Abstract: The performance of many components in intelligent transportation systems depends heavily on the quality of short-term traffic forecasts. We propose a new method for forecasting traffic based on type-2 fuzzy logic. Type-2 fuzzy logic is powerful in handling uncertainties, including uncertainties in measurements and data used to calibrate the parameters. In our formulation, the value of a membership function corresponding to a particular traffic state is no longer a crisp value. Rather, it is associated with a range of values that can be characterised by a function that reflects the level of uncertainty. Day-to-day traffic information is combined with real-time traffic information to construct fuzzy rules. The performance of the prediction procedure based on type-2 fuzzy logic is encouraging. The mean relative error is in the neighbourhood of 12% for occupancies and 5% for flows. A distinct advantage of a type-2 fuzzy logic-based traffic forecasting approach is that it can produce prediction intervals as a by-product of the fuzzy reduction process. Another desirable property of the proposed model is that the fuzzy engine formulated is usually tractable at every step, making it easy to incorporate site-specific information into the model-building process to obtain more accurate results.

1 Background

Traffic forecasting is often part of the daily routine in traffic operations, useful for implementing algorithms and deploying traffic control strategies. The accuracy level of traffic forecasts is fundamental to the performance of many components in intelligent transportation systems (ITS), such as adaptive ramp metering, transit signal priority control, variable message signs and advanced traveller information systems. The problem of short-term traffic forecasting is to determine the traffic volume or condition in the next time window, usually in the range 5–30 min, based on historical traffic information (day-to-day traffic data) and/or real-time information (traffic data obtained within the current time window or previous time windows).

In the past 20 years or so, many studies have been devoted to this area. Various approaches have been developed to forecast traffic flows, vehicle speeds or other traffic variables, including ones that are based on time series models [1, 2], Kalman filter theory [3], non-parametric methods [4, 5], simulation models [6, 7], local regression models [8–10], and neural network approaches [11–16]. These models usually generate only a single forecast value rather than a range of forecast values in the form of prediction intervals. Prediction intervals are very important, especially in traveller information systems. For such applications, prediction intervals can provide the user with a sense of how reliable the forecast is. They also protect the system provider from the potential liability of failing to provide the ‘right’ forecasts. This article makes use of type-2 fuzzy logic, described in detail in [17], to develop a model for forecasting traffic that would specifically take this feature into consideration.

It is well known that the day-to-day weekday traffic pattern at a given location usually repeats itself on a broad scale. Within a smaller time window such as 15–30 min, however, traffic conditions at a given location may vary from day to day, governed by the traffic conditions upstream and downstream, weather conditions, and other factors. For instance, a spillover of traffic jams from the downstream could reduce the capacity of the road, which in turn will reduce the flow level and induce congestion. Likewise, a capacity reduction upstream can also reduce the flow level which would in turn reduce the congestion level downstream. A distinct advantage of type-2 fuzzy logic is that it is very powerful in handling uncertainties. Our proposed traffic forecasting model based on type-2 fuzzy logic will explicitly consider both the day-to-day recurrent traffic pattern and the real-time fluctuation, which may behave similar to a random walk. By utilising membership functions in type-2 fuzzy logic capable of handling uncertainty, the proposed model can generate traffic forecasts with reasonable accuracy. Moreover, as a by-product of the type-2 fuzzy logic, the model will yield time-dependent prediction intervals for forecasts.

2 Formulation and properties of type-2 fuzzy logic

2.1 Basic concepts and fuzzy set in type-2 fuzzy logic

Compared with type-1 fuzzy logic, type-2 fuzzy logic has different definitions for membership functions. It also
Fuzzy logic membership functions

(a) An example membership function in type-1 fuzzy logic

(b) Deterministic membership function in type-2 fuzzy logic

(c) Random membership function in type-2 fuzzy logic

Fig. 1  Fuzzy logic membership functions

a) An example membership function in type-1 fuzzy logic
b) Deterministic membership function in type-2 fuzzy logic
c) Random membership function in type-2 fuzzy logic

has its own set of operators. With these operators and the extension principle, the properties of type-2 fuzzy logic can be derived from type-1 fuzzy logic. The definition of type-2 fuzzy sets is given by

\[ A = \left\{ (x, U, \mu_A(x, u)) : \forall x \in X, \forall u \in J_x \subseteq [0,1] \right\} \]  

