



HAL
open science

On the Fly Detection of Root Causes from Observed Data with Application to IT Systems

Lei Zan, Charles K Assaad, Emilie Devijver, Eric Gaussier, Ali Aït-Bachir

► **To cite this version:**

Lei Zan, Charles K Assaad, Emilie Devijver, Eric Gaussier, Ali Aït-Bachir. On the Fly Detection of Root Causes from Observed Data with Application to IT Systems. CIKM '24: The 33rd ACM International Conference on Information and Knowledge Management, Oct 2024, Boise, ID, United States. pp.5062-5069, 10.1145/3627673.3680010 . hal-04785797

HAL Id: hal-04785797

<https://hal.science/hal-04785797v1>


Submitted on 20 Nov 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

ON THE FLY DETECTION OF ROOT CAUSES FROM OBSERVED DATA WITH APPLICATION TO IT SYSTEMS

A PREPRINT

 **Lei Zan**

Univ. Grenoble Alpes, CNRS,
Grenoble INP, LIG,
EasyVista
F38000, Grenoble, France

 **Charles K. Assaad**

Sorbonne Université, INSERM,
Institut Pierre Louis d'Epidémiologie
et de Santé Publique,
F75012, Paris, France

 **Emilie Devijver**

Univ. Grenoble Alpes, CNRS,
Grenoble INP, LIG,
F38000, Grenoble, France

 **Eric Gaussier**

Univ. Grenoble Alpes, CNRS,
Grenoble INP, LIG
F38000, Grenoble, France

 **Ali Aït-Bachir**

EasyVista
F38000, Grenoble, France

ABSTRACT

This paper introduces a new structural causal model tailored for representing threshold-based IT systems and presents a new algorithm designed to rapidly detect root causes of anomalies in such systems. When root causes are not causally related, the method is proven to be correct; while an extension is proposed based on the intervention of an agent to relax this assumption. Our algorithm and its agent-based extension leverage causal discovery from offline data and engage in subgraph traversal when encountering new anomalies in online data. Our extensive experiments demonstrate the superior performance of our methods, even when applied to data generated from alternative structural causal models or real IT monitoring data.

1 Introduction

IT monitoring systems are described by metrics, as CPU usage, memory usage, or network traffic, and represented by continuous observational time series. In threshold-based IT monitoring systems, predefined thresholds are used to determine when an anomaly or an alert should be triggered [Ligus, 2013], where the thresholds are set manually or through algorithms leveraging offline (i.e., historical) data [Dani et al., 2015]. In IT systems with multiple interconnected subsystems, several metrics may go into an anomalous state during an incident. In this context, root cause analysis consists on identifying actionable root causes of the anomalies that can be used to resolve the incident. This process is crucial for mitigating impacts like significant financial losses during system outages ?.

Causal graphs can help infer root causes, but obtaining them from experts is often impractical. Causal discovery methods [Spirtes et al., 2000, Assaad et al., 2022a] aim to infer these relations, yet traditional approaches depend on untestable assumptions, need large data sets, and are unsatisfactory for real-world IT monitoring [Aït-Bachir et al., 2023]. This is especially true when causal relations are event-driven rather than continuous.

Understanding causal relations between binary time series offers clearer insights than using raw time series alone. In this paper, we transform raw time series into binary series using thresholds, then discover a causal graph from these binary series, showing causal relationships between threshold crossings. Root causes are then detected using graph traversal techniques. Our contributions are summarized as follows:

1. We propose a structural causal model for anomaly propagation in threshold-based IT monitoring systems, incorporating two distinct noise terms to simulate signal dissipation and external interventions, and providing insight into the event-based nature of anomaly propagation.

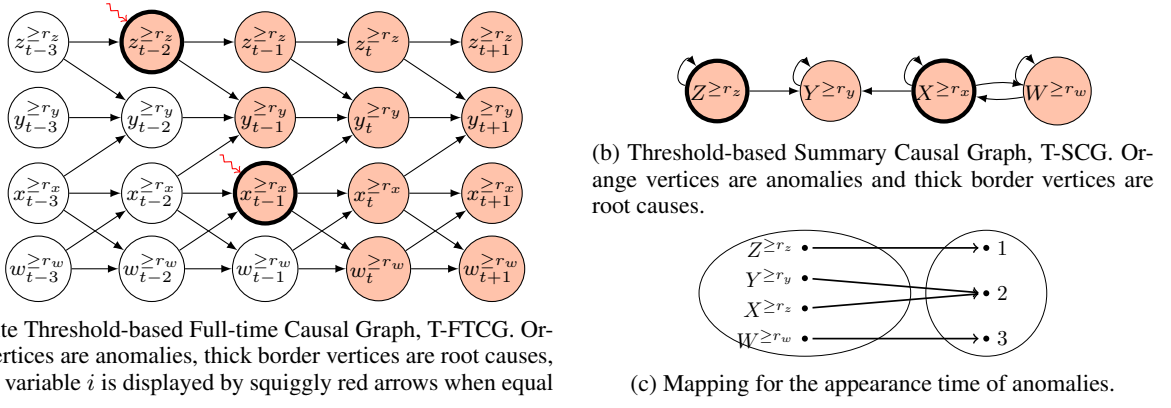


Figure 1: Example. Illustration of (a) a T-FTCG, (b) a T-SCG and (c) the mapping for the appearance time of anomalies on a system with four variables.

2. We introduce the T-RCA framework for detecting root causes, capable of identifying all true root causes under certain assumptions, and provide an empirical solution for scenarios where one assumption is not met.
3. We conduct extensive experiments with synthetic and real data to evaluate the method.

The remainder of this paper is organized as follows: Section 2 discusses related work. Section 3 introduces a new structural causal model for threshold-based IT systems. Section 4 presents an algorithm to detect root causes of anomalies in these systems. In Section 5, the method is compared to others on simulated and real datasets. Finally, Section 6 concludes the paper.

2 State of the Art

Considerable research efforts have recently focused on automated root cause analysis. Some methods utilize causal discovery to uncover causal graphs from anomalous data. For instance, CloudRanger [Wang et al., 2018] employs the PC algorithm [Spirtes et al., 2000] to discover causal graphs among anomalous time series, then identifies root causes using a random walk strategy based on transition matrices calculated from time series correlations. However, the PC algorithm was not originally designed for time series data, and correlations may not fully capture authentic causal effects. MicroCause [Meng et al., 2020] addresses these concerns by utilizing the PCMCI algorithm [Runge et al., 2019] for discovering the causal graph and partial correlations instead of correlations.

Other approaches focus on detecting root causes given the true causal graph of the normal regime. For example, EasyRCA [Assaad et al., 2023] identifies some root causes directly from the graph and detects the rest by comparing direct effects in normal and anomalous regimes. CIRCA [Li et al., 2022] employs a service through a graph structure and conducts a regression hypothesis test on anomalous data to identify deviations. However, these methods require a graph as input, which may not always be available.

In event-driven systems, event-based causal relations may be more useful. Van Houdt et al. [2021] introduced a method, called AITIA-PM, for identifying root causes within event logs [Rudnitckaia, 2016]. This approach leverages a test hypothesis that considers root causes as variables with the highest conditional dependence with anomalies. However, this method is not sound theoretically. Similarly, RCD [Ikram et al., 2022] focuses on discretizing data and identifying root causes through the PC algorithm. However, similarly to CloudRanger and MicroCause, it needs to run the causal discovery algorithm each time an anomaly arises.

From a broader perspective, Wang et al. [2023a] has proposed to model complex systems with interdependent network structures, and detect root causes using hierarchical graph neural networks. Wang et al. [2023b] has used some human feedback in a reinforcement learning fashion to reduce the number of queries. Some methods have also been proposed for inferring root causes of anomalies for non-temporal data, including Budhathoki et al. [2021, 2022]. Those studies are beyond the scope of this paper.

3 Threshold-based causal graphs and root causes

In this section, we present key concepts and assumptions. Lowercase letters represent observed variables, uppercase letters denote name-values or time series, blackboard bold letters indicate sets, and Greek letters represent constants.

We denote $\mathbb{1}_A$ as the indicator function of the event A , \mathcal{G} as a graph, \mathcal{B} the Bernoulli distribution, $\text{Pa}_{\mathcal{G}}(X)$, $\text{An}_{\mathcal{G}}(X)$ and $\text{Desc}_{\mathcal{G}}(X)$ as the sets of parents, ancestors and descendants of a vertex X in \mathcal{G} , respectively.

In IT systems, the data associated to diverse components of the system is commonly gathered in the form of time series.

Definition 1 (Time series). *For $t \in \mathbb{N}$, consider the random variable $x_t \in [0, 1]$. The sequence $\mathcal{X} = \{x_t; t \in \mathbb{N}\}$ is called a discrete time series. Let \mathbb{V} be the set of name-values of d different discrete time series in a system, $\mathbb{T} = \{\mathcal{X} = \{x_t; t \in \mathbb{N}\}; X \in \mathbb{V}\}$ is called a d -dimensional discrete time series.*

In practical scenarios, time series within IT systems are not observed across an infinite set of time points due to operational constraints, such as specific timeframes, resource limits, data storage, and monitoring capabilities with finite recording capacities or designated data collection periods. Subsequently, we consider discrete-time time series with continuous values unless explicitly stated otherwise.

In many IT systems, establishing a causal connection between two time series \mathcal{X} and \mathcal{Y} is not evident at every time step. An anomaly in \mathcal{X} can result in an anomaly in \mathcal{Y} when a specific time point $x_t \in \mathcal{X}$ exceeds a predefined threshold, triggering a corresponding time point $y_t \in \mathcal{Y}$ to breach its own threshold. For example, changes in network traffic may not impact firewall alerts at each time point, but when network traffic surpasses a threshold, it could indicate a security threat, increasing firewall alerts to signal potential intrusion attempts. In such cases, relying solely on raw time series is challenging, requiring binary thresholding for time series.

