SOFTWARE REQUIREMENTS

Railway Control Center

Author: Group 29 Yang Fuyi

Table of Contents

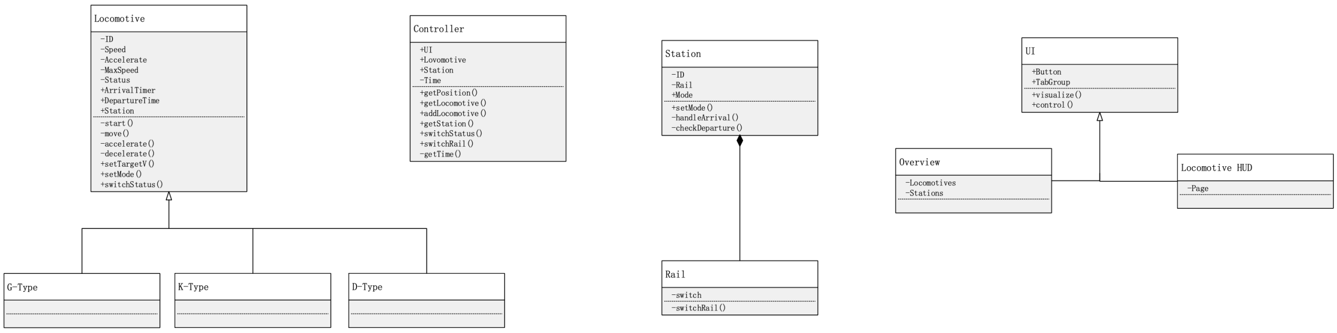
System Objective

In this project, we are going to develop a software(Railway Control Center) which can show how the locomotives bewtween stations work every moment, and design an automated algorithm to control them according to the train schedule published by maintainer, and we can also control trains by manpower.

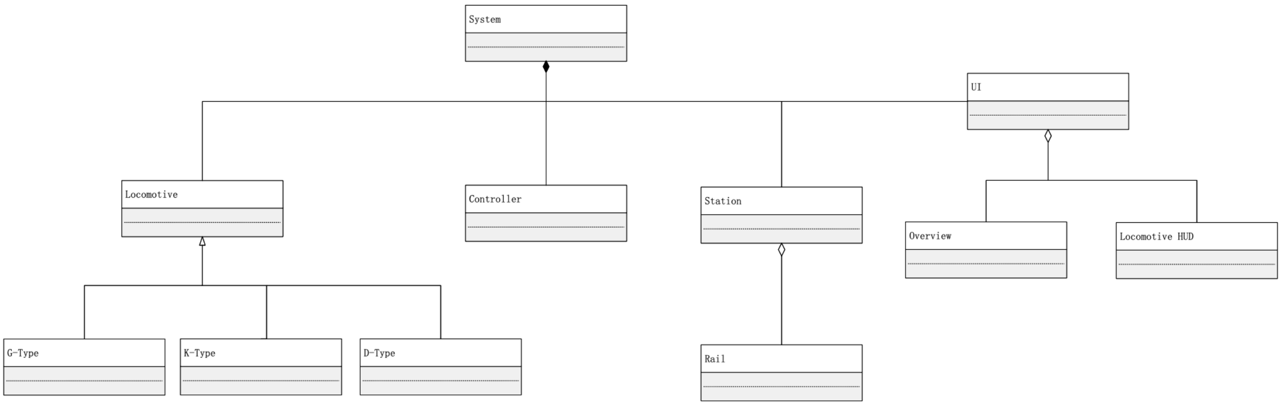
The software will take care of the graphic interface, display realtime movements of locomotives and determine how locomotive arrive, departure and order of into station and out station, by using switches. Also, there is a HUD provided for staff to control the locomotives speed or some other parameters by manpower. Besides, we will count all delays of locomotives.

