World Model, Predictive Model, and Behavioral Model of an Anticipatory Reasoning-Reacting System for Runway Incursion Prevention

Kai Shi, Kazunori Wagatsuma, Yuichi Goto, and Jingde Cheng
Department of Information and Computer Sciences, Saitama University
Saitama, 338-8570, Japan
+81-48-858-3785 - {shikai, wagatsuma, gotoh, cheng}@aise.ics.saitama-u.ac.jp
- http://www.aise.ics.saitama-u.ac.jp

Abstract
An anticipatory reasoning-reacting system anticipates based on anticipatory reasoning, which can draw new, previously unknown and/or unrecognized conclusions about some future event or events whose occurrence and truth are uncertain at the point of time when the reasoning is being performed. To perform anticipatory reasoning, we need to express the real world, predictive laws and behavioral patterns of the target domain as empirical theories represented by logical formulas which called world model, predictive model, and behavioral model correspondingly. However, there is no case to show what these models are and how to construct these models. To this end, this paper proposes a general procedure to construct these models, and presents a case study of runway incursion prevention. Besides, this paper also discusses the evaluation of the models.
Keywords : anticipatory reasoning-reacting system, prediction, anticipation, anticipatory reasoning, runway incursion.

1 Introduction

Anticipatory reasoning-reacting systems (ARRS) was proposed as a new type of reactive systems to provide highly reliable and highly secure systems to satisfy requirements of advanced applications [8]. An ARRS predicts possible failures and attacks by using anticipatory reasoning about failures and attacks based on logic systems, empirical knowledge and detected omens, informs its users about possible failures and attacks, and performs some operations to defend the system from possible failures and attacks anticipatorily by itself [15].

ARRS's key function is its ability of anticipatory reasoning, which can draw new, previously unknown and/or unrecognized conclusions about some future event or events whose occurrence and truth are uncertain at the point of time when the reasoning is being performed [9]. To perform anticipatory reasoning, we must have the knowledge of the real world, predictive laws, and behavioral patterns, while all these knowledge must be represented as logical formulas of a specific logic (or logics). We call these knowledge *anticipatory models*, which describe the specific theories or facts about the target domain.

In ARRSs, the anticipatory models are divided into three types called world model, predictive model, and behavioral model, which represent the knowledge of the real world, predictive laws, and behavioral patterns correspondingly. In an ARRS, we use the world model and the predict model to reason for predictions, while use the world model and the behavioral model to reason for next anticipatory actions.

It is a primary and difficult task to construct anticipatory models for a practical ARRS. The knowledge of one domain is rarely little and well-organized. We must analyze the experiential knowledge and extract theories and patterns. Even though we have built a model which can be described with natural language, we need choose a appropriate formal logic and figure out how to represent all these knowledge as logical formulas in the ARRS. However, there is no case to show what these models are and how to construct the models. Moreover, there is no general modeling methodology to guide people construct anticipatory models of ARRSs. In addition, there is no method to evaluate an anticipatory model.

This paper presents a general process to construct anticipatory models for ARRSs. By using a case study, this paper shows how to construct the world model, predictive model, and behavioral model of an ARRS for runway incursion prevention. This paper uses runway incursion prevention as a concrete problem, because: (1) runway incursion is one of the most serious problem threats to safety in air transport operations, thus the runway incursion prevention system should be reactive and highly reliable, (2) the runway incursion prevention system should not only detect existing runway incursion, but also aware the potential runway incursion, (3) even if the runway incursion prevention system can detect or predict the runway incursion, in addition, the system should give instructions about how to handle the crisis. This paper also discusses how to evaluate the models, and shows the assessment of the models we built.

2 Anticipatory Reasoning-Reacting Systems

2.1 Logical Basis of ARRSs

Reasoning is the process of drawing new conclusions from given premises, which are already known facts or previously assumed hypotheses. *Anticipatory reasoning* is a reasoning to draw new, previously unknown and/or unrecognized conclusions about some future event or events whose occurrence and truth are uncertain at the point of time when the reasoning is being performed [9]. To represent, specify, verify and reason about various objects in the real world and relationships among them in the future, any ARRS needs a right fundamental logic system to provide a criterion of logical validity for anticipatory reasoning as well as formal representation and specification language.

Logic deals with what entails what or what follows from what, and aims at determining which are the correct conclusions of a given set of premises. In logic, a sentence in the form of "if ... then ..." is usually called a *conditional proposition* or simply *conditional* which states that there exists a relation of sufficient condition between the "if" part and the "then" part of the sentence. In general, a conditional must concern two parts which are connected by the connective "if ... then ..." and called the *antecedent* and the *consequent* of that conditional, respectively.

A *formal logic system* L is a doublet $(F(L), \vdash_L)$ where $F(L)$ is a formal language which is the set of all *well-formed formulas* of L, and $\vdash_L$ is a *logical consequence relation* of L such that for $P \subseteq F(L)$ and $c \in F(L)$, $P \vdash_L c$ means that within the framework of L, c is a valid conclusion of premises P. For a formal logic system $(F(L), \vdash_L)$, a *logical theorem* t is a formula of L such that $\emptyset \vdash_L t$ where $\emptyset$ is the empty set. We use $Th(L)$ to denote the set of all logical theorems of L. A *formal theory* with premises P based on L, called a *L-theory with premises P* and denoted by $T_L(P)$, is defined as $T_L(P) =_{df} Th(L) \cup Th^e_L(P)$, and $Th^e_L(P) =_{df} \{\text{et} \mid P \vdash_L \text{et and et} \notin Th(L)\}$ where $Th(L)$ and $Th^e_L(P)$ are called the *logical part* and the *empirical part* of the formal theory, respectively, and any element of $Th^e_L(P)$ is called an *empirical theorem* of the formal theory.

Cheng proposed a new family of relevant logics system [3, 4], named *strong relevant logics* (SRLs) [6, 7], to underlie anticipatory reasoning. For discovery and predictions, reasoning based on SRLs is more effective than reasoning based on classical mathematical logic, because reasoning based on SRLs does not need to deal with the large number of useless logical theorems by rejecting the implicational paradoxes [3, 4] in classical mathematical logic.

There are several logics for different purposes deriving from SRLs: *Temporal relevant logic* [9] is to represent and reason about temporal knowledge, by introducing temporal operators, related axiom schemata and inference rules into SRLs. *Deontic relevant logic* [23] deals with normative notions such as obligation (ought), permission (permitted), and prohibition (may not) for underlying normative reasoning by introducing the deontic operators and related axiom schemata and inference rules into SRLs. *Spatial relevant logic* [12] is to represent and reason about spatial knowledge, by introducing temporal operators, related axiom schemata and inference rules into SRLs. *Spatio-temporal relevant logic* [11] is to represent and reason about spatial knowledge, by introducing temporal operators, related axiom schemata and inference rules into SRLs. *Three-dimensional spatio-temporal relevant logic* [13] is to represent and reason about mobile three-dimensional geometric objects, by introducing predicates and axiom schemata about solid-region connection, predicates and axiom schemata about point position and adjacency, and predicates and axiom schemata about movement of mobile objects into temporal relevant logics.

2.2 Overview of ARRSs

The most important feature of ARRSs is it can anticipate based on anticipatory reasoning. *Anticipation* is the action of taking into possession some thing or things beforehand, or acting in advance so as preclude the action another [20]. Anticipation can be divided into two phases. The first phase is *prediction*, which is the action to make some future events known in advance, especially on the basis of special knowledge, or statements about the future events. And the second phase is taking some actions according to the predictions.

