# Similarity Measures for Building Binary Utility Trees in the Approximate Evaluation of Influence Diagrams

Rafael Cabañas, Andrés Cano, and Manuel Gómez-Olmedo
Dept. Computer Science and Artificial Intelligence
University of Granada, CITIC-UGR, Spain
{rcabanas,acu,mgomez}@decsai.ugr.es

**Abstract.** Influence diagrams are a class of probabilistic graphical models used to represent and solve decision problems with uncertainty. The efficiency of their evaluation can be improved if probability and utility potentials are represented with binary trees instead of tables. The method for building a binary tree representing an approximate potential requires a similarity measure for comparing two potentials. Here we propose different similarity measures for building binary trees representing a utility potential and we test them with some IDs from the literature.

**Keywords:** probabilistic graphical models, influence diagrams, approximate computation, contextual-weak independencies, binary trees

## 1 Introduction

Influence Diagrams (IDs) [1, 2] provide an efficient framework for representing and solving decision problems with uncertainty. The quantitative information that defines an ID is given by a set of conditional probabilities for single variables given some other variables (*probability potentials*) and by a set of utility functions depending on given sets of variables (*utility potentials*). Traditionally, these potentials are represented using tables. However, the evaluation of large IDs with tables becomes unfeasible due to its computational cost. One solution consists of using alternative representations such as *binary trees* (BTs) [3–5] which is an efficient data structure for storing and managing quantitative information. This data structure takes advantage of *contextual-weak independencies* [6, 7] so that identical values can be grouped into a single one offering a compact storage. Moreover, when BTs are too large they can be pruned and converted into smaller trees leading to approximate encodings.

Given a potential represented as a table, there might be more than one equivalent BT with different sizes. Thus the task of building a minimal BT becomes crucial for the efficiency of the evaluation. In previous works [3, 5] a heuristic algorithm for building BTs from tables and pruning them is proposed. This method uses a similarity measure or divergence for comparing each of the intermediate BTs with the exact potential (e.g. *Kullback Leibler divergence*, *Cosine similarity*, *Extended Jaccard coefficient*, *Euclidean* distance, etc.). When building a BT representing a probability potential, Kullback Leibler divergence can be used. By contrast, in case of utilities, it is not clear which is the most suitable similarity measure. Herein we study some of them and explain how they can be used in an efficient way for building and pruning BTs representing utility potentials. These alternatives are tested with some IDs from the literature.

## 2 Basics

Let us first define the basic notation. We use upper-case letters for variables and lower-case for states. In this paper we will only consider discrete variables. Thus, the domain of a variable $X$ with $n$ possible states is $\Omega_X = \{x_1, x_2, \ldots, x_n\}$. Given a set of $m$ variables $\mathbf{X} = \{X_1, \ldots, X_m\}$, the domain of the joint variable is defined as $\Omega_{\mathbf{X}} = \times_{X_i \in \mathbf{X}} \Omega_{X_i}$, where $\times$ is the Cartesian product. Elements of $\Omega_{\mathbf{X}}$ are called *configurations* and denoted as $\mathbf{x}$.

### 2.1 Influence Diagrams

Influence diagrams (IDs) [2, 1] are a kind of probabilistic graphical model for solving decision problems under uncertainty. An ID over a set of chance and decision variables $\mathcal{U}_C \cup \mathcal{U}_D$ consists of a qualitative and a quantitative part. The qualitative part is a directed acyclic graph (DAG) $\mathcal{G}$ with three different types of nodes. *Chance* nodes (circles) are associated to the chance variables $\mathcal{U}_V$. *Decision* nodes (squares) are associated to decision variables $\mathcal{U}_D$. The set of *Utility* nodes (diamonds) is denoted as $\mathcal{U}_V$. The quantitative part is made of a set of *probability potentials* $\Phi$ (representing the uncertainty) and a set of *utility potentials* $\Psi$ (representing the user preferences). A probability potential over two disjoint sets of variables $\mathbf{X}_I$ and $\mathbf{X}_J$, denoted as $\phi(\mathbf{X}_I|\mathbf{X}_J)$, is a map $\phi : \Omega_{\mathbf{X}_{I\cup J}} \to [0, 1]$ such that $\sum_{\mathbf{x}_I \in \Omega_{\mathbf{X}_I}} \phi(\mathbf{x}_I|\mathbf{x}_J) = 1$ for each $\mathbf{x}_J \in \Omega_{\mathbf{X}_J}$. Similarly, a utility potential over $\mathbf{X}_I$, denoted as $\psi(\mathbf{X}_I)$, is a map $\psi : \Omega_{\mathbf{X}_I} \to \mathbb{R}$. Notice that utility potentials are not normalized. The immediate predecessors of a node $Y$ according to the $\mathcal{G}$ are called *parents* and denoted as $\Pi_Y$. For each chance node, a probability potential over the corresponding variable and its parents is defined, while, for each utility node, a utility potential over the parents should be assessed. The formal definition of an ID is:

