Journal of Hydroscience and Hydraulic Engineering
Vol. 9, No. 2, January, 1992, 81-90

A FLOOD CONTROL SUPPORT ENVIRONMENT BASED ON COOPERATING KNOWLEDGE-BASED SYSTEMS

By

T. TAKASAO

M. SHIIBA

T. HORI

Department of Civil Engineering, Kyoto University,
Yoshida-Honmachi Sakyo-ku, Kyoto, 606, Japan

and

H. SASAKI

NTT Software Labaratories, Nippon Telegraph and Telephone Corporation
1-9-1, Kohnan Minato-ku, Tokyo, 108, Japan

SYNOPSIS

A flood control environment, which is composed of the Inference System (**IS**) and the Procedural Knowledge System (**PKS**), is designed. **PKS** deals with problems which can be solved mathematically, while **IS** takes charge of problems which need a knowledge-based approach. **IS** is composed of self-contained knowledge-based systems, each of which is an expert system to deal with a partial issue of flood control problems. Knowledge-based systems are classified into two types of problems: a system with a deterministic inference process based on a production system and the other with a fuzzy inference process.

INTRODUCTION

Flood management by the use of storage reservoirs comprises various types of information gathering and processing, such as the observation of rainfall and discharge, the on-line prediction of those quantitative data and the comparison of observations and predictions with reservoir operation regulations. Hence it should integrate the two types of information processing: namely, the procedural one and the qualitative one. The former uses mathematical models such as rainfall and runoff prediction the latter mainly utilizes knowledge, in linguistic style, like reservoir operation regulations and experience of reservoir operators. Therefore, an attempt only based on an AI method which is popular in developing medical examination systems is not practical to design a flood control support system.

From this point of view, we propose a flood control support environment based on distributed and cooperating knowledge-based systems. The system proposed herein has two levels of cooperation among its subsystems. On the upper level of cooperation, a procedural knowledge system, which processes quantitative data with mathematical algorithm, is linked with an inference system, which deals with qualitative information and knowledge expressed in the form of language. They synthetically support flood control work through information processing in each system and information interchange between them.

The lower level of cooperation takes place among knowledge-based systems in the inference system. The inference system proposed here is composed of several self-contained knowledge-based systems, each of which is an expert system with an inference engine, a knowledge base and a data base.

Each knowledge-based system deals with a partial issue of a flood control problem. Flood control problems, such as how much water should be released from a storage reservoir, are solved through the inference process in each knowledge-based system and the communication among them. In this study, we adopt two types of knowledge-based systems according to the information types to be processed in each system: one with an inference process based on a production system and the other with a fuzzy inference process. The environment based on the two levels of cooperation described above can synthetically support the flood control work.

WHOLE STRUCTURE OF THE FLOOD CONTROL SUPPORT SYSTEM

In order to support the total process of flood control work and to harmonize the algorithmic approach and the knowledge-based approach, we have to note the followings in designing a whole structure of a flood control support system.

1. An accurate linkage should be maintained between a procedural knowledge system, which processes quantitative data based on mathematical algorithm, and an inference system, which mainly utilizes knowledge expressed in a linguistic form, and the two systems should be able to interchange data and information automatically and immediately.
2. Quantitative data such as rainfall and discharge must be automatically transmitted to the flood control support system from a data acquisition system. A system with much manual input work is not suitable for supporting flood management because backup information has to be provided without delay.
3. A flood control support system should be equipped with a graphical and interactive display system so that its users can easily understand backup information provided by the system.

