ネットワーク制御における資源共有と公平性

鶴　正人[†a)]　岩本　健志[†]

Resource Sharing and Fairness in Network Control

Masato TSURU[†a)] and Takashi IWAMOTO[†]

あらまし　要求される通信性能や同時利用数が際限なく増大していく中で，複数のユーザやアプリケーションによるネットワーク資源の効率的共有が重要である．本論文では，システム効率やユーザ間公平性の観点で「妥当な」資源共有の考え方として提案されているものを概説するとともに，例として，筆者らが扱っている，一対多ファイル転送において異なる部分を複数経路で同時転送しながらマルチキャストにより無駄な重複転送を減らす時間・空間スケジューリングを紹介し，スケジュール（解）の妥当性や妥当な解の探索方法について検討する．

キーワード　ネットワーク制御，資源共有，効用関数，公平性，複数経路マルチキャスト

1. まえがき

インターネットに代表される情報通信ネットワークは社会や経済のインフラストラクチャとして様々な活動を支えている．これまでも人によるインターネットの利用は急増を続けてきたが，今後はあらゆる物や場所がインターネットを介して通信を行う．要求される通信性能や同時利用数が際限なく増大していく中で，多数のアプリケーションやユーザを収容しつつ，それらの要求に応えるためには，少ない資源の有効利用・共有技術が重要である．

一般に，ネットワーク制御における資源共有とは，同時に発生する複数の通信間で資源の利用競合を制御/管理することであり，競合回避は基本的には，各通信が，時間，空間（経路），周波数（無線や光の）などを分けて利用することである．しかも，多数のユーザ（アプリケーション）の条件や要求は多様であり，かつ異なる種類の資源を組み合わせて利用する場合もある．よって，様々なレベル/スケールでの資源割当てが発生し，そこでは共有における「公平性」が重要な課題になる [1]〜[10]．一般に 3 種類の技術がある．

(i)　現実の複雑なネットワーク制御における資源割当て問題としてのモデル化．制御可能な「資源」の割当てと，それによって各ユーザが獲得する「性能」の関係を分析する必要がある．

(ii)　妥当な性能分配や割当ての定義．その定義に基づいた最良または最良に近い解の探索手法・アルゴリズム．特に環境・条件が動的に変化する場合の，適応的オンラインアルゴリズム．

(iii)　妥当な割当てを静的･動的に実現するネットワーク制御技術．集中形・分散形のアルゴリズムやプロトコル，その実装技術．ただし，明示的な (ii) を経ずに (iii) の制御アルゴリズム（通常は分散形の）を設計し，その制御の結果得られる割当ての妥当性を分析することも多い．

単純な資源割当ての例として，ある移動体通信網の無線基地局の配下で，2 台の無線端末（2 人のユーザ）だけが通信しており，2 人は下りのデータ通信に同じ無線周波数を使い，ユーザ 1（基地局に近い）とユーザ 2（遠い）が同時にインターネット上のサーバからファイルをダウンロードする状況を単純化したモデルを考える．性能を時間平均スループットとし，ユーザ j が周波数を独占的に使う場合のスループットを r_j $(r_1 > r_2)$，利用時間比をユーザ 1：ユーザ 2 $= p : (1-p)$ で時分割制御する．例えば，各 1 秒間の前半 p 秒をユーザ 1，後半 $(1-p)$ 秒をユーザ 2 に割当てる．

まず無限長ファイル転送の場合，p は固定で，ユーザ 1 とユーザ 2 のスループットは各々，$R_1 = r_1 p, R_2 =$

† 九州工業大学情報工学部，飯塚市
Faculty of Computer Science and Systems Engineering, Kyushu Institute of Technology, Iizuka-shi, 820–8502 Japan
a) E-mail: tsuru@cse.kyutech.ac.jp
DOI:10.14923/transcomj.2016IAI0001

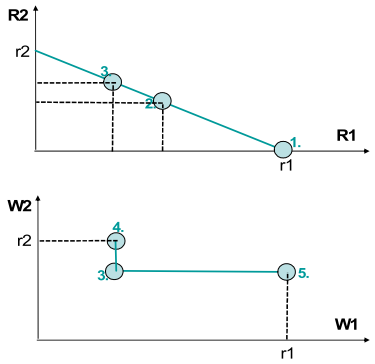

図 1　スループット 1〜無限長転送（上），有限長転送（下）
Fig. 1　Throuput-1: infinite (top)/finite (bottom) data transmission.

$r_2(1-p)$ となるので，図 1（上）において，$(r_1, 0)$ と $(0, r_2)$ 間の線分上の点が可能な性能の組 (R_1, R_2) になる．典型的な割当てを三つ挙げるが，一般的な意味付けは次節で行う．なお，“Pn” という表記は図中の点の番号 “n” に対応し，2 人のユーザへの性能分配の具体例を示す．

(P1)　スループット総和 $R_1 + R_2$ を最大にするのは，$p = 1$．しかし，$R_1 = r_1, R_2 = 0$ で極端に不公平である．

(P2)　2 人に同じ時間を割当てる（時間平等）場合，$p = \frac{1}{2}, R_1 = \frac{r_1}{2}, R_2 = \frac{r_2}{2}$ となる．

(P3)　性能を平等にするのは，$p = \frac{r_2}{r_1 + r_2}$ であり，$R_1 = R_2 = \frac{r_1 r_2}{r_1 + r_2}$ で，スループット総和は最小．

一方，同じ長さ（有限）をもつ個別のファイルの転送を考える．先に片方の受信が終われば，残ったユーザに全時間を割当てるので，p は途中で変化する．ユーザ 1 とユーザ 2 の受信完了時間を各々 T_1, T_2 とし，性能（スループット）を $W_j = 1/T_j$ で定義できる．図 1（下）において，$(\frac{r_1 r_2}{r_1 + r_2}, r_2)$, $(\frac{r_1 r_2}{r_1 + r_2}, \frac{r_1 r_2}{r_1 + r_2})$, $(r_1, \frac{r_1 r_2}{r_1 + r_2})$ の 3 点を結ぶ折れ線上の点が可能な性能の組 (W_1, W_2) になる．以下三つの例を挙げるが，“P3” は無限長転送の場合にも現れた分配である．

(P3)　性能が平等なら同時に受信が完了するので，無限長と同様に，$p = \frac{r_2}{r_1 + r_2}, W_1 = W_2 = \frac{r_1 r_2}{r_1 + r_2}$ であり，スループット総和は最小になる．

(P4)　先にユーザ 2 に全時間を割当てて（$p = 0$）受信を完了させ，その後ユーザ 1 に全時間を割当てる（$p = 1$）場合は，$W_1 = \frac{r_1 r_2}{r_1 + r_2}, W_2 = r_2$ となる．