Definition 2 (Binary thresholding of time series). *Consider a discrete time series $\mathcal{X} = \{x_t; t \in \mathbb{N}\}$ and a fixed threshold $r_x \in [0, 1]$. A binary thresholding of \mathcal{X} is the sequence $\mathcal{X}^{\geq r_x} = \{\mathbb{1}_{x_t \geq r_x}; t \in \mathbb{N}\}$.*

Each binary random variable in the sequence $\mathcal{X}^{\geq r_x}$ is represented as $x_t^{\geq r_x}$, where $t \in \mathbb{N}$, and the binary thresholding of a d -dimensional time series \mathbb{T} is the vector comprising the binary thresholdings of the respective time series. The subsequent definition introduces the concept of a causal graph over binary thresholded d -dimensional time series, illustrated in Figure 1a.

Definition 3 (Threshold-based full-time causal graph, T-FTCG). *Let \mathbb{T} be a d -dimensional time series in a system, \mathbb{V} the set of their name-values, and $r \in \mathbb{R}^d$ a vector of thresholds. A threshold-based full time causal graph $\mathcal{G}_{ft} = (\mathbb{T}^r, \mathbb{E}_{\mathbb{T}}^r)$ is an infinite directed acyclic graph where the set of vertices \mathbb{T}^r corresponds to the set of variables in the binary thresholding of \mathbb{T} and where the set of edges $\mathbb{E}_{\mathbb{T}}^r$ is defined as follows: for two time series $\mathcal{X}, \mathcal{Y} \in \mathbb{T}$, $\forall x_{t-\gamma} \in \mathcal{X}$, and $\forall y_t \in \mathcal{Y}$, $x_{t-\gamma}^{\geq r_x} \rightarrow y_t^{\geq r_y}$ in $\mathbb{E}_{\mathbb{T}}^r$ if and only if $x_{t-\gamma}^{\geq r_x}$ causes $y_t^{\geq r_y}$ at time t with a time lag of $\gamma > 0$ (no instantaneous causal relations).*

To establish a connection between the T-FTCG and the observational data, we adopt the following standard assumptions.

Assumption 1 (Causal Markov condition). *Let $\mathcal{G}_{ft} = (\mathbb{T}^r, \mathbb{E}_{\mathbb{T}}^r)$ be a T-FTCG. For each thresholded time series $\mathcal{X}^{\geq r_x} \in \mathbb{T}^r$, each vertex $x_t^{\geq r_x}$ is independent of its non descendants in \mathcal{G}_{ft} given its parents.*

Assumption 2 (Adjacency faithfulness). *Let $\mathcal{G}_{ft} = (\mathbb{T}^r, \mathbb{E}_{\mathbb{T}}^r)$ be a T-FTCG. Each two adjacent vertices are statistically dependent given any set of vertices.*

While a T-FTCG only illustrates the causes of an effect, the threshold-based dynamic structural causal model (an adaptation of the structural causal models introduced in Pearl [2000] to threshold-based systems) describes how an effect is quantitatively influenced by its causes.

Definition 4 (Threshold-based dynamic structural causal model, T-DSCM). *A threshold-based dynamic structural causal model associated with a T-FTCG $\mathcal{G}_{ft} = (\mathbb{T}^r, \mathbb{E}_{\mathbb{T}}^r)$ is a quadruple $\mathcal{M} = \langle (\mathbb{U}_t, \mathbb{I}_t), \mathbb{T}^r, f, (P(u_t), P(i_t)) \rangle$ where*

1. \mathbb{U}_t and \mathbb{I}_t are two sets of d -dimensional binary background time series (also called exogenous time series), such that $u_t^y \in U_t^y \in \mathbb{U}_t$ and $i_t^y \in I_t^y \in \mathbb{I}_t$ are determined by factors outside the model, for $Y \in \mathbb{V}$ and $t \in \mathbb{N}$;
2. \mathbb{T}^r is a d -dimensional binary observed time series (also called endogenous time series), such that each binary variable $y_t^{\geq r_y} \in \mathcal{Y}^{\geq r_y} \in \mathbb{T}^r$ is determined by variables in the model, for $t \in \mathbb{N}$;
3. for $\mathcal{Y}^{\geq r_y} \in \mathbb{T}^r$ and $t \in \mathbb{N}$, the structural mapping function is given by, for $u_t^y \in U_t^y \in \mathbb{U}_t$ and $i_t^y \in I_t^y \in \mathbb{I}_t$,

$$\begin{aligned} y_t^{\geq r_y} &:= f(\text{Pa}_{\mathcal{G}_{ft}}(y_t^{\geq r_y}), u_t^y, i_t^y) \\ &:= \left(\left(\bigvee_{x_{t-\gamma}^{\geq r_x} \in \text{Pa}_{\mathcal{G}_{ft}}(y_t^{\geq r_y})} x_{t-\gamma}^{\geq r_x} \right) \wedge u_t^y \right) \vee i_t^y; \end{aligned}$$

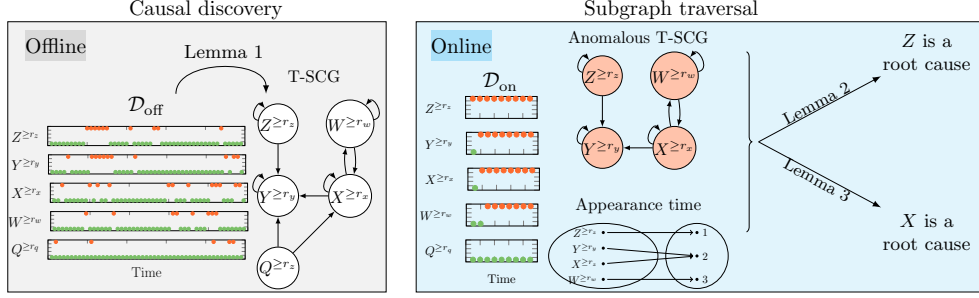


Figure 2: Overview of T-RCA. First step, on the offline dataset: a T-SCG is learned from \mathcal{D}_{off} . Second step, the anomalous T-SCG is deduced from the online dataset, as well as the appearance time. Last step, detection of the root causes using Lemmas 2 and 3.

4. for all $u_t^y \in \mathbb{U}_t^y \in \mathbb{U}_t$, $P(u_t^y) \sim \mathcal{B}(\epsilon_t^y)$ with $\epsilon_t^y \in (0, 1]$ and for all $i_t^y \in \mathbb{I}_t^y \in \mathbb{I}_t$, $P(i_t^y) \sim \mathcal{B}(\beta_t^y)$ with $\beta_t^y \in (0, 1)$.

Note that the background variable $u_t^y \sim \mathcal{B}(\epsilon_t^y)$ denotes the non-immunity of $y_t^{\geq r_y}$: if $u_t^y = 0$, then $y_t^{\geq r_y}$ is unaffected by its parents in the T-FTSCG. When $\epsilon_t^y = 1$, an anomaly will deterministically result in another anomaly. We avoid $\epsilon_t^y = 0$ to maintain the causal relation between $y_t^{\geq r_y}$ and its parents. Similarly, the background variable $i_t^y \sim \mathcal{B}(\beta_t^y)$ represents a hidden cause. If $i_t^y = 1$, then $y_t^{\geq r_y} = 1$ independently of its parents in the T-FTSCG. We avoid $\beta_t^y = 0$ (or $\beta_t^y = 1$) to prevent $y_t^{\geq r_y} = 1$ for all t (or 0).

The T-DSCM relies on the following standard assumption, implying the absence of *unobserved* confounding biases in the system.

Assumption 3. Let $\mathcal{M} = \langle (\mathbb{U}_t, \mathbb{I}_t), \mathbb{T}^r, f, (P(u_t), P(i_t)) \rangle$ be a T-DSCM. We assume that all variables u_t^y and i_t^y of the background time series \mathbb{U}_t and \mathbb{I}_t in \mathcal{M} are jointly independent.

Now we can define anomalies and root causes when considering a T-DSCM associated to a T-FTCG.

Definition 5 (Anomaly and root cause). Let $\mathcal{G}_{ft} = (\mathbb{T}^r, \mathbb{E}_{\mathbb{T}}^r)$ a T-FTCG and $\mathcal{M} = \langle (\mathbb{U}_t, \mathbb{I}_t), \mathbb{T}^r, f, (P(u_t), P(i_t)) \rangle$ its T-DSCM, and a sample satisfying \mathcal{M} . The random variable $x_t^{\geq r_x} \in \mathbb{T}^r$ is said to be anomalous if $x_t^{\geq r_x} = 1$ and a root cause if $i_t^x = 1$.

Graphically, we represent anomalies by orange nodes and root causes by thick borders, as depicted in Figure 1. By definition of a T-DSCM, each variable may be a root cause ($\epsilon_t^y \in (0, 1)$), all root causes are anomalies (if $i_t^y = 1$ then $y_t^{\geq r_y} = 1$), and each anomaly is either propagated from a root cause through a chain of anomalies or is itself a root cause: if $y_t^{\geq r_y} = 1$ either $i_t^y = 1$ or $\exists x_{t-\gamma}^{\geq r_x} \in Pa_{\mathcal{G}_{ft}}(y_t^{\geq r_y})$ such that $x_{t-\gamma}^{\geq r_x} = 1$ and $u_t^y = 1$.

Our main goal is to directly identify root causes from time series data using predefined thresholds. We aim to utilize causal discovery algorithms, leveraging historical data to infer a causal graph, but the T-FTCG is infinite. Under the following condition, it becomes finite, sometimes referred to as a window causal graph [Assaad et al., 2022a].

Assumption 4. Let \mathcal{G}_{ft} be a T-FTCG. All the causal relationships remain constant in direction throughout time in \mathcal{G}_{ft} .