**Definition 1** *An ID is a tuple* $\langle \mathcal{G}, \mathcal{U}_C, \mathcal{U}_D, \mathcal{U}_V, \Phi, \Psi \rangle$ *, where* $\mathcal{G}$ *is a DAG over* $\mathcal{U}_C \cup \mathcal{U}_D \cup \mathcal{U}_V$*, while* $\Phi = \{\phi(X, \Pi_X)\}_{X \in \mathcal{U}_C}$ *and* $\Psi = \{\psi(\Pi_U)\}_{U \in \mathcal{U}_V}$ *are sets of, respectively, probability and utility potentials.*

The decisions have a temporal order, $D_1, \ldots, D_n$, and the chance nodes are partitioned into a collection of disjoint sets $\mathcal{I}_i$ according to when they are observed. That is, there is a partial order $\mathcal{I}_0 \prec D_1 \prec \mathcal{I}_1 \prec \cdots \prec D_n \prec \mathcal{I}_n$. When evaluating an ID, we must identify an optimal strategy, denoted $\widehat{\Delta}$, maximizing the expected utility for the decision maker and compute the *maximum expected utility* $MEU(\widehat{\Delta})$. For each $D_i$, this optimal strategy includes an *optimal policy* $\widehat{\delta}_{D_i}$, which is a mapping that specifies the best action for the decision maker for each configuration in $\Omega_{\Pi_{D_i}}$. Assuming an additive decomposition of the utility, the $MEU(\widehat{\Delta})$ can be calculated as follows:

$$MEU(\widehat{\Delta}) = \sum_{\mathcal{I}_0} \max_{D_1} \cdots \max_{D_n} \sum_{\mathcal{I}_n} \prod_{X \in \mathcal{U}_C} \phi(X|\Pi_X) \left( \sum_{U \in \mathcal{U}_V} \psi(\Pi_U) \right) \tag{1}$$

### 2.2 Binary Trees

Traditionally, potentials have been represented using tables. However, alternative representations can be used to reduce the computation involved during the

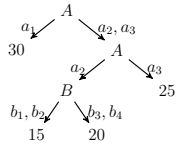
evaluation of an ID. For example, *binary trees* (BTs) are a compact data structure for representing potentials in BNs [5] and IDs [3, 4].

**Definition 2 (Binary Tree)** *A BT defined over the set of variables $\mathbf{X}_I$, denoted $\mathcal{BT}(\mathbf{X}_I)$, is a directed tree, where each internal node is labelled with a variable $X_i \in \mathbf{X}_I$ (random or decision), and each leaf node is labelled with a probability or a utility value. We use $L_t$ to denote the label of node $t$. Each internal node has always two children. We denote by $L_{lb(t)}$ and $L_{rb(t)}$ the labels (two subsets of $\Omega_{X_i}$) of the left and right branches of node $t$. Then, we denote by $t_l$ and $t_r$ the two children of $t$ ($t_r$ for the right child and $t_l$ for the left one).*

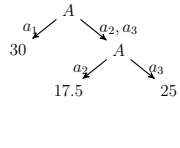
A binary tree $\mathcal{BT}(\mathbf{X}_I)$ represents a potential $\psi : \Omega_{\mathbf{X}_I} \rightarrow \mathbb{R}$ if for each $\mathbf{x}_I \in \Omega_{\mathbf{X}_I}$ the value $\psi(\mathbf{x}_I)$ is the number stored in the leaf node that is reached by starting from the root node and selecting the branch corresponding to state $x_i$ for each internal node labelled with $X_i \in \mathbf{X}_I$. Fig. 1 shows three different representations for the same utility potential.

| $\psi(A,B)$ | $b_1$ | $b_2$ | $b_3$ | $b_4$ |
|---|---|---|---|---|
| $a_1$ | 30 | 30 | 30 | 30 |
| $a_2$ | 15 | 15 | 20 | 20 |
| $a_3$ | 25 | 25 | 25 | 25 |

