A note about redundancy in Influence Diagrams

Enrico Fagiuoli† and Marco Zaffalon‡

† *Università degli Studi di Milano, Dipartimento di Matematica*
via Cicognara 7, 20129 Milano, Italia
‡ *IDSIA, Istituto Dalle Molle di Studi sull'Intelligenza Artificiale*
corso Elvezia 36, 6900 Lugano, Svizzera
fagiuoli@milano.ccr.it, zaffalon@idsia.ch

Influence Diagrams (IDs) are formal tools for modelling decision processes and for computing optimal strategies under risk. Like Bayesian networks, influence diagrams exploit the sparsity of the dependency relationships among the random variables in order to reduce computational complexity. In this note, we initially observe that an influence diagram can have some arcs that are not necessary for a complete description of the model. We show that while it may not be easy to detect such arcs, it is important, since a redundant graphical structure can exponentially increase the computational time of a solution procedure. Then we define a graphical criterion that is shown to allow the identification and removal of the redundant parts of an ID. This technical result is significant because it precisely defines what is relevant to know at the time of a decision. Furthermore, it allows a redundant influence diagram to be transformed into another ID, that can be used to compute the optimal policy in an equivalent but more efficient way.

Key words: Decision theory, Influence diagrams, Bayesian networks, Optimal policy

1 Introduction

Influence diagrams are graphical models used in decision theory for modelling decision processes in an uncertain domain. Like Bayesian networks [2,7], an ID is a structure built by qualitative and quantitative information [1,3,5,6,9–11]. The uncertainty is modeled by means of a set of random variables. The variables are represented by nodes of a directed graph whose arcs represent the dependency relationships among them. Also the decisions and the utility function are modeled with nodes of the graph. The graph constitutes the qualitative part. The quantitative part is related to a number of conditional distributions that must be associated to the random nodes. The ability to decompose the joint distribution of the random variables with the product of smaller conditional distributions local to the nodes is the main characteristic of IDs (and Bayesian networks). This allows the description of the model and the complexity of solution procedures to be reduced, in such a way that also large-sized models can be treated.

The decision nodes creates some differences between influence diagrams and Bayesian networks [13]. One of these is the effect of a redundant graphical description of the problem. In IDs a redundant graphical structure can have a great impact on the complexity of a solution procedure, up to inhibiting the achievement of the solution. In the paper we describe the problem in detail and introduce a graphical criterion that allows the redundant part of an ID to be identified and removed. By means of such a procedure, a generic influence diagram can be transformed into another, equivalent ID, without redundancy.

The paper is structured as follows. Section 2 formalizes the ID model, introduces some notations used in the sequel and outlines the similarities between IDs and Bayesian networks. Section 3 introduces the problem by means of some examples, and highlights the relationship between a redundant description and the solution complexity. Section 4 defines the criterion for removing the redundant parts of the graph and provides the formal justification of such a procedure. Section 5 applies the results to an example graph. Section 6 generally discusses the impact of decision nodes on complexity, outlining more precisely where the results of the paper come into use. Finally, a lemma used in section 4 is proved in the Appendix.

2 A formal description of influence diagrams

In this section we present a formal description of influence diagrams and of related concepts. The formalization follows the line introduced by Ndilikilikesha [4].

Let $G = (N, A)$ be a directed acyclic graph (DAG) where N is the set of nodes and $A \subseteq N \times N$ is the set of arcs.

For any node $t \in N$, define $\pi(t) = \{s \in N \,|(s,t) \in A\}$ and $\sigma(t) = \{s \in N \,|(t,s) \in A\}$, respectively the sets of direct predecessors of t and of direct successors of t. A node t is said a source if $\pi(t) = \emptyset$ or a barren if $\sigma(t) = \emptyset$. For any set $W \subseteq N$, let $\pi(W) = \cup_{t \in W} \pi(t)$ and $\underline{\pi}(W) = W \cup \pi(W)$.

