Memoirs of the Faculty of Engineering, Nagoya University
Vol. 42, No.2 (1990)

FUNDAMENTAL STUDY OF THE PROBLEM SOLVER FOR FLUIDS ENGINEERING

Ikuo NAKAMURA and Takashi WATANABE*

Department of Mechanical Engineering

(Received October 31, 1990)

Abstract

First of all, the issues of artificial intelligence present dualism, tool, mental act, pure and practical reason, symbols and the realizability of an artificial intelligence system. The main points of Chap. I are importance of rapid, complex and very long reasoning of AI as a tool, the difficulty of fluids engineering problems which require very profound knowledge, and igniting the mental act. In Chap. II, problems are considered, which appear when the domain of discourse is defined equations, and it is shown that in order to find the correspondence between such a world and the real world, we need practical reason. Also the meaning of symbols is discussed as a process of the compression of the world into symbol and reconstruction of the world from the symbol. Finally, the significance of idea related to AI is considered. Following chapters treat real problems.

In Chap. III an efficient method is presented for computational fluid dynamics, which produces a finite difference code automatically for general non-linear parabolic equations. Dimensional analysis has shown very important qualitative results for complex problems in many scientific and engineering branches. An automatic and very general dimensional analyzing system is described in Chap. IV and its high performance is shown by ways of examples. Discussion in Chap. V centers on an application of knowledge engineering to problems in fluid mechanics. The study is also intended to reveal information about the structure of the solving process of human beings in this field. The analyzing system simulates the student's way of processing. Concept data, equation data and inference data are assumed. Examples are given of the problems to determine material characteristic values using equations. Chapter VI considers the fundamental aspects of an application of artificial intelligence to a solving mechanism for elementary problems in fluids engineering. Problems in fluids engineering are classified into two categories, one with unchanging circumstances and the other with changing ones. A multiple layer production system is implemented and it is shown that such types of problems can be well represented.

* College of General Education, Nagoya University.

Contents

I. General Introduction

1. 1. Dualism

The various meanings of the object of AI (artificial intelligence) have been forcus for many philosophical disputes[1-1),1-2)]. In contrast to conventional technique aimed to work as an agent of a human being, which extends and multiples the external action of a man, AI technique is studied to make a machine which performs the work of agent of man and magnifies the intelligent ability of the human being. As a scientific study, the main problem of AI is to understand the mind of the human being by using mainly a computer and to simulate its action by a computer. Thus, AI is concerned with the mind of human being, and this is why so many philosophers have joined the discussion about AI's significance. The serious issue of whether AI is able to add something to the intelligence of man will be addressed in sec. 2.4.

According to Plato's consideration, philosophy is classified into natural philosophy concerning with nature and related topics, ethics which studies human nature and related problems, and logic which supplies an exact method to the other two branches[1-3]. Plato's analysis about technique is also of interest, for, according to his study, for example, a flute technique has three stages: the first is the technique of selecting and obtaining the right material for the flute, the second is to produce a flute from the material and the third is to play the flute. If we apply a consideration like Plato's to AI, the problem is whether the object of AI is nature or the human being. If the object of AI is nature, one of the AI methods to gain insight into one's own thought divides the self into the one seeing and the one seen. The latter is nature in this case, and the problem is the validity of such a procedure as an objective method. If the object of AI is the human being and pertains to ethics, which must treat the concept like free will, then the objectivity of AI as a science becomes ambiguous.

Following Plato's classification of the flute technique, we are aware of the relationship between the three stages. Flute playing technique has a close relationship with the structure of the flute, and the structure is related to the material used and to the technique for acquiring the material. Thus the question arises: Although contemporary discussion of AI neglects the problem of the structure of the human brain or computer hardware, is it an appropriate method of the problem as in the case of flute? This question is related to the correctness of dualism. The novelist Thomas Mann treated this problem very impressively in his novel, the Replaced Heads[1-4]. In the novel at first a man A is represented as a combination as in:

$$A = (A_H \mid A_B),$$

$$A_H\text{: The head of } A,$$

$$A_B\text{: The body of } A.$$

There is another man $B, B = (B_H \mid B_B)$, and their heads have been replaced in a very complicated fashion to become

$$A' = (A_H \mid B_B),$$

$$B' = (B_H \mid A_B),$$

and at first it was assumed as

$$A' = A \quad \text{and} \quad B' = B.$$

Thus, it is considered that the essential element constituting a man is the head. But after some time passed, A_H in $(A_H \mid B_B)$ was affected by B_B and A_H after replacement shows a resemblance not only in appearance but also in thinking with former B_H, and B_B in A' changes so as to resemble the former A_B under the influence of the A_H. As with A', B_H in B' approached the former A_H both in appearance and in thinking affected by replaced A_B in B' and A_B in B' shows features of former B_B invaded by power of B_H.

According to this novel, the head (brain) and the body of a man cannot be considered separately. And if this is a correct idea, we should include this fact in the AI-simulation of human intelligence.

The problem in cognition, the relationship among the object, the subject and the recognized idea is important for AI in terms of the question about the correspondence between the seen object and visualizing machine and judgment and thought obtained from the output of the machine. For example, this situation appears symbolically using a microscope by increasing the magnification of the instrument, the thinking about the microuniverse changes, and such an idea forms a strong motivation to improve the instrument. Also, in case of a computerized visualization, we should ask the meaning of a color produced by computer. In spite of the above issues, in this report we stand the position of dualism which means AI technique to be consisted of two senses, hardware and software. The reader mainly interested in the philosophical dimension may skip to chapter II.

1. 2. Tools

Papers on AI are dramatically increasing and reports of expert systems for diagnosis in various fields are especially numerous[1-5]. The present study tends to be outside this mainstream, considering more fundamental problems of applicability of AI with respect to author's special field of fluids engineering. From this point of view, it is convenient to classify AI systems into three: those which assist with the problem solving of fluids engineering as a tool, those which are used to analyze the structure of fluids engineering problems and the characteristic reasoning appearing in this field and those which help to describe pictorially or in words a flow field. The last system has fundamental importance, but only a preliminary discussion will be given.

CFD (Computational Fluid Dynamics) is the most valuable tool, but in this study we will discuss only its role in the problem solving to be considered in the next section. The other important tool in fluids engineering is the symbolic computation system. Since the mathermatical equations appearing in fluids engineering have much of variety, development of the system accessible to the user gives powerful tool to the research workers. For example, equations of fluids engineering concern incompressible fluids and compressible fluids which must include thermodynamics and shock wave, Newtonian and Non-Newtonian fluids which are formulated by very complicated constitutive equations, laminar and turbulent flow which is represented by using intrinsic probabilistic nature of flow, and MHD flow, magnetic fluids, reacting fluids, nematic fluids, multiphase fluids, flow in porous media, relativistic flow and son on.

Takeuti, a logician at the University of Illinois, pointed out that the present logic is very suitable to the problems which have small number of cases amenable to check each variation or have an infinite or a very large number of cases amenable to be took a limit. He suggested that when we can treat the medium cases by computer, which have too many cases intractable to the human being directly, there is a possibility to change our thinking method[1-6]. We consider that symbolic computation of fluids engineering may be one example in line with Takeuti's suggestion. Also, a remark proposed by Terada, an outstanding experimental physicist, is intimately related to Takeuti's inference. Terada said that if a problem has complete complexity, it may be treated as a statistical problem. And if a problem involves a small number of components and a small number of conditions we can treat it as a deterministic system. The most troublesome case has medium complexity[1-7]. Although some reservations may be necessary with regard to his remark after the chaos of a nonlinear problem is unveiled, it is still basically on the mark.

A brief review will be given about the research of fluids engineering in this direction. Van Dyke has pursued a computer extended series solution of various fluid mechanical equations[1-8]. Perry et al. uses computer to generate automatically Taylor series of Navier-Stokes equations (NS equations) in order to study a singular point of separated flow[1-9]. Constitutive equations of non-isotropic continuum mechanics are being studied using REDUCE by Xu et al.[1-10] MACSYMA is used to apply the theory of Lie-group to NS equations[1-11].

A very interesting example related to Takeuti's suggestion is the Fractal Fluid Mechanics developed by Schlechtendahl using REDUCE[1-12]. He asserts that the theory of turbulence and multiphase fluid should be able to treat the concept which allows a difference between the local physical value at a point and the average value over the region including that point, and then derives new momentum equations. In practise, the combination of CFD solver and AI technique is important and some progress has been made in the field of fan design[1-13].

A database may be used as a tool for fluids engineering, and the most important one is the database on material property of various fluids. The a database including thermal and chemical properties and supported by an expert system is useful for engineers. For example, an expert system is valuable which has a database including a very complex Morier chart of wet air and to assist in the design of air-conditioning systems. The other cases, difficult problems of the correctness of the data, changes in the values by new experiment and new materials added, arise in these databases[1-14].

Experimental techniques in fluids engineering have increased data as exponential of time, and deducing new knowledge from them involves tremendous work. This is why an AI system to produce databases from such experiments is necessary in the near future. This may also apply to the calculted results of CFD.

1. 3. Issues in automatic problem solving for fluids engineering

Some characteristic points of fluids engineering problem solver, special problem analysis, invention of solving method and necessary knowledge will be discussed in the following. Fluids engineering is a branch of engineering based on fluid mechanics as a branch of physics. As another engineering branch, it is used to govern and use the other things by fluids. Since the problem of fluid mechanics appears in more simple form than in fluids engineering, we consider the problem solving in fluid mechanics. In every problem we should restrict our attention to a domain of discourse and we restrict it as a solution set of NS equations (Navier-Stokes eqs.). As mathematical physics, fluid mechanics has axiomatic aspects and follows Truesdell's dogma that the essence of problem solving in physics is not a physical discussion about the problem but to solve the equation under suitable conditions.

The process is formulated as follows. The fundamental NS equation for incompressible Newtonian fluids is

$$\text{Equation of continuity} \quad \sum_{i=1}^{3} \frac{\partial u_i}{\partial x_i} = 0, \tag{1.1a}$$

$$\text{NS equation} \quad \frac{\partial u_i}{\partial t} + \sum_{j=1}^{3} u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu \sum_{j=1}^{3} \frac{\partial^2 u_i}{\partial x_j^2} + f_i, \quad (i = 1, 2, 3). \tag{1.1b}$$

Let the abbreviatd form be,

$$NS.\{\mathbf{u};\ \nu\} = 0, \tag{1.2}$$

where x_i denotes cartesian coordinate, u_i: velocity component, p: pressure, ρ: density of fluid, ν: kinematic viscosity of fluid and f_i: component of external force. Since density is not important, only ν appears as a parameter in Eq. (1.2).

Eq. (1.2) may be interpreted as an operation $NS.$ which selects a function set $\{\mathbf{u}\}_{NS}$ from suitable general function space. The concrete form of the solution function may be selected from $\{\mathbf{u}\}_{NS}$ under the constraints of the initial condition (A) and the boundary condition (B). We express this process as follows:

$$(A,\ B).\{\mathbf{u}\}_{NS} = \mathbf{u}_{solution} = \mathbf{u}_S. \tag{1.3}$$

In practice, using CFD solver (CFD), we approximate $\mathbf{u}_S$ as a subset: $\{\mathbf{u}_S\}_N \subset R^4$ where R denotes real number and R^4 indicates $\{x_1,\ x_2,\ x_3,\ t\}$ (exactly R in CFD is rational number). Let this procedure be

$$(CFD).[((A),\ (B)).\{\mathbf{u}\}_{NS}] = \{\mathbf{u}_S\}_N, \tag{1.4}$$

or write A, B as conditions in the following expression,

$$(CFD).[NS\{\mathbf{u},\ \nu\} = 0 \mid (A),\ (B)] = \{\mathbf{u}_S\}_N. \tag{1.5}$$

This equation means to derive a system of simultaneous equations by use of suitable discretizing method from $NS\{\mathbf{u},\ \nu\} = 0$ under the conditions of (A) and (B), then to solve the system numerically. This can be done in a simple flow problem. Since the output, $\{\mathbf{u}_S\}_N$, of a usual CFD solver involves too many numbers to be directly understandable by a human being, $\{\mathbf{u}_S\}_N$ should be compressed to a set $\{\mathbf{u}_S\}_{NC}$ which expresses meaningful numbers, graphs or animations.

From the viewpoint of AI, it is of interest how to construct an operator defined as

$$(CV).\{\mathbf{u}_S\}_N = \{\mathbf{u}_S\}_{NC} \tag{1.6}$$

where (CV). means a procedure to compress or to compact and visualize a set $\{\mathbf{u}_S\}_N$. Here is the contact point between a set $\{\mathbf{u}_S\}_N$ and a human being. The operator, (CV). should process $\{\mathbf{u}_S\}_{NC}$ to have fluid mechanical significances and understandable intuitively by experts and non-experts. In particular, the role of (CV). is important when $\mathbf{u}$ shows chaos or turbulence.

Aside from these problems, the present discussion concerns the condition (B). Two types of (B) are classified as the one related to infinite distance and the other connected with finite distance. The concrete form of (B) can be known only after the concrete flow field is defined. However, the flow field appears with infinite variety and it is impossible to express (B) as a function to vary its parameter to fit each case. For example, a sand-roughened surface has intrinsic stochastic character and can be described only like a non-anticipating function. For this reason, (B) can not be implemented in the computer beforehand.

If we assume that some general equation determining (B) is obtained, of course such an equation is not a usual equation but is constituted from usual equations, sentences and graphs. Then the method for solving such a general equation is an important application of AI. A more simple but still not solved problem for current CFD solver is how to determine the calculating region around a finite body. Since current CFD solvers cannot treat an infinite region, they are restricted to a finite region and the no exact condition far downstream region is not known. We should make some intuitive decision on the scale of the calculating region.

The next problem is the initial condition (A), which is related to (B), since (A) should be given for the entire flow region because Eq. (1.1b) is an elliptic partial differential equation. To determine (A) for turbulent flow is difficult.

Historically, many renowned mathematicians and theoretical physicists tried to solve fluid mechanical problems following Truesdell's dogma, but no one has been able to solve them in general so as to be applicable to practical fluids engineering. Fluids engineer developed clever solving methods which constitute with abstraction of the problem in some sense or so called approximate methods and experimental methods. In these methods, from the mathematical standpoint, some integration is performed for Eq. (1.1) and to reduce variables to obtain solvable algebraic equations. A fluids engineer does not usually integrate Eq. (1.1) in order to derive necessary relations, but he takes a suitable control surface, assumes a reasonable distribution of various quantities on it from his experience, and applies a physical law more fundamental than Eq. (1.1).

For example, let a pipe or general duct surface is defined by a suitable co-ordinate, while in the practical situation this is almost impossible for the real pipe. After integrating Eq. (1.1a) in the domain bounded by the control surface S which is composed of the sections A_1 and A_2 and the pipe wall as shown in Fig. 1.1, we obtain the mass conservation law by use of the divergence theorem of the following equation,

$$\iiint_V \sum_{i=1}^{3} \frac{\partial u_i}{\partial x_i} dV = \iint_S u_i n_i da = 0,$$

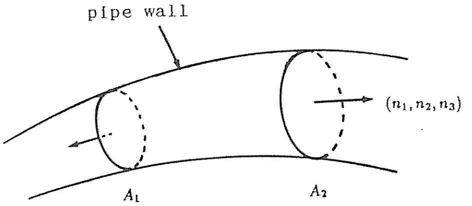


Fig. 1. 1. Control surface S.

where n_i denotes a component of a unit normal vector. Since fluid cannot penetrate the pipe wall, the above equation becomes as follows,

$$-\iint_{A_1} u_i n_i da = \iint_{A_2} u_i n_i da = Q, \tag{1.7}$$

where Q denotes discharge of the pipe. Area averaged velocity v is defined as $v = Q/A$ and since Q is constant, we obtain the following,

$$vA = Q = \text{const.} \tag{1.8}$$

In fluids engineering, without considering Eq. (1.1a) and from the first step, we assume a uniform velocity in a section and apply mass conservation law directly to the control surface to obtain Eq. (1.8).

Let x be the axial distance in Eq. (1.8), then $A = A(x)$ and Eq. (1.8) can be written as,

$$v = v(x) = Q/A(x). \tag{1.9}$$

In the real flow field velocity changes in the three-dimensional space, but in this treatment the velocity varies only with x and this method is called as one-dimensional approximation. However, from another point of view, this method may be identified as a one-dimensional abstract method since it applies to any shape section.

Another important equation for pipe flow is Bernoulli's equation. Let the viscosity of fluid be negligible for the moment and the flow be steady, then the resulting equation of Eq. (1.1b) is integrated along a streamline to obtain Bernoulli's equation. In fluids engineering, we apply the deep physical law of energy conservation directly to the control surface in Fig. 1.1 and derive the following Bernoulli's equation as,

$$\frac{1}{2}\rho v^2 + p + gz = \text{const}, \tag{1.10}$$

where g is the gravitational acceleration and z is the height of the duct center from the some level defined and $z = z(x)$. As with Eq. (1.9), no particular condition is imposed on the duct shape in Eq. (1.10). Also, the difference between the laminar flow and turbulent flow is neglected to obtain Eq. (1.10). Extension of Eq. (1.10) to the case having energy loss due to energy dissipation in the flow is done by introducing loss coefficient λ inferred from the dimensional analysis. The difference between laminar and turbulent state is presented only through the value λ, which is calculated theoretically for the laminar flow in straight pipe or simple duct and can be known only from the experiment for the turbulent flow. Even though the flow is laminar, it is difficult to calculate λ by present CFD solver for the duct of a complicated geometry.

The introduction of λ is very clever. If fluids engineers have adhered to solve Eqs. (1.1a) and (1.1b) directly in order to obtain a pressure drop in a pipe, they could not treat fluid flow reasonably, and water turbine, boiler, steam turbine etc. would not develop. Therefore, we cannot use electric power, transistor or computer, and Eqs. (1.1a) and (1.1b) will never be solved. However, fluids engineers were wise enough to find out that turbulent flow can be treated by the spirit of Kant's practical reason, that is the experimental method in this case, even though it cannot be completely analyzed by the pure reason of Kant. They performed many measurements and found the law of the pressure drop along a pipe line, which is still not calculated by any present CFD solver.

From the above discussion it is apparent that schemes like Eqs. (1.5) and (1.6) cannot be of practical use, even though the fundamental equations are obtained and a good partial differential equation solver is developed. The method to apply them to real engineering problems remains to be found. A true high quality problem solver should have the ability to derive equations like Eqs. (1.8) and (1.10) to obtain useful solutions to fit the situation encountered when it cannot solve exactly formulated problems like Eqs. (1.1a) and (1.1b). A very sophisticated intelligence is necessary to solve the problem with suitable modification and simplification when it is found too difficult to treat by exact formulation.

1. 4. Qualitative reasoning and the mental act

In this section, we consider the reasoning which is regarded as qualitative in fluids engineering. The sentence

> "Incompressible fluid flow through a pipe shows increaed velocity and decreased pressure at a reduced section"

is considered usually as an example of qualitative reasoning. This inference is deduced from the equations

$$Av = Q = \text{const.},\quad \frac{1}{2}\rho v^2 + p = \text{const.} \tag{1.11a,b}$$

In Eq. (1.11b), the gravitational term is omitted for the sake of convenience. The most important point of the reasoning process is to start or to trigger one's mental act to conjecture about what will happen if the duct area A is reduced. In this case, there are three parameters to be varied and the three mental actions which should be triggered are:

1. The mental act to change variables in the world on which the real world is projected.
2. Selection of the variable as the mental act.
3. The mental act dealing with how to change the selected variable.

In this regard, reason is defined as follows:

> The reason is to commit the mental act which is allowed and has no contradiction in the projected world.

From this point of view in quantitative reasoning, the reasoning mental act defined above produces some sets of numbers and the qualitative reasoning produces some thing other than numbers. However, to grant the objectivity to the qualitative reasoning, we should relay on the characteristics of numbers, since the most rigorous objective reasoning uses numbers, as has been pointed out[1-15]. So the qualitative reasoning is the mental act which produces the set of numbers different from the traditional structure. The examples are quantization and qualitative integral[1-16].

At this point another problem appears as to whether or not the reasoning act by a man using a pencil and paper is a mental act. We consider this is not a mental act because we can not distinguish the reasoning act using a pencil and paper from the reasoning act using a symbolic computation system, and from the reasoning act using an experimental apparatus. Of course, common man is not equal to Feynman and a common man cannot perform a long chain of calculations as his mental act. The mental act defined above is an idealized interior act of a human being.

References of chap. I

1-1) Torrance, S., ed., "The mind and the machine," (1984), Ellis Horwood, trans. Yoichiro, M., Sangyo Tosho.

1-2) McCarthy, J. and Hayes, P.J., "Some philosophical problems of artificial intelligence," trans. Miura, K., (1990), Tetsugakushobo.

1-3) Diogenis, L., "Lives of eminent philosophers," trans. Kaki, A., (1989), Iwanami.

1-4) Thomas Mann, "The replaced heads," trans. Takahashi, Y., (1951), Shinchosha.

1-5) Special issue of expert systems, J. Japan Soc. AI., (in Japanese), Vol. 5 No. 2 (1990-3).

1-6) Takeuti, G., "The mathematical world view," (in Japanese), (1982), Kinokuniya.

1-7) Terada, T., "An introduction to physics," (in Japanese), (1948), Iwanami.

1-8) Van Dyke, Computer-extended series, Ann. Rev. Fluid Mech., Vol. 16 (1984), 287.

1-9) Perry, A.E., and Chong, M.S., J. Fluid Mech., Vol. 13 (1986), 207.

