Report Number: WUCS-92-51

1992-12-01

Exact Dominance without Search in Decision Trees

Authors: Nilesh L. Jain and Ronald P. Loui

In order to improve understanding of how planning and decision analysis relate, we propose a hybrid model containing concepts from both. This model is comparable to [Hartman90], with slightly more detail. Dominance is simple concept in decision theory. In a restricted version of our model, we give conditions under which dominance can be detected without search: that is, it can be used as a pruning strategy to avoid growing large trees. This investigation follows the lead of [Wellman87].
The conditions seem hard to meet, but may nevertheless be useful in forward-chaining situations without focus, such as [Breese87]. It may be possible to extend this work to produce better heuristic pruning based on inexact dominance and heuristic ability.
Mainly, we contribute a detailed study of particular concept in a hybrid model that is the most detailed to date, further clarifying the relation between the two main paradigms for reasoning about preference among actions.
... **Read complete abstract on page 2.**

Exact Dominance without Search in Decision Trees

Complete Abstract:

In order to improve understanding of how planning and decision analysis relate, we propose a hybrid model containing concepts from both. This model is comparable to [Hartman90], with slightly more detail.

Dominance is simple concept in decision theory. In a restricted version of our model, we give conditions under which dominance can be detected without search: that is, it can be used as a pruning strategy to avoid growing large trees. This investigation follows the lead of [Wellman87].

The conditions seem hard to meet, but may nevertheless be useful in forward-chaining situations without focus, such as [Breese87]. It may be possible to extend this work to produce better heuristic pruning based on inexact dominance and heuristic ability.

Mainly, we contribute a detailed study of particular concept in a hybrid model that is the most detailed to date, further clarifying the relation between the two main paradigms for reasoning about preference among actions.

Exact Dominance without Search in Decision Trees

Nilesh L. Jain and R. P. Loui

WUCS-92-51

December 1992

Department of Computer Science
Washington University
Campus Box 1045
One Brookings Drive
St. Louis MO 63130-4899

Exact Dominance without Search in Decision Trees

Nilesh L. Jain
Medical Informatics Program and Dept. of Computer Science
Washington University

R. P. Loui
Depts. of Computer Science and Philosophy
Washington University[1]

Abstract.

In order to improve understanding of how planning and decision analysis relate, we propose a hybrid model containing concepts from both. This model is comparable to [Hartman90], with slightly more detail.

Dominance is a simple concept in decision theory. In a restricted version of our model, we give conditions under which dominance can be detected without search: that is, it can be used as a pruning strategy to avoid growing large trees. This investigation follows the lead of [Wellman87].

The conditions seem hard to meet, but may nevertheless be useful in forward-chaining situations without focus, such as [Breese87]. It may be possible to extend this work to produce better heuristic pruning based on inexact dominance and heuristic utility.

Mainly, we contribute a detailed study of a particular concept in a hybrid model that is the most detailed to date, further clarifying the relation between the two main paradigms for reasoning about preference among actions.

[1]In conversations with Mike Wellman, Jordan Kimberg, Michael Kahn, and Ian Flanigan. Jain is supported in part by National Library of Medicine NLM 5-T-15-LM07049. Loui is supported by National Science Foundation R-9008012 and The Center for Intelligent Computing Systems. Box 1045, Washington University, St. Louis MO 63130. loui@ai.wustl.edu. This paper was presented at the Workshop on Normative Systems, Palos Verdes, 1991.

1 Planning and Decision Trees

Planning's focus is on the compositionality of state space and state description, on the modularity of specification, and on computational cost. Decision theory's focus is on the structure of preference as it relates to measures of uncertainty.

No model yet in AI has found wide acceptance for generating a planning-style search space with quantitative probabilities and quantitative utilities.[2]

Earlier work by the second author identifies two principal obstacles [Loui90]. The first is the representation of utility, a matter of pragmatics. The mapping to reals of state descriptions in FOL is large. Regularities in the mapping permit representational shorthand; clever shorthands allow more regularities to be exploited. The second obstacle is finding a place for search, a matter of philosophy. Bayesian foundations, especially the independence postulate, a.k.a. The Sure-Thing Principle, render search meaningless. Why re-open a bound node if the existing valuation must equal its children's weighted aggregate value? There is no such thing as heuristic utility in the Bayesian model.

Decision theorists have considered compositionality of state space (discrete control), compositionality of state description (multi-attribute theory), and computational cost (branch and bound). This work differs in its palpable connection to planning. Multiple attributes are a special case of more expressive languages for describing state. In comparison, our descriptions of the world are more permissive about uncertainty.[3] Heuristic utility is a part of our model, which is strictly non-Bayesian. Finally, this work is keenly interested in control of search, which is not a traditional interest of decision theory in systems science, management science, economics, or philosophy.

Implicit in this work is that decision models are constructed from knowledge bases that define possible decision trees of combinatorial depth and description. Search and pruning are major concerns.[4]

Assume sufficient regularity in agent-preference. Assume that utility can be heuristic: that deeper, more detailed analysis is preferred to shallower, less detailed analysis. Assume that the search process avoids forming competing analyses of non-comparable depth.[5] The model next presented aims to completely different problems that would continue to plague the computational decision theorist.

