Research Article

Shafiza Ariffin Kashinath, Salama A. Mostafa*, David Lim, Aida Mustapha, Hanayanti Hafit, and Rozanawati Darman

A general framework of multiple coordinative data fusion modules for real-time and heterogeneous data sources

Abstract: Designing a data-responsive system requires accurate input to ensure efficient results. The growth of technology in sensing methods and the needs of various kinds of data greatly impact data fusion (DF)-related study. A coordinative DF framework entails the participation of many subsystems or modules to produce coordinative features. These features are utilized to facilitate and improve solving certain domain problems. Consequently, this paper proposes a general Multiple Coordinative Data Fusion Modules (MCDFM) framework for real-time and heterogeneous data sources. We develop the MCDFM framework to adapt various DF application domains requiring macro and micro perspectives of the observed problems. This framework consists of preprocessing, filtering, and decision as key DF processing phases. These three phases integrate specific purpose algorithms or methods such as data cleaning and windowing methods for preprocessing, extended Kalman filter (EKF) for filtering, fuzzy logic for local decision, and software agents for coordinative decision. These methods perform tasks that assist in achieving local and coordinative decisions for each node in the network of the framework application domain. We illustrate and discuss the proposed framework in detail by taking a stretch of road intersections controlled by a traffic light controller (TLC) as a case study. The case study provides a clearer view of the way the proposed framework solves traffic congestion as a domain problem. We identify the traffic features that include the average vehicle count, average vehicle speed (km/h), average density (%), interval (s), and timestamp. The framework uses these features to identify three congestion periods, which are the nonpeak period with a congestion degree of 0.178 and a variance of 0.061, a medium peak period with a congestion degree of 0.588 and a variance of 0.0593, and a peak period with a congestion degree of 0.796 and a variance of 0.0296. The results of the TLC case study show that the framework provides various capabilities and flexibility features of both micro and macro views of the scenarios being observed and clearly presents viable solutions.

Keywords: data fusion, heterogeneous data, real-time processing, traffic state estimation, software agent, fuzzy logic

* **Corresponding author: Salama A. Mostafa,** Department of Software Engineering, Center of Intelligent and Autonomous Systems, Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, 86400 Johor, Malaysia, e-mail: salama@uthm.edu.my
Shafiza Ariffin Kashinath: Department of Software Engineering, Center of Intelligent and Autonomous Systems, Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, 86400 Johor, Malaysia; Engineering R&D Department, Sena Traffic Systems Sdn. Bhd., 57000 Kuala Lumpur, Malaysia, e-mail: shafiza@gmail.com.my
David Lim: Engineering R&D Department, Sena Traffic Systems Sdn. Bhd., 57000 Kuala Lumpur, Malaysia, e-mail: david.lim@senatraffic.com.my
**Aida Mustapha:** Department of Mathematics and Statistics, Faculty of Applied Sciences and Technology, Universiti Tun Hussein Onn Malaysia, 84600 Panchor, Johor, Malaysia, e-mail: aidam@uthm.edu.my
Hanayanti Hafit, Rozanawati Darman: Department of Information Security and Web Technology, Center of Intelligent and Autonomous Systems, Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, 86400 Johor, Malaysia, e-mail: hana@uthm.edu.my, zana@uthm.edu.my

1 Introduction

Utilizing various resources of data in a complex system environment necessitates an efficient framework to ensure the maximization of data usage to produce an efficient result. Data fusion (DF) involves a certain mechanism of combining data from diverse sources to provide comprehensive data measurement [1]. When implementing DF with diverse characteristics of data obtained from various sensing methods, the fusion result has a higher likelihood of being accurate [2]. Therefore, data richness and appropriate DF methodology are essential in finding a solution for a complex data-oriented system [3].

Providing an appropriate and efficient request-response mechanism in a real-time, distributed, and complex DF system is the most critical system's feature. Aside from reliability, this type of system should be designed with decision-making capabilities and be realistic and responsive according to real-world scenarios. Examples of such system environments are smart city management, e-commerce, Internet of Things (IoT)-based systems, traffic light control, and city management. An example of traffic sensing and flow analysis is shown in Figure 1.
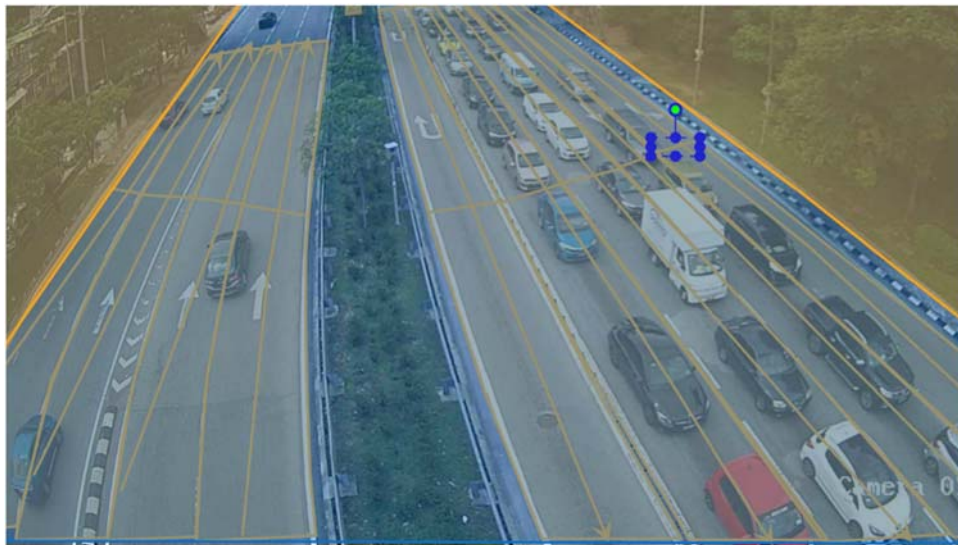
Figure 1: Kuala Lumpur congestion state.

Data feed in a real-time environment is manipulated by a DF technique and algorithm to improve different areas of the system. In various studies of many domains, DF proves its ability to improve decision-making [4], perform forecasting [5], state estimation [6], and even deal with missing data from the data collected [7]. At the end of each study, the result achieved is meant to enhance system operation to achieve better accuracy in the system's implementation. Regardless of the DF technique being implemented, the result achieves within small-scale observation-only, reflecting the real microlevel scenario [8]. In most circumstances, distributed system architecture necessitates interaction and information sharing among them to achieve a collective decision that accurately depicts the current situation. Collective decisions are critical, as they can improve the overall system performance and decision-making over time.

Performing DF for real-time and heterogeneous data sources is a challenging feat. It necessitates a variety of approaches to manage the data while considering the amount of processing time and complexity of system design. Several studies have demonstrated the importance of the coordinative decision in a distributed system architecture. The primary challenge addressed by this study is to formulate a combination of coordinative approach with multiple DF techniques which integrates real-time and heterogeneous data sources as a general framework. This framework aims to integrate the DF implementation model with

coordinative decision ability to enhance decision-making and system response when dealing with real-time and heterogeneous data.

The necessity to retrieve multimodal data demonstrates the importance of the data collection process to detect environmental changes. The ability to capture data from several sources allows the data-consuming system to do cross-checks and validations to ensure the consistency of the scenario being observed [9]. Each dataset may contain various data formats used by the DF method to produce some desired outcome. Real-time data are type of data that are updated regularly and given as acquired from its original source [10]. To ensure data reliability, systems that deal with real-time data typically require stable communication and sensor performance. Historical data are often used as a temporary measure when real-time data are unavailable for certain reasons [11].

Heterogeneous data are defined as data that have a wide range of variants and formats [12]. Despite the abundance of data features, processing heterogeneous data have their own challenges. Data accuracy is the essential criterion for any DF model; however, datasets may have issues such as uncertainty and incompleteness. To deal with this circumstance, an appropriate preprocessing methodology is required to ensure that the outcome of DF implementation is not impaired by data inaccuracy. Data are merged and aggregated in the first step of the DF stages, termed as preprocessing, to yield an enriched dataset.