1-10) Xu, Y.H., et al., Int. J. Engng. Sci., Vol. 25 No. 6 (1987), 771.

1-11) Ames, W.F. and Nucci, M.C., J. Eng. Math., Vol. 20 (1985) 182, and Vol. 21 (1987), 261.

1-12) Schlechtendahl, E.G., A theory of fractal fluids, Kernforschungzentrum Karlsruhe, KfK4070 (1986).

1-13) Andrews, A.E., AIAA J.,Vol. 26 No. 1 (1988), 40.

1-14) Delgrade, J.P. and Mylopoulos, Knowledge representation: features of knowledge in fundamentals of artificial intelligence, ed. Bibel, W. and Jorrand, Ph., (1985), 3, Springer.

1-15) Whitehead, A.N., "Science and modern world," trans. Ueda, Y. and Murakami, Y., (1954), 41, Sogensha.

1-16) Chiu, C.B., J. Japan Soc. AI, Vol. 4 No. 5 (1989), 528.

II. Fundamental Discussions

2. 1. Knowledge and fluids engineering world defined by equations

An important discussion about knowledge was already given by Plato long ago in one of his books of dialogues, Teaithetos. Refutation is made in the dialogue about the usual concept of knowledge, that is, knowledge about playing a flute, knowledge about cloth etc. All of these forms of knowledges are knowledge about something and a debate is given about pure knowledge, not knowledge of something. Discussion on the meaning of knowledge by philosophers has a long history. In these many ideas, Fichte's one is interesting from the stand point of AI. According to him, knowledge is the knowledge inevitable about something, and when knowledge represents deep knowledge, it necessarily has the character of an image[2-1]. This thought corresponds to the assertion by Delgrade and Mylopoulos[1-14] that a knowledge base must represent something. They define knowledge in AI as justified true belief and represent a knowledge base KB as follows,

$$KB = < KB_0, \vdash_L >. \tag{2.1}$$

Where

KB_0: a set of statements expressed by a language governed by a logic L,

$\vdash_L$: the derivability relation in L, ie., specifies what can be derived from the axioms, given the rules of inference in L.

Then knowledge α belongs to KB when α satisfies the condition as follows,

$$\alpha \in KB \quad \text{iff} \quad KB_0 \vdash_L \alpha. \tag{2.2}$$

Thus, the example knowledge base contains not just the statements in KB_0 but others that can be derived from them in L.

Now we discuss some problems of this definition from the viewpoint of fluids engineering. Let us assume KB_0 expresses fundamental equations of fluid mechanics including the usual laws of mechanics. For example, ideal fluids can be described by Eq. (1.1a), and following Euler's equation without external force and some suitable initial and boundary conditions (A) and (B), respectively, are obtained:

$$Eu\colon \quad \frac{\partial u_i}{\partial t} + \sum_{j=1}^{3} u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i}. \tag{2.3}$$

Then $KB_0 = (Eu,(A),(B)) \equiv KB_{OE}$ also includes laws of mechanics. L may be usual mathematics.

This fluid mechanical system has various invariants; for example, Kelvin's circulation conservation theorem:

$$Ke\colon \quad \oint_C \mathbf{u} \cdot d\mathbf{x} = \text{const.}, \tag{2.4}$$

where C means a curve moving with fluids. This theorem gives very important knowledge about the character of a vortex, which is a generic concept to describe fluid motion. Since the proposition Ke is derivable from KB_{OE} defined above, then:

$$Ke \in KB = < (Eu,(A),(B)), \vdash_L > . \tag{2.5}$$

We ask how to construct a program which can answer a question whether KB has Ke or not. How can a program be written to trigger $\vdash_L$ to derive sequentially Ke which is not included in KB_{OE} explicitly? Even an excellent student cannot derive Ke from Eq. (2.3) without a teacher's hints. Only a great theoretical physicist like Lord Kelvin can ignite the inference process to derive Ke from KB_{OE}. So we should ask what kind of a mental action Lord Kelvin has.

The set of flows represented by Eq. (2.3) includes potential flow, in which the velocity u_i can be obtained from a scalar potential ϕ as $u_i = \partial\phi/\partial x_i$. While it is also difficult to obtain this proposition from KB_{OE}, we assume it is possible. Further inference compatible with L shows that any body in such a flow receives no reaction force from the flow. This undoubtedly contradicts our experience, and this fact is called d'Alembert's paradox. Within the knowledge derivable from $KB_{OE} = (Eu,(A),(B))$, a body in a potential flow receiving no force is not a paradox but a compatible knowledge. The above fact indicates the thorny problem of the correctness of a knowledge base.

If a knowledge base $<KB_{OE}, \vdash_L>$ can be programmed and implemented on an AI computer, in the first place, a man having no knowledge about vortex can not ask the AI computer about Kelvin's theorem. It is not an ens for him in the meaning of Parmenides. The AI computer implementing these KB should teach a man like his teacher to do in order to manifest the AI computer's knowledge about Kelvin's theorem. Since the conventional definition of knowledge base is founded on very simple KB and L, another consideration should be given.

2. 2. *Pure reason and practical reason*

Following Kant, the aesthesia (i.e., sensors of a robot) is an ability to receive an idea (a subset of inputs representing some part of categories of the object) by incentive (physical inputs received by the sensors) from an object. Explanations in parentheses are the authors' interpretation according to the present context. However, the mode of receiving the idea, that is the sense datum, is not the aesthesia but the immediate mode to be possessed a priori by the immediacy. This thought means that the cognition of the object consists of the priori mode of the cognition. This signifies that the cognition does not depend on the object but the object does depend on the cognition, and this is a Copernician conversion according to Kant[2-2]. From this idea Einstein's advocacy follows immediately that because we have a theory we can recognize the object to be measured. Also Kant recognized the intelligence which has an ability of inference.

Although Kant's concerns are entirely philosophical, they may apply to the present context, and the knowledge base discussed in the previous section will be analyzed following Kant. If we restrict KB_0 in $< KB_0, \vdash_L > = KB$ to Euler's equation, that is, let $KB_0 = KB_{OE}$, then the whole knowledge obtainable from the KB is the one about perfect fluid. If we consider the object, the world is the whole knowledge as mentioned above, the ens permited to exist in the world is the only one compatible with KB. As pointed out by Einstein, shear stress and viscosity are not objects to be measured in the world of Euler's equation. A not insignificant example is that if we assume random initial condition there may arise turbulent flow in the world of KB produced by Euler's equation, but the decay of turbulence, the main problem of turbulent flow, is the non-ens in this world.

If we extend the knowledge base or knowledge system to the one which can treat viscous fluids, we must include NS equations Eqs. (1.1a) and (1.1b) to KB_0. In this case many difficult problems arise and there is no guarantee that the logic L in KB for viscous flow is as same as the one for Euler equation.

Now we assume that we can extend the KB to include CFD solver, Eq. (1.4), and further we can construct some kind of analytical inference system denoted as (AE). and (CV). of Eq. (1.6) to be implemented by an AI computer. This super fluids engineering problem solver can be written formally as

$$(CV).\left\{\begin{array}{l}(AE).\\ \qquad [NS.\{\mathbf{u},\nu\} = 0 \mid (A),(B)]\\ (CFD).\end{array}\right\} \tag{2.6}$$

There arises a question who asks this system a problem and what a problem it is.

This super solver can calculate the transition process from laminar flow to turbulent flow. But a man having no knowledge about the transition phenomenon cannot ask about it because the problem is the non-ens in the meaning of Parmenides. In this case he cannot ask the problem consciously but only accidentally, without knowing its meaning. However, in such case he will perhaps be able to understand the answer from the super solver only if the super solver perceives his ignorance about the problem and explains to him the significance of the problem with igniting his will to know it.

From the history of fluid mechanics we become conscious that almost all important phenomena have been unveiled by experiments, and theories were devised to explain them after discovery. For a leap from the world of the Euler equation to the one of the NS equations, we need a practical experiment in the real physical world or a mental act of thinking experiment. In both cases the main problem is how to trigger to act such an action in the real world or in the thinking world. Mathematical development of ideal fluid theory based on the a priori pure reason mode of the Euler equation has been done completely but it may lead to the serious mistake by which one considers that only the phenomena described by Euler equation can occur in nature. Practical experimental experience shows that the degree of agreement of the solutions of Euler equation to the experimental data is radically different case by case, or even in various regions of a flow field. The Euler equation solver cannot distinguish between these various cases by itself. The same proposition holds for NS equation solver. Any such system cannot clarify the limitation of its ability by itself.

But, let us consider the more fundamental question as to how an AI computer can count a number of things? Set theory constructs a natural number as a correspondence of null set, ϕ to 0, $\{\phi\}$ to 1, $\{\phi,\{\phi\}\}$ to 2, and so on. Here we do not discuss the construction method of natural number but consider the correspondence between 1 constructed like this and the real world. The problem is how to make an AI computer or system count a number of things. What kind of structure or architecture should be incorporated into an AI system which can answer the question in the situation: when there are a pencil, an eraser, a book, a paper, a tube of paint and a picture on a desk, the problem is to count the number of things on the desk? Also, there is a difficult problem of how to teach an AI system that the "one" appearing in statements like "there is one constitution in Japan," the one appearing in a statement "there is one cup of water," and the one appearing in a statement "I have one week vacation," corresponds to the set $\{\phi\}$.

In regard to this problem, the logician Takeuti's remarks are important[1–6]. He pointed out that for the cognition of a natural number, the action to move to next process is indispensable. Also Oka, a famous mathematician, said that a baby learns the meaning of the natural number one through various actions. According to Nishida, a philosopher, the cognition of the correspondence between thing and natural number needs an active intuition[2–3]. Thus, for the closed a priori mode of AI, it is difficult to obtain the correspondence between the knowledge and the real world as pointed out by Kant, and the knowledge acquisition should be performed by practical reason.

The typical case of this process is the invention of the microscope as mentioned previously. Owing to the practical reason appeared as microscope technique, we recognized bacteria and our world of knowledge was expanded. In fluids engineering, techniques of hot wire and flow visualization expanded our cognition of the flow world. What kind of magnification of intelligence of human being will be produced by AI? The microscope for intelligence composed of AI should be constructed practically in order to show its ability.

2. 3. *Symbol and perspective of intelligence*

In the present AI system, everything is performed as changes in the idea of numbers produced in a computer. Usually it is said to be done by the binary number system, but in reality the entity in the computer is a pattern of the electric potential distribution. And we assume the pattern to be a binary number system through the symbolization of the pattern by our mental act.

Symbols used in fluids engineering are not simple ones, but here they are used as a familiar system to the authors. For example, let the dependency of variables in Eq. (1.8) on the duct axis length be expressed and put in the following form:

$$A(x)v(x) - Q = 0. \tag{2.7a}$$

$A(x)$ is real positive, Q can take both signs, of which plus and minus means favorable and inverse flow in the x direction, respectively. Then $v(x)$ can take plus or minus value. In order to clarify the mathematical nature of the equation, it is put in the following form:

$$z(x)y(x) - w = 0, \quad z(x) > 0. \tag{2.7b}$$

By the same consideration, let Eq. (1.10) be in the following style:

$$\frac{1}{2} cy(x)^2 + t(\mathrm{x}) - d = 0. \tag{2.8}$$

From two equations (2.7b) and (2.8), we can solve simple fluids engineering problems, but an AI computer which knows only these equation systems can never realize what kind of world corresponds to these equation system. Although these equation systems can describe various fluid systems including air, water, oil, mercury and so on, if compressibility or cavitation becomes important, these equations fail to descirbe the real world and the condition cannot be derived by these equations.

Like the symbolization, the nature of the real world can be sampled and compressed to a symbol, and then some mappings from the real world into the world of real number become possible. Also, the process in the other direction — the reconstruction of the part of the real world from the symbols — should be made. Hence, the following schema can be formed:

$$RW \underset{(R).}{\overset{(M).}{\rightleftarrows}} SW. \tag{2.9}$$

where,

RW: a part of the real world

SW: a symbolic world

(M).: sampling and compression

(R).: reconstruction

The processes (M). and (R). include interpretation of the real world and the symbolic world, respectively. Of course, there arises the problem whether it is possible to regard (M). and (R). as worlds to construct a same schema for them or not.

If we consider the water flow in a pipe line, and if the AI computer predicts when one opens a tap he or she will be able to drink water, but only after he or she opens a real tap, he can know whether he or she can drink water or not. This situation asserts that the world is grasped totally by human beings only by the use of both the pure reason and the practical reason as a whole. The symbolization is eventually to entrust the essential action to the ignited mental act of the human being. The best example on this point is the $\epsilon-\delta$ argument of calculus. That is the argument which entrusts the mental practical act of the human being with taking further and further smaller positive real number. A man who is not ignited this mental practical act can never understand why the $\epsilon-\delta$ argument shows a process to take a limite.

Take another example. After symbolizing the flow world using Eq. (2.3), this symbolic world can be compressed to

$$\sum_{i=1}^{3} \frac{\partial^2 \phi}{\partial x_i^2} = 0, \quad \phi = \phi(x_i); \quad \text{velocity potential,} \tag{2.10}$$

under the condition of the irrotational flow: rot$\mathbf{u}$ = 0. This is Laplace's equation and appears in the theories of heat conduction, gravitational field, electro-magnetism and many other branches of science and engineering. So there are various reconstruction of the real worlds from the symbolic world of Eq. (2.10).

Let us consider a more simple situation. From a simple equation,

$$\mathrm{y} = ax^2 \tag{2.11}$$

there are many physical worlds that can be restored as follows. In the world of a spring it represents potential energy U of the spring:

$$U = \frac{1}{2} kx^2,$$

k : spring constant, x : elongation of the spring.

In the mechanical world of a mass point, kinetic energy K is

$$K=\frac{1}{2}mv^2,$$

m : mass of the pint, v : velocity of the point.

In the mechanical world of free fall, fall length H is

$$H=\frac{1}{2}gt^2,$$

g : gravitational acceleration, t : falling time.

In the electro-magnetic world, a capacitor stores energy W as

$$W=\frac{1}{2}CV^2,$$

C : capacitance, V : electric potential.

There are many other restorations allowed. From the above discussion it is apparent that the restoration of a real world from a symbolic world needs a not-symbolized world itself. In other words, we must use practical reason.

In this regards, Feynman's comment on the physical interpretation of an equation is very interesting. About the act to interpret an equation physically, he writes that it is an act to say how to use the equation to describe the experimental results[2–4]. According to the present context, he considered implicitly both the pure reason, the equation, and the practical reason to interpret it.

Also, the above discussion corresponds to Nietzsche's aphorism exactly.
"There is no occurrence in itself. The occurrence of things coming into existence is the group of thing interpreted by a interpreting ens." "A world which is interpreted uniquely by use of symbol is only a semblance, a fiction (compression and restoration by use of symbol—the present authors' comment). Since human beings govern the world using symbols, then the world inevitably has a fictious character. Furthermore, the thought as method for interpreting the symbolic world is inevitably plural as a projection fitting the present world."[2–5] The thought expressed by these aphorisms is the so called Perspectivism of intelligence originated by Nietzsche.

If we wish to reduce the diverse meanings resulting from the processes (M). and (R). in Eq. (2.9), we must become Phythagoreanist. For example, the number 9.8 which is the value of gravitational acceleration expressed in m/s^2 has almost no possibility of interpretation other than gravitational acceleration. We think this is the exact meaning of Feynman's remark: "We can not prove the mistake of a theory expressed ambiguously."[2–4] In other words, the theory must predict the number of numbers which can be compared with the experimental numbers. Also, we think that when we seek to make AI an exact science, we should construct an AI system from which we can derive a number comparable with the number obtained by an experiment which may be different from a conventional experiment.

2. 4. *Conditions for realizing an AI system*

As a conventional machine, an AI computer's ability is twofold,

(1) Simulation of the intelligent act of a human being.

(2) Expansion of the intelligent act of a human being as an aeroplane can give us flying ability.

In both cases, the judgment of the achievement of their goal is a problem. A proposed standard for such a judgment for (A) is Turing's test. However, it neglects completely the ability of vision, the ability of the sense of touch etc. Disregard for the ability of vision on the part of intelligence is a common imperfection of the present discussions on AI. For example, many medieval and Renaissance pictures were painted in order to express a world view, and an AI computer, which can realize such pictures, should have the vision ability. If we can construct such a machine system, the following questions remain.

What kind of output should be given by an AI computer which can process the picture as does the brain-vision system of a human being?

Assume that such a machine processes a picture. If the output of the machine is the same one as the picture, the machine does not perform any process. The impression produced by a picture in the human brain is not a simple conditioned reflex, but a synthesized thing forming one's own life experience and the stimulus from the picture and the atmosphere at that time.

Consider the more simple cases of flow visualization as shown in Fig. 2.1, which represents Kármán vortices in a cylinder wake. By seeing the figure, the impression of it obtained by a man is individual and he or she makes an individual mental act and sometimes expresses some particular output corresponding to each person. If he or she has a knowledge of Kármán vortex, the person can say that I see a Kármán vortex street. This is evidently the expression of compressing and symbolizing the world. In contrast, a person having no knowledge of a Kármán vortex cannot compress his or her impression and only can indicate the figure and say: "I see this." This method of communication is the one to transfer the information of a random phenomena totally (Kolmogorov). Conversely, a person having knowledge of a Kármán vortex can construct an image like Fig. 2.1 by hearing the words, while a person having no knowledge of it can get no idea from the words. Thus, the former can restore the world from the symbol, whereas the latter can not.

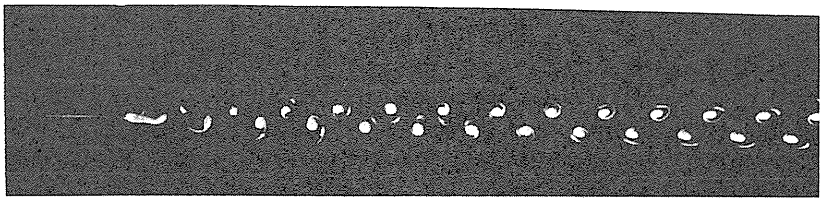


Fig. 2. 1. von Kármán vortex steet. $d = 3.5\,\mathrm{mm}$, $Re = 70$.
(Courtesy of Prof. M. Miyata, Yamanashi University.)

Prandtl (German fluids engineer) began to use these visualization technique as an efficient method to study various flows. He was a great scholar and discovered the boundary layer, but it seems that he did not completely understand the fluid mechanical meaning of various lines photographed on such a picture. It seems that the differences between streamline, particle path and streak line were not apparent for him. If an AI machine's output about

Fig. 2.1 is symbolized using these concepts of various lines, even Prandtl might struggle to understand the output. Here the problem arises with respect to (b), that is,

(c) An AI computer should have the ability to explain its output to human beings.

Usually a human being considers that firm belief about the truth of some results is obtained when he or she can follow the reasoning step by step as in the case of mathematical proof. However, one of the goals, (b), of AI is to perform the extraordinarily lengthy and complex reasoning very quickly which can not be checked by a man. When a man can follow such a reasoning of an AI computer step by step, such an AI computer performs (a) and has no function as (b). If we can implement the function (c) completely to some AI computer, it does not perform the function (b) completely. If a super AI computer has the function (b), its output may sometimes be non ens and sometimes be Delphic oracles for human beings.

The structure of a jet engine is completely different from a cuttlefish using a jet to swim, so it is not necessary that the knowledge processing in AI computer be the same as in a human being. However at the interface between man and AI computer, the input and the output of AI computer must fit the human being. If not, the AI computer has no mental connection with man as the self-proliferating automaton of von Neumann.

It is apparent from the above discussions that we should study the science of AI not only theoretically but also practically, experimentally and individually. At first sight, AI can be pursued by only the theoretical method, but it has an experimental aspect and permits no rigorous proof for its action. Fundamental AI theory is based on natural number and set theory. However, there is a Takeuti's notation: "The construction of the natural number is founded on the intuitive grasp of the mental act to go next or to annex. (This is the notion of active intuition, the central idea of philosopher Nishida.) Even in set theory, the idea of going to the next stage is inevitable." With respect to the action, that is "to do such a mental act" or "to begin such an interior act," we call it the triggering or the igniting of the mental act.

In what follows a proposition or a working hypothesis for general AI system will be discussed. In order for a person A to perform communication with an other person B about something, there must be some common ens or being for A and B. Our first assumption is,

(i) The idea, in the meaning of Plato, of a thing exists as the ens which is recognized commonly by both A and B about the thing.

It is permitted for both A and B to be the thoughts of a person. So we assume a duality of self in some sense. Another hypothesis is the proposal of Fujisawa for the theory of ideas[2–6]. According to his expression, the following mode can be formulated for the idea:

(ii) The idea of a desk is projected on this place in the real world now.

The other expression may be,

"The idea of desk makes a thing a desk when the idea is mapped on the thing existing here in the space at now"

This means that the idea is an abstract thing which makes something a thing by a mapping in the space and time. The space, in these sentences, may be an abstract one.

In Fig. 2.2, a picture expressing the AI system according to the above consideration is shown. Individual A which means a concrete thing like a desk but also abstract things like proposition and strings of symbols, exists in the domain of discourse D and has an image of an idea I_A in the world of idea I. In the human being's world, that is the world of representation M, I_A is projected on representation S_A and I_A is related to symbol M_A in the world of AI computer C.

