Knowledge Representation and Reasoning

University "Politehnica" of Bucharest
Department of Computer Science

Fall 2009

Adina Magda Florea
http://turing.cs.pub.ro/krr_09
curs.cs.pub.ro

Lecture 3

ASSUMPTION BASED REASONING

Lecture outline

- Role of assumptions
- Nonmonotonic reasoning
- Default reasoning
- Truth Maintenance Systems (TMS)

1. Role of assumptions

- Reason from facts together with a set of assumptions we are prepared to make

4 main applications of the idea:

- Nonmonotonic reasoning
- Model based diagnosis and recognition (abduction)
- Design
- Inductive learning

$$\frac{Q(a) \qquad (\forall x)\ (P(x) \rightarrow Q(x))}{P(a)}$$

2. Non monotonic reasoning

- KB - a set of formulas (consistent) in FOPL
- L – a logical system
- Th(KB) – set of provable theorems in KB
  - Th(KB) – fixed point operator - computes the closure of KB according to the rules of inference in L (the least fixed point of this closure process)

- **Monotonic reasoning** – (all) assumptions are true – facts
- F = Th(KB), apply inferences and get new facts to extend KB to KB1, then $\mathbf{KB1 \subseteq KB \cup F}$

Non monotonic reasoning

- **Non Monotonic reasoning** – facts (true assumptions) + beliefs / hypothesis (assumptions that are presumed to be true)

- F = Th(KB)
- Apply inferences and get new facts to extend KB to KB1
- Then $\mathbf{KB1} \subseteq \mathbf{(KB \setminus A)} \cup \mathbf{F}$, where A is the set of assumptions that were defeated by facts inferred in KB1

Why non monotonic reasoning?

- **The ABC murder story** (from *The Web of Belief*, Quine and Ullian, 1978)

- Let **A**lecu, **B**arbu and **C**ezar be suspects in a murder case (they all benefit from the murder).
- **A**lecu has an alibi, in a register of a respectable hotel in Arad.
- **B**arbu also has an alibi, for his brother-in-law testified that Barbu was visiting him in Buzau at the time.
- **C**ezar pleads alibi too, claiming to have been watching a ski meet in Cioplea, but we have only his word for that.

Why non monotonic reasoning?

- So we believe:

  (1) That **A**lecu has not commit the crime

  (2) That **B**arbu did not

  (3) That **A**lecu or **B**arbu or **C**ezar did

- But presently **C**ezar documents his alibi – he had the good luck to have been caught by television in the sidelines at the ski meet. A new belief is thus thrust upon us:

  (4) That **C**ezar did not

(1), (2), (3), (4) are **inconsistent**, we have to reject a belief

What is NMR good for?

- How we can extend the KB so that we can draw inferences based on facts and on the absence of the facts (*"know that P"* vs. *"do not know if P"*)?
- How can we efficiently update the KB when we add or delete a belief? See *Justifications*
- How can we use existing knowledge to solve conflicts, in case there are contradictory facts (derived by nonmonotonic inferences)

NMR approaches

Extend FOPL

- McDermott and Doyle – extend with a modal operator **M** – consistency of an assumption
- Reiter – use a **default rule of inference** – default reasoning
- Mc Carthy – **circumscription** – situations as objects to reason upon

NMR approaches

Use a meta approach

- **Truth Maintenance Systems (TMS)**
- A constraint system among objects in FOPL
- Keep a consistent subset of theorems, according to the constraints
- Perform inferences of the form *"If P is consistent then Q"*

3. Default reasoning

- "Most Ps are Qs"
- "Most Ps have property Q"
- Problems in FOPL

$$(\forall x)\ (\mathrm{Pasare}(x) \wedge \sim \mathrm{Pinguin}(x) \wedge \sim \mathrm{Strut}(x) \wedge \ldots \rightarrow \mathrm{Zboara}(x))$$

- If x is a bird, and if there is no contradictory evidence, then x flies"

3.1 Reiter's Default Logic

- Based on FOPL
- Introduces a new rule of inference to represent default reasoning
- P : Q / R
- "If P is true and it is consistent to assume Q then infer R"
- P, Q, R are wffs in FOPL
- Simplest rules - : P / P

Reiter's DL – formal definition

$$L = \langle A, \mathcal{F}, \mathcal{A}, \Re \rangle$$

- **Default rule** $$\frac{\alpha(x) : \beta_1(x), \ldots, \beta_m(x) \,/\, w(x)}{}$$

where

$$\alpha(x), \beta_1(x), \ldots, \beta_m(x), w(x)$$

are FOPL formulas

Premise Consequence

Reiter's DL – formal definition

$L = < A, F, A, \Re >$

- A **default theory** is a pair (D, W), where D is a set of default rules to be added to the inference rules of L and W is a set of wffs in *F*
- Be a default theory (D, W) – the rules of D have the form
  $(\alpha : \beta_1, \ldots, \beta_m / w)$ where $\alpha, \beta_1, \ldots, \beta_m, w$ are wffs in *F*.
- For any subset of formulas $\Gamma \subseteq F$, be S(Γ) the smallest set which satisfies the following properties:
  $W \subseteq S(\Gamma)$
  $Th(S(\Gamma)) = S(\Gamma)$ where $Th(\Gamma) = \{P \mid P \in F, \Gamma \xrightarrow[L]{} P\}$
  $(\alpha : \beta_1, \ldots, \beta_m / w) \in D$ and $\alpha \in S(\Gamma)$ and $\sim \beta_1, \ldots, \sim \beta_m \notin \Gamma$ then $w \in S(\Gamma)$
- A set **E $\subseteq$ F is an extension of Δ** iff **S(E) = E**, i.e., iff E is a fixed point of the S operator.

Any extension of a default theory can be seen as an acceptable (consistent) set of beliefs that we have about an incompletely specified world.