(P5)　先にユーザ 1 に全時間を割当てて（$p = 1$）受信を完了させ，その後ユーザ 2 に全時間を割当てる（$p = 0$）場合は，$W_1 = r_1, W_2 = \frac{r_1 r_2}{r_1 + r_2}$ であり，スループット総和は最大になる．

以降，**2.** では，資源割当て問題のモデル化における割当ての「妥当性」，特にその割当てにおける各ユーザの（獲得する・分配される）性能から見た妥当性に関して，提案されている考え方を概説する．なお性能が一種類である基本の場合を扱う．**3.** では，資源割当て問題の例として，筆者らが扱っている一対多ファイル転送の時間・空間スケジューリングを紹介し，スケジュール（解）の妥当性や探索方法について検討する．

2.　性能分配における公平性と効率性

一般に，資源割当ての結果として達成される，N 人のユーザへの性能の分配は，

- $\mathbf{x} = (x_1, x_2, \ldots, x_N)$

というベクトルで表現できる．以降，$\mathbf{x}$ を分配，x_i をユーザ i の配当，と呼ぶことにする．性能（各ユーザが獲得するサービス品質）として，異種複数のものを同時に考える場合には，x_i 自体をベクトルとして扱う必要があるが，ここでは簡単のために単一性能の分配を議論の対象とし，x_i はスカラー量とする．「性能」は非負かつユーザにとっては大きい方が望ましいものとする（$x_j \geq 0$）．可能な全ての分配の集合を χ とする．

ある性能分配になるようにシステム（ネットワーク）を制御するのが「資源割当て」機構であり，その分配の良し悪しを評価する様々なアプローチがある．

- 分配によって達成されるユーザごとの効用を考え，その和の最大化を目指す．あるいは，ユーザ毎配当の「平均」を全体の幸せと考え，その最大化を目指す．
- 異なる分配間の何らかの関係性や近さを定義し，それに基づき「妥当な」分配を目指す．

また，性能分配を実現するために割当てる資源は一般には異種複数であるが，本論文では **2.3** や **3.** の例においてリンク帯域幅という単一の資源を扱う．

2.1　効用関数，ユーザ間平均

各ユーザの幸せは必ずしも自分の配当に比例するとは限らないので，分配によって達成されるユーザの幸せを効用関数によって定義する．ユーザ i の効用関数は一般には，$U_i(\mathbf{x})$ であり，他ユーザの配当も自分の

効用に影響する可能性があるが，通常よく使われるのは自分の配当だけの関数として $U_i(x_i)$ の形を取る．また，ユーザの 1 人に「システム」を入れることでシステム（プロバイダ）側の効用を考慮することもできる．効用関数の単純な例として [2],

- $U_i(\mathbf{x}) = x_i$,
- $U_i(\mathbf{x}) = \log x_i$,
- $U_i(\mathbf{x}) = u_\alpha(x_i)$ ただし,

$$u_\alpha(x) \stackrel{\text{def}}{=} \begin{cases} x^{1-\alpha}/(1-\alpha) & \alpha \neq 1 \\ \log x & \alpha = 1 \end{cases} \quad (0 \leq \alpha)$$

などがあり，3 番目は上の二つを含む（α-効用関数と呼ぶ）．そして，その総和を最大にする分配を見つける．

$$\max_{\mathbf{x} \in \chi} \sum_{i=1}^{N} U_i(x_i) \tag{1}$$

一般には解は一意ではないが，最適制御の枠組みと相性が良い．

$U_i = x_i$ の和の最大化はシステム効率の最大化とも言える．例えば，システムが x_i に比例して課金する場合の儲けを最大にする．しかし前節の例でみたように，ユーザ間のバランスを一切考慮せず極端な不公平が起きる場合がある．一方，対数効用 $U_i = \log x_i$ は,「配当が今の配当の k 倍になることによる効用の増分はユーザによらず等しい」とするモデル（$\log kx_i - \log x_i = \log k$）なので，対数和の最大化には配当の少ないユーザをある程度優先する必要がある．$\alpha = 2$（$U_i = -\dfrac{1}{x_i}$）は，$\displaystyle\sum_{i=1}^{N} \frac{1}{x_i}$ の最小化と等価である．一般に，α が大きいほど配当の少ないユーザを優先的に扱う．

一方，配当そのものがそのユーザの幸せだとしても，その幸せのユーザ全体での和ではなく「平均」を最大化する考えもある．相加平均の最大化は和の最大化と等価であるが，一般に様々な「平均」$M(\mathbf{x})$ が定義できるので，適切な $M(\mathbf{x})$ を最大にする分配 $\mathbf{x} \in \chi$ を見つけることで，ユーザ間バランスとシステム効率の両方の考慮が可能になる．これも制約付き最大化問題であり，一般には解は一意ではない．

単純な例として，相加平均，相乗平均，p-次平均：

- $M(\mathbf{x}) = \dfrac{1}{N} \displaystyle\sum_{j=1}^{N} x_i$,
- $M(\mathbf{x}) = \left(\displaystyle\prod_{j=1}^{N} x_i \right)^{1/N}$,
- $M_p(\mathbf{x}) = \begin{cases} \left(\frac{1}{N} \sum_{j=1}^{N} x_i^p \right)^{1/p} & p \neq 0 \\ \left(\prod_{j=1}^{N} x_i \right)^{1/N} & p = 0 \end{cases}$,

などがある．相乗平均の最大化は対数和の最大化と等価である．p-次平均は $p = 0$ で（右からも左からも）連続であり，p に関して単調増大で,

- $p = 1, 0, -1$ で各々，相加，相乗，調和平均
- $p \to -\infty$ で最小値，$p \to \infty$ で最大値

という意味で前二つを含む．そして，p が小さいほど，最大化において配当の少ないユーザを優先的に扱う．

「性能」の性質によっては，和ではなく積の最大化が本質的に意味をもつこともある．例えば，ユーザ i の成功確率を配当 x_i とする場合，ユーザ間の成功・失敗が独立に発生するならば，全ユーザが成功する確率：$\displaystyle\prod_i x_i$ の最大化は一つの制御目的になりうる．

2.2 分配間の関係性や近さと公平性

分配間の最も基本的な優劣関係としてパレート優勢 (Pareto dominant) $\geq_p$ がある．$\mathbf{x}, \mathbf{y} \in \chi$ として,

- $\mathbf{x} \geq_p \mathbf{y} \Leftrightarrow x_i \geq y_i, \forall i \in \{1, 2, \ldots, N\}$

