

On Removing Ambiguity in Text Understanding

Simin Li, Yukihiro Itoh
Dept. of Computer Science, Shizuoka Univ.

This paper discusses how to remove a kind of ambiguity in a text understanding system based on simulation. The system simulates some events mentioned in text on a world model and observes the behavior of the model during the simulation. Through these processes, it can recognize the other events mentioned implicitly. However, in case the system infers plural number of possible world, it can't decide which one is consistent with the context. We deal with such ambiguities. The ambiguities, in some cases, can be removed by considering contextual information. To remove the ambiguities in the simulation, we define three heuristics based on the characters of the explanatory descriptions and propose an algorithm of "looking ahead". We implement an experimental system. According to the algorithm, the system finds out the supplementary descriptions in the following sentences and removes the ambiguities by using the contents of the supplementary descriptions.

Keyword: *Text Understanding, Simulation, Ambiguity, Heuristics.*

1. Introduction

In this paper, we discuss a method of removing ambiguities which appear in the process of text understanding based on simulation.

We have been studying a text understanding system paying attention to the importance of an imagerial world model. In [Itoh,92, Itoh,95], we showed that the ability to simulate the matters described by sentences on the imagerial world model is one of the basic abilities to understand texts. We also proposed a method of implementing an imagerial world model, a method of simulating on the model and a method of observing the model to extract propositional expressions. We introduced an experimental system in order to verify importance of the imagerial world model and validity of each method.

In order to clarify the point of issue, we have done the work under the following constraints.

1. We restrict our target texts to those explaining mechanical movements of machines in textbooks for junior high school students or encyclopedias for naive persons. The reasons are that we need to treat a relatively narrow domain in order to implement a world model imagerially in a simple way, and imagerial information seems to be quite important in the domain of mechanical movements.
2. We restrict the machines to those which are composed of solid parts and are illustrated with two dimensional figures. It also makes the implementation of the model easy.
3. We deal with only the sentences which explain states or movements of a whole machine or its parts. Though there are some sentences explaining a pressure of gaseous fuel, a flow of fluid, or etc., we neglect them. If we deal with such sentences, we should represent a pressure or a flow of fluid as well as positions and shapes of machine. In other words, the imagerial world model should hold information on multiple attributes. It makes the imagerial model complex.
4. We restrict the sentences to those having no ambiguity. In addition, we also restrict the texts to those which describe movements of a machine one by one along the time axis. The reason is that we want to concentrate on how to process each sentence and simplify how to control the processes for individual sentences.

However, there are many texts which don't satisfy the fourth constraint even if they satisfy the first and second constraints. Especially, if we remove the third constraint, it is necessary to deal with multiple attributes changing

simultaneously such as a location, pressure, and so on. In addition, our world model isn't governed by entire physical law and it should hold some information qualitatively. Therefore, the simulation process comes to have ambiguities.

In general, it is usual that ambiguities appear in the process to reproduce a qualitative and continuous world model by simulation based on discrete descriptions such as sentences. Thus, we relax the third constraint and discuss the method of removing ambiguities. In this paper, we aim to propose an elementary method to construct text understanding systems based on simulation.

In the next section, we survey ambiguities in text understanding and define the problems we deal with. We also show an outline of the text understanding system proposed in [Itoh,92, Itoh,95], and illustrate how our problems are appeared in the process of text understanding. Then, we show some heuristic characters which are applicable to solve the problems. In section 3, we show the detailed algorithm to remove ambiguities, and in section 4 we introduce our experimental system.

2. Fundamental discussion

2.1 Related works

As Johnson-Laird mentioned [Johnson-Laird,83], it can be regarded that language understanding process consists of two steps. The first step is understanding of literal meaning and the second step is understanding of significance. We get propositional expressions of sentences in the first step and we infer or imagine related facts and construct mental-models in the second step. We can find ambiguity in both steps. In the first step, we can interpret a sentence in multiple ways based on homonyms or ambiguities of the grammar. However, we put the ambiguities in the first step out of our consideration and concentrate our attention on the ambiguities in the second step.

The main task in the second step can be regarded as constructing a mental-model by inferring implicit matters from propositional expressions using general knowledge. Generally a text explains only some significant matters directly. It is impossible to explain all of matters which should be represented in the mental-model. We should infer implicit matters from some descriptions of the significant matters. Ambiguity in the second step appears in the process of inferring implicit matters.

