Solving CLQG Influence Diagrams Using Arc-Reversal Operations in a Strong Junction Tree

Anders L. Madsen

Anders.L.Madsen@hugin.com

HUGIN EXPERT A/S

Motivation

The influence diagram is a compact graphical model representation supporting decision making under uncertainty

- most architectures consider only the discrete case

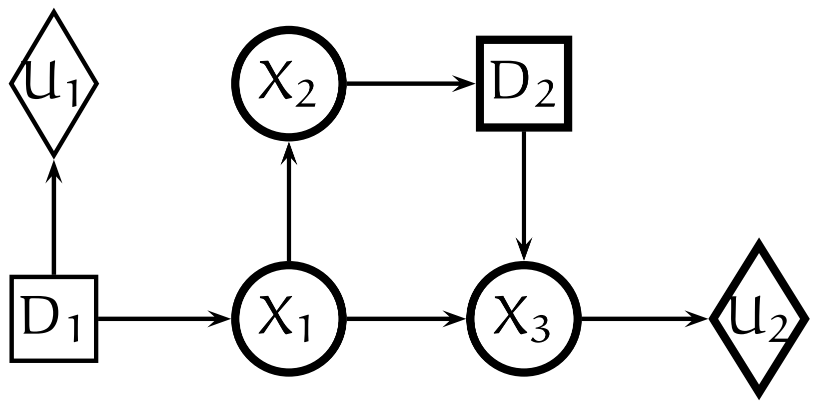
Decision problems often involve reasoning about entities of both discrete and continuous nature

- the problem of solving CLQG influence diagrams has received only limited attention

We present an architecture for representation and efficient, exact solution of CLQG influence diagrams using arc-reversal operations in Lazy Propagation

CLQG Influence Diagrams

A CLQG ID $\mathcal{N} = (\mathcal{X}, \mathcal{G}, \mathcal{P}, \mathcal{F}, \mathcal{U})$ consists of a graph $\mathcal{G}$ over chance, decision, and utility nodes, a set of probability functions, and a set of utility functions



We consider CLQG IDs with a additively decomposing utility function

- for $X \in \Gamma_C$ we have $f(Y|Z = z, I = i) = \mathcal{N}(\alpha(i) + \beta(i)z, \sigma^2(i))$
- discrete chance and decision nodes can only have discrete parents
- $U(x, i) = \sum_{\psi \in \Psi} \psi$ is of the form $x^T Q(i)x + R(i)x + S(i)$

Solving CLQG Influence Diagram

The variables of $\mathcal{N}$ induce an expected UF:

$$EU(\mathcal{X}) = P(\Delta_C \,|\, \Delta_D) \cdot f(\Gamma_C \,|\, \Delta, \Gamma_D) \cdot \sum_{u \in \mathcal{U}} u. \qquad (1)$$

An optimal strategy can be identified by eliminating variables from (1) in the reverse time order.

- The elimination is performed using a sequence of AR operations and barren node eliminations
- The calculations are organized in a strong junction tree

The AR Operation

The edge (X_i, X_j) is reversed by setting

$$p(X_j | Z_1, \ldots, Z_n) = \sum_{X_i} p(X_j | X_i, Z_1, \ldots, Z_n) p(X_i | Z_1, \ldots, Z_n),$$

$$p(X_i | X_j, Z_1, \ldots, Z_n) = \frac{p(X_j | X_i, Z_1, \ldots, Z_n) p(X_i | Z_1, \ldots, Z_n)}{p(X_j | Z_1, \ldots, Z_n)}.$$

