

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA**

João Paulo Taylor Ienczak Zanette

**LIMITAÇÕES DAS TÉCNICAS DE GERAÇÃO
AUTOMÁTICA DE PROGRAMAS DE TESTE PARA A
VERIFICAÇÃO DE MEMÓRIA COMPARTILHADA EM
MULTI-CORES**

Florianópolis

2017

João Paulo Taylor Ienczak Zanette

**LIMITAÇÕES DAS TÉCNICAS DE GERAÇÃO
AUTOMÁTICA DE PROGRAMAS DE TESTE PARA A
VERIFICAÇÃO DE MEMÓRIA COMPARTILHADA EM
MULTI-CORES**

Trabalho de Conclusão de Curso submetido ao Curso de Bacharelado em Ciências da Computação para a obtenção do Grau de Bacharel em Ciências da Computação.
Orientador: Prof. Dr. Luiz Claudio Villar dos Santos

Florianópolis
2017

FOLHA DE APROVAÇÃO DE PROPOSTA DE TCC

Acadêmico	João Paulo Taylor Ienczak Zanette
Título do trabalho	Limitações das Técnicas de Geração Automática de Programas de Teste para a Verificação de Memória Compartilhada em Multi-cores
Curso	Ciências da Computação/INE/UFSC
Área de Concentração	Hardware

Instruções para preenchimento pelo ORIENTADOR DO TRABALHO:

- Para cada critério avaliado, assinale um X na coluna SIM apenas se considerado aprovado. Caso contrário, indique as alterações necessárias na coluna Observação.

Critérios	Aprovado				Observação
	Sim	Parcial	Não	Não se aplica	
1.O trabalho é adequado para um TCC no CCO/SIN (relevância/abrangência)?					
2.O título do trabalho é adequado?					
3.O tema de pesquisa está claramente descrito?					
4.O problema/hipóteses de pesquisa do trabalho está claramente identificado?					
5.A relevância da pesquisa é justificada?					
6.Os objetivos descrevem completa e claramente o que se pretende alcançar neste trabalho?					
7.É definido o método a ser adotado no trabalho? O método condiz com os objetivos e é adequado para um TCC?					
8.Foi definido um cronograma coerente com o método definido (indicando todas as atividades) e com as datas das entregas (p.ex.Projeto I, II, Defesa)?					
9.Foram identificados custos relativos à execução deste trabalho (se houver)? Haverá financiamento para estes custos?					
10.Foram identificados todos os envolvidos neste trabalho?					
11.As formas de comunicação foram definidas (ex: horários para orientação)?					
12.Riscos potenciais que podem causar desvios do plano foram identificados?					
13.Caso o TCC envolva a produção de um software ou outro tipo de produto e seja desenvolvido também como uma atividade realizada numa empresa ou laboratório, consta da proposta uma declaração (ANEXO A) de ciência e concordância com a entrega do código fonte e/ou documentação produzidos?					

Avaliação	<input type="checkbox"/> Aprovado	<input type="checkbox"/> Não Aprovado
Professor Responsável	Prof. Dr. Luiz Claudio Villar dos Santos	
Orientador	Prof. Dr. Luiz Claudio Villar dos Santos	

RESUMO

Multiprocessadores em chip demandam o uso de protocolos para assegurar a consistência de memória compartilhada e a coerência de cache, os quais são implementados via hardware. A crescente complexidade dessas implementações torna o projeto de sistemas de memória suscetível a erros. Para endereçar esse problema, técnicas de verificação pré-silício têm sido estudadas no meio acadêmico. Por serem executadas em simuladores de uma plataforma real, essas técnicas não viabilizam a execução de testes demasiado extensos. Pensando nisso, algumas abordagens de geração adaptativa foram propostas, procurando maximizar a cobertura funcional dos testes sem aumentar a extensão dos mesmos. Neste trabalho é proposta a elaboração de uma nova técnica de geração automática de testes adaptativos. A técnica será concebida a fim de aumentar a eficácia em se expor erros de projeto no subsistema de memória compartilhada, em especial erros de coerência, e manter-se independente da organização da memória, permitindo o reuso em diversos sistemas.