(1)

where \( 0 \leq \mu_A(x, u) \leq 1 \). Different from the definition of a type-1 fuzzy set, which is usually defined as \( A = \{ (x, \mu_A(x)) : \forall x \in X \} \), where \( 0 \leq \mu_A(x) \leq 1 \), a type-2 fuzzy set has an additional dimension, \( u \), associated with the membership value \( \mu_A(x) \). In other words, for type-1 fuzzy logic, when \( x = x' \), its fuzzy set has a crisp membership value, \( \mu_A(x) \), as shown in Fig. 1a. A type-2 fuzzy set, however, has a membership function that would yield multi-valued \( \mu_A(x) \) for \( x = x' \), as shown in Fig. 1b and c. In particular, \( u \) can be viewed as a type-1 fuzzy set, with the membership function \( J_x \) in three-dimensional space. \( J_x \), a vertical slice of \( \mu_A(x, u) \), is called the secondary membership function, denoted by

\[ \mu_A(x = x', u) \equiv \mu_{\tilde{A}}(x') = \int_{u \in J_x} f_c(u)/u \]  

(2)

where \( 0 \leq f_c(u) \leq 1 \) and \( f_c(u) \) is the amplitude of a secondary membership function called a secondary grade. \( \int_{u \in J_x} f_c(u)/u \) in (2) means that the type-2 fuzzy set has a membership \( u \) associated with grade \( f_c(u) \) for \( x = x' \). Note that, as is customary in the fuzzy logic notation, \( \int_{u \in J_x} \) is not an integration operator but a symbol that represents the collection of all points of \( u \) in \( J_x \). \( f_c(u)/u \) is not a division operator; it means that the grade corresponding to the membership value \( u \) is \( f_c(u) \). An interval type-2 fuzzy set is a special case of type-2 fuzzy sets in which the secondary membership functions are defined by \( f_c(u) = 1 \), \( \forall u \in J_x \subseteq [0,1] \). For \( x = x' \), the primary membership value \( u \) can be represented as an interval \([l, r]\).

Since \( X' \subseteq X \), we can then drop the prime notation and refer to \( \mu_A(x) \) as a secondary membership function. The type-2 fuzzy set can be defined as

\[ \tilde{A} = \{ (x, \mu_A(x)) : \forall x \in X \} \]  

(3)

or

\[ \tilde{A} = \cap_{x \in X} \mu_{\tilde{A}}(x)/x = \cap_{x \in X} \left[ \int_{u \in J_x} f_c(u)/u \right]/x \]  

(4)

The domain of a secondary membership function is called the primary membership of \( x \). In (4), \( J_x \) is the primary membership of \( x \), where \( J_x \subseteq [0,1] \) for \( \forall x \in X \).

As in type-1 fuzzy logic, once the fuzzy set is defined, the fuzzy inference can be obtained based on the fuzzy set and the choice of operators for operations on the fuzzy set.

2.2 Operators of type-2 fuzzy logic

There are several set theoretic operations on general type-2 fuzzy sets [17], which are the basis of a type-2 fuzzy logic system.

2.2.1 Union of two type-2 fuzzy sets

The union of two type-2 fuzzy sets \( \mu_{\tilde{A}}(x) \) and \( \mu_{\tilde{B}}(x) \) is expressed by

\[ \mu_{\tilde{A} \cup \tilde{B}}(x) = \int_{u \in J_x} \int_{v \in J_y} f_c(u) * g_c(w)/v \equiv \mu_{\tilde{A}}(x) \prod \mu_{\tilde{B}}(x) \]  

