مجلة جامعة طرطوس للبحوث والدراسات العلمية _ سلسلة العلوم الهندسية المجلد (٧) العدد(٦) ٢٠٢٣
Tartous University Journal for Research and Scientific Studies - engineering Sciences Series Vol. (7) No. (6) 2023

دراسة وتقييم الأداء ضمن شبكات الربط التفرعية وفق بنية الفراشة

د.م. حسن محمد البستاني*
م. آلاء رجب محمد**

(تاريخ الإيداع ٢٠٢٣/٧/١١ . قُبِل للنشر في ٢٠٢٣/٨/٢٤)

□ ملخّص□

تُعدّ شبكة الفراشة من البنى المهمة في تنفيذ الدارات عالية التكامل (Very large scale integration VLSI) لتطبيقات شبكات الربط في الحوسبة التفرعية، حيث تمتلك العقدة عدد محدد من منافذ الإدخال/الإخراج أو ما يسمى bounded degree networks، وتضمن هذه البنية مسار محقق بين كل معالج وكل موقع ذاكرة من خلال عدة مراحل من المبدلات مرتبطة بحجم الشبكة، أما ما يعيب الشبكات المبنية على مخطط الفراشة عدم إعطائها المستوى المطلوب من حيث تسامح العطل، باعتبار أن خطأ وحيد في عقد التبديل أو في التوصيلات يؤدي الى حدوث فشل في الشبكة. يهدف هذا البحث الى دراسة بنية شبكة الفراشة وتنفيذها برمجيا باستخدام لغة بايثون، ودراسة طرق تحسين التسامح بالخطأ على بنية الفراشة وتأثيره على بارامترات الشبكة الأخرى مثل قياس الشبكة وعرض النطاق وعدد الوصلات والكلفة، بحيث يمكن إجراء القياس البرمجي لبارامترات الشبكة وفق أي حجم.

الكلمات المفتاحية: الشبكات التفرعية، مخطط الفراشة، التسامح بالعطل، الحواسيب فائقة الأداء.

*: مدرس في كلية هندسة تكنولوجيا المعلومات والاتصالات – جامعة طرطوس
**: طالبة ماجستير في قسم هندسة النظم الحاسوبية والالكترونية، كلية هندسة تكنولوجيا المعلومات والاتصالات، جامعة طرطوس - سورية

مجلة جامعة طرطوس للبحوث والدراسات العلمية _ سلسلة العلوم الهندسية المجلد (٧) العدد(٦) ٢٠٢٣
Tartous University Journal for Research and Scientific Studies - engineering Sciences Series Vol. (7) No. (6) 2023

Studying and evaluation the performance within the parallel networks according to the structure of the butterfly

Dr-Eng. Hasan Mohammad Albustani*
Eng. Ala'a Rajab Mohammad**

(Received 11/7/2023 . Accepted 24/8/2023)

□ ABSTRACT

The butterfly network is consider an important architecture in implementing the very large scale integration (VLSI) for the applications of the interconnection networks in parallel computing, where the node has a specific number of I/O ports or what is called bounded degree networks, and This architecture ensures a verified path between each processor and each memory location through many stages of switches which are linked to the network size, but the wrong thing in the network that depend on the required level in terms of fault tolerance, because one fault in the switch nodes or connections leads to network failure. This research aims to study the structure of the butterfly network and implementing it programmatically using Python language, and studying the ways to improve fault tolerance on the butterfly structure and its effect on other network parameters such as diameter, bandwidth, number of links and cost.

Keywords: parallel networks, butterfly diagram, fault tolerance ،Hight performance computer.

* Assistant Professor, Faculty of Communication and Information Engineering Technology – Tartous University, Tartous, Syria.
** Master Student at CESE Department, Faculty of Communication and Information Engineering Technology, Tartous University, Syria

1-مقدمة

تزداد الحاجة الى تنفيذ التطبيقات البرمجية بأداء زمني أفضل وخاصة ضمن تطبيقات الزمن الحقيقي والتطبيقات عالية التعقيد التي يتطلب تنفيذها متوالية من المراحل كتطبيقات الذكاء الصناعي وتطبيقات التشفير/فك التشفير، ومن هنا كانت الحاجة الى تطوير المعالجات فائقة الأداء[1] ، وتنفيذ التطبيقات ضمن بنى أكثر تعقيداً من البرمجة التسلسلية الاعتيادية، فظهرت الشبكات التفرعية والتي تهدف الى ربط N من المعالجات مع M موقع ذاكرة ضمن روابط وتشكيلات وسيطية متنوعة مثل Bus, Ring, Hypercube, Butterfly [2]. قامت شركة IBM بتطوير نظام حاسوبي يحتوي على 6562 شريحة dual-core AMD Opteron و 12240 Cell chips، وتستخدم شركة google عناقيد بحجم 8000 عقدة إلا أنها تعاني من فشل بمقدار 2/3 من العقد سنوياً أو ما يكافئ فشل عقدة كل 36 ساعة [3]. كذلك فإن أداء الحواسيب الفائقة يرتبط بشكل مباشر مع عدد المعالجات المستخدمة فيه حيث يحتوي IBM Blue Gene/L على 131072 معالج، ومع ازدياد عدد المعالجات المرتبطة فإن احتمالية فشل التجهيزات تزداد بشكل مضطرد، ويعتبر الزمن الوسطي بين عطلين MTBF من المعاملات المهمة في قياس أداء الحواسيب الفائقة، إذ أن هذا الزمن يمكن أن يكون أصغر من زمن تنفيذ بعض التطبيقات عالية التعقيد [4]. بالمقابل فإن تطور تقنية VLSI جعل بالإمكان وضع أنوية متعددة على ذات الشريحة والحصول على نظام مضمن متعدد النوى على شريحة 'SOC'، وظهرت الحاجة الى تطوير بنية الربط التفرعية بين تلك النوى بما يضمن الكفاءة والأداء[5] . تتضمن مخططات التوصيل للمعالجات على شريحة العديد من الأنماط حيث قدمت الدراسة [6] تقييم ومقارنة للشرائح متعددة المعالجات في بنى التوصيل: 2-D mesh, Rink, crossbar، أما البحث [7] فقد اعتمد في تقييمه لربط وفق بنية الشبكة المكعبة Cube Networks على خمسة بارامترات وهي السرعة وفقد الرزم و التغطية Coverage ودرجة الشبكة وأبعاد الشبكة، أما البحث [8] فقد شمل تحليل أداء شبكة الفراشة وتطويرها بما يخص التسامح مع العطل من خلال دمجها مع الشبكة المكعبة.

