Errors are Robustly Tamed in Cumulative Knowledge Processes

Anna Brandenberger*, Cassandra Marcussen†,
Elchanan Mossel*, Madhu Sudan†

*MIT, †Harvard University

Outline of this talk

1. Motivation
2. The model
3. Goals of error elimination
4. Main results
5. Proof ideas

Motivating Question

Can societal knowledge accumulation recover from errors?

Examples: Publications, software packages, information on the Web.

How does new knowledge develop? Rely on existing knowledge and form new derivations/code/creation.

The presence of errors: *New units can introduce errors (e.g. if someone proves an incorrect theorem).*

Motivating Question

Can societal knowledge accumulation recover from errors?

Examples: Publications, software packages, information on the Web.

How does new knowledge develop? Rely on existing knowledge and form new derivations/code/creation.

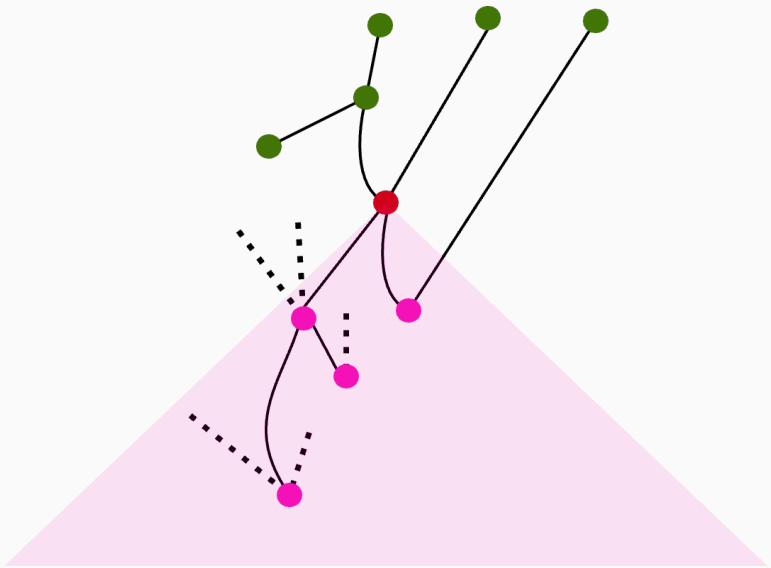
The presence of errors: *New units can introduce errors (e.g. if someone proves an incorrect theorem).*

Connection to learning: In distributed systems that learn (where different computational and AI agents contribute to a corpus of knowledge), errors arise and must be controlled.

Error propagation in knowledge accumulation

If a unit of knowledge is in error, it impacts the correctness of units that rely on it.

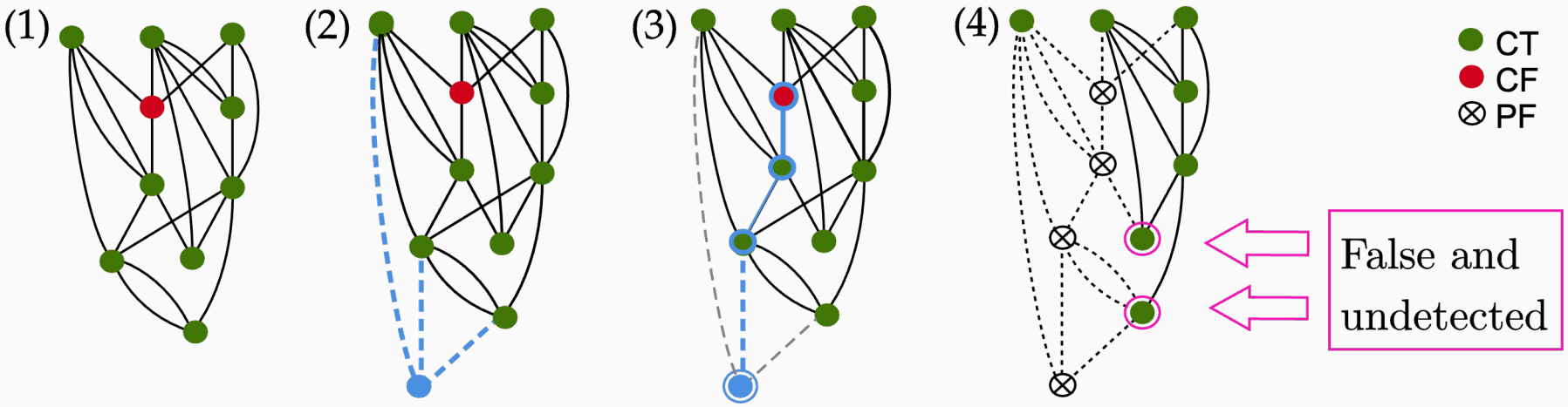
Can natural local checking heuristics stop errors from propagating?