The edge (Y_i, Y_j) is reversed by setting

$$Y_i | Z_1, \ldots, Z_n \sim \mathcal{N}((\alpha_{Y_i} + \beta_{Y_j}) + \sum_{i=1}^{n} (\beta_i + \delta_i) Z_i, \sigma^2_{Y_i} + \beta^2_{Y_j} \sigma^2_{Y_j}),$$

$$Y_j | Y_i, Z_1, \ldots, Z_n \sim \mathcal{N}(\mu, \sigma^2), \text{ where}$$

$$\mu = \frac{\alpha_{Y_j} \sigma^2_{Y_i} + \alpha_{Y_i} \beta_{Y_j} \sigma^2_{Y_j} + \beta_{Y_j} \sigma^2_{Y_j} Y_i + \sum_{i=1}^{n} (\delta_i \sigma^2_{Y_i} - \beta_i \beta_{Y_j} \sigma^2_{Y_j}) Z_i}{\sigma^2_{Y_i} + \beta^2_{Y_j} \sigma^2_{Y_j}}, \sigma^2 = \frac{\sigma^2_{Y_j} \sigma^2_{Y_i}}{\sigma^2_{Y_i} + \beta^2_{Y_j} \sigma^2_{Y_j}}$$

The AR operation is basically Bayes' rule and it corresponds to reversing an arc in the graph $\mathcal{G}$

Potentials

A potential is a triple $\pi = (\mathcal{P}, \mathcal{F}, \mathcal{U})$ over probability potentials, densities, and utility functions

- combination $\pi_{W_1} \otimes \pi_{W_2} = (\mathcal{P}_1 \cup \mathcal{P}_2, \mathcal{F}_1 \cup \mathcal{F}_2, \mathcal{U}_1 \cup \mathcal{U}_2)$.
- projection of $\pi_W = (\mathcal{P}_W, \mathcal{F}_W, \mathcal{U}_W)$ to a subset $V \subseteq W$ denotes the potential $\pi_V = \pi_W^{\downarrow V} = (\mathcal{P}_V, \mathcal{F}_V, \mathcal{U}_V)$ obtained by performing a sequence of variable eliminations of $W \setminus V$

Solving a CLQG ID involves combination and projection over potentials

Marginalization

Computing $\pi = \pi^{\downarrow \mathrm{dom}(\pi)-\{X\}} = (\mathcal{P}_V, \mathcal{F}_V, \mathcal{U}_V)$ includes marginalization of

- $X \in \Delta_C$: make barren and set $\pi^*_{\mathrm{dom}(\pi)-\{X\}} = (\mathcal{P}^*, \emptyset, \mathcal{U}^*)$ where

$$\begin{aligned}
\mathcal{P}^* &= \mathcal{P}_X \setminus \{p(X|C(X)) \in \mathcal{P}_X\}, \\
\mathcal{U}^* &= \mathcal{U} \setminus \mathcal{U}_X \cup \{(p(X|C(X)) \cdot \sum_{u \in \mathcal{U}_X} u)^{\downarrow C(X)}\},
\end{aligned}$$

- $X \in \Gamma_C$: make barren and set $\pi^*_{\mathrm{dom}(\pi)-\{X\}} = (\mathcal{P}, \mathcal{F}^*, \mathcal{U}^*)$ where

$$\begin{aligned}
\mathcal{F}^* &= \mathcal{F}_X \setminus \{f(X|C(X)) \in \mathcal{F}_X\}, \\
\mathcal{U}^* &= \mathcal{U} \setminus \mathcal{U}_X \cup \{(f(X|C(X)) \cdot \sum_{u \in \mathcal{U}_X} u)^{\downarrow C(X)}\}.
\end{aligned}$$

- $X \in \Delta_D$: set $\pi^*_{\mathrm{dom}(\pi)-\{X\}} = (\emptyset, \emptyset, \mathcal{U} \setminus \mathcal{U}_X \cup \{\max_D \sum_{u \in \mathcal{U}_X} u\})$
- $X \in \Gamma_D$: set $\pi^*_{\mathrm{dom}(\pi)-\{X\}} = (\emptyset, \emptyset, \mathcal{U} \setminus \mathcal{U}_X \cup \{(\sum_{u \in \mathcal{U}_X} u)^{\downarrow \{Z_1, \ldots, Z_n\}}\})$

The main contribution is the use of uni-variate CLG distributions only

Lazy Propagation (LP)