The nodes are partitioned into three sets: $N = C \cup D \cup \{v\}$. The nodes in C are called *chance nodes*, the nodes in D *decision nodes* and v is called *value* *node*. The arcs entering a chance or the value node are called conditioning arcs; the arcs entering a decision node are said *informative arcs*.

Any node $t \in N$ is associated to a variable $X_t \in \Omega_t$, where $|\Omega_t| < \infty$; a generic value of X_t is denoted by x_t. If W is a generic non-empty set of nodes, X_W denotes the vector of variables indexed by the elements of W and with values in $\Omega_W = \times_{t \in W} \Omega_t$. If t is a decision node, Ω_t is the set of the decisions associated with t; if t is a chance node, the values in Ω_t are the possible states of the random variable X_t and for any value of $X_{\pi(t)}$, it is defined the conditional probability distribution $P\left[X_t \middle| X_{\pi(t)}\right]$. The case of the value node is slightly different. The node v must express the utility value that is the consequence of a certain state of v parents and therefore should contain a function $u : \Omega_{\pi(v)} \to \Omega_v$. The same concept is expressed in a different way by considering X_v a random variable and realizing the function u by means of the probability

$$P\left[X_v \middle| X_{\pi(v)}\right] = \begin{cases} 1 \text{ if } X_v = u\left(X_{\pi(v)}\right) \\ 0 \text{ otherwise,} \end{cases} \tag{1}$$

in fact, $u\left(X_{\pi(v)}\right) = \sum_{X_v} X_v P\left[X_v \middle| X_{\pi(v)}\right]$. In the rest of the paper we use the probability description of the utility function. Using probability (1) has the advantage of treating the value node as a random variable.

Definition 1 *An **Influence Diagram** is a pair (G, P) such that:*

- *$G = (N, A)$ is a DAG such that $N = C \cup D \cup \{v\}$ and the following conditions are satisfied:*
  - *(i) v is barren;*
  - *(ii) there exists a directed path connecting all the decision nodes and only them (*single decision-maker property*);*
  - *(iii) the direct predecessors of any decision node are direct predecessors of all the subsequent decision nodes (*no-forgetting property*);*
- *P is the family of conditional distributions associated to the nodes in $C \cup \{v\}$.*

The nodes in D model the decisions by means of decision functions.

Definition 2 *Let $G = (G, P)$ be an ID and t a decision node. A function $d_t : \Omega_{\pi(t)} \to \Omega_t$ is called a **decision function** of t.*

Definition 3 *A **strategy** for an influence diagram is the function $s : \Omega_{\pi(D) \setminus D} \to \Omega_D$ resulting by the application of all the decision functions. A **partial strategy** s_K is a strategy related to a subset K of decision nodes.*

Note that, as in the case of the value node above, the decision function of t

can be expressed in an equivalent way by means of the following probability,

$$P\left[X_t \left| X_{\pi(t)}\right.\right] = \begin{cases} 1 \text{ if } X_t = d_t\left(X_{\pi(t)}\right) \\ 0 \text{ otherwise.} \end{cases}$$

When a strategy is fixed, the latter probability makes it possible to associate a set of conditional probabilities to the decision nodes and hence to *formally* treat the decision nodes too as chance nodes. In this case *any* node of the graph is *formally* a chance node. Since the graph is directed and acyclic, the model is *formally* a Bayesian network. For this reason the d-separation criterion [7] can be defined over an influence diagram. Notice that the d-separation does not depend on the fixed strategy and therefore is still a pure graphical criterion. In the present work we denote with

$$H \underset{M}{\perp} L$$

the d-separation of H and L by means of M, for any triple of disjoint sets of nodes (the same notation is also used for single nodes).

