamic systems. It comes with a set of customisable libraries of blocks that implement functions commonly used in modelling a system, including signal generators, integrators, unit delays, transfer functions, rate limiters, gains, summing blocks and so forth. Models are built using these blocks in a hierarchical fashion. Data is passed between these blocks via signal lines or buses. Since Simulink is integrated with MATLAB [2] it also provides access to a range of tools for algorithm development, data visualisation, data analysis and access, and numerical computation.

The Simulink programming language is well suited to the logical representation of electronic combat systems, however, it does have several drawbacks when used for engagement scenario simulation. Simulink simulations run slower when compared to simulations written in procedural languages such as C/C++ because Simulink is an interpreted language. One of the biggest problems with using Simulink is that, once a model has been developed, it is "notionally hardwired". This means that it is difficult to add or remove components to or from the simulation without having to make a lot of modifications to the existing Simulink model. Finally, in order to share Simulink models with others they must have a MATLAB and Simulink licence.

3. TRIREME ARCHITECTURE

3.1 TRIREME and Simulink

Despite its inherent limitations, Simulink was the programming language of choice for the purposes of this project, due to the relative ease in which threat and countermeasure systems can be modelled in detail, and the fact that it opens up opportunities for possible future collaboration with other Defence agencies and industries, both locally and internationally, involved in similar engagement modelling activities.

In designing the TRIREME architecture, some creative Simulink code development was required in order to:

- take an object-oriented approach to Simulink model development;
- make the signals in a Simulink model easier to trace and debug;
- allow for multiple instances of threats, target platforms and countermeasure systems;
- create a flexible architecture capable of being expanded over time; and
- design a user-friendly model development environment.

The result is a method for constructing electromagnetic (EM) models in Simulink, allowing for the addition (or subtraction) of radars, missiles and other objects into a TRIREME simulation without the need for large amounts of work to manually change the signal and bus configurations normally associated with Simulink's "hardwired" approach. Also, by following the TRIREME rules, sub-models can be developed by several developers and brought together into the TRIREME framework in a "plug-and-play" fashion. This then provides the ability to build up a library of objects which can be used to create an engagement simulation.

In general, most models developed in Simulink use the signal and bus order to select the required data for calculations, such as EM effects. With the methods used in TRIREME all the signals between the main components are uniquely named (following set naming rules) allowing the appropriate signal to be easily referenced. This makes the order of the signals significantly less important, resulting in a far more flexible, expandable model architecture.

Along with the rules defined in TRIREME, several Simulink tools (referred to as the TRIREME Toolbox) have been developed to simplify the construction of models by taking advantage of these rules and methods. TRIREME also simplifies the configuration and control of the models by configuring the framework separately from the model library and Toolbox blocks. Thus, if an upgrade is made to, say, the Target Aircraft model, only the associated Simulink block needs to be reconfigured, not the whole simulation or framework. It also allows for a "build" of a simulation to be created using the most up-to-date models (similar to doing a build and compile of C code).

3.2 TRIREME Framework

TRIREME consists of two frameworks, one for modelling Radio-Frequency (RF) guided threat engagements and the second for modelling InfraRed (IR) guided threat engagements. The main reason for this breakdown into two environments is to avoid slowing down Simulink execution times, however, the two frameworks could be combined into a single framework if required.

The initial requirement for TRIREME was to provide a capability for simulating one-on-one engagements (i.e. a single missile system against a single target platform employing one or more countermeasures). However, due to its flexible design, the TRIREME Framework, Toolbox and Rules allow for future expansion to simulate more complicated engagement scenarios. As such, simulation of many-on-many engagements is possible in the current version of TRIREME (i.e. incorporating multiple threats and targets employing countermeasures).

Each TRIREME framework allows models to be quickly inserted into or deleted from TRIREME configurations. A TRIREME configuration is defined as a collection of models from the TRIREME database that have been inserted into the TRIREME framework to perform a set task (generally in the form of running a simulation). For example, a configuration might be set up to perform a chaff effectiveness study against a Surface-to-Air Missile (SAM) system. The following discussion applies to both RF and IR frameworks.

