이벤트 발생에 따른 점수 계산을 통한 CAN 신호 탐색 기법

안종화, 손명환, 박부식, 장수현
한국전자기술연구원

ajw0305@keti.re.kr, sonmh221@gmail.com, pusik.park@keti.re.kr, shjang@keti.re.kr

CAN signal search technique by score calculation according to event occurrence

Ahn Jong Wha, Son Myeong Hwan, Park Pusik, Jang Soo Hyeon
Korea Electronics Technology Institute.

요 약

차량에 사용되는 수많은 전자제어장치들은 CAN(Controller Area Network) 버스를 사용해 서로 통신하고 있다. 장치들 사이에 주고받는 CAN 메시지에는 차량의 상태와 주행정보 등이 포함되어 있기 때문에 이를 분석해 사용한다면 사용자에게 유용한 서비스를 제공할 수 있다. 자동차 관련 연구활동이 활발해지는 만큼 CAN 데이터 분석의 필요성은 높아지고 있다. 이 논문은 수많은 CAN 데이터 중 원하는 특정 신호를 탐색하기 위한 이벤트 발생과 데이터 변화를 이용한 신호 탐색 방법을 제안하고 구현을 통한 검증 결과를 제시한다. 이 방법은 해당 이벤트 발생 시간에 변화하는 CAN 메시지를 찾아 바이트 단위와 비트 단위로 점수를 가감해 이벤트와의 관련성을 예측한다. 실험 결과 이벤트 발생 후 1-2 분 내에 원하는 단일 신호를 비트 단위로 정확하게 발견하였고 기존의 방법보다 우수한 성능을 확인하였다.

Ⅰ. 서 론

CAN 버스를 통해 차량의 상태와 주행정보 등의 데이터가 이동하기 때문에 CAN 메시지를 분석해 활용한다면 운전자에게 유용한 서비스를 제공할 수 있을 것이다. 그러나, 접근이 쉽다는 점 덕분에 CAN 버스를 통과하는 프레임을 쉽게 얻을 수 있을지 몰라도 CAN 메세지는 제조사, 차량의 모델에 따라 다르고 그 형식은 비밀로 관리되고 있어 차량용 CAN 데이터를 이용한 서비스를 개발하는데 장애가 있다. 따라서 CAN 메시지를 활용한 서비스를 제공하기 위해서는 필요한 CAN 메시지를 차량에 따라 빠르게 분석, 발견할 필요가 있다.

하지만, 캡처 된 수많은 CAN 메시지 중 원하는 신호를 발견하는 과정은 시간과 노력이 많이 소요된다. 이러한 문제를 해결하기 위해 분석 시간을 단축시키고 자동화 하기위한 연구가 진행되고 있으며[1][2][3][4] 우리는 앞선 연구들과 달리 필요한 특정 신호를 발견하는 시간을 단축시키기 위해 이벤트 발생에 따른 값의 변화를 분석해 신호를 분석하는 방식을 사용했다.

이어지는 장에서는 적용한 방식에 대한 상세 설명과 해당 방법을 사용해 진행한 테스트 과정과 결과를 제시할 것이다. 이어서 본 논문의 결론을 정리할 것이다.

Ⅱ. 이벤트 시점 기반의 점수제 방식

이 장에서는 이벤트를 입력하고 이벤트가 발생한 시점에서의 신호 값들의 변화를 관찰해 원하는 신호를 찾는 방법을 설명한다. 즉, 이벤트가 발생한 기간 또는 시점에서 비트 데이터의 변화에 따라 플러스 점수를 주고 이벤트와 관련 없는 시점에 발생한 비트 데이터의 변화는 마이너스 점수를 주어 결과적으로 시간이 흐른 뒤 이벤트와 관련된 메시지만 플러스 점수를 유지해 해당 신호를 발견하는 방법이다.

