

제한 주행시간을 만족하는 에너지 효율적인 전기자동차 주행 최적화 기법

(Energy Efficient Electric Vehicle Driving Optimization Method
Satisfying Driving Time Constraint)

백 돈 규^{1)*}
(Donkyu Baek)

요약 본 논문은 추가 비용 없이 전기자동차(EV) 주행 범위를 확장하기 위해 에너지 효율적인 전기자동차 주행 프로파일을 도출하는 새로운 시스템 수준의 프레임 워크를 소개한다. 이 논문은 먼저 운전 차량에 작용하는 힘과 모터 효율을 고려한 전기차 파워 트레인 모델을 구현한 후, 경로에 의해 정의된 주행 임무에 대한 최소 에너지 주행 프로파일을 도출한다. 이를 위해서 본 프레임워크는 먼저 최적화 문제를 공식화하고, 가중치 계수를 이용한 동적 프로그래밍 알고리즘을 사용하여 에너지 소비와 운전 시간을 모두 최소화하는 주행 프로파일을 도출한다. 본 논문은 주행 시간 제약을 만족시키기 위한 다양한 가중치 계수 도출 방법을 소개한다. 시뮬레이션 결과, 제안된 스케일링 알고리즘의 연산시간이 이전 검색 알고리즘 및 탐욕 알고리즘보다 각각 34 % 및 50 % 더 작음을 보여준다.

핵심주제어: 전기자동차(EV), 주행 프로파일, 다이나믹 프로그래밍 기법, 가중치 계수

Abstract This paper introduces a novel system-level framework that derives energy efficient electric vehicle (EV) driving speed profile to extend EV driving range without additional cost. This paper first implements an EV power train model considering forces acting on a driving vehicle and motor efficiency. Then, it derives the minimum-energy driving speed profile for a given driving mission defined by the route. This framework first formulates an optimization problem and uses the dynamic programming algorithm with a weighting factor to derive a speed profile minimizing both of energy consumption and driving time. This paper introduces various weighting factor tracking methods to satisfy the driving time constraint. Simulation results show that runtime of the proposed scaling algorithm is 34% and 50% smaller than those of the binary search algorithm and greedy algorithm, respectively.

Keywords: Electric vehicle, Driving speed profile, Dynamic programming method, Weighting factor

1. 서론

* Corresponding Author: donkyu@cbnu.ac.kr

Manuscript received March 19, 2020 / revised April 09, 2020 / accepted April 12, 2020

1) 충북대학교 전자공학부

기존의 내연기관 운송수단들은 에너지 효율이 나쁘고 환경오염을 야기하는 문제를 발생시키기

때문에 전기 모터로 동작하는 다양한 운송수단 및 시스템 (e-모빌리티)가 등장하였다. IHS Markit에 따르면, 현대에 이르러 배터리 기술의 발달로 인해, 전문가들은 미국에서 전기 자동차 판매량이 2020년 연간 35만개에서 2025년에는 연간 백만 개로 증가할 것으로 추정하였다. 또한, 시스템에 의해 운행되는 세계 자율 주행차 시장 규모는 2019년 543억에서 2026년 까지 5천억 달러를 기록하여 연평균 39.47%의 성장률을 기록할 것으로 예상된다 (Jiao et al., 2019).

자율주행에 대한 연구도 점차 본격화됨에 따라 보다 개선된 자율주행에 대한 연구도 등장하기 시작했다. 현재 자율주행에서 활발히 진행되고 있는 연구는 사람의 주행 패턴과 유사하게 운전하는 것이라면, 미래의 자율주행은 주어진 도착시간을 만족하면서 에너지를 최대한 절감하는 등 최적화된 자율주행이 유행할 것으로 보인다.

본 논문은 주행 시간 제한을 고려하면서 에너지를 최소화하기 위한 전기자동차 주행 최적화 프레임워크를 제안한다. 먼저 전기자동차의 주행에 따른 에너지 소비를 고려하는 에너지 모델을 도출한다. 그 이후에, 주행 문제를 수식화하여 널리 사용되는 다이나믹 프로그래밍 기법을 활용하여 최적 주행 프로파일을 도출한다.

본 논문은 단순히 주행 에너지만을 최소화하는 것이 아닌 주행시간과 에너지 모두를 최소화하고자 한다. 이 문제를 해결하기 위해 주행시간과 에너지 사이의 가중치 계수를 도출하는 알고리즘을 제안한다. 빠른 시간내에 주어진 문제에 적합한 가중치를 찾아내어 주어진 주행시간 제한을 만족하면서 에너지를 최소화하고자 한다. 시뮬레이션을 통해 확인한 결과, 제안한 가중치 계수 도출 알고리즘에 의한 총 연산시간이 이전 검색 알고리즘 및 탐욕 알고리즘보다 각각 34 % 및 50 % 더 작음을 보여준다.