[2]But see the recent thesis, [Hartman90], an independent, similar model. Another notable effort is [Star88].

[3]This is true even of our much simplified model, later in the paper.

[4][Breese87] and [Wellman87] are recent examples of this view, which Simon institutionalized.

[5]That would correspond to two arguments for decision, neither of which defeats the other.

2 General Model

This model is actually is not as general as it could be, since unnecessary ontological distinctions are made. However, it is more general than the very specialized model that follows it. Mostly, it is an attempt to bring planning with temporal reasoning together with the idea of conditional action, maximizing expected utility, and decision trees.

nodes. N is a set of *nodes.*[6] $n_0 \in N$ is the current node. Nodes are *described* by $d(n)$, which maps elements of N into consistent sets of sentences in some language, L. L is FOL in STRIPS or situation calculus; it is $\{a_0...a_k\}\times \mathbb{R}$ in real-valued k-attribute decision models; it is a temporal meta-language in contemporary thinking on planning.

Envisionment and analysis develops a tree of nodes. In decision analysis, the tree represents three distinct things: the temporal relations among actions, the analysis of uncertainty, and progression of search.[7] In this general model, the tree need represent only the latter two.

inference. Sentences in L may be interdependent; there may be intra-node inference. "on(a,b)","on(b,c)" $\in d(n)$ requires "above(a,c)"$\in d(n)$. Likewise, '*prob*("loaded(gun)")$>\frac{1}{2}$'$\in d(n)$ when '*is-provable* "loaded(gun)" '$\in d(n)$.[8] If "$< widgets, 3 >$" $\in d(n)$ then "$< widgets, 4 >$" must not be. If $d(s)$ contains DURING(*cough*, *main_course*), DURING(*main_course*, *dinner*), then it contains DURING(*cough*, *dinner*). We will not restrict the form of intra-node inference rules.

time. Note that sentences in a temporal logic can be part of the description of a node; times cited in a description can range freely. Thus in our model, ancestor relations between nodes need not impose temporal order. Depending on the temporal expressiveness of L, it may sometimes be incorrect to view a node as associated with a temporally distinct state of the world. It is always correct to view a node as a distinct stage in planning. Paths of nodes have to do with *planning-succession*, not necessarily with *world-succession*.

uncertainty. There is *descriptive uncertainty* when $d(n)$ is consistent but not maximal consistent. In that case, for any p such that $\{p\}\cup d(n)$ is consistent, we presume the probability $prob(p \mid d(n))$ is calculable, as a real or closed interval of reals.

[6]Some might call them "states", but see the paragraph on time.

[7]Mike Wellman points out that decision theory along the lines of Savage or Jeffrey contains no assumption about the temporal relations of actions; Raiffa's pictures of decision trees are superfluous. However, since the practice of drawing decision trees is so pervasive, it will be our reference for comparison.

[8]In this case, L is a meta-language of an FOL language.

Some nodes are *analysis nodes*, nodes at which uncertainty in the description is (partially) resolved. Children of analysis nodes have less uncertain descriptions (they are logically stronger); sibling nodes have descriptions that are inconsistent.

utility. For any node's description, utility u is also calculable, as interval- or point-valued. Bayesians require that $u(d(n)) = prob(p \mid n)u(\{p\} \cup d(n)) + prob(\bar{p} \mid n)u(\{\bar{p}\} \cup d(n))$, which we do not. Omitting this requirement allows heuristic utility.

action. A is a set of *action primitives*, or *primitives.* Primitives can be composed to form *(action) composites.* Included in a composite are the temporal relations among the primitives composed. In the simplest case, there are no temporal relations and the composite is an unordered set of actions. In classical decision analysis and situation calculus, a composite is linearly ordered: a sequence of primitives. In an expressive temporal logic, a composite may be a poset of primitives, or an even more complex entity invoking relations of temporal overlap and containment, which must be represented as a set of sentences.

A primitive $a \in A$ scheduled with time constraints t is written: $a@t$. Write $a_1@t_1$ composed with $a_2@t_2$ as: $a_1@t_1\&a_2@t_2$. In decision analysis, "&" is an operation that appends to sequences. In situation calculus, it is "|". In temporal logic, it is "∪", operating on the sets of sentences describing temporal constraints.[9]

control. Some nodes are *control nodes*, nodes at which the agent may schedule primitives.

In decision analysis, control is defined when a node is made a choice node; rule-governed determination of control is not an issue. In control theory, all controls are applicable at all times.

In planning, control of primitives is defined by preconditions of actions.

In general, it is not the case that any primitive can be scheduled to bear any temporal relations.

Suppose the description of a node includes sentences in a temporal logic describing the relation of some interval, ON_TRAIN, to reference times. Perhaps at an ancestor node, the action primitive, BOARD_TRAIN, was scheduled, which implied that ON_TRAIN bore those temporal relations. It is within the control of the agent further to schedule READ_BOOK to occur during ON_TRAIN. It may *not* be within the control of the agent to schedule READ_BOOK to co-begin BOARD_TRAIN. It may also be within the control of the agent to schedule further constraints on this same READ_BOOK action at some later node. Control of primitives and their schedulability would be defined by constraints in FOL.