Reiter's DL extensions

Ex 1

- (D,W)

W = { $\forall$x pinguin(x) $\rightarrow$ pasare(x)

$\forall$x pinguin(x) $\rightarrow$ ~zboara(x)

pasare(Pingu) }

D = { d:$\forall$x pasare(x) **:** zboara(x) ***/*** zboara(x) }

E = Th(W $\cup$ {pasare(Pingu) $\rightarrow$ zboara(Pingu) } ➔ zboara(Pingu) $\in$ E

1 extension

Reiter's DL extensions

Ex 2

- (D,W)

W = { $\forall$x liliac(x) $\rightarrow$ mamifer(x)

liliac(Coco)

pui(Coco) }

D = { d1: $\forall$x mamifer(x) **:** ~zboara(x) ***/*** ~zboara(x)

d2: $\forall$x liliac(x) **:** zboara(x) ***/*** zboara(x)

d3: $\forall$x pui(x) **:** ~zboara(x) ***/*** ~zboara(x) }

E1 = Th(W $\cup$ { Base(d1) $\cup$ Base(d3) } ➔ ~zboara(Coco) $\in$ E1

E2 = Th(W $\cup$ { Base(d2) } ➔ zboara(Coco) $\in$ E2

2 extensions

Reiter's DL extensions

- Ex 3 – no extension

  (D,W) W = { } D={d: **:**A ***/*** ~A}

- What shall we do when there are several extensions?
- Some solutions:
  - Preferences
  - Possible worlds approach

3.2 DR in inheritance systems

$$(\forall x)\,(Jucator-de-baschet(x) \rightarrow Barbat(x))$$

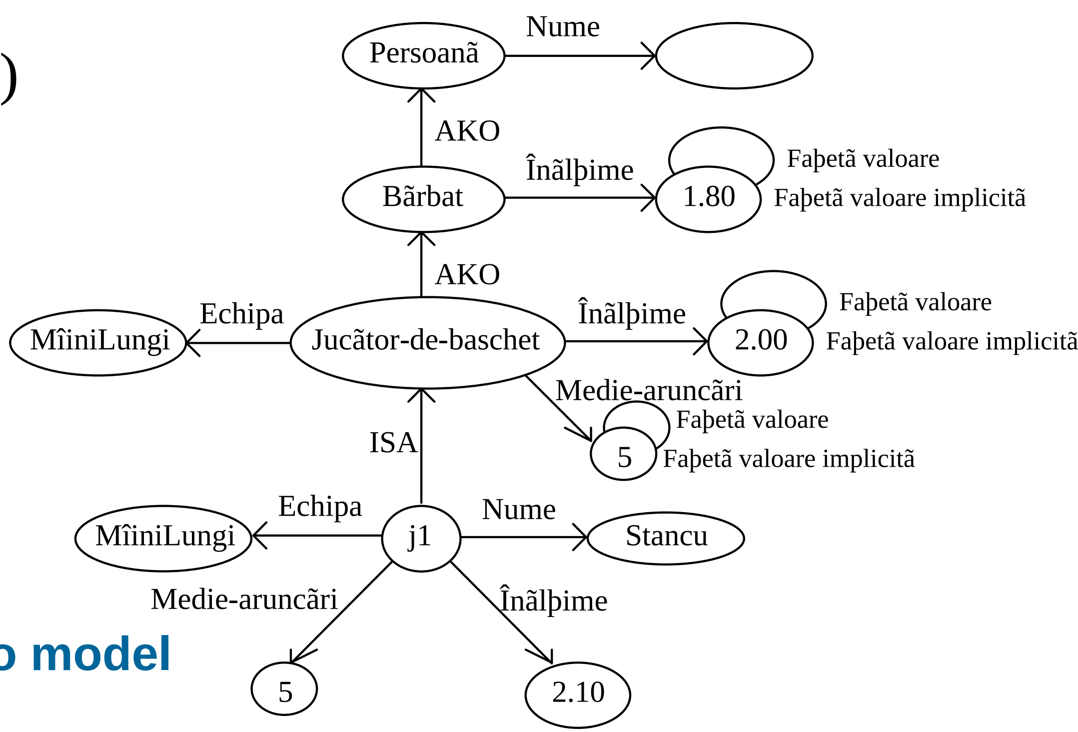
$$(\forall x)\,(Barbat(x) \rightarrow Persoana(x))$$

$$(\forall x)\,(Barbat(x) : Inaltime(x,1.80)/Inaltime(x,1.80))$$

$$(\forall x)\,(Jucator-de-baschet(x) : Inaltime(x,2.00)/Inaltime(x,2.00))$$

$$Jucator-de-baschet(Stancu)$$

$$\hat{I}naltime(Stancu,2.10)$$



Aim: Use Reiter's Default Logic to model inheritance

DR in inheritance systems

$$(\forall x)\,(\text{Jucator}-\text{de}-\text{baschet}(x) \rightarrow \text{Barbat}(x))$$

$$(\forall x)\,(\text{Barbat}(x) \rightarrow \text{Persoana}(x))$$

$$(\forall x)\,(\text{Barbat}(x) : \text{Inaltime}(x,1.80)/\text{Inaltime}(x,1.80))$$

$$(\forall x)\,(\text{Jucator}-\text{de}-\text{baschet}(x) : \text{Inaltime}(x,2.00)/\text{Inaltime}(x,2.00)$$

$$\text{Jucator}-\text{de}-\text{baschet}(\text{Stancu})$$

Barbat has the default value of height 1.80, provided Inaltime(x,1.80) is consistent, i.e., it is not defeated

Jucator-de-basket has the default value of height 2.00, provided Inaltime(x,2.00) is consistent, i.e., it is not defeated

If we know that Stancu is Jucator-de-basket, 2 extensions are possible

To prevent this, we can add:

$$(\forall x)\,(\text{Barbat}(x) : \sim \text{Jucator}-\text{de}-\text{baschet}(x) \wedge \text{Inaltime}(x,1.80)/\text{Inaltime}(x,1.80))$$

DR in inheritance systems

$$(\forall x)\,(Jucator-de-baschet(x) \rightarrow Barbat(x))$$

$$(\forall x)\,(Barbat(x) \rightarrow Persoana(x))$$

$$(\forall x)\,(Barbat(x) : Inaltime(x,1.80)/Inaltime(x,1.80))$$

$$(\forall x)\,(Jucator-de-baschet(x) : Inaltime(x,2.00)/Inaltime(x,2.00))$$

$$Jucator-de-baschet(Stancu)$$

What if we have several particular cases?
We have to add default rules for all

$$(\forall x)\,(Barbat(x) : \sim Jucator-de-baschet(x) \wedge Inaltime(x,1.80)/Inaltime(x,1.80))$$

$$(\forall x)\,(Barbat(x) : \sim Jucator-de-baschet(x) \wedge \sim Chinez(x) \wedge \sim Jocheu\,(x) \wedge Inaltime(x,1.80)/Inaltime(x,1.80))$$

