



**HAL**  
open science

# Un Algorithme de Routage Reconfigurable pour la Tolérance aux Fautes dans le micro-réseau DSPIN

Zhen Zhang, Alain Greiner

► **To cite this version:**

Zhen Zhang, Alain Greiner. Un Algorithme de Routage Reconfigurable pour la Tolérance aux Fautes dans le micro-réseau DSPIN. Colloque national GDR SOC-SIP, 2008, Paris, France. hal-00592277

**HAL Id: hal-00592277**

**<https://hal.science/hal-00592277>**

Submitted on 11 May 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Un Algorithme de Routage Reconfigurable pour la Tolérance aux Fautes dans le micro-réseau DSPIN

Zhen ZHANG, Alain GREINER  
Université Pierre et Marie Curie & LIP6  
4, Place Jussieu, 75252 Paris Cedex 05, France  
{Zhen.ZHANG, Alain.GREINER}@lip6.fr

## I. INTRODUCTION

Selon la prédiction d'un commentateur d'INTEL [1] : "Nous pourrions fabriquer une puce avec 100 milliards de transistors dans la prochaine décennie". Ceci concerne par exemple, les architectures de type MP2-SoC (Massively Parallel Multi Processor System on Chip) pouvant intégrer plusieurs milliers de coeurs de processeurs interconnectés par un micro-réseau intégré sur puce (Network-on-Chip). Le commentateur ajoute aussi : "Néanmoins 20 milliards de transistors seront défectueux après la fabrication et 10 milliards de transistors tomberont en panne dans la première année de fonctionnement", Ceci impose de prendre en considération la tolérance aux pannes dès la définition de l'architecture.

Comme l'architecture MP2-SoC contient un grand nombre de composants identiques, tels que des processeurs et des bancs mémoire, une méthode simple pour introduire la tolérance aux fautes consiste alors à prévoir des mécanismes de désactivation des composants défectueux et à redéployer l'application logicielle sur les processeurs et bancs mémoire restants. Malheureusement, cette méthode ne convient pas si la panne affecte une ressource partagée par tous les composants. La ressource partagée la plus critique est évidemment le micro-réseau lui-même. Un seul routeur défectueux compromet le fonctionnement global car les algorithmes de routage du NoC sont généralement conçus pour des topologies régulières, et un seul "trou" dans le réseau est donc fatal.

La reconfiguration du réseau nécessite de résoudre les 3 problèmes suivants :

- 
- A.** Le routeur défectueux doit être détecté et identifié par un mécanisme de test approprié.
  - B.** Un algorithme de routage déterministe, distribué et reconfigurable doit être défini et implémenté dans chaque routeur.
  - C.** Un bus de configuration robuste doit être implémenté dans le NoC pour distribuer les informations de reconfiguration aux routeurs.
- 

Dans cet article, nous nous consacrons au problème B, dans le cas d'une topologie de type 2D-Mesh.

Les algorithmes de routage de tolérance aux fautes pour les réseaux classiques publiés au cours des deux dernières

décennies peuvent être séparés en deux catégories selon leurs façons de traiter le problème de l'interblocage. Ils utilisent soit une technique de canal virtuel [2], soit le "Turn model" [3]. L'algorithme de reconfiguration d'un NoC doit respecter les contraintes suivantes :

---

<i>Surcoût faible</i>	Le surcoût matériel associé à l'implémentation de l'algorithme de routage reconfigurable doit être faible.
<i>Générique</i>	Il faut que l'algorithme de routage reconfigurable gère toutes les topologies dégradées quelque soit le routeur défectueux.
<i>Scalable</i>	L'algorithme de routage reconfigurable doit être indépendant de la taille du NoC.
<i>Deadlock-free</i>	Il faut que l'algorithme de routage reconfigurable garantisse l'absence d'interblocage.
<i>Déterministe</i>	Il faut que l'algorithme de routage reconfigurable garantisse l'acheminement dans l'ordre.

---

Nous présentons dans la suite de cet article un algorithme de routage reconfigurable basé sur l'approche "X-First", qui a été implanté et évalué sur l'architecture DSPIN [4], [5].