[9] The symbol "&", is chosen because it sometimes has asymmetric temporal significance, and sometimes does not.

We can continue to use the term *preconditions* to name the rules that define schedulability, regardless of temporal complexity; and to use the notation $pre(a)$.

dynamics. Scheduling primitive actions at a node adds to the action composite whose execution is currently under contemplation at that node. Scheduling primitive actions also makes the description of the resultant node differ from that of the control node.

Associated with each action primitive are inter-node inference rules which we continue to call *postconditions*: $post(a)$. These are the "causal" rules of contemporary planning.[10] In STRIPS, postconditions are both the ADD and $DELETE$ lists; in planning based upon temporal logic, node descriptions grow monotonically, so only an ADD list is needed.

For $a \in A$, and $n \in N$, the description of the node that contemplates a scheduled in n is a function of two things: the description of n, and the postconditions associated with a. We are reluctant to name the descendant node because it varies with L. In decision analysis, the name is $< n, a >$; in situation calculus, it is $a \mid n$; in temporal logic, it is a combination of the node, the action primitive, and the constraints on the scheduling of the primitive: $n\&a@t$.

It is a strong constraint to require the resultant description to be a *function* of node description and postcondition.[11] The reader is encouraged to consider how this relates to the frame problem and Yale Shooting.

events. E is a set of events.[12] Like action primitives, associated with each event are its preconditions and postconditions. Events occur whenever their preconditions, $pre(e)$ are satisfied. Unlike an action primitive, calculable probability determines which of the event's postconditions $post_i(e)$ is realized.[13] Nodes at which events are considered are *event nodes*.[14]

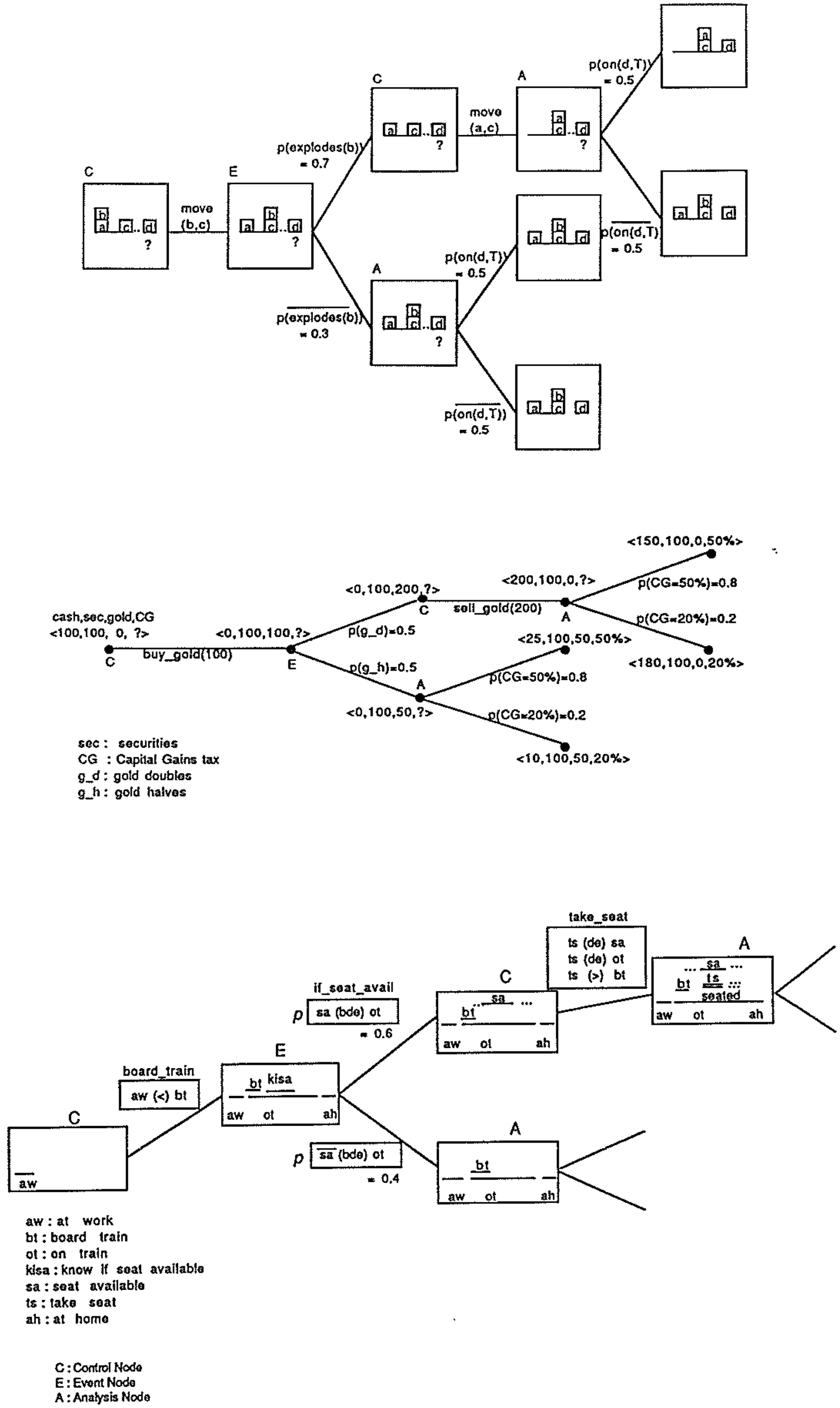
[10] Invoking the concept of causality is dubious for several reasons: *e.g.*, properties can be determined to hold at a time in part as the result of future action; one example is the property of BEING_THE_YEAR'S_BEST in November, which depends on what happens in December.

