LIMIDS for Decision Problems

Steffen L. Lauritzen and Dennis Nilsson

Department of Mathematical Sciences
Aalborg University

Overview

- LIMIDs versus ordinary IDs
- Single Policy Updating (SPU) for optimizing strategies
- SPU by local computation and message passing
- Reduction of LIMIDs
- Soluble LIMIDs
- Bounds by LIMIDs

Influence Diagrams

An *Influence Diagram* (ID) is a Bayesian network of *chance nodes*, augmented with *decision nodes* and *utility nodes*.

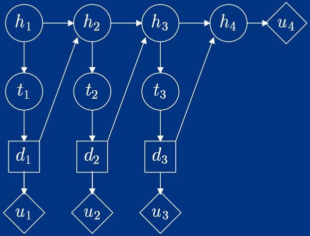
- Chance nodes Γ represented with circles
- Decision nodes Δ represented with squares. Parents observed before decision taken.
- Utility nodes Υ represented with diamonds. Total utility is sum of values at utility nodes.
- Decisions taken in specified order
- Previous decisions and observations remembered.

Influence Diagram

h is health

t is test

d is treat or not



t_1 observed when d_1 is taken. Then t_2 is observed and d_2 is taken *exploiting also knowledge of t_1 and d_1*, etc.

Specifications

Initially $p(h_1 = healthy) = .90$.

health	treatment	$p(h_{i+1} = healthy \mid h_i, d_i)$
healthy	*treated*	.90
healthy	*untreated*	.80
unhealthy	*treated*	.50
unhealthy	*untreated*	.10

$p(t_i = positive \mid healthy) = .10$,
$p(t_i = positive \mid unhealthy) = .80$

Utilities:
$u_i(treated) = -100, u_i(untreated) = 0, i = 1, \ldots 3$

$u_4(healthy) = 1000$, $u_4(unhealthy) = 300$

Policies and Strategies for ID

The *information set* I_d for decision $d \in \Delta$ consists of its parent nodes and the decision history:

$$I_d = \mathrm{pa}(d) \cup \bigcup_{d' < d} \left(\{d'\} \cup \mathrm{pa}(d')\right).$$

A (randomized) *policy* δ_d for $d \in \Delta$ specifies (distribution of) decision at d for each configuration x_{I_d} in its information set, and

A *strategy* $q = \{\delta_d, d \in \Delta\}$ is set of policies for all decisions.

Objectives for IDs

Strategy $q = \{\delta_d, d \in \Delta\}$ induces *joint distribution*

$$f_q(x) = \prod_{r \in \Gamma} p(x_r \,|\, x_{\mathrm{pa}(r)}) \prod_{d \in \Delta} \delta_d(x_d \,|\, x_{I_d}).$$

Aim is to identify strategies q which *optimize expected total utility*

$$\mathrm{EU} = \sum_x f_q(x) U(x) = \sum_x f_q(x) \sum_{u \in \Upsilon} U_u(x_{\mathrm{pa}(u)}).$$

ID represents decision problem more compactly than decision trees. Efficient solution algorithms exploit sparse structure of ID. Problem: *I_d may become large.*

LIMID

A LIMID (LImited Memory Influence Diagram) generalizes an ordinary ID:

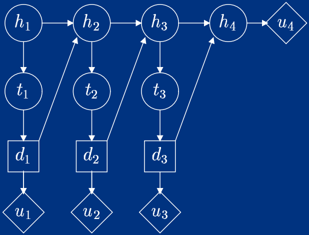
- *Relaxes assumption of complete memory:* parents of decision nodes indicate *exactly* available information
- *Relaxes assumption of specified order of decisions,* demanding only consistency
- *Avoids thereby complexity explosion* of computation of optimal strategies in e.g. POMDPs (Partially Observed Markov Decision Processes)
- May accomodate *multiple decision makers.*

LIMID Example

h is health

t is test

d is treat or not



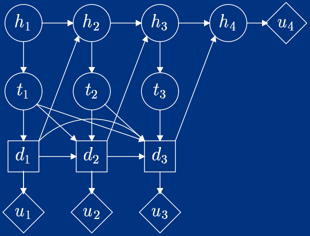
t_1 observed when d_1 is taken. Then t_2 is observed when d_2 is taken, but *t_1 and d_1 are not available then,* etc.

LIMID representation of ID

h is health

t is test

d is treat or not



Complete memory and specified order of decision explicitly represented in LIMID diagram.