Palavras-chave: EDA, verificação, pré-silício, memória compartilhada, multicore, coerência de cache

SUMÁRIO

1	INTRODUÇÃO	9
2	OBJETIVOS	11
2.1	Objetivo Geral	11
2.2	Objetivos Específicos	11
2.3	Método de Pesquisa	11
3	CRONOGRAMA	13
4	CUSTOS	15
5	RECURSOS HUMANOS	17
6	COMUNICAÇÃO	19
7	RISCOS	21
	REFERÊNCIAS	23
	 ANEXOS	 25
	ANEXO A – DECLARAÇÃO PADRÃO PARA EMPRESA OU LABORATÓRIO	27

1 INTRODUÇÃO

Depois de ter provocado uma mudança radical na forma de se projetar microprocessadores para PCs e servidores, o multi-processamento em *chip* ou *Chip Multi-Processing (CMP)* levou ao atual quadro onde *chips* com 8 ou mais *cores* são bastante comuns. Prevê-se o uso de *chips* com dezenas a centenas cores e a sobrevida da abstração de memória compartilhada como condição para viabilizar a programação de propósitos gerais (DEVADAS, 2013). Isso resulta no desafio de como manter coerência de memória compartilhada **quando se aumenta a escala do número de cores em um único chip**. Para suprir a abstração de memória compartilhada coerente, o subsistema de memória torna-se mais sofisticado e, portanto, mais suscetível a erros de projeto com o aumento massivo do número de *cores*. Além disso, como se espera que grande parte dos programas paralelos utilize bibliotecas para sincronização, a maioria dos programadores não precisa se preocupar com as regras de consistência (ADVE; GHARACHORLOO, 1996) do modelo de memória (HENNESSY; PATTERSON, 2011), o que tende a popularizar o uso de **modelos com máxima relaxação da ordem de programa** para aumentar o desempenho sem comprometer a programabilidade. Isso também contribui para aumentar a complexidade do hardware.

Portanto, a dificuldade de se validar sistemas computacionais baseados em CMP tende a aumentar dramaticamente a cada nova geração de produtos a serem lançados. Ora, **a validação da coerência e da consistência do subsistema de memória compartilhada**, o qual inclui múltiplos níveis de cache e protocolos complexos, constitui grande parte do esforço de se validar um sistema computacional baseado em CMP (WAGNER; BERTACCO, 2008). Como o número de estados induzidos por um protocolo de coerência cresce exponencialmente com o aumento do número de cores (SHIM et al., 2013), torna-se bastante desafiador o problema de verificar se o projeto de um chip baseado em múltiplos *cores* em larga escala implementa corretamente o comportamento esperado para o subsistema de memória compartilhada.

As primeiras técnicas de validação do sistema de memória foram propostas para serem aplicadas após a fabricação (**teste pós-silício**), através da execução de longos programas de testes aleatórios no próprio hardware do protótipo. Infelizmente, o simples reuso dessas técnicas é inadequado em tempo de projeto (para **verificação pré-silício**), porque seriam demasiadamente lentas quando os programas de teste são executados em simuladores, o que requer a limitação do tamanho do teste, o que restringe a qualidade da verificação. Por isso, as técnicas de verificação tem se mostrado frequentemente incapazes de encontrar erros sutis de projeto, que acabam passando para o hardware.

2 OBJETIVOS

2.1 Objetivo Geral

O objetivo geral desta proposta é o estudo de técnicas de geração de testes mais eficientes e eficazes, a serem aplicadas em tempo de projeto, para solucionar o problema de verificação de consistência e coerência de memória compartilhada em sistemas computacionais baseados em CMP.

2.2 Objetivos Específicos

- **Analisar técnicas existentes:** investigar as principais limitações das técnicas reportadas na literatura, a fim de identificar oportunidades de contribuição científica.
- **Implementar um gerador de testes dirigidos:** tomar como base um gerador reportado na literatura (e.g. (WAGNER; BERTACCO, 2008), (ELVER; NAGARAJAN, 2016)).
- **Comparar com outros geradores:** comparar o gerador implementado com um convencional de testes aleatórios (RAMBO; HENSCHER; SANTOS, 2011) e com um de testes aleatórios sob fortes restrições (ANDRADE; GRAF; SANTOS, 2016b).

2.3 Método de Pesquisa

O método a ser adotado consiste na realização de experimentos sobre uma **representação de projeto do sistema sob verificação**, ou seja, na execução de testes através da simulação daquela representação, utilizando como infraestrutura um **simulador** de domínio público denominado gem5 (THE...,). A representação de projeto utilizada para o subsistema de memória corresponde ao módulo *Ruby* daquele simulador, a qual permite a descrição das máquinas de estado dos protocolos de coerência.