يتم الاعتماد على الفائضية Redundancy في تطوير الأنظمة المختلفة فيما يخص تسامح مع العطل، وذلك بتوفير مصادر بديلة أو متوازية للعناصر مما يمكِّن من تلافي فشل المنظومة عند حدوث العطل، وقد اعتمدت دراسات عديدة على تطوير الفائضية بما يضمن وثوقيه أعلى للمنظومة، ومنها الدراسة [9] والتي قامت بتطبيق نموذج الفائضية غير المتناظر اعتماداً على خلايا ذاكرة NAND Flash بهدف رفع الموثوقية للذاكرة وتقليل فقد البيانات وتوصلت الى تقليل الأخطاء بمعدل 98.33%، أما البحث [10] فقد قدم تطوير نموذج فائضية معتمد على تحويل فورييه السريع FFT لبناء نظام متسامح العطل في شرائح المصفوفات القابلة لإعادة البرمجة حقلياً FPGA مع ضمان توفير موارد الشريحة. إن مصطلح الموثوقية في نماذج الشبكات يرتبط باستمرار عمل نموذج الربط (بين المعالجات التفرعية) وتحقيق متطلبات معدل النقل والإنتاجية والفقد لمدة زمنية كتابع للزمن [11].

تم في هذا البحث إجراء الاختبارات على شبكة الفراشة من حيث تغير الكلفة وأبعاد الشبكة مقابل زيادة درجة تسامح العطل، واختبار آلية التوجيه واختيار المسار وفق خوارزمية تحقق تسامح عطل وحيد في المرحلة الأولى من مخطط الفراشة، مما يساعد في اختيار نوع الشبكة الأفضل لتطبيق معين، عند أحجام مختلفة لمخطط التوصيل، من خلال مقارنة الجداول الحسابية لشبكة الفراشة مع أي نوع آخر من مخططات التوصيل، ويمكن اعتماد النتائج الحسابية للبحث اذ تم التركيز على دراسة أثر زيادة درجة تسامح الخطأ في شبكة الفراشة على التأخير (أبعاد الشبكة) والكلفة

(العدد الإجمالي للعقد والوصلات) في نوع من أنواع مخططات التوصيل التي تعتمد على ثبات درجة المبدل(مبدل بعدد مداخل ومخارج ثابت).

2-هدف البحث

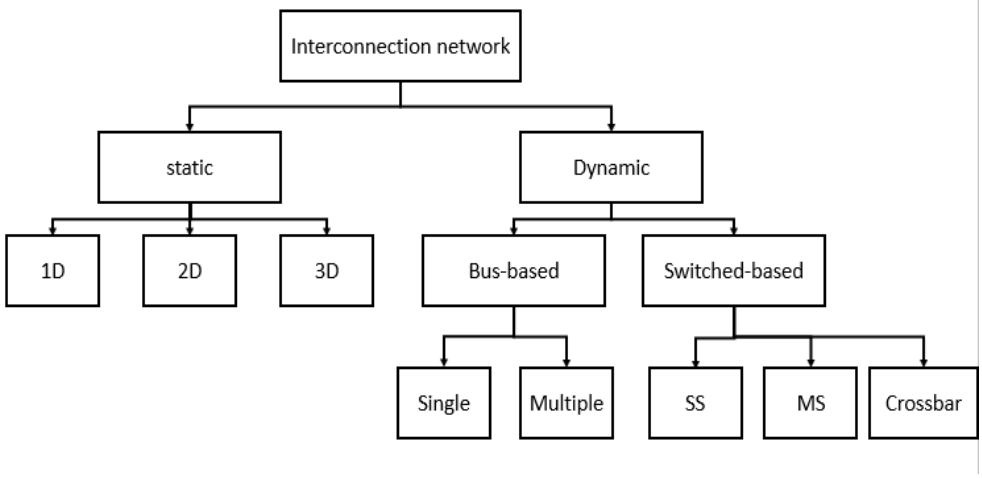
قمنا في هذا البحث بدراسة إحدى بنى الشبكات التفرعية وهي شبكة الفراشة، وتحليل أدائها وقياس بارامتراتها رياضياً وكذلك التنفيذ البرمجي لها، ودراسة حالات وطرق تطوير بنية الفراشة لتعزيز تكيفها مع الأعطال وكذلك إعادة البرمجة والقياسات لبارامترات الشبكة: التأخير، أبعاد الشبكة والكلفة وعرض النطاق في بنى شبكة الفراشة المحسنة حيث تمت الدراسة والمقارنة على أحجام مختلفة لشبكة الفراشة.

3-طرق البحث وموارده

3-1- لمحة عن أنواع شبكات الربط البيني

تصنف شبكات الربط البيني كما هو موضح الشكل (1):

- الديناميكية: وفيه يتم إنشاء الروابط أثناء الاتصال حسب الحاجة.
- الستاتيكية: لها مخطط توصيل ثابت يصنف وفقاً لنمط التوصيل.



الشكل (1): يمثل أنواع شبكة الربط البيني

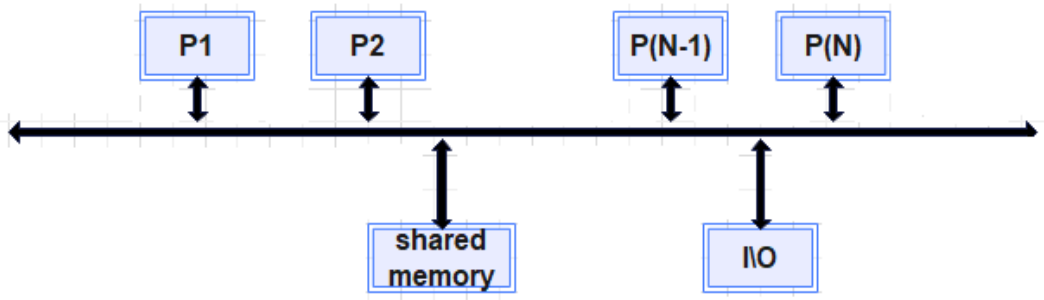
3-2-الشبكات الديناميكية:

بالنسبة للشبكات الديناميكية فيمكن تقسيمها الى:

1-الشبكات المعتمدة على الباص Bus-based Dynamic Interconnection Networks

تُعدّ الشبكات أحادية الناقل single bus أبسط طريقة توصيل، مكون من N معالج يتراوح عددها بين (2-50) معالج، لكل منها ذاكرته الخاصة من أجل تقليل حركة المرور بين الذاكرة والمعالج، وتكون متصلة بمسار مشترك حيث أنَ كل المعالجات تتصل مع ذاكرة مشتركة واحدة، وتقاس درجة التعقيد من حيث عدد المسارات المستخدمة، أما تعقيد الزمن فيقاس بكمية المدخلات لتأخير الخرج، ويكون التحكم مطلوب لتحديد أولوية النقل عندما يتنافس أكثر من معالج

للوصول للمسار، ويُقرّر باستخدام نظام أولوية معين (Rotating priority , equal priority , LRU priority، إلخ)، ويتم منح المعالج حق الوصول إلى المسار خلال فترة زمنية معينة، يوضح الشكل (2) بنية الشبكات المعتدة على الباص Bus– based.



