

フィルタリング関数の合成と実システムへの適用について

澤井里枝[†] 塚本昌彦[†]
寺田 努^{††} 西尾 章治郎[†]

筆者らはこれまで、情報フィルタリングの数学的基盤を構築するために、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、それらを合成することで、複数の手法を組合せたフィルタリングの性質を明らかにしてきた。フィルタリングの数学的基盤を構築することにより、フィルタリングの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などが可能となる。本稿では、さらに多様な組合せのフィルタリングの性質を明確にするため、これまでに定義したフィルタリング関数と、新たにフィルタリング結果の数に関する制約条件を付けたフィルタリング関数との合成関数の性質を明らかにする。また、本稿で明らかになる結果から、フィルタリング関数の枠組みを実システムへ適用する方法について考察する。

Composition of Filtering Functions and its Application of Practical Systems

RIE SAWAI,[†] MASAHIKO TSUKAMOTO,[†] TSUTOMU TERADA^{††}
and SHOJIRO NISHIO[†]

In our previous works, to establish mathematical foundation of information filtering, we defined a notion of filtering function that represents filtering as a function. Moreover, we clarified the characteristics of filtering consisting of multiple methods by composite filtering functions. The constructed mathematical foundation makes it possible to qualitatively evaluate various filtering methods, to optimize processing methods in filtering, and to design a declarative language for describing the filtering policy. In this paper, to clarify the characteristics of filtering that combines more huge variety of methods, we reveal properties of the composite functions of previously defined filtering function and filtering function with the constraint for number of the data in filtering result. Additionally, based on the properties clarified in this paper, we show how to apply the framework of filtering function to practical filtering systems.

1. はじめに

近年、ネットワークのブロードバンド化や、放送のデジタル化および多チャンネル化により、さまざまな放送型サービスが提供されるようになった^{3),6),7)}。このような環境では、多様で膨大なデータを受信できるが、一般にユーザが必要とする情報はごく一部に限られているため、受信データから必要なデータ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を取捨選択する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多数提案されている^{1),2),5),8)}。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、処理方法に関する基本的な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として定性的に表現した^{9),11)}。また、フィルタリング関数が満たす性質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。

実際のフィルタリングは、複数の手法を組合せて実現するのが一般的である。そこで筆者らは、複数の手法を組合せたフィルタリングの特性を明らかにするため、フィルタリング関数の合成について議論した^{10),12)}。文献10)では、全ての受信データを一つの受信機で一括に処理した場合と複数の受信機で並列に処理した場合のフィルタリング結果が等しくなるといったように、処理方法に関する性質を満たすフィルタリング関数に

[†] 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

対して、それらの合成関数の性質を明らかにした。一方、文献 12) で筆者らは、一般のフィルタリングにおいて最も重要かつ頻繁に利用されるセクションとランキングという 2 つの手法を独自の関数で定義し、それらの合成関数の性質を明らかにした。セクションとは、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。例えば、特定のキーワードを含むデータを蓄積するキーワードマッチングや、データの内容から評価値を計算し、評価値が閾値よりも大きい(あるいは小さい)場合に蓄積する手法などはセクションである。また、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、最も重要なデータを特定の数だけ選択する手法である。

しかし、文献 10) で取り扱うフィルタリング手法の分類が、処理方法の違いによるフィルタリング結果の等価性に基いているのに対し、文献 12) は、データの取捨選択を決定するアルゴリズムに基づいている。すなわち、両者で取り扱う性質がそれぞれ異なり、フィルタリング手法の分類基準も異なるため、各枠組みで取り扱う性質間の関係が明確でなかった。そこで本稿では、両者の枠組みを統合し、処理方法に関する性質およびフィルタリングアルゴリズムの両面から合成フィルタリング関数の性質を明らかにする。これまで、フィルタリング結果の数に関する性質は、文献 12) の枠組みでのみ取り扱ってきたが、新たに文献 10) の枠組みにも導入する。また、本稿では、一つの枠組みで取り扱うフィルタリング関数どうしの合成関数だけでなく、両枠組みで取り扱うフィルタリング関数の合成関数の性質を明らかにする。これまでに定義したさまざまな性質を統合的に取り扱えるようにすることで、本研究で構築する枠組みを実システムへ適用する際、システムの性能比較や最適化をより柔軟に行えるようになる。