This assumption implies that edges in the T-FTCG remain consistent over time (consistency throughout time). We denote γ_{\max} as the maximal temporal lag between a cause and its effect within the system. Understanding and validating the finite T-FTCG can be challenging for system experts in many applications, and errors may arise for finite sample size and large γ_{\max} , in estimation or when determining the exact lag between a cause and its effect. We introduce an abstraction of the T-FTCG illustrated in Figure 1b.

Definition 6 (Threshold-based Summary Causal Graph, T-SCG). Let \mathbb{V} be the set of name-value of d different time series in a system, and $\mathcal{G}_{ft} = (\mathbb{T}^r, \mathbb{E}_{\mathbb{T}}^r)$ the corresponding T-FTCG. The threshold-based summary causal graph $\mathcal{G} = (\mathbb{V}^r, \mathbb{E}^r)$ associated to \mathcal{G}_{ft} is given by $\mathbb{V}^r := \{X^{\geq r_x} \mid \forall X \in \mathbb{V} \text{ and } r_x \in \mathbb{R}^d\}$ and \mathbb{E}^r such that $X^{\geq r_x} \rightarrow Y^{\geq r_y}$ is in \mathbb{E}^r if and only if there exists $\gamma \in \{1, \dots, \gamma_{\max}\}$ such that $x_{t-\gamma}^{\geq r_x} \rightarrow y_t^{\geq r_y}$ in $\mathbb{E}_{\mathbb{T}}^r$.

Finally, we introduce the appearance time of anomalies, illustrated in Figure 1c.

Definition 7 (Appearance time of anomalies). Given a T-SCG $\mathcal{G} = (\mathbb{V}^r, \mathbb{E}^r)$ and observations of the time series, the appearance time of anomalies is the mapping $\tau : X^{\geq r_x} \mapsto \underset{t}{\operatorname{argmin}} \{x_t^{\geq r_x} = 1\}$.

4 Threshold-based root cause analysis

In IT monitoring systems, our primary focus is typically on identifying the root causes of existing anomalies, specifically those presently occurring. Thus, we differentiate between historical data, denoted as offline data \mathcal{D}_{off} , and current data, referred to as online data \mathcal{D}_{on} . Anomalies and root causes will be sought within the online data \mathcal{D}_{on} .

We propose T-RCA (Threshold-based Root Cause Analysis) for detecting root causes. This method involves causal discovery to uncover the T-SCG from offline data \mathcal{D}_{off} . When anomalies occur in online data \mathcal{D}_{on} , a graph traversal strategy is applied to the inferred T-SCG. An overview is provided in Figure 2. Section 4.1 outlines causal discovery, while Section 4.2 details graph traversal and presents the result that, under an additional assumption, our method identifies root causes precisely. Section 4.3 introduces an extension applicable when this assumption is violated.

4.1 Causal discovery of T-SCG

According to Pearl [1988], under certain assumptions and in the absence of instantaneous relations, it's possible to discover a directed and acyclic causal graph from observational data. We restate this result precisely within the framework of T-FTCG.

Lemma 1. *Let \mathcal{M} be a T-DSCM associated to a T-FTCG \mathcal{G}_{ft} . If Assumptions 1, 2, 3, 4 are satisfied then \mathcal{G}_{ft} is identifiable from the distribution induced by \mathcal{M} .*

In practice, any causal discovery algorithm capable of handling binary time series and aligned with our assumptions can be employed. Once we get the inferred T-FTCG $\hat{\mathcal{G}}_{\text{ft}}$, we can easily deduce the T-SCG $\hat{\mathcal{G}}$ using Definition 6. It is also possible to directly discover the T-SCG (assuming no instantaneous relations) [Assaad et al., 2022b]¹.

4.2 Root cause detection via subgraph traversal

Assuming we have a T-SCG $\mathcal{G} = (\mathbb{V}^r, \mathbb{E}^r)$ and online data \mathcal{D}_{on} containing observed anomalies, our objective is root cause detection. We can construct a colored graph, where vertices are colored if anomalies are present in \mathcal{D}_{on} . These colored vertices are denoted as \mathbb{A} , and $\mathcal{G}_{\mathbb{A}}$ represents the anomalous subgraph containing only these colored vertices. We first proceed by decomposing this graph into strongly connected components.

Definition 8 (Strongly connected component, (SCC)). *Let $\mathcal{G} = (\mathbb{V}^r, \mathbb{E}^r)$ be a T-SCG. A subset $\mathbb{S} \subseteq \mathbb{V}^r$ is a strongly connected component of \mathcal{G} iff \mathbb{S} is a maximal set of vertices where every vertex is reachable via a directed path in \mathcal{G} from every other vertex in \mathbb{S} .*

For instance, in Figure 1b, there are three SCCs: $\{Z^{\geq rz}\}$, $\{Y^{\geq ry}\}$, and $\{X^{\geq rx}, W^{\geq rw}\}$. For each SCC of size 1, we use the following lemma to test if the vertex in the SCC is a root cause.

Lemma 2. *Let $\mathcal{G}_{\mathbb{A}}$ be an anomalous subgraph of a T-SCG $\mathcal{G} = (\mathbb{V}^r, \mathbb{E}^r)$ in \mathcal{D}_{on} . If an anomalous vertex in $\mathcal{G}_{\mathbb{A}}$ does not have any anomalous parent in $\mathcal{G}_{\mathbb{A}}$ then it is a root cause.*

Proof. By Definition 4, the anomaly on a vertex $Y^{\geq ry}$ that does not have any anomalous parent cannot be propagated from other vertices. Thus $i_t^y = 1$ which implies that $Y^{\geq ry}$ is a root cause. \square

This lemma offers a direct method to identify certain root causes. For instance, in Figure 2, it detects $Z^{\geq rz}$ as a root cause.

Furthermore, using the notion of appearance time of anomalies, for each SCC of size greater than 1, we use the following lemma to test if one of the vertices in the SCC is a root cause.

Lemma 3. *Let $\mathcal{G}_{\mathbb{A}}$ be an anomalous subgraph of a T-SCG $\mathcal{G} = (\mathbb{V}^r, \mathbb{E}^r)$ in \mathcal{D}_{on} , τ be the appearance time of anomalies, and \mathbb{S} an SCC in $\mathcal{G}_{\mathbb{A}}$ of size $|\mathbb{S}| > 1$ such that $Pa_{\mathcal{G}_{\mathbb{A}}}(\mathbb{S}) \subseteq \mathbb{S}$. The vertex $\underset{X^{\geq rx} \in \mathbb{S}}{\text{argmin}}\{\tau(X^{\geq rx})\}$ is a root cause.*

Proof. Let \mathbb{S} be an SCC in an anomalous subgraph $\mathcal{G}_{\mathbb{A}}$ such that $|\mathbb{S}| > 1$ and $Pa_{\mathcal{G}_{\mathbb{A}}}(\mathbb{S}) \subseteq \mathbb{S}$. By Definition 4, the anomaly on $Y^{\geq ry}$ satisfying $\underset{X^{\geq rx} \in \mathbb{S}}{\text{argmin}}\{\tau(X^{\geq rx})\}$ cannot be propagated from other vertices, i.e.,

$$\left(\left(\bigvee_{x_{t-\gamma}^{\geq rx} \in Pa_{\mathcal{G}_{\mathbb{A}}}(y_t^{\geq ry})} x_{t-\gamma}^{\geq rx} \right) \wedge u_t^y \right) = 0. \text{ Thus } i_t^y = 1 \text{ which implies that } Y^{\geq ry} \text{ is a root cause. } \square$$

¹The idea proposed in Assaad et al. [2022b] consider a richer type of graph called extended summary causal graph, but when there are no instantaneous relations, the summary causal graph gives the same information as the extended summary causal graph.

In the example in Figure 1, we can deduce from this lemma that $X^{\geq r_x}$ is a root cause since $\tau(X^{\geq r_x}) < \min(\tau(Y^{\geq r_y}), \tau(W^{\geq r_w}))$.

It is noteworthy that the aforementioned lemmas generally do not guarantee the detection of all root causes. Specifically, these lemmas are incapable of detecting a root cause that is influenced by another root cause, i.e., $i_t^y = 1$ and simultaneously $\exists x_{t-\gamma}^{\geq r_x} \in Pa_{G_n}(y_t^{\geq r_y})$ such that $x_{t-\gamma}^{\geq r_x} \wedge u_t^y = 1$. This means that in Figure 1a, if $z_{t-2}^{\geq r_z}$ is a root cause then $y_{t-1}^{\geq r_y}$ cannot be a root cause. We argue that these undetectable root causes are rare in practice, therefore we introduce the following assumption to mitigate their impact.

Assumption 5. Let $\mathcal{G} = (\mathbb{V}^r, \mathbb{E}^r)$ be a T-SCG. We assume that if $X^{\geq r_x}$ is a root cause in \mathcal{D}_{on} , i.e., $\exists t \in \mathcal{D}_{on}$ such that $i_t^x = 1$, then there exists no $Z^{\geq r_z} \in An_{\mathcal{G}}(X^{\geq r_x})$, such that

- $Z^{\geq r_z}$ is a root cause in \mathcal{D}_{on} ; and
- for all $Y^{\geq r_y} \in Desc_{\mathcal{G}}(Z^{\geq r_z}) \cap An_{\mathcal{G}}(X^{\geq r_x}) \setminus \{X^{\geq r_x}, Z^{\geq r_z}\}$, $Y^{\geq r_y}$ is anomalous in \mathcal{D}_{on} .

Generally, a T-FTCG offers a more detailed system representation than a T-SCG. However, due to the absence of instantaneous relations and the timing of anomalies, along with Assumption 5, the T-FTCG doesn't provide an advantage in root cause detection over the T-SCG. The following theorem verifies the validity of T-RCA² under Assumption 5.

