

Inference in Belief Trees

Inference in Belief Trees

A Bayesian Network is a complete model for the variables and their relationships.

It can be used to answer queries about them.

Typical question:

Given observed variables, what is the updated knowledge about the other variables.

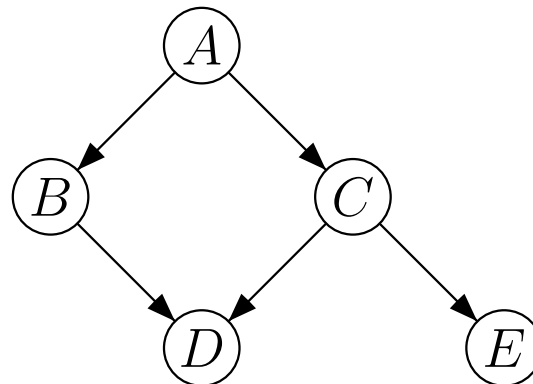
There are exact inference methods for this task.

Motivation

Choice of universe of discourse

	Variable	Domain	
A	metastatic cancer	$\{a_1, a_2\}$	$(\cdot_1 \text{ — present, } \cdot_2 \text{ — absent})$ $\Omega = \{a_1, a_2\} \times \dots \times \{e_1, e_2\}$ $ \Omega = 32$
B	increased serum calcium	$\{b_1, b_2\}$	
C	brain tumor	$\{c_1, c_2\}$	
D	coma	$\{d_1, d_2\}$	
E	headache	$\{e_1, e_2\}$	

Analysis of dependencies



Motivation

$$\left. \begin{array}{l} P(e_1 | c_1) = 0.8 \\ P(e_1 | c_2) = 0.6 \end{array} \right\} \text{headaches common, but more common if tumor present}$$

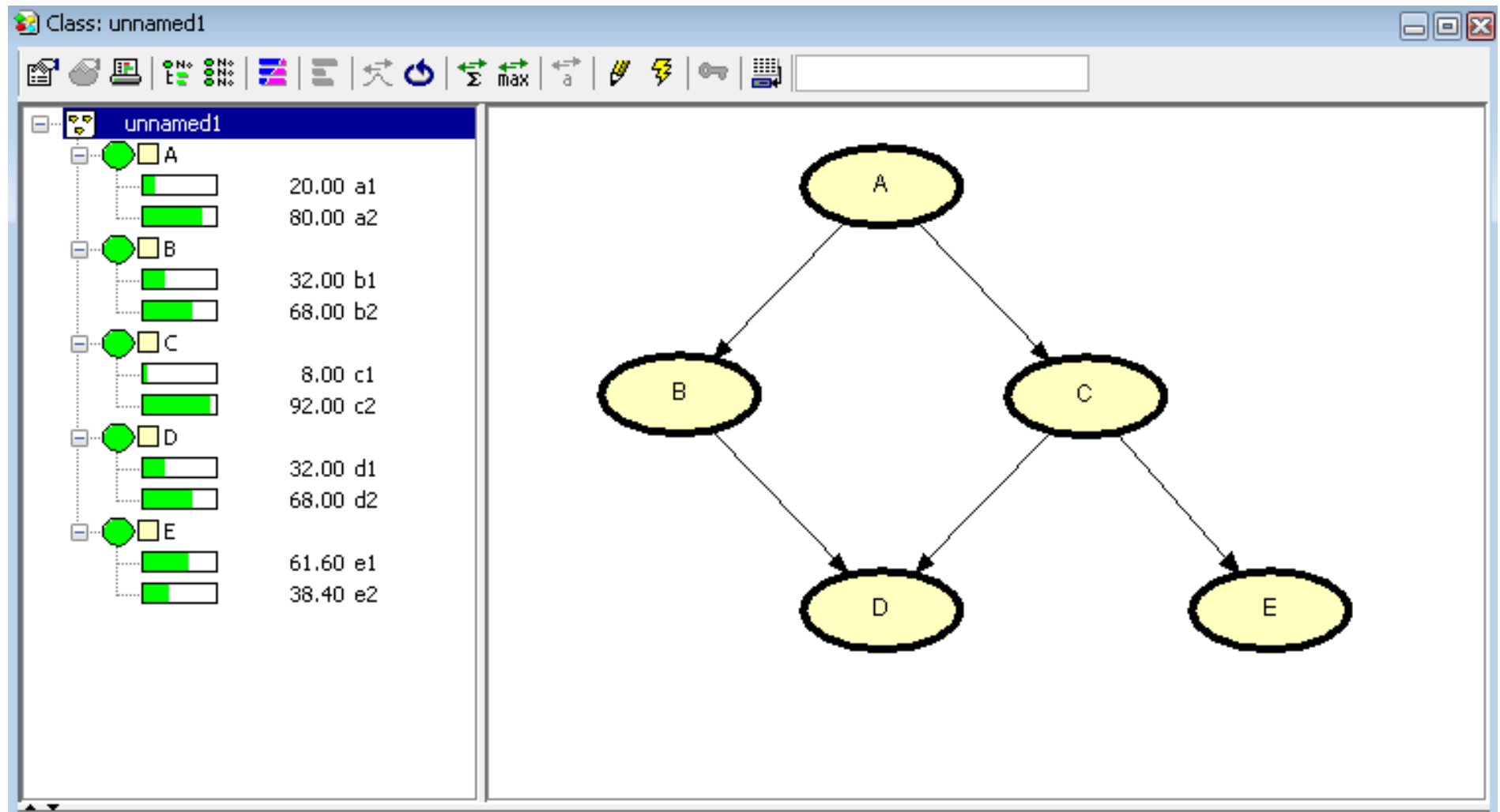
$$\left. \begin{array}{l} P(d_1 | b_1, c_1) = 0.8 \\ P(d_1 | b_1, c_2) = 0.8 \\ P(d_1 | b_2, c_1) = 0.8 \\ P(d_1 | b_2, c_2) = 0.05 \end{array} \right\} \text{coma rare but common, if either cause is present}$$

$$\left. \begin{array}{l} P(b_1 | a_1) = 0.8 \\ P(b_1 | a_2) = 0.2 \end{array} \right\} \begin{array}{l} \text{increased calcium uncommon,} \\ \text{but common consequence of metastases} \end{array}$$

$$\left. \begin{array}{l} P(c_1 | a_1) = 0.2 \\ P(c_1 | a_2) = 0.05 \end{array} \right\} \text{brain tumor rare, and uncommon consequence of metastases}$$

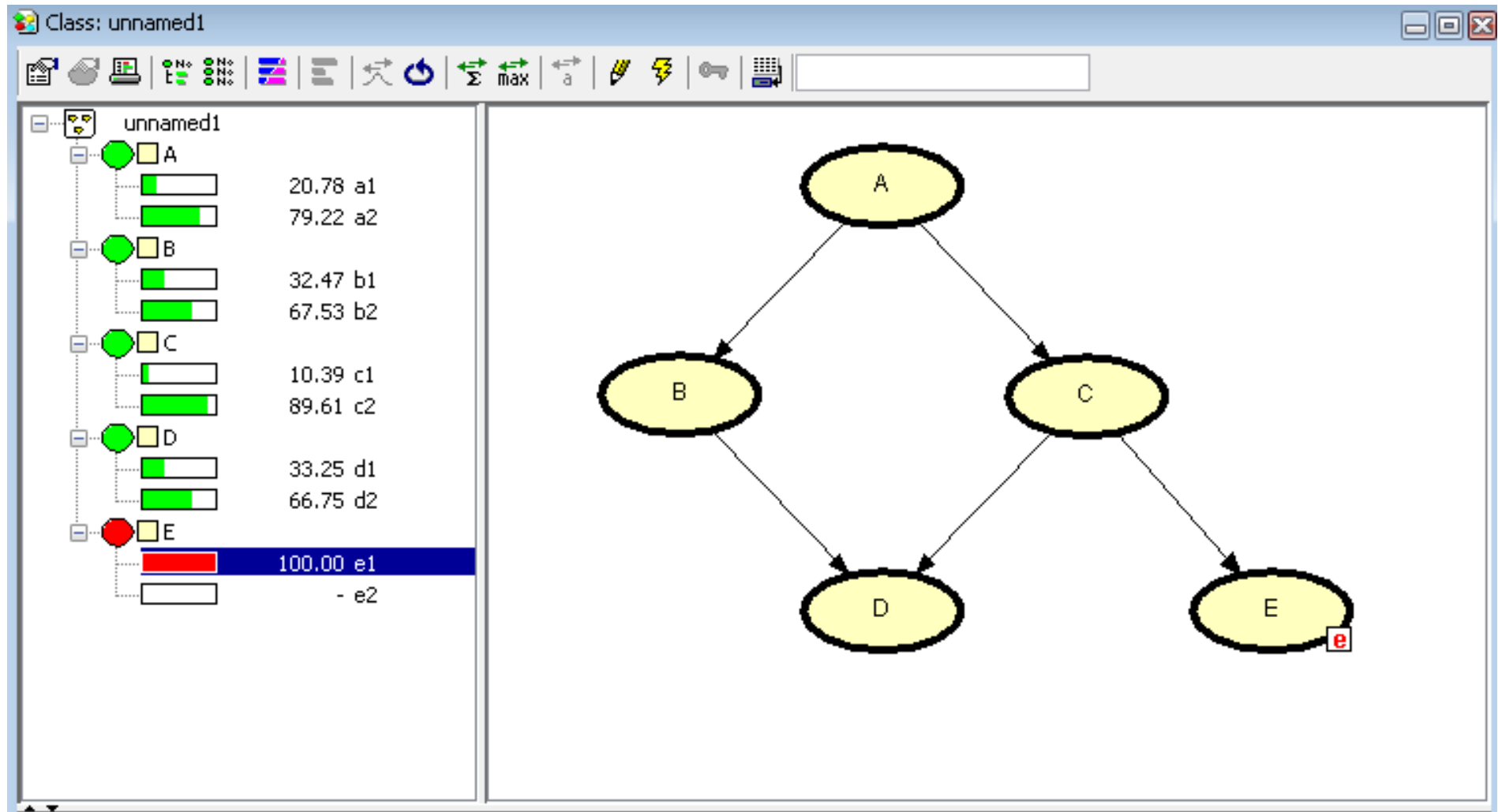
$$P(a_1) = 0.2 \quad \left. \right\} \text{incidence of metastatic cancer in relevant clinic}$$

Motivation



Marginal distributions in the HUGIN tool.

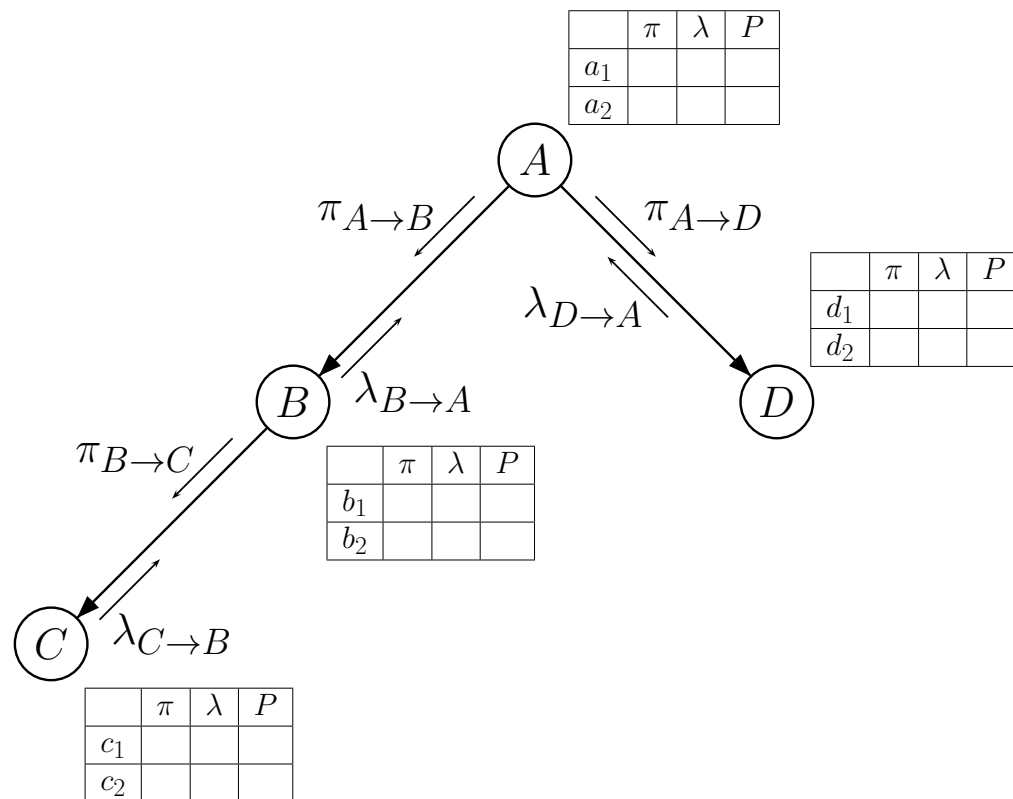
Motivation



Conditional marginal distributions with evidence $E = e_1$

Propagation in Belief Trees

Belief Tree:



Parameters:

$$P(a_1) = 0.1 \quad P(b_1 | a_1) = 0.7$$

$$P(b_1 | a_2) = 0.2$$

$$P(d_1 | a_1) = 0.8 \quad P(c_1 | b_1) = 0.4$$

$$P(d_1 | a_2) = 0.4 \quad P(c_1 | b_2) = 0.001$$

Desired:

$$\forall X \forall Y \in \{A, B, C, D\} : P(X | Y) = ?$$

Objective

Given: Belief network (V, E, P) with tree structure and $P(V) > 0$.
Set $W \subseteq V$ of instantiated variables where
a priori knowledge $W \neq \emptyset$ is allowed

Desired: $P(B \mid W)$ for all $B \in V$

Notation: W_B^- subset of those variables of W that belong
to the subtree of (V, E) that has root B

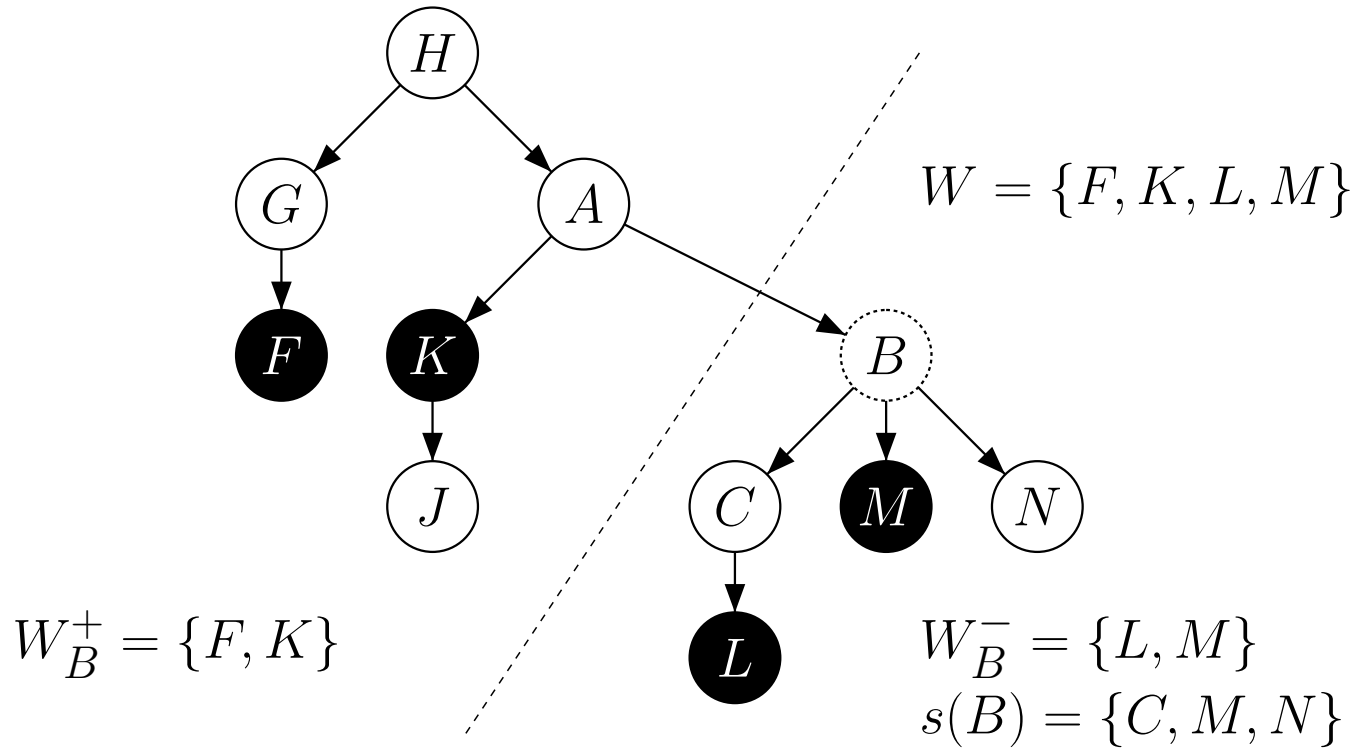
$$W_B^+ = W \setminus W_B^-$$

$s(B)$ set of direct successors of B

Ω_B domain of B

b^* value that B is instantiated with

Example



Decomposition in the Tree

$$\begin{aligned}P(B = b \mid W) &= P(b \mid W_B^- \cup W_B^+) \quad \text{with } B \notin W \\&= \frac{P(W_B^- \cup W_B^+ \cup \{b\})}{P(W_B^- \cup W_B^+)} \\&= \frac{P(W_B^- \cup W_B^+ \mid b)P(b)}{P(W_B^- \cup W_B^+)} \\&= \frac{P(W_B^- \mid b)P(W_B^+ \mid b)P(b)}{P(W_B^- \cup W_B^+)} \\&= \beta_{B,W} \underbrace{P(W_B^- \mid b)}_{\text{Evidence from "below"}} \underbrace{P(b \mid W_B^+)}_{\text{Evidence from "above"}}$$

π - and λ -Values

Since we ignore the constant $\beta_{B,W}$ for the derivations below, the following designations are used instead of $P(\cdot)$:

π -values and λ -values

Let $B \in V$ be a variable and $b \in \Omega_B$ a value of its domain. We define the π - and λ -values as follows:

$$\lambda(b) = \begin{cases} P(W_B^- | b) & \text{if } B \notin W \\ 1 & \text{if } B \in W \wedge b^* = b \\ 0 & \text{if } B \in W \wedge b^* \neq b \end{cases}$$

$$\pi(b) = P(b | W_B^+)$$

π - and λ -Values

$$\lambda(b) = \prod_{C \in s(B)} P(W_C^- | b) \quad \text{if } B \notin W$$

$$\lambda(b) = 1 \quad \text{if } B \text{ leaf in } (V, E)$$

$$\pi(b) = P(b) \quad \text{if } B \text{ root in } (V, E)$$

$$P(b | W) = \alpha_{B,W} \cdot \lambda(b) \cdot \pi(b)$$

λ -message

Let $B \in V$ be an attribute and $C \in s(B)$ its direct children with the respective domains $\text{dom}(B) = \{B_1, \dots, b_i, \dots, b_k\}$ and $\text{dom}(C) = \{c_1, \dots, c_j, \dots, c_m\}$.

$$\lambda_{C \rightarrow B}(b_i) \stackrel{\text{Def}}{=} \sum_{j=1}^m P(c_j | b_i) \cdot \lambda(c_j), \quad i = 1, \dots, k$$

The vector

$$\vec{\lambda}_{C \rightarrow B} \stackrel{\text{Def}}{=} \left(\lambda_{C \rightarrow B}(b_i) \right)_{i=1}^k$$

is called λ -message from C to B .

λ -Message

Let $B \in V$ an attribute and $b \in \text{dom}(B)$ a value of its domain.

Then

$$\lambda(b) = \begin{cases} \rho_{B,W} \cdot \prod_{C \in s(B)} \lambda_{C \rightarrow B}(b) & \text{if } B \notin W \\ 1 & \text{if } B \in W \wedge b = b^* \\ 0 & \text{if } B \in W \wedge b \neq b^* \end{cases}$$

with $\rho_{B,W}$ being a positive constant.

π -message

Let $B \in V$ be a non-root node in (V, E) and $A \in V$ its parent with domain $\text{dom}(A) = \{a_1, \dots, a_j, \dots, a_m\}$.

$j = 1, \dots, m :$

$$\pi_{A \rightarrow B}(a_j) \stackrel{\text{Def}}{=} \begin{cases} \pi(a_j) \cdot \prod_{C \in s(A) \setminus \{B\}} \lambda_{C \rightarrow B}(a_j) & \text{if } A \notin W \\ 1 & \text{if } A \in W \wedge a = a^* \\ 0 & \text{if } A \in W \wedge a \neq a^* \end{cases}$$

The vector

$$\vec{\pi}_{A \rightarrow B} \stackrel{\text{Def}}{=} \left(\pi_{A \rightarrow B}(a_j) \right)_{j=1}^m$$

is called π -message from A to B .

π -Message

Let $B \in V$ be a non-root node in (V, E) and A the parent node of B . Further let $b \in \text{dom}(B)$ be a value of B 's domain.

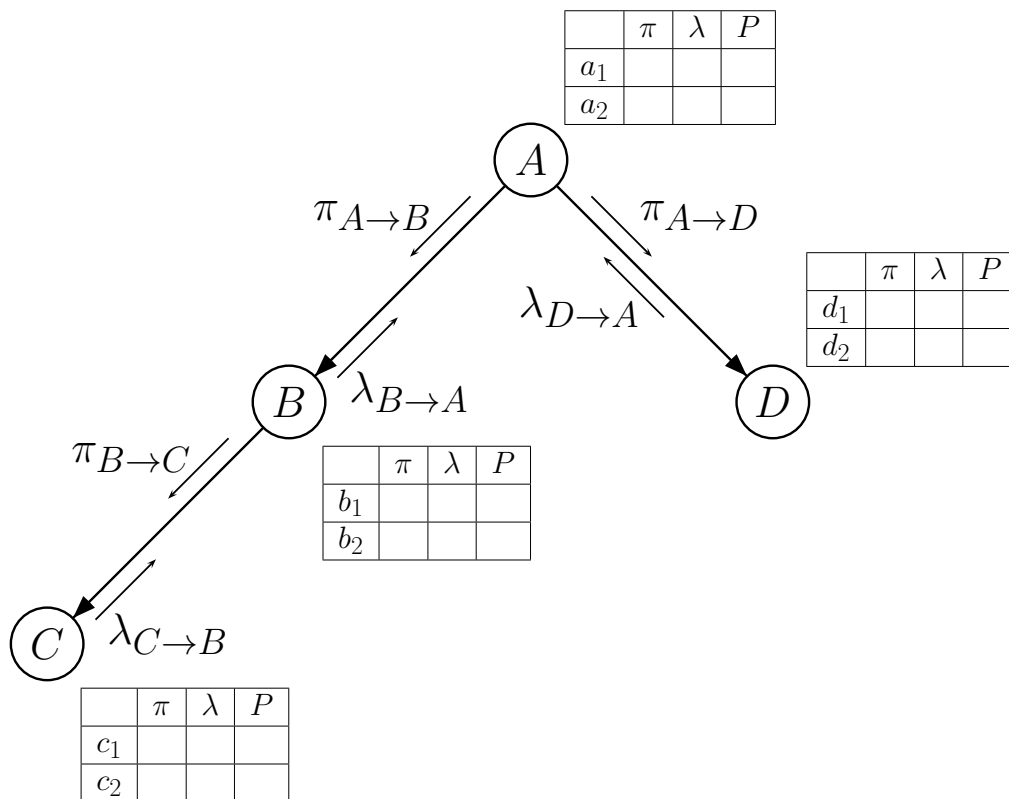
$$\pi(b) = \mu_{B,W} \cdot \sum_{a \in \text{dom}(A)} P(b \mid a) \cdot \pi_{A \rightarrow B}(a)$$

Let $A \notin W$ a non-instantiated attribute and $P(V) > 0$.

$$\begin{aligned} \pi_{A \rightarrow B}(a_j) &= \pi(a_j) \cdot \prod_{C \in s(A) \setminus \{B\}} \lambda_{C \rightarrow A}(a_j) \\ &= \tau_{B,W} \cdot \frac{P(a_j \mid W)}{\lambda_{B \rightarrow A}(a_j)} \end{aligned}$$

Propagation in Belief Trees

Belief Tree:



Parameters:

$$P(a_1) = 0.1 \quad P(b_1 | a_1) = 0.7$$

$$P(b_1 | a_2) = 0.2$$

$$P(d_1 | a_1) = 0.8 \quad P(c_1 | b_1) = 0.4$$

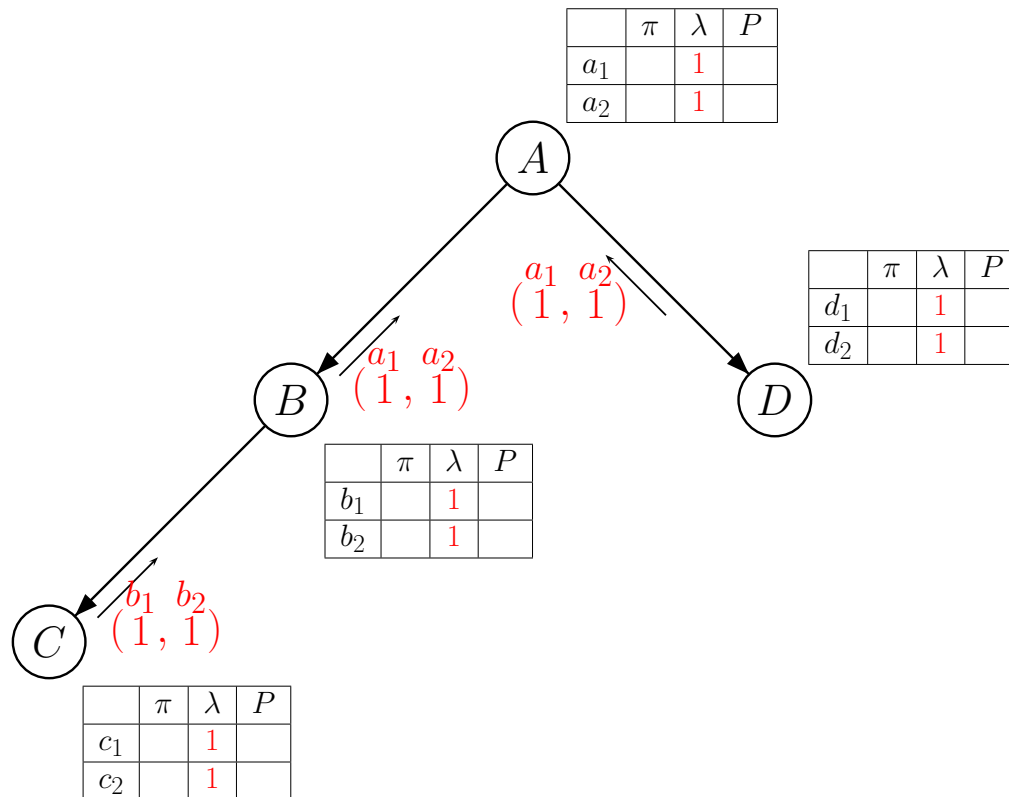
$$P(d_1 | a_2) = 0.4 \quad P(c_1 | b_2) = 0.001$$

Desired:

$$\forall X \in \{A, B, C, D\} : P(X | \emptyset) = ?$$

Propagation in Belief Trees (2)

Belief Tree:

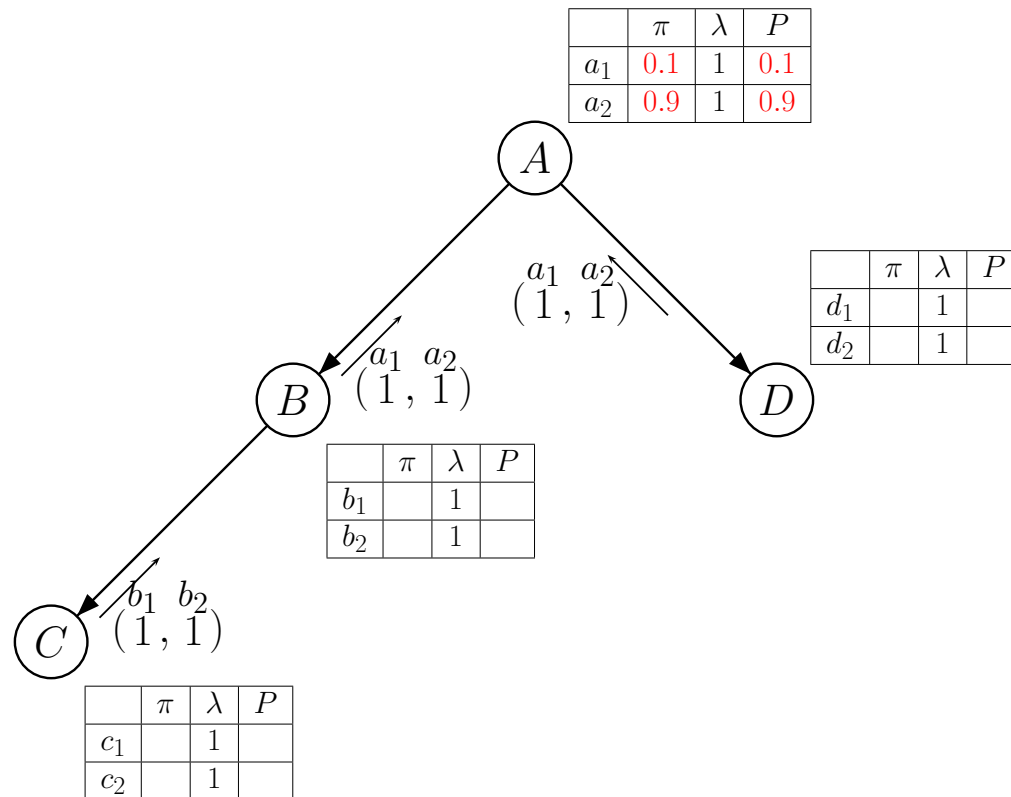


Initialization Phase:

Set all λ -messages and λ -values to 1.

Propagation in Belief Trees (3)

Belief Tree:



Initialization Phase:

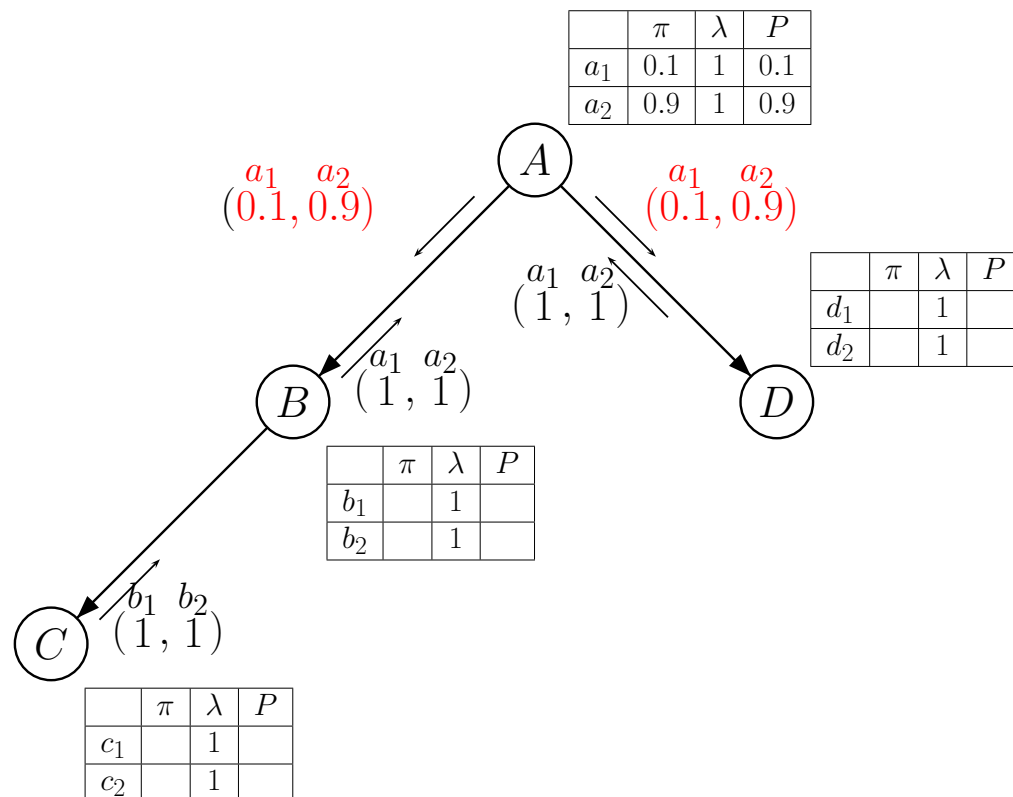
Set all λ -messages and λ -values to 1.

$\pi(a_1) = P(a_1)$ and

$\pi(a_2) = P(a_2)$

Propagation in Belief Trees (4)

Belief Tree:



Initialization Phase:

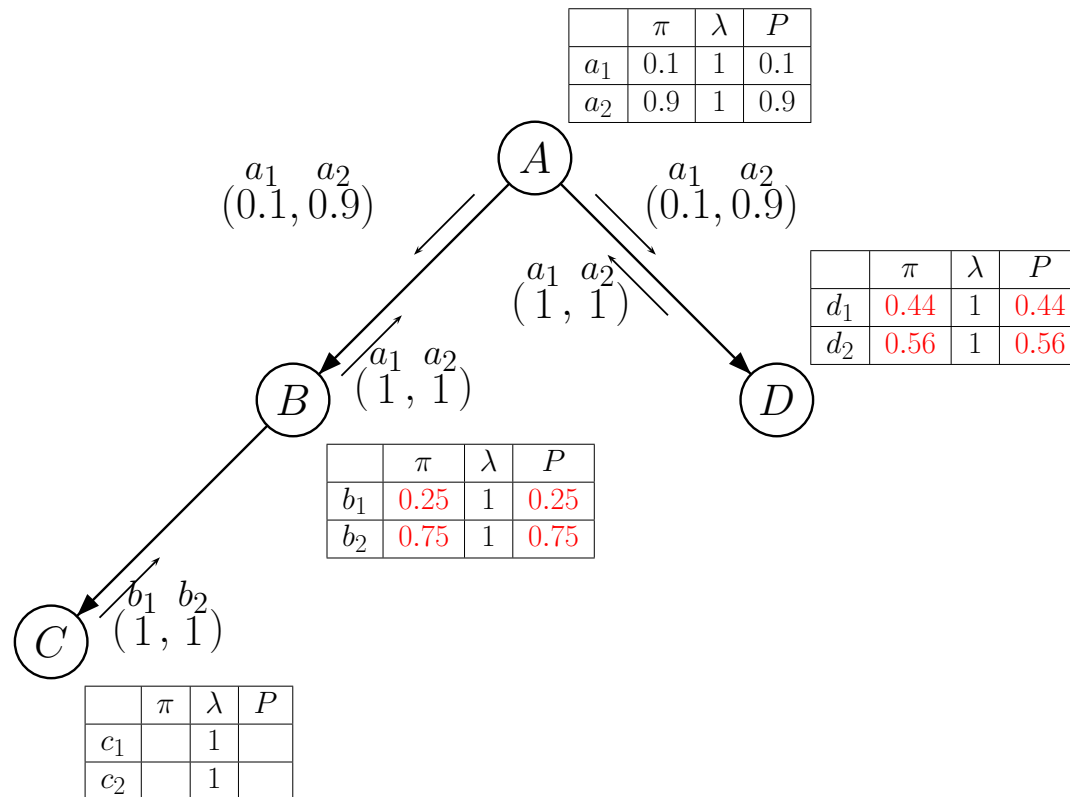
Set all λ -messages and λ -values to 1.

$\pi(a_1) = P(a_1)$ and
 $\pi(a_2) = P(a_2)$.

A sends π -messages to B and D.

Propagation in Belief Trees (5)

Belief Tree:



Initialization Phase:

Set all λ -messages and λ -values to 1.

