On SPI for Evaluating Influence Diagrams

Rafael Cabañas[1], Anders L. Madsen[2], Andrés Cano[1], and Manuel Gómez-Olmedo[1]

[1] Department of Computer Science and Artificial Intelligence CITIC,
University of Granada, Spain
{rcabanas,mgomez,acu}@decsai.ugr.es
[2] HUGIN EXPERT A/S and Department of Computer Science,
Aalborg University Aalborg, Denmark
anders@hugin.com

Abstract. An Influence Diagram is a probabilistic graphical model used to represent and solve decision problems under uncertainty. Its evaluation requires to perform a series of combinations and marginalizations with the potentials attached to the Influence Diagram. Finding an optimal order for these operations, which is NP-hard, is an element of crucial importance for the efficiency of the evaluation. The SPI algorithm considers the evaluation as a combinatorial factorization problem. In this paper, we describe how the principles of SPI can be used to solve Influence Diagrams. We also include an evaluation of different combination selection heuristics and a comparison with the variable elimination algorithm.

Keywords: Influence Diagrams, Combinatorial Factorization Problem, Exact Evaluation, Heuristic Algorithm.

1 Introduction

Influence Diagrams (IDs) [1,2] provide a framework to model decision problems with uncertainty for a single decision maker. The goal of evaluating an ID is to obtain the best option for the decision maker (*optimal policy*) and its utility.

Most of the evaluation algorithms proposed in the literature [3,4,5,6] require to perform a series of combinations and marginalizations with the probability and utility functions (potentials). Finding an optimal order for these operations, which is NP-hard [7], is an element of crucial importance for the efficiency of the evaluation. Thus the evaluation of an ID can be seen as a combinatorial factorization problem. This idea was already used to make inference in Bayesian Networks (BNs) with the first version of Symbolic Probabilistic Inference algorithm (SPI) [8] and with an improved algorithm in the SPI family called set-factoring [9]. In a related work [10] some experiments with SPI were performed to evaluate decision networks, however it was not given any detail of the algorithm. In this paper we describe the SPI algorithm for evaluating IDs taking into account the differences of an ID: two kind of potentials, the temporal order between decisions, etc. The experimental work shows how SPI can improve the efficiency of the evaluation on some IDs and different combination selection heuristics are compared.

The paper is organized as follows: Section 2 introduces some basic concepts about IDs and the motivation of this work; Section 3 describes how SPI can be used for evaluating IDs; Section 4 includes the experimental work and results; finally Section 5 details our conclusions and lines for future work.

A. Laurent et al. (Eds.): IPMU 2014, Part I, CCIS 442, pp. 506–516, 2014.

2 Preliminaries

2.1 Influence Diagrams

An ID [1,2] is a probabilistic graphical model for decision analysis under uncertainty with three kinds of nodes: *chance nodes* (circles) that represent random variables; *decision nodes* (squares) that correspond with the actions which the decision maker can control; and *utility nodes* (hexagons) that represent decision maker preferences.

We denote by $\mathscr{U}_C$ the set of chance nodes, by $\mathscr{U}_D$ the set of decision nodes, and by $\mathscr{U}_V$ the set of utility nodes. The decision nodes have a temporal order, $D_1,\ldots,D_n$, and the chance nodes are partitioned into a collection of disjoint sets according to when they are observed: $\mathscr{I}_0$ is the set of chance nodes observed before D_1, and $\mathscr{I}_i$ is the set of chance nodes observed after decision D_i is taken and before decision D_{i+1} is taken. Finally, $\mathscr{I}_n$ is the set of chance nodes observed after D_n. That is, there is a partial order: $\mathscr{I}_0 \prec D_1 \prec \mathscr{I}_1 \prec \cdots \prec D_n \prec \mathscr{I}_n$. Fig. 1 shows an example of an ID.

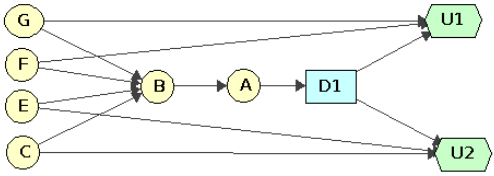

Fig. 1. An ID for a decision problem with one decision $D1$. The set of chance variables is partitioned into the sets: $\mathscr{I}_0 = \{A\}$ and $\mathscr{I}_1 = \{B,C,E,F,G\}$. The utility function is a sum of two local utilities, one associated to $D1$, G, and F and the other associated to $D1$, C, and E.