(5)

where \( v = u \lor w \) and \( v \) denotes the maximum \( \lor \)-norm (\( \lor \)-norm is a fuzzy union operation, maximum is the common \( \lor \)-norm operator). The symbol \( \prod \) can denote either minimum or product \( \land \)-norm (\( \land \)-norm is a fuzzy intersection operation; minimum and product are two common \( \land \)-norm operators). The symbol \( \prod \) denotes the join operator. Computationally, for any \( x \), the join operator will enumerate all the possible combinations of \( u \) and \( w \), take the maximum of \( u \) and \( w \) as the resulting primary membership value and take the minimum or product of the two secondary grades, \( f_c(u) \) and \( g_c(w) \), as the resulting secondary grade. This operation will give a new type-2 fuzzy set. In the interval type-2 fuzzy set, the join operator will be simplified as \( \prod_{i=1}^{n} f_{c_i} \), representing the join of \( n \) interval type-1 sets \( F_{c_i} \), having domains \( (l_1, r_1], ..., (l_n, r_n] \), respectively, or \( (l_1 \lor l_2 \lor ... \lor l_n) \lor (r_1 \lor r_2 \lor ... \lor r_n) \).

2.2.2 Intersection of two type-2 fuzzy sets

The intersection of two type-2 fuzzy sets, \( \mu_{\tilde{A}}(x) \) and \( \mu_{\tilde{B}}(x) \),
is expressed by
\[
\mu_{\tilde{A}}(x) = \int_{u \in \mathcal{U}} \int_{w \in \mathcal{W}} f_u(u) \cdot g_w(w) / v \\
\quad \equiv \mu_A(x) \prod_{i=1}^{n} \mu_B(x) \quad x \in X
\]
where \( v \equiv u \wedge w \) and \( \wedge \) denotes the minimum or product function. The symbol \( \prod \) here denotes the meet operator. Computationally, for any \( x \), the meet operator will enumerate all the possible combinations of \( u \) and \( w \), take the minimum or product of \( u \) and \( w \) as the resulting primary membership value and take the minimum or product of the two secondary grades, \( f_u(u) \) and \( g_w(w) \), as the resulting secondary grade. This operation will give a new type-2 fuzzy set. For the interval type-2 fuzzy set, the meet operator will be simplified as \( \prod_{i=1}^{n} F_i \), representing the meet of \( n \) interval type-1 sets \( F_1, \ldots, F_n \) having domains \([l_i, r_i], \ldots, [l_n, r_n]\] respectively, or \([l_1 \times l_2 \times \ldots \times l_n, r_1 \times r_2 \times \ldots \times r_n]\).

**2.2.3 Complement of type-2 fuzzy sets** The complement of a type-2 fuzzy set, \( \mu_\tilde{A}(x) \), is expressed by
\[
\mu_\tilde{A}(x) = \int_{u \in \mathcal{U}} f_u(u) / (1 - u) \equiv -\mu_A(x) \quad x \in X
\]
where \( \neg \) denotes the negation operator. Computationally, for any \( x \in X \), the primary membership value of the complement type-2 fuzzy set will be \( 1 - u \), with the secondary grade \( f_u(u) \).

**3 Type-2 fuzzy logic for short-term traffic forecasting**

In this section, we discuss the type-2 fuzzy logic system in conjunction with our application. A type-2 fuzzy logic system is a rule-based system comprising five components: fuzzifier, fuzzy rules, inference, type-reducer and defuzzifier, as shown in Fig. 2. All the rules have antecedents and consequents. Based on the input and the antecedents of the rules, the fuzzy inference process will compute a ‘firing level’ for each rule, combine the consequents of the rules according to the firing level and then generate the resulting type-2 fuzzy set. The type-reducer and defuzzifier will perform the type-reduction and defuzzification to get a crisp value from the type-2 fuzzy set. This crisp value is the output of the type-2 fuzzy logic system.

The type-2 fuzzy logic system developed for traffic forecasting has the following five assumptions:

1. All the type-2 fuzzy sets are interval type-2 fuzzy sets.
2. Antecedent and consequent membership functions are Gaussian primary membership functions.
3. Input membership functions are Gaussian primary membership functions, with uncertain standard deviation.
4. The fuzzy operations use product implication and \( t \)-norm.
5. The type-reduction uses a centre-of-sets method and the defuzzification process uses a simple average method.

Assumptions 1–3 are made because traffic is a non-stationary process full of noise. It is in general difficult to determine the exact probability density function for such a process. The interval type-2 fuzzy set and Gaussian primary membership functions are quite robust compared with other choices. Assumptions 4 and 5 are made for simplifications in computation.

