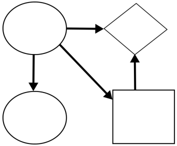
# Decision Graphs

## Probabilistic Graphical Models

L. Enrique Sucar, INAOE



# Outline

# Decision Models



- *Decision models*, whose aim is to help the decision maker to choose the *best* decisions under uncertainty.
- The best decisions are those that maximize the expected utility of an agent, given its current knowledge (evidence) and its objectives, under a decision-theoretic framework - *rational agents*
- We will describe two types of modeling techniques for problems with one or few decisions: decision trees and influence diagrams

# Decision Theory



- Decision Theory provides a normative framework for decision making under uncertainty
- It is based on the concept of *rationality*, that is that an agent should try to maximize its utility or minimize its costs
- An agent is not sure about the results of each of its possible decisions – consider the *expected utility*, which makes an average of all the possible results of a decision, weighted by their probability
- A rational agent must select the decision that maximizes its expected utility

# Fundamentals



- The principles of decision theory were initially developed in the classic text by Von Neuman and Morgensten, *Theory of Games and Economic Behavior*
- They established a set of intuitive constraints that should guide the preferences of a rational agent, which are known as the axioms of utility theory

# Elements

**Alternatives.** Are the choices that the agent has and are under his control. Each decision has at least two alternatives

**Events.** Are produced by the environment or by other agents; they are outside of the agent's control. Each random event has at least two possible results, and we can assign a probability to each one.

**Outcomes.** Are the results of the combination of the agents decisions and the random events. Each possible outcome has a different preference (utility) for the agent.

**Preferences.** These are established according to the agent's goals and objectives. They establish a value for the agent for each possible result of its decisions.

# Lottery

- The different scenarios are called *lotteries*
- In a lottery each possible outcome or *state*, $A$, has a certain probability, $p$, and an associated preference to the agent which is quantified by a real number, $U$
- A lottery $L$ with two possible outcomes, $A$ with probability $p$, and $B$ with probability $1 - p$, will be denoted as:

$$L = [P, A; 1 - P, B]$$

- If an agent prefers $A$ rather than $B$ it is written as $A \succ B$, and if it is indifferent between both outcomes it is denoted as $A \sim B$
- An outcome can be an atomic state or another lottery

# Axioms of Utility Theory



*Order:* Given two states, an agent prefers one or the other or it is indifferent between them.

*Transitivity:* If an agent prefers outcome $A$ to $B$ and prefers $B$ to $C$, then it must prefer $A$ to $C$.

*Continuity:* If $A \succ B \succ C$, then there is some probability $p$ such that the agent is indifferent between getting $B$ with probability one, or the lottery $L = [p, A; 1 - p, C]$.

# Axioms

*Substitutability:* If an agent is indifferent between two lotteries *A* and *B*, then the agent is indifferent between two more complex lotteries that are the same except that *B* is substituted for *A* in one of them.

*Monotonicity:* There are two lotteries that have the same outcomes, *A* and *B*. If the agent prefers *A*, then it must prefer the lottery in which *A* has higher probability.

*Decomposability:* Compound lotteries can be decomposed into simple ones using the rules of probability.

# Utility Principle



- **Utility Principle**: If an agent's preferences follow the axioms of utility, then there is a real-valued utility function $U$ such that:
  1. $U(A) \succ U(B)$ if and only if the agent prefers $A$ over $B$,
  2. $U(A) = U(B)$ if and only if the agent is indifferent between $A$ and $B$.
- **Maximum Expected Utility Principle**: The utility of a lottery is the sum of the utilities of each outcome multiplied by its probability:

$$U[P_1, S_1; P_2, S_2; P_3, S_3; ...] = \sum_j P_j U_j$$

# Expected Utility



- Expected utility (EU) of a certain decision $D$ taken by an agent, considering that there are $N$ possible results of this decision, each with probability $P$:

$$EU(D) = \sum_{j=1}^{N} P(result_j(D))U(result_j(D))$$

- The principle of **Maximum Expected Utility** states that a rational agent should choose an action that maximizes its expected utility

# Utility of Money



- In many cases it seems natural to measure utility in monetary terms; the more money we make based on our decisions, the better
- Suppose that you are participating in a game, such as those typical TV shows, and that you have already won one million dollars. The host of the game asks you if you want to keep what you have already won and finish your participation in the game, or continue to the next stage and gain $3,000,000
- The host will just flip a coin and if it lands on *heads* you will get three million, but if it lands on *tails* you will loose all the money you have already won
- What will your decision be?

