

Uma Análise sobre Ferramentas de Redes-em-Chip e seus Recursos para Uso no Ensino

Eduardo Alves da Silva, Cesar Albenes Zeferino

Laboratório de Sistemas Embarcados e Distribuídos
Universidade do Vale do Itajaí

Itajaí, Brasil

eduardoalves@edu.univali.br, zeferino@univali.br

Resumo — As Redes-em-Chip (do inglês, NoCs – Networks-on-Chip) foram propostas como solução para interconexão de núcleos em sistemas integrados de alta densidade. Essas redes já são utilizadas pela indústria e isso irá resultar em uma demanda pela formação de recursos humanos aptos a lidar com essa tecnologia. O ensino de conceitos sobre NoCs precisa ser apoiado por ferramentas que permitam ao estudante explorar o seu espaço de projeto, sendo que existem diversas ferramentas com tal capacidade. Porém, para facilitar o uso dessas ferramentas no ensino, é importante que estas ofereçam recursos de apoio adicionais aos necessários para o uso em atividades de pesquisa. Neste sentido, este trabalho apresenta uma análise sobre ferramentas de NoC descritas na literatura e identifica quais recursos úteis para o ensino que são mais comumente oferecidos. Também identifica quais ferramentas oferecem maior quantidade desses recursos, assim como evidencia lacunas a serem preenchidas para facilitar e acelerar a adoção dessas ferramentas com vistas à melhoria do processo de ensino e aprendizagem sobre NoCs.

Palavras-chave — Redes-em-Chip; Ferramentas; Ensino.

I. INTRODUÇÃO

O avanço no processo de fabricação de circuitos aumentou a capacidade de integração e viabilizou a construção de sistemas computacionais completos em um único chip. Esses sistemas são denominados de sistemas integrados ou SoCs (Systems-on-Chip) e são compostos por elementos reutilizáveis chamados de núcleos ou IPs (de Intellectual Property).

As Redes-em-Chip (NoCs – Networks-on-Chip) emergiram da necessidade de uma arquitetura de comunicação intrachip que atendesse os requisitos de comunicação de SoCs com dezenas de núcleos integrados, os quais não seriam suportados pelas tecnologias tradicionais, como o barramento [1].

O termo NoC foi proposto por Hemani et al. [2] em 2000, mesmo ano em que foram apresentados resultados experimentais da primeira NoC descrita na literatura, a SPIN [3]. Nesses 15 anos, várias alternativas arquiteturais para implementação de NoCs foram propostas. Um levantamento sobre as propriedades de comunicação dos SoCs e sobre os vários aspectos arquiteturais a serem considerados no projeto de NoCs foi descrito em [4]. Já o *survey* apresentado em [5] discutiu os problemas e desafios no projeto de NoCs a partir da análise de sessenta trabalhos reportados na literatura.

Conforme evidenciado nos estudos supracitados, o espaço de projeto das NoCs é amplo e vários aspectos arquiteturais devem ser considerados para atender a requisitos de desempenho e custo do sistema alvo. Considerando o fato de que o projeto de um SoC deve também atender a requisitos de *time-to-market*, é necessário utilizar ferramentas especializadas para acelerar a exploração do espaço de projeto [6].

Considerando o problema supracitado, diversos ambientes de simulação e avaliação de NoCs têm sido desenvolvidos e reportados na literatura, conforme o levantamento sobre ferramentas para modelagem, simulação, síntese e/ou extração de métricas de desempenho de NoCs é apresentado em [7]. Essas ferramentas foram desenvolvidas para servirem de infraestrutura em atividades de pesquisa e/ou desenvolvimento.

As NoCs já são uma realidade industrial e diversas empresas reportam o seu uso em projetos e produtos, como a Ezchip [8], a Intel [9] e a Arteris [10]. Esta última provê uma tecnologia de NoC adotada por outras empresas, tais como Samsung, LG e Toshiba. Portanto, brevemente, o estudo de NoCs precisará ser introduzido de forma regular em cursos de graduação e de pós-graduação a fim de formar profissionais aptos a lidar com essa tecnologia. Aspectos relacionados à arquitetura de NoCs, projeto de roteadores e avaliação de desempenho, entre outros, podem ser abordados em disciplinas sobre Arquiteturas de Computadores, Arquiteturas Paralelas, Redes de Computadores e Redes-em-Chip. Porém, para isso, é preciso dispor de ferramentas que não apenas auxiliem na exploração do espaço de projeto em pesquisa e/ou desenvolvimento, mas que também facilitem e acelerem o ensino e a aprendizagem dos conceitos relacionados às NoCs.

