ヒューマンインタフェース 80－2
音声言語情報処理 23－2
（1998. 10. 16）

マルチモーダル図形エディタのための漸進的な話し言葉処理手法

松原 茂樹† 河口 信夫‡ 外山 勝彦‡ 稲垣 康善‡

†名古屋大学言語文化部 ‡名古屋大学大学院工学研究科計算理工学専攻

mmg@inagaki.nuie.nagoya-u.ac.jp

概要

即時応答機能を備えたマルチモーダル音声対話システムのための話し言葉処理手法について述べる．ユーザの発話にいつでも即座に応答するために，システムは話し言葉の入力に対して同時進行的に解析を進める必要がある．また，発話途中の段階でも，それまでの入力からユーザの意図を推測することにより，即時性を一層高めることができる．本稿では，通常の話し言葉処理とタスクに依存した処理が入力に対して同時進行的にかつ協調的に振舞う枠組を提案する．作成された解析結果のうち，最も多くの情報を保持するものをシステムの理解結果とし，それをもとに応答を生成する．作図をタスクとするマルチモーダル図形エディタ Sync/Draw の解析モジュールとして本手法を説明する．

Incremental Spoken Language Processing for Multimodal Drawing Editor

Shigeki MATSUBARA†, Nobuo KAWAGUCHI‡, Katsuhiko TOYAMA‡ and Yasuyoshi INAGAKI‡

†Faculty of Language and Culture, Nagoya University
‡Department of Computational Science and Engineering, Nagoya University

mmg@inagaki.nuie.nagoya-u.ac.jp

ABSTRACT

Multimodal dialogue systems with quick responses are required to understand spoken language incrementally in order to respond to an utterance immediately. This paper proposes a method for incremental processing of spoken language. The method is composed of conventional incremental parsing and task-dependent parsing, and they behave in a synchronous and a cooperative way. At any time choosing the most appropriate one from results which the modules generates, the system can respond to the user in an early stage of the input. We explains the method with examples from a multimodal drawing editor.

1 はじめに

人間と計算機との間の自然なインタラクションの実現のため，グラフィカルユーザインタフェース (GUI) に音声対話インタフェースを融合したマルチモーダルインタフェースの研究が盛んに行われている [1, 8, 14, 18]．音声は人間にとって最も基本的なコミュニケーション手段であり，新たに音声入力が可能となれば，システムの使い勝手も向上し，より効率的なインタラクションの実現が期待できる．このようなマルチモーダルシステムを設計する上での重要なポイントは，それぞれのインタフェースが備える特長を損なうことなく，それらを有機的に融合することである．

GUI ベースの対話システムが備える特長として，視覚的フィードバックの即時性を挙げることができる．例えば，ポインティングデバイスを用いた入力に対して，システムがいつでもすぐさまリアクションを返すことにより，ユーザは自らの意図の伝達に成功したか否かを瞬時に確認することができ，加えて，次に行うべき操作に関する具体的な示唆が得られることもある．GUI におけるこの性質は，システムがもつ高度な対話性や操作性の一端を担っており，より密なインタラクションを実現する上でのポイントとなっている．GUI の特長を備えたマルチモーダルシステムを構築するために，即時的なフィードバック機能を備えた音声対話インタフェースが不可欠であり，それに適

した音声認識ならびに話し言葉処理が求められる．

我々は，音声を用いた効果的なマルチモーダルシステムには，発話同時理解が不可欠であると考える．発話同時理解とは，自然言語文の音声入力に対してそれとほぼ同時進行的に意味を理解し，入力途中の段階でもその時点までの解析結果に基づき随時，適切なフィードバックを返すものである [9]．そのような機能をもつシステムは，GUI の利点であるフィードバックの即時性，ならびに音声対話インタフェースの利点である入力の自然性や自由度の高さを兼ね備えることができるため，ユーザフレンドリな操作環境の提供が期待される．

本稿では，発話同時理解のための話し言葉処理手法について述べる．本手法は，自然言語の漸進的解析 [6]，すなわち，自然言語文に対して，左から右へできる限り語単位での構文・意味・語用解析をベースとしている．語が入力されるたびに文法と辞書を用いて解析を進め，その時点までの入力に対する解析結果を随時作成することにより，それを反映したリアクションを即座に返すことができる．