الشكل (2): بنية شبكة Bus based

أما الأنظمة متعددة الباص Multiple Bus Systems فيتم فيها استخدم عدة مسارات متوازية لتوصيل معالجات متعددة مع وحدات ذاكرة متعددة، حيث تقدِّم عدد من الميزات المرغوبة مثل الموثوقية العالية وسهولة توسيع حجم الشبكة، وهناك عدد من مخططات الاتصال الممكنة في هذه الحالة منها:

- The multiple bus with full bus–memory connection (MBFBMC)
- multiple bus with single bus memory connection (MBSBMC)
- multiple bus with partial bus–memory connection (MBPBMC)
- multiple bus with class–based memory connection (MBCBM)

يوضح الجدول (1) مقارنة بين معماريات الأنظمة متعددة الناقل من حيث عدد الوصلات والتحميل على الناقل[12].

الجدول (1): مقارنة بين معماريات Multiple bus system

Connection type	No. of Connection	Load on Bus i
MBFBMC	$B(N+M)$	$N+M$
MBSBMC	$BN+M$	$N+M_j$
MBPBMC	$B(N+M/g)$	$N+M/g$
MBCBMC	$BN+\sum_{j=1}^{K} M_j(j+B-K)$	$N+\sum_{j=\max(i+k-B,1)}^{k} M_j, 1 \le i \le B$

حيث أن:

k: تمثل عدد الفئات

g: عدد النواقل ضمن المجموعة

M_j: عدد نماذج الذاكرة في الفئة j

N: عدد المعالجات

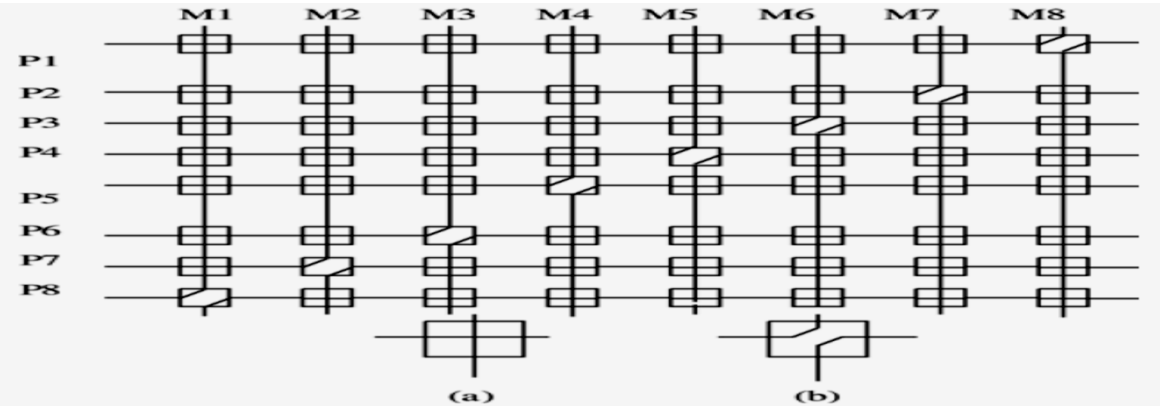
M: نماذج الذاكرة

B: عدد النواقل (الباص المتعدد)

2-شبكات الاتصال المعتمدة على المبدل switch based interconnection Networks:

وتقسم شبكات الاتصال المعتمدة على المبدل إلى:

- Crossbar network : يمكن أن توفر اتصالات متزامنة بين جميع مدخلاتها وجميع مخرجاتها، تحتوي على مبدل. يُقاس تعقيد الشبكة من حيث عدد المبدلات، وتعقيد التأخير المقاس من حيث الدخل لتأخير الخرج، يوضح الشكل (3) بنية شبكة crossbar [13].



الشكل (3): بنية شبكة crossbar network

- الشبكات وحيدة المرحلة Single stage network

تحتوي على العديد من النماذج مثل شبكة الفراشة Butterfly network والشبكة المكعبة cube network

- الشبكات متعددة المراحل Multistage Networks، ومن الأمثلة على هذه البنى:
- Shuffle_Exchange
- banyan network
- Omega network

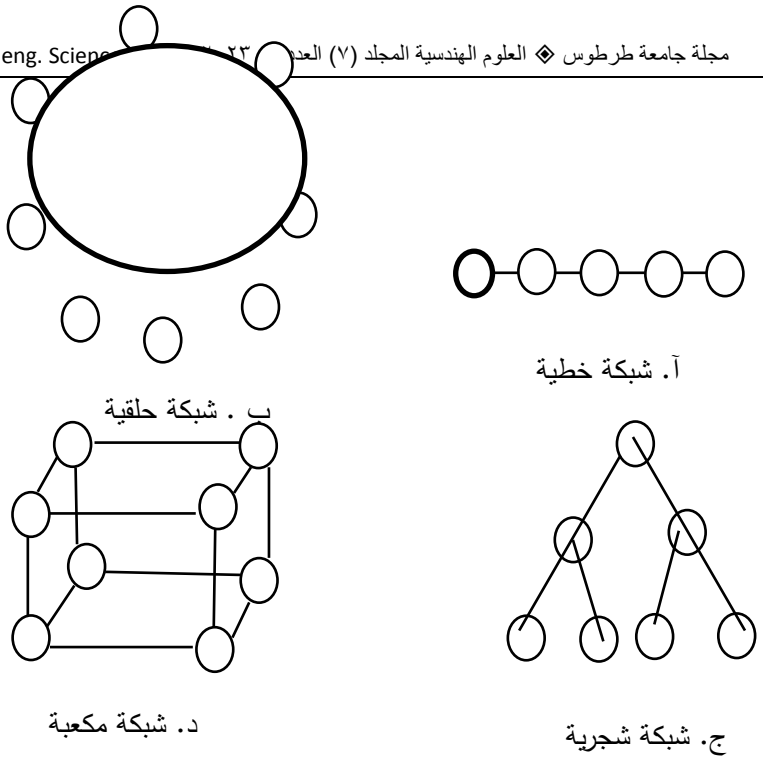
3-3-الشبكات الستاتيكية:

الصنف الثاني من شبكات الربط البيني فهو الشبكات Static interconnection network، وتعتبر الشبكات محدودة التوصيل Limited connected network، وتتصف أنها لا تقدم رابط مباشر من كل عقدة إلى باقي العقد في الشبكة وبالتالي يمكن تلخيص قيود هذه الشبكة على النحو الآتي:

1-الحاجة إلى نمط التوصيل بين العقد.

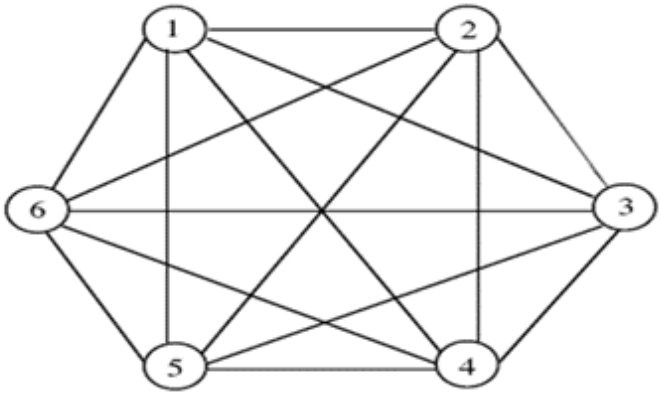
2-الحاجة إلى آلية لتوجيه الرسائل في جميع أنحاء الشبكة حتى تصل لأهدافها.

