

지능형열차도착예상정보 시스템을 이용한
열차제어 시스템의 성능향상에 관한 연구
**Study for Enhanced Train Control System with
Intelligent Full Prediction System**

김윤배* 윤희석**
Kim, Yun-Bae Yoon, Ho-Seok

Abstract

Optimization system for convergence point control is required for train control system, this paper introduces the way of enhanced optimization for convergence point with data of intelligence full prediction system. Also the result of the intelligence full prediction system is useful for train control system at the convergence point and passenger will take more accurate information from the prediction system.

서 론

열차제어에 있어 선로상의 교차 지점에 대한 최적화가 반드시 요구되는데, 본 연구에서는 선로의 교차지점의 향상된 최적화를 위하여 열차의 도착 출발 예고 시스템의 데이터를 이용하는 방법에 대한 연구이다. 또한 향상된 지능형 열차 도착 출발 정보시스템은 열차제어 뿐만 아니라 승객에게 보다 정확한 열차 도착 출발 예고 시스템을 제공할 수 있다.

완전 무인으로 동작되는 열차제어 시스템은 열차의 운행 제어에 있어 타임테이블을 이용하여 최적화를 이루고 있으며, 여러 가지 조건에 의해 열차가 지연된 경우 이를 자동으로 회복하는 기능을 갖추고 있다. 그러나 궤도회로를 이용한 고정폐색방식의 신호시스템에서는 대부분 기관사에 의하여 열차가 수동 또는 자동으로 운행되고 있다. 열차의 지연이 발생하고 이를 회복하는 절차로는 운행 시각표를 변경 하거나 열차를 운행하는 기관사의 제어에 의존할 수밖에 없으며 지연된 열차 운행을 회복하기에는 많은 시간이 소요된다. 또한 기관사에 의해 제어되는 열차의 출발 시간과 역과 역사이의 열차 속도는 기관사 또는 사령의 진로 지시에 좌우된다.

* 삼성SDS(주) SOC 개발 2팀 수석, 정회원

** 삼성SDS(주) SOC 2 사업부장 상무, 정회원

이와 같이 열차의 출발 시점과 열차의 속도가 불규칙적으로 제어되는 시스템에서의 열차의 도착 및 출발 정보를 제공하는 승객안내정보 시스템의 정확성에는 많은 변수가 작용하고 있어 정확한 정보를 승객에게 전달하기가 매우 어렵다.

본 연구에서 제시하는 향상된 지능형 열차 출발/도착 예고 시스템은 승객에게 정확한 정보를 전달하는 것 뿐 만 아니라 이 정보를 이용하여 열차제어에 이용하면 보다 향상된 열차제어를 할 수 있어 선로 이용률 증대에 기여할 수 있다.

본 론

1. 기존 승객안내정보 장치(PIS) 의 문제점

승객안내정보 장치(PIS: Prediction Information System)란 승강장에 설치된 표시장치를 이용하여 열차를 이용하는 승객에게 도착할 열차의 예상 도착정보 또는 대기 중인 열차의 예상 출발정보를 문자표시장치를 통해 승객에게 알려주고 또는 방송 시스템을 이용하여 음성정보를 전달하는 장치를 말한다. 설치된 설비에 따라 이용객에게 전달되는 정보의 종류는 다양하지만 가장 기본적인 것은 중간 역에서 다음 열차를 기다리는 승객에게 다음 열차의 예상 도착정보를 전달하는 것이라 할 수 있다. 승객의 입장에서는 다음열차가 정확히 어느 정도의 시간이 경과한 후에 도착할 지를 가장 필요하다고 할 수 있다. 국내의 승객안내정보 시스템이 제공하는 일반적인 정보는 하나 또는 그 이상의 연속되는 열차의 최근 정보를 승강장 상부에 설치된 문자 표시장치를 이용하여 현재 열차의 위치를 기반으로 표시하여주고 있다. 또한 열차가 승강장에 접근하고 있으면 경고음과 함께 열차의 도착을 최종적으로 음성정보를 안내하고 있다. 대부분의 정보는 열차의 도착정보를 시간적인 정보 보다는 위치 기반으로 제공되고 있으므로 승객으로서는 정확한 도착 예정 시간을 알 수 는 없다.

