構造工学論文集　Vol.35 A（1989年3月）　　土木学会

維持補修を考慮した橋梁設計のための コンサルテーションシステムの構築

A CONSULTATION SYSTEM FOR BRIDGE DESIGN CONSIDERING MAINTENABILITY

白石成人* 古田 均** 幸 和範*** 大谷裕生****
By Naruhito SHIRAISHI, Hitoshi FURUTA, Kazunori YUKI and Hiroki OHTANI

Maintenance problem has arised to be important and emergent, as a number of structural damage has occurred in recent years. From the reliability and economical point of view, it is desirable to accout for the easiness of maintenance and repair at the beginning stage of structural design. However, an adequate consideration on this matter can be achieved only when the designer has sufficient experience. In this paper, an attempt is made to develop a consultation system for the design of bridge structures considering their maintenability. The present system is a production system which consists of interpreter, rule base and working memory. It is written by FRANZ LISP and can be implemented on a 32 bit engineering workstation. Using this system, it is possible to realize a bridge design with good maintenability without difficulty. Several application examples are presented to illustrate the consultation system developed herein.

1．まえがき

構造物の損傷の増加に伴い、補修工事の困難さが表面化してきている。というのは現在の設計は、基本的には、「いかに合理的に構造物を設計することが出来るか」ということと、「いかに合理的に架設することが出来るか」という二つの点を重視しており、補修に対する考慮があまりなされていないからであると思われる。すなわち、現在の設計においては、「構造物の補修は設計の当初段階から見直すのではなく、構造物が破損した時点で補修工事をすれば良い」と考えられているふしが見受けられる。

しかし、大地震を経験した後などには、都市機能の早急な復旧の必要性から短時間に構造物を点検し、破損している部分は直ちに補修しなければならない。また、補修工事の費用は年々高まる一方である。したがって、補修工事をいかに経済的な形で実施するかという問題の解決が急務となっている。そのためには、あらかじめ補修に対する配慮を十分した上で設計を行う必要があると思われる。本研究では以上のことに鑑み、ブレインストーミング、補修マニュアルの参照、専門家との直接の対話等によって、補修を考慮した設計に対する知識を獲得した。そしてこれらの知識をコンピューターに入力蓄積するこ

* 工博　京都大学教授　工学部土木工学教室　（〒606　京都市左京区吉田本町）
** 工博　京都大学講師　工学部土木工学教室　（〒606　京都市左京区吉田本町）
*** 工修　阪神高速道路公団課長補佐　工務部工務一課　（〒541　大阪市東区北久太郎町4-68）
****京都大学大学院生　工学研究科修士課程　（〒606　京都市左京区吉田本町）

とによって、補修に対する専門的知識を十分に持っていない設計者に対して補修設計の支援を行うエキスパートシステムを構築することを試みた。本システムはいわゆるコンサルテーションシステムであり、システム構成という観点からはプロダクションシステム[1]に分類される。本システムを用いることによって、設計者が設計データをコンピューターに逐次入力すれば、補修設計の観点に立った種々の問題点を把握することが出来る。コンサルテーションシステムとしては大型計算機を用いるのではなく、利用者が容易に扱えることが望ましいので、本研究では32bitエンジニアリング・ワークステーション上にシステムを構築した。そのために既に京都大学の大型計算機上にUTILISPで作成されたシステムをワークステーション上のFRANZ LISPに移植した。また本システムを実際の設計例に適用し、その有効性について検討した。

2. 補修を考慮した設計に関するブレインストーミングとその結果

ブレインストーミングとは、A.F.Osborn[2]によって提案された手法で、数人の専門家が1つのテーブルを囲んで座り、1つのテーマが与えられ、その中でリラックスした雰囲気の下で自由に発言してもらうものである。ただし、下記の点に注意しなければならない。

①テーマを一つに絞り、分かりやすい言葉で述べる。
②他人のアイディアを絶対に批判してはならない。
③自由奔放なアイディアを歓迎する。
④アイディアの量は多い方がよい。
⑤他人のアイディアをヒントにしてそれをさらに改善したり組み合わせたりすることを奨励する。