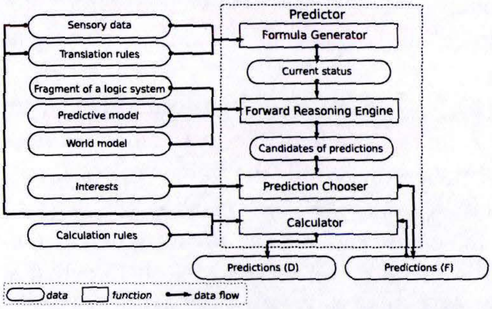

Fig. 1: Data flow diagram of a predictor

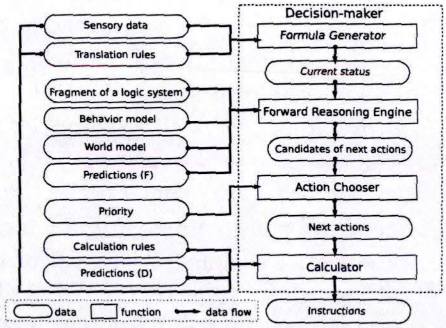

Fig. 2: Data flow diagram of a decision-maker

Now we present the procedure of a practical ARRS to predict and take actions and the usage of the world model, predictive model, and behavioral model [15] in ARRSs. In ARRSs, the common components include a predictor (Pr), a decision maker (DM), a logical theorem database (LTDB), and an empirical theory database (ETDB). A predictor receives several kinds of data and outputs predictions with quantitative information and predictions without quantitative information. Figure 1 shows a data flow diagram of a predictor. The formula generator translates the data of current status into logical formulas according to the translation rules. Then forward reasoning engine gets logical formulas which represent the current status, a fragment of a logic system, a predictive model, and a world model, and deduces candidates of predictions. Prediction chooser chooses non-trivial predictions from the candidates of predictions according to interests. After that calculator adds quantitative information to predictions chosen by the prediction chooser according to calculation rules, translation rules, and sensory data. The predictions with quantitative information (Prediction(D)) and predictions without quantitative information (Prediction(F)) are sent to a decision-maker. A decision-maker receives two kinds of predictions from a predictor, and outputs instructions to an ARRS. Figure 2 shows a data flow diagram of a decision-maker. In the decision-maker, forward reasoning engine gets logical formulas translated at the formula generator, a fragment of a logic system, predictions without quantitative information, a behavioral model, and a world model, and then it deduces candidates of next actions. Action chooser chooses appropriate actions from the candidates of next actions according to priority. Calculator adds quantitative information to next actions that the action chooser chose by using predictions with quantitative information, calculation rules, translation rules, and sensory data. The next actions with quantitative information are outputted as instructions. The logical theorem database stores fragments of logic systems underlying anticipatory reasoning or reasoning about actions. The empirical theory database stores empirical theories of a target domain as predictive models, behavioral models, or world models.

3 Model Construction of ARRSs

3.1 Features of the Models

A model is a simplified representation of something [14], which is represented by a set of logical formulas of a specific logic (or logics) in ARRSs. A *world model* is a set of empirical theories represented by logical formulas in the target domain except empirical theories related with time and behavior [15]. A *predictive model* is a set of empirical theories which are represented by logical formulas and related with time in a target domain of the system [15]. And a *behavioral model* is a set of empirical theories that are represented by logical formulas and related with behavior in a target domain of the system [15].

The world model has two functions: first we use the world model to represent any status of the real world in target domain, second the world model includes essential empirical knowledge (not related with time and behavior) about the real world in target domain. We consider the real world consists of objects, i.e., each individual in real world is an object. When taking space problem into account, we also regard a region as an object. Thus the current status of the real world is divided into two kind of information: the information of an object itself, and the information of relationships between objects. The information of an object itself generally includes its properties and status. The information of relationships between objects depends on the objects involving in the relationship. Besides, there are some concerned events in the real world. An event means a thing that happens, such as "it begins to rain." A classic typology distinguishes four sorts of events: activities, accomplishments, achievements, and states [22, 25]. In ARRSs, an event generally refers to a crisis, a failure, an attack, etc. Moreover there are also empirical knowledge (not related with time and behavior) about the real world, thus the world model must cover these knowledge and represent these knowledge as conditionals. A piece of empirical knowledge (not related with time and behavior) is like "if you walk in a heavy rain with an umbrella, then the umbrella is wet."

The predictive model is to represent the predictive knowledge used to make prediction. Predictive knowledge is time-related. And a piece of predictive knowledge specifies, that when a certain state (both past and current) occurred, some following state will be true in the future, such as "if the sky is cloudy and the air pressure is low, then it will probably rain soon." In ARRSs, these knowledge is a set of conditionals whose consequent of the future is true if and only if the antecedent holds. For any prediction, both the predicted thing and its truth must be unknown before the completion of that prediction. Therefore, the conclusion should not include the knowledge what we have known. Besides, the premises and conclusion should be relevant. Furthermore, We are only interested in certain kinds of predictions, while in the target system, they are mainly accidents/attacks and other predictions which can help to predict accidents/attacks.

The behavioral model is used to represent the behavioral knowledge used to choose action. For example, "if it rains now, then take the umbrella with you." is a piece of behavioral knowledge, while "take the umbrella" is an action. While the behavioral knowledge may be related with predictions, such as "if it will probably rain soon, then take the umbrella with you." The example's premise is a prediction, thus the action "take the umbrella" is an anticipatory action. The behavioral model has the following functions. First the behavioral model represents all behaviors involving in anticipation of the objects. Second it specifies that, when a certain event or state (such as crisis, attack) occurs, which actions should be taken. Third it specifies that, when a prediction about certain event or state is made, which anticipatory actions should be taken. The purpose of qualitative decision is to find out "what actions should be taken?" and "which actions should be taken first?". Thus, the results of qualitative decision is a set of candidates of next actions, which are labeled as "obligatory"/"permitted" and/or priorities. Therefore, the behavioral model is assembled by conditionals which can be used to get these results of qualitative decision.

3.2 General Procedure of Model Construction

Because a model is a set of logical formulas of a specific logic (or logics) in ARRSs, before we build a model, we must choose a logical basis for the model. Such a logic must satisfy the following requirements [9]. First, the logic must be able to underlie relevant reasoning as well as truth-preserving reasoning in the sense of conditional. Second, the logic must be able to underlie ampliative reasoning. Third, the logic must be able to underlie paracomplete and paraconsistent reasoning. Especially, the logic for prediction must be able to underlie temporal reasoning. In Section 2.1, we list some candidates of logical basis and their applicability. For a specific ARRS, we can use different logics for the world model, behavioral model, and predictive model, while we should make sure these logics do not conflict.

To construct world model, we use a vocabulary of predicates or constant terms to present the status of the real world. And if necessary, we collect and represent empirical knowledge (not related with time and behavior) as conditionals. The construction of the world model involves following steps. (1) In the target domain, list possible objects and their properties and statuses. (2) List relationships among the objects (if have). (3) List possible events and conditions produced by the environment we concern. (4) Determine the essential empirical knowledge (not related with time and behavior). (5) Formalize the information. For information got from step (1) to (3), we use a vocabulary to represent, while for information got from step (4), we use conditionals to represent.

The construction of the predictive model involves following steps. (1) Determine which events/conditions/statuses (accidents/attacks and other predictions which can help to predict accidents/attacks) we concern in the target domain, and list these events/conditions. (2) Determine the predictive knowledge related with these events/conditions. (3) Formalize the knowledge as conditionals, which satisfy that its antecedent is about the current and the past states, its consequent is about the future, any subformula in the consequent should not appears in the antecedent, the antecedent and consequent should share at least one same variable.