Domain Analysis

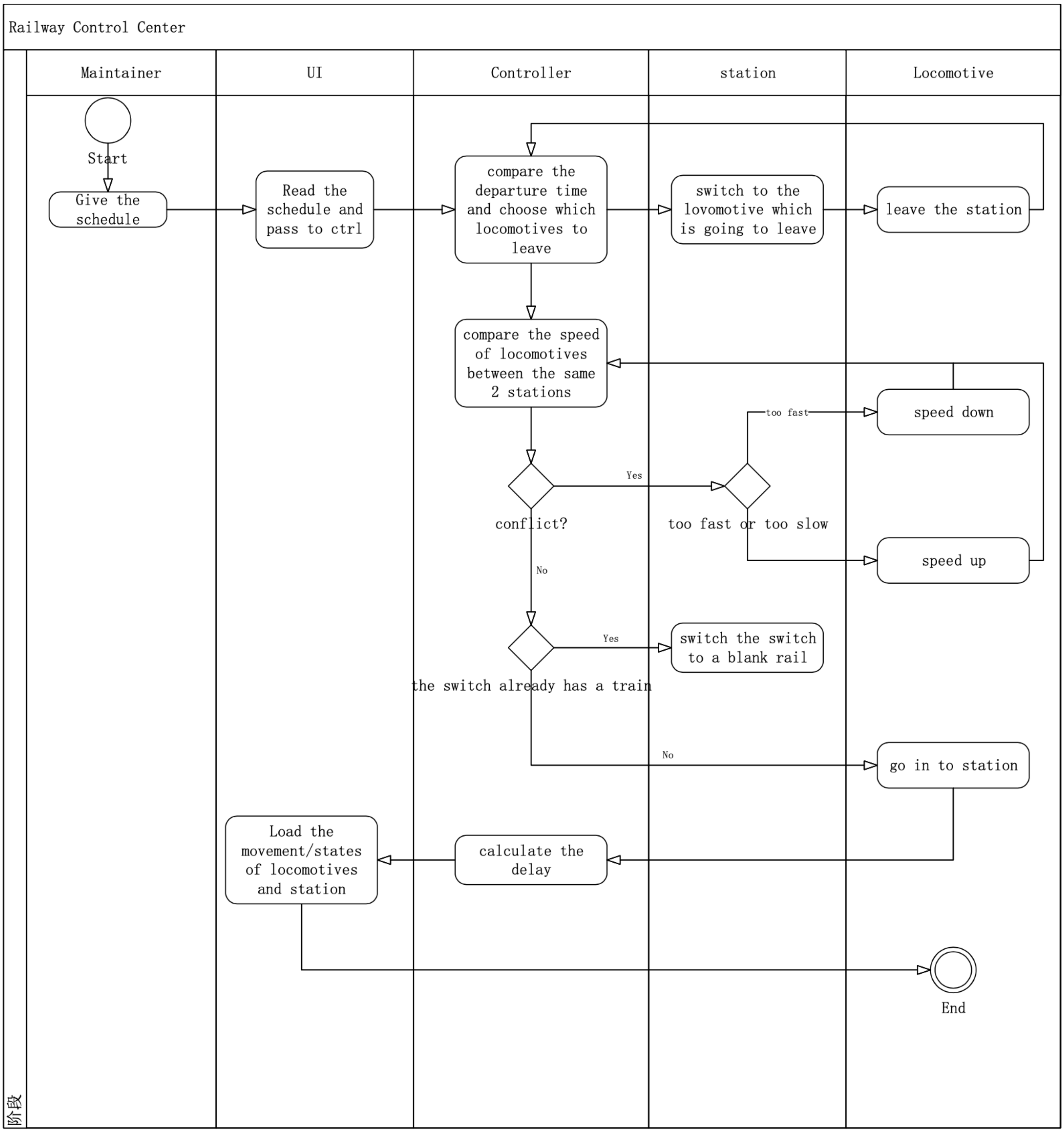
The participants of activities of the system can be categorized into the 3 types of locomotives(1 direction), a central controller ,7 stations(includes start and end) with 1 rail and n switches , a user interface for customer to see.



