Risk Assessment of Railway Transportation Systems using Timed Fault Trees

Zhaoguang Peng[a,b], Yu Lyu[b,*], Alice Miller[b], Chris Johnson[b], Tingdi Zhao[a]

[a]*School of Reliability and Systems Engineering, Beijing University of Aeronautics and Astronautics, Beijing, China*
[b]*School of Computing Science, University of Glasgow, Glasgow, United Kingdom*

Abstract

Safety is an essential requirement for railway transportation. There are many methods that have been developed to predict, prevent and mitigate accidents in this context. All of these methods have their own purpose and limitations. This paper presents a new useful analysis technique: timed fault tree analysis. This method extends traditional fault tree analysis with temporal events and fault characteristics. Timed Fault Trees (TFTs) can determine which faults need to be eliminated urgentlyand it can also provide how much time have been left at least to eliminate the root failure to prevent accidents. They can also be used to determine the time taken for railway maintenance requirements, and thereby improve maintenance efficiency, and reduce risks. In this paper, we present the features and functionality of a railway transportation system, and principles and rules of TFTs. We demonstrate the applicability of our framework by a case study on a simple railway transportation system.

Keywords:
Fault tree analysis, railway transportation systems, risks, risk assessment, timed fault trees

1. Introduction

System safety relies on robust safety design, good management, and efficient maintenance [1]. System safety is an essential requirement of a railway transportation system. Primary risks include derailment, collision and fire

*Correspondence to: Yu Lyu, School of Computing Science, University of Glasgow, Glasgow G12 8RZ, United Kingdom. Email: y.lyu.1@research.gla.ac.uk

to property and personnel [2]. Some of the key safety issues in railway transportation systems are discussed in [3].

Our work has been inspired by the GRAIL (GNSS Introduction in the RAIL sector) project that is under development cooperated with ERTMS European Rail Trail Traffic Management System (ERTMS), ESA (European Space Agency), and EC (European Commission). These projects have proved the feasibility of introducing Global Navigation Satellite System (GNSS) in railways and in particular ERTMS by means of theoretical studies and demonstrations.

The major difference between the GRAIL and current railway systems is that it involves unmanned operation. Trains will be navigated using satellites and driven by computers. The only operation to require human involvement is that of behind-the-scenes coordination and intervention in case of failure. The main problem with GNSS is that of navigation accuracy in terms of position and time. This inaccuracy is caused by signal obstacles (such as culverts, bridges, or buildings) encountered when the train is running. Different operation environments require different standards, and the requirement for the railway is different as well. If the navigation accuracy is not satisfied, there will be problem that is caused due to time factors in the train operation. Once the equipment fails or the accuracy reduces significantly, the train needs enough time to eliminate the fault, to prevent the accident.

There are still many unsolved problems related to GRAIL, for example the Man Equipment Environment problem, and research is still on going. Previous research involves the evaluation of satellite navigation for identification and management of ERTM, human centred junction signalling, and guidance on the use of selective door operation. However, there has been no analysis of time dependent properties. Our models include a notion of time to this context, and our analysis aims to identify key failures that lead to accidents. We aim to provide theoretical support for emergency plans and the design of industrial standards.

In this paper, we present a novel analysis technique: timed fault trees (TFTs). The purpose of TFTs is to analyse the relationship between safety and time in systems that are traditionally modelled using FTA. The questions that we want to address, that are not amenable to analysis using FTA include: if two parts of the system require maintenance, which part should be repaired first? How long can a repair wait, so that a given hazard can be avoided?

This paper is organised as follows. First, we introduce the analysis techniques. Second, we present the system that will be analysed. Third, we propose the model based on TFT, and we describe the analysis process using this technique. Then, We demonstrate the applicability of our technique by a case study on a simple railway transportation system. Finally, we conclude and propose directions for future research.

2. Analysis techniques

2.1. Traditional techniques

In order to render systems as safe as possible, a large number of analysis techniques have been developed, such as hazard and operability study (HAZOP), failure mode and effect analysis (FMEA), fault tree analysis (FTA) [4], functional hazard analysis, and event tree analysis (ETA). FTA is an important logic and probabilistic technique, and is mostly used in system reliability and safety [5].

HAZOP is a structured and systematic examination of a planned or existing process to identify and evaluate risks. The HAZOP technique is mainly used in chemical process systems, and is a qualitative technique that involves applying a set of guidewords (descriptors) to a number of parameters. FMEA is an effective analytical tool used to examine possible failure modes and to eliminate potential failure during system designs [6]. FMEA effectively depends on the members of the committee, and it is limited by their experience of previous failures, but also is unable to discover complex failure modes. ETA is a logical evaluative process that involves tracing forward in time or through a causal chain, whereas FTA is a deductive process. Although ETA allows one to identify the effect of a given event path on a system, it cannot pinpoint the specific event that leads to an accident.

FTA was first developed by H. Watson and A. B. Mearns at Bell Labs, and it was used to improve the reliability of the ICBM minuteman missiles system [3]. Traditional FTA has been applied to various applications. These applications include a number of high hazard industries such as nuclear power [7], the oil industry [8], and traffic [3], as well as applications in mechanical engineering [9, 10, 11]. In general, FTA is useful to analyse and predict system reliability and safety [12].

FTA is a powerful diagnostic technique used to demonstrate the root causes of undesired events using logical and functional relationships among components, processes, and subsystems [13]. A fault tree (FT) is a model which logically and graphically represents the various combinations of possible events, either faulty or normal, that occur in a system and lead to unexpected events or states [14]. FTs can be used to identify the cause of undesired events [5, 15]. Faults can be due to hardware failure, software error, or human error.

Traditional FTA involves events and gates, and employs Boolean algebra. Logic modelling is used to graphically represent relationships among basic events. FTA is usually carried out at two levels: a qualitative level in which a list of all possible combinations of events that lead to an event called the *Top Event* is determined (minimal cut sets). Traditional solution of fault trees involves the determination of the so-called Minimal Cut Sets (MCSs). Cut sets are the unique combinations of component failures that can cause system failure. Specifically, a cut set is said to be a minimal cut set if, when any basic event is removed from the set, the remaining events collectively are no longer a cut set [16]. Thus, a quantitative level in which the probability of the occurrence of the nodes in the tree can be calculated [7, 17].

Several methods have been proposed to improve FTA to solve specific problems. One of these involves the use of Binary Decision Diagrams (BDDs) [18, 19]. In this approach, a failure mode is represented using a Boolean equation, which can be manipulated mathematically. This approach overcomes some disadvantages of traditional FTA by enabling efficient and exact qualitative and quantitative analysis of fault trees [20]. However, the BDD approach does not involve direct analysis of a fault free, but of an alternative representation [21]. This can lead to problems (an error in the BBD representation may be hard to translate to the original context, for example, [21]).

2.2. Time dependent techniques

In [22], a time-dependent methodology for fault tree analysis is proposed. This has been developed to allow one to obtain exact and detailed probabilistic information for any fault tree. The approach involves successive calculation of probabilistic information related to a primary failure, mode failure (critical path), or top failure. The probabilistic information consists of existence probability, failure rate, and failure intensity. Note that, in the time-dependent approach, time is given as a function of information and not as a specific value, and thus cannot be used to label an associated fault tree.

Some other methods have been proposed to enable timed properties to be analysed using fault trees [23, 24, 25]. These include fault trees with temporal formulas, Fault Trees with Time Dependencies (FTTDs), and temporal fault tree. Fault trees with temporal formulas and FTTDs have both been developed from traditional fault trees and aim to allow safety analysis during the design of safety critical systems [23, 26]. Analysis using FTTDs is limited to single cause effects for causal OR gates. Temporal fault trees are used for qualitative analysis of top event faults [24].

There are also some other popular methods that allow one to model time, such as Stochastic Petri Nets (SPNs) and Markov models. The analysis of an SPN model is usually aimed at the computation of aggregated performance indices such as the average number of tokens in a place, the frequency of firing of a transition, and the average delay of a token [27]. Transition delays are assumed to be random variables from a negative exponential distribution. For analysis using TFTs, no such assumption is made.