The Framework is defined as being everything contained within the first (top) and second levels of the Simulink diagram hierarchy. Figure 1 illustrates the layout of the TRIREME framework at the top level in Simulink (example only). At this level GOTO and FROM blocks are used to pass signals between blocks. Signal lines were not used, because they would clutter the top-level diagram and make it unreadable. The blocks contained at this level have been colour-coded to symbolise their purpose: red for threat systems, green for environment, orange for missile system, white for simulation control and output graphics, and light blue for target platform and countermeasures (e.g. chaff, flare, jammer, decoy, etc.).

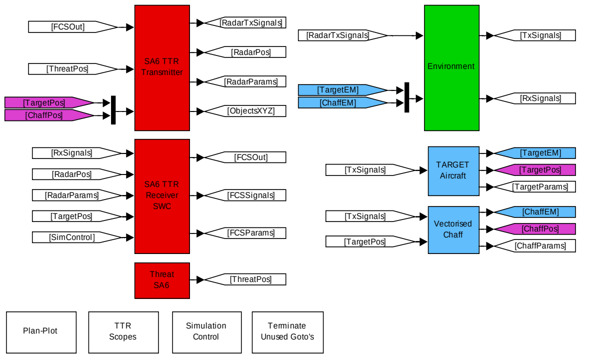

Figure 1: TRIREME Framework Layout - top level (example only)

The second level of the framework (Figure 2) contains TRIREME Toolbox blocks and the plug-and-play block representing the system of interest (e.g. target aircraft model or chaff model or environment model etc.).

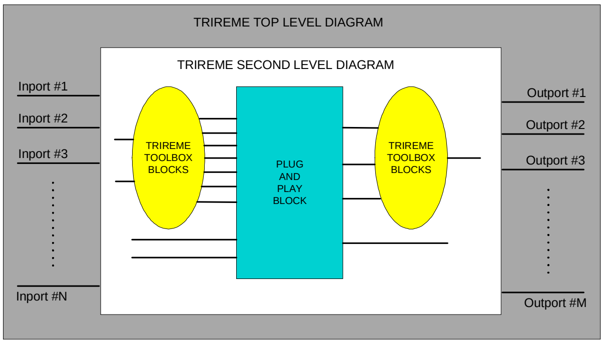

Figure 2: TRIREME Framework Layout - 2nd level

3.3 TRIREME Plug-and-Play Blocks

The plug-and-play block (within the second level of the Simulink diagram hierarchy) represents the location in TRIREME where models may be interchanged - for example changing a generic aircraft model to a specific model (e.g. F-111) by replacing its Simulink block with the appropriate new block from the model database. For models to be inserted into TRIREME, the associated Simulink block must have the same inport and outport signals as the model being replaced.

3.4 TRIREME Toolbox

The TRIREME Toolbox was designed to provide a set of utilities to manage and manipulate Simulink signals and buses. The Toolbox is part of the TRIREME Framework and, along with the signal/bus naming rules, it allows for components to be easily added to or deleted from TRIREME configurations. The Toolbox blocks rely on the naming of signals in Simulink. For the Toolbox to function correctly certain signals in TRIREME must be named using specific rules. The TRIREME Toolbox is made up of the following blocks: Combine Buses, Object Selector, Object Terminator, Signal Selector, Rename Signals and Name Vector Objects.

The Combine Buses block is used to combine two or more buses into a single bus. The Object Selector block is used to select Objects from buses. In TRIREME an Object is defined as any target platform or countermeasure system contained within the TRIREME configuration. The Object Terminator block is used to remove Objects from a bus. The Signal Selector block is used to select signals/variables from a bus. The Rename Signals block was designed to fix the problem of a named signal losing its name after being passed through certain Simulink blocks. The Name Vector Objects block names the output signals for vectorised objects.

3.5 TRIREME Rules

In order for a Simulink model to be compliant with the TRIREME Framework, it must comply with the TRIREME Rules. These Rules define conventions such as the naming of signals between blocks, the signals contained in the buses, restrictions on the use of Simulink blocks in specific locations, or even disallowing the use of certain Simulink blocks altogether.