以下、第 2 章でフィルタリング関数の概要を述べる。第 3 章では、これまでの枠組みを統合したフィルタリング関数の合成関数の性質を明らかにする。第 4 章では、本稿で明らかになった結果を、実際のフィルタリングシステムへ適用する場合の考察を行う。最後に第 5 章でまとめを行う。

2. フィルタリング関数

本章では、本稿の基礎となるフィルタリング関数の基本的な概念について述べる。

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度に、新たな受信データと前回のフィルタリング結果を合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 T とする。フィルタリング関数とは、任意の $T \subset T$ に対し、以下の 2 つの条件を満たす 2^T 上の関数 f のことをいう^{9),11)}。

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。

逐次等価性 (SE: Sequential Equivalence)

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

分配等価性 (DE: Distributed Equivalence)

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

並列等価性 (PE: Parallel Equivalence)

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

ここで、 S, T は T の任意の部分集合とする。逐次等価性は一括処理と逐次処理の結果が等価であることを意味する。同様に、分配等価性は一括処理と分配処理の結果が等価であり、並列等価性は一括処理と並列処理の結果が等価であることを意味する。これまでに筆者らは、これらの等価性間に図 1 に示すような包含関係があることを明らかにした^{9),11)}。図 1 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わかる。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 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。

2.3 セクション関数とランキング関数

文献 12) において、セクション関数とランキング関数を次のように定義した。

ある $X \subset T$ について、 X のセクション関数 B_X とは、任意の $S \subset T$ に対して $B_X(S) \triangleq S \cap X$ と定義される関数である。 X をこのセクション関数の潜在集合 (potential set) と呼び、蓄積条件を満たすデー

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。

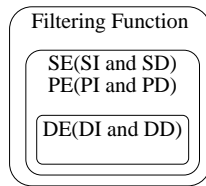


図1 等価性間の関係

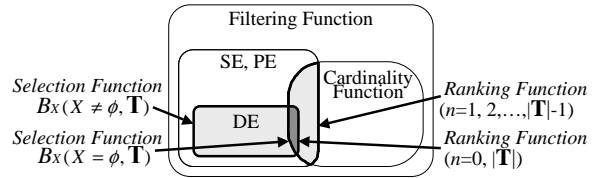


図2 セレクション関数とランキング関数の関係

タの集合を意味する。したがって、キーワードマッチングの潜在集合は、特定のキーワードを含むデータの集合であり、閾値を用いたフィルタリングの潜在集合は評価値が閾値よりも大きい(あるいは小さい)データの集合である。セレクション関数は $X = B_X(\mathbf{T})$ を満たす。

一方、ある全順序 $R = (\mathbf{T}, <)$ に対する n ランキング関数 f とは、任意の $X \subset \mathbf{T}$ に対し、ある $a \in \mathbf{T}$ について $f(X) \triangleq \{x \in \mathbf{T} | x < a\} \cap X$ と定義される度数 n の関数である。関数 f が度数 n であるとは、任意の $X \subset \mathbf{T}$ に対して

- X が無限集合、あるいは X が有限集合であり $|X| \geq n$ ならば $|f(X)| = n$.

- X が有限集合であり $|X| < n$ ならば $f(X) = X$.

この2つの条件が成立することである。度数 n であるフィルタリング関数を度数関数 (*cardinality function*) と呼ぶ。度数 n は蓄積するデータの個数を意味し、受信データの数とその数に満たない場合は、全ての受信データを蓄積する。また、全順序 $R = (\mathbf{T}, <)$ は、全データアイテムの重要度の順序を表す。“ $<$ ” はデータアイテム間の順序を表し、“ $d_1 < d_2$ ” は d_1 の方が d_2 よりも重要度が高いことを意味する。ランキング関数は、度数に合せてデータアイテム a よりも重要度が高いデータを蓄積するため、フィルタリングするデータ集合によって a は変化する。

セレクション関数とランキング関数の性質に関して、以下の定理が成立する¹²⁾。

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 が分配等価性を満たすことは同値である。 □

定理2 フィルタリング関数 f がある全順序 $(\mathbf{T}, <)$ に対して 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。 □

