

フォトニックパケットスイッチにおける 2×2 バッファスイッチのスケジューリング

竹森隆介† 馬場健一†† 村田正幸††† 北山研一†

† 大阪大学大学院工学研究科
〒565-0871 大阪府吹田市山田丘 2-1
TEL/FAX : 06-6879-7728/06-6879-7688
takemori@comf5.comm.eng.osaka-u.ac.jp
kitayama@comm.eng.osaka-u.ac.jp

††† 大阪大学サイバーメディアセンター
〒567-0047 茨木市美穂ヶ丘 5-1
TEL/FAX : 06-6879-8793/06-6879-8794
baba@cmc.osaka-u.ac.jp
††† 〒560-0043 豊中市待兼山町 1-3
TEL/FAX : 06-6850-6615/06-6850-6589
murata@cmc.osaka-u.ac.jp

あらまし フォトニックパケットスイッチを実現する際に重要な問題となるのは、光領域でスイッチ内でのパケットの競合を解決することである。現時点では、光 RAM の実現が困難であるため、光ファイバによる遅延線をメモリとして利用する方法が考えられている。本稿ではまず、提案した光ラベル処理技術に基づくフォトニックパケットスイッチを対象にスイッチを構成する共有バッファ型 2×2 バッファスイッチにおける新たなパケットスケジューリングアルゴリズムを提案した。次に 2×2 バッファスイッチおよびそれを組み合わせた多段スイッチ構成に対し新しいアルゴリズムを適用してシミュレーションにより性能を測った。また実現コストを考慮した評価も併せて行った。その結果、2×2 バッファスイッチ、多段スイッチにおいて新たなスケジューリングアルゴリズムが有効であり、また、トラヒックパターンによっては多段スイッチにおけるディフレーションが有効であることがわかった。

キーワード 光ラベル処理、フォトニックパケットスイッチ、2×2 バッファスイッチ、パケットスケジューリングアルゴリズム

A Scheduling Algorithm for 2×2 Buffered Switch in a Photonic Packet Switch

Ryusuke Takemori†, Ken-ichi Baba††, Masayuki Murata†††, Ken-ichi Kitayama†

† Graduate School of Engineering,
Osaka University
2-1 Yamada-Oka, Suita, Osaka 565-0871
TEL/FAX : 06-6879-7728/06-6879-7688
takemori@comf5.comm.eng.osaka-u.ac.jp
kitayama@comm.eng.osaka-u.ac.jp

††† Cybermedia Center, Osaka University
〒5-1 Mihogaoka, Ibaraki, Osaka 567-0047
TEL/FAX : 06-6879-8793/06-6879-8794
baba@cmc.osaka-u.ac.jp
††† 1-3 Machikaneyama, Toyonaka, Osaka 560-0043
TEL/FAX : 06-6850-6615/06-6850-6589
murata@cmc.osaka-u.ac.jp

Abstract For implementing a photonic packet switch, contention resolution greatly affects network its performance. In a photonic switch with packet buffers, we have to adopt a different approach from an electronic switch because there is no ready-to-use optical memory. That is, we use the delay lines consisting of fixed-length fibers. In this paper, at first, we propose a new packet scheduling algorithm for the 2×2 buffered switch that constructs the proposed packet switch architecture as a basic switching element using optical buffers. Next, we evaluate the performance of the 2×2 buffered switch and the multi-stage switch by applying our new algorithm through simulation experiments. We also consider the cost for implementing a photonic packet switch. We show that our new algorithm can make the performance of a photonic packet switch better and deflection in multi-stage switch is effective.

Key Words Photonic Label Processing, Photonic Packet Switch, 2×2 Buffered Switch, Packet Scheduling Algorithm

1 はじめに

近年における情報のマルチメディア化及びインターネットユーザの指数的增长により、将来のバックボーンネットワークの速度は Tbps 級のものが必要とされており、早急なスイッチング処理機能の光化が必要である。フォトニックネットワークとは、多重分離機能、スイッチング機能、ルーティング機能において超高速処理にフォトニック技術を導入したネットワークである。それによってエレクトロニクスデバイスの速度限界から生じるノードのボトルネック解消が可能となる。

これまで光伝送における多重方式として、時分割多重方式 (TDM; Time Division Multiplexing) と波長分割多重方式 (WDM; Wavelength Division Multiplexing) が主として研究されてきた。一方、無線分野で実用化されている符号分割多重アクセス方式 (CDMA; Code Division Multiple Access) をベースとする光多重化方式 (OCDM) は、TDM や WDM にはないユニークな特徴を有しているため注目を集めている。文献 [1] では、光符号相関技術に基づく光符号器とマッチドフィルタリングによるフォトニックラベル処理が提案されており、ラベル処理によるスイッチングの有効性が明らかになっている。また、OCDM 符号化器及び復号化器は光受動デバイスで実現できることも特徴の一つである。

そこで本稿では、フォトニックラベル処理に基づいたパケットスイッチを対象とする。フォトニックパケットスイッチの構成図を図 1 に示す [1]。図 1 において、入力パケットは光領域でラベル情報を読み取られ、その情報をもとにした制御信号によってスイッチが駆動しスイッチングされる。スイッチングされたパケットは光領域でラベル情報を書き換えられ出力される。