لقد تم تطوير عدد من أنماط التوصيل البيني المنتظمة لشبكات LCN، ويوضح الشكل (4) أمثله على هذا النوع من هذه الشبكات [14].



الشكل (4): أنماط شبكة SLC

النوع الثاني من شبكات التوصيل الستاتيكية هي الشبكات كاملة التوصيل Completely connected network، حيث يتم توصيل كل عقدة بجميع العقد الأخرى بالشبكة، وتتميز بضمان سرعة تسليم الرسائل من أي عقدة مصدر الى أي عقدة هدف ولكنها ذات كلفة مرتفعة بسبب عدد وصلاتها الكبير، يوضح الشكل (5) بنية هذه الشبكة.



الشكل (5): بنية الشبكات مكتملة التوصيل

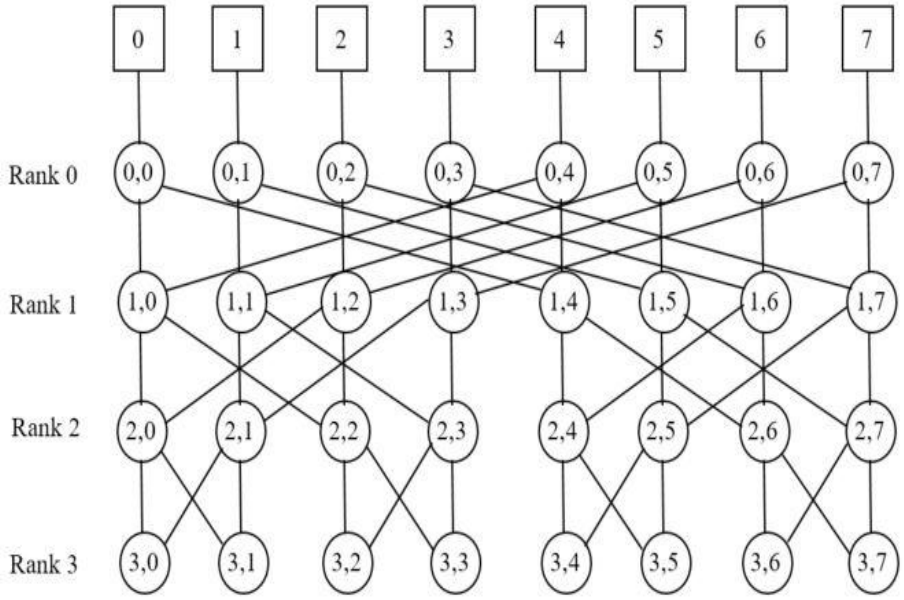
3-4- بنية شبكة الفراشة

يجب أن يكون لشبكة الاتصال البيني لنظام الذاكرة المشتركة متعدد المعالجات زمن انتقال منخفض وعرض نطاق ترددي مرتفع على عكس أنظمة الشبكات الأخرى، مثل الشبكات المحلية (LAN) أو الإنترنت، حيث أن المكونات الرئيسية لشبكة الربط هي:

- عقد المعالج والتي تتكون من معالج واحد أو أكثر بالإضافة إلى ذاكرات التخزين المؤقت والذاكرة الأساسية ومساعد الاتصال.
- عقد التبديل: تتصل عُقد تبديل المستوى الأعلى بعقد تبديل المستوى الأدنى في طوبولوجيا الشبكة متعددة المراحل، كما هو موضح في الشكل (6) [15]، حيث تتصل عقد التبديل في المرتبة 0 بعقد المعالج مباشرة، أما عقد التبديل في المرتبة 1 تتصل بعقد التبديل في الرتبة 0.

- الروابط (نقاط الربط): وهي أسلاك مادية بين عقدتي تبديل. يمكن أن تكون نقاط أحادية الاتجاه أو نقاط ثنائية الاتجاه.

بالنسبة لشبكة فراشة مكونة من P معالج، يجب أن تحتوي الشبكة $P(\log_2 P + 1)$ عقدة تبديل [16]. يوضح الشكل (6) شبكة مكونة من 8 عقد معالجة، مما يتطلب 32 عقدة تبديل. يتم تمثيل كل عقدة بالثنائية (المرتبة، رقم العمود). على سبيل المثال، يتم تمثيل العقدة في العمود 6 في المرتبة 1 بالثنائية (1,6) والعقدة في العمود 2 في المرتبة 0 يتم تمثيلها بالثنائية (0,2).



الشكل (6): بنية شبكة الفراشة

3-5- الأعطال وتسامح مع العطل

تصنف الأعطال التي تحصل ضمن أي نظام الى أعطال برمجية وأعطال عتادية، أما الأعطال العتادية فيمكن تصنيفها من خلال عدة معاملات أهمها: المدة الزمنية حيث يمكن أن تكون:

- الأعطال دائمة "Permanent"، والتي تستلزم تبديل العنصر.
- الأعطال العابرة Transientحيث يمكن استعادة الخواص الوظيفية للعنصر كاملة بعد مدة زمنية.
- الأعطال المتقطعة حيث تتبدل حالة العنصر بين فعال بدون أخطاء الى حالة عطل بشكل متكرر.

كذلك تصنف الأعطال وفق المرحلة الزمنية من حياة العنصر بدأ من مرحلة التصنيع أو التنفيذ أو التطوير أو ضمن مرحلة العمل نتيجة عطل عتادي[17] . يتم الاعتماد على الفائضية لضمان تسامح العطل في منظومة ما، وهي احتمالية توفر مصادر معالجة أعلى من الاحتياج المقدر للقيام بمهمة ما، وللفائضية أيضاً عدة أنواع: عتادية، وبرمجية، وزمنية. يتم إنجاز الفائضية العتادية من خلال وجود عناصر مادية متاحة ومتطابقة وظيفياً بحيث تمكن من تجاوز العطل في إحدى تلك العناصر [18].

يتم الاعتماد في دراسة الأعطال على مصطلح الوثوقية R(t) وهي احتمالية استمرارية المنظومة بالعمل دون عطل كتابع للزمن، ويرتبط بها الزمن الوسطي للعطل MTTF (main time to failure) وهو الزمن الوسطي لعمل النظام حتى حدوث العطل، والزمن الوسطي بين عطلين MTBF (main time between failure) وهو الزمن الوسطي بين عطلين متلاحقين،

