

協調問題解決方式による  
ソフトウェア再利用実験システム

7L-1

永松 祐嗣 千吉良 英毅  
(株)日立製作所 システム開発研究所

1. はじめに

ソフトウェアの生産性、信頼性向上を目標に、プログラムを部品化し再利用する種々の方式・技術が提案されている[1]。

ここでは、独立した複数の問題解決器(solver, agent)が、互いに協調して1つの問題を解く協調問題解決方式を用いたソフトウェア再利用実験システムについて述べる。

実験システムの目的は、以下のものである。

- (1) 複数の独立した部品を、問題解決器として協調動作させることにより処理の効率化を図る。
- (2) 部品に関する知識を、個々の問題解決器に蓄えることにより、知識の修正・変更が容易になる等、柔軟性、拡張性のあるシステムとする。

協調問題解決方式では、大局的な一貫性を保って、個々の問題解決器を協調動作させる必要がある。

以下、述語部品、実験システムの処理概要を説明した後、再利用システムにおいて、問題解決器である述語部品が、いかに協調動作するか述べる。

2. 述語部品

対象としているプログラムは、論理型言語であるので、ここでは、部品を述語部品と呼ぶ。

述語部品には、知識ベースと推論エンジンから構成される問題解決器を内蔵している。

問題解決器の知識ベースには、①部品の機能に関する知識、②述語部品の結合部分(インターフェース)に関する知識、③競合の解消ルール、④述語部品のソースコード等を蓄積している。

問題解決器の推論エンジンは、他の問題解決器と通信を行なう。受信した情報と、知識ベースにある知識を使って、他の問題解決器に情報を伝達したり、部品自身の振舞を決める。

述語部品として、図1に示す機能をもったものを用意した。プログラマは、要求仕様を作成する際は、述語部品内部の詳細機能は意識せずに、図1に示す抽象的な機能だけを考えればよい。

3. 処理概要

再利用システムは以下の順序により、処理を進める。

(1) 要求仕様の作成

グラフィックエディタとテキストエディタにより、システムの入力となるプログラマの要求仕様を定義する。グラフィックエディタで、図2に示すようなプログラムの処理機能を、図形を用いて記述する。図2は、プログラムの処理フローとデータフローを併用した記述形式になっている。図2の各ボックスごとに、テキストエディタで、ボックス入出力データとタイプ、ボックスの機能を記述する。

ボックスの機能記述では、図1の述語部品の機能名称を指定したり、入出力データの満たすべき条件を記述する。

- [1] 単体データと単体データが互いに作用する機能。
  - (1) 算術式。
  - (2) 比較式。
  - (3) get/put (データの取り出し/格納)。
- [2] 単体データと集合データが互いに作用する機能。
  - (1) add.  
データを集合データに新規に追加する。
  - (2) delete.  
集合データからデータを削除する。
  - (3) update.  
集合データ中からデータ内容を更新する。
  - (4) search.  
集合データからデータを抽出する。
  - (5) summary.  
集合データからデータを集計する。
- [3] 集合データと集合データが互いに作用する機能。
  - (1) merge.  
集合データと集合データを結合する。
  - (2) distribute.  
集合データを2つの集合データに分離する。
  - (3) sort.  
集合データを分類する。
- [4] I/O.
  - (1) read.
  - (2) write.

図1 述語部品

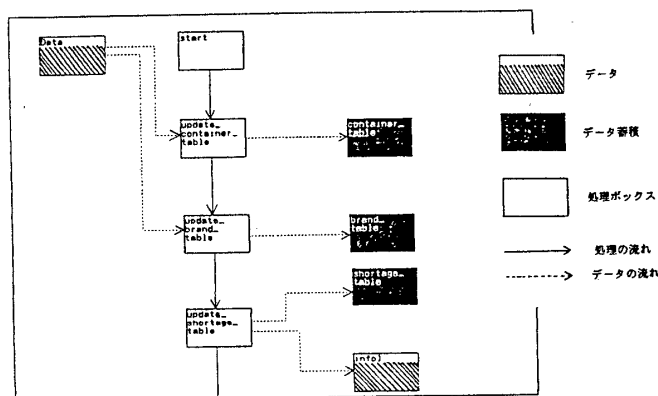


図2 仕様記述例

(2) 要求仕様の解析

システムのトップマネージャが要求仕様を解析し、スケルトンと述語部品検索情報を生成する。スケルトンとは、プログラムの制御構造を与える枠プログラムであり、プログラムのメイン部に相当する。

