Universidad Rey Juan Carlos

Máster Universitario en Ingeniería de Sistemas de Decisión

Curso Académico 2010/ 2011

Proyecto Final: Tesis de Máster

A Geometric Algorithm For Solving Multiobjective Influence Diagrams

Autor: Aitor Arronte Álvarez

Tutor: Román Efremov

Contents

1. Introduction

In this work we will present a novel method for solving influence diagrams with multiple objectives (MOID), using a geometric optimization algorithm. First of all we will present several approaches for representing and solving influence diagrams with a single objective; later we will present a method for solving influence diagrams with multiple objectives, and describe our method for solving multi objective problems using influence diagrams.

We will analyze the different approaches and further discuss the advantages and limitations of the method here presented. Our goals are to identify existing methods for solving MOID, discuss their limitations and propose alternative ways for solving decision problems with multiple objectives, using influence diagrams.

2. Influence Diagrams

Influence diagrams (IDs) were proposed by Howard and Matheson (1) as a tool to simplify modeling and analysis of decision trees (DT). DTs represent each decision or chance variable as a new level in a tree. The leaves of the tree are values that express ending configurations. Solving a decision problem requires finding the optimal path through this tree that maximizes the expected value.

With IDs instead of representing each level of that tree, we represent each variable as a single node in a graph connected to other nodes by arcs. There is a complete correspondence between between ID and DT in their functional characteristics.

2.1 Formal Definition and Semantics

An influence diagram is a directed acyclic graph G, with nodes corresponding to the variables (2). Variables are denoted by upper-case letters (X) and their states by lower-case letters (x). If there is an arc directed from node X to node Y, X is known as a parent of Y, and Y is a child of X. A node along with its parents forms the family for that node. The parents for node X are denoted $\mathrm{Pa}(X)$, and the family is denoted by $X\mathrm{Pa}(X)$. If there is a directed path from node X to node Y, we say that X is an ancestor of Y, and that Y is a descendent of X.

The variables in the diagram can be classified into three groups, $V = X \cup D \cup U$, where X are chance variables specifying the uncertain decision environment, D are the decision variables specifying the possible decisions to be made in the domain, and U are the utility variables representing a decision maker's preferences. Similar to Bayesian Networks, each variable $Xi \in X$ is associated with a conditional probability distribution $P(Xi|\mathrm{Pa}(Xi))$, where $\mathrm{Pa}(Xi)$ is the set of parents of Xi in G. Each decision variable $Dj \in D$ has multiple states, each state corresponding to a possible action. Incoming arcs into a decision variable are called information arcs, which require the chance variables originating these arcs to be observed before the decision is made. These variables are called the information variables of the decision. No-forgetting is typically assumed for an influence diagram, i.e., the information variables of earlier decisions are also information variables of later decisions, even though no explicit information arcs exist between them.

The criterion of decision making is represented by the value nodes. Solving an ID involves finding an optimal decision policy that maximizes the expected utility.

2.2 Solving Influence Diagrams

There are multiple methods and algorithms for solving influence diagrams. We are going to focus on the algorithm presented by Shachter (3) based on solving directly the ID by a series of transformations. The algorithm iteratively removes nodes from the diagram until all that remains is a value node and evidence nodes, representing the optimal expected value of the ID and the observations already made. The optimal strategy is determined in terms of an optimal policy for each decision before that decision is eliminated.

Transformations of the diagram. Solving influence diagrams implies a series of transformations to the original regular oriented diagram that maintain the optimal policy and therefore the maximum expected utility. These transformations are called value-preserving reductions. Thus a node is removed from the diagram if it is eliminated through a value-preserving operation. These transformations can be summarized as follows:

1. Barren node removal: A barren node is a node that has no successors; because of this, its value does not affect any other node regardless of its type. If a decision node is a barren node, any of its states would be considered optimal.

2. Chance node removal: when a chance node has only one child and that child is a value node, chance node removal eliminates the node from the diagram by taking expectation and the value node inherits its parents.

Example: Given the decision nodes *A*, *B*, *C*, chance node *X*, and value node *V* as in figure 1a ; the expectation operation by removing node *X* would be (transformation in figure 1b):

$$E[V|A,B,C] = \sum_{x} P\,\{X = x\,|A,B\}\,E\,[V|X = x, B, C]$$

Eq. 1

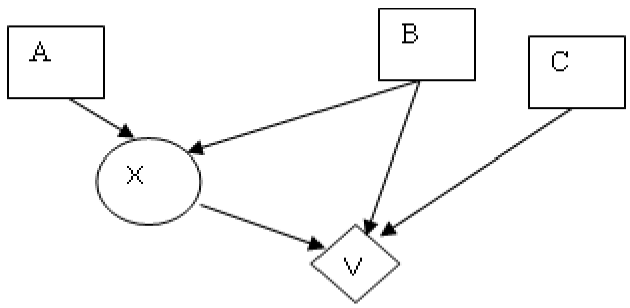

Figure 1a

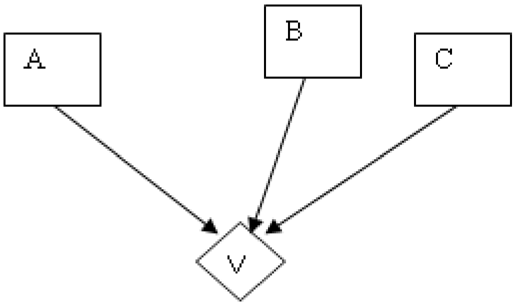

Figure 1b

1. Decision node removal: when a decision node has only one child and that child is a value node, and all parent nodes of the value node are also parents of the decision node, this node may be removed by optimal policy determination for each possible combination of the parents of the decision node. The optimal policy must be saved.

   Example: Given decision nodes *B* and *D* and value node *V*, we eliminate *D* by optimal policy determination (see figure 2a and 2b):

$$E[V|\mathrm{B}=\mathrm{b}] = max_d\ E[V|\mathrm{D}=\mathrm{d},\mathrm{B}=\mathrm{b}], for\ alll\ possible\ b$$

Eq. 2

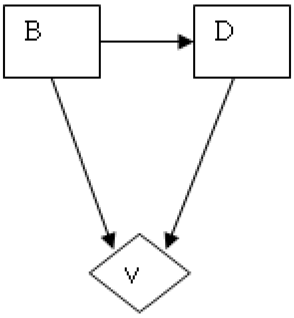

Fig. 2a

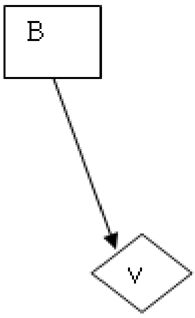

Fig 2b

2. If there are multiple value nodes, value node removal replaces two value nodes by one value node equal to their sum with the union of their parents weighted by the values of α and β; this additive value node will be called ALU, since the operation performed on the value nodes is a linear additive function, as shown in Eq. 3 and in Figure 3a and 3b.

$$E[V_1 + V_2|\, A, B, C] = E[V_1|\, A, B] + E[V_2|\, B, C]$$

Eq. 3

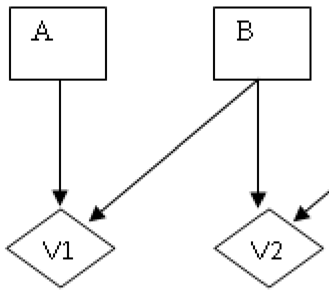

Figure 3a

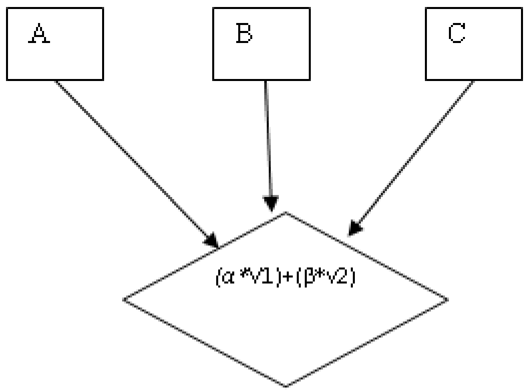

Figure 3b

3. Arc reversal applies Bayes' rule to reverse the arc between the uncertain nodes.
   Example: An arc from uncertain node *X* to uncertain node *Y* with parent sets *A*; *B* and *C* can be reversed, unless there is another directed path from *X* to *Y* ; if there were, then reversing the arc would create a directed cycle.