パレート境界 (Pareto front) は，パレート優性の「極大元集合」として定義される．$\mathbf{x}$ がパレート境界に含まれるのは，自分よりパレート優性な分配 $\mathbf{y}$ が存在しないとき．つまり，どんな $\mathbf{y}$ $(\neq \mathbf{x})$ に対しても，$\mathbf{y} \geq_p \mathbf{x}$ が成り立たないときである．

一方，パレート優性よりも弱い特定のある関係に対して，その「最大元」によって定義される「公平な分配」が，幾つかよく知られている [11]．最大元 $\mathbf{x}^*$ が存在するかどうかは，可能な分配の全体集合 $\mathcal{X}$ に依存するが，もし存在するなら一意であり，パレート境界内にある．

代表的なものとして,

- $\mathbf{x}^*$ が Max-Min 公平

$\Leftrightarrow \forall \mathbf{y} \in \mathcal{X}, \forall j \in \{1, 2, \ldots, N\}$ において

$$y_j > x_j^* \Rightarrow \exists i (y_i < x_i^* \leq x_j^*)$$

Max-Min 公平な分配 $\mathbf{x}^*$ はユーザ間の最小値を最大化する．なぜなら，対偶を考えると，$\mathbf{x}^*$ がユーザ間の最小値を最大化しないならば，すなわち，ある分配 $\mathbf{y}$ とユーザ i が，$x_i^* < \min_j y_j$ ならば，全てのユーザ j に対して，$x_i^* < y_j$ なので，$\mathbf{x}^*$ は Max-Min 公平ではない．

- $\mathbf{x}^*$ が比例公平 (Proportional Fair)

$\Leftrightarrow \forall \mathbf{y} \in \mathcal{X}, \forall j \in \{1, 2, \ldots, N\}$ において

$$\frac{y_j}{x_j^*} \geq 1 \Rightarrow \sum_{i \neq j} \left(1 - \frac{y_i}{x_i^*}\right) \geq \frac{y_j}{x_j^*} - 1$$

つまり $\sum_j^N \frac{y_j - x_j^*}{x_j^*} \leq 0$，あるいは $\frac{1}{N}\sum_j^N \frac{y_j}{x_j^*} \leq 1$ とも書ける．

比例公平な分配 $\mathbf{x}^*$ は対数和を最大化する：$1 \geq \frac{1}{N}\sum_j^N \frac{y_j}{x_j^*} \geq \left(\prod_j^N \frac{y_j}{x_j^*}\right)^{1/N}$ より $\prod_j y_j \leq \prod_j x_j^*$．

- 正数 α に対して，$\mathbf{x}^*$ が α-比例公平

$\Leftrightarrow \forall \mathbf{y} \in \mathcal{X}, \forall j \in \{1, 2, \ldots, N\}$ において

$$\sum_j^N \frac{y_j - x_j^*}{(x_j^*)^\alpha} \leq 0 \tag{2}$$

α の値で公平性が変わる：

- $\alpha \to 0$：公平性無視（和の最大化）
- $\alpha = 1$：比例公平
- $\alpha \to \infty$：Max-Min 公平

そして，α-比例公平な分配が存在するならば，それは α-効用関数の和を最大化する [12]．

別のアプローチとして，実現不可能かも知れない「全ユーザによっての理想の分配」を考え，分配間の何らかの意味の近さ（距離）を定義し，理想分配との近さを用いて実現可能な分配を評価することが提案された [13]．理想分配は，例えば，各ユーザが「自分以外に資源を共用するユーザが居ない場合に獲得できる性能」と同じ性能を獲得する場合と考えられる．前述の α-効用関数に対応して，その関数で特徴付けられる正値有限測度に関する「一般化情報量」：

$$D_\alpha(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^N x_i f_\alpha\left(\frac{y_i}{x_i}\right)$$

によって分配間の「近さ」（ただし非対称）を定義する．f_α は，α-効用関数に対応して $(\alpha > 0)$，

$$f_\alpha(u) \stackrel{\text{def}}{=} \begin{cases} \frac{1}{\alpha(1-\alpha)}(1 - u^\alpha + \alpha(u-1)) & \alpha \neq 1, \\ u \log u - u + 1 & \alpha = 1, \end{cases}$$

と定義する．例えば，

$$D_1(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^N (y_i(\log y_i - \log x_i) - (y_i - x_i)),$$

となる．ここで，全ユーザにとっての理想分配 $\mathbf{c}$（通常は $\mathcal{X}$ の外に存在する）を与え，α-効用で特徴づけられる D_α の意味で $\mathbf{c}$ に最も近い $\mathcal{X}$ 上の点を，(α-効用に基づく）ユーザが望む公平な分配と考える．

$$\min_{\mathbf{x} \in \mathcal{X}} D_\alpha(\mathbf{x}, \mathbf{c})$$

更に，上記の，ユーザが望む分配と配分総和（システム効率）を最大化する分配の「中間」を取ることにより，トレードオフを明示的に加えることができる．具体的には，

$$\max_{\mathbf{x} \in \mathcal{X}} \left(-\frac{a}{2} D_\alpha(\mathbf{x}, \mathbf{c}) + \frac{b}{2}\sum_{i=1}^N x_i\right) \tag{3}$$

という形の最大化を考え，二つの項のある種のスケールが一致するように，$a = \alpha, b = 1$ と置く．ここで，$\alpha = 1$ の場合の目的関数は，$\sum_{i=1}^N c_i \log x_i$ と等価になる．更に，$c_1 = c_2 = \cdots = c_N$，つまり全ユーザにとっての理想分配における配当が均等である場合には，対数和の最大化と等価になり，比例公平性との対応が付く．他の α においても同様の計算で，理想分配が均等ならば，「ユーザが望む分配とシステム効率を最大化する分配の中間」は，α-効用最大化と等価で，α-比例公平性との対応が付く．

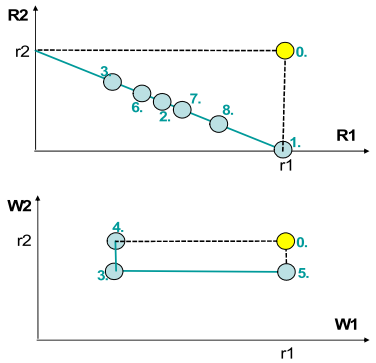

図 2　スループット 2〜無限長転送（上），有限長転送（下）
Fig. 2　Throuput-2: infinite (top)/finite (bottom) data transmission.