Qualitative kinematics is a framework of analyzing the behavior of a machine. The model of target machine consists of some variables representing the status of the parts and constraint on those variables. The system figures out a free space which represents all possible states of the machine on N-dimensional configuration space. (N is the number of the variables). Based on the idea of "kinematic pair" and "kinematic chain", the system can find the free space of the machine containing lower pair and some feasible upper pair. Then it divides the configuration space qualitatively (For example, CLOCK system uses "place vocabulary" to represents the qualitative configuration space. [Forbus *et al.*,87, Forbus *et al.*,91]) and envisions the qualitative behavior of the machine on the qualitative configuration space. In case the system discovers some ambiguities on the behavior through the qualitative envisioning, it uses quantitative methods to remove those ambiguities. Therefore, it can be regarded that the system firstly understands all possible movement of the machine and secondly analyzes the details of each individual behavior.

In contrast, our target text explains an individual behavior of the target machine and it seems reasonable to suppose that naive persons firstly try to understand the behavior. Thus we aim to construct a world model of the mentioned behavior first. For the purpose of constructing the model, our system simulates the mentioned movements one by one according to the text. Ambiguities appear during the simulation.

In the area of text understanding, the ambiguity is removed by considering contextual information. Script based text understanding systems previously prepare some scripts each of which represents a series of typical events. The systems interpret a sentence and fill some slots of a script [Schank and Abelson,77, Skim and Kkim,90]. However, it is impossible to prepare every possible context.

Rule based text understanding systems try to infer all the possible worlds from each sentence by using forward and/or backward reasoning algorithm, and searches a consistent path [Terenziani,93, Nishihara *et al.*,94, Takamatsu *et al.*,95]. While the systems don't have to prepare contexts previously, the problem of combination explosion might occur. It is difficult to manage a huge number of possible worlds.

2.2 The framework of our simulation based text understanding system

In this section, we introduce a text understanding system based on simulation on an imagerial world model [Itoh,92, Itoh,95]. It accepts a text explaining a mechanism of a simple machine. It understands each of sentences and constructs a context model corresponding to a mental-model.

Figure 1 shows the framework of our imagerial world model. To understand mechanisms of machines, it is indispensable to understand the spatial information (shapes, locations of their parts, and changes of them). We use a world model in which the spatial information is represented imagerially by using a two dimensional coordinate system. A movement of a part is represented by a set of micro-changes which correspond to very little changes. For example, a downstroke of a piston represented as integration of micro-changes toward the bottom of a cylinder. We call the micro-change "a trace of change". The traces are represented in terms of predicate expressions. For example, a trace of the movement "descend" is represented as "(dLM (0 1 0) *)", where "dLM" means "a differential linear movement" and "(0 1 0)" represents a downward direction. While the traces are represented in such a symbolic way, they can be compared with each other in imagerially.

Our system processes an input text in the following way. The system has an initial imagerial model of a target machine. Receiving a text, the system analyzes syntactic structure of each sentence. If a sentence is a complex sentence, the system divides it into each simple sentence. Then the system transforms the sentence into a propositional expression. In the expression, nominal concepts like "piston" and "gaseous fuel" are represented by frames which can be connected to objects in the imagerial world model, and verbal concepts like "descend" and "rotate" are represented in terms of predicate expressions. For example, "descend" is represented as "(continue *frame_id* (dLM (0 1 0) *))", here *frame_id* is an identifier of a frame representing a subjective noun.

Then the system starts to simulate the content of the simple sentence on the world model. The system tries to reproduce a trace of the change in the predicate expression of the sentence, and tries to decide an arrangement of the machine when the trace is actually reproduced in the model. If it finds a possible arrangement, it reproduces the world model at the next time step. In this process, the system uses the world knowledge such as the relation between volume and pressure of gases, the rules deciding movable range based on the connection of parts, and so on.

The system has the ability to observe the world model and recognize some facts which aren't mentioned in the text explicitly, and generate predicate expressions representing such facts. For example, in the simulation process of the sentence "the piston descends", it can recognize the movement of the crank, the rod and the other parts, and generate expressions like "the crank rotates". In other words, it can grasp the tendency of the sequence of traces and symbolize it. Moreover, after it grasps the tendencies of every movement of the parts, it can envision a scene where the tendencies of the movements of some parts might change, and reproduce the scene at a bound. We call the scene "a landmark scene". Then it continues to reproduce the trace and simulates the sentence. When it fails to reproduce the trace or finds repetition of the movement, it stops simulation, and starts to process the next simple sentence.

There are some cases that the contents of the following sentences have been already reproduced in the world model by the simulation of the previous sentence. So, the system firstly searches the content of a new sentence in the world model by matching the predicate expression transformed from the sentence with each predicate expression that the system has generated in the observing process. If failing, it regards that the sentence mentions another successive movement, and starts to simulate it.

