

# Simulation Tools for Transport Monitoring Systems in the Mining Industry

Fares Abu-Abed<sup>1,\*</sup>

<sup>1</sup>Tver State Technical University, A. Nikitin Street, 22, 170026, Tver, Russia

**Abstract.** This paper presents a computer modeling approach for monitoring a fleet of mining machines based on a software solution for traffic modeling. Computer simulations can help reduce prototyping costs and reduce the risk of initial launch failure by analyzing and tuning a prototype to test the most appropriate options. Using a computer modeling approach, we show in the first part of the article that the resulting vehicle monitoring metrics can be tested during the modeling process, instead of adding equipment to vehicles during the prototyping phase. Using real equipment in the prototype phase increases fleet downtime and decreases productivity. Using modern solutions for storing time series, we show how easy it is to analyze the data obtained as a result of modeling. In the second part of the article, we propose a workflow for integrating SUMO with a time series data warehouse through a software interface (API) called TraCI, which allows you to aggregate and visualize vehicle fleet data over time. At the end of this work, we discuss the measurement methodology and propose a potential solution for efficient data transmission.

## 1 Introduction

With current advances in Internet of Things (IoT) technologies, more and more enterprises are using these technologies in their business processes to better understand their workflows. The use of the "Internet of Things" along with big data technologies and cloud computing allows for constant monitoring of almost any device or vehicle. Analyzing this data can be useful in the mining industry, in particular because of the higher risks to human health and costly equipment damage. Despite the fact that there are already many solutions for monitoring equipment and personnel, there is still a lack of funds for monitoring the fleet of vehicles [1, 2], in particular in the mining industry. Recently, more and more solutions for monitoring transport are used [3], using cheaper solutions for computing devices and transport protocols, such as LoRaWAN [4].

Indeed, at the moment there is an abundance of solutions in the field of logistics and transportation of goods over long distances, and monitoring of transport using the global positioning system (GPS) is nothing new [5], however, most of the existing solutions can hardly be applied to the mining industry due to the specifics of the industry, differences in the vehicles themselves, requirements for the security policy of data transmission and

---

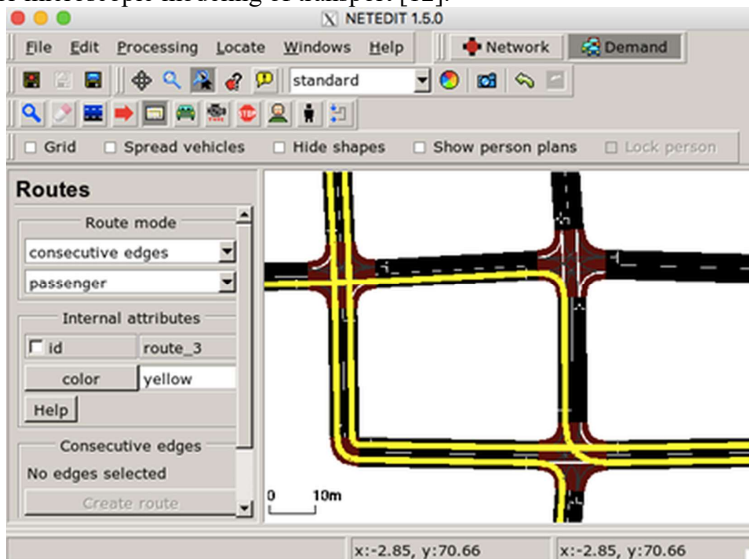
\* Corresponding author: [aafares@mail.ru](mailto:aafares@mail.ru)

interfaces of telemetry collection devices with vehicles. In their publication [6], they describe an approach to the observation of equipment used in mines, including the collection of data from cameras. While their work focuses more on the aspect of production management, we believe that it is also worth paying attention to the development of solutions for computer modeling and automatic collection of telemetry data from these models, controlling even parameters such as ambient temperature, humidity, pressure in tires and other variables that may be specific to this segment of the industry to ensure the safety and reliability of the mining process.

To achieve a reliable result when developing and implementing a new solution for monitoring or configuring an existing one, it is worth using computer modeling methods to reduce the cost of integration and the risks of failure in case of incorrect configuration. In fact, computer simulation programs are now common in many industries, especially those that involve higher risks of disaster if set up fails.

PTV Vissim [7] is one of the most advanced and modern solutions for computer simulation of traffic today, however, although it provides many useful functions typical for modeling urban traffic, these functions cannot be used in the context of vehicle modeling in mining industry. Therefore, for the purposes of this work, we decided to choose a smaller solution that still covers our needs in terms of functionality.