**3.1 The membership functions in the traffic forecasting model**

The interval type-2 fuzzy set has an upper membership function and a lower membership function. This property can be conveniently utilised to generate a prediction interval. In our formulation, Gaussian primary membership functions are used in two ways. We consider the use of a Gaussian primary membership function with a fixed standard deviation, \( \sigma \), but uncertain mean in the following form:
\[
\mu_A(x) = \exp \left[ -\frac{1}{2} \left( \frac{x - m}{\sigma} \right)^2 \right] \quad m \in [m_1, m_2]
\]
Denote \( \exp \left[ -\frac{1}{2} \left( \frac{x - m}{\sigma} \right)^2 \right] \) in (8) by \( \mathcal{N}(m, \sigma; x) \). For each value of \( m \), there is a corresponding membership curve. The choice of \( m_1 \) and \( m_2 \) is based on the historical information. In the case of the interval type-2 fuzzy set, the upper membership function, \( \pi_A^U(x) \), is defined by
\[
\pi_A^U(x) = \begin{cases} 
\mathcal{N}(m_1, \sigma; x) & x \leq m_1 \\
1 & m_1 \leq x \leq m_2 \\
\mathcal{N}(m_2, \sigma; x) & x > m_2
\end{cases}
\]
whereas the lower membership function, \( \pi_A^L(x) \), is defined by
\[
\pi_A^L(x) = \begin{cases} 
\mathcal{N}(m_2, \sigma; x) & x \leq m_2 \\
\mathcal{N}(m_1, \sigma; x) & m_1 \leq x \leq m_1 + m_2 \\
\mathcal{N}(m_1, \sigma; x) & x \geq m_1 + m_2
\end{cases}
\]
Similarly, we can consider the use of a Gaussian primary membership function with fixed mean, \( m \), but...
uncertain standard deviation:

\[ \mu_l(x) = \exp \left[ -\frac{1}{2} \left( \frac{x - m}{\sigma} \right)^2 \right] \quad \sigma \in [\sigma_1, \sigma_2] \]

(11)

For each value of \( \sigma \), there is a corresponding membership curve. In the case of the interval type-2 fuzzy set, the upper membership function, \( \mathcal{P}_l(x) \), is

\[ \mathcal{P}_l(x) = N(m, \sigma_2; x) \]

(12)

The lower membership function, \( \mathcal{P}_l(x) \), is defined by

\[ \mathcal{P}_l(x) = N(m, \sigma_1; x) \]

(13)

### 3.2 The design of a traffic forecasting model based on type-2 fuzzy logic

There are five steps involved in the design of a type-2 fuzzy logic-based traffic forecasting model:

**Step 1: Design of the fuzzifier** The input data of a fuzzy logic system are a set of crisp values. The function of the fuzzifier is to transform the crisp values into a set of fuzzy values, that is, variables with a fuzzy membership function. In the traffic forecasting model, the fuzzifier will take the traffic variable at the \( k \)-th time interval, \( x_k \), as an input to generate a type-2 fuzzy set. The membership functions used in our model are Gaussian primary membership functions with uncertain standard deviations given by

\[ \mu_k(x_k) = \exp \left[ -\frac{1}{2} \left( \frac{x_k - x'_k}{\sigma} \right)^2 \right] \quad \sigma \in [\sigma_1, \sigma_2] \]

(14)

It is reasonable to make the fuzzifier time dependent since the mean traffic variables at different time periods are very different (e.g. rush-hour traffic versus after-hours traffic). The variance of the traffic variable, however, falls into a range with its boundary values determined from data across different data sets. The resulting membership function is shown in Fig. 3. For simplicity in notation, we omit the subscript to denote days here as well as in the following subsections.

**Step 2: Construction of fuzzy rules** Once the set of type-2 fuzzy membership functions is defined, the next step is to construct the fuzzy rules for processing the fuzzy input. In our case, historical data are used to build the fuzzy rules. It is similar to a training process in which data sets are utilised one by one to establish the centre of the fuzzy sets that appear in the antecedents and consequents of the rules.