Taking these conditions into consideration, we introduce the whole structure of the flood control environment shown in Fig. 1. Inference Environment Management System (**IEMS**) provides an interface between the Data Acquisition System (**DAS**) and the Inference System (**IS**), and supervises **IS** and the Display System (**DS**). **IEMS**, keeping watch on the time when new data are transmitted from **DAS**, processes the transmitted data in a fixed form and gives them to **IS** with an instruction to start inference. In **IS**, supporting information such as a recommended release from a reservoir is prepared through the inference based on knowledge bases with reference to acquired data. If the predictions of rainfall or discharge are necessary in the inference process, **IS** calls the Procedural Knowledge System (**PKS**) automatically. **PKS**, composed of programs for on-line prediction of rainfall and runoff, returns the predictions to **IS**. Using the predictions returned from **PKS**, **IS** advances the inference for determination of supporting information in a continued series. When the inference process ends, **IS** gives the results to **IEMS**. Then **IEMS** comes into action again and puts the results of inference and the observed data in the Recording Data Base. Then **DS** receives instructions to display graphically supporting information such as recommended release with the observations transmitted from **DAS**.

IS and **PKS** are installed on different types of CPUs because the former needs a different way of information processing from the latter. Using two types of CPUs reduces the computational burden on a computer. Thus the flood control support system can immediately provide supporting information to its users who are required to make decisions without delay.

The flood control support system with the structure described above can easily be installed in local river management offices by only adding another work station to existing procedural knowledge systems and data acquisition systems so that **IEMS**, **IS** and **DS** as developed here can work.

THE INFERENCE SYSTEM

Framework of the Inference System for Supporting System on Flood Control

Flood control by the use of storage reservoirs comprises various units of information processing as mentioned in the introduction and the number of these units may increase as hydrologic observation systems are improved. Therefore, in order to support river managers in flood control problems which

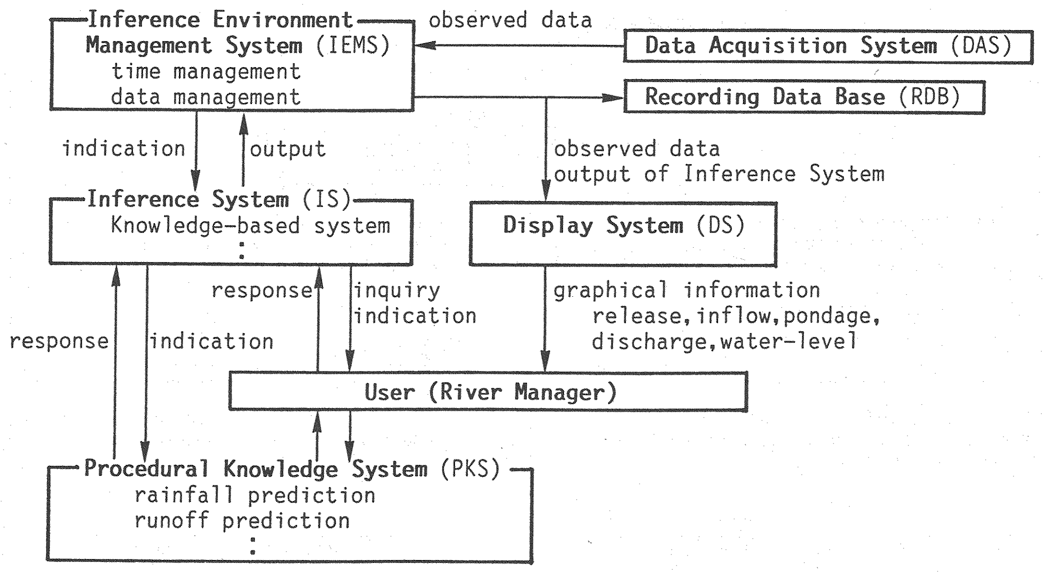

Fig. 1 Whole structure of the flood control support system.

will become more complex, it is necessary to design an inference system, the performance of which can be easily improved through the addition or correction of knowledge. From this viewpoint, we divide a flood control problem into several partial issues, make knowledge-based systems, each of which takes charge of a partial issue, and design an inference system in which these knowledge-based systems cooperate to solve the problem. Each knowledge-based system is an expert system with its own inference engine, data base and knowledge base, and has an interface to communicate with other systems. This framework of knowledge-based approach based on cooperating knowledge-based systems is called a distributed knowledge-base model (Smith and Davis (1)).

