



HAL
open science

SEDMA-un outil pour la Modélisation, l'Analyse et la génération automatique de programme pour les SEDs

Romain Pichard, Michel Combacau, Philippot Alexandre, Ramla Saddem,
Bernard Riera

► **To cite this version:**

Romain Pichard, Michel Combacau, Philippot Alexandre, Ramla Saddem, Bernard Riera. SEDMA-un outil pour la Modélisation, l'Analyse et la génération automatique de programme pour les SEDs. Colloque sur la Modélisation des Systèmes Réactifs (MSR), 2017, Marseille, France. hal-01882579

HAL Id: hal-01882579

<https://hal.science/hal-01882579>

Submitted on 27 Sep 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

SEDMA - un outil pour la Modélisation, l'Analyse et la génération automatique de programme pour les SEDs *

Romain Pichard¹, Michel Combacau², Alexandre Philippot¹, Ramla Saddem¹ et
Bernard Riera¹

¹ CReSTIC (EA3804), UFR Sciences Exactes et Naturelles, Université de
Reims-Champagne-Ardenne (URCA), Moulin de la Housse, 51687 Reims - France
`romain.pichard@univ-reims.fr`, `prenom.nom@univ-reims.fr`

² LAAS-CNRS, 7 avenue du colonel Roche, 31077 Toulouse Cedex - France
`combacau@laas.fr`

Résumé

SEDMA est un outil logiciel permettant la modélisation, l'analyse et la génération automatique de programme pour les systèmes à événements discrets. SEDMA a pour but de fournir un environnement unique pour l'enseignement et l'application des systèmes à événements discrets au travers de plusieurs formalismes de modélisation (automates à états finis, réseaux de Petri, contraintes...). Pour chaque formalisme des outils mathématiques et algorithmiques sont proposés, ceux-ci permettent la création ou bien l'analyse des modèles étudiés. Enfin, SEDMA permet la transformation des modèles vers différents langages tels que le C, le VHDL et le Structured Text (IEC 61131), la principale méthode utilisée est celle enseignée par M. Combacau à l'université Paul Sabatier de Toulouse.

1 Introduction

Les Systèmes à Événements Discrets (SED) permettent la modélisation [1] et le contrôle [2] d'une large gamme de systèmes. Il existe différents outils logiciels permettant la modélisation et l'analyse des SEDs : TINA [3], Romeo [4], Supremica [5], DESUMA [6], CPNtools [7], Grafchart [8], etc. Ces outils sont très performants et en général spécialisés sur un seul formalisme. Néanmoins pour l'enseignement des SEDs il est souvent nécessaire de travailler sur différents formalismes, la prise en main d'un logiciel spécialisé peut être compliqué et par conséquent rajouter une difficulté de compréhension à l'apprenant. Enfin, la plupart du temps ces logiciels ne permettent pas l'exportation des modèles créés vers un langage de programmation, il est donc parfois difficile de passer de la théorie à la pratique.

2 SEDMA

SEDMA a pour but de proposer un environnement unique pour l'apprentissage de plusieurs formalismes autour des SEDs. Actuellement il est possible de travailler sur les automates à états finis, les réseaux de Petri interprétés et les contraintes de sécurité [9]. Pour chacun de ces formalismes différentes fonctionnalités d'analyse, de conception et de simulation sont disponibles.

Néanmoins SEDMA n'a pas pour ambition de proposer l'ensemble des fonctionnalités présentes dans les logiciels spécialisés, c'est pourquoi il est possible d'échanger les modèles avec ces logiciels via un mécanisme d'import/export.

Afin de faciliter le passage à la pratique, SEDMA propose une transformation automatique des modèles vers différents langages de programmation. Actuellement il est possible de générer du code C, ST et VHDL.

*SEDMA provient initialement d'un projet de deux étudiants de Master 1 à l'université Paul Sabatier de Toulouse, Pierre Fernbach et Romain Pichard. L'idée originale et l'encadrement de ce projet a été fait par le Professeur Michel Combacau du LAAS-CNRS. Il est à présent développé par Romain Pichard au CReSTIC.