Theorem 1. Under Assumptions 1, 2, 3, 4, 5, T-RCA detects exactly the set of true root causes: if \mathbb{C} is the set of true root causes and $\hat{\mathbb{C}}$ is the set of root causes inferred by T-RCA, $\mathbb{C} = \hat{\mathbb{C}}$.

Proof. The soundness' proof (an inferred root cause by T-RCA is a true root cause) is directly given by Lemmas 2 and 3.

Let prove that under Assumption 5, a true root cause is necessarily inferred by T-RCA. Suppose that there exists $C \in \mathbb{C}$ such that C is not detectable by T-RCA. C is not a root vertex in the anomalous subgraph of the T-SCG, otherwise it would have been detected by Lemma 2. C does not belongs to an SCC \mathbb{S} where \mathbb{S} has a size greater than 1 and $Pa_{G_A}(\mathbb{S}) \subseteq \mathbb{S}$, otherwise it would have been detected by Lemma 3. Now, if C is not a root vertex and belongs to an SCC \mathbb{S} of size 1 it violates Assumption 5, since in this case, C must have a parent that is anomalous and is either a root cause or is propagated from a root cause that is an ancestor of C . Similarly, if C is not a root vertex and belongs to an SCC \mathbb{S} of size greater than 1 and $Pa_{G_A}(\mathbb{S}) \not\subseteq \mathbb{S}$, it also violates Assumption 5. Indeed, at least one member of \mathbb{S} has an anomalous parent that is not in \mathbb{S} and that is either a root cause or is propagated from a root cause that is an ancestor of C ; all other members have an anomalous parent in \mathbb{S} that is propagated from a root cause that is an ancestor of C . \square

4.3 Agent-based extension

When Assumption 5 is violated, T-RCA struggles to distinguish root causes among anomalies. To address this, we propose T-RCA-agent, an agent-based extension. T-RCA-agent first runs T-RCA to identify the initial batch of root causes associated with anomalies. An agent then rectifies incidents attributed to these root causes. If anomalies persist, T-RCA-agent reruns T-RCA on the remaining anomalous time series iteratively until no anomalies remain. Unlike T-RCA, T-RCA-agent guarantees root cause detection even when Assumption 5 is violated. Additionally, we theoretically determine the number of iterations needed for termination.

Proposition 1. Let m be the maximum number of root causes on the same directed path in the T-SCG and not satisfying Assumption 5. Under Assumptions 1, 2, 3, 4, T-RCA-agent identifies all root causes after m iterations.

Proof. Using Theorem 1, only the initial root cause of each directed path is identified. Subsequently, after agent intervention, the number of root causes in each path decreases by at least 1 (although multiple interventions may occur simultaneously, T-RCA selects one randomly), maintaining a maximum of $m - 1$ root causes per path. Anomalous variables in \mathcal{D}_{on} are updated accordingly, potentially assigning another root cause as the path's root. By induction, after employing T-RCA m times, all root causes are detected. \square

²The pseudo-code of T-RCA is available in Appendix.

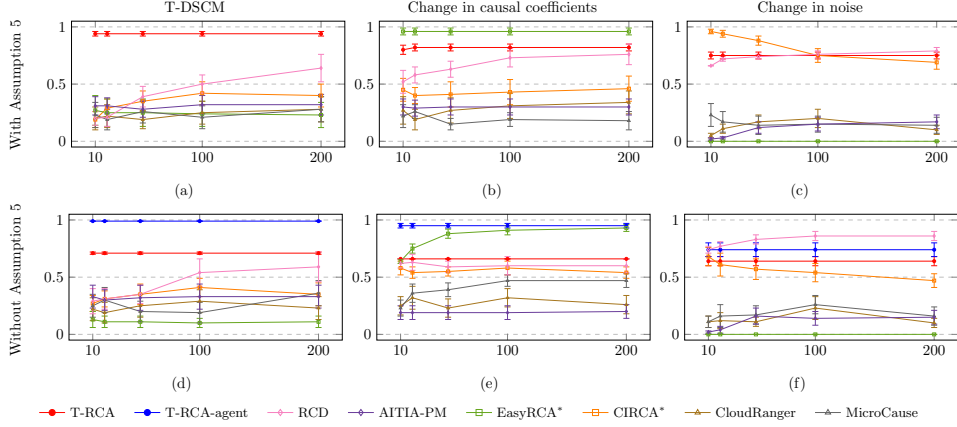


Figure 3: Average F1-score and its variance across 50 simulations are depicted for simulated data. The length of \mathcal{D}_{on} ranges from 10 to 200, generated from a T-DSCM (a and d), a DSCM with root causes experiencing changes in causal coefficients (b and e), and a DSCM with root causes undergoing changes in noise (c and f). Each model is assessed under two settings: one adhering to Assumption 5 (a, b, and c) and another that violates Assumption 5 (d, e, and f).

5 Experiments

In this section, we first describe the experimental setup³, then conduct a thorough analysis using simulated data generated from several models to evaluate our method, and finally we present an analysis using a real-world IT monitoring dataset⁴.

5.1 Experimental setup

T-RCA For causal discovery, we use the PCMCI algorithm [Runge et al., 2019] in which the G-squared test, adapted for binary thresholding of time series, is employed to find conditional independencies.

Baselines We compare our methods with 8 other methods⁵: RCD, CloudRanger, MicroCause, EasyRCA and EasyRCA*, CIRCA and CIRCA* and AITIA-PM. For simulated data, results of EasyRCA and CIRCA are excluded as they require a causal graph input, which we assume unavailable⁶. For RCD, CIRCA*, and AITIA-PM, we select the top two variables from their output ranking as inferred root causes, considering that each case has two genuine root causes. For T-RCA, EasyRCA*, and MicroCause, we set the maximum lag γ_{max} to 1, and maintain a fixed significance level of 0.01 across all methods. For EasyRCA*, CloudRanger, and MicroCause, a Fisher-z-test is utilized due to their use of raw (continuous) data as input. Additionally, in the case of CloudRanger and MicroCause, the walk length is set to 1000, and the backward step threshold is fixed to 0.1. Default values are used for all other hyperparameters.

Evaluation Accuracy of root cause detection is measured by the F1-score between the real causes \mathbb{C} and the inferred set $\hat{\mathbb{C}}$:

$$F1\text{-score}(\mathbb{C}, \hat{\mathbb{C}}) = \frac{2 \sum_{c \in \mathbb{C}} \mathbb{1}_{c \in \hat{\mathbb{C}}}}{2 \sum_{c \in \mathbb{C}} \mathbb{1}_{c \in \hat{\mathbb{C}}} + \sum_{\hat{c} \in \hat{\mathbb{C}}} \mathbb{1}_{\hat{c} \notin \mathbb{C}} + \sum_{c \in \mathbb{C}} \mathbb{1}_{c \notin \hat{\mathbb{C}}}}$$

5.2 Simulated data

We explore three settings, each with distinct data generation processes. The first setting assesses method effectiveness by generating data from a T-DSCM, while the second and third settings evaluate method robustness using a different model. Within each setting, we examine two cases: one where Assumption 5 holds and another where it’s violated. In the latter case, we introduce T-RCA-agent and set the number of iterations equal to the maximum number of genuine root causes in an active path. If T-RCA-agent identifies a genuine root cause, anomalous variables in \mathcal{D}_{on} are adjusted

³Experiments on computational time and robustness can be found in Appendix.

⁴Python code of our method and experiments: <https://github.com/leizan/T-RCA>.

⁵We consider a version of EasyRCA and CIRCA where we learn the graph from normal offline data using PCMCI, denoted as EasyRCA* [Assaad et al., 2023] and CIRCA*, respectively.

⁶These results can be found in Appendix.

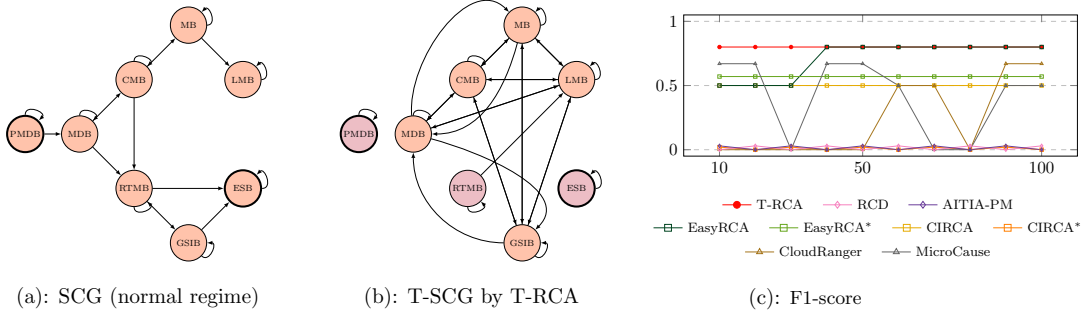


Figure 4: Real IT monitoring data: (a) the SCG provided by the experts, on the normal regime, where root causes correspond to the vertices with thick borders (PMDb and ESB); (b) the T-SCG learned by T-RCA, where inferred root causes correspond to purple vertices (PMDb and ESB); (c) F1-score for the IT monitoring data, varying the lengths of \mathcal{D}_{on} from 10 to 100.

accordingly, simulating actions taken by system engineers. Across all settings, \mathcal{D}_{off} is 20,000 units long, while we vary the lengths of \mathcal{D}_{on} in $\{10, 20, 50, 100, 200\}$.

5.2.1 Threshold-based system

The first data generation process is based on the T-DSCM outlined in Definition 4. We randomly generate 50 T-SCGs, each with 6 vertices, a maximal degree between 4 and 5, and exactly one root vertex. All lags in the associated T-FTCGs are set to 1. Subsequently, we generate one dataset for each T-SCG. Each time series is scaled to $[0, 1]$, with thresholds randomly selected from $U([0.7, 0.9])$. Every time series includes a self-cause, and with a probability of 0.3, $\epsilon_t^y < 1$, indicating anomalies may not always trigger in children. For \mathcal{D}_{off} , interventions are applied to each normal variable with a probability $\beta_t^y = 0.1$. We ensure no time series remains anomalous for more than 5 consecutive time points. In \mathcal{D}_{on} , if Assumption 5 holds, two vertices on the same active path are randomly chosen; otherwise, two vertices on different active paths are selected.