[11] This is not to say that all descriptive uncertainties are resolved. It says that the description of the resultant node is well-defined by the information associated with the action that led to it. That description may still contain uncertainty; *i.e.*, it may not be maximal consistent. This formally precludes non-deterministic resource-bounded theorem provers except as an approximation to the model.

[12] These are events on which action can be conditioned. Mike Wellman's thesis calls them "observables," since some would call our "analysis nodes" events. Basically, by distinguishing events, we simplify the specification of conditional action.

[13] Trivial postconditions, a no-operation or null set, simulate non-occurrence of the event.

[14] In planning upon temporal logic, event nodes are analysis nodes. But in decision theory, event nodes have a control aspect: they signify world-times at which control of the agent is relinquished (which can be signified by their position in the tree because world-time and planning-time are conflated). Given this, a third kind of node is established for the analysis of events. In planning, control is never relinquished.



C
move (b,c)
E
p(explodes(b)) = 0.7
p(explodes(b)) = 0.3
C
A
move (a,c)
A
p(on(d,T)) = 0.5
p(on(d,T)) = 0.5
p(on(d,T)) = 0.5
p(on(d,T)) = 0.5
cash,sec,gold,CG
<100,100, 0, ?>
C
buy_gold(100)
<0,100,100,?>
E
p(g_d)=0.5
p(g_h)=0.5
<0,100,200,?>
C
sell_gold(200)
<200,100,0,?>
A
p(CG=50%)=0.8
p(CG=20%)=0.2
<150,100,0,50%>
<180,100,0,20%>
A
<0,100,50,?>
p(CG=50%)=0.8
p(CG=20%)=0.2
<25,100,50,50%>
<10,100,50,20%>
sec : securities
CG : Capital Gains tax
g_d : gold doubles
g_h : gold halves
C
aw
board_train
aw (<) bt
E
bt kisa
aw ot ah
if_seat_avail
p sa (bde) ot = 0.6
p sa (bde) ot = 0.4
C
bt sa
aw ot ah
take_seat
ts (de) sa
ts (de) ot
ts (>) bt
A
bt sa ts seated
aw ot ah
A
bt
aw ot ah
aw : at work
bt : board train
ot : on train
kisa : know if seat available
sa : seat available
ts : take seat
ah : at home
C : Control Node
E : Event Node
A : Analysis Node

Figure 1

In decision analysis, descriptive uncertainty can be modeled by events. In general, though, they are different. Events force information to be revealed at a time. Events can add sentences to a node's description that may figure in the preconditions of contemplated future action. If L is atemporal, events can cause a sentence in a node's description to be *removed* from the resultant node's description, *e.g.*, WALLET-IN-POCKET. Analysis of descriptive uncertainty can never do this.

Events have to do with time and epistemics. They schedule information revealed, or description altered, or alternative ways the world could be, while the agent can still respond.

For example, after GO_TO_BOX_OFFICE, the BUY_GOOD_SEATS action primitive cannot be scheduled unless its precondition, GOOD_SEATS_AVAILABLE holds at the time of scheduling. Normally, this would be an impasse, because uncertainty about the world cannot be revealed upon demand. But the GO_TO_BOX_OFFICE primitive's postcondition satisfies the precondition of the event REVEAL_WHETHER_GOOD_SEATS_AVAILABLE.[15]

Events allow descendant nodes to contemplate response to scenarios, thereby relieving the language of having to represent conditional action.[16]

Names of nodes *downstream* of event nodes include the event: $n\&a'@t'\&e@t''\&a'''@t'''$; thus, action composites (which omit events) are not necessarily node names.

Figure 1 shows three trees, each with a different L.

3 Simplified Model

We study dominance for exactly the reasons of [Wellman87]. Checking dominance is one form of meta-reasoning on decision trees. Little meta-reasoning can be done on complex object languages. We simplify.

The description of a node, $d(n)$, is composed of properties $\{p_1, ..., p_k\}$. Instead of sets of sentences, let d's range be 3-valued k-vectors: the i-th attribute is 1 if p_i is provably true, 0, if provably false, and ? if not determinable relative to fixed computation. Properties may be anchored at times: some attributes' values, once fixed, remain fixed; others can change with action and event. There is no intra-node inference. Scheduling of primitives is implicit in node order (thus, the $@t$ part of a scheduled primitive $a@t$ is redundant, and we refer simply to *actions*).