The \( l \)-th fuzzy rule in the set with a total of \( M \) rules has the format:

\[ R^l : \text{IF } x_1^l \text{ and } \ldots \text{ and } x_p^l \text{ is } F_{p}^l, \text{ THEN } y \text{ is } G^l \]

where \( F_{p}^l \) is the antecedent, \( G^l \) is the consequent of the \( l \)-th fuzzy rule, \( x_1,\ldots,x_p \) are the input of the fuzzy logic system and \( y \) is the output for this rule, which will be utilised in fuzzy inference. In our model, rules are mostly developed based on historical information. Since the data available to us are quite limited, we use a single data set (data obtained on a particular day) to construct a single fuzzy rule. This does not have to be the case. There are many alternative ways to construct fuzzy rules. For example, with more data available, one can group data into subgroups based on various criteria and construct each fuzzy rule based on data from a subgroup. In our case, the data obtained on the real-time basis are also included to construct an additional rule similar to the ones described above.

**Step 3: Design of the fuzzy inference engine** Fuzzy inference is the key component of the fuzzy logic system. Based on the input and the antecedents of the rules, it calculates a ‘firing level’ for each rule and then applies these firing levels to the consequent fuzzy sets. In type-2 fuzzy logic systems, the output type-2 fuzzy set of the fuzzy inference of the \( l \)-th fuzzy rule, \( \mu_{\mathcal{B}}(y) \), is:

\[ \mu_{\mathcal{B}}(y) = \mu_{\mathcal{G}}(y) \prod \left\{ \left[ \prod_{x_i \in X_i} \mu_{\mathcal{K}}(x_i) \prod \mu_{\mathcal{F}}(x_i) \right] \prod_{x_p \in X_p} \mu_{\mathcal{K}}(x_p) \prod \mu_{\mathcal{F}}(x_p) \right\} \quad y \in Y \]

(15)

where \( \mu_{\mathcal{K}}(x_i) \) is the type-2 membership function of the input, \( \mu_{\mathcal{F}}(x_i) \) is the type-2 membership function of the antecedent \( i \) of the \( l \)-th rule, and \( \mu_{\mathcal{G}}(y) \) is the type-2 membership function of the consequent of the \( l \)-th rule. Equation (15) can be written as

\[ \mu_{\mathcal{B}}(y) = \mu_{\mathcal{G}}(y) \prod F^l(x') \]

(16)

where \( F^l(x') \) is the firing level of the input data.

Since the interval type-2 fuzzy sets are used for forecasting traffic, the firing level will also be an interval set:

\[ F^l(x') = \left[ f^l(x'), f^l(x') \right] \equiv \left[ f^l, f^l \right] \]

(17)

where

\[ f^l(x') = \sup_{x} \int_{x_i \in X_i} \ldots \int_{x_p \in X_p} \mu_{\mathcal{K}}(x_i) \mu_{\mathcal{F}}(x_i) \]

\[ \times \ldots \mu_{\mathcal{K}}(x_p) \mu_{\mathcal{F}}(x_p) / x \]

(18)
\[ \mathcal{F}^f(x') = \sup_x \int_{x \in X_1} \cdots \int_{x \in X_p} [\pi_{x_1}(x_1) \ast \pi_{x_p}(x_1)] \]

\[ \ast \cdots \ast [\pi_{y_1}(x_1) \ast \pi_{y_p}(x_1)]/x \]

The supremum is attained when terms in the brackets attain their supremum. Then,

\[ \mu_B(y) = \prod_{i=1}^{M} \mu_{B_i}(y) \]

which will combine the resulting type-2 sets of all the rules and generate a type-2 fuzzy set.

**Step 4: Type-reduction**

For traffic forecasting, the type-2 fuzzy set generated from the previous steps needs to be converted to a crisp value. This is realised through Steps 4 and 5, type-reduction and defuzzification.

Type-reduction generates the centroid type-1 fuzzy set of a type-2 fuzzy set. There are several other methods for type-reduction, such as centre-of-sums type-reduction, height type-reduction, modified height type-reduction and centre-of-sets type-reduction.