2.3 単純な例

1. の図 1 と同じ例で考える．図 2 には新しい分配（$P6, P7, P8$）と理想分配（$P0$）を追加した．これは，独占的に無線資源を利用できる場合の性能：$\mathbf{c} = (r_1, r_2)$ である．可能な分配においてユーザ 1 及びユーザ 2 への時間割当て比を各々 p, q と置くと，$p + q \leq 1$ であり，$q = 1 - p$ は無線資源を常に 100% 使うことに対応し，それを満たす分配が図の中の実線で描いた線分

や折れ線上の点である．

無限長転送の場合（図 2 上）は，$x_1 = R_1, x_2 = R_2$ の (x_1, x_2)-平面の第一象限内において，線分が可能な分配のパレート境界を形成する．線分上のどの点が最良かは考え方次第である．

(P1)　$p = 1$ ：総和 $x_1 + x_2$ を最大化する．

(P2)　$p = \dfrac{1}{2}$ ：対数和 $\displaystyle\sum_{i=1}^{2} \log x_i$ を最大化し，比例公平でもある．この分配 $(x_1^*, x_2^*) = (\dfrac{r_1}{2}, \dfrac{r_2}{2})$ は，パレート境界線分と双曲線 $x_1 x_2 = \text{const.}$ の接点になり，任意の可能な分配 (y_1, y_2) に対して $y_2 - x_2^* \leq -\dfrac{x_2^*}{x_1^*}(y_1 - x_1)$ が成り立つが，これは比例公平と等価．

$$\frac{y_1 - x_1^*}{x_1^*} + \frac{y_2 - x_2^*}{x_2^*} \leq 0$$

(P3)　$p = \dfrac{r_2}{r_1 + r_2}$ ：ユーザ間の最小性能を最大化し，Max-Min 公平でもある．

(P6)　$p = \dfrac{\sqrt{r_1 r_2} - r_2}{r_1 - r_2}$ ：$\dfrac{1}{R_1} + \dfrac{1}{R_2}$ を最小化（よって平均遅延時間を最小化）する．$\alpha = 2$ の α-効用や調和平均の最大化を意味する．

(P7)　$p = \dfrac{r_1 - \sqrt{r_1 r_2}}{r_1 - r_2}$ ：$\alpha = 1$ での理想との近さ $D_1(\mathbf{x}, \mathbf{c}) = \displaystyle\sum_{i=1}^{2} (r_i(\log r_i - \log x_i) - (r_i - x_i))$ を最小化する．

(P8)　$p = \dfrac{r_1}{r_1 + r_2}$ ：総和から理想との近さ $D_1(\mathbf{x}, \mathbf{c})$ を引いた値（の可変部）：$\displaystyle\sum_{i=1}^{2} r_i \log x_i$ を最大化する．

有限長転送の場合（図 2 下）は，$x_1 = W_1, x_2 = W_2$ として $P4, P5$ の点：$(\dfrac{r_1 r_2}{r_1 + r_2}, r_2)$ と $(r_1, \dfrac{r_1 r_2}{r_1 + r_2})$ のみがパレート境界であり，また折れ線上の任意の点でユーザ間の最小性能は同一（最大）である．しかし，スループット総和，対数和などを最大化し，平均遅延時間を最小化するのは，$P5$ の点 $(r_1, \dfrac{r_1 r_2}{r_1 + r_2})$ のみである．この点が比例公平になるかどうかは r_1, r_2 に依存する．いずれにせよ $P5$ が最良な分配と言えよう．

3. 事例：複数経路マルチキャストを利用した一対多ファイル転送

3.1 方式の概要

ある 1 台の送信者（送信ホスト）から同時に多数の受信者（受信ホスト）へ有限長の同一ファイルを転送することを考える（図 3）．例えば，分散したデータセンタ間でのデータやアプリケーションの分散重複配置のために必要である．一般に，受信者は接続トポロジーや帯域に関して不均一で，各受信者の受信完了時間は様々であるが，各受信者は自身の完了時間が短いことを望む．また中継ノードにはユニキャスト転送だけでなくマルチキャスト転送（指定されたパケットの多重コピー・転送）の機能を仮定する．ただし蓄積・圧縮送等の機能はもたないとする．

ここで，多数の受信者への同時並列ユニキャストは受信完了時間の面で有効ではなく送信者負荷の面でも資源割当て（スケジュール）として望ましくない．全受信者への単一木によるマルチキャストは送信者負荷を最小化するが，全受信者の受信完了時間が等しくなり，送信者との間のネットワークが良好で本来は早く受信が完了する受信者にとっては望ましくない．これらは **2.3** の有限長転送の例でいえば $P3$ に対応し，どちらも妥当とは言えない．一方，**2.3** の有限長転送の $P5$ に対応するものは，特定受信者による全資源の独占利用の逐次化，つまり，受信者ごとの（送信者からの）複数経路による最大フロー転送の逐次実行と考えられる．しかし，**2.3** の単純な問題設定とは異なり，ある受信者の完了時間を（それ以外の受信者の完了時間を増加させることなく）減少できる余地がある．例えば，現時点で最大フロー転送を行っている受信者以外の受信者へ，余っているリンク帯域を使ってマルチキャストできる場合である．更に，受信者ごとの送信者からの最大フロー転送の同時並列実行も，最大フロー経路間に重なりがあるために必ずしも有効ではない．

そこで，筆者らは，ネットワーク帯域を最大限利用し，一対多ファイル転送における各受信者の完了時間を短縮するために，ファイルを適切な大きさの均等なブロックに分割し，異なるブロックを複数経路で同時転送するとともにマルチキャストにより無駄な重複転送を減らす手法を提案し，OpenFlow 技術を用いて試作した [14], [15]．この手法は，一つのファイルを複数受信者へ複数フェーズに分けて転送し，トポロジーや帯域に応じて，ファイルのブロックへの分割，各フェーズの複数受信者への転送経路，各経路で転送するブロックの割当，という 3 種類の時間・空間割当てを決定する．前提として，この手法では，コントローラによる完全な集中制御によってトポロジーを把握し，転送を始める前に全ての受信が完了するまでの全体割当てを計算し，送信者へ転送スケジュールを指示し，

ネットワーク上のスイッチへ経路設定を指示する．

本論文では，この問題及び手法を，**2.3** よりも複雑なスケジュールの例として取り上げる．このスケジュール問題は，トポロジーや帯域に強く依存して，図 7 の SINET の例のようにノード数が多くても理想分配（各受信者が，自分以外の受信者が居ない場合に獲得できる性能と同じ性能を獲得する）を実現するスケジュールが存在し，それを容易に見つけることができる場合もあれば，ノード数が少なくても理想分配を実現するスケジュールが存在しない場合もある．特に，理想分配を実現するスケジュールが存在しない，または探索が困難な場合は，妥当な性能分配とそれを実現する妥当なスケジュールの発見方法が自明ではなく公平性を考慮する資源割当ての対象として興味深い．