(a)




(b)




(c)

**Fig. 1.** Potential represented as (a) a table, (b) an exact BT and (c) a pruned BT

The main advantage of BTs is that they allow the specification of *contextual-weak independencies* [6, 7]. That is, identical values of the potential can be grouped into a single branch, allowing a smaller representation. For example, the table in Fig. 1 requires 12 values while the exact BT requires 7 nodes. If BTs are too large, they can be pruned and converted into smaller trees, thus leading to approximate algorithms. When a tree is pruned, leaves with similar values are represented with a single leaf labelled with their mean.

Given a node $t$ in a $\mathcal{BT}(\mathbf{X}_I)$, the set of its *ancestors* is denoted by $\mathbf{X}_I^t$ and the set of *available states* of $X_i$ at $t$ is denoted by $\Omega_{X_i}^t$. If $X_i \in \mathbf{X}_I^t$, then $\Omega_{X_i}^t$ is the set of states labelling the outgoing branch of $X_i$ in its last occurrence in the path from the root to $t$. Otherwise, $\Omega_{X_i}^t$ is equal to $\Omega_{X_i}$. The *associated extended configuration* for the node $t$ is the multi-set $\mathbf{A}^t = \{\Omega_{X_i}^t | X_i \in \mathbf{X}_I\}$. We denote by $\psi^{R(\mathbf{A}^t)}$ the utility potential consistent with $\mathbf{A}^t$; it corresponds to the sub-tree where $t$ is the root node. Fig. 2 shows an example of these definitions.

$\Omega_A^t = \{a_2, a_3\}$ $\quad$ $\Omega_B^t = \{b_1, b_2, b_3, b_4\}$

$\mathbf{A}^t = \{\{a_2, a_3\}, \{b_1, b_2, b_3, b_4\}\}$

| $\psi(A,B)^{R(\mathbf{A}^t)}$ | $b_1$ | $b_2$ | $b_3$ | $b_4$ |
|---|---|---|---|---|
| $a_2$ | 15 | 15 | 20 | 20 |
| $a_3$ | 25 | 25 | 25 | 25 |

**Fig. 2.** Sets of available states, extended configuration for the node $t$ and its consistent potential. Consider the BT shown in Fig. 1 (b) and that $t$ is the node labelled with $A$ whose outgoing arcs are labelled with $a_2$ and $a_3$ respectively.

**Building and pruning a BT** When a BT is built, variables should be sorted in such a way that the most informative variables are situated at the highest nodes in the tree. BTs can be built from tables using a top-down approach, choosing at each step a variable and two partitions of its states that maximize the *information gain*. At each step, a $\mathcal{BT}_j$ approximating the potential is generated. This process stops when the exact BT is obtained. Fig. 3 shows the process for building a BT representing $\psi(A, B)$ from Fig. 1. More details are given in [3–5].

**Definition 3 (Information Gain)** *Let $\psi$ be the potential (probability or utility) to be represented as a tree $\mathcal{BT}_j$ and $\mathcal{BT}_j(t, X_i, \Omega^{t_l}_{X_i}, \Omega^{t_r}_{X_i})$ the tree resulting of expanding the leaf node $t$ with the candidate variable $X_i$ and a partition of its available states into sets $\Omega^{t_l}_{X_i}$ and $\Omega^{t_r}_{X_i}$. Let $D(\psi, \mathcal{BT}_j)$ be the distance between a potential and a tree. The information gain can be defined as:*

$$I(t, X_i, \Omega^{t_l}_{X_i}, \Omega^{t_r}_{X_i}) = \left| D(\psi, \mathcal{BT}_j) - D(\psi, \mathcal{BT}_j(t, X_i, \Omega^{t_l}_{X_i}, \Omega^{t_r}_{X_i})) \right| \qquad (2)$$

For computing the information gain, we need to use a distance or a similarity measure between an exact potential and a tree (e.g. Kullback Leibler, Euclidean distance, cosine, etc.). When building a BT representing a probability potential, Kullback Leibler divergence will be used. In case of utilities, there are different alternatives that are explained in Sections 3 and 4.