In [Itoh,92, Itoh,95], we don't have to consider the ambiguities in the above process. However, when we relax our constraints so that we can deal with pressure or flow of a gaseous fuel, we should handle the ambiguities. The ambiguities appear in the following situations:

- When the system infers the traces of the changes of all the parts. If multiple possible traces of a certain part are inferred and they conflict with each other, it is ambiguous that which of the traces is the actual one.
- When the system envisions a landmark scene, it is ambiguous that the movement of the machine continue until the scene or not. In addition, if the system envisions multiple landmark scenes based on multiple tendencies, it is ambiguous that which scene comes first. These ambiguities are emphasized by interminglement of the multiple feature-space in the world model (a two-dimensional spatial model and a pressure space representing the pressures of every gas).

We don't intend to deal with the problem by preparing parallel possible world models. The reason is that our imagerial world model is too large and complex to manage in such a way. Therefore, we have to investigate a light method of handling the ambiguity problems.

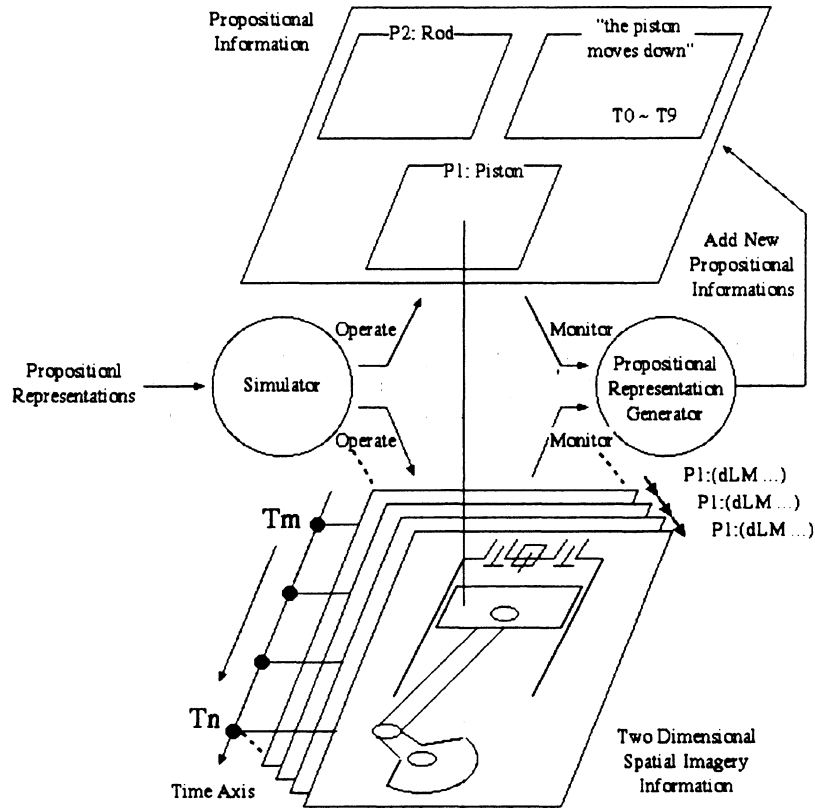


Figure 1: Two-Dimensional spatial imagery world model

2.3 Heuristics to remove ambiguities

Obviously the purpose of our target texts in textbooks is to explain mechanisms of mechanical tools and let readers understand the mechanisms accurately. Therefore, it seems that they meet some standards such as:

- S1: In the case that a text contains an ambiguous sentence from which plural movements or states of a part can be envisioned, the text also contains the other sentences which supplement the ambiguous sentence directly or indirectly if the envisioned movements or states are significant to understand the mechanisms.
- S2: The supplementary explanations should be put on the position not so far from the supplemented description. In almost all cases, the supplementation is described before the movements or the states of the next phase (the movements or states which occur after the movement described in the supplemented description finishes) are mentioned.

Under the assumption that the standards are always satisfied, we picked out following heuristics.

- H1: In the case that plural movements or states are envisioned from a description of a movement, we can handle by either of the following manners.
 - H1-1: Searching the supplementary description, the system can determine which movement or state should be selected. If a description mentions one of the plural movements or states directly, or characteristic movements or states easily derived from them, it is regarded as a supplementary description.

H1-2: The system can let the plural candidates alone and select a movement or state in adhoc way if any supplementary description isn't found. In this case, envisioned movements or states can be regarded to be trivial.

H2: When the system search the supplementary description, it can limits searching area according to this heuristics. Suppose the case that the system finds multiple candidates of a movement or a state of a part, starting from a description of a movement (M) of a part-of-machine. Searching area is limited to sentences which specify the movements or the states consequential upon M. Because, when a sentence which mentions something independent of M appears, it can be regarded that the focus of the explanation shifts to the next phase.

For example, a text describes "The piston descends." in the situation where the suction valve opens. The movement of the piston causes inflow of the gaseous fuel. It becomes ambiguous whether the pressure of the gaseous fuel in the cylinder decreases or not. In this case, if the following sentence specifies "The pressure of the gaseous fuel decreases.", the ambiguity is removed and we can judge that the pressure of the gaseous fuel decreases. However, if no description on the pressure nor the flow of the gaseous fuel appears and the sentence which mentions the independent movement as "The piston reaches the lower dead center and *the suction valve closes*.", we can regard that the pressure does not affect anything and the text does not care to call readers' attention to the pressure.