É do entendimento dos autores deste artigo que a adoção de ferramentas por professores em disciplinas de graduação e/ou de pós-graduação é facilitado se a ferramenta dispor de um conjunto de materiais de apoio ao ensino e à aprendizagem, como aqueles oferecidos por editoras em complemento a livros didáticos. Exemplos de materiais de apoio desejáveis para uso dessas ferramentas, incluem: (i) guia rápido para instalação e uso; (ii) manual descritivo sobre seus recursos; (iii) vídeo demonstrativo do seu uso; (iv) material de aula em formato de *slides* para o professor apresentar a ferramenta; e (v) exercícios práticos para consolidação da teoria abordada na disciplina. Considerando os serviços disponíveis atualmente na Web, outros recursos de valia para ensino são aqueles que promovem a colaboração entre os usuários, tais como: (i) fórum de discussão; e (ii) sistemas de controle de versionamento.

Nesse sentido, este trabalho buscou analisar as ferramentas de NoCs descritas na literatura e responder à seguinte pergunta de pesquisa: “As ferramentas de NoC oferecem recursos para facilitar seu uso no ensino e quais são esses recursos?”. Embora já existam estudos que analisam e comparam ferramentas de NoCs (ex. [7]), pelo conhecimento dos autores, não existe na literatura trabalho anterior com foco similar ao deste artigo.

O texto que segue está organizado em quatro seções. A Seção II descreve as ferramentas selecionadas e a Seção III analisa as características dessas ferramentas, com foco nos recursos que poderiam auxiliar no ensino. Concluindo, a Seção IV apresenta as considerações finais do trabalho.

II. FERRAMENTAS DE REDES-EM-CHIP

No presente estudo, foi realizada uma busca não exaustiva sobre ferramentas de NoCs descritas na literatura publicadas nas bases de dados digitais do IEEE e da ACM. O estudo tomou como ponto de partida o *survey* apresentado em [7], o qual foi estendido para ampliar e atualizar o conjunto de artigos a serem analisados. A seguir, é apresentada uma breve descrição de cada ferramenta selecionada, com a identificação de suas características básicas e da existência de documentação ou material adicional para facilitar o seu uso.

O NS-2 [11] é um simulador de redes de computadores implementado em C++. A configuração é realizada com *scripts* OTcl e viabiliza a definição de topologias e enlaces de rede e outras características. Em [12], o NS-2 foi utilizado em estudo de caso de tolerância a faltas em NoCs. Os desenvolvedores da ferramenta disponibilizam um ambiente *web* com informações sobre o projeto, vários documentos sobre o ambiente e um espaço para submissão de propostas educacionais [11].

O Noxim [13] é um simulador de NoCs implementado em SystemC e executável por linha de comando. Inicialmente o simulador suportava apenas a topologia em malha. Em [14], foi adicionado suporte às topologias *torus* e *twisted-torus*. O simulador possui uma página no controle de versão GitHub que possui informações a seu respeito [13]. O GitHub é um serviço de hospedagem na Web para projetos que usam o controle de versionamento Git.

O Nirgam [15] é um simulador de NoC implementado em SystemC com acurácia em nível de ciclos. Possui suporte a *plug-ins* para expansão dos modelos. Inicialmente oferecia as topologias malha e *torus*, mas topologias irregulares foram incluídas em [16]. O simulador possui um ambiente *web* e conta com manual e documentação do código-fonte.

O DARSIM [17] é um simulador paralelo de NoC em nível de ciclos e altamente configurável. Ele utiliza um mecanismo de simulação paralela que permite realizar *tradeoffs* entre a perfeita acurácia e a velocidade de simulação com alta acurácia. A geometria, a largura de banda, as dimensões do *crossbar* e os níveis de pipeline são configuráveis.

O HORNET [18] é uma versão posterior do DARSIM [17]. Assim como o seu predecessor, ele é um simulador paralelo altamente configurável, possui acurácia em nível de ciclos e é baseado em uma arquitetura de roteador *wormhole* de filas de entrada. Possui uma página na Web [19] que disponibiliza o simulador e o manual de usuário.

