Path Planning and Collision Avoidance of a Self-Driving Vehicle Based on Real-time Optimization

Kyoungchul Kong*

Department of Mechanical Engineering
Sogang University
Seoul, Korea 121-742
kckong@sogang. ac .kr
http: //robot ics . sogang . ac . kr

Abstract. A real-time path-planning algorithm for self-driving vehicles is introduced in this paper. The proposed algorithm is capable of collision avoidance, as well as calculation of an optimal path to reach destination. While conventional path planning algorithms such as the A-star algorithm and the Dijkstra algorithm are computationally demanding and do not consider dynamic obstacles, the proposed algorithm determines an optimal path considering both the destination and obstacles in real-time. Once the optimal direction is determined, the proposed method also calculates the desired steering angle and the desired vehicle speed for the control of driving and steering actuators.

Keywords: Unmanned vehicle, cost function, collision avoidance, real-time path planning

1 Introduction

Automotive engineers have intensively investigated mechanical technologies for improved driving performance and fuel-efficiency in past century. In recent years, their research focus has been moved to the development of intelligent algorithms to enhance the perceptibility, safety, and comfort of drivers[1]. In particular, an automatic driving technology is receiving a great attraction[2]. For example, an automatic parking system enables safe parking in a small space, and a cruise control system reduces the fatigue of drivers from long distance driving. More recently, a fully automated self-driving technology is being developed. It is expected that the self-driving technology will enable people with disabilities and elderly people to drive vehicles with guaranteed safety. Many automotive researchers and vehicle manufacturers are investigating on the self-driving technology, and several remarkable technologies have been introduced at the Urban Challenge competition held by the Defense Advanced Research Projects Agency (DARPA) of the United States[3] [6].

* Corresponding author.

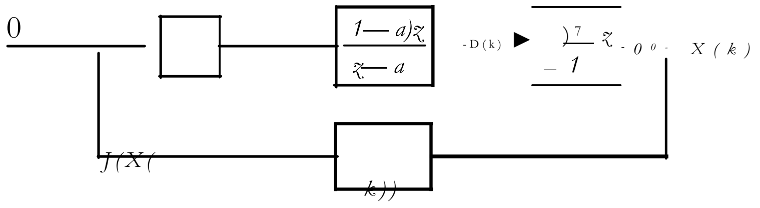

Fig. 1: Real-time path planning algorithm.

For the automatic driving of a vehicle, a path-planning method is necessary to calculate the desired speed and steering angle for following traffic lanes and avoiding obstacles. For the path-planning of navigation systems, the Dijkstra algorithm and the A-star algorithm are widely used. About three quarters of the self-driving vehicles participated in the Urban Challenge in 2007 utilized either the Dijkstra or the A-star algorithm[4]. Although the Dijkstra and A-star algorithms are convenient in planning a path in stationary situations, however, their desired performance is not necessarily guaranteed in dynamic environments[5]. Namely, these algorithms are computationally demanding, and thus they are not suitable if the desired path is to be generated/updated in every sampling instance.

In this paper, a real-time optimization method is applied to the path-planning problem. A cost function is defined considering dynamic obstacles (e.g., cutting-in and cutting-out vehicles and pedestrians) and stationary obstacles (e.g., traffic lanes), as well as the destination. Then the proposed method calculates an optimal path such that the cost function is minimized. Since the cost function is updated considering the dynamic environments, the calculated path is updated continuously. Once an optimal path is obtained for a given cost function, only a short period of time horizon is taken into the motion control algorithm of the vehicle, the principle of which is similar to the model predictive control method. Since the calculated path may include a non-differentiable point, which causes discomfort to a driver, a velocity control algorithm is applied in between the proposed path-planning method and the motion control algorithms of the vehicle.

2 Real-Time Path Planning

2.1 Real-Time Optimization

The real-time optimization algorithm applied to calculate an optimal path is as shown in Fig. 1. In the figure, z is an one-step advance operator, and $J(X(_k))$ is a cost function defined on a two-dimensional map. $J(X(_k))$ is designed such that it is minimal at $X(k) = Xd,$ where $X(k)$ and Xd are the current location of the vehicle and the destination, respectively. Given the location of the vehicle at the kth sampling instance, the desired position at the next step (i.e., the $(k + 1)$th sampling instance) is obtained as

$$X(k+l) = X(k) — 7D(k) \qquad (^1)$$

where k is the time index, and $X(k)$ is the current location (i.e., the latitude and longitude) of a vehicle, i.e.

$$X(k) — [x(k)\ y(k)1^T\ E\ \mathrm{R}^2 \qquad (2)$$

In Eq. (1), $D(_k)$ E R^2 is a vector related to the gradient of the cost function and is defined as

$$D(k) = aD(k\_1) + (1 — a)VJ(X(k)) \qquad (3)$$

where $VJ(X(_k))$ is the gradient of $J(X(_k))$ at $X(_k)$, i.e.

$$A\ X\ (k))\ \frac{-[mx_{(k)})\ 0J(x(k))1^T}{a\ x\quad a\ y\quad i} \qquad (4)$$

In practice, however, the cost function is not necessarily analytically differentiable, and the gradient of the cost function in Eq. (4) is numerically calculated by the Euler's approximation.

