

BIG MOBILITY DATA ANALYTICS FOR TRAFFIC MONITORING AND CONTROL

UDC ((004.6+528.85):656.054.1)

Natalija Stojanović, Dragan Stojanović

University of Niš, Faculty of Electronic Engineering, Department of Computer Science,
Niš, Republic of Serbia

Abstract. *With the overpopulation of large cities, the problems with citizens' mobility, transport inefficiency, traffic congestions and environmental pollution caused by the heavy traffic require advanced ITS solutions to be overcome. Recent advances and wide proliferation of mobile and Internet of Things (IoT) devices, carried by people, built in vehicles and integrated in a road infrastructure, enable collection of large scale data related to mobility and traffic in smart cities, still with a limited use in real world applications. In this paper, we propose the traffic monitoring, control and adaptation platform, named TrafficSense, based on Big Mobility Data processing and analytics. It provides a continuous monitoring of a traffic situation and detection of important traffic parameters, conditions and events, such as travel times along the street segments and traffic congestions in real time. Upon detecting a traffic congestion on an intersection, the TrafficSense application leverages the feedback control loop mechanism to provide a traffic adaptation based on the dynamic configuration of traffic lights duration in order to increase the traffic flows in critical directions at the intersections. We tested and evaluated the developed application on the distributed cloud computing infrastructure. By varying the streaming workload and the cluster parameters we show the feasibility and applicability of our approach and the platform.*

Key words: *Big Data, IoT, mobility data, streaming analytics, traffic monitoring, traffic adaptation, feedback control loop*

1. INTRODUCTION

Large cities are facing a lot of challenges in preserving the sustainability and providing quality of life of their citizens. Among the most serious challenges that must be urgently addressed are the city's transport and citizens' mobility that start to overwhelm city's transportation and streets infrastructure. Unsolved or improperly solved transport/mobility challenge causes traffic congestions, commute/transport delays, waste of productive time,

Received July 17, 2020

Corresponding author: Natalija Stojanović

University of Niš, Faculty of Electrical Engineering, Aleksandra Medvedeva 14, 18000 Niš, Republic of Serbia

E-mail: natalija.stojanovic@elfak.ni.ac.rs

traffic accidents, and extensive emission of air pollutants (especially CO₂, NO and NO₂) into the environment. The Smart Cities of tomorrow need a significant improvement in the way transport and mobility are organized with a traffic that is controlled and adapted to the fluctuating and dynamic needs of their citizens. There is an urgent need to exploit the large amounts of data generated by sensors carried by citizens, integrated in vehicles, and built in the street infrastructure by leveraging advanced Big Data methods and technologies to collect, store, process, analyze and visualize traffic and vehicle-related data for the improved traffic monitoring and control [1].

The advanced technologies in autonomous and connected cars equipped with plenty of sensors (GPS, LiDAR, street and dash cameras, etc.), and enriched with Vehicle to Vehicle (V2V) and Vehicle to Infrastructure (V2I) communications, as well as sensors integrated in road infrastructure, such as inductive loop detectors and cameras, generate increasing amounts of vehicle and traffic related data. Such Big traffic and mobility Data, fused with data originated from citizens' mobile devices and social media, are characterized by high Volume, Velocity, Variety of data sources, and Veracity, and need appropriate methods, technologies and systems to generate significant value to Intelligent Transportation Systems (ITS) [2].

In this paper, we propose advanced Big mobility and Internet of Things (IoT) data processing and analytics architecture and the platform for continuous collection of traffic and mobility-related data, that provide control and adaptation of traffic based on current situation, conditions and events. The collected traffic and vehicle related data is used for continuous (real-time) detection of average speed and travel times along street segments, heavy traffic and traffic congestions, as well as traffic patterns and their relations with contextual information. The analysis and mining of off-line (historical) Big mobility data provide detection of mobility patterns depending on the time of the day/week/month/year, a weather situation, mass events in the city, etc. The platform provides prediction of the traffic characteristics and conditions leveraging machine learning models and techniques and proactive reactions based on such a prediction.

By implementing architectural components for processing, analysis and visualization of Big Data collected by moving vehicles and in-road sensors, we developed the *TrafficSense* platform for traffic monitoring and control that provides a real-time traffic streaming analytics and detection of important traffic events and conditions, such as:

- Average travel times over street segments,
- Slow-moving traffic and traffic jams,
- Traffic stop for a longer time,
- Heavy traffic along street segments.

Upon detection of traffic conditions and events, the platform uses feedback control loop mechanism to change and adapt the traffic situation. Traffic control and adaptation are performed by changing the duration of the traffic lights at congested intersections and changing the street parameters (one-way, close/open lanes, a speed limit, etc.). In that way, the traffic adaptation provides a reduction of traffic congestions and heavy traffic conditions.

The main contributions of the paper are:

- We propose the *TrafficSense*, Big Streaming Data architecture and the platform for continuous processing and analytics of massive amount of traffic and mobility related data originated from vehicles and street infrastructure sensors.

- We develop the *TrafficSense* application that employs the feedback control loop mechanism to adapt traffic actuators (traffic lights, variable message signs, etc.) and traffic monitoring through a Web dashboard application.
- We evaluate experimentally the application and the platform using the realistic traffic simulation data, and prove the feasibility and applicability of our approach, as well as the performance and scalability with increasing traffic/mobility data loads and velocity.

The rest of the paper is organized as follows. In Section 2, some background information and the related work are presented. The architecture and the main components of the *TrafficSense* platform intended for traffic monitoring and control are described and discussed in Section 3. In Section 4, the implementation details and runtime behavior of the traffic analysis and control and adaptation of traffic lights by the *TrafficSense* application, are presented. The experimental evaluation and its results are presented in Section 5. The concluding remarks are given in the last Section along with the outline of future research directions.

