Neural Network-based Decision Class Analysis with Incomplete Information

김재경* · 이재광** · 박경삼**

Jae Kyeong Kim*, Jae Kwang Lee**, Kyung Sam Park**

Abstract

Decision class analysis (DCA) is viewed as a classification problem where a set of input data (situation-specific knowledge) and output data (a topological leveled influence diagram (ID)) is given. Situation-specific knowledge is usually given from a decision maker (DM) with the help of domain expert(s). But it is not easy for the DM to know the situation-specific knowledge of decision problem exactly. This paper presents a methodology for sensitivity analysis of DCA under incomplete information. The purpose of sensitivity analysis in DCA is to identify the effects of incomplete situation-specific frames whose uncertainty affects the importance of each variable in the resulting model. For such a purpose, our suggested methodology consists of two procedures: generative procedure and adaptive procedure. An interactive procedure is also suggested based the sensitivity analysis to build a well-formed ID. These procedures are formally explained and illustrated with a raw material purchasing problem.

Key word : Neural Network, Influence Diagram, Sensitivity Analysis, Decision Analysis, Decision Class Analysis

1. Introduction

The decision analysts have observed that a constructed decision model such as an influence diagram (ID) is usually applicable to only one specific problem even if the formulation of a real decision problem needs much time, efforts, and cost. They often investigate that some prior knowledge from the experience of modeling IDs can be utilized to resolve other similar decision problems. From this investigation, we considered a decision analysis to combine the prior knowledge so that we handle a set of similar decision problems simultaneously (Kim and Park, 1997).

Decision class analysis (DCA) regards decision analysis as an integrator of decision knowledge and treats a set of decisions having some degree of similarity as a single unit (Holtzman, 1989). DCA helps decision analysts to inexpensively model a decision problem from a cumulative set of decisions. Our previous study used a neural network for analyzing a class of decisions which results in the generation of IDs in the topological level (Kim and Park, 1997; Kim et. al., 1999). We considered DCA as a classification problem where a set of input data (situation-specific knowledge) and output data (a topological leveled ID). So the quality of resulting ID depends on the quality of input data given from decision maker (DM) with the help of domain expert(s). But it is not easy for the DM to know the situation-specific knowledge of a decision problem exactly. The input data may be imprecise vague or incomplete.

In this paper, we suggest a procedure for the sensitivity analysis of DCA with incomplete information. Sensitivity analyses identify those input parameters to which perturbations of the base-case value causes the greatest impact on the output measure. The input parameters of DCA are *situation frames* (i.e., the DM's circumstances) of an individual specific decision problem, and output parameters are abstracted corresponding specific decision variables (i.e., topological leveled chance nodes and arcs in influence diagram) for solving that problem. In the previous studies, DM is compelled to input the situation frame value even if he/she does not know them exactly (Chung et. al., 1992; Kim, 1991; Kim and Park, 1997). Based on next, analyzing sensitivity in DCA perturbs incomplete situation frames and examining the effect on the abstracted corresponding specific decision variables for solving the individual decision problem. As a result, the relative importance or sensitiveness of each decision variables (chance nodes and arcs) can also be known, which could result in the ID. These two steps of our methodology are named as generative procedure and adaptive procedure in sensitive analysis. Based on the result of sensitivity analysis, DM can modify the resulting ID with the help of domain experts and his/her decision-specific knowledge. This interactive procedure is also suggested to aggregate the

* Department of Management Information Systems, Kyonggi University
** Graduate School of Management, KAIST

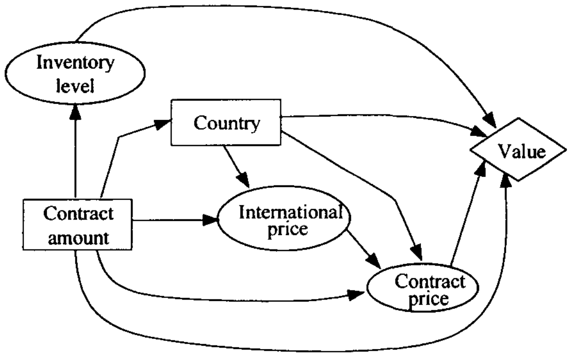
resulting IDs from the sensitivity analysis. The properties of our suggested procedure are summarized as follows: First, it is possible to model an ID for the specific decision problem with less time and costs. Second, domain experts easily append their additional information and knowledge in the adaptive process of resulting ID. Third, the sensitiveness or robustness of each variable of a resulting ID can be also known. So little resources will be focused on analyzing the sensitive variables, i.e., robust variables of an ID. We applied our procedure to a real world decision class problem, a raw material purchasing problem (Chung et. al., 1992).