When M_A is transformed into M_B by some transformation T_C in the AI computer, representation S_B having symbol M_B is related to representation S_A through the operation of T_M in the world of representation where T_M has symbol T_C. Similarly, in D, there are individual B and action T_D which have images S_B and T_M in M. Changes in D, M and C can be

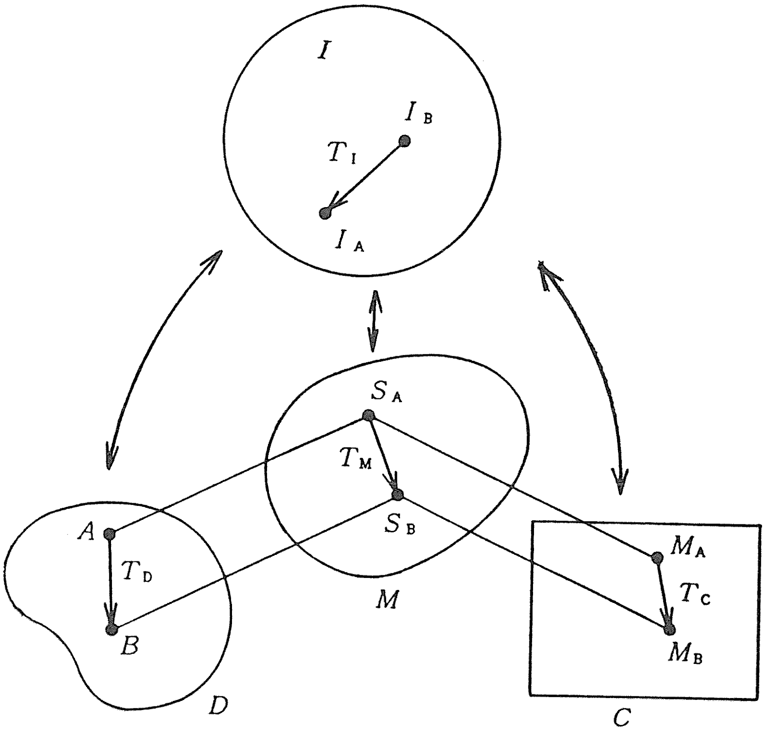


Fig. 2. 2. A scheme of the process of AI.
D: domain of discourse
M: human being's world
C: AI computer
I: world of idea

connected through these schema. The assumptions(i) and (ii) guarantee the existence of the idea of a thing and the possibility of restoration of a thing from the idea, respectively.

Also T_C, T_M and T_D should be considered as images of an idea in I. Aporia arises here as the third man discussed by Parmenides, which is the question about the relationship between the three worlds D, M, C and the world of idea I. This question may be considered to lead to endless retroaction. Here, we avoid this classical aporia to assume simply that in this case the real world consists of D, M and C and the relationship between them, and everything and mapping in the world are such a thing since they are the image of the idea.

The assumed schema given above is an axiom to establish the realizability of AI computer and the meanings of output from it. Without this assumption, we consider that the relationship between the domain of discourse — the representation in human being — and AI computer is lost.

References of chap. II

2-1) Nitta, Y., "History of philosophy," (in Japanese), (1989), 109, Kodansha.
2-2) Kosaka, S., "History of European philosophy," (in Japanese), (1984), 206, JICC Press.
2-3) Nishida, K., Active intuition in "Logic and life", ed. Ueda, K., (in Japanese), (1988), Iwanami.
2-4) Feynman, R., "The character of physical law," trans. Ezawa, H., (1968), 267, Diamond Press.
2-5) See ref. 2-1), p. 184.
2-6) Fujisawa, N., "Idea and the world," (in Japanese), (1980), 54, Iwanami.

III. Automatic Generation of FORTRAN Code for Finite Difference Method

3. 1. Introduction

When the differential equations governing the physical problem cannot be solved analytically, it is useful to solve the equations numerically for engineering as well as for understanding the physical phenomena. In order to obtain the numerical solutions, the computer program must be prepared by processing the equations with a routine and laborious procedure. In this paper, we attempted to carry out automatically such monotonous and straightforward work by the symbol manipulator of the computer. This automatic manipulation is expected to save time and labor and to avoid mistakes in processing.

Automatic programming has been studied for years[3-1~3-3]. In mechanical engineering, general-purpose programs, such as those by Kleinsteuer and Patterson[3-4], Rosten, Spalding and Tatchell[3-5], Rice and Boisvert[3-6] and Henry and Willmost[3-7], have been developed. On the other hand, some applications of the symbolic processing faculty of computers are reported, which perform derivation of motion equations[3-8], analysis of the incompressible Navier-Stokes equations[3-9], code generation for eigenvalue determination[3-10] and so on.

There are not many studies, at least known to the authors, dealing with automatic program generation for the finite difference method in fluid mechanics. Rosen and Okabayashi[3-11] tried automatic processing of the second-order parabolic partial differential equations governing the plasma fluid. They developed a FORMAC (Formula Manipulation Compiler) program which prepares, by means of the marching finite difference method, the linearized and difference equations and generates a part of the FORTRAN code necessary to calculate the unknowns at grid points. This kind of programming was also carried out by Takeda and Itoh[3-12]. Steinberg and Roache[3-13] wrote a symbolic manipulation program in VAXIMA for the linear second-order elliptic differential equation. Their program has as input the differential equation in some natural coordinates and has as output the FORTRAN subroutines which compute the coefficients necessary to solve the difference equation and the coefficients for determining the coordinate transformations. However, the equations to be processed by these formula manipulating programs are restricted, and these programs do no seem to be applicable to other types of problems.

A new simple program is herewith developed to use as input the nonlinear parabolic differential equations and some accompanied conditions and to produce as output the FORTRAN code for the iterative calculation to solve the nonlinear difference equations by Newton's method. Keller's box method[3-14] is used, so there is no restriction on the order of the differential equations. The manipulation program is written in LIST (evalquote LISP). The details of the LISP program and its adaptation for ordinary and partial differential equations are discussed.

3. 2. Formulation and automatic generation of FORTRAN code

3. 2. 1. Formulation

The parabolic differential equations are solved by an implicit method which obtains the net functions on one space coordinate simultaneously. This implicit method proceeds in the direction of other independent variables. Differential equations can be written in terms of a first-order system of the partial differential equations, by introducing new dependent variables, as follows:

$$\Psi_i(x_1,\cdots,x_l,y_1,\cdots,y_n,\frac{\partial y_1}{\partial x_1},\frac{\partial y_1}{\partial x_2},\cdots,\frac{\partial y_n}{\partial x_l}), \qquad (i=1,\cdots,n). \tag{3.1}$$

Here the equations are given in $(x_1,\cdots,x_l)$-space and $y_1,\cdots,y_n$ are the dependent variables including newly introduced ones.

Eqs. (3.1) are approximated by the central difference method. The simultaneous difference equation for Eqs. (3.1) can be written as

$$\Delta_{i\,j-1/2}(y_{1\,j-1},\cdots,y_{n\,j-1},\,y_{1\,j},\cdots,y_{n\,j})=0, \qquad (i=1,\cdots,n \quad j=1,\cdots,j_{max}). \tag{3.2}$$

where the two adjacent points on the space coordinate and the midpoint of these two points are denoted by the subscripts $j-1$, j and $j-1/2$, respectively, and the subscript *max* designates the total number of steps in the space coordinate. These difference equations generally contain known variables and coefficients. The nonlinear Eqs. (3.2) are solved by means of Newton's method. For the iterates λ, the higher-order iterates are introduced as follows:

$$y^{\lambda+1}_{p\,q}=y^{\lambda}_{p\,q}+\delta y^{\lambda}_{p\,q}, \qquad (p=1,\cdots,n \quad q=j-1,j \quad j=0,\cdots,j_{max}). \tag{3.3}$$

Using Eqs. (3.2) and (3.3) yields the linear system

$$\sum_{p=1}^{n}\sum_{q=j-1}^{j}A_{p\,q}\,\delta y^{\delta}_{p\,q}=-\Delta_{i\,j-1/2}(y^{\lambda}_{i\,j-1},\cdots y^{\lambda}_{n\,j}), \qquad (j=1,\cdots,n \quad j=1,\cdots,j_{max}). \tag{3.4}$$

which is necessary for the iteration process. Eq. (3.4) and appropriate boundary conditions are the system of $n\times(j_{max}+1)$ equations with $n\times(j_{max}+1)$ unknowns. The coefficients $A_{p\,q}$ are given by

$$A_{p\,q}=\frac{\partial\Delta_{i\,j-1/2}}{\partial y^{\lambda}_{p\,q}}. \tag{3.5}$$

3. 2. 2. Automatic generation of FORTRAN code

The LISP program has as input Eqs. (3.1), independent and dependent variables and the boundary conditions and produces the FORTRAN code using the procedure mentioned in Sec. 3.2.1. This FORTRAN code consists of the following three parts:

1. The coefficient matrix and the right-hand side of Eq. (3.4) are determined.
2. The linear system of Eq. (3.4) is solved using the ready-made subroutine program.
3. The higher-order iterates are obtained from Eq. (3.3).

In general, the executable FORTRAN program solving the difference equations includes some portions: input, output parts, the parts which set the known parameters in equations and examine the convergence of the iterative procedure, and the main part of the program which sets the system of algebraic linear equations and obtains the higher-order iterates. The calculation for determining the unknown net functions on one space coordinate is performed following the flow chart shown in Fig. 3.1. First, the values of independent variables on grid

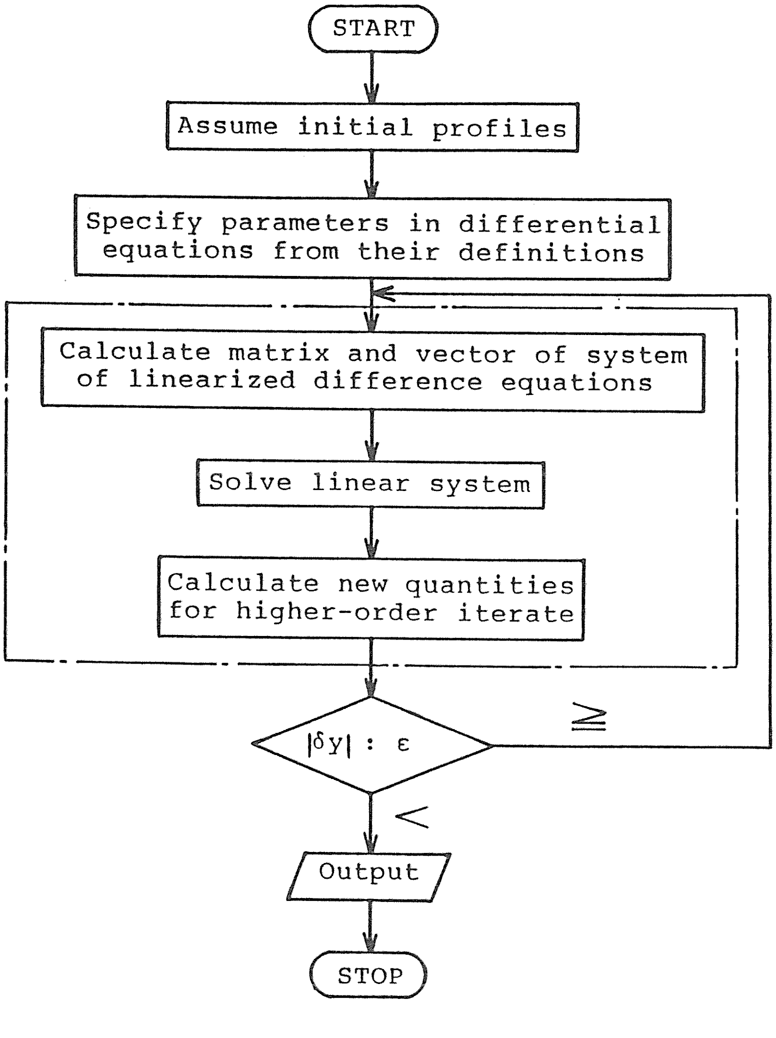

START
Assume initial profiles
Specify parameters in differential equations from their definitions
Calculate matrix and vector of system of linearized difference equations
Solve linear system
Calculate new quantities for higher-order iterate
|δy| : ε
≧
<
Output
STOP

Fig. 3. 1. Flow chart for calculation on one space coordinate.

points are determined and the initial profiles are assumed for the unknown functions which satisfy the boundary conditions. Second, the known parameters appearing explicitly in the equations are set. Then the iteration procedure is repeated until the solution converges. The present LIST program produces the required FORTRAN code. The produced part is enclosed by the chain line in Fig. 3.1. The executable FORTRAN program can be prepared by adding together the input, output parts, convergence examining part and others. The total procedure to solve the differential equations is shown in Fig. 3.2.

The LISP program has the FORTRAN code as output. Therefore, it is not the kind of general-purpose program which solves problems using existing subprograms and selecting an algorithm.

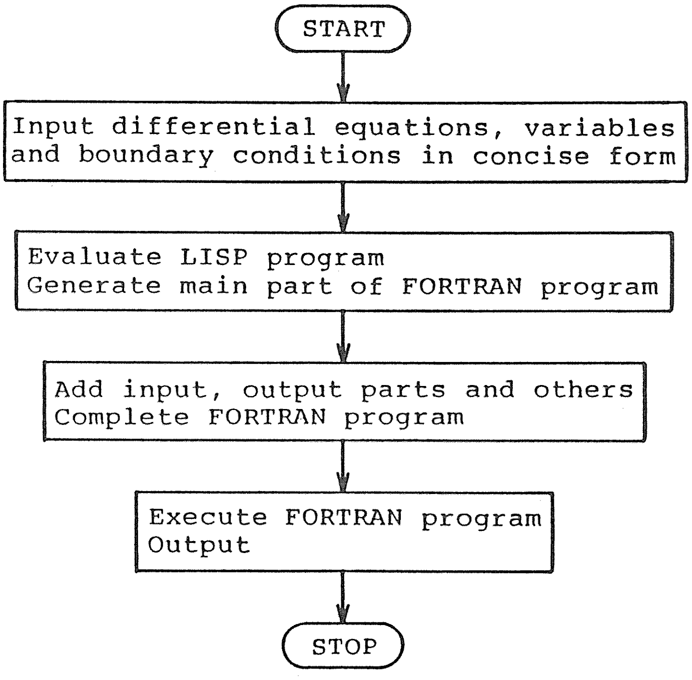

START
Input differential equations, variables and boundary conditions in concise form
Evaluate LISP program
Generate main part of FORTRAN program
Add input, output parts and others
Complete FORTRAN program
Execute FORTRAN program
Output
STOP

Fig. 3. 2. Present procedure to solve differential equations.

3.3. *Application of automatic generation*

An example of automatic generation for ordinary differential equations will be illustrated. As seen later, partial differential equations are also solved in a similar way. Let us consider the following ordinary differential equations:

$$\begin{aligned} &f''' + ff'' + \beta(1 - f'f' + S) = 0, \\ &S'' + fS' = 0, \end{aligned} \tag{3.6}$$

which have self-similar solutions of a compressible laminar boundary layer flow[3-15]. In Eqs. (3.6), f and S are a nondimensional stream function and a stagnation enthalpy, respectively, β is a pressure gradient parameter and primes denote the differentiation with respect to η, which is the nondimensional distance from the wall. The nondimensional velocity component in the direction of the main stream is given by f'. The boundary conditions are as follows:

$$\begin{aligned} &\eta = 0: \quad f = 0, \quad f' = 0, \quad S = S_0, \\ &\eta = \eta_e: \quad f' = 1, \quad S = 0, \end{aligned} \tag{3.7}$$

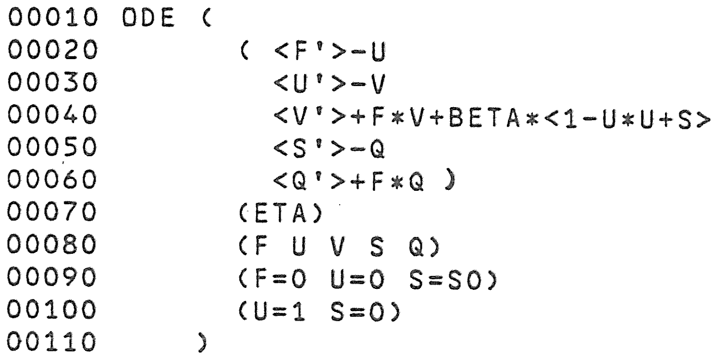
where the subscripts 0 and e denote the points on the wall and at the outer edge of the boundary layer, respectively. Following the box method, Eqs. (3.6) are replaced by a system of first-order equations, introducing new dependent variables u, v and q as

$$\begin{aligned} &f' - u = 0, \\ &u' - v = 0, \\ &v' + fv + \beta(1 - uu + S) = 0, \\ &S' - q = 0, \\ &q' + fq = 0. \end{aligned} \tag{3.8}$$

The boundary conditions are

$$\begin{aligned} &\eta = 0: \quad f = 0, \quad u = 0, \quad S = S_0, \\ &\eta = \eta_e: \quad u = 1, \quad S = 0. \end{aligned} \tag{3.9}$$

The form of the input data is shown in Fig. 3.3. In this figure, ODE at line number 10 denotes the LISP function which processes the ordinary differential equations. The input data



```
00010 ODE (
00020        ( <F'>-U
00030          <U'>-V
00040          <V'>+F*V+BETA*<1-U*U+S>
00050          <S'>-Q
00060          <Q'>+F*Q )
00070        (ETA)
00080        (F  U  V  S  Q)
00090        (F=0  U=0  S=SO)
00100        (U=1  S=0)
00110     )
```

Fig. 3. 3. Input data for ordinary differential equations.

for equations, variables and boundary conditions are regarded as arguments of the function ODE. In Fig. 3.3, the meanings of the data are as follows:

1. Between line numbers 20 and 60, input of the differential Eqs. (3.8). As parentheses play a special role in LISP system, the symbols $<$ and $>$ are used instead of them. Differentiations with respect to a space variable η, such as f', are denoted by <F'> and others.
2. At line number 70, input of the independent variable and variables to be expressed by the array element names in FORTRAN code, if any.
3. At line number 80, input of the dependent variables.
4. Between line numbers 90 and 100, input form of Eqs. (3.9), which are the boundary conditions at two points.

Taking the case of the third equation in Eqs. (3.8), Fig. 3.4 shows the process to obtain the

```
1. Input        <V'>+F*V+BETA*<1-U*U+S>
                           ↓
                ((DF V) + F * V + BETA * (1 - U * U + S))

2. Δ3 j-1/2 =   ((V (J) - V (J-1)) / DETA + 0.5 * (F (J) * V (J) + F (J-1) *
                V (J-1)) + 0.5 * (BETA * (1 - U (J) * U (J) + S (J)) + BETA
                * (1 - U (J-1) * U (J-1) + S (J-1))))

3. Unknowns             Coefficients
      δFj-1         ∂Δ3 j-1/2/∂Fj-1 = (0.5 * V (J-1))
      δFj           ∂Δ3 j-1/2/∂Fj   = (0.5 * V (J))
      δUj-1         ∂Δ3 j-1/2/∂Uj-1 = (0.5 * BETA * (- U (J-1) - U(J-1)))
      δUj           ∂Δ3 j-1/2/∂Uj   = (0.5 * BETA * (- U (J) - U (J)))
      δVj-1         ∂Δ3 j-1/2/∂Vj   = (0.5 * BETA)
      δVj           ∂Δ3 j-1/2/∂Vj-1 = (0.5 * BETA)

4.                Δ3 j-1/2
                    ↓
   Outputs       R(N+3)=-(V(J)-V(J-1))/DETA-0.5*(F(J)*V(J)+F(J-1)*V(J-1))
                &        -0.5*(BETA*(1-U(J)*U(J)+S(J))+BETA*(1-U(J-1)*U(J-1)
                &        +S(J-1)))
                  .        .        .        .        .        .
                  .        .        .        .        .        .
                  .        .        .        .        .        .
```

Fig. 3. 4. Processing of differential equation in LISP program.

difference equations and determine the coefficients in Eq. (3.4), which is as follows:

1. The equation is transformed into the list notation. In the transformed equation, DF stands for the prime.
2. Each term in the list notation is approximated by the central difference method. The dependent variables are replaced by the array element names. The subscripts $j-1$ and j correspond to two adjacent points on the space coordinate. Derivative (DF V) is replaced with

   (V(J) − (J-1))/DETA

   where DETA $= \eta_j - \eta_{j-1}$ is the interval of the grid line. The terms other than derivatives and the coefficients multiplied by derivatives are replaced with arithmetic averages. For example, F $*$ V is approximated by

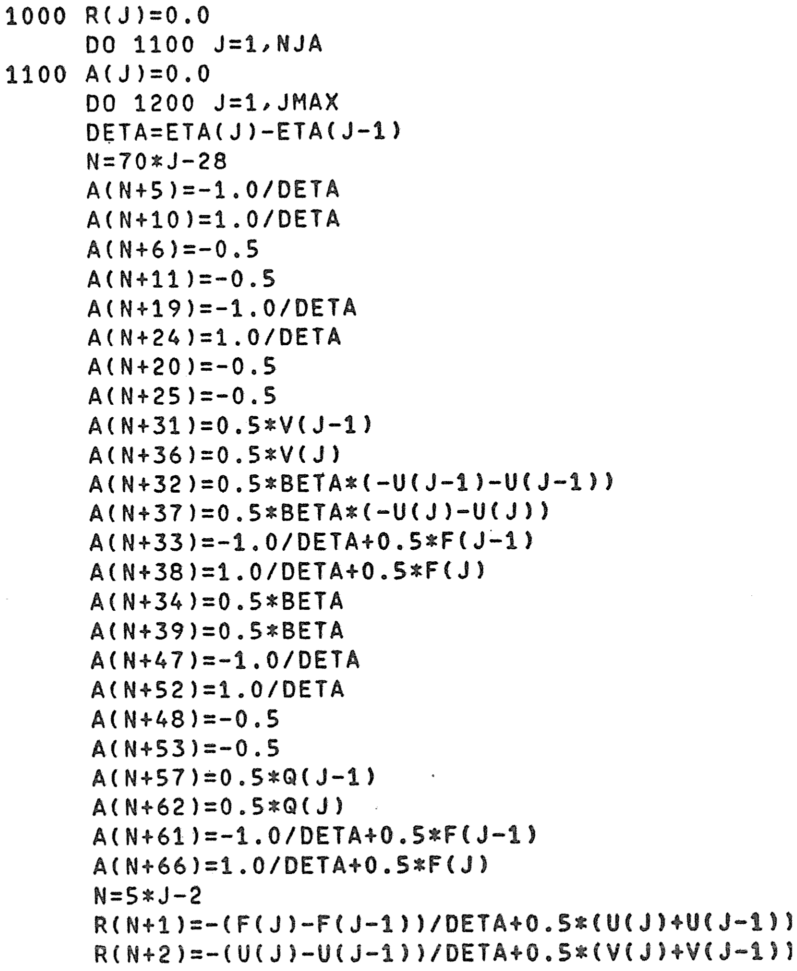
   0.5 $*$ (F(J) $*$ V(J) + F(J-1) $*$ (J-1)).