We do case studies to verify the validity of the heuristics. Table 1 shows the result. We apply the heuristics to 15 texts (totally 281 sentences, for example [Suzuki *et al.*,92]-[Shogakukan,87]) explaining the mechanism of four-stroke engine by ourselves. We find 48 sentences from which plural envisioning are derived, and verified that adequate candidate is successfully chosen by using the heuristics in 41 cases (85%).

In the rest of the cases (7 cases), all the trigger sentences of envisioning describe the upstroke of the piston during the compression stroke. During this stroke, the pressure of the gaseous fuel increases and the downward power piston received also increases. In this case, it is uncertain that whether the piston reaches the upper dead corner or the piston stops during the stroke. If the text doesn't mention that the piston reaches the upper dead corner, the crank continues to rotate toward the same direction, or something, we can't judge which envisioning is adequate by only logic, without our experience. So we think that it is reasonable to fail to select special candidate in the seven cases.

Sentences which lead ambiguities in our simulation	The ambiguities are removed	41
	The ambiguities aren't removed	7
Sentences which don't lead ambiguities in our simulation		233
Total		281

Table 1: Removing ambiguities by using the heuristics

3. A method of removing ambiguity

An algorithm to remove ambiguity is arranged as follows.

1. Detecting the ambiguities.
2. Looking ahead of the text by using the algorithm "looking ahead" and remove the ambiguities if possible.
3. Marking the remaining ambiguities as "let them alone".

3.1 Detection of ambiguities

As we mentioned in section 2.2, two types of ambiguity appear in the simulation. We call the former "ambiguity on change" and the latter "ambiguity on achievement".

The system recognizes an ambiguity on change when it tries to grasp the tendencies of changes. Those tendencies are represented by the predicate expressions. Then, the system detects the ambiguity in the following ways.

1. If multiple predicate expressions hold the same object, the same predicate, and different directions, then it recognizes the ambiguity.
2. If multiple predicate expressions hold the same object and inconsistent predicates, then it also recognizes the ambiguity.

For instance, if the system recognizes two possible traces which are expressed by "(continue gaseous_fuel (dAT pressure + *))" and "(continue gaseous_fuel (dAT pressure - *))" (here, "dAT" represents "a differential change of attribute"). They show opposite changes (increasing and decreasing) are occurring in pressure of the gaseous_fuel. Therefore, the system regards the change of the pressure of the gaseous_fuel as an ambiguous change.

The ambiguity on achievement has two sub-class. One is the ambiguity on whether an envisioned state can be reproduced or not. The other is the ambiguity on the order of achievements of multiple envisioned state. In this paper, we deal with only the former one and omit to discuss the latter one. If the former one is removed and the system can repeat the simulation of the text under each guess at the order, the latter one can be removed.

The ambiguity on achievement is recognized when the system envisions a landmark scene where the tendencies of the movements of some parts might change. In this stage, the system wonders if the movements of the machine actually continue until the scene is reproduced or not. Ambiguities of this type always occur when the system recognizes some tendencies of the movements from the traces of changes and envisions some scenes based on the tendencies.

3.2 Removing ambiguities

Our system removes ambiguities on the basis of the heuristics. The outline of the process of removing ambiguities is as follows:

1. The system looks ahead and transforms the next simple sentence into the predicate expression.
2. It checks whether the ambiguities can be removed or not.
3. If it can, it removes the ambiguities and finishes processing. It returns to the simulation process.
4. If it can't, it checks whether it should look the following sentence.
5. If it should, go to step 1.
6. If it needn't, it decides to let the ambiguities alone and marks them. Then it finishes processing and returns to the simulation process.

We explain the key points of the algorithm.

- (1) How to check whether the ambiguities can be removed or not.

In order to explain the mechanisms of mechanical tools, it is essential to explain movements of the parts and states of the machine at the important scenes. Therefore, the sentences in the texts are classified into two types, sentences mentioning movements of the machine and sentences mentioning the states of the machine. Note that we define the sentences mentioning the states so that the sentences which represent the phenomena of achievement like "the piston reaches the upper dead center" are belong to the type. Currently we don't deal with the other types of sentences but we confirm that they don't contribute to removing ambiguities through our case studies.

Thus, we should consider the following four combinations.

- [a] Removing ambiguities on changes by using a sentence mentioning a movement.
- [b] Removing ambiguities on changes by using a sentence mentioning a state.
- [c] Removing ambiguities on achievement by using a sentence mentioning a movement.
- [d] Removing ambiguities on achievement by using a sentence mentioning a state.