DR in inheritance systems

$$(\forall x)\,(Jucator-de-baschet(x) \rightarrow Barbat(x))$$

$$(\forall x)\,(Barbat(x) \rightarrow Persoana(x))$$

A more elegant way to deal with such cases is to add a predicate Diferit and specify one default rule

$$(\forall x)\,(Barbat(x) \wedge \sim Diferit(x, aspect1) \rightarrow Inaltime(x, 1.80))$$

$$(\forall x)\,(Jucator-de-baschet(x) \rightarrow Diferit(x, aspect1))$$

$$(\forall x)\ (Chinez(x) \rightarrow Diferit(x, aspect1))$$

$$(\forall x)\ (Jocheu(x) \rightarrow Diferit(x, aspect1))$$

$$(\forall x)(\forall y)\,(: \sim Diferit(x, y) / \sim Diferit(x, y))$$

4. TMS

Truth Maintenance Systems

- Non monotonic reasoning
- Increase efficiency of problem solving
- Good for:
  - validate assumptions
  - redraw abandoned conclusions
    - NMR
    - dependency directed backtracking (DDBkt)
  - control program actions
  - explain reasoning

4.1 DDBkt

(1) $x \in \{0,1\}$ (2) $a = e_1(x)$

(3) $y \in \{0,1\}$ (4) $b = e_2(x)$

(5) $z \in \{0,1\}$ (6) $c = e_3(x)$

(7) $b \neq c$ (8) $a \neq b$

$e_i(x) = (x+100000)!$, $i=1,2,3$

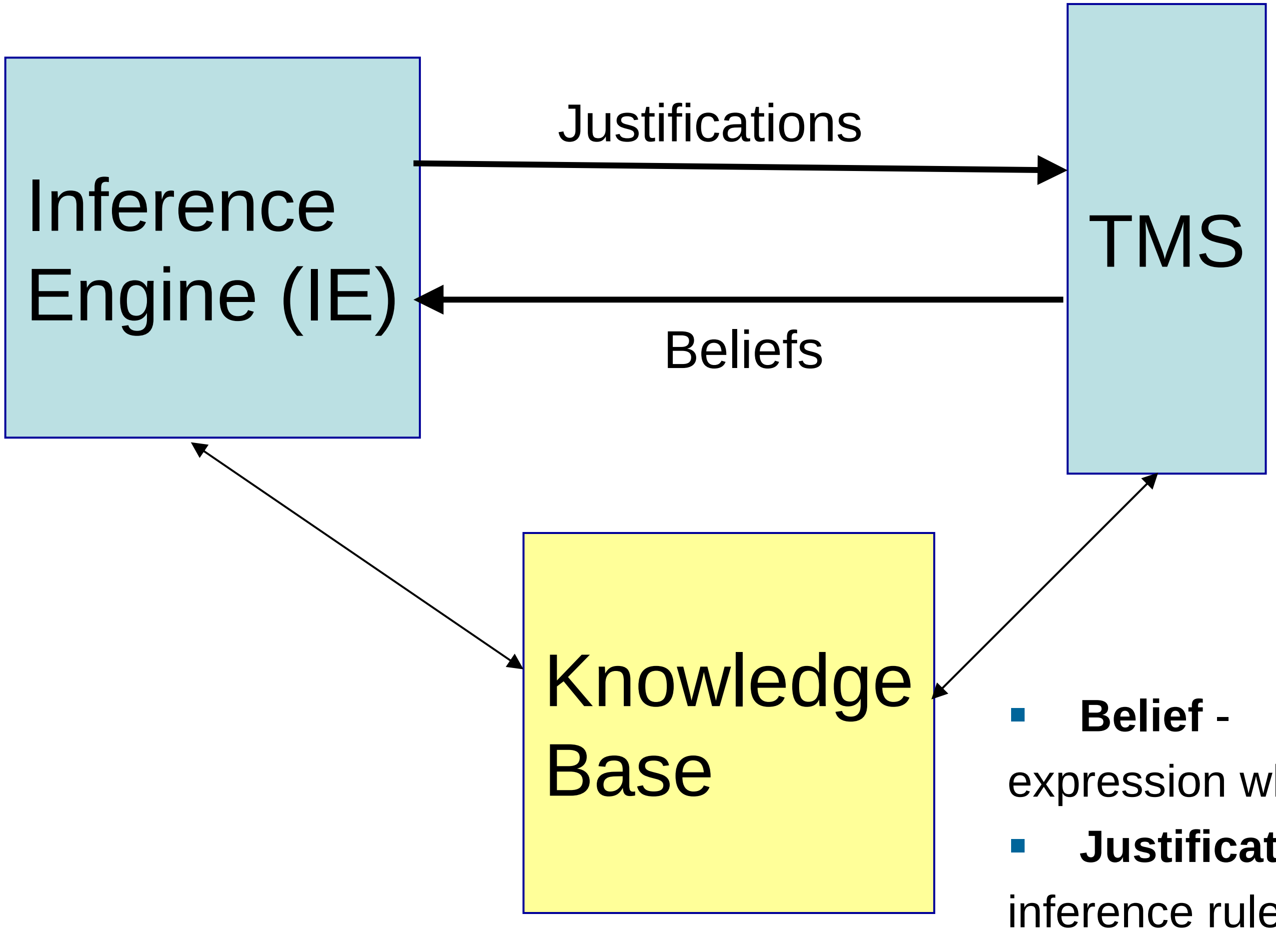
Find a, b, c to satisfy (1)-(8)

- x=0, y=0, z=0 and x=0, y=0, z=1 are rejected because of (7) and (8) – y's value
- backtrack to y
- $c = e_3(0)$ and $c = e_3(1)$ are lost

4.2 TMS principles

- Each action in the problem solving process has an *associated justification*
- When a contradiction is obtained, find the **minimal set of assumptions which generated the contradiction** – if we eliminate an element from this set, the justification for the contradiction is not valid any more and the contradiction is removed
- **Propagate** the effects of **adding a justification** and of **eliminating a belief** + keep consistency
- **Select an assumption** from the minimal set which generated the contradiction and **defeat** it