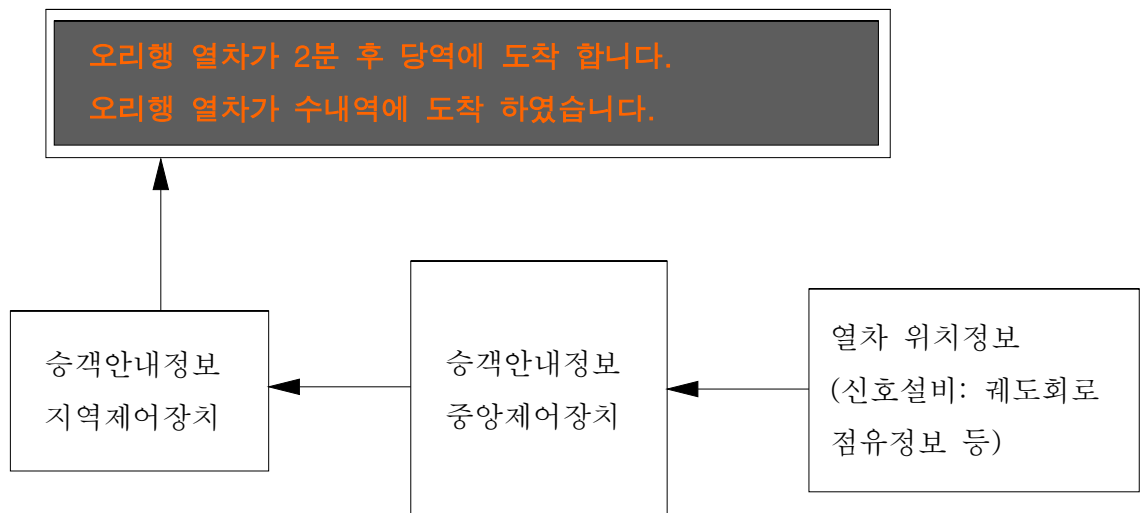


그림 1 승객안내정보 장치 (PIS) 의 개념도

예를 들면, 당 역(또는 열차의 행선지 표시) 도착 열차가 현재 어느 역을 출발 하였는지 또는 어느 역에 도착하였는지 등의 정보를 문자로 표시하여 주며, 표시장치의 구성에 따라 하나 또는 그 이상의 정보를 동시에 표시하고 있다. 대부분의 승객안내정보 장치의 동작 원리는 열차가 특정 지역을 지날 때 그 지점의 위치에 따라 이미 지정된 정보를 승객안내정보 시스템이 그 정보를 필요로 하는 역에 설치된 장치로 전송하고 이를 승강장에 설치된 표시장치에 나타내는 것이다. 이 처럼 도착 예정인 열차의 현재 위치의 변화에 따라 연속적인 정보를 전달하는 것이 아니라 이미 지정된 지점을 통과 해야만 그 정보를 생성하고 이를 이용하여 열차의 도착 예정 정보를 생성할 수 있다. 대부분 승객이 경험했을 수 있는 문제는 이미 열차가 전역을 출발 하였다고 표시한 후에 충분히 예상된 시간이 지남에도 예상된 열차가 승강장에 나타나지 않을 때가 있다. 이는 한 지점의 정보 생성 위치를 지난 후 다음 정보 생성 위치에 도달하기 전에 열차 문제가 있어 이동하지 않은 경우라고 할 수 있다. 이는 승객의 기대를 만족시키지 못하므로 승객에게 불편을 초래할 수 있다.

특히 열차안내정보 시스템이 시간의 정보를 같이 제공한다고 한다면 더욱 부정확한 정보를 제공할 수 있다. 예를 들어 열차가 전역을 출발한 경우 일반적으로 열차의 이동 시간을 계산하여 1분 후에 열차의 도착정보를 표시한 경우, 승객은 그 정보를 본 후 열차의 도착 시간을 예상할 수 있다. 그러나 앞에서 언급한 것처럼 예상치 못한 상황이 발생하여 열차의 전 역 출발정보를 표시한 후 열차가 이동할 수 없거나 선로의 여러 가지 조건에 의하여 정상 속도로 운행이 불가능한 경우 실제 열차의 도착시간은 계속 지연 될 것이다. 그러나 시간이 지남에 따라 이전에 발생한 정보가 계속 같이 시간을 표시하고 있다면 승객은 그 정보를 신뢰할 수 없을 것이다. 기존 시스템의 문제점을 정리하면 아래와 같다.

