Vol.2009-OS-112 No.17
2009/8/6

非対称な形状に適応する高バンド幅 multi-link Ethernet

米元大我[†1] 三浦信一[†2] 塙敏博[†1,†2]
朴泰祐[†1,†2] 佐藤三久[†1,†2]

我々はこれまで RI2N と呼ばれる Gigabit Ethernet を用いた高性能・耐故障性のあるマルチリンクネットワークシステムの開発を行ってきた．RI2N は PC クラスタにおける MPI 通信や，NFS のような一般的な UNIX ネットワークサービスにおいて，高バンド幅と耐故障性を提供することができる．本稿では，RI2N の最適化バージョンとして RI2N+を提案する．RI2N+は，システムのネットワークコストバランスを柔軟に調整できる，非対称構造のマルチリンク接続をサポートする．このようなネットワーク構造は，Linux の標準ディストリビューションとして広く使われている Linux Channel Bonding においてサポートされていない．RI2N+は非対称なネットワーク構造を自動的に検出し，リンク毎のトラフィック制御を行う．従来の RI2N でも非対称マルチリンクネットワークには対応可能であるが，RI2N+ではトラフィック制御の最適化により，このような構成のネットワークにおいて RI2N に比べ最大 30%の性能向上を達成する．また，RI2N+は非対称構造のネットワークにおいて，対称構造のネットワークに比べ最大 86%の性能を達成可能であることが確認された．

Flexible Multi-link Ethernet Binding System for PC Clusters with Asymmetrical Topology

Taiga Yonemoto,[†1] Shin'ichi Miura,[†2]
Toshihiro Hanawa,[†1,†2] Taisuke Boku[†1,†2]
and Mitsuhisa Sato[†1,†2]

We have been developing a multi-link binding network system for Ethernet named RI2N for high-throughput and fault-tolerant interconnection with Gigabit Ethernet. It can be used both for inter-node communication in MPI programs and traditional UNIX network services such as NFS. In this paper, we propose an optimized version of RI2N, named RI2N+, which allows asymmetrical multi-link connection for fitting to various cost-effective and flexible system configuration. Such a configuration cannot be supported by Linux Channel Bonding which is widely used in standard Linux distributions. RI2N+ automatically detects the asymmetric network configuration and controls the traffic distribution to multiple links. In basic performance evaluation under high traffic rate, we confirmed that the throughput of network with our proposed scheme is improved up to approximately 30% to that of original RI2N. RI2N+ also keeps high performance even in asymmetrical configuration with up to 86% of relative performance to the symmetrical case.

1. はじめに

近年，PC クラスタは，コストパフォーマンスの高さから HPC (High Performance Computing) において多くの局面で用いられている．そのようなクラスタ間のネットワークはシステム全体の性能を支える上で非常に重要である．SAN (System Area Network) と呼ばれる InfiniBand[1] や Myrinet[2] が存在するものの，MPI を利用した並列計算，NFS やリモートログイン，FTP といった従来の UNIX ネットワークサービスにおいて Ethernet は今もなお広く使われている．特に Gigabit Ethernet (GbE) は適度な性能と，安価な NIC やスイッチを提供する現在最もコストパフォーマンスの優れた Ethernet である．TOP500[3] にランクインするシステムの内，50 %を超えるシステムがクラスタ間ネットワークとして Ethernet を利用しており，その上，小規模な PC クラスタの多くは GbE と 24 ポート程度の安価な GbE スイッチで構成されている．

GbE はコストパフォーマンスの高いシステムを実現できる一方で，スループットやレイテンシの絶対的な性能は，InfiniBand DDR や QDR に比べて劣る．これら 2 つの問題の内，レイテンシの問題を解決することは非常に難しいが，スループットに関しては，複数の GbE リンクを論理的な 1 つのネットワークとして使用することで向上できる．Linux Channel Bonding[4] (以下 LCB) は今日の標準 Linux ディストリビューションに含まれており[5][6]，複数の Ethernet リンクの平行結線により上記コンセプトを実現する．LCB の balance-rr mode[4] は 1 ノード当りのリンク数を増加させることでバンド幅を向上させる．また，balance-rr モードを構成するリンクの 1 つが故障したとき，自動的にリンクの選択を行う耐故障性を併せ持つ．しかし，大きな HPC クラスタで LCB を利用するには，いくつ

†1 筑波大学大学院 システム情報工学研究科
Graduate School of Systems and Information Engineering, University of Tsukuba
†2 筑波大学 計算科学研究センター
Center for Computational Sciences, University of Tsukuba

かの問題がある．まず初めに，ネットワークの構成をシステムで利用する全ノードに予め登録しておかなければならない．また，現在の LCB ではノード数が最大 16 台に制限されることや，一般的なネットワークサービスに用いられるメッセージサイズにおいて，比較的スループットが低いことも問題点となっている[7]．