This paper unfolds as follows: Section 2 contains the background, brief illustrations of an influence diagram, a sensitivity analysis in decision analysis, the concept of DCA, and neural net in implementing DCA. In Section 3, we describe a concept of sensitivity analysis in DCA, and present a procedure for the implementation of sensitivity analysis in DCA. An interactive procedure adapting IDs was also proposed in section 3. The procedure is illustrated with a raw material purchasing problem in section 4. Finally, the conclusions are given in section 5.

2. Background

2.1. Influence Diagram

Influence diagrams (IDs) are developed as a model for representing complex decision problems based on incomplete and uncertain information from a variety of sources (Howard & Matheson, 1984).



[Figure 1] An influence diagram of a raw material purchasing problem.

Figure 1 shows an ID of a raw material purchasing problem originally described by Kim (1991). The ID can be viewed from three levels: topological, functional, and the numerical level (Howard & Matheson, 1984; Kim & Park, 1997). At the topological or relational level the nodes in the diagram represent the key variables in the system being modeled and the arcs or arrows identify conditional influences among them. The nature of these influences is specified at the functional level and further quantified at the numerical level. Each level would provide a stage of decision making in a given domain.

At the topological level, the ID is defined as an acyclic digraph $G = (N, A)$, where N is a finite set of nodes and A is a set of arcs, $A \subseteq N\times N$. This visual level of the ID explicitly reveals the flow of information, influences, and overall structure of the decision problem. The nodes are partitioned into three sets C, D and V. The chance node $c\in C$, which is circular shape, represents uncertain or certain states and the rectangular-shaped decision node $d\in D$ reveals variables whose values are chosen by the DM. The diamond-shaped value node $v\in V$ represents the objective to be maximized in expectation by the DM.

The functional level is concerned with how nodes are related. At the final level, numerical level, probability distributions from prior information, decision values and costs, and the utilities of the DM are assessed numerically for each node. Well-formed influence diagram (WFID) is a syntactically correct, completely assessed ID whose nodes have fully consistent distributions and outcomes (Holtzman, 1989). But in this research, we use WFID to refer a well-constructed decision model from which a decision is made without further modification of the model.

The traditional interactive procedure to generate an ID consists of a sequence of *value-preserving transformation* between domain expert(s), decision analyst(s), and DM (Kim, 1991). The value preserving transformation is a transformation of the ID which maintains feasibility and do not modify the optimal policy or maximal expected value. The process to expand an ID is made through the repetitive operation of adding nodes, and splitting nodes. Once the structure is reasonable, the diagram is further refined in more detail through the operation of node removal, merging nodes, and reversing an arc as well as adding and splitting nodes. It was shown that these operations satisfy the value-preserving transformation (Kim, 1991; Kim et. al., 1997; Shacter, 1986).

Once a WFID is built, the diagram is manipulated and evaluated for determining the optimal decision strategy. A direct solution procedure to automate IDs has been proposed by Olmsted (1984) and formalized by Shachter (1986). The procedure consists of value-preserving transformations, node removal and arc reversals, which correspond to the rollback procedure in decision tree models.

2.2. Sensitivity Analysis in Decision Analysis

The decision analysis process, presented by Howard (1988), iterates through formulation, evaluation, and appraisal phases. The focus of the decision analysis process is the conversation about the decision situation that leads to clarity of action. Sensitivity analyses identify those input parameters to which perturbations of the base-case value causes the greatest impact on the output measure. In each phase of the decision analysis process, sensitivity analysis is used to arrive at contingent conclusions or to tell its own story. In the formulation phase, deterministic sensitivity analysis helps the analyst identify the uncertainties that have the largest effects on the value of each alternative and to observe whether any alternative is dominated by the others. Usually, five to seven critical uncertainties are modeled probabilistically; the remaining uncertainties are set at deterministic values. The analyst creates a probabilistic model of the decision and assesses probability distributions for the critical uncertainties. Recently a sensitivity analysis to relevance is developed by Lowell (1994). Before assessing probability distributions for the critical uncertainties, analyzing sensitivity to identify which relevance relationships among the critical uncertainties need to be assessed and modeled. In the evaluation phase, the analyst determines the alternative that is logically consistent with the decision basis constructed in the formulation phase. Within the appraisal phase, one could analyze sensitivity to conditional probabilities, to risk attitude, and to time preference. For a discussion of sensitivity analysis, see Howard (1988).