# Example

- Let's see what the principle of maximum expected utility will advise us if we measure utility in dollars
- We calculate *Expected monetary value* (EMV):

  D1: EMV(D1) = $1 \times \$1,000,000 = \$1,000,000$

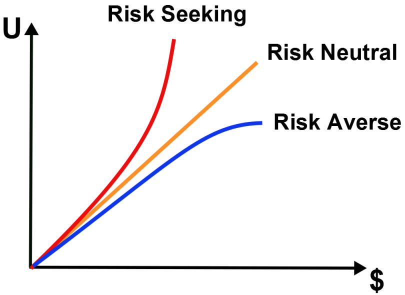
  D2: EMV(D2) = $0.5 \times 0 + 0.5 \times \$3,000,000 = \$3,000,000$
- Most of us would probably select to keep the one million. Are we not being *rational*?

# Utility vs. Monetary Value



- The relation between utility and monetary value is not linear for most people; instead they have a logarithmic relation which denotes *risk aversion*
- It is approximately linear for low values of money, but once we have a large amount of money (the amount will depend on each individual), the increase in utility given more money is no longer linear
- The utility–monetary value relation varies from person to person (and organizations) depending on their perception of risk; there are three basic types: risk aversion, risk neutral and risk seeking

# Utility-Monetary Value Relations



# Decision Trees

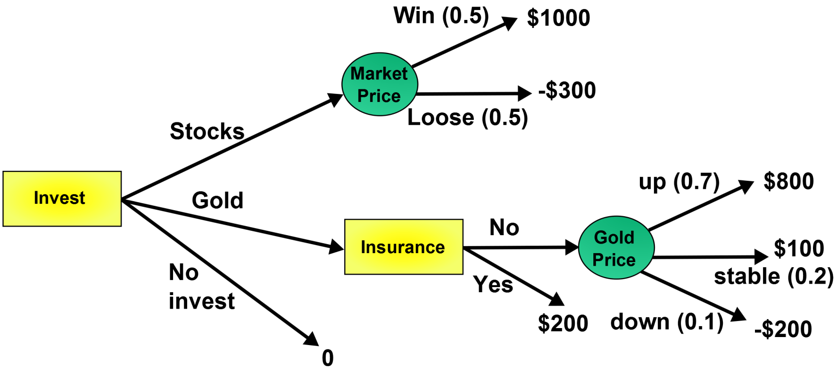


- A decision tree is a graphical representation of a decision problem, which has three types of elements or nodes that represent the three basic components of a decision problem: decisions, uncertain events and results

# Elements

- A *decision node* is depicted as a rectangle which has several *branches*, each branch represents each of the possible alternatives present at this decision point
- An *event node* is depicted as a circle, and also has several branches, each branch represents one of the possible outcomes of this uncertain event. A probability value is assigned to each branch
- The *results* are annotated with the utility they express for the agent, and are usually at the end of each branch of the tree (the leaves)

# DT - Example




- What should the investor decide?

# DT Evaluation



- To determine the best decision for each decision point, according to the maximum expected utility principle, we need to *evaluate* the decision tree
- The evaluation of a decision tree consists in determining the values of both types of nodes, decision and event nodes:
  - The value of a decision node $D$ is the maximum value of all the branches that emanate from it: $V(D) = max_j U(result_j(D))$.
  - The value of an event node $E$ is the expected value of all the branches that emanate from it, obtained as the weighted sum of the result values multiplied by their probabilities: $V(E) = \sum_j P(result_j(E))U(result_j(E))$

# Example - evaluation



Event 1 - Market Price:
$$V(E_1) = 1000 \times 0.5 - 300 \times 0.5 = 350.$$

Event 2 - Gold Price:
$$V(E_2) = 800 \times 0.7 + 100 \times 0.2 - 200 \times 0.1 = 560.$$

Decision 2 - Insurance: $V(D_2) = max(200, 560) = 560$ – No insurance.

Decision 1 - Investment: $V(D_1) = max(150, 560, 0) = 560$ – Invest in Gold.

Thus, in this case the best decisions are to invest in Gold without insurance.