This paper presents a general framework of Multiple Coordinative Data Fusion Modules (MCDFM) for heterogeneous and real-time data sources. The framework is intended to create a flexible solution that coordinates the collective decision by considering each node in a network to have a better picture of the real condition rather than relying on the decision state determined by the individual DF module. The major contributions of this work are as follows:

- Propose a Multiple Coordinative Data Fusion Modules (MCDFM) framework for environments with heterogeneous and real-time data sources. The proposed framework is meant to provide a robust, flexible, and responsive coordinative system with multiple data fusion design.
- Identify the major phases of the general DF framework and their specific purpose algorithms or methods such as data cleaning, windowing methods for preprocessing, extended Kalman filter (EKF) for filtering, fuzzy logic for local decision, and software agents for coordinative decision.
- Apply a traffic light control (TLC) system as a case study of a specific domain problem. A few conceptual scenarios of congestion conditions are used as examples in the case study.

The paper content is organized as follows: Section 2 reviews the literature on the standard DF framework and local and coordinative decisions. Section 3 presents the proposed MCDFM framework, while Section 4 describes a case study with example scenarios to show the applicability of the MCDFM framework. Section 5 concludes the paper.

2 Related works

This research aims to formulate a general framework that is applicable in various domain problems. The framework employs a coordinated approach and multiple data fusion techniques. A DF approach is divided into three groups: (1) statistical approach, (2) probabilistic approach, and (3) artificial intelligence [13]. From our observation of the literature, a basic DF implementation framework consists of the following standard processes: (1) preprocessing, (2) filtering, and (3) decision [1,14]. Preprocessing is the data collection stage, where a set of activities are performed on the datasets to obtain well-formatted data before forming sets of features. Data reliability needs to be considered as this is a prerequisite of achieving a good quality result. Removing noise, taking out outliers from the dataset, data normalization, conversion, and filtering are common methods for achieving data format standardization [15]. For heterogeneous kinds of input, various data need to be fused to produce the most useful features. At this stage, feature combination and filtering are performed on well-formed data. The filtering process tends to produce estimation, forecasting, and behavior prediction over a period based on the observed scenario [16]. Various features

produced from the previous stage are fused to achieve the best decision that fits the selected decision model or algorithm [17].

Observation and decision for collective elements provide meaningful insights to solve any domain problem compared to having them independently. Shahrbabaki et al. [18] proposed a DF model that performs traffic state estimation in signalized links that are incorporated with heterogeneous data from connected vehicles and loop sensors. This model, which is tested on a few links, gives the idea of vehicle accumulation in the downstream, upstream, and the whole link. The study shows the importance of having collective insights among all the nodes in the road network to better predict or estimate a global view. Akbar et al. [19] proposed a combination of a probabilistic model and Bayesian network technique on the heterogeneous data source for IoT applications. Various data inputs such as weather, event, time segment (morning, evening, etc.), and day (weekend or weekdays) are required to achieve the probability of real-time congestion. A real-time system requires scheduled input within a specific time interval to ensure the continued effectiveness of system operation, and this is the natural setup for MCDFM.

Izumi and Azuma [20] presented a DF model that does real-time pricing of power consumption on networks. An information-sharing mechanism is required in this model to achieve the desired result. They adopt distributed estimation in their model by implementing DF on the network. The study indicates the need for collective insights when dealing with network domain problems, and this is one of the features of a software agent. Saeedmanesh et al. [21] discussed multi-region of traffic state estimation. Extended Kalman Filter is chosen as a technique to produce the estimation. They break the urban network into predefined regions, and one traffic state variable is achieved per region, which summarizes each condition. Measurement of each region is collected and aggregated to achieve demand trajectories at the end of the study.

There are various ways of achieving coordinative decisions in the traffic-related study, such as software agent [22], ant colony algorithm [23,24], internet of agents (IoA) [25], artificial fish swarm algorithm (AFSA) [26], and junction-tree algorithm (JTA) [27]. The agent-based system approach acts as an independent entity that performs an autonomous task due to the observed system's state changes [28]. The software agent communicates and shares resources among each subsystem in the network, where each subsystem is treated as a node [14]. The system architecture's flexibility offers a great opportunity to have a macro view of the subject being observed [29].

Xu et al. [22] proved how an agent-based approach coordinates among TLC in a region to achieve a collective decision. Mostafa et al. [30] proposed a general framework of adjustable autonomy by implementing Fuzzy Logic as a decision-making algorithm. The study emphasizes the flexibility of the autonomous framework to respond to the environment by making full usage of the agent-based concept. This idea proves that the combination of DF and the agent-based system has a great potential to be further explored in the future. Bienzeisler et al. [31] proposed a model of DF in an agent-based environment. The study breaks the home locations in Hanover into separate regions to estimate work place distributions. Each agent performs their task to gather information from various areas in the city, including locations, travel patterns, vehicle volume, and people. The study proves that having an agent element in a framework provides a dynamic handling mechanism of a system, from a small-scale to a complex and huge system.

Kumar et al. [23] applied an ant colony algorithm to map each segment of Vellore district, Tamil Nadu, India, to identify the best route to reach a destination. Data collection is implemented by using the Internet of Vehicles (IoV) approach used by the ant colony algorithm to update each segment of the covered area. Rehman et al. [24] proposed a framework to optimize the route and city traffic by utilizing an ant colony optimization algorithm. The study is intended to find the best route by considering capacity, density, congestion level, and travel time based on information gathered from various intersections.

Bui and Jung [25] proposed a concept of IoA, which connects vehicles as a collaborative model that allows various agents to communicate without a centralized device to handle the communications. Agents have the ability to make their own decisions based on their observation to improve road traffic conditions by automatic negotiation with another agent. Ma and He [26] proposed a green wave traffic control system by adopting a genetic algorithm and AFSA, in which each multiple artificial fishes cooperate to find optimum solutions. The authors implement this model in a case study that consists of five consecutive intersections to execute green wave control to ensure smooth vehicle movement from an intersection to another. Lu et al.

[32] proposed an enhanced AFSA to improve traffic signal control efficiency by considering parameter estimation and global optimization of the road network to reduce delays and stopping times.

Zhao et al. [27] developed a JTA model to achieve traffic conditions at the intersection level. JTA with reinforcement learning (RL) algorithm treats each intersection as a local problem. Each local decision contributes in determining traffic coordination control decisions. Based on the simulation conducted, JTA-RL shows great improvement to both individual intersection and network traffic conditions. Zhu et al. [33] proposed a JTA-RL algorithm model to enhance coordination implementation of the intersections network. The test results show the benefit of observing traffic conditions as network agents compared to an independent agent.

For a data-driven system that deals with various sensors and data formats within a distributed, complex, and dynamic system design, the coordinative approach is one of the critical features. Interactions between independent entities in a distributed system indicate that information exchange and knowledge sharing are crucial mechanisms in designing coordinative frameworks [25]. The coordinative decision is derived from the interactions of various subsystems or modules that share domain knowledge and make a decision that meets certain criteria. In most circumstances, the collective decision allows for a clear macro view of the domain problem rather than relying on an individual perspective [22,31]. Furthermore, coordinative frameworks are resilient and are used in a variety of domain problems since they are flexible, responsive, and autonomous [30].

There are a few studies that show the needs of DF frameworks with a coordinative approach. Moattari and Majd [34] proposed a cooperative data fusion framework that consists of a sensor network to perform the distributed estimation. A compatible packet-based communication through a wireless sensor network lets each node in the network share local estimation with all neighbors before centralized estimation is produced. The study wanted to achieve a precise estimation by collecting local decisions at a central level. He et al. [35] proposed a centralized state estimator that processes local estimations from all sensors in the wireless network via a fusion center. The framework utilizes a multi-sensor JPDA filter to solve the tracking of distributed multiple targets. Fortino et al. [36] proposed a solution that combines the C-SPINE framework and multiple sensor DF in a Collaborative Body Sensor Network (CBSN) study. C-SPINE is a collaborative framework with a specific communication protocol and the ability to perform collaborative processing. The local decision represents the extracted features in the sensor fusion architecture of C-SPINE. The higher-level decision-maker fuses a combination of extracted features. Bienzeisler et al. [37] proposed a centralized DF system to deal with centrally collected data related to Hanover city. Each population group is treated as an agent, and they describe travel patterns, activity patterns, and trip distances for each group. MATSim simulation framework is used to describe the proposed approach. Elmas and Sönmez [38] proposed a DF framework to estimate fire danger levels, fire spreading speed, and forest fire detection. The study enhances the fire danger rating algorithm in Forest Fire Decision Support System (FOFDESS), which is a multi-agent framework for the forest fire Decision Support System. Ahmed and Abdel-Aty [39] conducted a study on real-time risk assessment on freeways. They focus on data collected from various sensors as well as handling mechanisms when some of the sources failed to provide the necessary data. Even if they do not mention coordinative decisions, having this functionality is important if the covered area grows. Table 1 shows the DF frameworks of the related works based on our analysis, including the characteristics of real-time and heterogeneous data sources.