2. BACKGROUND AND RELATED WORK

The increase in the amount and availability of mobile crowd-sensing and IoT data has fueled the vision of Smart Cities that will improve every aspect of our urban lives, including transport and mobility, health, energy use, environment preservation, manufacturing, etc. [3]. Big Data processing and analytics is classified as a batch and stream processing and corresponding components are included in the software system architectures proposed for data-intensive applications, known as the Lambda and Kappa¹ architecture [4]. The Lambda architecture comprises the Batch Layer, Speed Layer (Stream layer) and Serving Layer. The Batch processing and analytics is suitable for performing long-running queries over a data lake, and for analysis that leverages data mining and machine learning techniques to find patterns in data and extract knowledge. The Kappa architecture focuses only on the streaming data processing and finds its applications in the real-time detection and analytics of events. While the Hadoop² framework is primarily designed for the batch processing based on the MapReduce paradigm, there are several Big Streaming Data frameworks suitable for the Speed (Stream) Layer and the implementation of the Kappa architecture, such as Apache Storm³, Heron⁴, Spark⁵, and Flink⁶ [4]. The traffic stream processing enables performing data analytics on the fly, and a real-time detection of events and conditions based on data originated from vehicles, loop detectors, street sensors/cameras, weather data, etc. The results of the streaming analytics are often real-time and proactive responses and notifications to drivers and traffic control operators.

Torre-Bastida et al. [5] present the up-to-date research related to Big Data methods, technologies and tools in supporting transport and mobility applications, such as navigation, route planning, traffic monitoring, network design, and others. The paper provides an

¹ <http://milinda.pathirage.org/kappa-architecture.com/>

² <https://hadoop.apache.org/>

³ <https://storm.apache.org/>

⁴ <https://heron.incubator.apache.org/>

⁵ <http://spark.apache.org/>

⁶ <https://flink.apache.org/>

extensive literature review and an analysis of contributions in modeling, processing, analyzing and visualizing transport and mobility Big Data and concludes with trends and challenges that open up new research and advances in the field of transport and mobility.

The survey of Big Data analytics in the ITS framework is given in [2] along with the review of data collection and analytics methods and platforms, as well as Big Data analytics application categories. The authors describe several case studies and discuss open challenges of Big Data analytics applications in ITS, including road traffic accidents analysis, road traffic flow prediction, public transportation service plan, personal travel route plan, and others.

Guerreire et al. [6] propose an architecture based on Apache Spark and MongoDB technologies for batch processing of traffic related information collected from road sensors and telematics data. Almost the same authors in [7] propose a scalable architecture for the traffic management capable of processing the real-time traffic data captured from inductive loop counters integrated in a road network. Their approach enables real-time monitoring of the road network and detection of traffic events, based on the Apache Storm Big Streaming Data framework. They performed experimental evaluation and validation obtaining performance metrics that justify the use of Big Data streams management system with the real world, large-scale traffic loop sensor data.

The research performed within the CityPulse⁷ EU FP7 project and presented in [8] focuses on the Smart City framework for processing large-scale IoT data streams. The main goal is to enrich large-scale traffic data streams with semantic annotations, enabling adaptive processing, aggregation and federation of data. The authors discuss the challenges of smart adaptation and integration of various sensory data sources in extraction of useful knowledge intended for citizens and city authorities.

The application of Complex Event Processing (CEP) rules to detect events and congestions in traffic data streams is presented in [9]. The authors propose a dynamic adaptation and optimization of CEP rules and their thresholds by using clustering techniques and demonstrate the usefulness and applicability of their approach using a real-world use case of ITS to detect congestions in near real-time.

Ta-Shma et al. [10] propose so-called "hut" architecture for ingesting and analyzing IoT data, which combines both historical and real-time data and performs the batch data analysis to provide a context for real-time analysis. The authors implement the proposed architecture using the open source Big Data technologies and components and demonstrate the feasibility of their solution in two real-world Smart City use cases for transportation and energy management.

Amini et al. [11] propose a flexible architecture based on a distributed computing platform for real-time traffic control. The part of the architecture is implemented using Big Data technologies with intention to send information and messages to the traffic control logic. They demonstrate the proposed approach on a case study of controlling the opening and closing of a freeway hard shoulder lane based on the traffic density and crowdedness.

Schmid et al. [12] focus on self-adaptation in a large-scale vehicle navigation system. They present a novel approach based on the system model characterized by essential input and output parameters and provide a self-adaptation of the navigation system based on an analysis of streaming data coming from the system. They develop a system and a tool named the Crowdsourced Navigation system (CrowdNav), based on the SUMO traffic simulator [13],

⁷ <http://www.ict-citypulse.eu>

that provides a dynamic adaptation of navigation instructions according to the current traffic status and conditions through the open source RTX⁸ framework.

A novel architecture, named Theta, intended for adaptive data-driven systems that can change and adapt used data sources and data analysis algorithms at runtime to preserve the target quality of its outcome, is proposed in [14]. The authors present and describe an example of a vehicle management system where the adaptive Big Data analytics based on Theta architecture, could be successfully applied.

In [15], the authors present a Big Data platform, named Sipresk, to support analytics over large traffic data collected from various data sources. They provide an adaptation of the platform to the changing environment, such as the high workload, and the network failure, by leveraging a MAPE-K loop [16] based solution. The Sipresk platform is deployed and validated on the several use cases, such as finding average speed and congested segments in the major highways in Greater Toronto Area.

In contrast to some of the existing solutions, the *TrafficSense* architecture and the platform for processing and analytics of Big Streaming Data were developed by extending the Kappa architecture. We use historical (offline) traffic and mobility data to setup the configuration parameters for the traffic streaming analytics methods. As in [15], our platform implements feedback control loop mechanism based on the MAPE-K architecture for self-adaptive systems, but with the intention to perform the adaptation of the physical system under monitoring and control (i.e. traffic), not the platform itself in order to cope with high workload and failures. In contrast to [11], our platform performs the traffic adaptation through the traffic light duration control, as a more complicated scenario than just the opening and closing a shoulder lane on a highway. We demonstrate that our approach is feasible, applicable, scalable and can be easily deployed and operated in a real-world ITS scenario for traffic control and adaptation.

