KNOWLEDGE-BASED SYSTEM FOR SOIL IMPROVEMENT

By J. L. Chameau,[1] Member, ASCE, and J. C. Santamarina,[2] Associate Member, ASCE

ABSTRACT: The practice of soil improvement has received renewed attention in the last two decades. As a result, existing methods were improved and new ones developed. This paper presents a knowledge-based decision support system for the selection of soil improvement methods, Improve. The system uses a knowledge representation structure based on "windows" together with a best-first search algorithm. In this context, windows are mathematical representations of the restrictions to the values variables can take, combining the knowledge and its uncertainty in a unique entity. This form of knowledge representation has many advantages and allows for procedures not available in other systems, such as the development of composite solutions, the use of different evaluation functions, the search for lacunae, and the case-based representation of knowledge.

INTRODUCTION

Soil improvement is an old geotechnical practice. Its objective is the modification of the properties of the soil, including deformation, strength, permeability, and chemical characteristics to better adapt the soil to given needs. This field has received renewed attention in the last two decades, resulting in technical advances both by improving existing methods and by developing new ones, including soil reinforcement, geotextiles, dynamic consolidation, and jet grouting.

There are numerous methods and combinations of methods used to improve soils. The selection of the proper technique is dictated by case-specific conditions. Unfortunately, the ranges of applicability of the methods with respect to the relevant parameters are not clearly defined nor are most values of the parameters that characterize a given project. This is not unusual in the geotechnical engineering field, in which experience often plays a vital role. The nature of the soil improvement problem and the paucity of experts make this domain attractive for the development of a knowledge-based system. This paper presents a prototype knowledge-based decision support system, Improve, designed to help geotechnical engineers select soil improvement techniques.

CLASSIFICATION SYSTEMS USING WINDOWS

The selection of the best solution, or a group of potential solutions, from a set of alternatives can be seen as a classification problem. Many geotechnical engineering tasks conform to this model, such as the selection of foundation type, drilling or exploration equipment, and soil improvement method. Soil classification and mineral identification are also classification tasks. A

---

[1] Assoc. Prof., School of Civ. Engrg., Purdue Univ., West Lafayette, IN 47907.

[2] Asst. Prof., Dept. of Civ. and Envir. Engrg., Polytechnic Univ., Brooklyn, NY 11201.

Note. Discussion open until December 1, 1989. To extend the closing date one month, a written request must be filed with the ASCE Manager of Journals. The manuscript for this paper was submitted for review and possible publication on June 14, 1988. This paper is part of the ***Journal of Computing in Civil Engineering***, Vol. 3, No. 3, July, 1989. ©ASCE, ISSN 0887-3801/89/0003-0253/$1.00 + $.15 per page. Paper No. 23647.

new structure for classification systems (Santamarina 1987) is used in this paper to develop a knowledge system for the selection of soil improvement methods. The basic concepts behind this form of classification system are simple yet powerful when compared to more classical structures for knowledge representation.

A "window" is a restriction to the set of possible values a variable can take. In most cases, the boundaries of windows cannot be defined with certainty and there is a gradual transition between values that are possible and those that are not. This transition may truly represent one's perception or may be the result of vagueness in the available information. So defined, windows are fuzzy sets (Brown and Yao 1983). However, the term "window" relates better to the screening process modeled in Improve. The theory of fuzzy sets is not needed to understand the information discussed in this paper.

A window is a list of possibility numbers that characterize an object with respect to the variable of interest. In this paper, possibility is quantified between 0.0 and 1.0. For example, the window for the permeability of a sand could be defined as follows:

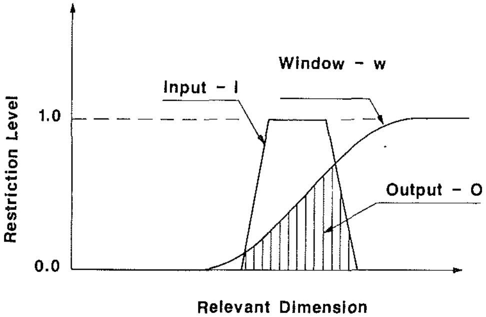
Permeability of a sand (k):
Centimeters per second: $(10^{-8}\ 10^{-7}\ 10^{-6}\ 10^{-5}\ 10^{-4}\ 10^{-3}\ 10^{-2}\ 10^{-1}\ 10^{0}\ 10^{1}\ 10^{2})$
Window: (0.0 0.0 0.0 0.4 0.8 1.0 1.0 0.8 0.4 0.0 0.0)

This window indicates that a material with a permeability of 10^{-3} cm/s definitely belongs to the category "sand" (acceptabilty = 1.0), while there is a lesser possibility that a material with $k = 10^{-1}$ cm/s belongs to that group (acceptability = 0.80).