- 문제1: 만약 열차가 한 Prediction Trigger Point 를 지난 후 다음 Prediction Trigger Point 를 예상된 시간 안에 도착하지 못한 경우 예상도착시각은 그 열차가 다음 Prediction Trigger Point 에 도착할 때까지 계속해서 이전 데이터를 표시하게 된다.
- 문제2: 기본기능 설계시 Timetable 을 고려하지 않았기 때문에 열차가 어느 한 지점의 Timing Regulation Location 에 일찍 도착한 경우 그 열차는 열차시간표에 의해 출발시간까지 그 지점에 대기하고 있어야 한다. 그러므로 실제 열차가 출발하여 다음 Prediction Trigger Point 에 도달할 때까지 계속해서 이전 데이터를 표시하게 된다.
- 문제3: 도착시각과 출발시각의 예상시간 계산하는 Module 이 분리되어 있어 열차가 회차 역 부근에서는 반대방향의 예상도착시간을 계산할 수 없다. 왜냐하면 그 열차가 언제 반대방향으로 회차하여 출발 할지 모르기 때문이다.

위에서 언급한 내용은 열차의 도착정보에 대한 것으로 열차의 출발 정보는 다르다고

할 수 있다.

■ **예상출발시각 (Departure Prediction Data)의 문제점:** 예상출발시각은 회차 역에서 열차가 회차 한 후 출발하기 전 승객에게 출발 전까지 남은 시간을 알려 주는데 ART가 이를 계산하여야 하는데 앞에서 말한 것과 같이 ART가 회차 역 주변의 모든 관련된 데이터를 고려하여 언제나 한가지 경우라도 변경된 경우 다시 계산하여야 하므로 상황에 따라서는 계산에 필요한 Load가 클 수 있으며 출발 역의 선행열차 상황을 고려하지 않았기 때문에 선행열차가 멈추어 있거나 지연된 경우 예상출발시간이 맞지 않는 경우가 많이 발생하였다.

2. 향상된 지능형 승객안내정보 시스템

기존 시스템에서 갖고 있는 문제점을 보완하고 열차의 실제 지연에 따라 승객안내정보 시스템도 실제 열차의 지연정보를 정확하게 반영하여 표시하여 줌으로써 열차의 예상 도착 시각을 승객에게 알려줄 수 있다. 또한 이러한 정보를 이용하여 열차제어에 사용하면 교차지점의 경합 알고리즘을 향상시켜 선로이용률을 향상시켜 줄 수 있다.

2.1 향상된 지능형 승객안내정보 시스템의 특징

향상된 설계의 특성으로는 예상 도착시각과 출발시각을 분리하지 않고 한 Module 이 담당하였고, 외부시스템인 PIS 을 위한 Prediction Data 뿐만 아니라 Central Control Center 즉 사령실의 운영자가 화면에서 필요한 위치에 열차의 예상 출발과 도착시간과 열차의 순서를 표를 이용하여 한눈에 보여 주고, Timetable 을 고려하여 Depot 에서 열차의 예상 진입시각과 예상 진출시각을 계산하고, 또한 연결된 다른 라인에서 진입할 열차와 진출할 열차까지 포함한 Prediction Data를 실제열차가 Timetable 데이터와 비교하여 얼마나 편차를 갖고 운전이 되는가를 나타내는 Logging Data를 기록하는 Train Lateness 도 함께 Prediction System 에 포함시켰다.

- PIS Prediction Data - 승객을 위한 예상열차운행정보
- HMI Prediction Data - 사령을 위한 예상열차운행정보
- Depot Prediction Data - Depot 관리자를 위한 예상열차운행정보
- MLCC Prediction Data - 기존라인과 신규라인을 위한 예상열차운행정보
- Train Lateness Data - 유지보수를 위한 실제열차운행과 시간표와의 편차기록

위와 같은 다섯 종류의 데이터를 생성해야하는 새로운 Intelligent Full Prediction System은 생성하는 모듈을 추가하고 기존에 사용하던 Prediction System은 삭제한다.

새로운 설계의 개념은 매 15초마다 전체라인에 운행하고 있는 열차들의 현 위치를 Snapshot 하고Timetable 에서 Depot와 연결선에서 진입할 열차의 정보를 읽어 모든 가상의 열차들을 향후 20분까지 가상 시간범위에서 모든 가상열차를 시뮬레이션을 통하여

시뮬레이션 시간 동안 가상열차들의 Prediction Point 를 지날 때 시간을 이용하여 Prediction Data 를 생성하여 시뮬레이션 종료 후 생성된 Prediction Data 를 TSS 또는 다른 Module 들이 읽을 수 있는 형태로 변환하여 저장한다. 물론 이론적으로는 가능하지만 실제로 구현하기 위해서는 고려해야 할 것들이 상당히 많았다.