Markov models consist of a countable set of states with transitions between them. They are useful for determining probabilistic properties such as: what is the probability that the system reaches a given state? There are some problems associated with risk analysis that Markov models cannot address. For example, how long does it take for an accident to happen, after the root cause (event) occurs? We are able to analyse this type of property with TFTs. In addition, Markov models can be large and cumbersome [14], and their generation error prone and tedious in some cases [5].

3. System description

The signal system, which consists of a set of traffic controls and train operation controls, is one of the most important electrical and mechanical systems in the rail transportation system. It is directly related to operation safety, operation efficiency, and service quality. It guarantees the safety of the passengers and trains, ensuring that the transportation is fast, frequent, and organised. Hazards discussed in this paper are mainly due to signal system failures.

China Railways High-speed (CRH) Electric Multiple Unit (EMU) signal system includes Chinese Train Control System (CTCS), Computer Based Interlocking system (CBI), Centralised Traffic Control System (CTC), and Centralised Signalling Monitoring system (CSM). The system composition is depicted in Figure 1, and can explained as follows:

- CTCS is a control system that ensures the safe running of the trains. It includes tree subsystems: Automatic Train Supervision (ATS), Automatic Train Protection (ATP), Automatic Train Operation (ATO).

- CBI is responsible for the safety interlocking relationship of the turnout, signal, and tracks. It receives command instructions from the ATS or operator, and sends out interlocking information to ATP or ATS.

- As the command centre of the railway, CTC is responsible for monitoring train running, tracking trains, adjusting trains' running plan and any temporary speed limit.

- Centralised Signalling Monitoring system (CSM) is responsible for monitoring all the above systems status in the signal system.

CTCS includes ATO, ATP, and ATS. The responsibility of the ATO, ATP, and ATS are described as follows.

- ATP is responsible for the safe distance between trains, over speed protection, and door control, which includes trackside equipment, interlocking equipment, and on-board equipment. Ground-based ATP transmits information to trains, and then the on-board ATP calculates information, and provides control information to make the trains run under the speed limit. The train doors can only be opened if appropriate information is detected by ATP and the required conditions are met.

- ATS supervises train operation. It is in charge of the transition to automatic switching, schedules the trains according to the train running plan and passenger traffic, selects and keeps routes, automatically or manually adjusts the stop and running time, and transfers command from operating control center (OCC) to the train. ATS includes the central computer and display equipment in OCC, control and recording equipment, field equipment (station, depot, and parking), and transmission channels.

- ATO is responsible for automatic adjustment of train speed, traction and braking instructions, and stopping the train within a given accuracy. The ATO equipment includes the controller, receive/transmit antennas, signs coil and so on. ATO is useful for enhancing passenger comfort and reducing the labor intensity of the drivers. Functions of ATO include auto-piloting, automatic speed control, automatic parking, designated parking, and door control.

The relationship between the ATO, ATP, and ATS is shown in Figure 2, and the details are described below. ATP is the heart of the safety of CTCS, and is essential for the security of train operation. ATS is a part of the top management and command center of the CTCS. ATO is responsible for the optimisation of the CTCS. A CTCS system relies on the coordination of the three subsystems.

ATO, which is under the supervision of the ATP, obtains the train's running instruction of ATS from ATP. ATO calculates the running speed according to the route status, determines and executes the control command. After arriving at a station, ATO issues a door open command after the appropriate safety condition has been satisfied (as demonstrated by an ATP check). At the same time, ATO transfers train information to a ground communicator via the Positive Train Identification (PTI) system antenna, which it then sends to ATS. ATS determines a new assignment according to the available train information, and sends it back to ATO through the track circuit. When entering a new track section, ATO will receive new ground information so as to adjust the speed, and flexibly switch to ATO mode.

In order to facilitate the procedure described above, the signal system requires a coordinated set of control systems: ground control, on-board control, field control, and central control. This system is responsible for traffic control, operation adjustment, and automatic pilot. Our new technique, TFTs, is a valuable tool for assessing risks in this context. It will help to determine which faults require urgent attention, and to evaluate the time available to fix a fault before an accident will occur as a consequence of the fault. TFTs can then be used to construct an emergency plan.

4. Models of TFTs

In this section, we present formal notation that is relevant to analysis technique using timed fault trees.

4.1. TFTs representation

TFTs are an extension of traditional FTs that follows the same top down approach but includes two additional time parameters. This allows us to discover *urgent* faults and a safe time window to repair faults. Time parameters have been included in the definitions of events and gates. Events have two time parameters: the *duration time* and the *start time* of the event. The gate has one time parameter, namely *delay time*. The delay time is the time between an input event and a corresponding output. For example, at an AND gate there may be a delay between the receipt of the two inputs and the output of their sum.

4.2. TFTs notation

In this section, we define the syntax of TFTs. In all cases, capital letters refer to events, and a superscript denotes an event in a sequence (e.g., $A^{(n)}$). Lowercase letters denote duration time (of a fault, event, or hazard), and the duration of event A say is denoted a (etc.). Similarly, the duration of $A^{(n)}$ is denoted $a^{(n)}$.

A duration time a is assumed to belong to interval $[a_{min}, a_{max}]$. (Similarly, $a^{(n)} \in [a^{(n)}_{min}, a^{(n)}_{max}]$). The start time of an event is denoted using the associated lower case letter followed by s. So the start time of event A is denoted as, where $as \in [as_{min}, as_{max}]$. (Similarly, $as^{(n)} \in [as^{(n)}_{min}, as^{(n)}_{max}]$).

Note the difference between a and as: they denote the duration and start time of event A respectively. As an example, suppose that A is the event "applying brakes" and it takes between 15 and 50 seconds for the train to stop. In this case $a_{min} = 15$ and $as = 0$.

We use the superscript * to denote the actual time that an event occurs (i.e., a^* or $a^{(n)*}$). This value depends on the events below A (and $A^{(n)}$) in the fault tree. We use the term *actual time* to refer to this value, and assume that $a^* \in [a^*_{min}, a^*_{max}]$, and $a^{(n)*} \in [a^{(n)*}_{min}, a^{(n)*}_{max}]$.

Gates are denoted $G^{(1)}, G^{(2)}, ..., G^{(n)}$, and we say that gate $G^{(i)}$ has index i. If A is an event, and $G^{(i)}$ a gate, r_A and $r_{G^{(i)}}$ denote the transition rates associated with A and $G^{(i)}$ respectively. The average duration of event A (respectively gate $G^{(i)}$) is denoted $\overline{A}$ (and $\overline{G^{(i)}}$).

The time delay between receiving all inputs to a gate, and production of an output is g, where $g \in [g_{min}, g_{max}]$. $Ar(A)$ represents the arrive rate of the event from the MCS. A higher $Ar(A)$ means that MCS spends less time between the occurrence of the basic event to the top event A. $N(t)$ represents the smallest unit of time t.

4.3. Properties of gates

In this section we introduce some properties of gates that are relevant to TFTs. Our definitions in Section 4.3 follow [14], from which further details can be found.