我々はこれまで，大規模な HPC クラスタで利用可能なマルチリンク Ethernet によるネットワークシステムとして，RI2N (Redundant Interconnection with Inexpensive Network) を提案してきた[8]．RI2N/DRV[7)9)10)] と呼ばれる最新のバージョンではネットワークドライバレベルの実装がされており，完全にユーザ透過であるため，アプリケーションレベルで一般的な Ethernet と互換性があり，あらゆる Ethernet サービスやプロトコルに適用できる．システム構成定義としては各ノードの NIC 数のみを登録すれば良いため，使用できるノード数に上限はなく，様々なネットワークトラフィックパターンにおいて LCB より高い性能を持つ．

これまで RI2N/DRV はネットワーク全体が全て完全な平行結線で接続されている「対称」なマルチリンク接続を対象としてきた．しかし，システムの潜在的な能力として，非対称なマルチリンクネットワークに対しても適用可能であることが分かっている．ここで，「非対称なマルチリンクネットワーク」とは，一つの論理的なチャンネルを構成するリンク数が各ノードで異なることを言う．例えば，大きな PC クラスタシステムにおいて性能と費用をバランスさせることを考えた場合，重要なサーバノード間は高バンド幅と耐故障性を持たせるためにマルチリンクで接続し，クライアントノードは費用を抑えるためにシングルリンクで接続を行うという構成が想定される．このような場合，クライアントとサーバの通信はシングルリンクで行い，サーバとサーバの通信はマルチリンクで行う．RI2N/DRV はオリジナルの機能としてこのようなネットワーク通信をサポートするが，非対称なネットワークではトラフィックの不均衡により性能向上が妨げられていることが分かった．そこで我々は RI2N/DRV を改良することで，非対称なネットワークへ適応させ，さらに性能の向上を図った．本稿では，この新システムの設計と実装，及び性能評価について述べる．

2. RI2N/DRV の非対称なネットワークへの適用

本節では RI2N/DRV の非対称なネットワークにおける利用を考える．まず初めに，RI2N/DRV のコンセプトと実装を簡単に説明し，非対称なネットワークで利用する際の問題点を解明する．

2.1 RI2N/DRV

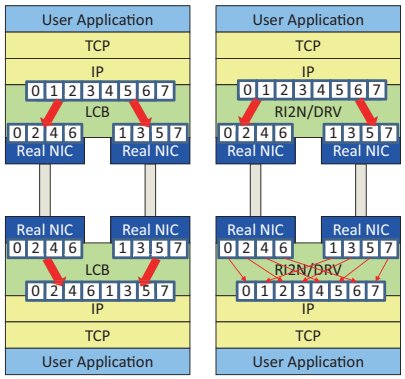
RI2N/DRV[7)9)10)] は高バンド幅・耐故障性を実現するために複数の Ethernet NIC をバインドした仮想的なネットワークデバイスである．これはリンクアグリゲーションを意味し，LCB[4] の baance-rr モードと似ている．しかし，LCB において TCP のような高レイヤプロトコルを用いた場合，パケット順序入れ替えが生じ，深刻な性能低下を招く．

LCB の balance-rr モードでは，連続した Ethernet パケットは複数のリンクへ交互にラウンドロビンで送信される．ここで，各 Ethernet リンク上ではパケット順序が不連続なものとなる．受信側のノードにおいても，各 NIC は不連続な Ethernet パケットを受信する．ハードウェア割り込みの回数を減少させるため，Linux デバイスドライバでは NAPI[11] や interrupt coalescing[12] 技術が導入されており，このような状況では NIC により起動された割り込みハンドラは不連続なパケットを一度に取り扱う．そして，上位層である TCP のプロトコルハンドラにそのままの形で渡される (**図 1(a)**)．TCP レイヤではそれらのパケットのシーケンス番号が抜けているように見えるため，大量のパケットロスが生じたように観測される．そのため実際にはパケットロスは生じていないにもかかわらず，最終的に多くの ACK パケットが再送要求として送信側へ送り返さる．これらの不必要な ACK パケットはトラフィックの妨げとなり，全体の性能は大幅に低下する．

RI2N/DRV はこの問題を次の方法で解決する (**図 1(b)**)．ここで，使用する NIC の数を 2 と仮定する．NIC にパケットが到着すると，IP のようなネットワークプロトコルハンドラに代わって，初めに RI2N/DRV ハンドラがパケットを取得する．RI2N/DRV ハンドラでは，一方の NIC からのパケットストリームを受信した後も，他方の NIC のパケットストリームを待って，一定期間パケットを保持する．他方の NIC でパケットが受信されると，元のパケットストリームを再構成するため，パケットをシーケンス番号通りに並べ替える．このとき物理的なパケットロスがなければ，2 つのパケットストリームから結合されたこのパケットストリームは元のパケットストリームと一致する．このように，リオーダリングを行うことで上位レイヤでのパケット再送要求を抑制することができる．本機構を用いることで，多くのトラフィックパターンと一般的なアプリケーションにおいて RI2N/DRV の性能は LCB を上回っていることを確認した．

