社団法人 電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
IEICE Technical Report

レイヤ１帯域オンデマンドサービスにおける スケジューリングアルゴリズムの基礎的検討

川崎 考蔵† 鯉渕 道紘‡1 漆谷 重雄‡1 廣安 知之‡2 三木光範‡2

†同志社大学大学院 〒610-0321 京都府京田辺市多々羅都谷 1-3

‡1 国立情報学研究所 〒101-8430 東京都千代田区一ツ橋 2-1-2

‡2 同志社大学 〒610-0321 京都府京田辺市多々羅都谷 1-3

E-mail :†kkawasaki@mikilab.doshisha.ac.jp, ‡1{koibuchi,urushi}@nii.ac.jp, ‡2 {tomo@is, mmiki@mail}.doshisha.ac.jp

あらまし 学術情報ネットワークなどの先進的なインターネットバックボーンではサービスとして柔軟なリソース配分による QoS，Bandwidth-on-Demand (BoD) などが求められている．本稿では，具体的な例として我が国の学術情報ネットワークである SINET3 におけるレイヤ 1 帯域オンデマンドサービスを念頭におき，ユーザのリクエストによるパスの優先度決定アルゴリズムを提案する．そして，複数のスケジューリングアルゴリズムと比較を行うことにより提案アルゴリズムの有効性の検証を行った．提案アルゴリズムは各ユーザの公平性を考慮したアルゴリズムであり，シミュレーション結果より提案アルゴリズムが有効であることが分かった．

キーワード 帯域オンデマンドサービス，SINET3, スケジューリング，インターネットバックボーン

Examination of Scheduling Algorithms of Layer-1 Bandwidth-on-Demand Service

Kozo KAWASAKI†, Michihiro KOIBUCHI‡1 and Shigeo URUSHIDANI‡1

Tomoyuki HIROYASU‡2 and Mitsunori MIKI‡2

†Graduate School of Engineering, Doshisha University 1-3 Tataramiyakodani, Kyotanabe, Kyoto, 610-0321 Japan

‡1 National Institute of Informatics 2-1-2 Hitotsubashi,Chiyodaku,Tokyo,101-8430 Japan

‡2 Graduate School of Engineering, Doshisha University 1-3 Tataramiyakodani, Kyotanabe, Kyoto, 610-0321 Japan

E-mail :†kkawasaki@mikilab.doshisha.ac.jp, ‡1{koibuchi,urushi}@nii.ac.jp, ‡2 {tomo@is, mmiki@mail}.doshisha.ac.jp

Abstract Network services that flexibly allocate resources, such as QoS and bandwidth-on-demand (BoD) are increasingly required to advanced Internet backbones that include science information networks. In this paper, we propose a user's request priority algorithm for the advanced Internet, especially Japanese academic information network called SINET3. The proposed algorithm provides each user's fairness, and simulation results show that they fit with advanced Internet backbone.

Keyword Bandwidth-on-Demand service，SINET3, scheduling algorithm, Internet backbone

1. はじめに

学術情報ネットワークなどの先進的なインターネットでは，増大するユーザからの需要(遠隔会議，グリッド等のリアルタイム性が要求される大規模データ転送等)により，高品質サービスとして柔軟な資源配分による QoS，Bandwidth-on-Demand (BoD) などが求められている[1]．我が国の学術情報ネットワークである SINET3 ではレイヤ 1 帯域オンデマンドサービス[2](以下 L1 BOD)を現在，試験的に提供しており，今後幅広いユーザが利用できるようになる見込みである．SINET3 における L1 BOD では，ユーザに任意の拠点間におけるネットワーク帯域を任意の時間帯で提供する[3]．ユーザは，使用したい拠点間，帯域，日時等を指定することにより，その時間帯で，そのユーザ専用のネットワーク帯域を専用線と同じように使うことができる．このサービスにより，ユーザはネットワークリソースを高品質で利用することが可能となる．これは，従来は高品質を得るために専用線を用いてきたユーザをインターネットバックボーンに収容することを可能にする技術であり，ネットワークの資源利用効率を向上させることも可能である．