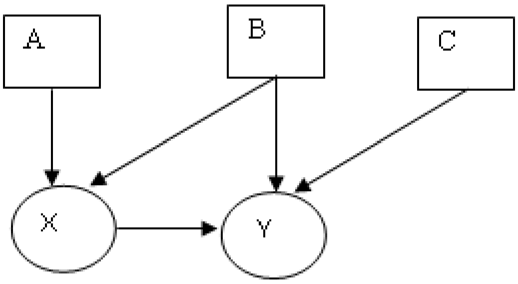

Figure 3a

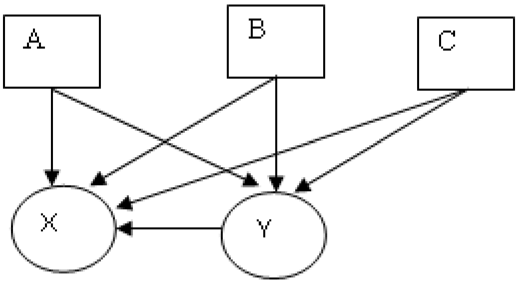

Figure 3b

Algorithm. Given a directed, acyclic influence diagram with ordered decision with no forgetting, can be solved by the following algorithm:

1. If there is a barren node, remove it.
2. If a chance node exists with the value node as a child, remove it by expectation. If any node remains in the diagram repeat step 2, otherwise terminate.
3. If a decision node exists with a value node as a child, and all parent nodes of the value node are also parents of the decision node, remove it by optimal policy determination. If any nodes remain in the diagram return to step 2, if there are more than one value node go to step 4, otherwise terminate.
4. If there is more than one value node, we will replace all the value nodes by one, as a result of the weighted sum of all, creating an ALU node. Terminate.

2.3 Example

In the following example we will demonstrate the algorithm shown in the previous section.

Economic Problem (first version). The Government decides to submit to the Congress a new policy for investing in technology, which involves training citizens to learn new technologies. The main goal of this proposal is to maximize public opinion regarding government in a difficult time of crisis. The decision variables of the problem are, the amount of the technological investment, and the type of training plan that the government will implement. The training plan decision is affected by a chance variable, which is the economic situation in the upcoming months, for which experts have given their opinions. The structure of this decision problem is shown in the following ID.

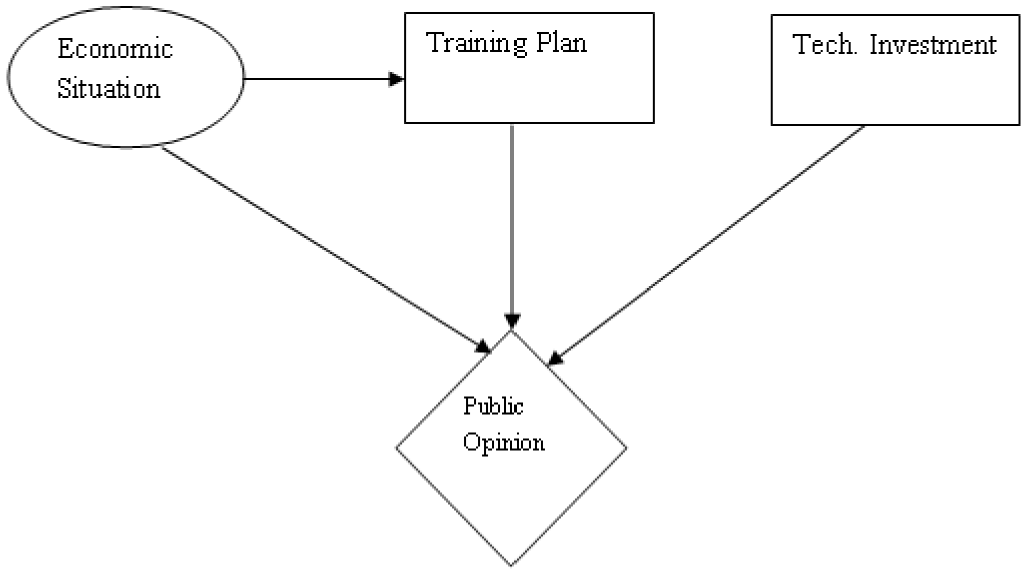

Figure 4

The following tables show the values of each node.

DECISION NODE	ALTERNATIVES
Tech. Investment	
	A
	B

For A =7 millions; B=5 millions

DECISION NODE	ALTERNATIVES
Training Plan	
	YES
	NO

CHANCE NODE		VALUES
Economic situation		
	IMPROVES	0,2
	REMAINS	0,6
	WORSENS	0,2

Table 1

VALUE NODE	PARENT NODES		
PUBLIC OPINION	TECH. INVESTMENT	ECONOMIC SITUATION	TRAINING PLAN
7	A	IMPROVES	YES
6	A	IMPROVES	NO
2	A	REMAINS	YES
3	A	REMAINS	NO
2	A	WORSENS	YES
1	A	WORSENS	NO
5	B	IMPROVES	YES
3	B	IMPROVES	NO
4	B	REMAINS	YES
4	B	REMAINS	NO
1	B	WORSENS	YES
2	B	WORSENS	NO

Table 2

We start solving the ID by eliminating the decision node "Training Plan" as it is a direct predecessor of the value node and has a chance node as a Parent. We apply optimal policy determination (as shown in *Eq. 2*). The results are shown in the following table and ID.

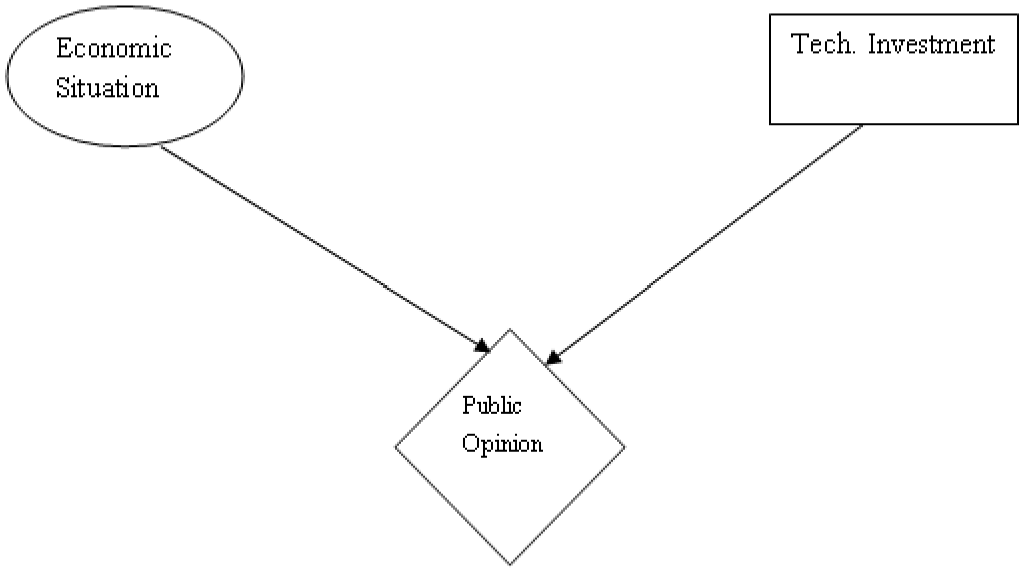

Figure 5

VALUE NODE	PARENT NODES		
PUBLIC OPINION	TECH. INVESTMENT	ECONOMIC SITUATION	TRAINING PLAN
7	A	IMPROVES	YES
3	A	REMAINS	NO
2	A	WORSENS	YES
5	B	IMPROVES	YES
4	B	REMAINS	YES
2	B	WORSENS	NO

The following step involves eliminating the chance node "Economic Situation", and applying the operation shown in *Eq. 1.* Afterwards we perform policy determination as a result of eliminating the decision node "Tech Investment". The results are shown in the following table.