RI2N/DRV のもう一つの特徴は，リンク故障とリンク回復の自動検出である．リンクが故障すると大量のパケットロスが生じるため，できるだけ早い故障検出が必要となる．RI2N/DRV ではノード同士が通信を行うとき，通信を行うノードとリンクの組み合わせを「endpoint」と呼ぶ．**図 2** に endpoint の例を示す．ノード A がデュアルリンクのネッ

Vol.2009-OS-112 No.17
2009/8/6



(a) LCB (b) RI2N/DRV

図 1 LCB におけるパケット順序入れ替え発生 (a) と RI2N/DRV におけるパケット順序整合 (b)

トワークを通して 3 つのノード B, C, D と通信を行う場合，ノード A は 6 つの通信相手 endpoint を持ち，endpoint 毎にネットワークの統計量を管理する．ネットワークの故障は，各 endpoint で受信したパケット量を監視することによって検出する．もし同一の相手ノードに対して複数の endpoint 間で，受信パケット数に大きな差が生じ，その差が一定の閾値を超えたなら，RI2N/DRV はパケットの受信が少なかったほうのリンクを故障と判断し，故障したリンクの利用を自動的に停止する．リンク故障後に受信パケット数のカウントを行うことはできないため，リンク回復を検出するために「ハートビートパケット」を用いる．通常，NIC の故障回復はケーブルやスイッチを人の手によって復旧させる．よって，ハートビートパケットの送信間隔は数秒に 1 度といった低い頻度で十分であり，ハートビートパケットが通常のトラフィックに与える影響はほぼ無視できる．

RI2N/DRV の実装では，ドライバレベルで Ethernet の全てのプロトコルに対応するため，Ethernet パケットのヘッダにおいて，パケットタイプが変更され，いくつかの制御情報が付け加えられている．加えて，受信側のノードの受信手続きは，追加された情報を適切に取り除く必要がある．以降，RI2N/DRV を単に RI2N と呼ぶこととする．RI2N/DRV の詳しい説明については 7)9)10) を参照されたい．

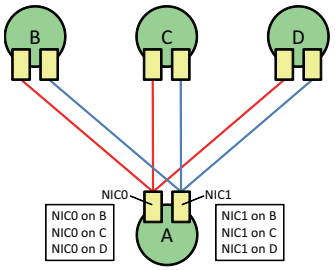


図 2 Node-A 上の通信相手 endpoint 管理情報

2.2 非対称なネットワーク

本来，LCB と RI2N は全てのノードにおいて同じリンク数が使われるものとして設計されている．また，近年の CPU とネットワーク性能のバランスを考えると，2 リンクの GbE が最も良いコストパフォーマンスが得られる．しかし，クラスタのノード数が増えてきた場合，全ノードに 2 枚の NIC を刺すことや，スイッチ，ケーブルのコストまで含めて考えると，費用面の負担が増加する．クラスタシステムを用途別に複数のパーティションに分けるような場合には，各ノードに必要となる NIC の数が変更することで柔軟でコストパフォーマンスの良いシステム構成が可能となる．このようなネットワーク構成を「非対称なマルチリンクネットワーク」と呼ぶことにする．**図 3** に非対称なネットワーク構成の例を示す．LCB では必ず全ノードが同一のリンク数を取らなければいけないため，このようなネットワークを構築することはできない．一方で RI2N には，そのような制限がなく，ノード毎にリンク数が異なるような，あらゆるネットワークを構築可能である．ただし，各ノードにおいて片方のリンクからのパケットを他方のリンクに振り分けるような制御は行われないため，非対称ネットワークの場合でも，どちらか一方のネットワークは完全に Ethernet のネットワーク接続構造を維持する必要がある．以降，本稿における非対称ネットワークはこれを必須条件とする．

RI2N が非対称なネットワークへ適用できることを確かめるために，**図 4** に示すネットワーク構成で簡単な測定を行った．測定環境を**表 1** に示す．図 4 中の各図において，灰色のノードはシングルリンクで，白色のノードはマルチリンクでサーバに接続されている．また，各図において上半分のノードはクライアントであり，下半分のノードはサーバである．以後，Switch-i に接続されるリンクを各ノード上の Link-i と呼び，全ノードの Link-i で構成されるネットワークを Network-i と呼ぶ．