The permeability of a soil *A* may not be precisely known. Then, deciding whether soil *A* is a sand involves "filtering" the input information about *A* through the window for sands and comparing the output to the input. If the input and the output are identical, it is concluded that *A* is a sand. If they are not identical, the level of similarity is a measure of how much soil *A* can be considered a sand. Alternative mathematical techniques can be used for filtering and for comparing output with input. Minimization and the ratio of cardinalities were selected to model these processes (Dubois and Prade 1980; Zimmermann 1985). Fig. 1 illustrates these operations and shows that a decision is made by determining the percentage of the input that is contained in the window. A numerical example follows:

Centimeters per second: $(10^{-8}\ 10^{-7}\ 10^{-6}\ 10^{-5}\ 10^{-4}\ 10^{-3}\ 10^{-2}\ 10^{-1}\ 10^{0}\ 10^{1}\ 10^{2})$
Input K_A: (0.0 0.0 0.6 1.0 0.3 0.0 0.0 0.0 0.0 0.0 0.0)
Window K_{sand}: (0.0 0.0 0.0 0.4 0.8 1.0 1.0 0.8 0.4 0.0 0.0)
Output: (0.0 0.0 0.0 0.4 0.3 0.0 0.0 0.0 0.0 0.0 0.0)
Cardinalities: $fc_{\text{input}} = 0.6 + 1.0 + 0.3 = 1.9.$
$fc_{\text{output}} = 0.4 + 0.3 = 0.70.$
Acceptability: $fc_{\text{output}}/fc_{\text{input}} = 0.7/1.9 = 0.37.$

A cardinality ratio ($fc_{\text{output}}/fc_{\text{input}}$) or acceptability value of 1.0 indicates that the input is fully compatible with the window, while 0.0 shows total rejection. In this formulation, the input must be normalized to 1.0. Other limitations and the axiomatic justification for the choice of operators are discussed in Santamarina (1987).



$$\text{Acceptability} = \frac{\text{Cardinality O}}{\text{Cardinality I}} = \frac{\text{Area under O}}{\text{Area under I}}$$

FIG. 1. Acceptability—Ratio of Cardinalities

Each soil improvement method can be defined by relevant dimensions, i.e., the physical characteristics and parameters that determine the applicability of the method. The windows for each dimension form a "stack of windows" that represents the method. Given a project, its characteristics are compared with the stack of windows for a soil improvement method *M* to decide whether *M* is adequate for the project. An acceptability value is calculated for each dimension and the smallest acceptability value is selected as a measure of the adequacy of the method *M* for the project. This process is repeated for all methods and the one with highest overall acceptability is selected.