L'interface utilisateur (figure 1) se veut simple et intuitive. Elle propose un ensemble de menus et d'outils, une vue arborescente sous forme de projet contenant les différents fichiers le composant (modèles, textes, images, pdfs...), une zone de dessin pour la saisie des modèles et une console pour l'affichage d'informations. L'interface est disponible en français et en anglais.

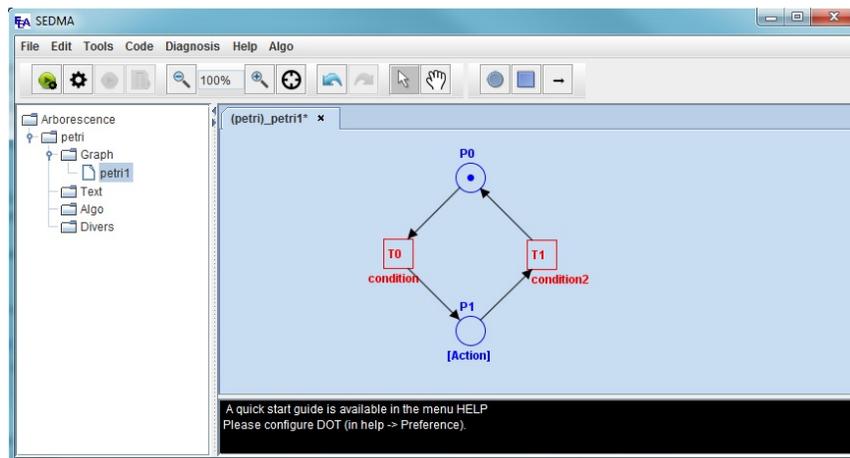


FIGURE 1 – Interface utilisateur de SEDMA

Les principales évolutions à venir sont : l'intégration d'un éditeur de Grafcet/SFC, l'ajout d'import/export vers d'autres logiciels spécialisés et l'amélioration de certains points de l'interface graphique.

Références

- [1] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. Springer Science & Business Media, Dec. 2009.
- [2] J. Zaytoon and B. Riera, "Synthesis and implementation of logic controllers – A review," *Annual Reviews in Control*, Mar. 2017.
- [3] B. Berthomieu, P.-O. Ribet, and F. Vernadat, "The tool TINA – Construction of abstract state spaces for petri nets and time petri nets," *International Journal of Production Research*, vol. 42, pp. 2741–2756, July 2004.
- [4] D. Lime, O. H. Roux, C. Seidner, and L.-M. Traonouez, "Romeo : A Parametric Model-Checker for Petri Nets with Stopwatches," in *Tools and Algorithms for the Construction and Analysis of Systems*, Lecture Notes in Computer Science, pp. 54–57, Springer, Berlin, Heidelberg, Mar. 2009.
- [5] K. Akesson, M. Fabian, H. Flordal, and R. Malik, "Supremica - An integrated environment for verification, synthesis and simulation of discrete event systems," in *2006 8th International Workshop on Discrete Event Systems*, pp. 384–385, July 2006.
- [6] L. Ricker, S. Lafortune, and S. Genc, "Desuma : A tool integrating giddes and umdes," in *Software tools, 8th international workshop on discrete-event systems*, 2006.
- [7] K. Jensen, L. M. Kristensen, and L. Wells, "Coloured Petri Nets and CPN Tools for modelling and validation of concurrent systems," *International Journal on Software Tools for Technology Transfer*, vol. 9, pp. 213–254, June 2007.
- [8] A. Theorin, *A Sequential Control Language for Industrial Automation*, vol. TFRT-1104. Department of Automatic Control, Lund Institute of Technology, Lund University, 2014.
- [9] B. Riera, A. Philippot, D. Annebique, and F. Gellot, "La commande par contraintes logiques de sécurité : principe, applications et mise en oeuvre," in *Modélisation des Systèmes Réactifs*, (Nancy), 2015.