

## 会話シミュレーションを基にした語学訓練用 知的 CAI システムの構成†

山本 秀樹\*\* 甲斐 郷子\*\*  
大里 真理子\*\* 椎野 努\*\*

本論文では、英会話という即応力、応用力を要する分野における環境型知的 CAI システムの構成、方式について提案する。本システムの目的は、実際に英語を母国語とする人と会話をしているような環境を学習者に提供することによって、学習者が会話の流れを理解し、話題に追従できるように訓練することである。システムは、まず入力文を構文解析し内部表現を生成する。このとき、文法誤りを含んだ文章を解析するための文法規則を知識ベースに持っているので、誤りを含んだ文からでも内部表現を生成できる。入力文が平叙文ならばその内部表現は、会話の状況を著る状況メモリに取り込まれる。疑問文ならば、その応答は状況メモリと知識ベースを検索して決定される。システムは、会話の目標をスタックに積むことによって会話を制御する。入力文がスタックの先頭の目標と関係があるかどうかによって学習者が会話の流れに追従しているかを判断する。システムは入力文中の単語、文法誤り、会話の流れとのずれを発見するとそれらに対する教育的メッセージを出力するとともに、システムからの発話文のレベルを変化させる。ワークステーション上に構築したシステムの応答時間は、1秒程度であり会話の臨場感を保持するのに十分な速度が得られている。

### 1. はじめに

伝統的な CAI には、(1)学習者が疑問点を質問できないこと、(2)入力の方法が穴埋め式や選択式等の特殊な方式や、数値、単語またはプログラムの断片のような制約の大きい表現に制限されていること、(3)出題するヒントや答えをあらかじめすべて用意しなければならないこと、(4)学習者の誤りの原因を推定できないこと、等の問題がある<sup>1)</sup>。これらの問題を解決するために AI 技術を用いた CAI システムの研究が行われており、既にいくつかの知的 CAI システムが実験的に構築されてきている<sup>2)</sup>。

それらを教授方法に注目して分類すると、対話を中心として教育を進める教授型教育システムと、計算機上に仮想的な実験環境を提供する環境型教育システムとに分けることができる<sup>3)</sup>。前者は、専門家の問題解決の知識を教えることに重点がおかれている。前者のシステムの例としては、南米の地理に関する知識を教育しようとした SCHOLAR<sup>4)</sup> や MYCIN の医療に関する知識ベースの知識を教育しようとした GUIDON<sup>5)</sup> などがある。これらのシステムを通じて学習できる内容は、問題解決のための表面的な知識で

あることが多く、その知識を現実の問題に対してどのように適用するか、といった応用力は十分に学習できないため、学習効果は限定されたものになっている。

一方、後者は対象となる系の挙動を理解させることに重点がおかれている。後者の例としては、蒸気タービンプラントの運転要員を訓練する STEAMER<sup>6)</sup> や、電子回路のシミュレーションを行い、故障診断の専門家の養成を目的としている SOPHIE<sup>7)</sup> などがある。これらのシステムは、いずれも、専門家がある物理現象を理解するときに使用するメンタルモデルを学習者に習得させることを目的としており、現実の問題に対して即応できる応用力をつけさせることができるため、学習効果は高いといえる。

本論文では、英会話という多分に即応力、応用力を要する分野に対する環境型知的 CAI システムの構成、方式について提案する<sup>8)</sup>。システムの目的は、学習者に対して実際に英語を母国語とする人と会話をしているような環境を提供することによって、学習者が会話の流れを理解し、話題に追従できるように訓練することである。システムは、自然言語処理機能と教育機能を併せ持ち、さらに、会話の臨場感を保つためにそれらの処理を高速に行うことが必要である。

会話のシミュレーションを行うための自然言語処理技術としては、一般に入力文の構文解析、意味同定、システム内の知識と入力文から得られる知識の併合、システムからの次の発話内容の決定、そして出力文の生成といった処理技術が必要である。これらの処理技

† Design of an Intelligent CAI System for Training a Foreign Language Based on a Simulation of Conversation by HIDEKI YAMAMOTO, KYOKO KAI, MARIKO OSATO and TSUTOMU SHIINO (Systems Laboratory, R. & D. Group, OKI Electric Industry Co., Ltd.).

\*\* 沖電気工業(株)研究開発本部総合システム研究所

術は、個別的には種々の方式、構成が研究されているが、それらはまだ必ずしも確立されたものとなっていない。システムとして有効な機能を実現するためには、個々の技術をどのように構成し、システムの中にいかに統合していくかが重要な問題である。本論文で提案する会話をシミュレートする機能を持つ知的 CAI システムの構成は、この問題の1つの解になる。

また、教育機能を実現するために、会話中での誤りをどのように定義し、同定し、教育に結び付けるかについて述べる。

本論文の構成は次のとおりである。第2章では、本システムの教育対象領域である英会話教育における本システムの目的を明らかにし、システムに要求される機能について述べる。第3章では、システムの構成と各部の機能について述べる。第4章では、会話をシミュレートするための会話のモデルとその内部表現について論じる。第5章では、会話の制御方式について述べる。第6章では、誤りの同定と学習者のモデルについて述べる。第7章ではシステムと学習者の会話例を示す。