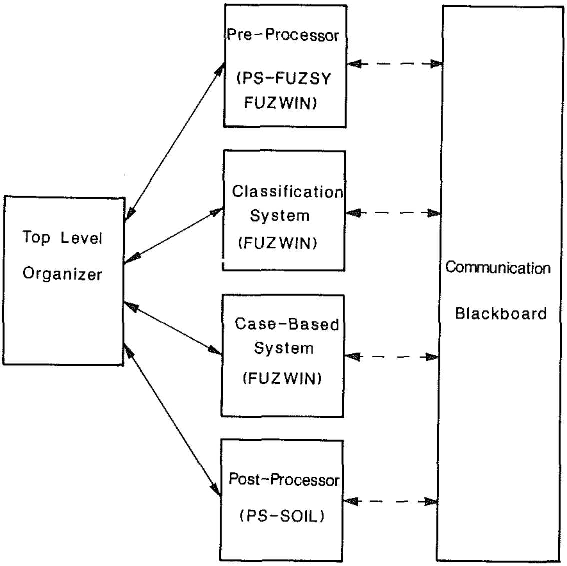


FIG. 2. Schematic Representation of Search Algorithm

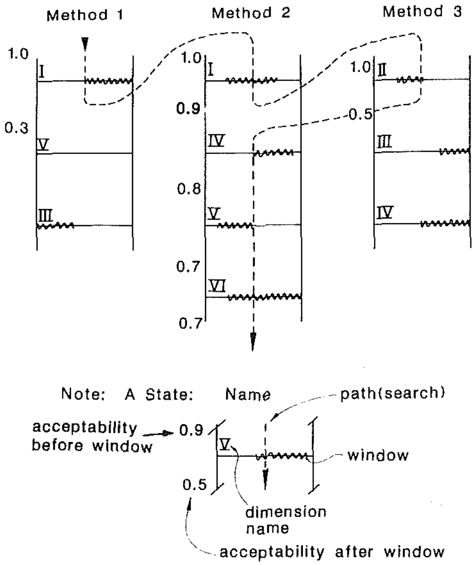


FIG. 3. Improve—General Organization

This selection process can be efficiently implemented using the best-first search algorithm (Nilsson 1980). In this case, the consideration of a method is abandoned as soon as any of its dimensions shows poor agreement with the characteristics of the project and search continues with other methods that show better matching. This algorithm is used in Improve. During the search, the system keeps track of the most critical dimension in each stack and its similarity value (Fig. 2).

IMPROVE: KNOWLEDGE AND STRUCTURE

The system consists of four parts: preprocessor, selection of methods, selection of similar cases, and postprocessor. All parts conform to the same format and a common storage "blackboard" is used for communication (Fig. 5).

Preprocessing

The preprocessor helps the user decide whether a soil improvement method is necessary, based on a single stack of eight windows. When the user response is in linguistic form, an internal translator is used to convert it to the numerical form of windows. Translation is frequently used in applications of fuzzy sets theory. The translator in Improve is based on empirical observations of the variation in fuzziness between extremes (Santamarina and Chameau 1987).

Selection of Best Method(s)

There are currently more than 40 methods in the data base. Most of them consist of a unique procedure; however, common combinations are also included (Appendix I). Each soil improvement technique is represented by a stack of windows. Only those dimensions that restrict the use of a method are included in the stack. Therefore, the number and the type of dimensions vary among the methods. The dimensions that are common to most stacks are listed in Appendix II. It is interesting to note the specificity of these dimensions in comparison to the general guidelines for the selection of soil improvement methods provided in the literature (Mitchell 1981; Welsh 1987). The experience with Improve showed that in most cases the relevance of a dimension cannot be established until it is tested on actual cases. A typical stack of windows is shown in Appendix III. Electro-osmosis is represetned by 11 dimensions and their corresponding windows. The windows are defined with respect to scales that are built in the system.

The search of the stacks of each method starts with the information that was provided by the user during preprocessing. Additional information is requested from the user as needed. The search results in the selection of the best alternative. The user may decide to continue the search to identify other methods with close acceptability values. All methods found are listed in the blackboard.

Checking Repository of Cases

A case history can be described in the same dimensional space where the methods are defined. Then, case histories can be represented by stacks of windows, and the same search algorithm may be used to select the case histories that best resemble the project. Improve incorporates a data base of case histories. The soil improvement methods used in those cases are potential alternatives for the project under study. At present, the repository of cases in Improve contains 50 case histories that were collected during system development. The representation of one of these cases is shown in Appendix IV.

Postprocessor

The postprocessor is a rule-based system that provides final information and suggestions. Design and construction guidelines for the selected soil improvement methods are retrieved from a data base, Soil. This data base consists of information pages at the nodes of a tree structure that facilitates access and search.

Concluding Consultation

When the postprocessor operations are concluded or whenever the user stops the system, control is returned to the computing environment. The user may then continue the session using other available procedures. For example the user may ask why a certain method was not selected, check the information that was provided to detect erroneous input, repeat the run modifying the input, use the Soil data base to study alternative soil improvement methods, or independently search the repository of cases.

Sample Run

Excerpts of a session with Improve are presented in Appendix V. The case under consideration is a 15-year-old embankment dam that is part of a major hydroelectric system. Increased seepage indicates the possibility of internal erosion taking place. The purpose of the consultation with Improve is to seek guidelines for potential repairs.