ただし，L1 BOD において複数のユーザで帯域を確保したい拠点間の使用リンク，使用日時が重複すると，全ユーザの希望した通りにネットワークを用いることができない場合が生じる．そのような状況が発生した場合，どのユーザの帯域予約(リクエスト)を受理し，またどのユーザのリクエストを却下するかを決定するリクエスト

スケジューリングが重要となる．

本報告では，L1 BOD におけるユーザのリクエストスケジューリングを決定するためのアルゴリズムの提案，検討を行う．そして，実インターネットバックボーンである SINET3 の L1 BOD を対象として具体的な検討，シミュレーションによる評価を行う．なお，本研究成果は SINET3 L1 BOD の運用，今後の方針とは無関係である．

2. レイヤ 1 帯域オンデマンドサービス（L1 BOD）

具体的な L1 BOD の例として，ここでは SINET3 について説明する．

SINET3 における L1 BOD の構成を Fig.1 に示す．L1 BOD は，主にユーザ，ユーザ装置，ネットワーク，オンデマンドサーバで構成される．

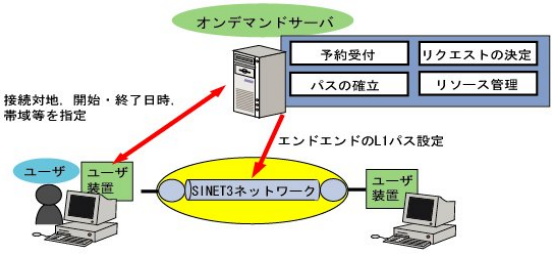


Fig.1　L1 BOD の構成

ユーザの希望パスのリクエストからパス確立までのプロセスについて以下に述べる．

・予約開始

ユーザは Web ベースのインタフェースを介してオンデマンドサーバに帯域確保のリクエストを行う．

・発着ノード・日時設定

ユーザは，パスを確立する発着拠点と，そのパスを使用してネットワーク通信を行いたい開始時間および終了時間をリクエストとして設定し，オンデマンドサーバに送信する．

・帯域・経路条件の設定

ユーザは，提示された使用可能帯域の範囲内で，使用帯域(150Mbps×n)と，経路条件を設定する．経路条件によってネットワーク通信の遅延時間を最小にするネットワーク経路を希望するか，遅延時間が最小ではない経路で通信を行うことを許可するかを決定する．

・要求受付完了通知

オンデマンドサーバは，上記の一連のプロセスを終了すると，ユーザの要求受付を完了し，その旨をユーザに通知する．

・優先度による抽選

オンデマンドサーバは，サブミットされた複数のリクエストをスケジューリング契機まで保持する．スケジューリング契機になると保持しているリクエストに関して，要求している時間，経路，帯域でパスの確保が可能かどうかを判断する．もし，パス確立が不可能なリクエストが出現した場合は，ある特定のアルゴリズムによって，保持したリクエスト群の中から受理するリクエストを決定し,その他のリクエストは却下する．

・予約成立または不成立の通知

オンデマンドサーバは，あるリクエストについて予約の成立・不成立の旨をそのリクエストをサブミットしたユーザに通知する．

・アベイラブルリソースの再計算

オンデマンドサーバは，ある予約が受理されると各リンクの使用予定状況を再計算する．

・パスの確立

オンデマンドサーバは，受理した各リクエストのパス確立の希望使用時間となれば，そのリクエストが希望する接続対地間のリンクのパスを希望する帯域分だけ確立する．

3. スケジューリングアルゴリズムの検討

3.1. スケジューリングアルゴリズムの要件

L1 BOD におけるスケジューリングアルゴリズムを検討する上で考慮しなければならない要件を以下に述べる．

(1) サブミットされた時期が早いリクエスト程優先

L1 BOD は予約サービスであるため，早くサブミットしたユーザのリクエスト程優先される．ただし，次条件の資源の利用の平等性が考慮される．

(2) 全ユーザが平等にネットワークリソースを利用

複数のユーザが SINET3 のネットワークリソースであるリンクの帯域を共有して利用する．そのため，ある特定のユーザによって常にリソースが占有され，他のユーザがリンクを利用することができないという状況は避けるべきである．