2.3. Neural Network in DCA

The formulation step in decision analysis has been accomplished manually, through lengthy interviews among decision analyst(s), DM, and domain experts who are intimately familiar with the problem domain. The model construction in practice is known to be a most complicated and burdensome process. Furthermore, the decision analyst may observe that a constructed decision model such as IDs is usually applicable to only one specific problem (Kim, 1991). Holtzman (1989) describes decision class analysis (DCA) which regards a decision analysis as an integrator of decision knowledge and treats a set of decisions having some degree of similarity as a single unit. In this paper, similarity among decision problems are interpreted in such a way that one or more key variables relevant to a given decision problem are admitted into the model of another decision problem. Specifically, an ID involves one or more nodes (i.e. key variables) being in another ID. We further denote a class of decision problems to be the collection of such IDs that their nodes are partially (or entirely) owned by each diagram within the class (Kim and park, 1997).

Whereas the end result of an individual decision analysis is a decision, the result of a DCA is an individual decision analysis. Thus, analyzing a class of decisions occurs at a higher level of abstraction than analyzing a single decision. In the construction of a decision model, typically, domain experts provide domain-specific knowledge, while a DM furnishes situation-specific knowledge, i.e., information about his or her current situation environment.

At the point of decision making with the ID, variables in the ID are changeable from the current specific situations. The specific situations may be decision nodes and situation frames. When given the situation-specific information of the DM, the DCA should abstract the corresponding specific decision variables for solving the individual problem. This interprets DCA as a classification problem. In contrast to rule-based systems, neural networks have a broad response capability because of their capability to provide the general classification of a set of inputs. They can capture a large number of cases quickly and provide reasonably accurate responses, even when incomplete or previously unseen inputs are given. More specific on the perspective of implementing a DCA using neural network appears in Kim & Park (1997).

3. Sensitivity Analysis in Decision Class Analysis

3.1. Sensitivity Analysis in DCA

Analyzing a class of decisions is composed of three steps: First, the DM decides decision node(s) to represent the decision-making purpose of a given problem. Second, the DM suggests knowledge of specific situations occurred at current circumstance and situation. Third, to obtain a single decision analysis, an ID is built based on the decision and the situation-specific knowledge. The third step is made of two phases: Phase I is to search for relevant chance and value nodes of the individual ID from the given decision nodes and specific situations (i.e., situation frames denoted in Section 2.3). Phase II elicits arcs among the nodes.

At second step, the well-represented situation-specific knowledge plays a major role to elicit a ID through the DCA. However, it is not easy for the DM to know the situation-specific knowledge exactly. For example, let s1 and s2 to be situation frames, where s1 is "variation of domestic economy", and s2 is "variation of foreign economy". It will be incorrect if we use an ID based on imprecise information like "s1 and s2 is important," when we don't have precise knowledge about the value of s1 and s2. Instead, it is more realistic to derive IDs from the combinations of each possible value of s1 and s2, then adapt the derived IDs considering domain specific knowledge of DM and domain experts. In this example, we can get four IDs from the four situations, like only s1 is important, only s2 is important, both s1 and s2 are important, and both s1 and s2 are unimportant. Our suggested methodology allows situation frames have incomplete values. When the DM does not have a complete information, it is another burden for him (her) to represent the situation-specific knowledge exactly. The basic idea of our methodology makes DCA to be useful and convenient for the DM.

Analyzing sensitivity in DCA starts from perturbing incomplete situation frames and examining the results of the abstracted corresponding specific decision variables (i.e., chance nodes and arcs in our problems) of the derived individual ID. It examines whether these perturbations cause variation in the *presence* of variables such as nodes or arcs in ID. So instead of focusing limited resources on modeling, quantifying and analyzing the uncertainty in these situation-specific sensitive variables, it makes DM focus on the modification or adaptation of resulting IDs. Furthermore, the result of sensitivity analysis in DCA can be used in sensitivity analysis in decision analysis, because the sensitivity analysis for DCA tells which nodes (or arcs) are important or not.