# Influence Diagrams

- Influence Diagrams (IDs) are a tool for solving decision problems that were introduced by Howard and Matheson
- We can view IDs as an extension of Bayesian networks that incorporates decision and utility nodes

# Representation

- An influence diagram is a directed acyclic graph, $G$, with 3 types of nodes:

  **Random nodes ($X$):** represent random variables as in BNs, with an associated CPT. These are represented as ovals.

  **Decision nodes ($D$):** represent decisions to be made. The arcs pointing towards a decision node are *informational*; that is, it means that the random or decision node at the origin of the arc must be known before the decision is made. Represented as rectangles

  **Utility nodes ($U$):** represent the costs or utilities associated to the model. Associated to each utility node there is a function that maps each permutation of its parents to a utility value. Represented as diamonds.

# Representation

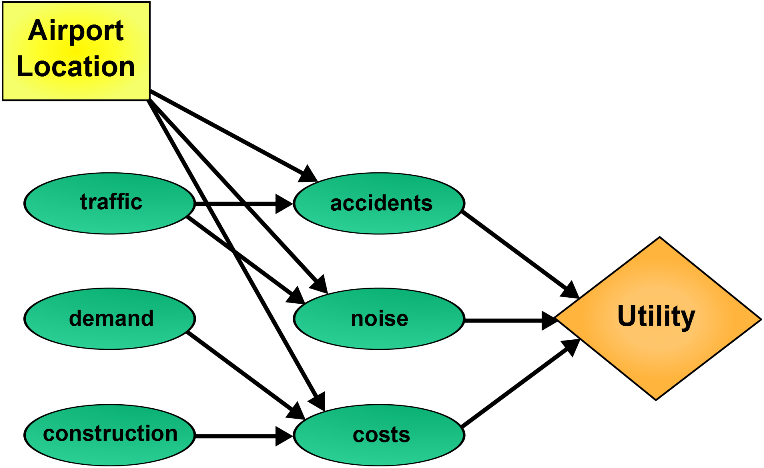
- Utility nodes can be divided into *ordinary* utility nodes, whose parents are random and/or decision nodes; and *super-value* utility nodes, whose parents are ordinary utility nodes. Usually the super-value utility node is the (weighted) sum of the ordinary utility nodes.
- There are three types of arcs in an ID:

  Probabilistic: they indicate probabilistic dependencies, pointing towards random nodes.

  Informational: they indicate information availability, pointing towards decision nodes. That is, $X \rightarrow D$ indicates that value of $X$ is known before the decision $D$ is taken.

  Functional: they indicate functional dependency, pointing towards utility nodes.

# ID - Example


Airport Location
traffic
accidents
demand
noise
Utility
construction
costs

# Order

- In an ID there must be a directed path in the underlying directed graph that includes all the decision nodes, indicating the order in which the decisions are made
- This order induces a partition on the random variables in the ID, such that if there are $n$ decision variables, the random variables are partitioned into $n + 1$ subsets
- Each subset, $\mathbf{R_i}$, contains all the random variables that are known before decision $D_i$

# Evaluation

- IDs are used to aid a decision maker in finding the decisions that maximize its expected utility
- The goal in decision analysis is to find an *optimal policy*, $\pi = \{d_1, d_2, ..., d_n\}$, which selects the best decisions for each decision node to maximize the expected utility, $E_\pi(U)$
- If there are several utility nodes, in general we consider that we have additive utility, so we will maximize the sum of these individual utilities:

$$E_\pi(U) = \sum_{u_i \in U} E_\pi(u_i) \quad (1)$$

# Simple ID

- *Simple* influence diagram has a single decision node and a single utility node
- We can simple apply BN inference techniques to obtain the optimal policy following this algorithm:
  1. For all $d_i \in D$:
     1. Set $D = d_i$.
     2. Instantiate all the known random variables.
     3. Propagate the probabilities as in a BN.
     4. Obtain the expected value of the utility node, $U$.
  2. Select the decision, $d_k$, that maximizes $U$.

# Complex IDs



- For more complex decision problems in which there are several decision nodes, the previous algorithm becomes impractical. In general, there are three main types of approaches for solving IDs:
  - Transform the ID to a decision tree and apply standard solution techniques for decision trees.
  - Solve the ID directly by variable elimination, applying a series of transformations to the graph.
  - Transform the ID to a Bayesian network and use BN inference techniques.