Results depicting the means and variances of the F1-score for each method are presented in Figure 3(a) and (d), with and without Assumption 5 respectively, where the values of thresholds are presumed to be known. Notably, when Assumption 5 holds, T-RCA consistently outperforms other methods, exhibiting low variance. The performance of T-RCA remains stable since it only requires knowledge of the anomalous variables in \mathcal{D}_{on} . However, in the other case, the performance of T-RCA declines due to the violation of Assumption 5, while T-RCA-agent demonstrates superior performance. All other methods have very low performances. Note that, as the length of \mathcal{D}_{on} increases, the performance of RCD improves.

5.2.2 Non-threshold based system with changes in causal coefficients

To evaluate the robustness of our method, we utilize a dataset introduced by Assaad et al. [2023] comprising 50 different Acyclic Summary Causal Graphs with self-causes. All lags are set to 1. \mathcal{D}_{off} is generated using the DSCM as $y_t = \sum_{x_{t-1} \in Pa_{\mathcal{G}_{ft}}(y_t)} a x_{t-1} + 0.1 \xi_t^y$, with $a \sim U([0.1, 1])$, $\xi_t^y \sim \mathcal{N}(0, 1)$, y_t denotes the value of the vertex y at time t , $Pa_{\mathcal{G}_{ft}}(y_t)$ denotes the direct parents of y_t in the FTCTG. For \mathcal{D}_{on} , similar to the previous setting, two different strategies are used to randomly select two vertices in each graph for intervention, which are considered as the root causes. The intervention changes the coefficients from all parents of the intervened variable by resampling them from $U([0.1, 1])$. The effect of each intervention propagates through the generating process to all the descendants of the intervened vertex. Thresholds for each time series are chosen empirically such that each variable in \mathcal{D}_{off} contains 90% of data below the threshold and 10% above it. In this setting, T-RCA and T-RCA-agent utilize thresholds to learn the T-SCG from \mathcal{D}_{off} . The selected root causes and their descendants in the graph are considered as anomalous variables, and this information is utilized by our proposed method, EasyRCA*, CloudRanger, and MicroCause.

Results depicting the means and variances of the F1-score for each method are presented in Figure 3 (b) and (e) for both cases. When Assumption 5 holds, EasyRCA* performs well with low variance, as the data generating process aligns with its settings. The performance of T-RCA follows EasyRCA*, outperforming other methods. In the other case, T-RCA suffers, while the performance of EasyRCA* increases along with the expansion of the length of \mathcal{D}_{on} . Notably, T-RCA-agent exhibits a comparable performance with EasyRCA*, and even better for a small length of \mathcal{D}_{on} .

5.2.3 Non-threshold based system with changes in noise

Finally, we utilize a dataset simulating a microservice architecture from the DoWhy package⁷. The graph consists of 11 vertices, and a lag of 1 is assumed between each pair of vertices to simulate time series data. Similar to the previous setting, two different strategies are used to randomly select two vertices in the graph for intervention. We generate 50 datasets for each case. Interventions are applied following Scenario 3 in the provided link, which involves shifting the value by a constant. Thresholds for each time series are determined by using the empirical strategy described earlier.

Results depicting the means and variances of the F1-score for each method are presented in Figure 3 (c) and (f) for both cases. When Assumption 5 holds, CIRCA* exhibits good performance with a small size of \mathcal{D}_{on} . However, its performance declines as \mathcal{D}_{on} increases. T-RCA maintains performance comparable to RCD. When Assumption 5 is violated, RCD outperforms other methods once \mathcal{D}_{on} exceeds 20, followed by T-RCA-agent, T-RCA, and CIRCA*. These methods clearly outperform others in both cases.

5.3 Real IT monitoring data

The dataset, provided by EasyVista and introduced in Assaad et al. [2023], consists of eight time series collected from an IT monitoring system with a one-minute sampling rate, as described in Assaad et al. [2023], Aït-Bachir et al. [2023].

For T-RCA and AITIA-PM methods, which rely on anomalies in historical data for root cause detection, consider all data preceding the onset of anomalies as \mathcal{D}_{off} , comprising more than 40,000 data points. However, for methods needing only normal data as \mathcal{D}_{off} , we use 1,000 data points from the normal part before anomalies. We vary the length of \mathcal{D}_{on} from 10 to 100 with a step of 10. Thresholds for each time series are selected such that each variable in \mathcal{D}_{off} contains 90% of data below the threshold and 10% above it. Each time series is anomalous within \mathcal{D}_{on} . For CIRCA, we consider only one ultra-metric, *Saturation*. Due to the inability to simulate system engineer interventions, T-RCA-agent is not part of the comparison.

The true T-SCG associated with this system is unknown, but EasyVista’s system experts have described the summary causal graph in the normal regime (when there are no anomalies), which is provided in Figure 4(a). Thus, for methods needing a graph, such as CIRCA and EasyRCA, the summary causal graph in the normal regime serves as the input graph. PMDB and ESB are expected to be the root causes of the anomalies. The T-SCG learned by T-RCA is given in Figure 4(b). The graph is denser than the SCG stated in Figure 4(a), but they do not encode the same information. Most importantly, the genuine root causes are discovered.

The F1-score of each method is presented in Figure 4(c). T-RCA consistently performs well, achieving an F1-score of 0.8, similar to EasyRCA which knows the SCG in the normal regime, when the length of \mathcal{D}_{on} exceeds 30. CIRCA and EasyRCA* exhibit reasonable performance. Meanwhile, the result also exhibits the importance of the correct causal graph for the method CIRCA and EasyRCA. The performance of CloudRanger and MicroCause varies considerably, influenced by random mechanisms in their respective methods.

6 Conclusion

We introduced a novel structural causal model tailored for representing threshold-based IT systems and presented the T-RCA algorithm for detecting root causes of anomalies within such systems. In its basic form, T-RCA assumes the absence of more than one root cause in an active path aligned with the derived causal graph. Additionally, we introduced an optimized agent-based extension of T-RCA, relaxing this assumption while necessitating minimal system interventions. Our experiments showcased the superiority of our methods, particularly on data generated from the specified structural causal model, but also on data from alternative models.

For future work, it would be interesting to consider instantaneous relations. In this case, having richer graphs, which are inherently acyclic, can offer advantages, especially when anomalies form a cycle in the T-SCG. Additionally, relaxing the assumption of joint independence among variables in \mathbb{U}_t and \mathbb{I}_t could provide further insights. Finally, exploring solutions that alleviate Assumption 5 without requiring external actions is of interest.

Acknowledgements

This work was partially supported by EasyVista, by MIAI@Grenoble Alpes (ANR-19-P3IA-0003), and by the CI-PHOD project (ANR-23-CPJ1-0212-01).

⁷https://www.pywhy.org/dowhy/v0.11.1/example_notebooks/gcm_rca_microservice_architecture.html

References

- C. K. Assaad, E. Devijver, and E. Gaussier. Survey and evaluation of causal discovery methods for time series. *Journal of Artificial Intelligence Research*, 73:767–819, feb 2022a.
- C. K. Assaad, E. Devijver, and E. Gaussier. Discovery of extended summary graphs in time series. In J. Cussens and K. Zhang, editors, *Proceedings of the Thirty-Eighth Conference on Uncertainty in Artificial Intelligence*, volume 180 of *Proceedings of Machine Learning Research*, pages 96–106. PMLR, 01–05 Aug 2022b.
- C. K. Assaad, I. Ez-zejjari, and L. Zan. Root cause identification for collective anomalies in time series given an acyclic summary causal graph with loops. In F. Ruiz, editor, *Proceedings of the Twenty-Sixth Conference on Artificial Intelligence and Statistics*, Proceedings of Machine Learning Research. PMLR, April 2023.
- A. Ait-Bachir, C. K. Assaad, C. de Bignicourt, E. Devijver, S. Ferreira, E. Gaussier, H. Mohanna, and L. Zan. Case studies of causal discovery from it monitoring time series. submitted, 2023.
- K. Budhathoki, D. Janzing, P. Bloebaum, and H. Ng. Why did the distribution change? In A. Banerjee and K. Fukumizu, editors, *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 1666–1674. PMLR, 13–15 Apr 2021.
- K. Budhathoki, L. Minorics, P. Bloebaum, and D. Janzing. Causal structure-based root cause analysis of outliers. In K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 2357–2369. PMLR, 17–23 Jul 2022.
- M.-C. Dani, F.-X. Jollois, M. Nadif, and C. Freixo. Adaptive threshold for anomaly detection using time series segmentation. In *Neural Information Processing: 22nd International Conference, ICONIP 2015, Istanbul, Turkey, November 9-12, 2015, Proceedings Part III 22*, pages 82–89. Springer, 2015.
- A. Ikram, S. Chakraborty, S. Mitra, S. Saini, S. Bagchi, and M. Kocaoglu. Root cause analysis of failures in microservices through causal discovery. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 31158–31170. Curran Associates, Inc., 2022.
- M. Li, Z. Li, K. Yin, X. Nie, W. Zhang, K. Sui, and D. Pei. Causal inference-based root cause analysis for online service systems with intervention recognition. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD ’22*, page 3230–3240, New York, NY, USA, 2022. Association for Computing Machinery. ISBN 9781450393850. doi: 10.1145/3534678.3539041.
- S. Ligus. *Effective Monitoring and Alerting*. O’Reilly, 2013. ISBN 9781449333522.
- Y. Meng, S. Zhang, Y. Sun, R. Zhang, Z. Hu, Y. Zhang, C. Jia, Z. Wang, and D. Pei. Localizing failure root causes in a microservice through causality inference. In *2020 IEEE/ACM 28th International Symposium on Quality of Service (IWQoS)*, pages 1–10, 2020. doi: 10.1109/IWQoS49365.2020.9213058.
- J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.
- J. Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, New York, NY, USA, 2000. ISBN 0-521-77362-8.
- J. Rudnitckaia. Process mining. data science in action. *University of Technology, Faculty of Information Technology*, pages 1–11, 2016.
- J. Runge, P. Nowack, M. Kretschmer, S. Flaxman, and D. Sejdinovic. Detecting and quantifying causal associations in large nonlinear time series datasets. *Science Advances*, 5(11):eaau4996, 2019. doi: 10.1126/sciadv.aau4996.
- P. Spirtes, C. N. Glymour, R. Scheines, and D. Heckerman. *Causation, prediction, and search*. MIT press, 2000.
- G. Van Houdt, B. Depaire, and N. Martin. Root cause analysis in process mining with probabilistic temporal logic. In *International Conference on Process Mining*, pages 73–84. Springer, 2021.
- D. Wang, Z. Chen, J. Ni, L. Tong, Z. Wang, Y. Fu, and H. Chen. Interdependent causal networks for root cause localization. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD ’23*, page 5051–5060, New York, NY, USA, 2023a. Association for Computing Machinery.
- L. Wang, C. Zhang, R. Ding, Y. Xu, Q. Chen, W. Zou, Q. Chen, M. Zhang, X. Gao, H. Fan, S. Rajmohan, Q. Lin, and D. Zhang. Root cause analysis for microservice systems via hierarchical reinforcement learning from human feedback. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD ’23*, page 5116–5125, New York, NY, USA, 2023b. Association for Computing Machinery.
- P. Wang, J. Xu, M. Ma, W. Lin, D. Pan, Y. Wang, and P. Chen. Cloudranger: Root cause identification for cloud native systems. In *2018 18th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing*, pages 492–502. IEEE, 2018.

