

철도청 사령실통합 신호설비 구축 프로젝트에서의 경합검지의 설계

Railway Conflict Detection Design for KORAIL Command Center Intergration Signal Equipment Construction Project

이주왕*, 김범식*, 문영현**, 홍효식***, 유광균***
Lee Ju-Wang*, Kim Bum-Sik*, Moon Young-Hyun**,
Hong Hyo-Sik***, Yoo Kwang-Kyun***

ABSTRACT

현재 사용되고 있는 열차열차운행관리 시스템(Railway Traffic Management System, RTMS)에서 경합 검지 및 해소업무는 역 및 선로의 정보(상태)연계 및 자동화의 미비로 인하여 착발선 결정, 착발시간 결정, 교행역, 대피역등에서 발생하는 경합 검지 및 해소(Railway Conflict Detection and Resolution)의 결정이 신속하게 이루어지지 못하고 있다.

본 논문은 국내외적으로 적용하고 있는 열차 경합 검지 방법을 조사 비교하여 가장 적절한 알고리즘을 개발, 구현하여 현재 진행중인 철도청 사령실 통합 신호설비 구축사업 프로젝트에 적용하고자 한다.

1. 서론

경제 성장에 따라 여객과 화물의 수송량이 급증하면서 대표적인 수송수단으로써 철도 수송의 중요도는 더해지고 있다. 철도는 안전성 및 신뢰성 확보가 기본적인 과제이다.

현재 사용되고 있는 열차운행관리 시스템(Railway Traffic Management System, RTMS)에서 경합 검지 및 해소업무는 역 및 선로의 정보(상태)연계 및 자동화의 미비로 인하여 착발선 결정, 착발시간 결정, 교행역, 대피역등에서 발생하는 경합 검지 및 해소(Railway Conflict Detection and Resolution)의 결정이 신속하게 이루어지지 못하고 있다. 또한 경합 해소에 있어서 비용(cost)적 측면, 여객열차/화물열차의 우선 요인에 따른 객관적인 판단과 같은 표준적인 요소를 적용하는데 어려움이 있다. 이로 인해 다수의 전문가들이 열차 경합 검지 및 해소 시스템(Railway Conflict Detection and Resolution System, RCDRS)를 하기위해 많은 시간과 노력 감수해야만 하며, 전문가의 실수로 인한 오류가 발생할 가능성도 있다. 따라서 현업무의 이해를 바탕으로 도

* :연세대학교, 전기전자공학부 석사과정, 학생회원

** :연세대학교, 전기전자공학부 교수, 정회원

*** :한국철도대학, 교수, 정회원

출된 개선방향의 정의를 통하여 경합 검지 및 예측 개발 방안을 제시하고자 하였다.

본 논문의 구성은 다음과 같다. 먼저 2장 본론은 열차 경합에 대해 설명하였고, 3장에서는 경합 검지 및 예측 방안을 제시하고, 4장에는 3장에서 논의된 방안으로 적절한 알고리즘을 제시하고 마지막으로 5장에서 결론을 도출하였다.

2. 본론

현행 시스템은 서울, 대전, 부산, 순천 그리고 영주 등으로 총 5개 지역본부로 분산되어 있어 업무의 중복을 줄이고, 자동화(Automation)된 열차집중제어장치(Central Traffic Control, CTC)를 만들고자 지역본부를 대전으로 통합하는 프로젝트를 실행하고 있다. 이에 따르는 열차 경합 검지 및 해소 소프트웨어를 개선하는 프로젝트 또한 포함하고 있다.

2.1 열차 경합의 정의 및 종류

열차경합(Train Conflict)이라는 것은 열차와 열차간의 충돌 또는 충돌위험으로 정의할 수 있다. 즉 두 대 이상의 열차가 미래의 같은 시각에 같은 선로를 점유하고자 하는 것을 열차경합이라고 할 수 있다. 이러한 열차 경합은 열차들이 사전에 계획된 스케줄에 따라 운행하도록 되어 있으나 전국에 걸친 철도 네트워크에서는 열차 스케줄 작성 당시에 고려하지 못하는 이해적인 상황이 발생하게 된다. 이로 인하여 열차 운행에 영향을 미치게 되고 이는 열차 지연이 이어져 열차경합을 발생하게 한다.

이런 열차경합의 종류로는 크게 5가지로 구분을 하였으며, 아래 표1. 경합의 종류 및 원인으로 나타내었고, 표2. 열차의 지연을 발생시키는 원인을 설명하였다.

