

INTERACTIVE GRAPH PRODUCTION SYSTEM

Kyota Aoki
Department of Information Science
Faculty of Engineering
Utsunomiya University
Utsunomiya 321-31
JAPAN

Kokichi Tanaka
Department of Information and Computer Sciences
Faculty of Engineering Science
Osaka University
Toyonaka 560
JAPAN

In this paper we propose a new production system called Interactive Graph Production System (IGPS) which represents situations which have interactions. An IGPS is constructed by two labelled directed graphs and two sets of production rules which control interactions and changing of situations. First, IGPS is defined. Then we show examples of IGPS: three coin problem and monkey and banana problem, for making clear structure and move of an IGPS. Next execution of IGPS is discussed. IGPS interpreter must have some functions for efficient execution of IGPS. And for making productions efficiently, we need graph production editing system which enables us to handle graph productions in graphical forms.

1. INTRODUCTION

Eversince production systems (PS) were first proposed by Post [7] as a general computational mechanism, the methodology has been a great deal of development and has been applied to a diverse collection of problems. A production system may be viewed as consisting three components: a set of rules, a data base, and an interpreter for the rules. In the simplest design, a rule is an ordered pair of symbol strings with a left and right hand sides, the set of rules has a predetermined total ordering and the data base is simply a collection of symbols.

Throughout much of the work reported, there appear to be two major views of PSs, as characterized on one hand by the psychological modeling efforts (PSG, PAS II, VIS, etc.) [5, 6] and on the other by the performance-oriented, knowledge-based expert systems (e.g. MYCIN, DENDRAL) [2, 3]. For the psychological modelers, production rules offer a clear, formal, and powerful way of expressing basic symbol processing acts, which form the primitives of information processing psychology. For the designer of knowledge-based system, production rules offer a representation of knowledge that is relatively easily accessed and modified, making it quite useful for systems designed for incremental approaches to competence.

Now we have trend to apply PSs to more and more complex problems. Those problems need more complex knowledge-bases. For instance, the

DENDRAL system uses a literal pattern match, but its patterns are graphs representing chemical classes. For expressing complex situations graphs are better than collections of literals for human understandability. In many cases graphs are used for describing situations, and we see many usage of graphs for explanations while collections of assertions are used for an internal representation of a system. So we want to construct a PS which treats not symbol strings or collections of assertions, but graphs.

In complex problems we find two situations interacting each other. For instance, in a problem solving for controlling a robot we find a robot and its environment interacting each other. We think that it is better than a description by a single situation which represents the robot and its environments, to describe the world by the set of two sub-situations: one represents the robot and the other does its environment; and changing of the world by interactive changing of two sub-situations. So we want to construct a PS which can express systems that interact each other also.

In this paper we will propose a new graph production system that has interactions called an Interactive Graph Production System (IGPS). An IGPS represents a situation by a set of two labelled directed graphs and changing of situations by rewriting rules of graphs. So we will construct an IGPS based on the graph grammar system which is discussed in [1]. In this paper we will first describe definitions of an

IGPS, next show some examples of IGPSs for making clear the method of descriptions and moves, and then discuss execution of an IGPS.

2. INTERACTIVE GRAPH PRODUCTION SYSTEM

In this section we will describe definitions of an IGPS. An IGPS is constructed by two labelled directed graphs and two sets of production rules which control interactions and changing of situations.

2.1 Situations of an IGPS

A situation of an IGPS is represented by a tuple,

$$(\sigma_0, \sigma_1),$$

where σ_0 and σ_1 are sub-situations described by labelled directed graphs. More formally, a sub-situation σ_i is represented by a 3-tuple,

$$(N_i, L_i, E_i),$$

where N_i is a set of nodes, L_i is a function: $N_i \rightarrow$ a set of labels, and E_i is a set of edges and is included into $N_i \times N_i$.

[Example 1] Here we show an example of a sub-situation.

Let $N_i = \{1, 2, 3\}$,

$$L_i(1) = \text{he},$$

$$L_i(2) = \text{is},$$

$$L_i(3) = \text{diligent},$$

and $E_i = \{(1, 2), (2, 3)\}$.