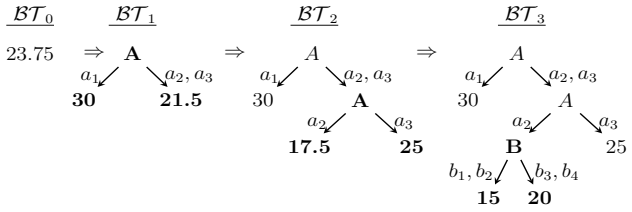



**Fig. 3.** Process for building a BT representing $\psi(A, B)$ from Fig. 1.

After the building process, the exact BT can be pruned in order to get an approximate but reduced one. Pruning a BT consists of replacing a terminal tree by the average value of its leaves. To decide if a terminal can be pruned, Eq. (2) is computed between the pruned and the non-pruned BTs. In general, the BT is pruned if the information gain is lower than a given threshold $\varepsilon$. However, this condition can vary depending on the similarity measure used (see Section 4.2). Depending on the variable and partitions chosen at each step, there might be more than one BT representing the same potential. For example, Fig. 4 shows two different BTs of different sizes representing the potential shown in Fig. 1. In general, we will try to obtain the smallest BT with similar values in close leaves in order to reduce the error when pruning.

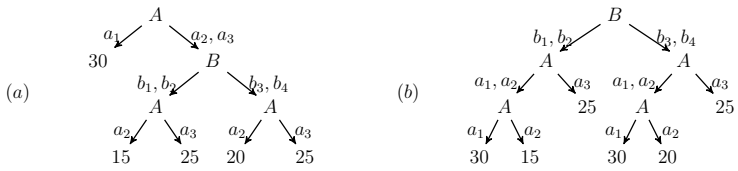



**Fig. 4.** Two different BTs of different sizes representing the potential in Fig. 1

## 3 Similarity between utility potentials

In previous section, a heuristic method for building and pruning a BT representing a potential was explained. The process is guided by the information gain (see Eq. (2)) which involves computing the distance or similarity between potentials. Herein we propose different alternatives for computing the distance between a utility potential $\psi$ and a $\mathcal{BT}$ approximating it.

### 3.1 Minkowski distances

Given two vectors of real numbers $x$ and $y$, the Minkowski distances are defined with the expression $D(x,y) = \left(\sum_{i=1}^{d} |x_i - y_i|^p\right)^{1/p}$. For $p = 2$ the Euclidean distance is obtained, and its expression for measuring the distance between a tree $\mathcal{BT}$ and a utility potential $\psi$ defined over $\mathbf{X}_I$ is:

$$D_{EU}(\psi, \mathcal{BT}) = \sqrt{\sum_{\mathbf{x}_I \in \Omega_{\mathbf{X}_I}} \left(\psi(\mathbf{x}_I) - \mathcal{BT}(\mathbf{x}_I)\right)^2} \tag{3}$$

The Euclidean distance can be normalized between $[0, 1]$ using Eq. (4). This new distance will be called *Euclidean normalized*. It can also be normalized using the exponential function and it will be called *Euclidean exponential* (see Eq. (5)).

$$D_{NORM}(\psi, \mathcal{BT}) = \frac{1}{1 + D_{EU}(\psi, \mathcal{BT})} \tag{4}$$

$$D_{EXP}(\psi, \mathcal{BT}) = e^{-D_{EU}(\psi, \mathcal{BT})^2} \tag{5}$$

In the Euclidean space, iso-similarities are concentric hyper-spheres around a considered point (see Fig. 5 (a),(b),(c)). Euclidean distance is translation invariant but scale variant.

### 3.2 Cosine Measure

Similarly, it can be used the cosine of the angle between two vectors [8]. The cosine distance between a tree $\mathcal{BT}$ and a utility potential $\psi$ defined on $\mathbf{X}_I$ is:

$$D_{COS}(\psi, \mathcal{BT}) = \frac{\sum_{\mathbf{x}_I \in \Omega_{\mathbf{X}_I}} \psi(\mathbf{x}_I) \cdot \mathcal{BT}(\mathbf{x}_I)}{\sqrt{\sum_{\mathbf{x}_I \in \Omega_{\mathbf{X}_I}} \psi(\mathbf{x}_I)^2} \cdot \sqrt{\sum_{\mathbf{x}_I \in \Omega_{\mathbf{X}_I}} \mathcal{BT}(\mathbf{x}_I)^2}} \tag{6}$$

Cosine measure is defined in the interval $[0, 1]$. It is decreasing: similar points have distance 1, while the highest distance is 0. Cosine distance is translation variant but scale invariant (see Fig. 5 (d)).