The equivalence between influence diagrams and Bayesian networks allows the factorization theorem to be extended to IDs in a straightforward way,

$$P_s\left[X_N\right] = \prod_{t \in C \cup \{v\}} P\left[X_t \left| X_{\pi(t)}\right.\right] \prod_{r \in D} P_s\left[X_r \left| X_{\pi(r)}\right.\right]. \tag{2}$$

In other words, the joint distribution is the product of the conditional distributions of the nodes. The subscript s indicates that a strategy must be fixed in order to define a joint distribution, since the decision nodes are interpreted as chance nodes only in this case. Different strategies lead to different joint distributions and therefore to different expected values for the utility. The expected value is defined as

$$E_s\left[X_v\right] = \sum_{X_v} X_v P_s\left[X_v\right] = \sum_{X_N} X_v \prod_{t \in N} P_s\left[X_t \left| X_{\pi(t)}\right.\right] \tag{3}$$

(where the product of conditional distributions is written in a shorter form, but it is clear that the subscript s is only related to decision nodes probabilities. We use the same notation in the rest of the paper).

Definition 4 *A strategy s^* is said **optimal** if for any other strategy s, $E_s\left[X_v\right] \leq E_{s^*}\left[X_v\right]$. The quantity $E_{s^*}\left[X_v\right]$ is said **optimal expected value**.*

A procedure *solves* an ID if it computes the optimal expected value and the associated optimal strategy.

3 Redundancy in Influence Diagrams

In this section we briefly emphasize the relationship between what is relevant to a decision and the computational complexity of an ID solution procedure. Consider the simple influence diagram in figure 1,

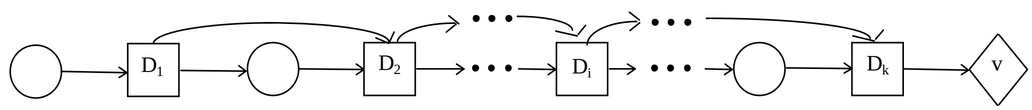

Fig. 1. A simple redundant ID

where the chance nodes are represented by circles, the decision nodes by boxes and v is the value node. The diagram is modified in order to satisfy the no-forgetting property, and the result is the diagram in figure 2.

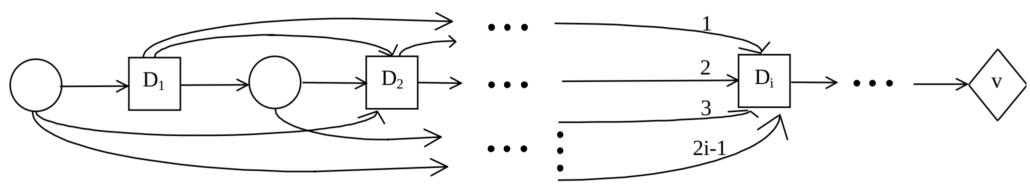

Fig. 2. The original diagram trasformed to satisfy the no-forgetting property

It is clear that the diagram describes the phenomenon with a redundant graph. In fact, since only D_k directly influences the state of the value node, it is not necessary to know the state of any other node in order to compute the optimal expected value. The same computation can be carried out in an equivalent way on the simple ID in figure 3.

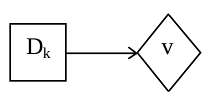

Fig. 3. The reduced influence diagram

In other words, the partial strategy related to the nodes $D_1, \ldots, D_{k-1}$ can be chosen arbitrarily because it does not influence the optimal expected value. Thus a solution procedure can be applied to the influence diagram in figure 3. If the solution algorithm had used the diagram 2, then part of the computation time would uselessly be spent. Let us suppose for simplicity that the decision variables are binary. The decision function of D_k, $d_{D_k} : \Omega_{\pi(D_k)} \longrightarrow \Omega_{D_k}$, depends on the state of $|\pi(D_k)|$ binary variables, i.e. on 2^{2k-1} states. The optimal decision function must be determined considering the optimal decision for any of the above states. For this reason a generic solution algorithm must at least take into account all such states. It follows that the complexity depends on a term $O\left(2^{2k-1}\right)$. The same task takes a negligible time on diagram 3.

Notice that, while the simple structure of diagram 1 allows an easy identification of the redundancy, it may not be that easy for more complex graphs like the graph in figure 4.