また、等価性を満たすフィルタリング関数、セレクション関数、ランキング関数の包含関係を図2に示す¹²⁾。

2.4 フィルタリング関数の合成

フィルタリング関数の合成関数は必ずしもフィルタリング関数になるとは限らない。そこで文献10)では、合成関数がフィルタリング関数となるための条件を次

表1 等価性を満たすフィルタリング関数の合成

$f \setminus g$	DE	SE, PE
DE	DE	\neg DE, \neg SE, \neg PE
SE, PE	SE, PE, \neg DE	\neg SE, \neg PE

のように示した。

フィルタリング関数 f, g に対して、 f が g にフィルタリング合成可能であるとは、合成関数 $f \circ g$ がフィルタリング関数であることをいう。このとき以下の定理が成り立つ¹⁰⁾。

定理3 フィルタリング関数 f, g に対して、 g がSI(またはそれと等価なPI, DI, C)を満たすならば、 f が g にフィルタリング合成可能である。

等価性を満たすフィルタリング関数の全ての組合せについて、合成関数の性質を表1に示す¹⁰⁾。表1中の各要素は f, g がそれぞれ行、列の性質をもち、 f が g にフィルタリング合成可能であるとき、合成関数 $f \circ g$ が満たす性質を表す。また、“ \neg ”はその性質を必ずしも満たさないことを表す。表1より、分配等価性を満たすフィルタリング関数 (g) を適用した後は、分配等価性、逐次等価性、並列等価性のいずれの性質を満たすフィルタリング関数を適用した場合でも、後に適用したフィルタリング関数 (f) の等価性を保つことが示された。

また、表2に、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示す¹²⁾。表2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、度数が $n \geq n'$ である場合のみ $f \circ g$ もランキング関数となる。

3. 合成関数の性質

本章では、前章で述べた各性質を満たすフィルタリング関数について、合成関数が満たす性質を明らかにする。

図2のDE(分配等価性)、SE(逐次等価性)、PE(並列等価性)が文献10)で取り扱ったフィルタリング関数であり、セレクション関数、ランキング関数、度数

表 2 セレクション関数とランキング関数の合成関数 $f \circ g$

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

表 3 反例 1

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

表 5 反例 3

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 4 反例 2

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

表 6 反例 4

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

関数が文献 12) で取り扱ったフィルタリング関数である。表 1, 表 2 に示したように, 等価性を満たすフィルタリング関数どうしを合成した場合や, セレクション関数とランキング関数を合成した場合の性質は既にならなっている。しかし, 等価性を満たすフィルタリング関数とセレクション関数・ランキング関数を合成した場合の性質や, 逐次等価性を満たすが度数関数ではないフィルタリング関数を合成した場合の性質は明らかになっていない。また, これまでは, 合成関数が合成前のフィルタリング関数の性質を満たすかどうかについてのみ明らかにしてきたが, 等価性を満たすフィルタリング関数どうしを合成した場合にランキング関数であるための条件を満たすか, といったように, 合成前のフィルタリング関数が満たさない性質を満たすかどうかは明らかにしていない。したがって, これらの未だ明らかになっていない性質について, 以下の補題を示す。

補題 1 f, g がセレクション関数であるとき, $f \circ g$ は必ずしもランキング関数でない。

《証明》 $T = \{a, b\}$ とする。表 3 に示す f は $\{a\}$ のセレクション関数であり, g は $\{a, b\}$ のセレクション関数である。 $f \circ g$ は度数関数でないので, ランキング関数でない。□

補題 2 f がセレクション関数であり, g が逐次等価性を満たすフィルタリング関数であるが度数関数でないとき, $f \circ g$ は必ずしも分配等価性および逐次等価性を満たさない。また, $f \circ g$ は必ずしもセレクション関

数およびランキング関数でない。

《証明》 $T = \{a, b, c\}$ とする。表 4 に示す f は $\{a\}$ のセレクション関数であり, g は逐次等価性を満たすが, 度数関数ではない。 $S = \{a, b\}, T = \{b, c\}$ のとき $f(g(S \cup T)) = f(g(S)) \cup f(g(T)), f(g(S \cup T)) = f(g(S \cup f(g(T))))$ を満たさない。したがって, $f \circ g$ はセレクション関数でもランキング関数でもない。□

補題 3 f がセレクション関数であり, g がランキング関数であるとき, $f \circ g$ は必ずしも分配等価性および逐次等価性を満たさない。

《証明》 $T = \{a, b\}$ とする。表 5 に示す f は $\{b\}$ のセレクション関数である。また, g は 1 ランキング関数である。 $S = \{b\}, T = \{a, b\}$ のとき $f(g(S \cup T)) = f(g(S)) \cup f(g(T)), f(g(S \cup T)) = f(g(S \cup f(g(T))))$ を満たさない。□

