

W-JS: A MODAL LOGIC OF KNOWLEDGE

Ma Xiwen
Guo Weide

Computer Science Institute, Peking University, Beijing

ABSTRACT

W-JS is a first-order predicate calculus on the modal theory of knowledge. It is based on natural deduction rules and accompanied by possible-world-accessibility semantics. As an example, the famous "Mr. S and Mr. P" puzzle is solved in W-JS.

I INTRODUCTION

Reasoning about knowledge is one of the areas which has received the most attention from researchers in AI. J. McCarthy and others (McCarthy, et al., 1978) have given a propositional modal logic of knowledge and some examples solved in it. But it is obvious that the famous Mr. S and Mr. P puzzle (see Appendix) cannot be effectively formulated in a propositional modal logic. And, in a personal communication, J. McCarthy let us know that he had challenged the modal logicians to solve this puzzle. What we present here is a solution, we believe. This is a brief version of our paper (Ma and Guo, 1982).

II W-LANGUAGES

W-languages are extensions of the usual first-order languages.

To express the statements about knowledge, we distinguish three cases:

$S : p$ (S knows that p)

$S ! p$ (S knows whether p)

$S * c$ (S knows what c is)

where p is a wff, c is a term, and S is a subject symbol. In addition, we use

$S \% p$

to denote " S accepts p " or " S doesn't know that $\neg p$ ".

Here is a short description of the syntax of W-languages.

Symbols: A symbol is one of the following:

Subject symbols;

Constant symbols;

Concept symbols;

Variable symbols;

Function symbols;

Predicate symbols;

$=, \neg, \rightarrow, \forall, : .$

Terms: A term is one of the following:

A constant symbol;

A concept symbol;

A variable symbol;

$f(t_1, \dots, t_n)$ where f is an n -ary function symbol, and $t_1, \dots, t_n$ are n terms.

Subject-terms: A subject-term is a finite string of subject symbols.

Formulas: A formula is one of the following:

$t_1 = t_2$ where t_1, t_2 are terms;

$P(t_1, \dots, t_n)$ where P is an n -ary predicate symbol, and $t_1, \dots, t_n$ are n terms;

$\neg p$ where p is a formula;

$p \rightarrow q$ where p and q are formulas;

$\forall x p$ where p is a formula and x is a variable symbol;

$S : p$ where p is a formula and S is a subject-term.

We will use the conventional abbreviations $p \wedge q$, $p \vee q$, $p \leftrightarrow q$, and $\exists x p$. In addition, the following abbreviations are new:

$S ! p$ for $S : p \vee S : \neg p$

$S * c$ for $\exists x S : c = x$

$S \% p$ for $\neg(S : \neg p)$

We also use the conventional terminology for bound/free variables and closed formulas. In addition, the substitution $p[t/a]$ will be used for arbitrary closed formula p , term t , and constant symbol a .

The deduction rules are (in the following rules, S , S_1 are subject-terms, $p, p_1, \dots, p_n, q, r$ are closed formulas, and G, G_1 are finite sets of closed formulas):

- (1) $G \vdash p$ provided $p \in G$
- (2) $G \vdash p$ provided $G \vdash G_1, G_1 \vdash p$
- (3) $G \vdash p$ provided $G, \neg p \vdash q, \neg q$
- (4) $p, p \rightarrow q \vdash q$
- (5) $G \vdash p \rightarrow q$ provided $G, p \vdash q$
- (6) $\vdash t = t$ provided t is a variable-free term
- (7) $G \vdash \forall x p[x/a]$ provided $G \vdash p$ and x is not free in any members of G
- (8) $p[t_1/a], t_1 = t_2 \vdash p[t_2/a]$ where t_1, t_2 are variable-free terms and p is a closed formula which is both concept-free and subject-free
or

where t_1, t_2 are terms which are both variable-free and concept-free and p is any formula

- (9) $\forall x p[x/a] \vdash p[t/a]$ where t is a variable-free term and p is a formula which is both concept-free and subject-free
or

where t is a term which is both variable-free and concept-free and p is any formula

Be careful with the rules (8) and (9).