Para desempenhar o papel de **verificador** da correção funcional do subsistema de memória representado, será utilizada uma técnica de análise automática dos eventos registrados durante a simulação, a qual foi desenvolvida localmente (FREITAS; RAMBO; SANTOS, 2013) e cujo código está portanto disponível para uso no laboratório hospedeiro.

O protótipo de gerador de testes dirigidos será implementado com base na descrição de seu algoritmo conforme descrito na literatura. Será escolhido o gerador cuja descrição for mais completa e leve à melhor reprodutibilidade.

Como representante das técnicas de geração convencional de testes aleatórios será utilizado o gerador denominado PLAIN (RAMBO; HENSCHER; SANTOS, 2011). Como representante das técnicas de geração convencional de testes aleatórios sob fortes restrições será utilizado o gerador denominado CHAIN (ANDRADE; GRAF; SANTOS, 2016b). Os códigos dos dois geradores estão disponíveis para uso no laboratório hospedeiro.

Os três geradores serão comparados ao se usar a mesma representação de projeto, o mesmo simulador e o mesmo verificador acima mencionados.

Para a simulação de erros de projeto, serão utilizados como base os cinco erros artificiais descritos em (ANDRADE; GRAF; SANTOS, 2016a), mas novos erros artificiais serão desenvolvidos como parte do trabalho proposto para aumentar a abrangência da comparação.

Para a comparação serão adotadas as seguintes métricas: cobertura funcional, esforço computacional e probabilidade de detecção de erros.

3 CRONOGRAMA

Etapas	Meses											
	jul.	ago.	set.	out.	nov.	dez.	jan.	fev.	mar.	abr.	mai.	jun.
Estudo dos modelos de consistência de memória	x											
Estudo de um protocolo de coerência (MESI)	x	x									x	
Estudo de geradores de testes aleatórios e testes dirigidos	x	x	x									
Familiarização com o simulador (gem5)		x	x									
Familiarização com a representação de memória (Ruby)			x									
Criação e caracterização de novos erros artificiais de projeto				x								
Implementação do protótipo do gerador de testes dirigidos					x	x	x	x	x	x		
Execução de testes produzidos com o gerador implementado											x	
Execução de testes produzidos com o gerador PLAIN											x	
Execução de testes produzidos com o gerador CHAIN											x	
Comparação dos geradores de acordo com as métricas											x	
Entrega do relatório de TCC I					o							
Entrega do rascunho da monografia											o	
Defesa do TCC												o

** Por técnicas entende-se aquelas já disponíveis na infraestrutura do laboratório e outras que apresentem implementação em domínio público*

4 CUSTOS

Item	Quant.	Valor Un. (R\$)	Total (R\$)
Material de Consumo			
CD	6	1,50	9,00
Folhas Impressas	500	0,20	100,00

5 RECURSOS HUMANOS

Nome	Função
João Paulo Taylor Ienczak Zanette	Autor
Prof. Dr. Luiz Claudio Villar dos Santos	Orientador/Professor Responsável
Gabriel Arthur Gerber Andrade	Ver função adequada
Marleson Graf	Ver função adequada
Renato Cislighi	Coordenador de Projetos

6 COMUNICAÇÃO

O que precisa ser comunicado	Por quem	Para quem	Melhor forma de Comunicação	Quando e com que frequência
Entregas	Autor	Coordenador de Projetos	Site do TCC	Nos dias estipulados pelas disciplinas
Reuniões com o orientador	Autor	Orientador	Presencial	Quinzenalmente
Revisões da monografia	Autor	Orientador, membros da banca	Papel impresso ou pdf	Período de elaboração da monografia
Dúvidas	Autor	Orientador, Membros da Banca ou Coordenador de Projetos	E-mail e/ou presencial	Quando necessário

7 RISCOS

Risco	Probabilidade	Impacto	Prioridade	Estratégia de Resposta	Ações Preventivas
Perda de dados (HD)	Média	Alto	Alta	Recuperação dos dados e aquisição de novo HD.	Versionar desenvolvimento remotamente. Gerar backups periodicamente.
Alteração no tema	Baixa	Alto	Alta	Modificar o escopo do tema ou adotar um novo tema.	Manter interação constante com orientador.
Alteração no cronograma	Baixa	Alto	Média	Diminuir o escopo do trabalho.	Monitorar continuamente as informações obtidas dos superiores imediatos.
Problemas de saúde	Baixa	Média	Média	Realizar tratamento e retomar as atividades o quanto antes.	Ter hábitos saudáveis e ser precavido.

