

Uma Nova Abordagem de Roteamento em NoCs Com Topologias Irregulares

A New Routing Approach in NoCs with Irregular Topologies

Samuel da Silva Oliveira¹
Monica Magalhães Pereira²
Márcio Eduardo Kreutz³

Resumo: Redes-em-Chip tem se tornado uma viável forma de comunicação para sistemas em chip. Para se obter uma comunicação satisfatória, ter uma boa forma de organizar a rede pode ser uma maneira para se conseguir esse resultado. Pensando nisso o uso de topologias irregulares pode ser uma opção para se conseguir otimizações mais satisfatórias, uma vez que permite, por concepção, a conexão de qualquer roteador com quaisquer outros. Nas topologias irregulares, devido à sua natureza, são necessários alguns artifícios para os pacotes percorrerem a rede do nodo origem até o destino. Para tal, normalmente se utilizam tabelas de roteamento em cada roteador. Nesse artigo é proposto um roteamento para topologias irregulares onde o caminho é armazenado dentro do flit de cabeçalho. Essa abordagem é comparada com a abordagem tradicional, a qual utiliza tabelas para guardar o caminho dos pacotes. Os resultados mostram que para valores de latência a abordagem proposta neste trabalho consegue obter resultados mais otimizados.

Palavras-chave: Redes-em-Chip; Roteamento; Topologias Irregulares.

Abstract: *Network-on-Chip has become a viable form of communication for systems. In order to obtain a satisfactory communication, having a good way to organize the network can be a way to achieve this result. Thinking about this, the use of irregular topologies may be an option to get more satisfactory optimizations, since it allows, by design, the connection of any router to any others. In irregular topologies, due to their nature, some artifacts are required for packets to travel through the network from the source node to the destination. Routing tables are usually used on each router. In this article we propose a routing for irregular topologies that uses a form of path storage within the flits themselves and compare this new approach with routing that uses tables to store the packet path. The results show that for latency values the approach proposed here is able to obtain better results.*

Keywords: *Irregular Topology; Network-on-Chip; Routing.*

¹ Mestrando em Sistemas e Computação na linha de pesquisa de Sistemas Integrados. Universidade Federal do Rio Grande do Norte.

{s.oliveira.15@hotmail.com}

² Doutora em Ciências da Computação. Universidade Federal do Rio Grande do Norte.

{monicapereira@dimap.ufrn.br}

³ Doutor em Ciências da Computação Universidade Federal do Rio Grande do Norte.

{kreutz@dimap.ufrn.br}

1 INTRODUÇÃO

Com a diminuição do tamanho dos transistores e com o crescente aumento do uso de sistemas em chip (*System-on-Chip - SoC*), surgiram os sistemas em chip multiprocessados (*Multiprocessor System-on-Chips - MP-SoCs*) (Benini e De Micheli, 2002), onde dentro de uma pastilha de silício podem ser alocados vários elementos de processamento (cores), memórias, dentre outros elementos que precisam se comunicar entre si. Para esse tipo de sistemas, barramentos tendem a perder a eficiência devido ao tamanho dos fios e compartilhamento temporal das trocas de mensagens. Nesse contexto, foi proposto um novo paradigma de comunicação chamado Redes-em-Chip (*Network-on-Chip - NoC*) (Jantsch et al., 2003)(Agarwal; Iskander; Shankar, 2009) como alternativa para suprir essas dificuldades temporais na comunicação.

Uma NoC consiste de um conjunto de roteadores interligados através de chaveamento, formando assim o que é chamado de topologia, afim de permitir a troca de mensagens com um certo grau de paralelismo. A topologia de redes-em-chip é um dos fatores determinantes para que um pacote trafegado consiga chegar ao seu destino, respeitando restrições da aplicação alvo.

Existem vários tipos de topologias que servem para diferentes tipos de projetos. Uma topologia regular serve bem de forma genérica quando não se conhece à priori a aplicação distribuída cujas tarefas serão mapeadas para a rede, quantos cores tem a aplicação ou a forma de comunicação da mesma. Topologias irregulares (Tosun; Ar; Ozdemir, 2012)(Dally e Towles, 2004) podem ser uma boa opção quando já se conhece o grafo da aplicação que será utilizado. Sendo assim, sabe-se quantos roteadores irão precisar, não sendo utilizados roteadores a mais que o necessário, como geralmente ocorre em uma topologia regular. Dessa forma, pode-se conseguir reduções na área de um chip. Outra vantagem diz respeito à latência, já que em uma topologia irregular muitas vezes a maioria dos roteadores estão interconectados, minimizando o número de roteadores intermediários entre um par fonte-destino. Mais uma vantagem aparece quando se utilizam aplicações com tarefas de tempo real, onde prazos estritos precisam ser cumpridos. Isso ocorre devido ao potencial de redução de latência, como dito acima.