補題 4 f が逐次等価性を満たすフィルタリング関数であるが度数関数ではなく, g がセレクション関数であるとき, $f \circ g$ は必ずしも分配等価性を満たさない。また, $f \circ g$ は必ずしもセレクション関数およびランキング関数でない。

《証明》 $T = \{a, b, c\}$ とする。表 6 に示す f は逐次等価性を満たすが度数関数ではない。また, g は $\{a, b, c\}$ のセレクション関数である。 $S = \{a, b\}, T = \{b, c\}$ のとき $f(g(S \cup T)) = f(g(S)) \cup f(g(T))$ を満たさない。したがって, $f \circ g$ はセレクション関数でない。また, $f \circ g$ は度数関数でないため, ランキング関数でない。□

表 7 反例 5

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

表 8 反例 6

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

表 9 反例 7

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

補題 5 f, g が逐次等価性を満たすフィルタリング関数であるが度数関数でないとき, $f \circ g$ は必ずしも分配等価性および逐次等価性を満たさない. また $f \circ g$ は必ずしもセレクション関数およびランキング関数でない.

◀ 証明 ▶ $T = \{a, b, c\}$ とする. 表 7 に示す f, g は逐次等価性を満たすが度数関数ではない. $S = \{a, b\}, T = \{b, c\}$ のとき $f(g(S \cup T)) = f(g(S)) \cup f(g(T)), f(g(S \cup T)) = f(g(S \cup f(g(T))))$ を満たさない. したがって, $f \circ g$ はセレクション関数でもランキング関数でもない. □

補題 6 f が逐次等価性を満たすフィルタリング関数であるが度数関数でなく, g がランキング関数であるとき, $f \circ g$ は必ずしも分配等価性および逐次等価性を満たさない. また, $f \circ g$ は必ずしもセレクション関数およびランキング関数でない.

◀ 証明 ▶ $T = \{a, b\}$ とする. 表 5 に示す f は逐次等価性を満たすが度数関数ではない. また g は 1 ランキング関数である. $S = \{b\}, T = \{a, b\}$ のとき $f(g(S \cup T)) = f(g(S)) \cup f(g(T)), f(g(S \cup T)) = f(g(S \cup f(g(T))))$ を満たさない. したがって, $f \circ g$ はセレクション関数

表 10 反例 8

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

表 11 反例 9

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

表 12 反例 10

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

でもランキング関数でもない. □

補題 7 f がランキング関数であり, g がセレクション関数であるとき, $f \circ g$ は必ずしも分配等価性を満たさない. また, $f \circ g$ は必ずしもセレクション関数でない.

◀ 証明 ▶ $T = \{a, b, c\}$ とする. 表 8 に示す f は 1 ランキング関数である. また g は $\{a, b\}$ のセレクション関数である. $S = \{a, b\}, T = \{b, c\}$ のとき $f(g(S \cup T)) = f(g(S)) \cup f(g(T))$ を満たさない. したがって, $f \circ g$ はセレクション関数でない. □

補題 8 f がランキング関数であり, g が逐次等価性を満たすフィルタリング関数であるが度数関数でないとき, $f \circ g$ は必ずしも分配等価性および逐次等価性を満たさない. また, $f \circ g$ は必ずしもセレクション関数およびランキング関数でない.

◀ 証明 ▶ $T = \{a, b, c\}$ とする. 表 9 に示す f は 1 ランキング関数であり, g は逐次等価性を満たすが, 度数関数ではない. $S = \{a, b\}, T = \{a, b, c\}$ のとき $f(g(S \cup T)) = f(g(S)) \cup f(g(T)), f(g(S \cup T)) = f(g(S \cup f(g(T))))$ を満たさない. したがって, $f \circ g$ はセレクション関数でもランキング関数でもない. □

補題 9 全順序 R に対し, f, g が R に対するランキング関数であるとき, $f \circ g$ は必ずしも分配等価性を満たさない. また, $f \circ g$ は必ずしもセレクション関数でない.