### 3.3 Extended Jaccard coefficient

Another popular similarity measure is the Extended Jaccard coefficient [9]. Let $\psi$ be a utility potential and a tree $\mathcal{BT}$ approximating the potential. Then, the Extended Jaccard coefficient is defined as follows:

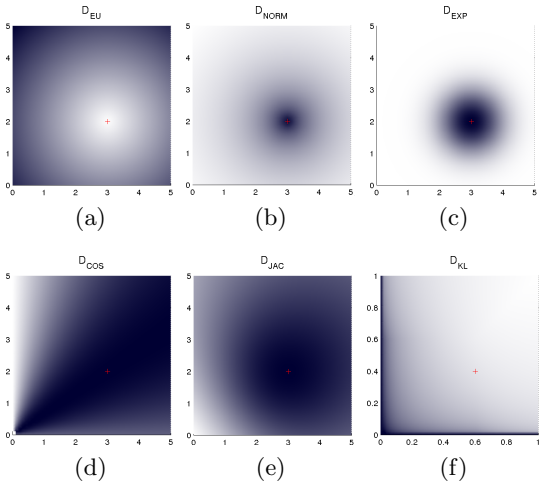



**Fig. 5.** Iso-similarities spaces with respect the considered the point (shown as a red cross). Darkest colours imply a higher value of the distance and vice-versa.

$$D_{JAC}(\psi, \mathcal{BT}) = \frac{\sum_{\mathbf{x}_I \in \Omega_{\mathbf{X}_I}} \psi(\mathbf{x}_I) \cdot \mathcal{BT}(\mathbf{x}_I)}{\left(\sum_{\mathbf{x}_I \in \Omega_{\mathbf{X_I}}} \psi(\mathbf{x}_I)^2\right) + \left(\sum_{\mathbf{x}_I \in \Omega_{\mathbf{X}_I}} \mathcal{BT}(\mathbf{x}_I)^2\right) - \sum_{\mathbf{x}_I \in \Omega_{\mathbf{X}_I}} \psi(\mathbf{x}_I) \cdot \mathcal{BT}(\mathbf{x}_I)} \tag{7}$$

The Extended Jaccard coefficient is defined in the interval $[0, 1]$. It is decreasing: similar points have distance 1, while the highest distance is 0. The iso-similarities are non-concentric hyper-spheres (see Fig. 5 (e)).

### 3.4 Kullback Leibler divergence

Traditionaly, Kullback Leibler divergence [10] has been used for measuring the discrepancy between two probability distributions. It is given by:

$$D_{KL}(\psi, \mathcal{BT}) = \sum_{\mathbf{x_i} \in \Omega_{\mathbf{x}_I}} \psi(\mathbf{x}_I) \cdot log \frac{\psi(\mathbf{x}_I)}{\mathcal{BT}(\mathbf{x}_I)} \tag{8}$$

Kullback Leibler divergence is a non-negative measure where the maximum similarity is given by the value 0 (see Fig. 5 (e)). The minimum similarity is given by the value $\infty$. Since Kullback Leibler divergence requires the potentials to be non-negative, we propose applying a transformation to the utility potentials similar to that proposed by Cooper [11, 12].

**Proposition 1.** *Let $\Psi = \{\psi(\Pi_U)\}_{U \in \mathcal{U}_V}$ be the set of utility potentials of an ID, and let max and min be the maximum and minimum utility values specified in the ID for all possible utility nodes and parent configurations. An equivalent normalized ID can be obtained if each utility potential $\psi(\Pi_U)$ is replaced by a new utility potential $\psi'(\Pi_U)$ such that:*

$$\psi'(\pi_U) = \frac{\psi(\pi_U) - min}{max - min} \qquad \forall \pi_U \in \Omega_{\Pi_U} \tag{9}$$

