Massively Parallel Parsing in ΦDMDIALOG: Integrated Architecture for Parsing Speech Inputs

Hiroaki Kitano, Teruko Mitamura and Masaru Tomita
Center for Machine Translation
Carnegie Mellon University
Pittsburgh, PA 15213 U.S.A.

Abstract

This paper describes the parsing scheme in the ΦDMDIALOG speech-to-speech dialog translation system, with special emphasis on the integration of speech and natural language processing. We propose an integrated architecture for parsing speech inputs based on a parallel marker-passing scheme and attaining dynamic participation of knowledge from the phonological-level to the discourse-level. At the phonological level, we employ a stochastic model using a transition matrix and a confusion matrix and markers which carry a probability measure. At a higher level, syntactic/semantic and discourse processing, we integrate a case-based and constraint-based scheme in a consistent manner so that a priori probability and constraints, which reflect linguistic and discourse factors, are provided to the phonological level of processing. A probability/cost-based scheme in our model enables ambiguity resolution at various levels using one uniform principle.

1. Introduction

This paper discusses a method of integrating speech recognition and natural language processing. In order to develop speech-based natural language systems such as a speech-to-speech translation system and a speech input natural language interface, an integration of speech recognition and natural language processing is essential, because it improves the recognition rate of the speech inputs. Improvement of the recognition rate can be attained by an integration of natural language processing with speech recognition, providing a more appropriate assignment of *a priori probability* to each hypothesis and imposes more constraints to reduce search space. Thus, the quality of the *language model* is an important factor. Since our goal is to create accurate translation from speech input, a sophisticated parsing and discourse understanding scheme are necessary. We propose an architecture for parsing speech inputs that integrates speech (phonological-level processing) and natural language processing with full syntactic/semantic analysis and discourse understanding.

In our system, we assume that an acoustic processing device provides a symbol sequence for a given speech input. In this paper, we assume that a phoneme-level sequence is provided to the system[1]. The phoneme sequence given from the phoneme recognition device contains substitution, insertion and deletion of phonemes, as compared to a correct transcription which contains only expected phonemes. We call such a phoneme sequence a *noisy phoneme sequence*. The task of phonological-level processing is to activate a hypothesis as to the correct phoneme sequence from this noisy phoneme sequence. Inevitably, multiple hypotheses can be generated due to the stochastic nature of phoneme recognition errors. Thus, we want each hypothesis to be assigned a measure of its being correct. In the stochastic models of speech recognition, a probability of each hypothesis is determined by $P(y|h) \times P(h)$. $P(y|h)$ is the probability of a series of input sequence being observed when a hypothesis h is articulated. $P(h)$ is an a priori probability of the hypothesis derived from the language model. Apparently, when phonological-level processing is the same, the system with a sophisticated language model attains a higher recognition rate, because a priori probability differenciates between hypotheses of high acoustic similarity which would otherwise lead to confusion. At the same time, we want to eliminate less-plausible hypotheses as early as possible so that the search space is kept within a certain size. We use syntactic/semantic and discourse knowledge to impose constraints which reduce search space, in addition to the probability-based pruning within the phonological level.

[1] We use Matsushita Institute's Japanese speech recognition system[Morii et. al., 1985] for a current implementation.

2. ΦDMDIALOG Project

2.1. Overview

ΦDMDIALOG is a speech-to-speech dialog translation system based on a massively parallel computational model [Kitano, 1989b] [Kitano et. al., 1989b] [2]. It accepts speaker-independent continuous speech inputs. Some of the significant features of ΦDMDIALOG include:

I. Use of a hybrid parallel paradigm as a basic computational scheme, which is an integrated model of a direct memory access (DMA) type of a massively parallel marker passing scheme and a connectionist network;

II. Dymanic utilization of knowledge from morphophonetics to discourse by distributively encoding this knowledge in a memory network on which actual computations are performed;

III. Integration of case-based and constraint-based processing to capture linguistically complex phenomena without losing cognitive realities;

IV. A cost-based ambiguity resolution scheme which applies to all levels of ambiguity (from phoneme recognition to discourse context selection)[Kitano et. al., 1989a];

V. Almost concurrent parsing and generation, so that a part of a sentence can be translated before the whole sentence is parsed[Kitano, 1989a].

The philosophy behind our model is to view parsing as a process on a dynamic system where the law of energy conservation, entropy production and other laws of physics can be effective analogies. We also demand that our model be consistent with psycholinguistic studies.

2.2. A Baseline Algorithm

We employ the hybrid parallel paradigm in order to model two distinct aspects of the parsing: information building and hypothesis selection. In the hybrid parallel paradigm, a parallel marker-passing scheme and a connectionist network are integrated and computations are performed directly in a memory network. Knowledge from the morphophonetic level to the discourse level is represented as a memory network which is consists of nodes and links. Several types of nodes are in the memory network.

Concept Sequence Class (CSC) captures configurational patterns of linguistic phenomena such as phoneme sequences, concept sequences and plan sequences. CSCs have an internal structure. The internal structure is composed of a label, IS-A links, a sequence, presuppositions, effects, and constraint equations. This structure is same for all CSCs except CSCs in the phonological layer.