4.3 TMS Structure



- **Belief** -
expression which can be true or false
- **Justification** -
inference rule (step) which lead to a belief
- **Premise** -
true fact (no justification needed)

TMS Structure

- Set of nodes in the TMS
- Every node has 2 states
  - IN (believed node)
  - OUT (node not believed)
- Set of valid assumptions = set of IN nodes in the TMS

TMS Structure

- The nodes may be:
  - **premises** – no justification needed, IN nodes
  - **belief** – may be believed (IN) or not (OUT)
  - **assumption** - with supporting justifications – IN nodes if the justification is true, OUT nodes otherwise
- A **justification** = set of nodes used to infer the justified node, composed of 2 lists:
  - **IN List** – nodes that have to be IN for the justification to be true / to support an IN node
  - **OUT List** – nodes that have to be OUT for the justification to be true / to support an IN node

The ABC Murder

Initial set of beliefs

Beneficiar (Alecu)　　　　~Alibi(Alecu)

Beneficiar(Barbu)

Beneficiar(Cezar)

R1: **if** Beneficiar(x)
and ifnot Alibi(x)
then Suspect(x)

Nonmonotonic production rule

$$\text{Beneficiar}(x) : \sim \text{Alibi}(x) \,/\, \text{Suspect}(x)$$

The ABC Murder

Rules

R1: **if** Beneficiar(x)
and ifnot Alibi(x)
then Suspect(x)

$$\text{Beneficiar}(x) : \ \sim \text{Alibi}(x) \,/\, \text{Suspect}(x)$$

R2: **if** Hotel(x,y)
and Departe(y)
and ifnot Falsificat(y)
then Alibi(x)

R3: **if** Aparat(x, y)
and ifnot Minte(y)
then Alibi(x)

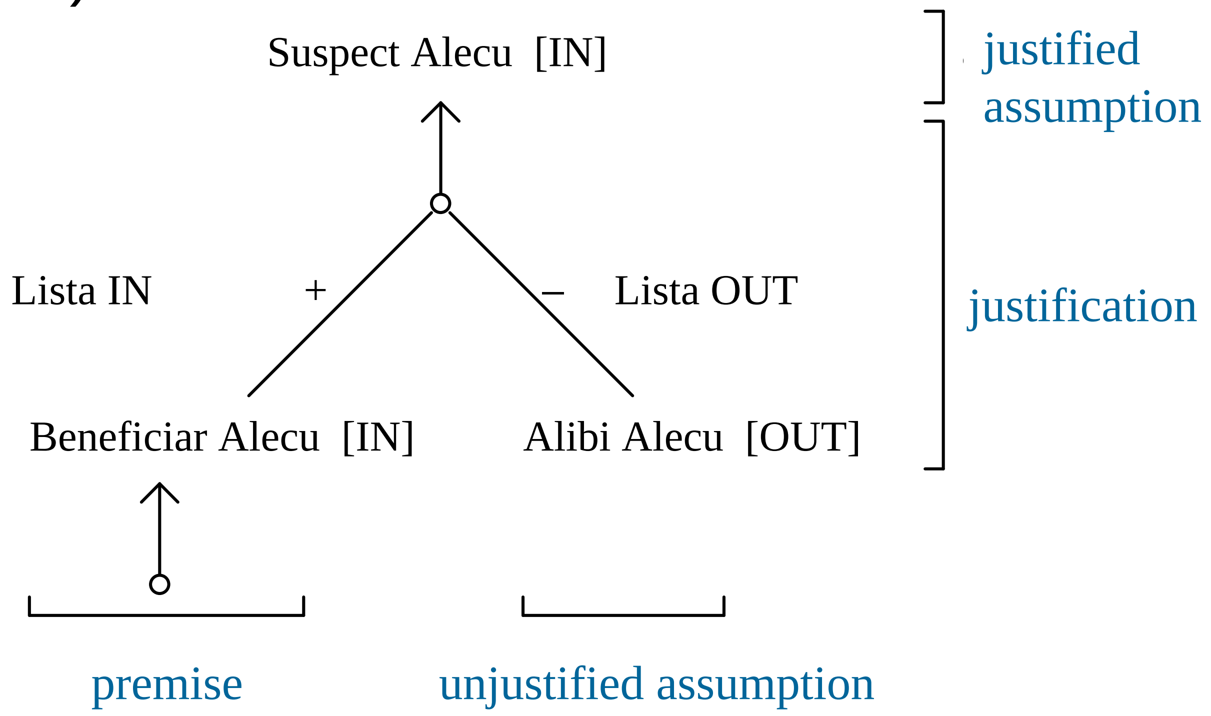
R4: **ifnot** ~Spune_adevar(x)
then Alibi(x)

TMS for the ABC Murder

(Node (IN_List   OUT_List))

N1 = Suspect (Alecu)
N2 = Beneficiar (Alecu)
N3 = ~Alibi(Alecu)

(N1, ((N2) (N3)))