2. 전기자동차 주행 최적화 배경 지식

2.1 전기자동차 주행 최적화 기법

일부 최적화 알고리즘은 주어진 주행에 대해

Dijkstra와 유사한 알고리즘 및 분석기법으로 해결된다 (Park et al., 2011). 다이나믹 프로그래밍 기법은 출발지에서 목적지까지의 모든 도로 정보를 고려하여 글로벌 최적을 도출하기 때문에 다양한 최적화 알고리즘들 사이에서 널리 사용된다 (Dib et al., 2012; Ozatay et al., 2014; Baek and Chang, 2019). 도로 정보는 하나의 도로가 N 개의 개별 위치로 정의되며, 각 위치별 차량 속도 범위는 최대 및 최소 속도 중 하나를 갖는 것으로 정의된다. 이 상태에서 각 개별 위치에서 어떤 속도로 주행하는 것인지 정하는 문제가 바로 최적 주행 문제이다. 일반적으로는 최적 주행 문제를 풀기 위해 다이나믹 프로그래밍 기법이 사용된다. 이 방식은 모든 경로를 연산하는 것이 아닌 현재까지 얻어진 최적 값으로부터 다음 위치에서의 최적 값을 도출하는 것으로써 폴리노미알 시간 안에 최적 값을 도출할 수 있다는 큰 장점을 갖는다.

하지만, 다이나믹 프로그래밍 기법의 가장 큰 한계는 다중 목적 문제를 풀 수 없다는 것으로, 주행 최적화 문제를 예로 들면, 1) 주행 에너지와 2) 주행 시간 모두를 만족하는 결과를 도출할 수 없다. 보통 이러한 문제를 해결하기 위해서는 가중치 계수 (Weighting factor)를 활용하여 우회적으로 다중 목적 문제를 풀 수 있지만 (Dib et al., 2012; Mensing et al., 2013; Baek and Chang, 2019), 주어진 주행 제약 시간을 만족하기 위한 가중치 계수를 어떻게 정해야 하는지에 대한 연구는 아직 이뤄지지 않고 있다. 또한, 사용자의 선호도에 따라 주행 에너지와 주행 시간 사이에서 단계적으로 선택할 수 있도록 에너지-지연시간 곱의 형태인 목적함수를 제시하기도 하였다 (Baek and Chang, 2019).

2.2 전기자동차 동역학 모델링

전기차 전력 모델링은 널리 알려진 동일한 차량 동역학 모델에서 시작할 수 있다. 전기차 전력 소비의 주요 비율은 전기차의 추진력에서 비롯된다. Fig. 1은 운전 중 전기차에 가해지는 힘을 보여주는데, 차량 동역학 모델은 물리 방정식으로 추진력의 원천을 설명한다. 이는 도로 기울기, 전기차 속

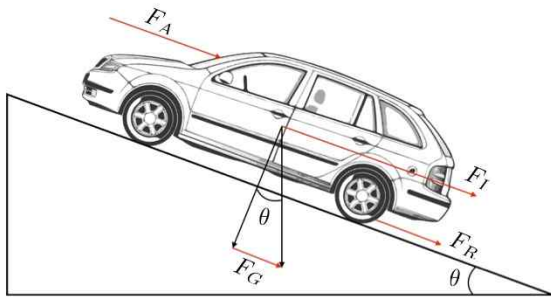


Fig. 1 Forces Acting on a Driving Vehicle

도, 전기차 가속 및 전기차 질량에 따라 단순화되며, (1)과 같이 정리된다 (Chang et al., 2014).

$$P_{Dynamics} = F \frac{ds}{dt} Fv = (F_R + F_G + F_I + F_A)v \quad (1)$$

$$\approx (\alpha + \beta \sin\theta + \gamma a)mv$$

차량 동역학 모델 (1)에서 F_R , F_G , F_I , F_A , θ , m , v 및 a 는 구름 저항, 경사 저항, 관성 저항, 공기 역학 저항, 도로 경사, 차량 무게, 속도 및 가속도를 의미한다. 그리고 주어진 모델의 계수인 α , β , γ 는 각각 구름 저항, 중력 저항, 관성 저항을 나타낸다. 이때, 공기 역학 저항 F_A 는 속도가 60 km 이상인 경우에만 영향이 있으나, 본 실험 대상 차량은 최대 35 km/h 이하에서만 동작하므로 생략하였다.