ブレインストーミングの内容は全てテープに記録し、後にそこから問題点を抽出しＫＪ法によってグループ化した。ＫＪ法[3][4]とは川喜田二郎の発案による問題の枠組み発見の手法であって、先ず情報を1行見出しにしてカードに記し、それを机上に広げて全体を眺め、親近性を感じるカードをグループ化してサブ問題を合成して行くのがその原理である。本研究でブレインストーミングを行った理由は、設計時において補修に関する配慮をすることの必要性は認められているもののその実際面での問題点および補修設計の考え方は未だ明らかになっていないと考えるからである。すなわち、具体的な知識獲得の段階では、一人の専門家に詳細なインタビューを行うことになるが、全体的な枠組みは多くの人間による議論を通じて構築することが望ましいと思われる。

本研究では阪神高速道路公団の4名の専門家を招いてブレインストーミングを行った。その結果、補修を考慮した設計において主に注目しなければならないのは、次の2つの項目であることがわかった。

①維持管理用空間の確保
②維持管理し易い構造

そこで阪神高速道路公団の専門家の助言と補修設計マニュアル[5]により、それぞれに必要な具体的な対策についての知識を獲得した。以下にその代表的なものを示す。

・点検や工事用の足場を確保する。

①鉄道や重要な道路上の構造物、航路上の構造物、他の営造物に隣接した構造物には、作業空間、足場空間および余裕空間を考え合わせた設計が必要である。（例えば阪神高速道路公団では、桁下空間（橋梁構造物の桁下から建築限界上端までの距離）は2.1m以上とるべきであるという規定がある。これは桁に補修用の足場をつける事を前提に決められたものである。）
②桁の点検のための高さを確保する必要がある（1桁の桁高は1.5m以上にしなければならないという規定がある）
③美観を考慮して、桁に設置する点検用検査路が桁の下縁を越えないようにする必要がある。

④排水管の水の流れをスムーズにするには桁高を高くする必要がある。

・維持管理しやすい構造を採用する。

①主桁腹板の切り欠きや、架け違いの構造は補修が困難であるので避けるべきである。

②天端面積の小さな橋脚も杳の補修が困難になるので避けるべきである。

③近年、鋼の材質が向上し、小さくスレンダーで応力的には問題がないような構造物が建　設可能となったので、構造物の応力チェックだけではなく、点検や維持補修が可能かどうかを十　分に検討する必要がある。

いずれにしても橋梁を設計する場合に、補修を考慮する必要があるといえよう。

3.　維持補修を考慮した設計のためのコンサルテーションシステム

本コンサルテーションシステムは、プロダクションシステムによって構築されている。プロダクションシステム[1]とは図1のように、推論機構であるインタープリタと、ルールベースとワーキングメモリー（データベースとも言う）より構成されている。ルールベースは「if・・・then・・・」の形で書かれているルールの集合である。ルールベースはあらかじめ構築したシステムに対して後にルールを追加しやすくするためにインタープリタとは独立した構造になっている。また、ユーザーが入力したデータはワーキングメモリーで管理される。つまり、インタープリタはルールベースとワーキングメモリーから、然るべき結論を導き出す働きをする。

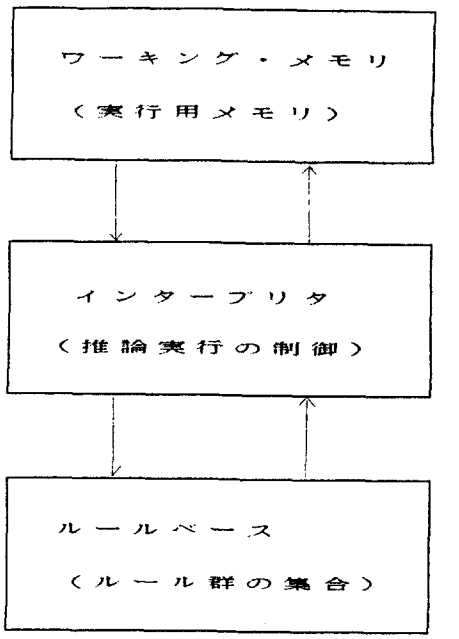

図1　システムの構成

ところが、このような複雑な動きをするプログラムを作成する場合、例えばFORTRANのようなコンパイル言語ではルールを逐次追加していくことは困難である。たとえその問題が解決したとしても、それらの言語は記号処理が不得意である。しかも、仮にそれによってシステムが完成したとしても、インタープリタとルールが独立した構造にはならず、後から新たなルールを追加するためにはプログラムそのものの構造を変更しなければならない。つまり、FORTRANやBASICは数値計算に関しては非常に都合の良い言語であるけれども、記号処理を主とするプログラムには適さない。それに対して記号処理言語LISPは複雑な数値計算は不得意であるが、記号処理には非常に適している。したがって、本研究ではLISPを用いてシステムを構築する。