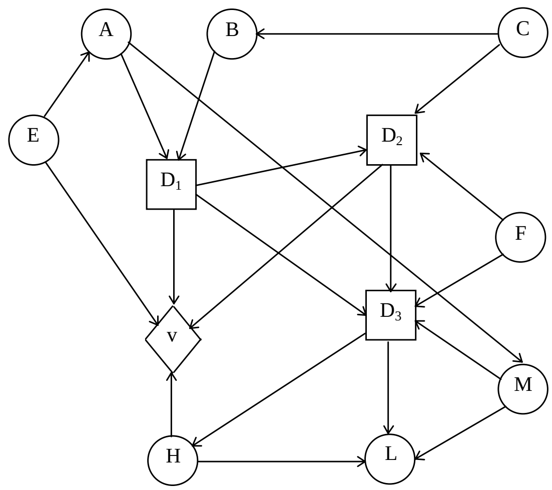

Fig. 4. A redundant influence diagram

In the next section we define a criterion for identifying the redundant arcs and we prove its correctness. Then we apply the transformation to graph 4, showing a substantial reduction of the structure.

4 Graphical identification of redundancies

The graphical criterion is based on some set of nodes defined over an influence diagram.

Definition 5 *Let $ID = (G, P)$ be an influence diagram, where $D = \{D_1, \ldots, D_k\}$ and the order of the decision nodes is $D_1 < D_2 < \ldots < D_k$. $\forall i = k, \ldots, 1$ we define the set*

$$\Gamma_{D_i} = \left\{ t \in \pi(D_i) \middle| t \underset{\pi(D_i)\setminus\{t\}}{\perp} v \quad in \quad G_i \right\}, \tag{4}$$

where $G_k = G$ and G_j, $0 \le j < k$, is the graph G in which all the arcs from Γ_{D_l} to D_l, $\forall l = k, k-1, \ldots,, j+2, j+1$, are removed.

Now we state a result, proved in the appendix, that is used in theorem 7.

Lemma 6 *Let $ID = (G, P)$ be an influence diagram.*

$$\Gamma_{D_i} \underset{\pi(D_i)\setminus\Gamma_{D_i}}{\perp} v \quad in\ G_i, \quad i = 1, \ldots, k.$$

Theorem 7 *Let $ID = (G, P)$ be an influence diagram, where $D = \{D_1, \ldots, D_k\}$ and the order of the decision nodes is $D_1 < D_2 < \ldots < D_k$. Removing all the arcs from Γ_{D_i} to D_i, $\forall i = k, \ldots, 1$, preserves the optimal expected value.*

PROOF. *Let us consider the expected value of the utility function for a fixed strategy s:*

$$E_s[X_v] = \sum_{X_v} X_v P_s[X_v]$$

$$= \sum_{X_v, X_{\underline{\pi}(D_k)}} X_v P_s\left[X_v, X_{\underline{\pi}(D_k)}\right]. \quad (5)$$

Notice that the last summation is taken over all the states of $X_{\underline{\pi}(D_k)}$, but many of the states of $\underline{\pi}(D_k)$ are not compatible*, in the sense that they do not agree with D_k decision function, i.e. there exist states x_{D_k} and $x_{\pi(D_k)}$ such that $P\left[x_{D_k} \middle| x_{\pi(D_k)}\right] = 0$. These states imply $P_s\left[X_v, X_{\underline{\pi}(D_k)}\right] = 0$ in the summation (5), and for this reason can be excluded. In the following derivation it is implicitly assumed to sum only over compatible states.*

Let us take formula (5) into account again.

$$\sum_{X_v, X_{\underline{\pi}(D_k)}} X_v P_s\left[X_v, X_{\underline{\pi}(D_k)}\right]$$

$$= \sum_{X_v, X_{\underline{\pi}(D_k)}} X_v P_s\left[X_v \middle| X_{\underline{\pi}(D_k)}\right] P_s\left[X_{\underline{\pi}(D_k)}\right], \quad (6)$$

for the chain rule. By the application of lemma 6, formula (6) becomes

$$\sum_{X_v, X_{\underline{\pi}(D_k)}} X_v P_s\left[X_v \middle| X_{\underline{\pi}(D_k)\setminus\Gamma_{D_k}}\right] P_s\left[X_{\underline{\pi}(D_k)}\right]$$