Um fator importante em uma topologia irregular é a forma de como rotear os pacotes, já que formas de roteamento convencionais não funcionam nesse tipo de topologia. Na literatura existem vários trabalhos que pesquisam formas de roteamento para topologias irregulares. Dentre essas formas, a mais conhecida se dá através de tabelas de roteamento. Neste trabalho é desenvolvida uma forma de roteamento que difere um pouco das formas encontradas na literatura. No método de roteamento, o caminho que cada flit deve percorrer é armazenado dentro do corpo do próprio flit. Na abordagem proposta, espera-se ter valores de latência médias menores, uma maior taxa de entrega de pacotes tempo real e uma menor área consumida. Para isso é realizada a comparação com topologias irregulares que utilizam tabelas de roteamento.

Esse artigo está organizado da seguinte forma: na segunda seção são apresentados os trabalhos relacionados, onde são mostrados algumas formas de roteamento em topologias irregulares usadas na literatura; na terceira seção são apresentadas, a motivação do trabalho e a descrição do problema, onde é explanado o porquê de se procurar novas formas de roteamento; na quarta seção é explicado como os experimentos foram realizados, mostrando assim a metodologia utilizada nessa pesquisa; na quinta seção, tem-se os resultados obtidos nessa pesquisa; a sexta seção conclui a proposta, apontando alguns possíveis trabalhos futuros.

2 TRABALHOS RELACIONADOS

Ajabshir e Tosun (2014) propõem um roteamento tolerante a falhas para uma topologia irregular. A proposta é de um roteamento baseado em tabelas que possa cobrir futuras falhas ou falhas que ocorram durante a fabricação de um chip. O roteamento apresentado em (Ajabshir e Tosun, 2014) deve funcionar quando utilizada a topologia dada em (Tosun et al., 2014), que é uma topologia irregular tolerante a falhas. Se alguma falha ocorrer em algum link é feito um cálculo para medir o nível de importância daquele link pela largura de banda e a quantidade de tráfego que passará nele. Caso ocorra algum problema no chip, o roteamento ativará links extras baseados nos custos de cada mensagem e nas tabelas de roteamento (TR) de cada roteador.

Moraveji; Sarbazai-Azad; Zomaya (2009) propõem uma metodologia geral para o desenvolvimento de algoritmos de roteamento para topologias irregulares. São propostos três novos algoritmos de roteamento e também é realizada a exploração de algoritmos já conhecidos e utilizados na literatura. São feitos diversos testes em seis algoritmos de roteamento, onde três desses são novos e propostos pelo autor e os outros já bem utilizados no meio acadêmico. Os resultados mostram que dependendo do cenário que a rede se encontra e como a topologia está arranjada, os algoritmos de roteamento propostos no trabalho tendem a serem superiores aos já existentes.

Duan; Zhang; Sun (2009) propõem um roteamento irregular para garantir *deadlock free* em topologias irregulares que são baseadas em redes mesh. O algoritmo proposto conta com o uso de alguns canais virtuais na rede para o auxiliarem no processo de roteamento. O intuito é garantir que não haja *deadlock* em uma NoC com elementos de processamento heterogêneos e que sejam ligados através de uma topologia irregular. O algoritmo funciona independentemente do tamanho e da estrutura da topologia mesh irregular. Os resultados obtidos mostraram-se satisfatórios com o uso do método proposto.

Xiao et al. (2013) também trabalha com MP-SoCs heterogêneos e topologias irregulares para realizar a comunicação na NoC. É proposto então uma técnica de roteamento baseado em tabelas para realizar a transição dos pacotes dentro da rede. O roteamento também conta com plataformas virtuais abertas e foi projetado usando SystemC TLM. Os experimentos foram simulados usando uma rede mesh irregular 16x16 e aplicações reais. Os resultados mostram eficiência na conectividade da rede e entrega dos pacotes.

Nesse trabalho é proposto um método de roteamento para topologias irregulares onde todo o percurso do pacote fica armazenado no cabeçalho do flit. Nessa pesquisa preocupa-se com a taxa de latência média, taxa de entrega dos pacotes e também o quanto em área será aumentada usando o método proposto. Para efeito de comparação e desempenho iremos também utilizar uma forma de roteamento baseado em tabelas, assim podemos verificar a eficiência da abordagem proposta.