TMS for the ABC Murder

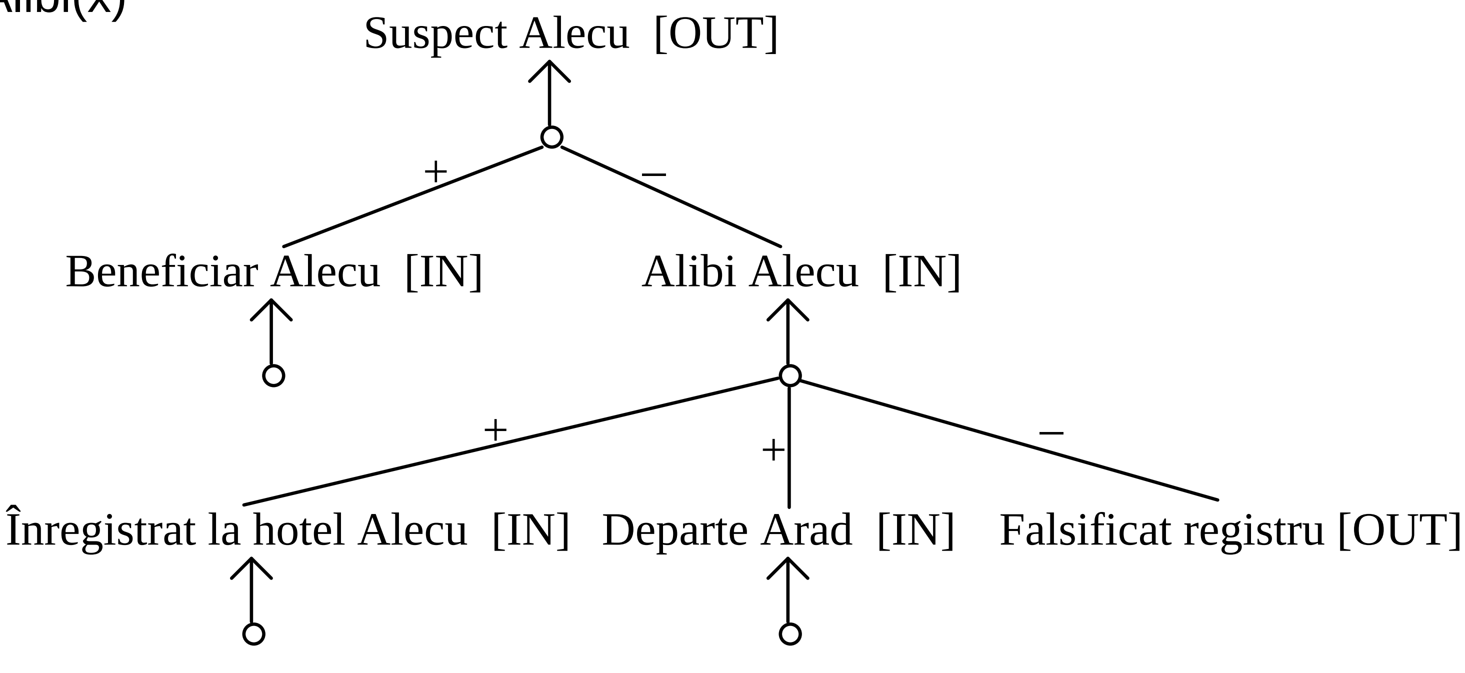
Suppose new beliefs are added

Inregistrat la hotel Alecu, Departe Arad, ~Falsificat registru

R2: **if** Hotel(x,y)
and Departe(y)
and ifnot Falsificat(y)

then Alibi(x)

R1: **if** Beneficiar(x)
and ifnot Alibi(x)
then Suspect(x)



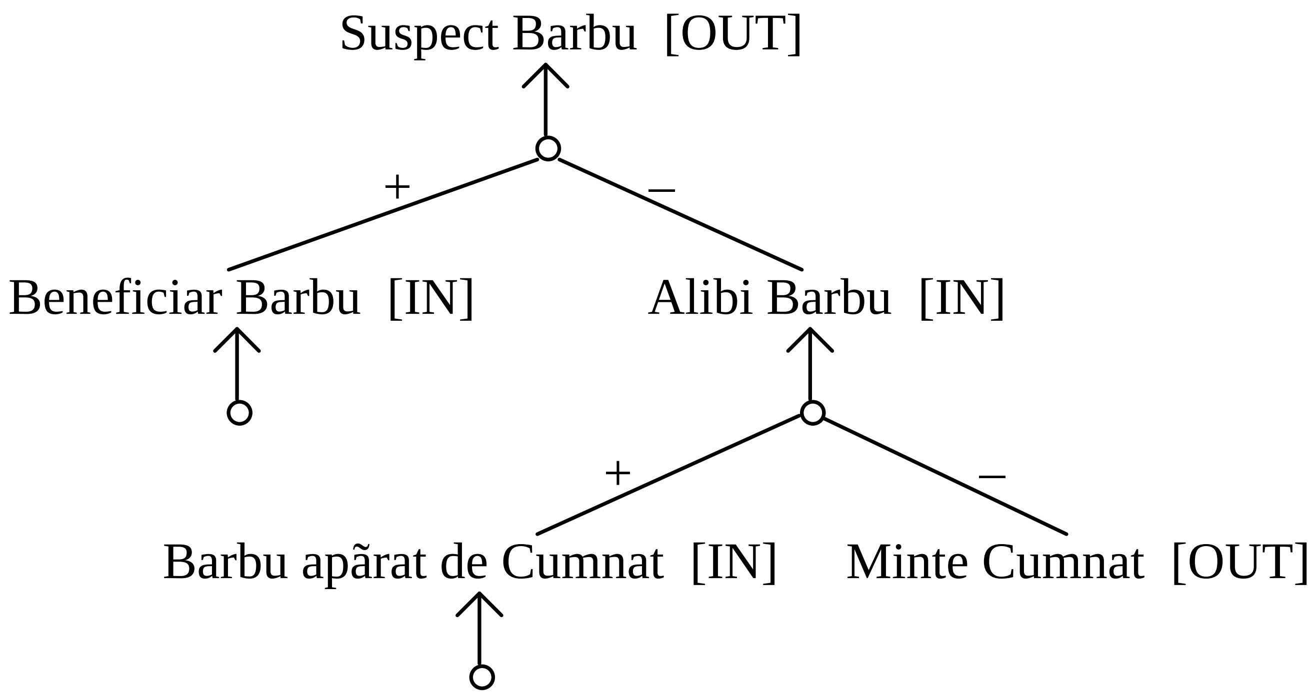
TMS for the ABC Murder

Suppose another new belief is added

Barbu aparat de Cumnat

R3: **if** Aparat(x, y)
and ifnot Minte(y)
then Alibi(x)

R1: **if** Beneficiar(x)
and ifnot Alibi(x)
then Suspect(x)