In the description of an ID, it is more convenient to think in terms of predecessors: the parents of a chance node X_i, denoted $pa(X_i)$, are also called *conditional predecessors*. The parents of a utility node V_i, denoted $pa(V_i)$, are also called conditional predecessors. Similarly, the parents of a decision D_i are called *informational predecessors* and are denoted $pa(D_i)$. Informational predecessors of each decision D_i, must include previous decisions and their informational predecessors (*no-forgetting assumption*).

The *universe* of the ID is $\mathscr{U} = \mathscr{U}_C \cup \mathscr{U}_D = \{X_1,\ldots,X_m\}$. Let us suppose that each variable X_i takes values on a finite set $\Omega_{X_i} = \{x_1,\ldots,x_{|\Omega_{X_i}|}\}$. Each chance node X_i has a conditional probability distribution $P(X_i|pa(X_i))$ associated. In the same way, each utility node V_i has a utility function $U(pa(V_i))$ associated. In general, we will talk about potentials (not necessarily normalized). The set of all variables involved in a potential ϕ is denoted $dom(\phi)$, defined on $\Omega_{dom(\phi)} = \times\{\Omega_{X_i}|X_i \in dom(\phi)\}$. The elements of $\Omega_{dom(\phi)}$ are called configurations of ϕ. Therefore, a *probability potential* denoted by ϕ is a mapping $\phi : \Omega_{dom(\phi)} \to [0,1]$. A *utility potential* denoted by ψ is a mapping $\psi : \Omega_{dom(\psi)} \to \mathbb{R}$. The set of probability potentials is denoted by Φ while the set of utility potentials is denoted by Ψ.

2.2 Motivation

The goal of evaluating an ID is to obtain an *optimal policy* δ_i for each decision D_i, that is a function of a subset of its informational predecessors. The optimal policy maximizes the *expected utility* for the decision.

Optimal policy: *Let ID be an influence diagram over the universe $\mathscr{U} = \mathscr{U}_C \cup \mathscr{U}_D$ and let $\mathscr{U}_V$ be the set of utility nodes. Let the temporal order of the variables be described as $\mathscr{I}_0 \prec D_1 \prec \mathscr{I}_1 \prec \cdots \prec D_n \prec \mathscr{I}_n$. Then, an optimal policy for D_i is*

$$\delta_{D_i}(\mathscr{I}_0, D_1, \ldots, \mathscr{I}_{i-1}) = \arg\max_{D_i} \sum_{\mathscr{I}_i} \max_{D_{i+1}} \cdots \max_{D_n} \sum_{\mathscr{I}_n} \prod_{X \in \mathscr{U}_C} P(X|pa(X)) \left(\sum_{V \in \mathscr{U}_V} U(pa(V)) \right) \quad (1)$$

For example consider the ID shown in Fig. 1. The optimal policy for D_1 can be calculated directly from Eq.1:

$$\delta_{D_1}(A) = \arg\max_{D_1} \sum_{G,F,E,C,B} P(G)P(F)P(E)P(C)P(B|C,E,F,G)P(A|B)U \quad (2)$$

where $U = (U_1(G,F,D_1) + U_2(E,C,D_1))$ is the sum of the local utilities. The table representing the joint probability of all chance variables might be too large. For that reason, some evaluation algorithms such as *Variable Elimination* (VE) for IDs [11] re-order the marginalizations of the variables as follows:

$$\delta_{D_1}(A) = \arg\max_{D_1} \sum_{G} P(G) \sum_{F} P(F) \sum_{E} P(E) \sum_{C} P(C) \sum_{B} P(B|C,E,F,G)P(A|B)U \quad (3)$$

Assuming that all the variables are binary and considering only the computations between probability potentials, the calculation of $\delta_{D_1}(A)$ requires 124 multiplications and 64 additions to marginalize out the variables in $\mathscr{I}_0$. Independently of the elimination ordering used to solve this ID, VE will always have to combine the marginal potentials with a large potential such as $P(B|C,E,F,G)$ or $P(A|C,E,F,G)$. However, with a re-order of the operations this situation can be avoided:

$$\delta_{D_1}(A) = \arg\max_{D_1} \sum_{B,C,E} \left((P(A|B)\,(P(E)P(C))) \left(\sum_{F,G} P(B|C,E,F,G)\,(P(F)P(G))\,U \right) \right) \quad (4)$$

Using Eq.4 the calculation of the optimal policy requires 72 multiplications and 32 additions. In some cases it could be better to combine small potentials even if they do not share any variable (e.g., $P(E)$ and $P(C)$). This combination will never be performed using VE since it is guided by the elimination ordering. Thus the efficiency of the evaluation can be improved if an optimization in the order of both operations, marginalization and combination, is performed [9].

