再帰的プログラムの意味論とトレース付きモノイダルカテゴリ

長谷川 真人

再帰的プログラムの表示的および操作的意味論は，プログラミング言語の理論において古典的な話題である．本稿では，近年発展した絡み目のような巡回的な数学的構造のためのカテゴリ論的な枠組みを応用することにより，再帰プログラムの意味論を統一的に扱うことができることを紹介する．特に，領域理論に基づく表示的意味論および，グラフ書き換えシステムにおける巡回的共有グラフによる表現は，この枠組みの具体例として説明することができる．

1 はじめに

プログラミング言語の理論において，再帰的プログラム，すなわち自分自身を呼び出すように書かれたプログラムの数学的基礎の確立は，古くから大きなテーマのひとつであり続けている．この問題に関しては，70年代までに，すでにふたつの本質的なアイデアが提案され，それぞれに大きな成果をあげてきた．ひとつは，Scottらにより創始された，プログラミング言語の表示的意味論およびその数学的基盤である領域理論で，この枠組みでは，再帰的プログラムの意味は「もっとも良い性質を持つ」不動点，すなわち最小不動点として特徴づけられる [14]． 最小不動点モデルは再帰的プログラムについて論ずるための強力な基礎となり，現在に至るまで活発に研究・応用されている．もうひとつは，再帰的プログラムを巡回的なグラフ構造として解釈する方法である．図1に示すように，再帰的プログラムを巡回的グラフとして表現し，その実行をグラフの書き換えとして実現する，というのが，その基本的な発想である．こちらは，Turnerらによりかなり早くから関数型プログラミング言語の有効な実装方法として用いられ [16]，後にグラフ書き換えに基づく計算の理論として定式化されて，さかんに研究されている．

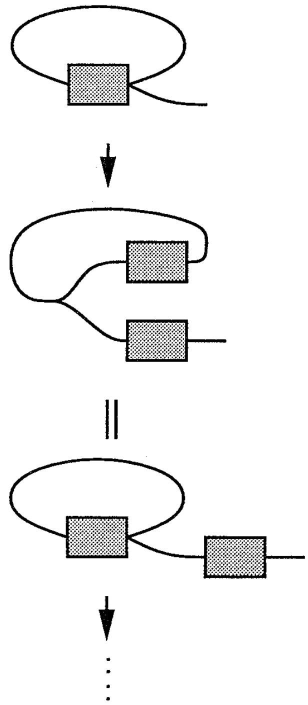

図1 巡回グラフ構造による再帰

しかしながら，これらふたつのアプローチを結び付ける理論的枠組みは，これまで知られていなかった．これ

Semantics of Recursive Programs and Traced Monoidal Categories

Masahito Hasegawa, 京都大学数理解析研究所, Research Institute for Mathematical Sciences, Kyoto University.

コンピュータソフトウェア, Vol.16, No.2 (1999), pp.62–66.
[小論文] 1998年8月3日受付.

は，それぞれの方法論がまるで違うためにあまり丁寧な考察がなされてこなかったことも一因であるが，最大の理由は，これらの方法に共通する数学的な構造がよくわかっていなかったことだと思われる．なぜこれらのアプローチが再帰的プログラムのよいモデルを与えるのか，他の方法では駄目なのかどうか，という基本的な疑問に答えるためにも，再帰的計算の解釈に不可欠な数学的構造を特定する必要がある．

この問題を考えるためのヒントが，近年発展した絡み目 [2]のような巡回的な数学構造のためのカテゴリ論的枠組みから得られた．一般に，再帰的プログラムや絡み目のような巡回的な構造について，構文的に直接推論することは，巡回的でない場合に比べて大変難しい．絡み目の理論の場合，その困難を補う強力な道具が絡み目の不変量で，これは，適当な代数的量をそれぞれの絡み目もしくはその表現に対応させてやることで，絡み目の分類を行うことを可能にするというものである．絡み目の不変量の考え方は，プログラミング言語の研究において，再帰的プログラムについて推論するために最小不動点を用いて表示的意味を与えてやることと，大変よく似ているように思われる．実際，これは単なる比喩ではない．それどころか，再帰的プログラムの意味論を理解するのに有意義な見方をもたらす，というのが，本稿の主張である．Joyal, Street と Verity は，トレース付きモノイダルカテゴリという構造が，絡み目の不変量のような理論を一般的なかたちで展開するために必要かつ十分であることを示した [10]．トレースは，絡み目の不変量などで巡回的な構造の解釈を与えるための鍵になっている部分を抽象化したものである．本稿では，このトレース付きモノイダルカテゴリの枠組みが，再帰的プログラムの統一的な意味論を与えるのに有効であることを示した，筆者らの最近の結果 [5] [6]を報告する．