(3) 全体のスループットを高く保つ

常にリクエストの処理数や，各リンクの使用率を高く保つことによって，サービスのスループットを高く保つことが望まれる．

3.2. スケジューリングアルゴリズム

L1 BOD で必要となるスケジューリングアルゴリズムは，ネットワーク資源の一部を一時的に占有するユーザを決定する．これは，PC クラスタ，並列計算機などの計算資源をユーザに配分するスケジューリングアルゴリズムの概念と極めて似ている．そこで，本報告では，PC クラスタ，並列計算機において一般的に利用されている Condor[4]スケジューリングアルゴリズムを L1 BOD に適用することを提案する．

3.2.1. Condor スケジューリングアルゴリズム

Condor は Wisconsin 大学[5]において開発され，分散コンピューティングにおいて，High Throughput Computing を念頭に遊休ノードを効率的に利用することを目的とし

たジョブスケジューリングシステムである．分散コンピューティング環境においては，ユーザ，リソースが複数存在する．そのため，あるユーザにリソースの分配が偏ってしまい，他のユーザがリソースを利用することができないという状況が想定される．

Condor は各ユーザに平等にリソースを分配するため，特有の優先度設定アルゴリズムを用いることにより，それぞれのユーザの優先度を設定している．これらの考え方、目的は BoD の優先度決定アルゴリズムのものと近い．

そこで本報告では Condor の優先度決定アルゴリズムを L1OD の優先度決定に応用させる．具体的には以下のような優先度決定式を利用して，ユーザのリクエストの優先順位を決定する．

$$priority(t) = \quad 0.5^{(dt/h)} \times Priority(t\text{-}dt) + (1\text{-}\ 0.5^{(dt/h)}) \times \rho(t)$$

上式において dt はサーバのスケジューリング間隔を表し，$\rho(t)$は時刻 t におけるユーザのリソース占有度を表す．h は優先度の半減期を表し，この値により，優先度の回復の早さを設定することが可能である．

3.2.2. L1 BOD スケジューリングアルゴリズム

上式を L1 BOD に適用する場合，リソース占有度を定義する必要がある．ここでは「リソース占有度ρ」を単純に確保した帯域と時間を用いて以下のように与えることにする．

$$\rho \ = Bands(Mbps) \ \times \ Time(step)$$

Bands はあるリクエストが使用する帯域幅であり，*Time* はその帯域幅を利用する時間である．

また，より採択率やネットワーク資源の平等性を考慮に入れた占有度として以下のようにすることも考える．

$$\rho \ = Bands(Mbps) \ \times \ Time(step) \ \times \ use\ Link\ num$$

use Link num はリクエストが使用するリンクの数を表す．以降からは使用リンク数を考慮に入れないアルゴリズムを提案アルゴリズム 1，リンク数を考慮に入れるアルゴリズムを提案アルゴリズム 2 とする．

提案するオンデマンドサーバにおける処理プロセスは Fig.2 の通りである．

オンデマンドサーバはスケジューリングが行われる時間になるまで待機する．その間，ユーザのリクエストを随時受け取り，オンデマンドサーバが持つキューに格納しておく．スケジューリング契機となれば，キューにユーザのリクエストが存在するかを確認する．キューにリクエストが格納されていない場合は，再度スケジューリング契機まで待機する．キューにユーザのリクエストが格納されている場合は，優先度決定式によりキューに格納されている各リクエストの優先度を計算する．