ところで、コンサルテーションシステムは設計者が手軽に利用できなければ意味を持たない。誰にでも簡単に、しかもいつでも利用できなければならない。そこで本研究ではこの点に着目し、京都大学大型計算機センターのUTILISP[6]によって書かれたプロダクションシステム・インタープリタ[7]をエンジニアリング・ワークステーション上のFRANZ LISP[8]に移植することを試みた。LISPにはMAC LISPやINTER LISP、その他にもCOMMON LISP, FRANZ LISP, UTILISP, UCI LISP ・・・・など数多くの種類があり、異種の言

語間ではソフトウエアの互換性がないことから、システムの移植は全て手作業で行わなければならない。
この移植したプロダクションシステム・インタープリタは 主に以下の働きをする。

①変数$rulesに代入されるルール群の中から、ifの部分が変数$working-memoryに代入されているワーキングメモリーのデータとマッチするルールを選択する（マッチング操作）
②マッチするルールのthen部を実行する。
③実行が終ると再び①に戻る。

したがって、ルールの中では$working-memoryと$rules以外はインタープリタを意識しなくてもよく、随時新たなルールを追加することが出来る。ルールの構造は前述の通りif部とthen部とからなっており、具体的には次の形をしている。

（rules 'name '（（rule-number if・・・then・・・）（rule-number if・・・then・・・）・・・・・・・・・））ここでnameはルール群名で、インタープリタで定義されている関数であるrulesによって$rulesに代入される。したがって、インタープリタは評価を与える関数evalによって、つまり（eval $rules）によってルール群を知ることが出来る。またrule-numberはルール番号であり、自由につけてよい。if部の・・・は次のような形で書かれる。

①一つずつの条件名を（）でくくる。
②二つ以上条件がある場合は、（a）（b）（c）・　・のように続けて書く。
③条件を書く順序はインタープリタが判断してくれるので関係ない。

また、then部における・・・は次のような形で書かれる。

①関数名の頭部にアスタリスク"*"をつけること。
②関数はLISPのシステムで定義されている関数でも自分で定義した関数でもよい。
③クオート「'」(quote)は省略する。

つまり、then以下の内容は（＊関数名　引数）の形を取ることになる。本システムではデータはこのルールのthen部でユーザーに質問する形式をとっており、それによってワーキングメモリーに蓄積されていく。

さて、本システムのルールに用いた知識は、

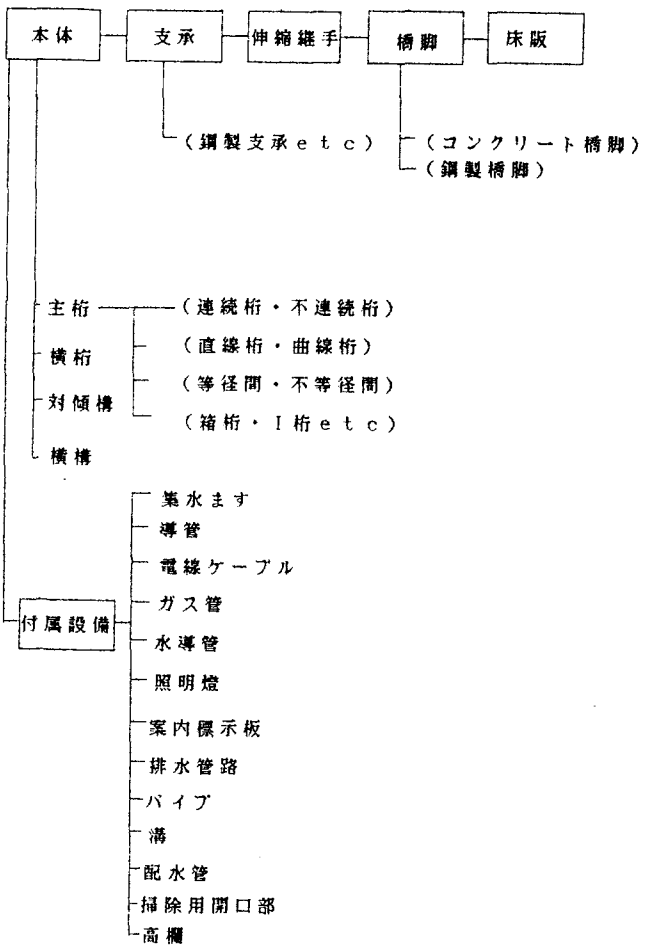

図２　本システムにおける橋梁の分類

まず補修マニュアル[5]等を参照し、さらに阪神高速道路公団の専門家との対話を通じて獲得した。これらの知識を図2のように階層的に分類して、ユーザがこのシステムを使用し易いようにルールを構成した。この際、以下のことに注意を払った。