(3) 述語部品の検索・整合化

述語部品検索情報により部品データベースから、要求仕様に一致する述語部品を、部品間の整合を図りながら、検索する。整合化できない部分は、プログラマが修正する。

(4) 述語部品の合成

整合化した述語部品とスケルトンを結合し、完成プログラムを合成する。

4. 再利用システムにおける協調問題解決

再利用システムでは、述語部品の検索・整合化において、協調問題解決処理を行う。以下、協調問題解決処理について、説明する(図3)。

トップマネージャが生成するスケルトンは、プログラムの制御構造を与える。即ち、述語部品の呼び出し順序を規定している。スケルトンにおいて、述語部品を結合する部分を、コンセントと呼ぶ。

各コンセントが独立に、コンセントに結合すべき、述語部品を次のように検索する。

トップマネージャは、各コンセントごとに、結合すべき述語部品を検索する部品マネージャを生成し、対応する述語部品検索情報を伝える。部品マネージャは、述語部品検索情報を、システムのデータベースに登録してある述語部品群にメッセージ伝達(ブロードキャスト)する。

メッセージを受けた述語部品は、それぞれ独立に、述語部品検索情報に適合するか調べる。自分自身が適合可能と判断したならば、該当する部品マネージャに応答する。

部品マネージャは、応答してきた複数の候補述語部品から、コンセントに結合可能な部品を1つ選択する。

この時、コンセントにそのまま結合できる述語部品は少なく、コンセントに結合できるように、述語部品を整合化しなければならない。従って、選択する部品は、整合化の最も容易な部品にする。

また、結合する述語部品がプログラムのグローバルな情報を含む場合、他のコンセントに結合する述語部品との整合性をとらなければならない。そこで、部品マネージャは他のコンセントの部品マネージャと通信して、それぞれ、どのような候補部品が応答してきたか情報を交換する。そして、部品マネージャは、関係する各コンセント全ての候補述語部品に対し、自分たちで互いに情報交換を行い、結合すべき部品を決めるように指示する。指示を受けた候補述語部品は、プログラム全体の整合性がとれるように、各コンセントで結合すべき部品を、相談して決める。

以上、各コンセントの部品マネージャと、述語部品が、プログラム全体の整合性をとりながら、協調してスケルトンに結合する部品を決めていく。

該当する述語部品がない場合や、検索した述語部品がプログラムの要求に合わない時や整合化できない時は、さらに細かい仕様を与えるか、ユーザがプログラムを修正・追加して、最終的にプログラムを完成させる。

本研究は、I C O Tの委託研究により進めたものである。

[参考文献]

[1] 古宮 誠一, 原田 実 : 部品合成による自動プログラミング, 情報処理, Vol. 28, No. 10, pp. 1329-1345 (1987)  
 [2] Smith, R.G., Davis, R. : Frameworks for Cooperation in Distributed Problem Solving, IEEE Trans. Syst. Man Cybern, vol. SMC-11, No. 1, pp. 61-70 (1981)

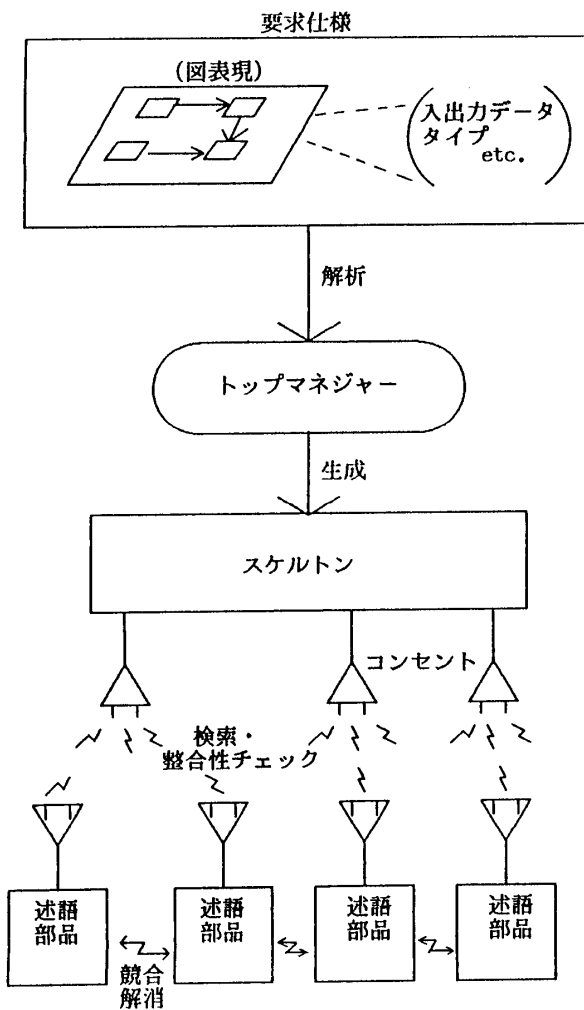


図3 述語部品の協調動作