COMMON PRACTICE AND TECHNICAL CONSTRAINTS

The main body of knowledge in Improve was acquired from R. D. Holtz. A combination of the "interval estimation" technique (Chameau and Santamarina 1987) with "domain description" (Hart 1986) was used. Expertise available in the literature was included.

Holtz also assisted in performance-feedback sessions, providing example cases and evaluating the reasoning and performance of the system (Gaines 1986; Waterman 1986). The first session was particularly illuminating. Improve provided satisfactory answers when common cases were tested. However, when more atypical problems were considered, the results were very poor (e.g., 480-km-long channel for water, in collapsible soil). The data base of methods was carefully scrutinized to conclude that windows were biased by common practice. For example, it had been considered that the maximum depth for the method of mix-in-place piles is about 20 m according to common practice; however, there is no real technical limitation for much deeper piles. In fact, Japanese experience shows mix-in-place piles as deep as 60 m and more. Further analysis showed that the evaluation of complex cases requires the systematic consideration of *technical limitations*. In the case of soil improvement, common practice poses additional constraints. Mathematically speaking, windows for common practice are subsets of windows for technical constraints.

All the windows in the data base of methods were reviewed to eliminate unjustified constraints. For example, windows for the "size of the area that can be treated" were removed from most stacks and relaxed in the remaining ones (i.e., wider windows). A total of 60% of the windows were relaxed. This is a measure of the additional limitations imposed by common practice in a given environment and may be one of the leading causes for nonoptimal decisions. A stack of windows based on common practice is compared to the stack for the same method based on technical constraints in Appendix VI.

Further testing of the system using case histories showed that some suggested methods were technically feasible but not commonly used in U.S. practice. This was not considered a negative feature. In fact, it is likely that such a system stimulates the user's creativity and results in better decisions.

FEATURES IN IMPROVE

Improve has features that are common to other knowledge-based systems, such as explanation capabilities; however, their implementation is very different as a result of the window form of knowledge representation. In addition, the window system allows for unique capabilities that are discussed next.

Combining Constraints

The data base of methods in Improve includes some composite solutions like "preloading and drains" and "heavy tamping and vibratory rollers." The user may also suggest the combination of other methods.

Given two soil improvement methods M_1 and M_2, the stack of the composite solution $M_{1 \text{ and } 2}$ is obtained by propagating the constraints (windows) that define each method. Constraint propagation may involve minimization, maximization, or compensatory combination. For example, given the depths that can be treated with vibratory rollers (VR) and with heavy tamping (HT), the depth that can be effectively treated by the combination of the two methods HT and VR is obtained by maximization:

Depth that can be treated:

Scale (m)	(0	1	3	5	9	15	27	45	80)
Depth—HT	(0	0	0	0.5	1	1	1	0.6	0)
Depth—VR	(1	1	0.2	0	0	0	0	0	0)
Depth—HT and VR	(1	1	0.2	0.5	1	1	1	0.6	0)

The experience with Improve showed that the selection of combination operator depends not only on the methods to be combined, but also on the characteristics of the project under consideration. The "combine" procedure that was implemented is interactive: the user selects the combination operator for each dimension; Improve generates the stack of windows for the composite method and checks its acceptability for the project under consideration. The new stack can be added to the data base of methods and becomes another alternative for subsequent runs.

The pseudocombination of methods using logical AND-OR operators is an option in some expert systems. However those approaches are less far-reaching than the method just described. The conjunctive combination of two alternatives, X-AND-Y, does not imply the conjunction of all its defining dimensions, x_i-AND-y_i. Otherwise, the composite solution would be at least as restrictive as the most restrictive component, defeating the purpose of composite alternatives. The procedure discussed above has the disadvantage of being interactive; however, it provides the flexibility needed to formulate realistic composite solutions.

Validation and Lacunae

The window form of knowledge representation facilitates the search for gaps in knowledge. A procedure "lacunae" was developed to check gaps in single dimensions and in the conjunction of any two. The search of lacunae with respect to any two attributes, e.g., "type of project" and "special geotechnical conditions," is equivalent to asking: "Is there any project for which there is no soil improvement method? If there is none, then, is there any project and a special geotechnical condition for which there is no soil improvement method?"