3 Symbolic Probabilistic Inference

3.1 Overview

As VE does, SPI removes all variables in the decision problem in reverse order of the partial ordering imposed by the information constraints. That is, it first sum-marginalizes

$\mathscr{I}_n$, then max-marginalizes D_n, sum-marginalizes $\mathscr{I}_{k-1}$, etc. This type of elimination order is called a strong elimination order [12]. The general scheme of SPI algorithm as presented in this paper is shown in Definition 1.

Definition 1 (SPI Algorithm)
Let ID be an influence diagram. Let Φ and Ψ be the set of all probability and utility potentials respectively. Let the temporal order of the variables be described as $\mathscr{I}_0 \prec D_1 \prec \mathscr{I}_1 \prec \cdots \prec D_n \prec \mathscr{I}_n$. Then, the procedure for evaluating the ID using SPI algorithm is:

1. *for $(k := n; k > 0; k := k-1)$*
   (a) *Remove chance variables in $\mathscr{I}_k$ (Definition 2)*
   (b) *Remove decision D_k (Definition 4)*
2. *Remove chance variables in $\mathscr{I}_0$ (Definition 2)*

VE considers the evaluation as a problem of finding the optimal elimination ordering whereas SPI considers it as a combinatorial factorization problem. That is, VE chooses at each step the best variable to remove while SPI chooses the best pair of potentials to combine and eliminate the variables when possible. In this sense SPI is finer grained than VE.

3.2 Removal of Chance Variables

In order to remove a subset of chance variables $\mathscr{I}_k$, our version of SPI considers probability and utility potentials separately: first, SPI tries to find the best order for combining all the relevant probability potentials in Φ^k (potentials containing any of the variables in $\mathscr{I}_k$). For that purpose, all possible pairwise combinations between probability potentials are stored in the set B. At each iteration, a pair of probability potentials is selected to be combined. The procedure stops when all variables has been removed. A variable can be removed in the moment it only appears in a single probability potential. This procedure is shown in Definition 2.

Definition 2 (Removal of a Subset of Chance Variables). *Let $\mathscr{I}_k$ be the set of variables to remove, let Φ and Ψ be the set of all current probability and utility potentials respectively in an ID. Then, the procedure for removing $\mathscr{I}_k$ is:*

1. *Set the relevant potentials:*

$$\Phi^k := \{\phi \in \Phi | \mathscr{I}_k \cap dom(\phi) \neq \emptyset\} \quad \Psi^k := \{\psi \in \Psi | \mathscr{I}_k \cap dom(\psi) \neq \emptyset\}$$

2. *Update $\Phi := \Phi \backslash \Phi^k$ and $\Psi := \Psi \backslash \Psi^k$*
3. *Initialize the combination candidate set $B := \emptyset$.*
4. *Repeat:*
   (a) *if $|\Phi^k| > 1$, then*
      i. *Add all pairwise combinations of elements of Φ^k to B which are not already in B.*
      ii. *Select a pair $p := \{\phi_i, \phi_j\}$ of B according to some criteria and combine both potentials: Set $\phi_{ij} := \phi_i \otimes \phi_j$*

iii. Determine the set $\mathbf{W}$ *of variables that can be sum-marginalized:*

$$\mathbf{W} := \{X \in dom(\phi_{ij}) \cap \mathscr{I}_k | \forall \phi \in \Phi^k \backslash p : X \notin dom(\phi)\}$$

iv. Update B by deleting all pairs p where $\phi_i \in p$ *or* $\phi_j \in p$.
v. Delete ϕ_i *and* ϕ_j *from* Φ^k.
else
i. Let ϕ_{ij} *be the single potential in* Φ^k.
ii. Determine the set $\mathbf{W}$ *of variables that can be sum-marginalized:*

$$\mathbf{W} := \{X \in dom(\phi_{ij}) \cap \mathscr{I}_k\}$$

iii. Delete ϕ_{ij} *from* Φ^k.
(b) Select the utility potentials relevant for removing $\mathbf{W}$*:*

$$\Psi^{\mathbf{W}} := \{\psi \in \Psi^k | \mathbf{W} \cap dom(\psi) \neq \emptyset\}$$