3.2. Generative Procedure in Sensitivity Analysis

Recall the notation of the ID: on the nodes, decision $d_k \in D$, chance $c_k \in C$ and value $v_k \in V$; and the arcs $a(i,j) \in A$ where i and j are the indices of the nodes. Denote a set S in which there are one or more situation frames $s_k \in S$. We further define the following sets:

$DS = \{ d_k \in D \mid k = 1, ..., p\} \cup \{ s_k \in S \mid k = 1, ..., n\}$,
$CV = \{ c_k \in C \mid k = 1, ..., q\} \cup \{ v_k \in V \mid k = 1, ..., r\}$,
$DCV = \{ d_k \in D \mid k = 1, ..., p\} \cup CV$,
$ARC = \{ a(i,j) \in A \mid i < j;\ i = 1, ..., p+q+r-1;\ j = 2, ..., p+q+r\}$,
where p, q, r, n are respectively the number of decision, chance, value nodes, and situation frames. The size of ARC is $|ARC| = {}_{p+q+r}C_2$ (where C is combination operator).

For each situation frame, $s_k \in S$, three values are possible which are 1, 0, and 'u'. It is 1 if it is present, it is 0 if it is not present, and it is 'u' if it is unknown whether it is present or not. If the value of s_k is 'u', then it can have both 1 and 0. Therefor, if t unknown situation frames exist among n situation frames, 2^t situation sets are possible, which is a combination of their possible values. We call the situation frame of which the value is 'u' as $s_{u(i)}$, where $i = 1, ..., t$. The meaning of the value of s_k and that of chance node should be distinguished. If the value of s_k is close to 1, the certainty of its relation to a current problem is to be increased. But as the value of chance nodes is close to 1, the certainty of its presence in the resulting ID is increased. The value of arc $a(i,j)$ ranges from –1 to 1. If its direction is (i,j) then $a(i,j)$ has positive value. if its direction is (j,i) then $a(i,j)$ has negative value. And if *ith* node and *jth* node is independent, i.e., if the arc between i and j does not exist then $a(i,j)$ becomes 0.

Based on the above notations, we suggest the following procedure to generate possible IDs.

Step(0) Set $s_{u(i)} = 0$ for each $s_{u(i)} \in S$, $i = 1, ..., t$.
  Set k = 0.
  Set L = 1.
Step(1) Generate L^{th} situation set S^L.
  1.0 Set i = 1.
  1.1 Let $s_{u(i)}$ = (k mod 2).
  1.2 Let k = (k – (k mod 2)) / 2.
  1.3 If k = 0 then goto Step (2).
  1.4 Otherwise i = i + 1. Goto 1.1.
Step(2) Perform Phase I.
Step(3) Perform Phase II to build an extreme ID, ID^L.
Step(4) If L= 2^t then Stop.
  Otherwise k = k +1.
  Goto Step(1).

Step(0) and Step(1) generate 2^t sets of situation frames for t unknown situation frames. Step(0) and Step(1) are carried out iteratively. Based on each situation set, an ID is built from Step(2) and Step(3). Both steps are done by two trained feed-forward neural nets (shortly termed NNs): NN I for Phase I and NN II for Phase II. The two neural nets are trained by the training set obtained from the knowledge of decision participants, using the back-propagation algorithm. We call this ID as an extreme ID. Step(4) is about stopping condition of the procedure, which stops until 2^t extreme IDs are built. An initial ID built on the imprecise information of 'u' situation frames is just one of extreme IDs. As DM is not certain of the value of 'u', it is not suggested to use an initial ID. Instead it is preferable to do a sensitivity analysis, i.e., generate all the extreme IDs based on all the possible sets of t imprecise situation frames, and then aggregate them into an ID considering DM's domain specific knowledge. Next section is about adaptation procedure. Before that, we explained phase I and phase II in brief.