フォトニックパケットスイッチにおいて生じる問題のひとつとしてパケットの競合がある。ルータ内などのスイッチでは同じ出線を目指すパケット同士の競合が起こる。従来の電気処理では、RAM (Random Access Memory) をバッファとして利用した蓄積交換 (store-and-forward) 技術によりパケットの競合を容易に解決できる。つまり競合が起こる可能性のある場合、到着したパケットをバッファに入れ、出線が空いたら 1 つずつ出力すればよい。

光領域でパケットの競合を解決するには光バッファを設ける (optical buffering)、迂回させる (deflection routing)、波長を変換する (wavelength conversion) などのいくつかの手法がある。バッファのデバイスについては、光 RAM がまだ実用化されていないため全く違ったアプローチが必要となる [2]。光バッファを設ける手法の一つに、固定長の光ファイバ遅延線をメモリとして用いる方法がある。フォトニックパケットスイッチにおいて出線が空くまで待たされるべきパケットは、光ファイバ遅延線に送られ、伝搬遅延時間を経て取り出される。遅延線を用いたバッファにおけるパケット競合解決に関する研究がさかんに行われている [3]—[9]。また、そのような遅延線バッファを用いたスイッチの構成方法として CORD (Contention Resolution by Delay lines)[3]、SLOB (Switch with Large Optical Buffer)[4] などがある。しかし、いずれも実現するには複雑な制御を必要とする。さらに実際に大規模ネットワークに適用しての性能評価は行われていない。

本稿では文献 [10] で提案したフォトニックラベル処理に

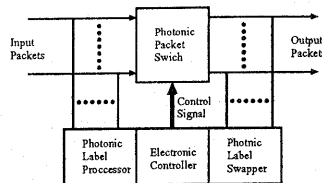


図 1: フォトニックパケットスイッチの構成図

基づいたパケットスイッチ構成を対象とする。スイッチ構成として、簡単に構成できる光ファイバ遅延線を用いた 2×2 バッファスイッチ及びそれらを組み合わせたセルフルーティング可能な多段スイッチを考える。また、基本的な性能を測るため、固定長パケットを取り扱うものとする。 2×2 バッファスイッチ及びそのパケットスケジューリングアルゴリズムはすでに提案されている [6] が、パケットが到着しない場合にも実パケットと同様にバッファに空パケットを挿入して制御するため、バッファの効率的な利用ができないという問題点がある。また、文献 [6] に示された方式を用いて多段スイッチ構成とした場合の解析及び性能評価は行われていない。

そこで、我々は文献 [10] でバッファの効率的な利用が可能なスケジューリングアルゴリズムを提案し、有効性を示した。しかしながら、バッファの状態数が多くなり、大規模なバッファ構成を用いた場合の制御が困難になるという問題点がある。本稿では、このアルゴリズムをさらに改良し、少ない状態数でバッファの有効利用を図ることのできる手法を提案する。

最後に本稿の構成を述べる。2 章では、本稿で対象とする 2×2 バッファスイッチ及びそれらを組み合わせた多段スイッチの構成について説明をする。3 章では、 2×2 バッファスイッチにおける新しいパケットスケジューリングアルゴリズムを提案する。4 章では、シミュレーションにより 2×2 バッファスイッチ及びそれらを組み合わせた多段スイッチの性能を評価し、パケット棄却率、平均待ち時間を性能指標に新しいアルゴリズムの有効性を示す。さらに、実現コストを考慮した検討も併せて行う。最後に第 5 章で本稿のまとめと今後の課題を述べる。

2 対象とするフォトニックパケットスイッチ構成

本稿で対象とするパケットスイッチ [10] は、図 1 に示すように、付加されたラベルに従ってスイッチングを行う。次節以降で説明する、 2×2 バッファスイッチおよび多段スイッチにおいても、宛先などのそれぞれのラベル処理を行い、スイッチングを決定する。すなわち、 2×2 バッファスイッチにおいては、バッファ制御のためのラベル処理を行い、多段スイッチ構成においては、スイッチ要素 (2×2 バッファスイッチなど) のラベル処理とは別に、スイッチ内ルーティングのラベル処理を行うものとする。

また、パケットスイッチは 2×2 スwitch素子と遅延線で構成される。 2×2 バッファスイッチでは、スイッチ素子と遅延線を次節で示すように構成する。多段スイッチでは、

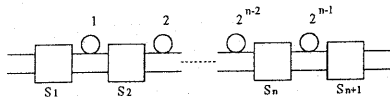


図 2: 2×2 バッファスイッチ

2×2 バッファスイッチをひとつのスイッチ要素と考え、それらを多段に組み合わせて構成する。すなわち、多段スイッチにおいてもスイッチ素子と遅延線で構成されている。実現コストを考えるとスイッチ素子の占める割合が大きく、スイッチ素子をどう配置し、スイッチをどう構成するかによって性能が異なると考えられる。多段スイッチでは性能向上を図るため前段にスイッチ素子等を加えルートを増やす方法と、各スイッチ要素にスイッチ素子を加え、バッファを増やす方法などが考えられる。次章以降でこれらの性能を明らかにしていく。