VALUE NODE	PARENT NODES		
PUBLIC OPINION	TECH. INVESTMENT	ECONOMIC SITUATION	TRAINING PLAN
1,4	A	IMPROVES	YES
1,8	A	REMAINS	NO
0,4	A	WORSENS	YES
3,6			
1	B	IMPROVES	YES
2,4	B	REMAINS	YES
0,4	B	WORSENS	NO
3,8			

The result shows that the optimal policy for the current problem would be:

$$\begin{bmatrix} B;\ IMPROVES;\ \ YES \\ REMAINS; YES \\ WORST; NO \end{bmatrix}$$

3. Multi Objective Influence Diagrams

Since their inception IDs have gone through different transformations and changes in the structure and representation, as well as in the solution procedures.

Originally, influence diagrams were solved by first converting them into decision trees, and then deploying the fold-back and average-out algorithm (1), later new algorithmic approaches for solving influence diagrams directly were developed (3). Tatman and Shachter (4) introduced the concept of a super value node, allowing for decomposition of the value function and the possible use of dynamic programming as part of the solution procedure.

Influence diagrams are usually defined predominantly as having either a single value node or a single terminal value node that represents the product or sum of other value nodes To represent a decision-maker as having only a single objective is both unrealistic and inadequate for modeling most decision problems (5). Incorporating multiple objectives within influence diagrams has always been accomplished through a multiattribute utility function as part of *multiattribute utility theory*. The main problems in this approach derive from the difficulty of formulating an accurate and reliable utility function, and from the need to specify this information prior to solving the decision problem (6).

In this section we are going to present a method for solving IDs with multiple objectives developed by Diehl and Haimes (7), which is the only method we have found so far in the literature.

The method presented in this section (5), allows and facilitates the decision maker for the *posterior* specification of his preferences after the multiobjective optimization problem has been solved. The main idea behind this method is to separate solutions or alternatives into two sets at each optimization step in the solution procedure, those that are inferior and those that are non-inferior. A solution is only considered inferior if another solution has values that are equal or better for all objectives being considered. Only the set of non-inferior solutions (also termed as efficient, non-dominated, or Pareto optimum) is kept for further analysis after each stage of the problem. Once the set of non-inferior solutions is determined, the chosen alternative(s) is identified by evaluating the decision-maker's preferred tradeoff values with a suitable method.

3.1 Structure of Influence Diagrams with Multiple Objectives

Incorporating multiple objectives into an ID is accomplished through minor transformations to the basic structure of the single objective ID. On the relational (graphical) level, the standard representation of the various elements of an influence diagram remains the same. A new type of value node, a *multi objective value node*, is defined as a value node that contains a vector of objectives. Each element of the vector is a different, measurable objective whose value depends on one or more of the direct predecessors to the value node. Let *r* be defined as a *k*-dimensional vector of objective functions as $r = [\mathrm{r}_1, \mathrm{r}_2, \ldots, \mathrm{r}_\mathrm{k}]$. The multi objective value node is represented by a diamond and inherits all of the standard properties of a regular value node, including the restriction that the value node can have no successors. There can be only one multi objective value node in a multi objective influence diagram.

On the numerical level the transformations are more acute. From the representation stand point, extra columns are necessary to store each value for each objective. When a chance node is removed after the removal of a decision node, additional columns will be needed to represent the various non-inferior combinations of decision rules for each outcome of a chance node. This is a result of having to compute the expectation of all possible combinations of non-inferior solutions when removing a chance node. When a decision node is removed, the maximized expected utility operation is replaced by a vector optimization operation. The maximization operation when a value node is removed, is understood as multi objective optimization.

In the following section we will detail the transformations needed in order to solve multi objective influence diagrams (MOID).

3.2 Solving MOID

Taking as a point of departure the transformations to the diagram outlined in sub-section 2.2, we will point now to the changes required in order to evaluate a MOID.

1. Barren node removal: the use of a multi objective value node does not affect the node removal transformation. Therefore a barren node can be removed from a MOID without affecting the solution.
2. Chance node removal: this transformation is greatly affected by the addition of a multi objective value node. The condition that the chance node has solely the value node as a direct successor stays the same, as well as the requirement that the value node inherits all of the direct predecessors of the chance node after the removal stays the same. Although the application of conditional expectation must be modified. There are two different cases for a chance node removal, prior to a decision node removal and after the removal of one or more decision nodes. We explain them separately:

2.1 First case: when there has not been a removal of a decision node, the procedure for eliminating a chance node is very similar to the one applied by Shachter (3); the main difference is that the expectation operation is performed on each objective of the value node.

2.2 Second case: when performing the expectation after the removal of any decision node, we have to average out each possible combination of decision rules on the diagram. If we have a chance node *m* with *J* possible outcomes $[\theta_1, \theta_2, \dots, \theta_J]$ that influence the value node *v*, each outcome θ_J has d_j non-inferior solutions associated with it. Therefore there are $\prod_j \{d_j\}$ possible combinations of solutions when expectation can be performed (8). Once the expectation operation is performed every non-inferior solution is kept for further analysis, the dominated solutions are discarded.

3. Decision node removal: eliminating a decision node reqVires transformations to the method explain in sub-section 2.2; all conditions for decision node removal remain the same as in 2.2, although the operation of maximizing the expected

utility is replaced by a vector optimization operation (5) (8). The first step in this vector optimization methodology is to combine the sets of non-inferior solutions of each alternative through a union operation, on the set resulting from this operation a new set of non-inferior solutions is determined through the vector optimization process. The dominated solutions are discarded. Formally we can define it as follows: given a decision node n with I alternatives, $[a_1, a_2, \dots, a_i]$ that is removed from an MOID, each alternative a_i of the decision node n has a set $S(a_i)$ of non-inferior decision rules associated with it; in the following equation we show it for the minimization case understood in the Pareto sense :

$$\boldsymbol{S}^* = \mathbf{min} \bigcup_{\boldsymbol{i}=\mathbf{1}}^{\boldsymbol{I}} \boldsymbol{S}(\boldsymbol{a_i})$$

Eq. 4

4. Arc Reversal: this transformation remains the same as in 2.2, since by definition the multi-objective value node cannot be a direct predecessor to any node in the diagram, it cannot be affected by the reversal transformation.

3.3 MOID Example

We now present an example of solving an MOID using the methodology shown in the previous section. The current example shows every step of the algorithm introduced in 3.2.

Economic MO problem (second version). The Government decides to submit to congress a new policy for investing in technology which involves training citizens to learn new technologies. The goals of this new policy are to revive the economy through empowering citizens with the learning of new technical skills and increase employment in a difficult time of financial crisis. The main objective of this policy is to maximize the creation of jobs through an investment and training plan, while at the same time minimize raising taxes. These two objectives will depend on the following chance variable, economic stability in the coming months, which experts have given their judgments. Congress must decide the feasibility of this proposal, which entails a decision to raise taxes or not, create a training plan or not, and determining the amount of the investment.

The structure of the decision problem is shown in the following MOID:

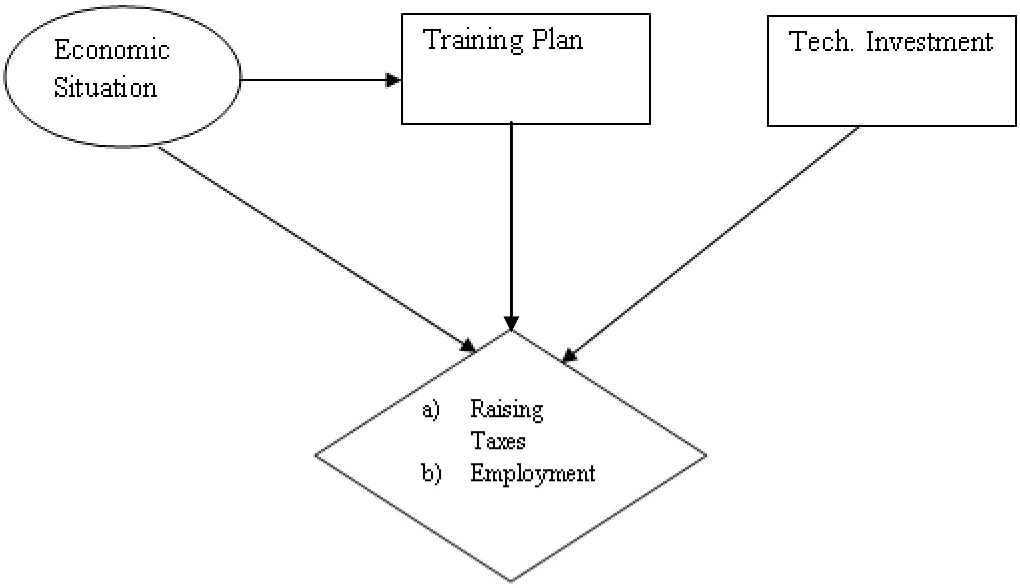

Figure 6

In *table 1* we show the different alternatives for each decision node as well as for the chance node, with its values. In *table 2,* we show the values for the multi objective value node and its parents.