◀ 証明 ▶ $T = \{a, b\}$ とする. 表 10 に示す f, g は同じ全順序に対する 1 ランキング関数である. $S =$

表 13 合成関数の性質

$f \backslash g$	DE		SE, PE	
	Selection	¬Cardinality	Cardinality	
			Ranking with R	Ranking with $R' (R \neq R')$ (cardinality n')
Selection	Selection, DE, SE, PE	-	-	-
SE, PE, ¬Cardinality	SE, PE	-	-	-
Ranking with R (cardinality n)	SE, PE	-	Ranking, SE, PE	$\begin{cases} \text{Ranking, SE, PE} & (if\ n \geq n') \\ - & (if\ n < n') \end{cases}$

$\{a\}, T = \{b\}$ のとき $f(g(S \cup T)) = f(g(S)) \cup f(g(T))$ を満たさない。したがって、 $f \circ g$ はセレクション関数でない。 □

補題 10 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、 $n \geq n'$ のとき、 $f \circ g$ は必ずしも分配等価性を満たさない。また、 $f \circ g$ は必ずしもセレクション関数でない。

《 証明 》 $T = \{a, b\}$ とする。表 11 に示す f, g は異なる全順序に対する 1 ランキング関数である。 $S = \{a\}, T = \{b\}$ のとき $f(g(S \cup T)) = f(g(S)) \cup f(g(T))$ を満たさない。したがって、 $f \circ g$ はセレクション関数でない。 □

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、 $n < n'$ のとき、 $f \circ g$ は必ずしも分配等価性および逐次等価性を満たさない。また、 $f \circ g$ は必ずしもセレクション関数でない。

《 証明 》 $T = \{a, b, c\}$ とする。表 12 に示す f は 1 ランキング関数であり、 g は 2 ランキング関数である。 $S = \{a, b\}, T = \{a, b, c\}$ のとき $f(g(S \cup T)) = f(g(S)) \cup f(g(T)), f(g(S \cup T)) = f(g(S \cup f(g(T))))$ を満たさない。したがって、 $f \circ g$ はセレクション関数でない。 □

等価性を満たすフィルタリング関数、セレクション関数、ランキング関数全ての組合せについて、合成関数の性質を表 13 に示す。表 13 には、上記の補題で明らかになった結果に加えて、表 1 と表 2 に示した性質も一緒にまとめた。各項目は、分配等価性 (DE)、逐次等価性 (SE)、並列等価性 (PE) を満たすかどうか、あるいはセレクション関数 (Selection)、ランキング関数 (Ranking) であるための条件を満たすかどうかを記す。また、性質名や関数名が記されていないものは、必ずしもその性質や関数となるための条件を満たさないことを意味する。“-” は、全ての性質あるいは関数となるための条件を必ずしも満たさないことを意味する。

表 13 に示す結果より、逐次等価性を満たすフィルタリング関数を度数関数であるものとそうでないものに分類したことで、これまでは必ずしも満たさないとされていた性質を満たす場合があることがわかった。例

えば、表 1 では、逐次等価性を満たすフィルタリングを合成する場合、逐次等価性を必ずしも満たさないことが示されている。しかし、それぞれ異なる全順序が与えられたランキング関数で、先に適用するフィルタリング関数の度数よりも後に適用するフィルタリング関数の度数の方が大きくなければ、逐次等価性を常に満たすことが明らかになった。

また、逐次等価性を満たすが度数関数ではないフィルタリング関数と逐次等価性を満たすフィルタリング関数とを合成する場合、本稿で扱う枠組みでは、セレクション関数・ランキング関数であるための条件や等価性を必ずしも満たさないことがわかった。

さらに、表 2 では、合成関数がセレクション関数またはランキング関数となるかどうかについてのみ示したが、表 13 により、等価性を満たすかどうかも明らかになった。例えば、これまでは、セレクション関数の後にランキング関数を合成する場合、度数関数であるための条件を満たさないでランキング関数でないということのみ明らかにし、等価性を満たすかどうかについては考慮していなかった。しかし、表 13 より、ランキング関数でなくても逐次等価性を満たすことが明らかになった。

4. 考 察

実際のフィルタリングシステムでは、ユーザの嗜好やフィルタリングの要求を獲得するために、フィルタリング言語を用いて記述するものが多数存在する。本章では、いくつかのユーザ要求を TQL⁴⁾ とフィルタリング SQL⁸⁾ という言語で記述し、本稿で示した性質から、各要求の実現方法について考察する。TQL とフィルタリング SQL とは、データベースへの問合せ言語である SQL をフィルタリングのために拡張した言語である。