In this study, we design an inference system composed of five knowledge-based systems for application to flood control by the Amagase Dam reservoir as shown in Fig. 2 (for the explanation about the Amagase Dam reservoir and its operational regulations, see APPLICATION). Let us explain the framework of these knowledge-based systems below. **KSDOR** has reservoir operation regulations in its rule base and takes a fundamental role in determining a release from the reservoir. **KSDOR** is the first system to begin inference when the Inference System receives the instruction to come into action. When **KSDOR** needs judgement from other knowledge-based systems for its application process of reservoir operation regulations, it sends messages to **KSMWL**, **KSRCR** and **KSMID**.

At each time during flood control, **KSMID** gives judgement on whether or not inflow discharge to the reservoir has already passed its peak (for detail rules regarding this judgement, see APPLICATION). **KSMWL** supports decisions on maximum water level at the lower reach of the river. It gives judgement on whether or not the water-level at the lower reach of the river has already passed its peak. Whether or not the remaining capacity of the reservoir is sufficient for control of the coming flood is judged by **KSRCR**. **KSRCR** plays an important role particularly in determining the time when the control strategy shifts to the second control stage from the first and determining the time when the second control stage should be terminated (see APPLICATION for explanation of the first and second control stages and details about the rule base). These knowledge-based systems have to utilize predictions of rainfall and discharge. And they can call the Procedural Knowledge System automatically when they need these data in their inference processes.

The three knowledge-based systems mentioned above, namely, **KSMID**, **KSMWL** and **KSRCR**, serve to support **KSDOR** in applying reservoir operation regulations. Furthermore, **KSACR** judges whether or not the difference between a current release and that determined by the above four

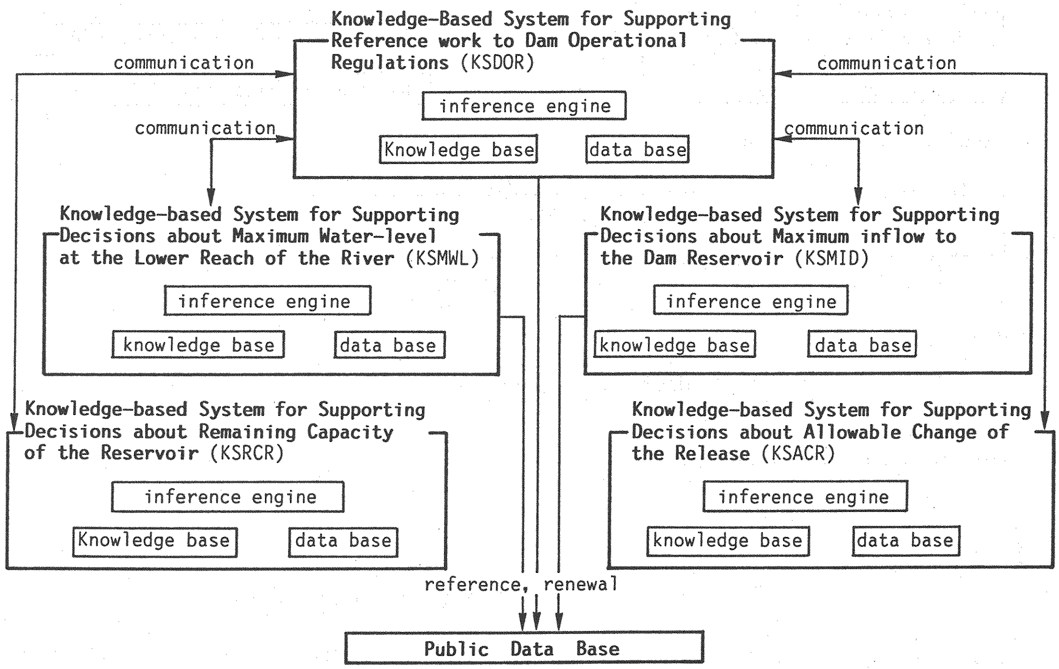

Fig. 2 Framework of the inference system for flood control supporting system.