3. THE TRAFFICSENSE PLATFORM ARCHITECTURE

Continuous monitoring, collection and analytics of massive amounts of traffic sensor, IoT and mobility data (Big Data) require high resource usage for processing and analytics along the data/control pipeline, increase computational costs required to analyze data, consume significant storage, and increase network usage. To provide streaming processing and analysis of relevant and up-to-date traffic-related information, and generation of proactive responses and notifications, we propose a distributed Big mobility data analytics architecture and the platform, *TrafficSense*. The *TrafficSense* platform is based on the edge-cloud compute continuum providing modular and distributed components and services for Big Data storage, processing, analytics and visualization. It supports the extraction of the high-level representations of the traffic features and conditions by exploring aggregation, fusion and analytics of IoT sensor and mobility data in real-time to provide continuous monitoring and prediction of critical traffic events and conditions.

The architecture of the *TrafficSense* platform, highlighting the basic components that play appropriate roles in Big Data processing and control flow, is shown in Fig. 1. The architecture consists of components deployed and executed on IoT/mobile devices and edge gateways at the edge of the architecture for: i) Sensor data collection and actuator

⁸ <https://github.com/Starofall/RTX/>

control ii) Sensor fusion and context enrichment, iii) Local processing, streaming analytics and event detection. The *TrafficSense* platform for management of Big streaming and offline data on a cluster/cloud computing infrastructure includes components for: iv) Big traffic streaming data processing and analytics, v) Storage, aggregation and mining of Big traffic data, and vi) Visualization and visual analytics by a dashboard application. The communication between *TrafficSense* IoT/mobile/edge components and the cluster/cloud components relies mainly on asynchronous message-passing and publish/subscribe mechanism supported by a message broker middleware.

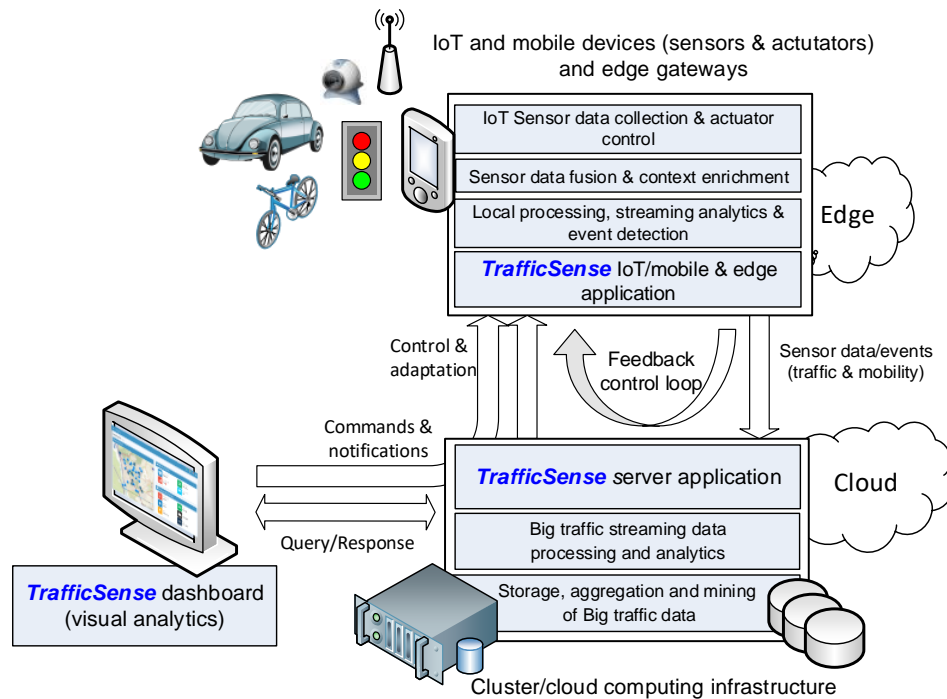


Fig. 1 The *TrafficSense* platform architecture

The edge components of the platform are based on the IoT and mobile sensor devices to collect data necessary for traffic monitoring and control. The data come from various sources, devices built in the vehicles, or integrated in the road infrastructure, such as induction loop detectors, video surveillance cameras, RFID tags, GPS, microwave radars, etc. The extensive review and the classification of the various traffic data collection methods and technologies can be found in [2, 17]. Such data can be enriched and correlated with external and environmental data sources collected within a Smart City infrastructure (weather conditions, environment pollution/noise, temperature, city events, social media, etc.). The fusion, context enrichment and streaming analytics of IoT and mobility traffic-related data are performed within *TrafficSense* IoT/mobile devices and edge computing components providing efficiency and minimal latency in detection of critical traffic events and conditions that require prompt actions. The processing and analysis of Big IoT,

mobility and traffic data streams are performed within a computer cluster and/or public, private or hybrid cloud infrastructure within the *TrafficSense* server (back-end) components. Also, the storage, aggregation, and mining over Big traffic-related data provide detection of mobility patterns depending on the contextual information, such as the time of the day/week/month/year, the weather situation, the mass events in the city, etc.

The results of Big streaming and offline data analytics are provided to a traffic control center and traffic operators through tailored visual analytics, dashboard application providing human-in-the-loop principle. The *TrafficSense* platform provides support for the development and operation of specific ITS applications with components deployed at the IoT/mobile and edge devices, as well as the cluster/cloud computing infrastructure. Such applications relies on the *TrafficSense* components for the Big traffic and mobility data processing and analytics tailored for specific application scenarios such as traffic flow prediction, traffic signal control, public transport management, emergency management, etc.

The *TrafficSense* architecture and the platform implement a feedback control loop mechanism and adhere to the MAPE-K model and architecture principle for self-adaptive systems [16]. The MAPE-K architecture includes *Monitor*, *Analyse*, *Plan*, *Execute* modules and a shared *Knowledge* base. The *Monitor* module collects data from sensors in environment, sends them for *Analysis* of monitored data, *Planning* response actions, and *Execution* of these actions through the actuators integrated in the environment. All modules are based on the common *Knowledge* representation of the system under monitoring, control and self-adaptation. The *TrafficSense* IoT/mobile/edge components implement *Monitor* features, *Analyse* and *Plan* functionalities are implemented and spread across the *TrafficSense* edge and cloud data processing and analytics components, while *Execute* capabilities are provided by *TrafficSense* IoT/mobile actuators, closing the feedback control loop. All mentioned components of the *TrafficSense* platform along the edge-cloud compute continuum provide analytics and machine learning capabilities and include the *Knowledge* component of MAPE-K required for making decisions about personalized traffic services, control and adaptation.