Concept Class (CC) represents concepts such as phonemes, concepts, and plans.

Concept Instance (CI) are instances of CCs. They are used to represent discourse entities[Webber, 1983] and instance of utterances.

Nodes are connected by labelled links. Abstraction links (IS-A) and compositional links (PART-OF) are typical types of links. The memory network is organized in a hierarchical manner. There are hierarchies of nodes representing concepts from specific instances (using CIs) to general concepts (using CCs) and hierarchies of structured nodes representing relations of concepts which are indexed into relevant concepts and specific instances (using CIs and their links). When CSCs represent specific cases, they are already co-indexed to the specific instances in the memory network. Abstract CSCs hold various constraints described as constraint equations, presuppositions and effects. These abstract CSCs are instantiated during parsing and newly created specific CSCs are indexed into the memory network as cases of utterance. Parsing with abstract CSCs is computationally more expensive than parsing with cases, but it maintains productivity of the knowledge.

Three types of markers (A-, P-, and C-Markers) are used for parsing. Two other types of markers, G- and V-Markers are used for generation; thus they are not described in this paper.

Activation Markers (A-Markers) contain information including discourse entities, features and cost. They propagate upward through abstraction links.

Prediction Markers (P-Markers) predict possible next activations. They contain binding lists (a list of role-instance pairs binded so far), a measure of cost, and linguistic and pragmatic constraints.

Contextual Markers (C-Markers) are used as an alternative to a connectionst network and indicate contextual priming. C-Markers are not used when the connectionist network is fully deployed.

---

[2] Φ indicates that our system is a speech input system. This notation is a tradition of the Center for Machine Translation. DM implies that the system was initially designed as a direct memory access (DMA) based system. However, our system evolved differently from the DMAP[Riesbeck and Martin, 1985] and now DM implies both DMA and *dynamics modeling* which reflects our philosophy of viewing a cognitive process as a dynamic process governed by the laws of physics. DIALOG means that our system translates dialogs.

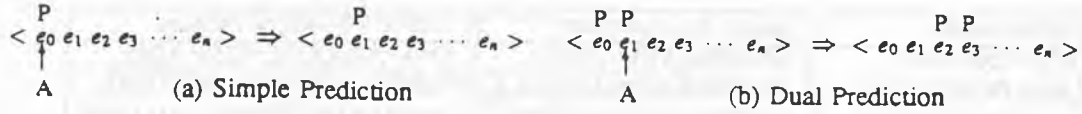

Figure 1: Movement of P-Markers

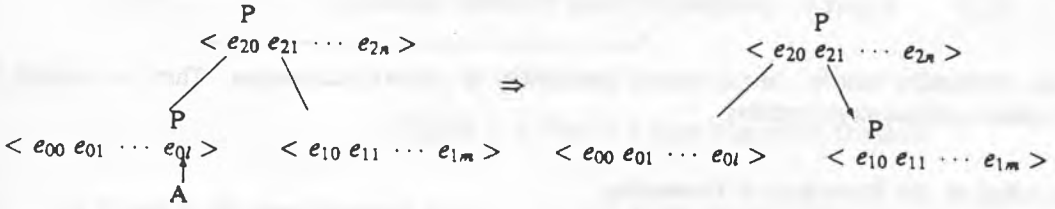

Figure 2: Movement of P-Markers in Layered Sequences

A basic cycle of our algorithm is as follows:

1. **Activation:**
   For each input symbol, a corresponding node is activated and an A-Marker is created. A unit of input may be either a phoneme or a word, depending on the input device. The A-Marker is passed up through IS-A links. The A-Marker contains information relevant to the processing of that layer.

2. **A-P-Collision:**
   When an A-Marker and a P-Marker collide at a certain element of a CSC, the P-Marker is moved to the next possible concept element of the CSC. At this stage, constraints are checked.

3. **Prediction:**
   As a result of moving P-Markers to the next possible element of the CSC, predictions are made describing possible next inputs.

4. **Recognition (Network Modification and Information Propagation):**
   When the CSC is accepted, (1) the memory network may be modified as a side-effect, and (2) an A-Marker containing aggregated information is passed up through IS-A links.

The movements of P-Markers on a CSC are illustrated in figure 1. In (a), a P-Marker (initially located on e_0) is hit by an A-Marker and moved to the next element. In (b), two P-Markers are used and moved to e_2 and e_3. In the dual prediction, two P-Markers are placed on elements of the CSC (on e_0 and e_1). This dual prediction is used for phonological processing.

Figure-2 shows movement of a P-Marker on the layers of CSCs. When the P-Marker at the last element of the CSC gets an A-Marker, the CSC is accepted and an A-Marker is passed up to the element in the higher layer CSC. Then, a P-Marker on the element of the CSC gets the A-Marker, and the P-Marker is moved to the next element. At this time, a P-Marker which contains information relevant to the lower CSC is passed down and placed on the first element of the lower CSC. This is a process of accepting one CSC and predicting the possible next word and syntactic structure.