O Sunfloor 3D [20] é uma ferramenta de projeto para síntese de NoCs com topologias 3D. Ela é capaz de buscar caminhos para os fluxos de comunicação, atribuir componentes da rede e colocá-los em suas devidas camadas. Métodos e algoritmos foram definidos para realizar a conexão dos componentes e foram feitos experimentos para validar os métodos propostos.

O Orion [21][22] é um simulador de redes de interconexão intrachip com foco na análise de consumo de energia com modelos de área e desempenho para avaliação. O projeto mantém página na Web [22] para *download* dos modelos e de relatórios técnicos.

O SUNMAP [23] é uma ferramenta de seleção e geração de topologias de NoCs de acordo com a aplicação. Realiza o mapeamento dos núcleos na topologia e busca a melhor relação de área para determinada configuração. Conta com modelos SystemC de simulação com acurácia em nível de ciclos para avaliação de desempenho.

O NoCGEN [24] é um gerador de NoC usado para criar descrições de NoCs simuláveis e sintetizáveis. Conta com um conjunto de componentes modularizados para diferentes configurações de roteadores, algoritmos de roteamento, largura do canal de dados e profundidade dos *buffers*. O modelo sintetizável é gerado em VHDL e o simulável em SystemC.

O BookSim 2.0 [25] é um simulador de redes de interconexão implementado em C++ com acurácia em nível de ciclos. Conta com alternativas de topologias, algoritmos de roteamento, controle de fluxo e microarquiteturas do roteador. Seu ambiente *web* [26] contém informações do projeto e guia do usuário. O código-fonte é versionado no GitHub.

O HeMPS [27] é um simulador MPSoC (Multi-Processor Systems-on-Chip) que utiliza uma NoC como infraestrutura de comunicação. Possui ferramentas para alocação e edição de tarefas, depuração e verificação do sistema. Memórias e microprocessadores são descritos em SystemC e a NoC em VHDL. Possui uma Wiki [28] com informações do projeto, guia inicial de uso e vídeo no YouTube.

O NoCScope [29] é uma ferramenta em Java com interface gráfica que realiza o monitoramento de tráfego dos fluxos de comunicação de um SoC baseado em NoC. Os níveis de avaliação foram classificados em escopos: fim-a-fim, ponto-a-ponto, pontos críticos, consumo de energia, *buffers*, entrada e saída.

O GSNOG UI [30] é uma ferramenta gráfica desenvolvida em Qt para análise de NoCs 3D e utiliza o simulador GSNOG [31], o qual é baseado em SystemC. Permite a configuração dos parâmetros arquiteturais da rede, opções de simulação, têm suporte a *benchmarks* e conta com gerador de grafo de tarefas e um monitor de tráfego em tempo de execução.

O gpNoCSim++ [32][33] é um simulador de NoC de propósito geral implementado em Java que conta com o projeto de componentes arquiteturais modulares. Têm suporte a expansão de topologias e demais componentes arquiteturais.

O gem5 [34] une aspectos de dois simuladores predecessores e provê um *framework* de simulação MPSoC altamente configurável. Possui maior parte dos modelos implementados em C/C++ e a configuração e o controle de simulação são feitos por meio de *scripts* em Python. O foco no simulador está nos aspectos arquiteturais do sistema em mais alto nível (processador, memória, tarefas). O simulador possui uma página de projeto [35] que conta com recursos de documentação como manuais, tutoriais e apresentações.

O TOPAZ [36] é um simulador de redes de interconexão de propósito geral que possibilita a modelagem de uma larga variedade de roteadores. Foi desenvolvido em C++, é orientado a objetos e possibilita a extensão e reuso dos componentes. Tem suporte à execução *multithread* e foi integrado à ferramenta gem5 [34]. É *open-source* e a página do projeto está hospedada no GitHub [37].

O HNOCS [38] é um simulador de NoCs *open-source* baseado em OMNET++[39], uma biblioteca e *framework* C++ para simuladores de redes. O HNOCS é modular, escalável, extensível e totalmente parametrizável para modelagem de NoCs heterogêneas. Possui uma página na Web [40] que inclui um fórum de questões e uma lista de discussão.