knowledge-based systems exceeds the allowable discharge value. A sudden change in the release from a storage reservoir imperils residents at the lower reach of the river and has bad effects on flood fighting work and on refuge activity. If **KSACR** draws the conclusion that the release selected by the above four systems is not acceptable, it gives **KSDOR** an instruction to make another inference, taking the fact into consideration. The reader may refer to Takasao *et al.* (2) for an example of communication between knowledge-based systems.

The framework of the inference system mentioned above is intended to support flood control by the Amagase Dam reservoir, but it is also valid for flood control by other reservoir systems, for operation regulations of other storage reservoirs have a similar structure and the Amagase Dam reservoir has one of the most complex operation regulations in Japan.

The introduction of cooperating knowledge-based systems enables us to modularize knowledge. Therefore the performance of the Inference System can easily be improved by addition of new knowledge-based systems which can treat new problems to be solved without reconstruction of whole knowledge bases.

The implementation of the Inference System based on cooperating knowledge-based systems requires programming techniques to generate all of the knowledge-based systems on a computer and make them communicate one another. These techniques can be accomplished through object-oriented programming (Cox (3)), in which a system to be implemented is divided into self-contained modules called "objects" and the processing is made by "message passing". Furthermore, knowledge-based systems to be generated in a computer have similar structure, that is, each of them has an inference engine, a knowledge base and a data base. The "class" description provided by the object-oriented programming makes it easy to implement the knowledge-based systems with similar structures. For this purpose we adopt Smalltalk-80 (Goldberg and Robson (4)), which is a typical object-oriented programming language, for implementation of the Inference System.

Design of each knowledge-based system

The knowledge-based systems introduced in the previous subsection are classified into two categories: one includes a system which has a knowledge base composed of deterministic production rules, and the other includes a system with a knowledge base composed of fuzzy rules. **KSDOR** is a typical knowledge-based system belonging to the former category. **KSMID** also belongs to the former when the Procedural Knowledge System (**PKS**) is equipped with stochastic runoff algorithm because the **PKS** can provide **KSMID** a probability distribution of the time of inflow peak. On the contrary, if **PKS** is not equipped with the stochastic algorithm but with a simulation model of future inflow, which can provide several cases of simulated hydrographs but cannot provide its probability distribution, **KSMID** needs to judge based on incomplete information. In this case, it is suitable that **KSMID** is equipped with a fuzzy inference structure (the reader may refer to Mamdani (5) concerning fuzzy inference theory,). The same statements are also true for **KSMWL** and **KSRCR**.

(1) A knowledge-based system with a deterministic inference process based on a production system

A knowledge-based system with a deterministic inference process has a knowledge base composed of production rules (the reader may refer to Bundy (6) about the framework of a production system), each of which represents knowledge in an if-then form. A deterministic inference process means a process in which a rule is adopted only when the condition part of the rule matches the data in a data base symbolically. The inference engine designed in this study has following characteristics:

1. When several rules match the data in a data base at the same time, the rule which matches the newest data is adopted. This way of conflict dissolution makes results of inference unrelated to the order of production rules, which enables it easier to correct a rule base.

2. While each knowledge-based system has a deterministic rule base, the data referred to in its inference process may be an uncertain datum when it is an answer from a knowledge-based system with a fuzzy inference process. A fuzzy knowledge-based system gives the inference results with certainty degree. For example, let us explain the case in which **KSMID** is a fuzzy knowledge-based system and it concludes that the inflow peak has passed with certainty degree of 0.8 and not 0.2. The knowledge-based system which received the answer has to infer, considering two cases of conflicts between data. In this case, it produces two inference processes for the data and the result of each inference process has its certainty degree equal to that of the datum. When a rule has two conditions and data matching them are both with certainty degrees, the result has a certainty degree which is the product of the certainty degrees of the two data. This inference method with the propagation of certainty degree provides the interface with fuzzy knowledge-based systems and enables the Inference System to treat both deterministic data and uncertainty data.

