フィルタリング関数におけるセレクションとランキングについて

澤井 里枝† 塚本 昌彦†
寺田 努†† 西尾 章治郎†

近年，さまざまな放送型サービスの普及による受信データの増加にともなって，フィルタリング技術に対する要求が急速に高まっている．この要求に対し，これまで多数のフィルタリング手法が提案されてきた．しかし，数学的な基盤がないために，定性的な評価や最適化，宣言的なフィルタリング言語の設計が十分できなかった．そこで本研究では，一般のフィルタリングで特に頻繁に用いられるセレクションとランキングという 2 つのフィルタリング手法に焦点を絞り，それらの手法に関する詳細な数学的基盤を構築することを目的とする．本研究では，これらの 2 つのフィルタリング手法をそれぞれ独自の関数で表現し，その性質を明らかにする．また，2 つの関数の合成関数の性質を示すことで，セレクションとランキングを組み合わせて用いた場合の特性を明らかにする．

On Selection and Ranking in Filtering Function

Rie Sawai,† Masahiko Tsukamoto,† Tsutomu Terada††
and Shojiro Nishio†

In recent years, due to the popularization of various broadcast services, the volume and variety of broadcast data to be received by users are rapidly increasing. In this environment, there is a strong demand for filtering techniques. However, since there is no mathematical foundation of these filtering methods, it is not possible to qualitatively evaluate various filtering methods, to optimize processing methods in filtering, and to design a declarative description language of the filtering policy. In this paper, focusing on selection and ranking, which are the most popular filtering methods used in general filtering, we present a detailed mathematical foundation of filtering using these two methods. First, we define these two filtering methods as functions, and clarify their properties. Second, we reveal the properties of composite function of these two functions, and characteristics of filtering using combination of selection and ranking methods.

1. はじめに

近年，新たな衛星の打ち上げや地上波放送のデジタル化により，多数の放送型サービスが提供されるようになった[8),9)]．このような環境では，放送されるデータ量が膨大になり，内容も広範囲にわたるため，その中からユーザが必要とする情報を探し出すことは非常にコストの高い作業である．そこで，自動的に受信データの取捨選択を行うため，さまざまなフィルタリング機構や，フィルタリングのためのユーザ要求記述言語が提案されている[2),3),7),10)]．しかし，各フィルタリング機構は，キーワードマッチングや関連フィードバックなど，それぞれ独自の手法によってデータのフィルタリングを行っているにもかかわらず，それらの手法を定性的に表現する数学的基盤がなかった．そのため，フィルタリングの性質の定性的な評価や処理手法の最適化，宣言的なフィルタリング言語の設計などができなかった．そこで，筆者らはこれまでにフィルタリングを関数として表す**フィルタリング関数**を定義し，フィルタリングの性質をフィルタリング関数が満たす制約条件によって定性的に表現することを可能にした[11),12)]．さらに，フィルタリング関数の各制約条件の相互関係を示すことで，さまざまなフィルタリング手法の性質間の関係を明らかにしてきた．

筆者らのこれまでの研究では，すべての受信データを一度に処理する方法と複数の受信機で並列に処理する方法で結果の等価性が保たれる，といったような処理方法に関する性質を定義してきた．しかし，あるフィルタリング手法が与えられたとき，その手法がどの性

† 大阪大学大学院情報科学研究科
Graduate School of Information Science and Technology, Osaka University
†† 大阪大学サイバーメディアセンター
Cybermedia Center, Osaka University

質を満たすかを調べる方法は明らかでなかったため，手法の処理手続きを詳細に調べたり，性質を表す制約条件とフィルタリング結果を比較したりするなど，複雑な方法をとる必要があった．また，これまではフィルタリングを関数として表現する際に，具体的なフィルタリング手法の性質を考慮していなかったため，一般に用いられているフィルタリングの詳細な特性を明らかにできなかった．そこで本研究では，フィルタリングの代表的な手法である**セレクション**と**ランキング**の 2 つの手法に焦点を絞り，各手法を独自の関数で定義することで，その詳細な特性を明らかにする．

セレクションを用いたフィルタリングとは，各データの取捨選択が潜在的に決まっている手法である．セレクションによるフィルタリングには，たとえば，データが持つキーワードとユーザの嗜好を表すキーワードとの論理演算を行うキーワードマッチング，データに有効期限を設定し，その有効期限と現在時刻との論理演算を行う手法，データの内容から評価値を計算し，評価値とユーザが設定した閾値との論理演算を行う手法などがある．一方，ランキングによるフィルタリングとは，ユーザの嗜好に応じて受信データを重要な順序に並べ，その上位から特定の数のデータを選択する手法である．これらの手法は，一般のフィルタリングにおいて基本となる手法である．本研究では，これまで定義したフィルタリング関数に制約条件を付加することで，セレクションとランキングを独自の関数として定義し，その詳細な特性を明らかにする．さらに本研究では，これら 2 つの関数の合成関数が持つ性質を明らかにする．

本研究で構築する体系を用いることで，セレクションとランキングの手法を組み合わせたフィルタリングはどのような特性を持つかが判断でき，環境に応じてより効率的な処理方法への変換が可能となる．また，セレクションとランキングを用いたフィルタリングの性質を定性的に評価できる．

以下，2 章でフィルタリング関数の概要を述べる．3 章でセレクションとランキングによるフィルタリング手法を定式化し，その性質を明らかにする．定式化した 2 つの手法の関係を 4 章で示し，5 章でそれらの関数の合成関数について述べる．6 章では，本稿で明らかになった結果をもとに，実際のフィルタリングシステムや関連研究を考察する．最後に 7 章でまとめを行う．

2. フィルタリング関数

本章では，本研究の基礎となるフィルタリング関数について述べる．

2.1 フィルタリング処理の分類

あるフィルタリング手法が与えられたとき，実際の処理方法は以下に示すいくつかのパターンに分類できる．

データアイテムを受信するたびに受信データと前回までのフィルタリング結果を合わせてフィルタリングする処理方法を**逐次処理**と呼ぶ．それに対し，放送データを受信側にある程度ためておいてから一括してフィルタリングする処理方法を**一括処理**と呼ぶ．また，データ集合を 2 つ以上の任意の集合に分割して各々フィルタリングし，結果をマージしたものをフィルタリング結果とする処理方法を**分配処理**と呼ぶ．さらに，分配処理の結果を再びフィルタリングする処理方法を**並列処理**と呼ぶ．

2.2 フィルタリング関数の性質

データアイテムの集合を $\mathbf{T}$ とする．フィルタリング関数とは，任意の $T \subset \mathbf{T}$ に対し☆，以下の 2 つの条件を満たす $2^{\mathbf{T}}$ 上の関数 f のことをいう[11),12)]．