# 4 Building and Pruning with Similarity Measures

## 4.1 Information Gain Computation

The general scheme for building and pruning a $\mathcal{BT}$ representing a utility potential $\psi$ is explained in Section 2.2. Herein we detail how this method can be adapted for each similarity measure given in the previous section. In particular, it is only required to explain how the information gain (Eq. (2)) is calculated. Suppose that we aim to represent a utility potential $\psi$ as a tree. Let $\mathcal{BT}_j$ be an intermediate tree in the building process and $\mathcal{BT}_j(t, X_i, \Omega^{t_l}_{X_i}, \Omega^{t_r}_{X_i})$ the tree resulting of expanding the leaf node $t$ with the candidate variable $X_i$ and a partition of its available states into sets $\Omega^{t_l}_{X_i}$ and $\Omega^{t_r}_{X_i}$. If the Euclidean distance (Eq. (3)) is the similarity measure used, then the information gain $I_{EU}$ can be calculated in the following way:

$$I_{EU}(t, X_i, \Omega^{t_l}_{X_i}, \Omega^{t_r}_{X_i}) = \left| \sqrt{sumSqr(\psi^{R(A^t)}) - \frac{(sum(\psi^{R(A^t)}))^2}{size(\psi^{R(A^t)})}} - \sqrt{sumSqr(\psi^{R(A^{t_l})}) - \frac{(sum(\psi^{R(A^{t_l})}))^2}{size(\psi^{R(A^{t_l})})} + sumSqr(\psi^{R(A^{t_r})}) - \frac{(sum(\psi^{R(A^{t_r})}))^2}{size(\psi^{R(A^{t_r})})}} \right| \tag{10}$$

where $sum(\psi)$ is the addition of all the values of a potential $\psi$, $sumSqr(\psi)$ is the addition of all its square values and $size(\psi)$ is the number of values in $\psi$. Due to space restrictions we do not include the proof of this expression. The first and second terms correspond with $D_{EU}(\psi, \mathcal{BT}_j)$ and $D_{EU}(\psi, \mathcal{BT}_j(t, X_i, \Omega^{t_l}_{X_i}, \Omega^{t_r}_{X_i}))$ respectively. Thus, expressions for computing the information gain using the euclidean normalized and exponential distances can be easily obtained from Eq. (2), (4) and (5). Similarly, using Eq. (2) and (6), the expression for computing the information gain with cosine similarity is:

$$I_{COS}(t, X_i, \Omega^{t_l}_{X_i}, \Omega^{t_r}_{X_i}) = \left| \frac{\frac{(sum(\psi^{R(A^t)}))^2}{size(\psi^{R(A^t)})}}{\sqrt{sumSqr(\psi^{R(A^t)})} \cdot \sqrt{\frac{(sum(\psi^{R(A^t)}))^2}{size(\psi^{R(A^t)})}}} - \frac{\frac{(sum(\psi^{R(A^{t_l})}))^2}{size(\psi^{R(A^{t_l})})} + \frac{(sum(\psi^{R(A^{t_r})}))^2}{size(\psi^{R(A^{t_r})})}}{\sqrt{sumSqr(\psi^{R(A^{t_l})}) + sumSqr(\psi^{R(A^{t_r})})} \cdot \sqrt{\frac{(sum(\psi^{R(A^{t_l})}))^2}{size(\psi^{R(A^{t_l})})} + \frac{(sum(\psi^{R(A^{t_r})}))^2}{size(\psi^{R(A^{t_r})})}}} \right| \tag{11}$$

If we use the extended Jaccard coefficient (see Eq. (7)), the expression for computing the information gain is defined as follows:

$$I_{JAC}(t, X_i, \Omega^{t_l}_{X_i}, \Omega^{t_r}_{X_i}) = \left| \frac{\frac{(sum(\psi^{R(A^t)}))^2}{size(\psi^{R(A^t)})} - \frac{(sum(\psi^{R(A^{t_l})}))^2}{size(\psi^{R(A^{t_l})})} - \frac{(sum(\psi^{R(A^{t_r})}))^2}{size(\psi^{R(A^{t_r})})}}{sumSqr(\psi^{R(A^t)})} \right| \tag{12}$$

The information gain using Kullback Leibler divergence can be computed using Eq. (13). Note that for using this similarity measure the utility potential $\psi$ must be normalized using Proposition 1.

$$I_{KL}(t, X_i, \Omega^{t_l}_{X_i}, \Omega^{t_r}_{X_i}) = \Big| sum(\psi^{R(A^t)}) \cdot \log(\left|\Omega^t_{X_i}\right| / sum(\psi^{R(A^t)})) + sum(\psi^{R(A^{t_l})}) \cdot \log(sum(\psi^{R(A^{t_l})}) / \left|\Omega^{t_l}_{X_i}\right|) + sum(\psi^{R(A^{t_r})}) \cdot \log(sum(\psi^{R(A^{t_r})}) / \left|\Omega^{t_r}_{X_i}\right|) \Big| \quad (13)$$