## 2. 英会話教育におけるシステムの目的

本章では、システムの扱う教育対象領域、学習者の知的レベル、そして教育目標を明らかにし、目標を達成するためにシステムに要求される機能について論じる。

我が国における英語教育は、英語の基本事項(表1)の記憶、および日本語から英語または英語から日本語への訳語を中心とした文法・訳読式教授法<sup>9)</sup>と呼ばれるものであり、英会話の教育として不十分であることは既に指摘されているとおりである。これを補うために、英会話学習用の教材として、音声テープ、CD、ビデオテープ等が市販されている。しかし、これらは、多くの例文の記憶、およびパターン化された会話の記憶には有効であるが、現実の会話に際し最も重要な

要素である状況に応じた応答、当意即妙な受け答え等に対する能力の養成には、あまり役に立たないことが多い。このような能力の訓練には、実際に英語を話すことのできる相手と会話をすることが望ましく、英語を母国語とし、教育技術を身に付けた教師による教育が最も効果を発揮することは論を待たない。しかし、そのような教師の数は少なく、また、教育を受ける側が自分の都合のよい時間に適切な教育を受けることは一般的に困難である。そのため、上述のような英会話教師から教育を受けるのと同様の効果をもたらす教育システムの開発が望まれる。

このような教育システムは表2に示す会話に必要な教育要素を学習者に教える必要があり、それは、システムが持つ特定の教授方法によって達成できる。表2の中で、現在の技術レベルで実現困難な5項、6項、および7項を除いた教育要素を教育するために必要な機能について述べる。

### (1) 会話の流れを理解する

システムは、会話の流れを理解し、学習者からの入力に誤りがある場合でも、その入力の意図するところを理解しなければならない。また、会話の流れに依存するような、間接的な発話行為にも対応できなければならない。

### (2) 話題の変化に追従する

会話の目標が決まると、ある程度は会話の流れを予

表2 英会話教育に必要な要素

Table 2 The teaching items for English conversation.

番号	会話に必要な教育要素	教授方法
1	表1に示す基本事項の正確な理解	基本事項の指導、演習.
2	会話の流れの理解	場面を想定しての会話シミュレーション.
3	話題への追従性	場面を想定しての会話シミュレーション.
4	会話特有の表現の理解	会話特有の表現の指導 場面を想定した会話のシミュレーション.
5	ヒアリング	会話文のイントネーションの指導. 会話で起る音の連結(リエゾン)などの指導.
6	発音	会話文のイントネーションの指導. 会話で起る音の連結(リエゾン)などの指導.
7	英語圏文化の理解	英語圏の歴史、文化の指導.

表1 英語学習の基本事項

Table 1 Basic skills for English training.

項目	学習内容
単語	綴り、発音、単語の持つ概念、使用方法、変化形、派生語など.
熟語	熟語の持つ概念、同意の熟語や単語.
構文	同意の構文、意味、使用方法.
文法	文型、品詞の特徴、文の構造、時制など.

測できる。しかし、学習者の興味の変化または学習の進度により会話の流れはダイナミックに変化すると考えられる。システムは学習者が主導する会話の流れに追随し、状況に応じて適切な会話が行えなければならない。

(3) 学習者のレベルに合わせた会話を行う

システムは、学習者の会話能力に合わせて会話のレベルを調節でき、また能力が増すにつれて徐々に会話のレベルを上げていくことができなければならない。

(4) 教育的なアドバイスを行う

学習者の入力に、単語や文法の誤りなどが含まれていた場合には、それに対して教育的なアドバイスができなければならない。

(5) 即時に回答する

会話の臨場感を保つために、システムの応答は即時でなければならない。

本システムは以上の機能を実現すべく構成されている。

### 3. システムの構成

本システムの構成を図1に示す。システムは、入力文理解部、質問応答部、会話制御部、出力文生成部の4つのモジュール、文法辞書知識ベース、対象知識ベース、会話制御知識ベース、出力文生成知識ベースの4つの知識ベース、モジュール間のインタフェースとなる予測メモリ、理解状態記述メモリ、状況メモリ