減少性（D: Decreasing）

$$f(T) \subset T$$

ベキ等性（ID: Idempotent）

$$f(f(T)) = f(T)$$

また，フィルタリング関数について以下のような性質が定義されている．

単調性（M: Monotone）

$$S \subset T \text{ ならば } \quad f(S) \subset f(T)$$

逐次増加性（SI: Sequential Increasing）

$$f(S \cup T) \subset f(S \cup f(T))$$

逐次減少性（SD: Sequential Decreasing）

$$f(S \cup T) \supset f(S \cup f(T))$$

分配増加性（DI: Distributed Increasing）

$$f(S \cup T) \subset f(S) \cup f(T)$$

分配減少性（DD: Distributed Decreasing）

$$f(S \cup T) \supset f(S) \cup f(T)$$

並列増加性（PI: Parallel Increasing）

$$f(S \cup T) \subset f(f(S) \cup f(T))$$

並列減少性（PD: Parallel Decreasing）

$$f(S \cup T) \supset f(f(S) \cup f(T))$$

一貫性（C: Consistency）

$$f(S) \supset f(S \cup T) \cap S$$

ここで，S，T は $\mathbf{T}$ の任意の部分集合とする．

これまでに筆者らは，これらの性質間に**図 1** に示す

☆ 本稿では $A \subset B$ は A が B の部分集合である（A = B の場合を含む）ことを意味するものとする．

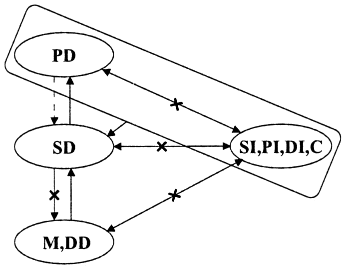

図 1 性質間の関係
Fig. 1 The relationship between the properties of the filtering function.

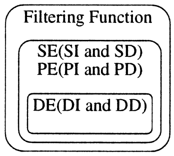

図 2 等価性間の関係
Fig. 2 The relationship between the equivalence properties.

ような相互関係があることを明らかにした．図 1 の矢印は包含関係を表し，包含関係が必ずしも成り立たないものには "×" を付す．たとえば，"M，DD"–"SD" 間の矢印は，単調性 M（または M と同値である分配減少性 DD）を満たすフィルタリング関数は逐次減少性 SD も満たすが，逐次減少性 SD を満たすフィルタリング関数は単調性 M（および分配減少性 DD）を必ずしも満たさないことを表す．また，"M，DD" のように，1 つの楕円内に列記した性質は同値であることを示す．さらに，異なる性質を囲った角丸四角の枠は枠内の性質をすべて満たす性質を表し，並列減少性 PD かつ逐次増加性 SI を満たすフィルタリング関数は逐次減少性 SD も満たすことを表す．ただし，並列減少性 PD を満たすフィルタリング関数が逐次減少性 SD を満たすかどうかは現在のところまだ明らかとなっていないため点線で示している．

ここで，逐次増加性かつ逐次減少性を満たす性質を**逐次等価性**（SE: Sequential Equivalence）と呼び，逐次等価性を満たすフィルタリングは一括処理と逐次処理の結果が等価となることを意味する．同様に，分配増加性かつ分配減少性を満たす性質を**分配等価性**（DE: Distributed Equivalence）と呼び，分配等価性を満たすフィルタリングは一括処理と分配処理の結果が等価となることを意味する．さらに，並列増加性かつ並列減少性を満たす性質を**並列等価性**（PE: Parallel Equivalence）と呼び，並列等価性を満たすフィルタリングは一括処理と並列処理の結果が等価となることを意味する．

図 1 に示す性質間の関係から，これらの等価性間の関係は**図 2** に示すような関係となる．図 2 はフィルタリング関数の包含関係を表し，分配等価性 DE を満たすフィルタリング関数は逐次等価性 SE（および SE と等価な並列等価性 PE）を満たすが，その逆は必ずしも成り立たないことを示す．図 2 より，一括処理と分配処理の結果が等価であるフィルタリングは，逐次処理や並列処理の結果とも等価となることが分かる．また，一括処理と逐次処理の結果が等価であるフィルタリングは，並列処理の結果とも等価となり，一括処理と並列処理の結果が等価であるフィルタリングは，逐次処理の結果とも等価となる．図 1，図 2 に示す性質間の関係より，ある性質を満たすフィルタリングが他の性質を満たすかどうかが判断でき，環境に応じてより効率的な処理方法に変換できる．

3. セレクション関数とランキング関数

前章で示したように，これまでは，フィルタリングの基本的な性質を関数が満たす制約条件として定義してきた．本章では，実際によく使われているフィルタリングの特性を明らかにするために，セレクションとランキングについて論じる．

セレクションとは，各データの取捨選択が潜在的に決まっている手法である．セレクションによるフィルタリングに，たとえば，キーワードマッチング，有効期限を設定するフィルタリング，閾値を用いたフィルタリングなどがある．キーワードマッチングでは，データが含むキーワードとユーザの嗜好を表したキーワードとの論理演算を行い，特定のキーワードを含むデータアイテムを蓄積し，含まなければ蓄積しない．データに有効期限を設定するフィルタリングでは，その有効期限と現在時刻との論理演算を行い，有効期限を過ぎていなければ蓄積する．閾値を用いたフィルタリングでは，各データにその内容から評価値を与え，評価値が閾値よりも大きい，あるいは小さい場合に蓄積する．閾値を用いたフィルタリングとして，たとえば距離の公理を用いたフィルタリング[6)]では，データを内容に応じて距離空間に置き，ユーザの嗜好を表した点からの距離と閾値との論理演算を行う．そして，閾値よりもユーザの嗜好を表した点に近いデータを蓄積する．セレクションは，簡単な論理演算によって必要なデータを絞り込むことができるので，フィルタリングにおいて最も多用される手法である．

ランキングとは，ユーザの嗜好に応じて受信データを重要な順序に並べ，その上位から特定の数のデータ

を選択する手法である．ランキングは，蓄積するデータ数を定めてフィルタリングする場合や，ユーザに対して，重要なデータを必要な順序に並べて提示する場合などに，頻繁に用いられる手法である．

以上のように，一般のフィルタリングでは，セレクションやランキングの手法を基本として処理を行っているものが多い．したがって，本研究では，セレクションとランキングの手法を定式化し，その性質を明らかにすることを目的とする．

3.1　セレクション関数

本節では，セレクションを定式化する．

ある $X \subset \mathbf{T}$ に対する X の**セレクション関数**（selection function）B_X とは，任意の $S \subset \mathbf{T}$ に対して

$$B_X(S) \triangleq S \cap X$$

と定義される関数のことをいう．X をこのセレクション関数の**潜在集合**（potential set）と呼ぶ．B_X がフィルタリング関数であることは容易に確かめられる．セレクション関数は

$$X = B_X(\mathbf{T})$$

を満たす．

また，関数 f が**強一貫性**（the strong consistency property）を持つとは，任意の $S, T \subset \mathbf{T}$ に対して

$$f(S) = f(S \cup T) \cap S$$

を満たすことをいう．これは，一貫性（C）の十分条件になっている．ここで，以下の 2 つの定理を示す．

定理 1　フィルタリング関数 f がセレクション関数であることと f が強一貫性を持つことは同値である．

《証明》

$$\exists X, \forall S, f(S) = S \cap X$$
$$\iff \forall S, T, f(S) = f(S \cup T) \cap S$$

