2020 IEEE 36th International Conference on Data Engineering (ICDE)

Kronos: Lightweight Knowledge-based Event Analysis in Cyber-Physical Data Streams

Mohammad Hossein Namaki*, Xin Zhang†, Sukhjinder Singh*, Arman Ahmed*
Armina Foroutan*, Yinghui Wu‡, Anurag K. Srivastava* Anton Kocheturov§

*Washington State University †University of California, Riverside
‡Case Western Reserve University ‡Siemens Corporation Technology
*{mnamaki, sukhjindersingh2, arman.ahmed, s.foroutan, anurag.k.srivastava}@wsu.edu
†xzhan261@ucr.edu ‡yxw1650@case.edu §anton.kocheturov@siemens.com

***Abstract*—We demonstrate Kronos, a framework and system that automatically extracts highly dynamic knowledge for complex event analysis in Cyber-Physical systems. Kronos captures events with anomaly-based event model, and integrates various events by correlating with their temporal associations in real-time, from heterogeneous, continuous cyber-physical measurement data streams. It maintains a lightweight highly dynamic knowledge base, enabled by online, window-based ensemble learning and incremental association analysis for event detection and linkage, respectively. These algorithms incur time costs determined by available memory, independent of the size of streams. Exploiting the highly dynamic knowledge, Kronos supports a rich set of stream event analytical queries including event search (keywords and query-by-example), provenance queries ("which measurements or features are responsible for detected events?"), and root cause analysis. We demonstrate how the GUI of Kronos interacts with users to support both continuous and ad-hoc queries online and enables situational awareness in Cyber-power systems, communication, and traffic networks.**

I. Introduction

Event-driven operational decision making in cyber-physical systems (CPS) such as smart grids [10], cloud services [4], and sensor networks requires real-time detection of complex events. These events are often jointly characterized by multiple richly attributed signals (anomalies), their spatio-temporal correlations, and additional contextual and environmental factors. A missing capacity in current data stream processing system is to support automatic extraction, integration and search for context-rich events with semantic knowledge from heterogeneous sensor data streams. This remains to be a main barrier for effective interpretation and transformation of event analysis to actionable knowledge in various CPS workforce.

Example 1: On September 8, 2011, a system disturbance occurred in Arizona, leading to cascading outages that affected $2.7M$ people in Arizona and Southern California for around 12 hours. The loss of a single transmission line initiated the event but was not the only cause of this huge blackout [2]. Phasor Measurement Units (PMUs) were introduced to power grids with the ability to capture time-coherent measurements across a geographically distributed area. PMUs are able to measure voltage and current, frequency, and change rate of frequency ranging from 10-60 samples per second [10]. These synchronized and real-time measurements is a step toward situational awareness. The precise and timely estimation of the fault location can avoid time-consuming and manual examination and restoration.

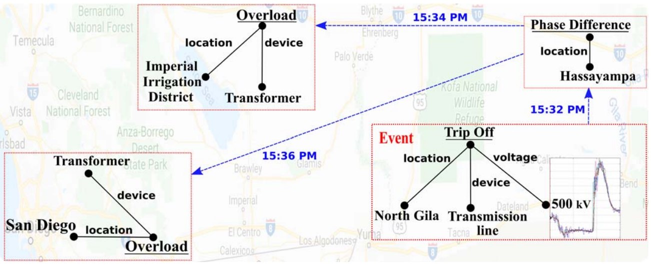


Fig. 1. Highly dynamic knowledge representation of a power outage as a chain of four cyber-power anomalies.

Fig. 1 shows a faction of regions that were affected by the 2011 blackout as well as sensors *e.g.,* PMUs, meters and weather data that captured various spatio-temporal information. It also shows a fraction of the knowledge graph extracted from physical events occurred in this outage. The nodes are events and edges are either event properties (in black) or causality relationships among events (in blue). The graph shows tripping off a 500 kilo-Voltage transmission line at North Gila led to Phase difference in Hassayampa that caused two transformers overload at two other locations. □

Despite that primitive indicators of the above power outage, events can be captured by anomaly detection [1] in e.g., PMU measurement data streams, a holistic detection of the entire multi-phase, multi-layer events requires dynamic linking of multiple anomalies modeled by *e.g.,* knowledge graphs as complex event model [9]. The need to integrate highly dynamic knowledge from data streams for holistic event analysis is evident in various applications.