DECISION NODE	ALTERNATIVES
Tech. Investment	
	A
	B

For A =7 millions; B=5 millions

DECISION NODE	ALTERNATIVES
Training Plan	
	YES
	NO

CHANCE NODE		VALUES
Economic situation		
	IMPROVES	0,2
	REMAINS	0,6
	WORSENS	0,2

Table 3

VALUE NODE		PARENT NODES		
RAISING TAXES	EMPLOYMENT	TECH. INVESTMENT	ECONOMIC SITUATION	TRAINING PLAN
3	60	A	IMPROVES	YES
2	45	A	IMPROVES	NO
4	30	A	REMAINS	YES
3	35	A	REMAINS	NO
5	10	A	WORSENS	YES
4	20	A	WORSENS	NO
2	50	B	IMPROVES	YES
1	10	B	IMPROVES	NO
4	40	B	REMAINS	YES
3	35	B	REMAINS	NO
5	20	B	WORSENS	YES
4	55	B	WORSENS	NO

Table 4

Solution of the Problem. First we start by eliminating from de MOID the node "Training Plan" as it has a chance node as a parent and both are direct predecessors of the value node; we perform the uninon operation and eliminate then and discard the dominated solutions. The MOID will transform into the following structure:

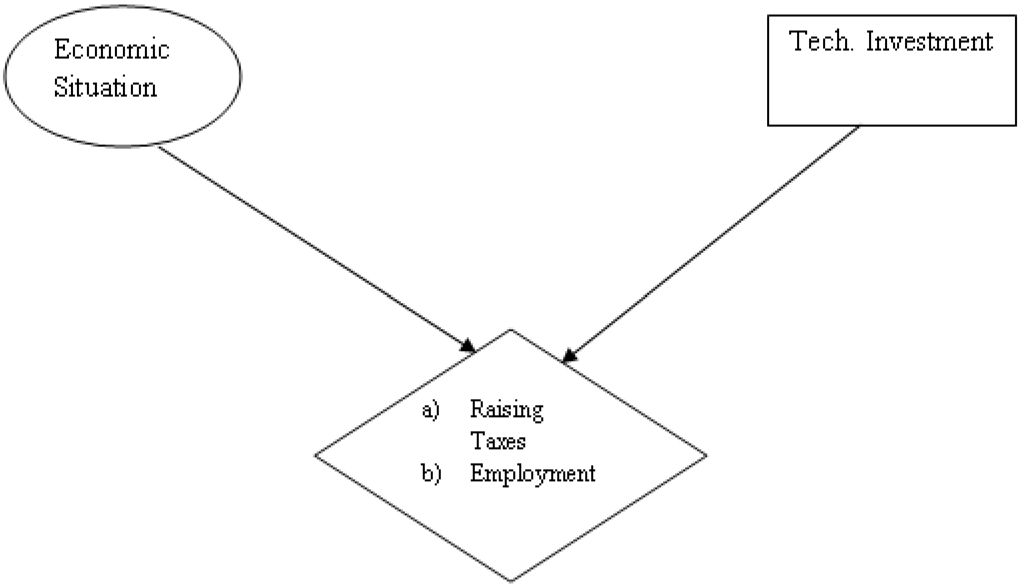

Fig. 7

At the numerical level, in the following table we mark with an asterisk the dominated solutions, and discard them for the following run. For each pair of alternatives (see table 5) of "Training Plan" we eliminate the dominated ones.

VALUE NODE		PARENT NODES		
RAISING TAXES	EMPLOYMENT	TECH. INVESTMENT	ECONOMIC SITUATION	TRAINING PLAN
3	60	A	IMPROVES	YES
2	45	A	IMPROVES	NO
*4	*30	A	REMAINS	YES
3	35	A	REMAINS	NO
*5	*10	A	WORSENS	YES
4	20	A	WORSENS	NO
2	50	B	IMPROVES	YES
1	10	B	IMPROVES	NO
4	40	B	REMAINS	YES
3	35	B	REMAINS	NO
*5	*20	B	WORSENS	YES
4	55	B	WORSENS	NO

Table 5

We can see how the strategies $[A; REMAINS, YES]$ is dominated by the strategy $[A; REMAINS, NO]$; we can see as well how $[B; WORSENS, YES]$ is dominated by $[B; WORSENS, NO]$.

In the following step we eliminate the chance node "Economic Situation" and perform the expectation operation by averaging out; we eliminate the dominated results obtaining the following numbers (marked with an asterisk are the dominated solutions which will be discarded for the following steps):

VALUE NODE		PARENT NODE	PREVIOUSLY REMOVED DECISION NODES		
RAISING TAXES	EMPLOYMENT	TECH. INVESTMENT	TRAINING PLAN GIVEN ECONOMIC IMPROVEMENT	TRAINING PLAN GIVEN ECONOMIC REMAINS	TRAINING PLAN GIVEN ECONOMIC WORSENS
3,2	37	A	YES	NO	NO
3	32	A	NO	NO	NO
3,6	45	B	YES	YES	NO
3	42	B	YES	NO	NO
*3,4	*37	B	NO	YES	NO
2,8	34	B	NO	NO	NO

Table 6

As a last step, we eliminate the node "Tech Investment" and perform the union operation amongst all the alternatives of the node (A and B); the optimization operation takes place and discard all dominated solutions. In the following table we show the final results:

VALUE NODE		PARENT NODE	PREVIOUSLY REMOVED DECISION NODES		
RAISING TAXES	EMPLOYMENT	TECH. INVESTMENT	TRAINING PLAN GIVEN ECONOMIC IMPROVEMENT	TRAINING PLAN GIVEN ECONOMIC REMAINS	TRAINING PLAN GIVEN ECONOMIC WORSENS
*3,2	*37	A	YES	NO	NO
*3	*32	A	NO	NO	NO
3,6	45	B	YES	YES	NO
3	42	B	YES	NO	NO
2,8	34	B	NO	NO	NO

Table 7

The final result shows how we end up having 3 optimal solutions to the problem. The set of optimal strategies are:

$$\begin{bmatrix} B; YES, IMPROVES \\ YES, REMAINS \\ NO, WORSENS \\ B; YES, IMPROVES \\ NO, REMAINS \\ NO, WORSENS \\ B; NO, IMPROVES \\ NO, REMAINS \\ NO, WORSENS \end{bmatrix}$$

3.4 Analysis of the Method

The method presented in this section for solving MOID (7) allows to solve multi objective optimization problems with a MOID using a Pareto Frontier, without the limitations of single value methods (9). The set of points that are Pareto efficient can be then studied using appropriate multicriteria decision making method.

This method does not imply preferences over certain alternatives, giving the decision maker a set of solutions and the flexibility of selecting the most appropriate one, when utility functions results are single values that might be not so easily understood by the decision maker.

This method may have limitations from the computational point of view. As the number of objectives increases we can have a large set of non-dominated solutions, which can be computationally intensive; single utility functions scale better in these contexts with multiple stage decisions.

4. Solving Multi Objective Influence Diagrams using the Estimation Refinement Method Technique

We present a new method for solving MOIDs using the Estimation Refinement Method (ERM) (10). First, we present the ERM technique and later, we will present our algorithm for solving MOID.

4.1 Introduction: Polyhedral approximation methods based on evaluation of a support function

In this section we describe iterative methods for the approximation of convex bodies by sequences of polytopes.