3 MOTIVAÇÃO E DESCRIÇÃO DO PROBLEMA

Ter uma boa comunicação em um MP-SoC é indispensável para o bom funcionamento do mesmo, principalmente quando é trabalhado com aplicações que executam tarefas de tempo real (*Real-Time - RT*). Essas tarefas demandam uma baixa latência, para que assim possam cumprir o prazo de entrega dos pacotes RT. Para então se conseguir essa diminuição na latência e maiores chances do cumprimento do deadline o mais indicado seria o uso de topologias irregulares (Tosun; Ar; Ozdemir, 2012), já que a ausência de alguns roteadores intermediários pode ajudar aos pacotes chegarem mais rapidamente ao seu destino.

Outro fator importante e que contribui para atrasos dos pacotes é a forma de roteamento dos mesmos (Zeferino, 2003). Devido a natureza da topologia irregular, formas comuns de roteamento não podem ser utilizadas, por isso se faz um uso de roteamentos especiais. Na literatura comumente encontramos roteamentos baseados em tabelas, onde cada roteador possui uma tabela que guarda o caminho para todos os roteadores na topologia ou os roteadores vizinhos. Quando um pacote entra no roteador ele consulta a tabela para saber por qual porta ele deverá ser encaminhado.

Roteamentos baseados em tabelas podem levar a um incremento considerável da área dos roteadores, já que as tabelas ocupam espaços dentro dos mesmos; quanto maior for a tabela, maior será a área do roteador. Considerando essa situação, este trabalho tem como objetivo, verificar e propor uma abordagem de roteamento, a qual consegue diminuir a latência média da rede sem, no entanto, implicar em um aumento significativo da área ocupada pelo circuito integrado. Além disso, espera-se aumentar a taxa de entrega de pacotes tempo real. Pensando nisso, buscou-se a criação de um novo método de roteamento, onde toda a informação do caminho de tráfego dos dados fosse armazenada dentro dos flits de cada pacote. Dessa forma, além de não implicar em um grande aumento em área, um ganho em latência pode ser conseguido, uma vez que o tempo de acesso às tabelas de roteamento não será somado à latência.

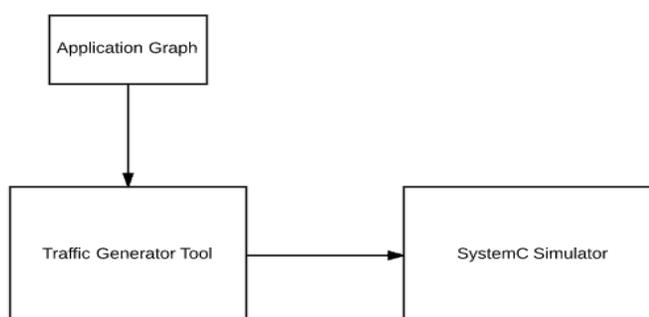
4 DESCRIÇÃO DOS EXPERIMENTOS

Para esse estudo foram criados 4 grafos de aplicações sintéticas utilizando a ferramenta TGFF (Dick; Rhodes; Wolf, 1998). A ferramenta cria grafos através de alguns parâmetros configurados em um arquivo de extensão .tgffopt. Os grafos foram criados para representar 4 aplicações com tarefas de tempo real, para que possam-se executar os testes.

Também foi desenvolvida uma ferramenta geradora de tráfego, usando a linguagem C++, onde tem-se como entrada o grafo da aplicação, após essa entrada é gerado um arquivo com as configurações de quantidade de pacotes a serem enviados e prazo de deadline (dado em nanossegundos), além do padrão de comunicação referente ao grafo dado.

Além dessa ferramenta, foram desenvolvidos dois simuladores em SystemC TLM (Ghenassia, 2006)(Zeferino et al, 2007) que são usados para simular a comunicação em uma NoC de topologia irregular. A comunicação leva em conta aplicações com tarefas de tempo real. No arquivo de entrada do simulador é passado o deadline dos pacotes e o simulador se baseia nisso para parar a execução da comunicação. As ferramentas são completamente modularizadas, sendo todas chamadas através da função main, com isso consegue-se obter uma facilidade para substituir qualquer componente da rede. Um algoritmo de menor caminho é utilizado para auxiliar no roteamento, assim traçando o caminho que um pacote deve seguir a partir de um nodo fonte até o seu destino. Na Figura 1 pode ser observado o diagrama e a conexão entre as ferramentas de geração de tráfego e de comunicação da NoC.