표1. 경합의 종류 및 원인

| 예상 경합 | 경합 종류 | 경합 원인 |
|-------|--------|--|
| 충돌 | 열차추월경합 | 노선에서 선행열차의 불필요한 감속 및 정차 등으로 인한 지연 |
| | 열차교행경합 | 단선에서 열차의 지연으로 계획된 교행이 사전 계획에 따라 발생하지 않은 경우 |
| 충돌위험 | 열차경로경합 | 열차의 지연으로 두열차가 동일 도착선 요구할 경우 |
| | 열차수렴경합 | 두 개 이상의 노선으로부터 열차의 합류가 사전계획에 따라 일어나지 않는 경우 |
| | 운전시격경합 | 선행 열차의 불필요한 감속 및 정차로 인한 지연 |

표2. 열차 지연의 원인

| 종류 | 원인 |
|------|---------------------------------|
| 선로점유 | -선행 열차 이상 -사고에 의한 선로 일시 사용중지 |

| | |
|----------|---|
| | -현시각열차의 열차 점유(구원열차, 모터카등) |
| 계획이탈 | -교행변경, 대피, 착발선 변경 연쇄지연에 따른 열차지연 -승하차지연, 환산초과, 소화물 적하, 조성입환 지연 -입시열차 투입에 따른 선로 트래픽증가 |
| 물리적 제한조건 | -동력차, 발전차, 객화차 이상에 따른 지연 -선로장애 또는 전차선 송전 장애에 따른 지연 -신호 장치 고장 |

2.2 경합 검지 및 해소 시스템(RCDRS)의 개요

경합 검지 및 해소시스템(RCDRS)은 트래픽 관리시스템의 일부로 시스템은 현용되는 운전정리 방식의 자동화/표준화하는 1단계와 경합의 사전 예측을 통한 해소책을 제시하는 2단계로 구분하여 단계별 구축을 하게 된다. 시스템은 운전정리 시스템과 경합검지/예측 및 해소를 담당하는 기능으로 구성되며 경합요인의 검지를 위하여 내부 시스템인 자동진로설정, 열차스케줄 작성시스템 등과 연계되고, CTC 이외의 시스템과 인터페이스 기능을 수행하도록 되어있다.

또한 경합검지 및 해소 시스템(RCDRS)은 열차간의 경합 발생을 사전에 예측하고 이를 해소하기 위한 객관적인 판단근거가 되는 비용(cost)요소를 도출하여 최적의 해소방안을 사용자에게 제시한다. 시스템 기능으로는 경합예측, 잠재적 경합설정, 최적화 방안(목적식 관리, 비용산정 및 우선순위 부여), 경합해소 방안을 제시하는 시스템으로 구축한다.

그림 1은 이러한 경합 검지 및 해소 시스템의 내·외부 인터페이스 관계를 이해하기 쉽게 표현한 것이다.

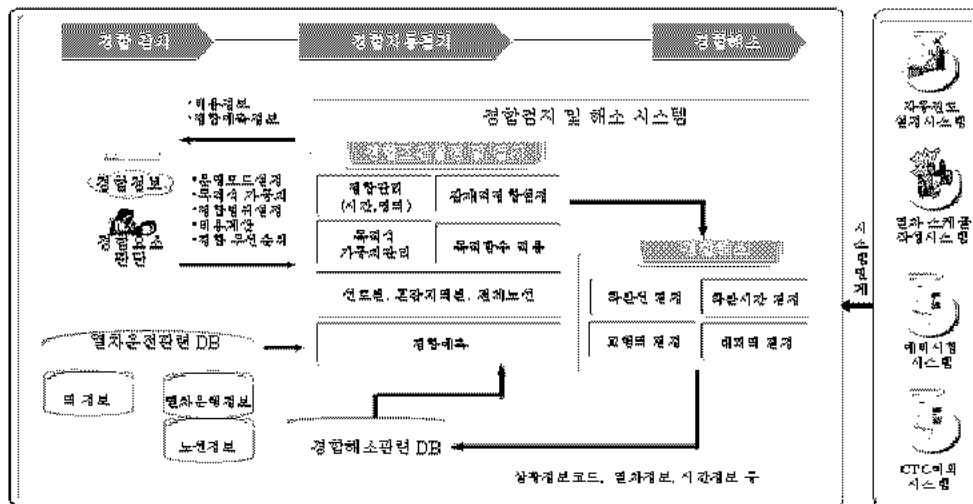


그림 1. 경합 검지 및 해소 시스템(RCDRS)

3. 경합 검지 및 예측 방안

운행열차의 경합 상황을 예측하기 위하여 경합의 범위와 잠재적인 경합을 정의하고, 시스템은 이를 기반으로 경합을 예측하도록 구축한다. 사령은 열차그래프의 지속적인 감시를 통하여 운전상황을 감시하며 지연상황 및 이례상황 시에 현재 데이터를 중심으로 예측 열차그래프의 전개를 통하여 경합을 예측한다.