4. TRAFFIC STREAMING ANALYTICS FOR TRAFFIC LIGHTS ADAPTATION

The traffic monitoring and adaptation scenario explored in this paper is based on the use of Floating Car Data (FCD) or, more specifically, the GPS probe data, consisting of the location (latitude, longitude) and the time, collected from vehicles moving along the city's streets. The Big traffic and mobility data streams are collected by IoT mobile devices and edge gateways, and processed by traffic streaming analytics application in a traffic control center. The processing and analytics provide detection of traffic conditions and events in real-time, such as average speed along the street segments and traffic congestions at intersections. By employing a feedback control loop the application provides the traffic adaptation through traffic actuators, traffic lights and variable message signs (VMS).

Instead of real IoT and in-vehicle devices, and dedicated *TrafficSense* IoT/mobile application for traffic data collection, we use SUMO⁹ [13], an open source, microscopic and continuous traffic simulator, developed by the Institute of Transportation Systems at the German Aerospace Center to achieve realistic traffic and mobility data (FCD)

⁹ <https://www.eclipse.org/sumo/>

generation (Fig. 2). We implemented the *TrafficSense* server application for traffic streaming analytics and traffic adaptation using contemporary Big Data technologies.

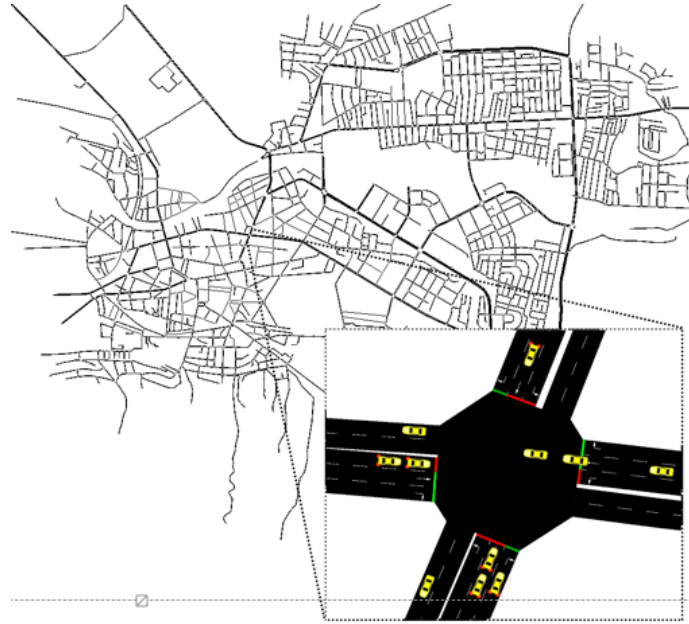


Fig. 2 SUMO microscopic traffic simulator

In order to provide the control of the simulation and traffic parameters, TraCI¹⁰ (Traffic Control Interface) is used. TraCI provides programming access to the running traffic simulation, allowing retrieval of values that characterize simulated vehicles in motion and manipulation of their behaviour "on-line". The streaming mobility (FCD) and traffic data originating from the SUMO simulator are published to the specific topics of the Apache Kafka¹¹ message broker. We choose Kafka broker among other alternatives, such as RabbitMQ¹², due to its high efficiency and message throughput, as well as scalability when deployed to the cluster/cloud infrastructure. The Kafka message broker receives and collects mobility and traffic data published to a specific topic by various sources, in our scenario only the SUMO simulation data, but in a real settings, also data from RFIDtags, inductive loops, street sensors/cameras, etc.. The data are received by the subscribed *TrafficSense* back-end application that is based on the server components of the *TrafficSense* platform for processing and analytics of Big Streaming Data related to traffic and vehicle mobility. The *TrafficSense* application is implemented using the Apache Spark framework and Spark Core, Structured Streaming and SparkML components that provides APIs for batch and streaming data processing and analytics at the massive data scale. Comparing to other Big Data frameworks, such as Apache Storm and Heron, Apache Spark includes components for both batch and streaming data analytics and machine learning, and also is more mature

¹⁰ https://sumo.dlr.de/docs/TraCI/Interfacing_TraCI_from_Python.html

¹¹ <https://kafka.apache.org/>

¹² <https://www.rabbitmq.com/>

than the Apache Flink framework. In the future developments we will consider use of Apache Beam¹³, an open source, unified model for defining both batch and streaming data processing pipelines that can be executed by supported distributed processing frameworks, such as Apache Flink, Apache Spark, and Google Cloud Dataflow [4].

To provide control and self-adaptation of the traffic simulation generated by SUMO we extended CrowdNav¹⁴ tool with the self-adaptation functionalities (*CrowdNav-SA*) that can change and adapt the traffic simulation and simulated vehicle parameters in a real-time using TraCI interface. *CrowdNav-SA* can subscribe to receive messages on the particular Kafka topics and, based on local JSON configuration, performs adaptation of the traffic in real time while the simulation is running. Upon receiving Kafka messages published by *TrafficSense* application that inform on congestions on particular streets and intersections, *CrowdNav-SA* calls the SUMO operations via TraCI API to change the simulated traffic parameters and re-configure the simulation. In this way, the *TrafficSense* application and the *CrowdNav-SA* tool implement *Analyse-Plan-Execute* actions of the MAPE-K framework and close the feedback control loop for traffic control and adaptation.

The *TrafficSense* application provides dashboard functionality through a real-time Web dashboard application for monitoring of traffic and mobility of vehicles implemented using Leaflet¹⁵ JavaScript map library and MapBox¹⁶ API for providing background geographic maps. To provide dynamic visualization of traffic congestions occurring and a slow traffic over particular street segments, Eclipse Vert.x¹⁷, an event-driven application framework, was used. We have implemented the Vert.x service and the Web dashboard application connected through Web Socket interface that visualizes the vehicles in motion, dynamic travel times along the street segments, and traffic congestions, as the results of the processing and analysis performed by Spark Structured Streaming jobs (Fig. 3).