For the sake of computational efficiency, we employ the centre-of-sets type-reduction method. Instead of combining the type-2 sets from the fuzzy inference of all the rules before reduction, the centre-of-sets type-reduction makes use of the centroid method to reduce the resulting type-2 sets from each rule and obtain a type-1 set \([y_i^l, y_i^r]\) for each rule \(i\). The weighted combination of these type-1 sets is then used to get the final type-1 set \([y_0, y_f]\):

\[ y_1 = \frac{\sum_{i=1}^{M} f_i^l \cdot y_i^l}{\sum_{i=1}^{M} f_i^l} \]

and

\[ y_r = \frac{\sum_{i=1}^{M} f_i^r \cdot y_i^r}{\sum_{i=1}^{M} f_i^r} \]

where \(f_i^l, f_i^r\) are the firing level corresponding to \(y_i^l\) and \(y_i^r\) of rule \(i\) that will maximise \(y_r\) and minimise \(y_l\). \(f_i^l, f_i^r\) can be enumerated in the interval \([f_i^l, f_i^r]\).

**Step 5: Defuzzification**

Defuzzification is the last step to get the final forecast result. The defuzzification of a type-2 fuzzy logic system is identical to the defuzzification of a type-1 fuzzy logic system. There are also several methods for the defuzzification of a type-1 or a type-2 fuzzy logic system, such as the centroid defuzzifier, centre-of-sums defuzzifier, height defuzzifier, modified height defuzzifier and centre-of-sets defuzzifier. A commonly used defuzzification method is the centroid method

\[ y_c(x) = \frac{\sum_{i=1}^{N} y_i \cdot \mu_B(y_i)}{\sum_{i=1}^{N} \mu_B(y_i)} \]

in which the range of \(y\) is discretised into \(N\) points. The subscript ‘\(c\)’ denotes the centroid method. In the case of the interval set, we can defuzzify the interval \([y_l, y_r]\) from type-reduction using the average of \(y_l\) and \(y_r\). Hence, the crisp value of the type-2 fuzzy logic system used in our model is simply

\[ y(x) = \frac{y_l + y_r}{2} \]

### 4 Model validation

#### 4.1 Data description

The data used for testing the performance of the proposed model are from the California PATH database. They were collected from a section of a 7-mile long freeway segment on Interstate 880, the Nimitz Freeway, in Alameda County, California, between Alverado Niles Road and S/R 238 for the Freeway Service Patrol program [18]. The specific site where the data were collected is a northbound segment between A Street and Hesperian Blvd. A total of 24 data sets with each data set covering a single day are used. Data for each day cover a time period between 5 a.m. and 10 a.m. and between 2 p.m. and 8 p.m.

Fig. 4a is a plot of flow over time for all 24 data sets. The plot exhibits a clear pattern of recurrent traffic. On each day, the onset of peak traffic, during which flow increases steadily, occurs almost at the same time. Likewise, the end of the peak traffic also takes place at the same time of day. If we compare the flow levels of a given time period across days, the flow levels for the same time period from different days appear to be similar. Fig. 4b is a plot of flow over occupancy for the same 24 data sets. It becomes clear that day-to-day traffic conditions are not similar. On certain days there is heavy congestion, whereas on some other days congestion appears to be fairly mild. Further investigation of the data set (the result is not plotted here) reveals that within a smaller time window traffic patterns could vary substantially from day to day.

#### 4.2 Performance measures

The 24 data sets shown in Fig. 4 are used for model calibration and prediction. We consider two cases. In the first case, we use only historical information for rule construction. In the second case, we use both historical and real-time information for rule construction. For the real-time data, we use only the data from the current time interval. A total of 30 runs are made. In each run, we randomly select a data set for testing the proposed model. In the process of building the fuzzy logic system, we randomly choose 15 of the remaining data sets to construct the fuzzy rules. This will ensure that the data set used for testing will not be used for rule construction. The performance of the model is compared against three commonly used performance measures for prediction models, the mean absolute error (MAE), the mean relative error (MRE) and the mean square error (MSE), defined as:

\[ \text{MAE} = \frac{\sum_{n=1}^{N} \sum_{i=1}^{I} |x^n_i - \hat{x}^n_i|}{NI} \]

\[ \text{MRE} = \frac{\sum_{n=1}^{N} \sum_{i=1}^{I} \frac{|x^n_i - \hat{x}^n_i|}{x^n_i}}{NI} \]

and

\[ \text{MSE} = \frac{\sum_{n=1}^{N} \sum_{i=1}^{I} (\hat{x}^n_i - x^n_i)^2}{NI} \]

where \(x^n_i\) is traffic forecasts made at time \(i - 1\) for the occupancy level at \(i\) in run \(n\). \(N\) is the total number of runs and \(I\) is the total number of data points for prediction in each run. A plot of observed occupancy versus the prediction interval is given in Fig. 5 for data collected at 5-min intervals. The performance measures
Fig. 4  Plots of the traffic data used in the study

(a) Flow vs. time plot for all 24 data sets.