キューに格納されているリクエストを優先度の昇順に並び替え，優先度の小さいリクエストから順にキューから取り出して処理を行う．キューからリクエストを取り出せば，はじめに，そのリクエストを最も通信遅延の少ない経路(最小遅延経路)でそのリクエストを満足することが可能かを確認する．最小遅延経路でそのリクエストを満足することが可能ならば，その条件でリクエストを受理する．最小遅延経路でリクエストを満足することが不可能ならば，そのリクエストが非最小遅延経路での通信を許可するかを確認する．非最小遅延経路での通信を許可しない場合，そのリクエストを却下する．非最小遅延経路での通信を許可する場合は，その時点でそのリクエストを満足することができる最も通信遅延の少ない経路でリクエストを受理する．いかなる経路もそのリクエストを満足することが不可能な場合はそのリクエストを却下する．

1 つのリクエストを受理または，却下すれば，そのリクエストを処理済みとしてキューから除外する．そしてキューにリクエストが存在するかを確認するプロセスに戻る．

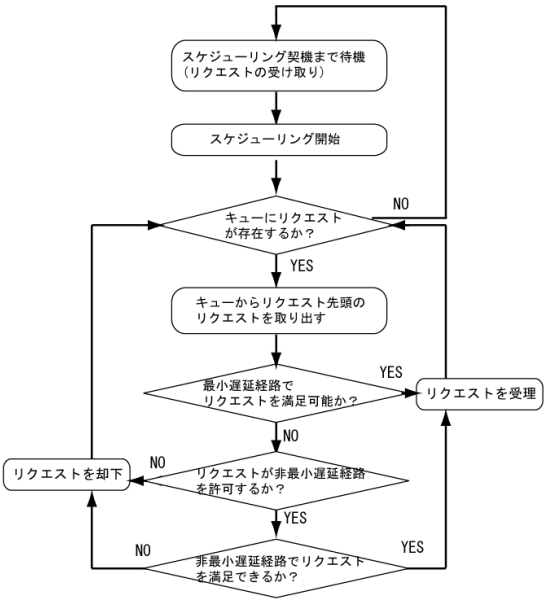


Fig.2　提案スケジューリングアルゴリズム

4. スケジューリングアルゴリズムの検証

4.1. L1 BOD シミュレータの実装

3.2.2.にて設計したリクエストスケジューリングアルゴリズムの有効性を検証するため，L1 BOD を模擬したシミュレータを作成した．作成したシミュレータはユーザによるリクエストの発生から，サーバによる帯域の確保またはユーザのリクエストの reject までを模擬する．作成したシミュレータを利用して 3 種のトポロジを利用して，数値計算を行った．

オンデマンドサーバにおけるパラメータは(1)スケジューリング間隔，(2)提案アルゴリズムにおける半減期である．また，ユーザに関するパラメータは(1)接続開始拠点，(2)接続終端拠点，(3)希望使用帯域，(4)希望使用時間，(5)リクエスト送信間隔，(6)通信遅延の許可または禁止である．ユーザはこれらの情報を基にして定期的にリクエストの送信をオンデマンドサーバに行う．リンクは

各両端の拠点の情報を保持しており，(1)許容帯域，(2)通信遅延が存在する．

4.2. シミュレーション結果

4.2.1. Topology1

Fig.3 に示すようなシンプルなトポロジーにおいてシミュレーションを行い，提案アルゴリズムにより，どのようなスケジューリングがなされるかを確認した．このときの各ユーザの行動制御を Table.1 に示す．Table.1 において，from はユーザが所属する拠点を表し，to はユーザが接続を希望する拠点を表す．allow latency は通信遅延を許すかどうかということを表し，True ならば最小遅延の経路以外の経路での接続を許可する．use band は使用する帯域を示し，use time は接続時間を示す．start at はリクエストをサブミットして何 step 後に帯域を確保するかを示し，wait time はリクエストを送信した後，その値の step だけ待機することを示す．

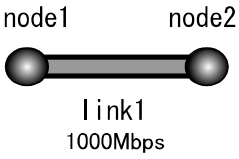
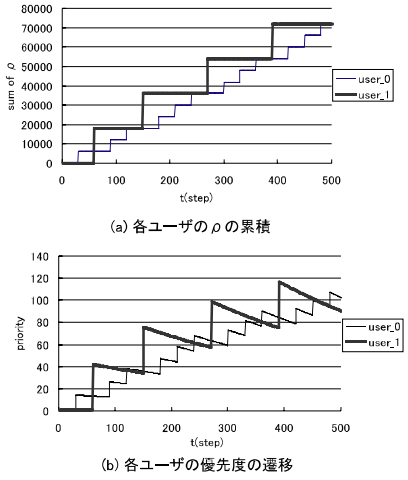