일반적인 차량 역학 모델은 모터 효율이 100%라고 가정하기 때문에 차량 역학 모델에 기초한 전력 추정에는 한계가 있다. 일부 작업에서는 모터 효율을 상수로 가정하고 다른 작업에서는 제조업체에서 모터 효율 맵을 가져 와서 전기차 전력 소비 모델과 통합하려고 했다 (Hayes et al., 2011; Rahimi-Eichi and Chow, 2014). 그러나 이러한 종류의 접근 방식은 정확도에 한계가 있으며, 모터 효율을 정확히 모델링하기 위해서는 각 모터 손실의 원인을 개별적으로 분석 후 고려해야 한다. (2)는 모터손실 모델을 나타낸다.

$$L = k_i \omega + k_w \omega^3 + k_c Q^2 + C \quad (2)$$

모터 손실(L)은 모터에 적용되는 각속도와 토크에 따라 정의할 수 있으며, (2)와 같이 나타낸다. 모터 손실은 크게 마찰손실, 풍량 손실, 토

크 손실로 이루어져 있다. (2)에서 k_i , k_w , k_c 는 각각 마찰계수, 풍량 계수, 전류와 토크 사이의 스케일링 된 계수를 의미한다. 따라서, 전기자동차 동역학 모델과 모터효율 모델을 모두 고려한 파워트레인 모델은 (3)과 같이 정리할 수 있다.

$$P \approx Tv + C_0 + C_1v + C_2v^2 + C_3T^2 \quad (3)$$

$$T \approx (\alpha + \beta \sin\theta + \gamma a)$$

이때, T는 모터 토크를 나타낸다 (Baek and Chang, 2019).

최근에는 뉴로-퍼지 시스템 기반의 배터리 잔여용량 예측 기법이 제안되고 있다. 기존에 사용되었던 동역학 기반 모델링에서 더 나아가 배터리 온도, 전장 시스템에 의한 전력, 그 외에 기타 전력 소비를 모두 포함함으로써 더욱 배터리 잔량 예측 모델을 제시하고 있다 (Zahid et al., 2018). 하지만, 매우 많은 주행데이터 및 모니터링 결과를 요구하고, 주행 특성 뿐 아니라 다양한 환경 변화에 대한 모델링을 위해 시스템이 매우 복잡해지기 때문에 본 연구에는 적합하지 않다.

한편, 전기자동차의 주행에너지 뿐만 아니라, 태양광 패널을 활용한 충전 모델, 교통량에 따른 속도 모델, 배터리 특성 모델을 모두 고려한 통합 전력 운용 모델링에 대한 연구도 진행되고 있다(Chen et al., 2019). 제안한 모델을 활용하여 하루 24시간 동안 발생하는 전기자동차 배터리의 충, 방전 시뮬레이션을 수행하였다. 다만, 해당 논문은 주행 최적화에 집중한 본 논문의 주제와는 다소 떨어져있다.

3. 전기자동차 최적 주행 프레임워크

전기자동차의 최적 주행 경로를 도출하기 위해서는 먼저 특정 주행 패턴에 따라 전기차의 에너지 소비를 예측하는 것이다. 전기자동차의 최적 주행 경로를 도출하는 가장 간단한 방법 중 하나는 등속주행 기법이다. 먼저, 일정 속도(v)로 가속(a)한 다음에, 그 이후로는 목적지에 도착할 때까지 꾸준히 같은 속도(v)로 주행하는 것이다. 이때, 전체 에너지 소비를 최소화하는 v와 a를 도출하는 것이다. 이 문제를 풀기 위해

서는 같은 거리를 가는 다양한 속도-가속도 조합에 대해 전체 에너지 소비를 도출한 다음 비교하는 것이다. 3장에서 얻은 에너지 모델을 활용하여 주어진 기울기, 가속도, 속도, 무게 정보를 알면 에너지 소비를 예측할 수 있다.

이 방식은 해당 주행 경로의 기울기가 일정할 경우, 최적 값을 보여준다. 하지만, 기울기가 매 위치별로 달라지는 경우, 기울기에 따라 최적 값이 달라지고 그에 따라 속도를 변경해야하기 때문에 등속 주행 방법으로는 최적 주행 기법을 얻어낼 수 없다.

한편, 단위 거리별로 최적 속도를 갖는 문제는 Fig. 2와 같이 나타낸다. 시작점 s에서 도착점 t까지의 하나의 주행 경로를 M개의 위치로 나눈 다음에, 각 위치별로 N개의 속도 후보군 중에 최적의 속도를 도출하는 문제이다. 여기서 하나의 위치 별로 단 한 개의 노드만 선택될 수 있으며(v = N개의 노드 중 1개), s부터 t까지 총 M개의 위치 중 각각(v)에 대해 발생하는 에너지를 모두 더해야 하므로, (4)와 같이 수식화할 수 있다.

$$\begin{aligned} & \text{Min} \sum_{d=1}^M \sum_{v=1}^N \left\{ \Delta t \frac{x(d-1,v)P_{d-1,v} + x(d,v)P_{d,v}}{2} \right\} \\ & \sum_{v=1}^N x(d,v) = 1 \quad \forall v \\ & x(d,v) = 0,1 \quad \forall d,v \end{aligned} \tag{4}$$