①ルールベースを橋梁の各部分毎に分類し、ユーザーがシステムを利用し易いように整理した。

②同じくユーザーが利用し易いように漢字で画面出力をするようにした。

③新たにルールを追加し易くするために、ルール本体と日本語文字を取り扱った部分を分割した。

④ユーザーが新たな関数を容易に定義出来るように、専用のファイルを用意した。

⑤少しでも実行速度が向上するように、多くのルールベースを構成して、それを効果的に利用した。

⑥number-okなる関数を定義することによって、数値条件そのものを簡単にルールに書き込めるようにした。

①ではシステムがユーザーに対して、まず橋梁のどの部分について知りたいのかを数段階に分けて質問してくる形式を採用し、その選択はアルファベット記号で入力する方式をとることにした。そのためには、まずダミールールを設定しなければならない。これは(rules kyouryou '((rule1 if (dum)then (*run))))で書かれたルールである。ここで第一アトムのrulesはルール名kyouryouにクオート以下のリストを代入する関数である。また、then部の(*run)は関数の定義ファイルで定義された関数であり、ユーザーに対して橋梁のどの部分に対してコンサルティングを行うのかを質問し、そのルールに制御を移動する関数である。また、システム起動前に初期値として$rules には(kyouryou)を$working-memory には(dum)を代入しておかなければならない。この時にインタープリタは、「kyouryou」というルールベース中のif部が(dum)であるルールのthen部(run)を実行する。

②ではプロダクションシステムの使用性に注目し、日本語文字列を用いたシステムを構築するためにエンジニアリング・ワークステーションのsj（日本語を使用するためのソフトウエア）を起動し、日本語編集用のエディターであるjviによってプログラムを編集した。ただし、sjで使用可能な日本語文字が直ちにLISPで使用できるわけではなく、LISPのモードで使用不可能な文字も存在するので注意が必要である。

③では、ルールのthen部に質問事項や忠告事項を書くことはプログラム編集上好ましくないことであるので、それらの日本語文字部分は別のファイルで取り扱うことにした。日本語文字列をアルファベットの変数に代入することによってルールベースの編集の簡略化を計ったものである。

⑤では、インタープリタは一つのルールベースについてif部をマッチングさせていくわけであるから、ルールベース数を増加させることによって一つのルールベース当りのルール数を減少させ、それによって演算実行速度を高めようとしている。

⑥では数値条件、例えば「x が 1.2m 以上である」を(ge x 1.2)と書き表し、そのままルールで使用できるようにnumber-okなる関数を定義し、ルールが編集し易くなるように考慮した。