3.2.1. Phase I

In order to search for relevant chance and value nodes from the given decision nodes and situation frames, NN I is used to represent the relation between chance and value nodes, and decision nodes and situation frames. The training set of input and desired output pairs to learn NN I is represented as $\{(DS_m, CV_m)| \ m = 1, ..., M_1\}$, where M_1 is the training set size, and the $DS_m \in (x\text{: } 0 \le x \le 1)^{p+n}$ and $CV_m \in (x\text{: } 0 \le x \le 1)^{q+r}$ respectively are the valued vector of *DS* and *CV*. These training pairs can be generated from the case studies of the similar problems within the class, with the help of the DMs and domain experts.

3.2.2 Phase II

In this phase, a trained NN II has to be prepared to elicit the influences among output nodes of Phase I. The training set of input and desired output to learn NN II is given by $\{(DCV_m, ARC_m)| \ m = 1, ..., M_2\}$, with the $DCV_m \in (x\text{: } 0 \le x \le 1)^{p+q+r}$ and $ARC_m \in (x\text{: } -1 \le x \le 1)^{|ARC|}$. Namely, the desired output of Phase I becomes the input of Phase II, and the desired output of Phase II is the strength of arcs of an ID. Hence the training pairs can be generated consecutively based on those of Phase I. A more detail of Phase I and Phase II including training process with a case example is expressed in our previous paper (Kim & Park, 1997).

3.3. Adaptive Procedure in Sensitivity Analysis

We call the initial ID generated from imprecise value of 'u' an *initial ID*, ID_{INI}. Three definitions are given before explaining the aggregation process for reconciling different extreme IDs.

Definition: A *basic ID*, ID_{BAS} is the minimal ID which is the intersection of all the *extreme* IDs.
$ID_{BAS} = DCV_{BAS} \cup ARC_{BAS}$.
$DCV_{BAS} = \bigcap_1^L DCV_k$, where L is the number of situation sets and DCV_k is the DCV of *k*th situation set.

$ARC_{BAS} = \bigcap_1^L ARC_k$, where L is the number of situation sets and ARC_k is the ARC of *k*th situation set.

Definition: A *super ID*, ID_{SUP} is the maximal ID which is the union of all the *extreme* IDs.
$ID_{SUP} = DCV_{SUP} \cup ARC_{SUP}$.
$DCV_{SUP} = \bigcup_1^L DCV_k$.
$ARC_{SUP} = \bigcup_1^L ARC_k$.

Definition: A *supplementary ID*, ID_{SPL} is the difference of ID_{SUP} and ID_{BAS}.
$ID_{SPL} = ID_{SUP} - ID_{BAS}$.
$DCV_{SPL} = DCV_{SUP} - DCV_{BAS}$.
$ARC_{SPL} = ARC_{SUP} - ARC_{BAS}$.

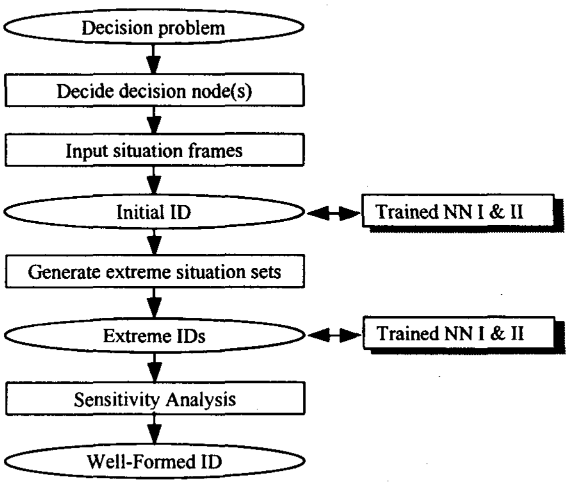
A basic ID is built by finding the intersection of all extreme IDs. It means the overlapped or intersected part of extreme IDs. The super ID, the union of all extreme IDs, represents a possible extent of situation information or knowledge of the given problem. The difference between super ID and basic ID, ID_{SPL} implies the degree of incompleteness or impreciseness of DM's knowledge about situation frames of a given decision problem. If DM know situation frames exactly for a given decision problem, then ID_{SPL} does not exist because only one extreme ID is generated. As DM has the knowledge of domain specific knowledge, he/she prefers to modify the ID_{BAS} or ID_{SUP} instead of using ID_{INI}. The procedure suggests for the DM to modify the ID_{INI} for the given decision problem. The basic idea of this procedure is that it starts from ID_{BAS}, which is stable or robust not with standing 'u' value. ID_{BAS} corresponds to sensitive variables in the sensitivity analysis of decision analysis. Next the procedure adds up the accepted part of ID_{SPL} to make the ID_{BAS} to be a WFID.

