A Pragmatic Approach To Mathematical Programming :
Reusing Knowledge Embedded In The Corporate Database

박 성주, 권 오병, 이 창희
(한국과학기술원 경영과학과)

요즘 의사결정지원체제에서 모형화 지원에 관한 연구가 활발해지고 있다. 모형화 지원에 있어서 데이타베이스는 모형화를 위한 기본 자료의 저장소의 역할만을 할 뿐 아니라 모형화를 위한 지식이 내재되어있다. 특히 지식베이스가 구축되지 못한 환경에서 모형화를 할 때에는 데이타베이스에 있는 지식을 활용하는 것이 유용하다. 따라서 이 논문에서는 데이타베이스의 내재된 지식으로부터 수리모형을 유도해내는 과정을 객체지향적 데이타베이스를 예로 하여 보이고자 한다. 이의 해결을 위해 General Intelligence 와 Fit 이론을 사용하였다.

I. Introduction

In decision support systems, the overall perspective about database has been a primitive base for problem solving or information processing, and it was sometimes adopted in model management systems as alternative model representation or manipulation methodology such as model query language or model as data perspectives. However, *the perspectives overlooked most of the hidden domain knowledge in database and made it kept idle. In fact, the database already contains useful information for model building, e.g. object, relationship, set, data type, index structure, unit of data, and so on. It is more important when 1) the information system is data-oriented and 2) the decisionmaker has database only and knowledge base is not available. It is fact that until today a decisionmaker wouldn't have knowledge base separately.*

Thus, our motivation is to improve the usage of available domain knowledge embedded in database which would enhance the modeling power of mathematical programming. We focus on the above issue and are willing to extract the knowledge which is embedded in database. Then, we will restructure the knowledge in a compatible form with mathematical programming.

To perform it, our solution is a two-phased modeling from data model shown in Fig. 1. In this paper, we will shortly introduce phase I and then consider phase II.

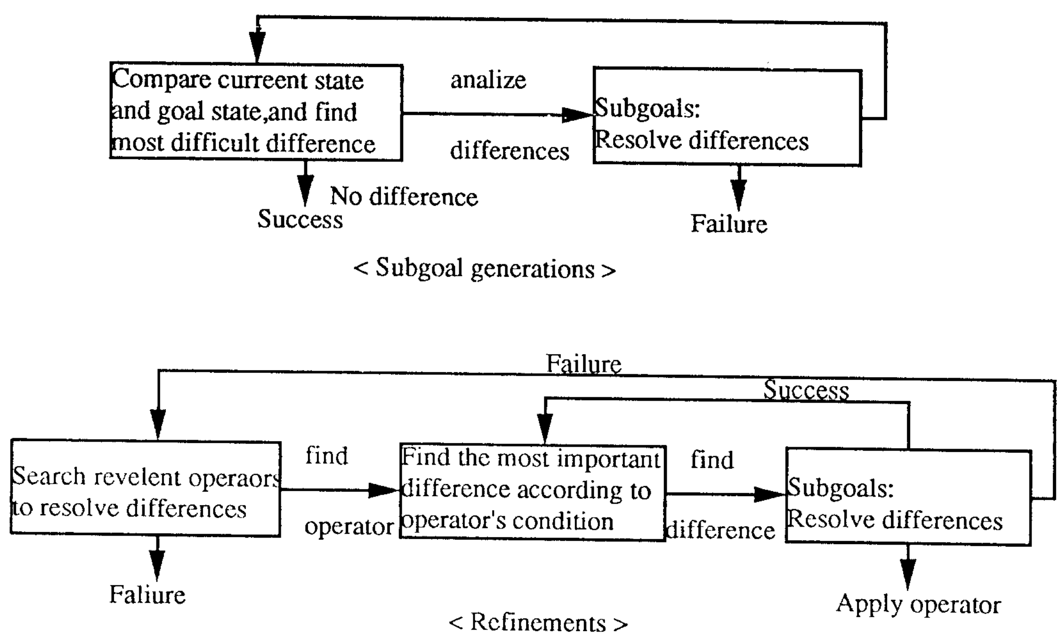
II. Iris: The Object-Oriented Data Model

One of the most frequent domains using mathematical programming is production planning. Typical example Object-Oriented model for the production planning is shown in Fig. 1 and if the model is implemented in Iris, a prototype OODBMS, the example can be shown as Fig. 2.