一般に，自然言語の漸進的解析では，語が入力されるたびに，それまでに入力された表現の意味を順次構成することにより，段階的に意味解釈結果を作成する [5, 12]．従来の漸進的解析手法では，意味の構成は構文規則の適用に対応しており，その規則には通常の自然言語解析と同様のものを用いることが多い．どのような規則を用いるのかは，意味の構成タイミングに大きく影響を与えるが，早い段階での構成が可能となれば，リアクションを生成するタイミングを早めることができる．実際，アプリケーションシステムによっては，それが対象とするタスク固有の知識を活用することにより，通常の規則によって定められるタイミングよりも早い段階で意味を構成できる場合がある．

そこで本稿では，通常の話し言葉処理とタスクに依存した処理が，入力に対して同時進行的にかつ協調的に振舞う枠組を提案する．通常の話し言葉処理では一般に用いられる文法規則を用いて，また，タスクに依存した処理ではタスク固有の規則を用いて，それぞれ漸進的な解析処理を実行する．それぞれの解析結果は共有データとして記述され，その中で意味の構成が最も進展している結果を，その時点での解とする．タスクに依存した処理の導入により，従来の漸進的解釈に基づく対話インタフェースに比べ，より早い段階で応答できる可能性がある．また，誤った結果が作成されることにより，場合によっては，不適切な応答を返すこともあるが，通常の解析処理では少なくとも文全体の入力が完了した後で正しい解析結果を作成できるため，それらの協調により誤りの修復を容易に行える．

本稿では，本手法を説明するために，マルチモーダル図形エディタ Sync/Draw[18, 11, 9] への応用について述べる．Sync/Draw は，日本語単語の発声やポインティング入力のたびに，それを逐次解析し，結果を図形としてフィードバックする，発話同時理解に基づくマルチモーダル対話の実験システムである．システムのタスクは作図であり，解析部では，作図タスクに依存した解析と通常の日本語話し言葉の解析が行われる．作図タスクに依存した解析により，早い段階でユーザの意図を反映した描画結果を出力するため，ユーザはその結果を逐次確認しながら安心して操作を進めることができる．

2 タスク依存解析に基づく話し言葉処理

本節では，マルチモーダル音声対話システムのための話し言葉処理として，通常の文法を用いた解析と対話タスクに依存した解析との協調に基づく漸進的解析手法について述べる．なお以下では，本手法が対象とするシステムとして，図形エディタ Sync/Draw[9, 11, 18] を想定する．Sync/Draw は，日本語音声とマウスポインティングの入力に対して即座に応答するマルチモーダル対話システムである．認識，解析，描画の 3 つのモジュールから構成されており，それらが入力に対して同時進行的に処理を実行することにより，即時応答を実現している．

2.1 従来の漸進的解析手法の問題点

システムがユーザの発話を同時理解するためには，その解析モジュールは，話し言葉を漸進的に解析する必要がある．これまでに，自然言語を漸進的に解析するための手法がいくつか提案されており [5, 6, 12]，できる限り小さな単位での左から右への処理を実現している．自然言語を解析した結果は，木構造で表現されることが多く，その解析プロセスは，木構造を構成するプロセスとして実現できるが，漸進的解析では，できる限り早い段階で順次，木構造の構成を行う必要がある．早めに構成できれば，それをその時点でのシステムの解とし，それをもとに即座にリアクションを返すことができる可能性がある．

漸進的解析では多くの場合，通常の自然言語解析で用いられる文法と同じようなものが用いられるが，木構造を構成するタイミングは，どのような文法規則を用いるかに大きく依存する．例えば，描画エディタのユーザによる発話

(1) その赤い円の中心を始点とする線をここまで描く

を漸進的に解析する場合，例えば，コンビネータ範疇文法 [16] を用いれば，「その赤い円」に対して，語が入力されるたびに順次木構造を構成することができる[1]．すなわち，「その赤い」が入力された段階で，それに対する木構造を構成することができ，それを用いた照応解決を通して，エディタ上に存在する，ある