2.1 2×2 バッファスイッチ

2 入力 2 出力スイッチ素子を n ($n \geq 2$) 段接続し、 i 段目のスイッチ素子 S_i と $i+1$ 段目のスイッチ素子 S_{i+1} の間に 2^{i-1} タイムスロット分の長さを持つ遅延線を利用した光バッファを設け構成したものを 2×2 バッファスイッチと考える [3]。これを図 2 に示す。このように構成することで、キュー長 T ($T \leq 2^{n-1} - 1$) のときに到着したパケットは、 T の二進表現が $a_{n-1}a_{n-2} \dots a_2a_1$ ($a_i = 0$ または 1 ; $i = 1, 2, \dots, n-1$) とすると S_i において、 $a_i = 0$ の場合下側出力方向へスイッチングし $a_i = 1$ の場合上側の遅延線にスイッチングされる。即ち到着パケットに対しスイッチ入力時において待たせたいタイムスロット数の情報をラベルとして与えることでハードウェアによりパケットを T タイムスロット待たせることが簡単に実現できる。例えば到着パケットを 6 タイムスロット待たせたい場合、スイッチ入力時パケットには 6 (0...0110) という情報ラベルを与える。さらに、増やしたいバッファ数に対し、必要となるハードウェア数が対数的な数となり、経済的にも拡張性に富んでいる。本稿では、2×2 バッファスイッチを共有バッファ型スイッチとして用いる [6]。パケットの出力先に関して、上側の出線を目指すパケットを U 、下側の出線を目指すパケットを L と表す。到着したパケットは U 、 L どちらも同一のバッファに挿入される。スイッチ素子数 M に対して、バッファ数 l は

$$l = 2^{M-1} - 1 \quad (1)$$

で表される。

2.2 多段スイッチの構成

超高速フォトニックネットワークで高度かつ多様なサービスを実現するためには、交換系における複雑な制御処理の不要なセルフルーティング方式が適している [11]。本稿では多段スイッチ構成としてセルフルーティング可能でしかも優れたスケラビリティを持つバンヤンスイッチの一つであるベースライン (Baseline) 型 $2^N \times 2^N$ スwitch構成を対象とする [7][10][12]。16×16 スwitch構成におけるセルフルーティングの例を図 3 に示す。

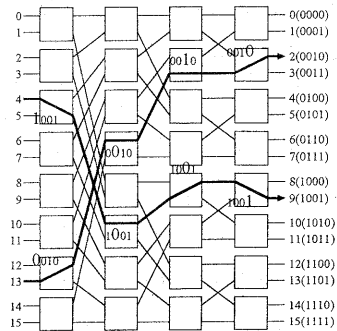


図 3: 16×16 スwitch構成

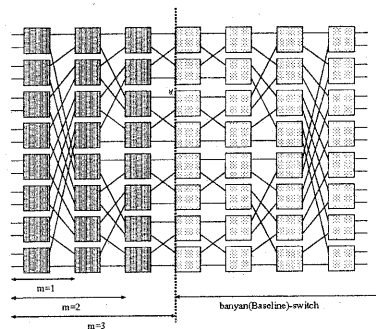


図 4: ディフレクションを考慮した 16×16 スwitch

バンヤンネットワークはセルフルーティング可能なスイッチではあるが内部リンク数が少ないため、バッファを設けていてもパケットの棄却が起こりやすいというデメリットがある。そこで、バンヤンネットワークの前段にソーティングネットワークを設けることにより中間リンクの競合を減少させることができる。ソーティングネットワーク内でのパケットの競合については目的ポートとは異なるポートにスイッチングするディフレクションを行うことによって棄却を回避することが可能である。図 4 にディフレクションを考慮したスイッチ構成の例を示す。

また、フォトニックパケットスイッチを構成するデバイスのうち、コスト面で支配的となるのはスイッチ素子である。スイッチ素子を (1) 2×2 バッファスイッチ内のバッファ部分に利用する方法と、(2) ソーティングネットワークを接続し、ディフレクションルーティングのために用いる方法とではどちらの方法がよりパケットの競合を回避できるか考えるのは現実的に重要である。実際に $2^N \times 2^N$ フォトニックパケットスイッチを構成すると、(1) に必要なスイッチ素子数 $M_1 = 2^{N-1}$ (縦のスイッチ段数) $\times N$ (横のスイッチ段数) $\times M'_1$ (2×2 バッファスイッチ内に実装するスイッチ素子数) となり、(2) で最大数のソーティングネットワークを接続すると $M_2 = 2^{N-1}$ (縦のスイッチ段数) $\times (2N-1)$ (横のスイッチ段数) $\times M'_2$ (2×2 バッファスイッチ内に実装するスイッチ素子数) となる。

3 パケットスケジューリング アルゴリズム

3.1 従来のアルゴリズム

文献 [6] では、図 2 に示す 2×2 バッファスイッチを共有バッファ型スイッチとして利用した時のパケットスケジューリングアルゴリズムが提案されている。このアルゴリズムではパケットが到着しなかった場合にもパケットが疑似的に到着したものとして扱う。すなわち、空パケットが到着したものとして U 、 L と同様にラベルを与えてバッファに挿入し、常に U 、 L のキュー長の合計がバッファ数と等しくなるように制御する。しかし実際にパケットが到着しないにも関わらず空パケットがバッファに挿入されるものとして取り扱うためバッファの有効利用が図られていない。例えば、バッファにパケットがない場合でも空パケットがバッファにあると考え、到着したパケットはすぐに出力されずに必ず空パケットの後ろに挿入される。当然のことながらスループットやパケット棄却率は劣化する。