ويتم تحديده بالزمن الوسطي للعطل مضاف اليه الزمن الوسطي للإصلاح (main time to repair) MTTR كما في المعادلة (1):

$$MTBF = MTTF + MTT \ldots..(1)$$

أما الإتاحية A(t) فهي معدل الزمن الوسطي الذي تبقى فيه المنظومة في حالة عمل، ويتضمن هذا الزمن حدوث أعطال غير مؤثرة. ويمكن عندها الوصول الى أنظمة متاحة طويلة الزمن فقط في حال وجود صيانة للعطل، ويشار الى الأنظمة المتاحة طويلة الأمد كما في المعادلة (2) والمعادلة (3):

$$A = \lim_{t\to\infty} A(t) \ldots..(2)$$

$$A = \frac{MTTF}{MTBF} = \frac{MTTF}{MTTF+MTTR} \ldots..(3)$$

3-6- دراسة تسامح الخطأ في شبكة الفراشة

بفرض أن pr هو احتمالية ورود طلب من معالج الى وحدة ذاكرة خلال كل دورة. يتم توجيه هذا الطلب إلى أي من وحدات الذاكرة N ذات الاحتمالية المتساوية $1/N$. ومن ثم، فإنّ احتمال قيام معالج معين بإنشاء طلب لوحدة ذاكرة هو pr/N، بافتراض أن كل معالج يقدم طلباً مستقلاً عن طلباته السابقة، حتى في حال لم يتم تلبية طلبه السابق، فسيقوم المعالج بإنشاء طلب جديد ومستقل. نحسب هذا الاحتمال بمرحلة، بدءًا من المدخلات (المعالجات)، حيث $i=k-1$ مروراً بمراحل التبديل وصولاً إلى المخرجات (الذواكر)، حيث $i=0$. بدءًا من $i = k-1$، سيتم تقسيم طلبات الذاكرة لكل معالج (باحتمالية pr) بالتساوي بين سطري الخرج لصندوق التبديل، الذي يتصل به المعالج. أي أن احتمال أن يحمل خط الخرج لصندوق التبديل في المرحلة $(k-1)$ طلباً تم إنشاؤه بواسطة أحد المعالجات هو $pr / 2$. لأنه يمكن إنشاء طلب على خط الخرج هذا بواسطة أي من المعالجات، $(p_r{}^{k-1})$ هو احتمال اتحاد الحدثين المتناظرين (لكل منهما الاحتمال $pr / 2$). باستخدام قوانين الاحتمال الأساسية يمكن كتابة المعادلة (4) [19].

$$p_r{}^{k-1} = \frac{p_r}{2} + \frac{p_r}{2} - (p_r/2)^2 = p_r - p_r{}^2/4 \ldots\ldots\ldots (4)$$

بافتراض أن q_l تمثل احتمالية حدوث عطل على وصلة ما، فإن $p_l=1-q_l$ هي احتمالية كون الوصلة خالية من الأعطال، عندها فإن احتمالية أن يقوم المبدل بتمرير الطلب من أحد المداخل في المرحلة i−1 الى أحد المخارج في المرحلة i كما في المعادلة (5).

$$p_l\, p_r{}^i/\, 2 \ldots\ldots\ldots(5)$$

وبالتالي فإن المعادلة العودية الناتجة كما في المعادلة (6).

$$p_r{}^{(i-1)} = p_l p_r{}^{(i)} - \left(p_l p_r{}^{(i)}\right)^2/4 \ldots\ldots\ldots (6)$$

في شبكة فراشة بأبعاد $2^k * 2^k$ فإن المسار للوصول بين أي معالج الى أي ذاكرة يتضمن العبور من خلال k مبدل و K+1 وصلة في بنية مكونة من K مرحلة، وبالتالي فإن احتمالية أن يكون زوج المعالج – موقع الذاكرة في حالة اتصال يعطى بالمعادلة (7).

$$(1-q_l)^{k+1}(1-q_s)^k = p_l{}^{k+1} p_s{}^k \ldots\ldots (7)$$

وباعتبار وجود 2^{2k} من الأزواج مداخل/مخارج فإن العدد المتوقع من الأزواج المتصلة يعطى بالمعادلة (8).

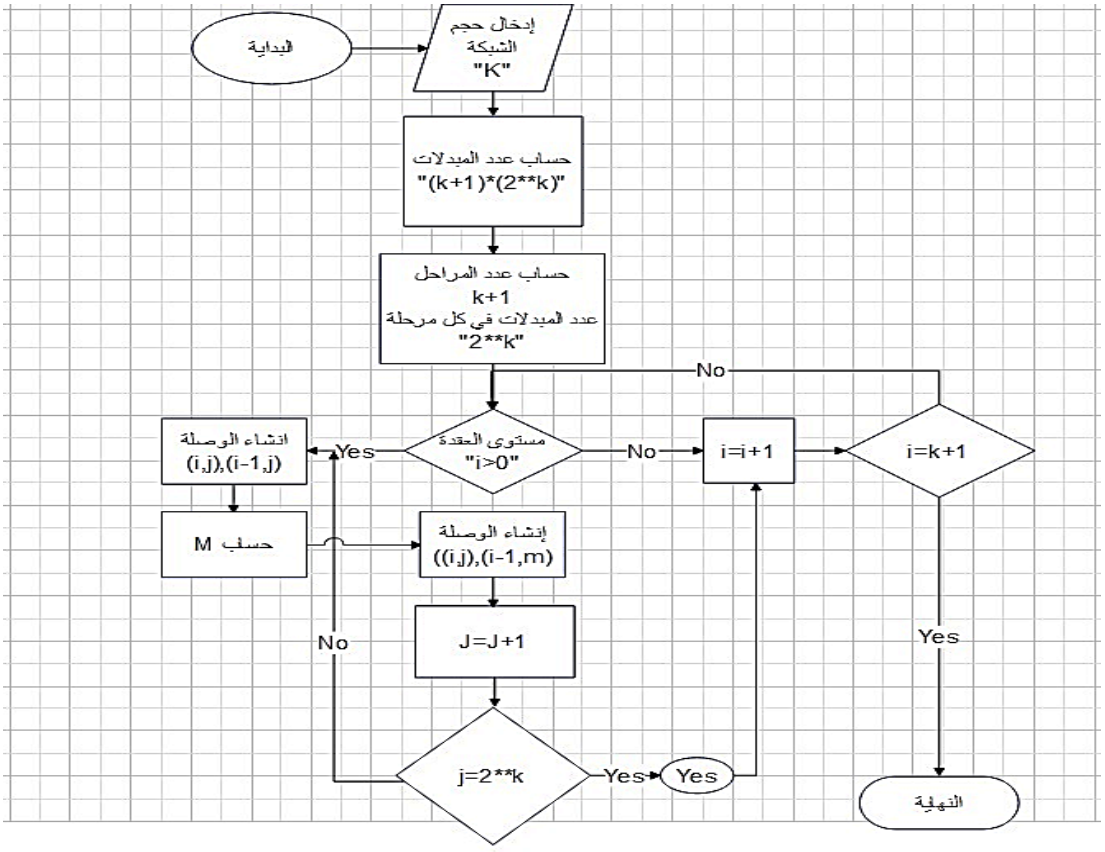
$$Q = 2^{2k} p_l{}^{k+1} p_s{}^k \ldots. (8)$$

