

2M-4

プログラム可視化システムの概要

市川 至, 小野越夫, 毛利友治

富士通株式会社

1.はじめに

ソフトウェアの再利用や、設計とプログラムの一貫性の確認を支援するために、そのソフトウェアの動作を理解することが必要となってきた。従来は、文字情報を用いた方法がとられていたが、ワークステーション技術の発展により、図形情報を有効に使用できるようになってきており、図形的表示によるプログラムの構造の理解や、アニメーション表示による動作の理解を促進する研究がなされている[1]。

このひとつのアプローチとして、ビジュアル・プログラミング環境がある。これは、図形情報を利用してプログラミングを行うことをめざしたものであり、図形的なプログラミング言語や仕様記述言語などが用いられる。例えば、PegaSys[2]は、図形を形式的なプログラム設計書とするものである。また、別のアプローチとしては、VIPS[3]やPROEDIT2[4]のようなビジュアル・デバッガがあげられる。これは、プログラム動作をプログラミング言語レベルの抽象度の図を用いて表示するものである。

我々のプログラム可視化システムは、ソフトウェアの再利用や、設計とプログラムの一貫性の確認を支援するために、設計段階で用いられる抽象度を持った図形を用いて、プログラムの実行動作を可視化するものである。この図形の抽象度は、PegaSysなどで使用されるような図形と同じであり、VIPS、PROEDIT2のものよりも抽象度が高いと考えている。さらに、プログラム可視化システムは、複数の異なったビュー(View, 視点)からの同時の可視化をサポートして、人間の理解をより促進する。

本発表では、プログラム可視化システムの概要について発表する。

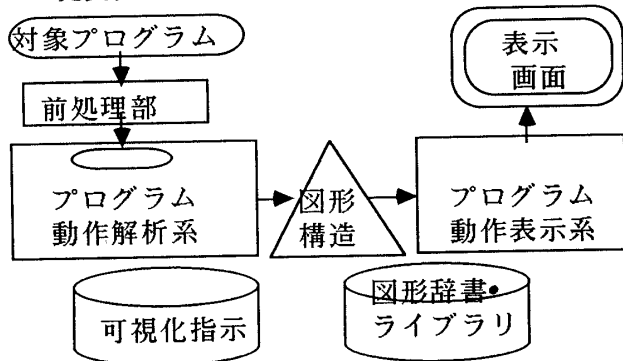


図 1. システムの基本構成

2.プログラム可視化システムの基本構成

我々は、論理型オブジェクト指向言語ESP[5]を対象プログラムの言語として、プログラム可視化システムを開発している。この、プログラム可視化システムの基本構成の概要を図1.に示す。

システムの処理は、前処理部、プログラム動作解析系、図形構造、プログラム動作表示系からなる。このほか、可視化される対象プログラムに対して、可視化のための知識として、可視化指示、図形辞書・ライブラリが与えられる。

可視化システムは、前処理部によって処理された対象プログラムの実行動作を、プログラム動作解析系において検出・解析し、表示図形のモデルである図形構造に対する可視化動作を合成し、プログラム動作表示系によって表示画面上のアニメーションを行なう。

2.1 プログラムと図形との対応

プログラムと図形との対応は、「可視化指示」により定義される。可視化指示は、動作の対応をとる「可視化命令」と、要素の対応をとる「写像定義」からなる。

可視化命令は、プログラム中の実行動作列 S と図形構造での可視化動作 A との対応を、Hoarの公理形式「 $P \{S\} Q, A$ 」で定義したものである。ここで、 P, Q はそれぞれ、 S の実行の直前、直後に成立すべき、プログラム状態に関する陳述(条件式)であり、その意味は「 S の実行前に P が成立し、実行後に Q が成立するならば、 A が行なわれる」と解釈する。

写像定義は、プログラム中の要素を表す表現式 p と、図形の要素を表す表現式 g とを「 $p \rightarrow g$ 」の形で定義したものである。

2.2 基本アーキテクチャ

プログラム動作解析系では、対象プログラムの実行動作列を検出し、可視化命令で定義された実行動作列が実行されると、図形構造に対する可視化動作を合成する。対象プログラムの実行動作列の検出は、いくつかのプログラム上の点の制御の通過によって観測する。検出方法として、我々は、「トレーサ埋め込み方式」を採用した。

トレーサ埋め込み方式は、実行情報を収集するソフトウェア・プローブとしてトレーサを対象プログラムに埋め込み、実行動作を検出するものである。ここでは、「可視化のために検出されるプログラムの動作は、全体の動作の一部で十分である」および「埋め込んだトレーサにより対象プログラムの動作が変化しない」と仮定している。

可視化命令で定義される動作列を検出するのに、必要かつ十分なトレーサを埋め込む処理を行なうのが、

「前処理部」である。前処理部でどのようなトレーサを埋め込んだかについての情報は、プログラム動作解析部に伝えられる。

2.3 図形構造

我々のプログラム可視化システムは、多次元・階層的なビューをサポートし、可視化された表示画面では、複数のウィンドウにそれぞれのビューにあたる図が、同時にアニメーション表示される。このために、プログラム可視化システムには「図形構造」と呼ばれる表示モデルが内部に用意されている。

可視化された表示画面の図形は、単位的な図形である「絵素」と他の図形から合成される図形である「設計図」からなる階層的な概念的構造を各ビューごとに持っている。この概念的構造を図形オブジェクトによるDAG(Directed Acyclic Graph)構造で実現したものが、図形構造である。