O HLS-DoNoC [41] é um simulador em nível de sistema para exploração organizacional dinâmica de NoCs na fase de projeto. Com a maior parte implementada em C++, é capaz de simular várias técnicas de monitoramento e reconfiguração dinâmica baseadas em *clusters*.

O McSimA+[42] é um simulador desenvolvido em C++ que permite a simulação em nível de sistema ou de aplicação. Suporta a arquitetura x86 e possui um conjunto de componentes como memória cache, RAM, sistema de interconexão (barramento, *crossbar* e NoCs), além de suporte à clusterização. O projeto possui uma página na Wikipedia [43].

No PolyNOC [44], foi usado um modelo de satisfação para capturar a experiência do usuário em aplicações multimídias com uma ferramenta de simulação de sistema embarcado baseado em NoC. O simulador foi implementado em C++ e a NoC foi baseada numa topologia em malha. O foco deste trabalho é um pouco diferente dos demais, mas utiliza uma NoC para avaliar aplicações multimídia.

Em [45], foi proposto um simulador de NoC parametrizável com interface gráfica em Qt e simulador em SystemC. O simulador é capaz de tratar cargas de trabalho baseadas em traços com suporte à alocação de tarefas para exploração de aplicações específicas com NoCs.

Em [46], foi proposto um método de simulação que pode calcular a temporização em nível de ciclos com chaveamento *wormhole*. A acurácia obtida no modelo proposto foi a mesma de uma implementação RTL, enquanto o tempo de simulação foi mantido tão rápido tanto quanto uma descrição TLM.

O BrownPepper [47] é um ambiente de avaliação de desempenho de NoC que possui acurácia em nível de ciclos. O simulador foi implementado em SystemC e a ferramenta conta com interface gráfica em GTK+ para a configuração dos parâmetros e análise dos resultados.

O RedScarf [48], uma nova versão do BrownPepper [47], é um ambiente multiplataforma com suporte à execução *multithread*. Conta com interface gráfica em Qt e recursos que facilitam a integração de novos componentes arquiteturais. Embora possua um manual de usuário, a ferramenta ainda não está disponível publicamente.

III. ANÁLISE

Apenas metade das ferramentas analisadas é disponível ao público. A Tabela I apresenta um sumário sobre os recursos oferecidos pelas doze ferramentas públicas e responde à pergunta de pesquisa que motivou este estudo. Apresenta-se a seguir uma análise sobre os resultados obtidos na pesquisa.

Como é de se esperar, a maior parte das ferramentas públicas possui página na Web com informações e *links* para *download* de documentos e arquivos de instalação, sendo que duas utilizam apenas o GitHub, provavelmente para concentrar

as informações em um único local. Apenas uma ferramenta não oferece uma página com informações e documentos, mas seus arquivos de instalação são disponibilizados em repositório *web*.

É comum que ferramentas ofereçam um guia rápido para instalação e uso. Porém, pouco mais da metade das ferramentas públicas analisadas o faz. Outras incluem as orientações para instalação no manual de usuário, sendo que o grau de detalhamento dos manuais dessas ferramentas varia muito.

Poucas ferramentas disponibilizam um vídeo demonstrativo e apenas duas oferecem material em formato de *slides* para apresentação dos seus recursos em sala de aula. Essas ferramentas são a gem5 e a NS-2, sendo que esta última é a única que oferece roteiros para atividades práticas. Isso provavelmente pode ser atribuído pelo fato de ser uma ferramenta mais madura e de escopo mais amplo, já que é um simulador de redes de computadores.

Algumas ferramentas oferecem um fórum de discussão ou utiliza de sistemas de controle de versionamento *on-line* para colaboração entre seus usuários, sendo que foram identificadas quatro ferramentas que oferecem os dois recursos (gem5, TOPAZ, HEMPS e Noxim).

IV. CONCLUSÕES

Este artigo apresentou um estudo sobre ferramentas de NoCs com foco na análise da disponibilidade de recursos para apoio ao ensino. Observou-se que há ferramentas que oferecem recursos que podem facilitar o seu uso no ensino, porém, poucas disponibilizam material instrucional que, no entendimento destes autores, é essencial para facilitar e acelerar a adoção de uma ferramenta em atividades de ensino. Acredita-se que a falta de tais recursos decorre de o fato dessas ferramentas terem sido desenvolvidas para uso em pesquisa.

