

# Aspect-oriented Formal Specification Modeling of Air Traffic Control System

Guo Shihang, Zhang Lichen

School of Computer Science and Technology Guangdong University of Technology Guangzhou  
510000, China

381997826@qq.com

**Keywords:** Aspect-oriented, ATCS, Formal specification, Object-Z

**Abstract.** Because of lack of aspect-oriented programming technology during the early stage of soft programming, this paper developed a method which combines the aspect-oriented programming technology with formal specification language. Aspect-oriented programming technology separates the different concerns of system. Formal specification language is a language based on logic and mathematics, so it is good to redundancy of ambiguity and reliability of system. This paper is going to extend the Object-Z to have the ability of aspect-oriented programming technology and use it to model the air traffic control system at the end.

## 1. Introduction

Cyber-physical system is a real-time system composed of computing, communication and control<sup>[1]</sup>. Any small error will cause large damage to human beings. Formal modeling language is based on rigorous mathematics, so it is more useful for reducing ambiguity than informal language. Object-Z is good object-oriented formal modeling language. So, this paper proposes a modeling of Air Traffic Control System(ATCS) using Object-Z.

Aspect-oriented method is original from object-oriented method. It is common that many objects have the same operation or not-functional factors. The aspect-oriented method extracts these same points into a common module that can be injected into the objects which need. We will extend the Object-z support the aspect-oriented ability.

## 2. ATCS System

Air Traffic Control System manages the take off, landing and flying of aircrafts to achieve the safety of flying. Its most ability is to protect aircrafts from accidents while flying. Aircrafts fly following the Route which composed of Waypoints. So, aircrafts fly from one Waypoint to another Waypoint and the sequences of Waypoints is given every time. Fig.1 show the human interface of ATCS.

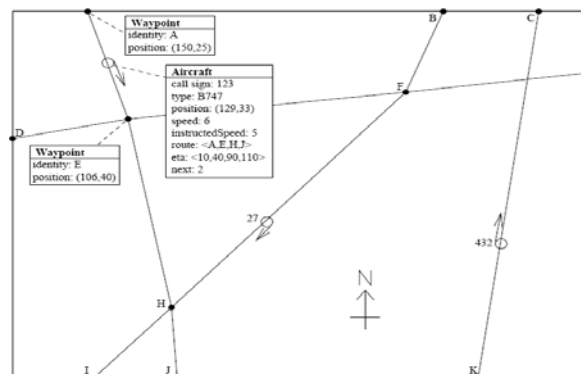


Fig.1 The Human Interface of ATCS

Because of the amounts of aircrafts increasing and the crosses of routes, aircrafts on the same route or aircrafts having the same waypoints need to keep safe distance with each other. There is two conflicts in aircrafts flying.

1) Overtaking Conflict

Overtaking Conflict means that two aircrafts fly by given sequences and the sequences have same Waypoints. Because of the same Waypoints, two aircrafts may meets the Waypoints. Fig.2 shows the Overtaking Conflict.

2) Convergence Conflict

As for Convergence Conflict means two aircrafts have same subsets of given sequences. So, in this situation, one aircraft may chase another one. Convergence Conflict is showed in Fig.3.

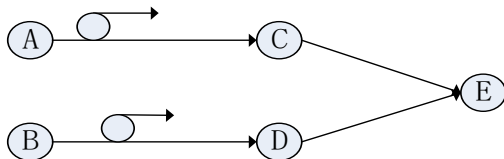


Fig.2 Overtaking Conflict



Fig.3 Convergence Conflict

### 3. ATCS Object-Z Model.

Based on the introduction of ATCS above, ATCS system have these three objects: Waypoint, Aircraft and Screen.

#### 3.1 Waypoint

Waypoint class is used to record the information of Waypoints. It includes *identity*, *position*, and *connected-Waypoints* attributes. *RouteLength* operation is used to calculate the distance of two Waypoints. Fig.4 shows the Obejct-Z model of Waypoint.

