

部分ウェイト方式による WDM 波長予約プロトコル

東京電機大学 理工学部 情報システム工学科

高須 晴久 桧垣 博章

E-mail: {haru,hig}@higlab.k.dendai.ac.jp

全光ネットワークにおける広帯域通信環境構築技術として WDM(Wavelength Division Multiplexing: 波長多重化) が用いられている。送信元コンピュータから送信先コンピュータまで単一の波長を動的に割り当てるパス多重化は、光スイッチでの波長変換を必要としない点で優れているが、ひとつの光ファイバリンクで複数の通信に同一の波長を割り当てることができないという制約を守る波長割り当て手法が必要となる。特に、大規模全光ネットワークを実現するためには、波長割り当て状況を分散的に管理する手法を用いることが求められる。本論文では、波長割り当て時に、経路上のあるスイッチで割り当て可能な波長が存在しない場合に対して、従来のドロップ方式、ホールディング方式とは異なる新たな方式として部分ウェイト方式を提案する。この方式は、従来の2つの方式よりも波長割り当てに要する時間を短縮し、全光ネットワークのスループットを向上させることができる。最後に、本方式を用いたフォワードリザーベーションプロトコルを設計する。

Wavelength Reservation with Partial Waiting for Release in WDM Network

Haruhisa Takasu Hiroaki Higaki

Department of Computers and Systems Engineering

Tokyo Denki University

E-mail: {haru,hig}@higlab.k.dendai.ac.jp

For achieving high-bandwidth all-optical computer networks, WDM(Wavelength Division Multiplexing) has been developed. In path-multiplexing where same wavelength is reserved in every optical fiber link from a source computer to a destination one, it is required not to assign same wavelength to multiple communications in fiber link. In order to realize a large scale all-optical network, distributed control and management in wavelength reservation are required for reducing communication overhead. (conventionally) dropping and holding methods have been proposed for supporting cases where no available wavelength is achieved in an intermediate optical switch along a transmission route. In order to achieve shorter reservation time and higher throughput of application messages, this paper proposes a new method, partial waiting. Furthermore, a novel reservation protocol, i.e, a forward reservation protocol with partial waiting is designed.

1 背景と目的

近年、ネットワークアプリケーションの高度化とマルチメディア化、リアルタイム配送要求の高まりに対応するための広帯域ネットワークの技術開発が進められている。ここでは、物理層に光ファイバを用いた光ネットワークの構築が進められている。現在、従来の電気信号の伝達を基礎としたネットワークが大部分であり、これらとの共存が必要とされることから、スイッチ(ルータ)は電気信号を処理する装置として構成されている。そのため、光ネットワークとの接続には、OE(光-電気)変換、EO(電気-光)変換を必要としており、この処理による伝送遅延の拡大、スループットの低下が問題となっている。そこで、光スイッチを用いた全光ネットワークの実現が望まれる。

全光ネットワークにおける広帯域通信技術として WDM(Wavelength Division Multiplexing: 波長多重化)がある。これは、各光ファイバを同時に複数の異なる波長の光を用いた信号の交換に使用する多重化の実現手法である。WDMにおける多重化の手法は、パス多重化とリンク多重化に分類される。パス多重化は、送信元コンピュータから送信先コンピュータまでの経路上にあるすべてのリンクにおいて同一の波長の光を用いた通信を行なう多重化手法である。一方、リンク多重化は送信元コンピュータから送信先コンピュータまでの各リンクにおいて異なる波長の光を用いることを許す多重化の手法である。後者においては、ある経路において、あるスイッチへの入力波長と出力波長が異なる場合には、このス

スイッチにおける波長変換が必要となるため、これに要するオーバーヘッドが問題になる。一方、前者においては波長変換機能をスイッチが持たなくても良いために、変換オーバーヘッドの問題は回避することができる。しかし、送信元コンピュータから送信先コンピュータまでの経路上にあるすべてのリンクで同一の波長の光を用いなければならない。同一のリンク上を複数の異なる通信が同一の波長の光を用いて通信することはできないことから、経路上のすべてのリンクで他の通信に用いられていない波長が存在するときのみ通信が可能となる。

このような波長の検出を行なうために、波長の割り当て状況に関する情報を管理するための単一のサーバを用いる方法が提案されているが、大規模な全光ネットワークでの適用は通信のオーバーヘッドが大きく、困難である。これに対して、分散型手法が提案されている。分散型波長割り当て手法は、割り当て要求メッセージの配送方向、未割り当て波長の検出に失敗したときの処理方法、割り当て波長数により分類される。ここで、未割り当て波長の検出失敗時の処理方法については、ドロップ方式とホールディング方式が提案されている。ホールディング方式は、通信要求の発生に応じて動的に波長を割り当てる環境では、ドロップ方式よりも波長割り当ての成功率、メッセージのスループットが大きい。しかし、波長割り当て処理がある光スイッチでその波長の解放待ちによって一時的に停止している間は、他の光スイッチに対する波長の割り当て要求メッセージを転送しないために、波長割り当ての成功率、メッセージのスループット