   Then the difference equation, denoted by $\Delta_{3\,j-1/2}$, is obtained.

3. For this difference equation, the coefficients multiplied by unknowns δf_{j-1}, δf_j, $\cdots$ in Eq. (3.4) are obtained by the derivative of $\Delta_{3\ j-1/2}$ with respect to each term, F(J-1), F(J), $\cdots$ in the equation of list expression.
4. The difference equation and the coefficients in the list notation are written in FORTRAN as output. In the statement in FORTRAN, the right-hand side of Eq. (3.4) corresponds to the array name R.

Evaluating the LISP function in Fig. 3.3 yields the main part of the FORTRAN program, a part of which is shown in Fig. 3.5. In Fig. 3.5, the array name A denotes the coefficient matrix in Eq. (3.4). The dependent and independent variables are written in the array element names. The input type of the coefficient matrix and the right-hand side vector may depend on the subprogram which solves the linear system. However, the present LISP program can easily be adapted for any type of subprogram by a minor change of the defined LISP function.

The executable FORTRAN program is constructed by adding the necessary parts to the



```
1000 R(J)=0.0
     DO 1100 J=1,NJA
1100 A(J)=0.0
     DO 1200 J=1,JMAX
     DETA=ETA(J)-ETA(J-1)
     N=70*J-28
     A(N+5)=-1.0/DETA
     A(N+10)=1.0/DETA
     A(N+6)=-0.5
     A(N+11)=-0.5
     A(N+19)=-1.0/DETA
     A(N+24)=1.0/DETA
     A(N+20)=-0.5
     A(N+25)=-0.5
     A(N+31)=0.5*V(J-1)
     A(N+36)=0.5*V(J)
     A(N+32)=0.5*BETA*(-U(J-1)-U(J-1))
     A(N+37)=0.5*BETA*(-U(J)-U(J))
     A(N+33)=-1.0/DETA+0.5*F(J-1)
     A(N+38)=1.0/DETA+0.5*F(J)
     A(N+34)=0.5*BETA
     A(N+39)=0.5*BETA
     A(N+47)=-1.0/DETA
     A(N+52)=1.0/DETA
     A(N+48)=-0.5
     A(N+53)=-0.5
     A(N+57)=0.5*Q(J-1)
     A(N+62)=0.5*Q(J)
     A(N+61)=-1.0/DETA+0.5*F(J-1)
     A(N+66)=1.0/DETA+0.5*F(J)
     N=5*J-2
     R(N+1)=-(F(J)-F(J-1))/DETA+0.5*(U(J)+U(J-1))
     R(N+2)=-(U(J)-U(J-1))/DETA+0.5*(V(J)+V(J-1))
```

Fig. 3. 5. Part of generated FORTRAN code for input in Fig. 3.3.

code in Fig. 3.5. The velocity and stagnation enthalpy profiles obtained by this FORTRAN program are shown in Fig. 3.6.

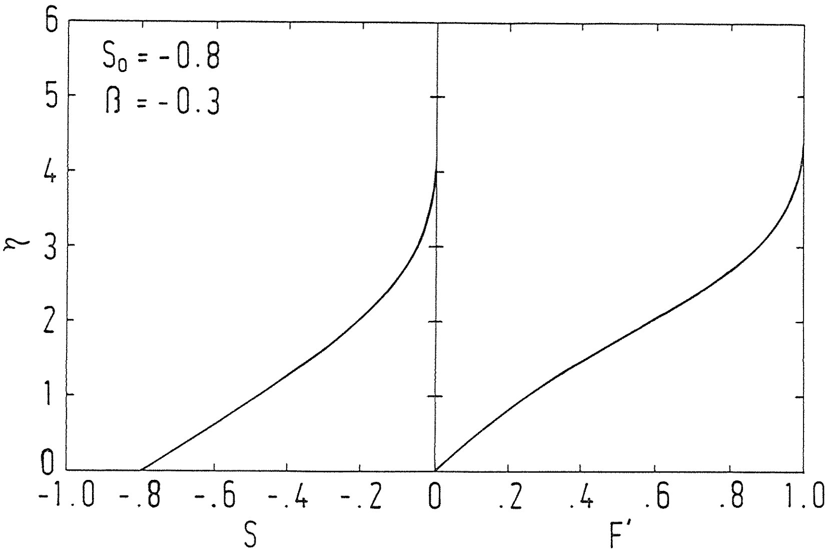


Fig. 3. 6. Velocity and stagnation enthalpy profiles of similar compressible boundary layer.

3. 4. Boundary layers on rotating cone in linearly retarded external streams

The authors have investigated the flow on a rotating circular cylinder in a retarded axial flow$^{3-16,3-17)}$. As an example for the partial differential equations, the LISP program is applied to the problem of the boundary layer on a rotating cone in an axial flow. When the external flow was the potential flow for a cone, Koh and Price$^{3-18)}$ solved this problem by means of the finite difference method. In this chapter, the external flow is assumed to be linearly retarded with the distance from the apex along the generator. In the coordinate system shown in Fig. 3.7, the continuity, momentum and energy equations are

$$\begin{aligned}
&\frac{\partial U}{\partial x}+\frac{U}{x}+\frac{\partial W}{\partial z}=0,\\
&U\frac{\partial U}{\partial x}+W\frac{\partial U}{\partial z}-\frac{V^2}{x}=U_e\frac{dU_e}{dx}+\nu\frac{\partial^2 U}{\partial z^2},\\
&U\frac{\partial V}{\partial x}+W\frac{\partial V}{\partial z}+\frac{UV}{x}=\nu\frac{\partial^2 V}{\partial z^2},\\
&U\frac{\partial T}{\partial x}+W\frac{\partial T}{\partial z}=\frac{\nu}{Pr}\frac{\partial^2 T}{\partial z^2},
\end{aligned}\qquad(3.10)$$

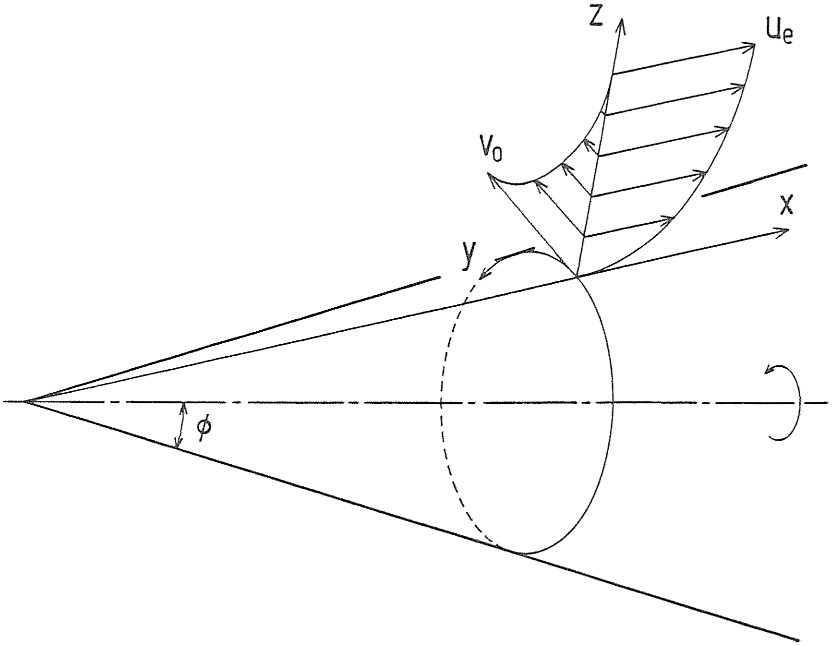


Fig. 3. 7. Flow past a rotating cone.

with the boundary conditions

$$\begin{aligned} &z=0: \quad U=W=0, \quad V=V_0=\omega x\,\mathrm{sint}\phi, \quad T=T_0, \\ &z=z_e: \quad U=U_e, \quad V=0, \quad T=T_e. \end{aligned} \tag{3.11}$$

Here U, V and W are the velocity components in the x, y and z directions, respectively, T is temperature, ν the kinematic viscosity, Pr the Prandtl number, ω an angular velocity of the cone and ϕ the cone half angle. The external flow U_e is assumed to be

$$U_e = U_m - cx, \tag{3.12}$$

where c is a constant. Using the stream function ψ given by

$$xU=\frac{\partial\psi}{\partial z}, \quad xW=-\frac{\partial\psi}{\partial x}, \tag{3.13}$$

and nondimensional quantities

$$\begin{aligned} &x^* = \frac{cx}{U_m}, \quad \eta=\sqrt{\frac{U_m}{\nu x}}\,z, \\ &f=\frac{\psi}{x\sqrt{U_m \nu x}}, \quad g=\frac{V}{V_0}, \quad \theta=\frac{T-T_e}{T_0-T_e}. \end{aligned} \tag{3.14}$$

Eqs. (3.10) and (3.11) are transformed into

$$
\begin{aligned}
& f''' + \frac{3}{2} ff''\alpha^2 x^{*2} g^2 + x^*(x^* - 1) + x^* f'' \frac{\partial f}{\partial x^*} - x^* f' \frac{\partial f'}{\partial x^*} = 0, \\
& g'' + \frac{3}{2} fg' - 2f'g + x^* \frac{\partial f}{\partial x^*} g' - x^* f' \frac{\partial g}{\partial x^*} = 0, \\
& \frac{\theta''}{Pr} + \frac{3}{2} f\theta' + x^* \frac{\partial f}{\partial x^*} \theta' - x^* f' \frac{\partial \theta}{\partial x^*} = 0,
\end{aligned}
\tag{3.15}
$$

and

$$
\begin{aligned}
& \eta = 0: \quad f = 0, \quad f' = 0, \quad g = 1, \quad \theta = 1, \\
& \eta = \eta_e: \quad f' = 1 - x^*, \quad g = 0, \quad \theta = 1,
\end{aligned}
\tag{3.16}
$$

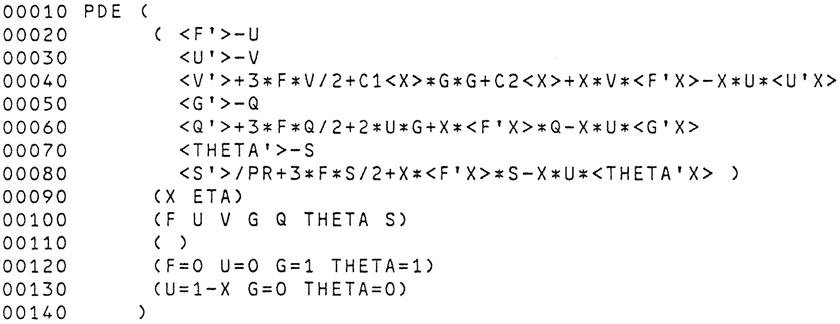
where

$$\alpha = \omega \sin(\frac{\phi}{c}), \quad \frac{U}{U_m} = f'.$$

Although the energy equation can be solved independently after the calculation of the momentum equations, in this example Eqs. (3.15) are solved simultaneously. New dependent variables u, v, q and s are introduced as follows:

$$f' - u = 0, \quad u' - v = 0, \quad g' - q = 0, \quad \theta' - s = 0. \tag{3.17}$$

Then Eqs. (3.15) and (3.16) are written in terms of the first-order derivatives. The input data in this case are shown in Fig. 3.8. The LISP function PDE processes the partial differential equations. For simplification of the expression, the terms $x^*(x^* - 1)$ and $\alpha^2 x^{*2}$ are replaced by C1<X> and C2<X>, respectively, in Fig. 3.8. Of course, these replacements are not always necessary. The differences of the input form from that for the function ODE in Fig. 3.3 are as follows:



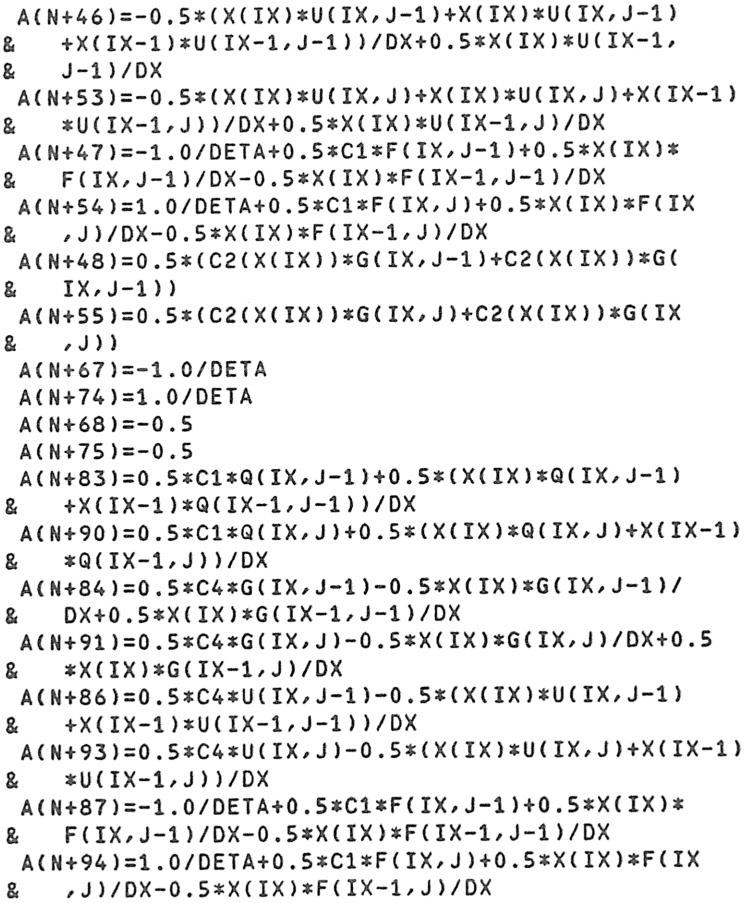
```
PDE (
      ( <F'>-U
        <U'>-V
        <V'>+3*F*V/2+C1<X>*G*G+C2<X>+X*V*<F'X>-X*U*<U'X>
        <G'>-Q
        <Q'>+3*F*Q/2+2*U*G+X*<F'X>*Q-X*U*<G'X>
        <THETA'>-S
        <S'>/PR+3*F*S/2+X*<F'X>*S-X*U*<THETA'X> )
      (X ETA)
      (F U V G Q THETA S)
      ( )
      (F=0 U=0 G=1 THETA=1)
      (U=1-X G=0 THETA=0)
    )
```

Fig. 3. 8. Input data for partial differetial equations.

1. Derivatives with respect to timelike variables, such as $\partial f/\partial x^*$, are denoted by <F'X> and others. In this example, x^* is the only one timelike variable.
2. Among the known functions of the independent variables, such as C1 and C2 in this example, the functions which are expected to be expressed by the FORTRAN array element names must be specified as the fourth argument of the function PDE. In this example, the position is at line number 110.

After evaluating the LISP function in Fig. 3.8, the FORTRAN code in Fig. 3.9 is obtained automatically. In Fig. 3.9, the independent and dependent variables are written by one- and two-dimensional array element names, respectively, and the subscript for timelike variable x^* (X in input) is denoted by IX. The known functions C1 and C2 which are not specified at line number 110 in Fig. 3.8 can be defined by the statement functions or the function subprograms.

The LISP program can also be applied to the case of the variable kinematic viscosity and Prandtl number, and to the case of the turbulent boundary layer equations closed by using an appropriate turbulence model. Furthermore, as was suggested earlier, the problems with



```
 A(N+46)=-0.5*(X(IX)*U(IX,J-1)+X(IX)*U(IX,J-1)
&    +X(IX-1)*U(IX-1,J-1))/DX+0.5*X(IX)*U(IX-1,
&    J-1)/DX
 A(N+53)=-0.5*(X(IX)*U(IX,J)+X(IX)*U(IX,J)+X(IX-1)
&    *U(IX-1,J))/DX+0.5*X(IX)*U(IX-1,J)/DX
 A(N+47)=-1.0/DETA+0.5*C1*F(IX,J-1)+0.5*X(IX)*
&    F(IX,J-1)/DX-0.5*X(IX)*F(IX-1,J-1)/DX
 A(N+54)=1.0/DETA+0.5*C1*F(IX,J)+0.5*X(IX)*F(IX
&    ,J)/DX-0.5*X(IX)*F(IX-1,J)/DX
 A(N+48)=0.5*(C2(X(IX))*G(IX,J-1)+C2(X(IX))*G(
&    IX,J-1))
 A(N+55)=0.5*(C2(X(IX))*G(IX,J)+C2(X(IX))*G(IX
&    ,J))
 A(N+67)=-1.0/DETA
 A(N+74)=1.0/DETA
 A(N+68)=-0.5
 A(N+75)=-0.5
 A(N+83)=0.5*C1*Q(IX,J-1)+0.5*(X(IX)*Q(IX,J-1)
&    +X(IX-1)*Q(IX-1,J-1))/DX
 A(N+90)=0.5*C1*Q(IX,J)+0.5*(X(IX)*Q(IX,J)+X(IX-1)
&    *Q(IX-1,J))/DX
 A(N+84)=0.5*C4*G(IX,J-1)-0.5*X(IX)*G(IX,J-1)/
&    DX+0.5*X(IX)*G(IX-1,J-1)/DX
 A(N+91)=0.5*C4*G(IX,J)-0.5*X(IX)*G(IX,J)/DX+0.5
&    *X(IX)*G(IX-1,J)/DX
 A(N+86)=0.5*C4*U(IX,J-1)-0.5*(X(IX)*U(IX,J-1)
&    +X(IX-1)*U(IX-1,J-1))/DX
 A(N+93)=0.5*C4*U(IX,J)-0.5*(X(IX)*U(IX,J)+X(IX-1)
&    *U(IX-1,J))/DX
 A(N+87)=-1.0/DETA+0.5*C1*F(IX,J-1)+0.5*X(IX)*
&    F(IX,J-1)/DX-0.5*X(IX)*F(IX-1,J-1)/DX
 A(N+94)=1.0/DETA+0.5*C1*F(IX,J)+0.5*X(IX)*F(IX
&    ,J)/DX-0.5*X(IX)*F(IX-1,J)/DX
```

Fig. 3. 9. Part of generated FORTRAN code for input in Fig. 3.8.

multiple timelike variables, such as unsteady two-dmensional boundary layer, can also be processed by the present LISP program.

The flows on a rotating cone in linearly retarded axial streams are investigated using the FORTRAN code in Fig. 3.9. The initial profiles at the apex can be obtained by solving Eqs. (3.15) and (3.16) at $x^* = 0$. The Prandtl number is set at 0.7. Fig. 3.10 shows the profiles of the axial, circumferential velocity components and temperature at $\alpha = 1$, $x^* = 0.25$. At a relatively small α, the axial velocity component is retarded, and finally the flow separates from the cone wall. Fig. 3.11 shows the variation of the axial velocity component at $\alpha = 4$.

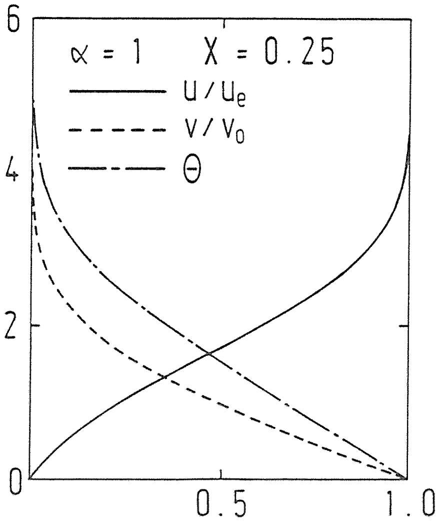


Fig. 3. 10. Profiles of axial, circumferential velocity components and temperature at relatively small rotation speed.