[1] 詳細については，[4] を参照されたい．

赤い図形をユーザが意図した参照したい図形であると認識し，即座に適切な応答を行える．

さらに，より即時性の高いフィードバック機能をもつシステムでは，「その赤い円の中心を始点」が発話された時点で，その中心を端点とする線を描くことが求められる可能性がある．1) 始点とは線の一部である，2) 円の中心も線の始点も点である，3) ユーザの目的は作図である，などといった図形や描画に関する知識から，ユーザの意図は上記のような線を描くことであるといった推測が可能だからである．このような応答を可能にするためには，「その赤い円の中心を始点」が発話された時点で，それに対する木構造を構成することにより意味理解を行う必要がある．しかし，通常の自然言語解析に用いる文法を駆使して漸進的解析を行っても，上で示したようなタイミングでの木構造の構成を実現することは困難である．

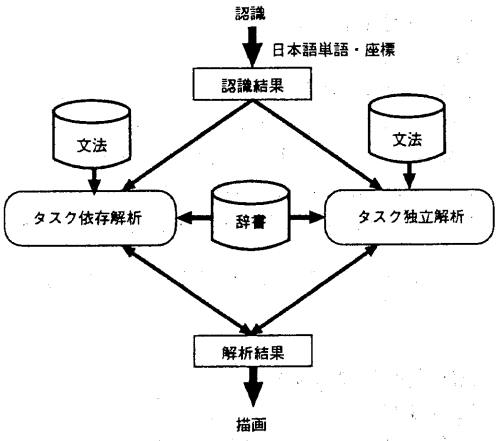


図 1: タスク独立解析とタスク依存解析の協調処理

2.2 タスク依存解析とタスク独立解析

前節で述べた問題を解決するため，適切な即時応答を実現するための話し言葉解析手法が必要である．重要なことは，即時応答タイミングの適切さとは，音声対話システムがどのようなアプリケーションを対象としているのかによって異なるということである．例えば，メール検索システムに対して「松原から」と発話した場合には，システムはその時点で，松原から届いたメールの検索を実行することが望ましいが，地理案内システムでの「名古屋から」という発話に対して，どのような応答を行えばよいかは不明である．この違いはまさに，展開される対話がどのようなタスクを指向しているかによる．すなわち，即時応答機能を備えたシステムは，それが達成すべきタスクに固有の解析を実行する必要がある．Sync/Draw の場合は，タスクは作図であるため，例文 (1) に対して前節で示したような応答が可能となるような解析を要する．

そこで本稿では，作図タスクに固有の解析 (タスク依存解析と呼ぶ) と，タスクに依存しない，通常の日本語解析用の文法規則に基づく解析 (タスク独立解析と呼ぶ) との協調に基づく話し言葉処理手法を提案する．図 1に提案する手法の構成を示す．発声された日本語単語，もしくは，ポインティングされた位置が認識されると即座にタスク依存解析，及び，タスク独立解析を実行する．いずれの解析においても，単一化文法 [15] の枠組に基づいて記述した辞書と文法を用いる．語彙項目は，通常のものにタスク固有の情報を加えたものであり，両解析に共通して用いられる．語彙項目の例を図 4及び 5に示す[2]．素性 dom が，作図タスクに依存した情報を値として保持するものであり，例えば，「線」の語彙項目における dom の値は，線を描くのに必要な情報の集合として記述される．一方，文法には，両解析にそれぞれ固有の文法規則を設ける．タスク独立解析には，日本語話し言葉を通常の方法で処理するために，日本語句構造文法 (JPSG)[3] に準拠した句構造規則を用い，タスク依存解析には，描画や図形に関する知識を活かすことが可能な規則を用いる．図 2及び 3に，それぞれの解析で用いられる句構造規則の一部を示す．なお解析には，上昇型チャート法 [10] を用いる．

タスク依存解析とタスク独立解析は，日本語単語やポインティング座標が認識されるたびに解析を実行し，途中までの入力に対する部分解析結果を随時求め，それを共有のデータとして記録する[3]．このため，複数の部分解析結果が作成されることになるが，描画部はそれらを比較し，最も適切な結果を優先する．即時応答に適した解析結果，すなわち，最も単一化処理が進展した素性構造の意味表現素性 (sem restr) の値をもとに，図を画面上に描画する．