However, the third combination ([c]) doesn't appear in the textbooks, so we concentrate [a], [b], and [d].

[a] Removing ambiguities on changes by using a sentence mentioning a movement

In our simulator, each movement is represented by using traces of changes, and each trace is represented in the form of predicates. If a predicate derived from a sentence matches with one of those changes, the system can remove the ambiguity on the change.

[b] Removing ambiguities on changes by using a sentence mentioning a states

The system reproduces the state and checks whether the direction of each change turns to the state or not. If only one of the changes passes the check, the ambiguity can be removed.

[d] Removing ambiguities on achievement by using a sentence mentioning a state

The system reproduces three states in the world model, the current state, the state mentioned by the sentence, and the state wondered about. If the third state can be placed between the first one and the second one, the system can regard that the wondering state is actual.

In the process [b] and [d], our imagerial world model can be effectively used to check the conditions. For example, the ambiguity on achievement of the piston in the compression stroke is removed by using the sentence "the piston reaches the upper dead center". In this stroke, the system simulates the upstroke of the piston and sets up the landmark scene. In this case, the landmark scene is set up on the basis of the movement of the rod. While the piston goes up, the rod changes the direction of its swing. The system wonders that the piston goes up until the rod changes the direction or the piston stops rising before that. Then the system points the center of gravity of the piston at the current state on the two-dimensional coordinate system (call the point "A"). It also points the center of gravity of the piston at the time when the piston is located at the upper dead center (call the point "B"), and at the time when the piston reaches the position at the landmark scene (call the point "C"). Then it checks whether "C" is on the segment AB geometrically.

(2) How to check whether the system should look the next sentence.

If the text is well-polished and meets the standard S2 mentioned in section 2.3, the ambiguities are removed before the movements or the states of the new phase are mentioned.

Suppose that the simulation is triggered by a sentence (trigger sentence, TS) in the text, and the ambiguities are detected in the simulation process. In this stage, the system has a set of predicate expressions (P) which is generated during the simulation of TS. Here, a predicate expression in P shows the change or state consequential upon the content of TS.

On the basis of H2, to remove the ambiguities, the system looks ahead within the sentences which specify the consequential changes or states. So the system makes matching the predicate expression of the looked-ahead sentence with P. If the system finds it in P, the sentence is regarded as one which specifies a consequence. Then the system looks next sentence. If the matching is failed, the system regards that the sentence mentions independent matter of TS. The system, therefore, finishes looking ahead.

(3) How to integrate the ambiguity removing process with simulation process

For all simple sentences, the system processes them in either of the following ways.

- The system simulates the content of the simple sentence on the world model.
- The system searches the predicate expression transformed from the simple sentence in the world model, and links the sentence with the expression.

The looked-ahead sentences are not simulated in the process of removing ambiguity. Almost all of them are merely made matching. Therefore, there is no overhead of the re-simulation of those looked-ahead sentences.

Through removing an ambiguity, the system matches looked-ahead sentences with the predicate expressions in the world model. After the system returns to the simulation process, it can re-use the result of the matching to save the cost of matching. However, the cost is quite less than that of the simulation process and then it does not progress the efficiency of the whole process. So our present system does not re-use the results of matching in the "looking-ahead" process.

4. The experimental system and examples of its process

4.1 System configuration

Figure 2 shows the system configuration. The system is mainly implemented by GCL. Only the part visualizing figures is implemented by Tcl/Tk. The initial imagerial model is presented previously.

The system accepts an input text, and it analyzes sentences to transform into dependency trees. Then it divides each tree into a set of trees each of them corresponds a simple sentence. The *Understanding Unit* accepts the set of dependency trees and interprets them one by one to simulate or confirm the content of each simple sentence on the world model. The *Propositional Expression Generator* watches the world model and generates predicate expressions. The predicate expressions are used in the confirmation. The *Ambiguity Detecting Unit* keeps watching on the simulation process. When it detects some ambiguities, the system starts to remove them. It looks ahead of the sets of trees and tries to remove the ambiguities. After removing them, it continues to interpret the trees.

4.2 Example of the processing

Figure 4 and 5 show examples of the process of the system. In the experiments, the text shown in figure 3 is input. The left side of figure 4 and 5 shows the outputs of the system and right side of “#” shows comments which we add.