(b) Flow vs. occupancy plot for all 24 data sets.
for prediction of flows and occupancies are listed in Tables 1 and 2, respectively. The MRE is in the neighbourhood of 12% for occupancies and 5% for flows. As indicated in the tables, in all cases, using both historical and real-time information improves the performance of the model. Though the improvement appears to be marginal, it should be noted that the results presented here were obtained under the situation in which parameters in the model were not fine-tuned in the calibration process and, for illustration purposes, the real-time data we use are only those from the current time interval (5 min). The improvement is more pronounced in the performance measures for occupancies than in those for flows in terms of MSEs (a 17% reduction in occupancy estimation versus an 11% reduction in flow estimation), suggesting that flows are more recurrent than occupancies. This is reasonable since changes in occupancies are tied closely with traffic conditions, which often fluctuate widely from day to day on a small time scale.

Table 1: Performance measures for forecasting occupancy

<table>
<thead>
<tr>
<th>Method</th>
<th>Historical information only</th>
<th>Both historical information and real-time information</th>
</tr>
</thead>
<tbody>
<tr>
<td>MRE (%)</td>
<td>12.48</td>
<td>11.97</td>
</tr>
<tr>
<td>MAE</td>
<td>1.67</td>
<td>1.54</td>
</tr>
<tr>
<td>MSE</td>
<td>8.05</td>
<td>6.71</td>
</tr>
</tbody>
</table>

Table 2: Performance measures for forecasting flows

<table>
<thead>
<tr>
<th>Method</th>
<th>Historical information only</th>
<th>Both historical information and real-time information</th>
</tr>
</thead>
<tbody>
<tr>
<td>MRE (%)</td>
<td>5.84</td>
<td>5.61</td>
</tr>
<tr>
<td>MAE</td>
<td>5.97</td>
<td>5.75</td>
</tr>
<tr>
<td>MSE</td>
<td>65.75</td>
<td>58.49</td>
</tr>
</tbody>
</table>

5 Concluding remarks and future research

A traffic forecasting model based on type-2 fuzzy logic is proposed and discussed. Different from type-1 fuzzy logic, a type-2 fuzzy logic model can handle rule uncertainties with three-dimensional membership functions. Compared with existing models, the model based on type-2 fuzzy logic will generate not only traffic forecasts but also prediction intervals without additional effort. Moreover, the process of model formulation is fully tractable, making it easy to incorporate site-specific information into the forecasting model for better prediction results. The model is robust in the sense that there is no fitting process involved when using the historical or real-time data. Consequently, the overfitting problem embedded in many goodness-of-fit-type models will not arise.

Our preliminary result shows that type-2 fuzzy logic is promising in forecasting traffic variables. The performance of the model in terms of the measures of performance used in this study is either comparable to or better than that of other approaches reported in the literature. There is still room to further improve the performance of the proposed model. For example, currently in the model-building process we have made no attempt to tune the parameters to optimise the performance of the model. This is in part because there are only limited data sets available to us. In future studies, we should obtain data from other sites and automate the model calibration process. Currently, simplification has been made in constructing the fuzzy logic rules. We use only the interval fuzzy set for the fuzzy engine and use the simple average in the defuzzification process (24). It is interesting to see that the performance of the model seems to be reasonably good even with the simplification we have made. Further tests are needed to examine whether the same simplification would work equally well for data from other sites. Research is ongoing to test other methods for constructing fuzzy rules and tie the use of a specific method with the characteristics of the traffic conditions at specific sites.

![Fig. 5 The plot of predicted occupancy versus observed occupancy](image-url)
6 References

16 Alessandru, C. and Ishaq, S. ‘Hybrid model-based and memory-based traffic prediction system’, Transportation Research Record No 1879. TRB, National Research Council, 2004 pp. 59–70