トを低下させることになる。この問題を解決するために、本論文では、波長割り当てがある光スイッチでその波長の解放待ちによって一時的に停止している場合でも、波長の割り当て要求メッセージの転送を行なう部分ウェイト方式を提案する。また、これに基づいたプロトコルを設計する。

2 WDMにおける波長割り当て

本論文で対象とする全光ネットワークの概略を図1に示す。各コンピュータには、光信号を送受信するため

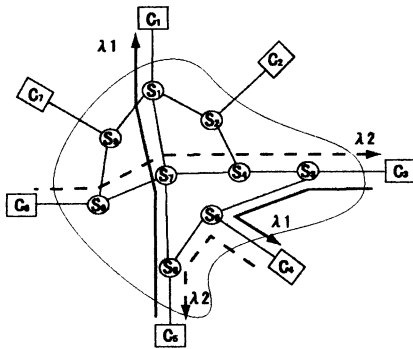


図1: 全光 WDM ネットワーク

のネットワークインタフェースが装着されているものとする。電気信号を送受信するためのインタフェースを持つコンピュータは、EO変換、OE変換を行なうメディアコンバータを介して全光ネットワークに接続される。本論文における波長割り当て問題においては、これらのコンピュータを対等に扱うことができるため、これらを区別することはしない。各コンピュータは、光ファイバによって1つ以上の光スイッチに接続される。各光スイッチは、光ファイバを介して1つ以上の他の光スイッチに相互接続されている。全光ネットワークに接続された2つのコンピュータ間の通信は、光ファイバを通して、1つ以上の光スイッチによる光信号の中継によって実現される。コンピュータと光スイッチおよび光スイッチ間を接続する光ファイバにおいては、互いに干渉しない複数の異なる波長の異なる光を同時に用いることによって、複数の通信を多重化することができる。これが波長多重化(WDM: Wavelength Division Multiplexing)である。各光ファイバでは、数波長から数1000波長の光を多重化することが可能であるが、全光ネットワーク全体が必要とされる通信、すなわち通信が行なわれる可能性のあるすべてのコンピュータ対 (C_s, C_d) に対して、使用する波長を静的に割り当てることは困難である。そこで、発生する通信要求に対して、使用する波長を動的に割り当て、通信の終了とともに割り当てられた波長を解放し、他の通信に割り当てる、という手法を用いる。

全光ネットワークで各通信に対して動的に波長を割り当てるWDMの実現方法は、リンク多重化とバス多重化に分類される。リンク多重化においては、各光ファイバリンクは独立に多重化される。送信元コンピュータ C_s から送信先コンピュータ C_d への配送経路上にある光ファイバリンク列 (L_1, \dots, L_l) ($l \geq 2$)において、 C_s と C_d との間の通信を実現するために、各光ファイバリンク L_i において割り当てられる波長 λ_i は、以下の要求条件を満足するように決定される。

- λ_i は、リンク L_i において他の通信に割り当てられていない。

この条件によって、 C_s と C_d との間の通信が光ファイバリンク L_i において他の通信の光信号によって干渉されることはない。リンク多重化では、各 λ_i は独立に決定できることから、各 L_i において1つ以上の未割り当て波長が存在するならば、 C_s と C_d との間の通信を実現することができる。ただし、 $\lambda_i \neq \lambda_{i+1}$ である場合には、これらのリンクに共通に接続する光スイッチにおいて波長変換を行なうことが必要とされる。そのため、各光スイッチには、波長変換機能を備えることが前提とされる。一方、バス多重化においては、送信元コンピュータ C_s から送信先コンピュータ C_d への配送経路上にある光ファイバリンク列 (L_1, \dots, L_l) において、 C_s と C_d との間の通信を実現するための各光ファイバリンク L_i に割り当てられる波長 λ_i は、以下の条件を満足することが求められる。

- λ_i は、リンク L_i において他の通信に割り当てられていない。
- すべての λ_i は等しい。

リンク多重化と同様、第1の条件によって、複数の通信がある光ファイバリンクにおいて同一波長の光を用いることを回避することができる。また、第2の条件によって、バス多重化では、光スイッチにおける波長変換を必要としない。 C_s から C_d まで単一波長の光を用いて通信するため、波長変換による信号の減衰を受けることがない点で優れている。しかし、 (L_1, \dots, L_l) で共通に割り当てられていない波長 λ が存在するとは限らない、という問題がある。すなわち、各光ファイバリンク L_i において他の通信に割り当てられていない波長の集合を F_i とすると、 C_s と C_d との間の通信に割り当てることができる波長 λ は $\lambda \in \bigcap_{1 \leq i \leq l} F_i$ である。そのため、 $\bigcap_{1 \leq i \leq l} F_i = \emptyset$ である場合には割り当てに失敗する。以降では、波長変換機能を光スイッチで必要としない点で導入が容易であり、また、波長変換による光信号の減衰を回避することによって、再送信や伝達遅延の拡大を避けることが可能なバス多重化手法について議論する。