Figura 1: Diagrama de Funcionamento do Simulador



Fonte: Autoria Própria

Ambos os simuladores usam arbitragem Round-Robin, podem possuir até 4 portas para comunicação entre os roteadores e ainda conta com mais uma porta local para a comunicação com elemento de processamento (EP), comutação por pacotes e chaveamento *wormhole*. A única diferença entre esses simuladores é a forma do roteamento, onde o primeiro utiliza um roteamento a base de tabelas, onde as tabelas são formadas assim que é passado o arquivo de entrada contendo o grafo da aplicação, e os nodos fonte e destino. A tabela contém 3 campos, o primeiro campo é o core de destino, o segundo é a porta de saída que o pacote deverá tomar e o terceiro é a quantidade de hops, ou seja, quantos saltos aquele pacote levará para chegar ao destino por aquela porta de saída. A escolha da porta é feita tomando como base o campo da tabela com o core destino. Cada roteador contém em sua tabela apenas o menor caminho entre ele e os outros roteadores da topologia. Na Tabela 1 abaixo tem-se um exemplo de como é a Tabela usada no roteamento dos pacotes.

Tabela 1: Exemplo de Tabela de Roteamento

Destino	Porta	Hops
1	LOCAL	0
2	NORTE	1
3	OESTE	3
4	NORTE	1

Fonte: Aatoria Própria

O segundo simulador usa uma forma de roteamento onde todo o caminho que o pacote deve percorrer se encontra dentro do corpo de cada flit contido no pacote. Através do grafo de entrada os nodos origem e destino são lidos para se calcular o caminho que cada pacote deverá realizar até o seu destino. Esse caminho é colocado no flit em fila, onde o elemento frontal dessa fila corresponde ao próximo roteador que o flit terá que percorrer. Assim que o flit avança para o próximo roteador no caminho até o destino, esse valor é eliminado da fila. A operação se repete até que a fila acabe, o que significa que o flit chegou ao seu destino sendo então encaminhado à porta local do roteador destino. A Figura 2 exemplifica o corpo de um flit utilizado nessa abordagem.

Figura 2: Pacote Com Caminho de Tráfego

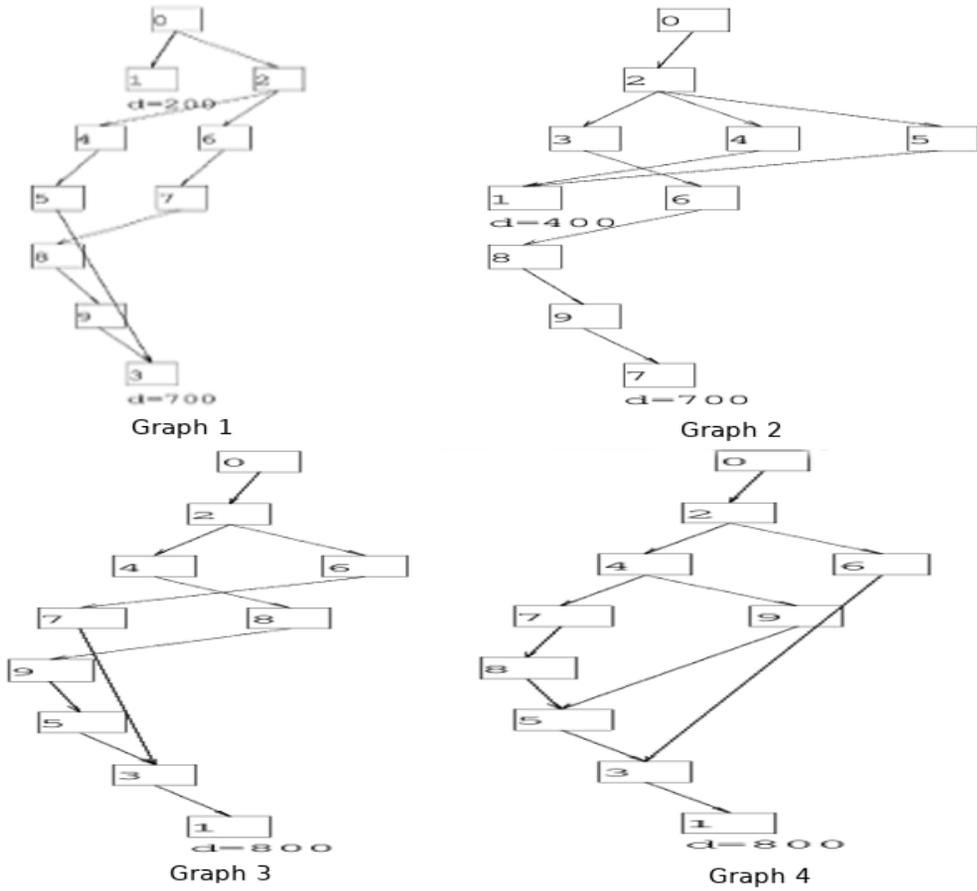
Caminho	Destino	Payload	Tipo
1 4 5 2 7	7	0101110001	2