Let R^m be the Euclidean space with a distance *d*, let *C* be a convex body from R^m; ER methods for polyhedral approximation of the body *C*, are based on evaluating the support function of C.

$$g_C(u) = \max\{< u, y >: y \in C\},$$

for directions *u* that belong to the unit Sphere of directions

$$S = \{u \in R^m : < u, u > = 1\}.$$

We, assume that the support function value can be found for any $u \in S$.

In the case of a multi objective optimization problem, computing the value of the support function for a direction *u* from the unit ball consists in solving a convex optimization problem.

$g_Y(u) = \max\{< u, y > : y = f(x), x \in X\}$. where X is a decision space and

$y: X \to R^m, y = f(x)$ is a transformation from the decision space into the criteria space.

4.2 Estimation Refinement Method

The ERM is an adaptive method for polyhedral approximating the compact convex bodies, which is asymptotically optimal. A convex body C is approximated adaptively by a sequence P_0, P_1, ... , P_k, ... of internal polyhedra with the increasing number of vertices that belong to the boundary of C. The vertices of the polyhedron P_k include all vertices of $P_{(k-1)}$ plus a new vertex.

The optimality of the adaptive methods is related to their ability to adapt the directions $\{u_1, u_2, \dots, u_k, \dots\}$ in which the support function of the approximated body C is calculated, in order to find the next vertex. These methods need the support function calculation to identify directions through the approximation process (11). As we approximate the points to the body C , we construct the Convex Edgeworth-Pareto Hull (CEPH) of Y as a system of linear inequalities (see figure 6). We define the CEPH as the set of points $y \in R^m$ such that (minimization case):

$$Y_p = \{y \in R^m : y \geq f(x), x \in X\}$$

Eq. 4

A modification of the ERM is used for approximating the CEPH (see figure 7).

4.3 ERM Algorithm for Approximating the CEPH

The algorithm was formulated in (12). The initial approximation is given by a simplex, plus the restrictions corresponding to the marginal feasible values for each objective, described both as a list of m+1 vertices as well as a solution set of a linear inequality system (see figure 6).

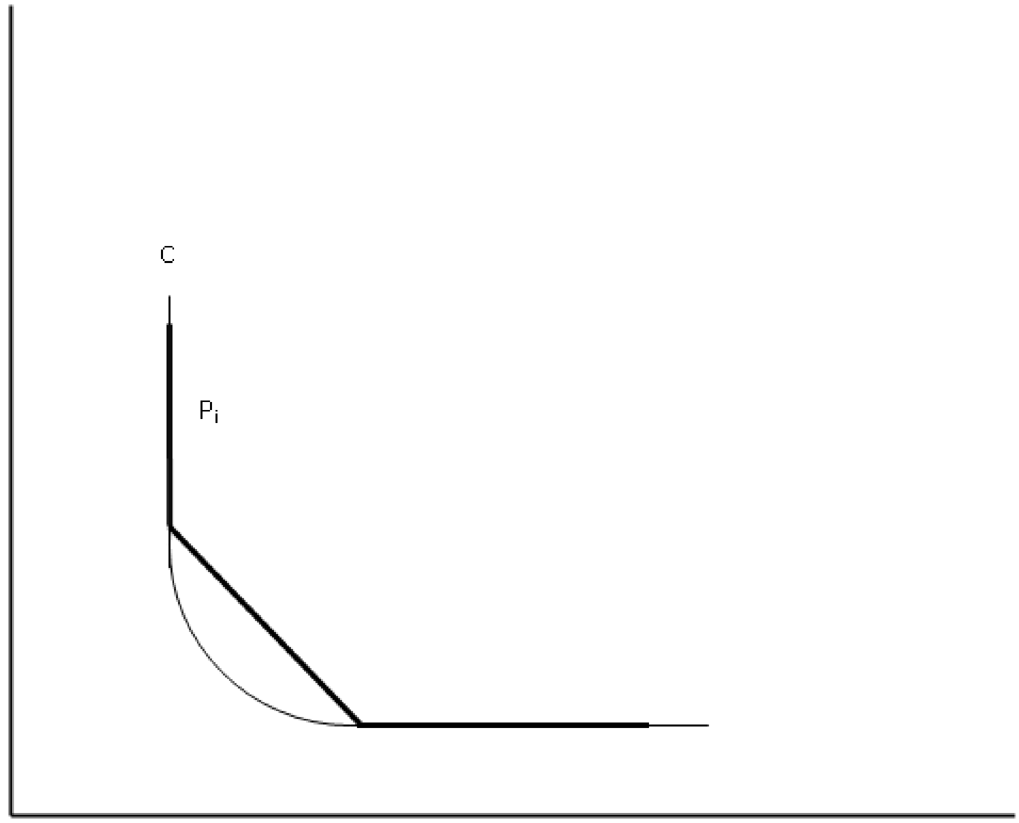

Figure 6

Let $U(P)$ denote the finite set of unit outer normals to the facets of the approximating internal polyhedron. The finite set $U(P)$ is defined if the polyhedron P is given in the form of a solution set of a linear inequality system.

Let us describe the $(k + 1)$-th iteration of the method. Prior to the iteration, we should have constructed the internal polyhedron P_i^k in the form of a solution set of the linear inequality system and in the form of a list of its vertices.

Step 1:

The direction u^* from $U(P^k)$ is found that solves

$$\max\{ (g_C(u) - g_{P^k}(u)) : u \in U(P^k) \};$$

If $\left| g_C(u^*) - g_{P^k}(u^*) \right| \le \varepsilon$ then stop

where ε is the given precision.

The point y^* is selected such as

$$< u^*, y^* >= g_C(u^*).$$

Step 2:

Let P_i^{k+1}= conv $\{y^*, P_i^k\}$

$U(P_i^{k+1})$ is constructed upon constructing conv $\{y^*, P_i^k\}$ in the form of a solution set of a linear inequality system.

4.4 Solving Influence diagrams with multiple objectives using ERM

Our approach to solve MOIDs using the ERM method will adopt the method for solving IDs introduced by Shachter (3) shown in section 2.2, as a base method for solving single objective IDs. We use as well the formalism presented in (4) for IDs with multiple value nodes and a single ALU node, to solve our MOID.

4.4.1 Structure of the diagram

In order to use the ERM technique for solving a MOID of the form given in 4.4, we will consider the following set of auxiliary diagrams. We will consider a MOID, $\mathcal{V}(u) = X \cup D \cup V \cup SU(u)$ with a super-value node $SU(u)$ where $u \in R^m$, $||u|| = 1$, such that:

- Every objective presented in the diagram will have its own value node V associated with it. Therefore if there are m objectives ($m = |V|$), we will have m value nodes.

- Every value node associated with an objective in an initial MOID, $\mathcal{V}$, will not be a terminal value node. We introduce a super value node $SU(u)$, similar to the one presented in (4). This super value node will be terminal and Additive Linear Utility type (ALU) as presented in 2.2.

- The ALU node will be represented by a vector u of m dimensions, where m is the number of objectives of the MOID $\boldsymbol{\mathcal{V}}$.

4.4.2 Solving MOID applying the ERM

In order to solve MOIDs, we will use an algorithm for solving uni-objective ID together with the ERM method. The initial diagram transformations applied in the method by Shachter will remain in our method.

The ALU node SU (u) is initialized with the weights of the respective objectives of the problem: $u_1, u_2, \ldots\ u_m$. In the framework of our method the array $u=\ (u_1, u_2, \ldots\ u_m)$ represents a direction in which the CEPH of the feasible set in the criteria space of the model, given by our MOID, has to be estimated.

At each run of the algorithm, new directions $U(P^{k+1})$ are calculated. We initialize the algorithm with default vectors $U(P^0)$= {(0, … , 0, -1), (0, … ,-1, 0) (-1, … , 0, 0)} in the case of minimization for all objectives.