3. Phonological Parsing

This section describes phonological-level activities. We assume a noisy phoneme sequence, as shown in Figure 3, to be the input of the phonological-level processing. In order to capture the stochastic nature of speech inputs, we adopt a probabilistic model similar to that used in other speech recognition research. First, we describe a simple

kaigi ni sanka shitai nodesu	*youshi ha arimasuka*	*oname wo onegai shimasu*
DAI*I*IPAUTAQPAINO*EKU	BJOHIRAARI*ATAWA	O*A*AEJOORE*EISI*AS@
BAII*IPAA=KAS@PAINODUSU	JOSJUWAARINAOQZAA	WO*A*AEJOORE*EEHJANA
BAII*I*IPAU=KAIQPAI*O*ESU	IOUSIWAARIMAUQKA	WONA*AEJOBO*E*EIHJAH@
KAIIMIPAA=KAS@PEEI*ODESU	JOOSIHAKARI*AUQKA	O*A*AEJO*O*E*EEISINAKU
KAI*I*IPAA=ZAS@PAIWO*USJU	IOOSJUWAWARI*AACA	O*A*AEJOO*E*EEIHJAZU

Figure 3: Examples of Noisy Phoneme Sequences

model using a static probability matrix. In this model, probability is context-independent. Then, we extend the model to capture context-dependent probability.

3.1. The Organization of the Phonological Processing

The algorithm described as a baseline algorithm is deployed on phonetic-level knowledge. In the memory network, there are CSCs representing the phoneme sequence for each lexical entry. The dual prediction method is used in order to handle deletion of a phoneme.

We use a probabilistic model to capture the stochastic nature of speech processing. Probability measures involved are: a priori probability given by the language model, a confusion probability given by a confusion matrix, and a transition probability given by a transition martix.

A priori probability is derived from the language model and is a measure of which phoneme sequence is likely to be recognized. A method of deriving a priori probability is described in the section on syntax/semantic parsing and discourse processing.

A confusion matrix defines the output probability of a phoneme when an input symbol is given. Given an input sign i_i, the confusion matrix a_{ij} determines the probability that the sign i_i will be recognized as a phoneme p_j. It is a measure of the distance between symbols and phonemes as well as a measure of the cost of hypotheses that interpret the symbol i_i as the phoneme p_j. In the context-dependent model, the confusion matrix will defined as a_{ijk} which gives a probability of a symbol i_i to be interpreted as a phoneme p_j at a transition t_k. We call such matrix a *dynamic confusion matrix.*

A transition matrix defines the transition probability which is a probability of a symbol i_{i+1} to follow a symbol i_i. For an input sequence $i_0\ i_1\ \cdots\ i_n$, the a priori probability of transition between i_0 and i_1 is given by b_{i_0,i_1}. Since we have a finite set of input symbols, each transition can be indexed as t_k. The transition probability and the confusion probability are intended to capture the context-dependency of phoneme substitutions – a phenomena whereby a certain phoneme can be actually articulated as other phonemes in certain environments.

3.2. Context-Independent Model

First, we explain our algorithm using a simple model whose confusion matrix is context-independent. Later, we describe the context-dependent model which uses a dynamic confusion matrix. Initially, P-Markers contain a priori probability (π_l) given by the language model. In ΦDMDIALOG, the language model reflects full natural language knowledge from syntax/semantics to discourse. The P-Markers are placed on each first and second element of CSCs representing expected phoneme sequences. For an input symbol i_i, A-Markers are passed up to all phoneme nodes that have a probability(b_{ij}) greater than the threshold (Th). When a P-Marker, which is at i-th element, and an A-Marker collide, the P-Marker is moved to the i+1-th and i+2-th elements of the sequence (This is a dual prediction). When the next input symbol i_{i+1} generates an A-Marker that hits the P-Marker on the i+1-th element, the P-Marker is moved using the dual prediction method. The probability density measure computed on the P-Marker is as follows:

$$ppm(i) = ppm(i-1) \times a_{i_{i-2},i_{i-1}} \times b_{p_{i-2},i_{i-2}} \quad (1)$$

$$ppm(0) = \pi_l \quad (2)$$

where $ppm(i)$ is a probability measure of a P-Marker at the i-th element of the CSC which is a probability of the input sequence being recognized as a phoneme sequence traced by the P-Marker.