4- تنفيذ الشبكة برمجياً

4-1- تنفيذ شبكة الفراشة

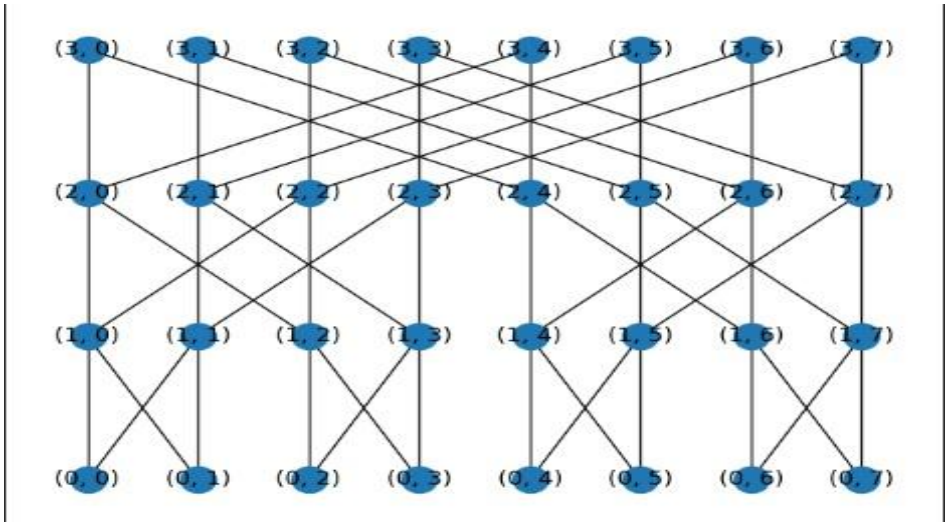
يعتمد إنشاء مخطط شبكة الفراشة بشكل عام على حجم الشبكة K وترتبط جملة البارامترات بشكل مباشر بهذا المعامل. إن التكوين البرمجي للخوارزمية كما هو مبين في الشكل (7). ويمكن تلخيص عمل الخوارزمية على النحو الآتي:

- ينطلق من تحديد قيمة المعامل K من قبل المستخدم
- حساب عدد المبدلات (K+1)*(2^K) وعدد المراحل K+1 وكذلك عدد المبدلات في كل مرحلة 2^k.
- لأجل أي مبدل له احداثيين، الأول i ويدل على المرحلة والثاني j ويدل على ترتيب المبدل ضمن المرحلة ويرمز لكل مبدل الثنائية (j , i).
- لأجل أي مبدل في المرحلة i>0: يرتبط هذا المبدل مع مبدلين في المرحلة i−1 ، احدهما هو المبدل (i−1,j) ويسمى المبدل القريب أو المباشر، ومع المبدل (i−1,M) ويسمى المبدل البعيد.
- يتم حساب M من خلال تغيير البت i (الأكثر أهمية MSB) من المكافئ الثنائي للدليل j.



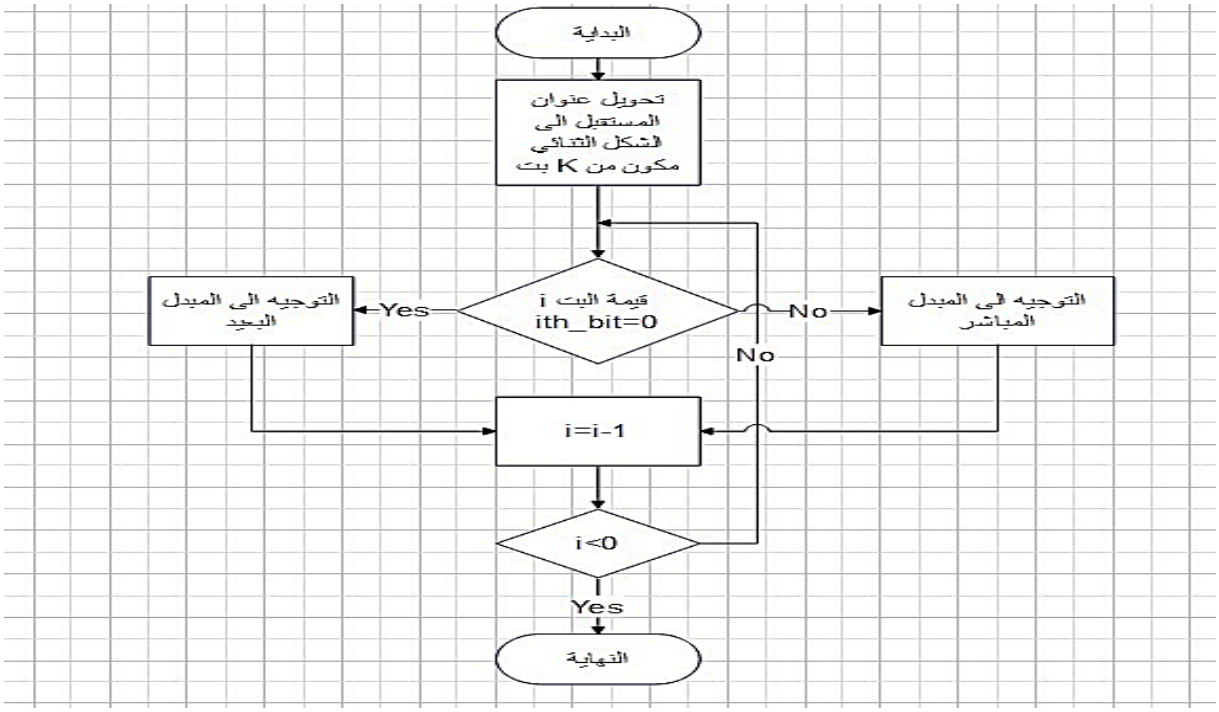
الشكل (7): مراحل بناء مخطط شبكة butterfly من مقطع برمجي بلغة بايثون بحجم يحدده المستخدم.

أما شبكة الفراشة المولدة من خلال البرنامج ومن أجل ثلاث مراحل فتظهر في الشكل (8).



الشكل (8): مخطط شبكة الفراشة بثلاث مراحل مولد باستخدام لغة بايثون

أما آلية توجيه الرسائل من المعالجات الى مواقع الذاكرة موضحة في خوارزمية التوجيه كما في الشكل (9)، حيث يتم اختيار المبدل التالي في المسار في المستوى $i-1$ بالاعتماد على قيمة البت i من المقابل الثنائي لعنوان المستقبل، فإذا كانت قيمة هذا البت 0 يتم اختيار المبدل المباشر والا يتم اختيار المبدل البعيد، وهو ما يضمن مسار واحد محقق للوصول الى أي موقع ذاكرة من أي معالج.



الشكل (9): مخطط يوضح خوارزمية التوجيه في شبكة الفراشة Butterfly.

يمكن قياس بارامترات شبكة الفراشة كما في الجدول (2)، حيث A كلفة المبدل، و S عدد المبدلات و B كلفة الوصلة و L عدد الوصلات.