2 トレース付きモノイダルカテゴリ

モノイダルカテゴリ $\mathbf{C} = (\mathbf{C}, \otimes, I, a, l, r)$ とは，カテゴリ $\mathbf{C}$ と，それに付随するテンソル積またはモノイダル積と呼ばれる関手 $\otimes : \mathbf{C}^2 \to \mathbf{C}$，単位対象と呼ばれる $\mathbf{C}$ の対象 I，テンソル積と単位対象の結合律・単位律に相当し適当な公理を満たす可逆な自然変換 $a_{A,B,C} : (A\otimes B)\otimes C \xrightarrow{\sim} A\otimes(B\otimes C)$，$l_A : I\otimes A \xrightarrow{\sim} A$ および $r_A : A\otimes I \xrightarrow{\sim} A$ の組のことをいう．モノイダルカテゴリで，$a_{B,C,A}\circ c_{A,B\otimes C}\circ a_{A,B,C} = (id_B \otimes c_{A,C})\circ a_{B,A,C}\circ (c_{A,B}\otimes id_C)$ (双線型性) と $c^{-1}_{A,B} = c_{B,A}$(対称性) を満たす可逆な自然変換 $c_{A,B} : A\otimes B \xrightarrow{\sim} B\otimes A$ を持つものを，**シンメトリックモノイダルカテゴリ**と呼ぶ．なお，この定義から対称性を除き，かわりに c^{-1} の双線型性を仮定したより一般的な構造がブレイド付きモノイダルカテゴリ (c はブレイドと呼ばれる) であり，絡み目の不変量の議論において重要であるが，本稿では深く立ち入らないことにする．以上の概念の詳細については [9] [12]を参照されたい．

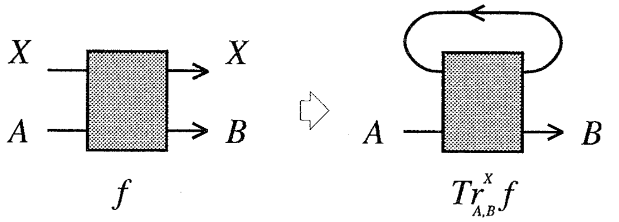

図2 トレース

いま，シンメトリックモノイダルカテゴリ $\mathbf{C}$ が与えられているものとする．このとき，$\mathbf{C}$ 上の**トレース** [10]とは，$\mathbf{C}$ の対象 A, B, X で添字付けられた関数の族

$$Tr^X_{A,B} : \mathbf{C}(A\otimes X, B\otimes X) \to \mathbf{C}(A, B)$$

で，以下の条件を満たすものである．

- (Left/Right Tightening, Sliding) Tr はパラメータ A, B に関して natural であり，また X に関して dinatural である．
- (Vanishing) $Tr^I_{A,B} f = r_B \circ f \circ r^{-1}_A$，また $Tr^{X\otimes Y}_{A,B} f = Tr^X_{A,B}(Tr^Y_{A\otimes X, B\otimes X}(a^{-1}_{B,X,Y}\circ f\circ a_{A,X,Y}))$.
- (Superposing) $Tr^X_{C\otimes A, C\otimes B}(a^{-1}_{C,B,X}\circ(id_C\otimes f)\circ a_{C,A,X}) = id_C \otimes Tr^X_{A,B} f$.
- (Yanking) $Tr^X_{X,X} c_{X,X} = id_X$.