$$= \sum_{X_v, X_{\underline{\pi}(D_k)}} X_v P_s\left[X_v \middle| X_{\underline{\pi}(D_k)\setminus\Gamma_{D_k}}\right] P\left[X_{D_k} \middle| X_{\pi(D_k)}\right] P_{s_{D\setminus\{D_k\}}}\left[X_{\pi(D_k)}\right] \quad (7)$$

$$= \sum_{X_v, X_{\underline{\pi}(D_k)}} X_v P_s\left[X_v \middle| X_{\underline{\pi}(D_k)\setminus\Gamma_{D_k}}\right] s_{D_k} P_{s_{D\setminus\{D_k\}}}\left[X_{\pi(D_k)}\right], \quad (8)$$

emphasizing that the probability $P\left[X_{D_k} \middle| X_{\pi(D_k)}\right]$ is the decision function of D_k.

Reordering the terms and denoting $f\left(X_v, X_{\underline{\pi}(D_k)\setminus\Gamma_{D_k}}\right) = X_v P_s\left[X_v \middle| X_{\underline{\pi}(D_k)\setminus\Gamma_{D_k}}\right]$, *formula (8) becomes*

$$=\sum_{X_{\pi(D_k)}} P_{s_{D\setminus\{D_k\}}}\left[X_{\pi(D_k)}\right]\left(\sum_{X_v, X_{D_k}} f\left(X_v, X_{\underline{\pi}(D_k)\setminus\Gamma_{D_k}}\right) s_{D_k}\right). \tag{9}$$

Let us now consider the maximum of the expected value,

$$E_{s^*}\left[X_v\right] = \max_s E_s\left[X_v\right]$$

$$= \max_s \sum_{X_{\pi(D_k)}} P_{s_{D\setminus\{D_k\}}}\left[X_{\pi(D_k)}\right]\left(\sum_{X_v, X_{D_k}} f\left(X_v, X_{\underline{\pi}(D_k)\setminus\Gamma_{D_k}}\right) s_{D_k}\right),$$

by (9),

$$= \max_{s_{D\setminus\{D_k\}}} \sum_{X_{\pi(D_k)}} P_{s_{D\setminus\{D_k\}}}\left[X_{\pi(D_k)}\right]\left(\max_{s_{D_k}} \sum_{X_v, X_{D_k}} f\left(X_v, X_{\underline{\pi}(D_k)\setminus\Gamma_{D_k}}\right) s_{D_k}\right),$$

Let us consider the inner maximization. The partial strategy $s_{D\setminus\{D_k\}}$ *can be considered fixed. In such a case,* $\forall X_{\pi(D_k)} \in \Omega_{\pi(D_k)}$, f *describes the set of values* $F_{X_{\pi(D_k)}} = \left\{f\left(X_v, X_{\underline{\pi}(D_k)\setminus\Gamma_{D_k}}\right) | X_v \in \Omega_v, X_{D_k} \in \Omega_{D_k}\right\}$. *The maximization of the inner sum is equivalent to choosing,* $\forall X_{\pi(D_k)} \in \Omega_{\pi(D_k)}$, *the maximum of* $F_{X_{\pi(D_k)}}$ *by fixing the associated state of* X_{D_k} *(and, consequently, the state of* X_v*). Let us consider any two states* $x'_{\pi(D_k)}, x''_{\pi(D_k)} \in \Omega_{\pi(D_k)}$ *that differ only for the state of* Γ_{D_k}, *i.e.* $x'_{\pi(D_k)\setminus\Gamma_{D_k}} = x''_{\pi(D_k)\setminus\Gamma_{D_k}}$ *and* $x'_{\Gamma_{D_k}} \neq x''_{\Gamma_{D_k}}$. *Since* f *is not function of* $X_{\Gamma_{D_k}}$, *the sets* $F_{x'_{\pi(D_k)}}$ *and* $F_{x''_{\pi(D_k)}}$ *must be equal, and therefore the chosen maximum is the same. Hence, if an arc from* Γ_{D_k} *to* D_k *existed, it would not be informative, since* D_k *does not need to know the state of* Γ_{D_k} *in order to carry out the optimization.*