The *TrafficSense* platform and the application were implemented, deployed and evaluated on a cluster of computers running the Apache Spark platform and Apache Hadoop/HDFS using container virtualization for maximum flexibility and scalability. The Docker images for SUMO simulator, the *CrowdNav-SA* tool, and the Kafka message broker have been created and deployed at the master computer node, and started as containers. Traffic simulation data generated by SUMO are published to dedicated Kafka topics and consumed by the subscribed *TrafficSense* application executed as a series of Spark Structured Streaming Jobs. The

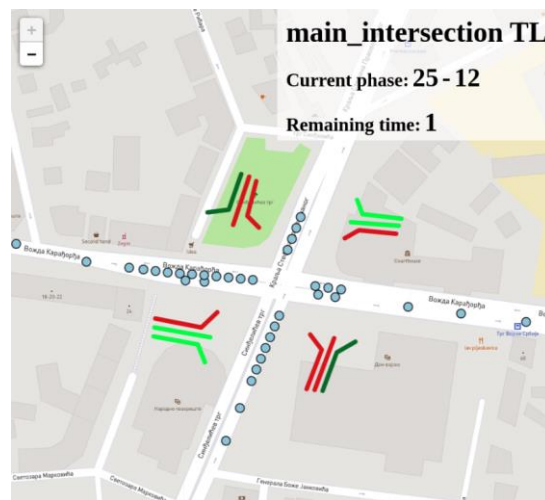


Fig. 3 The screenshot of the *TrafficSense* dashboard

¹³ <https://beam.apache.org/>

¹⁴ <https://github.com/Starofall/CrowdNav>

¹⁵ <https://leafletjs.com/>

¹⁶ <https://www.mapbox.com/>

¹⁷ <https://vertx.io/>

application continuously processes and analyses incoming streaming mobility (GPS probe) data and stores the results on a HDFS and Cassandra NoSQL database. Based on OpenStreetMap¹⁸ street network data, *TrafficSense* continuously detects the number of vehicles on the street segments, the average speed along street segments, and the number of slow or non-moving vehicles on the intersections that cause traffic congestions. Upon detection of congested street segments and intersections, the *TrafficSense* application publishes appropriate messages to the Kafka topic that is consumed by the subscribed *CrowdNav-SA* traffic adaptation tool. The *CrowdNav-SA* tool provides methods and operations to change and adapt the simulation parameters of the running SUMO simulation, such as the duration of street lights in certain directions at intersections, the speed limit in crowded streets, the traffic mode (one/two ways) for certain street segments, etc. This reconfiguration performs the feedback control loop mechanism and cause self-adaptation of the traffic to overcome traffic congestions and heavy traffic [16].

The general architecture of the *TrafficSense* server application indicating technologies used for the implementation, is shown in Fig. 4.

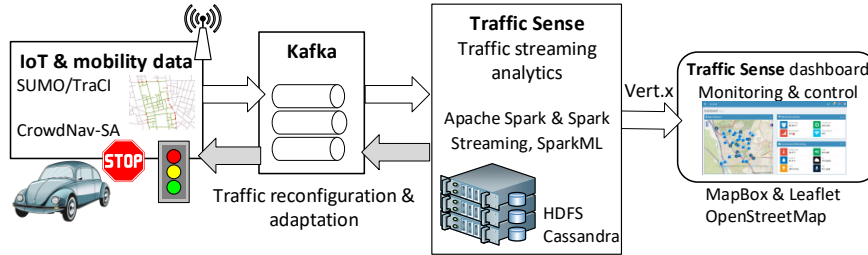


Fig. 4 *TrafficSense* application for traffic streaming analytics – used technologies

We have performed several experiments for the traffic adaptation through re-configuration of the traffic lights duration for the main city square in the city of Niš for various parameters related to the traffic conditions and stream processing and analytics. The parameters that enable detection of a vehicle that is almost stopped at the intersection are the minimal speed S_{min} (close to 0 km/h) and the time period in which a vehicle moves at that speed or slower. The detection of traffic status at the intersection is defined by the threshold values representing the number of stopped vehicles at that intersection (V_{min} , V_{max}):

- $Number_of_stopped_vehicles < V_{min}$ → *Low traffic*
- $V_{min} < Number_of_stopped_vehicles < V_{max}$ → *Normal traffic*
- $V_{max} < Number_of_stopped_vehicles$ → *Heavy traffic*

Two other configurable parameters are related to traffic lights. The TL_{inc} represents the percentual increase of the traffic light duration, and TL_{dec} – the percentual decrease of the traffic lights duration to allow more or less vehicles passing in that direction. It means, when the application detects a heavy traffic in particular direction at the intersection, it adapts and increases the traffic light duration for $TL_{inc}\%$. In contrast to it, when the light traffic is detected, the *TrafficSense* application decreases the duration of traffic light for $TL_{dec}\%$ for that direction at the intersection. Regarding the streaming data processing and analytics, there are several parameters that can be configured before the simulation and evaluation start, such as *window size*, *sliding windows*, *watermark*, etc.

¹⁸ <https://www.openstreetmap.org/>

We have executed several SUMO simulations varying simulation parameters to collect large (simulated) traffic data and empirically determine the most suitable values of the *TrafficSense* application parameters related to the main intersection, shown in Fig. 3. In the future research, we plan to provide a dynamic determination and calculation of these parameters by using the continuous traffic analytics and application of machine learning techniques. In a real world scenario, to appropriately define these parameters, the application should also take into the account the traffic context, such as the time of the day/week/year, the weather conditions, the city mass events, and similar conditions and situations that affect the city traffic.

We performed several traffic simulations with varying parameters and achieved significant improvement in the traffic flow at the main intersection leveraging the feedback control loop mechanism and the traffic lights adaptation. The experiment performed for 1000 of vehicles moving in the city of Nis, at the main intersection, where the durations of green traffic lights are 25s and 14s, for going straight and turning left respectively. The other parameters are defined as: $S_{min} = 0 \text{ km/h}$, $V_{min} = 4$, $V_{max} = 8$, $TL_{inc} = TL_{dec} = 50\%$.