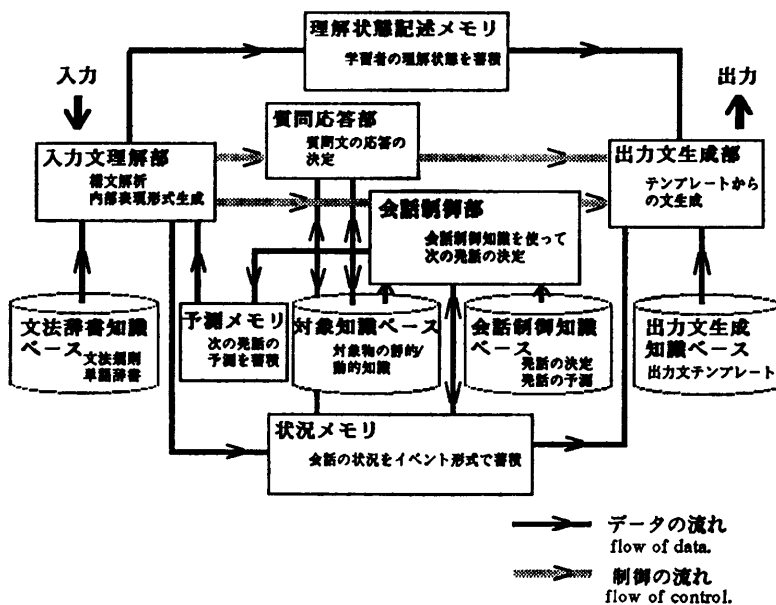


図1 システムの構成  
Fig. 1 The configuration of the system.

の3つの短期記憶からなる。以下に各部の概要を説明する。

(1) 入力文理解部

入力文理解部は、入力文の構成解析を行い解析木を生成し、それを基に内部表現を生成する。本システムでは、内部表現形式をイベント形式（詳細は4.2節で述べる）と呼び、変換された入力文をイベントと呼ぶ。入力文理解部は、構文解析と同時に、単語・文法などの誤りも調べ、その情報を理解状態記述メモリに記録する。入力文が平叙文の場合には、イベントを状況メモリ中に埋め込み、会話制御部を起動する。入力文が質問文や命令文の場合、イベントを質問応答部に送る。

(2) 質問応答部

質問応答部は、質問文や命令文のイベントに対する答える内容を決定する。そのため、必要な知識を対象知識ベースと状況メモリの中から探し出す。答える内容は、出力文生成部に送られる。

(3) 会話制御部

会話制御部は、会話の流れに沿った発話を実現するために、会話制御知識を使って次の発話内容を決定する。発話内容は、出力文生成部に送られる。

(4) 出力文生成部

出力文生成部は、応答を高速に行うためテンプレート方式<sup>10)</sup>で出力文を生成する。出力文生成部は、質問応答部、または会話制御部から発話内容を指定されて起動される。学習者のレベルに合わせて出力文の複雑さを変化できるように、1つの発話内容に対して、テンプレートは複数用意してある。出力文生成部は、発話内容と理解状態記述メモリの内容によって使用するテンプレートを決定する。テンプレートを埋めるデータは、直接指定される場合と、状況メモリから探し出される場合がある。

(5) 知識ベース

本システムには、会話制御知識、対象知識、文法辞書知識、そして出力文生成知識の4種類の知識ベースがある。

会話制御知識は、会話を制御するための知識でありイベント形式とルール形式から構成される（詳細は

4.2 節で述べる)。会話制御知識は、次の発話内容の決定や、学習者の次の入力に対する予測に使用される。

対象知識は、会話の際に必要な対象物についての静的な知識や、手続き的な知識である。例えば、ホテルのチェックインの会話の場合では、ホテルの空き部屋の情報などが静的な知識であり、2部屋以上の予約をした場合に、1部屋の料金から複数の部屋の料金を計算する知識などが手続き型の知識である。この知識は、質問応答部と会話制御部で使用される。

文法辞書知識は、学習者の入力文をイベント形式に変換するための文法知識と単語の辞書からなる。文法知識は、正しい文を処理する文法だけでなく、誤りを含んだ文を処理する文法、誤りに対する教育的メッセージ情報、内部形式を生成するための知識を含んでいる。この文法辞書知識ベースは入力文理解部によって使用される。

出力文生成知識は、出力文生成部が使用する英文のテンプレートからなる。

#### (6) 状況メモリ

状況メモリは、会話の状況を反映する記憶であり、会話の進行に伴って刻々と変化する。状況メモリは、次のようにモジュール間のインタフェースになっている。入力文理解部は、学習者の入力文をイベントに変換して状況メモリに埋め込む。質問応答部は、質問の答えを探し出すために状況メモリをアクセスする。会話制御部は、次の発話を決めるために状況メモリをアクセスする。出力文生成部は、出力文の生成のために状況メモリをアクセスする。

#### (7) 予測メモリ

予測メモリは、次の入力で予測される入力文のイベントと、その予測がはずれたときの動作を決める予測ルールを記憶する。予測メモリの内容は、学習者の発話ごとに会話制御部によって変更される。

#### (8) 理解状態記述メモリ

理解状態記述メモリは、学習者の理解状態を反映している記憶である。学習者の理解状態は、システムとの会話の中の誤りによって決定される。詳細は、6章で述べる。

## 4. 会話モデルの内部表現

### 4.1 会話のモデル

計算機との会話を実現したシステムとして、ELIZA<sup>11)</sup>、BASEBALL<sup>12)</sup> などがあるが、これらのシステ

ムでは、学習者が入力した文に対して、構文情報や意味情報を詳細に解析せず、入力文のキーワードを探すためのパタンを用意し、それとマッチする情報を使用して出力文を生成し会話を実現している。このようなシステムとの会話は、自然言語が持つ構文情報や意味情報を使用していないため単調な会話になり、会話の流れへの追従性などの訓練には十分でない。