Fonte: Aatoria Própria

No flit tem-se o campo do caminho, onde pode-se ver uma fila contendo todo o percurso que os pacotes farão. Tem-se o campo destino que diz pra que roteador esse pacote está sendo enviado. O campo payload é o responsável por conter os dados que estão sendo trafegados. E no campo tipo tem-se a indicação se aquele flit é um flit de cabeçalho, dados ou terminador. O roteamento utiliza o algoritmo de *Floyd-Warshall* para verificar o grafo da aplicação e escolher o caminho mínimo para as duas abordagens de roteamento.

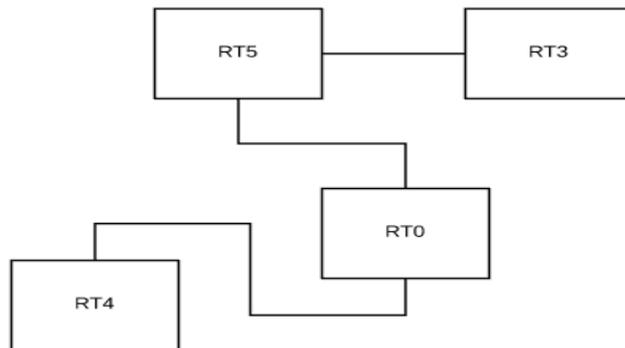
Foram testadas 4 aplicações com tarefas de tempo real, cada uma contendo 10 tarefas. Os grafos das aplicações que foram utilizados nesse trabalho foram gerados usando a ferramenta TGFF (Dick; Rhodes; Wolf, 1998). Os grafos foram gerados de forma que os comportamentos das aplicações sofressem variações em relação ao número de comunicações entre elas. Os testes foram feitos em uma topologia irregular previamente definida, uma vez que o foco do trabalho consiste na avaliação da eficiência de roteamento da abordagem proposta. Desse modo, não se fazem necessários mapeamentos específicos dos roteadores na rede. Na Figura 3 pode-se observar os grafos das aplicações e na Figura 4 um exemplo da topologia irregular como a utilizada nesse trabalho. Cada roteador é interligado a no máximo dois outros roteadores. Os roteadores pares se ligam com outros roteadores através da sua porta norte com a porta sul do outro roteador, já os roteadores ímpares se ligam através da sua porta leste com a porta oeste do outro roteador.