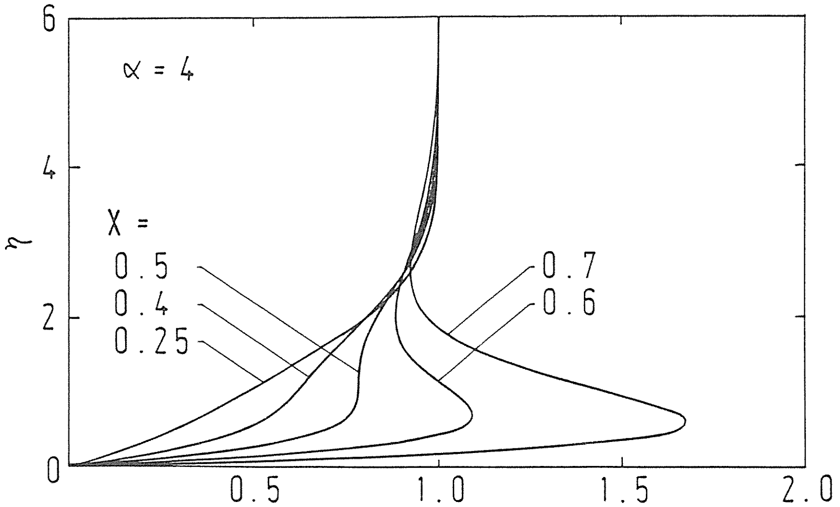


Fig. 3. 11. Development of axial velocity component.

As the flow develops, the velocity is accelerated in spite of the external adverse pressure gradient. Then an overshoot occurs in the middle portion of the boundary layer, where the circumferential velocity component is not small. Figs. 3.12, 3.13 and 3.14 show the variations of

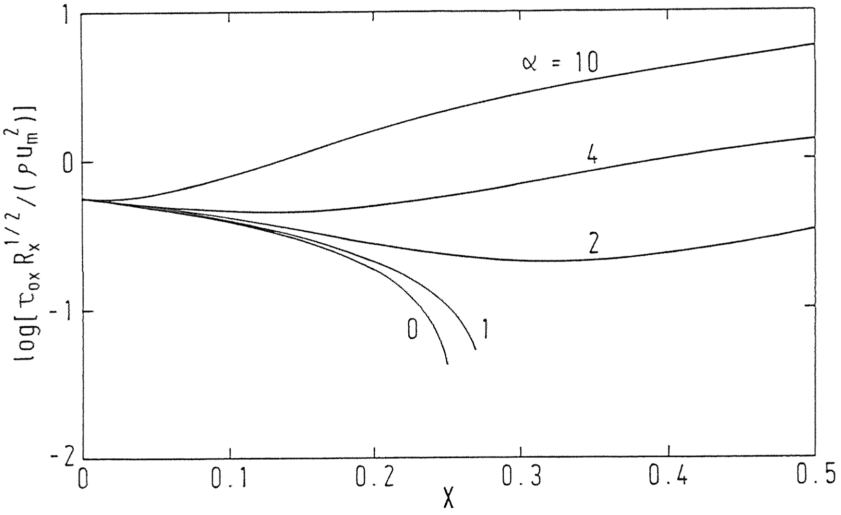


Fig. 3. 12. Variation of axial wall shear stress in the axial direction.

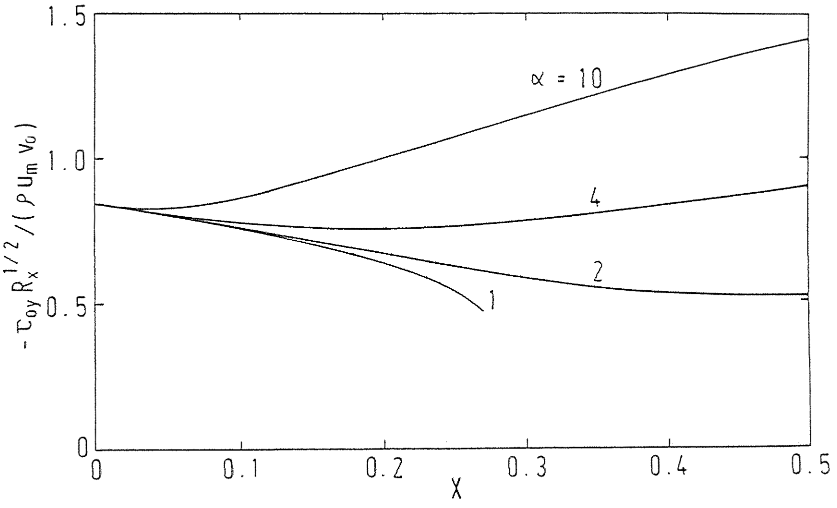


Fig. 3. 13. Variation of circumferential wall sthear stress in the axial direction.

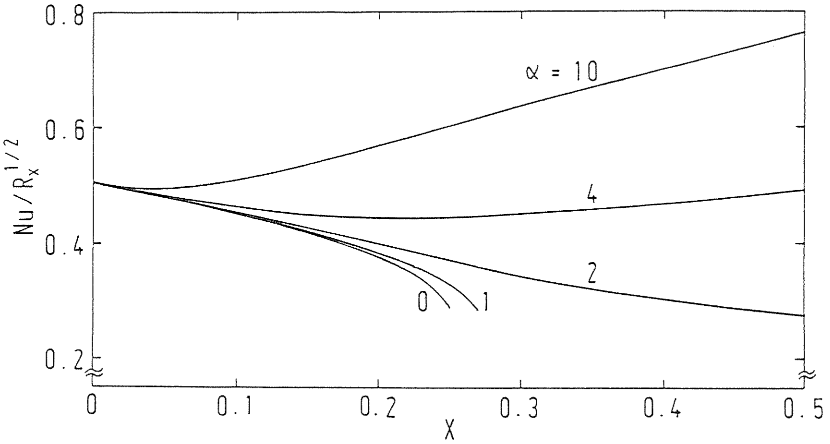


Fig. 3. 14. Variation of local heat transfer coefficient in the axial direction.

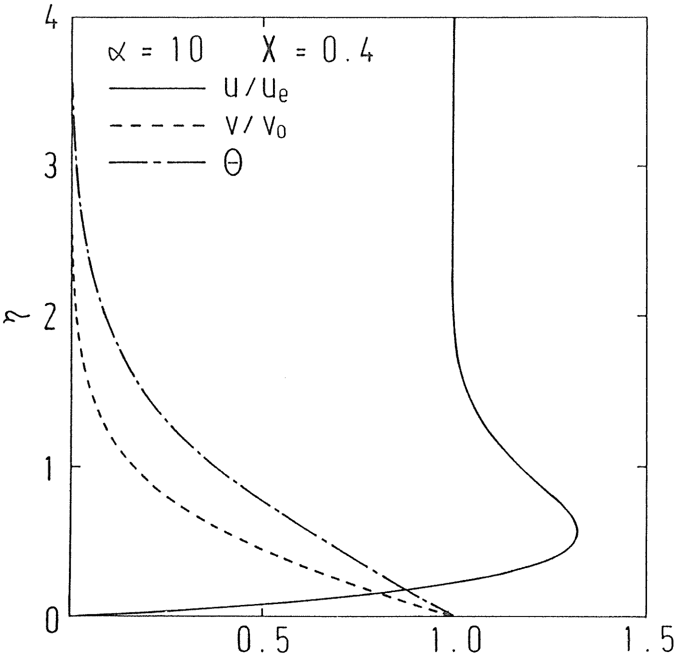


Fig. 3. 15. Profiles of axial, circumferential velocity components and temperature at relatively large rotation speed.

the axial, circumferential wall shear stresses τ_{0x}, τ_{0y} and local Nusselt number

$$Nu = \frac{hx}{k}. \tag{3.18}$$

Here h is the local heat transfer coefficient and k is the thermal conductivity. In the figure, ρ is density, R_x is the Reynolds number defined by $U_m x/\nu$. In Fig. 3.12, τ_{0x} is shown in logarithmic scale. At $\alpha = 0$ and 1, the profiles terminate at the x position where no convergence of the solution is obtained near the separation point as is the case in the two-dimensional boundary layer. As α increases and the effect of rotation grows, the absolute values of τ_{0x}, t_{0y} and Nu become larger and the flow ceases to separate from the wall. Fig. 3.15 shows the velocity and temperature profiles at $\alpha = 10$, $x^* = 0.4$. In comparison with the profile at $\alpha = 4$ in Fig. 3.11, the axial velocity component is further accelerated, while the profiles of the circumferential velocity component and temperature are monotonous.

3. 5. Conclusions

The LISP program is developed, which processes the differential equations and produces a certain FORTRAN code. The equations are formulated by the box method and Newton's method. This part of the FORTRAN code is needed for solving both the ordinary and partial differential equations in one space and/or multiple timelike coordinates. The form of input data to the LISP program is simple and the equations are written almost as they are in the natural expressions. This LISP program avoids tedious and time-consuming work and possible mistakes in symbolic operation by hand.

The boundary layer on a rotating cone in a linearly retarded axial flow is investigated. When the effect of rotation is small, the axial velocity component is decelerated and the boundary layer separates from the wall. As the effect of rotation increases, the axial velocity component is accelerated and the separation ceases to occur. Then the axial, circumferential wall shear stresses and the local heat transfer coefficient become large.

Acknowledgment

The authors wish to express their sincere appreciation to associate Prof. S. Yamashita, Department of Mechanical Engineering, Faculty of Engineering, Gifu University, for his incisive suggestions in performing the research of this chapter.

References of chap. III

3-1) Rice, J.G., Build Program Technique, (1981), John-Wiley & Sons.

3-2) Sugahara, M. and Sigenaga, M., Automatic Programming by Specification Written in Japanese, Information Processing Society of Japan (in Japanese), AI-64-8 (1989).

3-3) Harada, M. ed., Automatic Programming Handbook (in Japanese), (1989), Ohmu Publ.

3-4) Kleinsteuer C. and Patterson M.R., Comput. Meths. Appl. Mech. Engng., Vol. 27 (1981), 1.

3-5) Rosten, H.I., Spalding, D.B. and Tatchell D.G., Engineering Software III, Adey, R. ed., (1983), 639, Springer-Verlag.

3-6) Rice, J.R. and Boisvert, R.F., Solving Elliptic Problems Using ELLPACK, (1985), Springer-Verlag.

3-7) Henry, M.P. and Willmott, A.J., Int. J. Heat Mass Transfer, Vol. 33 No. 4 (1990), 703.

3-8) Garrad, A.D. and Quarton, D.C., J. Sound Vibration, Vol. 109 No. 1 (1986), 65.

3-9) Fletcher, C.A.J. and Armfield, S.W., Com. Appl. Num. Meths., Vol. 4 (1988), 219.

3-10) Luppi, J. and Pajunen, P., J. Compt. Phys., Vol. 88 (1990), 15.

3-11) Rosen, B. and Okabayashi, M., Nuclear Fusion, Vol. 13 (1973), 3.

3-12) Takeda, I. and Itoh, S., JAERI-M 5697C (1974).

3-13) Steinberg, S. and Roache, P.J., J. Comput. Phys., Vol. 57 (1985), 251.

3-14) Cebeci, T. and Bradshaw, P., Momentum Transfer in Boundary Layers, (1977), 213, Hemisphere Publ.

3-15) Choen, C.B. and Reshotko E., NACA Rep. 1293 (1956).

3-16) Nakamura I. Yamashita, S. and Watanabe T., Bull. JSME, Vol. 27 (1984), 1887.

3-17) Nakamura I. Yamashita, S. and Watanabe T., Acta Mechanica, Vol. 57 (1985), 69.

3-18) Koh, J.C.Y. and Price, J.F., Trans. ASME, Ser. C, Vol. 89 (1967), 139.

IV. Automatic Dimensional Analyzing System

4. 1. Introduction

Recently and in the days to come, one of the demands for computers is their ability to assist the intelligent action of human reasoning. Studies in the area in mechanical engineering seem to be focused on CAD systems, autonomy of industrial robots and so on. With the great development of computers and the advance in knowledge engineering, intelligent technology will hopefully be applied to not only design and manufacturing but also to support systems to analyze physical phenomena. From this point of view, the application possibilities of computer's symbolic processing ability in fluid mechanics is examined. In this chapter, a well-defined dimensional analysis is considered and a computer support system is developed.

When the relation between variables governing physical phenomena is not know, a dimensional analysis based on the pi theorem is a very effective method to understand and explain the problems. Dimensional analysis is formulated by the dimensional homogeneity of the terms in governing equations. This method has a sound mathematical background and systematic organization[4-1,4-2]. Recently, advanced methods of dimensional analysis have been proposed[4-3,4-4] and approaches to system formulation have been reported[4-5,4-6].

Dimensional analysis using the pi theorem is written in list processing language LISP and

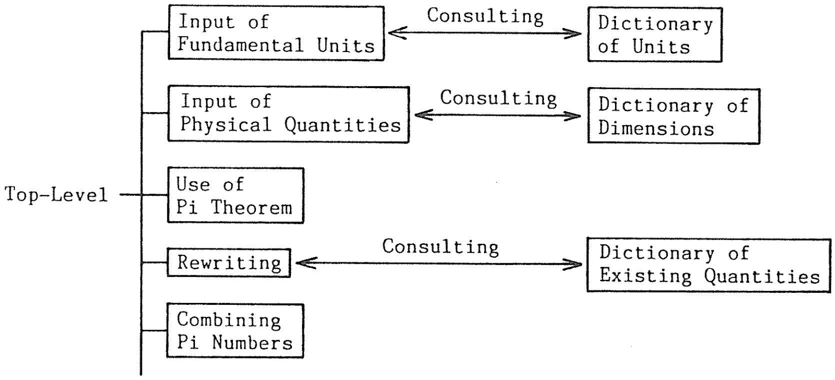


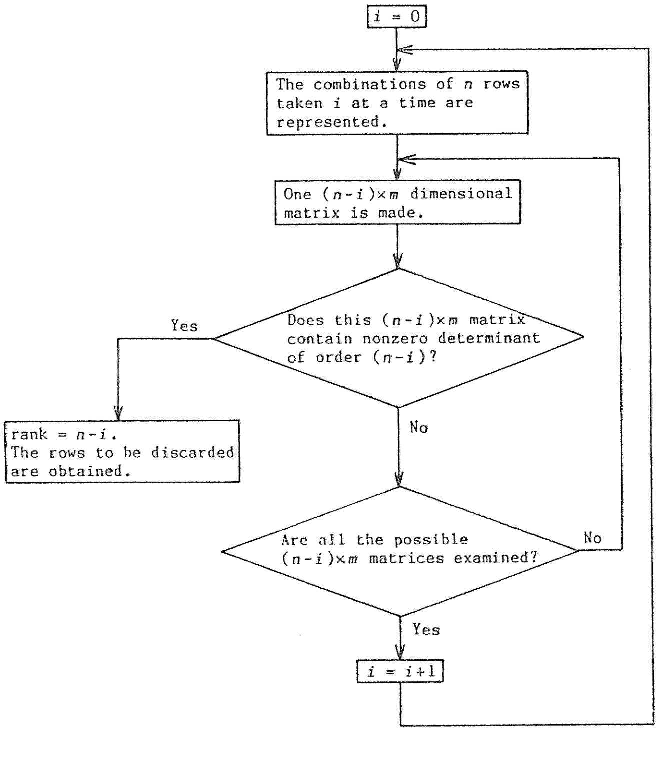
Fig. 4. 1. Architecture of dimensional analyzing system.

a computer aided support system is developed. The overall structure of the system is shown in Fig. 4.1. The analyzing system is composed of functional modules. While it is easy for experts in fluid mechanics to carry out dimensional analysis on fluid mechanical problems, the analysis of problems in thermodynamics seems to be unfamiliar territory. The present support system facilitates this kind of situation. Physical variables used frequently are stored in the data base with their dimensions and when they are used the system consults with data base and refers to their dimensions. This function is useful to prevent input errors. Analyzed non-dimensional variables are rearranged by using the well known variables and suggestions to understand the physical meaning of the results will be offered. Hereinafter, the details of the basic concepts and the architecture of the analyzing system are explained.

4. 2. Formulation and construction

4. 2. 1. Dimensional analysis

Suppose that the physical phenomena are governed by m parameters in any dimensional formula and that k is the maximum number of physical variables which do not constitute nondimensional quantities. Then $m - k$ is the number of nondimensional pi numbers[4-7]. In pi theorem, k corresponds to the rank of the dimensional matrix and does not exceed the number of primary quantities, which is equal to the row number of the dimensional matrix. The rank of dimensional matrix, k, is determined by reducing the row of the dimensional matrix until a nonsingular matrix composed of physical quantities is found[4-8]. This process is shown in Fig. 4.2. First, i which is the number of rows to be reduced, is 0. The combinations



i = 0
The combinations of n rows taken i at a time are represented.
One (n-i)×m dimensional matrix is made.
Does this (n-i)×m matrix contain nonzero determinant of order (n-i)?
Yes
rank = n-i. The rows to be discarded are obtained.
No
Are all the possible (n-i)×m matrices examined?
No
Yes
i = i+1

Fig. 4. 2. Flow chart to determine rank of dimensional matirx.

of n rows taken i at a time are determined and dimensional matrices with $n - i$ rows are constructed. When a nonsingular matrix is found in these matrices, the rank is $n - i$. If all the matrices with $n - i$ rows are singular, i is replaced by $i + 1$ and the same process is repeated. Once the rank and the rows to be extracted are determined, one of the $k \times k$ nonsingular matrices composed by k physical quantities is selected and used to be combined with other quantities into $m - k$ nondimensional forms.

For the same number of physical quantities, the larger the number of the rank, the smaller the number of nondimensional forms and the pi numbers are expected to have more sophisticated meanings. Therefore, the extended pi theorem is formulated, which makes a distinction between the components of vector quantity[4-9]. For example, by introducing a length measured in the vertical direction and a length measured in the horizontal direction, a number of independent dimensions is increased and more useful result is obtained. The present system can analyze vector quantities by this directional method.

4. 2. 2. Registration of physical quantity dimensions

Representative physical quantities and their dimensions are stored in the system data base. The system requires, as an input of governing variables, the meaning of physical quantities and the variable names. Here, the examples of the meaning of physical quantities are velocity and length, while the examples of variable names are the symbols v and l. Physical quantities and their dimensions are placed on the property list of LISP system. One property list of the symbol EXAMPLE is shown in Fig. 4.3. This property list consists of pairs of property names and property values. In Fig. 4.3, the property name P-NAMEI indicates the value P-VALUE1 and the name P-NAME2 denotes the value P-VALUE2, and P-VALUE1 and P-VALUE2 are referred to as P-NAME1 and P-NAME2, respectively. Representative physical quantities and their dimensions are placed on the property list of the symbol DICTIONARY. Fig. 4.4 shows this list in primary units of length, mass and time. The meaning of the physical quantity VELOCITY is a property name and its value is (1 0 -1), which is the dimensions of the velocity. When the user designates the variable U as velocity, the system registers the meaning of U and its dimensions (1 0 -1).

The meanings of variables are placed on the property list of symbol MEANING shown in Fig. 4.5. The list of all the used meanings of physical quantities is referred to by the property name REF and the list of registered velocity U and V is referred to by VELOCITY.

The prescribed dimensions of quantities are formulated in the primary units of length, mass, time and temperature. The dimensions in other primary units are converted from these units. In case the dimensions of an input variable are unknown, its dimensions must be specified in some primary units. Therefore, the system is designed to enable the user to select primary units.

4. 2. 3. Transformation of nondimensional quantities

There are many quantities which are normally employed and known to represent physical relations between variables appearing in phenomena. For example, the Reynolds number is the relation between inertia forces and viscous forces and the kinematic viscosity is the ratio of viscosity to density. The transformation of nondimensional parameters obtained by the pi theorem and their arbitrary combination into some well known quantities helps one understand the phenomena and find more meaningful forms.

In this system, some of the well known quantities are prepared on the property list of a symbol EXISTING shown in Fig. 4.6. In this figure, the property name RE is the abbreviation of the Reynolds number and its property value denotes that the physical meaning of

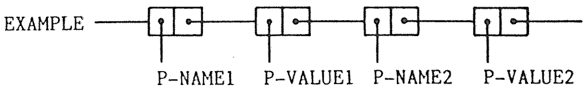


Fig. 4. 3. Property list.

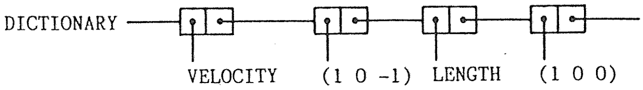


Fig. 4. 4. Physical quantities and its dimensions in data base.

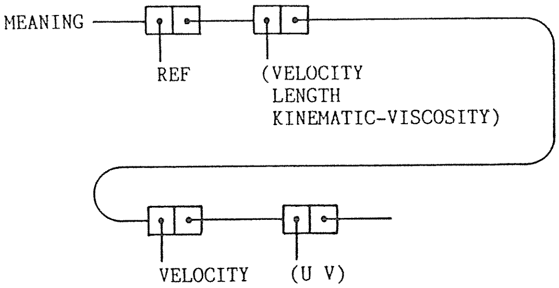


Fig. 4. 5. Meanings of variables stored in property list.