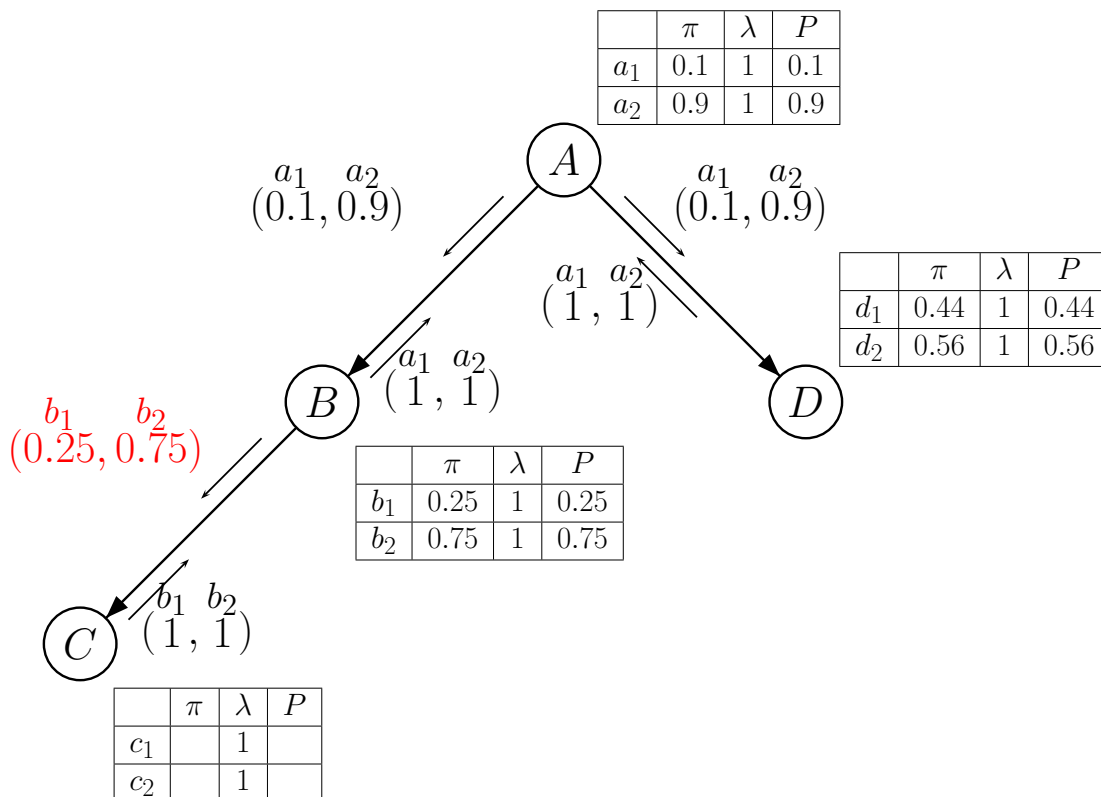
$\pi(a_1) = P(a_1)$ and
 $\pi(a_2) = P(a_2)$.

A sends π -messages to B and D.

B and D update their π -values.

Propagation in Belief Trees (6)

Belief Tree:



Initialization Phase:

Set all λ -messages and λ -values to 1.

$\pi(a_1) = P(a_1)$ and
 $\pi(a_2) = P(a_2)$.

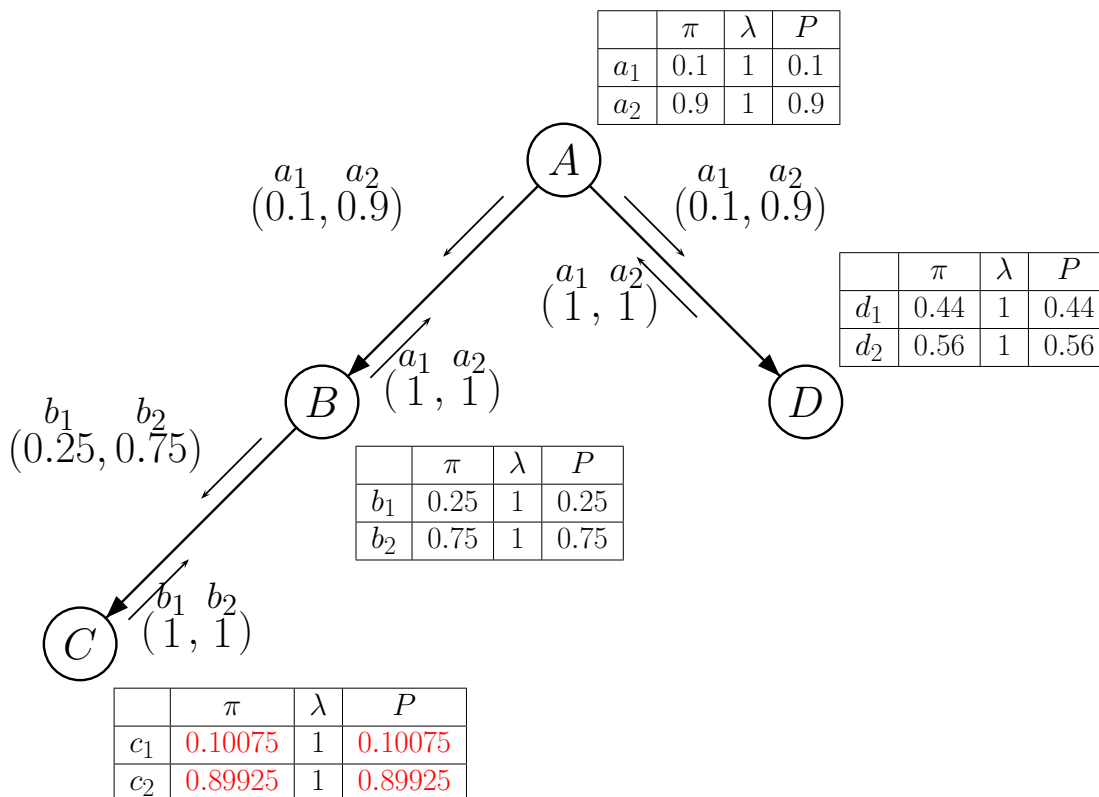
A sends π -messages to B and D.

B and D update their π -values.

B sends π -message to C.

Propagation in Belief Trees (7)

Belief Tree:



Initialization Phase:

Set all λ -messages and λ -values to 1.

$\pi(a_1) = P(a_1)$ and
 $\pi(a_2) = P(a_2)$.

A sends π -messages to B and D.

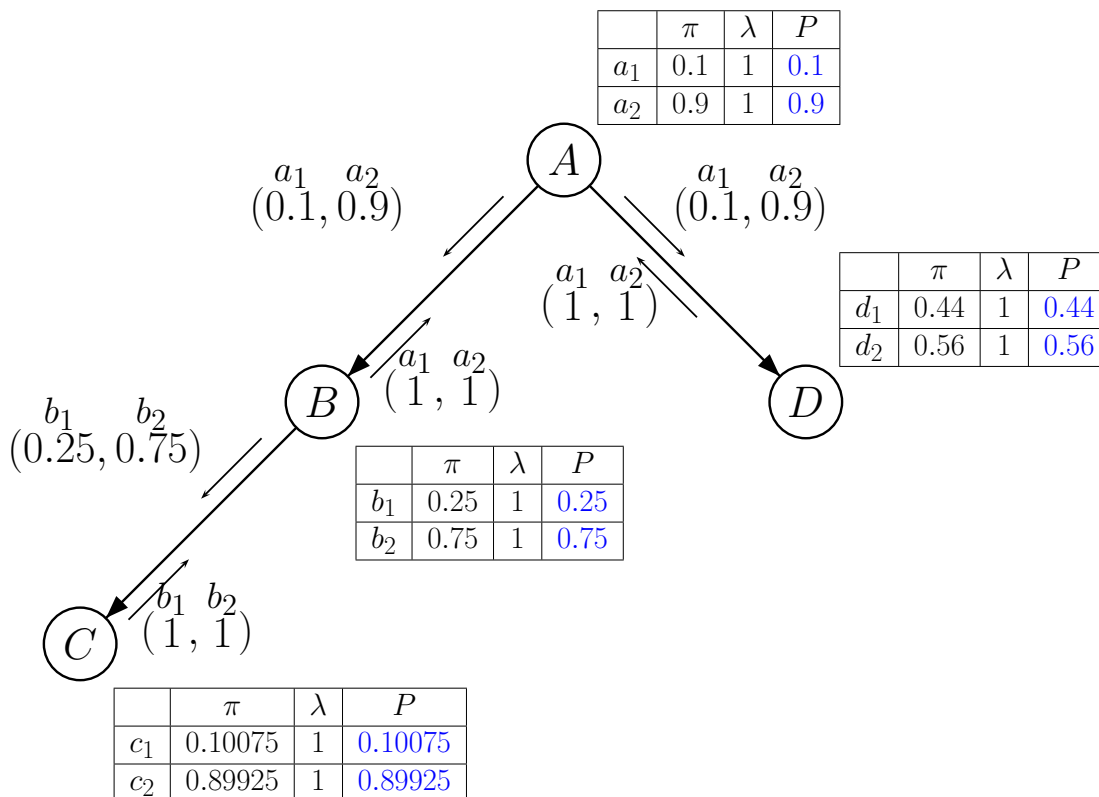
B and D update their π -values.

B sends π -message to C.

C updates its π -value.

Propagation in Belief Trees (8)

Belief Tree:



Initialization Phase:

Set all λ -messages and λ -values to 1.

$\pi(a_1) = P(a_1)$ and
 $\pi(a_2) = P(a_2)$.

A sends π -messages to B and D.

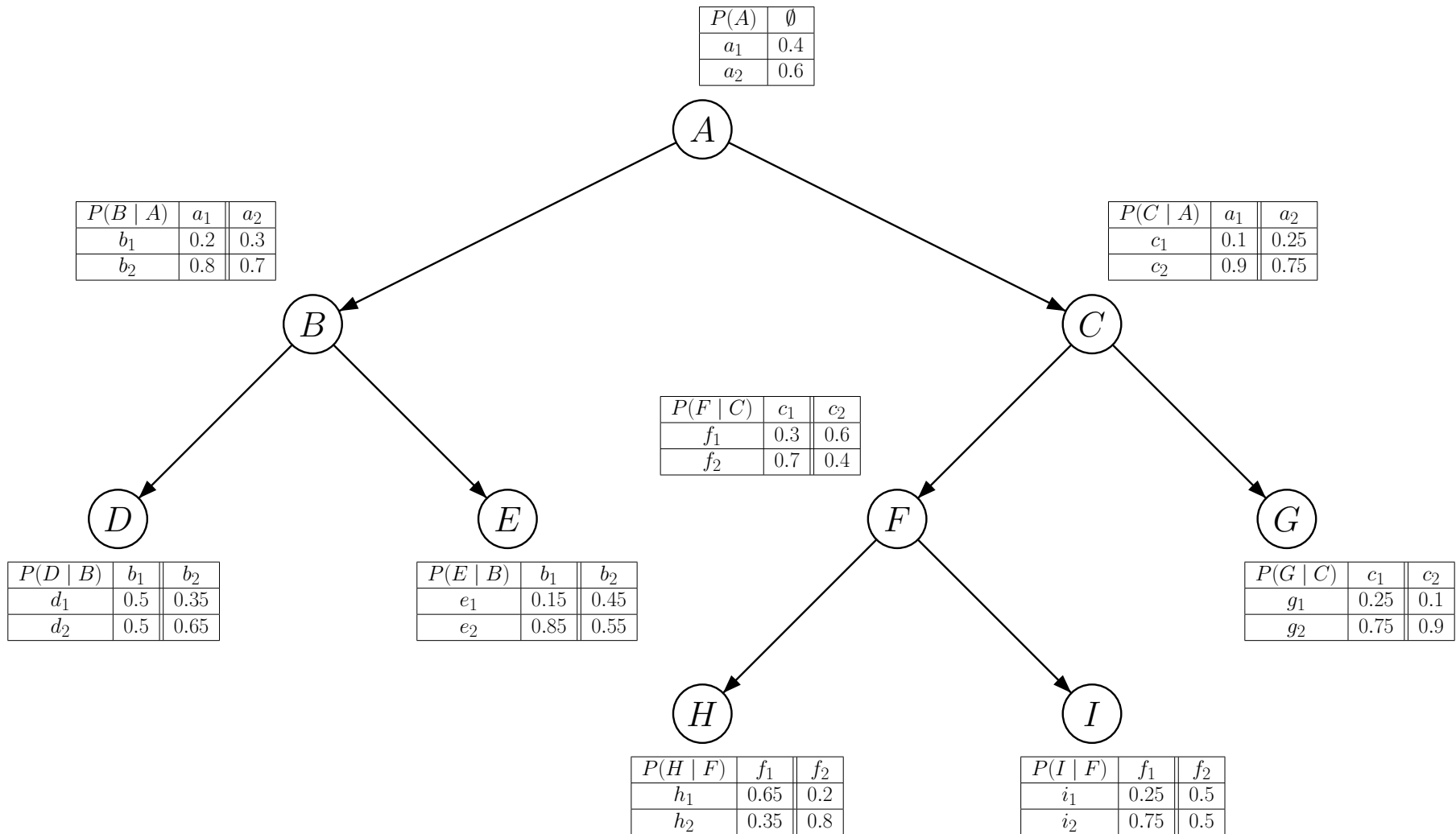
B and D update their π -values.

B sends π -message to C.