In summary, Fortino et al. [36] utilized the CBSN framework and multi-sensor data fusion to detect emotion. Multi-sensor data fusion enhances data filtering to be incorporated into existing CBSN. Moattari and Majd [34] proposed a cooperative fusion of target position estimation in a heterogeneous network. They use a wireless network as a means of communication to transmit and share local estimation for centralized decisions. He et al. [35] conducted a study on a centralized fusion strategy for a distributed multi-target tracking of sensor networks. A centralized fusion center processes local estimations to generate a good estimation. Bienzeisler et al. [37] proposed an agent-based simulation and DF framework to model Hanover city. An agent represents each population group. However, in this study, the details of the DF model and implementation are not discussed. Elmas and Sönmez [38] utilized an existing multi-agent-based framework known as FOFDESS to enhance the fire danger rating mechanism. They merge the data fusion framework to better estimate the overall performance of the forest fire decision support system. This indicates that any decision support system that works around various data formats requires a holistic framework that supports the decision-making process, coordinative mechanism, and data processing.

Table 1: DF frameworks characteristics of real-time and heterogeneous data sources

Domain	Framework description	Global decision	Research goal	Ref.
Sensor	A cooperative data fusion framework for heterogeneous sensor networks made up of multiple nodes that perform both local and centralize estimation	Centralize estimation	Perform target position estimation & information sharing	[34]
Sensor	A distributed framework for large-scale sensor network to obtain local estimation and perform information fusion	Centralize estimation	Sensor fusion of sensor network	[35]
Health care	A framework that adopts collaborative body sensor networks (CBSN) which consist of sensing layer, analysis layer, and dissemination layer to perform DF on multi-sensor fusion	Collaborative detection	Automatic detection of handshakes and emotion reactions	[36]
Transport model	A centralized DF framework to improve quality of living in Hanover city based on various sources of data.	Agent-based	Combination of centralized DF agents to enhance transportation model	[37]
Forest fire	A DF framework that collaborated with multi-agent of forest fire decision support system to achieve fire danger rating, fire spread speed estimation, and fire detection	Multi-agent	Forest fire detection, prediction, and management	[38]
Freeway	A framework for assessing freeway conditions that incorporates data from different sources. This framework includes several techniques for data fusion, risk assessment, and protection to addresses several issues including travel time estimation and route guidance	Not discussed	Real-time assessment of a freeway	[39]

3 The multiple coordinative data fusion modules framework

A coordinative framework consists of several subsystems that work together to improve the decision-making process. This approach has been applied to various kinds of the distributed system [40]. Network communication, system complexity, and storage facilities are some of the criteria that must be considered in the development of coordinated solutions [41,42]. Each independent subsystem in the network performs a specific task assigned to it [43]. The ability to exchange local decisions through a dedicated communication channel is essential, but accomplishing a coordinative decision requires more than just that. Responsive, scalable, intelligent, and autonomous are some of the common features of coordinative decision [42], regardless of techniques implemented, as discussed in Section 2. This section presents a general framework of MCDFM. A detailed discussion is further elaborated in the following subtopics.

3.1 Modelling and design

In coordinative network approaches, the ability to share information among subsystems in the network is important to enhance the network-based system's judgment and decision-making process. Subsequently, in the MCDFM framework, we assume that each node in the network performs DF to produce local decisions before performing any global data manipulation with a coordinative decision approach. The MCDFM consists of several DF modules. For each module, there are three main DF processing phases: preprocessing, filtering, and decision. Figure 2 shows the MCDFM framework that consists of three main stages to achieve local and coordinative decisions. The following subtopics provide a brief overview of the preprocessing, filtering, and decision-making processes that are applied in Section 4.

The decision for each DF module, indicated by $DF_{1...n}$ is known as a local decision. This module evaluates the condition or state of the data source variables being observed over time. Agent-based module
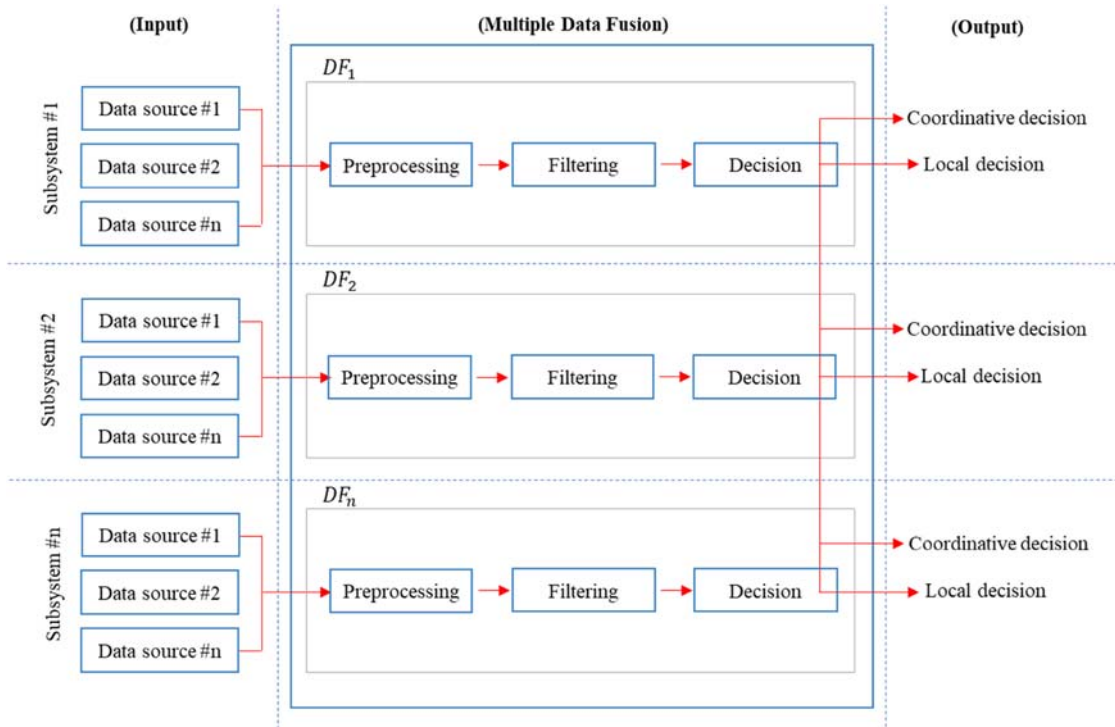

Figure 2: The MCDFM framework.

provides mechanisms that play essential roles in achieving coordinative decision. The agents enable the modules to interact in attempt to better describe an observed situation and make accurate decisions. Depending on the selected DF method's ability, knowledge sharing and information exchange are performed by the agents to achieve a decision at a higher or coordinative level.

Let x represents the data that needs to be incorporated, which consists of $\{x_1, x_2, ..., x_n\}$. x represents various kinds of data from various sources such as sensors or cloud-based systems, in the form of historical or real-time data. Let P be the preprocessing algorithms of the proposed DF module, which consists of $\{p_1, p_2, ..., p_n\}$. During the P stage, various algorithms are implemented depending on the sensing method and type of data collected to ensure data completeness, consistency, and format. Let F represents the filtering technique, which consists of $\{f_1, f_2, ..., f_n\}$. The complexity of the sets of data has a close relationship with the selection of methods to produce an estimation of features. For example, homogeneous and heterogeneous data may require different DF methods, and in some scenarios, a combination of both data complexities requires more than one method to perform feature estimation.

Let D be the decision method that is incorporated in the proposed module to achieve the desired outcome. Each block which consists of P, F and D is denoted with DF, which consists of $\{DF_1, DF_2,..., DF_n\}$. As shown in Figure 2, each DF module produces a local decision at the output. Let D^l represents the local decision for each DF, which consists of $\{d_1^l, d_2^l, ..., d_n^l\}$. Let D^c be the coordinative output, which consists of $\{d_1^c, d_2^c, ..., d_n^c\}$.

