function DECISION-THEORETIC-AGENT($\boldsymbol{E}_t$) **returns** an action
  inputs: $\boldsymbol{E}_t$, the percept at time t
  static: BN, a belief network with nodes $\mathbf{X}$
    $Bel(\boldsymbol{X})$, a vector of probabilities, updated over time

  $\widehat{Bel}(\mathbf{X}_t) \leftarrow \sum_{\mathbf{X}_{t-1}} \mathbf{P}(\mathbf{X}_t \mid \mathbf{X}_{t-1}{=}\mathbf{x}_{t-1}, A_{t-1})\, Bel(\mathbf{X}_{t-1}{=}\mathbf{x}_{t-1})$
  $Bel(\mathbf{X}_t) \leftarrow \alpha\, \mathbf{P}(\mathbf{E}_t \mid \mathbf{X}_t)\, \widehat{Bel}(\mathbf{X}_t)$
  $action \leftarrow \arg\max_{A_t} \sum_{\mathbf{X}_t} \left[Bel(\mathbf{X}_t{=}\mathbf{x}_t) \sum_{\mathbf{X}_{t+1}} P(\mathbf{X}_{t+1} = \mathbf{x}_{t+1} \mid \mathbf{X}_t{=}\mathbf{x}_t, A_t)\, U(\mathbf{x}_{t+1}) \right]$
  return *action*

Figure 17.9 Detailed design for a decision-theoretic agent.

where **E** and **X** are random variables ranging over percepts and states, respectively. What this means is that given a state of the world, the chances of the sensor giving a certain reading will be the same today as it was yesterday. This does *not* mean, for example, that the sensor can never break; it just means that we have to include in **X** all the variables that are important to the sensor's performance. The advantage of the stationary sensor model is that the fixed model $\mathbf{P}(\mathbf{E}|\mathbf{X})$ can then be used at each time step.

The basic idea of a sensor model can be implemented very simply in a belief network, because the assumption embodied in Equation (17.6) effectively isolates the sensor variables for a given time step from the rest of the network. Figure 17.10(a) shows an abstract belief network fragment with generalized state and sensor variables. The sensor model itself is the conditional probability table associated with the percept node. The direction of the arrow is the crucial element here: the state of the world *causes* the sensor to take on a particular value.[4] The sensor model is therefore an example of the general principle, stated in Chapter 7 and again in Chapter 15, that causal models are to be preferred when possible. If the sensor gives a perfect report of the actual state, then the sensor model—the conditional probability table—will be purely deterministic. Noise and errors in the sensor are reflected in the probabilities of "incorrect" readings. In fact, we have already seen examples of sensor models of this kind: in the burglar-alarm network (Figure 15.1), both *JohnCalls* and *MaryCalls* can be viewed as sensor nodes for the *Alarm* state variable. Their conditional probability tables, shown in Figure 15.2, show how reliable they are as sensors.

The next step is to break apart the generalized state and sensor variables into their components. Typically, each sensor only measures some small aspect of the total state. An example is shown in Figure 17.10(b), where temperature and pressure gauges measure the actual temperature and pressure of some system. Decomposing the overall sensor model in Figure 17.10(a) into its separate components greatly reduces the size of the CPTs, unless the sensors are very badly designed. As an example of how not to couple sensor nodes to state nodes: consider two sensors

[4] Of course, the process of *inference* will go the other way: evidence arrives at the sensor node, and is propagated to the state variable. The inference process essentially *inverts* the sensor model. This is basically what makes a Kalman filter simple. If $P(E|X)$ is Gaussian, then $P(X|E)$ is also Gaussian with an identical distribution, so that inversion is trivial.