項書き換え系におけるビューの変更とプログラム実行の視覚化

小川 徹, 田中 二郎

筑波大学 電子・情報工学系

1. はじめに

我々の作成した CafePie[1,2]は,代数的仕様記述 言語 CafeOBJ[3]をビジュアルプログラミング[4] として扱うためのシステムである. CafeOBJ 言語 の一側面である項書き換え系は,項の書き換えの みで計算が進む計算モデルであり,その形式の単 純さと優れた代数的意味論が特徴である.

今回,我々は,項の構成要素である関数の視覚的 部分(ビュー)を編集可能にすることで,項全体

のビューを再定義できる枠組みを提 案する.新しく定義したビューはプ ログラムの編集や実行に用いること ができる.CafePie 上にビューのカ スタマイズ機構を実装した.

2. CafePie の概要

CafePie は, CafeOBJ 言語におけ る一つのモジュールを視覚化 / 編集 し,項書き換え系の視覚化によるプ ログラム実行を実現している.**モジ ュール**は,ソート / 演算 / 変数 / 等 式から構成される(演算は関数を, 等式は項の書き換え規則を指す).

図1に CafePie の実行画面を示す. 画面中央に Natural Number に足し 算を定義したモジュール SIMPLE-NAT を表示している.図1右にある のが,「2+3」を実行したときの項書き換えのトレ

ース結果である.

アイコンを用いたプログラムの作成:ファイルから ロードすることで, CafeOBJ のモジュールを自動 的に視覚化する(図1).図1左にある基本アイコ ンを使用してモジュールの必要な部分を編集する. モジュールの編集は,アイコンのドラッグ&ドロ ップで行うことができる. 項書き換えによるプログラムの実行:プログラムと 同じ枠内にゴールとなる項を作成する.作成した ゴールを左上部にあるモジュールラベルにドラッ グ&ドロップすることで,プログラム実行である 項書き換えのトレースが開始される.実行が開始 されると,CafePie はゴールが変化していく様子を アニメーションによって表示する.実行後は, CafePie は図1 右のようにトレース結果を帯状に 表示する.



図 1 CafePie の画面のスナップショット

3. 項におけるビューのカスタマイズ

CafePie では,図1右のトレース結果にあるよう に,「2+3」という Natural Number の項のモデ ルを木構造というデフォルトのビューで視覚化す る.ここでは,変数名や演算名をノードで,演算 の引数関係をエッジで表現している.同様に,

push(E3, push(E2,push(E1, empty))) というスタックの項を視覚化すると,図2 左のよ うなデフォルトのビューが与えられる.

View Alteration and Visualization of Program Execution in Term Writing System Tohru Ogawa and Jiro Tanaka

Institute of Information Sciences and Electronics, University of Tsukuba



図 2 スタックの視覚化

このデフォルトの木構造ビュー(図 2 左)を積み 木ビュー(図 2 右)に変更する場合を考えてみる.

新しいビューの作成: 各演算のビューを編集するこ とで,項全体のビューが決定する.図2 左のスタ ックに含まれる演算は empty と push である.項 が木構造ビューで表示される場合,演算 push のビ ューは 図3 左のようになる.演算 push 上からポ ップアップメニューを呼び出すことで,ビュー作 成ウインドウが左側に表示される.そのあとで, ビュー作成ウインドウ中に新しいビューを作成す る(図3右).



図 3 関数 push のビューの作成

ビューの切り替え:演算上に作成したビューは,自 由に変更することができる.図3のように,演算 pushには2つのビュー(木構造ビューと積み木ビ ュー)が定義されている.新しく作成した積み木 ビューに切り替えたい場合には,図3右の上から ポップアップメニューを呼び出すことで行う.ビ ューの切り替えが行われると,演算 push は図3右 だけが表示される.そのあとで,モジュール内の スタックのビューは,図2から図4へと切り替わ る(この場合は,演算 empty に対するビューも作 成して,同時に切り替えている).また,逆に図4 から図2にビューを戻すことも可能である.

演算の引数は,項における部分項に対応している.ビューの作成で行っていることは,各演算に 対する図形の変更と,演算とその引数との位置関係の変更である.また,必要に応じて矩形や楕円, イメージといった図形を併用することで,さまざ まなビューを作成することができる.

4. ビューを用いた実行表示

ユーザによってカスタマイズされたビューは, 書き換え規則の上でそのまま扱うことができ,また,項の書き換え過程の表示にも使用可能である.

等式は,項の書き換え前を左辺項で,書き換え 後を右辺項で与えることで,書き換え規則を表現 する.等式の中で現れている各演算のビューを切 り替えることで,書き換え規則の上にビューの変 更を反映させることも可能である.

項書き換え過程という実行の表示についても同 じことが言える.実行の表示は,与えられた項(ゴ ール)の変化によって表現される.ゴールの中に 現れる演算はプログラム中に与えられており,一 意に対応付けられている.対応する各演算のビュ ーを切り替えることで,ゴール全体のビューも容 易に変更することができる.スタックでは,図4 のような,ビューを用いた実行表示ができる.



図 4 ビューを用いた実行表示

参考文献

[1] T. Ogawa and J. Tanaka, CafePie: A Visual Programming System for CafeOBJ, In *Cafe: An approach to Industrial Strength Algebraic Formal Methods*, Elsevier Science, pp.145-160, 2000.

[2] 小川 徹, 田中 二郎, ドラッグ&ドロップを用 いたビジュアルプログラミングシステム, *情報処 理学会論文誌 プログラミング*, 2002, (to appear).

[3] R, Diaconescu and K. Futatsugi, *CafeOBJ Report*, World Scientific, 1998.

[4] B. A. Myers, Taxonomies of Visual Programming and Programming Visualization, *Journal of Visual Languages and Computing*, 1(1): pp.97-123, 1990.