

Z言語を対象とした対話型診断システム(3)

4V-8

吉田玉緒 松田昇 岡本敏雄

電気通信大学大学院 情報システム学研究所

1 研究の目的

一般に仕様書の解釈の相違が原因となっているバグの発見は非常に難しく、高いコストを要する。この問題に対応するために様々な形式的仕様記述言語が開発されている。Z言語(Z)はその一例である。Zによるコスト削減は評価できるが、習得は容易でない。本研究はZを習得しようとしているユーザを支援するためにZで記述された仕様書(Z仕様書)の診断・記述およびユーザの状態の診断を行なうシステムを構築することを目的とする。具体的にはユーザの作成したZ仕様書の表記の間違いや矛盾を指摘し、ユーザからの質問に解答し、ユーザがZ仕様書の続きを作成するのに困難な場合は解答例を提示する。更に、Z仕様書の診断結果からユーザの状態を推論し、適切なアドバイスを行なう。

2 システムの概要

ソクラテス式問答法による対話は、複雑な主題を個人指導するのに適切である。ユーザの作成したZ仕様書に誤りを検出した場合に、反例や類似解からユーザの誤りに気づかせるのが望ましい。しかし、そのような知的教育システムの持つ強力な対話機構を持たなくとも、ユーザに自力で誤りに気づくようにしむけることにより、これに近い効果が得られるのではないかと考えた。このために工夫したのは、(1)ユーザの作成したZ仕様書をシステムが診断する際に、誤りの具体的な内容をそのまま出力するのではなく、誤りの存在を示唆するにとどめ、ユーザから要求があった場合に誤りの詳しい内容を出力することと、(2)ユーザからZ仕様書の続きの作成を指示された場合に、作成途中のものに不十分な記述があったとしても、それに基づいた仕様記述を行なうことの2点である。

本システムの診断モジュール構築にあたってZスキーマの定型パターンを提案した[4]。本システムは、この

Interactive Diagnosing System on Z Notation(3)
 Tamao Yoshida, Noboru Mastuda, Toshio Okamoto
 The Graduate School of Information System,
 University of Electro-Communications
 1-5-1 Chofugaoka, Chofu-shi, Tokyo, 182 Japan

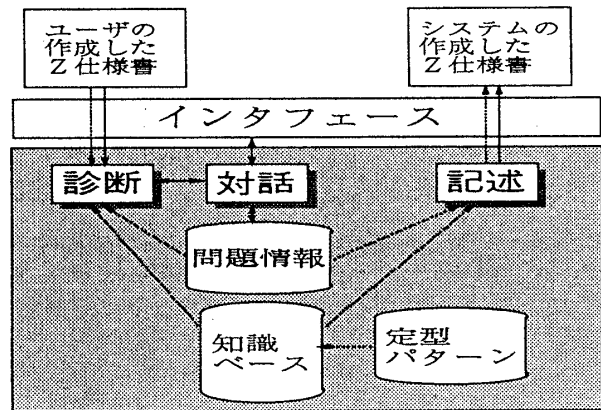


図1: システム構成図

定型パターンとZの表記法や集合・関数などに関する知識ベースを用いることによりZ仕様書の理解や記述を行なっている。本システムの概要を図1に示す。

3 システムによるZ仕様書の診断

本システムでは、ユーザの作成したZ仕様書に対する誤り診断として、(1)単純にZの表記法に関する表記法診断、(2)データの状態やスキーマの相互関係などの論理的矛盾に関する論理性診断、(3)課題の要求事項を満足しているかどうかに関する正解性診断の3種類の診断を行なっている。まず、ユーザの作成したZ仕様書に対して、表記法の診断を行なう。その後、定型パターンを用いてZスキーマを認識する。そして、個々のスキーマから抽出したデータ状態や正常系・異常系などのスキーマの相互関係に対して論理的矛盾・問題情報との不一致を検証する。

4 システムによるユーザ状態の診断

システムは前節で述べた診断の過程を経ることでユーザの誤りを把握する。それらの誤り箇所・内容に関する情報やその履歴を基に、ユーザの状態に関するルール・ベースを駆動してユーザの状態を分類する。システムが想定しているユーザの状態の一部を以下に挙げる。例えば、システムはユーザの作成したZ仕様書のデータ定義の部分が誤っており、且つ単純な表記の誤りでないならば、ユーザは2.のデータの状態に関する理解が不十分である状態である、などと判定している。