- 가상공간에서의 열차를 시뮬레이션하기 위하여 모든 Track Circuit의 평균열차의 이동시간이 필요하다. (현장에서 2개월간 기록하여 평균 통과시간 만듦)
- 중앙제어장치의 화면에 데이터를 표시할 윈도우가 필요.
- 시뮬레이션을 위한 새로운 Route, Signal, Trip Section 등 변경 또는 추가할 Customer Data가 필요하였다.

2.2 향상된 지능형 승객안내정보 시스템의 구현

실행시간을 최소화 하기위해 Prediction Generation 에 필요한 모든 데이터를 Local Memory 로 복사를 하고 시뮬레이션 동안에는 데이터를 Local Memory 영역만을 사용하여 읽고 써서 Prediction Data를 생성한다. 시뮬레이션을 마친 후 Local Memory 영역에 생성된 Prediction Data를 전체 시스템이 사용할 수 있는 공용데이터 영역으로 DMLAPI Function 을 사용하여 복사를 함으로서 처리속도를 현격히 줄일 수 있다.

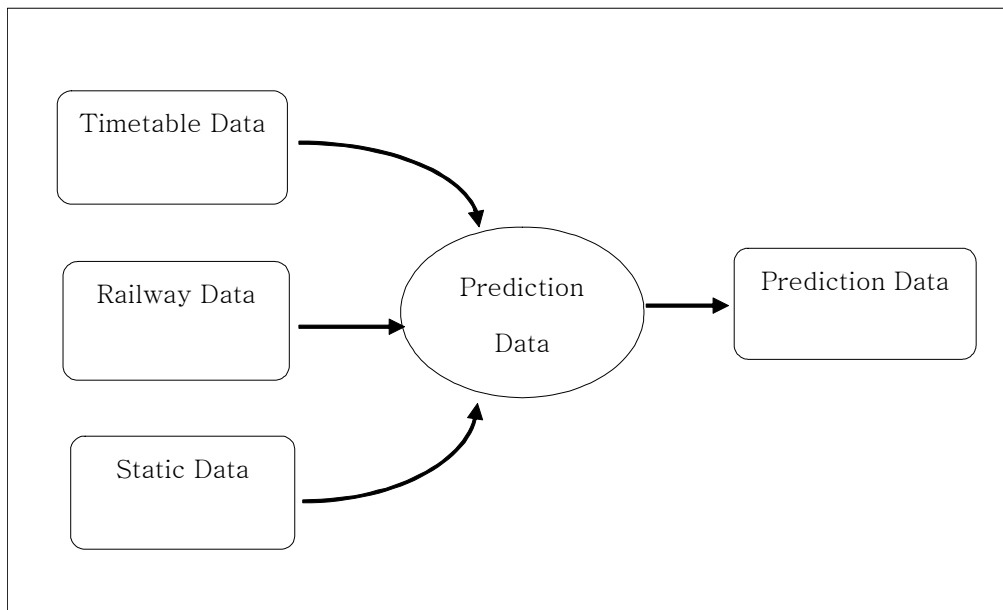


그림 2 시스템 구성 개념도

Function 을 사용하여 복사를 함으로서 처리속도를 현격히 줄일 수 있다.

위의 [그림 4]와 같이 Prediction Data Generation System 은 Timetable Data, Railway Data, 그리고 Static Data 를 바탕으로 Prediction Data Generation System에서 주어진 시간 동안 실제와 같은 열차운행 Simulation 을 거쳐 Prediction Data 를 생성한 후 각각 필요한 공유데이터 영역에 저장한다.

LIC(Local Interface Controller) Processor 에 위치한 GPR(Generate Prediction Module) 이 기동할 때 먼저 Timetable Data, Railway Data 그리고 Static Data Class의 Object 를 생성하여 초기에 필요한 모든 데이터를 DMLAPI 를 이용하여 읽어 Local Memory 영역에 복사를 한 후 Simulation 을 하지 않는 시간에 Dynamic 하게 공유데이터의 변경된 내용만 Local Memory 에 업데이트 한다.

Prediction Data 를 생성하기 위한 데이터를 아래에서 예를 보여 주었지만 각각의 데이터 내용의 설명은 여기서 생략하기로 한다.