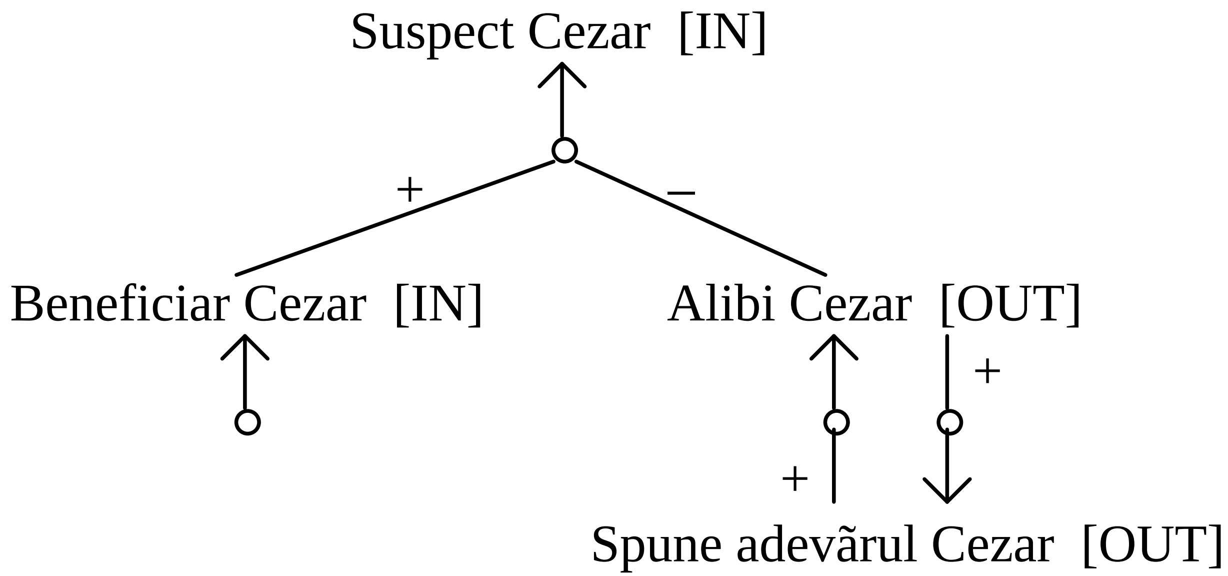


TMS for the ABC Murder

What about Cezar?

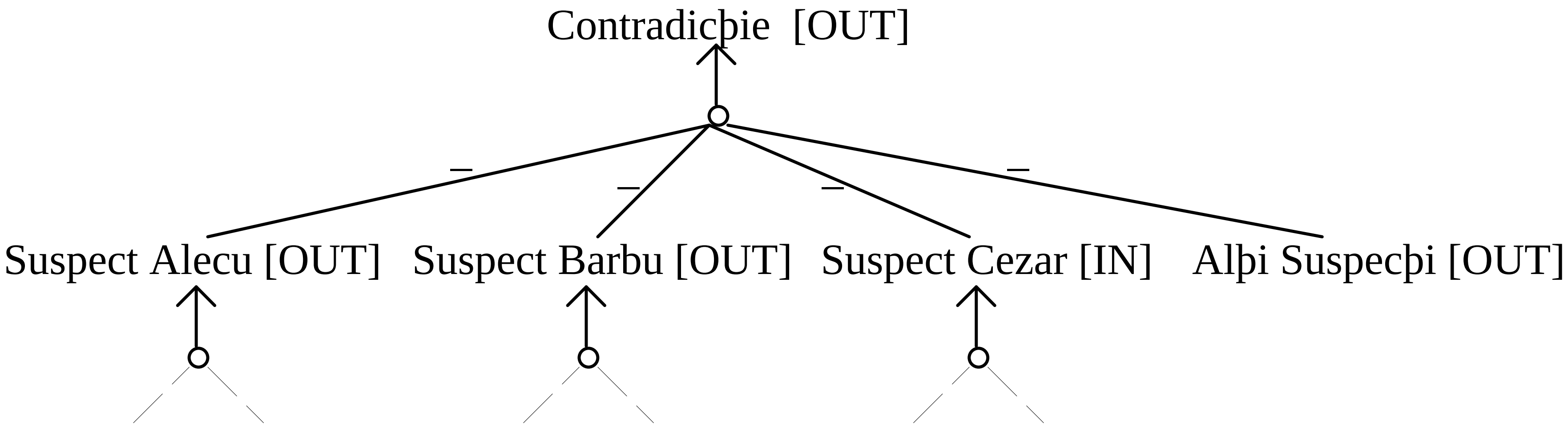
R4: **ifnot** ~Spune_adevar(x)
then Alibi(x)

R1: **if** Beneficiar(x)
and ifnot Alibi(x)
then Suspect(x)