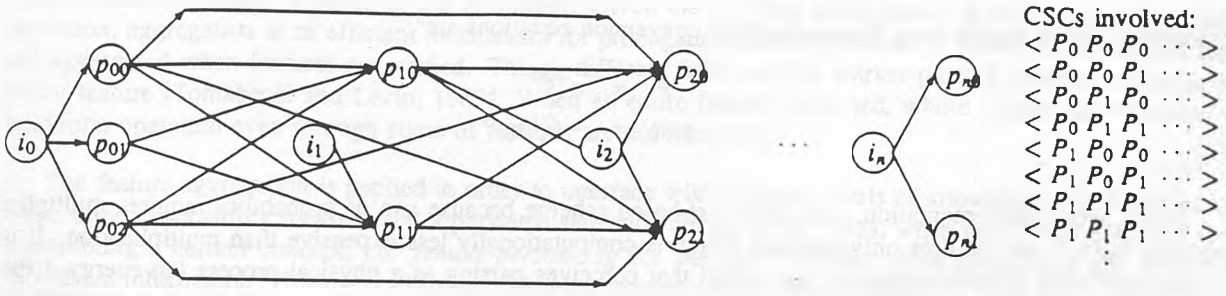

Figure 4: A Part of a State-Transition Diagram

In Figure-4, an input sequence is $i_0\ i_1\ \cdots\ i_n$. p_{ij} in the diagram denotes a phoneme P_j at i-th element of the CSC. p_{ij} is a state rather than an actual phoneme, and P_j in the CSC refers to the actual phoneme. P-Markers at p_{00}, p_{01}, p_{02}, P-Markers on the 0-th element of the CSCs referring P_0, P_1, and P_2, respectively, are hit by A-Markers. Eventually, P-Markers are moved to the next element of CSCs. For instance, p_{00} will move to $p_{10}, p_{11}, p_{20}, p_{21}$ depending on which CSC the P-Marker is placed on. Probabilities are computed with each movement. A P-Marker at p_{11} has the probability π_0. When the P-Marker received an A-Marker from i_1, the probability is re-computed and it will be $\pi_0 \times b_{i_0,p_{00}} \times a_{p_{i_0},p_{i_1}}$. Transitions such as $p_{00} \rightarrow p_{21}$ and $p_{00} \rightarrow p_{20}$ insert an extra phoneme which does not exist in the input sequence. Probability for such transitions are computed in such a way as: $\pi_0 \times b_{i_0,p_{00}} \times a_{i_0,\phi} \times b_{i_2,p_{20}} \times a_{\phi,i_2}$. A P-Marker at p_{10} does not get an A-Marker from i_1 due to the threshold. In such cases, a probability measure of the P-Marker is re-computed as $\pi_0 \times b_{i_0,p_{00}} \times a_{i_0,noise}$. This represents a decrease of probability due to an extra input symbol.

P-Markers at the last element (p_n) and one before the last (p_{n-1}) are involved in the word boundary problem. When a P-Marker at p_n is hit by an A-Marker, the phoneme sequence is accepted and an A-Marker that contains the probability and the phoneme sequence is passed up to the syntactic/semantic-level of the network. Then, the next possible words are predicted using syntactic/semantic knowledge, and P-Markers are placed on the first and the second element of the phoneme sequence of the predicted words. When a P-Marker at p_{n-1} is hit by an A-Marker, the P-Marker is moved to p_n and, independently, the phoneme sequence is accepted, due to the dual prediction, and the first and the second elements of the predicted phoneme sequences get P-Markers.

3.3. The Context-Dependent Model

The context-dependent model can be implemented by using the dynamic confusion matrix. The algorithm described above can be applied with some modificaitions. First, A-Markers are passed up to phonemes whose maximun output probability is above the threshold. Second, output probability used for probability calculation is defined by the dynamic confusion matrix.

$$ppm(i) = ppm(i-1) \times a_{i_{i-2},i_{i-1}} \times b_{p_{i-2},i_{i-2},k} \qquad (3)$$

where k denotes a transition from i_{i-2} to i_{i-1}. It is interesting that our context-dependent model is quite similar to the *Hidden Markov Model (HMM)* when the transition of the state of P-Markers are synchronously determined by, for example, certain time intervals. We can implement a forward-passing algorithm and the Viterbi algorithm [Viterbi, 1967] using our model. This implies that when we decide to employ the HMM as our speech recognition model, instead of a current speech input device, it can be implemented within the framework of our model.

3.4. Probability Cost Equality

Since we have been using the cost-based ambiguity resolution scheme [Kitano et. al., 1989a], the equivalency of the probabilistic approach and the cost-based approach need to be discussed. Our motivation in introducing the cost-based scheme was to perceive parsing as a dynamic process. Thus the hypothesis with the least cost, hence minimum workload, is selected as the best hypothesis. When a stochasity is introduced, the process that requires more workload is less likely to be chosen. Thus, qualitatively, higher probability means less cost and lower

probability means higher cost. Probability/cost conversion equations are[3]:

$$P = e^{\frac{-cost}{C}} \quad (4)$$

$$cost = -C \log P \quad (5)$$

In the actual implementation, we use a cost-based scheme because use of probability requires multiplication, whereas use of cost requires only addition which is computationally less expensive than multiplication. It is also a straightforward implementation of our model that perceives parsing as a physical process (an energy dispersion process). Thus, in the cost-based model, we introduce an *accumlated acoustic cost (AAC)* as a measure of cost which is computed by:

$$aac(i) = aac(i-1) + cc_{i_{i-1},p_{i-1}} + tc_{i_{i-2},i_{i-1}} - pe \quad (6)$$

where $aac(i)$, $cc_{i_{i-1},p_{i-1}}$, $tc_{i_{i-2},i_{i-1}}$ and pe are an AAC measure of the P-Marker at i-th element, confusion cost between i_{i-1} and p_{i-1}, transition cost between i_{i-2} and i_{i-1}, and phonetic energy, respectively. Phonetic energy reflects an influx of energy from external acoustic energy.