が成立することを証明する．

i)　$(\Longrightarrow)$ $f(S) = S \cap X$ より

$$\begin{aligned} f(S \cup T) \cap S &= ((S \cup T) \cap X) \cap S \\ &= ((S \cap X) \cup (T \cap X)) \cap S \\ &= (S \cap X \cap S) \cup (T \cap X \cap S) \\ &= (S \cap X) \cup (T \cap X \cap S) \\ &= S \cap X \\ &\qquad (\because S \cap X \supset T \cap X \cap S) \\ &= f(S). \end{aligned}$$

ii)　$(\Longleftarrow)$ $f(S) = f(S \cup T) \cap S$ において $T = \mathbf{T}$ とおくと，任意の S に対し

$$\begin{aligned} f(S) &= f(S \cup \mathbf{T}) \cap S \\ &= f(\mathbf{T}) \cap S \end{aligned}$$

が成立する．この式で $X = f(\mathbf{T})$ とおけばよい．□

定理 2　フィルタリング関数 f がセレクション関数であることと，f が分配等価性を満たすことは同値である．

《証明》

$$\exists X, \forall S, f(S) = S \cap X$$
$$\iff \forall S, T, f(S \cup T) = f(S) \cup f(T)$$

が成立することを証明する．

i)　$(\Longrightarrow)$ $f(S) = S \cap X$ より

$$\begin{aligned} f(S \cup T) &= (S \cup T) \cap X \\ &= (S \cap X) \cup (T \cap X) \\ &= f(S) \cup f(T). \end{aligned}$$

ii)　$(\Longleftarrow)$ 任意の $S \subset \mathbf{T}$ に対して

$$\mathbf{T} \supset S$$
$$f(\mathbf{T}) \supset f(S)$$
$$(\because \mathrm{DD} \Leftrightarrow \mathrm{M} \text{ より，} f \text{ は M を満たす})$$
$$f(\mathbf{T}) \cap S \supset f(S) \cap S$$
$$f(\mathbf{T}) \cap S \supset f(S) \quad (\because \mathrm{D})$$

が成立する．また，

$$f(S) \supset f(S \cup \mathbf{T}) \cap S$$
$$(\because \mathrm{DI} \Leftrightarrow \mathrm{C} \text{ より，} f \text{ は C を満たす})$$
$$\supset f(\mathbf{T}) \cap S$$

が成立する．これら 2 式より，任意の S に対して $f(S) = S \cap f(\mathbf{T})$ となる．この式で $X = f(\mathbf{T})$ とおけばよい．□

定理 1，2 より，強一貫性，または分配等価性を満たすフィルタリングは，セレクションのみを用いた手法と見なすことができ，セレクションによるフィルタリングと同様の特性を持つことが分かる．したがって，フィルタリング結果を式に適用することで強一貫性や分配等価性を満たすことは明らかとなっているが，処理手法が明確でなかったフィルタリングについて，そのフィルタリングのポリシーを簡単な論理演算によって表現できることが明らかとなった．また，図 2 より，セレクションによるフィルタリングは，分配等価性 DE，逐次等価性 SE，並列等価性 PE を満たす．ゆえに，一括処理，分配処理，逐次処理，並列処理すべての結果が等価となり，環境に応じてより効率的な処理方法へと等価変換できる．

3.2　ランキング関数

次に，ランキングの手法を定式化する．

全順序 $R = (\mathbf{T}, <)$ における，$a \in \mathbf{T}$ の**下界**（lowerbound of a）R_a とは，$\{x \in \mathbf{T} | x < a\}$ のことをいう．$2^{\mathbf{T}}$ 上の関数 f が**度数**（cardinality）n（$\in \mathbf{N}$; 0 以上の整数全体からなる集合）であるとは，任意の $X \subset \mathbf{T}$ に対して

$$\begin{cases} |f(X)| = n \ (X \text{ が無限集合，あるいは} \\ \qquad\qquad X \text{ が有限集合であり } |X| \geq n \text{ のとき}) \\ f(X) = X \ (X \text{ が有限集合であり } |X| < n \text{ のとき}) \end{cases}$$

が成立することとする．また，度数 n のフィルタリング関数を**度数関数**（cardinality function）と呼ぶ．ここで，全順序 R が与えられているものとする．関数 f が任意の $X \subset \mathbf{T}$ に対し，ある a について $f(X) = R_a \cap X$ と表せるとき，f は**閉じている**（closed）といい，閉じた関数を**閉関数**（closed function）と呼ぶ．閉関数について，次の補題が成立する．

補題 1　全順序が与えられているとき度数 n の閉関数は一意である．□

ある全順序 R に対する n **ランキング関数**（n-ranking function）とは，R に対する度数 n の閉関数とする．ランキング関数がフィルタリング関数であることは容易に確かめられる．R に対する n ランキング関数は，ユーザの嗜好に応じて受信データを重要な順序に並べ，その上位から n 個のデータを選択する手法を表す．実際のフィルタリングでは，ユーザの嗜好やデータの内容を明確に表現できないため[2)]，必ずしも全順序を決定できるとは限らない．本研究では，評価の同じデータが複数存在するなどの理由により順序関係が明確でないデータに対して，任意の順序を定義し，全順序をあらかじめ与えておくものとする．ここで，以下の定理を示す．

定理 3　フィルタリング関数 f がある全順序（$\mathbf{T}, <$）に対する n ランキング関数であることと，f が度数 n で逐次等価性を満たすことは同値である．□

定理 3 より，逐次等価性を満たし，かつ度数が定まるフィルタリングは，ランキングを用いた手法と見なすことができ，ランキングによるフィルタリングと同様の特性を持つことが分かる．したがって，フィルタリング結果を式に適用することで逐次等価性を満たすことは明らかとなっているが，処理手法が明確でなかったフィルタリングは，度数が定まれば，その手法をある全順序に対するランキング関数によって表現できることが明らかとなった．また，図 2 より，ランキングによるフィルタリングは，逐次等価性 SE，並列等価性 PE を満たすため，一括処理，逐次処理，並列処理の結果が等価となるが，分配等価性 DE を満たさないため，分配処理との結果は必ずしも等価とならない．ゆえに，環境に応じて分配処理以外の処理方法へと等価変換できる．

4. セレクション関数とランキング関数の関係

前章までに，セレクション関数は分配等価性を満たし，ランキング関数は逐次等価性を満たすことを明らかにした．また，2 章で，逐次等価性は分配等価性を包含することを述べた．本章では，以下の補題を用いて，セレクション関数とランキング関数の関係を明らかにする．

補題 2　ϕ のセレクション関数と 0 ランキング関数は同値であり，$\mathbf{T}$ のセレクション関数と $|\mathbf{T}|$ ランキング関数は同値である．