The Fig. 5 shows the traffic conditions at the 5th and 10th minute of the traffic simulation, without (a), (c) and with (b), (d) traffic lights control and adaptation.

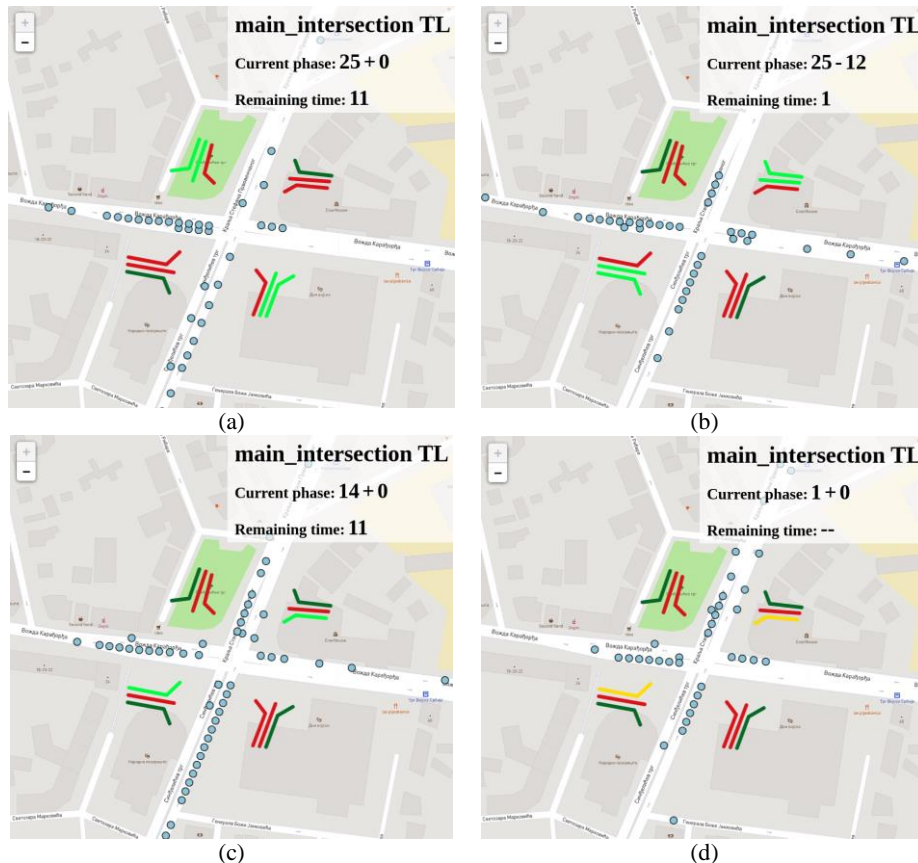


Fig. 5 The traffic conditions at the main intersection without (a), (c) and with (b), (d) traffic lights adaptation at 5th minute (a), (b) and 10th minute (c), (d) of the simulation

We have continuously collected the number of stopped vehicles at the main intersection during 30 min of simulation without and with traffic lights control and adaptation (Fig. 6). According to the average number of stopped vehicles during the 30 minute long simulation (dashed lines) we can conclude that the traffic flow is improved for about 39.4% (for $TL_{inc} = 30\%$) and 49.75% (for $TL_{inc} = 50\%$) when the traffic lights control and adaptation are applied.

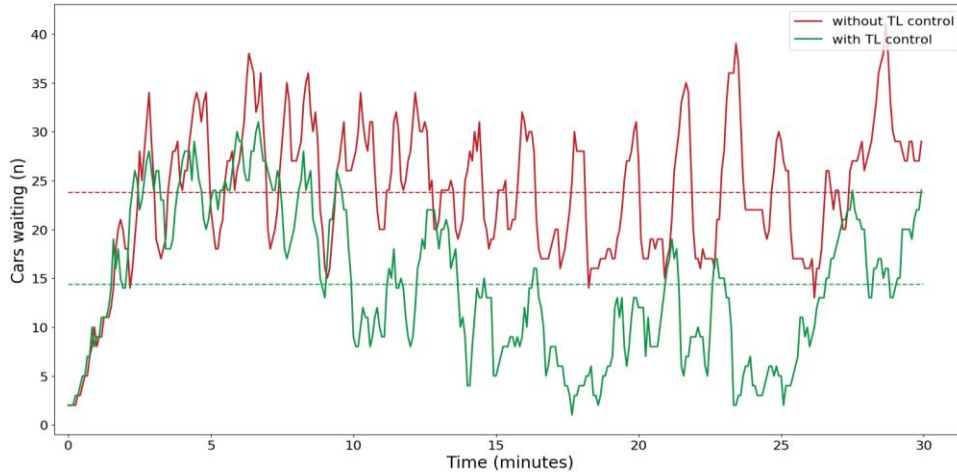


Fig. 6 The continuous number of stopped vehicles at the intersection without (red) and with (green) traffic lights control and adaptation in time

5. EXPERIMENTAL EVALUATION

We performed experiments to evaluate the performance and scalability regarding the processing and analytics of Big Streaming Data related to traffic and mobility. For the purpose of testing and evaluation of the *TrafficSense* platform and the application for traffic streaming analytics and adaptation, we deployed the application on the two Future SOC Lab computing architectures:

- The cluster of 9 virtual machines: 1 master (8 Cores-2GHz and 8GB RAM), slaves (4 Cores-2GHz and 4GB RAM)
- Multi-core computer with Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20GHz, with two sockets, 40 CPU cores, 256GB RAM.

Apache Hadoop/HDFS, Spark, Kafka and Cassandra were installed and configured on both platforms, and the containerization of Spark and Cassandra using Docker were performed on the multi-core computer. The *CrowdNav-SA* tool and the *TrafficSense* application were developed using Python programming language and PySpark API is used for accessing Apache Spark Core and Structured Streaming and ML features.