Lazy Propagation

- an inference architecture based on message passing in a strong junction tree $\mathcal{T}$
  - $\mathcal{T}$ guides the elimination process. It is constructed by moralization and strong triangulation
- Initialization of $\mathcal{T}$: the core of *lazy* evaluation is to maintain potential decompositions until combination becomes mandatory by variable elimination
  - potentials assigned to a clique are not combined
- Message passing in $\mathcal{T}$: messages $\pi_{A \to B}$ are computed by elimination of variables
  - decision variables are eliminated by maximization
  - chance variables are eliminated by summation/integration

Lazy Propagation (LP)

Evaluation of a CLQG ID using Lazy propagation in a strong junction tree

- after initialization each clique C holds a potential $\pi_C = (\mathcal{P}, \mathcal{F}, \mathcal{U})$
- message passing $\pi_{A \to B} = \left(\pi_A \otimes \left(\otimes_{C \in ne(A) \setminus \{B\}} \pi_{C \to A}\right)\right)^{\downarrow B}$
- local computation enables exploitation of barren variables and independence relations between variables

Policy optimization is performed as part of message passing

Main contributions

- AR operations and barren node eliminations as projection operation
- illustrate use of distributive law
- illustrate advantage of decomposition
- performance evaluation

Distributive Law

Different researchers have exploited that the distributive law of algebra (DL) can be exploited in the solution process

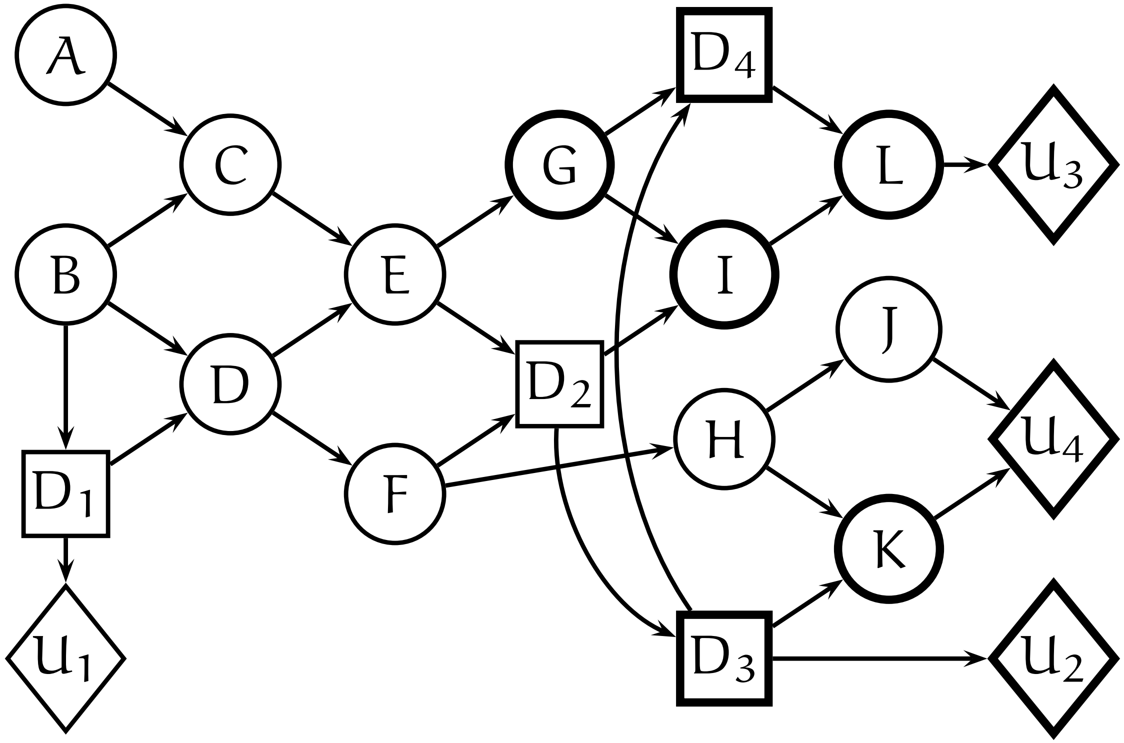
$$U(Y, T, Z) = \sum_{X} P(X) \left(U(X, Y, Z) + U(X, T) \right).$$