《証明》ϕ のセレクション関数を S_ϕ，0 ランキング関数を R_0 とすると，任意の $T \subset \mathbf{T}$ に対して

$$S_\phi(T) = R_0(T) = \phi$$

となる．また，$\mathbf{T}$ のセレクション関数を $S_{\mathbf{T}}$，$|\mathbf{T}|$ ランキング関数を $R_{|\mathbf{T}|}$ とすると，任意の $T \subset \mathbf{T}$ に対して

$$S_{\mathbf{T}}(T) = R_{|\mathbf{T}|}(T) = T$$

となる．□

補題 3　$n \neq 0, |\mathbf{T}|$ のとき，n ランキング関数はセレクション関数でない．

《証明》次の補題を示す．

補題 4　関数 f が度数 n $(0 < n < |\mathbf{T}|)$ の度数関数ならば，f は必ずしも単調性を満たさない．

《証明》f が単調性を満たすと仮定する．$0 < n < |\mathbf{T}|$ なので

$$|f(\mathbf{T})| = n$$

$$0 < |\mathbf{T} - f(\mathbf{T})| = |\mathbf{T}| - n < |\mathbf{T}|$$

である．したがって

$$0 < |f(\mathbf{T} - f(\mathbf{T}))| \leq n$$

となる．ここで，任意の $x \in f(\mathbf{T} - f(\mathbf{T}))$ に対して

$$x \in f(\mathbf{T} - f(\mathbf{T})) \subset \mathbf{T} - f(\mathbf{T})$$

なので

$$x \notin f(\mathbf{T}).$$

したがって

$$f(\mathbf{T} - f(\mathbf{T})) \not\subset f(\mathbf{T})$$

が成立する．一方，

$$\mathbf{T} - f(\mathbf{T}) \subset \mathbf{T}$$

であるので，f が単調性を満たすことに反する．□

補題 4 より，フィルタリング関数 f が度数 n $(n \neq 0, |\mathbf{T}|)$ のとき，f は必ずしも単調性を満たさないこ

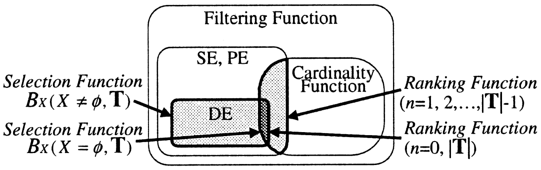

図 3　セレクション関数とランキング関数の関係
Fig. 3　The relationship between the selection function and the ranking function.

とが分かる．また，定理 2 より，セレクション関数は単調性を満たすことが分かっているので，f はセレクション関数でない．　□

補題 5　$X \neq \phi, \mathbf{T}$ のとき，X のセレクション関数はランキング関数でない．

《証明》$X \neq \phi, \mathbf{T}$ のとき，X のセレクション関数 S_X は，ある $T \subset \mathbf{T}$ に対して $|S_X(T)| \neq 0$，任意の $T \subset \mathbf{T}$ に対して $|S_X(T)| \neq |\mathbf{T}|$ となる．

ここで，$S_X(T)$ を度数 n $(0 < n < |\mathbf{T}|)$ のセレクション関数とすると，補題 4 より，S_X が単調性を満たすことに矛盾する．したがって，S_X は度数関数でない．すなわち，S_X はランキング関数でない．　□

本章で示した補題より，セレクション関数，ランキング関数，度数関数，等価性を満たすフィルタリング関数の関係を**図 3** に示す．補題 2，補題 3，補題 5 より，セレクション関数かつランキング関数であるフィルタリング関数は，どのようなデータ集合をフィルタリングしても，空集合をフィルタリング結果とするフィルタリング関数と，受信データをすべてフィルタリング結果とするフィルタリング関数のみであることが明らかになった．また，度数関数が分配等価性を満たすことと度数が 0 あるいは $|\mathbf{T}|$ であることが同値であることも明らかになった．

5. 合成関数の性質

一般に，実際のフィルタリングシステムは，複数の手法を組み合わせて用いている．そこで本章では，前章までに述べたセレクション関数とランキング関数の合成について考え，合成関数の性質を明らかにする．ここで，セレクション関数とランキング関数の合成関数がフィルタリング関数であることは容易に確かめられる．

合成関数について以下の補題が成立する．

補題 6　関数 f，g がセレクション関数ならば，$f \circ g$ はセレクション関数である．

《証明》$f(S) = S \cap X$，$g(S) = S \cap Y$ とおくと，

$$\begin{aligned} f \circ g(S) &= f(g(S)) = f(S \cap Y) \\ &= (S \cap Y) \cap X \\ &= S \cap (X \cap Y) \qquad (1) \end{aligned}$$

より，$f \circ g$ はセレクション関数である．　□

補題 7　f がセレクション関数，g がランキング関数ならば，$f \circ g$ でセレクション関数，およびランキング関数でないものが存在する．　□

補題 8　f がランキング関数，g がセレクション関数ならば，$f \circ g$ でセレクション関数，およびランキング関数でないものが存在する．　□

補題 9　ある全順序 R に対して，f が n ランキング関数，g が n' ランキング関数ならば，$f \circ g$ は R に対する $min(n, n')$ ランキング関数である．

《証明》

i)　$n > n'$ のとき

任意の $S \subset \mathbf{T}$ に対して，$|g(S)| \leq n' < n$ なので $f(g(S)) = g(S)$ となる．g は R に対する n' ランキング関数なので，$f \circ g$ も R に対する n' ランキング関数である．

ii)　$n \leq n'$ のとき

f，g は閉じているので，任意の S に対し，ある $a_0, a_1 \in \mathbf{T}$ について

$$f(S) = \{x \in S | x < a_0\} \qquad (2)$$

$$g(S) = \{x \in S | x < a_1\} \qquad (3)$$

が成立する．f は度数 n，g は度数 n' なので，

1. $n \leq n' \leq |S|$ のとき $n = |f(S)| \leq |g(S)| = n'$
2. $n \leq |S| < n'$ のとき $n = |f(S)| \leq |g(S)| = |S|$
3. $|S| < n \leq n'$ のとき $|f(S)| = |g(S)| = |S|$

となる．したがって，式 (2)，(3) より

$$|\{x \in S | x < a_0\}| \leq |\{x \in S | x < a_1\}| \qquad (4)$$

が導き出される．ここで次の補題が成立する．

補題 10　全順序 $R = (\mathbf{T}, <)$，任意の $S \subset \mathbf{T}$ に対し，ある $a_0, a_1 \in \mathbf{T}$ について，$\{x \in S | x < a_0\}, \{x \in S | x < a_1\}$ が有限集合のとき，$|\{x \in S | x < a_0\}| \leq |\{x \in S | x < a_1\}|$ ならば $a_0 < a_1$．　□

補題 10 より $a_0 < a_1$ となる．したがって

$$\begin{aligned} f(g(S)) &= f(\{x \in S | x < a_1\}) \\ &= \{x \in S | x < a_0\} = f(S). \end{aligned}$$

f は R に対する n ランキング関数なので，$f \circ g$ も R に対する n ランキング関数である．　□

補題 11　全順序 R，R' に対し，f が R に対する n ランキング関数，g が R' に対する n' ランキング関数であり，$n \geq n'$ ならば，$f \circ g$ は R' に対する n' ランキング関数である．　□

補題 12　全順序 R，R' に対し，f が R に対する n ランキング関数，g が R' に対する n' ランキング関数であり，$n < n'$ ならば，$f \circ g$ でランキング関数