以降の理論値計算（**3.3** のシミュレーション含む）においては，理想環境を仮定し，この目的のために利用できるリンク帯域は保証されていて，制御対象外の外乱（背景トラヒック等）は無視でき，パケットロスはなく，OpenFlow のパケットコピーもフルレートで行われ，指定した送信レートが維持でき，ファイル長は大きくて距離に依存した伝播遅延等も無視できる．

送信者 S，受信者 R_j $(j = 1, 2, \ldots, N)$，S から R_j への最大フロー（最大可能集約転送帯域）を W_j とする．最大フローはエンドツーエンド経路を定めないが，実際の転送では，各ブロックを S から R_j までのある経路を通る転送フローによって運ぶ必要がある．そこで，単位転送レート（帯域幅）B が存在し，各リンク帯域幅は B の倍数，かつ S から R_j への最大フローを $M_j = W_j/B$ 本の帯域幅 B の経路によって実現できるものとする．S から出発する個々の転送レート B のフローが通る経路を単位経路，R_j への単位経路の本数 M_j を R_j の「最大フロー量」と呼ぶ．$\{M_j\}$ の最小公倍数を C として，長さ L のファイルを C 個のブロックに分割する．単位経路上の 1 ブロック転送に必要な時間は $L/(BC)$ である．

あるスケジュールにおける，受信者 R_j への性能配当は受信完了時間 T_j の逆数としての時間平均スループット $1/T_j$ であり，分配は $(1/T_1, 1/T_2, \ldots, 1/T_N)$ となる．しかし，スケジュールと各受信者が獲得する性能との関係は簡易な数式表現をもたず，関数の最大化としての解析的扱いは困難である．受信者 R_j の受信完了時間 T_j の下限値（性能上限値）$U_j = L/W_j$ は，S から R_j への最大フローを使って転送する場合の受信完了時間である．$T_j^* = T_j/U_j$ を正規化受信完了時間と呼ぶ（一般に $T_j^* \geq 1$）．各 R_j に対してその下限値を実現するスケジュールは少なくとも一つ存在するが，全受信者に対して同時に下限値となる「理想分配」を実現する完璧スケジュール（全ての j に対して $T_j^* = 1$）は必ずしも存在しない．よって妥当なスケジュールを決めるには，単一の効用関数の最大化では十分ではない．

スケジュール決定手順の全体構成は以下である．

(1) R_j を「優先受信者」として，R_j が未受信の全ブロック（仮に K 個）を M_j 本の単位経路を使って転送する時間区間を（優先受信者 R_j に対応する）「フェーズ」と呼ぶ．K ブロックを M_j 本の単位経路に分散して並列転送し，$\left(\lceil \frac{K}{M_j} \rceil \times \frac{L}{BC}\right)$ 時間で終了する．R_j はこのフェーズでファイル受信が完了する．

(2) フェーズにおいて，まだ使い切ってないリンク帯域を利用して，マルチキャストにより優先受信者 R_j 以外の受信者（非優先受信者）へブロックを転送する．このマルチキャスト経路は，S から R_j への一つの単位経路上のあるノード X を分岐点として，別の受信者 R_k への単位経路のうちで X を通るものを利用して，R_k まで経路を延長することで形成される．通過する帯域に空きがある限り，各非優先受信者へ経路を分岐しブロック転送を行う．R_j への各単位経路で転送するブロックの決定と各単位経路から非優先受信者への経路分岐の決定を「ブロック割当」と呼ぶ．ブロック割当の良し悪しが，各受信者の最終的な全ブロック（= ファイル）受信完了時間に影響する．

(3) あるフェーズが完了すると，全ブロックを受信完了していない受信者の中から優先受信者を選択し，次フェーズを開始する．これを全受信者が全ブロックを受信完了するまで繰り返す．

(4) 結局，一つのスケジュールは，優先受信者（フェーズ）の順序と各フェーズ内のブロック割当てから成る．そこで，N 人の受信者の全ての順列を $\mathbf{S}$ として，$s_p \in \mathbf{S}$：優先受信者としての受信者順序，$s_n \in \mathbf{S}$：非優先受信者としての受信者順序，を与え，フェーズの順序を s_p によって定め，各フェーズ内での非優先受信者へのブロック割当を順序 s_n に従って逐次的に行い，全体スケジュールを決定する．逐次的ブロック割当では，割当て順序が回ってきた非優先受信者はリンク帯域が空いている限り，経路分岐によって受信可能なブロックを割当てる．

その際，他の非優先受信者が受信または割当済みブ

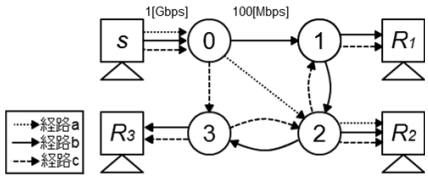

図 3　受信者 3 人で帯域が同じ場合の単位経路
Fig. 3　Unit routes over homogeneous links to three receivers.

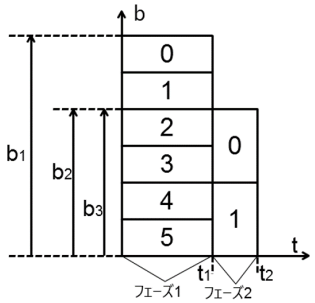

図 4　受信者 3 人の場合のブロック 0〜5 の転送スケジュール
Fig. 4　Transmission schedule of blocks (0 to 5) for three receivers.

ロックを優先的に選択する，というヒューリスティックを使う．これには次フェーズ以降で受信者間の未受信ブロックをできるだけ共通化する意図がある．

(5)　あるスケジュールで達成される性能分配を，そのスケジュールによるブロック割当・転送をシミュレートすることで得る．そして，受信者順序組 (s_p, s_n) を様々に変えて得られたスケジュールの中から，達成できる性能分配が「良い」ものを選択する．

図 3 の具体例を説明する．ホスト (S, R_1, R_2, R_3) とスイッチ (0, 1, 2, 3) 間が 1000[Mbps]，スイッチ間が 100[Mbps] で接続されており，S から R_1, R_2, R_3 への最大フローは，それぞれ $b_1 = 300$，$b_2 = 200$，$b_3 = 200$[Mbps] である．単位転送レートは 100，単位経路 a, b, c，最大フロー量（単位経路の本数）は $M_1 = 3$，$M_2 = M_3 = 2$，ブロックの分割数はそれらの最小公倍数 6 である．ここで，優先受信者順 $s_p = (R_1, R_2, R_3)$，非優先受信者順 $s_n = (R_1, R_2, R_3)$ を与えると，六つのブロックを転送するための図 4 のスケジュールが決定する．フェーズ 1 では，優先受信者 R_1 は全ブロックを受信完了し，R_2, R_3 はブロック 0, 1 が未受信となる．フェーズ 2 では，優先受信者 R_2，非優先受信者 R_3 とも，全ブロックを受信完了する．各受信者の受信完了時間は各々の下限値と等しく，完璧スケジュールになっている．また，送信者が送信する総データ量のファイル長に対する比は，(i) 各受信者への単一経路ユニキャストの同時並列の場合は 3，(ii) 全受信者への単一木マルチキャストの場合は 1 であるのに対し，本方式では $\frac{4}{3}$ となり，(i) に比べて重複転送を減らし，重複転送が全くない (ii) に近いことがわかる．