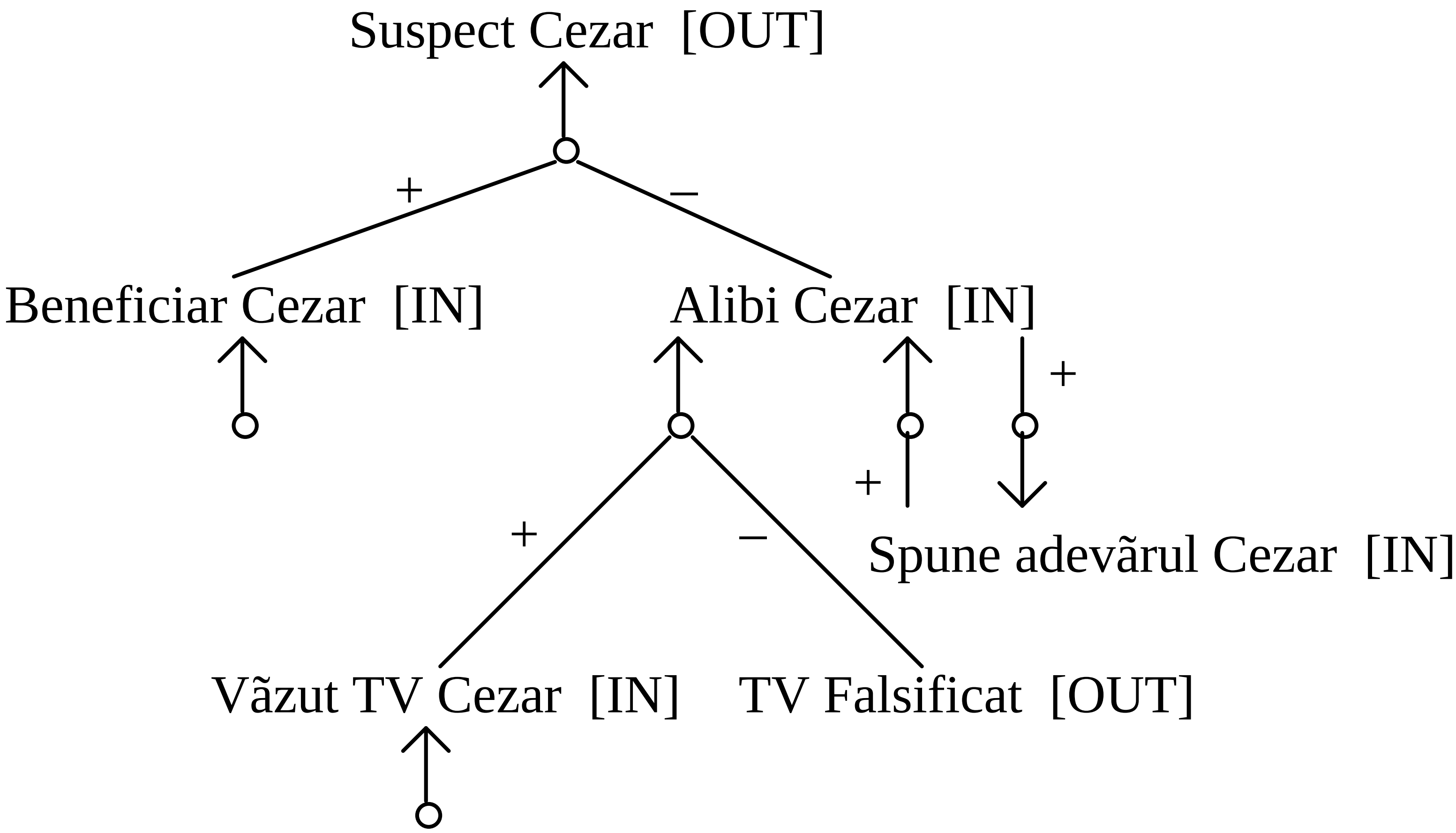
TMS for the ABC Murder

Represent contradiction



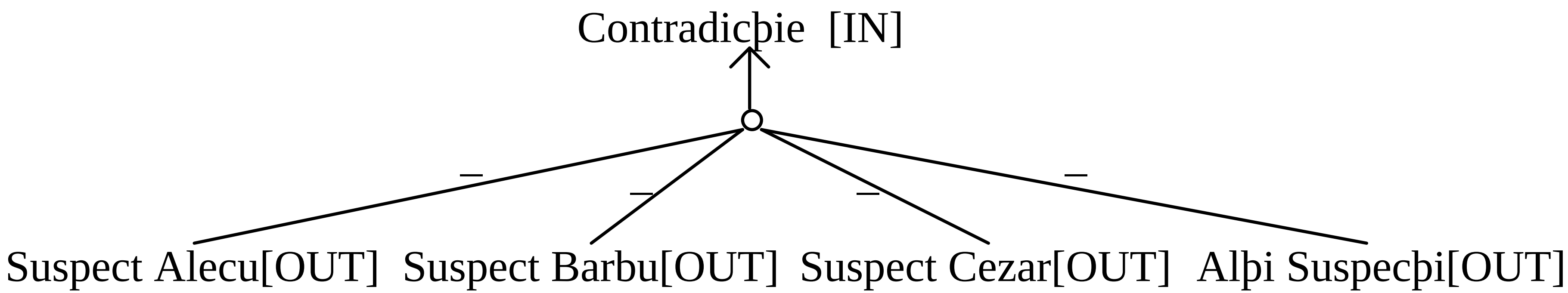
TMS for the ABC Murder

New beliefs are added



TMS for the ABC Murder

The new beliefs bring a contradiction



Removing the contradiction

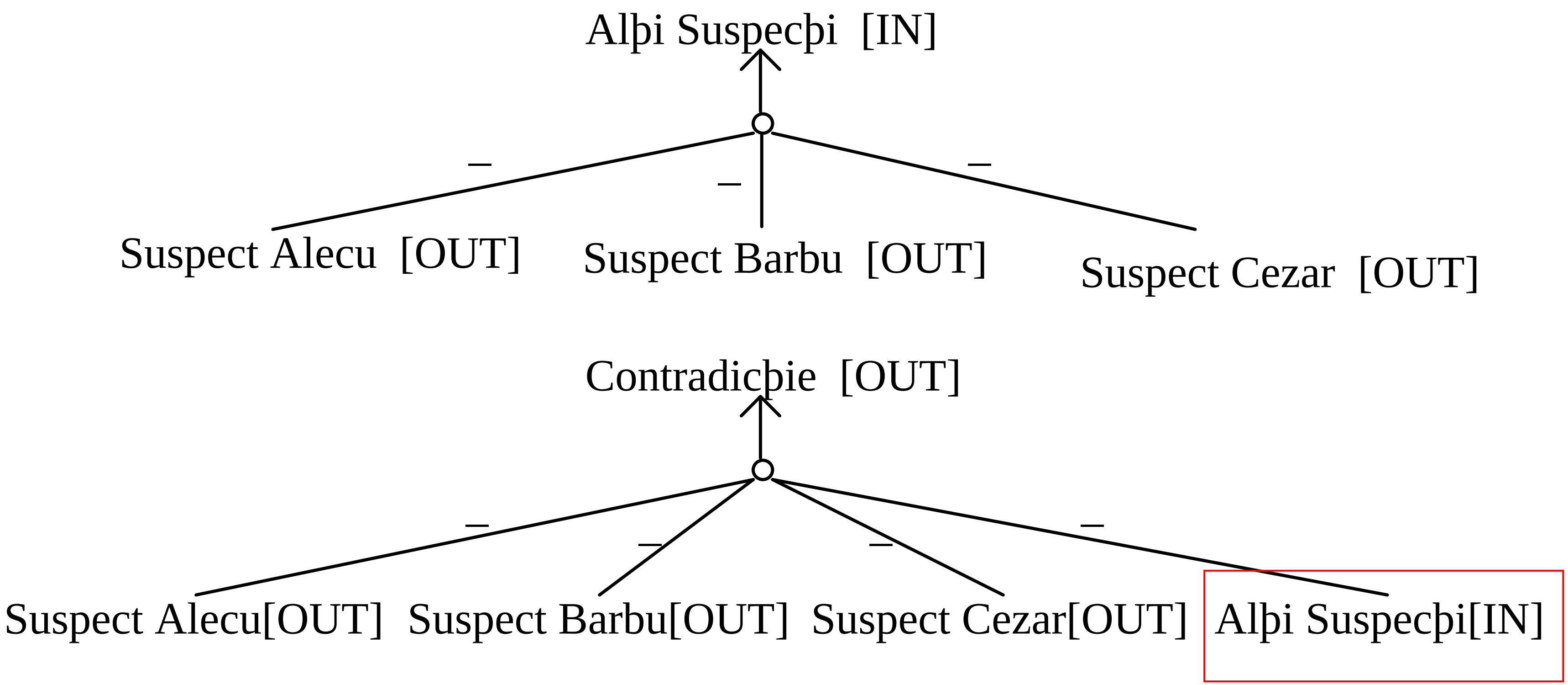
- Identify the minimal set of beliefs that brought the contradiction