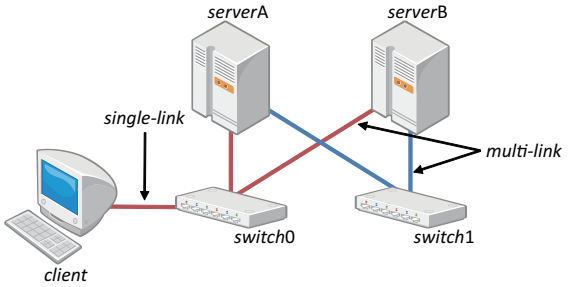


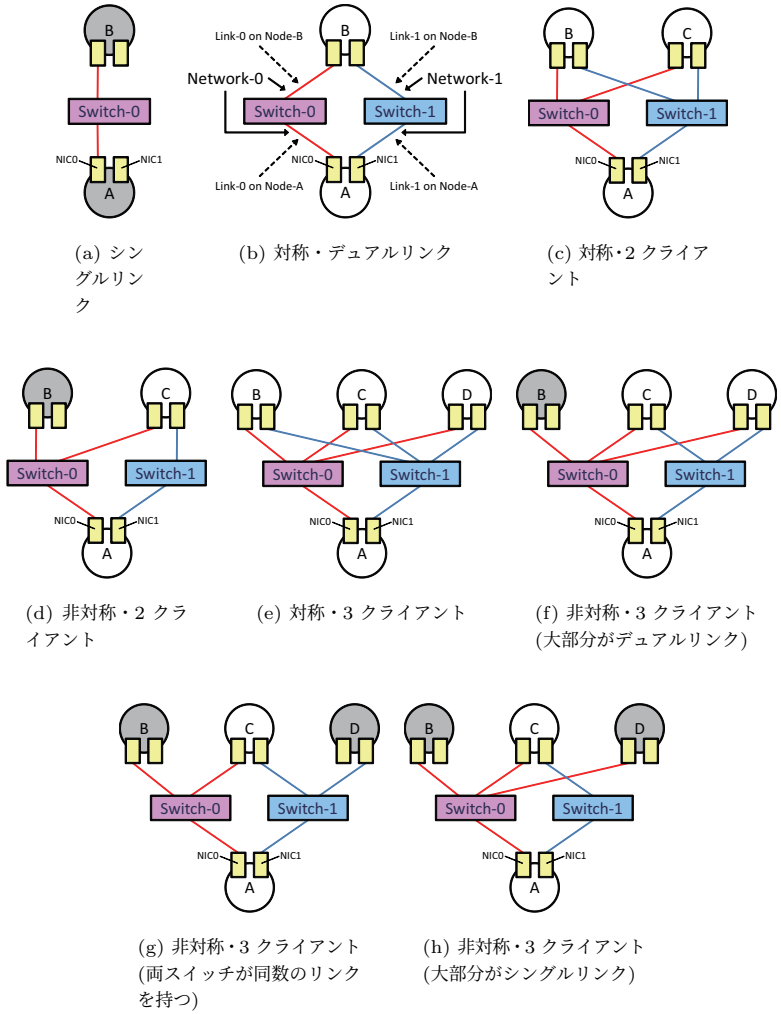
図 3 非対称なネットワーク構造によるマルチリンク Ethernet 接続

表 1 実験環境

Item	Specification
CPU	Xeon 5110 Dual-Core 1.6GHz
Memory	DDR2 2048MB
Kernel	2.6.27.9-73.fc9.x86_64
NIC	Intel PRO/1000PT dual port 1000base-T
Switch	Dell PowerConnect 5324 (24 ports Gigabit Ethernet switch)

これらの各ネットワーク構成において，クライアントからサーバに対し，バースト的な単方向連続通信実験を行う．サーバはデュアルリンクの Ethernet を持つため，理論上パケット受信時の最高スループットはシングルリンクのときに比べ，2 倍となる．**表 2** に結果を示す．表から分かるように，非対称ネットワークにおけるスループットは，**図 4(b)** のような対称ネットワークに比べ低くなっている．特に**図 4(h)** に示す，2 台のクライアントがシングルリンクで接続され，1 台のクライアントがデュアルリンクで接続されているような，非対称性が強い構成では最もスループットが低い結果となった．

この結果は以下のようにして起こると考えられる．例えば**図 4(d)** の場合，Node-B からのトラフィックは全て Network-0 を通るが，Node-C からのトラフィックは Network-0 と Network-1 に均等に分かれる．Node-B と Node-C がバースト転送を行う今回の場合，Switch-0 の負荷は Switch-1 に比べて増大する．Ethernet スイッチにおいて，同一送信先への大量のトラフィックはパケットロスを生じさせるため，送信側ではパケットロスを減らそうとして輻輳制御が働き，ウィンドウサイズは急速に縮小される．従って，この場合，Node-B と Node-C は Switch-0 におけるパケットロスに起因して輻輳制御の影響を受ける．



(a) シングルリンク　(b) 対称・デュアルリンク　(c) 対称・2 クライアント

(d) 非対称・2 クライアント　(e) 対称・3 クライアント　(f) 非対称・3 クライアント (大部分がデュアルリンク)

(g) 非対称・3 クライアント (両スイッチが同数のリンクを持つ)　(h) 非対称・3 クライアント (大部分がシングルリンク)

図 4 様々な対称／非対称構成のマルチリンクネットワーク

表 2 従来の RI2N の性能

Topology in Fig. 4	Throughput [MB/s]
(a)	112.2
(b)	223.1
(c)	213.8
(d)	142.9
(e)	211.2
(f)	160.6
(g)	195.7
(h)	120.4