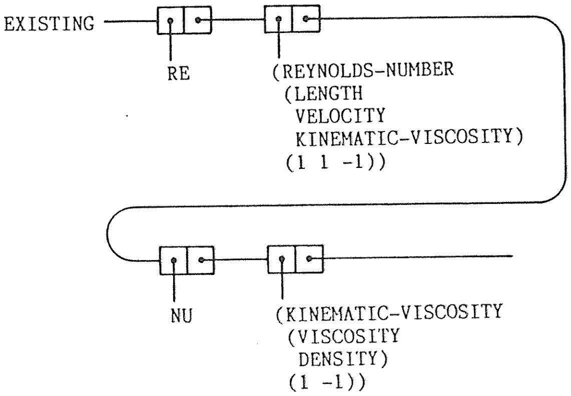
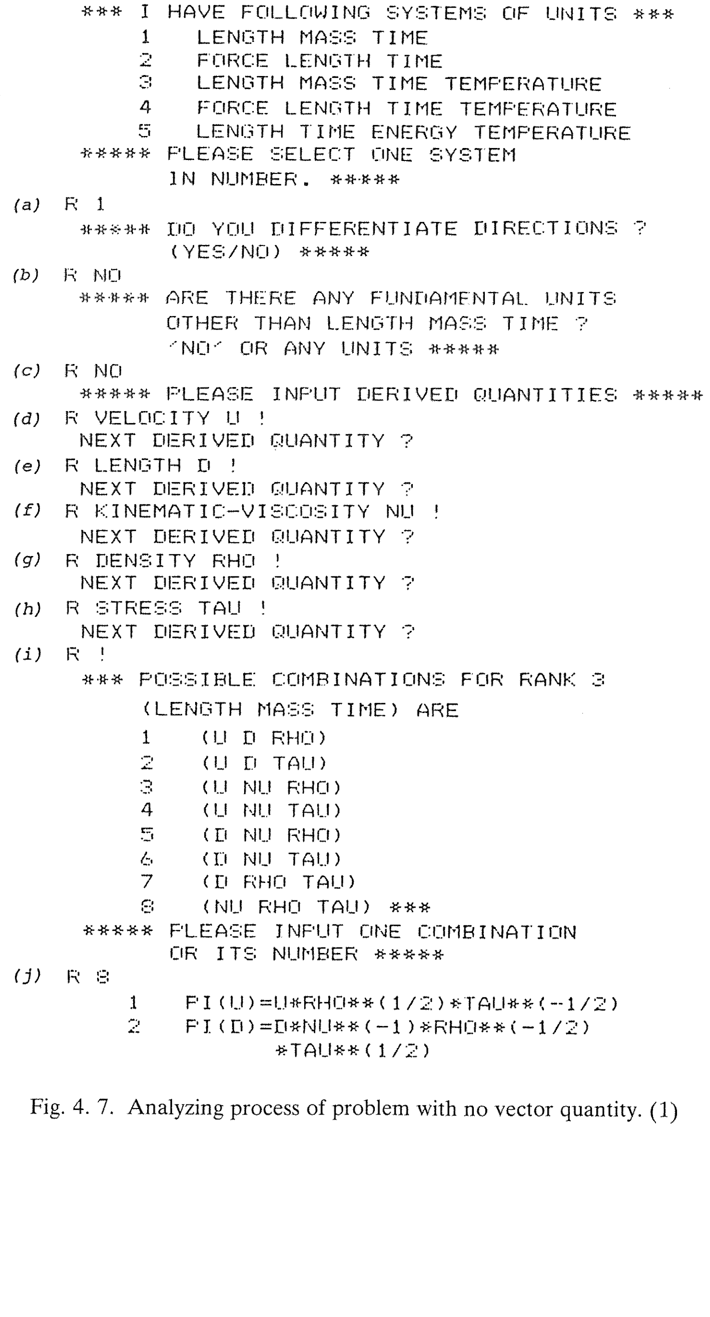


Fig. 4. 6. Known nondimensional quantities in data base.

variable RE is the Reynolds number which is the product of length, velocity and kinematic viscosity to the minus one power. The meanings of physical quantities composing pi numbers are stored. When the combinations of meanings of physical quantities in pi numbers include the meanings of known quantities on the property list of a symbol EXISTING, the transformation is suggested. To be concrete, while it is known that the kinematic viscosity consists of viscosity and density, an obtained pi number contains the meanings of density, viscosity and others, the transformation using the kinematic viscosity is offered. Even though the transformation into the kinematic viscosity is carried out, the kinematic viscosity may be used to transform into other nondimensional quantities.

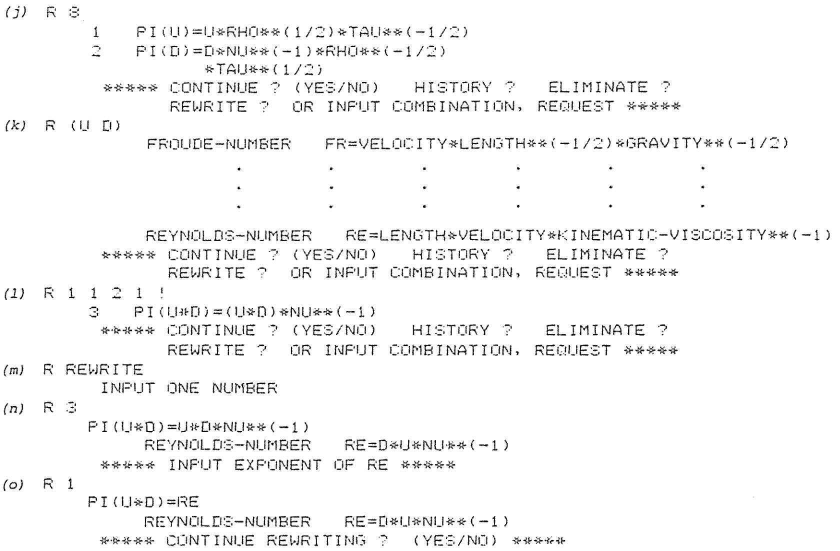
4. 3. Examples

Let the system analyze the fully developed flow in a circular pipe. The following quantities may be concerned in this flow: velocity U, diameter of pipe D, kinematic viscosity ν, density ρ and wall shear stress τ. Fig. 4.7 and 4.8 shows the analyzing process of this problem.



```
         *** I HAVE FOLLOWING SYSTEMS OF UNITS ***
             1   LENGTH MASS TIME
             2   FORCE LENGTH TIME
             3   LENGTH MASS TIME TEMPERATURE
             4   FORCE LENGTH TIME TEMPERATURE
             5   LENGTH TIME ENERGY TEMPERATURE
         ***** PLEASE SELECT ONE SYSTEM
               IN NUMBER. *****
(a)     R 1
         ***** DO YOU DIFFERENTIATE DIRECTIONS ?
               (YES/NO) *****
(b)     R NO
         ***** ARE THERE ANY FUNDAMENTAL UNITS
               OTHER THAN LENGTH MASS TIME ?
               'NO' OR ANY UNITS *****
(c)     R NO
         ***** PLEASE INPUT DERIVED QUANTITIES *****
(d)     R VELOCITY U !
          NEXT DERIVED QUANTITY ?
(e)     R LENGTH D !
          NEXT DERIVED QUANTITY ?
(f)     R KINEMATIC-VISCOSITY NU !
          NEXT DERIVED QUANTITY ?
(g)     R DENSITY RHO !
          NEXT DERIVED QUANTITY ?
(h)     R STRESS TAU !
          NEXT DERIVED QUANTITY ?
(i)     R !
         *** POSSIBLE COMBINATIONS FOR RANK 3
             (LENGTH MASS TIME) ARE
             1    (U D RHO)
             2    (U D TAU)
             3    (U NU RHO)
             4    (U NU TAU)
             5    (D NU RHO)
             6    (D NU TAU)
             7    (D RHO TAU)
             8    (NU RHO TAU) ***
         ***** PLEASE INPUT ONE COMBINATION
               OR ITS NUMBER *****
(j)     R 8
            1    PI(U)=U*RHO**(1/2)*TAU**(-1/2)
            2    PI(D)=D*NU**(-1)*RHO**(-1/2)
                       *TAU**(1/2)
```

Fig. 4. 7. Analyzing process of problem with no vector quantity. (1)



```
(j)  R 8
        1    PI(U)=U*RHO**(1/2)*TAU**(-1/2)
        2    PI(D)=D*NU**(-1)*RHO**(-1/2)
                   *TAU**(1/2)
         ***** CONTINUE ? (YES/NO)   HISTORY ?   ELIMINATE ?
                REWRITE ?  OR INPUT COMBINATION, REQUEST *****
(k)  R (U D)
              FROUDE-NUMBER   FR=VELOCITY*LENGTH**(-1/2)*GRAVITY**(-1/2)
                     .         .         .         .         .         .
                     .         .         .         .         .         .
                     .         .         .         .         .         .

              REYNOLDS-NUMBER   RE=LENGTH*VELOCITY*KINEMATIC-VISCOSITY**(-1)
         ***** CONTINUE ? (YES/NO)   HISTORY ?   ELIMINATE ?
                REWRITE ?  OR INPUT COMBINATION, REQUEST *****
(l)  R 1 1 2 1 !
        3    PI(U*D)=(U*D)*NU**(-1)
         ***** CONTINUE ? (YES/NO)   HISTORY ?   ELIMINATE ?
                REWRITE ?  OR INPUT COMBINATION, REQUEST *****
(m)  R REWRITE
         INPUT ONE NUMBER
(n)  R 3
        PI(U*D)=U*D*NU**(-1)
              REYNOLDS-NUMBER   RE=D*U*NU**(-1)
         ***** INPUT EXPONENT OF RE *****
(o)  R 1
        PI(U*D)=RE
              REYNOLDS-NUMBER   RE=D*U*NU**(-1)
         ***** CONTINUE REWRITING ?  (YES/NO) *****
```

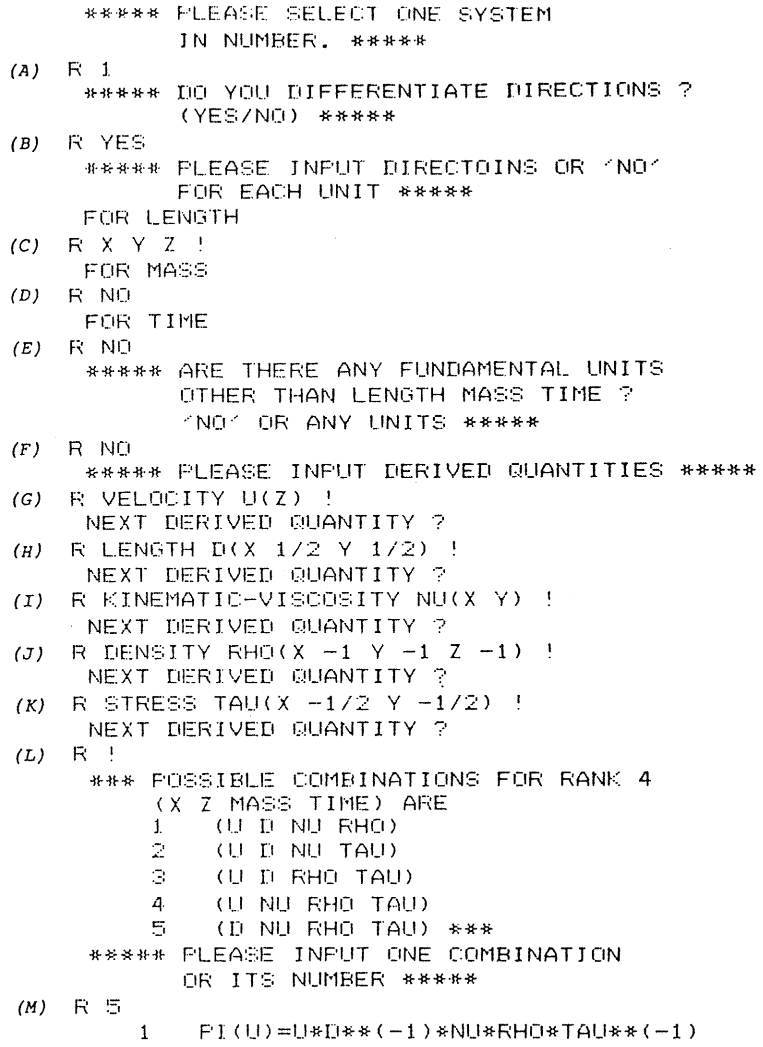
Fig. 4. 8. Analyzing process of problem with no vector quantity. (2)

In these figures and following ones, the italics in parentheses (*a*), (*A*), (*b*), (*B*) and so on denote input lines by the user and character R at these lines are input prompt of the LISP system. In Fig. 4.7, length, mass and time are selected as the primary units. Since any vector quantity does not appear and since there is no primary unit except those selected at input (*a*), replies NO are put at lines (*b*) and (*c*), respectively. Lines from (*d*) to (*i*) denote the inputs of governing quantities of this problem. At each line, the meaning of physical quantity and its variable name are presented and an exclamation mark denotes a terminator of input. Velocity U is presented at line (*d*), and length D, kinematic viscosity ν, density ρ and shear stress τ are put at lines (*e*), (*f*), (*g*) and (*h*), respectively. The input of terminator ! at line (*i*) denotes that all the physical quantities are specified. Then the system determines the rank of the dimensional matrix and shows all the possible combinations of quantities, which can be used to form nondimensional forms of other quantities. At line (*j*), eighth combination is selected to obtain nondimensional forms other than kinematic viscosity, density and shear stress.

Fig. 4.8 shows the continuation of this process. Two pi numbers of velocity U and diameter D are obtained. By the input at line (*k*), stored known quantities composed of velocity and length are presented. It can be recognized that the Reynolds number contains the product of the length and the velocity. At line (*l*) it is indicated to make a nondimensional product of the first power of the first pi number PI(U) and the first power of the second pi number PI(D). Then the third nondimensional form of quantity $U \times D$ is obtained. This third pi number can be transformed into the Reynolds number by the process at lines (*m*), (*n*) and (*o*).

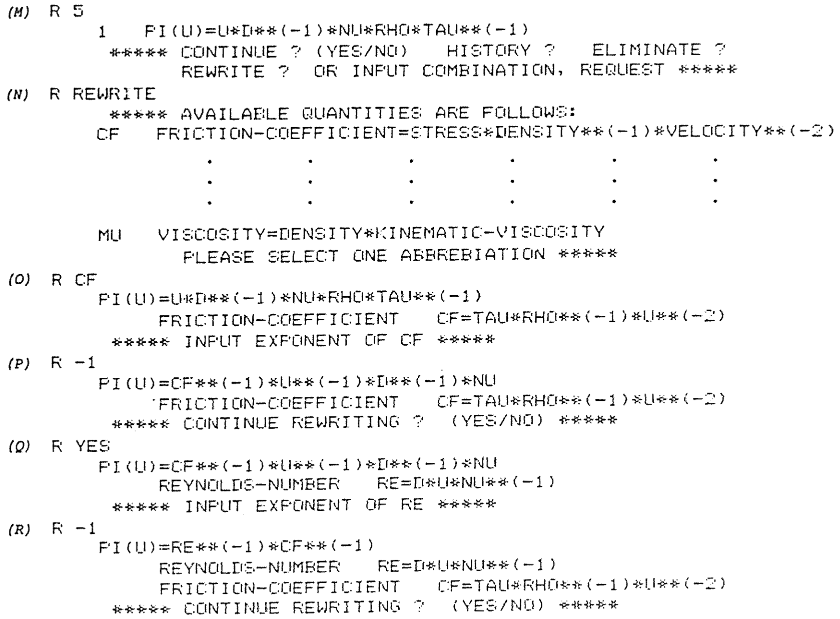
Fig. 4.9 and its continuation Fig. 4.10 shows the process of the problem which contains a vector quantity. To begin the vector analyzing, YES is put at line (*B*). Input (*C*) directs to distinguish between lengths measured in the x, y and z directions. In this problem, it is assumed that z is the axial direction and x and y directions are perpendicular to the axial direction. The dimensions of physical quantities are determined by the suggestion of Huntley[4-9].

Since the flow direction coincides with the axial direction, the primary length unit of the velocity is the length in the z direction. Because of the symmetry of the pipe about its axis the dimension of the diameter is written as the square root of x times the square root of y. For the same reason, the dimensions of the kinematic viscosity, density and shear stress are determined. The inputs of these quantities are at lines from (G) to (K). The meanings of the inputs are obvious. Fig. 4.10 shows the resultant pi number and its transformation. It can be concluded that the friction coefficient is inverse proportional to the Reynolds number.



```
          ***** PLEASE SELECT ONE SYSTEM
                 IN NUMBER. *****
(A)     R 1
          ***** DO YOU DIFFERENTIATE DIRECTIONS ?
                 (YES/NO) *****
(B)     R YES
          ***** PLEASE INPUT DIRECTOINS OR 'NO'
                 FOR EACH UNIT *****
          FOR LENGTH
(C)     R X Y Z !
          FOR MASS
(D)     R NO
          FOR TIME
(E)     R NO
          ***** ARE THERE ANY FUNDAMENTAL UNITS
                 OTHER THAN LENGTH MASS TIME ?
                 'NO' OR ANY UNITS *****
(F)     R NO
          ***** PLEASE INPUT DERIVED QUANTITIES *****
(G)     R VELOCITY U(Z) !
          NEXT DERIVED QUANTITY ?
(H)     R LENGTH D(X 1/2 Y 1/2) !
          NEXT DERIVED QUANTITY ?
(I)     R KINEMATIC-VISCOSITY NU(X Y) !
          NEXT DERIVED QUANTITY ?
(J)     R DENSITY RHO(X -1 Y -1 Z -1) !
          NEXT DERIVED QUANTITY ?
(K)     R STRESS TAU(X -1/2 Y -1/2) !
          NEXT DERIVED QUANTITY ?
(L)     R !
          *** POSSIBLE COMBINATIONS FOR RANK 4
               (X Z MASS TIME) ARE
               1   (U D NU RHO)
               2   (U D NU TAU)
               3   (U D RHO TAU)
               4   (U NU RHO TAU)
               5   (D NU RHO TAU) ***
          ***** PLEASE INPUT ONE COMBINATION
                 OR ITS NUMBER *****
(M)     R 5
             1    PI(U)=U*D**(-1)*NU*RHO*TAU**(-1)
```

Fig. 4. 9. Analyzing process of problem with a vector quantity. (1)



```
(M)  R 5
        1    PI(U)=U*D**(-1)*NU*RHO*TAU**(-1)
         ***** CONTINUE ? (YES/NO)    HISTORY ?    ELIMINATE ?
                REWRITE ?  OR INPUT COMBINATION, REQUEST *****
(N)  R REWRITE
         ***** AVAILABLE QUANTITIES ARE FOLLOWS:
        CF   FRICTION-COEFFICIENT=STRESS*DENSITY**(-1)*VELOCITY**(-2)
                  .       .       .       .       .       .
                  .       .       .       .       .       .
                  .       .       .       .       .       .

        MU   VISCOSITY=DENSITY*KINEMATIC-VISCOSITY
                PLEASE SELECT ONE ABBREBIATION *****
(O)  R CF
        PI(U)=U*D**(-1)*NU*RHO*TAU**(-1)
             FRICTION-COEFFICIENT    CF=TAU*RHO**(-1)*U**(-2)
         ***** INPUT EXPONENT OF CF *****
(P)  R -1
        PI(U)=CF**(-1)*U**(-1)*D**(-1)*NU
             FRICTION-COEFFICIENT    CF=TAU*RHO**(-1)*U**(-2)
         ***** CONTINUE REWRITING ?  (YES/NO) *****
(Q)  R YES
        PI(U)=CF**(-1)*U**(-1)*D**(-1)*NU
             REYNOLDS-NUMBER    RE=D*U*NU**(-1)
         ***** INPUT EXPONENT OF RE *****
(R)  R -1
        PI(U)=RE**(-1)*CF**(-1)
             REYNOLDS-NUMBER    RE=D*U*NU**(-1)
             FRICTION-COEFFICIENT    CF=TAU*RHO**(-1)*U**(-2)
         ***** CONTINUE REWRITING ?  (YES/NO) *****
```

Fig. 4. 10. Analyzing process of problem with a vector quantity. (2)

4. 4. Conclusions

The system written in LISP is developed, which is convenient to analyze the phenomena by using the pi theorem. It is demonstrated that even the simple inputs can deduce useful results. This system suggests the possibilities of transformation of nondimensional quantities obtained by the pi theorem into other forms within known physical quantities and helps the user to understand the phenomena.

Dimensional analysis based on the pi theorem is systematically formulated, and it can be implemented in the present study. There are other methods to find pi numbers governing the phenomena; the method which tries to find scale factors in the problems[4-10], and the method which analyzes based on the governing equations and their boundary conditions[4-11]. To implement these methods, their procedure and algorithms in the human process must be specified in some ways and various data need to be readjusted. The development in this area will be an excellent application of knowledge engineering and contribute greatly to it.

References of chap. IV

4-1) Bridgeman, P.W., Dimensional Analysis, (1921), Harvard Univ. Press.

4-2) Birkhoff, G., Hydrodynamics, (1950), 77, Princeton Univ. Press.

4-3) Muciek, A. and Szata, M., Bull. De L'academine Polinaise des Sciences, Vol. 24 No. 7-8 (1981), 97.

4-4) Pethö, A. and Kumar, S., Int. J. Heat Mass Transfer, Vol. 29 No. 1 (1986), 157.

4-5) Murota, K., Japan J. Appl. Math., Vol. 2 (1985), 471.
4-6) Acezél J., J. Math. Anal. Appl., Vol. 119 (1986), 389.
4-7) Van Driest, E.R., Trans. ASME, J. Appl. Mech., Vol. 13 (1946), A-34.
4-8) Langhaar, H.L., Dimensional Analysis and Theory of Models, (1951), 37, John Wiley & Sons, Inc.
4-9) Huntley, H.E., Dimensional Analysis, (1952), MacDonald & Co.
4-10) Schuring, D.J., Scale Models in Engineering, (1977), Pergamon Press.
4-11) Kline, S.J., Similitude and Approximation Theory, (1965), McGraw-Hill.