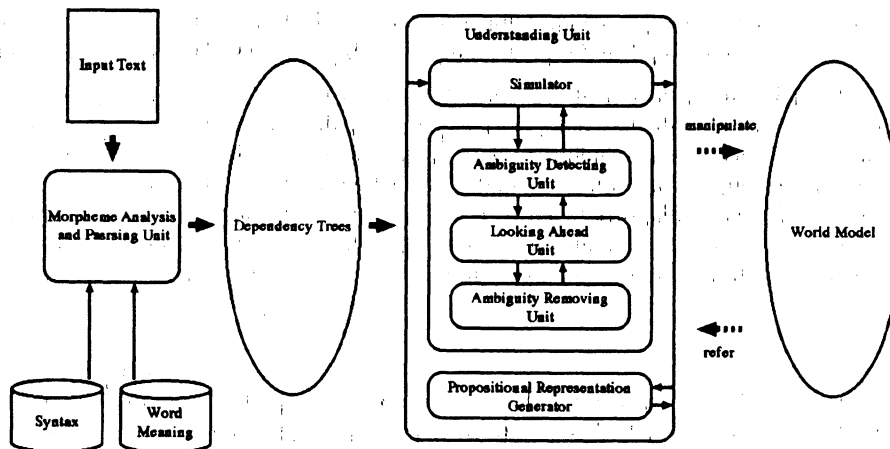


Figure 2: The system configuration

- 吸気弁が開き、クランクが回転してピストンが下降すると、混合気がシリンダ内に吸い込まれる。
(When the suction valve opens and the crank rotates, the piston descends, and the gaseous fuel is inhaled.)
- 吸気弁が閉じてピストンが上昇すると、混合気は強く圧縮される。
(Successively the suction valve closes and the piston rises, then the gaseous fuel is strongly compressed.)
- ピストンが上死点に達し、点火プラグが圧縮した混合気に点火して混合気を爆発させると、その圧力でピストンが動かされてクランクが回転する。
(When the piston reaches the upper dead center, the spark plug ignites the gaseous fuel and the fuel explodes. Then the piston moved by the pressure and the crank rotates.)
- 排気弁が開き、ピストンが上昇して、燃焼ガスを空気に排出する。
(After that, the exhaust valve opens, the piston rises, and combustion gas is exhaust in the air.)

Figure 3: Input text

When the system simulates the sentence “the crank rotates” in the explanation of the suction stroke, the behavior of the pressure of the gaseous fuel becomes ambiguous. Figure 4 shows the process to remove the ambiguity.

In the step (1), the system detects the ambiguities on changes. The following several steps show the ambiguities that system detects. The predicate expressions used in the system and their meaning are shown in the figure. The system detects the ambiguities on changes of the pressure of the gaseous fuel in the cylinder and the intake.

In the step (2), the system starts to look ahead of the following text. In this situation, the system has already simulated and recognized that “the piston descends”, “the gaseous fuel is inhaled into the cylinder” simultaneously with the rotation of the crank. However the content of the sentence “the suction valve closes” is not reproduced in the simulation yet. Therefore, when the system looks the sentence, it stops looking ahead. As a result of the “looking ahead” process, the system concludes that the ambiguities on the changes of the pressure are trivial (step (3)). It lets them alone and continue the simulation.

```

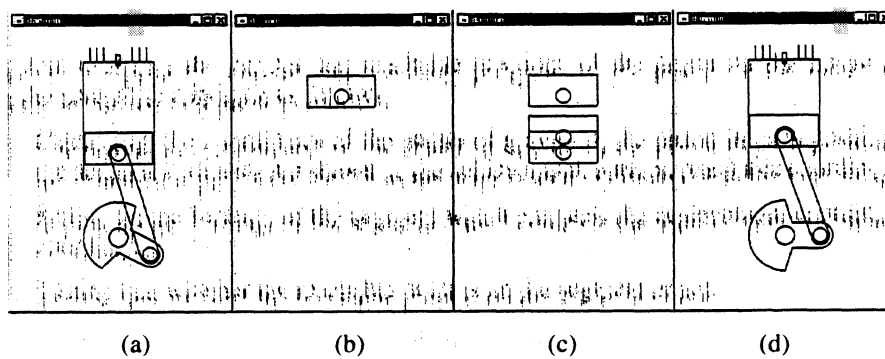
◆◆◆ ambiguity on change ◆◆◆
PROP_510 : ((CONTINUE GAS_4 (DAT PRESSURE - *) NIL) NIL) # (1)
PROP_505 : ((CONTINUE GAS_4 (DAT PRESSURE + *) NIL) NIL) # the pressure of the gaseous fuel
# in the cylinder decreases
# the pressure of the gaseous fuel
# in the cylinder increases
◆◆◆ ambiguity on change ◆◆◆
PROP_509 : ((CONTINUE GAS_3 (DAT PRESSURE + *) NIL) NIL) # the pressure of the gaseous fuel
# in the intake increases
PROP_504 : ((CONTINUE GAS_3 (DAT PRESSURE - *) NIL) NIL) # the pressure of the gaseous fuel
# in the intake decreases
◆◆◆ ambiguities on achievement ◆◆◆
BREAK_POINT_554
reachable state : STATE_553
BREAK_POINT_551
reachable state : STATE_538
★★★★★ look ahead ★★★★★ # (2)
The sentence describes as follows.
(CONTINUE PISTON (DLM (0 1 0) *) NIL) # the piston descends
The content has already been simulated.
The sentence describes as follows.
(CONTINUE GAS_4 (DFLOW * SPACE_3)) # the gaseous fuel is inhaled
# into the cylinder
The content has already been simulated.
The sentence describes as follows.
(CLOSE SUCTION_VALVE) # the suction valve closes
The content is not simulated.
★★★★★ finish ★★★★★
Following ambiguities are not removed.
◆◆◆ ambiguity on change ◆◆◆
PROP_510 : ((CONTINUE GAS_4 (DAT PRESSURE - *) NIL) NIL)
PROP_505 : ((CONTINUE GAS_4 (DAT PRESSURE + *) NIL) NIL)
◆◆◆ ambiguity on change ◆◆◆
PROP_509 : ((CONTINUE GAS_3 (DAT PRESSURE + *) NIL) NIL)
PROP_504 : ((CONTINUE GAS_3 (DAT PRESSURE - *) NIL) NIL)
Let them alone. # (3)
Garbage collecting...done
    
```