Node-B はシングルリンクでのみ接続されているため，輻輳制御の影響を受けるのは当然である．一方の Node-C は別に Network-1 があるにも関わらず，Node-B と同様の輻輳制御の影響を受ける．理論上，Node-C はパケットを Network-1 のみでも送信できる．しかし，RI2N は Network-0 と Network-1 の両方のネットワークをラウンドロビンで，同等にパケットを送信してしまうため，Network-1 のトラフィックも，Network-0 でのパケットロスが原因で縮小されたウィンドウサイズに制限されてしまう．

ここまでをまとめると，様々な構成を持つネットワークに対して RI2N は適用可能であるが，大量のトラフィックを転送する場合には全体のスループットはネットワークが非対称性が増すになるにつれて減少する．従って，リンクの追加によってバンド幅が増強されているにもかかわらず効果が得られていないことになる．

3. RI2N による動的接続状態関知

本稿では前節で示した問題を解決するために，RI2N の改良を行う．非対称なネットワークで RI2N が非効率である理由は，混雑しているリンクとスイッチにおいて輻輳制御が働いてしまうためであった．ここで図 4(d) のようなネットワーク構成における，解決策を考える．Switch-0 での輻輳制御を避けるためには，Node-A に接続されている Link-0 と Link-1 を適切なバランスで利用すれば良い．これは Node-C の 2 つのリンクが必ずしも同量のパケットを送信しないことを意味する．RI2N の現在の実装においては，Node-C の各リンクはラウンドロビンに従って交互にパケットを送信する．よって，ネットワーク構成の非対称性に従い，トラフィックのバランスを変えるようにする必要がある．

新たな RI2N ではネットワーク全体の構成を記述したような複雑な設定ファイルを必要とせず，従来の RI2N と同様に，通信ノード間で接続情報を交換し合う．例えば Node-B と Node-C の間に通信が発生していない場合，Network-0 は Node-B と Node-C に共有され，Network-1 は Node-C に占有されることを Node-A は知っておく必要がある．こういった情報の取得は，動的かつ自動的に RI2N によって行われる．この接続情報交換メカニズムを以後,「動的接続状態関知」と呼ぶ.

接続情報の管理を動的に設計した理由は 2 つある．

(1) システムにネットワーク構成等を記述した設定ファイルを登録する必要がなく，管理が容易である．

(2) NIC やケーブル，スイッチに故障が起きた場合，その情報を動的にトラフィック制御へ反映させることができる．

もちろん，非対称なネットワークにおいてシングルリンク接続のネットワークで故障が起きた場合，トラフィックの回復をすることはできない．冗長なネットワークを設定するかどうかはユーザ次第である．従って，本改良においても従来の RI2N と同様にバンド幅の向上と耐故障性の実現は重要となる．

4. RI2N+の実装

本節では動的接続状態関知に対応した改良版の RI2N の実装について述べる．以後，改良版 RI2N を RI2N+と呼ぶ.

4.1 設計概要

RI2N では各ノードに複数の NIC があり，ノード毎の endpoint 数 (=NIC 数) は同じである．RI2N+では通信相手のノードがいくつの endpoint を持っているかを自ノードの endpoint 毎に監視している．例えば図 4(d) の場合，Node-A における Network-0 の endpoint は通信相手の Node-B と Node-C の endpoint について '2' という情報をカウンタに持っており，Network-1 に関しては Node-C の endpoint について '1' という情報を持つ．このカウンタの情報は後述するシステムのパラメータによって，一定期間保たれる．カウンタの値はハートビートパケットに埋め込まれ，通信相手の endpoint へと送られる．2.1 節で述べたように，ハートビートパケットはそこまで高くない頻度で定期的に送信される．従来の RI2N のハートビートパケットはリンクの生存情報だけを持っていたが，RI2N+のハートビートパケットはこれに加え，いくつの endpoint が 1 つのスイッチを共有しているかという情報を伝える役割も持つ.

本手法では送信側のノードがスイッチの混雑を判断し，制御を行う．そのため，自身が持つ各 NIC に送信パケットを単純なラウンドロビンでは送出しない．RI2N+では，自ノー

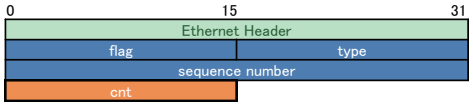


図 5 RI2N+におけるハートビートパケット拡張

ドの各 endpoint に「weight」という概念を導入し，この weight 値に従ったパケット割り当てを行うことで，全体のトラフィックのバランスを維持する．weight を決定するアルゴリズムには様々な物が考えられるが，今回は endpoint のカウンタ値に反比例するように weight を決定する，最も簡単なアルゴリズムを採用した．例えば図 4(d) の場合，Node-A から Node-C へ伝えられた Link-0 と Link-1 の情報は「2:1」であり，それに従って Node-C は「1:2」の割合で Link-0 と Link-1 へ送信パケットを割り当てる．このトラフィック制御により，非対称なネットワークにおける全体のトラフィックは平均化される．