The construction of the behavioral model involves following steps. (1) List possible actions that the system can take to prevent accidents/attacks. (2) Decide which events/conditions cause to take these actions. (3) Formalize the information. For information got from step (1), we use a vocabulary to represent, while for information got from step (2), we use conditionals to represent, while such conditionals must satisfies that its consequent is about actions, its antecedent is about either event/state or prediction of event/state.

The procedure of construction above is general, may varies when build different models for different target problem. The more detail will be elaborated in the Section 4.

3.3 Evaluation Issues

In this paper, the evaluation issues are considered to have two aspects: (1) how to evaluate a general procedure of model construction, and (2) how to evaluate the constructed models.

To evaluate a general procedure, we have the following evaluation questions: (1) Does anyone can construct good models by using the process? (2) Does the procedure can apply to any target domain? (3) Does anyone can construct models easily? All these questions are not easy to answer, thus this paper does not discuss them deeply.

To evaluate models, we have the following evaluation factors/questions. (1) Completeness: Does the model include all necessary information of the target domain? (2) Relevance: Does the model include non-related information? (3) Generality: Does the model can be used for many cases in the target domain? (4) Correctness: Does the model map the information to logical formulas correctly? All these questions are not easy to answer, thus this paper does not discuss them deeply. In practice, we can also use *test sets* to evaluate models. A test set is a set of empirical/historical data about crises, attacks, failures, etc. in the target domain. If a test set is big enough, we can find out whether a model is complete, relevant, general, and correct. Besides, the test set can also used to improve the models.

4 Construction of Models for Runway Incursion Prevention

4.1 Overview of Runway Incursion Prevention

A runway incursion occurs whenever an aircraft, vehicle, person or object on the ground creates a collision hazard with an aircraft that is taking off or landing at an airport under the supervision of Air Traffic Control (ATC) [1]. The world's deadliest aviation accident was the result of a runway incursion which occurred on March 27, 1977, when two commercially operated Boeing 747s collided, killing 583 people on board both aircraft. [24]. The number of incursions reported in the U. S. rose from 186 in 1993 to 431 in 2000, an increase of 132 percent [24]. Several States and international organizations have embarked on extensive programs to reduce the risk of runway incursions. A number of factors are likely to be responsible for the continuing increase in runway incursions, including traffic volume, capacity-enhancing procedures and aerodrome design [19].

Traditionally, to prevent runway incursions, both pilots and controllers rely on visual cues, occasional communications by radio, and their memories [27]. Pilots rely on visual aids such as airfield markings, signs, and lighting, in conjunction with a paper chart of the airport to navigate from point to point on the surface. Pilots use a radio channel to obtain from ATC a route to follow while on the surface. Generally, a ground controller will issue this taxi route to pilots using explicit instructions and a strict protocol so that there is no misunderstanding of voice communications. The pilot must then memorize this route, or write it down, re-state it to the controller for confirmation, and then follow the signs and markings to the destination while avoiding other surface traffic and obstructions. Meanwhile, the ground controller must remember the routes given to all aircraft and monitor aircraft movements so that no one is directed into a potential conflict. If there is a potential for conflict, hold-in-position instructions can be issued over the radio channel to constrain aircraft movements. Flight crews perform surveillance on the airport surface using primarily the "see and avoid" principle to maintain safe separation. Similarly, ATC performs the surveillance task based primarily on visual cues. Occasionally, both pilots and controllers will use radio communications to confirm positions of relevant traffic. While the traffic alerting and collision avoidance system (TCAS) provides traffic advisories to flight crews in flight, it is not intended for use on the airport surface.

In recent years, people invented several modern equipments and systems to decrease the chance of runway incursions [27]. The Federal Aviation Administration (FAA) continues to investigate upgrades to surface surveillance technology to support ATC responsibilities. Technologies such as ASDE-3/ASDE-X radars [16] (figure 3 is an example of the ASDE-X display), multi-lateration systems, and in-pavement loops have been tested. Also, the airport movement area safety system (AMASS) [26] has been developed with the hope of detecting hazardous situations on the surface and alerting ATC using data derived from an airport surveillance system. NASA developed the runway incursion prevention system (RIPS) [5] which integrates several advanced technologies into a surface communication, navigation and surveillance system for flight crews and air traffic controllers. RIPS combines a head-down cockpit display of an electronic moving map of airport runways and taxiways with a head-up display that gives the pilot real-time guidance. The system shows and sounds an alert if another plane or vehicle is about to encroach onto the runway. RIPS also uses specially developed computer software, GPS signals and ground technologies developed by the FAA's Runway Incursion Reduction Program.

However almost all existing runway incursion prevention systems are until now passive. These system mainly give warnings in response to the specific circumstances, but have no ability to do something actively and anticipatorily by themselves. Because of the high speed of aircraft, we still need new method to gain more time before the collision happens. If we can predict by using ARRS, we may save more time to react. Besides at the critical moment, human being are tend to make mistakes, but machine do not. The ARRS can choose anticipatory actions according the current or future critical state, and even take the actions in place of human.

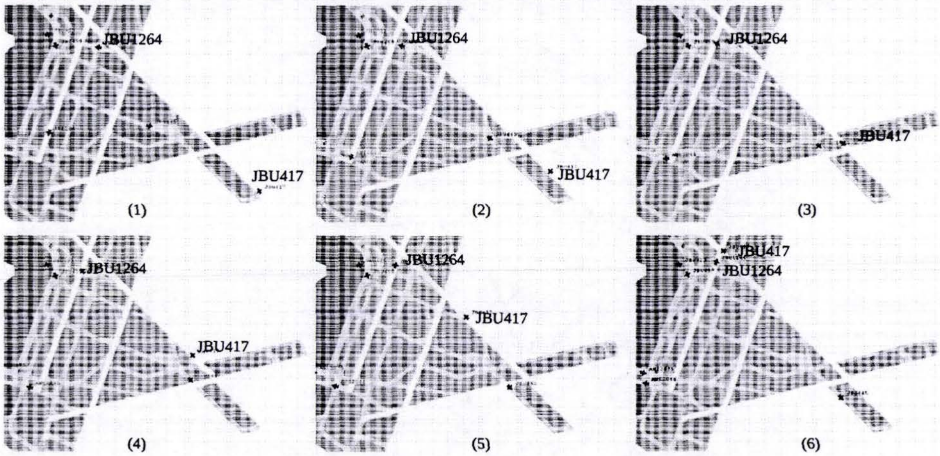

Fig. 3: Almost collision on the runway at BOS (ASDE-X display) [17]

4.2 A Scenario of Runway Incursion

Now we give a classical scenario of runway incursion. The scenario is about two planes' almost collision on the runway at Boston's Logan Airport the night on 24 Nov 2010 [17]. Figure 3 is the screenshots of ASDE-X display which recorded the whole process. JetBlue flight 1264 had just landed before taking a wrong turn on its way to the arrival gate, putting it directly in the path of another plane JetBlue 417 speeding up on the runway for takeoff. At the last second, the air traffic controller Mark Libby noticed this and radioed the JetBlue flight to stop from entering the runway where the other plane was about to take off.

4.3 World Model

Because the runway incursion involves mobile three-dimensional (in fact two-dimensional) geometric objects, we choose three-dimensional spatio-temporal relevant logic as the logical basis for the world model. For the same reason, we choose three-dimensional spatio-temporal relevant logic as the logical basis for the predictive model too.