なお，このような小規模の例は，数値シミュレーションだけでなく実機による検証も行った．OpenFlow スイッチとして Linux-PC 上のソフトウェアスイッチ OVS を用い，カーネル設定はデフォルトのまま（例えば受信待ち最大パケット数 1000 個）とした．受信者 6 台までの実験を行い，ほぼ理論値通りの性能（受信完了時間）を得た．

3.2　スケジュール探索

前述のように，s_p（優先受信者としての受信者順序）と s_n（非優先受信者としての受信者順序）を一つ与え，それから得られるスケジュールをシミュレートして各受信者の理論受信完了時間を計算する．そして，受信者順序組 (s_p, s_n) を変えて得られる様々なスケジュールのうち,「良い性能分配」を実現するものを探索する．このとき，受信者数が多く探索空間が巨大な場合に，試行する受信者順序組 (s_p, s_n) の生成方法，選別方法が課題となる．ここで，性能分配そのもの（全受信者の受信完了時間のベクトル）に関するパレート境界を求めようとすると集合として大きくなる恐れがある．

そこで，探索における基本性能指標として

- 最長完了時間：最も転送に時間がかかった受信者の受信完了時間（一対多ファイル転送全体が完了するまでの時間）
- 平均完了時間：全受信者の受信完了時間の平均

の二つを用いる．試行した全ての受信者順序組の中で，(最長完了時間，平均完了時間) という性能指標空間上のパレート境界を実現するものを「良い」受信者順序組とする．大まかに言えば，それまでに試した受信者順序組の中でのパレート境界を解候補集合とし，新しい順序組を試すごとに，逐次的に集合を更新していく．**2.3** の例でも有限長ファイルの転送での最長完了時間と平均完了時間の親和性は高く，パレート境界は小さい集合になることが期待できる．

一方，受信者順序組の生成に関しては，

- s_p は，最大フロー量の降順とし，最大フロー量が同じ受信者はランダムに順序をつける．
- s_n は，s_p と同じ順序及びランダムに選択した

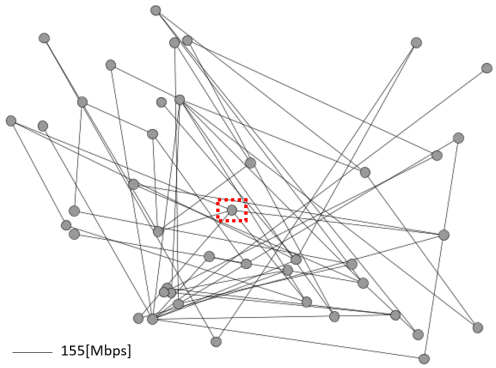

図 5　RENATER トポロジー
Fig. 5　RENATER topology.

順序.

とし，検索回数を制限する幾つかのパラメタを用意して，上述のパレート境界を逐次的に更新する．優先受信者を最大フロー量の降順にする理由は，**2.3** の例でも見られたように，早く完了できる受信者を先に完了させる方が平均完了時間の短縮につながりやすいからである．なお，ネットワーク規模が大きいと最大フロー量が等しい受信者は多数存在し，s_p も多数の選択肢から良い順序を見つける必要がある．

ここで，最長完了時間の下限値（= 各受信者の下限値の最大）は $\max_i U_i$，平均完了時間の下限値（= 各受信者の下限値の平均）は $\frac{1}{N}\sum_i U_i$ として定義されるが，もし平均完了時間が下限値に達した場合は，全受信者において受信完了時間の下限値を実現していることになり，スケジュールとして完璧なのでそこで検索は終了する．

3.3　具体例での評価

二つの大規模な実ネットワークを例としたモデルに対して，本方式の性能を数値シミュレーションで調べた結果を示す．各ノード（スイッチ）には 1 台の送信または受信ホストが直結されているものとし，スイッチと同一視する．まず，送信者 1 台（点線で囲んだノード），受信者 42 台，リンクが全て 155[Mbps] の，RENATER [16] と呼ばれる図 5 のネットワークを対象とする．ただしノードと送受信ホスト間のリンク（図には現れない）は帯域無限大とする．このとき，送信ファイル長 100[MByte] に対し，受信者ごとの受信完了時間の下限値は 4 種類（1.29, 1.72, 2.58, 5.17），最長完了時間の下限値は 5.17，平均完了時間の下限値は 2.65 である（単位は秒）．

表 1　代表的な達成性能パターン（RENATER）
Table 1　Typical patterns of aquired performance (RENATER).

項番	最長完了時間	平均完了時間	フェーズ数	順序組数
1	5.17	2.73	6	1
2	5.17	2.78	8	1
3	5.17	2.83	5,7	2
4	5.17	2.87	6〜9	9
5,6	5.17	2.95	5〜9	43

本方式では，トポロジーやその中の受信者位置に基づいて事前にスケジュールを計算（探索）し，それに従った 1 対多ファイル転送を実行する．その際，**3.2** で述べたスケジュール探索のアルゴリズムでは探索してもより良い解が見つからない状態が続くと停止するようになっており，探索の試行回数は事前には確定しない．探索過程のランダム性により実行するごとに得られるスケジュールは異なり，それによって実現される性能（各受信者の受信完了時間）が異なるので，スケジュール探索で見つかる解の特性・範囲を知り，また見つかる解の性能の安定性を調べるために，試行回数が 200 回程度になるような検索調整パラメタの設定範囲の中で，そのパラメタや乱数の種を変えて，200 回程度の検索試行でスケジュールを決定するシミュレーションを独立に 600 通り行った．

受信者 42 人中の最悪値（最長完了時間）と平均値（平均完了時間）を性能指標とし，600 回の各シミュレーションで得られた解の性能は，(最長完了時間，平均完了時間) 空間上の 103 通りの位置に存在し，それらの性能の平均値及び標準偏差値（単位は秒）は，最長完了時間の平均 5.43；標準偏差 0.343，平均完了時間の平均 2.97；標準偏差 0.089 であった．これらの値及び表 1 の結果から，最長完了時間及び平均完了時間の下限値が各々5.17 と 2.65 であることと合わせ，少なくとも RENATER トポロジーにおいては本方式は安定してよい性能を実現していると言える．また，送信者が送信する総データ量のファイル長に対する比は，受信者ごとのユニキャストの場合は 42，全受信者への単一木マルチキャストの場合は 1 であるのに対し，本方式では 3 から 4 の範囲（600 通りのシミュレーション全体）であり，マルチキャストには及ばないが重複転送を削減している．

103 通りの実現された性能の中の典型的な 5 種類を表 1 に示す．最長完了時間はどれも下限値に達しているが，平均完了時間は，2.73 が最良（1 回のみ）で，多くの場合は得られた解は 2.8 や 2.9 台であった．つ

表 2　代表的な性能分配の特性（RENATER）
Table 2　Typical characteristics of performance distribution (RENATER).