Os autores deste estudo são responsáveis pelo desenvolvimento da ferramenta RedScarf. Com base na pesquisa realizada, como trabalho futuro, pretende-se produzir um conjunto de recursos de apoio focados no ensino para preencher a lacuna observada e, então, disponibilizá-la publicamente. Também pretende-se aplicar e avaliar a ferramenta e o material produzido na disciplina “Redes-em-Chip” do Programa de Pós-Graduação em Computação da Univali e então reportar os resultados à comunidade acadêmica de forma a propor o seu uso em outras instituições de ensino.

AGRADECIMENTOS

Este trabalho foi apoiado pelo PROSUP/CAPES.

TABELA I. RECURSOS OFERECIDOS PELAS FERRAMENTAS DE NOCS DISPONÍVEIS AO PÚBLICO

Recursos	Ocorrências	NS-2	gem5	TOPAZ	HeMPS	Noxim	BookSim	Nirgam	HNOCS	McSimA+	HORNET	Orion	gpNoCsim++
Página na Web	11 (92%)	X	X	CV	X	CV	X	X	X	X	X	X	
Guia rápido	7 (58%)	X	X	X	X	X			X	X			
Manual de usuário	7 (58%)	X	X	X		X	X	X			X		
Vídeo demonstrativo	4 (33%)	YT	X		X			YT					
Slides para aula	2 (17%)	X	X						2				
Roteiros para atividades práticas	1 (8%)	X											
Forum de discussão	6 (50%)	X	X	X	X		X		X				
Controle de versionamento	6 (50%)		X	X	X	X	X			X	X		
Quantidade de Recursos		7	7	5	5	4	4	3	3	3	2	1	0

Obs: CV: Utiliza sistema *web* de controle de versionamento para hospedar a página do projeto
YT: Vídeo disponibilizado no YouTube, mas não referenciado na página do projeto