十分な会話の訓練を行うために、本システムでは、文法辞書知識を使って構文解析、意味解析を十分行い意味情報を扱った会話を行う。また、会話のモデルとして目標指向の会話モデルを用いる。目標指向の会話モデルとは、会話には目標があり、それを達成するために会話が情報が交換するというモデルである。このモデルは、以下の解析を容易にする。

- (1) 話題の変化を予測する。
- (2) 間接的な発話行為を理解する。
- (3) 相手の発話を予測する。

このモデルに基づいて会話をシミュレートすることで、一方が主導権を持った会話だけでなく、双方が主導権を持つ会話が可能となり、会話の訓練を十分に行うことができる。

## 4.2 会話モデルの内部表現

### 4.2.1 発話文の内部表現

本システムでは、文の内部表現としてイベント形式と呼ぶ知識表現形式を採用した。イベントの持つスロット名とその意味を表3に示す。

表3 イベント中のスロット名の意味  
Table 3 The meanings of slot in an event.

スロット名	スロット値の表す意味
Predicate	動作
Agent	動作の主体
Object	動作の対象
Truth-value	否定か肯定か
Want	可能、要求、それ以外か
From	動作の始点
To	動作の終点
Tense	過去、現在、未来のいずれか
Method	動作を実行するための方法
Place	動作を実行する場所
Sentence-type	文の種類
Others	上記のスロットに入らないその他の情報

```
(room(roomtype single)
  (number 1)
  (sound quiet)
  (view good))
```

図2 “a quiet single room with good view.”を表すオブジェクトリスト

Fig. 2 An object list of “a quiet single room with good view.”

object スロットの値はオブジェクトリスト, others スロットの値はプロパティリストと呼ばれるリスト形式をとり, 修飾関係を持つ対象を表すことができる。プロパティリストは, 属性と値の対リストを要素とするリストであり, オブジェクトリストは, 1つのカテゴリ値とプロパティリストの要素からなるリストである。例えば, “a quiet single room with good view” (静かで眺めのよいシングル部屋) は, 図2に示すようにオブジェクトリストで表現できる。ここで, リストの第1要素の room はカテゴリ値であり, room-type や number は属性, そして single や 1 がそれぞれの属性値である。

sentence-type スロットは, 文の種類を表すスロットである。文の種類は, 平叙文, YES/NO で答えられる疑問文, それ以外の疑問文の3つに大別される。

イベント形式は, 各モジュールのインタフェースになっている状況メモリの構成要素の表現形式である。入力文理解部は, 入力文からイベントを生成し, それを状況メモリに埋め込む。会話制御部, 質問応答部は, 状況メモリのイベントを参照して次の発話を決定する。イベントは, いつもすべてのスロットが埋まっているとは限らない。

#### 4.2.2 会話制御知識の内部表現

本システムでは, 会話の中で発話された1つの文は, 目標を達成するために必要な情報を補うものとみなす。スロットは, 会話によって得られる知識を入れるための器であると考えられる。このスロットを埋めていくのが会話であり, 必要なスロットがすべて埋まったとき, その会話の目標が達成されたことになる。

会話制御知識は, 予測イベントと呼ばれる値の埋っていないスロットを持つイベントと, それを埋めるためのルールから構成される。ルールの条件部は, 会話の状況に応じてルールの動作部を起動するか否かを決定するために使用される。会話の状況は, 状況メモリのイベントを参照することで判断できる。

ルールの動作部には, 次の4つの動作のいくつかを記述できる。(1)実際に発話を行うために出力文生成部を起動する, (2)他の会話制御知識を活性化する,

```
C...Clerk, T...Traveller
C: May I help you?
T: I want a room.
C: Do you have a reservation?
T: Yes.
C: May I have your name?
T: My name is John Tenniel.
```

図3 ホテルのチェックイン時の会話

Fig. 3 A subdialogue of a hotel check-in situation

```
知識 I D: 1
目 標: ホテルにチェックインする
予測イベント: 予約イベント, 支払イベント, ...
ル - ル 0: if 予約イベントのスロットが埋っていない
           then 知識 ID2 を活性化する
ル - ル 1: if 支払イベントのスロットが埋っていない
           then 知識 ID3 を活性化する

知識 I D: 2
目 標: 予約を確認する
予測イベント: 予約イベント, 名前イベント, ...
ル - ル 0: if 予約イベントのスロットが埋っていない
           then “Do you have a reservation?” を発話する
ル - ル 1: if 名前イベントのスロットが埋っていない
           then “May I have your name?” を発話する

知識 I D: 3
目 標: 代金を支払う
予測イベント: 支払方法イベント, カードイベント, ...
ル - ル 0: if 支払方法イベントのスロットが埋っていない
           then “Do you pay by card or ...?” を発話する
ル - ル 1: if カードイベントのスロットが埋っていない
           then “Please show me your card.” を発話する
```

図4 ホテルのチェックイン時の会話の会話制御知識

Fig. 4 Conversation control knowledges for a hotel check-in situation.