A Pseudo-code of T-RCA

The pseudo-code of T-RCA is presented in Algorithm 1. The algorithm outlined begins at line 1 by converting d -dimensional observational time series from two datasets, \mathcal{D}_{off} and \mathcal{D}_{on} , into binary thresholded series using predefined thresholds (\mathbb{R}^d). In line 2, using an event-based causal discovery algorithm on \mathcal{D}_{off} , a T-SCG is constructed. In line 3 and 4, from \mathcal{D}_{on} , anomalous vertices are identified and their times of appearance are mapped, employing Definitions 5 and 7 respectively. In line 5, the algorithm initializes an empty set for root causes, $\hat{\mathbb{C}}$. In lines 6-11, the algorithm analyzes each SCC within each anomalous subgraph. If all parents of an SCC are contained within the same SCC and it consists of a single vertex, that vertex is added to $\hat{\mathbb{C}}$ as per Lemma 2. For SCCs containing multiple vertices, the vertex with the earliest appearance time of anomaly is selected for $\hat{\mathbb{C}}$.

Algorithm 1: T-RCA

Data: Two datasets \mathcal{D}_{off} and \mathcal{D}_{on} of d -dimensional observational time series \mathbb{T} , maximum lag γ_{max} , set of thresholds for each time series \mathbb{R}^d

Result: Set of root causes $\hat{\mathbb{C}}$

- 1 Provide the binary thresholded time series according to thresholds \mathbb{R}^d
 - 2 Discover the T-SCG $\hat{\mathcal{G}} = (\mathbb{V}^r, \mathbb{E}^r)$ using an event-based causal discovery algorithm on \mathcal{D}_{off} and \mathbb{R}^d // Lemma 1
 - 3 Deduce the set of anomalous vertices $\mathbb{A} \subseteq \mathbb{V}^r$ from \mathcal{D}_{on} using Definition 5
 - 4 Deduce the mapping of appearance time of anomalies τ from $\hat{\mathcal{G}}$ and \mathcal{D}_{on} using Definition 7
 - 5 $\hat{\mathbb{C}} = \emptyset$
 - 6 **foreach** SCC $\mathbb{S} \in \hat{\mathcal{G}}_{\mathbb{A}}$ **do**
 - 7 **if** $Pa_{\hat{\mathcal{G}}_{\mathbb{A}}}(\mathbb{S}) \subseteq \mathbb{S}$ **then**
 - 8 **if** $size(\mathbb{S}) = 1$ **then**
 - 9 $\hat{\mathbb{C}} = \hat{\mathbb{C}} \cup \mathbb{S}$ // Lemma 2
 - 10 **else**
 - 11 $\hat{\mathbb{C}} = \hat{\mathbb{C}} \cup \{X^{\geq r_x}\}$ such that $X^{\geq r_x} \in \mathbb{S}$ and $\tau(X^{\geq r_x}) < \tau(Y^{\geq r_y})$ for all $Y^{\geq r_y} \in \mathbb{S} \setminus \{X^{\geq r_x}\}$
// Lemma 3
-

B Examples of Insufficiency of correlation and time

It might appear that under Assumption 5, the time of anomaly and/or the association (possibly conditional) between anomalies might be sufficient to detect root causes. We provide two examples showing that this is, in fact, not true, highlighting the crucial role of the causal discovery step.

Example 1 (Insufficiency of time and dependence). *Consider a scenario involving three time series, where the underlying T-SCG among them are depicted in Figure 5 (a). The lag between X and Y is 1, while the lag between X and Z is 2. Assuming $\epsilon_t^y < 1$ and ignoring self-causes, multiple interventions are applied to X at different time steps to induce anomalies.*

When identifying the root cause(s) of $Z^{\geq r_z} = 1$ (denoted as $Z_{=1}^{\geq r_z}$) solely based on temporal information, both $X_{=1}^{\geq r_x}$ and $Y_{=1}^{\geq r_y}$ are included, as they occur before $Z_{=1}^{\geq r_z}$. Similarly, solely relying on dependence also points to both $X_{=1}^{\geq r_x}$ and $Y_{=1}^{\geq r_y}$ as root causes, given their connections to $Z_{=1}^{\geq r_z}$. Even when considering temporal and dependence information together, $Y_{=1}^{\geq r_y}$ remains identified as the root cause.

Example 2 (Insufficiency of time and conditional dependence on a single variable). *Consider another scenario involving four time series, where the underlying T-SCG among them is depicted in Figure 5 (b). The lag between W , X and Z , X is 1, while the lag between W , Y and Z , Y is 2. Assuming $\epsilon_t^y < 1$ (i.e., uncertainty in anomaly propagation) and ignoring self-causes, multiple interventions are applied independently on W and Z at different time steps to induce anomalies. Here, we aim to detect all root causes of $Y^{\geq r_y} = 1$ (denoted as $Y_{=1}^{\geq r_y}$).*

Firstly, using temporal order, $W_{=1}^{\geq r_w}$, $X_{=1}^{\geq r_x}$ and $Z_{=1}^{\geq r_z}$ are considered as potential causes of $Y_{=1}^{\geq r_y}$ because they precede it.



Figure 5: (a) demonstrates a scenario where time and dependence fail to detect all root causes. (b) demonstrates a scenario where time and conditional dependence on a single variable fail to detect all root causes.

Then, using dependence conditioning on a single variable cannot guarantee the detection of root causes. For example, when conditioning on $W_{=1}^{\ge r_w}$ (resp. $Z_{=1}^{\ge r_z}$), $X_{=1}^{\ge r_x}$ becomes dependent on $Y_{=1}^{\ge r_y}$, because of $Z_{=1}^{\ge r_z}$ (resp. $W_{=1}^{\ge r_w}$). Consequently, $X_{=1}^{\ge r_x}$ is incorrectly considered a root cause.

Even using a special measure that sums over multiple conditional dependence, we will not be able to guarantee the detection of root causes: as previously, between $X_{=1}^{\ge r_x}$ and $Y_{=1}^{\ge r_y}$, even though we do a sum of multiple terms of conditional dependence, but within each term, we only condition on one variable, which will not break the dependence between these two variables.

Furthermore, even if we search for the highest conditional dependence, as done in AITIA-PM [Van Houdt et al., 2021], there is no guarantee to find all root causes. For instance, consider again the T-SCG in Figure 5 (b), but now assume $\epsilon_t^y = 1$. Then, the special measure (used by AITIA-PM) that sums over multiple conditional dependence between $Y_{=1}^{\ge r_y}$ and $X_{=1}^{\ge r_x}$ is :

$$\begin{aligned} & \{ \Pr(Y_{=1}^{\ge r_y} | X_{=1}^{\ge r_x} \wedge W_{=1}^{\ge r_w}) - \Pr(Y_{=1}^{\ge r_y} | X_{=0}^{\ge r_x} \wedge W_{=1}^{\ge r_w}) \\ & + \Pr(Y_{=1}^{\ge r_y} | X_{=1}^{\ge r_x} \wedge Z_{=1}^{\ge r_z}) - \Pr(Y_{=1}^{\ge r_y} | X_{=0}^{\ge r_x} \wedge Z_{=1}^{\ge r_z}) \} = 2 \end{aligned}$$

which is higher than the same special measure between $Y_{=1}^{\ge r_y}$ and $Z_{=1}^{\ge r_z}$ equal to:

$$\begin{aligned} & \{ \Pr(Y_{=1}^{\ge r_y} | W_{=1}^{\ge r_w} \wedge Z_{=1}^{\ge r_z}) - \Pr(Y_{=1}^{\ge r_y} | W_{=0}^{\ge r_w} \wedge Z_{=1}^{\ge r_z}) \\ & + \Pr(Y_{=1}^{\ge r_y} | W_{=1}^{\ge r_w} \wedge X_{=1}^{\ge r_x}) - \Pr(Y_{=1}^{\ge r_y} | W_{=0}^{\ge r_w} \wedge X_{=1}^{\ge r_x}) \} = 0. \end{aligned}$$

C Complementary experiments

C.1 Extension of Section 5.2 - varying the length of \mathcal{D}_{on}

In Figure 6, we present the results of the means and variances of the F1-score for each method, considering a larger length of \mathcal{D}_{on} up to 2,000 samples. These results align with the three settings delineated in Section 5.2. In addition, in this analysis, we include the results of EasyRCA and CIRCA, which require a causal graph as input. In general, EasyRCA shows comparable performance to T-RCA but falls slightly behind T-RCA-agent. The good performance of EasyRCA is not surprising since it relies on an input graph. Nevertheless, it is noteworthy that T-RCA consistently surpasses CIRCA in performance, which was unexpected.