3.2 提案するアルゴリズム

そこでこのような問題を解決するため、新しいアルゴリズムでは到着パケットがない場合に空パケットを挿入するのではなく、そのまま空きスペースとして扱う [10]。あるバッファの状態において、現在のバッファの状態と入力パターンから可能な全てのラベルの与え方を考え、その中から適切なものを決定する。バッファ数 l の場合、バッファの状態数は、1 つの遅延線に対して、「パケットなし」、「 U が入っている」、「 L が入っている」の 3 状態が考えられるので 3^l となる。しかし適切なラベルを与えることにより取りえないバッファの状態を取り除くことができ、状態数を $(l+1)l/2$ にまで減らすことができる。バッファの状態を状態変数とし、入力からその状態が遷移する状態遷移図を作成し、これを基にスケジューリングを行うことによって、より効率的なバッファの利用が可能となり、従来のアルゴリズムに比べその性能が改善される。しかし、バッファ数が増えたとその状態数が $O(n^2)$ で増加するためにバッファ数を増やした場合に適応することが困難となる [10]。

そのため、本稿ではバッファ数を増やした場合に適応するためにバッファ内の空きスペースを限定して状態数を減らす。もし、限定した空きスペースが限定数以上となる場合には、従来方式と同様に空パケットを挿入するという手法を取っている。ここでは、扱う空きスペースの数を $depth$ と定義する。バッファの状態は d ($d = 0, 1, \dots, depth$) と s ($s = 0, 1, \dots, l$) により (d, s) で表される。 d はバッファ内の空きスペースの数、 s はバッファ内の L の個数に相当する。これよりバッファに入っている U の個数は $l - s - d$ と表される。バッファ内の L の個数が同じでも空きスペースを扱うことで状態が異なる。バッファ内の L の個数とどの位置が空きスペースになっているかで状態は決定できる。バッファ数 7 の場合の例を図 5 に示す。

2 本の入力線から到着するパケットが、「 U と L 」、「どちらも U 」、「どちらも L 」、「 U 1つのみ」、「 L 1つのみ」、「どちらからも入力なし」をそれぞれ「 UL 」、「 $2U$ 」、「 $2L$ 」、「 U 」、「 L 」、「 N 」と表す。 U と L は対称であるため、状態数の半分の状態および状態遷移を決定することにより

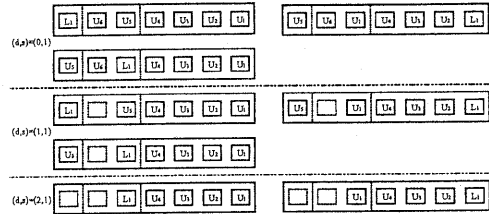


図 5: 空きスペースを扱った場合のバッファ状態の例 (バッファ数 7)

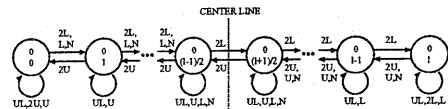


図 6: 従来の状態遷移図 ($depth = 0$)

全体の状態および状態遷移が確定する。 $(d, s) = (i$ ($i = 0, 1, \dots, depth$), j ($j = 0, 1, \dots, l$)) の時のバッファ内の U を L に、 L を U に置き換えると状態は状態遷移図の中心線から線対称に移動し $(d, s) = (i, l - j - i)$ となる。 $(d, s) = (i, j)$ の時に到着する U 、 L はそれぞれ $(d, s) = (i, l - j - i)$ の時に到着する L 、 U と同様に扱われることになる。すなわち、入力「 $2U$ 」、「 $2L$ 」、「 U 」、「 L 」をそれぞれ「 $2L$ 」、「 $2U$ 」、「 L 」、「 U 」に置き換えることで全体の状態遷移図が完成する。

従来のアルゴリズムを利用した場合のバッファの状態遷移図を図 6 に示す [6]。従来方式では空きスペースが 1 つもないため、 $depth = 0$ とする。状態遷移図の各状態の上段の数字が d を、下段の数字が s を表している。バッファが状態 $(d, s) = (0, j)$ の時に到着した L には j というラベルを与え、 U には $l - j$ というラベルを与える。 U と L が同時に到着した場合でもそれぞれに対して 2.1 章に示したように始めに与えられたラベルに従ったスイッチングを行えばよい。このようにスイッチングした場合にも途中の素子間でパケットの競合は起こらない。状態 0 または l の時に到着パケットが「 $2U$ 」または「 $2L$ 」の場合、 U 、 L がそれぞれ 1 つずつ棄却されることになる。つまり状態遷移図の中心の状態、ここでは $(l-1)/2$ または $(l+1)/2$ に近づくようにバッファを制御することにより棄却率を下げるができる。新しいアルゴリズムではその点を考慮して、スケジューリングを行う。状態数の半分の状態および状態遷移が確定すれば全体の状態および状態遷移が確定することを利用してこれ以降ではバッファ内の L の個数よりも U の個数の方が多い、または L と U の個数が等しいものとして説明する。

まず、状態 $(d, s) = (0, j)$ の時に到着したパケットには空きスペースがない従来方式と同様にラベルを与えれば良い。すなわち到着した L には j というラベルを与え、 U には $l - j$ というラベルを与える。状態 $(d, s) = (0, j)$ からの状態遷移は、入力により次のようになる。