4.3.1. AND gate

Definition 1. ***(AND gate)***. *The output A occurs only if all of the inputs $B^{(1)}, B^{(2)}, .., B^{(n)}$ occur. This is depicted in Figure 3.*

In order to express the rules relating to a hazard more simply, we first consider the case where there are only two inputs (see Figure 4). Suppose that event A is the hazard in this case. If the delay of the gate G can be ignored, the minimum value of as is $max(b^{(1)*}_{min}, b^{(2)*}_{min})$; the maximum value of as is $max(b^{(1)*}_{max}, b^{(2)*}_{max})$. The derivation of the AND rule is shown as follows,

- Arrival rate: we assume that the duration time of event and the delay time of a gate are random variables selected from a uniform distribution. It follows that the average duration of event $B(n)$ is:

$$N(\overline{B^{(i)}}) = \frac{N(b^{(i)}_{min}) + N(b^{(i)}_{max})}{2} \tag{1}$$

The average delay time of gate $G(i)$ is:

$$N(\overline{G^{(n)}}) = \frac{N(g^{(n)}_{min}) + N(g^{(n)}_{max})}{2} \tag{2}$$

The transition rate rates are:

$$r_{B^{(i)}} = \frac{1}{N(\overline{B^{(i)}})}, r_{G^{(n)}} = \frac{1}{N(\overline{G^{(n)}})} \tag{3}$$

The arrival rate of event A is:

$$Ar(A) = r_{G^{(n)}} * max(r_{B^{(1)}}, r_{B^{(2)}}, ..., r_{B^{(n)}}) \tag{4}$$

- Actual time: by the definition of the AND gate we can calculate the values of a^*_{min} and a^*_{max}, using the minimum and maximum actual values of $B^{(1)}$ and $B^{(2)}$ as illustrated in Figure 5. It can be shown that:

$$a^*_{min} = g_{min} + a_{min} + max(b^{(1)}_{S_{min}} + b^{(1)}_{min}, b^{(2)}_{S_{min}} + b^{(2)}_{min}) \tag{5}$$

$$a^*_{max} = g_{max} + a_{max} + max(b^{(1)}_{S_{max}} + b^{(1)}_{max}, b^{(2)}_{S_{max}} + b^{(2)}_{max}) \tag{6}$$

Extending this result to the case of n inputs (as in Figure 3), we get:

$$a^*_{min} = g_{min} + a_{min} + max(b^{(1)}_{S_{min}} + b^{(1)}_{min}, b^{(2)}_{S_{min}} + b^{(2)}_{min}, ..., b^{(n)}_{S_{min}} + b^{(n)}_{min}) \tag{7}$$

$$a^*_{max} = g_{max} + a_{max} + max(b^{(1)}_{S_{max}} + b^{(1)}_{max}, b^{(2)}_{S_{max}} + b^{(2)}_{max}, ..., b^{(n)}_{S_{max}} + b^{(n)}_{max}) \tag{8}$$

4.3.2. OR gate and XOR gate

Definition 2. ***(OR gate)***. *The output A occurs only if at least one of the inputs $B^{(1)}, B^{(2)}, .., B^{(n)}$ occurs. This is depicted in Figure 6.*

Definition 3. ***(XOR gate)***. *The output A occurs if either of the inputs $B^{(1)}$ and $B^{(2)}$ occurs, but not the both. This is depicted in Figure 8.*

As before, in order to express the rules more simply, we initially restrict ourselves to the two input case (see Figure 7). Suppose that event A is the hazard in this case, and $B^{(1)}$ and $B^{(1)}$ are the inputs of the OR gate. The derivation of the OR and XOR rules is shown as follows,

- Arrival rate: this is as for the AND gate, we assume that the duration time of an event and delay time of a gate are random variables selected from a uniform distribution. The average duration of event $B^{(n)}$ and delay of gate $G^{(i)}$ are the same as those in Equations (1) and (2) respectively. The transition rates are also the same as those in Equation (3). Any of the $B^{(i)}$ can cause the event A in the OR gate or XOR gate. In Figure 6, there are n inputs, and each input has a corresponding arrival rate. Thus, the arrival rate of event A corresponding to each input $B^{(i)}$ is:

$$Ar(A) = r_{G^{(n)}} * r_{B^{(i)}} \tag{9}$$

- Actual time: by the definition of the OR gate we can calculate the values of a^*_{min} and a^*_{max}, using the minimum and maximum actual values of $B^{(1)}$ and $B^{(2)}$ as illustrated in Figure 7. It can be shown that:

$$a^*_{min} = g_{min} + a_{min} + min(b^{(1)}_{S_{min}} + b^{(1)}_{min}, b^{(2)}_{S_{min}} + b^{(2)}_{min}) \tag{10}$$

$$a^*_{max} = g_{max} + a_{max} + max(b^{(1)}_{S_{max}} + b^{(1)}_{max}, b^{(2)}_{S_{max}} + b^{(2)}_{max}) \tag{11}$$

Extending this result to the case of n inputs (as in Figure 6), we get:

$$a^*_{min} = g_{min} + a_{min} + min(b^{(1)}_{S_{min}} + b^{(1)}_{min}, b^{(2)}_{S_{min}} + b^{(2)}_{min}, ..., b^{(n)}_{S_{min}} + b^{(n)}_{min}) \tag{12}$$

$$a^*_{max} = g_{max} + a_{max} + max(b^{(1)}_{S_{max}} + b^{(1)}_{max}, b^{(2)}_{S_{max}} + b^{(2)}_{max}, ..., b^{(n)}_{S_{max}} + b^{(n)}_{max}) \tag{13}$$

When $n = 2$, the OR result obtained above is the same as for XOR result (see Equations (10) and (11)).

4.3.3. Voting gate

Definition 4. ***(Voting gate)****. The output A occurs when at least r inputs occur. The details are depicted in Figure 9.*

- Arrival rate: as for the AND gate, we assume that the duration time of event and delay time of a gate are random variables selected from a uniform distribution. The average duration of event $B^{(n)}$ and gate $G^{(i)}$ are the same as those in Equations (1) and (2) respectively. The transition rates are also the same as those in Equation (3). In Figure 6, there are n inputs, and each input corresponds to an arrival rate. The arrival rate of event A is the same as the AND gate (see Equation (4)).

- Actual time: since we have $B^{(1)}_{min} \leq B^{(2)}_{min} \leq ... \leq B^{(r)}_{min} \leq ... \leq B^{(n-1)}_{min} \leq B^{(n)}_{min}$, we get:

$$a^*_{min} = g_{min} + a_{min} + B^{(r)}_{min} \tag{14}$$

$$a^*_{max} = g_{max} + a_{max} + max(b^{(1)}_{max}, b^{(2)}_{max}, ..., b^{(n)}_{max}) \tag{15}$$

4.4. Analysis process

In this section we outline the analysis approach using TFTs.

1. Complete the fault tree, and find the MCS. This step is similar to that for traditional FTA.
2. Assign each event and gate a minimum and maximum duration and delay between inputs and outputs respectively.
3. Set the initial start time of the basic fault to 0.
4. From the bottom up, according to the rules of the TFT model, incrementally calculate the minimum and maximum actual time of each event.
5. Calculate the actual time of the hazard.
6. Analyse the chronological relationship between the hazard and the MCS, and calculate the urgent basic fault whose actual time is nearest to the hazard time.
7. Calculate the arrival rate of the hazard. Each MCS corresponds to an arrival rate of the hazard. Calculate each arrival rate, and sort arrival rates in ascending order. The arrival rate reflects the average risk of the basic fault. Thus, we get the average urgent basic fault.

8. Propose a solution to the hazard and use the TFT to prove that the hazard will not occur in the future.

In this paper, the time of an event and the gate is expressed via two basic parameters: minimal time and maximal time. These values can be obtained from field statistics, experiments, and from values obtained using similar equipment. How much time we need to stop a train in the emergency situation? When we carry out the TFT method, we only need to know the minimal and the maximal time of events and gates.

It is, of course, of greater value to use TFT analysis to prevent accidents, and to produce emergency plans for hypothetical situations so that we are prepared for future disasters. However, we can only demonstrate the effectiveness of our approach by applying it to accidents that have occurred in the past. In the next section, we demonstrate our method by using it to analyse a railway transportation accident from 2011.

5. Risk assessment

5.1. Case study

Nowadays, rail safety is an extremely important issue in China. In 2011, two high-speed trains travelling on the Yongwen railway line collided on a viaduct in the suburbs of Wenzhou City, Zhejiang Province [28]. In Figure 10, we show that a 16-car CRH1B EMU working train D3115 between Hangzhou and Fuzhou had apparently been brought to a stand by a lightning strike. As it was moving off around 20 minutes later, it was hit from the rear by Beijing-Fuzhou train D301, operated by a 16-car CRH2E. Six cars were derailed, of which four fell off the 20 meters high viaduct. 40 people were killed, at least 192 were injured.