Outline of this talk

1. Motivation
2. The model
3. Goals of error elimination
4. Main results
5. Proof ideas

Modeling knowledge accumulation and errors



Societal knowledge: Sequence of labeled DAGs

State of the process at time t: DAG with labels $\{\{CF, CT\}, PF\}$.

- Labels: CF = conditionally false; CT = conditionally true; PF = proclaimed false.
- Observed: PT = {CF, CT} or PF.
- True = node and all ancestors are CT. False otherwise.

Modeling knowledge accumulation

Ingredients:

1. *Corpus of knowledge:* DAG with edge $u \to v$ if v depends on u.

Modeling knowledge accumulation

Ingredients:

1. *Corpus of knowledge:* DAG with edge $u \to v$ if v depends on u.
2. *Growth model:* One new node joins at each timestep, attaching to existing nodes. Which nodes does it attach to?

Modeling knowledge accumulation

Ingredients:

1. *Corpus of knowledge:* DAG with edge $u \to v$ if v depends on u.
2. *Growth model:* One new node joins at each timestep, attaching to existing nodes. Which nodes does it attach to?
3. *Error introduction:* New nodes may be in error. Random error? Adversarial?

Modeling knowledge accumulation

Ingredients:

1. *Corpus of knowledge:* DAG with edge $u \to v$ if v depends on u.
2. *Growth model:* One new node joins at each timestep, attaching to existing nodes. Which nodes does it attach to?
3. *Error introduction:* New nodes may be in error. Random error? Adversarial?
4. *Error detection:* Check nodes *locally* (in a constant radius), see if they are in error (inspect node to see if it's ● or ⊗), and remove dependent nodes that were checked.

Modeling knowledge accumulation

Ingredients:

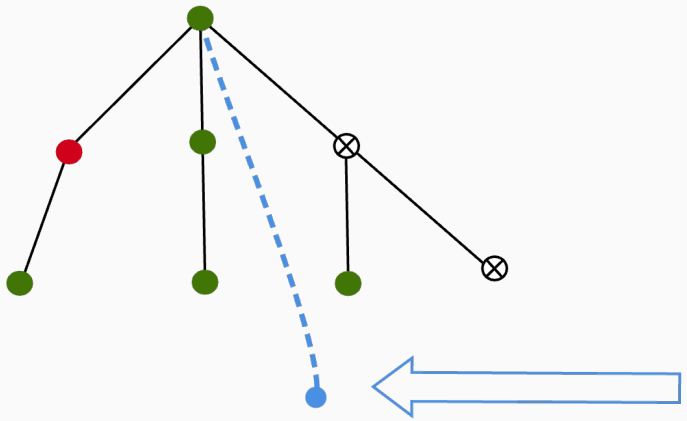
1. *Corpus of knowledge:* DAG with edge $u \to v$ if v depends on u.
2. *Growth model:* One new node joins at each timestep, attaching to existing nodes. Which nodes does it attach to?
3. *Error introduction:* New nodes may be in error. Random error? Adversarial?
4. *Error detection:* Check nodes *locally* (in a constant radius), see if they are in error (inspect node to see if it's ● or ⊗), and remove dependent nodes that were checked.

Robustness goal: Prove results that assume very little about the specifics of the growth model and error placement.

Prior work

"Is this correct? Let's check!" (Ben-Eliezer, Mikulincer, Mossel, Sudan, ITCS 2023): Studied this question in a specific setting.

1. Each new unit connects to *one* existing unit according to a *fixed attachment procedure* (preferential attachment).
2. Random errors.