(c) Sum-marginalize variables in $\mathbf{W}$ *from* ϕ_{ij} *and* $\Psi^{\mathbf{W}}$*. A probability potential* $\phi_{ij}^{\downarrow \mathbf{W}}$ *and a set of utility potentials* $\Psi^{\downarrow \mathbf{W}}$ *are obtained as a result (Definition 3).*
(d) Update the set of variables to remove: $\mathscr{I}_k := \mathscr{I}_k \backslash \mathbf{W}$
(e) Update the set of relevant potentials:

$$\Phi^k := \Phi^k \cup \{\phi_{ij}^{\downarrow \mathbf{W}}\} \qquad \Psi^k := (\Psi^k \setminus \Psi^{\mathbf{W}}) \cup \Psi^{\downarrow \mathbf{W}}$$

Until $\mathscr{I}_k = \emptyset$
5. *Update* $\Phi := \Phi \cup \Phi^k$ *and* $\Psi := \Psi \cup \Psi^k$

In Definition 2 only probability potentials are combined while utility potentials are not. Let us suppose that we aim to remove a variable X from a set of probability potentials $\{\phi_1, \ldots, \phi_k, \phi_{ij}\}$ and from a set of utility potentials $\{\psi_1, \ldots, \psi_l, \psi_m, \ldots + \psi_n\}$. Let ϕ_{ij} and $\{\psi_m, \ldots, \psi_n\}$ be the potentials containing X. Then, the removal of X can be made using Eq.5.

$$\sum_X \phi_1 \cdots \phi_k \phi_{ij} (\psi_1 + \cdots + \psi_l + \psi_m + \cdots + \psi_n) =$$
$$= \phi_1 \cdots \phi_k \left(\sum_X \phi_{ij} \right) \left(\psi_1 + \cdots + \psi_l + \frac{\sum_X (\phi_{ij} (\psi_m + \cdots + \psi_n))}{\sum_X \phi_{ij}} \right) \quad (5)$$

The utility potentials must be combined with ϕ_{ij} which is the resulting potential of combining all potentials containing X. For that reason, the utilities can only be combined when a variable can be removed. That is the moment when ϕ_{ij} has been calculated. The procedure for sum-marginalizing a set of variables (Definition 3) involves finding good order for summing the utility potentials. The procedure for that is quite similar to the procedure for combining probabilities, however the combination candidate set B can contain singletons as well. The reason for that is that in some cases it could be better to apply the distributive law [5,11].

Definition 3 (Sum-Marginalization). *Let ϕ be a probability potential and $\Psi^{\mathbf{W}}$ a set of utility potentials relevant for removing the chance variables in $\mathbf{W}$. Then, the procedure for sum-marginalizating $\mathbf{W}$ from ϕ and $\Psi^{\mathbf{W}}$ is:*

1. *Initialize the combination candidate set $B := \emptyset$.*
2. *Repeat:*
   (a) *Add all pairwise combinations of elements of $\Psi^{\mathbf{W}}$ to B which are not already in B.*
   (b) *Add to B all potentials in $\Psi^{\mathbf{W}}$ that contains any variable of $\mathbf{W}$ which is not present in any other potential of $\Psi^{\mathbf{W}}$, that is a variable that can be removed.*
   (c) *Select a pair $q := \{\psi_i, \psi_j\}$ or a singleton $q := \{\psi_i\}$ from B according to some criteria.*
   (d) *If q is a pair, then $\psi_{ij} := \psi_i + \psi_j$. Otherwise, $\psi_{ij} := \psi_i$*
   (e) *Determine the set $\mathbf{V}$ of variables that can be sum-marginalized:*

$$\mathbf{V} := \{X \in dom(\psi_{ij}) \cap \mathbf{W} | \forall \psi \in \Psi^{\mathbf{W}} \backslash q : X \notin dom(\psi)\}$$

   (f) *Sum-marginalize $\mathbf{V}$, giving as a result:*

$$\phi^{\downarrow \mathbf{V}} := \sum_{\mathbf{V}} \phi \qquad \psi^{\downarrow \mathbf{V}} := \sum_{\mathbf{V}} (\phi \otimes \psi_{ij}) / \phi^{\downarrow \mathbf{V}}$$