We determine the objects involve in runway incursion include the following objects: aircrafts, vehicles (which could contact with controllers), other objects (such as vehicle or people is detected by radar or other method, but the controllers can not contact with it). Because this case is related to space, we need to investigate the regions, which we also view as objects. For one airport, all regions compose a two-dimensional map which can be expressed as an airport diagram. Figure 4 shows the airport diagram of Boston Logan International Airport. Because the ramps, terminals and hangars have no relationship with runway incursion, we take for granted that the airport is only constructed by two kind of regions, one is the runway, another is the taxiway. To be more specific, every runway or taxiway is made of intersection(s) and way(s). An intersection is a junction where two

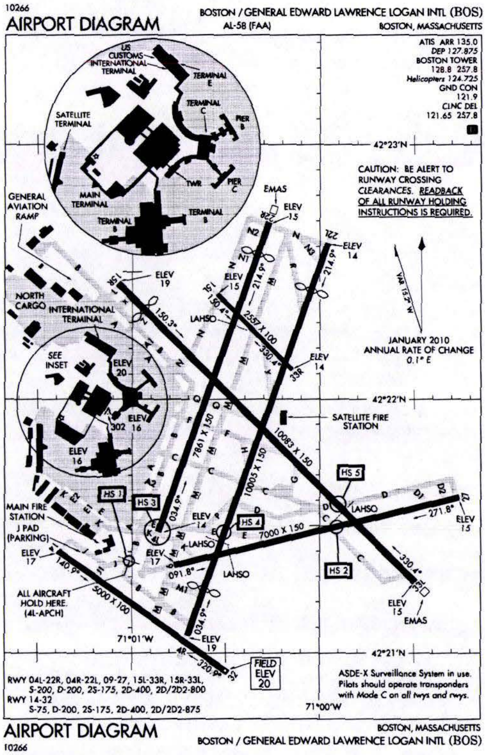

Fig. 4: Airport diagram (BOS) [2]

or more runway(s) or taxiway(s) cross each other. And a way is the part of a runway or taxiway between two intersections, that means there is no intersection in one way. So we classify the regions into four categories: runways, taxiways, intersections, and roads. After all categories of objects and regions are determined, we express these objects and regions as logical formulas. Such a formula is constructed by a predicate and a term which is logical expression that refers to objects [21]. We define unary predicates to construct these formulas, e.g., formula

$Aircraft(o_1)$

means "o_1 is an aircraft". Then we build the predicate vocabulary of all objects and regions of the runway incursion prevention problem, shows in table 1.

Then we determine the properties and statuses of for each objects. We define a vocabulary of unary predicate to represent these properties and statuses, shows in table 2. In this case, we only cares the status of aircraft, status of runway, and runway incursion.

To determine the relationships between different objects and regions, because three-dimensional spatio-temporal relevant logic defines a lot of predicates to represent these relationship [13], we use these predicates as the first choice, afterwards, we define new predicates to represent special relationship of the target domain. In this paper, we utilize predicates

Table 1: Vocabulary of objects/regions

Formula	Meaning
$Aircraft(o_1)$	o_1 is an aircraft.
$Vehicle(o_1)$	o_1 is a vehicle.
$Other(o_1)$	o_1 is an object can not communicate with controllers.
$Runway(r_1)$	r_1 is a runway.
$Taxiway(r_1)$	r_1 is a taxiway.
$Intersection(r_1)$	r_1 is an intersection.
$Way(r_1)$	r_1 is a section of way.

Table 2: Properties, statuses, or events of objects/regions

Object	Formula	Meaning
Aircraft	$TakingOff(o_1)$	o_1 is an aircraft, and it is taking off.
	$Landing(o_1)$	o_1 is an aircraft, and it is landing.
	$CanNotStop(o_1)$	Aircraft o_1 reaches the takeoff speed and can not stop.
Runway	$RI(r_1)$	An runway incursion happens on runway r_1.
	$Active(r_1)$	Runway r_1 is occupied to take off or land.

$A(o_1, p_1)$

means "object o_1 arrives at point p_1 (A 'point' means a spatial point. And variable p_1 is called point variable.)",

$C(r_1, r_2)$

means "region r_1 connects to region r_2",

$I(p_1, r_1)$

means "point p_1 is in region r_1", and

$Pa(r_1, r_2)$

means "r_1 is part of r_2" from three-dimensional spatio-temporal relevant logic[13]. The relationships between regions involve the representation of geographic information, will be discussed in the next step. The most common relationship in runway incursion prevention is the aircrafts/vehicles/other objects on the runway/taxiway/way/intersection. we can use the formula

$\exists p_1(A(o_1, p_1) \wedge I(p_1, r_1))$

to represent this relationship that an object o_1 is on region r_1. In order to deal with the special relationships in the problem, we define binary predicates

$Occupied(o_1, r_1)$

means "runway r_1 is active and it is occupied by aircraft o_1 to take off or land",

$RIby(o_1, r_1) =_{df} \exists o_2 \exists p_1 (Occupied(o_2, r_1) \wedge I(p_1, r_1) \wedge A(o_2, p_1) \wedge (o_1 \not\equiv o_2))$

means "object o_1 causes runway incursion of runway r_1",

$TakeOffFrom(o_1, r_1) =_{df} Aircraft(o_1) \wedge TakingOff(o_1) \wedge Runway(r_1) \wedge Occupied(o_1, r_1)$

means "aircraft o_1 is taking off from runway r_1",

$LandOn(o_1, r_1) =_{df} Aircraft(o_1) \wedge Landing(o_1) \wedge Runway(r_1) \wedge Occupied(o_1, r_1)$ means "aircraft o_1 is landing on runway r_1". Because the problem is related to space, to deal with the spatio-temporal problem, the next step is to determine how to represent geographic information which may be two-dimensional or three-dimensional. There are four categories of regions in this problem: runways, taxiways, intersections, and roads. In order to represent the geographic information as formulas, we need to distinguish different regions which could realized by giving an unique name to different regions. The unique name expresses as a formula construct by one predicate and one constant symbol. For example, *Taxiway*(N) stands for the taxiway N in Figure 4. Almost every runway and taxiway has a name like Figure 4. Because every runway have two name, e.g., in Figure 4 runway 15R and 33L is the same runway, we use the smaller string (by ascii order) 15R to stand for 33L. Thus, *Runway*($15R$) stands for runway 15R or 33L. Then we need name every way and intersection using a unique constant symbol. For example, in Figure 4, the first intersection from north to south of 15R can be express as $Intersection(15R\_L)$. Then the second intersection is $Intersection(15R\_X)$. The way between $Intersection(15R\_L)$ and $Intersection(15R\_X)$ is $Way(15R\_L\_X)$. And the way between $Intersection(15R\_X)$ and $Intersection(15R\_Z)$ is $Way(15R\_X\_Z)$. Because both runways and taxiways are composed by intersections and roads, the map is constructed indeed by intersections and roads. Then we use primitive binary predicate C with variable objects of way and intersection to represent the airport diagram, for example the geographic information of runway 15R (from north to south) are

$C(15R\_L, 15R\_L\_X)$,
$C(15R\_L\_X, 15R\_X)$,
$C(15R\_X, 15R\_L\_Z)$,
... ,
$C(9\_15R, 15R\_9\_C)$,
$C(15R\_9\_C, 15R\_C)$.

Similarly, we can represent the whole airport diagram like this. Besides, for each runway or taxiway, we use binary predicate *Pa* to specify which intersections and roads construct it. Here we take *Runway*($15R$) as an example (from north to south):

$Pa(15R\_L, 15R)$,
$Pa(15R\_L\_X, 15R)$,
$Pa(15R\_X, 15R)$,
... ,
$Pa(15R\_C, 15R)$.