# Variable Elimination



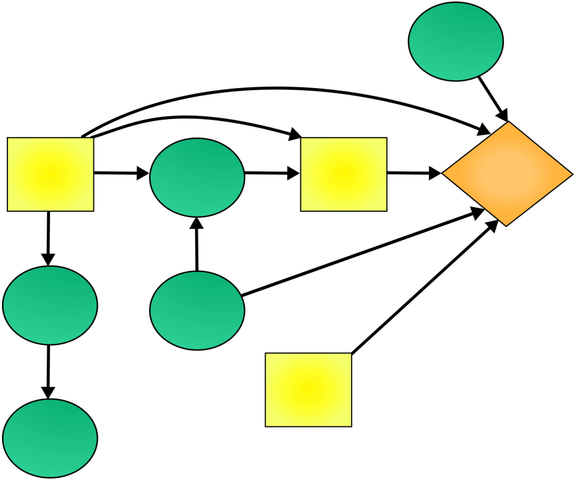
- VE is based on evaluating the decision nodes one by one according to a certain order
- Decision nodes that have been evaluated can be eliminated from the model, and this process continues until all the decision nodes have been evaluated
- To apply this technique the influence diagram must be *regular*; that is, it satisfies the following conditions:
    1. The structure of the ID is a directed acyclic graph.
    2. The utility nodes do not have successors.
    3. There is a directed path in the underlying directed graph that includes all the decision nodes, indicating the order in which the decisions are made.

# Transformations

- To evaluate the decision nodes, it is usually necessary to perform a series of transformations to the ID:
  - Eliminate *barren* nodes, random or decision nodes that are leaf nodes in the graph –they do not affect the decisions.
  - Eliminate random nodes that are parents of the utility node and do not have other children –the utility is updated according to the value of the node
  - Eliminate decision nodes that are parents of the utility node where their parents are also a parent to the utility node – evaluate the decision node and take the decision that maximizes the expected utility
  - In case none of the previous operations can be applied, invert an arc between two random variables. To invert an arc between nodes $i$ and $j$ it is required that there be no other trajectory between these nodes. Then the arc $i \rightarrow j$ is inverted and each node inherits the parents of the other node.

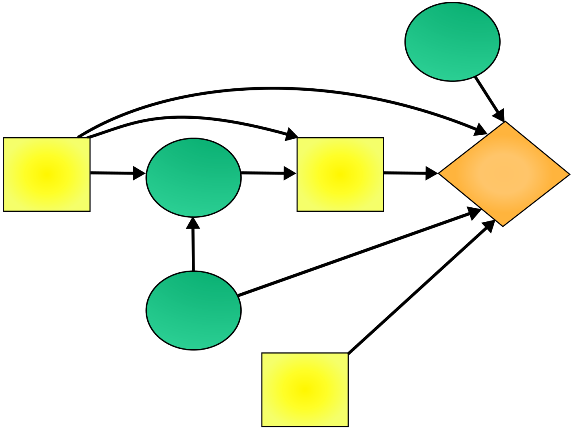
# VE - graphical example

- Initial model



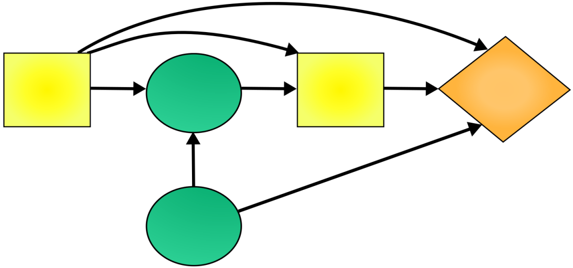
# VE - example

- Eliminate two barren nodes



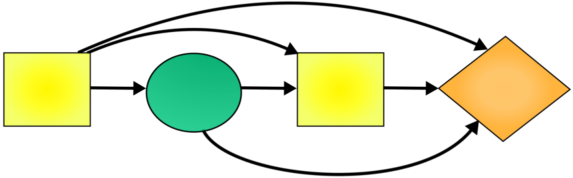
# VE - example

- Eliminate top random node
- Evaluate first decision (bottom)



# VE - example

- Invert arc
- Eliminate random node



# VE - example



- Evaluate second decision node
- Eliminate random variable
- Evaluate third decision node
- This concludes the example - the 3 decision nodes are evaluated and the optimal policy is defined