In Figure 11, a TFT model is shown that represents a simple CRH EMU signal system in a rear-end collision as we introduced in Section 3. In a normal state, a train is no closer than a specified safe distance from another train. If a train detects that another train ahead has come to within that safe distance from it, its ATP will send a brake signal to the ATO, and the ATO will brake the train. When the accident occurred in 2011, a train failed to detect that another train had come within the safe distance and so the brake signal was not activated, resulting in a rear-end collision.

We can obtain the time values associated with the gate and events in Figure 11 from known values for similar equipment. For example, the maximum velocity of a train can reach to 350 km/h in China, and the required safe distance is 6-8 km. Therefore, if the brake systems fail, there can be a crash in 64.7 seconds. For this reason, the minimum time value of $G^{(1)}$ is assigned to be 64.7 seconds. In China, it is required that a change on an equipment state should become visible in the ATS central display within 1 second. Similarly, a command should be issued to the controlled system within 1 second.

The times corresponding to each gate and event are shown in Tables 1 and 2. In Table 3, we give the actual time for each event, obtained through the application of TFTs. From Table 3, we can calculate the minimum time between fault and hazard, which could help set the standard for maintenance. We can get the MCS: $< B^{(4)}, B^{(7)} >, < B^{(4)},^{B}(6) >, < B^{(5)}, B^{(6)} >$ $, < B^{(5)}, B^{(7)} >, < B^{(9)}, B^{(10)} >$. The time relationship of MCS and the hazard is shown in Figure 12. The maximum actual time of $B^{(4)}$ is the closest to the time of the hazard. As a result, $B^{(4)}$ should be prevented with the greatest urgency.

Next, the transition rate is calculated based on the rules of TFTs. For example, as the smallest time unit is 0.1 seconds in this case, the duration time of $B^{(1)}$ is from 0.1 to 1 seconds. So, $N(b)^{i}_{min}$ is 1, and $N(b)^{i}_{max}$ is 10. According to Equations (1) and (3) (see Section 4.3), the average duration of event $B^{(1)}$ $(N(\overline{B^{(1)}}))$ is 5. The transition rate of $B^{(1)}$ $(r_{B^{(1)}})$ is 0.2.

The durations and rates of the events and gates are shown in Table 4. According to Equation (4) (see Section 4.3) and the rules of TFTs, the arrival rate of A corresponding to each MCS is as shown in Table 5. As we can see from Table 5, the MCSs with minimum arrival rates are $< B^{(4)}, B^{(6)} >$ and $< B^{(4)}, B^{(7)} >$.

To clarify the results shown in Table 5, we illustrate by calculating MCS $< B^{(4)}, B^{(7)} >$. The actual times of event $B^{(4)}$ and $B^{(7)}$ are in the intervals $[0.5, 10]$ and $[0.1, 5]$ respectively. Events $B^{(4)}$ and $B^{(7)}$ refer to "erroneous acquisition of points positions" and "wrong localisation initialisation" respectively.

The hazard (A) will occur if both events $B^{(4)}$ and $B^{(7)}$ occur. As we can see from Figure 11, if $B^{(4)}$ occurs, the hazard will take place after 52 seconds, and if $B^{(7)}$ occurs, the hazard will take place after 57 seconds. Therefore, it follows that the absolute safe times of maintenance of $B^{(4)}$ and $B^{(7)}$ are 52 seconds and 57 seconds respectively.

Suppose that event $B^{(7)}$ has occurred. If event $B^{(4)}$ subsequently occurs then the train will be in danger until the fault is eliminated. Hence, if fault

$B^{(4)}$ is not eliminated in 52 seconds, some other effective measure will need to be taken to prevent an accident.

5.2. Applicability of the approach

According to the analysis above, in order to avoid a similar hazard, the train should have a detection device to detect a "wrong localisation initialisation" event (e.g., $B^{(4)}$). Once the fault is detected, an emergency preparatory scheme or program of prevention should be immediately launched. At the same time, the "wrong localisation initialisation" fault should be checked, and the localisation initialisation of the system should be updated in order to avoid the hazard.

In this case, $B^{(4)}$ is the basic failure, which is more urgent to be corrected. When there is more than one failure, this analysis technique can provide answers to questions such as "which failure should be eliminated first?" and "which should be eliminated next?" Thus, analysis using TFTs can improve railway maintenance. In this case, the minimal time between the basic failure $B^{(4)}$ to the accident is 52 seconds. By acquiring this vital time, we can calculate how much time we have to take measures to prevent the accident. This information allows us to set maintenance standards.

Our analysis through the case study has demonstrated the applicability and benefits of our analysis technique in allowing us to calculate the minimal time between a fault and an accident. This information is crucial in maintaining railway safety.

5.3. FTs, DFTs, and TFTs

Whereas traditional fault tree analysis uses a top down decomposition method to break down an accident by logical analysis in order to identify the MCS, TFT analysis is used to determine time aspects of critical failures. As TFTs are similar to traditional FTs, some aspects of the two approaches are similar. However, whereas traditional FTs use the probability of individual events to calculate the probability of a top events, with TFTs we use the time aspects of events and gates to calculate timed properties of the system. More importantly, TFTs can be applied to a system at design time.

Dynamic fault trees (DFTs) are an extension of the traditional FTA technique that combine FTA with Markov analysis for sequence-dependent problems. Traditional FTA is based on static fault logic and static failure modes, while DFT analysis is a modelling method based on dynamic logical relationships. A DFT has two special gates (the functional dependency gate and the spares gate [29]), which have been added specifically to analyse computer based systems [14]. However, although DFTs allow one to model faults, it is necessary to translate them to another formalism (such as a Markov, or Bayesian net model) for analysis. In addition, DFTs do not include any notation of time.

As discussed above, traditional FTs and DFTs do not include any way of modelling time. In system design, time aspects are critical. For example, it is important to measure risk tolerant time, failure time, and fault delay time. In FTA, although it is possible to determine logical relationships between events, one cannot view their chronological relationship.

6. Conclusions and future work

In this paper, we present a novel accident analysis technique for analysing railway transportation safety. Time aspects are critical for assessing and preventing railway risks and thus maintaining safety of a railway system or any other complex high-speed safety-critical system. Errors in time calculations can lead to serious accidents. Practically, this technique will provide railway risk analysts, managers, and engineers with a methodology and a tool to improve their safety management and to set maintenance standards.

Timed fault tree analysis is an extension of traditional fault tree analysis, and there are strong similarities between the approaches. The major difference is that in TFTs, time parameters have been included for both events and gates. In addition to the usual cause and effect analysis that is offered by traditional FTs, TFTs allow us to model and reason about the time between events. Like traditional FTs, TFTs enable the generation of MCSs, but they also allow us to identify the most urgent fault. Analysis using TFTs therefore complements traditional fault tree analysis.

In this paper, we introduce the signal system of the CRH and then provide the rules of TFTs and the corresponding analysis process. Then, we demonstrate the applicability of our framework by way of a case study on a simple railway transportation system. We illustrate the use of TFTs by determining the time between a fault and a potential accident, and thus how much time there is to eliminate the fault and prevent an accident.

In future work, we aim to improve some aspects of our technique. For example, TFTs can solve the problems “which” (which root cause is most urgent to be eliminated) and “when” (how long before the root cause must be eliminated). However, it cannot solve the problem “how“ (how the root cause can be eliminated). Moreover, TFTs rely on the time values associated with the events and gates, which are sometimes hard to obtain. How to deal with this issue using TFTs will be the focus of future work.

References

[1] F. D. Carlo, O. Borgia, M. Tucci, Risk-based inspections enhanced with bayesian networks, Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability 225 (3) (2011) 375–386. doi:10.1177/1748006XJRR368.

[2] M. An, S. Huang, C. J. Baker, Railway risk assessment - the fuzzy reasoning approach and fuzzy analytic hierarchy process approaches: A case study of shunting at waterloo depot, Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit 221 (3) (2007) 365–383. doi:10.1243/09544097JRRT106.