for $v \in \{$ ●, ● $\}$
connect to v
with probability

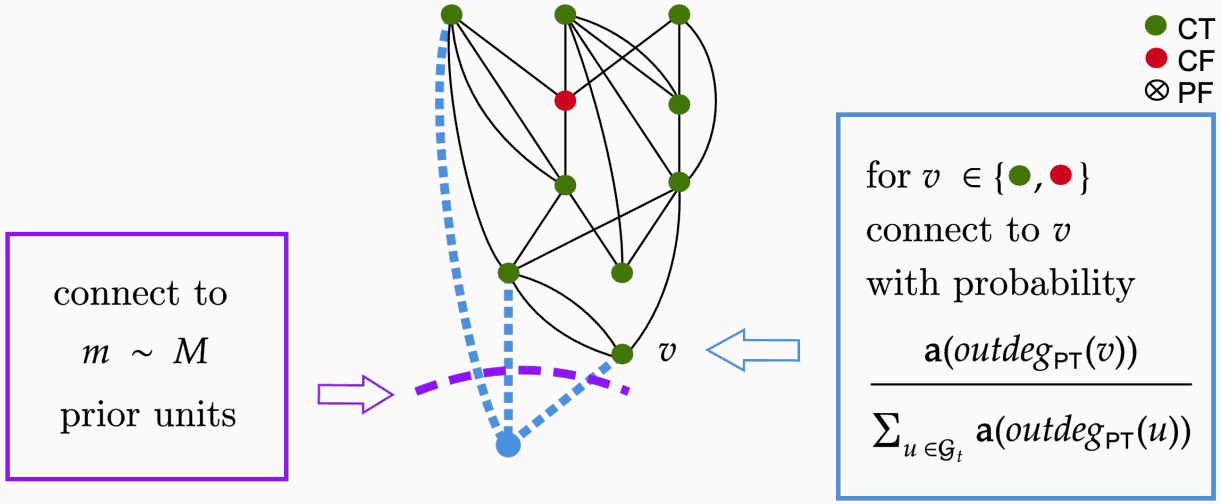
$$\frac{outdeg_{\mathsf{PT}}(v) + 1}{\sum_{u \in \mathcal{G}_t} (outdeg_{\mathsf{PT}}(u) + 1)}$$

● CT
● CF
⊗ PF

Our work: Robust family of processes

Our work: ("Errors are Robustly Tamed in Cumulative Knowledge Processes", COLT 2024)

1. More flexible requirements on growth.



   1.1 Require a to be (b_1, b_2) regular: for all d, $b_1 \leq \mathsf{a}(d+1) - \mathsf{a}(d) \leq b_2$.
   1.2 Regularity encompasses many network models (preferential attachment, uniform attachment).

2. Random *and adversarial* errors.

Outline of this talk

1. Motivation
2. The model
3. Goals of error elimination
4. Main results
5. Proof ideas

Goals: What is achievable?

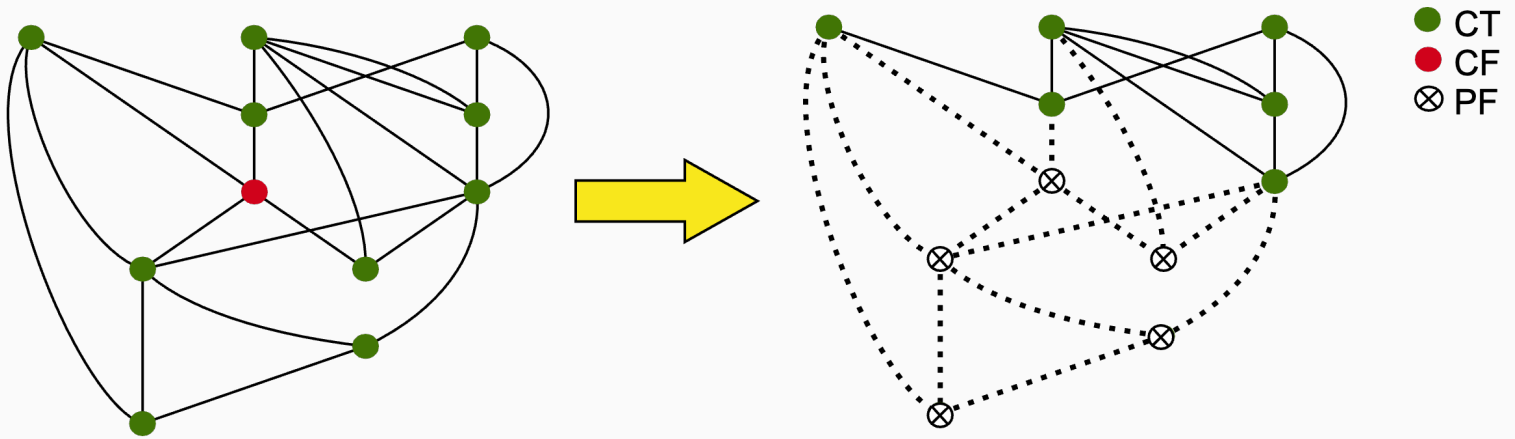
We cannot hope for no errors. Each new node introduces an error with some probability, so errors will always be present in the process.

We *can* guarantee that the effect of each error is eliminated eventually, when checking for errors is sufficiently frequent and extensive.