C_s と C_d との間の通信に用いる波長 λ を決定するためには、 C_s と C_d との間の配送経路 (L_1, \dots, L_l) 上の各リンク L_i における他の通信に割り当てられていない波長の集合 F_i が必要となる。全光ネットワークのすべてのリンク L_i に対する F_i を単一のサーバで保持する集中型方式が提案されている[1, 2, 4, 6]。この方式では、単一サーバにすべてのリンクにおける波長割り当て状況に関する情報が保持されることから、未割り当て波長の検索と検出が容易に実現できる。しかし、通信要求の発生時と消滅時、すなわち、通信の開始時と終了時に必ずこのサーバにアクセスすることが必要となる。大規模な全光ネットワークにおいては、波長割り当ての要求と解放とをサーバに伝達する通信オーバーヘッドが生じるとともに、サーバに対して通信トラフィックの集中が起こるため、集中型手法を適用することは困難である。そこで、各リンクにおける波長の割り当て状況に関する情報を、そのリンクにおける送信元スイッチ(またはコンピュータ)で管理する分散型手法が提案されている[3, 5, 7, 8]。ここでは、光スイッチ S_i と S_j との間を接続する光ファイバリンク(以下では、光スイッチとコンピュータを接続するリンクに関しても同様であるとして、その記述を省略する。) L_{ij} における未割り当て波長の集合 F_{ij} を S_i が管理する。この方法では、集中型手法の問題を解決することが可能である。ただし、 C_s から C_d までの配送経路 (L_1, \dots, L_l) 上の各リンク L_i における他の通信に割

り当てられない波長の集合 F_i ($1 \leq i \leq l$) から、 C_s から C_d への通信に用いることが可能な波長 $\lambda \in \cap_{1 \leq i \leq l} F_i$ を決定するためには、これらのリンクに接続する光スイッチ間の通信プロトコルと割り当て波長の決定アルゴリズムが必要となる。WDMにおけるパス多重化のための分散型波長割り当て手法は、以下のように分類することができる。

- 割り当て要求メッセージの転送方向による分類
- 未割り当て波長検出失敗時の処理による分類
- 割り当て波長数による分類

以下の節では、それぞれについて説明する。

2.1 割り当てメッセージ転送方向

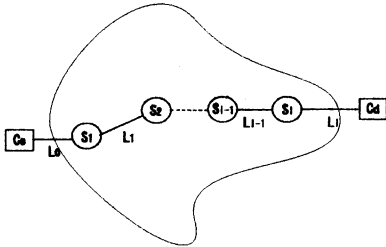


図 2: WDM 配送経路図

図 2 のように、送信元コンピュータ C_s から送信先コンピュータ C_d への配送経路上にある光スイッチ列を (S_1, \dots, S_l) 、 C_s と S_1 との間のリンクを L_0 、 S_i と S_{i+1} との間のリンクを L_i ($1 \leq i \leq l-1$)、 S_l と C_d との間のリンクを L_l とする。 C_s で発生した C_d へのメッセージ配送要求に対して、それをリンク列 $\langle L_0, \dots, L_l \rangle$ を経由して配送する際の波長 $\lambda \in \cap_{0 \leq i \leq l} F_i$ (ただし、 F_i はリンク L_i における未割り当て波長の集合) を検出し、各 L_i において λ を C_s と C_d との間の通信に割り当てる。割り当ての手続きは、 L_i の波長割り当て状況に関する情報を管理している S_i (ただし、 L_0 については C_s) において行なわれる。このとき、波長の割り当てが L_0 から順に行なわれるフォワードリザベーション方式と L_l から逆順に行なわれるバックワードリザベーション方式とに分類される。

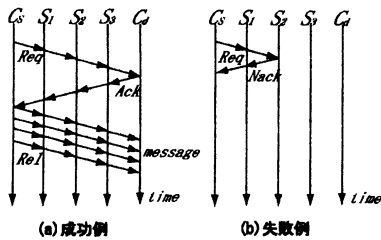


図 3: フォワードリザベーション方式

フォワードリザベーション方式(図 3)では、まず、 C_s から C_d へ波長要求メッセージ Req が (S_1, \dots, S_l) を順に配送される。 Req メッセージには、予約すべき波長の集合 $Req.Wavelength$ が含まれている。 Req メッセージを受信した S_i は、 C_s から C_d への通信に用いることができる波長が L_i の未割り当て波長の集合 F_i に存在するか