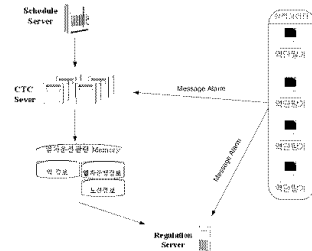


그림2. 실시간(Real-Time) 업데이트 방법

그림2 에서 보는바와 같이 최소 스케줄 서버에서 열차 운용 스케줄을 작성한다. 이렇게 작성된 스케줄은 CTC 서버에 DB로 구축이 되고, 구축된 데이터들은 상호 인터페이스가 있는 다른 서버들과의 접속을 자유롭게 하기 위하여 열차운전관련 메모리로 만들어 지게 된다. 이렇게 만들어진 데이터들은 각 역의 단말기들을 통해 당일 운행되고 있는 열차들에 대해 실시간(Real-Time)으로 CTC 서버에 정보가 업데이트되는 동시에 열차운전관련 메모리로 만들어지게 되고, 동시에 열차 정합 검지 서버에도 알려지게 된다.

그러면 정합 검지 서버는 열차가 계획된 스케줄대로 운행하고 있는지 만약 지연이 발생하였다면 열차의 지연으로 인하여 다른 열차와 정합이 발생할지에 대한 조건을 조사하게 되고 정합 발생 조건을 충족시켰을 경우, 알고리즘을 실행하여 정합이 발생되는지 검지 및 예측을 하게 된다.

여기서 열차운전관련 메모리에 만들어진 데이터들은 필요한 정보를 더욱 빠르게 검색할 수 있도록 도와준다.

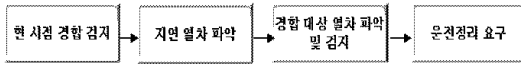


그림3. 열차 정합 흐름도

열차운전관련 메모리에 열차운행정보, 역정보 등이 존재하며 그 클래스 속에는 다양한 변수들이 존재를 하게 된다. 표 3은 그 클래스 속에 존재하는 변수들을 정리하였다.

표3. 열차운전관련 메모리의 변수

| 구 조 | 내 용 |
|--------|---|
| 열차운행정보 | 운행일자, 선코드, 열차번호, 이전 열차번호, 다음 열차번호, 노선코드, 방향, 열차종별, 분할함명코드 |
| 역정보 | 열차번호, 순서, 역코드, 속도종별코드, 정차구분코드, 플랫폼 번호, 도착시각,도착진로코드, 출발시각, 출발진로코드, 실적 도착시각, 실적도착진로코드, 실적 출발시각, 실적 출발진로코드 |

표 3과 같은 데이터들이 실시간(Real-Time)으로 각 역에서 출발, 도착시간이 업데이트 되고, 이는 자동적인 열차경합 검지(Automatic Conflict Detection)의 기초자료가 된다. 최종적으로 경합상황의 최적 해결을 위해서는 경합이 시작되는 장소와 시점을 적절하게 인식, 즉, 시간-거리(Time-Distance)그래프 상에서 발생 가능성이 있는 경합의 위치와 시점을 미리 파악하는 것이다.

정규/실제 열차시간 비교 결과가 각각의 역에 설정되어 있는 여유시간(Tolerance)내에 있다면, 예측 열차선(Forecasting)은 정규 열차선(Nominal Line)을 평행 이동함으로써 실제 차선(Actual Line)에 첨부된다. 이때 열차시각표에 따른 최소 정차시간과 출발시간을 준수한다는 가정이 따라야 된다.