In this paper, we propose an approach to monitoring mining vehicles using urban transport modeling software (SUMO) [8, 9], applying it to the mining industry. We decided to use SUMO because it is one of the mature traffic simulation packages that is still in active development [10], and this is the tool we used to propose a method for verifying the configuration of a traffic management system [11]. In addition, the use of SUMO is due to the need for microscopic modeling of transport [12].



**Fig. 1.** Model creation process in SUMO.

Time series.

Most of the time, telemetry information is presented in the form of time series - points distributed in time, as a rule, at a regular interval. Examples of time series data are: air temperature taken at regular intervals, stock prices, etc. In the mining industry, there are measurements that may be of interest to a particular sub-sector of the industry. However, when it comes to monitoring a fleet of vehicles, there may be some common metrics that can be used to gain a better understanding of their performance.

The time series format is also convenient for monitoring, since it makes it easier to detect outliers in data [13], and the use of modern advances in machine learning allows faster and more accurate detection of anomalies in time series data [14].

There are many time series database solutions nowadays, some of them are tailored to specific needs such as monitoring software applications, while others are general purpose databases that can be used in a variety of projects, from data logging simulation prior to registration of telemetry devices of the "Internet of things". In fact, the development of time series data storage facilities is currently an active area of development [15, 16].

In this paper, we will use InfluxDB, which is a CDB that does not depend on any other DBMS using the SQL-like language InfluxDB Query Language (InfluxQL) [16] and has proven its work in IoT applications [17]. Our reasoning for using it is that while it is fairly new on the market compared to, for example, OpenTSDB, it has more features and, most importantly, an integrated dashboard, which frees us from installing a separate dashboard solution like like Grafana [18]. Of course, in real applications, the choice of a database depends not only on the appearance and functionality of the dashboard and the like, but this is a separate extensive topic beyond the scope of this article.

## 2 Method used

Model creation.

The model consists of two parts: the SUMO road network with a graphical representation of the mining site, and a Python program that will use TraCI to drive the simulation to access the simulation data and use it to collect telemetry data. Since it takes a lot of time to create a realistic scenario for traffic simulation [19], we will allow ourselves some assumptions, since our goal is to present the concept of using computer modeling in monitoring a mining fleet without reference to a real mining site.

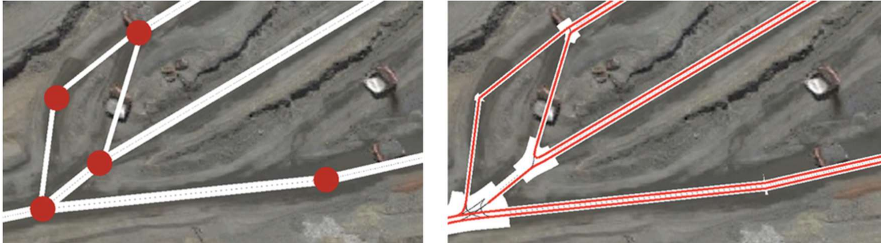
The SUMO model basically describes the road networks and routes that vehicles can travel along. Since this environment allows you to use a background image for the model, creating a map of the mining site should not be difficult if there is already a graphical representation of the terrain that will be used in the model. We will use a portion of Bingham Canyon in Utah, USA, as an example of representing a mining site using a satellite image freely available from GoogleMaps (Figure 2):



**Fig. 2.** Bingham Canyon, Utah (Copyright Google, copyright Maxar Technologies, State of Utah, USDA Farm Service Agency).

By importing this image as the background of the model, you can recreate a segment of the road network for that particular section. Of course, for real applications, the map can be larger and multiple background images can be used together. By placing intersections and

connecting them to edges, a road network is created that can then be used to define transport routes across the network (Figure 3).



**Fig. 3.** Connected junctions create a road network with routes.

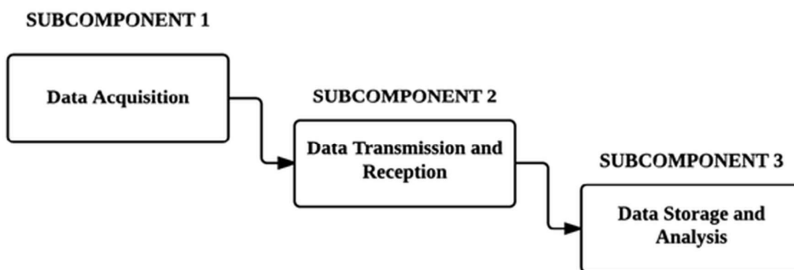
After creating the model, the next part of the model will be in a Python program that will use TraCI to run and control the simulation. Therefore, we will not run the simulation in SUMO itself, but with the help of a third-party program. In the same program, vehicle management should be done to unify the process in the future.