以下、分配等価性を満たすフィルタリング、すなわち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含む逐次等価性を満たすフィルタリングについて、それぞれ考察する。

4.1 セレクションの合成

図 3 の記述は、“Mary” の “News” に関する要求を満たし、かつキーワード “Weather” を含むデータが欲

```
m IN Mary.News
AND m.words = {'Weather'}
```

図3 TQLによるユーザ要求の記述例

```
EXTRACT *
FROM A_Broadcast
WHERE best(100)
AND best(30, Broadcast_Time, DESC)
```

図4 フィルタリングSQLによるユーザ要求の記述例1

```
EXTRACT *
FROM A_Broadcast
WHERE best(3)
AND GENRE = News
```

図5 フィルタリングSQLによるユーザ要求の記述例2

しい」というユーザ要求を表し、「Mary」の「News」に関する問合せで抽出されたデータを抽出する」というセレクションと「キーワード「Weather」を含むデータを抽出する」というセレクションを組合せた手法で実行される。したがって、表13より、この手法は分配等価性を満たすため、一括処理、分配処理、逐次処理、並列処理のうちいずれの処理方法で実行しても一貫したフィルタリング結果が得られる。また、合成関数がセレクション関数であるための条件を満たすことから、ある $X \subset T$ に対して $f(S) = S \cap X$ という一つの論理演算により実装できる。一般に、適用する関数の数が多いほど処理にかかる時間が増すため、両者の演算が比較的単純であり、受信機が2手法を同時に行えるだけの処理能力をもつならば、一つの演算処理へと簡素化することで処理の効率化が図れる。

4.2 ランキングの合成

図4の記述は、「A_Broadcast」から放送されたデータのうち、重要度の高さが100位以内で、放送日時の最新度が30位以内のデータが欲しい」というユーザ要求を表し、「重要度の高さが100位以内のデータを抽出する」というランキング f と「放送日時の最新度が30位以内のデータを抽出する」というランキング g を組合せた手法で実行される。 f はデータの重要度、 g は送信日時を基準とした全順序がそれぞれ与えられている。また、 f の度数は100、 g の度数は30である。ゆえに、表13より、 f の後に g を実行する場合は、いずれの等価性も必ずしも満たさないことから、一括処理の結果と、分配処理、逐次処理、並列処理の結果が必ずしも

```
EXTRACT *
FROM A_Broadcast AS X
WHERE PREFER TV_program
AND NOT EXISTS
(EXTRACT *
FROM A_Broadcast AS Y
WHERE Y.RE = X.ID
AND GENRE = TV_program)
```

図6 フィルタリングSQLによるユーザ要求の記述例3

```
EXTRACT *
FROM A_Broadcast AS X
WHERE best(100)
AND NOT EXISTS
(EXTRACT *
FROM A_Broadcast AS Y
WHERE Y.RE = X.ID)
```

図7 フィルタリングSQLによるユーザ要求の記述例4

等価とはならない。逆に、 g の後に f を実行する場合は、逐次等価性を満たすことから、一括処理、逐次処理、並列処理の結果が等価となり、環境に応じて自由に処理方法を変更しても一貫したフィルタリング結果が保証できる。さらに、合成関数がランキング関数であるための条件を満たすことから、ランキングによる処理を一度行う方法へと変換できる。

4.3 セレクションとランキングの合成

図5の記述は「A_Broadcast」から放送されたデータのうち、重要度がベスト3で、ジャンル「News」に属するデータが欲しい」という要求を表し、「最も重要度の高い3個のデータを抽出する」というランキング f と「ジャンル「News」に属するデータを抽出する」というセレクション g を組合せた手法で実行される。したがって、表13より、 g の後に f を実行した場合、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。一方、 f の後に g を実行した場合、いずれの性質も満たさないため、一括処理の結果と、分配処理、逐次処理、並列処理の結果が必ずしも等価とならない。ゆえに、処理の途中で処理方法を変換できないため、あらかじめ最適な処理方法を決めておく必要がある。さらに、どちらの手法を先に行って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ため、現在のところ、一つの処理演算に変換する方法は明確になっていない。