The most important consequence of the search for lacunae was system validation, i.e., finding missing or incorrect information in the knowledge base during the develoment of Improve. The current version of the data base of soil improvement methods was searched for lacunae in several dimensions and in their conjunctions. These dimensions include type of project, special geotechnical conditions, capacity to reduce deformation, particle size, depth of the layer to be improved, environmental freedom, and time available. The following observations were made:

1. One dimension at a time: (1) The extremes of some dimensions reflect the extent of applicability of soil improvement methods, e.g., depth about 80 m and

size of area to be treated of about 100,000 km^2 (some case histories exceed these values); (2) there are minor gaps close to the extremes of some dimensions such as "reducing deformation" or "increasing strength;" these gaps support the use of soil improvement as a solution where the situation is "bad," but not "extremely bad;" and (3) most gaps were minor indicating that there are soil improvement methods for most conditions when one dimension prevails.

2. Conjunction of two dimensions: (1) Current soil improvement methods produce at least minor environmental effects, particularly those oriented to improving deep layers and dispersive or collapsible soils; and (2) there are difficulties in implementing a soil improvement plan in very short time periods, especially if the purpose is to reduce deformations, if the project is a tunnel, or if the soil is dispersive or swelling.
3. More than two dimensions: Improve allows to search for gaps when more than two dimensions are involved by responding "unknown" to all but the questions that pertain to the dimensions under study. In this case, a large number of gaps surface. This indicates that the state of the art is inefficient when projects have several relevant constraints (e.g., complex profile, strict environmental restrictions, and limited execution time).

The search for lacunae shows that it is possible to learn from the development and use of knowledge-based systems in civil engineering.

Evaluation Function

A geotechnical expert's comprehension of a problem is affected by a large number of factors, including those that are case-specific, context-dependent, and subjective. These factors deeply affect the subject's perception of the problem and the evaluation of the relevant parameters (dimensions). For example, given two different cases for soil improvement, the same individual may concentrate on one attribute in one case but consider all relevant dimensions in the second one. It is known that the choice of either one of the evaluation functions, "dimensionwise" or "holistic," is task and individual dependent, but it is not possible to predict the evaluation function a decision maker will select in a given event (Wallsten 1980).

Knowledge is represented with windows in three different parts of Improve, allowing for the comparative study of evaluation functions. Performance testing showed that a holistic averaging model was most appropriate for the preprocessing stage. This evaluation function averages the acceptability of all dimensions, reducing the importance of individual conditions while focusing on the overall picture exhibited by the data. This is a forgiving model; for example, even if the design has been "independently verified" but all other dimensions strongly indicate the need for soil improvement, then Improve suggests soil improvement with high possibility.

The constraint satisfaction methodology used in the selection of soil improvement methods is consistent with the thresholding holistic evaluation function. In this evaluation mode, the decision is controlled by the most critical dimension in the stack. The same evaluation function was used for the selection of case histories; however, the choice in this case is less obvious.

Case-Based Representation

It has been observed that experts make decisions based upon the recollection of previous cases. This is very relevant in geotechnical engineering since so much emphasis is placed on experience.

Improve allows the user to incorporate new cases in the repository of cases; a project studied with the system can be automatically saved in this data base at the end of a session. Increasing the size of the repository of cases increases the likelihood that the system will suggest adequate solutions. As the repository of cases enlarges, it becomes possible to request the system to extract the windows that characterize a given method using constraint propagation. This process may take place with experts and resembles learning by proceduralization.

Representing knowledge through case histories in a computerized system allows students and engineers to practice and develop their expertise. They can study the cases, compare their own decisions to that of the system, and ask for explanations and justifications. Facilitating and improving the transfer of knowledge from experienced individuals to students may be the most rewarding aspect of knowledge-based systems in coming years.

FINAL COMMENTS AND CONCLUSIONS

Knowledge-based systems are part of the technical response to the increased demand for tools to handle information. When well designed, they can help bring the state of the art to practice. These systems can also help to train professionals, to recognize gaps in knowledge, and to transfer the knowledge and accumulated experience of a few to a large number of practitioners.

Not every experiential decision process in civil engineering is ready to be distilled into a decision support system. Soil improvement is particularly attractive: it is a well-defined domain, the selection of methods is well determined by the job characteristics and the required soil improvement, documented cases exist, and qualitative variables enter the decision process.