否かを調べる。すなわち、各 S_i では、受信した Req メッセージに含まれる $Req.Wavelength$ を $Req.Wavelength \cap F_i$ に更新する。もし $Req.Wavelength \neq \phi$ であるならば、 $Req.Wavelength$ に含まれる波長を L_i において C_s と C_d との間の通信に割り当て、更新された $Req.Wavelength$ を含む Req メッセージを S_{i+1} へ送信する。 $Req.Wavelength \neq \phi$ を満足したまま Req メッセージを C_d まで配送することができるならば、 C_d が受信した Req メッセージに含まれる $Req.Wavelength$ に含まれる波長 λ を用いて C_s と C_d との間の通信を行なうことができる。 C_d から C_s へ λ を含む肯定応答メッセージ Ack を (S_l, \dots, S_1) を逆順に配送する。このとき、各 S_i は、 L_{i-1} から受信した波長 λ の光信号を L_i へ転送するように設定するとともに、 L_i において C_s と C_d との間の通信に割り当てた λ 以外の波長を解放し、他の通信のための使用を可とする。 C_s は、 Ack を受信すると波長 λ を用いて C_d へのメッセージを L_0 を用いて送信することが可能となる。メッセージの配送が終了したならば、 C_s から C_d へ波長解放メッセージ Rel を (L_0, \dots, L_l) を順に配送する。 Rel メッセージを受信した S_i は、 L_i について予約した波長 λ を解放する。もし、 Req メッセージの配送中に S_i において $Req.Wavelength = \phi$ となる更新が行なわれたならば、この時点では $\cap_{0 \leq i \leq l} F_i = \phi$ であることになり、 C_s と C_d との間で $\langle L_0, \dots, L_l \rangle$ を経由する通信に割り当て可能な波長は存在しないことが検出される。次節に示す条件を満足することによって波長割り当てを中断する場合には、否定応答メッセージ $Nack$ を $\langle L_0, \dots, L_{i-1} \rangle$ を逆順に C_s まで配送する。 $Nack$ メッセージを受信した各 S_i は、 C_s と C_d との間の通信のために L_i において割り当てた波長を解放する。 $Nack$ メッセージを受信した C_s は、一定時間待機した後、波長の割り当て処理を再開することが可能である。 $\langle L_0, \dots, L_l \rangle$ の全体に共通の未割り当て波長が検出できないとき、 L_i に割り当てられた波長はメッセージの配送に使われない。さらに、この間、他の通信にこの波長を割り当ててを不可能にしてしまう。このフォワードリザベーション方式では、配送経路 $\langle L_0, \dots, L_l \rangle$ 上の各リンク L_i の S_i における波長割り当ての際に $\langle L_{i+1}, \dots, L_l \rangle$ の未割り当て波長についての情報を S_i が持たないことから、このように一時的に割り当てられながら、メッセージの配送には使われない波長割り当てを多く生じてしまうという問題点がある。

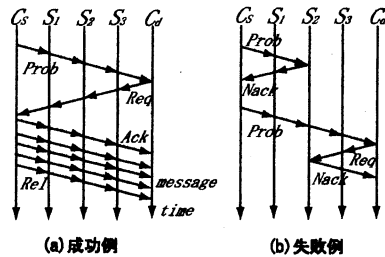


図 4: バックワードリザベーション方式

バックワードリザベーション方式は、この問題を解決する波長割り当て方式である(図 4)。ここでは、各リンク L_i における波長割り当てを行なう前に、 C_s から C_d への配送経路 $\langle L_0, \dots, L_l \rangle$ において共通の未割り当て波長 $\cap_{0 \leq i \leq l} F_i$ の検出を行なう。そして、検出された波長を用いて各リンク L_i における波長の割り当てを行なう。 $\lambda \notin \cap_{0 \leq i \leq l} F_i$ となる波長 λ の割り当て