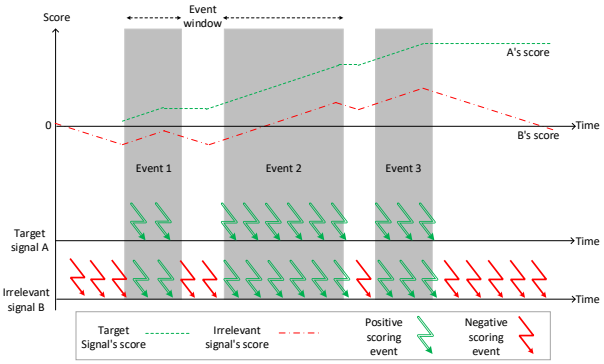

Figure 1 점수 계산 예

해당 방식에 대한 예시를 Figure 1 에서 확인할 수 있다. Figure 1 에서 A 는 찾고자 하는 신호, B 는 이벤트와 관계없이 데이터가 변하는 신호다. 신호 A 와 B 는 주기적으로 수신되고 있으나 신호 A 의 경우 세 개의 이벤트 영역에서만 데이터가 변화하여 Negative scoring event 가 없이 Positive scoring event 만 존재한다. 하지만, 신호 B 의 경우 데이터가 계속해서 변화하기 때문에 이벤트 구간에서는 Positive scoring event, 이벤트 구간 밖에서는 Negative scoring event 를 가지고 있다.

신호 A 를 보면 이벤트 구간 밖에서는 점수가 변하지 않지만 이벤트 구간 내에서는 Positive scoring event 가 발생해 점수가 증가한다. 모든 이벤트가 끝난 뒤에는 값의 변화가 없기 때문에 일정한 점수를 유지하게 된다. 그러나 신호 B 의 경우 이벤트 구간 밖에서도 데이터가 변화하여 Negative scoring event 가 발생하게 되고 첫 번째 이벤트 영역에 도달하기 전에 이미 음수 값의 점수를 갖는다. 3 개의 이벤트 구간을 지나면 신호 B 는 Positive scoring event 에 의해 양의 값의 점수를 갖게 되지만 모든 이벤트 종료 이후 시간이 지나면 다시 Negative scoring event 의 영향을 받아 점수가 감소한다. 결과적으로 시간이 지나면 목표로 했던 신호 A 만 양의 값의 점수를 갖게 되어 A 신호를 찾을 수 있다.

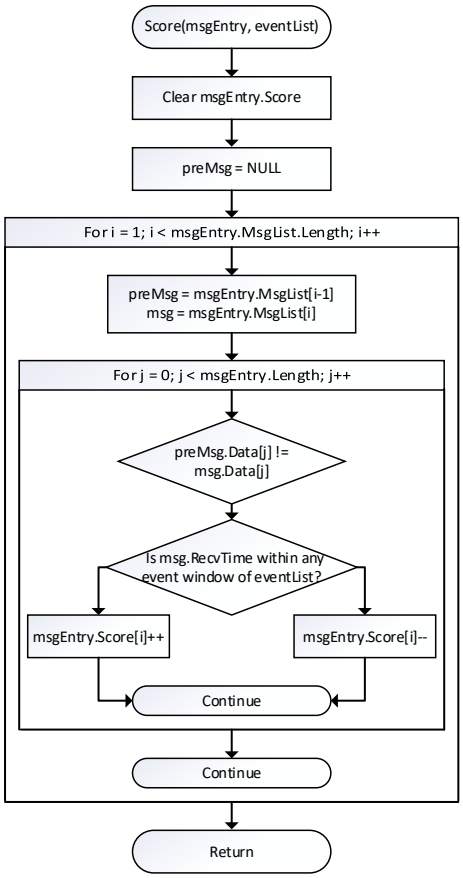

Figure 2 점수 계산 알고리즘

점수는 Figure 2 의 과정을 통해 계산한다. 우선 점수를 초기화한 뒤 해당 메시지에 대해 수신한 모든 프레임을 체크하여 각각의 바이트 데이터의 변화를 찾는다. 이 과정에서 값이 변하지 않았다면 다음 프레임을 이어서 탐색하고 값이 변하였을 경우 해당 메시지의 수신 시간을 확인한다. 수신 시간이 이벤트 탐지 범위 안이라면 점수를 더하고 이벤트 범위 밖이라면 점수를 뺀다. 또한 점수 계산 시 바이트 단위의 계산과 동시에 비트 단위 점수도 계산하여 해당 이벤트와 관련된 신호의 경계도 추정할 수 있도록 한다. 모든 프레임에 대한 검사가 완료되면 다른 메시지에 대해서도 이 과정을 반복한다.

Ⅲ. 실험 및 결과

이벤트를 설정하고 점수 계산 방식을 통해 원하는 메시지를 찾을 수 있는지 실험을 진행하였다. 실험에 사용된 차종은 현대 자동차의 아반떼이다.

