



Evento	Salão UFRGS 2014: SIC - XXVI SALÃO DE INICIAÇÃO CIENTÍFICA DA UFRGS
Ano	2014
Local	Porto Alegre
Título	Desenvolvimento da ferramenta VeriGraph: algoritmos de "matching" e reconhecimento de condições negativas de aplicação.
Autor	RICARDO GABRIEL HERDT
Orientador	RODRIGO MACHADO

O projeto Verites dedica-se ao estudo de técnicas e ao desenvolvimento de ferramentas para verificação, validação e teste de sistemas computacionais. Uma de suas linhas de pesquisa envolve gramáticas de grafos, as quais consistem de modelos computacionais onde um grafo descreve o estado do sistema e um conjunto de regras de reescrita define seu comportamento. Gramáticas de grafos têm diversas aplicações em sistemas computacionais, sendo úteis, por exemplo, na modelagem de sua evolução estrutural. Para tanto, foi desenvolvido o modelo de *gramática de grafos de segunda ordem* (GGSO), onde uma gramática de grafos é o estado de sistema e um conjunto de regras de segunda ordem definem suas transformações.

Existem diversas ferramentas para visualização, escrita e análise de gramáticas de grafos, cada qual com recursos voltados para abordagens distintas do problema. Algumas delas oferecem recursos de análise estática do sistema, outras restringem-se à análise dinâmica. Neste contexto, iniciou-se o desenvolvimento do VeriGraph, uma ferramenta que permitirá o emprego de ambas técnicas de análise, além de oferecer — através da natureza genérica das estruturas e algoritmos escolhidos e da flexibilidade da linguagem utilizada no desenvolvimento (Haskell) — um ambiente de fácil prototipação de novos modelos teóricos (como GGSO's).

Neste projeto, o aluno participa diretamente do desenvolvimento do VeriGraph. Além de atuar nas escolhas e definições estruturais do programa, coube a ele investigar e implementar um algoritmo de *matching* de grafos, cuja eficiência assume um papel central no desempenho do programa. A estratégia adotada foi a modificação de um algoritmo de CSP (*Constraint Satisfaction Problem*) existente de modo a tratar características específicas do problema. A flexibilidade desta abordagem permitiu, por exemplo, que a busca eliminasse *matches* geradores de conflitos de reescrita, reduzindo a complexidade das etapas seguintes do programa. Outro recurso implementado nesta primeira etapa do projeto é o reconhecimento de *NAC's* (condições negativas de aplicação), necessárias para a definição de regras de reescrita mais complexas.