(3)対象知識を更新する, (4)次の発話を予測する。

ここで述べた目標指向の会話とその会話制御知識を図3, 4に示す。図3は, ホテルのチェックインの場面の会話例である。チェックインという目標を達成する会話は, 予約を確認する, 名前を知る, といった副目標を持つ。図4は, この会話の会話制御知識を示している。

#### 5. 会話の制御

本章では, 目標指向の会話モデルに基づいた発話内容の同定と, 会話の制御方式について論じる。

## 5.1 発話内容の同定

### 5.1.1 予測による発話の同定

目標指向の会話を行っている場合、人間は次の発話を予測することで、省略文を解釈したり、話題の転換を認識したりしていると考えられる。したがって、省略文を補完したり、話題の転換を認識したりする機能を実現しようとする、システムには、次の発話を予測する機能が要求される。

本システムは、会話制御知識の予測イベントを次の入力として予測されるイベントとし、予測メモリに置くことで予測を行う。予測メモリに置かれる予測イベントは、1つとは限らない。一方、自然言語の構文の曖昧性などの理由で、入力文理解部は、省略がない場合でも1つの入力文から複数のイベントを生成することがある。会話の流れを理解するためには、入力文から生成された複数のイベントと複数の予測イベントから、学習者が真に意図するイベントを生成しなければならない。本システムでは、そのために put 型のマッチング方式を導入した。

put 型でマッチするとは、入力イベントのすべてのスロットの値が、予測イベントの対応するスロットの値に反していない場合と定義する。put 型でマッチした結果得られるイベントを学習者の意図したイベントとする。

例えば、システムは、“What kind of room do you want?” という発話を行った後、図 5(1)に示す予測イベントを予測メモリに置く。この質問に対し、学習者が“I want a single room.”の省略文として、

```
predicate: possess
agent      : student
object     : (room (roomtype))
```

(1) 予測イベント

(1) An expectation event.

```
predicate: nil
agent      : nil
object     : (nil (nil single))
```

(2) 省略文“single.”の入力イベント

(2) An event of abbreviated input sentence “single.”

```
predicate: possess
agent      : student
object     : (room (roomtype single))
```

(3) put 型のマッチングの結果得られるイベント

(3) The result of put-type match between Fig. (1) and Fig. (2).

図 5 put 型のマッチングの例

Fig. 5 An example for put-type match.

“Single” と入力したとする。図 5(2)は、省略文の入力イベントであり図 5(3)は、図 5(1)と図 5(2)を put 型でマッチングした結果得られるイベントで、これが学習者の意図したイベントになる。

### 5.1.2 以前の発話内容との照応

会話の中で何度も現れる1つの対象を、システム内部でも1つの対象として扱う必要がある。そのため学習者の意図したイベントと、以前の発話内容である状況メモリのイベント間の照応関係を考慮しなければならない。本システムでは、照応関係のあるイベントとは、put 型でマッチするイベントと定義する。マッチした場合は、状況メモリ中のマッチしたイベントのスロットに入力イベントの情報を付加し、マッチしない場合は、状況メモリに入力イベントを新たに付加する。

状況メモリへの付加は、そのイベントの sentence-type が平叙文であるものについてだけ行う。

### 5.1.3 同じ意味を持つが表現が異なる発話の処理

会話では、異なる表現の文が同じ意味を持つことがある。例えば、“I want to stay here.”と“I want a room.”のような異なる表現が、ホテルのフロントのチェックインの最初の会話では、「部屋がほしい」という同じ意味を表現していると考えられる。

本システムでは、このような異なる表現から同じ1つの意味を推論するためにデーモンを用いている。デーモンは、入力イベントを状況メモリに埋め込む際に起動される。このデーモンは、入力イベントと同じ意味を持つイベントを状況メモリに埋め込む。これにより上記の例では、どちらの入力に対しても「部屋がほしい」という意味のイベントが埋め込まれる。

### 5.1.4 発話と関連する内容の推論

会話では、1つの発話からいくつかの関連する事柄を推論できる場合がある。例えば、ホテルのチェックイン時の最初の会話で、“I have a reservation.”という旅行者の発話から、「旅行者は、宿泊しようとしている。」ということを推論できる。

本システムでは、発話と関連する内容の推論を行うために5.1.3項と同じようにデーモンを用いている。このデーモンも、入力イベントを状況メモリに埋め込む際に起動され、入力イベントから推論可能なイベントを状況メモリに埋め込む。