を試みないことによって、割り当て処理の実行中に C_s と C_d との配送経路 (L_0, \dots, L_l) 上にあるリンク L_i において、波長割り当てに失敗する確率を低減する。まず、 C_s から C_d へ未割り当て波長検出メッセージ $Prob$ が (S_1, \dots, S_l) を順に配送される。 $Prob$ メッセージには、未割り当て波長の集合 $Prob.Wavelength$ が含まれている。 C_s では $Prob.Wavelength := F_0$ として、 L_0 を用いて $Prob$ メッセージが S_1 に送信される。このとき、 L_0 における波長割り当ては行なわれない。 $Prob$ メッセージを受信した S_i は、受信した $Prob$ メッセージに含まれる $Prob.Wavelength$ を $Prob.Wavelength \cap F_i$ に更新する。もし、 $Prob.Wavelength \neq \phi$ であるならば、更新した $Prob.Wavelength$ の値を含む $Prob$ メッセージを L_i を用いて S_{i+1} に送信する。このとき、 L_i における波長の割り当ては行なわれない。一方、 $Prob.Wavelength = \phi$ であるならば、 S_i は、この時点では $\cap_{0 \leq i < l} F_i = \phi$ であることを検出する。 S_i は、 $Nack$ メッセージを L_{i-1} を用いて S_{i-1} へ送信する。各 S_i が S_{i-1} へと $Nack$ メッセージを転送することにより、 C_s が S_1 から $Nack$ メッセージを受信することができる。 $Nack$ メッセージを受信した C_s は、一定時間待機した後、未割り当て波長の検出処理を再開することが可能である。さて、 C_d が $Prob$ メッセージを受信したならば、 $Prob.Wavelength \neq \phi$ である。この $Prob.Wavelength$ に含まれる波長は、この時点で $\cap_{0 \leq i < l} F_i$ に含まれる波長である。ここで、 $Probed := Prob.Wavelength$ とする。 C_d は、 $Probed$ に含まれる波長のみを対象として、各リンク L_i における波長割り当ての手続きを開始する。ここでは (L_0, \dots, L_l) に対して、 L_l から逆順に波長割り当てを行なう。 $Req.Wavelength \subset Probed$ を含む波長要求メッセージ Req を C_d が L_l を用いて S_l に送信する。 $Prob$ メッセージを受信した S_i は、受信した $Prob$ メッセージに含まれる $Prob.Wavelength$ を $Prob.Wavelength \cap F_i$ に更新する。もし、 $Prob.Wavelength \neq \phi$ であるならば、 L_i において、 $Prob.Wavelength$ に含まれる波長を C_s と C_d との間の通信に割り当て、更新された $Prob.Wavelength$ を含む $Prob$ メッセージを S_{i-1} (S_1 においては C_s) へ L_{i-1} を用いて送信する。同様に、 $Prob$ メッセージを受信した C_s は、受信した $Prob$ メッセージに含まれる $Prob.Wavelength$ を $Prob.Wavelength \cap F_0$ に更新する。もし $Prob.Wavelength \neq \phi$ であるならば、 L_0 において $Prob.Wavelength$ に含まれる波長を C_s と C_d との間の通信に割り当てる。ここで $Prob.Wavelength$ に含まれる波長のうちの1つである λ を用いて C_s と C_d との間の通信を行なうことができる。そこで、 C_s は、 $\lambda \in Prob.Wavelength$ を1つ選択し、 L_0 において C_s と C_d との間の通信に割り当てられた λ 以外の波長のうち、 λ 以外のものをすべて解放することによって他の通信に割り当ててを可とし、 λ を含む肯定応答メッセージ Ack を L_0 を用いて S_l に送信する。 L_{i-1} を用いて配送された Ack メッセージを受信した S_i は L_{i-1} から受信した波長 λ の光信号を L_i へ転送するように設定するとともに、 L_i において C_s と C_d との間の通信に割り当てられた波長のうち、 Ack メッセージに含まれる λ 以外のものを解放し、 Ack メッセージを L_i を用いて S_{i+1} (S_1 においては C_d) に送信する。 C_d では、受信した Ack メッセージを破棄するのみである。これによって、波長 λ を用いた C_s と C_d との間のメッセージ配送が可能となる。メッセージの配送が終了したならば、 C_s から C_d へ波長解放メッセージ Rel を (L_0, \dots, L_l) を順に配送する。 Rel メッセージを受信した S_i は、 L_i について予約した波長 λ を解放する。もし、 Req メッセージの配送中に S_i において $Req.Wavelength = \phi$ となる更新が行なわれたならば、この時点では $\cap_{0 \leq i < l} F_i = \phi$ である

ことになり、 C_s と C_d との間で (L_0, \dots, L_l) を経由する通信に割り当て可能な波長は存在しないことが検出される。 Req メッセージの配送は、 $Prob$ メッセージによる未割り当て波長の検出手続き終了後に行なわれるため、 $Prob$ メッセージ配送時に未割り当てであった波長も Req メッセージ配送時には割り当て済みであることが考えられる。そのため、 Req メッセージを受信した S_i (または C_s) において $Req.Wavelength = \phi$ となることがある。つまり、この時点では $(\cap_{0 \leq i < l} F_i) \cap Probed = \phi$ であることになり、 C_s と C_d との間で (L_0, \dots, L_l) を経由する通信に割り当て可能な波長が $Probed$ には存在しないことが検出される。次節に示す条件を満足することによって波長割り当てを中断する場合には、否定応答メッセージ $Nack$ を (L_i, \dots, L_l) を順に用いて C_d まで配送する。 $Nack$ メッセージを受信した各 S_i は、 C_s と C_d との間の通信のために L_i において割り当てた波長を解放する。 $Nack$ メッセージを受信した C_d は、一定時間待機した後、波長の割り当て処理を再開することができる。

2.2 未割り当て波長検出失敗処理

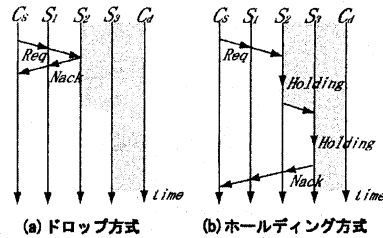


図5: ドロップ方式とホールディング方式

光スイッチ S_i において、 C_s と C_d との間の通信に用いることのできる (L_0, \dots, L_l) に共通の未割り当て波長の検出に失敗した場合の処理方法によって、WDM における波長割り当て手法は、ドロップ方式とホールディング方式とに分類することができる(図5)。ドロップ方式では、未割り当て波長の検出に失敗したならば、 S_i はただちに $Nack$ メッセージを送信する。この方式では、 S_i による検出の失敗までに割り当てを終わっている波長、すなわち、フォワードリザベーション方式では (L_0, \dots, L_{i-1}) において、バックワードリザベーション方式では (L_{i+1}, \dots, L_l) において割り当てを終わっている波長を短時間で解放することができる点で優れている。これによって、他の通信がこれらのリンクにおける波長の割り当てに成功する確率を高めることが可能となる。しかし、経路の大部分で割り当てに成功している場合 (S_i がフォワードリザベーション方式では C_d に、バックワードリザベーション方式では C_s に近い場合) には、波長を割り当てられたにも関わらずメッセージの配送に用いられないリンクが多数、長時間存在することになり、これらが他の通信の波長割り当てを妨げる可能性が高くなる。また、フォワードリザベーション方式では C_s 、バックワードリザベーション方式では C_d に近いリンクでは、波長の割り当てと失敗による解放が繰り返されることになる。