All data types are fed to each DF module (denoted with DF_1 to DF_n). At the output of the DF preprocessing, filtering, and decision, the result D^l is produced. The MCDFM framework contains n number of DF modules that contribute to the coordinative approach. A coordinative output, D^c, is achieved after reasoning and manipulation of all D^ls made by each DF module. We design the proposed framework to fit a data-responsive application that requires a high-efficiency rate and accuracy. However, the efficiency and accuracy are tightly related to the chosen algorithms in the preprocessing, filtering, and decision phases. For MCDFM discussion purpose, we provide a conceptual example scenario of the TLC system to understand the implementation better. This issue is elaborated on in Section 4.

3.1.1 Preprocessing

Preprocessing is a low-level fusion that gathers data in its raw form. In this phase, certain data preprocessing is required to avoid data incompleteness, inaccuracy, and inconsistency [44]. In this discussion, examples of preprocessing methods are data cleaning and windowing. Some data conversion is required to produce well-formatted data, and incomplete data must be handled for subsequent processing. Data is processed in a batch where each batch consists of a group of data that appears within a certain time interval. In this example, each measurement is set for every 60 s.

3.1.2 Filtering

Filtering is a process of integrating various data sources to fuse features from the datasets [45]. For the example scenario, we choose the Extended Kalman Filter (EKF) as an estimation technique, which is an estimation technique for a nonlinear system model. This approach focuses on two main steps called prediction, as shown in (1) and update step (2).

$$x_t = A \cdot x_{t-1} + w_{t-1} \tag{1}$$

$$z_t = H \cdot x_t + v_t \tag{2}$$

x_t represents estimation state, z_t represents measurement, w_t represents noise process, v_t represents measurement noise, and A represents state matrix.

3.1.3 Decision

The decision phase consists of decision-making algorithms that use the features as input and produce more specific features or classes. The local decision represents the decision made based on the local data, while the coordinative decision represents the decision made based on the manipulation of the local decisions. The decision-making algorithm is statistical, logical, or heuristic, depending on the data and the application. Each TLC is treated as an agent, and each agent produces a local decision by implementing fuzzy logic. Fuzzy logic is a technique that measures certain states within 0 and 1. There are sets of rules that need to be defined according to the features that need to be considered in the decision-making process [46]. To achieve a coordinative decision, each agent interacts and shares information. A software agent is chosen as a coordinative agent that makes the coordinative decision and provides a particular response to each subsystem [47].

3.2 Case study

The proposed framework is intended for any field of study that requires MCDFM to support the system operation. This case study explains the implementation of the MCDFM framework by considering the TLC system. The case study scenario focuses on three road intersections along with Jalan Klang Lama, Kuala Lumpur, which are labeled as S903, S904, and S905. Figure 3 shows the location of the MCDFM framework case study.
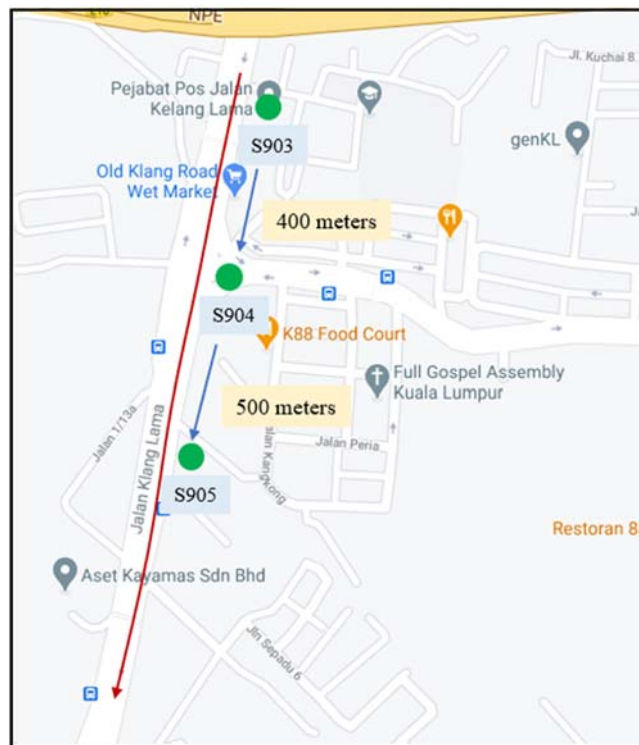
Figure 3: The location of the case study in Kuala Lumpur.

Each TLC is equipped with its own control unit to handle traffic operations and to collect raw traffic data. The intersections network faces heavy traffic flow during a certain time of the day from S903 (Kuala Lumpur), going down to S905 (Puchong). The distance between S903 and S904 is 450 meters, while S904 to
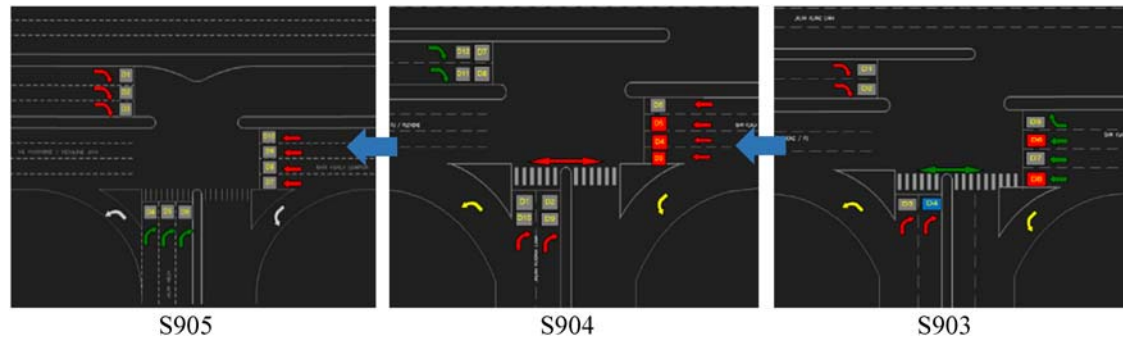
Figure 4: Heavy flow direction from S903 to S905 (right to the left).

S905 is 500 meters. Figure 4 shows the overview of the heavy flow directions. The blue color arrows show heavy traffic direction movement from one intersection to the adjacent junction.

Due to the short distance between S904 and S903, S904 tends to have a spillover queue that interrupts S903 performance during the peak hours of the day. During this condition, whenever the green signal is given to KL – Puchong direction from S903 to S904, no car can leave the yellow box at S903 due to S904 overwhelmed traffic condition. This scenario proves that other intersections can influence each intersection's performance in the same movement direction, which indicates that they should be observed as a network of intersections. The coordinative frameworks can give a better overview of intersection conditions to improve TLC decision-making.

4 Testing and results

4.1 Test setting

This section further discusses the applicability of the MCDFM framework in a TLC system. The major goal of this case study is to show how the suggested framework addresses traffic congestion as a domain problem. Table 2 shows example data of S904 for movement direction from north to south.

Table 2: Raw data samples of S904

Scenario	Raw data					
	Heading	Source	Speed	Lat	Lon	Time
Nonpeak period	N-S	GPS16	60	33.71243	112.1047	11:04:01
		GPS17	65	33.70251	112.1071	11:04:01
		GPS18	55	33.70194	112.1070	11:04:01
		GPS16	62	33.71248	112.1064	11:04:04
		GPS17	72	33.70247	112.1147	11:04:06
Medium peak period	N-S	GPS19	61	33.70145	112.1045	16:30:11
		GPS20	55	33.71021	112.1104	16:30:12
		GPS19	55	33.70123	112.1078	16:30:14
		GPS20	50	33.71042	112.1181	16:30:15
		GPS21	52	33.71248	112.1045	16:30:15
Peak period	N-S	GPS22	35	33.84397	112.1347	18:05:12
		GPS22	30	33.84397	112.1347	18:05:20
		GPS23	25	33.84404	112.1356	18:05:13
		GPS23	24	33.84404	112.1356	18:05:20
		GPS24	20	33.84405	112.1357	18:05:30

The sample data is divided into three categories, which are nonpeak, medium peak, and peak periods. Available features in the example data are source, speed, latitude (lat), longitude (lon), and time. The heading indicates the vehicle movement direction where "N" stands for north while "S" refers to south. The source represents each individual vehicle identified as GPS16, GPS17, GPS18, GPS19, GPS20, GPS21, GPS22, GP23, and GPS24. Speed represents vehicle speed, while lat and lon represent latitude and longitude of a vehicle. Time in the sample data indicates the time a record is captured.