表 1 セレクション関数とランキング関数の合成関数 $f \circ g$
Table 1 Characteristics of composite function $f \circ g$.

f \ g	Selection	Ranking with R	Ranking with R' ($R \neq R'$) (cardinality n')
Selection	Selection	–	–
Ranking with R (cardinality n)	–	Ranking	$\begin{cases} \text{Ranking} & (\text{if } n \geq n') \\ - & (\text{if } n < n') \end{cases}$

でないものが存在する． □

以上の補題から，セレクション関数とランキング関数の合成関数が，セレクション関数，ランキング関数のうちどちらの関数になるかをまとめたものを**表 1** に示す．表 1 中の “–” は，合成関数 $f \circ g$ が必ずしもセレクション関数やランキング関数にならないことを表す．ランキング関数どうしの合成関数においては，f と g に同じ全順序 R が与えられているとき，必ず $f \circ g$ もランキング関数となり，f と g で異なる全順序が与えられているとき，$n \geq n'$ である場合のみ $f \circ g$ もランキング関数となる．

ランキングによるフィルタリングでは，必要な情報を多く含むデータほどより順位を高くしたり，新しいデータほどより順位を高くしたりするといったように，フィルタリングのポリシーによって与えられる全順序が異なる．したがって，表 1 より，ランキングによる手法を組み合わせたフィルタリングにおいて，両者が同じフィルタリングのポリシーでデータの順序を決定する場合は必ずランキング関数の条件を満たすが，それぞれが異なるポリシーでデータの順序を決定する場合は，先のフィルタリングで抽出するデータ数の方が後のフィルタリングで抽出するデータ数よりも小さい場合のみランキング関数の条件を満たすことが明らかとなった．

さらに，セレクション関数とランキング関数の合成関数は，必ずしもセレクション関数，あるいはランキング関数とは限らないことが分かった．

6. 考　　察

本章では，実際に用いられているいくつかのフィルタリング手法を取り上げ，本稿で示した性質から，各手法で実現できる処理方法について述べる．また，それらの手法を合成関数に適用した場合の性質について議論する．

6.1 セレクションとランキングの性質

これまで，あるフィルタリング手法が 2.2 節で示したどの性質を満たすかを明らかにするには，一括処理や分配処理，逐次処理，並列処理を行ったときのフィルタリング結果から，特定の性質を満たすための要因と阻害要因を考慮したり，実際に受信データとフィルタリング結果を性質の式に代入したりする，といった方法で調べる必要があった．しかし，セレクションによるフィルタリングは分配等価性を満たし，ランキングによるフィルタリングは逐次等価性を満たすことが明らかになったことで，あるフィルタリングがセレクションあるいはランキングの手法を用いていることが分かれば，満たす性質を調べる必要がなくなった．

逆に，あるフィルタリングが分配等価性を満たすことが分かれば，その手法をセレクションの簡単な演算式によって表現でき，逐次等価性を満たす度数関数であることが分かれば，ランキングの式によって表現できる．この結果を利用することで，未知のフィルタリングを簡単な演算処理によって実装できるかどうかが判断できる．

セレクションによるフィルタリングとしては，キーワードマッチングを用いたフィルタリングに，XML 文書をフィルタリングする XFilter [1] や，問合せをグループ化する NiagaraCQ [5] などがある．また，閾値を用いたフィルタリングに，距離の公理を用いたフィルタリング [6] や，データやユーザのプロファイルをベクトル表現し，それらのベクトル積が閾値を超えるものを抽出する SIFT [14] などがある．これらは，定理 2，図 2 より，一括処理，分配処理，逐次処理，並列処理すべての結果が等価となることが分かる．また，補題 2，補題 5 より，ϕ あるいは $\mathbf{T}$ のセレクション関数はランキングの特性も満たす．

ランキングによるフィルタリングに，たとえばコンテンツによるフィルタリングをオプションとして選択した場合の ProfBuilder [13] などがある．このようなフィルタリングは，定理 3，図 2 より，一括処理，逐次処理，並列処理の結果が等価となる．さらに，補題 2，補題 3 より，特にランキングが度数 0 あるいは $|\mathbf{T}|$ のときは，セレクションであるための条件を満たすので，分配等価性を満たす．したがって，一括処理，分配処理，逐次処理，並列処理の結果がすべて等価となる．

上記の性質により，セレクションやランキングによるフィルタリングでは，ネットワークの帯域に十分余裕がある場合，一括処理を選択することでサーバの負

荷を軽減でき，ネットワークが混んできた場合は，複数のサイトから同時にデータを受信する並列処理に切り替えても等しい結果が得られる．また，受信機の計算能力が低い場合は，受信機を複数用意することでスループットを上げることができ[4]，記憶容量が比較的小さい場合は逐次処理に等価変換できる．

しかし，度数が 0 あるいは $|\mathbf{T}|$ 以外のランキングによるフィルタリングでは，一括処理や逐次処理，並列処理の結果と分配処理の結果が等価となることを保証できない．つまり，フィルタリング結果の等価性を維持しながら，処理の途中で処理方法を分配処理へと変換することができない．したがって，実装の段階でフィルタリング環境を十分調査し，最適な処理方法を決定しておくことが必要である．

6.2 セレクションとランキングの合成

セレクションによるフィルタリングとランキングによるフィルタリングを合成した場合，どちらのフィルタリングを先に行っても，必ずしもセレクション関数やランキング関数であるための条件を満たさない．したがって，合成したフィルタリングは，セレクションやランキングを用いたフィルタリングと同様の特性を持たない．

一方，セレクションによるフィルタリングを合成した場合，セレクション関数であるための条件を満たすため，一括処理，分配処理，逐次処理，並列処理の結果がすべて等価となる．したがって，たとえば NiagaraCQ と SIFT を組み合わせたシステムでは，処理中でも環境に応じてより効率的な処理方法へと変換することができる．また，フィルタリングの計算コストを軽減するために，先に簡単な論理演算によってデータを絞り込んでおいてから，後で複雑な論理式を適用するといったフィルタリングの場合でも，すべての処理方法による結果が等価となる．

ランキングによるフィルタリングの合成において，先のフィルタリングで抽出するデータ数が後のフィルタリングで抽出するデータ数よりも小さい場合や，両者が等しい順序にデータを並べる場合は，必ずランキング関数であるための条件を満たす．したがって，このようなフィルタリングでは，逐次等価性を満たすため，一括処理，逐次処理，並列処理の結果が等価となる．

7. おわりに

本稿では，一般のフィルタリングにおいて広く用いられているセレクションとランキングを独自の関数で定義し，その性質を明らかにした．このように，特定の手法に焦点を絞った議論を行うことで，その特性を詳細に調べることができる．また，本稿で示した体系を実際にセレクションやランキングを用いているフィルタリングに適用することで，環境に応じてより効率的なフィルタリング処理が実現できることを述べた．