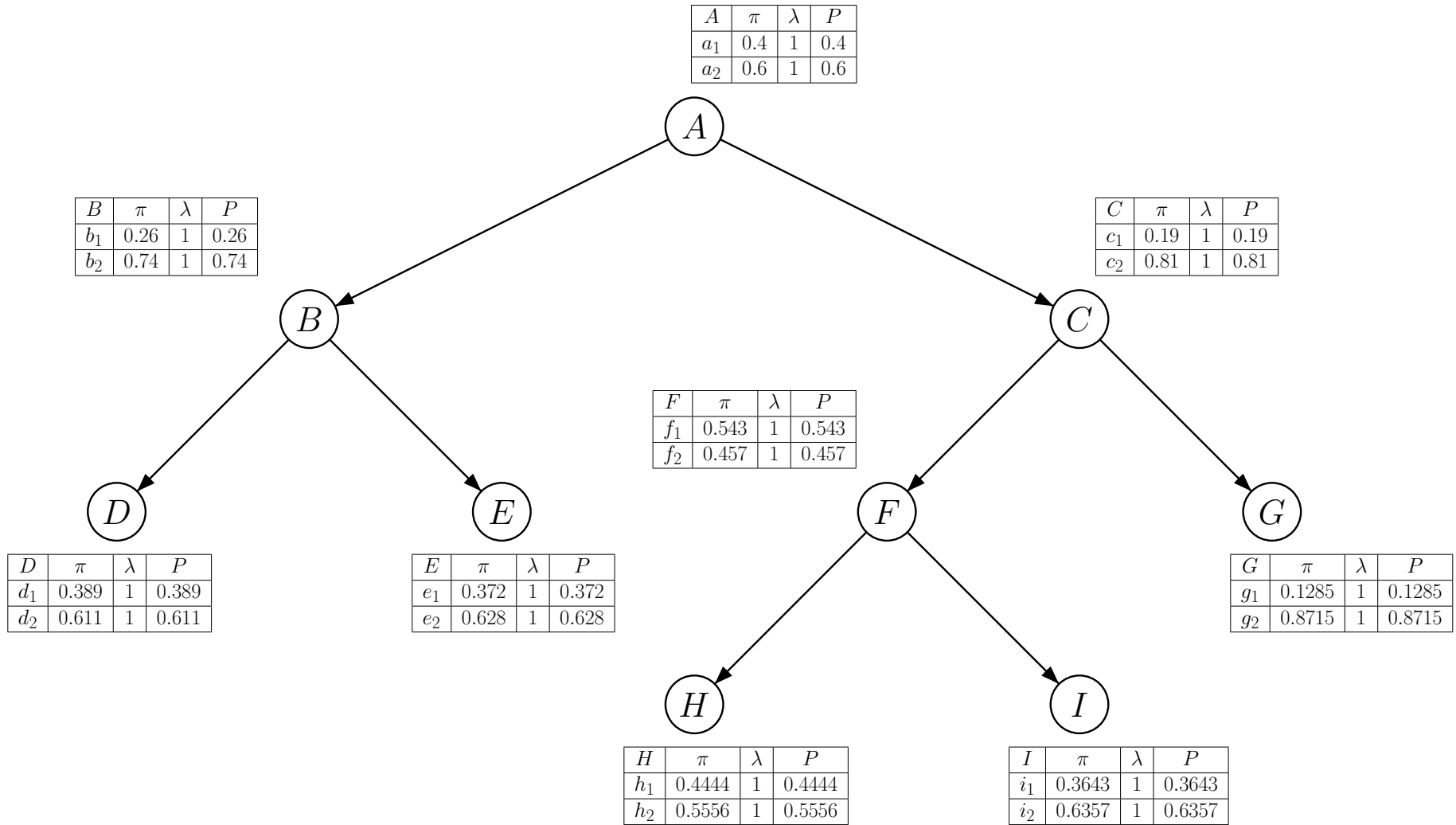
C updates its π -value.

Initialization finished.

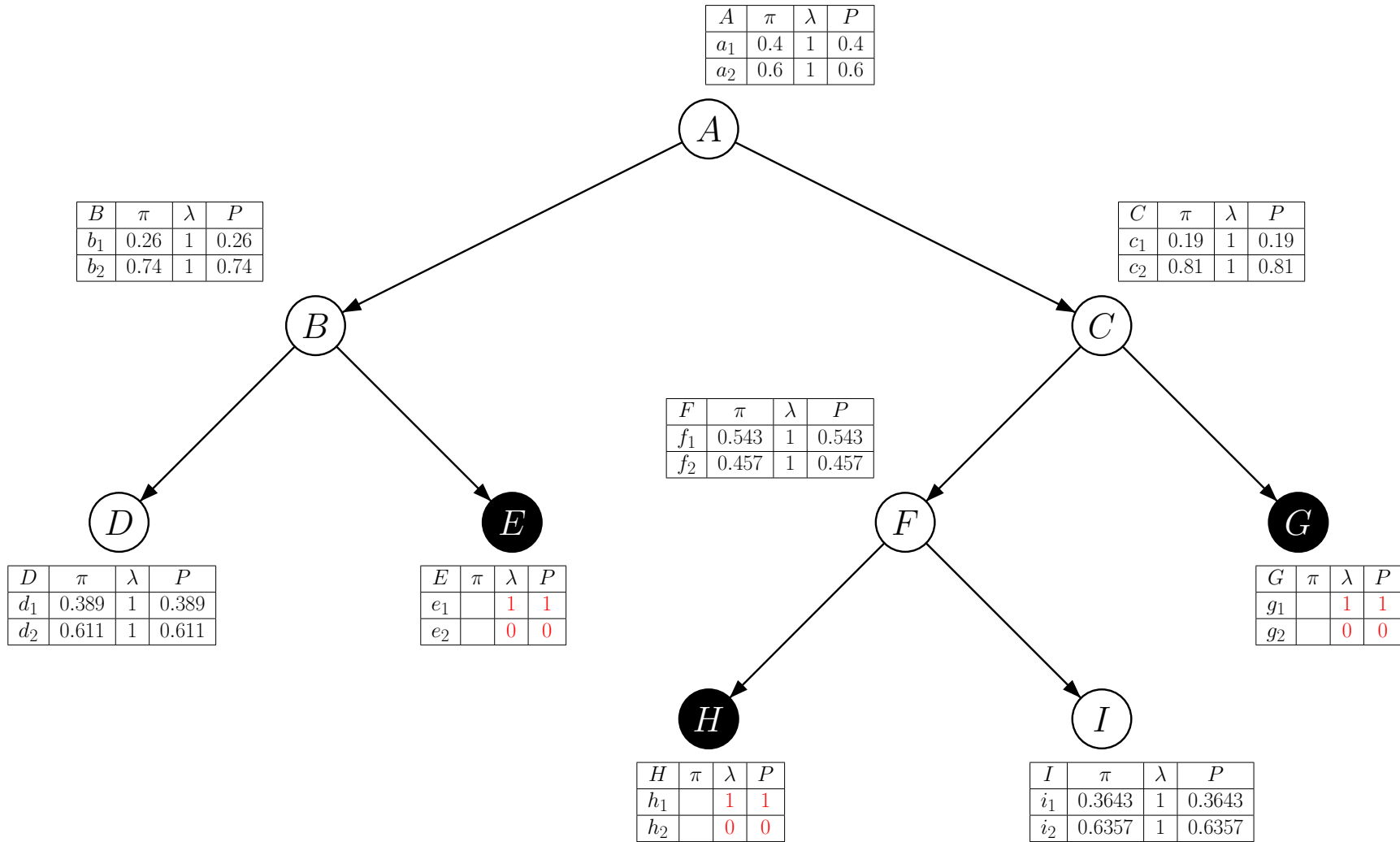
Larger Network (1): Parameters



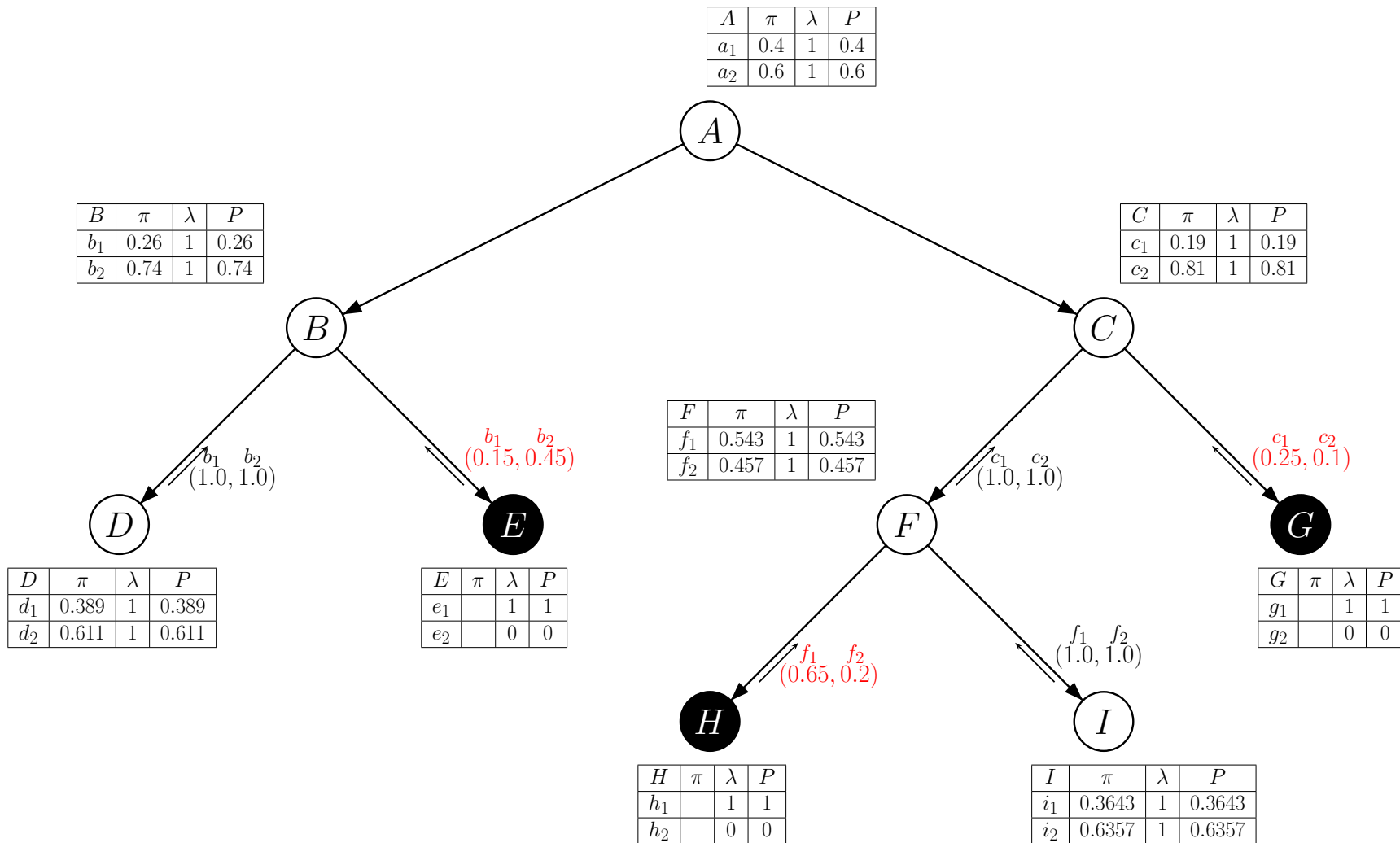
Larger Network (2): After Initialization