[3] E. Jafarian, M. A. Rezvani, Application of fuzzy fault tree analysis for evaluation of railway safety risks: an evaluation of root causes for passenger train derailment, Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit 226 (1) (2012) 14–25. doi:10.1177/0954409711403678.

[4] F. I. Khana, T. Husaina, Risk assessment and safety evaluation using probabilistic fault tree analysis, Human and Ecological Risk Assessment: An International Journal 7 (7) (2001) 1909–1927. doi:10.1080/20018091095483.

[5] W. Vesely, J. Dugan, J. Fragola, J. Minarick, J. Railsback, Fault Tree Handbook with Aerospace Applications, NASA Office of Safety and Mission Assurance, 2002.

[6] N. Xiao, H.-Z. Huang, Y. Li, L. He, T. Jin, Multiple failure modes analysis and weighted risk priority number evaluation in fmea, Engineering Failure Analysis 18 (4) (2011) 1162–1170. doi:10.1016/j.engfailanal.2011.02.004.

[7] A. Volkanovski, M. Čepin, B. Mavko, Application of the fault tree analysis for assessment of power system reliability, Reliability Engineering & System Safety 94 (6) (2009) 1116–1127. doi:10.1016/j.ress.2009.01.004.

[8] Y.-F. Wang, M. Xie, M. S. Habibullah, K.-M. Ng, Quantitative risk assessment through hybrid causal logic approach, Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability 225 (3) (2011) 323–332. doi:10.1177/1748006X10397370.

[9] L. Meshkat, J. B. Dugan, J. D. Andrews, Dependability analysis of systems with on-demand and active failure modes, using dynamic fault trees, IEEE Transactions on Reliability 51 (2) (2002) 240–251. doi:10.1109/TR.2002.1011531.

[10] Y. Dutuit, F. Innal, A. Rauzy, J.-P. Signoret, Probabilistic assessments in relationship with safety integrity levels by using fault trees, Reliability Engineering & System Safety 93 (12) (2008) 1867–1876. doi:10.1016/j.ress.2008.03.024.

[11] B. S. Mastryukov, O. A. Fomicheva, Using the fault-tree method to analyze possible causes of accidents in the casting-rolling complex at omk-steel in vyksa, Metallurgist 54 (9-10) (2011) 561–565. doi:10.1007/s11015-011-9339-0.

[12] H.-Z. Huang, X. Tong, M. J. Zuo, Posbist fault tree analysis of coherent systems, Reliability Engineering & System Safety 84 (2) (2004) 141–148. doi:10.1016/j.ress.2003.11.002.

[13] V. R. Renjith, G. Madhu, V. L. G. Nayagam, A. B. Bhasic, Two-dimensional fuzzy fault tree analysis for chlorine release from a chlor-alkali industry using expert elicitation, Journal of Hazardous Materials 183 (1-3) (2010) 103–110. doi:10.1016/j.jhazmat.2010.06.116.

[14] C. A. Ericson, Hazard Analysis Techniques for System Safety, Wiley, 2005.

[15] Y.-T. Tsai, Applying a case-based reasoning method for fault diagnosis during maintenance, Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 223 (10) (2009) 2431–2441. doi:10.1243/09544062JMES1588.

[16] D. Kececioglu, Reliability Engineering Handbook: Volume 2, DEStech Publications, 2002.

[17] G. Merle, J.-M. Roussel, J.-J. Lesage, A. Bobbio, Probabilistic algebraic analysis of fault trees with priority dynamic gates and repeated events, IEEE Transactions on Reliability 59 (1) (2010) 250–261. doi:10.1109/TR.2009.2035793.

[18] Y. Mo, F. Zhong, H. Liu, Q. Yang, G. Cui, Efficient ordering heuristics in binary decision diagram-based fault tree analysis, Quality and Reliability Engineering International 29 (3) (2013) 307–315. doi:10.1002/qre.1382.

[19] H.-Z. Huang, H. Zhang, Y. Li, A new ordering method of basic events in fault tree analysis, Quality and Reliability Engineering International 28 (3) (2012) 297–305. doi:10.1002/qre.1245.

[20] R. Remenyte-Prescott, J. D. Andrews, An efficient real-time method of analysis for non-coherent fault trees, Quality and Reliability Engineering International 25 (2) (2009) 129–150. doi:10.1002/qre.955.

[21] L. M. Bartlett, S. Du, New progressive variable ordering for binary decision diagram analysis of fault trees, Quality and Reliability Engineering International 21 (4) (2005) 413–425. doi:10.1002/qre.674.

[22] W. E. Vesely, A time-dependent methodology for fault tree evaluation, Nuclear Engineering and Design 13 (2) (1970) 337–360. doi:10.1016/0029-5493(70)90167-6.

[23] J. Magott, P. Skrobanek, A method of analysis of fault trees with time dependencies, in: Proceedings of the 19th International Conference on Computer Safety, Reliability and Security (SAFECOMP 2010), Vol. 1943 of LNCS, Springer, 2010, pp. 176–186. doi:10.1007/3-540-40891-6_16.

[24] G. K. Palshikar, Temporal fault trees, Information and Software Technology 44 (3) (2002) 137–150. doi:10.1016/S0950-5849(01)00223-3.

[25] J. Magott, P. Skrobanek, Method of time petri net analysis for analysis of fault trees with time dependencies, IEE Proceedings - Computers and Digital Techniques 149 (6) (2002) 257–271. doi:10.1049/ip-cdt:20020804.

[26] J. Magott, P. Skrobanek, Timing analysis of safety properties using fault trees with time dependencies and timed state-charts, Reliability Engineering & System Safety 97 (1) (2012) 14–26. doi:10.1016/j.ress.2011.09.004.

[27] P. J. Haas, Stochastic Petri Nets: Modelling, Stability, Simulation, Springer, 2002.

[28] C. M. A. Jackson, What happened at Wenzhou?, Railway Gazette International 167 (9) (2011) 25.

[29] J. B. Dugan, S. J. Bavuso, M. A. Boyd, Dynamic fault-tree models for fault-tolerant computer systems, IEEE Transactions on Reliability 41 (3) (1992) 363–377. doi:10.1109/24.159800.

Tables

Gate	Duration time (s)
$G^{(1)}$	[61.7, 90]
$G^{(2)}$	[0.1, 2]
$G^{(3)}$	[1, 4]
$G^{(4)}$	[0.5, 2]
$G^{(5)}$	[0.1, 1]

Table 1: Duration time of the gates.

Event	Duration time (s)
A	[0, 0.1]
$B^{(1)}$	[0.1, 1]
$B^{(2)}$	[1, 7]
$B^{(3)}$	[0.1, 3]
$B^{(4)}$	[0.5, 10]
$B^{(5)}$	[1, 5]
$B^{(6)}$	[0.1, 1]
$B^{(7)}$	[0.1, 5]
$B^{(8)}$	[0.1, 5]
$B^{(9)}$	[0.1, 1]
$B^{(10)}$	[0.1, 1]

Table 2: Duration time of the events.

Event	**Actual time (s)**
A	[62, 114]
$B^{(1)}$	[0.9, 24]
$B^{(2)}$	[2.5, 21]
$B^{(3)}$	[0.7, 10]
$B^{(4)}$	[0.5, 10]
$B^{(5)}$	[1, 5]
$B^{(6)}$	[0.1, 1]
$B^{(7)}$	[0.1, 5]
$B^{(8)}$	[0.3, 7]
$B^{(9)}$	[0.1, 1]
$B^{(10)}$	[0.1, 1]

Table 3: Actual time of the events.

Event and Gate	**Duration time (s)**	**Transition rate**
A	76	0.013
$B^{(1)}$	5	0.2
$B^{(2)}$	40	0.025
$B^{(3)}$	15	0.667
$B^{(4)}$	52	0.019
$B^{(5)}$	30	0.033
$B^{(6)}$	5	0.2
$B^{(7)}$	25	0.04
$B^{(8)}$	25	0.04
$B^{(9)}$	5	0.2
$B^{(10)}$	5	0.2
$G^{(1)}$	76	0.013
$G^{(2)}$	10	0.1
$G^{(3)}$	25	0.04
$G^{(4)}$	13	0.077
$G^{(5)}$	25	0.04

Table 4: Durations and transition rates of events and gates.