Using DL the expression is rewritten:

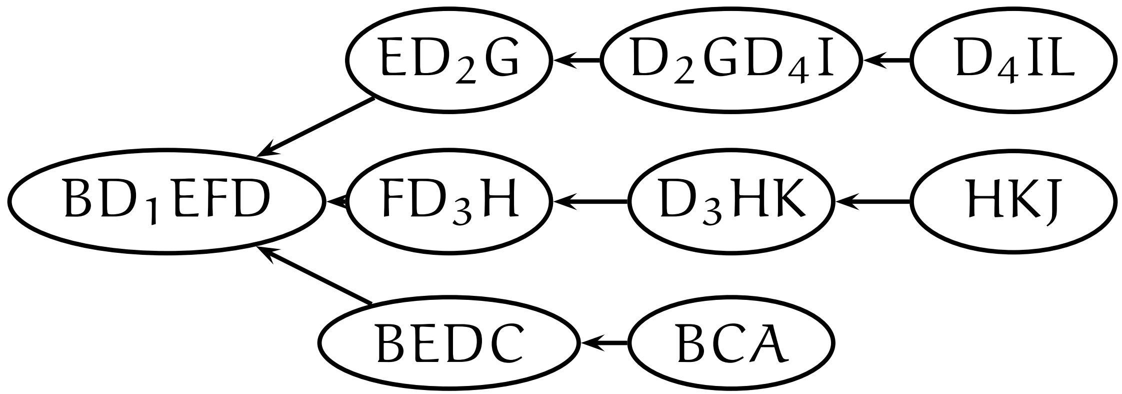
$$U(Y, Z) + U(T) = \sum_{X} P(X) U(X, Y, Z) + \sum_{X} P(X) U(X, T).$$

Decomposition of Potentials

The CLQG ID by Jensen, Jensen & Dittmer



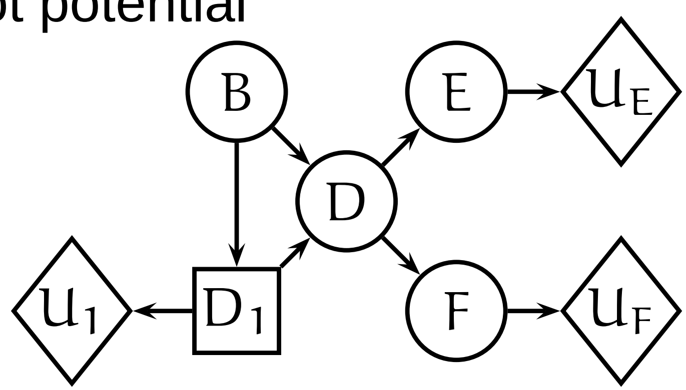
Strong Junction Tree of CLQG ID



At the root we may compute (more efficiently)

$$EU(\hat{\Delta}) = \sum_B P(B) \max_{D_1} \Big(U_1(D_1) + \sum_D P(D \mid B, D_1) \big(\sum_E P(E \mid D) U(E) + \sum_F P(F \mid D) U(F) \big) \Big)$$

Domain graph of root potential



Performance Analysis - Random Networks

	Time		Space	
$\|\|\mathcal{X}\|\|$	LARP	HDE	LARP	HDE
20	4.27	N/A	$1,953,125$	N/A
20	0.93	1.25	$390,625$	$1,953,125$
20	0.03	0.24	$3,125$	$390,625$
25	0.13	N/A	$15,625$	N/A
25	0.64	1.74	$78,125$	$1,953,125$
50	4.67	10.16	$1,048,576$	$8,388,608$
50	24.31	N/A	$4,194,304$	N/A
50	7.22	28.64	$1,048,576$	$16,777,216$