4. Syntactic/Semantic Parsing

Unlike most other language models employed in speech recognition research, our language model is a complete implementation of a natural language parsing system. Thus, complete semantic interpretations, constraint checks, ambiguity resolution and discourse interpretations are performed. The process of prediction is a part of parsing in our model, thereby attaining an integrated architecture of speech input parsing. In syntactic/semantic processing, the central focus is on how to build the informational content of the utterance and how to reflect syntactic/semantic constraints at phonological-level activities. Throughout the syntactic/semantic-level and discourse-level, we use a method to fuse constraint-based and case-based approaches. In our model, the difference between a constraint-based process and a case-based process is a level of abstraction; the case-based process is specific and the constraint-based process is more abstract. The constraint-based approach is represented by various unification-based grammar formalisms [Pollard and Sag, 1987] [Kaplan and Bresnan, 1982]. We use semantic grammar which combines syntactic and semantic constraints[4]. In our model, propagation of features and unification are conducted as a *feature aggregation* by A-Markers and *constraints satisfaction* performed by operations involving P-Markers. The case-based approach is a basic feature of our model. Specific cases of utterances are indexed in the memory network and reactivated when similar utterances are given to the system. One of the motivations for the case-based parsing is that it encompasses *phrasal lexicons* [Becker, 1975][5]. The scheme described in this section is applied to discourse-level processing and attains an integration of the syntactic/semantic-level and the discourse-level.

4.1. Feature Aggregation

Feature aggregation is an operation which combines features in the process of passing up A-Markers so that minimal features are carried up. Due to the hierarchical organization of the memory network, features which need to be carried by A-Markers are different depending on which level of abstraction is used for parsing. When knowledge of cases is used for parsing, features are not necessary because this knowledge is already indexed to specific discourse entities. Features need to be carried when more abstract knowledge is used for parsing. For example, the parsing of a sentence *She runs* can be handled at different levels of abstraction using the same mechanism. The word *she* refers to a certain discourse entity so that very specific case-based parsing can directly access a memory which recalls previous memory in the network. Since previous cases are indexed into specific discourse entities, the activation can directly identify which memory to recall. When this word *she* is processed in a more abstract level such as PERSON, we need to check features such as number and gender. Thus, these features need to be contained in the A-Marker. Further abstraction requires more features to be contained in the A-Marker. Therefore, the case-based process and the constraint-based process is treated in one mechanism. Aggregation is a cheap operation since it simply adds

---

[3]The equations are based on the Maxwell-Boltzmann distribution $P = e^{\frac{-\epsilon}{kT}}$.

[4]This does not preclude use of unification grammar formalism in our system. In fact, we are now developing a cross-compiler that compiles grammar rules written in LFG into our network. Designing of a cross-compiler from HPSG to our network is also underway.

[5]Discussions on benefits of phrasal lexicons for parsing and generation are found in [Riesbeck and Martin, 1985] [Hovy, 1988].

new features to existing features in the A-Marker. Given the fact that unification is a computationally expensive operation, aggregation is an efficient mechanism for propagating features because it ensures only minimal features are aggregated when features are unified. This is different from another marker-passing scheme which carries an entire feature [Tomabechi and Levin, 1989]. When an entire feature is carried, whole features are involved in the unifiction operation even through some of features are not necessary.

The feature aggregation is applied in order to interface with different levels of knowledge. At the phonological level, only a probability measure and a phoneme sequence are involved. Thus, when an A-Marker hits a CC node representing a certain concept, i.e. *female-person-3sg* for *she*, the A-Marker does not contain any linguistically significant information. However, when the A-Marker is passed up to more abstract CC nodes, i.e. *person*, linguistically significant features are contained in the A-Marker and unnecessary information is discarded. When a sentence is analyzed at the syntactic/semantic-level, a propositional content is established and is passed up to the discourse-level by an A-Marker, and some linguistic information which is necessary only within the syntactic/semantic-level is discarded.

4.2. Constraint Satisfaction

Constraint is a central notion in modern syntax theories. Each CSC has *constraint equations* which define the constraints imposed for that CSC depending on their level of abstraction. CSCs representing specific cases do not have contraint equations since they are already instanciated and the CSCs are indexed in the memory network. The more abstract the knowledge is the more they contain constraint equations. Feature structures and constraint equations interact in two stages. At the prediction stage, if a P-Marker placed on the first element of the CSC already contains a feature structure that is non-nil, the feature structure determines, according to the constraint equations, possible feature structures of A-Markers that subsequent elements of the CSC can accept. At an A-P-Collision stage, a feature structure in the A-Marker is tested to see if it can meet what was anticipated. If the feature structure passes this test, information in the A-Marker and the P-Marker is combined and more precise predictions are made on what can be acceptable in the subsequent element. For *She runs*, we assume a constraint equation *(AGENT NUM = ACTION NUM)* associated with a CSC, for example, <AGENT ACTION>. When a P-Marker initially has a feature structure that is nil, no expectation is made. In this example, at an A-P-Collision, an A-Marker has a feature structure containing (NUM = 3s) constraints for the possible verb form which can follow, because the feature in the A-Marker is assigned in the constraint equation so that *(AGENT NUM 3s)* requires *(ACTION NUM 3s)*. This guarantees that only a verb form *runs* can be legitimate[6]. When predicting what comes as a *ACTION*, P-Markers can be passed down via IS-A links and only lexical entries that meet *(ACTION NUM 3s)* can be predicted. When we need to relax grammatical constraints, P-Markers can be placed on every verb form, but assign higher a priori probabilities for those which meet the constraint. A unification operation can be used to conduct operations described in this section. As a result of parsing at the syntactic/semantic-level, the propositional content of the utterance is established. Since our model is a memory-based parsing model, the memory network is modified to reflect what was understood as a result of previous parsing.