Fig.3 Topology1

Table.1 Topology1 におけるユーザの行動制御

user name	from	to	allow latency	use bands	use time	start at	waittime
user_1	node1	node2	FALSE	300	20	10	30
user_2	node1	node2	FALSE	900	20	10	30

また，サーバは 1step 毎にスケジューリングを行い，優先度の半減期は 150step とした．シミュレーション結果を Fig.4 に示す．



(a) 各ユーザの ρ の累積

(b) 各ユーザの優先度の遷移

Fig.4 Topology1 のシミュレーション結果

Fig.4 の(a)から，500step までで，同程度の占有度が各ユーザに与えられていることが確認できる．また(b)から，(a)でユーザのリクエストが受理されると priority 値が上昇していることが確認できる．これにより次回以降のスケジューリングにその情報が利用されていること確認できる．

4.2.2. Topology2

Fig.5 に示すような複数経路が存在するメッシュ型のトポロジにおいてシミュレーションを行った．また提案アルゴリズムの有効性の検証を行うため，ランダムに accept を行うリクエストを決定する「random アルゴリズム」,最近最も利用されていないユーザのリクエストを優先する Least Recently Used な考えを利用した「LRU アルゴリズム」，単純に accept 回数が少ないユーザを優先する「accept number アルゴリズム」として，それぞれ同条件においてシミュレーションを行った．また提案アルゴリズムにおいては，半減期を 10step，100step，1000step においてシミュレーションを行った．ユーザの行動制御を Table.2 に示す．また各リンクは限界帯域を 1000Mbps として，遅延時間は全てのリンクにおいて同一に設定した.それぞれのシミュレーション結果を Fig.6～8 に示す．

Fig.6,7 において，(a)，(b)，(c)はそれぞれ提案アルゴリズム 1,2 における半減期を 10step，100step，1000step としたときの占有度の遷移を表している．結果から提案アルゴリズム 1,2 は他のアルゴリズムよりも各ユーザの step 毎の分散が少ないことが確認できる．各ユーザの占有度の累積の遷移はそれぞれのユーザで異なっていることが確認できる．これは及びユーザが所属する拠点の位置が大きく影響を与えていると考えられる．特に高い占有度の遷移を持つユーザは User_3 である．User_3 は，リクエストの希望する帯域は他のユーザより低く，希望使用時間が比較的長い．これにより，帯域を長く利用でき，かつネットワークの空き容量が少なくても希望帯域が少ないために採択され易くなっていると考えられる．また，半減期が長くなるほど，占有度の遷移が近いユーザがさらに遷移が近くなり，優先度のばらつきも少なくなることが確認できる．また，提案アルゴリズム 2 の結果から,使用リンク数の少なくなる User_5 は占有度が高く遷移することが確認できる．

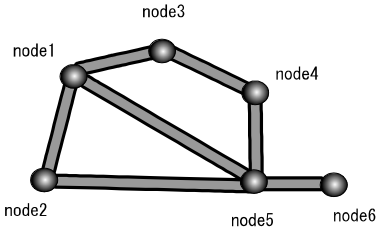


Fig.5 Topology2

Table.2 Topology2 におけるユーザの行動制御

user name	from	to	allow latency	use bands	use time	start at	wait time
user_1	node1	node6	FALSE	642	13	17	30
user_2	node2	node6	TRUE	529	14	28	30
user_3	node3	node6	TRUE	364	19	14	30
user_4	node4	node6	FALSE	483	12	22	30
user_5	node5	node6	FALSE	654	19	11	30