After we finish constructing these formulas for each runway and taxiway, we get the topology construction of the airport, which include all geographic information we need.

Next step is to determine the common knowledge involving in the problem and to represent the knowledge as conditionals. Specifically, these knowledge in world model do not include knowledge related with time and behavior. Although it is difficult to find out all of the knowledge in one domain, the domain experts would help to collect these knowledge as much as possible. Here are some examples. The conditional

ETW1: $\forall o_1 \forall r_1 (Occupied(o_1, r_1) \Rightarrow Active(r_1))$
means "if a runway is occupied by any aircraft, the runway is active",

ETW2: $\forall o_1 \forall o_2 \forall r_1 \exists p_1 ((Occupied(o_1, r_1) \wedge (o_2 \not\equiv o_1) \wedge A(o_2, p_1) \wedge I(p_1, r_1)) \Rightarrow RIby(o_2, r_1))$
is rewritten from the definition of runway incursion,

ETW3: $\forall r_1 \forall r_2 ((Active(r_1) \wedge Pa(r_2, r_1)) \Rightarrow Active(r_2))$
means "for any runway which is occupied, any part (intersection(s) or way(s)) of this runway is occupied",

ETW4: $\forall o_1 \forall r_1 \forall r_2 ((Occupied(o_1, r_1) \wedge Pa(r_2, r_1)) \Rightarrow Occupied(o_1, r_2))$
is derived from ETW3,

ETW5: $\forall o_1 \forall r_1 \forall r_2 ((TakeOffFrom(o_1, r_1) \wedge Pa(r_2, r_1)) \Rightarrow TakeOffFrom(o_1, r_2))$
is derived from ETW4,

ETW6: $\forall o_1 \forall r_1 \forall r_2 ((LandOn(o_1, r_1) \wedge Pa(r_2, r_1)) \Rightarrow LandOn(o_1, r_2))$
is derived from ETW4,

ETW7: $\forall o_1 \forall o_2 \forall r_1 ((Occupied(o_1, r_1) \wedge (o_2 \not\equiv o_1) \wedge LandOn(o_2, r_1)) \Rightarrow RIby(o_2, r_1))$
is derived from the definition of runway incursion,

ETW8: $\forall o_1 \forall o_2 \forall r_1 ((Occupied(o_1, r_1) \wedge (o_2 \not\equiv o_1) \wedge TakeOffFrom(o_2, r_1)) \Rightarrow RIby(o_2, r_1))$
is derived from the definition of runway incursion.

4.4 Predictive Model

The first step is to determine which events/conditions/statuses of certain objects we concern in the specific problem. For objects, such as regions, whose statuses can not change, we do not make prediction about them, because their status can not vary by themselves over time. We only focus on those statuses of object changing over time. In this case, the only status that we concern is the location of objects, include aircraft, vehicle, and other object, because when a runway is active and there is another object locating at the runway, runway incursion happens.

The second step is to specify all of the empirical knowledge related with these statuses. The status that we concern is the location, thus the consequent of conditional must related with location. Besides the conditionals of predictive model are related time. Therefore the antecedent of the conditional is related with "now" and/or "past", while the consequent is related with "future". Here is an example.

ETP1: $\forall o_1 \forall r_1 \forall r_2 \forall r_3 ((Aircraft(o_1) \wedge Intersection(r_1) \wedge Way(r_2) \wedge Intersection(r_3) \wedge C(r_1, r_2) \wedge C(r_2, r_3) \wedge P(\exists p_1 (A(o_1, p_1) \wedge I(p_1, r_1))) \wedge (\exists p_2 (A(o_1, p_2) \wedge I(p_2, r_2)))) \Rightarrow F(\exists p_3 (A(o_1, p_3) \wedge I(p_3, r_3))))$.
This conditional is valid because aircraft can not draw back, when an aircraft reach one end of a way, the aircraft will arrive the other end. In another word, for each way connecting two intersection, if one aircraft passed one of these intersections and now is on the way, the aircraft will arrive the other intersection. In this example we use temporal operator P to indicate "it has been the case at least once in the past up to now", and temporal operator F to indicate "it will be the case at least once in the future from now" [13].

4.5 Behavioral Model

We choose deontic relevant logic as the logical basis of the behavioral model. Although we choose a different logic basis to build behavioral model, there is no conflict with previous world model we built, because the deontic relevant logics are obtained by introducing the deontic operators and related axiom schemata and inference rules into strong relevant logic [10].

The first step is to determine possible actions that the system can take to prevent accidents/attacks, and list them. For an aircraft, its behaviors include

$TaxiTo(o_1, r_1)$

means "to taxi the aircraft o_1 on the given way/intersection r_1",

$TaxiA(o_1, r_1)$

means "taxi across", i.e., "to let the aircraft o_1 cross the region r_1",

$TaxiWD(o_1, r_1)$

means "taxi without delay", i.e., "to let the aircraft o_1 cross the region r_1 (mainly an intersection of the runway) but using a minimum of time",

$TaxiI(o_1, r_1)$

means "taxi immediately", which is as same as taxi without delay, but used only in an imminent emergency,

$StopTakeoff(o_1)$

means "to let the aircraft o_1 stop taking off",

$StopLand(o_1)$

means "to let the aircraft o_1 stop landing",

$Hold(o_1)$

means "to let the aircraft o_1 hold, stop going forward, stay in the current position",

$Contact(o_1)$

means "to let the aircraft o_1 contact the controllers".

For the vehicles those could communicate with controllers (so that they can accept the commands of controllers), the behaviors include

$TaxiTo(o_1, r_1)$, $TaxiA(o_1, r_1)$, $TaxiWD(o_1, r_1)$, $TaxiI(o_1, r_1)$, $Hold(o_1)$, $Contact(o_1)$.

The meaning of them is as same as these for aircrafts. And for other objects, because controllers can not communicate and give commands to them, they have no anticipatory behavior.

The second step is to determine which events/conditions cause to take the anticipatory actions. The most critical event is runway incursion which could be present tense

$RIby(o_1, r_1)$

or future tense

$F(RIby(o_1, r_1))$.

Besides we shall consider the rules of air traffic control. For example, the rule "when an aircraft is landing or taking off, vehicles shall not be permitted to hold closer to the runway-in-use than: (1) at a taxiway/runway intersection - at a runway holding position, and (2) at a location other than a taxiway/runway intersection - at a distance equal to the separation distance of the runway-holding position." [18] specifies the status that a vehicle near the active runway should be noted, which the formulas

$\forall o_1 \forall r_1 \forall r_2 \exists p_1 (Active(r_1) \wedge Vehicle(o_1) \wedge C(r_1,r_2) \wedge A(o_1,p_1) \wedge I(p_1,r_2))$

and

$\forall o_1 \forall r_1 \forall r_2 \exists p_1 (Active(r_1) \wedge Other(o_1) \wedge C(r_1,r_2) \wedge A(o_1,p_1) \wedge I(p_1,r_2))$

express such conditions.

The third step is to specify the conditionals whose antecedent is about the statuses and events mentioned in last paragraph, while the consequent is about anticipatory actions. To deal with status $F(RIby(o_1,r_1))$, we have conditionals

ETB1: $\forall o_1 \forall r_1 (F(RIby(o_1,r_1)) \Rightarrow (O(Hold(o_1)))$,

ETB2: $\forall o_1 \forall r_1 ((F(RI(r_1)) \wedge TakeOffFrom(o_1,r_1) \wedge \neg CanNotStop(o_1)) \Rightarrow (P(StopTakeoff(o_1)) \wedge O(Contact(o_1))))$,

ETB3: $\forall o_1 \forall r_1 ((F(RI(r_1)) \wedge Landing(o_1,r_1)) \Rightarrow (P(Stopland(o_1)) \wedge O(Contact(o_1))))$.