V. Mechanism for Solving Fluid Property Problems

5. 1. Introduction

There are many domain specific systems which are devised to carry out human intelligent activities. Some of them seem to have more practical ability than experts' [5-1]. The researches in knowledge engineering are carried out mainly in the areas of diagnosis, design, manufacturing and control engineering[5-2].

In this chapter, fluid mechanics are considered as an object of study, and the process by which students use to solve textbook problems is simulated. Applications of knowledge engineering in this area are still few and this study is intended to be a contribution in this regard. As a practical use of symbolic manipulation processing of computers, a dimensional analysis system has been coded and its utility has been shown by the present authors[5-3]. The present automatic problem solver is constructed by the simulation of the human thought process. This study is expected to clarify the knowledge structures and the solving mechanism in this area. It also would seem to advance the systematic methods of teaching.

After a pioneer system was reported to solve the mathematical problems[5-4], solving systems which accepts primary statics and arithmetic problems were developed[5-5)~5-8]. Meanwhile, the studies to apply the solving systems to computer-aided instruction have been progressing[5-9)~5-11].

In these situations, resolution of fluid property problems is the first step for an ambitious research plan. These problems are well-posed ones based on full of experiences and emerge on the first pages of most textbooks. In what follows, we present details on this problem solving system.

5. 2. Basic method to solve fluid property problems

Consider the simple problem of obtaining a specific volume of fluid, in which we follow the process of a student in solving this problem and then try to create some human models. Here, the proposed model is one of the examples and it does not mean a universal model. It is assumed that students already knows the physical meaning of technical terms and symbols appearing in problems and have the set of concepts to understand problems. Confronted with a problem, students have some representation about the problem: given quantities, wanted quantities, constraints and so on. This makes a set whose elements are various kinds of understood concepts. If the specific volume is given in the statement of a problem, the problem can be solved. If the specific volume v is not given, it may be obtained from the density ρ by using the expression

$$\rho=\frac{1}{v}\,. \tag{5.1}$$

If the density is given, the problem can be solved. If the density is not known, a subproblem to obtain it is set up and the resolution of this subproblem is pursued. When the density can not be determined, it is impossible to obtain the specific volume through the expression (5.1). On the other hand, the relation between specific volume v, pressure p, general gas constant R, molecular weight M and absolute temperature T is known to be

$$pv = \frac{RT}{M}, \tag{5.2}$$

and the problem of obtaining the specific volume is divided into subproblems to find pressure, gas constant, molecular weight and absolute temperature. In this way, problems are transformed into subproblems by using relations between quantities. In case all the solutions to subproblem is found, the original problems are solved. However, if one fails to solve the subproblems, other expressions and/or methods must be found. In other words, in the process to solve this problem, search strategies are needed to determine whether solutions can be found through the expression (5.1) or the relation (5.2) must be taken into account.

In the example above, a subproblem to obtain density is formulated by the expression (5.1). It is generally thought that density is a characteristic of a fluid and that it can be known by consulting data books and textbooks as a reference[5-12]. Therefore, the properties of matter can be formulated in a system data base. Besides the density, the molecular weight and specific volume are properties also.

One of the subproblems developed by the expression (5.2) contains the absolute temperature. When the absolute temperature is not given, the search for it is continued. Even if one fails to obtain the absolute temperature, it may be assumed that the Celsius temperature θ is 15 centigrade. And by using the expression

$$T = \theta + 273.15, \tag{5.3}$$

the absolute temperature may be obtained. In this case, the assumption of the Celsius temperature is one of the well used methods. A similar method may be adopted to obtain pressure in the standard condition. These kinds of methods are defined as general decisions.

Consider the subproblem of finding the gas constant. The gas constant and acceleration of gravity are experimental constants which are difficult to determine experimentally in the solving process. These subjects are beyond the scope of this chapter. In general, experimental constants are not clearly denoted in problem sentences. When they cannot be found by any method, it is common for standard values to be determined by considering the conditions of problem. In problem solving, this default knowledge cannot be disregarded.

The general decision and default knowledge must be used only in case the quantities are not given or cannot be obtained. Therefore, we must determine when such knowledge is to be used.

Expressions (5.1) and (5.2) are used not only to obtain the specific volume. Expression (5.1) may be used in the problems to find the density. These expressions represent dependencies of quantities and can be utilized in various situations. However, it is a mattter of course that the state of equation of gas (5.2) is effective only when the fluid is liquid.

Problem solving involves many relations between physical quantities, and these quantities are expressed in various units. Thus, it is necessary to standardize units of quantities to apply expressions and solution methods.

A rough outline of the process for solving a fluid property problem is shown in Fig. 5.1.

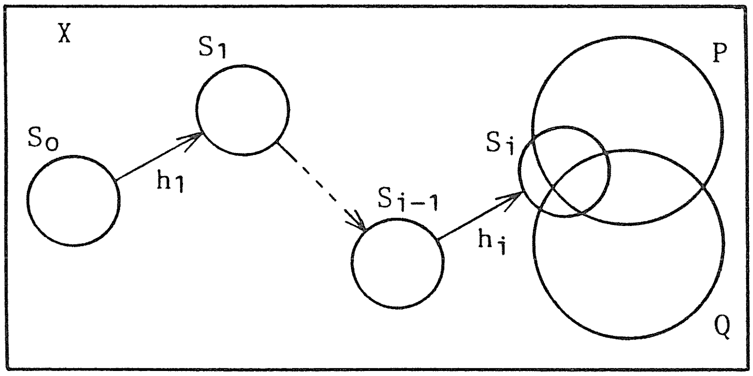


Fig. 5. 1. Rough sketch of solving process using relations.

Space X is a set of unknowns and data in problems. Each of the quantities to be obtained in a problem makes set in X, whose element is the unknown itself. Consider one of these sets, S_o. In Fig. 5.1, P is a set of given quantities in problem and Q is quantities prescribed by the general decision and default knowledge. It is difficult to clarify the ways to define set Q and it seems that experienced teachers have more excellent ability for it than students. From set S_o, transformation h_i is applied to the set of unknowns, and a set of unknowns in subproblems is formed. In this figure, if all the quantities in S_i are known, the quantities in S_{i-1} can be obtained. The transformation which uses quantities replaced by previous transformations is never applied. When all elements in set S_i are included in set P and/or Q, the problem can be solved by a general decision, default knowledge and given values. On the other hand, if no transformation is found to make S_i contained in P and/or Q, it is concluded that the problem cannot be solved.

In problem solving, transformations, general decision and default knowledge depend on the characteristics of the problem areas. In terms of the above considerations, a solving system is coded.

5. 3. System architecture

The prototype system has the architecture shown in Fig. 5.2 and it is written in Interlisp-D. In the Figure, the main loop controls the overall processing.

Syntactic and semantic analysis are applied to the input sentences. Inner expression in system is formed and the working memory is renewed. For convenience, inputs are expressed in Japanese sentences. An example of an input sentence is shown in Fig. 5.3 and it is translated into the inner expressions shown in Fig. 5.4 (a), (b). The inner expression in Fig. 5.4 (a) has the same meanings as those in Fig. 5.4 (b). The consistency of the two expressions is maintained in processing and both of them are used properly. In Fig. 5.4, CONCEPT means object in the problem, GOAL means wanted value and DATA means given value. When the unit is omitted in the input sentence, a default unit system, whose primary units are meter, kilogram, second and absolute temperature, is adopted. In Fig. 5.3, an explicit expression <bar> means that the pressure is 1.0 bar. Problem solving is carried out by expressing the current state in working memory. When the input sentence contains no unknown and is not regarded as a problem, replacement of the working memory, advisory system or other subsystems runs according to the meaning of input.

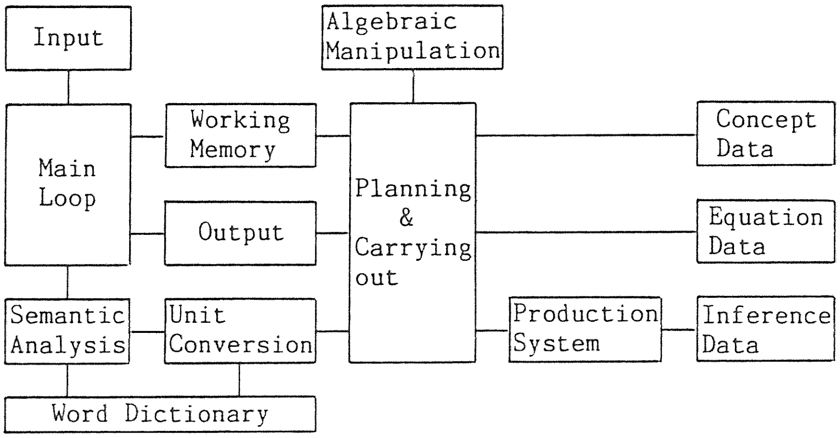
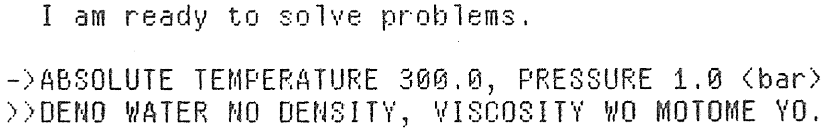
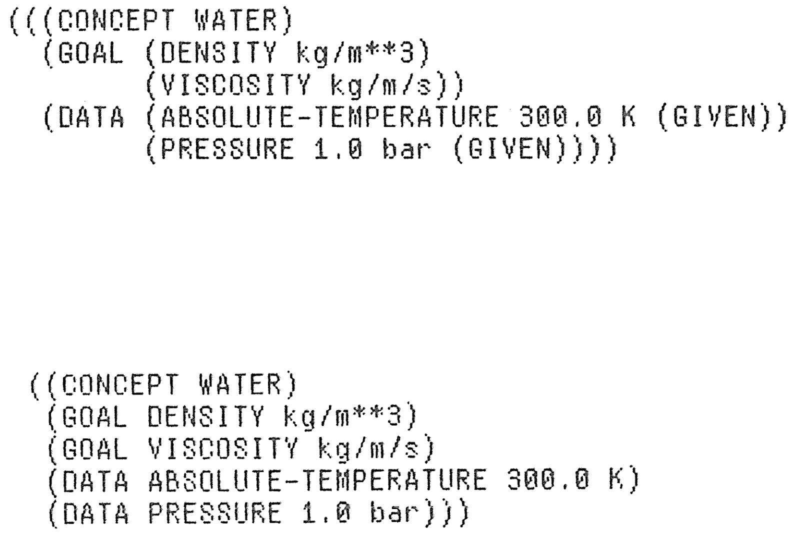


Fig. 5. 2. Architecture of the solving system.



```
  I am ready to solve problems.

->ABSOLUTE TEMPERATURE 300.0, PRESSURE 1.0 <bar>
>>DENO WATER NO DENSITY, VISCOSITY WO MOTOME YO.
```

Fig. 5. 3. Input of problem sentence.



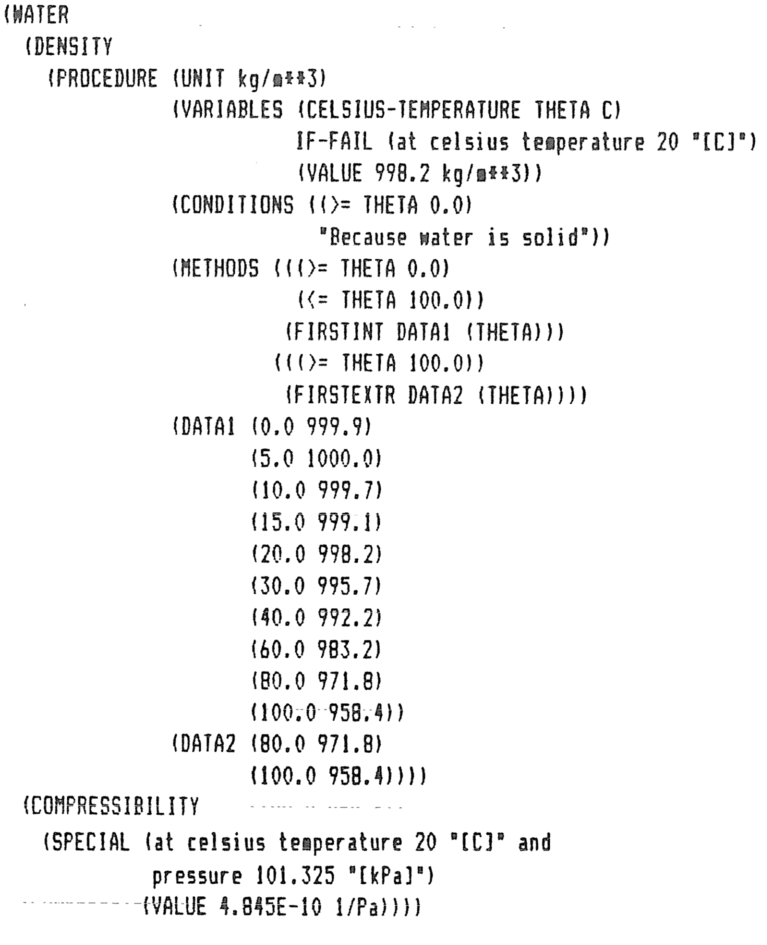
```
(((CONCEPT WATER)
  (GOAL (DENSITY kg/m**3)
        (VISCOSITY kg/m/s))
  (DATA (ABSOLUTE-TEMPERATURE 300.0 K (GIVEN))
        (PRESSURE 1.0 bar (GIVEN))))
```

```
((CONCEPT WATER)
 (GOAL DENSITY kg/m**3)
 (GOAL VISCOSITY kg/m/s)
 (DATA ABSOLUTE-TEMPERATURE 300.0 K)
 (DATA PRESSURE 1.0 bar))
```

Fig. 5. 4. Inner expressions of the input in Fig. 5.3.

Given a problem, plans to solve the problem are formed. Following the process explained in Sec. 5.2, paths from unknowns to knowns are searched by formulating subproblems. In case this search process fails, it is concluded that the problem is not resolvable. When the solution plan is formed, a practical calculation is made by following the plan and values of unknowns are obtained. Since the planning system and calculator are independent of the data base of expressions, default decision and default knowledge, a wide variety of problems can be handled by data supplements.

The data base of characteristics of the objects is named Concept Data, the data base of relations between physical quantities is named Equation Data and the data base for general decision and default knowledge are called Inference Data. Formation of solution plans and calculation are processed according to these data. Fig. 5.5 shows a part of Concept Data, which contains the density and the compressibility of water. In the expression of density, PROCEDURE denotes that the density is evaluated by some procedure, UNIT denotes the unit of obtained density, VARIABLES denotes requisite variables used in this procedure, CONDITIONS denotes the preconditions to apply this procedure. METHOD is the main



```
(WATER
  (DENSITY
    (PROCEDURE (UNIT kg/m**3)
               (VARIABLES (CELSIUS-TEMPERATURE THETA C)
                          IF-FAIL (at celsius temperature 20 "[C]")
                          (VALUE 998.2 kg/m**3))
               (CONDITIONS ((>= THETA 0.0)
                            "Because water is solid"))
               (METHODS (((>= THETA 0.0)
                          (<= THETA 100.0))
                         (FIRSTINT DATA1 (THETA)))
                        (((>= THETA 100.0))
                         (FIRSTEXTR DATA2 (THETA))))
               (DATA1 (0.0 999.9)
                      (5.0 1000.0)
                      (10.0 999.7)
                      (15.0 999.1)
                      (20.0 998.2)
                      (30.0 995.7)
                      (40.0 992.2)
                      (60.0 983.2)
                      (80.0 971.8)
                      (100.0 958.4))
               (DATA2 (80.0 971.8)
                      (100.0 958.4))))
 (COMPRESSIBILITY
   (SPECIAL (at celsius temperature 20 "[C]" and
             pressure 101.325 "[kPa]")
            (VALUE 4.845E-10 1/Pa))))
```

Fig. 5. 5. Characteristic data accompanied with water. (Concept Data) part of the procedure and DATA1 and DATA2 are data used in METHOD. In the VARIABLES list, the sentences following IF-FAIL indicate that, in case of failure to find the Celsius temperature, the density is assumed to be 998.2 kg/m^3 at 20°C. This kind of assumption is regarded as a general decision. In the figure, the compressibility has a value in a special condition and is also evaluated by the general decision.

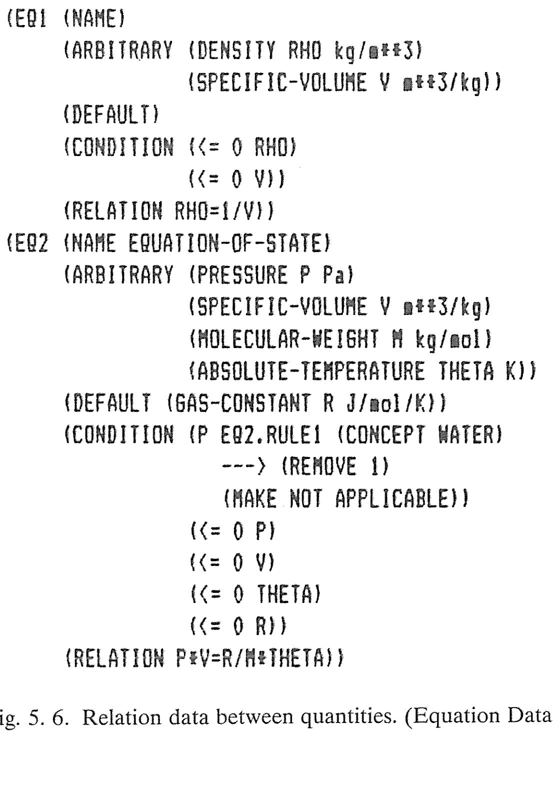
Equation Data is shown in Fig. 5.6. In the description of a relation, a distinction is made between variables obtainable by default knowledge or by other arbitrary variables. In the solution process, relations are used when one of the arbitrary variables is a variable to be obtained. The equation is transformed by the algebraic manipulation and the expression of one arbitrary variable is determined. In the equation EQ1, density and specific volume are the arbitrary variables. The list of CONDITION denotes the conditions to validate EQ1. The expression EQ1 is effective when the density ρ and the specific volume v are not less than 0. RELATION denotes the expression itself. Expression EQ2 is the equation of state. In the CONDITION list,

(P EQ1.RULE1 (CONCEPT WATER)

$->$ (REMOVE 1) (MAKE NOT APPLICABLE))

is a production rule written in the subset system of OPS5[5-13)]. The meaning of this rule is as follows: if the object considered at present is water and there is an element

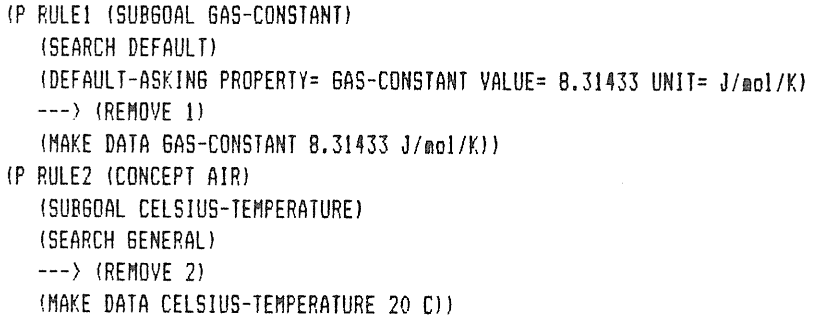
(CONCEPT WATER)



```
(EQ1 (NAME)
     (ARBITRARY (DENSITY RHO kg/m**3)
                (SPECIFIC-VOLUME V m**3/kg))
     (DEFAULT)
     (CONDITION ((<= 0 RHO)
                ((<= 0 V))
     (RELATION RHO=1/V))
(EQ2 (NAME EQUATION-OF-STATE)
     (ARBITRARY (PRESSURE P Pa)
                (SPECIFIC-VOLUME V m**3/kg)
                (MOLECULAR-WEIGHT M kg/mol)
                (ABSOLUTE-TEMPERATURE THETA K))
     (DEFAULT (GAS-CONSTANT R J/mol/K))
     (CONDITION (P EQ2.RULE1 (CONCEPT WATER)
                   ---> (REMOVE 1)
                   (MAKE NOT APPLICABLE))
                ((<= 0 P)
                ((<= 0 V)
                ((<= 0 THETA)
                ((<= 0 R))
     (RELATION P*V=R/M*THETA))
```

Fig. 5. 6. Relation data between quantities. (Equation Data) in the working memory, it is concluded that this relation is not applicable.

Fig. 5.7 gives examples of Inference Data. Inference Data are expressed in production rules. As the path search proceeds, the unknowns in subproblems denoted by SUBGOAL are set in the working memory. When a solution cannot be obtained by knowns in the problem, default knowledge and general decision are used. RULE1 asserts that when the unknown of a current subproblem is the gas constant and the default knowledge can be used, if the user permits to determine that the gas constant is 8.31433 J/mol/K, then a new element



```
(P RULE1 (SUBGOAL GAS-CONSTANT)
   (SEARCH DEFAULT)
   (DEFAULT-ASKING PROPERTY= GAS-CONSTANT VALUE= 8.31433 UNIT= J/mol/K)
   ---> (REMOVE 1)
   (MAKE DATA GAS-CONSTANT 8.31433 J/mol/K))
(P RULE2 (CONCEPT AIR)
   (SUBGOAL CELSIUS-TEMPERATURE)
   (SEARCH GENERAL)
   ---> (REMOVE 2)
   (MAKE DATA CELSIUS-TEMPERATURE 20 C))
```

Fig. 5. 7. Default decision and general knowledge. (Inference Data)

(DATA GAS-CONSTANT 8.31433 J/mol/K)

is added to the working memory. In RULE1, DEFAULT-ASKING is a Lisp funciton which inquires of the user if the default value is adopted.