## 5.2 発話内容の決定

### 5.2.1 スタックによる会話の焦点制御

本システムでは、会話の焦点を管理するために焦点

スタックを導入する。会話の中で、活性化された会話制御知識は、焦点スタックに積まれる。一番最近活性化された会話制御知識は、スタックの先頭にある。

平叙文の入力イベントは、Grice の会話の協調原則<sup>13)</sup>により一番最近活性化された会話制御知識のルール条件部の条件を満たすものと予測できる。会話制御部は、ルールが発火すると、そのルールの動作部を実行する(4.2.2項参照)。動作部を実行した結果、別の会話制御知識が活性化されると、会話の焦点が移動する。活性化している会話制御知識中に発火可能なルールが1つもないときは、焦点スタックから一番上の会話制御知識が取り除かれ会話の焦点は前に戻る。

また、5.1.3項、5.1.4項で述べたデーモンによって、相当する会話制御知識をスタックに積むことができる。これによって、会話の流れからずれた発話への対応が可能になる。会話の流れからのずれの同定については、6.3節で述べる。

### 5.2.2 質問に対する発話内容の決定

疑問文の入力イベントに対する答えは、質問応答部が状況メモリまたは対象知識ベース中のイベントから関連のあるイベントを見つけ出して生成する。関連するイベントを見つけ出すために、get型のマッチングを導入する。

イベントAがイベントBとget型でマッチするとは、イベントA、Bのすべてのスロットについて(1)Bのスロットに値があるときは、Aのスロットの値は、Bのスロットの値に等しいか、または変数である、(2)Bのスロットに値がないときは、Aのスロットにも値がない、場合であると定義する。

質問応答部は、入力イベントのsentence-typeがYES/NO型質問ならば、get型でマッチしたイベントの真偽スロットを見て答えを生成する。それ以外の質問ならば、入力イベントのスロットの値に変数を割り付けget型のマッチングを行う。そして、get型でマッチしたイベント中の対応するスロットの値から答えを生成する。

### 5.3 発 話

会話制御部や質問応答部で出力する情報が決まると、出力文生成部が実際の英文を生成する。このとき、学習者のレベルに応じた英文を生成することが重要になる。学習者のレベルは、誤りなどを反映している理解状態記述メモリを参照して決める。

学習者のレベルが低い場合は、答えやすい質問、すなわち、YES/NO型や選択型の疑問文を発話する。

学習者のレベルが高い場合で、システムの応答文が簡単すぎると判断された場合は、再び、会話制御部を起動してさらに発話すべき情報を探す。学習者レベルに対し十分な複雑さに達した時点で応答文を発話する。

## 6. 誤りの同定

### 6.1 単語の綴り誤り

本システムでは、キー入力の間違い、記憶違いなどで文法辞書知識にない文字列が入力された場合でも会話を中断しないために、正しい単語を見つけ出すための機構<sup>14)</sup>を入力文理解部に持たせた。

### 6.2 文法、構文の誤り

文法、構文誤りを含む文を解析する場合、通常の構文解析が失敗した後に、文法誤りを探索するのでは効率が悪い。そこで、本システムは、学習者が誤って使用する可能性のある文法規則(誤り文法規則と呼ぶ)と、その誤りに対する教育的なアドバイスを、正しい文法規則とともに、文法辞書知識として持ち、誤りを含んだ文も1回で構文解析を行う。文法誤りがある文は、構文解析の結果として生成される構文解析木に文法誤りがあったという情報が付加され、システムは、その文法誤りを学習者に指摘し、教育的なアドバイスを行う。入力文理解部は、構文解析木に文法誤りが含まれているか否かにかかわらずイベントを生成する。すなわち、文法誤りを含んだ文が入力されても会話を続けることができる。

```
NP → "a" N1
NP → N2
N1 → "room"
N2 → "rooms"
NP* → "a" N2
(*: 誤り文法)
```

(1) 文法辞書知識の例

(1) Grammar and dictionary knowledges.

```
NP*
 / \
 "a" N2
     |
     "rooms"
```

(2) "a rooms" の構文解析木  
(2) A parsed tree of "a rooms."

複数形には a を付けてはならない

(3) 教育的メッセージ

(3) An instructive message.

図 6 文法誤りの同定

Fig. 6 Identification of grammatical error of utterance of a student.

図6に文法誤りの同定の例を示す。図6(1)は、文法規則知識の例である。左辺が NP\* の文法規則が誤り文法規則である。図6(2)は、誤り文法規則を使った構文解析木である。システムは、この構文解析木を見て誤りを発見し図6(3)の教育的メッセージを学習者に提示する。

### 6.3 会話の流れに関する誤り

5章で述べた会話制御の機構により、どのように会話の流れに関する誤りを指摘できるかについて論じる。

焦点スタックの一番上の会話制御知識のルールの場合に合わない入力イベントがあると、会話の流れからずれたことになる。このずれには、妥当なものやそうでないものがある。システムの質問に対して学習者が答えた直後に少し異なる話題について話をし、もとに戻るのとは妥当である。しかし、疑問文に対して答えなかったり、以前の話題に戻ったり、現在の話題と大きく異なる話題に変えたりするのは、妥当でない。