- (10) $S : p \vdash p$
- (11) $S : p_1, \dots, S : p_n, G \vdash S : q$ provided $p_1, \dots, p_n, G \vdash q$ where all formulas in G are both concept-free and subject-free
- (12) $\vdash S : S_1 : p \vee S : \neg S_1 : p$ where all subject symbols in S occur in S_1

III JS-SEMANTICS OF W-LANGUAGES

JS-semantics is a possible-world-accessibility semantics of W-languages. The details will not be presented here.

Within JS-semantics, the deduction rules are sound. But the completeness problem is still open.

IV EXAMPLE

We will solve the Mr. S and Mr. P puzzle as below.

Let the subject symbols $S_0, S_1, S_2, P_0, P_1, P_2$ denote Mr. S and Mr. P at different times, the concept symbol c denotes the pair of the selected numbers and the function symbols s, p denote the sum and product.

Thus, we have:

$S_0 * s(c)$ (At time 0, Mr. S knows what the sum is.)

$\forall x (s(x) = s(c) \rightarrow S_0 \% c = x)$ (At time 0, Mr. S only knows what the sum is.)

$P_0 * p(c)$ (At time 0, Mr. P knows what the product is.)

$\forall x (p(x) = p(c) \rightarrow P_0 \% c = x)$ (At time 0, Mr. P only knows what the product is.)

Let K_0 be the conjunction of the above four formulas. Then we have

$S_0 P_0 : K_0$ (at time 0, Mr. S and Mr. P jointly know that K_0 .)

When Mr. S said: I know you don't know what c is, but I don't know either. It is just said that

$S_0 : \neg P_0 * c \wedge \neg S_0 * c$

which we will denote by D_0 .

Thus there is a distinction between P_0 's knowledge and P_1 's, and it is just D_0 :

$\forall x (P_0 \% (D_0 \wedge c = x) \rightarrow P_1 \% c = x).$

We will use K_1 to denote the conjunction of the above formula and

$S_0 P_0 : K_0$

D_0

$P_1 * p(c)$

Thus we have

$S_1 P_1 : K_1$

Similarly, we will use D_1 to denote

$P_1 * c$

and K_2 to denote the conjunction of

$S_1 P_1 : K_1$

D_1

$$S_2 * s(c)$$

$$\forall x (S_0 \% (D_1 \wedge c = x) \rightarrow S_2 \% c = x)$$

Thus we have

$$S_2 P_2 : K_2$$

Finally, we denote

$$S_2 * c$$

by D_2 .

We will solve the puzzle by deducing a suitable first-order formula from

$$S_2 P_2 : K_2 \wedge D_2$$

First of all, we can prove

$$(SP1) \quad P_0 * p(c) \vdash \forall x (p(x) = p(c) \rightarrow x = c) \rightarrow P_0 * c$$

$$(SP2) \quad \forall x (p(x) = p(c) \rightarrow P_0 \% c = x) \vdash P_0 * c \rightarrow \forall x (p(x) = p(c) \rightarrow x = c)$$

From these we easily obtain

$$(SP3) \quad K_0 \vdash P_0 * c \leftrightarrow \forall x (p(x) = p(c) \rightarrow x = c)$$

and

$$(SP4) \quad S_0 : K_0 \vdash S_0 : \neg P_0 * c \leftrightarrow S_0 : \exists x (p(x) = p(c) \wedge \neg x = c)$$

We define $E_0(z)$ as $\exists y_0 (p(y_0) = p(z) \wedge \neg y_0 = z)$, then

$$(SP5) \quad S_0 : K_0 \vdash S_0 : \neg P_0 * c \leftrightarrow S_0 : E_0(c)$$

Proceeding as in (SP1) – (SP3), we can obtain

$$(SP6) \quad K_0 \vdash S_0 * c \leftrightarrow \forall x (s(x) = s(c) \rightarrow x = c)$$