4.2 Test scenario

Three different scenarios are identified: nonpeak, medium peak, and peak periods. However, only a few sample data are included for discussion purposes to emphasize three different traffic scenarios. Each unique source (sample data in Table 2) represent a vehicle. The accumulated count of unique sources within a predefined time interval represents the number of vehicles at the intersection. The speed column in the data holds the speed of vehicles at a specific location within a specific time interval. Finally, the location of each vehicle describes the distance or gap between vehicles on lanes. These five kinds of input go through preprocessing, filtering, and decision phases, based on selecting suitable methods for each phase. DF techniques chosen for the MCDFM framework are shown in Table 3.

Table 3: Modelling and setting of the MCDFM framework

Data (x)	Preprocessing (P)	Filtering (F)	Decision (D)	Coordinative decision
Source Speed Latitude Longitude Time	Data cleaning, windowing	Extended Kalman Filter (EKF)	Fuzzy Logic	Software agent

The techniques utilized for preprocessing steps are data cleansing and windowing. The estimation technique chosen for the MCDFM framework is EKF, while Fuzzy Logic is used as a decision phase method to produce a congestion level at an intersection.

In this scenario, each TLC acts as an independent DF module (DF_1, DF_2,…, DF_n) of MCDFM; therefore, each TLC is represented as DF_1 (S903), DF_2 (S904), and DF_3 (S905). Each TLC collects the required traffic data ($\{x_1, x_2, …, x_n\}$), which in this case, are speed, source, time, latitude, and longitude within a certain interval of time (t). After performing preprocessing ($\{p_1, p_2, …, p_n\}$), filtering (F), and decision (D), local decision (y) is generated. Average count, average speed, and average density are features that are produced within a specific interval (60 s) from sets of data. The average speed feature is generated from the estimation of single column data over time, while average count and average density are derived from a combination of more than one column that exists in the dataset. Fuzzy logic utilizes all of the features to determine a crisp set of congestion indicators. Table 4 shows traffic features and their variable states.

Table 4: Traffic features and their states

Feature	State
Average count	Low, medium, and high
Average speed	Low speed, medium speed, and high speed
Average density	Low density, medium density, and high density

Fuzzy sets of average count features are identified as low volume, medium volume, and high volume. This value tells how many vehicles pass through an intersection within a specific time interval. Average speed indicates the distance a vehicle travels within a specific time interval. This variable is categorized as low speed, medium speed, and high speed. Figure 5 shows the coordinative calculation of (a) minimum speed, (b) average speed, and (c) maximum speed to find a relational pattern between the streets in terms of speed.

The average density measures the number of vehicles occupying a certain length of the road segment by lane. This feature reflects the actual vehicle flow at a specified time, and usually, movement patterns are easily predicted during the nonpeak hour and peak hour periods. It is identified as low density, medium density, and high density. Figure 6 shows the average density calculation in the coordinate approach to obtain the global view of the coordinate features.

Fuzzy logic fuses all of these features to achieve a decision for the individual TLC. Table 5 shows an example of the decision of fuzzy logic. In this example, fuzzy rules are set in five congestion level indicators: very low congestion, low congestion, medium congestion, high congestion, and very high congestion. In Table 5, only three examples are given to demonstrate fuzzy membership. These membership functions need to be converted to a crisp value that drives the system to better decisions and improvements.

This congestion indicator describes the situation of the covered area of each individual TLC at the microlevel. The coordinative approach has an important role to play to obtain a global view of the intersections network. The interactions between the *DF* modules ($\{DF_1, DF_2 ..., DF_n\}$) gather local decisions to improve the overview of the road condition for certain vehicle movement direction. For example, the heaviest flow and heavy traffic jam created at S905 are the microlevel of observation, but to solve this traffic issue, a clear picture of the whole network condition (from S903 to S905) is considered. Figure 7 shows the GUI of Sena Traffic Control Center (Trafficsens) of the case study.

4.3 Results and discussion

Contingency actions provide various mechanisms to deal with unforeseen real-time circumstances at intersections. TLC operations at S903 and S904 should be responsive and adaptive to the current condition in order for S905 to avoid the worst-case scenario. A coordinative decision visualizes the big picture of the problem being observed, and this is how the response mechanism reacts according to the need of the system. For TLC-related study, coordinative output could help in determining traffic movement with the heaviest flow, incident detection, sensor fault, and green time optimization. Table 6 shows the result of *DF* module based on the features derived from three different test scenarios at S904. This result is presented in 60 s time frame.

In this case study, road users along Jalan Klang Lama experience heavy traffic during peak periods in the morning, when most vehicles enter the city center and leave the city toward the residential areas in the evening. The nonpeak period occurs during holiday seasons or public holidays. At other times, road users experience the medium peak periods, especially for Kuala Lumpur – Puchong direction (S903 to S905).

From the results, the intersections experience very smooth traffic movement during the nonpeak periods by achieving very low congestion levels. This condition occurs in two different ways. First, when there are fewer vehicles on the road with medium speed but low density, or a medium number of vehicles with medium speed and low density. Second, during the medium peak period, vehicle speed and density are at the average level. This scenario illustrates a smooth traffic movement with the average number of stops compared to the previous scenario.