今後の課題を以下に示す．

- 新たな関数の定義
  たとえば，蓄積するディスク容量に制限を設ける手法は，各データのサイズとディスクの空き容量を考慮してデータの取捨選択をするので，セレクションやランキングの手法だけでは実現できない．また，一般のフィルタリングで用いられる，データ内容の相関性やユーザの嗜好の変化を考慮に入れた手法も定式化できない．このような本稿で述べたもの以外の手法について，処理手法を本稿同様に独自の関数で定義し，その性質を明らかにする．
- 合成関数への条件の付加
  本稿で論じた合成関数では，必ずしももとの関数の性質が保たれない．しかし，フィルタリング関数を合成する際，特定の条件を追加することで合成関数の性質が保たれる可能性がある．

謝辞 本研究は，科学技術振興調整費任期付研究者支援「情報フィルタリングの数学的基盤の確立」，および文部科学省振興調整費「モバイル環境向 P2P 型情報共有基盤の確立」の研究助成によるものである．ここに記して謝意を表す．

参 考 文 献

1) Altinel, M. and Franklin, M.J.: Efficient filtering of XML documents for selective dissemination of information, *Proc. 26th International Conference on Very Large Data Bases* (*VLDB2000*), pp.53–64 (2000).

2) Belkin, N.J. and Croft, W.B.: Information filtering and information retrieval: two sides of the same coin?, *Comm. ACM*, Vol.35, No.12, pp.29–38 (1992).

3) Bell, T.A.H. and Moffat, A.: The design of a high performance information filtering system, *Proc. SIGIR '96*, pp.12–20 (1996).

4) Bowen, T.F., Gopal, G., Herman, G., Hickey, T., Lee, K.C., Mansfield, W.H., Raitz, J. and Weinrib, A.: The datacycle architecture, *Comm. ACM*, Vol.35, No.12, pp.71–81 (1992).

5) Chen, J., DeWitt, D.J., Tian, F. and Wang, Y.: NiagaraCQ: a scalable continuous query system for internet databases, *Proc. ACM SIGMOD2000*, pp.379–390 (2000).

6) カンギョウビ，大和田俊和，浅田一繁，飯沢篤志，

古瀬一隆：情報放送システムにおける距離の近似を利用したフィルタリング方式，電子情報通信学会第11回データ工学ワークショップ（DEWS2000）論文集（CD-ROM）(2000).
7) 森田昌宏：情報フィルタリングに関する研究動向，JAIST Research Report, IS-RR-93-9I, 北陸先端科学技術大学院大学情報科学研究科 (1993).
8) 西　正，野村敦子：多チャンネル放送の衝撃，中央経済社 (1997).
9) Satellite Magazine. http://www.satemaga.co.jp
10) 澤井里枝，寺田　努，塚本昌彦，西尾章治郎：フィルタリング SQL：フィルタリングのためのユーザ要求記述言語，電子情報通信学会第11回データ工学ワークショップ（DEWS2000）論文集（CD-ROM）(2000).
11) Sawai, R., Tsukamoto, M., Loh, Y.H., Terada, T. and Nishio, S.: Functional properties of information filtering, *Proc. 27th International Conference on Very Large Data Bases (VLDB2001)*, pp.511–520 (2001).
12) 澤井里枝，塚本昌彦，寺田　努，Loh Yin Huei, 西尾章治郎：情報フィルタリングの関数的性質について，電子情報通信学会論文誌 D-I, Vol.J85-D-I, No.10, pp.939–950 (2002).
13) Wasfi, A.M.A.: Collecting user access patterns for building user profiles and collaborative filtering, *Proc. 1999 International Conference on Intelligent User Interfaces*, pp.57–64 (1999).
14) Yan, T.W. and Garcia-Molina, H.: The SIFT information dissemination system, *ACM Trans. Database Syst.*, Vol.24, No.4, pp.529–565 (1999).

付録　定理，補題の証明

《補題 1 の証明》 ある全順序 $R = (\mathbf{T}, <)$ が与えられているとき，度数 n の閉関数 f_1，f_2 が，ある $X \subset \mathbf{T}$ に対して $f_1(X) \neq f_2(X)$ であると仮定する．

f_1，f_2 は，ある $a, b \in \mathbf{T}$ に対して，$f_1(X) = R_a \cap X$，$f_2(X) = R_b \cap X$ とおける．ここで，$f_1(X) \neq f_2(X)$ から $a \neq b$ なので，$|\{x \in X | x < a\}| \neq |\{x \in X | x < b\}|$ となる．これは，f_1，f_2 が同じ度数 n であることに反する．□

《定理 3 の証明》

$$\exists R, \forall T, \exists a, f(T) = \{x \in T | x < a\}$$
$$\begin{cases} |f(T)| = n & (|T| \geq n) \\ f(T) = T & (|T| < n) \end{cases}$$
$$\Longleftrightarrow \forall S, \forall T, f(S \cup T) = f(S \cup f(T))$$
$$\begin{cases} |f(T)| = n & (|T| \geq n) \\ f(T) = T & (|T| < n) \end{cases}$$

であることを証明する．

i) $(\Longrightarrow)$

1)　$|T| < n$ のとき

$f(T) = T$ となるので，度数 n であることを満たす．また，$f(S \cup f(T)) = f(S \cup T)$ が成り立つ．

2)　$|T| \geq n$ のとき

f は閉じているので，ある $a_0, a_1 \in \mathbf{T}$ について

$$f(S \cup T) = \{x \in S \cup T | x < a_0\} \quad (5)$$
$$f(T) = \{x \in T | x < a_1\} \quad (6)$$

となる．f は度数 n なので，式 (5) は

$$|f(S \cup T)| = |\{x \in S \cup T | x < a_0\}| = n \quad (7)$$

を満たし，かつ

$$\{x \in S \cup T | x < a_0\} \supset \{x \in T | x < a_0\} \quad (8)$$

であることから

$$|\{x \in T | x < a_0\}| \leq n \quad (9)$$

が成立する．さらに，f が度数 n であることから式 (6) は

$$|f(T)| = |\{x \in T | x < a_1\}| = n \quad (10)$$

を満たす．補題 10，式 (9)，(10) より，$a_0 < a_1$ となるので，任意の $x \in f(S \cup T)$ に対して式 (5)，(6) より，

$$x \in S \cup f(T)$$

が成立する．したがって

$$\{x \in S \cup f(T) | x < a_0\} = \{x \in S \cup T | x < a_0\}.$$

ゆえに $f(S \cup f(T)) = \{x \in S \cup f(T) | x < a_0\}$ に対して

$$f(S \cup f(T)) = f(S \cup T)$$
$$|f(S \cup f(T))| = n$$

が導き出される．

ii) $(\Longleftarrow)$

$|\mathbf{T}| \leq n$ のとき成り立つことは自明である．

$|\mathbf{T}| > n$ を仮定する．$R \subset \mathbf{T} \times \mathbf{T}$ を以下のように定義する．まず，

$$|f(\mathbf{T})| = n$$
$$A = f(\mathbf{T}) = \{a_1, a_2, \ldots, a_n\}$$
$$A^- = \{a_1, a_2, \ldots, a_{n-1}\}$$