REFERÊNCIAS

- [1] C. A. Zeferino, "Redes-em-Chip: Arquiteturas e modelos para avaliação de área e desempenho," Tese de Doutorado, Universidade Federal do Rio Grande do Sul, 2003.
- [2] A. Hemani, A. Jantsch, S. Kumar, A. Postula, J. Oberg, M. Millberg e D. Lindqvist, "Network on chip: An architecture for billion transistor era," In *Proc. of the IEEE NorChip Conf.*, vol. 31, 2000.
- [3] P. Guerrier e A. Greiner, "A generic architecture for on-chip packet-switched interconnections," *Proc. of the Conf. on Design, Automation and Test in Europe*. ACM, 2000.
- [4] T. Bjerregaard e S. Mahadevan, "A survey of research and practices of Network-on-Chip," *ACM Computing Surveys (CSUR)* 38.1 (2006): 1.
- [5] A. Agarwal, C. Iskander e R. Shankar, "Survey of network on chip (noc) architectures and contributions," *J. of Engineering, Computing and Architecture*, vol. 3, n. 1, 2009, p. 21-27.
- [6] L. Benini e G. De Micheli. *Networks on chips: technology and tools*. Academic Press, 2006.
- [7] A. B. Achballah e S. B. Saoud, "A survey of network-on-chip tools," *Int. J. of Advanced Computer Science and Applications*. vol. 4, n. 9, p. 61-67, 2013.
- [8] EZchip. *TILE-Gx72 Processor: Product Brief*. 2013.
- [9] M. Gries, U. Hoffmann, M. Konow e M. Riepen, "SCC: A Flexible Architecture for Many-Core Platform Research," in *Computing Science Engineering*, vol. 13, n. 6, 2011, p. 79-83.
- [10] J.-J. Lecler e G. Baillieu, "Application driven network-on-chip architecture exploration & refinement for a complex SoC," *Design Automation for Embedded Systems*, vol. 15, n. 2, p. 133-158, 2011.
- [11] *The Network Simulator - ns-2*. Disponível em: <http://www.isi.edu/nsnam/ns/>
- [12] M. Ali, M. Welzl, A. Adnan e F. Nadeem, "Using the Ns-2 network simulator for evaluating network on chips (NoC)," In *Emerging Technologies, 2006. ICET'06. Int. Conf. on*, p. 506-512. IEEE, 2006.
- [13] *Noxim: Network on Chip Simulator*, 2015. Disponível em: <https://github.com/davidepatti/noxim>.
- [14] K. Swaminathan, D. Thakyal, S. G. Nambiar, G. Lakshminarayanan e S. Ko, "Enhanced Noxim simulator for performance evaluation of network on chip topologies," In *Engineering and Computational Sciences (RAECS), 2014 Recent Advances in*, p. 1-5. IEEE, 2014.
- [15] NIRGAM: a Simulator for NoC Interconnect Routin Application Modeling. Disponível em: <http://nirgam.ecs.soton.ac.uk/>.
- [16] N. Choudhary, M. S. Gaur e V. Laxmi, "Irregular NoC Simulation Framework: IrNIRGAM," In *Int. Conf. on Emerging Trends in Networks and Computer Communications (ETNCC)*. 2011, p. 1-5. IEEE, 2011.
- [17] M. Lis, K. S. Shim, M. H. Cho, P. Ren, O. Khan e S. Devadas, "DARSIM: a parallel cycle-level NoC simulator," In *MoBS 2010-Sixth Annual Wksp. on Modeling, Benchmarking and Simulation*. 2010.
- [18] P. Ren, M. Lis, M. H. Cho, K. S. Shim e C. W. Fletcher, "HORNET: A Cycle-Level Multicore Simulator," In *IEEE Transactions on Computer-aided design of integrated circuits and systems*. vol. 31, n. 6, 2012, p. 890-903.
- [19] *HORNET*, 2011. Disponível em: <http://csg.csail.mit.edu/hornet/>.
- [20] C. Seiculescu, S. Murali, L. Benini, e G.D. Micheli, "SunFloor 3D: A Tool for Networks on Chip Topology Synthesis for 3-D Systems on Chips", *J. IEEE Trans. on CAD of Integrated Circuits and Systems*, 2010, p.1987-2000.
- [21] A. B. Kahng, B. Li, L. Peh e K. Samadi, "Orion 2.0: A power-area simulator for interconnection networks," In *IEEE Transactions on Very Large Scale Integration (VLSI) systems*, 2012, p. 191-196.
- [22] *Orion: A Power-Performance Simulator for Interconnection Networks*, 2012. Disponível em: <http://projects.csail.mit.edu/cgi-bin/wiki/view/LSPgroup/OrionPage>.
- [23] S. Murali e G. De Micheli, "SUNMAP: a tool for automatic topology selection and generation for NoCs," In *Proc. of the 41st annual Design Automation Conf.*, ACM, 2004, p. 914-919.
- [24] J. Chan, S. Parameswaran, "NoCGEN: a template based reuse methodology for Networks-on-Chip architecture," In *Proc. of 17th Int. Conf. on VLSI Design*, 2004, p. 717-720.