Real-time knowledge inference for event detection is also evident in the following scenarios. 1) Traffic flow management and accident prevention requires fast detection of transportation events and their spatio-temporal interactions from traffic data streams [5]. 2) To increase the revenue and guarantee the service level agreements (SLAs) in cloud-based services, events that violate SLAs, their cascading effects and associated environmental factors should be recognized in time. 3) Malicious activities coincide with each other to form multi-

2375-026X/20/$31.00 ©2020 IEEE
DOI 10.1109/ICDE48307.2020.00165

phase power cyberattacks, such as masked attacks [7] that uses DDoS as decoy to distract defense efforts from true intrusion. Deriving cyber-physical events and their temporal associations as highly dynamic knowledge suggest useful defense and enable root-cause analysis in complex systems.

Kronos. Kronos is a light-weight knowledge extractor for anomaly-based complex event analysis in data streams. It has the following unique features that differ from current systems.

Lightweight & Flexible Event Model. Kronos serves as a general event model to support fast extraction of highly dynamic knowledge. It does not make assumption on the underlying streaming data and can be utilized in various anomaly-based event models. It adopts (a) a class of lightweight, primitive "star-shaped" entity model, which can be easily synthesized to more complex events, and (b) an ensemble-based event detection that supports the "plug-in" base detectors registered by users. This enables Kronos to (1) describe various complex events using primitive entities as building blocks, (2) adapt to extract user-defined events by allowing users to register anomaly detectors without additional manual effort.

Online Knowledge Extraction. Kronos uses a package of window-based anomaly detection and link inference algorithms to maintain the dynamic knowledge graph. The algorithms are optimized with incremental mining, multi-threading and sampling techniques to ensure the performance under tunable response time and memory constraints.

Online Event Analytical Queries. Kronos supports various analytical queries including keyword search [8] to only explore interesting events based on the users' requirements, provenance-queries to find out the root-cause of events [11], spatio-temporal queries to ask which events occurred in a time range and a specific region, and example-based queries to find similar events to the given entities [6].

Visual Exploration. Kronos provides user-friendly panels for users to configure the analysis session, to easily formulate queries, to monitor the streaming data, and to validate the events and relationships by exploring the extracted knowledge.

Below we give an overview of Kronos (Section II), presenting its key enabling techniques including real-time event extraction (Section II-C), online relation discovery (Section II-D), supported query classes (Section II-E), and its architecture (Section II-F). We will demonstrate each component of Kronos and show how it supports anomaly analysis of streaming data via an interactive user-interface (Section III). We will also demonstrate use cases including real-world Cyber-power event analysis, intrusion detection, and traffic control.

II. System Overview

We start with the knowledge model used by Kronos.

A. Lightweight Knowledge Model

Data streams. Kronos adopts window-based data stream model and it reads multiple streams $\mathcal{S}$. A measurement data stream is a pair (S, W), where data stream $S \in \mathcal{S}$ is an infinite sequence of tuples $\{d_1, d_2, \ldots\}$, and W is a sliding window that caches the latest $|W|$ tuples in S with tunable size.

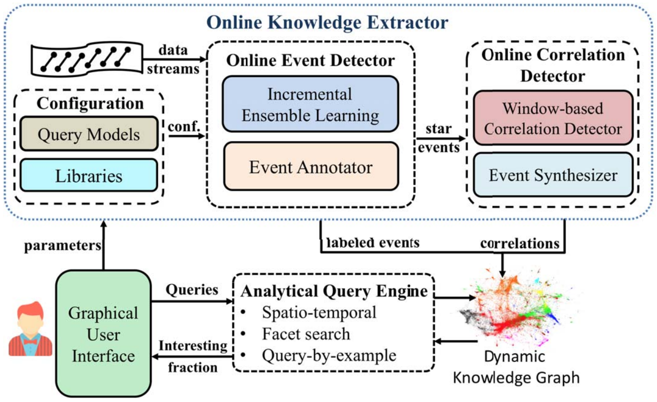


Fig. 2. Architecture of Kronos (with workflow embedded).

Star Events. Kronos adopts a primitive event model called *star event*. A star event is a two-level tree $P(v)$ with a root node v that refers to a single detectable anomaly in data streams, and a set of leaves as its adjacent property entities. Each node in $P(v)$ carries a tuple of attribute-value pairs (*e.g.,* timestamp, type of anomaly, spatiotemporal features). The lightweight star events serve as atomic units that are manipulated by Kronos as building blocks for complex events.

Dynamic Knowledge Graph. The highly dynamic event knowledge in Kronos is represented as an attributed knowledge graph $G = (V, E)$. Each node $v \in V$ (resp. edge $e = (v, v') \in E$) is a tuple that encodes an entity (resp. a relation tuple between entities v and v'). There are mainly three types of nodes: cyber-physical entities (*e.g.,* substations, sensors, servers, softwares, PMUs), event entities (anomalies, *e.g.,* 'Trip off', 'Overload'), and environmental entities (*e.g.,* substation, location, temperature); accordingly, the relations include CPS topology (among cyber-physical entities), spatiotemporal correlation (among event entities), and environmental relations (between event and cyber-physical or environmental entities).

