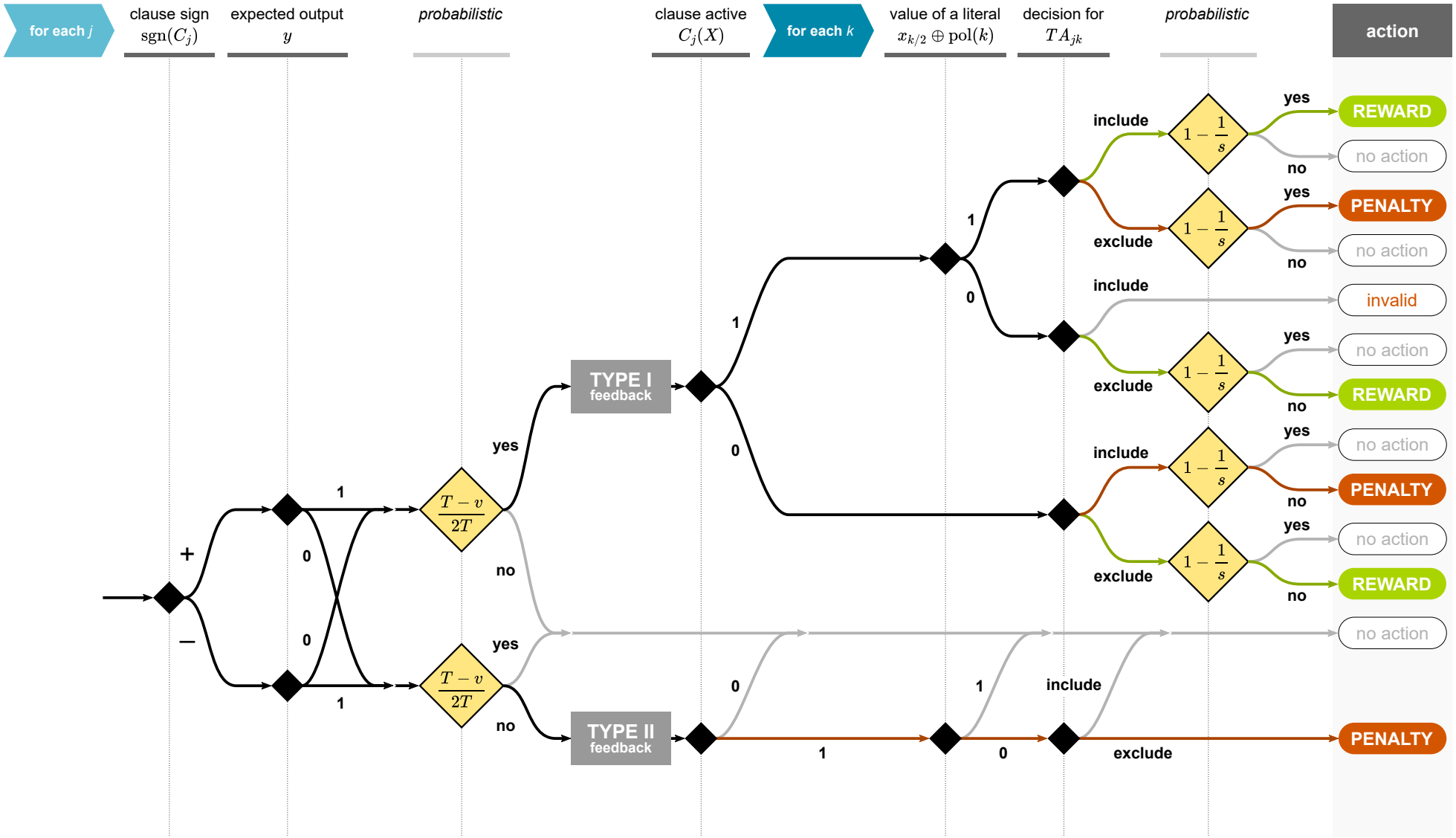
SINGLE-CLASS TSETLIN MACHINE
FEEDBACK DECISION TREE



- The decision tree represents a single-class Tsetlin Machine (TM) for Boolean input vector of size m and consisting of n clauses.
- The decision is calculated for a given input vector $X = (x_0, \ldots, x_{m-1})$ and the expected output y; $y = 1$ if the input vector X matches the class, and $y = 0$ if it does not.
- Each clause C_j for $0 \leq j < n$ has either positive or negative voting defined by the function $\operatorname{sgn}(C_j)$. Usually, even clauses vote $(+1)$, and odd clauses vote (-1).
- *Class confidence* v is the sum of all clause votes respecting their signs:

$$v = \sum_{j=0}^{n-1} C_j(X) \cdot \operatorname{sgn}(C_j).$$

- Type I and Type II feedbacks have different probabilities, which are calculated based on the class confidence v and the *learning threshold* T.
- Each clause activation $C_j(X)$ is calculated as a conjunction of $2m$ literals, one positive and one negative for each of m inputs; i.e., $x_{k/2}$ and $\overline{x}_{k/2}$ for $0 \leq k < 2m$:

$$C_j(X) = \bigwedge_{k=0}^{2m} \left(x_{k/2} \wedge \overline{x}_{k/2}\right).$$

- Positive or negative k-th literal can be expressed as $x_{k/2} \oplus \operatorname{pol}(k)$, where $\operatorname{pol}(k)$ is a literal *polarity* function defined as:

$$\operatorname{pol}(k) = \begin{cases} 0 & \text{if } k \text{ is even (positive literal)} \\ 1 & \text{if } k \text{ is odd (negative literal)} \end{cases}$$

  therefore, the clause activation $C_j(X)$ can also be written as:

$$C_j(X) = \bigwedge_{k=0}^{2m} x_{k/2} \oplus \operatorname{pol}(k).$$

- The state of a single Tsetlin automaton TA_{jk} drives the decision to include or exclude k-th literal in the clause C_j.
- The probability of issuing a feedback action depends on the *learning rate* $s > 1$.

SINGLE-CLASS TSETLIN MACHINE
OPTIMISED DECISION TREE (based on literal feedback)