4.2 ハートビートパケットと送信方式の改良

図 5 は RI2N+で用いられるハートビートパケットの拡張を示す．「sequence number」フィールドは TCP のような上位レイヤとは独立に，RI2N レベルでのパケット順序を保持する．RI2N+で新たに付け加えられた「cnt」フィールドは前節で述べた endpoint カウンタを保持する．本手法では endpoint カウンタ情報をハートビートパケットに同梱し，各 NIC の weight はハートビートの間隔で更新される．

例えば**図 4(f)** に示すネットワーク構成の場合，Node-B, Node-C, Node-D は Node-A の endpoint カウンタから weight を計算し，**表 3** と**表 4** に示す情報を持つことになる．ノードがパケットを送信する際，RI2N+はこれらの weight に比例して各 NIC にパケットを割り当てる．この送信割合は次のハートビートパケットが届くまで維持される．

ここで，ハートビートパケットによる endpoint カウンタ情報の非同期更新には少し問題がある．例えば図 4(d) のネットワーク構成において，Node-C は Node-A から Network-0 と Network-1 経由で 2 つのハートビートパケットを受け取る．Network-0 と Network-1 は独立しているため，これら 2 つのハートビートパケットは同時に届く保証がない．よって，Node-A の endpoint カウンタ情報は両ハートビートパケットが Node-C へ届き，処理されるまで正しい情報が反映されない．しかし，我々はこの非同期問題は非常に小さく，無視できると考える．現段階では詳細な解析は行っておらず，今後の課題である．

表 3 構成 (f) における Node-B 上の endpoint 管理情報

Device	# of links	Weight
NIC0	3	-

表 4 構成 (f) における Node-C と Node-D 上の endpoint 管理情報

Device	# of links	Weight
NIC0	3	2
NIC1	2	3

表 5 RI2N+におけるスループットの向上

Topology	Throughput [MB/s]		Ratio [%]
	RI2N/DRV	RI2N/DRV+	
(b)	223.1	223.1	100.0
(c)	213.8	213.9	100.0
(d)	142.9	161.7	113.2
(e)	211.2	210.3	99.6
(f)	160.6	181.6	113.1
(g)	195.7	195.2	99.7
(h)	120.4	153.8	127.8

5. 性能評価

本節では，RI2N+の性能評価を述べる．LCB は非対称なネットワークをサポートしていないので，RI2N と RI2N+を様々なネットワーク構成で比較する．本評価で用いる測定環境は表 1 に同じである．また，評価に使用するネットワーク構成も図 4 のものと同じである．

5.1 平均スループット

まず，図 4 に示すネットワーク構成においてクライアントからサーバへ片方向のバースト転送を行い，その平均スループットを比較する．

表 5 に結果を示す．以後，図 4 に示す各ネットワーク構成を単に (a)〜(h) と言うこととする．また,「スループット」とは Node-A で観測した 1 分間のスループットの平均を言い,「ratio (性能比) 」とは RI2N と RI2N+を比較した時の相対性能を言う．なお，(a) は単にシングルリンクの接続を行ったネットワークであり，評価からは省略する．

初めに，対象なネットワーク構成である (b), (c), (e) は RI2N と比べてほとんど変化はなかった．(e) においては僅かな性能低下が見られるが，それは 0.4 %に過ぎず，測定誤差の

範疇であると考えられる．これら 3 つのネットワークは Node-A の endpoint カウントがそれぞれ 2 リンクであり，同じである．従って，クライアント上の 2 枚の NIC の weight は同じであり，送信されるパケットは単純ラウンドロビンによって等しく割り当てられる．ここで示された結果は，対象なネットワークにおいて RI2N+が RI2N と同等の性能を持つことを意味しており，RI2N+が RI2N に対する性能的互換性を保つことを示している．

次に，非対称なネットワークについて検証する．非対称なネットワークである (d), (f), (h) において，RI2N+は RI2N に比べてそれぞれ 13.2%, 13.1%, 27.8%という性能向上を見せた．ここで注目すべきは，(h) のように非対称性が強く，トラフィックの偏りが激しいネットワークほど，RI2N+の効果が大きいという点である．

(g) は非対称なネットワークに分類されるが，RI2N でも比較的高い性能が得られている．なぜなら (g) はある意味で対称的なネットワークであり，Switch-0 と Switch-1 のトラフィックはほぼ同じだからである．実際に Node-A の Link-0 と Link-1 におけるトラフィック量は Node-B, Node-D から送信されるパケットおよび Node-C のデュアルリンクから送信されるパケットで同量となる．事実，RI2N の性能は対象な場合の (e) と近く，RI2N+は RI2N とほぼ同じ性能となっている．

3 つのクライアントを持つ非対称なネットワークである (f), (h) と，同じく 3 つのクライアントを持つ対称なネットワークである (e) の性能を比較したとき (f), (h) の (e) に対する相対性能は，それぞれ 86%, 73%となる．ネットワークが非対称であることにより，性能は低下したが，RI2N+では高い性能を得ることができた．