CLQG IDs have discrete variables only and $|\mathcal{X}| \leq 25$ implies $||X|| = 5$ while $|\mathcal{X}| = 50$ implies have $||X|| = 2$

Performance Analysis - JJD network

Different versions of the CLQG ID network of Jensen, Jensen & Dittmer

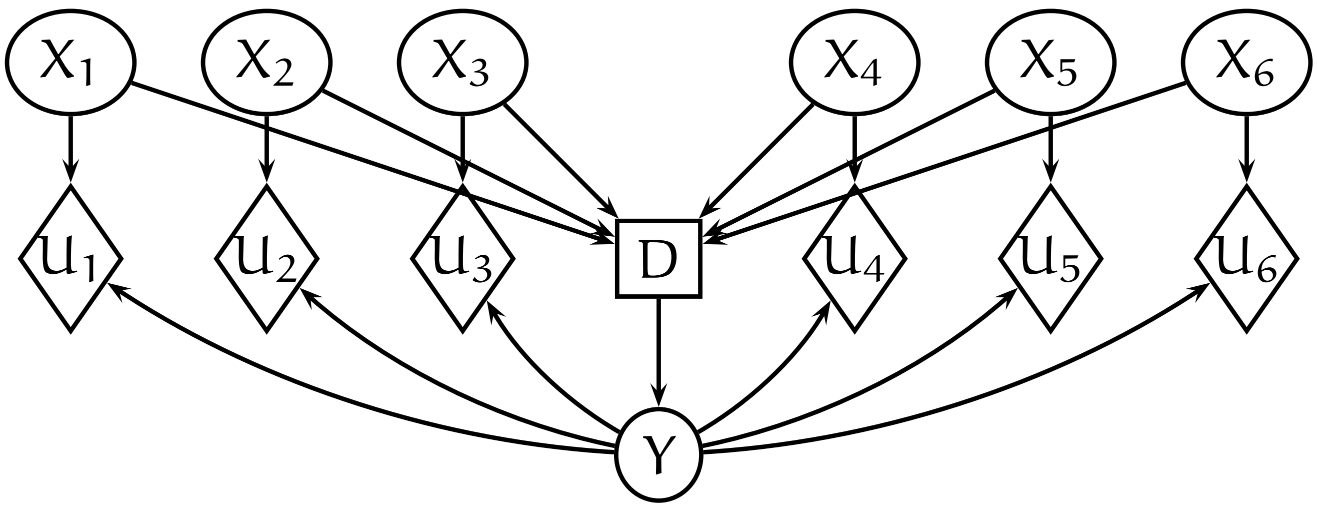
jjd	$\|\mathcal{C}\|$	$\max_{C \in \mathcal{C}} s(C)$	$s(\mathcal{C})$
d	9	$9,765,625$	$10,640,625$
m	9	$9,765,625$	$10,188,826$
c	9	1	1

Performance evaluation

	Time		Space	
jjd	HDE	LARP	HDE	LARP
d	3.87	0.53	$9,765,625$	$390,625$
m	-	0.35	-	$390,625$
c	-	0.03	-	1

Performance Analysis - DL

A naive example with $\|Y\| = 100$, $\|X_i\| = 5$ and $\|D\| = 10$



The (Strong) junction tree is a single clique

Using HDE, LARP and LARP with DL the average time costs in seconds (over ten runs) are 2.91, 16.73, and 0.49.

Conclusion

The main contributions of the paper is an architecture for solving CLQG IDs using arc-reversal in Lazy propagation

- Results of a preliminary performance evaluation are promising
- Future work includes extending the architecture to support the LIMID representation

Related work includes Lauritzen&Jensen on evidence propagation in CLG BNs // Madsen&Jensen and Madsen&Nilsson on solving influence diagrams by *lazy* evaluation // Poland and Kenley&Shachter on linear-quadratic Gaussian influence diagrams // Madsen&Jensen on solving CLQG IDs // Cobb&Shenoy on MTEs