REFERÊNCIAS

ADVE, S. V.; GHARACHORLOO, K. Shared memory consistency models: a tutorial. *Computer*, IEEE, v. 29, n. 12, p. 66–76, Dec 1996. ISSN 0018-9162.

ANDRADE, G. A. G.; GRAF, M.; SANTOS, L. C. V. dos. In: *Ver tese*. [S.l.: s.n.], 2016.

ANDRADE, G. A. G.; GRAF, M.; SANTOS, L. C. V. dos. Chain-Based Pseudorandom Tests for Pre-Silicon Verification of CMP Memory Systems. In: *34th IEEE International Conference on Computer Design (ICCD)*. [S.l.: s.n.], 2016. p. 552–559.

DEVADAS, S. Toward a coherent multicore memory model. *Computer*, IEEE, n. 10, p. 30–31, 2013.

ELVER, M.; NAGARAJAN, V. McVerSi: A test generation framework for fast memory consistency verification in simulation. In: *IEEE Int. Symp. on High Performance Computer Architecture (HPCA)*. [S.l.: s.n.], 2016. p. 618–630.

FREITAS, L. S.; RAMBO, E. A.; SANTOS, L. C. V. dos. On-the-fly verification of memory consistency with concurrent relaxed scoreboards. In: *Design, Automation, and Test in Europe (DATE)*. [S.l.: s.n.], 2013. p. 631–636. ISBN 978-1-4503-2153-2.

HENNESSY, J. L.; PATTERSON, D. A. *Computer Architecture: A Quantitative Approach*. 5th. ed. [S.l.]: Morgan Kaufmann Publishers Inc., 2011. ISBN 012383872X, 9780123838728.

RAMBO, E.; HENSCHER, O.; SANTOS, L. dos. Automatic generation of memory consistency tests for chip multiprocessing. In: *IEEE Int. Conf. on Electronics, Circuits and Systems (ICECS)*. [S.l.: s.n.], 2011. p. 542–545.

SHIM, K. S. et al. Design tradeoffs for simplicity and efficient verification in the execution migration machine. In: *IEEE. Computer Design (ICCD), 2013 IEEE 31st International Conference on*. [S.l.], 2013. p. 145–153.

THE Gem5 Simulator. Disponível em: <www.m5sim.org>. Acesso em: Julho 2012.

WAGNER, I.; BERTACCO, V. MCjammer: Adaptive Verification for Multi-core Designs. In: *Design, Automation, and Test in Europe (DATE)*. [S.l.: s.n.], 2008. p. 670–675. ISSN 1530-1591.

Anexos

**ANEXO A – DECLARAÇÃO PADRÃO PARA EMPRESA OU
LABORATÓRIO**

**DECLARAÇÃO DE CONCORDÂNCIA COM AS CONDIÇÕES PARA O
DESENVOLVIMENTO DO TCC NA INSTITUIÇÃO**

Declaro estar ciente das premissas para a realização de Trabalhos de Conclusão de Curso (TCC) de Ciência da Computação e Sistema de Informações da UFSC, particularmente da necessidade de que se o TCC envolver o desenvolvimento de um software ou produto específico (ex: um protocolo, um método computacional, etc.) o código fonte e/ou documentação completa correspondente deverá ser entregue integralmente, como parte integrante do relatório final do TCC.

Ciente dessa condição básica, declaro estar de acordo com a realização do TCC identificado pelos dados apresentados a seguir.

Instituição	ECL/INE/CTC
Nome do Responsável	Prof. Dr. Luiz Claudio Villar dos Santos
Cargo/Função	Prof. INE/CTC
Fone de Contato	(48) 3721 7549
Acadêmico	João Paulo Taylor Ienczak Zanette
Título do trabalho	Verificação pré-silício de memória baseada em testes dirigidos adaptativos
Curso	Ciências da Computação/INE/UFSC

Florianópolis, 14 de Dezembro de 2015.

Professor Responsável

Prof. Dr. Luiz Claudio Villar dos Santos