The relationship among different participants are shown as follows:

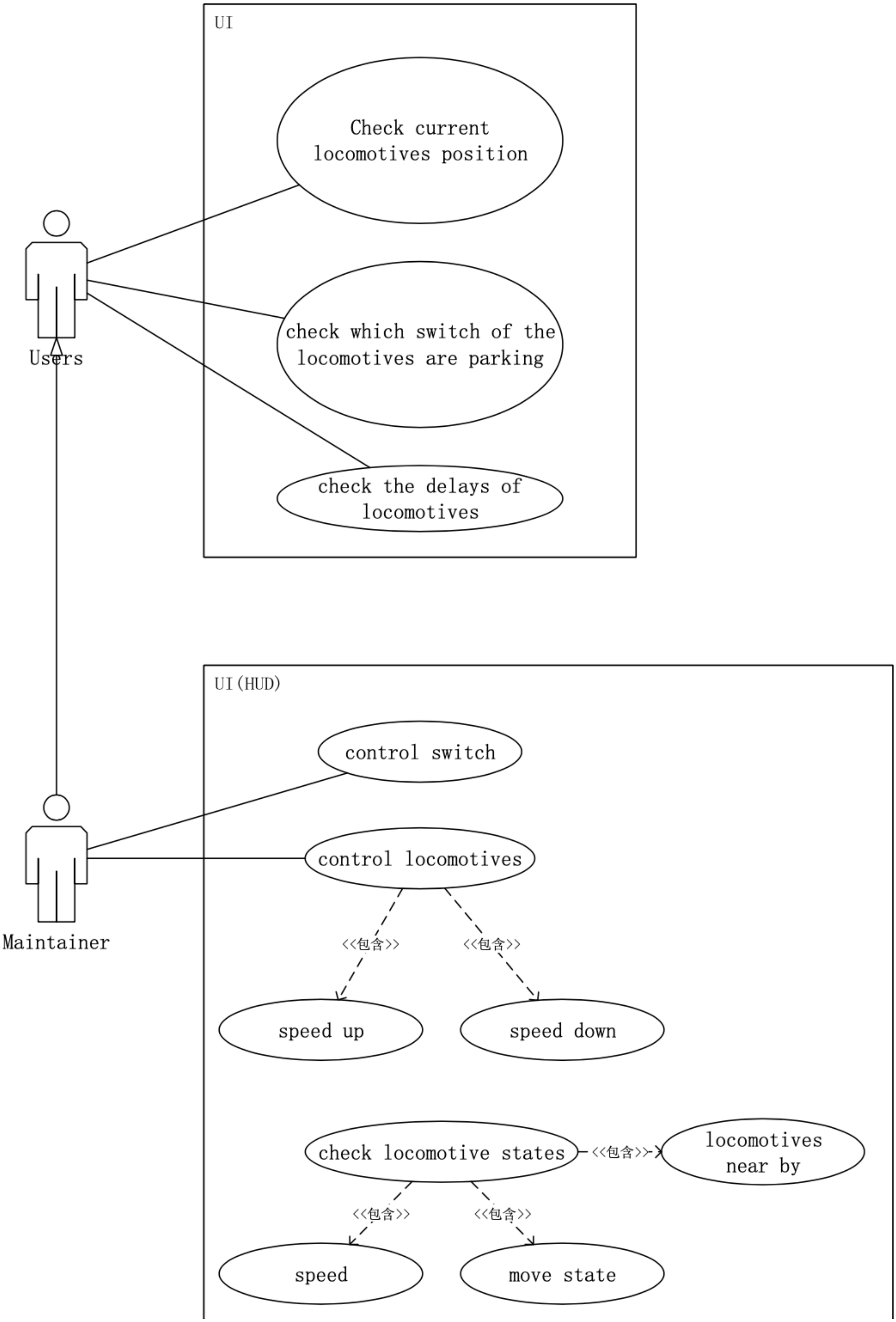


The activity diagram below shows the critical behaviors of controller controlling locomotices and reactions with stations automatively :