이때, x(d,v)는 각 노드 별 식별 함수로, 만약 해당 노드가 선택되었다면 1, 선택되지 않았다면 0 값을 가진다. 앞서 정의한 대로, 하나의 위치 d에 대해 단 하나의 속도만 가질 수 있으므로, 모든 v에 대해 단 하나의 노드만이 1을 가질 수 있다. P_{d,v}는 노드 d, v에서 발생한 전력을

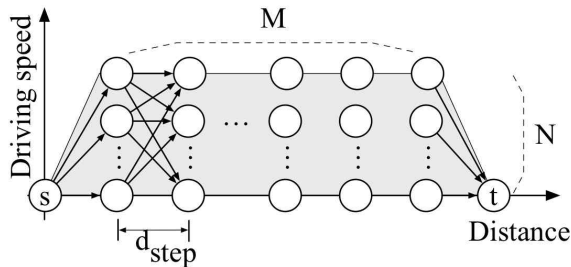


Fig. 2 Optimal Driving Problem Formulation

의미하는 것으로, d-1위치에서 d 위치까지의 평균 에너지를 도출하기 위해 평균 전력에 주행 시간 Δt를 곱하였다.

이 문제는 동적 프로그래밍을 통해 도출 가능하다 (Dib et al., 2012; Mensing et al., 2013; Ozatay et al., 2014). 동적 프로그래밍은 목적 함수가 최소가 되는 값과 그 경로를 각 노드에 저장한다. 그리고, 다음 위치로 넘어가면서 이전에 저장한 데이터를 그대로 활용하기 때문에 폴리노미알 시간 안에 결과물을 도출할 수 있다.

4. 최적주행 가중치 계수 도출 기법

만약 주행 시간 제한이 존재한다면, 앞서 언급한 동적 프로그래밍 기법으로는 풀기가 매우 까다롭다. 그 이유는 동적 프로그래밍 기법은 단 하나의 목적함수를 최소화하는 방향으로 해답을 도출해 나가기 때문이다. 즉, 오직 에너지를 최소화하는 주행 경로만을 찾을 뿐, 제약조건을 고려하지 못한다.

따라서, 이러한 문제를 해결하기 위한 방법으로, 많은 논문들이 가중치 계수 (Weighting factor)를 활용한다. 즉, 임의의 가중치 계수 ω를 이용하여 ω(에너지) + (1-ω) (시간)을 하나의 목적함수로 활용할 수 있다. 예를 들어, 에너지와 주행 시간을 모두 고려하기 위한 최적 주행 프로그래밍 문제는 (5)와 같이 수식화 할 수 있다.

$$\begin{aligned} & \text{Min} \sum_{d=1}^M \sum_{v=1}^N \left\{ \omega \left(\Delta t \frac{x(d-1,v)P_{d-1,v} + x(d,v)P_{d,v}}{2} \right) \right. \\ & \left. + (1-\omega)\Delta t \right\} \\ & \sum_{v=1}^N x(d,v) = 1 \quad \forall v \\ & x(d,v) = 0,1 \quad \forall d,v \end{aligned} \tag{5}$$

이때, 가장 중요한 점은 올바른 가중치 계수를 도출하는 것이다. 가중치 계수는 단지 주행 시간과 에너지 소비를 모두 고려한 주행을 한다는 것이지, 주어진 주행 시간을 만족하는 최소 에너지 주행을 한다는 것이 아니다. 따라서, 우리가 원하는 제약 조건을 주행 시간으로 두고 에너지 소비를 최소화

하는 문제를 풀기 위해서는 최적의 가중치 계수를 찾아야 한다. 하지만, 현실적으로 최적의 가중치 계수를 찾는 것을 불가능하며, 주행 시간 제한을 만족할 때까지 가중치 계수를 계속 바꿔가면서 문제를 풀어야 한다. 하지만, 모든 이전 연구에서는 가중치 계수를 언급하기만 하고, 최적의 가중치 계수를 찾기 위한 방법은 설명하지 않는다.

따라서, 본 논문은 주행 시간 제한을 만족하는 에너지 최적 주행 결과를 위한 가중치 계수를 도출하기 위한 세 가지 알고리즘을 제안한다. 첫 번째로 이진 검색 알고리즘 (Binary search algorithm)은 $O(\log n)$ 연산시간 내에서 최적의 가중치를 도출하는 가장 유명한 검색 알고리즘 중 하나이다. 이 알고리즘은 전체 솔루션 공간의 중간 점 (0.5)에서 최적의 가중치를 추적하기 시작한다. 그런 다음, 알고리즘은 나머지 반쪽 영역에서 다음 중점을 찾아 가중치 계수 후보로 두고 연산한다.

$$\omega = \omega_{old} + a \frac{1}{2^{n+1}} \quad (6)$$

$$a = \begin{cases} 1 & \text{if } deadline > drivingtime \\ -1 & \text{otherwise} \end{cases}$$

(6)은 이진 검색 알고리즘의 가중치 도출 수식을 나타내며, 변수 a 를 사용하여 다음 검색을 위한 가중치 계수를 결정한다. 만약, ω_{old} 를 사용했을 때, 제한 주행시간 (Deadline)을 만족하는 경우, 1이 되고, 그렇지 않은 경우 -1이 된다. 이때, n 은 연산 횟수를 의미한다.