として

$$R^{(1)} \triangleq \{(a_i, a_j) | 1 \leq i \leq j \leq n\}$$
$$R^{(2)} \triangleq \{(x, y) | x \in A, y \in \mathbf{T} - A\}$$
$$R^{(3)} \triangleq \{(x, y) | x, y \in \mathbf{T} - A, x \in f(A^- \cup \{x, y\})\}$$

と定義し，

$$R \triangleq R^{(1)} \cup R^{(2)} \cup R^{(3)}$$

とする．$R^{(1)}$ は集合 A の全順序集合を表し，$R^{(2)}$ は A の要素と $\mathbf{T}-A$ の要素間の順序を表す．また，$R^{(3)}$ は $\mathbf{T}-A$ の全順序集合を表す．任意の $T \subset \mathbf{T}$ に対して $A \cap T \subset f(T)$ となるので，f によって $R^{(1)}$ を特定できない．ゆえに，A の全順序は任意の順序で定義できる．一方，$R^{(2)}$ と $R^{(3)}$ は定義のとおり f によって特定できる．

ここで，f が度数 n であり逐次等価性を満たすならば R が全順序であることを示せばよい．R が全順序であることは

1)　$\forall x \in \mathbf{T}, (x,x) \in R$

2)　$\forall x,y \in \mathbf{T}, (x,y) \in R, (y,x) \in R$ ならば $x=y$

3)　$\forall x,y,z \in \mathbf{T}, (x,y) \in R, (y,z) \in R$ ならば $(x,z) \in R$

4)　$\forall x,y \in \mathbf{T}, (x,y) \in R$ あるいは $(y,x) \in R$

を示すことで証明できる．以下にそれぞれが成り立つことを示す．

1)　$R^{(1)} \supset \{(a_i,a_j) | 1 \le i = j \le n\}$ より

$$\forall x \in A, (x,x) \in R^{(1)}. \tag{11}$$

また，$R^{(3)} \supset \{(x,x) | x \in \mathbf{T}-A, x \in f(A^- \cup \{x\})\}$ より

$$\forall x \in \mathbf{T}-A, (x,x) \in R^{(3)} \tag{12}$$

となる．したがって式 (11)，(12) より，1) が導き出される．

2)　$R^{(1)} = \{(a_i,a_j) | 1 \le i \le j \le n\}$ より，$\forall a_i, a_j \in A$ に対して

$$(a_i,a_j) \in R^{(1)} \text{ ならば } i \le j \tag{13}$$
$$(a_j,a_i) \in R^{(1)} \text{ ならば } j \le i \tag{14}$$

が成り立つ．式 (13)，(14) より，$i=j$ となるので，

$$\forall x,y \in A, (x,y) \in R^{(1)}, (y,x) \in R^{(1)} \text{ ならば } x=y \tag{15}$$

となる．また，$R^{(3)} = \{(x,y) | x,\ y \in \mathbf{T}-A,\ x \in f(A^- \cup \{x,y\})\}$ より，$\forall x,y \in \mathbf{T}-A$ に対して $(x,y) \in R^{(3)}$ ならば

$$f(A^- \cup \{x,y\}) = A^- \cup \{x\}, \tag{16}$$

$(y,x) \in R^{(3)}$ ならば

$$f(A^- \cup \{x,y\}) = A^- \cup \{y\} \tag{17}$$

となる．ここで $x \ne y$ と仮定すると，式 (16)，(17) の結果が矛盾する．したがって，

$$\forall x,y \in \mathbf{T}-A, (x,y) \in R^{(3)}, (y,x) \in R^{(3)} \text{ ならば } x=y \tag{18}$$

が成り立つ．ゆえに式 (15)，(18) より 2) が導き出される．

3)　$\forall x,y,z \in \mathbf{T}$，$(x,y) \in R$，$(y,z) \in R$ である x，y，z について以下のように場合分けする．

(a)　$x,y,z \in A$ のとき
$R^{(1)} = \{(a_i,a_j) | 1 \le i \le j \le n\}$ より，$\forall a_i, a_j, a_t \in A, (a_i,a_t) \in R^{(1)}$ ならば

$$i \le t \tag{19}$$

となり，また $(a_t,a_j) \in R^{(1)}$ ならば

$$t \le j. \tag{20}$$

式 (19)，(20) より $i \le j$ となるので，任意の $a_i, a_j, a_t \in A$ に対して，$(a_i,a_t) \in R^{(1)}$ かつ $(a_t,a_j) \in R^{(1)}$ ならば $(a_i,a_j) \in R^{(1)}$ が成立する．

(b)　$x,y \in A$，$z \in \mathbf{T}-A$ のとき
$(x,y) \in R^{(1)}$，$(y,z) \in R^{(2)}$ ならば，$x \in A$，$z \in \mathbf{T}-A$ なので，$R^{(2)} = \{(x,y) | x \in A, y \in \mathbf{T}-A\}$ より $(x,z) \in R^{(2)}$ が成立する．

(c)　$x \in A$，$y,z \in \mathbf{T}-A$ のとき
$(x,y) \in R^{(2)}$，$(y,z) \in R^{(3)}$ ならば，$x \in A$，$z \in \mathbf{T}-A$ なので $R^{(2)} = \{(x,y) | x \in A, y \in \mathbf{T}-A\}$ より $(x,z) \in R^{(2)}$ が成立する．

(d)　$x,y,z \in \mathbf{T}-A$ のとき
後述の 4) より，任意の $x,y \in \mathbf{T}-A$ に対して，$(x,y) \in R^{(3)}$ あるいは $(y,x) \in R^{(3)}$ である．ここで任意の $x,y,z \in \mathbf{T}-A$ に対して，$(x,y) \in R^{(3)}$，$(y,z) \in R^{(3)}$ ならば $(z,x) \in R^{(3)}$ と仮定する．任意の $S,T \subset \mathbf{T}$ に対して $f(S \cup T) = f(S \cup f(T))$ が成立しているので，同じ引数に対し

$$\begin{aligned} & f(A^- \cup \{x,y,z\}) \\ & = f((A^- \cup \{y\}) \cup (A^- \cup \{x,z\})) \\ & = f((A^- \cup \{y\}) \cup f(A^- \cup \{x,z\})) \\ & = f((A^- \cup \{y\}) \cup (A^- \cup \{z\})) \\ & = f(A^- \cup \{y,z\}) \\ & = A^- \cup \{y\} \end{aligned} \tag{21}$$

$$\begin{aligned} & f(A^- \cup \{x,y,z\}) \\ & = f((A^- \cup \{x\}) \cup (A^- \cup \{y,z\})) \\ & = f((A^- \cup \{x\}) \cup f(A^- \cup \{y,z\})) \\ & = f((A^- \cup \{x\}) \cup (A^- \cup \{y\})) \\ & = f(A^- \cup \{x,y\}) \\ & = A^- \cup \{x\} \end{aligned} \tag{22}$$

の 2 通りの結果が得られる．式 (21)，(22) は矛盾するので，任意の $x,y,z \in \mathbf{T}-A$ に対して，$(x,y) \in R^{(3)}$，$(y,z) \in R^{(3)}$ ならば $(x,z) \in R^{(3)}$ が成立する．