## II. "X-FIRST" & 2D-MESH

Un routeur du NoC de type 2D-Mesh est un noeud dans la topologie de type 2D-Mesh. La coordonnée d'un noeud est un couple  $(Y,X)$ . Avec l'algorithme de routage "X-First" [6], les paquets sont routés premièrement vers la direction X et puis vers la direction Y. Donc le chemin entre un noeud  $(y,x)$  et un noeud  $(y',x')$  est un **chemin unique** :

$$L = \{R_0(y, x), \dots, R_{|x'-x|}(y, x'), \dots, R_{|y'-y|+|x'-x|}(y', x')\}$$

Si un routeur est défectueux, il constitue un trou dans la grille 2D, et ce trou coupe les **chemins uniques** sur lequel il se trouve. Donc il faut que l'algorithme de routage reconfigurable rétablisse les **chemins uniques coupés**.

## III. L'ALGORITHME DE ROUTAGE RECONFIGURABLE

*L'idée principale de l'algorithme de routage déterministe, distribué et reconfigurable est de router les paquets par un contour sans cycle autour du trou pour rétablir les chemins uniques coupés.*

a) *Définition - Les voisins*: Dans une topologie de type 2D-Mesh, un trou (Y,X) a 4 noeuds voisins directs (N,S,W,E) et 4 noeuds voisins indirects (NE,NW,SE,SW), comme le montre la Figure 1.1.

b) *Définition - Le contour naturel*: Les 8 voisins d'un trou (Y,X) définissent un contour naturel, qui divise la topologie en deux parties : la partie normale A et la partie défective B.

c) *Définition - Un contour sans cycle*: Il existe deux cycles dans le graphe de dépendance des canaux (CDG) d'un contour naturel défini par 8 voisins, comme le montre la Figure 1.2. Dally [6] a démontré qu'il y a des risques d'interblocage s'il y a des cycles dans le CDG. Conformément aux principes du "Turn Model" nous interdisons les deux tournants dans le noeud NE afin de couper les 2 cycles dans le CDG, comme le montre la Figure 1.3.

d) *Définition - Les nouveaux chemins uniques*: Normalement, un trou coupe 8 chemins uniques  $L_i$   $0 \leq i \leq 7$ . Il nous oblige de rétablir 8 nouveaux chemins uniques  $NewL_i$   $0 \leq i \leq 7$  par un contour sans cycle, comme le montre la Figure 1.4.

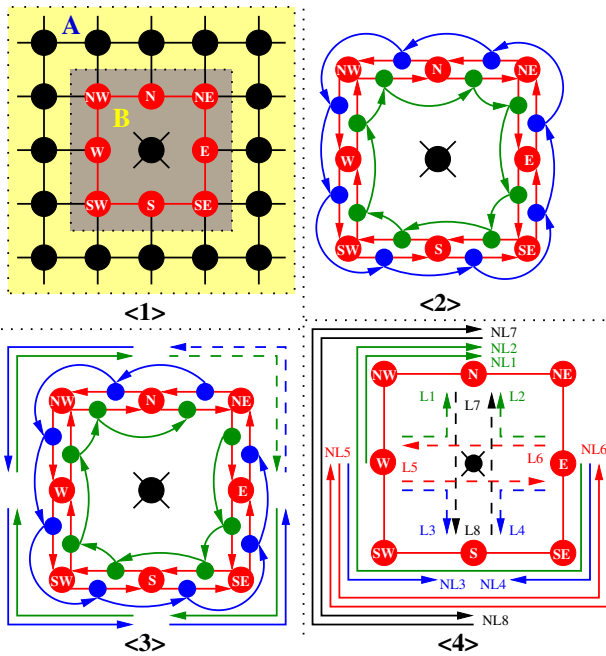


FIG. 1. L'image 1 montre les voisins d'un trou et un contour naturel. L'image 2 montre le CDG d'un contour naturel. L'image 3 montre un contour sans cycle. L'image 4 montre les 8 nouveaux chemins uniques  $NewL_i$  remplaçant les 8 chemins uniques coupés  $L_i$ .