Global and Maximum Strategies

Joint distribution induced by q is now

$$f_q(x) = \prod_{r \in \Gamma} p(x_r \,|\, x_{\mathrm{pa}(r)}) \prod_{d \in \Delta} \delta_d(x_d \,|\, x_{\mathrm{pa}(d)}).$$

A *global maximum strategy* $\hat{q}$ maximizes expected utility, i.e. it satisfies

$$\mathrm{EU}(\hat{q}) \geq \mathrm{EU}(q) \text{ for all strategies } q.$$

Global maximum strategies cannot be computed for general LIMIDs without complexity explosion.

Local Maximum Policies and Strategies

$\delta'_{d_0} * q$ obtained from q by replacing δ_{d_0} with δ'_{d_0}:

$$\delta'_{d_0} * q = \{\delta'_{d_0}\} \cup \{\delta_d : d \in \Delta \setminus \{d_0\}\}$$

A *local maximum policy* for q at d, satisfies

$$\mathrm{EU}\left(\tilde{\delta}_d * q\right) = \sup_{\delta'_d} \mathrm{EU}\left(\delta'_d * q\right).$$

$\tilde{q}$ is a *local maximum strategy* if all its policies are local maximum.

Related to concept of *Nash equilibrium*.

Single Policy Updating

SPU begins at fixed strategy q^0 and iteratively updates policies in some order to local maximum policies: If δ_0 is to be updated for q^l, we let

$$q^{l+1} = \delta_{d_0}^{l+1} * q^l$$

where *$\delta_{d_0}^{l+1}$ is chosen to be local maximum for q^{l+1}.* $\delta_{d_0}^{l+1}$ is chosen to be non-randomized.

SPU is related to Howard's *policy iteration* for infinite horizon Markov decision processes.

Properties of SPU

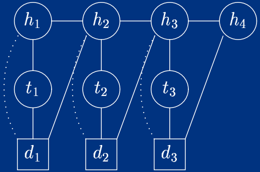
- *Each step of SPU improves* $\mathrm{EU}(q)$
- *For finite state spaces, SPU converges to local maximum strategy in finite number of steps*
- Conditions can be given for strategy found by SPU to be *global maximum*
- Each update can be found by *local computation algorithm* using message passing in junction tree
- Algorithms can be given for *simplifying* a LIMID before applying SPU, without loss of utility.

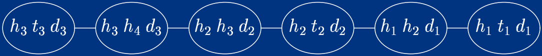
From LIMIDs to Junction Trees

Begin with LIMID $\mathcal{L}$ with all three types of node present.

1. *Moralization:* Form $\mathcal{L}^m$ from $\mathcal{L}$ by
   (a) adding undirected edges between any pair of unconnected nodes with common child
   (b) drop directions on all edges
   (c) remove utility nodes Υ
2. *Triangulation:* Form $\bar{\mathcal{L}}^m$ by adding edges to $\mathcal{L}^m$ to make it triangulated
3. *Form junction tree* of triangulated graph $\bar{\mathcal{L}}^m$.

Moral graph and Junction Tree of Example





Potentials and their Combination

For $W \subseteq V = \Delta \cup \Gamma$, a *potential* on W is a pair $\pi_W = (p_W, u_W)$ of functions on $\mathcal{X}_W$ with $p_W \geq 0$.

p_W is the *probability part* and u_W is the *utility part.*

We *identify two potentials* if they have identical probability parts and their utility parts agree almost surely w.r.t. their probability parts.

The *combination* $\pi_{W_1} \otimes \pi_{W_2}$ of π_{W_1} and π_{W_2} is

$$\pi_{W_1} \otimes \pi_{W_2} = (p_{W_1} p_{W_2}, u_{W_1} + u_{W_2}).$$

Marginalization of Potentials

The *sum-marginal* $\sum_{W\setminus W_1} \phi_W$ of a function ϕ_W is

$$\Big(\sum_{W\setminus W_1} \phi_W\Big)(x_{W_1}) = \sum_{y_W : y_{W_1} = x_{W_1}} \phi_W(y_W).$$

The *marginalization* $\pi_W^{\downarrow W_1}$ of π_W onto W_1 is

$$\pi_W^{\downarrow W_1} = \left(\sum_{W\setminus W_1} p_W \,,\ \frac{\sum_{W\setminus W_1} p_W u_W}{\sum_{W\setminus W_1} p_W} \right).$$

The convention $0/0 = 0$ is used throughout.