4.4 逐次等価性を満たすフィルタリングの合成

図6の記述は、「A_Broadcast」から放送されたデータのうち、ジャンル「TV_program」に属するデータが欲しい。また、できればその情報に関して最新

のものがよい」というユーザ要求を表し、「ジャンル“TV_program”に属するデータを抽出する」というセレクション f と「更新データが存在するデータの評価値を下げてフィルタリングする」という手法 g を組合せた手法で実行される。 g はデータの相関性を考慮した手法であり、特定のデータと一緒にフィルタリングすることでデータの評価を下げる。 g のような手法は、逐次等価性を満たすが¹¹⁾、度数関数ではない。したがって、表 13 より、 g の後に f を実行する場合は、いずれの等価性も必ずしも満たさないため、全ての処理方法によるフィルタリング結果が必ずしも等しくはならない。一方、 f の後に g を実行する手法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しくなることを保証できる。ただし、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ことから、一つの演算処理に変換する方法が明確ではない。

図 7 の記述は「“A_Broadcast” から放送されたデータのうち、重要度がベスト 100 のデータが欲しい。また、できればその情報に関して最新のものがよい」というユーザ要求を表し、「重要度の高さが 100 位以内のデータを抽出する」というランキング f と「更新データが存在するデータの評価値を下げてフィルタリングする」という手法 g を組合せた手法で実行される。したがって、表 13 より、 f の後に g を実行しても、 g の後に f を実行しても、等価性を必ずしも満たさない。したがって、全ての処理方法によるフィルタリング結果が必ずしも等しくはならない。また、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ことから、一つの演算処理に変換する方法が明確ではない。

5. おわりに

本稿では、等価性を満たすフィルタリング関数、セレクション関数、ランキング関数について、合成関数の性質を明らかにした。特に、逐次等価性を満たすフィルタリングを度数関数であるものとそうでないものに分類したことで、分類前には明らかでなかった性質を明確にできた。また、合成前に満たさない性質を合成後は満たすかどうかについても明らかにしたことで、本稿で扱う性質を広く利用した考察が行える。さらに、本稿で示した体系を実際に複数の手法を組合せた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処理方法へと動的に変更できる。

今後は、本研究で構築した枠組みを実際に適用したフィルタリングシステムを実装する。システムは、受信機の処理能力やメモリ容量、放送データの流量やコンテンツの特徴など、さまざまな環境の要素に応じて、

より処理コストの低い処理方法を動的に実行できるようにする。そして、あらゆる環境における各処理方法のコストを測定・比較し、本研究で構築した枠組みの有効性を示す予定である。

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 21 世紀 COE プログラム（研究拠点形成費補助金）、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) N. J. Belkin and W. B. Croft: “Information filtering and information retrieval: two sides of the same coin?,” *Communications of the ACM*, vol. 35, no. 12, pp. 29–38 (1992).
- 2) T. A. H. Bell and A. Moffat: “The design of a high performance information filtering system,” in *Proc. SIGIR '96*, pp. 12–20 (1996).
- 3) 衛星放送協会ホームページ:
<http://www.eiseihoso.org>.
- 4) D. Goldberg, D. Nichols, B. M. Oki, and D. Terry: “Using collaborative filtering to weave an information TAPESTRY,” *Communications of the ACM*, vol. 35, no. 12, pp. 61–70 (1992).
- 5) 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).
- 6) 西正, 野村敦子: “多チャンネル放送の衝撃,” 中央経済社 (1997).
- 7) Satellite Magazine:
<http://www.satemaga.co.jp>.
- 8) 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).
- 9) R. Sawai, M. Tsukamoto, Y. H. Loh, T. Tera-da, and S. Nishio: “Functional properties of information filtering,” in *Proc. 27th International Conference on Very Large Data Bases (VLDB2001)*, pp. 511–520 (2001).
- 10) 澤井里枝, 塚本昌彦, 寺田努, Loh Yin Huei, 西尾章治郎: “フィルタリング関数の合成とその性質について,” 情報処理学会研究報告 (データベースシステム研究会 2001-DBS-125(2)), vol. 2001, no. 71, pp. 61–68 (2001).
- 11) 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).
- 12) 澤井里枝, 塚本昌彦, 寺田努, 西尾章治郎: “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,” 情報処理学会論文誌: データベース, vol.43, no. SIG12(TOD16) (2002, 掲載予定).