以上のようにいくつかの点について改良を試みたものの、まだ以下に示すようにいくつかの問題点が残されている。

①ルールの数が増え過ぎると、ルールの組合せを考えることにより、ルールを追加するための労力が大きくなる。

②ルールの数が増え過ぎると、それにほぼ比例して実行速度が遅くなる。

以上の問題点を将来、少しでも改善して行くことが望まれる。

4. 適用例

本節ではコンサルテーションシステムの適用例を2例示す。

まず、図3はある現存する箱桁橋梁である。そしてその橋梁の断面図中に円で囲んだ部分（沓）の拡大図をその上に示している。この橋梁の主桁に関するコンサルテーションシステムを実行したものを図4に示す。まずシステムを起動するとa)床版～f)付属設備が表示されるのでｃの主桁を選択する。このとき図5に示される「橋梁」というルール群の中でマッチング操作を行うことにより、rule-3のthenの部分を実行する。つまりルール群名を「本体」に変えることによって、以後のマッチング操作は「本体」というルール群の中で行われる事になる。この操作は図2に示す知識ベースの配分にしたがって推論を行う。次の段階では、a)主桁～nil)前画面、と表示されるのでa)の主桁を選択する。ただしこの画面で、nilを選択すると前画面のa)～f)の表示に戻り、再度ルール群「橋梁」に戻ることが出来る。そこで、主桁を選択すると、例えば次のrule-4にマッチし、そのthen部を実行する。ここで関数ansは引き数のうちでどの値をとるかを質問する関数である。また、number-ansは引き数の値をユーザに質問する関数である。

```
(rules '本体 (•••••(rule-4 if (主桁)
                    then (*ans '連続である '連続でない)
                         (*ans '直線桁である  '曲線桁である)
                         (*ans '等径間である '不等径間である)
                         (*number-ans 'boxの幅は何メートルですか)
                         (*number-ans 'boxの高さは何メートルですか)
                         (*number-ans '桁下空間は何メートルですか)
                         (*number-ans 'シュウの高さは何メートルですか) )••••)
```

rule-4を実行することによって、主桁は連続桁であるか連続桁でないかを聞いてくるので連続桁を選択する。ここで、同様の操作によりさらに、直線桁か曲線桁か（この橋梁は直線桁である）、等径間か等径間でないか（この橋梁は等径間）等、順次質問をしてくる。最後に、ｂｏｘの幅（3.3m）、高さ（2.186m）、桁下空間、沓の高さ（0.165m）などを答えると、質問した値を逐次ワーキングメモリーに書き込んでいくので次の推論を実行することが出来る。やはりマッチング操作は図5の「本体」というルール群の中で行われる。その結果rule-65のthen部が実行される。ここで関数prineは、図6で定義される文章を表示する関数である。この橋梁の例では特に沓の高さが問題になっている事がわかる。設計者はこの結果を見て橋梁の設計をもう一度検討してみる必要がある。これは、沓の高さが低いために補修のときにジャッキを入れることが出来ない例である。この橋梁特有の問題点以外に、この場合はまず構造細目等でいままでに多くの損傷を呈している構造に対する注意事項が示されている。その後点検、検査を容易にするための構造に対する示唆と、補修を容易にするための準備、腐食等に大きな影響をもつ水回りに対する注意事項が示されている。すなわち、このタイプの橋梁に対する一般的な注意点を示してコンサルテーションは終了する。

次に別の橋梁例について考えてみる。図7にその断面図を示す。これらの図から桁下空間などの必要な数値を計算し、コンサルテーションを実施すると、図8の結果が得られる。やはりここでも主桁についてのコンサルテーションを実行することにする。まず前例と同様にa)床版～f)付属設備の表示が表れるのでｃの「本体」を選択する。このとき図5のルール群「橋梁」の中でマッチング操作が行われ、rule-3のthen部である(*change-rule '本体)を実行する。ここでａの主桁を選択すると次に連続性、直線性、径間、桁の種類、架線の有無、断面内部にはいれるかどうかなど順次質問してくるのでそれに対して答えていく。するとシステムは前例と同様にマッチング操作をルール群「本体」の中で行い、rule52のthen部を実行する。つまり、図6で定義される文章番号を関数prineの引き数とすることによって、その文章を表示することになる。その結果、桁下空間が少なすぎるということがわかり、設計者はこれを参考に、さらに他の注意事項を考慮してもう一度設計を考えなければならない。その他一般的な注意事

項として、前の例で示された項目以外に、止水工の設置、負反力が生じない構造の奨励、ダイアフラムに対する注意点等が表示される。

a：床版
b：支承
c：本体
d：伸縮継ぎ手
e：橋脚
f：付属設備

cを選択すると

a：主桁
b：横桁
c：その他
ｎｉｌ：前画面へ

aを選択する。

a：連続桁である
b：不連続桁である

aを選択する

a：直線桁である
b：曲線桁である

aを選択する

a：等径間である
b：不等径間である

aを選択する

a：箱桁
b：合成Ｉ桁
c：その他

aを選択する

a：架線がある
b：架線はない

bを選択する

a：断面内部に入れる
b：入れない

aを選択する

boxの幅は何mですか?
3.3
boxの高さは何mですか?
2.816
桁下空間は何mですか?
(ここでは不明なので、入力しない)
シュウの高さは何mですか?
0.165
(ここで3秒の間があく)
・シューの高さが低いので、問題があります！！
・車の車輪の軌跡の近くに主桁を配置する必要があります
・主桁に段差を設けないようにしてください
・マンホールを各橋脚近辺に最低１カ所もうけてくださ
い。また、マンホールが橋脚からはなれたところにある
場合は橋脚までの検査路を設ける必要があります。
・ジャッキアップのときの横方向移動に対する横方向移
動制限装置も併せて設置してください。
・現場継手部には止水口を施し、排水口及び換気孔を設けてください