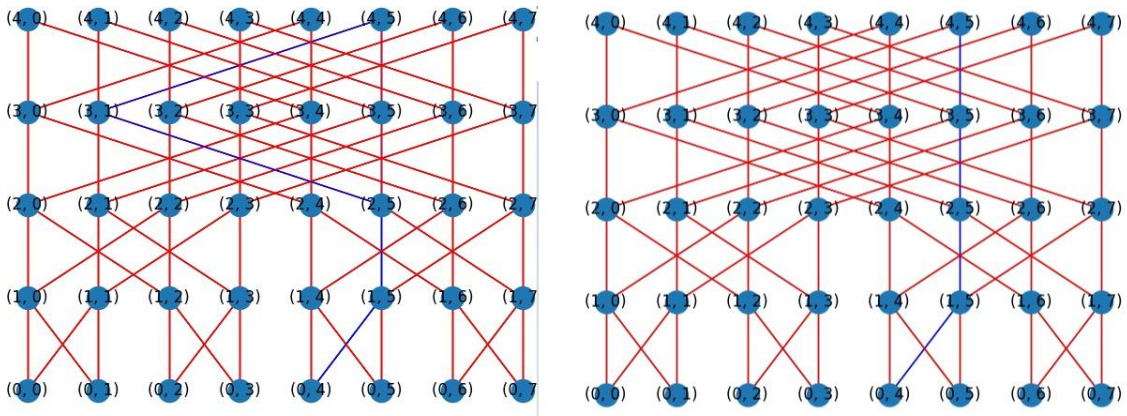
الجدول (2): بارامترات شبكة الفراشة المولدة

Diameter	Cost	Links	Degree
١+$Log_2(N/2)$	A*s+B*L	$Log_2(N/2)$*N	**Switch in/out**
شبكة بحجم 8 مداخل			
3	12A+16L	16	**4**
شبكة بحجم 16 مداخل			
4	32A+48L	48	4
شبكة بحجم 32 مداخل			
5	80A+128L	128	4

4-2- تنفيذ شبكة الفراشة ذات المرحلة الإضافية

تمّ تصميم شبكة الفراشة جديدة بإضافة مرحلة جديدة متطابقة مع المرحلة الأولى في شبكة الفراشة العادية كمخطط توصيل، تُدعى هذه البنية الجديدة Extra stage butterfly، ويضمن هذا الاجراء معالجة حدوث العطل في المرحلة الأولى من الشبكة، من خلال آلية توجيه الآتية:

- عندما لا يتضمن مخطط الفراشة أي عطل فإن التوجيه يتم بين المرحلة الإضافية والمرحلة الأولى وفق المسار المباشر، ويبقى التوجيه في المراحل اللاحقة كما هو في حالة شبكة الفراشة.
- عندما يحدث عطل في المرحلة الأولى (تمّ نمذجته ضمن كود برنامج البايثون) يتم التوجيه وفق المسار البعيد الى المرحلة الأولى، وهذا ما يبينه الشكل (10) المولد من قبل البرنامج، حيث يتم تجاوز حدوث عطل في المرحلة الأولى وتأمين مسار جديد الى المستقبل (المسارات الافتراضية باللون الأحمر، والمسار البديل باللون الأزرق).



الشكل (10) اختيار المسارات البديلة برمجاً في شبكة Extra stage butterfly

يمكن اجراء القياس لمعاملات هذه الشبكة كما في الجدول (3).

الجدول (3): معاملات شبكة butterfly ذات المرحلة الإضافية

Diameter	cost	Links	Degree
$2+Log_2(N/2)$	A*s+B*L	$(Log_2(N/2)+1)*N$	**Switch in/out**
شبكة بحجم 8 مداخل			
4	16A+66B	24	**4**
شبكة بحجم 16 مداخل			
5	40A+64B	64	4
شبكة بحجم 32 مداخل			
6	96A+160B	160	4

5-مناقشة النتائج

وكما هو ملاحظ من الجدول (3) بالمقارنة مع الجدول (2)، فإن تكلفة الشبكة الجديدة وعقد الربط قد ازدادت، مقابل الحصول على شبكة متسامحة مع العطل، يمكن لشبكة الفراشة ذات المرحلة الإضافية أن تحقق تسامح خطأ وحيد في المرحلة الأولى من مخطط التوصيل، يقابل ذلك زيادة المسار بطول قفزة واحدة وزيادة الكلفة مع ازدياد حجم الشبكة. تعتبر شبكة الفراشة ذات كلفة منخفضة وتأخير قليل لكنها غير مناسبة للتطبيقات التي تتطلب استمرارية العمل، حيث أنها غير متسامحة مع العطل، وقد أمكن تطوير شبكة الفراشة بإضافة مرحلة (تكرار المرحلة الأولى) كشكل من أشكال الفائضية للحصول على تسامح خطأ وحيد، بالمقابل أدى ذلك الى ارتفاع كلفة الشبكة، وزيادة التأخير، كما أظهرته الجداول، حيث يمكن اعتماد الجداول والمقارنة مع قيم وبارامترات مخططات توصيل أخرى، عند الحاجة الى تصميم شبكة بحجم معين، مما يساعد في الاختيار الأنسب لنوع المخطط من حيث الكلفة والتأخير ودرجة تسامح الخطأ المطلوب.

- اعتمدت الدراسة [20] على دراسة بارامترات عدة أنماط من شبكة الفراشة، وعند أحجام مختلفة للشبكة، وتوصلت أن النمط Torus-Butterfly يحقق كلفة أقل من النمطين Hyper-Butterfly و Torus-embedded-Hypercube ، كذلك بينت الدراسة ارتفاع كلفة وأبعاد أي نمط من أنماط شبكة الفراشة عند زيادة عدد المعالجات، لم تذكر الدراسة [15] تأثير الفائضية على حجم الشبكة والكلفة اذ تتطلب الفائضية إضافة عناصر الى الشبكة تهدف الى تجنب فشل مخطط التوصيل عند حدوث الأعطال في عناصر الشبكة. بين البحث دراسة تأثير مرحلة إضافية الى شبكة الفراشة كنوع من أنواع الفائضية مما يؤمن تسامح عطل وحيد في المرحلة الأولى من مخطط الفراشة، وبين ارتفاع الكلفة والأبعاد لنموذج شبكة الفراشة المتسامح مع العطل، من خلال خوارزمية توجيه معدلة عن خوارزمية التوجيه الخاصة بشبكة الفراشة، وتم اجراء التنفيذ البرمجي لخوارزميات التوجيه ورسم المخططات وقياس بارامترات الشبكات من خلال لغة بايثون، حيث أنها توفر المكتبة NetworkX المتخصصة برسم الشبكات وقياس معاملاتها.