Example 2: Fig. 1 illustrates a snapshot of an event knowledge graph G from multiple measurement data streams $\mathcal{S}$. The graph G encodes a complex event that consists of four star events: Trip off, Phase difference, and two Overload anomalies, with associated location (*e.g.,* 'North Gila'), anomaly readings (*e.g.,* '500 kV'), and devices (*e.g.,* 'transmission line'). The power outage event is revealed by the consecutive occurrence of the four anomalies as suggested by their spatiotemporal correlation, suggesting that "Trip off" is a cause of "Phase difference" which leads to two Overload events. □

B. Workflow of Kronos

Given multiple data streams $\mathcal{S}$, Kronos extracts and maintain the knowledge graph G_t at each timestamp t. Kronos workflow involves two main steps: online event detection and correlation inference, and an optional step of complex event synthesizing for specified event topology.

Online event detection. The online event detection takes as input a data stream (S, W) and extracts a set of star events. (1) It first performs anomaly detection with an *online ensemble learner* to detect anomalies from the decisions of a set of registered base detectors. (2) It then annotates the events with a set of domain-specific classification rules. Kronos tracks the relevant measurement and features, and assemble anomlies to star events. The labeled star events are then stored as building blocks for more complex events upon analytical queries.

Correlation Inference. Once the significant event entities are detected, Kronos performs a window-based incremental correlation analysis among the timestmaped star events with association models such as Granger causality [3], and temporal association rules learned in dynamic networks [7].

When no prior description of complex events is available, Kronos induces complex events with top-k strongly correlated star events by established strength measurement. Users can specify complex events with prior knowledge by specifying event analytical queries.

Configuration of Knowledge Extraction. At any time, users are able to change the source of data streams, tune the configuration parameters such as thresholds, size of windows or sample size, and plug in new anomaly models and detectors. In addition, Kronos maintains a built-in library that bookkeeps registered query classes including keyword search [8] and provenance queries [6], and event classifiers that label the detected events. New query classes, event classifiers, and base learners can be easily plugged into Kronos.

We next introduce the details of each component.

C. Online Star Event Extraction

Extracting primitive star events is already challenging for real-world CPS data streams, due to lack of labeled examples, and the fact that there is "no single winner" (classifier) to capture all the rich types of anomalies seen in complex events. Kronos hits two birds with one stone by approaching an *online unsupervised ensemble detection* framework [10].

Unsupervised Ensemble of Stream Anomalies. The general Kronos anomaly ensemble framework exploits a library $\mathcal{M}$ of k registered *base anomaly classifier* (*e.g.,* DBSCAN, change-point detection, regression). Given a window W of cached measurement data, each base classifier $M \in \mathcal{M}$ computes a binary vector $M(W)$, such that $M(W)[j]$ is either 1 (anomaly) or 0 (normal) for each tuple (feature vector) t_j in W, by computing an anomaly score $M[j]$ of t_j (deviation of t_j from the normal behavior) and verify if $M[j]$ is above a threshold. An unsupervised Maximum Likelihood Estimation (MLE) learning is invoked to estimate a weighted combination $\hat{y}$ of the binary vectors from each base models, specified for a tuple t in W as

$$\hat{y} = \mathsf{sign} \sum_{i=1}^{k} (M_i(t) \cdot \log \alpha_i + \log \beta_i) \qquad (1)$$

where each base model M_i has two weights: $\alpha_i = \frac{\psi_i \eta_i}{(1-\psi_i)(1-\eta_i)}$, and $\beta_i = \frac{\psi_i(1-\psi_i)}{\eta_i(1-\eta_i)}$, determined by its sensitivity ψ_i (resp. specificity η_i) that refers to the fraction of correctly identified anomalies (resp. normal data) in W. In a nutshell, $\hat{y}$ approximates the joint probability of a correct guess [10] by the weighted combination of $\mathcal{M}$ registered in the library.

When no labeled anomalies are available, Kronos initializes ψ and η by a pseudo ground-truth from majority voting of base models. This ensures a cold-start without assuming available training examples, and continuously improves Kronos by refining ψ and η upon verified anomalies from user feedback. Kronos supports multi-threaded parallelization where each base detector can process samples independently.