Minimal Cut Sets	**Arrival rate of A**
$B^{(7)}, B^{(4)}$	$1/(2 * 10^8)$
$B^{(2)}$	$1/(2 * 10^8)$
$B^{(3)}$	$1/(1.1 * 10^8)$
$B^{(4)}$	$1/(1.1 * 10^8)$
$B^{(5)}$	$1/(4.8 * 10^8)$
$B^{(6)}$	$1/(4.8 * 10^8)$

Table 5: The arrival rate of each MCS.

Figures

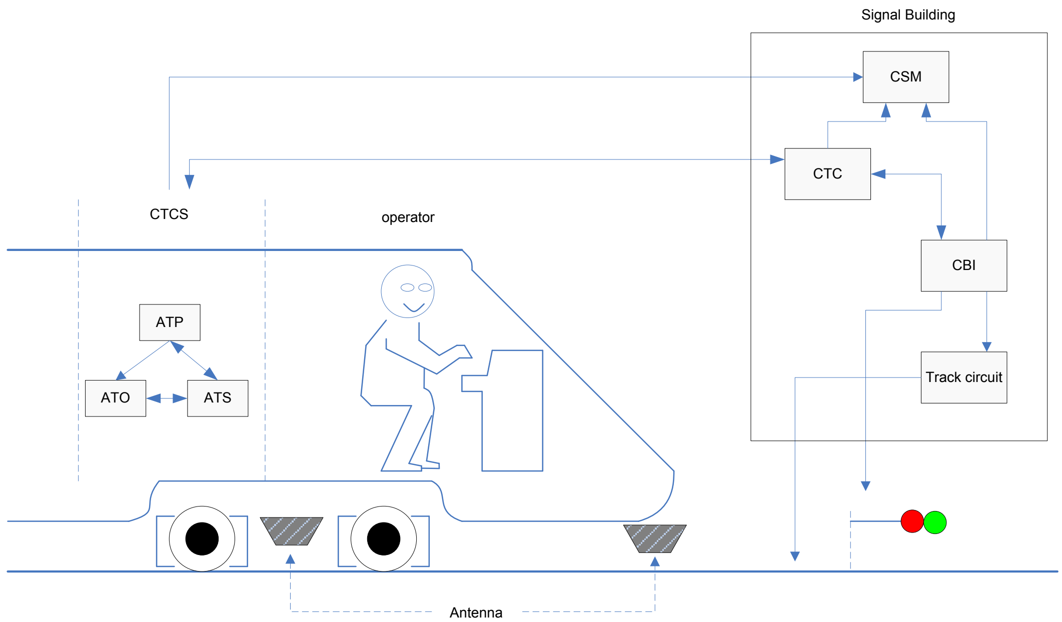

Figure 1: System composition.

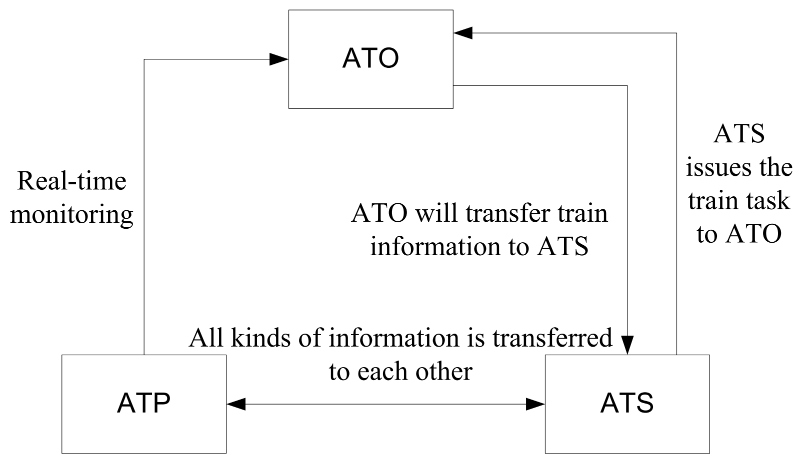

Figure 2: Chinese Train Control System.

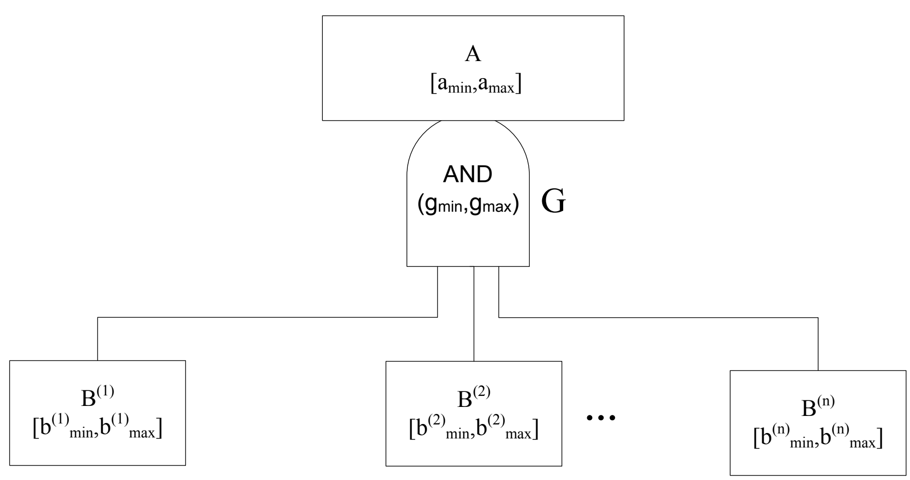

Figure 3: AND gate.

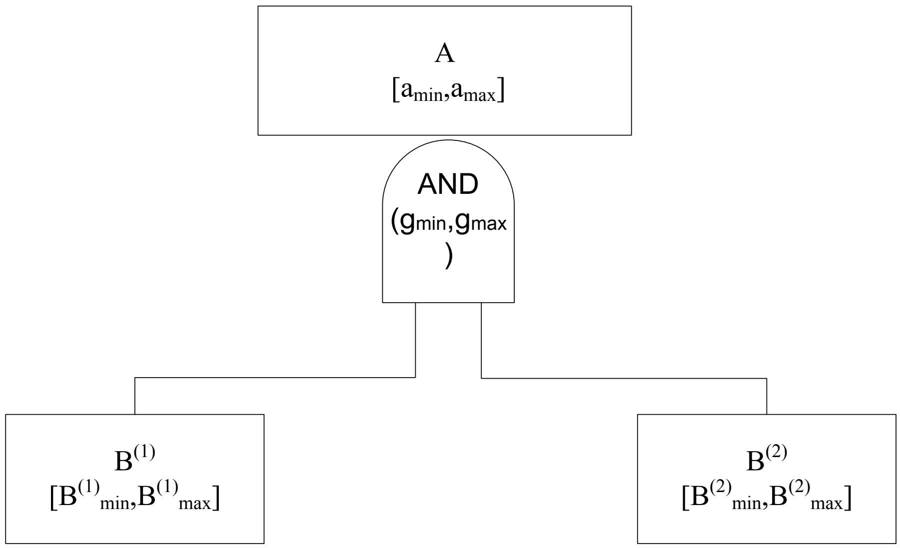

Figure 4: AND gate ($B^{(1)}$, $B^{(2)}$ inputs).