6-التوصيات والأعمال المستقبلية

قمنا في هذا البحث بعرض البنية التفصيلية لشبكة الفراشة، ودراسة طرق تحسين التسامح مع العطل في هذه الشبكة، من خلال استخدام لغة بايثون لرسم الشبكات بمختلف الأحجام ودراسة خوارزمية الإنشاء والتوجيه ضمن شبكة الفراشة، واستهدف البحث دراسة إحدى طرق تسامح العطل في شبكة الفراشة بالاعتماد على الفائضية من خلال إضافة مرحلة الى بنية الفراشة ودراسة تأثير المرحلة الإضافية على كل من: قياس الشبكة والكلفة وعدد الوصلات .بينت الدراسة أن جملة بارامترات الشبكة ترتبط بالمعامل K والذي يحدد حجم الشبكة، فيما يخص التسامح بالعطل فقد بينت الدراسة بالقوانين الاحتمالية وبالتنفيذ البرمجي أن خطأ وحيد في وصلة من الوصلات يؤدي الى حدوث فشل في توصيل الرسائل بين المعالج وموقع ذاكرة، وبينت أن استخدام شبكة الفراشة ذات المرحلة الإضافية يمكِّن من استمرار عمل مخطط الفراشة في حال حدوث عطل في المرحلة الأولى من الوصلات، من خلال خوارزمية توجيه تعمل على توجيه الرسائل من خلال مسار بديل، بالمقابل فإن ذلك يؤدي الى زيادة في عدد الوصلات والكلفة الإجمالية للشبكة وتزداد هذه الكلفة بزيادة حجم الشبكة، وكذلك زيادة أبعاد الشبكة بمقدار قفزة، مما يعني زمن أطول في توصيل الرسائل بين المعالج والذاكرة.

يمكن تطوير البحث في الدراسات القادمة من خلال المحاور الآتية:

- دراسة حدوث العطل في المراحل التالية حيث تمت الدراسة على حدوث عطل في المرحلة الأولى من شبكة الفراشة.
- حساب احتمالية حدوث الأعطال في عقد التبديل، حيث اعتمد البحث على دراسة احتمالية العطل في الوصلات.
- دراسة أساليب موازنة الحمل في تطوير تسامح العطل، اذ اعتمدت الدراسة على الفائضية لضمان تسامح الخطأ.

6-المراجع

[1] Nengzhi, J., Jianwu, Z., Haili, X., Xiaoning, W., & Yulin, S. (2020). Comparative research on high-speed networks of high performance computing cluster based on MPIGRAPH. 2020 IEEE 6th International Conference on Computer and Communications (ICCC). https://doi.org/10.1109/iccc51575.2020.9344976

[2] Xia, Y., & Zhou, L. (2022). Improved clustering algorithm based on hypercube. *2022 International Conference on Machine Learning, Control, and Robotics (MLCR)*. https://doi.org/10.1109/mlcr57210.2022.00015

[3] Xuejun Yang, Yunfei Du, Panfeng Wang, Hongyi Fu, & Jia Jia. (2009). FTPA: Supporting fault-tolerant parallel computing through parallel Recomputing. *IEEE Transactions on Parallel and Distributed Systems*, *20*(10), 1471-1486.
https://doi.org/10.1109/tpds.2008.231

[4] Du, Y., Wang, P., Fu, H., Jia, J., Zhou, H., & Yang, X. (2007). Building single fault survivable parallel algorithms for matrix operations using redundant parallel computation. *7th IEEE International Conference on Computer and Information Technology (CIT2007)*.
https://doi.org/10.1109/cit.2007.2

[5] Suganya, K., & Nagarajan, V. (2011). Efficient run-time task allocation in reconfigurable multiprocessor system-on-Chip with network-on-Chip. *2011 International Conference on Computer, Communication and Electrical Technology (ICCCET)*.
https://doi.org/10.1109/icccet.2011.5762458

[6] Villanueva, J. C., Flich, J., Duato, J., Eberle, H., Gura, N., & Olesinski, W. (2009). A performance evaluation of 2D-mesh, ring, and crossbar interconnects for chip multi-processors. *Proceedings of the 2nd International Workshop on Network on Chip Architectures*.
https://doi.org/10.1145/1645213.1645226

[7] K K, Jena S, T VG. Performance Evaluation, Comparison and Identification of Efficient Hypercube Interconnection Networks. Research Square; 2021.
DOI: 10.21203/rs.3.rs-287722/v1.

[8] Shi, W., & Srimani, P. (n.d.). Hyper-butterfly network: A scalable optimally fault tolerant architecture. *Proceedings of the First Merged International Parallel Processing Symposium and Symposium on Parallel and Distributed Processing*, 1998.
https://doi.org/10.1109/ipps.1998.670008

[9] Xu, B., Liu, Z., Yu, D., Xu, Y., & Huang, M. (2018). Asymmetric redundancy: An enhanced redundancy strategy for MLC NAND flash memory storage systems. *2018 Eighth International Conference on Instrumentation & Measurement, Computer, Communication and Control (IMCCC)*.
https://doi.org/10.1109/imccc.2018.00075

[10] Garcia-Astudillo, L. A., Lindoso, A., Portela, M., & Entrena, L. (2020). Evaluation of a reduced precision redundancy FFT design. *2020 XXXV Conference on Design of Circuits and Integrated Systems (DCIS)*.
https://doi.org/10.1109/dcis51330.2020.9268634

[11] Zhang, H., Huang, N., & Liu, H. (2014). Network performance reliability evaluation based on network reduction. *2014 Reliability and Maintainability Symposium*.
https://doi.org/10.1109/rams.2014.6798440

[12] L. Murray, H. Cancela, G. Rubino, A splitting algorithm for network reliability estimation, IIE Transactions 45 (2) (2013)
https://doi.org/10.1080/0740817X.2012.677574

[13] Bistouni, F., & Jahanshahi, M. (2020). Impact of raising switching stages on the reliability of interconnection networks. *Journal of the Institute of Electronics and Computer*, *2*(1), 93-120. https://doi.org/10.33969/jiec.2020.21007

[14] Tutsch, D., & Hommel, G. (2002). Generating systems of equations for performance evaluation of multistage interconnection networks. *Journal of Parallel and Distributed Computing*, *62*(2), 228-240. https://doi.org/10.1006/jpdc.2001.1776

[15] Bistouni, F., & Jahanshahi, M. (2016). Reliability analysis of fault-tolerant bus-based interconnection networks. *Journal of Electronic Testing*, *32*(5), 541-568. https://doi.org/10.1007/s10836-016-5601-5

[16] Anjana, A. (2020). On improving fault tolerance of IoT networks through butterfly networks implemented at services layer. *International Journal of Advanced Trends in Computer Science and Engineering*, *9*(2), , 2096-2115.
https://doi.org/10.30534/ijatcse/2020/184922020

[17] Wunderlich, H. (2009). Software-based hardware fault tolerance for many-core architectures. *2009 24th IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems*.
https://doi.org/10.1109/dft.2009.36

[18] Simone, L. D., Mauro, M. D., Longo, M., Natella, R., & Postiglione, F. (2023). Multi-provider IMS infrastructure with controlled redundancy: A Performability evaluation. *IEEE Transactions on Network and Service Management*, 1-1.
https://doi.org/10.1109/tnsm.2023.3282745

[19] Moazez, M. (2012). Design and implementation of multistage interconnection networks for soc networks. *International Journal of Computer Science, Engineering and Information Technology*, *2*(5), 1-11. https://doi.org/10.5121/ijcseit.2012.2501

[20] Latifah, L., Ernastuti, E., & Kerami, D. (2012). Embeddings on torus-butterfly interconnection network. *International Journal of Applied Information Systems*, *4*(9), 38-41. https://doi.org/10.5120/ijais12-450817