Event annotation. Kronos adopts a decision tree to incorporate domain-specific rules to annotate anomalies with specific type of events (*e.g.,* 'Overload', 'Trip off'). A star event $P(v)$ is assembled by constructing an event entity v and associating its CPS and environmental entities (*e.g.,* the sensor which captures v, critical measures involved in the decision tree).

D. Online Top-K Correlation Inference

Kronos library also incorporates a class of classifiers (*e.g.,* lagged correlation, Granger causality [3], and graph temporal association rules [7]) to decide whether two input star events are spatio-temporally correlated. To avoid unnecessary pairwise checking, Kronos incrementally tracks top-k pairs of star events from two consecutive windows W and W' with strong correlation, triggered by a new star event $P(v)$ detected in W'. Specifically, it performs quick estimation of the upper bound of the correlation scores specified by the correlation model between $P(v)$ and previously cached events up to the size of W, and prunes unpromising pairs given current top-k pairs.

E. Analytical Query Processing

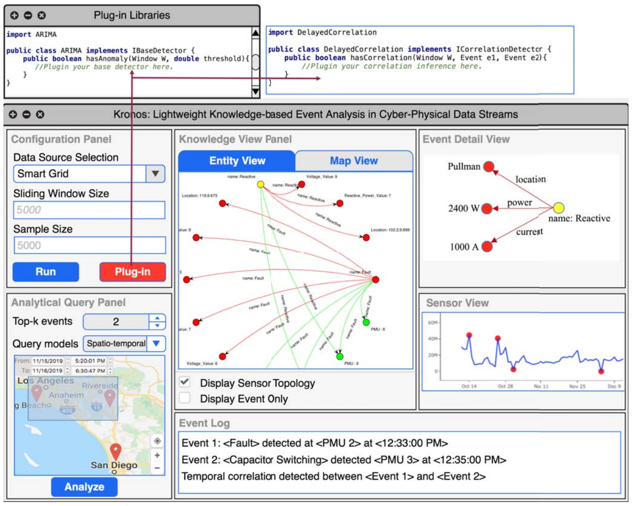
Kronos supports several classes of user-friendly event analytical queries to search the dynamic knowledge graph, as illustrated in Table I. An event analytical query specifies search predicates (*e.g.,* keywords, spatiotemporal ranges) and computes a subgraph of the dynamic knowledge graph induced by relevant star events, their associated CPS and environment entities, and their spatiotemporal correlations. Kronos query engine supports their instances specified as both predefined continuous queries (for event monitoring) or one-time queries (for ad-hoc analysis), which are processed by corresponding online graph search algorithms *e.g.,* [8].

F. Kronos *Architecture*

Kronos adopts a three-tier architecture depicted in Fig. 2. (1) The interactive interface layer allows users to visually configure the analyzing process, issue analytical queries, validate the extracted knowledge and plug in base anomaly classifiers and correlation models (Section III). (2) The online event detector includes both incremental ensemble learning and event annotators that are reading data streams from various sources. The correlation detector component performs online correlation inference triggered by newly detected alerts. (3) The star events and correlations are assembled to update the underlying RDF Kronos knowledge base.

Query Model	Syntax (Query Template)	Query Instance
Spatio-temporal [5]	Top-<k> most recent events occurred <time criteria> in <location criteria>	Top-10 most recent events occurred "between Friday and Sunday" in "Cali. substation" (Continuous)
Facet Search [8]	Top-<k> most recent events that contain a keyword from $<k_1, \ldots, k_n>$	100 most recent events that contain "Phase" or "Overload" (Continuous)
Root-cause [11]	Top-<k> events causing the event $<P(v)>$	5 events causing the event "Phase diff." at Hassayampa (One time)
Query-by-example [6]	Top-<k> most similar events to the event $<P(v)>$	Top-5 Most similar events to the event "Transformer Overload at San Diego" (Continuous)

TABLE I
ANALYTICAL QUERY MODELS AND EXAMPLES IN Kronos



```
import ARIMA

public class ARIMA implements IBaseDetector {
    public boolean hasAnomaly(Window W, double threshold){
      //Plugin your base detector here.
    }
}
```

```
import DelayedCorrelation

public class DelayedCorrelation implements ICorrelationDetector {
    public boolean hasCorrelation(Window W, Event e1, Event e2){
      //Plugin your correlation inference here.
    }
}
```

Data Source Selection: Smart Grid
Sliding Window Size: 5000
Sample Size: 5000
Top-k events: 2
Query models: Spatio-temporal
From: 11/15/2019 5:20:01 PM
To: 11/16/2019 6:30:47 PM
☑ Display Sensor Topology
☐ Display Event Only