Properties of Potential Operations

- *Combination is commutative and associative:*

$$\pi_{W_1} \otimes \pi_{W_2} = \pi_{W_2} \otimes \pi_{W_1}$$

$$(\pi_{W_1} \otimes \pi_{W_2}) \otimes \pi_{W_3} = \pi_{W_1} \otimes (\pi_{W_2} \otimes \pi_{W_3}).$$

- *Marginalization is consonant:* For $W \supseteq W_1 \supseteq W_2$

$$(\pi_W^{\downarrow W_1})^{\downarrow W_2} = \pi_W^{\downarrow W_2}.$$

- *Marginalization is distributive over combination:*

$$(\pi_{W_1} \otimes \pi_{W_2})^{\downarrow W_1} = \pi_{W_1} \otimes \pi_{W_2}^{\downarrow W_1}.$$

These properties are the Shafer–Shenoy axioms!

Loading the Junction Tree

Begin with an initial strategy q for LIMID $\mathcal{L}$ and assign vacuous potentials $(p = 1, u = 0)$ to all cliques

1. Assign each $v \in \Delta \cup \Gamma$ to some clique $C(v) \supseteq \mathrm{fa}(v)$
2. For each $v \in \Delta \cup \Gamma$, multiply p_v or δ_v, as appropriate, onto probability part of $\pi_{C(v)}$.
3. Assign each $u \in \Upsilon$ to some clique $C(u) \supseteq \mathrm{pa}(u)$
4. For each $u \in \Upsilon$, add U_u to utility part of $\pi_{C(u)}$.

Joint potential is now

$$\pi_V = \otimes\{\pi_C : C \in \mathcal{C}\} = (p_V, u_V) = (f_q, U).$$

Collecting Messages

To update policy at d during SPU, choose clique R as root of $\mathcal{T}$ with $R \supseteq \mathrm{fa}(d)$.

Place one mailbox on each branch of $\mathcal{T}$, for messages.

Next, *collect messages* towards R by allowing $C \in \mathcal{T}$ to send iff it has received from neighbours further from root.

Here, a *message* $\pi_{C_1 \to C_2}$ from C_1 to C_2 is

$$\pi_{C_1 \to C_2} = \left(\pi_{C_1} \otimes \left(\otimes_{C \in \mathrm{ne}(C_1) \setminus \{C_2\}} \pi_{C \to C_1}\right)\right)^{\downarrow C_2},$$

where $\mathrm{ne}(C_1)$ are neighbours of C_1 and $\pi_{C \to C_1}$ is the message from C to C_1.

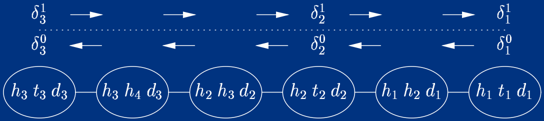
Updating Policy at d during SPU

Shafer-Shenoy axioms ensure $\pi_V^{\downarrow R} = \pi_R \otimes (\otimes_{C:C \neq R} \pi_C))^{\downarrow R}$ so updating can be done as:

1. *Retract* policy for d from π_R to obtain $\tilde{\pi}_R$.
2. *Collect* to R to obtain $\pi_R^* = (\tilde{\pi}_R \otimes (\otimes_{C:C \neq R} \pi_C))^{\downarrow R}$
3. *Marginalize* further to $\pi_{\mathrm{fa}(d)}^* = (\pi_R^*)^{\downarrow \mathrm{fa}(d)}$.
4. *Optimize* by finding $\tilde{\delta}_d(x_{\mathrm{pa}(d)})$ to maximize utility in $\mathrm{fa}(d)$
5. *Complete* by multiplying $\tilde{\delta}_d$ onto π_R^* to get $\tilde{\pi}_R^*$.

Partial Collect Propagation

When repeating message passing for other decisions d, *messages only need to be collected from previous root to new root.* This is *Partial collect propagation:*



Flows of messages in basic example during one cycle of SPU when updating first d_3, then d_2, and finally d_1.

Irrelevance and Separation

A *trail* τ from node a to node b in a LIMID $\mathcal{L}$ is *blocked* by S if it contains a node $n \in \tau$ such that

- either $n \in S$ and arcs of τ do not meet head-to-head at n, or
- n and all its descendants are not in S, and arcs of τ meet head-to-head at n.