(2) A knowledge-based system with a fuzzy inference process

In designing a fuzzy knowledge-based system, how to decide fuzzy rules is the most important problem. Relations between a condition and a conclusion expressed by these rules are essentially uncertain. Therefore, even if rules are determined through an interview to reservoir operators, it is necessary to design a rule making framework which enables systematic correction of the rules based on the system performance. Taking this point into account, we introduce parameters which denote an effect of each condition on a conclusion and provide a rule making system which automatically produces a fuzzy rule set when the parameters are given.

Let us take **KSMID** as an example to observe the rule making process. In judging whether or not the reservoir inflow has passed its peak, **KSMID** uses four input variables: estimation about the accuracy of inflow prediction model (x_1), the ratio of the maximum inflow observations acquired until then to the maximum of inflow predictions during the coming five hours (x_2), the elapsed time since the maximum inflow is observed (x_3) and the current rainfall (x_4). The output variable (y) is denoted by a number from -1 to 1 according to whether the inflow peak has passed or not (the closer to 1, with greater certainty the system concludes that the inflow peak has passed). Using these input and output

Table 1 Sample data for explanation of a fuzzy rule making system.

(a) Values of weight parameters according to x_1

x_1	α	β	θ
good	0.80	0.10	0.10
meduim	0.50	0.25	0.25
bad	0.20	0.40	0.40

(b) Values of p_2, p_3 and p_4 in making a fuzzy rule set

x_2	p_2	x_3	p_3	x_4	p_4
large	-1.0	short	-1.0	much	-1.0
medium	0.0	medium	0.0	medium	0.0
small	1.0	long	1.0	little	1.0

(c) Pre-determined range of θ for the scopes of y

y	θ
small	$-1.00 \leq \theta < -0.70$
more or less small	$-0.70 \leq \theta < -0.25$
medium	$-0.25 \leq \theta < 0.25$
more or less large	$0.25 \leq \theta < 0.70$
large	$0.70 \leq \theta < 1.00$

variables, inference rules can be made based on the following concept: when the estimation of the inflow prediction model (x_1) is good, reservoir operators attach importance to x_2 rather than x_3 and x_4, while they attach importance to x_3 and x_4 in the case that x_1 is bad. In this rule framework, x_1 can be regarded as a special variable because it determines the effect of x_2, x_3 and x_4 on y. Therefore, in making fuzzy inference rules, we introduce the parameters α, β and γ, each of which denotes the weight of each input variable on the output, and use the variable θ to indicate whether the inflow peak has passed. Furthermore, let p_2, p_3 and p_4 be the variables used in making rules, which correspond to x_2, x_3 and x_4, respectively. Then we use the following relation among them:

$$\theta = \alpha x_2 + \beta x_3 + \gamma x_4, \quad \alpha + \beta + \gamma = 1, \quad \alpha, \beta, \gamma \geq 0$$

β is a value close to 1 when the inflow prediction model is good, while α and γ will be rather high in the case of an unreliable prediction model. Once the parameters are determined, we put the combination of -1, 0 and 1 into p_2, p_3 and p_4 according to the combination of sub-scopes of these input variables and computes the value of θ. Then the sub-scope of the output variable according to the combination of sub-scopes of the input variables is determined by comparing the computed θ value with predetermined range of y for its sub-scopes.

For example, let us use the values shown in Table 1, and consider the problem: which sub-scope to select for y in the case where x_1 is "good", x_2 is "large", x_3 is "medium" and x_4 is "little". In this case, the rule making system compute θ like the following:

$$\theta = 0.80 \times -1.0 + 0.10 \times 0.0 + 0.10 \times 1.0 = -0.7.$$

According to Table 1, the scope of output variable which corresponds this condition should be "little". Consequently, the system automatically produces the fuzzy inference rule like the follwing based on the value of θ:

IF x_1 is good and x_2 is large and x_3 is medium and x_4 is little THEN y is "little".