- [25] N. Jiang, D. U. Becker, G. Michelogiannakis, J. Balfour, B. Towles, J. Kim e W. J. Dally, "A Detailed and Flexible Cycle-Accurate Network-on-Chip Simulator," In *Proc. of the 2013 IEEE Int. Symp. on Performance Analysis of Systems and Software*, 2013.
- [26] *BookSim Interconnection Network Simulator*. Disponível em: <http://nocs.stanford.edu/cgi-bin/trac.cgi/wiki/Resources/BookSim>.
- [27] E. A. Carara, R. P. de Oliveira, N. L. V. Calazans, F. G. Moraes, "HeMPS – A Framework for NoC-Based MPSoC Generation," In *IEEE Int. Symp. on Circuits and Systems (ISCAS)*, 2009. p. 1345-1348.
- [28] *HeMPS*, 2011. Disponível em: <https://corfu.pucrs.br/redmine/projects/hemps/wiki>.
- [29] L. Möller, L. S. Indrusiak e M. Glesner, "NoCSCOPE: A graphical interface to improve Networks-on-Chip monitoring and design space exploration," In *4th Int. Design and Test Wksp. (IDT)*, 2009. p.1-6.
- [30] P. Gottschling, H. Ying e K. Hofmann, "GSNOG UI – A Comfortable Graphical User Interface for Advanced Design and Evaluation of 3-Dimensional Scalable Networks-on-Chip," In *Int. Conf. on High Performance Computing and Simulation (HPCS)*, 2012. p. 261-267.
- [31] H. Ying, A. Jaiswal, M. A. Abd El Ghany, T. Hollstein, and K. Hofmann, "A Simulation Framework for 3-Dimension Networks-on-Chip with Different Vertical Channel Density Configurations," *IEEE DDECS*, Estonia, 2012.
- [32] H. Hossain, M. Ahmed, A. Al-Nayem, T. Z. Islam e M. M. Akbar, "GpnoCSim - A General Purpose Simulator for Network-On-Chip," In *Int. Conf. on Information and Communication Technology, 2007. ICT'07, 2007*. p.254-257.
- [33] *gpNoCSim++*, 2013. Disponível em: <http://sourceforge.net/projects/gpnoCSimpp/>.
- [34] N. Binkert et al., "The gem5 simulator," in *ACM SIGARCH Computer Architecture News*, vol. 39, n. 2, 2011. p. 1-7.
- [35] *The gem5 Simulator*, 2015. Disponível em: <http://gem5.org/>.
- [36] P. Abad, P. Prieto, L. G. Menezes, A. Colaso, V. Puente e J.-A. Gregorio, "TOPAZ: Na Open-Source Interconnection Network Simulator for chip Multiprocessors and Supercomputers," In *Sixth IEEE/ACM Int. Symp. on Networks-on-Chip*, 2012. p. 99-106.
- [37] *TOPAZ*, 2015. Disponível em: <https://github.com/abadp/tpzsimul>.
- [38] Y. Ben-Itzhak, E. Zahavi, I. Cidon e A. Kolodny, "HNOCS: Modular open-source simulator for Heterogeneous NoCs," In *Int. Conf. on Embedded Computer Systems (SAMOS)*, 2012. p. 51-57.
- [39] *OMNet++: Discrete Event Simulator*, 2015. Disponível em: <http://omnetpp.org/>.
- [40] *HNOCS: Modular Open-Source Simulator for Heterogeneous NoCs*, Disponível em: <http://hnocs.eew.technion.ac.il/>.
- [41] L. Guang, E. Nigussie, J. Plosila, J. Isoaho e H. Tenhunen, "HLS-DoNoC: High-Level Simulator for Dynamically Organizational NoCs," In *IEEE 15th Int. Design and Diagnostics of Electronic Circuits & Systems (DDECS)*, 2012. p. 89-94.
- [42] J. H. Ahn, S. Li, S. O e N. P. Jouppi, "McSimA+: A Manycore Simulator with Application-level+ Simulation and Detailed Microarchitecture Modeling," In *IEEE Int. Symp. on Performance Analysis of Systems and Software (ISPASS)*, 2013, p. 74-85.
- [43] *McSimA+*, 2014. Disponível em <https://en.wikipedia.org/wiki/McSimA%2B>.
- [44] S. D. Ponpandi, Z. Zhang e A. Tyagi, "PolyNOC – A Polymorphic thread simulator for NOC communication based embedded systems," In *Int. Conf. on Reconfigurable Computing and FPGAs (ReConFig)*, 2013. p. 1-8.
- [45] D. Ghosh, P. Ghosal e S. P. Mohanty, "A Highly Parameterizable Simulator for Performance Analysis of NoC Architectures," In *Int. Conf. on Information Technology (ICIT)*, 2014. p. 311-315.
- [46] T.-S. Hsu, J.-L. Chiu, C.-K. Yu e J.-J. Liou, "A Fast and Accurate Network-on-Chip Timing Simulator with a Flit Propagation Model," In *20th Asia and South Pacific Design Automation Conf. (ASP-DAC)*, 2015, p. 797-802.
- [47] C. A. Zeferino, J. V. Bruch e M. R. Pizzoni, "BrownPepper: a SystemC-based Simulator for Performance Evaluation of Networks-on-Chip," In *17th Int. Conf. On Very Large Scale Integration (VLSI-SOC)*, 2009. p. 223-226.
- [48] E. A. Silva, "RedScar: ambiente para avaliação de desempenho de Rede-em-Chip," Trabalho Técnico Científico de Conclusão de Curso, Universidade do Vale do Itajaí, 2014.