Definitions: Let k be the number of iterations of the algorithm. Let Δ be a generic strategy for class of IDs(u), where u determines the weights of the ALU node. Let ID($\boldsymbol{\mathcal{V}}$ (u)) for a specific u be an optimal strategy of $\boldsymbol{\mathcal{V}}$ (u). Let $\boldsymbol{\mathcal{V}}$ (u, Δ) be the result of applying the strategy Δ to the ID $\boldsymbol{\mathcal{V}}$(u). Let $V_i(\Delta)=\boldsymbol{\mathcal{V}}(U(P^0)_i, \Delta)$, that is, $V_i(\Delta)$ is the result (value) of i-th objective when applying the strategy Δ to the MOID $\boldsymbol{\mathcal{V}}$. Let ε be the estimated precision of the approximation, and $\delta(P^k, u_i)=max\ \{<u, v>: v \in P^k\}$ the support function for P^k.

Algorithm. We now present the algorithm for solving MOID, integrating the ERM technique.

Step0

Initialize ALU node $\longleftarrow$ $U(P^0)$

for $i = 1, \ldots, m$ /*$m = |U(P^0)|$*/

$\Delta := ID(\mathcal{V}(u_i))$

for $j = 1, \ldots, m$

$v_i^j := V_i(\Delta)$

end

end

OUTPUT: $P^0 = \{x \in conv(v_i : i = 1, \ldots, m) : x + R^k\_(x)\}$

Initialize ALU node $\longleftarrow$ $U(P^k)$

Step k+1

for $i = 1, \ldots, |U(P^k)|$

$\Delta := ID(\mathcal{V}(u_i))$

for $j = 1, \ldots, m$

$v_i^j := V_i(\Delta)$

end

end

$i^* := Argmax\,\{f(i) = (v_i, u_i) - \delta(P^k, u_i) : i := 1, \ldots, |U(P^k)|\}$

$P^{k+1} := conv\,(P^k, v_{i^*})$

$d_{k+1} := f(i^*)$

If $d_{k+1} \leq \varepsilon$ *then stop*

Else: $k := k+1$

4.5 ERM ID Example

Economic MO problem (third version). We take the same example presented in section 3.3. solving it with the algorithm presented in section 4.4.2.

The structure of the decision problem is shown in the following MOID:

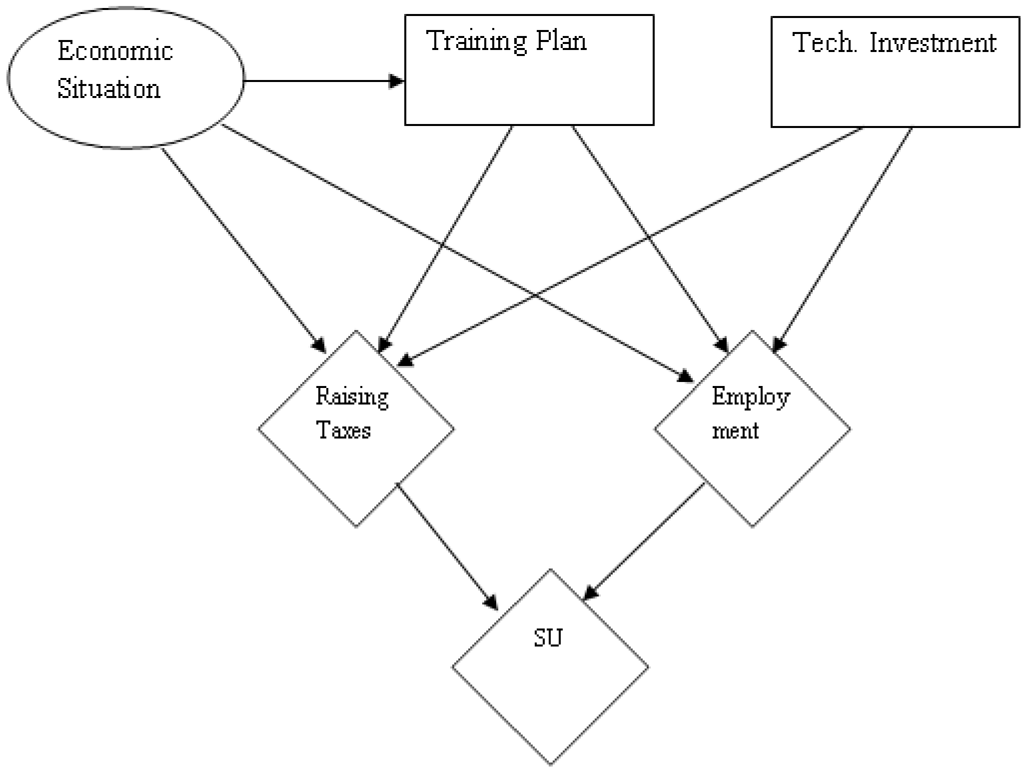

Figure 8

The values for each node are shown in table 8, and the values for each objective are shown in tables 9 and 10 (objectives are *"Raising Taxes"* and *"Employment"*):

DECISION NODE	ALTERNATIVES
Tech. Investment	
	A
	B

For A =7 millions; B=5 millions

DECISION NODE	ALTERNATIVES
Training Plan	
	YES
	NO

CHANCE NODE		VALUES
Economic situation		
	IMPROVES	0,2
	REMAINS	0,6
	WORSENS	0,2

Table 8

VALUE NODE	PARENT NODES		
RAISING TAXES	TECH. INVESTMENT	ECONOMIC SITUATION	TRAINING PLAN
3	A	IMPROVES	YES
2	A	IMPROVES	NO
4	A	REMAINS	YES
3	A	REMAINS	NO
5	A	WORSENS	YES
4	A	WORSENS	NO
2	B	IMPROVES	YES
1	B	IMPROVES	NO
4	B	REMAINS	YES
3	B	REMAINS	NO
5	B	WORSENS	YES
4	B	WORSENS	NO

Table 9

VALUE NODE	PARENT NODES		
EMPLOYMENT	TECH. INVESTMENT	ECONOMIC SITUATION	TRAINING PLAN
60	A	IMPROVES	YES
45	A	IMPROVES	NO
30	A	REMAINS	YES
35	A	REMAINS	NO
10	A	WORSENS	YES

20	A	WORSENS	NO
50	B	IMPROVES	YES
10	B	IMPROVES	NO
40	B	REMAINS	YES
35	B	REMAINS	NO
20	B	WORSENS	YES
55	B	WORSENS	NO

Table 10

Steps in the algorithm. First we start by applying the vector of weights u=(-1, 0) that is, we calculate the optimal policy for “Raising Taxes”.

As it was shown for the single objective case, first we eliminate the decision node “Training Plan” by optimal policy determination. After we eliminate the chance node “Economic Situation” and apply the expectation operation (see results in table 11).

VALUE NODE	PARENT NODES		
ALU NODE	TECH. INVESTMENT	ECONOMIC SITUATION	TRAINING PLAN
-1,8	REMAINS	A	NO
-0,4	IMPROVES	A	NO
-0,8	WORSENS	A	NO
-1,8	REMAINS	B	NO
-0,2	IMPROVES	B	NO
-0,8	WORSENS	B	NO

Table 11

As a last step we eliminate the decision node “Tech Investment” by optimal policy determination (see results in table 12).

VALUE NODE	PARENT NODES		
ALU NODE	ECONOMIC SITUATION	TECH. INVESTMENT	TRAINING PLAN
-1,8	REMAINS	A	NO
-0,4	IMPROVES	A	NO
-0,8	WORSENS	A	NO

-3			
-1,8	REMAINS	B	NO
-0,2	IMPROVES	B	NO
-0,8	WORSENS	B	NO
-2,8			

Table 12

As we can see the optimal policy will be

$$\mathrm{OP_1} = \begin{bmatrix} B; NO; REMAINS \\ NO; IMPROVES \\ NO; WORSENS \end{bmatrix}$$

We obtain now the value for the objective “Employment” given the policy

$$\mathrm{OP_1} = \begin{bmatrix} B; NO; REMAINS \\ NO; IMPROVES \\ NO; WORSENS \end{bmatrix}$$

We obtain the following results:

VALUE NODE	PARENT NODES		
EMPLOYMENT	ECONOMIC SITUATION	TECH. INVESTMENT	TRAINING PLAN
21	REMAINS	B	NO
2	IMPROVES	B	NO
11	WORSENS	B	NO
34			

Table 13

So the values for the optimal policy determined for weights (-1, 0) are [2,8; 34]; these points belong to the CEPH.