En fait, dans une topologie de type 2D-Mesh, il y a 9 types de contour sans cycle associées à 9 positions possibles pour le trou : le trou peut être dans un des 4 coins, sur une des 4 faces, ou à l'intérieur de la grille 2D. Nous avons démontré que pour ces neuf configurations, peut rétablir les chemins uniques coupés.

e) *La vérification d'absence d'interblocage*: Notre algorithme de routage reconfigurable est une combinaison de "X-First" et de contour sans cycle : dans la partie normale

A, les routeurs routent les paquets utilisant l'algorithme "X-First"; dans la partie défective B, les routeurs routent les paquets utilisant un contour sans cycle. Nous avons démontré formellement que cette combinaison garantit l'absence d'interblocage pour toutes les topologies possédant un seul routeur défectueux dans une grille  $10 * 10$ , en utilisant l'outil ODI [7] développé par Sami Taktak au LIP6.

#### IV. DSPIN RECONFIGURABLE

Le micro-réseau DSPIN a été conçu au LIP6 et implémenté matériellement par ST Microelectronics. Sa caractéristique est d'être extrêmement compact et peu coûteux en surface de silicium. Nous avons introduit le mécanisme de reconfiguration par logiciel de l'algorithme de routage X-First décrit ci-dessus. La configuration est réalisée en écrivant dans un registre de configuration de 4 bits implanté dans chaque routeur. Nous appelons DSPIN/reconfigurable l'architecture ainsi modifiée, et nous avons cherché à évaluer le coût en matériel, et le coût en performance de cette reconfigurabilité.

Le surcoût matériel d'un routeur de DSPIN/reconfigurable (évalué après synthèse placement et routage) par rapport au routeur DSPIN non reconfigurable est environ 8%. Par ailleurs, nous avons simulé, en utilisant l'environnement SoCLib [8] un réseau DSPIN reconfigurable pour une grille  $5 * 5$ , et pour toutes les topologies ayant un routeur défectueux, et nous avons mesuré la latence moyenne des paquets. Nous avons pu observer que pour une charge du réseau réaliste (c'est à dire  $< 15\%$ ), l'impact sur la latence est négligeable même lorsqu'un trou est situé au centre de la topologie. En revanche, le seuil de saturation peut être sensiblement diminué.

#### V. CONCLUSION

Nous avons défini et validé un algorithme de routage reconfigurable pour la tolérance aux fautes dans des NoCs possédant une topologie de type 2D-Mesh. Cet algorithme peut être utilisé pour toute topologie ayant un routeur ou une région défectueuse. Cet algorithme a un surcoût matériel très faible, et un impact sur les performances presque négligeable. DSPIN/reconfigurable est donc un micro-réseau bien adapté aux architectures MP2-SoCs.

#### RÉFÉRENCES

- [1] S. Furber, "Living with Failure : Lessons from Nature?" *ETS'06-Volume 00*, pp. 4–8, 2006.
- [2] J. Duato, "A Theory of Fault-Tolerant Routing in Wormhole Networks," *IEEE TPDS*, vol. 8, no. 8, pp. 790–802, 1997.
- [3] C. Glass and L. Ni, "Fault-tolerant wormhole routing in meshes," *Fault-Tolerant Computing, 1993*, pp. 240–249, 1993.
- [4] I. Panades, A. Greiner, A. Sheibanyrad, and G. STMicroelectronics, "A Low Cost Network-on-Chip with Guaranteed Service Well Suited to the GALS Approach," *NanoNet'06*, pp. 1–5, 2006.
- [5] DSPIN, <http://www.lip6.fr/Direction/2005-05-13-DSPIN.pdf>.
- [6] W. Dally and C. Seitz, "Deadlock-free message routing in multiprocessor interconnection networks," *IEEE Transactions on Computers*, vol. 36, no. 5, pp. 547–553, 1987.
- [7] S. Taktak, E. Encrenaz, and J. Desbarbieux, "A Tool for Automatic Detection of Deadlock in Wormhole Networks on Chip," *HLDVT, 2006.*, pp. 203–210.
- [8] "Projet SOCLIB :Plateforme de modélisation et de simulation de systèmes intégrés sur puce," <http://www.soclib.fr>.