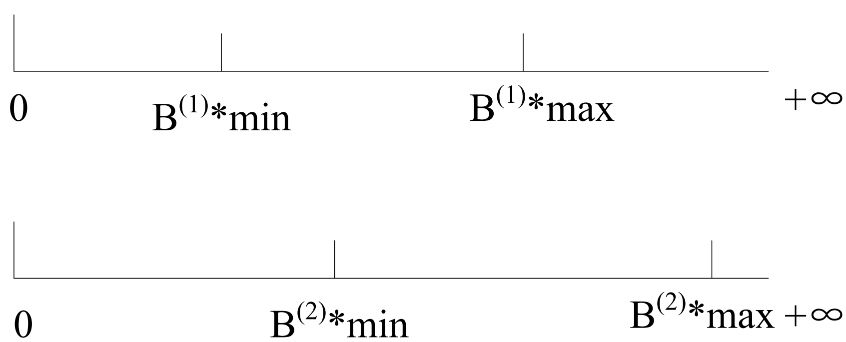

Figure 5: Actual time of $B^{(1)}$ and $B^{(2)}$.

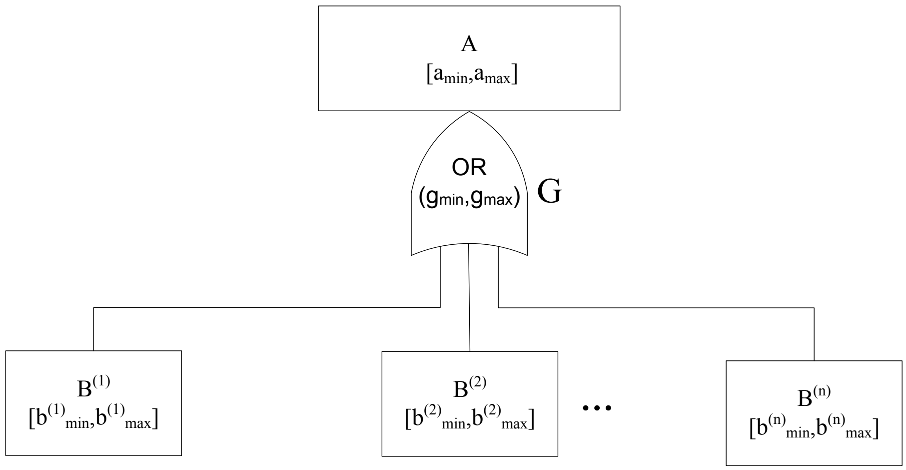

Figure 6: OR gate.

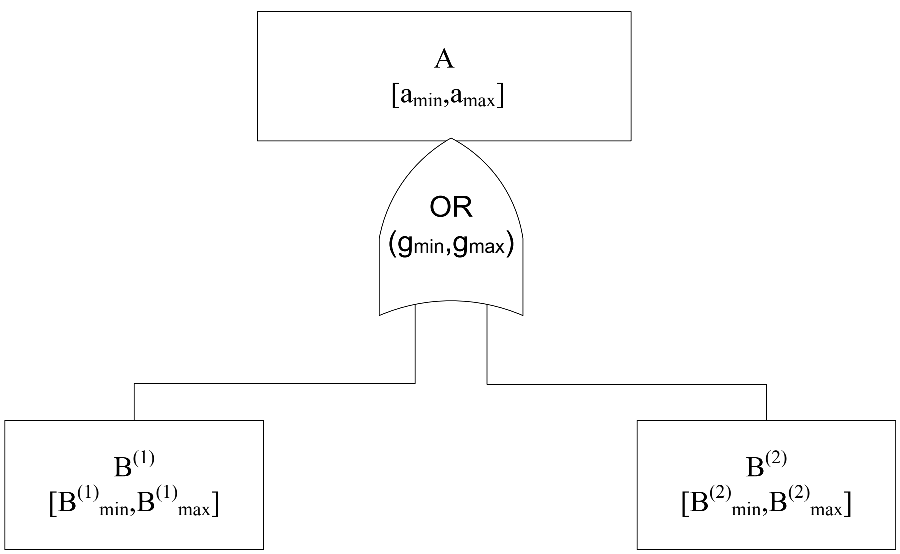

Figure 7: OR gate ($B^{(1)}$, $B^{(2)}$ inputs).

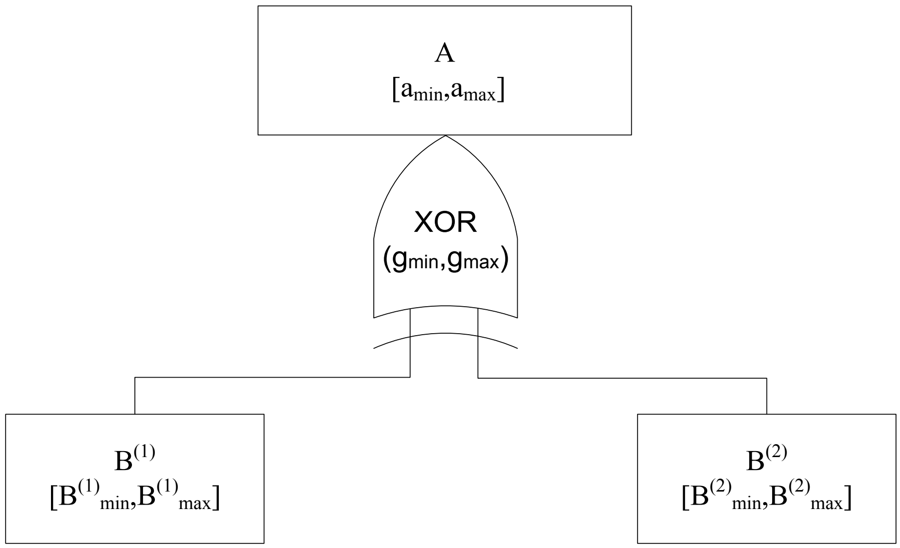

Figure 8: XOR gate.

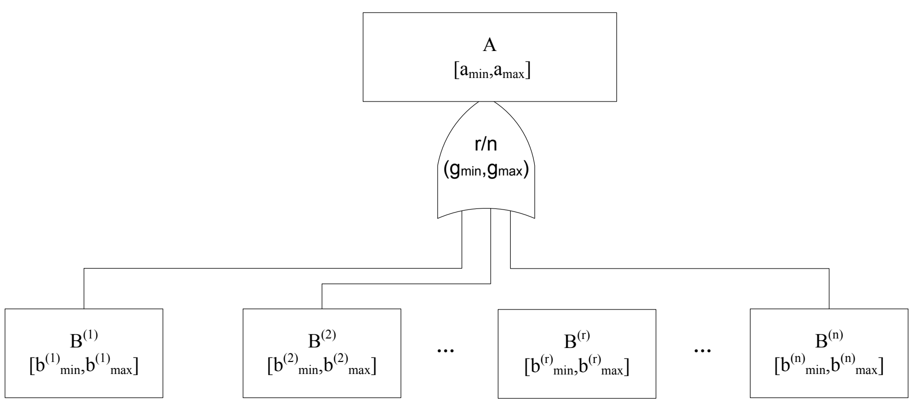

Figure 9: Voting gate.