A trail that is not blocked is *active.* Two subsets A and B of nodes are *d-separated* by S if all trails from A to B are blocked by S. We write $A \perp_{\mathcal{L}} B \,|\, S$.

Graphoid axioms

For all disjoint subsets A, B, C, and D of nodes of $\mathcal{L}$:

(C1) if $A \perp_{\mathcal{L}} B \,|\, C$ then $B \perp_{\mathcal{L}} A \,|\, C$;

(C2) if $A \perp_{\mathcal{L}} B \,|\, C$ and $D \subseteq B$, then $A \perp_{\mathcal{L}} D \,|\, C$;

(C3) if $A \perp_{\mathcal{L}} B \,|\, C$ and $D \subseteq B$, then $A \perp_{\mathcal{L}} (B \setminus D) \,|\, (C \cup D)$;

(C4) if $A \perp_{\mathcal{L}} B \,|\, C$ and $A \perp_{\mathcal{L}} D \,|\, (B \cup C)$, then $A \perp_{\mathcal{L}} (B \cup D) \,|\, C$;

(C5) if $A \perp_{\mathcal{L}} B \,|\, (C \cup D)$ and $A \perp_{\mathcal{L}} C \,|\, (B \cup D)$ then $A \perp_{\mathcal{L}} (B \cup C) \,|\, D$.

Reduction of LIMIDs

A node $a \in \mathrm{pa}(d)$ is *non-requisite for d* if

$$(\Upsilon \cap \mathrm{de}(d)) \perp_{\mathcal{L}} a \mid (\mathrm{fa}(d) \setminus \{a\}).$$

Arcs from non-requisite parents can be removed without loss of utility.

Any LIMID $\mathcal{L}$ has a unique minimal reduction $\mathcal{L}_{\min}$ obtained by successive removal of non-requisite arcs.

Extremal Decision Nodes

A decision node d is *extremal* in $\mathcal{L}$ if

$$(\Upsilon \cap \mathrm{de}(d)) \perp_{\mathcal{L}} \mathrm{fa}(\Delta \setminus \{d\}) \mid \mathrm{fa}(d).$$

An *optimum policy* for d is one which is a local maximum policy at d *for all strategies* q.

The *uniform strategy* is the strategy where all policies are uniform distributions.

If d is extremal in $\mathcal{L}$, then it has an optimum policy. Further any local maximum policy at d for the uniform strategy is an optimum policy for d.

Soluble LIMIDs

If d_0 is extremal, we may begin with uniform strategy and optimize δ_0 locally as described. *This ensures δ_0 to be optimum for all strategies.*

We may then convert d_0 to chance node and search for new extremal decision node d_1, etc.

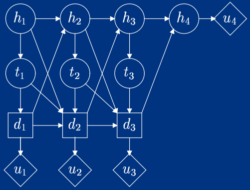
LIMID is *soluble* if this converts all decision nodes.

Starting from uniform strategy and updating by SPU in this order *yields global maximum strategy in a single cycle.*

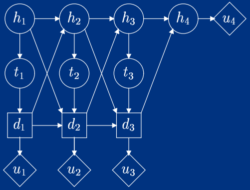
Influence diagrams are soluble.

Reductions of soluble LIMIDs are soluble.

Soluble Modifications and Bounds



(a)



(b)

Soluble modifications, such as (a) and its minimal reduction (b), *can give narrow upper bounds on optimal expected utility,* to evaluate quality of local maximum strategy found by SPU.

Example revisited

Best LIMID strategy (found by SPU):

$\delta_1 = never$, $\delta_2 = \delta_3 = direct$.

Best ID strategy:

$\delta_1 = never$, δ_2: treat when both t_1 and t_2 are positive. δ_3: treat if you did last time or if t_3 is positive.

Expected utilities of selected strategies:

always	*never*	*direct*	best LIMID	best ID
586	669	718	727	729

Upper bound on EU obtained from soluble cover (b): 732.

Summary

- *LIMIDs represent wider class* of decision problems
- *SPU by message passing is efficient* way of calculating locally optimal policies, also for ordinary IDs
- Algorithms for *reducing LIMIDs* to be minimal
- There are algorithms which *identify soluble LIMIDs*
- LIMIDs can be used to *find bounds,* e.g. for optimal strategies in POMDPs (Nilsson and Hoehle 2001)
- LIMIDs can be modified to MAIDs, (Multi Agent IDs) for *games* (Koller et al. 2001, Smerilli 2001)
- Paper in *Management Science* September 2001.