To deal with status $RIby(o_1,r_1)$ (although runway incursion should not happen by using the ARRS, these conditionals are just in case), we have conditionals

ETB4: $\forall o_1 \forall r_1 \forall r_2 \exists p_1 \exists r_3 ((RIby(o_1,r_1) \wedge A(o_1,p1) \wedge I(p_1,r_2) \wedge Intersection(r_2) \wedge connect(r_2,r_3) \wedge \neg Pa(r_3,r_1)) \Rightarrow O(TaxiI(o_1,r_3)))$,

ETB5: $\forall o_1 \forall r_1 ((RI(r_1) \wedge TakeOffFrom(o_1,r_1) \wedge \neg CanNotStop(o_1)) \Rightarrow O(StopTakeoff(o_1)))$,

ETB6: $\forall o_1 \forall r_1 ((RI(r_1) \wedge Landing(o_1,r_1)) \Rightarrow O(StopLand(o_1)))$.

And for other conditions, we have

ETB7: $\forall o_1 \forall r_1 \forall r_2 \exists p_1 ((Active(r_1) \wedge C(r_1,r_2) \wedge Vehicle(o_1) \wedge A(o_1,p_1) \wedge I(p_1,r_1)) \Rightarrow O(Hold(o_1) \wedge Contact(o_1)))$,

ETB8: $\forall o_1 \forall o_2 \forall r_1 \forall r_2 \exists p_1 ((LandOn(o_2,r_1) \wedge C(r_1,r_2) \wedge Vehicle(o_1) \wedge A(o_1,p_1) \wedge I(p_1,r_1)) \Rightarrow (O(Hold(o_1) \wedge Contact(o_1)) \wedge P(Stopland(o_2)) \wedge O(Contact(o_2)))$,

ETB9: $\forall o_1 \forall o_2 \forall r_1 \forall r_2 \exists p_1 ((TakeOffFrom(o_2,r_1) \wedge C(r_1,r_2) \wedge Vehicle(o_1) \wedge A(o_1,p_1) \wedge I(p_1,r_1) \wedge \neg CanNotStop(o_1)) \Rightarrow (O(Hold(o_1) \wedge Contact(o_1)) \wedge P(StopTakeoff(o_2)) \wedge O(Contact(o_2)))$,

ETB10: $\forall o_1 \forall o_2 \forall r_1 \forall r_2 \exists p_1 ((LandOn(o_2,r_1) \wedge C(r_1,r_2) \wedge Other(o_1) \wedge A(o_1,p_1) \wedge I(p_1,r_1)) \Rightarrow O(Stopland(o_2) \wedge Contact(o_2)))$,

ETB11: $\forall o_1 \forall o_2 \forall r_1 \forall r_2 \exists p_1 ((TakeOffFrom(o_2,r_1) \wedge C(r_1,r_2) \wedge Other(o_1) \wedge A(o_1,p_1) \wedge I(p_1,r_1) \wedge \neg CanNotStop(o_1)) \Rightarrow O(StopTakeoff(o_2) \wedge Contact(o_2)))$.

These conditionals are easy to understand, thus we do not explain them here.

4.6 Anticipation with the Models

Because the state of the real world changes over time, we need to determine when to update the formulas which used to represent the changeable facts of the real world. The only changeable fact in this case is the location of objects. Therefore, once the location of a object move from one region to another, the fact of this object should be rebuild.

Let us show how the ARRS use these models to deal with almost collision mentioned in section 4.2. When the ARRS initializes, it builds the constant information, which include the information of objects, such as

$Aircraft(JBU1264), Aircraft(JBU417), \dots$.

and geographic information

$Runway(15R)$, $Runway(22R)$, $Taxiway(Q)$, ... , $Intersection(\underline{M}\_Q)$, $Way(Q\_4L\_\underline{M})$, $Intersection(4L\_Q)$, $Way(4L\_15R\_Q)$, $Intersection(4L\_15R)$, ... , $Pa(4L\_15R, 15R)$, ... , $C(\underline{M}\_Q, Q\_4L\_\underline{M})$, $C(Q\_4L\_\underline{M}, 4L\_Q)$, $C(4L\_Q, 4L\_15R\_Q)$, $C(4L\_15R\_Q, 4L\_15R)$, ...

Then the ARRS begin to work. The ARRS use the procedure of subsection 4.3 to generate formulas to represent the facts of the current world. When one object's status changes, the corresponding formulas of this object will update. Meanwhile the ARRS predicts for this object which status changes. For this case, at the beginning, aircraft JBU417 is taking off from runway 33L(15R) as shows in Figure 3 (1). And the aircraft JBU1264 is on $Intersection(4L\_Q)$. Thus, the current state is

$TakeOffFrom(JBU417, 15R)$,

$\exists p_1(A(JBU1264, p_1) \wedge I(p_1, \underline{M}\_Q))$, ...

Moreover, for JBU417, by using the definition of predicate $TakeOffFrom$, we get

$$Aircraft(JBU417) \wedge TakingOff(JBU417) \wedge Runway(15R) \wedge Occupied(JBU417, 15R).$$

And by using conditional ETW1, we get

$Active(15R)$.

In Figure 3 (2), when aircraft JBU1264 comes to $Way(Q\_4L\_\underline{M})$ from $Intersection(\underline{M}\_Q)$, the status about JBU1264 updates

$$\exists p_1(A(JBU1264, p_1) \wedge I(p_1, Q\_4L\_\underline{M})) \wedge P(\exists p_2(A(JBU1264, p_1) \wedge I(p_1, \underline{M}\_Q))).$$

When in Figure 3 (4), the status of JBU1264 is

$$\exists p_1(A(JBU1264, p_1) \wedge I(p_1, 4L\_15R\_Q)) \wedge P(\exists p_2(A(JBU1264, p_1) \wedge I(p_1, 4L\_Q))).$$

For every time the status of JBU1264 changes, besides generating the formulas to represent the facts of it, the ARRS also predicates for it. In the time which Figure 3 (4) shows, by using conditional ETP1, we get

$$(Aircraft(JBU1264) \wedge Intersection(4L\_Q) \wedge Way(4L\_15R\_Q) \wedge Intersection(4L\_15R) \wedge C(4L_Q, 4L\_15R\_Q) \wedge C(4L\_15R\_Q, 4L\_15R) \wedge \exists p_1(A(JBU1264, p_1) \wedge I(p_1, Q\_4L\_\underline{M})) \wedge P(\exists p_2(A(JBU1264, p_1) \wedge I(p_1, \underline{M}\_Q)))) \Rightarrow F(\exists p_3(A(JBU1264, p_3) \wedge I(p_3, 4L\_15R)).$$

On this basis, by using axiom PRCC2 [13]

$\forall p_1 \forall r_1 \forall r_2((I(p_1, r_1) \wedge Pa(r_1, r_2)) \Rightarrow I(p_1, r_2))$,

we get

$F(\exists p_3(A(JBU1264, p_3) \wedge I(p_3, 15R))$.

Furthermore, by using conditional ETW2, we get

$F(RIby(JBU1264, 15R))$,

which means "JBU1264 will cause runway incursion in runway 15R".