4.2.3. Topology3

ユーザが接続開始する場所が大きく影響するトポロジとして，Fig.9 に示すようなツリー型のトポロジにおいてシミュレーションを行った．

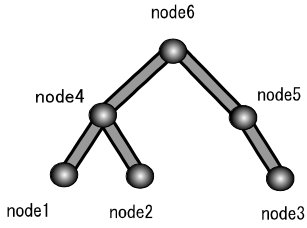


Fig.9 Topology3

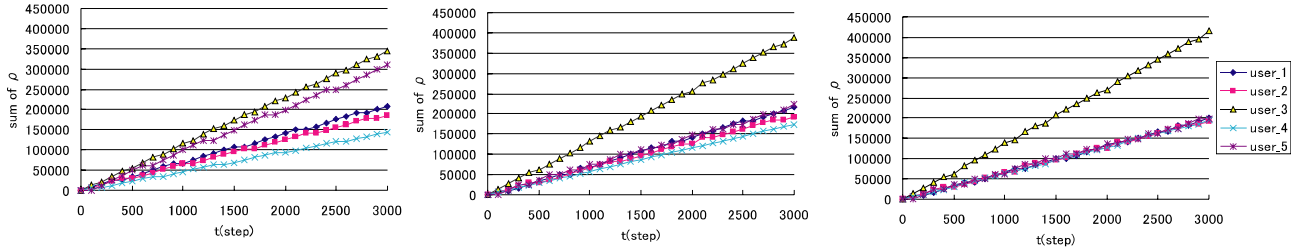
また 4.2.2.と同様に，提案アルゴリズムの他に「random アルゴリズム」「LRU アルゴリズム」「accept number アルゴリズム」においてシミュレーションを行った．ユーザの行動制御を Table.3 に示す．

Table.3 Topology3 におけるユーザの行動制御

user name	from	to	allow latency	use bands	use time	start at	wait time
user_1	node1	node6	TRUE	745	12	19	30
user_2	node2	node6	FALSE	227	25	15	30
user_3	node3	node6	TRUE	269	28	11	30
user_4	node4	node6	FALSE	456	24	11	30
user_5	node5	node6	FALSE	584	14	17	30

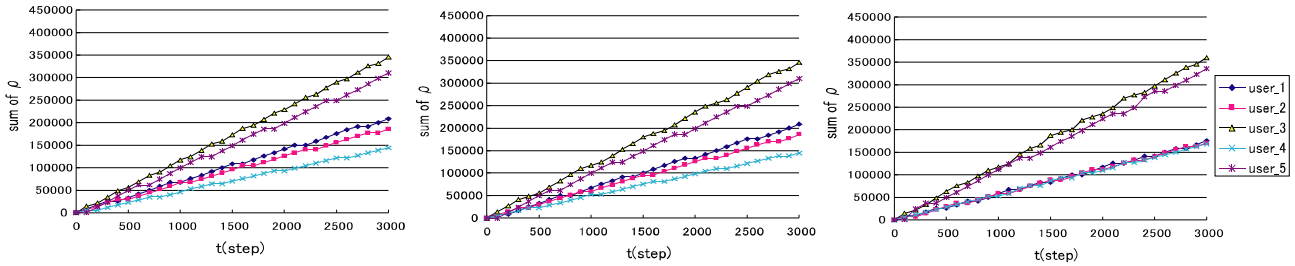
また，各リンクは限界帯域を 1000Mbps として，遅延時間は全て同一とした．それぞれのアルゴリズムにおけるシミュレーション結果を Fig.10～12 に示す．

シミュレーション結果から，提案アルゴリズム 1 では半減期が長くなると，目的ノードとの距離が長いユーザと近いユーザの占有度の差が小さくなっているのに対し，他のアルゴリズムでは差が大きいことが確認できる．これは，提案アルゴリズム 1 には目的ノードが遠い場合の不公平性を除去する機能が存在することを意味する．また各アルゴリズムにおいて占有度の履歴に差が少なく，また占有度は 2 極化していることが確認できる．これは各ユーザが node6 の接続を行うための経路が一通りしかないためであると考えられる．また提案アルゴリズム 2 の結果から，使用リンクが少なくなる User_5 のリソース占有度の遷移が高く保たれることが確認できる．