We can extract the followings from real corporate database.

Data Element = < Object, Index, DataType, Unit >

If we rearrange the definition as the form of data element, then it will be as PQUANT = < PRODUCT, k, Quantity, null >.



< Fig. 1 Two-Phased modeling >

III. Phase I: General Intelligence modeling for plausible LP model

The key activities of general intelligence are (1) analyzing current state

```
CREATE TYPE Product
  ( name Charstring REQUIRED,
    p-quantity Quantity,
    index k,
    unit null,

CREATE TYPE Quantity SUBTYPE OF integer ) ;
```

< Fig. 2 An Example >

to a series of differences and decomposing them into a set of subproblems (2) and selecting relevant operators to remove the gap. During the problem solving processes, they require a set of initial state, goal state, and operators. We can define a problem P in a form of quadruple as follows: [1]

$P = < I(s), G(s), M, S >$

where, I(s) : initial state

G(s) : goal state

M : Operator

S : a set of state

The term strategy in this paper means that a series of activities to build LP models in plausible forms. It allows problem definition, analyze differences and apply modeling engines. The typical problem solving steps are typically problem identification --> subproblem generation --> difference resolution. For LP problem we suggest the following steps.

[PROBLEM IDENTIFICATION]

[1] Define problem as (Min or Max) + (Object) + (Attribute) where (Attribute) is identified as optimizing variable.

[2] Determine decision variable(s) as (Object) + (Attribute).

[3] Identify decision variable as Initial state and optimizing variable as Goal state in forms of <Entity, Data type, Index, Unit >

[SUBPROBLEM GENERATION]

[4] Analyze differences between optimizing variable and decision

variable(s).

[5] Enumerate the candidates to resolve the differences. Identify them

	I(s)	$I(s) \cap G1(s)$		$I(s) \cap G1(s) \cap \ldots \cap Gn(s)$	G(s)
Characteristic Differnce	CI	CI1	..	CG	CG
Structural Difference	SI	SI1	..	SG	SG
Unit Difference	UI	UI1	.	UG	UG
Operators		O1	..	On	

Deduced Constraint : $I(s) \cap G1(s) \cap \ldots \cap Gn(s)$ .BOUND. G(s)

where, Ci : data type of ith state
Si : index set of ith state
Ui : unit of ith state
Oi : operator applied in ith state

< Fig. 3 Strategy Table >

as RHSs.

[6] Decompose problem into subproblems and each of them candidates for RHS and decision variable(s).

[DIFFERENCE RESOLUTION]

[7] For all subproblems analyze differences. (The difference priority is Characteristic difference, Structural difference and Unit difference)

[8] Apply relevant modeling engines.

[9] Perform [7] and [8] until the following conditions are satisfied.

(1) If no more differences are found, then stop. The subproblem is solved.

(2) If any differences are found and no relevant modeling engines can be applied, then stop. The subproblem is unsolvable.

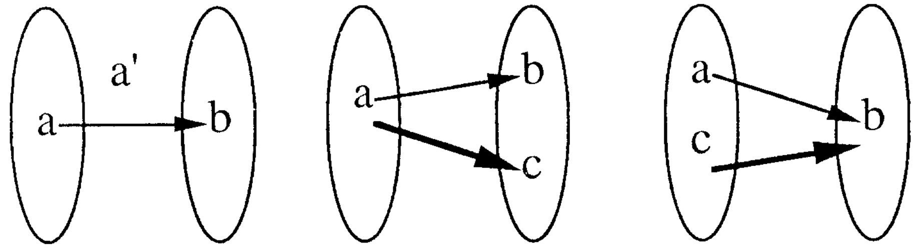
Procedure to deduce plausible is described in Fig. 3. When deduced constraint set is made, there remains two operations, deciding (in)equalities (≥,≤, or =) and instantiation through database interface or external user inputs, to complete a plausible model. As for deciding (in)equalities, since knowledge in data dictionary does not contain these kinds of information, decision maker should determine the (in)equalities. Once they are acquired externally, they can be stored for later use. Storing facility is omitted in this paper, because it misses main stream.