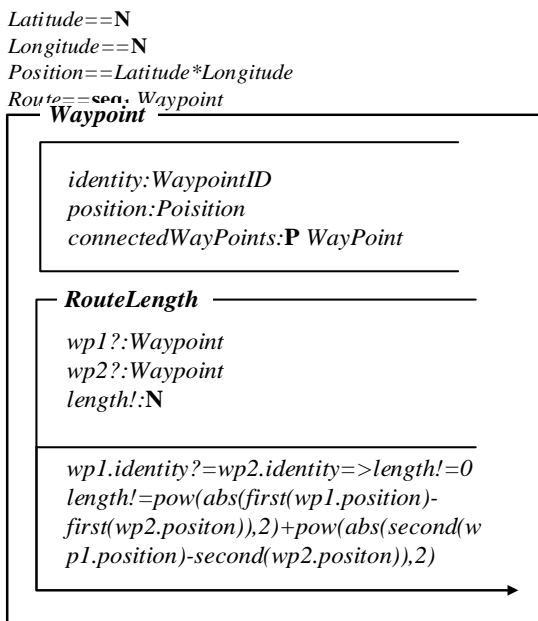


Fig.4 Waypoint Model

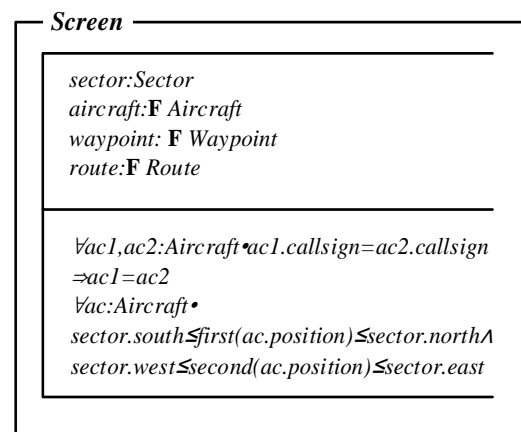


Fig.5 Screen Model

#### 3.2 Screen

Screen class is used to record the information of screen. It includes *sector*, *aircraft*, *waypoint* and *route* attributes. Its Object-Z is modeled in Fig.5.

#### 3.3 Aircraft

Aircraft class is used to record information of aircrafts. It is show in Fig.6. *route* is a sequence of Waypoints like <A,B,E,F>. *eta* is the corresponding time sequence of Waypoints when aircrafts arrive on the Waypoints. *next* means the number of Waypoints that aircrafts will arrive on.

There are some operations of Aircraft class. *Move* represents the aircrafts flying. *UpdateTime* is responsible for update time operation of an aircraft. *DistanceToWaypoint* is used to calculate the distance of a aircraft to a Waypoint. *HazardAlarm* means the alarm operation when a aircraft is too near to another aircraft.

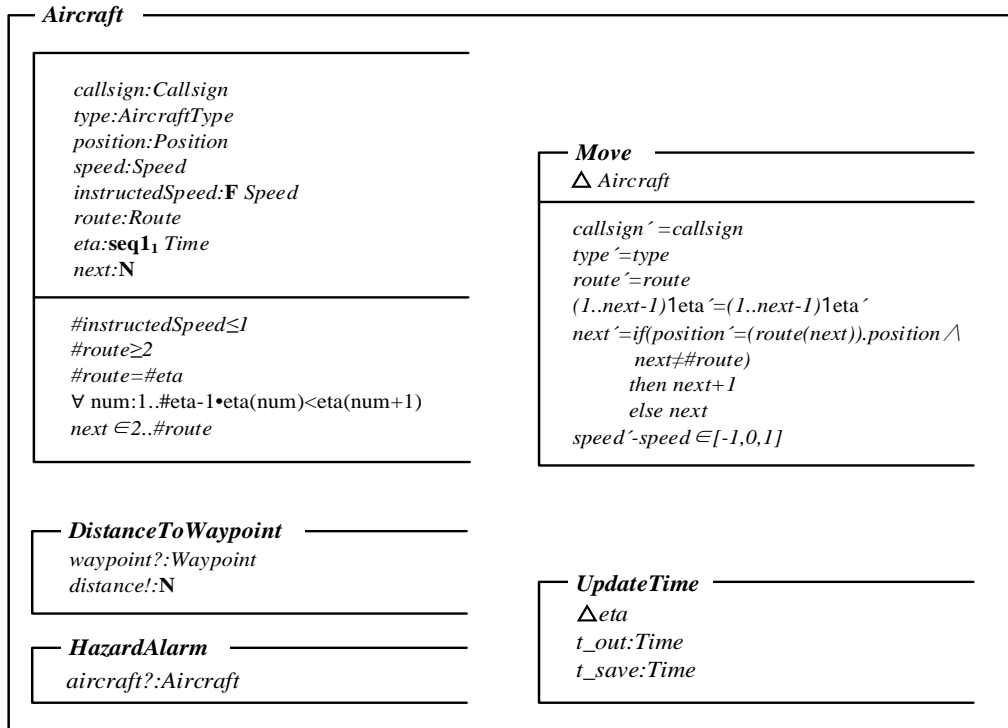


Fig.6 Aircraft Model

#### 4. Object-Z Aspect-oriented Extension.

Aspect-oriented method needs to tackle the two problems, aspect model and injection. Aspect modules solves the same points, and injection solves the way adding aspect modules into objects which need aspect modules. On the base of Object-Z meta schema, We introduce Aspect Schema to model the aspect. The Aspect Schema is illustrated in Fig.7. Fig.7 shows the PointCutState schema which defines the pointcut of aspect-oriented method. Moreover, It defines the PointCut Operation which defines the operations of aspects.