   (g) *If q is a pair, remove ψ_i and ψ_j from $\Psi^{\mathbf{W}}$ and any element in B containing them. Otherwise, only remove ψ_i from $\Psi^{\mathbf{W}}$ and any element in B containing it.*
   (h) *Update $\phi := \phi^{\downarrow \mathbf{V}}$ and $\Psi^{\mathbf{W}} := \Psi^{\mathbf{W}} \cup \{\psi^{\downarrow \mathbf{V}}\}$*
   (i) *Update the set of variables to remove: $\mathbf{W} := \mathbf{W} \backslash \mathbf{V}$*

   Until $\mathbf{W} = \emptyset$
3. *Return ϕ and $\Psi^{\mathbf{W}}$*

3.3 Removal of Decision Variables

Once all variables in $\mathscr{I}_k$ are removed using algorithm in Definition 2, a similar procedure must be performed to remove a decision variable D_k (see Definition 4). However, this removal does not imply the combination of any probability potential since any decision is d-separated from its predecessors [11]. Thus, any probability potential $\phi(D_k, \mathbf{X})$ must be directly transform into $\phi(\mathbf{X})$ if D_k is a decision and $\mathbf{X}$ is a set of chance variables that belong to $\mathscr{I}_i$ with $i < k$. This property is used at step 2 of Definition 4.

Definition 4 (Removal of a Decision Variable). *Let D_k be the decision variable to remove, let Φ and Ψ be the set of all current probability and utility potentials respectively. Then, the procedure for removing D_k is:*

1. *Set the relevant potentials:*

$$\Phi^k := \{\phi \in \Phi | D_k \in dom(\phi)\} \qquad \Psi^k := \{\psi \in \Psi | D_k \in dom(\psi)\}$$

2. *Foreach $\phi \in \Phi^k$, remove D_k by restricting ϕ to any of the values of D_k. The set of potentials $\Phi^{\downarrow D_k}$ is given as a result.*

3. *Max-marginalize variable D_k from Ψ^k and record the policy for D_k. A new potential $\psi^{\downarrow D_k}$ is obtained as a result (Definition 5).*
4. *Update the set of potentials in the ID:*

$$\Phi := (\Phi \backslash \Phi^k) \cup \Phi^{\downarrow D_k} \qquad \Psi := (\Psi \backslash \Psi^k) \cup \{\psi^{\downarrow D_k}\}$$

Definition 5 (Max-Marginalization). *Let Ψ be a set of utility potentials and D a decision variable. Then, the procedure for max-marginalizating D from Ψ is:*

1. *Initialize the combination candidate set $B := \emptyset$.*
2. *While $|\Psi| > 1$:*
   (a) *Add all pairwise combinations of elements of Ψ to B which are not already in B.*
   (b) *Select a pair $q := \{\psi_i, \psi_j\}$ according to some criteria and sum both potentials giving as a result ψ_{ij}.*
   (c) *Update B by deleting all pairs p where $\psi_i \in p$ or $\psi_j \in p$.*
   (d) *Update $\Psi := \Psi \backslash \{\psi_i, \psi_j\} \cup \{\psi_{ij}\}$*
3. *Let ψ^D be the single potential in Ψ.*
4. *Max-marginalize D, giving as a result: $\psi^{\downarrow D} := \max_D \psi^D$*
5. *Return $\psi^{\downarrow D}$*

3.4 Heuristics

During the removal of the chance variables, at each iteration a pair of probability potentials is selected to be combined (Definition 2, step 4.a.ii). For that, some heuristics used with VE can be adapted for selecting a pair. Let $p := \{\phi_i; \phi_j\}$ be a candidate pair to be combined, let $\phi_{ij} = \phi_i \otimes \phi_j$ be the resulting potential of the combination and let $\mathbf{W}$ be the set of variables that can be removed. Then, the heuristics *minimum size* [13], *minimum weight* [12] and *Cano and Moral* [14] are defined as:

$$min\_size(p) = |dom(\phi_i) \cup dom(\phi_j)| = |dom(\phi_{ij})| \tag{6}$$

$$min\_weight(p) = \prod_{X \in dom(\phi_{ij})} |\Omega_X| \tag{7}$$

$$Cano\_Moral(p) = \frac{\prod_{X \in dom(\phi_{ij})} |\Omega_X|}{\prod_{Y \in \mathbf{W}} |\Omega_Y|} \tag{8}$$

Li and D'Ambrosio [9] also proposed an heuristic that selects a pair that minimises the score s_1 and maximises the score s_2:

$$s_1(p) = |dom(\phi_{ij})| - |\mathbf{W}| \tag{9}$$

$$s_2(p) = |dom(\phi_i)| + |dom(\phi_j)| \tag{10}$$

Any of the heuristics previously mentioned can also be used for selecting a pair of utility potentials at steps 2.*c* and 2.*b* of Definitions 3 and 5 respectively. These heuristics will be considered in the experimental analysis.