5.2 スループットの内訳

全体のスループットのうち，各ノードがどれぐらいのトラフィックを流しているか確認するために，(h) において Node-B, Node-C, Node-D のトラフィックを Node-A で観測する．**図 6** は，その 60 秒間の観測において，1 秒毎のスループットの変化を示した物である．ここで，Node-B, Node-C, Node-D からのトラフィックは全体のトラフィックとは独立して示してある．

RI2N に比べて RI2N+における Node-C のトラフィックは非常に増加していることが分かる．一方，Node-B と Node-D はシングルリンクしか搭載していないので，それらのノードによるトラフィックの増加は少ない．Node-C の Link-0, Link-1 に対する weight (パケットの送信割り当て比率) は，Node-A の endpoint カウンタ情報によって 1:3 となっており，全体のスループットが向上した主な要因は Switch-1 を経由して Link-1 へ転送された Node-C からのトラフィックの増加が大きかったからであることが分かる．

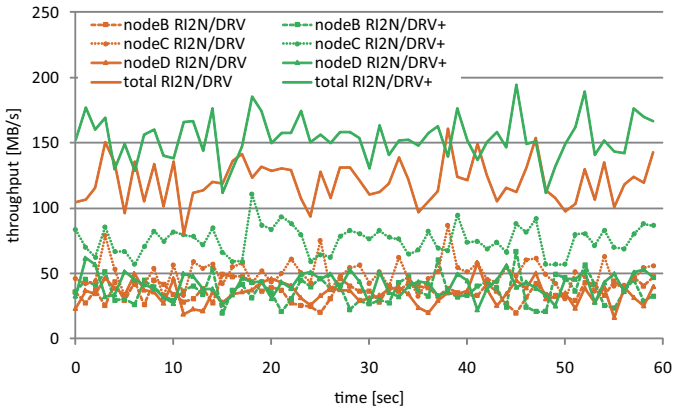


図 6 構成 (h) におけるスループットの状況

5.3 動的接続状態関知

次に RI2N+における動的接続状態関知の機能性について検証する．(c) において，これまでの通りクライアントからサーバへバースト転送を行い，Node-B の Link-1 に接続されるケーブルを抜くことで物理的な断線を発生させてみた．全体のスループットがどのように変化するか 1 秒毎に観測した結果を**図 7** に示す．実線は Node-A で観測した全体のスループットを表し，赤色と青色の一組の棒グラフは Node-C における NIC0 と NIC1，つまり Link-0 と Link-1 の weight を示す．ケーブルは t=5 [sec] のときに抜かれ，t=15 [sec] のときに再び接続された．ケーブルの切断と接続は人手によって行われたため，これらはおおよそのタイミングである．また今回の場合，ハートビートパケットの送出間隔は 2 秒で設定してあるため，再接続を検出するまで最大約 2 秒の遅延がある．

この結果から，ケーブル切断後，Node-A における Link-1 の endpoint カウントが 1 になることで，Node-C における Link-1 の送信割合が 2 倍となっていることが分かる．通信リンクの減少により，全体のスループットは低下しているが，動的接続状態関知により送信割合が適切に変化していることが確認された．リンクが故障している間，ネットワークの構成は (d) と同等となり，スループットの平均も (d) と同様になっていることが分かる．このこともまた，RI2N+の動的接続状態関知が正常に働いていることを意味する．

5.4 今後の課題

RI2N+が動的にリンクの接続変化を関知し，接続リンク数のバランスによって送信パケッ

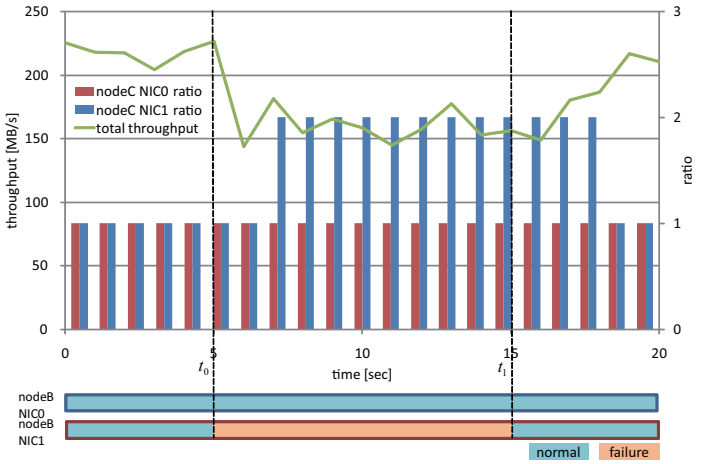


図 7 動的接続状態関知によるネットワーク構成変化の検出

トの割り当てリンク割合を変化させた結果，非対称ネットワークにおいて RI2N より高いスループットが得られていることがここまでの性能評価を通して分かった．しかし，現段階の weight 決定アルゴリズムはとても単純であり，更なる性能向上には改善が必要であると考える．