タスク依存解析は主に，作図タスクにおける図形の描画や図形そのものの知識を活用した処理を行っており，それは早めにユーザの意図を推測することに相当する．タスクが定まるとその対話の展開もある程度制限されるため，ユーザの意図を正しく推測でき，早い段階で適切な応答を生成可能であるが，ときには推測に誤ることもある．(1) の「その赤い円の中心を始点」を発話した時点で線を描いたとしても，その後の発話は例えば，「とする線をここに動かす」かもしれないし，「とする線を消す」かもしれない．しかし，タスク独立解析では，少なくとも文全体が入力された後には，正しい解析結果を作成することができる

[2]ここでは簡単のため，次節の説明に必要な素性のみを示す．

[3]大域データのため，異なる解析モジュールで作成された結果でも，解析に用いることができる．

(I) 【下位範疇化規則】
M → C H
⟨ M head ⟩ = ⟨ H head ⟩
⟨ H subcat ⟩ = ⟨ M subcat ⟩ ∪ ⟨ C ⟩
⟨ M dom ⟩ = ⟨ H dom ⟩ ∪ ⟨ C dom ⟩
⟨ M sem index⟩ = ⟨ H sem index⟩
⟨ M sem restr⟩ = ⟨ H sem restr⟩ ∪ ⟨ C sem restr ⟩

図 2: タスク独立解析のための規則の例

(II) 【図形処理規則】
M → I H
⟨ M head ⟩ = ⟨ H head ⟩
⟨ M subcat ⟩ = ⟨ H subcat ⟩
⟨ H dom ⟩ = ⟨ M dom⟩ ∪ ⟨ I ⟩
⟨ M sem index⟩ = ⟨ H sem index⟩
⟨ M sem restr⟩ = ⟨ H sem restr⟩ ∪ ⟨ C sem restr ⟩

図 3: タスク依存解析のための規則の例

ため，ユーザの意図の推測に誤った場合でも，後戻りすることなくそれを修復することができる．

2.3 解析例

Sync/Draw に対する入力を用いた解析例を以下に示す[4]．例 (2) は，線入力を宣言した後でその具体的な位置を指定する操作例である．

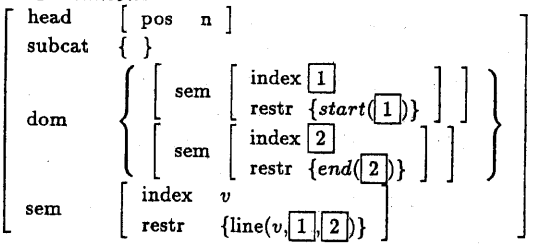
(2) 線を $\nabla_{(10,20)}$ ここからここ $\nabla_{(30,40)}$ までひく

図 6に，(2) の発話に対する解析プロセスと描画結果を示す．横軸は時間の経過を示しており，解析プロセスの部分は，素性構造が作成されるタイミングとどの構造を用いて構成されたか，さらには，その構成に用いた規則を表している．素性構造は簡単のため，その構文範疇を用いて表す．単語またはポインティングのすぐ上の範疇は語彙項目に相当し，それ以外の範疇については，タスク独立解析により作成された場合を実線で，タスク依存解析による生成を点線で示している．図の上部にある素性構造ほど，単一化処理が多く行われたことを示しており，最も上部の素性構造 (四角で囲まれた範疇) の意味表現が，その時点での描画処理に用いられる．

まず，「線を」が入力された時点ではタスク独立解析で処理が実行される．図 4の語彙項目と図 2に示す下位範疇化規則を用いて単一化を行い，「線を」に対する素性構造を作成する．この時点では，「線」，