- OUT some belief to remove contradiction
  - Select an assumption (node) N and add a valid justification to a node N' ∈ OUT List of N

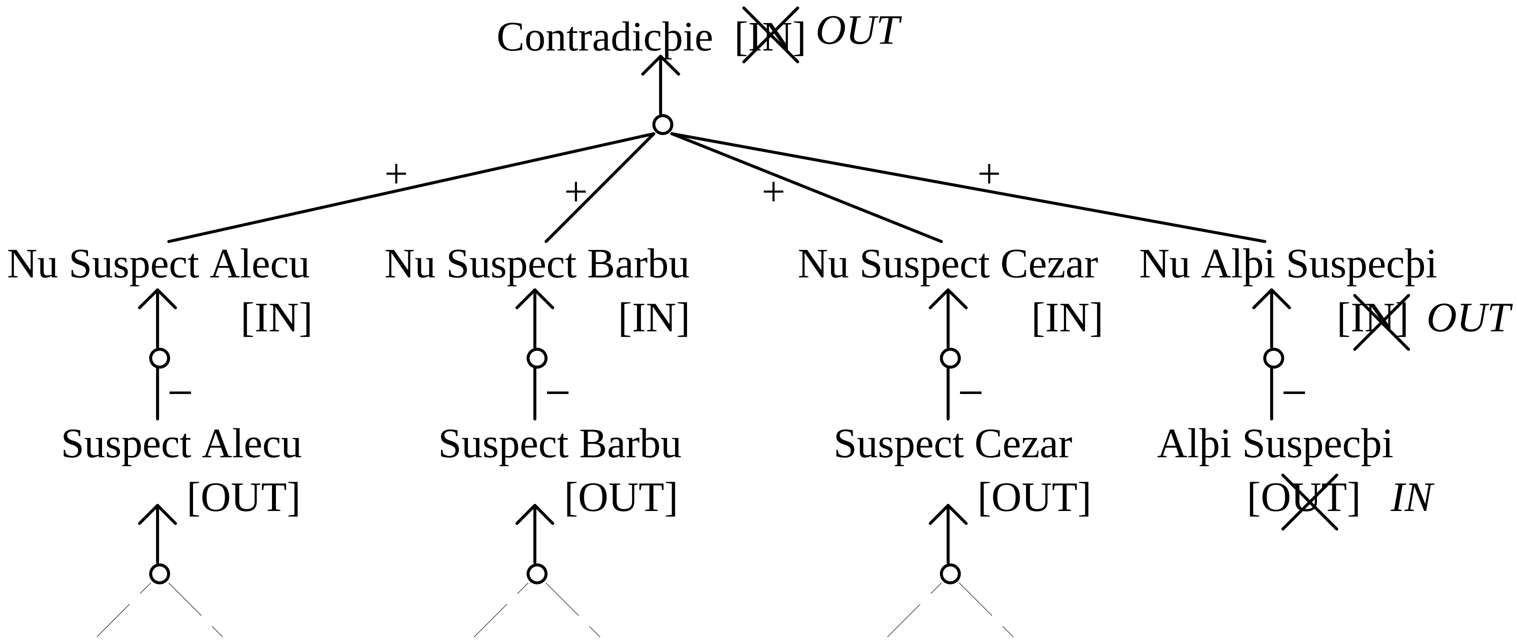
    OR
  - OUT a node N' ∈ IN List of N by adding a valid justification to a node N" ∈ OUT List of N'

Removing the contradiction

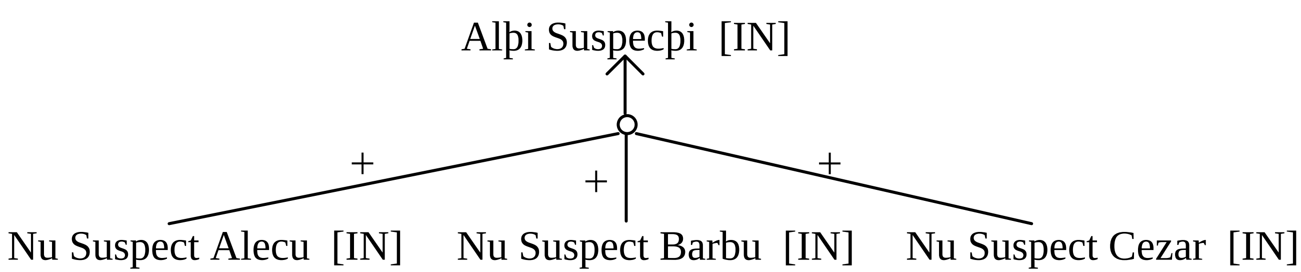
- Justification to remove the contradiction
- {Suspect Alecu, Suspect Barbu, Suspect Cezar, Alti Suspecti}



Removing the contradiction



(a)



(b)

Data structures for a TMS

Data structures

(1) **Node.** Contains the following slots:

- *Value* – the representation of the associated fact; a unique value, which is identical with the representation in the KB
- *Label* – state of the node - IN or OUT.
- *NodeJustification* – list of justifications which justify a given node. Note that a node may have several justifications
- *IsConsequence* – list of justifications in which the node take part. It is formed of 2 lists:
  - *ConsecIN* – list of justifications in which the node appears in the IN list
  - *ConsecOUT* - list of justifications in which the node appears in the OUT list
- *Contradiction* – a flag indicating if the node is a contradiction

Data structures for a TMS

(2) Justification. Contains the following slots:

- *Type* – represents the inference type of a justification, namely premise, Modus Ponens, rule, inheritance, etc. Depends on the Inference Engine and it is given by the IE to the TMS
- *Consequence* – the node the justification justifies
- *Premises* – list of nodes that participated in the inference, formed of the **INList** of nodes and the **OUTList** of nodes

(3) An indexing structure and mechanism to allow fast search of nodes in the TMS