トレースは，図2のように巡回構造をつくり出す演算ととらえるとわかりやすい．この表記法を用いて，上記の条件は，図3のように簡潔に表現することができる．

トレースを持つシンメトリックモノイダルカテゴリを，**トレース付きモノイダルカテゴリ**と呼ぶ．代表的な例として，任意の体の上の有限次元線型空間と線型写

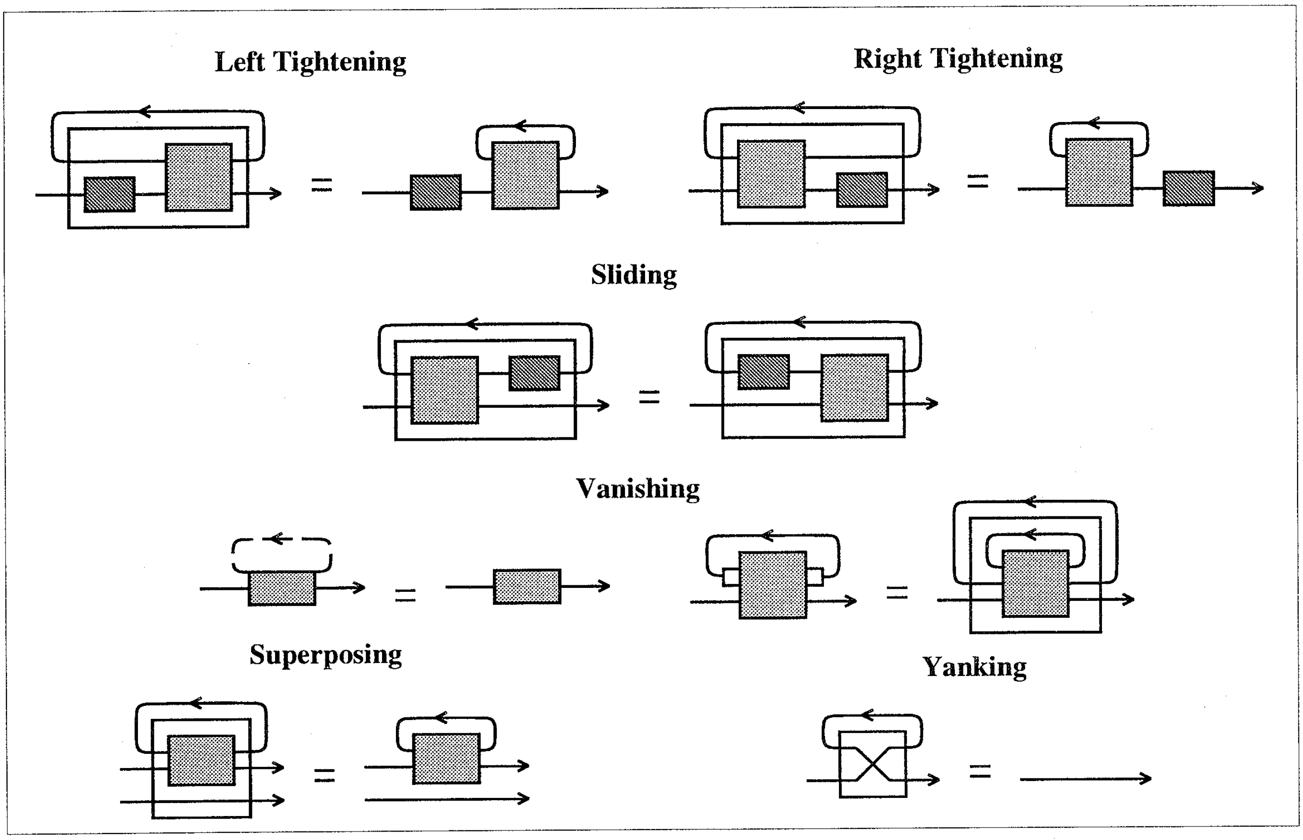

図3 トレースの公理

像のなすカテゴリは，トレース付きモノイダルカテゴリになっている(トレースは行列の対角和として与えられる)．よく似た例として，対象が集合，射が集合間の二項関係で与えられるカテゴリ **Rel** もトレースを持つ．集合 A, B, X と， $A \times X$ と $B \times X$ の間の二項関係 R が与えられたとき， $Tr^{X}_{A,B}R$ は，以下のように特徴づけられる A と B の間の二項関係である．

$$(a,b) \in Tr^{X}_{A,B}R \;\Leftrightarrow\; \exists x \in X\ ((a,x),(b,x)) \in R$$