Using this approach for making fuzzy rules, we can clarify the decision making strategy of reservoir operators, and hence the flood control support system can be used not only as a decision aid for flood control but as a training simulator for flood control work. This system can automatically simulate the

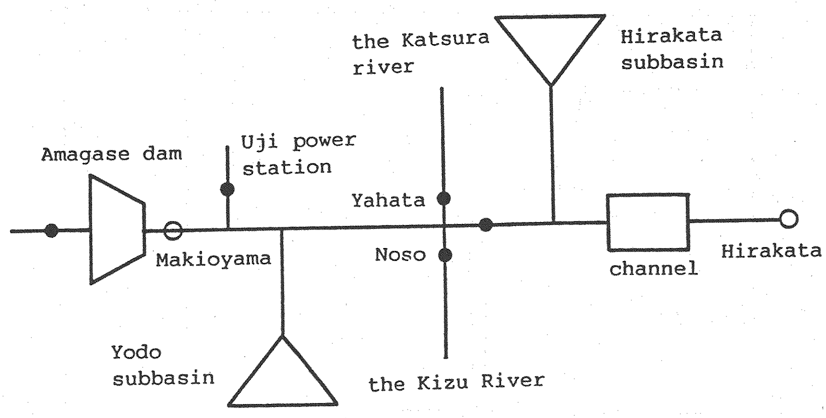

Fig. 3 A Reservoir for system application.

process of decision making itself. Reservoir operators can realize the defects in their decision making principles thanks to the results simulated by the system and can correct their decision making.

APPLICATION TO THE REAL RESERVOIR SYSTEM

In this section, we will test the flood control support system designed in this study on the Amagase Dam reservoir using the data from the flood caused by Typhoon No. 10 in 1982. The Amagase Dam reservoir is located on the upper stream of the Yodo River (see Fig. 3) and has a flood control capacity of 2.0×10^7 m^3. The core systems of the flood control support system, **IEMS**, **IS** and **DS**, are installed using Smalltalk-80 on Sony Tektronix Artificial Intelligence System 4404.

The control strategy indicated by reservoir operation regulations (Yodo River Dams Control Office (7)) has two parts: int the first control stage, the release from the reservoir should be 840m^3/s if the inflow discharge is more than 840m^3/s and *less than its maximum*, and in the second control stage, the release from the reservoir should be 160m^3/s *until the water-level at Hirakata point reaches its peak*). These regulations are stored in the knowledge base of **KSDOR**. The italicized parts of the above-mentioned regulations of the Amagase Dam reservoir require reservoir operators to make judgements on the maximum inflow to the reservoir, the maximum water-level at Hirakata point and the remaining reservoir capacity, each of which is supported by **KSMID**, **KSMWL** and **KSRCR**, respectively. In this application, we assume that the stochastic runoff prediction systems are available in **PKS** and do not use fuzzy rules but deterministic production rules for the implementation of these three systems. **KSMID** concludes that the inflow discharge has reached its maximum when none of the inflow predictions during the coming five hours exceeds the maximum of the inflow observations acquired until then. In the same way, **KSMWL** concludes that the water-level at Hirakata point has reached its maximum when none of the water-level predictions during the coming five hours exceed the maximum of the observations acquired until that time. **KSRCR** computes the future reservoir pondage at 15 minutes' intervals based on inflow predictions under the condition that the second control stage begins 30 minutes later. Using these predictions, **KSRCR** concludes that the remaining capacity is not sufficient for the second control stage when either of the following two cases occurs: firstly, if the prediction of reservoir inflow at the time when the prediction of reservoir pondage exceeds the capacity is more than 840m^3/s, and secondly if, during the next 150 minutes, the second control stage causes the reservoir pondage to exceed the flood control capacity.

