

DESENVOLVIMENTO E IMPLEMENTAÇÃO DE FUNCIONALIDADES NO NADZORU 2: EDITOR E GERADOR DE CÓDIGO¹

Breno Niehues dos Santos², Yuri Kaszubowski Lopes³, Adilson Krischanski⁴, Nicole Carolina Mendes⁵

¹ Vinculado ao projeto “Ferramentas para controle supervísório e aplicações na robótica móvel”

² Acadêmico (a) do Curso de Engenharia Elétrica – CCT – Bolsista PROBIC

³ Orientador, Departamento de Ciência da Computação – CCT – yuri.lopes@udesc.br

⁴ Acadêmico (a) do Curso de Ciência da Computação – CCT – Bolsista PROBIC

⁵ Acadêmico (a) do Curso de Ciência da Computação – CCT – Voluntária IC

Desenvolvido pelo Grupo de Automação de Sistemas e Robótica (GASR) da UDESC, o Nadzoru [1] é uma ferramenta computacional para auxílio à síntese de supervisores para Sistemas a Eventos Discretos (SEDs). O Nadzoru permite a modelagem, síntese, análise, simulação e geração de código para controladores utilizando a Teoria de Controle Supervísório, combinando funcionalidades de outros programas do gênero em uma só ferramenta. Inicialmente escrito em Lua, sua primeira versão é utilizada nas matérias finais da graduação em Engenharia Elétrica, no PPGEEL e em diversas pesquisas, por exemplo, no controle de enxames de robôs [2].

O objetivo deste trabalho é contribuir com a produção da segunda versão da ferramenta, o Nadzoru 2, que está sendo desenvolvido em Python e em GTK, um *toolkit* para criação da interface gráfica. Os trabalhos realizados previamente já haviam resolvido a grande maioria da implementação da estrutura de dados para autômatos, porém a ferramenta não continha meios gráficos de modelá-los, sendo, então, necessário desenvolver o Editor de autômatos da ferramenta. Outra função implementada no Nadzoru 2 foi a geração de códigos para a plataforma Arduino, que foi escrita com auxílio do mecanismo de *template* Jinja2.

Diversas funcionalidades foram adicionadas no Editor de autômatos e nos locais necessários para acomodar tais mudanças. Foram implementadas tanto a parte lógica, para carregar autômatos salvos em arquivo do com extensão “.xml”, quanto a parte gráfica, para permitir que o usuário consiga efetuar tal ação facilmente. Neste mesmo quesito, também foi adicionada a importação de arquivos gerados na primeira versão do Nadzoru. Ainda no Editor de autômatos, foram feitas várias modificações e adições à caixa de ferramentas, como as ferramentas com função de editar estados/transições, de excluir estados/transições e de enumerar estados. Além disso, também se implementou a lógica para registrar cliques nos estados e nos arcos das transições – onde estes são renderizados utilizando a biblioteca de gráficos 2D Cairo. Um exemplo de autômato em uma janela com o Editor aberto está apresentado na Figura 1.

Posteriormente, viu-se a necessidade de criar uma janela contendo a lista das operações para a síntese de supervisores e a interface gráfica que a acompanha para permitir a entrada de parâmetros. Essa janela é gerada dinamicamente, possibilitando que a adição de novas operações seja feita de maneira descomplicada.

Com a implementação das operações, notou-se a necessidade de gerenciar os autômatos abertos na memória que não estão abertos no editor, já que as vezes se faz necessário fechar, editar ou salvar os supervisores que estão carregados, porém não se encontram abertos nas abas. A solução foi criar uma janela para gerenciamento dos autômatos abertos na memória que permite, de maneira simples, clonar, renomear, salvar, fechar, simular e editar tais autômatos. Além disso,

foi criada a funcionalidade de abrir os autômatos no editor direto pelo submenu *Edit*, como acontece usualmente em outras ferramentas nos submenus da categoria “Abertos recentemente”.

A última grande contribuição feita ao projeto foi a implementação do gerador de código, que permite a geração do código em C para microcontroladores da plataforma Arduino a partir dos supervisores abertos no Nadzoru. A geração de código acontece utilizando o mecanismo de *template* Jinja2, que permite a criação de modelos de arquivos de texto onde seções do texto são adicionadas dinamicamente com a entrada de parâmetros. Assim como todo o projeto, têm-se a intenção de simplificar o processo de adições de novas funcionalidades, portanto, a interface gráfica é gerada de acordo com os modelos implementados na ferramenta, o que possibilita a adição de novos controladores futuramente, como CLPs. Um exemplo da tela de geração de código é apresentado na Figura 2.

Iniciou-se ainda o desenvolvimento de uma barra de status que futuramente irá passar ao usuário informações relevantes, como erros ou observações, porém, esta funcionalidade ainda não se encontrada implementada no Nadzoru 2.

Por fim, é interessante comentar que para trabalhos futuros ainda há muito o que aprimorar e implementar na ferramenta, como por exemplo, finalizar a barra de status, implementar uma função de desfazer/refazer e aprofundar o funcionamento da simulação de supervisores.

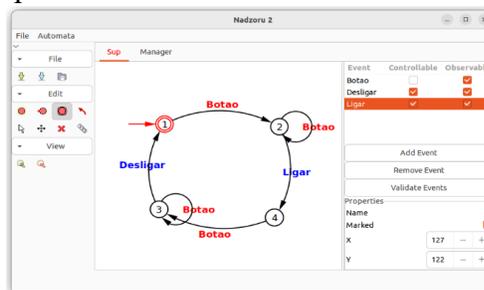


Figura 1. Janela com uma aba do Editor de autômatos aberta.

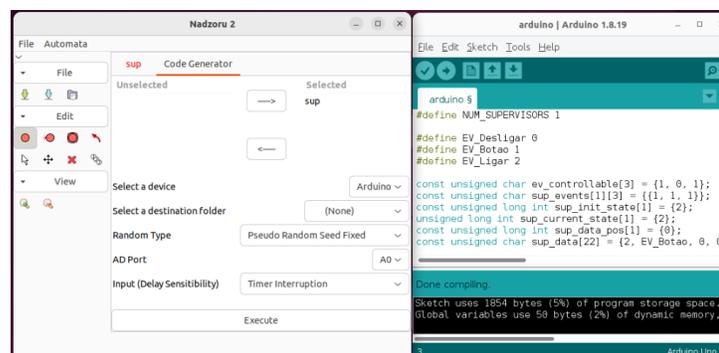


Figura 2. Janela com uma aba do gerador de códigos aberta e código resultante na IDE do Arduino.

Palavras-chave: Controle Supervisório. Sistemas de Eventos Discretos. Geração de Código.

Referências:

- [1] LP Pinheiro, YK Lopes, AB Leal, RSUR Junior. Nadzoru: A software tool for supervisory control of discrete event systems. *FAC-PapersOnLine* 48 (7), 182-187, 2015.
- [2] YK Lopes, SM Trenkwalder, AB Leal, TJ Dodd, R Groß. Supervisory control theory applied to swarm robotics. *Swarm Intelligence* 10 (1), 65-97, 2016.