Title	AC規則を含む項書換え系の停止性について
Author(s)	中野，賢司
Citation	
Issue Date	1997-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1018
Rights	
Description	Supervisor:外山 芳人，情報科学研究科，修士

AC 規則を含む項書換え系の停止性について

中野賢司

北陸先端科学技術大学院大学 情報科学研究科

1997 年 2 月 14 日

キーワード： AC-TRS, 停止性、単純化順序、AC 適合性、積み上げ化、再編化順序、SDL 順序.

項書換え系 (Term Rewriting System, TRS) とは、項の書き換えによって計算の過程を表す数学的な計算モデルである。TRS は、等式を左辺から右辺への書換え規則と見なすことにより、自然な計算モデルが得られる。そのため、TRS は関数型言語、代数的仕様のような等式に基づくプログラミングや、等式論理による定理自動証明やプログラムの変換・検証などを研究する上で重要な役割を果たしている。

計算モデルとしての TRS の重要な性質として、合流性と停止性があげられる。このうち、停止性は書き換えによる計算が必ず終了することを示す。TRS をプログラムと見なせば、停止性が保証されている TRS は、任意の入力に対し必ず出力を返すことを保証するので、停止性は重要な性質である。

しかし、TRS の停止性は一般には決定不能である。そのため、TRS が停止性を持つための十分条件についての研究が進められている。TRS の停止性を示す方法として、項に整礎な順序をつけて証明する方法が用いられている。順序として意味論的順序と構文論的順序の 2 つがある。意味論的方法は、書き換え規則の項に整礎な領域への対応を与えることにより、停止性を証明する方法である。この方法は、領域の決定に人間の経験と直観に依存する面が大きく、順序の判定の自動化が困難である。一方、構文論的順序による方法は、与えられた関数記号上の順序を項の構造に基づいて項上の順序に拡張する方法である。この方法は、機械的な手続きによる二項の比較が可能であるという利点をもつ。そのため、複雑な項にも対応でき、逆に比較の前提となる関数記号上の順序が項の構造から推定できるなど、自動化に有利な点を持っている。構文論的順序では、単純化順序と呼ばれる諸性質を満たす順序が、停止性の証明に用いられる。そのような順序の代表的な例としては D.Plaisted による部分項経路順序（Path of Subterm Ordering）、N.Dershowitz による再帰経路順序（Recursive Path Ordering）および P.Lescanne による再帰分解順序 (Recursive Decomposition Ordering）などが提案されている。

AC 項書換え系（AC-TRS）は、簡略化の方向が定められない等式 AC を含む項書換え系で、AC は結合律（Associativity）と交換律（Commutativity）を表す等式の集合である。AC-TRS では、簡略化の方向が定められない等式を取り扱う。

しかし、TRS に書換えの方向が定まらない等式が含まれると、等式を無限に適用した書換えが可能となり、通常の TRS のような停止性は満たさない。そのため、AC-TRS では項の等式による同値類上での順序によって停止性を考える必要がある。単純化順序で AC-TRS の停止性の証明するためには、単純化順序を同値類上での比較が行なえるように拡張する必要がある。

２章では、TRS の基本的な定義、および諸性質について述べる。また、TRS の停止性についての従来までの手法について説明する。

３章では、AC-TRS への構文論的順序の適用について、これまでどのような手法が提案されてきたかについて示す。AC-TRS の停止性に関する研究では、順序判定の自動化が容易であるなどの理由で、構文論的順序による方法が広く用いられている。そのなかでも、停止性を示すことが可能な順序として単純化順序がある。しかし、AC-TRS では同値類と同値類との関係の比較について考えなくてはならないため、単純化順序を AC-TRS の停止性の証明に直接用いることはできない。そこで、同値類上での順序に単純化順序を拡張するために AC に関する適合性という性質が提案されている。この性質を満たすために、従来までに平坦化が提案された。

f を AC 関数記号、X, Y を項の多重集合とすると、

$$f(f(X), Y) \rightarrow_{Fl} f(X, Y)$$

平坦化は、内部に連続して表れる AC 関数記号を１つに縮退させる。そして、内部の部分項を多重集合と見なすことによって、交換律、結合律に関する同値類の代表元を得ることができる。この操作により、同値類の比較は同値類の代表元である項の比較に変換されるので、AC 適合性を満たす単純化順序の定義が期待できる。