Then the sub-situation (N_i, L_i, E_i) is the

graph: $\text{he} \xrightarrow{\text{is}} \text{diligent}$.

2.2 Structure of IGPS

An IGPS is a 7-tuple:

$$S = (C, V, R, i_0, i_1, P_0, P_1),$$

where C is a set of labels of sub-situations, V is a set of variables whose ranges are subsets of C , R is a function: $V \rightarrow 2^C$ which defines the ranges of variables, i_0 and i_1 are two finite initial sub-situations whose sets of labels

are C , and P_0 and P_1 are sets of productions which are applied to two sub-situations: σ_0 and σ_1 , respectively.

In this paper we will describe elements of C by strings of lower case letters and elements of V by strings of upper case letters.

2.3 Variables

An element of V is v_i whose range is $R(v_i) \in C$. A variable v_i can have a value of an element of $R(v_i)$.

[Example 2] Here we show an example of a variable.

Let $C = \{\text{station, school, boy, girl}\}$,

$$V \ni \text{PLACE},$$

and, $R(\text{PLACE}) = \{\text{station, school}\}$.

Then the value of 'PLACE' can be 'station' or 'school', but it can not be 'boy' nor 'girl'.

2.4 Productions

We describe a production of P_0 or P_1 by the form:

$$(G_1) \quad G_2 \Rightarrow G_3,$$

where G_1 is a labelled directed graph whose $|N_1|$ is a non-negative integer and the set of labels is CUV , G_2 is a labelled directed graph whose $|N_2|$ is a natural number and the set of labels is CUV , and G_3 is a labelled directed graph whose set of labels is $CUV \cup \{\text{null}\}$ and satisfies the conditions listed below.

$$N_3 \supset N_2 \quad \text{and} \quad E_3 \supset E_2,$$

and any $n_3 \in N_3 : L_3(n_3) \in V$;

there exists n_2 or n_1 :

$$n_2 \in N_2 \quad \text{and} \quad L_2(n_2) = L_3(n_3),$$

and/or $n_1 \in N_1 \quad \text{and} \quad L_1(n_1) = L_3(n_3)$.

At an application of a production each variable has one value. When two labelled directed graphs are compared, a variable v and a constant c are compatible if the value of v is c .

[Example 3] We show an example of a production in Fig. 1. In this example C , V and R of the IGPS are same as those in Example 2.

2.5 Effect of Application of Production

Let a situation of an IGPS be (σ_0, σ_1) , and let a production $p_{ij}: (G_1) G_2 \Rightarrow G_3$; be included in P_i ($i=0, 1$). Then the production p_{ij} may be applied, if Condition 1 and 2 are satisfied.

[Condition 1] Let σ be σ_{1-i} , and let G be G_1 , then Condition 1' is satisfied. If G is an empty graph, then this condition is satisfied.

[Condition 1']

Let $\sigma = (N, L, E)$,

then there exists $N' \subset N$: $(N', L, E \cap N' \times N')$ and G are compatible.

When Condition 1 is satisfied, the production p_{ij} can be applied if Condition 2 is satisfied.

[Condition 2] Let G_2' be a graph which is made by rewriting labels of G_2 which are elements of V and used in G_1 as the value of the variable. Let σ be σ_i and G be G_2' , then Condition 1' is satisfied.

If a production satisfying Condition 1 and 2 is applied, then a sub-graph of σ_i matching G_2 is rewritten as G_3' . Here G_3' is a graph made by rewriting labels which are elements of V as the value of the label.

In an IGPS, production has a structure described in 2.4, therefore deletion of nodes or edges is not available. But in an IGPS the special label 'null' expresses that the node will be not rewritten nor referred. Hence an IGPS interpreter can delete a node whose label is 'null' and edges which connect those nodes, there are no differences in moves of the IGPS.

[Example 4] We show an example of a production and its application in Fig. 2. In this example,

$C = \{ \text{monkey, at, place1, place2, box, move} \}$,

$V = \{ \text{PLACEX, PLACEY} \}$,

and $R(\text{PLACEX}) = R(\text{PLACEY}) = \{ \text{place1, place2} \}$.