Model instantiation has been already considered such as matrix generation. After the plausible model is completed, decision maker can test whether the model is fit to real situations. If it does not fit, then we can think two possibilities, (a) errors in estimating parameter values or wrong data inputs and (b) unfound model structure that is not captured by data models because the relationships which are not necessary in corporate database. Regarding (b), the structure of plausible model should be changed through user intervention. Thus, we use the theory of fit to generate scenarios toward a true model. This is another phase of modeling supports.

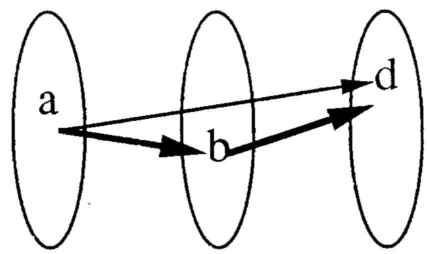
IV. Phase II: The theory of fit - model validation toward true model

In modeling support system, it is so difficult to affirm model validity. In fact, it is almost impossible for open system like modeling support system which would interact with users to guarantee validity completely. [2] One of the typical efforts for model validity function is to test feasibility by using past data [3], and specially test by causal modeling or path analysis. [4] Because the activity of Phase I modeling is only to reconstruct the datamodel as LP model form, it is true that the phase does not take into account the semantic knowledge which is not acquired in the data model (dictionary). In view of modeling support facilities, their main concern is not to make true models, but to help decision makes searching possible relationships between decision variable(s) and other attribute variables and hence give implications for true models.

The concept of fit has served as an important building block for theory construction in various areas of research. It can generate other possible conditions or relationships that have not been considered. To define the fit precisely, there are several alternative perspectives of fit in six types, fit as moderation, mediation, matching, Gestalts, profile deviation, and covariance. [5] Except of fit as interaction and matching that give few meaningful implications, four concepts of fit are available and make it possible to generate alternative structures to be changed toward true model.



(a)File Profile Deviation (b) Fit as Gestalts (c) Fit as covariance



(d) FIt as Mediation

< Fig. 4 Model validation >

Since the decision making to select appropriate structures is wholly by decision maker, the testing procedure is an interactive mode with him.

4.1 Fit as Profile Deviation: Intervention of attributes that have null data type

The idea of profile deviation is to a) develop an ideal profile, b) add differential weights and c) use a baseline model. If data type of a variable(coefficient) is null, then we think that no Characteristic differences are detected. For instance, PQuantjk and SQuantkm have data type of 'Quantity' and hence GPS-based modeling will conclude that the constraint is $\Sigma_j PQuant_{jk} \geq \Sigma_m SQuant_{km}$. But if the domain has special semantic constraints about this relationship like "PQuant should be greater than two times of SQuant", then the decision maker should update constraint as $\Sigma_j PQuant_{jk} \geq 2^* \Sigma_m SQuant_{km}$. Because the coefficient "2" has no data types, it is not detected in the modeling engine for Characteristic difference. Thus, a validity check should provide the scenario of the entrance of coefficients or parameters with no data types.