しかし、平坦化による変形のみでは単純化順序を満たさないことがわかっている。従来の研究では、他にいくつかの変形を組み合わせて順序を定義することにより、単純化順序の性質を実現している。Bachmair らの分配規則による変形での APO（Associative Path Ordering）では関数記号の順序関係に大きな制限が加わる。また、Kapur らの AC 順序による疑似コピーおよび 繰り上げ操作との組合わせでは、APO の関数記号の制限は起こらないが、変形の過程で非決定的な手続きを含んでいるため、コンピューターへの実装の点ではやや難がある。Rubio らによる解釈操作による方法では、項の集合から最大の順序をもつ項を選択する手続きを持つために、基本的に項は全順序であることを仮定している。これらの順序は、平坦化の欠点を補う別の変形を提案し、平坦化と組み合わせて順序を定義しているが、順序に加わる制限が減少するとともに変形自体が複雑化するという傾向が見られる。

４章では、平坦化に代わる新しい変形である積み上げ化について考察する。本研究では、平坦化と異なる視点から AC-TRS に適用する順序を考察するために、平坦化に代わる変形として積み上げ化を提案した。

$$f(f(X),Y) \to_{St} f^2(X,Y)$$

従来の平坦化では、連続で表れた AC 関数記号を 1 個に縮退させるのに対し、積み上げ化では項の深さに関する情報を最外関数の次数によって保存するので、より効果的な比較が期待できる。しかし、平坦化の場合と同様に積み上げ化もそれ単独では単調性を満たさないため、単純化順序とはならない。そこで、別の変形と組み合わせて単純化順序の性質を満たす必要がある。

平坦化では、順序に関連した関数記号が縮退した結果、単調性に矛盾が生じることが知られている。一方、積み上げ化では平坦化のような縮退による矛盾は生じない。しかし、内部に存在した AC 関数記号 fが最外に移動することで、この fの引数でなかった部分項が fの引数となる。ここで、fの次数を部分項への重み付けと見なすと、積み上げ化による fの次数の変化により関数記号と部分項の関係が崩れる。そのため、AC 関数記号による部分項への適正でない重み付けが問題となる。

そこで、AC 適合性を満たす単純化順序を実現するためには、積み上げ化による偏った重み付けに対応する変形を考える必要がある。本論文では、重み付けの偏りを適正にする変形を目指し、その可能性をもった変形による順序を２つ提案した。

１つは、再編化という変形を導入した再編化順序である。再編化では関数記号を部分項に再分配し、項を再構成する。このとき、重み付けが問題となるような部分項を探索し、関数記号の分配を適正に行なうことで、積み上げ化による重み付けの偏りを解消する。再編化順序は、推移律を除く他の性質は満たしているが、場合分けの多様さにより推移律の証明が困難である。

もう１つは、分離操作、および繰り上げ化という変形を導入した SDL 順序である。重み付けの偏りが矛盾の原因であるが、全ての項がその影響を受けるわけではない。そこで重み付けが起こると支障のある部分項の条件を考察する。そして、再編化とは逆に、該当する部分項を元の項から分離したり、重み付けの影響を受けない形に変形するなどして、重み付けの偏りを解消する。しかし、SDL 順序は単調性に反例が生じる。

本論文では、積み上げ化の問題点を明らかにし、再編化順序、SDL 順序を提案した。また、これらの順序が問題となる条件を明らかにし、その原因と改良方法を考察した。