For these reasons it is possible to remove all the arcs from Γ_{D_k} *to* D_k, *and then to compute the optimal decision function for* D_k. *The resulting graph is* G_{k-1} *by definition. Now observe that the above proof can be applied in the same way to the new graph (in which the last decision node is* D_{k-1} *with the associated set* $\Gamma_{D_{k-1}}$*) and so on, up to the first decision node.* ∎

The previous theorem allows every arc from Γ_{D_i} to D_i, $i = k, \ldots, 1$, to be removed without altering the exact optimal expected value. This means that the influence diagram can be transformed into an equivalent and simpler one.

5 Example

Consider the diagram in figure 4. We transform the diagram in order to make it satisfy the no-forgetting property, as shown in figure 5.

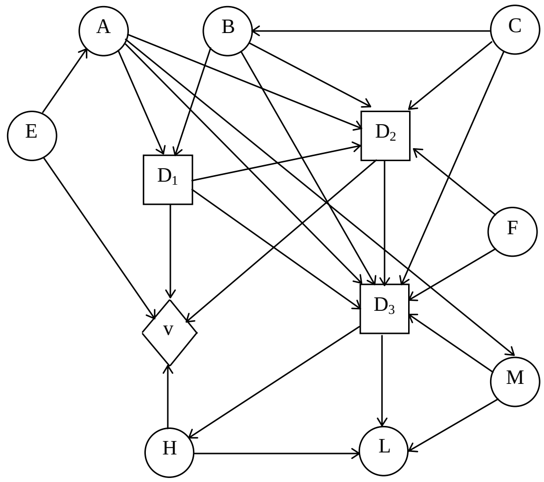

Fig. 5. The ID satisfying the no-forgetting property

Next, we notice that node L is barren and as such can be removed [8].

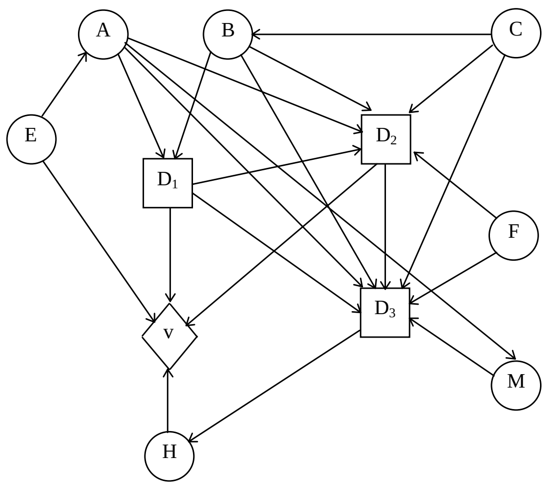

Fig. 6. The removal of barren node L

In order to identify the redundant arcs, we must build the sets Γ_{D_i}, $i = 3, 2, 1$. We begin with Γ_{D_3} on the basis of its definition. We see that $D_1 \notin \Gamma_{D_3}$ because $\neg \left(D_1 \perp_{\{D_2,D_3,A,B,C,F,G\}} v \right)$ since D_1 is a direct predecessor of v. For the same reason $D_2 \notin \Gamma_{D_3}$. $A \notin \Gamma_{D_3}$, because of the path $A \leftarrow E \rightarrow v$. $B \in \Gamma_{D_3}$, because $B \perp_{\{D_1,D_2,D_3,A,C,F,M\}} v$. With similar arguments, it is easy to see that $C, F, M \in \Gamma_{D_3}$, and hence $\Gamma_{D_3} = \{B, C, F, M\}$. Theorem 7 allows to remove any arc leaving $\{B, C, F, M\}$ and entering D_3. The resulting graph is G_2 for definition. Figure 7 shows graph G_2 where node M is removed because it becomes barren after the arcs removal (as a note, observe that Γ_{D_2} should be computed on G_2. Instead, in the sequel it is computed on the graph of figure 7, since it is simpler and equivalent to G_2 under the policy optimization point of view. This also applies to the case of G_1).