C.2 Extension of Section 5.2.1 - new data generating process

In addition to the setting discussed in Section 5.2.1, we explore three other settings based on T-DSCM, varying the lengths of the online set, \mathcal{D}_{on} , from 10 to 2,000 samples. Firstly, we distinguish if self-causes are considered for each vertex in the T-SCG or not (denoted as $Y_{t-1} \in Pa_G(Y_t)$ or $Y_{t-1} \notin Pa_G(Y_t)$). Secondly, we distinguish $\epsilon_t^y = 1$ (i.e., certainty in anomaly propagation, referred to as $\epsilon_t^y = 1$) and $\epsilon_t^y < 1$ (i.e., uncertainty in anomaly propagation, referred to as $\epsilon_t^y = 1 - 0.3^{|Pa_{G_n}(Y_t) \cap \mathbb{A}|}$). The results are presented in Figure 7 which shows that when Assumption 5 is satisfied, overall T-RCA performs best followed by EasyRCA and RCD. When Assumption 5 is violated, T-RCA-agent performs best followed by T-RCA when $y_{t-1} \in Pa_{G_{f_t}}(y_t)$ and $\epsilon_t^y = 1$ and by RCD in the other cases.

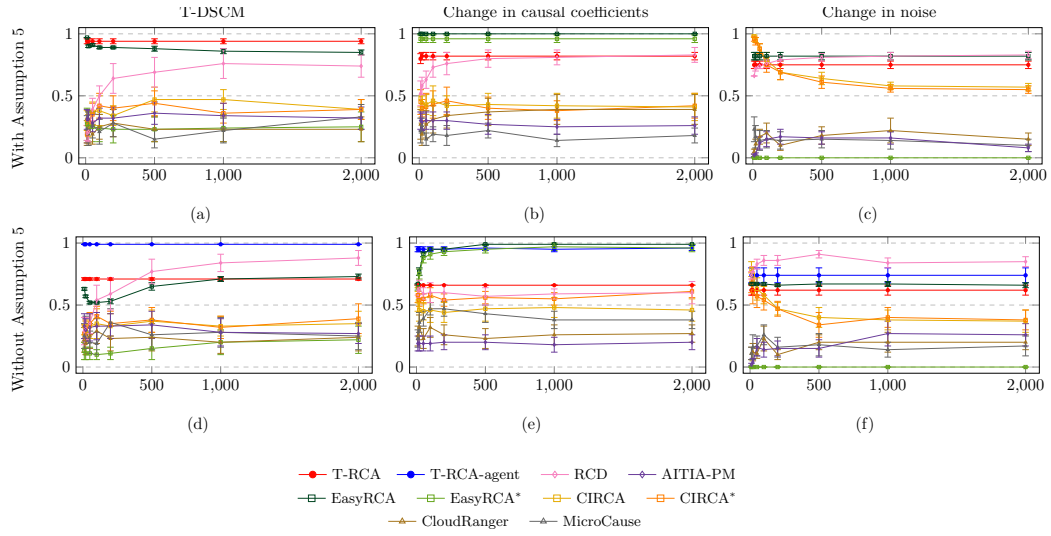


Figure 6: Mean of F1-score averaged over 50 simulations, and associated variance, for the three data generating processes, the lengths of \mathcal{D}_{on} varying from 10 to 2,000.

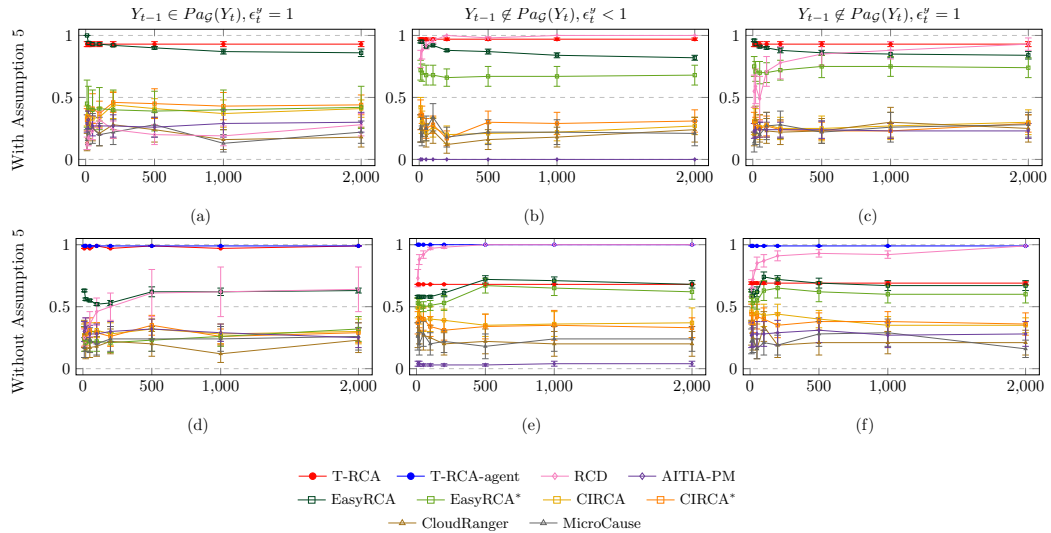


Figure 7: Mean of F1-score averaged over 50 simulations, and associated variance, for other three settings based on T-DSCM, the lengths of \mathcal{D}_{on} varying from 10 to 2,000.

C.3 Execution time analysis

Based on the setting discussed in Section 5.2.1, where Assumption 5 holds, the execution time of each method is analyzed with \mathcal{D}_{on} lengths varying within the range of [50, 500, 2000]. Additionally, the online part of T-RCA (T-RCA_{online}) and the offline part of T-RCA (T-RCA_{offline}) are analyzed separately.

The mean of the logarithm of execution time (plus 1) in seconds, averaged over 50 simulations, along with the associated variance for each method, is presented in Figure 8. The results indicate that the execution time of the offline part of T-RCA (T-RCA_{offline}) is the fastest among methods that need to reconstruct a graph based on \mathcal{D}_{off} . In contrast, the execution time of the online part of T-RCA (T-RCA_{online}) is negligible.

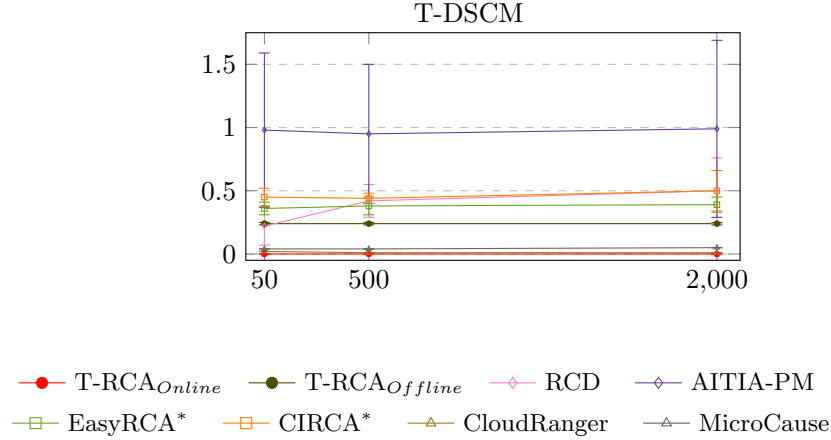


Figure 8: Mean of the logarithm of execution time (plus 1) in seconds, averaged over 50 simulations, along with the associated variance, is presented for the data-generating processes based on T-DSCM under Assumption 5. The analysis separately considers the online part of T-RCA (T-RCA_{online}) and the offline part of T-RCA (T-RCA_{offline}), with \mathcal{D}_{on} lengths varying within the range of [50, 500, 2000].

C.4 Robustness according to the choice of the threshold- equal thresholds

C.4.1 Simulated data

For our proposed methods, T-RCA and T-RCA-agent, thresholds are required for binarizing each time series. Here, we aim to investigate how changes in thresholds impact the performance of our proposed methods across the same cases outlined in Section 5.2. We vary the threshold for each variable by controlling the proportion of data below this threshold in \mathcal{D}_{off} from 0.8 to 0.98 in steps of 0.02. Specifically, a proportion of 0.8 implies that the threshold for each time series is chosen to ensure that 80% of data in \mathcal{D}_{off} are smaller than this threshold.

In the setting of T-DSCM, results depicting the means and variances of the F1-scores are presented in Figure 9 (a and d), where the dashed line illustrates the performance of the method when the thresholds are correctly chosen for each time series. However, for the other two settings, true thresholds for the time series do not exist, hence there are no dashed lines in the corresponding results. The solid line illustrates the performance of the method with varying thresholds, which does not surpass the dashed line for each method. This is because, during the data generating process, the threshold is randomly chosen for each time series, potentially resulting in different thresholds in practice. However, for simplicity, we adopt the same rule to choose the threshold for each time series, which does not guarantee that the threshold for each variable is properly chosen. It is worth noting that in the setting based on T-DSCM, correctly chosen thresholds aid our proposed method in detecting root causes. As the relationship between two variables in this setting depends on thresholds, choosing thresholds higher than the correct one reduces examples for the relations where anomaly causes anomaly, thereby decreasing the performance of our method. Similarly, choosing thresholds lower than the correct one includes examples that do not accurately represent the relations where anomaly causes anomaly, also reducing the performance of our method. Overall, in this setting, the performance of our proposed methods fluctuates with changes in thresholds, but generally, the variance remains within an acceptable range. Practically, the threshold value associated with the highest performance of our proposed methods can serve as a reference for selecting the optimal threshold to monitor the time series.