For the cluster of nine virtual machines, *TrafficSense* is executed using one Spark Driver running at the Master node and eight Spark Executors running at Worker nodes, allocating 2.5GB RAM each and executing on all available CPU cores. Also, the HDFS

and Spark daemons are executed in the background, consuming at least one core. In the multi-core computer setting, one Spark master and 2 Spark workers are started as containers, executing 10 Executors each consuming 4 cores and 16GB of RAM.

We evaluated the maximum workload with 10000 simulated vehicles that report location/time data each second, generating more than 25 million messages sent by three independent streams to the dedicated Kafka topic. The *TrafficSense* application processed and analyzed the whole streaming workload for about 15 minutes, so the peak throughput is almost 28000 messages/second. This means that the application can receive 28000 messages from the Kafka *traffic* topic at each second, perform processing and analytics to detect the status of the traffic at each street segment and intersection, and generate and publish control and adaptation messages to the another Kafka topic to be consumed by *CrowdNav-SA* that dynamically re-configures SUMO simulation parameters (traffic lights duration). The tests with six streams sending 50 million of messages shows similar performance, and the application completes the streaming workload for about 32 minutes. The experiments shows that even the experimental cluster with configuration of 9 commodity computers (virtual machines) would successfully monitor and control traffic lights in a real-world deployment, with more than 10000 vehicles reporting their position at each second. In the multi-core computer setting, the *TrafficSense* application shows similar performance in average for completing processing and analysis of both 25 million and 50 million of messages and adapting traffic lights at the crowded intersections. The experiments proved the usability of the proposed platform and the application in a real world scenario. To evaluate the performance limits of the application, in the simulated scenario, we defined that each simulated vehicle reports its location every 1 second. In the real world scenarios, such high location reporting rate is usually not applied; vehicles expect to report their locations at 10-30 seconds interval. The average number of CPU cores employed at each Worker nodes by Spark Executors and their tasks during the duration of the simulation is shown in Fig. 7, presenting the workload of the cluster over time.

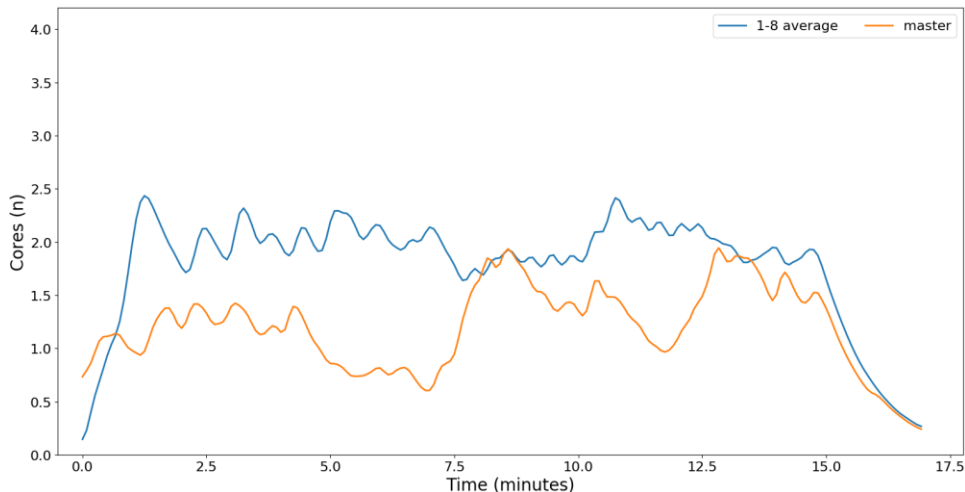


Fig. 7 The average workload of CPU cores at each Worker nodes during execution

We did not perform experiments to detect the *TrafficSense* application performance limits that would be reached mainly because of the communication bottleneck between *CrowdNav-SA* tool and the application via Kafka. The main advantages of the *TrafficSense* platform and the application are their openness and scalability owing to the foundational Big Data technologies, Apache Kafka and Spark. Both the Kafka broker and the Apache Spark framework are fully scalable, and can easily scale-out to a large number of computer nodes without any change in the application. Such scalability provides improved performance and throughput for more voluminous and faster streaming workload and the *TrafficSense* platform and the application could be employed even in the heavy traffic scenario, e.g. during the rush hour in the city of Belgrade.

6. CONCLUSION

In this paper, we propose the *TrafficSense* architecture and the platform for traffic monitoring and control and implement the *TrafficSense* application for traffic adaptation through dynamic traffic lights duration based on processing and analytics of large-scale vehicle location data streams. We developed the application using open source Big Data technologies and tools, leveraging feedback control loop techniques through MAPE-K approach for self-adaptive systems.

The feasibility of the proposed architecture and the platform is demonstrated in the *TrafficSense* application using realistic microscopic traffic simulator and near real-world use case of traffic adaptation in the city of Nis. We demonstrate the application functionality through a successful real-time control and adaptation of traffic congestions at the intersections and the system scalability and feasibility using experiments with large number of vehicles whose locations are collected through several high-speed data streams entering the system. The application operates more than satisfactory both in terms of functionality, as well as performance and scalability.

Currently the system threshold values and configuration parameters necessary for the adaptation are defined by using historical traffic data analysis for each simulation/scenario. One of the prominent directions for the future research and development is to advance the *TrafficSense* platform for the more realistic scenarios providing run time adaptation of the system configuration and parameters, depending on the traffic context and situation, weather and road conditions, and other external data sources. Also, the adaptation for each intersection is currently performed independently of others. The future research will provide the integrated traffic light control and adaptation that depends on the several connected intersections depending on the traffic flow and detected traffic patterns (e.g. home-work during week days, residential area – center of nightlife for weekend evenings). The future improvements of the *TrafficSense* platform will include a fusion of various traffic and street sensor data and the integration of new traffic control mechanisms, such as variable message signs, speed limits, and street directions (one/two ways), and their run time control and adaptation through feedback control loop mechanism.