[15]The relation of knowledge to events and control is not fixed here. If the problem's model makes the precondition for buying good seats the *knowing* of whether they are available, then whether they are available is not enough. So *knowing* whether they are available must be the result of the event. If the precondition for action is just that they are available (as for an idiot savant who can sense good seats without knowing), then it suffices that availability is revealed by the event; we do not need knowledge of availability.

[16]Seen broadly, action composites *are* programs. Requiring linear order on action primitives removes parallelism. Allowing events eliminates the need for conditioning.

Postconditions have a limited form: they are masks that coerce attribute value. They operate on each attribute independently, and are unconditional.[17] For example, the mask $< 0, 1, \lambda, ? >$ for 4-vectors forces the first two attributes to false and true, respectively. It does not alter the third attribute, and it forces the fourth attribute to be of unknown truth value. It would coerce $<?, 1, 0, 0>$ to $< 0, 1, 0, ? >$. If an action primitive's postcondition mask forces an attribute to ?, then the primitive is *uncertainty-introducing*. If an action's precondition requires each attribute to be ? whenever its mask forces that attribute to 0 or 1, then it is *uncertainty-resolving*. We allow both kinds of actions.

Dominance for finitely drawn trees (dominance at a fixed horizon, one kind of heuristic dominance) occurs when two actions at a node can be succeeded by equivalent subtrees, with equivalent probabilities. Leaf nodes correspond: their valuations should consistently favor one action. In general, the horizon is not fixed. Dominance must hold for all ways in which the tree can be extended.

Example. The example in figure 2 will be used in the remaining sections. It examines HAL's situation in *2001: A Space Odyssey* when he decides that the crewmen are useless. HAL seeks the right sequence in which to kill all crewmembers. Relative to killing crewmen silently, killing in a way that could alert mission control has no structural effects; in each scenarion, it has relative disutility or the same utility .

4 Sufficient Conditions for Exact Dominance

An attribute is *permanent* if downstream nodes cannot change its value once its value is certain. Permanent attributes essentially record that actions and events have occurred. Dominance would be trivial if all attributes were permanent. Our conditions pertain to impermanence.

$$\begin{aligned}&\mathit{perm}(m) \text{ iff}\\ &\quad \text{for all } \eta \in A \cup E,\\ &\quad \text{if } [\mathit{post}(\eta)]_m \neq \lambda \text{ then } [\mathit{pre}(\eta)]_m = ?.\end{aligned}$$

(where $[d]_m$ is the value of the m-th attribute in a description d).

We renumber attributes so that those relevant to utility appear first.

$$\begin{aligned}&u\text{-}\mathit{irrelevant}(m) \text{ iff}\\ &\quad \text{for all } d \in \{0, 1, ?\}^k,\\ &\quad u(d) = u(d\,|_m^0) = u(d\,|_m^1) = u(d\,|_m^?);\end{aligned}$$

where $d\,|_m^v$ is the description d altered, making the m-th attribute v.

$$u_R = \{m : \neg u\text{-}\mathit{irrelevant}(m)\};$$

[17] Conditional change of value makes dominance all but impossible to ascertain.

actions

a_1 kill sleeping crewmen silently
a_2 kill sleeping crewmen, alerting mission control
a_3 kill Frank
a_4 kill Dave
a_5 tell mission control of killing

attributes

p_1 sleeping crewmen are dead
p_2 Frank is dead
p_3 Dave is dead
p_4 information about killing is available to mission control
p_5 mission control is awake

The preconditions and postconditions for the actions are

	pre	*post*
a_1	$\overline{p_1}$	$< 1, \lambda, \lambda, 0, \lambda >$
a_2	$\overline{p_1}$	$< 1, \lambda, \lambda, 1, \lambda >$
a_3	$\overline{p_2} \wedge p_1$	$< \lambda, 1, \lambda, \lambda, \lambda >$
a_4	$\overline{p_3} \wedge p_1$	$< \lambda, \lambda, 1, \lambda, \lambda >$
a_5		$< \lambda, \lambda, \lambda, 1, \lambda >$

$u(< p_1, p_2, p_3, p_4, p_5 >) = 10p_1 + 20p_2 + 30p_3 + 50(p_1 p_2 p_3 - p_4 p_5)$. Probability information will be given as needed.

Figure 2

without loss of generality, $u_R = \{1, ..., \#(u_R)\}$.

For a choice of actions a_i, a_j scheduled at n, let the attributes where they differ be

$$\Delta_{ij} = \{m : [d(n\&a_i)]_m \neq [d(n\&a_j)]_m\},$$

The following three conditions suffice for one action, a_i, dominating another, a_j, at a node, n.

$a_i DOM_n a_j$,
i.e., for all $n', u(n\&a_i\&n') \geq u(n\&a_j\&n')$, *etc.*,
if

1. (structural equivalence)
for all $\eta \in A \cup E$,
for all $\delta \in \Delta_{ij}$,

$$\delta \notin pre(\eta);$$

This says that no discernable difference at n can have a structural effect. It can be weakened, but not without combinatorics and theorem-proving among preconditions.