「 UL 」の場合 $(d, s) = (0, j)$

「 $2U$ 」の場合 $(d, s) = (0, \max(j-1, 0))$

「2L」の場合 $(d, s) = (0, j+1)$

「U」の場合 $(d, s) = (\min(j, 1), \max(j-1, 0))$

「L」の場合 $(d, s) = (1, j+1)$

「N」の場合 $(d, s) = \begin{cases} (\min(j+1, 1), j); \text{depth} = 1 \\ (\min(j+1, 2), \max(j-1, 0)); \text{depth} = 2 \end{cases}$

状態 $(d, s) = (i, j)$ の時に到着したパケットに与えるラベルについて考える。Uが到着した場合、バッファ内のUの個数の方がLの個数よりも多いことから、そのUには待ちスロット数が最小となる $l-j-i$ というラベルを与える。Lに与えるラベルは入力および今の状態から決定する。入力が「UL」の場合、 $i=1$ の時、Lにも待ちスロット数が最小となる j というラベルを与えると、到着したU、Lそれぞれに与えられたラベルの最下位ビットが同じになり1段目のスイッチで衝突が起きてしまう。そのためLには $j+1$ というラベルを与える。 $i=1$ かつ $j = (l-1)/2$ の時はバッファ内のUの個数とLの個数が等しいためどちらか一方ランダムに待ちスロット数が最小となるラベルを与える。 $i=2$ の時、Lにも待ちスロット数が最小となる j というラベルを与える。到着したU、Lそれぞれに与えられたラベルの最下位ビットは異なり、衝突は起こらない。入力が「2L」の場合、 $i=1$ かつ j が偶数の時、それぞれ待ちスロット数が最小となる $j, j+1$ というラベルを与える。 $i=1$ かつ j が奇数の時、待ちスロット数が最小となるラベルを与えると状態が偏り、従来方式よりもパケットの棄却が起りやすくなるため、それぞれ $j+1, j+2$ というラベルを与える。 $j = (l-1)/2 - 1$ の時はそれぞれ $j, j+1$ というラベルを与える。 $i=2$ の時も、待ちスロット数が最小となるラベルを与えると従来方式よりもパケットの棄却が起りやすくなるため、 j が偶数の時、 $j+2, j+3$ というラベルを与える。 $i=2$ かつ $j = (l-1)/2 - 1$ の時は $j, j+1$ というラベルを与える。 j が奇数の時、 $j+1, j+2$ というラベルを与える。入力が「L」の場合、 j が偶数の時、待ちスロット数が最小となる j というラベルを与え、 j が奇数の時は $j+1$ というラベルを与える。状態 $(d, s) = (1, j)$ からの状態遷移は、入力により次のようになる。

$$\begin{aligned} \text{「UL」の場合 } (d, s) &= \begin{cases} (0, j+1); j \neq (l-1)/2 \\ (0, (l-1)/2); j = (l-1)/2 \\ (0, (l+1)/2); j = (l-1)/2 \end{cases} \\ \text{「2U」の場合 } (d, s) &= \begin{cases} (\min(j, 1), \max(j-1, 0)); j \text{ が偶数} \\ (0, j); j \text{ が奇数} \end{cases} \\ \text{「2L」の場合 } (d, s) &= \begin{cases} (1, j+1); j \text{ が偶数} \\ (0, j+2); j \text{ が奇数かつ } j \neq (l-1)/2 \\ (0, j+1); j = (l-1)/2 \end{cases} \\ \text{「U」の場合 } (d, s) &= \begin{cases} (1, j); \text{depth} = 1 \\ (\min(j+1, 2), \max(j-1, 0)); \text{depth} = 2 \text{ かつ } j \text{ が偶数} \\ (1, j); \text{depth} = 2 \text{ かつ } j \text{ が奇数} \end{cases} \\ \text{「L」の場合 } (d, s) &= \begin{cases} (1, \min(j+1, (l-1)/2)); \text{depth} = 1 \\ (2, j); \text{depth} = 2 \text{ かつ } j \text{ が偶数} \\ (1, \min(j+1, (l-1)/2)); \text{depth} = 2 \text{ かつ } j \text{ が奇数} \end{cases} \\ \text{「N」の場合 } (d, s) &= \begin{cases} (1, \min(j+1, (l-1)/2)); \text{depth} = 1 \\ (2, j); \text{depth} = 2 \text{ かつ } j \neq (l-1)/2 \\ (2, j-1 \text{ または } j); \text{depth} = 2 \text{ かつ } j = (l-1)/2 \end{cases} \end{aligned}$$

状態 $(d, s) = (2, j)$ からの状態遷移は、入力により次のようになる。

$$\text{「UL」の場合 } (d, s) = \begin{cases} (1, j+1); j \text{ が偶数} \\ (0, j+2); j \text{ が奇数} \end{cases}$$