Figure 4: An example of removing process of ambiguity on change

When the system simulates the sentence “the piston rises” in the explanation of the compression stroke, the behavior of the piston becomes ambiguous. The system sets the landmark at the point the direction of the swing of rod will just change. However it is ambiguous that the piston rises across the landmark or stops at the halfway. Figure 5 shows the process that the system removes the ambiguities on achievement. In the figure, (a) shows the current state. The output of the system means:

- (1) The system detects the ambiguity on the achievement. Following several steps show the content of the ambiguities on the achievement. “BREAK_POINT_??” are the symbol name of the landmarks.
- (2) The system finds the statement “the piston reaches the upper dead center” during the “looking ahead” process. Here, LOCATION_7 means the upper dead center.
- (3) The system generates a state that the piston is placed in the upper dead center and shows the state imagerial (b).

- (4) The system overprint the current and reachable positions of the piston on the image (c) and then it checks the geometric condition as follows.
 - i. Calculating the coordinates of the center of gravity of the piston in each position. In the figure the actual coordinates are shown as the achievement, current, reachable coordinates.
 - ii. Setting up the formula of the segment which connects the achievement coordinates and current coordinates.
 - iii. Testing that whether the reachable point is on the segment or not.
- (5) The remaining ambiguities on achievement are disregarded.
- (6) As a result, the system fixes the ambiguity and continues the simulation in the context "the piston reaches the upper dead center" (d).



```

◆◆◆ ambiguities on achievement ◆◆◆ # (1)
BREAK_POINT_1789
  reachable state : STATE_1784
BREAK_POINT_1781
  reachable state : STATE_1776
BREAK_POINT_1773
  reachable state : STATE_1768

★★★★★ look ahead ★★★★★
The sentence describes as follows.
(CONTINUE GAS_4 (DAT PRESSURE + *)) # the pressure of the gaseous fuel
                                     # in the cylinder increase
The content has already been simulated.

The sentence describes as follows. # (2)
(REACH PISTON LOCATION_7) # the piston reaches
                           # the upper dead center
The content describes PISTON reaches LOCATION_7.

Show the achievement state. # (3)
Show the current, reachable and achievement state # (4)

achievement coordinates : (150 92,5)
current coordinates : (150 173,02780000000001)
reachable coordinates : (150,0 151,078)

The ambiguity on achievement BREAK_POINT_1773 is removed.

The sentence describes as follows. # the spark plug ignites
(IGNITE GAS_4) # the gaseous fuel
The content is not simulated.

★★★★★ finish ★★★★★
Following ambiguities are not removed.

◆◆◆ ambiguities on achievement ◆◆◆
BREAK_POINT_1781
  reachable state : STATE_1776

Let them alone. # (5)
Continue the simulation. # (6)

TIME : TIME_1984
STATE : STATE_1986
    
```

Figure 5: An example of removing process of ambiguity on achievement

5. Conclusion

In this paper we discussed the problems on ambiguity appearing in the text understanding system based on simulation. We discussed the standard style of explanatory descriptions and defined three heuristic rules to remove the ambiguity. We proposed a method of looking ahead of the text and remove the ambiguities by using the heuristics. Applying the method, the system can remove the ambiguity based on the context through light process.

Though we put such a restriction on the texts as they should be arranged along the time axis, we think it is not difficult to remove it. If sentences of a text are not arranged along the time axis, we can infer the order using aspect and tense by some algorithms (for example [Era *et al.*,93]) and we can confirm consistency of each inferred assumption by checking how smooth we can simulate the text on the basis of the assumption.