The releases from 12:00 a.m. on August 1 to 12:00 p.m. on August 2 in 1982 were decided by the flood control support system based on the rule bases as mentioned above. Each process of inference by the system took about 2 or 3 seconds at the longest, in this example. Figure 4 shows a typical result of reservoir control during the whole term by the flood control support system. According to the control results shown in Fig. 4, the time when the system began the second control stage was later than the actual time of the inflow peak. One reason for this time lag is that the judgement on the

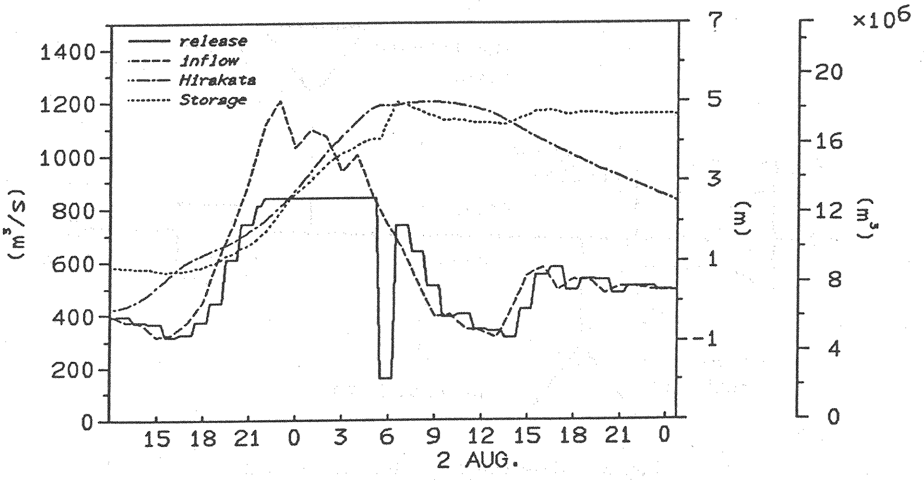

Fig. 4 Typical result of reservoir control by the flood control support system.

time of inflow peak has to be based on predictions of inflow discharge and the control stage indicated by the regulations cannot be redone. Another reason is that the knowledge-based system, **KSRCR**, concluded that the remaining capacity of the reservoir was not sufficient for the second stage control right after **KSMID** concluded that the inflow peak had passed.

Figure 5 shows the sample output of the Display System. The screen is divided into four sub-windows and observations and releases determined by the flood control support system are displayed graphically. These windows are automatically renewed according to data acquisition and progress of the inference process. The explanation windows of the inference process and its reasoning should be added to the Display System, which is now under development.

CONCLUSION

In this study, we have proposed a flood control support system based on the combined use of traditional algorithmic approach and knowledge-based approach. In the knowledge-based approach, we introduced into the inference system two types of knowledge-based approach: the first has a knowledge base composed of deterministic production rules while the second has fuzzy inference rules. Design of flood control support systems along this concept is promising because it can provide an environment which enables to treat various types of information necessary in flood control.

This research was partially supported by River Environment Fund (No. 201) with a principal investigator of Takuma Takasao).

REFERENCES

1. Smith, R.G., and R. Davis: Frameworks for cooperation in distributed problem solving, *IEEE Transactions on Systems, Man, and Cybanetics, SMC-11(1)*, 61-71, 1981.

2. Takasao, T., M. Shiiba and T. Hori: Design of a flood control support system based on a distributed knowledge-base model, *Proc. of Pacific International Seminar on Water Resources Systems, Tomamu*, pp.272-287, 1989.

3. Cox, B.J.: Object oriented programming; an evolutionary approach, Trans. Maekawa, M. (ed.) (Tokyo: Toppan, 1988).

4. Goldberg, A., and D. Robson: Smalltalk-80: the language and its implementation, Trans. Aiiso, H. (Tokyo: Ohm Publishing Company, 1987), 1987.