두 번째, 탐욕 알고리즘 (Greedy algorithm)은 초기 가중치 계수에서 시작하여, 차차 가중치 값을 바꿔가면서 최적 가중치 값에 수렴해가는 방법이다. (7)은 다음 검색을 위한 가중치 결정 방법을 나타낸다. 만약, 주어진 가중치 계수를 사용한 결과 값이 특정 값으로 수렴하지 않고 진동한다면 추적 단계를 절반으로 줄인다.

$$\omega = \omega_{old} + a\omega_{step} \quad (7)$$

$$\omega_{step} = \begin{cases} \frac{\omega_{step}}{2} & \text{oscillates} \\ \omega_{step} & \text{otherwise} \end{cases}$$

이때, 초기 가중치 계수는 등속 주행을 가정함으로써 (8)과 같이 도출한다. 먼저 목적함수는

에너지와 주행 시간의 가중치에 의한 합으로 나타낼 수 있으며, 각 에너지와 시간은 등속 주행 시 발생하는 에너지로 나타낼 수 있다. 이때, P 는 앞서 파워트레인 값을 의미한다.

$$Obj. = \omega(energy) + (1-\omega)(time) \quad (8)$$

$$= \omega \int P \frac{dt}{ds} ds + (1-\omega) \left(\frac{dt}{ds} \right) ds$$

$$= \omega \int \left(T + C_0/v + C_1 + C_2v + \frac{C_3 T^2}{v} \right) ds + (1-\omega) \left(\frac{1}{dv} \right) ds$$

이때, (9)와 같이 목적함수가 주행 속도에 따라 최소값이 될 때(속도로 미분되는 값이 0)의 가중치 값을 도출하며, 그 결과, (10)과 같이 목적함수가 최소가 되는 가중치 초기값을 도출한다.

$$\frac{dObj.}{dv} \Big|_{v=v_{opt}} = \omega \int \frac{-C_0 + C_2v_{opt}^2 - C_3T^2}{v_{opt}^2} ds \quad (9)$$

$$+ (1-\omega) \int -\frac{1}{v_{opt}^2} ds = 0$$

$$\therefore \omega = \frac{1}{1 - C_0 + C_2v_{opt}^2 - C_3T^2} \quad (10)$$

마지막 스케일링 알고리즘 (Scaling algorithm)은 탐욕 알고리즘을 더욱 개선한 알고리즘으로, (11)과 같이 최적 가중치 값에 수렴하기 위해 단순히 특정 값을 더하는 것이 아닌, 제한 주행시간과 현재 주행 시간의 차이만큼 스케일링하여 더해주는 것이다. 이때, 가중치 초기값은 앞서 구한 (8-10)을 통해 도출한다.

$$w = w_{old} + \frac{deadline - drivingtime}{deadline} w_{step} \quad (11)$$

5. 시뮬레이션 결과 및 분석

5.1 시뮬레이션 설정

최소 에너지 주행 프로파일을 검증하기 위해 Fig. 3에서 보이는 경량 맞춤형 전기차를 사용하였다 (Chang et al., 2014). 해당 전기차의 무게는 481kg이며 연속 1.5kW 및 최대 4.0kW 전

력을 가진 2 개의 BLDC 트랙션 모터를 사용한다. 최대 주행 속도는 35km/h이다. 해당 전기차는 76.8V 전압과 총 3.7kWh Li-FePO4 배터리 팩으로 구성되어있다.

이때, 휠에 적용되는 전체 토크가 양수이면 전기차는 전력 손실을 포함한 배터리 에너지를 소비하지만, 만약 모터에 작용하는 전체 토크가 음수이면 전기차는 전기 및 전기 시스템에 의한 전력 손실 없이 배터리 에너지를 소비하지 않는다. 또한, 2kW 이내로 회생 제동에 의한 에너지 하베스팅이 발생하여 배터리를 충전시키게 된다.

Table 1에서 명시된 해당 전기차의 에너지 모델은 Hong et al.(2016)이 제안한 모델에서 가지고 왔으며, 최대 차량 속도는 앞서 언급한 대로 35 km/h이며, 주행 중 최소 속도는 10km/h, 시



Fig. 3 Electric Vehicle for Driving in a Campus

작 시간에서의 최소 가속은 전기차가 10m 이내에 최소 차량 속도에 접근하도록 설정하였다.

본 시뮬레이션은 Matlab을 통해 진행되었다. on-line 시뮬레이션을 목표로 구성하였기에, Matlab을 구동한 서버의 성능은 휴대성을 고려하여 단일 CPU 2.7 GHz Intel Core i5와 8 GB 1867 MHz DDR3로 구성하였다.