Figura 3: Grafos das Aplicações utilizadas



Fonte: Autoria Própria

Figura 4: Exemplo do Modelo de Topologia Irregular Utilizado



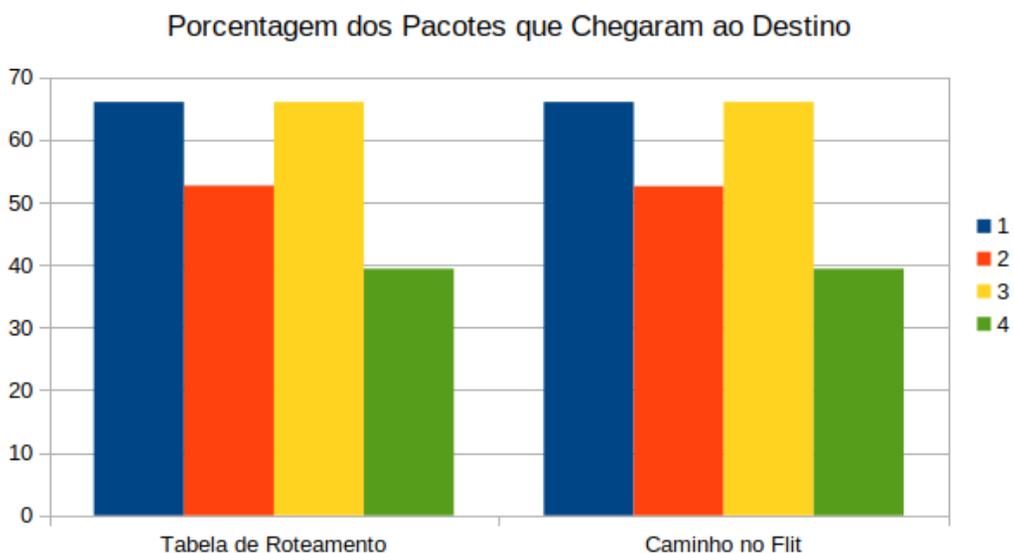
Fonte: Autores

5 RESULTADOS

Os experimentos e simulações foram feitos com 4 aplicações, mostradas na Figura 3. Também foi utilizado o T Test para uma análise estatística e para realizar a comparação entre os resultados. Com esse teste pode-se observar se houve ou não uma grande mudança entre os dados analisados.

Os resultados das comparações podem ser vistos nas Figuras abaixo e os testes estatísticos nas Tabelas abaixo. Cada comparação pode ser observada em gráfico que mostra o desempenho referente as duas abordagens testadas para cada aplicação. As Tabelas por sua vez mostram os resultados estatísticos de comparações entre os conjuntos de dados.