In Eq. (3), a is a real-valued constant in the range of 0 and 1, and is related to the steering angle. If a is close to zero, the path-planning algorithm selects an optimal path (i.e., the desired location of the next step) without consideration of physical limitations of the vehicle, and thus the calculated path may be difficult to follow due to the drastic change in the steering angle. On the other hand, if a is close to one, the direction of calculated path changes smoothly, but the optimality in the calculated path is lost.

The parameter 7 E JR in Eq. (1) represents the distance between the current location and the location at the next step (i,e., $X(k)$ and $X(_{k+1})$). If the calculation period is constant, 7 is proportional to the travel velocity of the vehicle, i.e.

$$7(k) = V(k)dt \qquad (5)$$

where dt is the calculation period of the path-planning algorithm, and $v(_k)$ E JR is the current travel velocity of the vehicle.

In order to make $v(_k)$ converge to zero when the vehicle reaches its destination, the travel velocity is controlled by a feedback control algorithm shown in Fig. 2. The parameters K_1 and $K2$ shown in the figure are coefficients that determine the convergence speed of $v(_k)$, and a saturation function is introduced into the loop to avoid excessive velocity and acceleration during driving. Due to the saturation function, the vehicle seeks the optimal path within the preprogrammed ranges of speed and acceleration. The block labeled as *Norm* in Fig. 2 is defined to

$$e(k) — IIXd — x(k)\ ||\ E$$ [redacted] (6)

where $X\,d$ is the destination, i.e., the desired position of the vehicle.

The acceleration of the vehicle is determined according to the magnitude of an error in position of the vehicle. Namely, if the current location of the vehicle is

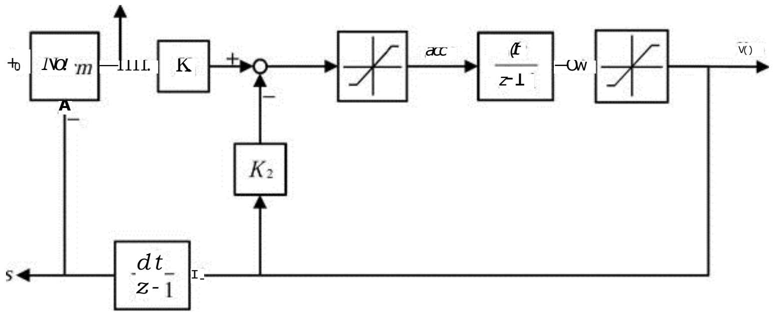

Fig. 2: Travel velocity control algorithm

far from the destination, it may fully accelerate in order to reach the destination in a short time. On the other hand, if the vehicle is near the destination, it may decelerate for the precise control of vehicle position. When the vehicle is on a straight line path such that $X(k) = [s(k)\ 0 1^T$, the transfer function from Xd to $X(k)$ becomes

$$\frac{X}{Xd} \quad \frac{K1dt^2}{z^2 + K2(dt - 2)Z \pm Kidt^2 - K2dt \pm 1} \tag{7}$$

Notice that the transfer function in Eq. (7) does not possess
any root outside of the unit circle and has the DC-gain of one, and thus the position of the vehicle (i.e., $X(k)$) asymptotically converges to the destination, Xd, if Xd is stationary. Consequently, the speed of the vehicle, v(k), also converges to zero as $X(k)$ approaches Xd.

2.2 Cost Function

For the proposed real-time optimization algorithm to be applied to the path-planning problem, a cost function must satisfy the following conditions:

1. $J(X(k)) > 0$ for all $k > 0$, where $J(X(k)) = 0$ only if $X(k) = Xd$ for some k.
2. $VJ(X(k))$ is finite for all $X(k)$.

The cost function may consist of a function for seeking the direction to the destination and function(s) for avoiding obstacles. A second-norm function is appropriate for seeking the direction to the destination, i.e.

$$J1(X(k)) = a1IIX(k) \tag{8}$$

where a_l is a constant that determines the slope of the overall cost function. $J_1(X(k))$ represents the distance between the current location and the destination, and has its minimum at $X(k) = Xd$. Namely, the path-planning algorithm reduces the distance from the current location to the destination by minimizing $Jl(X(k))$.