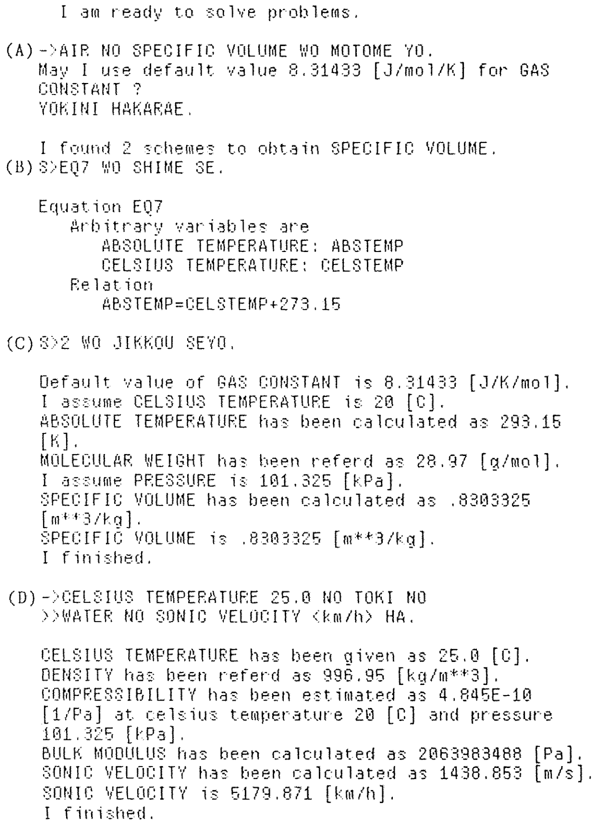
There are some levels in the path searching process. Initially, one tries to find a solution only by using data in the problem. If the solution cannot be found, values that may be omitted are supplied. In this system, the levels are as follows: first, knowns in the original problem are used, then the default knowledge and finally the general knowledge are applied.

5. 4. *Examples*

Consider a problem in which the unknown is the value of specific volume of air. Fig. 5.8 shows the interaction process in this problem solving. Since no known value appears in the input sentence of the problem at line (A), a system inquiry about the usage of default value is made in consequence of the production rules. The planning process is continued by permitting to use all of the default decision data and general knowledge data, and the two methods shown in Fig. 5.9 are obtained. A second method is selected at line (C) and the specific volume is determined. At line (D), it is commanded to find the sonic velocity, in unit km/h, of water at Celsius temperature 25.0°C. It is shown that the density and the compressibility are evaluated and the sonic velocity is obtained.

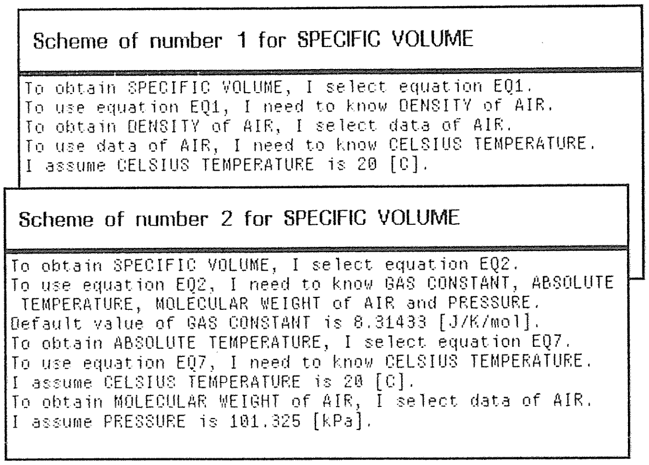
5. 5. *Conclusions*

Through application of knowledge engineering to fluid mechanics, fluid property problems are considered and a system which simulates the human solving process is developed.



```
     I am ready to solve problems.

(A)->AIR NO SPECIFIC VOLUME WO MOTOME YO.
   May I use default value 8.31433 [J/mol/K] for GAS
   CONSTANT ?
   YOKINI HAKARAE.

   I found 2 schemes to obtain SPECIFIC VOLUME.
(B)S>EQ7 WO SHIME SE.

   Equation EQ7
      Arbitrary variables are
         ABSOLUTE TEMPERATURE: ABSTEMP
         CELSIUS TEMPERATURE: CELSTEMP
      Relation
         ABSTEMP=CELSTEMP+273.15

(C)S>2 WO JIKKOU SEYO.

   Default value of GAS CONSTANT is 8.31433 [J/K/mol].
   I assume CELSIUS TEMPERATURE is 20 [C].
   ABSOLUTE TEMPERATURE has been calculated as 293.15
   [K].
   MOLECULAR WEIGHT has been referd as 28.97 [g/mol].
   I assume PRESSURE is 101.325 [kPa].
   SPECIFIC VOLUME has been calculated as .8303325
   [m**3/kg].
   SPECIFIC VOLUME is .8303325 [m**3/kg].
   I finished.

(D)->CELSIUS TEMPERATURE 25.0 NO TOKI NO
   >>WATER NO SONIC VELOCITY <km/h> HA.

   CELSIUS TEMPERATURE has been given as 25.0 [C].
   DENSITY has been referd as 996.95 [kg/m**3].
   COMPRESSIBILITY has been estimated as 4.845E-10
   [1/Pa] at celsius temperature 20 [C] and pressure
   101.325 [kPa].
   BULK MODULUS has been calculated as 2063983488 [Pa].
   SONIC VELOCITY has been calculated as 1438.853 [m/s].
   SONIC VELOCITY is 5179.871 [km/h].
   I finished.
```

Fig. 5. 8. Solving process to obtain specific volume of air and sonic velocity of water.



Scheme of number 1 for SPECIFIC VOLUME

```
To obtain SPECIFIC VOLUME, I select equation EQ1.
To use equation EQ1, I need to know DENSITY of AIR.
To obtain DENSITY of AIR, I select data of AIR.
To use data of AIR, I need to know CELSIUS TEMPERATURE.
I assume CELSIUS TEMPERATURE is 20 [C].
```

Scheme of number 2 for SPECIFIC VOLUME

```
To obtain SPECIFIC VOLUME, I select equation EQ2.
To use equation EQ2, I need to know GAS CONSTANT, ABSOLUTE
 TEMPERATURE, MOLECULAR WEIGHT of AIR and PRESSURE.
Default value of GAS CONSTANT is 8.31433 [J/K/mol].
To obtain ABSOLUTE TEMPERATURE, I select equation EQ7.
To use equation EQ7, I need to know CELSIUS TEMPERATURE.
I assume CELSIUS TEMPERATURE is 20 [C].
To obtain MOLECULAR WEIGHT of AIR, I select data of AIR.
I assume PRESSURE is 101.325 [kPa].
```

Fig. 5. 9. Strategies to find the specific volume of air.

Study in this area has been neglected. Knowledge about fluid property problems is resolved into four parts: characteristic quantities of the objects under consideration, relations between quantities, default decisions about the experimental constants and general knowledge in standard circumstances. The ability of this system is demonstrated. This system can be applied to similar problems by updating the data base. It is also effective to advize users, who are not familiar with problems in property problems.

It is very difficult to clarify the correctness and the limitation of the solving system based on the analysis of practical problems. Therefore, one must apply the system to various kinds of problems and prove its validity by experiment. So far, this system works well. Further improvements expected to provide a more effective and more universal process are as follows: precise expression of current situations[5-6),5-11)], learning system[5-14),5-15)], metaknowledge to control the process[5-6),5-16)], checking system of the results and intelligent interface system.

References of chap. V

5-1) Nilson, G.L., Introducing Artificial Intelligence, (1984), 175, NCC Publ.

5-2) Special Issue on the Applications of Artificial Intelligence in Machine Industries, J. JSME (in Japanese), Vol. 89, No. 815 (1986).

5-3) Watanabe, T. and Nakamura, I., Trans. JSME (in Japanese), Vol. 51, No. 465 (1985), 1678.

5-4) Bobrow, D.G., (Minsky, M.L. ed.), Semantic Information Processing, (1968), 146, MIT Press.

5-5) Novak, G.S., Proc 5th IJCAI, (1977), 286.

5-6) Bundy, A. Byrd, L., Luger, G., Mellish, G., Palmer, M., Proc. 6th IJCAI, (1979), 1017.

5-7) Sakurai, I. and Shimura, M., Information Processing Society of Japan (in Japanese), AI-34 (1984).

5-8) Takagi, A., Itoh, Y., Ohta, Y., Kamihiro, H. and Obara, Y., Information Processing Society of Japan (in Japanese), AI-39 (1985).

5-9) Sleeman, D.H., Artificial Intelligence, Vol. 16, No. 2 (1981), 171.

5-10) Anzai, Y., Uesato, Y. and Tamura, A., Information Processing Society of Japan (in Japanese), AI-41 (1985).

5-11) Anzai, Y., Kondoh, K., Nakamura, H., Tamura, A. and Ura, S., (Suzuki, H. ed.), Object Oriented (in Japanese), (1985), Chap. 8, Kyoritsu Publ.

5-12) Nakamura, I. and Osaka, H., Mechanical Fluid Mechanics (in Japanese), (1982), Kyoritsu Publ.

5-13) Hayes-Roth, F., Waterman, D.A. and Lenat, D.B., Building Expert Systems, (1983), Addison-Wesley.

5-14) Silver, B., Proc. 8th IJCAI, (1983), 429.

5-15) Sakurai, I. and Shimura, M., Information Processing Society of Japan (in Japanese), AI-46 (1986).

5-16) Bundy, A. and Welham, B., Artificial Intelligence, Vol. 16, No. 2 (1981), 189.

VI. Application of Production System to Problems with Changing Conditions

6. 1. Introduction

Applications of artifical intelligence are being made in various areas and in addition to fault diagnosis, facility layout and plant control systems[6-1)], an approach to the combination of computational fluid dynamics and expert system is developing[6-2),6-3)].

In this chapter, practical textbook problems are taken up and an attempt is made to clarify the resolution process. This means not only the significance of the automatic problem solving system in physical problems but also the contribution of the modeling and simulation to the knowledge representation and knowledge manipulation in fluid mechanics.

It is said that the study of knowledge representation must find out a description which holds the meanings in wide variety of problems and guarantees the correctness of processing done by formal transformations[6-4]. Following this opinion, it is necessary to carry out study to confirm the validity of the knowledge representation for the practical problems as well as universal research in abstract area. This seems to be the same case of requirement for both the theory of isotropic turbulence and the detailed experiment of individual flows.

One of the aims of this study is to investigate the applicability of artificial intelligence technologies to engineering and confirm their defects. Attempts to solve elementary mathematical and physical problems automatically have already been made[6-5),6-6]. Here, for the textbook problems in fluids mechanics, which had not been considered previously, we developed a solving system based on the production system.

One formulation of the elementary problems in fluid mechanics is proposed, which clasifies problems into those with no changing conditions and those with change of same. The former are called the static problems which are solved in a unique world. The latter are termed multiple problems in which a solution is obtained by using the relations between static problems. Though this kind of formulation is not a universal one, a fluid property problem can be called a static one, while a system of pipes constitutes a multiple one.

6. 2. Syntactic and semantic analysis of input sentences

In order to achieve smooth interaction with a computer, a well-designed interface is requisite. Therefore, a system to translate Japanese sentences into corresponding inner expressions is adopted. It is based on the extended Japanese LINGOL[6-7]. The syntactic analysis is applied to the input sentence by using internal grammar and a dictionary, and compositions of phrases and words are checked. Then semantic analysis routines stored in inner grammar and dictionary are employed to extract the meanings, and the meanings are registered in the global working memory of the production system. Properties used in the production system to express the data in fluid mechanics problems are as follows:

(1) WANT (main goal): unknown appearing in the given problem.
(2) GOAL (subgoal): unknown created during the solution process.
(3) PROP (known): known quantity appearing in the problem or created during the solution process.
(4) NOTKNOWN (unsolvable): property determined not to be solvable.
(5) LOC (position): configuration of objects.
(6) ACTBYM (action object method): action applied to object by using method.
(7) CONDITION (condition): condition that must be satisfied by the current object considered.
(8) TRANS (transition): changing mode in state of object considered.

The criterion for the selection of properties and the proof of their appropriateness are difficult to explain and the application to concrete problems would presumably contribute to their reasonable confirmation.

One of the results of the syntactic and semantic analysis is shown in Fig. 6.1. In this figure, undetermined values are denoted by the symbol ?, and the units of quantities are transformed into fundamental units of the system.

```
At the pressure 2 kgf/cm2 and the temperature 45 C the specific volume of a
perfect  gas is 0.481 m3/kg.  Determine the gas constant and the  molecular
wright.

 ((WANT (GAS_CONSTANT PERFECT_GAS ?))
  (WANT (MOLECULAR_WEIGHT PERFECT_GAS ?))
  (PROP (SPECIFIC_VOLUME PERFECT_GAS 0.481))
  (PROP (TEMPERATURE SPECIFIC_VOLUME 318.15))
  (PROP (PRESSURE PERFECT_GAS 196140.0)))

A  2 m high by 1 m wide rectangular gate is making an angle of  30  degrees
with  the horizontal and its upper edge is 20 cm below the  water  surface.
Determine  the resultant force on the gate and the location of  the  acting
point.

 ((WANT (LOCATION ACTING_POINT ?))
  (WANT (RESULTANT_FORCE GATE ?))
  (PROP (DEPTH UPPER_EDGE 0.2))
  (LOC (INCLINATION  GATE HORIZONTAL (ANGLE 0.5235)))
  (PROP (WIDTH GATE 2.0))
  (PROP (HEIGHT GATE 1.0))
  (PROP (SHAPE GATE RECTANGLE)))
```

Fig. 6. 1. Results of semantic analysis.

6. 3. *Static problems*

The basic production system consists of the working memory, production rules and the inference engine[6–8)], and forward reasoning is used. Fig. 6.2 shows the production rules used to obtain the viscosity. Variables are denoted by symbols starting with $>$ or $<$, and symbols starting with * are Lisp functions. For example, the rule VIS-6 states that if the goal is the viscosity of some object OBJ and its kinematic viscosity ν and density ρ are known, the

```
(VIS-2 ((GOAL (VISCOSITY >OBJ ?))
        (PROP (STATE <OBJ WATER))
        (PROP (TEMPERATURE <OBJ >T)))
       ->
       ((*CHOICE (*EVAL TABLE_1.5))))
(VIS-6 ((GOAL (VISCOSITY >OBJ ?))
        (PROP (KINEMATIC_VISCOSITY <OBJ >NU))
        (PROP (DENSITY <OBJ <RHO)))
       ->
       ((*PRINT*ANSWER (*SETQ W1 (VISCOSITY <OBJ <RHO <NU))))
        (*POP (GOAL (VISCOSITY <OBJ ?)))
        (*PUSH (PROP (*EVAL W1)))))
(VIS-1 ((GOAL (VISCOSITY >OBJ ?))
        (PROP (STATE <OBJ AIR)))
       ->
       ((*CHOICE (*EVAL TABLE_1.4))))
(VIS-3 ((GOAL (VISCOSITY >OBJ ?))
        (PROP (STATE <OBJ WATER)))
       ->
       ((*CHOICE ((0.001138 <15 C, 1 atom>)))))
```

Fig. 6. 2. Example of production rules.

viscosity μ can be determined by using the relation

$$\upsilon = \rho \times \nu.$$

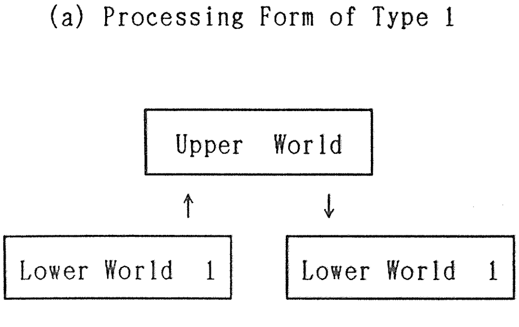
Rule VIS-1, on the other hand, asserts that if the goal is the viscosity and the object is air, then it can be selected from the prescribed table 1.4 by using the Lisp function CHOICE.

6. 4. Multiple problems

Consider the problems with changing conditions such as are the energy conservation problems and problems of the expansion and contraction of gases. One of the typical problems is that of momentum, which requires the force resulting from momentums of a water jet before and after collision with a plate. The local problems before and after the collision may have different quantities but they can be assumed to have the same kinds of properties in the working memory and the same rule base. While the results of the local problems are obtained at a lower level, the resultant force is found by combining these results and by using the rules for the momentum at the higher level. An outline of this process is presented in Fig. 6.3 (a). The upper world gets the results from the lower worlds and obtains the final consequences.

Another typical problem is the one which requires the state of a water jet before or after the collision and can be solved by using some other state and momentum theory. A solution process for this sort of problems is shown in Fig. 6.3 (b). The upper world gets the known result from one of the lower worlds and transfers the information to the other lower world which obtains the final result.

To realize the solution process shown in Fig. 6.3, a multilayered production system is developed, which can initiate the inference in any world at nay time. Each world has its own



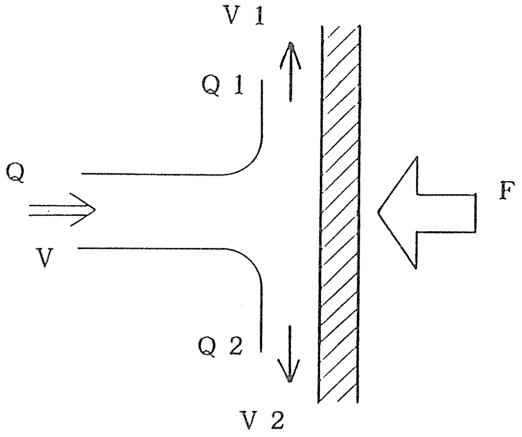
(a) Processing Form of Type 1

(b) Processing Form of Type 2

Fig. 6. 3. Problem processing in multiple worlds.

working memory and production rules. The working memory can be modified by the external worlds.

One example of the moment problems and their inner expression, which corresponds to the contents of the working memory, is shown in Fig. 6.4. This problem belongs to those in Fig. 6.3 (a). In the upper world rules are prescribed, which directs the lower worlds to obtain momentums and finds the acting force. The solution process for this problem is shown in Fig. 6.5. The inference proceeds in accordance with the rules.



```
((WANT (FORCE PLANE ?))
 (TRANS (((PROP (VELOCITY WATER 9.0))
           (PROP (MASS_FLOW_RATE WATER 20.8)))
          ((PROP (VELOCITY WATER 0.0))
           (PROP (MASS_FLOW_RATE WATER 20.8)))
          WATER COLLISION_WITH_PLANE)))
```

Fig. 6. 4. Representation of momentum problem.

6. 5. Conclusions

A solving production system is developed in the light of human solving ability in the elementary problems of fluids mechanics. Problems are classified into static problems with unchanging conditions and multiple problems with changing conditions. It is shown that the standard production system can be readily applied to static problems. To formulate multiple problems, a multilayered production system, which represents worlds performing independent inferences, is designed. More straightforward architecture is constructed to solve multiple problems by the multilayered production system than by the standard production system, and the efficiency is increased. As a practical example, a process for solving a momentum problem is demonstrated.

```
In which world do you execute inference?
 (multiple world   unique world)

***** Start inference in multiple world. *****
----- Start inference before COLLISION_WITH_PLANE. -----

What is the state of WATER?
 (gas   liquid   solid)
You selected liquid as a state of WATER.

<<<<< Matched rule is MOMENTUM_OF_FLUID_2 >>>>>
  .     .     .     .     .     .     .     .
  .     .     .     .     .     .     .     .
  .     .     .     .     .     .     .     .
<<<<< Matched rule is MOMENTUM_OF_FLUID_1 >>>>>

Momentum of water is 187.0315 (N.s)

----- Finish inference before COLLISION_WITH_PLANE. -----

----- Start inference after COLLISION_WITH_PLANE. -----
  .     .     .     .     .     .     .     .
  .     .     .     .     .     .     .     .
  .     .     .     .     .     .     .     .
<<<<< Matched rule is MOMENTUM_OF_FLUID_1 >>>>>

Momentum of water is 0 (N.s)

----- Finish inference before COLLISION_WITH_PLANE. -----

<<<<< Matched rule is CHANGE_IN_MOMENTUM_2 >>>>>

FORCE on PLANE is -187.0315 (kN)

The problem is solved.
```

Fig. 6. 5. Processing in multiple worlds.

References of chap. VI

6-1) Special issue on expert systems, J. Japanese Society for Artificial Intelligence (in Japanese), Vol. 3, No. 1 (1988).

6-2) Tong, S.S. AIAA-86-0242 (1986).

6-3) Andrews, A.E., AIAA J., Vol. 26, No. 1 (1988), 40.

6-4) Ohsuga, S., (in Japanese), Vol. J69-D, No. 7 (1986), 1009.

6-5) Takagi, A., Ito, Y., Ohta, Y., Uehiro, K. and Obara, T., Information Processing Society of Japan (in Japanese), AI-39 (1985).

6-6) Bundy, et al., Proc 6th IJCAI, (1979), 1017.

6-7) Tanaka, H., Motoyoshi, H. and Yamanashi, M., Cognitive Psychology Learned by LISP 3 (in Japanese), (1983), Univ. Tokyo Press.

6-8) Saitoh, Mizoguchi, H., Bulding Intelligent System (in Japanese), (1982), Ohmu Publ.