これらを認識するために、システムはデーモンを使用する。入力文のイベントが状況メモリに埋め込まれる際にデーモンは、起動され、焦点スタックの会話制御知識を上から順次調べていきマッチするルールを探す。スタックの中にマッチするルールがあれば、学習者が現在の話題に追隨できていないことになり、システムは理解状態記述メモリに会話の追隨性の誤りを記録する。さらに、話題に追隨できていないことにより、当てはまるルールがあった会話制御知識より上の会話制御知識は、すべて破壊されたものとしてもとに戻る。これにより、学習者の会話に追隨する。

スタックの中にマッチするルールがない場合は、マッチする新しい会話制御知識を会話制御知識ベースから探す。もし、マッチする会話制御知識が存在したときは、妥当な話題の転換とみなし、その会話制御知識を焦点スタックの一番上に積んで、学習者の話題に追隨する。この話題に関する会話が終了すると、会話制御知識は焦点スタックから降ろされ話題は以前のものに戻る。

会話制御知識ベースの会話制御知識のどれともマッチしないときは、大きく異なる話題に転換しようとしているか、現在の話題を理解していないとみなし、理解状態記述メモリに会話の追隨性の誤りを記録する。そして、会話の流れを戻すための発話を行う。

### 6.4 学習者モデル

本システムでは、上記で述べたように学習者の発話

の誤り等から理解状態を把握している。文法誤りについては、システムの内部に、あらかじめ学習者の犯しそうな誤りを持たせておき、それにより学習者の入力の誤りを見つけ理解状態を把握しようとするバグーモデル<sup>10)</sup>である。

会話の流れの誤りについては、システムの内部に会話のモデルを持ち、それに基づいて誤りを同定するという方式を用いている。

## 7. システムと学習者の会話例

ホテルのフロントでの会話を例にして本システムと学習者の会話を図7に示す。

①でまず、システムは、学習者が「宿泊したいという意志を持つ」ということを理解し、チェックインという目標を達成しようとしていることを理解する。その結果、副目標として「予約の確認」を設定し②を発話する。

システムは、予約の確認の次の話題として③で名前を尋ねている。予約者の名前と部屋のタイプは、あら

C...Clerk (System), T...Traveller (Student)

- C: May I help you?  
 T: I want a room. ...①  
 C: Do you have a reservation? ...②  
 T: Yes.  
 C: May I have your name? ...③  
 T: My name is John Tenniel.  
 C: Just a moment, please.  
 I'm sorry, I can't find your name. ...④  
 T: Check my name, again. ...⑤  
 C: May I have your name, again?  
 T: My name is John Tenniel.  
 C: Yes, here it is. Mr. Tenniel, we have a reservation for a twin for tonight for you. Is that right? ...⑥  
 T: Yes.  
 C: Twin is \$110. OK? ...⑦  
 T: It's too expensive. ...⑧
- 綴りが正しくありません  
 誤: expensiv  
 正: expensive
- C: There is \$100. OK?  
 T: OK. By the way, do you know the good place to have dinner around here? ...⑩  
 C: Yes, I think our restaurant is good. Would you fill in this card?  
 T: No. I have a credit card. ...⑪  
 C: Please put your name and address here. ...⑫  
 T: All right.

図7 システムと学習者の会話例

Fig. 7 A subdialogue of a hotel check-in situation between this system and a student.



はじめ対象知識ベースに静的知識として格納されており、システムは、入力された名前があるかどうかを、手続き的知識を使って調べる。④では、学習者の当意即妙性を確かめるために、いったん「名前が見つからない」と答えている。このような発話も、会話制御知識に記述することができる。これに対して、学習者は⑤で、もう一度調べるように要求している。⑥では、予約が見つかったのでその内容の確認を取っている。

次の⑦では、チェックインのための値段の交渉の副目標を設定している。⑧の学習者の答えに含まれる代名詞 *it* が、状況メモリのイベントを探索して \$110 を指していることを理解する。また、⑨で示すように単語の綴り誤りを指摘し正しく解釈する。

⑩では、学習者が話題を転換しているが、システムは、それに追隨して応答を返している。そして、この話題が終わるともとの話題に戻り、さらに話を進めている。⑪の学習者の答えは、会話の流れからはずれているので、応答文のレベルを変えて⑫の言い方を取っている。

## 8. おわりに

本論文では、英会話のシミュレーションを行う知的 CAI システムの機能を分析し、システムの構成とその実現方法について述べた。本システムは、英会話教育のための会話モデルとして目標指向の会話モデルを採用し、それに基づいて会話をシミュレートする。これにより、双方が主導権を持つ会話が可能になり、会話の訓練を十分行うことができる。システムは、学習者の発話の中の単語、文法の誤りなどを同定し、その誤りを指摘する。さらに、誤りを含んだ発話からでも正しい意味を理解し、会話を続けることができる。システムからの発話は、学習者のレベルに沿って変化する。

システムは、ワークステーション ELIS 上に LISP で構築されている。応答時間は、1秒程度であり会話の臨場感を保持するのに十分な処理速度である。また、音声合成装置を接続することによりある程度のヒアリングの訓練にも役立つように考慮してある。