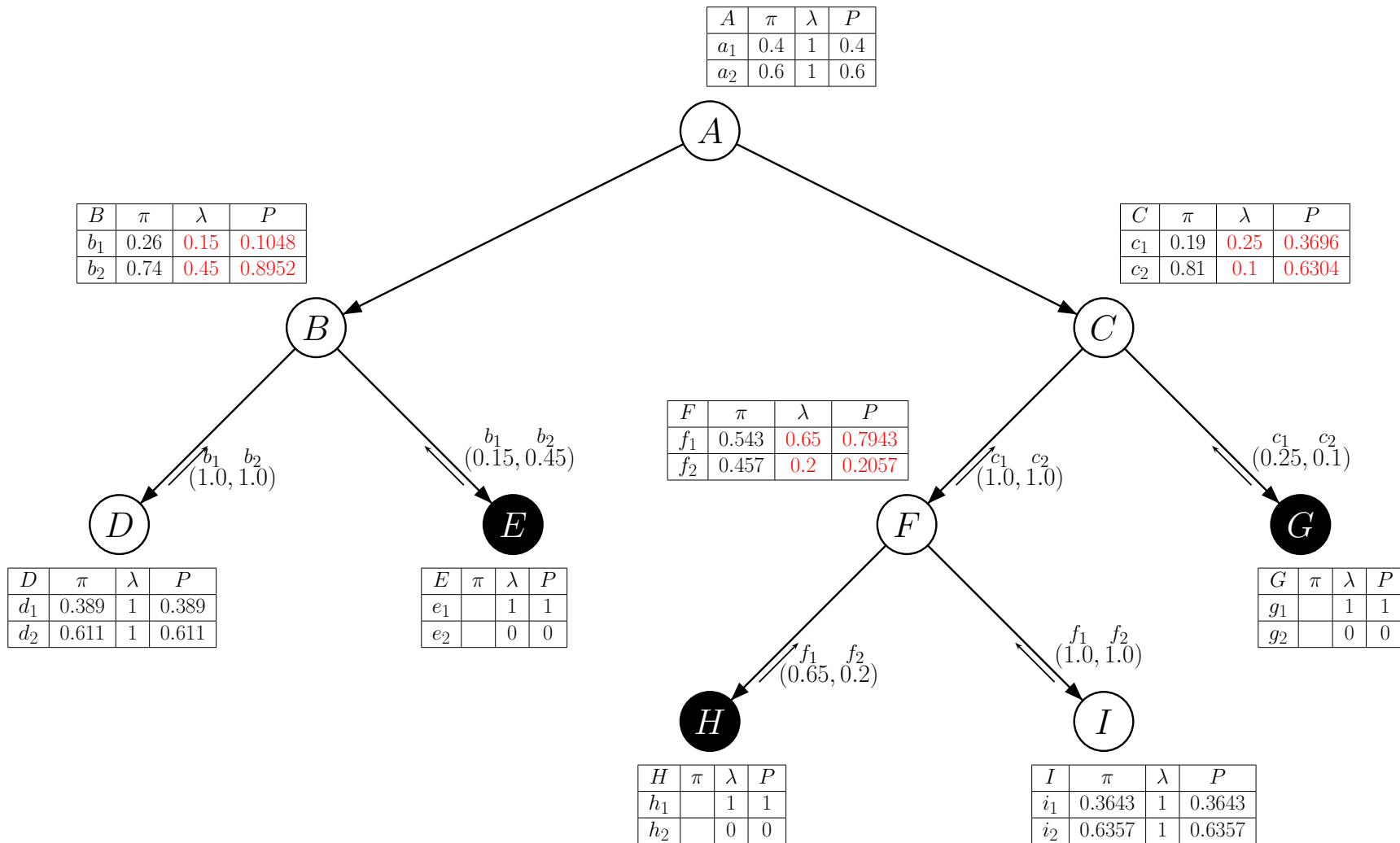
Larger Network (3): Set Evidence e_1, g_1, h_1



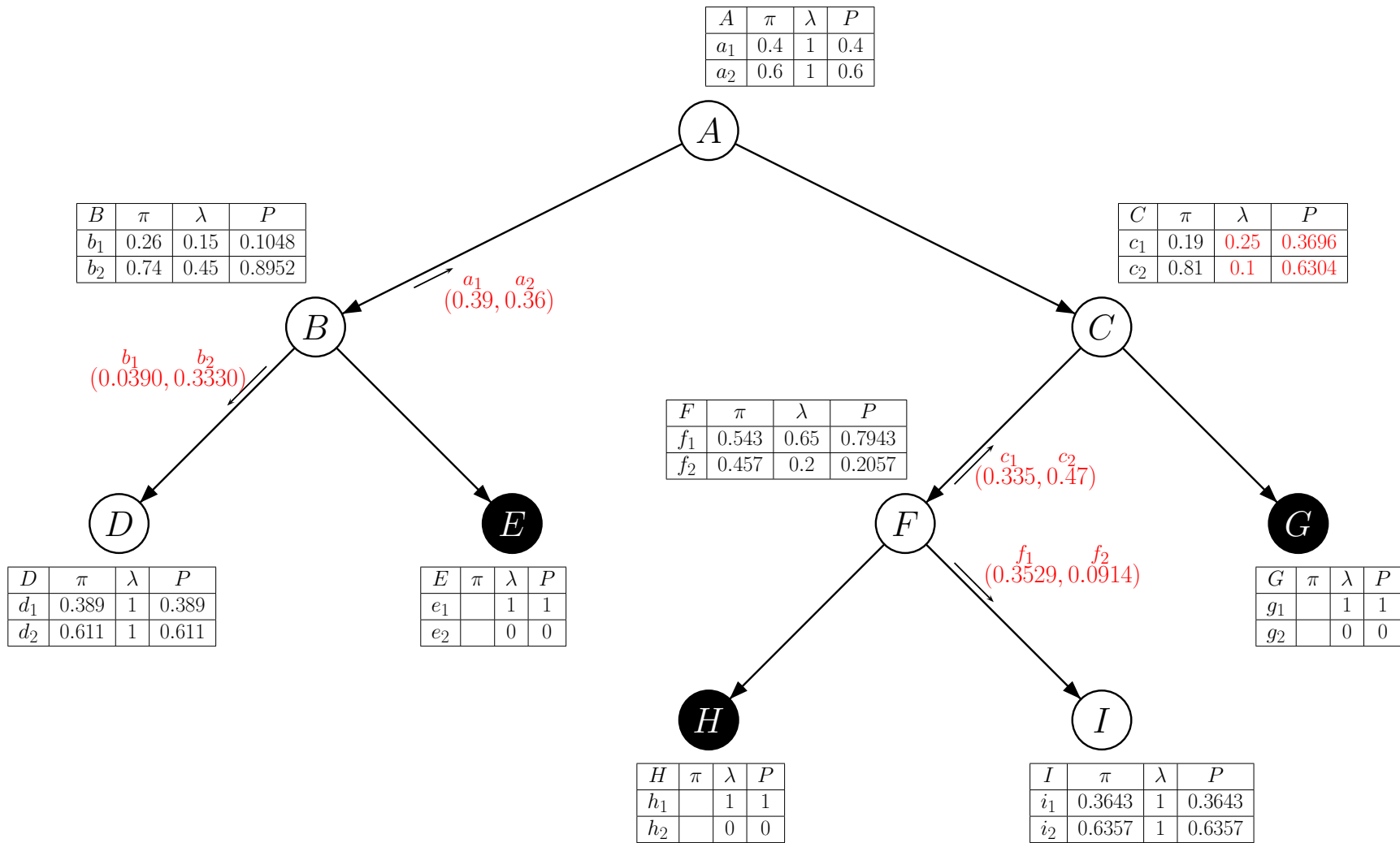
Larger Network (4): Propagate Evidence



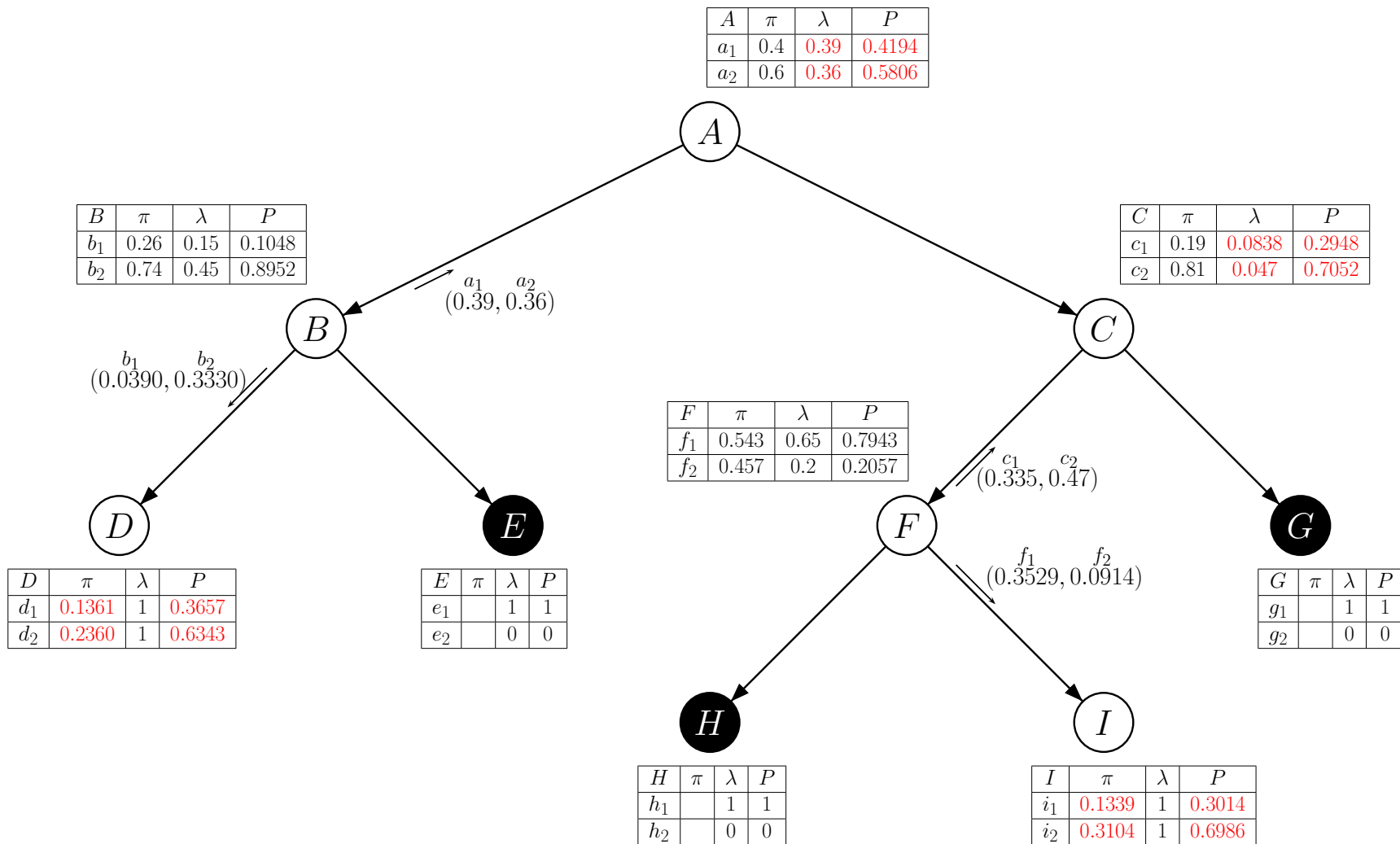
Larger Network (5): Propagate Evidence, cont.



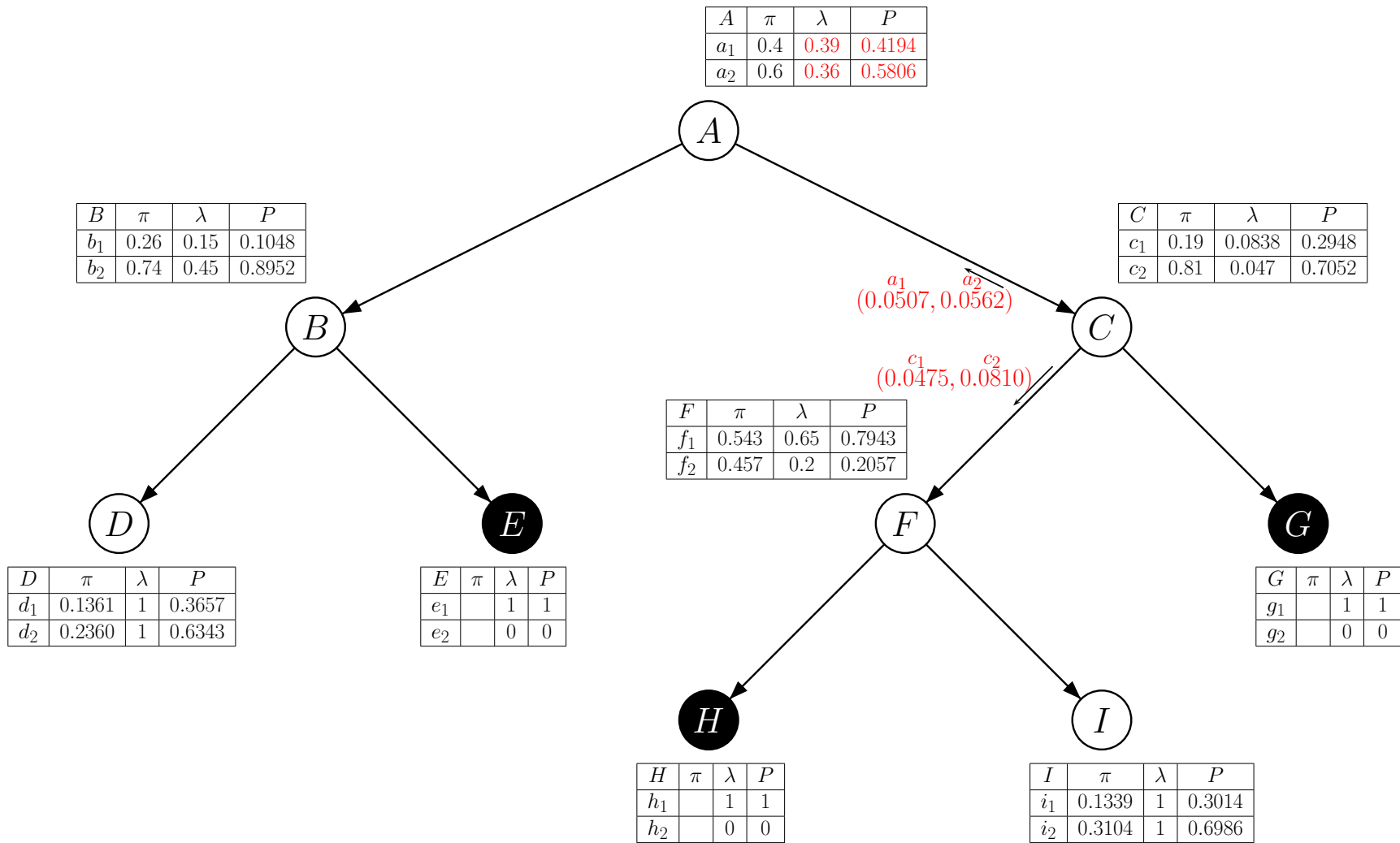
Larger Network (6): Propagate Evidence, cont.