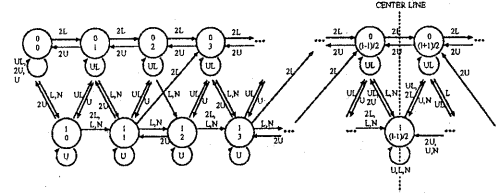


図 7: depth = 1 の状態遷移図

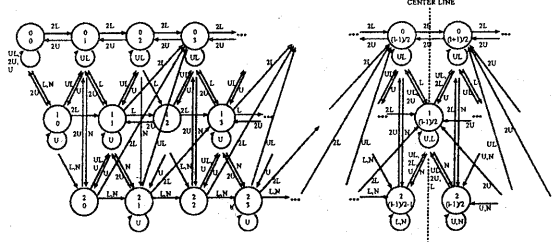


図 8: depth = 2 の状態遷移図

$$\begin{aligned} \text{「2U」の場合 } (d, s) &= \begin{cases} (0, j+1); j \text{ が偶数} \\ (1, j); j \text{ が奇数} \end{cases} \\ \text{「2L」の場合 } (d, s) &= \begin{cases} (0, j+3); j \text{ が偶数かつ } j \neq (l-1)/2 - 1 \\ (1, j+1); j = (l-1)/2 - 1 \\ (1, j+2); j \text{ が奇数} \end{cases} \\ \text{「U」の場合 } (d, s) &= \begin{cases} (1, j+1); j \text{ が偶数} \\ (2, j); j \text{ が奇数} \end{cases} \\ \text{「L」または「N」の場合 } (d, s) &= (2, \min(j+1, (l-2)/2 - 1)) \end{aligned}$$

例として $\text{depth} = 1$ の状態遷移図を図 7 に、 $\text{depth} = 2$ の状態遷移図を図 8 に示す。これらの状態遷移に従ったバッファスケジューリングを行うスイッチの評価を次章で行う。

4 性能評価

4.1 シミュレーションモデル

計算機シミュレーションに際して、フォトニックパケットスイッチのモデル化を以下に行う。まずパケットの到着過程はポアソン過程とする。スイッチングのための処理時間、およびパケットの伝搬遅延時間は 0 とする。また、本稿で用いる性能指標として、パケット棄却率、平均待ち時間を用いる。パケット棄却率は、生じたパケットに対して、パケットがバッファ溢れにより棄却され、消失する確率と定義する。平均待ち時間は、入力ポートから目的の出力ポートまでルーティングされるまでにかかった平均タイムスロット数と定義する。負荷は入線 1 本あたりのパケット生起率と定義する。4.2 節では、パケット生起率はどの入線も同じとし、生起パケットの目指す出線はランダムとする。4.3 節では、コストを考慮した性能評価において

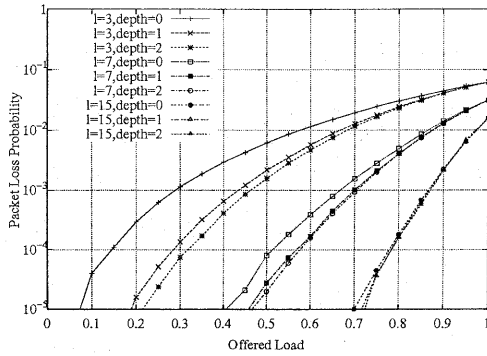


図 9: 2×2 バッファスイッチにおけるパケット棄却率

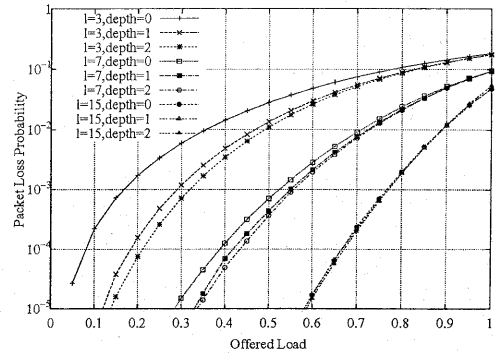


図 11: 16×16 スイッチにおけるパケット棄却率

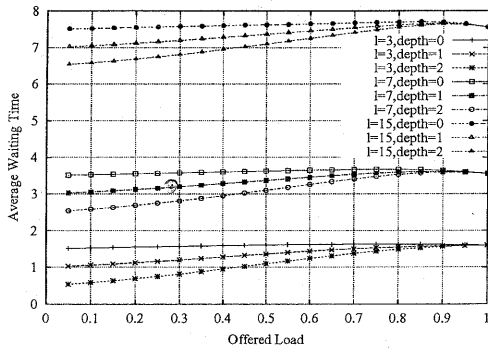


図 10: 2×2 バッファスイッチにおける平均待ち時間

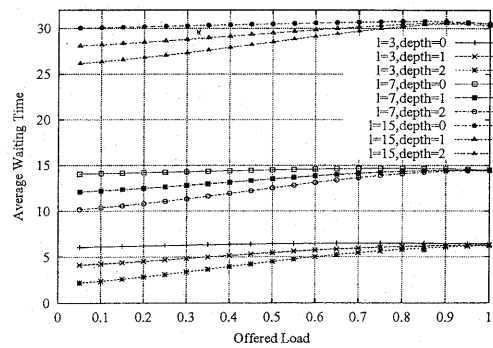


図 12: 16×16 スイッチにおける平均待ち時間

不均一なトラフィックを扱う。

4.2 2×2 バッファスイッチの性能評価

本節では、2.1節で述べた2×2バッファスイッチに従来のパケットスケジューリングアルゴリズム ($depth = 0$) と新しいアルゴリズム ($depth = 1, 2$) を適用した場合の性能をシミュレーションにより評価する。バッファ数 $l = 3, 7, 15$ の場合のパケット棄却率と平均待ち時間をそれぞれ図 9、図 10に示す。 $depth$ が大きくなると負荷の低い場合の棄却率、平均待ち時間の特性が改善されていることが分かる。バッファ数が多くなるにつれて $depth$ を増やす効果が小さくなっていくが、これはバッファ数に対し、 $depth$ の値が相対的に小さくなる、すなわち、すべての空きスペースを考えた場合の状態数に比べ $depth$ を限定した状態数の割合が小さくなるため、空きスペースを有効に活用できないことが要因である。