The procedure is as follows. It iterates until DM is satisfied. We let iteration number h.

Step(0) Set h=1.
Obtain $ID_{BAS(h)}$ and $ID_{SUP(h)}$ from L extreme IDs.
Step(1) Obtain $ID_{SPL(h)}$ from $ID_{BAS(h)}$ and $ID_{SUP(h)}$.
Step(2) Select a node or an arc arbitrary from $DCV_{SPL(h)}$ and $ARC_{SPL(h)}$ respectively.
Step(3) DM decides whether the node and the arc is accepted or not.
Step(4) Modify $ID_{BAS\ (h)}$ by applying model expansion procedure considering result of Step(3).
Step(5) If $ID_{SPL(h)}$ is a null set then stop. $ID_{BAS\ (h)}$ is a WFID.
Otherwise set h=h+1.
Go to Step(2).

Step(2) selects a node or an arc from $ID_{SPL(h)}$. In Step(2), a selection is made arbitrary. Another heuristic selection method is considering the priority of each node and each arc. The priority is determined based on "*weight*" of each element. For example, if node A is present in more extreme IDs than B then, the weight of node A is higher than node B. The weight of each element in $DCV_{SPL(h)}$ and $ARC_{SPL(h)}$ provides a helpful information to DM, when he/she adds up the $ID_{BAS(h)}$. In Step(5), the fact that $ID_{SPL(h)}$ is a null set means that the procedure is terminated because there is no remaining nodes or arcs in the sets of $DCV_{SPL(h)}$ and $ARC_{SPL(h)}$.

Figure 2. shows the overall procedure of the sensitivity analysis to build a WFID.



[Figure 2] Overall procedure for sensitivity analysis in DCA

4. An Illustrative Example

4.1 A Raw-Material Purchasing Problem

A raw-material purchasing problem in a textile cooperation: The company makes some types of synthetic products. The main raw materials of these products are TPA (terephthalic acid), DMT (dimethy terephthalate) and EG (ethylene glycol). Presently, the raw materials are imported from the foreign market.

[Table 1] Summary of the raw material purchasing problem.

Node name	Symbol	Content
Decision	d1	Contract amount
	d2	Country
Chance	c3	Product demand
	c4	Inventory level
	c5	International price
	c6	Spot price
	c7	Contract price
	c8	Reliability
	c9	Transportation time
	c10	Quality
Value	v11	Value
Situation	s1	Purchasing TPA raw-material
	s2	Purchasing DMT raw-material
	s3	Purchasing EG raw-material
	s4	Variation of OPEC policy
	s5	Variation of domestic economy
	s6	Variation of foreign economy
	s7	Variation of opposite company policy

The state related with the decisions is varied with the types of goods and the affecting specific situations.

According to a given raw material and the specific situation, variables affecting the decision made a difference. In this problem, decisions of a similar type are to be made frequently and these decisions strongly depend on the specific situations (situation frames).

Once the DCA is implemented using neural networks, we can greatly reduce the time and save large amounts of duplicate efforts for modeling an ID for an individual decision problem. Using the notations described in the previous section, we restrict the boundary of the raw-material purchasing problem in Table 1.

4.2 Generative Procedure for the Example

In Training NN I, the number of input processing elements is $|DS|$ = 9 (the number of decision nodes and situation frames) and the number of output processing elements is $|CV|$ =9 (the number of chance and value nodes) (see Table 1). In training NN II, the number of input processing elements is $|DCV|$ = 11.

Table 2 shows a current situation frames of a raw material purchasing decision problem. In this problem, the textile cooperation has to decide country and amount.

[Table 2] A situation frames of the example.