When a situation of the IGPS is (σ_0^0, σ_1^0) of

(boy $\xrightarrow{\text{PLACE}}$ boy $\xrightarrow{\text{PLACE}} \Rightarrow$ girl $\xrightarrow{\text{PLACE}}$

Fig. 1 An example of a production of IGPS.

Fig. 2, if production-1 of Fig. 2 is an element of P_0 of the IGPS, then in production-1 the value of PLACEX is place2 and the value of PLACEY is place1, and the production can be applied to σ_0^0 . If the production is applied to σ_0^0 , then a situation of the IGPS becomes to be (σ_0^1, σ_1^0) of Fig. 2.

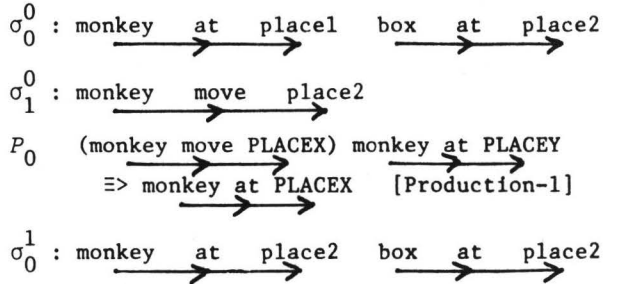


Fig. 2 Example of a production and its application.

2.6 Moves of an IGPS

One move of an IGPS is constructed by two sub-moves. When a situation of an IGPS is (σ_0^0, σ_1^0) , one sub-move is an application of an element of P_0 to σ_0^0 , or preservation of σ_0^0 when no element of P_0 can not be applied to σ_0^0 . Let the result be (σ_0^1, σ_1^0) . Next one sub-move is an application of an element of P_1 to σ_1^0 , or preservation of σ_1^0 when no elements of P_1 can not be applied to σ_1^0 . Let the result be (σ_0^1, σ_1^1) .

3. SOME EXAMPLES OF IGPS

In this section we show some examples of IGPS for making clear the structure of an IGPS and the moves of an IGPS.

3.1 Three Coin Problem

Here we show the three coin problem of Jackson [4].

Problem: Given three coins initially HHT (i. e., head, head, tail), in exactly three moves

make all coins show the same face. A move consists of flipping a coin over.

We show below elements of an IGPS which describes the three coin problem and chunks of knowledge for solving the problem.

$C = \{ \text{start, cont, flip, end, coin1, coin2, coin3, inc, count0, count1, count2, count3, head, tail, op, counter} \},$
 $V = \{ \text{COINX, COINY, COINZ, CONT, COUNT, COUNTALL, COUNTNEXT, STATE, STATEX, STATEY} \},$
 $R(\text{COINX}) = R(\text{COINY}) = R(\text{COINZ}) = \{ \text{coin1, coin2, coin3} \},$
 $R(\text{CONT}) = \{ \text{cont, end} \},$
 $R(\text{STATE}) = R(\text{STATEX}) = R(\text{STATEY}) = \{ \text{head, tail} \},$
 $R(\text{COUNTALL}) = \{ \text{count0, count1, count2, count3} \},$
 $R(\text{COUNTNEXT}) = \{ \text{count1, count2, count3} \},$
 $R(\text{COUNT}) = \{ \text{count0, count1} \},$
 $i_0 = \text{start} \rightarrow \text{cont},$
 $i_1 = \begin{array}{ll} \text{inc} \rightarrow \text{count0} \rightarrow \text{count1} & \text{coin1} \rightarrow \text{head} \\ \text{inc} \rightarrow \text{count1} \rightarrow \text{count2} & \text{coin2} \rightarrow \text{head} \\ \text{inc} \rightarrow \text{count2} \rightarrow \text{count3} & \text{coin3} \rightarrow \text{tail} \\ \text{head} \leftarrow \text{op} \rightarrow \text{tail} & \text{counter} \rightarrow \text{count0} \end{array}$

P_0 and P_1 are shown in Fig. 3 and Fig. 4, respectively.