4.3. Prediction

From the viewpoint of predicting the next hypothesis at the phonological level, case-based parsing provides the most specific prediction and gives high a priori probability. Prediction by more abstract knowledge provides less specific predictions and gives weaker a priori probability compared to case-based prediction. Thus, we have a set of hypotheses with strong preferences predicted by the case-based process and a set of hypotheses (this includes hypotheses predicted by the case-based process) predicted by the constraint-based process. Of course, the strength of the preference is dependent on the level of abstraction the parsing has required. Even in the constraint-based process, if the level of abstraction is low, the prediction has strength comparable to the case-based prediction.

5. Integration of Discourse Knowledge

At the discourse-level, the focus is on how to recognize the intention of the utterance, interpret discourse phenomena and predict next possible utterances. ΦDMDIALOG uses discourse knowledge such as (1) discourse plans, and (2)

[6] When we use abstract notation such as NP or VP, the same mechanism applies and captures linguistic phenomena.

discourse entities and their relations. We use hierarchical discourse plan sequences, represented by CSCs[7], to represent and provide specificity as well as productivity of discourse plans. Hierarchical discourse plan sequences represent possible sequences of utterance plans which may be actually performed by each speaker. Plan hierarchies are organized for each participant of the dialog in order to capture complex dialog often taking place in a mixed-initiative dialog. Each element of the plan sequence represents a domain-specific instance of a plan or an utterance type [Litman and Allen, 1987] which can be dynamically derived from abstract dialog knowledge and domain knowledge. Abstract plan sequences are close to plan schemata described in [Litman and Allen, 1987] since they represent very generic constraints as well as the relationship between an utterance and a domain plan. There is also knowledge for the discourse structure[Cohen and Fertig, 1986] [Grosz and Sidner, 1985]. When an element of the plan sequence of this abstraction is activated, the rest of the elements of the plan sequence have constraints imposed which are derived from the information given to the activated elements. This ensures coherence of the discourse. When a plan sequence case is activated, it simply predicts the next plan elements because these specific plan sequences are regarded as records of past cases and, thus, most constraints are already imposed and the sequence is indexed according to the specific constraints. In addition, use of order constraints of CSC representations allows us to handle order-freeness of subdialog conversations. Furthermore, unlike scripts or MOPs[Schank, 1982], a plan sequence has an internal structure which enables our model to impose constraints which ensure coherency of the discourse processing.

As a result of the discourse understanding, possible next utterances can be predicted. P-Markers are passed down to nodes representing these utterances. Eventually, they reach the phonological level and give a priori probability to each hypothesis. Similar to predictions from syntactic/semantic-level, the strength of the prediction is dependent upon the level of abstract knowledge involved.

6. A Cost-based Ambiguity Resolution Scheme

A cost-based disambiguation scheme is a method of evaluating each hypothesis based on the cost assigned to it. Costs are added when (1) phonemes are replaced, inserted, or dropped during recognition of noisy speech inputs (we use a cost converted from a probability measure at the phonological-level), (2) a new instance is created, (3) a concept without contextual priming is used, or (4) constraints are assumed when using CSCs. Costs are subtracted when (1) a concept with discourse prediction is used, or (2) a concept with contextual priming is used. Basic equations are:

$$CSC_i = \sum_j CC_{ij} + \sum_k constraints_k + bias_i \quad (7)$$

$$CC_j = LEX_j + instantiateCI - priming_j \quad (8)$$

$$LEX_l = -C \log P \quad (9)$$

where CC_{ij}, $constraints_k$, $bias_i$ denote a cost of the j-th element of CSC_i, a cost of assuming the k-th constraints, and the lexical preference of CSC_i, respectively. LEX_j, $instantiateCI$, $priming_j$ denote a cost of the lexical node LEX_j, a cost of creating new CI by referential failure, and contextual priming, respectively. LEX_j is a cost converted from the probability measure at the phonological level as described earlier. The accumlated acoustic cost, computed by the equation (6), can be used instead of converting probability by equation (9). Then, the cost-based scheme is adopted at every level of processing. In the cost-based disambiguation scheme, we choose the least costly hypothesis based on the above equations.