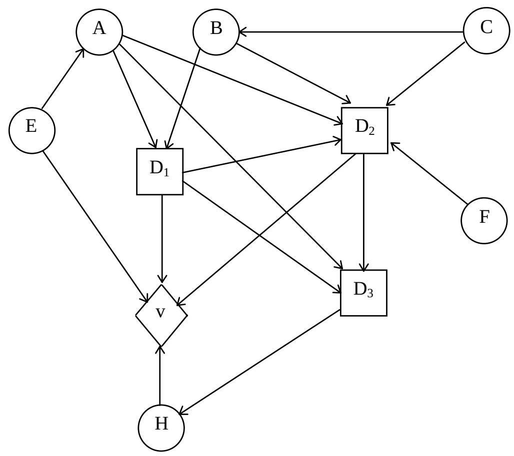

Fig. 7. The removal of non-informative arcs of D_3

The graph in figure 7 is used to build the set Γ_{D_2} that is equal to $\{B, C, F\}$. The graph G_1 (with F removed) is shown in figure 8.

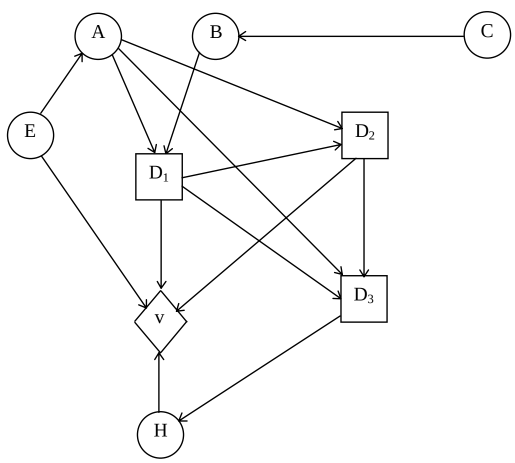

Fig. 8. Graph G_1

By using the graph in figure 8 it is easy to see that $\Gamma_{D_1} = \{B\}$, and therefore the arc from B to D_1 is removed too. The removal of barren nodes produces the final diagram (figure 9).

Notice that in the original graph the maximum number of parents for a decision node is 7, whereas in the latter that number is 3. If the variables are binary, for instance, the computational complexity for determining the optimal decision functions passes from a $O\left(2^7\right)$ term to a $O\left(2^3\right)$ term.

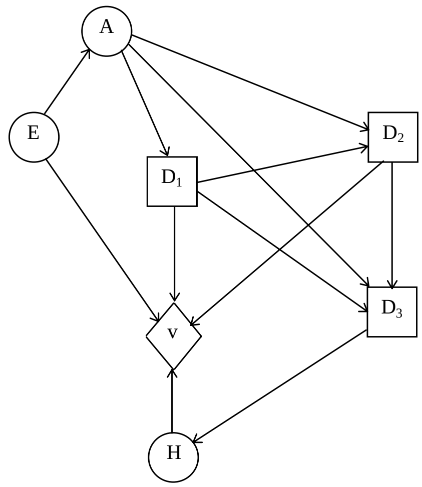

Fig. 9. The final diagram

6 Comments

The presence of decision nodes makes a relevant difference between influence diagrams and Bayesian networks when dealing with complexity issues. In fact, since the no-forgetting property is required, the optimization is related to the whole strategy space, that is the set of *joint-decisions*. Hence the size of the strategy space grows exponentially because of the number of joint states of the decisions (this is outlined by the number of arcs entering the last decision node). When there are *many* states, the problem is intractable.

This sensitivity of the model to the presence of arcs entering decision nodes is the reason why it is important not to introduce redundant arcs that have a heavy computational effect on the resolution. Theorem 7 enables to automatically detect the redundant arcs, significantly reducing the computational time.

Anyway, the problem of the size of the strategy space remains relevant. In literature it is tackled with by assuming the hypothesis of decomposability of the utility function, i.e. when the utility can be built by means of smaller utilities, in a constructive way [12]. A possible development of the present work might be its extension to such a case.