These rules then facilitate the design and implementation of models in the TRIREME framework and allow for the Toolbox blocks to “dynamically rewire” the simulation when models are added or subtracted.

3.6 Multiple objects in TRIREME

In constructing and running engagement simulations, it is useful to be able to change the number of instances of a particular Object, rather than having a fixed number of instances "hard-coded" in the model. For example, a user may wish to vary the number of chaff rounds deployed in an engagement for a chaff effectiveness study or include multiple aircraft in a scenario. TRIREME allows for the inclusion of multiple instances of Objects in one of two ways: by using either the duplication method or the vectorisation method.

The duplication method is based on the concept of making a copy of the plug-and-play block representing the system of interest and then adding it into the TRIREME Framework. Figure 3 illustrates this idea using the chaff model. The top red block is the original chaff block, and the two red blocks below are the copies. The two extra chaff rounds are connected into the TRIREME framework by using the TRIREME Toolbox. Input parameters for each of the chaff rounds may be entered into a dialog box by clicking on their respective blocks. Although it is possible to manually add copies of plug-and-play blocks into a TRIREME configuration, this process is time consuming and prone to error. As such, the process has been automated in TRIREME through the use of a MATLAB script and the addition of a GUI that allows the user to specify how many instances of an Object they wish to include in the simulation.

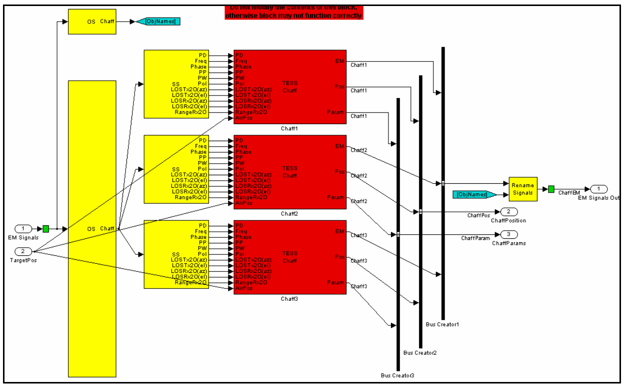

Figure 3: Duplication Method

The vectorisation method works by having the user enter vectors of parameter data into the plug-and-play block's GUI (where each element in a vector corresponds to a separate instance of an object) and then using a single block to perform the required calculations. This concept is referred to in the Simulink on-line help page for blocks as 'Dimensionalised' and 'Scalar Expansion'. Figure 4 provides an example of the vectorised chaff block. This single block is capable of simulating any number of chaff rounds. Note, however, that increasing the number of components in a TRIREME configuration will increase simulation execution time.

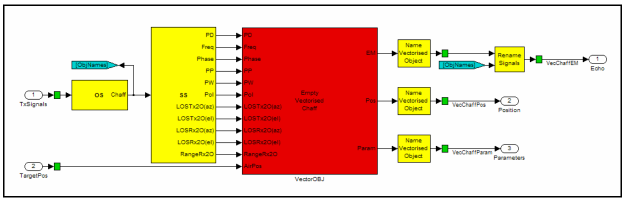

Figure 4: Vectorisation Method

In TRIREME, the vectorisation method is the preferred approach for representing multiple instances of an object, since it has several advantages over the duplication method:

- When the number of instances for a vectorised Object changes, no physical modifications need to be made to the TRIREME Framework.
- There is no need to design, implement and test a new GUI.
- The vectorisation method is computationally more efficient.

The only drawback to vectorisation is that it makes a model more complicated to design, implement and debug. TRIREME currently allows for both methods.

3.7 TRIREME Model Database

The TRIREME model database consists of a library of models that can be inserted into the TRIREME Framework, to build up a scenario of interest (configuration) and then run as a simulation. The TRIREME Framework, Toolbox and model library are under configuration management using the Merant Dimensions software package [3].