Our model parses utterances under a given context. Thus, the cost assigned to a certain hypothesis is not always the same. It is dependent on the context; that is, the initial conditions of the system when the utterance is entered. The initial condition of the system is determined based on the previous course of discourse. The major factors are the state of the memory network modified as a result of processing previous utterances, contextual priming, and predictions from discourse plans. The memory network is modified based on the knowledge conveyed by the series of utterances in the discourse as described briefly in the previous section. Contextual priming is imposed either by using a C-Marker passing or by a connectionist network. The mechanism of assigning preference is based on top-down prediction using discourse knowledge. Such prediction provides a priori probability π_l at the phonological-level.

[7]This means that order-strict or order-free constraints apply in determining the order of the plan sequence.

The cost-based ambiguity resolution scheme is applied to the reference problem including definite and indefinite reference, pronoun reference, etc. We use activation/cost-based reference where each reference hypothesis incures cost and the least-cost hypothesis will be selected. The cost for each hypothesis is computed based of activation levels of each discourse entities and semantic restrictions. The method does not assume a layered network [Tomabechi and Levin, 1989] and, thus, we can coherently handle problems including the reference to the related objects.

7. Preliminary Evaluations and Discussions

Currently, ΦDMDIALOG is being tested on the conference registration domain based on simulated telephone conversation experiments by ATR. The use of dialog-level knowledge has proven to be effective in in reducing the perplexity of the task. We took as an example a small test set from the ATR corpus, and the perplexity of this task with no prediction knowledge was 247.0. Using sentential level knowledge this figure was reduced to 19.7, and using dialog level knowledge it was reduced to 2.4. However, the problem is that (1) the domain of our experiment is relatively small, and (2) when we cover more complex discourse, prediction from the discourse-level may be less specific. We are now evaluating our model with larger test sets.

We employ the probabilistic model for the following reason: the use of phonological knowledge alone, such as phonological rules and distinctive feature theory, cannot sufficiently cope with the stochastic nature of speech recognition. However, phonological knowledge would be useful for analyzing and estimating probability matrices. By contrasting feature types, such as voicing, instead of collecting all the phonemic data, we would reduce the amount of data needed for building the probability matrices[Church, 1987].

The hierarchical organization of the memory network is a key feature in integrating constraint-based and case-based processing. Although we suffer from some overhead by concurrently parsing one sentence at different levels of abstraction, the capability of handling both specific and abstract knowledge in a consistent manner seems more significant. The feature aggregation method is a useful technique to keep overhead to a minimum.

The implementation of ΦDMDIALOG on a parallel machine is an interesting topic. We believe the benefits of our model can be best explored with parallel machines and that its implementation may be relatively straightforward. Actually, a part of our model has been implemented on a custom VLSI[Kitano, 1988].

8. Related Works

Several efforts have been made to integrate speech and natural language processing. [Tomabechi et. al., 1988] attempts to extend the marker-passing model to speech input. Their model uses *environment* without probabilistic measure which would allow environmental rules to be applied. Since misrecognitions are somewhat stochastic, lack of the probability measure seems a shortcoming in their model. The MINDS system [Young et. al., 1989] is an attempt to integrate speech and natural language processing implementing layered prediction. They reported that use of layered prediction involving discourse knowledge reduced the perplexity of the task. This is consistent with our claim. [Church, 1987] discusses speech recognition using phonetic knowledge such as environment and a distinct feature matrix. We share similar motivations, but we try to incorporate this knowledge in a probabilistic model. [Saito and Tomita, 1988] [Kita et. al., 1989] and [Chow and Roukos, 1989] are examples of approaches to integrate speech with unification-based parsing, but, unfortunately, discourse processing has not been incorporated. Marker-passing models of parsing such as [Riesbeck and Martin, 1985] and [Tomabechi and Levin, 1989] captured only one side of parsing (case-based or constraint-based), in contrast to our model which incorporates both aspects in one scheme.

9. Conclusion

This paper describes a method of speech-natural language integration in ΦDMDIALOG. The probability/cost-based model is used to capture the stochastic nature of speech inputs. The language model in our model is a parser itself and directly connected to the phoneme processing by means of cost measures, a priori probability, and constraints to limit search space. Addition of the discourse understanding scheme further improved the power of the language model to constrain and predict phonological processes. As a result, reduction of the perplexity was observed and the recognition rate was improved. Feature aggregation in the hierarchically organized memory network was a useful scheme to integrate case-based and constraint-based parsing. The parallel marker-passing approach seems a viable alternative for designing an integrated architecture for parsing speech inputs.

Acknowledgement

We would like to thank members of the Center for Machine Translation for useful discussions. Especially, Hideto Tomabechi, Hiroaki Saito and Jaime Carbonell helped us with insightful advice. Discussions on speech recognition with Sheryl Young and Wayne Ward were especially useful. Lyn Jones was patient enough to proofread this paper for us. We also would like to thank ATR Interpreting Telephony Laboratories for allowing us to use a corpus of the conference registration domain for our research. Matsushita Research Institute is allowing us to use their speech recognition system.

Appendix: Implementation

ΦDMDIALOG has been implemented on IBM-RT-PC which runs CMU-CommonLisp on the Mach operating system and HP-9000 runs HP-CommonLisp. Speech recognition and synthesis devices (Matsushita Research Institute's Japanese speech recognition device and DECTalk) are connected to perform real-time speech-to-speech translation.