一方、ホールディング方式は、未割り当て波長の検出に失敗した S_i が、 L_i における波長割り当てが可能になることを一定時間待つ方式である。フォワードリザベーション方式では、 S_i は S_{i-1} から Req メッセージを受信するが、リンク L_i の未割り当て波長の集合 F_i に対して、 $Req.Wavelength \cap F_i = \phi$ を得る。ここで、ドロップ

ブ方式が直ちに *Nack* メッセージを S_{i-1} に送信するのに対して、ホールディング方式では、タイマを設定して、 $Req. Wavelength \cap F_i \neq \phi$ となるのを待つ。もし、タイマ切れ以前に L_i において波長 $\lambda \in Req. Wavelength$ が解放されたならば、 L_i について λ を C_s と C_d との間の通信に割り当て、 $Req. Wavelength := \{\lambda\}$ とした *Req* メッセージを L_i を通して S_{i+1} に送信する。このような波長の解放が発生しないままタイマ切れが発生した場合には、*Nack* メッセージを用いてドロップ方式と同様に $\langle L_0, \dots, L_{i-1} \rangle$ において割り当て済みの波長を解放する。同様に、バックワードリザベーション方式では、 S_i は S_{i+1} から *Req* を受信するが、リンク L_i の未割り当て波長の集合 F_i に対して、 $Req. Wavelength \cap F_i = \phi$ を得る。ここで、ドロップ方式がただちに *Nack* メッセージを S_{i+1} に送信するのに対して、ホールディング方式では、タイマを設定して $Req. Wavelength \cap F_i \neq \phi$ となるのを待つ。もし、タイマ切れ以前に L_i において波長 $\lambda \in Req. Wavelength$ が解放されたならば、 L_i について λ を C_s と C_d との間の通信に割り当て、 $Req. Wavelength := \{\lambda\}$ とした *Req* メッセージを L_{i-1} を通して S_{i-1} に送信する。このような波長の解放が発生しないまま、タイマ切れが発生した場合には、*Nack* メッセージを用いて、ドロップ方式と同様に $\langle L_{i+1}, \dots, L_i \rangle$ において割り当て済みの波長を解放する。この方式では、 S_i がただちに割り当て済みの波長の解放を行なうことによって割り当て処理を中断するドロップ方式に対して、待機時間を設定することによって、波長の割り当て後にメッセージの配送が行なわれなまま解放されるリンクの発生を減少させることができる。ただし、ホールディングタイマのタイムアウトによって割り当てられた波長を解放した場合には、ドロップ方式と比べてより長い時間、メッセージ配送に用いられなまま、この波長を他の通信に対して割り当てられることを妨げるようになるという欠点がある。

2.3 割り当て波長数

Req メッセージの配送によって、各 S_i はリンク L_i における波長の割り当てを行なう。ここでは、 L_i の未割り当て波長の集合を F_i として、受信した *Req* メッセージに含まれる $Req. Wavelength$ を $Req. Wavelength \cap F_i$ に更新し、これが ϕ でない場合には $Req. Wavelength$ に含まれるすべての波長を L_i において C_s と C_d との間の通信に割り当てる。フォワードリザベーション方式では、 C_s から送信される *Req* メッセージには、 L_0 における未割り当て波長の集合 F_0 の部分集合 (ϕ を除く) を $Req. Wavelength$ として含むことができる。同様に、バックワードリザベーション方式では、 C_d から送信される *Req* メッセージには、 L_i における未割り当て波長の集合 F_i の部分集合 (ϕ を除く) を $Req. Wavelength$ として含むことができる。 $Req. Wavelength := F_0$ (または F_i) とする方式をアグレッシブ方式、 $\exists \lambda \in F_0$ (または F_i) として $Req. Wavelength := \{\lambda\}$ とする方式をコンサーバティブ方式という。 S_i において、 $Req. Wavelength = \phi$ となったときに割り当て処理が中断することから、アグレッシブ方式の方がコンサーバティブ方式に比べて $\langle L_0, \dots, L_i \rangle$ 全体の波長割り当てに成功する確率が高い。ただし、各リンク L_i において、より多数の波長を割り当てることから、他の通信における波長割り当てを妨げる可能性も高いという問題がある。未割り当て波長の検出に失敗し、*Nack* メッセージが返送された場合には、アグレッシブ方式では一般に、ある一定の時間待機するのに対し、コンサーバティブ方式では失敗したものは異なる波長 $\lambda' \neq \lambda$ を F_0 (または F_i) から選択して、ただちに *Req* メッセージを送信する方法が一般に用いられる。