The information gain as given in Eq. (10),(11),(12) or (13) only depends on the values of the potential consistent with node $t$, and it can be locally computed. Moreover, some computations may have already been done: during the building process, $sum(\psi^{R(A^t)})$ and $sumSqr(\psi^{R(A^t)})$ were computed when the father node was expanded. Similarly, when pruning, $sum(\psi^{R(A^{t_l})})$, $sumSqr(\psi^{R(A^{t_l})})$, $sum(\psi^{R(A^{t_r})})$ and $sumSqr(\psi^{R(A^{t_r})})$ were calculated when children were pruned.

### 4.2 Pruning Condition

Depending on the measure used, the pruning conditions may change slightly. Suppose that $t$ is the root of a terminal tree labelled with $X_i$, $t_l$ and $t_r$ its children, $\Omega^{t_l}_{X_i}$ and $\Omega^{t_r}_{X_i}$ the sets of states for the left and right child respectively. If Euclidean distance is used, the terminal tree is pruned if:

$$I(t, X_i, \Omega^{t_l}_{X_i}, \Omega^{t_r}_{X_i}) \leq \varepsilon \cdot (max - min) \quad (14)$$

where $max$ and $min$ are the maximum and minimum utility values in the ID. The remaining measures are either normalized (Euclidean normalized, Euclidean exponential, Cosine and Extended Jaccard) or they compare normalized potentials (Kullback Leibler divergence). Thus the pruning condition will be:

$$I(t, X_i, \Omega^{t_l}_{X_i}, \Omega^{t_r}_{X_i}) \leq \varepsilon \quad (15)$$

## 5 Empirical Validation

For an empirical validation, we consider six IDs from the literature [13–17]. These IDs are evaluated using BTs for representing the potentials and considering the similarity measures explained in Sections 3 and 4 for building and pruning them. The evaluation algorithm used is *variable elimination* (VE) [18, 19]. Fig. 6 shows computation time for the exact evaluation ($\varepsilon = 0.0$) of each ID using different similarity measures. Notice that the time measured also includes the time for building and pruning the BTs. The computation time is similar for all the similarity measures. Thus we conclude that none of them introduce a large overhead.

When evaluating an ID with pruned BTs there are two objectives to consider: memory requirements and error of the approximation. These two objectives are controlled with the threshold for pruning $\varepsilon$. High values of $\varepsilon$ will evaluate an ID with low memory requirements but with a high error and vice-versa. Here we aim to test if using any of the similarity measures, best approximate solutions are obtained. For that purpose, we evaluate each ID using each similarity measure and different threshold values $\varepsilon$ in the interval $[0, 1]$. For each evaluation,

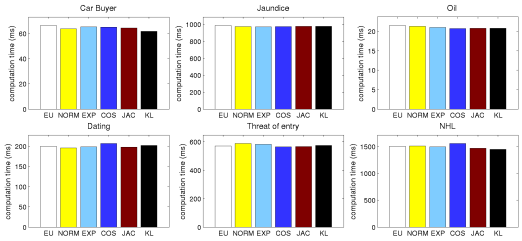



**Fig. 6.** Time for the exact evaluation of each ID using each similarity measure.

we measure the size of all the potentials along the evaluation and the error produced for computing the MEU (see Eq. (1)). Given a similarity measure, the set of pairs *(error, size)* compose a solution set. Pareto front and hyper-volume are computed for every solution set. The hyper-volume [20] is a unary indicator that measures the area of the dominated portion of the space. It is defined in the interval $[0, 1]$, being 1 the optimal solution and 0 the worst. In other words, a higher hyper-volume value stands for a better approximate solution. Table 1 shows the hyper-volume values obtained for each ID and similarity measure. It can be observed that results obtained with Euclidean distance (EU) are, in general, better than those obtained with the rest of measures.