### 3 Results and Discussion

#### Developing a Python Program Using TraCI.

In this part of the article, we develop a model by creating a small Python program that will run and interact with a simulation created in NETEDIT. Our development environment looks like this: Python 3.7 programming language on macOS High Sierra 10.13 using pipenv virtual environment manager and the TraCI library that comes with the SUMO installation. This library allows you to run external simulations in server mode and automatically connect to it from the program. TraCIAPI allows us to manipulate the simulation by creating and deleting vehicles in the simulation by assigning any newly created vehicle to the corresponding route that was previously defined in the model. After the vehicles are created, we will request them at regular intervals to receive telemetry information. In this example, we will ask them to measure acceleration in meters per second squared (the default sumo unit for acceleration) and angle in degrees. Since we are using InfluxDB as our time series database, we are also using the Python library provided by InfluxDB. The InfluxDB instance runs on the same machine using the binaries provided on the official site.

Designing a simulation model includes three stages: data collection, data transfer and storage of the received data (Fig. 4), which fully describe the process of working with data from the vehicle fleet monitoring system.



**Fig. 4.** General data work processes in the monitoring system.

#### Integration with InfluxDB and data collection.

After setting up and running the model, the last step is to add integration with InfluxDB to collect and store time series data from the model. As mentioned earlier, this example

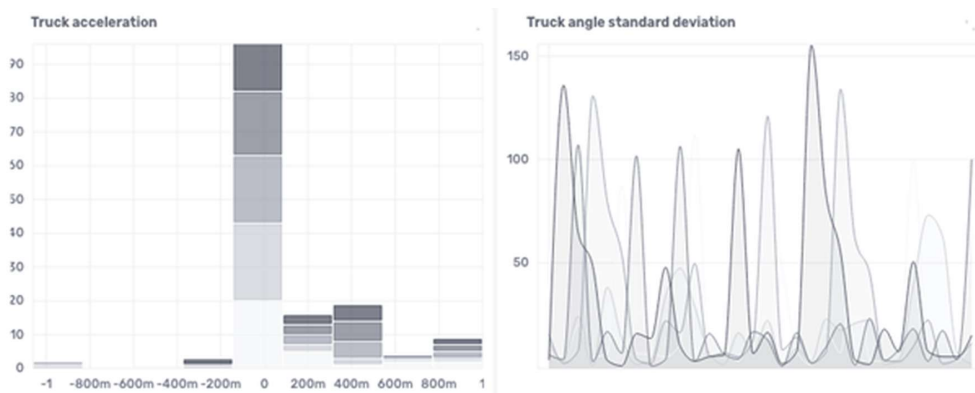
records the acceleration and slope of each vehicle during the simulation, which is done using the time series point record routine - this is possible with the Python client library. Since we are hosting the InfluxDB server on the same machine as the simulation, we can avoid network configuration steps such as setting up NAT and port forwarding.

Using the client API, it is possible to store each measurement from each vehicle at some predefined interval as a time series. In addition, in order to simulate the implementation of a real embedded system device that can often collect telemetry and then average the result, we will aggregate the measurements in each frame and by the time of sending, average the sum of all recent measurements by the number of collected samples, essentially simulating the real behavior.

After starting the simulation and recording the measurements into the database for some time, the next step is to enter the dashboard and set up the measurement visualization. Since InfluxDB comes with an already integrated GUI, we will use that instead of some additional solution.

Given the data we have, we will plot two graphs, one for each of our measurements:

- Median truck acceleration - this indicator is perhaps the most interesting, as it allows you to analyze the characteristics of vehicle movement. We'll show this metric as a stacked bar chart for each of the five trucks.
- Standard deviation of truck steering angle - Another metric that can help monitor driver behavior is angle and how it changes over time (Figure 5).



**Fig. 5.** Truck median acceleration in m/s<sup>2</sup> and angle standard deviation in degrees.

It should be noted that in our experiments we obtained rather high values of this metric, primarily due to the rather sharp angles of the trajectories in the model.

#### *Efficient data transmission and storage*

Efficient data transmission is of great importance in general and for vehicle telemetry in particular. LoRaWAN is currently showing promise as an efficient way to transfer data for IoT (Internet of Things) solutions. LoRaWAN is a low power broadband networking technology that has been successfully used in industrial IoT applications using mobile data networks [4].