その他の例については [10] [6]などを参照されたい．

なお，トレースの概念は，一般にブレイド付きモノイダルカテゴリに対して定義でき ([10]を参照)，前述したように絡み目の不変量を議論するのに重要な役割を果たすが，本稿では，再帰的プログラムの解釈に必要かつ十分な，シンメトリックな場合だけに限定した定義を与えた．

3 トレースと不動点

はじめに述べた再帰的プログラムへのふたつのアプローチを，トレース付きモノイダルカテゴリの枠組みによって，絡み目不変量の場合と本質的に同様の方法で，統一的に理解することが可能である．

3.1 表示的意味論

通常の表示的意味論において用いられているのは，テンソル積が直積になっているような場合である．そのような状況では，再帰的プログラムの解釈に用いる不動点演算子が存在することと，トレースが存在することがまったく同値であることが， Martin Hyland と筆者によって独立に証明された [5]：

定理 1 (**Hyland**/**長谷川**) テンソル積が直積であるようなモノイダルカテゴリ **C** においては，以下の条件は同値である．

1. **C** はトレースを持つ．
2. **C** は dinaturality と diagonal property を満たす不動点演算子を持つ．
3. **C** は naturality と Bekic property を満たす不動点演算子を持つ． □

条件 2, 3 の詳細は省略する ([5]を参照のこと)．ここでは，これらの条件は，従来の再帰的プログラムのモデルのほとんどすべてについて成立しているとい

うことを指摘するにとどめておく．とくに，領域理論の知られているすべてのモデルはトレースを持つ．このことは，最小不動点演算子がこの定理の条件2および3を満たすことからわかる．より直接的には，たとえば最小元を持つ ω 完備半順序と連続関数のカテゴリでは，$f=\langle f_0, f_1\rangle : A\times X \to B\times X$ について，$Tr^X_{A,B}f : A\to B$ は以下のように与えられる．

$$(Tr^X_{A,B}f)(a) = f_0(a, \bigcup\{(\lambda x.f_1(a,x))^n(\bot_X)\})$$

実際，領域理論の多くのモデルでは，条件2, 3を満たす不動点演算子はただひとつしか存在しないことが知られており [15]，そのような場合にはトレースは最小不動点演算子を特徴づけている．また，最小不動点によらないモデル(たとえば長谷川立による組み合わせ論的な不動点の研究 [7]を参照)であっても，ほとんどの場合にトレースが再帰的プログラムの解釈を特徴づけていることがわかっている．

3.2 巡回的グラフ書き換え系

一方，再帰的プログラムの巡回的グラフ表現がトレース付きモノイダルカテゴリを与えていることは，たとえば図1を見れば直感的には明らかであろう．正確には、巡回的グラフ書き換え系のモデルを与えるには，テンソル積が必ずしも直積になっていないような，表示的意味論よりももう少し一般化した場合を考える必要があるが [6]，筆者は，そのような場合についても，トレースの存在が再帰的プログラムのモデルを与えることを示している：

定理 2 (長谷川) テンソル積が直積であるようなモノイダルカテゴリから，トレース付きモノイダルカテゴリへのシンメトリックモノイダル随伴が存在するとき，このトレース付きモノイダルカテゴリはdinaturalな不動点演算子を持つ． □