Symbol	Content	
d1	Contract amount	Certainly yes
d2	Country	Certainly yes
s1	Purchasing TPA raw-material	Certainly no
s2	Purchasing DMT raw-material	Certainly no
s3	Purchasing EG raw-material	Certainly yes
s4	Variation of OPEC policy	Certainly yes
s5	Variation of domestic economy	Probably yes
s6	Variation of foreign economy	Probably no
s7	Variation of opposite company policy	Certainly yes

Given the situation frame with incomplete information, s5 and s6, we obtained the output of Phase I using NN I. The output became the input of Phase II, and NN II elicited arcs among the nodes. Figure 1 shows the resulting ID, an initial ID. The initial ID is generated under incomplete information of situation frame, "variation of domestic economy" and "variation of foreign economy". As s5 and s6 are incomplete situation frames, we performed the generative procedure in sensitivity analysis. As a result, four extreme situation sets are generated. The extreme situation sets and the DCV of the extreme IDs are presented in Table 3.

[Table 3] Extreme situations and extreme IDs.

s1	s2	s3	s4	s5	s6	s7	DCV of the extreme ID	
0	0	1	1	0	0	1	{d1, d2, c5, c7, v11}	ID_1
0	0	1	1	1	0	1	{d1, d2, c4, c5, c7, v11}	ID_2
0	0	1	1	0	1	1	{d1, d2, c3, c5, c6, c7, v11}	ID_3
0	0	1	1	1	1	1	{d1, d2, c3, c4, c5, c6, c7, c8, v11}	ID_4

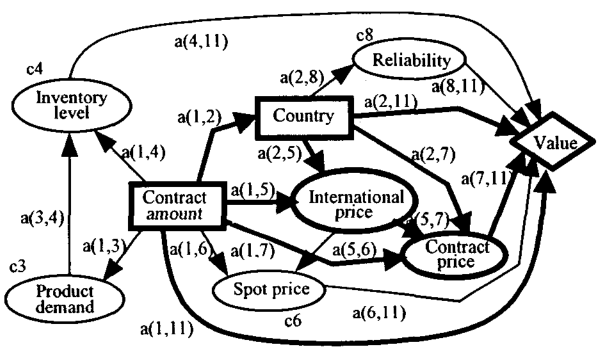
4.3 Adaptive Procedure for the Example

In aggregation process, four extreme IDs are generated using trained NN I and NN II based on extreme situation sets. All the extreme IDs reflect the possible variation of the incomplete information, s5 and s6. To analyze the sensitivity, an adaptive procedure is executed. First, $ID_{SUP(0)}$, $ID_{BAS(0)}$, and $ID_{SPL(0)}$ are obtained from the four extreme IDs. Figure 3 graphically represents the $ID_{SUP(0)}$, $ID_{BAS(0)}$, and $ID_{SPL(0)}$. The entire diagram represents $ID_{SUP(0)}$. The $ID_{BAS(0)}$ consists of the following $DCV_{BAS(0)}$ and $ARC_{BAS(0)}$, which is presented as bold line in Figure 3.

$DCV_{BAS(0)} = \{ d_1, d_1, c_5, c_7, v_{11} \}$
$ARC_{BAS(0)} = \{$ a(1,2), a(1,5), a(1,7), a(1,11), a(2,5), a(2,7), a(2,11), a(5,7), a(5,11), a(7,11) $\}$.

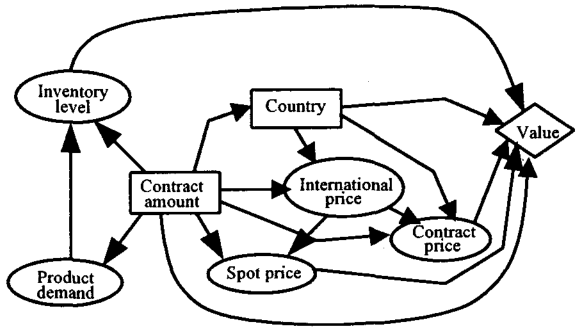
Therefore the $ID_{SPL(0)}$ is obtained from the difference between the $ID_{SUP(0)}$ and the $ID_{BAS(0)}$ as follows;

$DCV_{SPL(0)} = \{ c_3, c_4, c_6, c_8 \}$
$ARC_{SPL(0)} = \{$ a(1,3), a(1,4), a(1,6), a(2,8), a(3,4), a(4,11), a(5,6), a(6,11), a(8,11) $\}$.