The peak period is when almost every vehicle in the lanes experiences the greatest number of stops. This means fewer vehicles pass through within a certain interval, with a low average speed value and high density. This is the condition when the road is fully occupied, having a high congestion level. Each traffic data from the intersections in the network needs to go through the same DF stages (preprocessing, filtering, and decision) to produce a congestion level value as a local decision. An example of a local decision is
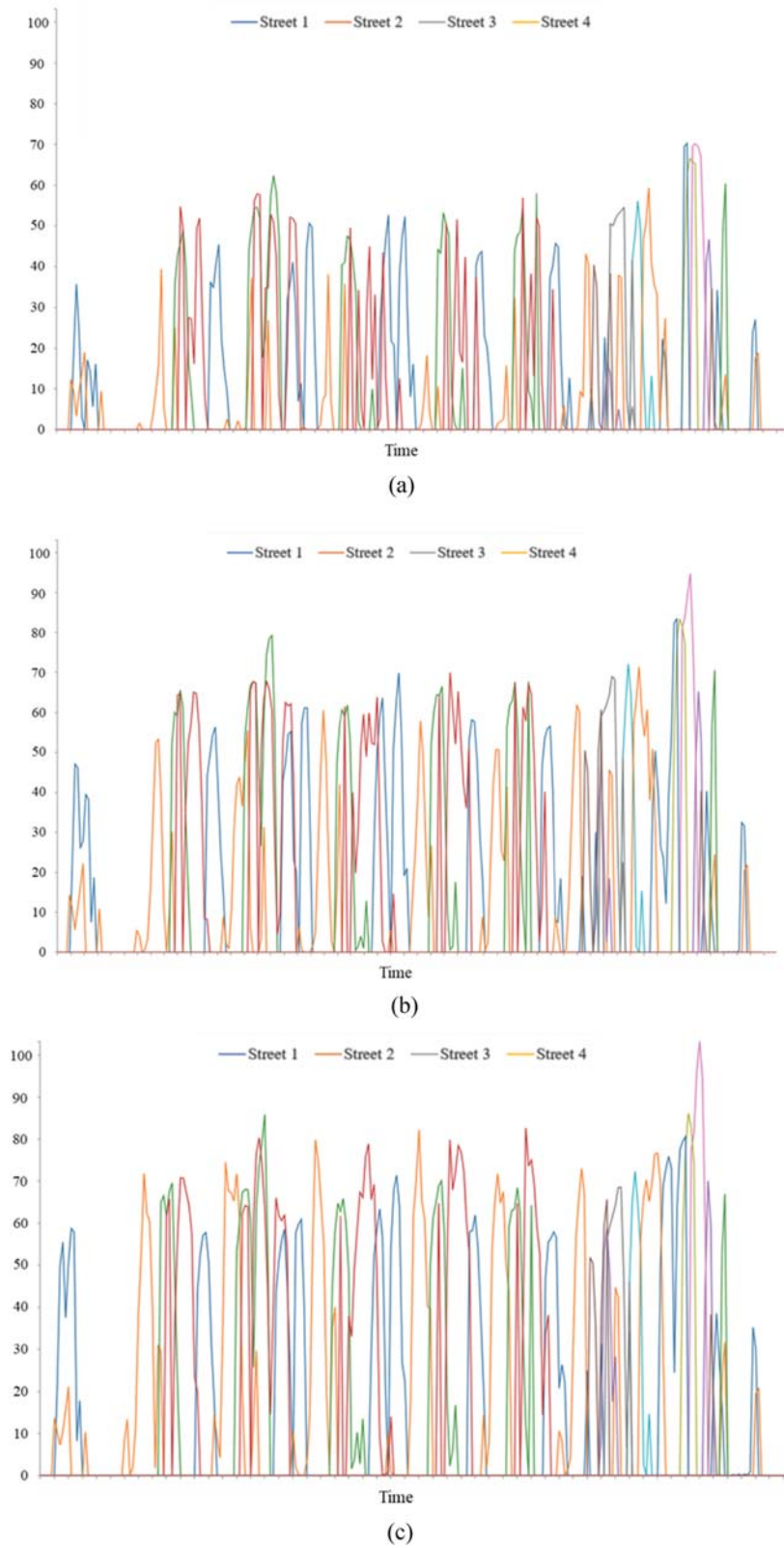
Figure 5: Speed coordinative calculation: (a) minimum speed calculation, (b) average speed calculation, and (c) maximum speed calculation.
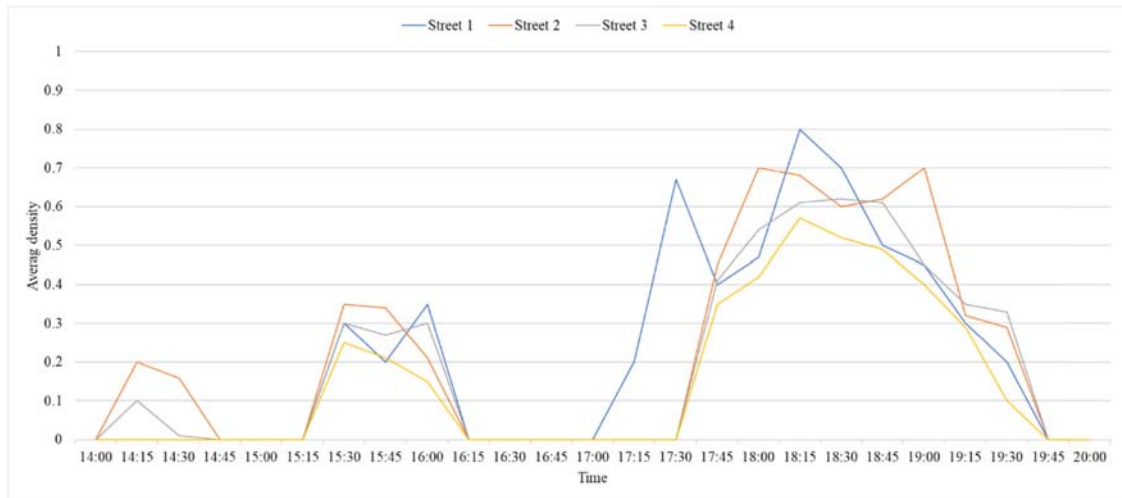
Figure 6: Calculation of average density as a coordinate feature.

Table 5: Decision rule examples of fuzzy logic

| Fuzzy rules	Congestion level
IF average count is *Low* **AND** average speed is *High speed* AND average density is *Low density* **THEN** congestion level is *Very low congestion*	Very low congestion (VLC)
IF average count is *Low* **AND** average speed is *Medium speed* AND average density is *Low density* **THEN** congestion level is *Low congestion*	Low congestion (LC)
IF average count is *Medium* **AND** average speed is *Medium speed* AND average density is *Medium density* **THEN** congestion level is *Medium congestion*	Medium congestion (MC)
IF average count is *Low* **AND** average speed is *Low speed* AND average density is *Medium density* **THEN** congestion level is *High congestion*	High congestion (HC)
IF average count is *Low* **AND** average speed is *Low speed* AND average density is *High density* **THEN** congestion level is *High congestion*	Very high congestion (VHC)
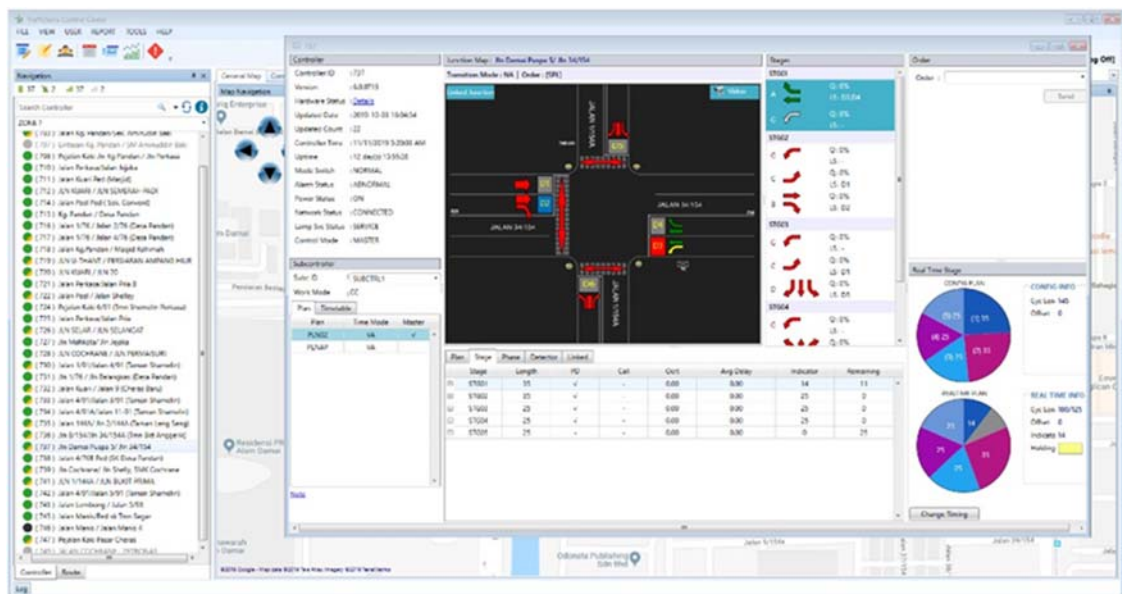
Figure 7: The Sena traffic control center (Trafficsens).

Table 6: DF results at S904

Scenario	Features					Local decision
	Avg count	Avg speed (km/h)	Avg density (%)	Interval (s)	Time	Congestion level
Nonpeak period	4	45	34	60	11:04:00	0.25
	3	50	31	60	11:05:00	0.24
	4	61	19	60	11:06:00	0.14
	4	60	23	60	11:07:00	0.14
	3	72	17	60	11:08:00	0.12
Medium peak period	3	40	52	60	16:30:00	0.64
	2	60	61	60	16:31:00	0.56
	2	62	50	60	16:32:00	0.50
	1	50	73	60	16:33:00	0.60
	2	45	67	60	16:34:00	0.64
Peak period	1	20	76	60	18:05:00	0.79
	1	24	87	60	18:06:00	0.83
	2	35	85	60	18:07:00	0.75
	1	25	91	60	18:08:00	0.80
	1	25	95	60	18:09:00	0.81