[4] $\nabla_{(x,y)}$ で座標 (x,y) へのポインティング入力を示す．

「線」の語彙項目



「を」の語彙項目

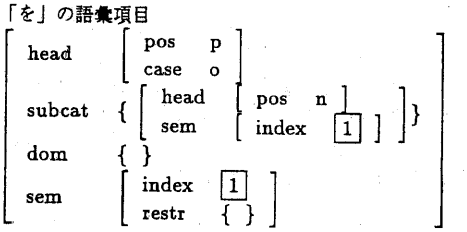
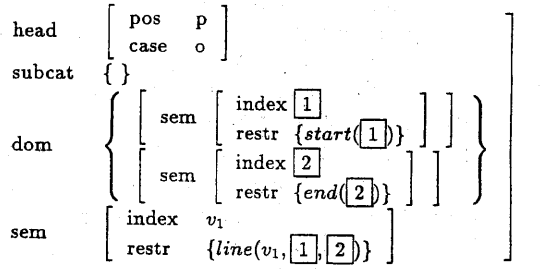


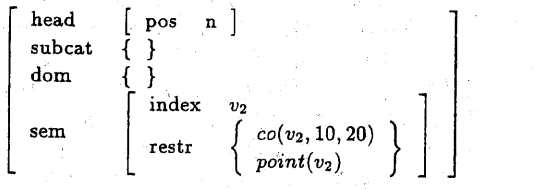
図 4: 語彙項目の例 (1)

「を」，及び「線を」に対する素性構造が生成されているが，「線を」に対するものが最も多くの入力情報を保持するため，それが解となる．すなわち，「線を」に対する素性構造



から，その意味表現 $line(v_1, \boxed{1}, \boxed{2})$ を出力し，それに対して描画処理を実行する (画面上には反映されない)．

次の「$\nabla_{(10,20)}$ ここ」の入力に対しては，複数のモダリティの同期処理が必要となる．マルチモーダル対話の場合，一般に，音声以外のモダリティの入力の意味は，タスクや状況によって異なるため，これをタスク依存解析で処理する．すなわち，図 5の語彙項目と図 3の図形処理規則を用いて単一化し，素性構造



を作成し，画面上に点を描画する．「$\nabla_{(10,20)}$ ここ」と「から」から作成した素性構造の意味表現は，

「$\nabla_{(x,y)}$」の語彙項目

$$\left[\begin{array}{ll} \text{head} & \left[\begin{array}{ll} \text{pos} & \text{ptg} \end{array}\right] \\ \text{subcat} & \{\ \} \\ \text{dom} & \{\ \} \\ \text{sem} & \left[\begin{array}{ll} \text{index} & v \\ \text{restr} & \{co(v,x,y)\} \end{array}\right] \end{array}\right]$$

「ここ」の語彙項目

$$\left[\begin{array}{ll} \text{head} & \left[\begin{array}{ll} \text{pos} & \text{n} \end{array}\right] \\ \text{subcat} & \{\ \} \\ \text{dom} & \{\left[\begin{array}{ll} \text{sem} & \left[\begin{array}{ll} \text{index} & \boxed{1} \\ \text{restr} & \{co(\boxed{1},\boxed{2},\boxed{3})\} \end{array}\right] \end{array}\right]\} \\ \text{sem} & \left[\begin{array}{ll} \text{index} & \boxed{1} \\ \text{restr} & \{\text{point}(\boxed{1})\} \end{array}\right] \end{array}\right]$$

図 5: 語彙項目の例 (2)

$$co(v_2, 10, 20) \wedge point(v_2) \wedge start(v_2)$$

である．タスク独立解析では，これ以上解析を進めることはできないが，タスク依存解析においては，これと「線を」に対する素性構造とを用いて単一化を行うことができ，意味表現

$$line(v_1, v_2, \boxed{2}) \wedge point(v_2) \wedge start(v_2) \wedge co(v_2, 10, 20)$$

を得る．これは始点が定まった線を意味する．

同様に，「$\nabla_{(30,40)}$ ここまで」の入力に対しても，タスク独立解析により終点を定めることができ，さらに，タスク依存解析の結果，始点と終点の定まった線を描画することができる．これは，動詞「引く」が入力される前の段階で，ユーザの意図が線の描画であることを推測したことを意味しており，対話タスクが作図であることによるものである．