Railway Control Center
Maintainer
UI
Controller
station
Locomotive
Start
Give the schedule
Read the schedule and pass to ctrl
compare the departure time and choose which locomotives to leave
switch to the lovomotive which is going to leave
leave the station
compare the speed of locomotives between the same 2 stations
speed down
too fast
Yes
conflict?
too fast or too slow
speed up
No
Yes
switch the switch to a blank rail
the switch already has a train
No
go in to station
Load the movement/states of locomotives and station
calculate the delay
End
阶段

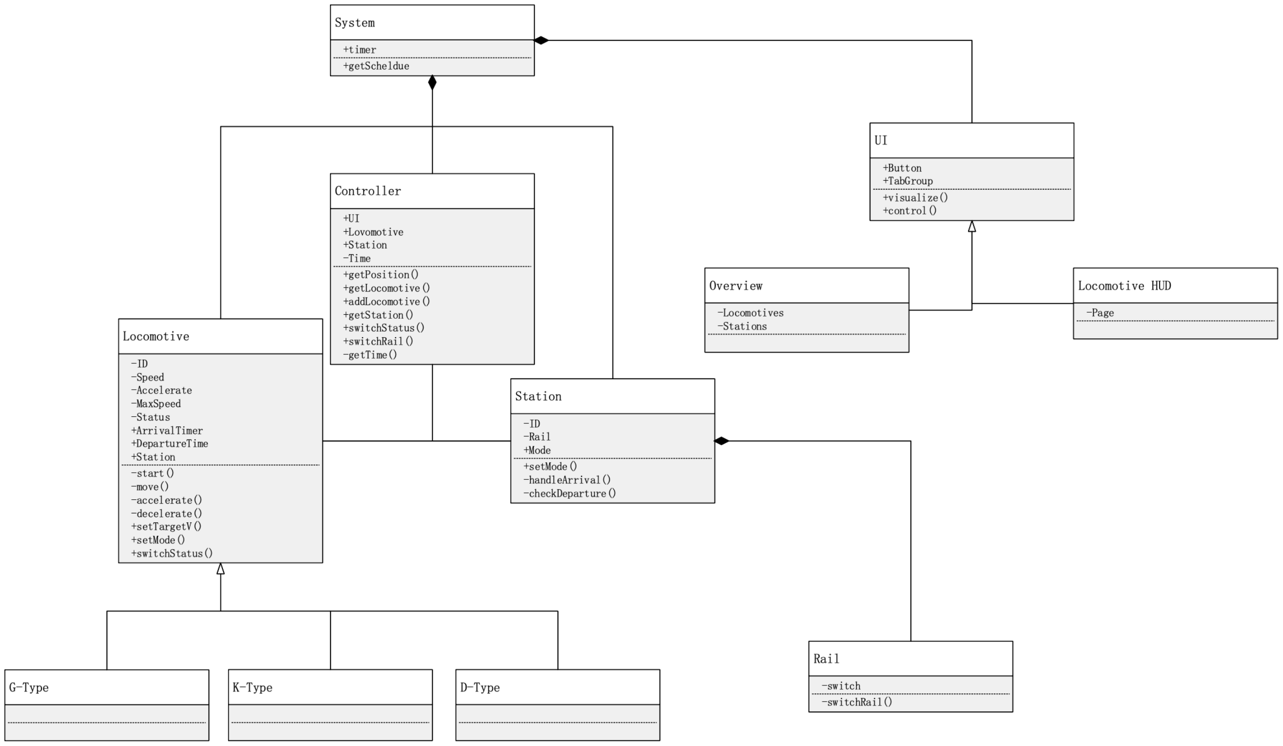
Use Cases

The following use case diagrams show how customers interact with System user interface at the center and how maintainers can control the locomotives by using UI to pass signals.