We also use this to bound the fraction of undetected False nodes in terms of the probability of new errors and the probability of checking.

Error Elimination and Survival

Error elimination: For all CF nodes, there is a time at which everything in the sub-DAG of the node is checked and found to be False (PF).



Error survival: (the opposite) With positive probability, there exists a CF node such that at all times at least one node in its sub-DAG is not detected False (i.e. is CT or CF).

Outline of this talk

1. Motivation
2. The model
3. Goals of error elimination
4. Main results
5. Proof ideas

Main results: Error Elimination

Error Elimination (short version): Under a broad class of CKPs with sufficiently high p and k, natural error elimination heuristics will eliminate all of the errors in the process even when each new unit of knowledge checks at most a constant number of units it depends on.

Error Elimination (long version): For CKPs with the Exhaustive BFS checking mechanism (or other more intensive mechanisms), for every $b \in \mathbb{R}_{\geq 0}$ and $(0, b)$-regular attachment function and every random variable $M \geq 1$, and any adversary bound $r \in \mathbb{Z}_{\geq 0}$, there exists $q_0 > 0$ such that for every low enough adversarial probability $q \leq q_0$, and every error probability $\epsilon < 1$, there exists $p_0 < 1$ and $k_0 \in \mathbb{N}$ such that for all large enough checking parameters $p \geq p_0$ and $k \geq k_0$, error elimination occurs.

Main results: Error Survival

Error Survival (short version): Under a broad class of CKPs, natural error elimination heuristics will *not* eliminate all of the errors in the process when the checks are not performed with high enough probability, for any checking depth.

Error Survival (long version): For CKPs with the Exhaustive BFS checking mechanism (or other less intensive mechanisms), for every $b \in \mathbb{R}_{\geq 0}$ and $(\mathbf{a}(0), b)$-regular attachment function $\mathbf{a}$, and every bounded random variable $M \geq 1$, there exists $q_0, r_0 > 0$ such that for every adversary with $q \leq q_0$ and any $r \leq r_0$, and every error probability $\epsilon > 0$, there exists $p_0 < 1$ such that for any $k \in \mathbb{N}$ and small enough checking parameters $p \leq p_0$, error survival occurs.

Outline of this talk

1. Motivation
2. The model
3. Goals of error elimination
4. Main results
5. Proof ideas

Proof of Error Elimination

For each CF node u, let $\mathcal{G}_t^u$ be the sub-DAG rooted at u.

Proof Idea:

1. Define a potential $\Phi(\mathcal{G}_t^u)$ on $\mathcal{G}_t^u$ such that

$$(\#\text{ of nodes in } \mathcal{G}_t^u) \leq \Phi(\mathcal{G}_t^u).$$

2. Show that $\{\Phi(\mathcal{G}_t^u)\}$ is a positive super-martingale that converges to 0 almost surely.

Proof of Error Elimination: The Potential

Define $\Phi(\mathcal{G}_t^u)$ that captures: "When a new node joins the sub-DAG $\mathcal{G}_t^u$, how far will it be from a detectably-erroneous node?"

Detectably-erroneous: Recall that a check can detect that a node is False if it introduced a new error (●) or was already found to be in error ($\otimes$) by another check.

Proof of Error Elimination: The Potential

Define $\Phi(\mathcal{G}_t^u)$ that captures: "When a new node joins the sub-DAG $\mathcal{G}_t^u$, how far will it be from a detectably-erroneous node?"

Detectably-erroneous: Recall that a check can detect that a node is False if it introduced a new error (●) or was already found to be in error (⊗) by another check.

The potential: Define $\Phi(\mathcal{G}_t^u)$ as:

$$\sum_{v \in \mathcal{G}_t^u} \mathsf{a}(\deg(v)) \cdot \exp\left(\text{distance to CF (●) or PF (⊗) ancestor}\right).$$

Proof of Error Survival

For each CF node u, let $\mathcal{G}_t^u$ be the sub-DAG rooted at u.

Proof Idea:

1. Define a potential $\Phi(\mathcal{G}_t^u)$ on $\mathcal{G}_t^u$ such that

$$(\#\text{ of nodes in } \mathcal{G}_t^u) \geq \Psi(\mathcal{G}_t^u).$$

2. Show that for all t, $\Psi(\mathcal{G}_t^u) > 0$ with positive probability.

Proof of Error Survival: The Potential

Define a potential $\Psi(\mathcal{G}_t)$ that has a positive expected change at each time-step.

$$\Psi(\mathcal{G}_t) = \#\text{CF nodes} + \#\text{leaves}.$$

Thank you!