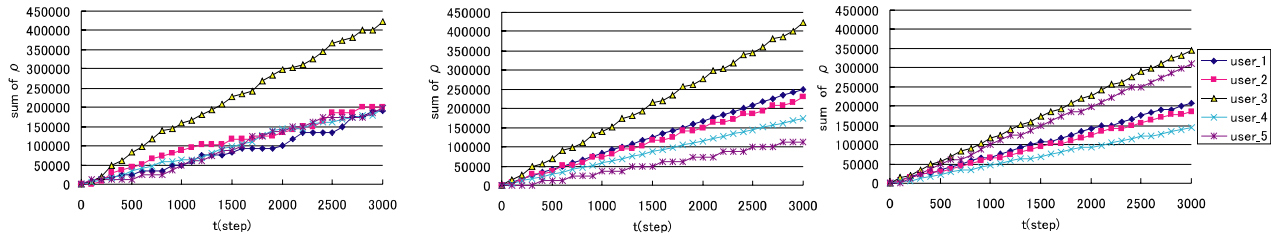
(a) 半減期10におけるρの累積　(b) 半減期100におけるρの累積　(c) 半減期1000におけるρの累積

Fig.6 Topology2 の提案アルゴリズム 1 におけるシミュレーショ結果



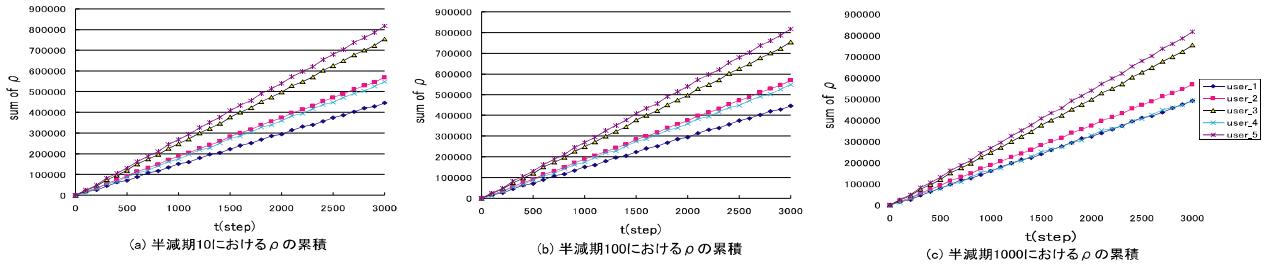
(a)半減期10におけるρの累積　(b)半減期100におけるρの累積　(c)半減期1000におけるρの累積

Fig.7 Topology2 における提案アルゴリズム 2 おけるシミュレーション結果



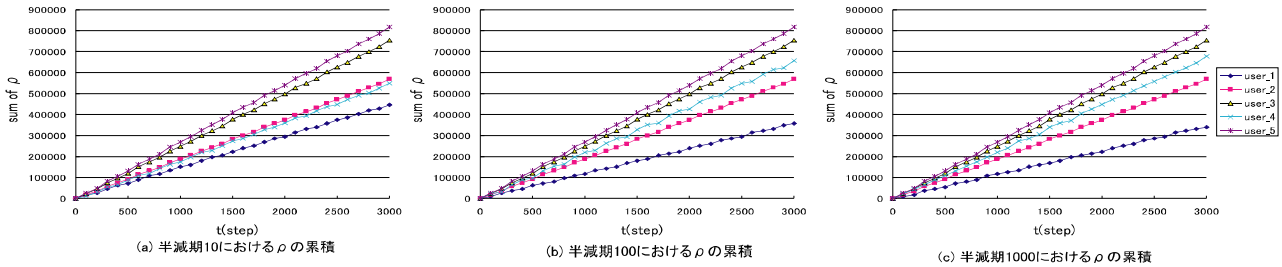
(a)randomアルゴリズムによるρ累積　(b)LRUアルゴリズムによるρの累積　(c)accepte numbeアルゴリズムによるρの累積