5.2 전기자동차 최적 주행 시뮬레이션 결과

Fig. 4는 제한 주행시간에 따른 주행 프로파일을 보여준다. 가장 위쪽 그래프는 주행 도로의 고도를 의미한다. 약 350m와 1000m 위치에서 언덕이 있음을 알 수 있다. 두 번째 그래프는 주행 프로파일을 나타낸다. Y값은 각 위치별 주행 속도를 나타낸다. 빨간 선은 주행시간 제약이 없는 주행 프로파일을 나타내며 파란색

Table 1 Coefficients of EV Power train Model (Hong et al., 2016)

Coeff.	Value	Coeff.	Value
α	0.32	C_1	7.39
β	10.11	C_2	20.62
γ	1.08	C_3	0.0019
C_0	4.0		

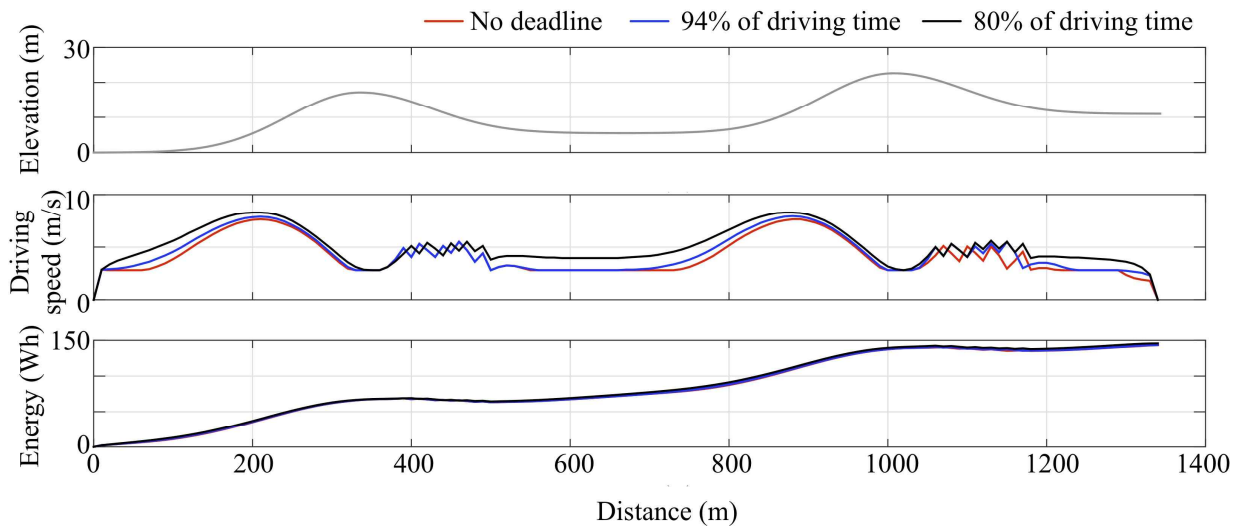


Fig. 4 Minimum-energy Driving Speed Profiles by Deadline Constraints

과 검은 색 선은 주행시간 제약이 있는 주행 프로파일을 나타낸다. 이 둘의 주행시간은 앞서 주행시간 제약이 없이 주행한 주행 프로파일에 의한 주행시간 대비 각각 94%와 80%이다. 가장 아래쪽 그래프는 각 주행 프로파일 별 에너지 소비를 나타낸다.

일반적으로 모든 주행 프로파일에서 오르막 이전에 속도를 미리 올리며, 오르막 동안엔 속도를 줄인다. 또한, 내리막에서는 회생제동을 통해 속도가 갑작스럽게 떨어지고 다시 중력에 의해 속도가 증가함을 알 수 있다.

주행 시간이 80%인 주행 프로파일 (검은 색)의 경우, 주행 시간을 줄이기 위해 평지 주행 시 속도를 더욱 올렸다. 반면, 오르막에서는 속도를 추가로 올리지 않았다. 따라서, 주행 시간을 20 % 줄였으나, 전체 에너지 소비는 겨우 1.77 % 증가함을 알 수 있다.

5.3 주행시간제약을 만족하는 최적 주행 결과

이와 같이 주행 시간에 제약을 받는 주행 프로파일 도출 문제의 경우, 연산 시간을 줄이는 것이 매우 중요하다. 그 이유는 연산시간이 짧다면 실제 주행 중에 실시간으로 최적 주행 프로파일을 도출하여 활용할 수 있기 때문이다. 만약, 연산시간이 매우 느리다면 주행 중 특정 사고나 교통변화로 인해 새로운 경로를 도출해야 할 때 즉시 최적의 주행 프로파일을 도출하지 못하게 된다.