3 部分ウェイト方式

前節までの3項目による分類にしたがって導き出される $2^3 = 8$ 通りの波長割り当てプロトコルが論文 [3] などに提案されている。ここで2.2節で述べた未割り当て波長の検出に失敗した場合の処理について議論する。2.2節では以下の2方式について述べた。

- ドロップ方式
- ホールディング方式

ドロップ方式では、 C_s から C_d への配送経路 $\langle L_0, \dots, L_i \rangle$ 上のすべてのリンクでの波長割り当てが成功しない場合は、ただちにこれらを解放することによって、これらのリンク (の一部) を配送経路として用いる他の通信の波長割り当ての成功率を低減させないという性質がある。しかし、通信要求の発生に対して動的に波長を割り当てる WDM 環境においては、 S_i において L_i の未割り当て波長の集合 F_i と $\langle L_0, \dots, L_{i-1} \rangle$ で共通に未割り当て状態にあった波長の集合 $Req. Wavelength$ (S_i が受信した *Req* メッセージに含まれるもの) との積集合 $Req. Wavelength \cap F_i$ が ϕ であるとしても、一定時間内に $Req. Wavelength$ に含まれる波長が解放されることが期待できる。論文 [3] では、ホールディング方式がドロップ方式よりもより高い波長割り当て成功率を実現していることが示されている。

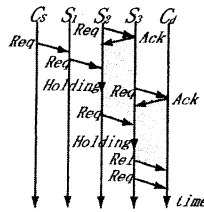


図6: ホールディング方式の問題点

ところが、ホールディング方式では、 S_i における波長割り当てにおいて波長の解放待ちが発生した場合、波長割り当て手続きが開始されていない他のリンクの割り当て状況についての情報を管理する光スイッチへは *Req* メッセージを配送しない方法を採用している。これは、 S_i における L_i での波長割り当てに失敗した場合、それまでに割り当てた波長は、各リンクにおいてメッセージの配送に用いられなまま解放される無駄な割り当てとなり、他の通信への波長割り当てを妨げると考えているからである。ここで図6のフォワードリザベーション方式を考える。ここでは $\langle L_0, \dots, L_{i-1} \rangle$ では、波長 λ の割り当てに成功したが、 S_i において L_i での λ の解放を待っている状態を示している。このとき、 $\langle L_{i+1}, \dots, L_i \rangle$ では、波長 λ は他の通信に割り当てられていないが、 S_i での解放待ちの間に S_j ($i+1 \leq j \leq l$) で他の通信に λ が割り当てられてしまい、 S_i での λ の解放後に S_j で再度解放待ちになってしまうことが考えられる。このように、解放待ちが解除されたあと、再び他のリンクで解放待ちとなることが繰り返されると、 $\langle L_0, \dots, L_i \rangle$ 全体での λ の割り当てに要する時間が長くなる。この間、 $\langle L_0, \dots, L_{i-1} \rangle$ で割り当てられている波長はメッセージの配送に用いることができず、また、他の通信にも割り当てることができない。このため、全光ネットワーク全体としての波長割り当ての成功率およびメッセージの配送のスループットを低下させることになる。

そこで、本論文では、未割り当て波長の検出に失敗した場合の光スイッチの処理方式に基づく波長割り当て方式の分類に、従来提案されていたドロップ方式、ホールディング方式に加えて部分ウェイト方式を導入する。ここでは、光スイッチ S_i においてリンク L_i に割り当て可能な波長が存在しない場合、ホールディング方式と同様に、 L_i における割り当てについてはタイマを導入し、タイムアウトまでに現在は他の通信に割り当てられている光スイッチでは波長要求メッセージ Req の送信を行わないのに対して、部分ウェイト方式では Req の送信を行なう。このとき、 S_i が受信した Req メッセージに含まれる $Req.Wavelength$ に含まれる波長のうち、どれが解放されるかを予測することは困難であることから、 S_i は受信した Req メッセージに含まれる $Req.Wavelength$ の値を変更せずに、そのままコピーした Req メッセージを送信することとする。他の通信に割り当てられた波長の解放によって S_i において L_i の波長の割り当てが可能となったことを他の光スイッチに通知する方法は、他の分類項目(割り当て要求メッセージの配送方向と割り当て波長数)によって異なることから、本節では述べない。部分ウェイト方式の導入によって、他の2つの分類項目(割り当て要求メッセージの配送方向と割り当て波長数)との組み合わせによって4種類の波長割り当てプロトコルを設計することが可能である。本論文では、割り当て波長数を1としたコンサバティブ方式に基づくプロトコルのうち、フォワードリザベーション方式を用いるものについて以下で述べる。