$(\text{COINX} \rightarrow \text{STATE} \text{ COINY} \rightarrow \text{STATE}) \text{ cont} \Rightarrow \text{flip} \rightarrow \text{cont}$
 $\text{counter} \rightarrow \text{COUNT}$
 $\downarrow$
 $\text{COINX} \quad (\text{I})$
 $(\text{COINX} \rightarrow \text{STATE} \text{ COINY} \rightarrow \text{STATE}) \text{ cont} \Rightarrow \text{flip} \rightarrow \text{end}$
 $\text{COINZ counter} \rightarrow \text{count2}$
 $\downarrow$
 $\text{COINZ} \quad (\text{II})$

Fig. 3 P_0 of the IGPS which represents the three coin problem.

$\text{flip} \rightarrow \text{COINX} \text{ COINX} \rightarrow \text{STATEX} \text{ STATEX} \leftarrow \text{op} \rightarrow \text{STATEY}$
 $(\downarrow) \text{ counter} \rightarrow \text{COUNTALL}$
 $\text{CONT} \text{ inc} \rightarrow \text{COUNTALL} \rightarrow \text{COUNTNEXT}$
 $\Rightarrow \text{COINX} \rightarrow \text{STATEY} \text{ STATEX} \leftarrow \text{op} \rightarrow \text{STATEY}$
 $\text{counter} \rightarrow \text{COUNTNEXT}$
 $\text{inc} \rightarrow \text{COUNTALL} \rightarrow \text{COUNTNEXT}$

Fig. 4 P_1 of the IGPS which represents the three coin problem.

In this IGPS the sub-situation σ_1 which is initially i_1 expresses a situation of coins, and the sub-situation σ_0 which is initially i_0 expresses a situation of a process of solving. Productions of P_0 generate moves which fit in the situation with referring a sub-situation σ_1 by getting knowledges of coins' situation and how many times the move is with G_1 . A production of P_1 expresses the move which is generated by P_0 . In σ_1 "inc $\rightarrow$ count0 $\rightarrow$ count1" expresses the move of a counter. And "head $\leftarrow$ op $\rightarrow$ tail" expresses that 'head' and 'tail' are opposite faces of each other.

In Table 1, we show changes of a sub-situation σ_0 of the IGPS. And in Table 2, changes of a sub-situation σ_1 of the IGPS are shown. In Table 2 we omit part of sub-situation σ_1 which do not change for making short.

Table 1 Process of changing of σ_0 .

applied	σ_0 which is the result of application of the production
I	$\text{start} \rightarrow \text{flip} \rightarrow \text{cont}$ $\downarrow$ coin1
I	$\text{start} \rightarrow \text{flip} \rightarrow \text{flip} \rightarrow \text{cont}$ $\downarrow \quad \downarrow$ $\text{coin1} \text{ coin1}$
II	$\text{start} \rightarrow \text{flip} \rightarrow \text{flip} \rightarrow \text{flip} \rightarrow \text{end}$ $\downarrow \quad \downarrow \quad \downarrow$ $\text{coin1} \text{ coin1} \text{ coin3}$

Table 2 Process of changing of σ_1 .

value of σ_1 which is the result of application of the production
$\text{coin1} \text{ coin1} \rightarrow \text{tail} \text{ coin2} \rightarrow \text{head} \text{ coin3} \rightarrow \text{tail}$ $\text{coin1} \text{ coin1} \rightarrow \text{head} \text{ coin2} \rightarrow \text{head} \text{ coin3} \rightarrow \text{tail}$ $\text{coin3} \text{ coin1} \rightarrow \text{head} \text{ coin2} \rightarrow \text{head} \text{ coin3} \rightarrow \text{head}$

3.2 Monkey/banana Problem

The familiar monkey and banana problem is formulated as an IGPS. In three coin problem, IGPS always generates correct answers, but in the present case IGPS expresses a process of monkey's trial and error process. Here, monkey

does not want to do a move which can not be carried out. And if monkey can take banana, he must take it. We show elements of the IGPS below.