以上より，3) が導き出される．

表 2　反例 1
Table 2　Counter example 1.

x	$f(x)$	$g(x)$	$f(g(x))$
ϕ	ϕ	ϕ	ϕ
$\{a\}$	ϕ	$\{a\}$	ϕ
$\{b\}$	$\{b\}$	$\{b\}$	$\{b\}$
$\{a,b\}$	$\{b\}$	$\{a\}$	ϕ

表 3　反例 2
Table 3　Counter example 2.

x	$f(x)$	$g(x)$	$f(g(x))$
ϕ	ϕ	ϕ	ϕ
$\{a\}$	$\{a\}$	ϕ	ϕ
$\{b\}$	$\{b\}$	$\{b\}$	$\{b\}$
$\{c\}$	$\{c\}$	$\{c\}$	$\{c\}$
$\{a,b\}$	$\{a\}$	$\{b\}$	$\{b\}$
$\{a,c\}$	$\{a\}$	$\{c\}$	$\{c\}$
$\{b,c\}$	$\{b\}$	$\{b,c\}$	$\{b\}$
$\{a,b,c\}$	$\{a\}$	$\{b,c\}$	$\{b\}$

4)　R_1，R_2，R_3 の定義より，

$$\forall a_i, a_j \in A, (a_i, a_j) \in R^{(1)} \text{ あるいは } (a_j, a_i) \in R^{(1)} \quad (23)$$
$$\forall x \in A, \forall y \in \mathbf{T} - A, (x, y) \in R^{(2)} \quad (24)$$
$$\forall x, y \in \mathbf{T} - A, (x, y) \in R^{(3)} \text{ あるいは } (y, x) \in R^{(3)} \quad (25)$$

であることが成立する．したがって，4) が導き出される．

以上より，R が全順序であることが示された．この R について

$$\forall T, \exists a, f(T) = \{x \in T | x < a\}$$
$$\begin{cases} |f(T)| = n & (|T| \geq n) \\ f(T) = T & (|T| < n) \end{cases}$$

が成立する．□

《補題 7 の証明》 $\mathbf{T} = \{a, b\}$ とする．**表 2** に示す f はセレクション関数，g は 1 ランキング関数である．$X = f(g(\mathbf{T})) = \phi$ に対して，$S = \{b\}$ のとき $f(g(S)) = S \cap X$ を満たさないので $f \circ g$ はセレクション関数ではない．また，$S = \{b\}$，$T = \{a, b\}$ のとき $f(g(S \cup T)) = f(g(S \cup f(g(T))))$ を満たさないので $f \circ g$ はランキング関数ではない．□

《補題 8 の証明》 $\mathbf{T} = \{a, b\}$ とする．**表 3** に示す f は 1 ランキング関数，g はセレクション関数である．$X = f(g(\mathbf{T})) = \{b\}$ に対して，$S = \{a, c\}$ のとき $f(g(S)) = S \cap X$ を満たさないので $f \circ g$ はセレクション関数ではない．また，$f \circ g$ は度数が決まらないのでランキング関数でない．□

表 4　反例 3
Table 4　Counter example 3.

x	$f(x)$	$g(x)$	$f(g(x))$
ϕ	ϕ	ϕ	ϕ
$\{a\}$	$\{a\}$	$\{a\}$	$\{a\}$
$\{b\}$	$\{b\}$	$\{b\}$	$\{b\}$
$\{c\}$	$\{c\}$	$\{c\}$	$\{c\}$
$\{a,b\}$	$\{a\}$	$\{a,b\}$	$\{a\}$
$\{a,c\}$	$\{a\}$	$\{a,c\}$	$\{a\}$
$\{b,c\}$	$\{b\}$	$\{b,c\}$	$\{b\}$
$\{a,b,c\}$	$\{a\}$	$\{b,c\}$	$\{b\}$

《補題 10 の証明》 $a_1 < a_0$ と仮定すると，$\forall y \in \{x \in S | x < a_1\}$ に対して $y < a_1 < a_0$ であることから

$$y \in \{x \in S | x < a_0\}$$

が成り立つ．したがって，

$$\{x \in S | x < a_0\} \supset \{x \in S | x < a_1\}$$

かつ，$a_1 \neq a_0$ を仮定すると

$$\{x \in S | x < a_0\} \neq \{x \in S | x < a_1\}$$

となる．ゆえに

$$|\{x \in S | x < a_0\}| > |\{x \in S | x < a_1\}|$$

が導き出される．これは

$$|\{x \in S | x < a_0\}| \leq |\{x \in S | x < a_1\}|$$

であることに反する．□

《補題 11 の証明》 補題 9 の証明の i) と同様に証明できる．□

《補題 12 の証明》 $\mathbf{T} = \{a, b, c\}$ とする．**表 4** に示す f は 1 ランキング関数，g は 2 ランキング関数である．$S = \{a, b\}$，$T = \{a, b, c\}$ のとき $f(g(S \cup T)) = f(g(S \cup f(g(T))))$ を満たさないので $f \circ g$ はランキング関数ではない．□

(平成 14 年 6 月 19 日受付)
(平成 14 年 9 月 29 日採録)

（担当編集委員　仲尾 由雄）

澤井　里枝

2000 年大阪大学工学部電子情報エネルギー工学科卒業．2002 年同大学院工学研究科博士前期課程修了．現在，同大学院情報科学研究科マルチメディア工学専攻博士後期課程に在学中．

塚本　昌彦（正会員）

1987年京都大学工学部数理工学科卒業．1989年同大学院工学研究科修士課程修了．同年シャープ（株）に入社，同社研究員．1995年大阪大学大学院工学研究科講師．1996年より同大学院工学研究科助教授，2002年より同大学院情報科学研究科マルチメディア工学専攻助教授，現在に至る．工学博士．モバイルコンピューティング，分散知識ベースシステムの研究開発に従事．ACM，IEEE等7学会各会員．

寺田　　努（正会員）

1997年大阪大学工学部情報システム工学科卒業．1999年同大学院工学研究科博士前期課程修了．2000年同大学院工学研究科博士後期課程退学．同年より大阪大学サイバーメディアセンター助手，現在に至る．2002年より同大学院情報科学研究科マルチメディア工学専攻助手を併任．アクティブデータベース，モバイルコンピューティング，データ放送の研究に従事．

西尾章治郎（正会員）

1975年京都大学工学部数理工学科卒業．1980年同大学院工学研究科博士課程修了．工学博士．京都大学工学部助手，大阪大学基礎工学部および情報処理教育センター助教授，大阪大学大学院工学研究科情報システム工学専攻教授を経て，2002年より同大学院情報科学研究科マルチメディア工学専攻教授となり，現在に至る．2000年より大阪大学サイバーメディアセンター長を併任．この間，カナダ・ウォータールー大学，ビクトリア大学客員．データベース，知識ベース，分散システムの研究に従事．現在，ACM Trans. on Internet Technology，Data & Knowledge Engineering，DataMining and Knowledge Discovery，The VLDB Journal等の論文誌編集委員．本学会フェロー含め，ACM，IEEE等8学会の会員．