본문에서는 앞서 설명한 세 가지 가중치 계수 도출 방식에 따른 연산시간을 비교하였다. 만약 주행 시간이 제한 시간보다 크게 줄어든다면 그에 따라 주행 에너지 소비는 그만큼 늘어나게 된다. 따라서, 가장 중요한 것은 주행 제한 시간을 만족하면서 가장 에너지를 적게 소비하는 주행 프로파일을 도출하는 것이다.

5.4 최적주행 가중치계수 추적 분석

Fig. 5는 제한 주행 시간이 각각 354초 (a), 24초 (b)일 때, 세 가지 가중치 계수 추적 방법에 의한 주행 프로파일 도출 시 각각의 주행 시간

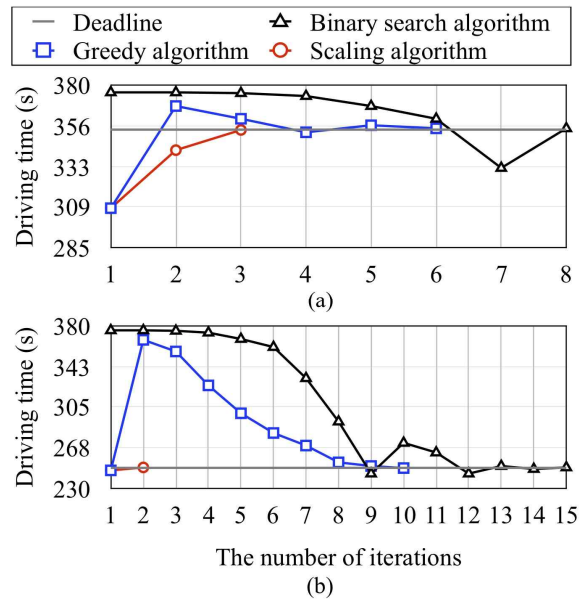


Fig. 5 Weighting Factor Tracking Results in Different Deadline Constraints (a) 354 s and (b) 248 s

변화를 보여준다. 회색 선은 제한 주행 시간을 표시하였으며, 검은 색, 파란 색, 붉은 색은 각각 이진 검색 방법, 탐욕 검색 방법, 스케일링 검색 방법에 의한 주행시간 변화를 보여준다. 이 중에 스케일링 알고리즘은 가장 빠르게 제약 주행시간에 도달한다. 반면, 탐욕 알고리즘 역시 빠르게 제약 주행시간에 도달하긴 하지만, 어느 정도의 진동을 통해 6번의 반복 후 제약 주행시간에 도달한다. 하지만, 이진 검색 알고리즘의 경우, 추적 시 상당한 단계를 거친 후 결과를 얻는다. 그 이유는 검색 초기 값에서 큰 차이가 있기 때문이다. 한편, Fig. 5 (b)는 248초인 제약 주행시간에 대해 가중치 계수 추적을 보여준다. 보이는 바와 같이 스케일링 알고리즘은 바로 결과를 찾았으나, 탐욕 알고리즘은 미리 정해진 추적 단계 (ω_{step})가 너무 작기 때문에 많은 반복을 수행한다.

5.5 최적주행 가중치계수 도출 방법에 따른 연산시간 비교

Fig. 6은 여러 벤치마크(다양한 도로 기울기

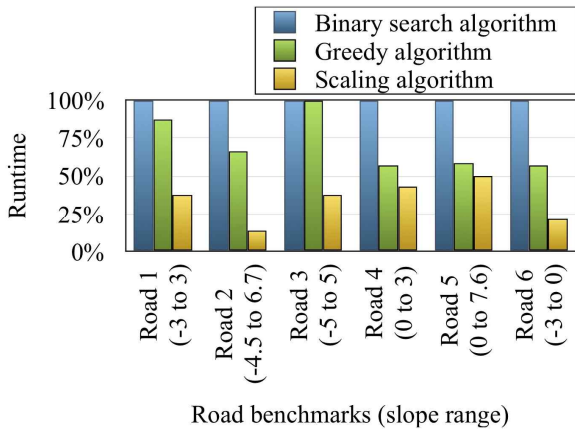


Fig. 6 Runtime Analysis by the Weighting Factor Tracking Algorithms

를 갖는 테스트 도로들)에서 세 가지 가중치 계수 추적 알고리즘 별 연산시간을 보여준다. 스케일링 알고리즘은 각각 이진 검색 및 탐욕 알고리즘의 연산시간 대비 34 % 및 50 %의 연산시간을 요구한다. 또한, 스케일링 알고리즘은 목표 주행시간에 도달하기 위해 7 회 이상 반복을 초과하지 않으며, Matlab을 활용한 본 시뮬레이션 환경에서 30 초가 걸린 것을 확인하였다. 만약, C / C ++ 구현을 통해 메모리 및 라이브러리 접근을 보다 효율적으로 수행하고, 다수 CPU를 활용한 병렬 연산을 수행한다면 런타임이 10 배 이상 줄어들 것으로 예상된다. 따라서, 실제 자동차 주행 시 5초 이내에 최적 주행 프로파일을 도출하여 실시간으로 활용 가능할 것으로 예상된다.