項番	$\overline{V}$	$\overline{T^*}$	$\overline{V^*}$
1	4.05	1.05	9.67
2	4.00	1.07	9.53
3	3.84	1.11	9.28
4	3.80	1.13	9.19
5	3.65	1.16	8.85
6	3.81	1.14	9.08

まり RENATER において，**3.2** の方式の 200 回程度の探索では，最長完了時間が下限値になるようなスケジュールがほぼ確実に見つかるが，よほど運が良くない限り，そのスケジュールでの平均受信完了時間は 2.8[s] を切らない．逆に表の 1 番（2.73）より更に短く，下限値 2.65（またはそれに極めて近い値）を実現するスケジュールが存在するかどうかは判らない．平均完了時間が下限値になることは，全受信者が自身の下限値 U_i で完了する理想分配と等価であり，一般には平均完了時間が下限値になるスケジュールは存在しないことがある．

5 種類の性能の各々からそれを実現するスケジュールを一つずつ（5 番目のパターンからはフェーズ数が 5 の場合と 9 の場合から一つずつ計二つ）選択し，それら六つの具体的なスケジュールの特性を調べた．(時間平均）スループット $V_j = 10/T_j$（ただし 10 倍）及び正規化完了時間 $T_j^* = T_j/U_j$，に基づく指標：

- 平均スループット $\overline{V} = \dfrac{1}{N}\sum_j \dfrac{10}{T_j}$
- 平均正規化完了時間 $\overline{T^*} = \dfrac{1}{N}\sum_j \dfrac{T_j}{U_j}$
- 平均正規化スループット $\overline{V^*} = \dfrac{1}{N}\sum_j \dfrac{10U_j}{T_j}$

を定義し，表 2 に示す．**2.** で見たように，(時間平均）スループット指標は完了時間指標よりもシステム効率を重視する．また正規化は理想分配との違い（比）を表現するが，一般に利用できる資源条件が良い（下限完了時間が短い）受信者を重視する．よって $\overline{V^*}$ の差は，システム効率に関する差をより強く反映する．5 番と 6 番を比較すると，探索に利用した基本性能指標 (最長完了時間，平均完了時間) に関して同じ値を達成するが，表 2 の指標に関しては 6 番が勝っている．逆に言えば，5 番よりも 6 番を選択するためには，探索に別の性能指標も利用する必要がある．

詳しく見るために，全受信者の正規化完了時間 T_j^*

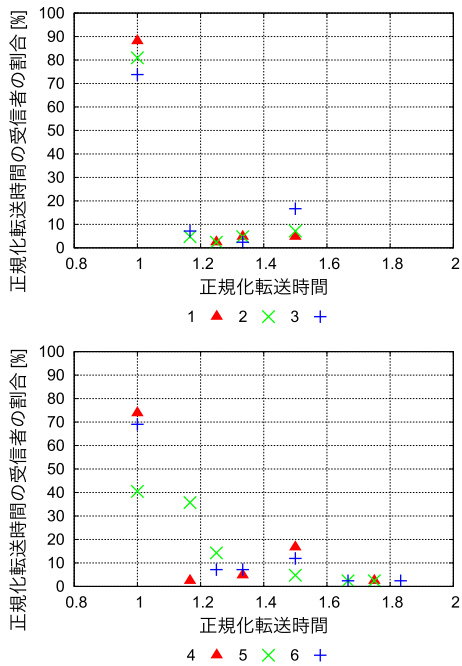

図 6　代表的な性能分配（RENATER）
Fig. 6　Typical performance distribution (RENATER).

の分布を図 6 に示す．1 番，2 番は，探索しても滅多に得られないだけあって，ほとんどの受信者が完了時間の下限値 (1) を達成して自身に利用可能なリンク帯域を使い切っている．ただし下限値の 1.5 倍の完了時間が掛かる受信者も存在する．表 2 及び図 6 から，システム効率と公平性（ただし受信者間での正規化完了時間の均等化の意味で）の両面で 1 番が最も望ましいと考えられる．3 番，4 番でも 7 割以上の受信者が完了時間の下限値を達成するが，4 番の方が受信者間のばらつきが大きい．5 番と 6 番は異なる分布を示している．5 番は完了時間の下限値を達成する受信者が半分以下であり，6 番の方がシステム効率で大幅に勝り，表 2 もそれを定量的に示しているが，図 6 からは，公平性に関しては 5 番が勝るとも言える．

次に，送信者 1 台（点線で囲んだノード），受信者 73 台，$1, 10, 40$[Gbps] の帯域が混在する，SINET [16] と呼ばれる図 7 のネットワークを対象とする．ただしノードと送受信ホスト間のリンク（図には現れない）は帯域無限大とする．このとき，送信ファイル長 1000[MByte] に対し，受信者ごとの受信完了時間の下限値は 2 種類（$0.4, 8.0$），最長完了時間の下限値は 8.0，平均完了時間の下限値は 6.9 である（単位は秒）．

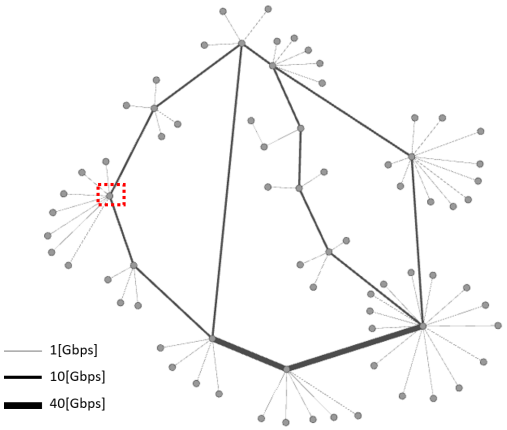

図 7　SINET トポロジー
Fig. 7　SINET topology.