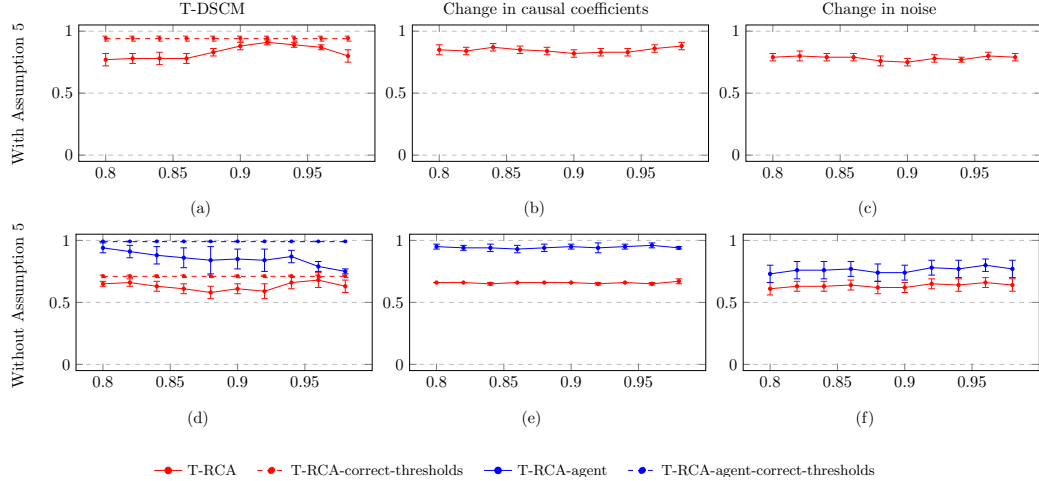


Figure 9: Mean of the F1-score, averaged over 50 simulations, and associated variance, for the three data generating processes of our proposed methods. We vary the threshold for each time series by controlling the proportion of data smaller than this threshold in \mathcal{D}_{off} from 0.8 to 0.98 in steps of 0.02. The dashed line represents the performance of the methods when the thresholds of time series are correctly chosen

For settings corresponding to Section 5.2.2 and Section 5.2.3, results depicting the means and variances of the F1-scores are presented in Figure 9 (b and e) and Figure 9 (c and f), respectively. In these settings, the performance of our proposed methods remains consistent with low variance when varying the thresholds, as the causal mechanisms in these two settings are continuous.

From the results above, we can conclude that if the causal mechanism is based on thresholds, such as event-based relations, misspecification of the threshold for time series will degrade the performance of our proposed method. However, when the causal mechanism is continuous, misspecification of the threshold for time series will not significantly impact the performance of our proposed method.

C.4.2 Real IT monitoring data

Similarly, we vary the threshold for each time series by controlling the proportion of data smaller than this threshold in \mathcal{D}_{off} from 0.8 to 0.98 in steps of 0.01 on real IT monitoring data, and the F1-score is shown in Figure 10.

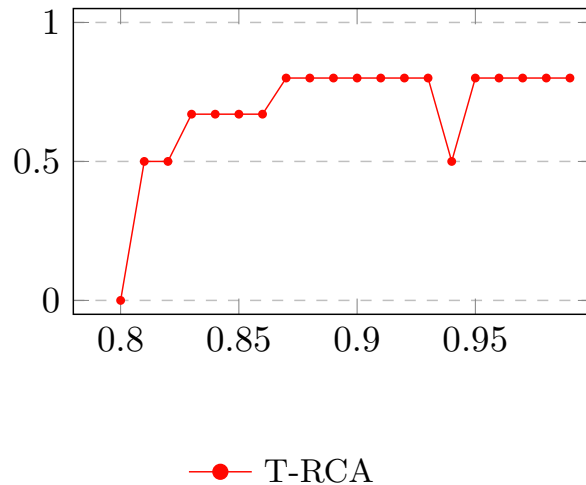


Figure 10: F1-score on Real IT monitoring data for our method, varying the threshold for each time series by controlling the proportion of data smaller than this threshold in \mathcal{D}_{off} from 0.8 to 0.98 in steps of 0.01.

When thresholds are chosen much lower than the correct one, examples that inaccurately represent the relations where anomaly causes anomaly are included, leading to a decline in the performance of T-RCA. Then, with the increasing of thresholds, the performance of T-RCA tends to increase. Except, at the point where the proportion is 0.94, the performance of the method exhibits significant variance compared to nearby points, as we adopt the same rule to choose the threshold for each time series for simplicity. However, in practice, the correct threshold for each time series may vary. The result also confirms, on the other hand, that our system is threshold-based.

C.5 Robustness according to the choice of the threshold - different thresholds

C.5.1 Simulated data

We also selected an alternative strategy to test the robustness of our method by varying the thresholds. Specifically, we vary the thresholds for each time series by offsetting them from the correct thresholds, ranging from -1 to 1 in steps of 0.01.

The results given in Figure 11 which shows that when the specified threshold is a higher than the true threshold then T-RCA (and T-RCA-agent) manage to keep a good performance. This means that when a system expert is hesitating between two different values (but not significantly different) for the threshold, it is better to choose the higher one.

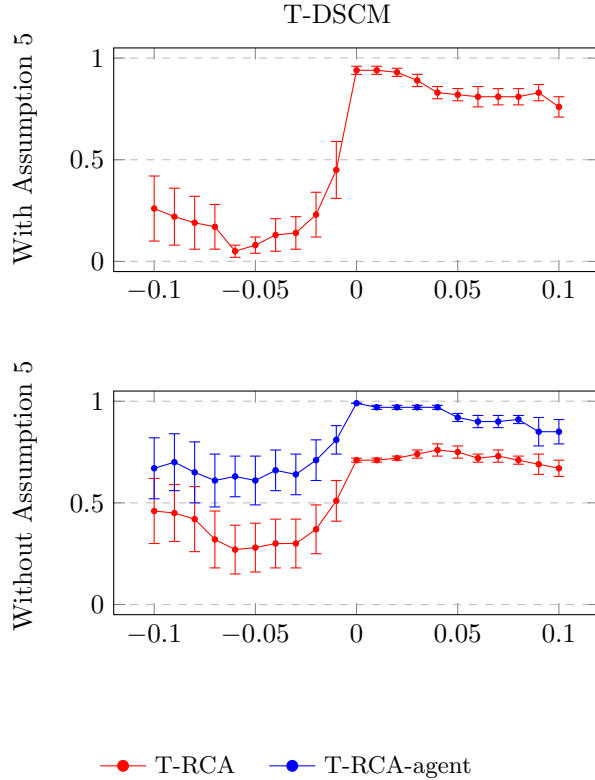


Figure 11: Mean of the F1-score, averaged over 50 simulations, and associated variance, for the data generating processes based on T-DSCM of our proposed methods. We varied the thresholds for each time series by offsetting them from the correct thresholds, ranging from -1 to 1 in steps of 0.01.

C.5.2 Real IT monitoring data

Since we do not have the true thresholds for each time series, we start here by fine-tuning the thresholds that gives the best results then we perform the same analysis that was done for simulated data.

Fine-tuning thresholds To fine-tune the thresholds we tested randomly several threshold and selected the one that gives the highest F1-score.

Surprisingly, we were able to find several sets of thresholds for which T-RCA can identify the root causes with an F1-score equals to 1. This might confirms, that our system is threshold-based.

Varying thresholds with respect to the fine-tuned thresholds Here we choose the thresholds that correspond to the smallest values for which T-RCA can identify the root causes with an F1-score equals to 1.

As before, we vary the thresholds for all time series by iteratively decreasing the fine-tuned thresholds by 0.01 or increasing them by 0.01, repeated five times on each side. However, an exception is made for the time series RTMB, where we decrease and increase the thresholds by 0.0001. This exception is warranted as all values of this time series are smaller than 0.01.

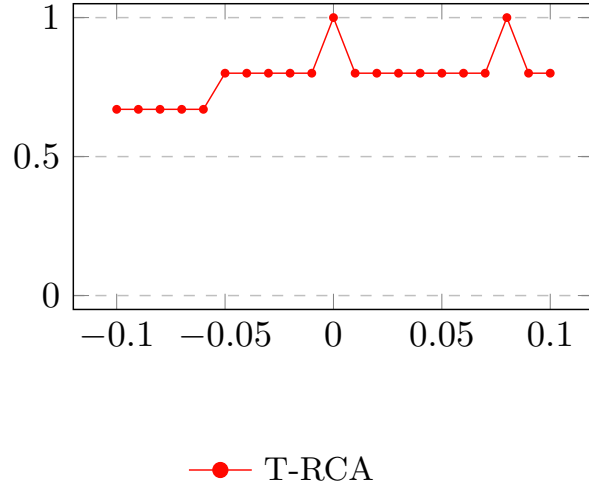


Figure 12: F1-score on Real IT monitoring data for our method, varying the threshold for each time series by offsetting them from the fine-tuning thresholds. We varied the thresholds from -1 to 1 in steps of 0.01. Specifically, for RTMB, due to its small values, we adjusted its threshold from -0.001 to 0.001 in steps of 0.0001.

The results given in Figure 12, again which again shows that if a system expert is undecided between two slightly different threshold values, it is better to opt for the higher one.