Timetable Data

- Trip Header Data
- Trip Element Data
- Train Order List
- Train Depot Entry/Exit List
- Train MLCC Entry/Exit List

Static Data

- Trip Section Data
- Prediction Route Data
- Prediction Signal Data
- Prediction Track Circuit Data
- Regulation Location Data
- Reversal Location Data
- Route Track Circuit Mapping Data
- Trip Section Route Mapping Data

Railway Data

- Train Model Data
- Convergence Point Data
- Route Status Data
- Layover Status Data

2.3 향상된 시스템 데이터 정확성

완성된 시스템의 데이터는 시뮬레이션을 통한 모든 Timetable Data, 모든 열차의 현재 위치, 그리고 가장 최근의 철도의 신호상태를 적용하여 열차의 실제 운행과 거의 유사하게 구현하여 얻기 때문에 Data 의 오류가 적고 승객안내정보 시스템으로(PIS) 전송할 뿐만 아니라 사령실 요원에게 까지 실시간으로 모든 중요 Timing Regulation Point 와 Convergence Point 의 실제 열차 통과 순위를 향후 20분 까지 정확하게 파악할 수 있으며, Depot 및 연결선에서의 진입열차 정보까지 한 눈에 볼 수 있어 전체 철도의 운영을 원활히 하는데 도움을 주고 또한 Timetable Data 를 수정하는데 있어서 향후 20분 까지 열차의 위치를 예측할 수 있어 정확성을 기할 수 있다. 또한 실제 열차의 운행 기록을 실시간으로 Logging System 에 기록하고 함께 Prediction System 이 생성한 Prediction

Data를 기록하기 때문에 향후 실제 열차의 운행과 Prediction System 생성한 Data를 비교할 수 있고, 비교한 결과 Intelligent Full Prediction System 의 정확성의 95% 이상의 정확성을 나타내고 있다.

2.4 속도향상 및 데이터 저장

Prediction System 을 완성한 후 성능시험을 하였다. 최대 운행 중인 열차의 수가 25편성 Train Depot 에서 진입할 열차가 5편성 연결선 에서 진입할 열차가 5편성인 경우 Simulation Execution Time 은 평균 200ms 에서 450ms 이고 생성한 Prediction Data를 공유데이터 영역에 입력하는 시간은 50ms 에서 500ms 정도로 측정되었다. Simulation 시간은 Simulation 할 가상 열차의 수와 비례하지만 Predication Data를 업데이트 하는 시간은 처음에 데이터를 만들 때는 많은 시간이 걸리지만 이미 만들어진 데이터의 값을 바꿀 때는 시간이 절약된다. 그러므로 열차가 처음 생성되면 그 열차가 향후 20분 동안 통과할 Prediction Point 만큼 새로운 Record 를 생성하기 때문이다. 그러나 전체적인 Performance 는 기존의 Prediction 보다 50% 이상 속도가 향상된 것으로 조사 되었다. 비록 생성할 데이터도 많아지고 정확도도 향상되었지만 새로운 설계와 초기의 문제점 발견 그리고 조속한 해결로 얻은 결과였다. 매 Prediction Cycle 마다 생성되는 모든 Prediction Data는 System Event Logging Storage 에 저장되어 진다.

2.5 향상된 시스템의 응용

새로운 Prediction System 설계 및 구현을 통하여 철도신호제어 시스템에 사용되는 모든 데이터를 깊이 이해하는데 도움이 되었고 기존 시스템에서 갖고 있던 문제를 해결하기 위한 방법으로 Local Memory를 사용하였다. 이때 처음으로 모든 소프트웨어 제작을 설계 시부터 Object Oriented 방법론과 Class 를 구축하고 C 보다는 C++로 구현하여 시스템 Debugging 과정과 Implementation 그리고 현장의 문제를 해결할 때 실행 시간을 단축할 수 있었다. 이러한 방법으로 설계 때부터 예상되는 문제를 발견하고 조기에 해결한다면 상당한 시간 단축을 할 수 있다.

3. 맺음말

시스템의 설계단계에서 예상되는 문제를 조기에 발견하여 조치하면 최종단계에서 발견한 경우에 비하여 많은 시간을 줄일 수 있다는 것을 Intelligent Full Prediction System Project 를 통해 경험하였다. 또한 문제의 발견도 중요하지만 이를 해결하기 위한 신속히 Risk Management 조직을 만들어 각 부문의 전문가의 의견을 합리적으로 수렴할 수 있는 유연한 대응이 필요하다. 철도신호제어 시스템 성격상 Prediction 시스템 전체의 복잡성과 전문성 때문에 설명이 부족한 부분도 있다고 생각한다.