試行回数が最大でも 50 回程度になるような検索調整パラメタの設定範囲で，そのパラメタや乱数の種を変えて，シミュレーションを独立に 100 通り行った所，多くの場合に平均完了時間が下限値を実現する完璧スケジュールが見つかった．このとき，送信者が送信する総データ量のファイル長に対する比は，受信者ごとのユニキャストの場合は 73，全受信者への単一木マルチキャストの場合は 1 であるのに対し，本方式では 1.9 から 2.5 の範囲（100 通りのシミュレーション全体）であり，重複転送を削減している．

完璧スケジュールが存在し，かつ容易に見つかる理由は，トポロジーの特徴にあると考えられる．図 7 からわかるように，10G と 40G[bps] のリンクによる基幹ループ上にある送信ノードが 10G[bps] リンクで挟まれているため，基幹ループ上のスイッチ (11 個) に直収される受信者（拠点受信者）の最大フローは全て 20G[bps]，基幹ループから 1G[bps] で延びた先のスイッチ (62 個) に直収される受信者（末端受信者）の最大フローは全て 1G[bps] で，ファイルは 20 個のブロックに分割される．第 1 フェーズで全拠点受信者へ全 20 個のブロックを転送できるスケジュールは容易に見つかり，この間に各末端受信者が（マルチキャストによって）受信する 1 個のブロックがもし共通になっていれば，共通に必要な残り 19 個のブロックを，第 2 フェーズで全末端受信者への単一木マルチキャストを用いて転送でき，全受信者が各自の下限値で受信完了する．つまり，SINET はどんな受信者の順序で探索を行っても最良な解に行き着きやすいトポロジーと言える．

4. むすび

本論文では，ネットワーク制御における資源共有に関して，システム効率やユーザ間公平性の観点からの資源割当て（解）の妥当性や妥当な解の探索方法について議論した．古くから知られた問題ではあるが，資源の種類や制御の方法等の前提条件が変わると新しい問題が提起される．

複雑なスケジュールの例として 1 対多ファイル転送を取り上げ，OpenFlow 技術を想定した，単一受信者への複数経路上の最大フロー転送と複数受信者へのマルチキャストとを組み合わせた手法を紹介した．スケジュール探索として，ある解（スケジュール）での 1 対多ファイル転送における全受信者中の最悪値（最長完了時間）と平均値（平均完了時間）をその解の 2 次元の性能指標とし，ある種のランダム検索によって性能指標上のパレート境界を探す方法を採用し，大規模ネットワークの例を用いて，得られた解の性質を調べた．

向上させたい量として質の異なるものを複数個考える場合は「多目的最適化」として活発に研究されている．そこでは，スカラー量の最大化に基づいて解を探す **2.1** のアプローチは必ずしもうまくいかない．もしそれらの量に対応した個別の効用関数を定義して何らかのスケーリングによって加算可能にできるならば，複数種の効用関数の和を統合した効用関数とみなすことができる．しかしうまくモデル化できないことも多く，妥当な解の候補集合を定義したり見つけたりするために，**2.2** で述べた関係や近さを用いたアプローチが有用になる．

謝辞　本研究は JSPS 科研費 25330108 の助成を受けていた．また，内田真人教授（千葉工業大学）及び Mario Koeppen 教授（九州工業大学）のご助言に感謝する．

文　献

[1] L. Georgiadis, M. Neely, et al., “Resource allocation and cross-layer control in wireless networks,” Foundations and Trends in Networking, vol.1, no.1, pp.1–144, 2006.
[2] S. Shakkottai and R. Srikant, “Network optimization and control,” Foundations and Trends in Networking, vol.2, no.3, pp.271–379, 2007.
[3] B. Radunovic and J.-Y. Boudec, “A unified framework for max-min and min-max fairness with applications,” IEEE/ACM Trans. Networking, vol.15, no.5, pp.1073–1083, 2007.

[4] T. Lan, D. Kao, et al., “An axiomatic theory of fairness in network resource allocation,” Proc. IEEE INFOCOM’10, pp.1–9, 2010.
[5] C. Wong, S. Sen, et al., “Multi-resource allocation: Fairness-efficiency tradeoffs in a unifying framework,” Proc. IEEE INFOCOM’12, pp.1206–1214, 2012.
[6] E. Danna, “Upward max min fairness,” Proc. IEEE INFOCOM’12, pp.837–845, 2012.
[7] W. Ogryczak, H. Luss, et al., “Fair optimization and networks: A survey,” J. Applied Mathematics, ID 612018, 2014.
[8] T. Bonald and J. Roberts, “Multi-resource fairness: Objectives, algorithms and performance,” Proc. ACM SIGMETRICS’15, pp.31–42, 2015.
[9] J. Guo, F. Liu, et al., “Fair network bandwidth allocation in IaaS datacenters via a cooperative game approach,” IEEE/ACM Trans. Networking, vol.24, no.2, pp.873–886, 2016.
[10] X. Chen, H. Cai, et al., “Multi-criteria routing in networks with path choices,” Proc. IEEE ICNP’15, pp.334–344, 2015.
[11] M. Koeppen, “Relational optimization and its application: From bottleneck flow control to wireless channel allocation,” INFORMATICA, vol.24, no.3, pp.413–433, 2013.
[12] J. Mo and J. Walrand, “Fair end-to-end window-based congestion control,” IEEE/ACM Trans. Networking, vol.8, no.5, pp.556–567, 2000.
[13] M. Uchida and J. Kurose, “An information-theoretic characterization of weighted alpha-proportional fairness,” Proc. IEEE INFOCOM’09, pp.1053–1061, 2009.
[14] A. Nagata, Y. Tsukiji, and M. Tsuru, “Delivering a file by multipath-multicast on openflow networks,” Proc. IEEE INCoS’13, pp.835–840, 2013.
[15] 岩本健志，岡本洋平，鶴　正人，“OpenFlow による複数経路マルチキャストを利用した一対多ファイル転送の分析と改良,” 信学技報，NS2015-167, 2015.
[16] The Internet Topology Zoo, http://www.topology-zoo.org/ (2016-02-28 accessed)

（平成 28 年 3 月 10 日受付，6 月 1 日再受付，
7 月 1 日早期公開）

鶴　　正人　（正員）

1985 京大院・数理工学専攻修了．沖電気工業，長崎大学総合情報処理センタ助手，通信・放送機構研究員等を経て，2006 年より九州工業大学情報工学部教授．博士（情報工学）．情報通信ネットワークの計測，制御，管理に関する研究に従事．IEEE，ACM，電子情報通信学会，情報処理学会，ソフトウェア科学会各会員．

岩本　健志

2015 九州工業大学情報工学部電子情報工学科卒．現在同大学院在学中．OpenFlow を用いた多地点間ファイル転送の研究に従事．