$C = \{ \text{mky, box, ban, place1, place2, place3, at, over, on, has, move, push, climb, take, start, cont, cont', end} \}$,

$V = \{ \text{PLACEX, PLACEY, DO, DOX} \}$,

$R(\text{PLACEX}) = R(\text{PLACEY}) = \{ \text{place1, place2, place3} \}$,

$R(\text{DO}) = \{ \text{climb, take, move, start} \}$,

$R(\text{DOX}) = \{ \text{push, move} \}$.

P_0 and P_1 of the IGPS are shown in Fig. 5 and Fig. 6, respectively. The initial sub-situation

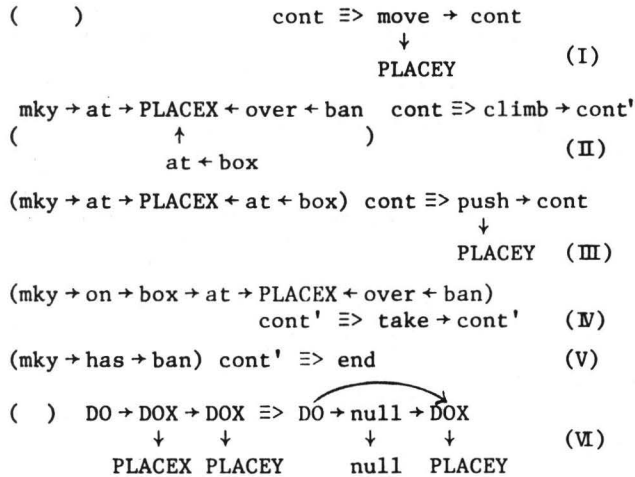


Fig. 5 P_0 of the IGPS which represents the monkey and banana problem.

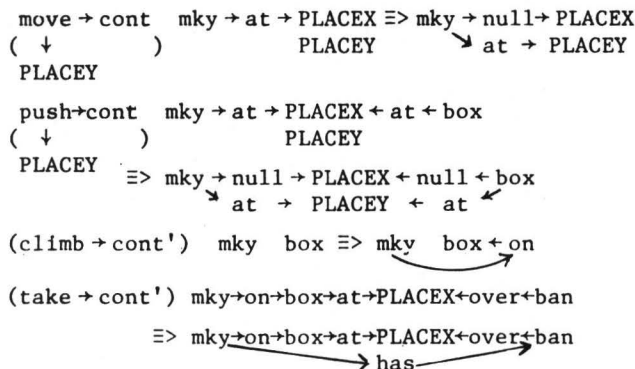


Fig. 6 P_1 of the IGPS which represents the monkey and banana problem.

i_0 is shown in Table 3, and i_1 is shown in Fig. 7. A process of monkey's trial and error are shown in Table 3.

Table 3 Process of changing of σ_0 .

applied production	σ_0 which is the result of application of the production
i_0	start $\rightarrow$ cont
I	start $\rightarrow$ move $\rightarrow$ cont $\downarrow$ place3
I	start $\rightarrow$ move $\rightarrow$ move $\rightarrow$ cont $\downarrow$ $\downarrow$ place3 place2
VI (I)	start $\rightarrow$ null $\rightarrow$ move $\rightarrow$ cont $\downarrow$ $\downarrow$ null place2
III (I)	start $\rightarrow$ move $\rightarrow$ push $\rightarrow$ cont $\downarrow$ $\downarrow$ place2 place1
III (I)	start $\rightarrow$ move $\rightarrow$ push $\rightarrow$ push $\rightarrow$ cont $\downarrow$ $\downarrow$ $\downarrow$ place2 place1 place3
VI (I, III)	start $\rightarrow$ move $\rightarrow$ null $\rightarrow$ push $\rightarrow$ cont $\downarrow$ $\downarrow$ $\downarrow$ place2 null place3

mky $\rightarrow$ at $\rightarrow$ place1 box $\rightarrow$ at $\rightarrow$ place2 ban $\rightarrow$ over $\rightarrow$ place3

Fig. 7 The i_1 of the monkey and banana problem.