If there exist obstacles, a penalizing function should be added to the cost function such that $J(X_{(k)})$ becomes large at the location of the obstacles. Many functions can be utilized for this purpose; in this paper, a fractional function and a modified convex function are proposed as penalizing functions. The fractional function is defined as

$$j_2\ (X\ (k)) = \frac{a2}{11X\ (k) - X_{o}(k)II} \qquad (9)$$

where $X_{0(k)}$ E $I^{:2}$ represents the location of an obstacle in the absolute coordinate system. The parameter a_2 adjusts the range of the effect of the obstacle; if a_2 is large, the vehicle may detour the obstacle keeping a large distance, and if is is small, then the obstacle would not affect the path-planning result. It should be noted that if $X\,d$ is far from $X_{0}\ (k)$, $J_2\ (X\,d)$ is close to zero, and thus the effect due to the penalizing function is negligible at the location of the destination. If the obstacle is placed near the destination, however, the cost function around the destination may be significantly distorted, and it may be impossible to generate a path to reach the destination. Another issue is that $J_2\ (X_{(k)}) \text{—>} 00$ as $X_{(k)}$ $X_{0(k)}$, which makes it difficult to implement the proposed path-planning algorithm in practice.

An alternative penalizing function is

$$J3\ (X\ (0) = \max\ ("^{2 3}_{7.}\ (r - II\ X(k) - X0(k)\ II), 0) \qquad (10)$$

where r is the range of the effect of an obstacle. Note that if the distance between $X_{(k)}$ and $X_{o(k)}$ is greater than r, $J_3(X_{(k)})$ is zero; the penalizing function does not affect the cost function around the destination if II $Xo(k)$ — Xd II > r. The parameter a_3 is to determine the magnitude of $J_3(X_0(0)$, i.e., the magnitude of the penalizing function at the location of the obstacle. Unlike $J_2\ (X_{(k)})$ in Eq. (9), $J3(X_{(k)})$ is finite for all $X_{(k)}$'s.

The final cost function is a combination of the functions in Eqs. (8)—(10), i.e.

$$J(X_{(k)}) = Ji\ (x(k)) + J3(X_{(k)}) \qquad (11)$$

Notice that the final cost function satisfies the two conditions, i.e., $J(X_{(k)}) > 0$ for all $k > 0$, and $VJ(X_{(k)})$ is finite for all $X_{(k)}$. If there exist multiple obstacles, the penalizing function $J_3\ (X_{(k)})$ may be defined for each obstacle.

3 Conclusions

This paper introduced a path-planning algorithm that seeks a path to a destination considering obstacles. Since the proposed method was based on the real-time optimization algorithm that continuously decreases the value of an arbitrary cost function, the proposed algorithm sought an optimal path even in dynamically changing environments, such as moving obstacles and traffic lanes. The cost function was defined as a combination of multiple functions, including a main function that directs to the destination and penalizing functions that reflect the characteristics of obstacles. In addition to the path-planning algorithm, the proposed method included a control algorithm for the travel velocity of the vehicle, such that the physical limitations of a vehicle could be considered in the path-planning procedure.

Since the proposed method seeks an optimal path in every sampling instance and is not computationally demanding, it is effective for self-driving vehicles. The field-test of the proposed method will be the topic of a following paper.

Acknowledgments. Parts of this paper were previously presented at the 2011 KSME Annual Conference. This research was supported in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012R1A1A1008271) and in part by Sogang Research Fund (201214004).

References

1. Na, J., Park, S.B., Kim, J.H.: Development of Path Planning Algorithm for Unmanned Vehicle Navigation System: Conference on Geospatial Information Systems Annual Conference, pp. 83-88, (2009)
2. Jung, B.M., Yeo, I.J., Cho, C.S.: Autonomous Traveling of Unmanned Golf-Car using GPS and Vision system: Journal of the Korean Society for Precision Engineering, vol. 26, no. 6, pp. 74-80 (2009)
3. Cho, K.H., Ahn, D.J., Kim, G.S., Kim, Y.I.: Development of a Path Planning Algorithm for Unmanned Vehicle: Journal of Korean Society of Industrial Application, vol. 14, no. 2, pp. 53 57 (2011)
4. Woo, H.J., Park, S.B., Cho, K.H., Kim, J.H.: Research of the Optimal Local and Global Path Planning for Unmanned Ground Vehicle: KSAE Annual Conference, pp. 988-992 (2009)
5. Hart, P.E., Nilsson, N.J., Raphael, B.: A Formal Basis for the Heuristic Determination of Minimum Cost Paths: IEEE Transactions of Systems Science and Cybernetics, vol. SSC-4, no. 2, pp. 100-107 (1968)
6. The DARPA Urban Challenge, http://archive.darpa.mil/grandchallenge