System Architecture

As the class diagram shown below, the whole system architecture is composed of 4 main parts: UI, locomotives, stations and the central controller.



Besides, we have 3 different types of locomotives which have different property(max speed, accelerate,disaccelerate...), and two different function UI, one is for users to check status and the other is for maintainers to obverse and control these locomotives.

Software Requirements

R1: Overview UI

- R1.1: The locomotives should be visualized:
    - R1.1.1: Users should have access to get locomotives position;
    - R1.1.2: Users should have access to know which switch the locomotives are;
    - R1.1.3: The number of locomotives should loaded on UI;
- R1.2: The Stations should be visualized:
    - R1.2.1: The status in the station(is there any locomotives in which switch) should be displayed;
    - R1.2.2: The number of stations should loaded on UI;
- R1.3: Current time should be displayed;
- R1.4: Schedules can be taken into controller;
- R1.5: Delays should be displayed.

R2: Locomotive HUD UI

- R2.1: Locomotive status should be displayed：
    - R2.1.1: Locomotives' speed;
    - R2.1.2: Locomotives' max speed;
    - R2.1.3: Locomotives' status(speed up/down, constant...);
- R2.2: Maintainer can send signals to controller to realize controlling.
    - R2.2.1: maintainer can switch locomotive status(set max speed/cruising/stop signals...) manually;
    - R2.2.2: maintainer can switch station rail manually;

R3: Controller

- R3.1: Controller should have the ability to get some property:
    - R3.1.1: controller should have access to the time of the system;
    - R3.1.2: controller should have access to the position of locomotives;
    - R3.1.3: controller should take care of the schedule;
    - R3.1.4: controller should have the ability to add locomotives;
    - R3.1.5: controller should have the ability to get stations' situations;
    - R3.1.6: controller should be able to get the nearest locomotives of one another;
- R3.2: Controller should have the ability to control locomotives:
    - R3.2.1: controller should be able to let train speed up/down;
    - R3.2.2: controller should calculate the safe speed of locomotives to make sure there won't be conflict;
    - R3.2.3: controller should let locomotives change their status to make sure there won't be conflict;
    - R3.2.4: controller should let locomotives speed down if it is arrive to a station;
- R3.3: Controller should have the ability to control stations:
    - R3.3.1: controller should be able to switch the switches to make sure that the locomotive can arrive in a station and won't hit another locomotive;
    - R3.3.2: controller should make sure if a train is arrive or has departed;
    - R3.3.3: controller should decide how trains depart to find the least delay(or less than certain delay)according to the schedule;
- R3.4: Controller should update the UI to make sure users can get the information.

R4: Station

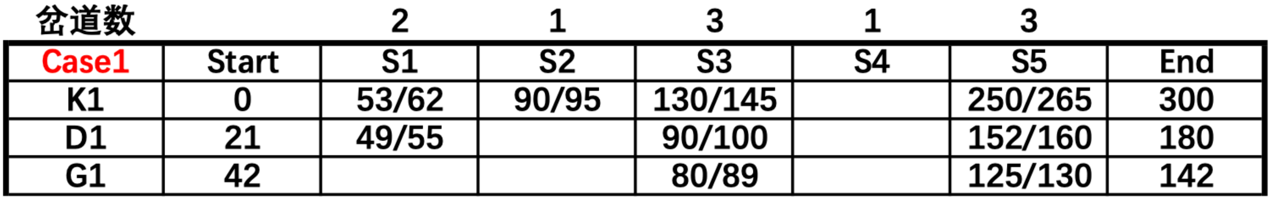
- R4.1: Station should have multi-rails and have switches to change rail.