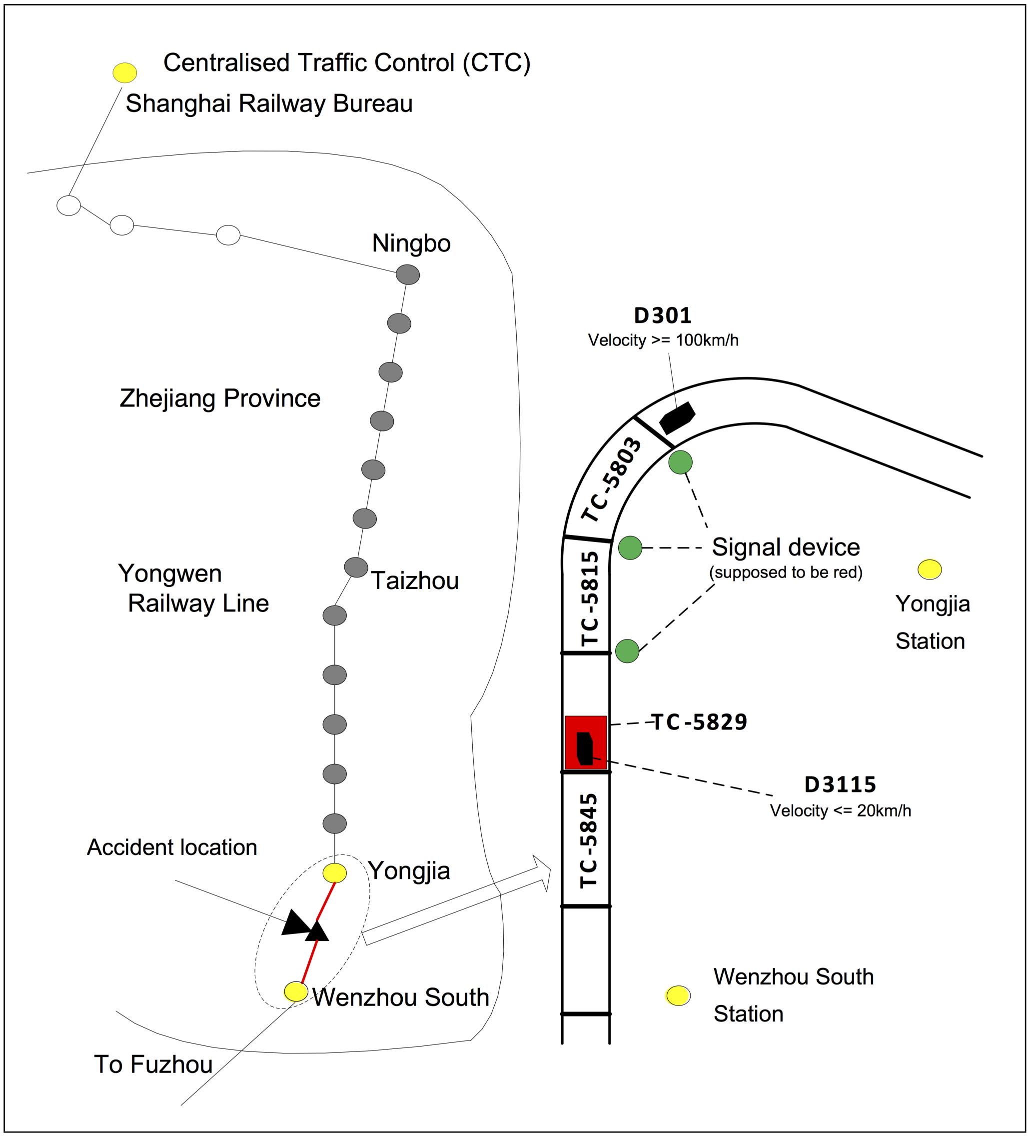

Figure 10: The Yongwen railway line and the accident.

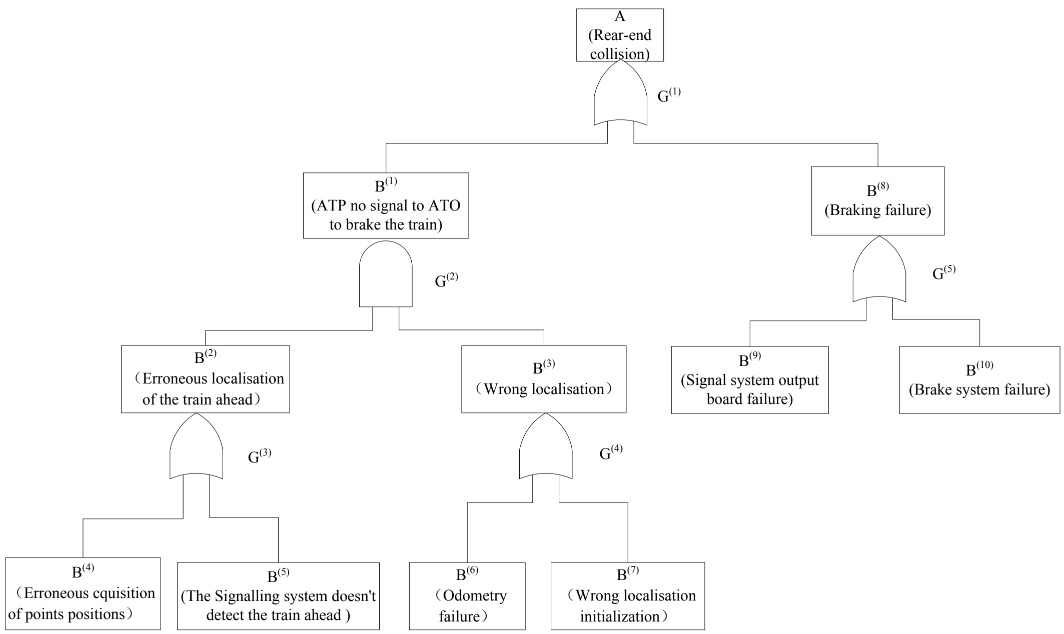

Figure 11: A fault tree representation of a CRH EMU in a rear-end collision.

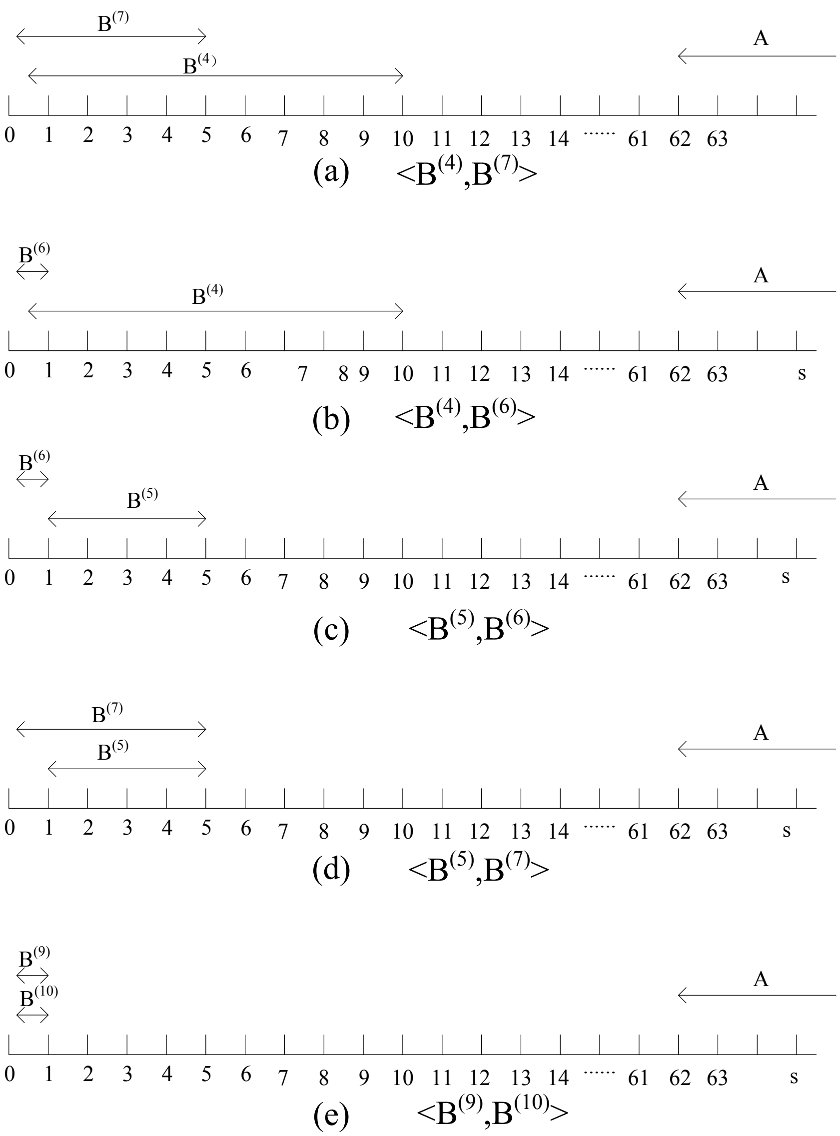

Figure 12: The actual time of MCS and hazard.