術語の詳細およびこのような数学構造を用いたグラフ書き換え系の意味論については，[6]を参照されたい．

3.3 新しいモデルの意義

こうして得られた新しい再帰的プログラムのモデルでは，従来のモデルよりも精密に再帰的プログラムの分類を行うことができ，より緻密にプログラムの性質を調べる基礎となることが期待される．たとえば，巡回的ラムダ計算 [3]において，図4に示した不動点演算子のふたつの実装は，異なるふるまいを示すことが知られている(図5)．関数型言語の実現で用いられる必要呼び戦略 (call-by-need strategy) では，左の実装が関数クロージャのかたちをした値にのみ不動点演算子として働くのに対し，右の実装は任意の値について有効である [11]．このことは，実際に我々の新しいモデルにおいてこれらの意味を与えてやることにより確認できる．具体的な例として，先にのべたトレース付きモノイダルカテゴリ **Rel** は，集合と関数のカテゴリからのシンメトリックモノイダル随伴を持ち，型付き巡回的ラムダ計算のモデルを与えているが，**Rel** におけるこれらの演算子の解釈は，左のほうが二項関係 $R : X\to X$ に $(x,x)\in R$ を満たすような任意の $x\in X$ を対応させるのに対し，右のほうは $S=\{x\in X \mid \exists s\in S\ (s,x)\in R\}$ を満たすような任意の S の元を対応させる．前者は R が関数，すなわち関数クロージャの解釈の場合に通常の意味での(関数としての)不動点を与えるが，後者は，任意の値の解釈に対し，関係としての不動点を与えていることがわかる [5] [6]．

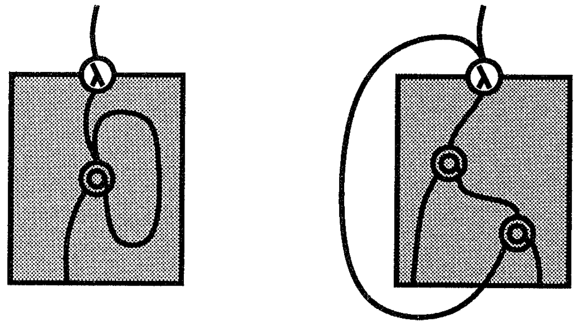

図 4 巡回的ラムダ計算における不動点演算子

4 おわりに

本稿では，再帰的プログラムの意味論が，トレース付きモノイダルカテゴリという，絡み目のような巡回的数学構造のためのカテゴリ論的枠組みの中で展開できることを報告した．とくに強調しておきたいのは，このアイデアが，単に知られていた事柄を説明しなおすにとどまらず，新たな再帰的プログラムのためのモデルの発見につながっているということである．この研究に関連して，トレース付きモノイダルカテゴリを基礎に置いた，