We continue by applying the weights u=(0, -1) and applying the same procedure shown before.

ALU NODE	ECONOMIC SITUATION	TECH. INVESTMENT	TRAINING PLAN
21	REMAINS	A	NO
12	IMPROVES	A	YES
4	WORSENS	A	NO

37			
24	REMAINS	B	YES
10	IMPROVES	B	YES
11	WORSENS	B	NO
45			

Table 14

The optimal value for *"Employment"* in this case is 45 and the optimal strategy is:

$$\mathrm{OP}_2 = \begin{bmatrix} \mathrm{B}; YES; REMAINS \\ YES; IMPROVES \\ NO; WORSENS \end{bmatrix}$$

We obtain now the value for the objective "Raising taxes" given the policy

$$\mathrm{OP}_2 = \begin{bmatrix} \mathrm{B}; YES; REMAINS \\ YES; IMPROVES \\ NO; WORSENS \end{bmatrix}$$

RAISING TAXES	ECONOMIC SITUATION	TECH. INVESTMENT	TRAINING PLAN
2,4	REMAINS	B	YES
0,4	IMPROVES	B	YES
0,8	WORSENS	B	NO
3,6			

Table 15

As we can see in table 15, the values for the optimal policy determined for weights (0,1) are [3,6 ; 45]. In figure 9, the initial approximation is represented.

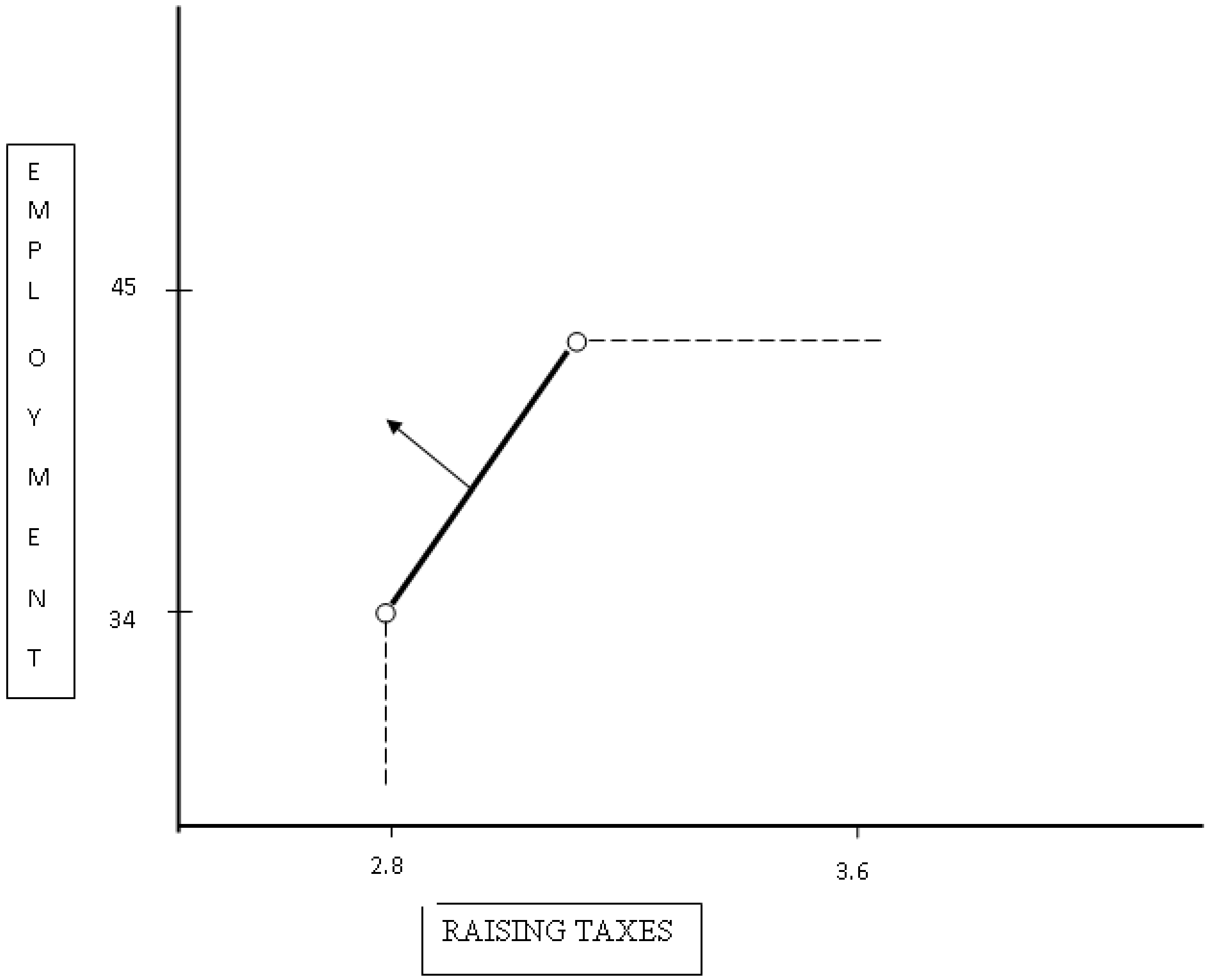

Figure 8

We have found the initial approximation. We will estimate now the hyperplane. We need to know the normal to it, that can be found using the coordinates of the two previously found points.

The resulting direction would be u = (-11, 0.8)/||-11, 0.8|| where ||.|| is the Euclidean norm. The result is found by taking into account that our objectives are the minimization of "Raising Taxes" and the maximization of employment, in figure 9 we can see the representation of the resulting direction.

We apply the vector of weights (-11, 0.8) to the ALU node and reinitialize the procedure, obtaining the following results (see table 16, 17 and 18).

ALU NODE	PARENT NODES		
	TECH. INVESTMENT	ECONOMIC SITUATION	TRAINING PLAN
15	A	IMPROVES	YES
14	A	IMPROVES	NO
-20	A	REMAINS	YES
-5	A	REMAINS	NO
-47	A	WORSENS	YES
-28	A	WORSENS	NO
18	B	IMPROVES	YES
-3	B	IMPROVES	NO
-12	B	REMAINS	YES
-5	B	REMAINS	NO
-39	B	WORSENS	YES
0	B	WORSENS	NO

Table 16

ALU NODE	TECH. INVESTMENT	ECONOMIC SITUATION	TRAINING PLAN
3	A	REMAINS	NO
-3	A	IMPROVES	YES
-5,6	A	WORSENS	NO
-5,6			
3,6	B	REMAINS	NO
-3	B	IMPROVES	YES
0	B	WORSENS	NO
0,6			

Table 18

The optimal policy would be:

$$\mathrm{OP}_3 = \begin{bmatrix} \mathrm{B}; NO, REMAINS \\ YES, IMPROVES \\ NO, WORSENS \end{bmatrix}$$

Now we calculate the values for *"Raising Taxes"* and *"Employment"* for OP_3

Raising Taxes (OP_3)= 3

Employment(OP_3)= 42

The resulting vector for OP_3 will be (3, 42)

We obtain 2 new directions: the normals of respective facets of CEPH passing through the points corresponding to the strategies OP_1 and OP_3 and OP_2 and OP_3 correspondingly:

(-7, 0.2) and(-3, 0.6) .