Figura 5: Porcentagem dos Pacotes que Chegaram ao Destino



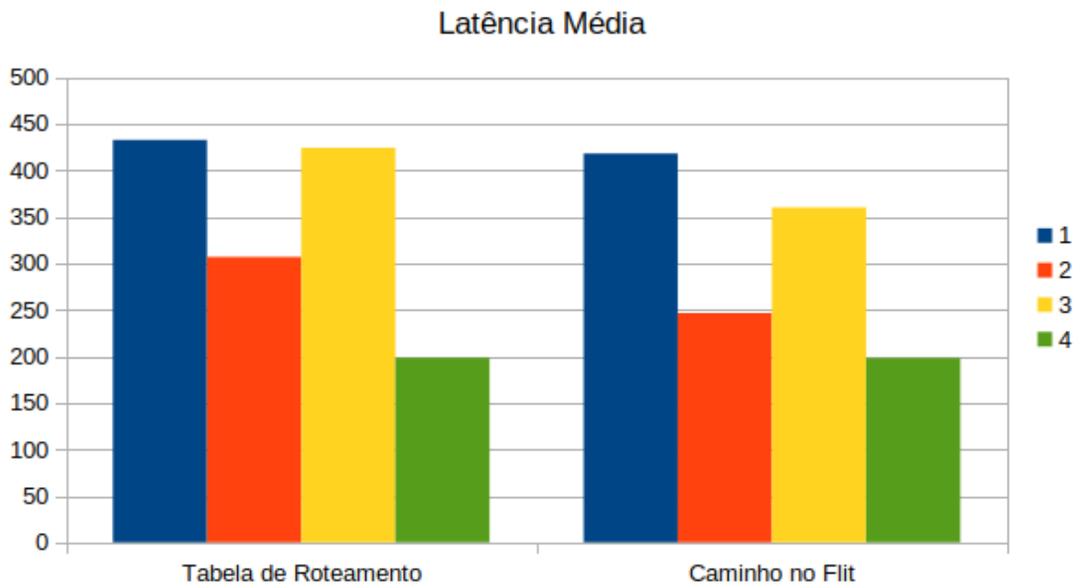
Fonte: Dados da Pesquisa

Tabela 2: Teste T - Porcentagem de Pacotes que Chegaram ao Destino

T Test	0,391
--------	-------

Fonte: Dados da Pesquisa

Figura 6: Latência Média



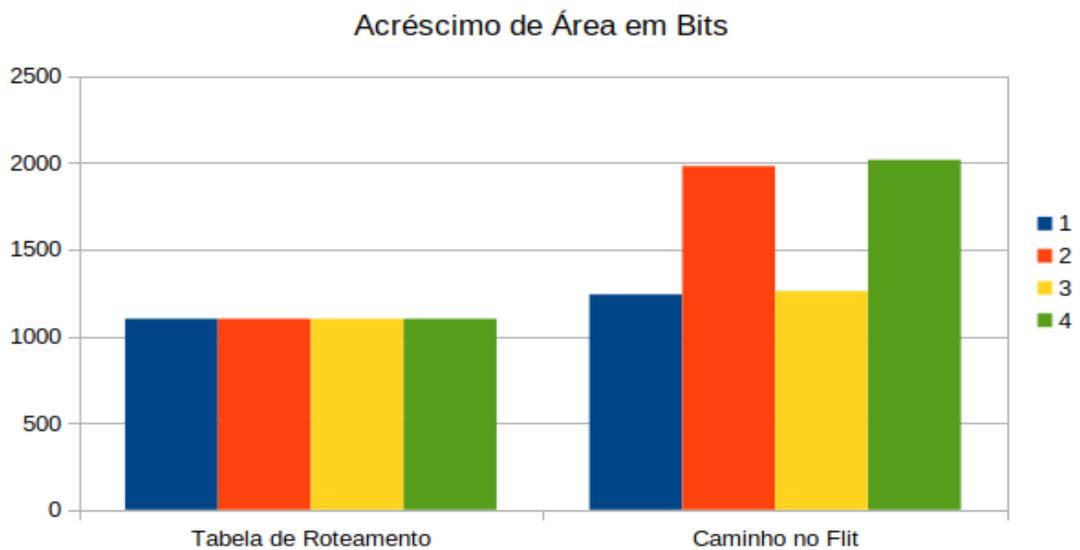
Fonte: Fonte: Dados da Pesquisa

Tabela 3: Test T - Latência Média

T Test	0,119
--------	-------

Fonte: Dados da Pesquisa

Figura 7: Acréscimo de Área em Bits



Fonte: Dados da Pesquisa

Tabela 4: Test T - Acréscimo de Área em Bits

T Test	0,119
--------	-------

Fonte: Dados da Pesquisa

Foi utilizado nesse trabalho o T Test no modo bicaudal com tipo pareado e usando a taxa de relevância padrão de 5%; quanto mais longe de 0,05 para baixo, maior a relevância de diferença entre os dois conjuntos de teste.

Na Figura 5 é mostrada a porcentagem da quantidade de pacotes que chegaram dentro do prazo. Na Figura 6 são mostradas as latências médias de cada aplicação, para os dois tipos de roteamento. Já a Figura 7 mostra o quanto em bits é aumentado em cada método de roteamento. Essa quantidade de bits é calculada para o roteamento em que o caminho é armazenado no cabeçalho do flit, calculando a quantidade de bits que é usada para representar o número do roteador que é colocado na fila do caminho. Como são utilizados exemplos com valores que variam de 0 a 9, são necessários 4 bits para representar o valor de cada roteador acrescido ao caminho. No roteamento por tabela usa-se o mesmo método para o cálculo da área. São pelo menos 10 linhas nas tabelas usadas, de acordo com a topologia apresentada e grafos das aplicações testadas. Cada linha contém 3 campos. Os campos de destino e canais precisam de 4 bits para ser representados, já o campo para a porta de saída precisa de 3 bits. O fato de todas as aplicações terem 10 tarefas, implica em um mesmo tamanho para todas as tabelas.

As Tabelas 2, 3 e 4 são responsáveis por mostrar o resultado dos testes estatísticos para pacotes entregues, latência e área, respectivamente. Como pode-se observar pelas Tabelas, a variação do número de pacotes entregues no prazo é mínima. Observando a Figura 5, percebe-se que a quantidade de pacotes entregues teve quase a mesma porcentagem usando os dois métodos de roteamento. Já a Figura 6, observa-se que referente a latência média, a diferença foi expressiva, mostrando assim que para a otimização de latência, o roteamento desenvolvido nesse trabalho consegue obter resultados mais expressivos. Isso ocorre devido ao acréscimo de tempo necessário para acessar as tabelas de roteamento em cada roteador e para procurar a linha que contém o destino do pacote. Em relação a área, observa-se na Figura 7 uma

diferença significativa mostrada pelo gráfico e pelo Test T, onde pode-se verificar que o roteamento com tabelas consome uma área menor. Isso ocorre devido ao impacto de se colocar o caminho total no flit de cabeçalho.

6 CONCLUSÃO E TRABALHOS FUTUROS

Nesse trabalho se buscou desenvolver um novo método para roteamento em topologias irregulares. O método proposto sugere o armazenamento das coordenadas de todo o caminho dos pacotes em seu flit de cabeçalho. Para efeito de experimentos, o método apresentado foi comparado com um tipo de roteamento utilizando tabelas. Os resultados mostraram que em relação a latência média foi possível obter-se um resultado com menores valores quando comparado ao de tabelas. Quanto aos prazos de entrega dos pacotes, os valores se mantiveram muito semelhantes. Por fim, em termos de área os resultados são menos eficientes. Conclui-se então que a abordagem proposta nesse trabalho pode ser considerada eficaz nos casos em que se pode priorizar a otimização de latência em detrimento de um maior custo em área.