図形辞書・ライブラリは、絵素についての定義が格納される「絵素辞書」と、部分設計図、設計図の定義が格納される「設計図ライブラリ」からなり、図形構造の各図形オブジェクトについての、形状、操作、構成の定義がなされている。

3. 実験システム

論理型オブジェクト指向言語 ESPプログラムを対象とする、プログラム可視化システムをPSIワークステーション上に試作し、可視化実験をおこなった。

実用規模のプログラムとして、並列論理型言語実行系のKL1ソフトウェア・シミュレータ[6]について可視化をおこなった。なお、プログラム可視化システムの試作ソフトウェアのソースのサイズは約76kB、対象プログラムのソースのサイズは約18kB、必要とした可視化指示の個数28個、表示される設計図11枚、埋

め込まれたトレーサ総数69個であった。

図2に、その可視化実行例を示す。

4. まとめ

ESPプログラムの動作および構造を可視化する、プログラム可視化システムの試作を行なった。可視化および多重化・階層化ビューのサポートが、プログラムの動作理解に有用であることが判明した。

可視化指示および可視化システムの正当性が保証されていれば、異常な表示があった場合に、それは対象プログラムの異常となり、故障の発見にも使用できる。ただし、現在のところ、可視化指示の正当性の確認は手作業である。

プログラム設計システムなどとの連携により、正当な可視化指示を自動生成すること、および、対象言語を並列処理言語へ拡張することが、将来の課題である。

謝辞

本研究は第5世代コンピュータプロジェクトの一環として行われた。御支援頂きました長谷川ICOT第1研究室室長を初めとする方々に、深く感謝いたします。

参考文献

- [1] Grafton, G.B. and Ichikawa, T.: "Visual Programming," IEEE Computer, special issue for visual programming, pp.6-9 (August 1985).
- [2] Moriconi, M. and Jare, D. F.: "Visualizing Program Design Through PegaSys," IEEE Computer, special issue for visual programming, pp.72-83 (August 1985).
- [3] Isoda, S., Shimomura, T. and Ono, Y.: "VIPS: A Visual Debugger," IEEE Software, pp.8-19 (May 1987).
- [4] Morishita, S. and Numao, M.: "Prolog Computation Model BPM and its Debugger PROEDIT2," Logic Programming '86 (Proceedings of the 5th Conference Tokyo, Japan, June 1986), ed. Wada, E., Lecture Notes in Computer Science 264, pp.147-158, Springer-Verlag, Berlin (1987).
- [5] Chikayama, T.: "Unique Features of ESP," Proc. of FGCS'84, pp.292-298 (1984).
- [6] Ohara, S. and Tanaka, J. et al.: "A Prototype Software Simulator for FGHC," Logic Programming '86 (Proceedings of the 5th Conference Tokyo, Japan, June 1986), ed. Wada, E., Lecture Notes in Computer Science 264, pp.46-57, Springer-Verlag, Berlin (1987).
- [7] 石崎他: 「プログラム可視化システムにおける動作解析手法」, 情報処理学会第38回(1989).
- [8] 神子他: 「プログラム可視化システムにおける動作表示手法」, 情報処理学会第38回(1989).
- [9] 相川他: 「プログラム可視化システムにおけるユーザ・インタラクション」, 情報処理学会第38回(1989).

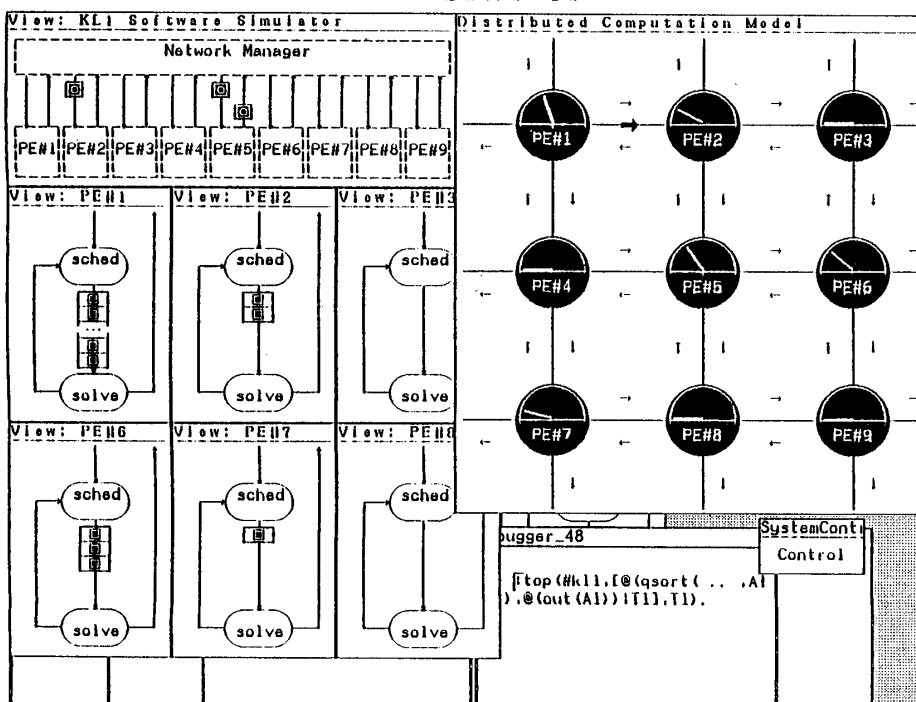


図2. 試作システムでの可視化実行例