3.5 Example

Let us consider the ID in Fig. 1 to illustrate the behaviour of the SPI algorithm as described in this paper. In order to simplify the notation, $\phi(X_1,\ldots,X_n)$ will be denoted $\phi_{X_1,\ldots,X_n}$. First, SPI proceeds to remove variables in the chance set $\mathscr{I}_1 = \{B,C,E,F,G\}$ using the algorithm in Definition 2. The initial combination candidate set is:

$$\{\phi_C;\phi_E\},\{\phi_C;\phi_F\},\{\phi_C;\phi_G\},\{\phi_C;\phi_{BCEFG}\},\{\phi_C;\phi_{AB}\},\{\phi_E;\phi_F\},\{\phi_E;\phi_G\},\{\phi_E;\phi_{BCEFG}\},$$
$$\{\phi_E;\phi_{AB}\},\{\phi_F;\phi_G\},\{\phi_F;\phi_{BCEFG}\},\{\phi_F;\phi_{AB}\},\{\phi_G;\phi_{BCEFG}\},\{\phi_G;\phi_{AB}\},\{\phi_{BCEFG};\phi_{AB}\}$$

If the *minimum size* heuristic is used for selecting the next pair of potentials, there are 6 pairs minimizing this score. Let us suppose that the pair $\{\phi_C;\phi_E\}$ is chosen, then the resulting potential is ϕ_{CE}. There is not any variable that can be removed, since C and E appear in other potentials (e.g., ϕ_{BCEFG}). Then, the set B is updated by removing pairs containing ϕ_C or ϕ_E and by adding new pairwise combinations with ϕ_{CE}:

$$\{\phi_{CE};\phi_F\},\{\phi_{CE};\phi_G\},\{\phi_{CE};\phi_{BCEFG}\},\{\phi_{CE};\phi_{AB}\},\{\phi_F;\phi_G\},$$
$$\{\phi_F;\phi_{BCEFG}\},\{\phi_F;\phi_{AB}\},\{\phi_G;\phi_{BCEFG}\},\{\phi_G;\phi_{AB}\},\{\phi_{BCEFG};\phi_{AB}\}$$

The process will continue by choosing a pair to combine until all variables have been removed. The whole process is shown in Fig. 2 in a factor graph [7]. Nodes without any parent correspond to initial potentials while child nodes correspond to the resulting potentials of a combination. The numbers above each potentials indicate the combination ordering and arcs labels indicate the variables that are sum-marginalized.

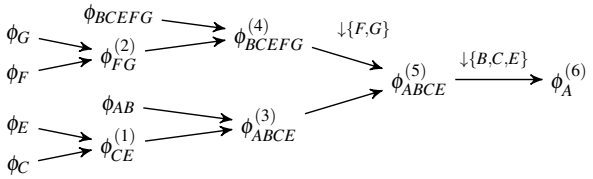

Fig. 2. Combination order of the probability potentials obtained using SPI for removing the chance set $\mathscr{I}_1 = \{B,C,E,F,G\}$ during the evaluation of the ID shown in Fig.1

In the 4^{th} iteration, after generating the potential ϕ_{BCEFG}, variables F and G can be removed. Then, the algorithm in Definition 3 is executed in order to combine utility potentials and max-marginalize these variables: the combination candidate set of utility potentials is $B := \{\{\psi_{D_1FG}\}\}$ and the resulting potentials are ϕ_{BCE} and ψ_{D_1BCE}. Similarly, in the 5^{th} iteration, variables B,C and E can be removed. Now, the combination candidate set contains a pair and a singleton, that is $B := \{\{\psi_{D_1CE};\psi_{D_1BCE}\},\{\psi_{D_1BCE}\}\}$. The element selected from B is the pair $\{\psi_{D_1CE};\psi_{D_1BCE}\}$. The variables B,C and E can be removed after adding both utility potentials in the pair, thus it is not needed to perform any additional iteration. The resulting potentials are ϕ_A and ψ_{D_1A} which are also, in this case, the resulting potentials of algorithm in Definition 2. SPI will now proceed to remove decision D_1 using Definition 4 and chance variable A using Definition 2.

