

CONTRIBUIÇÕES PARA O DESENVOLVIMENTO DA FERRAMENTA COMPUTACIONAL NADZORU 2 ¹

Vinícius Amancio², André Bittencourt Leal³, Yuri Kaszubowski Lopes⁴.

¹ Vinculado ao projeto “Controle Supervisório de Sistemas a Eventos Discretos: em direção à Indústria 4.0”

² Acadêmico do Curso de Engenharia Elétrica – CCT – Bolsista PROBIC/UDESC

³ Orientador, Departamento de Engenharia Elétrica – CCT – andre.leal@udesc.br

⁴ Departamento de Ciência da Computação – CCT – yuri.lopes@udesc.br

O objetivo deste trabalho consiste em contribuir para a reformulação do Nadzoru, uma ferramenta computacional desenvolvida por integrantes do Grupo de Pesquisa em Automação de Sistemas e Robótica – GASR da UDESC para auxiliar na síntese de supervisores para Sistemas a Eventos Discretos (SEDs). A versão original do Nadzoru foi desenvolvida em linguagem Lua, que conta com uma reduzida comunidade de desenvolvimento e uma base de dados limitada, o que dificulta a manutenção e a atualização da ferramenta. Assim, os coordenadores do projeto optaram por desenvolver uma nova versão do Nadzoru em Python, que conta com grande número de desenvolvedores e uma base de dados mais extensa, com diversas funções já implementadas.

Dessa forma, com o objetivo de identificar aspectos positivos de outras ferramentas que possam ser incorporados na nova versão do Nadzoru, foi feita uma análise comparativa entre a versão atual do Nadzoru e as principais ferramentas de mesmo propósito. Dentre as ferramentas de destaque podemos citar: IDES3, Supremica, TCT, DESTool e UltraDES. As principais funcionalidades suportadas por ferramentas para SEDs são: *i*) composição síncrona; *ii*) produto; *iii*) componente acessível; *iv*) componente coacessível, *v*) minimização; *vi*) cálculo do supervisor; *vii*) edição gráfica dos autômatos; *viii*) importação e exportação de arquivos compatíveis com outras ferramentas; *ix*) geração automática de código; *x*) simulação de autômatos; *xi*) identificação de problemas relacionados à implementação. No levantamento realizado neste trabalho identificou-se que as funcionalidades *i-vi* são presentes em todas as ferramentas. As demais funcionalidades são como segue: IDES3 conta ainda com *vii-viii*, Supremica conta com *vii* e *ix*; e DESTool com *vii* e *x*. Também se percebe dessa comparação, que o TCT e o UltraDES possuem eficiência computacional bastante acima das demais, permitindo maior velocidade nos cálculos e maior capacidade de tratar problemas de grande porte. Ainda, o Nadzoru é a única ferramenta que conta com identificação de problemas que podem surgir na implementação da lógica de controle, tais como: efeito avalanche, simultaneidade, problema de escolha e sincronização inexata. Duas outras funcionalidades se destacam no Nadzoru: a capacidade de simulação de autômatos (presente apenas no DESTool) e a geração automática de código para diferentes dispositivos de controle. Esses são importantes características do Nadzoru que deverão ser mantidos na nova versão.

A ferramenta em desenvolvimento deve prover funcionalidades que consistem em uma série de algoritmos que operam sobre conjuntos de dados dinâmicos, os quais consistem em agrupamento de elementos que podem crescer, encolher ou sofrer mudanças ao longo do tempo. Assim, esses conjuntos dinâmicos agregam elementos que representam estados, eventos e transições de autômatos. Em particular, as transições representam ligações lógicas entre elementos dos vários conjuntos dinâmicos: um estado de origem, um estado de destino e o evento relacionado. Desta forma, a alteração de um conjunto acaba por impactar outros conjuntos. Por exemplo, ao se remover um estado ou evento todas as transições associadas devem ser removidas. A representação