図４　本システムの適用例（１）

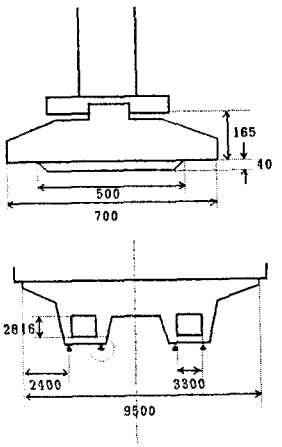

図３　適用した橋梁（１）

a：床版
b：支承
c：本体
d：伸縮継ぎ手
e：橋脚
f：付属設備

cを選択すると

a：主桁
b：横桁
c：その他
ｎｉｌ：前画面へ

aを選択する。

a：連続桁である
b：不連続桁である

aを選択する

a：直線桁である
b：曲線桁である

aを選択する

a：等径間である
b：不等径間である

bを選択する

a：箱桁
b：合成Ｉ桁
c：その他

aを選択する

a：架線がある
b：架線はない

bを選択する

a：断面内部に入れる
b：入れない

aを選択する

boxの幅は何mですか?

2.3

boxの高さは何mですか?

1.05
桁下空間は何mですか?

0.39

シュウの高さは何mですか?

(ここでは不明なので入力しない)

(ここで3秒の間があく)

・桁下空間を2.1m以上にする必要があります。！！
・床版端部に二重の止水工を設置する必要があります。
・ジャッキアップ用補強をおいて下さい。
・車の車輪の軌跡の近くに主桁を配置した
ほうが良いと思われます。
・マンホールを各橋脚基部に最低１個設け、
端部に設置するのはできるだけ避けて下さい。
また、雨水や鳥などが入らないように注意して下さい。
・ジャッキアップのときの横方向移動に対する
横方向移　動制限装置も併せて設置してください。
・少なくとも負反力が発生しないような設計
をする必要があります。
・現場継手部には止水工を施し、排水口及び
換気工を設けて下さい。
・中空断面に入ることが出来る場合にはダイヤフラム
に少なくとも0.5m×0.6mの大きさのマンホールを設置し、
そのコーナーには丸みをつけて下さい。
中空断面の入口は閉鎖できるようにして下さい。

図８　本システムの適用例（２）

```
(rules '橋梁
    (rule-1 if   (床版)
            then (*change-rule '床版))
    (rule-2 if   (支承)
            then (*change-rule '支承))
    (rule-3 if   (本体)
            then (*change-rule '本体))
    (rule-4 if   (伸縮継手)
            then (*change-rule '伸縮継手))
    (rule-5 if   (橋脚)
            then (*change-rule '橋脚))
    (rule-6 if   (付属設備)
            then (*change-rule '付属設備)))


(rules '本体
    (rule-1 if **************
            then ************)
              .
              .
    (rule-52 if (主桁)(連続)(直橋)(不等径間)(箱桁)(架橋なし)(断面入)
              (box幅  =x)(box高さ =y)(桁下空間 =z)(壁の高さ =w)
              (=ge =x 1.5)(=ge =y 1.5)(=ge =z 2.1)(=ge =w 0.5)
             then (*prine r5 r11 r12 r14 r19 r35 r38 r37 r48)
                  (*change-rule 'start))
              .
              .
    (rule-65 if (主桁)(連続)(直橋)(等径間)(箱桁)(架橋なし)(断面入)
               (box幅  =x)(box高さ =y)(桁下空間 =z)(壁の高さ =w)
               (=ge =x 1.5)(=ge =y 1.5)(=ge =z 2.1)(=ge =w 0.5)
            then (*prine r49 r14 r18 r19 r35 r37)
                 (*change-rule 'start)))
```