After we get the prediction, we use behavioral model to generate anticipatory behaviors. By using conditional ETB1, we get

$F(RIby(JBU1264, 15R)) \Rightarrow (O(Hold(JBU1264)))$,

means "JBU1264 is obligatory to stop and stay in the current position". This action is what the controller Mark Libby did that day. Besides if using conditional ETB2, and the current speed of JBU417 satisfies

$\neg CanNotStop(JBU417)$,
we get
$P(StopTakeoff(JBU417)) \wedge O(Contact(JBU417))$,
which means "JBU417 could consider about stop taking off and contact with the air traffic controllers". Comparing to the controllers, the ARRS could "think more" when facing the crisis.

4.7 Evaluation

In this section, we evaluate the models constructed for runway incursion prevention by using the test set. The paper [5] proposed some test scenarios to evaluate their runway incursion prevention system. We use these scenarios as a test set to evaluate our models. The results show our models can handle these scenarios, besides our models realize more anticipation than [5], and can choose proper actions automatically, which are made by human in [5].

5 Concluding Remarks

This paper presented the feature of anticipatory models of ARRSs, and proposed a general procedure to construct anticipatory models for ARRSs. Besides, we constructed the anticipatory models for a case study of runway incursion prevention. This paper also discussed the evaluation of both the procedure of model construction and the models.

There are several future works/questions related to this paper: Can we answer all questions presented in Section 3.3? Can we develop some tools to formalize the knowledge? Can we develop some automatic/semi-automatic tools to construct these models?

References

[1] Federal Aviation Administration. *Runway Incursion Severity Trends at Towered Airports in the United States: 1997-2000*. FAA Publication, June 2001.

[2] Federal Aviation Administration. Airport diagram (bos), April 2011.

[3] A. R. Anderson and N. D. Belnap Jr. *Entailment: The Logic of Relevance and Necessity*, volume 1. Princeton University Press, 1975.

[4] A. R. Anderson, N. D. Belnap Jr., and J. M. Dunn. *Entailment: The Logic of Relevance and Necessity*, volume 2. Princeton University Press, 1992.

[5] R. Cassell, C. Evers, J. Esche, and B. Sleep. *NASA Runway Incursion Prevention System (RIPS) Dallas-Fort Worth demonstration performance analysis*. NASA, 2002.

[6] J. Cheng. The fundamental role of entailment in knowledge representation and reasoning. *Journal of Computing and Information, Special Issue: 8th International Conference of Computing and Information*, 2(1):828–848, 1996.

[7] J. Cheng. A strong relevant logic model of epistemic processes in scientific discovery. In *Proc. Information Modelling and Knowledge Bases XI, Frontiers in Artificial Intelligence and Applications*, volume 61, pages 136–159. IOS Press, 2000.

[8] J. Cheng. Anticipatory reasoning-reacting systems. In *Proc. International Conference on Systems, Development and Self-organization*, pages 161– 165, 2002.

[9] J. Cheng. Temporal relevant logic as the logical basis of anticipatory reasoning-reacting systems. In *Proc. Computing Anticipatory Systems: CASYS - 6th International Conference, AIP Conference Proceedings*, volume 718, pages 362–375. AIP, 2004.

[10] J. Cheng. Strong relevant logic as the universal basis of various applied logics for knowledge representation and reasoning. In *Proc. the 2006 conference on Information Modelling and Knowledge Bases XVII*, pages 310–320. IOS Press, 2006.

[11] J. Cheng. Qualitative spatio-temporal reasoning about moving objects in three-dimensional space. In *Proc. 3rd International Symposium on Advances in Computation and Intelligence, Lecture Notes in Computer Science*, volume 5370, pages 637–648. Springer-Verlag, 2008.

[12] J. Cheng and Y. Goto. Representing and Reasoning about Spatial Knowledge Based on Spatial Relevant Logic. In *Proc. Conceptual Modeling for Advanced Application Domains, ER 2004 Workshops CoMoGIS, CoMWIM, ECDM, CoMoA, DGOV, and eCOMO, Lecture Notes in Computer Science*, volume 3289, pages 114–126. Springer-Verlag, 2004.

[13] J. Cheng, Y. Goto, and N. Kitajima. Anticipatory reasoning about mobile objects in anticipatory reasoning-reacting systems. In *Proc. Computing Anticipatory Systems: CASYS - 8th International Conference, AIP Conference Proceedings*, volume 1051, pages 244–254. AIP, 2008.

[14] J. Daintith and E. Wright. *A Dictionary of Computing*. Oxford University Press, sixth edition, 2008.

[15] Y. Goto, R. Kuboniwa, and J. Cheng. Development and maintenance environment for anticipatory reasoning-reacting systems. *International Journal of Computing Anticipatory Systems*, 24:61–72, 2011.

[16] D. M. McAnulty, A. Koros, and A. Poston. *Airport surface detection equipment - Model X early user involvement event.* FAA Publication, 2001.

[17] NATCA National Office. Boston logan controller's nov. 24 save, 2010. http://www.natca.com/index.aspx.

[18] International Civil Aviation Organization. *Rules of the Air and Air Traffic Services*. ICAO, thirteenth edition, 1996. Doc-4444-RAC/501.

[19] International Civil Aviation Organization. *Manual on the Prevention of Runway Incursions*. ICAO, first edition, 2007. Doc 9870 AN/463.

[20] R. Rosen. *Anticipatory systems*. Pergamon Press, 1985.

[21] S.J. Russell and P. Norvig. *Artificial intelligence: a modern approach*. Prentice Hall, 2009.

[22] G. Ryle. The concept of mind. *Hutchinson, London*, 1949.

[23] T. Tagawa and J. Cheng. Deontic Relevant Logic: A Strong Relevant Logic Approach to Removing Paradoxes from Deontic Logic. In *Proc. PRICAI 2002: Trends in Artificial Intelligence, 7th Pacific Rim International Conference on Artificial Intelligence, Lecture Notes in Artificial Intelligence*, volume 2417, pages 39–48. Springer-Verlag, 2002.

[24] K. Thomas. The increasing risk of runway incursions - the most dangerous part of air travel may be the time spent on the ground. *The Journal of Air Law and Commerce*, 67(2):545–591, 2002.

[25] Z. Vendler. Verbs and times. *The philosophical review*, 66(2):143–160, 1957.

[26] M. Watnick and J.W. Ianniello. Airport movement area safety system. In *Proc. Digital Avionics Systems Conference, 1992. Proceedings., IEEE/AIAA 11th*, pages 549–552. IEEE Computer Society Press, 1992.

[27] S.D. Young, D.R. Jones, United States. National Aeronautics, Space Administration, and Langley Research Center. Runway incursion prevention: a technology solution. In *Proc. Annual International Air Safety Seminar*, volume 54, pages 221–238. Flight Safety Foundation, 2001.

A Test Set and Results

In order to evaluate the models of runway incursion prevention, we refer to [5], and use the same test scenarios of [5]. In the test, we do not construct the precise and complete geographic information, because we do not have sufficient information. However, this does not affect the results.

A.1 Scenario 1 - Arrival/Taxi

Figure 5 shows the scenario that an aircraft $Aircraft_1$ will land on the runway R_1 while a vehicle $Vehicle_1$ on taxiway T_1 will cross the runway.

