Automatic Generation of Goal-Tree from Statecharts Requirements Specification

Junbeom Yoo*, Sungdeok Cha*, and Han Seong Son**

*Korea Advanced Institute of Science Technology 373-1 Kuseong-dong, Yuseong-ku, Daejeon, 305-701, Korea Emai:{ jbyoo ,cha}@salmosa.kaist.ac.kr
**Korea Atomic Energy Research Institute 150 Deokjin-dong, Yuseong-ku, Daejeon, 305-353, Korea Email: hsson@kaeri.re.kr

INTRODUCTION

Formal specification and analysis of software in nuclear power plants are known more important as the size and complexity of the software being used increases. Recently the fault tree analysis is applied to the software requirement specification written in Statecharts [1], which is the widely used formal specification language for safety critical systems. However, the existing automatic fault tree generation techniques [2,3] are not suitable for nuclear power plants' software that operates periodically with the input events. This article proposes an event-based goal tree generation technique that can be used as a candidate for the fault tree and that is suitable for the characteristics of the software in nuclear power plants.

STATE-BASED FAULT-TREE GENERATION FROM STATECHARTS

Statecharts is a visual formal specification language, which is widely used in the area of safety critical systems such as aerospace or satellite process controller. It is a finite state machine supplemented by hierarchy, broadcasting, and concurrency concept. Fig.1 illustrates the Statecharts specification for the controller that controls the temperature of coolant in the reactor.

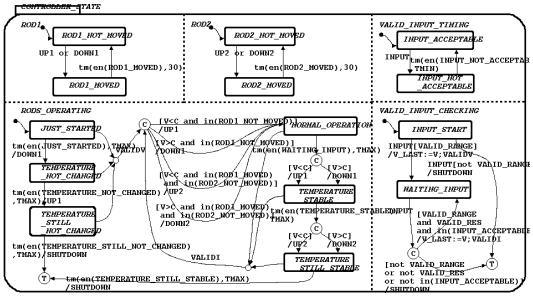


Fig. 1 Statecharts specification for coolant temperature controller

In the approaches in [2,3], The fault-tree is automatically constructed by backward simulation [4] on the Statecharts specification. Therefore the fault-tree generated is based on the simulation states. No further analysis information for events and guarding conditions is provided because it makes the fault tree too large to be interpreted easily [4].

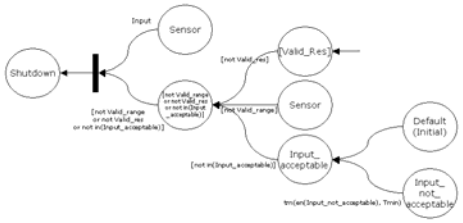
EVENT-BASED GOAL-TREE GENERATION FROM STATECHARTS

The software for nuclear power plants controller operates periodically, i.e. it receives system input events, calculates with them, and then emits system outputs. The information of event in the fault tree corresponding to system inputs is important for the safety analysis of the software controllers. Therefore, the goal-tree, which places importance on the events and is automatically generated, can be used as a suitable candidate for fault-tree in this area.

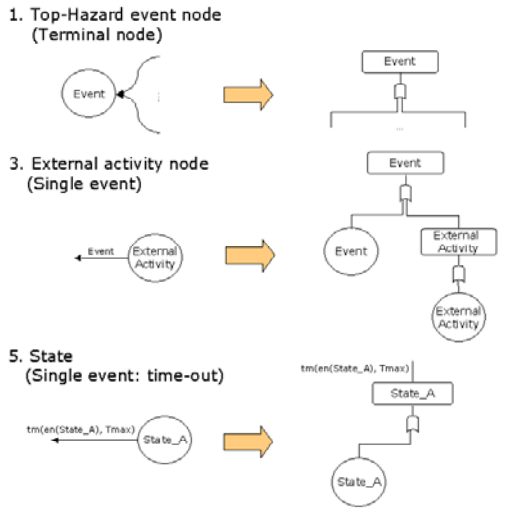
The automatic generation procedure of event-based goal-tree is as follows: **(Step 1)** decide an event as the root node of the goal-tree. **(Step 2)** transform the Statecharts specification into a kind of digraph [5]. **(Step 3)** transform labeled digraph into goal-tree. **(Step 4)** resolve the cycle problems in previous steps. **(Step 5)** convert the goal-tree into fault-tree by negating the root node. Fig.2(a) is the excerpt of the labeled digraph for the event "shutdown" in Fig.1, and Fig.2(b) are some templates for Step 3.

RESULTS

In this article, we propose a technique for the automatic generation of event-based goal-trees from Statecharts requirements specification. It is well suited for the characteristic of nuclear power plants' software and this characteristic makes it a good candidate for the fault-tree.



(a)Labeled digraph for evnet "shutdown"



(b)Teamplates for transforming labeled digrah into goal-tree

Fig.2 Goal tree generation for event “shutdown”

REFERENCES

1. D. Harel,, “On Visual Formalism,” *Communication of ACM*, 31, 5, pp.514-530, (1988)
2. V. Ratan, K. Partridge, J. Reese, and N.G. Leveson, “Safety Analysis Tools for Requirements Specifications,” *in Proc. COMPASS '96,* pp149 –160 (1996).
3. R. Mojdehbakhsh, S. Subramanian, R. Vishnuvajjala, W. Tsai, and L. Elliott, “A Process for Software Requirements Safety Analysis,” *in Proc. Intl. Symp. Software Reliability Engineering*, pp45–54 (1994).
4. E.M. Clarke, O. Grumberg, and D.A.Peled, Model Checking, *MIT Press*, (1999)
5. D.L. Iverson, “Automatic Translation of Digraph to Fault-Tree Models,” *in Proc. Ann. Reliability and Maintainability Symp.*, pp354-362, (1992)