2. (conditional probabilistic equivalence)
for all $\Delta' \subseteq \Delta_{ij}$,
for all $m \in \{1, ..., k\}$,
for all $\Gamma \subseteq \{\ulcorner p_l = v \urcorner : l \in \{1, ..., k\}$ and $v \in \{0, 1\}\}$:

$$\text{prob}(\ulcorner p_m = v \urcorner \mid \Gamma \cup \{\ulcorner p_l = v \urcorner : perm(l) \text{ and } [d(n\&a_i)]_l = v\}) =$$

$$\text{prob}(\ulcorner p_m = v \urcorner \mid \Gamma \cup \{\ulcorner p_l = v \urcorner : perm(l) \text{ and } [d(n\&a_j)]_l = v\});$$

And a corresponding condition for event postconditions (substitute $post_i(e)$ for $\ulcorner p_m = v \urcorner$ as the object of probability, and quantify over e and i).

This says that given certain conditions, any attribute m takes on the value v with the same probability descending from a_i as from a_j. The conditions must contain what is permanent after action is taken in n. Counterxamples can be given to any weakening of this condition.[18]

[18] Quantifying over Γ makes the permanent attributes unnecessary, except that including the latter gives a lower bound on the conditions, which has computational usefulness.

3. (utility advantage for any differences maintained)

The following must hold for every Δ' s.t.
$\{l : perm(l)\} \cap \Delta_{ij} \subseteq \Delta' \subseteq \Delta_{ij}$:
For all ρ_i, ρ_j satisfying:

for all $l \in \Delta'$,
$[\rho_i]_l = [d(n\&a_i)]_l$ and $[\rho_j]_l = [d(n\&a_j)]_l$

it is the case that:

$$\Phi(\rho_j) \leq \Phi(\rho_i);$$

(and for some Δ', for some ρ_i, ρ_j satisfying the same conditions, $\Phi(\rho_j) < \Phi(\rho_i)$).

This says that no matter what combination of differences survives downstream (and all the permanent differences must survive), a_i's descendants have superior valuation. Again, this condition cannot be weakened.

Efficient Computation. The first condition is trivial.

The second condition is inferred through probabilistic irrelevance: either because conditional independence is asserted [Pearl88], or because there is ignorance about refined reference classes [Kyburg82]. If irrelevance is asserted, the condition is trivial. Checking irrelevance could involve combinatorics.

The last condition is determined by comparing vectors differing in some of the attributes in Δ_{ij}, but equivalent in all other respects.

Checking dominance will depend on how utility functions can be factored into $u_1, u_2, ...$, and aggregated through a function Φ. Assume w.l.o.g. there exists a partition $\{\pi_1, ..., \pi_\nu\}$ of u_R and functions $\{u_1, ..., u_\nu\}$, each $u_i : \pi_i \rightarrow \mathbb{R}$ (in the worst case, $\nu = 1$); and there is a regimen, Φ s.t.

for all $d_1, d_2 \in \{0,1\}^{\#(u_R)}$,
$u(d_1) \leq u(d_2)$ iff
$\Phi(< u_1(d_1), ..., u_\nu(d_1) >) \leq \Phi(< u_1(d_2), ..., u_\nu(d_2) >)$,

and Φ is separable in at least one of the following senses: for all i, γ, ξ, if $\gamma \leq \xi$ (thinking of γ and ξ as real-valued outputs of u_i), then

ceteris-paribus (cp-) (all other things being equal):
for all $\rho \in \mathbb{R}^\nu$, $\Phi(\rho \,|_i^\gamma) \leq \Phi(\rho \,|_i^\xi)$.

pareto (p-) (all other things being in the same direction):
for all $\rho_1, \rho_2 \in \mathbb{R}^\nu$,
if for all $j \neq i, u_j([\rho_1]_j) \leq u_j([\rho_2]_j)$
then $\Phi(\rho_1 |_i^\gamma) \leq \Phi(\rho_2 |_i^\xi)$.

lexicographic (lex-) (all more important things being in the same direction):
for all $\rho_1, \rho_2 \in \mathbb{R}^\nu$,
if for all $j < i, u_j([\rho_1]_j) \leq u_j([\rho_2]_j)$
then $\Phi(\rho_1 |_i^\gamma) \leq \Phi(\rho_2 |_i^\xi)$.

The table in figure 3 shows four functions, each separable in a different way. The lower lines in the table give the attribute in which to fix a difference, and the partition.

Lex-separability implies p-separability; p-separability implies cp-separability.

Let $\mathcal{AP}$ (Affected-Partitions) $= \{q : \pi_q \cap \Delta' \textit{is non-empty}\}$.

Let $\mathcal{AA}$ (Affected-Attributes) $= \bigcup_{\pi \in \mathcal{AP}} \{l : l \in \pi\}$.

Let $\mathcal{IA}$ (Iterator-Attributes) $= \{l : l \in \mathcal{AA} \text{ and } l \notin \Delta'\}$.
Also, $\mathcal{IA}(q) = \{l : l \in \pi_q\} \cap \mathcal{IA}$.