**Table 1.** Hyper-volume values for each ID and similarity measure obtained from comparing error versus memory requirements.

| | EU | NORM | EXP | COS | JAC | KL |
|---|---|---|---|---|---|---|
| Car Buyer | **0.139** | 0.129 | 0.125 | 0.041 | 0.138 | 0.138 |
| Jaundice | 0.859 | 0.805 | 0.804 | 0.803 | **0.861** | 0.86 |
| Oil | **0.442** | 0.407 | 0.193 | 0.22 | 0.192 | 0.192 |
| Dating | **0.227** | 0.22 | 0.215 | 0.095 | 0.059 | 0.215 |
| Threat of entry | **0.78** | 0.684 | 0.684 | 0.685 | 0.685 | 0.684 |
| NHL | **0.668** | 0.652 | 0.649 | 0.65 | 0.522 | 0.589 |

## 6 Conclusions and Future Work

In this paper we have proposed different similarity measures for comparing the utility potentials involved in an ID. It has been explained how they can be used in an efficient way during the process of building and pruning a BT representing a utility potential. In the experimental part, these similarity measures have been compared showing that, all of them have a similar cost in terms of time but better approximate solutions are obtained using the Euclidean distance.

## Acknowledgments

This research was supported by the Spanish Ministry of Economy and Competitiveness under project TIN2013-46638-C3-2-P, the European Regional Development Fund (FEDER), the FPI scholarship program (BES-2011-050604). The authors have also been partially supported by "Junta de Andalucía" under project P10-TIC-06016.

## References


1. Olmsted, S.: Representing and solving decision problems. Dissertation Abstracts International Part B: Science and Engineering **45**(3) (1984)
2. Howard, R.A., Matheson, J.E.: Influence diagram retrospective. Decision Analysis **2**(3) (2005) 144–147
3. Cabañas, R., Gómez-Olmedo, M., Cano, A.: Approximate inference in influence diagrams using binary trees. In: Proceedings of the Sixth European Workshop on Probabilistic Graphical Models (PGM-12). (2012)
4. Cabañas, R., Gómez-Olmedo, M., Cano, A.: Using binary trees for the evaluation of influence diagrams. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems (in press)
5. Cano, A., Gómez-Olmedo, M., Moral, S.: Approximate inference in Bayesian networks using binary probability trees. International Journal of Approximate Reasoning **52**(1) (2011) 49–62
6. Boutilier, C., Friedman, N., Goldszmidt, M., Koller, D.: Context-specific independence in Bayesian networks. In: Proceedings of the 12th International Conference on UAI, Morgan Kaufmann Publishers Inc. (1996) 115–123
7. Wong, S., Butz, C.: Contextual weak independence in Bayesian networks. In: Proceedings of the 15th conference on Uncertainty in AI, Morgan Kaufmann Publishers Inc. (1999) 670–679
8. Dhillon, I., Modha, D.: Concept decompositions for large sparse text data using clustering. Machine learning **42**(1) (2001) 143–175
9. Strehl, A., Strehl, E., J., G.: Value-based customer grouping from large retail data-sets. In: Proceedings of the SPIE Conference on Data Mining and Knowledge Discovery. (2000) 33–42
10. Cover, T.M., Thomas, J.A.: Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing). Wiley-Interscience (2006)
11. Cooper, G.F.: A method for using belief networks as influence diagrams. In: Proceedings of the 4th Workshop on UAI. (1988) 55–63
12. De Campos, C., Ji, Q.: Strategy selection in influence diagrams using imprecise probabilities. (2008) 121–128
13. Hugin Expert network repository. `http://www.hugin.com/technology/samples`
14. Bielza, C., Gómez, M., Insua, R.S., Fernández del Pozo, J.A., Barreno García, P., Caballero, S., Sánchez Luna, M.: IctNEO system for jaundice management. Revista de la Real Academia de Ciencias Exactas, Físicas y Naturales **92**(4) (1998) 307–315
15. Qi, R., Zhang, L., Poole, D.: Solving asymmetric decision problems with influence diagrams. In: Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann Publishers Inc. (1994) 491–497
16. König, C.L.: Representing asymmetric decision problems with decision analysis networks. Master Thesis Advanced Artificial Intelligence (2012)
17. Lucas, P., Taal, B.: Computer-based decision support in the management of primary gastric non-Hodgkin lymphoma. UU-CS (1998-33) (1998)
18. Zhang, N., Poole, D.: Exploiting causal independence in Bayesian network inference. Journal of Artificial Intelligence Research **5** (1996) 301–328
19. Jensen, F., Nielsen, T.D.: Bayesian networks and decision graphs. Springer Verlag (2007)
20. Zitzler, E., Brockhoff, D., Thiele, L.: The hypervolume indicator revisited: On the design of Pareto-compliant indicators via weighted integration. In: Evolutionary Multi-Criterion Optimization, Springer (2007) 862–876