R5: Locomotive

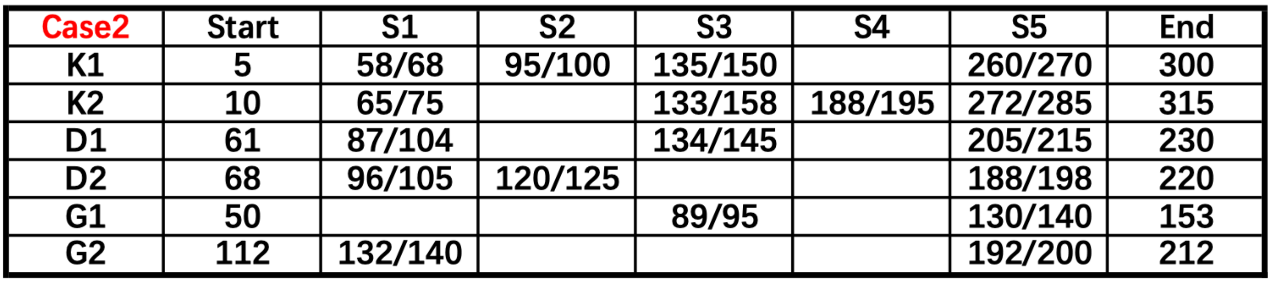
- R5.1: Locomotive should have typical property:
    - R5.1.1: 3 different types of Locomotives have different accelerate,max_speed and curising speed;
    - R5.1.2: locomotives' speed should no largaer than their maximum speed;
    - R5.1.3: locomotives' delay shpuld no larger than the certain delay;
- R5.2: Locomotives can be allowed to have some error about arriving stations(position).

R6: Schedule

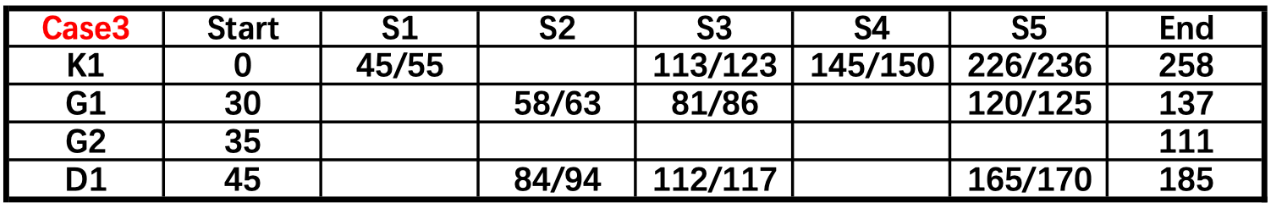
- R6.1: Schedule form will be shown like this:



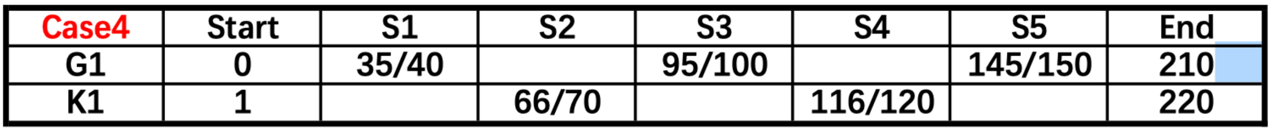
岔道数		2	1	3	1	3	
Case1	Start	S1	S2	S3	S4	S5	End
K1	0	53/62	90/95	130/145		250/265	300
D1	21	49/55		90/100		152/160	180
G1	42			80/89		125/130	142



Case2	Start	S1	S2	S3	S4	S5	End
K1	5	58/68	95/100	135/150		260/270	300
K2	10	65/75		133/158	188/195	272/285	315
D1	61	87/104		134/145		205/215	230
D2	68	96/105	120/125			188/198	220
G1	50			89/95		130/140	153
G2	112	132/140				192/200	212



Case3	Start	S1	S2	S3	S4	S5	End
K1	0	45/55		113/123	145/150	226/236	258
G1	30		58/63	81/86		120/125	137
G2	35						111
D1	45		84/94	112/117		165/170	185



Case4	Start	S1	S2	S3	S4	S5	End
G1	0	35/40		95/100		145/150	210
K1	1		66/70		116/120		220