Fig.8　Topology2 におけるその他のアルゴリズムにおけるシミュレーション結果



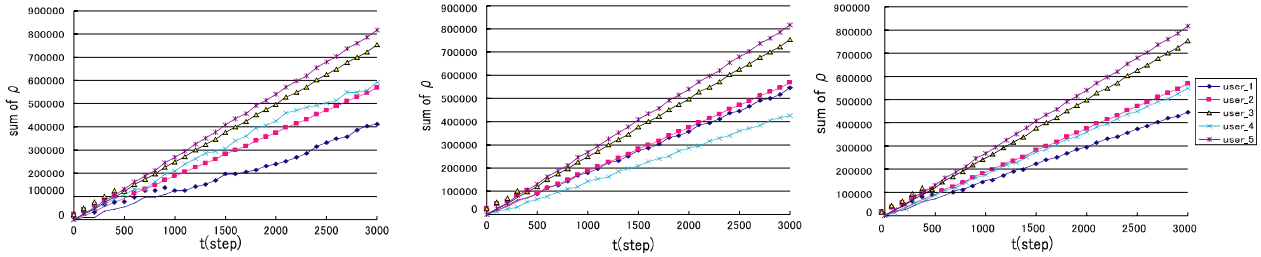
(a) 半減期10における ρ の累積　(b) 半減期100における ρ の累積　(c) 半減期1000における ρ の累積

Fig.10 Topology3 の提案アルゴリズム 1 におけるシミュレーショ結果



(a) 半減期10における ρ の累積　(b) 半減期100における ρ の累積　(c) 半減期1000における ρ の累積

Fig.11 Topology3 の提案アルゴリズム 2 におけるシミュレーショ結果



(a) randomアルゴリズムによる ρ 累積　(b) LRUアルゴリズムによる ρ の累積　(c) accepte numbeアルゴリズムによる ρ の累積

Fig.12 Topology3 におけるその他のアルゴリズムにおけるシミュレーション結果

5. おわりに

本稿では，インターネットバックボーンにおける L1 BOD のスケジューリングアルゴリズムの検討を行った．提案したスケジューリングアルゴリズムは Condor の優先度決定アルゴリズムを基にした．また，L1 BOD のシミュレータを作成し，ユーザ及びそのリクエストのテストケース，提案スケジューリングアルゴリズム及び比較用スケジューリングアルゴリズムを実装することにより，その有効性の検討を行った．

本稿では簡単なトポロジを利用してスケジューリングの検討を行った．今後，実際のネットワークに適用する場合には，次の 2 点が重要な課題となる．まず，実際のネットワークでの利用を想定した場合，複数のルーティング経路が考えられる．それぞれのリクエストに対するルーティングの割り付けによって，全体のスケジューリングの結果が異なることが予想されるため，各リクエストに対するルーティングとスケジューリングを同時に考慮し，最適化問題を解決する必要がある．次に，スケジューリングに対して地理的な影響も検討する必要がある．すなわち，ネットワーク中央部での通信を要求するリクエストを選択してしまうとネットワークの端と端を結ぶような通信を要求するジョブは採択されづらくなってしまう．このような地理的な要因を資源の利用の平等性の考慮すべき点として検討する必要があろう．今回のスケジューリングにおいては，スケジューリングを行うリクエストはキューに入った順にルーティングを割り付けることとしているが，ある一定時間内にキューに存在するリクエストをどの順番でわりつけるかについて検討する課題は，上記の課題が克服された後の発展的課題であると考えられる．

参　考　文　献

[1] SINET3: http://www.sinet.ad.jp/sinet3/
[2] レイヤ 1 帯域オンデマンドサービス: http://www.nii.ac.jp/news_jp/2008/02/post_50.shtml
[3] “Resource Allocation and Provision for Bandwidth/ Networks on Demand in SINET3,” S. Urushidani,et al., IEEE BoD 2008 Workshop, Salvador da Bahia, Apr. 2008
[4] Condor project: http://www.cs.wisc.edu/condor/
[5] University of Wisconsin: http://www.wisc.edu/