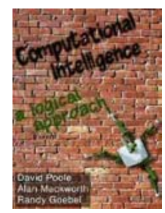
Chapter 10 (Part 2): Using Uncertain Knowledge

- Independence Assumptions



D. Poole, A. Mackworth, and R. Goebel, *Computational Intelligence: A Logical Approach*, Oxford University Press, January 1998

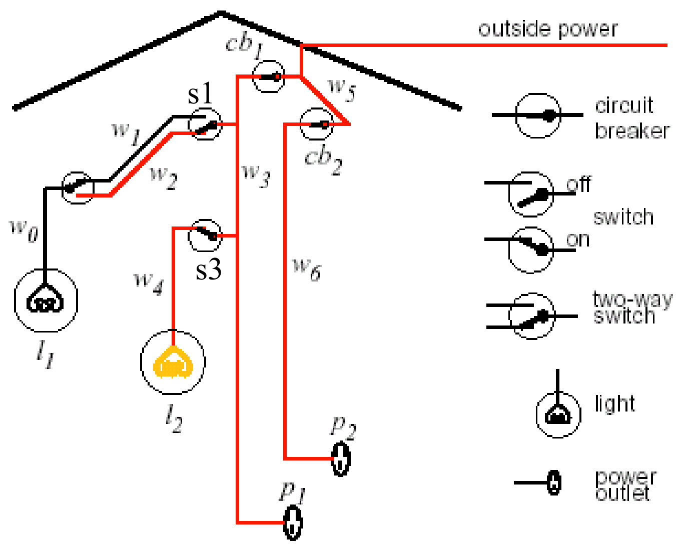
Independence Assumptions

- Conditional Independence
  - Random variable x is independent of random variable y given random variable z if, for all a_i, b_j and c_k,

$$P(x = a_i | y = b_j \wedge z = c_k) = P(x = a_i | z = c_k).$$

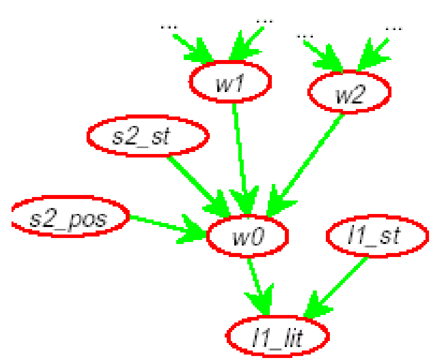
That is, knowledge of y’s value doesn’t affect your belief in the value of x, given a value of z.

Example Domain (Diagnostic Assistant)



Idea of Belief Networks

- Whether l_1 is lit (l_1_lit) depends only on the status (functioning or not) of the light (l_1_st) and whether there is power in wire w_0. Thus, l_1_lit is independent of the other variables given l_1_st and w_0. In a belief network, w_0 and l_1_st are parents of l_1_lit.



- Similarly, w_0 depends only on whether there is power in w_1, whether there is power in w_2, the position of switch s_2 (s_2_pos), and the status of switch s_2 (s_2_st).

Examples of Independence

- The identity of the president of this country is independent of whether light l_1 is lit given whether there is outside power.
- Whether there is someone in a room is independent of whether a light l_2 is lit given the position of switch s_3.
- Whether light l_1 is lit is independent of the position of light switch s_2 given whether there is power in wire w_0.
- Every other variable may be independent of whether light l_1 is lit given whether there is power in wire w_0 and the status of light l_1 (if it's ok, or if not, how it's broken).

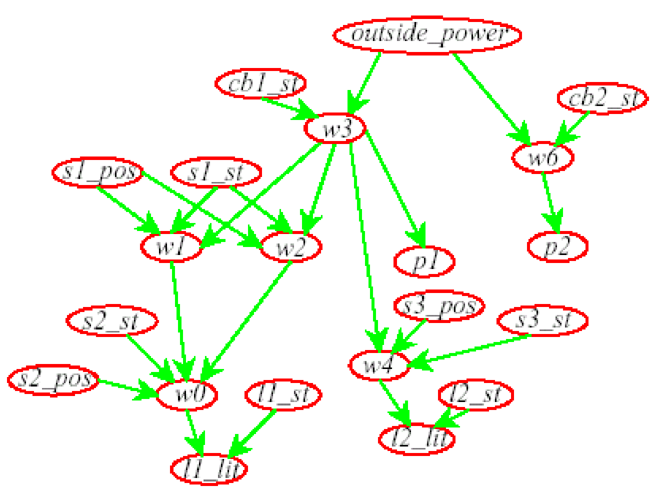
Belief Networks

- A belief network is a directed acyclic graph (DAG) with nodes denoting random variables.
- The parents of a node n are those variables on which n directly depends.
- A belief network is a graphical representation of dependence and independence:
- A variable is independent of its nondescendents given its parents.