6. 결론

전기자동차 시대는 가까운 미래에 올 것으로 예상되나, 내연기관차 대비 약 22%인 주행거리는 전기자동차 구매자에게는 큰 허들이 되고 있다. 최소 에너지 주행은 별도의 큰 투자 및 연구 없이 빠르게 주행거리 문제를 해결하는 방법 중 하나이고, 이에 대한 다양한 연구가 진행되고 있다. 본 논문에서는 기존에 전혀 다뤄지지 않은 주행 시간을 만족하는 최적 주행 기법에

대한 분석을 진행하고, 주어진 제한 주행시간을 만족하기 위한 최적의 가중치 계수를 찾는 알고리즘을 검증하였다. 머지않은 미래에 전기차 시대가 더욱 도래하고 자율주행 차가 대중화될 때, 제한 주행시간을 만족하는 최소 에너지 주행 프로파일은 가장 효율적이고 보편적으로 사용될 것으로 보인다.

References

- Baek, D., and Chang, N. (2019). Runtime Power Management of Battery Electric Vehicles for Extended Range With Consideration of Driving Time, *Transactions on Very Large Scale Integration Systems*, 27(3), 549-559.
- Chen, Y., Baek, D., Kim, J., Cataldo, S., Chang, N., Macii, E., Vinco, S., and Poncino M. (2019). A SystemC-AMS Framework for the Design and Simulation of Energy Management in Electric Vehicles, *IEEE Access*, 7, 25779-25791.
- Chang, N., Baek, D., and Hong, J. (2014). Power Consumption Characterization, Modeling and Estimation of Electric Vehicles, *IEEE/ACM International Conference on Computer-Aided Design*, Nov. 2-6, San Jose, USA.
- Dib, W., Chasse, A., Domenico, D., Moulin, P., and Sciarretta, A. (2012). Evaluation of the Energy Efficiency of a Fleet of Electric Vehicle for Eco-driving Application, *Oil and Gas Science and Technology - Revue de IFP Energies Nouvelles*, 67(4), 589-599.
- Hayes, J., Oliveira, R., Vaughan, S., and Egan, M. (2011). Simplified Electric Vehicle Power Train Models and Range Estimation, *Vehicle Power and Propulsion Conference*, Sept. 6-9, Chicago, USA.
- Hong, J., Park, S., and Chang, N. (2016). Accurate Remaining Range Estimation for Electric Vehicles, *Asia and South Pacific*

Design Automation Conference, Jan. 25-28, Macau, China.

IHS Markit Release (2019). *US Electric Vehicle Loyalty and Volumes Reach Record Highs, according to IHS Markit*, <https://www.businesswire.com/news/home/20190415005540/en/Electric-Vehicle-Loyalty-Volumes-Reach-Record-Highs> (Accessed on Dec. 30th, 2019)

Jiao, N., Ghaffarzadeh, K., and Jiang, L. (2019). *Autonomous Cars and Robotaxis 2020-2040: Players, Technologies and Market Forecast*, IDTechEX.

Mensing, F., Bideaux, E., Trigui, R., and Tattgrain, H. (2013). Trajectory Optimization for Eco-driving taking into Account Traffic Constraints, *Transportation Research Part D: Transport and Environment*, 18, 55-61.

Ozatay, E., Onori, S., Wollaeger, J., Ozguner U., Rizzoni, G., Filev, D., Michelini, J., and Cairano, S. (2014). Cloud-based Velocity Profile Optimization for Everyday Driving: A Dynamic-programming-based Solution, *Transactions on Intelligent Transportation Systems*, 15(6), 2491-2505.

Park, S., Rakha, H., Ahn, K., and Moran, K. (2011). Predictive Eco-cruise Control: Algorithm and Potential Benefits, *Forum on Integrated and Sustainable Transportation Systems (FISTS)*, June 29-July 1, Vienna, Austria.

Rahimi-Eichi, H. and Chow, M. (2014). Big-data Framework for Electric Vehicle Range Estimation, *IECON 2014 - 40th Annual Conference of the IEEE Industrial Electronics Society*, Oct. 29-Nov. 1, Dallas, USA, pp. 5628-5634.

Zahid, T., Xu, K., Li, W., Li, C., and Li, H. (2018). State of Charge Estimation for Electric Vehicle Power Battery using Advanced Machine Learning Algorithm under Diversified Drive Cycles, *Energy*, 162, 871-882.



백 돈규 (Donkyu Baek)

- 정회원
- 한양대학교 전자컴퓨터공학과 공학학사
- 한국과학기술원 전기전자공학부 공학석사
- 한국과학기술원 전기전자공학부

공학박사

- (현재) 충북대학교 전자정보대학 전자공학부 조교수
- 관심분야: SoC설계, 임베디드 시스템, 사이버 물리시스템