References

[Becker, 1975] Becker, J. D., *The Phrasal Lexicon*, Bolt, Beranek and Newman Technical Report 3081, 1975.

[Chow and Roukos, 1989] Chow, Y.L. and Roukos, S., "Speech Understanding using a Unification Grammar," In *Proc. of ICASSP- IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1989.

[Church, 1987] Church, K., *Phonological Parsing in Speech Recognition*, Kluwer Academic Publishers, 1987.

[Cohen and Fertig, 1986] Cohen, P. and Fertig, S., "Discourse Structure and the Modality of Communication," *International Symposium on Prospects and Problems of Interpreting Telephony*, 1986.

[Grosz and Sidner, 1985] Grosz, B. and Sidner, C., "The Structure of Discourse Structure," *CSLI Report No. CSLI-85-39*, 1985.

[Hovy, 1988] Hovy, E. H., *Generating Natural Language Under Pragmatic Constraints*, Lawrence Erlbaum Associates, 1988.

[Kaplan and Bresnan, 1982] Kaplan, R. and Bresnan, J., "Lexical-Functional Grammar: A Formal System for Grammatical Representation," In Bresnan (Ed.), *The Mental Representation of Grammatical Relations*, MIT Press, 1982.

[Kita et. al., 1989] Kita, K., Kwabata, T. and Saito, H., "HMM Continuous Speech Recognition using Predictive LR Parsing," In *Proc. of ICASSP - IEEE International Conference on Acoustic, Speech, and Signal Processing*, 1989.

[Kitano, 1988] Kitano, H., "Multilingual Information Retrieval Mechanism using VLSI," In *Proceedings of RIAO-88*, 1988.

[Kitano, 1989a] Kitano, H., "A Massively Parallel Model of Natural Language Generation for Interpreting Telephony: Almost Concurrent Processing of Parsing and Generation," In *Proceedings of the Second European Conference on Natural Language Generation*, 1989.

[Kitano, 1989b] Kitano, H., "A Model of Simultaneous Interpretation: A Massively Parallel Model of Speech-to-Speech Dialog Translation," In *Proceedings of the Annual Conference of the International Association for Knowledge Engineers*, 1989.

[Kitano et. al., 1989a] Kitano, H., Tomabechi, H. and Levin, L., "Ambiguity Resolution in DMTRANS PLUS," In *Proceedings of the Fourth Conference of the European Chapter of the Association for Computational Linguistics*, 1989.

[Kitano et. al., 1989b] Kitano, H., Tomabechi, H., Mitamura, T. and Iida, H., "A Massively Parallel Model of Speech-to-Speech Dialog Translation: A Step Toward Interpreting Telephony," In *Proceedings of the European Conference on Speech Communication and Technology (EuroSpeech-89)*, 1989.

[Kitano et. al., ms.] Kitano, H., Iida, H., Mitamura, T. and Tomabechi, H., Manuscript, "An Integrated Discourse Understanding Model for Interpreting Telephony under a Direct Memory Access Paradigm," Carnegie Mellon University, 1989.

[Litman and Allen, 1987] Litman, D. and Allen, J., "A Plan Recognition Model for Subdialogues in Conversation," *Cognitive Science 11* (1987), 163-200.

[Morii et. al., 1985] Morii, S., Niyada, K., Fujii, S. and Hoshimi, M., "Large Vocabulary Speaker-Independent Japanese Speech Recognition System," In *Proceedings of ICASSP - IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1985.

[Pollard and Sag, 1987] Pollard, C. and Sag, I., *Information-based Syntax and Semantics*, volume 1, CSLI, 1987.

[Riesbeck and Martin, 1985] Riesbeck, C. and Martin, C., "Direct Memory Access Parsing," *Yale University Report 354*, 1985.

[Saito and Tomita, 1988] Saito, H. and Tomita, M., "Parsing Noisy Sentences," In *Proceedings of COLING-88*, 1988.

[Schank, 1982] Schank, R., *Dynamic Memory: A theory of learning in computers and people*, Cambridge University Press, 1982.

[Tomabechi et. al., 1988] Tomabechi, H., Mitamura, T. and Tomita, M., "Direct Memory Translation for Speech Input: A Massively Parallel Network for Episodic/Thematic and Phonological Memory," In *Proceedings of the International Conference on Fifth Generation Computer Systems*, 1988.

[Tomabechi and Levin, 1989] Tomabechi, H. and Levin, L., "The Head-driven Massively-parallel Constraint Propagation: Head-features and subcategorization as interacting constraints in associative memory," In *Proceedings of CogSci-89*, 1989.

[Viterbi, 1967] Viterbi, A.J., "Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm," In *IEEE Transactions on Information Theory* IT-13(2): 260-269, April, 1967.

[Webber, 1983] Webber, B., "So What Can We Talk About Now?" In *Computational Models of Discourse*, The MIT Press, 1983.

[Young et. al., 1989] Young, S., Ward, W. and Hauptmann, A., "Layering Predictions: Flexible use of Dialog Expectation in Speech Recognition," In *Proceedings of IJCAI-89*, 1989.