Components of a belief network

- **A belief network consists of:**
  - **a directed acyclic graph with nodes labeled with random variables**
  - **a domain for each random variable**
  - **a set of conditional probability tables for each variable given its parents (including prior probabilities for nodes with no parents).**

Example Belief Network



Example belief network (continued)

- The belief network also specifies:
  - The domain of the variables:
    $w_0, \ldots, w_6$ have domain {live, dead}
    s_1_pos, s_2_pos, and s_3_pos have domain {up, down}
    s_1_st has {ok, upside_down, short, intermittent, broken}.
  - Conditional probabilities, including:
    $P(w_1 = live \mid s_1\_pos = up \wedge s_1\_st = ok \wedge w_3 = live)$
    $P(w_1 = live \mid s_1\_pos = up \wedge s_1\_st = ok \wedge w_3 = dead)$

    Priors:
    $P(s_1\_pos = up)$
    $P(s_1\_st = upside\_down)$

Alternate definition of belief network

- If $x_1, \ldots, x_n$ is a total ordering of the variables of interest,

$$P(x_1 = v_1 \wedge \ldots \wedge x_n = v_n)$$

$$= \prod_{i=1}^{n} p(x_i = v_i \mid x_{i-1} = v_{i-1} \wedge \ldots \wedge x_1 = v_1) \qquad (1)$$

$$= \prod_{i=1}^{n} p(x_i = v_i \mid \pi_{x_i} = \pi_{v_i}). \qquad (2)$$

- (1) is by the chain rule. (2) is a definition of the parents π_{xi} of x_i: those predecessors of x_i that render x_i independent of the other predecessors π_{vi} are their corresponding values.
- A graph constructed in this manner is automatically acyclic.

Constructing Belief Networks

- To represent a domain in a belief network, you need to consider:
  - What are the relevant variables?
  - What is the relationship between them? This should be expressed in terms of local influence.
  - What values should these variables take?
  - How does the value of one variable depend on the variables that locally influence it (its parents)? This is expressed in terms of the conditional probability tables.

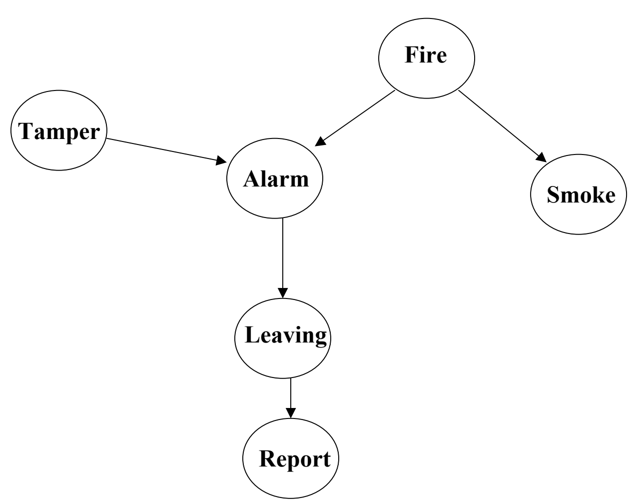
Using Belief Networks

- The power network can be used in a number of ways:
  - Conditioning on the status of the switches and circuit breakers, whether there is outside power and the position of the switches, you can simulate the lighting.
  - Given values for the switches, the outside power, and whether the lights are lit, you can determine the posterior probability that each switch or circuit breaker is ok or not.
  - Given some switch positions and some outputs and some intermediate values, you can determine the probability of any other variable in the network.

Belief Network Inference

- Three main approaches to determine posterior distributions in belief networks:
  - Exploiting the structure of the network to eliminate (sum out) the non-observed, non-query variables one at a time.
  - Search-based approaches that enumerate some of the possible worlds, and estimate posterior probabilities from the worlds generated.
  - Stochastic simulation where random cases are generated according to the probability distributions.

Example of Belief Network Inference



Example (Cont'ed)

- **Query (inference in a belief network):**
  P(ta | sm='yes' $\wedge$ re='yes') ?
  since:

  $P(z \mid y_1=\nu_1,\ldots,y_j=\nu_j) =$

  $P(z,y_1=\nu_1,\ldots,y_j=\nu_j) \,/\, (\sum_z P(z, y_1=\nu_1,\ldots,y_j=\nu_j))$

  We first need to compute:

  $P(ta \wedge sm=\text{'yes'} \wedge re=\text{'yes'}) = \sum_{le} \sum_{al} \sum_{fi} P(ta,fi,sm,al,le,re)=$

  $\sum_{le} \sum_{al} \sum_{fi} P(ta).P(fi).P(sm=\text{'yes'}|fi).P(al \mid ta,fi).P(le \mid al).P(re \mid le).$

Example (Cont'ed)

- **First sum out fi:**

$\sum_{fi} P(ta).P(fi).P(sm=\text{'yes'}|fi).P(al \mid ta,fi).P(le \mid al).P(re \mid le)$
$=P(ta).P(le \mid al).P(re=\text{'yes'} \mid le).f_1(al,ta).$

- **Sum out al:**

$\sum_{al} P(ta).P(le \mid al).P(re=\text{'yes'} \mid le).f_1(al,ta)$
$= P(ta).P(re=\text{'yes'} \mid le).f_2(le,ta)$

- **Sum out le:**

$\sum_{le} P(ta).P(re=\text{'yes'} \mid le).f_2(le,ta)$
$=P(ta).f_3(ta).$

- **P(ta | sm='yes' $\wedge$ re='yes')=** $P(ta).f_3(ta) \,/\, P(sm=\text{'yes'} \wedge re=\text{'yes'})$