Acknowledgement: *This paper was realized as a part of the projects "Studying climate change and its influence on the environment: impacts, adaptation and mitigation" (III 43007), supported by the Ministry of Education, Science and Technological Development of the Republic of Serbia within the framework of integrated and interdisciplinary research for the period 2011-2020. The authors would like to thank to the Hasso Plattner Institute (HPI) for granting access to the Future SOC Lab cloud computing infrastructure.*

REFERENCES

- [1] A. Luckow, K. Kennedy, F. Manhardt, E. Djerekarov, B. Vorster and A. Apon, "Automotive big data: Applications, workloads and infrastructures," *In IEEE International Conference on Big Data (Big Data)*, Santa Clara, CA, pp. 1201-1210, 2015. [Online]. Available: <http://dx.doi.org/10.1109/BigData.2015.7363874>
- [2] L. Zhu, F. R. Yu, Y. Wang, B. Ning and T. Tang, "Big Data Analytics in Intelligent Transportation Systems: A Survey," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 1, pp. 383-398, Jan. 2019. [Online]. Available: <http://dx.doi.org/10.1109/TITS.2018.2815678>
- [3] D. Stojanović, B. Predić, N. Stojanović, "Mobile crowd sensing for smart urban mobility," In: Capineri, C, Haklay, M, Huang, H, Antoniou, V, Kettunen, J, Ostermann, F and Purves, R. (eds.) *European Handbook of Crowdsourced Geographic Information*, pp. 371-382. London: Ubiquity Press. 2016. [Online]. Available: <http://dx.doi.org/10.5334/bax>.
- [4] T. Akidau, S. Chernyak, R. Lax, *Streaming Systems*. O'Reilly Media, Inc., 2018.
- [5] A. I. Torre-Bastida, J. Del Ser, I. Laña, M. Ilardia, M. N. Bilbao and S. Campos-Cordobés, "Big Data for transportation and mobility: recent advances, trends and challenges," in *IET Intelligent Transport Systems*, vol. 12, no. 8, pp. 742-755, 2018. [Online]. Available: <http://dx.doi.org/10.1049/iet-its.2018.5188>
- [6] G. Guerreiro, P. Figueiras, R. Silva, R. Costa and R. Jardim-Goncalves, "An architecture for big data processing on intelligent transportation systems. An application scenario on highway traffic flows," 2016 IEEE 8th International Conference on Intelligent Systems (IS), Sofia, pp. 65-72, 2016. [Online]. Available: <http://dx.doi.org/10.1109/IS.2016.7737393>
- [7] P. Figueiras, Z. Herga, G. Guerreiro, A. Rosa, R. Costa, R. Jardim-Gonçalves, "Real-Time Monitoring of Road Traffic Using Data Stream Mining," *IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, Stuttgart, pp. 1-8, 2018. [Online]. Available: <http://dx.doi.org/10.1109/ICE.2018.8436271>.
- [8] D. Puiu et al., "CityPulse: Large Scale Data Analytics Framework for Smart Cities," in *IEEE Access*, vol. 4, pp. 1086-1108, 2016. [Online]. Available: <http://dx.doi.org/10.1109/ACCESS.2016.2541999>.
- [9] A. Akbar, F. Carrez, K. Moessner, J. Sancho and J. Rico, "Context-aware stream processing for distributed IoT applications," *IEEE 2nd World Forum on Internet of Things (WF-IoT)*, Milan, pp. 663-668, 2015. [Online]. Available: <http://dx.doi.org/10.1109/WF-IoT.2015.7389133>.
- [10] P. Ta-Shma, A. Akbar, G. Gerson-Golan, G. Hadash, F. Carrez, K. Moessner, "An Ingestion and Analytics Architecture for IoT Applied to Smart City Use Cases," in *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 765-774, April 2018. [Online]. Available: <http://dx.doi.org/10.1109/JIOT.2017.2722378>.
- [11] S. Amini, I. Gerostathopoulos and C. Prehofer, "Big data analytics architecture for real-time traffic control," *5th IEEE International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, Naples, pp. 710-715, 2017. [Online]. Available: <http://dx.doi.org/10.1109/MTITS.2017.8005605>.
- [12] S. Schmid, I. Gerostathopoulos, C. Prehofer, T. Bures, "Self-adaptation based on big data analytics: a model problem and tool." In *2017 IEEE/ACM 12th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)*, pp. 102-108., 2017. [Online]. Available: <http://dx.doi.org/10.1109/SEAMS.2017.20>
- [13] P. A. Lopez et al., "Microscopic Traffic Simulation using SUMO," *21st International Conference on Intelligent Transportation Systems (ITSC)*, Maui, HI, pp. 2575-2582, 2018, [Online]. Available: <http://dx.doi.org/10.1109/ITSC.2018.8569938>.

- [14] V. Theodorou, I. Gerostathopoulos, S. Amini, R. Scandariato, C. Prehofer, M. Staron, "Theta Architecture: Preserving the Quality of Analytics in Data-Driven Systems," *In European Conference on Advances in Databases and Information Systems*, pp. 186-198. Springer, Cham, 2017. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-67162-8_19
- [15] H. Khazaei, S. Zareian, R. Veleda, M. Litoiu, "Sipresk: A Big Data Analytic Platform for Smart Transportation, " In: Leon-Garcia A. et al. (eds) *Smart City 360°*. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 166, 2016. Springer, Cham, [Online]. Available: http://dx.doi.org/10.1007/978-3-319-33681-7_35
- [16] E. Rutten, N. Marchand, D. Simon, "Feedback Control as MAPE-K Loop in Autonomic Computing," *In: Software Engineering for Self-Adaptive Systems III. Assurances*. de Lemos R., Garlan D., Ghezzi C., Giese H. (eds) *Lecture Notes in Computer Science*, vol. 9640. pp 349-373, Springer, Cham. 2018. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-74183-3_12
- [17] J. Lopes, J. Bento, E. Huang, C. Antoniou and M. Ben-Akiva, "Traffic and mobility data collection for real-time applications," *13th International IEEE Conference on Intelligent Transportation Systems*, Funchal, 2010, pp. 216-223, [Online]. Available: <http://dx.doi.org/10.1109/ITSC.2010.5625282>.