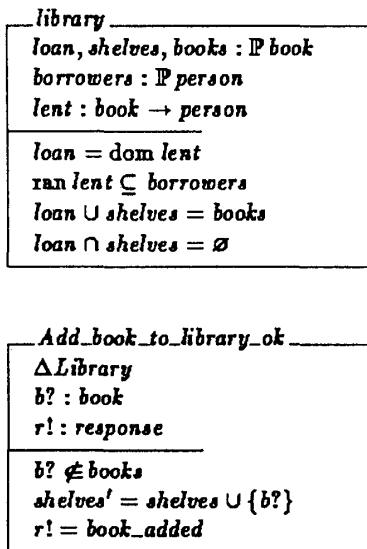


図 2: Z仕様書

1. Z仕様書を途中まで記述して、そこまでをシステムに診断させようと思った
2. 問題が扱っているデータの状態に関する理解が不十分である
3. Zスキーマの構成に関する理解が不十分である
4. 当該スキーマの記述の仕方を知らない
5. 見直しを怠った程度で、つい、うっかりした

これらの状態に対して、システムはシステムに装備されている機能を紹介したり、記述を促したりするメッセージを出力する。

4.の当該スキーマの記述の仕方を知らない、という診断の例を以下に挙げる。この診断を行なうルールは、

- if (1) ある記述事項が欠落していて、
 (2) それ以降に記述すべき事項が存在していて、
 (3)(2)の事項が形式的に正しいならば、
 ユーザは欠落している記述事項の記述を忘れたか、または、その記述の仕方を知らない、と見なす。

である。図2に示すZ仕様書では、当該システムの扱うデータの状態として *loan*、*shelves*、*books*、*borrowers* などの集合および *lent* といった関数を定義しているが、これらの集合の要素となる *book* や *person* についての規定(基本型)の記述が欠落している。そこで、システムは、ユーザは状態空間スキーマ(図2の *library*)や操作スキーマ(図2の *Add_book_to_library_ok*)を記述していることからZスキーマの表記法は知っているが、

基本型の記述がないことから、基本型の記述を忘れたか、または基本型の記述の仕方を知らないものと判断する。そして、基本型の記述の仕方をする方法としてシステムの機能を紹介した後、診断情報の表示に関する指示入力を促すメッセージを出力する。

5 システムによるZ仕様書自動生成

ユーザがZ仕様書の作成に行き詰まることもある。本システムは、ユーザがZ仕様書の続きの作成をシステムに指示した場合に解答例を提示する。しかし、ユーザが作成したZ仕様書に誤りがある場合は、できるだけユーザ自身でその誤りに気づいて貰いたい。そこで、システムが提示するのは、解答例とは言っても作成途中のZ仕様書に基づいたものとしている。

ユーザが状態空間スキーマに集合や関数の定義域・値域相互の関係を記述しておらず、それらの関係について明確に認識していなくとも、記述の不十分な状態空間スキーマに基づいてシステムが作成したZ仕様書を見た場合に「何故、システムはこのような記述をするのだろう」と考えれば、状態空間スキーマの記述に不十分な点があったことに気づく可能性がある。実際にZ言語の初心者には仕様書を作成して貰った実験では、記述が進んでから前のスキーマとの矛盾に気づくことが多かった。このようなことからシステムがZ仕様書の続きを作成することによりユーザの内省を誘発する可能性があると考ええる。

6 今後の課題

本稿では、ユーザのZ習得を支援するための(1)Z仕様書の診断機能、(2)ユーザの状態の診断機能、(3)Z仕様書の自動生成機能について述べた。

システムによるユーザの状態の診断は、現在、静的に行なっている。システムによる診断の妥当性向上のために、システムの診断結果をシステム自身がユーザとの対話を通じて再検討する機能を今後の課題と考えている。今後、その機能の検討および実装を行なう。

参考文献

- [1] J.M.Spivey : The z notation:2nd ed, Prentice Hall, 1992
- [2] A.Collins : Processes in acquiring knowledge, Schooling and the acquisition of knowledge,pp.339-363, 1976
- [3] E.Wenger 著, 岡本敏雄, 溝口理一郎監訳: 知的C A I システム, オーム社, 1990
- [4] 吉田玉緒, 松田昇, 岡本敏雄: Z言語を対象とした対話型診断システム, 情報研報 CE-38, Vol.95, No.111, 1995