図 6の解析プロセスから明らかなように，各時点で最上部に位置する素性構造，すなわち，画面上の描画結果の作成に用いられた素性構造は，タスク依存解析により作成される場合もあれば，タスク独立解析によることもある．これは，通常の話し言葉解析よりも早い段階でユーザの発話に応答していることを意味しており，両解析の協調処理の有効性を示している．

(3) ここ $\nabla_{(10,20)}$ から $\nabla_{(30,40)}$ ここまで線をひく

なお，(3) のような発話がなされた場合にも，「線」が発話された段階で，線の描画を行える．

3 まとめ

本稿では，発話に対する即時応答機能を備えたマルチモーダル対話システムのための話し言葉処理手法として，タスク独立解析とタスク依存解析との協調に基づく枠組を提案した．作図をタスクとするマルチモーダルシステム Sync/Draw の入力を用いて本手法を説明した．作図タスクに関する知識を活用することにより，ユーザの意図を入力途中の早い段階で推測することができ，通常の漸進的解析手法を用いた場合と比較して，より即時性の高いフィードバックの実現が可能となる．この手法を Sync/Draw の解析部として実現することにより，より早い段階での応答が可能となるとともに，これまで扱えなかった複雑な話し言葉にも対処可能となることが予想され，Sync/Draw の能力の向上が期待できる．これまでに発話同時理解に基づく音声対話システムがいくつか提案されているが [2, 7, 13, 17]，今後は，それらの話し言葉処理手法との比較検討を行う予定である．

本稿では，話し言葉処理の即時性の実現に主眼をおいて述べたが，それ以外に必要な要素として，頑健性が挙げられる．すなわち，省略や倒置，言い直しなどといった話し言葉特有の言語現象に対して対処可能な枠組が求められる．本稿で提案した手法は，現在のところそれらを処理することはできないが，タスク独立解析に文法的不適格文の解析機能を付加し，さらに，タスク依存解析と協調することにより，対処可能であると考えられ，これについても今後，検討を要する課題となっている．

参考文献

[1] 安藤ハル, 北原 洋一, 畑岡 信夫：インテリアデザイン支援システムを対象としたマルチモーダルインタフェースの評価, 信学論, Vol.J77-D-II, No.8, pp.1417–1428 (1994).

[2] Ehsani, F., Hatazaki, K., Noguchi, J. and Watanabe, T.: Interactive Speech Dialogue System Using Simultaneous Understanding, *Proc. of 3rd Int. Conf. on Spoken Language Processing*, pp. 879–882 (1994).

[3] 郡司 隆男：日本語句構造文法, 人工知能学会誌, Vol.1, No.2, pp.203–210 (1986).

[4] Haddock, N.J.: Incremental Interpretation and Combinatory Categorial Grammar, *Proc. of 10th Int. Joint Conf. on Artificial Intelligence*, pp. 661–663 (1987).

[5] Haddock, N.J.: Computational Models of Incremental Semantic Interpretation, *Language and Cognitive Processes*, Vol.4, No.3/4, pp. 337–368 (1989).

[6] Inagaki, Y. and Matsubara, S.: Models for Incremental Interpretation of Natural Language. *Proc. of 2nd Symposium on Natural Language Processing*, pp. 51–60 (1995).

[7] 伊藤 慶明, 木山 次郎, 関 進, 小島 浩, 張 建新, 岡 隆一：同時複数話者の会話音声及びジェスチャのリアルタイム統合理解による Novel Interface