The window form of knowledge representation incorporates the knowledge and its uncertainty in a unique entity. This structure facilitates the use of different evaluation functions, the development of composite solutions, the search for lacunae, and the case-based representation and accumulation of knowledge.

Appendix I. METHODS INCLUDED IN IMPROVE

- Densification blasting.
- Blasting and vibratory rollers.
- Vibratory probe.
- Vibratory probe and vibratory rollers.
- Vibrocompaction.
- Vibrocompaction and vibratory rollers.
- Compaction piles.
- Heavy tamping.
- Heavy tamping and vibratory rollers.
- Vibratory rollers.
- Preloading.
- Preloading and drains.
- Surcharge fills.
- Surcharge fills and drains.

- Dynamic consolidation.
- Electro-osmosis.
- Drains.
- Particulate grouting.
- Chemical grouting.
- Pressure injected lime.
- Displacement grout.
- Electrokinetic injection.
- Jet grouting.
- Remove and replace.
- Admixture stabilization.
- Displacement blasting.
- Prewetting loess.
- Prewetting swelling clay.
- Structural fill.
- Lightweight fill.
- Mix-in-place piles.
- Mix-in-place walls.
- Heating.
- Freezing.
- Stone columns.
- Root piles.
- Soil nailing.
- Strip reinforcement.
- Moisture barriers.
- Geotextiles.
- Berms.

APPENDIX II. DIMENSIONS COMMON TO MOST METHODS

- Type of project.
- Environmental freedom.
- Time available.
- Importance of increasing strength.
- Importance of reducing deformation.
- Importance of modifying permeability.
- Position (depth) of the layer.
- Distance to the neighbor/layer depth.
- Structure width/layer depth.
- Special soil type.
- Particle size.
- Relative density.
- Saturation conditions.
- Stratum (covered or uncovered).
- Stage (built or not-built).
- Is surface above water?
- Is surface treatment possible?
- Is layered construction possible?
- Duration of improvement (permanent or temporary).
- Equipment particular to each alternative.
- Materials required by each method.

APPENDIX III. STACK OF WINDOWS FOR ELECTRO-OSMOSIS

Size area: (1 1 1 1 0.9 0.8 0.5)
Time available: (0 0 0 0.4 0.6 0.8 0.9 1 1)
Increase stength: (1 1 1 1 1 1 1 0.9 0.7)
Reduce deformation: (1 1 1 1 1 0.9 0.7 0.5 0.3)
Lower permeability: (1 1 0.8 0.3 0 0 0 0 0)
Depth—H: (1 1 1 1 1 1 0.9 0.7 0.5)
Special case: (0.9 0 0 0 0 1)
Particle size: (0.3 1 1 0.6 0 0)
Saturation: (1 0.3 0)
DC power and wiring: (0 0.1 0.2 0.4 0.6 0.8 0.9 1 1)
Anodes and cathodes: (0 0.1 0.2 0.4 0.6 0.8 0.9 1 1)

Internal Scales

Size of the area that can be treated:

Square meters:	100	500	1,000	5,000	10,000	5,000	100,000
Window:	1	1	1	1	0.9	0.8	0.5

Depth that can be treated:

Meters:	0	1	3	5	9	15	27	45	80
Window:	1	1	1	1	1	1	0.9	0.7	0.5

APPENDIX IV. STACK OF WINDOWS FOR CASE HISTORY

Case 2 is in Bulgaria. They needed to support structures' footings on loess. The decision was to use heavy tamping dropping 70 kN concrete weights from 10 m. The improvement affected maximum depths of 5–6 m.

Project: (1 0 0 0 0)
Environmental freedom: (0 0 0 0 0.1 0.6 1 1 1)
Time available: (0 0 0.2 0.8 1 1 1 1 1)
Increase strength: (1 1 1 1 0.6 0.1 0 0 0)
Reduce deformation: (1 1 1 1 1 1 0.6 0.1)
Lower permeability: (1 1 0.4 0 0 0 0 0)
Special case: (0 0 0 1 0 0)
Depth—H: (0.5 1 1 0.9 0.4 0 0 0 0)
D-neighbor/H: (1 1 0.2)
Particle size: (0.4 0.7 1 1 0.8 0.3)

APPENDIX V. IMPROVE—SAMPLE RUN (USER'S RESPONSES ARE IN ITALICS)

Do you need instructions (yes/no)?
→ *no*

. . . . .