shown in Figure 8, which shows five congestion levels (very low congestion, VLC; low congestion, LC; medium congestion, MC; high congestion, HC; and very high congestion, VHC).
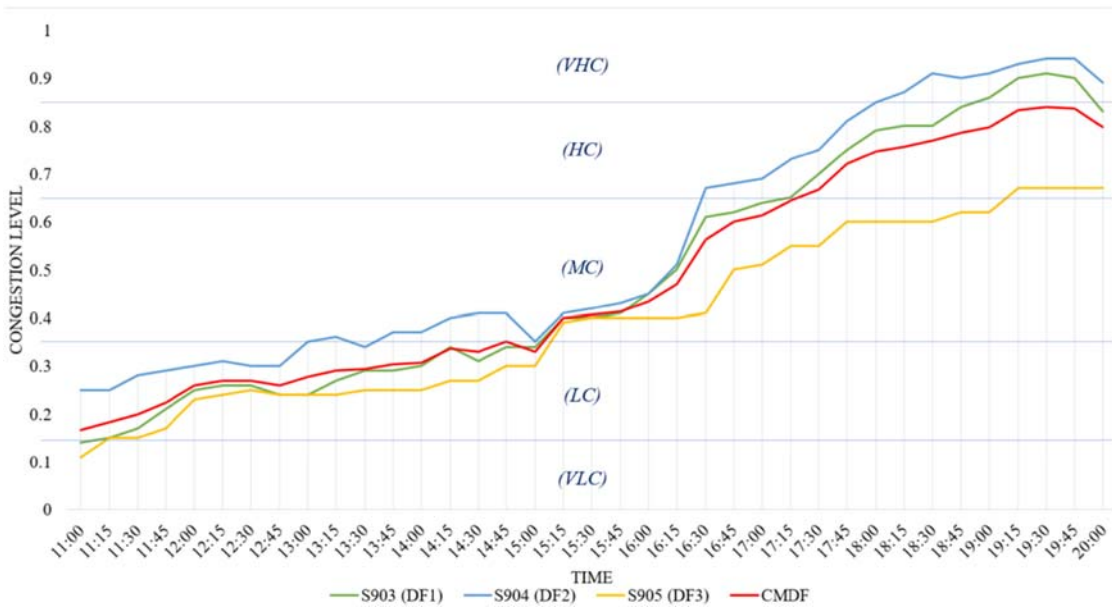

Figure 8: The local and coordinative decisions of each DFM.

A congestion level gives the idea of each intersection's current situation, whether it is experiencing smooth vehicle movement or suffering from overloaded conditions. Figure 8 depicts a 15-minute congestion level situation for all S903 (DF_1), S904 (DF_2), and S905 (DF_3) and the coordinative MCDFM. The faded blue line in the chart shows the boundary of five congestion levels (VLC, LC, MC, HC, and VHC). Table 7 shows a summary of the DF module's decisions that describe the transition of traffic conditions over time.

Table 7: The decisions of the DFMs at each intersection

Scenario	Time of day	Congestion level			
		S903	**S904**	**S905**	**MCDFM**
Nonpeak period	11:00	VLC	LC	VLC	VL
	15:15	LC	LC	LC	LC
Medium peak period	16:30	MC	HC	MC	MC
Peak period	17:45	HC	HC	MC	HC
	19:00	VHC	VHC	MC	HC

The collective output from each *DF* module is the essential step to achieve a coordinative decision. In this example scenario, each TLC acts as a software agent that provides autonomous response to the environment condition. The reaction depends on the situation the intersections network is experiencing as the outcome of the coordinative decision. Based on the scenario discussed, the coordinative decision informs which movement direction suffers from HC and VHC condition together with the time. With coordinative decisions achieved, various control mechanisms can be deployed to improve traffic conditions. Based on Table 7, during the nonpeak periods, all the intersections experience VLC and LC, which indicates that the green time duration given to each TLC manages to control the number of vehicles that pass through each intersection.

The medium peak period shows that S903, S905, and MCDFM experience MC conditions, and only S905 experiences HC with high vehicle volume. However, during the peak period, S903 and S904 experience VHC conditions, S905 experiences MC, and the entire route of the MCDFM experiences HC. Consequently, road users experience a higher number of stops to wait for green, stuck in the yellow crisscross grid (box) area, and longer travel time. The coordinative results of the MCDFM during the peak period at 17:45 and 19:00 are categorized as HC and VHC accordingly. This coordinative decision helps the TLCs provide a better mechanism for handling these situations, such as performing dynamic green time allocations based on the needs and traffic coordination for a set of traffic movements that suffer from heavy congestions. This action reduces the number of vehicles at each intersection quicker, while reducing the time required to solve this problem.

This work attempts to propose the MCDFM framework for environments with heterogeneous and real-time data sources. The proposed framework provides a robust, flexible, and responsive coordinative system of multiple data fusion design. The related works, such as Moattari and Majd [34] and He et al. [35], proposed a centralized estimation for global or coordinative decision-making, which manifests the disadvantages of the centralized systems. Fortino et al. [36] proposed a collaborative DF framework of a specific domain of homogeneous body sensor networks. Bienzeisler et al. [37] and Elmas and Sönmez [38] proposed a multi-agent system's collaborative decisions and considered the centralized approach. Comparing with the related works, the MCDFM framework provides coordinative features as part of multiple data fusion modules. The coordinative framework of a multi-agent system provides various merits other than the ability to communicate. The abilities to share resources, perform information exchange, and produce coordinative decisions that contribute to the flexibility and robustness are some other merits that we attempt to fully utilize in the MCDFM framework.

We address a few limitations of the MCDFM framework related to its development and validation as follows: (i) The proposed framework is yet to be implemented in a real-world TLC system and needs more proper testing. (ii) More data, methods, and evaluation metrics need to be considered in the testing scenarios. (iii) The framework needs to be tested in several domains to demonstrate and verify its general applicability.

5 Conclusion

This research aims to establish a Multiple Coordinative Data Fusion Modules (MCDFM) framework for distributed system environment that deals with heterogeneous and real-time data. The general framework

is intended for any domain requiring local and coordinative output to be shared among subsystems to provide a better control mechanism. For discussion and illustration purposes, a TLC system is selected as a conceptual scenario for discussion in this paper. Observation of network of intersections is a good example to present the idea of this proposed framework as each TLC imitates an independent agent's capability. In this scenario, the proposed framework is illustrated with three intersections, S903, S904, and S905, that are controlled by TLCs in a road network. Three intersection scenarios are considered in the discussion, and they are nonpeak period, medium peak period, and peak period. Based on the case study, local output interprets a micro view of the individual intersection, while the coordinative output depicts a macro view of the actual network condition. The case study intends to demonstrate the applicability and usefulness of the framework in solving traffic congestion as a domain problem. The framework successfully identifies three congestion periods of the nonpeak period with a congestion degree of 0.178, a medium peak period with a congestion degree of 0.588, and a peak period with a congestion degree of 0.796. The measurements of the nonpeak period show a variance of 0.061, the medium peak period shows a variance of 0.0593, and the peak period shows a variance of 0.0296. The small values of the variances indicate the consistency of the performance. The coordinative framework is the most important criterion to ensure traffic condition is retrieved at the microlevel to have an accurate picture of the real (macro) problem. The abilities to share resources, perform information exchange, and produce coordinative decisions are some of the important features of the MCDFM. The proposed framework is meant to provide a robust, flexible, and fully responsive DF system implemented with it. In the future, we plan to implement this framework in a real-world TLC system and report the results. We shall also attempt to investigate and study the possibility of adding new algorithms and methods in the preprocessing, filtering, and decision phases of the framework to enhance its performance.

Conflict of interest: The authors have no conflicts of interest to declare. The authors certify that the submission is original work and is not under review at any other publication. All authors have seen and agreed with the contents of the manuscript and there is no financial interest to report.

Data availability statement: The used dataset of this research is available online and has a proper citation within the paper contents.

References

[1]   Lou S, Zhang D, Li J. Study on traffic multi-source data fusion. Int J Cognit Inform Nat Intell. 2019;12(2):63–75.

[2]   Wang C, Zhu Q, Shan Z, Xia Y, Liu Y. Fusing heterogeneous traffic data by kalman filters and gaussian mixture models. IEEE 17th International Conference on Intelligent Transportation Systems (ITSC), Qingdao, China; 2014.

[3]   Liu K, Huang S. Integration of data fusion methodology and degradation modelling process to improve prognostics. IEEE Trans Autom Sci Eng. 2016;13(1):344–54.

[4]   Bris AL, Chehata N, Ouerghemmi W, Wendl C, Postadjian T, Puissant A, et al. Decision fusion of remote-sensing data for land cover classification. Multimodal Scene Underst. 2019;2017:341–82.

[5]   Kim S, Kim H. A new metric of absolute percentage error for intermittent demand forecasts. Int J Forecast. 2016;32(3):669–79.

[6]   Zhu L, Guo F, Polak JW, Krishnan R. Urban link travel time estimation using traffic states-based data fusion. IET Intell Transp Syst. 2018;12(7):651–63.