4 Experimental Work

For testing the SPI algorithm, a set of 10 IDs found in the literature are used: NHL and IctNeo are two real world IDs used for medical purposes [15,16]; the oil wildcatter's problem with one and two utilities [17,18]; the Chest Clinic ID [19] obtained from the Asia BN; an ID representing the decision problem in the poker game [11]; an ID used at agriculture for treating the mildew [11]; finally, three synthetic IDs are used: the motivation example shown in Fig.1 with binary and not binary variables and the ID used by Jensen et al. in [6]. Each ID is evaluated using the SPI and the VE algorithms with the heuristics shown in Section 3.4. The *Li and D'Ambrosio* heuristic is not used with the VE algorithm because it is a specific heuristic for the SPI algorithm. An efficiency improvement used in both algorithms consists on discarding any unity probability potential generated.

Table 1 shows the total number of operations needed for each evaluation, that is the number of multiplications, divisions, additions and maximum comparisons. The ratio of the number of operations using SPI to the number of operations using VE is also shown. It can be observed that SPI requires a lower number of operations than VE in 7 out of 10 IDs when using the *minimum size* and the *Cano and Moral* heuristic. By contrast, if the *minimum weight* heuristic is used instead, SPI offers better results in 6 out of 10 IDs. Comparing *Li and D'Ambrosio* heuristics with the rest, it can be seen that this criteria only offers better results in 2 out 10 IDs.

Table 1. Number of operations needed for evaluating each ID using SPI and VE algorithms and different heuristics

	min_size			*min_weight*			*cano_moral*			*li_dambrosio*
ID	SPI	VE	ratio	SPI	VE	ratio	SPI	VE	ratio	SPI
NHL	$\mathbf{2.74 \cdot 10^6}$	$5.04 \cdot 10^6$	0.54	$6.96 \cdot 10^6$	$\mathbf{4.95 \cdot 10^6}$	1.41	$\mathbf{6.96 \cdot 10^6}$	$8.80 \cdot 10^6$	0.79	$2.05 \cdot 10^7$
IctNeo	$2.42 \cdot 10^6$	$\mathbf{4.34 \cdot 10^5}$	5.57	$2.36 \cdot 10^6$	$\mathbf{3.90 \cdot 10^5}$	6.04	$2.36 \cdot 10^6$	$\mathbf{4.36 \cdot 10^5}$	5.4	$1.07 \cdot 10^6$
Oil Wildcatter	**125**	150	0.83	**125**	150	0.83	**125**	157	0.8	125
Oil Split Costs	**137**	162	0.85	**137**	162	0.85	**137**	169	0.81	137
Chest Clinic (Asia)	**598**	657	0.91	**598**	625	0.96	**598**	645	0.93	682
Poker	4499	**1775**	2.53	4499	**1831**	2.46	4499	**1582**	2.84	$6.12 \cdot 10^4$
Mildew	$\mathbf{2.63 \cdot 10^4}$	$3.31 \cdot 10^4$	0.8	$\mathbf{2.36 \cdot 10^4}$	$3.31 \cdot 10^4$	0.72	$\mathbf{2.36 \cdot 10^4}$	$2.60 \cdot 10^4$	0.91	$5.02 \cdot 10^4$
motivation binary	**324**	511	0.63	**324**	511	0.63	**324**	513	0.63	356
motivation not binary	**1753**	6559	0.27	**1753**	2273	0.77	**1753**	1919	0.91	4597
Jensen et al.	922	**533**	1.73	922	**533**	1.73	922	**545**	1.69	746

5 Conclusions and Future Work

In this work we have described how the SPI algorithm can be used for evaluating IDs, which considers the evaluation as combinatorial factorization problem. That is, SPI tries to find an optimal order for the operations of marginalization and combination. Thus, SPI is finer grained than VE. Moreover, we also propose adapting some of the heuristics that VE uses for selecting the next pair of potentials to combine.