# Transformation to a BN

- To transform an ID to a BN, the basic idea is to transform decision and utility nodes to random nodes, with an associated probability distribution
- A decision node is converted to a discrete random variable by considering each decision, $d_i$, as a value for this variable, and using a uniform distribution as a CPT
- A utility node is transformed to a binary random variable by *normalizing* the utility function so it is in the range from 0 to 1, that is:

$$P(u_i = 1 \mid Pa(u_i)) = val(Pa(u_i))/maximum(val(Pa(u_i))) \quad (2)$$

# Inference



- After the previous transformation, and considering a single utility node, the problem of finding the optimal policy is reduced to finding the values of the decision nodes that maximize the probability of the utility node: $P(u = 1 \mid D, R)$
- This probability can be computed using standard inference techniques for BNs; however, it will require an exponential number of inference steps, one for each permutation of $D$

# Inference



- Given that in a *regular* ID the decision nodes are ordered, a more efficient evaluation can be done by evaluating the decisions in (inverse) order
- That is, instead of maximizing $P(u = 1 \mid D, R)$, we maximize $P(D_j \mid u = 1, R)$.
- We can recursively optimize each decision node, $D_j$, starting from the last decision, continuing with the previous decision, and so on, until we reach the first decision

# ID Limitations



- Traditional techniques for solving IDs make two important assumptions:

  Total ordering: all the decisions follow a total ordering according to a directed path in the graph.

  Non forgetting: all previous observations are remembered for future decisions.
- In some domains, such as in medical decision making, a total ordering of the decisions is an unrealistic assumption
- For a system that evolves over a large period of time, the number of observations grows linearly with the passing of time, so the non-forgetting requirement implies that the size of policies grows exponentially

# Limited memory influence diagrams

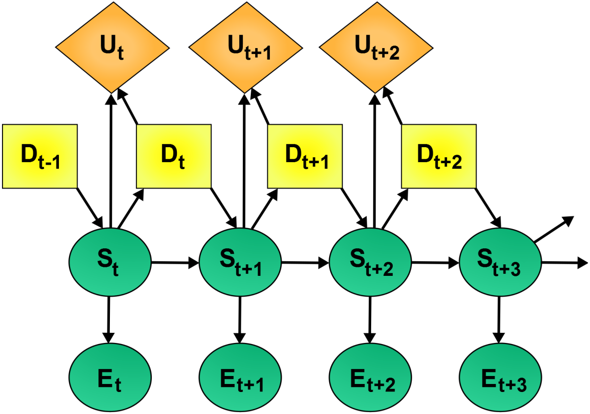


- Limited-memory influence diagrams (LIMIDs) are an extension of influence diagrams
- The term limited-memory reflects the property that a variable known when making a decision is not necessarily remembered when making a posterior decision
- Eliminating some variables reduces the complexity of the model so it is solvable with a computer, although at the price of obtaining a sub-optimal policy

# Dynamic Decision Networks

- Another extension is applied for sequential decision problems, that involve several decisions over time
- A sequential decision problem can be modeled as a *dynamic decision network* (DDN) –also known as a *dynamic influence diagram* which can be seen as an extension of a DBN, with additional decision and utility nodes for each time step
- In principle, we can evaluate a DDN in the same way as an ID, considering that the decisions have to be ordered in time
- However, as the number of time epochs increases, the complexity increases and can become computationally intractable
- DDNs are closely related to *Markov decision processes* which are the topic of the next chapter

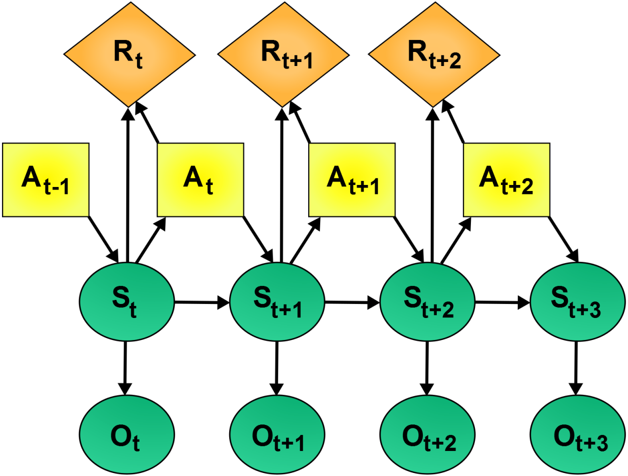
# DDN - example



# Decision-Theoretic Caregiver

- The objective of the *caregiver* is to guide a person in completing a task using an adequate selection of prompts, e.g. *cleaning one's hands*
- The system acts as a caregiver that guides an elderly or handicapped person in performing this task correctly
- A dynamic decision network (DDN) is used to model the user behavior and make the optimal decisions at each time step based on the user's behavior (observations) and the objectives of the system (utilities)