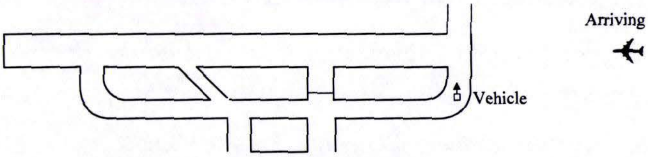

Fig. 5: Scenario 1

Let the intersection of the runway R_1 and taxiway T_1 is intersection $R_1\_T_1$, and the vehicle $Vehicle_1$ is on the way W_1. Based on the world model, we get

$Runway(R_1) \wedge Taxiway(T_1) \wedge Way(W_1) \wedge Intersection(R_1\_T_1) \wedge Pa(W_1, T_1) \wedge Pa(R_1\_T_1, T_1) \wedge Pa(R_1\_T_1, R_1) \wedge C(R_1\_T_1, W_1)$,

$Aircraft(Aircraft_1) \wedge LandOn(Aircraft_1, R_1)$,

$Vehicle(Vehicle_1) \wedge (\exists p_1 (A(Vehicle_1, p_1) \wedge I(p_1, W_1)))$.

Based on ETW6, we get

$LandOn(Aircraft_1, R_1\_T_1)$.

Although we could not predict base on the current status, if we use ETB8 of behavioral model, we get

$O(Hold(Vehicle_1) \wedge Contact(Vehicle_1)) \wedge P(Stopland(Aircraft_1)) \wedge O(Contact(Aircraft_1))$

means "the vehicle must hold in the current position and contact with the controllers, while the aircraft could think about stop landing, and must contact with the controllers".

In [5], the solution is: the aircraft gives up landing and initiate a go-around maneuver, after that the vehicle cross the runway.

A.2 Scenario 2 - Departure/Taxi

Figure 6 shows the scenario that an aircraft $Aircraft_1$ is taking off from the runway R_1 while a vehicle $Vehicle_1$ on taxiway T_1 will cross the runway.

Let the intersection of the runway R_1 and taxiway T_1 is intersection $R_1\_T_1$, and the vehicle $Vehicle_1$ is on the way W_1. Based on the world model, we get

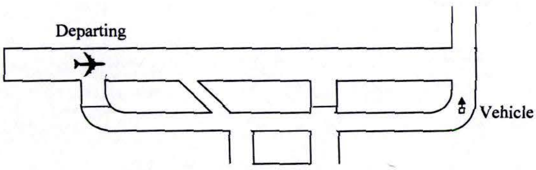

Fig. 6: Scenario 2

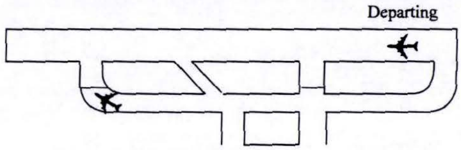

Fig. 7: Scenario 3

$Runway(R_1) \wedge Taxiway(T_1) \wedge Way(W_1) \wedge Intersection(R_1\_T_1) \wedge Pa(W_1,T_1) \wedge$
$Pa(R_1\_T_1,T_1) \wedge Pa(R_1\_T_1,R_1) \wedge C(R_1\_T_1,W_1)$,
$Aircraft(Aircraft_1) \wedge TakeOffFrom(Aircraft_1,R_1)$,
$Vehicle(Vehicle_1) \wedge (\exists p_1(A(Vehicle_1,p_1) \wedge I(p_1,W_1)))$.
Based on ETW5, we get
$TakeOffFrom(Aircraft_1,R_1\_T_1)$.
Although we could not predict base on the current status, if we use ETB7 of behavioral model, we could get
$O(Hold(Vehicle_1) \wedge Contact(Vehicle_1))$
means the vehicle must hold in the current position and contact with the controllers. Besides, if
$\neg CanNotStop(Aircraft_1)$
holds, we can use ETB9 and get
$O(Hold(Vehicle_1) \wedge Contact(Vehicle_1)) \wedge P(StopTakeoff(Aircraft_1)) \wedge$
$O(Contact(Aircraft_1))$
means "the aircraft could think about stop taking off, and must contact with the controllers".

In [5], the solution is: the aircraft reject the take off by stopping on the runway, after that the vehicle cross the runway.

A.3 Scenario 3 - Taxi/Departure

Figure 7 shows the scenario that an aircraft $Aircraft_1$ is taking off from the runway R_1 while an aircraft $Aircraft_2$ just passed the intersection I_1 and now is on the taxiway T_1.

Let the intersection of the runway R_1 and taxiway T_1 is intersection $R_1\_T_1$, and the aircraft $Aircraft_2$ is on the way W_1. Based on the world model, we get
$Runway(R_1) \wedge Taxiway(T_1) \wedge Way(W_1) \wedge Intersection(R_1\_T_1) \wedge Intersection(I_1)$
$\wedge Pa(W_1,T_1) \wedge Pa(R_1\_T_1,T_1) \wedge Pa(R_1\_T_1,R_1) \wedge Pa(I_1,T_1) \wedge C(R_1\_T_1,W_1) \wedge C(I_1,W_1)$,
$Aircraft(Aircraft_1) \wedge TakeOffFrom(Aircraft_1,R_1)$,
$Aircraft(Aircraft_2) \wedge P(\exists p_1(A(Aircraft_2,p_1) \wedge I(p_1,I_1))) \wedge (\exists p_2(A(Aircraft_2,p_2)$
$\wedge I(p_2,W_1)))$.
To use ETP1 of predictive model, we get
$F(\exists p_1(A(Aircraft_2,p_1) \wedge I(p_1,R_1\_T_1)))$.
By using axiom PRCC2 [13]
$\forall p_1 \forall r_1 \forall r_2((I(p_1,r_1) \wedge Pa(r_1,r_2)) \Rightarrow I(p_1,r_2))$,
we get

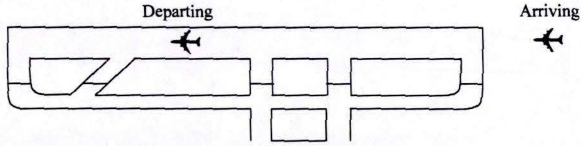

Fig. 8: Scenario 4

$F(\exists p_1(A(Aircraft_2,p_1)\wedge I(p_1,R_1))$.
Furthermore by using conditional ETW2, we get
$F(RIby(Aircraft_2,R_1))$,
means "$Aircraft_2$ will cause runway incursion in runway R_1".

Then we generate anticipatory behaviors. By using conditional ETB1, we get
$F(RIby(Aircraft_2,R_1)) \Rightarrow (O(Hold(R_1)))$,
means "$Aircraft_2$ is obligatory to stop and stay in the current position". And if using conditional ETB2, and the current speed of $Aircraft_1$ satisfies
$\neg CanNotStop(Aircraft_1)$,
we get
$P(StopTakeoff(Aircraft_1))\wedge O(Contact(Aircraft_1))$,
means "$Aircraft_1$ could consider about stop taking off and contact with the air traffic controllers".

In [5], the solution is: let the both airplanes hold in the current position.

A.4 Scenario 4 - Arrival/Departure

Figure 8 shows the scenario that an aircraft $Aircraft_1$ is taking off from the runway R_1 while an aircraft $Aircraft_2$ will land on the same runway.

Based on the world model, we get
$Aircraft(Aircraft_1)\wedge TakeOffFrom(Aircraft_1,R_1)$,
$Aircraft(Aircraft_2)\wedge LandOn(Aircraft_2,R_1)$.
Based on ETW7, we get
$RIby(Aircraft_2,R_1)$,
while based on ETW7, we get
$RIby(Aircraft_1,R_1)$.
It means both $Aircraft_1$ and $Aircraft_2$ cause the runway incursion. Then by using ETB2, we get
$P(StopTakeoff(Aircraft_1))\wedge O(Contact(Aircraft_1))$,
while by using ETB3, we get
$P(Stopland(Aircraft_2))\wedge O(Contact(Aircraft_2))$.
It means both aircraft must contact with the controllers and should consider about stop landing or taking off.

In [5], the solution is: let the both airplanes stop landing and taking off.