4.3 多段スイッチにおける性能評価

本節では、2.2節で述べた多段スイッチ (16×16 スイッチ) の性能をシミュレーションにより評価する。まず、2×2 バッファスイッチに従来のパケットスケジューリングアルゴリズム ($depth = 0$) と新しいアルゴリズム ($depth = 1, 2$) を採用した場合のパケット棄却率、平均待ち時間をそれぞれ図 11、図 12に示す。多段スイッチにおいても従来方式よりも提案方式が良好な特性を示していることが分かる。パケット棄却率、平均待ち時間ともに2×2 バッファスイッチと同様の傾向を示すことが分かる。

次に、16×16 スイッチ構成において、実現コストを考慮するためスイッチ全体のスイッチ素子数を同数にした場合にスイッチ素子をバッファ部分に利用する方法及びデフレーションのために用いる方法での性能の比較を行う。ここでは $M_1' = 7$ 、 $M_2' = 4$ とし ($M = 224$)、式 (1) より2×2 バッファスイッチ内に実装できるバッファ数はそれぞれ $l_1 = 2^{7-1} - 1 = 63$ 、 $l_2 = 2^{4-1} - 1 = 7$ となる。従来のパケットスケジューリングアルゴリズム ($depth = 0$) と新しいアルゴリズム ($depth = 2$) を採用した場合のパケッ

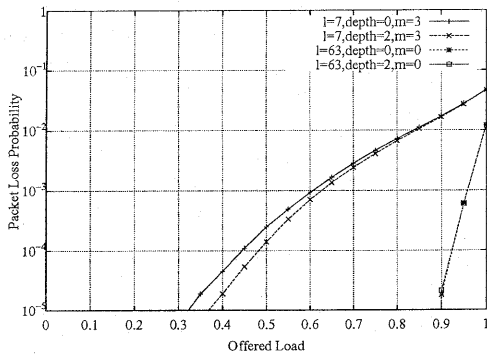


図 13: スイッチ素子数を同数とした場合のケット棄却率

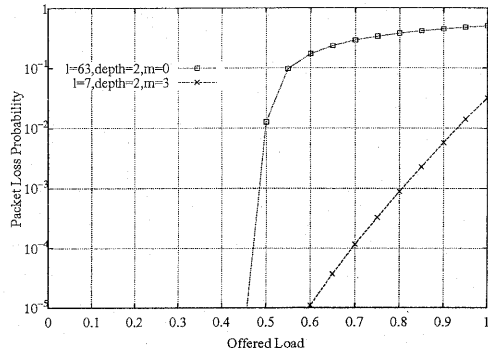


図 15: 不均一なトラヒックにおけるケット棄却率

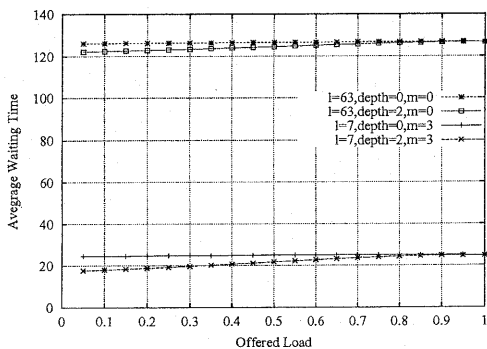


図 14: スイッチ素子数を同数とした場合の平均待ち時間

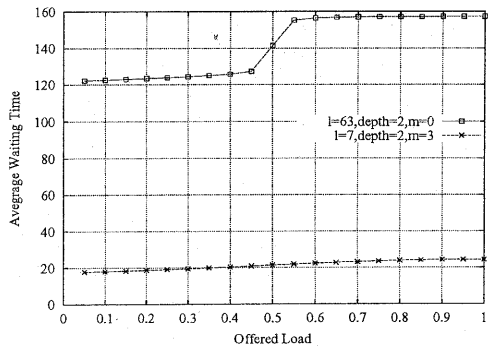


図 16: 不均一なトラヒックにおける平均待ち時間

ト棄却率及び平均待ち時間をそれぞれ図 13、図 14 に示す。ケット棄却率及び平均待ち時間の特性ともに大きな差が生じていることが分かる。バッファ数が多い方が棄却率は減少するが、スイッチが負荷の低い状態で動作する時には待ち時間が非常に大きくなる。均一なトラヒックにおいてはケット棄却率を下げるにはバッファ数を増やす方が効果があるが、負荷の低い状態では、ディフレクションを用いることが有効である。新しいアルゴリズムを用いることで負荷の低い状態における特性に多少の改善が見られる。

次に、 16×16 スイッチに不均一なトラヒックを与えた場合の性能評価を行う。不均一なトラヒックとして入力ポート 0~7 からのみ到着パケットがあり、それらは全て出力ポート 8~15 を目指す (図 3 参照) ようなトラヒックを想定して評価を行う。ここでも実現コストを考慮に入れ、スイッチ全体のスイッチ素子数を同数にする。スケジューリングは新しいアルゴリズム ($depth = 2$) を採用する。ケット棄却率及び平均待ち時間の特性をそれぞれ図 15、図 16 に示す。ディフレクションを行った場合のケット棄却率及び平均待ち時間の特性が大幅に良くなることが分かる。