図５　本システムのルールの例（１）

```
              .
              .
(setq r5 '|桁下空間を２．１ｍ以上にする必要があります。|)
              .
              .
(setq r11 '|床版端部に二重の止水工を設置する必要があります。|)
(setq r12 '|ジャッキアップ用補強をおいて下さい。|)
              .
              .
(setq r14 '|車の車輪の軌跡の近くに主桁を配置したほうが良いと思われます。|)
              .
              .
(setq r18 '|主桁に段差を設けないようにしてください|)
(setq r19 '|マンホールを各橋脚近辺に最低１カ所もうけてください。また、マンホール
が橋脚からはなれたところにある場合は橋脚までの検査路を設ける必要があります。|)
(setq r20 '|マンホールを各脚柱基部に最低１個設け、端部についたマンホールは出来る
だけさけてください。また、雨水や鳥などが入らないように注意してください。|)
              .
              .
(setq r35 '|ジャッキアップのときの横方向移動に対する横方向移動制限装置も併せて設
置してください。|)
(setq r36 '|少なくとも負反力が発生しないような設計をする|)
(setq r37 '|現場継手部には止水口を施し、排水口及び換気孔を設けてください。|)
              .
              .
(setq r48 '|中空断面に入ることが出来る場合にはダイヤフラムに少なくとも０．５ｍ×０．
６ｍの大きさのマンホールを設置し、そのコーナーには丸みをつける。中空断面の入口は
閉鎖できるものとする|)
(setq r49 '|シュウの高さが低いので問題があります！！|)
              .
              .
```

図６　本システムのルールの例（２）

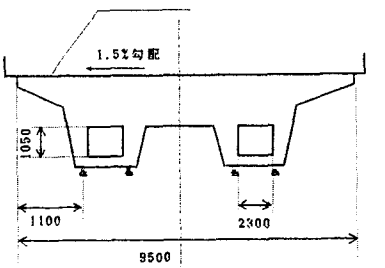

図７　適用した橋梁（２）

5．　結論および今後の課題

本研究では、橋梁を設計する際に補修を考慮する必要があるかどうか、またあるとすればどのような点に注意するべきであるかを検討した。まずブレインストーミングを行い、補修という観点からみた現在の橋梁の問題点を抽出した。それによると、補修を考慮していない橋梁構造物は数多く存在し、場合によっては補修工事が非常に困難な橋梁も存在することが判明した。そしてそれらの橋梁のほとんどが、維持管理用の空間を確保していないか、または維持管理し易い構造になっていないかのどちらかであることが分かった。

そこで、従来の補修設計マニュアルを整理し、専門家の助言を得てその知識を補足することによって、補修を考慮した設計に関する知識を表すプロダクションルールを作成した。そして、それをコンピューターに入力することによって、コンサルテーションシステムを構築した。本システムはプロダクションシステムとして作成されており、推論機構であるインタープリタと知識ベースが独立しているために、簡単にルールの追加修正を行うことが出来る。

いまだ、補修に関する知識が系統だっておらず、断片的なものであることを考えると、現時点ではプロダクションシステムを用いることがエキスパートシステム作成に対して最良であると考えられる。しかも本研究では、京都大学大型計算機センターのUTILISPによって構築されたプロダクションシステム・インタープリタを、エンジニアリング・ワークステーション上のFRANZ LISPに移植した。したがって、本研究のコンサルテーションシステムを、エンジニアリング・ワークステーション上で構築することができた。この事によって、システム開発者にとってもユーザーにとっても、これまでより自由で快適な環境のもとでシステムを使用できるようになった。

本研究で構築したプロダクションルールは、設計マニュアルなど、既存のルールを主に用いたためにその質・量ともに不十分なものであった。したがって、将来橋梁構造物を設計する場合、補修に関わるルールをさらに充実させ、体系的な知識にまとめることが必要である。

参考文献

1）安部憲広・滝　寛和：エキスパートシステム入門 、共立出版(1986)
2）寺野寿朗：システム工学入門、pp. 58-71、共立出版(1986)
3）日本能率協会：グループKJ法入門、松尾隆編(1973)
4）白石、古田、尾崎：都市高速道路の安全性・信頼性の概念について、第14回安全工学シンポジウム、pp. 285-291, 1987
5）阪神高速道路公団；維持管理を考慮した鋼構造物の計画と設計・施工(1985)
6）富士通：UTILISP マニュアル(1984)
7）白石、古田、馬野、川上：ＲＣ床版の耐用性評価システムに関する基礎的研究、土木学会論文集、第386号/I-8 、pp. 285-291、1987.
8）R. ウィレンスキー著　平林真一・河田亭・世古忠／訳：LISP CRAFT上・下、講談社(1987)

(1988年10月12日受付)