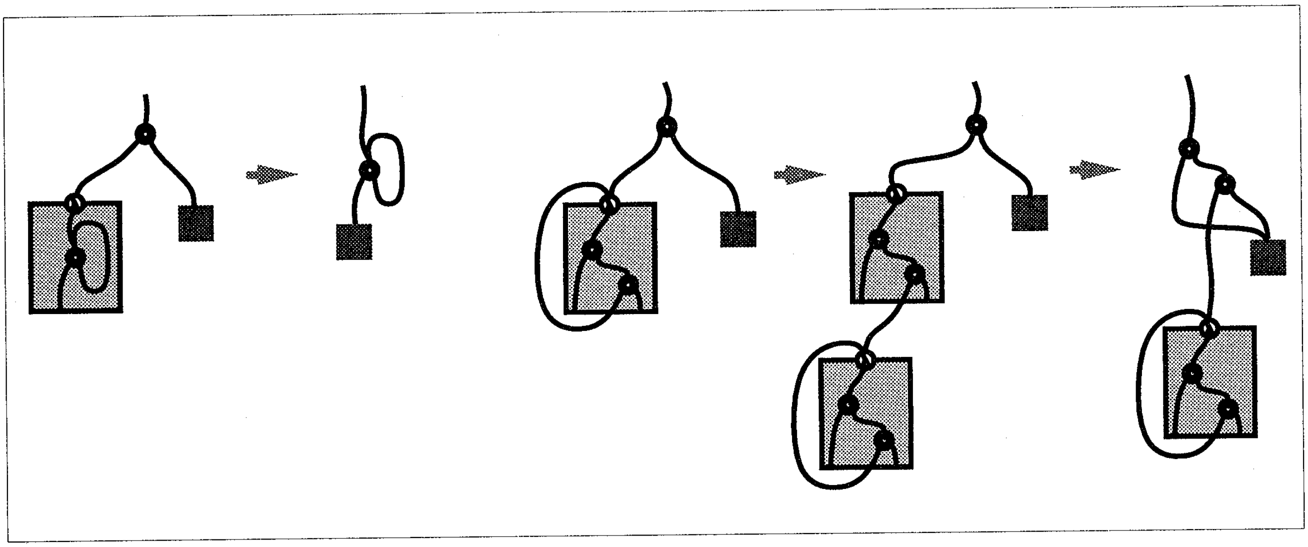

図 5 不動点演算子のふるまいの例

再帰的または巡回的なプログラムの構造とそのための数学理論の研究が，さまざまな側面から進められていることを記しておく (Abramsky 他 [1]; Blute 他 [4]; 三好 [13]; Jeffrey [8]).

謝辞

本研究に関して，Martin Hyland, John Power 両氏から有益なコメントと励ましを頂いた．

参考文献

[1] Abramsky, S., Blute, R. and Panangaden, P. : Nuclear and Trace Ideals in Tensored *-Categories, to appear in *J. Pure Appl. Algebra.*

[2] Adams, C. C. : *The Knot Book*, W. H. Freeman, 1994. 邦訳：「結び目の数学」(金信泰造訳), 培風館, 1998.

[3] Ariola, Z. M. and Blom, S. : Cyclic Lambda Calculi, *Proceedings, Theoretical Aspects of Computer Software*, Springer LNCS 1281 (1997), pp. 77–106.

[4] Blute, R. F., Cockett, J. R. B. and Seely, R. A. G. : Feedback for Linearly Distributive Categories: Traces and Fixpoints, submitted for publication.

[5] Hasegawa, M. : Recursion from Cyclic Sharing: Traced Monoidal Categories and Models of Cyclic Lambda Calculi, *Proc. Typed Lambda Calculi and Applications*, Springer LNCS 1210 (1997), pp. 196–213.

[6] Hasegawa, M. : *Models of Sharing Graphs (A Categorical Semantics of Let and Letrec)*, PhD thesis, ECS-LFCS-97-360, University of Edinburgh (1997); a revised version will be published as a volume of Distinguished Dissertation Series, Springer-Verlag, 1999.

[7] Hasegawa, R. : The Generating Functions of Lambda Terms, *Proc. Combinatorics, Complexity, Logic (DMTCS '96)*, Springer (1996), pp. 254–263.

[8] Jeffrey, A. : Premonoidal Categories and a Graphical View of Programs, technical report 98-004, School of Computer Science, DePaul University (1998); available from `http://klee.cs.depaul.edu/premon/`.

[9] Joyal, A. and Street, R. : Braided Tensor Categories, *Adv. Math.* 102 (1993), pp. 20–78.

[10] Joyal, A., Street, R. and Verity, D. : Traced Monoidal Categories, *Math. Proc. Cambridge Philos. Soc.* **119** (1996), pp. 447–468.

[11] Launchbury, J. : A Natural Semantics for Lazy Evaluation, *Proc. 21st ACM Symposium on Principles of Programming Languages* (1993), pp. 144–154.

[12] Mac Lane, S. : *Categories for the Working Mathematician (Second Edition)*, Graduate Texts in Mathematics 5, Springer-Verlag, 1998.

[13] Miyoshi, H. : Rewriting Logic for Cyclic Sharing Structures, *Proc. 3rd Fuji International Symposium on Functional and Logic Programming* (1998), pp. 167–186.

[14] Scott, D. : A Type-theoretical Alternative to CUCH, ISWIM, OWHY, privately circulated memo, Oxford (1969); published in *Theoret. Comput. Sci.* 121 (1993), pp. 233–247.

[15] Simpson, A. K. : A Characterisation of the Least-fixed-point Operator by Dinaturality. *Theoret. Comput. Sci.* 118 (1993), pp. 301–314.

[16] Turner, D. A. : A New Implementation Technique for Applicative Languages, *Software — Practice and Experience* 9 (1979), pp. 31–49.