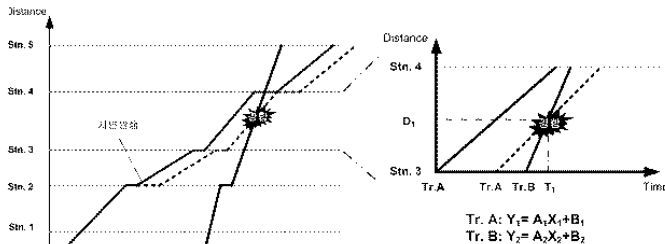


그림4. 열차 DIA

위의 가정을 만족하였을 경우 그림4와 같은 지연이 발생된 열차와 스케줄에 맞게 운행되는 두 대의 열차 그래프를 그릴 수 있게 된다. 본 그림은 열차의 출발시간의 지연으로 발생된 경합을 나타내었고, 각 역마다의 열차 도착시간과 출발시간을 알 수 있으므로 해서 후속 열차와의 경합을 비교시 후속열차의 도착시간이 지연발생된 열차의 도착시간보다 빠르게 도착하면 경합이 발생하는 것으로 간주하였다. 이렇게 후속열차가 먼저 도착을 하게 되면 그 역에서 각각의 열차에 대해 1차방정식을 세우게 되고, 각 열차에 대한 방정식에서 교차점을 찾게 된다. 이 교차점이 경합이 발생하는 시간(T_1)과 거리(D_1)가 되는 것이다.

4. 경합 검지 알고리즘

많은 계산적인 일의 기본적인 연산은 검색(Searching)이다. 즉 이전에 저장된 많은 정보의 양에서 어떤 특정한 부분의 정보를 검색하는 것이다.

경합 검지에 있어도 가장 중요한 것은 빠른 시간 안에 경합이 발생할 수 있는 열차를 검색하는 것일 것이다.

그림5는 경합검지의 전체적인 흐름도를 나타내는 것이다.

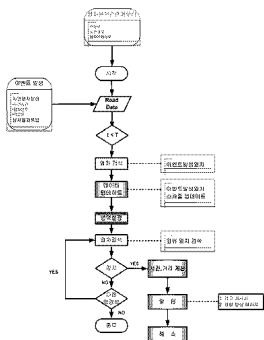


그림5. 결합 검색 흐름도

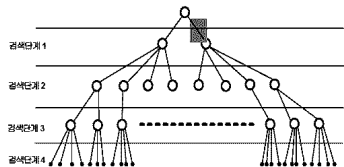


그림6. 검색 트리 구조

그림5의 흐름도는 그림2에서 실시간으로 업데이트 되는 데이터를 수신하여 열차를 검색하고 그 열차를 중심으로 앞뒤열차를 검색, 이 열차들과 시공간적인 범위내에서 충돌이 발생되는지 검지 하고, 결합 검지서 해소 알고리즘에 데이터 전송 및 열차 사령에게 알리게 된다. 여기서 검지의 중요단계인 열차 검색단계부분의 구조는 그림6과 같이 하향식 트리 구조로 구성했으며 각 단계 별 조건을 설정하여 검색을 빠르게 할 수 있도록 해야 한다.

5. 결론 및 향후 연구과제

본 논문의 철도 결합은 결합 검지로만 끝나는 것이 아니라 이를 통하여 그 다음 개발 단계인 결합해소로 문제해결이 이루어진다. 이를 위해서는 자동 결합 해소의 시간적, 공간적 범위를 어느 정도 할 것인가를 고려할 필요가 있다.

지금 현재 진행되고 있는 철도청 사령실용화 실험실비 구축 프로젝트에서 사용되는 운영환경은 결합 검지 및 해소 서버는 Alpha Server DS20, Alpha Server ES40을 사용하며, OS는 UNIX급인 TRUE64사용하고 개발언어로는 ANSI C를 사용하기로 되어있다.

그러나 실험은 Windows환경에서 Visual C/C++로 구축하여 실험하여 개발환경이 다르다는 것이다. 지금 현재 본 알고리즘을 UNIX환경으로 전환하고 있는 과정중이다.

참고문헌

1. Rok Sosic and Jun Gu, (1994) "Efficient Local Search with Conflict Minimization: A Case Study of the n-Queens Problem", IEEE TRANSACTION ON KNOWLEDGE AND DATA ENGINEERING, vol 6, no 5, October 1994
2. V. Salim and X. Cai, (1995), "Scheduling Cargo Trains using Genetic Algorithms", IEEE, p224-227
3. Te-Wei Chiang and Hai-Yen Hau, (1995), "Railway Scheduling System Using

Repair-based Approach", IEEE, p71-78

4. 오석문, (2003), "최적화 해법을 이용한 열차경합 해소와 한국철도 적용방안", 한국철도학회 학술대회논문집
5. 노학래, "열차 모니터링 시스템에서 열차경합 자동검지 개요",
6. 유영훈, 황종규, 조근식, "열차 운용 스케줄링 전문가시스템의 설계 및 구현"
7. ROBERT SEDGEWICK 著, 민용식, 오삼권 共譯, (1997), "C++ 알고리즘", 대영사
8. 이재규 著, (2001), "C로 배우는 알고리즘", 세화출판사