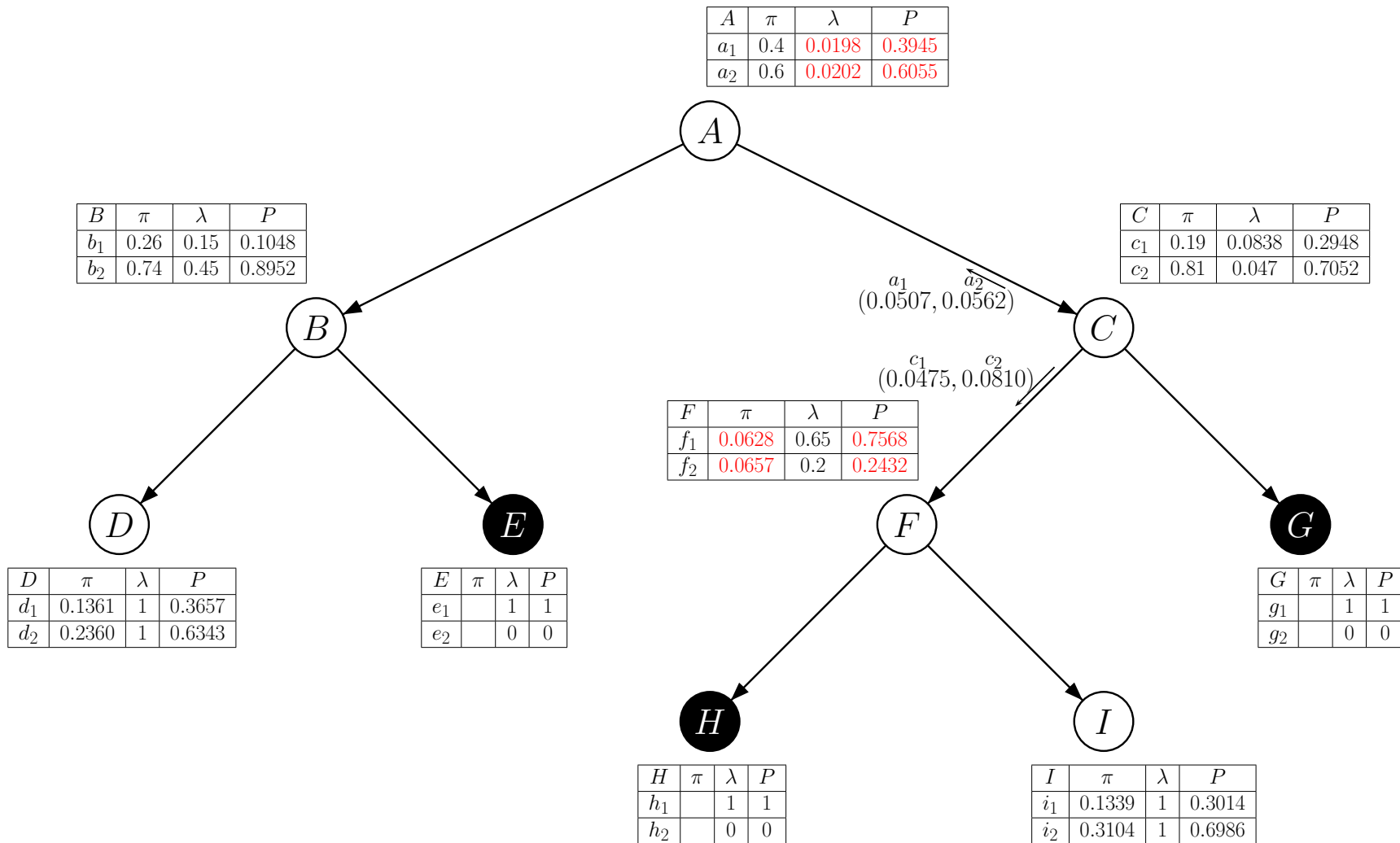
Larger Network (7): Propagate Evidence, cont.



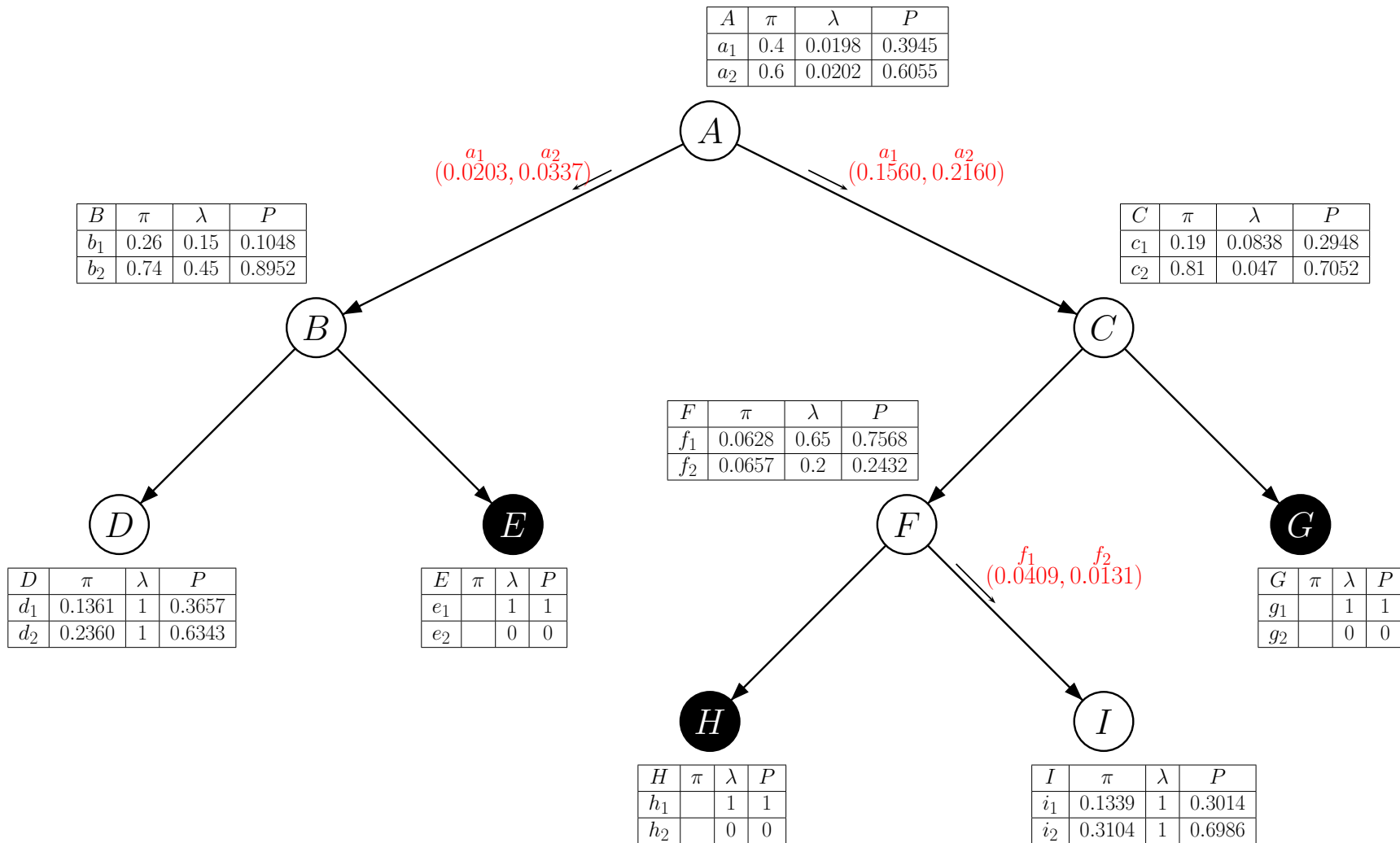
Larger Network (8): Propagate Evidence, cont.



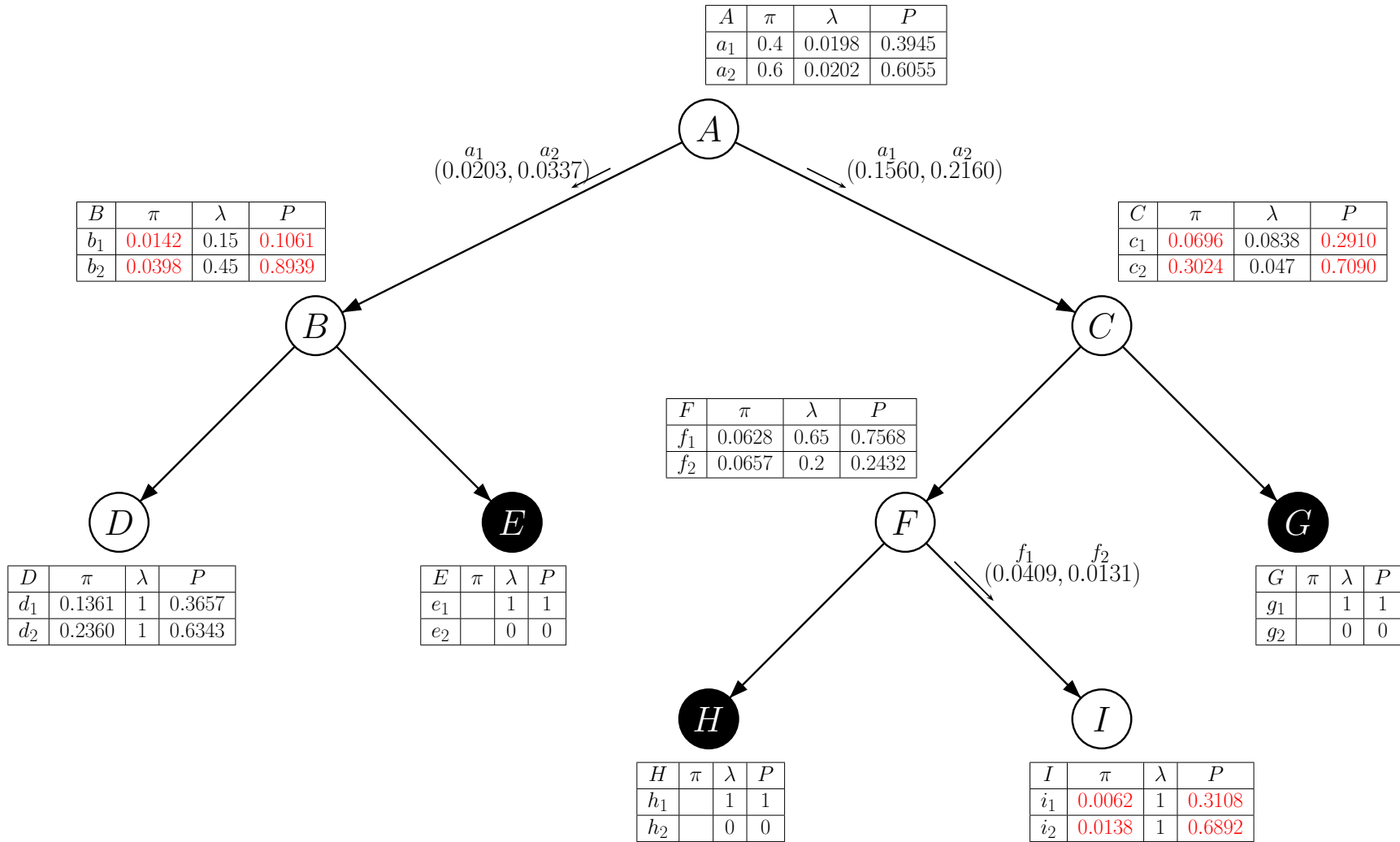
Larger Network (9): Propagate Evidence, cont.