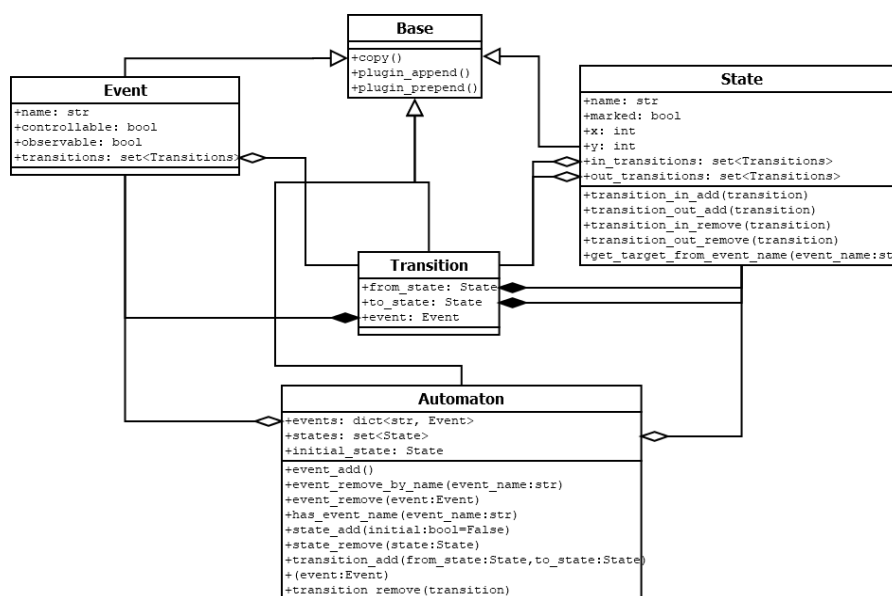
de conjuntos de dados dinâmicos em memória é realizada por estruturas de dados que definem a organização, as formas de processamento e de acesso dos elementos (dados) dos conjuntos dinâmicos.

Por questões de eficiência de certos algoritmos, é desejável que cada elemento tenha uma listagem de suas associações. Por exemplo, cada evento carrega internamente uma lista de referências para as transições que está envolvido; o mesmo ocorre com os estados, com listas das transições de saída e das transições de entrada que este estado está relacionado. Por um lado, esta técnica reduz o esforço na busca de todas as transições relacionadas a um dado elemento; na contramão, há um esforço maior para a manutenção destas estruturas de dados. Existe, portanto, um *'trade-off'* de quais operações/algoritmos serão mais ou menos eficientes devido as estruturas de dados selecionadas. Portanto, é fundamental conhecer as funcionalidades mais usuais da ferramenta de forma a guiar o projeto dos requisitos das estruturas de dados.

A partir do levantamento de funcionalidades de outras ferramentas, este trabalho colaborou na definição das estruturas de dados para a nova versão do Nadzoru. O bolsista identificou requisitos e apontou inviabilidades durante as etapas de projeto das estruturas de dados de forma a garantir que um conjunto de funcionalidades usuais fossem factíveis com as estruturas de dados do Nadzoru 2. Sendo assim, pode-se dizer que as atividades realizadas no decorrer desta iniciação científica contribuíram para o desenvolvimento da nova versão do Nadzoru.

A estrutura de dados projetada é ilustrada na Figura 1, na forma de um diagrama UML, e possui as classes *Base*, *State*, *Event*, *Transition* e *Automaton*. A classe *Base* provê funcionalidades de um sistema de *plugin*, que permite a extensão das funcionalidades, de modo que todas as classes herdam da classe *Base* (indicado pelas ligações com setas brancas). A principal classe é a *Automaton*, que consiste na agregação (indicada por losangos brancos) dos eventos e estados do Autômato. As transições em um autômato se dão de um estado de origem, com um evento, para um estado de destino, de modo que a classe *Transition* tem uma relação de composição (losangos pretos) das classes *Event* e *State*. O projeto está hospedado no repositório do GASR no endereço <https://github.com/GASR-UDESC/Nadzoru2>.

Figura 1: Diagrama UML do autômato



Palavras-chave: Sistemas a Eventos Discretos. Ferramenta Computacional. Controle Supervisório.