The TRIREME model library incorporates a suite of generic models developed by Tactical Technologies Inc (TTI) as part of their Simulink-based Tactical Engagement Simulation Suite (TESS)[1] [4], including models of chaff, flares, jammers, decoys, clutter, aircraft, missile airframe and autopilot, and various threat radars and sensors. In addition to the Simulink-based TESS models, various other C/C++-code models have successfully been integrated into TRIREME using the TRIREME Rules and Toolbox, demonstrating the flexibility of the TRIREME Framework.

Much of the TRIREME effort at DSTO is now focussed on the development of system-specific models for RF- and IR-guided Surface-to-Air Missile (SAM) and Air-to-Air Missile (AAM) systems and countermeasure systems. DSTO has also developed its own generic models for use in TRIREME including chaff, RF jammer, flare, conscan missile seeker and a simple UV missile warning system.

Custom tools have been developed to analyse simulation runs. These tools include Simulink blocks for observing positional information and EM signals of interest to the user. (e.g. scopes for displaying received power, Doppler, etc.).

3.8 TRIREME Configurations

To set up a scenario in TRIREME to run as a simulation, one first has to build a TRIREME configuration - which basically means populating the TRIREME Framework with models of interest from the TRIREME model database. Once a configuration has been built, it can be saved for future simulation studies. Configurations can be readily modified by adding or deleting the blocks (models) representing the systems as required.

Within a TRIREME configuration, the user can set up specific runs by changing model parameters via GUI's (masks) for each plug-and-play block.

Currently, the building of a TRIREME configuration from scratch is a manual process, however, plans are underway for the implementation of scripts to automate this process.

3.9 Speeding Up TRIREME

Simulink simulations can be extremely slow to execute and the more Objects (and hence Simulink blocks) present in a TRIREME configuration, the slower it runs. It is anticipated that there may be a requirement to carry out Monte-Carlo TRIREME runs, and provide TRIREME component models that can be "wrapped" into other simulation environments that run in real time.

---

[1] In order to use these third party models with TRIREME, the user must first acquire the appropriate TESS licence from TTI.

By using the The Mathworks' add-on Simulink toolbox, the Real-Time Workshop (RTW), Simulink models can be converted to C code that can be compiled into an executable that runs much faster than the original model. Initial testing has shown that the TRIREME simulations can be made to run 20-80 times faster using this facility.

4. FUTURE

TRIREME is a work in progress and there are several plans for its improvement and expansion. In addition to the development of a range of specific threat and countermeasure models, the current focus is on use of The Mathworks RTW Toolbox to create executable versions of TRIREME configurations that can be used by ADF operational staff for carrying out countermeasure effectiveness studies, and that can be shared with international R&D partners without the need to release source code.

A user friendly GUI and a suite of visualisation tools for the executable version of TRIREME are currently under development.

Expansion of the TRIREME model database to include realistic representations of real platform signatures is a priority, as is the validation of TRIREME system models through comparison with field trials data and other methods.

TRIREME currently has an air-environment flavour, due to its PA10 origin, however, the addition of land and/or maritime platforms and systems could be easily achieved.

5. SUMMARY

TRIREME is a flexible simulation architecture for the building and running of military engagement scenarios in Simulink. Through creative application of the Simulink language, it provides many of the benefits associated with Simulink-based system modelling whilst avoiding some of the drawbacks associated with a "notionally hardwired" programming language.

A library of system models has been developed that can readily be added to (or deleted from) a TRIREME configuration and the user can vary the number of objects in a simulation without having to make any changes to the configuration itself or the models involved.

TRIREME currently allows for the simulation of a variety of one-on-few scenarios between a guided weapon and a number of aircraft employing a range of electronic countermeasures and other electronic warfare systems, but given the flexible nature of the architecture this could be readily extended to include land and maritime environments as well as multiple threats and a variety of platforms.

REFERENCES

1. URL: http://www.mathworks.com/products/simulink
2. URL: http://www.mathworks.com/products/matlab
3. URL: http://www.serena.com/Products/dimensions/home.asp
4. URL: http://www.tti.on.ca