まず，現在のアルゴリズムでは経路上のスイッチを共有するノードの数により，endpoint カウントが決定され，その逆数比により weight が制御されるが，このアルゴリズムに従うと RI2N+は搭載する全ての NIC を常に使用することとなる．仮に，あるリンクがネットワーク的に「非常に弱い」リンクであったとしても，RI2N+は該当リンクを使用し続ける．例えば，(d) における Node-C の理論上のパケット送信割り当て最適解は「1:2」ではなく「0:1」である．もし，Node-C から送られる全てのパケットが Network-1 を経由するならば，Node-B は Network-0 を占有することができ，Node-A における Link-0 と Link-1 の受け取るパケット数はバースト転送時に同量となる．しかし，ネットワークの構成が更に複雑になってくると，その判断は一層難しくなるため，我々はよりエレガントで効果的なアルゴリズムを考えねばならない．

また，RI2N+の現在の実装では，各リンクの endpoint 数をカウントしているだけである．このカウントは接続の有無を検出しているに過ぎず，実際のトラフィック量を認識できていない．性能評価では片方向のバースト転送を用いたが，これは最も単純かつ一定のトラフィックを発生させるものである．しかしながら，実際のアプリケーションでは各送信ノードからの通信トラフィックは異なる．さらに言うと，1 台のノードにおいてさえ，計算中にトラフィックパターンが変化するかもしれない．従って，このような接続の有無のみの情報ではトラフィックバランスを制御する上で不十分であると考える．この課題は我々にとって最も重要な課題である．

6. おわりに

本稿では，非対称なネットワークに対してトラフィック制御を行い，ネットワーク全体のトラフィックを平均化するマルチリンク Ethernet 制御システム，RI2N+を提案した．マルチリンク Ethernet 接続に関する Linux 上のデファクトスタンダードである LCB では，非対称なネットワークをサポートしていない．しかし，大規模なクラスタ上で非対称なネットワークを構築できることは，非常に重要なことであると我々は考える．従来の RI2N でも非対称ネットワークには結果的に対応できていたが，元来このような状況を想定した実装とはなっていなかった．

RI2N+におけるパケット送信制御システムの性能として，まず非対称なネットワーク上でデータ転送スループットが向上したこと，そして RI2N に比べて性能が約 30 %向上したことが確認された．ハートビートパケットにより動的にリンクの接続数を関知できるため，もしネットワークの構成が NIC やケーブル，スイッチの故障等により変化しても RI2N+は対応することができる．この特徴はオリジナルの RI2N の耐故障機能より継承した物であり，RI2N+は性能面と機能面において RI2N と上位互換性を持つ．

今後の課題として，より効率的にトラフィックを平均化するアルゴリズムの検討が挙げられる．また，動的トラフィック量検出と全ノードの通信トラフィックが不均衡であるアプリケーションにおいて，最高の性能を出すことについても考えなければならない．

謝辞 本研究の一部は，JST-CREST 研究領域「実用化を目指した組込みシステム用ディペンダブル・オペレーティングシステム」，研究課題「省電力でディペンダブルな組込み並列システム向け計算プラットフォーム」による．

参 考 文 献

1) InfiniBand Trade Association: InfiniBand. http://www.infinibandta.org/.
2) Myricom: Myri-10G Solution. http://www.myri.com/.

Vol.2009-OS-112 No.17
2009/8/6

3) TOP 500 Supercomputing Sites: TOP500. http://www.top500.org/.
4) Davis, T.: Linux Ethernet Bonding Driver. http://sourceforge.net/projects/bonding.
5) Red Hat, Inc.: Red Hat Linux. http://www.redhat.com/.
6) Novell, Inc.: SUSE Linux. http://www.novell.com.
7) Miura, S., Hanawa, T., Yonemoto, T., Boku, T. and Sato, M.: RI2N/DRV: Multi-link Ethernet for High-Bandwidth and Fault-Tolerant Network on PC Clusters, *The Workshop on Communication Architecture for Clusters with IPDPS2009*, pp. 1–7 (2009).
8) Okamoto, T., Miura, S., Boku, T., Sato, M. and Takahashi, D.: RI2N/UDP: High bandwidth and fault-tolerant network for a PC-cluster based on multi-link Ethernet, *The Workshop on Communication Architecture for Clusters with IPDPS2007*, pp.1–8 (2007).
9) 岡本高幸，三浦信一， 朴泰祐， 塙敏博，佐藤三久：ユーザ透過に利用可能な高性能・耐故障マルチリンク Ethernet 結合システム，情報処理学会論文誌コンピューティングシステム， Vol.1, No.1, pp.12–27 (2008).
10) 三浦信一，米元大我， 塙敏博， 朴泰祐，佐藤三久：高性能・耐故障マルチリンク Ethernet 結合システムの性能評価，情報処理学会研究報告（ハイパフォーマンスコンピューティング）Vol.2009-HPC-120 No.9，情報処理学会 (2009).
11) Salim, J. H., Olsson, R. and Kuznetsov, A.: Beyond softnet, *ALS '01: Proceedings of the 5th annual Linux Showcase & Conference*, Berkeley, CA, USA, USENIX Association, pp.18–18 (2001).
12) Intel Corporation: Intel PRO Network Connections User Guides.