本システムの会話制御知識ベースと対象知識ベースに、会話以外の教育分野に関する教授型の知識を入れることにより、教授型と環境型の両機能を合わせ持った知的 CAI システムを実現することも可能である。本システムで用いた誤り文の意味を理解する方式や、目標指向の対話の制御方式は、他の分野の知的 CAI

システムや、データベース、エキスパートシステム等の自然言語インタフェースとしても有効である。

謝辞 本システムの開発に協力していただいた沖テクノシステムズラボラトリ(株)加藤正明氏、沢山ゆかり氏、浅野雅代氏に深謝する。

## 参考文献

- 1) 大槻, 山本: 知的 CAI のパラダイムと実現環境, 情報処理, Vol. 29, No. 11, pp. 1255-1265 (1988).
- 2) Sleeman, D.H. and Brown, J.S. (eds.): *Intelligent Tutoring Systems*, Academic Press, London (1982).
- 3) 豊田, 中村: 知的 CAI における知識表現と教授法, 情報処理, Vol. 29, No. 11, pp. 1266-1274 (1988).
- 4) Carbonell, J.R.: AI in CAI, *IEEE Trans. Man-Mach. Syst.*, Vol. MMS-11, No. 4, pp. 190-202 (1970).
- 5) Clancy, W.J.: *Knowledge-based Tutoring: The GUIDON Program*, MIT Press, Cambridge, Massachusetts (1987).
- 6) Hollan, J.D., Hutchins, E.L. and Weitzman, L.M.: STEAMER: An Interactive, Inspectable, Simulation-Based Training System, in Kearsley, G. (ed), *Artificial Intelligence and Instruction: Application and Method*, pp. 111-134, Addison-Wesley, Reading, Massachusetts (1987).
- 7) Brown, J.S., Burton, R.R. and Bell, A.G.: SOPHIE, A Step toward a Reactive Learning Environment, *Int. J. Man-Mach. Stud.*, Vol. 7, No. 5, pp. 675-696 (1975).
- 8) 山本ほか: 英会話教育用 ICAI システムの構築について, 情報処理学会コンピュータと教育研究会報告, Vol. 88, No. 48 (1988).
- 9) 清水: 英語科教育法, 開拓社, 東京 (1980).
- 10) Bobrow, D.G. et al.: Gus, A Frame-Driven Dialog System, *Artif. Intell.*, Vol. 8, pp. 155-173 (1977).
- 11) Weizenbaum, J.: ELIZA—A Computer Program for the Study of Natural Language Communication between Man and Machine, *Comm. ACM.*, Vol. 9, No. 1, pp. 36-45 (1966).
- 12) Green, B.F. et al.: BASEBALL: An Automatic Question Answerer, in Feigenbaum, E.A. and Feldman, J. (eds.), *Computers and Thought*, McGraw-Hill, New York (1963).
- 13) Grice, H.P.: Logic and Conversation, in Cole, P. and Morgan, J. (eds.), *Syntax and Semantics*, Academic Press, London (1975).
- 14) 大里ほか: 英会話教育用知的 CAI システムにおける構文解析, 第 37 回情報処理学会全国大会

論文集, pp.1993-1994 (1988).

- 15) Burton, R.R.: Diagnosing Bugs in a Simple Procedural Skill, in Sleeman, D.H. and Brown, J.S. (eds.), *Intelligent Tutoring Systems*, Academic Press, London (1982).

(平成元年2月10日受付)

(平成元年4月11日採録)



山本 秀樹 (正会員)

1961年生。1984年京都大学工学部電気工学科卒業。同年沖電気工業(株)入社。以来、総合システム研究所において、データベースマシン、知識ベースシステム、知的 CAI システムの研究に従事。一般システム理論、会話モデル、知識表現に興味を持つ。ACM, 人工知能学会各会員。



甲斐 郷子 (正会員)

1959年生。1984年九州大学工学部情報工学科卒業。1986年同大学院修士課程修了。同年沖電気工業(株)入社。以来、総合システム研究所において、ソフトウェア工学、AI 応用システムの研究に従事。特に、知的 CAI, 対話制御に興味を持つ。人工知能学会会員。



大里真理子 (正会員)

1961年生。1983年九州大学文学部社会学科卒業。同年沖電気工業(株)入社。以来、エキスパートシステム、自然言語処理システム、CAI システムの開発に従事。現在、同社総合システム研究所にて、知的 CAI の研究に従事。CAI 学会会員。



椎野 努 (正会員)

昭和16年生。昭和39年名古屋大学工学部電気工学科卒業。同年沖電気工業(株)入社。マイクロ波通信システムの研究、無線伝送システム自動設計の研究、ソフトウェア設計支援システムの研究等を経て、現在研究開発本部総合システム研究所において機械翻訳、自然言語処理、知的 CAI, エクスパートシステム等、人工知能関連の研究開発に従事。工学博士。電子情報通信学会、人工知能学会、日本ソフトウェア科学会、日本心理学会、IEEE 各会員。