4. EXECUTION OF IGPS

In this section we discuss execution of IGPS. The core of execution of IGPS is an IGPS interpreter, which receives an IGPS and execute the IGPS. And we need a production editing system for easy description of productions. The production editing system enables us to edit productions on a graphic display unit and generate sets of productions.

4.1 IGPS Interpreter

We implement an IGPS interpreter which executes an IGPS originally in accordance with the definitions of IGPS. But the IGPS interpreter has some functions for making processing fast and description of productions easy.

The IGPS interpreter has two modes except a basic mode. In the basic mode the IGPS interpreter applies one production at one time according to the definition of the IGPS. In expanded mode 1, the IGPS interpreter tries to apply productions which are permitted to apply at one time. And in expanded mode 2, the IGPS interpreter tries to apply all productions to all sub-graphs at one time while productions can be applied. In those expanded modes, the number of decisions on whether a production may be applied or not decreases, so processing time at one application of a production decreases.

In the definition of IGPS we did not define how to select a production, so we must now decide how to select a production. In most of production systems, for instance RPS [8], productions are ordered and the first production which matches the data-base is applied. But if we use rule-ordering for selection of a production, we can not describe the monkey and banana problem as in 3.2. Therefore the IGPS interpreter enables us to decide a selection method of a production among three methods. The first method is the conventional rule-ordering. The second one enables us to specify a priority of productions at each move. And the last one enables us to specify an algorithm which decides a priority of productions at each move.

4.2 Production Editing System

A production of IGPS is constructed by three tuple of labelled directed graphs. For editing productions efficiently, we need some functions, which enables us to define a set of constant C , a set of variables V and a range function R , and to input labelled directed graphs, and to check inputted productions. Input of labelled directed graphs can be done by inputting a label of each node and a tuple of a head and a tail of each edge using punched cards, but we can not inspect graphs efficiently using a list of a label of each node and a list of a tuple of a head and a tail of each edge. So the production editing system must enable us to edit productions: three tuple of labelled directed graphs, using graphical description. Therefore the production editing system uses a graphic display unit for display of productions. And the production editing system enables us to edit productions interactively.

5. CONCLUSION

In this paper, we have proposed a new production system: IGPS. IGPS is a formal system which expresses a structure and moves of a

system which has interaction. And we have shown examples of an IGPS for making clear descriptive power and moves of IGPS.

In the field of artificial intelligence, we treat a diverse collection of problems. Some of them have interaction in their situations. For expressing these problems we need a production system which can express interaction. Of course conventional production systems can express a system which has interaction. But simplicity, modularity and other favourable PS's features are lost. IGPS can express a system which has interaction while increasing favourable PS's features. And further using labelled directed graphs, IGPS can express very complex situations.

ACKNOWLEDGEMENTS

The authors appreciate many helpful conversation with several graduate students and members of Prof. Tanaka's Lab. of Osaka University.

REFERENCES

- [1] Aoki, K. and Tanaka, K., "The ability of the Interactive Systems of Context-free Form," Trans IECE, Vol. 61-D, No. 8, pp 525-532, 1978 (in Japanese).
- [2] Davis, R., Buchanan, B.G. and Shorliffe, E. H., "Production rules as a representation for a knowledge-based consultation program," AI Memo 226, Computer Science Dept., Stanford University, 1975.
- [3] Feigenbaum, E.A., et. al., "On Generality and Problem Solving - a case study involving the DENDRAL program," Machine Intelligence 6, Edinburgh University Press, pp 165-190, 1971.
- [4] Jackson, P.C., Introduction to Artificial Intelligence, Petrocilli, New York, 1974.
- [5] Moran, T.P., The Symbolic Hypothesis: A Production System Model, Computer Science Department, Carnegie-Mellon University, 1973.
- [6] Newell, A., "Production Systems: Models of Control Structures," Visual Information Processing, Academic Press, pp 463-526, 1973.
- [7] Post, E., "Formal Reductions of the General Combinatorial Problem," Am. Jnl. Math., Vol. 65, pp 197-263, 1943.
- [8] Vere, S.A., "Relational Production Systems," Artificial Intelligence, Vol. 8, No. 1, pp 47-68, 1977.