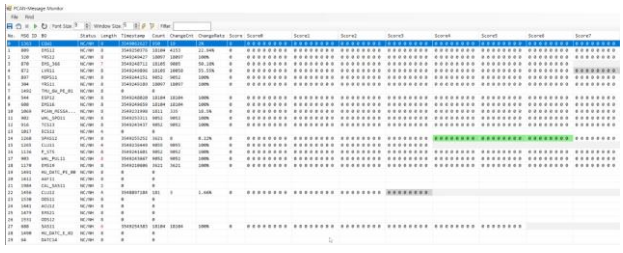

Figure 3 계산 결과 메시지 발견

첫 번째 테스트는 상태정보와 같은 불 연속적인 신호를 측정하기 위해 차량의 오른쪽 뒷문의 상태정보를 찾는 것을 목표로 잡고 문을 열고 닫을 때마다 이벤트를 추가하였다. window size 를 500ms 로 설정하고 실험을 수행한 결과 이벤트 종료 직후 15 개의 메시지의 점수가 증가한 것을 확인할 수 있었다. 그러나 약 1 분 뒤 Figure 3 과 같이 하나의 메시지를 제외하고 모두 점수가 0 이하로 감소하였고 남아있는 메시지를 조사해보니 0x553 메시지의 23 번째 비트의 값이 8 번 바뀐 것을 확인할 수 있었다. 이 정보를 OpenDBC 에 공개된 메시지 정보와 비교해보니 0x553 메시지의 23 번째 비트가 CF_Gway_RRDrSw 신호에 해당함을 확인해 메시지 ID 뿐 아니라 신호의 구체적인 비트 위치까지 찾았음을 확인할 수 있었다.

두 번째 실험은 차량의 핸들을 돌려가며 연속적인 데이터의 변화에 반응하는 CAN 신호를 탐색하였다. 해당 테스트의 경우 이벤트가 특정 시점이 아닌 범위에 걸쳐 발생했기 때문에 이벤트와 관련 없는 신호의 값 변화가 첫 번째 테스트보다 많이 측정되었고 그에 따라 해당 신호들이 탈락하는데 걸리는 시간도 첫 테스트 보다 오래 걸렸다. 하지만 시간이 지나자 두 개의 메시지의 점수만 양수 값을 유지하게 되었고 OpenDBC 의 내용을 참조해 두 신호 모두 핸들 조작과 관련 있는 신호임을 확인할 수 있었다.

Ⅳ. 결론

처음에 CAN 버스를 통해 전달되는 메시지 중 특정 이벤트와 관련된 신호를 쉽고 빠르게 찾기 위해 분산의 차를 사용하는 방법을 적용했으나 찾고자 하는 신호를 직관적으로 알아보기 어려웠다. 그래서 이를 개선하기 위해 이벤트의 발생 순간을 기록하고 메시지, 비트 별로 값의 변화에 따른 점수제를 적용해 실험을 진행하였다.

몇 가지 이벤트를 실험 한 결과 짧은 시간안에 이벤트와 관련된 소수 메시지의 점수만 양수를 유지했고 OpenDBC 정보와 비교한 결과 해당 신호가 찾고자 했던 신호와 일치함을 확인할 수 있었다. 또한, 비트 단위의 점수도 계산하여 비트 단위의 추적도 가능했고 이를 통해 신호의 범위도 추정 가능했다. 따라서 이를 활용하면 DBC 정보가 없는 메시지의 신호도 분류할 수 있을 것이다.

테스트 결과를 통해 특정 신호를 추적하여 발견하는데 본 연구의 방법이 효과가 있음을 확인할 수 있었다.

ACKNOWLEDGMENT

이 논문은 2021 년도 한국연구재단의 국제협력사업 지원을 받아 연구되었음 (NRF-2021K1A3A1A61003229)

참 고 문 헌

[1] Huybrechts T., Vanommeslaeghe Y., Blontrock D., Barel G. V., & Hellinckx P. “Automatic Reverse Engineering of CAN Bus Data Using Machine Learning Techniques”, Nov. 2017

[2] Marchetti M., & Stabili D. “READ: Reverse engineering of automotive data frames”, Apr. 2019

[3] Buscemi A., Turcanu I., Castignani G., Crunelle R., & Engel T. “CAN Match: A Fully Automated Tool for CAN Bus Reverse Engineering based on Frame Mathching” Dec. 2021

[4] W Choi., S Lee., K Joo., H Jo., & D Lee. “An Enhanced Method for Reverse Engineering CAN Data Payload” Apr. 2021.