Question: problems-with-permeability
Format: linguistic: xl vl ql sl in sh qh vh xh
→ *very high*

. . . . .

Question: design-was-independently-reviewed

```
Format:  linguistic: xl vl ql sl in sh qh vh xh
→  very high
I believe that you should consider soil-improvement with a possibility of
  0.78
More specific questions about the project. Please, indicate the type of
  project or structure that forces the need for soil improvement. Choose
  among: foundation, tunnel, excavation, fill, other.
→  why
Because each type of project has its own characteristics which condition
  the applicability of some methods. Note: Embankments are included
  under "fill," while their foundations should be listed as "foundation."
  Your answer, please:
→  fill
. . . . .
Question:  lower-permeability
Format:  linguistic: xl vl ql sl in sh qh vh xh
→  very high
Question:  depth-H
Format:  (0 1    3 5 9 15 27 45 80 m)
→  (0 0.4 1 1 1  1  1  1  1)
Question:  special-case
Format:  organic swell dispersion collapse liquefaction other
→  dispersion
. . . . .
Question:  tanks-agitators-injector
Format:  linguistic: xl vl ql sl in sh qh vh xh
→  unknown
The selected method is:  jet grouting
With an acceptability of:  0.91
Most critical condition:  depth-H
Another solution . . . (yes/no):  yes
Question:  is a surface-treatment possible
Format:  linguistic: xl vl ql sl in sh qh vh xh
→  unknown
Question:  available materials for a barrier
Format:  linguistic: xl vl ql sl in sh qh vh xh
→  why
I am considering moisture-barriers, with a current acceptability of 0.9.
  Your answer, please:
→  very high
The selected method is:  moisture-barriers
With an acceptability of:  0.9
Most critical condition:  size-area
Another solution . . . (yes/no):  yes
. . . . .
Question:  is a drill-rig available
Format:  Linguistic: xl vl ql sl in sh qh vh xh
→  unknown
The selected method is:  mix-in-place-walls
With an acceptability of:  0.9
Most critical condition:  size-area
```

```
Another solution . . . (yes/no):  yes
Question:  stage
Format:  built  not-built
→  built
. . . . .
I am considering chemical-grouting, with a current acceptability of 0.6.
  Your answer, please:
→  unknown
. . . . .
The selected method is:  chemical-grouting
With an acceptability of:  0.6
Most critical condition:  special-case
Another solution . . . (yes/no):  no
Would you like to see similar case histories (yes/no)?  yes
I remember a case in New Mexico:  Navajo Dam. The Bureau of
  Reclamation decided to control the permeability of the potentially
  dispersible core with a concrete wall built by Soletange. The maximum
  depth of treatment was 100 m.
The resemblance with your case is:  1.0
The most different condition is:  none
Another memory (yes/no):  yes
Question:  saturation
Format:  (saturated  partially-saturated  dry)
→  (1 0.3 0)
I remember the case of an embankment that failed on first filling because
  of dispersion. They use gypsum (and considered lime) for the repairs
  and for the treatment of the reservoir water.
The resemblance with your case is:  0.56
The most different condition is:  time-available
Another memory (yes/no):  no
Would you like information on the selected methods?  yes
About jet grouting (yes/no)?  yes
Jet grouting:
Design (may be used to form cut-off walls)
  Diameter < 3 m (usual is < 1 m)
  Grout:  rapid set cement 1:1 to 1:10 fly ash − w/c = 1 − 2
  Nozzle pressure:  15 to 70 MPa—may add air jetting
  Withdraw:  0.1 to 0.5 m/min (see r:011)
Control
  Control grout—check grouting parameter during construction
  Record grout take—test recovered samples
  May use standard penetration test, cone penetration test, pressuremeter,
    and shear wave velocity and load tests.
  Walls: measure permeability
  Columns:  measure load-deformation capacity
  Monitor constructed facility
Cost:  $250–650/m^3
Related cards:  c:077
References:  r:010  r:0.11
. . . . .
About moisture barriers (yes/no)?  no
```

About mix-in-place walls (yes/no)? *no*
About chemical grouting (yes/no)? *no*
Final recommendations:
Recall some action may be needed to reduce environmental effects.
Recall modifications to the construction or soil improvement procedures may be needed to reduce time.

. . . . .