Event 1: <Fault> detected at <PMU 2> at <12:33:00 PM>
Event 2: <Capacitor Switching> detected <PMU 3> at <12:35:00 PM>
Temporal correlation detected between <Event 1> and <Event 2>

Fig. 3. Visual Exploration of Dynamic Event Knowledge

III. DEMONSTRATION

Setup. We demonstrate Kronos with real-world and simulated data streams to show its application in smart grid resiliency, Cyber attack detection, and road traffic analysis.(1) Real-time digital simulator is used for modeling the power system and Cyber communication protocols. RTDS dataset has in total 20 minutes of measurements from 3 phases of 16 sensors (*e.g.,* voltage, current), including 5 minutes of normal operation and 15 minutes of operations with 3 types of injected anomalies *e.g.,* fault, capacitor, and load switching. (2) IDS records daily intrusion activities over a Cyber-network and includes seven families of attacks (*e.g.,* Brute Force SSH, DDoS). Streams were generated from 41 servers and contain 80 attributes (*e.g.,* duration, number of packets) from the network flow. (3) METR-LA, a highway traffic dataset contains information collected by loop detectors in the highways of Los Angeles. We select 207 sensors and collect 4 months of data.

Scenarios. We showcase the following scenarios.

Ease of use. We invite the users to experience interactive user-interface of Kronos (Fig. 3). Accessing the "Configuration" panel, users are able to select data stream sources and tune size of windows/samples. Using query panel, users can issue various analytical queries including keyword search [8], provenance queries, root cause analysis [11], and query-by-example. The extracted knowledge and topology of the sensor networks are visualized in entity and geographical map views, respectively. Detected events and correlation among them are also updating in the console that shows the source, time, and label of the entities. Kronos supports interactive knowledge exploration: users are able to drill down to 1) the specific sensors to monitor measured values over time in sensor view, and 2) the detailed properties in event view.

Cyber-power analysis. Cyber attacks are simulated by malicious code-injection in RTDS script to send unwanted "switch on" commands that increase the load and power usage of the system. This causes overloading in a transmission line that increases its temperature and can lead to sagging. As a defensing mechanism the relays then "trip off" this line. Losing this part of the network, an overload is then occurred. Kronos successfully detects overload, power increase, and fault as a result of sagging and their relationships.

Intrusion detection. Over IDS, we aggregate the network feature values of 5-minutes intervals and extract 80 snapshots. As an example, Kronos correctly identifies active upload of IRC softwares by attackers that led to DDoS attacks, taking advantage of IRC botnets in the victim host.

We also demonstrate scenarios on METR-LA that detects sudden traffic jams on a road (*e.g.,* due to an accident) that propagated to other roads, measured by nearby sensors.

Acknowledgement. This work is supported in part by Siemens, the Department of Energy under DE-IA0000025 for UI-ASSIST, and NSF under CPS 1932574, RAISE: C-Accel Pilot 1937143, and EPCN 1933279.

REFERENCES

[1] V. Chandola, V. Mithal, and V. Kumar. Comparative evaluation of anomaly detection techniques for sequence data. In *ICDM*, 2008.
[2] F. E. R. Commission. Arizona-southern california outages on september 8, 2011: Causes and recommendations. *FERC and NERC Staff*, 2012.
[3] M. Eichler. *Causal inference in time series analysis*. 2012.
[4] B. Hoßbach and B. Seeger. Anomaly management using complex event processing. In *EDBT*, 2013.
[5] W. Liu, Y. Zheng, S. Chawla, J. Yuan, and X. Xing. Discovering spatio-temporal causal interactions in traffic data streams. In *SIGKDD*, 2011.
[6] M. H. Namaki, Q. Song, and Y. Wu. Navigate: Explainable visual graph exploration by examples. In *SIGMOD*, 2019.
[7] M. H. Namaki, Y. Wu, Q. Song, P. Lin, and T. Ge. Discovering graph temporal association rules. In *CIKM*, 2017.
[8] M. H. Namaki, Y. Wu, and X. Zhang. Gexp: Cost-aware graph exploration with keywords. In *SIGMOD*, 2018.
[9] K. Teymourian and A. Paschke. Semantic enrichment of event stream for semantic situation awareness. In *Semantic Web*. 2016.
[10] M. Zhou, Y. Wang, A. K. Srivastava, Y. Wu, and P. Banerjee. Ensemble-based algorithm for synchrophasor data anomaly detection. *IEEE Transactions on Smart Grid*, pages 2979–2988, 2018.
[11] B. Zong, Y. Wu, J. Song, A. K. Singh, H. Cam, J. Han, and X. Yan. Towards scalable critical alert mining. In *SIGKDD*, 2014.