[7]   Li L, Du B, Wang Y, Qin L, Tan H. Estimation of missing values in heterogeneous traffic data: application of multimodal deep learning model. Knowl Based Syst. 2020;194:105592.

[8]   Tian D, Zhou J, Sheng Z. An adaptive fusion strategy for distributed information estimation over cooperative multi-agent networks. IEEE Trans Inf Theory. 2017;63(5):3076–91.

[9] Lahat D, Adali T, Jutten C. Challenges in multimodal data fusion. EUSIPCO 2014 – 22nd European Signal Processing Conference, Lisbon, Portugal; 2014. p. 101–5.

[10] Jabbar S, Malik KR, Ahmad M, Aldabbas O, Asif M, Khalid S, et al. A methodology of real-time data fusion for localized big data analytics. IEEE Access. 2018;6:24510–20.

[11] Mannini L, Carrese S, Cipriani E, Crisalli U. On the short-term prediction of traffic state: an application on urban freeways in Rome. Transp Res Proc. 2015;10:176–85.

[12] Wang L. Heterogeneous data and big data analytics. Autom Control Inf Sci. 2017;3:8–15.

[13] Lau BPL, Marakkalage SH, Zhou Y, Hassan NU, Yuen C, Zhang M, et al. A survey of data fusion in smart city applications. Inf Fusion. 2019;52(2019):357–74.

[14] El Fouzi N-E, Leung H, Kurian A. Data fusion in intelligent transportation systems: progress and challenges – a survey. Inf Fusion. 2010;12(1):4–10.

[15] Shen G, Han X, Zhou J, Ruan Z, Pan Q. Research on Intelligent analysis and depth fusion of multi-source traffic data. IEEE Access. 2018;6:59329–35.

[16] Kashinath SA, Mostafa SA, Mustapha A, Mahdin H, Lim D, Mahmoud MA, et al. Review of data fusion methods for real-time and multi-sensor traffic flow analysis. IEEE Access. 2021;9:51258–76.

[17] Wahab OA, Otrok H, Mourad A. A cooperative watchdog model based on dempster-shafer for detecting misbehaving vehicles. Comp Commun. 2014;41:43–54.

[18] Shahrbabaki MR, Safavi AA, Papageorgiou M, Papamichail I. A data fusion approach for real-time traffic state estimation in urban signalized links. Transp Res Part C Emerg Technol. 2018;92:525–48.

[19] Akbar A, Kousiouris G, Pervaiz H, Sacho J, Ta-Shma P, Carrez F, et al. Real-time probabilistic data fusion for large-scale iot applications. IEEE Access. 2018;6:10015–27.

[20] Izumi S, Azuma S. Real-time pricing by data fusion on networks. IEEE Trans Ind Inform. 2018;14(3):1175–85.

[21] Saeedmanesh M, Kouvelas A, Geroliminis N. A real-time state estimation approach for multi-region MFD traffic systems based on extended Kalman filter. 2019 TRB Annual Meeting Online (19-02756). Transportation Research Board; 2019.

[22] Xu M, An K, Vu LH, Ye Z, Feng J, Chen E. Optimizing multi-agent based urban signal traffic signal control system. J Intell Transp Syst. 2018;23(4):357–69.

[23] Kumar PM, Manogaran G, Sundarasekar R, Chilamkurti N, Varatharajan R. Ant colony optimization algorithm with internet of vehicles for intelligent traffic control system. comput Netw. 2018;144:154–62.

[24] Rehman A, Rathore MM, Paul A, Saeed F, Ahmad RW. Vehicular traffic optimisation and even distribution using ant colony in smart city environment. IET Intell Transp Syst. 2018;12(7):594–601.

[25] Bui K-HN, Jung JJ. Internet of agents framework for connected vehicles: a case study on distributed traffic control system. J Parallel Distrib Comput. 2018;116:89–95.

[26] Ma C, He R. Green wave traffic control system optimization based on adaptive genetic-artificial fish swarm algorithm. Neural Comput Appl. 2015;31(7):2073–83.

[27] Zhao Y, Ma J, Shen L, Qian Y. Optimizing the junction-tree-based reinforcement learning algorithm for network-wide signal coordination. J Adv Transp. 2020;2020:1–11.

[28] Mostafa SA, Mustapha A, Hazeem AA, Khaleefah SH, Mohammed MA. An agent-based inference engine for efficient and reliable automated car failure diagnosis assistance. IEEE Access. 2018;6:8322–31.

[29] Campbell ME, Ahmed NR. Distributed data fusion: neighbors, rumors, and the art of collective knowledge. IEEE Control Syst. 2016;36(4):83–109.

[30] Mostafa SA, Darman R, Khaleefah SH, Mustapha A, Abdullah N, Hafit H. A general framework for formulating adjustable autonomy of multi-agent systems by fuzzy logic. KES International Symposium on Agent and Multi-Agent Systems: Technologies and Applications. Cham: Springer; 2018 June. p. 23–33.

[31] Bienzeisler L, Lelke T, Wage O, Thiel F, Friedrich B. Devel-opment of an agent-based transport model for the city of hanover using empirical mobility data and data fusion. Transportation Res Procedia. 2019;47:99–106.

[32] Lu B, Wang Q, Wang Y. An improved artificial fish swarm algorithm for traffic signal control. Int J Simul Process Model. 2019;14(6):488–99.

[33] Zhu F, Abdul Aziz HM, Qian X, Ukkusuri SV. A junction-tree based learning algorithm to optimize network wide traffic control: a coordinated multi-agent framework. Transp Res Part C Emerg Technol. 2015;58:487–501.

[34] Moattari M, Majd VJ. Cooperative fusion for distributed estimation of target position in heterogeneous networks with communication delays. Iran J Sci Technol Trans Electr Eng. 2019;43(1):201–10.

[35] He S, Shin H-S, Rsourdos A. Distributed joint probabilistic data association filter with hybrid fusion strategy. IEEE Trans Instrum Meas. 2020;69:1–15.

[36] Fortino G, Galzarano S, Gravina R, Li W. A framework for collaborative computing and multi-sensor data fusion in body sensor networks. Inf Fusion. 2014;22:50–70.

[37] Bienzeisler L, Lelke T, Wage O, Thiel F, Friedrich B. Development of an agent-based transport model for the city of hanover using empirical mobility data and data fusion. Transp Res Proc. 2020;47:99–106.

[38] Elmas C, Sönmez Y. A data fusion framework with novel hybrid algorithm for multi-agent decision support system for forest fire. Expert Syst Appl. 2011;38:9225–36.

[39] Ahmed M, Abdel-Aty M. A data fusion framework for real-time risk assessment on freeways. Transp Res Part C. 2013;26:203–13.

[40] Luo F, Zhang T, Wei W. A coordinative planning framework for cyber-power distribution system. 2016 IEEE Power Energy Society General Meeting (PESGM); 2016

[41] Yan G, Wang H. Autonomous coordinated control strategy for complex process of traffic information physical fusion system based on big data. IEEE Access. 2020;8:148370–77.

[42] Mostafa SA, Gunasekaran SS, Ahmad MS, Ahmad A, Annamalai M, Mustapha A. Defining tasks and actions complexity-levels via their deliberation intensity measures in the layered adjustable autonomy model. 2014 International Conference on Intelligent Environments. Shanghai, China: IEEE; 2014, June. p. 52–5.

[43] Du W, Ding S. A survey on multi-agent deep reinforcement learning: from the perspective of challenges and applications. Artif Intell Rev. 2021;54(5):3215–38.

[44] Sidek O, Quadri SA. A review of data fusion models and systems. Int J Image Data Fusion. 2012;3(1):3–21.

[45] Gravina R, Alinia P, Ghasemzadeh H, Fortino G. Multi-sensor fusion in body sensor networks: state-of-the-art and research challenges. Inf Fusion. 2017;35:68–80.

[46] Zaied ANH, Al Othman W. Development of a fuzzy logic traffic system for isolated signalized intersections in the state of Kuwait. Expert Syst Appl. 2011;38:9434–41.

[47] Mostafa SA, Ahmad MS, Mustapha A, Mohammed MA. A concise overview of software agent research, modeling and development. Softw Eng. 2017;5(1):8–25.