APPENDIX VI. COMMON PRACTICE VERSUS TECHNICAL CONSTRAINTS

Stack Based on Common Practice

Electro-Osmosis
Project: (1 0.7 0.8 0.3 1) (*d*)
Size area: (0.9 1 1 0.9 0.8 0.5 0.3) (*r*)
Time available: (0 0 0 0 0.1 0.5 0.9 1 1) (*r*)
Environmental freedom: (0.2 0.5 1 1 1 1 1 1) (*d*)
Increase strength: (1 1 1 1 1 1 1 0.9 0.7)
Reduce deformation: (1 1 1 1 1 0.9 0.7 0.5 0.3)
Lower permeability: (1 1 0.8 0.3 0 0 0 0 0)
Depth—*H*: (1 1 1 1 1 1 0.7 0.2 0) (*r*)
Special case: (0.6 0 0 0 0 1) (*r*)
Particle size: (0.3 1 1 0.6 0 0)
Saturation: (1 0.3 0)
DC power and wiring: (0 0.1 0.2 0.4 0.6 0.8 0.9 1 1)
Anodes and cathodes: (0 0.1 0.2 0.4 0.6 0.8 0.9 1 1)

Stack Based on Technical Constraints

Electro-Osmosis
Size area: (1 1 1 1 0.9 0.8 0.5)
Time available: (0 0 0 0.4 0.6 0.8 0.9 1 1)
Increase strength: (1 1 1 1 1 1 1 0.9 0.7)
Reduce deformation: (1 1 1 1 1 0.9 0.7 0.5 0.3)
Lower permeability: (1 1 0.8 0.3 0 0 0 0 0)
Depth—*H*: (1 1 1 1 1 1 0.9 0.7 0.5)
Special case: (0.9 0 0 0 0 1)
Particle size: (0.3 1 1 0.6 0 0)
Saturation: (1 0.3 0)
DC power and wiring: (0 0.1 0.2 0.4 0.6 0.8 0.9 1 1)
Anodes and cathodes: (0 0.1 0.2 0.4 0.6 0.8 0.9 1 1)

Note: r = window was relaxed; and d = window was deleted (i.e., fully relaxed).

APPENDIX VII. REFERENCES

Brown, C. B., and Yao, J. T. P. (1983). "Fuzzy sets and structural engineering." *J. Struct. Engrg.*, ASCE, 109(5), 1211–1225.

Chameau, J. L., and Santamarina, J. C. (1987). "Membership functions I: Comparing methods of measurement." *Int. J. Approximate Reasoning*, 1, 287–301.

Dubois, D., and Prade, H. (1980). *Fuzzy sets and systems*, Academic Press, New York, N.Y.

Gaines, B. (1986). "An overview of knowledge acquisition and transfer." *Proc. Knowledge Acquisition for Knowledge-Based Systems Workshop*, Banff, Alta., Canada.

Hart, A. (1986). *Knowledge acquisition for expert systems.* McGraw-Hill Book Co., New York, N.Y.

Mitchell, J. K. (1981). "Soil improvement, state of the art report." *Proc. 10th Int. Conf. on Soil Mech. and Found. Engrg.*, 4, 509–565.

Nilsson, N. J. (1980). *Principles of artificial intelligence*, Tioga Publishing Co., Palo Alto, Calif.

Santamarina, J. C. (1987). "Fuzzy sets and knowledge systems in geotechnical engineering," thesis presented to Purdue University at West Lafayette, Ind., in partial fulfillment for the requirements for the degree of Doctor of Philosophy.

Santamarina, J. C., and Chameau, J. L. (1987). "Membership functions II: Trends in fuzziness and implications." *Int. J. Approximate Reasoning*, 1, 303–317.

Waterman, D. A. (1986). *A guide to expert systems.* Addison Wesley, Reading, Mass.

Wallsten, T. S., ed. (1980). "Process and models to describe choice and inference behavior." *Cognitive processes in choice and decision behavior*, Erlbaum Assoc., Hillsdale, N.J.

Welsh, J. P., ed. (1987). "Soil improvement—A ten year update." *Geotechnical Special Publication No. 12*, ASCE, New York, N.Y.

Zimmermann, H. J. (1985). *Fuzzy set theory and its applications.* Kluwer-Nijhoff, Boston, Mass.