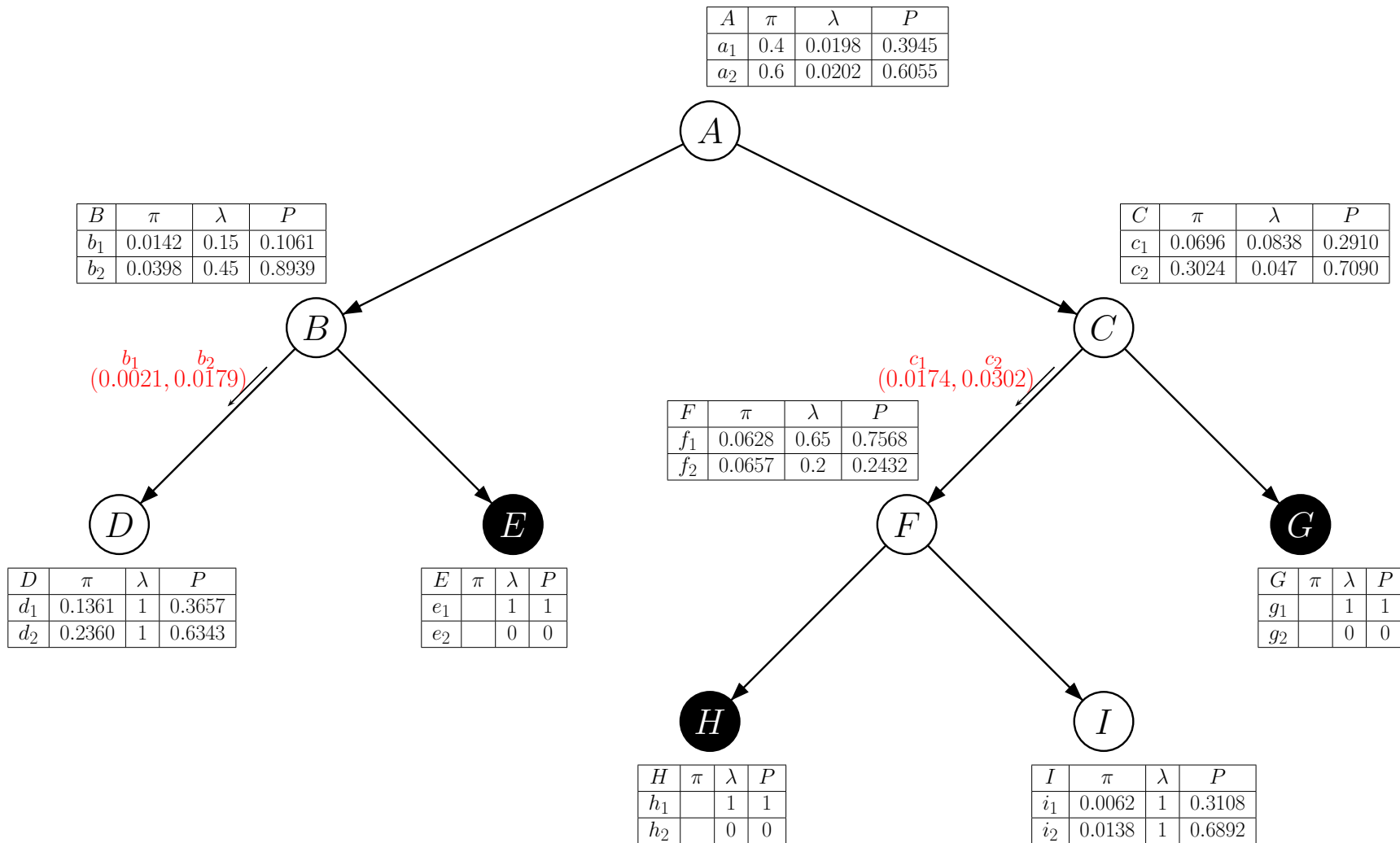
Larger Network (10): Propagate Evidence, cont.



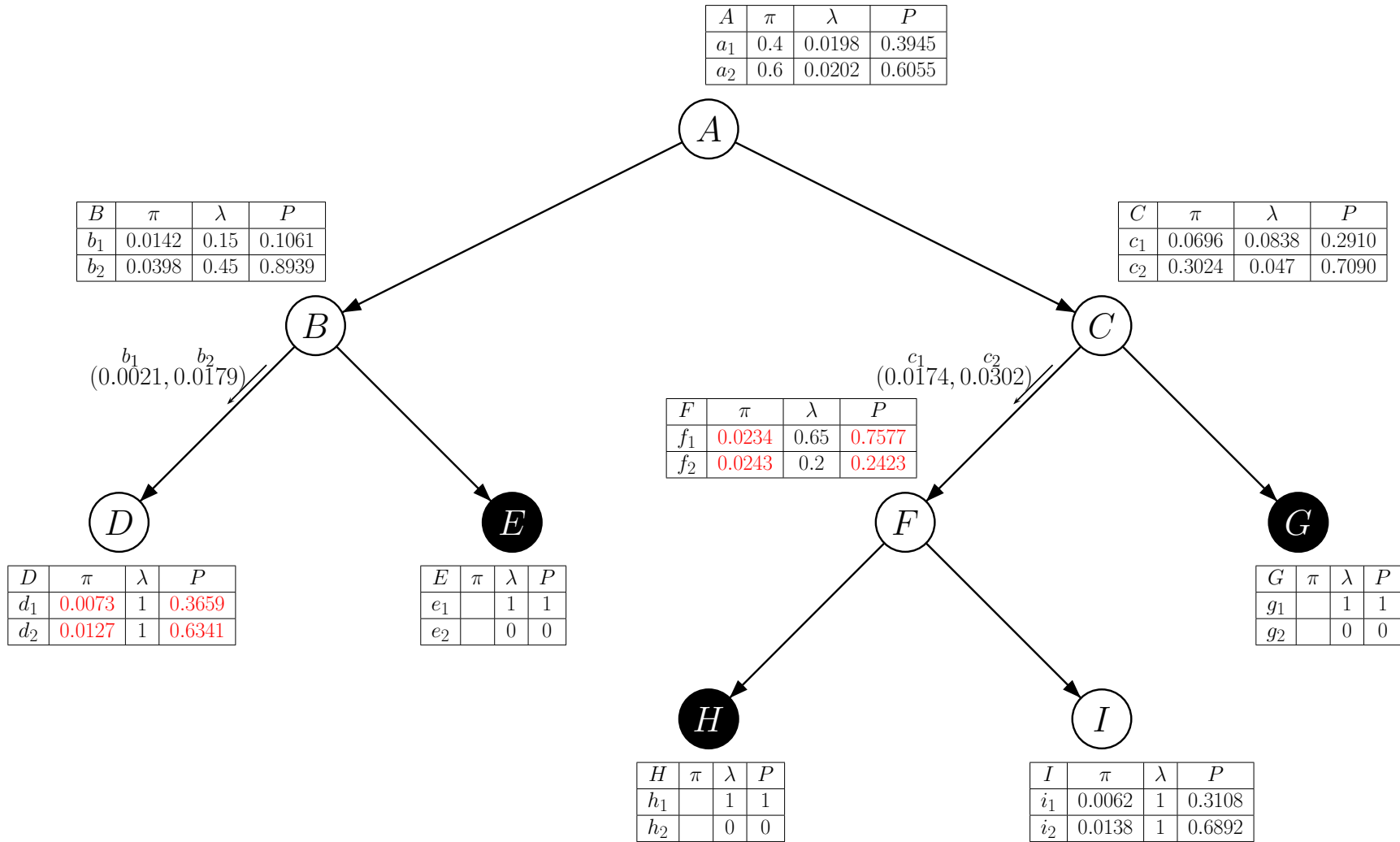
Larger Network (11): Propagate Evidence, cont.



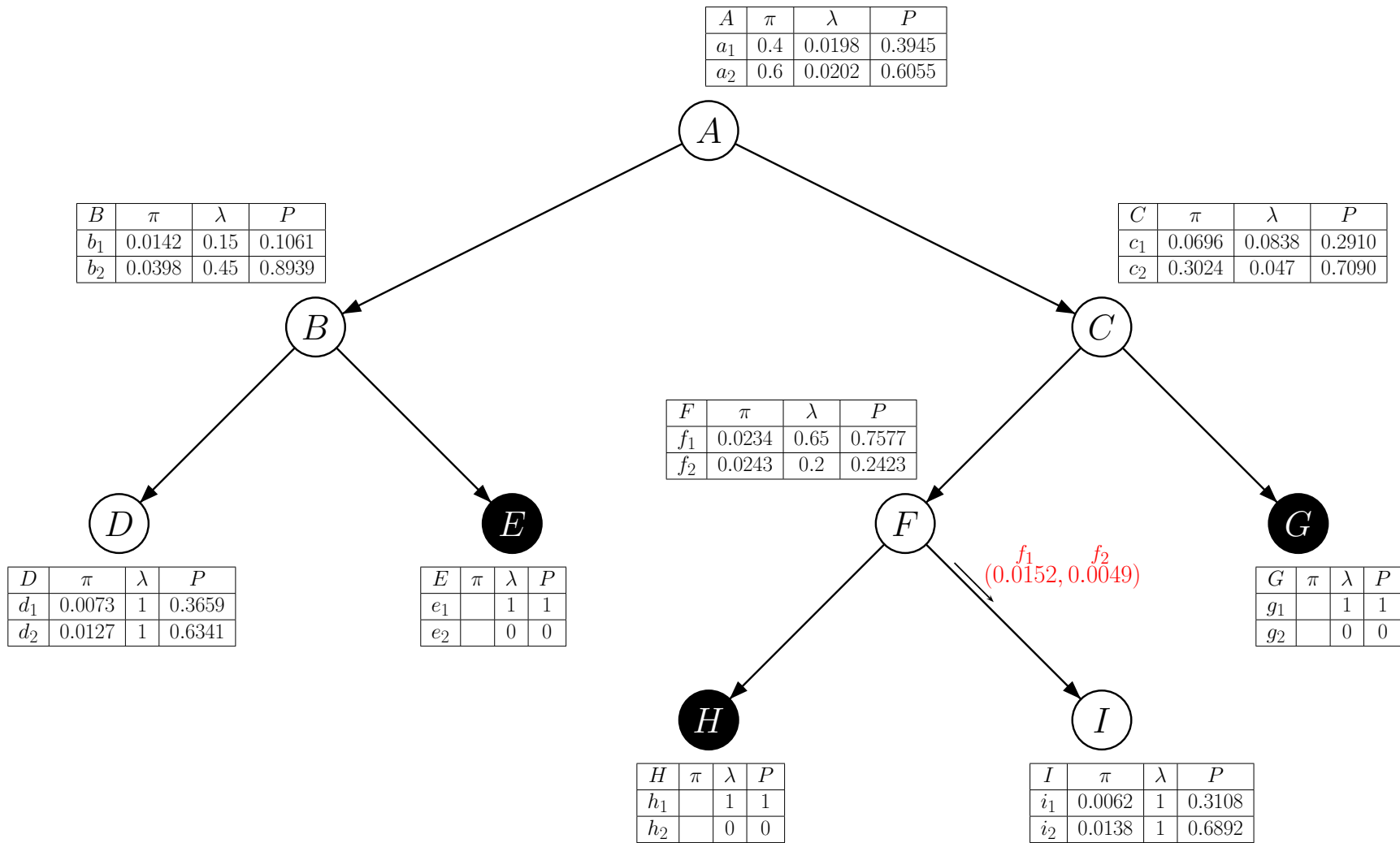
Larger Network (12): Propagate Evidence, cont.



Larger Network (13): Propagate Evidence, cont.



Larger Network (14): Propagate Evidence, cont.



Larger Network (15): Finished