In addition, it is important to investigate a method to remove ambiguities in texts by using the texts themselves but also figures given with the texts. As for the texts explaining mechanism of some machines, they are accompanied by figures in the almost all the cases. The integration of information of texts and figures is one of interesting topics in natural language understanding. In order to study the integration, we think it is important to clarify the algorithm using texts alone and its limitation. We proposed an algorithm to remove ambiguities using texts alone. On the basis of the algorithm, we will sophisticate the algorithm and study on new algorithm using both texts and figures.

REFERENCES

- [Itoh, 92] M.Itoh, S.Kubo and Y.Itoh: "Texts Understanding using a Two Dimensional Spatial Imagery World Model", In *Proceedings of PRICAI 92*, pp.589-595, 1992.
- [Itoh, 95] M.Itoh, S.Kubo and Y.Itoh: "Text Understanding Using a Two Dimensional Spatial Imagery", *Transactions of IEICE D-II*, Vol.J78-D-II, No.6, pp.931-945, 1995. (in Japanese).
- [Johnson-Laird, 83] P.N.Johnson-Laird: *Mental Models*, Cambridge University Press, Cambridge, 1983.
- [Forbus *et al.*, 87] K.D.Forbus, P.Nielsen, and B.Faltings: "Qualitative kinematics: A framework", In *Proceedings of IJCAI87*, pp.430-435, 1987.
- [Forbus *et al.*, 91] K.D.Forbus, P.Nielsen, and B.Faltings: Qualitative spatial reasoning: the CLOCK project, *Artificial Intelligence*, 51(1-3):417-472, 1991.
- [Schank and Abelson, 77] R.C.Schank and R.P.Abelson: *Scripts, Plans, Goals, and Understanding*, Lawrence Erlbaum, Hillsdale, 1977.
- [SKim and KKim, 90] S.Kim and J.Kim: "Situation Assessment through Script Matching", In *Proceedings of PRICAI 90*, pp.310-315 1990.
- [Terenziani, 93] P.Terenziani: "Integrating Linguistic and Pragmatic Temporal Information in Natural Language Understanding: the case of "When Sentences", In *Proceedings of IJCAI 93*, pp.1304-1309, 1993.
- [Nishihara *et al.*, 94] N.Nishihara, K.Morita and F.Saito: "A Reasoning Method for Natural Language Based on a Term Logic System - An Application to a Question-Answering System by Japanese Sentences -", *Journal of JSAI*, Vol.9, No.4, pp.537-547, 1994. (in Japanese).
- [Takamatsu *et al.*, 95] S.Takamatsu, N.Izumi, K.Kise and K.Fukunaga: "First-Order Logic Based Situational and Dynamic Interpretation of Natural Language Descriptions in Hardware Design Specifications", *Journal of JSAI*, Vol.10, No.5, pp.720-730, 1995. (in Japanese).
- [Suzuki *et al.*, 80] H.Suzuki, *et al*: *Gijutsu, Katei (Techniques and Households)*, Vol.2, pp.9-42, Kairyudo Co., 1980. (in Japanese).
- [Suzuki *et al.*, 92] H.Suzuki, *et al*: *Gijutsu, Katei (Techniques and Households)*, Vol.2, pp.86-119, Kairyudo Co., 1992. (in Japanese).
- [Ishida *et al.*, 92] H.Ishida, *et al*: *Atarashii Gijutsu, Katei (A New Edition of Techniques and Households)*, Vol.2, pp.79-112, Tokyo Shoseki Co., 1992. (in Japanese).

[Hiroyasu, 73] H.Hiroyasu: *Wakaru Nainenkikan (Plain Explanation of Internal Combustion Engine)*, pp.2-3, Nissin Publ. Co., 1973. (in Japanese).

[Hatakeyama, 78] S.Hatakeyama: *Jidousha Kougaku (Automotive Engineering)*, pp.3-4, Rikogaku Publ. Co., 1978. (in Japanese).

[Shogakukan, 80] Shogakukan: *Banyuu hyakka daijiten (Universal Encyclopedia)*, p.328, Shogakukan Co., 1980. (in Japanese).

[Iwanami Publ, 85] Iwanami Publ: *Kagaku no Jiten (Encyclopedia of Science)*, pp.1173-1174, Iwanami Publ. Co., 1985. (in Japanese).

[Shogakukan, 87] Shogakukan: *Nihon Daihyakka Zensho (Japanese' Encyclopedia)*, p.86, Shogakukan Co., 1987. (in Japanese).

[Era *et al.*, 93] K.Era, K.Kai, J.Nakamura, S.Yoshida: "Analysis of Temporal Relation in Stories Using Tense and Aspect Expressions", *Technical Report of IEICE*, NLC93-6, 1993. (in Japanese).