Como trabalhos futuros, pretende-se usar uma heurística para realizar o mapeamento em tempo de projeto e testá-la frente à diferentes taxas de injeção, visando diminuir ainda mais a latência da rede. Também busca-se diminuir a quantidade de roteadores utilizados na topologia, causando impacto positivo na área utilizada, sem comprometer a eficiência da rede. Por fim pretende-se fazer uso de WiNoCs (*Wireless Network-on-Chip*)(Chen e Dai, 2015) (Opoku et al., 2015) , para também se obter um ganho em consumo de energia.

REFERÊNCIAS

Luca Benini and Giovanni De Micheli. **Networks on chips: A new soc paradigm**. Computer-IEEE Computer Society-, 35(EPFL-ARTICLE-165542):70–78, 2002.

Axel Jantsch, Hannu Tenhunen, et al. **Networks on chip**, volume 396. Springer, 2003.

S Tosun, Y Ar, and S Ozdemir. **Application-specific topology generation algorithms for network-on-chip design**. IET computers & digital techniques, 6(5):318–333, 2012.

Vahid Babaei Ajabshir and Suleyman Tosun. **Fault-tolerant routing for irregular-topology-based network-on-chips**. In Computing and Networking (CANDAR), 2014 Second International Symposium on, pages 123–129. IEEE, 2014.

Suleyman Tosun, Vahid Babaei Ajabshir, Ozge Mercanoglu, and Ozcan Ozturk. **Fault-tolerant irregular topology design method for network-on-chips**. In Digital System Design (DSD), 2014 17th Euromicro Conference on, pages 631–634. IEEE, 2014.

Reza Moraveji, Hamid Sarbazi-Azad, and Albert Y Zomaya. **A general methodology for routing in irregular networks**. In Parallel, Distributed and Network-based Processing, 2009 17th Euromicro International Conference on, pages 155–160. IEEE, 2009.

Xinming Duan, Dakun Zhang, and Xuemei Sun. **Routing schemes of an irregular mesh-based noc**. In Networks Security, Wireless Communications and Trusted Computing, 2009. NSWCTC'09. International Conference on, volume 2, pages 572–575. IEEE, 2009.

Qingli Xiao, Hongzhou Fu, Lu Zheng, and Dexue Zhang. **An efficient routing scheme for irregular mesh nocs**. In Electronics Information and Emergency Communication (ICEIEC), 2013 IEEE 4th International Conference on, pages 121–124. IEEE, 2013.

Robert P Dick, David L Rhodes, and Wayne Wolf. **Tgff: task graphs for free**. In Proceedings of the 6th international workshop on Hardware/software codesign, pages 97–101. IEEE Computer Society, 1998.

F Ghenassia. **Transaction-level modeling with systemc: Tlm concepts and applications for embedded systems**. Secaucus, 2006.

Cesar Albenes Zeferino, Jaison Valmor Bruch, Thiago Felski Pereira, Marcio Eduardo Kreutz, and Altamiro Amadeu Susin. **Avaliação de desempenho de rede-em-chip modelada em systemc**. In Proceedings of the 27rd Congress of Brazilian Computer Society-WPerformance, pages 559–578, 2007.

Ankur Agarwal, Cyril Iskander, and Ravi Shankar. **Survey of network on chip (noc) architectures & contributions**. Journal of engineering, Computing and Architecture, 3(1):21–27, 2009.

William James Dally and Brian Patrick Towles. **Principles and practices of interconnection networks**. Elsevier, 2004.

Cesar Albenes Zeferino. **Redes-em-chip: arquiteturas e modelos para avaliação de área e desempenho**. 2003.

Jenhui Chen and Peng Dai. **Multicast transmission with energy-proportional power-gating scheme for wireless interconnects noc**. In Globecom Workshops (GC Wkshps), 2015 IEEE, pages 1–6. IEEE, 2015.

Michael Opoku Agyeman, Ji-Xiang Wan, Quoc-Tuan Vien, Wen Zong, Alex Yakovlev, Kenneth Tong, and Terrence Mak. **On the design of reliable hybrid wired-wireless network-on-chip architectures**. In Embedded Multicore/Many-core Systems-on-Chip (MCSoc), 2015 IEEE 9th International Symposium on, pages 251–258. IEEE, 2015.