In figure 9 we show the new point and the 2 new directions obtained:

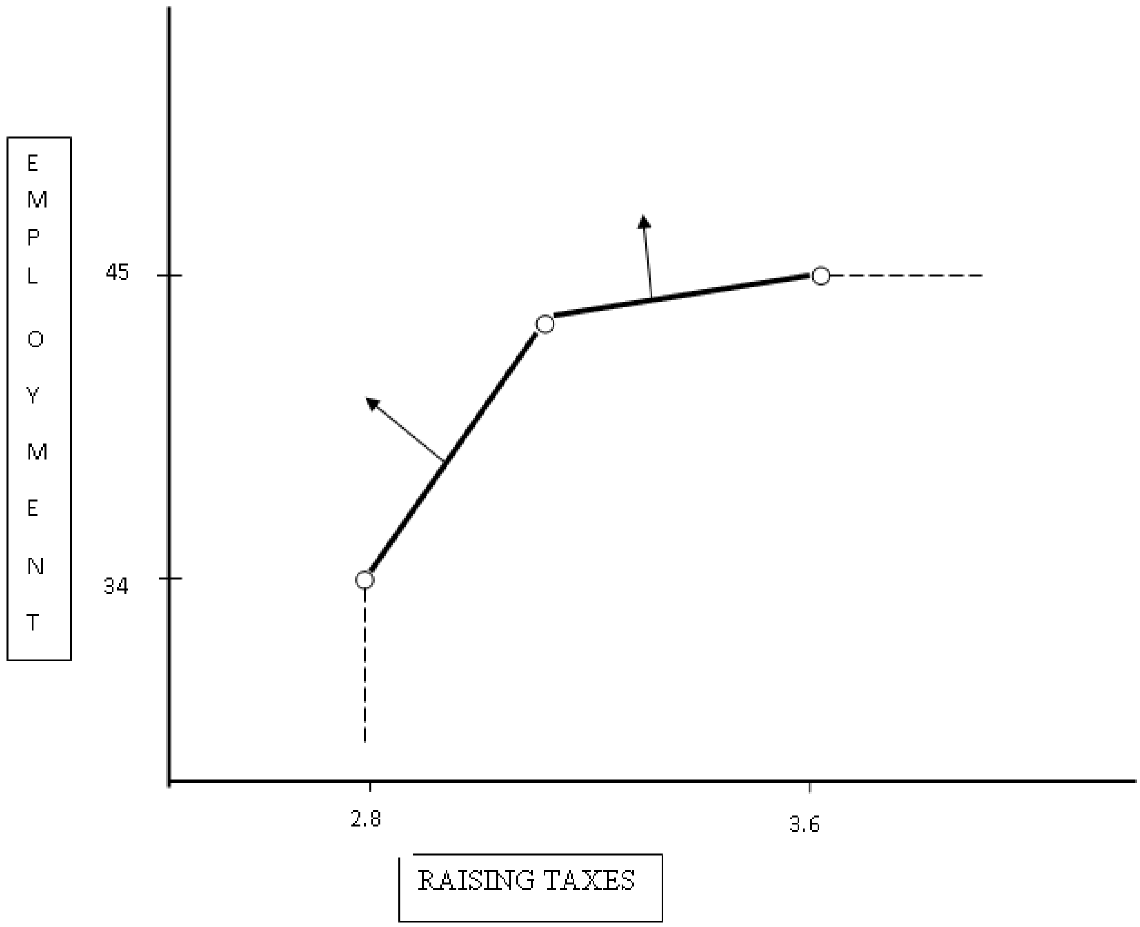

Figure 9

We initialize the procedure with the direction (-2.85, 0.57) getting the following results (see table 19):

ALU NODE	TECH. INVESTMENT	ECONOMIC SITUATION	TRAINING PLAN
7.87	A	REMAINS	NO
5.47	A	IMPROVES	YES

0.46	A	WORSENS	NO
13.79			
8,21	B	REMAINS	YES
4,79	B	IMPROVES	YES
4,45	B	WORSENS	NO
17.44			

Table 19

The optimal policy would be:

$$\mathrm{OP}_{4=}\begin{bmatrix} \mathrm{B}; YES, REMAINS \\ YES, IMPROVES \\ NO, WORSENS \end{bmatrix}$$

We reinitialize the procedure with the direction (-8, 0.2). obtaining the following optimal policy:

$$\mathrm{OP}_{5=}\begin{bmatrix} \mathrm{B}; NO, REMAINS \\ YES, IMPROVES \\ NO, WORSENS \end{bmatrix}$$

As we can see the optimal policies OP_4 and OP_5 have been already obtained, therefore the algorithm would stop at this point. The set of optimal policies will be constituted then by the following strategies.

$$\begin{bmatrix} B; NO, REMAINS \\ NO, IMPROVES \\ NO, WORST \\ \mathrm{B}; YES, REMAINS \\ YES, IMPROVES \\ NO, WORST \\ \mathrm{B}; NO, REMAINS \\ YES, IMPROVES \\ NO, WORST \end{bmatrix}$$

4.5.1 Analysis of the Method

In the method presented in 4.4.2 we have shown how the ERM algorithm for approximating the CEPH can be applied to solve MOOP with IDs.

The main advantages of our method can be outlined as follows:

1. The approximating process can be interrupted at any time, and we will still obtain an approximation of the CEPH and know the precision. In a method for the construction of the Pareto Frontier, like the one presented by Haimes et al., interruption may bring unpredictable results, and precise methods require the whole algorithm to be run until the end.
2. If the number of non-dominated strategies is high, this method will computationally perform well, while the method presented by Haimes may lead to intense computational requirements.
3. In our current method, any standard algorithm for solving a single objective ID can be used, while the method by Haimes has to be implemented from the beginning and existing algorithms cannot be used. However, note that the performance of the method of Haimes has not been tested yet, so its practical limitations are unknown.

One of the limitations of our method is the complexity created by increasing the number of objectives. This limitation is due to the calculation of the directions from each point. It has being experimentally proven that it can perform well up to 7-8 objectives (10). Another limitation is that the method constructs only extreme Pareto points (that is, the points that form the CEPH).

5. Results and Further Research

In this work we have presented the problem of solving multi objective influence diagrams (MOID). First, we have presented one standard algorithm (3) and solve an example with a single objective. Second, we have introduced MOID as it was develop by (7) and highlighted the main power of MOID to solve MOOP. Third, we have presented a new method for solving MOID using the ERM with an algorithm for solving IDs.

The algorithm developed shows how MOOP can be structured and solved using IDs with multiple value nodes and an ALU node. We combine the clearer representation of these IDs, already presented in (4), with the ERM technique for approximating the CEPH in MOOP, which simplifies the construction of the Pareto frontier and presents a set of feasible solutions in a clear manner.

This method, as mentioned in section 5, can perform very well with a large number of variables, but has limitations from the point of view of the number of objectives, having an upper limit of 7 to 8 objectives (10), new benchmarks would be of great use in order to see the performance of the ERM technique implemented in an ID setting, which we are considering in the near future.

Further research includes also using these methods to solve sequential decision making with multiple objectives.

References

1. **Matheson, R. A. Howard and J. E.** Influence Diagrams. [ed.] R.A. Howard and J.E.Matheson. *The Principles and Applications of Decision Analysis.* 1984, Vol. II, pp. 719-762.

2. **Shachter, R. and D. Bhattacharjya.** Solving influence diagrams: exact algorithms. [book auth.] J. J. Cochran. *Encyclopedia of Operations Research and Management Science* . s.l.: Wiley, 2010.

3. **Shachter, R. D.** Evaluating influence diagrams. *Operations Research.* Vol. 34, 6.

4. **J. A. Tatman, R. D. Shachter.** Dynamic programming and influence diagrams. *IEEE Transactions on Systems Science and Cybernetics.* 1990, Vol. 20.

5. **Haimes, Y. Y.** *Risk Modeling, Assessment, and Management.* New York: Wiley, 1998.

6. **G. E. Evans.** An overview of techniques for solving multiobjective mathematical programs. *Management Science.* 1984, Vol. 30, 11.

7. **Michael Diehl, Yacov Y.Haimes.** Influence Diagrams With Multiple Objectives and Trade Off Analysis. 2004, Vol. 34.

8. **Y. Y. Haimes, D. Li, and V. Tulsiani.** Multiobjective decision-tree analysis. *Risk Analysis.* 1990, Vol. 10, 1.

9. **Haimes, V. Changkong and Y.** *Multiobjective Decision Making: Theory and Methodology.* New York: Elsevier, 1983.

10. **Lotov, A. V., Bushenkov, V. A., and Kamenev, G. K.** *Interactive Decision Maps. Approximation and Visualization of Pareto Frontier.* Dordrecht : Kluwer, 2004.

11. **Lotov, V.A Bushenkov and A.V.** Methods for the construction and application of generalized reachable sets. 1982.

12. **Chernykh, O. L.** Approximation of the Pareto-hull of a convex set by polyhedral sets. *Computational Mathematics and Mathematical Physics.* 35:8, 1033–1039, 1995.