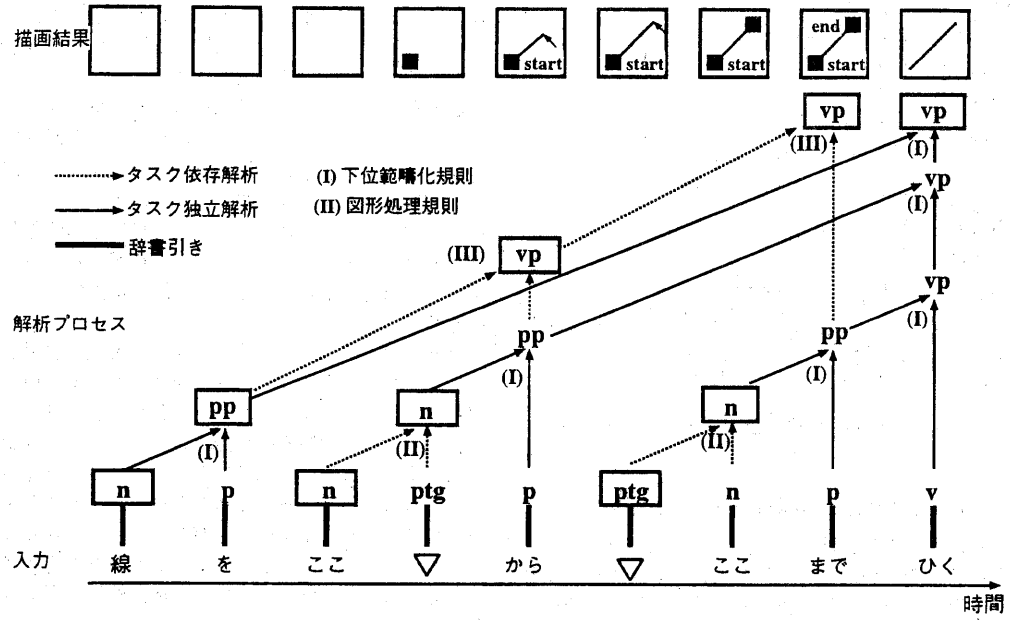


図 6: 例文 (2) の発話に対する解析プロセスと描画結果

System, 情報処理学会研究報告, SLP-7, pp. 17–22 (1995).

[8] 神尾 広幸, 松浦 博, 正井 康之, 新田 恒雄：マルチモーダル対話システム MultiksDial, 信学論, Vol.J77-D-II, No.8, pp.1429–1437 (1994).

[9] 河口 信夫, 松原 茂樹, 外山 勝彦, 稲垣 康善：発話同時理解に基づくマルチモーダル図形エディタ, 情報処理学会研究報告, SLP-22, pp. 1–8 (1998).

[10] Kay, M.: Algorithm Shemata and Data Structures in Syntactic Processing, *Technical Report CSL-80-12*, Xerox PARC (1980).

[11] Matsubara, S., Yamamoto, H., Kawaguchi, N., Inagaki, Y. and Toyama, K.: An Interactive Multimodal Drawing System based on Incremental Interpretation, *Proc. of 15th Int. Joint Conf. on Artificial Intelligence Workshop on Intelligent Multimodal Systems*, pp. 55–62 (1997).

[12] Milward, D. and Cooper, R.: Incremental Interpretation: Applications, Theory, and Relationship to Dynamic Semantics, *Proc. of 15th Int. Conf. on Computational Linguistics*, pp.748-754 (1994).

[13] 中野 幹生, 宮崎 昇, 平沢 純一, 堂坂 浩二, 川端 豪：多重文脈を用いた逐次的な発話理解, 情報処理学会研究報告, SLP22-4, pp. 21–26 (1998).

[14] 西本 卓也, 志田 修利, 小林 哲則, 白井 克彦：マルチモーダル入力環境下における音声の協調的利用 – 音声作図システム S-tgif の設計と評価, 信学論 (DII), Vol.J79-D-II, pp.2176–2183 (1996).

[15] Shieber, S.M.: *An Introduction to Unification-based Approaches to Grammar*, CSLI Lecture Notes Series 4, CSLI, Stanford University (1987).

[16] Steedman, M.J.: Combinatory Grammar and Human Sentence Processing, In J.L. Garfield (ed.), *Modurality in Knowledge Representation and Natural Language Understanding*, pp. 187–205, MIT Press (1987).

[17] 竹林 洋一：音声自由対話システム TOSBURG II – ユーザ中心のマルチモーダルインタフェースの実現に向けて –, 信学論 (DII), Vol.J77-D-II, No.8, pp.1417–1428 (1994).

[18] 山本 博之, 松原 茂樹, 河口 信夫, 稲垣 康善：自然言語の漸進的解釈に基づくマルチモーダル対話システム, インタラクティブシステムとソフトウェア *IV*, pp. 121-130, 近代科学社(1996).