[Figure 3.] Initial basic ID and super ID

Among the ID_{SUP}, four chance nodes, $\{ c_3, c_4, c_6, c_8 \}$ and nine arcs, { a(1,3), a(1,4), a(1,6), a(2,8), a(3,4), a(4,11), a(5,6), a(6,11), a(8,11) } are included in ID_{SPL}. Adaptive procedure is carried out to add up the ID_{BAS} until the ID_{SPL} becomes a null set. In the adaptive procedure, the nodes c_3, c_4 and c_6 are accepted by DM. Also the arcs, a(1,3), a(1,4), a(1,6), a(3,4), a(4,11), a(5,6), a(6,11), are accepted also. However, DM regards that node c_8 and arcs, a(2,8), and a(8,11) are not important. Therefore, nodes c_3, c_4, c_6 and c_8 are removed from the DCV_{SPL}, so that the DCV_{SPL} and ARC_{SPL} became a null set. In summary, the nodes c_3 and c_6 are added to the ID_{INI}. Figure 4 represents the final WFID for the raw material purchasing problem.



[Figure 4] Final ID of the raw material purchasing problem

5. Conclusions

Analyzing a class of decisions has been applied to model a decision problem efficiently. To make a resulting ID to represent a real problem well, it is essential to suggest an exact situation-specific knowledge. This paper proposed a procedure for the sensitivity analysis of DCA with incomplete information. The use of neural networks to generate IDs in the topological level results in a good performance. But the generated ID from imprecise situation-specific knowledge does not usually represent well a real problem. So it needs a sensitivity analysis by perturbing the value of imprecise situation-specific variables. Our suggested procedure is helpful when DM wants to reflect their domain-specific knowledge or information in the modeling process. With the suggested procedure, the DM can easily perform a sensitivity analysis of DCA and obtain a well represented ID of a specific decision problem. Our procedure is applied to a real world decision class problem. The interactive procedure combined with neural networks is expected to be a basic methodology of intelligent decision support system to build a decision model. Developing an intelligent decision support system based on this research is a subsequent research area. Besides on this, other promising research area are to adapt a case based reasoning approach to define a class of decisions.

References

Chung, T.Y., J.K. Kim & S.H. Kim, "Building an influence diagram in a knowledge-based decision system. Expert Systems With Applications, Vol.4, (1992), 33-44.

Holtzman, S. Intelligent Decision Systems. Reading, MA: Addison-Wesley, 1989.

Howard, R.A. Decision analysis: practice and promise. Management Science, Vol.34, (1994), 679-695.

Howard, R.A. & J. Matheson, Influence diagrams. in Howard, R.A. & Matheson, J.E. (eds). Readings on the Principles and Applications of Decision Analysis. Vol. II, Menlo Park, CA, Strategic Decision Group, 1984.

Kim, J.K. A Knowledge-Based System for Decision Analysis. Ph.D. thesis, Department of Industrial Engineering, KAIST, 1991.

Kim, J.K., J.K. Lee & S.H. Kim. "An Interactive Approach to Build an Influence Diagram Based on Neural Networks", Journal of Decision Systems, Vol.8, no2, (1999).

Kim, J.K., & K.S. Park, (1997). "Neural network-based decision class analysis for building topological-level influence diagram", Int. J. Human-Computer Studies, Vol.46, (1997).

Kim, J.K. & S.C. Chu, "Sensitivity analysis in the decision class analysis using neural networks", Proceedings of the Third World Congress on Expert Systems, 1988.

Kim, J.K. & S.C Chu, "Analyzing a Class of Decision Problems: Neural Network Based Approach", Proceedings of the 31st Annual Hawaii International Conference on Systems Sciences, 1988.

Lowell, D.G. "Examining Sensitivity to Relevance in the Decision Analysis Process", Working Paper, Strategic Decision Group, 2440 Sand Hill Road, Menlo Park, California, 94025, 1994.

Lee, J.K., J.K. Kim, & S.H. Kim. "A Case-Based Reasoning Approach to Build an Influence Diagram", Proceedings of the Fifth Pacific Asia Conference on Expert Systems (PACES), Los Angeles, 1999.

Olmsted, S.M., On Representing and Solving Decision Problems. Ph.D. thesis, Department of Engineering-Economic Systems, Stanford University, 1984.

Rumelhart, D. & J. McClelland, Parallel Distributed Processing. Vol. 1, Cambridge: MIT Press, 1986.

Shachter, R.D. "Evaluating influence diagrams", Operations Research, Vol. 34, (1986), 871-882.