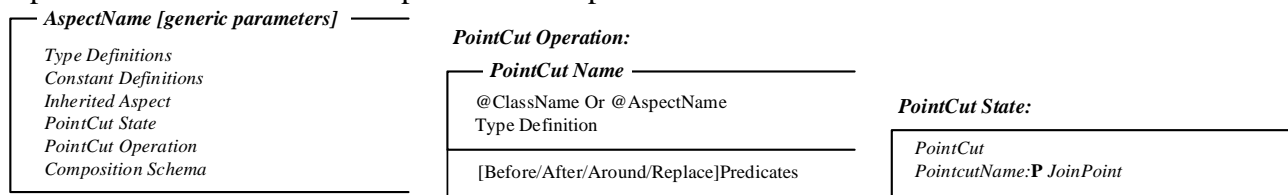


Fig.7 Aspect Schema Definition

#### 5. ATCS Aspect-oriented Model.

##### 5.1 Distance Aspect

Fig.8 is the Distance Aspect of ATCS. @Aircraft and @Waypoint means Distance Aspect crosscuts the Aircraft and Waypoint object. And the PointCut State Schema in the Distance Aspect defines two pointcuts, *DTW* and *RL*. *DTW* crosscuts the *DistanceToWaypoint* operation of Aircraft while *RL* crosscuts the *RouteLength* operation of Waypoint.

*DTW* PointCut Operation is used to calculate the distance of an aircraft to a Waypoint. It' mark is 'Replace', so it means this PointCut Operation will replate the corresponding operation of pointcuts. Its calculation formula is the sum of the distance of an aircraft to the nearest next Waypoint and the distance of left Waypoints.

*RL* PointCut Operation' mark is same as *DTW* PointCut Operation. It calculates the distance of two Waypoints. Its calculation formula is the sum of all directly near Waypoints' distances.

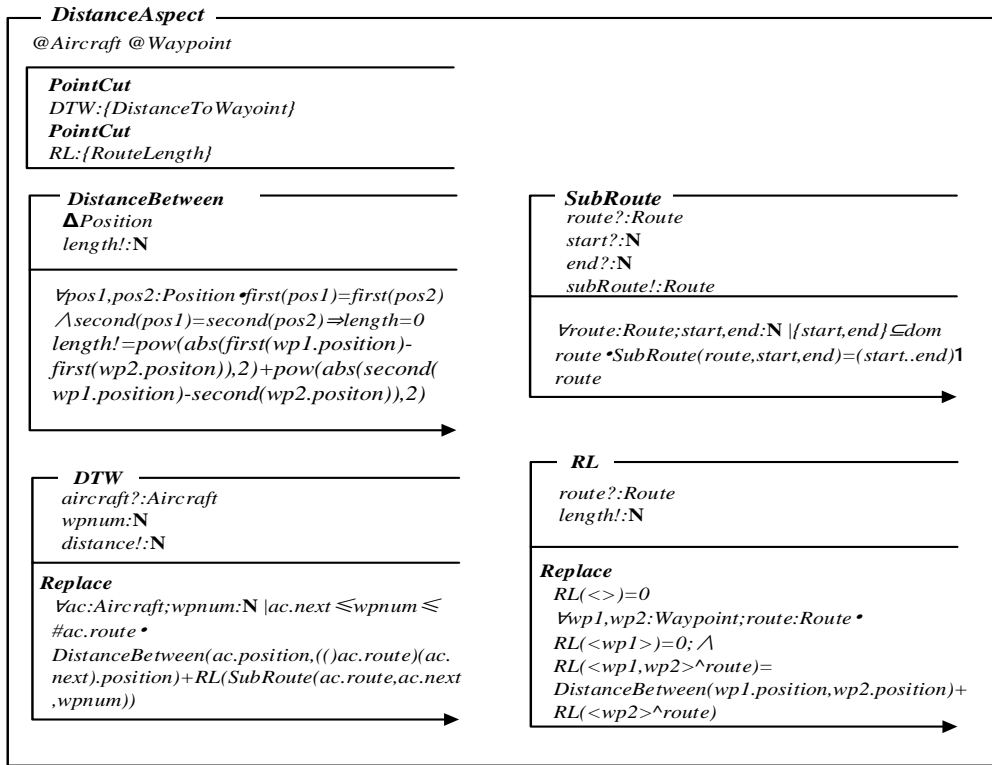


Fig.8 Distance Aspect

## 6. Summary

This paper focus on the Object-Z aspec-oriented extension. The method combines the goodness of object-oriented method and aspect-oriented method. It is useful for reduce of complexity of CPS modeling. In the future, we will pay attention to the verification and support tools.

## References

- [1]. Wang Zhong-jie, Xie Lu-lu. Cyber-physical Systems A Survey[J]. ACTA Automatica Sinica, 2011, 37(10): 1157-1166.
- [2]. Gu Tian-long. Formal Specification Method of Software Development [M]. Beijing: China Higher Education Press, 2005:159-201.
- [3]. Prakash S, Teltumde, PawanKumar I. Patil, Vishal R. Ambhire. Formal Specification Concepts in Critical Analysis[J]. IJCEM International Journal of Computational Engineering & Management, 2011, 14(1):128-134.
- [4]. Zhu Chen-xi. The analysis and design method of Cyber-Physical System based on AADL[D].Guangzhou: Guangdong University of Technology, 2014.05.
- [5]. YU Hui-qun, XU Hao, LIU Dong-mei, SHAO Zhi-qing. An Aspect-Oriented Modeling Method Based on Object-Z [J]. Journal of East China University of Science and Technology, 2007, 33(3): 383-388.