In addition, using an efficient time series database is essential for large-scale applications. Despite the fact that in this work we used InfluxDB and they provide scalable solutions that showed good performance in a real IoT solution [17], choosing the right solution for storing time series depends on the organization and existing infrastructure, as well as general requirements for the volume of processed data traffic. Because of this, we advise you to pay attention to research specifically aimed at analyzing and comparing different time series of database solutions [15, 16].

## 4 Conclusions

In this paper, presented the concept of a system for collecting telemetric data from a mining machine park using a computer modeling approach. This approach can significantly reduce integration costs by avoiding vehicle downtime as both vehicles can be simulated and data collection is particularly important in the early stages of fleet management system integration. This approach can also help identify possible metrics that can be used to monitor vehicle fleets and modify those metrics during the post-processing phase of data analysis in a time series database dashboard. With the help of the SUMO tool for computer modeling of traffic, an exemplary environment for modeling a mountain area was created. This simulation was started externally by the program using the TraCI API, which gave more flexibility and the ability to receive telemetry information from the model environment. The collected telemetry was then stored in the InfluxDB time series database. By presenting the simulated test data, we have shown that this approach can be used to simulate telemetry collection scenarios that can assist in the management of a mining fleet.

This paper did not consider the feedback method, which could serve as an additional link in the personnel decision-making process. Potentially useful information can be obtained from telemetry data. Measured feedback can improve the efficiency of operations (for example, by choosing more optimal routes) or even the safety of workers by adding the ability to automatically send important notifications of events to staff based on telemetry data.

In further research, would like to introduce a metric scoring system that would help guide the decision-making process for fleet management tasks. As part of this framework, also would like to introduce a more robust collection of vehicle metrics such as vibration, humidity, using more advanced region and object features in SUMO that could help track vehicles during simulation. Along with these metrics, another important improvement would be the introduction of geographic position tracking by simulating GPS tracking using a coordinate mapping approach with translation of coordinates between geographic and Cartesian coordinate systems within the SUMO model.

## References

1. F.N. Abu-Abed, L.G. Naumova, **3(6)**, 21-35 (2019)
2. R. Safiullin, *Intelligent onboard transport systems in automobile transport* (Direct-Media, Moscow-Berlin, 2017)
3. S. Čaušević, A. Čolaković, A. Haskovic, *The model of transport monitoring application based on Internet of Things* (ISCSTD, Opatija, 2018)
4. J. Navarro-Ortiz, *IEEE Communications Magazine*, **2**, 60-67 (2018)
5. F. Abu-Abed, A. Khabarov *E3S Web of Conferences*, **41**, 01028 (2018)
6. S. Chaulya, G. Prasad, *Mine Transport Surveillance and Production Management System* (Springer, Berlin, 2016)
7. M. Fellendorf, P. Vortisch, *International Series in Operations Research & Management Science*, **145**, 63-93 (2010)
8. F. Abu-Abed, A. Ivanov, *Bulletin of the Kuzbass State Technical University*, **5**, 69-76 (2020). DOI: 10.26730/1999-4125-2020-5-69-76
9. F. Abu-Abed, *E3S Web of Conferences*, **41**, 01025 (2018)
10. D. Krajzewicz, J. Erdmann, M. Behrisch, L. Bieker // *International Journal on Advances in Systems and Measurements*, **1**, 128-138 (2012)

11. A. Ivanov, F. Abu-Abed, International Journal of Recent Technology and Engineering (IJRTE), **8(2)**, 4662-4666 (2019)
12. A.P. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, *Microscopic Traffic Simulation using SUMO* (21 ITSC, Maul, 2018)
13. Y. Flötteröd, R. Hilbrich, E. Wiessner, *21st International Conference on Intelligent Transportation Systems* (21 ITSC, Maul, 2018)
14. H. Jagadish, N. Koudas, Muthukrishnan, **99**, 7-10 (1999)
15. S. Kanarachos, S. Christopoulos, A. Chroneos, M. Fitzpatrick, Expert Systems with Applications, **1**, 292-304 (2017)
16. M. Fadhel, E. Sekerinski, S. Yao, *Comparison of Time Series Databases for Storing Water Quality Data* (Springer, Berlin, 2019)
17. A. Bader, O. Kopp, M. Falkenthal, *Survey and Comparison of Open Source Time Series Databases* (BTW, Prague, 2017)
18. M. Nasar, M. Kausar, International Journal of Innovative Technology and Exploring Engineering, **5**, 1850-1857 (2019)
19. Grafana Labs. *Grafana: The open observability platform*. URL: <https://grafana.com>
20. L. Bieker, D. Krajzewicz, A. Morra, C. Michelacci, F. Cartolano, Modeling Mobility with Open Data (AICS, Orlando, 2015)