The iteration and amount for each kind of separability is:

(cp-separable):
Iterate over all of $\mathcal{IA}$, to fix the values of u_θ for every $\theta \in \mathcal{AP}$. If these values all favor a_i, then through cp-separability, the aggregate Φ favors a_i. The cost is $2^{\#(\mathcal{IA})} = 2^{\Sigma_{q \in \mathcal{AA}} \#(\mathcal{IA}(q))}$.

(p-separable):
For each $\theta \in \mathcal{AP}$, check for an a_i advantage, fixing the values of attributes in $\mathcal{IA} \cap \pi_\theta$. P-separability guarantees that independent advantages in each θ are reflected in the aggregate. The cost is $\Sigma_{q \in \mathcal{AP}} 2^{\#(\mathcal{IA}(q))}$.

(lex-separable):
Only the first affected partition need be checked for a_i advantage if there is such an advantage (otherwise, it has the same worst case as p-separability). The cost is $2^{\#(\mathcal{IA}(min_{q \in \mathcal{AP}}(q)))}$.

None of these will hold unless each u_l for $l \in \mathcal{AP}$ is separable in a sense very much like cp-separability (where, γ and ξ are fixed by Δ').

These costs are for each Δ', but should not be multiplied by the size of Δ_{ij}'s powerset. In fact, taking Δ' to be the entire Δ_{ij} gives the dominant term in all instances.

notseparable	*ceteris-paribus*	*pareto*	*lexicographic*
$u(< 0,0 >) = 9$	$u(< 0,0 >) = 8$	$u(< 0,0 >) = 3$	$u(< 0,0,0 >) = 0$
$u(< 0,1 >) = 3$	$u(< 0,1 >) = 3$	$u(< 0,1 >) = 8$	$u(< 0,0,1 >) = 1$
$u(< 1,0 >) = 8$	$u(< 1,0 >) = 9$	$u(< 1,0 >) = 7$	$u(< 0,1,0 >) = 10$
$u(< 1,1 >) = 4$	$u(< 1,1 >) = 4$	$u(< 1,1 >) = 9$	$u(< 0,1,1 >) = 11$
			$u(< 1,0,0 >) = 100$
			$u(< 1,0,1 >) = 101$
			$u(< 1,1,0 >) = 110$
			$u(< 1,1,1 >) = 111$
$\Delta = \{1\}$	Δ *is not relevant here*	$\Delta = \{1\}$	$\Delta = \{2\}$
partition $\{\{1\},\{2\}\}$	*partition* $\{\{1\},\{2\}\}$	*partition* $\{\{1\},\{2\}\}$	*partition* $\{\{1\},\{2,3\}\}$

Figure 3

Permanence of attributes was used only for the conditioning probabilities. In the examination of differential utility, we do not assume that all attributes whose uncertainties are resolved will retain their values at downstream nodes (although the more numerous the permanent attributes, the fewer subsets Δ' to be checked). The only reason that probability conditions need be permanent is that conditional independence must hold with respect to something; the only attributes that retain their values downstream are permanent.

The Example Again. For actions a_1 and a_2, $\Delta_{12} = \{4\}$. All the attributes are u-relevant; p_4 is permanent. The partition on the attributes is $\pi_1 = \{1,2,3\}$ and $\pi_2 = \{4,5\}$. The utility functions on the partitions are $u_1(< p_1, p_2, p_3 >) = 10p_1 + 20p_2 + 30p_3 + 50p_1p_2p_3$ and $u_2(< p_4, p_5 >) = -30p_4p_5$. The aggregator Φ is simply $u_1 + u_2$. Clearly, Φ is p-separable. $d(n_0)$ is $< 0, 0, 0, ?, ? >$.

We see that actions a_1 and a_2 can be scheduled at node n_0. We shall now show that $a_1 DOM_{n_0} a_2$.

The first requirement for dominance is structural equivalence: none of the elements of Δ_{12} appear in any of the preconditions. We have not given any probabilistic information: assume all probabilities are conditionally independent of $\ulcorner p_4 = 1 \urcorner$, and of $\ulcorner p_4 = 0 \urcorner$.[19]

Finally, we want to check for the utility advantage. In this example, $\mathcal{AP} = \{2\}$, $\mathcal{AA} = \{4,5\}$ and $\mathcal{IA} = \{5\}$. We now need to compare the values of Φ for $p_5 = 0$ and $p_5 = 1$. We see that in each case, the utility is greater for action a_1. Thus, at n_0, any action composite beginning with a_2 is dominated by the corresponding action composite beginning with a_1.

5 Other Forms of Dominance

Dominated Riskless Primitives. Consider a_5 in the example: telling mission control of the killing. It is an action that serves no useful function; there is no scenario in which its postcondition contributes positively to utility. Is it dominated? At n_0, it can be compared with alternative primitives. But the other primitives have different descendant structure: after a_5, any action is legitimate, including a_5; after any other action, that action may not be repeated.

We call this a dominated riskless primitive. It is characterized by contributing postconditions that (1) are u-relevant, non-structural (do not appear in any preconditions), and cannot have positive effects on utility; or (2) are u-irrelevant, non-structural, and increase the probability of dominated riskless events; or (3) are u-irrelevant and occur only in the preconditions of a dominated riskless primitive or dominated riskless event. Dominated riskless events are similarly characterized.