# Model



# Model



- **States.** The state space is characterized by the activities (hand gestures) carried out by a person. In this case, the state variable has 6 possible values: $s_1$ = *opening the faucet*, $s_2$ = *closing the faucet*, $s_3$ = *using the soap*, $s_4$= *drying the hands*, $s_5$ = *taking the towel* and $s_6$ = *washing the hands*.
- **Observations.** These correspond to the information obtained by a visual gesture recognition system that tries to recognize the activity performed by the person while washing their hands

# Model

- **Actions**. Actions are audible prompts to help the person complete the task. There are 8 actions that correspond to the possible prompts considered by the system: $a_1$ = *open the faucet*, $a_2$ = *close the faucet*, $a_3$ = *put soap on the hands*, $a_4$ = *wash the hands*, $a_5$ = *take the towel*, $a_6$ = *dry the hands*, $a_7$ = *Null* and $a_8$ = *call for help*
- **Rewards**. Rewards are associated with the preferences of the different actions selected by the system. In the caregiver setting, we must consider the prompts, clarity of the prompts and user's response. Three different reward values were used: $+3$ indicates a preference, $-3$ indicates a penalty, and $-6$ is used for selecting the action *call for help*. The idea is that asking for help should be the last option

# Evaluation

- The model was evaluated in terms of its: (i) sensitivity relative to the number of stages or lookahead, (ii) efficiency in terms of the time required to solve the model for selecting the next action, and (iii) performance, comparing the actions selected by the system with a human caregiver
- The expected utility increases as the lookahead is increased, tending to stabilize after 6 or 7 stages. However, the selected actions do not vary after a lookahead of 4, so this value was selected
- To evaluate the optimality (or near optimality) of the actions selected by the system, its decisions were compared to those of a human performing the same task

# Evaluation



- To evaluate the optimality (or near optimality) of the actions selected by the system, its decisions were compared to those of a human performing the same task
- A preliminary evaluation was done with normal persons simulating that they had problems washing their hands. Ten adults participated in the experiment, divided in two groups
- The first group was guided to complete the task of washing their hands by verbal prompts given by the system, the control group was guided by verbal instructions given by a human assistant

# Results

- The aspects evaluated in a questionnaire given to each participant were: (i) clarity of the prompt, (ii) detail of the prompt, and (iii) effectiveness of the system. The evaluation scale is from 1 (worst) to 5 (best)

| | Human | System |
|---|---|---|
| Clarity | 4.4 | 3.9 |
| Detail | 4.6 | 3.6 |
| Effectiveness | 4.2 | 3.6 |

- The results indicate a small advantage when the best prompt is selected by a human
- Two aspects show a small difference (0.6 or less), and one shows a more significant difference (detail of the prompt). This last aspect has to do with the verbal phrase recorded to be associated with each prompt, which could be easily improved

# Book

Sucar, L. E, *Probabilistic Graphical Models*, Springer 2015 – Chapter 10

# Additional Reading (1)

- Cole, S., Rowley, J.: Revisiting Decision Trees. Management Decision, 33(8), 46–50, (1995)
- Cooper, G.: A Method for Using Belief Networks as Influence Diagrams. In: Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence (UAI), 55–63 1988
- Howard, R., Matheson, J.: Influence Diagrams. In: R. Howard, J. Matheson (eds.), Readings on the Principles and Applications of Decision Analysis, Strategic Decisions Group, Menlo Park (1984)
- Lauritzen, S., Nilsson, D.: Representing and Solving Decision Problems with Limited Information. Management Science, 47, 1235–1251 (2001)

# Additional Reading (2)

- Montero, A., Sucar, L.E.: Decision-Theoretic Assistants Based on Contextual Gesture Recognition. Annals of Information Systems (to be published).
- Shachter, R. D.: Evaluating Influence Diagrams. Operations Research, 34(6), 871–882 (1986)
- van Gerven, M. A. J., Díez, F.J., Taal, B.G., Lucas, P.J.F.: Selecting Treatment Strategies with Dynamic Limited-Memory Influence Diagrams. Artificial Intelligence in Medicine , 40(3), 171–186 (2007)
- von Neumann, J., Morgenstern, O.: Theory of Games and Economic Behavior. Princeton University Press, Princeton, New Jersey (1944)