4.2 Fit as Gestalts: Interaction within RHSs

"Fit as Gestalts" is the activity to clustering several variables into a concept. The term Gestalts is relatively few and very different from one another, both in terms of the scores of, and relationships among, variables. [6] It implies in modeling that if some RHSs are interrelated and can be compressed,then they can be congruenced in one constraint. For instance, as shown in Fig.4(b), if there are two relationships, LHS 'a' and candidate for RHS 'b' and 'c', then it is possible there is unknown interaction between b and c. Having no semantic knowledge, the GPS-based modeling will deduce two constraint block, $AX \geq b$ and $AX \geq c$, but in fact the true constraint might be a form of $AX \geq f(b,c)$. For example, if a company produces products and sells them for domestic and exports, then the GPS-based modeling which might has no information about selling strategy would conclude that two constraints, $\Sigma_j PQuant_{jk} \geq \Sigma_m DOM\_SQuant_{km}$ and $\Sigma_j PQuant_{jk} \geq \Sigma_m EXP\_SQuant_{km}$, but the decisionmaker can know that DOM_SQuant_{km} and EXP_SQuant_{km} are not independent and it should be updated as $\Sigma_j PQuant_{jk} \geq \Sigma_m DOM\_SQuant_{km} + \Sigma_m EXP\_SQuant_{km}$.

4.3 Fit as Covariance: Insertion of other decision variables

"Fit as covariance" is something like factor analysis. A bundle of variables is grouped by a new variable. The fit is required when there are internal inconsistencies. Another possibility is the insertion of unconsidered decision variables as in Fig. 4(c). It is possible that other variables would affect RHS. For example, if selected decision variable is $PQuant_{jk}$ but a company meets the demand not only produces product directly but also sublets externally, then deduced plausible constraint $\Sigma_j PQuant_{jk} \geq \Sigma_m SQuant_{km}$ should be changed as $\Sigma_j PQuant_{jk} + SubContract_k \geq \Sigma_m SQuant_{km}$ because a new variable $SubContract_k$ entered into the constraint.

4.4 Fit as Mediation: Intervention of new decision variables

"Fit as mediation" specifies a significant intervening mechanism between an antecedent variable and the consequent variable. It specifies the existence of intervening (indirect) effects between an antecedent variable and its consequent variable as in Fig. 4(d). For instance, if a decision variable is PQuant and RHS candidate is Demand, then the plausible model block would be $\Sigma_j \Sigma_k PQuant_{jk} \geq \Sigma_m Demand_m$. In fact, however, SQuant would directly affect Demand and PQuant only to affect indirectly but rather affect SQuant.Thus, the new variable SQuant intervenes and divide the plausible model into a) $\Sigma_j PQuant_{jk} \geq \Sigma_m SQuant_{km}$ and b) $\Sigma_k Squant_{km} \geq Demand_m$.

V. Conciuding Remarks

The paper began with the condition that database system has been constructed in corporate environment before knowledge base is developed. Under the condition it is so perplex and expensive to provide another information repository only for mathematical programming, and hence, practical modeling supports should be able to utilize the database which plays a role of alternative knowledge base. To realize this, General Intelligence and the theory of fit are applied.

We expect that our approach to extract MS/OR modeling knowledge from database compatible with mathematical programming can make it possible to apply advanced Decision Support Systems.

[References]

[1] GÛVENIR, H.A. and G.W. ERNST, "Learning Problem Solving Strategies Using Refinement and Macro Generation", Artificial Intelligence, pp. 209 - 243, 1990.

[2] LIANG, T.P., "Modeling By Analogy: A Case-Based Approach to Automated Linear Program Formulation", 25th HICSS, pp. 276 - 283, 1991.

[4] H.B. ASHER, Causal Modeling, Beverly Hills, CA: Sage Publications Inc., 1983.

[3] ATA. N., COURTNEY, J.F., and D.B. PARADICE, "A Prototype DSS for Structuring and Diagnosing Managerial Problems", IEEE Transactions on Systems,Man, and Cybernetics, Vol. 18, No.6, Nov/Dec, pp. 899 - 907, 1988.

[5] VENKATRAMAN, N., The Concept of Fit in Strategy Research: Toward Verbal and Statistical Correspondence, Academy of Management Review, Vol. 14, No. 3, pp. 423 - 444, 1989.

[6] MILL, D., Toward a New Contingency Perspective: The Search for Organizational Gestalts, Journal of Managemnet Studies,Vol. 18, pp. 259 - 280, 1981.