They are eliminable through pre-processing.

[19] Usually, this would be conditional independence of the difference, given what is permanent. But in this case, the difference is the only thing that is permanent.

There are also riskless primitives that can only contribute positively to utility; whenever their preconditions are satisfied, it behooves the agent to schedule them.

Stochastic Dominance. Another important class of dominance is stochastic dominance. There is stochastic dominance between two primitives if their cumulative probability distributions over utility values never cross. If so, the lower cumulative probability curve is preferred.

We currently seek conditions for stochastic dominance in our model. Instead of seeking stability in the utility function over combinations of attributes, we seek stability in conditional cumulative probability distributions.

Mixed Dominance. It is also possible to determine that an action dominates another because uncertainty allows a partition of descendants: on the first partition, one kind of dominance holds; on the other partition, a different kind holds.

6 Discussion of Dominance

How prevalent is dominance? This depends on the knowledge base. As [Wellman87] notes in a slightly different way, dominance does not guarantee that the remaining undominated alternatives are small in number; but that's because of the large initial number of alternatives, not because dominance is ineffective. When two primitives are the only control possibilities at a node, dominance often halves the number of action composites: m independent dominance observations can divide the number by 2^m.

As a rule of thumb, dominance will occur whenever two primitives are essentially of the same type, but differ in some non-structural attribute which has factorable effect on utility. Dominance happens when there is a paucity of probability information, out of which much probabilistic irrelevance arises (*e.g.*, when probabilities are generated through reference classes). Our conditions are more general than this rule of thumb, but in practice, the added generality is usually not worth the bother of checking.

This paper is not about heuristic dominance, which can exclude a class of composites prematurely; we sought conditions under which upstream exclusion was guaranteed to be correct.

Still, there is an important class of heuristic analyses worth mentioning: those based on abstraction of detail [Loui90]. Some heuristic strategies systematically ignore low-probability events or postconditions, some restrict focus to a subset of attributes as they relate to dynamics, control, and utility. In these strategies, dominance may appear frequently. If, at a level of abstraction, analysis can ignore minor structural differences, the first two conditions for dominance are more easily satisfied, and the third is more easily ascertained.

The irony of this study and Wellman's is that dominance is unimportant in decision theory. The tree is stipulated, not generated, and at minimum, the n leaves are labeled with their utilities. The value of at most $n-1$ interior nodes must be inferred, with a little multiplication and addition for each analysis node, and a max-element query. If structural isomorphism is obvious, dominance requires only n comparisons, but risks that the computation will be fruitless, requiring the induction of node-value anyway.

Dominance is a panda's thumb, a concept that lost its meaning when the theory of risk parted the theory of games. Games had compositional structure: decision trees did not. As we embed the theory of risk once again in a formalism with compositionality, we try to use that thumb again.

7 Bibliography

Allen, J. "Towards a general theory of action and time," *Artificial Intelligence 23*, 1984.

Allen, J., Tate, A., and Hendler, J. *Readings in Planning.* Morgan Kaufman, 1990.

Bell, D., Raiffa, H., and Tversky, A. *Decision Making.* Cambridge, 1988.

Breese, J. "Knowledge representation and inference in intelligent decision systems," doctoral dissertation, Stanford University, 1987.

Davidson, D. *Essays on Actions and Events.* Oxford, 1980.

Feldman, J. and Sproull, R. "Decision theory and AI II: the hungry monkey," *Cognitive Science 1*, 1977.

Gärdenfors, P. and Sahlin, N., eds. *Decision, Probability, and Utility.* Cambridge, 1988.

Hanks, S. and McDermott, D. "Nonmonotonic logics and temporal projection," *Artificial Intelligence 33*, 1987.

Hartman, L. "Decision theory and the cost of planning," doctoral dissertation, University of Rochester, 1990.

Kreps, D. *Notes on The Theory of Choice.* Westview, 1988.

Kyburg, H. "The reference class," *Philosophy of Science 50*, 1982.

Loui, R. "Two heuristic functions for decision," in Proc. AAAI Spring Symposium on Limited Rationality, 1988.

Loui, R. "Defeasible specification of utilities," in H. Kyburg, et.al., eds., *Knowledge Representation and Defeasible Reasoning.* Kluwer, 1990.

March, J. "Bounded rationality," in J. Elster, ed., *Rational Choice.* New York University, 1986.

McCarthy, J. and Hayes, P. "Some philosophical problems from the standpoint of artificial intelligence," in B. Webber and N. Nilsson, eds., *Readings in AI.* Morgan Kaufman, 1981.

Pearl, J. "Evidential reasoning under uncertainty," in H. Shrobe, ed., *Exploring Artificial Intelligence.* Morgan Kaufman, 1988.

Shoham, Y. *Reasoning about change.* MIT, 1987.

Simon, H. "A behavioral model of rational choice," in *Models of Thought.* Yale, 1979.

Star, S. "Generating decision structures and causal explanations for decision making," *Proc. Uncertainty in AI*, 1988.

Wellman, M. "Dominance and subsumption in constraint-posting planning," *Proc. IJCAI*, 1987.