3.1 フォワードリザベーション方式

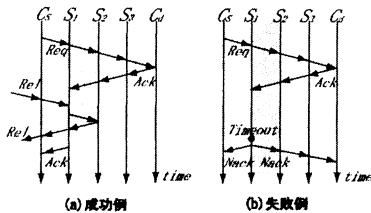


図 7: 部分ウェイトフォワードリザベーション方式

部分ウェイト方式を導入した、フォワードリザベーション・コンサバティブ方式による WDM 波長割り当てプロトコルを設計する(図 7)。送信元コンピュータ C_0 から送信先コンピュータ C_d までの配送経路は、光スイッチ列 $\{S_1, \dots, S_i\}$ からなっており、 C_0 と S_1 との間のリンクを L_0 、 S_i と S_{i+1} との間のリンクを L_i 、 S_i と C_d との間のリンクを L_i とする。また、各 L_i の未割り当て波長の集合を F_i とし、 S_i がこの情報を管理するものとする。 C_0 は、波長 $\lambda \in F_0$ を選択すると、 $Req.Wavelength := \{\lambda\}$ の値を含む Req メッセージを L_0 を用いて S_1 に送信する。各 L_i において、割り当て要求波長 λ が他の通信に割り当て済みでないならば、 λ を C_0 と C_d との間の通信に割り当てる。一方、 λ が他の通信に割り当て済みであるならば、 S_i は λ が L_i において解放されるのをタイマを設定して待つとともに、配送経路上の次の光スイッチ S_{i+1} に λ の割り当て要求メッセージ Req を送信する。 S_i は解放待ちの間に肯定応答メッセージ Ack を受信しても、それを転送せずに保持する。また、 $Nack$ を受信した場合についてはタイムアウト以前に L_i で λ の割り当てが S_i で可能となったならば、受信済みあるいは以降に受信する Ack メッセージ

の転送を可とする。 C_0 は、 S_1 からの Ack メッセージを受信することによって $\{L_0, \dots, L_i\}$ における波長 λ の割り当てを完了したことを検出し、メッセージの配送を開始する。一方、タイムアウトにより L_i における λ の割り当てが不可能とされた場合には、 λ の値を含む $Nack$ メッセージを S_{i-1} と S_{i+1} の両方に送信する。 S_{j-1} から $Nack$ メッセージを受信した S_j は、 L_j における波長 λ の割り当てを解放するとともに、 $Nack$ メッセージを S_{j+1} に転送する。同様に、 S_{j+1} から $Nack$ メッセージを受信した S_j は、 L_j における波長 λ の割り当てを解放するとともに、 $Nack$ メッセージを S_{j-1} に転送する。なお、 S_j が L_j における λ の解放待ちである場合には、この待ち状態を解除するものとする。 $Nack$ メッセージを受信した C_0 は、 $\lambda' \in F_0$ 、 $\lambda' \neq \lambda$ を満足する波長 λ' の割り当てを波長 λ の場合と同様にして開始する。

4 まとめと今後の課題

本論文では、WDM における波長割り当て手法の分類項目である未割り当て波長の検出に失敗した場合の処理として、従来のドロップ方式とホールディング方式とは異なる「部分ウェイト方式」を提案した。本方式の導入により、波長の割り当てに要する時間オーバーヘッドを削減することができる。また、これを用いた単一波長割り当てプロトコルの1つを設計した。本論文ではコンサバティブ方式のみを対象としたが、アグレッシブ方式でも同様のプロトコルを設計することが可能であると考えられる。今後はそれらのプロトコルを設計するとともに性能評価実験によるそれぞれの手法の利点と欠点を明らかにする。

参考文献

- [1] Chlamtac, I., Ganz, A. and Karmi, G., "Ligth-path Communications : An Approach to High-Bandwidth optical WANs," IEEE Transactions on Communications, vol. 40, No.7 pp. 1171-1182(1992).
- [2] Lee, K.C. and Li, O.K., "A Circuit Rerouting Algorithm for All-optical Widearea Networks," Proceedings of the 13th IEEE INFOCOM, pp. 954-961(1994).
- [3] Mei, Y. and Qiao, C., "Efficient Distributed Control Protocols for WDM All-Optical Networks," Proceedings of the 6th International Conference on Computer Communications and Networks, pp. 150-153 (1997).
- [4] Ramaswami, R. and Sivarajan, K.N., "Optical Routing and Wavelength Assignment in All-optical Networks," Proceedings of the 13th IEEE INFOCOM, pp. 534-543(1994).
- [5] Ramaswami, R. and Segall, A., "Distributed Network Control for Wavelength Routed Optical Network," Proceeding of the 15th IEEE INFOCOM, pp. 138-147(1996).
- [6] Subramaniam, S., Azizoglu, M. and Somani, A.K., "Connectivity and Soarse Wavelength Conversion in Wavelength-routing Networks," Proneedings of the 15th IEEE INFOCOM, pp. 148-155(1996).
- [7] Qiao, C. and Mei, Y., "Wavelength Reservation Under Distributed Control," Proceedings of the IEEE/LEOS Summer Topical Meeting on Broadband Optical Networks, pp. 45-46 (1996).
- [8] Yuan, X., Melhem, R. and GuptaGupta, R., "Distributed Path Reservation Algorithm for Multiplexed All-Optical Interconnection Networks," Proceedings of the 3rd IEEE Symposium on High Performance Computer Architecture, pp. 38-47(1997).