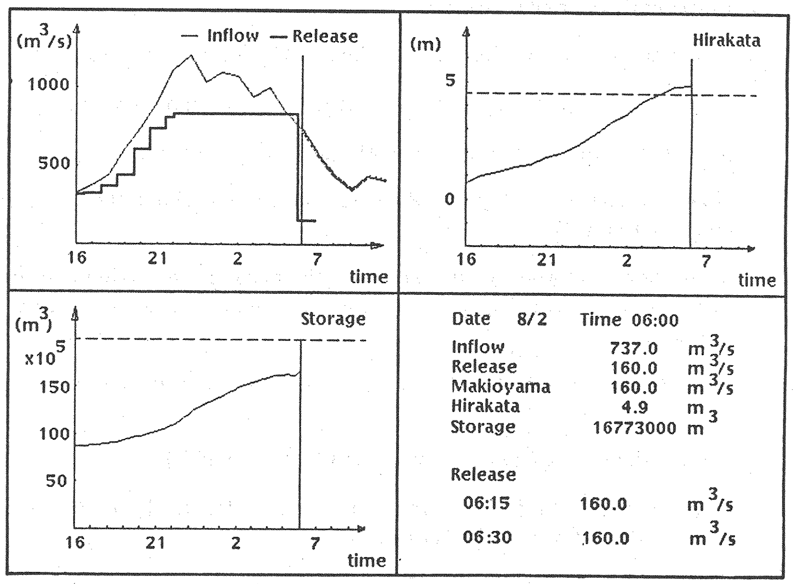

Fig. 5 Sample output of the Display System.

5. Mamdani, E.H.: Advances in the linguistic synthesis of fuzzy controller, Int. J. Man-Machine Studies, Vol.8, No.6, pp.669-678, 1976.

6. Bundy, A., R.M. Burstall, S. Weir, and R.M. Young: Artificial intelligence: an introductory course, Trans. Nagao, M. (ed.) (Tokyo: Kindai Kagaku, 1981), 1978.

7. Ministry of Construction Yodo River Dams Control Office: Administration of Yodo river, a leaflet, (in Japanese).

APPENDIX–NOTATION

The following symbols are used in this paper:

DAS = Data Acquisition System

DS = Display System;

IEMS = Inference Environment Management System;

IS = Inference System;

KSACR = knowledge-based system which supports decisions about allowable change of release dischrge;

KSDOR = knowledge-based system which supports reference work to dam operational regulations;

KSMID = knowledge-based system which judges whether or not inflow dischrge to a dam reservoir has alreday passed its peak;

KSMWL = knowledge-based system which supports decisions about maximum water level at a lower reach of a river;

KSRCR = knowledge-based system which judeges whether or not the remaining capacity of a reservoir is sufficient for control of coming flood;

PKS = Procedural Knowledge System;

p_2 = parameter used in the fuzzy rule making, which correponds to x_2;

p_3 = parameter used in the fuzzy rule making, which correponds to x_3;

p_4 = parameter used in the fuzzy rule making, which correponds to x_4;

x_1 = input variable of a fuzzy inference rule (estimation about the accuracy of inflow prediction model);

x_2 = input variable of a fuzzy inference rule (the ratio of the maximum of inflow observations acquired till then to the maximum of inflow predictions during the coming five hours);

x_3 = input variable of a fuzzy inference rule (elapsed time since the maximum inflow is observated);

x_4 = input variable of a fuzzy inference rule (current rainfall);

y = output variable of a fuzzy inference rule , which denotes by a number from -1 to 1 whether the inflow peak has passed or not;

α = parameter which denotes the effect of input variable x_2 on the output;

β = parameter which denotes the effect of input variable x_3 on the output;

γ = parameter which denotes the effect of input variable x_4 on the output; and

θ = variable to indicate whether the inflow peak has passed, which is used in the rule making process.

(Received August 24, 1990 ; revised December 20, 1991)