$l = 63, depth = 2, m = 0$ の平均待ち時間が階段状になっている。これは、1 段目のスイッチにおいて負荷 0.5 付近でパケットの棄却率が急激に上がり、パケットの待ち時間もその付近で急激に上がるためである。このような不均一なトラヒックにおいては、スイッチ内の一部分にかかる負荷をディフレクションを用いることでスイッチ全体に分散でき、スイッチ全体のバッファを効率よく利用できているためと考えられる。

5 まとめと今後の課題

本稿では、フォトニクスラベル処理に基づいたパケットスイッチ構成を提案した。まず、光ファイバを用いた共有バッファ型 2×2 バッファスイッチにおける新しいスケジューリングアルゴリズムを提案し従来のスケジューリングアルゴリズムとの性能の比較を計算機シミュレーションにより行った。その結果、提案方式はより効率的なバッファ利用が図ることができ、ケット棄却率、平均待ち時間どちらの特性も従来方式より改善できることを示した。

また、 2×2 バッファスイッチを組み合わせて構成した多段スイッチにおける提案方式と従来方式の性能の比較も行い、 2×2 バッファスイッチの場合と同様の改善が見られることがわかった。

さらに、多段スイッチにおいてはパケット競合の解決法としてディフレクションを用いることも考えられ、バッファによる解決法とディフレクションを用いる解決法との性能の比較を実現コストを考慮して行なった。その結果、均一なトラフィックにおいてはバッファによる解決法が良く、不均一なトラフィックにおいてはディフレクションを用いる解決法が良いことがわかった。

最後に今後の課題としては、本稿では固定長パケット、同期スイッチング、を仮定したネットワークを対象に性能評価を行った。しかし、今後のインターネットの発展を考える上では可変長パケット、非同期スイッチングを対象とするのが現実的である。そのようなネットワークに提案方式を適用し、その有効性を明らかにする必要がある。

参考文献

- [1] K. Kitayama, N. Wada, and H. Sotobayashi, "Architectural considerations for photonic IP router based upon optical code correlation (Invited)," *Journal of Lightwave Technology*, Vol. 18, No. 12, December 2000.
- [2] S. Yao and B. Mukherjee, "Advanced in Photonic Packet Switching: An Overview," *IEEE Communications Magazine*, pp. 84-94, February 2000.
- [3] I. Chlantan, A. Fumagalli, L. G. Kazovsky, P. Melman, W. H. Nelson, P. Poggiolini, M. Cerisola, A. Choudhury, T. K. Fong, R. T. Hofmeister, C. L. Lu, A. Mekittikul, D. J. Sabido, C. J. Suh and E. W. M. Wong, "CORD:Contention Resolution by Delay Lines," *IEEE Journal on Selected Areas in Communications*, Vol. 14, No. 3, pp. 1014-1029, June 1996.
- [4] D. Hunter, W. D. Cornwell, and T. H. Gilfedder, "SLOB: A Switch with Large Optical Buffers for Packet Switching," *IEEE Journal of Lightwave Technology*, Vol. 16, No. 10, pp. 1725-1736, October 1998.
- [5] D. Hunter, M. C. Chia, and I. Andonovic, "Buffering in Optical Packet Switches," *IEEE Journal of Lightwave Technology*, Vol. 16, No. 12, December 1998.
- [6] D. Hunter, D. Cotter, R. B. Alunad, W. D. Cornwell, T. H. Gilfedder, P. J. Legg, and I. Andonovic, " 2×2 Buffered Switch Fabrics for Traffic Routing, Merging, and Shaping in Photonic Cell Networks," *IEEE Journal of Lightwave Technology*, Vol. 15, No. 1, pp. 86-100, January 1997.
- [7] C-S. Wu, G-K. Ma, and B-S. P. Lin, "Extended Baseline Architecture for Non-blocking Photonic Switching," *IEEE Journal of Lightwave Technology*, Vol. 15, No. 5, pp. 771-778, May 1997.
- [8] R. Kannan, D. Lee, K. Y. Lee, and H. F. Jordan, "Optical TDM Sorting Networks for High-Speed Switching," *IEEE Transactions on Communications*, Vol. 45, No. 6, pp. 723-736, June 1997.
- [9] R. L. Cruz and J-T. Tsai, "COD: Alternative Architecture for High Speed Packet Switching," *IEEE/ACM Transactions on Communications*, Vol. 4, No. 1, pp. 11-21, February 1996.
- [10] 竹森隆介, 馬場健一, 村田正幸, 北山研一, "フォトニックパケットスイッチに最適なバッファ構成手法," 電子情報通信学会フォトニックネットワークをベースとする次世代インターネット技術 (PNI) 研究会, July 2001.
- [11] S. Nishi, I. Saeki, H. Tode and K. Murakami, "A Photonic IP Switching Technique Using Code Division Multiplexing," *IEICE Transactions on Communications*, Vol. E83-B, No. 10, pp. 2321-2330, Oct. 2000.
- [12] A. Pattavina, "Switching Theory," JOHN WILEY & SONS, 1997.
- [13] K. L. Hall and K. A. Rauschenbach, "All-Optical Buffering, of 40-Gb/s Data Packets," *IEEE Photonics Technology Letter*, Vol. 10, No. 3, pp. 442-444, March 1998.