On the other hand, we can prove

$$(SP7) \quad K_0 \vdash S_0 : E_0(c) \leftrightarrow \forall x (s(x) = s(c) \rightarrow E_0(x))$$

Thus, the ordinary first-order calculus will give

$$(SP8) \quad S_0 : K_0 \vdash S_0 : \neg P_0 * c \wedge \neg S_0 * c \leftrightarrow \forall x (s(x) = s(c) \rightarrow E_0(x)) \wedge \exists x (s(x) = s(c) \wedge \neg x = c)$$

Let $E_1(z)$ be

$$\forall y_1 (s(y_1) = s(z) \rightarrow E_0(y_1)) \wedge \exists y_1 (s(y_1) = s(z) \wedge \neg y_1 = z).$$

Then we have

$$(SP9) \quad S_0 : K_0 \vdash D_0 \leftrightarrow E_1(c).$$

Further, at time 1, we obtain

$$(SP10) \quad K_1 \vdash \forall x (P_0 \% (E_1(c) \wedge c = x) \rightarrow P_1 \% c = x)$$

and we can prove

$$(SP11) \quad K_1 \vdash \forall x (p(x) = p(c) \wedge E_1(x) \rightarrow P_1 \% c = x)$$

and then

$$(SP12) \quad K_1 \vdash P_1 * c \rightarrow \forall x (p(x) = p(c) \wedge E_1(x) \rightarrow x = c)$$

Let $E_2(z)$ be

$$\forall y_2 (p(y_2) = p(z) \wedge E_1(y_2) \rightarrow y_2 = z).$$

Then we have

$$(SP13) \quad K_1 \vdash P_1 * c \rightarrow E_2(c)$$

Using (SP9), we have

$$(SP14) \quad P_1 : K_1 \vdash P_1 : E_1(c)$$

which will deduce

$$(SP15) \quad P_1 : K_1 \vdash E_2(c) \rightarrow P_1 * c$$

thus,

$$(SP16) \quad P_1 : K_1 \vdash P_1 * c \leftrightarrow E_2(c)$$

or

$$(SP17) \quad P_1 : K_1 \vdash D_1 \leftrightarrow E_2(c)$$

Similarly, let $E_3(z)$ be

$$\forall y_3 (s(y_3) = s(z) \wedge E_2(y_3) \rightarrow y_3 = z)$$

Then we can prove

$$(SP18) \quad K_2 \vdash S_2 * c \rightarrow E_3(c)$$

and finally

$$(SP19) \quad S_2 P_2 : K_2, D_2 \vdash E_3(c)$$

where $E_3(c)$ is an abbreviation of

$$\begin{aligned} & \forall y_3 (s(y_3) = s(c) \wedge \\ & \quad \forall y_2 (p(y_2) = p(y_3) \wedge \\ & \quad \quad \forall y_1 (s(y_1) = s(y_2) \\ & \quad \quad \quad \rightarrow \exists y_0 (p(y_0) = p(y_1) \wedge \neg y_0 = y_1)) \wedge \\ & \quad \quad \exists y_1 (s(y_1) = s(y_2) \wedge \neg y_1 = y_2) \\ & \quad \quad \rightarrow y_2 = y_3) \\ & \quad \rightarrow y_3 = c) \end{aligned}$$

This is what we wanted.

APPENDIX

Mr. S and Mr. P puzzle:

Two numbers m and n are chosen such that

$$1 < m \leq n < 100.$$

Mr. S is told their sum and Mr. P is told their product.
The following dialogue ensues:

Mr. S: I know you don't know the numbers. I
don't know them either.

Mr. P: Now I know the numbers.

Mr. S: Now I know them too.

In view of the above dialogue, what are the numbers?

REFERENCES

- [1] John McCarthy, et al., "On the Model Theory of Knowledge," Memo AIM 312, Stanford University, 1978.
- [2] Ma Xiwen, Guo Weide, "W-JS: A Modal Logic about 'Knowing'," Computer Research and Development, No. 12, Vol. 19, Beijing, 1982.