7 Appendix

Proof of Lemma 6. *We must show that* Γ_{D_i} *and* v *are d-separated in* G_i *when* $\underline{\pi}\,(D_i) \setminus \Gamma_{D_i}$ *are known,* $\forall i = 1, \ldots, k$. *The proof is made for a generic*

$i \in \{1, \dots, k\}$. *By contradiction, suppose that*

$$\exists t \in \Gamma_{D_i} : \neg \left(t \underset{\underline{\pi}(D_i) \setminus \Gamma_{D_i}}{\perp} v \right).$$

Hence there exists a path in G_i, w, that d-connects t and v when $\underline{\pi}(D_i) \setminus \Gamma_{D_i}$ are known. By definition of Γ_{D_i}, w must be blocked in G_i when also the remaining nodes of Γ_{D_i} except t are known. Hence, by means of d-separation, some nodes of $\Gamma_{D_i} \setminus \{t\}$ must be on w (at least one), in such a way that when they are istantiated, w becomes blocked. Consider the last node of $\Gamma_{D_i} \setminus \{t\}$ along w, in the direction from t to v. Call it t'.

When $\underline{\pi}(D_i) \setminus \Gamma_{D_i}$ are known, the path w', from t' to v along w, d-connects t' and v. For definition of Γ_{D_i},

$$t' \underset{\underline{\pi}(D_i) \setminus \left\{t'\right\}}{\perp} v$$

holds and therefore, by instantiating also all the remaining nodes of $\Gamma_{D_i} \setminus \left\{t'\right\}$, the path must be blocked. This is impossible, since no node of $\Gamma_{D_i} \setminus \left\{t'\right\}$ is on w'. ▮

References

[1] Howard, R.A., "From influence to relevance to knowledge", in: R.M. Oliver and J.Q. Smith (eds.), *Influence Diagrams, Belief Nets and Decision Analysis*, John Wiley & Sons, New York, 1990, 3–23.

[2] Jensen F.V., *An introduction to Bayesian networks*, Springer Verlag, New York, 1996.

[3] Matzkevich, I., and Abramson, B., "Decision analytic networks in artificial intelligence", *Management Science* 41/1 (1995) 1–22.

[4] Ndilikilikesha, P., "Potential influence diagrams", *International Journal of Approximate Reasoning* 10 (1994) 251–285.

[5] R.M. Oliver and J.Q. Smith (eds.), *Influence diagrams, belief nets and decision analysis*, Proceedings of the Conference Influence Diagrams for Decision Analysis, Inference and Prediction, Engineering Systems Research Center, University of California at Berkeley, John Wiley & Sons, 1988.

[6] Owen, D.L., "The use of influence diagrams in structuring complex decision problems", in: R.A. Howard and J.E. Matheson (eds.), *Readings on the Principles and Applications of Decision Analysis*, 2, Strategic Decision Group, Menlo Park, CA, 1984, 763–771.

[7] Pearl, J., *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann, San Mateo, CA, 1988.

[8] Shachter, R.D., "Evaluating influence diagrams", *Operations Research*, 34/6 (1986) 871–882.

[9] Shachter, R.D., "An ordered examination of influence diagrams", *Networks*, 20 (1990) 535–563.

[10] Shenoy, P.P., "A comparison of graphical techniques for decision analysis", *European Journal of Operation Research*, 78 (1994) 1–21.

[11] Smith, J.Q., "Handling multiple sources of variations using influence diagrams", *European Journal of Operations Research*, 86 (1995) 189–200.

[12] Tatman, J.A., and Shachter, R.D., "Dynamic programming and influence diagrams", *IEEE Transactions on Systems, Man, and Cybernetics*, 20/2 (1990) 365–379.

[13] Zaffalon, M., *Inferenze e Decisioni in Condizioni di Incertezza con Modelli Grafici Orientati*, Ph.D. Thesis, Matematica Computazionale e Ricerca Operativa, Università degli Studi di Milano, Italia, 1997