The experimental work shows that, in many cases, the SPI algorithm can reduce the number of operations needed to evaluate an ID compared to VE. However, SPI does not strictly dominates VE. For that reason, a line of future research could be determining which features of an ID make that SPI offers better results. The efficiency of SPI for evaluating some IDs also depends on the heuristic used, thus another line of future research could be looking for alternative heuristics. One method that improves the efficiency of the evaluation is Lazy Evaluation (LE) [6], [20], which is based on message passing in a *strong junction tree*. The SPI algorithm was already proposed as method for computing the messages in the LE of Bayesian networks [21]. Thus similar ideas could be applied for computing the messages in the LE of IDs.

Acknowledgments. This research was supported by the Spanish Ministry of Economy and Competitiveness under project TIN2010-20900-C04-01, the European Regional Development Fund (FEDER), the FPI scholarship programme (BES-2011-050604) and the short stay in foreign institutions scholarship EEBB-I-13-07111. The authors have also been partially supported by "Junta de Andalucía" under projects TIC-06016 and P08-TIC-03717.

References

1. Howard, R.A., Matheson, J.E.: Influence diagram retrospective. Decision Analysis 2(3), 144–147 (2005)
2. Kjræulff, U.B., Madsen, A.L.: Bayesian Networks and Influence Diagrams: A Guide to Construction and Analysis, vol. 22. Springer (2012)
3. Shachter, R.D., Bhattacharjya, D.: Solving influence diagrams: Exact algorithms. Wiley Encyclopedia of Operations Research and Management Science (2010)
4. Shachter, R.: Evaluating influence diagrams. Operations Research, 871–882 (1986)
5. Madsen, A., Jensen, F.: Lazy evaluation of symmetric Bayesian decision problems. In: Proceedings of the 15th Conference on Uncertainty in AI, pp. 382–390. Morgan Kaufmann Publishers Inc. (1999)
6. Jensen, F., Jensen, F.V., Dittmer, S.L.: From Influence Diagrams to junction trees. In: Proceedings of the 10th International Conference on Uncertainty in AI, pp. 367–373. Morgan Kaufmann Publishers Inc. (1994)
7. Bloemeke, M., Valtorta, M.: A hybrid algorithm to compute marginal and joint beliefs in Bayesian networks and its complexity. In: Proceedings of the 14th conference on Uncertainty in AI, pp. 16–23. Morgan Kaufmann Publishers Inc. (1998)
8. Shachter, R.D., D'Ambrosio, B., Del Favero, B.: Symbolic probabilistic inference in belief networks. In: AAAI, vol. 90, pp. 126–131 (1990)
9. Li, Z., d'Ambrosio, B.: Efficient inference in Bayes networks as a combinatorial optimization problem. IJAR 11(1), 55–81 (1994)
10. D'Ambrosio, B., Burgess, S.: Some experiments with real-time decision algorithms. In: Proceedings of the 12th International Conference on Uncertainty in AI, pp. 194–202. Morgan Kaufmann Publishers Inc. (1996)
11. Jensen, F., Nielsen, T.: Bayesian networks and decision graphs. Springer (2007)
12. Kjærulff, U.: Triangulation of graphs – algorithms giving small total state space. Research Report R-90-09, Department of Mathematics and Computer Science, Aalborg University, Denmark (1990)

13. Rose, D.: A graph-theoretic study of the numerical solution of sparse positive definite systems of linear equations. Graph Theory and Computing 183, 217 (1972)
14. Cano, A., Moral, S.: Heuristic algorithms for the triangulation of graphs. In: Bouchon-Meunier, B., Yager, R.R., Zadeh, L.A. (eds.) IPMU 1994. LNCS, vol. 945, pp. 98–107. Springer, Heidelberg (1995)
15. Lucas, P., Taal, B.: Computer-based decision support in the management of primary gastric non-hodgkin lymphoma. UU-CS (1998-33) (1998)
16. 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), 307–315 (1998)
17. Raiffa, H.: Decision analysis: Introductory lectures on choices under uncertainty (1968)
18. Dawid, P., Lauritzen, S.L., Spiegelhalter, D.J.: Probabilistic networks and expert systems: Exact computational methods for Bayesian networks. Springer (2007)
19. Goutis, C.: A graphical method for solving a decision analysis problem. IEEE Transactions on Systems, Man and Cybernetics 25(8), 1181–1193 (1995)
20. Madsen, A., Jensen, F.: Lazy propagation: a junction tree inference algorithm based on lazy evaluation. Artificial Intelligence 113(1-2), 203–245 (2004)
21. Madsen, A.: Variations over the message computation algorithm of lazy propagation. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 36(3), 636–648 (2005)